PC-based Systems
ELOP II

Resource type manual

Edition 0611

HIMA

HIMA Paul Hildebrandt GmbH
Industrial Automation

HI 800 119 BEA

Caution

The safety-related systems as described in this manual can be used for several different purposes. The knowledge of regulations and the technically perfect transfer carried out by qualified staff are prerequisites for the safe installation, start-up and for the safety during operation and maintenance of the safety-related systems.

In case of unqualified interventions into the automation devices, de-activating or bypassing safety functions, or if advices of this manual are neglected (causing disturbances or impairments of safety functions), severe personal injuries, property or environmental damage may occur for which we cannot take liability.

Important Notes

All HIMA products mentioned in this manual are protected with the HIMA trade-mark. As not differently noted down this is possibly also valid for other mentioned manufactueres and their products.

All listed modules are CE certified and meet the requirements of the EMC Guideline of the European Community.

All technical statements and data in this manual have been worked out very carefully, and effective checks and inspections have been applied. This manual may however contain flaws or typesetting errors. Therefore HIMA does not offer any warranties nor assume legal reponsibility nor any liability for the possible consequences of any errors in this manual. HIMA would appreciate being informed on possible errors.

The technology is subject to changes without notice.

Delivery Conditions

For our deliveries and services apply the “General Conditions for Delivery of Products and Services of the German Electrical Industry “ - edition January 2002 -, resp. the “Conditions of Delivery for System Software and Peripheral Devices for the HIMA Automation System“ (e. g. programmer units, printers, screen monitors). The products of this price list are subject to the valid export regulations.

Eventual complaints can be recognized only when we are being notified within 14 days after receipt of the merchandize.

The prices shown in a special list are valid ex works, packing charges excluded. The prices are subject to change.

Contents

1 Creating a configuration 1
1.1 Properties: Buses 1
1.1.1 Editing HIBUS 2
1.1.2 PES Masters 7

2 Configuring the resource 10
2.1 Instancing a program type to the resource 11
2.2 Configuring a resource 12
2.2.1 Editing Cabinet layout 13
2.2.2 Resource properties 19

3 Resource-related functions in the Editor 35
3.1 Assigning system variables 35
3.2 Creating Events 36
3.3 Variables for transmission with HIPRO 37
3.4 Variables for transfer to external systems 40

4 Communication between HIMA PESs 42
4.1 Protocols 42
4.1.1 Safety-related transfer: HIPRO-S 42
4.1.2 Non-safety related transfer: HIPRO-N 42
4.2 HIBUS Hardware 42
4.3 Redundancy 42
4.4 Configuring data exchange 43
4.4.1 Safety-related data transfer: HIPRO-S 43
4.4.2 Non-safety related data transfer: HIPRO-N 44
4.4.3 Variables declaration 44
4.5 Configuring data interchange to ELOP slaves 45
4.5.1 Configuration in ELOP 46
4.5.2 Bus configuration 46
4.5.3 Resource configuration 47
4.5.4 Variables declaration 47

5 Communicating with external systems 49
5.1 BUSCOM Communication 49
5.2 3964R Communication 51

6 Generating the program code 52
6.1 Code generator 52
6.2 Compiling and loading PES masters 52
6.3 Loading the program into the controller 54
6.4 Download/Reload 57
6.4.1 OS Download 57

7 Online Test 58
7.1 Starting the online test 58
7.2 OLT Window 58
7.2.1 Operating Buttons 59
7.2.2 Program Structure Tree 60
7.3 Functions in the ONLINE Test 60
7.3.1 OLT Fields 61
7.3.2 Forcing I/Os 61

8 Documentation in the configuration 65
8.1 Documentation functions of the configuration 65
8.1.1 Bus docu 65
8.1.2 PES Master Docu 66
8.1.3 CFG Docu 67
8.2 Documentation functions of the resource 68
8.2.1 CRF Docu 68
8.2.2 RES Docu 70
8.2.3 RES Docu (generated) 71
8.2.4 RES Docu (loaded) 71

9 Safety related functions 73
9.1 Building an application 74
9.1.1 Create the logic diagrams 74
9.1.2 Code generation 74
9.1.3 Download 74
9.1.4 Function check of the PES 74
9.2 Modification of an application 75
9.2.1 Modification of logic diagrams 75
9.2.2 Code generation 75
9.2.3 Download 76
9.2.4 Function check of the PES 76

10 HIMA Standard Blocks 77

11 Resource Types 79
11.1 Current Resource Types 79
11.2 Former Resource Types 80
11.2.1 H41q 80
11.2.2 H41 and H51 80

12 Subracks 81
12.1 Current subracks 81
12.2 Old subracks 82

13 Modules 83
13.1 Current modules 83
13.2 Old modules 84

14 Supported IEC Functions and Data Types 85
14.1 Supported data types 85
14.2 Supported functions and function blocks 86

15 System Variables 90
15.1 READ System variables of the BOOL type 91
15.1.1 I/O Error 91
15.1.2 HIBUS.Resource-Name.error 91
15.1.3 SYSTEM.Force individual switch outputs 91
15.1.4 SYSTEM.Force individual switch inputs 91
15.1.5 SYSTEM.Force main switch outputs 91
15.1.6 SYSTEM.Force main switch inputs 91
15.1.7 SYSTEM.Prohibited access 91
15.1.8 SYSTEM.Single-channel 91
15.1.9 SYSTEM.normal 91
15.2 WRITE System variables of the BOOL type 91
15.2.1 I/O Error acknowledgment 91
15.2.2 SYSTEM.Logic emergency stop 92
15.3 READ System variables of the UINT/WORD type 92
15.3.1 HIBUS.Resource-Name.Receive counter 92
15.3.2 SIO Receive Counter 92
15.3.3 I/O.Error code 1st I/O bus 94
15.3.4 I/O.Error position 1st I/O bus 94
15.3.5 I/O.Error code 2nd I/O bus 94
15.3.6 I/O.Error position 2nd I/O bus 95
15.3.7 SYSTEM.RAM/EPROM 95
15.3.8 SYSTEM.Runversion 95
15.3.9 SYSTEM.Codeversion 95
15.3.10 SYSTEM.Number of prohibited accesses 95
15.3.11 SYSTEM.Errormask1 96
15.3.12 SYSTEM.Errormask2 96
15.3.13 SYSTEM.Errorcode 97

Configuration

1 Creating a configuration

Configurations are the highest-level structuring tools, and group together the PESs (Programmable Electronic Systems) of a complete system or subsystem. For each of these PESs they contain a resource that complies with IEC 61131-3. A configuration can be visualized as a bus system to which the PESs are connected. Configurations contain:

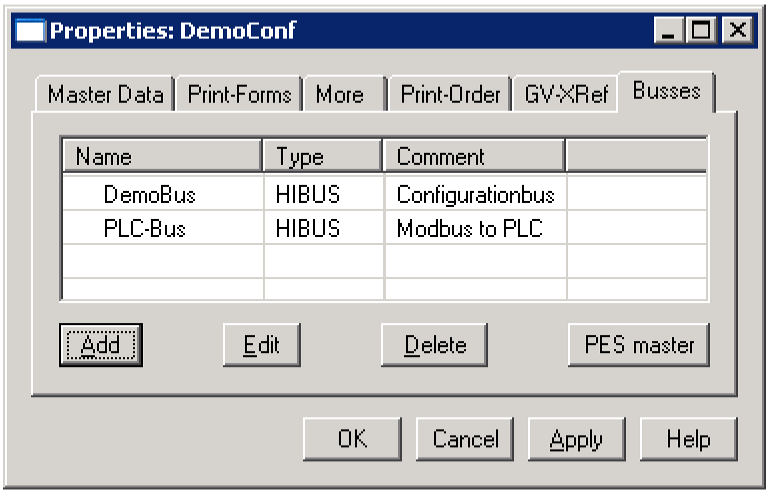
- Resources
- Global variables
- Libraries
- Structuring folders
- Connections

1.1 Properties: Buses

Once you have defined the resources of your configuration you can identify the main buses and their communication parameters. You can then access the bus parameters when configuring the serial interfaces of the resources.

Note:
You will need at least one bus for communicating between the PADT (Programming and Debugging Tool = programming unit) and the resources for loading and for online operations!

To define a bus, click on ´Properties´ in the context menu of the configuration.



Name	Type	Comment	
DemoBus	HIBUS	Configurationbus	
PLC-Bus	HIBUS	Modbus to PLC	

Configuration properties window, ´Buses´ tab

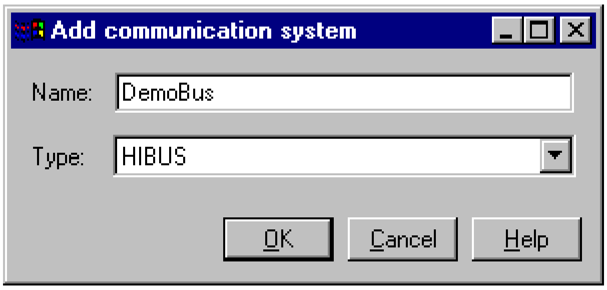
Previously defined buses are listed in this dialogue box. Columns in the index display the main bus data in table form:

- Name: Name of the bus
- Type: Type of communication system (currently HIBUS only)
- Commentary: any comments on the bus

Command buttons

You can add new buses, edit and delete existing buses and open a dialogue box to compile and download the PES masters of all buses.

- ´Add´: Add a communication system
  Give the bus a unique name and select the type of communication system (only the ´HIPRO´ type is currently available). Confirm your choice with ´OK´ to open a dialogue box as described below under ´Edit´.



Name: DemoBus

Type: HIBUS

Dialogue box ´Add communication system´

- Edit: Edit a bus configuration
  Mark the bus you want to edit and click on ´Edit´. You will see a dialogue box in which you can enter comments about the bus, define the bus stations and set up the bus communication parameters (See “Editing HIBUS” on page 2.).
- Delete: Delete a bus configuration
  Mark the bus you want to delete and click on ´Delete´. The bus will be deleted without a confirm prompt.
- PES Masters: Compile and download PES masters
  you will see a dialogue box listing all the defined PES masters for compiling and downloading (See “PES Masters” on page 7.).

1.1.1 Editing HIBUS

HIBUS is used to transmit data between individual HIMA PESs and for communication between the PESs and the PADT (Programming and Debugging Tool: programming unit in IEC, the PC with ELOP II) or with external systems.
The protocols used for communication between individual PESs are based on the master/slave principle, and several masters can be connected to a bus as well as several slaves. If more than one master is connected to a bus then they share control over the bus cyclically. Each station is identified by a user-defined bus station number (BSN). Up to 31 logic bus stations can work on a bus, whether as slaves or as masters.

Note:
The redundant central modules of a PES are addressed with a bus station number (BSN).

The following can be used as master systems:

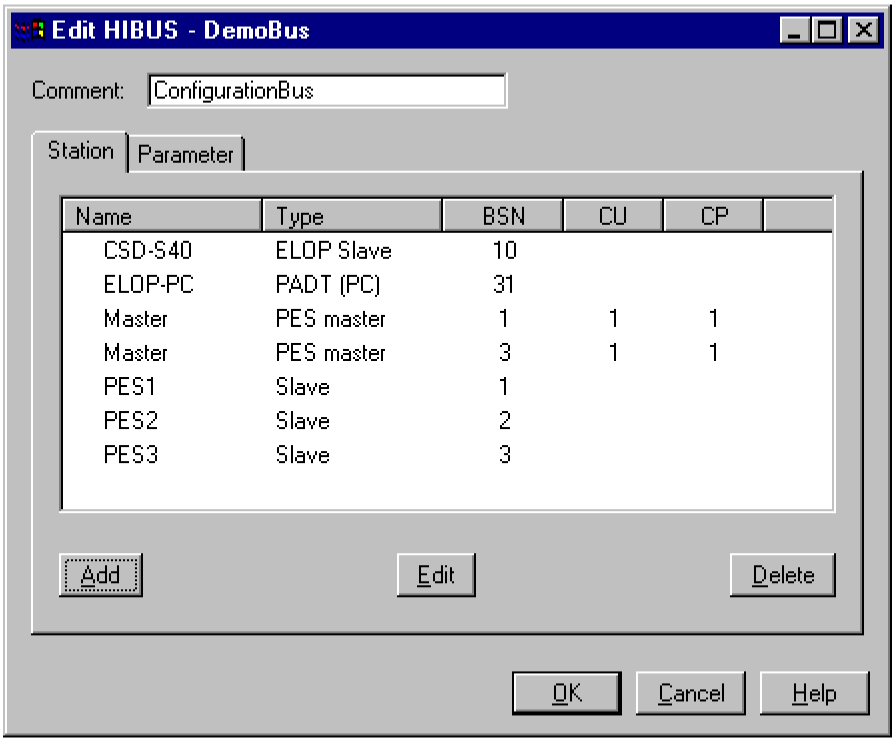
- Programming units (PADT)
- Coprocessor modules (CM) in the H51 or H51q systems as PES masters

The following can be used as slave systems:

- HIMA-PES type A1, H11, H41 and H51 programmed with ELOP
- HIMA-PES type A1, H11, H41, H51, also H41q and H51q programmed with ELOP II

When you add or edit a bus you will see a dialogue box in which you can enter a description of the HIBUS, the stations and the bus communication parameters. The dialogue box offers ´Stations´ and ´Parameters´ tabs for this purpose.

Stations tab



Comment: ConfigurationBus

Name	Type	BSN	CU	CP
CSD-S40	ELOP Slave	10		
ELOP-PC	PADT (PC)	31		
Master	PES master	1	1	1
Master	PES master	3	1	1
PES1	Slave	1		
PES2	Slave	2		
PES3	Slave	3		

Dialogue box ´Edit HIBUS´, ´Stations´ tab

The ´Stations´ index lists previously defined bus stations. Columns in the index display the main bus station data in table form:

- Name: Name of the bus station
- Type: Type of bus station
- BSN: Bus station number
- CU: Assigned central module of the PES master
  1 left CU, 2 = right CU
- CM: Coprocessor module in which the PES master runs
  1.3 = first to third CM according to the slot next to the CU

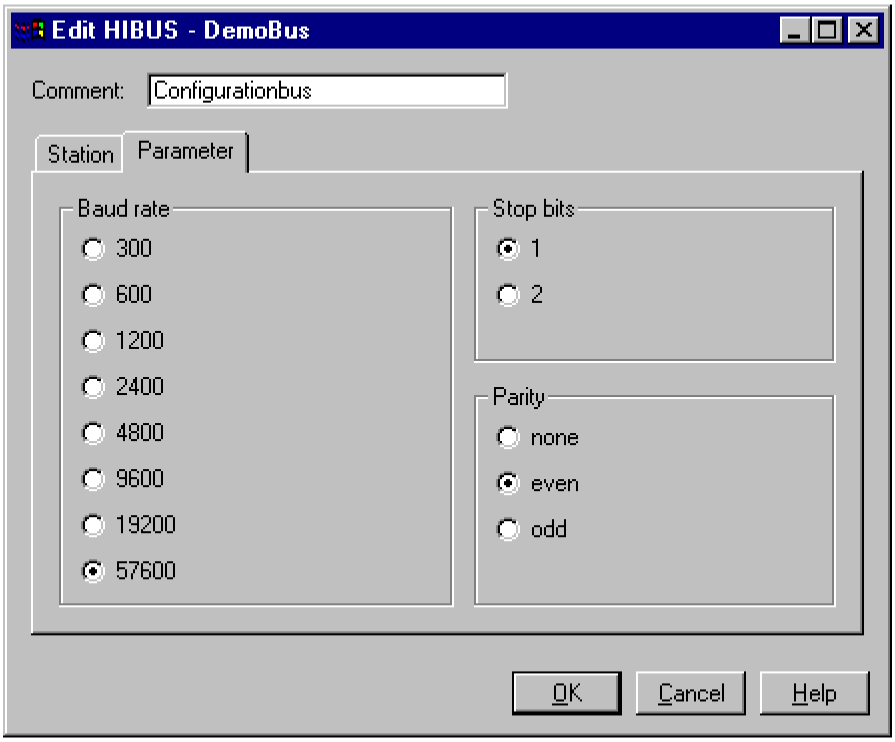
Command buttons

You can add new bus stations and edit or delete existing stations.

- Add: Add a bus station
  You will see a dialogue box as described under ´Edit´.
- Edit: Edit a bus station
  Mark the bus station you want to edit and click on ´Edit´. You will see a dialogue box in which you can enter the name and type of the bus station and any parameters. See “HIPRO Stations” on page 5.
- Delete: Delete a bus station
  Mark the bus station you want to delete and click on ´Delete´. The bus station will be deleted without a confirm prompt.

Parameters tab

The ´Parameters´ tab in the ´Edit HIBUS´ dialogue box is used to define the communication parameters of the bus.



Comment: Configurationbus

Baud rate

○ 300
○ 600
○ 1200
○ 2400
○ 4800
○ 9600
○ 19200
● 57600

Stop bits

● 1
○ 2

Parity

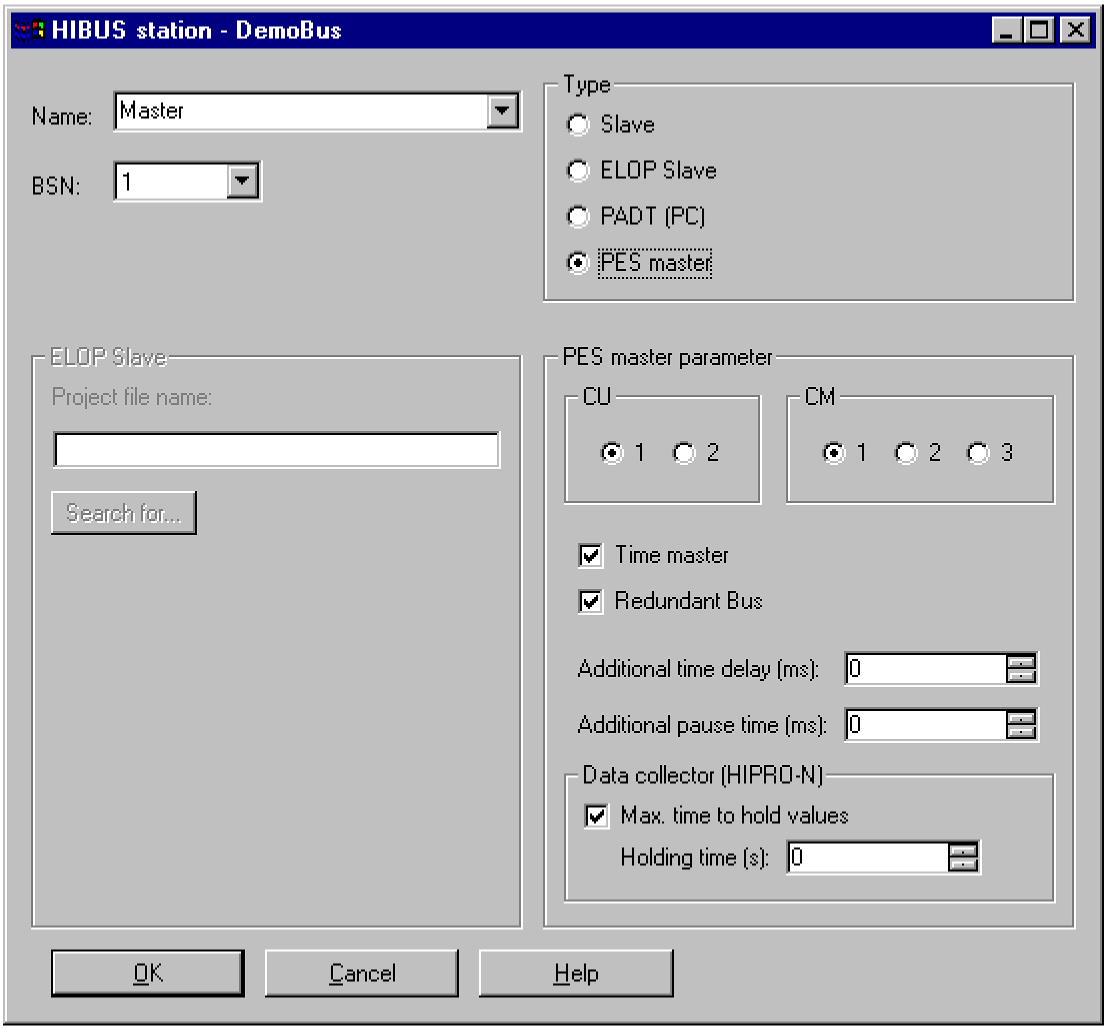
○ none
● even
○ odd

Dialogue box ´Edit HIPRO´, ´Parameters tab

You can set the baud rate, the number of stop bits and the parity of the serial bus communication here. These parameters can then be accessed when the serial interfaces of the bus stations are configured. Subsequent changes to bus parameters will be automatically entered and can be made centrally.

1.1.1.1 HIBUS stations

Clicking on the ´Add´ or ´Edit´ command button in the ´Edit HIBUS´ dialogue box opens another dialogue box in which you can enter the name, type and bus station number (BSN) of the station.



HIBUS station - DemoBus

Name: Master

BSN: 1

Type

○ Slave

○ ELOP Slave

○ PADT (PC)

● PES master

ELOP Slave

Project file name:

PES master parameter

CU: ● 1 ○ 2

CM: ● 1 ○ 2 ○ 3

☑ Time master

☑ Redundant Bus

Additional time delay (ms): 0

Additional pause time (ms): 0

Data collector (HIPRO-N)

☑ Max. time to hold values

Holding time (s): 0

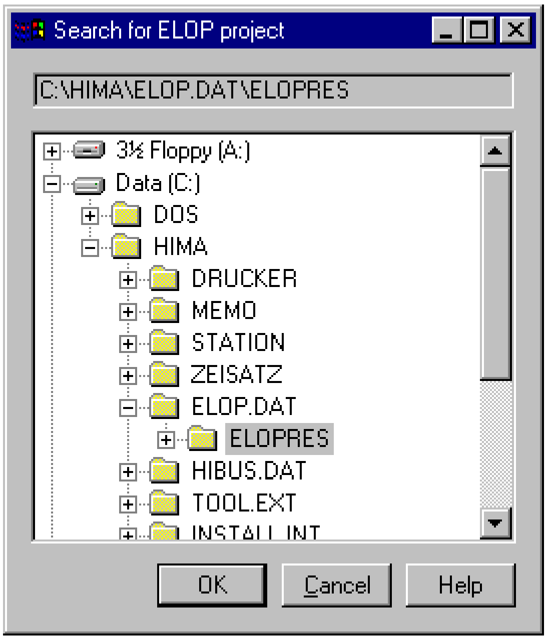
´HIBUS stations´ dialogue box

- Type: Type of the bus station
- Slave: PES programmed with ELOP II
- ELOP Slave: PES programmed with the ELOP firmware (See “Configuring data interchange to ELOP slaves” on page 45.)
- PC master: PADT
- PES masters: Coprocessor modules (CM) in the H51 or H51q systems as PES masters
- Name: Name of the bus station
  The name of a previously defined resource can be selected from the list box for ´slave´ type stations. For bus stations of the ´PC master´ or ´PES master´ type you must enter the name yourself.

Note:
The name may not be more than 8 characters long and should begin with a letter. Many special characters such as ´umlauts´ cannot be mapped in the controller and are therefore replaced by question marks.

- BSN: Bus station numbers
  station number for PESs is set by hardware switches on the central module and must be the same as the number entered here. The BSN must be unique.
- ELOP Slave: Project file name of the ELOP Slave
  the path and file name of the ELOP project are entered here for ´ELOP Slave´ type bus stations. You can open a dialogue box to select the project folder by clicking on the

´Search...´ command button:



´Search for ELOP project´ dialogue box

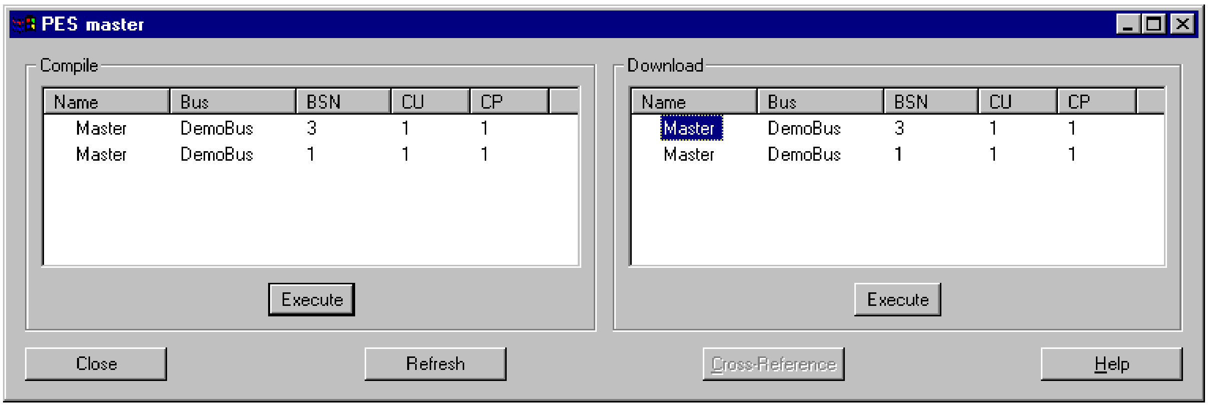
Projects created with the ELOP firmware are usually stored in the [Drive]:\HIMA\ELOP.DAT folder.

- PES master parameters: Parameters for ´PES master´ type bus stations
- CU: Central module which the PES master is next to.
  1 left CU, 2 = right CU
- CM: Coprocessor module containing the PES masters.
  1.3 = first to third CM according to the slot next to the central module
- Time master: Activate time master function
  PES master can synchronize the time of all PESs connected to the bus by periodically transmitting the time of the central module assigned to it to all other bus stations. Here you can define whether the created PES master will be a time master
- Redundant bus:
  The PES master uses a reserve bus to handle data traffic if the normal bus fails
- Increase time delay (ms):
  For communications over modem or satellite links this setting can be increased to allow for the delay due to the carrier
- Increase time interval (ms):
  For communications over modem or satellite links this setting can be used to increase the interval between receiving the last transmission and starting the next one
- Restricted holding time of settings on call failure:
  Here you can define the response of the PES master if it cannot export data from this controller because a call to a PES has failed. This setting is only active when the HIPRO-N protocol is used (PES master as data centre).
  If this control box is highlighted, the values of the failed controller are held by the PES master for all the other PESs in the data interchange for the preset time (in seconds). When the holding time times out the values are written to FALSE or 0 if no new call to the failed controller has been made in that time.
  If the control box is not highlighted then values are frozen until a new call is made.

Note:
If you want the target controllers to detect when values are frozen then this must be included in the design, e.g. by transmitting a constantly changing signal.

1.1.2 PES Masters

PES masters are coprocessor modules of the H51 and H51q systems which have a program for controlling the transfer of data between the PESs (slaves). A PES master is always assigned to a PES and can service up to 31 slaves.
You can open a dialogue box for compiling the PES master programs and downloading the compiled programs to the coprocessor modules by using the ´PES masters´ option in the context menu of the configuration, or by clicking on the ´PES masters´ command button in the ´Buses´ tab in the properties window of the configuration.



Compile:

Name	Bus	BSN	CU	CP
Master	DemoBus	3	1	1
Master	DemoBus	1	1	1

Download:

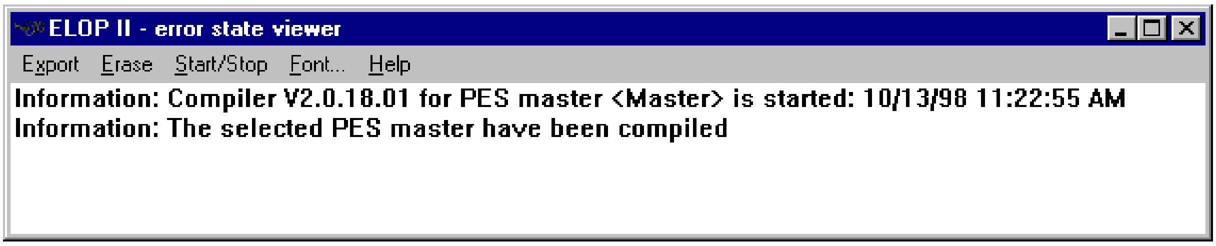
Name	Bus	BSN	CU	CP
Master	DemoBus	3	1	1
Master	DemoBus	1	1	1

PES-Master

Caution! If a PES master has to be loaded, all control panels (CPs) and online tests (OLTs) have to be closed.

You can compile and download PES masters, update the data in the list boxes and open a cross-reference window in which the variables transmitted by the PES masters are documented.

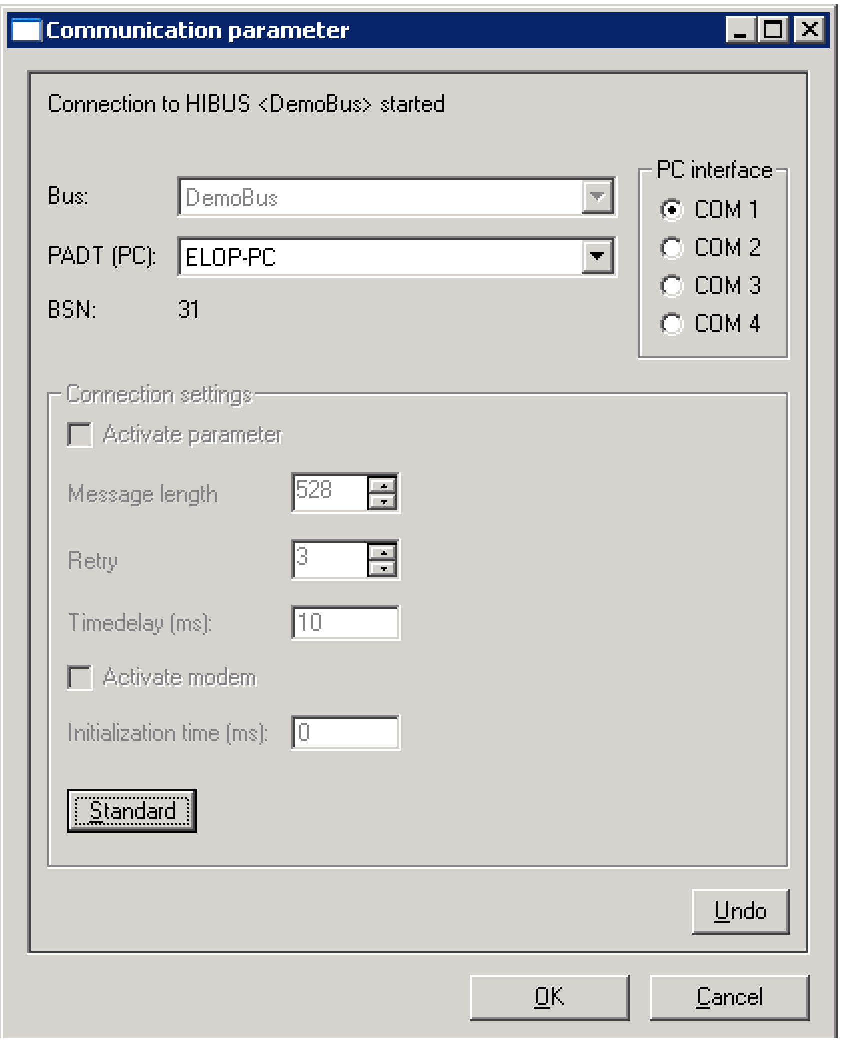
- ´Execute´ under the ´Compile´ list box: Compile the PES masters
  Highlight the PES master(s) you wish to compile and click on ´Execute´. The selected PES masters will be compiled. The error status display opens and shows you how the compile routine is progressing.



```
Information: Compiler V2.0.18.01 for PES master <Master> is started: 10/13/98 11:22:55 AM
Information: The selected PES master have been compiled
```

Error status display after running the PES master code generation

- ´Execute´ under the ´Download´ list box: Transfer the PES masters to the coprocessor modules.
  Highlight the PES master(s) you wish to load and click on ´Execute´. You will see a dialogue box in which you can enter the PADT communication parameters.



Connection to HIBUS <DemoBus> started

Bus: DemoBus

PADT (PC): ELOP-PC

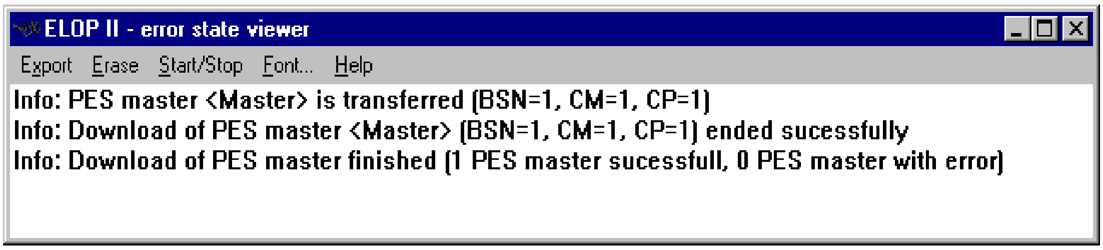
BSN: 31

PC interface
● COM 1
○ COM 2
○ COM 3
○ COM 4

Connection settings
☐ Activate parameter
Message length 528
Retry 3
Timedelay (ms): 10
☐ Activate modem
Initialization time (ms): 0

Dialogue box ´Communication parameters

Select the name of the required PC master from the list box and define the serial interface of the PC that will be used for transmitting data. Click on ´OK´ to transmit the selected PES masters. The error status display opens and shows you how the PES master download is progressing.

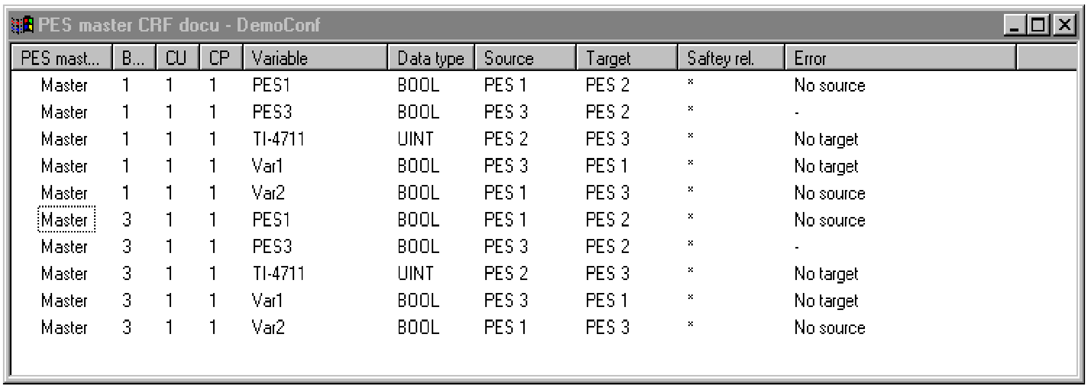


```
Info: PES master <Master> is transferred (BSN=1, CM=1, CP=1)
Info: Download of PES master <Master> (BSN=1, CM=1, CP=1) ended sucessfully
Info: Download of PES master finished (1 PES master sucessfull, 0 PES master with error)
```

Error status display after running the PES master download

When the download is complete you will see a dialogue box containing information about the status of the transfer.

- ´Update´: Update data in both lists
  Action is possible when the code generator for a resource is running while the ´PES master´ dialogue box is open. Instead of quitting the dialogue box and opening it again when the code generator run is complete, you can call ´Update´.
- ´Cross-reference´: Output documentation of the data transmitted by the PES masters.



PES master CRF docu - DemoConf

PES mast...	B...	CU	CP	Variable	Data type	Source	Target	Saftey rel.	Error
Master	1	1	1	PES1	BOOL	PES 1	PES 2	×	No source
Master	1	1	1	PES3	BOOL	PES 3	PES 2	×	-
Master	1	1	1	TI-4711	UINT	PES 2	PES 3	×	No target
Master	1	1	1	Var1	BOOL	PES 3	PES 1	×	No target
Master	1	1	1	Var2	BOOL	PES 1	PES 3	×	No source
Master	3	1	1	PES1	BOOL	PES 1	PES 2	×	No source
Master	3	1	1	PES3	BOOL	PES 3	PES 2	×	-
Master	3	1	1	TI-4711	UINT	PES 2	PES 3	×	No target
Master	3	1	1	Var1	BOOL	PES 3	PES 1	×	No target
Master	3	1	1	Var2	BOOL	PES 1	PES 3	×	No source

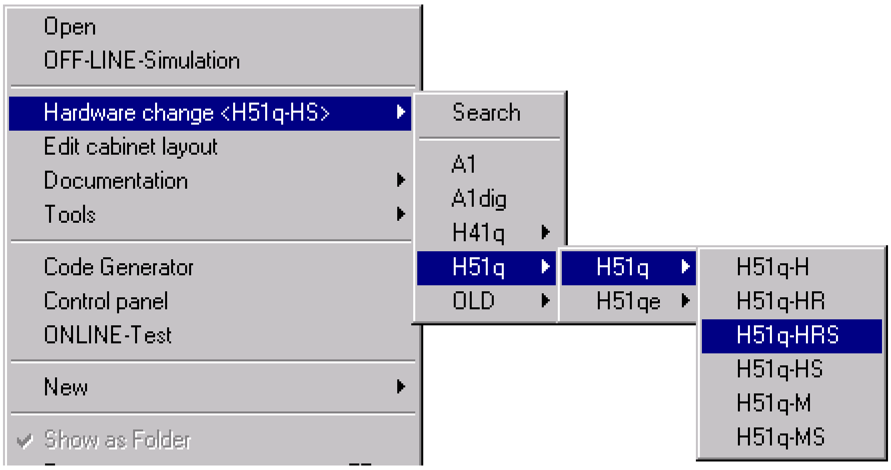
CRF-Documentation

2 Configuring the resource

Resources structure the configuration of a system in the next stage. Each resource symbolizes a PES and contains one or more programs that are run under the controller by zero or more tasks. Resources are created with the context menu of the configuration (´New´ option in the ´Resource´ submenu) and they can contain the following objects:

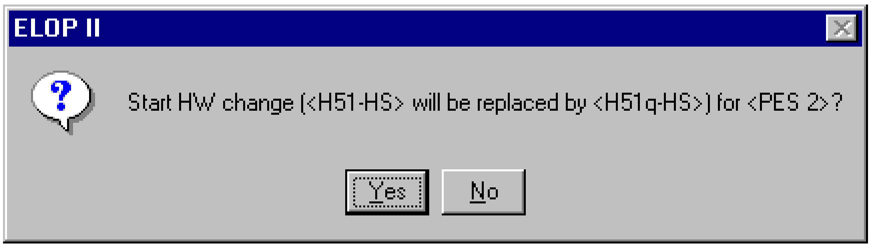
- Program instances
- Type instances
- Tasks
- Global variables
- Libraries
- Structuring folders
- Connections

A newly generated resource is not yet standardized (“generic“) and does not yet possess any special, manufacturer-specific properties. You only define the hardware of the target PES when you assign a resource type. In this way the resource is given the properties and functions of a certain controller type which are associated with the resource type.
To assign a resource type, proceed as follows:
Click on the resource with the right-hand mouse button and point the mouse pointer to ´RT Assignment´ in the context menu which now appears. Now click on the desired resource type in the following submenu. Alternatively you can assign a resource type interactively by selecting the ´Search´ submenu option:



Dialogue box RT assignment

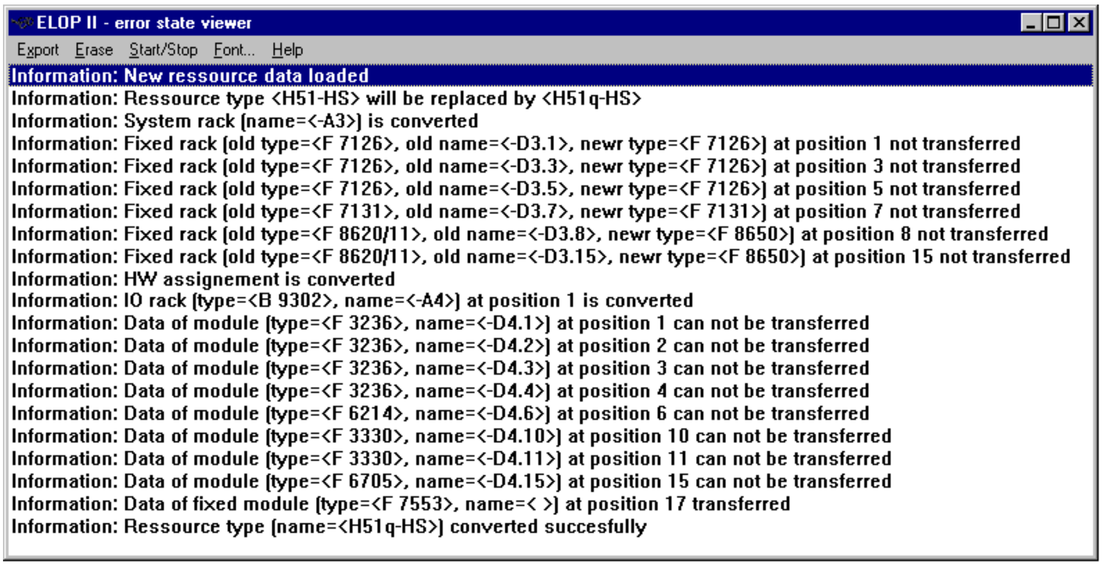
Once a resource type has been assigned the ´RT Assignment´ option in the context menu of the resource changes to the option.
You can change an assigned resource type at any time:
Click on the resource with the right-hand mouse button and point the mouse pointer to ´Hardware change <*Resource Type*>´ in the context menu which now appears. Now click on the desired new resource type in the following submenu, or start the ´RT assignment´ dialogue from the ´Search´ submenu option. You are prompted to confirm your choice before the action is executed:


ELOP II

Start HW change (<H51-HS> will be replaced by <H51q-HS>) for <PES 2>?

Dialogue box ´Start HW change ...´

When you confirm by clicking on ´Yes´ the error status display opens and shows you how the hardware change is progressing.



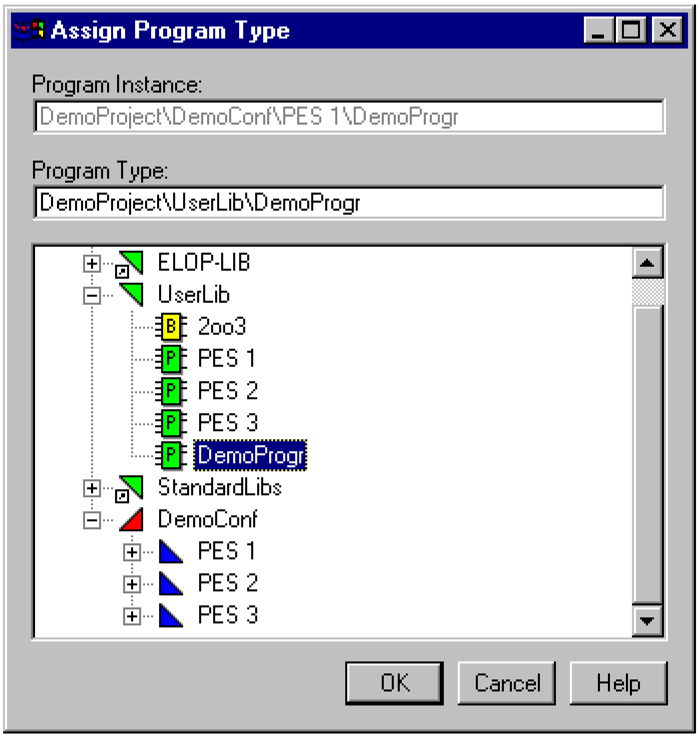
```
Information: New ressource data loaded
Information: Ressource type <H51-HS> will be replaced by <H51q-HS>
Information: System rack (name=<-A3>) is converted
Information: Fixed rack (old type=<F 7126>, old name=<-D3.1>, newr type=<F 7126>) at position 1 not transferred
Information: Fixed rack (old type=<F 7126>, old name=<-D3.3>, newr type=<F 7126>) at position 3 not transferred
Information: Fixed rack (old type=<F 7126>, old name=<-D3.5>, newr type=<F 7126>) at position 5 not transferred
Information: Fixed rack (old type=<F 7131>, old name=<-D3.7>, newr type=<F 7131>) at position 7 not transferred
Information: Fixed rack (old type=<F 8620/11>, old name=<-D3.8>, newr type=<F 8650>) at position 8 not transferred
Information: Fixed rack (old type=<F 8620/11>, old name=<-D3.15>, newr type=<F 8650>) at position 15 not transferred
Information: HW assignement is converted
Information: IO rack (type=<B 9302>, name=<-A4>) at position 1 is converted
Information: Data of module (type=<F 3236>, name=<-D4.1>) at position 1 can not be transferred
Information: Data of module (type=<F 3236>, name=<-D4.2>) at position 2 can not be transferred
Information: Data of module (type=<F 3236>, name=<-D4.3>) at position 3 can not be transferred
Information: Data of module (type=<F 3236>, name=<-D4.4>) at position 4 can not be transferred
Information: Data of module (type=<F 6214>, name=<-D4.6>) at position 6 can not be transferred
Information: Data of module (type=<F 3330>, name=<-D4.10>) at position 10 can not be transferred
Information: Data of module (type=<F 3330>, name=<-D4.11>) at position 11 can not be transferred
Information: Data of module (type=<F 6705>, name=<-D4.15>) at position 15 can not be transferred
Information: Data of fixed module (type=<F 7553>, name=< >) at position 17 transferred
Information: Ressource type (name=<H51q-HS>) converted succesfully
```

Error status display after running a hardware change

When the conversion is complete you will see a dialogue box containing information about the status of the resource type change.

2.1 Instancing a program type to the resource

Program instances execute the functionality that is defined in an associated program type declaration on the PES of your resource. They do not move or copy anything - they merely create an instance (an example) of the program type declaration. After instancing you can continue editing the program instance by double clicking. Whereas the program type declaration describes the functionality of a specific controller, in the program instance the properties of the target PES can be accessed as well.
Several instances of a program type can be created in different resources. To create a program instance select the ´New´ option from the context menu of the appropriate resource, then select the ´Program instance...´ submenu. A dialogue box now opens in which you can select the program type to which to assign the program instance.



Assign Program Type

Program Instance: DemoProject\DemoConf\PES 1\DemoProgr

Program Type: DemoProject\UserLib\DemoProgr

Dialogue box ´Assign program type´

To assign the program type, highlight the assignment target in the project structure window, and the name will be immediately matched under the ´Program type´ input box. The program type will be instanced to the resource when you confirm with ´OK´.

Note:
IEC 61131-3 allows a resource to contain a number of program instances that are run by zero or more tasks in the PES. This option is not available with the current PESs of HIMA, i.e. only one program instance can be contained.

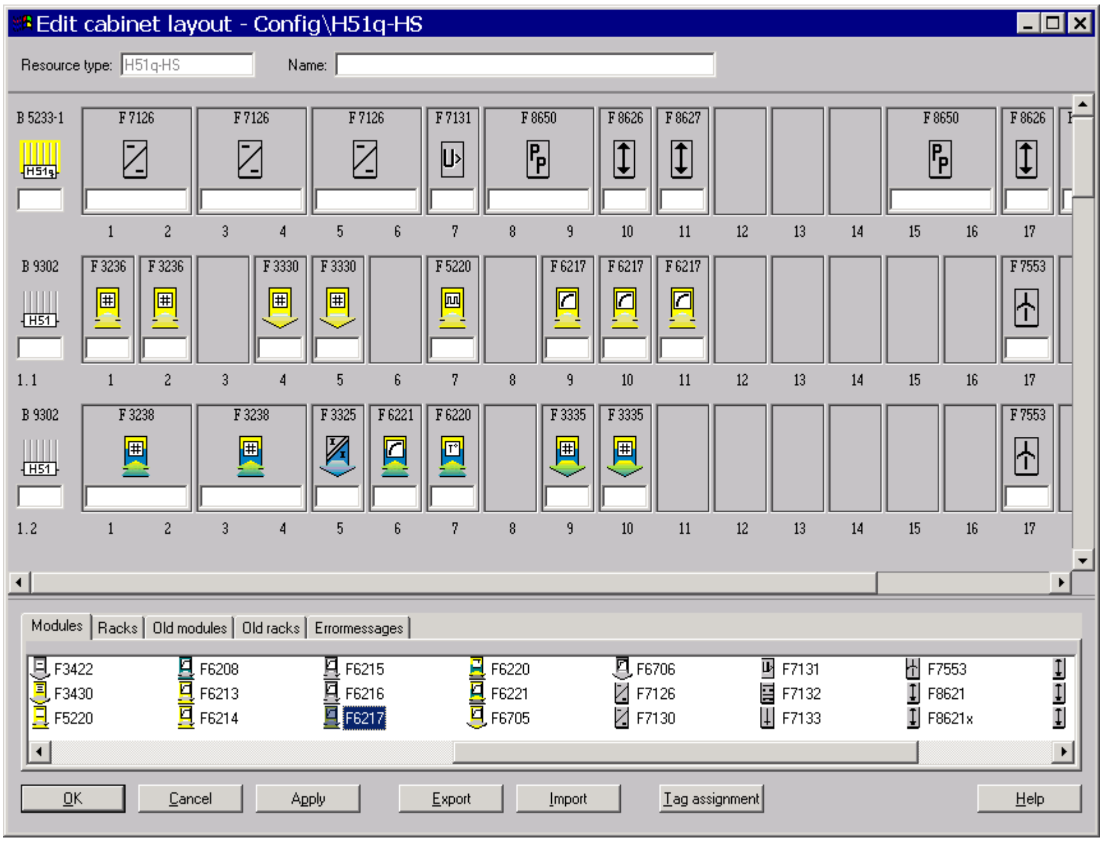
You can assign another program instance to a resource at any time by deleting the program instance in the resource using its context menu, and then repeating the steps described above.

2.2 Configuring a resource

If the resource has been assigned a resource type, you can access the special properties and functions of the Programmable Electronic System (PES) which it defines.
You can use the ´Edit cabinet´ option in the context menu of the resource to define which modules are installed in your PES, and where. You can also set up parameters for the modules you use (See “Editing Cabinet layout” on page 13.).
You can configure your resource with the ´Properties´ option in the context menu of the resource. The tabs described below are displayed for the HIMA PESs as well as the default properties of all objects. See “Resource properties”.

2.2.1 Editing Cabinet layout

To define the modules of your PES click on the resource with the right-hand mouse button. Now click on the ´Edit cabinet´ option in the context menu which now appears. A dialogue box opens with a schematic of the cabinet and you can define and parameterize the modules to be used.



Edit cabinet layout - Config\H51q-HS

Resource type: H51q-HS Name:

Dialogue box ´Edit cabinet´

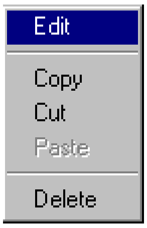
The subracks are shown at the top of the dialogue box. A symbol on the left indicates the subrack type, and on the right are the subrack slots. All available modules and subracks are listed in the lower part of the dialogue box and can be selected by index pages.
You can enter comments about the PES in the ´Name´ input box on the first line next to the resource type, e.g. you can enter the name of the cabinet where the PES is installed. The subrack at the top symbolizes the central unit. Wildcard symbols for 16 I/O subracks are shown under the central unit for the H51 and H51q systems. The numbers beneath the wildcard describe the subrack addressing (e.g. 1-3: I/O bus/ cubicle 1, I/O subrack 3) that must be set using the coding switch on the connecting module. The numbers 1...21 under the module slots correspond to the physical arrangement of the slots in the subrack. To use an I/O subrack, select a subrack from the ´Subracks´ or ´Old subracks´ index page and drag it to a subrack wildcard symbol with the ´Drag&Drop´ feature. You can enter a commentary (e.g. a location code) about the selected subrack in the input box beneath it.

Note:
The subrack address you assign here will also be used to display errors in the central module display. To assist fault-finding you should use addresses that correspond to the physical code number of the subrack in the cubicle.

You can now select the modules you wish to use from the ´Modules´ or ´Old modules´ index page and use Drag&Drop to move them to a module slot in the subrack. Module information can also be entered in the commentary box.
If you drag a module to an assigned slot, the data of the original module will be accepted (if possible). In this way you can change the type of a previously placed module without having to enter new data.

Context menu

You can open a context menu for the corresponding object above a commentary box, a positioned subrack or a positioned module.



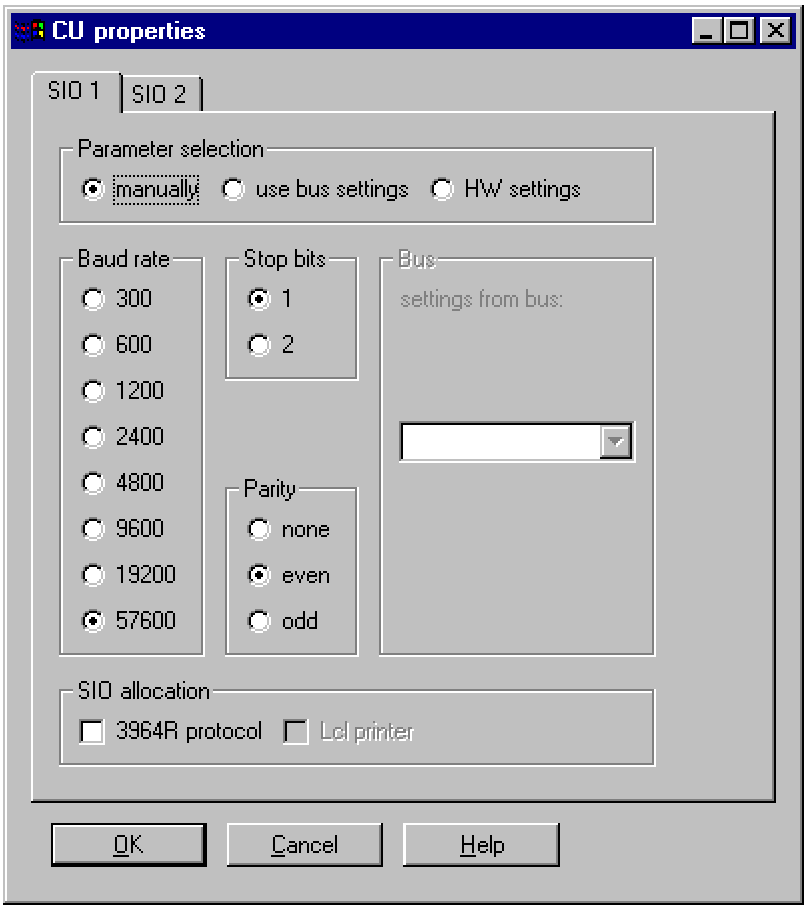
Context menu

Editing modules

Double clicking on a central module, a coprocessor module or an I/O module, or selecting the ´Edit´ option from the context menu of the module opens a module-specific dialogue box for you to configure the module´s parameters.

2.2.1.1 CU properties, CM properties

A dialogue box opens for central or coprocessor modules for you to configure the communication parameters of the module´s two serial interfaces.



Parameter selection
● manually ○ use bus settings ○ HW settings

Baud rate
○ 300
○ 600
○ 1200
○ 2400
○ 4800
○ 9600
○ 19200
● 57600

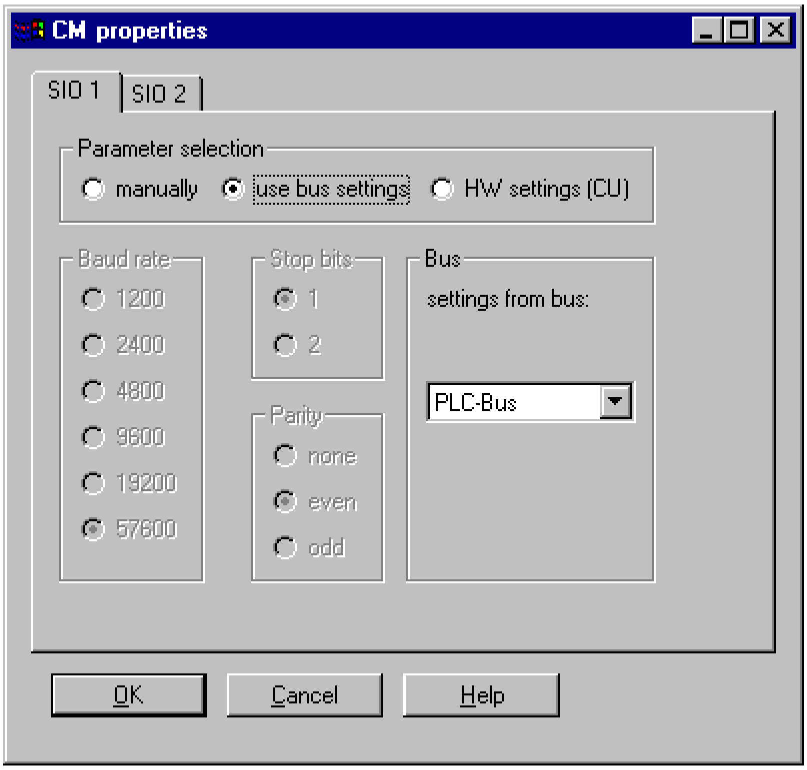
Stop bits
● 1
○ 2

Parity
○ none
● even
○ odd

Bus
settings from bus:

SIO allocation
□ 3964R protocol □ Lcl printer

Dialogue box ´CU properties´



Parameter selection
○ manually ● use bus settings ○ HW settings (CU)

Baud rate
○ 1200
○ 2400
○ 4800
○ 9600
○ 19200
● 57600

Stop bits
● 1
○ 2

Parity
○ none
● even
○ odd

Bus
settings from bus:
PLC-Bus

Dialogue box ´CM properties´

There are three modes of parameter selection for configuring a serial interface:

- Manual:
  Set up all the parameters yourself (baud rate, stop bits, parity)
- By bus configuration:
  ´Enter parameters from bus´ selection box is activated. When you select a bus from the list, the system will enter the parameters of that bus, and any subsequent changes to the bus configuration will then be entered automatically.
- Hardware presetting:
  Serial interface is not configured by ELOP II, and the parameters set by switch S1 on the central module are used instead.

A special communication protocol for the interfaces of the central module can also be selected using the small control box in the ´SIO reservation´ box:

- 3964R communication:
  Reserved interface communicates as a slave system with protocol 3964R.
- Lcl printer:
  Printer for logic plan controlled logging (Lcl) is connected to this interface. This is only available for central module interface 2.

Note:
In a redundant central unit the left and right hand halves of the unit can be regarded as a single logic system so they have the same parameters. It is therefore irrelevant whether the parameters on the left hand CU/CMs or right hand CU/CMs are configured.

2.2.1.2 Editing the tag name

Each I/O module consists of inputs and/or outputs to which you can assign symbolic names (tag names). As soon as you have assigned tag names to the I/O modules, variables of the program instance can be assigned to those tag names. In this way you can define I/O addresses for the variables

There are two basic methods for assigning addresses:

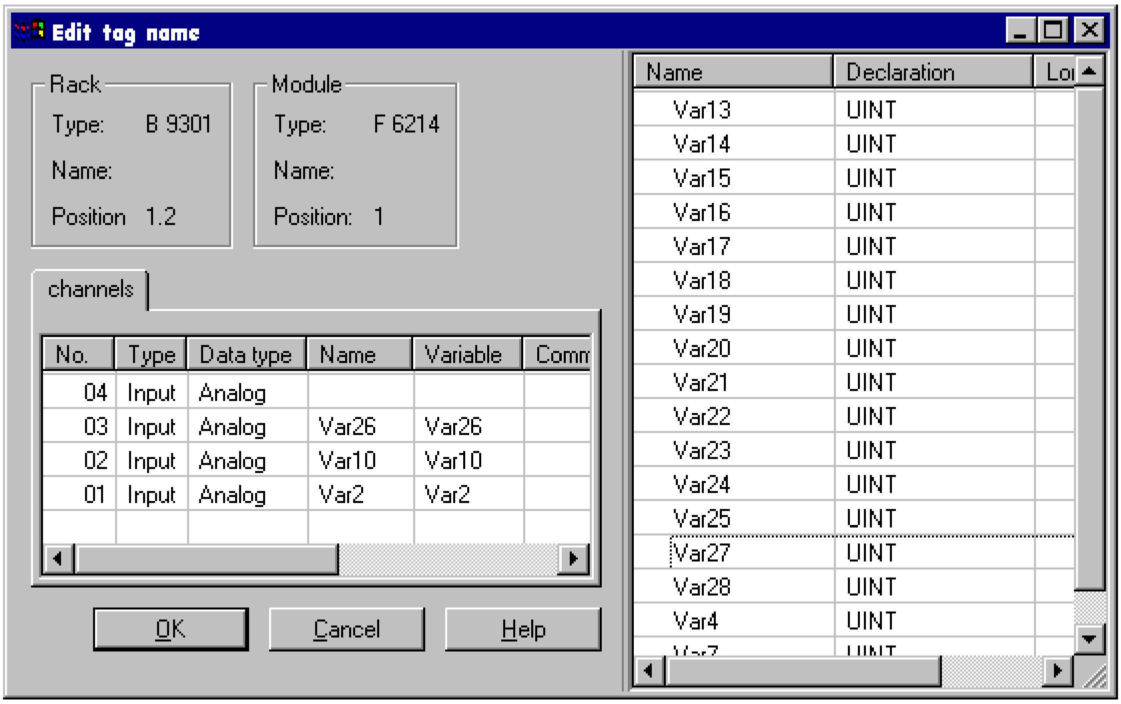
First define the I/O modules of the PES, then create the logic functionality

- Edit the cabinet
- Define the I/O modules to be used
- Create tag names for the I/O channels
- Create the program type
- Instance the program type to the resource
- Edit the program instance
- Assign the defined tag names to the variables of the program instance

First create the logic functionality of the PES, then define the I/O modules

- Create the program type
- Instance the program type to the resource
- Edit the cubicle
- Define the I/O modules to be used
- Define tag names for the I/O channels and assign the defined variables of the program instance to them.

Double clicking on an I/O module or selecting the ´Edit´ option from the context menu of the I/O module opens a module-specific dialogue box for you to assign tag names to the channels of the I/O module.



Rack
Type: B 9301
Name:
Position 1.2

Module
Type: F 6214
Name:
Position: 1

No.	Type	Data type	Name	Variable	Comm
04	Input	Analog			
03	Input	Analog	Var26	Var26	
02	Input	Analog	Var10	Var10	
01	Input	Analog	Var2	Var2	

Name	Declaration	Lo
Var13	UINT	
Var14	UINT	
Var15	UINT	
Var16	UINT	
Var17	UINT	
Var18	UINT	
Var19	UINT	
Var20	UINT	
Var21	UINT	
Var22	UINT	
Var23	UINT	
Var24	UINT	
Var25	UINT	
Var27	UINT	
Var28	UINT	
Var4	UINT	
Var7	UINT	

Dialogue box ´Edit tag name´

This dialogue box contains the individual I/O channels of the module and the tag names which have already been assigned. Details of the subrack and module are listed above the index. Columns in the index display the main data of the I/O channels of the module in table form:

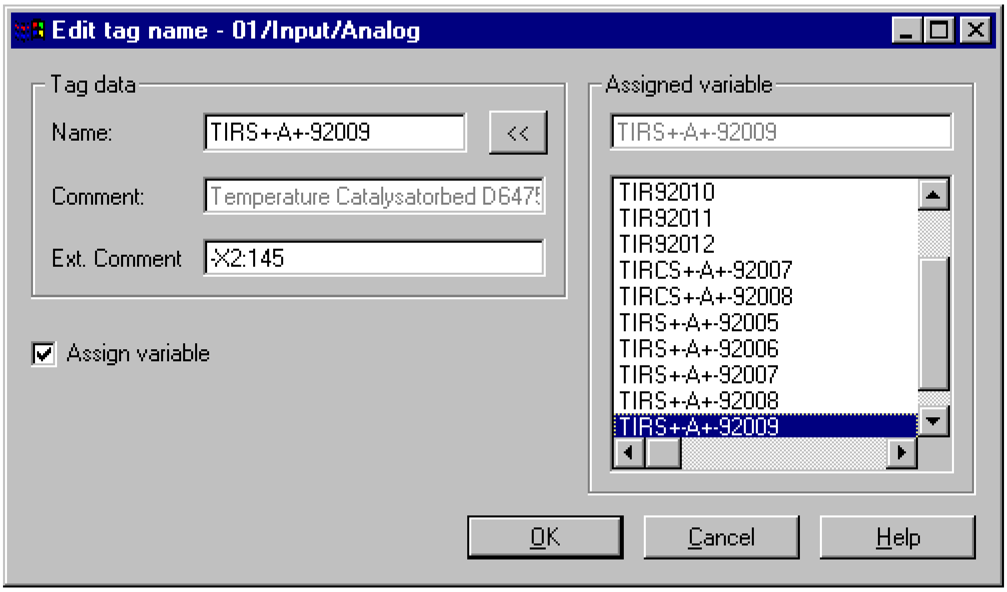
- No.: Channel number of the I/O module
- Type: I/O type of the channel
  ´Input´ or ´Output´
- Data type: The data type of the channel
  ´Digital´ for assigning single-bit variables (variables of the BOOL data type), ´Analog´ for assigning multiple-bit variables (variables of the WORD, INT or UINT data type)

- Name: tag name of the channel
- Variable: assigned variable of the program instance of the resource
- Comment: A commentary on the tag name or long name of the assigned variable
- Ext. comment: any additional commentary

On the right side a variable list whit all not already assigned variables of the program instance of the respective resource is shown. Via the context menu filter and sort command are available.

Editing tag names

A variable from the variable list can be assigned by easily drag&drop the variable to a channel. If a variable was already assigned to this channel, the system overwrite the assignment and prompt to overwrite or keep the tag name.
A second possibility is to double click on a module channel which opens a dialogue box in which you can edit the tag data:



Edit tag name - 01/Input/Analog

Tag data

Name: TIRS+-A+-92009

Comment: Temperature Catalysatorbed D647!

Ext. Comment: -X2:145

☑ Assign variable

Assigned variable

TIRS+-A+-92009

TIR92010
TIR92011
TIR92012
TIRCS+-A+-92007
TIRCS+-A+-92008
TIRS+-A+-92005
TIRS+-A+-92006
TIRS+-A+-92007
TIRS+-A+-92008
TIRS+-A+-92009

Dialogue box ´Edit tag´

You can configure for a module channel an tag name, a commentary and an additional commentary, and assign the I/O channel a variable from the program instance of the resource.

- Name: Enter an tag name

Quick method:

- In the previous dialogue box ´Edit tag name´, highlight the channel number you wish to edit.
- Click in the ´Name´ column of the module channel.
- Enter the tag name in the text box.
- Press ENTER.

Note:
If you use identical symbolic identifiers for the tag names and for the variables of the program instance, then tag names and variables can be assigned automatically.

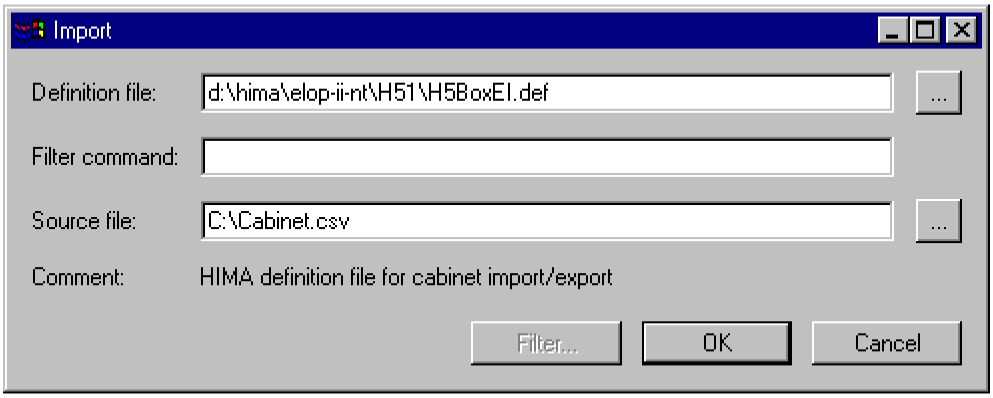
- Commentary: Your comments about the tag name
- Extra commentary: additional comments on the tag name

Quick method:

- In the previous dialogue box ´Edit tag name´, highlight the channel number you wish to edit.
- Click in the ´Extra comment´ column of the module channel.
- Enter your extra comments in the text box.
- Press ENTER.
- Assign variable:
  If a program type is instanced in the resource, then the variables of this program instance can be accessed. Assigning a variable to an tag name defines the I/O address of that variable.
- Click on the ´Assign variable´ control box.
- Double click on a variable name in the list box.
- The variable name will appear in the ´Assigned variable´ box, and the ´Comment´ box displays the long name of the variable.
- Click on command button ´<<´ if you want to use the variable name as the tag name.

2.2.1.3 Import and Export of IO configuration

You can import and export the IO configuration including the allocation with variables via a CSV file. The function is similar to the import and export of variables in POUs. Before importing from a file, the system deletes all IO configuration.



Definition file: d:\hima\elop-ii-nt\H51\H5BoxEI.def

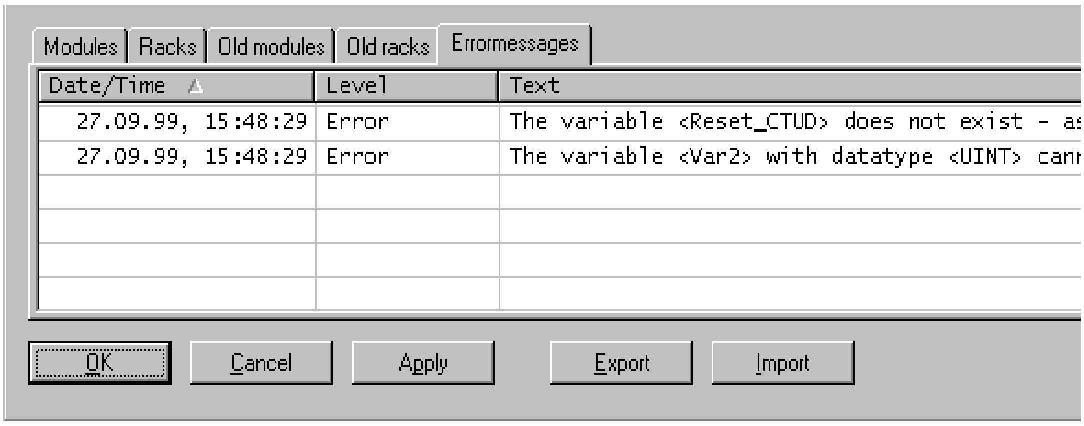
Filter command:

Source file: C:\Cabinet.csv

Comment: HIMA definition file for cabinet import/export

Dialogue box ´Import´

If errors occur during importing the system will list all errors in the register ’error messages’.



Date/Time	Level	Text
27.09.99, 15:48:29	Error	The variable <Reset_CTUD> does not exist - as
27.09.99, 15:48:29	Error	The variable <Var2> with datatype <UINT> cann

Tab ´Error messages´

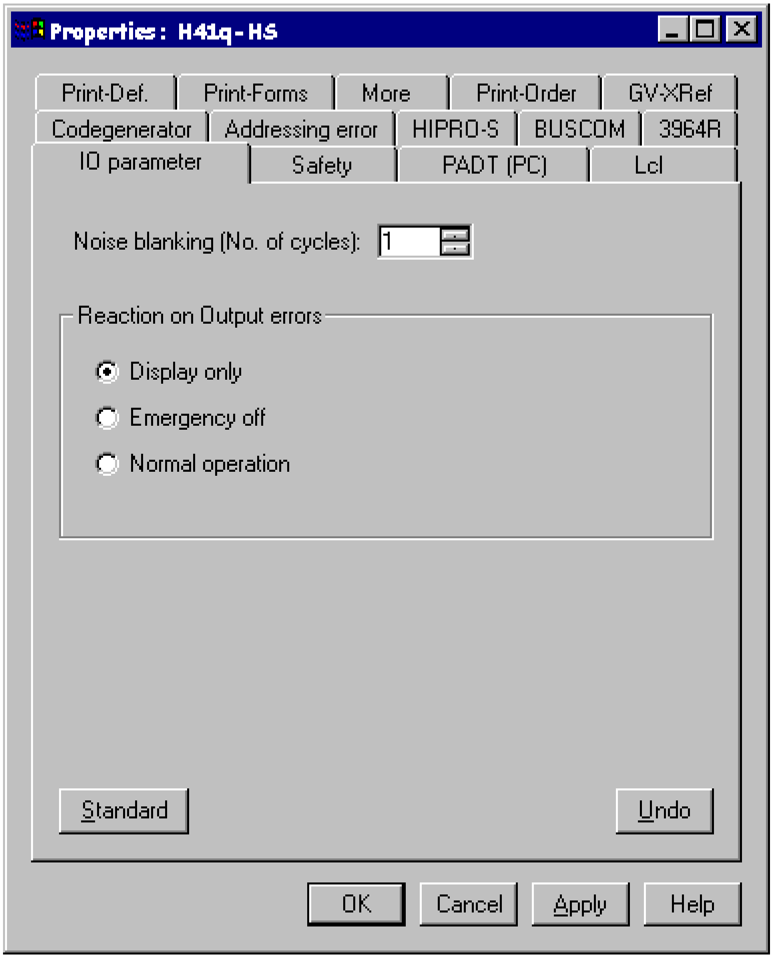
Tag-assignment

The tag assignment displays all program and hardware I/O variables. You can check and refresh the assignment between hardware and program variables after the import of the cabinet layout.

2.2.2 Resource properties

You can configure your resource with the ´Properties´ option in the context menu of the resource. The tabs described below are displayed for the HIMA PESs as well as the default properties of all objects.

2.2.2.1 I/O parameters



Noise blanking (No. of cycles): 1

Reaction on Output errors

● Display only

○ Emergency off

○ Normal operation

Resource properties window, ´IO parameters´ tab

This tab is used to define the response of the PES to I/O faults.

- Noise blanking (cycles):
  The comprehensive tests of all components detect any deviation from the specification. The operating system tolerates transient faults with its integrated noise blanking feature. In addition, this input box defines the number of cycles for which I/O faults will be tolerated. The value 0 activates integrated noise blanking.

Note:
The system limits the number of noise blanking cycles to (safety time/ watchdog time)-2.

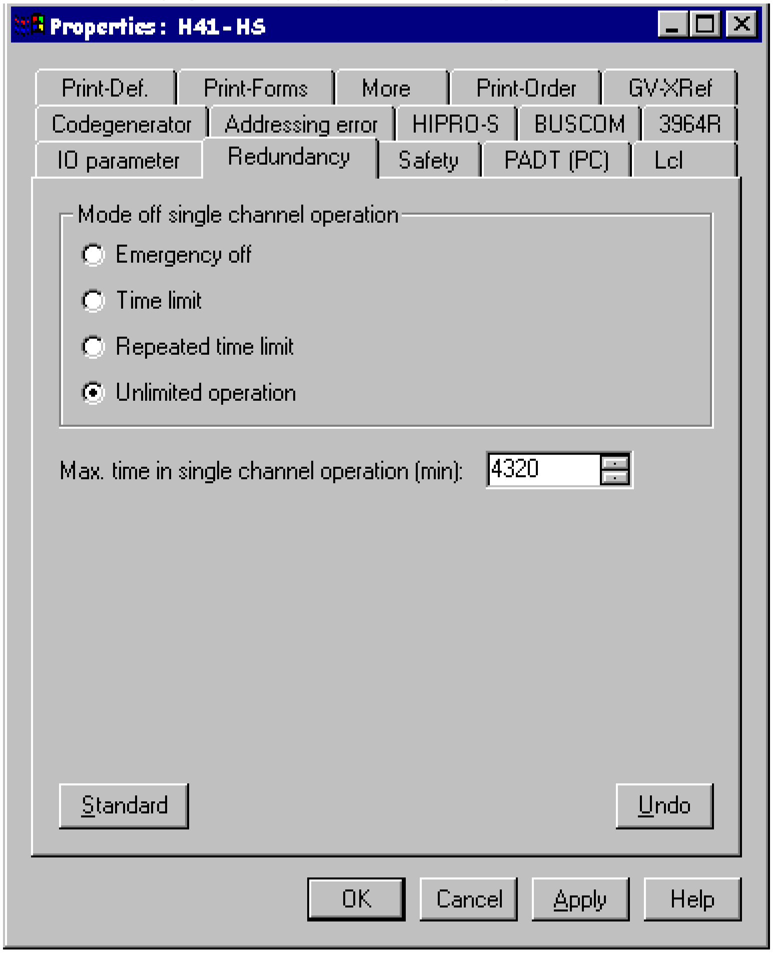
Example	1	2	3
Cycle time	100 ms	200 ms	200 ms
Watchdog time	300 ms	500 ms	500 ms
Safety time	1 s	2 s	1 s
Max. number of blanking cycles	1	2	0*

Table 1: Noise blanking
*Integrated noise blanking is active in this case.

The response of the controller to a fault in a testable output amplifier can be defined by one of 3 different parameters depending on safety and/or availability.

- Display only:
  Faulty modules are shut down by the integral safety shutdown feature. If a module does not permit shutdown, then the shutdown of the subrack by the connection module is not safety-related. When an H8-STA-3 function block is used (see also the description of the function block) all the modules entered on the block are shut down by the integrated safety shutdown if a module is faulty.
- Emergency stop:
  General shutdown (emergency stop) of the PES in the event of a fault in an output amplifier or an I/O bus fault. If the PES has a redundant I/O bus, only the central module which has the fault in its I/O bus shuts down.
- Normal operation:
  Equal ´Display only´, but the connecting module is shut down safety-related by the shutdown of the watchdog signal.

2.2.2.2 Redundancy (H41/H51 systems only)



Mode off single channel operation

○ Emergency off

○ Time limit

○ Repeated time limit

● Unlimited operation

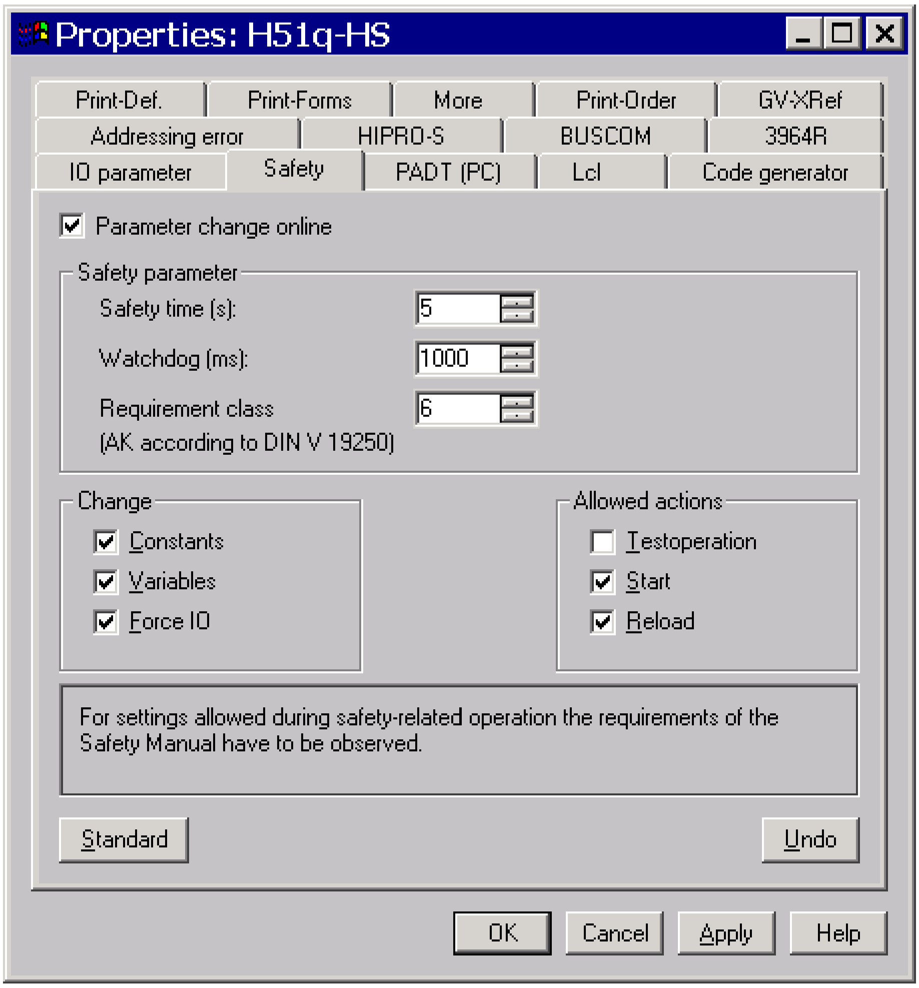
Max. time in single channel operation (min): 4320

Resource properties window, ´Redundancy´ tab

On this tab you can define the response of a redundant PES in the event of failure by a central unit (not for H41q and H51q systems). Depending on the required level of safety and/or availability you can specify the redundancy loss mode by one of four different parameters:

- Emergency stop
  Emergency stop trips (general system shutdown) immediately when central unit fails.
- Timer limit
  One central unit fails, the other central unit runs on for the time in minutes specified in the ´Max. time in single channel operation´ box. When the timer times out a general PES shutdown is performed. If the faulty module is replaced by a functional module in the remaining time, then redundant mode is resumed and there is no further time restriction.
- Repeated time limit
  As for timer mode, but each time the ACK button is pressed on the central module the time in the ´Max. time 1-channel´ box is uploaded again and the shutdown is delayed.
- Unlimited mode:
  Unlimited 1-channel operation.
- Max. time in single channel operation (min):
  n minutes for single-channel central unit operation until general shutdown, valid for ´Time limit´ and ´Repeated time limit´. This time is process-dependent and must be agreed with the accepting authority.

2.2.2.3 Safety



☑ Parameter change online

Safety parameter

Safety time (s):	5
Watchdog (ms):	1000
Requirement class (AK according to DIN V 19250)	6

Change

☑ Constants
☑ Variables
☑ Force IO

Allowed actions

☐ Testoperation
☑ Start
☑ Reload

For settings allowed during safety-related operation the requirements of the Safety Manual have to be observed.

Resource properties window, ´Safety´ tab

This tab is used to define safety-critical parameters and to specify the permitted actions with the PADT during safety-related operation of the PES.

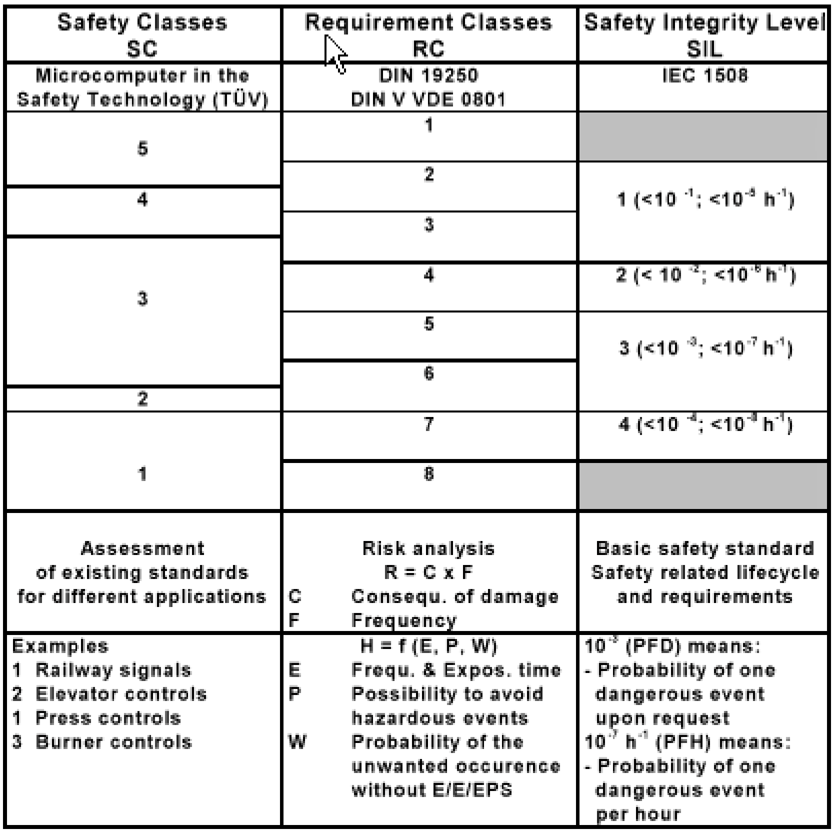
Note:
The assignments that are possible during safety-related operation are not rigidly bound by any particular class of requirements but they must be agreed with the approving authority responsible for each use of the PES.

- ´Parameters change online´:
  The changes to the safety parameters specified on this tab are disabled if the control box is deactivated. In this case a modification requires a download or reload (if available) of the program.

Note: If the change of constants is denied, also these parameters can not be changed.

- Safety time (s) (H41/H51only):
  Self-tests of a PES are divided into foreground and background tests. Foreground tests are used to detect dangerous first faults and are performed within a cycle or within the safety time (in seconds) that is specified in this input box. Background tests are additional test routines used to detect faults that can have adverse safety-technical effects in conjunction with other faults. Background tests are performed in a longer time interval. Shortening the safety time therefore increases the cycle time. With long safety times some tests are spread over several cycles. The safety time must be matched to suit the controlled process.

- Watchdog (ms):
  Input box shows the cycle monitoring time for the system in milliseconds. It must not be more than half the safety time.
- Requirements class (to DIN V 19250):
  The safety-related use of the PES, this input box indicates the safety requirement (requirement classes 0 to 6 to DIN V 19250).



Safety Classes SC	Requirement Classes RC	Safety Integrity Level SIL
Microcomputer in the Safety Technology (TÜV)	DIN 19250 DIN V VDE 0801	IEC 1508
5	1	
4	2	1 (<10 $^{-1}$; <10^{-5} h^{-1})
3	3	
	4	2 (< 10 $^{-2}$; <10^{-6} h^{-1})
	5	3 (<10 $^{-3}$; <10^{-7} h^{-1})
2	6	
1	7	4 (<10 $^{-4}$; <10^{-8} h^{-1})
	8	
Assessment of existing standards for different applications	Risk analysis R = C x F C Consequ. of damage F Frequency	Basic safety standard Safety related lifecycle and requirements
Examples 1 Railway signals 2 Elevator controls 1 Press controls 3 Burner controls	H = f (E, P, W) E Frequ. & Expos. time P Possibility to avoid hazardous events W Probability of the unwanted occurence without E/E/EPS	10^{-3} (PFD) means: - Probability of one dangerous event upon request 10^{-7} h^{-1} (PFH) means: - Probability of one dangerous event per hour

Requirement class 6 is identical with SIL 3 IEC/EN 61508.

- Constants:
  Constants (VAR CONST) and system parameters cannot be changed online by the PADT unless the control box is activated.

Note: If this option is deactivated during Online-Test no further modifications can be made!

- Variables:
  Variables cannot be changed (set) online unless the control box is activated.
- Forcing IO:
  Force main switch and individual switches for the inputs and outputs of the PES cannot be set unless the control box is activated.
- Test mode
  Deactivating the control box disables execution of the following PADT commands:
  Start (if the PES is in RUN mode; the command is always executed in STOP)
  Stop
  Step
  Continue AP

- Hot start/Cold start:
  Deactivating the control box disables execution of all start commands of the PADT if the PES is in RUN mode; the commands are always executed in STOP.
- Reload:
  Reload cannot be executed in the Reload mode when the control box is deactivated.

2.2.2.4 PADT (PC)

You can use this tab to define the communication with the PADT (PC) to the controllers via Ethernet or serial connection.

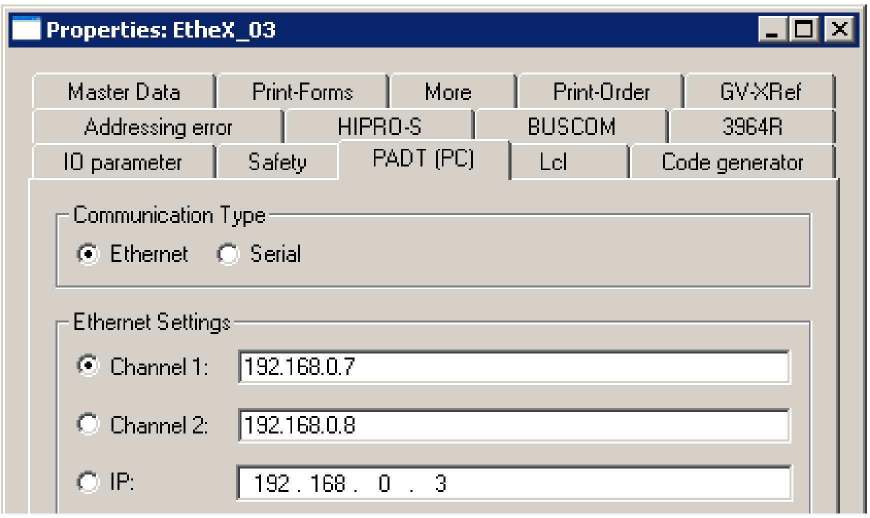
Communication Type

Ethernet:
Select "Ethernet" to establish a ELOP II TCP connection, which makes a fast data exchange possible between a PADT (PC) and the central module F 865x.

Requirements for a ELOP II TCP connection

- Central module F 865x from OS Version (05.21) on
- communication module F 8627X or F 8628X from OS Version 4.x on
  Also refer to the manuals of the F 8627X and F 8628X!

Note:
The resource name must have eight characters and the last two characters (Res-ID) must be numbers!
The ID (DIP switches 1 to 7) of the F 865x and the RES-ID of the resource name must be equal.
The ID_IP (DIP switch) on the communication module must be activated!



Communication Type
● Ethernet ○ Serial

Ethernet Settings
● Channel 1: 192.168.0.7
○ Channel 2: 192.168.0.8
○ IP: 192 . 168 . 0 . 3

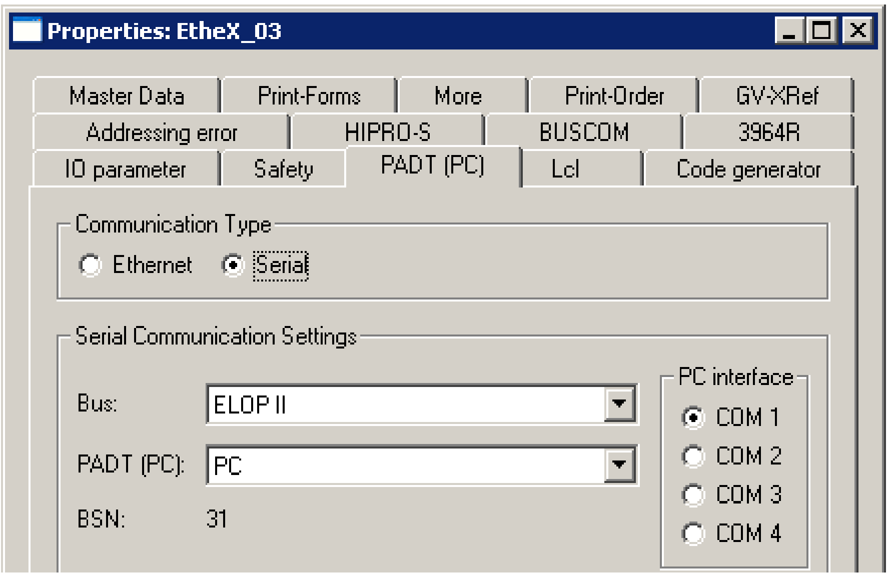
Resource properties window, ´Programming Unit´ tab "Communication Type Ethernet"

Ethernet Settings
The IP addresses of the PES are determined from the resource name by ELOP II and can be selected in the field "Ethernet Settings".

- Channel 1:
  ELOP II TCP connection
  via the communication module 1 (DIP switch 2/1 = ON).
- Channel 2:
  ELOP II TCP connection
  via the communication module 2 (DIP switch 2/1 = OFF).
- IP:
  Select the entry field "IP" to enter an IP address manually.
  The manual input of an IP address is required for special operating system with an individually set IP address range of the F 8627X / F 8628X.

Note:
The ELOP II control panel must be opened from a resource with the same Res-ID as the ID (DIP switch 1 to 7) of the F 865x.
If the Res-ID of the resource does not match with the ID (DIP switches 1 to 7) of the F 865x, no ELOP II TCP connection via the communication module can be established.

Serial:
Select "serial" to communicate via the serial RS 485 interface to the controller.



Communication Type
○ Ethernet ● Serial

Serial Communication Settings
Bus: ELOP II
PADT (PC): PC
BSN: 31

PC interface
● COM 1
○ COM 2
○ COM 3
○ COM 4

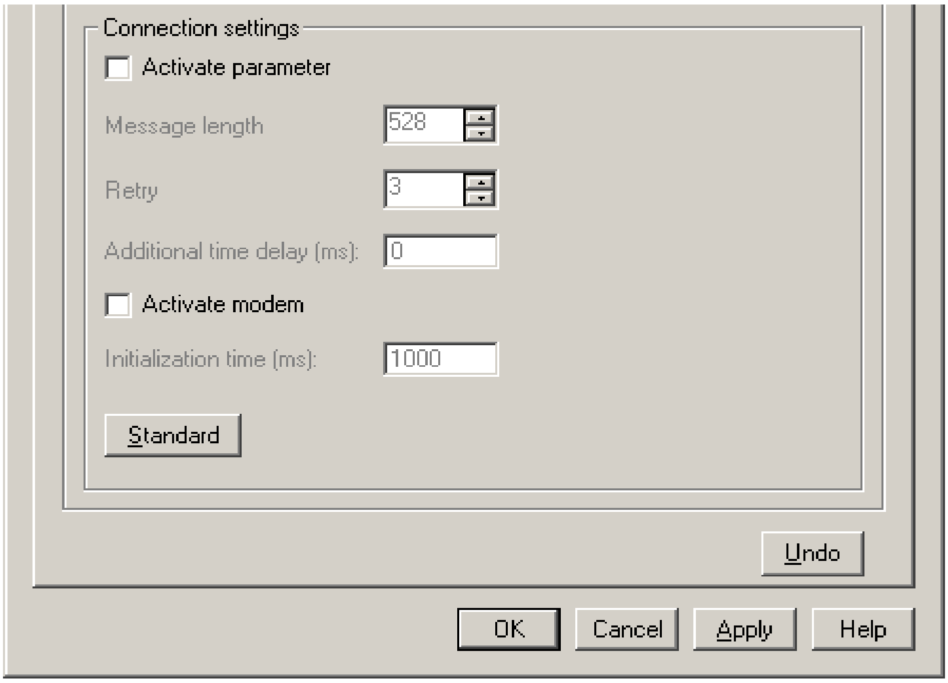
Resource properties window, ´Programming Unit´ tab "Communication Type Serial"

Serial Communication Settings

- Bus:
  Select a previously defined bus for the communication with the PES
  (see "Bus definitions").
- PADT (PC):
  Select the PADT (PC) you want to access the selected bus.
- PC interface:
  Define the communication interface by which the PADT (PC) is connected to the bus.

Note:
If using an interface converter e.g. H 7505 please note that ELOP II is not switching the DTR Signal for control of the conversion direction as a standard. An RS 485 interface is recommended.

Connection settings
In addition you can configure "Connection settings" for the communication from the PADT to the PESs (e.g. if you use a telephone modem).



Connection settings

□ Activate parameter

Message length: 528

Retry: 3

Additional time delay (ms): 0

□ Activate modem

Initialization time (ms): 1000

Resource properties window, ´Programming Unit´ tab "Connection Settings"

- Activate parameter:
  The check box "Activate parameter" activates or deactivates the user settings of the parameters "Message length", "Retry" and "Timedelay (ms)".
  Activate the check box to change the parameter values. The displayed parameter values are used for the connection.
  Deactivate the check box to disabled the user settings of the parameter values. The standard parameter values are used for the connection.

Note:
Communication disturbance (e.g. electronic disturbance) can be reduced by setting shorter telegram lengths and can be tolerated by more repetitions of transmissions.

- Message length:
  The "Message length" (16 up to 528) defines the maximum length of a telegram that is send from the PADT to the PES.
  Standard value:528
  Range of values:16..528

- Retry:
  Number of send attempts (0 up to 10) if the PES does not answer.
  Standard value:3
  Range of values:0..10

- Timedelay (ms):
  After this time a new send attempt is undertaken.
  When communicating e.g. via a telephone modem this time can be increased to allow for the delay in response by the bus stations caused by the modem.
  Standard value:0 ms
  Range of values:0..60 000 ms

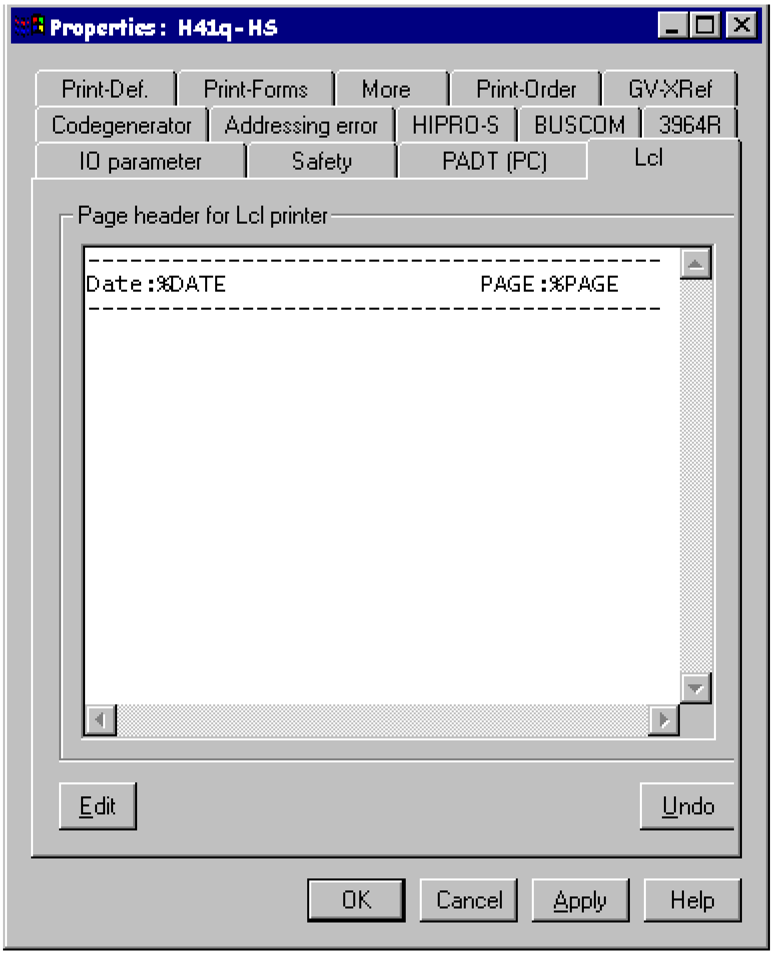
- Activate modem:
  If the PC master is connected to the selected bus via a modem, then highlight this control box. The DTR signal will then be switched first when operation starts.

- Initialization time (ms):
  Define the time needed by the modem for dialing here.
  Standard value:1000 ms
  Range of values:0..65 535 ms

- Standard:
  Click on "Standard" to set the parameters to the standard values.

2.2.2.5 Lcl Texts

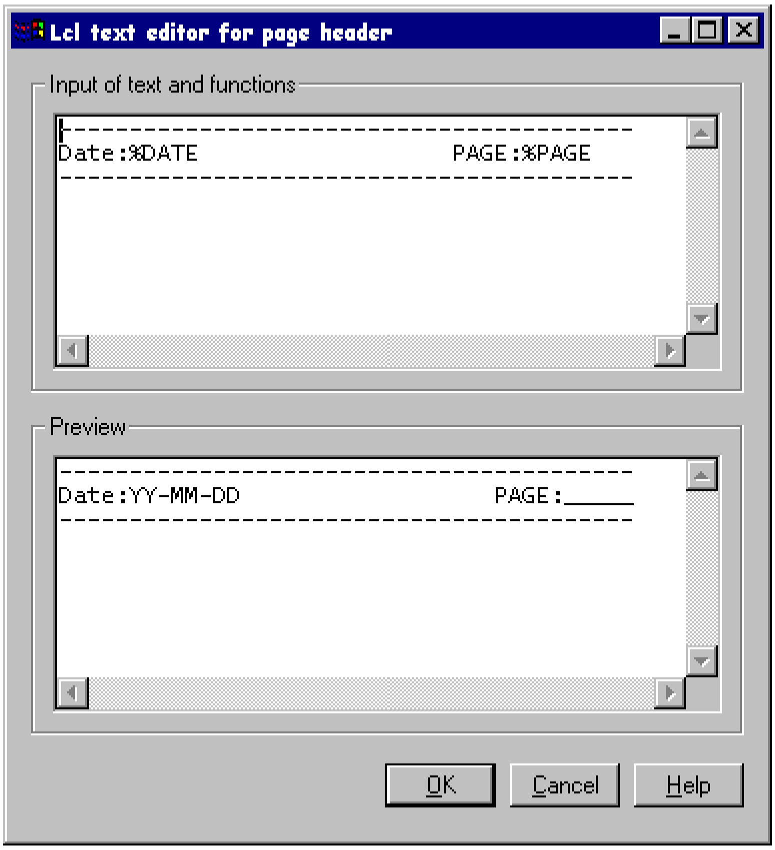
Logic controlled logging (Lcl) is used to record events (Boolean signal changes with time of day) on the central module and to print off the events - with interpretation - on a connected printer. You can define a page header for this report printout on this tab. A print preview of the Lcl page header is displayed in the ´Page header for Lcl output´ box.



```
----------------------------------------
Date:%DATE                   PAGE:%PAGE
----------------------------------------
```

Resource properties window, ´Lcl´ tab

Click on the ´Edit´ command button to start interactive header creation.



```
-----------------------------------------
Date:%DATE                      PAGE:%PAGE
-----------------------------------------
```

```
-----------------------------------------
Date:YY-MM-DD                      PAGE:_____
-----------------------------------------
```

Dialogue box ´Lcl text editor for page header´

The dialogue is divided in two parts: you enter the texts and wildcards for the page header in the top half, and the bottom half displays a print preview. The following wildcards can be used:

- %CR: word wrapping (carriage return)
- %DATE: current date in format YY-MM-DD
- %PAGE: current page number of the printout

2.2.2.6 Code generator

The user program is translated into machine code (code generation) once you have entered the complete function, variables and resource declaration. The machine code required for the PES is generated depending on the selected resource type. After the code is generated the user program can be loaded into the PES memory
Here you are able to configure the codegenerator settings

- Acknowledge start of code generator
  Start of the code generator must be explicitly confirmed again if this option is selected
- Create reloadable code
  A reloadable code must be generated before online changes can be made.

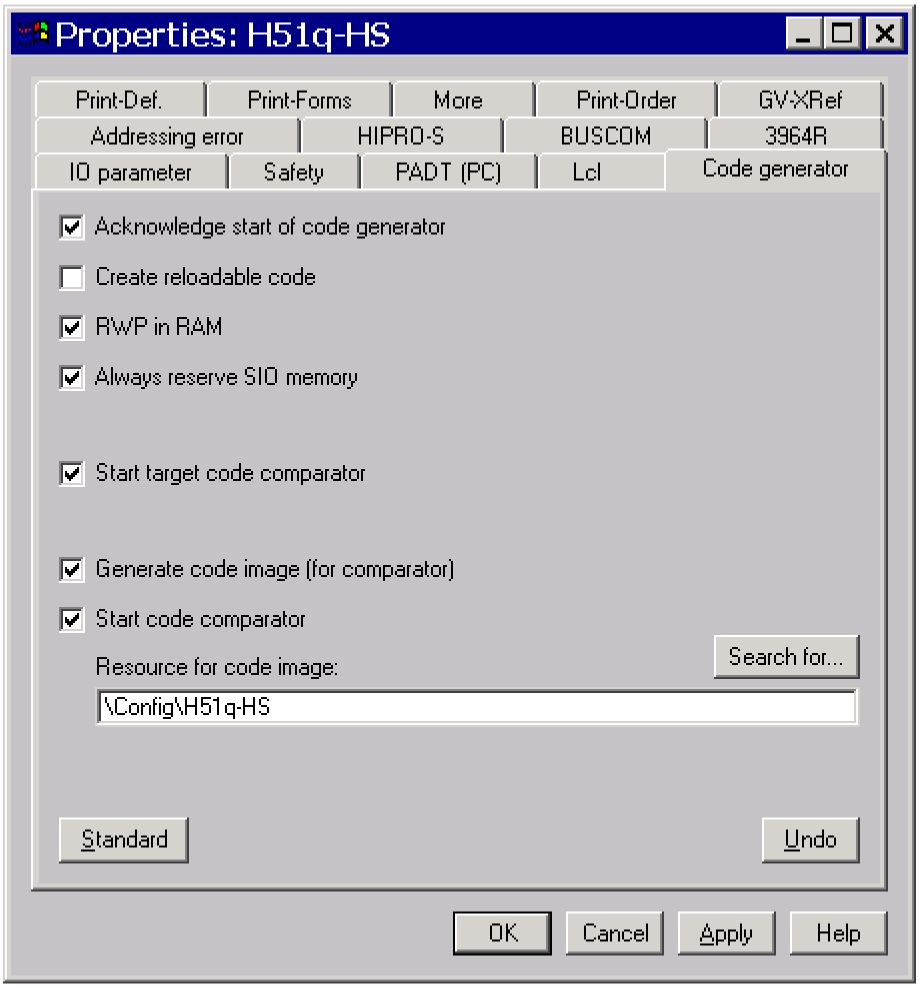
Note:
A reloadable code cannot always be generated. This is not possible when modules or communication variables are inserted or deleted.

- SLP in RAM
  The SLP area that is normally stored in the flash EPROM can be stored in the RAM if required. This will then enable variables to be changed during operation.
- Always reserve SIO memory (only for H41q/H51q)
  If this parameter is disabled, the communication settings can not be changed online. If the SIO memory is reserved, the change of communication settings via reload is possible

In resources H41q/H51q the following options for safety related revision control and code generation are available on top of these. These options are certified by TÜV Product Service (further Information in the chapter ´Procedure for safety related PES´).

- Start target code comparator
  By selecting this option the target code is generated in two separate tasks and compared afterwards. Random errors of the hardware platform would be detected this way.
- Generate code compare image
  The code compare image is needed for a later safety related revision check.
- Start code comparator
  By selecting this option changes within a resource will be detected. You need to select a resource to compare with. This could be exactly the same resource or a backup of an earlier revision.

.



Properties: H51q-HS

☑ Acknowledge start of code generator

□ Create reloadable code

☑ RWP in RAM

☑ Always reserve SIO memory

☑ Start target code comparator

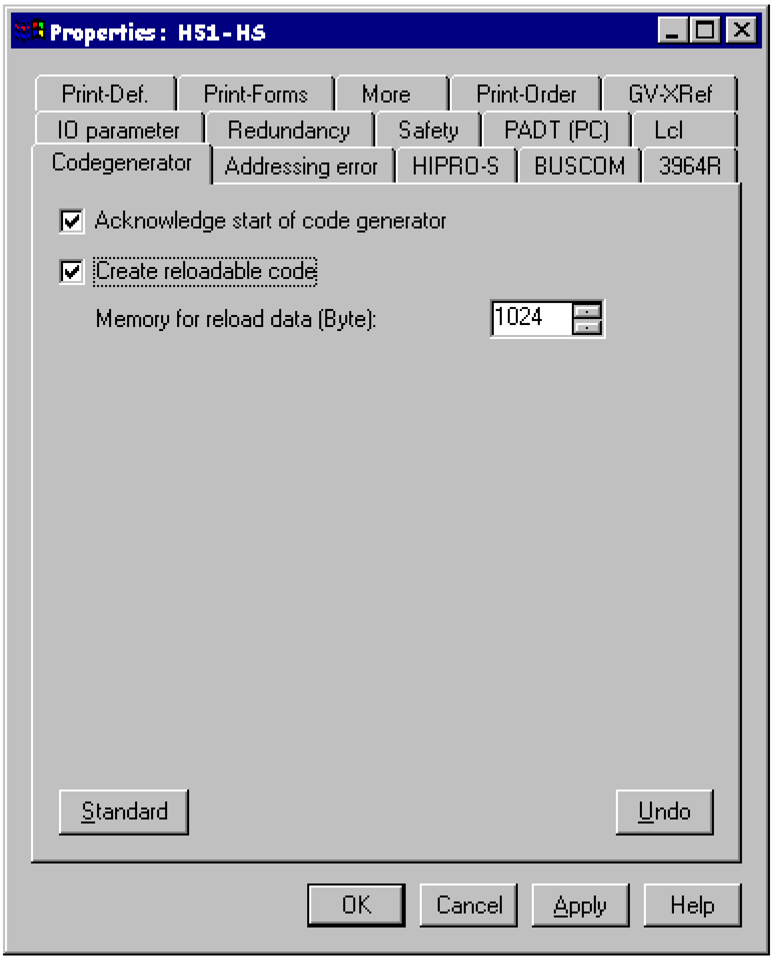
☑ Generate code image (for comparator)

☑ Start code comparator

Resource for code image: \Config\H51q-HS

Resource properties window, ´Code generator´ tab for H41q/H51q family

Next you see the properties for the PES A1, A1dig, H11, H41, H51. The options for the safety related codegeneration and revision check are not available. The option ´RWP in RAM´ is deleted, since there is no other option in these systems.
Instead it is necessary to configure a memory area for reload data, since there is only a restricted area available. The default of 1024 bytes is a good compromise between program size and sufficient space for online changes



☑ Acknowledge start of code generator

☑ Create reloadable code

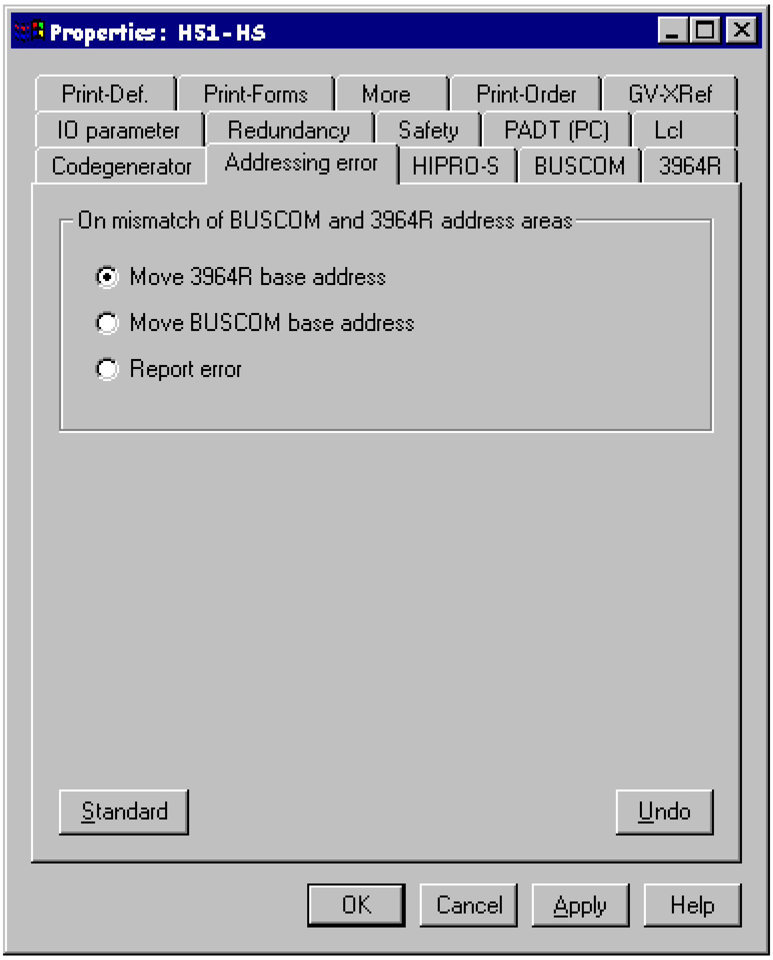
Memory for reload data (Byte): 1024

Resource properties window, ´Code generator´ tab for systems A1, H11, H41/H51 family

2.2.2.7 Address conflicts

If address conflicts occur during code generation, you can use this function to select a response. The same basic address (0) is specified as default for communication via 3964R and for BUSCOM, which is why there can be address conflicts:

- Move 3964R basic address
  If this option is selected the 3964R basic address is automatically moved to a value that will avoid conflict with BUSCOM addresses used at the same time.
- Move BUSCOM basic address
  If this option is selected the BUSCOM basic address is automatically moved to a value that will avoid conflict with 3964R addresses used at the same time.
- Generate error
  If this option is selected, addresses will not be moved automatically and instead you will see an error message during compiling which draws your attention to the conflict. You must then make a manual correction.



On mismatch of BUSCOM and 3964R address areas

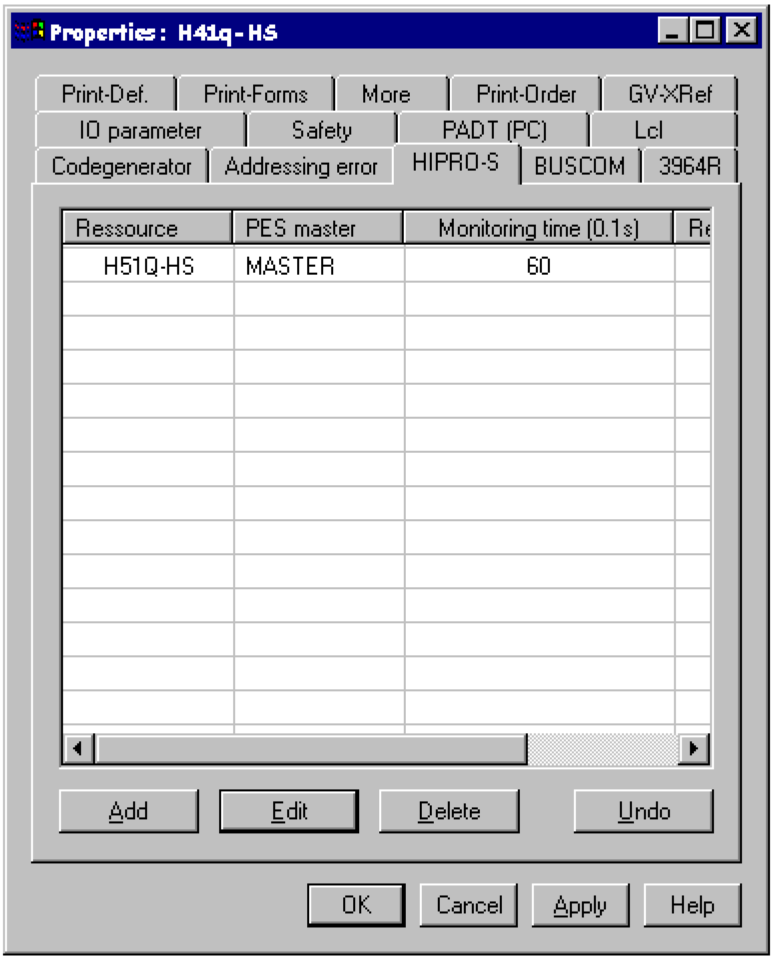
● Move 3964R base address

○ Move BUSCOM base address

○ Report error

Resource properties window, ´Addressing errors´ tab

2.2.2.8 HIPRO-S

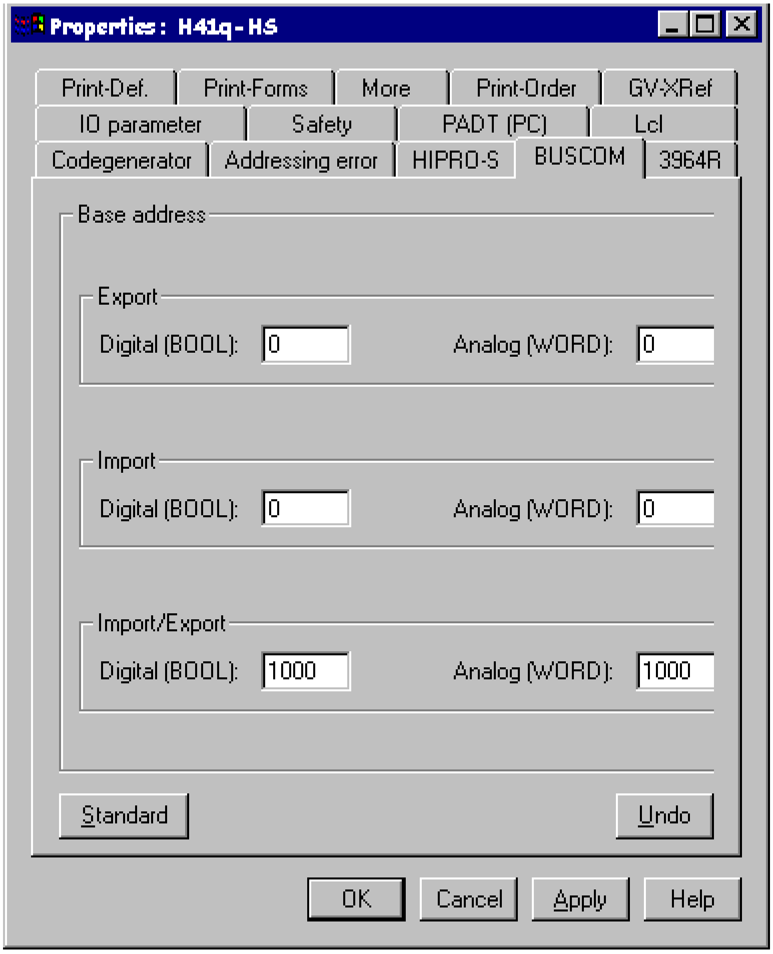


Ressource	PES master	Monitoring time (0.1s)	Re
H51Q-HS	MASTER	60	

Resource properties window, ´HIPRO-S´ tab

The HIPRO-S protocol is used for safety-related communication between two or more PESs. The controllers communicate across the configured communication system.
For each resource it must be configured with which other resources safety related data exchange is performed. The variables for the data exchange are configured in the variables declaration editor of the program instance (See “Safety-related data transfer: HIPRO-S” on page 43.)

2.2.2.9 BUSCOM protocol



Base address

Export
Digital (BOOL): 0
Analog (WORD): 0

Import
Digital (BOOL): 0
Analog (WORD): 0

Import/Export
Digital (BOOL): 1000
Analog (WORD): 1000

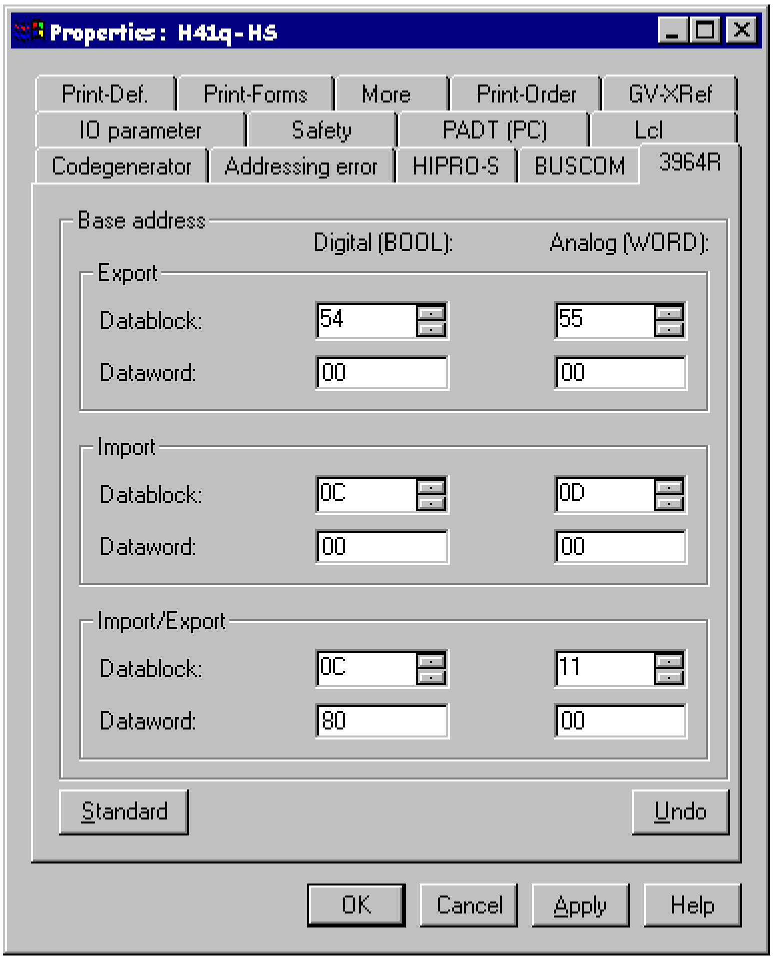
Resource properties window, ´BUSCOM´ tab

Here you can enter the basic addresses for BUSCOM, e.g. for READ and WRITE of Variables by a MODBUS master:

- Export
  Here are the basic addresses starting from which a MODBUS master can export variables of the PES with function code 1 (digital (bool)) or 3 (analog (word)).
- Import
  Here are the basic addresses starting from which a MODBUS master can import variables of the PES with function codes 5 and 15 (digital (bool)) or 6 and 16 (analog (word)).
- Export/Import
  Some control systems are unusual in that they immediately re-export the variables that they import. They only change the function code without changing the basic address, so a separate memory area must be reserved for write/read accesses of this type.

Note
For H41q/H51q systems the valid address range for BUSCOM base addresses is 0 to 2048 and 4096 to 8192.

2.2.2.10 3964R Protocol



Base address

		Digital (BOOL):	Analog (WORD):
Export	Datablock:	54	55
	Dataword:	00	00
Import	Datablock:	0C	0D
	Dataword:	00	00
Import/Export	Datablock:	0C	11
	Dataword:	80	00

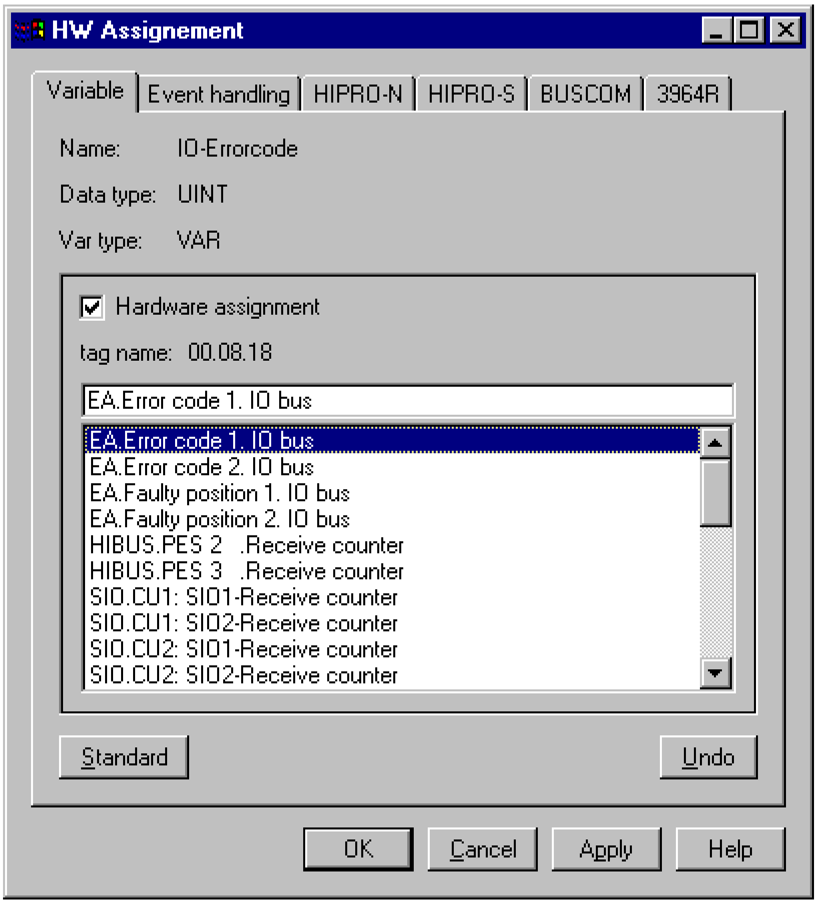
Resource properties window, ´3964R´ tab

The data blocks and data words required for communication via the 3964R protocol can be entered here.

3 Resource-related functions in the Editor

Once a program has been instanced to a resource, a number of resource-related items can be configured as you edit the program instance in the Editor.

3.1 Assigning system variables



HW Assignement

Name: IO-Errorcode

Data type: UINT

Var type: VAR

☑ Hardware assignment

tag name: 00.08.18

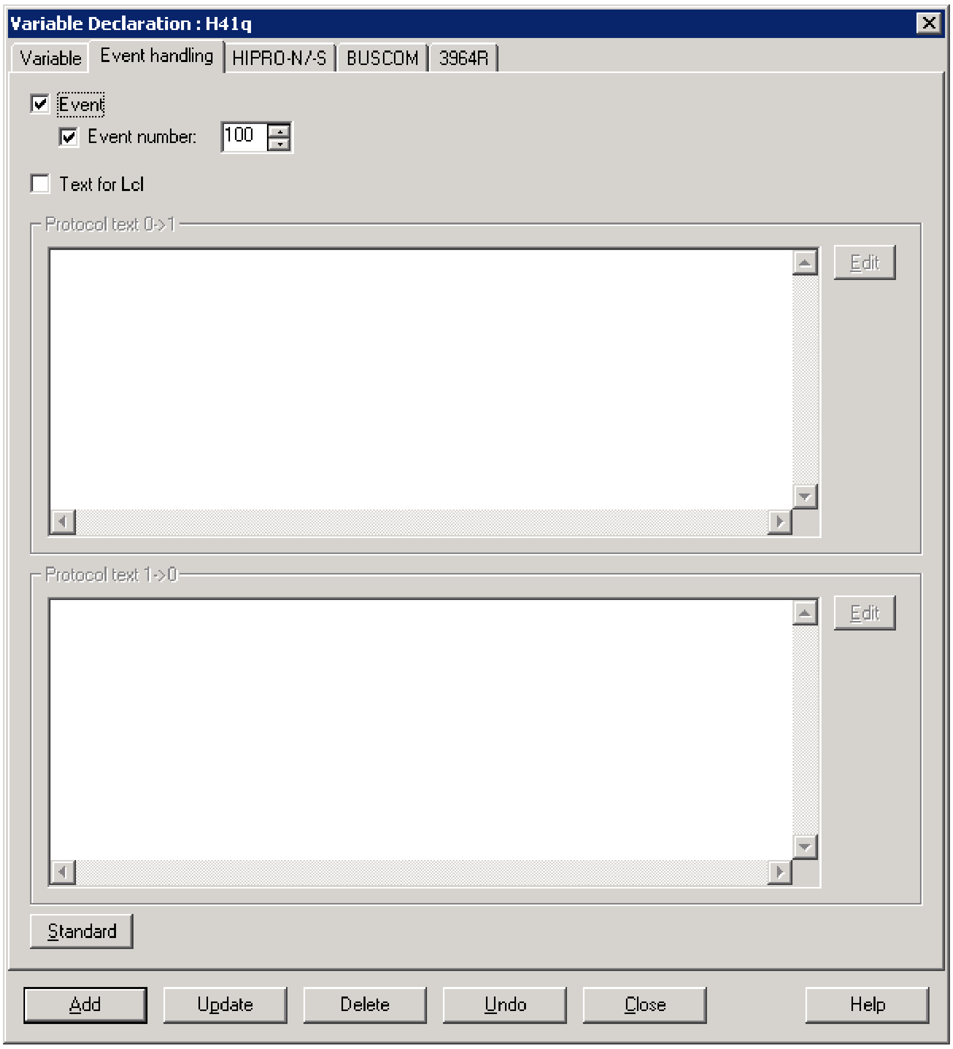
EA.Error code 1. IO bus

EA.Error code 1. IO bus
EA.Error code 2. IO bus
EA.Faulty position 1. IO bus
EA.Faulty position 2. IO bus
HIBUS.PES 2 .Receive counter
HIBUS.PES 3 .Receive counter
SIO.CU1: SIO1-Receive counter
SIO.CU1: SIO2-Receive counter
SIO.CU2: SIO1-Receive counter
SIO.CU2: SIO2-Receive counter

Variable properties window, ´Variable´ tab

To access the dialogue shown above, call the context menu of a variable used in the logic and then select the ´Properties´ option.
You will see different system variables in the selection window depending on the type of variable you have clicked on (BOOL or UINT). The system information is available after you click on ´Assign hardware´ and select a system variable.

3.2 Creating Events



☑ Event

☑ Event number: 100

□ Text for Lcl

Protocol text 0->1

Protocol text 1->0

Variable properties window, ´Events´ tab

If the check box "Event" is activated, all data required for the event configuration (e.g. time stamp) are created.
For the codes to read the events of the HIMA PES, please look up the manual of the reading device (e.g. PLS).

The user must accomplish the numbering of the events for all variables either manually or automatically.

Manual numbering of events:
If the check button "event number" is activated, the event number for this variable can be set from 0 up to 2047 in the input field "event number".
Event numbers may not be used twice (Error is recognized during the code generation)!

Automatical numbering of events:
Gaps in the event numbering, are filled up by the automatic numbering of the events (beginning with the smallest address).
The event number is created by an alpha-numeric sort of the events. For sorting the system adds the character ´@´ to variables without hardware assignment. The tag name is used for the variables with hardware assignment.
If generating of reloadable code is selected, the system tries to keep the existing event numbers while deleting or adding events. If events are deleted the numbering of events therefore will have gaps. If events are added, first all gaps will be used and afterwards additional events will be added at the end.

3.3 Variables for transmission with HIPRO

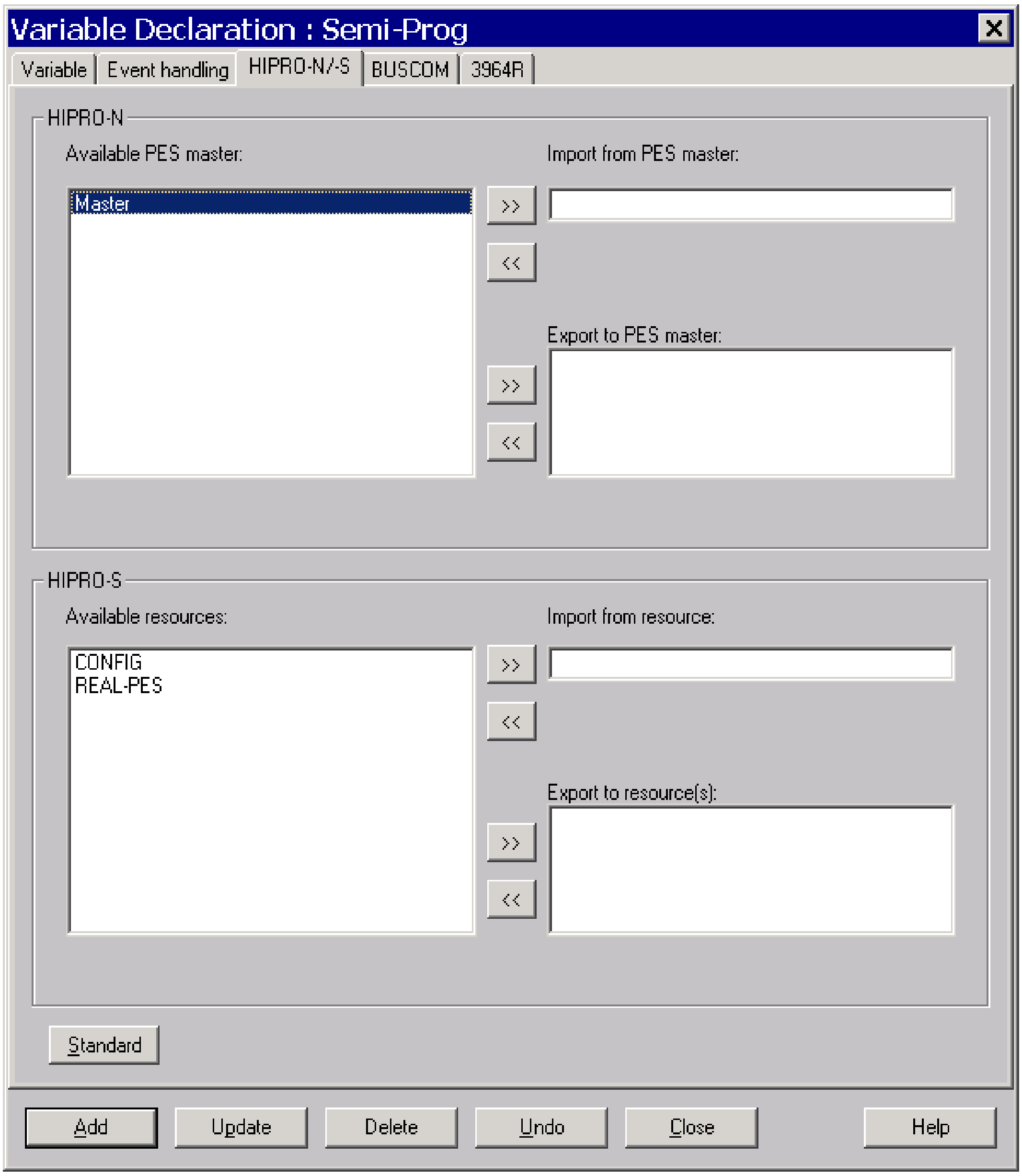
HIPRO-N and HIPRO-S are the non-safety related and safety-related protocols used for data interchange between HIMA PESs. They work on the master/slave principle, and several masters can be connected to a bus as well as several slaves.

A maximum of 255 bus stations can be addressed on a HIPRO-N/S.
Because up to 8 bus stations per controller are possible in PESs that belong to the H41/H51 and H41/H51q family (2 CPUs and 6 coprocessor modules), this maximum number must be divided by 8 to arrive at the number of bus stations.
Up to 31 bus stations in the H41/H51 and H41q/ H51q family can therefore be connected to a HIPRO-N/S as master or slave.

Every bus station on the HIPRO-N/S is identified by a user-defined bus station number (BSN) that is set on the system´s central module by means of DIP switches. Since every master/slave in a bus station can be connected to a separate bus, each bus station can be connected to up to 8 buses. The number of controllers that can interchange safety-related data on the HIPRO-S is limited to 6 in the H41/H51 family and 64 bus stations in the H41q/H51q family.

Functions of the PES master

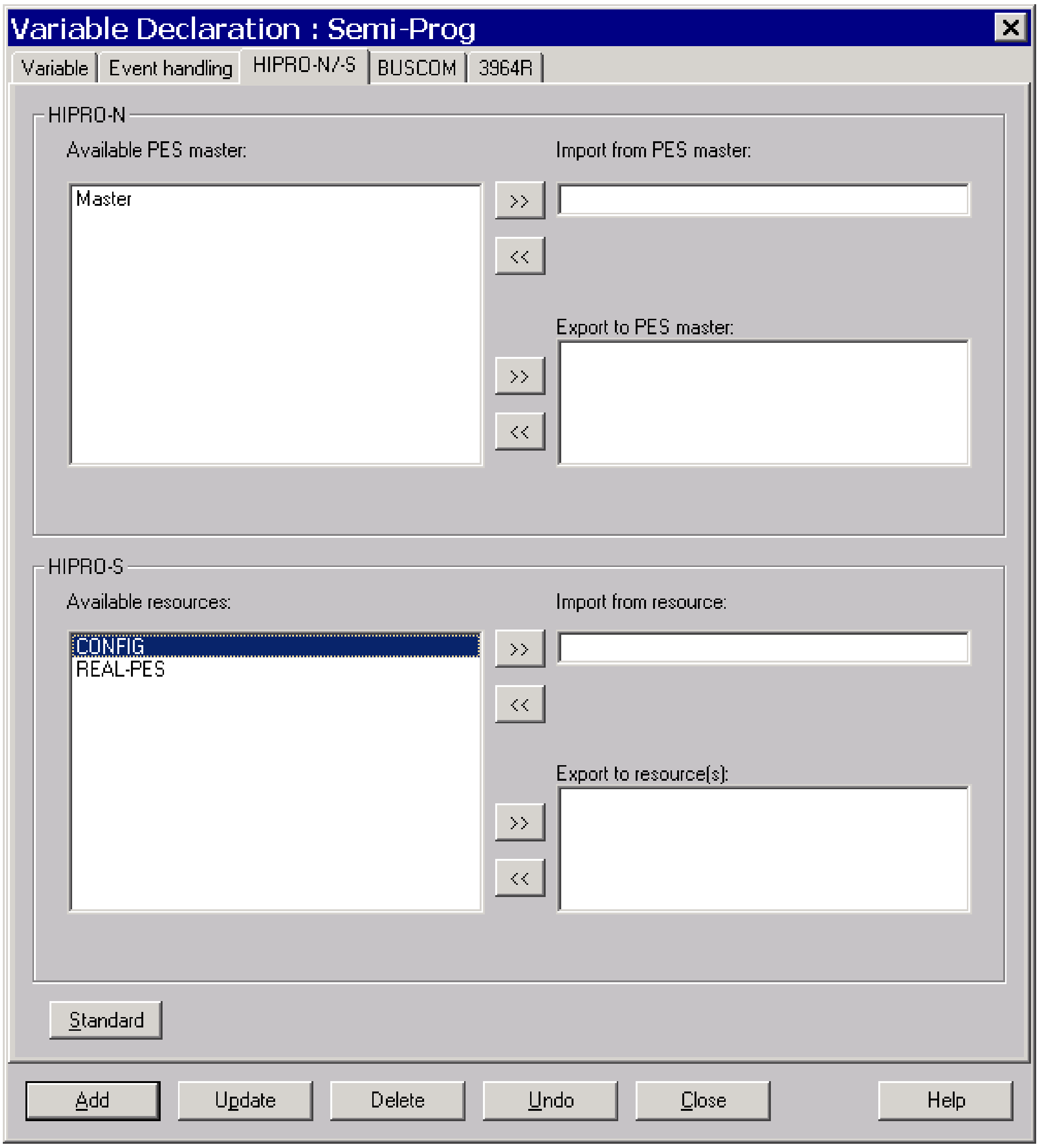
- PES masters as data centre for HIPRO-N
  With this type of data transfer the PES master acts as the data centre, reading the data of the connected slaves, compiling the transmission and then sending it to the appropriate PESs.
  If data cannot be exported from a PES because of a failed connection during PES master operation, the PES master writes the data from this PES to the other PES to FALSE or, if configured, holds the data at its last value prior to the connection failure for a specified time or until the connection is restored.
  If changes to the user program of a resource are made in the configuration, then only the user program of the affected PES and PES master need to be re-compiled and reloaded.
- PES master for starting safety-related transmissions
  The PES Master does not act as the data centre for safety-related transmissions via the HIPRO-S between different PESs, it only starts the transmissions. The necessary configurations for the PES master are made in the settings of the resource.
  The variables of the sending controller are transmitted directly to the receiving controller. Notionally, this type of transfer is like a point-to-point connection, except that it runs via the HIPRO-S. Individual transmissions are protected by codes and signatures.
  All variables that are exported to another resource must be the same as the variables imported from that resource.



Variable properties window, ´HIPRO-N/-S´ tab

In this tab you can define by which PES master the variables should be transferred non safety-related.
The data of the resource just edited can either be imported from a PES master (´Import from PES master´) or exported to several other PES master (´Export to PES master´).

Note:
You can only specify one PES master for ´Import from PES master´ because only a single master can access the variables for importing, whereas with ´Export to PES master´ several masters can be specified as different masters can be permitted to access variables for export.



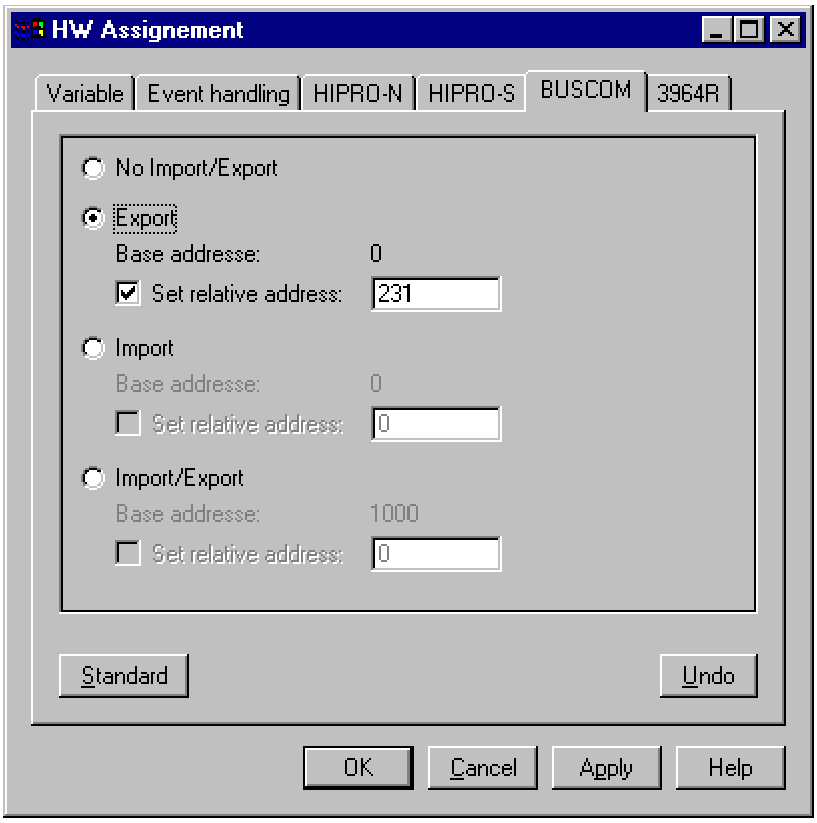
Variable properties window, ´HIPRO-S´ tab

In this tab you can define two PESs between which data will be transferred safety-related. All resources on the HIPRO-S from which data can be imported or to which data can be exported are displayed under ´Available resources´.

Note:
You can only specify one resource for ´Import from resource´ because the currently edited resource may only take data directly from another (point-to-point connection), whereas several resources can be specified for ´Export to resources´ because the currently edited resource may send its data to several resources.

3.4 Variables for transfer to external systems

In the following tabs you can define options for access by external systems (MODBUS master, OPC Server, Profibus-DP master or 3964R master) to the HIMA PES variables.



○ No Import/Export

● Export

Base addresse: 0

☑ Set relative address: 231

○ Import

Base addresse: 0

□ Set relative address: 0

○ Import/Export

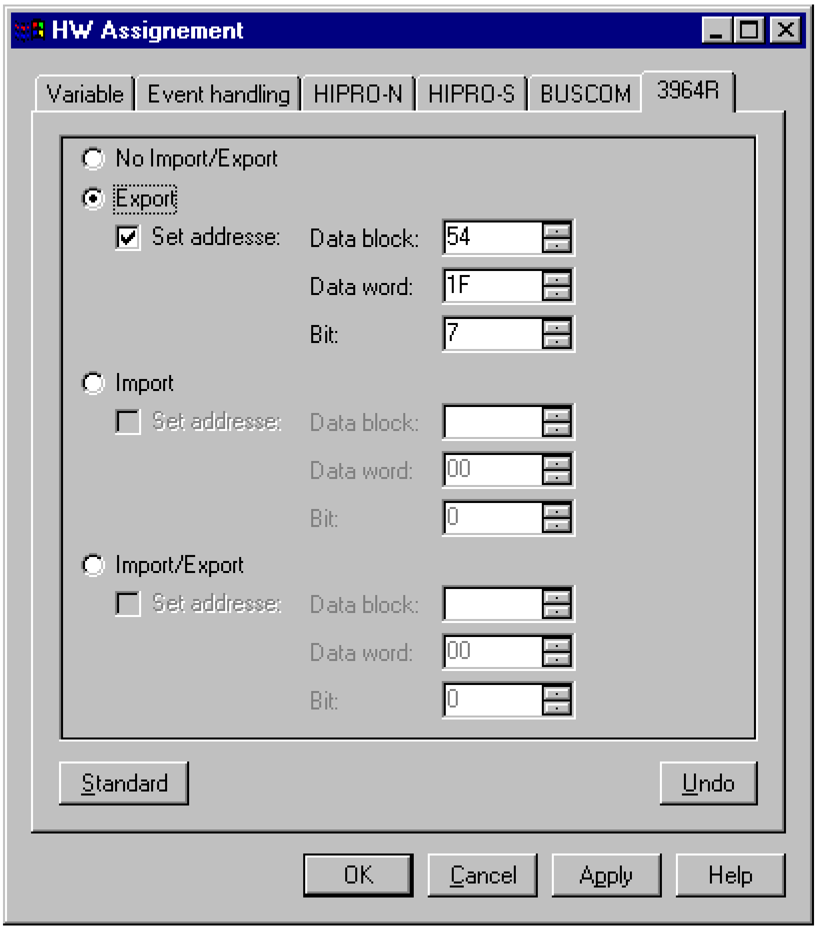
Base addresse: 1000

□ Set relative address: 0

Variable properties window, tab

In the BUSCOM communication tab you can define whether a variable will be exported, imported or im- and exported by the master. Selecting one of these points activates the option of defining a relative address.

Note:
If you do not use the option of defining relative addresses manually, then all BUSCOM variables will be sorted in alphanumeric order and addressed sequentially.
For each entry the address range is 0..6142. The codegenerator moves the addresses 0..2047 to 0..2047 and 2048..6142 to 4096..8192 automatically.



○ No Import/Export

● Export

☑ Set addresse: Data block: 54

Data word: 1F

Bit: 7

○ Import

□ Set addresse: Data block:

Data word: 00

Bit: 0

○ Import/Export

□ Set addresse: Data block:

Data word: 00

Bit: 0

Variable properties window, ´3964R´ tab

Manual settings for the 3964R protocol can be made here similar to the configuration of BUSCOM variables.

Communication

4 Communication between HIMA PESs

The communication system must be configured for communication with the programming unit (PADT) or for communication between HIMA PESs and between a HIMA PES and the HIMA OPC server.
Only HIBUS is currently available as the communication system, and this requires a bus configuration (See “Creating a configuration” on page 1.)
The bus configuration contains all the PESs, PES masters and PADTs that are connected to the bus and which can therefore interchange data. The PES master in a coprocessor module controls the communication.
The BUSCOM configuration is used for communicating on the Ethernet.

4.1 Protocols

Data transfer between HIMA PESs can be both safety-related and non-safety related.

4.1.1 Safety-related transfer: HIPRO-S

The PES master only controls data traffic. Data is exchanged directly between two PESs. Safety-related communication between HIMA PESs is certified according to AK6.

4.1.2 Non-safety related transfer: HIPRO-N

The PES master acts as the data centre. A PES master reads and writes the interchanged data which is buffered in the PES master.

4.2 HIBUS Hardware

There is a range of system components available for setting up a HIBUS between HIMA PESs (terminals for DIN rail mounting, converters and standard cables). You will find data sheets and wiring options in the system catalogue for H41q/H51q.

Note:
The coprocessor module in which a PES master is loaded must be connected to the HIBUS. This will also connect the PES (slave) to the HIBUS. A separate connection is not permitted.

4.3 Redundancy

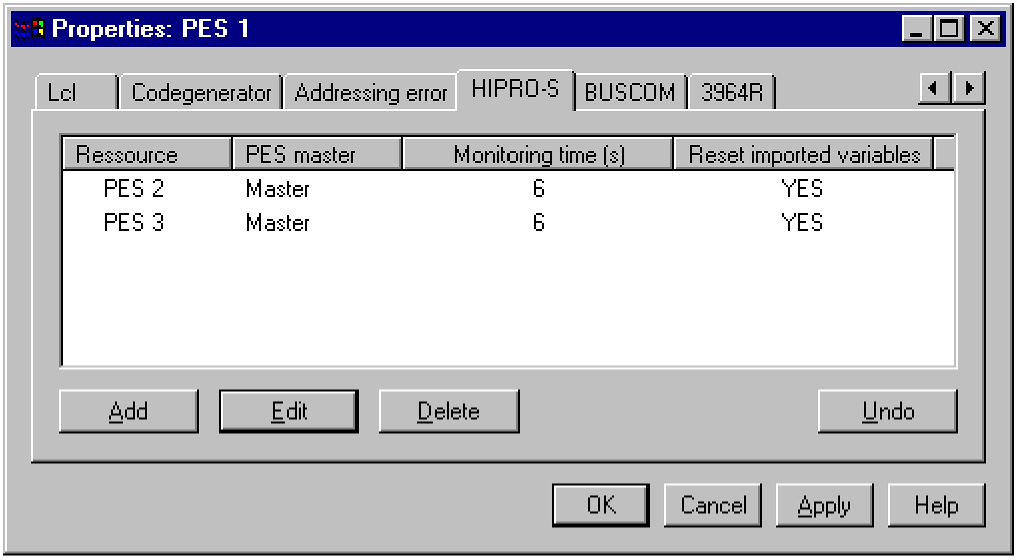
The HIBUS can be connected redundantly to the PESs and the PES master. Only one free interface is required for this in the PESs; the redundant bus connection of the PES master must be configured in the bus configuration accordingly.
The PES master performs a continuous connection check on both buses. If there is no connection to a PES on the main bus (the bus that is connected to the first interface of the PES master), then communication with that PES is routed via the reserve bus (the bus on the second interface of the PES master).
The same name must be used in the bus configuration for redundant PES masters. Full redundancy is achieved when two PES masters are used in different PESs.

4.4 Configuring data exchange

Different configurations must be made depending on the type of data transmission (safety-related or not).

4.4.1 Safety-related data transfer: HIPRO-S

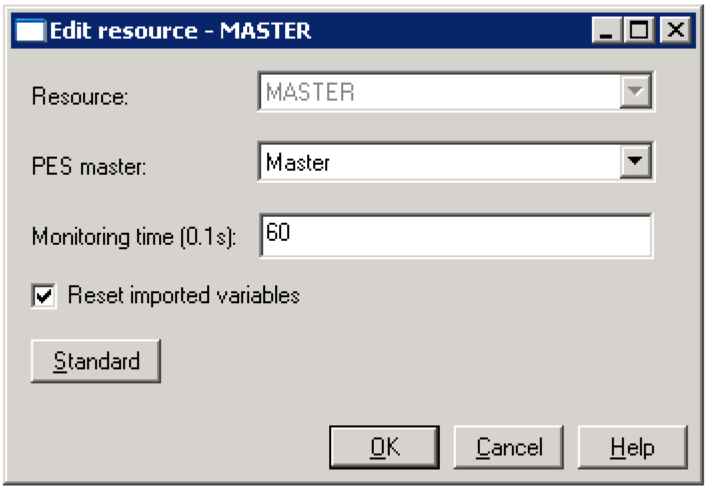
Safety-relate data transfer takes place between two PESs, so for each resource you must configure which of the PESs on the bus will be used to transfer data.



Ressource	PES master	Monitoring time [s]	Reset imported variables
PES 2	Master	6	YES
PES 3	Master	6	YES

Properties resource

In this tab you can add, edit or delete a resource. Up to 64 resources can be specified in PES H41q/H51q. This number is limited to 6 in PES H41/H51.



Resource: MASTER

PES master: Master

Monitoring time [0.1s]: 60

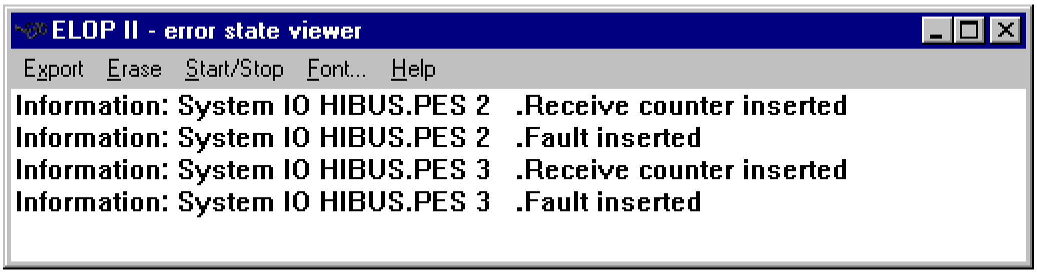
☑ Reset imported variables

Edit resource

- ´Resource´
  The resources contained in the bus configuration are available here in a selection list.
- ´PES master´
  Here enter the PES master that will control data traffic to this resource. All the PES masters contained in the bus configuration are available in a selection list.
- ´Monitoring time´
  If no data are written from the resource inside this time, then they are reset.

Now activate the settings with the ´OK´ button. System variables will now be automatically generated for each resource.

.


ELOP II - error state viewer
Export Erase Start/Stop Font... Help

```
Information: System IO HIBUS.PES 2  .Receive counter inserted
Information: System IO HIBUS.PES 2  .Fault inserted
Information: System IO HIBUS.PES 3  .Receive counter inserted
Information: System IO HIBUS.PES 3  .Fault inserted
```

Error-Status-Display after closing the properties dialog

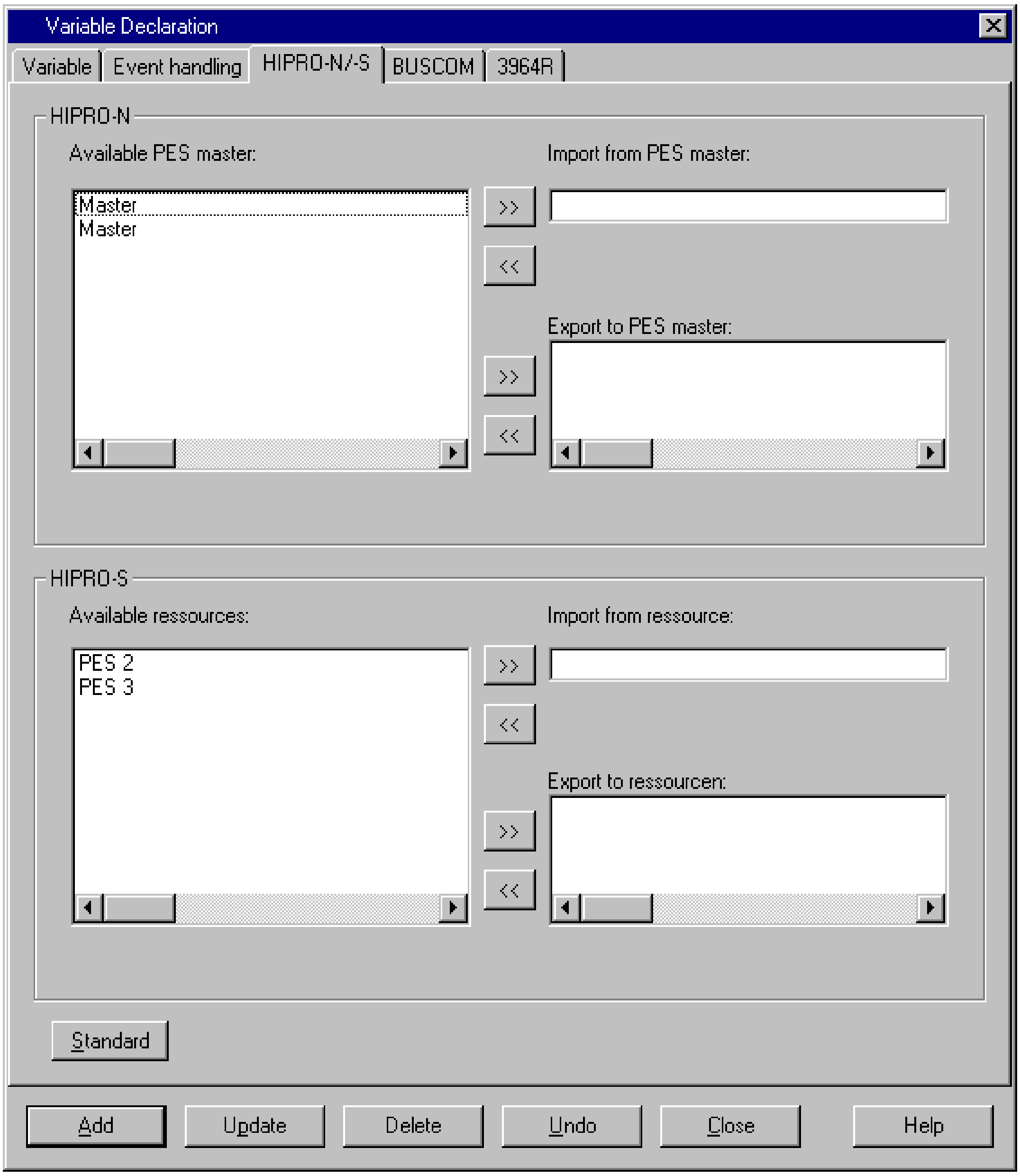
For each resource, the data to be transmitted is configured within the program instance in the variables declaration.

4.4.2 Non-safety related data transfer: HIPRO-N

For each resource, the data to be transmitted is configured within the program instance in the variables declaration.

4.4.3 Variables declaration

In the HIPRO tab you can specify where each variable will be imported from or exported to. The resource must be declared for safety-related data transfer and the PES master for non-safety related data transfer.
The resources and PES masters configured in the bus configuration are available for selection.



Variable declaration HIPRO

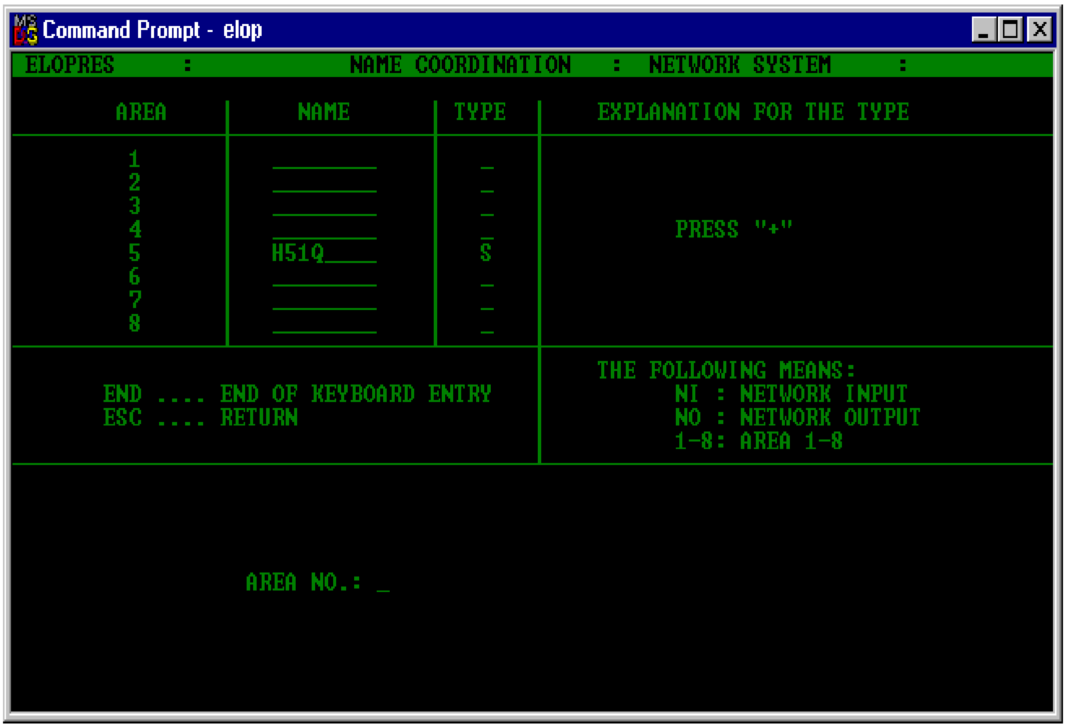
Note:
For each imported variable there must be precisely one variable that is exported. With safety-related data interchange this creates a 1:1 relationship, for non-safety related data interchange the relationship is 1:n, as an exported variable can be imported into several resources because the PES master acts as the data centre.

4.5 Configuring data interchange to ELOP slaves

It is not only possible to transfer data between PES programmed with ELOP II, but also with PES programmed with ELOP (DOS based programming software from HIMA). The protocol HIPRO-S is also used for this data transfer. By this way there is no data transfer between PES programmed with ELOP.

4.5.1 Configuration in ELOP

First of all you need to add a network (area 3-8) in the project in ELOP and configure the variables (please refer to the ´ELOP manual: Programming, Monitoring, Documentation). All blocks need to be filled with 16 variables (no empty entry).

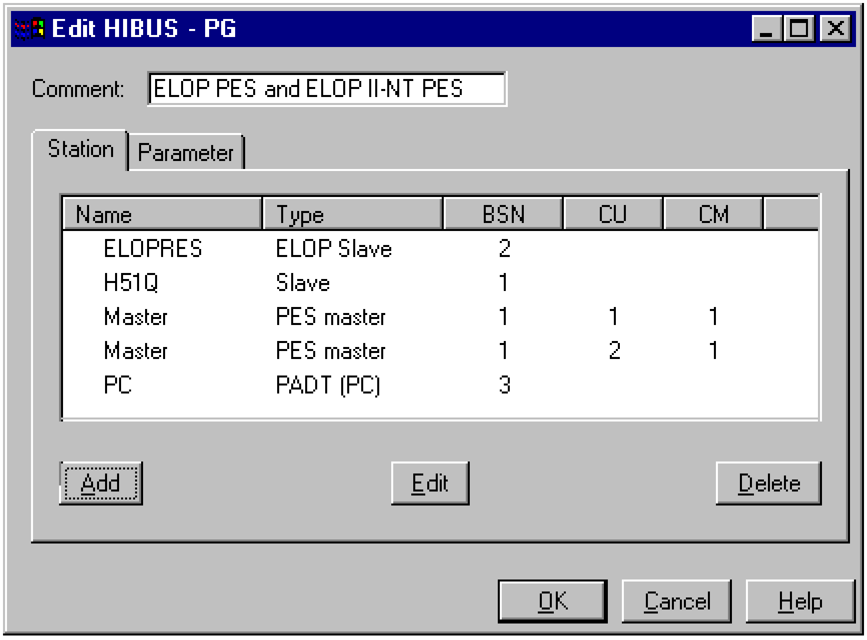


```
ELOPRES      :           NAME COORDINATION   :  NETWORK SYSTEM      :

        AREA     |     NAME      | TYPE |    EXPLANATION FOR THE TYPE

         1
         2          ________        _
         3          ________        _
         4          ________        _            PRESS "+"
         5          H51Q____        S
         6          ________        _
         7          ________        _
         8          ________        _

                                         |   THE FOLLOWING MEANS:
       END .... END OF KEYBOARD ENTRY    |         NI  : NETWORK INPUT
       ESC .... RETURN                   |         NO  : NETWORK OUTPUT
                                         |         1-8: AREA 1-8


                  AREA NO.: _
```

Name coordination: network system in ELOP

4.5.2 Bus configuration

In the bus configuration the PES programmed with ELOP need to be configured as ´ELOP Slave´ (See “HIBUS stations” on page 4.) The project name is automatically used for the resource name after configuration of the project file.



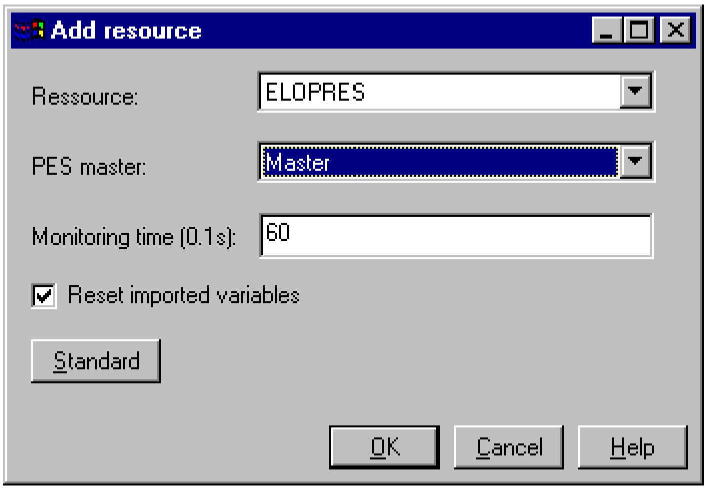
Comment: ELOP PES and ELOP II-NT PES

Name	Type	BSN	CU	CM
ELOPRES	ELOP Slave	2		
H51Q	Slave	1		
Master	PES master	1	1	1
Master	PES master	1	2	1
PC	PADT (PC)	3		

Bus configuration with ELOP slave

4.5.3 Resource configuration

The safety related data transfer is done between two PES. Therefore in the ELOP II resource the PES programmed with ELOP needs to be added in the register ´HIPRO-S´.



Ressource: ELOPRES

PES master: Master

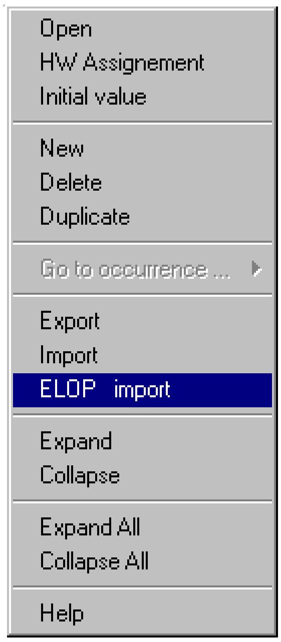
Monitoring time (0.1s): 60

☑ Reset imported variables

Properties resource

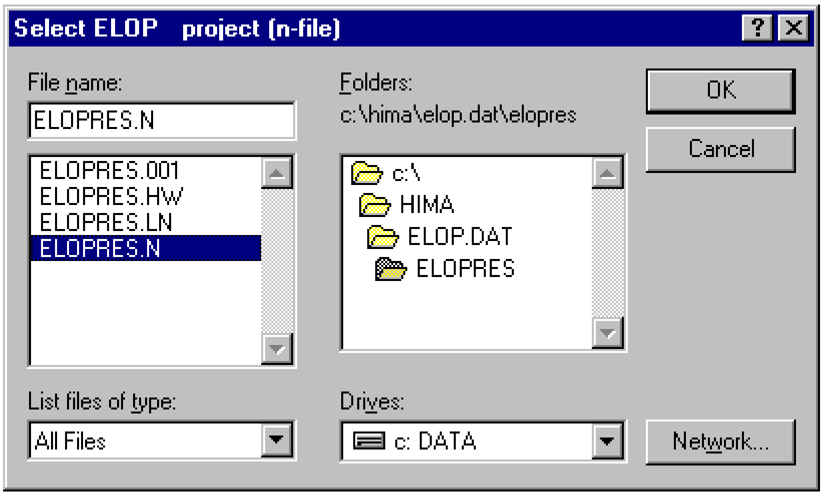
4.5.4 Variables declaration

The safety related transfer requires identical names in the source and target PES. Since the variable names in ELOP are restricted to 8 characters and ELOP has no import function, the variables should be configured first of all in ELOP (as described before).
The variables can then be imported in ELOP II.
In the context menu of the variables declaration select ´ELOP Import´.



Context menu of the program instance

Next you have to select the n-file of the ELOP project. You find this file in the ELOP project directory:
[drive]:\HIMA\ELOP.DAT\[PROJECT NAME]\[PROJECT NAME].N



Select ELOP project (n-file)

File name: ELOPRES.N

ELOPRES.001
ELOPRES.HW
ELOPRES.LN
ELOPRES.N

Folders: c:\hima\elop.dat\elopres

c:\
HIMA
ELOP.DAT
ELOPRES

List files of type: All Files

Drives: c: DATA

Select ELOP project (n-file)

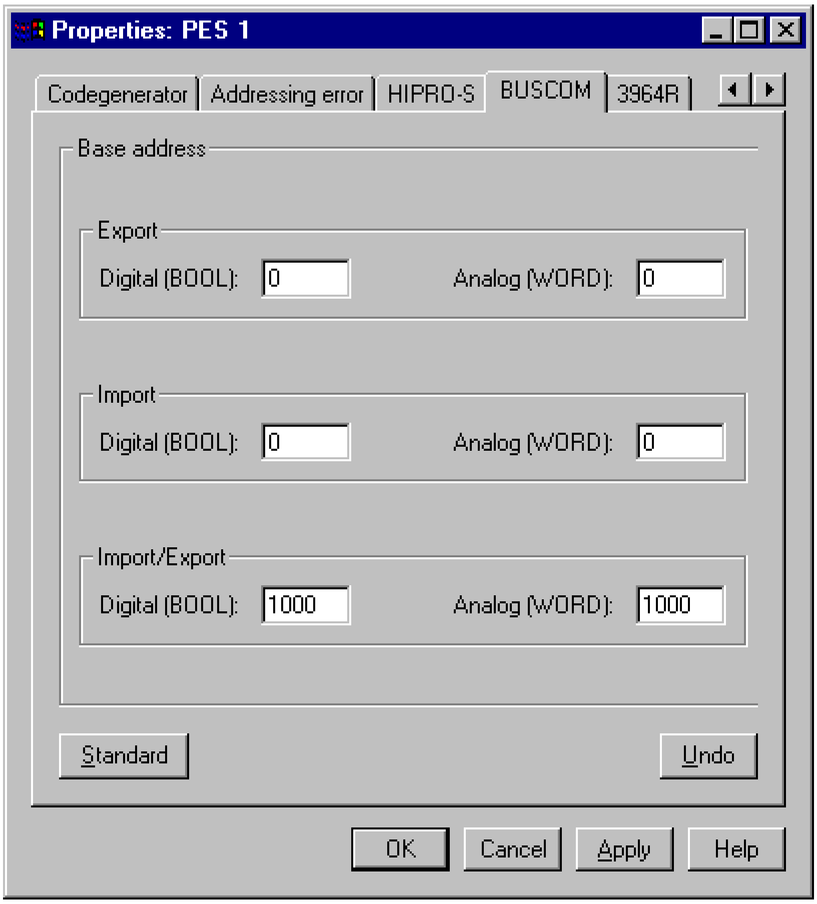
The variables configured in ELOP will then be imported. The HIPRO configuration is added automatically. The binary variables in ELOP are configured as data type BOOL and the digital variables are configured as data type UINT.
After loading the PES master the safety related data transfer is starting.

5 Communicating with external systems

Different protocols can be used to communicate with external systems. The BUSCOM definition is used for the configuration

5.1 BUSCOM Communication

The tab ´BUSCOM´ is used for the setting of base addresses for bus communication. The settings are used for MODBUS, OPC and fieldbuses.



Base address

Export
Digital (BOOL): 0
Analog (WORD): 0

Import
Digital (BOOL): 0
Analog (WORD): 0

Import/Export
Digital (BOOL): 1000
Analog (WORD): 1000

Properties resource

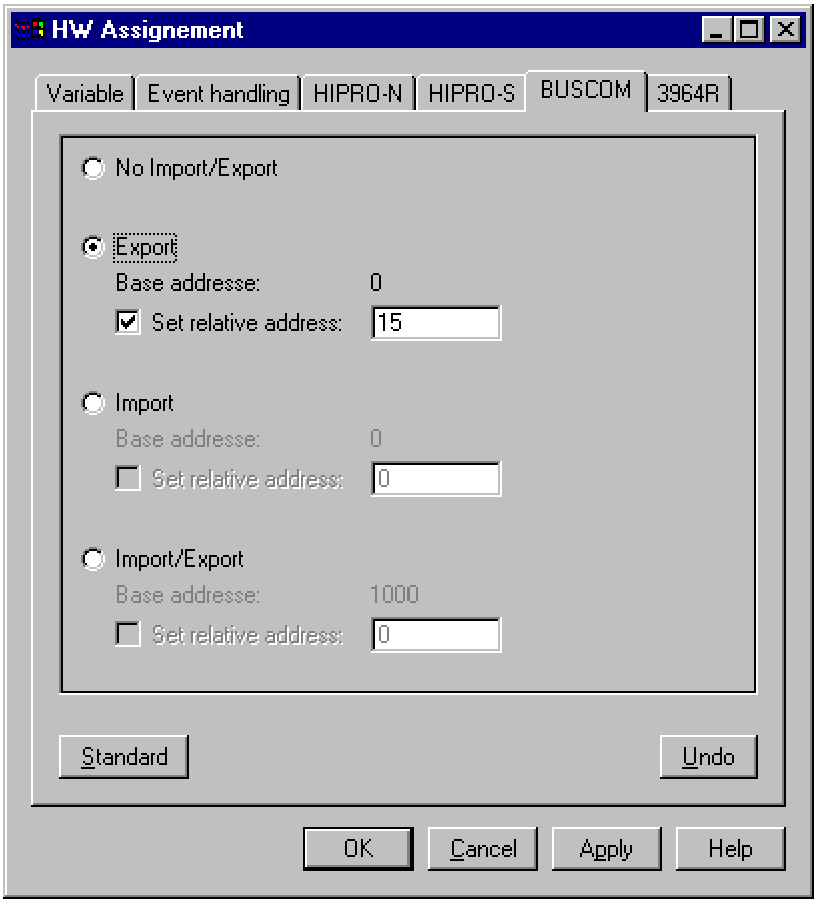
The HIMA PES have implemented the MODBUS function codes 1, 3, 5, 6, 8, 15 and 16

- ´Export´: Digital
- Function code 1: Variable is exported to the external system
- ´Export´: Analog
  Function code 3: Variable is exported to the external system
- ´Import´: Digital
  Function code 5/15: Variable is imported from the external system
- ´Import´: Analog
  Function code 6/16: Variable is imported from the external system
- ´Export/Import´: Digital
  Function code 1 and 5/15: Variable is exported to and imported from the external system
- ´Export´: Analog
  Function code 1 and 6/16: Variable is exported to and imported from the external system

The variables that are available for data transfer are configured in the variables declaration. The relative addresses can be assigned either automatically (in alphanumeric order) or manually.

Note
For H41q/H51q systems the valid address range for BUSCOM base addresses is 0 to 2048 and 4096 to 8192.

.


HW Assignement

Variable | Event handling | HIPRO-N | HIPRO-S | BUSCOM | 3964R

○ No Import/Export

● Export
Base addresse: 0
☑ Set relative address: 15

○ Import
Base addresse: 0
□ Set relative address: 0

○ Import/Export
Base addresse: 1000
□ Set relative address: 0

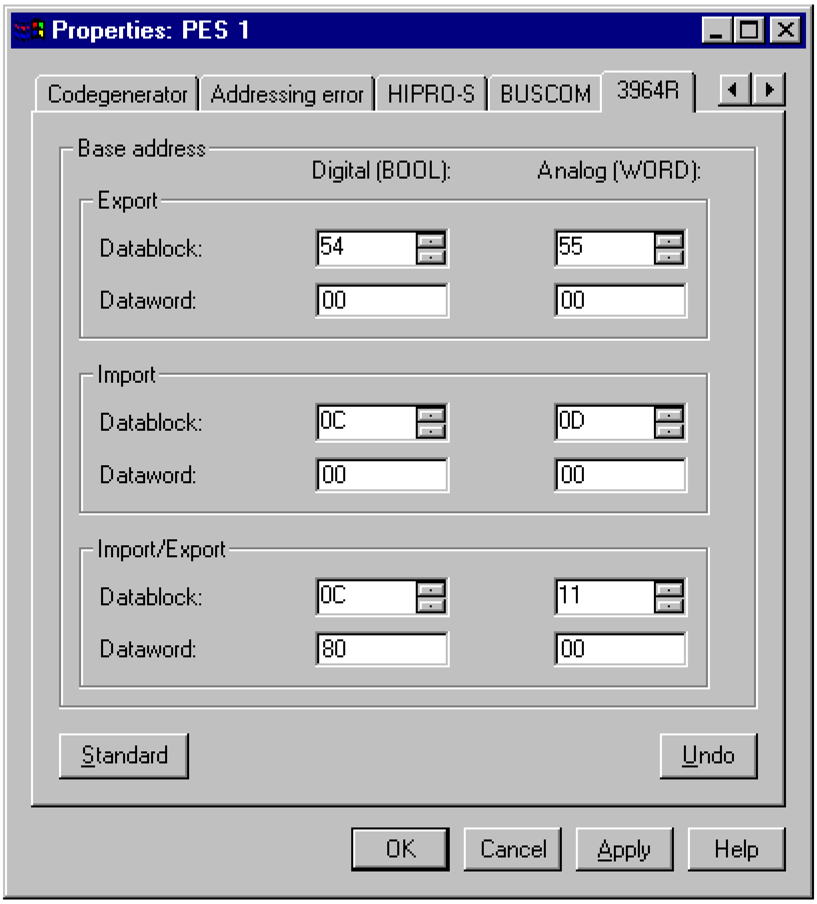
HW assignment, BUSCOM

Note
For each entry the address range is 0..6142. The codegenerator moves the addresses 0..2047 to 0..2047 and 2048..6142 to 4096..8192 automatically.

5.2 3964R Communication

HIMA PESs support the 3964R protocol. The data blocks can be defined individually for exporting and importing

.



Base address

	Digital (BOOL):	Analog (WORD):
Export		
Datablock:	54	55
Dataword:	00	00
Import		
Datablock:	0C	0D
Dataword:	00	00
Import/Export		
Datablock:	0C	11
Dataword:	80	00

Properties resource

The variables and their relative addresses are configured in the variables declaration.

Code

6 Generating the program code

Once you have created the program and have function tested it with off-line simulation, you must generate the program code before you can load the program into the PES.

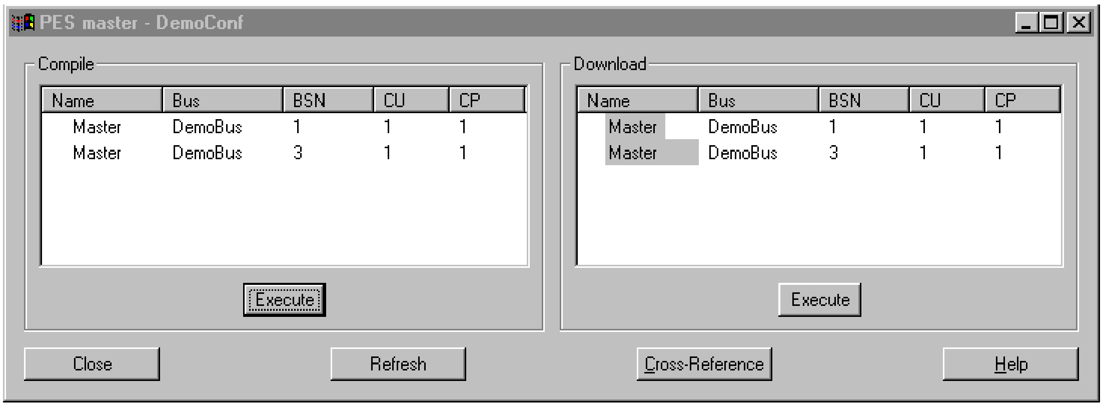
6.1 Code generator

The code generator is started from the context menu of the resource in which the program will be loaded. All error and status messages generated during the compile run will be displayed in a separate window.
After the compile run, the version numbers of the project will be shown in this window.

- Code version
  Changes when the code is changed
- Program version (A1, H11 and H41/H51 systems only)
  same as codeversion but decimal display
- Data version
  Changes only when variables are changed
- Area version
  Changes only when the I/O level is changed
- Run version
  Changes when the code and parameters are changed.

6.2 Compiling and loading PES masters

Having defined variables for data interchange and generated code for the resources, you must now generate the code for the PES masters. To do this, select ´PES masters´ [´PES-Master´] in the context menu of the configuration. You will see the defined PES masters:



Compile

Name	Bus	BSN	CU	CP
Master	DemoBus	1	1	1
Master	DemoBus	3	1	1

Download

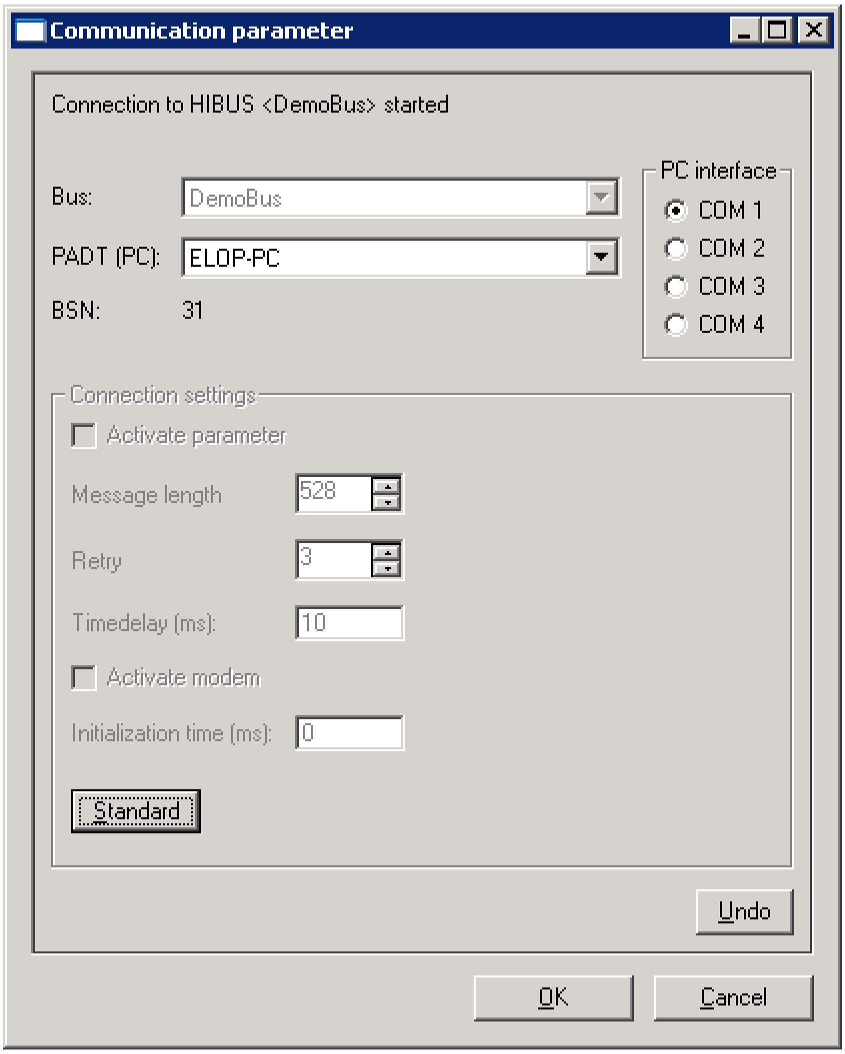
Name	Bus	BSN	CU	CP
Master	DemoBus	1	1	1
Master	DemoBus	3	1	1

PES master Compile and Download

All the defined PES masters are listed in the ´Compile´ list box. The PES masters which still have to be compiled are highlighted in grey.
The ´Download´ list box contains all the PES masters that have already been compiled. The grey highlighted PES masters are the ones which are not currently loaded in the coprocessor module.

You can now compile and download the PES masters, update the data in the list boxes and open a cross-reference window showing the variables transferred by the PES masters.

- ´Execute´ under the Compile list box:
  Codes generated for the selected PES masters.
- Execute´ under the Download list box:
  Selected PES masters are loaded. A dialogue window opens to allow you to specify the communication parameters of the PADT.



Communication parameter

Connection to HIBUS <DemoBus> started

Bus: DemoBus

PADT (PC): ELOP-PC

BSN: 31

PC interface

● COM 1
○ COM 2
○ COM 3
○ COM 4

Connection settings

□ Activate parameter

Message length 528

Retry 3

Timedelay (ms): 10

□ Activate modem

Initialization time (ms): 0

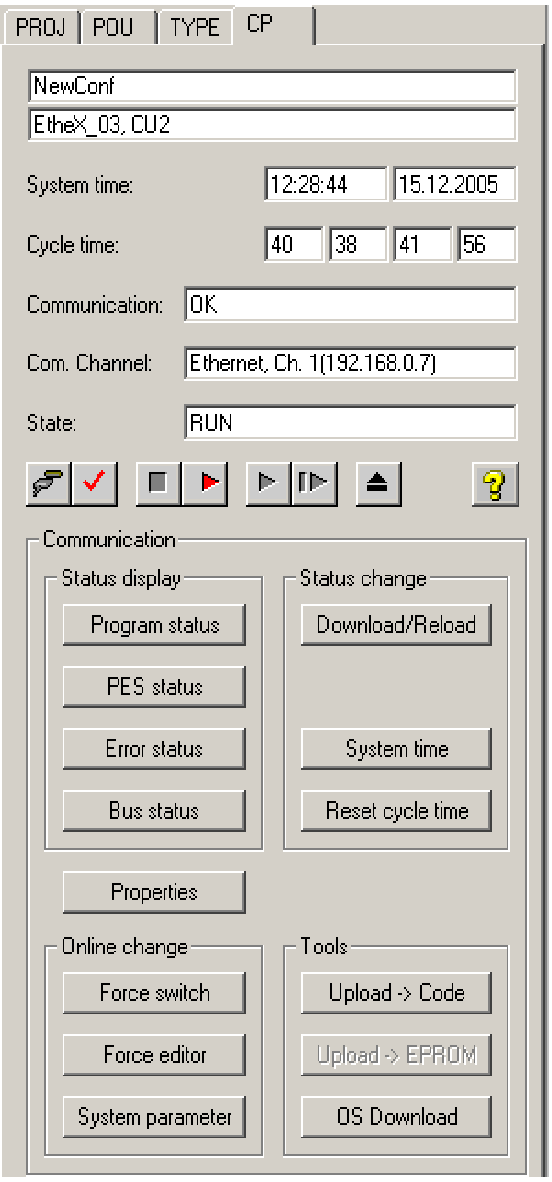
Communication parameter

Here you select the PES master you want from a list box and define the PC interface you wish to use for data transfer.

6.3 Loading the program into the controller

The Control Panel is used to load the program into the controller and to operate the PES. You start it from the context menu of the appropriate resource.

Note:
Within a configuration, several control panels (CPs) and online tests (OLTs) can be opened simultaneously.
If an application program download/reload or an operating system download to a resource has to be carried out, then all CPs and OLTs of the other resources have to be closed.



NewConf

EtheX_03, CU2

System time: 12:28:44 15.12.2005

Cycle time: 40 38 41 56

Communication: OK

Com. Channel: Ethernet, Ch. 1(192.168.0.7)

State: RUN

Control panel

The control panel displays the following information:

- Name of the configuration
- Name of the resource and central module (CM1 or CM2)
- Time and date of the PES
- Cycle time of the PES (current, minimum, average, maximum)
- Communication status (OK, no connection)
- Communication channel (Ethernet, Ch1 or Ch2) or (Serial, COMx)

Status of the PES

Display	Meaning
No connection	No connection to the PES, data interchange is interrupted
Mono	PES is in RUN mode, but a central module has gone down in the redundant controller
RUN	PES is in RUN mode
PGSTOP, outputs on LOW	PES was stopped by the ELOP II station and the outputs were reset
PGSTOP, outputs held	PES was stopped by the ELOP II station, outputs not reset
Break point, outputs held	A break point was reached during program run, the system was stopped and the outputs not reset
ERROR STOP	PES was stopped by the operating system because of an error and all outputs have been reset

The operating buttons have the following meaning

Button	Meaning
🔌	Initializing the communication after a connection loss
✓	Acknowledge of denied actions (blinking if acknowledge acquired).
■	Stop the PES. Outputs can be held or reset.
▶	Start the PES. Coldstart, Warmstart or Hotstart possible
▷	Continue after a break point occurred in the PES
|▷	Cycle step of the PES (only in Stop mode). You can execute precisely one cycle for test purpose.
⏏	Quit the control panel
?	Online help for the control panel

Table 2: Buttons inside the control panel

Status display
In this area you find the functions for the different information about the program, the PES and the bus.

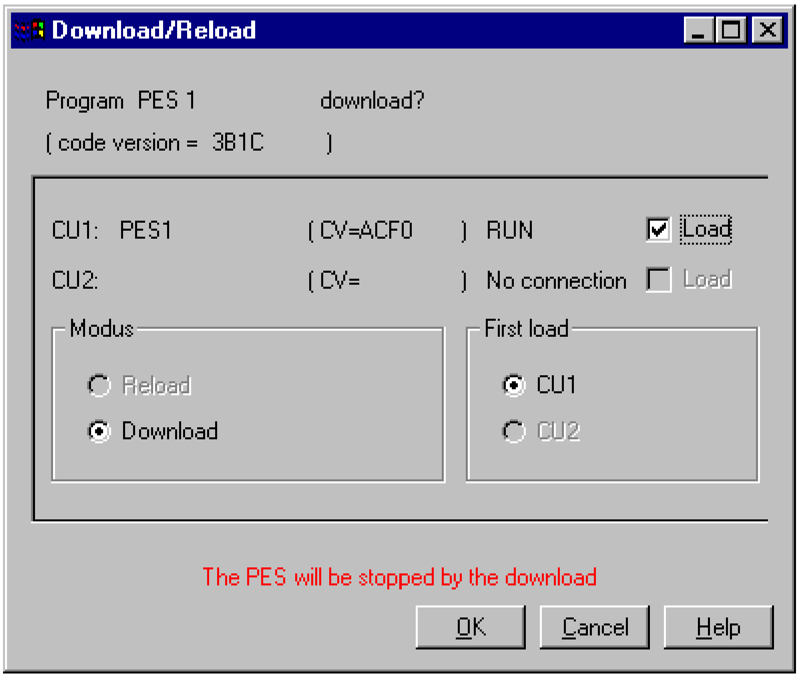
Status change
In this area you find the functions for downloading the generated code, the change of the system time and the resetting of the calculated cycle times.

Online change
In this area you find the functions for setting and resetting the force switches, display and change of the system parameters.

Tools
In this area you find the functions for uploading the code, for creating a file for programming an EPROM (H41/H51 only) and for downloading the operating system (H41q/H51q only).

6.4 Download/Reload

Click on the ´Download/Reload´ button to open a window in which you can select which central module will be loaded in which mode.



Download/Reload

Program PES 1 download?

(code version = 3B1C)

CU1: PES1 (CV=ACF0) RUN ☑ Load

CU2: (CV=) No connection □ Load

Modus

○ Reload

● Download

First load

● CU1

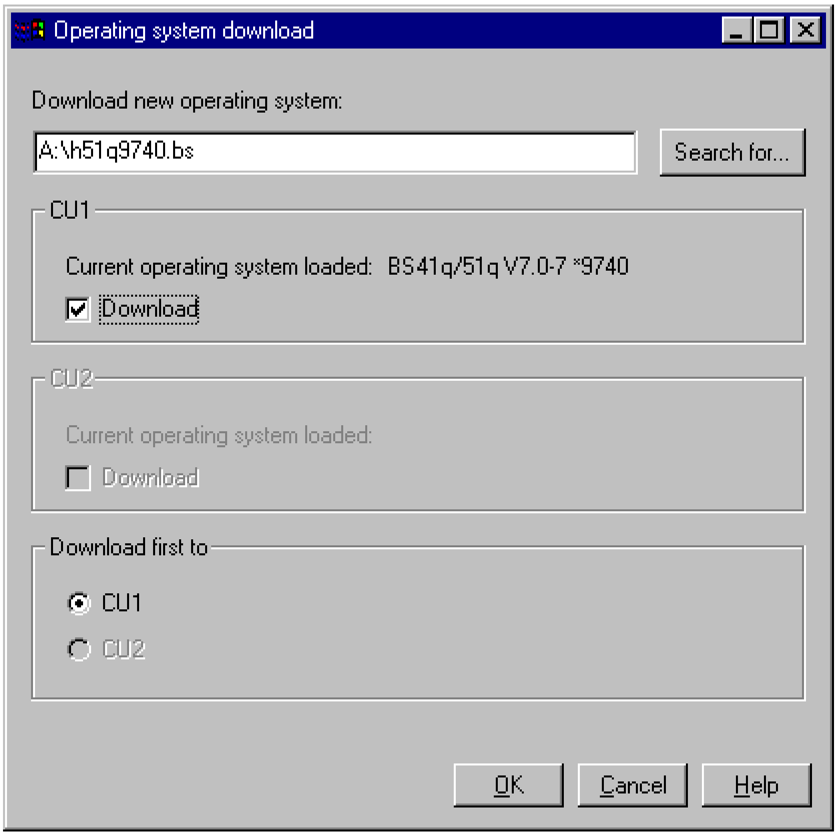
○ CU2

The PES will be stopped by the download

Download/Reload

6.4.1 OS Download

You can update the operating system in PES H41q/H51q with this menu option. You will need an updated operating system file to do this.



Operating system download

Download new operating system:

A:\h51q9740.bs

CU1

Current operating system loaded: BS41q/51q V7.0-7 *9740

☑ Download

CU2

Current operating system loaded:

□ Download

Download first to

● CU1

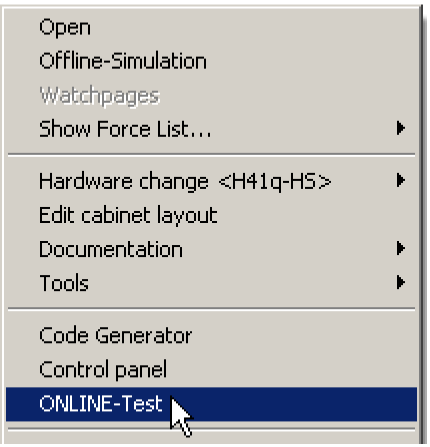
○ CU2

OS-Download, H41q and H51q

7 Online Test

7.1 Starting the online test

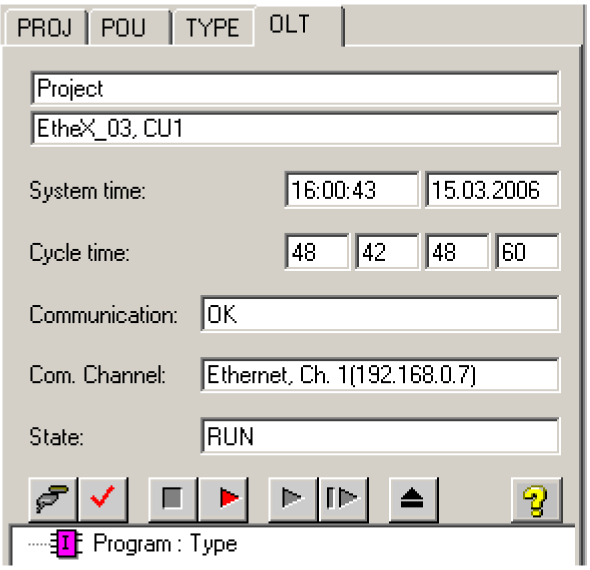
Start the online test via the contect menu of the resource.



Context menu of the resource

You will now see the online test window.

7.2 OLT Window



Project

EtheX_03, CU1

System time: 16:00:43 15.03.2006

Cycle time: 48 42 48 60

Communication: OK

Com. Channel: Ethernet, Ch. 1(192.168.0.7)

State: RUN

Program : Type

OLT window

The OLT window displays the following information

- Name of the configuration
- Name of the resource
- Time and date of the PES
- Cycle time of the PES (current, minimum, average, maximum)
- Communication status (OK, no connection)
- Com. Channel (Ethernet, Ch1 or Ch2) or (Serial, COMx)

Status of the PES

Display	Meaning
No connection	No connection to the PES, data interchange is interrupted
Mono	PES is in RUN mode, but a central module has gone down in the redundant controller
RUN	PES is in RUN mode
PGSTOP, outputs on LOW	PES was stopped by the ELOP II station and the outputs were reset
PGSTOP, outputs held	PES was stopped by the ELOP II station, outputs not reset
Break point, outputs held	A break point was reached during program run, the system was stopped and the outputs not reset
ERROR STOP	PES was stopped by the operating system because of an error and all outputs have been reset

Table 3: Status

7.2.1 Operating Buttons

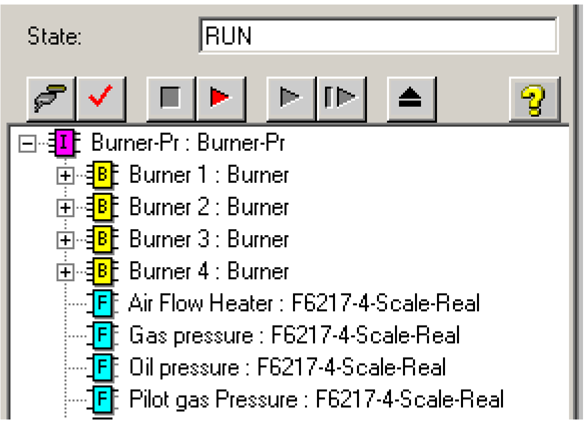
Explanation of the buttons from left to right.

Button	Meaning
🔌	Initializing the communication after a connection loss
✓	Acknowledge of denied actions (blinking if acknowledge acquired).
■	Stop the PES. Outputs can be held or reset.
▶	Start the PES. Coldstart, Warmstart or Hotstart possible
▷	Continue after a break point occurred in the PES
|▷	Cycle step of the PES (only in Stop mode). You can execute precisely one cycle for test purpose.
⏏	Quit the online test
?	Online help for the online test

Table 4: Operating buttons

7.2.2 Program Structure Tree

The program structure tree is used to open the online test of the program or of individual blocks.



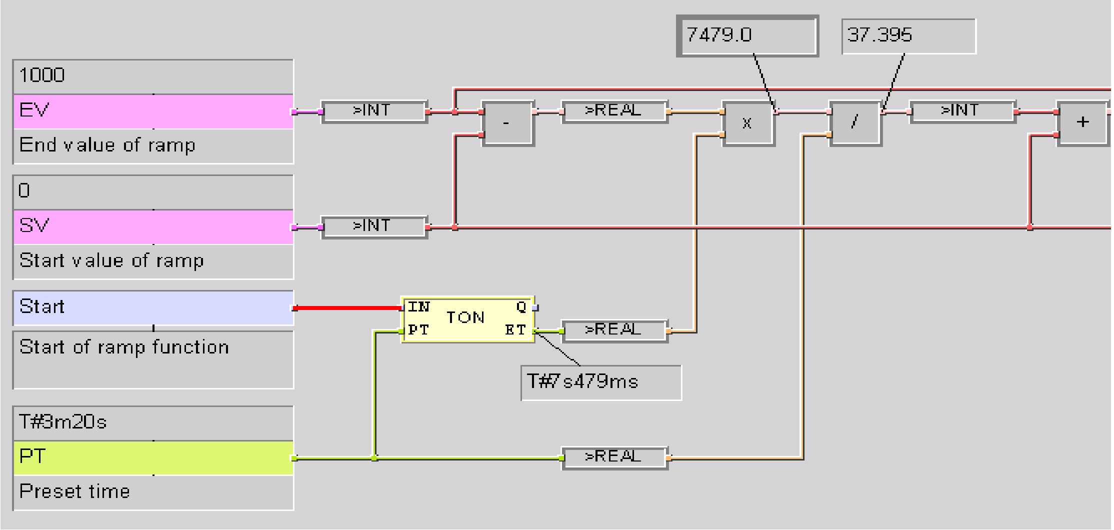
State: RUN

- Burner-Pr : Burner-Pr
  - Burner 1 : Burner
  - Burner 2 : Burner
  - Burner 3 : Burner
  - Burner 4 : Burner
  - Air Flow Heater : F6217-4-Scale-Real
  - Gas pressure : F6217-4-Scale-Real
  - Oil pressure : F6217-4-Scale-Real
  - Pilot gas Pressure : F6217-4-Scale-Real

Structure tree of the OLT window

Select the program (violet symbol) or the desired block (yellow symbol for a function block, blue symbol for a function) with the mouse, then double-click on the symbol.

7.3 Functions in the ONLINE Test

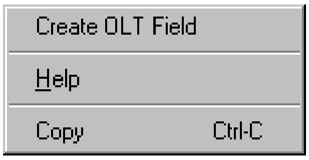


Online test opened

The status of a BOOL type variable is indicated by the color of the interconnecting lines. Blue means FALSE and red means TRUE.

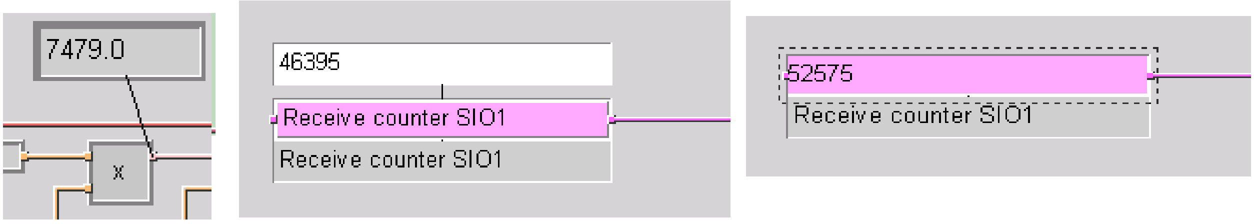
7.3.1 OLT Fields

The status of a variable can also be displayed in an OLT field that can be generated. An OLT field is always needed to display values (all data types except BOOL). You can generate OLT fields at any node or directly on a variable. Position your mouse pointer on the desired node or variable and call the context menu by clicking the right-hand mouse button once.



Drag the OLT field to the desired position and click the left-hand mouse button. The OLT field is now ready to display the value or status.

The value of a variable can be displayed, by moving the cursor above the value field and pressing ALT and left mouse button.



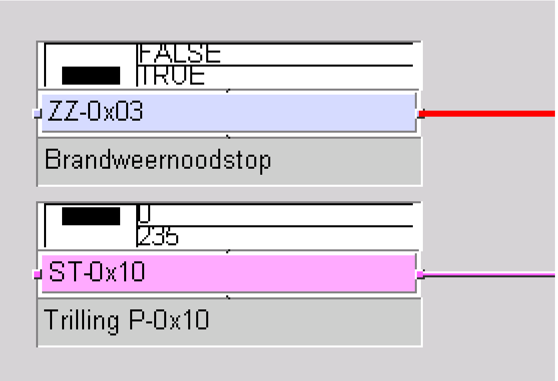
OLT Fields and value displays

7.3.2 Forcing I/Os

In the online test you can force all variables that are defined as physical input or outputs, or local variables that are not overwritten by the program.
When an input variable is forced, then this forced value will be used throughout the logic.
When an output variable is forced however, only the physical output is forced. If the output is interrogated in the logic, then the value defined by the logic will be used.
Forcing can be done directly in the logic with the OLT field, or it can also be done using the Force graphic in the Control Panel (CP).

7.3.2.1 Forcing IO in the OLT field

The OLT field for the IO is divided up into two columns and two lines. The left hand column contains a square.
An empty square means that the force main switch is off. A filled square means that the force main switch is on.
A square on the first line means the variable is not forced. A (filled) square on the second line means the variable is forced.

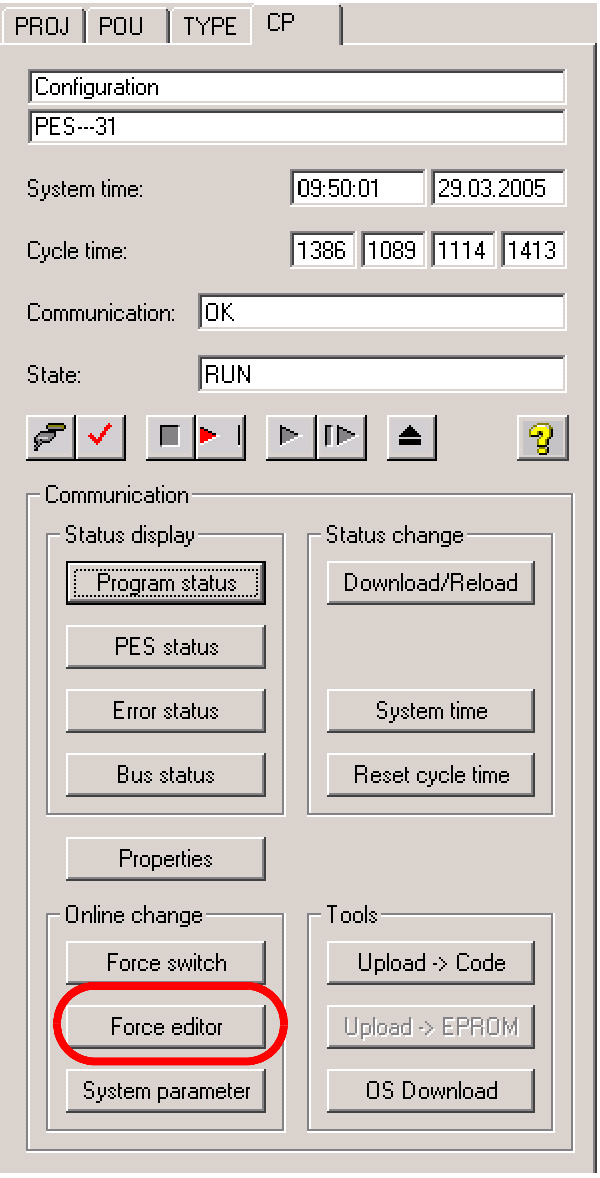


OLT field

The value of the variable of the input or output is shown on the second column, first line.
The force value is shown on the second line.

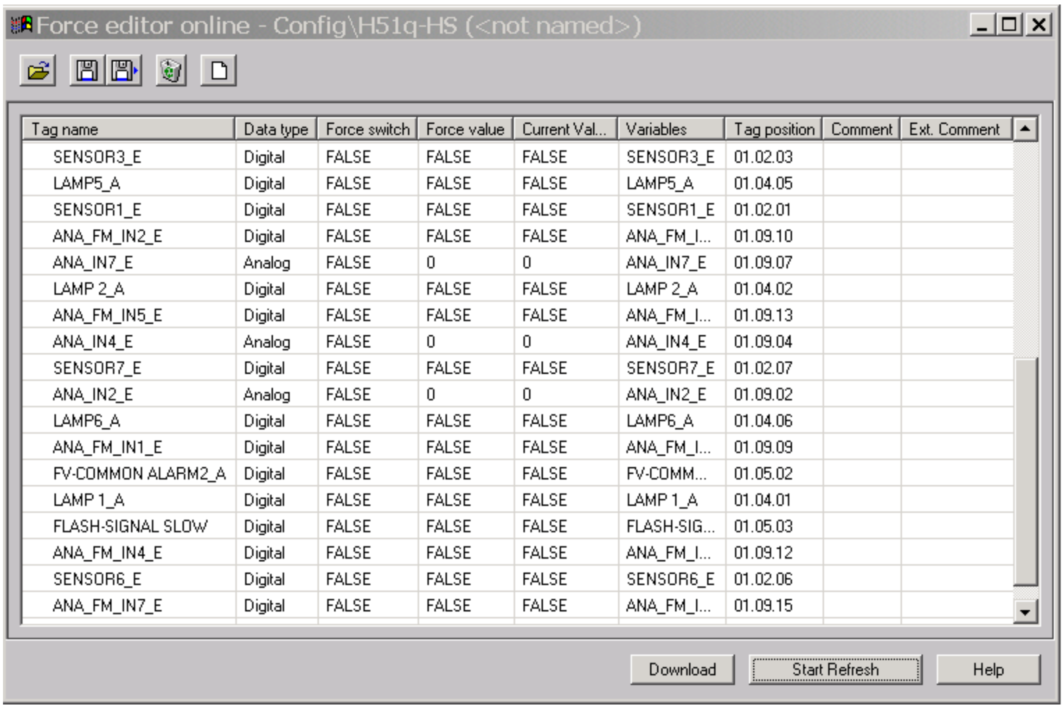
7.3.2.2 Forcing IO in the Force graphic

Call the Force graphic with the Control Panel´s ´Force image´ button.



Control Panel

The force editor shows all variables that are combined with a tag name. The force editor has the following columns:
Tag name, data type, Force switch, Force value, current value, variables, tag position, comment and ext. comment.
You can toggle the Force switch between FALSE and TRUE simply by double clicking with the mouse.
The digital (Boolean) force values can be changed in the same way. Double click on an analog (word) force value to change it, then confirm with Enter.



Tag name	Data type	Force switch	Force value	Current Val...	Variables	Tag position	Comment	Ext. Comment
SENSOR3_E	Digital	FALSE	FALSE	FALSE	SENSOR3_E	01.02.03		
LAMP5_A	Digital	FALSE	FALSE	FALSE	LAMP5_A	01.04.05		
SENSOR1_E	Digital	FALSE	FALSE	FALSE	SENSOR1_E	01.02.01		
ANA_FM_IN2_E	Digital	FALSE	FALSE	FALSE	ANA_FM_I...	01.09.10		
ANA_IN7_E	Analog	FALSE	0	0	ANA_IN7_E	01.09.07		
LAMP 2_A	Digital	FALSE	FALSE	FALSE	LAMP 2_A	01.04.02		
ANA_FM_IN5_E	Digital	FALSE	FALSE	FALSE	ANA_FM_I...	01.09.13		
ANA_IN4_E	Analog	FALSE	0	0	ANA_IN4_E	01.09.04		
SENSOR7_E	Digital	FALSE	FALSE	FALSE	SENSOR7_E	01.02.07		
ANA_IN2_E	Analog	FALSE	0	0	ANA_IN2_E	01.09.02		
LAMP6_A	Digital	FALSE	FALSE	FALSE	LAMP6_A	01.04.06		
ANA_FM_IN1_E	Digital	FALSE	FALSE	FALSE	ANA_FM_I...	01.09.09		
FV-COMMON ALARM2_A	Digital	FALSE	FALSE	FALSE	FV-COMM...	01.05.02		
LAMP 1_A	Digital	FALSE	FALSE	FALSE	LAMP 1_A	01.04.01		
FLASH-SIGNAL SLOW	Digital	FALSE	FALSE	FALSE	FLASH-SIG...	01.05.03		
ANA_FM_IN4_E	Digital	FALSE	FALSE	FALSE	ANA_FM_I...	01.09.12		
SENSOR6_E	Digital	FALSE	FALSE	FALSE	SENSOR6_E	01.02.06		
ANA_FM_IN7_E	Digital	FALSE	FALSE	FALSE	ANA_FM_I...	01.09.15		

Force image

7.3.2.3 Function buttons and functions in the Force graphic

Explanations of the individual buttons are given in the table below:

Button	Meaning
📂	This button opens a Force graphic which has been previously saved.
💾	Save: when you are making changes to a Force graphic and want to save them, save the graphic with this button.
💾	Save As: use this button to save your Force graphic under a new name
🗑	Delete: this button deletes a previously saved Force graphic which is no longer needed
🗋	Reset Force values: this button is used to set all Force switches in the graphic to FALSE, the Force values of the digital (Boolean) variables to FALSE and of the analog (word) variables to 0.
Download	Download: use this button to transfer your Force graphic from the ELOP II station to the PES.
Start Refresh	Upload: use this button to load the currently active Force graphic from the PES to the ELOP II station.
Help	Help: activate this button to call online help

Table 5: Buttons of the force image

Documentation

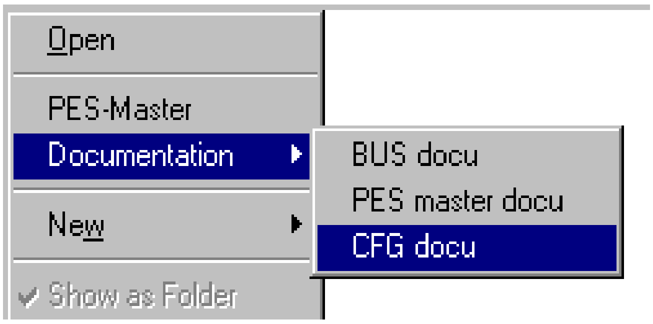
8 Documentation in the configuration

You will find in the context menu of the configuration and the resource apart from the base functions contents and print (refer to the base manual) an additional item documentation.

8.1 Documentation functions of the configuration

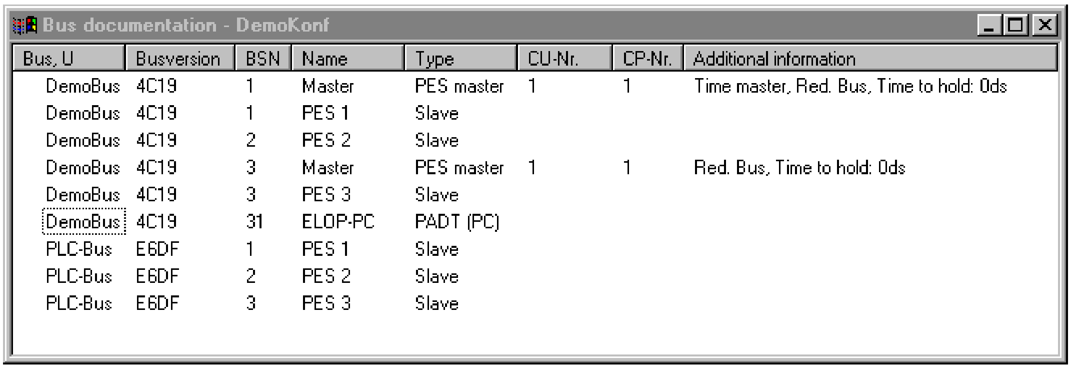
In the context menu of the configuration you will find in documentation the following items:

- BUS docu
- PES master docu
- CFG docu



Context menu of the configuration

8.1.1 Bus docu



Bus, U	Busversion	BSN	Name	Type	CU-Nr.	CP-Nr.	Additional information
DemoBus	4C19	1	Master	PES master	1	1	Time master, Red. Bus, Time to hold: 0ds
DemoBus	4C19	1	PES 1	Slave			
DemoBus	4C19	2	PES 2	Slave			
DemoBus	4C19	3	Master	PES master	1	1	Red. Bus, Time to hold: 0ds
DemoBus	4C19	3	PES 3	Slave			
DemoBus	4C19	31	ELOP-PC	PADT (PC)			
PLC-Bus	E6DF	1	PES 1	Slave			
PLC-Bus	E6DF	2	PES 2	Slave			
PLC-Bus	E6DF	3	PES 3	Slave			

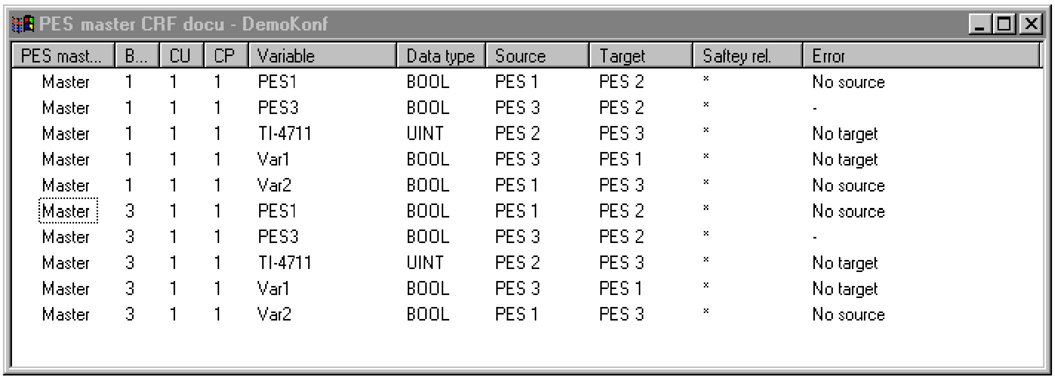
Bus documentation of the configuration

Bus Docu is an online documentation that provides information about your bus definitions. The table below gives explanations about the 8 columns that are displayed:

Column	Meaning
Bus	Name of the bus overview, the display can be in ascending or descending alphabetical order.
Bus version	The system creates a version number with your bus overview. This version number changes when you make changes to your bus overview.
BSN	Bus station number: each system that is connected to the bus is assigned a bus station number. This number can only occur once on the bus.
Name	Name of the connected system. The resource name must be used for the PES.
Type	System type. There are 3 types: Slave, PC master and PES master.
CU No.	Number (1 or 2) of the central module assigned to the PES master. Left central module = 1 and right central module = 2.
CM No.	Number of the coprocessor module used as the PES master. Slot 1, 2 or 3 next to the central module.
Additional information	Extra information.

Table 6: Key to column

8.1.2 PES Master Docu



PES mast...	B...	CU	CP	Variable	Data type	Source	Target	Saftey rel.	Error
Master	1	1	1	PES1	BOOL	PES 1	PES 2	×	No source
Master	1	1	1	PES3	BOOL	PES 3	PES 2	×	-
Master	1	1	1	TI-4711	UINT	PES 2	PES 3	×	No target
Master	1	1	1	Var1	BOOL	PES 3	PES 1	×	No target
Master	1	1	1	Var2	BOOL	PES 1	PES 3	×	No source
Master	3	1	1	PES1	BOOL	PES 1	PES 2	×	No source
Master	3	1	1	PES3	BOOL	PES 3	PES 2	×	-
Master	3	1	1	TI-4711	UINT	PES 2	PES 3	×	No target
Master	3	1	1	Var1	BOOL	PES 3	PES 1	×	No target
Master	3	1	1	Var2	BOOL	PES 1	PES 3	×	No source

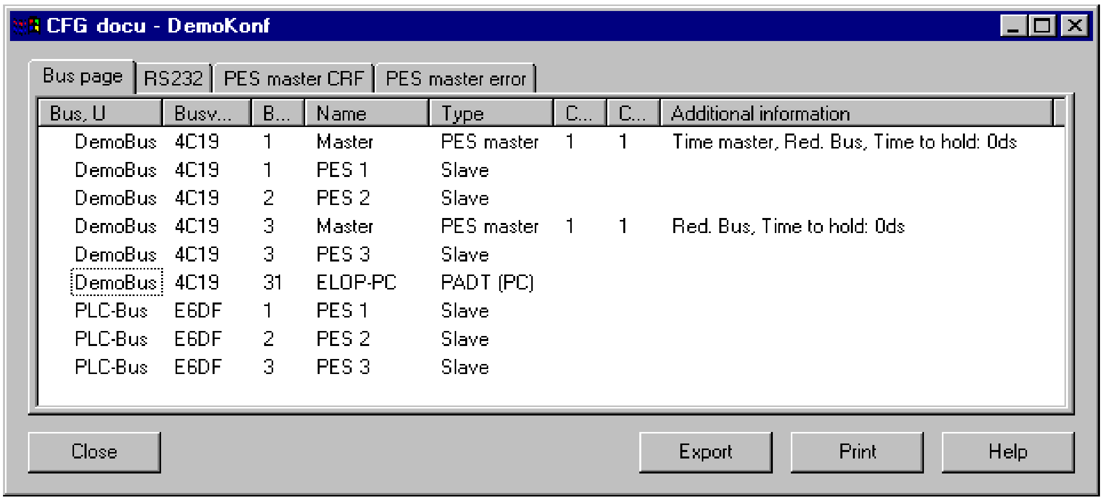
PES master documentation of the configuration

PES Master Docu is an online documentation that gives you information about your bus definitions. The table below gives explanations about the 10 columns that are displayed:

Column	Meaning
PES Master	Name of the PES master project. The display is in ascending or descending alphabetical order
BSN	Bus station number of the PES master
CU	Number of the central module assigned to the PES master. 1 = Left central module, 2 = right central module.
CM	Slot of the PES master (coprocessor module), 1, 2 or 3
Variable	Name of the variable transmitted from one HIMA PES to another HIMA PES.
Data type	Type of the variable (BOOL, WORD or UINT)
Source	Resource name of the source PES
Target	Resource name of the target PES
Safety	Flags whether variable transmission is safety related or non-safety related
Error	Error information: No source available Several sources available No target available Diff. data blocks

Table 7: Key to columns

8.1.3 CFG Docu



Bus, U	Busv...	B...	Name	Type	C...	C...	Additional information
DemoBus	4C19	1	Master	PES master	1	1	Time master, Red. Bus, Time to hold: 0ds
DemoBus	4C19	1	PES 1	Slave			
DemoBus	4C19	2	PES 2	Slave			
DemoBus	4C19	3	Master	PES master	1	1	Red. Bus, Time to hold: 0ds
DemoBus	4C19	3	PES 3	Slave			
DemoBus	4C19	31	ELOP-PC	PADT (PC)			
PLC-Bus	E6DF	1	PES 1	Slave			
PLC-Bus	E6DF	2	PES 2	Slave			
PLC-Bus	E6DF	3	PES 3	Slave			

CFG documentation of the configuration

CFG documentation consists of:

- Bus list, this bus list is the same as the bus documentation
- RS232, transfer parameters such as baud rate, stop bits, parity
- PES master CRF, same as PES Master Docu
- PES master error, error information of the PES Master Docu

In the CFG docu the following actions can be performed:

Button	Meaning
Close	Close the CFG docu
Export	Export the CFG docu in an ASCII file
Print	Print CFG docu
Help	Start help for CFG docu

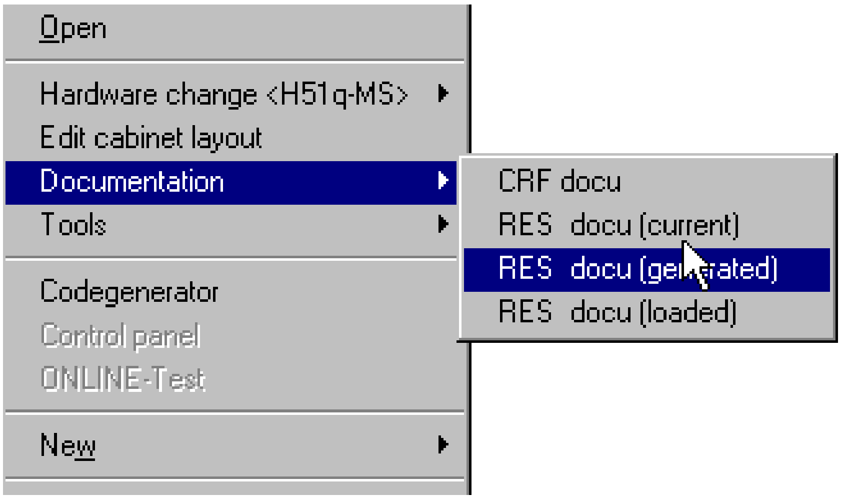
Table 8: Function keys in the CFG docu

8.2 Documentation functions of the resource

In the context menu of the resource you will find in documentation the following items:

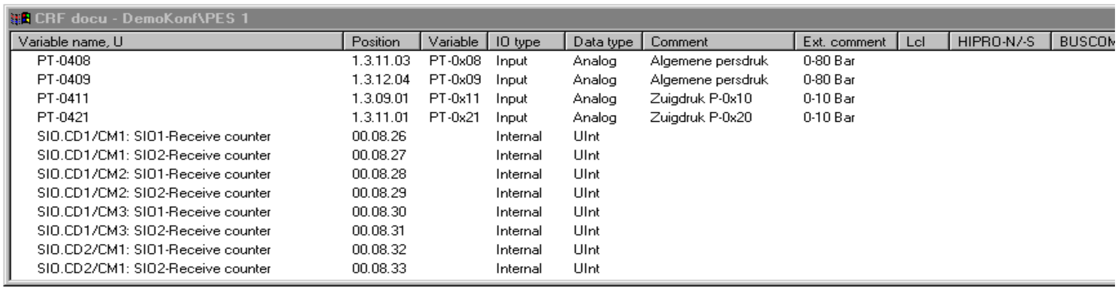
- CRF docu
- RES docu (current)
- RES docu (generated)
- RES docu (loaded)

.



Context menu of the resource

8.2.1 CRF Docu


CRF docu - DemoKonf\PES 1

Variable name, U	Position	Variable	IO type	Data type	Comment	Ext. comment	Lcl	HIPRO-N/-S	BUSCOM
PT-0408	1.3.11.03	PT-0x08	Input	Analog	Algemene persdruk	0-80 Bar			
PT-0409	1.3.12.04	PT-0x09	Input	Analog	Algemene persdruk	0-80 Bar			
PT-0411	1.3.09.01	PT-0x11	Input	Analog	Zuigdruk P-0x10	0-10 Bar			
PT-0421	1.3.11.01	PT-0x21	Input	Analog	Zuigdruk P-0x20	0-10 Bar			
SIO.CD1/CM1: SIO1-Receive counter	00.08.26		Internal	UInt					
SIO.CD1/CM1: SIO2-Receive counter	00.08.27		Internal	UInt					
SIO.CD1/CM2: SIO1-Receive counter	00.08.28		Internal	UInt					
SIO.CD1/CM2: SIO2-Receive counter	00.08.29		Internal	UInt					
SIO.CD1/CM3: SIO1-Receive counter	00.08.30		Internal	UInt					
SIO.CD1/CM3: SIO2-Receive counter	00.08.31		Internal	UInt					
SIO.CD2/CM1: SIO1-Receive counter	00.08.32		Internal	UInt					
SIO.CD2/CM1: SIO2-Receive counter	00.08.33		Internal	UInt					

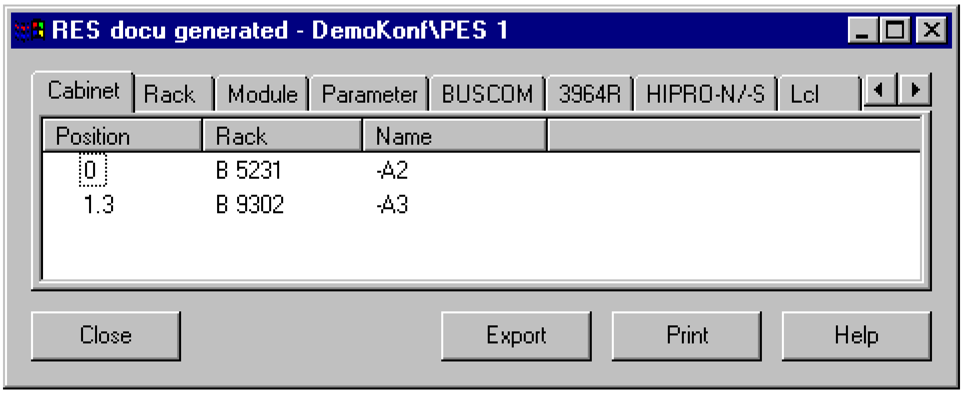
CRF Docu

CRF Docu is an online documentation that gives you information about the inputs and outputs. It comprises 11 columns. The table below contains further information:

Column	Meaning
Tag name	Name of the input/output in the hardware assignment, or the name of the system variable
Position	Position and channel of the input/output. The first digit is the cabinet and next digit is the subrack for H51 and H51q systems, followed by the slot, and finally the channel. 0 is always output for cabinet and subrack for H41, H41q, H11 and A1.
Variable	Name of the variable as defined in the program. May differ from the tag name, but is usually the same.
I/O type	This indicates whether it is an input, an output or a system variable (internal).
Data type	I/O data types can be digital (Boolean variables) and analog (UINT variables). Data types for system variables are BOOL and UINT
Commentary	Shows the long name from the variable declaration
Ext. Commentary	Shows the commentary from the variables declaration
Lcl	Flags whether I/O is used for Lcl E for Event P for Logging Lcl
HIPRO-N/-S	Flags that I/O is transmitted by HIPRO -/E HIPRO-N Export to PES master -/I HIPRO-N Import from PES master IS/- HIPRO-S Import from PES master ES/- HIPRO-S Export to PES master
BUSCOM	Flags that I/O is transmitted via BUSCOM R master reads variable W master writes variable
3964R	Flags that I/O is transmitted by 3964 R

Table 9: Explanations of CRF Docu

8.2.2 RES Docu



Position	Rack	Name
0	B 5231	-A2
1.3	B 9302	-A3

RES Docu

RES Docu consists of:

- Cabinet
  the documentation of the cabinet contains the cabinet number and the subrack number of the used subrack, the short name (column rack) and the name (input in edit cabinet)
- Rack
  the documentation of the rack contains the cabinet number, the subrack number, the position of the used subracks and modules, the short names and names.
- Module
  the documentation of the module contains all subrack, modules and channels.
- Parameters (system parameters)
  the documentation of parameters contains all settings of the system parameters
- BUSCOM
  the documentation of BUSCOM contains all variables which are transferred via a bus system, currently MODBUS and the addresses.
- 3964R
  the documentation of 3964R contains all variables which are transferred via a bus system with the protocol 3964 R and the addresses.
- HIPRO-N/-S
  the documentation of HIPRO contains all variables which are transferred via HIPRO with the concerning PES master
- Lcl
  the documentation of Lcl contains all variables which are printed via Lcl with the text for status TRUE and FALSE.
- CRF (tag name, position, variable name, I/O type, data type, commentary, extra commentary, Lcl, HIPRO, BUSCOM, 3964R)

You can perform the following actions in RES Docu:

Button	Meaning
Close	Close the RES Docu
Export	Export the RES Docu to an ASCII document
Print	Print off RES Docu
Help	Call online help

Table 10: Function buttons in RES Docu

RES Docu displays your current project status. This may well be different from RES Docu (generated) or RES Docu (loaded).

8.2.3 RES Docu (generated)

RES Docu (generated) is based on the resource data with which the code was generated.
CG Error is output in addition to RES Docu. CG Error is the same as the “Program.ERR” file.
This file contains the code generator messages.
All other points are the same as for RES Docu.

8.2.4 RES Docu (loaded)

RES Docu (loaded) is the same as RES Docu, but is based on the resource data that was loaded into the PES.

Safety aspects

9 Safety related functions

ELOP II has safety related functions, which guarantee for the system family H41q/H51q that

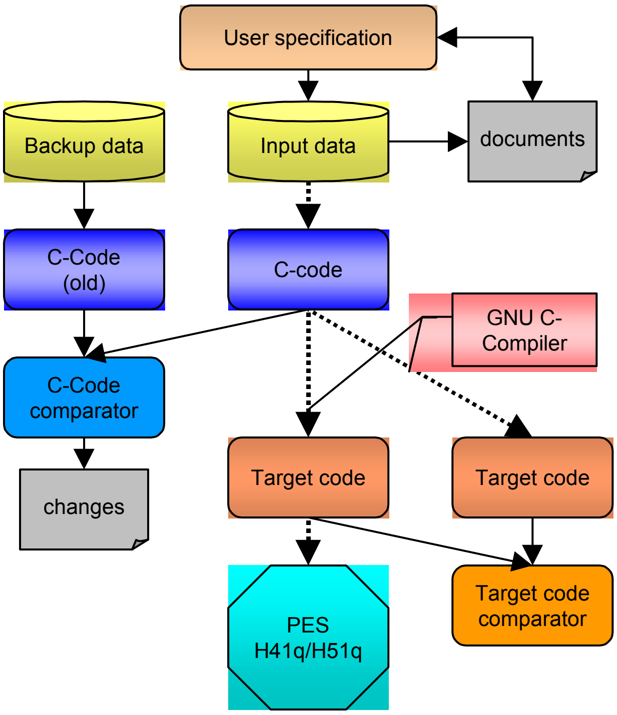
- the programming system (PS) works correctly, i.e. no error occurs on the programming system
- the PS is used correctly, which means that user error are excluded

During the first start up of a safety related PES (programmable electronic system) a complete function test is necessary to proof the safety of the system
After modification of an application it is only necessary to perform a function test for the modifications. A complete function test is **not** required.

The important functions for this in ELOP II are:

- C-code comparator
- target code comparator
- proven GNU-C-compiler

The C-code comparator detects changes in the application program. The target code comparator compares two target codes, which have been generated by the proven GNU-C-compiler in two different tasks. By this errors of the non safety related PC can be detected.



Principle scheme for the code generation.

9.1 Building an application

In this chapter you will find a basic overview on the procedure to build an application. Please also refer to the safety manual for the system family H41q/H51q.

9.1.1 Create the logic diagrams

The program is built based on a specification for the application, which should be available as specification or function requirements. First the basic functions are programmed in ELOP II using POU’s function and function block. Using these basic function you then built more complex functions which are used to build up the program. Application parts which function do not depend on each other have to be separated in different functions or function blocks.

The main targets building your application should be:

- Easy to understand
- Easy to follow up
- Easy to modify

ELOP II provides several non safety related functions such as version management, revision control in the documentation object and off-line simulation to assist in the design of logic diagrams. After a modification the user name (=user in Windows) and the date are stored automatically (see also ´revision service´).

9.1.2 Code generation

After the program is assigned to the resource, you should check the settings for the code generator (See “Code generator” on page 28.). To avoid random errors of the not safety related PC, please activate the target code comparator. To allow the safety related determination of future modifications please select to generate a code compare image.
Of course you should check also the other settings of the resource especially the safety parameter (See “Safety” on page 22.)

9.1.3 Download

For the download it is important that you select the bus configuration which matches the real bus connection. By this a download to the wrong PES is avoided, due to the bus station number.

9.1.4 Function check of the PES

After downloading the program to the resource a complete function test must be performed in order to check the correct design according to the original specification. Again it is helpful to check the base function first and afterwards the more complex functions.
The function test should be carried out by a person which was not involved in the program design. The check can be documented by selecting ´Is checked´ in the context menu of the program instance for each page. The user name (=user in Windows) and the date are stored automatically (see also ´revision service´).
Do not forget to backup the data.

9.2 Modification of an application

In this chapter you will find a basic overview on the procedure to modify an application. Please also refer to the safety manual for the system family H41q/H51q.

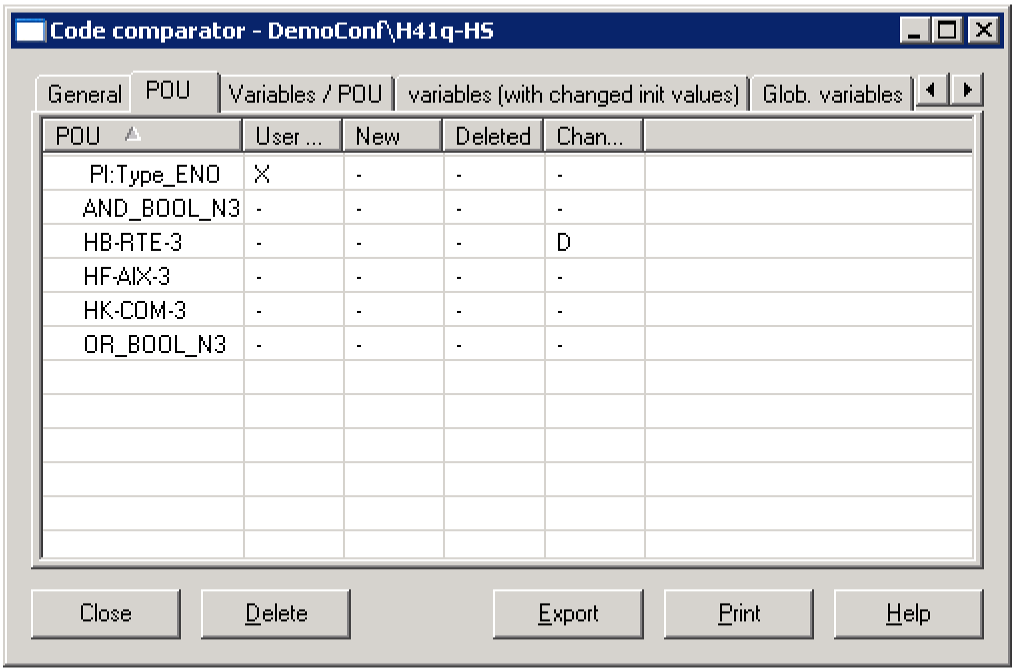
9.2.1 Modification of logic diagrams

The program is modified according to specification for the application, which should be available as specification or function requirements. You have to follow the basic rules on how to build a logic, especially the separation of application parts which function do not depend on each other in different functions or function blocks.

The check should be carried out by a person which was not involved in the program modification. The modification and the check rest. the release must be documented. ELOP II provides the non safety related functions revision control in the documentation object and revision service for that purpose.

9.2.2 Code generation

In the resource properties you need to select a resource for code image. This can be either a backup of the project on another drive or exactly the same resource. With this setting the system will determine safety related the changes between the current and the code image.
Note:
In order to be able to perform the safety related comparison, it is necessary to generate a code image before during code generation (See “Code generator” on page 28.).
The modifications are displayed per POU.
.


Code comparator - DemoConf\H41q-HS
General
POU
Variables / POU
variables (with changed init values)
Glob. variables
Close
Delete
Export
Print
Help

POU	User ...	New	Deleted	Chan...
PI:Type_ENO	X	-	-	-
AND_BOOL_N3	-	-	-	-
HB-RTE-3	-	-	-	D
HF-AIX-3	-	-	-	-
HK-COM-3	-	-	-	-
OR_BOOL_N3	-	-	-	-

Output of the code comparator

All POU’s are listed and the POE designed by the user are clearly marked. In ´changed´ you will find the modifications per POU (C: code has been modified; D: Data has been modified, e.g. modification of variables). Modifications are also shown for variables, variables with changed init values, global variables, external variables, constant variables and variables with IO assignment are listed in separate tabs. The result of the Code comparator is automatically stored in a file called „H5Cvgl.TXT" within the resource directory

9.2.3 Download

For the download it is important that you select the bus configuration which matches the real bus connection. By this a download to the wrong PES is avoided, due to the bus station number.

9.2.4 Function check of the PES

After downloading the program to the resource the modifications need to be checked in order to prove a correct design.
The function test should be carried out by a person which was not involved in the program modification. The check can be documented by selecting ´Is checked´ in the context menu of the program instance for each page.
Do not forget to backup the data.

Blocks

10 HIMA Standard Blocks

HIMA standard blocks are the blocks that are contained in ELOP-LIB.
They are:

- H8-STA-3, grouping of safety-relevant testable outputs
- H8-UHR-3, date and time
- HA-LIN-3, temperature linearization
- HA-PID-3, PID controllers
- HA-PMU-3, configurable transducers
- HA-RTE-3, watchdog for analog (word) testable input modules F 6213 and F 6214
- HB-BLD-3, modules and line diagnosis of testable outputs
- HB-BLD-4, modules and line diagnosis of testable redundant outputs
- HB-RTE-3, watchdog for testable input modules F 3235, F 3237 and F 3238
- HF-AIX-3, Parameterisation of a channel of the 8-fold input module F 6221
- HF-CNT-3, configuration of counter module F 5220
- HF-TMP-3, configuration of thermo couple module F 6220
- HK-AGM-3, H51, H51q PES master supervision
- HK-COM-3, Parameterisation and monitoring of the communication modules F 8625, F 8626, F 8627(X) and F 8628(X)
- HK-LGP-3, Lcl evaluation and configuring
- HK-MMT-3, MODBUS master with telephone modem
- HZ-DOS-3, diagnosis without safety
- HZ-FAN-3, error display of testable I/O modules

For closer information to the standard blocks please refer to the **ELOP II** online help.

Lists and References

11 Resource Types

The selection of resource types is called from the resource context menu. Find the *RT Assignment* option if no resource is assigned yet. If a resource is already assigned, this option is replaced by the *Hardware Change* option.

11.1 Current Resource Types

- A1
- A1dig
- H41qc-M
- H41qc-MS
- H41qc-H
- H41qc-HS
- H41qc-HR
- H41qc-HRS
- H41qce-M
- H41qce-MS
- H41qce-H
- H41qce-HS
- H41qce-HR
- H41qce-HRS
- H51q-M
- H51q-MS
- H51q-H
- H51q-HS
- H51q-HRS
- H51qe-M
- H51qe-MS
- H51qe-H
- H51qe-HS
- H51qe-HR
- H51qe-HRS

11.2 Former Resource Types

11.2.1 H41q

The following H41q systems do not have the possibility to place a communication module F 8621, F 8621A, F 8625, F 8626, F 8627 or F 8628.

- H41q-M
- H41q-MS
- H41q-H
- H41q-HS
- H41q-HR
- H41q-HRS
- H41qe-M
- H41qe-MS
- H41qe-H
- H41qe-HS
- H41qe-HR
- H41qe-HRS

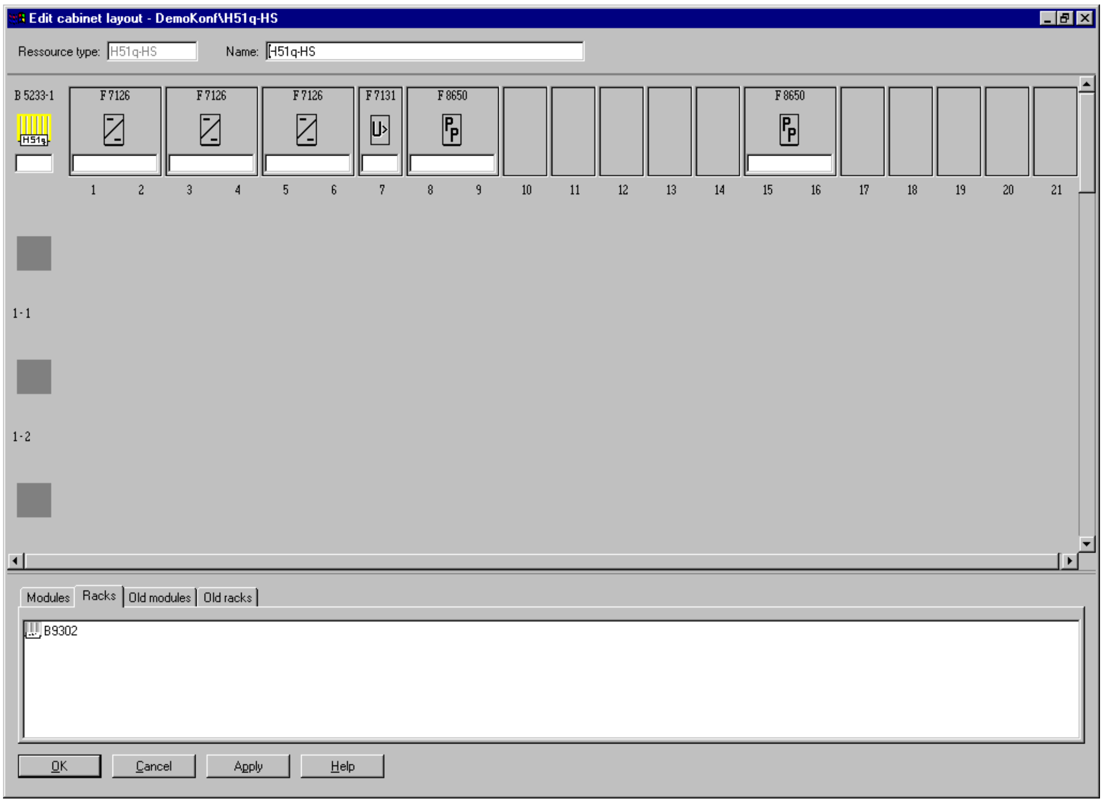
11.2.2 H41 and H51

All H41/H51 systems are listed which require the operating system version V6.0-6.

- H41-M
- H41-MS
- H41-H
- H41-HS
- H41-HR
- H41-HRS
- H51-M
- H51-MS
- H51-H
- H51-HS
- H51-HRS

12 Subracks

When using H51 or H51q systems, you must also define the required I/O subracks. This is done in the *Edit Cabinet* option of the resource.



Edit cabinet layout - DemoKonf\H51q-HS

Ressource type: H51q-HS

Name: H51q-HS

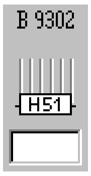
Edit cabinet

In the H11, H41, A1, A1dig and H41q systems there is no need to assign an I/O subrack because here the I/O modules belong to the central module or in the central subrack. No further extension is possible in these systems.

We distinguish between *Subracks* and *Old Subracks*.

12.1 Current subracks

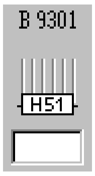
The I/O subrack B 9302 is available for the H51q and H51 systems.
To insert the subrack, Drag & Drop it to the left onto the grey box with the desired position.



You must use this subrack with new systems.

12.2 Old subracks

The B 9301 subrack is still available for H51 systems. This subrack cannot be used in the H51q system.



13 Modules

As with the subracks, we distinguish between *Modules* and *Old Modules*.
Only the current modules in the *Modules* directory are used with new systems, of course.

13.1 Current modules

The current modules include:

- F 3221, 16-fold input module, digital (BOOL)
- F 3222, 8-fold input module, digital (BOOL) for proximity switches
- F 3223, 4-fold input module, digital (BOOL) for proximity switches in intrinsically safe circuits
- F 3224, 4-fold input module, digital (BOOL) for proximity switches in intrinsically safe circuits, with open-circuit monitoring
- F 3236, 16-fold input module, digital (BOOL), safety-related
- F 3237, 8-fold input module, digital (BOOL), safety-related, for proximity switches
- F 3238, 8-fold input module, digital (BOOL), safety-related, for proximity switches in intrinsically safe circuits, with line diagnosis
- F 3240, 16-fold input module, digital (BOOL), safety-related, 110 V DC, 127 V AC
- F 3248, 16-fold input module, digital (BOOL), safety-related, 48 V DC / AC
- F 3322, 16x output module, digital (BOOL), 500 mA (12 W)
- F 3325, 6-fold supply unit (Ex)i for F 6221
- F 3330, 8-fold output module, digital (BOOL), safety-related, 500 mA (12 W)
- F 3331, 8-fold output module, digital (BOOL), safety-related, 500 mA (12 W), with line diagnosis
- F 3332, 4-fold output module, digital (BOOL), 2 A (48 W)
- F 3333, 4-fold output module, digital (BOOL), 2 A (48 W), safety- related
- F 3334, 4-fold output module, digital (BOOL), 2 A (48 W), safety- related, with line diagnosis
- F 3335, 4-fold output module (Ex)i, safety-related
- F 3348, 8-fold output module, safety-related, 48 V
- F 3349, 8-fold output module, safety-related, 24 V / 48 V, 500 mA
- F 3422, 8-fold relay module, switching voltage 60 V =/~
- F 3430, 4-fold relay module, safety-related, 250 V AC, 110 V DC
- F 5220, 2-fold counter module, safety-related up to 1MHz
- F 6208, signal converter (Ex)i analog
- F 6213, 4-fold input module, analog, safety-related
- F 6214, 4-fold input module, analog, safety-related, for transmitter
- F 6215, 8-fold input module, analog
- F 6216, 8-fold input module, analog, for transmitter
- F 6217, 8-fold input module, analog, for transmitter
- F 6220, 8-fold thermocouple input module (Ex)i, safety-related
- F 6221, 8-fold analog input module (Ex)i, safety-related
- F 6705, 2-fold output module, analog, safety-related
- F 6706, 2-fold output module, analog
- F 7126, power supply for H51, H51q, 24V/5V
- F 7130, power supply for H41, H41q, 24V/5V
- F 7131, power supply monitor for H51, H51q
- F 7132, 4-fold distribution board
- F 7133, 4-fold distribution board with fuse monitor
- F 7553, connecting module I/O bus for 9302
- F 8621, communication module, left CU

- F 8621x, communication module, right CU
- F 8625, ethernet module, left CU, 10 Mbit
- F 8625x, ethernet module, right CU, 10 Mbit
- F 8626, Profibus-DP module, left CU
- F 8626x, Profibus-DP module, right CU
- F 8627, ethernet module, left CU, 100 Mbit
- F 8627x, ethernet module, right CU, 100 Mbit
- F 8628, Profibus-DP module, left CU
- F 8628x, Profibus-DP module, right CU

Move the required module to the desired position in the desired subrack with Drag & Drop.

You do not need to define central modules, they are defined by their resource type assignment.

13.2 Old modules

These modules can only be used in H51 systems. They cannot be used in the H51q.
The following modules are available:

- F 3225, 16-fold input module, digital (BOOL), safety-related
- F 3227, 8-fold input module, digital (BOOL)
- F 3228, 16-fold input module, digital (BOOL)
- F 3235, 8-fold input module, digital (BOOL), safety-related, with line diagnosis
- F 3311, 16-fold output module, digital (BOOL), 200 mA
- F 3312, 4-fold output module, digital (BOOL), 1 A
- F 3313, 8-fold output module, digital (BOOL), safety-related, 400 mA
- F 3314, 4-fold output module, digital (BOOL), safety-related, 1 A
- F 3321, 16-fold output module, digital (BOOL), 500 mA
- F 3323, 8-fold output module, digital (BOOL), safety-related, 500 mA, with line diagnosis
- F 3412, 8-fold relay module
- F 3413, 8-fold relay module
- F 5202, 14 bit ring counter
- F 5203, 14 bit ring counter
- F 6103, input module, analog, for PT100 (Ex)i
- F 6204, input module, analog, (Ex)i
- F 6207, input module, analog, (Ex)i, for thermocouples
- F 6701, 2-fold output module, analog
- F 7105, 6-fold distribution board with fuse monitor
- F 7125, 12-fold distribution board with fuse monitor
- F 7129, 4-fold distribution board with fuse monitor
- F 7531, 6-fold distribution board
- F 7541, connecting module for B 9301

None of the old modules has integrated safety shutdown; they can only be used in subrack B 9301.

Use current modules only for new projects.

14 Supported IEC Functions and Data Types

IEC 61131-3 contains data types, standard functions and standard blocks but they are not all supported by each PES.

14.1 Supported data types

The supported data types vary between the different systems. The systems H41/H51, H11 and A1 require operating systems with versions 6.0-6 for operation with ELOP II.

Type	H41/H51, H11, A1	H41q/H51q	Comment
ANY			any data type
ANY_NUM			any number
ANY_REAL			any real
REAL	X	X	Real according to IEC 559
LREAL	-	-	long real according to IEC 559
ANY_INT			any integer
SINT	X	X	short integer (-128...127)
INT	X	X	integer (-32768...32767)
DINT	-	X	double integer (-2147483648 ... 214783647)
LINT	-	-	long integer (-2^{63} ... 2^{63} -1)
USINT	X	X	unsigned short integer (0...255)
UINT	X	X	unsigned integer (0...65535)
UDINT	-	X	unsigned double integer (0...4294967295)
ULINT	-	-	unsigned long integer (0...2^{64} -1)
ANY_BIT			any bit
BOOL	X	X	1 Bit (FALSE, TRUE)
BYTE	X	X	Byte, 8 Bit
WORD	X	X	Word, 16 Bit
DWORD	-	X	Double Word, 32 Bit
LWORD	-	-	Long Word, 64 Bit
STRING	-	-	String
ANY_DATE			Any Date
TIME_OF_DAY	-	-	Time (TOD)
DATE	-	-	Date
DATE_AND_TIME	-	X	Date and Time (DT)
TIME	X	X	Time

Table 11: Supported data types

X: available
-: not available

14.2 Supported functions and function blocks

The supported functions and function blocks vary between the different systems. The systems H41/H51, H11 and A1 require operating systems with versions 6.0-6 for operation with ELOP II.

Function	H41/H51, H11, A1	H41q/H51q
ANY_TO_BOOL	-	X
ANY_TO_BYTE	-	X
ANY_TO_WORD	-	X
ANY_TO_DWORD	-	X[1]
ANY_TO_LWORD	-	-
ANY_TO_STRING	-	-
ANY_TO_SINT	-	X
ANY_TO_INT	X[2]	X
ANY_TO_DINT	-	X[1]
ANY_TO_LINT	-	-
ANY_TO_USINT	-	X
ANY_TO_UINT	X[2]	X
ANY_TO_DINT	-	X[1]
ANY_TO_ULINT	-	-
ANY_TO_REAL	X[2]	X
ANY_TO_LREAL	-	-
ANY_TO_TIME_OF_DAY	-	X[1]
ANY_TO_DATE	-	X[1]
ANY_TO_DATE_AND_TIME	-	X[1]
ANY_TO_TIME	-	X[1]
TRUNC_SI	-	X
TRUNC_I	X	X
TRUNC_DI	-	-
TRUNC_LI	-	-
TRUNC_US	-	X
TRUNC_UI	X[1]	X
TRUNC_UD	-	-
TRUNC_UL	-	-
BCD_TO_INT	-	X
BCD_TO_USINT	-	X
BCD_TO_UINT	-	X
BCD_TO_UDINT	-	X
INT_TO_BCD	-	X

Table 12: supported functions and function blocks

Function	H41/H51, H11, A1	H41q/H51q
USINT_TO_BCD	-	X
UINT_TO_BCD	-	X
UDINT_TO_BCD	-	X
ABS	-	X
MOVE	X	X
ADD	X	X
MUL	X[2)]	X
SUB	X	X
DIV	X[2)]	X
MOD	-	X
EXPT	-	-
SQRT	X	X
LN	X	X
LOG	-	-
EXP	X	X
SIN	-	-
COS	-	-
TAN	-	-
ASIN	-	-
ACOS	-	-
ATAN	-	-
SHL	-	X
SHR	-	X
ROL	-	X
ROR	-	X
AND	X	X
OR	X	X
XOR	X	X
NOT	X	X
SEL	X	X
MAX	-	X[1)]
MIN	-	X[1)]
LIMIT	-	X
MUX	-	X
GT	X[2)]	X
GE	X[2)]	X
EQ	X	X

Table 12: supported functions and function blocks

Function	H41/H51, H11, A1	H41q/H51q
LE	X[2)]	X
LT	X[2)]	X
NE	X	X
LEN	-	-
LEFT	-	-
RIGHT	-	-
MID	-	-
CONCAT	-	-
INSERT	-	-
DELETE	-	-
REPLACE	-	-
FIND	-	-
ADD_TOD_T	-	X
ADD_DT_T	-	X
ADD_T_T	-	X
SUB_TOD_T	-	X
SUB_DT_T	-	X
SUB_T_T	-	X
SUB_TOD_TOD	-	X
SUB_DT_DT	-	X
SUB_T_T	-	X
MUL_T	-	X[1)]
DIV_T	-	X[1)]
CONCAT_D_TOD	-	X
SR	X	X
RS	X	X
SEMA	-	X
R_TRIG	X	X
F_TRIG	X	X
CTU	-	X
CTD	-	X
CTUD	-	X
TP	X	X
TON	X	X
TOF	X	X
RTC	-	X
PACK	-	X

Table 12: supported functions and function blocks

Function	H41/H51, H11, A1	H41q/H51q
UNPACK2	-	X
UNPACK4	-	X
UNPACK8	-	X
FORCEMRK		X

Table 12: supported functions and function blocks

X: available
-: not available
$^{1)}$: except variables of datatype REAL
$^{2)}$: only for REAL, INT, UINT

15 System Variables

The system variables provide information from the system and are used to transfer information to the system. The following system variables are available:

- UINT SIO.ZB1: SIO1 receive counter
- UINT SIO.ZB1: SIO2 receive counter
- UINT SIO.ZG1/CB1: SIO1 receive counter
- UINT SIO.ZG1/CB1: SIO2 receive counter
- UINT SIO.ZG1/CB2: SIO1 receive counter
- UINT SIO.ZG1/CB2: SIO2 receive counter
- UINT SIO.ZG1/CB3: SIO1 receive counter
- UINT SIO.ZG1/CB3: SIO2 receive counter
- UINT SIO.ZB2: SIO1 receive counter
- UINT SIO.ZB2: SIO2 receive counter
- UINT SIO.ZG2/CB1: SIO1 receive counter
- UINT SIO.ZG2/CB1: SIO2 receive counter
- UINT SIO.ZG2/CB2: SIO1 receive counter
- UINT SIO.ZG2/CB2: SIO2 receive counter
- UINT SIO.ZG2/CB3: SIO1 receive counter
- UINT SIO.ZG2/CB3: SIO2 receive counter
- BOOL I/O.error acknowledgment
- UINT I/O.error code 2nd I/O bus
- UINT I/O.error position 2nd I/O bus
- UINT I/O.error code 1st I/O bus
- UINT I/O.error position 1st I/O bus
- BOOL I/O.error
- BOOL SYSTEM.Logic-Emergency stop
- UINT SYSTEM.RAM/EPROM
- UINT SYSTEM.Runversion
- UINT SYSTEM.Codeversion
- BOOL SYSTEM.Force individual switch outputs
- BOOL SYSTEM.Force individual switch inputs
- BOOL SYSTEM.Force main switch outputs
- BOOL SYSTEM.Force main switch inputs
- UINT SYSTEM.Number of prohibited accesses
- BOOL SYSTEM.Prohibited access
- BOOL SYSTEM.Single-channel
- UINT SYSTEM.Errormask2
- UINT SYSTEM.Errormask1
- UINT SYSTEM.Errorcode
- BOOL SYSTEM.normal

The system variables of the type UINT can also be assigned to a variable of the type WORD.

15.1 READ System variables of the BOOL type

READ system variables provide you with information from the operating system.

15.1.1 I/O Error

This system variable is TRUE when the operating system has detected one or more testable I/O modules as faulty. Use this system variable to display an I/O error.

15.1.2 HIBUS.Resource-Name.error

The system variable exists for each resource with safety-related communication via HIPRO. ´Resource-Name´ is replaced by the original resource name.
The variable is TRUE, if the resource does not receive any data inside the entered supervision time.

15.1.3 SYSTEM.Force individual switch outputs

This system variable is TRUE when at least one output variable is forced. An output variable is a variable to which an I&C name is assigned, i.e. it is assigned to an output module.

15.1.4 SYSTEM.Force individual switch inputs

This system variable is TRUE when at least one input variable is forced. An input variable is a variable to which an I&C name is assigned, i.e. it is assigned to an input module.

15.1.5 SYSTEM.Force main switch outputs

This system variable is TRUE when the FORCE main switch for outputs is on.

15.1.6 SYSTEM.Force main switch inputs

This system variable is TRUE when the FORCE main switch for inputs is on.

15.1.7 SYSTEM.Prohibited access

This system variable is TRUE for a cycle when an attempt has been made to run a prohibited function.
It is configured in the properties (safety) of the resource.

15.1.8 SYSTEM.Single-channel

This system variable is TRUE when one central module has failed in a system with two central modules.

15.1.9 SYSTEM.normal

This system variable is TRUE when no errors are present in the system. Use this system variable for a general status display of the system.

15.2 WRITE System variables of the BOOL type

You use WRITE system variables to transfer information to the operating system.

15.2.1 I/O Error acknowledgment

When this system variable is set to TRUE the I/O error is acknowledged. The following functions are performed:

- Acknowledge a displayed I/O error.
  Error display is reset and the system is checked again. If the system finds the error again, the position of the faulty module is displayed.
- Restart the test routines of the testable I/O modules that were shut down.

15.2.2 SYSTEM.Logic emergency stop

When this system variable is TRUE there is a general system shutdown. All outputs are de-energized and the system immediately enters the safe condition.
The system variable can be wired with an external signal or with a signal generated from the logic.
The system is set to RUN again by pressing the ACK button on the central modules.

15.3 READ System variables of the UINT/WORD type

READ system variables provide you with information from the operating system.
You can use either the UINT or WORD data type for the following system variables. Depending on the data type used, the value will subsequently be displayed in decimal or hexadecimal in the OLT field.

15.3.1 HIBUS.Resource-Name.Receive counter

The system variable exists for each resource with safety-related communication via HIPRO. ´Resource-Name´ is replaced by the original resource name.
The counter will be increased with each received transmission. The value has the range of 0...65535. After the value has reached the end value, the value starts again with 0.

15.3.2 SIO Receive Counter

All receive counters work as follows:
The receive counter is increment by 1 for every transmission received at this interface. The value range is from 1 to 65535, or 0001 to FFFF. The counter is reset to 1 when the maximum value is reached.

15.3.2.1 SIO.ZB1: SIO1-receive counter

Receive counter first interface of the left hand central module.

15.3.2.2 SIO.ZB1: SIO2-receive counter

Receive counter second interface of the left hand central module.

15.3.2.3 SIO.ZG1/CB1: SIO1 Receive counter

Receive counter first interface of the first coprocessor module that is assigned to the left hand central module.

15.3.2.4 SIO.ZG1/CB1: SIO2 Receive counter

Receive counter second interface of the first coprocessor module that is assigned to the left hand central module.

15.3.2.5 SIO.ZG1/CB2: SIO1 Receive counter

Receive counter first interface of the second coprocessor module that is assigned to the left hand central module.

15.3.2.6 SIO.ZG1/CB2: SIO2 Receive counter

Receive counter second interface of the second coprocessor module that is assigned to the left hand central module.

15.3.2.7 SIO.ZG1/CB3: SIO1 Receive counter

Receive counter first interface of the third coprocessor module that is assigned to the left hand central module.

15.3.2.8 SIO.ZG1/CB3: SIO2 Receive counter

Receive counter second interface of the third coprocessor module that is assigned to the left hand central module.

15.3.2.9 SIO.ZB2: SIO1-receive counter

Receive counter first interface of the right hand central module.

15.3.2.10SIO.ZB2: SIO2-receive counter

Receive counter second interface of the right hand central module.

15.3.2.11SIO.ZG2/CB1: SIO1 Receive counter

Receive counter first interface of the first coprocessor module that is assigned to the right hand central module.

15.3.2.12SIO.ZG2/CB1: SIO2 Receive counter

Receive counter second interface of the first coprocessor module that is assigned to the right hand central module.

15.3.2.13SIO.ZG2/CB2: SIO1 Receive counter

Receive counter first interface of the second coprocessor module that is assigned to the right hand central module.

15.3.2.14SIO.ZG2/CB2: SIO2 Receive counter

Receive counter second interface of the second coprocessor module that is assigned to the right hand central module.

15.3.2.15SIO.ZG2/CB3: SIO1 Receive counter

Receive counter first interface of the third coprocessor module that is assigned to the right hand central module.

15.3.2.16SIO.ZG2/CB3: SIO2 Receive counter

Receive counter second interface of the third coprocessor module that is assigned to the right hand central module.

15.3.3 I/O.Error code 1st I/O bus

Displays the faulty channels of the module shown in system variable I/O.Error position 1st I/O bus. A display is only possible when the module has line diagnosis.
Value output:

Bit No. 1...8	Dec.	Hex	Error
00 000 000	0	0	No error
00 000 001	1	1	Error in circuit channel 1
00 000 010	2	2	Error in circuit channel 2
00 000 100	4	4	Error in circuit channel 3
00 001 000	8	8	Error in circuit channel 4
00 010 000	16	10	Error in circuit channel 5
00 100 000	32	20	Error in circuit channel 6
01 000 000	64	40	Error in circuit channel 7
10 000 000	128	80	Error in circuit channel 8
11 111 111	255	FF	Faulty module

Table 13: Faulty channel display

If several external circuits are faulty then a corresponding value is displayed.

15.3.4 I/O.Error position 1st I/O bus

This system variable contains the position of a faulty I/O module of the first I/O bus. The value corresponds to the bus number, the subrack and the position of the module. If there is more than one faulty module the module with the lowest position is always displayed.
1405 means: bus 1, subrack 4, position 05.

You should use the UINT data type so the value is displayed in this format.

15.3.5 I/O.Error code 2nd I/O bus

Displays the faulty channels of the module shown in system variable I/O.Error position 2nd I/O bus. A display is only possible when the module has line diagnosis.
Value output:

Bit No. 1...8	Dec.	Hex	Error
00 000 000	0	0	No error
00 000 001	1	1	Error in circuit channel 1
00 000 010	2	2	Error in circuit channel 2
00 000 100	4	4	Error in circuit channel 3
00 001 000	8	8	Error in circuit channel 4
00 010 000	16	10	Error in circuit channel 5
00 100 000	32	20	Error in circuit channel 6
01 000 000	64	40	Error in circuit channel 7
10 000 000	128	80	Error in circuit channel 8
11 111 111	255	FF	Faulty module

Table 14: Faulty channel display

If several external circuits are faulty then a corresponding value is displayed.

15.3.6 I/O.Error position 2nd I/O bus

This system variable contains the position of a faulty I/O module of the second I/O bus. The value corresponds to the bus number, the subrack and the position of the module. If there is more than one faulty module the module with the lowest position is always displayed.
1405 means: bus 1, subrack 4, position 05.

You should use the UINT data type so the value is displayed in this format.

15.3.7 SYSTEM.RAM/EPROM

This system variable displays the type, the memory used for the resource and the memory mapping.

Value		Redundant System			
Hex	Dec	left central module		right central module	
		NLS	SLP	NLS	SLP
0	0	RAM	RAM	RAM	RAM
1	1	EPROM	RAM	RAM	RAM
2	2	EPROM	EPROM	RAM	RAM
100	256	RAM	RAM	EPROM	RAM
101	257	EPROM	RAM	EPROM	RAM
102	258	EPROM	EPROM	EPROM	RAM
200	512	RAM	RAM	EPROM	EPROM
201	513	EPROM	RAM	EPROM	EPROM
202	514	EPROM	EPROM	EPROM	EPROM

Table 15: RAM/EPROM Mapping

15.3.8 SYSTEM.Runversion

This system variable displays the current RUN version of the resource.
You must use the WORD data type for this variable so that the display is identical with the representation on the diagnostic display of the central module.

15.3.9 SYSTEM.Codeversion

This system variable displays the current Code version of the resource.
You must use the WORD data type for this variable so that the display is identical with the representation on the diagnostic display of the central module.

15.3.10 SYSTEM.Number of prohibited accesses

This system variable shows how often an attempt has been made to call a prohibited action or function.

15.3.11 SYSTEM.Errormask1

Error mask 1 displays faults detected in the central modules and in the I/O bus. The set error bits have the following meaning:

Error bit 1...16	Hex	Dec	Fault type
0000 0000 0000 0000	0	0	No fault
0000 0000 0000 0001	1	1	CPU
0000 0000 0000 0010	2	2	CTC (time-IC)
0000 0000 0000 0100	4	4	Hardware watchdog
0000 0000 0000 1000	8	8	Memory fault
0000 0000 0001 0000	10	16	Program crash
0000 0000 0010 0000	20	32	Time-out
0000 0000 0100 0000	40	64	Dev. CTXC/hardware clock
0000 0000 1000 0000	80	128	Hardware clock
0000 0001 0000 0000	100	256	Connection to I/O level
0000 0010 0000 0000	200	512	Power supply monitor
0000 0100 0000 0000	400	1024	Address test I/O BT
0000 1000 0000 0000	800	2048	Time delay other ZB
0001 0000 0000 0000	1000	4098	Outputs not 0 on start-up
0010 0000 0000 0000	2000	8192	Dev. CTC/hardware clock can be tolerated
0100 0000 0000 0000	4000	16384	Not used
1000 0000 0000 0000	8000	32768	Memory unequal

Table 16: Error mask 1

If several faults occur simultaneously, a value is output that shows the fault bits at their corresponding positions. This means that several bits can be set at the same time.
Faults for the coprocessor modules are only output when they are defined in the cubicle.

15.3.12 SYSTEM.Errormask2

Error mask 2 displays general faults in the power supply, faults in coprocessor modules, active fault blanking, and the allocation of faults to the central modules. The set error bits have the following meaning:

Error bit 1...16	Hex	Dec	Fault type
0000 0000 0000 0000	0	0	No fault
0000 0000 0000 0001	1	1	Backup battery F 71xx ZB1
0000 0000 0000 0010	2	2	Backup battery F 71xx ZB2
0000 0000 0000 0100	4	4	Power supply 1
0000 0000 0000 1000	8	8	Power supply 2
0000 0000 0001 0000	10	16	Power supply 3
0000 0000 0010 0000	20	32	Fault blanking active
0000 0000 0100 0000	40	64	Fault ZB1

Table 17: Error mask 2

Error bit 1...16	Hex	Dec	Fault type
0000 0000 1000 0000	80	128	Fault ZB2
0000 0001 0000 0000	100	256	Copr. module 1 ZB 1
0000 0010 0000 0000	200	512	Copr. module 2 ZB 1
0000 0100 0000 0000	400	1024	Copr. module 3 ZB 1
0000 1000 0000 0000	800	2048	Copr. module 1 ZB 2
0001 0000 0000 0000	1000	4098	Copr. module 2 ZB 2
0010 0000 0000 0000	2000	8192	Copr. module 3 ZB 2
0100 0000 0000 0000	4000	16384	Backup battery on ZB 1
1000 0000 0000 0000	8000	32768	Backup battery on ZB 2

Table 17: Error mask 2

If several faults occur simultaneously, a value is output that shows the fault bits at their corresponding positions. This means that several bits can be set at the same time.
Faults for the coprocessor modules are only output when they are defined in the cubicle.

15.3.13 SYSTEM.Errorcode

The display of the error code is used for a detailed analysis of a fault that has occurred. The meaning of the error code can be taken from the description of the appropriate operating system.

Index

Numerics

3964R base addresses 34
3964R communication 51
3964R variables 41

A

Address conflicts 30

B

Building an application 74
BUS docu 65
Bus documentation 65
BUSCOM 49
BUSCOM variables 40
BUSCOM, MODBUS base addresses 33

C

C-code comparator 73
CFG docu 65
CFG documentation 67
Code generator 28, 52
Code generator settings 30
Compiling program code 52
Configuration 1
Control panel 54
Creating a configuration 1
CRF Docu 68
CRF docu 68
CRF documentation 68
CU, CM configure interfaces 14

D

Data types 85
Documentation configuration 65
Documentation resource 68

E

Edit cabinet 13
Editing modules 14
ELOP II bus 1
ELOP II bus assignment 2
ELOP n-file 48
Event creating 36

F

fieldbuses 49

Force image, graphic 64

Forcing I/Os 62

Forcing in force image 63

Forcing variables 61

Function check of the PES 74, 76

H

HIBUS 2

HIBUS parameter 4

HIMA Standard blocks 77

HIPRO 42

HIPRO edit 3

HIPRO Hardware 42

HIPRO redundancy 42

HIPRO stations 4

HIPRO-N 42

HIPRO-N variables 38, 44

HIPRO-N/-S variables 37

HIPRO-S 32, 42

HIPRO-S PES master 43

HIPRO-S variables 39

I

I/O parameters 19

Instancing program type 11

L

Lcl page header 27

Lcl page header editing 27

M

MODBUS 49

Mode configuring serial interface 15

Modification of an application 75

Modules current 83

Modules old 84

O

OLT panel 58

Online Test 58

Online test 60

OPC 49

OS download 57

P

PADT bus 24

PES master 7

PES master compiling 7, 52

PES master CRF 9
PES master docu 65
PES master documentation 66
PES master download 7, 53
PES master functions 37
PES master update 8
Program download/reload 57
proven GNU-C-compiler 73

R
Redundancy parameters 21
RES docu (current) 68
RES docu (generated) 68
RES docu (loaded) 68
Res documentation 70
Res documentation generated 71
Res documentation loaded 71
Resource 10
Resource hardware change 10
Resource properties 12, 19
Resource type assignment 10
Resource types current 79
Resource types predecessor 80

S
Safety parameters 22
Saftey related functions 73
Special interface settings 15
Subracks 81
Supported functions and function blocks 86
System variables 90
System variables assignment 35

T
Tag name editing 15, 17
target code comparator 73

HIMA
... the safe decision.

HIMA Paul Hildebrandt GmbH
Industrial Automation
P.O. Box 1261 • D-68777 Bruehl • Germany
Telephone: (+49 6202) 709-0 • Telefax: (+49 6202) 709-107
E-mail: info@hima.com • Internet: www.hima.com

(0611)