

Sistemas baseados em PC

ELOP II Factory

a partir da versão 4.1 build 6111 da gestão de projetos
e da versão 7.56.10 do Hardware Management

Primeiros passos



HIMA Paul Hildebrandt GmbH
Automação industrial

HI 800 529 CPA

Todos os produtos HIMA mencionados neste manual estão protegidos pela marca registrada da HIMA. A não ser que seja mencionado de outra forma, isso também se aplica a outros fabricantes mencionados e seus produtos.

É proibido repassar ou reproduzir este documento, comercializar ou compartilhar o seu conteúdo, exceto onde explicitamente autorizado. Infrações nos dão direito a indenização.

Todos os dados e avisos técnicos neste manual foram elaborados com o máximo de cuidado, considerando medidas efetivas de controle de garantia de qualidade. Mesmo assim, erros não podem ser totalmente excluídos.

Por isso, a HIMA avisa que não pode assumir garantia ou aceitar responsabilidade legal ou direitos de indenização pelas consequências oriundas de informações incorretas. A HIMA sempre agradece a comunicação de eventuais erros.

Os dados técnicos estão sujeitos a alterações sem notificação prévia.

Informações mais detalhadas encontram-se na documentação no CD-ROM e na nossa homepage em <http://www.hima.com/>

Favor encaminhar solicitações de informações para:

HIMA Paul Hildebrandt GmbH
Postfach 12 61
D-68777 Brühl

Tel: +49 (6202) 709 0
Fax: +49 (6202) 709 107

Email: info@hima.com

Índice

1	VISÃO GERAL RESUMIDA	1
1.1	VOLUME DE FORNECIMENTO	1
1.2	INFORMAÇÕES SOBRE ESTE MANUAL	1
1.3	SUORTE.....	2
2	INSTALAÇÃO	3
2.1	O QUÊ É NECESSÁRIO?.....	4
2.2	ONDE SERÁ INSTALADO?	4
2.3	COMO SE INICIA A INSTALAÇÃO?.....	5
2.3.1	Assim inicia-se a instalação	5
2.4	DESINSTALAÇÃO	7
2.4.1	Desinstalação do Hardware Management.....	8
2.4.2	Desinstalação da gestão de projetos	9
2.4.3	Desinstalação do controlador do Hardlock	9
3	INTRODUÇÃO AO ELOP II FACTORY.....	11
3.1	INICIAR O ELOP II FACTORY	11
3.2	DIVISÃO DA TELA DA GESTÃO DE PROJETOS	12
3.2.1	Barra de título.....	13
3.2.2	Barra de menu.....	13
3.2.3	Barra de símbolos	14
3.2.4	Statusleiste.....	14
3.2.5	Janela de estrutura.....	15
3.2.6	Menu de contexto para objetos	16
3.2.7	Área de trabalho	17
3.2.8	Editor da linguagem de blocos funcionais (FBS Editor)	17
3.2.9	Editor de documentos	18
3.2.10	Indicador de status e erros	19
3.2.11	Ajuda online.....	19

3.3	DIVISÃO DA TELA DO HARDWARE MANAGEMENT.....	20
3.3.1	Janela de estrutura.....	22
3.3.2	Menu de contexto	23
3.3.3	Editor de sinais	24
3.3.4	Ajuda online.....	25
4	OS OBJETOS NA JANELA DE ESTRUTURA.....	27
4.1	PROJETO – PROJECT.....	27
4.2	BIBLIOTECA – LIBRARY	27
4.2.1	Tipo de programa – Program Type.....	27
4.2.2	Tipo de bloco funcional – Function Block Type	28
4.2.3	Função – Function	28
4.3	CONFIGURAÇÃO – CONFIGURATION.....	29
4.3.1	Recurso – Resource	29
4.3.2	Instância de programa, instância de tipo – Program Instance, Type Instance	30
4.4	DOCUMENTAÇÃO – DOCUMENTATION	30
5	FUNÇÕES BÁSICAS DO ELOP II FACTORY	31
5.1	GESTÃO DE PROJETOS	31
5.2	BIBLIOTECAS (DE BLOCOS).....	32
5.3	EDITOR DA LINGUAGEM DE BLOCOS FUNCIONAIS	33
5.3.1	Maximizar a área de trabalho	34
5.3.2	Maximizar e restaurar áreas de novo	35
5.4	ESQUEMAS DE FUNÇÕES NA ABORDAGEM CENTRADA NO MEIO	36
5.4.1	Zoom	37
5.5	AJUSTAR AS PROPRIEDADES DA ÁREA DE DESENHO.....	38
5.6	CRIAR A LÓGICA	39
5.6.1	Drag & Drop de variáveis.....	39
5.6.2	Drag & Drop de blocos funcionais	39
5.6.3	Conectar elementos.....	39

5.7	ELABORAR A GERENCIAR A DOCUMENTAÇÃO.....	40
5.7.1	Modelos para imprimir documentos	41
5.8	SIMULAÇÃO OFFLINE DE ESQUEMAS DE FUNÇÕES.....	43
5.9	ONLINE TEST (POWER FLOW).....	44
6	CONCEITOS PARA TIPOS DE RECURSO	45
6.1	CRIAÇÃO DE UM RECURSO	46
6.2	ATRIBUIÇÃO DE UM TIPO DE PROGRAMA A UM RECURSO	48
6.3	ATRIBUIR UM TIPO DE RECURSO	49
6.4	ATRIBUIÇÃO DE MÓDULOS DE E/S	50
6.4.1	Atribuir módulos	50
6.4.2	Atribuir sinais aos canais de E/S	51
6.4.3	Atribuir sinais de sistema.....	51
6.5	GERADOR DE CÓDIGO	51
6.6	CONTROL PANEL.....	52
6.7	DOCUMENTAÇÃO DO HARDWARE.....	52
7	PROJETODEEXERCÍCIOS.....	53
7.1	CRIAR UM PROJETO.....	53
7.2	CRIAR UMA BIBLIOTECA E ELABORAR PROGRAM ORGANISATION UNITS (POU).....	55
7.2.1	Criar uma biblioteca	55
7.2.2	Criar unidades de organização do programa “Program Organisation Units” (POU).....	57
7.3	EDITAR BLOCOS FUNCIONAIS	59
7.3.1	Declaração de variáveis	59
7.3.2	Definir a declaração de interfaces (representação gráfica)	62
7.3.3	Introdução de lógica na área de desenho do bloco	66
7.3.4	Terminar a edição do bloco	71
7.4	EDITAR O TIPO DE PROGRAMA OU UMA INSTÂNCIA DE TIPO.....	72
7.4.1	Declaração de variáveis	72
7.4.2	Definição da declaração de interfaces	72
7.4.3	Introdução de lógica na área de desenho	72

7.5	CRIAÇÃO DE UM RECURSO:	73
7.6	SIMULAÇÃO OFFLINE	77
7.6.1	Simulação offline de um programa	77
7.6.2	Simulação offline de um bloco definido pelo usuário	82
7.7	ATRIBUIÇÃO DE HARDWARE DO SISTEMA DE COMANDO	84
7.7.1	Atribuir um tipo de recurso.....	84
7.7.2	Definir os ajustes de iniciar do programa.....	87
7.7.3	Ajustar as propriedades do módulo de comunicação	88
7.7.4	Adicionar a um recurso modular módulos de entrada/saída.....	91
7.7.5	Criar uma Remote I/O (RIO).....	93
7.8	SINAIS	97
7.8.1	Definição da diferença entre sinal e variável	97
7.8.2	Definir sinais	98
7.8.3	Atribuir sinais aos canais de E/S do hardware	103
7.8.4	Atribuir sinais aos sinais de sistema	106
7.9	COMUNICAÇÃO COM OUTROS SISTEMAS DE COMANDO HIMATRIX	107
7.9.1	Comunicação Peer-to-Peer (comunicação P2P)	107
7.10	GERAÇÃO DE CÓDIGO	115
7.11	PARAMETRIZAR O PC E OS SISTEMAS DE COMANDO	120
7.11.1	Parametrizar o aparelho de programação (PC) para a comunicação	120
7.11.2	Parametrizar o sistema de comando para a comunicação	122
7.11.3	Para um sistema de comando com projeto desconhecido	125
7.11.4	Ativar os ajustes de fábrica.....	131
7.12	CARREGAR E INICIAR PROGRAMA (CONFIGURAÇÃO DE RECURSO)	135
7.13	O FORCE EDITOR	139
7.13.1	Salvar a seleção de sinais e carregar.....	145
7.13.2	Forçar um sistema já forçado	148
7.14	ONLINE TEST (POWER FLOW).....	150

7.15	DOCUMENTAÇÃO.....	152
7.15.1	Documentação do software.....	152
7.15.2	Documentação do hardware	159
7.16	ARQUIVAR	162
7.17	RESTAURAR.....	165
8	ANEXO.....	169
8.1	GLOSSÁRIO	169
8.2	ÍNDICE REMISSIVO	177
8.3	LISTA DE ABREVIACÕES	183

1 Visão geral resumida

1.1 Volume de fornecimento

Pertencem ao ELOP II Factory:

- Este manual
O manual “Primeiros passos” lhe permite uma familiarização rápida e fácil com a operação do ELOP II Factory. Além disso, oferece, ao lado de uma visão geral sobre as funções, instruções passo-a-passo para a elaboração de um projeto.
- Um CD-ROM
Além do software ELOP II Factory, o CD-ROM contém alguns programas auxiliares e a documentação completa para os sistemas atuais HIMatrix.
- Um Hardlock (Dongle)
O Hardlock serve à gestão de licenças do software protegido ELOP II Factory (proteção contra uso não autorizado).

1.2 Informações sobre este manual

Neste manual, o usuário encontrará informações importantes para se familiarizar com as mais importantes funções do ELOP II Factory no contexto de um treinamento ou para trabalho autodidata.

O Capítulo 2 explica a instalação do ELOP II Factory. Os Capítulos 3 a 6 descrevem de forma geral como manusear e operar o ELOP II Factory. Usuários sem conhecimentos prévios deveriam ler estas seções cuidadosamente.

O Capítulo 7 contém lições com exercícios com ajuda dos quais os usuários que já dispõem de conhecimentos básicos de ELOP II Factory podem se familiarizar com a elaboração de projetos ou aprofundar os seus conhecimentos.

No Anexo, no Capítulo 8, há explicações sobre os termos técnicos utilizados, o índice remissivo e o índice de abreviações.

1.3 Suporte

No caso de dúvidas sobre a operação ou para comunicar erros de programa e sugestões de melhora há as seguintes opções à sua disposição.

Dúvidas frequentes	Capítulo neste documento	Perguntas e respostas relacionadas a temas básicos
News, FAQs, Download	Nossa homepage www.hima.com	Novidades, dúvidas frequentes, blocos funcionais
Perguntas e sugestões	Por E-Mail: Support@hima.com Telefone: +49-(0)6202-709 185 Fax: +49-(0)6202-709 199	Entre 9:00 h e 17:00 h

Nota: Este manual é parte dos documentos de trabalho dos seminários ELOP II Factory na HIMA. Devido ao caráter poderoso do ELOP II Factory, aqui apenas as funções mais importantes do programa podem ser mostradas.

Para aprofundar os conhecimentos, recomenda-se participar de um seminário.

2 Instalação

Neste Capítulo:

- O quê é necessário?
- Onde será instalado?
- Como se inicia a instalação?
- Desinstalação

O ELOP II Factory é um programa protegido por um dispositivo de hardware. O Hardlock (Dongle) deve ser conectado à interface paralela ou à interface USB.



Fig. 1: Hardlock para a porta da impressora e porta USB

Para poder acessar o Hardlock, um controlador deve ser instalado no computador. Para instalar o controlador no Windows 2000/XP®, um usuário com direitos de administrador deve estar logado. Em casos de dúvida, deve ser consultado o administrador do sistema.

2.1 O quê é necessário?

Além do PC, são necessários o Hardlock e o CD-ROM de instalação.

Os requisitos de hardware do computador são os seguintes:

	Mínimo	Recomendado
Processador	Intel Pentium III® 800 MHz	Intel Pentium IV® 3 GHz
RAM	256 MB	1024 MB
Placa de vídeo	8MB XGA 1280x1024 TRUE-Colour	128 MB XGA 1280x1024 TRUE-Colour
Disco rígido	Mín. 500MB para ELOP II Factory mais o espaço para os programas de aplicação	
Sistema operacional	Windows 2000® Professional a partir de Service Pack 1 Windows XP® até Service Pack 2	

Tabela 1: Requisitos de hardware do PC

Se uma impressora estiver ligada ao Hardlock, a mesma deve estar ligada, pois algumas impressoras possuem resistência de ligação insuficiente no estado desligado.

2.2 Onde será instalado?

A instalação é efetuada no disco rígido local.

O ELOP II Factory Versão 4.1 / 7.56.10 apenas precisa ser instalado uma vez por um usuário qualquer e está à disposição em alemão e inglês a todos os usuários, se os mesmos possuíam ao menos o status de usuário principal.

2.3 Como se inicia a instalação?

A instalação do ELOP II Factory e de todos os componentes adicionais ocorre a partir do menu de instalação do CD-ROM.

Além disso, o CD-ROM contém a documentação completa do software e da família de sistemas HIMatrix em arquivos PDF. Também está incluído o programa de visualização necessário, Adobe Acrobat Reader®.

Nota: Certifique-se de que ao instalar o controlador do Hardlock esteja logado no PC com direitos de administrador.

2.3.1 Assim inicia-se a instalação

1. Inserir o CD-ROM na unidade de CD. O menu do CD inicia-se automaticamente. Se isso não ocorrer, abrir o diretório root do CD-ROM a partir do Windows Explorer e clicar duas vezes no arquivo "setup.bat".
2. Selecionar o idioma desejado para a superfície de menus.
3. Selecionar o software de Hardlock a ser instalado.

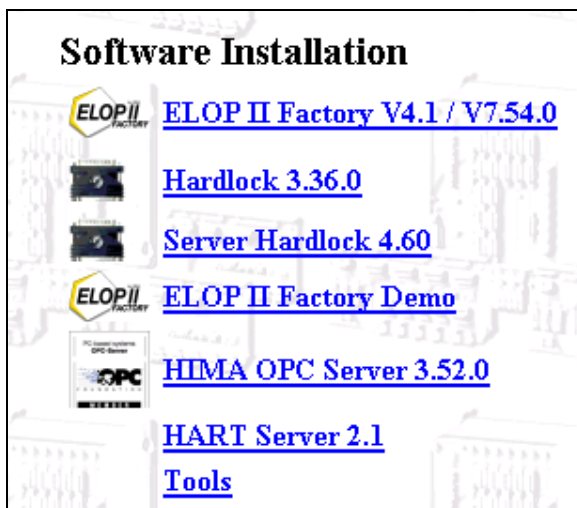


Fig. 2: Instalação do controlador do Hardlock

4. Seguir as instruções de instalação.
5. Depois da instalação do controlador do Hardlock, clicar em **ELOP II Factory V4.1 / V7.56.10** para a instalação do ELOP II Factory. ELOP II Factory não é instalado de forma específica para os usuários, mas está à disposição para todos os usuários.
6. Selecionar se durante a instalação deve ser criada uma entrada no menu Iniciar do Windows e se gostaria de criar um ícone na área de trabalho. A entrada no menu Iniciar deveria ser sempre selecionada. O ícone pode ser criado depois ainda.

Depois da instalação com êxito, o ELOP II Factory pode ser aberto imediatamente. O Control Center e a gestão de projetos podem ser iniciados a partir do menu Iniciar do Windows, na pasta ELOP II Factory (veja também Capítulo 7).

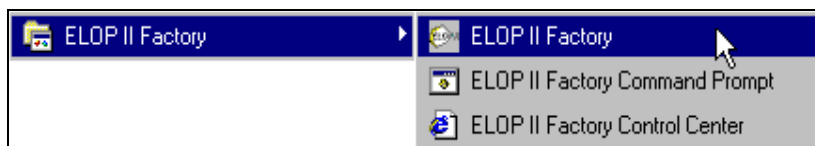


Fig. 3: Menu Iniciar para o ELOP II Factory

2.4 Desinstalação

A desinstalação completa deve ser efetuada em três passos para a versão 4.1 do ELOP II Factory:

1. Desinstalação do Hardware Management.
2. Desinstalação da gestão de projetos.
3. Desinstalação do controlador do Hardlock (se não planejar um Update do ELOP II Factory).

Nota: Se um update para o ELOP II Factory deve ser executado, o Hardware Management e a gestão de projetos devem ser desinstalados antes. A desinstalação do controlador do Hardlock não é necessária.

Reiniciar o computador antes de executar o update.

Se quiser desinstalar a versão 4.1 do ELOP II Factory, a desinstalação do Hardware Management e da gestão de projetos pode ser efetuada num único passo. Para este fim, proceder como descrito no seguinte Capítulo.

2.4.1 Desinstalação do Hardware Management

- ❑ Abrir no Windows o menu **Iniciar, Configurações, Painel de controle, Adicionar ou remover programas.**

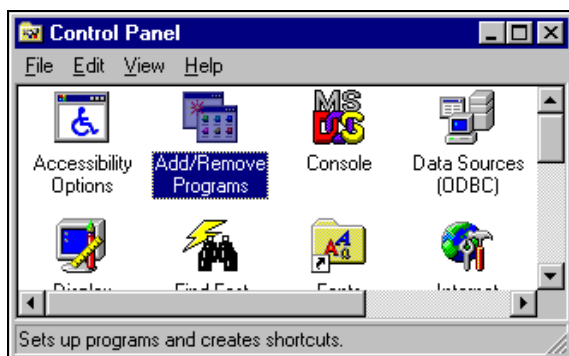


Fig. 4: Desinstalar software

- ❑ Clicar na janela “Adicionar ou remover programas” em “ELOP II Factory Hardware Management” e depois no botão **Alterar/Remover**.

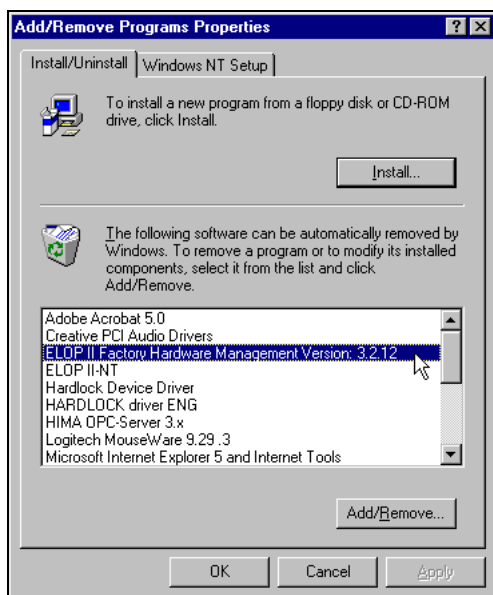


Fig. 5: Hardware Management

2.4.2 Desinstalação da gestão de projetos

- ☐ Abrir o Control Center do ELOP II Factory. Selecionar a função **Administration, Uninstall**.

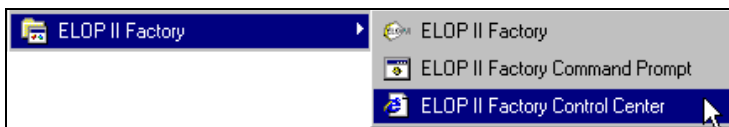


Fig. 6: Chamar o Control Center

- ☐ Executar a função **Uninstall**.

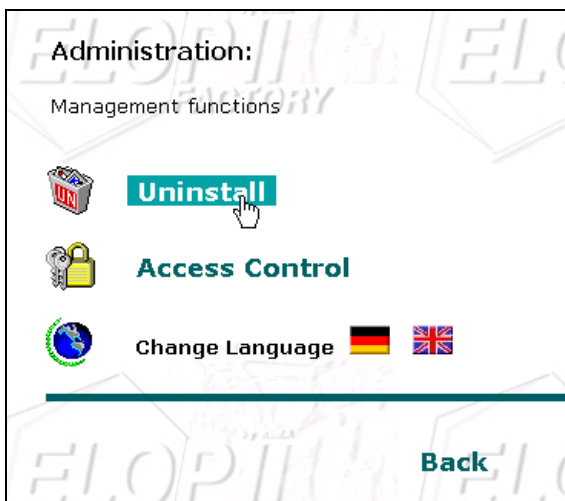


Fig. 7: Desinstalar a gestão de projetos

2.4.3 Desinstalação do controlador do Hardlock

- ☐ Abrir no Windows o menu **Iniciar, Configurações, Painel de controle, Adicionar ou remover programas**.
- ☐ Clicar na janela “Adicionar ou remover programas” em “Hardlock Device Driver” e depois no botão **Alterar/Remover**.

3 Introdução ao ELOP II Factory

Neste Capítulo:

- Iniciar o programa
- Elementos da superfície do ELOP II Factory
- Barra de título e de menus
- Barra de símbolos e barra de status
- Divisão das janelas, janelas de estrutura e de trabalho
- Indicador de status e erros

O ELOP II Factory é um programa de múltiplas funções. O acesso é facilitado pela interface de usuário intuitiva no estilo Windows.

3.1 Iniciar o ELOP II Factory

Selecionar no menu Iniciar do Windows o registro “Programas” e lá, “ELOP II Factory”.

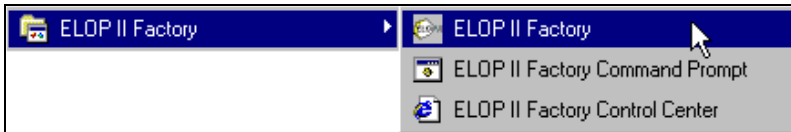



Fig. 8: Menu Iniciar do ELOP II Factory

Alternativamente, também é possível iniciar a gestão de projetos no ELOP II Factory Control Center ou a partir do ícone , na área de trabalho.

O ELOP II Factory consiste em duas janelas:

- 1. Project Management** Gestão de projetos, serve para a elaboração de todos os programas de aplicação e para arquivar ou restaurar projetos.
- 2. Hardware Management** Gestão do hardware, serve para definir todos os dados específicos do hardware. O Hardware Management apenas é aberto ao criar ou abrir um projeto.

3.2 Divisão da tela da gestão de projetos

Depois de iniciar o ELOP II Factory, aparece a tela padrão, conforme Fig. 9. A tela padrão consiste basicamente nos seguintes elementos:

1. Barra de título
2. Janela de estrutura
3. Barra de menus
4. Barra de símbolos para a gestão de projetos
5. Área de trabalho
6. Barra de símbolos para o editor da linguagem de blocos funcionais (FBS Editor)
7. Indicador de status e erros
8. Barra de status com informação de coordenadas do editor do esquema de funções

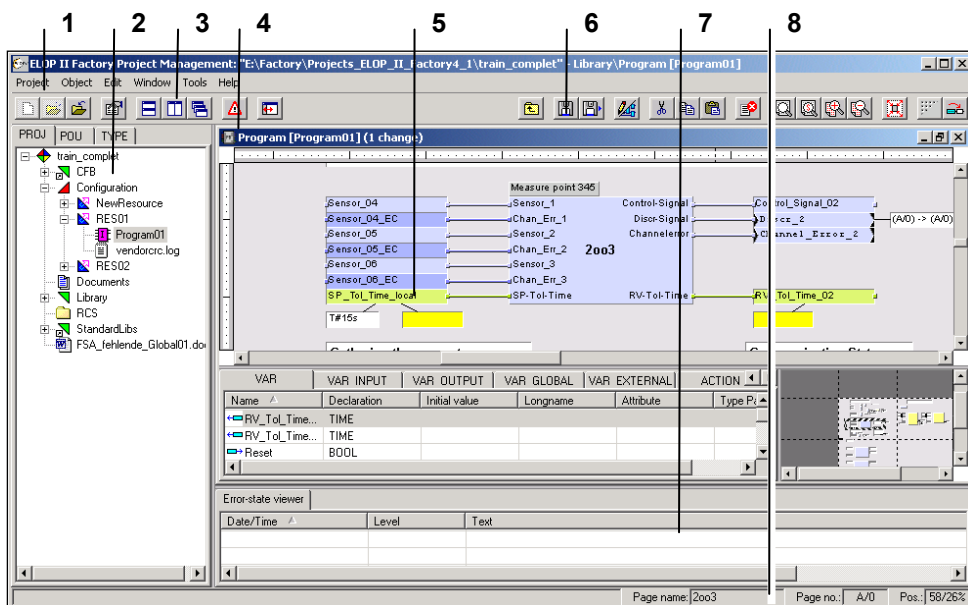


Fig. 9: Tela padrão ELOP II Factory

3.2.1 Barra de título

Além das funções padrão para minimizar, maximizar e fechar a janela, a barra de título também contém informações sobre o projeto e o bloco de função editado.

1. Programa
2. Projeto aberto
3. Bloco aberto
4. Minimizar, Maximizar, Fechar

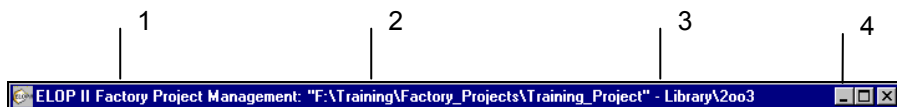


Fig. 10: Barra de título

3.2.2 Barra de menu

O ELOP II Factory é operado pela barra de menu. A maioria das funções do ELOP II Factory são oferecidas na barra de menu.



Fig. 11: O menu “Project”

3.2.3 Barra de símbolos

A barra de símbolos encontra-se abaixo da barra de menus e está dividida nas duas partes “Project Management” e “Function Block Diagram Editor”.



Fig. 12: Barra de símbolos para o projeto



Fig. 13: Barra de símbolos para o editor da linguagem de blocos funcionais

Nota: Ao posicionar o cursor brevemente sobre um botão, aparece um texto “Quick Info” (breve texto de ajuda).

3.2.4 Statusleiste

Na barra de status, na margem inferior da janela, aparecem textos de informação e de ajuda da gestão de projetos e do editor da linguagem de blocos funcionais, bem como a posição atual do cursor.



Fig. 14: Barra de status

3.2.5 Janela de estrutura

Na janela de estrutura, o projeto é representado na sua estrutura hierárquica. É possível escolher entre três visualizações com grau variado de detalhamento.

Projeto total

Todas as UOPs¹

Todos os tipos de dados

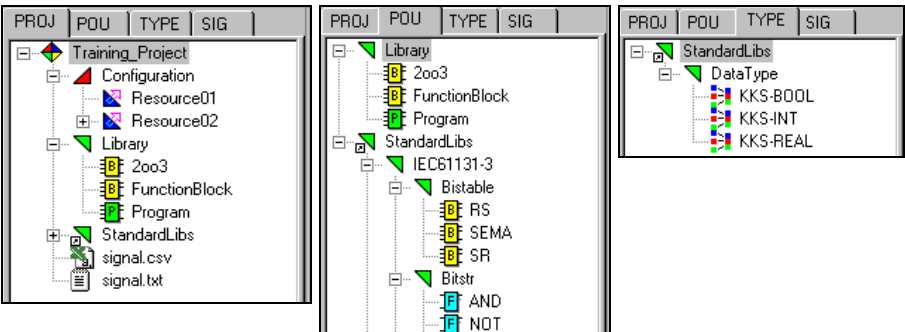


Fig. 15: Janela de estrutura

¹ Program Organisation Unit

3.2.6 Menu de contexto para objetos

Com um clique direito do mouse sobre um objeto da janela de estrutura abre-se o menu de contexto correspondente do objeto. Os comandos individuais são seleccionados como usualmente, mediante um clique esquerdo do mouse.

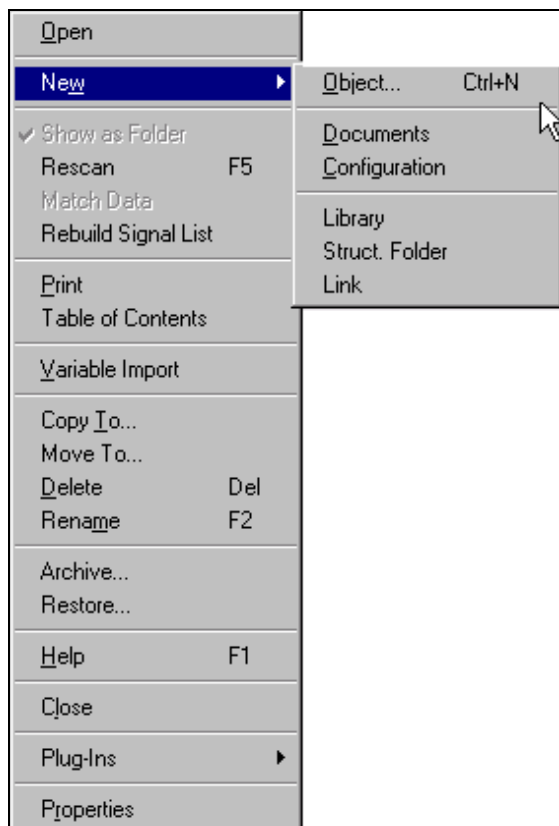


Fig. 16: Menu de contexto do projeto

3.2.7 Área de trabalho

Na área de trabalho (veja Fig. 9), editar os objetos de dados com o

- editor da linguagem de blocos funcionais ou com o
- Editor de documentos.

3.2.8 Editor da linguagem de blocos funcionais (FBS Editor)

Com o editor da linguagem de blocos funcionais são criados os esquemas de funções na linguagem de blocos funcionais “Function Block Diagram Language (FBD)” ou linguagem de sequências “Sequential Function Chart Language (SFC)”.

O editor da linguagem de blocos funcionais (Fig. 17) contém as áreas

1. Área de desenho
2. Editor de declaração de variáveis
3. Janela de visão geral
4. Editor de declaração de interfaces

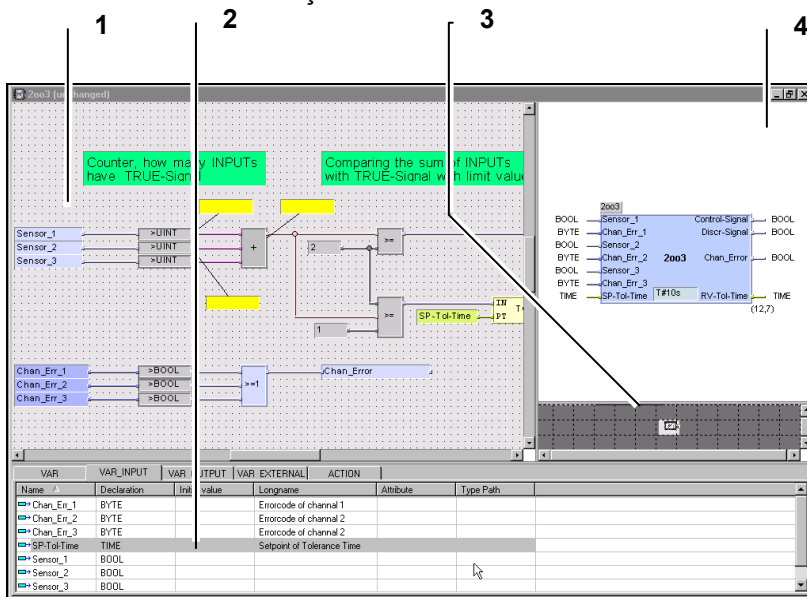
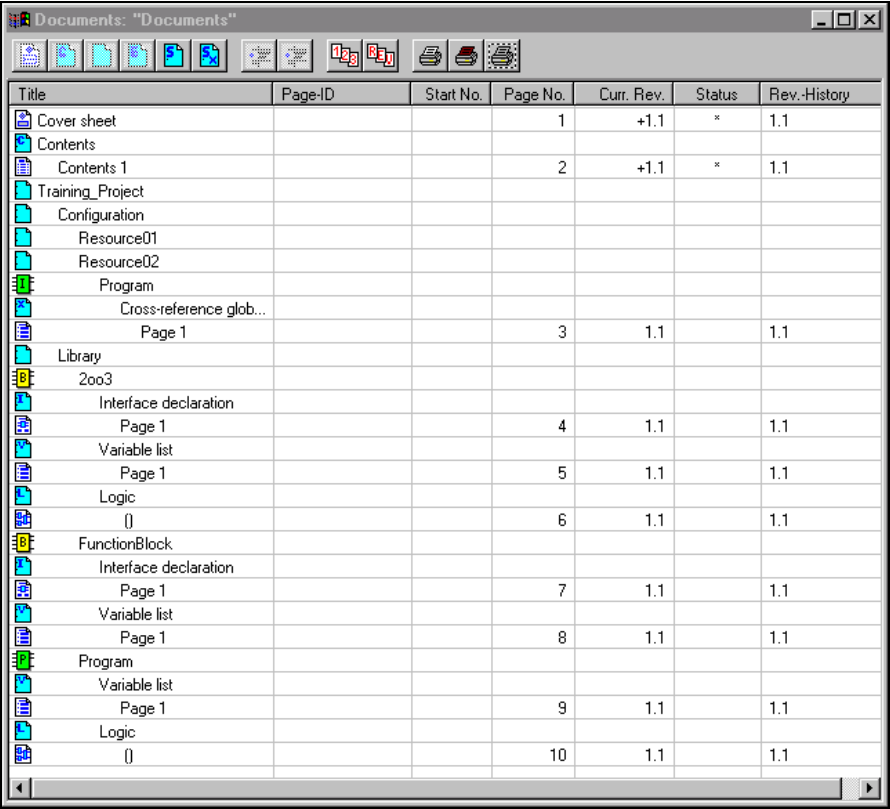


Fig. 17: Editor da linguagem de blocos funcionais

3.2.9 Editor de documentos


Com o editor de documentos, a documentação é elaborada a partir dos objetos na janela de estrutura. Pode ser usada uma gestão de revisões conjunta para todos os documentos. Para uma descrição detalhada, veja Capítulo 7.15.

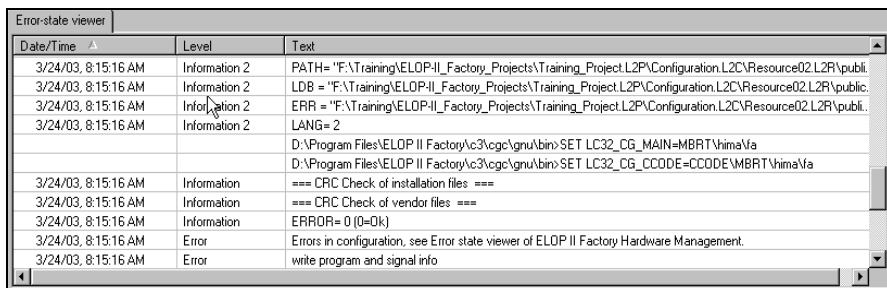


Title	Page-ID	Start No.	Page No.	Curr. Rev.	Status	Rev.-History
Cover sheet			1	+1.1	*	1.1
Contents						
Contents 1			2	+1.1	*	1.1
Training_Project						
Configuration						
Resource01						
Resource02						
Program						
Cross-reference glob...						
Page 1			3	1.1		1.1
Library						
Zoo3						
Interface declaration						
Page 1			4	1.1		1.1
Variable list						
Page 1			5	1.1		1.1
Logic						
()			6	1.1		1.1
FunctionBlock						
Interface declaration						
Page 1			7	1.1		1.1
Variable list						
Page 1			8	1.1		1.1
Program						
Variable list						
Page 1			9	1.1		1.1
Logic						
()			10	1.1		1.1

Fig. 18: Editor de documentos

3.2.10 Indicador de status e erros

No indicador de status e erros “Error-State-Viewer”, as mensagens de erro e status são emitidas de forma central. A incidência de uma nova mensagem é assinalada por um símbolo  que pisca na barra de tarefas do Windows.



Date/Time	Level	Text
3/24/03, 8:15:16 AM	Information 2	PATH= "F:\Training\ELOP-II_Factory_Projects\Training_Project.L2P\Configuration.L2C\Resource02.L2R\publi...
3/24/03, 8:15:16 AM	Information 2	LDB = "F:\Training\ELOP-II_Factory_Projects\Training_Project.L2P\Configuration.L2C\Resource02.L2R\publi...
3/24/03, 8:15:16 AM	Information 2	ERR = "F:\Training\ELOP-II_Factory_Projects\Training_Project.L2P\Configuration.L2C\Resource02.L2R\publi...
3/24/03, 8:15:16 AM	Information 2	LANG= 2
		D:\Program Files\ELOP II Factory\c3\cgc\gnu\bin\SET LC32_CG_MAIN=MBRT\hima\va
		D:\Program Files\ELOP II Factory\c3\cgc\gnu\bin\SET LC32_CG_CCODE=CCODE\MBRT\hima\va
3/24/03, 8:15:16 AM	Information	=== CRC Check of installation files ===
3/24/03, 8:15:16 AM	Information	=== CRC Check of vendor files ===
3/24/03, 8:15:16 AM	Information	ERROR= 0 (0=Ok)
3/24/03, 8:15:16 AM	Error	Errors in configuration, see Error state viewer of ELOP II Factory Hardware Management.
3/24/03, 8:15:16 AM	Error	write program and signal info

Fig. 19: Indicador de status e erros

3.2.11 Ajuda online

Na ajuda online, encontram-se explicações detalhadas sobre todas as funções no ELOP II Factory. Com ajuda do índice, é possível receber ajuda sobre palavras-chave rapidamente.

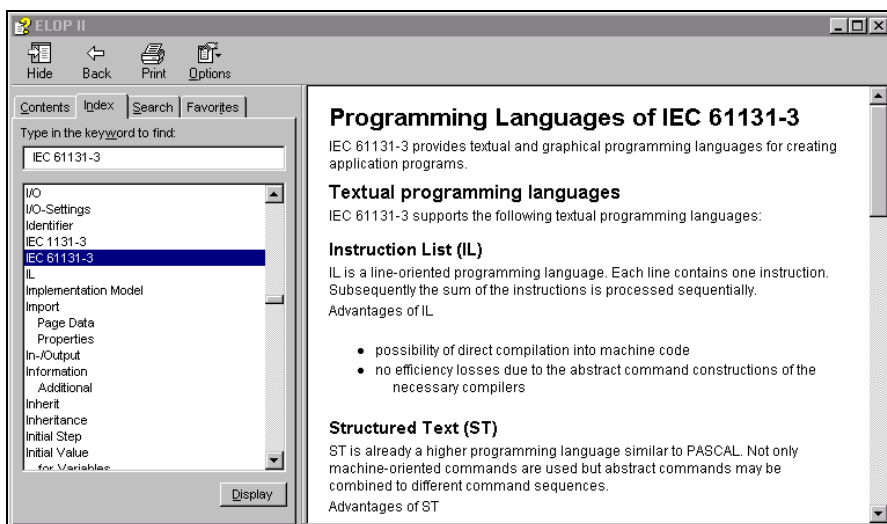


Fig. 20: Ajuda online

3.3 Divisão da tela do Hardware Management

1. Barra de títulos
2. Barra de menu
3. Árvore do projeto
4. Área de trabalho
5. Indicador de status e erros

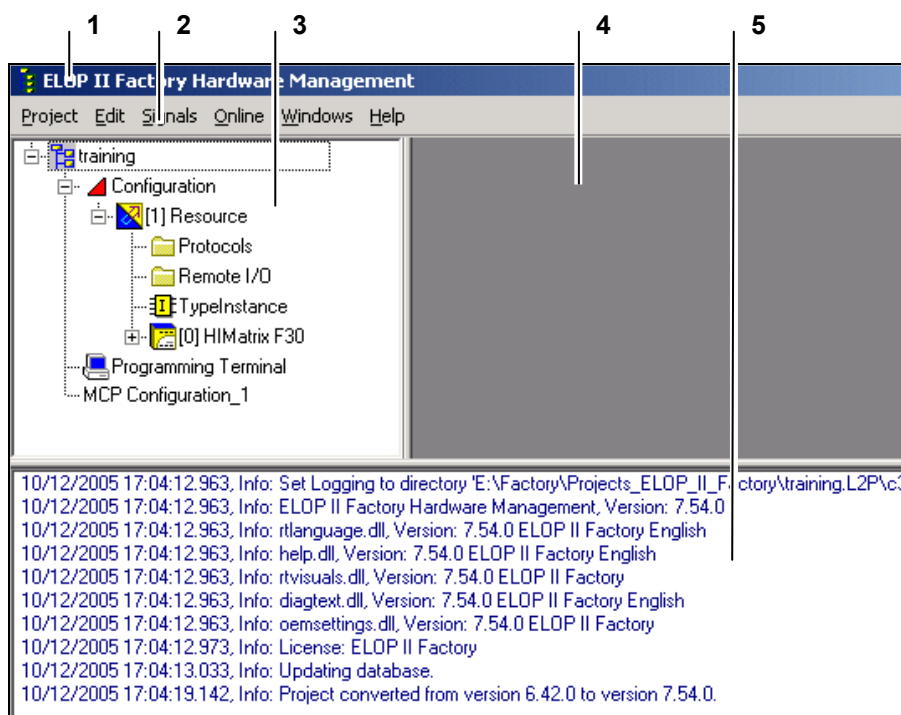


Fig. 21: Tela padrão no Hardware Management

O Hardware Management é operado pela barra de menus:

- ☐ Clicar no menu para abrir o mesmo.
- ☐ Selecionar a opção que deve ser executada. Com um clique do botão esquerdo do mouse, o comando é executado.

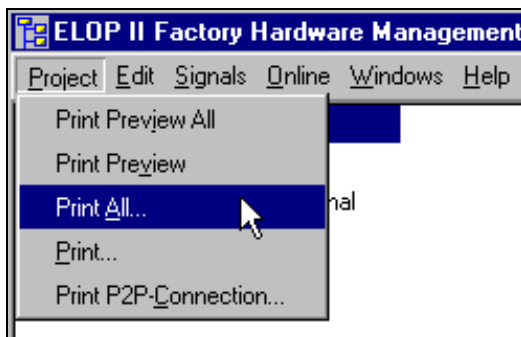


Fig. 22: O menu “Project”

3.3.1 Janela de estrutura

- | | |
|---|------------------------------------|
| 1 | Nome do projeto |
| 2 | Configuração |
| 3 | Lista de recursos |
| 4 | Instância de programa |
| 5 | Lista de protocolos de comunicação |
| 6 | Lista de Remote I/O |
| 7 | Remote I/O |
| 8 | Componentes e módulos |
| 9 | Tipo de recurso |

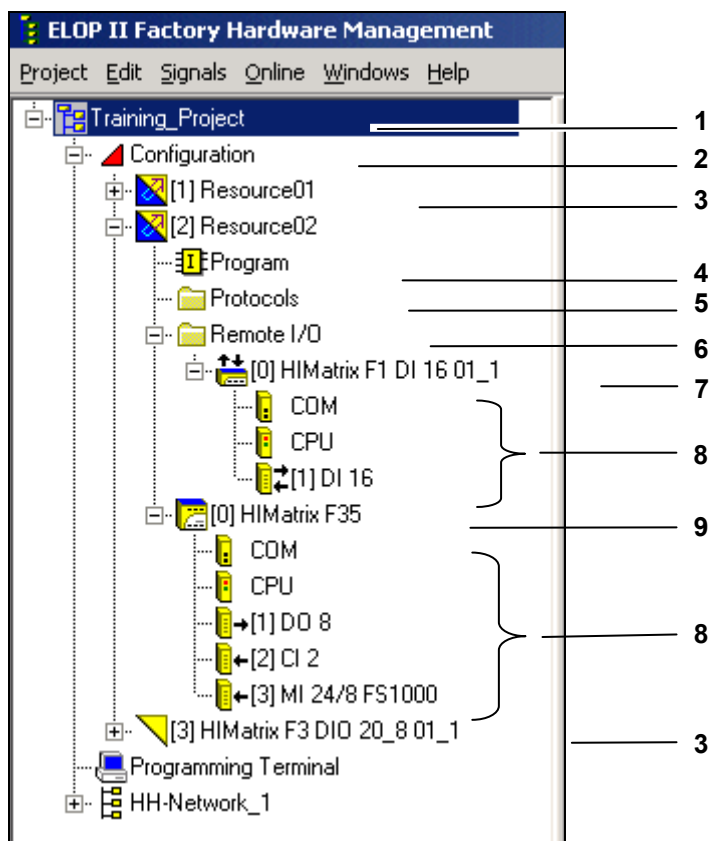


Fig. 23: Janela de estrutura

3.3.2 Menu de contexto

Com um clique direito do mouse sobre um objeto na janela de estrutura abre-se o menu de contexto correspondente do objeto. Os comandos individuais são selecionados como usualmente, mediante um clique esquerdo do mouse.

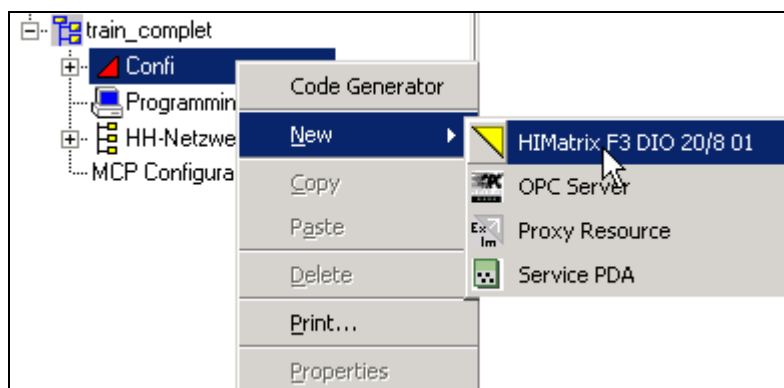


Fig. 24: Menu de contexto

3.3.3 Editor de sinais

O editor de sinais “Signal Editor” é selecionado pelo menu **Signals**.

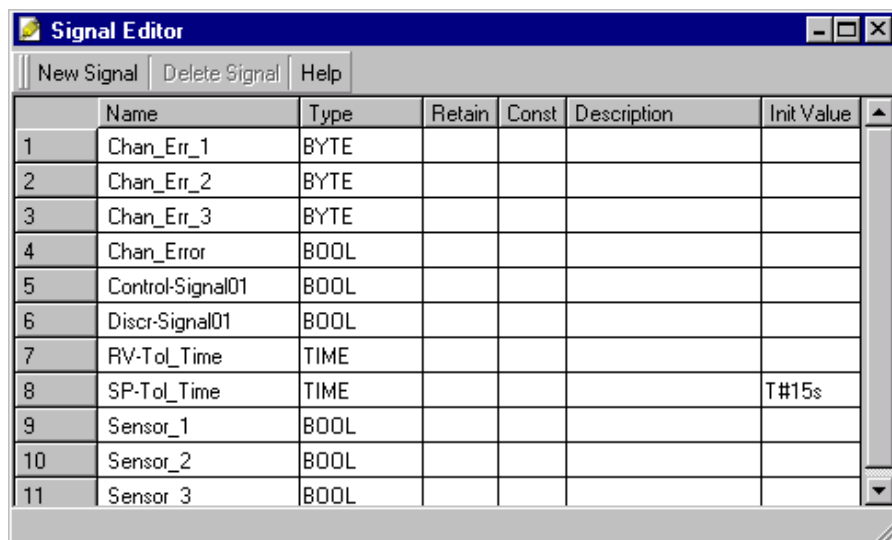


Fig. 25: Signal Editor

Todas as variáveis que devem ser transferidas de uma área de validade (p.ex., programa) para uma outra área de validade (p.ex., nível de E/S), devem receber uma definição de atribuição. Isso ocorre com ajuda de sinais no editor de sinais.

Depois de definir o sinal, o sinal é copiado mediante Drag & Drop para as respectivas áreas e, assim, é criada uma entrada de utilização ou uma referência cruzada no editor de sinais. Mediante um clique com o botão direito do mouse sobre um sinal as referências cruzadas “Cross References” podem ser exibidas.

Cross Reference			
Signal	Access	Usage	Position
Analog01	W	/Confi/[23] RES01/[0] HIMatrix F35/[3] MI 24/8 FS1000	AI[01].Value
	R	/Confi/RES01/Prog01/Type	Pg.-Name 'analog' Pg.-No. A/1

Fig. 26: Referências cruzadas do sinal Analog01

3.3.4 Ajuda online

Através da função de menu **Help**, **Contents**, recebem-se informações sobre todos os temas do Hardware Management.

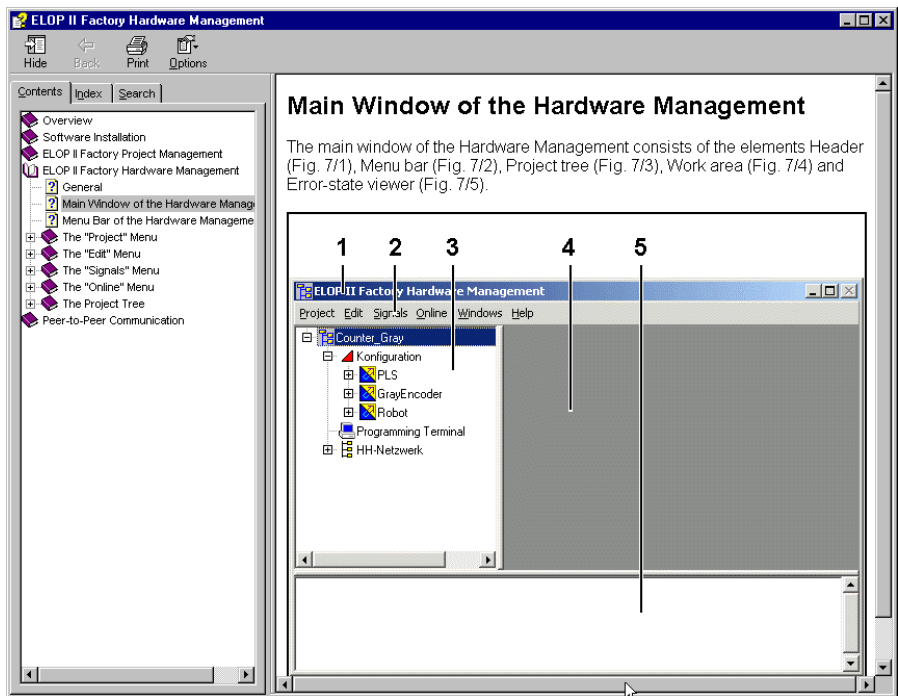


Fig. 27: Ajuda online


4 Os objetos na janela de estrutura

Neste Capítulo:


- Estrutura hierárquica dos objetos na janela de estrutura
- Significado dos objetos

Na janela de estrutura (veja também Fig. 15), todos os objetos do projeto são representados e gerenciados na sua estrutura hierárquica.


4.1 Projeto – Project

O projeto  é o objeto de nível superior. Todos os demais objetos são criados abaixo de um projeto. Sempre apenas um projeto pode estar aberto no ELOP II Factory.


4.2 Biblioteca – Library

A biblioteca  contém funções, blocos funcionais e programas. Na terminologia de acordo com IEC 61131-3, também são chamados de Unidades de Organização de Programa “Program Organisation Units” (abreviado UOP - POU). No ELOP II Factory há bibliotecas padrão com POUs prontas. Com ajuda dessas POUs são criadas suas próprias funções, blocos funcionais e programas, mais complexos.

4.2.1 Tipo de programa – Program Type


O tipo de programa  contém todas as funções de uma aplicação. Um tipo de programa pode ser atribuído a vários sistemas de comando para a execução. O respectivo sistema de comando, então, é uma instância de programa do tipo de programa declarado na biblioteca.

4.2.2 Tipo de bloco funcional – Function Block Type


O tipo de bloco funcional  contém as funções parciais de uma aplicação, comparável a um subprograma. O tipo de bloco funcional também pode ser usado para estruturar o programa correspondente à estrutura da instalação. O tipo de bloco funcional pode armazenar temporariamente valores em variáveis locais. O valor de saída depende dos valores de entrada e dos valores armazenados temporariamente (exemplo típico: Flip-Flop, Timer).

Além disso, com o tipo de bloco funcional é possível acessar variáveis externas.


4.2.3 Função – Function

A função  contém funções básicas de uma aplicação. Ao contrário do tipo de bloco funcional, uma função não consegue memorizar estados. O valor de saída depende exclusivamente dos valores de entrada (exemplo típico: AND, OR).



4.3 Configuração – Configuration


A configuração  agrupa sistemas de comando em unidades lógicas entre as quais pode haver uma conexão de comunicação.

4.3.1 Recurso – Resource


O recurso  é o termo definido na IEC 61131-3 para um sistema alvo que executa a tarefa de comando, ou seja, o sistema de comando HIMatrix. Um recurso é criado dentro de uma configuração, na gestão de projetos.


O símbolo acima mostrado de um recurso corresponde à representação na gestão de projetos. No Hardware Management, podem ser encontrados dois objetos diferentes dentro da configuração:

1.  A contrapartida ao recurso criado na gestão de projetos.
2.  Identificação de uma Remote I/O que pode se comunicar com vários recursos. A Remote I/O é criada no Hardware Management e não pode conter nenhuma parte lógica.

Remote I/Os que apenas podem se comunicar com um recurso (um recurso chamado de nível superior) são representadas pelo símbolo . Na árvore de projetos estão abaixo do recurso de nível superior na hierarquia, no diretório “Remote I/O”. Veja também Fig. 23.


4.3.2 Instância de programa, instância de tipo – Program Instance, Type Instance

A instância de programa  é uma referência a um tipo de programa já existente numa biblioteca. O programa é executado neste recurso.

Uma instância de tipo tem um símbolo parecido  que a instância de programa, porém, não há referência a um tipo de programa na biblioteca. A instância de tipo tem o seu tipo de programa totalmente exclusivo no seu próprio diretório.

A instância no Hardware Management por definição tem um símbolo amarelo.

4.4 Documentação – Documentation

A documentação  permite a compilação dos objetos a serem documentados mediante simples Drag & Drop. É possível aplicar uma gestão de revisões para todos os documentos contidos.

Nesse caso, trata-se da documentação da lógica (do programa de aplicação).

*Nota: A documentação do hardware é exibida e impressa no Hardware Management e pode ser selecionada como função no menu **Project**.*

5 Funções básicas do ELOP II Factory

Neste Capítulo:

- Gestão de projetos
- Criação de funções
- Gestão de documentos do software

Este capítulo explica os conceitos básicos do ELOP II Factory e serve à sua melhor compreensão. Os exemplos da prática encontram-se no Capítulo 7.

5.1 Gestão de projetos

A gestão de projetos é a central organizacional para o trabalho com o ELOP II Factory. Na janela de estrutura é exibido em primeiro lugar a pasta do projeto (veja também a este respeito Fig. 15, parte esquerda).

É possível

- abrir ou fechar um projeto (mediante a barra de menus ou os botões da barra de símbolos),
- arquivar um projeto (pelo menu de contexto) ou
- restaurar um projeto (pelo menu “Project”).

Com ajuda do projeto modelo que também pode ser aberto pela barra de menus, é possível efetuar várias definições prévias para a elaboração.

Um projeto aberto é mostrado na janela de estrutura.

5.2 Bibliotecas (de blocos)

Um projeto pode consistir em um número livre de bibliotecas e, assim, de blocos.

As bibliotecas podem ser criadas dentro de um projeto, uma configuração, um recurso ou dentro de uma outra biblioteca.

Assim, é possível obter uma determinada estruturação das bibliotecas de acordo com a instalação.

As funções básicas da IEC 61131-3 estão contidas na biblioteca “StandardLibs” que é automaticamente integrada ao criar um novo projeto.

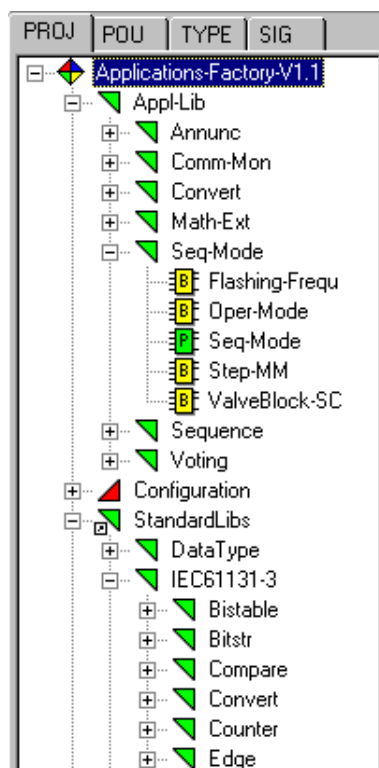


Fig. 28: Bibliotecas na janela de estrutura

5.3 Editor da linguagem de blocos funcionais

O editor da linguagem de blocos funcionais é iniciado ao abrir uma unidade de organização de programa (função, bloco funcional ou programa).

As áreas

- Área de desenho
- Editor de declaração de interfaces
- Editor de declaração de variáveis
- Janela de visão geral

são representadas como no último projeto salvo, numa janela abaixo da área de trabalho da gestão de projetos.

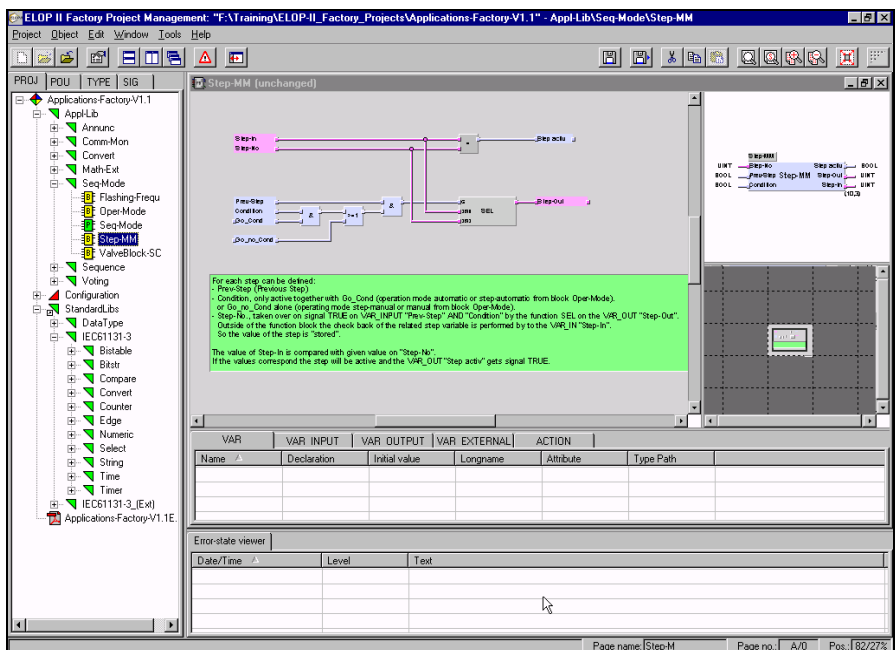




Fig. 29: Editor da linguagem de blocos funcionais na área de trabalho

Para alternar entre a área de trabalho máxima e a visão geral, foram previstas funções especiais.

5.3.1 Maximizar a área de trabalho

É possível exibir e ocultar a visualização da estrutura da Fig. 29 mediante o botão  da parte esquerda da barra de símbolos.

O mesmo vale para o indicador de status e erros e o botão . Com o mesmo é possível aumentar a área de trabalho para o editor da linguagem de blocos funcionais.

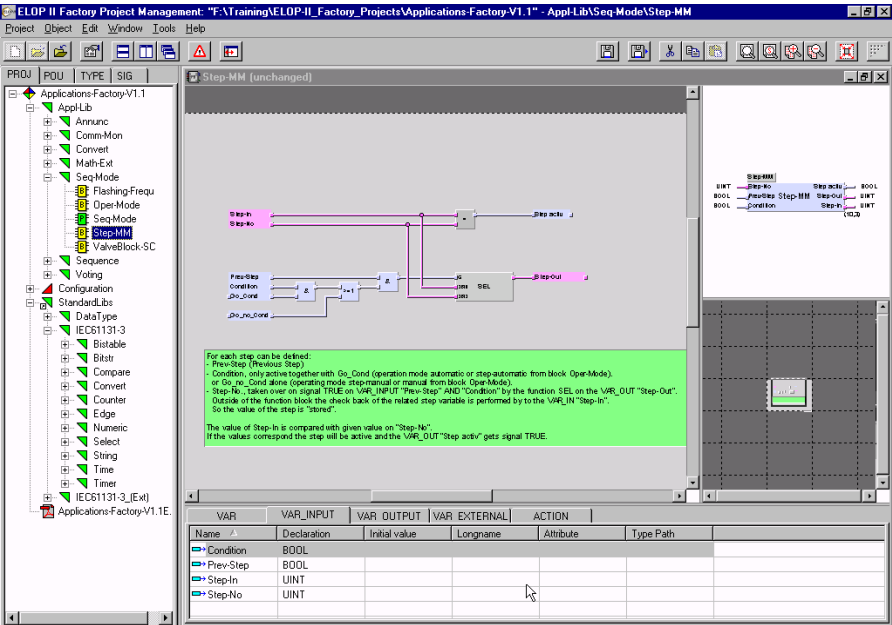


Fig. 30: FBD editor na área de trabalho maximizada

Nota: O tamanho da área de trabalho poder ser adaptada adicionalmente puxando a margem em direção ao editor de declaração de variáveis e ou editor de declaração de interfaces.

5.3.2 Maximizar e restaurar áreas de novo

Ao clicar numa das janelas na área de trabalho (Fig. 30), essa janela é ativada.

Com ajuda do botão , é possível maximizar a janela ativa.

Mediante novo clique no botão, as áreas do editor da linguagem de blocos funcionais são novamente mostradas no seu tamanho original.

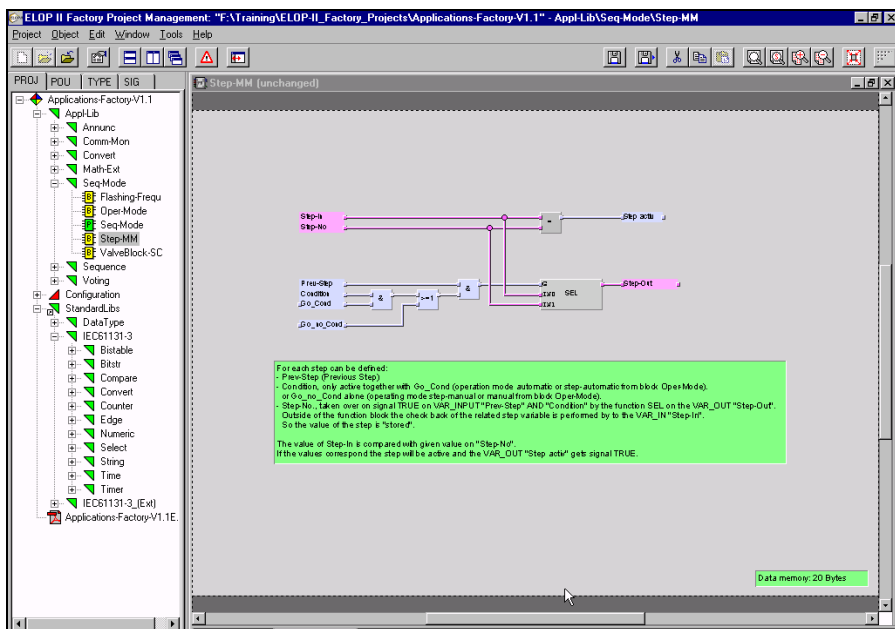


Fig. 31: Área de desenho “Drawing field” maximizada

5.4 Esquemas de funções na abordagem centrada no meio

O conceito básico do ELOP II Factory dispensa a inserção posterior de páginas individuais, criando um esquema de tamanho livre.

A posição de uma página é indicada mediante coordenadas. As colunas são denominadas por letras maiúsculas e as linhas, por algarismos.

A primeira página, por padrão, é a página com as coordenadas A/0. Logo que se inserir um elemento nesta página, a mesma se torna ativa.

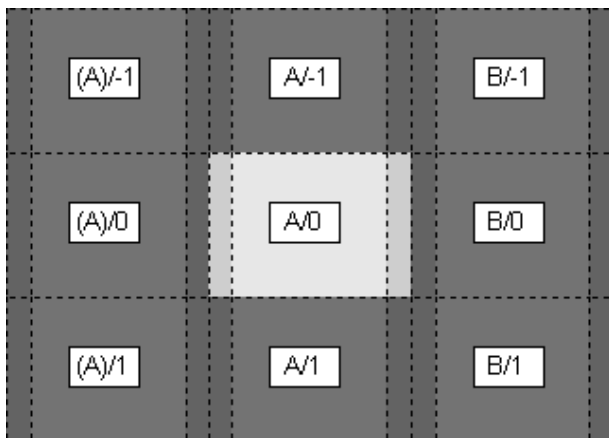


Fig. 32: Numeração de folhas no esquema de funções

A página ativa é representada em tons claros. No momento em que posicionar um elemento numa página adjacente, esta página também se torna ativa e, assim, será exibida em tons claros. Desta forma, o esquema de funções pode ser expandido em qualquer direção.

Nota: Se precisar inserir uma página entre páginas existentes num esquema já criado, é possível deslocar uma página. Para este fim, selecionar no menu de contexto da tela **Plug-Ins, Move page**.

Essa função apenas deve ser utilizada durante o desenvolvimento de um projeto, porém, não com um programa de aplicação que já está em operação. Pois, como as páginas que estão “atrapalhando” são deslocadas, as prioridades de processamento podem ser alteradas.

A janela de visão geral mostra a visão de cima do esquema de funções. Assim, é possível navegar entre as páginas clicando numa das páginas na visão geral.

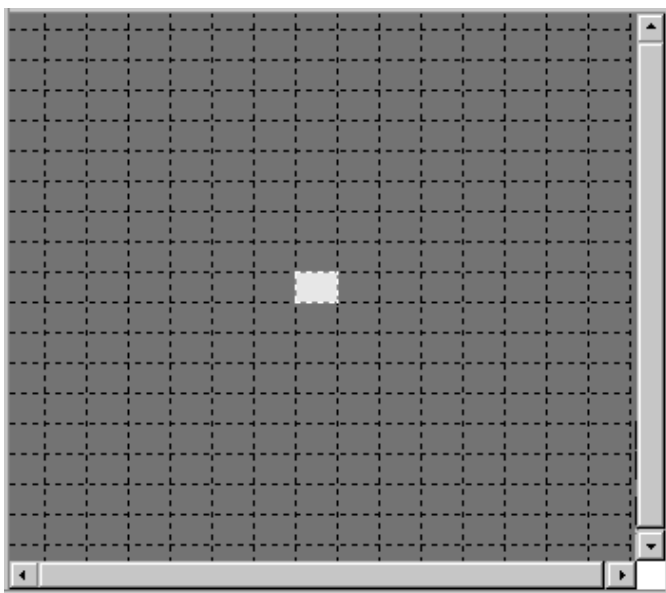



Fig. 33: Janela de visão geral

Nota: No menu de contexto da tela no menu “Plug-ins” encontra-se a função “Page list”. Ela também pode ser usada para acessar determinadas páginas.

5.4.1 Zoom

Com os botões  na barra de símbolos, é possível aumentar ou diminuir a representação na área de desenho e na janela de visão geral.

5.5 Ajustar as propriedades da área de desenho

As seguintes propriedades da área de desenho podem ser ajustadas:

- Definição de barras de campos de valor (recomendação: ajustar a largura do campo de valor a zero).
- Tamanho dos campos de valor, campos de comentário e conectores.
- Ativar ou desativar comentários anexados automaticamente (AC).

As propriedades podem ser chamadas pelo menu de contexto da área de desenho. Para este fim, abrir a mesma com um breve clique direito do mouse.

5.6 Criar a lógica

Com essas funções básicas, cria-se a lógica no esquema de funções.

5.6.1 Drag & Drop de variáveis

Variáveis são criadas no editor de declaração de variáveis. Aqui diferencia-se entre

- Variáveis (locais)
- Variáveis de entrada
- Variáveis de saída e
- Variáveis externas ou no nível de programa:
- Variáveis globais

Para utilizar, clicar no editor de declaração de variáveis sobre uma variável e puxar a mesma para a posição desejada na área de desenho. É criado um campo de valor com um nome de variável. Um símbolo na frente da variável no editor de declaração de variáveis indica o tipo de utilização na área de desenho (veja a este respeito também o Capítulo 7.3.1).

Nota: Também é possível importar variáveis de uma fonte de dados externa. Leia a este respeito a ajuda online.

5.6.2 Drag & Drop de blocos funcionais

Para utilizar um bloco, selecionar o mesmo na janela de estrutura de uma biblioteca e puxar o mesmo para a posição desejada na área de desenho.

5.6.3 Conectar elementos

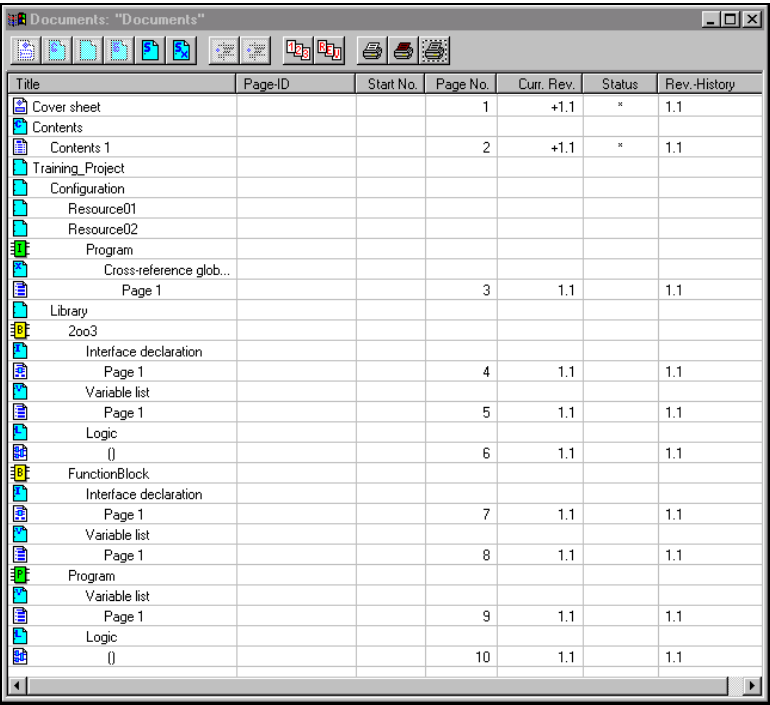
Os elementos colocados na área de desenho podem ser ligados por linhas entre as suas conexões. Aqui deve ser observado que há conexões para entradas e saídas que em certos casos não podem ser ligadas de forma livre entre si.

5.7 Elaborar a gerenciar a documentação

O ELOP II Factory oferece na gestão de projetos para os esquemas de software uma gestão de documentos com serviço de revisões.



Assim, há a possibilidade de criar diferentes revisões de documentos.

A gestão de documentos, além da impressão completa com serviço de revisão conjunto, também detecta as alterações de páginas individuais em seus documentos; assim, também é possível imprimir apenas as folhas alteradas (veja também Capítulo 7.15).



Title	Page-ID	Start No.	Page No.	Curr. Rev.	Status	Rev.-History
Cover sheet			1	+1.1	*	1.1
Contents						
Contents 1			2	+1.1	*	1.1
Training_Project						
Configuration						
Resource01						
Resource02						
Program						
Cross-reference glob...						
Page 1			3	1.1		1.1
Library						
Zoo3						
Interface declaration						
Page 1			4	1.1		1.1
Variable list						
Page 1			5	1.1		1.1
Logic						
()			6	1.1		1.1
FunctionBlock						
Interface declaration						
Page 1			7	1.1		1.1
Variable list						
Page 1			8	1.1		1.1
Program						
Variable list						
Page 1			9	1.1		1.1
Logic						
()			10	1.1		1.1

Fig. 34: Gestão de documentos

Nota: Para atualizar alterações na lógica também na documentação, depois de abrir, clicar no botão “Update table of contents”  e depois “Create revision” .

5.7.1 Modelos para imprimir documentos

Ao imprimir documentos, formulários DXF servem como modelos de impressão. No ELOP II Factory, há um conjunto de formulários padrão para a impressão de todos os objetos.

Cover sheet		D1 6020...	
ELOP II project documentation			
Customer:	D2 customer 1 long D3 customer 2 long	status / revision L2 = Delivery State *	date D14 D16 D17 D18 D19 D20 D21 D22 D23 D24 D25 D26 D27 D28 D29 D30 D31 D32 D33 D34 D35 D36 D37 D38 D39 D40 D41 D42
Order No.:	D4 customer no. cust. 1 long D5 Order no. customer 2 D6 Project title 1 long D7 Project title 2 long D8 Project title 3 long		
Project title:	D9 Enduser 1 D10 Enduser 2 D11 Enduser 3		
Enduser:	D12 Project engineer		
Project-engineer:			

Fig. 35: Capa com os “números de chaves” introduzidos

Nas propriedades do objeto de documentação é possível alterar as entradas padrão para os modelos de impressão e também campos individuais nos formulários DXF.

Properties: Documents

Master Data | Print-Forms | More | Var.-List | Contents | Signal-XRef | Signallist

Name	Key	Value	Value of Parer
Revision a (8 C)	28	1.1	1.1
Date revision a (8 C)	29	27.04.04	27.04.04
Name revision a (8 C)	30	Lämmer	Lämmer
Revision sign b (1 C)	31	2	2
Revision b (8 C)	32	1.2	1.2
Date revision b (8 C)	33	12.03.05	12.03.05
Name revision b (8 C)	34	Lämmer	Lämmer
Revision sign c (1 C)	35	3	3
Revision c (8 C)	36	1.3	1.3
Date revision c (8 C)	37	24.06.05	24.06.05
Name revision c (8 C)	38	Lämmer/ML	Lämmer/ML
Cover	COVER	*** Entries for the cover sheet ***	*** Entries for th
Job number (10 C)	D1	6020...	6020...
Enduser 2 (33 C)	D10	Enduser 2	Enduser 2
Enduser 3 (33 C)	D11	Enduser 3	Enduser 3
Project engineer (33 C)	D12	Project engineer	Project engineer
Status/Revision 1 (21 C)	D13	LZ = Delivery State 1.1	LZ = Delivery S
Revision date 1 (8 C)	D14	27.04.04	27.04.04
Revision name 1 (29 C)	D15	Lämmer/ML	Lämmer/ML

☒ Include Parent Data

Fig. 36: Definições de impressão de um projeto

Nota: A impressão da documentação de hardware deve ser efetuada no Hardware Management (veja também Capítulo 7.15.2).

5.8 Simulação offline de esquemas de funções

Com a simulação offline, é possível verificar a correção lógica dos esquemas de funções criados no PC, sem precisar usar para este fim um sistema de comando programável “programmable electronic system (PES = PLC)”. Para isso, os esquemas de funções são compilados pelo ELOP II Factory e processados pelo PC.

A simulação offline apenas pode ser executada em instâncias de programa ou instâncias de tipo dentro de um recurso.

A simulação offline representa o esquema de funções de forma animada. Com ajuda de campos de teste online (campos OLT), é possível mostrar valores individualmente em qualquer local no esquema de funções. Em caso de valores booleanos, as linhas são representadas em cores adicionalmente.

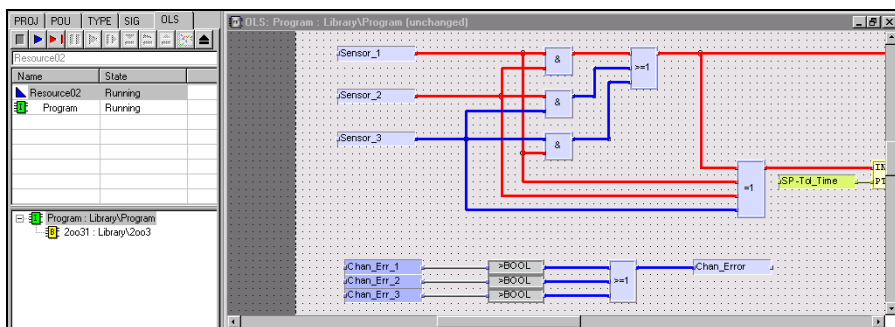


Fig. 37: Simulação offline de um esquema de funções

5.9 Online Test (Power Flow)

O teste online na gestão de projetos serve para rastrear durante a operação do sistema de comando todos os valores das variáveis e dos sinais dentro da representação da lógica.

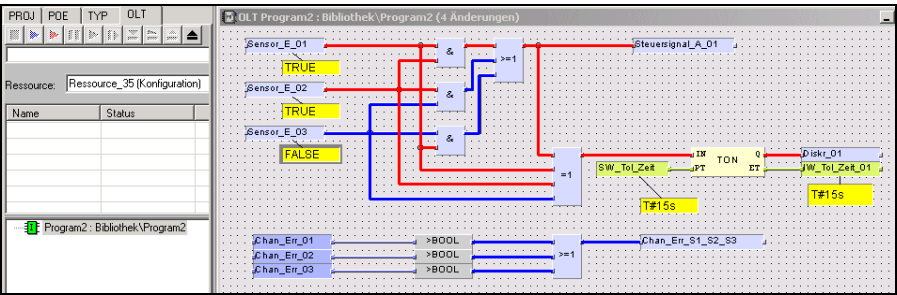


Fig. 38: Teste ONLINE

6 Conceitos para tipos de recurso

Neste Capítulo:

- Atribuição de tipos de sistema de comando e módulos de E/S
- Geração de código
- Carregar ao sistema de comando
- Funções Online
- Documentação do hardware

Depois de ter elaborado e testado o esquema de funções de forma independente do hardware, ocorre a atribuição a um determinado recurso.

Um recurso, de acordo com a IEC 61131-3, é um sistema que executa um programa e opera o nível de E/S.

O tipo do respectivo sistema é seleccionado de uma lista de todos os tipos de recurso disponíveis.

6.1 Criação de um recurso

Um recurso é um equipamento que pode conter um programa de aplicação. Equipamentos que não podem conter um programa de aplicação são denominados de Remote I/O.

Uma exceção é o HIMatrix F3 DIO 20/8 01 que é tratado como um recurso, apesar de ser uma Remote I/O.

- Recursos (p.ex., HIMatrix F30, F35, F60)
A criação de um recurso ocorre na configuração, na gestão de projetos. Este recurso, antes de mais nada, é um “recurso neutro” ao qual ainda não foi atribuído nenhum tipo específico de hardware.
- HIMatrix F3DIO20/8 01
É criado na configuração, no Hardware Management. Na criação ocorre simultaneamente a atribuição do tipo. Como essa Remote I/O possui as mesmas possibilidades de comunicação como um recurso, é tratado no Hardware Management como um recurso.
- Remote I/Os (p.ex., HIMatrix F2, F3DIO20/8 02)
Esses equipamentos só podem comunicar-se de forma direcionada à segurança com o seu recurso de nível superior atribuído. A criação de uma Remote I/O ocorre no Hardware Management, no diretório “Remote I/O” do recurso de nível superior. Veja também Capítulo 6.4.1.

Passo 1: Criação de um recurso:

- ☐ Abrir o menu de contexto da configuração na gestão de projetos.
- ☐ Clicar sobre **New, Resource**.

Passo 2: Renomear o recurso:

- ☐ Clicar duas vezes lentamente sobre o nome do recurso. Um campo de introdução de dados se abre e o nome pode ser alterado.
- ☐ Ou Clicar com o botão direito do mouse no recurso e selecionar a função **Rename** no menu de contexto.

Passo 3:

Criação de uma HIMatrix F3 DIO 20/8 01:

- ☐ Alternar ao Hardware Management.
- ☐ Abrir o menu de contexto da configuração.
- ☐ Clicar em **New** e selecionar **HIMatrix F3 DIO 20/8 01**.

Passo 4:

Renomear o HIMatrix F3 DIO 20/8 01:



- ☐ Abrir o menu de contexto do recurso e selecionar **Properties**.
- ☐ Introduzir no campo "Name" um nome.
- ☐ Introduzir no campo "System ID [SRS]" um valor maior do que 0.
- ☐ Confirmar a introdução com **OK**.

Nota:

*Também é possível renomear uma Remote I/O, como acima descrito. Porém, nesse caso, observar que o tipo de Remote I/O deve permanecer no nome.
Com exceção do tipo de equipamento, todos os demais parâmetros podem ser novamente alterados.*

Para criar uma Remote I/O, veja Capítulo 6.4.

6.2 Atribuição de um tipo de programa a um recurso

Ao criar um recurso, o mesmo já está associado a uma instância de tipo . Essa instância de tipo pode ser usada diretamente, clicando duas vezes nela e editando a mesma. Ou, então, é possível apagar a instância de tipo e atribuir ao recurso um tipo de programa que já foi criado numa biblioteca. A instância criada com isso é uma instância de programa  (veja também Capítulo 4.3.2).

A atribuição de um programa a um recurso ocorre na gestão de projetos:

- ☐ Para este fim, abre-se o menu de contexto do recurso.
- ☐ Selecionar **New, Program Instance**.
- ☐ Selecionar na janela de diálogo um tipo de programa da biblioteca.
O tipo de programa aparece a seguir com o mesmo nome na janela de estrutura no recurso como instância de programa.

6.3 Atribuir um tipo de recurso

O tipo de recurso é atribuído no Hardware Management:

- ☐ Abrir o menu de contexto do recurso.
- ☐ Selecionar **Properties**.
- ☐ Selecionar no campo “Type” o tipo de recurso desejado.
- ☐ Introduzir no campo “System ID [SRS]” um valor maior do que 0 e clicar em **Apply**.

Nota: O System ID (corresponde à proporção de sistema do SRS = System-Rack-Slot) pode ser comparado a um número de participante e só pode ser usado uma única vez no projeto.

Estrutura de recurso sem tipo de recurso atribuído

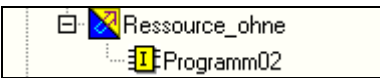


Fig. 39: Estrutura de recurso sem tipo atribuído

Estrutura de recurso com tipo de recurso atribuído

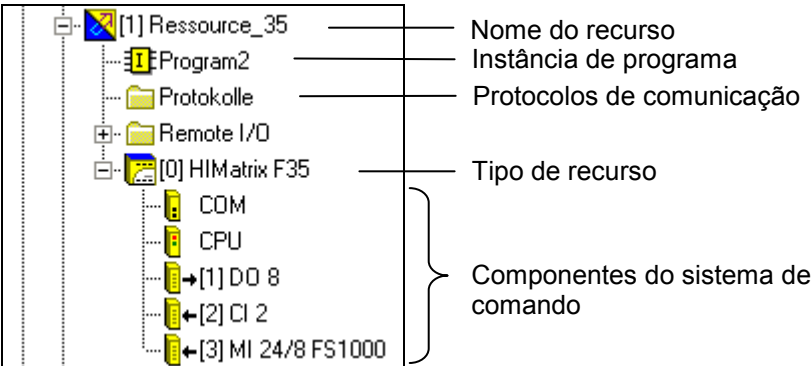


Fig. 40: Estrutura de um F35

6.4 Atribuição de módulos de E/S

6.4.1 Atribuir módulos

Num sistema modular (p.ex., HiMatrix F60), podem ser atribuídos até seis módulos de entrada/saída e no máximo 64 Remote I/Os (RIOs). A um sistema compacto podem ser atribuídas no máximo 64 Remote I/Os.

Para atribuir módulos de entrada/saída, proceder como segue:

- ☐ Abrir o menu de contexto da lista de tipos de recursos no Hardware Management.
- ☐ Selecionar **New** e depois o módulo de E/S desejado.
- ☐ Indicar nas propriedades do módulo o slot correto (veja também Capítulo 7.7.4).

Para atribuir Remote I/Os, proceder como segue:

- ☐ Abrir o menu de contexto da lista *Remote I/Os*.
- ☐ Selecionar **New** e depois a Remote I/O desejada.
- ☐ Alterar nas propriedades o Rack ID para um valor maior do que 0.

6.4.2 Atribuir sinais aos canais de E/S

Sinais que foram definidos no editor de sinais anteriormente pode ser atribuídos aos canais de hardware individuais (veja também Capítulo 7.7.5).

- ☐ Abrir o editor de sinais no menu **Signals**.
- ☐ Selecionar no menu de contexto de um módulo **Connect Signals**.
- ☐ Organizar janelas lado a lado verticalmente.
- ☐ Puxar os sinais do editor de sinais por Drag & Drop para os canais a serem usados.

Veja também Capítulos 7.7.5 e 7.8.3.

6.4.3 Atribuir sinais de sistema

Sinais de sistema são sinais que contêm informações sobre o status da CPU, da comunicação entre os sistemas ou os módulos de E/S. Adicionalmente, servem para a parametrização de canais de E/S.

O procedimento é idêntico ao da conexão de canais de E/S, como descrito no Capítulo 6.4.2.

O significado dos sinais de sistema pode ser conferido nos manuais de sistema e nas folhas de dados dos sistemas de comando (veja também Capítulo 7.8.3).

6.5 Gerador de código

O gerador de código compila as introduções gráficas das funções da área de desenho para obter um código executável no sistema eletrônico programável (PES) e gera uma versão inequívoca desse código.

O gerador de código é iniciado a partir do menu de contexto de um recurso. De acordo com o tipo de equipamento, isso ocorre na gestão de projetos ou no Hardware Management.

O resultado do gerador de código é emitido no indicador de status e erros. Veja também Capítulo 7.10.

Nota: Para aplicações direcionadas à segurança, o gerador de código deve ser iniciado duas vezes e as somas de verificação (CRCs) devem ser comparadas entre si. Só se as somas de verificação forem idênticas está garantido que o código está livre de erros.

6.6 Control Panel

Através do Control Panel, é carregado um programa ao sistema de comando, o sistema de comando é iniciado, parado e os parâmetros de conexão podem ser alterados, etc.

Nota: Antes que o código criado pelo gerador de código possa ser carregado ao sistema de comando, todos os endereços IP nos sistemas de comando precisam estar corretamente configurados.

- ☐ Para iniciar o Control Panel para um recurso, clicar com o botão direito do mouse sobre o recurso na árvore de estrutura e depois selecionar no menu de contexto a função **Online, Control Panel** (veja também Capítulo 7.12).

6.7 Documentação do hardware

No Hardware Management, a documentação da estrutura do hardware e da configuração do hardware são impressas.

É possível imprimir a documentação completa de todos os recursos, ou apenas a parte do objeto anteriormente selecionado.

7 Projeto de exercícios

7.1 Criar um projeto

Passo 1: Iniciar o ELOP II Factory:

- ☐ Clicar no menu Iniciar do Windows em **Programas**, item de menu **ELOP II Factory** e no programa **ELOP II Factory**.
- ☐ Alternativamente, também é possível clicar duas vezes no **ícone** ELOP II Factory , na área de trabalho.




Fig. 41: Iniciar o ELOP II Factory

Passo 2: Criar novo projeto:

- ☐ Selecionar do **Project** a função **New**.



Fig. 42: Criar novo projeto

- ☐ Ou clicar no símbolo  , na barra de símbolos.

Passo 3: Introduzir o caminho do projeto e o nome.

- ☐ Selecionar o diretório onde o novo projeto deve ser criado na árvore de diretórios (Fig. 43, à esquerda). Introduzir no campo Object name“ o nome para o novo projeto e clicar em **OK**.

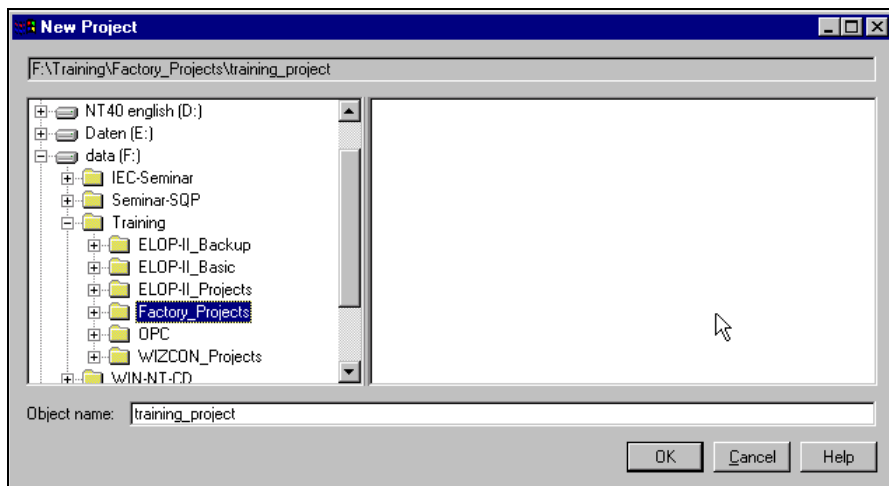


Fig. 43: Criar projeto

Na janela de estrutura da gestão de projetos aparece o novo projeto. No projeto está contida a biblioteca padrão e a biblioteca CFB. Se blocos CFB (Certified Function Blocks) estiverem instalados, são disponibilizados aqui. Ao criar o projeto, imediatamente é criada uma configuração com um recurso e uma instância de tipo. Ao mesmo tempo, abre-se o Hardware Management.

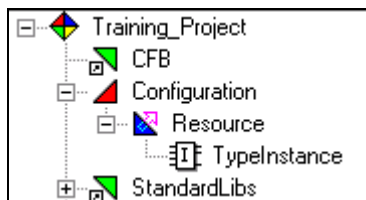


Fig. 44: Estrutura do novo projeto criado

7.2 Criar uma biblioteca e elaborar Program Organisation Units (POU)

7.2.1 Criar uma biblioteca

Passo 1: Como primeiro elemento de estrutura, cria-se uma biblioteca. Mais tarde, ela irá conter as POU's definidas pelo usuário:

- ☐ Clicar na janela de estrutura sobre o projeto.
- ☐ Abrir o menu de contexto mediante clique com o botão direito do mouse.
- ☐ Selecionar **New, Library**.

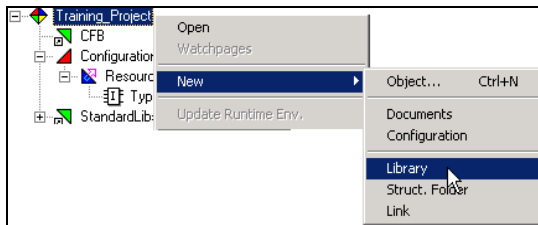


Fig. 45: Criar uma biblioteca

Uma nova biblioteca com o nome de “NewLib” é criada no caminho do projeto.

Passo 2:

Alterar o nome da nova biblioteca de “NewLib” para “Library”:

Ou:

- ☐ Imediatamente depois da nova criação da biblioteca, introduzir o novo nome. O nome pré-ajustado já está marcado para renomear. Encerrar a introdução com a tecla “ENTER”.

Ou:

- ☐ Clicar lentamente duas vezes em “NewLib”. Com o segundo clique, o nome é marcado para renomear. Introduzir o novo nome e encerrar a introdução com a tecla “ENTER”.

Ou:

- ☐ Clicar na janela de estrutura “NewLib”.
- ☐ Abrir o menu de contexto com o botão direito do mouse.
- ☐ Selecionar **Rename** e alterar o nome para “Library”.

7.2.2 Criar unidades de organização do programa “Program Organisation Units” (POU)

Com a ajuda de POUs, um programa é dividido em muito “programas parciais”, e, assim, torna-se mais fácil organizar o mesmo. As POUs deveriam conter lógica que é necessária várias vezes da mesma forma. Veja a este respeito também as explicações no Capítulo 4.2.

Passo 1: Criar na biblioteca “Library” suas próprias unidades de organização do programa (POUs):

- ☐ Clicar em “Library” e abrir o menu de contexto.
- ☐ Selecionar **New, Function Block Type**.

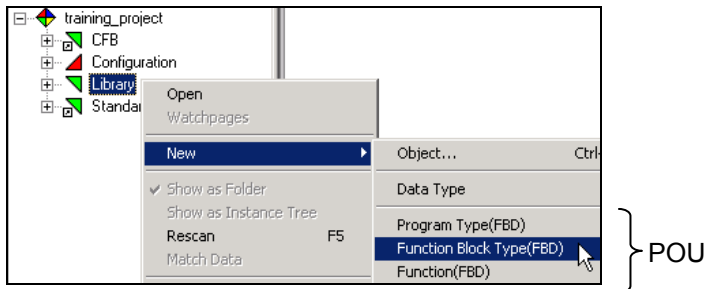


Fig. 46: Criar tipo de bloco funcional

- ☐ Clicar no sinal de “+” antes de “Library”.
Na estrutura é exibida uma nova POU.
- ☐ Renomear o tipo de bloco funcional como descrito no passo 2.
- ☐ Repetir o passo 3 e criar desta maneira um **Program Type** caso não queira trabalhar com a instância de tipo que já existem no recurso. Diferençar, veja Capítulo 4.3.2.
- ☐ Renomear o tipo de programa como descrito no passo 2.

Depois desses passos, a estrutura de projeto se apresenta assim:

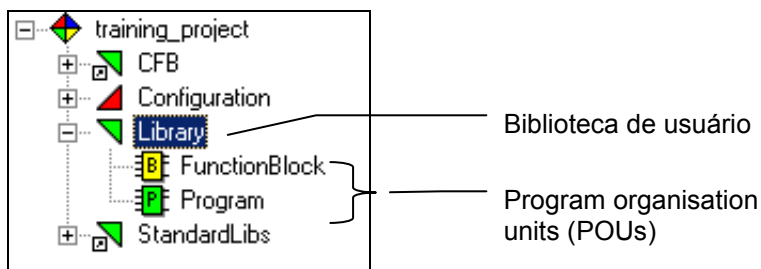



Fig. 47: Estrutura de projeto depois desse exercício

7.3 Editar blocos funcionais

Passo 1: Abrir o bloco funcional:

- ☐ Abrir o bloco funcional da Fig. 47 com clique duplo no símbolo do bloco funcional . O FBD Editor se abre e mostra as diferentes áreas onde é possível efetuar definições.

7.3.1 Declaração de variáveis

Passo 2: Selecionar o tipo de variável:

- ☐ Clicar no registro com o tipo de variável desejado. Selecionar, p.ex., “VAR_INPUT”, como mostrado na Fig. 48.

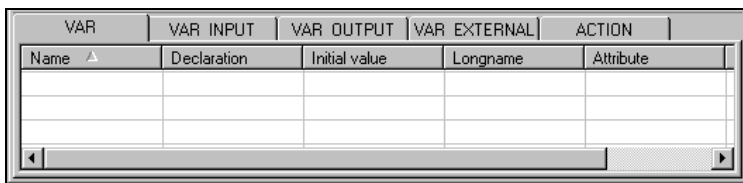


Fig. 48: Editor de declaração de variáveis

Nota: Para as POU's do tipo função ou bloco funcional, VAR_INPUT e VAR_OUTPUT são usadas como variáveis de entrada e saída à POU do próximo nível mais elevado. O tipo de variável VAR_EXTERNAL apenas pode ser usado dentro de um bloco funcional, mas não dentro de uma função. Uma variável do tipo VAR_EXTERNAL é reconhecida em todos os locais onde uma variável do tipo VAR_EXTERNAL ou VAR_GLOBAL foi criada com o mesmo nome. Se uma variável do tipo VAR_EXTERNAL deve ser conectada a canais de entrada ou saída, ou se o valor da variável deve ser exibido no Force Editor, então a variável deve ser criada no editor de sinais e puxada via Drag & Drop para a declaração de variáveis ou a área de desenho do bloco funcional (veja também Capítulo 7.7.5 a este respeito).

Passo 3: Introduzir os dados da variável:

- ☐ Abrir o diálogo da declaração de variáveis mediante clique duplo na área vazia do registro de variáveis.
- ☐ Sobrescrever o valor prédefinido “I1” no campo “Name” com “Variable1”.
- ☐ Selecionar o tipo de dados no campo “Declaration”, p.ex., BOOL.
- ☐ Definir a “posição” das variáveis no bloco. No exemplo na Fig. 50, a “Variable1” é mostrada na segunda posição superior esquerda.

Variable Declaration : FunctionBlock

VAR_INPUT

Declaration (direct input)

Name: Variable1

Declaration: BOOL

Initial value:

Declaration (dialog driven)

Derivation types

Direct Subrange

Array Enumeration

Position

Position: 2

Inverted

alternate I/O-identifier

Techn. unit/scaling

Techn. unit:

min. value: max. value:

internally represented as

Long name: Description

Physical address:

Attributes:

BYREF

CONST

RETAIN

Comment:

Add Update Delete Undo Close Help

Fig. 49: Declaração de uma variável dentro do bloco funcional

- Se necessário, introduzir no campo “Long name” um nome significativo por extenso para a “Variable1”. O nome extenso serve como descrição adicional da variável.
- Ao acionar o botão **Add**, a variável é inserida na lista de variáveis.

O diálogo permanece aberto e uma nova variável com o mesmo nome, mas numeração contínua é preparada. A nova variável possui o mesmo tipo de dados e é colocada na próxima posição. O “Long name” permanece inalterado.

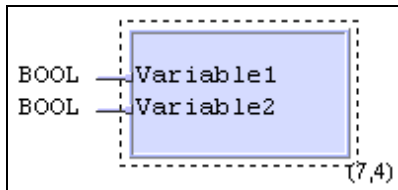


Fig. 50: Declaração de interfaces

Nota: Com VAR_INPUT e VAR_OUTPUT, a variável aparece na posição indicada da declaração de interfaces.

7.3.2 Definir a declaração de interfaces (representação gráfica)

Passo 1: Definir as propriedades do bloco na janela de declaração de interfaces:

- ☐ Clicar com o botão direito do mouse no bloco e selecionar **Properties** do menu de contexto.

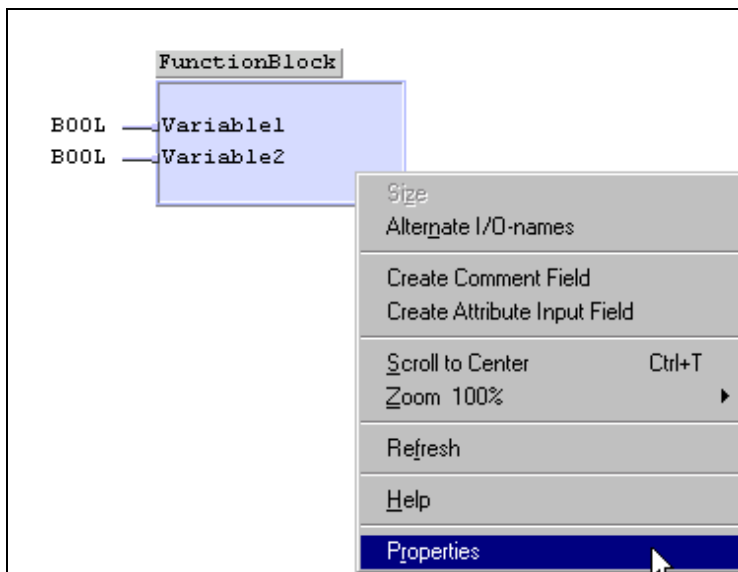


Fig. 51: Menu de contexto do bloco

Passo 2:

Introduzir uma denominação do bloco:

- ☐ Introduzir no campo “POU text” um nome para o bloco. O texto do bloco deveria ser idêntico com o nome do bloco na biblioteca. Conforme necessário o tipo de fonte pode ser adaptado pelo botão **Edit font...**

O nome introduzido aparece no meio do bloco (veja Fig. 54).

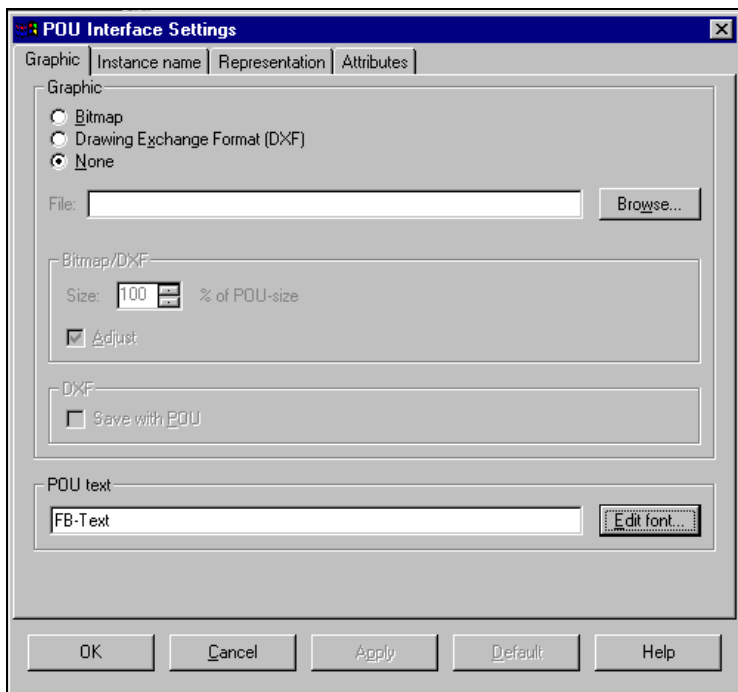


Fig. 52: Introduzir o texto do bloco

Passo 3:

Definir um nome de instância:

- ☐ Selecionar o registro **Instance name**.
- ☐ Introduzir no campo “Instance name” um nome e ativar a caixinha de controle **Display**.
- ☐ Adaptar **Font** e **Alignment** conforme necessário.
O nome da instância por padrão aparece do lado superior esquerdo do bloco (veja Fig. 54).

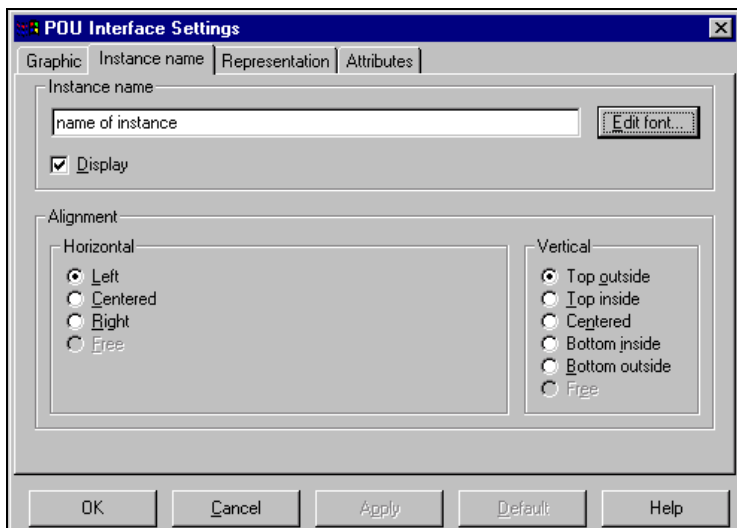


Fig. 53: Introduzir e formatar o nome de instância

Nota:

O texto do bloco serve para a descrição da lógica dentro do bloco, p.ex., “2 out of 3”. O texto permanece idêntico para todas as instâncias do bloco.

O nome de instância aqui introduzido serve como ajuste prévio. O nome de instância pode ser adaptado às função ao usar o bloco, p.ex., “2 out of 3 Temperature Monitoring Point 15”.

Se não ocorrer nenhuma adaptação, o nome ainda irá conter um índice de contagem, para ser inequívoco.

Passo 4:

Adaptar o tamanho do bloco:

- Apontar o mouse para o canto inferior direito. Quando uma seta dupla preta aparecer, manter o botão esquerdo do mouse acionado e alterar o tamanho do bloco conforme às suas preferências.

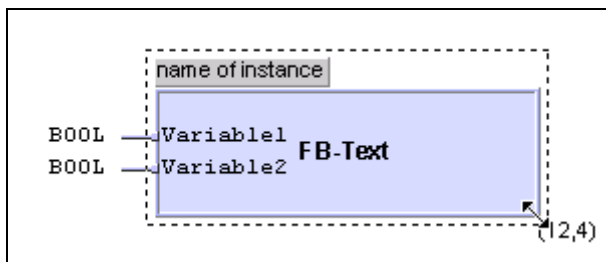


Fig. 54: Adaptar a representação

7.3.3 Introdução de lógica na área de desenho do bloco

Passo 1: Posicionar elementos de lógica das bibliotecas na área de desenho mediante Drag & Drop:

- ☐ Abrir na janela de estrutura a biblioteca **StandardLibs**, clicando no símbolo “+”.
- ☐ Abrir a biblioteca **IEC61131-3** das StandardLibs, e nela o elemento **Bitstr**.
- ☐ Clicar com o botão esquerdo do mouse em **AND**, manter o botão do mouse pressionado e puxar o bloco da janela de estrutura para a área de desenho.
Ao puxar, aparece a previsão do bloco.
- ☐ Depois de soltar o botão do mouse, o bloco é posicionado no local desejado.

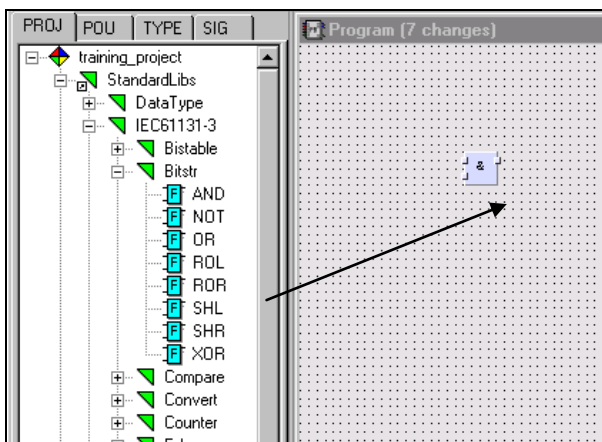


Fig. 55: Drag & Drop de blocos funcionais

Nota: O procedimento para a inserção de blocos das bibliotecas via Drag & Drop também vale para blocos funcionais definidos pelo usuário.

Passo 2:

Completar os dados da folha:

Como, ao posicionar o bloco **AND**, foi inserido o primeiro elemento nesta página, (veja a este respeito também Capítulo 5.4), abre-se a automaticamente janela de diálogo “Edit Page Data”.

- ☐ Introduzir nos campos “Short name” e “Long name” denominações significativas para a página.

Edit Page Data

Page data

Page no. **A.0** Short name:

Long name:

created by (logon-ID): **vsnetz** Date: **03/06/03**

tested by: Date:

Labeling systems

ULS:

PPLS:

☒ Inherit labeling systems from page options

Page-ID: ☒ Inherit

OK Help

Fig. 56: Editar dados da página

Nota:

Se necessário, bloquear a numeração automática do “Shortname” em “Page data” e nas propriedades da área de desenho e atribuir o seu próprio nome.

Passo 3: Ampliar a lógica com outros blocos:

- ☐ Inserir outros blocos funcionais das bibliotecas na área de desenho, como descrito no **Passo 1**.
- ☐ Duplicar blocos idênticos mantendo a tecla **CTRL** pressionada e puxando um bloco funcional já existente com o botão esquerdo do mouse para uma outra posição na área de desenho. Soltar primeiro a tecla do mouse, caso contrário, o bloco apenas é deslocado.

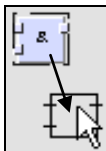


Fig. 57: Copiar bloco funcional

Nota: Se blocos são posicionados em sobreposição ao inserir, a ação é interrompida com um sinal acústico. Se o ponteiro do mouse estiver numa posição proibida, aparece um símbolo de proibição.

Passo 4: Ligar a grade e usar o zoom:

- ☐ Ligar a grade (Toggle grid).

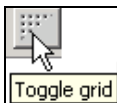


Fig. 58: Botões para grade

- ☐ Ajustar o zoom até o recorte da tela que deve ser editada.



Fig. 59: Botões para zoom

Nota: Os botões do lado direito referem-se à área de desenho do bloco funcional aberto.

Passo 5:

Inserir variáveis na lógica:

- Clicar na lista de variáveis sobre um nome de variável, manter o botão esquerdo do mouse pressionado e puxar a variável para a área de desenho. Ao puxar, é exibida a previsão do campo de valor.

Depois de soltar o botão do mouse, a variável é posicionada e no campo de valor aparece o nome da variável.

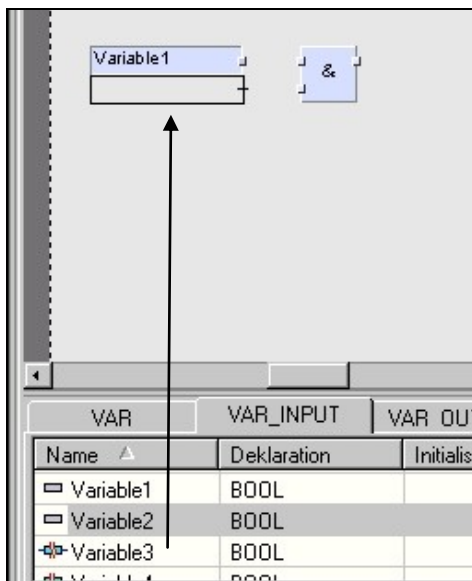


Fig. 60: Drag & Drop de variáveis

Nota:

*Se tiver (nenh)um comentário associado anexo, o anexo automático de comentários pode ser ativado (desativado) nas propriedades da área de desenho em **Comment**, **Attached Comment**, **AC value field drawing field**.*

Passo 6:

Desenhar linhas de conexão entre as variáveis (campos de valores) e blocos de lógica.

- ☐ Colocar o ponteiro do mouse sobre o ponto de conexão da variável (= output).
- ☐ Manter o botão esquerdo do mouse pressionado e puxar uma linha para a direita.
- ☐ Puxar a linha até a entrada do bloco e soltar o botão do mouse.

O resultado é uma linha de ligação entre dois pontos de conexão.

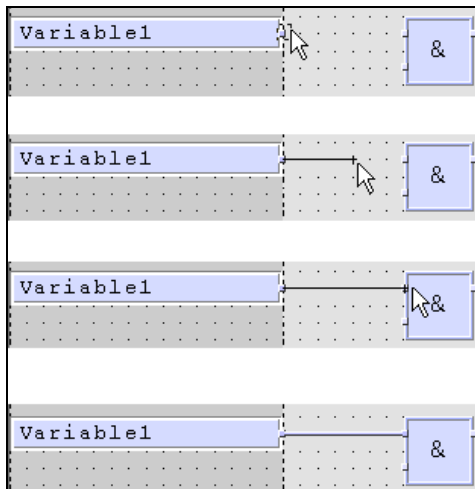



Fig. 61: Desenhar uma linha

Nota: A cor da linha depende do respectivo tipo de dados (BOOL, integer, real etc.).
Se a linha for puxada para além do ponto de conexão ou se o tipo de dados da entrada não combinar com o tipo de dados da variável, aparece uma placa de proibição.



Nota: A posição das linhas de conexão pode ser alterada conforme necessário editando a mesma com o mouse manualmente.
Para este fim, manter a tecla de Shift pressionada e adicionalmente segurar o botão esquerdo do mouse e manter. Agora, puxar a linha de conexão para a nova posição. Soltar o botão do mouse e depois a tecla de Shift.

7.3.4 Terminar a edição do bloco

Depois de terminar a lógica e a declaração de interfaces, o bloco precisa ser armazenado mediante clique no símbolo do disquete .

Nota: Cada bloco novo ou alterado deve ser testado offline, para poder detectar comportamento incorreto o mais precocemente possível (veja também Capítulo 7.6).

7.4 Editar o tipo de programa ou uma instância de tipo

- Abrir o programa ou a instância de tipo (veja Capítulo 7.2.2 como criar) com clique duplo no símbolo  ou .
O FBD Editor se abre e mostra as diferentes áreas onde é possível efetuar definições.

7.4.1 Declaração de variáveis

O procedimento para a declaração de variáveis é idêntico à declaração nos blocos funcionais (veja Capítulo 7.3.1).

Nota: Para um programa são usadas as variáveis VAR, VAR_GLOBAL e VAR_EXTERNAL. Variáveis com ligação a canais de entrada/saída, sinais de comunicação ou sinais de sistema precisam ser primeiramente criadas com o editor de sinais no Hardware Management e depois puxadas via Drag & Drop para dentro da declaração de variáveis ou da área de desenho do programa. São criadas automaticamente no registro VAR_EXTERNAL (veja também Capítulo 7.8).

7.4.2 Definição da declaração de interfaces

Esta função é dispensada para POU's do tipo programa. Como o programa em si não pode ser utilizado dentro de outras POU's, não há declaração de interfaces.

7.4.3 Introdução de lógica na área de desenho

O procedimento para a introdução de lógica corresponde à introdução no bloco funcional (veja Capítulo 7.3.3).

7.5 Criação de um recurso:

Ao criar o projeto, já é criada uma configuração com um recurso e uma instância de tipo. Se outros recursos forem necessários, estes ainda devem ser criados.

Se estiver trabalhando com uma versão mais antiga do ELOP II Factory, que não gera automaticamente a configuração, a mesma ainda precisa ser criada antes.

Para este fim, abrir o menu de contexto do projeto e com um clique do botão direito do mouse no nome do projeto, selecionar **New**, **Configuration**. Se necessário, atribuir um novo nome à configuração.

Passo 1:

Criar um novo recurso dentro da configuração:

- ☐ Clicar na janela de estrutura com o botão direito do mouse sobre a configuração e abrir o menu de contexto.
- ☐ Selecionar **New**, **Resource**.

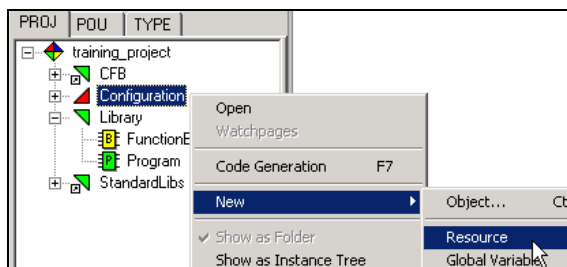


Fig. 62: Criação de um recurso

Nota: Na gestão de projetos, o recurso é criado apenas como um chamado recurso neutro “neutral resource”. O tipo de recurso é atribuído mais tarde no Hardware Management.

Passo 2: Atribuir uma instância de programa ao recurso:

Normalmente, uma instância de tipo que possui o seu próprio tipo de programa já está atribuída a um recurso. Se esta estrutura deve ser usada, é possível editar a lógica diretamente na instância de tipo (ou seja, dentro do seu tipo).

Porém, se dentro do recurso deve ser processado um programa que está disponível como tipo na biblioteca, este tipo de programa deve ser atribuído explicitamente antes. Anteriormente, a instância de tipo existente deve ser excluída: clicar com o botão direito do mouse na instância de tipo e selecionar **Delete** do menu de contexto.

- ☐ Clicar na janela de estrutura com o botão direito do mouse sobre o recurso e abrir o menu de contexto.
- ☐ Selecionar **New, Program instance...**

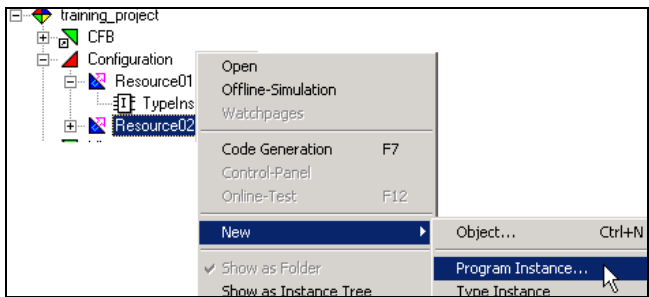


Fig. 63: Criar nova instância de programa

Passo 3:

Selecionar o programa que deve ser processado no recurso:

- ☐ Selecionar o seu programa da biblioteca do projeto.
O bloco de programa é transferido como instância de programa.

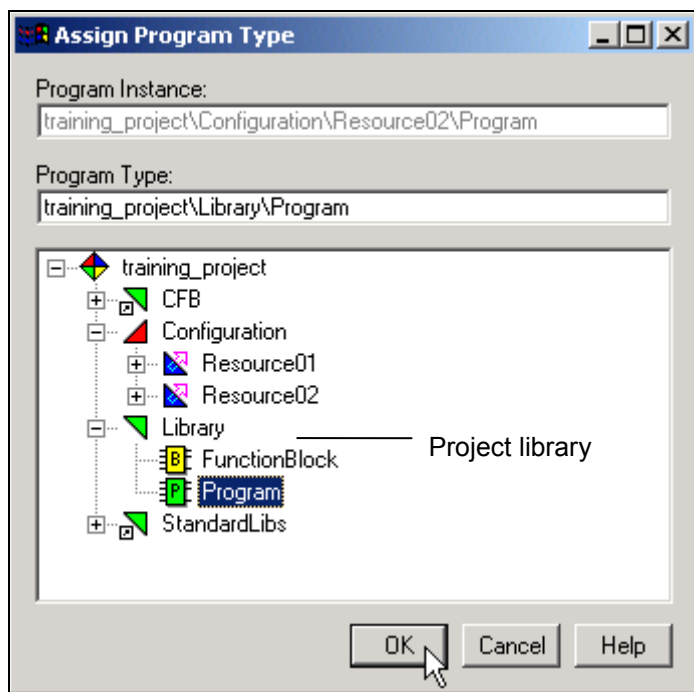


Fig. 64: Atribuir tipo de programa

Depois desses passos, a estrutura de projeto se apresenta assim:

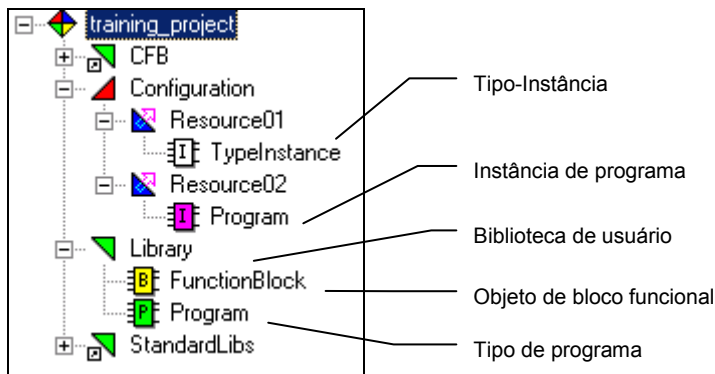


Fig. 65: Estrutura do projeto

Nota: A instância de programa só é um ponteiro para um tipo de programa da biblioteca.
O tipo de programa na biblioteca do usuário contém o “software”, ou seja, a lógica e as variáveis.

7.6 Simulação Offline

7.6.1 Simulação offline de um programa

Na simulação offline, a função de um bloco ou programa é testada sem utilizar o sistema de comando (hardware). Assim, erros de programação podem ser detectados e eliminados já antes da colocação em funcionamento.

O programa a ser testado deve estar atribuído a um recurso (veja também Capítulo 7.5).

Passo 1: Chamar a simulação offline:

- ☐ Abrir o menu de contexto do recurso mediante clique com o botão direito do mouse.
- ☐ Selecionar **Offline-Simulation**.

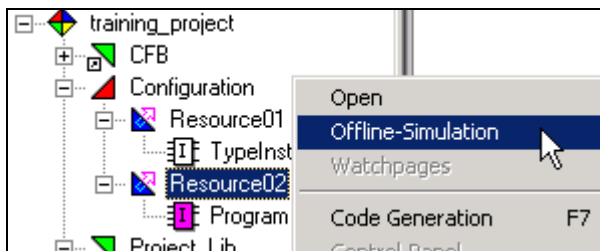


Fig. 66: Ativar a simulação offline

Nota: Depois da geração de código apenas para a lógica, uma nova aba “OLS” (offline simulation) se abre na janela de estrutura.

Passo 2: Iniciar a simulação Offline:

Normalmente, a simulação offline já está no status “Run”. Em versões mais antigas, é possível iniciar a simulação com o botão **Cold boot**.

- ❑ Clicar no botão  **Cold boot**.

Depois de iniciar, o status muda de “Stopped” para “Running”.

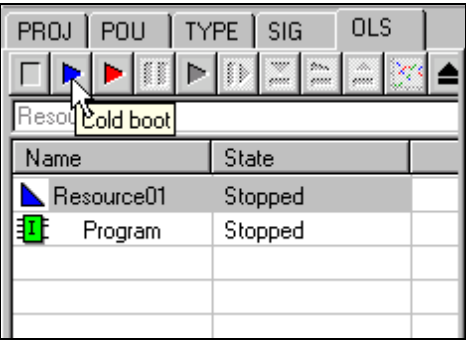


Fig. 67: Iniciar a simulação offline

Passo 3: Abrir o programa na simulação offline:

- ☐ Clicar na janela de estrutura duas vezes sobre o projeto para abrir o programa.

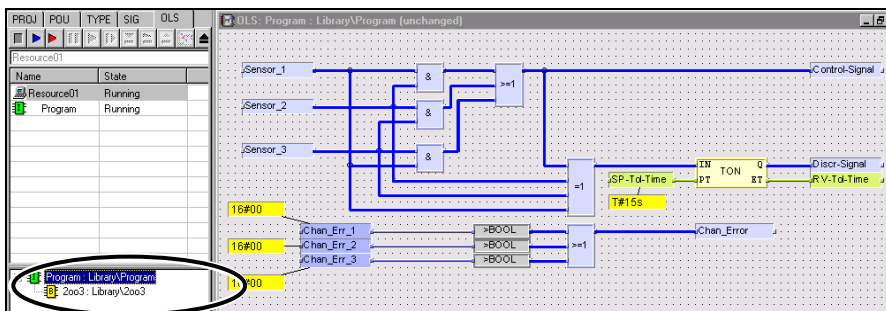


Fig. 68: Simulação offline do programa

Nota: É possível abrir vários blocos simultaneamente e observar a sequência de sinais entre os blocos. Também é possível abrir os blocos diretamente na lógica, na área de desenho.

Passo 4: Alterar o estado dos sinais e testar a lógica:

a) Alteração com um campo Online Test (campo OLT):

- ☐ Clicar no campo de valor e segurar o botão esquerdo do mouse.
- ☐ Puxar o mouse para fora do campo de valor e soltar num local livre na tela. Uma previsão do campo OLT é mostrada.
- ☐ Posicionar o campo OLT com um clique do mouse.
- ☐ Alterar o estado de sinal da variável mediante clique duplo no campo OLT.

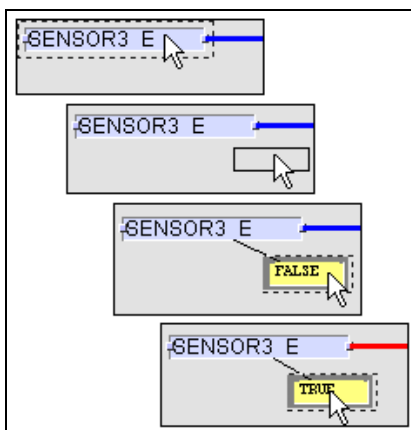


Fig. 69: Criar um campo OLT

Nota: Depois de inserir campos OLT, ao fechar o bloco funcional, aparece uma solicitação de confirmação “Save change?” Com **Yes**, os campos OLT são salvos com o projeto. No caso de **No**, os campos OLT criados são descartados. Campos OLT já podem ser criados através do menu de contexto, ao criar o programa do respectivo elemento.

b) Alteração direta no campo de valor:

- ☐ Colocar o ponteiro do mouse sobre o campo de valor cujo valor deve ser alterado.
- ☐ Pressionar a tecla ALT e manter pressionada. O valor da variável é exibido.
- ☐ Alterar o valor com um clique do mouse sobre o campo de valor.
- ☐ Soltar a tecla ALT. Novamente é exibido o nome da variável.

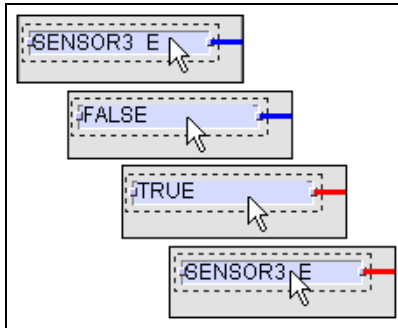


Fig. 70: Comutar a visualização de campos de valor mediante tecla ALT

Nota: Apenas é possível alterar valores que não foram escritos pela lógica.

Passo 5: Encerrar a simulação Offline:

- ☐ Para este fim, clicar em **Close OLS**, na barra de símbolos.



Fig. 71: Encerrar a simulação offline

7.6.2 Simulação offline de um bloco definido pelo usuário

Passo 1: Integrar o bloco funcional a ser testado num programa:

- ☐ Puxar o bloco a ser testado para dentro de um programa e salvar o mesmo.

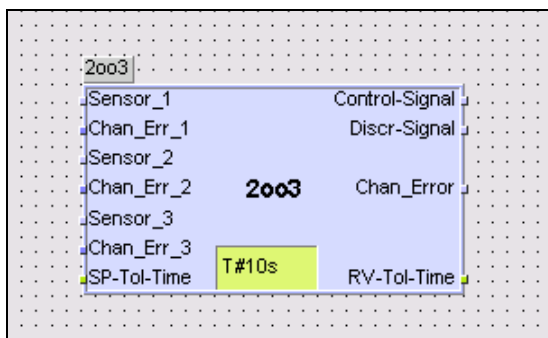


Fig. 72: Bloco sem ligação

Nota: Se for o caso, ligar as repostas necessárias (p.ex., para sinais de grupo) para que o bloco possa ser testado inclusive a função de retroalimentação.

Passo 2: Iniciar a simulação offline como descrito em Capítulo 7.6.1.

- ☐ Depois de ter aberto o programa, clicar no bloco com um clique duplo nele.

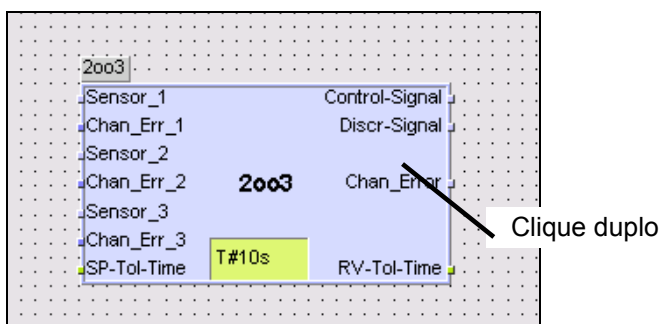


Fig. 73: Abrir bloco mediante clique duplo

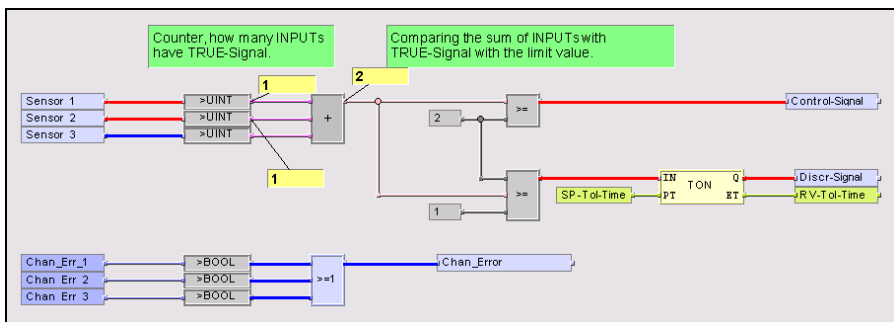


Fig. 74: Lógica no bloco

- ☐ Testar a função do bloco alterando via clique do mouse com a tecla ALT pressionada os valores VAR_INPUT (eventualmente, VAR_EXTERNAL).
- ☐ Encerrar a simulação offline de novo (veja também Capítulo 7.6.1, passo 5).

7.7 Atribuição de hardware do sistema de comando

Todos os ajustes relacionados ao hardware devem ser efetuados no Hardware Management. O Hardware Management é exibido numa janela separada que é chamada ao criar ou abrir um projeto.

7.7.1 Atribuir um tipo de recurso

Passo 1: Abrir as propriedades do recurso:

- ☐ Clicar no Hardware Management com o botão direito do mouse sobre o recurso e abrir o menu de contexto.
- ☐ Selecionar **Properties**.

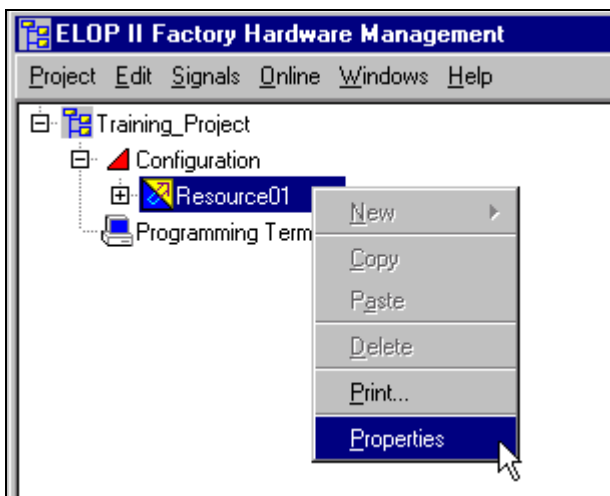


Fig. 75: Abrir as propriedades do recurso

Passo 2:

Selecionar o tipo de recurso:

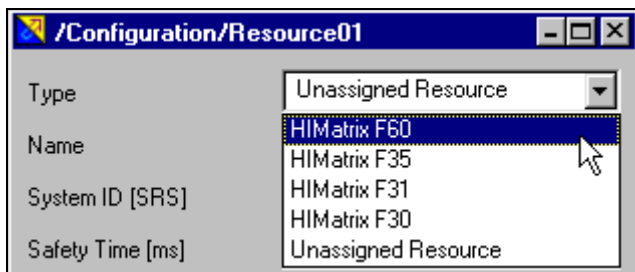


Fig. 76: Selecionar o tipo de recurso

- ☐ Selecionar na janela de seleção “Type” o tipo do sistema de comando HIMatrix.
- ☐ Introduzir no campo “System ID [SRS]” uma identificação do sistema e clicar em **Apply**.

Nota: O ID de sistema (SRS = System-Rack-Slot) corresponde a um número de participante e pode ser usado apenas uma vez no projeto.
Valores de 1 a 65535 são possíveis.

Passo 3: Editar propriedades do recurso:

Nota: Somente depois de ter transferido ao projeto o tipo de recurso e o ID de sistema com o botão **Apply**, os outros parâmetros podem ser alterados também.

- ☐ Alterar os outros parâmetros do recurso conforme necessário.
- ☐ Encerrar a introdução com **OK**.

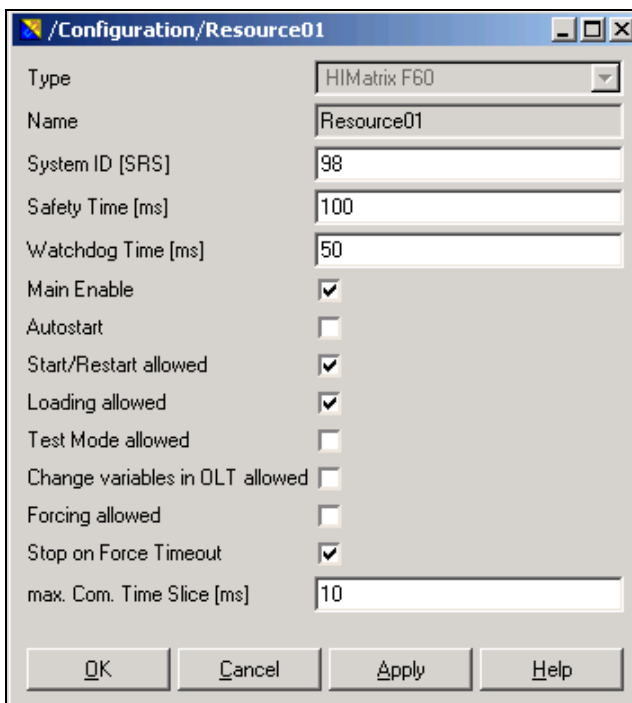


Fig. 77: System ID

Nota: As descrições exatas das propriedades podem ser consultadas na ajuda online, clicando no botão **Help**.

Nota: Com ajuda de propriedade “Autostart” ativa-se o arranque automático do programa de aplicação depois de ligar a tensão de operação. Se e como o programa de aplicação arranca (cold/warm start) é definido nas propriedades da instância de programa, no Hardware Management. (Veja próximo capítulo.)

7.7.2 Definir os ajustes de iniciar do programa

Nas propriedades do recurso foi definido se por princípio deve ocorrer o Autostart ou não. Aqui define-se de que forma.

- ☐ Abrir o menu de contexto da instância de programa via clique com o botão direito do mouse.
- ☐ Abrir o menu de seleção e selecionar a variante de iniciar desejada.

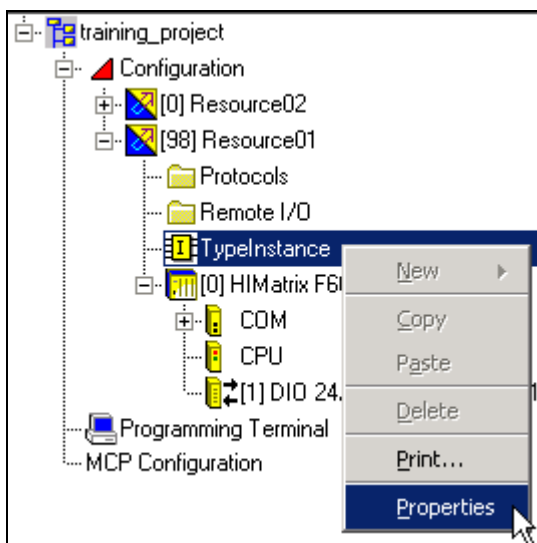


Fig. 78: Chamar as propriedades da instância de programa

As seguintes liberações de Autostart são possíveis:

Warmstart:

Sinais com o atributo “Retain” mantêm o seu valor atual.

Coldstart:

Todas as variáveis e sinais são resetados ao seu valor inicial.

Off:

Não há arranque automático. O programa de aplicação deve ser iniciado pelo aparelho de programação.

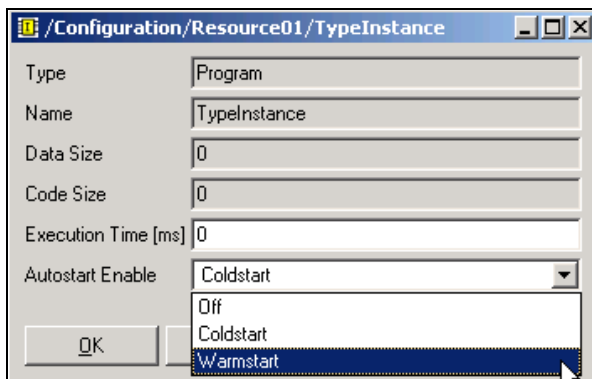


Fig. 79: Variantes de iniciar

7.7.3 Ajustar as propriedades do módulo de comunicação

A comunicação entre o aparelho de programação, os sistemas de comando e entre os sistemas de comando entre si ocorre via Ethernet. A comunicação via Ethernet ocorre através do protocolo UDP/IP. Para este fim, o usuário atribui a cada sistema de comando e Remote I/O na rede um endereço IP (propriedades do módulo COM).

Os endereços IP são endereços lógicos e não possuem relação fixa à interface de comunicação do sistema de comando. Somente o endereço MAC como endereço físico está atribuído de forma fixa à interface de comunicação.

O endereço MAC é programado de forma fixa durante a fabricação do sistema de comando.

Um endereço IP é um número dual de quatro bytes. Cada byte é indicado como número decimal.

Um endereço IP é composto pelo ID de rede (Net ID), Subnet ID, e ID de nóduo (núduo = participante, também Host ID) em conjunto. A definição qual parte do endereço IP contém o ID de rede mais ID de subrede, é efetuada na máscara de subrede. Veja a este respeito o seguinte exemplo.

	192	168	0	25
IP Address	11000000	10101000	00000000	00011001
Subnet Mask	255	255	252	0
	11111111	11111111	11111100	00000000

- Todos os Bits do endereço IP que são mascarados com “1” na máscara de subrede, pertencem ao ID de rede mais ID de subrede.
- Todos os Bits do endereço IP que são mascarados com “0” na máscara de subrede, pertencem ao ID de nóduo.

Mediante o cálculo, no exemplo acima resultam $2^{10} - 1 = 1023$ IDs possíveis de nóduos. Os valores 0 e 255 não são permitidos no último byte.

Importante: O endereço de rede dentro de uma configuração deve ser idêntico para todos os participantes, se não se usam gateway ou roteador.

Se o aparelho de programação e os sistemas de comando não estiverem numa rede própria fechada, então, os parâmetros de rede podem ser ajustados livremente.

Nota: Se o aparelho de programação e os sistemas de comando forem participantes da mesma rede que também é usada por terceiros, entrar em contato com o seu administrador de rede para a atribuição de endereços IP.

Passo 1:

Definir as propriedades do módulo COM:

- ☐ Expandir recurso ou Remote I/O na árvore de estrutura para que todos os módulos se tornem visíveis.

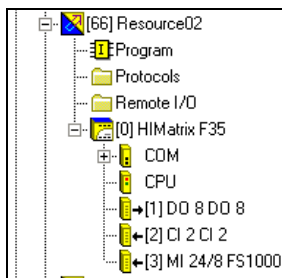


Fig. 80: Estrutura de um recurso ou uma Remote I/O

- ☐ Selecionar **Properties** no menu de contexto do módulo COM.
- ☐ Adaptar o “IP Adress”.
Exemplo: 192.168.0.60.

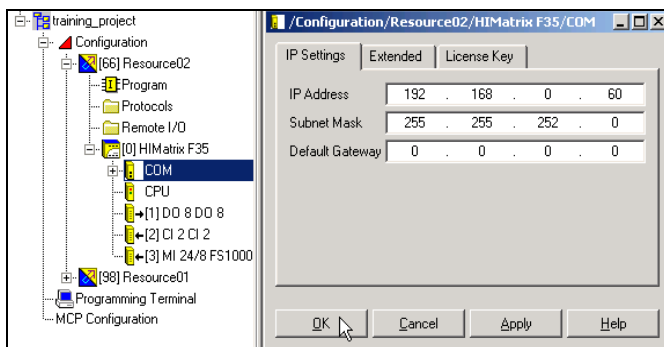


Fig. 81: Propriedades do módulo COM

- ☐ Numa rede fechada, manter os ajustes padrão para Subnet Mask. Caso contrário, ajustar a Subnet Mask conforme as indicações do administrador da rede.
- ☐ Se não for usado um gateway padrão, manter o endereço em “0.0.0.0”.

7.7.4 Adicionar a um recurso modular módulos de entrada/saída

Passo 1: Selecionar os módulos de entrada/saída:

- ☐ Clicar no símbolo “+” antes do recurso para abrir a estrutura do recurso.
- ☐ Clicar com o botão direito do mouse no tipo de recurso.
- ☐ Selecionar **New** no menu de contexto e a seguir um módulo.

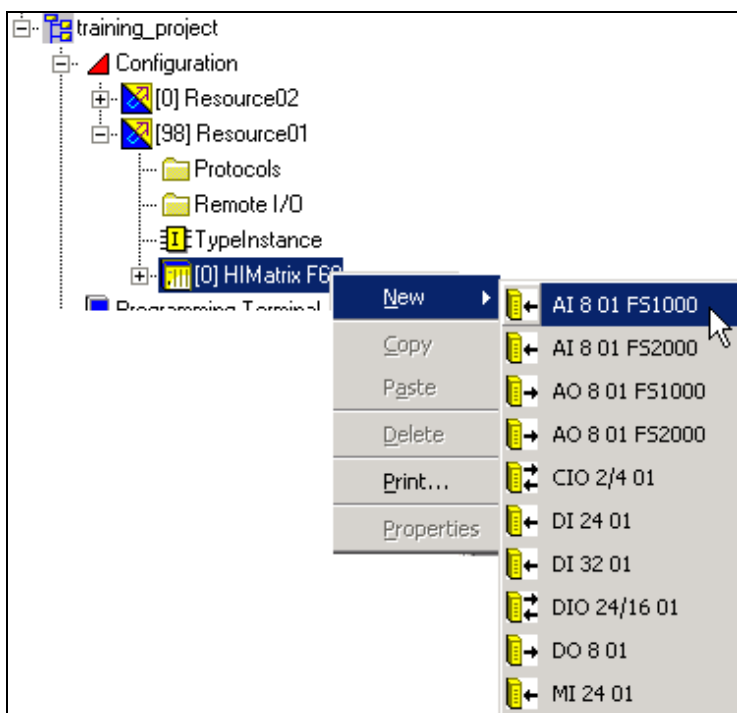


Fig. 82: Adicionar módulos de entrada/saída

Passo 2:

Atribuir um slot ao módulo de entrada/saída:

- ☐ Abrir o menu de contexto do módulo de entrada/saída e Clicar em **Properties**.

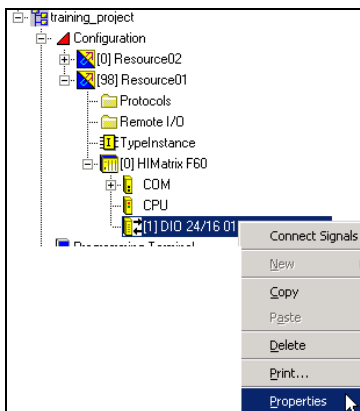


Fig. 83: Chamar as propriedades do módulo

- ☐ Introduzir o slot desejado e eventualmente um outro nome para o equipamento.
- ☐ Encerrar a introdução com **OK**.

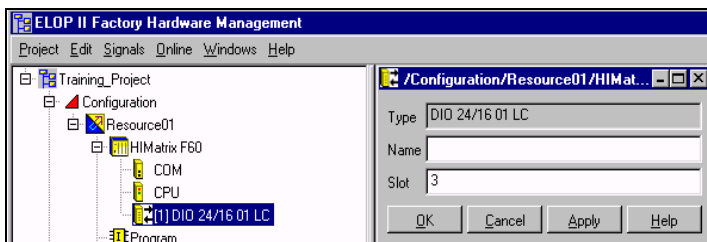


Fig. 84: Indicar o número de slot

Nota

Um novo módulo inserido sempre tem o número de slot 1. O número de slot pode ter valores entre 1 e 6 e apenas pode existir uma vez num recurso.

Os slots são numerados sequencialmente da esquerda à direita. A fonte de alimentação e a CPU não possuem número de slot, pois as suas posições são fixas.

7.7.5 Criar uma Remote I/O (RIO)

Ha dois tipos de Remote I/O:

1. HIMatrix F3 DIO 20/8 01
Essa Remote I/O possui as mesmas capacidades como um recurso em relação à comunicação via safe**e**thernet e também tem um ID de sistema. Por isso, daqui em diante é considerado como um recurso.
2. Remote I/Os do tipo HIMatrix F1x, F2x ou F3x. Estas Remote I/Os apenas possuem capacidades restritas de comunicação e apenas podem trocar dados com os seus recursos de nível superior. Na árvore de estrutura, Remote I/Os são agrupadas no diretório "Remote I/O" como parte de um recurso. Remote I/Os precisam ser parametrizadas com um ID de Rack.

Passo 1: Criação de uma HIMatrix F3 DIO 20/8 01:

- ☐ Abrir o menu de contexto da configuração no Hardware Management com um clique do botão direito do mouse.
- ☐ Selecionar **New** e depois **HIMatrix F3 DIO 20/8 01**.

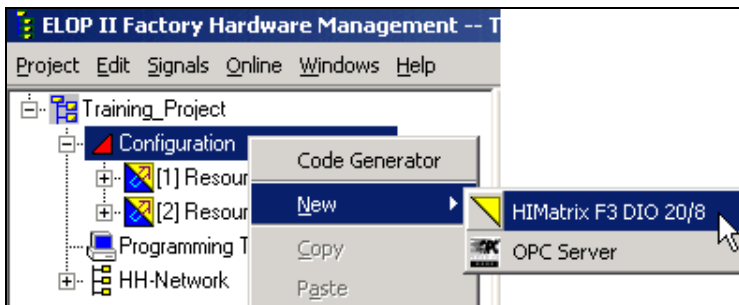


Fig. 85: Criação de HIMatrix F3 DIO 20/8 01

Passo 2:

Alterar o nome do recurso e atribuir um ID de sistema ao mesmo:

- ☐ Clicar com o botão direito do mouse no recurso e selecionar **Properties** no menu de contexto.
- ☐ Introduzir no campo “Name” um nome.
- ☐ Introduzir no campo “System ID [SRS]” um valor maior do que 0.
- ☐ Confirmar a introdução com **OK**.

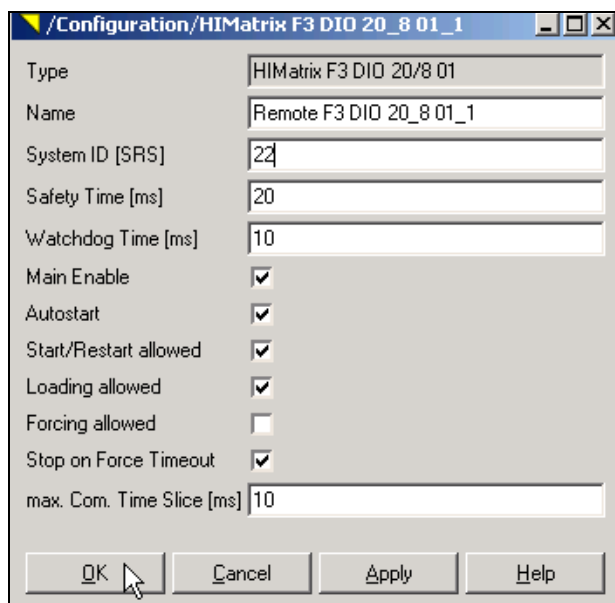


Fig. 86: Propriedades de uma HIMatrix F3 DIO 20/8 01

Nota: Com exceção do tipo de equipamento, todos os demais parâmetros podem ser novamente alterados mais tarde.

Passo 3:

Criar uma Remote I/O:

- ☐ Expandir no Hardware Management o diretório do recurso de nível superior. Abrir o menu de contexto do subdiretório **Remote I/O** com o botão direito do mouse.
- ☐ Selecionar **New** e depois a Remote I/O desejada.

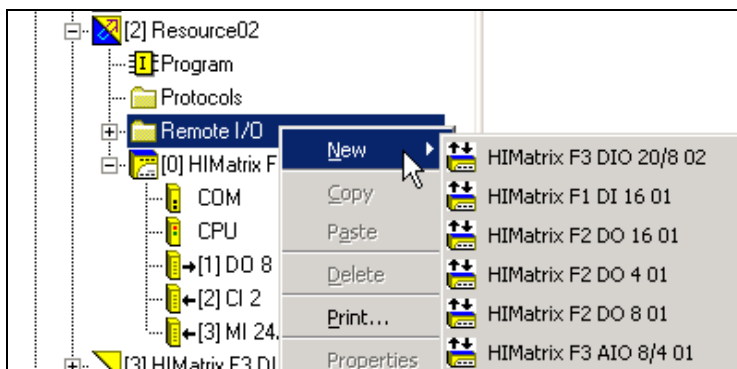


Fig. 87: Inserir uma Remote I/O

Nota:

Se ainda não se criou uma rede HH para a comunicação Peer-to-Peer ou se o recurso de nível superior ainda não foi atribuído a um grupo simbólico, abre-se uma janela de diálogo. Escolher uma rede HH e um grupo simbólico ou criar elementos novos.

Passo 4:

Atribuir um ID de Rack à Remote I/O:

- ☐ Abrir o menu de contexto da Remote I/O com um clique do botão direito do mouse e seleccionar **Properties**.
- ☐ Alterar o Rack ID para um valor > 0 e ≤ 511 .

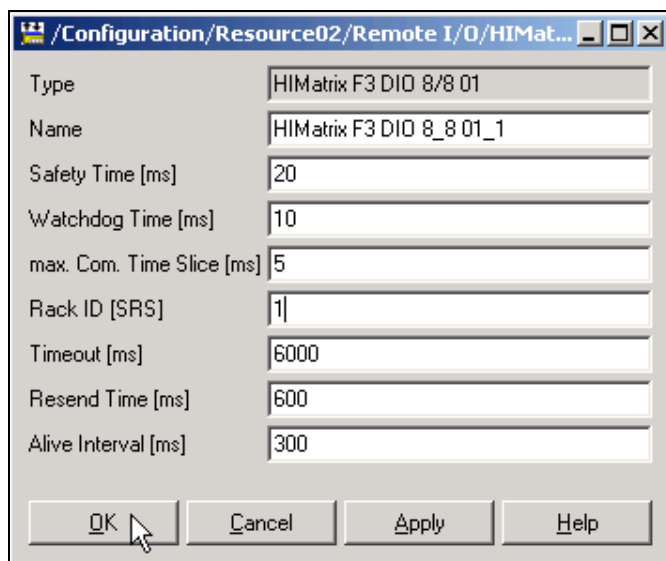


Fig. 88: Alterar ID de Rack

7.8 Sinais

7.8.1 Definição da diferença entre sinal e variável

Uma **variável** é um representante para um valor dentro da lógica do programa de aplicação. Pelo nome da variável, o local na memória com o valor armazenado é simbolicamente endereçado.

Sinais são utilizados para a troca de dados entre componentes individuais de um recurso (p.ex., programa de aplicação, canais de E/S) e para a troca de dados segura e não segura com outros recursos. O nome de sinal esconde todas as definições de atribuições para as transferências de dados definidas via Drag & Drop.

Se o valor de uma variável da instância de programa deve ser usado numa outra área, deve ser criado um sinal no editor de sinais do Hardware Management. Depois, o sinal é arrastado via Drag & Drop para a declaração de variáveis ou para a área de desenho do programa.

Neste momento, a variável do programa com o mesmo nome é transferida ao registro VAR_EXTERNAL ou é criada lá se ela ainda não existir. A seguir, o sinal é atribuído a um canal de entrada/saída, a um sinal de sistema ou a um parceiro de comunicação.

Se a variável vinculada a um sinal deve ser utilizada em vários blocos funcionais como VAR_EXTERNAL, ela deve ser definida da mesma forma também nestes locais.

Se for utilizar um bloco funcional já programado que contenha variáveis do tipo VAR_EXTERNAL que ainda não estão definidas no editor de sinais, isso deve ser feito agora e a utilização do sinal deve ser assinalada mediante Drag & Drop.

Nota: Idealmente, determina-se antes do início da programação quais sinais são necessários. Estes sinais são criados primeiramente no editor de sinais e depois são definidos no programa ou bloco funcional via Drag & Drop. Isso também vale para todas as variáveis que não recebem vinculação ao hardware, porém, cujos valores devem ser forçados durante a operação através do Force Editor.

7.8.2 Definir sinais

Passo 1: Abrir o editor de sinais:

- ☐ Clicar na barra de menu e, **Signals**.
- ☐ Clicar no menu **Signals** a função de menu **Editor**.

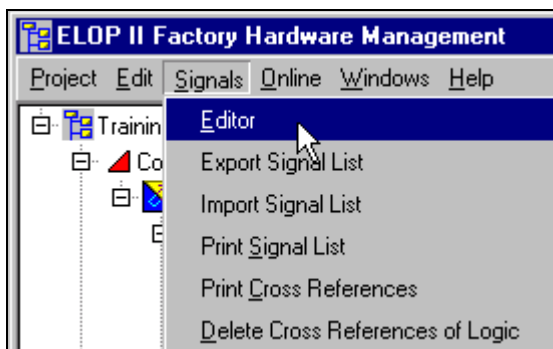


Fig. 89: Abrir o editor de sinais

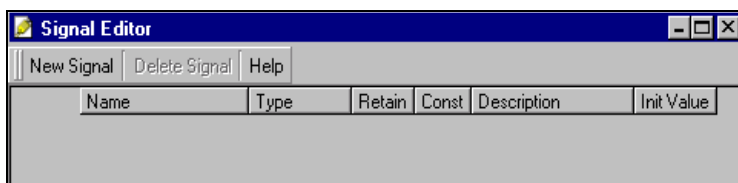


Fig. 90: Editor de sinais, ainda sem sinais

Passo 2:

Definir sinal:

- ☐ Clicar no botão **New Signal**.
Uma nova linha de edição se abre.
- ☐ Introduzir dados nos campos “Name” e “Type”. Estas introduções são obrigatórias.

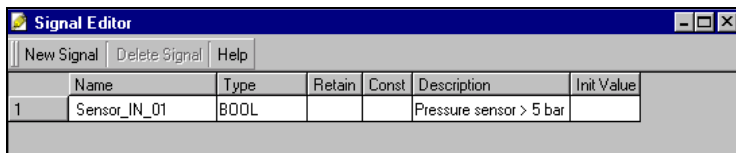


Fig. 91: Introdução de dados de sinais

- ☐ “Retain” ou “Constant” são colocados via clique duplo neste campo, ou resetados mediante novo clique duplo.
- ☐ Observar letras maiúsculas e minúsculas.
O editor de sinais não permite dois sinais com grafia idêntica.

Importante: Nunca ativar simultaneamente “Retain” e “Constant”. Isso irá causar mensagens de erro durante a geração de código.

O programa de aplicação deve ter como acessar sinais com o atributo “Retain” tanto para leitura quanto escrita. Porém, o acesso em modo de escrita a sinais com o atributo “Constant” não é possível.

Passo 3:

Sinalizar a utilização do sinal:

A utilização de sinais em blocos ou no programa de aplicação é definida via Drag & Drop - arrastar e soltar. Para este fim, proceder como segue:

- ☐ Caso houver outros aplicativos abertos, além do ELOP II Factory (p.ex., programa de e-mail), minimizar estes aplicativos.
- ☐ Clicar com o botão direito do mouse na barra de tarefas do Windows. Selecionar **Tile Windows Horizontally** no menu de contexto e alinhar gestão de projetos e Hardware Management um em cima do outro (veja Fig. 93).

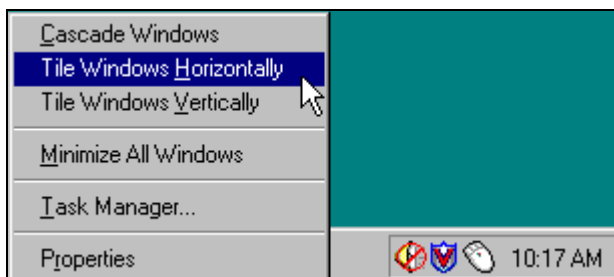


Fig. 92: Organizar janelas horizontalmente

- ☐ Aumentar a área de desenho do programa ou bloco funcional para ter o máximo de espaço para o posicionamento (Fig. 93, parte superior).
- ☐ Aumentar a área da lista de sinais no editor de sinais para ver o máximo de sinais (Fig. 93, parte inferior).

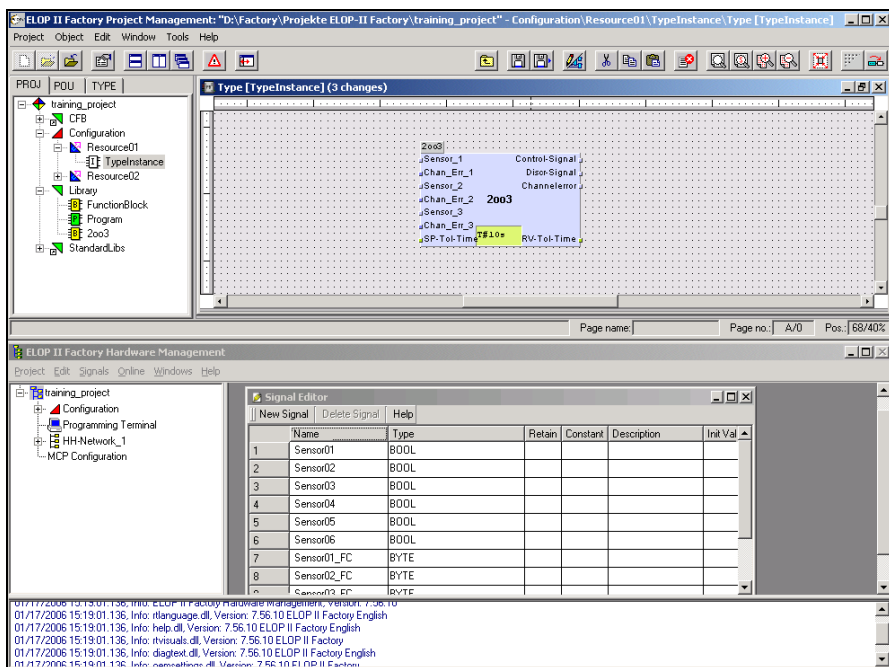


Fig. 93: Área de desenho e editor de sinais maximizados

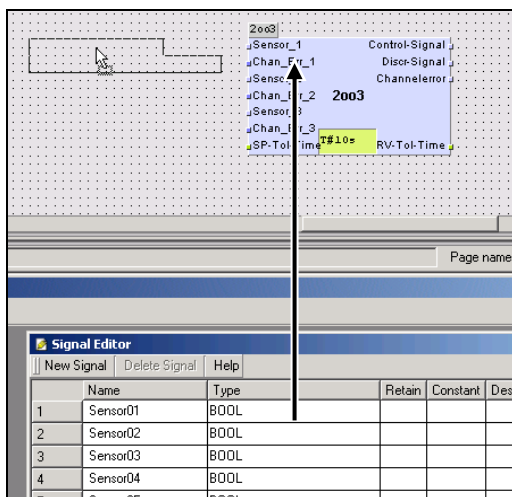


Fig. 94: Drag & Drop de sinais para a área de desenho

- ❑ Arrastar um sinal via Drag & Drop do editor de sinais para a área de desenho e posicionar o sinal no local desejado. O sinal é criado automaticamente no registro VAR_EXTERNAL (Fig. 94).

Nota: O sinal não pode estar no estado de editar. Se for o caso, encerrar a edição com ENTER.

*Nota: Se já existir uma variável com grafia idêntica do nome no registro VAR_EXTERNAL, a variável é atualizada com os novos dados.
Se já existir uma variável com o mesmo nome num outro registro sem ser o VAR_EXTERNAL, a variável é transferida ao registro VAR_EXTERNAL depois de uma confirmação e atualizada com os novos dados.*

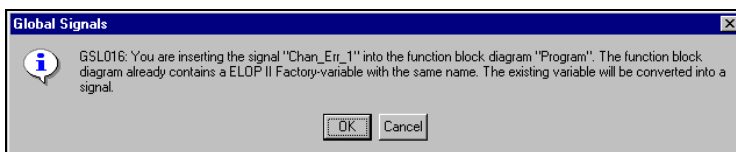


Fig. 95: Transformar uma variável em um sinal

7.8.3 Atribuir sinais aos canais de E/S do hardware

Passo 1: Abrir a janela para a atribuição de sinais:

- ☐ Abrir o editor de sinais (veja Capítulo 7.8.2).
- ☐ Clicar com o botão direito do mouse em um módulo de E/S e selecionar no menu de contexto **Connect Signals**.

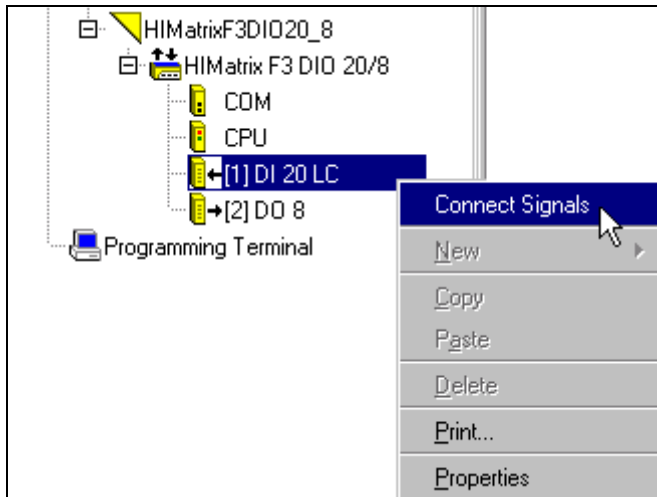


Fig. 96: Atribuir sinais aos canais de entrada/saída

Passo 2: Organizar janelas:

- ☐ Organizar o editor de sinais e a janela “Signal Connections” lado a lado verticalmente.

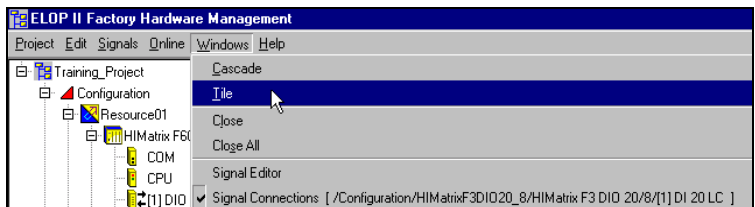


Fig. 97: Organizar janelas

Passo 3: Atribuir sinal:

- ❑ Clicar no editor de sinais em um sinal e arrastar o mesmo via Drag & Drop para a coluna “Signal” da atribuição de sinais.

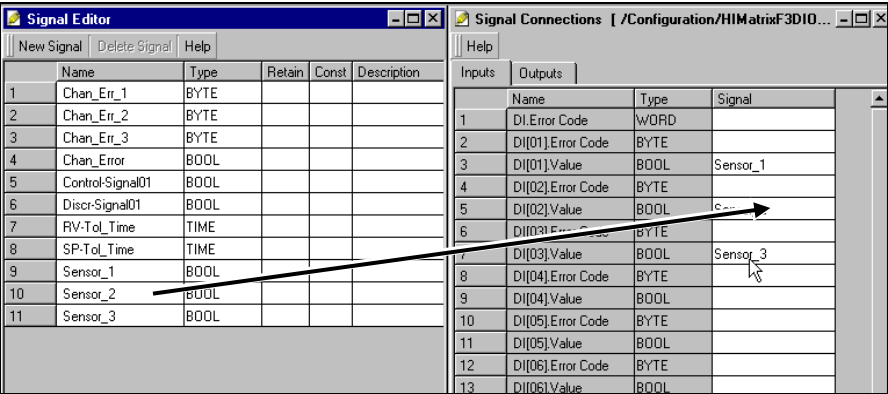


Fig. 98: Atribuir sinais

Nota: É possível selecionar vários sinais ao mesmo tempo e arrastar os mesmos simultaneamente para a janela “Signal Connections”. Para este fim, os sinais devem estar alinhados na sequência dos canais no editor de sinais. Em cada janela, é possível ordenar os elementos em ordem ascendente pelo nome, tipo de dados ou outros critérios, clicando no título da coluna.

Nota: Um módulo é representado no ELOP II Factory pelas entradas e saídas de sinais, mesmo se o módulo apenas possuir saídas físicas. Um módulo fornece sinais diagnósticos em forma de códigos de erro de módulos e códigos de erro de canais. A direção dos dados informa se está se tratando de uma entrada ou saída.

Códigos de erro para entradas e saídas físicas encontram-se no registro “Inputs”, pois trata-se de valores de entrada para o programa de aplicação.

Os parâmetros são atribuídos no registro “Outputs”, independente do fato de se tratar de parâmetros para entradas ou saídas físicas.

A este respeito, também veja “Signals and Error Codes...”, nas folhas de dados individuais.

	Módulo de entrada		Módulo de saída	
	Entrada	Saída	Entrada	Saída
Sinal de hardware do campo ou para o campo	X	-	-	X
Códigos de erro dos canais ou do módulo	X	-	X	-
Parametrização ou configuração dos canais	-	X	-	X

7.8.4 Atribuir sinais aos sinais de sistema

De forma semelhante como no capítulo anterior, o editor de sinais deve estar aberto e a atribuição de sinais da CPU deve ser aberta.

Para a atribuição em si, proceder como descrito no Capítulo 7.8.3.

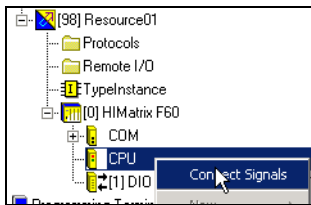


Fig. 99: Abrir a janela para a atribuição de sinais de sistema

Observar também a descrição detalhada dos sinais nos manuais dos sistemas de comando HiMatrix, com ênfase especial nos capítulos a respeito dos sinais de sistema. Especialmente as entradas alimentação com corrente, estado de temperatura e tempo de forcing remanescente devem ser monitoradas.

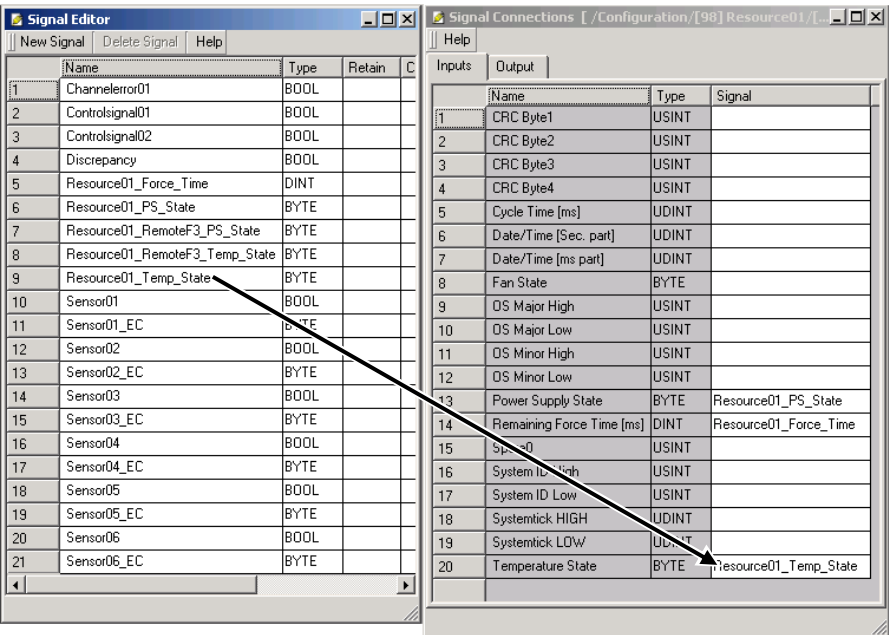


Fig. 100: Entradas de CPU

7.9 Comunicação com outros sistemas de comando HIMatrix

A comunicação com outros sistemas de comando em primeiro lugar serve para trocar sinais entre estes sistemas de comando.

7.9.1 Comunicação Peer-to-Peer (comunicação P2P)

Entende-se como comunicação Peer-to-Peer a comunicação entre dois nódulos dentro da mesma rede sem precisar de um master de comunicação para isso.

No momento, a comunicação Peer-to-Peer apenas é possível entre sistemas de comando da família HIMatrix. Idealmente, os sistemas de comando são criados como recursos no mesmo projeto. Se até agora no seu projeto de treino apenas tiver um recurso, repetir os procedimentos descritos nos Capítulos 7.5 a 7.8.

Para que dois ou mais sistemas de comando HIMatrix possam trocar sinais entre si, primeiramente deve ser criada uma rede e devem ser definidos os nódulos (participantes) na rede. Depois, devem ser definidos os parceiros de comunicação para cada nódulo. Isso ocorre no Editor Peer-to-Peer do respectivo recurso. A seguir, são determinados os sinais de comunicação e os sinais de supervisão para cada conexão Peer-to-Peer.

A configuração ocorre no Hardware Management.

Passo 1: Criar rede e grupo simbólico:

- ☐ Selecionar no menu de contexto do projeto **New, HH-Network**.

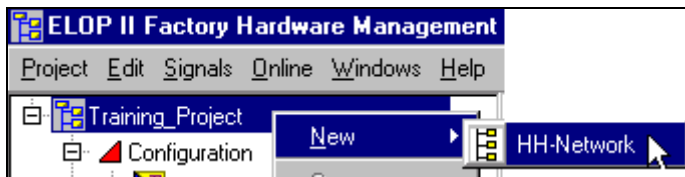


Fig. 101: Inserir rede HH

A nova rede criada é acrescentada à árvore do projeto e já contém um grupo simbólico.



Fig. 102: Grupo simbólico

- ☐ Caso necessário, alterar o nome no menu de contexto da rede, nas **Properties**.

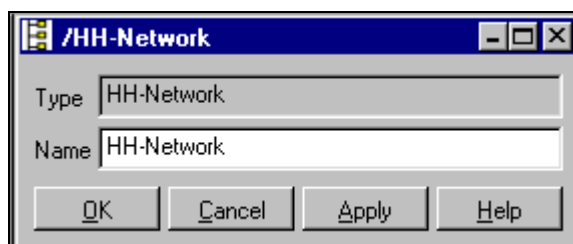


Fig. 103: Propriedades da rede

- ☐ NO menu de contexto do grupo simbólico, abrir **Properties**.
- ☐ Caso desejado, renomear o grupo simbólico no campo "Name".
- ☐ Ajustar o perfil "Fast" (ajuste padrão).

Type	Token Group
Name	Token Group
Profile	Fast
Token Group ID	4
Protocol Mode	Raw
Link Mode	TCS direct
Response Time [ms]	16
Bus Cycle Time [ms]	0
Token Alive Timeout [ms]	0
Primary Timeout [ms]	0
Secondary Interval [ms]	0
Link Mode (Extern)	TCS direct
Response Time (external) [ms]	16

Fig. 104: Propriedades do grupo simbólico

*Nota: "Fast" é o perfil padrão para redes nos quais se usam Switches.
 "Medium" é o perfil padrão para redes que utilizam Hubs.
 O ajuste "None" serve para a configuração de rede manual.
 Porém, devido ao grande número de parâmetros, a mesma é muito complexa e apenas deve ser executada por usuário experientes.
 Se na mesma rede houver recursos com perfis diferentes, os recursos devem ser atribuídos a diferentes grupos simbólicos.
 Cada grupo simbólico deve ter um ID individual.*

- Passo 2:** Definir os participantes de um grupo simbólico:
- Para definir os membros de um grupo simbólico, chamar o **Node Editor** no menu de contexto do grupo simbólico.

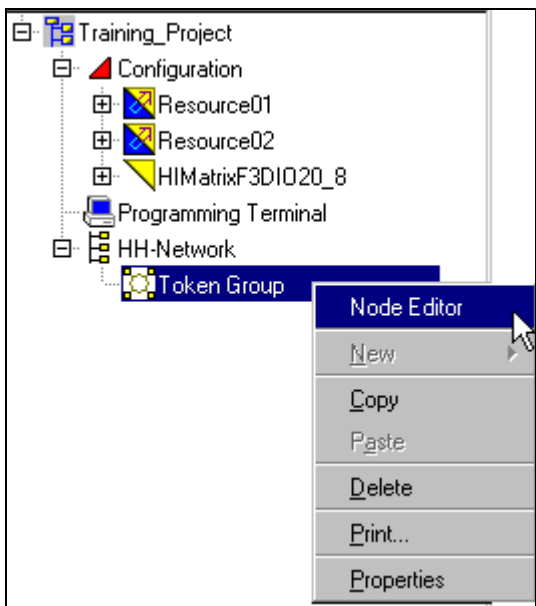


Fig. 105: Abrir o editor de nós

- Arrastar via Drag & Drop aqueles recursos da árvore de estrutura para o editor de nós que devem pertencer ao grupo simbólico selecionado.

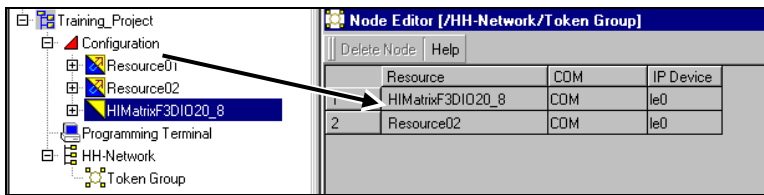


Fig. 106: Definir os participantes de um grupo simbólico

Passo 3:

Definir parceiros de comunicação para um recurso:

- ☐ Abrir o **Peer-to-Peer Editor** no menu de contexto de um recurso.
- ☐ Via Drag & Drop, arrastar os recursos com as quais o recurso deve se comunicar da árvore de estrutura para o editor Peer-to-Peer.

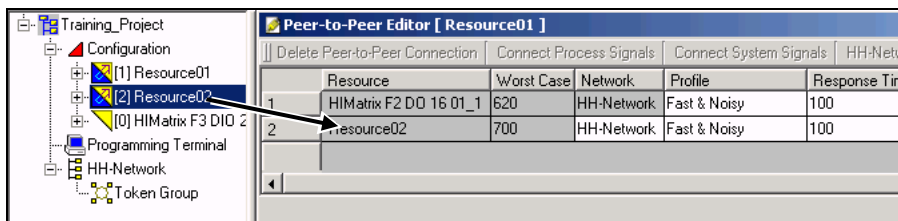


Fig. 107: Divulgar os parceiros de comunicação

- ☐ Se ainda não estiver presente, inserir o nome da rede na coluna “Network” (observar letras maiúsculas/minúsculas) ou arrastar o nome de rede via Drag & Drop da árvore de estrutura para a coluna “Network”.
- ☐ Na coluna “Profile”, ajustar o perfil desejado. O perfil pré-ajustado é “Fast & Noisy”.

Nota: Se estiver definindo uma conexão Peer-to-Peer para um recurso, o canal de retorno (Return Path) necessário para o parceiro de comunicação é criado automaticamente. Por isso, apenas precisa definir uma direção para um caminho de comunicação.

Nota: Uma explicação detalhada de todos os perfis encontra-se na ajuda online do Hardware Management, em “Peer-to-Peer Communication -> Peer-to-Peer Network Profiles”.

Passo 4:

Atribuir os sinais de processo para a comunicação Peer-to-Peer:

Apenas é possível atribuir sinais de processo para recursos. No caso de Remote I/Os do registro “Remote I/O”, através da atribuição dos sinais às entradas/saídas do hardware já são determinados os sinais de processo.

- ☐ Abrir o Editor Peer-to-Peer.
- ☐ Clicar no editor Peer-to-Peer no número de linha do lado esquerdo do nome de um dos parceiros de comunicação. Isso marca a linha e os botões na barra de botões se tornam ativos.

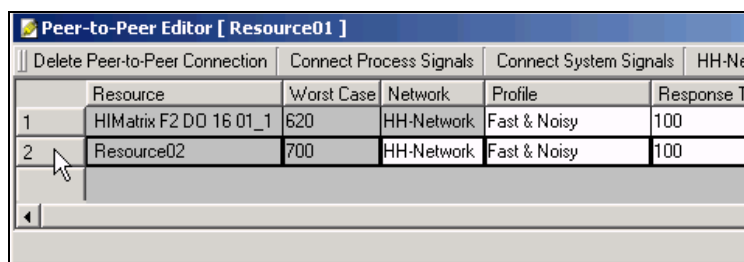


Fig. 108: Conectar sinais de processo

- ☐ Clicar na barra de botões em **Connect Process Signals**.
- ☐ Selecionar o registro para a direção da comunicação.

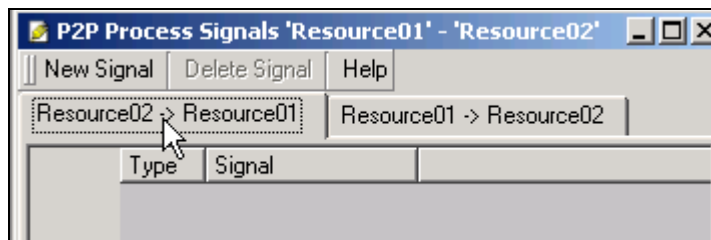


Fig. 109: Selecionar direção da comunicação

- ☐ Abrir o editor de sinais (Menu **Signals, Editor**) e colocar ao lado da janela “P2P Process Signals”.
- ☐ Posicionar o cursor num nome de sinal, no editor de sinais e arrastar o mesmo via Drag & Drop in “P2P Process Signals”.

Nota: É possível selecionar vários sinais ao mesmo tempo e arrastar os mesmos simultaneamente para a janela “Peer-to-Peer Process Signals”.

Observar a direção de transferência selecionada.

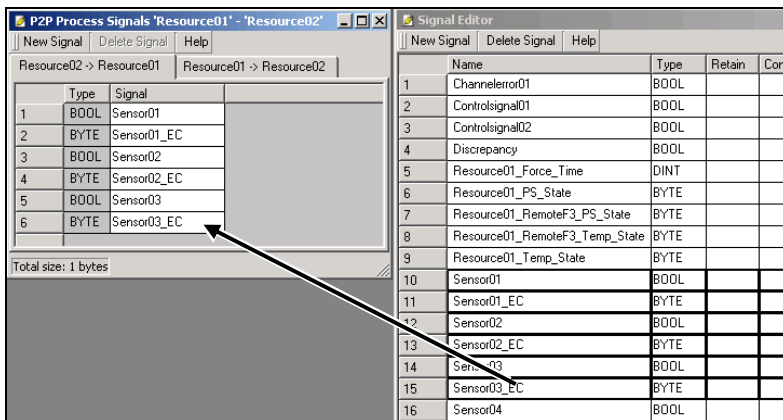


Fig. 110: Atribuir sinais de processo

Fig. 111 mostra os sinais que são enviados de “Resource01” para “Resource02”.

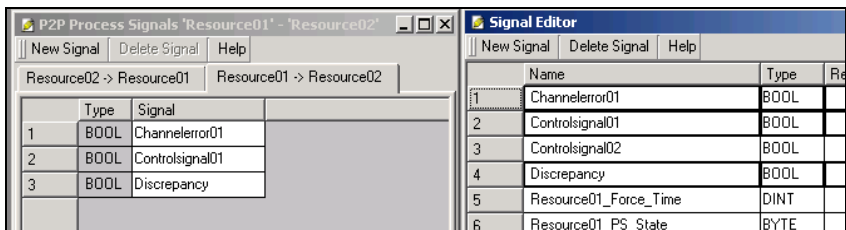


Fig. 111: Sinais de exportação do “Resource01”

Fig. 112 mostra os sinais que são enviados de “Resource02” para “Resource01”.

Type	Signal
1	BOOL Sensor01
2	BYTE Sensor01_EC
3	BOOL Sensor02
4	BYTE Sensor02_EC
5	BOOL Sensor03
6	BYTE Sensor03_EC

Total size: 1 bytes

Name	Type	Retain	Cons
1	Channelerror01	BOOL	
2	Controlsignal01	BOOL	
3	Controlsignal02	BOOL	
4	Discrepancy	BOOL	
5	Resource01_Force_Time	DINT	
6	Resource01_PS_State	BYTE	
7	Resource01_RemoteF3_PS_State	BYTE	
8	Resource01_RemoteF3_Temp_State	BYTE	
9	Resource01_Temp_State	BYTE	
10	Sensor01	BOOL	
11	Sensor01_EC	BYTE	
12	Sensor02	BOOL	
13	Sensor02_EC	BYTE	
14	Sensor03	BOOL	
15	Sensor03_EC	BYTE	
16	Sensor04	BOOL	

Fig. 112: Sinais de importação do “Resource01”

Passo 5: Atribuir os sinais de sistema para a comunicação Peer-to-Peer:

Diferente de sinais de processo, os sinais de sistema sempre são vistos a partir do recurso cujo editor Peer-to-Peer estiver aberto. São sinais de supervisão relativos ao status da comunicação ou do sistema de comando. Isso deve ser efetuado para todas as conexões, também as conexões às Remote I/Os.

- ☐ Clicar no editor Peer-to-Peer no número de linha do lado esquerdo do nome de um dos parceiros de comunicação. Isso marca a linha e os botões na barra de botões se tornam ativos.
- ☐ Clicar nos botões **Connect System Signals**.

Peer-to-Peer Editor [Resource01]					
Delete Peer-to-Peer Connection Connect Process Signals Connect System Signals HH-Netw					
	Resource	Worst Case	Network	Profile	Response
1	Remote F3 DIO 20_8 01_1	620	HH-Networl	Fast & noisy	100
2	Resource02	700	HH-Networl	Fast & Noisy	100

Fig. 113: Conectar sinais de sistema

	Name	Type	Signal
1	Connection State	UINT	Resource01_COM_Status_To_Resource02
2	Receive Timeout	UDINT	
3	Response Time	UDINT	
4	Version	WORD	

Fig. 114: Entradas dos sinais de sistema

Nota: A descrição exata dos sinais pode ser consultada nos manuais de sistema, no capítulo Sistema operacional, subcapítulo “Sinais de sistema de uma conexão Peer-to-Peer”.

7.10 Geração de código

Depois de finalizar a configuração e o programa de aplicação, a configuração de recurso é gerada que depois disso pode ser carregada ap sistema de comando.

- Independente do tipo de equipamento, cada sistema de comando precisa ser configurado.
- Remote I/Os não podem conter um programa de aplicação. Além da configuração, é necessária a configuração dos canais de entrada/saída. A definição dos canais de entrada/saída também faz parte da configuração do recurso.
- Recursos podem conter um programa de aplicação. O programa de aplicação também faz parte da configuração do recurso.

Passo 1: Executar a geração de código para o recurso:

a.) Para recursos:

- ☐ Vá à gestão de projetos (ALT + TAB).
- ☐ Abrir o menu de contexto do recurso e selecionar **Code Generation**.
As Remote I/Os atribuídas automaticamente são geradas também.
- ☐ Para iniciar a geração de código para todos os recursos de uma só vez, abrir o menu de contexto da configuração e selecionar **Code Generation**.

b.) Para HIMatrix F3 DIO 20/8 01:

- ☐ Abrir no Hardware Management o menu de contexto do módulo F3 DIO 20/8 01 e selecionar **Code Generation**.
- ☐ Para iniciar a geração de código para todas as F3 DIO 20/8 01 de uma só vez, abrir o menu de contexto da configuração e selecionar **Code Generation**.

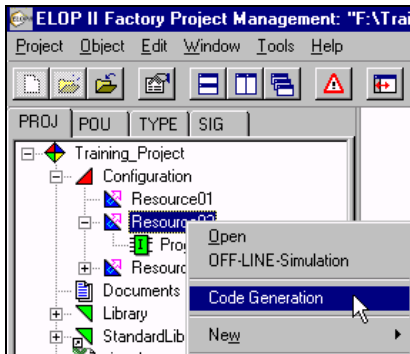


Fig. 115: Iniciar a geração de código

Observar também as mensagens nas indicações de erros/status da gestão de projetos e do Hardware Management.
Erros na configuração tanto quanto erros na lógica do programa de aplicação podem causar a interrupção da geração de código.

Error-state viewer		
Date/Time	Level	Text
18.01.2006, 13:54:33	Information	CRC Check of installation files ok
18.01.2006, 13:54:33	Information	CRC Check of used certified function blocks started
18.01.2006, 13:54:33	Information	No certified function blocks were used.
18.01.2006, 13:54:33	Information	POST-Compiler finished
18.01.2006, 13:54:33	Information	MCG004: Binary code generation finished. Applies to: <Configuration\Resource02>
18.01.2006, 13:54:33	Information	MCG018: Errors=0/Warnings=0
18.01.2006, 13:54:33	Information	MCG009: Code generation completed without errors. Applies to: <Configuration\Resource02>

01/18/2006 13:54:17.378, Info: ELOP II Factory Hardware Management, Version: 7.56.18
 01/18/2006 13:54:33.322, Info: [Resource02] Code generation finished with CRC: 16#e4e585fa
 01/18/2006 13:54:33.332, Info: [Resource02] Code generation finished. Warnings: 0, Errors: 0.

Fig. 116: Indicadores de erros/status após a geração de código

Nota: No caso de alertas, também observar as mensagens anteriores.

Passo 2: Iniciar o gerador de código uma segunda vez:

- ☐ Se a geração de código teve êxito, iniciar o gerador de código uma segunda vez como no passo 1.

Para aplicações direcionadas à segurança, o gerador de código deve ser iniciado duas vezes e as somas de verificação (CRCs) devem ser comparadas entre si. As versões do código devem ser idênticas. Isso evita erros que podem ser causados por um PC padrão não seguro. Mais detalhes encontram-se no Manual de segurança.

Passo 3: Comparar os dois valores de CRC:

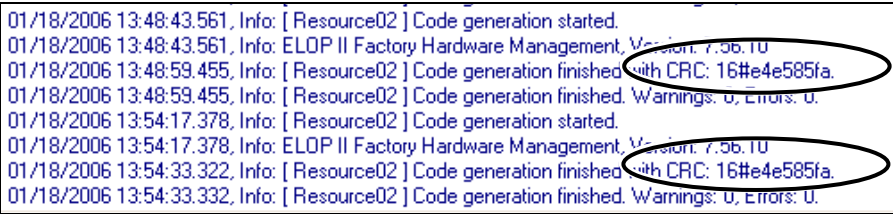


Fig. 117: Indicador de status e erros do Hardware Management com os dois valores de CRC

No caso de usar uma versão mais antiga do ELOP II Factory, o valor de CRC não é exibido no status de erro. O valor está disponível através da janela “About Configuration”.

- ☐ Selecionar no Hardware Management, no menu de contexto do recurso **About Configuration**.

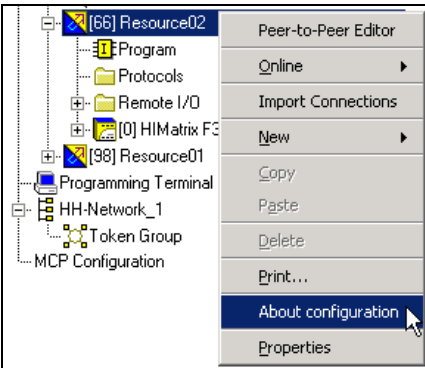
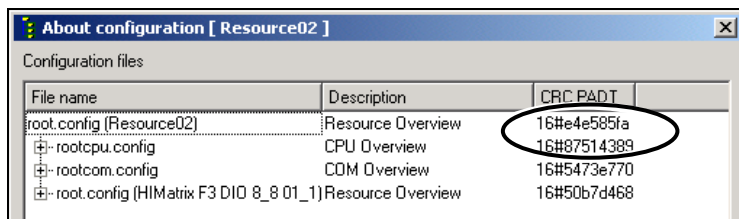


Fig. 118: Abrir as informações sobre a configuração

- ☐ Anotar a soma de verificação da coluna “CRC PADT” para “root.config” do resumo do recurso.
Ou salve a imagem da tela num arquivo de Word.



File name	Description	CRC.P&DT
root.config (Resource02)	Resource Overview	16#e4e585fa
[-] rootcpu.config	CPU Overview	16#87514389
[-] rootcom.config	COM Overview	16#5473e770
[-] root.config (HiMatrix F3 DIO 8_8 01_1)	Resource Overview	16#50b7d468

Fig. 119: Anotar as somas de verificação

- ☐ Efetuar a geração de código novamente, como descrito no **Passo1**.
- ☐ Abrir a janela “.About configuration”.
- ☐ Comparar a soma de verificação da segunda geração de código com a soma de verificação anteriormente anotada.

Atenção: Somente se as somas de verificação forem idênticas, o código pode ser carregado ap recurso (veja Capítulo 7.12).

7.11 Parametrizar o PC e os sistemas de comando

Até agora, todos os ajustes foram efetuados no projeto, p.ex., quais endereços IP os sistemas de comando devem ter.

Para que os sistemas de comando realmente trabalhem com estes ajustes, devem ser parametrizados de forma correspondente.

7.11.1 Parametrizar o aparelho de programação (PC) para a comunicação

Passo 1: Ajustar o endereço IP do aparelho de programação (PC):

- ☐ Clicar em **Start** e selecionar **Settings, Network and Dial up Connections, LAN**.
- ☐ Abrir as **Properties** para o protocolo TCP/IP.
- ☐ Marcar o campo de opção “Use the following IP Address” e introduzir no campo “IP Address” um endereço IP compatível com as sua rede no projeto.
- ☐ Introduzir uma máscara de sub-rede válida no campo “Subnet Mask”.

Nota: O endereço IP do aparelho de programação deve estar na mesma sub-rede que os endereços IP dos sistemas de comando. Caso contrário, uma entrada de roteamento para a sub-rede dos sistemas de comando é necessária no aparelho de programação.

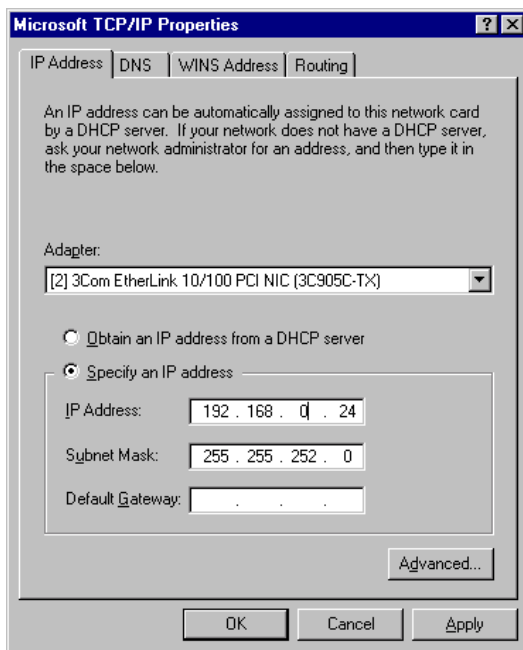


Fig. 120: Ajustar endereço IP do aparelho de programação

Nota: Windows 2000 e Windows XP por via de regra não precisam ser reinicializados para tornar o ajuste ativo.

7.11.2 Parametrizar o sistema de comando para a comunicação

A seguinte descrição vale para recursos e Remote I/Os da mesma forma.

Para um sistema de comando valem os seguintes ajustes efetuados em fábrica (se não especificado diferentemente):	
IP address	192.168.0.99
Subnet Mask	255.255.252.0
System.Rack ID for resources	60000.0
System.Rack ID for Remote I/Os	60000.1
User name	Administrator
Password	Sem senha

Passo 1: Ligar o sistema de comando a uma alimentação com tensão:

- ☐ Separar o sistema de comando de todas as conexões externas (Ethernet, canais de entrada/saída).
 - ☐ Ligar o sistema de comando a uma alimentação com corrente com dimensionamento suficiente.
 - ☐ Controlar se após a inicialização (cerca de 30 segundos, LED PROG piscando) o equipamento entrar em RUN (LED RUN acende) ou se permanecer em STOP (LED RUN piscando, no caso do F60, o LED STOP acende permanentemente).
- Se o sistema de comando estiver no estado STOP depois do boot, é possível continuar com o **Passo 2**.
 - Se o sistema de comando estiver no estado RUN depois do boot, o mesmo deve ser parado agora (veja Capítulo 7.11.3).

Passo 2:

Configurar endereço IP e máscara de sub-rede do sistema de comando:

- ☐ Ligar a placa de rede do aparelho de programação com uma conexão de Ethernet do sistema de comando.
- ☐ Alternar ao Hardware Management.
- ☐ Selecionar **Online**, **Communication Settings**, no menu de contexto do recurso ou da Remote I/O.

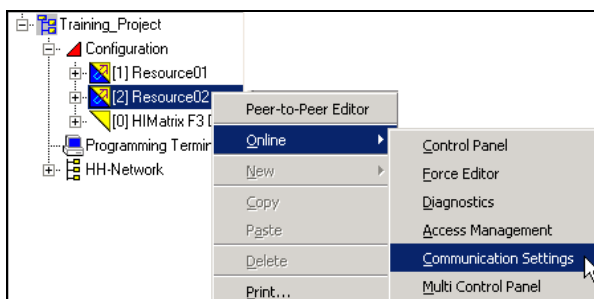


Fig. 121: Chamar parâmetros de conexão

Na janela “Communication Settings” são exibidos os dados ajustados no projeto:

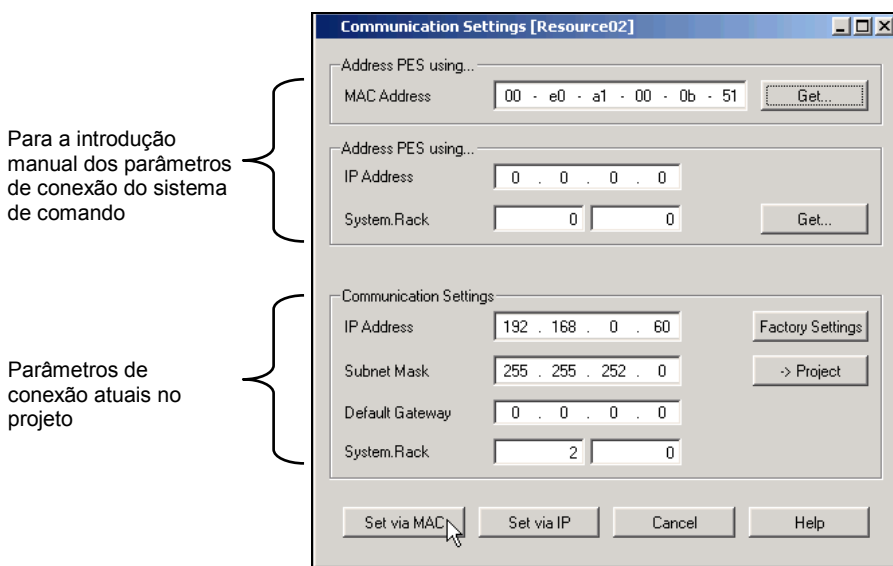


Fig. 122: Ajustar endereço IP e System.Rack ID via endereço MAC

- ☐ Introduzir o “endereço MAC”. O adesivo com o endereço MAC encontra-se na parte inferior do sistema de comando, perto das conexões de rede, ou, no caso do F60, diretamente no módulo CPU.
- ☐ Clicar em **Set via MAC**.

Nota: Se o endereço IP atual e o System.Rack ID do sistema de comando forem conhecidos, os novos parâmetros também podem ser ajustados mediante **Set via IP**.

- ☐ Introduzir para a autenticação um nome de usuário com direitos de administrador. O ajuste de fábrica é “Administrator”, sem senha.

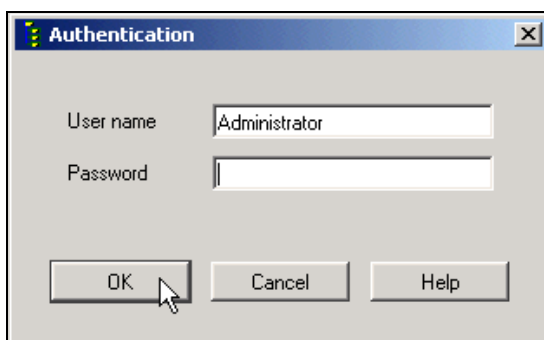


Fig. 123: Autenticação

Nota: Observar a mensagem no indicador de status e erros para o ajuste dos parâmetros com êxito.
Se o nome de usuário introduzido não for aceito, verificar a introdução correta.
Se o acesso com direitos de administrador não for conhecido, o sistema de comando deve ser resetado para os ajustes de fábrica. Veja a este respeito Capítulo 7.11.4.

Depois de ajustar os parâmetros de conexão, é possível carregar a sua aplicação ao(s) recurso(s) (veja Capítulo 7.12).

7.11.3 Para um sistema de comando com projeto desconhecido

Se um sistema de comando depois de inicializar entrar automaticamente no estado RUN, deve ser parado para configurar os dados de conexão.

Isso ocorre mediante um recurso vazio.

Passo 1: Criar o recurso vazio:

- ☐ Alternar ao Hardware Management.
- ☐ Selecionar no menu de contexto a configuração **New**, **HIMatrix F3 DIO 20/8 01**.

Nota: Para a configuração dos parâmetros de comunicação de um sistema de comando, o tipo de recurso não tem significado, ou seja, o tipo de recurso selecionado não precisa corresponder ao tipo de recurso real. Um número livre de sistemas de comando em série pode ser configurado com o mesmo objeto de recurso.

- ☐ Abrir com clique duplo em **HIMatrix F3 DIO 20/8 01** as **Properties** do recurso.
- ☐ Renomear o **HIMatrix F3 DIO 20/8 01** para **Dummy** é introduzir um System ID qualquer.
- ☐ Encerrar a edição com **OK**.

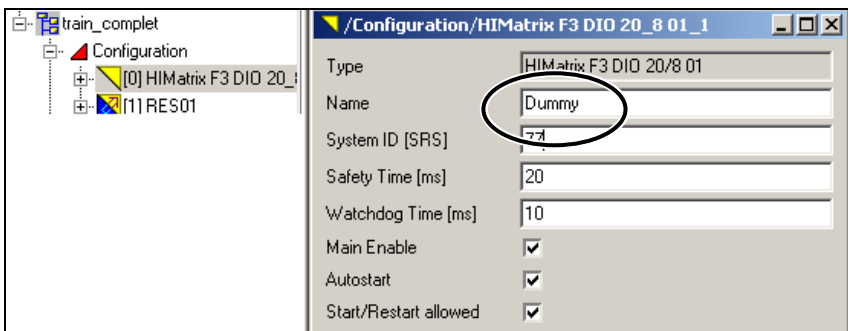


Fig. 124: Editar propriedades do recurso Dummy

Passo 2: Ligar o sistema de comando:

- ☐ Ligar a placa de rede do aparelho de programação com uma conexão de Ethernet do sistema de comando.

Passo 3: Ler os parâmetros de comunicação atuais do sistema de comando:

- ☐ Selecionar **Online, Communication Settings**, no menu de contexto do recurso Dummy.

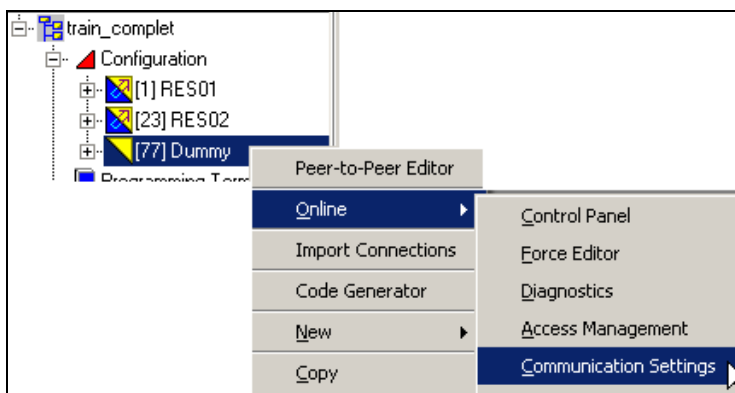


Fig. 125: Abrir a janela “Communication Settings”

- Introduzir o “endereço MAC”. O adesivo com o endereço MAC encontra-se na parte inferior do sistema de comando, perto das conexões de rede, ou, no caso do F60, diretamente no módulo CPU.

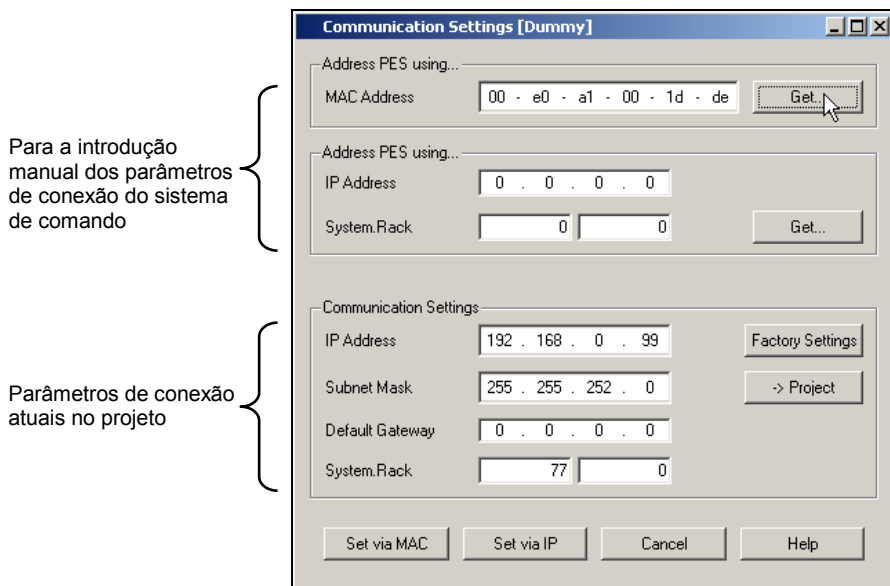


Fig. 126: Introduzir o endereço MAC

- ❑ Clicar no botão **Get...** ao lado do endereço MAC.
Os parâmetros de comunicação são exibidos nos campos do campo de grupo “Communication Settings”.

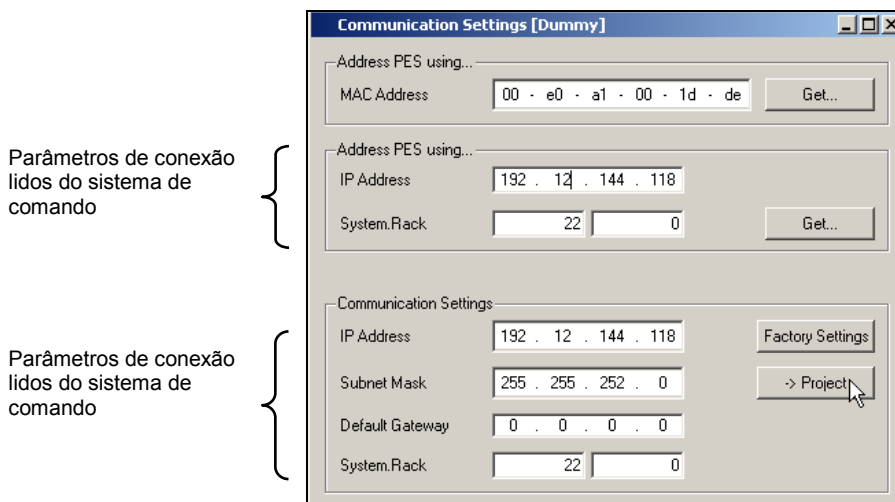


Fig. 127: Dados lidos do sistema de comando

- ❑ Clicar no botão **-> Project**.
Os parâmetros de conexão lidos do sistema de comando são transferidos para o recurso vazio (Dummy).

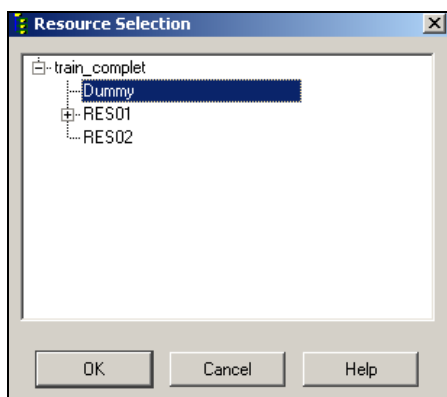


Fig. 128: Seleção do recurso no projeto

Passo 4: Parar o sistema de comando com ajuda do Control Panel:

Nota: Manter a janela dos parâmetros de conexão aberta.

- ☐ Selecionar **Online, Control Panel** no menu de contexto do recurso.

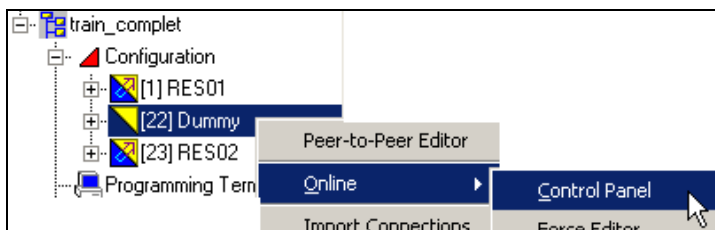


Fig. 129: Abrir o Control Panel do recurso Dummy

- ☐ Logar-se com o nome de usuário e modo de acesso “Administrator”.
Caso tenha conhecimento de um outro usuário configurado no sistema de comando com direitos de administrador, entre com estes dados.

Nota: Se o nome de usuário introduzido não for aceito, verificar a introdução correta.
Se o acesso com direitos de administrador não for conhecido, o sistema de comando deve ser resetado para os ajustes de fábrica. Veja a este respeito Capítulo 7.11.4.

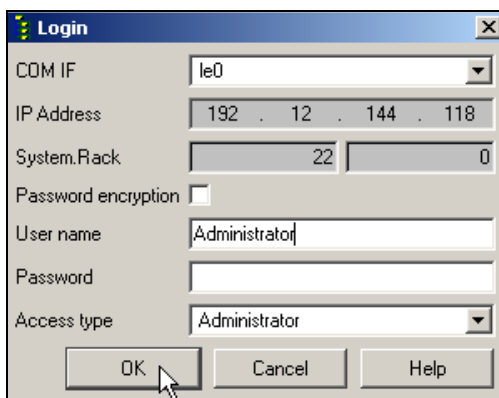


Fig. 130: Logar-se no Control Panel

- ☐ Clicar em **Stop** na barra de botões ou selecionar no menu **Resource, Stop**.
- ☐ Confirmar a solicitação de segurança com **Yes**.

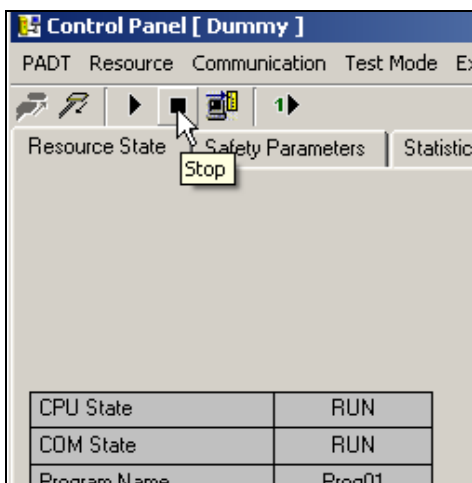


Fig. 131: Parar o sistema de comando

Passo 5:

Alterar ID de sistema e endereço IP de acordo com os dados do projeto:

- ☐ Fechar o Control Panel.
- ☐ Clicar no recurso cujos parâmetros de conexão gostaria de ajustar.
Se a janela **Communication Settings** ainda estiver aberta, os parâmetros de conexão do recurso selecionado são exibidos no campo de grupo na área inferior da janela.
- ☐ Para a configuração de endereço IP e System.Rack ID, continuar como descrito no Capítulo 7.11.2, a partir de Fig. 122.
- ☐ Excluir sem falta o recurso Dummy, se não for mais necessário.

7.11.4 Ativar os ajustes de fábrica

Nota: *É necessário resetar o sistema de comando aos ajustes de fábrica se o nome de usuário e a senha para o acesso como administrador não forem conhecidos.
Se apenas o endereço IP ajustado no momento no sistema de comando não for compatível com a sua rede, é possível estabelecer contato mesmo assim mediante uma entrada de "Route Add" no PC. Consultar o seu administrador de rede a este respeito.*

Passo 1: Acionar o botão de reset:

Atenção: *O botão de reset não pode receber forças elevadas. Portanto, apenas exercer pressão leve.*

Nota: *O botão de reset é acessível por um pequeno buraco na parte superior que se encontra aprox. 4–5 cm da borda esquerda. No caso do F60 e F20, há um pequeno orifício na parte frontal do equipamento.*

- ☐ Separar a alimentação com corrente ap sistema de comando.
- ☐ Acionar o botão de reset com uma caneta fina de material não eletrocondutor.
- ☐ Conectar o sistema de comando à alimentação com corrente mantendo o botão de reset pressionado por no mínimo 20 segundos.

Nota: *Depois da inicialização, o sistema de comando entra no estado STOP. Todos os parâmetros de conexão foram resetados para os ajustes de fábrica.
A partir da versão 10.42 do sistema operacional do módulo COM, depois da inicialização com o botão de reset acionado, a carga de um programa de aplicação é bloqueada!
Portanto, os parâmetros de conexão e a gestão de usuários no sistema de comando devem ser parametrizados seguido de novo boot normal.*

Passo 2:

Ajustar os parâmetros desconexão de acordo com os dados de projeto:

- ☐ Ligar a placa de rede do aparelho de programação com uma conexão de Ethernet do sistema de comando.
- ☐ Alternar ao Hardware Management.
- ☐ Selecionar **Online**, **Communication Settings**, no menu de contexto do recurso ou da Remote I/O.

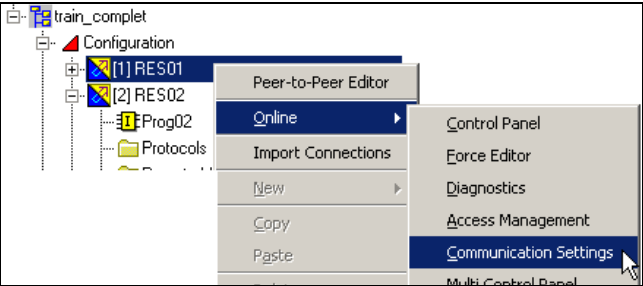


Fig. 132: Chamar parâmetros de conexão

Na janela “Communication Settings” são exibidos os dados ajustados no projeto.

Para a introdução manual dos parâmetros de conexão do sistema de comando

Parâmetros de conexão atuais no projeto

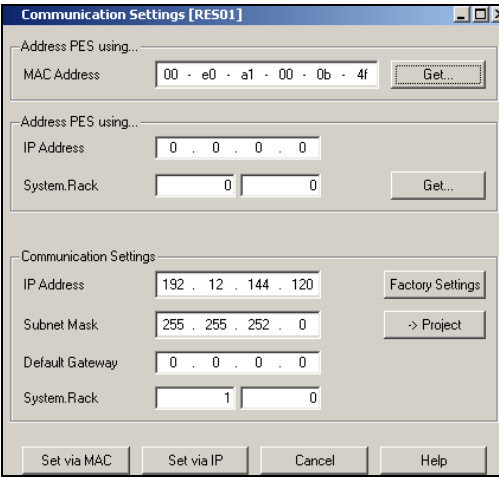


Fig. 133: Ajustar endereço IP e System.Rack ID via endereço MAC

- ☐ Introduzir o “endereço MAC”. O adesivo com o endereço MAC encontra-se na parte inferior do sistema de comando, perto das conexões de rede, ou, no caso do F60, diretamente no módulo CPU.
- ☐ Clicar em **Set via MAC**.
- ☐ Autenticar-se como Administrator, sem senha.

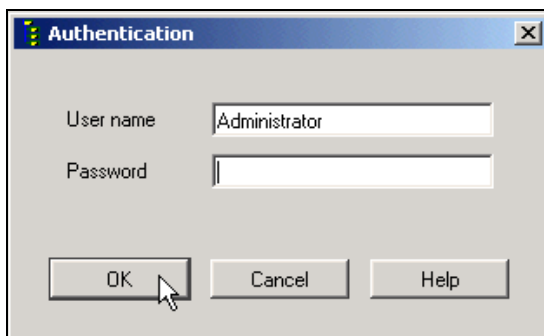


Fig. 134: Autenticação

Passo 3:

Configurar a gestão de usuários para o usuário padrão:

- ☐ Selecionar no menu de contexto do recurso **Online**, **Access Management**.

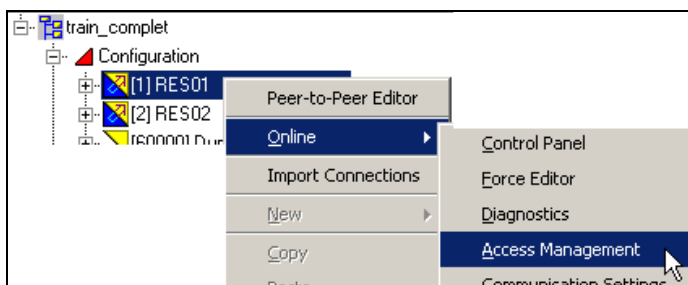



Fig. 135: Abrir a gestão de usuários

- ☐ Estabelecer a comunicação com o sistema de comando clicando no botão **Connect** .

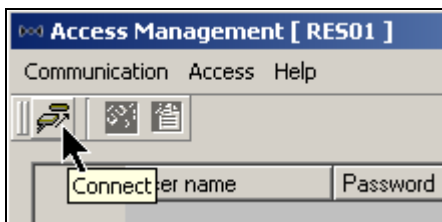



Fig. 136: Estabelecer a conexão ao sistema de comando

- ☐ Logar-se como Administrator, sem senha (tecla de atalho CTRL + a).
- ☐ Clicar no botão **Default account**  para resetar a gestão de usuários ao ajuste padrão.

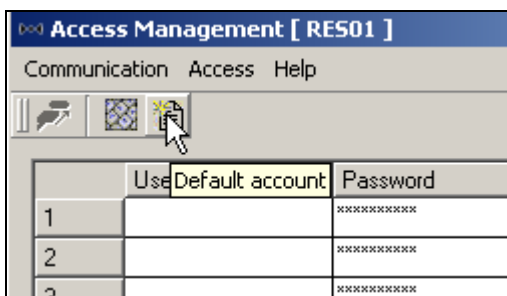


Fig. 137: Ativar ajuste padrão

Passo 4: Dar um novo boot no sistema de comando:

- ☐ Desligar a alimentação com tensão do sistema de comando e ligar novamente logo depois.

Agora é possível carregar a sua aplicação ao(s) recurso(s) (veja Capítulo 7.12).

7.12 Carregar e iniciar programa (configuração de recurso)

Antes de poder carregar uma configuração de recurso a um recurso, a geração de código para o recurso (veja Capítulo 7.10) deve ter sido executada e o aparelho de programação e o recurso devem ter parâmetros de conexão válidos (Capítulo 7.11).

Passo 1: Abrir o Control Panel:

- ☐ Selecionar **Online**, **Control Panel** no menu de contexto do recurso.

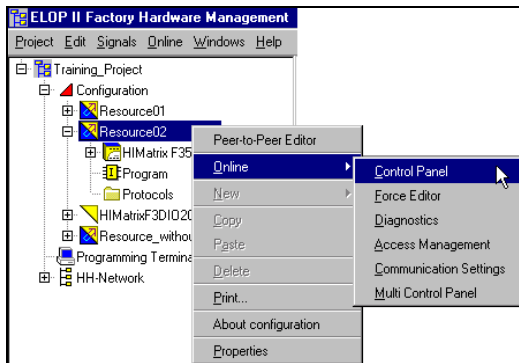


Fig. 138: Abrir o Control Panel

- ☐ Logar-se como administrador com o modo de acesso “Administrator” (tecla de atalho: CTRL + A).

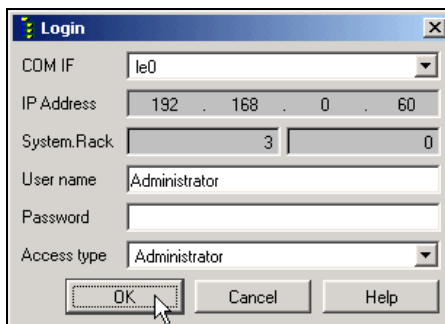


Fig. 139: Introduzir o nome de usuário e o modo de acesso

Passo 2: Carregar a configuração de recurso:

- ☐ O sistema de comando deve estar no estado “STOP”. Se for o caso, executar a função de menu **Resource, Stop**.

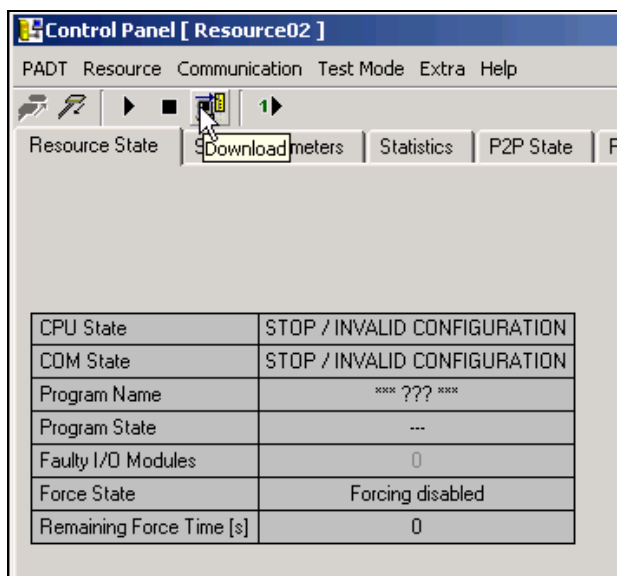


Fig. 140: Control Panel

- ☐ Clicar no botão **Download** .
Aparece uma confirmação de segurança (veja Fig. 141).

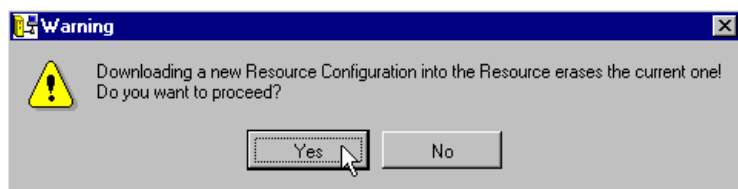


Fig. 141: Solicitação de confirmação antes de carregar

O processo de carregar inicia logo que a confirmação de segurança foi respondida com **Yes**.

Passo 3: Iniciar o programa:

☐ Clicar no botão **Coldstart** .

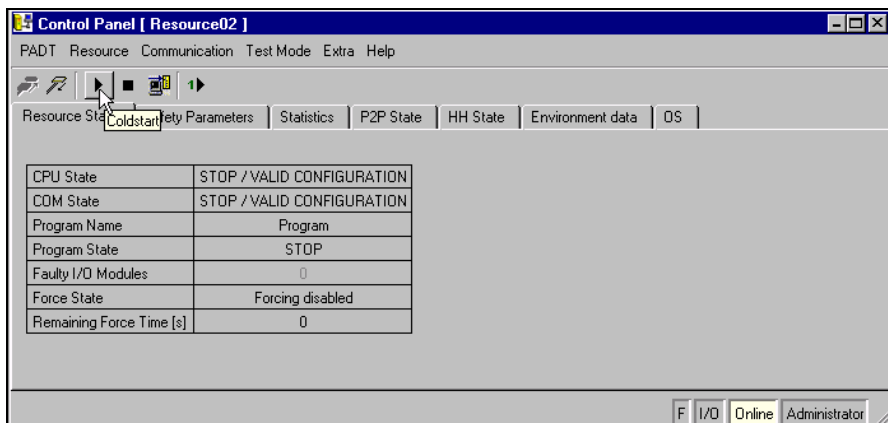


Fig. 142: Recurso no estado STOP

Após arranque a frio, os valores de “CPU State”, “COM State” e “Program State” mudam para RUN.

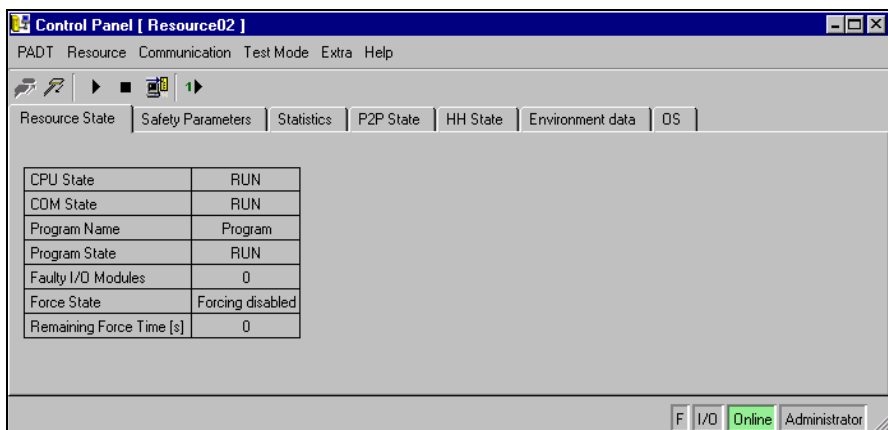


Fig. 143: Recurso no estado RUN

Nota: As funções **Start**, **Stop** e **Download** também podem ser executadas pelo menu **Resource**.

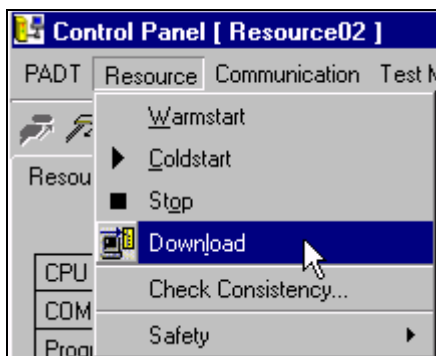


Fig. 144: Menu Resource

Um programa de aplicação carregado ao sistema de comando não pode ser lido de volta ao aparelho de programação!

Por isso, é muito importante arquivar os dados. Veja a este respeito Capítulo 7.16.

7.13 O Force Editor

Entende-se como Forcing - forçar o processo de manualmente sobrescrever sinais com valores que não resultam do processo mas que são definidos pelo usuário enquanto o sistema de comando está processando o programa de aplicação.

Em primeiro lugar, o Forcing serve para testar o programa de aplicação para casos que não ocorrem na operação normal ou apenas raras vezes, portanto apenas podem ser testados até certo ponto. Além disso, o Forcing é utilizado na localização geral de erros e durante trabalhos de serviço e reparação.

Aviso!

No início do processamento da lógica, o valor de Forcing da do sinal forçado é transferido à variável externa correspondente da lógica. Se durante o processamento da lógica ocorrer um acesso de escrita a essa variável, o novo valor escrito se torna efetivo dentro da lógica para acessos posteriores de leitura a essa variável. Fora da área da lógica, p.ex., no acesso da comunicação a outros sistemas ou nas saídas de hardware, o valor forçado do sinal de qualquer forma será o valor válido.

Passo 1:

Abrir o Force Editor:

- ☐ Clicar no menu de contexto de um recurso em **Online**, **Force Editor**.

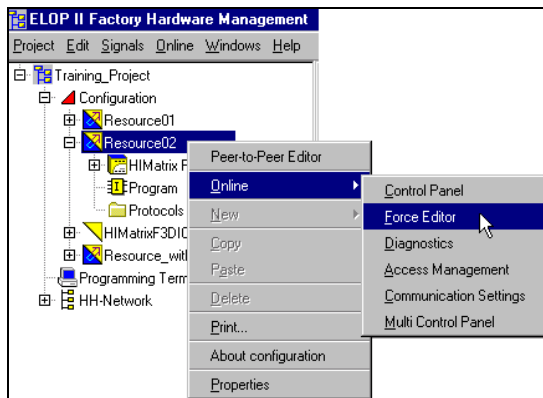


Fig. 145: Abrir o Force Editor

- ☐ Se ainda não foi estabelecida a comunicação entre o recurso e o aparelho de programação, é necessário se logar. Se o Force Editor abrir OFFLINE, é possível ativar o login pela função de menu **Resource, Forcing Online**.

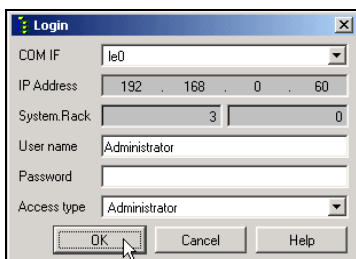


Fig. 146: Diálogo de login

Passo 2: Resetar valores forçados antigos:

Nota: Apenas executar o passo seguinte se o Forcing não estiver ativo. Caso contrário, proceder como descrito no Capítulo 7.13.2.

- ☐ Selecionar a função de menu **Resource, Clear Force Values on Resource**.

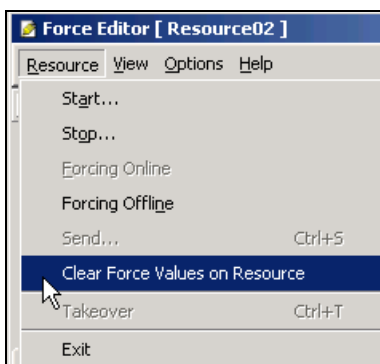


Fig. 147: Resetar valores de Forcing

Passo 3:

Adaptar a visualização de sinais no Force Editor:

- ☐ Com **Configure** abrir a janela “Select signals to view” (inicialmente, não há sinais no Force Editor).

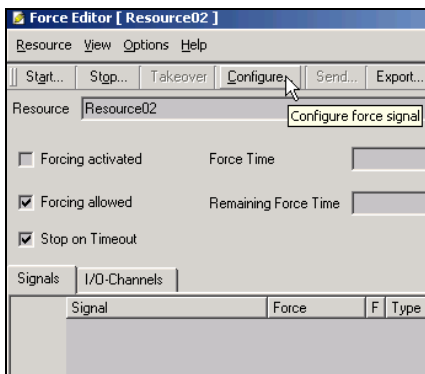


Fig. 148: Force Editor

Na seleção de sinais são marcados os sinais que devem ser exibidos ou forçados no Force Editor.

- ☐ Clicar nas caixinhas de seleção ou em **Select all**. Finalizar com **OK**.

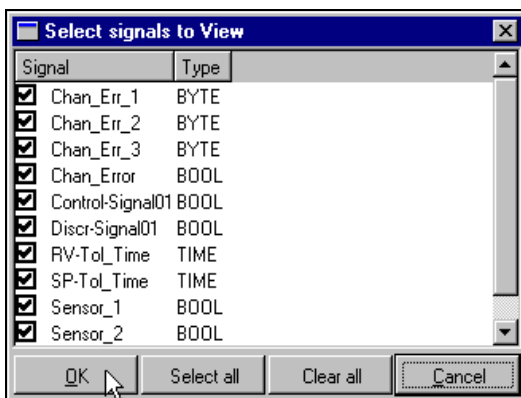


Fig. 149: Seleção de sinais

Passo 4: Preparar o Forcing:

- ☐ Introduzir na coluna “Force” o valor desejado de Forcing.
- ☐ Na coluna “F” é indicado se o sinal deve ser forçado (clique duplo).

	Signal	Force	F	Type	R-Value	R-Force	RF
1	Chan_Err_1	16#00		BYTE	16#00	16#00	
2	Chan_Err_2	16#00		BYTE	16#00	16#00	
3	Chan_Err_3	16#00		BYTE	16#00	16#00	
4	Chan_Error	FALSE		BOOL	FALSE	FALSE	
5	Control-Signal01	FALSE		BOOL	FALSE	FALSE	
6	Discr-Signal01	FALSE		BOOL	FALSE	FALSE	
7	RV-Tol_Time	T#0ms		TIME	T#0ms	T#0ms	
8	SP-Tol_Time	T#10s	✓	TIME	T#15s	T#0ms	
9	Sensor_1	1	✓	BOOL	FALSE	FALSE	
10	Sensor_2	FALSE		BOOL	FALSE	FALSE	
11	Sensor_3	FALSE		BOOL	FALSE	FALSE	

Fig. 150: Lista de sinais no Force Editor

Nota: Valores booleanos podem ser indicados com TRUE e FALSE ou com “1” e “0”.
 Observar nos sinais de tempo a unidade.

- ☐ Os valores de Forcing e a seleção dos sinais a serem forçados são transferidos ao sistema de comando com o botão **Send** - enviar.

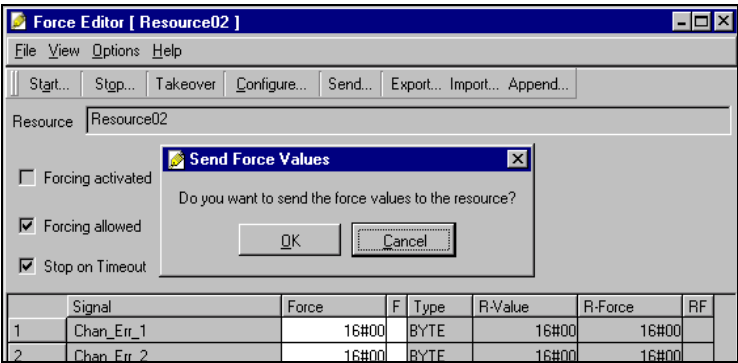


Fig. 151: Enviar os valores a serem forçados

Visão geral dos sinais no Force Editor depois do envio. O Forcing ainda não está ativado!

☐ Forcing activated Force Time sec.

☒ Forcing allowed Remaining Force Time sec.

☒ Stop on Timeout

	Signal	Force	F	Type	R-Value	R-Force	RF
1	Chan_Err_1	16#00		BYTE	16#00	16#00	
2	Chan_Err_2	16#00		BYTE	16#00	16#00	
3	Chan_Err_3	16#00		BYTE	16#00	16#00	
4	Chan_Error	FALSE		BOOL	FALSE	FALSE	
5	Control-Signal01	FALSE		BOOL	FALSE	FALSE	
6	Discr-Signal01	FALSE		BOOL	FALSE	FALSE	
7	RV-Tol_Time	T#0ms		TIME	T#0ms	T#0ms	
8	SP-Tol_Time	T#10s	<input checked="" type="checkbox"/>	TIME	T#15s	T#10s	<input checked="" type="checkbox"/>
9	Sensor_1	1	<input checked="" type="checkbox"/>	BOOL	FALSE	TRUE	<input checked="" type="checkbox"/>
10	Sensor_2	FALSE		BOOL	FALSE	FALSE	
11	Sensor_3	FALSE		BOOL	FALSE	FALSE	

Online Administrator

Fig. 152: Lista dos sinais a serem forçados

- A coluna “R-Value” contém o valor do sinal como resulta do processo ou da lógica.
- A coluna “R-Force” contém o valor de Forcing que substitui o valor R-Value durante o Forcing.
- Se o valor de Forcing se torna ativo ao iniciar o Forcing ou não depende do fato de ter um ganchinho na coluna “RF”.

Passo 5:

Iniciar Forcing:

- ☐ Ao acionar o botão **Start**, aparece uma solicitação do “Force Time” em segundos.

Start forcing

Force Time sec.

Attention: When you start forcing, all Force Values configured on the Resource become effective (not only those displayed). If you are not sure whether other settings have remained on the Resource besides yours, you can clear all Force Values through a command from the Resource menu.

Start Cancel

Fig. 153: Indicação do tempo de forcing

Nota: Depois de esgotar o tempo de forcing, o valor de Forcing é novamente substituído pelo valor R-Value.
Se nas propriedades do recurso foi ativado “Stop on Timeout”, o sistema de comando entra no estado de STOP depois do processo de Forcing terminar.

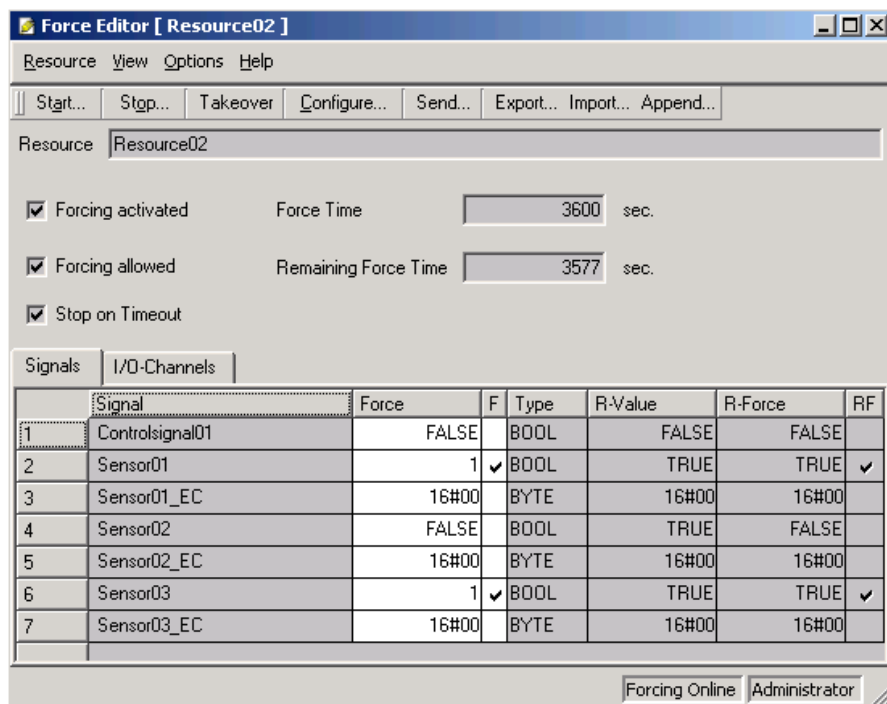


Fig. 154: Forcing ativado

- ☐ Para parar o Forcing manualmente, clicar no botão **Stop**. Neste caso o sistema de comando permanece no estado RUN, pois o Timeout não foi alcançado e os ajustes de Forcing continuam ativos.
- ☐ Para ter certeza que todos os valores de Forcing foram resetados, executar a função **Clear Force Values on Resource**, no menu “Resource”, antes de sair do Force Editor.

7.13.1 Salvar a seleção de sinais e carregar

Para não perder a visão geral sobre uma multidão de sinais, uma seleção de sinais pode ser gravada e carregada novamente em um momento posterior.

Passo 1: Preparar a seleção:

- ☐ Abrir o Force Editor e selecionar os sinais que devem ser gravados com **Configure**.

Passo 2: Salvar a seleção:

- ☐ Salvar a seleção com ajuda do botão **Export**, com nome desejado e a extensão “.fdi”.

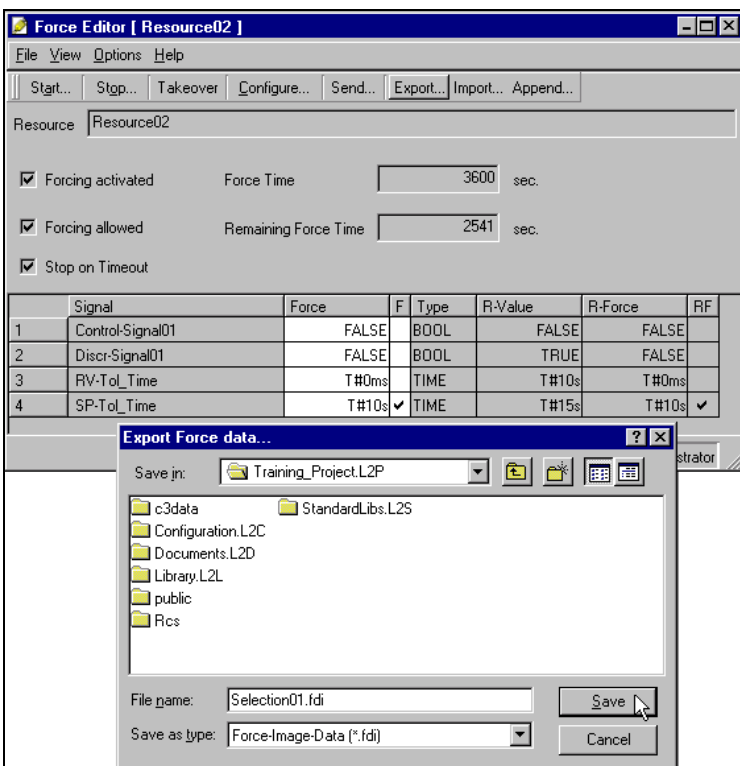


Fig. 155: Salvar a seleção de sinais (exportar)

- Passo 3:
- Importar seleção de sinais salv

Importar:

Abrir com ajuda do botão **Import** a janela de diálogo “Import Force data...”.

Selecionar o arquivo desejado e clicar em **Open** ou clique duplo diretamente no arquivo.

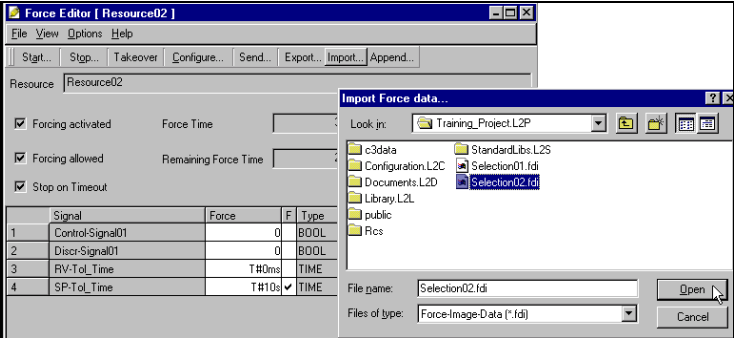


Fig. 156: Carregar a seleção de sinais (importar)

Nota: Os sinais atuais são substituídos pela seleção carregada.

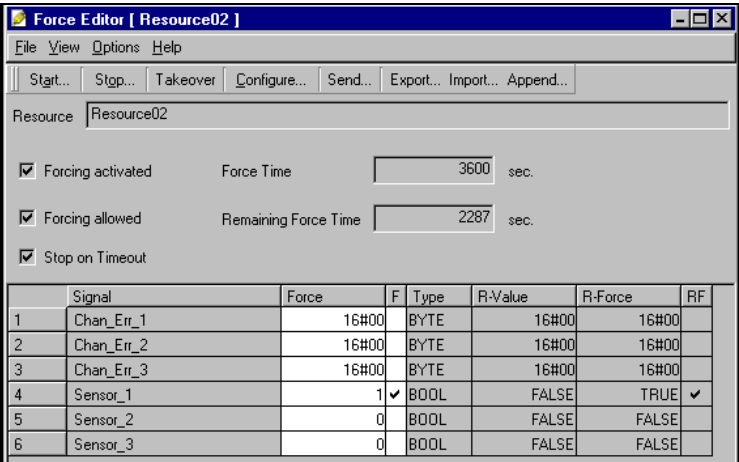


Fig. 157: Force Editor depois de carregar a seleção de sinais

Passo 4:

Anexar a seleção de sinais gravada:

- ☐ Abrir com ajuda do botão **Append** a janela de diálogo “Append Force data...”.
- ☐ Selecionar o arquivo desejado e clicar em **Open** ou clique duplo diretamente no arquivo.

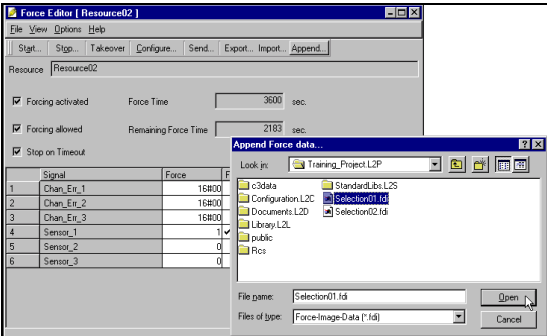


Fig. 158: Anexar seleção de sinais

Nota:

Os sinais atuais permanecem no Force Editor e os novos sinais carregados são anexados à lista de sinais.

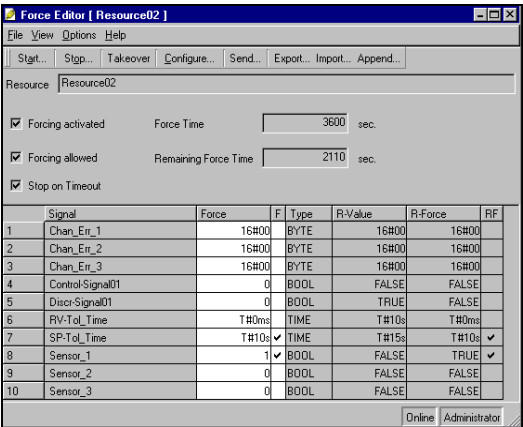


Fig. 159: Force Editor depois de anexar a seleção de sinais

Nota:

Para sistemas de comando com sistema operacional ≥ 4 , é possível exibir com o Force Editor também canais de E/S que não estão vinculados a um sinal.

7.13.2 Forçar um sistema já forçado

Se entrar num sistema para forçar e o mesmo já está sendo forçado, deveria gravar a situação atual de Forcing.

Passo 1: Abrir o Force Editor:

- Abrir o Force Editor conforme descrito no Capítulo 7.13.

Passo 2: Gravar a situação atual de Forcing:

- Marcar todos os sinais com **Configure**, **Select all**.
- Clicar no botão **Takeover**, para carregar os ajustes de Forcing ao PC.

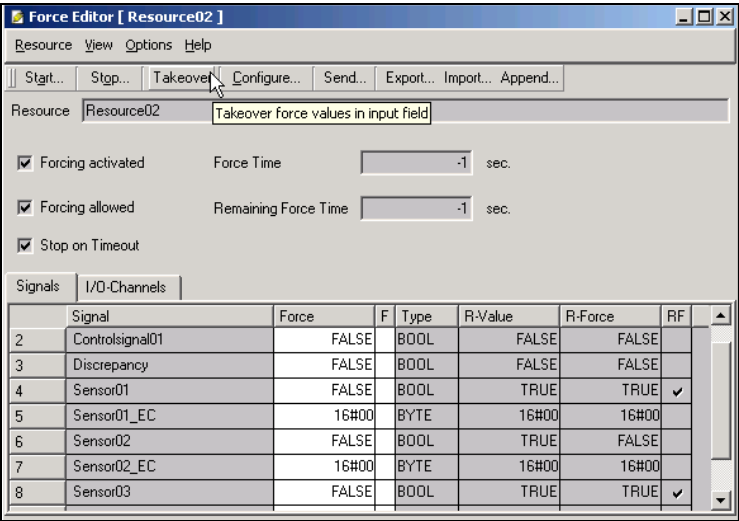


Fig. 160: Carregar os ajustes de Forcing no recurso ao PC

	Variable	Value	Force	Type	Value	Force
4	Sensor01	TRUE	✓	BOOL	TRUE	TRUE ✓
5	Sensor01_EC	16#00		BYTE	16#00	16#00
6	Sensor02	FALSE		BOOL	TRUE	FALSE
7	Sensor02_EC	16#00		BYTE	16#00	16#00
8	Sensor03	TRUE	✓	BOOL	TRUE	TRUE ✓
9	Sensor03_EC	16#00		BYTE	16#00	16#00

Fig. 161: Valores transferidos

- ☐ Selecionar **Export** e salvar estes ajustes (Fig. 155, p.145).
- ☐ Forçar os novos sinais a serem forçados como descrito no Capítulo 7.13, sem resetar os ajustes de Forcing no recurso.
- ☐ Antes de sair do Force Editor, carregar novamente a situação gravada com **Import** e enviar ao sistema de comando. (Fig. 156, p.146)

7.14 ONLINE Test (Power Flow)

O ONLINE Test na gestão de projetos serve para rastrear durante a operação os valores das variáveis e dos sinais utilizados na lógica.

O teste online apenas é possível se para todas as alterações na lógica foi gerado código e carregado ao sistema de comando e se o sistema de comando estiver no estado RUN. O Control Panel do recurso deve estar aberto e deve ter conexão ao sistema de comando. O sistema de comando deve ter um sistema operacional a partir da versão 4.0.

Passo 1: Abrir o teste online:

- ☐ Expandir na gestão de projetos a “Configuration”.
- ☐ Abrir o menu de contexto do recurso e clicar em **ONLINE Test**.

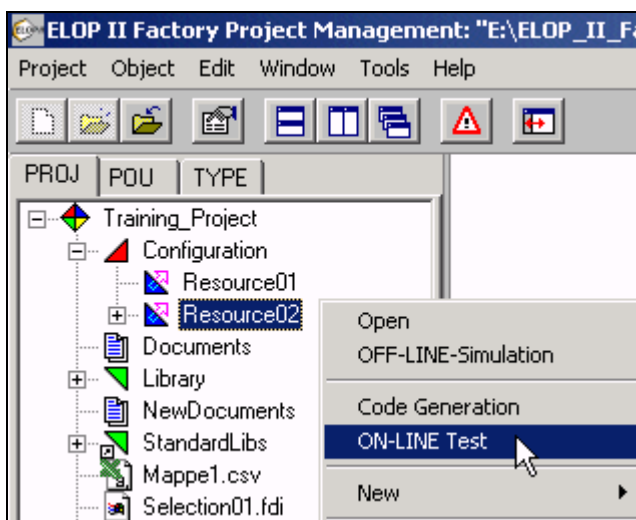


Fig. 162: Selecionar ONLINE Test

- ❑ Abrir a instância de programa indicada no teste online mediante clique duplo.

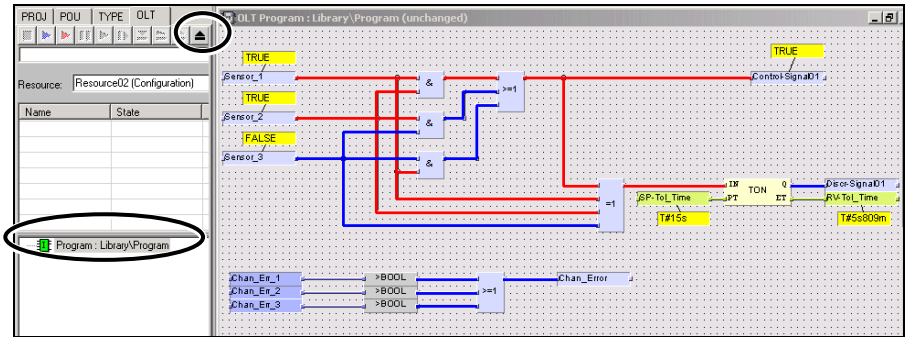


Fig. 163: ONLINE Test

Nota: Os valores são exibidos nos campos de teste online (campo OLT). Nos valores booleanos, a linha de conexão é marcada em cores (azul = FALSE, vermelho = TRUE). Campos OLT também podem ser completados durante o teste online e ou sair o teste online podem ser salvos com o teste.

- ❑ Criar os campos OLT com um clique do botão direito do mouse sobre os elementos e selecionar **Create OLT Field**.
- ❑ Pela seta preta para cima, é possível fechar o teste online.

7.15 Documentação

7.15.1 Documentação do software

Num objeto de documentação da gestão de projetos, é possível criar e organizar a impressão da lógica. Com o objeto de documentação, mais tarde todos as POU's (blocos) são impressas. A documentação do hardware ocorre de forma separada no Hardware Management.

Passo 1: Criar um novo objeto de documentação:

- ☐ Clicar com o botão direito do mouse no "Project".
- ☐ Selecionar no menu de contexto **New, Documents**.

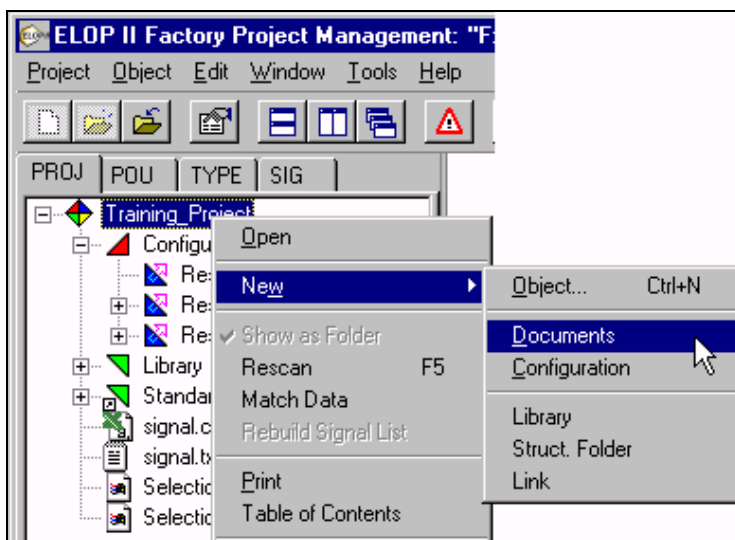


Fig. 164: Criar um objeto de documentação

Passo 2:

Alterar o nome da nova documentação:

- ☐ Clicar com o botão esquerdo do mouse duas vezes lentamente sobre o nome. Alterar o nome no campo de introdução de dados.

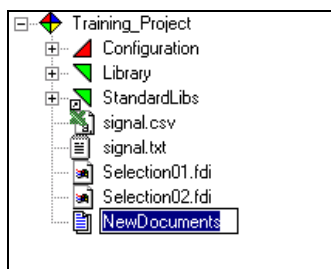


Fig. 165: Renomear objeto de documento

Passo 3:

Inserir os dados do projeto na documentação:

- ☐ Abrir a documentação mediante clique duplo.
- ☐ Completar a lista de sinais do Hardware Management.

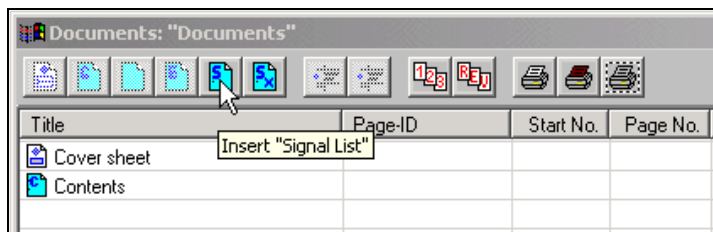


Fig. 166: Inserir a lista de sinais na documentação da lógica

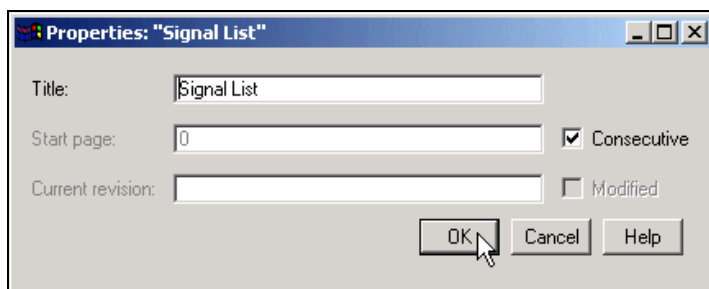


Fig. 167: Alterar os nomes da lista de sinais

- Completar a lista referências cruzadas de sinais do Hardware Management.

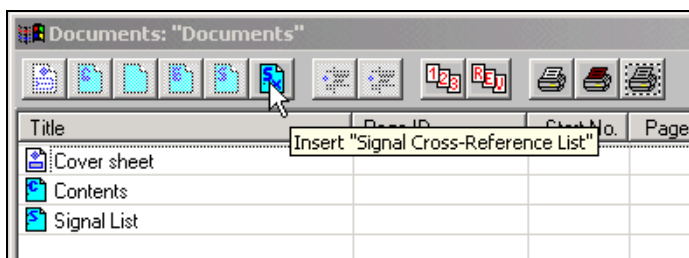


Fig. 168: Completar a lista de referências cruzadas

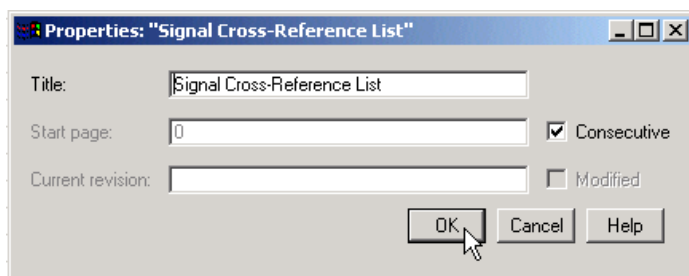


Fig. 169: Renomear a lista de referências cruzadas

- Clicar no nome do projeto, manter a tecla do mouse pressionada e arrastar o projeto para dentro da documentação.
A documentação agora mostra todos os elementos contidos no projeto.

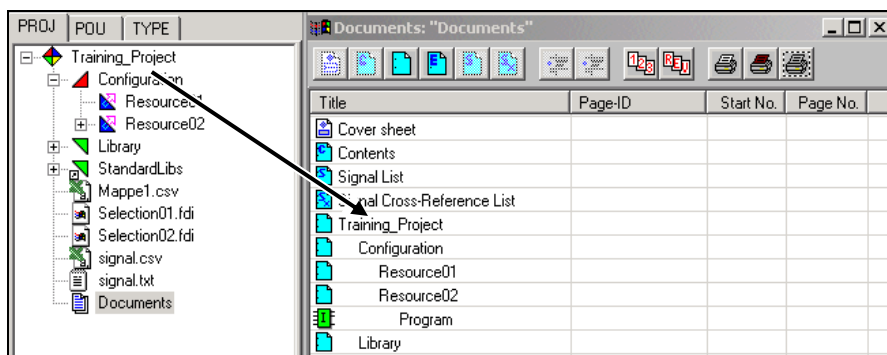
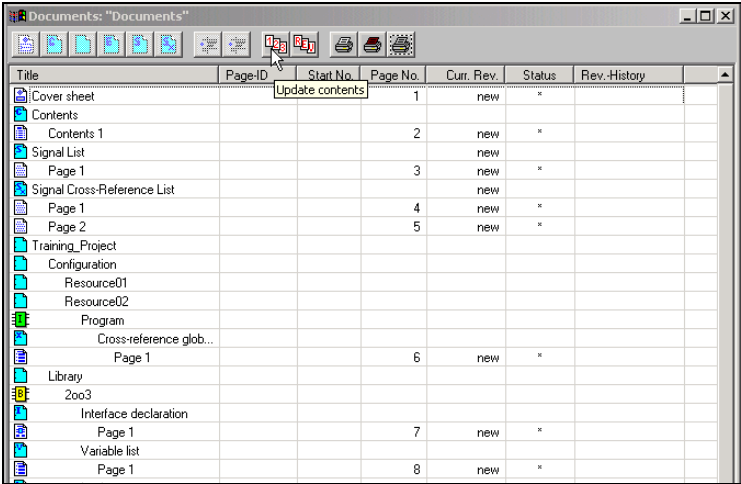


Fig. 170: Adicionar projeto à documentação

Nota: Também é possível inserir apenas objetos do seu projeto individualmente numa documentação.

Passo 4: Atualizar o sumário:

☐ Clicar no botão **Update contents** .



The screenshot shows a software window titled "Documents: Documents". It features a toolbar with various icons, including a button labeled "Update contents" which is highlighted by a mouse cursor. Below the toolbar is a table with the following columns: Title, Page-ID, Start No., Page No., Curr. Rev., Status, and Rev.-History. The table lists various document components and their corresponding page numbers and revision status.

Title	Page-ID	Start No.	Page No.	Curr. Rev.	Status	Rev.-History
Cover sheet			1	new	*	
Contents						
Contents 1			2	new	*	
Signal List				new		
Page 1			3	new	*	
Signal Cross-Reference List				new		
Page 1			4	new	*	
Page 2			5	new	*	
Training_Project						
Configuration						
Resource01						
Resource02						
Program						
Cross-reference glob...						
Page 1			6	new	*	
Library						
Zoo3						
Interface declaration						
Page 1			7	new	*	
Variable list						
Page 1			8	new	*	


Fig. 171: Atualizar o sumário

Depois de atualizar o sumário, as páginas individuais são mostradas com sua numeração.

As seções no objeto de documentação correspondem à pasta do projeto e às pastas de bibliotecas. A estrutura pode ser identificada pelos recuos.

Nota: É possível mudar a ordem dos elementos ou excluir elementos individualmente. Depois sempre deveria atualizar o sumário.

Passo 5: Definir um estado de revisão:

- ☐ Clicar no botão **Create revision** 

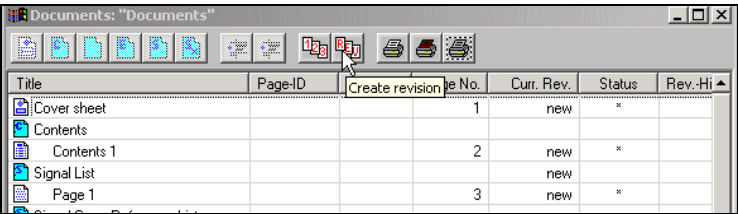


Fig. 172: Criar revisão

- ☐ Inserir um estado de revisão e clicar em **Create**.

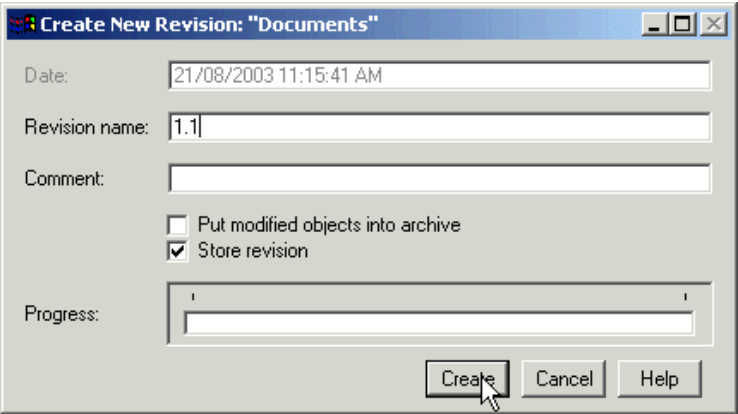


Fig. 173: Inserir um estado de revisão

Passo 6:

Atualizar as entradas da capa:

- ☐ Clicar com o botão direito do mouse no objeto de documentação e selecionar **Properties** no menu de contexto.
- ☐ Introduzir os dados para as linhas “Made by”, “Make date”, “Enduser 1” e “Ordernumber” na coluna “Value”. No mínimo as linhas listadas devem ser preenchidas.

Name	Key	Value	Value of Parent	Data Type
Drawing no. general (23...)	13			STRING
Made by (8 C)	14	Lämmer	Name	STRING
Make date (8 C)	15	12.05.05	Date	STRING
Check name (8 C)	16			STRING
Check date (8 C)	17			STRING
Status (7 C)	18	LZ	LZ	STRING
Conversion (8 C)	19			STRING
Based (18 C)	20			STRING
Special notes (10 C)	26			STRING
Revision sign a (1 C)	27	1	1	STRING
Revision a (8 C)	28	1.1		STRING
Date revision a (8 C)	29	12.05.05		STRING
Name revision a (8 C)	30	Lämmer		STRING
Revision sign b (1 C)	31	2	2	STRING
Revision b (8 C)	32			STRING
Date revision b (8 C)	33			STRING
Name revision b (8 C)	34			STRING
Revision sign c (1 C)	35	3	3	STRING
Revision c (8 C)	36			STRING

Fig. 174: Janela de edição das entradas de capa.

Nota: No caso de alterações, também os respectivos campos para elas devem ser modificados.

Passo 7:

Imprimir a documentação:

- ❑ Iniciar a impressão para páginas individuais, alteradas ou todas, via menu de contexto, ou através dos botões na barra de símbolos.

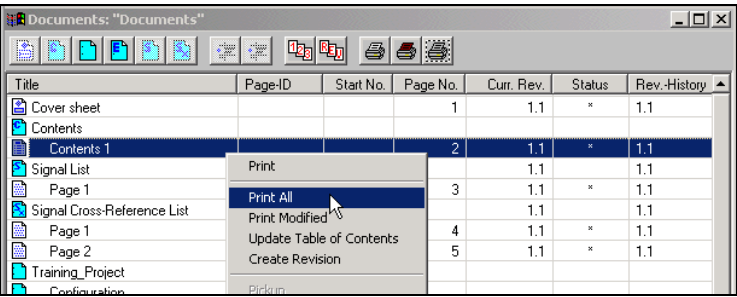


Fig. 175: Impressão pelo menu de contexto

1. Imprimir tudo
2. Imprimir páginas alteradas
3. Imprimir páginas seleccionadas

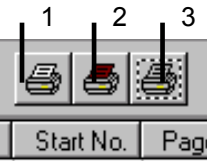


Fig. 176: Iniciar impressão

7.15.2 Documentação do hardware

A documentação do hardware pode ser impressa por inteiro ou apenas elementos isolados.

7.15.2.1 Imprimir a lista referências cruzadas de sinais

Para uma boa rastreabilidade dos sinais no software e hardware, uma lista dos sinais com os seus locais de utilização será necessária.

Passo 1:

Imprimir a lista de referências cruzadas:

- ☐ Clicar no menu **Signals** em **Print Cross References**.



Fig. 177: Imprimir a lista de referências cruzadas

Nota:

*A lista de sinais também pode ser impressa no menu **Signals** com **Print Signal List**, sem as referências cruzadas.*

7.15.2.2 Imprimir a documentação de recursos

Nota: A documentação de recursos contém todos os dados referentes ao hardware inclusive conexões de E/S, sinais Peer-to-Peer e outros sinais de comunicação.

- Passo 1:** Imprimir a documentação de recursos:
- ☐ Alternar ao Hardware Management.
 - ☐ Selecionar **Print...** no menu de contexto do recurso

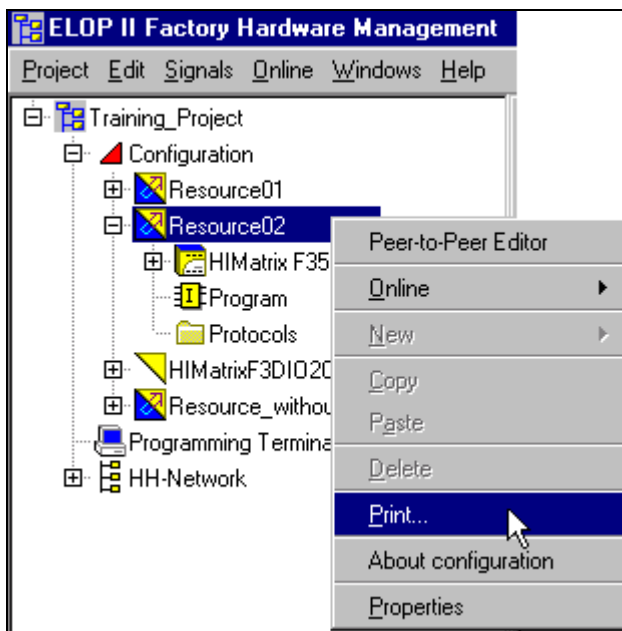


Fig. 178: Imprimir a documentação de recursos

Nota: Também é possível iniciar a impressão de todos os recursos em conjunto (veja Capítulo 7.15.2.3).

7.15.2.3 Imprimir a documentação do Hardware Management

Passo 1: Imprimir a documentação inteira:

- ☐ Clicar no Hardware Management em **Project**.
- ☐ Selecionar
 - **Print All...**, para imprimir a documentação inteira,
 - **Print...**, para imprimir elementos anteriormente marcados,
 - **Print Preview**, para a previsão da documentação de um objeto anteriormente marcado,
 - **Print Preview All**, para a previsão da documentação do projeto inteiro.



Fig. 179: Imprimir a documentação do Hardware Management

7.16 Arquivar

Para se obter uma cópia de segurança do programa de aplicação, deve ser arquivado o projeto.

Um programa de aplicação carregado ao sistema de comando não pode ser lido de volta ao aparelho de programação!

Sempre que um projeto tiver alcançado um estado intermediário importante ou tiver sido carregado num sistema de comando, o projeto deveria ser arquivado. A arquivagem ocorre na gestão de projetos.

Passo 1: Arquivar iniciar:

- ☐ Clicar com o botão direito do mouse no projeto.
- ☐ Selecionar **Archive...** no menu de contexto do recurso

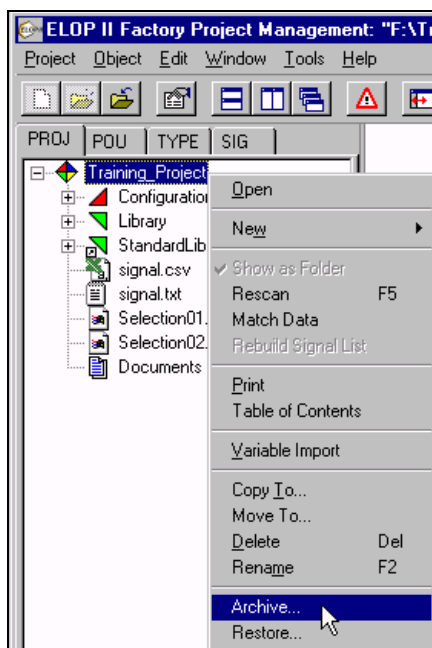


Fig. 180: Arquivar um projeto

Passo 2:

Introduzir o nome do arquivo:

- ☐ O nome do arquivo é definido clicando em **Browse...**, na janela de diálogo (Fig. 182), selecionando um diretório e introduzindo o nome do arquivo no campo “Target-file”.

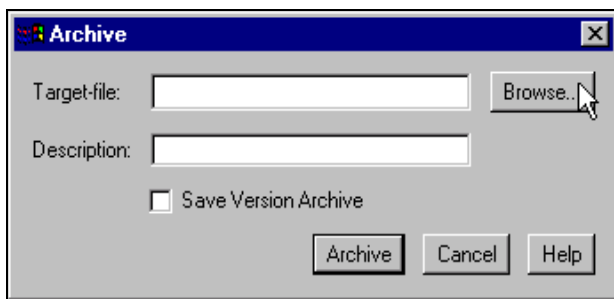


Fig. 181: Localizar o caminho do arquivo

- ☐ Introduzir no campo “Object name”, (veja abaixo), o nome do arquivo sem extensão. Também a data (ano_mes_dia) sempre deve ser parte do nome.

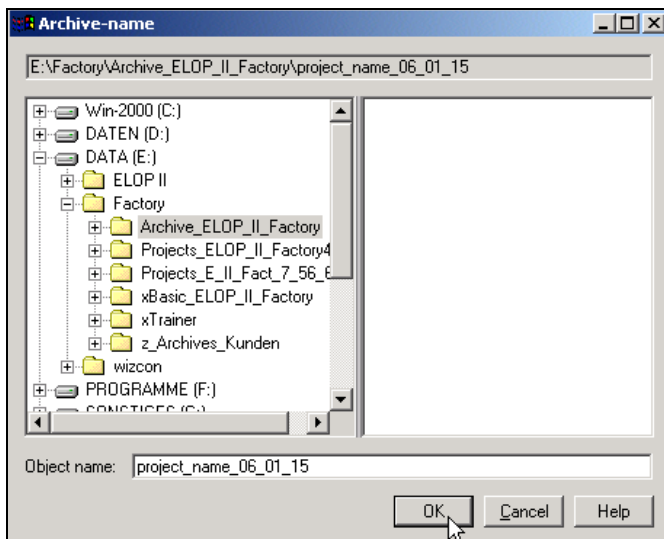


Fig. 182: Definir o nome do arquivo

- ☐ Introduzir no campo “Description” um comentário para identificar o seu projeto.
- ☐ Clicar no botão **Archive**.

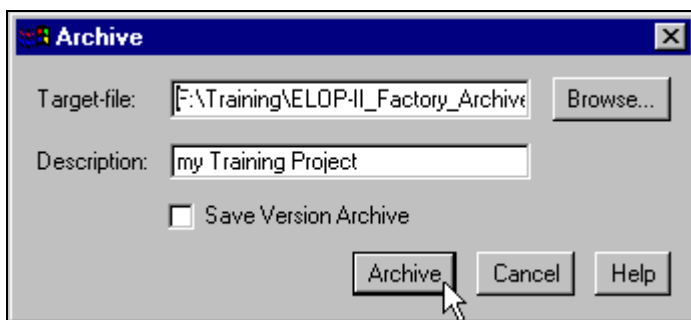


Fig. 183: Descrição do arquivo

O projeto é arquivado para o diretório selecionado com a extensão “.L3P”, em três arquivos.

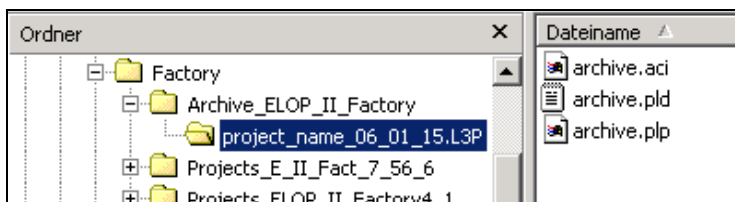


Fig. 184: Projeto arquivado

7.17 Restaurar

Nota: Para poder restaurar um projeto de um arquivo no ELOP II Factory, nenhum outro projeto pode estar aberto.

Passo 1: Iniciar o processo de restaurar um projeto:

- ☐ Clicar no menu **Project** em **Restore Project...**

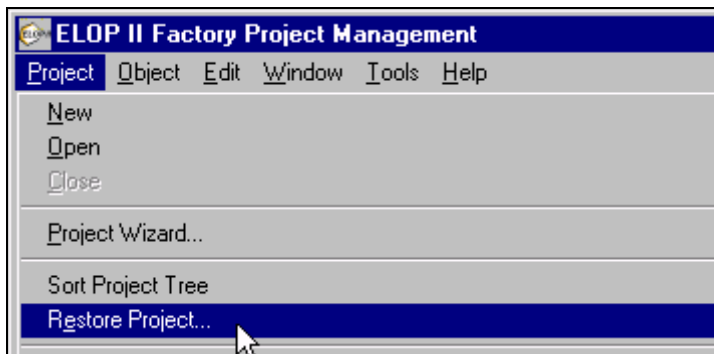


Fig. 185: Chamar “Restore Project...”

Passo 2:

Selecionar o projeto a ser restaurado:

- ❑ Clicar no botão **Browse...** (para o diretório de destino), para definir o caminho numa janela de seleção (Fig. 188) clicando com o mouse.

Caminho para onde o projeto restaurado deve ser salvo.

Caminho e nome de arquivo do projeto arquivado a ser restaurado.

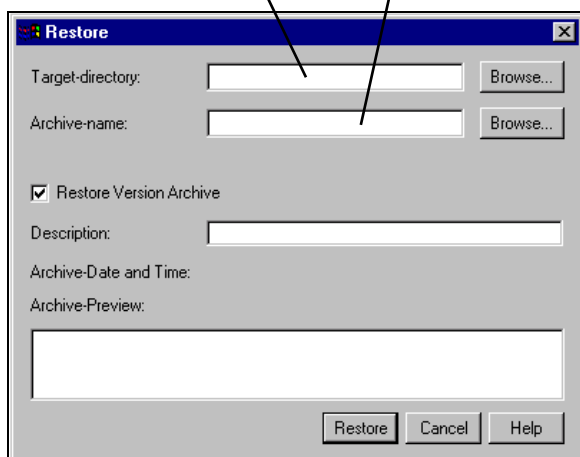


Fig. 186: Restaurar arquivo

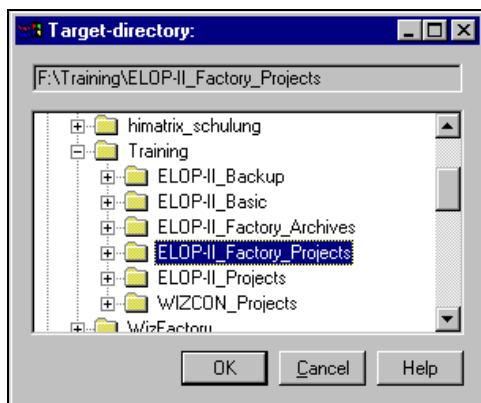


Fig. 187: Selecionar diretório de destino

- ☐ Clicar no botão **Browse...** (para o nome de arquivo), para definir o caminho numa janela de seleção (Fig. 188) clicando com o mouse.
- ☐ Marcar do lado esquerdo da janela de diálogo o caminho onde o arquivo se encontra. Do lado são exibidos apenas os arquivos que são detectados como tais.
- ☐ Marcar o arquivo desejado.
- ☐ Clicar em **OK** para finalizar a seleção.

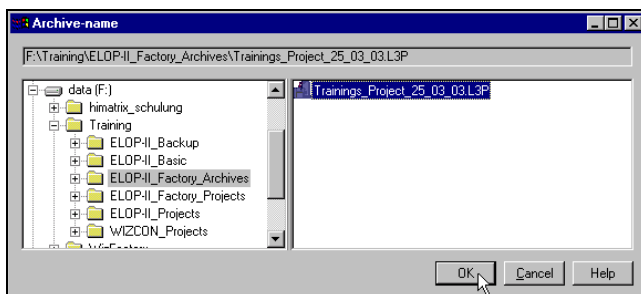


Fig. 188: Selecionar arquivo

Passo 3:

Finalizar a restauração:

- ☐ Clicar no botão **Restore**.

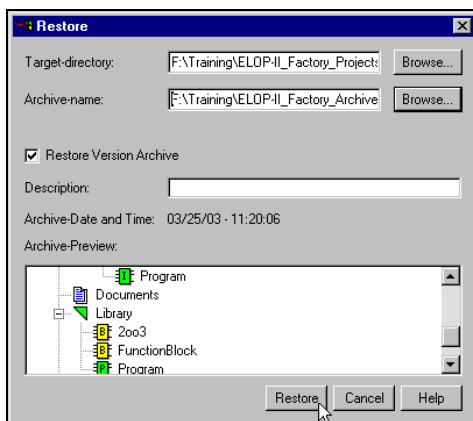


Fig. 189: Finalizar a restauração

Depois de restaurar o projeto com êxito, o mesmo é exibido na janela de estrutura.

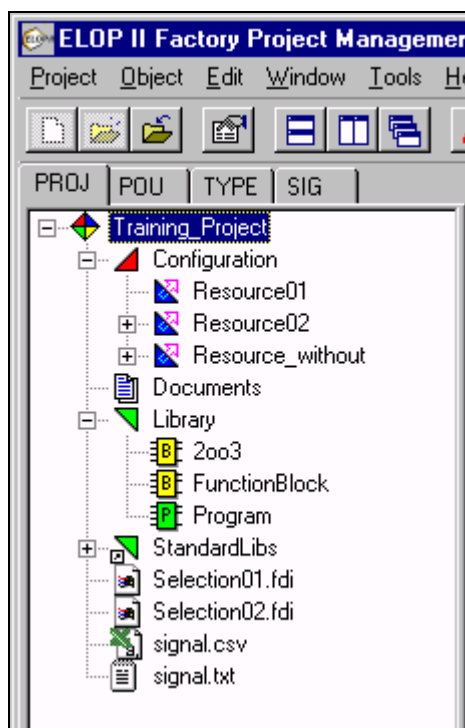


Fig. 190: Projeto restaurado

8 Anexo

8.1 Glossário

Abordagem centrada no meio

Método do ELOP II Factory para apresentar a lógica da função: A primeira página do esquema de funções está no centro de um esquema de funções teoricamente de tamanho ilimitado que pode se expandir para todas as direções.

Área de desenho

Área do editor da linguagem de blocos funcionais onde a lógica é programada.

Área de trabalho

Área onde o objeto de dados é editado com a ajuda de editores.

Árvore do projeto

A representação em formato de árvore da estrutura dentro do projeto.

Barra de menu

Barra horizontal na qual se indicam os nomes de todos os menus.

Barra de símbolos

Barra com símbolos que permitem o acesso rápido a comandos.

Barra de status

Linha que emite informações de status na margem inferior do ELOP II Factory.

Barra de título

Barra horizontal na margem superior de uma janela que exibe o título de uma aplicação incl. objeto editado, ou o nome de uma função chamada.

Bloco funcional (FB)

O bloco funcional é a unidade de organização do programa (POU) do tipo FUNCTION_BLOCK. Dentro de um bloco funcional, é possível criar a lógica de função. Um FB consegue memorizar valores do passado (p.ex, Timer, Flip-Flop).

Bloco

Unidade de organização do programa (POU) que é usada e conectada no editor da linguagem de blocos funcionais. Os blocos estão na biblioteca padrão de blocos ou nas bibliotecas específicas do usuário.

Conexão

Não se trata de um verdadeiro objeto de dados, mas da definição de um caminho para um objeto (p.ex., biblioteca de blocos) que não foi criado dentro do projeto. Uma conexão para a biblioteca padrão é criada em cada projeto automaticamente.

Configuração

Conceito do mundo de PES. Recursos que formam uma unidade dentro de um projeto são criados dentro de uma configuração.

Controladora do Hardlock

Software de sistema que permite a comunicação com o Hardlock.

Dongle

Relevante para a versão completa. Veja também Hardlock.

DXF

Drawing eXchange Format; formato de troca de dados da empresa Autodesk. Padrão industrial para a troca de desenhos entre diferentes sistemas de CAD.

Editor da linguagem de blocos funcionais (FBD editor)

Editor com o qual a lógica dos blocos funcionais é criada.

Editor de declaração de interfaces(também chamado de Editor de blocos)

Área do editor da linguagem de blocos funcionais, onde a aparência gráfica de um bloco é definida.

Editor de declaração de variáveis

Área do Editor da linguagem de blocos funcionais, onde as variáveis do bloco são criadas e definidas.

Editor de documentos

Editor para coletar, estruturar e imprimir unidades de organização do programa (POUs) e objetos. Gerencia objetos do projeto atual em um documento geral.

Editor de sinais

No editor de sinais, todos os sinais são definidos.

Endereço IP

Endereçamento individual de um PES ou aparelho de programação para a comunicação.

Endereço MAC

Endereço específico do equipamento que já é atribuído pelo fabricante e que é único no mundo. Serve para estabelecer a primeira comunicação com o equipamento, para poder efetuar configurações específicas do projeto posteriormente.

Focalizar

Opção de navegação no ELOP II Factory. A área visível e exibida no editor da linguagem de blocos funcionais pode ser centrada na visão de página inteira ou na posição do ponteiro do mouse. Serve para a navegação rápida na lógica de funções.

Fonte

Tipo e tamanho da letra.

Formatstring

Elemento de uma linguagem de script própria do ELOP II Factory para a documentação. Uma sequência de caracteres que determina o tipo e tamanho de comentários ou referências cruzadas e que pode conter instruções de formatação.

Função (FUN)

Uma unidade de organização do programa (POU) do tipo FUNCTION. No caso de uma função, os estados de saída são determinados pelos estados de entrada a cada ciclo. Isso significa que numa função não pode ser usada lógica que tenha comportamento de armazenar (p.ex., AND, OR)

Gestão de documentos

Função integrada no Editor de documentos, com a qual diversos objetos são coletados para imprimir estes documentos como conjunto e aplicar uma postura única de revisão.

Gestão de projetos

1. O programa principal do ELOP II Factory que trabalha orientado na aplicação. Com a gestão de projetos, um projeto é criado, gerenciado, arquivado e restaurado.
2. Janela de aplicação dentro da qual a estrutura do projeto é representada e todos os editores em relação a criação da lógica são iniciados.

Grupo simbólico

Todos os sistema de comando HIMatrix que trocam sinais entre si, devem estar listados em grupos simbólicos.

Hardlock

Proteção de hardware para o pacote de programa ELOP II Factory. Um conector que é conectado à interface paralela de um computador. Condição para a operação do ELOP II Factory. O mesmo também contém os direitos de acesso para partes do produto do ELOP II Factory e produtos acessórios.

Hardware Management

A totalidade de dados e propriedades relativos ao hardware são editados aqui. Definir o tipo de recurso, definir os sinais e atribuir aos canais dos recursos e definir a comunicação entre os recursos, etc.

Importação/exportação de variáveis

Funcionalidade do ELOP II Factory com a qual as listas de variáveis de arquivos ou bancos de dados externos (p.ex., arquivo CSV, arquivos Excel, bancos de dados) podem ser importadas para dentro de um projeto.

Indicador de status e erros

Uma área dentro da gestão de projetos ou no Hardware Management onde as mensagens de erros e status do ELOP II Factory são emitidas.

Instância de programa

A utilização correta de um tipo de programa. Uma instância de programa executa no sistema de comando do recurso aquela funcionalidade que está definida na declaração do tipo de programa correspondente.

Instância

Utilização concreta de uma unidade de organização do programa (POU) num programa.

O programa em si também é instanciado para a sua utilização dentro de um recurso (veja também instância de programa).

Janela de estrutura

Área composta de várias páginas de registros e que oferece diferentes visões sobre a estrutura do projeto carregado.

Janela de visão geral

Serve para a navegação rápida e simples na área do editor da linguagem de blocos funcionais que representa as páginas de lógica do editor da linguagem de blocos funcionais numa visão geral reduzida. Dentro desta janela, mediante mouse e teclado, é possível colocar o foco sobre a posição que deve ser visualizada na área de desenho do Editor.

Liberação

Ativar uma função do ELOP II Factory ou de um produto complementar num Hardlock.

Linguagem de blocos funcionais (FBS)

Uma linguagem de programação para descrever redes com elementos booleanos, aritméticos e semelhantes, que trabalham simultaneamente.

Linguagem de sequências

Uma linguagem de programação para a descrição de processos sequenciais e paralelos na lógica do esquema de funções com controle do tempo e dos eventos (sequências de passos).

Maximizar

Aumentar uma janela até o tamanho máximo.

Menu de contexto

Menu que é exibido diretamente acima do objeto selecionado depois de acionar o botão direito do mouse. O menu contém comandos que podem ser aplicados ao objeto.

Minimizar

Diminuir uma janela até o tamanho de um ícone.

ONLINE Test (Power-Flow)

Além de muitas funções online oferecidas pelo Hardware Management, também há um teste online na gestão de projetos. O mesmo serve para rastrear durante a operação do sistema de comando todos os valores das variáveis e dos sinais dentro da representação da lógica.

Online

Funções que lêem e carregam dados de e para os recursos. Carregar, iniciar e parar o programa. Acompanhar e forçar sinais etc.

Pasta

Mesmo significado de diretório. Uma pasta pode conter tanto outras pastas quanto objetos de arquivo.

Projeto modelo

Projeto do ELOP II Factory que foi instalado e que contém os ajustes prévios para um projeto. Cada novo projeto é criada a partir deste modelo. O projeto modelo pode ser adaptado.

Projeto

Objeto de pasta no qual todos os outros objetos são contidos. Para poder trabalhar na gestão de projetos, deve ser aberto um objeto de projeto.

Proxy Resource

Um objeto de recurso Proxy apenas está presente num projeto A para representar um recurso que de fato existe num projeto B. É necessário para uma comunicação de Ethernet segura para além do limite do projeto entre sistemas de comando HIMatrix de vários projetos.

Quick-Info

Texto de ajuda breve que é emitido quando o ponteiro do mouse é posicionado sobre um botão.

Recurso

Meio de estruturação da IEC61131-3, que corresponde a uma unidade central do sistema PES. Em um objeto de recurso, são criadas as instâncias de programa do lado da gestão de projetos. Do lado do Hardware Management, o tipo de recurso é atribuído e todas as demais configurações e atribuições são efetuadas.

Rede

Significado na norma IEC 61131-3: todos os elementos que são ligados uns aos outros de forma gráfica.

Registro

Elementos de abas em janelas que oferecem ao usuário diversas informações e opções de seleção relacionadas e facilitam a navegação por diversas páginas.

Revisão

Conceito do gerenciamento de documentos do ELOP II Factory. Uma revisão é uma versão verificada ou revisada de um objeto de documento sempre em relação ao documento total. Com ajuda da gestão de revisões, é possível criar várias revisões.

Simulação offline

Programa que permite o teste gráfico da instância de programa ou da unidade de organização do programa: a lógica é “animada”. Assim, erros podem ser precocemente detectados e eliminados.

Sinal

Todos os valores que podem ser lidos ou escritos pelas entradas e saídas do hardware são sinais. Da mesma forma todos os valores que podem ser trocados pela comunicação e também informações de sistema inclusive valores de diagnóstico. Um sinal também contém uma prescrição de atribuição entre diferentes áreas do sistema de comando inteiro.

System ID (SRS)

O System ID (SRS = System-Rack-Slot) pode ser comparado a um número de participante e só pode ser usado uma única vez no projeto. Teoricamente podem ocorrer valores de 1 a 65535.

Tipo de dados

Define as características da faixa de valores de uma variável.

Tipo de programa (PROG)

Uma unidade de organização do programa (POU) do tipo PROGRAM. O tipo de programa representa o nível mais elevado de uma POU, ou seja, contém a lógica completa na forma de funções e blocos funcionais.

Variável

Denominação para uma memória de dados que pode receber valores que são determinados pelo tipo de dados, bem como pelas definições na declaração de variáveis.

Zoom

Opção de navegação no ELOP II Factory. A área visível e exibida no editor da linguagem de blocos funcionais pode ser aumentada ou diminuída.

8.2 Índice remissivo

.L3P.....	164
Abordagem centrada no meio	169
Administrador.....	3
Adobe Acrobat Reader	5
Ajuda online	19, 25, 39
Aparelho de programação	88
Área de desenho	39, 169
Área de trabalho	12, 17, 35, 169
Maximizar a ~	34
Arquivar.....	162
Árvore de diretórios	54
Árvore do projeto	169
Atribuição de hardware.....	84
Barra de menus	12, 13, 21, 169
Barra de símbolos.....	12, 14, 169
Barra de status	12, 14, 169
Barra de título	12, 13, 169
Baustein	
Salvar ~	71
Biblioteca	27, 32, 39
Criar ~	55
Renomear ~	56
Bitstr	66
Bloco	13
Denominação do ~	63
Drag&Drop	39
Texto do ~	63
Bloco funcional	169
Criar tipo de ~.....	57
Duplicar ~	68
Editar ~	59
linguagem de ~.....	17, 172
tipo de ~	28
BOOL	43
Campo de valor	39
Campo OLT	
Criar ~	80
CD-ROM	II
Centrado no meio	36

Comunicação	
Ajustes de fábrica	122
Conectar sinais de processo	112
Conectar sinais de sistema	114
Direção de transferência	113
Editor de nódulos	110
Grupo simbólico	108
Grupo simbólico, propriedades	109
Peer-to-Peer	107
Perfil de uma conexão P2P	111
Rede HH	108
Conexão	170
Configuração	29, 32, 170
Control Center	6, 11
Control Panel	
Abrir	135
Coordenadas	36
~ colunas	36
~ linhas	36
Criar HIMatrix F3 DIO 20/8 01	93
Declaração de interfaces	
Definir ~	62
Declaração de variáveis	59
Desinstalação	7
Documentação	18
Elaborar a ~	40
Documentação do hardware	159
Documentação de todos os recursos	161
Documentação de um recurso	160
Documentação do software	
Atualizar sumário	155
Entradas da capa	157
Nova	152
Objeto de documentação	152
Renomear objeto	153
Documentos	
Editor de ~	17, 18, 170
gestão dos ~	40, 171
Dongle	1, 170
DXF	41, 170
Editar dados da página	67
Editor da linguagem de blocos funcionais	12, 14, 17, 33, 34, 170
Editor de declaração de interfaces	17, 33, 170
Editor de declaração de variáveis	17, 33, 34, 39, 59, 170

Endereço IP	88, 170
Definir	90
Parametrizar aparelho de programação	121
Endereço MAC	88, 170
Estrutura do projeto	76
Explorer.....	5
FBD editor.....	59
Focalizar	171
Fonte.....	171
Force Editor	139
Forcing	
Anexar seleção de sinais	147
Carregar seleção de sinais	146
Enviar valores	142
Gravar situação atual de Forcing	148
Salvar a seleção de sinais	145
Seleção de sinais	141
Tempo de forcing	143
Formatstring.....	171
Função	28, 171
Geração de código	115
Gerador de código	51, 116
Gestão de projetos	6, 11, 12, 31, 171
Grade	68
Grupo simbólico	171
Hardlock.....	1, 3, 4, 171
Controladora do ~	3, 170
Hardware do computador	4
Hardware Management	172
IEC 61131-3.....	66
Impressora	4
Indicador de status e erros	11, 12, 19, 34, 172, 183
Informações sobre a configuração	118
Iniciar	11, 53
Instalação	3, 6
Iniciar ~	5
Instância	172
Nome de ~	64
Introdução de lógica na área de desenho	66
Janela de estrutura	12, 15, 172
bibliotecas na ~	32
Objetos na~	27
Janela de visão geral	17, 33, 34, 172
Liberação	172

Linguagem de sequências	17, 173
Linha de conexão	
Desenhar ~	39, 70
Lista de variáveis	69
Lógica	
Criar a ~	39
Maximizar	173
Mensagem de erro	19
Menu de contexto	16, 23, 173
Menu Iniciar	6
Minimizar	173
Modelos de impressão	41, 42
Módulo	
Definir slot	92
Módulos de entrada e saída	
~ Adicionar	91
Definir slot	92
NewLib	55
Nome longo	61
Numeração de folhas	36
Online	173
ONLINE Test	150, 173
Página	
~ ativa	36
Pasta	173
Pasta de ~	31
PES	
Definir tipo de ~	85
Pograma	
Editar ~	72
Program Organization Unit (POU)	27
Programa	
Arranque a frio/arranque quente	87
Atribuir tipo de ~	75
Carregar	135
Criar tipo de ~	57
Iniciar ~	137
Instância de ~	27, 30, 76, 172
Nova ~	74
Propriedades de iniciar	87
Tipo de ~	27, 30, 174

Projeto.....	13, 15, 27, 32, 173
~ modelo	31
Arquivar o ~	162
caminho do ~	54
Novo	53
Restaurar	
Diretório de destino	166
Restaurar ~	165
Projeto modelo.....	173
Quick Info.....	14, 173
Recurso.....	29, 32, 174
Atribuir um tipo de ~	84
Tipo de ~	45
Recurso Proxy	173
Rede	174
Registro.....	174
Remote I/O	29, 93
Criar ~	95
Definir Rack ID	96
Resource	
Criar ~	73
Restaurar	165
Revisão	174
gestão de ~	18, 30
Simulação offline	
Alterar campo de valor	81
Simulação Offline.....	43, 77, 174
Encerrar ~	81
iniciar ~	78
Sinais	
Definir ~	98
Sinal de sistema.....	106
Sinais de sistema.....	51
Sinal	97, 174
Atribuir ~ a entradas/saídas do hardware	103
Editor	24, 98, 170
Utilização na lógica	100
SRS.....	49, 85, 174
StandardLibs.....	66
Subnet Mask	89
Suporte técnico	2
System ID	49, 85, 174

Tela padrão	12
~ na gestão de projetos	12
~ no Hardware Management	20
Tipo de dados	174
Valores de CRC	118
Variable	
Drag&Drop	39
Variáveis	
~externas	28
Variável	97, 175
Criar ~	59
Importar/Exportar	172
Versão do código	51
Zoom	37, 175

8.3 Lista de abreviações

AS	A blauf- S prache - Linguagem de sequências
AWL	A nweisungs- L iste - Lista de instruções
CA	C omentário A nexo
Campo OLT	C ampo para teste online
CG	C ode- G enerator - Gerador do código
CONST	Constante
CRC	C yclic R edundancy C heck
CRF	C ross R eference (Info sobre entradas e saídas)
CSV	Formato de dados para a função importar/exportar, formato ASCII com comas “,” como separador (C omma s eparated v alue)
D&D	D rag and D rop (Arrastar e soltar)
DXF	Formato gráfico padrão da AutoCAD, para originais
E/S	E ntrada/ S aída
FB	F unction B lock - Bloco funcional
FBD	F unction B lock- D iagram - Diagrama de blocos funcionais
FBS	F unktions- B austein- S prache - Linguagem de blocos funcionais
FSA	F ehler- S tatus- A nzeige - Visualização de Erros e Estados
FUN	F unção tipo de bloco
GV	G lobale V ariable - Variável global

HW	H ardware
MC	M ódulo C entral
OLS	O ffline- S imulation
OLT	O NLINE- T est
P2P	P eer- t o- P eer, comunicação ponto-a-ponto
PADT	P rograming and D ebugging T ool (PC)
PES	P rogramable E lectronic S ystem (sistema de comando)
POU	P rogram- O rganisation- U nit (bloco funcional)
RES	R essource - Recurso
RETAIN	Comportamento de aderência
RIO	R emote I/O , sistema de comando sem programa de aplicação
RT	R esource T ype
SRS	S ystem- R ack- S lot

HIMA
... a decisão segura



HIMA Paul Hildebrandt GmbH
Automação industrial
Postfach 1261 68777 Brühl

Telefone: (06202) 709-0 Telefax: (06202) 709-107

Email: info@hima.com Internet: www.hima.com

(0622)