

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/324953618>

Introduction to Industrial Communication Networks, Integration Systems and Control (PCWorx Express) – Apostilled material of the practical class – 001: Interbus and ILC 131 Starter...

Technical Report · May 2018

DOI: 10.13140/RG.2.2.28657.58726

CITATIONS

0

READS

2

2 authors:



Hermes Jose Loschi

University of Campinas

53 PUBLICATIONS 37 CITATIONS

[SEE PROFILE](#)



Yuzo Iano

University of Campinas

122 PUBLICATIONS 204 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Using Big Data to Improve Prediction Power Generation in Photovoltaic Systems [View project](#)



Pattern Recognition for Partial Discharge Analysis in Instrument Transformers [View project](#)



Aula Prática – 001 – 1ºSem 2018

Introduction to Industrial Communication Networks, Integration Systems and Control (PCWorx Express)

Prof. Dr. Yuzo Iano
Prof. MSc. Eng. Hermes José Loschi

Designação:	Aula Prática -001-1ºSem 2018
Tópicos:	Software and Hardware Architecture, Interbus and ILC 131 Starterkit Configuration, Software Model of IEC 61131, Program Organization Units and Data Management.
Revisão:	v00.001-2017
Professor Responsável:	Prof. Dr. Yuzo Iano

Termos gerais e condições de uso da documentação técnica

A publicação e atualização desta documentação técnica, não implicam necessariamente em qualquer obrigação por parte da Feec/Unicamp em fornecimento subsequente.

Os usuários desta documentação são responsáveis pela correta aplicação desta documentação, no âmbito dos cursos de pós-graduação da Feec/Unicamp, em conformidade com as normas e regulamentações vigentes.

Este documento, incluindo todas as ilustrações nele contidas, é protegido por direitos autorais.

Quaisquer alterações no conteúdo ou a publicação de extratos desta documentação é proibida.

Phoenix Contact reserva-se o direito de registrar seus próprios direitos de propriedade intelectual para as identificações de produtos da Phoenix Contact aqui usados.

Registro de tais direitos de propriedade intelectual de terceiros é proibida.

Outras identificações de produtos podem ser objeto de proteção legal, mesmo quando possam não estar indicada como tal.

Sumário

1.	Overview do ILC 131 STARTERKIT	4
1.1.	Requerimentos do sistema.....	4
1.2.	Instalando e operando	5
1.2.1.	ILC 131 STARTERKIT Estrutura.....	5
1.2.2.	Diagrama Básico do Circuito Interno.....	8
1.2.3.	Power Supply.....	10
1.2.4.	Digital inputs and outputs.....	12
1.2.5.	Instalando o software PC WORX EXPRESS	14
2.	Aplicação.....	14
2.1.	Criação de Projetos	14
2.2.	Armazenamento.....	16
3.	Áreas de Trabalho	17
3.1.	IEC programming Workspace	18
3.2.	Bus Configuration Workspace.....	19
3.3.	Process Data Workspace.....	20
4.	Configuração	21
4.1.	Interface Serial.....	21
4.2.	Rede Ethernet TCP/IP	21
4.2.1.	Configuração do Windows	22
4.2.2.	Configuração do IP da Controladora via software IPAssign	27
4.3.	Adição de cartões de interface.....	29
4.3.1.	Seleção automática:	29
4.3.2.	Seleção Manual:	32
5.	Overview Arquitetura de Hardware	33
6.	Operação.....	34
6.1.	Compilando um projeto	34
6.2.	Download.....	35
6.3.	Execução do projeto	36
6.4.	Análise.....	37
6.5.	Debug.....	37
6.6.	Diagnóstico	38
6.7.	Sequencia de Desenvolvimento de um Projeto	40
7.	Estrutura do projeto no PC WORX.....	41
7.1.	Configuração e Recurso	41
7.2.	TASK	43
8.	Program Organization Units (Unidades de Organização de Programas).....	44
8.1.	Linguagens suportadas.....	44
8.2.	IEC 61131.....	45
8.3.	Uso de Programs (Programas)	47
8.4.	Uso de Function Block (Blocos de Funç.....	48
8.5.	Uso de Functions (Funções)	50
9.	Gerenciamento de Dados	51
9.1.	Criando variáveis de dados do processo.....	51
9.2.	Princípio para declaração de variáveis	53
9.3.	Sintaxe de uma constante.....	55
9.4.	Exemplos de representação de constantes.....	55
9.5.	Variáveis GLOBAIS	56
9.6.	Variáveis LOCAIS	56
9.7.	Variáveis de Interface	57
10.	Exercício final	58
10.1.	Vamos criar nosso próprio exemplo?	59

1. Overview do ILC 131 STARTERKIT

O ILC 131 STARTERKIT é montado em uma placa e contém o ILC 131 ETH, incluindo um cartão de entrada analógica (IB IL AI 2/SF-ME). O hardware do ILC 131 STARTERKIT está completamente montado. A montagem é alimentada pela fonte de alimentação fornecida.

Para operar o conjunto, simplesmente conecte a fonte de alimentação do ILC 131 ETH à tensão de alimentação (tomada 100~230 VCA) e conecte o ILC 131 ETH ao seu PC através do cabo Ethernet fornecido.

Depois de alimentado corretamente, acione o interruptor no módulo de comutação, e os ledes LEDs de entrada do controlador (I1 ... I8) estarão à mostra.

1.1. Requerimentos do sistema

Sistemas operacionais suportados:

- Windows®XP SP3
- Windows®Vista Business SP2
- Windows®7 Professional (32/64 bit) SP1
- Windows®8 Professional (32/64 bit)
- Windows®8 Ultimate



Requerimentos hardware

Hardware requirements for PC WORX EXPRESS	
CPU	Pentium 4, 1 GHz (2 GHz recommended)
RAM	1 Gbyte (minimum), 2 Gbyte (recommended)
Hard disk space	2 Gbyte free memory space
DVD drive	Yes
Interfaces	1 x Ethernet (TCP/IP)
Monitor	XGA, resolution of 1024 x 768 pixels (minimum), SXGA, resolution of 1280 x 1024 (recommended)
Operating devices	Keyboard, mouse
Web browser	Java Standard Edition SE 6 (or later) with at least Java Runtime Environment JRE 6 (Version 1.6.x or later)

Software de programação

Software	Software version
PC WORX EXPRESS	≥ 6.30

Requerimento versão de Firmware

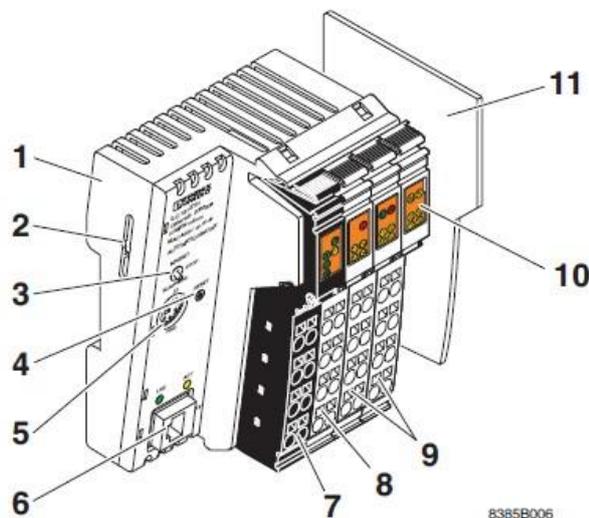
Device	Firmware version
ILC 131 ETH	≥ 4.01

Obs.: Se a versão do firmware em seu dispositivo for anterior à especificada, atualize seu firmware. O firmware mais recente pode ser baixado de: phoenixcontact.net/products.

1.2. Instalando e operando

1.2.1. ILC 131 STARTERKIT Estrutura

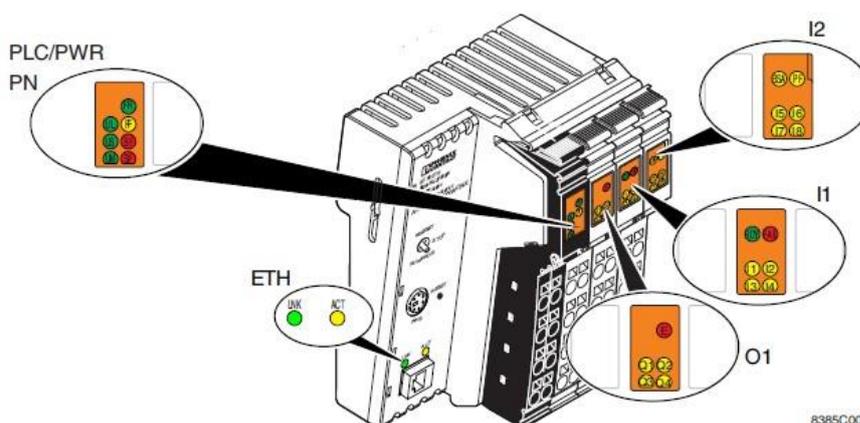
A estrutura do ILC 131 ETH está ilustrada abaixo:



O controlador *Inline* consiste nos seguintes componentes (sem o cartão analógico):

1. Base eletrônica;
2. Slot para a memória de parametrização / suporte do cartão (cartão SD - Acessório);
3. Seletor de modo;
4. Botão de reinicialização;
5. Interface RS-232;
6. Conexão Ethernet;
7. Conector 1: pontos terminais para alimentação de tensão;
8. Conector 2: pontos terminais de saída digital;
9. Conectores 3 e 4: pontos de terminais de entrada digital;
10. Diagnósticos e Indicadores de *status*;
11. Placa final;

Os indicadores de status são usados para diagnosticar a presença de erro, de maneira rápida, conforme ilustrado na figura abaixo e detalhado na tabela a seguir.

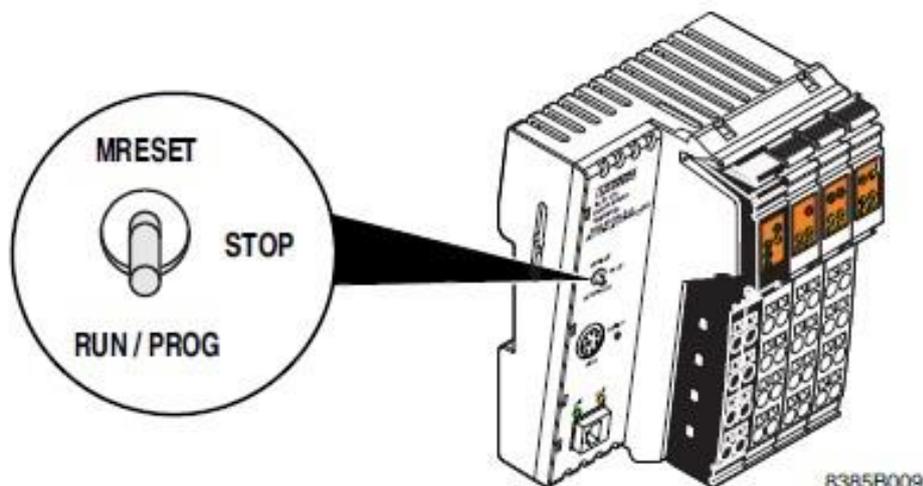


Designação	Cor	Status	Significado
ETH: State of the Ethernet interface			
LNK	Green	Off	Conexão não estabelecida com sucesso
		On	Conexão estabelecida com sucesso (link): o controlador <i>Inline</i> pode entrar em contato com outro dispositivo de rede.
ACT	Yellow	Off	Transmissão de dados inativa
		On or Flashing	Transmissão de dados ativa (atividade): a interface Ethernet está enviando ou recebendo dados.
PN: PROFINET			
BF	Red	Status do erro de comunicação / comunicação PROFINET (BusFail)	
		Off	O controlador PROFINET de nível superior estabeleceu uma conexão de comunicação ativa com o dispositivo PROFINET.
		On	O dispositivo PROFINET não estabeleceu uma conexão de comunicação ativa para um controlador PROFINET de nível superior.
SF	Red	Erro de grupo (PROFINET)	
		Off	Diagnóstico PROFINET não presente
		On	Diagnóstico PROFINET presente
PLC: Diagnósticos do controlador <i>Inline</i>			
FR	Green	Controlador <i>Inline</i> executado	
		Off	Sistema de tempo de execução IEC 61131 não está pronto para operar.
		Flashing	O sistema de execução IEC 61131 foi inicializado com sucesso. Função de controle no estado STOP ativada, programa não processado
FF	Yellow	Falha	
		On	Ocorreu um erro de execução no programa do sistema em execução no IEC 61131.
		Off	Não ocorreu nenhum erro de execução no programa do sistema em execução no IEC 61131
PWR: Fornecimento de tensão			
UL	Green	Fornecimento 24V (U_{ILC}) para gerar as tensões U_L e U_{ANA}	
		Off	Tensão de alimentação não presente.
US	Green	On	Tensão de alimentação presente (Indicação se a tensão de alimentação 24 V U_{ILC} presente).
		Alimentação 24 V para circuito de segmento	
UM	Green	Off	Tensão de alimentação não presente.
		On	Tensão de alimentação presente.
IL: INTERBUS diagnósticos			
RDY	Green	INTERBUS pronto para operar / transmissão de dados ativo (INTERBUS pronto / em execução).	
		Off	INTERBUS não está pronto para operar.
		Flashing	INTERBUS em estado PRONTO ou ATIVO
FAIL	Red	On	INTERBUS no estado RUN.
		Falha	
		Off	Não ocorreu nenhum erro
FAIL	Red	On	Um dos seguintes erros ocorreu: -Erro do barramento conectado (barramento remoto / barramento local) -Erro da controladora

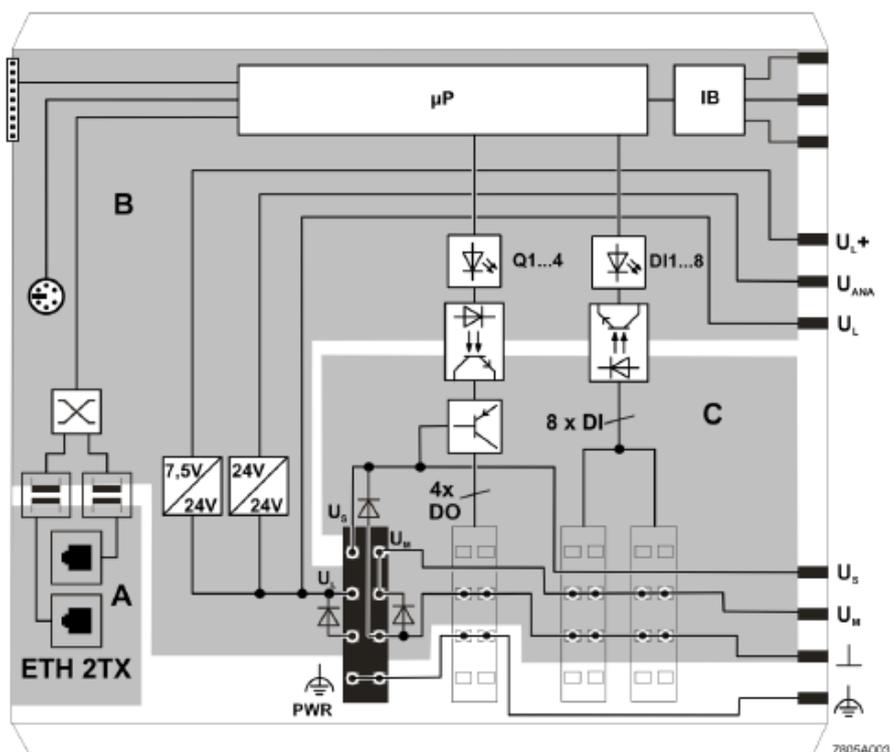
Designação	Cor	Status	Significado
BSA	Yellow		Segmento do barramento abortado
		Off	Segmentos no barramento conectados, estão ligados.
		On	Um ou mais segmentos no barramento conectados foram desligados
PF	Yellow		Falha periférica
		Off	Sem falha periférica em um dispositivo no barramento conectado.
		On	Falha periférica em um dispositivo no barramento conectado (barramento local ou barramento remoto)
I/O: digital inputs and outputs			
I1 a I8	Yellow		Entradas 1 a 8
		Off	A entrada correspondente não está configurada.
		On	A entrada correspondente está configurada
E	Yellow		Erro
		Off	Sem curto-circuito / sobrecarga em uma ou mais saídas.
		On	Curto-circuito / sobrecarga pelo menos uma das saídas (1 a 4)
Q1 a Q4	Yellow		Saídas 1 a 4
		Off	A saída correspondente não está configurada
		On	A saída correspondente está configurada

O seletor de modo é usado para definir o estado operacional do controlador. As posições RUN / PROG e STOP têm uma função de botão alternativo e o MRESET posiciona uma função de botão. Depois de soltar o botão na posição MRESET, ele retorna para STOP.

Modo operacional	Explicação
RUN/PROG	O controlador está no estado RUN. O programa de aplicação é processado. O software PC Worx Express pode ser usado para modificações no programa e configuração, bem como para a função de monitoramento on-line.
STOP	O controlador está no estado STOP. Processamento do programa de aplicação foi parado.
MRESET	Defina o seletor de modo na seguinte sequência para excluir os dados de retenção e o programa de aplicação: <ul style="list-style-type: none"> • Coloque o interruptor na posição MRESET por três segundos. • Solte o interruptor por menos de três segundos. • Coloque o interruptor na posição MRESET por três segundos.



1.2.2. Diagrama Básico do Circuito Interno



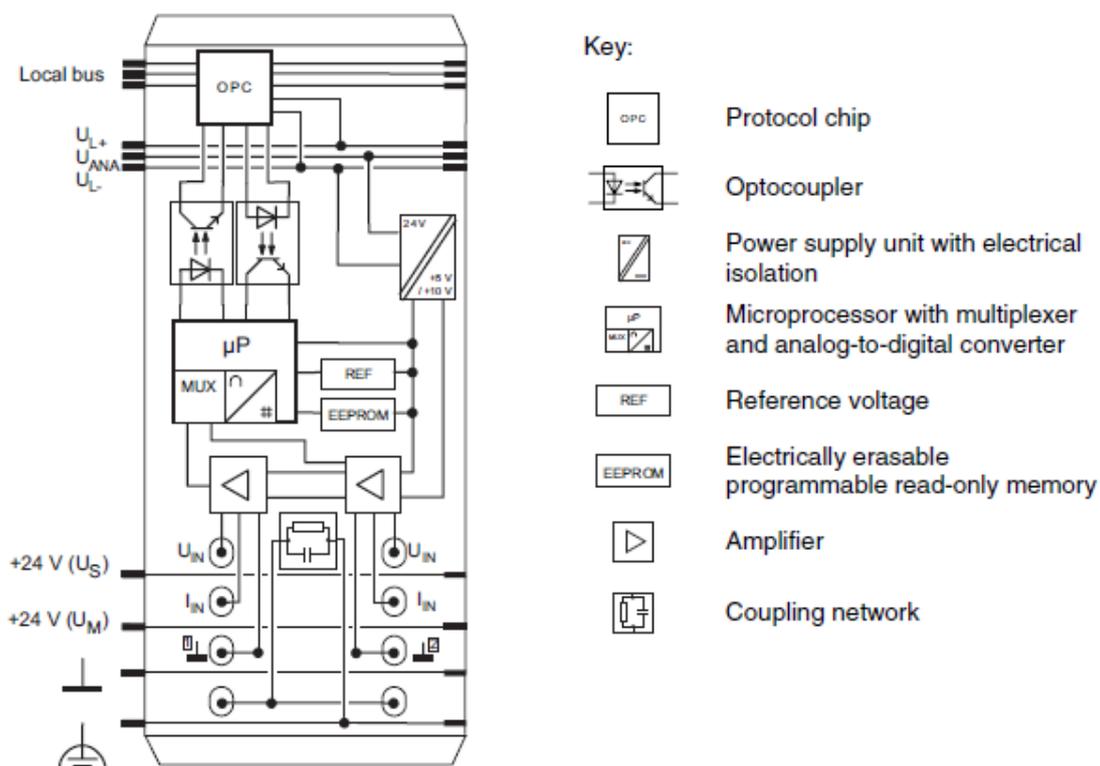
Key:

	Microprocessor		Inverter
	Protocol chip		LED
	RS-232 interface		Optocoupler
	Transmitter		NPN transistor
	RJ45 socket		Ethernet switch
	SD card holder (the SD card is not supplied as standard)		

As áreas destacadas como cinza no diagrama básico do circuito elétrico, representam as seguintes áreas isoladas:

- A: Ethernet interface
- B: Logic
- C: I/O

A interface do ILC 131 ETH torna possível a criação do barramento local (Local bus). O Local Bus é automaticamente criado para direcionar as conexões dos cartões digitais e/ou analógicos, com o controlador *inline*. No ILC STARTERKIT essa conexão foi realizada com o cartão de entrada analógica IB IL AI 2/SF-ME (vide imagem a seguir com um diagrama do circuito interno).



Exemplos de conexão. A Figura 7 e a Figura 8 mostram esquemas de conexão (sem conector de proteção). Um lado da blindagem deve ser conectado no sensor ou em um ponto apropriado para o aterramento do potencial ser funcional.

Connection of Active Sensors

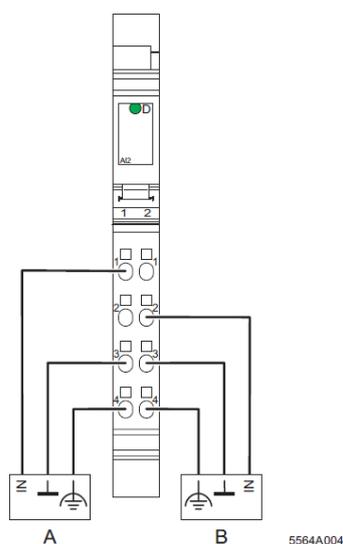


Figure 7 Connection of active sensors in 2-wire technology

- A Active sensor with voltage output (channel 1)
- B Active sensor with current output (channel 2)

Connection of Passive Sensors

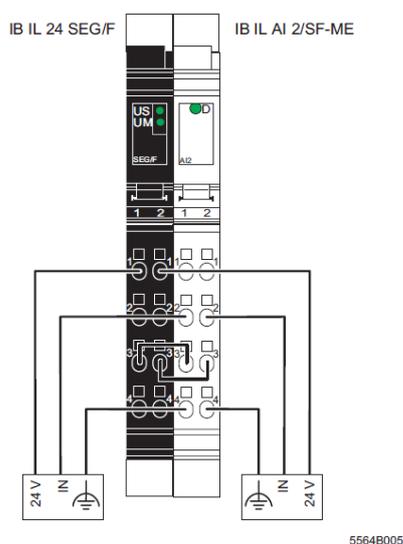


Figure 8 Connection of two passive sensors in 2-wire technology

Figure 8 shows the passive sensor supply. The sensors are supplied through a pre-connected segment terminal with a fuse. The sensors can also be supplied from an external power supply unit.

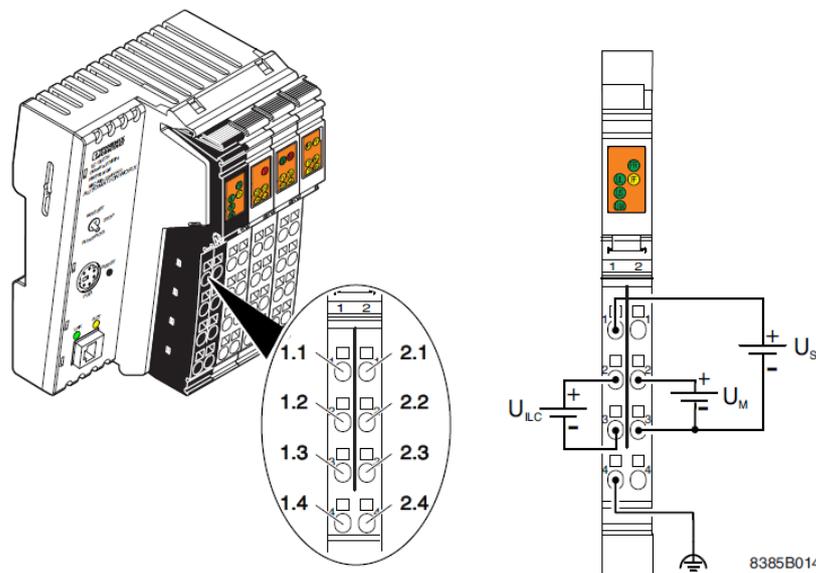
1.2.3. Power Supply

Escolha uma fonte de alimentação adequada para as correntes que serão demandadas da sua aplicação. A seleção depende da configuração do barramento, das correntes máximas resultantes e do tipo de fornecimento de tensão (fornecimento separado de U_{ILC} , U_M e U_S , ou fornecimento de tensão único).

Forneça a alimentação do controlador *Inline*, usando fontes externas de tensão de 24 V DC. A tensão admissível varia de 19,2 V DC a 30 V DC.

- Conecte as fontes de alimentação ao conector da fonte de alimentação, conforme mostrado a seguir.
- Insira o conector no controlador *Inline*.
- Ligue as fontes de alimentação.
- Os LEDs UL, UM e US se acendem e, após cerca de 10 segundos, os LEDs FR e RDY ficam acessos. O controlador *Inline* agora está totalmente inicializado.

Se os LEDs não acenderem ou começar a piscar, há uma falha grave no controlador *Inline*. Nesse caso, entre em contato com a Phoenix Contact.



O fornecimento de tensão no segmento (1.1) e o fornecimento principal (2.1, 2.2) devem ter o mesmo potencial de referência.

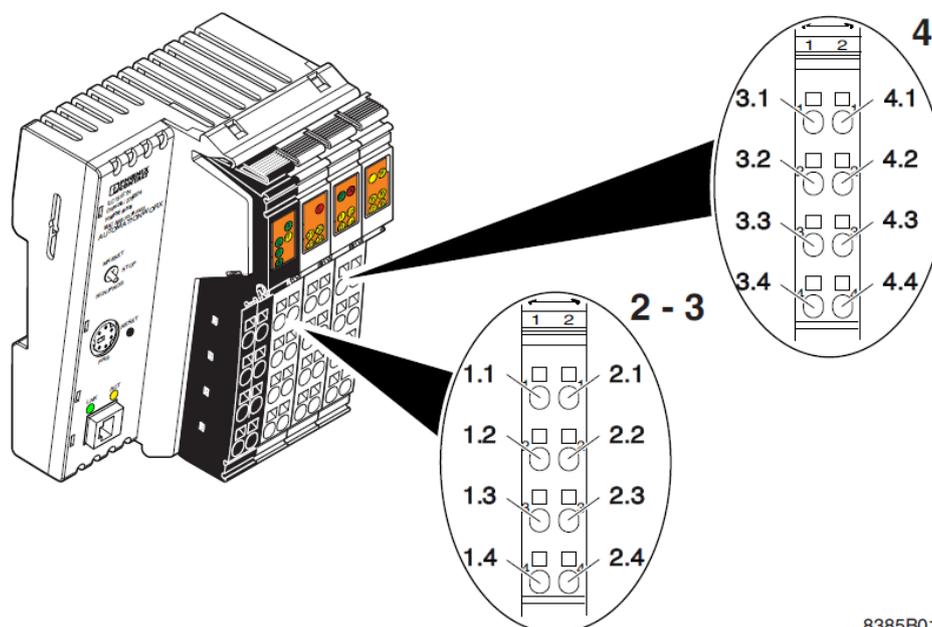
Existem várias maneiras de fornecer a tensão para o barramento:

- A tensão do barramento pode ser fornecida separadamente nos pontos terminais 1.1 e 2.3 (GND).
- As conexões 1.1 e 2.1 (ou 2.2) podem ser conectadas em ponte para garantir que o circuito do barramento tenha fornecimento a partir do circuito principal.

Ponto Terminal	Tarefa		Observação
Conector 1	Conector de alimentação		
1.1	24 VDC (U_S)	Fornecimento de tensão para barramento 24 V	A tensão fornecida é diretamente encaminhada barramento Interbus. Obs. Proteja a tensão de alimentação externamente de acordo com a carga conectada (dispositivos do barramento local) com 8 A, máximo.
1.2	24 VDC (U_{ILC})	Fornecimento de 24V	O “poder de comunicação” de 7,5 V (U_L) para o ILC e os dispositivos do barramento local conectados é gerado a partir desta tensão. A tensão analógica de 24 V (U_{ANA}) para os dispositivos do barramento local conectados também é gerada. Obs.: Proteja a tensão de alimentação externamente de acordo com a carga conectada (dispositivos do barramento local) com 2 A, máximo.
2.1, 2.2	24 VDC (U_M)	Fornecimento principal de tensão 24 V	A tensão principal, encaminhada para os dispositivos do barramento. Obs. Proteja a tensão de alimentação externamente de acordo com a carga conectada (dispositivos do barramento local) com 8 A, máximo.
1.3	LGND	Potencial logico de referência do aterramento (<i>Logic ground reference potential</i>)	Potencial de aterramento dos cartões de entradas e saídas analógicas e digitais.
2.3	SGND	Potencial de referência do segmento do aterramento (<i>Segment ground reference potential</i>)	O potencial de referência é uma referência de aterramento para o fornecimento de tensão principal e do barramento.
1.4, 2.4	FE	Aterramento Funcional (<i>Functional earth ground</i>)	O controlador <i>Inline</i> é aterrado quando é encaixado em um trilho DIN, já aterrado.

1.2.4. Digital inputs and outputs

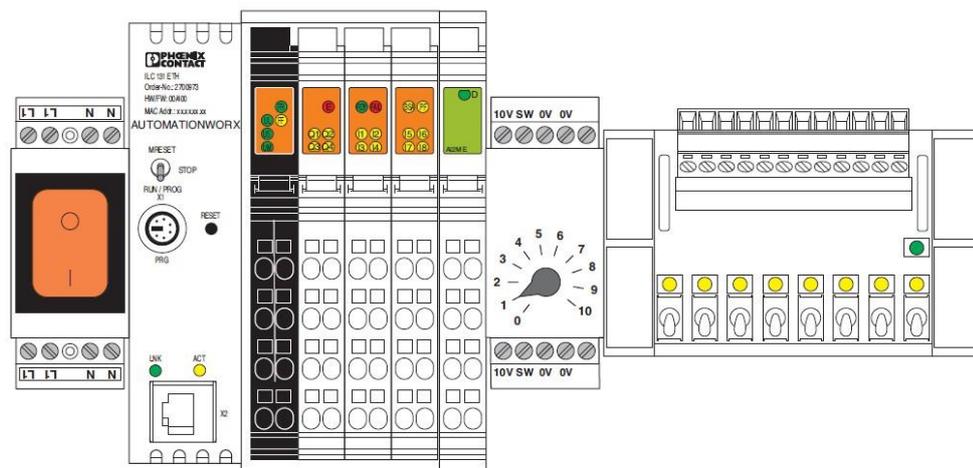
Existem oito entradas 24VDC e quatro saídas 24VDC.



8385B015

Ponto Terminal	Tarefa	Observação
Conector 2 Ponto terminal de saída		
1.1	Q1	Saída 1
2.1	Q2	Saída 2
1.2, 2.2	LGND	Potencial de aterramento dos cartões
1.3, 2.3	FE	Aterramento Funcional
1.4	Q3	Saída 3
2.4	Q4	Saída 4
A tensão de saída é 24 V DC a partir do fornecimento de tensão do barramento (U_S)		
Conector 3 Ponto terminal de entrada		
1.1	I1	Entrada 1
2.1	I2	Entrada 2
1.2, 2.2	24 V	Fornecimento de tensão principal U_M
1.3, 2.3	LGND	Potencial de aterramento dos cartões
1.4	I3	Entrada 3
2.4	I4	Entrada 4
Conector 4 Ponto terminal de entrada		
1.1	I5	Entrada 5
2.1	I6	Entrada 6
1.2, 2.2	24 V	Fornecimento de tensão principal U_M
1.3, 2.3	LGND	Potencial de aterramento dos cartões
1.4	I7	Entrada 7
2.4	I8	Entrada 8

Portanto, a estrutura completa do ILC 131 STARTKIT é ilustrada figura abaixo.

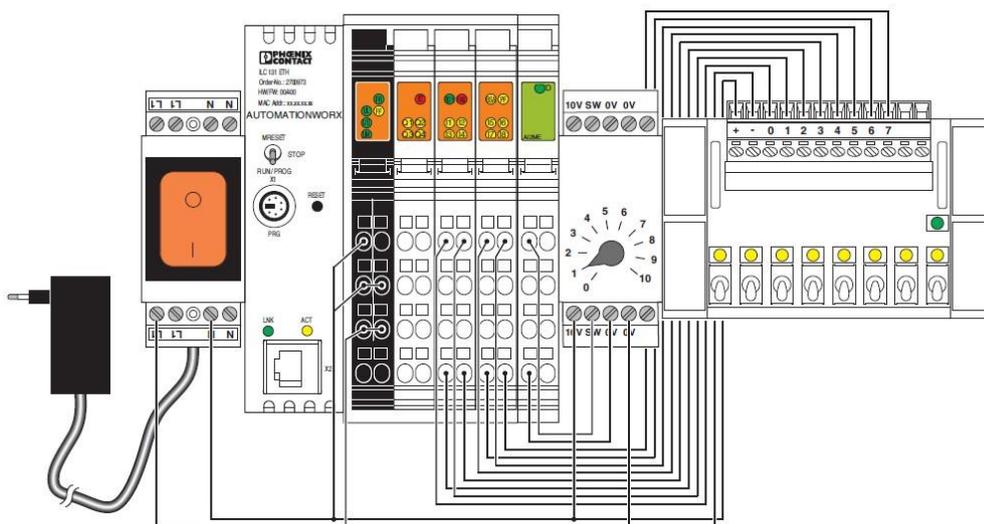


ILC STARTERKIT: Componentes instalados (potenciômetro e circuito de chaves bit 0 ou 1)

Com as seguintes entradas conectadas:

ILC STARTERKIT: Entradas usadas no exemplo

Device	Input	Signal at	Variable
ILC 131 ETH	Input I1	Plug 3, terminal point 1.1	ONBOARD_INPUT_BIT0
	Input I2	Plug 3, terminal point 2.1	ONBOARD_INPUT_BIT1
	Input I3	Plug 3, terminal point 1.4	ONBOARD_INPUT_BIT2
	Input I4	Plug 3, terminal point 2.4	ONBOARD_INPUT_BIT3
	Input I5	Plug 4, terminal point 3.1	ONBOARD_INPUT_BIT4
	Input I6	Plug 4, terminal point 4.1	ONBOARD_INPUT_BIT5
	Input I7	Plug 4, terminal point 3.4	ONBOARD_INPUT_BIT6
	Input I8	Plug 4, terminal point 4.4	ONBOARD_INPUT_BIT7
IB IL AI 2/SF-ME	Input I9	Plug 5, terminal point 1.1	Input_Analog



ILC STARTERKIT: Exemplo de cabeamento para as entradas e as conexões de fornecimento de tensão.

1.2.5. Instalando o software PC WORX EXPRESS

ATENÇÃO: Antes da instalação, feche todas as aplicações Windows abertas. Esta versão PC WORX EXPRESS pode ser instalada em paralelo a uma versão anterior existente. Primeiro instale o software PC WORX EXPRESS do DVD que está no conjunto ILC 131 STARTERKIT.



Depois de inserir o DVD na unidade, a página inicial do programa de instalação é aberta automaticamente.

Se a página de início do programa de instalação não abrir automaticamente, abra o programa de instalação através do arquivo "index.html" ("Computador, DVD Drive, Index.html").

Você é levado para a página principal:

- Selecione "Programas";
- Clique no "AUTOMATIONWORX Software Suite 2013 1.81;
- (PC WORX EXPRESS 6.30.767) "link e baixe o arquivo zip;
- Extraia o arquivo zip para uma pasta;
- Inicie o arquivo "Setup.exe" descompactado;
- Siga as instruções no programa de instalação.

ATENÇÃO: Durante a instalação, o programa de instalação pergunta quais componentes do ILC 131 STARTERKIT DVD devem ser instalados.

- Na caixa de diálogo que será aberta, especifique se deseja instalar o PC WORX EXPRESS apenas ou outros componentes também.
- Para instalar usando as configurações padrão, inicie o PC WORX EXPRESS através de "Iniciar, Programas, Phoenix Contact, AUTOMATIONWORX Software Suite 1.81, PC WORX EXPRESS 6.xx".

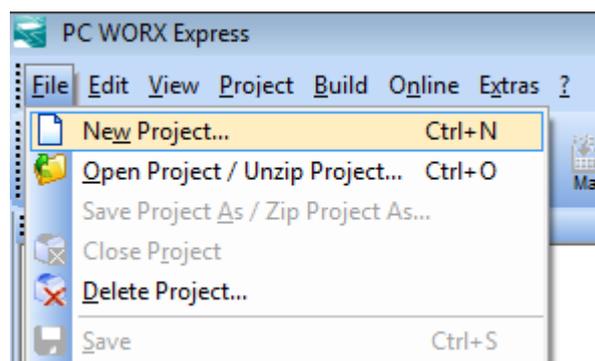
2. Aplicação

O PCWorx Express é um software gratuito, que tem como propósito desenvolver aplicações para as seguintes famílias de controladoras da Phoenix Contact:

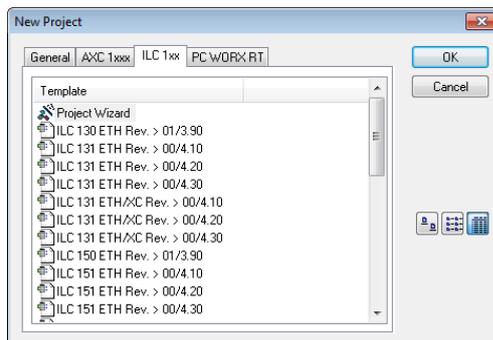
- ILC 1xx
- AXC1xxx

2.1. Criação de Projetos

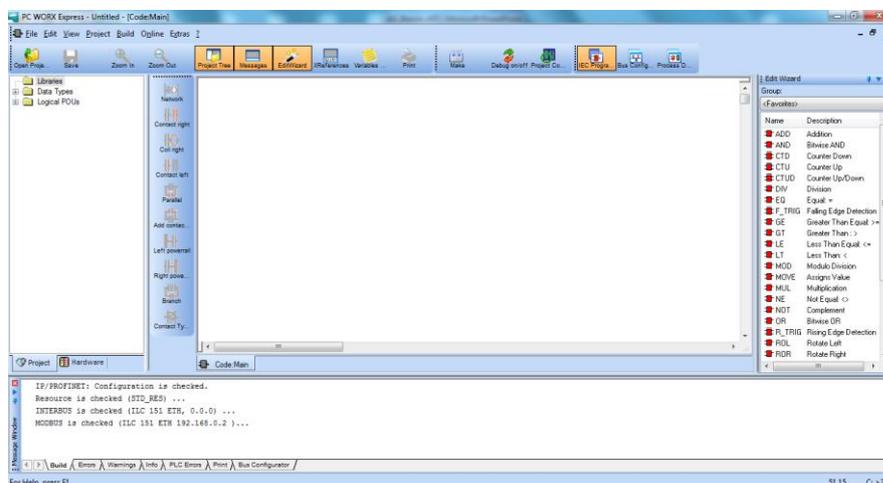
Para criar um novo projeto no PC WORX, abra o menu File -> New Project:



Ir aparece um tela para selecionar a controladora, onde deve ser selecionado o modelo correto, com sua respectiva verso de Firmware:

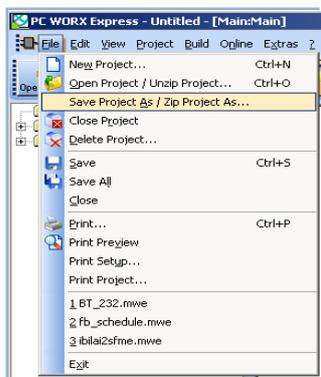


Logo aps selecionar o modelo correto (controladora e verso do firmware), ir aparecer a seguinte tela Inicial:

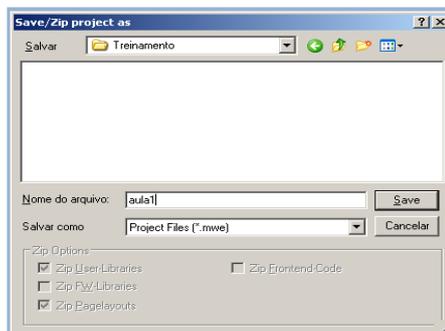


2.2. Armazenamento

Ao selecionar a opção “Salvar como”, o projeto em PC WORX é salvo com o nome escolhido e esses arquivos são então utilizados para processamento posterior.



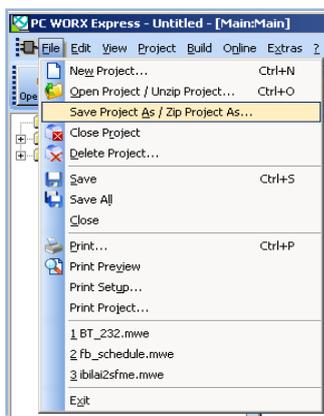
Salve seu projeto com a extensão *.mwt



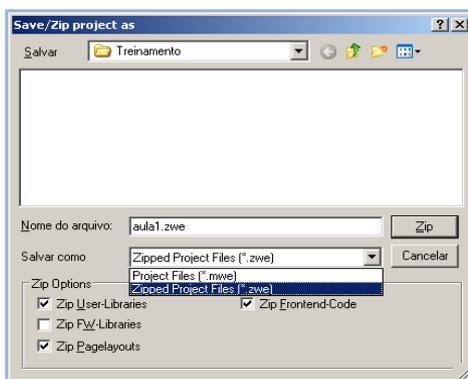
Será criado o arquivo do projeto e um diretório com o mesmo nome, que contém arquivos relacionados ao programa, gerados automaticamente.

Nome	Tam...	Tipo
aula1		Pasta de arquivos
aula1.mwe	4 KB	PC WORX project

Da mesma forma, quando realizado o armazenamento no formato *.zwe, o arquivo de cabeçalho e diretório são combinados em um arquivo *.zwt (com novo nome, se necessário).



O Projeto também pode ser salvo em modo compactado com a extensão *.zwt



O arquivo do projeto e respectivo diretório são unificados com a extensão *.zwt

Nome	Tam...	Tipo
aula1.zwe	256 KB	PC WORX project



O PC Worx sempre trabalha com a extensão *.mwt e com o respectivo diretório no formato descompactado. Assim, toda vez que um arquivo *.zwt é aberto, automaticamente é descompactado o seu conteúdo no diretório em uso.

Nome	Tam...	Tipo
aula1.zwe	256 KB	PC WORX project

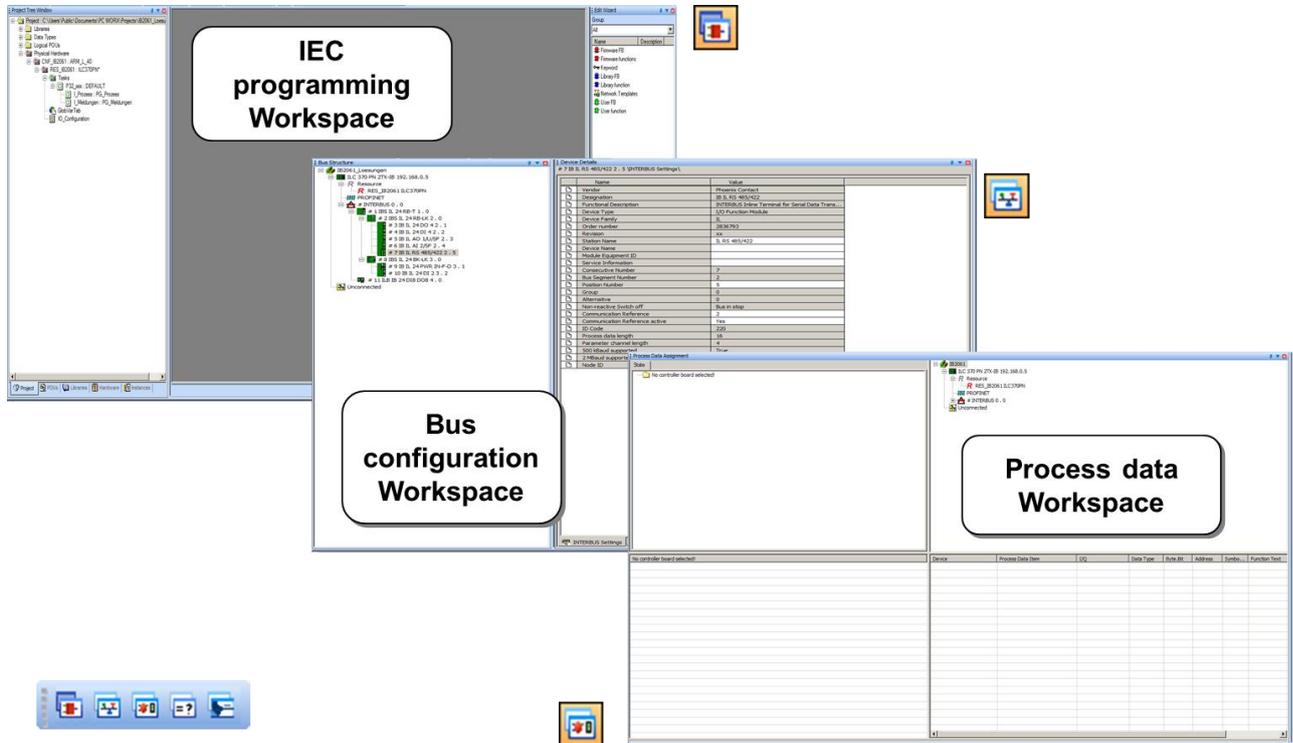
Nome	Tam...	Tipo
aula1		Pasta de arquivos
aula1.mwe	4 KB	PC WORX project
aula1.zwe	256 KB	PC WORX project

Exercício:

Crie e salve um novo projeto.

3. Áreas de Trabalho

As designações e janelas mostrados a seguir como padrões são às principais funções do PC WORX. Ou seja, isto aplica-se em três áreas de trabalho principais. A barra de menus garante rápida transição entre as diferentes áreas.



IEC programming Workspace: programação

Programação baseada na IEC 61131 (programas, blocos de função e funções) para a criação de tarefas do CLP, declarando os tipos de dados definíveis pelo usuário, integrando bibliotecas e outras funções relacionadas ao software.

Bus configuration Workspace: configuração de Hardware

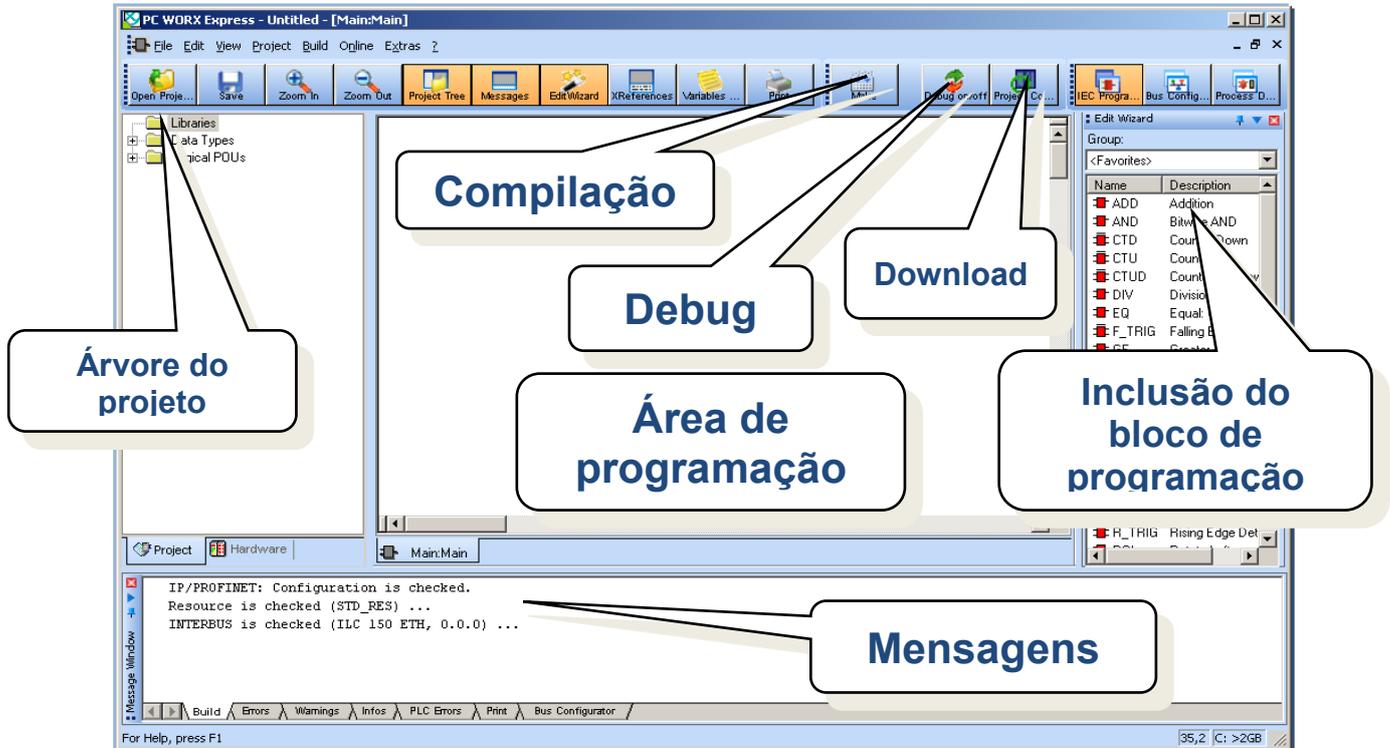
Criação dos sistemas de rede suportados pelo controlador (INTERBUS e PROFINET), edição dos dispositivos utilizados e gerenciamento de dados do dispositivo em geral.

Process Data Workspace: associação de variáveis

Neste ambiente é efetuada a associação das interfaces de entrada e saída dos cartões (analógicos ou digitais) adicionados à controladora como variáveis dos programas e/ou variáveis das respectivas bibliotecas.

3.1. IEC programming Workspace

Programação baseada na IEC 61131 (programas, blocos de função e funções) para a criação de tarefas do CLP, declarar os tipos de dados definíveis pelo usuário, integrar bibliotecas e outras funções relacionadas ao software.



A área de trabalho (workspace) para programação é baseada na IEC 61131 e é dividida entre a árvore do projeto (project tree window) e o assistente de edição (edit wizard). Além de uma exibição completa do projeto, a árvore do projeto oferece uma visualização específica para a unidade de organização do programa (POU), bibliotecas, tarefas (tasks) bem como a estrutura para a chamada das funções “chamar função” dos programas no controlador (instances).

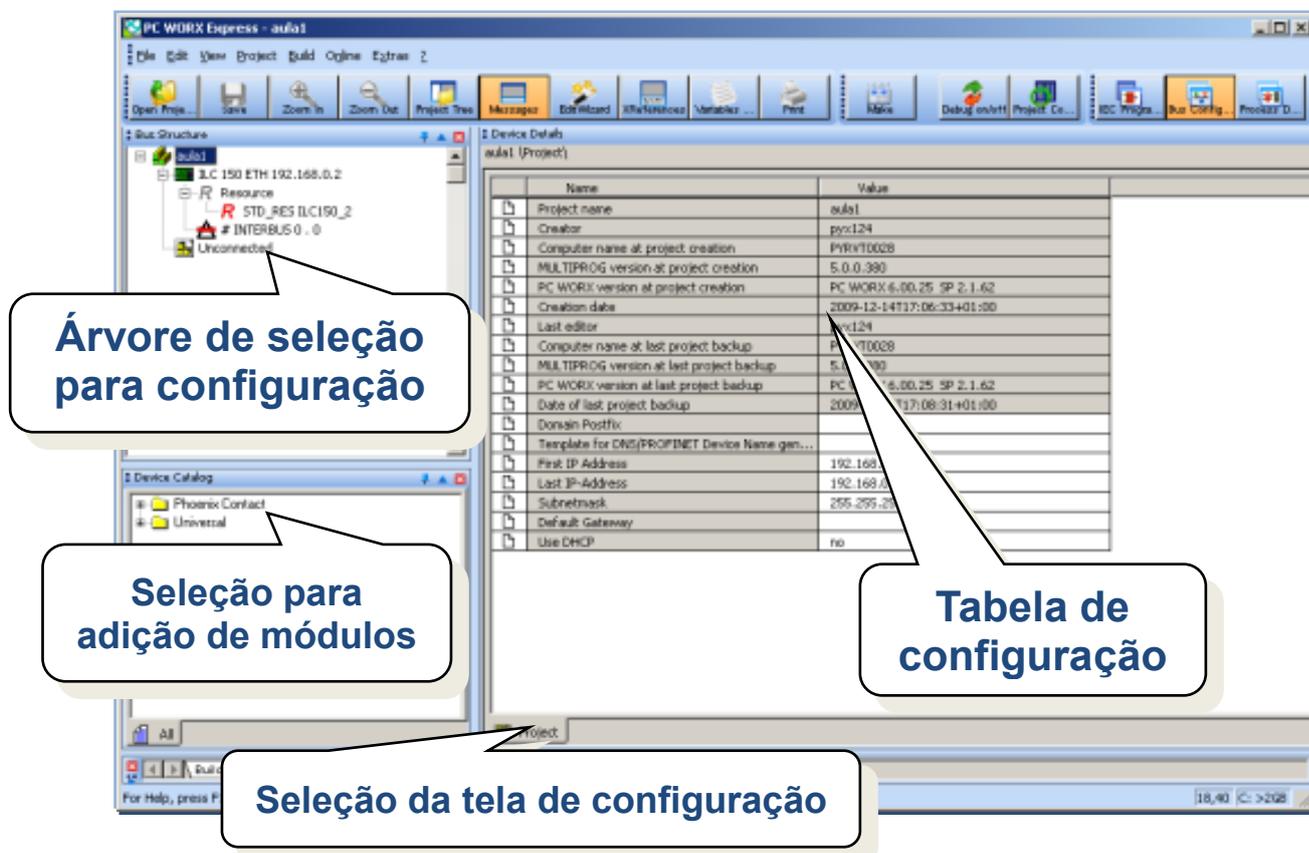
O assistente de edição é sensível ao contexto. Isso significa que, dependendo do elemento “em trabalho”, o assistente oferece ajuda para a criação de *data types* definidos pelo usuário ou na criação de um programa. A área de trabalho central (cor cinza na imagem) é utilizada para as worksheets abertas (code worksheets e tabelas de variáveis [variable tables]).

Assim como em muitas outras aplicações, estas janelas podem ser organizadas em diferentes estilos. Para maximizar a área de trabalho, pode-se esconder a árvore do projeto, o assistente de edição e a janela de mensagens (que iram aparecer automaticamente durante o processo de compilação). Os atalhos pré-definidos são:

- Shift+F8 Mostrar/Esconder a arvore de projeto (project tree)
- Ctrl+F2 Mostrar/Esconder a janela de mensagens (message window)
- Shift+F2 Mostrar/Esconder o assistente de edição (edit wizard)

3.2. Bus Configuration Workspace

Criação dos sistemas de rede suportados pelo controlador (INTERBUS e PROFINET), edição dos dispositivos utilizados e gerenciamento de dados do dispositivo em geral.



Normalmente, em um novo projeto a configuração de hardware é implementada, inicialmente, na workspace de configuração do barramento, que se trata do ambiente de trabalho inicial para o usuário. Como padrão, as janelas. *Bus configuration*, *Device details* and *Device catalog* são mostrados.

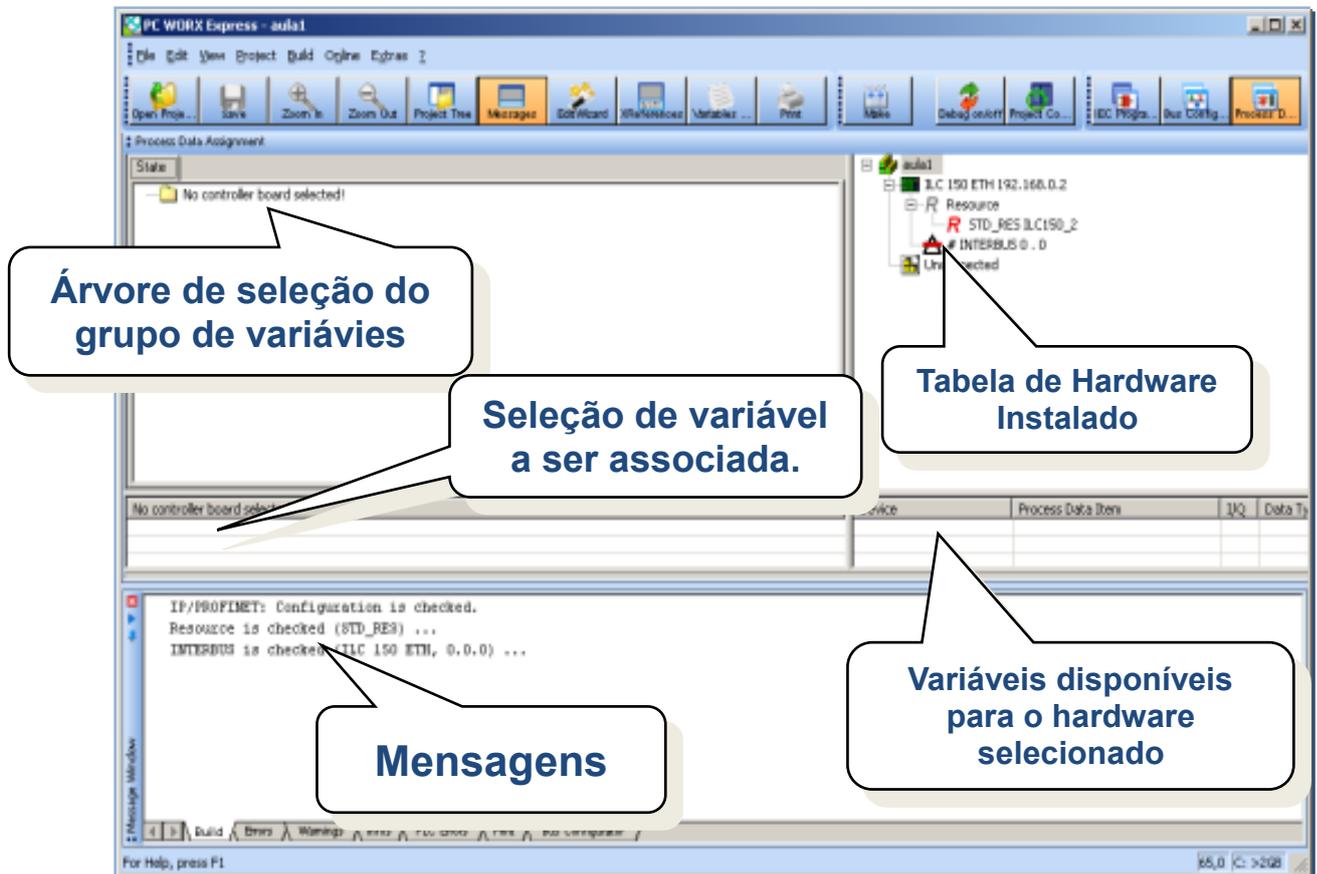
Os detalhes do dispositivo estão diretamente relacionados na configuração do barramento. O elemento selecionado na configuração do barramento (bus configuration) é apresentada com os detalhes do dispositivo.

Dependendo do tipo de elemento (projeto, diretório do barramento ou dispositivos de barramento), a quantidade e os tipos de guias são ajustados nesta janela "*Device details*". As informações sobre as páginas que precisam ser exibidas são geradas a partir do dispositivo ou do sistema de arquivos. Isso explica, em alguns casos o *delay* para a exibição. O arquivo de texto é "traduzido" e exibido de forma amigável na aba *Device details*.

Após a configuração do sistema de barramento, informações adicionais ou complementares sobre o hardware podem ser inseridas no *Device details* (exemplo: identificação do equipamento, nome da estação, etc.).

3.3. Process Data Workspace

Neste ambiente é efetuada a associação das interfaces de entrada e saída dos cartões (analógicos ou digitais) adicionados à controladora como variáveis dos programas e/ou variáveis das respectivas bibliotecas.



O *Process data workspace* é dividido em quatro quadrantes. O lado esquerdo (*Symbol/Variable*) mostra o PLC e sua estrutura. Dependendo do elemento selecionado, as variáveis globais correspondentes são apresentadas na tabela da parte de baixo.

O lado direito apresenta o hardware do sistema de barramento. Dependendo do elemento selecionado, os itens correspondentes ao *Process data workspace* são exibidos:

- Para dispositivos simples: apenas os objetos do dispositivo selecionado
- Para os acopladores de rede: todos os *process data items* disponíveis em um respectivo ramo
- Para o controlador do sistema: todos os objetos dependentes deste controlador

Nesta janela, as variáveis de programa estarão interligadas aos objetos do *process data items* da topologia de rede conectada.

4. Configuração

O caminho de comunicação para a conexão do dispositivo de programação ao sistema de controle é selecionado na guia **Comunicação** quando o elemento de controle (ILC 131 ETH) é selecionado. Dependendo do tipo de sistema de controle, diferentes mídias estão disponíveis.

4.1. Interface Serial

Todo sistema de controle fornece uma interface serial para a qual um cabo de conexão IBS padrão ou o cabo de conexão com conector Mini-DIN podem ser usados.

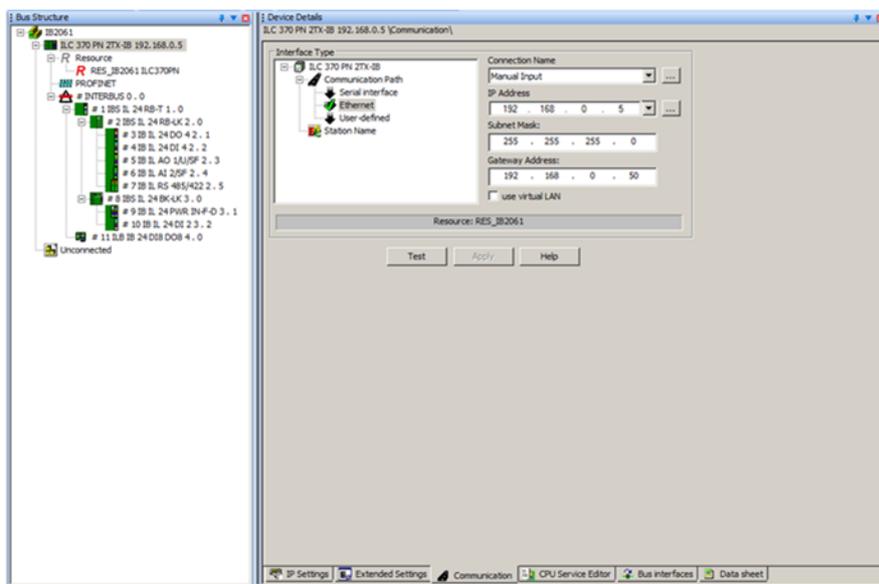
Ao pressionar o botão Test, é ativado um estabelecimento de conexão que é exibido na linha de status.

Através do botão **Aplicar**, você pode especificar o caminho de comunicação para o sistema de controle e projeto atual.

Ao usar sistemas de controle previamente configurados, a mensagem “Os nomes de conexão não coincidem” podem aparecer. Isto é para evitar um acesso acidental a um sistema de controle.

4.2. Rede Ethernet TCP/IP

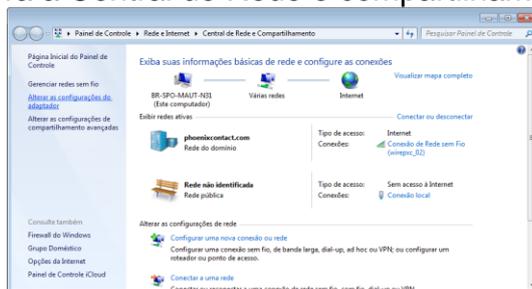
Além da interface serial, a maioria dos sistemas de controle oferece a possibilidade de se comunicar via TCP/IP. Para este tipo de comunicação, o endereço IP deve ser salvo no sistema de controle



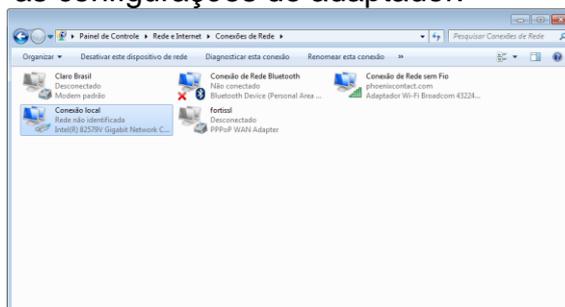
Antes de iniciar a configuração do endereço IP da controladora, certifique-se que o computador utilizado esteja na mesma família de endereços.

4.2.1. Configuração do Windows

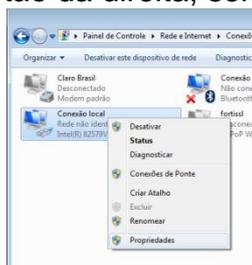
No Windows 7, abra a Central de Rede e compartilhamento:



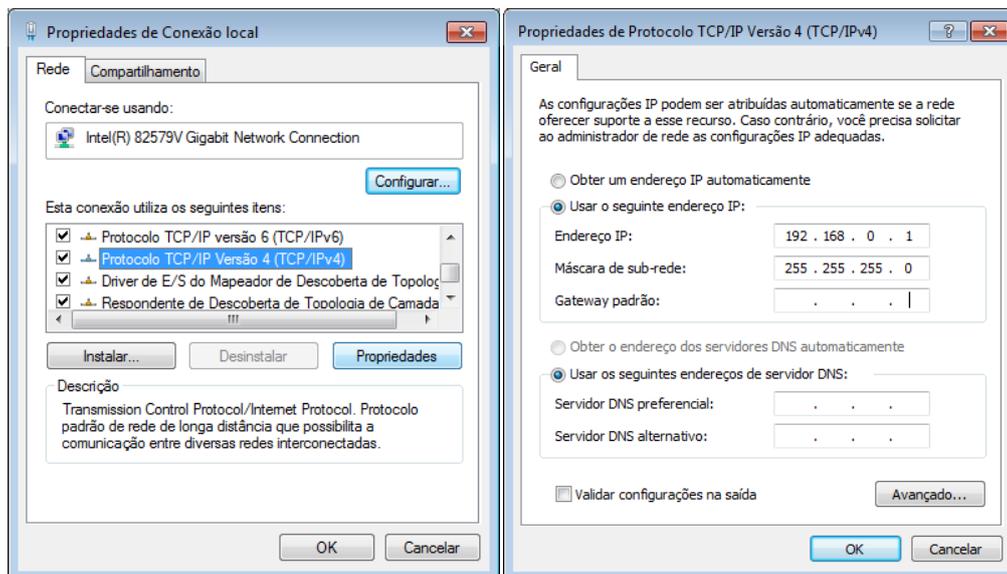
Selecione Alterar as configurações do adaptador:



Na Conexão local, com o botão da direita, selecione Propriedades:

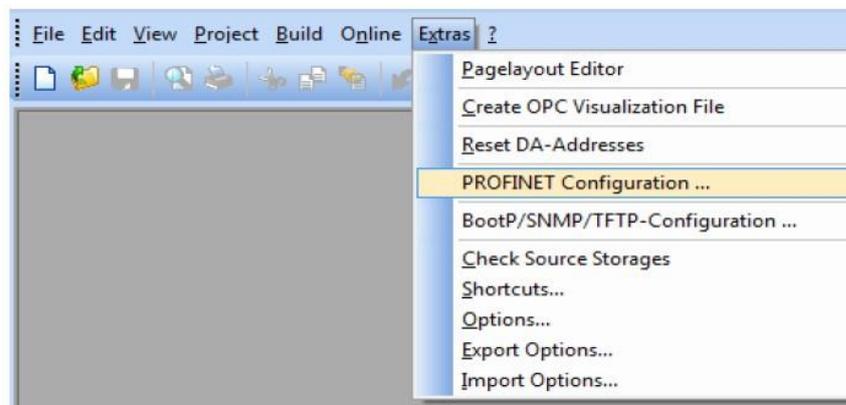


Selecione Protocolo TCP/IP Versão 4 e clique em propriedades e altere os campos conforme o quadro abaixo:



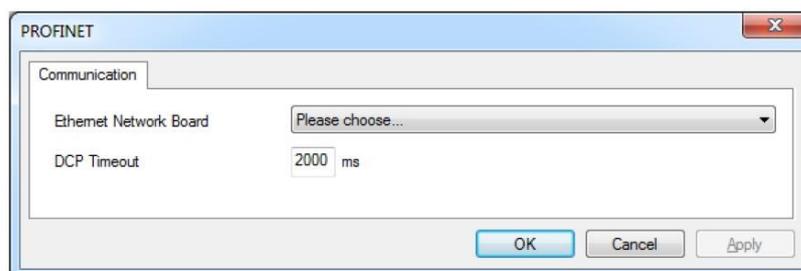
Configuração do IP da Controladora via PCWorx

Agora, antes que o endereço IP possa ser atribuído através do protocolo DCP PROFINET, você deve primeiro selecionar a placa de rede Ethernet correta. No menu "Extras", selecione o comando "Configuração PROFINET"

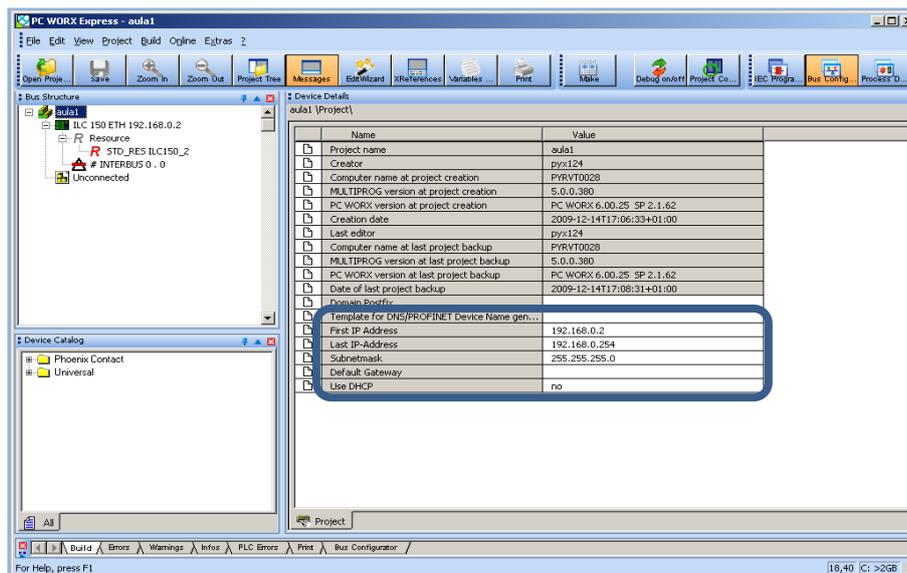


A janela "PROFINET" é aberta.

- Selecione a placa de rede Ethernet desejada.
- Mantenha a configuração de tempo limite DCP em 2000 ms.

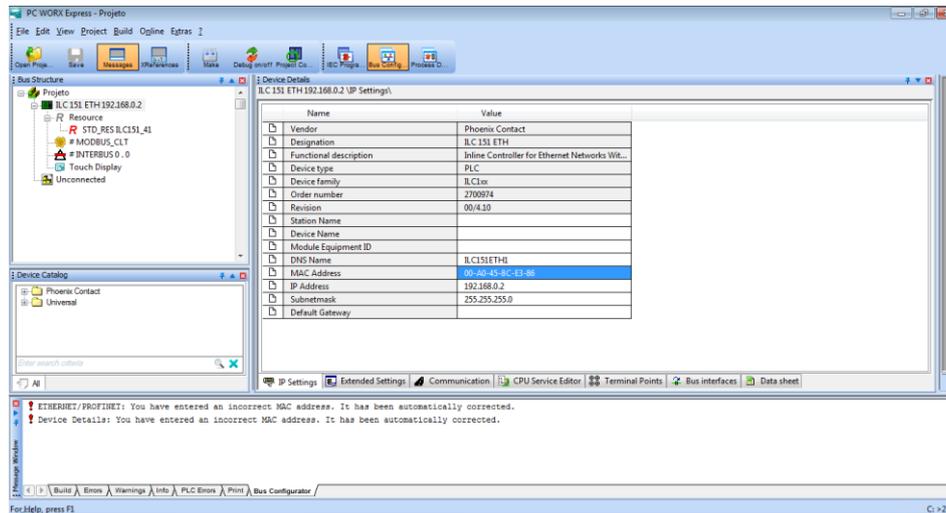


No Bus Configuration Workspace, clique na raiz do projeto e defina o intervalo dos endereços de rede TCP-IP, máscara de rede, e gateway, se aplicável:

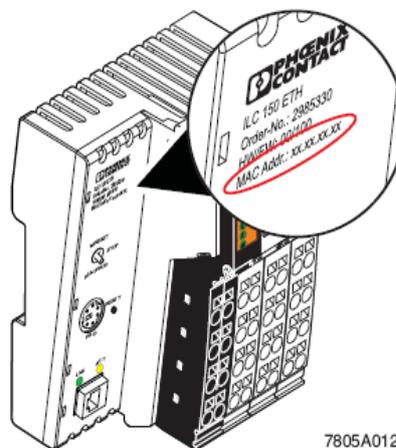


O endereço IP é atribuído através do protocolo DCP PROFINET no software PC WORX EXPRESS através da janela "Detalhes do dispositivo":

- Selecione o controlador na Bus Configuration Workspace.
- Selecione a guia "Configurações de IP".
- Insira o MAC ADDRESS, endereço IP desejado e máscara de rede. PCWorx define por padrão o endereço IP 192.168.0.2 e máscara de rede 255.255.255.0:



O MAC ADDRESS pode ser identificado conforma a imagem abaixo:

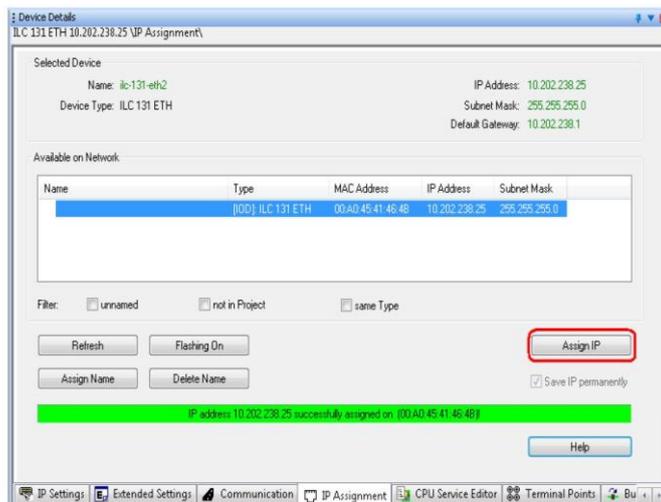


A guia "Atribuição de IP" é usada para atribuição de endereço IP real com DCP.

- Selecione a guia "Atribuição de IP".

Depois de selecionar a guia "Atribuição de IP", caso a rede PROFINET não seja encontrada:

- Desative a caixa de seleção na área "Filtro".
- Selecione o controlador.
- Clique em "Atribuir IP" para iniciar a atribuição de endereço IP com DCP



Em seguida, verifique a comunicação entre o PC WORX EXPRESS e o ILC 131 ETH:

- Na janela "Detalhes do dispositivo", selecione a guia "Comunicação".
- Clique no botão "Testar".

Se a comunicação não puder ser estabelecida entre PC WORX EXPRESS e ILC 131 ETH, verifique novamente se sua rede permite as configurações de IP usadas no projeto de exemplo.

A atribuição de endereço IP através de um servidor BootP é uma alternativa à atribuição de endereço IP através do protocolo DHCP PROFINET. Em uma rede Ethernet, o BootP é usado para atribuir um endereço IP a um cliente BootP usando um servidor BootP.



Para este exemplo (por padrão para o ILC 131 STARTERKIT), o ILC 131 ETH (cliente BootP) envia um BootP_Request como uma transmissão para a rede. O endereço MAC do transmissor é enviado com o BootP_Request para fornecer uma identificação exclusiva. Se o servidor BootP foi ativado no PC WORX EXPRESS, o PC WORX EXPRESS responde com um BootP_Reply. PC WORX EXPRESS usa este BootP_Reply para informar o ILC 131 ETH do seu endereço IP e máscara de sub-rede.

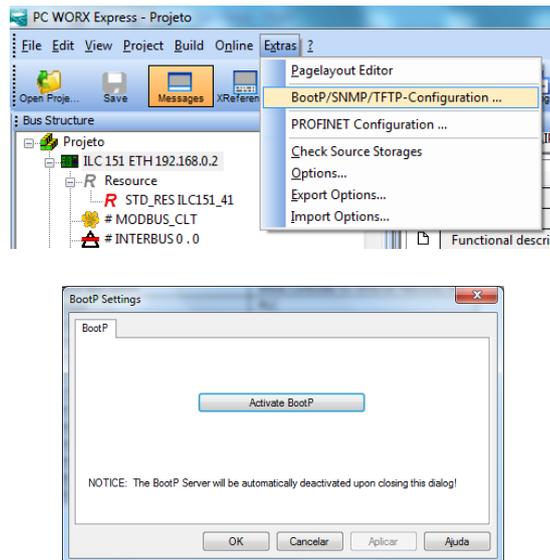
Certifique-se de que:

- O servidor BootP conhece o endereço MAC enviado pelo cliente BootP.
- Um endereço IP e uma máscara de sub-rede correspondentes foram atribuídos em PC WORX EXPRESS para o endereço MAC.
- Uma vez que os dados IP foram transferidos para o ILC 131 ETH com sucesso, o PC WORX EXPRESS envia uma mensagem de confirmação correspondente.

As configurações de IP para o controlador são feitas quando o projeto é criado. Para configurar o endereço IP no PC WORX EXPRESS, proceda conforme descrito abaixo:

- Estabeleça uma conexão Ethernet entre o seu PC e o controlador.
- Mude para o espaço de trabalho da Configuração de barramento.
- Selecione o *Inline controller node*.
- Selecione a guia "Configurações IP" na janela "Detalhes do dispositivo".
- Digite o endereço MAC do controlador no dispositivo. Começa com 00.A0.45.

Ative o BootP:



BOOTP, abreviação do protocolo Bootstrap, é um protocolo usado para permitir que um dispositivo de rede Ethernet obtenha um endereço IP na rede, o chamado direcionamento estático. Esse protocolo permite a alocação automática de endereços de rede, mas não é capaz de alocá-los dinamicamente, como faz o DHCP, sucessor do BOOTP.



Um dispositivo que deseja obter um endereço IP transmite uma solicitação de BootP que identifica o dispositivo por seu endereço MAC, um número de identificação de seis octetos (ex: 00: A0: 45: 08: CD: 8D) que é atribuído de forma exclusiva a um dispositivo.

Um servidor BootP na rede, vê a solicitação e envia uma resposta BootP contendo o endereço IP desejado (ex: 192.168.1.10) para o dispositivo, tornando-o agora acessível para comunicações de rede de nível superior usando esse endereço IP.

Cicle a alimentação da controladora:



Observe a associação de IP e feche a janela BootP Settings. O controlador recebe o endereço IP especificado no projeto para o controlador. A seguinte mensagem aparece na "Message Window" na guia "Bus Configurator":



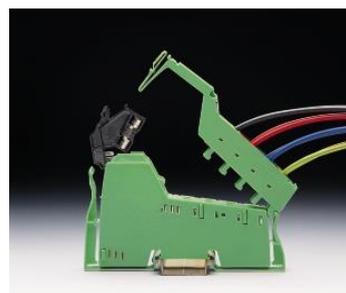
4.2.2. Configuração do IP da Controladora via software IPAssign

O Software IPAssign é gratuito e está disponível em <https://www.phoenixcontact.com/online/portal/pc>

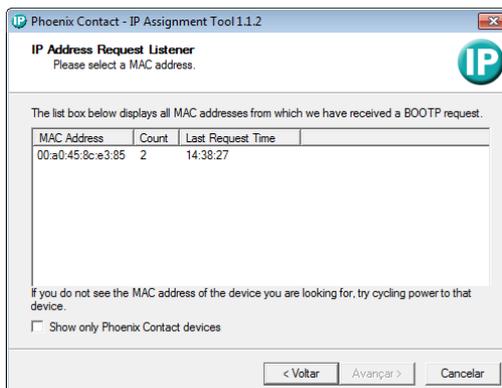
Execute o IPAssign e clique em avançar:



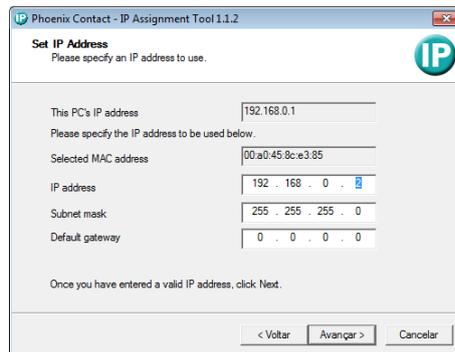
Cicle a alimentação da controladora:



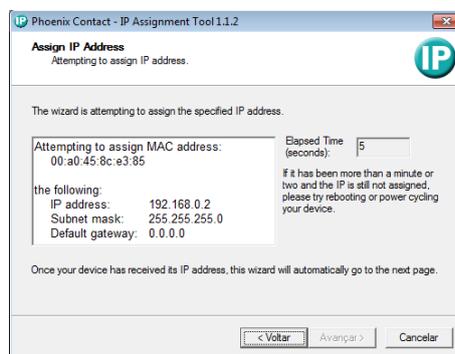
Selecione o MAC ADDRESS:



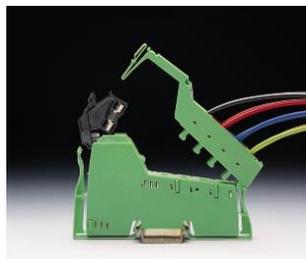
Defina o endereço IP desejado e a máscara de rede:



Aguarde um pouco a contagem...



Caso não associe automaticamente, cicle a alimentação da controladora:



A seguinte tela será exibida:

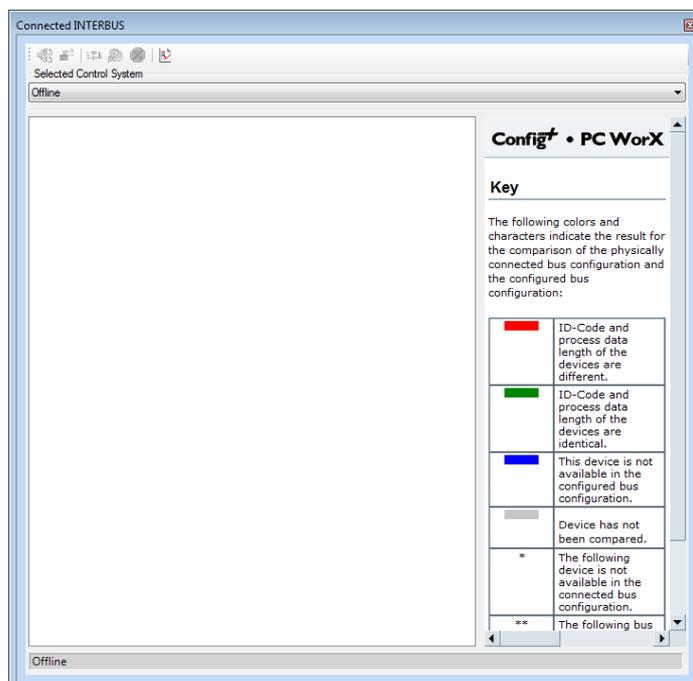
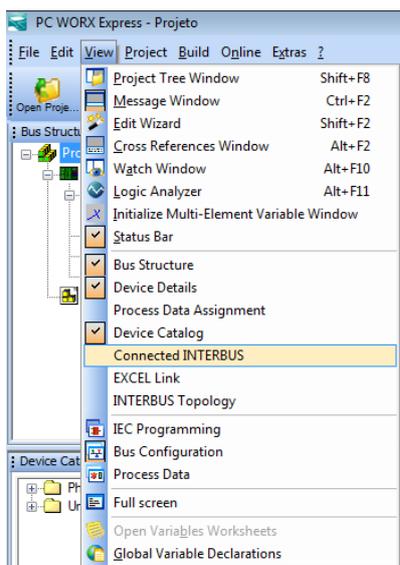


4.3. Adição de cartões de interface

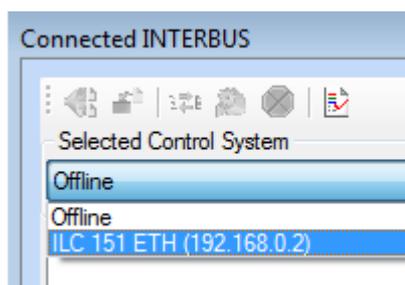
Nesta seção serão apresentados os modos de incorporação de cartões ao projeto.

4.3.1. Seleção automática:

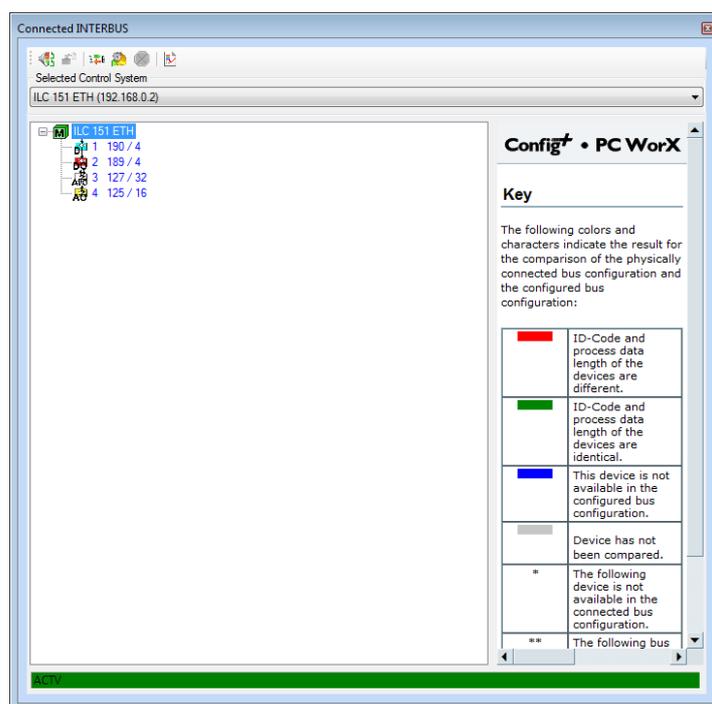
Abra o Connected Interbus no menu “View -> Connected Interbus”:



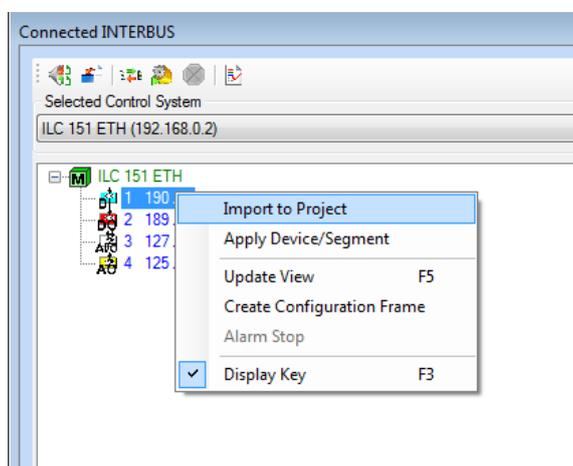
Entre on-line selecionando a controladora:



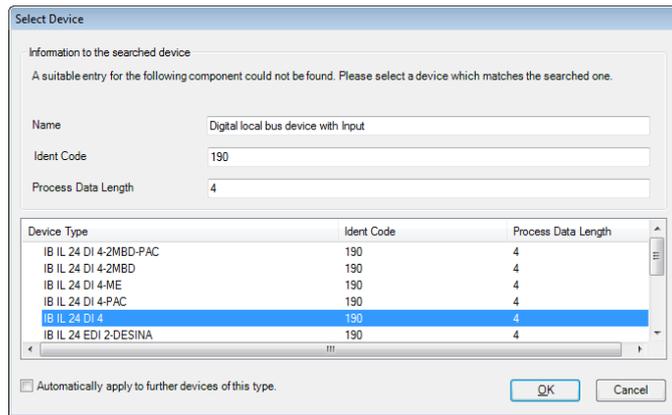
A seguinte tela será exibida:



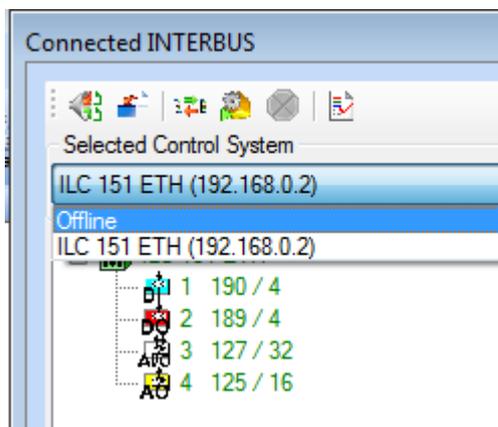
Click com o botão da direita do mouse em cima do primeiro cartão, e selecione “Import to Project”:



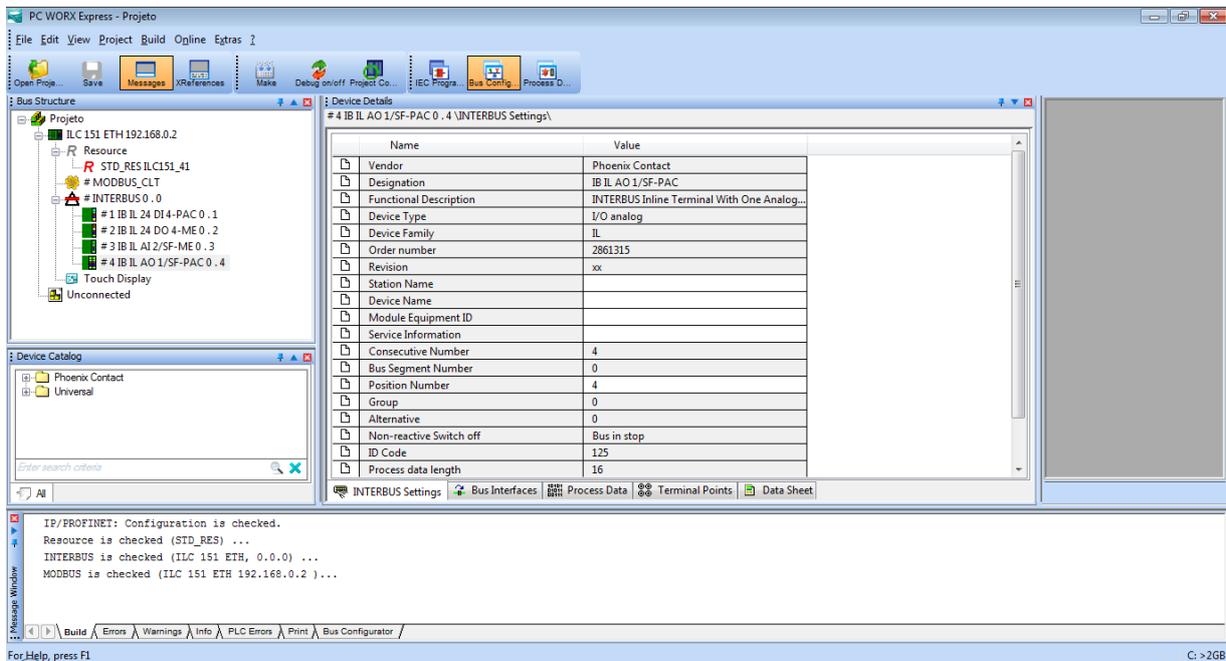
Obs.: Serão apresentadas telas de seleção, na ordem dos cartões instalados. Certifique-se de escolher corretamente os cartões:



Ao finalizar, lembre-se sempre de voltar ao modo "Offline":

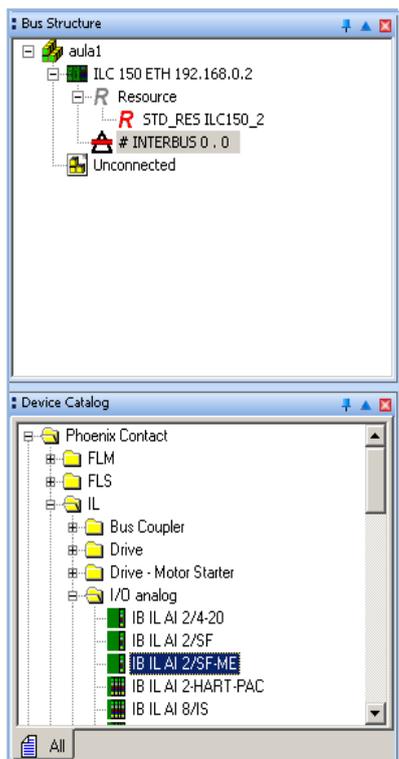


Visualize a estrutura no *Bus Configuration*:

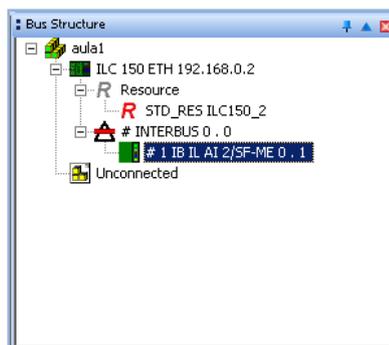


4.3.2. Seleção Manual:

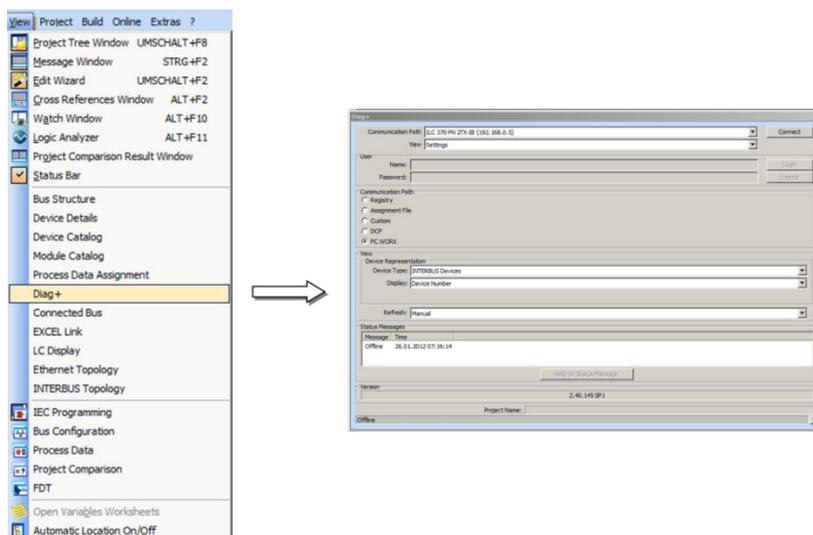
Selecione a raiz da rede *Interbus* e abra o *Device Catalog*. Na pasta IL (In Line) selecione a respectiva família de cartões, identifique o cartão, clique em cima dele e arraste até a rede *Interbus*:



Repita o procedimento para todos os cartões instalados:



O software Diag + 2.0 integrado no PC WORX PROFESSIONAL oferece uma extensa função de diagnóstico para INTERBUS e PROFINET, que são os sistemas de barramento suportados pelos sistemas de controle.



Exercício

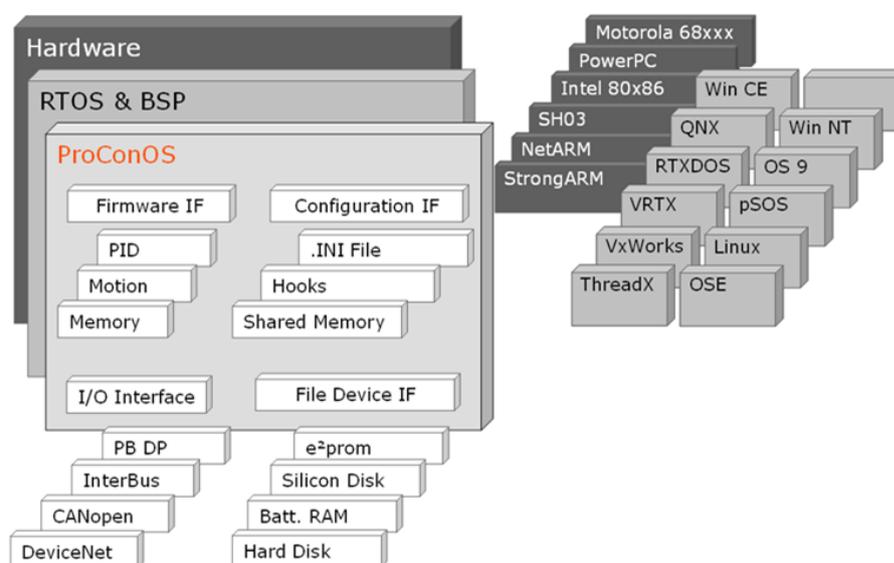
Em cima do projeto anteriormente criado, configure a rede Ethernet e depois adicione os cartões no modo automático.

5. Overview Arquitetura de Hardware

O ProConOS (Sistema Operacional de Controle Programável) é necessário para a prestação de serviços de sistemas típicos do CLP em plataformas de hardware específicas ou padrão.

Isso inclui o carregamento e o processamento de programas do CLP, bem como o fornecimento de funções de depuração (debug) para programação, instalação e manutenção de máquinas e sistemas controlados por CLP.

O ProConOS (ilustração do conceito na figura abaixo) é usado como um sistema operacional para a maioria dos sistemas de controle da Phoenix Contact. Isso oferece a vantagem de que todos os sistemas de controle podem ser parametrizados e programados no mesmo ambiente de desenvolvimento (PC WORX).

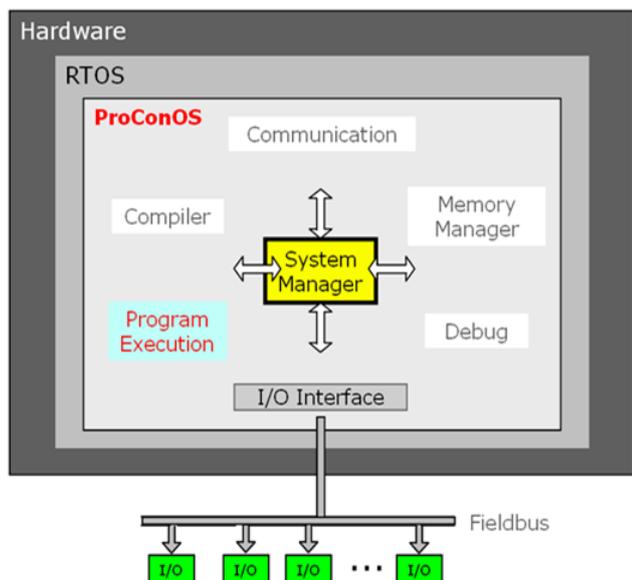


O ProConOS é baseado em sistemas operacionais multitarefa padrão que são controlados por meio de prioridades da tarefa. No ProConOS, existem três níveis de prioridade diferentes:

- Nível de prioridade para tarefas de usuário;
- Nível de prioridade para tarefas do ProConOS;
- Nível de prioridade para tarefas do gerenciador de sistema;

Esta classificação garante que o tempo de computação do processador seja sempre distribuído em favor das tarefas desenvolvidas. Este é o pré-requisito interno ProConOS para aplicações de tempo determinístico (dada uma certa entrada, será produzido sempre a mesma saída).

Portanto, deve-se trabalhar com o conceito de multitarefa na arquitetura do sistema ProConOS, que visa assim obter uma resposta de tempo calculável, otimização do desempenho, redução do tempo de resposta e a reação específica aos erros de tempo de execução.



As tarefas do usuário incluem tarefas cíclicas, tarefas de eventos e tarefas DEFAULT. Algumas observações:

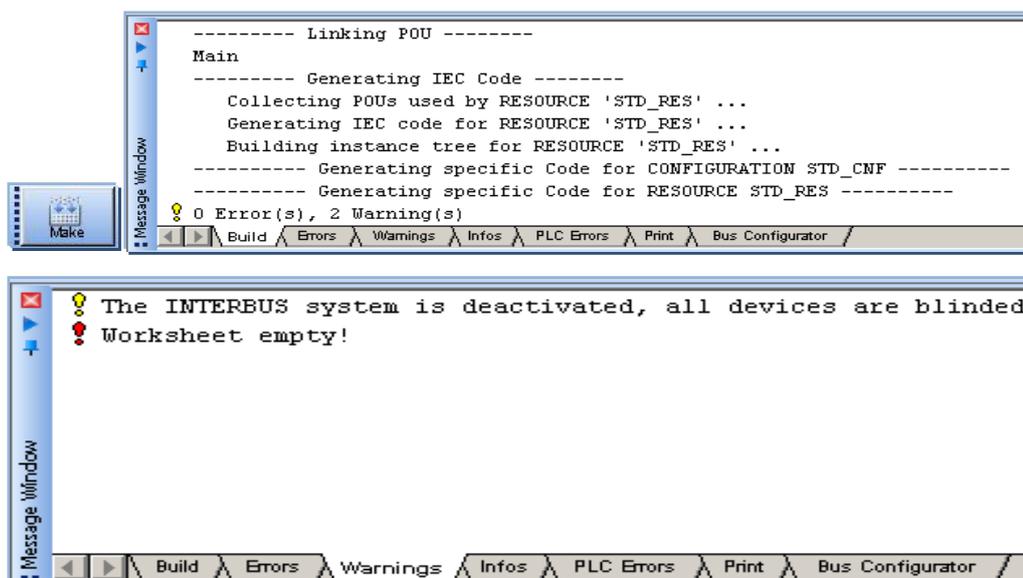
- Dentro de um determinado intervalo de tempo, as tarefas cíclicas são executadas ciclicamente de acordo com a prioridade definida pelo usuário.
- Uma reação a eventos ocorrendo acíclicamente, e interrupção de hardware, podem interferir nas tarefas de evento.
- A tarefa DEFAULT é a tarefa com a prioridade mais baixa e é executada quando nenhuma outra tarefa está ativa. A tarefa DEFAULT é uma tarefa cíclica.

6. Operação

6.1. Compilando um projeto

Os projetos em PCWorx para as controladoras da Phoenix Contact devem ser compilados antes de serem descarregados. Esta operação é realizada através do botão abaixo:

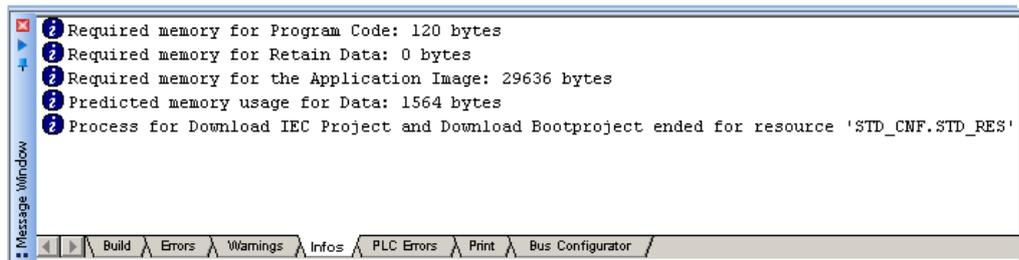
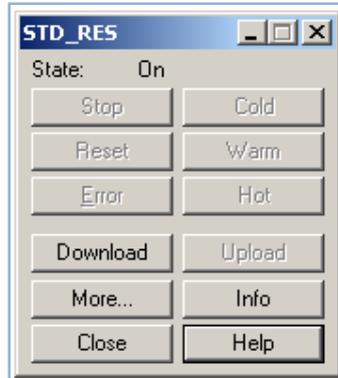
Ao compilar um projeto pela primeira vez, selecione o comando "Rebuild Project" no menu "Build" (opção para o PC Worx Professional).



Para outros processos de compilação, você também pode usar o comando "Make" Sempre quando um projeto for compilado é importante observar as mensagens no *Message Window*, para verificação de Erros e Alarmes (*Warnings*)

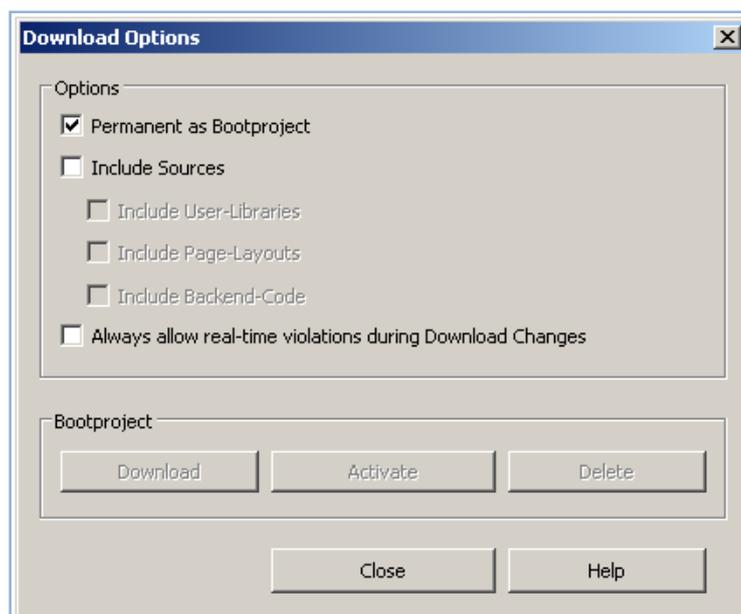
6.2. Download

Para descarregar o projeto na controladora, abra a tela “*Project Control Dialog*” através do botão abaixo e clique em Download:



Atenção: Se apenas o botão "Close" estiver ativado na tela *Project Control Dialog*, provavelmente não há conexão válida com o ILC 131 ETH. Nesse caso, verifique o caminho de comunicação.

Através do botão “More” são apresentadas algumas opções:



Após a bem-sucedida compilação do projeto (configuração do barramento e programa de aplicação), o código de máquina pode ser descarregado através do item Download. Esta opção armazena o código compilado na memória principal volátil (RAM).

Na opção "More" podem ser incluídos dois tipos de download que ocorrerão juntamente com o download para a memória RAM:

- *Permanent as Bootproject* – o projeto compilado será salvo também na memória flash do controlador.
- *Include Source* – o programa compilado será descarregado para a RAM do controlador com o código fonte, possibilitando executar um Upload.
- *Incluir bibliotecas de usuários* - Todas as bibliotecas de usuários incluídas estão incluídas no programa fonte.
- *Incluir Layouts de Página* - Todos os layouts de página que são usados estão incluídos no programa fonte.
- *Incluir código de backend* - Todos os dados do projeto fonte, incluindo a compilação, estão incluídos no programa fonte.
- *Always allow realtime violations during Download Changes* - Quando é selecionado, aplica-se o seguinte procedimento padrão: na próxima vez que o "Download Changes" for executado, o sistema não tenta observar as condições em tempo real (isto é, a execução da tarefa do usuário no horário exato) no controlador. Em vez disso, o sistema para a execução do programa no controlador, transmite o projeto, copia todos os dados instanciados e, em seguida, retoma a execução do programa modificado.

Área "Bootproject"

- Botão "Download": o projeto compilado é enviado separadamente para o controlador.
- Botão "Ativar": o projeto compilado salvo no controlador está ativado. Após a ativação, o projeto compilado pode ser executado.
- Botão "Excluir": o projeto compilado salvo no controlador é excluído.



Observação: Somente após o download do arquivo fonte (*Include source*), o projeto pode ser relido da memória de parametrização. Não é possível decodificar o código da máquina.

6.3. Execução do projeto

O programa pode ser iniciado por 3 modos: *Cold*, *Warm* e *Hot*



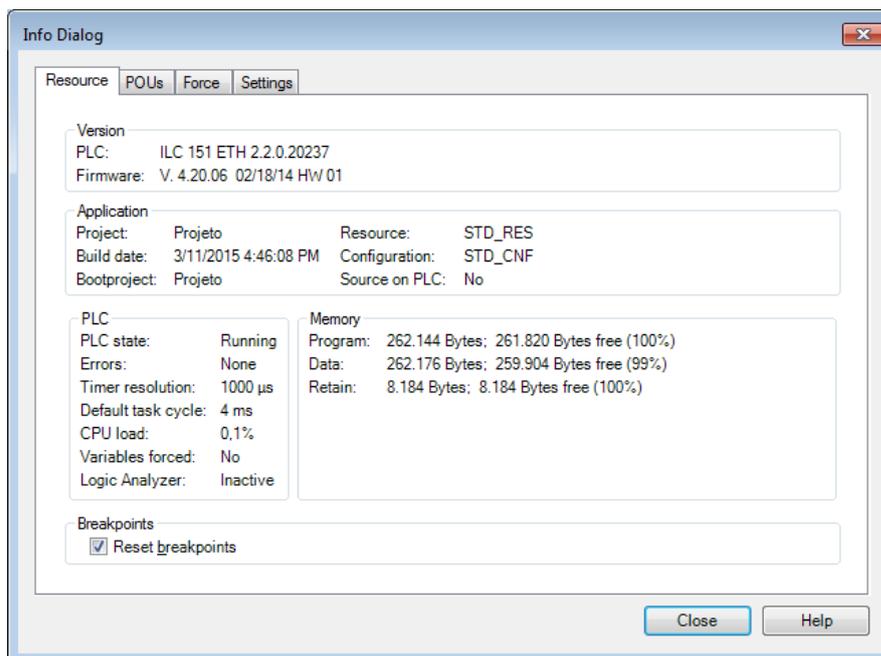
Cold – O programa inicia sem manter nenhum valor de variável

Warm – Mantém os valores das variáveis retentivas

Hot – Mantém os valores das variáveis retentivas e discretas

6.4. Análise

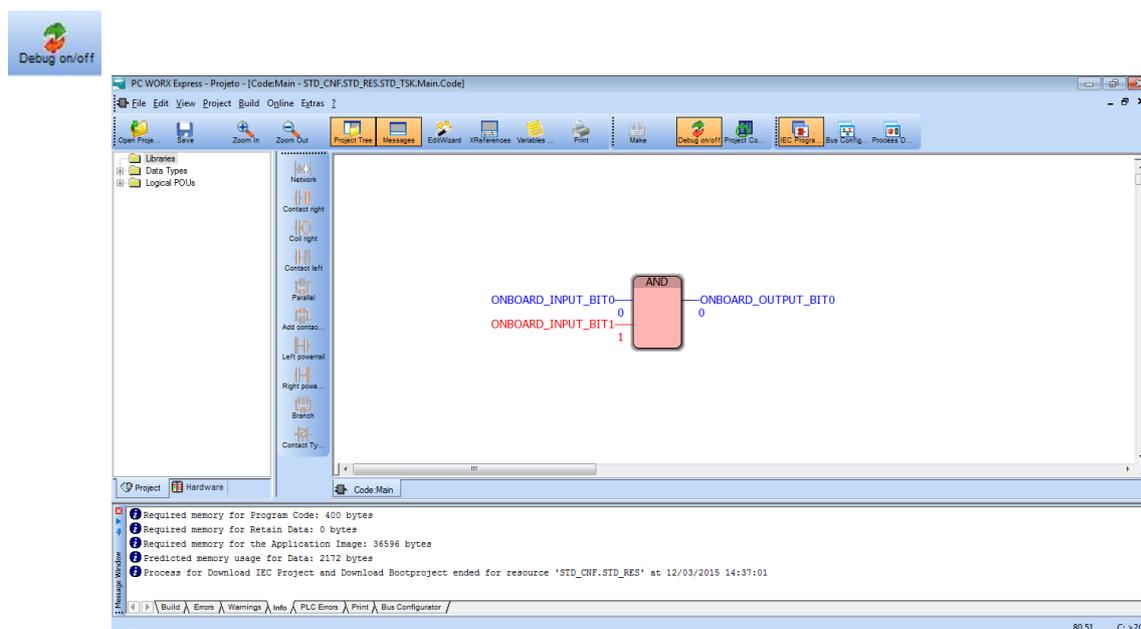
Através do botão Info do *Project Control Dialog*, é acessada a tela de informações:



Esta tela é importante para visualizar a carga da CPU, tempo de ciclo da TASK DEFAULT e utilização da memória.

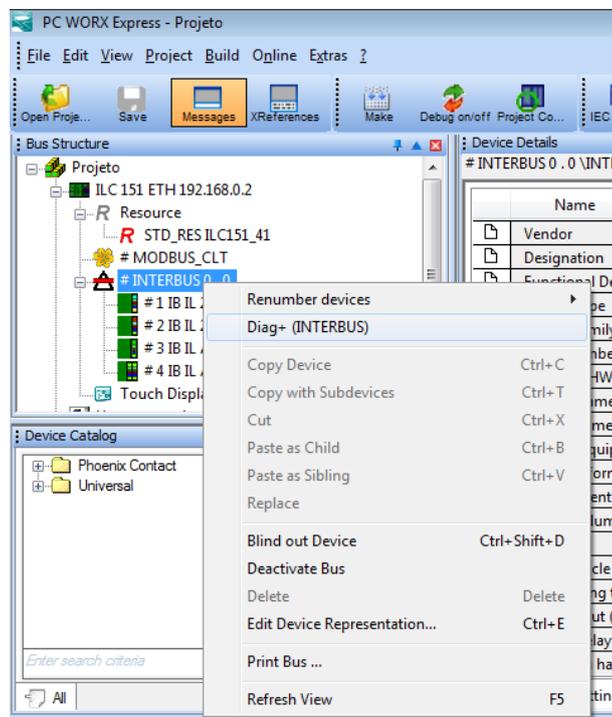
6.5. Debug

Uma vez que o projeto foi compilado e descarregado, é possível visualizar o estado das variáveis através do botão abaixo:



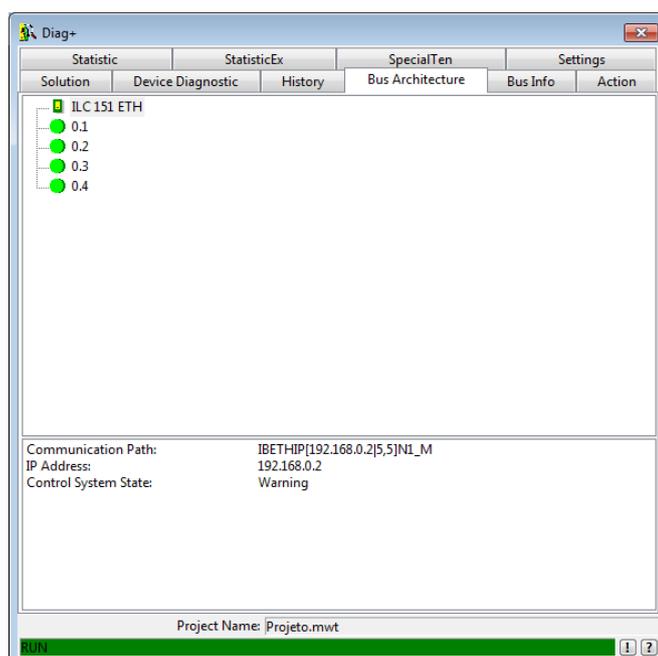
6.6. Diagnóstico

Para acessar a tela de diagnóstico, vá ao *Bus Configuration Workspace*, clique com o botão direito do mouse em *InterBus* e selecione a opção “Diag+ INTERBUS”.

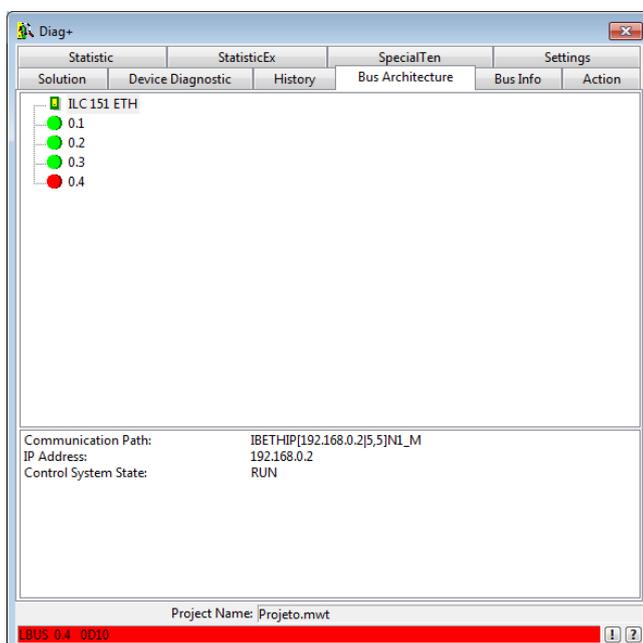


As Principais telas são “*Bus Architecture*”, “*Solution*” e “*Action*”

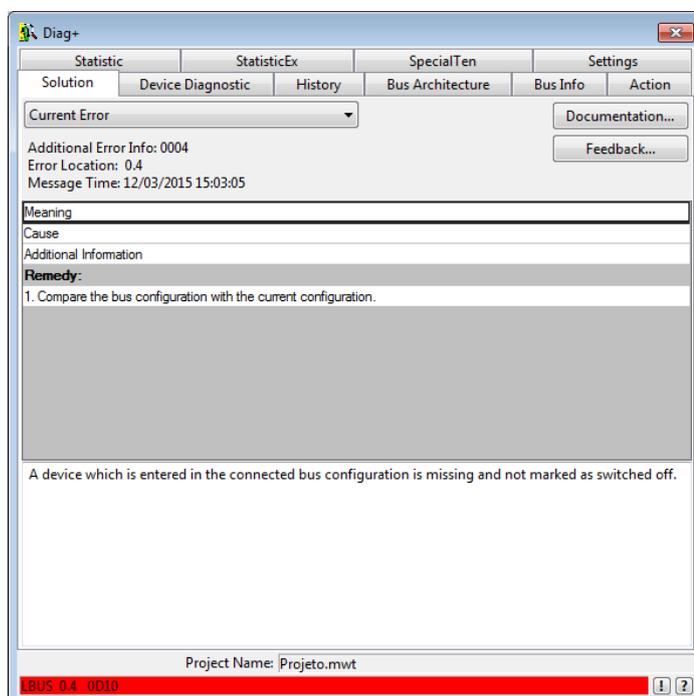
Na tela “*Bus Architecture*”, é possível visualizar o estado dos cartões instalados. Caso todos estejam funcionando corretamente, estarão acompanhados de um símbolo em verde:



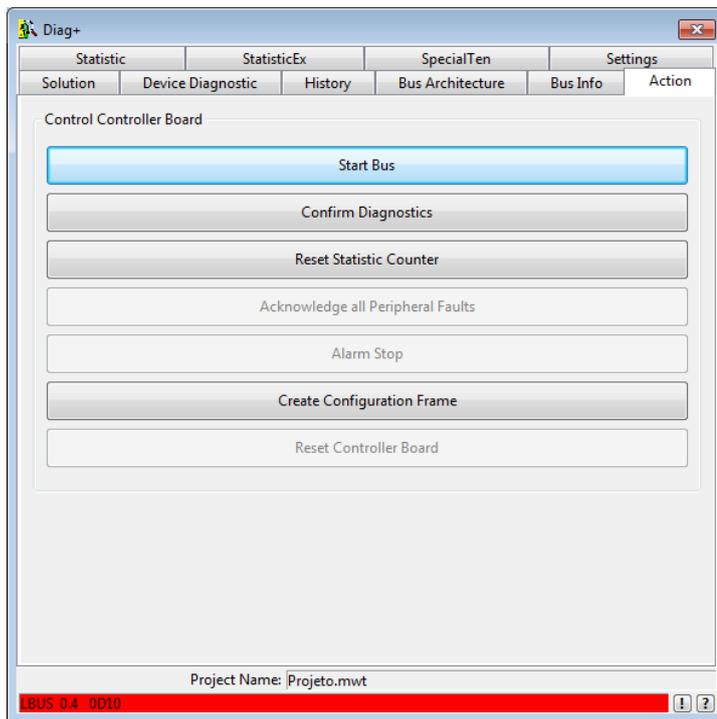
Caso haja algum problema com algum dos cartões, estes estarão acompanhados de um símbolo em vermelho:



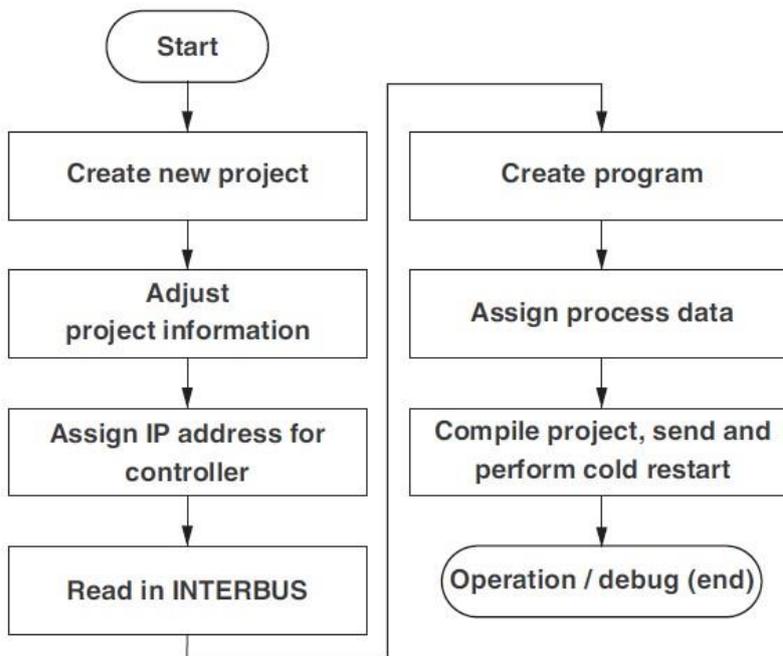
Na tela “*Solution*”, é possível visualizar possíveis soluções para o problema:



Na tela “Action”, pode-se confirmar o diagnóstico e tentar reiniciar a rede *Interbus*:



6.7. Sequencia de Desenvolvimento de um Projeto

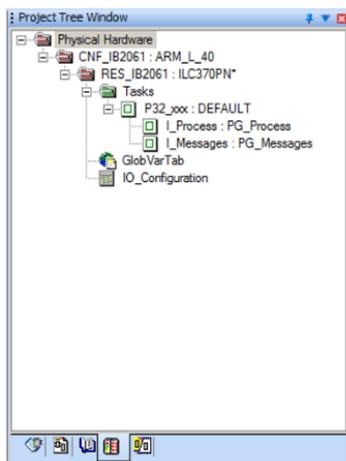


Exercício:

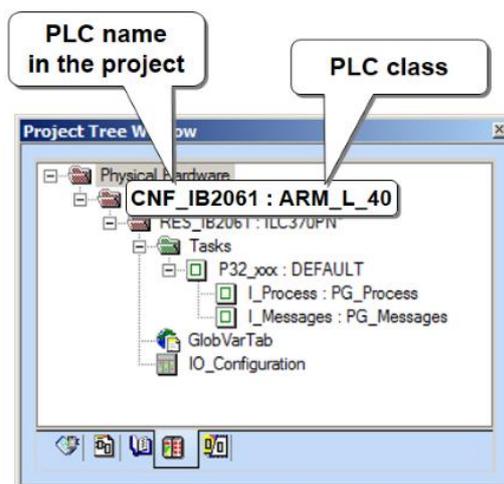
Com o projeto até então criado, acesse o Diag+ e visualize o estado da rede. Tente fazer “*Make - Download*” com a controladora na posição “*Stop*”.

7. Estrutura do projeto no PC WORX

No PC WORX, a estrutura de hardware do CLP consiste: do rack do CLP, do processador do CKP e do gerenciamento da capacidade do processador. A parte da árvore do projeto “*Project Tree Window – IEC Programming Workspace*”, para a qual a estrutura do hardware é salva, pode ser vista em uma guia separada.



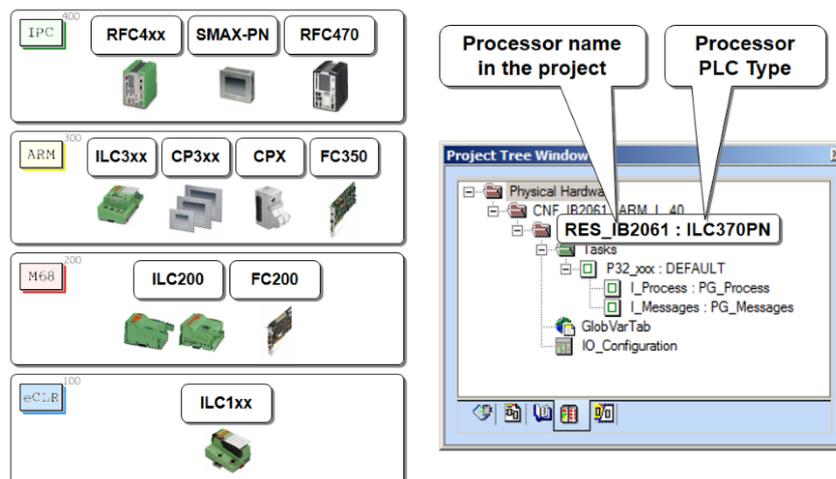
7.1. Configuração dos Recursos



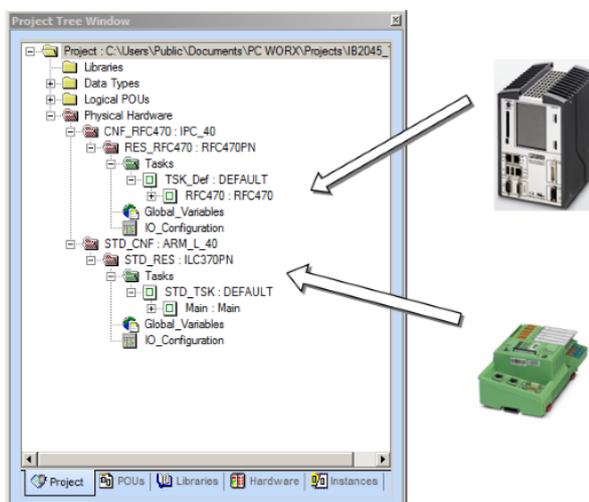
De acordo com a IEC 61131, o elemento de nível superior da estrutura de hardware para um CLP é chamado de configuração (CNF). Para sistemas de controle clássicos, isso corresponde ao rack do CLP e serve para a classificação da classe de desempenho.

Quatro classes de desempenho diferentes estão disponíveis no PC WORX, e podem ser observadas na aba “*Hardware*” da tela “*Project Tree Window*”. Classe 400 conta com processadores INTEL. Na classe 300, os processadores ARM são usados, e para os sistemas de controle da classe 200 os processadores Motorola são usados. O eCLR conta com sistemas de controle de classe 100, que utilizam o eCLR.

Devido à possibilidade de trabalhar com vários sistemas de controle, além da classe de desempenho do sistema de controle, um nome deve ser atribuído. Os modelos de projeto no PC WORX como padrão usam o nome STD_CNF (configuração padrão). O termo Configuração usada aqui não está de modo algum vinculado à configuração INTERBUS.



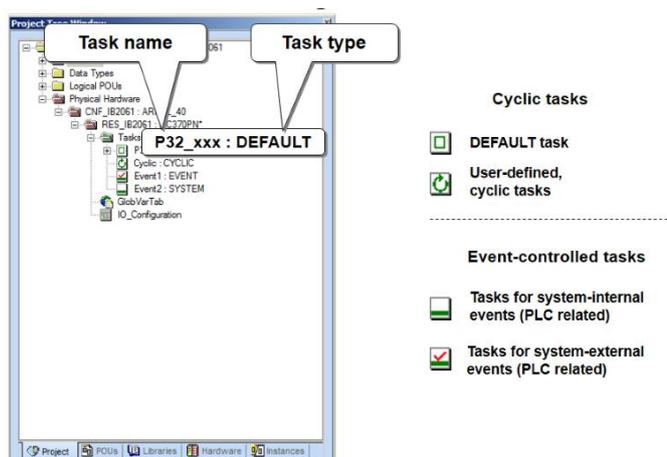
Tal como acontece com a classe de desempenho, além de inserir o tipo de processador, um nome para o elemento *Resource* deve ser atribuído. Os modelos de projeto no PC WORX como padrão usam o nome STD_RES (recurso padrão), conforme detalhes da figura acima.



Quanto ao hardware na configuração do barramento e ao CLP, o PC WORX suporta vários sistemas. No entanto, esta documentação apenas se refere a um sistema de controle por projeto.

Os programas criados na parte superior da árvore do projeto podem ser atribuídos a diferentes configurações distribuídas e, portanto, sistemas de controle. Os sistemas de controle usam os mesmos algoritmos, mas processam-se individualmente (condição proporcionada pela task).

7.2. TASK

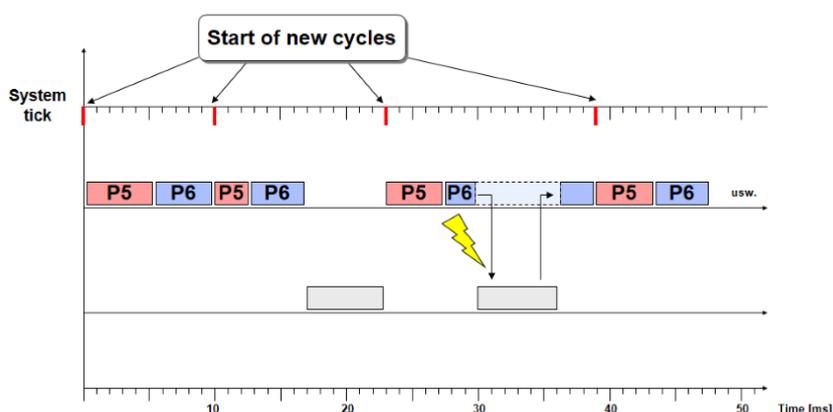


As tarefas “Task” são divididas em duas classes: tarefas cíclicas e controladas por eventos, como ilustrado na figura acima.

O processo em um CLP geralmente apresenta execução cíclica. A TASK DEFAULT como uma tarefa cíclica, portanto, pode ser usada para a maioria das aplicações.

Ela funciona com a menor prioridade e, como tempo de ciclo mínimo, oferecendo um $t_{min} = 2 * tick$ do sistema. Para processadores IPC e ARM, o tick do sistema é definido como 1 ms, para os processadores M68 e eCLR é definida em 5 ms. A tarefa DEFAULT só pode ser criada uma vez, por sua baixa prioridade e pode ser interrompida por qualquer outra tarefa.

O tempo de ciclo da TASK DEFAULT corresponde ao tempo de resposta da maioria dos controladores: Diretamente após a execução de um ciclo, o próximo é iniciado. Não há intervalos fixos. Devido à programação, tempos de ciclo podem flutuar significativamente. A interrupção da tarefa padrão (TASK DEFAULT) por uma tarefa do servidor (debug ou comunicação) também resulta em variações no tempo de ciclo (figura a seguir).



Por outro lado, as tarefas definidas pelo usuário (CYCLIC TASKS) oferecem a possibilidade de selecionar livremente prioridade e tempo de ciclo dentro do escopo das condições técnicas. A prioridade pode ser definida entre 0 (mais alto) e 31 (mais baixo).



A segunda classe de tarefas cobre as tarefas SYSTEM e EVENT (EVENT-CONTROLLED TASKS). As tarefas do sistema "SYSTEM TASKS" permitem uma reação aos eventos internos do CLP (por exemplo, reinício a frio, início a quente, divisão por 0 etc.). Isso inclui estados operacionais regulares e irregulares do sistema de controle.

Cabe ressaltar que TASKS do tipo SYSTEM e EVENT dependem da família de controladoras aplicada. Podem executar programas em função de um evento como COLD/HOT/WARM START, dependendo da família da controladora, definidos pelo usuário, entre outros. Entretanto, no PCWorx Express só é possível trabalhar com a TASK DEFAULT (CYCLIC TASKS).

8. Program Organization Units (Unidades de Organização de Programas)

8.1. Linguagens suportadas

A IEC 61131-3 suporta as seguintes linguagens de programação:

- FBD: *Function Block Diagram* - Diagrama de Blocos
- LD: Ladder
- ST: *Structured Text* - Texto Estruturado
- IL: *Instruction List* - Lista de Instruções
- SFC: *Sequential Functional Chart* - Fluxograma Seqüencial

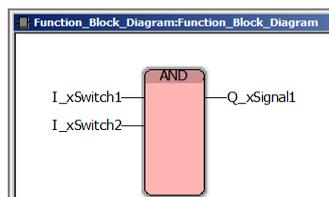
Instruction list

```
Instruction_List:Instruction_List
1 Ld I_xSwitch1
2 AndN I_xSwitch2
3
4 St Q_xSignal1
```

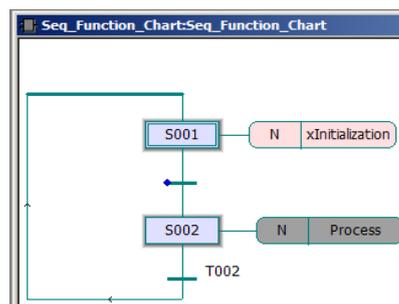
Structured Text

```
Structured_Text:Structured_Text
1 Q_xSignal1 := I_xSwitch1 & Not I_xSwitch2;
```

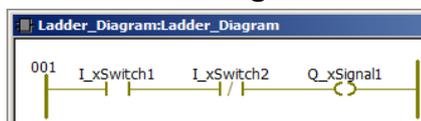
Function block diagram



Sequential functional chart



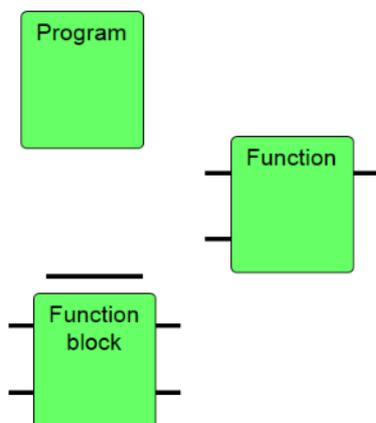
Ladder diagram



O PCWorx Express suporta apenas FBD, LD e ST. As linguagens IL e SFC estão presentes no PCWorx Professional.

8.2. IEC 61131

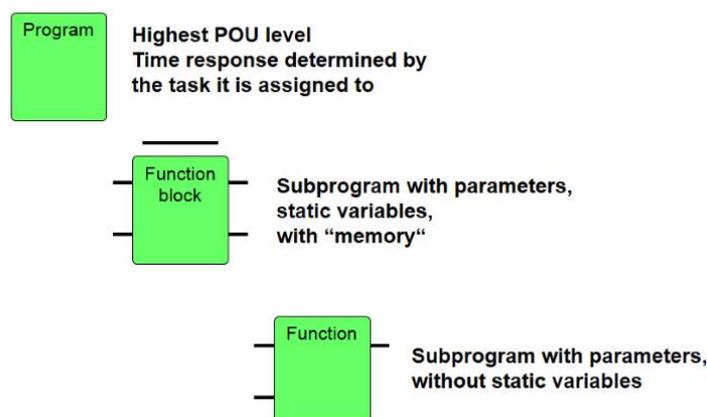
IEC 61131



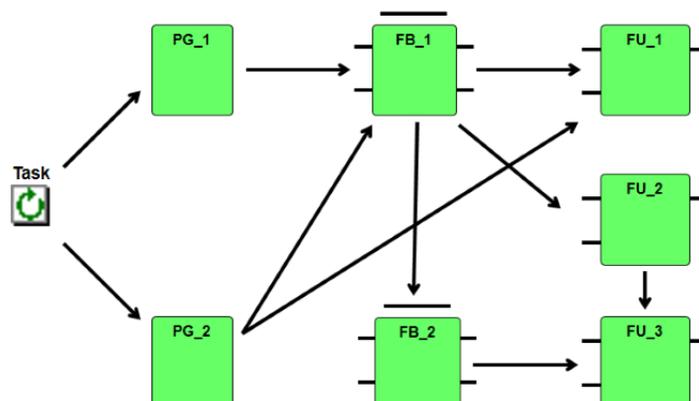
De acordo a IEC 61131, três tipos diferentes de POUs (Program Organization Units) são definidos: *Program*, *Function Block* e *Function*. Em geral, a programação pode ser implementada como um desestruturado alinhamento de comandos. No entanto, o uso desses três tipos permite um grau muito mais elevado de clareza no projeto, oferecendo vantagem a eficiência da programação na fase de desenvolvimento.

Ao seguir a IEC 61131, podemos definir as características e as áreas de aplicação para os tipos de POUs, através da descrição individual, bem como obter uma primeira visão geral sobre a forma como os dados podem ser trocados entre POUs configuradas.

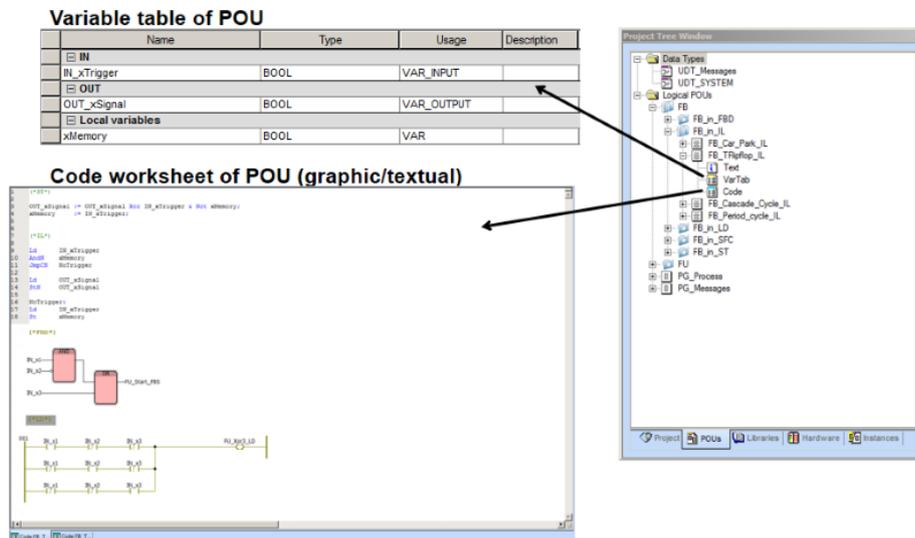
A figura abaixo mostra uma visão geral da hierarquia dos três tipos de POU e suas propriedades básicas.



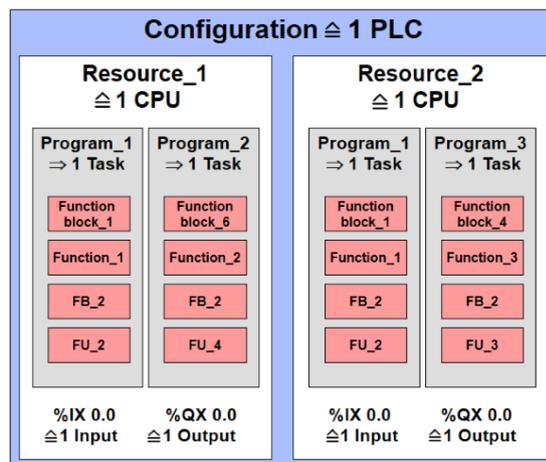
A estrutura para "chamar" as POUs é organizada hierarquicamente. O grau de complexidade para "chamar" as POU depende da memória da controladora. Por razões de segurança, chamadas recursivas não são permitidas (ilustração figura abaixo).



Uma tabela de variáveis locais está disponível para salvar as variáveis necessárias para o funcionamento da POU. Para o desenvolvimento do algoritmo desejado, um ou mais *worksheets* gráficas ou textuais estão disponíveis para o programador.

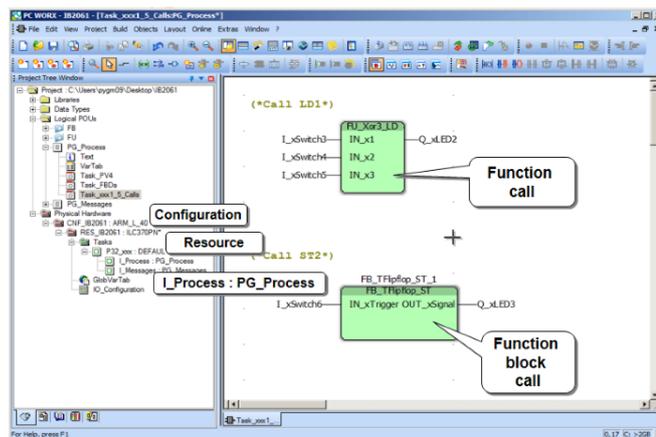


Terminologia



- **Configuração:** Descrição (física e lógica) de um sistema CLP completo.
- **Recurso:** Oferece suporte para todos os recursos que são necessários para o desempenho do programa; Interface entre um programa e os canais de I / O (input e output) físicos do CLP.
- **Tarefas:** Tempo para controlar a execução de diferentes partes de aplicação.
- **Programa (PG):** Unidade de organização do programa (POU), é atribuída a uma tarefa.
- **Bloco de funções (FB):** Subprograma (POU), contém dados estáticos.
- **Função (FU):** Subprograma (POU), sem dados abrangentes.

Implementação



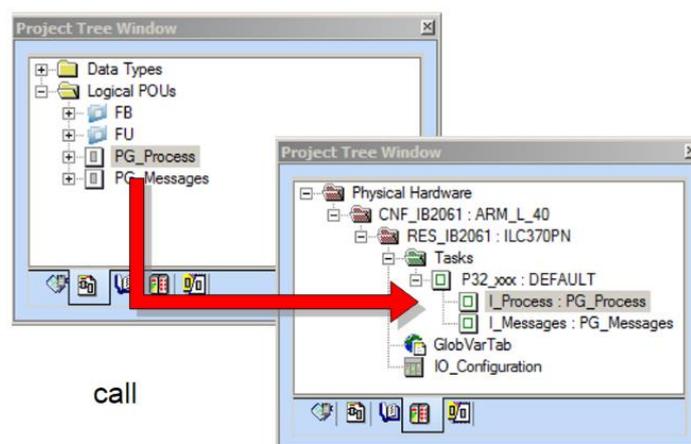
The program *PG_Process* is assigned to the instance *I_Process*.

- O lado esquerdo do *workspace* exibe a descrição do projeto de acordo com a IEC 61131.
- A parte superior da árvore do projeto contém a descrição da parte lógica do projeto: bibliotecas, tipos de dados e POUs lógicas.
- A parte inferior descreve a estrutura física: tipo (s) de CPU, estrutura da tarefa, acesso aos dados do processo.
- O lado direito do espaço de trabalho exibe os detalhes de elementos individuais.

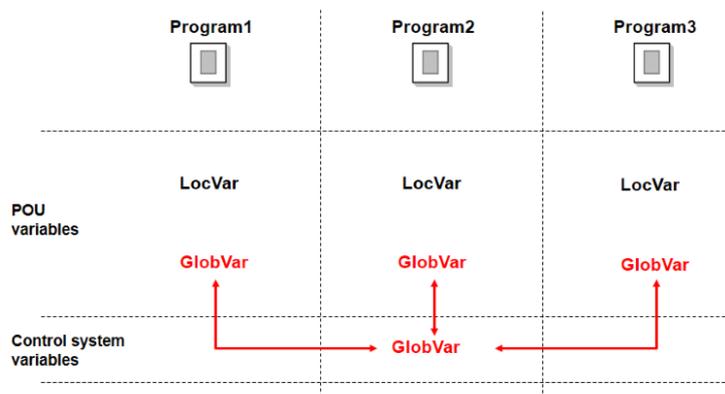
8.3. Uso de Programs (Programas)

O programa é uma POU do mais alto nível. Em um projeto, pelo menos uma POU desse tipo é necessária. Os programas, bem como as outras POUs, são salvos na pasta *Logical POUs* e, nesse ponto, não têm efeito no projeto de automação. Somente quando um programa na forma de uma instância de programa é usado em uma tarefa, o código salvo no programa é executado no CLP. Como padrão o programa é nomeado "Main".

Uma instância de um programa é a área de memória da CPU, na qual um código disponível como um programa é realizado de acordo com uma definição de tarefa. O PC WORX suporta múltiplas instâncias de programas (TASK DEFAULT)



Uma das principais tarefas dos programas é ligar os sinais de hardware à programação. As variáveis globais são a única maneira de estabelecer este link. Eles também são usados para trocar informações globais sobre o sistema de controle. Para programas, variáveis globais são a única possibilidade de trocar dados com outros elementos de programação.

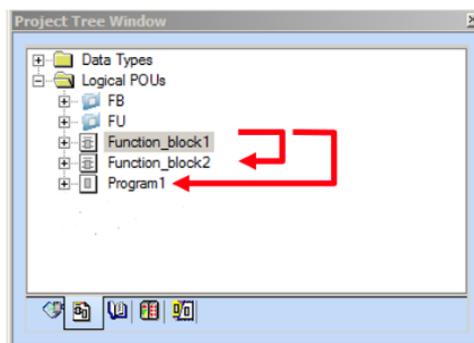


8.4. Uso de Function Block (Blocos de Funções)

A criação de blocos de funções permite que o código de um programa criado, possa ser utilizados, idem ao programa, entretanto, as POUs parametrizadas como FB (*Function Block*) também oferecem a vantagem do encapsulamento.

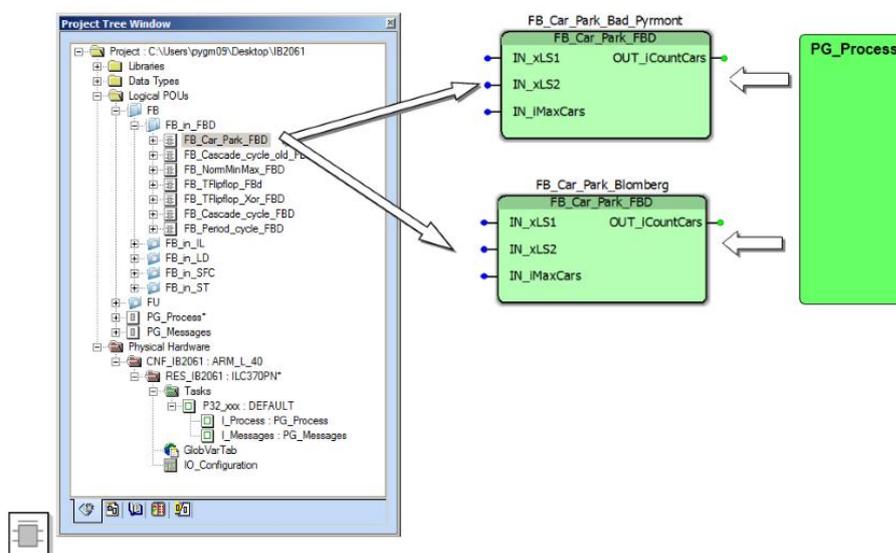
Isso significa que uma programação ampla pode ser criada, por exemplo, em um idioma baseado em texto. A chamada, incluir a conexão de apenas alguns valores de entrada e saída, pode ser realizada em um idioma gráfico, que oferece maior clareza.

Conforme mostrado na figura abaixo, o FB podem ser chamados em outros FB, bem como em programas. Não é possível uma chamada recursiva.



Como os FB podem conter valores diferentes, durante os vários ciclos do programa, para cada chamada de um FB, a memória individual deve estar disponível no sistema de controle. Esta memória é chamada de instância de FB e pode ser comparada a uma variável complexa.

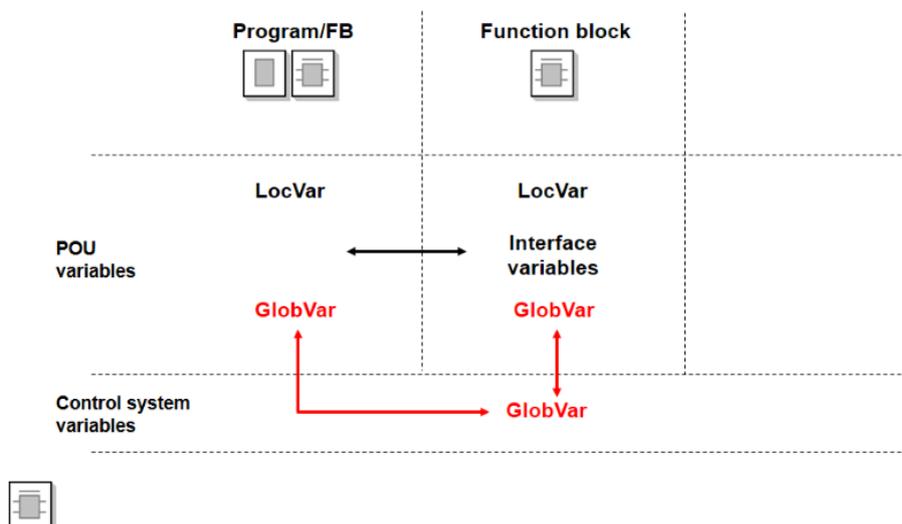
A instância do FB está listada na lista de variáveis da POU na qual o FB é chamado.



Na figura acima temos um exemplo, o mesmo tipo de algoritmo (FB_Car_Park_FBD) é chamado duas vezes: o programa PG_Process usa o bloco de função Car_Park duas vezes para tarefas diferentes. Para ambas as tarefas, a memória individual está disponível (com os nomes FB_Car_Park_Bad_Pyrmont e FB_Car_Park_Blomberg).

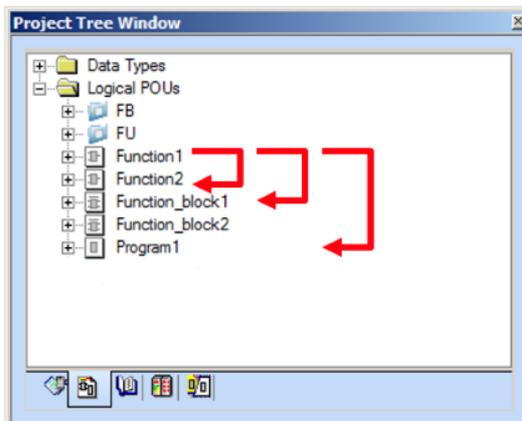
Em relação à troca de dados, os FB oferecem a maior flexibilidade. De acordo com a abordagem clássica, a troca de dados só é realizada por meio de parâmetros de entrada e saída (variáveis de interface).

Para aplicações específicas, no entanto, pode fazer sentido trocar dados através de variáveis globais. A desvantagem para o programador ao usar este método é que não há transparência direta quanto aos dados processados durante a chamada.



8.5. Uso de Functions (Funções)

As funções (*FU – Functions*) podem ser usadas em outras funções, FB e programas (ilustração a seguir). Assim como com os blocos de função, uma chamada recursiva não é possível.



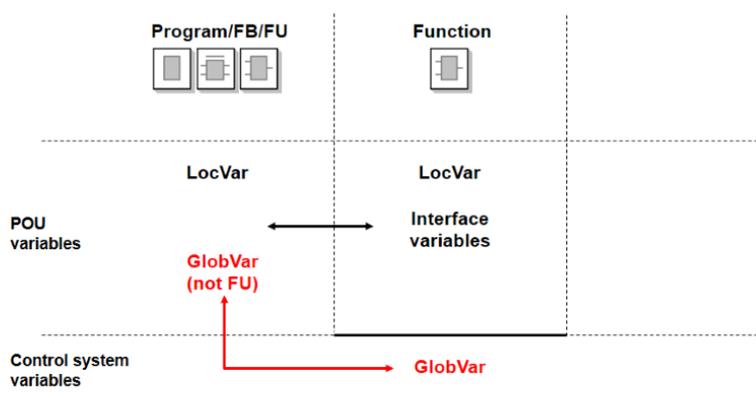
Call



Ao contrário dos FB, as FU são limitadas na medida em que não podem salvar dados em vários ciclos de CLP. As FU também não são aplicáveis se o programa deve fornecer mais de um parâmetro de saída, pois as funções só podem retornar um único resultado.

A vantagem das FU em relação aos FB reside no fato de que uma FU não precisa ser instanciada. Para uso múltiplo, isso significa que a mesma memória é usada uma e outra vez para execução do programa.

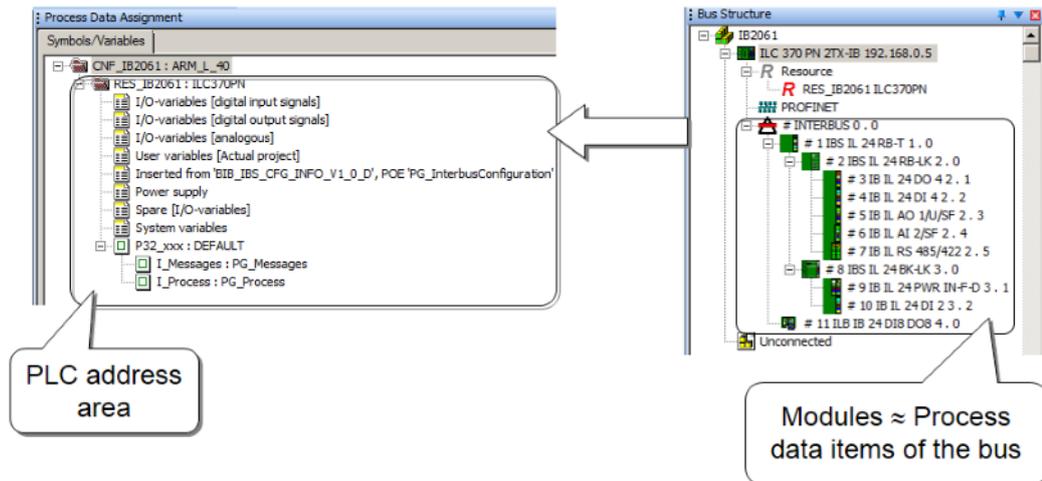
No que diz respeito à troca de dados, as FU estão claramente definidas. Eles devem, a partir de pelo menos um parâmetro de entrada, retornar exatamente um resultado. Internamente, o valor de retorno da FU deve ser salvo em um parâmetro já disponível devido à criação da função na árvore do projeto. As funções não podem acessar variáveis globais (ilustração abaixo).



9. Gerenciamento de Dados

Este capítulo aborda os parâmetros que determinam uma variável e as aplicações variáveis que podem ser usadas nas POU's de acordo com a IEC 61131. Além disso, as variáveis de dados do processo, a conexão entre variáveis globais e locais, bem como as aplicações disponíveis em FU e FB também são abordados.

Os itens de dados do processo disponibilizados pelos “*Bus Structure – Process Data Workspace*” através dos arquivos .XML, devem estar disponíveis para a aquisição de sinais de entrada e saída do PLC (ilustração abaixo).



O espaço de trabalho “*Process Data Workspace*” do PC WORX, oferece as ferramentas certas para criar variáveis com base nos itens de dados do processo ou conectar variáveis existentes com itens de dados de processo disponíveis. Ambos os métodos são descritos a seguir.

9.1. Criando variáveis de dados do processo

Para a criação de variáveis de dados do processo, selecione o elemento correspondente à CPU do CLP no quadrante superior do lado esquerdo “*Process Data Workspace*”. A menos que o usuário tenha renomeado o elemento após a criação do projeto, em um projeto baseado em um modelo de projeto padrão, o elemento possui a designação STD_RES (recurso padrão).

Dependendo do elemento selecionado no quadrante superior do lado direito “*Bus Structure – Process Data Workspace*”, no quadrante inferior do lado direito apresenta-se:

- Arquivos de sistema (PROFINET e/ou INTERBUS). Todos os itens de dados de processo são exibidos;
- Para *bus couplers*, todos os itens de dados de processo atribuídos ao *bus segment*, são exibidos;
- E para módulos individuais, apenas os itens de dados do processo atribuídos ao módulo são exibidos.

No filtro (click com o botão direito do mouse, em cima da aba “*Process Data Item*”, no quadrante inferior do lado direito), você pode selecionar itens de dados de processo padrão e / ou individual, itens de dados de processo conectados e / ou desconectados.

Acesse agora o *Process Data Workspace*: Selecione agora no quadro superior direito o primeiro cartão presente no *INTERBUS*. Repare no quadro inferior direito as conexões de I/O disponíveis. Os cartões digitais requerem variáveis do tipo *BOOL* e os analógicos *WORD*. Repare que as entradas são marcadas com a cor verde e as saídas em magenta.

A criação de variáveis de dados do processo é feita selecionando a função “*Create Variable*” no menu de contexto de uma linha de dados de processo selecionada (ilustração a seguir).

The screenshot shows the 'Process Data Item' table with the following data:

Device	Process Data Item	I/Q	Data Type	Byte.Bit	Address
# 3 IB IL 24DO 42. 1	1.1	Q	BOOL	0.0	
# 3 IB IL 24DO 42. 1	2.1	Q	BOOL	0.1	
# 3 IB IL 24DO 42. 1	1.4	Q	BOOL	0.2	
# 3 IB IL 24DO 42. 1	2.4	Q	BOOL	0.3	
# 3 IB IL 24DO 42. 1	-DO 4	Q	BITSTRIN...	0.0	

The context menu is open, and the 'Create Variable' option (Ctrl+E) is highlighted. Below the table, the variable definition table is shown:

Symbol/Variable	Data Type	Process Data Item
Q_2_1_1_4	BOOL	# 3 IB IL 24DO 42. 1 \ 1.4



Certifique-se de que os itens de dados do processo não interferem entre si. Isso levaria a uma mensagem de erro durante a compilação do projeto. As variáveis criadas são sempre inseridas no grupo de variáveis “Auto”, que, se não foi criado até agora, é criado automaticamente pelo PC WORX Professional (PC WORX Express insere o grupo de variáveis no grupo “Default”).

O nome das variáveis criadas automaticamente, segue o seguinte padrão:

<Direção de dados> _ <Device ID IBS / PN> _ <ponto de conexão>

Os caracteres não permitidos em um nome de variável são substituídos por sublinhados. O nome das variáveis só pode ser adaptado nas tabelas de variáveis, não na atribuição de dados do processo.

Após a associação de hardware é possível renomear a lista de variáveis criadas. Para tanto, no *IEC Programming Workspace*, acesse a aba *Hardware*, abra a janela “*Hardware*”, e renomeie a lista “*Auto*” ou “*Default*”.

9.2. Princípio para declaração de variáveis

De acordo com a IEC 61131, todas as letras maiúsculas e minúsculas, todos os números e os sublinhados como separador ou símbolo principal, podem ser usados como um conjunto de caracteres para nomes de variáveis.

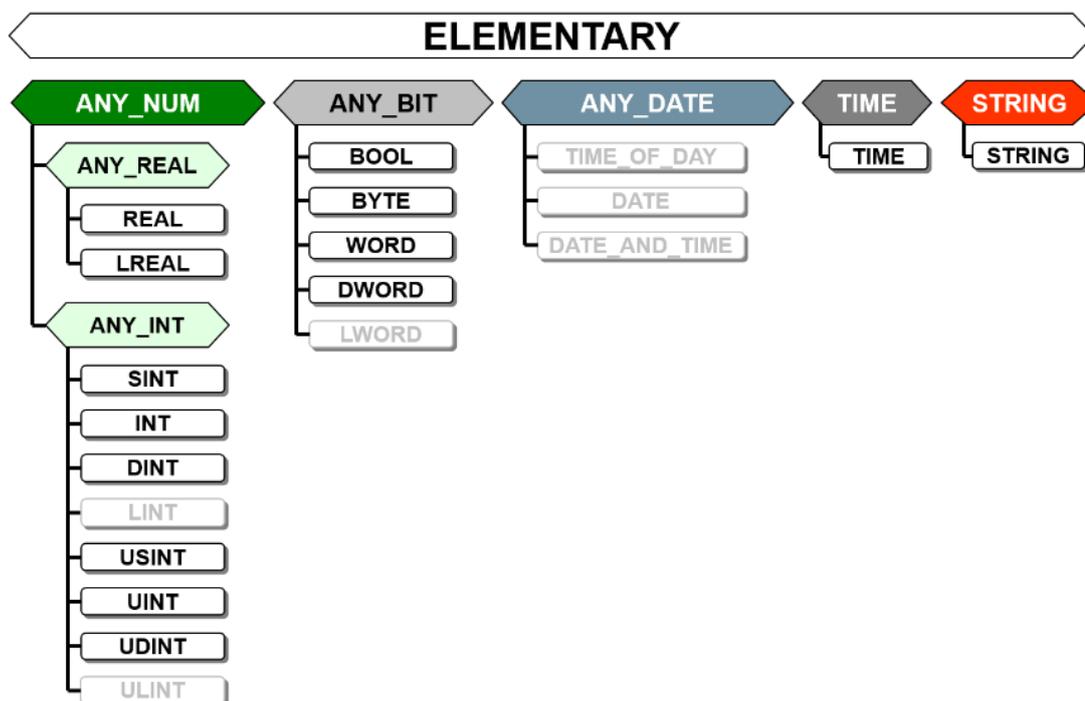
O PC WORX oferece a oportunidade de habilitar o uso de designações DIN. Quando esta possibilidade é utilizada, a programação das POU's escritas em texto estruturado deve ser única.

Menu *Extras* → *Options* → *Build* → *Use DIN-identifier*

Um parâmetro variável importante é o tipo de dados. Após a instalação do PC WORX, uma variedade de tipos de dados definidos pela IEC 61131 estão disponíveis. O uso de uma variável indica sua validade sobre o CLP e a função que pode assumir para a troca de dados de sua POU com outras POU's e com o CLP.

A descrição de uma variável é opcional e pode conter todos os caracteres. No *IEC Programming Workspace*, acesse a aba *Hardware*, abra a janela *Global_Variables*, para renomear as variáveis caso necessário.

A maior parte dos tipos de dados definidos pelo IEC 61131 está disponível como um padrão no PC WORX. Esses tipos de dados são divididos em grupos. Certas FU e FB não precisam necessariamente ser equipados com a variável de um determinado tipo de dados, mas com um tipo de dados de um grupo (por exemplo, blocos de função da lógica booleana AND, OR, XOR, e NOT, NAND, NOR e XNOR com ANY_BIT). Vide na ilustração e tabela a seguir os tipos de dados.



Tipos de dados numéricos

Keyword	Tipo de dado	Tamanho [Bits]	Amplitude
SINT	8 bit integer (with sign bit)	8	-128..127
INT	Integer (with sign bit)	16	-32768..32767
DINT	Double integer (with sign bit)	32	-2.147.483.64..2.147.483.647
LINT	64 bit integer (with sign bit)	64	-9.223.372.036.854.775.808.. 9.223.372.036.854.775.807
USINT	8 bit Integer (w/o sign bit)	8	0..255
UINT	Integer (w/o sign bit)	16	0 ..65.535
UDINT	Double integer (w/o sign bit)	32	0..4.294.967.295
ULINT	64 bit integer (w/o sign bit)	64	0..18.446.744.073.709.551.615
REAL	32-bit floating-point value	32	+/- 1.5*10 ⁻⁴⁵ .. +/- 3.4*10 ³⁸
LREAL	64-bit floating-point value	64	

Tipos de dados lógicos

Keyword	Tipo de dado	Tamanho [Bits]	Amplitude
BOOL	Boleano	1	0..1 _{HEX} (also false / true)
BYTE	Bit string 8	8	0..FF _{HEX}
WORD	Bit string 16	16	0..FFFF _{HEX}
DWORD	Bit string 32	32	0..FFFF FFFF _{HEX}
LWORD	Bit string 64	64	0..FFFF FFFF FFFF FFFF _{HEX}

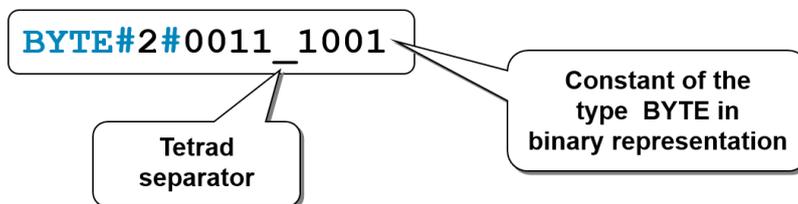
Tipos de dados estendidos

Keyword	Tipo de dado	Tamanho [Bits]	Amplitude
TIME	Tempo	32	0..4.294.967.295ms(cf. UDINT)
DATE	Data		
TOD	Tempo do dia		
DT	Data e Tempo		
STRING	String (standard)	80[bytes]	

9.3. Sintaxe de uma constante

Para a declaração de constantes, ou valores iniciais, A sintaxe formalmente correta para declarar constantes (de acordo com IEC 61131) é a seguinte:

`<Data type>#<Basis>#<Constant value><Unit>`



O tipo de dado deve ser um dos descritos na seção anterior.

A base pode ser:

- 2: Binária
- 8: Octal
- 10: Decimal
- 16: Hexadecimal

Para valores de tempo temos:

- ms: Mili-segundos
- s: Segundos
- h: Horas
- d: Dia

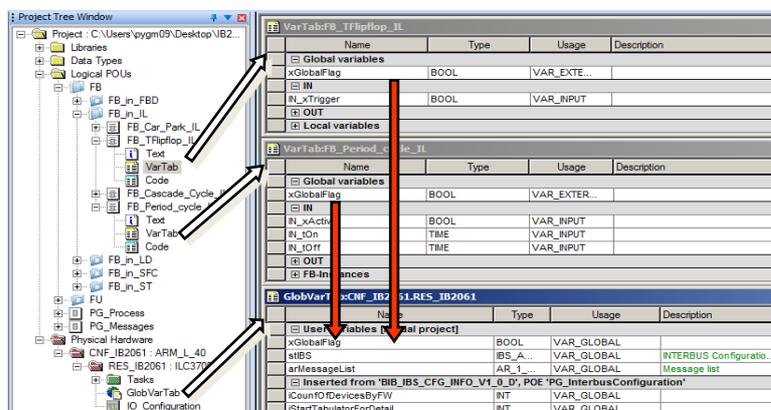
9.4. Exemplos de representação de constantes

Esta seção mostra possíveis formatos para constantes. Lembre-se de que, para alguns tipos de dados, é permitido usar uma sintaxe abreviada que é frequentemente encontrada na programação.

Integer constants	<code>Int#12456</code>	<code>-12</code>	<code>0</code>	<code>12_456</code>	<code>+986</code>
Real constants	<code>Real#1.6e3</code>	<code>-12.0</code>	<code>0.0</code>	<code>0.456</code>	<code>+2.635e-12</code>
Word constants <small>Also BYTE and DWORD</small>	<code>Word#12094</code>	<code>Word#16#2F3E</code>	<code>Word#2#0010_1111_0011_1110</code>		
Bool constants	<code>Bool#1</code>	<code>Bool#0</code>	<code>True</code>	<code>False</code>	
Time constants	<code>Time#1.64s</code>	<code>t#2d_14h_3.5s</code>			
String constants	<code>String#'Pause?'</code>		<code>'Jetzt bitte!'</code>		

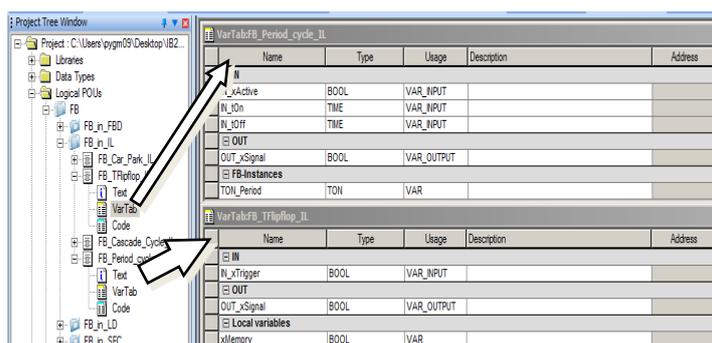
9.5. Variáveis GLOBAIS

Variáveis declaradas como GLOBAIS podem ser lidas e escritas por qualquer POU declarada e implementado no projeto.



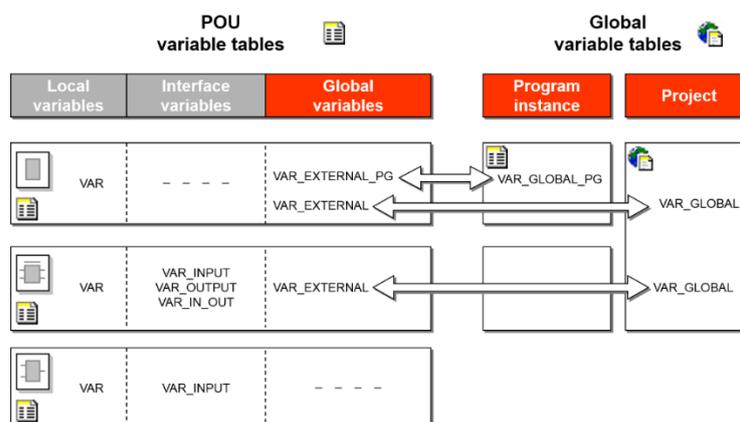
9.6. Variáveis LOCAIS

As variáveis LOCAIS

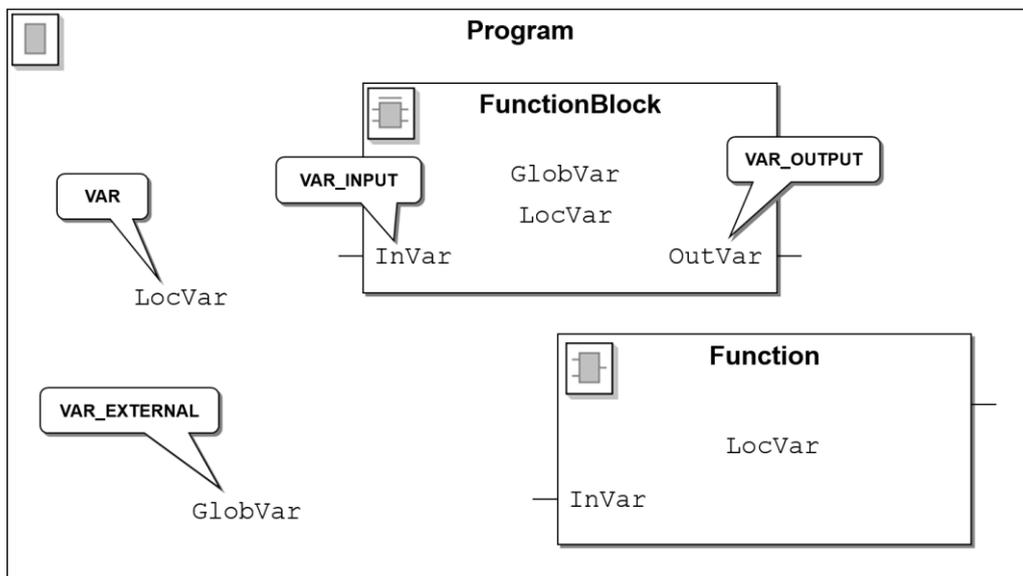


Em uma tabela de variáveis locais, as variáveis locais e globais que devem ser vinculadas entre si no programa na POU são inseridas.

Portanto, a tabela de variáveis locais não é a designação para uma tabela de variáveis local, mas para a tabela local que pertence a uma POU para variáveis globais e de uso local (ilustração a seguir).



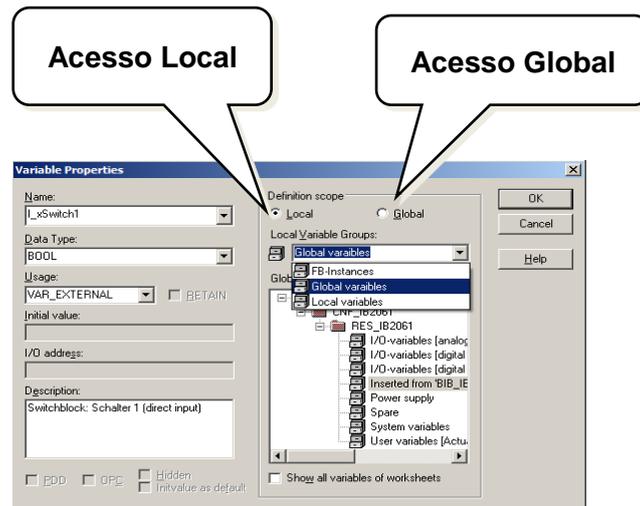
9.7. Variáveis de Interface



Além das aplicações já descritos, está figura mostra as possibilidades de utilização nas FU e FB.

	Program	Function Block	Function
Variáveis aplicáveis globalmente	VAR_GLOBAL	VAR_GLOBAL	Não disponível
Variáveis de interface locais	Não disponível	VAR_INPUT VAR_OUTPUT VAR_IN_OUT	VAR_INPUT (Saída extra declarada)
Variáveis locais	VAR	VAR	VAR

O uso VAR_IN_OUT para FU é especialmente interessante para a transmissão de grandes quantidades de dados (variáveis de campo e estrutura), não sendo necessário copiar dados.



A caixa de diálogo “*Variable Properties*” permite o acesso a variáveis nas tabelas de variáveis locais e às tabelas de variáveis globais, e, também permite visualizar e alterar as variáveis existentes e novas declarações.

10. Exercício final

Exemplo de projeto “**FirstSteps_ILC131**”

Efetue o carregando o projeto de exemplo a partir do DVD (disponível no ILC 131 Starterkit):

- Salve o arquivo "FirstSteps_ILC131.zwe" em seu PC, por exemplo, sob o caminho padrão utilizado.
- Selecione "Abrir Project / Unzip Project ..." no item de menu "Arquivo".
- Selecione o arquivo "FirstSteps_ILC131.zwe" no caminho sob o qual você salvou o arquivo.
- Confirme, caso o prompt perguntar se o projeto deve ser extraído para o diretório contendo o arquivo *.zwe, com "Sim". Ou selecione um diretório diferente. Lembre-se que o PC WORX EXPRESS extrai o arquivo automaticamente.
- Se o PC WORX EXPRESS solicitar que você substitua layouts (por exemplo, layouts de página) ou Bitmaps, confirme este prompt com "Não para todos".
- Uma vez que PC WORX EXPRESS descompactou o projeto, ele está disponível para você visualizar / editar.

Complete as seguintes etapas antes de iniciar o projeto:

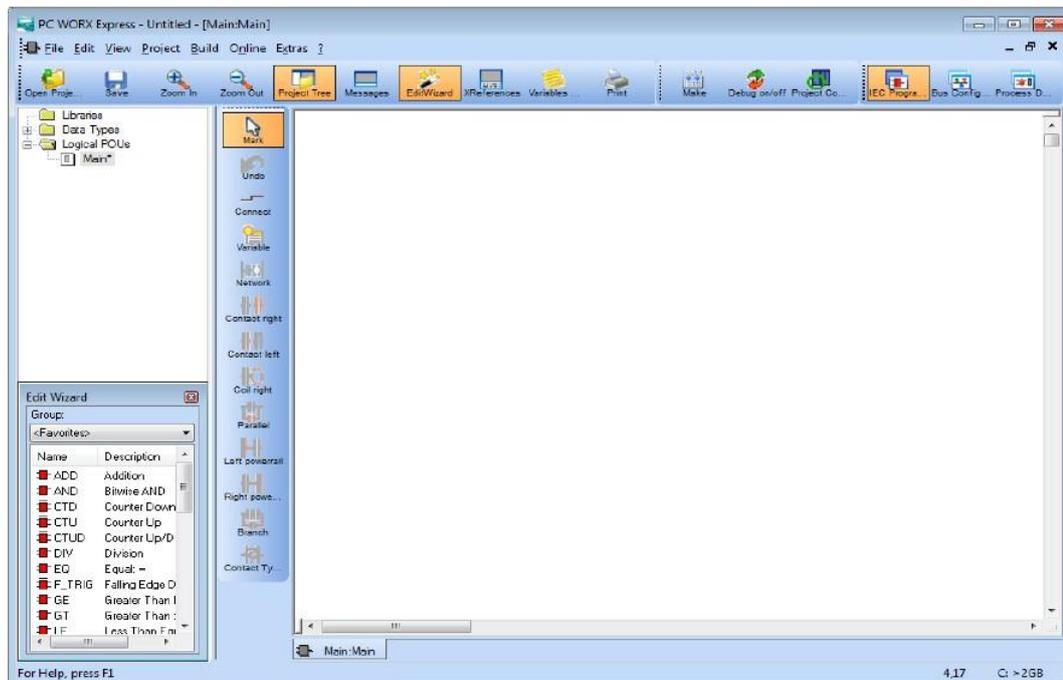
- Verifique as configurações de IP no projeto de exemplo. Adapte as configurações se isso for necessário para seu ambiente de rede. Para fazer isso, proceda conforme descrito na **Seção 4.2 Rede Ethernet TCP-IP**.
- Compile o projeto de exemplo, envie-o para o controlador e execute “cold restar”.

Pergunta: Após a análise e teste do Exemplo de projeto “FirstSteps_ILC131”, caso necessário a criação deste projeto a partir do “zero” qual deveria ser o passo a passo mais adequado?

10.1. Vamos criar nosso próprio exemplo?

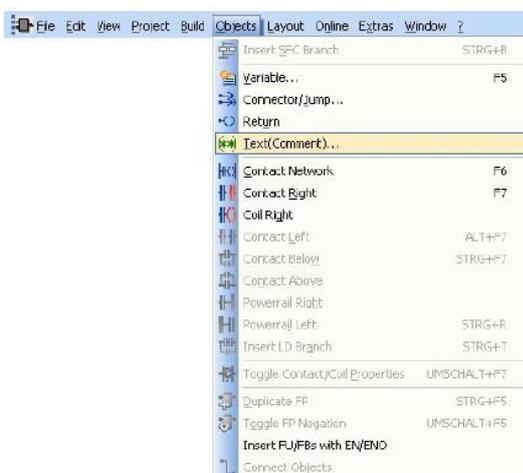
Para programação, proceda da seguinte forma:

- Mude para o espaço de trabalho *IEC Programming workspace*;
- Clique duas vezes em "Main" na "Project Tree Window" para ativar a programação *IEC programming interface*;
- Na planilha "Main", clique na posição vazia onde você gostaria de inserir o bloco de função.

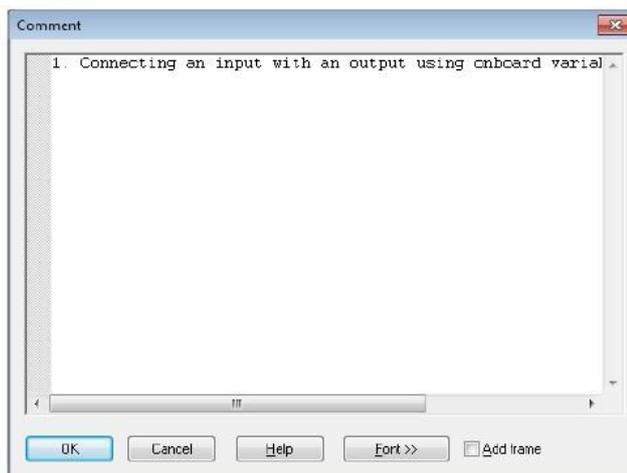


Inserindo um comentário:

- Clique na posição desejada na planilha para definir um cursor;
- Para inserir um comentário, selecione o comando "Texto (Comentário) ..." no menu "Objetos".



Na caixa de diálogo "Comentar" que se abre, insira o comentário desejado para a POU "Main".

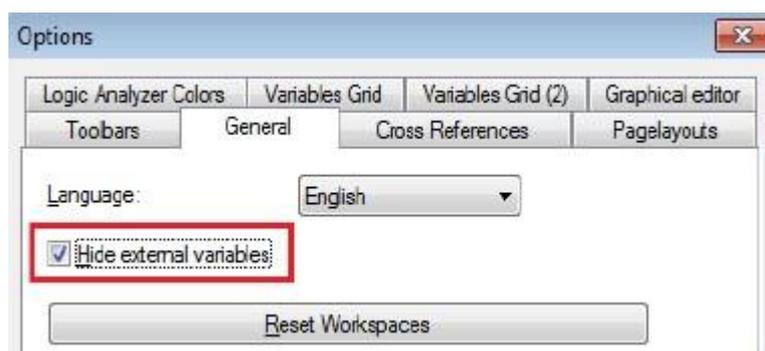


Criando variáveis

As entradas e saídas de sinais já associadas ao ILC 131 ETH são mapeadas como variáveis do sistema. A variável para processar o valor de entrada analógica (Input_Analog), por exemplo, é criada como uma variável de usuário. No PC WORX EXPRESS, o uso de uma variável deve ser declarado.

Uma variável que é usada dentro de uma POU é declarada como uma variável local usando uma das palavras-chave VAR, VAR_INPUT, VAR_OUTPUT ou VAR_IN_OUT. Uma variável que é usada em todo o projeto é declarada como uma variável global usando a palavra-chave VAR_GLOBAL.

Certifique-se de que a caixa de seleção "*Hide external variables*" esteja selecionada na caixa de diálogo "Opções" (menu "Extras, Opções ..."), ilustração a seguir:



Variável local / global

Quando uma nova variável é declarada no PC WORX EXPRESS, a validade só pode ser definida através de "Usage". Ao selecionar uma variável que já foi declarada, sua validade pode ser determinada através dos ícones no menu suspenso "Name".

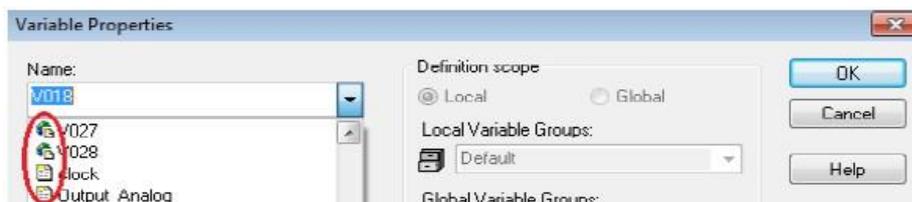


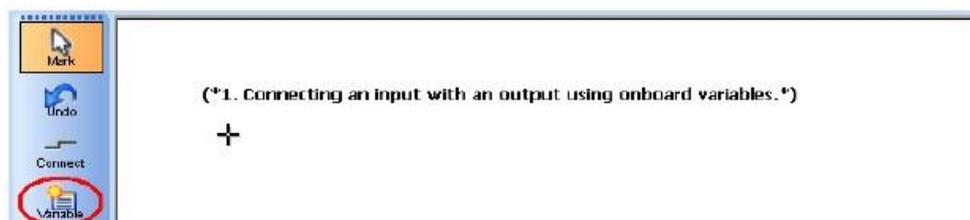
Figure 5-25 "Variable Properties" window – icons

-  : Local variable
-  : Global variable

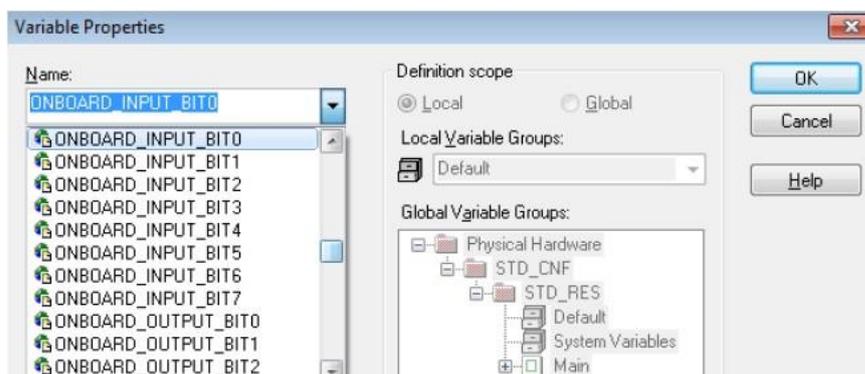
Grupos de variáveis locais e grupos de variáveis globais: Por razões de compatibilidade com o PC Worx, os itens "Local Variable Groups" e "Global Variable Groups" são mostrados na caixa de diálogo "Propriedades da variável", mas não podem ser configurados.

Passo 1: Ligação direta de uma entrada do controlador com uma saída do controlador usando variáveis internas:

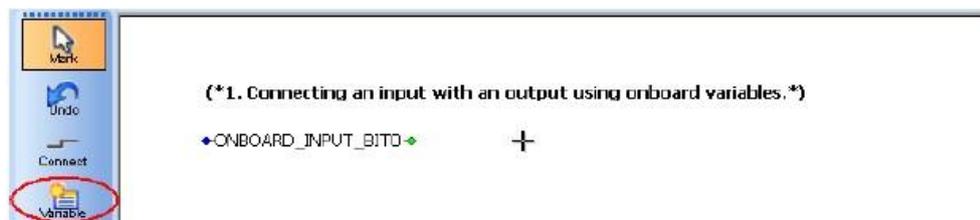
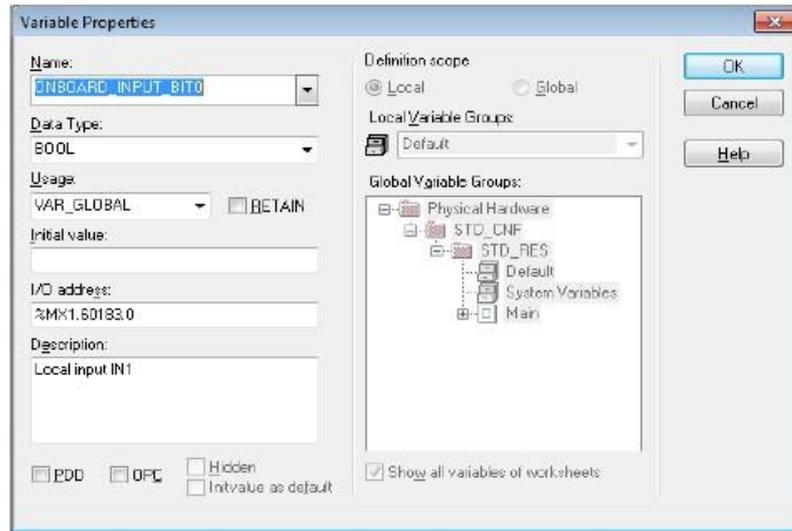
- Criar variáveis (variáveis de usuário / variáveis do sistema);
- Vincular variáveis;
- Clique na posição desejada na planilha para definir um cursor;
- Clique em "Variable" para adicionar uma variável à planilha, alternativamente, selecione o comando "Variable" no menu de contexto para o cursor;



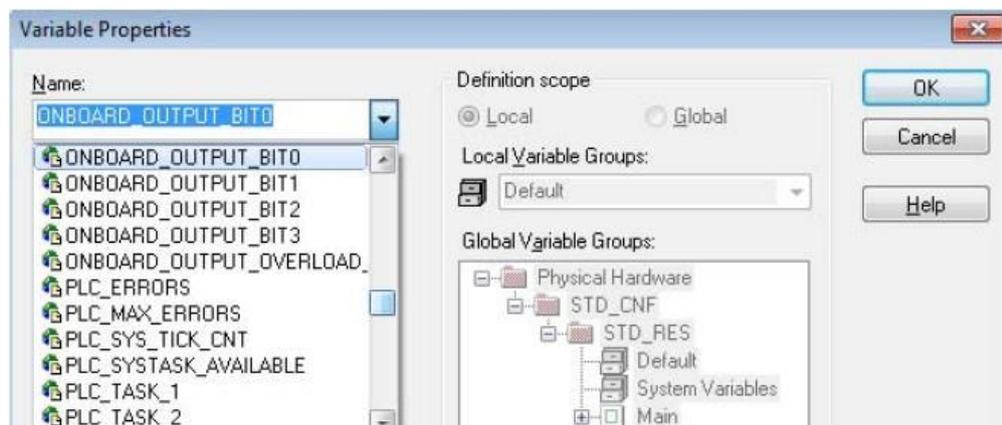
- Na caixa de diálogo "Variable Properties", selecione a variável onboard "ONBOARD_INPUT_BIT0" (entrada do controlador I1).



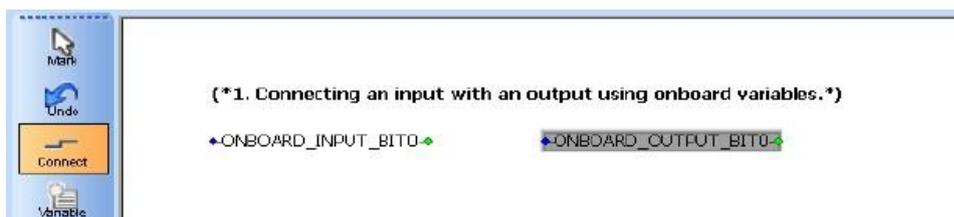
- The data type is "BOOL".
- The usage is "VAR_GLOBAL".
- Confirme suas entradas clicando em "OK".



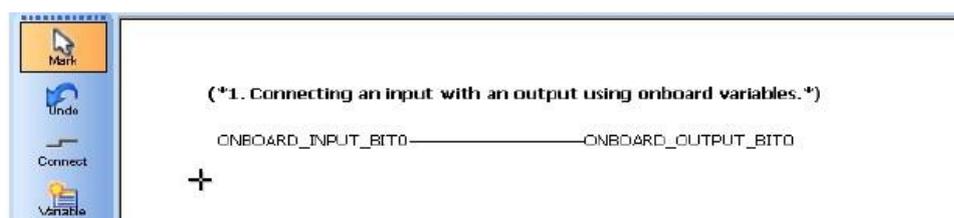
- Clique novamente na worksheet (veja a ilustração supracitada) para definir um cursor;
- Clique em "Variable" para adicionar outra variável à planilha;
- Na caixa de diálogo "Variable Properties", selecione a variável "ONBOARD_OUTPUT_BIT0" (saída do controlador Q1).



- The data type is "BOOL".
- The usage is "VAR_GLOBAL".
- Confirme suas entradas clicando em "OK".



- Clique em "Connect" para estabelecer uma conexão entre as variáveis criadas acima;
- Clique primeiro na saída para a variável ONBOARD_INPUT_BIT0 e depois na entrada para a variável ONBOARD_OUTPUT_BIT0.



- Compile o programa que foi criado. Clique no ícone "Make" na barra de ferramentas.



Uma vez que o processo de compilação estiver completo, PC WORX EXPRESS exibe uma mensagem correspondente na janela de mensagem.

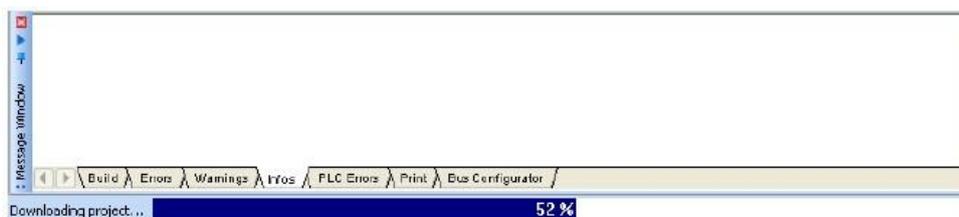
Se o código do programa não tiver sido inserido corretamente, a mensagem indica o número de erros e avisos (guia "Build"). Informações mais detalhadas sobre erros / avisos são exibidas nas guias "Erros" e / ou "Warnings".

Para testar o programa, transmita-o para o controlador:

- Clique no ícone "Project Control Dialog" na barra de ferramentas;

Se apenas o botão "Close" estiver ativado na Project Control Dialog que se abre, não há conexão válida com o ILC 131 ETH. Nesse caso, verifique o caminho de comunicação.

- Com a tela "Project Control Dialog" aberta, click no botão "Download", O processo de download irá começar.





Uma vez que o download está completo, PC WORX EXPRESS exibe uma mensagem correspondente na janela de mensagem. Realizar uma *cold restart* para ativar o projeto

- Para fazer isso, clique em "cold" na caixa de Project Control Dialog

O ILC 131 ETH foi iniciado com sucesso se o LED FR verde estiver ligado.

- Ative o interruptor 0 do módulo de comutação para testar o programa;
- Se todas as etapas descritas acima tiverem sido realizadas corretamente, os LEDs I1 e Q1 acendem no controlador.

Passo 2: Uso de FU da biblioteca de FU:

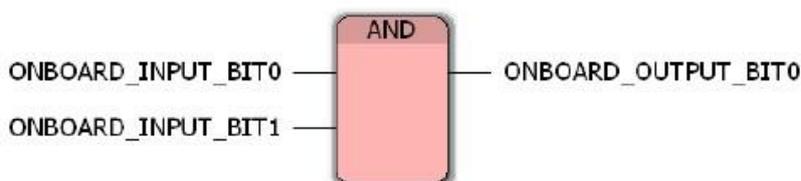


- Selecionar um FU da biblioteca FU e inserindo-o na planilha;
- Conexão das entradas e saídas dos FB;
- Vincular as variáveis a bordo usando um FN;

No assistente de edição "edit wizard", selecione o FB "AND" clicando e arrastando para a worksheet.

- O FU é inserido na planilha na posição do cursor.

(*AND relation between two onboard input variables connected to an onboard output variable*)



Se você conhece o nome do FU desejado, outra opção é inserir o nome do FU diretamente usando o teclado do seu PC, basta inserir o cursor na *worksheet* e digitar o nome do FU.

Insera as variáveis do sistema **ONBOARD_INPUT_BIT0**, **ONBOARD_INPUT_BIT1** e **ONBOARD_OUTPUT_BIT0** na *worksheet* (conforme ilustração acima).

- Compile o programa para o controlador, realizando uma *cold restart* e execute o teste do programa.

Passo 3: Conectar FU usando o exemplo de geração de clock

- Clock fixo;
- Clock variável.



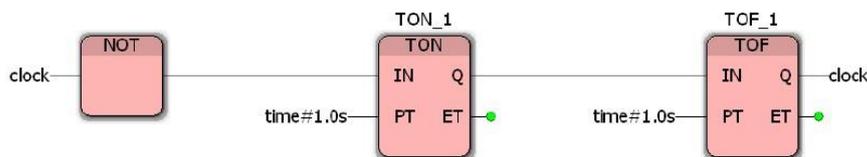
Informação sobre a geração do clock: Conectando dois elementos de atraso (TON / TOF) sucessivamente e retornando o sinal de saída (clock) gera-se um sistema "oscilante", que é usado para geração de clock.

Clock fixo

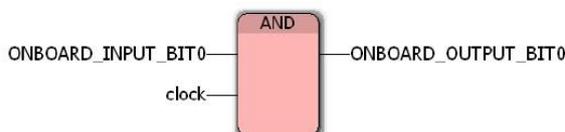
Para inserir o código do programa no worksheet de acordo com a figura abaixo, execute as seguintes etapas:

- Insira os comentários no worksheet;
- Insira os FU NOT, TON e TOF do *edit wizard* no worksheet;
- Insira as variáveis do Sistema ONBOARD_INPUT_BIT0 e ONBOARD_OUTPUT_BIT0;

(* Clock generation*)



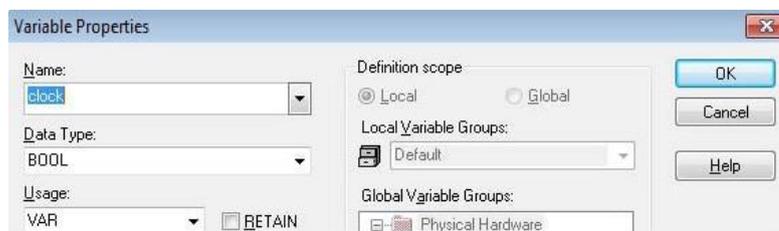
(* AND relation between an onboard input variable and a clock variable connected to an onboard output variable*)



- Insira a variável do usuário "clock".

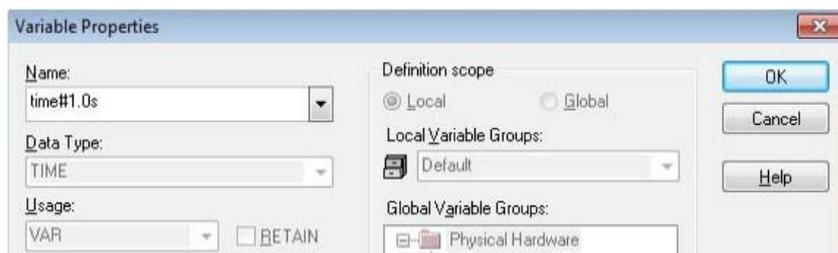


Para criar uma variável de usuário, especifique um nome (campo "Name") e tipo de dados (campo "Data Type"). No campo "Usage", especifique se a variável deve ser usada somente nesta planilha (local: VAR) ou em outras planilhas (global: VAR_GLOBAL).



- O nome é "CLOCK".
- O tipo de dados é "BOOL".
- O uso é "VAR".
- Defina uma constante de tempo para as entradas PT dos blocos de função TON e TOF.

- Clique duas vezes na entrada correspondentes PT. Na caixa de diálogo "Variable Properties", insira a constante de tempo de acordo com a figura abaixo.

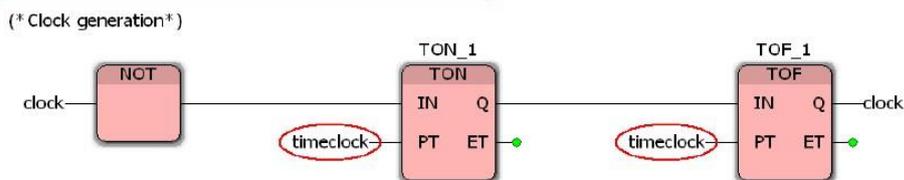
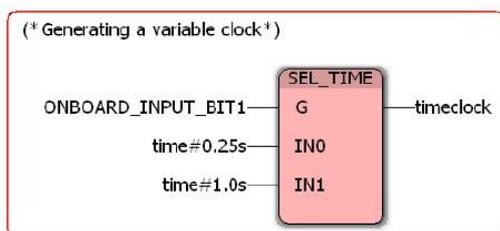


- Estabeleça todas as conexões necessárias entre variáveis e FUs;
- Compile o programa para o controlador, realizando uma *cold restart* e execute o teste do programa.

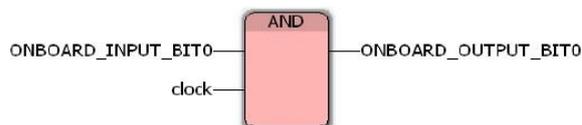
Clock variável

Para inserir o código do programa no worksheet de acordo com a figura abaixo, execute as seguintes etapas:

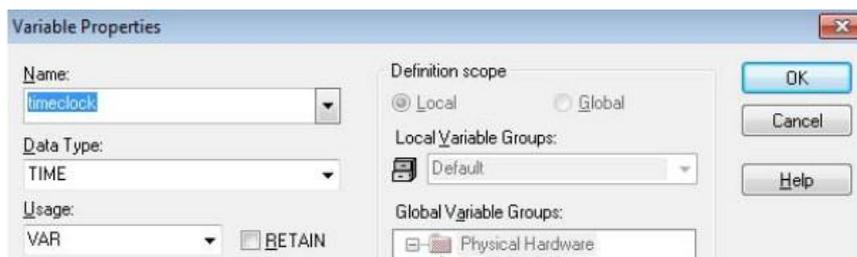
- Insira os comentários no worksheet;
- Insira o FU SEL_TIME do *edit wizard* no worksheet;
- Insira a variável do sistema ONBOARD_INPUT_BIT1;



(* AND relation between an onboard input variable and a clock variable connected to an onboard output variable*)



- Insira a variável do usuário "timeclock";

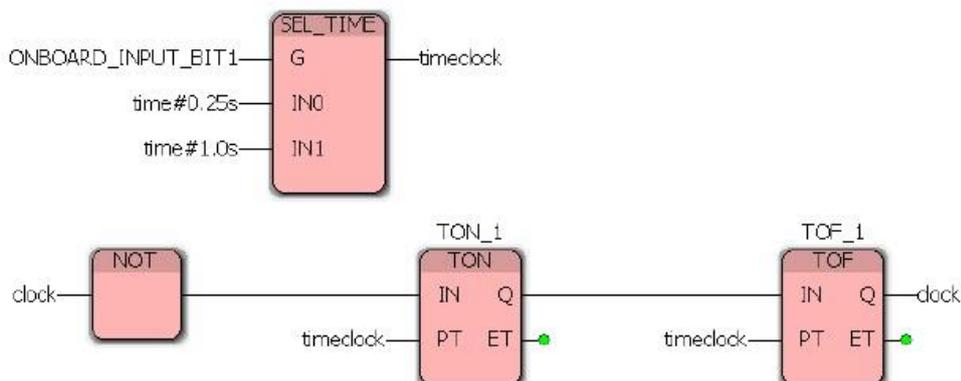


- Defina as constantes de tempo para as entradas IN0 / IN1 da FU SEL_TIME;
- Estabeleça todas as conexões necessárias entre variáveis e as FUs.
- Compile o programa para o controlador, realizando uma *cold restart* e execute o teste do programa.

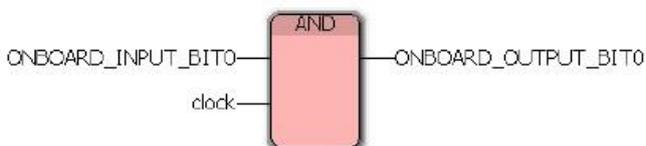
Passo 4: Ligação direta de duas variáveis de valor analógico / atribuição de dados

- Criar variáveis de valor analógico;
- Processar a atribuição de dados de uma entrada do dispositivo para uma variável de valor analógico;

(* Generating a variable clock*)



(* AND relation between an onboard input variable and a clock variable connected to an onboard output variable*)



(* Connecting an analog input variable with an analog output variable*)



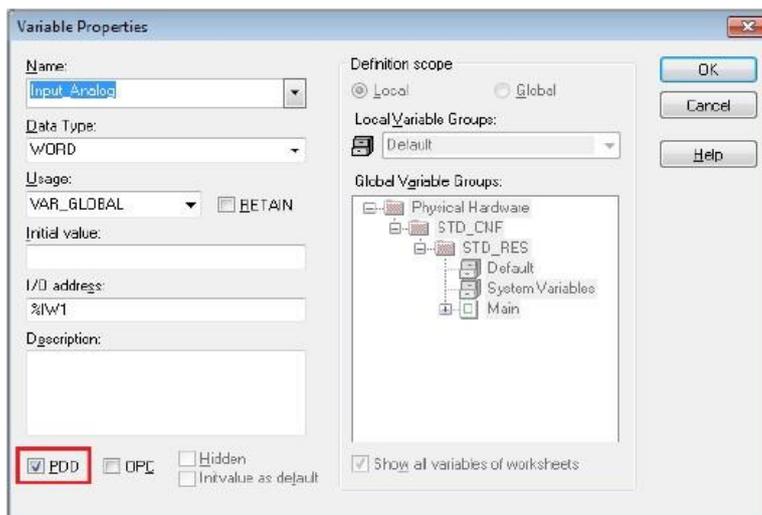
Para inserir o código do programa no worksheet de acordo com a figura abaixo, execute as seguintes etapas:

- Insira o comentário no worksheet;
- Insira a variável "Input_Analog" no worksheet;

- Digite o texto "Input_Analog" no campo "Name";
- Defina o tipo de dados como WORD (valor analógico);
- No campo "Usage", selecione a entrada "VAR_GLOBAL" porque a variável será usada para transferir seu valor para outras ferramentas de software;
- Selecione a "PDD" checkbox.



Quando a caixa de seleção "PDD" está ativada, os dados do processo são transferidos para programas de visualização (WebVisit por exemplo). Os dados do processo são transferidos usando um arquivo CSV.



- Confirme suas entradas clicando em "OK";
- Insira a variável "Output_Analog".
- Conecte ambas as variáveis;

Atribuindo dados do processo

Os dados de processo e as variáveis de controle são atribuídos na área de trabalho do *Process Data Workspace*.

- Mude para *Process Data Workspace*.

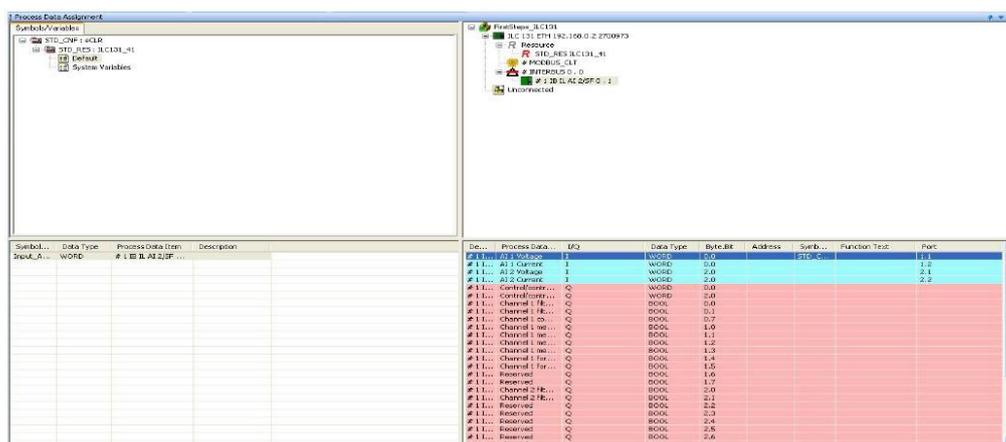
Vincule os dados do processo do cartão de entrada analógica à variável "Input_Analog" (vide figura abaixo):

- Clique na controladora na janela superior direita. A configuração padrão é exibida na janela superior esquerda ("*Symbols/Variables*");
- Na janela superior esquerda, "*Symbols/Variables*", clique no "STD_RES: ILC131_44";
- Na janela superior direita, clique no dispositivo para o qual você gostaria de vincular os dados do processo às variáveis (exemplo: IB IL AI 2 / SF-ME);
- Clique no item de dados do processo a ser vinculado (exemplo: ponto terminal 1.1).

Uma vez que a variável "Input_Analog" foi criada como uma variável global (VAR_GLOBAL), ela é exibida na janela inferior esquerda:

- Use o "arraste e solte" para vincular o item de dados do processo "ponto terminal 1.1" selecionado à variável "Input_Analog".

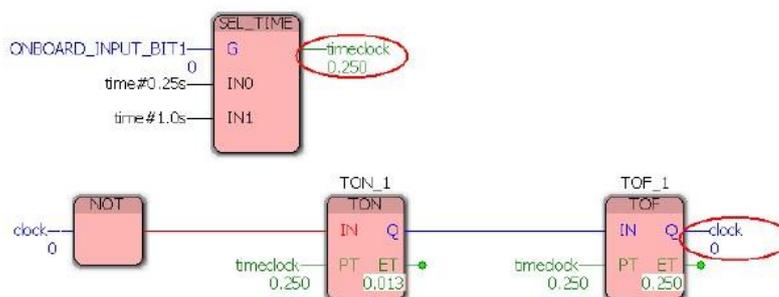
Na janela inferior esquerda, do *Process Data Workspace*, entre a variável e o item de dados do processo é exibida.



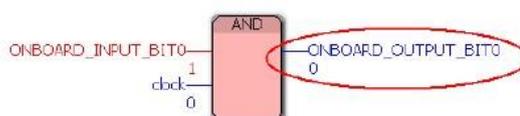
- Compile o programa para o controlador, realizando uma cold restart e execute o teste do programa.

O status do programa da POU "Main" é exibido na planilha "Main: Main" (vide figura abaixo). Os interruptores 0 e 1 do módulo de comutação e / ou o potenciômetro podem ser ativados para modificar os estados das variáveis exibidas:

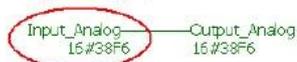
(* Generating a variable clock*)



(* AND relation between an onboard input variable and a clock variable (1 second) connected to an onboard output variable*)

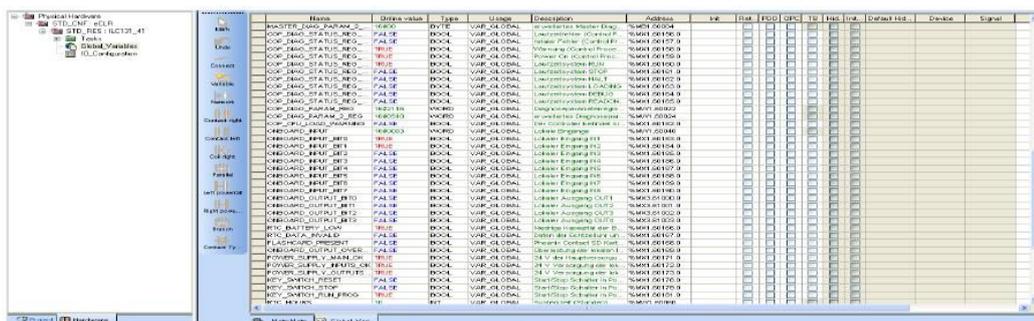


(* Connecting an analog input variable with an analog output variable*)



Mude para o espaço de trabalho *IEC Programming workspace* (Only PC Worx Professional).

- Abra "*Hardware*" clicando na guia "*project tree window*";
- Para exibir as variáveis globais, clique duas vezes em "*Global_Variables*";
- A tabela que está aberta exibe o status das variáveis globais (ilustração abaixo);



Os seguintes estados são exibidos neste espaço de trabalho, por exemplo:

Variable	Online value	Description
ONBOARD_INPUT_BIT0	TRUE	Switch 0 ON
ONBOARD_INPUT_BIT1	TRUE	Switch 1 ON
Input_Analog	16#445B	Analog input value at input I1
ONBOARD_OUTPUT_BIT0	TRUE	Clock signal at output Q1 of the controller

- Abra a "Project Tree" clicando na guia "Project";
- Clique duas vezes na POU "Main" e depois no ícone "Variables Worksheet" na barra de ferramentas.
- O status das variáveis usadas na POU "Main" é exibido na tabela que se abre.



Os seguintes estados são exibidos neste espaço de trabalho, por exemplo.

Variable	Online value	Description
ONBOARD_INPUT_BIT0	TRUE	Switch 0 ON
ONBOARD_OUTPUT_BIT0	TRUE	Clock signal at output Q1 of the controller
clock	TRUE	Variable (clock signal)
TON_1	-	Instance of function block TON (switch-on delay)
TOF_1	-	Instance of function block TOF (switch-off delay)
timeclock	1000	Variable (clock frequency)
ONBOARD_INPUT_BIT1	TRUE	Switch 1 ON
Input_Analog	16#445A	Analog input value at input I1
Output_Analog	16#445A	Analog variable that is described with the value of the Input_Analog variable