

Software de Programação

Atos A1

MA.007.00-05/10

Manual de utilização

2010



Este manual não pode ser reproduzido, total ou parcialmente, sem autorização por escrito da **Schneider Electric**.

Seu conteúdo tem caráter exclusivamente técnico/informativo e a **Schneider Electric** se reserva no direito, sem qualquer aviso prévio, de alterar as informações deste documento.

Termo de Garantia

A **Schneider Electric Brasil Ltda.** assegura ao comprador deste produto, garantia contra qualquer defeito de material ou de fabricação, que nele apresentar no prazo de 360 dias contados a partir da emissão da nota fiscal de venda.

A **Schneider Electric Brasil Ltda.** restringe sua responsabilidade à substituição de peças defeituosas, desde que o critério de seu Departamento de Assistência Técnica, se constate falha em condições normais de uso. A garantia não inclui a troca gratuita de peças ou acessórios que se desgastem naturalmente com o uso, cabos, chaves, conectores externos e relés. A garantia também não inclui fusível, baterias e memórias regraváveis tipo EPROM.

A **Schneider Electric Brasil Ltda.** declara a garantia nula e sem efeito se este produto sofrer qualquer dano provocado por acidentes, agentes da natureza, uso em desacordo com o manual de instruções, ou por ter sido ligado à rede elétrica imprópria, sujeita a flutuações excessivas, ou com interferência eletromagnética acima das especificações deste produto. A garantia será nula se o equipamento apresentar sinais de ter sido consertado por pessoa não habilitada e se houver remoção e/ou alteração do número de série ou etiqueta de identificação.

A **Schneider Electric Brasil Ltda.** somente obriga-se a prestar os serviços referidos neste termo de garantia em sua sede em São Paulo - SP, portanto, compradores estabelecidos em outras localidades serão os únicos responsáveis pelas despesas e riscos de transportes (ida e volta).

- Serviço de Suporte **Schneider Electric**

A **Schneider Electric** conta com um grupo de técnicos e engenheiros especializados aptos para fornecer informações e posicionamentos comerciais, esclarecer dúvidas técnicas, facilitar e garantir serviços técnicos com qualidade, rapidez e segurança..

Com o objetivo de criar um canal de comunicação entre a **Schneider Electric** e seus usuários, criamos um serviço denominado AssisT. Este serviço centraliza as eventuais dúvidas e sugestões, visando a excelência dos produtos e serviços comercializados pela **Schneider Electric**.

Este serviço está permanentemente disponível com uma cobertura horária das 7h30m às 18h, com informações sobre plantão de atendimento técnico durante os fins de semana e feriados, tudo que você precisa fazer é ligar para 0800 7289 110. O AssisT apresentará rapidamente a melhor solução, valorizando o seu precioso tempo.



Para contato com a **Schneider Electric** utilize o endereço e telefones mostrados atrás deste Manual.

CONVENÇÕES UTILIZADAS

- Palavras em outras línguas são apresentadas em *itálico*, porém algumas palavras são empregadas livremente por causa de sua generalidade e frequência de uso. Como, por exemplo, às palavras *software* e *hardware*.

Números seguidos da letra h subscrita (ex:1024h) indicam numeração hexadecimal e seguidos da letra b (ex:10b), binário. Qualquer outra numeração presente deve ser interpretada em decimal.

- O destaque de algumas informações é dado através de ícones localizados sempre à esquerda da página. Cada um destes ícones caracteriza um tipo de informação diferente, sendo alguns considerados somente com caráter informativo e outros de extrema importância e cuidado. Eles estão identificados mais abaixo:



NOTA: De caráter informativo, mostra dicas de utilização e/ou configuração possíveis, ou ressalta alguma informação relevante no equipamento.



OBSERVAÇÃO: De caráter informativo, mostra alguns pontos importantes no comportamento / utilização ou configuração do equipamento. Ressalta tópicos necessários para a correta abrangência do conteúdo deste manual.



IMPORTANTE: De caráter informativo, mostrando pontos e trechos importantes do manual. Sempre observe e analise bem o conteúdo das informações que são identificadas por este ícone.



ATENÇÃO: Este ícone identifica tópicos que devem ser lidos com extrema atenção, pois afetam no correto funcionamento do equipamento em questão, podendo até causar danos à máquina / processo, ou mesmo ao operador, se não forem observados e obedecidos.

Índice

CAPÍTULO 1.....	17
A Norma IEC61131-3	17
Convenção da nomenclatura.....	17
Representação das variáveis	17
Diferenças na nomenclatura entre as famílias	18
Nomenclatura utilizada na família Atos MPC4004BF	18
Nomenclatura utilizada na família Atos MPC6006	20
Nomenclatura utilizada na família Atos Expert BF	22
Program Organization Unit (POU).....	24
Projeto de forma hierárquica	25
Variáveis de escopo Local e Global	25
Visão Geral.....	26
O que é um Program?	26
O que é uma Função (Function)?	27
O que é um Bloco de Função (Function Block)?.....	27
O que é uma Tarefa (Task)?	27
Tarefas (Tasks) preemptivas e não-preemptivas	28
Tarefa (Task) não-preemptiva	28
Tarefa (Task) preemptiva	29
Tipos de dados	30
Declaração de constantes	31
CAPÍTULO 2.....	35
Sobre o software Atos A1 Soft	35
Configurações mínimas para rodar o Atos A1Soft.....	35
Como utilizar os módulos na programação?	35
Configurando hardware	36

Atos MPC4004BF	36
Jumpers de endereçamento	37
Regras para a inclusão de unidades no(s) bastidor(es)	39
Atos MPC6006L	39
Propriedades do Hardware	40
Módulos com troca quente	40
Configurando entradas da CPU	41
Configurando saídas da CPU	43
Verificação de Hardware	45
Identificação Automática dos módulos	46
Atualizando entradas e saídas	48
Trocar de controlador	49
Tabela de equivalência de módulos	50
Gerenciador do Projeto	53
Atalhos de Teclado	54
Comandos principais	54
Editor Ladder	54
Projeto	54
Ferramentas	54
Gerenciador de Variáveis	55
Barra de Menu	56
Menu Arquivo	56
Menu Editar	56
Menu Exibir	56
Menu Projeto	56
Menu Ferramentas	56
Menu <i>Ajuda</i>	57
Configurando conexão do controlador	57
Tarefas Preemptivas	59
Tarefa Preemptiva por evento	59
Atos MPC4004BF	59

Atos MPC6006L	62
Tarefa preemptiva por tempo	64
Download/Upload	67
Realizando Download do projeto	67
Realizando Upload do projeto	68
Fonte do projeto	70
Configuração de Memória	71
Visualização dos Recursos Utilizados.....	73
Declaração de Array	75
Declaração de String	77
Struct – Estrutura de dados.....	78
Criando uma Struct.....	78
Inserindo Membros.....	79
Usando Struct.....	80
Array de Struct.....	81
Declaração no ladder	82
Declaração do Array no ladder	83
Programação Online.....	83
Alterações em programação Online:	83
Ladder:	84
OPÇÕES GENÉRICAS	84
PROGRAMA (PROGRAM).....	84
FUNÇÃO (FUNCTION) DO SISTEMA.....	84
FUNÇÃO (FUNCTION) DO USUÁRIO	85
BLOCOS FUNCIONAIS (FUNCTIONS BLOCKS) DO SISTEMA.....	85
BLOCOS FUNCIONAIS (FUNCTIONS BLOCKS) DO USUÁRIO	85
Supervisão.....	87
O que é supervisão?	87
Supervisão Ladder	88
Supervisão de Variáveis.....	89
Supervisão gráfica.....	90

Force de variáveis	92
Atribuir valor	94
Simulador	95
O que é o simulador	95
Simulando um projeto.....	98
Guia “Simulador de I/O”	98
Restrições do Simulador	101
Segurança do Projeto.....	102
Ferramentas do CLP	103
Como acessar?	103
Interface de Comunicação	104
CLP Info.....	105
Firmware.....	106
Data/Hora	107
Dump de Memória	108
Bin Pré-Compilado	109
Troca global de variáveis	110
Gerenciador de Variáveis Globais.....	112
Biblioteca de usuário	114
Exportando Funções, Blocos Funcionais e Tipos de dados	114
Visualizando biblioteca de usuário	117
Importação e Exportação de variáveis para o Excel.....	118
Exportando variáveis para o Excel.....	118
Exemplo de arquivo exportado:.....	120
Importando variáveis do Excel	121
Configurações do Atos A1 Soft	124
Backup.....	124
Backup de arquivo.....	125
Exemplo para quantidade de backups = 3:	125
Função Auto Save	126
Referência Cruzada.....	126

CAPÍTULO 3	129
Configuração de IHM.....	129
Propriedades da IHM.....	130
Tipos de IHM	131
Contraste da tela	131
Alarmes da IHM.....	132
Receitas.....	133
Configuração dos Frames	133
Receitas via IHM	134
Armazenar	134
Recuperar	134
Senha da IHM.....	135
Função Senha	136
Inserir TELA.....	137
Propriedades da Tela	138
Programação de Telas	139
Campo Edit/View	139
Campo Bargraph	144
Campo True/False	145
Campo Selector	146
Campo StringList	147
String associada a uma variável	147
String associada a uma seqüência de variáveis booleanas.....	147
Editor de String	148
Campo Animation	149
Style Constante	150
Style Variável.....	150
Texto.....	150

Navegação das telas	151
Tela Zero	151
Teclas da IHM	151
Programa de usuário	152
Navegação das telas	152
Teclas K.....	155
Teclas F.....	156
Auxílio Manutenção	157

CAPÍTULO 4..... 159

Como começar	159
Barra de ferramentas Ladder	159
Inserindo Blocos e Funções	161
Variáveis no ladder	163
Adicionando variáveis no ladder	163
Filtrando Variáveis no Ladder	165
Função (Function)	167
Inserindo Função	167
Programação da Função	167
Declaração de parâmetros	168
Declaração de variáveis locais	169
Tabela da norma IEC 61131	170
Programação no Ladder.....	170
Utilizando Função.....	171
Bloco Funcional (<i>Function Block</i>)	172
Inserindo Bloco Funcional	172
Programação do Bloco Funcional	173
Declaração de parâmetros	173
Declaração de variáveis locais	174

Programação no Ladder	175
Utilizando Bloco Funcional	175

CAPÍTULO 5.....177

Access Path.....	177
Canais de comunicação	177
Configuração dos canais	178
RS232 – Serial A	178
RS485 – Serial B	179
APR03 – Slave de comunicação	180
Modbus – Slave de comunicação	180
Modo Slave.....	181
Modo Mestre.....	182
Códigos de erro ou falha	184
Códigos APR03.....	184
Códigos Modbus.....	184
Configuração de Frames	185
Frames de comunicação	185
Propriedades do frame APR03	185
Propriedades do frame Modbus	186
Frame Broadcast	187
Frame Modbus para tipo de dado de 4 bytes.....	187
Formato de comunicação de dado tipo 4 bytes	188
Comunicando STRUCT via Modbus	190
Situações especiais.....	191
Membro Booleano:	191
Membro Byte:	192
Membros Bool e Byte:	192
Formas de atualização de Frame.....	193

Atualização Cíclica da CPU	193
Atualização por tempo definido	194
Atualização por Evento	195
Atualização por Tempo e Evento	196
Funções Modbus	197
Read Coil Status - (0x01)	197
Read Input Status - (0x02)	198
Read Holding Registers - (0x03)	198
Read Input Registers - (0x04)	199
Force Single Coil - (0x05).....	200
Preset Single Register - (0x06)	201
Force Multiple Coils - (0x0F)	202
Preset Multiple Registers - (0x10)	203
Range de endereçamento MODBUS	204
Exception Response (ERROR)	205
CRC - Cyclic Redundacy Check	205
Verificando configurações de comunicação via IHM.....	205
Módulo de comunicação	208
Slave de comunicação 4004.72M	210
Slave de comunicação 4004.72R.....	212
Slave de comunicação 4004.72P	213
Conexões do módulo Slave:	214
Slave de comunicação 6006.72	216
Módulo de comunicação 6006.72C.....	220
Configuração dos dispositivos na rede CANopen.....	220
Adicionando um dispositivo na rede:.....	220
Configurando um dispositivo	223
Configurando um PDO	225
Configurando um SDO	226

CAPÍTULO 1

IEC61131-3

A Norma IEC61131-3

Convenção da nomenclatura

Representação das variáveis

Sinal inicial (IEC std)	Identificação de memória	Tamanho do dado		DESCRIÇÃO
%	M (Acesso à memória) I (Entrada física do CLP) Q (Saída física do CLP)	X	(1 bit)	Acesso às variáveis booleanas.
		W	(16 bits)	Acesso às variáveis com 16 bits de tamanho: INT, UINT e WORD.
		D	(32 bits)	Acesso às variáveis com 32 bits de tamanho: DINT, UDINT, DWORD,
		T	(32 bits)	TIME, DATE, TOD e DATE_AND_TIME.
		R	(32 bits)	Acesso às variáveis com 32 bits de tamanho: REAL
		A	-----	O conteúdo dessa região é definido pelo usuário conforme a necessidade do projeto.

Exemplos de acesso a dados utilizando a nomenclatura IEC61131-3:

%I0.0	Leitura de sinal da entrada E0 da CPU (slot 0, entrada 0)
%IW8.1	Leitura de sinal da entrada proveniente de uma placa analógica (slot 8, entrada 1)
%Q0.0	Escrita de sinal na saída S0 da CPU (slot 0, saída 0)
%QW8.1	Escrita de sinal na saída proveniente de uma placa analógica (slot 8, saída 1)
%MX1	Acesso à leitura/escrita de variável booleana na memória do CLP
%MW1	Acesso à leitura/escrita de variável de 16 bits na memória do CLP
%MD1	
%MT1	Acesso à leitura/escrita de variável de 32 bits na memória do CLP
%MR1	
%MA1	Acesso à leitura/escrita de variável na memória do CLP



Ao acessar entradas e saídas físicas da CPU, a nomenclatura do tamanho do dado (X) não é utilizada, sendo necessário apenas informar a posição física (slot, posição) que o dado se encontra. A definição dos slots é feita ao montar a configuração de hardware no aplicativo.

Diferenças na nomenclatura entre as famílias

Nomenclatura utilizada na família Atos MPC4004BF

Para manter a compatibilidade da família Atos MPC4004 em relação às películas de polycarbonato dos módulos, foi adotado para o controlador Atos MPC4004BF o início de endereçamento das variáveis de E/S em 0 (base hexadecimal), por exemplo: E0, EA, S1, SB etc.

O endereçamento da família Atos MPC4004 é feito através de um jumper que varia de A até H. Algumas unidades digitais e analógicas receberam um jumper adicional, denominado STG ou Jumper de Grupo. Este jumper permite que haja um aumento da quantidade de pontos digitais e analógicos.

O slot referente ao módulo é a sua posição no bastidor, essa posição é definida através do jumper ST1.

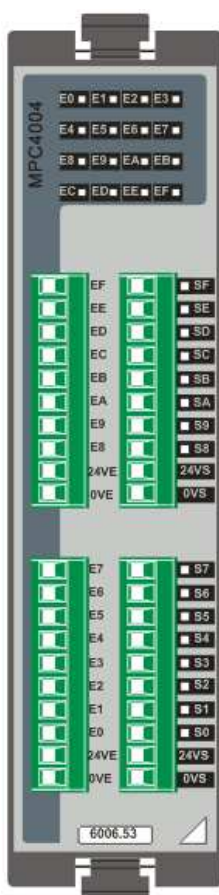
Sinal inicial (IEC)	Localidade da variável	Tamanho do dado		1º dígito	2º dígito
%	I	----	(1 bit)	Slot	Posição da entrada / saída
	Q	W	(16 bits)		

Exemplos:

%I0.0	Leitura de sinal da entrada E0 da CPU (slot 0, entrada 0)
%IW8.1	Leitura de sinal da entrada proveniente de uma placa analógica (slot 8, entrada 1)
%Q0.0	Escrita de sinal na saída S0 da CPU (slot 0, saída 0)
%QW8.1	Escrita de sinal na saída proveniente de uma placa analógica (slot 8, saída 1)

Assim como podemos verificar na imagem a seguir, a nomenclatura da família Atos MPC4004 utiliza a base hexadecimal denominando a letra **E** para as entradas e **S** para as saídas.

A tabela abaixo mostra a nomenclatura das entradas e saídas da família Atos MPC4004, considerando o módulo no slot 1.



Entrada	Nomenclatura IEC61131	Saída	Nomenclatura IEC61131
E0	%I1.0	S0	%Q1.0
E1	%I1.1	S1	%Q1.1
E2	%I1.2	S2	%Q1.2
E3	%I1.3	S3	%Q1.3
E4	%I1.4	S4	%Q1.4
E5	%I1.5	S5	%Q1.5
E6	%I1.6	S6	%Q1.6
E7	%I1.7	S7	%Q1.7
E8	%I1.8	S8	%Q1.8
E9	%I1.9	S9	%Q1.9
EA	%I1.A	SA	%Q1.A
EB	%I1.B	SB	%Q1.B
EC	%I1.C	SC	%Q1.C
ED	%I1.D	SD	%Q1.D
EE	%I1.E	SE	%Q1.E
EF	%I1.F	SF	%Q1.F

Nomenclatura utilizada na família Atos MPC6006

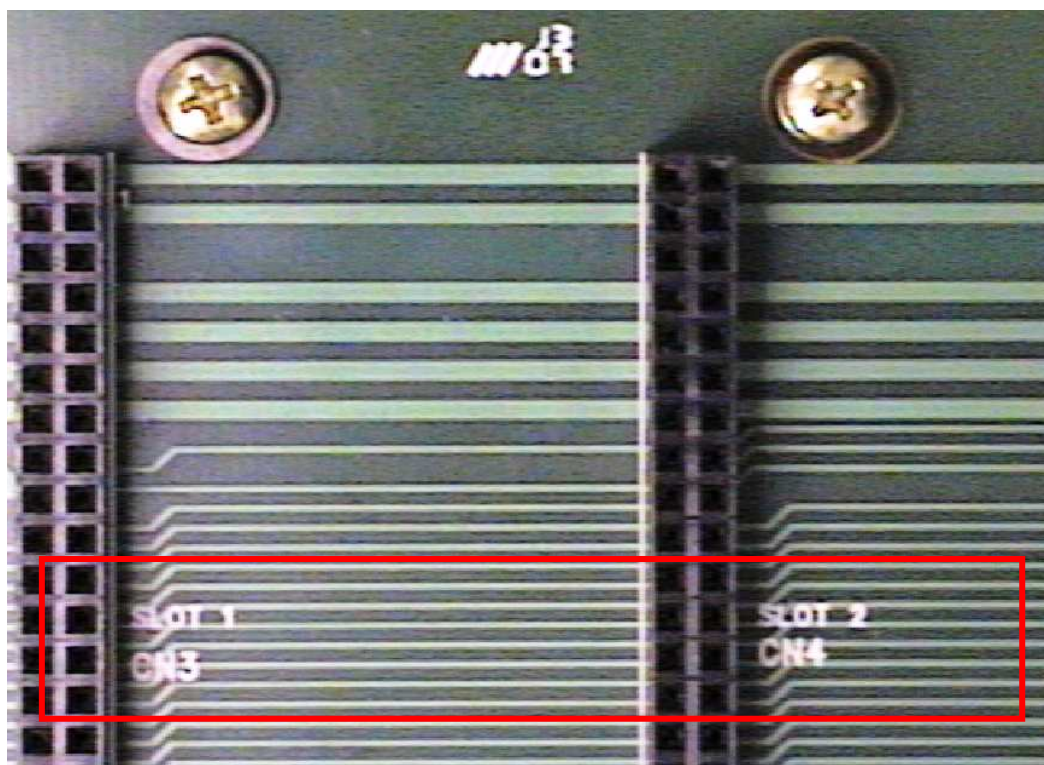
Para a família Atos MPC6006 foi alterada a nomenclatura dos endereços de entradas e saídas do controlador, possibilitando através da inserção de mais um dígito, a informação do bastidor em que se encontra o módulo.

Verifica-se também uma modificação no início do endereçamento das variáveis de E/S, que passam a iniciar em 1 (base decimal), por exemplo: I1, I13, Q6, Q15, etc.

Nessa nova configuração temos a seguinte estrutura de endereçamento:

Sinal inicial (IEC)	Localidade da variável	Tamanho do dado		1º dígito	2º dígito	3º dígito
%	I Q	Nenhum	(1 bit)	Bastidor	Slot	Posição da entrada / saída
		W	(16 bits)			
		D	(32 bits)			

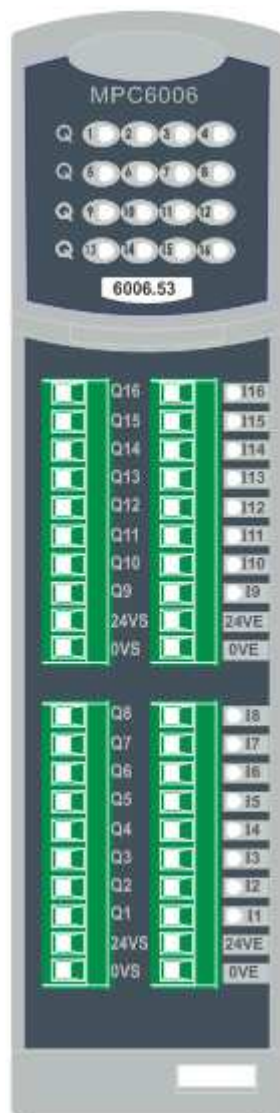
A família Atos MPC6006 não possui jumpers de configuração de posição. O passo referente ao módulo é a sua posição física no bastidor, essa posição é visualizada no backplane, conforme mostra a imagem a seguir:



Exemplos:

%I1.1.1	Entrada digital I1 (bastidor 1, slot 1, entrada 1)
%IW2.3.2	Entrada de uma placa analógica ou de temperatura (bastidor 2, slot 3, entrada 2)
%ID1.1.2	Efetivo de contagem dos contadores da CPU (bastidor 1, slot1, efetivo de contagem 2)
%Q1.1.8	Saída digital Q8 (bastidor 1, slot 1, saída 8)
%QW1.5.3	Saída de uma placa analógica ou de temperatura (bastidor 1, slot 5, saída 3)

Na família Atos MPC6006 o endereçamento é mostrado de forma decimal utilizando a letra **I** para as entradas e **Q** para as saídas, conforme pode ser verificado na figura e na tabela a seguir.



Entrada	Nomenclatura IEC61131	Saída	Nomenclatura IEC61131
I1	%I1.2.1	Q1	%Q1.2.1
I2	%I1.2.2	Q2	%Q1.2.2
I3	%I1.2.3	Q3	%Q1.2.3
I4	%I1.2.4	Q4	%Q1.2.4
I5	%I1.2.5	Q5	%Q1.2.5
I6	%I1.2.6	Q6	%Q1.2.6
I7	%I1.2.7	Q7	%Q1.2.7
I8	%I1.2.8	Q8	%Q1.2.8
I9	%I1.2.9	Q9	%Q1.2.9
I10	%I1.2.10	Q10	%Q1.2.10
I11	%I1.2.11	Q11	%Q1.2.11
I12	%I1.2.12	Q12	%Q1.2.12
I13	%I1.2.13	Q13	%Q1.2.13
I14	%I1.2.14	Q14	%Q1.2.14
I15	%I1.2.15	Q15	%Q1.2.15
I16	%I1.2.16	Q16	%Q1.2.16



A tabela acima considerou o módulo no bastidor 1 e passo 2.

Nomenclatura utilizada na família Atos Expert BF

Para a família Atos Expert BF a nomenclatura dos endereços de entradas e saídas do controlador são parecidas com a família Atos MPC6006, com diferença apenas nos dois primeiros dígitos.

Nessa nova configuração temos a seguinte estrutura de endereçamento:

Sinal inicial (IEC)	Localidade da variável	Tamanho do dado		1º dígito	2º dígito	3º dígito
%	I Q	Nenhum	(1 bit)	Sempre 1	1 – Digital 2 – Analógico	Posição da entrada / saída
		W	(16 bits)			
		D	(32 bits)			

Exemplos:

Endereço	I Q	Nenhum W D	1º Dígito	2º Dígito	3º Dígito
%I1.1.1	Entrada	1 bit	1	1 – Digital	Entrada digital I1
%IW1.2.2	Entrada	W – 16 bits	1	2 - Analógica	Entrada Analógica +I2 / -I2
%ID1.1.2	Entrada	D – 32 bits	1	1 – Digital	Efetivo de contagem dos contadores da CPU (efetivo de contagem 2)
%Q1.1.8	Saída	1 bit	1	1 – Digital	Saída digital 8 – Q8
%QW1.2.3	Saída	W – 16 bits	1	2 – Analógica	Saída analógica +Q3 / -Q3

Top view of the E2850-20/0S board showing the connector layout. The board is dark blue with a lighter blue border. The top edge features a 24V power header, a 0V ground header, and a 24VS header. The bottom edge features a 24VE power header, a 0VE ground header, and a 24VS header. The left edge features a 24V power header, a 0V ground header, and a 24VS header. The right edge features a 24V power header, a 0V ground header, and a 24VS header. The board is populated with various components including resistors, capacitors, and integrated circuits.

Entrada	Nomenclatura IEC61131	Saída	Nomenclatura IEC61131	Entrada	Nomenclatura IEC61131	Saída	Nomenclatura IEC61131
I1	%I1.1.1	Q1	%Q1.1.1	I9	%I1.1.9	Q9	%Q1.1.9
I2	%I1.1.2	Q2	%Q1.1.2	I10	%I1.1.10	Q10	%Q1.1.10
I3	%I1.1.3	Q3	%Q1.1.3	I11	%I1.1.11	+Q1 e -Q1	%Q1.2.1
I4	%I1.1.4	Q4	%Q1.1.4	I12	%I1.1.12	+Q2 e -Q2	%Q1.2.2
I5	%I1.1.5	Q5	%Q1.1.5	I13	%I1.1.13		
I6	%I1.1.6	Q6	%Q1.1.6	I14	%I1.1.14		
I7	%I1.1.7	Q7	%Q1.1.7	+I1 e -I1	%I1.2.1		
I8	%I1.1.8	Q8	%Q1.1.8	+I2 e -I2	%I1.2.2		



Como o endereçamento acima se trata de um combo não expansível de entradas e saídas, não é necessário considerar o número do bastidor, nem do slot.

Program Organization Unit (POU)

A norma IEC61131-3 descreve os programas (*Programs*), funções (*Functions*) e blocos de função (*Function Blocks*) como sendo POUs (*Program Organization Units*).

Estes componentes permitem que você estruture e organize seu projeto de maneira a minimizar a escrita de códigos e a manutenção de todo o sistema.

A criação e utilização de funções (*Functions*) e blocos de função (*Function Blocks*) cria uma portabilidade de código entre projetos, de maneira que o programador não precisa digitar "n" vezes o mesmo código quando existe a necessidade de utilizá-lo várias vezes no mesmo projeto.

O programa (*Program*) é um POU utilizado para estruturar o projeto. É dentro dele que todas as funções (*Functions*) e blocos de função (*Function Blocks*) serão chamados para compor a aplicação.

A norma IEC estimula a reutilização desde o nível mais macro, com programas (*Programs*), até o nível mais micro, com funções (*Functions*) e blocos de função (*Function Blocks*). A recursividade não é permitida dentro de uma POU, para garantir a estabilidade da aplicação.

Tipo de POU	Aplicado como	Comentário
Programa	Instância de um Programa	Permite reutilização no nível macro, como programas para reatores, transportadores, caldeiras, etc.
Bloco Funcional	Instância de um Bloco Funcional	Possibilita a reutilização desde simples a complexas estratégias de controle e algoritmos, como controle PID, filtros, motores, etc.
Função	Função	Usada para tratamento comum de dados, como lógica E, OU, seno, cosseno, soma e etc.

Projeto de forma hierárquica

Uma importante característica da norma é a ênfase dada ao projeto de forma hierárquica.

Isto significa que um sistema de controle pode ser dividido em níveis de complexidade de cima para baixo e estruturado de baixo para cima em função das necessidades de controle.

Isto implica na utilização de funções (*Functions*) e blocos de função (*Function Blocks*) e padrões, através de bibliotecas que podem ser fornecidas pelo fabricante do CLP, ou pela criação de blocos e funções específicos definidos pelo usuário para atender às necessidades do processo.

O uso de bibliotecas baseadas nos blocos padrões permite a portabilidade de soluções para diferentes sistemas de controle.

Variáveis de escopo Local e Global

A norma exige a declaração de variáveis dentro de diferentes elementos de software, tais como programas (*Programs*) e blocos de função (*Function Blocks*). As variáveis podem utilizar nomes com significado abrangente (simbólicos) e serem de diferentes tipos de dados. As variáveis podem ser de alocação dinâmica e associadas a posições de memória (representação direta). O escopo das variáveis é local ao elemento de software que as declara, permitindo acesso dentro do próprio elemento que pode ser uma configuração, recurso, programa (*Program*), bloco de função (*Function Block*) ou função (*Function*). Variáveis também podem ser de escopo global, sendo acessadas por todos os elementos contidos no mesmo, incluindo os elementos aninhados.

Variáveis de representação direta:

Posições de memória do CLP podem ser acessadas usando variáveis de representação direta. A representação direta permite a leitura e escrita de dados em posições conhecidas de memória, tais como entradas, saídas e endereços internos. As variáveis de representação direta têm seu uso restrito aos programas (*Programs*), não permitindo que blocos de função (*Function Blocks*) façam acesso direto para garantir a reutilização dos blocos. A notação utilizada é padronizada para permitir a portabilidade.

Visão Geral

A representação de variáveis segundo a norma IEC61131-3 permite que os diferentes tipos de dados existentes no controlador possam ser acessados de maneira clara e objetiva.



Todo o mapeamento de memória pode ser acessado pelo Gerenciador Global de Variáveis.

O que é um Program?

Um programa (*Program*) IEC pode ser construído a partir de diferentes elementos de software.

Tipicamente, um programa (*Program*) consiste de um número de blocos de função interconectados, capazes de trocar dados através das conexões de software. Um programa (*Program*) pode acessar as variáveis do CLP e comunicar com outros programa (*Program*). A execução de diferentes partes de um programa (*Program*), blocos de função (*function blocks*), por exemplo, pode ser controlada usando tarefas (*Tasks*).

As principais diferenças de um programa (*Program*) em relação aos blocos de função (*Function Blocks*) são:

- Programas (*Programs*) podem conter declarações de variáveis de endereçamento direto, ou seja, endereçamento direto de pontos de entrada e saída.
- Programas (*Programs*) podem conter declarações de variáveis globais, as quais podem ser acessíveis pelos blocos de função (*Function Blocks*) através do uso de variáveis externas.
- Programas (*Programs*) podem conter variáveis de acesso, as quais permitem o acesso remoto pelos serviços de comunicação.
- Programas (*Programs*) podem conter instâncias de blocos funcionais, mas não de outros programas, isto é, estes não podem ser aninhados.
- As instâncias de blocos de função de um programa podem ser executadas por diferentes tarefas de controle.
- Programas (*Programs*) podem ser instanciados somente dentro de recursos.

O que é uma Função (*Function*)?

Funções (*Functions*) são elementos de software que não aparecem no modelo de software. Funções (*Functions*) não possuem persistência, existindo apenas em tempo de execução, assim como sub-rotinas. Portanto, não possuem estados internos, ou seja, sempre produzem o mesmo resultado para o mesmo conjunto de entradas.

Funções (*Functions*) podem ter apenas uma saída, sem considerar a saída ENO para controle de execução, ao contrário dos blocos de função (*Function Blocks*) que podem ter várias. O resultado pode ser um tipo de dado simples de múltiplos elementos (vetores e estruturas).

As funções não são instanciáveis, isto é, só existem em tempo de execução, não sendo necessário um identificador para alocação de memória para dados. As funções trigonométricas são os tipos mais comuns de funções.

O que é um Bloco de Função (*Function Block*)?

O conceito de bloco de função (*Function Block*) é um dos mais importantes da norma IEC61131-3, para permitir o projeto de software de forma hierárquica e estruturada. Blocos de função (*Function Blocks*) podem ser utilizados para a criação de elementos de software totalmente reutilizáveis, desde a criação de outros blocos de função (*Function Blocks*) mais simples, até programas (*Programs*) complexos.

As principais características dos blocos de função (*Function Blocks*) são que estes possuem um conjunto de dados, os quais podem ser alterados por um algoritmo interno. Somente o conjunto de dados é mantido na memória para uma determinada instância do bloco de função (*Function Block*).

Os dados possuem persistência, ou seja, possuem estados internos que são mantidos entre uma execução e outra. Blocos de função (*Function Blocks*) podem ser utilizados para a criação de outros blocos de função (*Function Blocks*) (blocos derivados) aumentando ainda mais a capacidade de reutilização do software.

O que é uma Tarefa (*Task*)?

Uma tarefa (*Task*) é um mecanismo de escalonamento muito útil para sistemas de tempo real, que executa programas (*Programs*) ou blocos de função (*Function Blocks*) periodicamente ou em resposta a um evento (mudança de estado de alguma variável booleana), permitindo a execução de programas em diferentes taxas.

A necessidade de se executar programas em taxas diferentes tem por objetivo atender as exigências de tempo de resposta do processo sob controle e de otimizar o uso da capacidade de processamento do CLP.

Um forno, por exemplo, que possui uma capacidade térmica muito grande, pode ser controlado por um algoritmo que executa uma vez a cada minuto, permanecendo estável. Já as funções de intertravamento de segurança de uma máquina ferramenta devem ser executadas a cada 5 ms.

A cada tarefa podemos atribuir um período de execução e uma prioridade.

A norma IEC assume tarefas em diferentes recursos sempre executam de forma independente. Entretanto, em algumas implementações, pode ser necessária a utilização de mecanismos de sincronização.

Neste contexto, uma simples execução de um programa (*Program*) ou bloco de função (*Function Block*) implica que todos os elementos de software dentro dos mesmos são processados uma vez.

A norma IEC não define nenhum mecanismo implícito para execução de programas (*Programs*). Ou seja, um programa (*Program*) ou bloco de função (*Function Block*) ficará aguardando a sua execução até que seja associado a uma determinada tarefa (*Task*) e esta seja ativada por uma execução periódica ou por um determinado evento.

Um bloco de função (*Function Block*) não devidamente associado a uma tarefa, será executado na mesma tarefa (*Task*) do programa (*Program*) onde esteja contido, ou seja, programa (*Program*) pai.

Existem 2 tipos diferentes de tarefa (*Task*): **Preemptivas** (ou periódicas) e **Não-Preemptivas** (não-periódicas), sendo estas divididas em 3 subcategorias: **Cíclica**, **Tempo** e **Evento**.

Tarefas (*Tasks*) preemptivas e não-preemptivas

Quando existem tarefas múltiplas, normalmente são atribuídos diferentes intervalos e prioridades para cada uma. Para permitir o funcionamento das tarefas, podem ser utilizados dois métodos de escalonamento, o preemptivo e o não-preemptivo. O método adotado por um CLP pode alterar significativamente o comportamento do sistema.

Tarefa (*Task*) não-preemptiva

Neste tipo de escalonamento uma tarefa sempre completa seu processamento, uma vez iniciado. Quando a tarefa termina, a tarefa de maior prioridade à espera do processador é escalonada. Caso haja empate na prioridade, a tarefa que está esperando há mais tempo é escalonada. Após a sua execução, uma tarefa só será escalonada, quando o seu intervalo de execução se esgotar. O intervalo entre a execução de tarefas pode variar muito neste tipo de escalonamento.

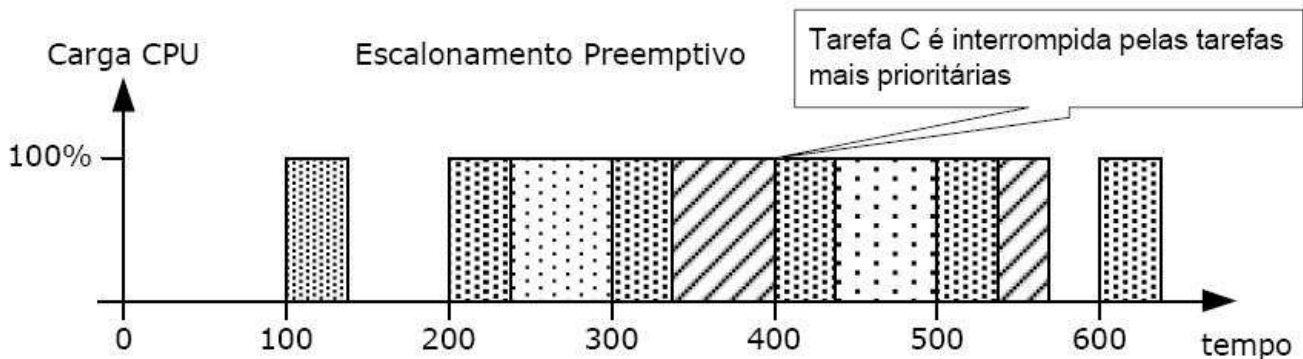
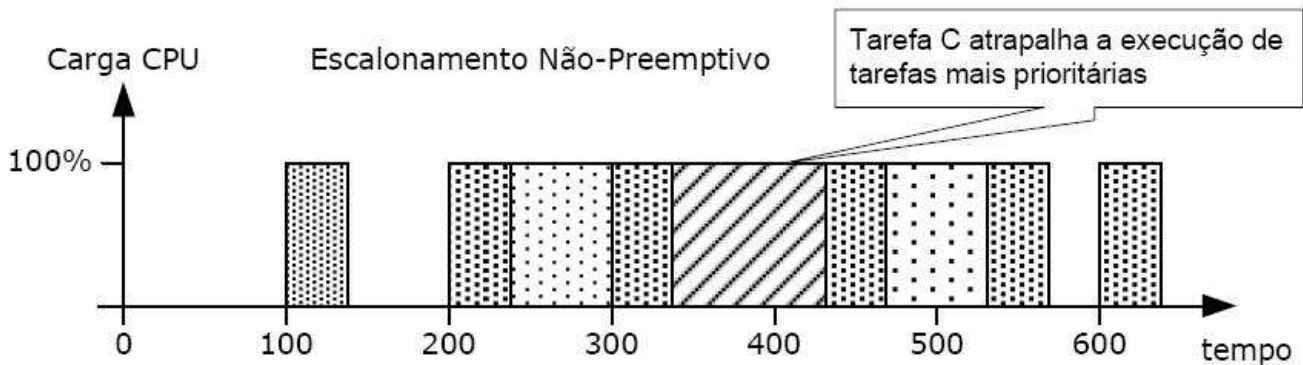
Uma tarefa que demore um pouco mais em um loop, irá atrasar todas as demais tarefas. Isto torna impossível prever com exatidão quando uma determinada tarefa será executada e caracteriza o sistema como não determinístico. Apresentado uma pior característica para aplicação em sistemas de controle.

- Não-preemptiva CÍCLICA: Executada toda vez que a tarefa for escalonada;
- Não-preemptiva por TEMPO: A tarefa é executada quando for escalonada e o tempo configurado foi atingido;
- Não-preemptiva por EVENTO: A tarefa é executada quando for escalonada e o evento vinculado a ela estiver em nível alto, ou seja, TRUE.

Tarefa (Task) preemptiva

É recomendado para sistemas que devem apresentar comportamento determinístico no tempo. Neste sistema quando o intervalo de uma tarefa de maior prioridade vence, a tarefa em execução sofre preempção (é suspensa) e a nova tarefa de prioridade maior passa a executar imediatamente. Quando a tarefa de maior prioridade termina, a tarefa suspensa anteriormente volta a executar do ponto onde parou.

- Preemptiva por TEMPO: Executa-se a tarefa no instante em que seu tempo configurado for atingido;
- Preemptiva por EVENTO: Executa-se a tarefa no instante da transição de FALSE para TRUE do evento (variável) vinculado a ela.



-  Tarefa A, Prioridade 0, Período 100 ms
-  Tarefa B, Prioridade 1, Período 200 ms
-  Tarefa C, Prioridade 2, Período 300 ms

Tipos de dados

A norma IEC61131-3 trabalha com diversos tipos de dados no CLP. Cada um possui características específicas, listadas abaixo:

TIPO DE DADO	TAMANHO (em memória)	INTERVALO
BOOL	1 bit	TRUE e FALSE
INT	16 bits	-32768 a +32767
UINT	16 bits	0 a 65535
WORD	16 bits	0 a FFFF
DINT	32 bits	-2147483648 a +2147483647
UDINT	32 bits	0 a 4294967295
DWORD	32 bits	0 a FFFFFFFF
REAL	32 bits	-3.40282346638528860e+38 a 3.40282346638528860e+38 Underflow: 1.1754943508222875e-38
DATE	32 bits	01/01/1970 a 19/01/2038
TIME	32 bits	0 a 49d17h2m47s290ms
TIME_OF_DAY	32 bits	00:00:00 a 23:59:59
DATE_AND_TIME	32 bits	01/01/1970 a 19/01/2038 00:00:00 a 23:59:59
STRING	-----	Caracteres ASCII
ARRAY	-----	[0...1999]
STRUCT	-----	-----

Declaração de constantes

É possível utilizar constantes diretamente nos Programas (*Programs*), Funções (*Functions*) e Blocos funcionais (*Functions Block*) do programa. Para isto, deve-se observar a nomenclatura IEC de declaração de constantes, mostrada abaixo.

A sintaxe de declaração de uma constante é composta de 2 partes: **Tipo de dado** e **Valor**.

Ao declarar uma variável, deve-se especificar seu tipo de dado (através da janela de variáveis globais). Da mesma maneira, ao declarar uma constante no programa, deve-se especificar o tipo de dado a ser tratado. A tabela abaixo mostra as sintaxes de declaração de constantes para todos os tipos de dados da norma IEC61131-3.

Tipo de Dado	Linguagem	Sintaxe
BOOL	Ladder	BOOL#0 bool#0 BOOL#1 bool#1 BOOL#FALSE bool#FALSE BOOL#TRUE bool#TRUE
INT	Ladder	INT# <i>número</i> int# <i>número</i>
UINT	Ladder	UINT# <i>número</i> uint# <i>número</i>
DINT	Ladder	DINT# <i>número</i> dint# <i>número</i>
UDINT	Ladder	UDINT# <i>número</i> udint# <i>número</i>
WORD	Ladder	WORD# <i>número</i> word# <i>número</i>
DWORD	Ladder	DWORD# <i>número</i> dword# <i>número</i>
REAL	Ladder	REAL# <i>número</i> real# <i>número</i> Utilizar somente ponto (.), tipo REAL não aceita vírgula (,).

Tipo de Dado	Linguagem	Sintaxe
TIME	Ladder	TIME#49d17h2m47s290ms TIME#49d_17h_2m_47s_290ms time#49d17h2m47s290ms time#49d_17h_2m_47s_290ms
		T#49d17h2m47s290ms T#49d_17h_2m_47s_290ms t#49d17h2m47s290ms t#49d_17h_2m_47s_290ms
DATE	Ladder	DATE#yyyy-mm-dd date#yyyy-mm-dd
		D#yyyy-mm-dd d#yyyy-mm-dd
TOD (TIME_OF_DAY)	Ladder	TIME_OF_DAY#hh:mm:ss time_of_day#hh:mm:ss
		TOD#hh:mm:ss tod#hh:mm:ss
DT (DATE_AND_TIME)	Ladder	DATE_AND_TIME#yyy/mm/dd hh:mm:ss date_and_time#yyy/mm/dd hh:mm:ss
		DT#yyy/mm/dd/ hh:mm:ss dt#yyy/mm/dd hh:mm:ss
STRING	Ladder	STRING#'texto'

Ao declarar uma constante, é possível também, definir a base numérica em que ela está sendo inserida. As bases suportadas pela norma são: 2 (binária), 8 (octal), 10 (decimal) e 16 (hexadecimal), sendo a base 10 o padrão utilizado ao inserir constantes no programa.

A sintaxe para definir a base numérica de uma constante (quando não decimal) é mostrada na próxima tabela. Esta tabela também lista todos os tipos de dados que suportam outras bases numéricas.

Tipo de Dado	Linguagem	BASE	Sintaxe
INT	Ladder	2 (bin)	INT#2# <i>número</i>
		8 (oct)	INT#8# <i>número</i>
		16 (hexa)	INT#16# <i>número</i>
UINT	Ladder	2 (bin)	UINT#2# <i>número</i>
		8 (oct)	UINT#8# <i>número</i>
		16 (hexa)	UINT#16# <i>número</i>
DINT	Ladder	2 (bin)	DINT#2# <i>número</i>
		8 (oct)	DINT#8# <i>número</i>
		16 (hexa)	DINT#16# <i>número</i>
UDINT	Ladder	2 (bin)	UDINT#2# <i>número</i>
		8 (oct)	UDINT#8# <i>número</i>
		16 (hexa)	UDINT#16# <i>número</i>
WORD	Ladder	2 (bin)	WORD#2# <i>número</i>
		8 (oct)	WORD#8# <i>número</i>
		16 (hexa)	WORD#16# <i>número</i>
DWORD	Ladder	2 (bin)	DWORD#2# <i>número</i>
		8 (oct)	DWORD#8# <i>número</i>
		16 (hexa)	DWORD#16# <i>número</i>
REAL	Ladder	2 (bin)	REAL#2# <i>número</i>
		8 (oct)	REAL#8# <i>número</i>
		16 (hexa)	REAL#16# <i>número</i>

CAPÍTULO 2

Software Atos A1 Soft

Sobre o software Atos A1 Soft

Configurações mínimas para rodar o Atos A1 Soft

- **Processador:** Pentium III 500MHz com 256Mb de RAM;
- **Vídeo:** 800x600 pixels (fontes pequenas);
- **Espaço disponível no HD:** 110 Mb;
- **Sistema Operacional:** 2000, XP ou Vista.

Como utilizar os módulos na programação?

Os módulos definidos na configuração de hardware são acessados através do mapeamento de memória global de E/S, disponível na janela de variáveis globais do sistema. Para acessar o mapeamento de memória do hardware configurado no seu projeto, abra a guia "**Variáveis Globais**" no **Gerenciador de Projetos** e clique em **I/O**.



A tabela de **variáveis globais** mostra o mapeamento de cada um dos módulos no projeto.

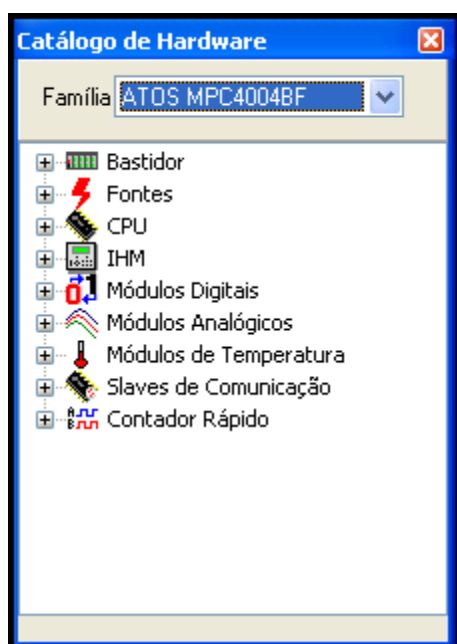
Pela guia "**Placas de Expansão**" escolha o módulo que deseja inspecionar. Ao fazer isso o mapeamento do módulo escolhido é mostrado logo abaixo em formato de uma tabela.

Variáveis Globais			
Grupo		Placas de expansão	
I/O		Slot A1 / 4004.05BF / - CPU 8E "N" / 8S "N" [24Vcc]	
Endereço		Slot A1 / 4004.05BF / - CPU 8E "N" / 8S "N" [24Vcc]	
		Slot A2 / 4004.54G / - Módulo digital 16E "P" / 16S "P" [24Vcc]	
		Slot A3 / 4004.61/A / - Módulo analógico 4E [V - 0 a +10Vcc / A - 0 a +20mA] 4S [A - 0 a +20mA]	
%I0.0		BOOL	DIGITAL INPUT 0
%I0.1		BOOL	DIGITAL INPUT 1
%I0.2		BOOL	DIGITAL INPUT 2
%I0.3		BOOL	DIGITAL INPUT 3
%I0.4		BOOL	DIGITAL INPUT 4
%I0.5		BOOL	DIGITAL INPUT 5
%I0.6		BOOL	DIGITAL INPUT 6
%I0.7		BOOL	DIGITAL INPUT 7
%Q0.0		BOOL	DIGITAL OUTPUT 0
%Q0.1		BOOL	DIGITAL OUTPUT 1
%Q0.2		BOOL	DIGITAL OUTPUT 2
%Q0.3		BOOL	DIGITAL OUTPUT 3
%Q0.4		BOOL	DIGITAL OUTPUT 4
%Q0.5		BOOL	DIGITAL OUTPUT 5
%Q0.6		BOOL	DIGITAL OUTPUT 6
%Q0.7		BOOL	DIGITAL OUTPUT 7

Nesta tabela é possível definir nomes para as variáveis de E/S para utilizá-las no programa *ladder*.

Configurando hardware

Atos MPC4004BF



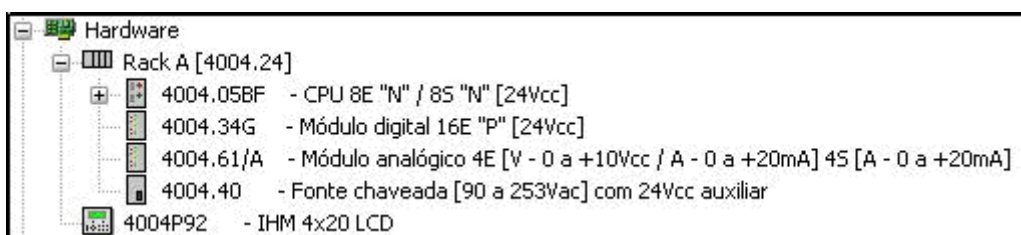
Para criar uma configuração de hardware no Atos A1 Soft, clique com o botão direito do mouse sobre o item **Hardware** no **Gerenciador de Projetos**. No menu pop-up que abrir, clique sobre a opção "**Ver catálogo de hardware**" para abrir uma janela com a lista dos módulos disponíveis.

Os módulos são separados por tipo (digitais, analógicas, etc.).

A configuração mínima para um projeto consiste em um bastidor de 2 slots com fonte de alimentação e CPU. Para inserir um bastidor, clique e arraste o módulo para o ícone de configuração de hardware.

Dentro do bastidor existem as posições das placas (slots), que devem ser preenchidos com os módulos desejados no projeto (clitando sobre o módulo e arrastando na posição desejada no bastidor).

Exemplo de configuração de hardware:



Jumpers de endereçamento

O endereçamento na família Atos MPC4004 é normalmente feito através de um jumper que varia de A até H.

Algumas unidades digitais e analógicas da família Atos MPC4004 receberam um jumper adicional, denominado STG ou **Jumper de Grupo**. Este jumper permite que haja um aumento da quantidade de pontos digitais e analógicos da família Atos MPC4004.

Cada tipo de placa (digital ou analógica) possui seu jumper de grupo em uma posição default indicado abaixo.

Posição default do jumper STG:

- **STG em A** = módulos digitais;
- **STG em B** = módulos analógicos.

Módulos que possuem **Jumper de Grupo**:

DIGITAIS	
4004.31G	Módulo de expansão com 16S "N"
4004.31H	Módulo de expansão com 16S "N" (Troca a Quente)
4004.32G	Módulo de expansão com 16S "P"
4004.32H	Módulo de expansão com 16S "P" (Troca a Quente)
4004.33G	Módulo de expansão com 16E "N"
4004.33H	Módulo de expansão com 16E "P/N" (Troca a Quente)
4004.34G	Módulo de expansão com 16E "P"
4004.38G	Módulo de expansão com 8E "N" ou "P"
4004.53G	Módulo de expansão com 16E/16S "N"
4004.53H	Módulo de expansão com 16E "P" ou "N" /16S "N" (Troca a Quente)
4004.54G	Módulo de expansão com 16E /16S "P"
4004.54H	Módulo de expansão com 16E "P" ou "N" /16S "P" (Troca a Quente)
4004.55G	Módulo de expansão com 32 entradas "N"
4004.55H	Módulo de expansão com 32 entradas "P" ou "N" (Troca a Quente)
4004.56G	Módulo de expansão com 32 entradas "P"
4004.58G	Módulo de expansão com 16E "P" ou "N" /16S relé
ANALÓGICAS	
4004.62G	Módulo de expansão 8E Analógicas (tensão ou corrente)
4004.62/P	Módulo de expansão 4E Analógicas (tensão ou corrente)
4004.63G	Módulo de expansão 8S Analógicas (tensão)
4004.63/P	Módulo de expansão 4S Analógica (tensão)
4004.64G	Módulo de expansão 8S Analógica (corrente)
4004.64/P	Módulo de expansão 4S Analógica (corrente)

Resumo geral das quantidades de uma mesma placa no bastidor:

TIPO DE E/S	QUANT. CS	JUMPERS
08 Entradas / 08 Saídas	08	ST1: A-H
16 Entradas / 16 Saídas	15	ST1: A-H STG: A e B
16 Entradas	15	ST1: A-H STG: A e B
16 Saídas	15	ST1: A-H STG: A e B
32 Entradas	15	ST1: A-H STG: A e B
Temperatura	08	ST2: A-H
E/S Analógica	08	ST1: A-H
Módulos MAC	15	ST1: A-H STG: A e B
Contador Rápido	01	ST2: G e H
Multiplex	01	ST2: A-H
Específico	08	ST2: A-H

O Jumper de Grupo permite que uma determinada placa possa ser configurada para trabalhar no grupo A ou no grupo B. Cada grupo possui 8 endereçamentos, desta forma o módulo que possuir o Jumper de Grupo pode ser utilizado em 16 slots.

Como a CPU possui uma expansão digital, o número máximo será de 15 módulos a serem utilizados.

Regras para a inclusão de unidades no(s) bastidor(es)

- Os módulos 4004.60 e 4004.61 só podem ser colocadas num total de 08;
- Os módulos digitais com 8 pontos, como por exemplo, 4004.37, 4004.51, 4004.39, etc., só podem ser inseridos no total de 07 módulos;
- O Contador Rápido e o Multiplex, só podem ser alocados uma única vez;



O contador rápido utiliza 02 CS's fixos: G e H (GRUPO B). Caso os CS do contador já tenham sido utilizados por outro módulo, o Atos A1 Soft informa o usuário. A placa não possui nenhum jumper de endereçamento.

Atos MPC6006L



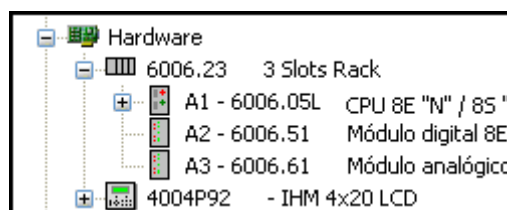
Para criar uma configuração de hardware no Atos A1 Soft, clique com o botão direito do mouse sobre o item **Hardware** no **Gerenciador de Projetos**. No menu pop-up que abrir, clique sobre a opção "**Ver catálogo de hardware**" para abrir uma janela com a lista dos módulos disponíveis.

Os módulos são separados por tipo (digitais, analógicas, etc.).

A configuração mínima para um projeto consiste em um bastidor de 1 slots com a CPU. Para inserir um bastidor, clique e arraste o módulo para o ícone de configuração de hardware.

Dentro do bastidor existem as posições das placas (slots), que devem ser preenchidos com os módulos desejados no projeto (clitando sobre o módulo e arrastando na posição desejada no bastidor)

Exemplo de configuração de hardware:

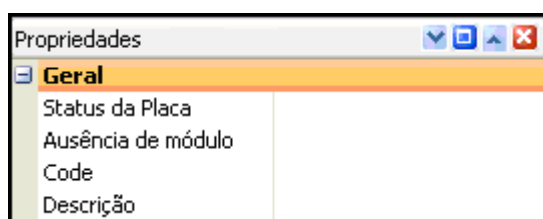


Propriedades do Hardware

Os módulos da família Atos MPC6006 possuem algumas características que podem ser configuradas através do software Atos A1 Soft.

Verificação do módulo

O usuário pode optar por configurar uma variável Booleana para informar quando o módulo está ausente e um registro para verificar o Status dessa placa. Esta opção é configurada na guia Propriedades do módulo selecionado, como mostrado na imagem a seguir:



Descrição dos campos:

- **Status da Placa:** Variável do tipo UINT que indica o status da placa, sendo que cada valor indica um status da placa:
 - Status da Placa = 0
 - Status da Placa = 1
- **Ausência de módulo:** Variável booleana que é acionada (TRUE) quando o módulo configurado estiver ausente;
- **Code:** Código do modelo da placa configurada;
- **Descrição:** Informações sobre a placa selecionada, como quantidade e tipo das entradas / saídas, tensão de trabalho.

Uma vez definidas essas variáveis, as mesmas poderão ser utilizadas no ladder (programação das telas de alarme da IHM, por exemplo) para sinalizar o evento de troca a quente durante o processo.

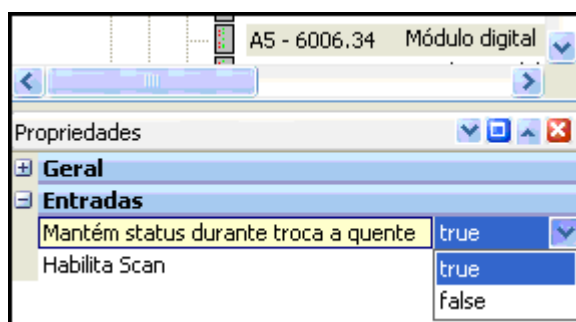
Módulos com troca quente

Todos os módulos da família Atos MPC6006 possuem a característica de troca quente.

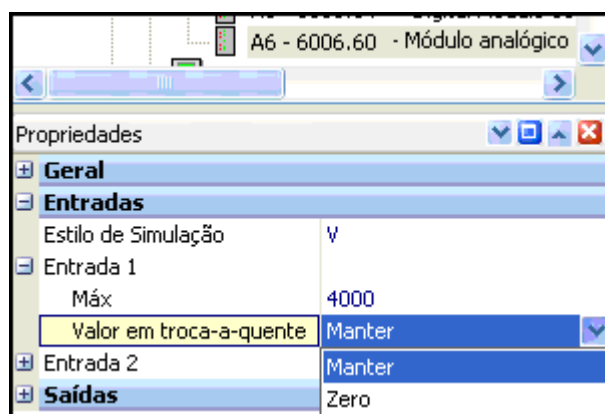
Durante a troca quente as CPU pára de atualizar as saídas, mas para as entradas o comportamento da CPU depende das configurações realizadas.

- Para as entradas digitais:
 - Desligar as entradas em troca a quente;
 - Manter o status das entradas em troca a quente;
- Para as entradas analógicas e de temperatura:
 - Zerar o valor das entradas em troca a quente;
 - Manter o valor das entradas em troca a quente;

A seleção destas opções é feita na guia Propriedades do módulo selecionado, como mostrado nas imagens abaixo:



Configuração para módulos digitais



Configuração para módulos analógicos



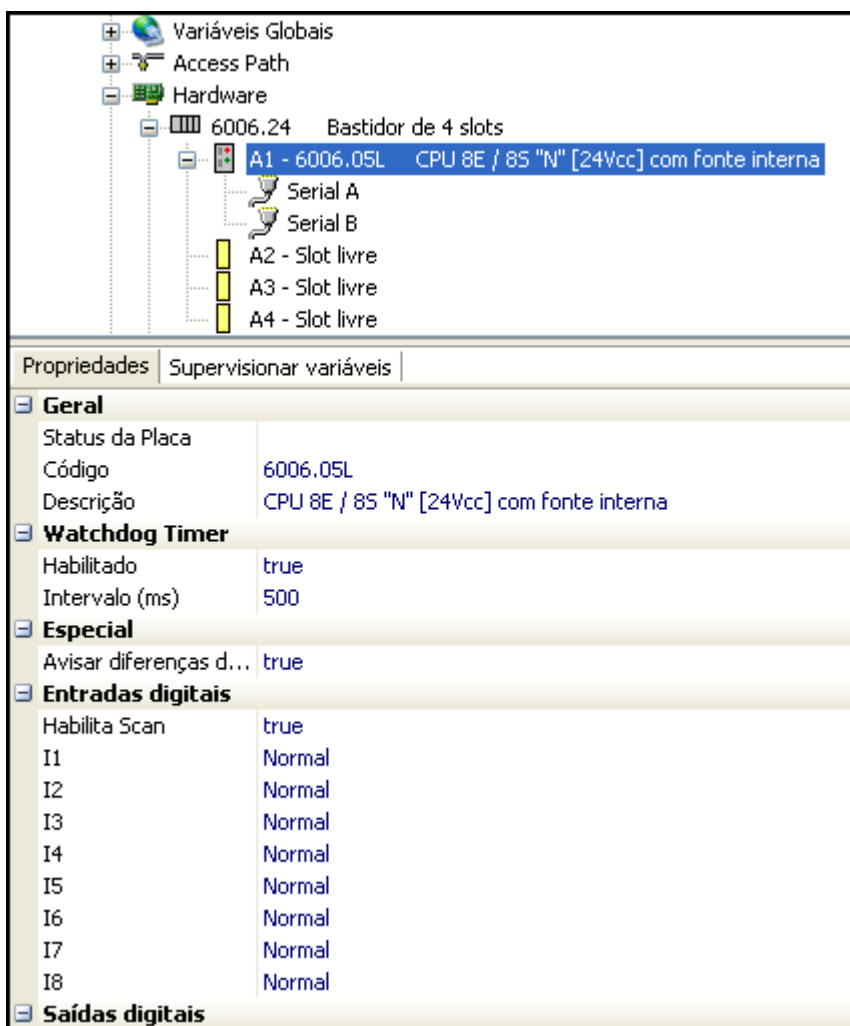
A configuração das entradas de temperatura é feita através do campo “Valor em troca-quente” como nos módulos analógicos.

Configurando entradas da CPU

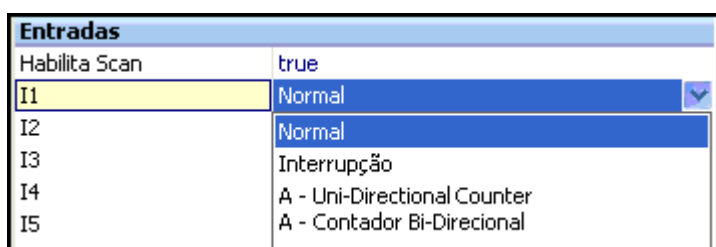
Para o controlador programável Atos MPC6006, as entradas do hardware são configuradas conforme sua aplicação.

Para realizar essa configuração, o seguinte procedimento deve ser seguido:

- Selecione a CPU correspondente, no rack configurado, na guia Propriedades será possível verificar as configurações atuais, conforme mostra a figura a seguir:



Na guia Propriedades, selecionado a entrada, é possível escolher o tipo que será utilizado na aplicação, conforme mostrado na figura a seguir:



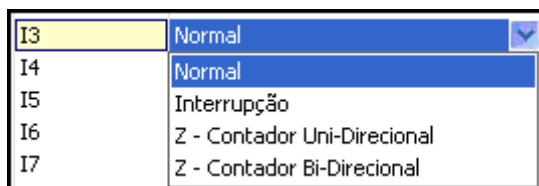
Descrição dos campos:

- **Normal:** Entradas digitais normal, associadas à entrada I da CPU.
- **Interrupção:** Deve ser selecionada esta opção quando utilizada a entrada como variável de habilitação de tarefa preemptiva por evento. (Veja mais em Tarefas preemptivas por evento).
- **Contador Uni-Direcional:** Configura entrada como contador uni-direcional.
- **Contador Bi-Direcional:** Configura a entrada como contador bi-direcional.

Quando configurado como contador bi-direcional o software automaticamente configura as entradas complementares.

Por exemplo: Se configurada a entrada I1 como bi-direcional, automaticamente o software configura a entrada I2 como B- Contador BI-Direcional.

As entradas I3 e I6 podem ser configuradas como o zero do contador, conforme mostra a figura a seguir.



A entrada I3 corresponde ao zero do contador 1 e a entrada I6 ao zero do contador 2.

Configurando saídas da CPU

Para o controlador programável Atos MPC6006, assim como as entradas, saídas do hardware são configuradas conforme sua aplicação.

Para realizar essa configuração, o seguinte procedimento deve ser seguido:

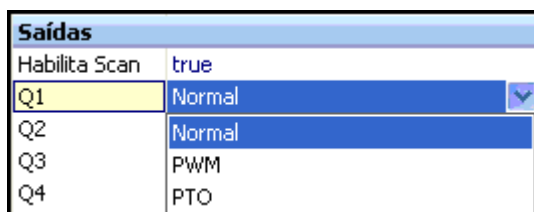
Selecione a CPU correspondente, no rack configurado, na guia Propriedades será possível verificar as configurações atuais, conforme mostra a figura abaixo:



Na guia Propriedades, selecionado a saída, é possível escolher o tipo que será utilizado na aplicação. Apenas as saídas **Q1**, **Q2**, **Q7** e **Q8** podem ser configuradas.

Configurando as saídas Q1 e Q2

Quando utilizado os blocos PTO ou PWM, as saídas devem ser configuradas, caso essa configuração não seja realizada uma mensagem de erro será visualiza ao compilar o projeto. A imagem abaixo mostra a configuração disponível para essas duas saídas.

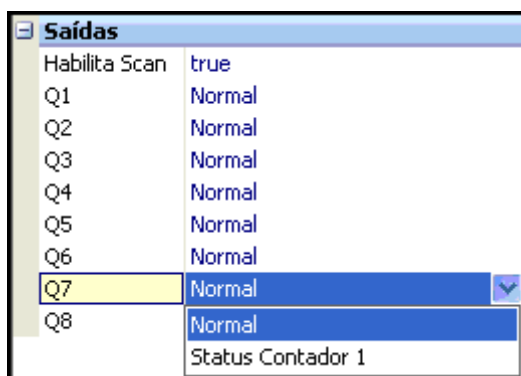


Descrição dos campos:

- **Normal:** Saída digital normal, associada a saída Q da CPU.
- **PWM:** Deve ser configurada essa opção quando utilizado uma instrução PWM configurada com a essa saída.
- **PTO:** Seleciona-se essa opção quando utilizado uma instrução PTO utilizando essa saída no bloco.

Configurando as saídas Q7 e Q8

Quando utilizado o bloco HS COUNTER as saídas devem ser configuradas como status do contador, caso essa configuração não seja realizada uma mensagem de erro será visualiza ao compilar o projeto. A imagem abaixo mostra a configuração disponível para essas duas saídas.



Descrição dos campos:

- **Normal:** Saída digital normal, associada a saída Q da CPU.
- **Status Contador:** Deve ser configurada dessa forma, quando utilizado a instrução HS_COUNTER, sendo que a saída Q7 é referente ao Contador 1 e a saída Q8 ao Contador 2.

Verificação de Hardware

A verificação de hardware é uma característica do Atos MPC6006L que compara a configuração de hardware física no bastidor com a configuração feita no programa de usuário.

Caso existam diferenças nessas configurações até três formas de aviso podem ser visualizadas, conforme mostrado abaixo:

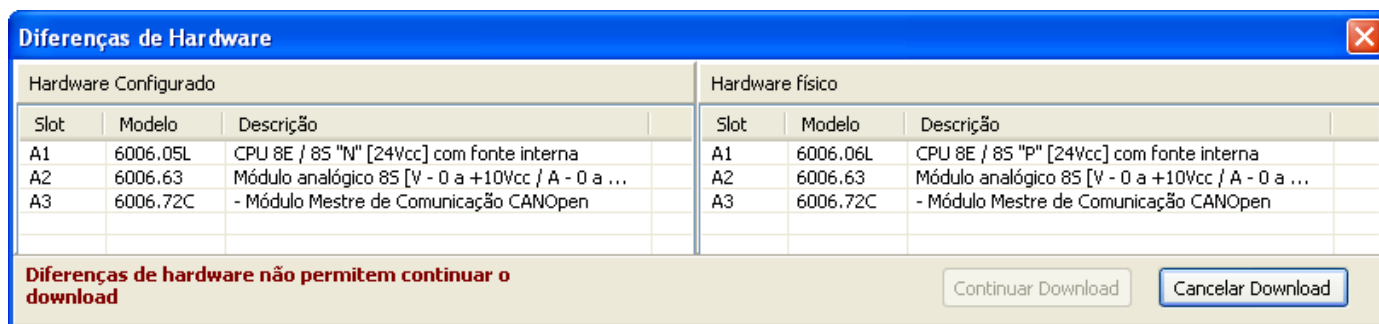
- **Hardware**

Caso haja essa diferença, o LED ST2, localizado no frontal da CPU da família Atos MPC6006 pisca em uma cadência de 1s.

- **Software**

Através do software

Atos A1 Soft o aviso dessa diferença é mostrado ao iniciar o envio do program de usuário. A mensagem abaixo é mostrada informando qual a diferença encontrada, só será permitido o envio do programa caso seja selecionado a opção "Continuar Download" e não haja diferença de CPU.

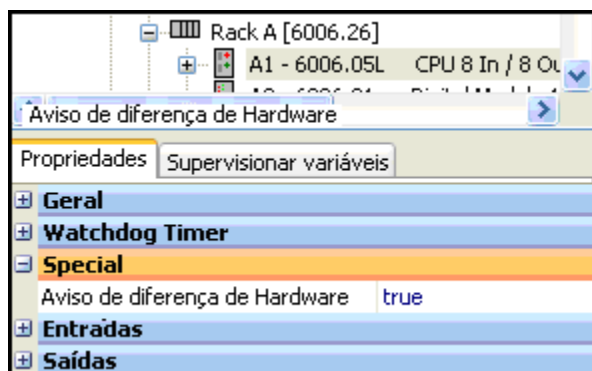


O aviso através do Hardware e do Software não pode ser desabilitado, com isso sempre que houver diferenças de configurações de Hardware o LED deve piscar e a mensagem acima ser visualizada.

- **IHM**

Na IHM o aviso de diferença de configuração de Hardware é verificado sempre na inicialização da CPU, essa opção é configurada pelo usuário. Para realizar essa configuração, o seguinte procedimento deve ser seguido:

- Selecione a CPU configurada, no item Hardware do Gerenciado de Projeto;
- Na guia Propriedades, a configuração é feita através do item Especial;



Se desabilitada (FALSE) após a inicialização do CLP, caso haja diferença de hardware, não será mais visualizada a mensagem mostrada a seguir.



ERRO DE CONFIGURACAO
DE HARDWARE

Se desabilitada (FALSE) após a inicialização do CLP, caso haja diferença de hardware, não será mais visualizada a mensagem: ERRO DE CONFIGURAÇÃO DE HARDWARE.



Essa opção não desabilita o LED de diagnóstico, que indica a falha na configuração de Hardware.

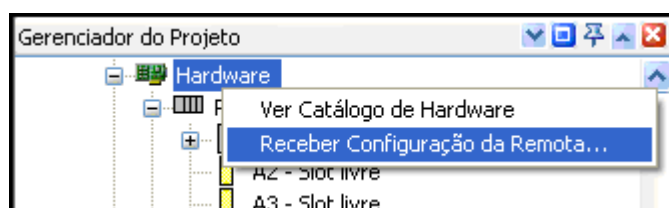
Mais informação sobre *LEDs de diagnóstico* no Manual de Hardware.

Identificação Automática dos módulos

O recurso de Identificação Automática dos módulos disponível no software Atos A1 Soft permite que o usuário primeiro configure fisicamente o Hardware e logo em seguida busque essa configuração através do software, mantendo dessa forma a compatibilidade do hardware configurado no projeto com o configurado fisicamente.

Para buscar essas informações, o procedimento abaixo deve ser seguido.

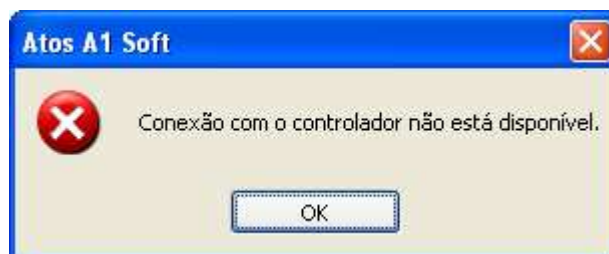
- Na guia Gerenciador do Projeto, clique com o botão direito do mouse sobre o item Hardware;
- No menu pop up visualizado, selecione a opção Receber Configuração da Remota..., conforme mostrado na figura a seguir.



- Após esse procedimento será possível visualizar a janela Hardware Remoto, mostrada abaixo, nela deverá conter as informações da configuração encontrada pelo Software Atos A1 Soft;



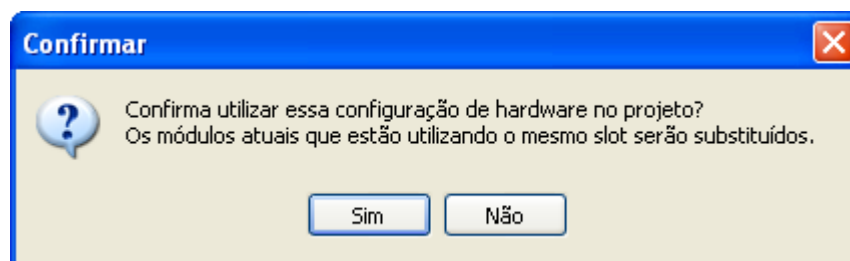
Caso o CLP não esteja conectado, ou haja falhas na comunicação, a mensagem abaixo será mostrada.



- Selecionando o botão  uma mensagem de alerta será mostrada ao usuário, conforme mostra a figura a seguir.



Caso existam módulos configurados anteriormente no projeto, depois de confirmada a mensagem abaixo, todos serão apagados e substituídos pelo Hardware identificado.



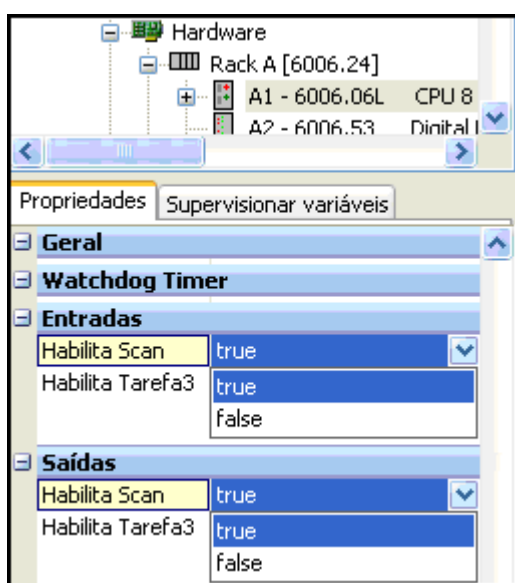
Atualizando entradas e saídas

A leitura das entradas e atualização das saídas pode ser definida quando será realizada, através do SCAN (na varredura) ou na execução de Tarefas preemptivas.



Não é necessário ter a mesma configuração para a entrada e para saída.

A configuração dessa atualização é feita na guia Propriedades, conforme mostra a figura abaixo:



Essa mesma configuração pode ser feita para entradas e saídas de módulos digitais e analógicos.

Com isso, a atualização de entradas e saídas irá depender se o campo Habilita é TRUE ou FALSE.

Por exemplo:

- Se Habilita Scan=True, as entradas são lidas e saídas atualizadas a cada varredura.
- Se Habilita Tarefa=True e Habilita Scan=False, a leitura das entradas e atualização das saídas será feita quando a tarefa for executada.



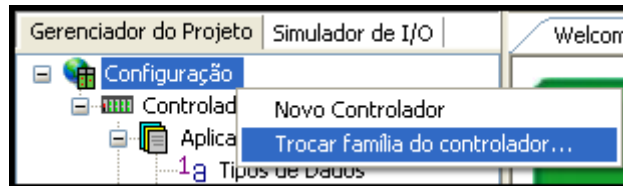
Só serão visualizadas as para essa configuração as tarefas preemptivas. Conversão de controladores

Trocar de controlador

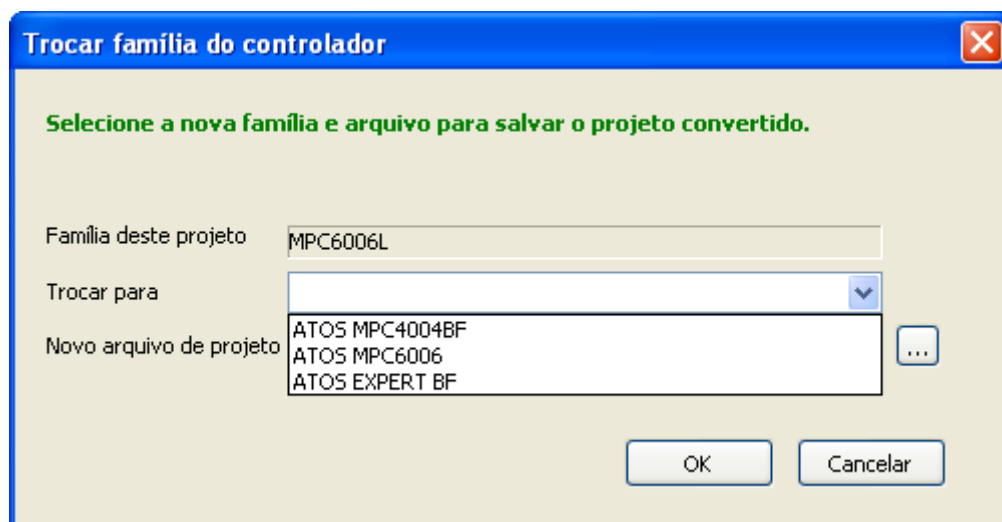
Para utilizar o mesmo programa de usuário nos diferentes controladores programáveis contemplados pelo software Atos A1 Soft é necessário fazer a conversão dos controladores.

Para converter o controlador programável deve-se seguir o procedimento abaixo:

- No Gerenciador do Projeto, com o botão direito do mouse clique no item Configuração;
- No menu pop-up visualizado selecione a opção Trocar driver do controlador, conforme mostrado na figura abaixo:



Logo em seguida será visualizada a janela Trocar driver do controlador, conforme mostrado na imagem abaixo.



Descrição dos campos:

- **Driver deste projeto:** Esse campo mostra qual o controlador atual utilizado no projeto;
- **Trocar para:** Campo destinado para o usuário selecionar para qual controlador o programa de usuário será convertido.
- **Novo arquivo de projeto:** Caminho do novo projeto.



A conversão do Hardware configurado irá depender do controlador de destino, se o item do Hardware não existir no controlador de destino será retirado da configuração. Clique aqui para verificar a [Tabela de Equivalência de Hardware](#).

Tabela de equivalência de módulos

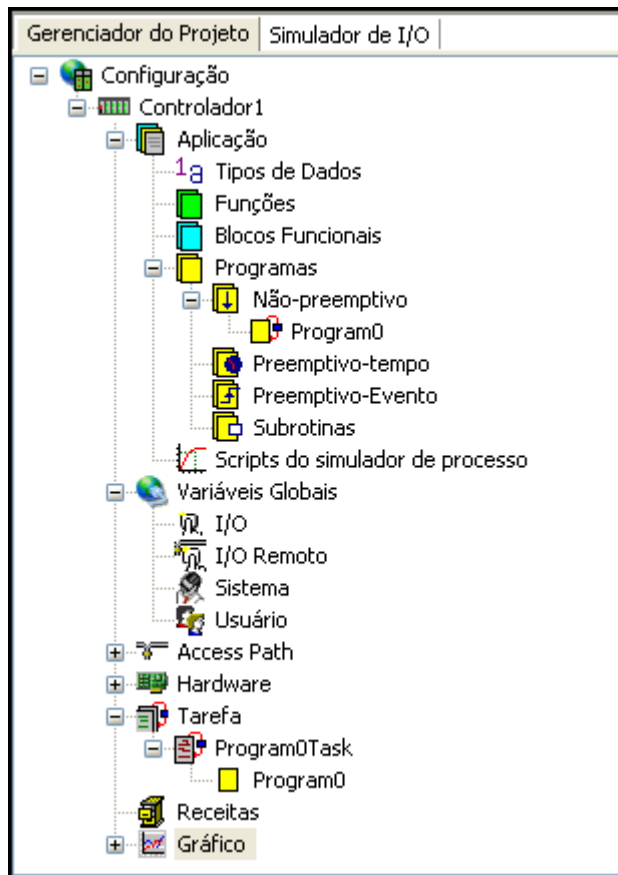
Controladores programáveis disponíveis para conversão				
Descrição	Atos MPC4004BF	Atos MPC6006	Atos MPC6006L	Atos Expert BF
CPU 24Vcc 8 E / 8 S Tipo NPN	4004.05BF	6006.05	6006.05L	2850.00
CPU 24Vcc 8 E / 8 S Tipo PNP	4004.06BF	6006.06	6006.06L	2850.10
CPU, IHM 14E / 10S Tipo NPN	4004.05BF	6006.05	6006.05L	2850.00
CPU, IHM 14E / 10S Tipo PNP	4004.06BF	6006.06	6006.06L	2850.10
Bastidor de 01 passo	Não tem * (4004.22)	Não tem*(6006.22)	6006.21	Não tem
Bastidor de 02 passos	4004.22	6006.22	6006.22	Não tem
Bastidor de 03 passos	Não tem	6006.23	6006.23	Não tem
Bastidor de 04 passos	4004.24	6006.24	6006.24	Não tem
Bastidor de 06 passos	4004.26	6006.26	6006.26	Não tem
Bastidor de 06 passos c/ expansão	4004.26R	6006.26R	Não tem	Não tem
Bastidor de 08 passos	4004.28	6006.28	Não tem	Não tem
Bastidor de 08 passos c/ expansão	4004.28R	6006.28R	Não tem	Não tem
Bastidor de 10 passos	4004.2A	6006.2A	Não tem	Não tem
Bastidor de 10 passos c/ expansão	4004.2AR	6006.2AR	Não tem	Não tem
Bastidor de 12 passos	4004.2C	6006.2C	Não tem	Não tem
Bastidor de 12 passos c/ expansão	4004.2CR	6006.2CR	Não tem	Não tem
Expansão 16S tipo NPN	4004.31 /G/H	6006.31	6006.31	Não tem
Expansão 16S tipo PNP	4004.32 /G/H	6006.32	6006.32	Não tem
Expansão 16E tipo NPN	4004.33 /G/H	6006.33	6006.33	Não tem
Expansão 16E tipo PNP	4004.34 /G	6006.34	6006.34	Não tem
Expansão 8S tipo NPN	Não tem	6006.41	6006.41	Não tem
Expansão 8S tipo	Não tem	6006.42	6006.42	Não tem

PNP				
Expansão 8S [Relé]	4004.37	Não tem	Não tem	Não tem
Expansão 8S [TRIAC]	4004.39	Não tem	Não tem	Não tem
Expansão 8E [110 Vac]	4004.35	Não tem	Não tem	Não tem
Expansão 8E [220 Vac]	4004.35/A	Não tem	Não tem	Não tem
Expansão 8E tipo NPN	4004.38G	6006.43	6006.43	Não tem
Expansão 8E tipo PNP	4004.38G	6006.44	6006.44	Não tem
Expansão 8E/8S tipo NPN	4004.51	6006.51	6006.51	Não tem
Expansão 8E/8S tipo PNP	4004.52	6006.52	6006.52	Não tem
Expansão 8E/8S tipo PNP/NPN	4004.57	Não tem	Não tem	Não tem
Expansão 16E/16S tipo NPN	4004.53 /G/H	6006.53	6006.53	Não tem
Expansão 16E/16S tipo PNP	4004.54 /G/H	6006.54	6006.54	Não tem
Expansão 16E/16S tipo PNP/NPN	4004.58 /G	Não tem	Não tem	Não tem
Expansão 32E NPN	4004.55 /G/H	6006.55	6006.55	Não tem
Expansão 32E PNP	4004.56 /G	6006.56	6006.56	Não tem
Expansão 2E/2S analógica	4004.60 /A	6006.60	6006.60	Não tem
Expansão 2E/2S analógica [bipolar]	4004.60N	Não tem	Não tem	Não tem
Expansão 4E/4S analógica	4004.61 /A	6006.61	6006.61	Não tem
Expansão 4E/4S analógica [bipolar]	4004.61N	Não tem	Não tem	Não tem
Expansão 8E analógica	4004.62 /G	6006.62	6006.62	Não tem
Expansão 4E analógica	4004.62M/P	6006.62/M	6006.62/M	Não tem
Expansão 8S analógica	4004.63 /G [V]	6006.63 [V / A]	6006.63 [V / A]	Não tem
Expansão 8S analógica	4004.64 /G [V]	Não tem ->* 6006.63	Não tem -> *6006.63	Não tem
Expansão 4S analógica	4004.63 /M/P	6006.63/M	6006.63/M	Não tem
Expansão 4S analógica	4004.64/M/P	Não tem-> *6006.63/M	Não tem -> 6006.63/M	Não tem
Expansão 4E termopar	4004.65J /K	6006.65	6006.65	Não tem

Expansão 8E termopar	4004.66J /K	6006.66	6006.66	Não tem
Multiplex Botões e LEDs	4004.70	6006.70	6006.70	Não tem
Expansão 4E PT100	4004.75/P /P1/P2	6006.75	6006.75	Não tem
Expansão 8E PT100	4004.76/P /P1/P2	6006.76	6006.76	Não tem
Expansão 4E PT100 4E 0-10V	4004.85 / P2	6006.85	6006.85	Não tem
Fonte chaveada [90 a 253Vac]	4004.40 /F/R	6006.40	Não tem	Não tem
Fonte chaveada [9 a 36Vcc]	4004.40/A	Não tem	Não tem	Não tem
Fonte chaveada [36 a 60Vcc]	4004.40/D	Não tem	Não tem	Não tem
Fonte chaveada [18 a 60Vcc]	4004.40/G	Não tem	Não tem	Não tem
Slave de comunicação APR [Mestre/Escravo]	4004.72R	Não tem	Não tem	Não tem
Slave de com. Modbus RTU [Mestre/Escravo]	4004.72M	Não tem	Não tem	Não tem
Slave de comunicação genérica	4004.72G	Não tem	Não tem	Não tem
Módulo de contagem rápida	4004.87	Não tem	Não tem	Não tem
Módulo de contagem rápida + 2S analog.	4004.87SA	Não tem	Não tem	Não tem

[*] Módulo que será verificado após a conversão.

Gerenciador do Projeto



O gerenciador de projeto mostra através de uma árvore hierárquica de opções, todas as informações relativas ao projeto.

- **Controlador 1:** Ao criar um projeto, ele vem automaticamente com um controlador definido. Num mesmo projeto, mais de um controlador pode ser gerenciado. Para isto, clique em "Configuração" e adicione os outros controladores a serem utilizados no projeto;
- **Tipos de Dados:** Mostra estruturas (structs) definidas pelo usuário e os padrões Atos para utilização no programa;
- **Funções:** Lista todas as *Funções* criadas pelo usuário;
- **Blocos Funcionais:** Lista todos os *Blocos Funcionais* criados pelo usuário;
- **Programas:** Mostra todos os Programas do projeto, separados entre preemptivos e não-preemptivos;
- **Variáveis Globais:** Variáveis globais do controlador. Este item mostra separadamente todo o mapeamento de memória global do CLP, separado em três partes: Sistema, I/O (hardware) e Usuário;
- **Access Path:** Configuração de comunicação em rede com outros dispositivos;
- **Hardware:** Configura o hardware do CLP. Para mais informações verifique o tópico Configuração de hardware deste Help.
- **Tarefa:** Mostra todas as Tarefas configuradas no projeto, com seus respectivos Programas associados em ordem e prioridade de execução no controlador.
- **Receitas:** Arquivo de receitas realiza o armazenamento ou a recuperação de até 16 frames de dados. Configuração dos frames tanto para utilizar receitas por meio da IHM quanto com bloco funcional Recipe.
- **Gráfico:** Configuração dos gráficos para supervisão.

Atalhos de Teclado

Comandos principais

Ctrl+A	•	Abrir projeto.
Ctrl+N	•	Novo projeto.
Ctrl+S	•	Salva projeto.
Ctrl+P		Imprimir projeto

Editor Ladder

F1	Abre o help da instrução sob o foco
F3	Procurar próxima
F11	Mostrar Grid
Ctrl + C	Copiar
Ctrl + F	Procurar
Ctrl + O	Procurar saída
Ctrl + U	Substituir
Ctrl + V	Colar
Ctrl + X	Recortar
A	Inserir contato aberto
B	Inserir bloco
C	Inserir comentário de linha
F	Inserir contato fechado
G	CALL p/ subrotina
I	Inserir nova linha abaixo da linha corrente
J	Inserir JUMP p/ Label
L	Inserir Label
O	Inserir saída
P	Reabre braço fechado
Q	Abre lista de saídas
S	Abre lista de entradas
U	Inserir nova linha acima da linha corrente
[	Abre braço pararelo
]	Fecha braço pararelo

Projeto

F5	•	Envia o projeto para o CLP
F9	•	Inicia modo de supervisão

Ferramentas

Ctrl+F2	•	Calculadora
---------	---	-------------

Gerenciador de Variáveis

Edição

Para editar um campo no gerenciador de variáveis deve-se seguir os seguintes passos:

- Selecione o campo desejado;
- Habilite a edição, isso pode ser feito de três maneiras:
 - Com mais um clique;
 - Através da tecla F2;
 - Iniciando a digitação no campo

A edição é encerrada através das teclas: ENTER ou TAB.

Navegação

A navegação entre as células do gerenciador de variáveis é feita através das setas direcionais do teclado.



Para navegação nenhum campo da janela deve estar habilitado para edição.

Barra de Menu

Menu Arquivo

- | | |
|---------------|---------------------------------|
| Novo | • Novo projeto. |
| Abrir | • Abrir projeto. |
| Salvar Como | • salva projeto com outro nome. |
| Save | • Salva projeto corrente. |
| Close project | • Fecha projeto corrente. |
| Exit | • Sai do aplicativo. |

Menu Editar

- | | |
|-----------------|--|
| Desfazer | • Desfazer última ação. |
| Refazer | • Refazer última ação. |
| Recortar | • Recortar texto ou linha de programa. |
| Copiar | • Copiar texto ou linha de programa. |
| Colar | • Colar texto ou linha de programa. |
| Apagar | • Apagar texto ou linha de programa. |
| Selecionar Tudo | • Selecionar todo o conteúdo corrente. |
| Localizar | • Procurar por instrução/variável no programa. |
| Substituir | • Substituir variáveis. |

Menu Exibir

- | | |
|------------------------|--------------------------------------|
| Gerenciador do projeto | • Gerenciador do projeto. |
| Mensagens | • Janela de mensagens do sistema. |
| Properties | • Janela de propriedades. |
| Supervisão | • Janela de supervisão de variáveis. |
| Tools | • Lista de ferramentas p/ edição. |
| Zoom + | • Aumenta zoom do ladder. |
| Zoom - | • Diminui zoom do ladder. |

Menu Projeto

- | | |
|--------------------|---|
| Compilar | • Compilar projeto corrente. |
| Simulação | • Iniciar/Parar simulação do projeto. |
| Download | • Envio do projeto para o CLP. |
| Upload | • Leitura do projeto do CLP. |
| Online | • Entrar em modo de supervisão. |
| Programação Online | • Entrar em modo de programação online. |
| Segurança | • Inserir senha de abertura no projeto. |

Menu Ferramentas

- | | |
|-------------|--|
| CLP | • <ul style="list-style-type: none">➢ Configuração da interface de comunicação;➢ Ler informações do CLP;➢ Gravação de firmware;➢ Ler data/hora do CLP. Sincronismo com relógio do PC;➢ Dump de memória;➢ Bin pré-compilado. |
| Calculadora | • Abre a calculadora do Windows. |
| Idioma | • Alterar idioma entre Português e Inglês. |
| Opções | • Abre janela de configuração/preferências do aplicativo e do projeto. |

Menu *Ajuda*

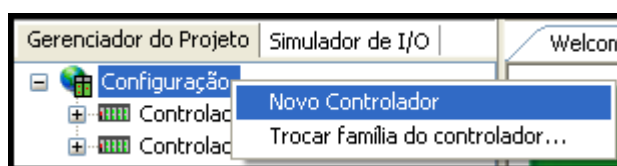
Help do Atos A1 Soft
Help de instruções
Catálogo de Hardware
Sobre

- Help do Atos A1 Soft.
- Conjunto de instruções do Atos A1 Soft.
- Catálogo de Hardware do Atos A1 Soft.
- Dados do aplicativo (versão).

Configurando conexão do controlador

Ao criar um projeto, automaticamente é criado um controlador. Num mesmo projeto, mais de um controlador pode ser configurado.

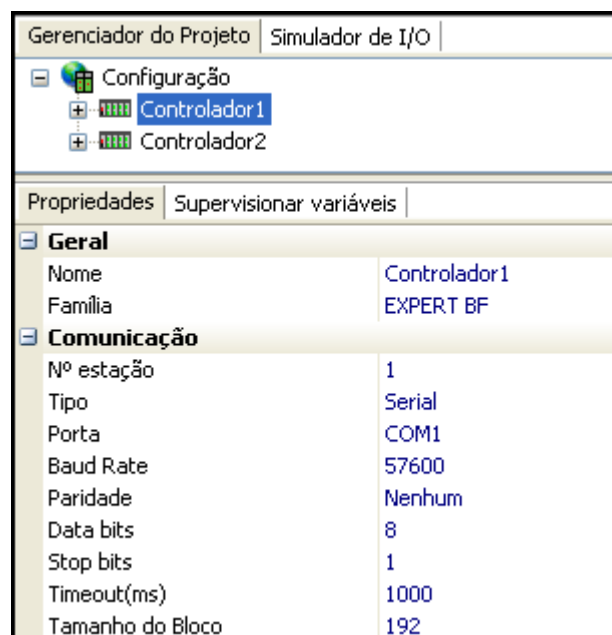
Para adicionar um novo controlador, clique em "*Configuração*" e adicione os outros controladores a serem utilizados no projeto, conforme mostra a figura abaixo:



Cada controlador existente no projeto pode ter uma configuração específica de conexão.

Essa configuração podem ser visualizada na guia "Propriedades" do controlador selecionado, onde os campos de configuração dependem do tipo de comunicação: **Serial** ou **TCP/IP**, conforme mostrado a seguir.

Configurando a comunicação Serial:



Descrição dos campos:

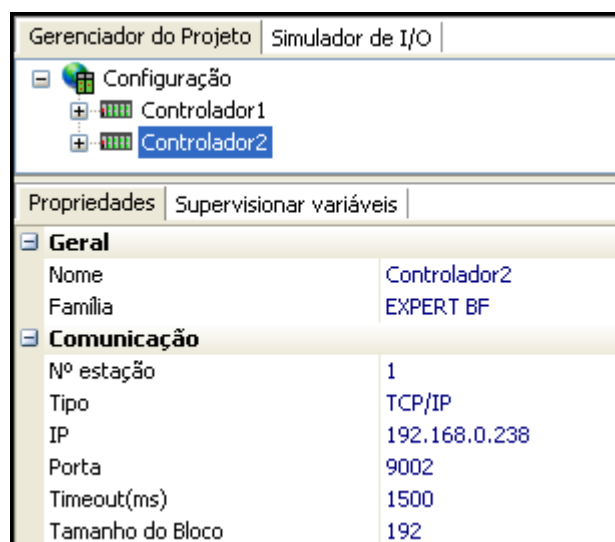
Geral

- **Nome:** Nome definido para o controlador;
- **Família:** Família do controlador;

Comunicação

- **Nº estação:** Número do controlador;
- **Tipo:** Tipo de comunicação - Serial ;
- **Porta:** Porta de comunicação;
- **Baud Rate:** Define a taxa de comunicação;
- **Paridade:** Define paridade - Nenhuma, Par ou Ímpar;
- **Data bits:** Define quantidade de bits de comunicação - 7 ou 8;
- **Stop bits:** Define quantidade de bits de término de comunicação- 1 ou 2;
- **Timeout(ms):** Tempo em milissegundos que o Atos A1 Soft aguarda a resposta do controlador antes de sinalizar algum tipo de falha;
- **Tamanho do Bloco:** Quantidade de bytes dos frames de comunicação entre o Atos A1 Soft e o controlador.

Configurando a comunicação TCP/IP:



Descrição dos campos:

Geral

- **Nome:** Nome definido para o controlador;
- **Família:** Família do controlador;

Comunicação

- **Nº estação:** Número do controlador;
- **Tipo:** Tipo de comunicação - TCP/IP;
- **IP:** Endereço IP para comunicação;
- **Porta:** Porta de comunicação;
- **Timeout(ms):** Tempo em milissegundos que o Atos A1 Soft aguarda a resposta do controlador antes de sinalizar algum tipo de falha;
- **Tamanho do Bloco:** Quantidade de bytes dos frames de comunicação entre o Atos A1 Soft e o controlador.

Tarefas Preemptivas

É recomendado para sistemas que devem apresentar comportamento determinístico no tempo.

Neste sistema quando o intervalo de uma tarefa de maior prioridade vence, a tarefa em execução é suspensa e a de prioridade maior passa a executar imediatamente.

Quando esta de maior prioridade termina, a tarefa suspensa anteriormente volta a executar do ponto onde parou.

Tarefa Preemptiva por evento

Atos MPC4004BF

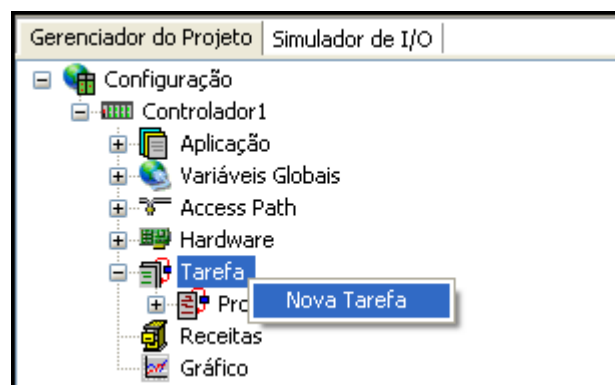
A tarefa preemptiva por evento é solicitada sempre que habilitada a entrada E2, executando o(s) programas associados a ela.



Para o controlador Atos MPC4004BF é permitida apenas uma tarefa preemptiva por evento, associada sempre a entrada E2 da CPU.

A criação dessa tarefa pode ser feita diretamente no item "Tarefa" do Gerenciador do Projeto ou na criar um novo programa.

Através do item Tarefa, clique com o botão direito do mouse e marque a opção "Nova tarefa", conforme mostra a figura abaixo:



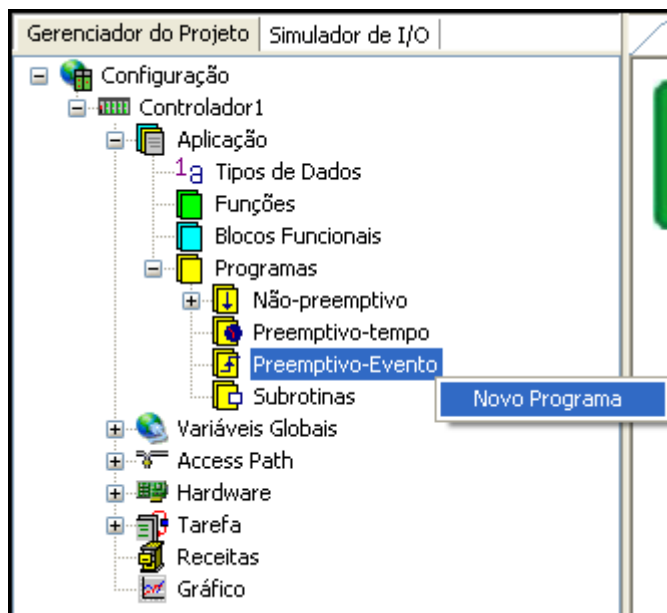
Nesta situação a janela "Propriedades da Tarefa", mostrada abaixo, será visualizada.



Descrição dos campos:

- **Nome:** Nome definido para tarefa que será criada;
- **Tipo:** O tipo definido para tarefa Preemptivo ou Não-Preemptivo;
- **Trigger:** Tipo do acionamento da tarefa;
- **Entrada:** Entrada que irá solicitar a tarefa, somente entrada E2.

Para criar a tarefa através de um novo programa, clique com o botão direito sobre o item Preemptivo-Evento do Gerenciador do Projeto e selecione a opção Novo Programa, conforme mostra a figura a seguir.



Nesta situação a janela "Propriedades do Programa" que será visualizada, e assim como a janela "Propriedades da Tarefa" os campos Nome, Tipo, Trigger e Entrada estarão disponíveis para configuração, conforme mostra a figura abaixo:

Propriedades do Programa

POU **Atos A1 Soft**

Nome do Programa
Programa1

Linguagens IEC61131

☒ Ladder

☐ FBD (Diagrama de blocos funcionais)

☐ SFC (Sequential Function Chart)

☐ IL (Lista de Instruções)

☐ ST (Texto Estruturado)

Configuração da Tarefa

☒ **Nova Tarefa**

Nome TaskPrograma1

Tipo Preemptivo

Trigger Evento

Entrada I0.2

☐ **Tarefa existente**

☐ **Não associar a uma Tarefa**

OK Cancelar

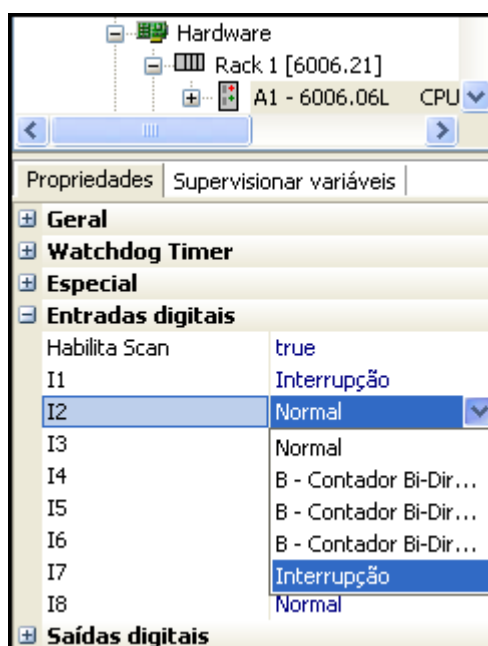
Atos MPC6006L

A tarefa preemptiva por evento é solicitada sempre que habilitada a entrada configurada, executando o(s) programas associados a ela.



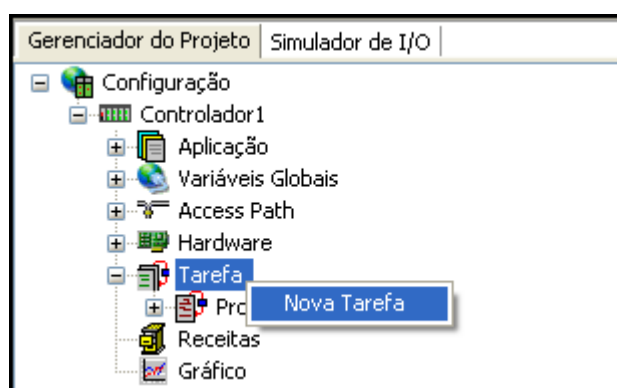
É permitida a configuração de apenas uma tarefa preemptiva para cada entrada da CPU.

Para utilizar as entradas como interrupção é necessário configurá-las. Para isso na aba Propriedades da CPU configurada no projeto, altere a configuração das entradas, conforme mostrado na imagem a seguir:

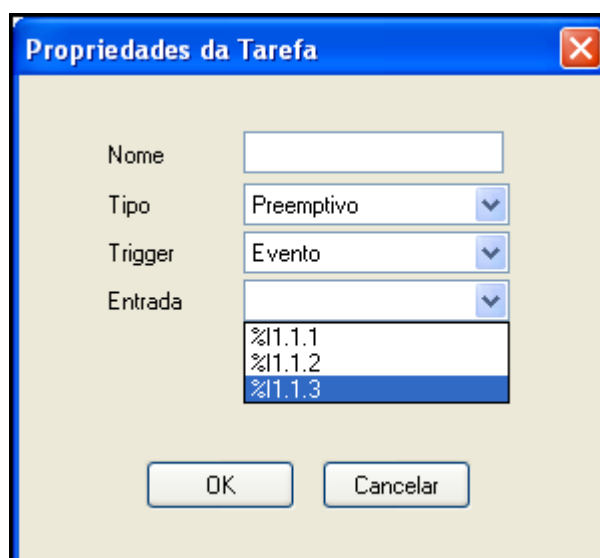


A criação dessa tarefa pode ser feita diretamente no item "Tarefa" do Gerenciador do Projeto ou na criar um novo programa.

Através do item Tarefa, clique com o botão direito do mouse e marque a opção "Nova tarefa", conforme mostra a figura abaixo:



Nesta situação a janela "Propriedades da Tarefa", mostrada abaixo, será visualizada.



A janela "Propriedades da Tarefa" possui os seguintes campos:

- Nome: Campo de texto vazio.
- Tipo: Menu suspenso com "Preemptivo" selecionado.
- Trigger: Menu suspenso com "Evento" selecionado.
- Entrada: Menu suspenso com opções "%I1.1.1", "%I1.1.2" e "%I1.1.3".

Botões "OK" e "Cancelar" estão localizados na base da janela.

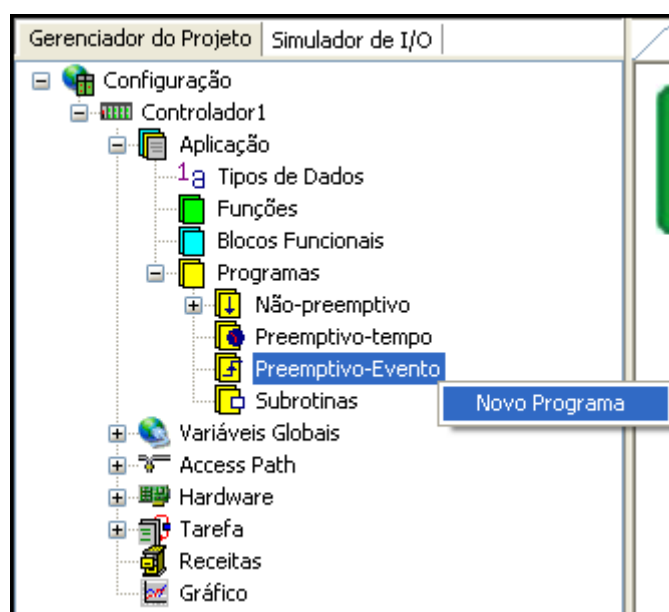
Descrição dos campos:

- **Nome:** Nome definido para tarefa que será criada;
- **Tipo:** O tipo definido para tarefa Preemptivo ou Não-Preemptivo;
- **Trigger:** Tipo do acionamento da tarefa;
- **Entrada:** Entrada que irá solicitar a tarefa.



No campo Entrada, só serão mostradas as entradas que já foram configuradas como interrupção.

Para criar a tarefa através de um novo programa, clique com o botão direito sobre o item Preemptivo-Evento do Gerenciador do Projeto e selecione a opção Novo Programa, conforme mostra a figura a seguir.



Nesta situação a janela "Propriedades do Programa" que será visualizada, e assim como a janela "Propriedades da Tarefa" os campos Nome, Tipo, Trigger e Entrada estarão disponíveis para configuração, conforme mostra a figura abaixo:

Propriedades do Programa

POU Atos A1 Soft

Nome do Programa
Programa1

Linguagens IEC61131

☒ Ladder

☐ FBD (Diagrama de blocos funcionais)

☐ SFC (Sequential Function Chart)

☐ IL (Lista de Instruções)

☐ ST (Texto Estruturado)

Configuração da Tarefa

☒ **Nova Tarefa**

Nome TaskPrograma1

Tipo Preemptivo

Trigger Evento

Entrada

☐ **Tarefa existente**

☐ **Não associar a uma Tarefa**

OK Cancelar

Tarefa preemptiva por tempo

A configuração das Tarefas preemptivas por tempo é igual para todos os controladores **Atos MPC4004BF** e **Atos MPC6006L**.

Tendo como base o conceito de tasks preemptivas, a tarefa preemptiva por tempo é solicitada sempre que atingido o tempo configurado executando o(s) programa associado a ela.

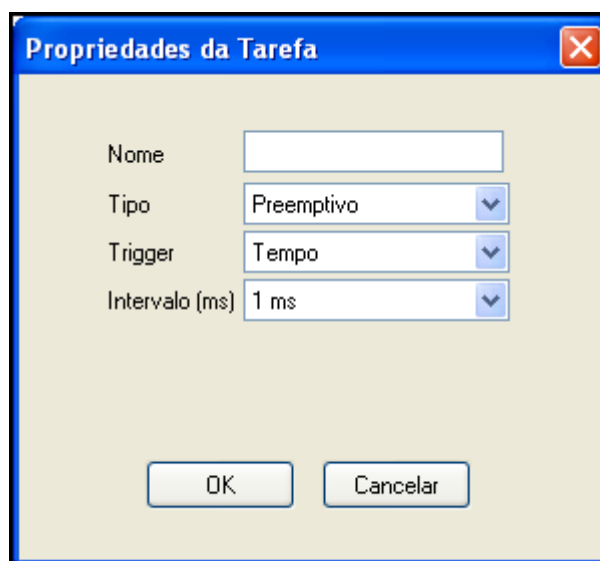
É permitida a configuração de apenas uma tarefa preemptiva por tempo para cada projeto.

A criação dessa tarefa pode ser feita diretamente no item "Tarefa" do Gerenciador do Projeto ou na criar um novo programa.

Através do item Tarefa, clique com o botão direito do mouse e marque a opção "Nova tarefa", conforme mostra a figura abaixo:



Nesta situação a janela "Propriedades da Tarefa" será visualizada, conforme mostrado abaixo:



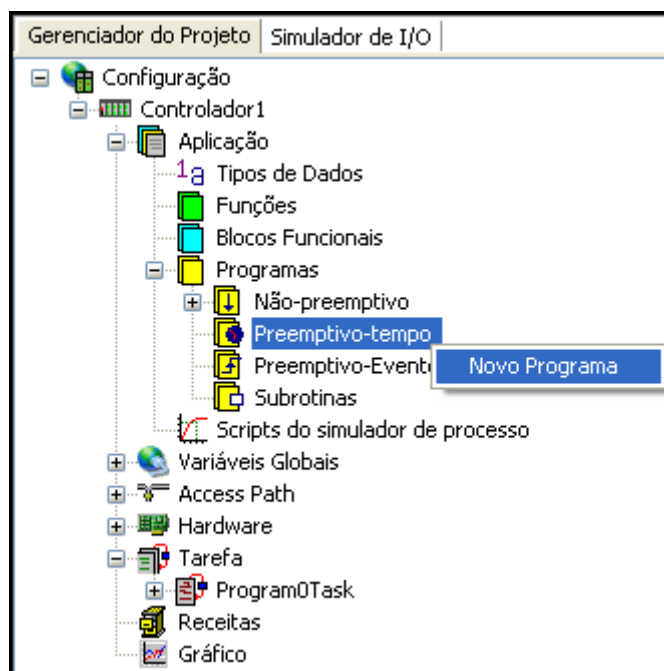
Descrição dos campos:

- **Nome:** Nome definido para tarefa que será criada;
- **Tipo:** O tipo definido para tarefa Preemptivo ou Não-Preemptivo;
- **Trigger:** Tipo do acionamento da tarefa;
- **Intervalo (ms):** Intervalo de tempo definido para o acionamento da tarefa.

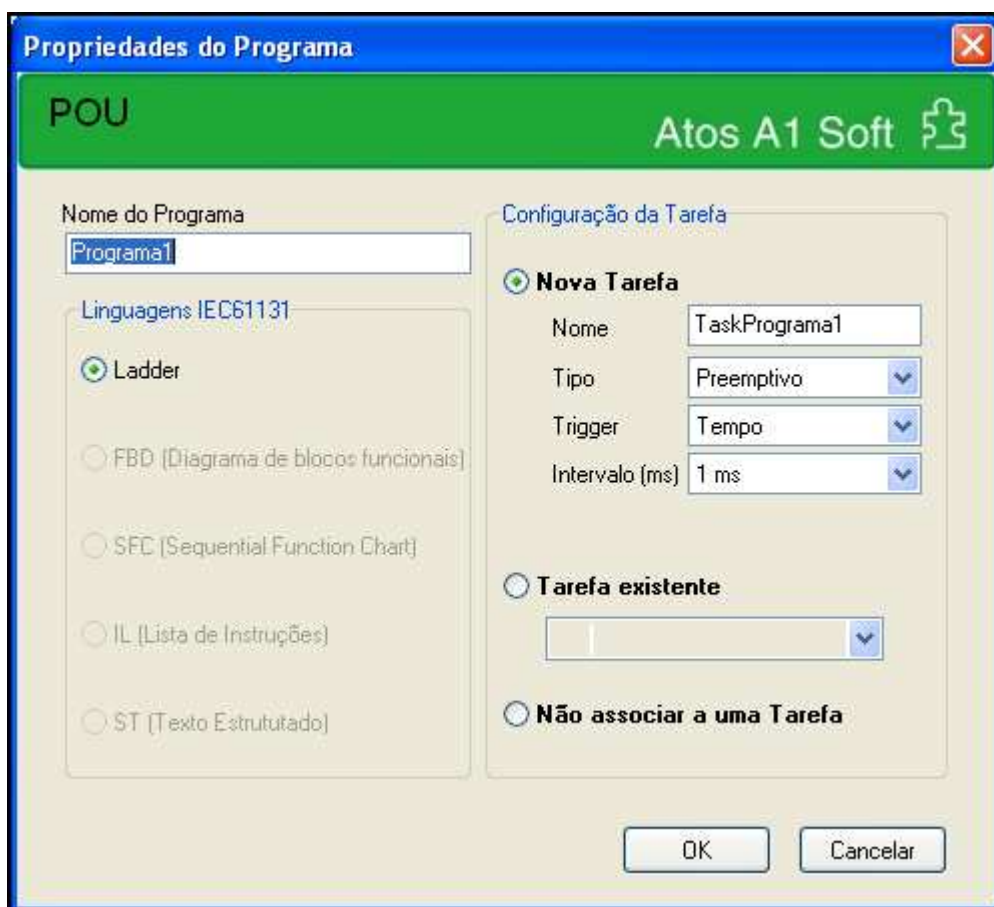


O campo Intervalo, só será mostrado depois de configurado o Tipo Preemptivo e o Trigger Tempo.

Para criar a tarefa através de um novo programa, clique com o botão direito sobre o item Preemptivo-tempo do Gerenciador do Projeto e selecione a opção Novo Programa, conforme mostra a figura a seguir.



Nesta situação a janela "Propriedades do Programa" que será visualizada. Assim como a janela "Propriedades da Tarefa" os campos Nome, Tipo, Trigger e Intervalo estarão disponíveis para configuração, conforme mostra a figura abaixo:



Download/Upload

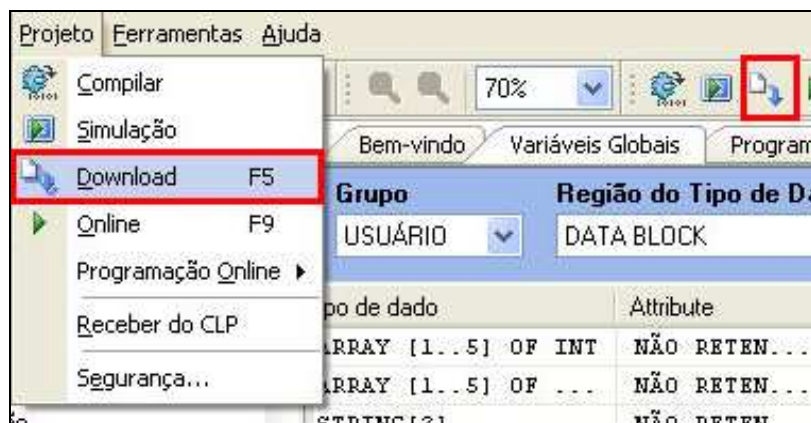
Realizando Download do projeto

Esta opção permite enviar o programa desenvolvido no Atos A1 Soft para o CLP. Antes de realizar o envio, o software automaticamente faz a compilação do projeto, verificando os possíveis erros existentes.

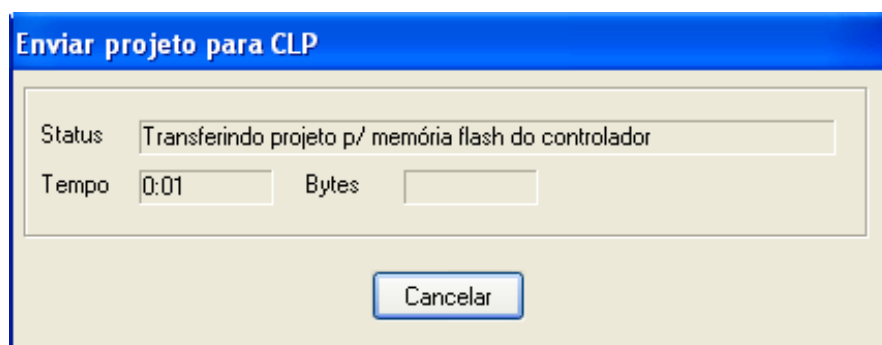
Os avisos e erros de compilação são mostrados na janela de mensagens, acessada pelo menu Exibir, opção Mensagens.

Para enviar o programa para o CLP, siga o seguinte procedimento:

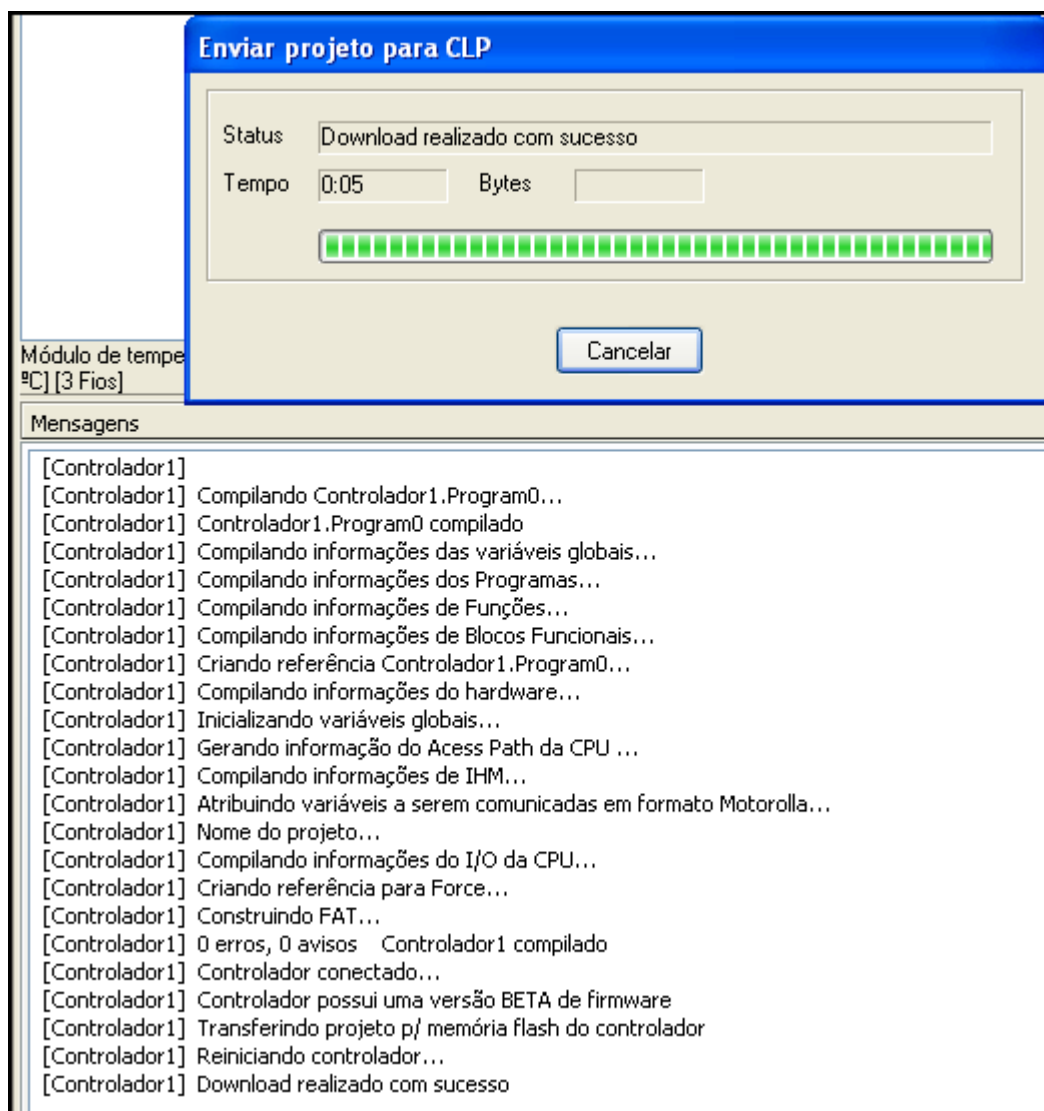
- No menu Projeto selecione a opção “Download”, ou através do botão , conforme mostra a figura abaixo:



Após a compilação automática, caso nenhum erro seja encontrado, inicia-se o envio do programa. O status do envio é feito através da janela “Enviar projeto para o CLP” mostrada abaixo.



Após enviado o projeto para o CLP, será mostrada no campo Status da janela, a mensagem "Download realizado com sucesso", mensagem que também poderá ser visualizada na janela de Mensagens do Atos A1 Soft, conforme mostrado nas figuras abaixo.



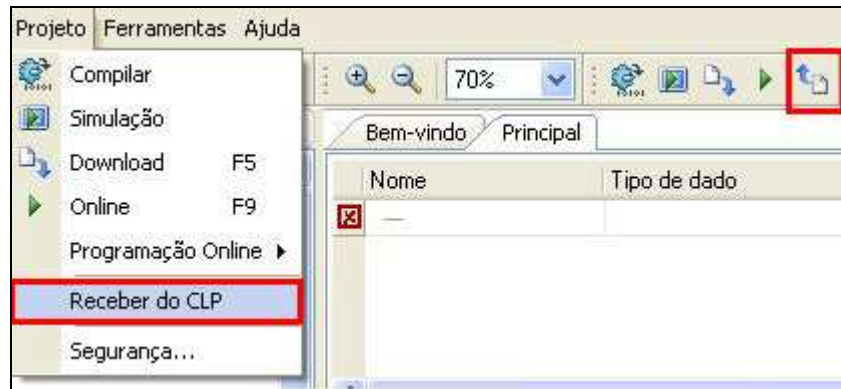
O procedimento de envio de projeto para o CLP não transfere o arquivo fonte com todas as informações do projeto. Para isto, deve-se utilizar a opção “Enviar projeto com fonte” ([Ver Fonte do Projeto página 70](#)).

Realizando Upload do projeto

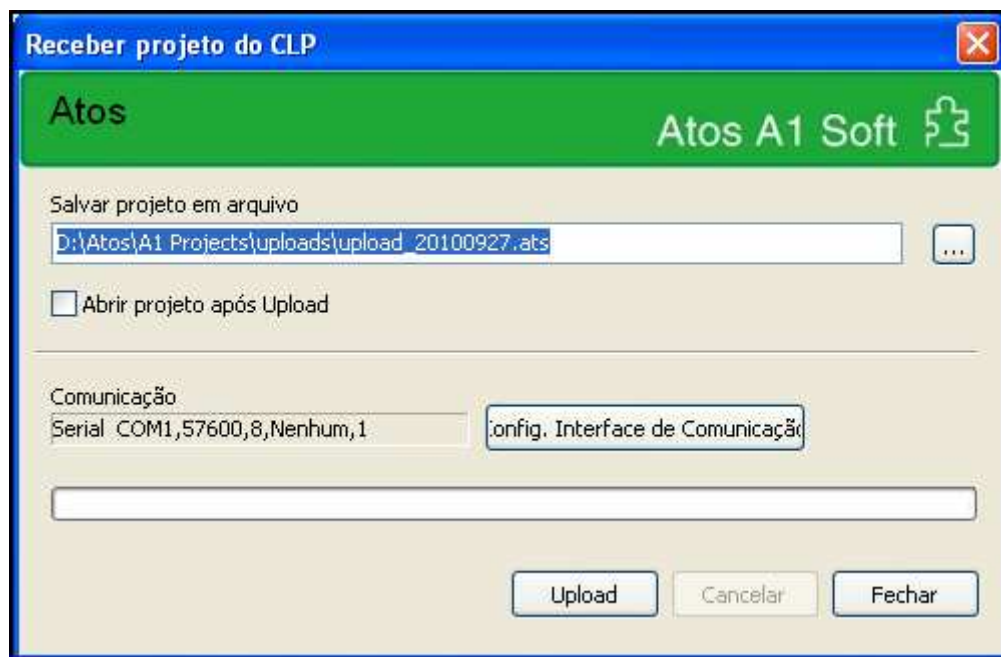
Este recurso permite receber um programa do CLP, mas para isso o programa deve ter sido enviado junto com o fonte do projeto (ver Fonte do Projeto página 69).

Verificada essa condição para o recebimento do projeto, os procedimentos a serem realizados são os seguintes:

- No menu Projeto selecione a opção “Receber Projeto”, ou através do botão , conforme mostra a figura a seguir:



Com isso a janela “Receber do CLP” será aberta, como mostrada abaixo.



- O software automaticamente gera um arquivo para que o programa que será recebido seja gravado, ou clique no botão , para selecionar outro arquivo.
- Através do botão , a janela “Ferramentas do CLP” (Ver Ferramentas do CLP página.103) é aberta, possibilitando que se realize configurações de comunicação entre o CLP e o software;
- Com as configurações corretas, clique no botão . O programa gravado no CLP será transmitido ao software e salvo no arquivo selecionado.

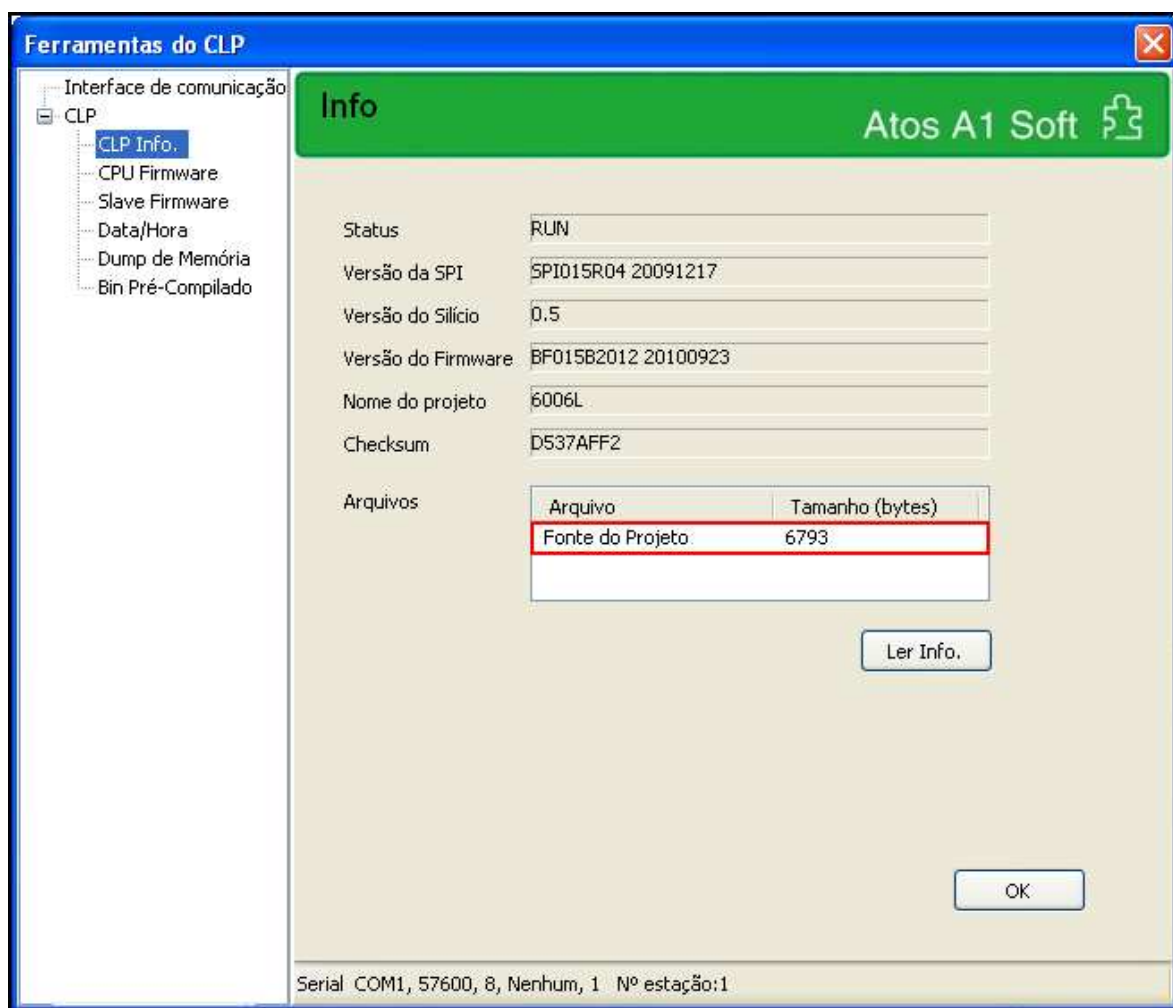
Fonte do projeto

O fonte do projeto é o arquivo ATS gerado pelo software Atos A1 Soft. Este arquivo pode ser armazenado no CLP junto com o arquivo compilado. Para isto deve-se utilizar a opção "Enviar projeto c/ fonte".

Como verificar se o CLP possui fonte do projeto

Para verificar se o programa gravado no CLP foi transferido com o fonte, utilize a janela de Ferramentas do CLP, selecionando o item CLP Info. .

Clique no botão "Ler Info.". Os campos da janela serão preenchidos com as informações do CLP conectado, e no campo Arquivos será mostrado se o programa gravado possui o arquivo fonte do projeto, conforme o exemplo abaixo:

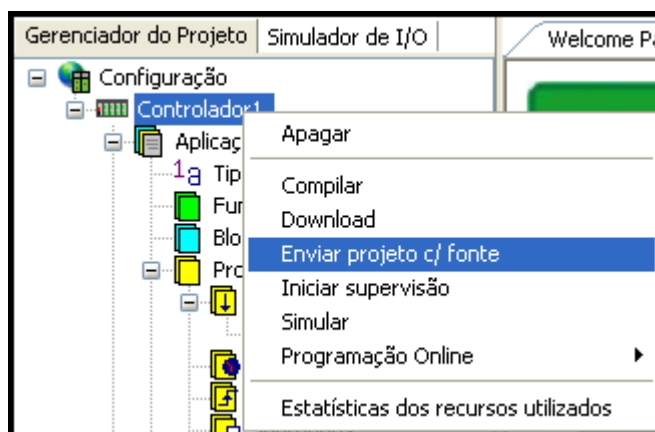


Somente será possível realizar o recebimento do programa se existir no CLP o arquivo fonte do projeto.

Como enviar um programa com fonte do projeto

Para enviar um programa com fonte do projeto o seguinte procedimento deve ser realizado:

- No gerenciador de projeto, clique com o botão direito sobre o controlador do respectivo programa, e selecione a opção Enviar projeto com fonte, conforme mostra a figura abaixo:

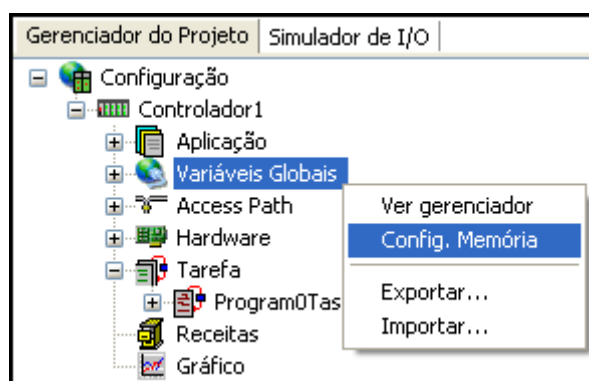


Realizado o download com sucesso, o programa gravado no CLP possuirá o arquivo fonte necessário para realizar o Upload, pelo software Atos A1 Soft.

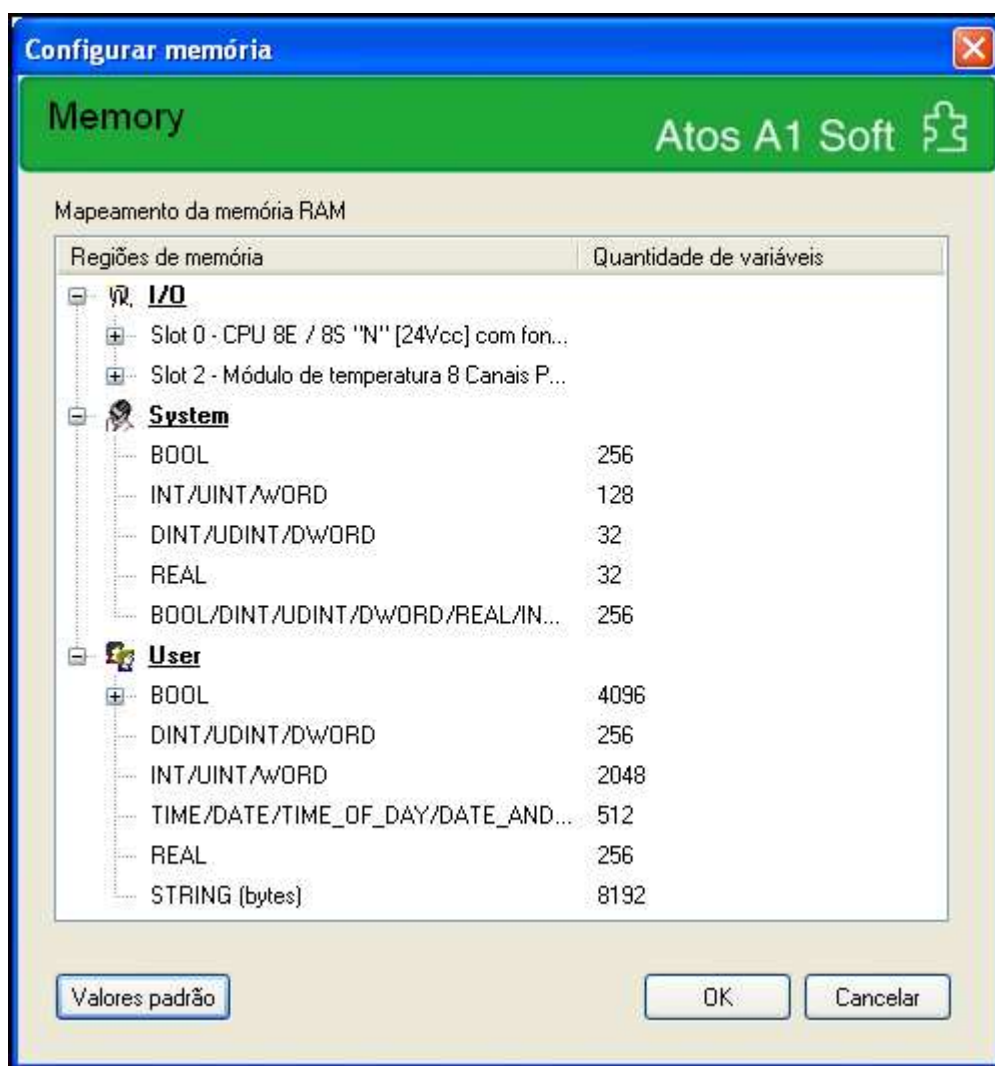
Configuração de Memória

A configuração do mapeamento de memória é um recurso presente nas CPUs Atos MPC4004BF que permite redefinir o tamanho de cada uma das regiões de usuário do CLP.

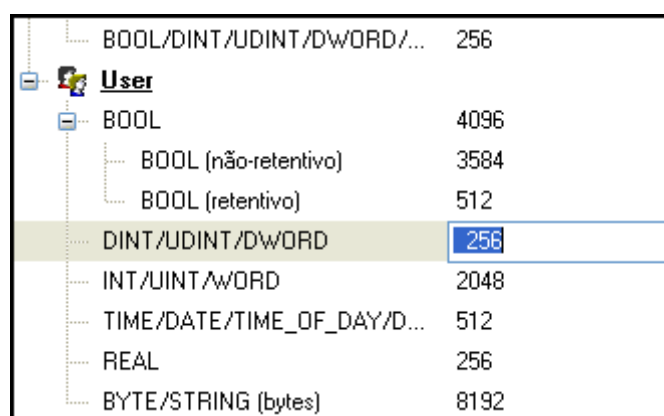
Para realizar essa configuração, na guia "Gerenciador de Variáveis", clique com o botão direito sobre o item "Variáveis Globais", conforme mostra a figura abaixo:

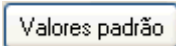


Selecione a opção Config. Memória para visualizar a janela "Configuração de Memória", mostrada abaixo.



Qualquer região de usuário pode ter seu tamanho alterado, conforme mostrado abaixo.

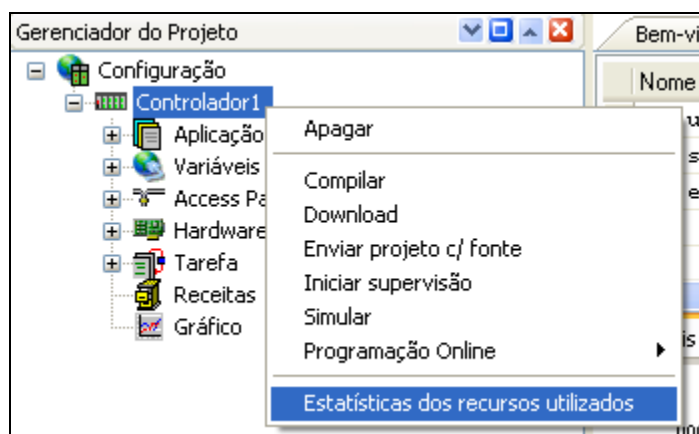


Para retornar a configuração para os valores padrão do software, clique no botão .

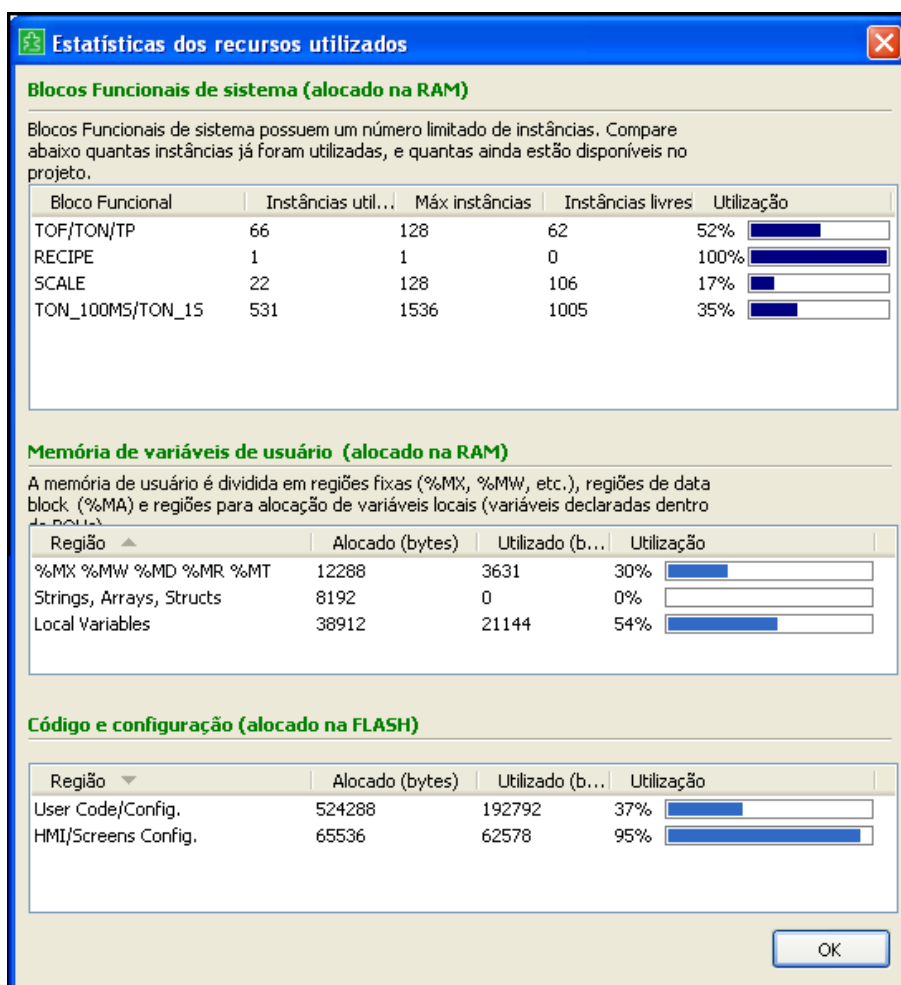
Visualização dos Recursos Utilizados

O Software Atos A1 Soft permite a visualização dos recursos que estão sendo utilizados no projeto. Com isso é possível verificar a quantidade de recursos ainda disponíveis.

Para acessar esta informação, clique com o botão direito sobre o controlador do projeto, e selecione o item "Estatística dos recursos utilizados", conforme mostrado na figura abaixo:



Descrição da janela Estatística dos recursos utilizados



Blocos funcionais de sistema

A quantidade de blocos funcionais é limitada. Através dessa janela verifica-se o

quanto já está sendo utilizado e a quantidade ainda disponível para uso.

Descrição dos campos:

- **Bloco Funcional:** Define a que bloco ou grupo de blocos funcionais se referem os dados;

No caso de um grupo de blocos, deve-se observar que as quantidades visualizadas são referentes a todo o grupo, e não a cada bloco separadamente.



- **Instâncias utilizadas:** Quantidade de blocos funcionais que estão sendo utilizados no projeto;
- **Máx. instâncias:** Quantidade máxima de blocos que podem ser utilizados;
- **Instâncias livres:** Quantidade de blocos que ainda podem ser utilizados no projeto;
- **Utilização:** Mostra em porcentagem e graficamente o quanto está sendo utilizado o bloco/grupo em relação a quantidade máxima.

Memórias de variáveis do usuário

As variáveis utilizadas pelo usuário são divididas em regiões:

- Fixa: %MX,%MW,%MD,etc;
- Data Block: %MA;
- Locais: variáveis que são declaradas dentro das POU's.

Para cada região existe um limite de memória disponível.

Descrição dos campos:

- **Região:** A região (fixa, data block ou local) a que se referem os dados;
- **Alocado (bytes):** Quantidade de memória em bytes, reservada para a determinada região de variáveis;
- **Utilizado (bytes):** Quantidade de memória em bytes, que está sendo utilizada pela região;
- **Utilização:** Mostra em porcentagem e graficamente o quanto está sendo utilizada a memória reservada para a região de variáveis.

Declaração de Array

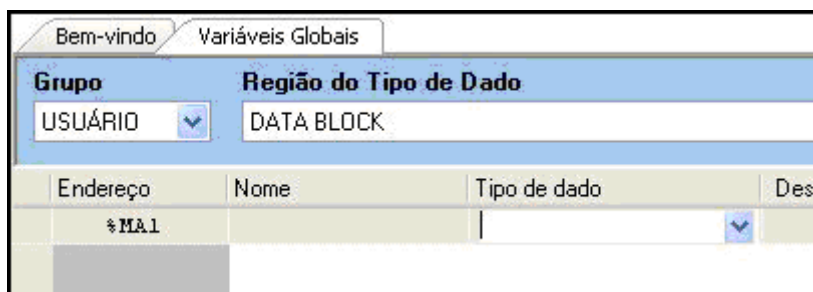
Tipo de dado: ARRAY	
TAMANHO	Até 2000 posições.
INTERVALO	-----

Forma de declarar variáveis do tipo ARRAY no item **Variáveis Globais**.

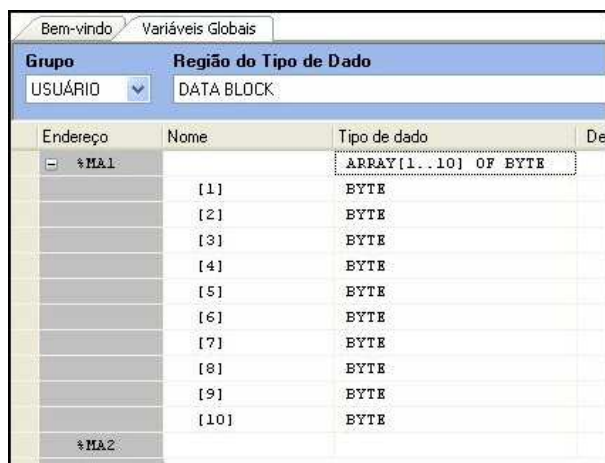
No grupo de variáveis de **usuário**, marque a opção **DATA BLOCK** no campo **Região do tipo de dado**.

Para declarar um ARRAY para qualquer tipo de dado (exceto STRING) faça:

- Insira no campo **Nome** um nome para a variável, pois será necessário na declaração do ladder;
- Selecione com o cursor do mouse o campo **Tipo de dado** para habilitar escrita;



- Em seguida digite **ARRAY[X..Y] OF AAAA**, onde:
- **X** representa a primeira posição da variável no ARRAY;
- **Y** representa a última posição da variável no ARRAY;
- **AAAA** representa o nome do tipo de dado, por exemplo, BOOL, BYTE, WORD, DWORD, INT, DINT, UINT, UDINT, REAL, DATE, TIME_OF_DAY, DATE_AND_TIME e TIME.



Veja o exemplo da figura ao lado, na qual foi declarado um ARRAY de BYTE com 10 posições.

Quando declaramos um Valor inicial no cabeçalho do array, todas as posições assumem este mesmo valor, conforme mostra a figura abaixo:

Variáveis Globais						
Grupo		Região do Tipo de Dado				
USUÁRIO		DATA BLOCK				
Endereço	Nome	Tipo de dado	Attribute	Valor Inicial	De	
- %MA1	tempo	ARRAY [1..5] OF ...	NÃO RETEN...	0		
	tempo[1]	TIME	NÃO RETEN...	0		
	tempo[2]	TIME	NÃO RETEN...	0		
	tempo[3]	TIME	NÃO RETEN...	0		
	tempo[4]	TIME	NÃO RETEN...	0		
	tempo[5]	TIME	NÃO RETEN...	0		



Nesse caso a alteração do valor inicial para cada posição NÃO é possível.

Para atribuir valores diferentes para cada posição, o campo Valor inicial do cabeçalho deverá estar em branco, conforme o exemplo abaixo:

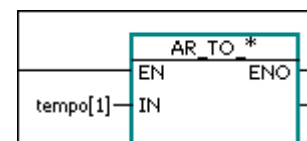
Variáveis Globais						
Grupo		Região do Tipo de Dado				
USUÁRIO		DATA BLOCK				
Endereço	Nome	Tipo de dado	Attribute	Valor Inicial	De	
- %MA1	tempo	ARRAY [1..5] OF ...	NÃO RETEN...			
	tempo[1]	TIME	NÃO RETEN...	10		
	tempo[2]	TIME	NÃO RETEN...	15		
	tempo[3]	TIME	NÃO RETEN...	16		
	tempo[4]	TIME	NÃO RETEN...	22		
	tempo[5]	TIME	NÃO RETEN...	20		

Para excluir uma variável do tipo array, selecione-a pelo cabeçalho com o cursor do mouse, clique com o botão direito e escolha a opção **Apagar variável**.

- **Declaração de um Array no ladder**

Na programação do ladder a variável deve ser declarada da seguinte forma:

nome da variável[Z], onde **Z** é a posição do dado no array, conforme mostrado na figura ao lado.



Declaração de String

Tipo de dado: String	
TAMANHO	256 caracteres.
INTERVALO	Caracteres ASCII
Exemplo de constante	STRING#'Atos'

Outra forma de declarar uma variável do tipo STRING é no item **Variáveis Globais**.

No grupo de variáveis de **usuário**, marque a opção **DATA BLOCK** no campo **Região do tipo de dado**.

Para declarar uma variável STRING com o número de caracteres faça:

- Selecione no campo **Tipo de dado** o item STRING;
- Em seguida digite **[xxx]** na frente da palavra STRING, onde **xxx** representa a quantidade de caracteres.

Veja exemplo da figura abaixo, na qual foi declarada uma variável do tipo STRING com 200 caracteres.

Endereço	Nome	Tipo de dado	Desc
%MA1		STRING[200]	
%MA2			

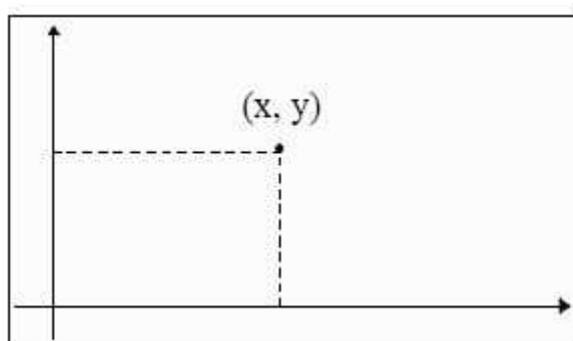
Caso queira excluir a variável, selecione com o cursor do mouse, clique com o botão direito e marque a opção **Apagar variável**.

Struct – Estrutura de dados

Struct é um tipo de dado criado na forma de estrutura capaz de agrupar diferentes tipos de variáveis. Pode-se fazer uma analogia com uma estante de livros (**estrutura**), no qual pode comportar um determinado número de livros (**dados**). O espaço de cada livro estará garantido, esteja ou não na estante.

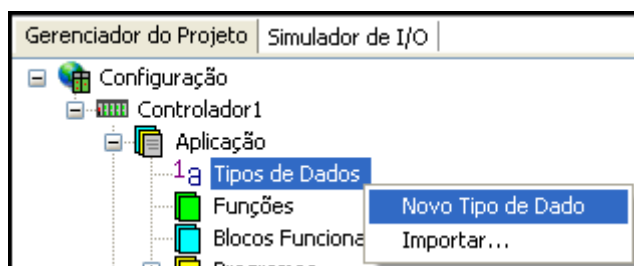
Fixando e identificando o local (**membro**) onde cada livro (**dado**) deve ser armazenado, sempre será possível encontrar e guardar cada um dos livros.

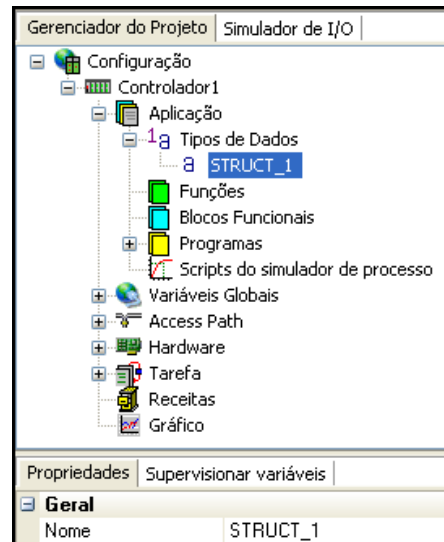
Matematicamente uma Struct seria armazenar um **dado** em um determinado ponto por meio das coordenadas (X,Y) de um plano cartesiano, onde **X** simboliza a estrutura (**Struct**) e **Y** uma variável agrupada pela estrutura.



Criando uma Struct

Para criar uma Struct no Atos A1 Soft, selecione com o cursor do mouse o item **Tipos de dados** na área do **Gerenciador do projeto**. Clique com o botão direito do mouse para exibir a caixa de informação **Novo tipo de dado** e marque esta opção. Caso seja necessário excluir a Struct inserida, marque a opção **Apagar tipo de dado**.

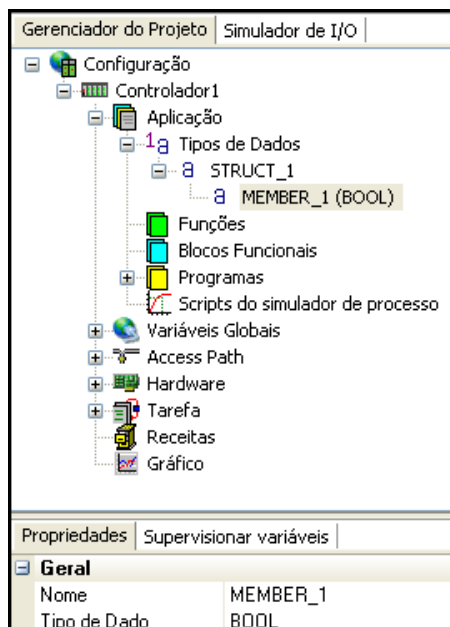
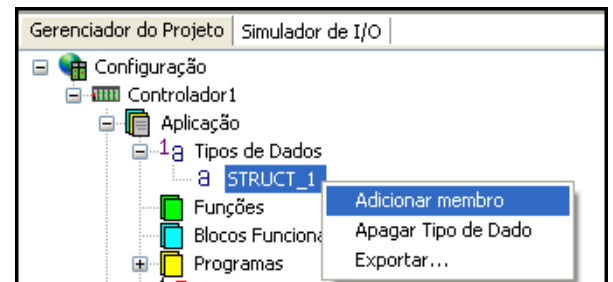




Depois de inserida a Struct, a guia de **Propriedades** irá exibir um campo para nomear a Struct inserida. Para visualizar esta guia novamente, basta selecionar com o cursor do mouse a Struct desejada.

Inserindo Membros

Para inserir um membro na Struct criada, selecione com o cursor do mouse a Struct na área do **Gerenciador do projeto**, clique com o botão direito do mouse para exibir a caixa de informação e marque esta opção **Adicionar membro**.



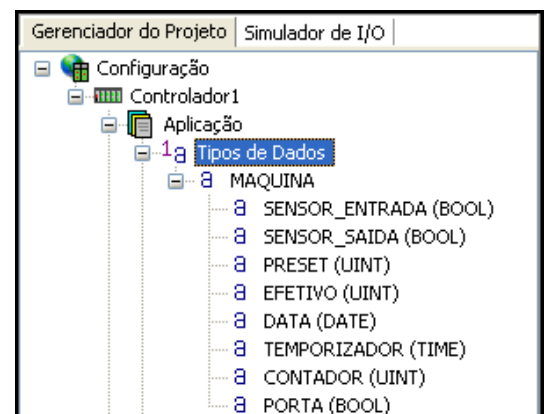
Caso seja necessário excluir algum membro da Struct, selecione o membro a ser excluído, clique com o botão direito do mouse e marque a opção **Apagar membro**.

Depois de inserido o membro, a guia de **Propriedades** irá exibir um campo para nomear o membro inserido e selecionar o tipo de dado.

Todos os tipos de dados primários são suportados para um membro, **exceto** os tipos de dados **String** e **Array**.

Para visualizar esta guia novamente, basta selecionar com o cursor do mouse o membro da Struct desejado.

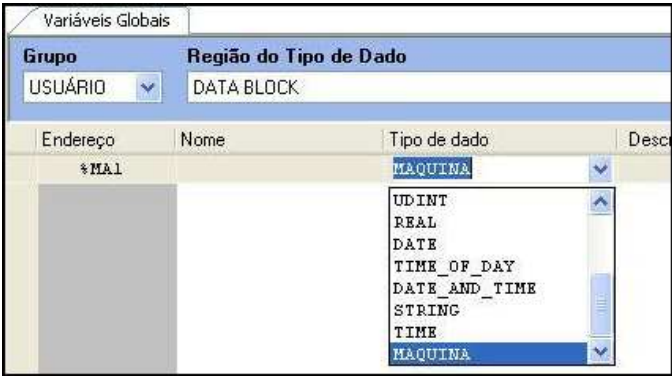
A figura ao lado é um exemplo de uma Struct nomeada como MAQUINA e com seus devidos membros nomeados e definidos com tipos de dados, formando assim o novo tipo de dado MAQUINA que agrupa variáveis de diferentes tipos de dados.



Usando Struct

Para utilizar o novo tipo de dado criado, o usuário deverá declará-la no **grupo de variáveis de usuário** em **Variáveis Globais**. Marque a opção **DATA BLOCK** no campo **Região do tipo de dado**.

Localize no campo **Tipo de dado** o nome da Struct desejada (novo tipo de dado criado) dentre os tipos de dados existente do sistema. No campo **Nome**, **obrigatoriamente** deve ser inserido um nome para a variável que utiliza o novo tipo de dado (Struct). Este nome da variável será necessário na declaração em ladder.



Após ter feito isso, será exibido o tipo de dado (Struct) com todas as variáveis internas listadas (membros) que compõe o mesmo.



Array de Struct

O modo de declarar não é muito diferente que um Array convencional das variáveis de sistema. Forma de declarar:

- Insira no campo **Nome** um nome para a variável, pois será necessário na declaração do ladder;
- Selecione com o cursor do mouse o campo **Tipo de dado** para habilitar escrita;
- Em seguida digite **ARRAY[X..Y] OF AAAA**, onde:
 - **X** representa a primeira posição da variável no ARRAY;
 - **Y** representa a última posição da variável no ARRAY;
 - **AAAA** representa o nome do Struct (tipo de dado), por exemplo, MAQUINA.

Variáveis Globais		
Grupo	Região do Tipo de Dado	
USUÁRIO	DATA BLOCK	
Endereço	Nome	Tipo de dado
- *MA1	MAQUINA_1	ARRAY[1..3] OF MAQUINA
+	MAQUINA_1[1]	MAQUINA
+	MAQUINA_1[2]	MAQUINA
+	MAQUINA_1[3]	MAQUINA

Caso seja necessário excluir um array, selecione o mesmo com o cursor do mouse, clique com o botão direito e marque a opção **Apagar variável**.

Na programação do ladder a variável deve ser declarada da seguinte forma:

nome do array[Z].nome da variável no Struct, onde **Z** é a posição da variável no array.

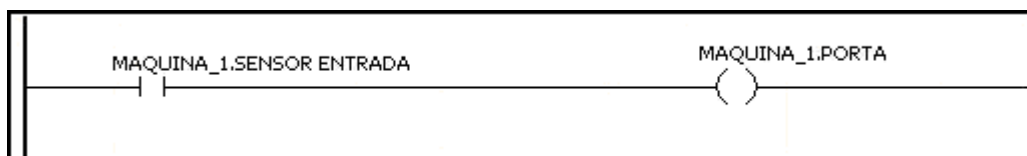
Variáveis Globais		
Grupo	Região do Tipo de Dado	
USUÁRIO	DATA BLOCK	
Endereço	Nome	Tipo de dado
*M1	MAQUINA_1	ARRAY[1..3] OF MAQUINA
	MAQUINA_1[1]	MAQUINA
	SENSOR ENTRADA	BOOL
	SENSOR SAÍDA	BOOL
	PRESET	UINT
	EFETIVO	UINT
	DATA	DATE
	TEMPORIZADOR	TIME
	CONTADOR	UINT
	PORTA	BOOL
	MAQUINA_1[2]	MAQUINA
	SENSOR ENTRADA	BOOL
	SENSOR SAÍDA	BOOL
	PRESET	UINT
	EFETIVO	UINT
	DATA	DATE
	TEMPORIZADOR	TIME
	CONTADOR	UINT
	PORTA	BOOL
	MAQUINA_1[3]	MAQUINA
	SENSOR ENTRADA	BOOL
	SENSOR SAÍDA	BOOL
	PRESET	UINT
	EFETIVO	UINT
	DATA	DATE
	TEMPORIZADOR	TIME
	CONTADOR	UINT
	PORTA	BOOL

Declaração no ladder

Para declarar uma variável no ladder pertencente a uma Struct, **primeiro deve ser inserido o nome da Struct precedido com ponto (.) e em seguida o nome da variável**.

Para o exemplo demonstrado abaixo temos:

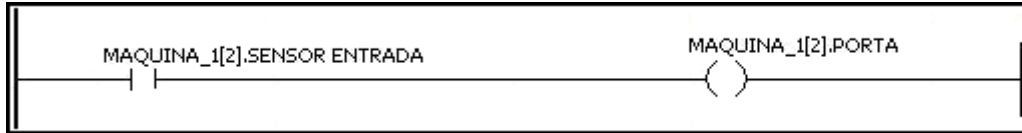
- Contato de entrada - MAQUINA_1.SENSOR ENTRADA;
- Saída - MAQUINA_1.PORTA;



Declaração do Array no ladder

Na programação do ladder a variável deve ser declarada da seguinte forma:

nome do array[Z].nome da variável no Struct, onde **Z** é a posição da variável no array.



Programação Online

A opção "**Programação Online**", recurso existente para CPUs Atos MPC4004BF, permite realizar alterações no programa de usuário, sem a necessidade de passar o CLP para o modo "**Prog**", evitando assim parar a máquina ou processo.



Este recurso deve ser utilizado com total cuidado e atenção por parte do usuário, pois alterações no programa feitas de forma indevida podem causar danos ao operador ou a máquina/processo.

A Schneider Electric se isenta de qualquer responsabilidade sobre danos causados pelo uso indevido deste recurso.

Basicamente as alterações online devem ser entendidas como sendo "**pequenas alterações**" em ladder, onde o usuário possui total controle de que não haverá risco ao processo ou a operadores, excluindo assim grandes alterações.

Alterações em programação Online:

- **Hardware:** NENHUMA alteração pode ser feita;
- **IHM:** NENHUMA alteração pode ser feita;
- **Ladder.**

Ladder:

ATENÇÃO para o que é:

☒ PERMITIDO e ☐ NÃO PERMITIDO na *Programação Online*.

OPÇÕES GENÉRICAS

- ☒ Apagar e inserir uma linha inteira;
- ☒ Copiar e colar conjunto de linhas;
- ☒ Apagar, inserir e modificar contatos e saídas;
- ☐ Criar subrotinas;
- ☐ Criar e apagar CALLs de subrotinas;
- ☒ Criar novas variáveis globais e utilizá-las no ladder;
- ☐ Criar novas variáveis locais;
- ☐ Mudar inicialização de variáveis globais;
- ☐ Mudar inicialização de variáveis locais.

PROGRAMA (PROGRAM)

- ☐ Criar novos programas (*Programs*) em tarefas (*Tasks*) já existentes;
- ☐ Criar novas tarefas (*Tasks*);
- ☐ Apagar tarefas (*Tasks*) já existentes realocando programas (*Programs*) para outras tarefas (*Tasks*);
- ☐ Em programa (*Program*) já existente, criar novas variáveis locais;
- ☐ Em programa (*Program*) já existente, mudar inicialização de variáveis locais.

FUNÇÃO (FUNCTION) DO SISTEMA

- ☒ Inserir novas funções (functions);
- ☒ Apagar funções (functions) existentes;
- ☒ Alterar variáveis de entrada e saída.

FUNÇÃO (FUNCTION) DO USUÁRIO

- ✖ Criar novas funções (*functions*);
- ✖ Apagar, renomear, compilar ou exportar funções (*functions*) existentes;
- ☑ Apagar funções (*functions*) inseridas no programa;
- ☑ Inserir uma função (*functions*) já existente;
- ☑ Alterações com base nas opções genéricas;
- ✖ Alterar parâmetros e variáveis na estrutura da função (*function*).

BLOCOS FUNCIONAIS (FUNCTIONS BLOCKS) DO SISTEMA

- ☑ Inserir novos blocos funcionais (*functions blocks*);
- ☑ Apagar blocos funcionais (*functions blocks*) existentes;
- ☑ Alterar variáveis de entrada e saída.

BLOCOS FUNCIONAIS (FUNCTIONS BLOCKS) DO USUÁRIO

- ✖ Criar novos blocos funcionais (*functions blocks*);
- ☑ Inserir blocos funcionais (*functions blocks*) existentes;
- ✖ Apagar, renomear, compilar ou exportar blocos funcionais (*functions blocks*) existentes;
- ☑ Alterações com base nas opções genéricas;
- ✖ Alterar parâmetros e variáveis na estrutura dos blocos funcionais (*function block*).

Alterações de maior complexidade devem ser tratadas como sendo alterações pertinentes a um “**startup**”, onde condições são controladas e intertravamentos feitos para evitar operações indevidas durante a depuração da lógica.

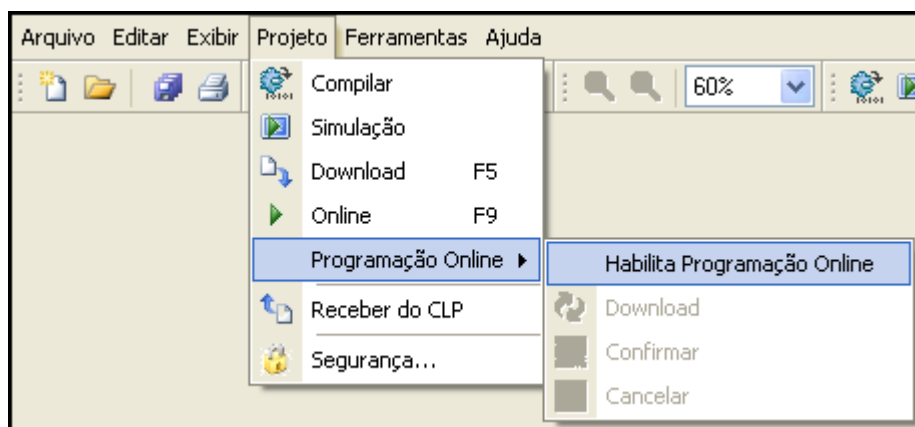
Portanto a modificação online não deve ser entendida como um recurso de desenvolvimento da máquina ou processo, e sim um recurso para corrigir pequenos desvios em campo.

Por motivos de segurança, para habilitar a programação online, o CLP deve estar conectado ao PC. A supervisão “**NÃO**” pode estar habilitada, pois o canal serial estando ocupado impede que o item **Programação online** seja habilitado.

Para habilitar a programação online no CLP, é necessário estar com o mesmo projeto armazenado na CPU aberto no software do Atos A1 Soft. Caso o projeto seja diferente, o Atos A1 Soft informa que o projeto do CLP não é igual ao projeto atual, impedindo que o menu de programação online seja habilitado.

A programação Online é habilitada da seguinte forma:

1. No menu "**Projeto**", posicione o mouse sobre a opção "**Programação Online**";
2. Espere até a opção se expandir e escolha a opção "**Habilita programação Online**", como mostrado abaixo:



Caso o programa armazenado na CPU seja diferente do aberto no software, o Atos A1 Soft informa que o projeto do CLP não é igual ao projeto atual, impedindo que o menu de programação online seja habilitado.

3. Feito isso, o aplicativo verifica com o CLP alguns parâmetros de inicialização da programação online, habilita as opções disponíveis no menu e o ícone  pode ser visto no canto superior direito do software, enquanto o modo de programação Online estiver habilitado. Feito isso, o aplicativo verificará com o CLP alguns parâmetros de inicialização da programação online e habilitará as opções disponíveis no menu.

Descrição do menu "Programação Online"

- **Download:** Esta opção realiza o envio do programa de usuário alterado para ser armazenado memória RAM do CLP. O usuário pode realizar quantos downloads desejar durante a programação online;
- **Confirmar:** Após realizar todas as alterações no CLP, selecione esta opção para gravar o novo programa de usuário modificado em memória FLASH do CLP. Isso se deve ao fato do controlador estar trabalhando em memória RAM. Com a opção "Confirmar", o Atos A1 Soft automaticamente salva a mesma versão do projeto no arquivo .ATS, garantido que ambos os programas sejam os mesmos (CLP e Atos A1 Soft).

Em caso de queda de energia, um procedimento padrão da CPU Atos MPC4004BF é carregar o programa de usuário (armazenado em memória FLASH) para a memória RAM. Se a programação online não for confirmada, em casos que o controlador for resetado, o programa de usuário será substituído pelo presente na memória FLASH, perdendo assim as alterações realizadas na programação online.

- **Cancelar:** Durante a programação online o usuário pode mandar cancelar. Neste caso, a programação online será desabilitada e o programa no CLP não será salvo na memória FLASH, ou seja, quando for efetuado um boot, o último programa em FLASH será carregado.

O programa local (projeto no Atos A1 Soft) permanece como o usuário deixou (com as modificações). **Cabe ao usuário não salvar as modificações feitas no projeto.**



Se o usuário mandar salvar o projeto durante a programação online, o Atos A1 Soft irá avisá-lo de que o projeto no CLP também deve ser salvo (em FLASH) para que a aplicação continue sendo válida (programa local continue sendo igual ao programa no CLP). Se o usuário não desejar salvar no CLP, a programação online não será mais válida, pois os dois programas estarão diferentes, obrigando o usuário a realizar o download "NORMAL" para o CLP novamente.

Supervisão

O que é supervisão?

A supervisão é um recurso disponível no software Atos A1 Soft, que permite monitorar variáveis de processo em tempo real, através de gráficos, janelas de supervisão de variáveis e no próprio ladder do projeto.

Além disso, permite ao usuário fazer modificações em tempo real dos valores e status de todas as variáveis do processo, sejam elas globais ou locais.

- **Tipos de supervisão disponíveis no Atos A1 Soft:**
 - Supervisão de Variáveis;
 - Supervisão Gráfica;
 - Supervisão Ladder.

Para iniciar a supervisão do CLP, selecione na guia Projetos, o item Online, ou pressione o botão  da barra de ferramentas.

O ícone , existente no canto superior direito do software, permanece piscando durante o modo de supervisão.



Durante a supervisão NÃO é possível editar o programa-usuário.

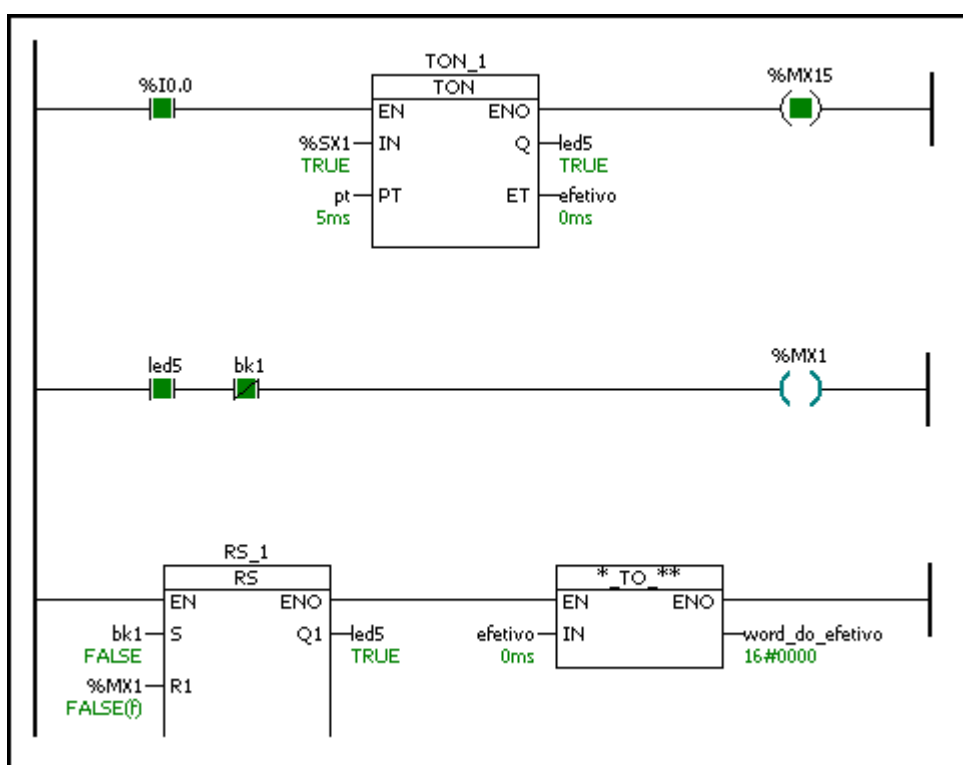
Supervisão Ladder

O recurso de supervisão ladder permite a visualização e análise do funcionamento do projeto no CLP. O status das entradas e saídas e o conteúdo das variáveis são exibidos no próprio programa ladder.

A representação da supervisão no ladder é feita da seguinte maneira:

- As saídas e os contatos NA são marcados com um retângulo colorido, sempre que estiverem ativos, ou seja, TRUE;
- As saídas invertidas e os contatos NF são marcados com um retângulo colorido, sempre que não estiverem ativos, ou seja, FALSE;
- As instruções exibem o conteúdo das variáveis de entrada e saída, na cor verde, imediatamente abaixo das mesmas.

Exemplo de supervisão no ladder:



Para iniciar a supervisão ladder, é necessário ter um projeto aberto.

1. Ative a janela do programa a ser supervisionado.

2. Inicie a supervisão através da guia *Projetos* ou clicando no botão correspondente na barra de ferramentas do aplicativo *Atos A1 Soft*.

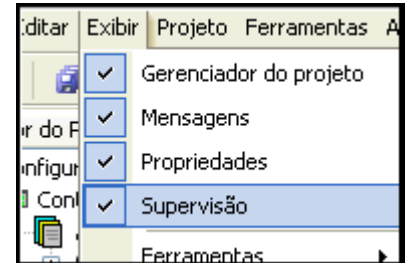
Supervisão de Variáveis

A janela de supervisão de variáveis permite visualizar/alterar o conteúdo de qualquer variável do CLP conectado ao Atos A1 Soft.

Em uma janela, o usuário pode determinar quais variáveis serão analisadas.

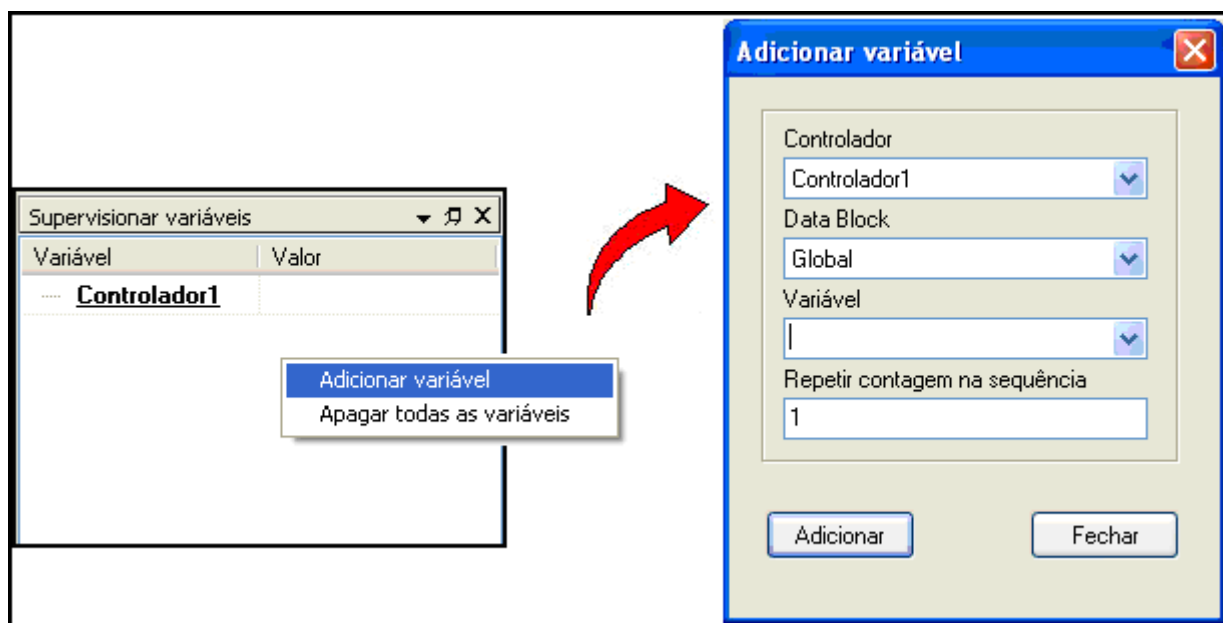
A supervisão de variáveis do Atos A1 Soft possui o recurso Force de Variáveis, que permite alterar o status de qualquer variável booleana do controlador.

Para visualizar a janela de supervisão de variáveis, é necessário que a mesma esteja habilitada. Isso pode ser verificado no menu Exibir, item supervisão, conforme mostra a figura ao lado.



- **Como adicionar variáveis**

Clique com o botão direito do mouse sobre a janela "Supervisionar variáveis". No menu pop-up apresentado, selecione a opção "Adicionar variável", conforme mostra a figura abaixo:



Na janela "Adicionar variável", temos os seguintes campos:

- **Controlador:** Define em qual controlador se encontra a variável;
- **Data Block:** Escopo da variável, no caso apenas as variáveis globais podem ser supervisionadas;
- **Variável:** Variável que se deseja adicionar;
- **Repetir contagem na sequência:** Número de variáveis que serão adicionadas na sequência a partir da variável já definida.

Supervisão gráfica

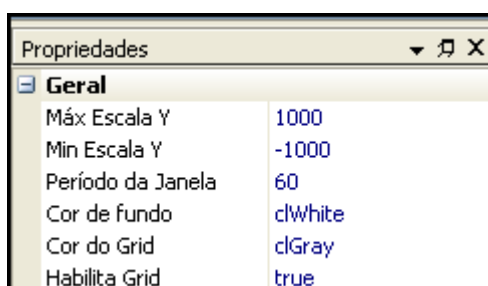
A supervisão gráfica permite visualizar o comportamento de uma ou mais variáveis em função do tempo.



Para adicionar um novo gráfico, clique com o botão direito do mouse sobre o item "Gráfico" do Gerenciador de Projetos e selecione "Novo Gráfico", conforme mostra a figura abaixo:

O número máximo de janelas de supervisão gráfica é de 16, onde cada janela pode ter até 20 penas (variáveis).

Depois de inserir o gráfico é necessário definir seus parâmetros na guia "Propriedades" do gráfico selecionado.



Descrição dos campos:

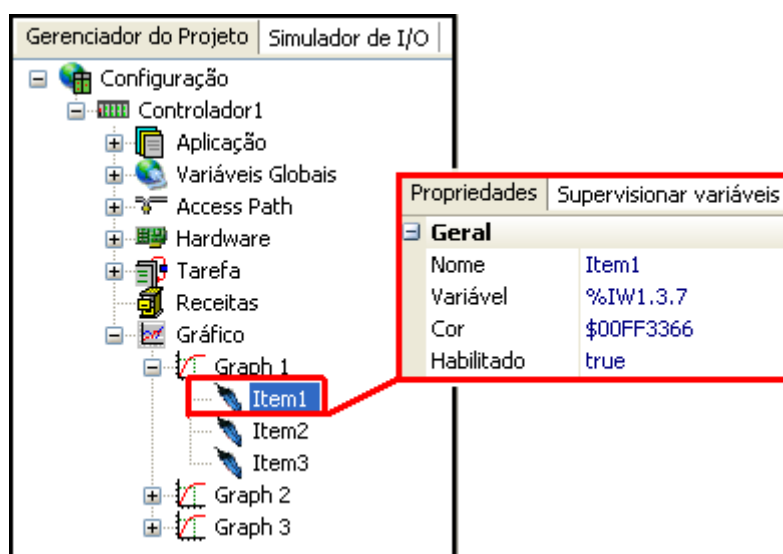
- **Máx Escala Y:** Valor máximo no eixo Y (vertical), valor entre -200000 a 200000;
- **Min Escala Y:** Valor mínimo do eixo Y, valor entre -200000 a 200000;
- **Período da Janela:** Tempo do ciclo do gráfico, valor limite do campo:360.
- **Cor de fundo:** Cor referente ao fundo do gráfico.
- **Cor do Grid:** Cor referente à grade das linhas de referência do gráfico.

Depois de configurado o gráfico é necessário vincular os eventos que contêm as variáveis que serão supervisionadas, esses eventos são chamados de Penas.

Para adicionar uma Pena, clique com botão direito do mouse sobre o item do gráfico desejado na guia de Gerenciador do Projeto, e selecione a opção Adicionar Pena, conforme a figura abaixo:



Após adicionada a pena, é necessário configurá-la, isso é feito na guia Propriedades, selecionando a pena em questão.



Descrição dos campos:

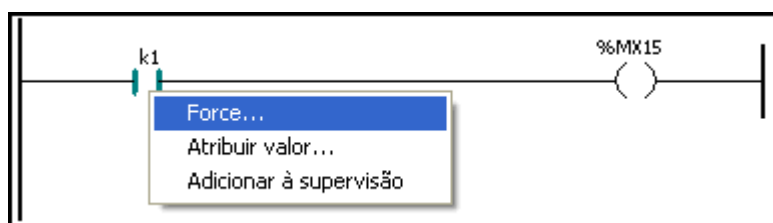
- **Nome:** Nome da pena;
- **Variável:** Endereço ou nome da variável que será visualizada no gráfico, tipos de dados permitidos: INT, DINT, UINT, REAL, BOOL;
- **Color :** Cor da linha referente a essa variável no gráfico;
- **Enabled:** Define se a pena está habilitada ou não, habilitada quando TRUE e desabilitada quando FALSE.

Force de variáveis

O recurso Force permite ao usuário alterar para TRUE ou FALSE, o status de qualquer variável booleana do controlador. É utilizado para analisar o efeito de um ou mais variáveis booleanas no programa de usuário, facilitando testes de simulação de diagramas lógicos em bancadas. O force é um recurso do aplicativo disponível somente durante a supervisão do projeto.

Para abrir a janela de FORCE utilize um dos dois caminhos a seguir:

- Diretamente no programa, clique com o botão direito do mouse sobre uma variável booleana e selecione o item "Force", conforme mostrado na figura abaixo;



- Através da janela de Supervisão de variáveis, clique com o botão direito sobre a variável que deseja forçar (conforme figura abaixo), e selecione o item "Force".



Na janela "Force de variável BOOLEANA", preencha os campos conforme a descrição abaixo:



Descrição dos campos:

- **Controlador:** O controlador que se encontra a variável;
- **Escopo:** O escopo da variável (global ou local);
- **Instância:** Número do bloco que será alterada a variável. Utilizada somente para functions e funtion blocks;
- **Variável:** A variável que será forçada;



O force é desativado sempre que o CLP entra em modo PROG, ou quando o usuário sair do modo de supervisão.

Na estrutura das functions e dos funtion blocks o recurso de Force está disponível apenas para variáveis de escopo local. Para alterar parâmetros, o force deverá ser feito no programa do usuário, ou seja, fora da estrutura da function ou do function block.



O "force" pode ser usado com a máquina ou o processo real em funcionamento. Nesse caso, esta facilidade somente deve ser usada por programadores altamente conhecedores da máquina ou processo em questão, uma vez que é possível forçar variáveis booleanas que possam ser responsáveis pela segurança da máquina ou do operador.

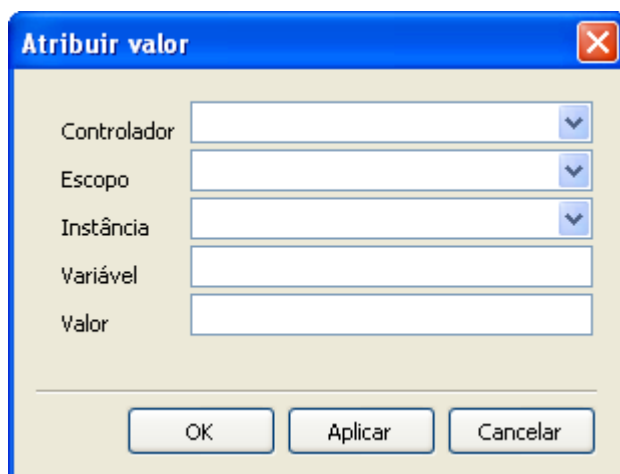
A manipulação de variáveis numéricas é feita através da opção "Atribuir valor...".

Atribuir valor

Esse recurso está disponível durante a supervisão do programa e é utilizado para alterar o valor de variáveis no projeto.

Assim como no Force, podemos abrir a janela Atribuir valor através do ladder ou da variável desejada na janela Supervisão de variáveis, clicando com o botão direito do mouse, selecionando o item "Atribuir valor".

Ao clicar no item "Atribuir valor", a seguinte janela será mostrada:



A janela "Atribuir valor" possui um título azul com um botão de fechar (X) no canto superior direito. O corpo da janela é amarelo e contém cinco campos de entrada: "Controlador", "Escopo" e "Instância" são menus suspensos com setas para baixo; "Variável" e "Valor" são campos de texto simples. Na base da janela, há três botões: "OK", "Aplicar" e "Cancelar".

Descrição dos campos:

- **Controlador:** O controlador que se encontra a variável;
- **Escopo:** O escopo da variável (global ou local);
- **Instância:** Número do bloco que será alterada a variável. Utilizada somente para functions e function blocks;
- **Variável:** A variável que será forçada;
- **Valor:** Valor atribuído à variável;

Descrição dos botões:

- **Ok:** Atribui o novo valor à variável definida e fecha a janela;
- **Aplicar:** Atribui o novo valor à variável, e mantém a janela aberta para próximas alterações;
- **Cancelar:** Cancela a operação, ignorando as configurações feitas.

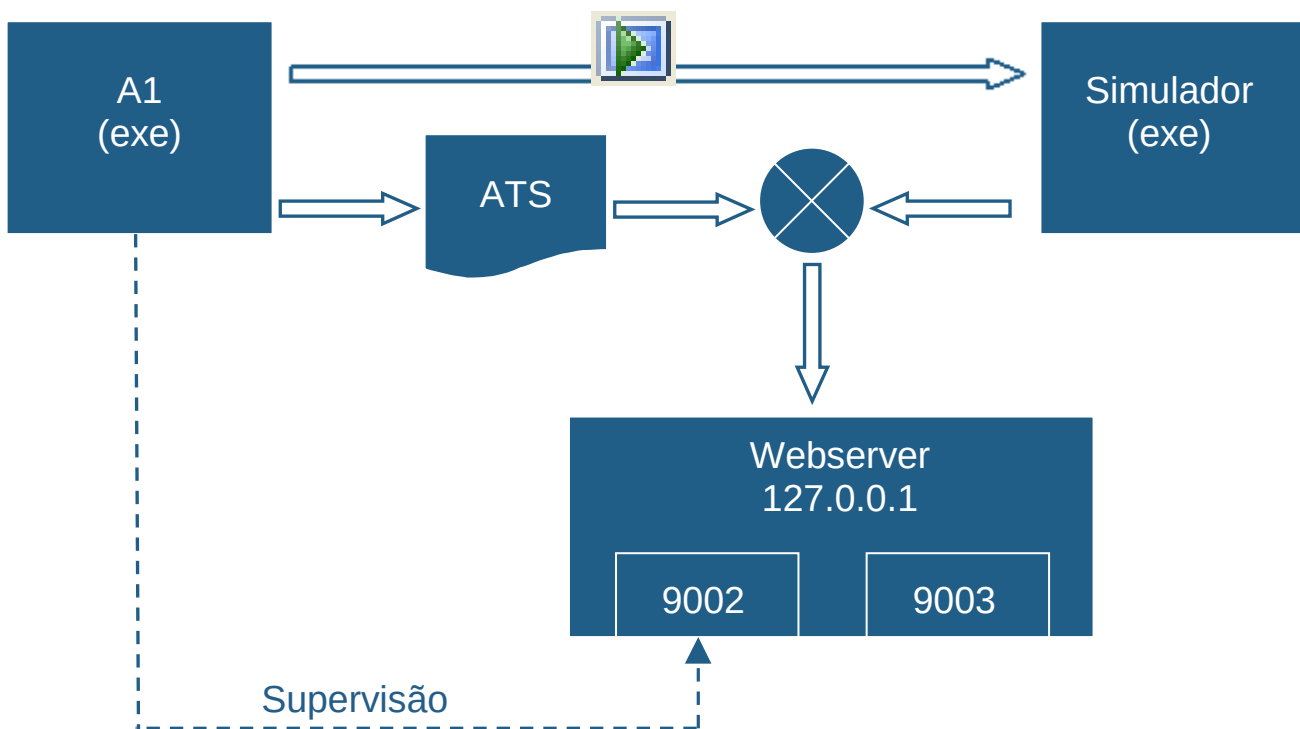
Simulador

O que é o simulador

O simulador do projeto é capaz de reproduzir o comportamento do controlador. Isso permite que o programa possa ser testado sem a necessidade do hardware.

Através de janelas representando as entradas/saídas e supervisão do ladder é possível inserir dados no sistema para testar o comportamento dos equipamentos da máquina que se pretende simular.

No fluxograma abaixo temos o esquema de funcionamento do simulador juntamente com o software Atos A1 Soft.



Após enviar o comando de início de simulação , o simulador inicia um Webserver que funciona no *localhost* do computador.

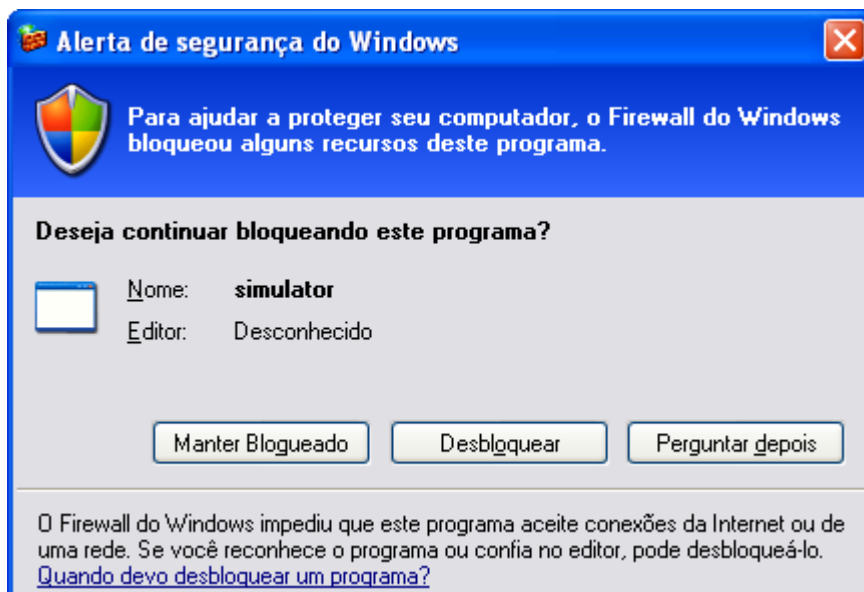
O IP *localhost* de um PC é 127.0.0.1, sendo possível configurar as portas de comunicação através do menu Ferramentas do software Atos A1 Soft. Segue abaixo a configuração default das portas de comunicação:

- RS-232 : porta 9002
- RS-485 : porta 9003



O Atos A1 Soft sempre utiliza a porta configurada como RS-232 para a supervisão do simulador.

Como o simulador é um novo processo a ser executado, o Firewall do Windows envia um Alerta de Segurança, mostrado na imagem a seguir:



Depois de desbloqueado, a mensagem acima não será mais visualizada pelo usuário ao iniciar a simulação.

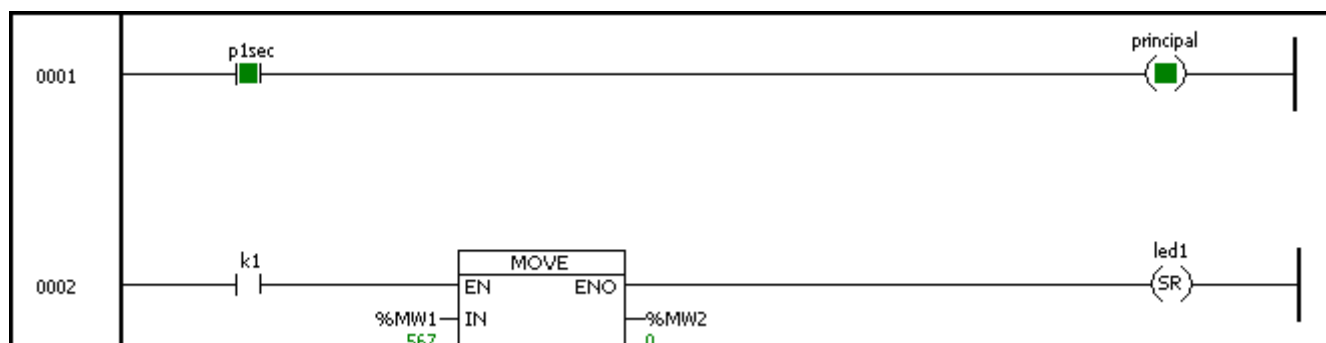
Durante a simulação o processo pode ser verificado no Gerenciador de Tarefas do Windows, mostrado abaixo.



Recursos disponíveis na Simulação

Durante a simulação do projeto os seguintes itens poderão ser simulados:

Ladder



Entradas / Saídas digitais

ED0 (%I1.1.1)	
ED1 (%I1.1.2)	
ED2 (%I1.1.3)	
ED3 (%I1.1.4)	
ED4 (%I1.1.5)	
ED5 (%I1.1.6)	
ED6 (%I1.1.7)	
ED7 (%I1.1.8)	
SD0 (%Q1.1.1)	
SD1 (%Q1.1.2)	
SD2 (%Q1.1.3)	
SD3 (%Q1.1.4)	
SD4 (%Q1.1.5)	
SD5 (%Q1.1.6)	
SD6 (%Q1.1.7)	
SD7 (%Q1.1.8)	

Entradas / Saídas analógicas

EA1 (%IW1.3.1)	5,72V	
EA2 (%IW1.3.2)	0,00V	
EA3 (%IW1.3.3)	3,02V	
EA4 (%IW1.3.4)	7,46V	
SA1 (%QW1.3.1)	3,89 V	
SA2 (%QW1.3.2)	3,89 V	
SA3 (%QW1.3.3)	3,89 V	
SA4 (%QW1.3.4)	3,89 V	

Módulos de temperatura

T1 (%IW9.1)	0,00 °C	
T2 (%IW9.2)	183,50 °C	
T3 (%IW9.3)	322,50 °C	
T4 (%IW9.4)	139,10 °C	
T5 (%IW9.5)	133,20 °C	
T6 (%IW9.6)	106,50 °C	
T7 (%IW9.7)	174,60 °C	
T8 (%IW9.8)	0,00 °C	

No ladder pode-se verificar o seguinte funcionamento:

- As saídas e os contatos NA são marcados com um retângulo colorido, sempre que estiverem ativos, ou seja, TRUE;
- As saídas invertidas e os contatos NF são marcados com um retângulo colorido, sempre que não estiverem ativos, ou seja, FALSE;
- As instruções exibem o conteúdo das variáveis de entrada e saída, imediatamente abaixo das mesmas.

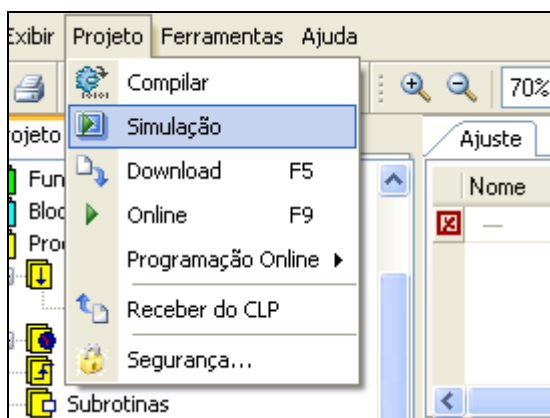
Simulando um projeto

Existem duas maneiras de iniciar a simulação do projeto:

- Através do botão  existente na barra de Comunicação ;

OU

- Acessando o item “Simulação” do menu “Projeto”, conforme mostrado na figura abaixo:



Antes de iniciar a simulação, o software compila o projeto. A simulação é iniciada somente se não existir nenhum erro de compilação.

Durante o processo de simulação o ícone , existente no canto superior direito do software, permanece piscando, indicando a supervisão do simulador pelo Atos A1 Soft;



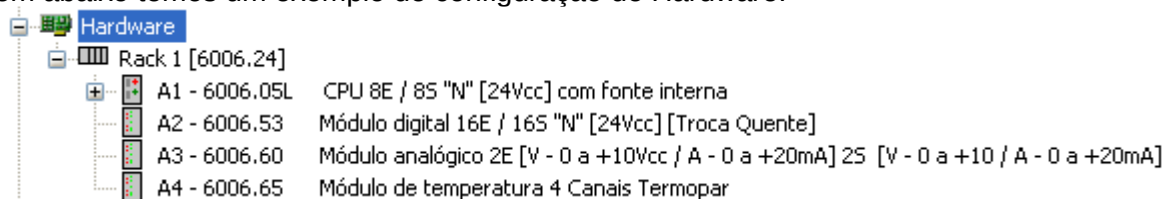
Durante a simulação NÃO é possível editar o programa usuário.

Guia “Simulador de I/O”

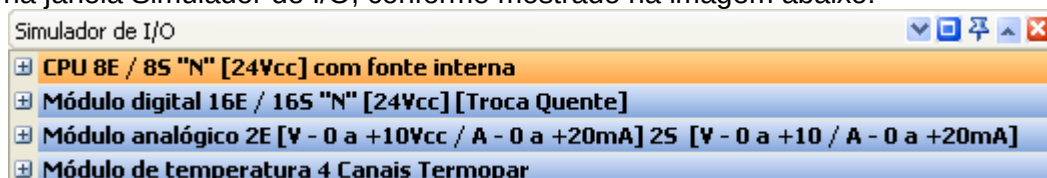
Ao iniciar a simulação, a guia “Simulador de I/O” é habilitada. Esta guia lista todos os módulos declarados na configuração de hardware do projeto, possibilitando o controle dos pontos de entrada e saída do hardware.

A janela de simulação irá mostrar os módulos na sequência em que foram configurados no item “Hardware” do Gerenciador do projeto.

Na imagem abaixo temos um exemplo de configuração de Hardware.



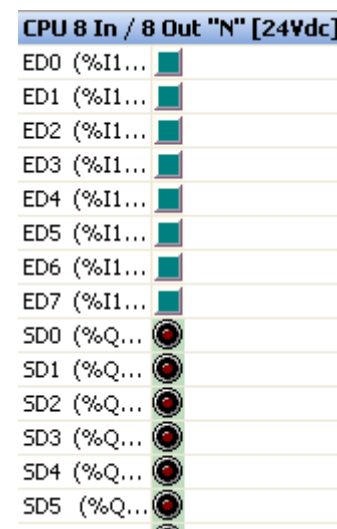
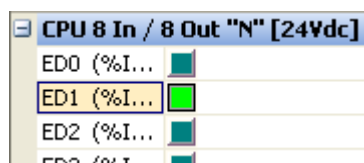
Em seguida, após iniciar a simulação do projeto podemos verificar a sequência em que os módulos são visualizados na janela Simulador de I/O, conforme mostrado na imagem abaixo:



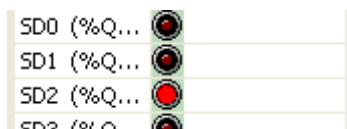
Simulação das entradas / saídas digitais

A figura ao lado mostra a representação da simulação das entradas e saídas da CPU configurada.

Através dessa janela é possível acionar as entradas digitais, clicando nos botões referentes a cada uma, assim que acionada o ícone do botão é aceso para indicar o status da entrada, conforme a figura abaixo:



As saídas digitais são apenas para visualização, caso esta seja acionada pelo programa durante sua execução, o ícone referente a essa saída acionada será alterado, como mostra a figura abaixo:



Simulação das entradas / saídas analógicas

Nos módulos analógicos, as entradas e saídas também serão visualizadas, sendo que, assim como para as digitais, somente as entradas podem ser alteradas.

A figura abaixo mostra as entradas e saídas analógicas durante a simulação do projeto.

Módulo analógico 2E [V - 0 a +10Vcc / A - 0 a +20mA] 2S [V - 0 a +10Vcc]		
EA1 (%I...	0,00V	
EA2 (%I...	0,00V	
SA1 (%Q...	0,00 V	
SA2 (%Q...	0,00 V	

Para alterar a entrada analógica, movimente o cursor sobre a barra, para isso mantenha pressionado o botão esquerdo do mouse, com seu cursor sobre o objeto, conforme mostra a figura abaixo:

- Módulo analógico 2E [V - 0 a +10Vcc / A - 0 a +20mA] 2S [V - 0 a +10Vcc]		
EA1 (%I...	2,61V	
EA2 (%I...	0,00V	
SA1 (%Q...	0,00 V	
SA2 (%Q...	0,00 V	

Simulação dos módulos de temperatura

A simulação dos módulos de temperatura é similar à das entradas analógicas, conforme verificado na imagem abaixo:

Módulo de temperatura 4 Canais Termopar		
T1 (%IW...	0,00 °C	
T2 (%IW...	0,00 °C	
T3 (%IW...	0,00 °C	
T4 (%IW...	0,00 °C	
OVF1 (%I...		
OVF2 (%I...		
OVF3 (%I...		
OVF4 (%I...		

Assim como nas entradas analógicas, a entrada de temperatura é alterada movimentando o cursor sobre a barra, para isso o botão esquerdo do mouse deve ser mantido pressionado com seu cursor sobre o objeto.

Desta forma cada módulo inserido no projeto, tem um item correspondente na janela Simulador de I/O, onde seus parâmetros são alterados de acordo com o tipo de módulo.

Restrições do Simulador

O simulador possui um conjunto de instruções que são fisicamente dependentes do hardware. Estas instruções não podem ser simuladas, quando utilizadas no projeto.

Instruções que não simulam no controlador Atos MPC4004BF

- ANGLE COUNTER;
- ANGLE SIMULATE;
- CPU COUNTER;
- HS COUNTER;
- STEP MOTOR PULSE;
- STEP MOTOR FASE;

Instruções que não simulam no controlador Atos MPC6006L

- ANGLE COUNTER;
- ANGLE SIMULATE;
- CPU COUNTER;
- HS COUNTER;
- PWM;
- PTO;
- STEP MOTOR PULSE;
- STEP MOTOR FASE;

Para um projeto que possuir alguma dessas instruções, ao iniciar a simulação será informado pela janela de mensagens do Atos A1 Soft da restrição:

Não é possível simular o FB [nome do bloco funcional]

Restrições de comunicação

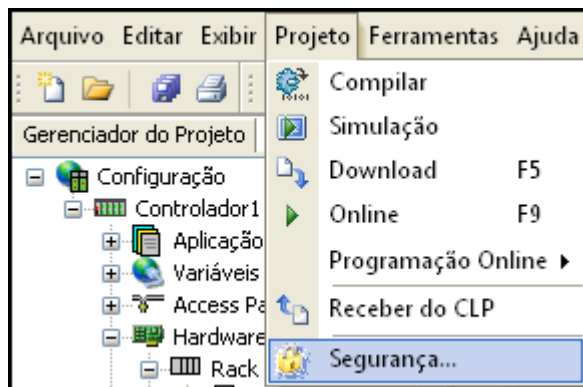
Recursos disponíveis na comunicação durante a SIMULAÇÃO						
	APR03 Escravo	APR03 Mestre	Modbus Escravo	Mobus Mestre	Instrução SPRINT	Instrução RPRINT
RS-232	SIM	NÃO	NÃO	NÃO	NÃO	NÃO
RS-485	SIM	NÃO	SIM	NÃO	NÃO	NÃO

Nas situações acima, a simulação do restante do projeto é feita normalmente, porém sem a atualização das instruções mencionadas na janela de mensagem.

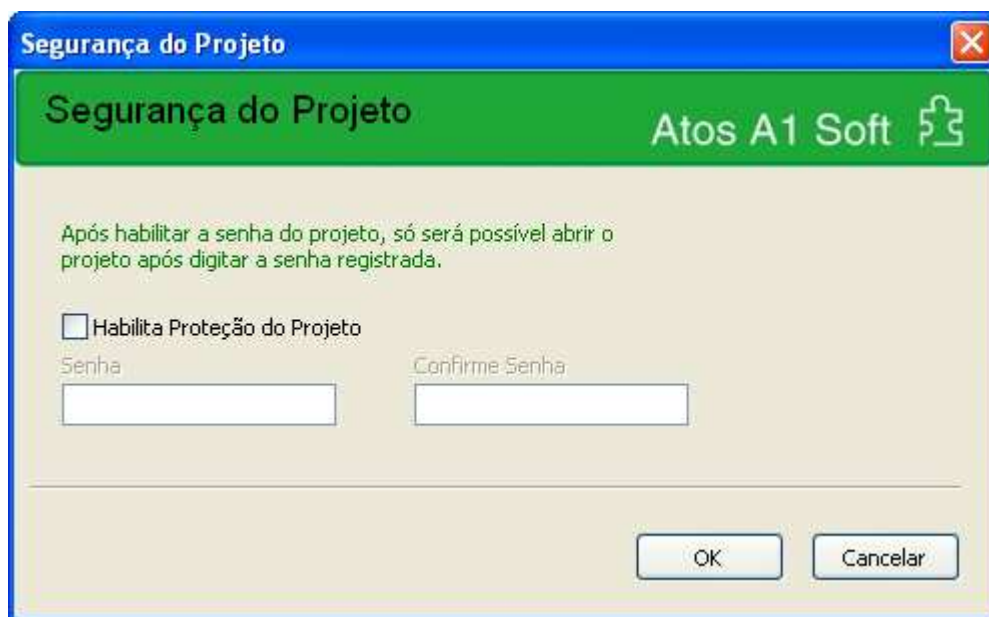
Segurança do Projeto

Esse recurso permite que o usuário proteja, através de uma senha, as informações do projeto.

Para habilitar este recurso, no menu Projeto, selecione o item "Segurança", conforme mostra a figura abaixo:



Após selecionar o item "Segurança" a seguinte janela será mostrada:



Nesta janela ao habilitar a proteção do projeto, o campo Password será desbloqueado, permitindo que se configure uma senha.

Confirme senha no campo abaixo e clique em OK para salvar as alterações.

Abrindo projetos protegidos por senha.

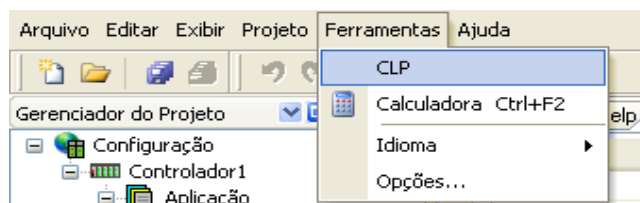
Ao abrir um projeto com proteção, informe a senha previamente definida na janela mostrada ao lado.

Ferramentas do CLP

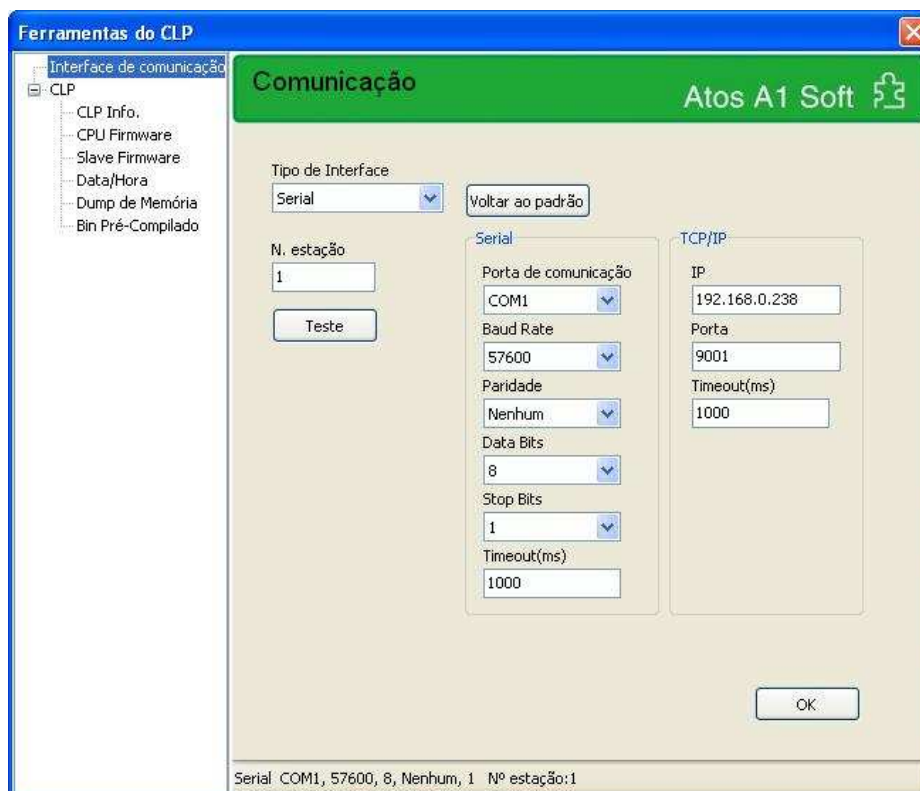
Como acessar?

O item Ferramentas do CLP é utilizado para realizar diversas configurações no CLP.

Para acessar a janela de configuração, no menu Ferramentas, selecione a opção "CLP", conforme mostra a figura abaixo:



Ao clicar na opção CLP, o usuário pode verificar a janela "Ferramenta do CLP" conforme mostrado abaixo:

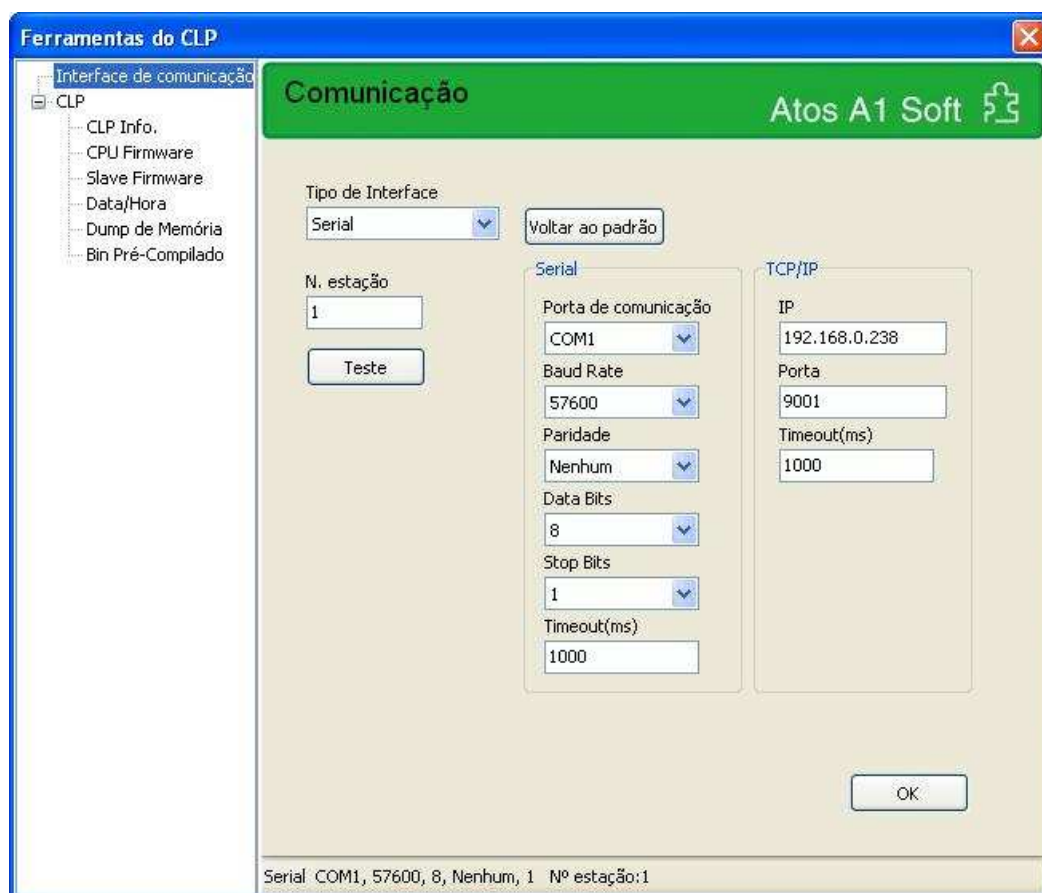


Nesta janela são feitas configurações com relação a:

- Comunicação:
 - Interface de Comunicação;
- CLP:
 - Informações do CLP;
 - Firmware;
 - Data/Hora;
 - Dump de Memória;
 - Bin Pré-Compilado;

Interface de Comunicação

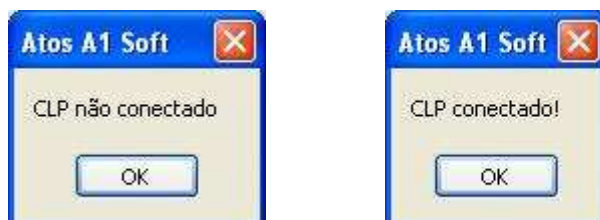
A opção "Interface de Comunicação" realiza testes de comunicação entre o aplicativo Atos A1 Soft e o CLP.



De acordo com o tipo de interface escolhida (Serial ou TCP/IP), define-se alguns parâmetros:

- **Serial:** Porta de comunicação, Baud Rate, Paridade, Data Bits, Stop Bits e Timeout.
- **TCP/IP:** IP, Porta e Timeout.

Clicando no botão "Teste" podemos verificar se o CLP está conectado ao aplicativo, recebendo uma mensagem de resposta, conforme mostrada abaixo:

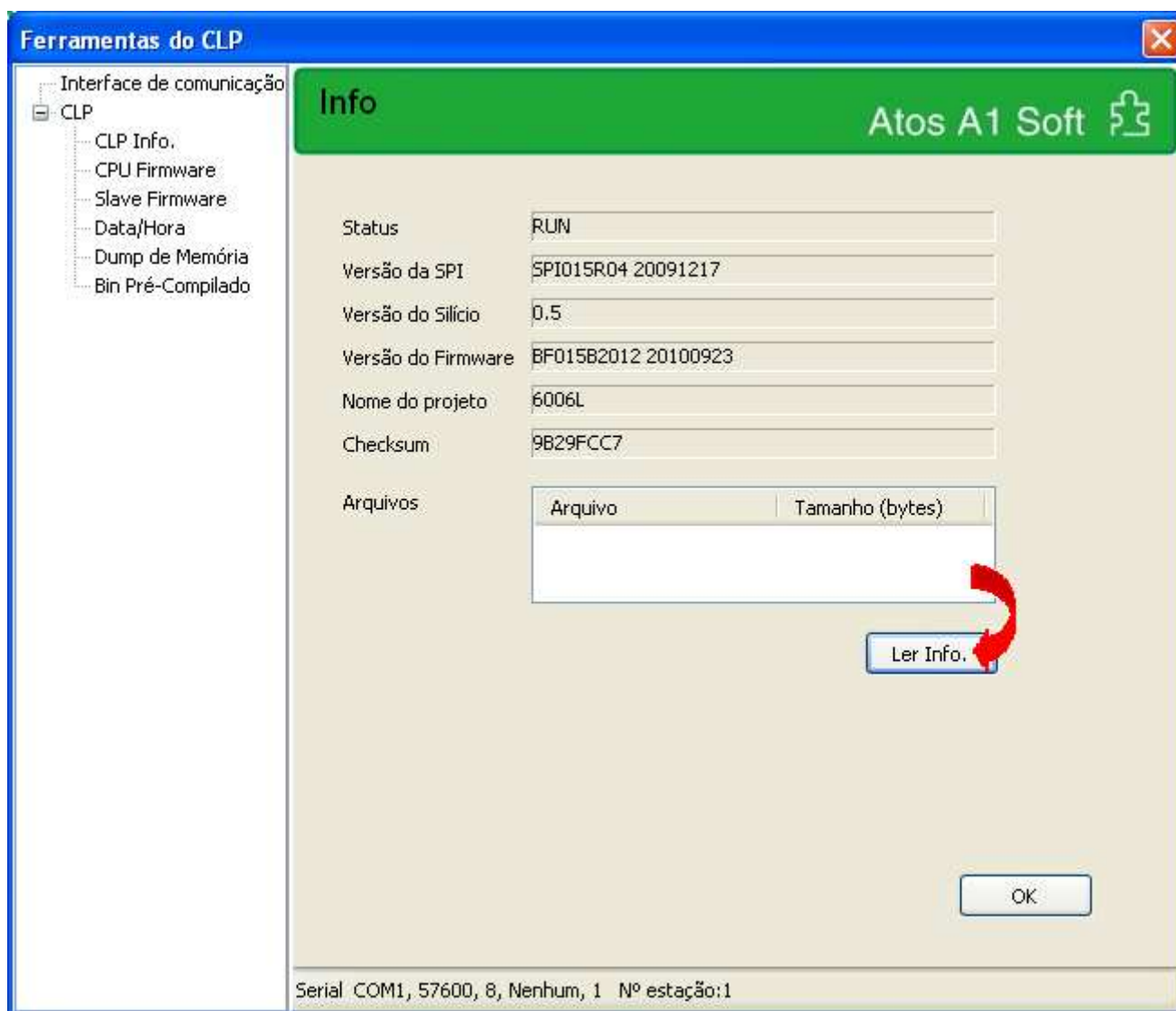


Verifique se as configurações de comunicação do Atos A1 Soft estão iguais as do CLP.

CLP Info.

A opção "CLP Info." disponibiliza ao usuário a visualização das informações do CLP conectado ao aplicativo.

Para visualizar essas informações clique no botão "Ler Info", conforme mostra a figura abaixo:

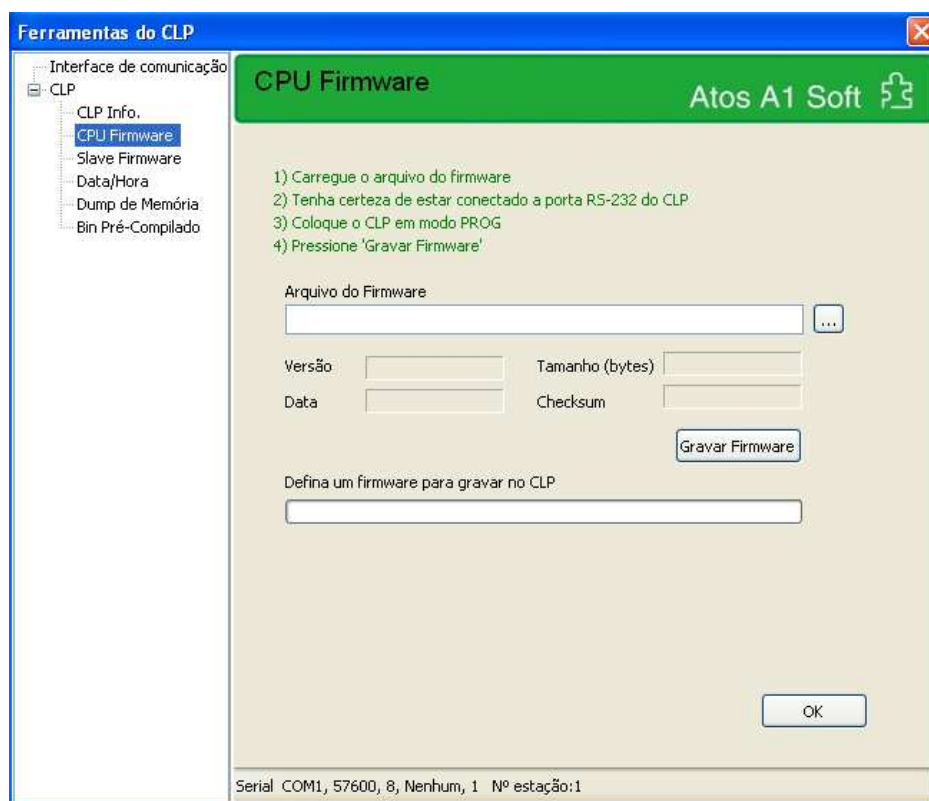


Descrição dos campos:

- **Status:** Modo do CLP (RUN/PROG);
- **Versão SPI:** Memória de inicialização do firmware;
- **Versão do silício:** Versão do processador da CPU (Ver mais);
- **Versão do Firmware:** Versão do firmware gravado na CPU;
- **Nome do Projeto:** Nome do projeto gravado no CLP;
- **Checksum:** Informação de comparação entre Atos A1 Soft e CLP. Acessado durante supervisão e programação online.
- **Appends:** Arquivos armazenados na CPU (exemplo: arquivo .ats do projeto).

Firmware

A opção "Firmware" permite realizar a gravação de firmware no CLP.



Para gravação do firmware é necessário observar as condições iniciais:

- O CLP deve estar conectado a porta RS232 e deve estar no modo PROG.

Depois de verificadas as condições iniciais, clique no botão , selecione um arquivo de firmware, como mostra a janela ao lado, e clique no botão .

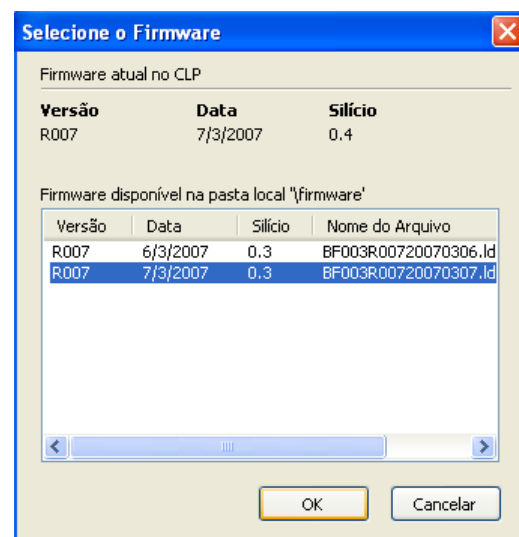
Aguarde até que uma mensagem de gravação concluída seja mostrada, informando que o firmware foi gravado com sucesso no CLP.



O processo de gravação do firmware pode demorar um certo tempo, dependendo do PC utilizado (mais informações das configurações mínimas para rodar o Atos A1 Soft no item Sobre).



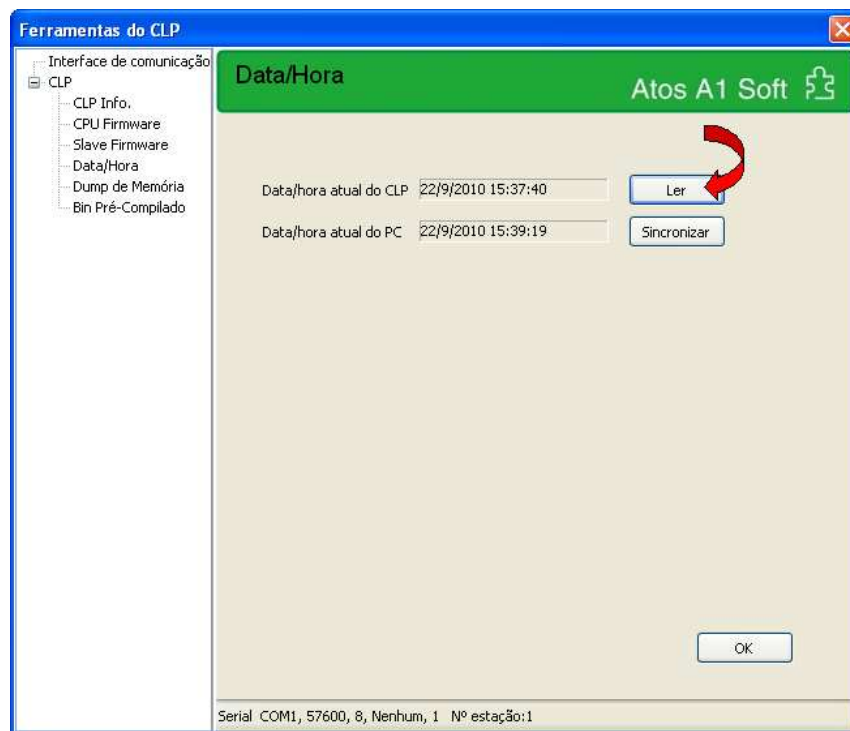
Sempre que gravado um novo firmware é necessário reenviar o projeto para o CLP.

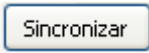


Data/Hora

A opção "Data/Hora" permite ao usuário verificar e ajustar a DATA e a HORA do CLP com a configurada no PC.

Para isso verifique primeiro os dados do CLP através do botão , conforme mostra a figura abaixo:



Na necessidade de ajustar o relógio do CLP, clique no botão .

A seguinte mensagem é mostrada no fim da sincronização.

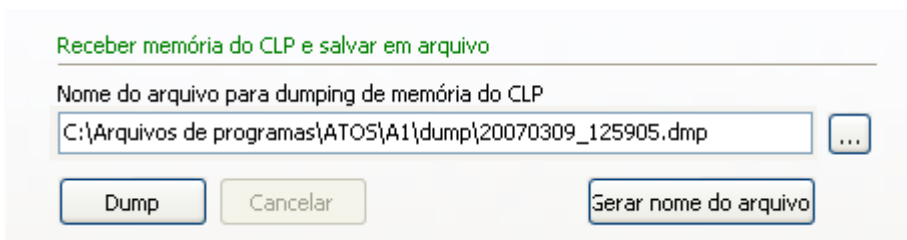


Dump de Memória

A opção "Dump de Memória" permite armazenar ou enviar, um arquivo binário da memória de usuário do CLP.

Para **receber memória do CLP e salvar**, os seguintes passos devem ser seguidos:

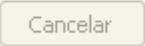
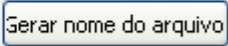
- Clique no botão ;
- Clique no botão , para receber memória do CLP e armazená-la no arquivo selecionado.



Receber memória do CLP e salvar em arquivo

Nome do arquivo para dumping de memória do CLP

C:\Arquivos de programas\ATOS\A1\dump\20070309_125905.dmp

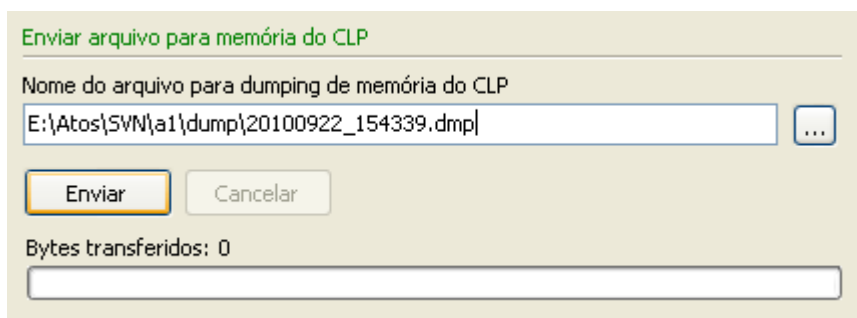
  



O arquivo Dump é armazenado na pasta A1\dump.

Para **enviar arquivo para memória do CLP**, os seguintes passos devem ser seguidos:

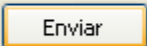
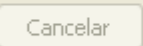
- Através do botão , selecione um arquivo de memória Dump armazenado;
- Clique no botão . A memória do arquivo será transferida para o CLP.



Enviar arquivo para memória do CLP

Nome do arquivo para dumping de memória do CLP

E:\Atos\SVN\A1\dump\20100922_154339.dmp

Bytes transferidos: 0

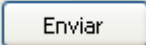


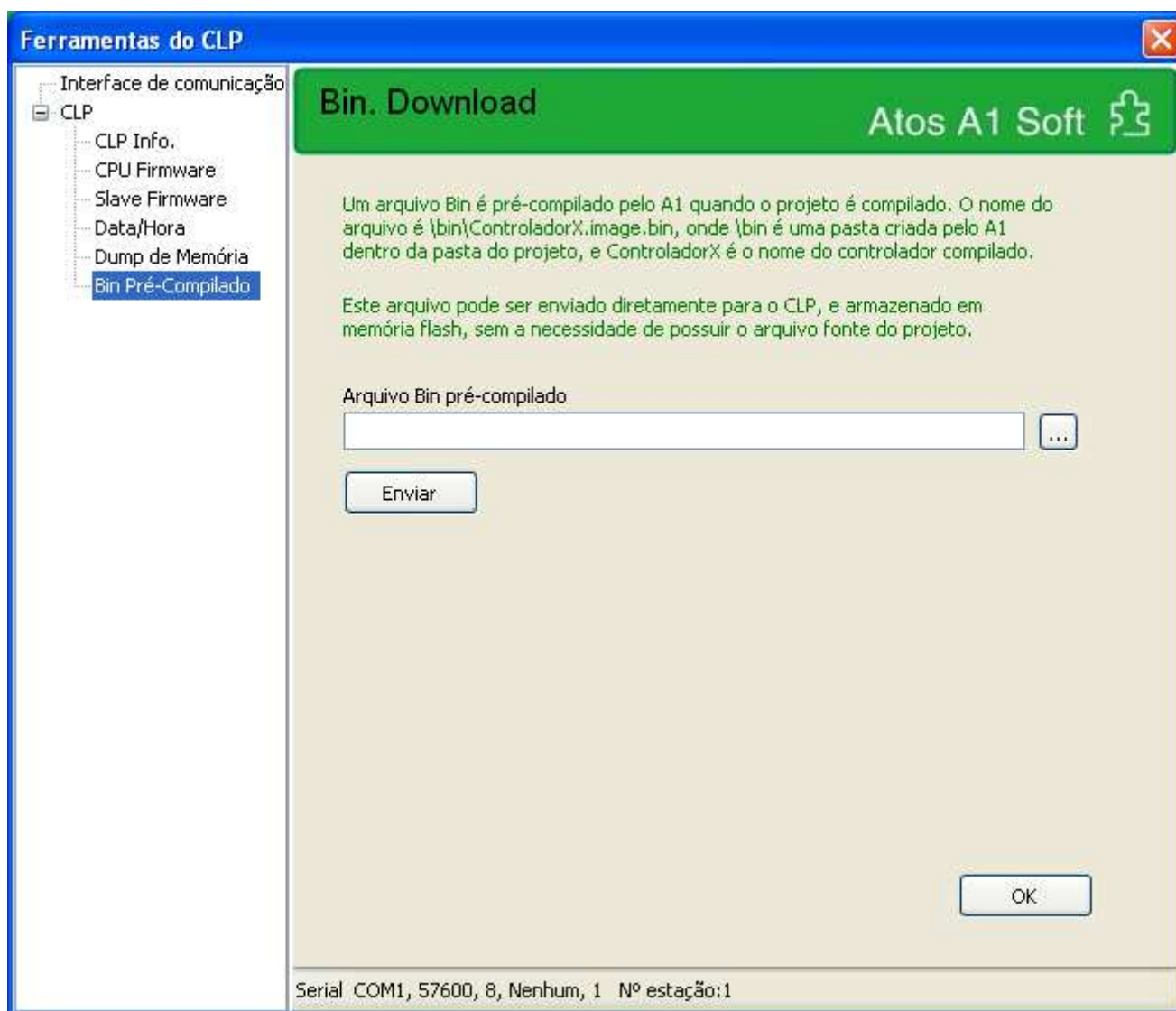
Bin Pré-Compilado

O arquivo pré-compilado é armazenado no CLP (memória flash) sem a necessidade de envio do arquivo fonte do projeto, sua principal aplicação é fornecer um projeto, sem mostrar seu código fonte.

O arquivo Bin é pré-compilado pelo Atos A1 Soft quando o projeto é compilado/enviado, ele é armazenado na pasta bin, existente dentro da pasta do projeto. O arquivo é identificado pelo nome do controlador compilado.

Por exemplo: Temos o arquivo .bin\controlador1.image.bin, sendo que controlador1 é o nome do controlador compilado).

Para enviar um arquivo BIN clique no botão  e selecione o arquivo desejado. Clique no botão  para gravar o projeto no CLP.



Troca global de variáveis

A troca global de variáveis é utilizada para substituir uma variável em vários pontos do projeto, ao mesmo tempo.

A figura a seguir mostra a janela de troca global de variáveis, com as descrições dos campos em sequência.

Descrição dos campos

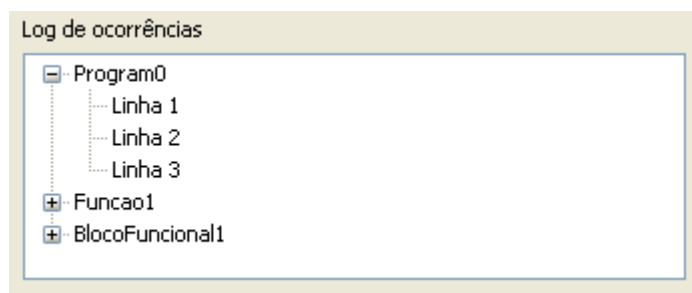
- **Variável Atual:** Variável que deseja substituir;
- **Nova Variável:** O novo nome desejado para a variável;

No escopo, o usuário pode optar por mudar as variáveis globais ou as locais, conforme mostrado na figura abaixo.



As opções de escopo só são habilitadas quando uma janela de POU estiver aberta no projeto.

O **Log de ocorrências** registra em quais linhas a variável foi substituída.



A opção **encontrar palavra completa** define se o conteúdo do campo Variável atual é o nome completo de uma variável, ou um texto que deve ser substituído em todas as ocorrências encontradas.

Exemplo de troca global com a opção encontrar palavra completa desmarcada:

- Variável atual: 'TESTE'
- Nova variável: 'MOTOR'

Ocorrências encontradas:

- PR_TESTE_TANQUE1
- SP_TEMP_TESTE_BOMBA1

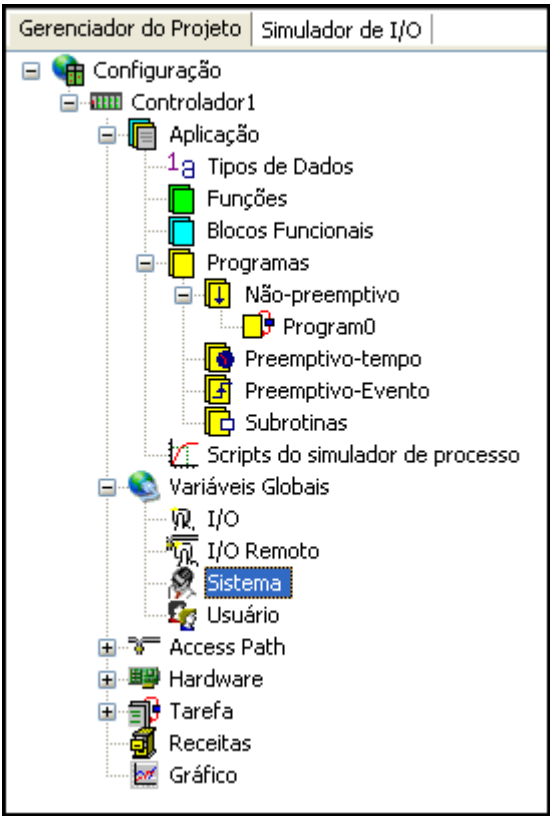
Após substituição:

- PR_**MOTOR**_TANQUE1
- SP_TEMP_**MOTOR**_BOMBA1

Gerenciador de Variáveis Globais

Na guia Variáveis Globais é possível visualizar e configurar todas as variáveis de escopo global, ou seja, podem ser acessadas por elementos de qualquer programa do projeto.

Para acessar essa guia, na janela “Gerenciador do Projeto”, dê um duplo-clique em uma das três opções disponíveis no item “Variáveis Globais”, conforme mostra a figura a seguir:



Após isso a guia Variáveis Globais, mostrada abaixo, estará disponível para o usuário.

Variáveis Globais		Help					
Grupo		Região do Tipo de Dado					
USUÁRIO		BOOL					
Endereço	Nome	Tipo de dado	Atributo	Valor Inicial	Descrição	Endereço físico	Posição Modbus
%MX1		BOOL	NÃO RETEN...			1800	7001
%MX2		BOOL	NÃO RETEN...			1801	7002
%MX3		BOOL	NÃO RETEN...			1802	7003
%MX4		BOOL	NÃO RETEN...			1803	7004
%MX5		BOOL	NÃO RETEN...			1804	7005
%MX6		BOOL	NÃO RETEN...			1805	7006
%MX7		BOOL	NÃO RETEN...			1806	7007
%MX8		BOOL	NÃO RETEN...			1807	7008

Descrição dos campos:

- **Grupo:** Define o grupo que pertence a variável;
 - **I/O:** Grupo das variáveis associadas ao Hardware configurado no projeto;
 - **Sistema:** Grupo de variáveis com funcionalidades definidas pelo sistema, que podem ser utilizadas mas não modificadas;
 - **Usuário:** Grupo de variáveis livres para utilização nos programas, podendo ser modificadas.
- **Região do Tipo de Dado:** Agrupa os [Tipos de dados](#) de forma a manter os que possuem características semelhantes dentro de uma mesma região.
- **Endereço:** Endereço associado à variável, segundo a norma IEC61131.
- **Nome:** Nome atribuído ao endereço da variável. Esse nome pode ser usado diretamente no programa, não sendo necessário declarar o endereço;
- **Tipo de dado:** Define qual o tipo de dado correspondente a cada variável dentro de cada região;
- **Atributo:** atribui a variável como sendo:
 - **Constante:** Variável permanece sempre com valor constante que deve ser inserido no campo Valor Inicial. Nada pode alterar o valor desta variável, tanto na programação em ladder quanto via comunicação ou supervisão do CLP;
 - **Retentiva:** Variável retém sempre o último valor armazenado quando o CLP for reinicializado;
 - **Não Retentiva:** Variável não retém o último valor armazenado quando o CLP for reinicializado, porém carrega o valor default de acordo com o tipo de dado seguindo a tabela da norma IEC 61131;
- **Valor Inicial:** Define o valor que a variável será inicializada;
- **Descrição:** Não é obrigatório, um campo para descrever a variável em questão;
- **Endereço físico:** Endereço utilizado na comunicação de dados, em APR03;
- **Posição Modbus:** Endereço utilizado na comunicação de dados, em ModBus.

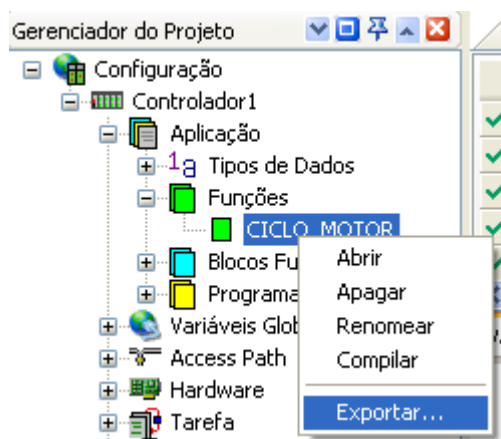
Biblioteca de usuário

Exportando Funções, Blocos Funcionais e Tipos de dados

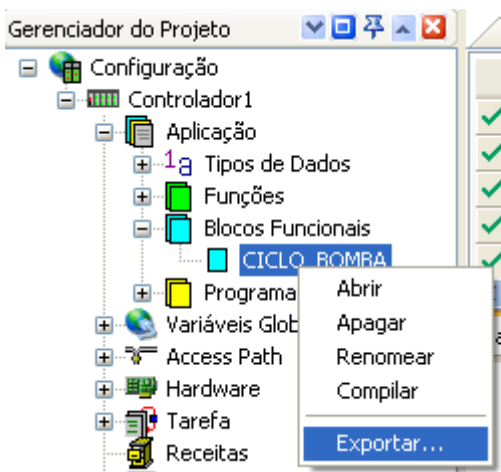
Para exportar uma POU ou um tipo de dado da aplicação do Atos A1 Soft, selecione com o cursor do mouse o item desejado na área do Gerenciador do projeto.

Clique com o botão direito do mouse para exibir o menu pop-up e selecione a opção "Exportar...", conforme mostrado nas figuras abaixo.

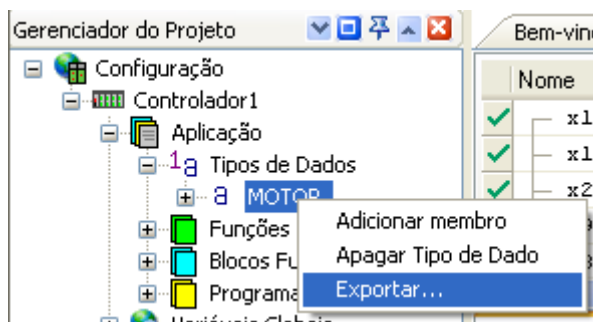
- **Exportando Funções:**



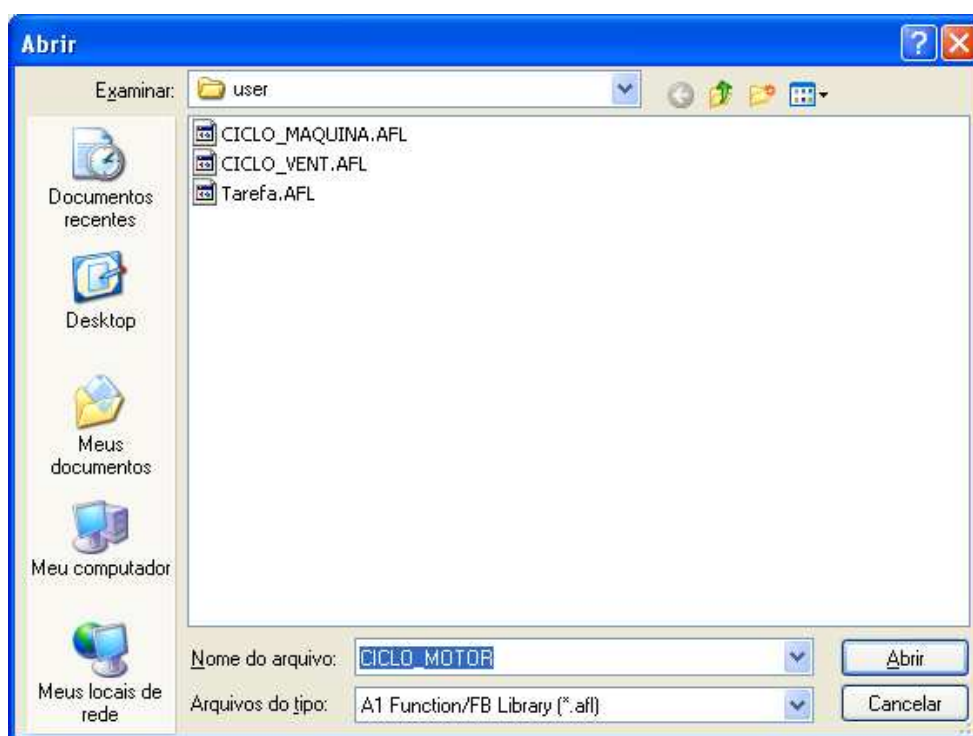
- **Exportando Blocos Funcionais:**



- **Exportando Tipo de dado:**



Ao seleccionar opção "Exportar...", a janela abaixo será mostrada para a definição do local onde o POU ou o tipo de dado será salvo.



A pasta padrão onde os arquivos são salvos é: **"../A1/lib/user"**.



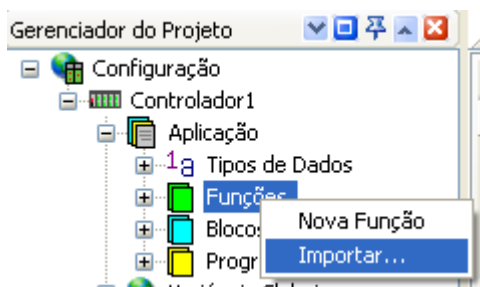
As funções e blocos funcionais serão salvos com a extensão **.afl** (**A1 Function/Function Block Library**) e os tipos de dados como **.adl** (**A1 Datat Type Library**).

Importando Funções, Blocos Funcionais e Tipos de dados

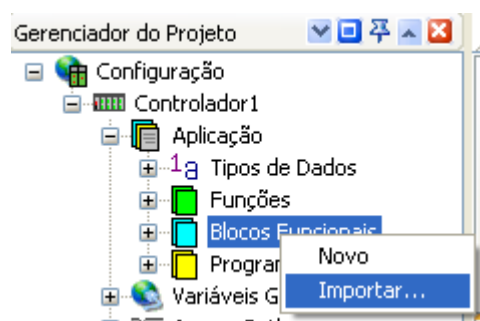
Para importar uma POU ou um tipo de dado da aplicação do Atos A1 Soft, selecione com o cursor do mouse a opção que se deseja importar na área do Gerenciador do projeto.

Clique com o botão direito do mouse para exibir o menu pop-up e selecione a opção "Importar...", conforme mostrado nas figuras abaixo.

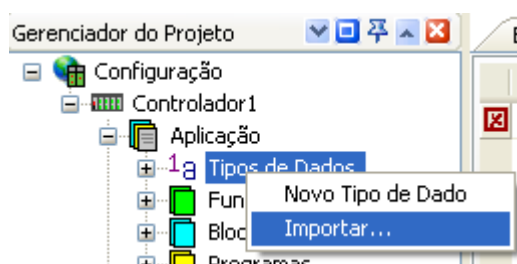
- **Importando Funções:**



- **Importando Blocos Funcionais:**



- **Importando Tipos de dado:**



Depois de clicar na opção "Importar...", será visualizada a janela para selecionar a POU ou tipo de dado que se deseja importar.

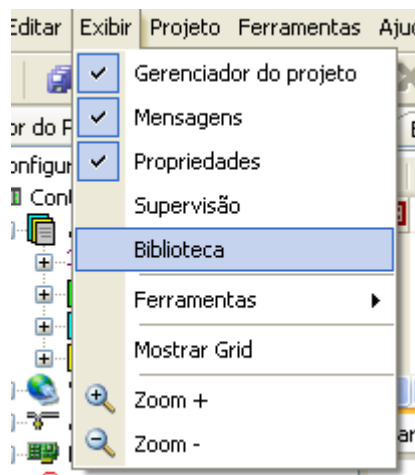


Apesar de possuírem a mesma extensão uma função não poderá ser importada para o item Blocos Funcionais, da mesma forma que um bloco funcional não poderá ser importado para o item Funções, mostrando nessas situações uma mensagem de erro.

Visualizando biblioteca de usuário

Para visualizar a biblioteca do projeto é necessário que a opção "Biblioteca" esteja habilitada.

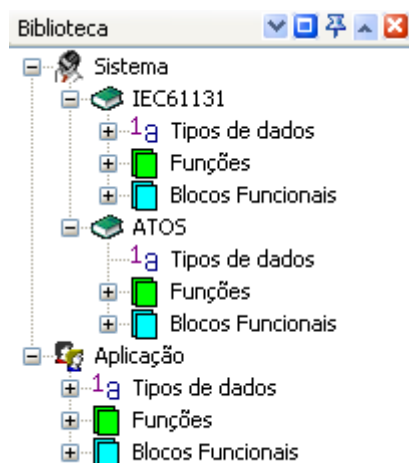
Para isso selecione no menu Exibir a opção "Biblioteca", conforme mostra a figura abaixo:



Nesta situação a guia "Biblioteca" será visualizada, conforme mostrado abaixo.

Esta guia contém a relação de Tipos de dados, Funções e Blocos Funcionais referentes a:

- **Sistema:** IEC61131 e Atos;
- **Aplicação:** Relação de POU's e tipos de dados criados e/ou importados pelo usuário no projeto.

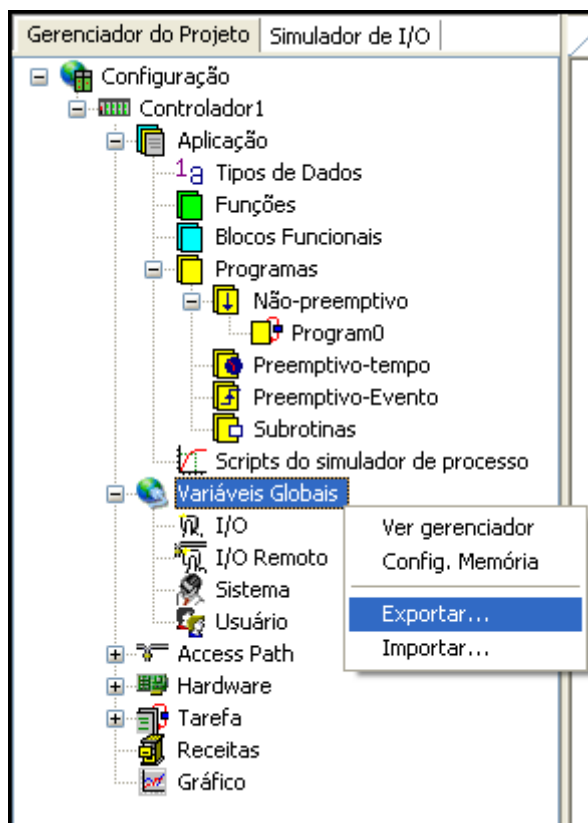


Importação e Exportação de variáveis para o Excel

Exportando variáveis para o Excel

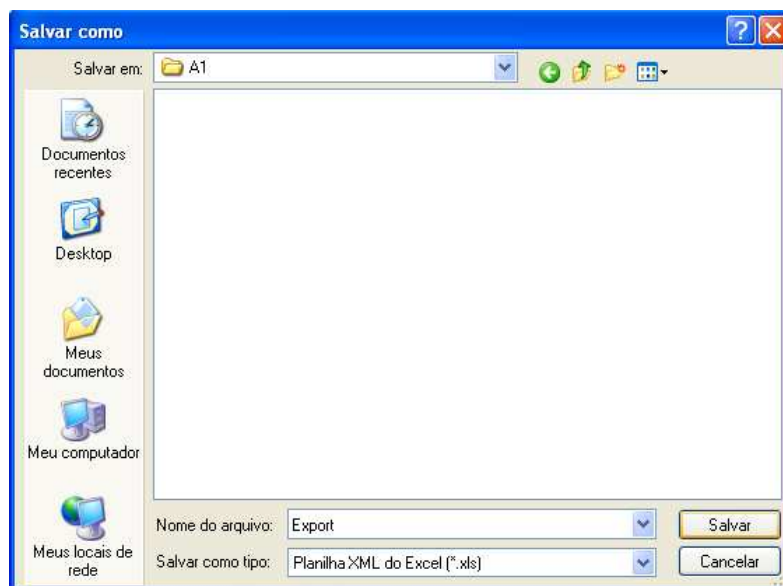
Para exportar variáveis para o Excel o seguinte procedimento deve ser seguido:

Na guia “Gerenciador do projeto” clique com o botão direito do mouse em “Variáveis globais”, selecione o item “Exportar...”, conforme a figura abaixo.



Ao selecionar a opção “Exportar”, a janela abaixo será exibida.

Indique o caminho onde o arquivo será salvo.

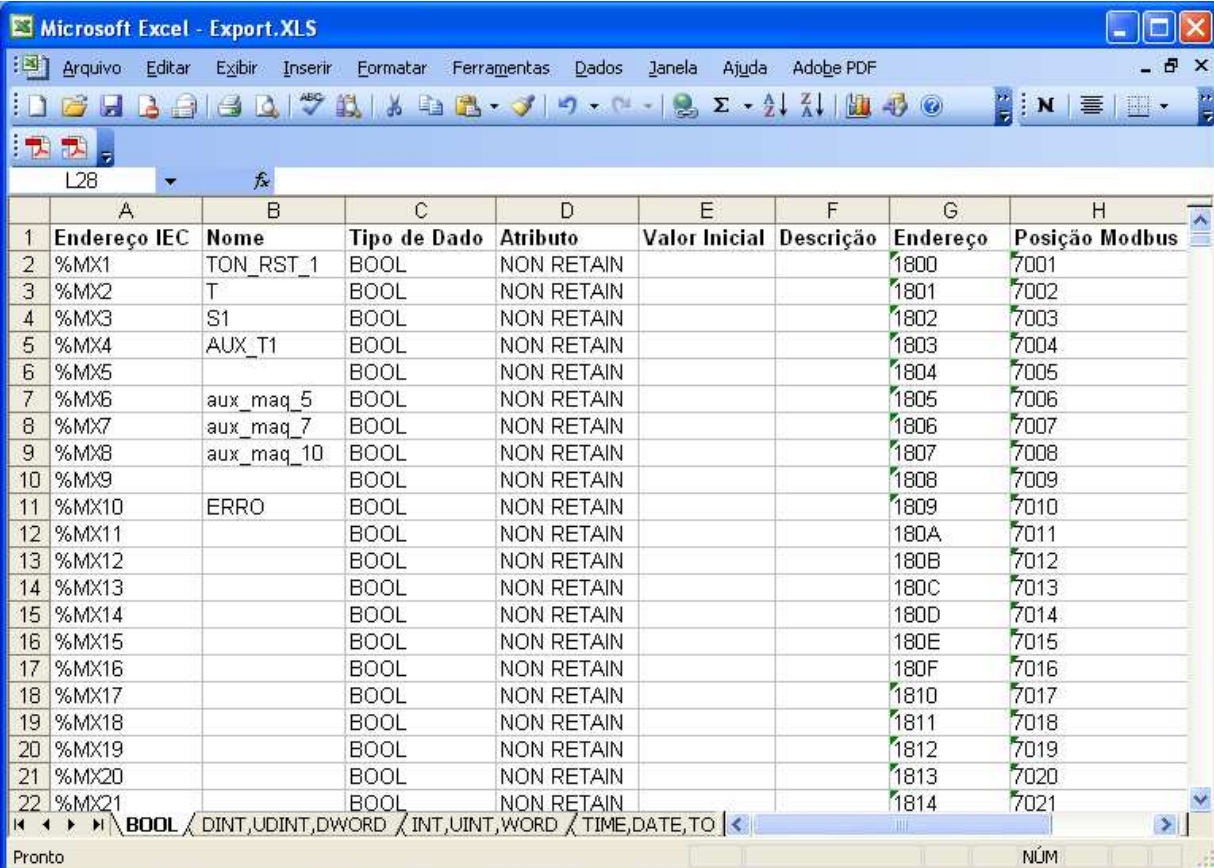


“O arquivo gerado pela exportação possui a extensão “.xls”, não sendo possível alterar essa extensão.



O arquivo será visualizado em planilhas do Excel, as variáveis estão separadas por região de tipo de dado, sendo cada região em uma aba.

Exemplo de arquivo exportado:



The screenshot shows a Microsoft Excel window titled "Microsoft Excel - Export.XLS". The window contains a table with 8 columns: A (Endereço IEC), B (Nome), C (Tipo de Dado), D (Atributo), E (Valor Inicial), F (Descrição), G (Endereço), and H (Posição Modbus). The table lists 21 rows of data, starting from row 1 to row 22. The data includes IEC addresses like %MX1, %MX2, etc., names like TON_RST_1, T, S1, AUX_T1, aux_maq_5, aux_maq_7, aux_maq_10, and ERRO, data types like BOOL, and attributes like NON RETAIN. The Modbus positions range from 7001 to 7021. The status bar at the bottom shows "Pronto" and "NÚM".

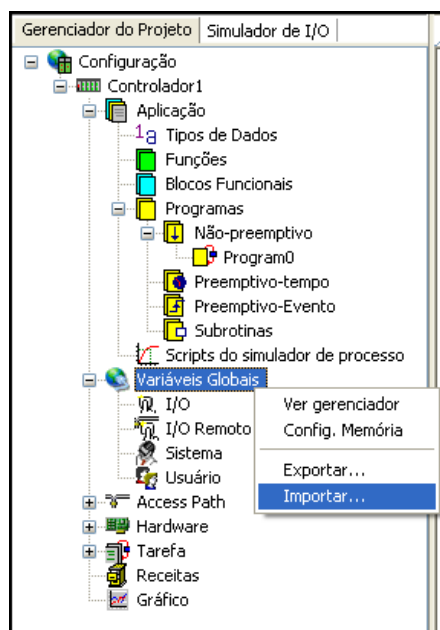
	A	B	C	D	E	F	G	H
	Endereço IEC	Nome	Tipo de Dado	Atributo	Valor Inicial	Descrição	Endereço	Posição Modbus
1	%MX1	TON_RST_1	BOOL	NON RETAIN			1800	7001
2	%MX2	T	BOOL	NON RETAIN			1801	7002
3	%MX3	S1	BOOL	NON RETAIN			1802	7003
4	%MX4	AUX_T1	BOOL	NON RETAIN			1803	7004
5	%MX5		BOOL	NON RETAIN			1804	7005
6	%MX6	aux_maq_5	BOOL	NON RETAIN			1805	7006
7	%MX7	aux_maq_7	BOOL	NON RETAIN			1806	7007
8	%MX8	aux_maq_10	BOOL	NON RETAIN			1807	7008
9	%MX9		BOOL	NON RETAIN			1808	7009
10	%MX10	ERRO	BOOL	NON RETAIN			1809	7010
11	%MX11		BOOL	NON RETAIN			180A	7011
12	%MX12		BOOL	NON RETAIN			180B	7012
13	%MX13		BOOL	NON RETAIN			180C	7013
14	%MX14		BOOL	NON RETAIN			180D	7014
15	%MX15		BOOL	NON RETAIN			180E	7015
16	%MX16		BOOL	NON RETAIN			180F	7016
17	%MX17		BOOL	NON RETAIN			1810	7017
18	%MX18		BOOL	NON RETAIN			1811	7018
19	%MX19		BOOL	NON RETAIN			1812	7019
20	%MX20		BOOL	NON RETAIN			1813	7020
21	%MX21		BOOL	NON RETAIN			1814	7021

Importando variáveis do Excel

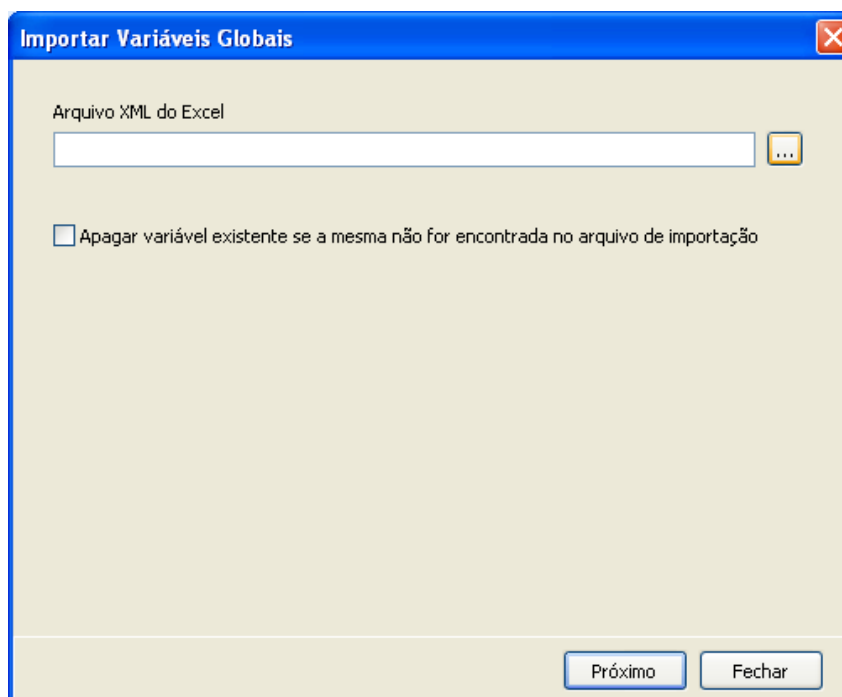
As variáveis a serem importadas devem estar declaradas em um arquivo .xls. Para isso deve ser usado o recurso exportar variáveis, utilizando o arquivo exportado como um template.

Para importar variáveis, o seguinte procedimento deve ser seguido:

Na guia “Gerenciador do projeto” clique com o botão direito do mouse em “Variáveis globais” e selecione o item “Importar...”, conforme a figura abaixo.



A janela abaixo será exibida. Indique o caminho do arquivo ou clique no ícone  para selecionar um arquivo. Clique em “Próximo” para continuar.





Com o checkbox selecionado, as variáveis já declaradas no Atos A1 Soft que não constarem no arquivo de importação serão apagadas.

Selecione quais regiões de variáveis deseja importar. Clique em “Próximo” para continuar.

Planilha	Descrição
☒ BOOL	Região de usuário
☒ DINT, UDINT, DWORD	Região de usuário
☒ INT, UINT, WORD	Região de usuário
☐ TIME, DATE, TOD, DT	Região de usuário
☐ REAL	Região de usuário
☐ DATA_BLOCK	Região de usuário
☐ Slot A1	Região de I/O
☐ Slot A2	Região de I/O
☐ Slot A3	Região de I/O
☐ Slot A4	Região de I/O

Selecionar Tudo Desmarcar Tudo

Anterior **Próximo** Fechar

A janela abaixo será exibida durante o processo de importação.

Importar Logs...

Atualizando variável %MX2
Atualizando variável %MX3
Atualizando variável %MX4
Atualizando variável %MX6
Atualizando variável %MX7
Atualizando variável %MX8
Atualizando variável %MX10

Importando planilha DINT, UDINT, DWORD

Atualizando variável %MD1
Atualizando variável %MD2
Atualizando variável %MD3
Atualizando variável %MD4
Atualizando variável %MD5
Atualizando variável %MD6

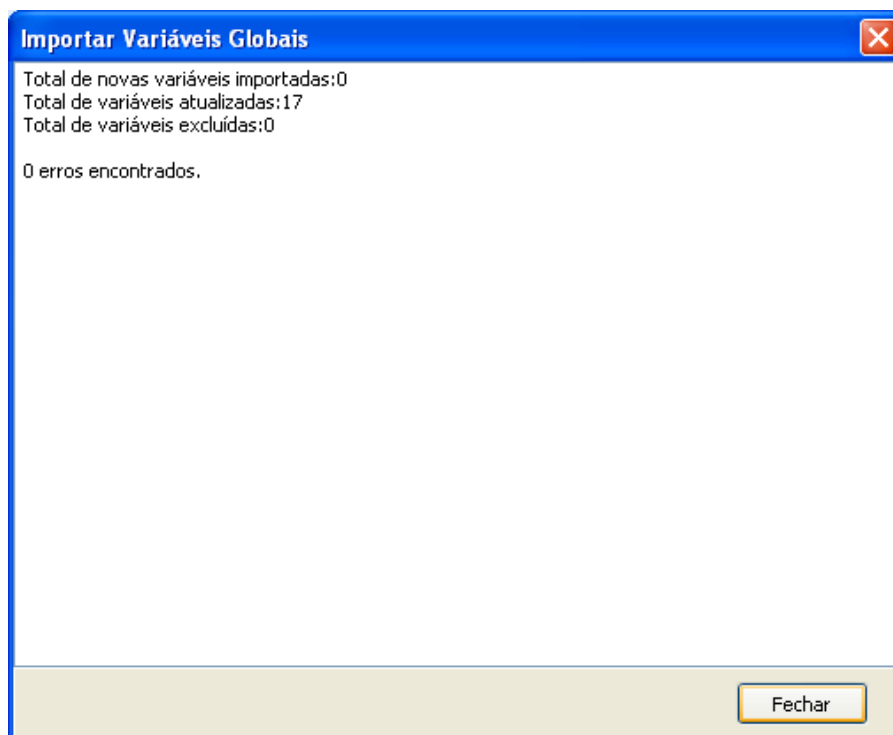
Importando planilha INT, UINT, WORD

Atualizando variável %MW1
Atualizando variável %MW2
Atualizando variável %MW3

Anterior **Próximo** Fechar

Ao clicar em “Próximo” a janela abaixo será exibida com o resultado da importação.

Clique em “Fechar” para finalizar a operação.



Durante o processo de importação das variáveis, é possível que existam erros cujos sintomas mais comuns estão listados abaixo:

- Variáveis com nomes repetidos;
- Variáveis sem o endereço IEC (%MX1, %MW1);
- Tipo de dado errado;
- Endereço IEC que não comece com “%”;
- Colunas invertidas, exemplo: coluna 1 “Endereço” no lugar da coluna 3 “Tipo de dado”;
- Variáveis que o nome comece com números ou caracteres diferentes de letras ou “_”;
- Campo "Nome" com uma quantidade de caracteres maior que o Atos A1 Soft suporta (Nome = 64 caracteres)

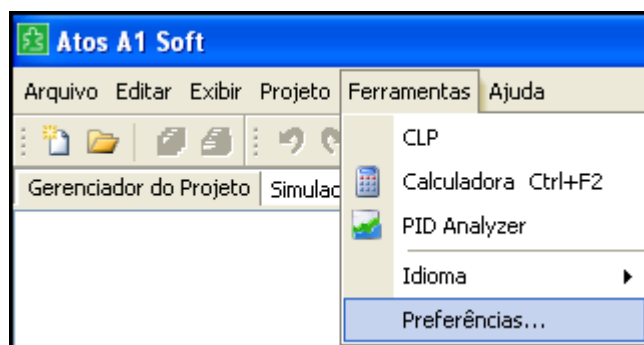
Configurações do Atos A1 Soft

Backup

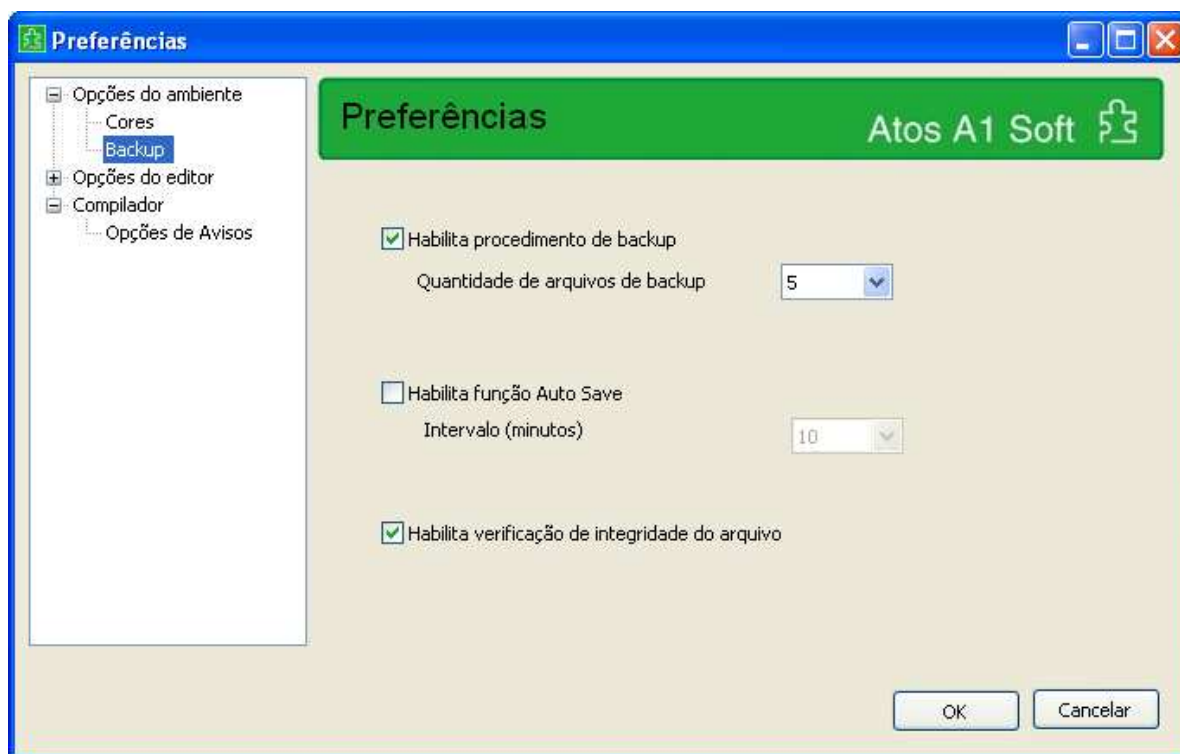
O backup é um recurso disponível no software Atos A1 Soft, que possibilita armazenar diversas versões de ATS em arquivos de backup.

Para configurar o backup o seguinte procedimento deve ser seguido:

No menu “Ferramentas” selecione a opção “Preferências...”, conforme a figura abaixo:



Ao selecionar a opção “Preferências...”, a janela abaixo será exibida com as configurações do backup.



Descrição dos campos:

- Habilitar procedimento de backup - (Padrão: habilitado)
- Quantidade de arquivos de backup (varia de 1 a 10. Padrão: 5)
- Habilitar função Auto Save (Padrão: desabilitado)
- Intervalo (em minutos) (opções de 10/30/60 min. Padrão 30 min.)
- Habilita verificação de integridade do arquivo (Padrão: habilitado)

Backup de arquivo

O backup de arquivo é uma função automática do Atos A1 Soft que por padrão já vem habilitada. O usuário pode ter a opção de desabilitá-la, porém uma mensagem de Warning é exibida informando os riscos envolvidos em desabilitar essa função.

Ao salvar o programa, o Atos A1 Soft copia o arquivo salvo anteriormente para a mesma pasta mudando a extensão do arquivo para *.bk1. Caso já exista um arquivo *.bk1, o mesmo será renomeado para *.bk2. Esse procedimento é realizado até a quantidade estipulada nas opções da ferramenta. Após isso a versão mais antiga é descartada.

Exemplo para quantidade de backups = 3:

1. Criado o arquivo Teste.ats.

2. Primeira tentativa de salvar o projeto:
Teste.ats

Quando o arquivo é salvo pela primeira vez, nenhum arquivo de backup é salvo, pois não existia arquivo anterior;

3. Segunda tentativa de salvar projeto:
Teste.ats → Teste.bk1
Teste.ats sobrescrito com novo conteúdo;

4. Terceira tentativa de salvar projeto:
Teste.bk1 → Teste.bk2
Teste.ats → Teste.bk1
Teste.ats sobrescrito com novo conteúdo;

5. Quarta tentativa de salvar projeto:
Teste.bk2 → Teste.bk3
Teste.bk1 → Teste.bk2
Teste.ats → Teste.bk1
Teste.ats sobrescrito com novo conteúdo;

6. Quinta tentativa de salvar projeto:
Teste.bk 3 → Descartado
Teste.bk 2 → Teste.bk3
Teste.bk1 → Teste.bk2
Teste.ats → Teste.bk1
Teste.ats sobrescrito com novo conteúdo;

Função Auto Save

O *Auto Save* é uma funcionalidade do Atos A1 Soft que faz a recuperação de um projeto em caso de arquivo corrompido.

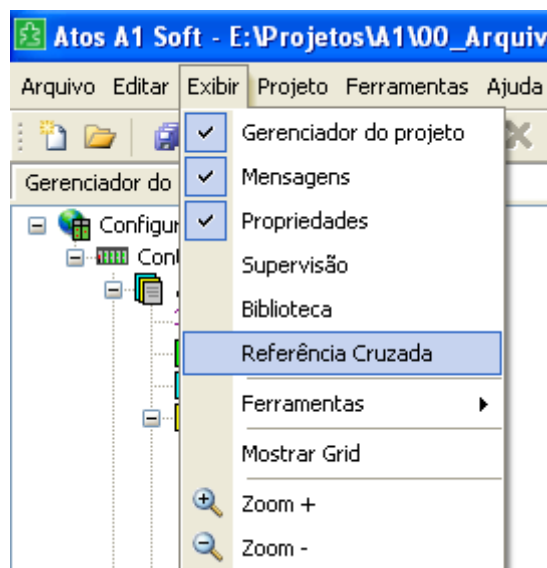
A função *Auto Save* cria um arquivo temporário na pasta do projeto enquanto existir alterações não salvas. O intervalo de tempo da função pode ser configurável no menu Ferramentas, opção Backup;

Referência Cruzada

A referência cruzada é um recurso disponível no software Atos A1 Soft, que permite localizar variáveis globais sinônimas, relacionadas em lugares distintos no projeto.

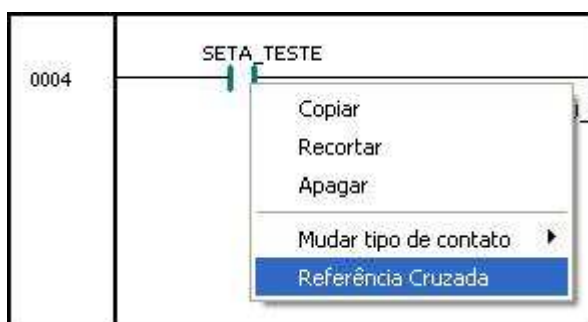
Para acessar a Referência Cruzada:

No Menu "Exibir" selecione "Referência Cruzada", conforme a figura abaixo:



Ou

No Ladder com o foco na variável, clique com o botão direito do mouse. No menu pop-up selecione a opção "Referência Cruzada", conforme a figura abaixo:



Ao selecionar a opção “Referência Cruzada”, a janela abaixo será exibida.

Endereço IEC	Nome	Tipo de Dado	Componente	Posição 1	Posição 2	Posição 3
%MX240	SETA_TESTE	BOOL				
%MX240	SETA_TESTE	BOOL	LADDER	POU: Programa_ciclico	Linha: 6	CONTACT
%MX240	SETA_TESTE	BOOL	LADDER	POU: Interrupcao_E0	Linha: 4	CONTACT

Descrição dos campos:

- Tipo de Filtro – Nome, Endereço IEC ou Tipo de Dado.
- Valor do Filtro – O que deseja referenciar.

Descrição das Colunas:

- Endereço IEC – Mostra o endereço da variável
- Nome – Mostra o nome da variável
- Tipo de Dado – Mostra o Tipo de Dado da variável
- Componente – Mostra em qual componente existe essa variável (Ex: Ladder, IHM, Access path e Receita)
- Posição – Indica as posições da variável dentro do componente. (Ex: Componente: LADDER – Posição 1: “POU Programa_ciclico” – Posição 2: “Linha: 2” – Posição 3: “CTU.CV”)

Exemplos de filtragem:

Filtro por Nome

No campo “Tipo de Filtro” selecionar “Nome”

No campo “Valor do Filtro” Digite o nome da Variável a ser filtrada, conforme a figura abaixo:

Endereço IEC	Nome	Tipo de Dado	Componente	Posição 1	Posição 2	Posição 3
%MD25	CNT_RESET	DINT				
%MD25	CNT_RESET	DINT	LADDER	POU: Programa_ciclico	Linha: 2	CTU.CV

Filtro por Endereço IEC

No campo “Tipo de Filtro” selecionar “Endereço”

No campo “Valor do Filtro” Digite o Endereço IEC da Variável a ser filtrada, conforme a figura abaixo.

Referência Cruzada das Variáveis Globais

Tipo de Filtro: **Endereço** Valor do Filtro: **%MD1** **Atualizar**

Endereço IEC	Nome	Tipo de Dado	Componente	Posição 1	Posição 2	Posição 3
%MD1	PT_beg	DINT				
%MD1	PT_beg	DINT	LADDER	POU: Interrupcao_E0	Linha: 3	MOVE.OUT
%MD1	PT_beg	DINT	LADDER	POU: Interrupcao_E0	Linha: 4	CPU_COUNTER.PT_beg

Filtro por Tipo de Dado

No campo “Tipo de Filtro” selecionar “Tipo de Dado”

No campo “Valor do Filtro” selecionar o Tipo de dado da Variável a ser filtrada, conforme a figura abaixo.

Referência Cruzada das Variáveis Globais

Tipo de Filtro: **Tipo de Dado** Valor do Filtro: **DATE_AND_TIME** **Atualizar**

Endereço IEC	Nome	Tipo de Dado	Componente	Posição 1	Posição 2	Posição 3
%MT153	efetivoDAT	DATE_AND_TIME				
%MT153	efetivoDAT	DATE_AND_TIME	LADDER	POU: Comunic_Serial_72R	Linha: 10	*_TO_*.OUT
%MT153	efetivoDAT	DATE_AND_TIME	LADDER	POU: Comunic_Serial_72R	Linha: 21	NE.IN1
%MT153	efetivoDAT	DATE_AND_TIME	ACCESS PATH	4004.72R A2 Channel 2	Frame 10	Variável Inicial
%MT157	LEITURA_DAT	DATE_AND_TIME				
%MT157	LEITURA_DAT	DATE_AND_TIME	LADDER	POU: Comunic_Serial_72R	Linha: 21	NE.IN2
%MT157	LEITURA_DAT	DATE_AND_TIME	ACCESS PATH	4004.72R A2 Channel 2	Frame 21	Variável Inicial

CAPÍTULO 3

IHM

Configuração de IHM

Para inserir uma IHM no projeto, selecione com o mouse o item **Hardware** “ Hardware” na área de **Gerenciamento de Projeto**, clique com o botão direito e selecione a opção **Ver Catálogo de Hardware**. Neste catálogo, será apresentada uma lista de IHMs possíveis de serem configuradas.

Para adicionar uma IHM, dê um duplo clique na referência selecionado ou arraste para o item **Hardware**. Se desejar excluir a IHM, clique com o botão direito do mouse no ícone da IHM e selecione a opção **Apagar**.

Propriedades da IHM

Para visualizar as propriedades da IHM, selecione o ícone "🖥️" da IHM com o cursor do mouse e automaticamente a guia **Propriedades** exibirá características da IHM e itens para configuração.

Descrição:

Propriedades		Supervisionar variáveis
IHM		
Contraste	83	
Linguagem	Português	
Habilita Receitas	false	
Direção da Navegação	Up/Down	
Código	6006P92	
Descrição	- IHM 4x20 LCD	

Alarmes	
Primeira variável	
Primeira tela	
Quantidade	0
Tempo ON (ms)	1600
Tempo OFF (ms)	1000
Timeout (ms)	5000

Teclado	
Tecla ESC	
Teclas K	
K1	
K2	
K3	
K4	
K5	
K6	
K7	
K8	
K9	
K0	
Teclas F	
F1	
F2	
F3	
F4	

Senhas	
Senha 1	
Senha 2	
Senha 3	
Senha 4	
Senha 5	
Senha 6	
Senha 7	
Senha 8	

IHM

- Altera o contraste da tela numa escala de 0 (menor contraste) a 100 (maior contraste) **A alteração do contraste é apenas no LCD da IHM representado no projeto do Atos A1 Soft;**
- Possibilita a configuração do idioma para exibição das mensagens do sistema;
- Habilita ou não a configuração de [receitas por meio da IHM](#);
- Exibe código da IHM;
- Exibe uma pequena descrição da IHM mostrando o tipo de display e o tamanho da tela.
- Configura a direção de navegação de telas: Up/Down ou Down/Up

Alarmes:

- Esta guia permite a [configuração de alarmes](#) da IHM.

Teclado:

- Configuração das teclas ESC, K e F como chamada de tela. Indique o número da tela de chamada no campo da tecla a ser associada. Veja também sobre LEDs da IHM.

Senhas:

- Configuração de até oito níveis de senhas de permissão para edição dos campos nas telas.

Tipos de IHM

O Atos A1 Soft programa duas famílias distintas de IHM:

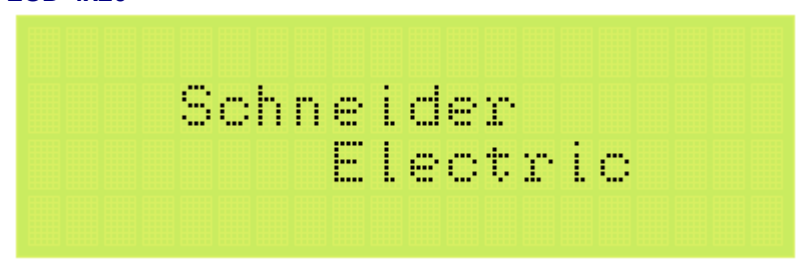
- LCD 2x20 (IHMs de display LCD e Negative LCD, com 2 linhas de 20 caracteres);
- LCD 4x20 (IHMs de display LCD e LCD Big Digits, com 4 linhas de 20 caracteres).

O número de telas disponíveis para programação depende apenas do espaço disponível na memória do CLP.

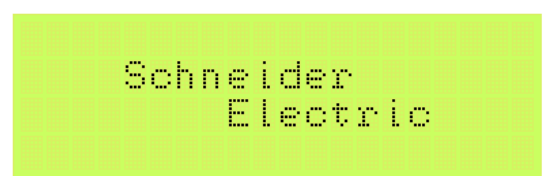
LCD 2x20



LCD 4x20

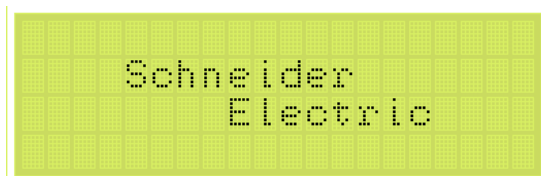


Contraste da tela



O software Atos A1 Soft possibilita alterar o contraste da tela numa escala de 0 (menor contraste) a 100 (maior contraste). Para alterar o contraste, selecione com o cursor do mouse o ícone da IHM em Gerenciador do Projeto. A alteração do contraste é apenas no LCD da IHM representado no projeto do

Atos A1 Soft;



Alarmes da IHM

A função Alarmes na guia **Propriedades da IHM** permite ao programador alertar, através de mensagens na tela da IHM, a ocorrência de alarmes/alertas ocorridos na máquina ou processo.

Descrição dos campos:

Alarmes	
Primeira variável	
Primeira tela	
Quantidade	0
Tempo ON (ms)	1600
Tempo OFF (ms)	1000
Timeout (ms)	5000

- **Primeira variável:** Indica qual é a primeira variável booleana (variáveis **globais de usuário**) da sequência para disparar as telas de alarme.
- **Primeira tela:** Indica qual é a primeira tela da sequência de telas de alarmes. As telas de alarmes são associadas, na sequência direta, com as variáveis booleanas de alarmes, definidas no campo Primeira variável.

- **Quantidade:** Indica quantas telas e variáveis booleanas de alarmes serão utilizadas, a partir do primeiro booleano/tela de alarme definido no seu respectivo campo. A quantidade máxima é de 64 alarmes.
- **Tempo ON (ms):** Define o período em que a tela de alarme será exibida durante a intermitência.
- **Tempo OFF (ms):** Define o período em que a tela de alarme não será exibida durante a intermitência.
- **Timeout:** Define o período de interrupção da exibição dos alarmes. Ao pressionar qualquer tecla da IHM, os alarmes são interrompidos, habilitando as telas de navegação. Caso mais nenhuma tecla seja pressionada, após o tempo de timeout, os alarmes voltam a aparecer na tela.

Receitas

Arquivo de Receitas é utilizado em aplicações onde existe a necessidade de se carregar, em endereços pré-determinados, conjuntos de parâmetros previamente configurados e armazenados na memória do CLP.

O arquivo de receitas pode ser trabalhado de duas formas diferentes:

- Ladder - Por meio do *Bloco Funcional Recipe*;
- Via IHM.

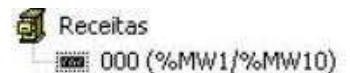
Essa função realiza o armazenamento ou a recuperação de até **16 frames** de dados. O conjunto de frames programados formará uma **Receita**.

Selecionando o ícone Receitas  **Receitas**, a guia de **Propriedades** apresentará um campo **Quantidade** onde é editado o número de receitas (Mínimo **1** e Máximo **256**).

Configuração dos Frames

Para adicionar um novo Frame e configurá-lo, basta selecionar com o cursor do mouse o ícone Receitas  localizado na guia **Gerenciador do Projeto**, clicar com o botão direito e selecionar a opção **New Frame** (Máximo **16 Frames**). Se desejar excluir o Frame, clique com o botão direito do mouse no ícone do Frame e selecione a opção *Apagar*.

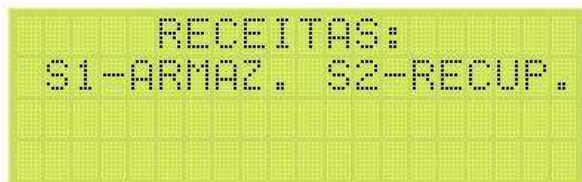
Em cada frame, identifique o primeiro e o último endereço das variáveis que serão utilizadas, respectivamente nos campos Início e Fim.



A configuração dos frames deve ser feita tanto para trabalhar com arquivo de receitas **via IHM** quanto **Bloco Funcional Recipe**.

Receitas via IHM

Utilizando o arquivo Receitas via IHM, basta inserir os dados nas variáveis declaradas nos Frames, pressionar a tecla  e em seguida a tecla . Será exibida a tela de Receitas com as opções de **Armazenar** uma receita (tecla ) ou Carregar uma receita (tecla )



Armazenar

Nesta tela é exibido o número da receita atual e o local para edição do número da receita onde serão armazenados os dados configurados.



Recuperar

Nesta tela é exibido o número da receita atual e o local para edição do número da receita a ser recuperada (carregada) com os dados configurados.



Para confirmar a recuperação ou armazenamento de uma receita pressione a tecla .

Para sair da tela de receitas pressione a tecla .

Senha da IHM

Este item permite a configuração de até oito níveis de senha para edição dos campos nas telas. As senhas podem ser alfa-numéricas, com limite de oito caracteres e letras (maiúsculas) disponíveis no teclado da IHM (A, B, C, D, E e F).

Não é possível utilizar como senha letras acentuadas, letras minúsculas, símbolos e outros tipos de caracteres não compatíveis com o teclado da IHM.

As senhas são classificadas:

- **Segurança Principal:** Inserida no campo *Password 1*, dá acesso a todas as telas para edição;
- **Senhas intermediárias:** Inseridas nos campos *Password 2* a *7*, são níveis de prioridades;
- **Segurança Menor:** Inserida no campo *Password 8*, é o nível mais baixo de prioridade para edição das telas.

Os níveis de prioridade são obedecidos do maior nível até o menor, ou seja, telas com nível de acesso 6 poderão ser editadas por usuários com senhas do *Password 1* até 6.

Já telas com nível de acesso 1 (Segurança Principal), poderão ser editadas apenas por usuários que possuem a senha do *Password 1*.

As telas com nível de Segurança Menor (*Password 8*) podem ser editadas por todos os usuários dos outros níveis de acesso.

Função Senha

Sempre que o usuário editar uma tela que contenha nível de acesso, deverá inserir o *password* correspondente ao nível ou nível superior.

Pressionando a tecla de edição  será exibida a seguinte tela:



No momento em que a senha de acesso (password) for inserida, todas as telas configuradas com aquela senha estarão habilitadas para edição. Pressione a tecla enter  para confirmar a senha (password).



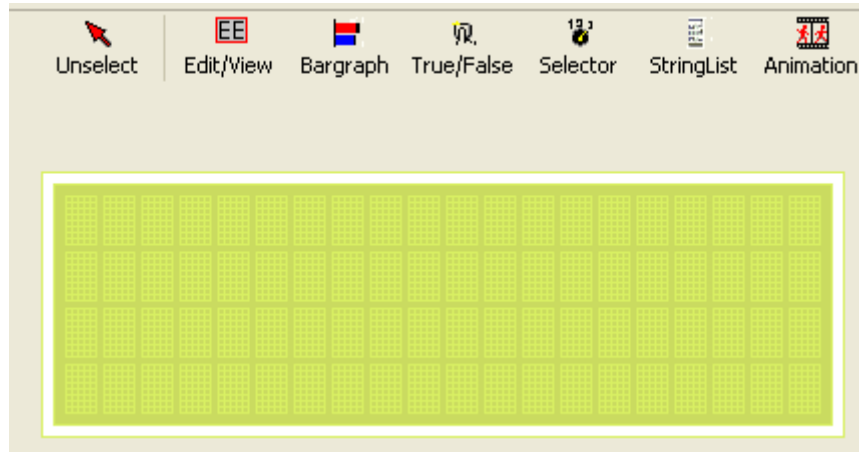
Para retornar com a senha do nível de acesso das telas programadas, pressione o botão  e digite novamente a senha. Confirme a operação pressionando a tecla enter .



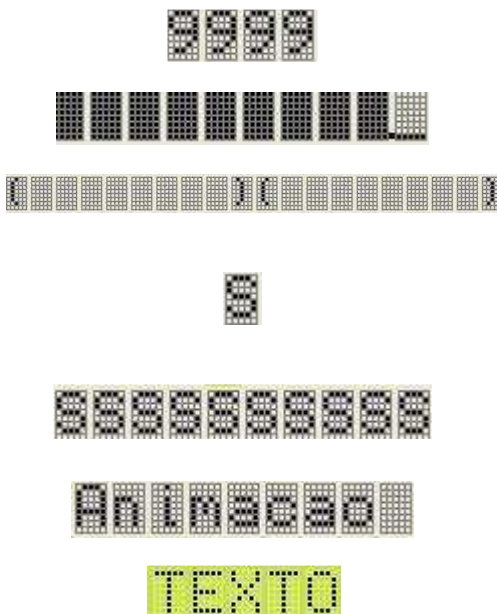
Inserir TELA

Clique com o botão direito do mouse sobre a IHM do projeto e selecione a opção **Inserir Tela**.

Cada tela possui um campo de propriedades para configuração de características (Propriedades da Tela). Na parte superior da tela (**Área de Objetos**) são exibidos os objetos (Edit/View, Bargraph, True/False, Selector, StringList e Animation) para serem inseridos.



Descrição dos objetos:



- **Edit/View:** Campo de visualização e/ou edição de dados associados à memória do CLP.
- **Bargraph:** Campo de visualização gráfica de dados associados à memória do CLP.
- **True/False:** Campo de liga (True) e desliga (False) um dado associado à memória do CLP.
- **Selector:** Campo do tipo chave seletora. O dado na posição indicada é presetado para "True", enquanto os dados das outras posições são presetados automaticamente para "False".
- **Stringlist:** Campo do tipo string que possibilita alternar, na tela, textos previamente digitados, de acordo com o tipo de dado associado.
- **Animation:** Campo de animação, no qual possibilita configurar um texto e animá-lo acionando uma variável booleana.
- **Texto:** Campo que permite a edição de textos na tela da IHM.

Propriedades da Tela

Para visualizar as propriedades da Tela, basta selecionar o ícone da tela “” com o cursor do mouse e automaticamente a guia **Geral** exibirá características da tela de visualização e de configuração.

Descrição dos campos:

Propriedades		Supervisionar variáveis
Geral		
Id	0	
Nome	Tela0	
Nível de segurança	Acesso livre	
Tela-alvo S1		
Tela-alvo S2		
Descrição		
Navegação	Início/Fim	

- **ID:** Campo de visualização mostra o número de identificação da tela. Este número é seqüencial;
- **Nome:** Campo de edição do nome da tela;
- **Nível de segurança:** Seleciona o nível de segurança para acesso aos campos de edição da tela;

- **Tela Alvo - S1 e S2:** Possibilita configurar as teclas S1 e S2 como chamada de tela;
- **Navegação:** Configura a tela como sendo de Início, Intermediário, Fim ou Início/Fim.

Programação de Telas

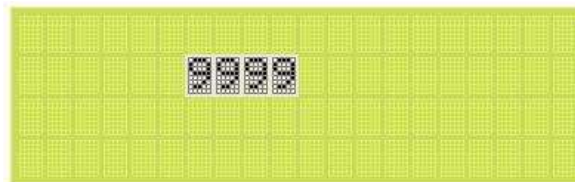
Campo Edit/View

Para inserir um campo *Edit/View*, selecione com o mouse o item **Edit/View** na área de **objetos** da tela e clique com o cursor do mouse na posição desejada da tela.

Para cada campo *Edit/View* adicionado será criado um ícone  referente na área do **Gerenciador do Projeto**. Se desejar excluir o campo, clique com o botão direito do mouse no ícone do campo e selecione a opção **Apagar**.

Selecione o ícone do campo para alterar suas propriedades visualizadas na guia **Geral**.

Na edição de campo a tecla  insere sinal de negativo, a tecla  apaga dado digitado e a tecla ESC retorna ao valor anterior.



Descrição dos campos:

Para %MA.., %MD.. e %MW..

Geral	
Coluna	7
Linha	2
Variável	
Descrição	
Tamanho	4
Precisão	0
Min	0
Máx	0
Editável	true

- **Coluna:** Posicionamento do campo na tela em relação à coluna;
- **Linha:** Posicionamento do campo na tela em relação à linha;
- **Variável:** Insere o endereço da variável associada ao campo;



Para variável do tipo STRING, o campo edit/view só poderá ser de visualização, mesmo que Editável=TRUE, não será possível a edição do campo pela IHM.

Para %MR..

Geral	
Coluna	10
Linha	4
Variável	%MR1
Descrição	
Tamanho	1
Precisão	0
Min	
Máx	
Notação Científica	true
Editável	true

- **Descrição:** Insere uma descrição referente ao campo;
- **Tamanho:** Especifica o tamanho do campo, ou seja, número de caracteres que ocupará na tela da IHM;
- **Precisão:** Especifica a precisão do campo;
- **Min.:** Especifica o valor mínimo de edição no campo (habilitado somente quando campo **Editável** for TRUE);
- **Máx.:** Especifica o valor máximo de edição no campo (habilitado somente quando campo Editável for TRUE);
- **Editável:** Configura campo como sendo apenas de visualização ou edição e visualização.

Para %MT..

Geral	
Coluna	10
Linha	4
Variável	%MR1
Descrição	
Tamanho	1
Precisão	0
Min	
Máx	
Notação Científica	true
Editável	true

ESPECIAIS

- **Notação Científica:** Configura campo para notação científica (Exemplo 9,8E+10).
- **Máscara:** Configura campo com máscaras previamente configuradas ou estilo digitado pelo usuário (Ver detalhes a seguir).

Notação Científica:

O uso da notação científica é de uso exclusivo para campos que utilizem tipo de dado REAL (%MR..).

A notação científica é uma forma de representar números, em especial muito grandes (100000000000) ou muito pequenos (0,00000000001). É baseada no uso de potências de 10, no Atos A1 Soft a potência 10 é representada pela letra maiúscula **E**.

Exemplo:

$$100000 = 1,0.10^5 = 1.0E+05$$

$$0,00001 = 1,0.10^{-5} = 1.0E-05$$

O número de dígitos do expoente **E** são 2 dígitos mais o sinal de + ou -. No campo **Precisão** da guia de **Propriedades** do campo real é definido o número de dígitos após o ponto decimal. O valor será expresso sempre com um dígito antes do ponto decimal.

Exemplo:

VALOR	N.C.	A1
Precisão = 0 → 900000	= 9,0.10 ⁵	= 9E+05
Precisão = 1 → 980000	= 9,8.10 ⁵	= 9.8E+05
Precisão = 2 → 987000	= 9,87.10 ⁵	= 9.87E+05
Precisão = 3 → 987600	= 9,876.10 ⁵	= 9.876E+05
Precisão = 4 → 987650	= 9,8765.10 ⁵	= 9.8765E+05

Teclas:

-  insere o sinal de negativo;
-  apaga dígito;
-  insere ponto decimal;
-  baixo insere E (base 10).

Exemplo:

Inserir o valor **-0,000085**. Campo com precisão = 1.

1º Digite 85 ou 8, tecla  para inserir ponto decimal e em seguida 5;

2º Pressione  para inserir o sinal negativo;

3º Pressione  para inserir **E** (base 10);

4º Digite 6 e pressione  para inserir o sinal negativo do expoente;

Resultado no campo da IHM é **-8.5E-06**

Inserindo um valor sem ponto decimal e expoente da base **E**, **automaticamente o valor será ajustado para o formato de notação científica** e até mesmo arredondado conforme a precisão do campo real. A regra de arredondamento para o campo real é da seguinte forma:

- Valor final maior ou igual a 5, acréscimo de uma unidade no dígito a ser arredondado e eliminam-se os dígitos restantes;

Precisão = 3 → 94965 = 9.497E+04

Precisão = 1 → 94965 = 9.5E+04

Precisão = 3 → -54837 = -5.484E+04

Precisão = 2 → -0,00005436759 = 5.44E-05

- Valor final menor do que 5, mantém o valor arredondado e eliminam-se os dígitos restantes.

Precisão = 3 → 94964 = 9.496E+04

Precisão = 2 → 92365 = 9.2E+04

Precisão = 3 → -54831 = -5.483E+04

Precisão = 2 → -0,0000543328 = 5.43E-05

Visualização na IHM:

```

REAL
%MR1:  6.022E+23
%MR2:  9.854E+10
%MR3: -5.426E-26
    
```



Quando campo EDIT/VIEW for apenas de visualização, ou seja, Editável=FALSE e a variável mostrada for do tipo REAL, deve-se observar:

O TAMANHO do campo e da variável a ser mostrada, pois se a variável for maior que o campo ocorre erro de overflow.

O OVERFLOW nessa situação mostra na tela da IHM os caracteres "#####", no lugar da variável.

Máscaras

Os campos **Edit/View** na IHM relacionados com data e hora. Para cada tipo de dado (TIME, DATE, TIME_OF_DAY e DATE_AND_TIME) a guia de propriedades exibe um grupo de máscaras pré-definidas.

O usuário pode configurar o tipo de máscara que desejar, de acordo com os itens do tipo de dado disponível, onde pode inserir caracteres de separação e ordenar da forma que desejar os dados, tanto para edição quanto visualização (ver exemplos).

TIME

- **dd** - dias;
- **hh** - horas;
- **mm** - minuto;
- **ss** - segundos;
- **d** - décimo de segundo;
- **mmm** - milésimo de segundo.

Máscara	mmm
Editável	mmm ss:d ss:mmm mm:ss:mmm hh:mm:ss dd:hh:mm:ss

Exemplo

Máscara mm#hh#dd

Máscara mm-ss-d

Máscara hh*mm*ss*mmm

DATE

- **dd** - dia;
- **MM** - mês;
- **yy** - ano;
- **yyyy** - século e ano.

Máscara	dd/MM/yyyy
Editável	dd/MM/yyyy dd-MM-yyyy dd.MM.yyyy dd/MM/yy dd-MM-yy dd.MM.yy MM/dd/yyyy MM-dd-yyyy

Exemplo

Máscara yy#dd#MM

Máscara dd_MM_yyyy

Máscara yyyy*MM*dd

TIME OF DAY

- **HH** - hora no formato 24 h;
- **hh** - hora no formato 12 h;
- **mm** - minutos;
- **ss** - segundos.



Exemplo

Máscara HH*mm*ss

Máscara hh#mm#ss

Máscara hh_ss

DATE AND TIME

- **dd** - dia;
- **MM** - mês;
- **yy** - ano;
- **yyyy** - século e ano.
- **HH** - hora no formato 24 h;
- **hh** - hora no formato 12 h;
- **mm** - minutos;
- **ss** - segundos.



Exemplo

Máscara #hh:mm:ss*dd/MM/yy#

Máscara HH:mm MM-dd-yyyy

Máscara dd*MM*yyyy hh#mm#ss

Tipos de dado para o campo Edit/View:

Cada dado possui características específicas listadas abaixo:

TIPO DE DADO	TAMANHO (em memória)	INTERVALO
INT	16 bits	-32768 a +32767
UINT	16 bits	0 a 65535
WORD	16 bits	0 a FFFF
DINT	32 bits	-2147483648 a +2147483647
UDINT	32 bits	0 a 4294967295
DWORD	32 bits	0 a FFFFFFFF
REAL	32 bits	-3.40282346638528860e+38 a 3.40282346638528860e+38 Underflow: 1.1754943508222875e-38
DATE	32 bits	01/01/1970 a 31/12/2038
TIME	32 bits	0 a 49d17h2m47s290ms
TIME_OF_DAY	32 bits	00:00:00 a 23:59:59
DATE_AND_TIME	32 bits	01/01/1970 0:0:0 a 31/12/2038 23:59:59
STRING	-----	caracteres ASCII

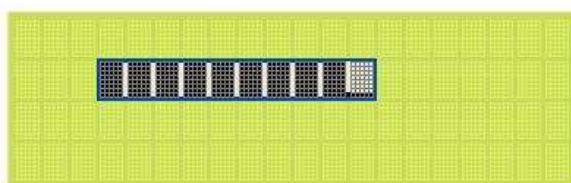
Campo Bargraph

Para inserir um campo Bargraph, selecione com o mouse o item **Bargraph** na área de **objetos** e clique com o cursor do mouse na posição desejada da tela.

Para cada campo *Bargraph* adicionado será criado um ícone  referente na área do **Gerenciador do Projeto**. Se desejar excluir o campo, clique com o botão direito do mouse no ícone do campo e selecione a opção **Apagar**.

Selecione com o cursor do mouse o ícone do *Bargraph* para exibir suas propriedades na guia **Geral**.

O *Bargraph* representa, de forma gráfica, o valor contido na variável associada. A representação pode ser feita na forma de um gráfico de barras contínuo, ou de um cursor na tela. O preenchimento do gráfico varia proporcionalmente ao conteúdo da variável associada, de acordo com a escala configurada.



Descrição dos campos:

Geral	
Coluna	3
Linha	2
Variável	
Descrição	
Tamanho	10
Min	0
Máx	100
Style	☐ ☐ ☐ ☐

- **Coluna:** Posicionamento do campo na tela em relação à coluna;
- **Linha:** Posicionamento do campo na tela em relação à linha;
- **Variável:** Insere o endereço da variável associada ao campo;
- **Descrição:** Insere uma descrição referente ao campo;
- **Tamanho:** Especifica o tamanho do campo, ou seja, número de caracteres que ocupará na tela da IHM;

- **Mín.:** Limite inferior da escala do Bargraph;
- **Máx.:** Limite superior da escala do Bargraph;
- **Style:** Seleciona o modo de exibição do *Bargraph*: gráfico de barras ou desvio (cursor na tela).

Campo True/False

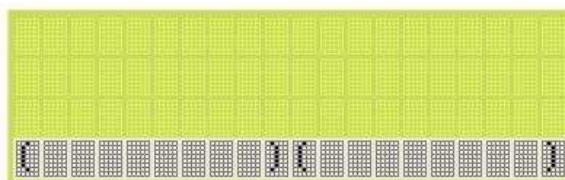
Para inserir um campo *True/False*, selecione com o mouse o item **True/False** na área de **objetos** e clique com o cursor do mouse na posição desejada da tela.

O campo *True/False* comuta, entre *True* e *False*, o status da variável associada.

Para cada campo *True/False* adicionado na tela será criado um ícone  referente na área do **Gerenciador do Projeto**. Se desejar excluir o campo, clique com o botão direito do mouse no ícone do campo e selecione a opção **Apagar**.

Selecione o ícone do campo para alterar suas propriedades visualizadas na guia **Geral**.

Pressionando a tecla Edit  da IHM o campo *True/False* será alterado de acordo com as teclas  para *True* e  para *False*.



Descrição dos campos:

Geral	
Coluna	1
Linha	4
Variável	
Descrição	
Text for True	
Text for False	

- **Coluna:** Posicionamento do campo na tela em relação à coluna. Não é possível editar;
- **Linha:** Posicionamento do campo na tela em relação à linha. Não é possível editar;
- **Variável:** Insere o endereço da variável associada ao campo;
Texto ON: Texto que será exibido na tela associado ao status ON;
- **Texto OFF:** Texto que será exibido na tela associado ao status OFF.

Campo Selector

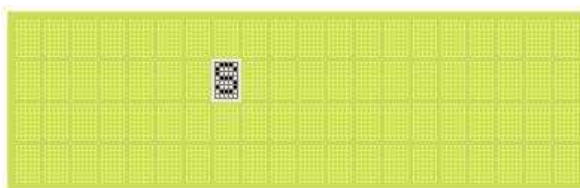
Para inserir um campo *Selector*, selecione com o mouse o item **Selector** na área de **objetos** da tela e clique com o cursor do mouse na posição desejada da tela.

Para cada campo *Selector* adicionado na tela será criado um ícone  referente na área do **Gerenciador do Projeto**. Se desejar excluir o campo, clique com o botão direito do mouse no ícone do campo e selecione a opção **Apagar**.

Selecione o ícone do campo para alterar suas propriedades visualizadas na guia **Geral**.

Através do campo *Selector*, o usuário edita, para *True*, o status de uma variável booleana da seqüência associada ao campo. Todas as outras variáveis da seqüência serão automaticamente setadas para *False*. Para selecionar a variável a ser setada para *True*, o usuário deverá editar no campo *Selector*, a posição em que esta variável aparece na seqüência:

- Para setar a primeira variável booleana da seqüência, o número 1 deve ser digitado no campo;
- Para setar a segunda variável booleana, o número 2, etc.;
- Para desligar todas as variáveis da seqüência, o usuário deverá digitar “0” no campo *Selector*.



Descrição dos campos:

Geral	
Coluna	8
Linha	2
Variável	
Descrição	
Posições	4

- **Coluna:** Posicionamento do campo na tela em relação à coluna;
- **Linha:** Posicionamento do campo na tela em relação à linha;
- **Variável:** Insere o endereço da primeira variável booleana da seqüência associada ao campo;

- **Descrição:** Insere uma descrição referente ao campo;
- **Posições:** Número de variáveis booleanas da seqüência associada ao campo *Selector*. Varia de 1 a 9.

Campo StringList

Para inserir um campo *StringList*, selecione com o mouse o item ***StringList*** na área de **objetos** e clique com o cursor do mouse na posição desejada da tela.

Para cada campo *String* adicionado na tela, será criado um ícone  referente na área do **Gerenciador do Projeto**. Se desejar excluir o campo, clique com o botão direito do mouse no ícone do campo e selecione a opção **Apagar**.

Selecione o ícone do campo para alterar suas propriedades visualizadas na guia **Geral**.

O campo *String* alterna, na tela, textos previamente digitados, de acordo com o conteúdo de variáveis ou sequência de variáveis booleanas associadas a ele.

Há dois modos diferentes de operação, de acordo com o tipo de dado selecionado:

- **Variáveis ou sequência de variáveis booleanas.**

String associada a uma variável

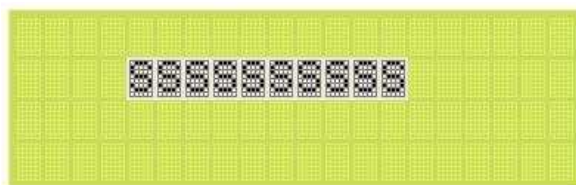
Cada texto do campo *string* é associado a um número. Será exibido o texto cujo número for igual ao valor da variável associada ao campo *string*. Os números associados aos textos são dados em ordem crescente, a partir do número zero, desta forma, o primeiro texto está associado ao número 0, o segundo texto, ao 1, etc.

String associada a uma sequência de variáveis booleanas

Cada texto do campo *string* é associado a uma variável booleana da sequência. Será exibido o texto cuja variável booleana associada estiver em ON.

As variáveis booleanas são associadas aos textos em ordem direta, a partir do segundo texto, desta forma, o segundo texto está associado ao primeiro booleano, o terceiro texto está associado ao segundo booleano, etc. O primeiro texto será exibido quando todas variáveis booleanas estiverem desligadas.

Quando mais de um booleano é acionado, o CLP mostra o *string* correspondente ao primeiro booleano que está acionado, ou seja, se o primeiro e o último booleano estiverem ligados, o texto exibido será aquele cuja *string* está relacionada a primeira variável booleana.



Descrição dos campos:

Geral	
Coluna	5
Linha	2
Variável	
Descrição	
Tamanho	10
Strings	SSSSSSSSSS
Style	Normal

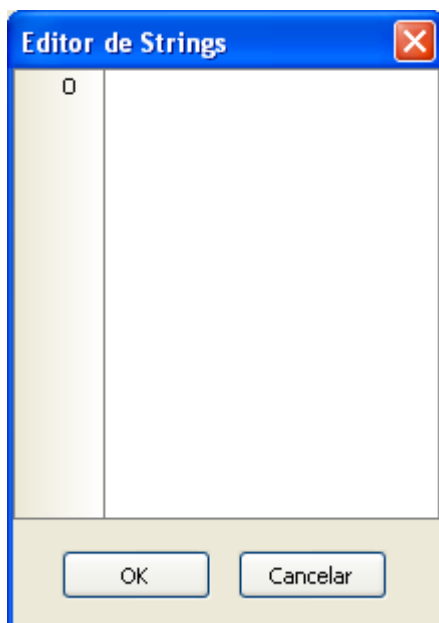
- **Coluna:** Posicionamento do campo na tela em relação à coluna;
- **Linha:** Posicionamento do campo na tela em relação à linha;
- **Variável:** Insere o endereço da primeira variável associada ao campo;
- **Descrição:** Insere uma descrição referente ao campo
- **Tamanho:** Especifica o tamanho do campo, ou seja, número de caracteres que ocupará na tela da IHM;

- **Strings:** Abre janela para o Editor de *Strings*;
- **Style:** Especifica o estilo de exibição da *string* como sendo normal ou piscante.

Editor de String

Nesta janela é configurado o texto da string e a quantidade de strings (Máximo 255).

Para adicionar uma nova linha da string, basta pressionar Enter e para apagar a linha pressiona Backspace.



Campo Animation

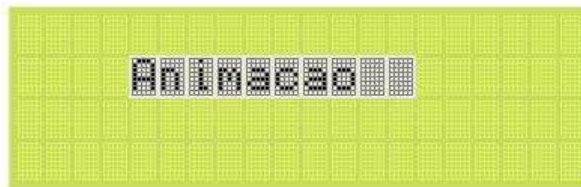
Para inserir um campo *Animation*, selecione com o mouse o item **Animation** na área de **objetos** e clique com o cursor do mouse na posição desejada da tela.

Para cada campo *Animation* adicionado será criado um ícone  referente na área do **Gerenciador do Projeto**. Se desejar excluir o campo, clique com o botão direito do mouse no ícone do campo e selecione a opção **Apagar**.

Selecione o ícone do campo para alterar suas propriedades visualizadas na guia **Geral**.

O *Animation* apresenta um texto animado que pode ter associado uma variável *string*. A animação é acionada por uma **variável booleana** associada ao campo.

A configuração da animação pode ser feita tanto na **Guia de Propriedades (Style Constante)** ou através de variáveis associadas a campos de edição (**Style Variável**).



Descrição dos campos:

Geral	
Coluna	5
Linha	2
Descrição	
Tamanho	10
Habilita variável	
Variável de status	
Style	Normal

- **Coluna:** Posicionamento do campo na tela em relação à coluna;
 - **Linha:** Posicionamento do campo na tela em relação à linha;
 - **Descrição:** Insere uma descrição referente ao campo
 - **Tamanho:** Especifica o tamanho do campo, ou seja, número de caracteres que ocupará na tela da IHM;
- **Habilita Variável:** Define a variável **booleana** que inicia a animação. Para que a animação fique sempre ativa basta digitar *TRUE* (Always ON) neste campo;
 - **Variável de Status:** Define a variável **booleana** que sinaliza o término de um ciclo da animação. Caso não queira visualizar o status, deixar este campo em branco.

Para os itens **Texto**, **Direção** e **Velocidade** existem dois modos de configuração:

Style Constante

- **Texto:** O texto da animação é definido no campo **Valor**.
- **Direção:** A direção da animação do texto é definida no campo **Valor**. A animação pode ser da direita para esquerda ou da esquerda para direita.
- **Velocidade:** A velocidade da animação (resolução 100ms) também é definida no campo **Valor**. O valor mínimo aceito neste campo é 1. Quanto maior for este valor menor será a velocidade da animação.

Style Variável

- **Texto:** O texto da animação será o conteúdo de uma variável do tipo *String*, definida no campo **Variável**.
- **Direção:** Com a variável booleana ligada (True), o deslocamento é da **direita para esquerda**. Quando desligada (False) o deslocamento é da **esquerda para direita**.
- **Velocidade:** A velocidade da animação (resolução 100ms) é definida por uma variável do tipo **UINT** (inteiro não sinalizado) declarada no campo **Variável**. O valor mínimo permitido é 1. Quanto maior o valor, menor será a velocidade da animação.

Texto

Permite a edição de textos na tela da IHM. Selecione com o cursor o caracter na tela da IHM onde será digitado o texto.

Para movimentar o cursor sobre a tela, utilize as setas direcionais do teclado ou o cursor do mouse.



Na edição de textos o LCD da IHM aceita apenas caracteres não acentuados.

Schneider
Electric

Navegação das telas

A navegação de telas na IHM é determinada diretamente pela ordem e seqüência com que elas foram programadas no projeto.

A seqüência em que elas aparecem na lista de telas (Gerenciador **do Projeto**) é a mesma com que elas serão exibidas na tela da IHM. Quando pressionamos o botão "**Seta para cima**", a tela posterior à tela atual é exibida na interface.

Quando pressionamos o botão "**Seta para baixo**", é exibida a tela anterior à atual. Se a tela de número 1 está sendo exibida na tela, e pressionamos o botão "**Seta para cima**", será exibida a tela 2; se pressionarmos o botão "**Seta para baixo**", a tela 0 será exibida.

Usualmente, é utilizado o conceito de blocos de telas para a programação de uma IHM. Por este conceito, criamos uma seqüência de telas com alguma característica funcional em comum; a seguir, definimos a primeira tela como "**início de bloco**" (Início) e a última como "**fim de bloco**" (Fim), limitando o conjunto. Associamos então a primeira tela de nosso bloco a um dos métodos de chamada de tela.

Para uma única tela configurada utilize a opção **Início/Fim**.

Telas configuradas entre uma tela inicial e uma tela final devem ser configuradas como **Intermediária**.

Para o usuário navegar para as outras telas da seqüência, utiliza-se as teclas de navegação da IHM 
ou .

Há 4 métodos de chamada de telas:

- Tela Zero;
- Teclas da IHM;
- Teclas de navegação S1 e S2;
- Programa-usuário.

Tela Zero

A tela de número "0" da lista de telas é utilizada pela IHM como tela default. É a tela exibida ao ligar o CLP. O sistema retorna as telas na seqüência que elas foram acessadas.

Teclas da IHM

Cada tecla da IHM pode ser associada a uma tela programada. Pressionando a tecla, a tela associada será exibida na IHM (Clique para saber como programar as Teclas K ou Teclas F da IHM).

Programa de usuário

Para chamar telas da IHM através do programa-usuário, trabalha-se em conjunto com uma variável do tipo UINT e uma variável do tipo BOOLEANA.

- Variável UINT do sistema: **ACTUAL_SCR** - número da tela atual e **TARGET_SCR** - número da tela alvo.

A variável booleana **GO_SCREEN** é um *trigger*, sensível à borda de subida; a cada pulso nesta variável, será chamada a tela cujo número estiver gravado na variável de sistema **TARGET_SCR**.

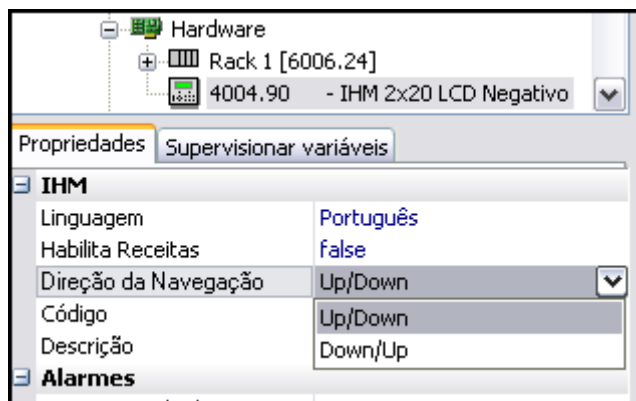
Para chamar a tela de número 12, por exemplo, devemos primeiro gravar o número 12 na variável **TARGET_SCR**, para logo após dar um pulso na booleana **GO_SCREEN**. Automaticamente, a tela 12 irá aparecer na IHM.

Navegação das telas

A navegação de telas na IHM é determinada diretamente pela ordem e seqüência com que elas foram programadas no projeto.

A seqüência em que elas aparecem na lista de telas (Gerenciador do Projeto) é a mesma com que elas serão exibidas na tela da IHM.

A direção das teclas “Seta para cima” e “Seta para baixo” é configurada nas propriedades da IHM no campo “Direção da navegação”, sendo default a configuração (up/down).



UP/ Down: Quando pressionamos o botão "**Seta para cima**", a tela posterior à tela atual é exibida na interface.

Quando pressionamos o botão "**Seta para baixo**", é exibida a tela anterior à atual.

Exemplo: Se a tela de número 1 está sendo exibida na tela e pressionamos o botão

"**Seta para cima**" será exibida a tela 2; se pressionarmos o botão "**Seta para baixo**", a tela 0 será exibida.

Down/ UP: Quando pressionamos o botão **"Seta para baixo"**, a tela posterior à tela atual é exibida na interface.

Quando pressionamos o botão **"Seta para cima"**, é exibida a tela anterior à atual.

Exemplo: Se a tela de número 1 está sendo exibida na tela, e pressionamos o botão

"Seta para baixo" será exibida a tela 2; se pressionarmos o botão **"Seta para cima"**, a tela 0 será exibida.

Usualmente, é utilizado o conceito de blocos de telas para a programação de uma IHM.

Por este conceito, criamos uma seqüência de telas com alguma característica funcional em comum; a seguir, definimos a primeira tela como "início de bloco" (Início) e a última como "fim de bloco" (Fim), limitando o conjunto. Associamos então a primeira tela de nosso bloco a um dos métodos de chamada de tela.

Para uma única tela configurada utilize a opção Início/Fim.

Telas configuradas entre uma tela inicial e uma tela final devem ser configuradas como Intermediária.

Para o usuário navegar para as outras telas da seqüência, utiliza-se as teclas de navegação da IHM



ou

Há 4 métodos de chamada de telas:

- Tela Zero
- Teclas da IHM
- Teclas de navegação S1 e S2
- Programa-usuário

Este um exemplo de como é a navegação (UP/Down) das telas utilizando a tecla  e .

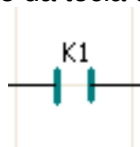


Teclas K

As Teclas K podem ser usadas de duas maneiras diferentes:

1. Na lógica do programa *Ladder*;
2. Como chamada de tela.

Lógica no programa ladder: Cada tecla possui um nome relacionado na memória do CLP, portanto pode ser usado na lógica do programa apenas o nome da tecla desejada.



Chamada de tela: A guia **K Keys** possibilita ao programador sua utilização como chamada de tela, isto é, através de um campo dentro da guia o programador indica qual tela deverá ser mostrada ao se pressionar a tecla desejada.



Caso não se utilize as teclas como chamada de tela, o campo de indicação da tela alvo deverá permanecer vazio, assim nenhuma tela estará relacionada ao respectivo botão.

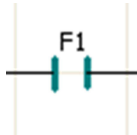
Todos os botões são programados individualmente podendo ser programados de forma não seqüencial.

Teclas F

As Teclas F podem ser usadas de duas maneiras diferentes:

1. Na lógica do programa *Ladder*;
2. Como chamada de tela.

Lógica no programa ladder: Cada tecla possui um nome relacionado na memória do CLP, portanto pode ser usado na lógica do programa apenas o nome da tecla desejada.



Chamada de tela: A guia **F Keys** possibilita ao programador sua utilização como chamada de tela, isto é, através de um campo dentro da guia o programador indica qual tela deverá ser mostrada ao se pressionar a tecla desejada.



Caso não se utilize as teclas como chamada de tela, o campo de indicação da tela alvo deverá permanecer vazio, assim nenhuma tela estará relacionada ao respectivo botão.

Todos os botões são programados individualmente podendo ser programados de forma não seqüencial.

Todas as teclas F presentes na IHM estão acompanhadas de um led que também possui seu respectivo endereço na memória do CLP, mas não existe nenhum intertravamento entre as teclas F e os leds. A utilização desses leds fica a cargo da necessidade do programador.

Auxílio Manutenção

Para ter acesso à tela de auxílio à manutenção é necessário ter uma IHM programada na configuração de hardware (clique aqui para ver como realizar essa configuração).

A tela de auxílio à manutenção é acessada pela tecla  da IHM. Para sair, pressione-a novamente.

Representação da tela de auxílio à manutenção:



Descrição dos campos:



Tipo de variável: Define qual o tipo de variável que está sendo supervisionada pelo auxílio à manutenção.

Utilize as teclas  e  para realizar a navegação.

Endereço da variável: Mostra o endereço corrente, visualizado pelo auxílio à manutenção. Utilize as teclas  e  para navegar dentro da região de memória do tipo de dado escolhido para supervisionar. As teclas numéricas também podem ser usadas para digitar um

endereço. Utilize a tecla  como *backspace* ao digitar um endereço.

Leitura da variável: Mostra o valor da variável selecionada.

- *Variáveis booleanas:* representadas como *TRUE* (ligada) ou *FALSE* (desligada). Para variáveis em modo *FORCE*, seu status será indicado pela string "(force)" ao lado do status da variável (conforme mostrado na figura ao lado).

Variáveis numéricas: São mostradas conforme o tipo de dados definido no campo "Tipo de variável" da tela.



CAPÍTULO 4

Editor

Como começar

Barra de ferramentas Ladder

A barra de ferramentas *ladder* possui todos os componentes para a edição de seu programa.



Botão	Descrição
	Inserir nova linha (padrão: abaixo da linha corrente). Tecla de atalho: I (insere linha abaixo,) U (insere linha acima).
	Inserir contato (padrão: contato aberto). Para escolher outro tipo de contato, clique na seta do botão para abrir o menu de opções. Tecla de atalho: A (contato aberto), F (contato fechado) e S (abre lista de entradas). Os outros tipos de contato são acessados através desta barra de ferramentas.
	Abrir braço paralelo. Tecla de atalho: [
	Fechar braço paralelo. Tecla de atalho:] Outro recurso interessante é a possibilidade de fechar o braço paralelo com o mouse. Para isto, clique sobre a conexão paralela em aberta e arraste para a posição onde deseja fechar o braço. Uma indicação em vermelho mostra se é possível fechar o braço na posição desejada. A tecla P reabre braço paralelo fechado.
	Inserir bloco de função (<i>Function</i> ou <i>Function Block</i>). Tecla de atalho: B (abre janela de blocos disponíveis).
	Inserir saída (padrão: saída normal). Para escolher outro tipo de saída, clique na seta do botão para abrir o menu de opções. Tecla de atalho: O (insere saída normal) e Q (abre lista de saídas). Os outros tipos de saída são acessados através desta barra de ferramentas.
	Inserir comentário na linha corrente. Durante a edição do comentário, pressione Ctrl+ENTER para mudar de linha. ESC cancela a edição do comentário, ENTER confirma e sai de edição. Tecla de atalho: C
	Inserir LABEL em uma linha. Teclas de atalho: L
	Inserir JUMP para um <i>Label</i> . Teclas de atalho: J



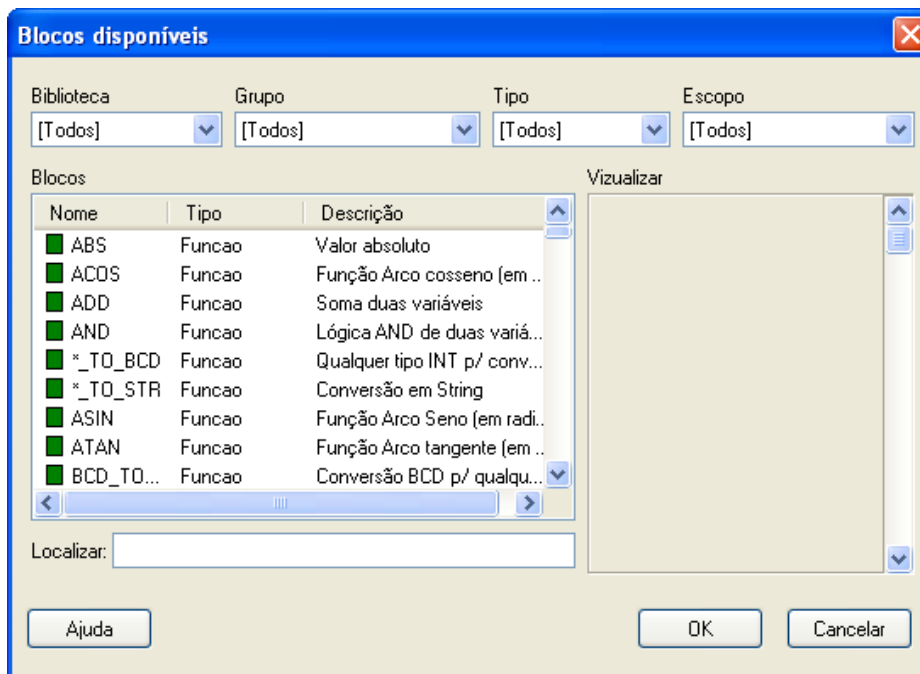
Inserir chamada (*CALL*) para uma sub-rotina.

Teclas de atalho: G

OBS: Uma sub-rotina é um Programa (*Program*) criado que não tem nenhuma Tarefa (*Task*) associada a ele.

Inserindo Blocos e Funções

Para inserir uma Função ou Bloco Funcional, pressione a tecla  da barra de ferramentas *ladder*. A janela abaixo é aberta para escolher um dos blocos disponíveis.



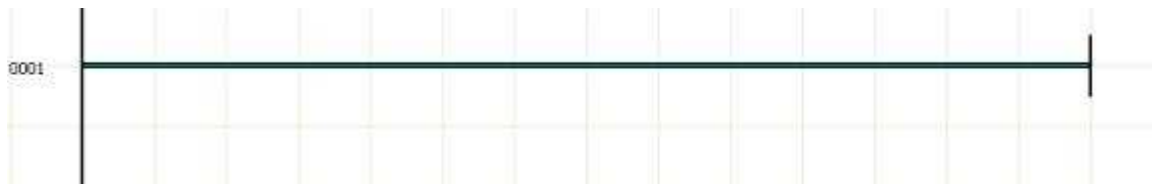
Ferramentas de filtro para encontrar uma instrução:

Biblioteca	O filtro por biblioteca de instruções separa as instruções Atos das instruções IEC61131 (bibliotecas padrão disponíveis na instalação do software). Outras bibliotecas podem ser criadas para agrupar as funções e blocos funcionais de usuário de modo a facilitar sua localização nesta janela.
Grupo	O filtro por grupo separa as instruções pelo tipo de execução. Ex: Aritméticas, Logarítmicas, Comparação, Conversão, etc.
Tipo	O filtro "Tipo" pode ser definido para mostrar funções, blocos funcionais ou ambos.
Localizar	Digite o nome da instrução, ou parte do nome para esta ferramenta de filtro mostrar na janela de blocos somente as instruções que comecem com as letras digitadas. Muito útil quando já se conhece o nome da instrução. Ao encontrar a instrução desejada, pressione ENTER para automaticamente inserí-la no <i>ladder</i> .

Para obter informações sobre o funcionamento de alguma instrução, pressione o botão "**Ajuda**" com o foco na instrução desejada.

Ao criar um projeto (Menu *Arquivo*, opção *Novo*), clique sobre o item **Program0** no **gerenciador do projeto**. Isso abrirá uma janela *ladder* para edição.

Na barra de ferramentas *ladder*, clique no botão  para inserir uma nova linha *ladder* no programa (ou utilize um atalho de teclado pelo *ladder*: tecla **I**).



Várias linhas podem ser inseridas, sequencialmente, durante a edição do programa (como mostrado abaixo).



O foco do cursor de edição *ladder* é identificado por uma linha verde (neste caso, na linha 2).

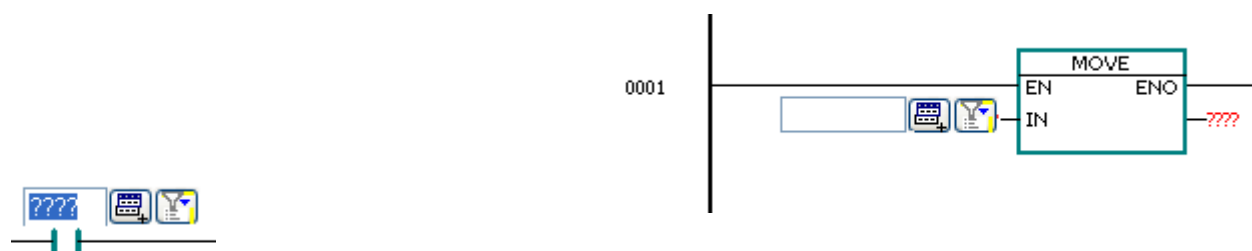
Ao compilar o projeto, deve-se observar se não existem linhas em branco ou somente com contatos declarados. O compilador identifica a linha com problema informando qual o defeito

Variáveis no ladder

Adicionando variáveis no ladder

Através desse recurso do Atos A1 Soft é possível adicionar variáveis locais ou globais diretamente no ladder, sem que seja necessário acessar a janela de Gerenciador de Variáveis Globais ou a Guia de Variáveis Locais.

Esse recurso estará habilitado sempre que um contato ou um parâmetro de uma função ou bloco funcional estiver em edição, conforme mostrado nas imagens abaixo:



Para adicionar uma variável basta clicar no botão mostrado na figura ao lado.



O usuário pode definir o nome da variável no ladder ou na janela Adicionar/editar instância da variável.

Logo após será visualizada a janela Adicionar/editar instância da variável, conforme mostra a figura seguir.

Descrição dos campos:

- **Escopo:** Define-se qual o escopo da variável.
- **Local:** Variáveis válidas apenas no POU que foi declarada;
- **Global:** Variáveis válidas em todo o projeto. Para as variáveis Globais é necessária a configuração de mais dois campos, conforme verificamos na imagem acima.
- **Grupo:** Define o grupo que pertence a variável;
 - **I/O:** Grupo das variáveis associadas ao Hardware configurado no projeto;
 - **Sistema:** Grupo de variáveis com funcionalidades definidas pelo sistema, que podem ser utilizadas mas não modificadas;
 - **Usuário:** Grupo de variáveis livres para utilização nos programas, podendo ser modificadas.
- **Região do Tipo de Dado:** Agrupa os Tipos de dados de forma a manter os que possuem características semelhantes dentro de uma mesma região.

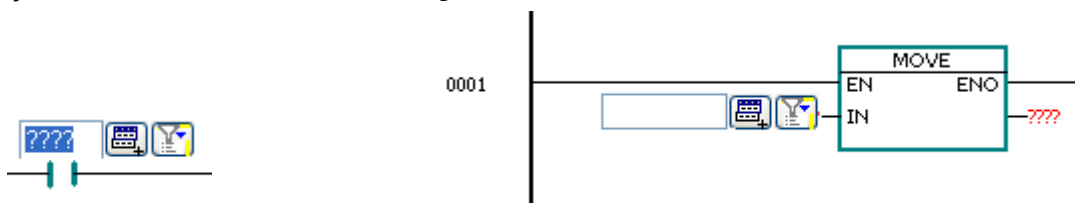
- **Nome:** Nome atribuído ao endereço da variável. Esse nome pode ser usado diretamente no programa, não sendo necessário declarar o endereço;
- **Atributo:** atribui a variável como sendo:
 - **Constante:** Variável permanece sempre com valor constante que deve ser inserido no campo Valor Inicial. Nada pode alterar o valor desta variável, tanto na programação em ladder quanto via comunicação ou supervisão do CLP;
 - **Retentiva:** Variável retém sempre o último valor armazenado quando o CLP for reinicializado;
 - **Não Retentiva:** Variável não retém o último valor armazenado quando o CLP for reinicializado, porém carrega o valor default de acordo com o tipo de dado seguindo a tabela da norma IEC 61131;
- **Endereço:** Endereço associado à variável, segundo a norma IEC61131.
- **Valor Inicial:** Define o valor que a variável será inicializada;
- **Tipo de dado:** Define qual o tipo de dado correspondente a cada variável dentro de cada região;
- **Descrição:** Não é obrigatório, um campo para descrever a variável em questão;

Na parte inferior da janela verifica-se uma região de visualização, onde são mostradas todas as variáveis já declaradas de acordo com as configurações definidas na janela.

Filtrando Variáveis no Ladder

Através desse recurso do Atos A1 Soft é possível filtrar/localizar diretamente no ladder, variáveis locais ou globais declaradas anteriormente.

Esse recurso estará habilitado sempre que um contato ou um parâmetro de uma função ou bloco funcional estiver em edição, conforme mostrado nas imagens abaixo:

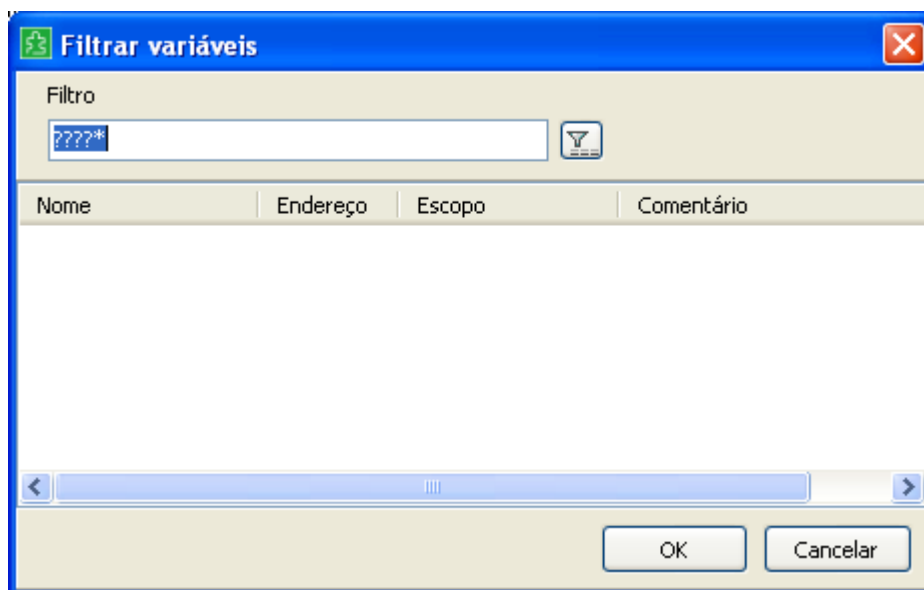


Para localizar uma variável basta clicar no botão mostrado na figura ao lado.



A variável a ser localizada pode ser um contato já configurado no ladder ou ser definida na janela Filtrar variáveis.

Logo após será visualizada a janela Filtrar variáveis, conforme mostra a figura abaixo:



Descrição dos campos:

- **Filtro:** Define-se o nome da variável a ser localizada.



O símbolo (*) é utilizado para localizar todas as variáveis que possuem pelo menos parte do conteúdo definido no campo Filtro.

Na região abaixo do campo Filtro são listadas todas as variáveis encontradas, mostrando os seguintes dados de cada uma.

- **Nome:** Nome atribuído à variável;
- **Endereço:** Endereço associado à variável global, segundo a norma [IEC61131](#).



Como as variáveis locais não possuem endereços associados, o campo será preenchido por [--].

- **Escopo:** Define-se qual o escopo da variável.
 - **Local:** Variáveis válidas apenas no POU que foi declarada;
 - **Global:** Variáveis válidas em todo o projeto.
- **Comentário:** Se definido alguma descrição à variável, esta será mostrada nesse campo.

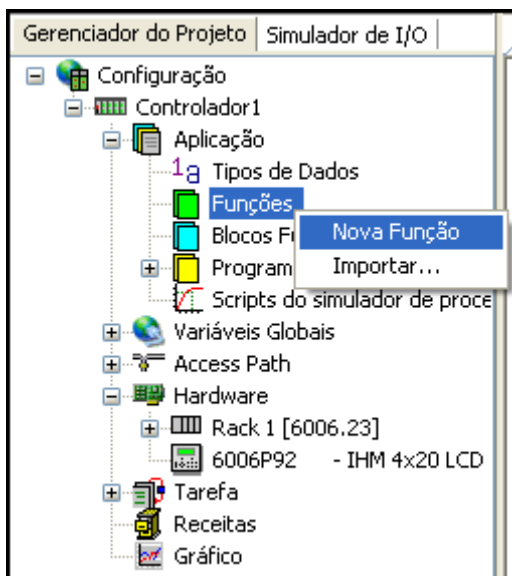
Função (Function)

Funções (*Functions*) são estruturas que permitem ao usuário separar seus programas em blocos. Com a utilização das funções os programas podem ser otimizados, reduzindo o número de linhas na programação do ladder.

Funções (*Functions*) podem ter **apenas uma saída (retorno)**, sem considerar a saída ENO para controle de execução e até 31 parâmetros de entrada. As funções não são instanciáveis, isto é, só existem em tempo de execução, não sendo necessário um identificador para alocação de memória para dados.

Os parâmetros declarados nas funções devem ser somente de escopo local, ou seja, são aquelas que só têm validade dentro do bloco no qual são declaradas. A característica que torna as variáveis locais tão importantes é justamente a de serem exclusivas do bloco. Com isso, é possível ter várias funções com uma variável local chamada x, por exemplo, e elas não apresentarão conflito entre elas.

Inserindo Função



Para criar uma Função no Atos A1 Soft, selecione com o cursor do mouse o item **Funções** na área do **Gerenciador do projeto**. Clique com o botão direito do mouse para exibir a caixa de informação e marque a opção **Nova Função**. Caso seja necessário excluir a Função inserida, marque a opção **Apagar**.

Depois de inserida uma Função, a guia de **Propriedades** irá exibir informações a respeito da mesma, tais como, nome, linguagem em que foi programada e quantidade de linha do programa. Para visualizar esta guia novamente, basta selecionar com o cursor do mouse a função desejada.

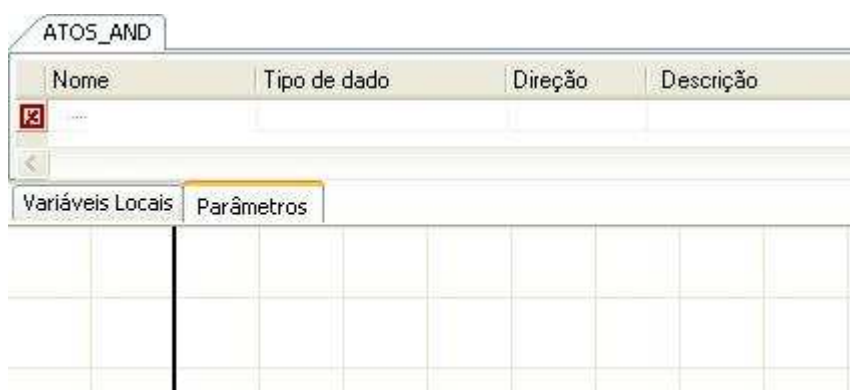
Para nomear a função, selecione-a com o cursor do mouse, clique com o botão direito e marque a opção **Renomear** ou apenas selecione ícone da função com o cursor.

Programação da Função

Para iniciar a programação da Função no Atos A1 Soft, selecione com o cursor do mouse a função na área do **Gerenciador do projeto**. Clique com o botão direito do mouse para exibir a caixa de informação e marque a opção **Abrir** ou apenas dando um duplo clique no ícone da função. Automaticamente será exibida uma aba na área de programação com o nome da função criada. Esta aba é composta por:

- Aba de declaração dos parâmetros;
- Aba de declaração de variáveis locais;
- Região de programação em ladder da função.

O exemplo abaixo mostra uma função criada para realizar a lógica AND.



Declaração de parâmetros

Os parâmetros declarados podem ter um total de 32 elementos além da entrada EN e da saída ENO. Obrigatoriamente um dos parâmetros deve ser uma saída, pois a função necessita deste parâmetro de retorno.

Os campos de declaração do parâmetro são:

- Nome do parâmetro com até 6 caracteres;
- Tipo de dado do parâmetro;
- Direção - indica se o parâmetro é de entrada (IN) ou de saída (OUT);
- Descrição do parâmetro.

O exemplo abaixo mostra a declaração dos parâmetros da função.



Declaração de variáveis locais

Variáveis locais são válidas apenas dentro da função no qual são declaradas. Com isso, é possível inserir mais de uma função que comporte os mesmos nomes das variáveis locais.

Para a declaração de variáveis locais são necessários preencher os campos:

- Nome da variável local;
- Tipo de dado da variável;
- Atributo - atribui a variável como sendo:
 - **Constante:** Variável permanece sempre com valor constante que deve ser inserido no campo **Valor Inicial**. Nada pode alterar o valor desta variável, tanto na programação em ladder quanto via comunicação ou supervisão do CLP;
 - **Retentiva:** Variável retém sempre o último valor armazenado quando o CLP for reinicializado;
 - **Não Retentiva:** Variável não retém o último valor armazenado quando o CLP for reinicializado, porém carrega o valor default de acordo com o tipo de dado seguindo a **tabela da norma IEC 61131**;
- Valor inicial da variável;
- Descrição da variável.

A figura abaixo mostra os campos de declaração das variáveis locais.

The screenshot shows the 'ATOS_AND' window with a table for declaring local variables. The table has five columns: 'Nome', 'Tipo de dado', 'Atributos', 'Valor Inicial', and 'Descrição'. Below the table, there are two tabs: 'Variáveis Locais' (selected) and 'Parâmetros'.

Nome	Tipo de dado	Atributos	Valor Inicial	Descrição

At the bottom of the window, there are two tabs: 'Variáveis Locais' and 'Parâmetros'.

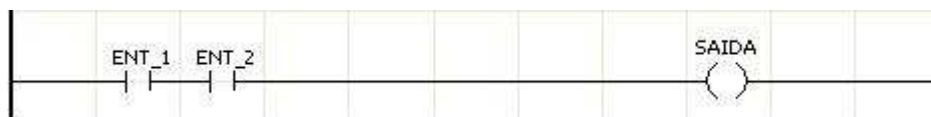
Tabela da norma IEC 61131

Tipo de dado	Valor inicial
BOOL, INT e DINT	0
UINT e UDINT	0
BYTE, WORD e DWORD	0
REAL	0.0
TIME	T#0S
DATE	D#0001-01-01
TIME_OF_DAY	TOD#00:00:00
DATE_AND_TIME	DT#0001-01-01-00:00:00
STRING	" " (String vazia)

Programação no Ladder

Depois de declarados todos os parâmetros e variáveis locais inicia-se a programação da função na região do ladder.

O exemplo da figura abaixo mostra a lógica no ladder para a função ATOS_AND criada para exemplo de função.

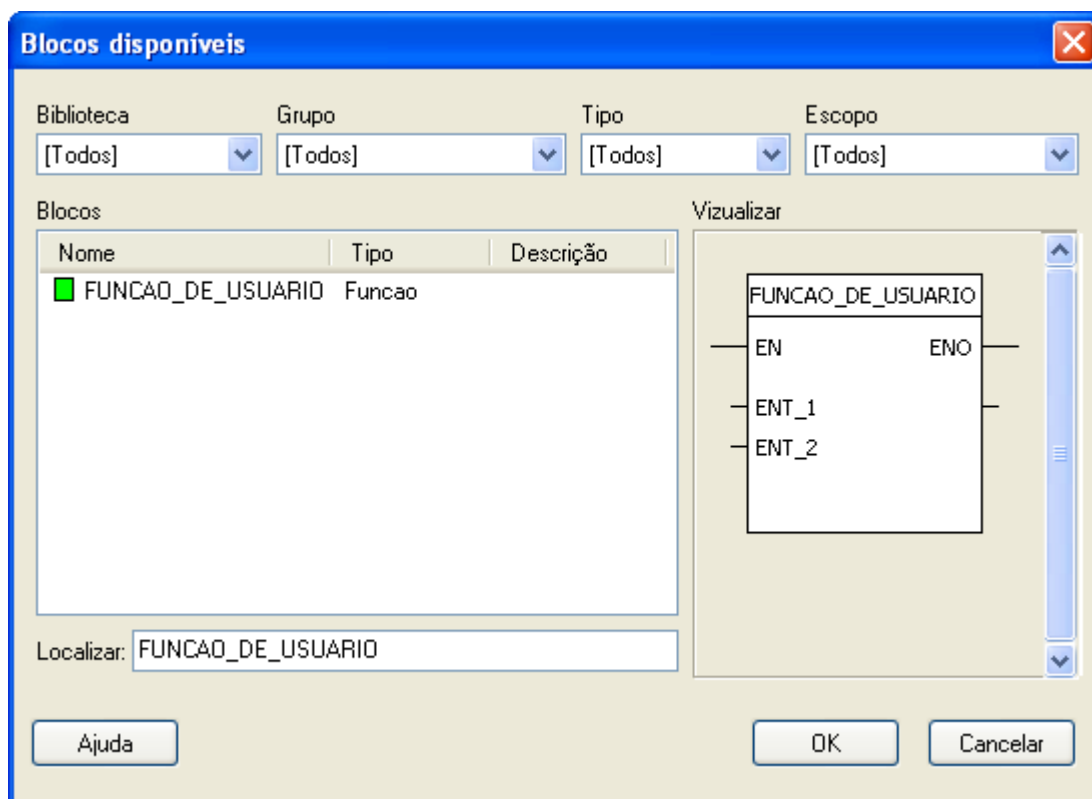


A programação da função no ladder pode utilizar: lógica de contatos e funções de sistema do Atos A1 Soft.

O Atos A1 Soft possibilita a compilação individual de cada função. Para fazer a compilação, selecione o ícone da função na área do **Gerenciador do projeto** com o curso do mouse, clique com o botão direito e selecione a opção **Compilar**. Todas as mensagens serão exibidas na guia **Mensagens**.

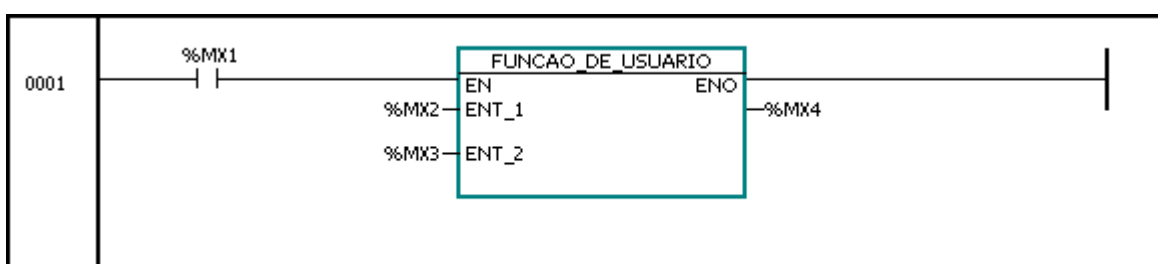
Utilizando Função

Para utilizar a nova função criada no projeto basta localizá-la no item Insere Função ou Bloco Funcional. A janela exibe o nome da função e a figura do bloco montado.



Depois de inserido no ladder, o bloco da função funciona com tipos de dados que foram declarados nos parâmetros e executa a função quando a entrada EN for habilitada.

A figura abaixo é um exemplo da função no ladder.



Depois de inserir a função no programa (ladder) e em seguida for alterada alguma característica dos parâmetros da função, deve-se atualizar as instâncias da função. Para isso, selecione a função no programa com o cursor do mouse, clique com o botão direito e marque a opção **Ajustar instância ao protótipo**. Depois de ter feito o ajuste das instâncias, o usuário deve validar o ajuste, selecionando a função no programa com o cursor do mouse, botão direito, opção **Validar ajuste da instância**.

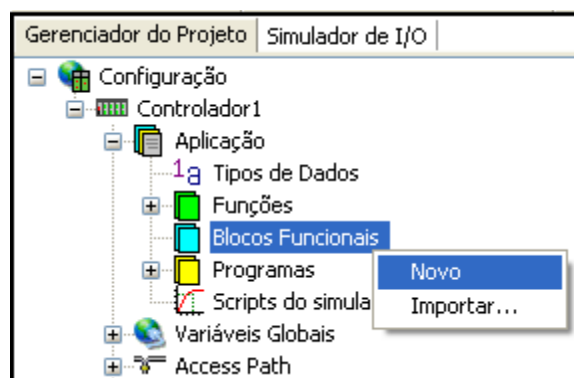
Bloco Funcional (*Function Block*)

Blocos Funcionais (*Function Blocks*) são estruturas que permitem ao usuário separar seus programas em blocos. Com a utilização dos blocos funcionais os programas podem ser otimizados, reduzindo o número de linhas na programação do ladder.

Blocos Funcionais (*Function Blocks*) podem ter até 32 parâmetros entre saídas e entradas, sem considerar a saída ENO e a entrada EN. Os blocos funcionais são instanciáveis, isto é, existem em tempo de execução ou não, sendo necessário um identificador para alocação de memória para dados.

Os parâmetros declarados nos blocos funcionais podem ser de escopo local e global. A característica que torna as variáveis locais tão importantes é justamente a de serem exclusivas do bloco. Com isso, é possível ter vários blocos funcionais com uma variável local chamada **x**, por exemplo, e elas não apresentarão conflito entre elas.

Inserindo Bloco Funcional



Para criar um bloco funcional no Atos A1 Soft, selecione com o cursor do mouse o item **Blocos Funcionais** na área do **Gerenciador do projeto**. Clique com o botão direito do mouse para exibir a caixa de informação e marque a opção **Novo**. Caso seja necessário excluir o bloco inserido, marque a opção **Apagar**.

Depois de inserido o bloco funcional, a guia de **Propriedades** irá exibir informações a respeito do mesmo, tais como, nome, linguagem em que foi programado e quantidade de linhas do programa. Para visualizar esta guia novamente, basta selecionar com o cursor do mouse o bloco desejado.

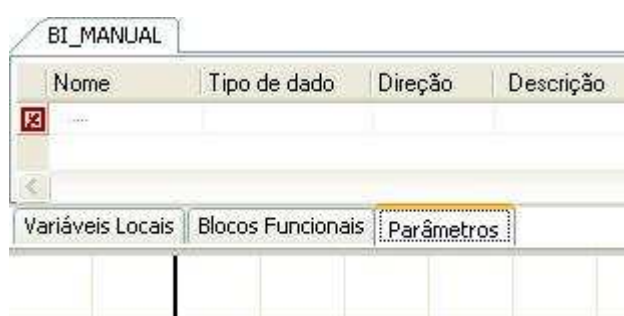
Para nomear o bloco, selecione-o com o cursor do mouse, clique com o botão direito e marque a opção **Renomear** ou apenas selecione ícone do bloco com o cursor.

Programação do Bloco Funcional

Para iniciar a programação do bloco funcional no Atos A1 Soft, selecione com o cursor do mouse o bloco na área do **Gerenciador do projeto**. Clique com o botão direito do mouse para exibir a caixa de informação e marque a opção **Abrir** ou apenas dando um duplo clique no ícone do bloco. Automaticamente será exibida uma aba na área de programação com o nome do bloco criado. Esta aba é composta por:

- Aba de declaração dos parâmetros;
- Aba de declaração de variáveis locais;
- Aba de identificação dos blocos funcionais de sistema utilizados na programação;
- Região de programação em ladder do bloco funcional.

O exemplo abaixo mostra um bloco funcional criado para realizar um controle BI MANUAL.



Declaração de parâmetros

Os parâmetros declarados podem ter um total de 32 elementos além da entrada EN e da saída ENO.

Os campos de declaração do parâmetro são:

- Nome do parâmetro com até 6 caracteres;
- Tipo de dado do parâmetro;
- Direção - indica se o parâmetro é de entrada (IN) ou de saída (OUT);
- Descrição do parâmetro.

O exemplo abaixo mostra a declaração dos parâmetros da função.



Declaração de variáveis locais

Variáveis locais são válidas apenas dentro do bloco funcional no qual são declaradas. Com isso, é possível inserir mais de um bloco que comporte os mesmos nomes das variáveis locais.

Para a declaração de variáveis locais são necessários preencher os campos:

- Nome da variável local;
- Tipo de dado da variável;
- Atributo - atribui a variável como sendo:
 - **Constante:** Variável permanece sempre com valor constante que deve ser inserido no campo **Valor Inicial**. Nada pode alterar o valor desta variável, tanto na programação em ladder quanto via comunicação ou supervisão do CLP;
 - **Retentiva:** Variável retém sempre o último valor armazenado quando o CLP for reinicializado;
 - **Não Retentiva:** Variável não retém o último valor armazenado quando o CLP for reinicializado, porém carrega o valor default de acordo com o tipo de dado seguindo a **tabela da norma IEC 61131**;
- Valor inicial da variável;
- Descrição da variável.

A figura abaixo mostra os campos de declaração das variáveis locais.

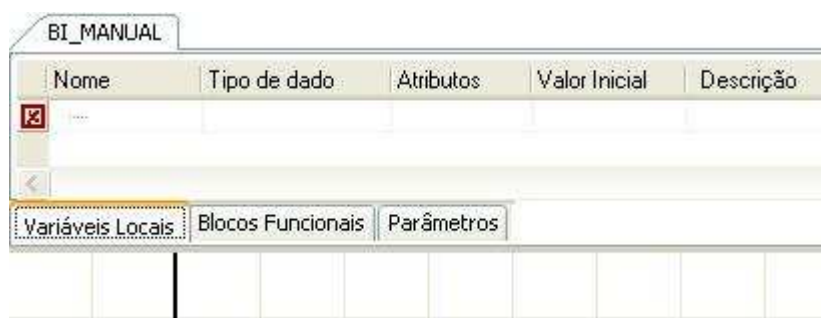


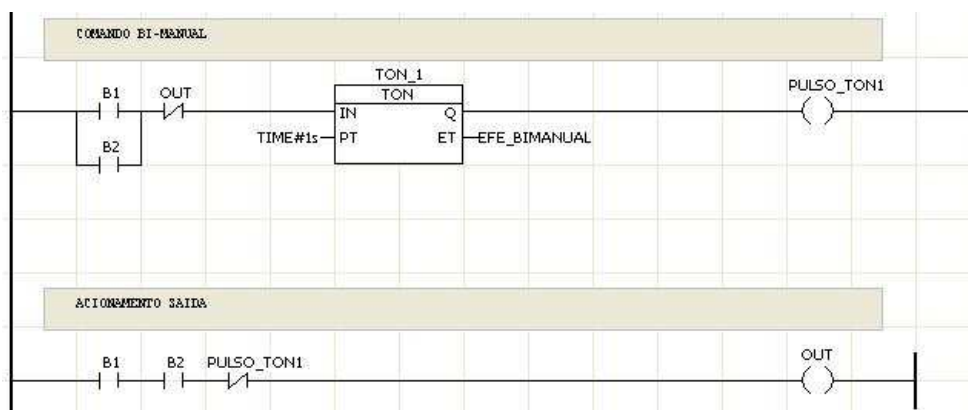
Tabela da norma IEC 61131

Tipo de dado	Valor inicial
BOOL, INT e DINT	0
UINT e UDINT	0
BYTE, WORD e DWORD	0
REAL	0.0
TIME	T#0S
DATE	D#0001-01-01
TIME_OF_DAY	TOD#00:00:00
DATE_AND_TIME	DT#0001-01-01-00:00:00
STRING	" " (String vazia)

Programação no Ladder

Depois de declarados todos os parâmetros e variáveis locais inicia-se a programação do bloco funcional na região do ladder.

O exemplo da figura abaixo mostra a lógica no ladder para o bloco BI_MANUAL criado para exemplo.

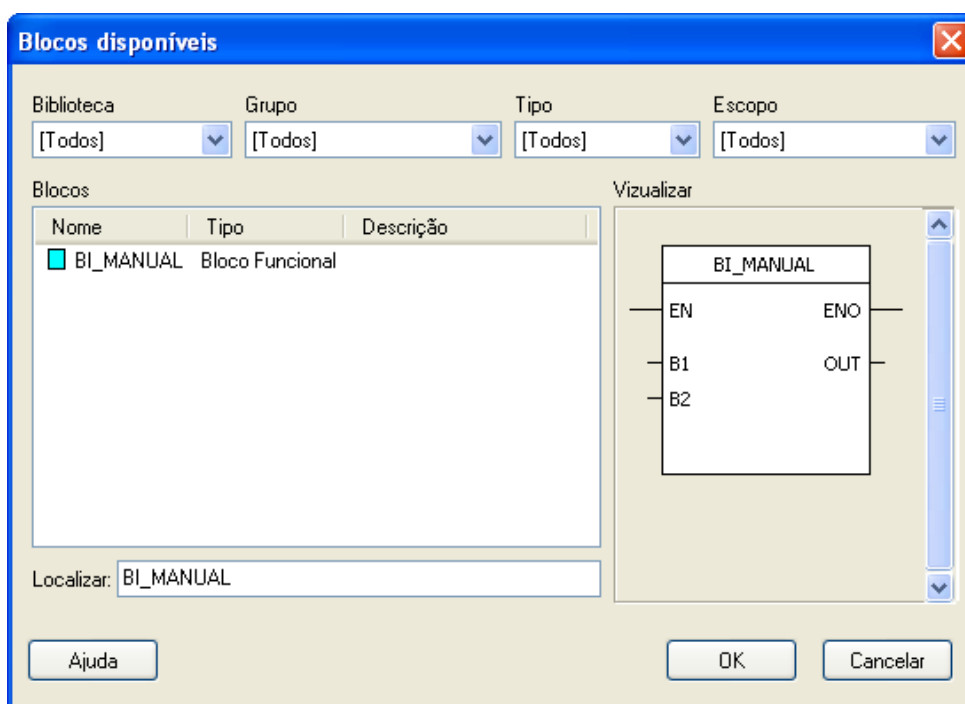


A programação do bloco no ladder pode utilizar: lógica de contatos, funções e blocos funcionais de sistema do Atos A1 Soft.

O Atos A1 Soft possibilita a compilação individual de cada bloco. Para fazer a compilação, selecione o ícone do bloco na área do **Gerenciador do projeto** com o curso do mouse, clique com o botão direito e selecione a opção **Compilar**. Todas as mensagens serão exibidas na guia **Mensagens**.

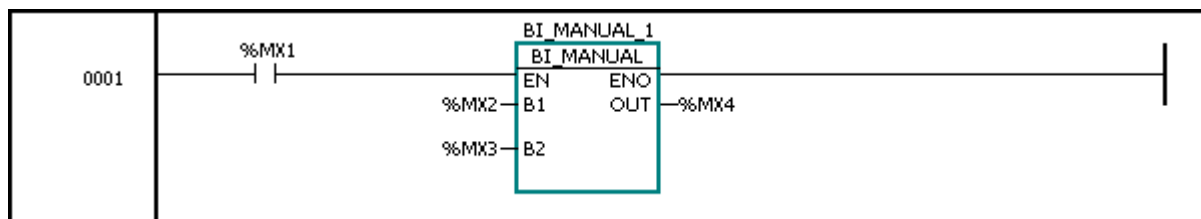
Utilizando Bloco Funcional

Para utilizar o novo bloco funcional criado no projeto basta localizá-lo no item **Inserir Função ou Bloco Funcional**. A janela exibe o nome do bloco funcional e a figura do bloco montado.



Depois de inserido no ladder, o bloco funcional funciona com tipos de dados que foram declarados nos parâmetros e é executado quando a entrada EN for habilitada.

A figura abaixo é um exemplo do bloco funcional no ladder.



Depois de inserir bloco no programa (ladder) e em seguida for alterada alguma característica dos parâmetros do bloco funcional, deve-se atualizar as instâncias do bloco. Para isso, selecione-o no programa com o cursor do mouse, clique com o botão direito e marque a opção **Ajustar instância ao protótipo**. Depois de ter feito o ajuste das instâncias, o usuário deve validar o ajuste, selecionando o bloco no programa com o cursor do mouse, botão direito, opção **Validar ajuste da instância**.

CAPÍTULO 5

Access Path

Access Path é um recurso das CPUs Atos para comunicação em rede com outros dispositivos, possibilitando a troca de informações.

A comunicação Access Path é o mecanismo através do qual um CLP pode atuar como mestre de rede. Consiste em uma lista de frames executada pelo CLP ciclicamente, por ativação de uma variável booleana ou mesmo por uma base de tempo pré-estabelecida individualmente nas configurações de cada frame.

A comunicação dos frames é realizada em paralelo com o processamento do programa de usuário. Nestes frames, são programadas a leitura e a escrita, pelo CLP mestre, de variáveis nos dispositivos escravos de rede.

A programação Access Path é útil, por exemplo, no transporte de alarmes das estações onde o programador terá, além das informações do processo controlado pelo mestre, as informações das estações supervisionadas.

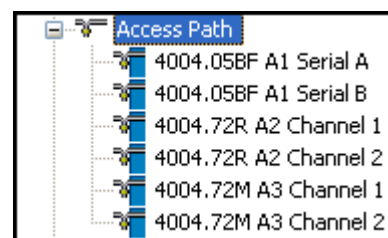
As CPUs Atos possibilitam utilizar dois tipos de protocolos de comunicação simultâneos para a trocar dados entre si: **APR03** ou **Modbus RTU**. Utilizando-se os dois protocolos suportados, é possível construir uma rede mestre-escravo de até 31 pontos por canal. Em ambos os casos, **apenas um** dos dispositivos conectados a rede **atuará como mestre**.

Canais de comunicação Protocolos de comunicação

RS232 (Canal A)	RS485 (Canal B)	RS232 (Canal A)	RS485 (Canal B)
Mestre	Mestre	APR03	APR03
Slave	Mestre	APR03	Modbus
Mestre	Slave	Modbus	APR03
Slave	Slave	Modbus	Modbus

Configuração dos canais

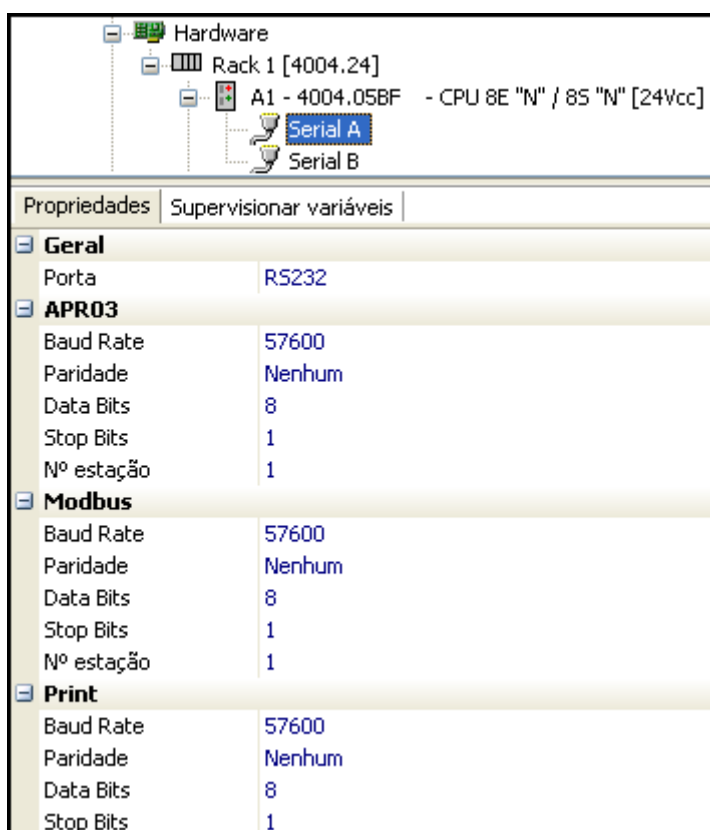
Quando uma CPU ou um módulo de comunicação slave (por exemplo, 4004.72R) for inserido na configuração de **Hardware**, automaticamente o item **Access Path** (🔧 Access Path) exibe um ícone do(s) módulo(s), conforme exemplo da figura ao lado, com os canais de comunicação para a configuração na área do **Gerenciador de Projeto**.



Para a CPU Atos é necessário configurar a estação e a taxa de comunicação dos protocolos (APR03 ou Modbus) de cada canal (**RS232** ou **RS485**), selecionado no item **Hardware** na área do Gerenciador de Projeto.

RS232 – Serial A

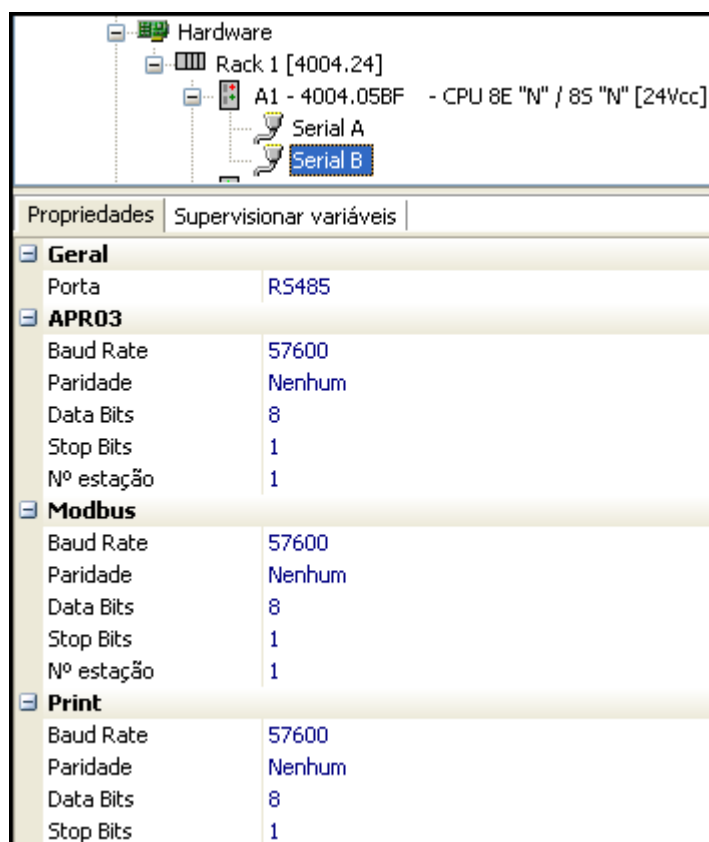
Exemplo:



Utilizando 8 bits no campo **Data Bits** tanto para APR03 e Modbus, não será possível configurar paridade (par ou ímpar) com **stop bit** igual a 2, ou seja, se desejar utilizar paridade, o número de stop bit deve ser igual a 1 ou alterar o valor no campo **Data Bits** para 7, podendo assim configurar paridade e 2 stop bits.

RS485 – Serial B

Exemplo:

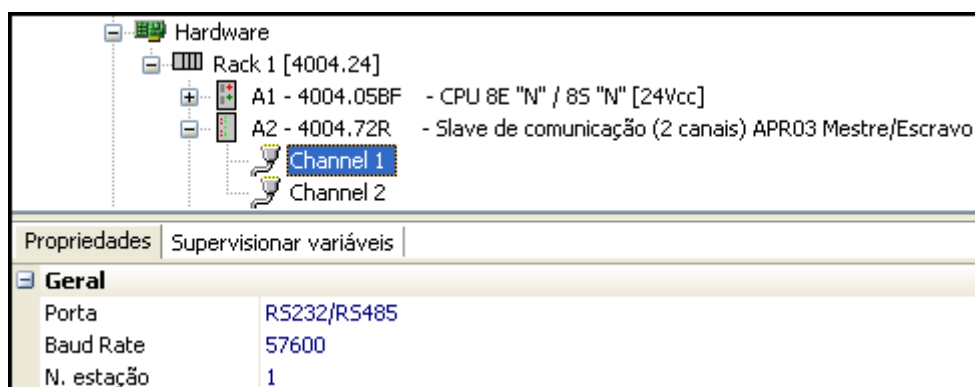


Utilizando 8 bits no campo **Data Bits** tanto para APR03 e Modbus, não será possível configurar paridade (par ou ímpar) com **stop bit** igual a 2, ou seja, se desejar utilizar paridade, o número de stop bit deve ser igual a 1 ou alterar o valor no campo **Data Bits** para 7, podendo assim configurar paridade e 2 stop bits.

Para os módulos de comunicação slave que utilizam os protocolos **APR03** ou **Modbus**, a configuração da estação e da taxa de comunicação é feita da mesma forma na área do **Gerenciador de Projeto** no item **Hardware**.

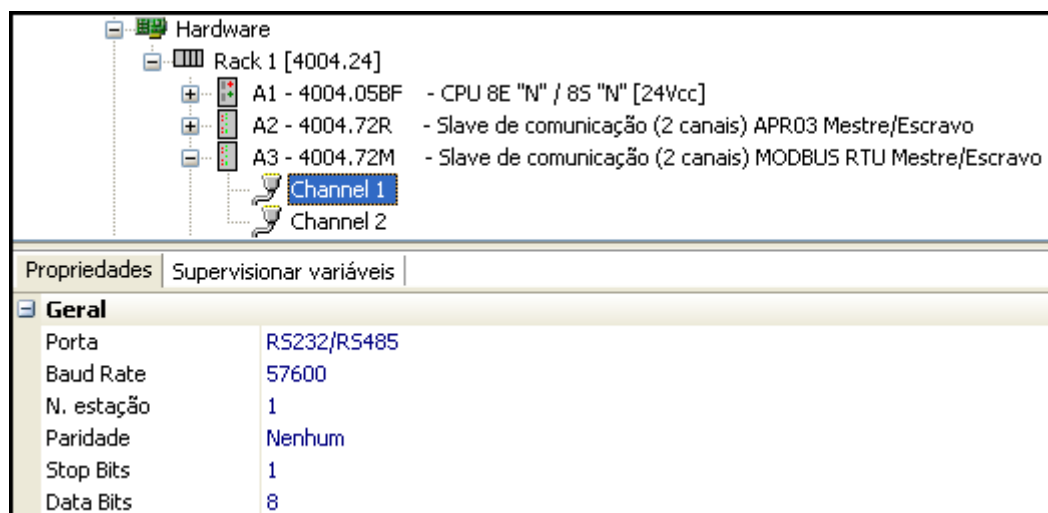
APR03 – Slave de comunicação

Exemplo:



Modbus – Slave de comunicação

Exemplo:

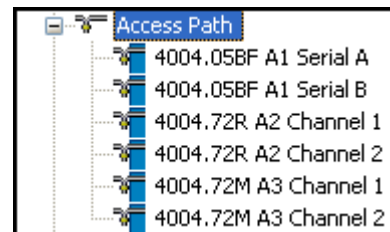


Modo Slave

- Definindo modo de comunicação SLAVE

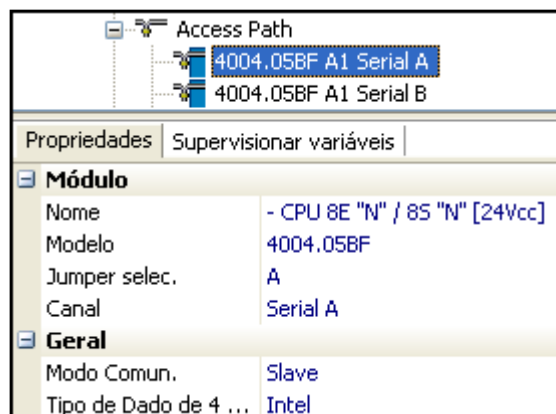
A configuração dos canais de comunicação para o **modo Slave (escravo)**, tanto para CPU quanto para os módulos de comunicação slave é realizada no item **Access Path**.

Selecione o canal de comunicação com o cursor do mouse, na guia de **Propriedades** marque no campo **Modo de Comunicação** a opção **Slave**.



Para os canais de comunicação da CPU também é necessário selecionar o tipo de protocolo (APR03 ou Modbus) no campo **Protocolo**.

- Modo slave para protocolo **APR03**:
 - Protocolo**: APR03
 - Modo Comun.:** Slave
- Modo slave para protocolo **Modbus**
 - Protocolo**: Modbus
 - Modo Comun.:** Slave
 - Tipo de dado Real**: Neste campo, coloque o formato de comunicação das variáveis do tipo REAL.



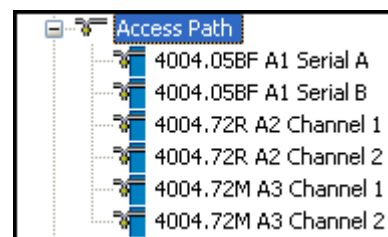
No modo slave não é necessário configurar nenhum frame de comunicação, pois essa é uma característica do modo mestre, nestes frames são configurados as funções que serão executadas na unidade remota (slave).

A comunicação **Modbus** é habilitada através das variáveis de sistema **MODBUS_A (RS232)** e/ou **MODBUS_B (RS485)**.

Modo Mestre

A configuração dos canais de comunicação para o **modo Mestre**, tanto para CPU quanto para os módulos de comunicação slave é realizada no item **Access Path** na área do **Gerenciador do Projeto**.

Selecione o canal de comunicação com o cursor do mouse, na guia de **Propriedades** marque no campo **Modo de Comunicação** a opção **Mestre**. Lembrando que, para os canais de comunicação da CPU também é necessário selecionar o tipo de protocolo (APR03 ou Modbus) no campo **Protocolo**.



CPUs no **modo Mestre** necessitam do acionamento das **variáveis de sistema MASTER_A** para habilitar modo mestre no canal **RS232** ou **MASTER_B** para habilitar modo mestre no canal **RS485**. A localização dessas variáveis de sistema pode ser visualizada no item **Variáveis Globais**.



Não é possível acionar o modo supervisão do Atos A1 Soft no canal de comunicação que estiver configurado como mestre e ativo. Para enviar o programa de usuário pelo canal é necessário desacionar a variável de sistema MASTER_A ou MASTER_B de acordo com o canal de comunicação ou mudar a posição da chave no frontal da CPU de RUN para **PROG**.

Descrição dos campos:

Geral	
Protocolo	APR03
Modo Comun.	Mestre
Habilitado	true
Timeout (ms)	200
Qtde Alarmes	1
Variável inicial de alarme	
Variável inicial de falha	

- **Protocolo:** Define o protocolo de comunicação do canal em APR03 ou Modbus;
- **Modo Comun.:** Define a estação como Mestre de rede ou escravo (Slave);
- **Enabled:** Em TRUE habilita o Access Path sem apagar os frames configurados;
- **Timeout (ms):** Tempo em milisegundos que a CPU aguarda a resposta de um frame antes de sinalizar algum tipo de falha;

- **Variável inicial de alarme:** Variável do **tipo booleana**. Neste campo pode ser inserido o endereço (Exemplo %MX1) ou o nome dado à variável (Exemplo Alarme_1).

Esta variável está relacionada com o campo **Qtde Alarmes**, ou seja, com 25 alarmes configurados serão fixadas 25 variáveis a partir do endereço inicial de alarme.

Quando a primeira estação apresentar algum erro a primeira variável inicial de alarme será acionada. Se a décima estação apresentar algum erro a décima variável de alarme será acionada.

Exemplo:

Na rede de comunicação estão dispostos 9 dispositivos slaves comunicando com protocolo APR03.

- **Variável inicial de alarme:** %MX1 (Booleano). Aciona quando a estação correspondente apresenta alguma falha;
- **Qtde Alarmes:** 9 (Tipo de dado UINT);
- **Variável inicial de falha:** %MW1 (Tipo de dado UINT). Exibe o **código de erro** apresentado na estação.

Estação	1°	2°	3°	4°	5°	6°	7°	8°	9°
Variável inicial de alarme	%MX1	%MX2	%MX3	%MX4	%MX5	%MX6	%MX7	%MX8	%MX9
Variável inicial de falha	%MW1	%MW2	%MW3	%MW4	%MW5	%MW6	%MW7	%MW8	%MW9

- **Qtde Alarmes:** Variável do **tipo UINT**, que relaciona o número de estações na rede com a quantidade de alarmes, ou seja, com 25 slaves na rede deverão ser colocados 25 alarmes. O uso de alarmes é opcional, porém promove agilidade e praticidade na identificação de falhas em uma ou mais estações;
- **Variável inicial de falha:** Variável do **tipo UINT**, neste campo pode ser inserido o endereço (Exemplo %MW1) ou o nome dado à variável (Exemplo FALHA_1).

Esta variável está relacionada com o campo **Qtde Alarmes**, ou seja, com 25 alarmes configurados serão fixadas 25 variáveis a partir do endereço inicial de falha.

Quando a primeira estação apresentar algum erro a primeira variável inicial de falha irá exibir um código que relaciona o tipo de erro. Se a décima estação apresentar algum erro a décima variável de falha irá exibir qual o tipo de falha.

Exemplo:

Na rede de comunicação estão dispostos 9 dispositivos slaves comunicando com protocolo Modbus.

- **Variável inicial de alarme:** %MX1 (Booleano). Aciona quando a estação correspondente apresenta alguma falha;
- **Qtde Alarmes:** 9 (Tipo de dado UINT);
- **Variável inicial de falha:** %MW1 (Tipo de dado UINT). Exibe o **código de erro** apresentado na estação.

Estação	1°	2°	3°	4°	5°	6°	7°	8°	9°
Variável inicial de alarme	%MX1	%MX2	%MX3	%MX4	%MX5	%MX6	%MX7	%MX8	%MX9
Variável inicial de falha	%MW1	%MW2	%MW3	%MW4	%MW5	%MW6	%MW7	%MW8	%MW9

Códigos de erro ou falha

Códigos APR03

Se **Variável de Alarme** exibir:

1	Erro de Timeout;
2	Erro no tamanho do frame (Quantidade de dados não suportado);
3	Erro de função;
4	Erro de endereço;
5	Tamanho de bloco inválido;
6	Erro de Checksum;
7	Tamanho de buffer no mestre não suportado.

Códigos Modbus

Se **Variável de Alarme** exibir:

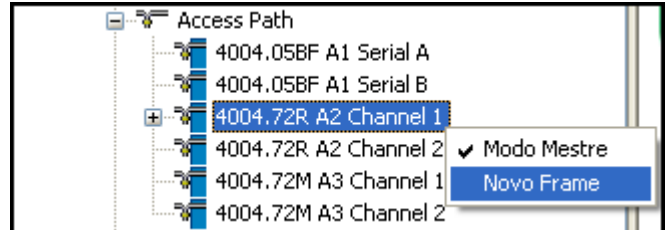
1	Função ilegal;
2	Endereço ilegal;
3	Dado ilegal;
9	Erro de Timeout (Falha na comunicação);
10	Erro de CRC;
11	Erro no tamanho do frame (Quantidade de dados não suportado);
12	Erro de resposta

Configuração de Frames

Frames de comunicação

Para adicionar um frame no Access Path selecione com o cursor do mouse o **canal de comunicação** desejado, clique com o botão direito e escolha a opção **Novo frame**.

Para apagar um frame, selecione o frame desejado com o cursor do mouse, clique com o botão direito e escolha a opção **Apagar**.



A quantidade máxima de frames configurados para cada canal são **1000 frames**.



Frames só poderão ser configurados no [modo mestre](#).

Propriedades do frame APR03

Propriedades		Supervisionar variáveis
Geral		
Variável Inicial		
Qtde Bytes	1	
Endereço remota (hex)	0000	
Nº estação da remota	1	
Direção	Receber da Remota	
Tempo Atualização(ms)	0	
Variável Atualização		

- **Variável Inicial:** Endereço (**Formato %M...**) ou nome dado à variável na área de **Variáveis Globais**;
- **Qtde Bytes:** Número de bytes transportados; Cada frame pode transportar 255 bytes para variáveis numéricas e 255 variáveis booleanas.
- **Endereço remota (hex):** Endereço da variável em hexadecimal da estação remota (endereços localizados no item Variáveis Globais, tabela de Endereço Físico para cada tipo de dado);

- **Nº estação da remota:** Número de estação da unidade remota, se configurado o número **0** define-se que o frame é um [frame broadcast](#);
- **Direção:** Define o fluxo de dados entre a CPU (Mestre) e a unidade remota (Slave);
 - **Receber da Remota:** Leitura de informação na SLAVE (Slave -> CPU);
 - **Enviar para Remota:** Escrita de informação na SLAVE (CPU -> Slave).
- **Tempo Atualização (ms):** Tempo de atualização do frame em milissegundos ([Ver as formas de Atualização de frame](#));
- **Variável Atualização:** Variável do **tipo booleana**. Neste campo pode ser inserido o endereço (Exemplo %MX1) ou o nome dado à variável (Exemplo Alarme_1). Quando esta variável estiver em TRUE será habilitado o frame que realizará o transporte de dados de acordo com sua configuração ([Ver as formas de Atualização de frame](#)).

Propriedades do frame Modbus

Propriedades		Supervisionar variáveis
Geral		
Variável Inicial		
Qtde de variáveis	1	
Endereço remota	0	decimal
Nº estação da remota	1	
Funcao	Nenhum Programa	
Tempo Atualização(ms)	0	
Variável Atualização		

- **Variável Inicial:** Endereço (**Formato %M...**) ou nome dado à variável na área de **Variáveis Globais**;
- **Qtde Bytes:** Número de variáveis transportadas; Cada frame pode transportar **126 variáveis (de 2 bytes)** do tipo **Register** (Numéricas) ou **256 variáveis** do tipo **Coil/Input** (Booleana).
- **Endereço remota:** Endereço da variável em hexadecimal ou decimal quando for utilizada a **posição da variável**;

Para comunicação Modbus entre CPUs, utilizar a **tabela de Posição Modbus** visível no item **Variáveis Globais** do **Gerenciador do Projeto**.

Variáveis Globais							
Grupo		Região do Tipo de Dado					
USUÁRIO		BOOL					
Endereço	Nome	Tipo ...	Atributo	Valor Inicial	Descrição	Endereço físico	Posição Modbus
%MX1		BOOL	NÃO RETENTIVO			1800	7001
%MX2		BOOL	NÃO RETENTIVO			1801	7002
%MX3		BOOL	NÃO RETENTIVO			1802	7003
%MX4		BOOL	NÃO RETENTIVO			1803	7004
%MX5		BOOL	NÃO RETENTIVO			1804	7005
%MX6		BOOL	NÃO RETENTIVO			1805	7006
%MX7		BOOL	NÃO RETENTIVO			1806	7007
%MX8		BOOL	NÃO RETENTIVO			1807	7008
%MX9		BOOL	NÃO RETENTIVO			1808	7009
%MX10		BOOL	NÃO RETENTIVO			1809	7010

- **Nº estação da remota:** Número de estação da unidade remota, se configurado o número **0** define-se que o frame é um [frame broadcast](#);
- **Função:** Função Modbus a ser executada ([Ver Funções Modbus](#));
- **Tempo Atualização (ms):** Tempo de atualização do frame em milissegundos ([Ver Atualização de frame](#));
- **Variável Atualização:** Variável do **tipo booleana**. Neste campo pode ser inserido o endereço (Exemplo %MX1) ou o nome dado à variável (Exemplo Alarme_1). Quando esta variável estiver em TRUE será habilitado o frame que realizará o transporte de dados de acordo com sua configuração ([Ver Atualização de frame](#)).

Frame Broadcast

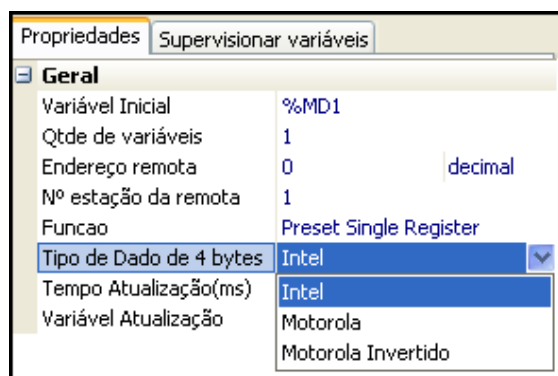
Um frame broadcast é caracterizado pela configuração do número **0** na estação remota.

Sua função é enviar os dados configurados no frame para todas as estações ligadas fisicamente à rede, ou seja, ao invés de direcionar os dados para apenas uma estação, todas receberão a informação enviada.



Essa funcionalidade só pode ser utilizada com funções de escrita.

Frame Modbus para tipo de dado de 4 bytes



Para comunicação de um tipo de dado de 4 bytes é necessário, além das configurações básicas mostradas no item "Propriedades do frame Modbus" que se configure o formato de comunicação, conforme mostra a figura ao lado.

Existem três tipos de formato de comunicação para o tipo de dado de 4 bytes em Modbus:

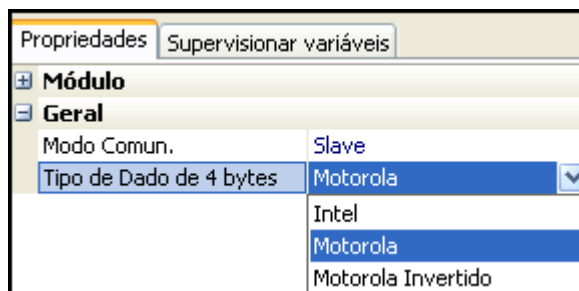
- Intel;
- Modcom;
- Modcom Invertido;

Formato de comunicação de dado tipo 4 bytes

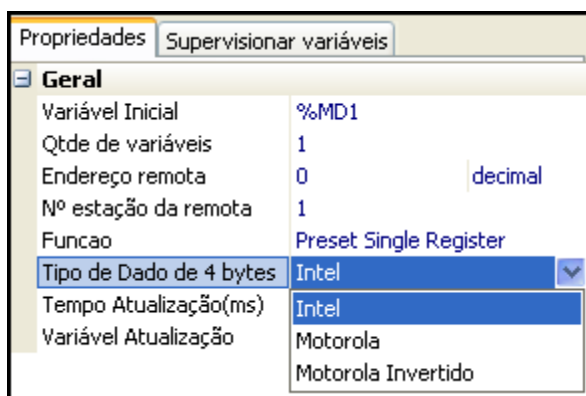
Para o protocolo Modbus, é necessário configurar o formato de comunicação do tipo de dado de 4 bytes.

Essa configuração é feita tanto no modo SLAVE, como no frame do modo mestre, conforme mostrado nas figuras abaixo.

- Modo Slave



- Frame configurado no modo Mestre



Descrição do formato de comunicação:

- **Intel**

No formato Intel de comunicação, o byte menos significativo é colocado na menor posição de memória do dado.

A figura abaixo mostra o formato Intel.



- **Modcom**

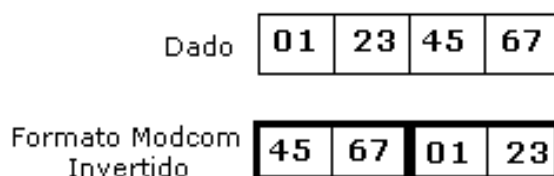
O formato Modcom utiliza para os dados de 4 bytes dois registros consecutivos, como especificado pela IEEE (*Institute of Electrical and Electronics Engineers*).

Com isso temos o dado da seguinte maneira:



- **Modcom Invertido**

O formato Modcom invertido possui o mesmo conceito do Modcom, porém envia os registros de forma invertida, conforme mostrado abaixo.



Comunicando STRUCT via Modbus

A comunicação de estruturas não é tratada da mesma maneira que os outros tipos de dados, por isso alguns detalhes devem ser observados:

- Quando uma Struct é declarada no datablock todos seus membros são tratados como registros e armazenados em words (16 bits). A posição modbus é definida de acordo com o tamanho do registro, conforme mostrado na figura abaixo:

Variáveis Globais									
Grupo		Região do Tipo de Dado							
USUÁRIO		DATA BLOCK							
Endereço	Nome	Tipo de dado	Atributo	Valor Inicial	Descrição	Access Path	Endereço físico	Posição Modbus	
%MA1		STRUCT	RETENTIVO			TRUE			
	MEMBER_1	DWORD	RETENTIVO				4800	33001	
	MEMBER_2	UINT	RETENTIVO				4804	33003	
	MEMBER_3	WORD	RETENTIVO				4806	33004	
%MA2			RETENTIVO			FALSE			

Exemplos:

MEMBER_1 – Variável do tipo DWORD (32 bits) ocupa 2 words;

MEMBER_2 – Variável do tipo UINT (16 bits) ocupa 1 word;

MEMBER_3 – Variável do tipo WORD (16 bits) ocupa 1 word.

- Como todos os membros de uma Struct são tratados como registros, as funções modbus a serem usadas devem ser:

03 - Read Holding Registers;

16 - Preset Multiple Registers;

Situações especiais

Membro Booleano:

- Os membros booleanos de uma STRUCT possuem tamanho de 8 bits. Nesta situação para uma STRUCT contendo dois membros do tipo BOOL (seqüenciais), os mesmos ocupam a mesma posição modbus, sendo alocados no MSB e LSB do registro, conforme mostrado na figura abaixo.

Variáveis Globais								
Grupo		Região do Tipo de Dado						
USUÁRIO		DATA BLOCK						
Endereço	Nome	Tipo de dado	Atributo	Valor Inicial	Descrição	Access Path	Endereço físico	Posição Modbus
- %MA1		STRUCT	RETENTIVO			TRUE		
	MEMBER_1	WORD	RETENTIVO				4800	33001
	MEMBER_2	BOOL	RETENTIVO				4802	33002 LSB
	MEMBER_3	BOOL	RETENTIVO				4803	33002 MSB
- %MA2			RETENTIVO			FALSE		

Observação: Nas situações em que existe somente um membro booleano na STRUCT, o dado é armazenado conforme indicado em sua posição Modbus, conforme mostrado na figura abaixo:

Variáveis Globais								
Grupo		Região do Tipo de Dado						
USUÁRIO		DATA BLOCK						
Endereço	Nome	Tipo de dado	Atributo	Valor Inicial	Descrição	Access Path	Endereço físico	Posição Modbus
- %MA1		STRUCT	RETENTIVO			TRUE		
	MEMBER_1	WORD	RETENTIVO				4800	33001
	MEMBER_2	BOOL	RETENTIVO				4802	33002 LSB
- %MA2			RETENTIVO			FALSE		

Nesta situação, o membro MEMBER_2, está armazenado no LSB do registro 33002. O MSB deste mesmo registro não possui nenhum valor válido, devendo ser desprezado.

Membro Byte:

- Os membros do tipo BYTE de uma STRUCT possuem tamanho de 8 bits. Nesta situação para uma STRUCT contendo dois membros do tipo BYTE (seqüenciais), os mesmos ocupam a mesma posição modbus, sendo alocados no MSB e LSB do registro, conforme mostrado na figura abaixo.

Variáveis Globais								
Grupo		Região do Tipo de Dado						
USUÁRIO		DATA BLOCK						
Endereço	Nome	Tipo de dado	Atributo	Valor Inicial	Descrição	Access Path	Endereço físico	Posição Modbus
- %MA1		STRUCT	RETENTIVO			TRUE		
	MEMBER_1	WORD	RETENTIVO				4800	33001
	MEMBER_2	BYTE	RETENTIVO				4802	33002 LSB
	MEMBER_3	BYTE	RETENTIVO				4803	33002 MSB
- %MA2			RETENTIVO			FALSE		

Observação: Nas situações em que existe somente um membro byte na STRUCT, o dado é armazenado conforme indicado em sua posição Modbus, conforme mostrado na figura abaixo:

Variáveis Globais								
Grupo		Região do Tipo de Dado						
USUÁRIO		DATA BLOCK						
Endereço	Nome	Tipo de dado	Atributo	Valor Inicial	Descrição	Access Path	Endereço físico	Posição Modbus
- %MA1		STRUCT	RETENTIVO			TRUE		
	MEMBER_1	WORD	RETENTIVO				4800	33001
	MEMBER_2	BYTE	RETENTIVO				4802	33002 LSB
- %MA2			RETENTIVO			FALSE		

Nesta situação, o membro MEMBER_2, está armazenado no LSB do registro 33002. O MSB deste mesmo registro não possui nenhum valor válido, devendo ser desprezado.

Membros Bool e Byte:

Para os membros de uma STRUCT cujos tipos de dados BOOL e BYTE estejam mesclados e seqüenciais, estes ocupam a mesma posição modbus, sendo alocados no MSB e LSB do registro, conforme mostrado na figura abaixo.

Variáveis Globais								
Grupo		Região do Tipo de Dado						
USUÁRIO		DATA BLOCK						
Endereço	Nome	Tipo de dado	Atributo	Valor Inicial	Descrição	Access Path	Endereço físico	Posição Modbus
- %MA1		STRUCT	RETENTIVO			TRUE		
	MEMBER_1	WORD	RETENTIVO				4800	33001
	MEMBER_2	BOOL	RETENTIVO				4802	33002 LSB
	MEMBER_3	BYTE	RETENTIVO				4803	33002 MSB
- %MA2			RETENTIVO			FALSE		

Formas de atualização de Frame

Para CPUs a configuração do frame no Access Path permite 4 formas diferentes de atualização da seguinte forma:

- Atualização cíclica da CPU;
- Atualização por tempo definido no campo **Tempo de Atualização (ms)**;
- Atualização por evento, quando acionada a variável definida no campo **Variável de Atualização**;
- Atualização por tempo (**Tempo de Atualização**) e evento (**Variável de Atualização**).



Para as formas de atualização descritas a seguir, é necessário que as variáveis de sistema **MASTER_A** e/ou **MASTER_B** que definem modo mestre nos canais de comunicação **A** (RS232) ou **B** (RS485) estejam habilitadas. Para comunicação com protocolo **Modbus** habilitar as variáveis de sistema **MODBUS_A** e/ou **MODBUS_B**.

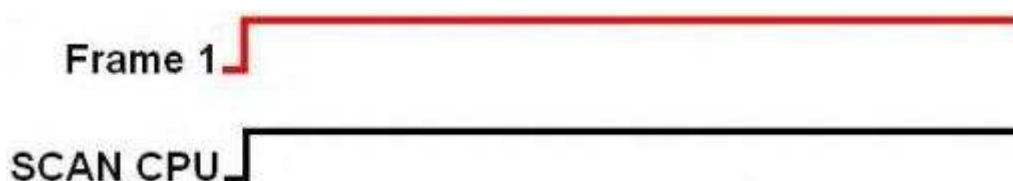
Atualização Cíclica da CPU

Modo de configuração:

Tempo Atualização(ms) 0
Variável Atualização

Após carregar a CPU com o programa de usuário e acionar a variável de sistema correspondente ao canal de comunicação para habilitar mestre de rede (**MASTER_A** e/ou **MASTER_B**), o frame configurado será **atualizado ciclicamente** com a **varredura da CPU**, ou seja, os dados relativos ao frame serão transmitidos constantemente.

Exemplo para 1 Frame:



Para mais de um frame configurado, os mesmos serão atualizados seqüencialmente conforme declarados até o último frame da lista. No final da lista retorna-se ao primeiro frame com o reinício do processo de atualização.

Atualização por tempo definido

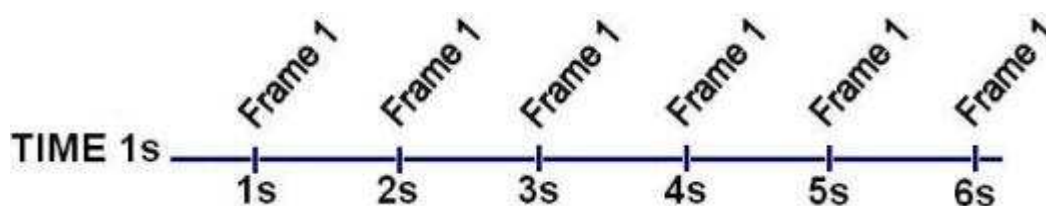
Modo de configuração:

Tempo Atualização(ms)	1000
Variável Atualização	

Após carregar a CPU com o programa de usuário e acionar a variável de sistema correspondente ao canal de comunicação para habilitar mestre de rede (MASTER_A e/ou MASTER_B), o frame configurado será atualizado de acordo com o tempo inserido no campo **Tempo Atualização (ms)**.

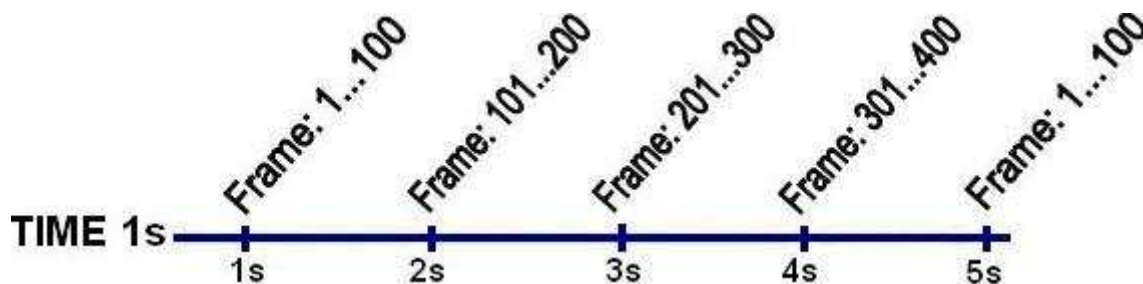
Para o exemplo dado, foi inserido um tempo de 1000 ms equivalente a 1 s.

Exemplo para 1 Frame:



A cada 1 s o Frame 1 é atualizado.

Para mais de um frame configurado com o mesmo tempo de atualização, os mesmos serão atualizados sequencialmente conforme declarados até o final do tempo configurado, ou seja, com 400 frames configurados e ao final do tempo (exemplo: 1s) a lista de frames foi atualizada até o frame 100, no início do próximo período (exemplo: próximo 1s) o frame a ser atualizado será o frame 101 e assim sucessivamente até o último frame da lista.



No final da lista de frames, retorna-se ao primeiro frame com o reinício do processo de atualização.

Atualização por Evento

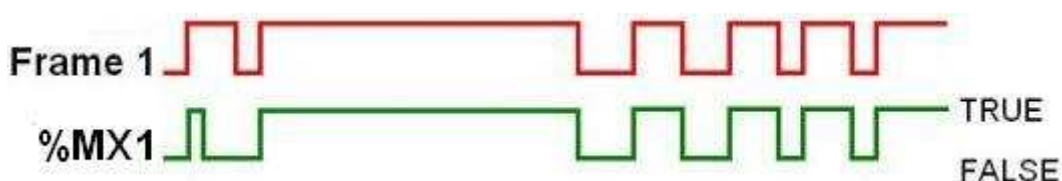
Modo de configuração:

Tempo Atualização(ms)	0
Variável Atualização	%MX1

Após carregar a CPU com o programa de usuário e acionar a variável de sistema correspondente ao canal de comunicação para habilitar mestre de rede (MASTER_A e/ou MASTER_B), o frame configurado será atualizado de acordo com o status da variável booleana inserida no campo **Variável Atualização**.

Para o exemplo dado, foi inserida a variável booleana %MX1.

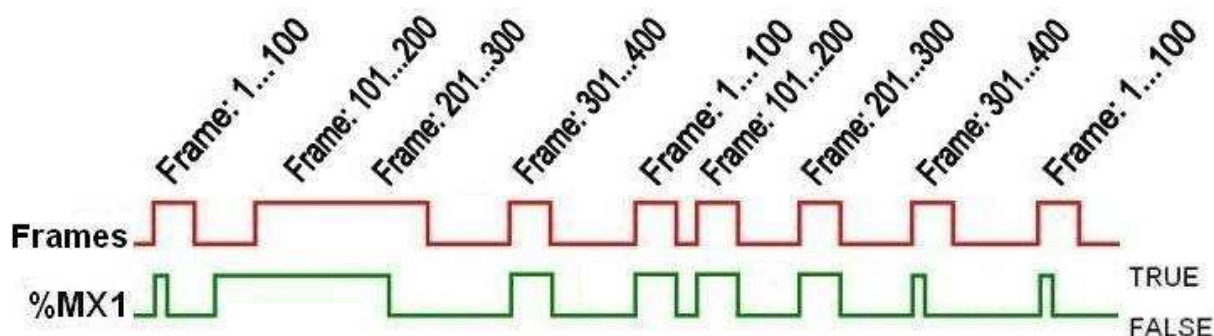
Exemplo para 1 Frame:



Quando o status da variável %MX1 passar de FALSE para TRUE o Frame1 será atualizado. Caso a variável seja desacionada enquanto há transferência de dados e o frame não tenha completado a transmissão, **o frame atualizado não interrompe a transmissão** sem antes **acabar** de realizar toda a ação de **transferência** de dados.

Para mais de um frame configurado com a mesma variável de atualização, os mesmos serão atualizados seqüencialmente conforme declarados até o final da lista enquanto a variável estiver habilitada, ou seja, se a variável for desabilitada, o frame que estiver transmitindo dados termina a transmissão e na próxima transição de FALSE para TRUE da variável segue a atualização do próximo frame da lista que não havia sido atualizado na transição anterior da variável. Este processo é sucessivamente realizado até a atualização do último frame da lista, retornado em seguida para o primeiro frame.

Exemplo para mais Frames:



Atualização por Tempo e Evento

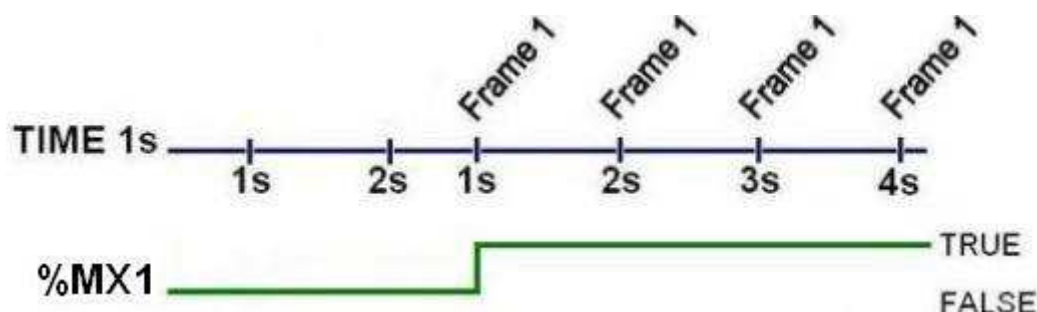
Modo de configuração:

Tempo Atualização(ms) 1000
Variável Atualização %MX1

Após carregar a CPU com o programa de usuário e acionar a variável de sistema correspondente ao canal de comunicação para habilitar mestre de rede (MASTER_A e/ou MASTER_B), o frame configurado será atualizado de acordo com o status da variável booleana inserida no campo **Variável Atualização** e com o tempo inserido no campo **Tempo Atualização (ms)**.

Para o exemplo dado, foi inserida a variável booleana %MX1 e tempo de 1000 ms equivalente a 1 s.

Exemplo para 1 Frame:



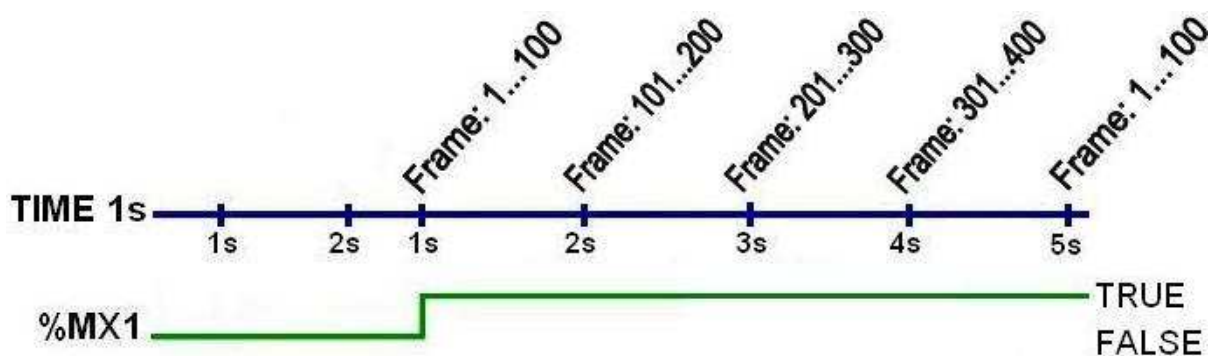
Enquanto a variável de atualização não for habilitada é transcorrido o tempo de atualização. Havendo a transição da variável %MX1 de FALSE para TRUE, inicia-se uma nova contagem e a atualização do Frame 1. Mantendo a variável em TRUE o frame é atualizado na cadência de tempos definido no campo **Tempo Atualização (ms)**.

Desabilitando a variável de atualização, o frame termina a transmissão dos dados para não haver perda de informação.

Para mais de um frame configurado com a mesma variável de atualização e o mesmo tempo, estes frames serão atualizados seqüencialmente conforme declarados até o final da lista.

Desabilitando a variável sem que a lista de frames seja finalizada, será aguardada a transição de FALSE para TRUE da variável, iniciando a atualização a partir do frame não transmitido na última transição até o final da lista, retornado em seguida para o primeiro frame.

Exemplo para mais Frames:



Funções Modbus

- (0x01) - Read Coil Status;
- (0x02) - Read Input Status;
- (0x03) - Read Holding Registers;
- (0x04) - Read Input Registers;
- (0x05) - Force Single Coil;
- (0x06) - Preset Single Register;
- (0x0F) - Force Multiple Coils;
- (0x10) - Preset Multiple Registers;
- (ERROR) - [Exception Response](#).



Todos os dados contidos nos frames estão no formato **hexadecimal**.

Read Coil Status - (0x01)

Esta função permite ler o status de variáveis booleanas. É permitido ler no máximo 256 variáveis booleanas.

Os frames desta função são:

MASTER							
End.da Slave	Função	Endereço HI	Endereço LO	Número de pontos HI	Número de pontos LO	CRC LO	CRC HI
	01						

SLAVE							
End.da Slave	Função	Nº de Bytes	Dado 1	Dado 2	Dado 3	CRC LO	CRC HI
	01						

Onde o dado pode ser:

Dado 1 Bit7 Bit6 Bit5 Bit4 Bit3 Bit2 Bit1 Bit0
 %MX8 %MX7 %MX6 %MX5 %MX4 %MX3 %MX2 %MX1

Dado 2 Bit7 Bit6 Bit5 Bit4 Bit3 Bit2 Bit1 Bit0
 %MX16 %MX15 %MX14 %MX13 %MX12 %MX11 %MX10 %MX9

Exemplo: Ler as variáveis %MX1 até %MX16 na estação 07:

TX: 07 01 1B 58 00 10 BA 97

RX: 07 01 02 01 00 30 6C

Neste exemplo temos que a variável booleana %MX1 está ligada (01, bit1 = 1) e todas as demais desligadas.

Read Input Status - (0x02)

Esta função permite ler o status de entradas digitais. É permitido ler no máximo 256 entradas digitais.

Os frames desta função são:

Master							
End.da Slave	Função	Endereço HI	Endereço LO	Número de pontos HI	Número de pontos LO	CRC LO	CRC HI
	02						

SLAVE							
End.da Slave	Função	Nº de Bytes	Dado 1	Dado 2	Dado 3	CRC LO	CRC HI
	02						

Exemplo: Ler as entradas E0 até E7 na estação 07:

TX: 07 02 00 00 00 08 79 AA

RX: 07 02 01 02 20 C1

Neste exemplo temos que a Entrada E1 está ligada (02, bit2=1) e todas as demais desligadas.

Read Holding Registers - (0x03)

Esta função permite ler variáveis numéricas de até 32 bits. É permitido ler no máximo 63 variáveis de 4 bytes.

Os frames desta função são:

MASTER							
End.da Slave	Função	Endereço HI	Endereço LO	Número de pontos HI	Número de pontos LO	CRC LO	CRC HI
	03						

SLAVE										
End.da Slave	Função	Nº de Bytes	Dado HI 1	Dado LO 1	Dado HI 2	Dado LO 2	Dado HI 3	Dado LO 3	CRC LO	CRC HI
	03									

Exemplo: Ler os efetivos de 2 temporizadores na estação 07:

TX: 07 03 75 30 00 04 5E 6C

RX: 07 03 08 00 0A 00 00 29 5E 00 00 48 D1

Temporizador 01: 000A 0000

Temporizador 02: 295E 0000

Read Input Registers - (0x04)

Esta função permite ler entradas analógicas. É permitido ler no máximo 126 entradas analógicas.

Os frames desta função são:

MASTER

End.da Slave	Função	Endereço HI	Endereço LO	Número de pontos HI	Número de pontos LO	CRC LO	CRC HI
	04						

SLAVE

End.da Slave	Função	Nº de Bytes	Dado HI 1	Dado LO 1	Dado HI 2	Dado LO 2	Dado HI 3	Dado LO 3	CRC LO	CRC HI
	04									

Exemplo: Ler 2 entradas analógicas da estação 07:

TX: 07 04 00 80 00 02 70 45

RX: 07 04 04 03 C1 00 66 4C 16

Entrada analógica E1: 03C1

Entrada analógica E2: 0066

Force Single Coil - (0x05)

Esta função permite a alteração do status de uma variável booleana.

O Status do Force deve ser:

0xFF00 - forçar para TRUE a variável booleana.

0x0000 - forçar para FALSE a variável booleana.

Os frames desta função são:

MASTER

End.da Slave	Função	Endereço HI	Endereço LO	Status do Force HI	Status do Force LO	CRC LO	CRC HI
	05						

SLAVE

End.da Slave	Função	Endereço HI	Endereço LO	Status do Force HI	Status do Force LO	CRC LO	CRC HI
	05						

Exemplo1: Ligar a saída %Q0.0 na estação 07:

TX: 07 05 0B B8 FF 00 0E 5D

RX: 07 05 0B B8 FF 00 0E 5D

Exemplo2: Desligar a saída %Q0.0 na estação 07:

TX: 07 05 0B B8 00 00 4F AD

RX: 07 05 0B B8 00 00 4F AD

Preset Single Register - (0x06)

Esta função permite pré-setar um valor em uma variável numérica de até 16 bits (2 bytes).

Os frames desta função são:

MASTER							
End.da Slave	Função	Endereço HI	Endereço LO	Dado HI	Dado LO	CRC LO	CRC HI
	06						

SLAVE							
End.da Slave	Função	Endereço HI	Endereço LO	Dado HI	Dado LO	CRC LO	CRC HI
	06						

Exemplo: Fazer com que a variável %MW1 seja igual a 0934:

TX: 07 06 27 10 09 34 85 85

RX: 07 06 27 10 09 34 85 5A

Force Multiple Coils - (0x0F)

Esta função permite a alteração do status de mais de uma variável booleana. É permitido escrever no máximo 256 variáveis.

Os frames desta função são:

MASTER

End.da Slave	Função	End. HI	End. LO	Qtde. de variáveis HI	Qtde. de variáveis LO	Nº de bytes	Dado1	Dado2	CRC LO	CRC HI
	0F									

SLAVE

End.da Slave	Função	End. HI	End. LO	Qtde. de variáveis HI	Qtde. de variáveis LO	CRC LO	CRC HI
	0F						

Onde o dado pode ser:

Dado 1 Bit7 Bit6 Bit5 Bit4 Bit3 Bit2 Bit1 Bit0

%MX8 %MX7 %MX6 %MX5 %MX4 %MX3 %MX2 %MX1

Dado 2 Bit7 Bit6 Bit5 Bit4 Bit3 Bit2 Bit1 Bit0

%MX16 %MX15 %MX14 %MX13 %MX12 %MX11 %MX10 %MX9

Exemplo: Ligar todas variáveis entre %MX1 e %MX16 na estação 07.

TX: 07 0F 1B 58 00 10 02 FF FF 6E 29

RX: 07 0F 1B 58 00 10 D3 56

Preset Multiple Registers - (0x10)

Esta função permite presetar múltiplas variáveis numéricas. É permitido escrever no máximo 126 variáveis de 2 bytes.

Os frames desta função são:

MASTER

End.da Slave	Função	End. HI	End. LO	Qtde. de variáveis HI	Qtde. de variáveis LO	Nº de bytes	Dado HI 1	Dado LO 1	Dado HI 2	Dado LO 2	Dado HI 3	Dado LO 3	CRC LO	CRC HI
	10													

SLAVE

End.da Slave	Função	End. HI	End. LO	Qtde. de variáveis HI	Qtde. de variáveis LO	CRC LO	CRC HI
	10						

Exemplo: Presetar as variáveis %MW1, %MW2 e %MW3 na estação 07 na seguinte maneira:

%MW1: 0999

%MW2: 2340

%MW3: 1523

TX: 07 10 27 10 00 03 06 09 99 23 40 15 23 06 FB

RX: 07 10 27 10 00 03 8B 1F

Range de endereçamento MODBUS

Cada função modbus comunica com um determinado range de variáveis, conforme mostra a tabela abaixo:

- 01 - Read Coil Status
- 02 - Read Input Status
- 03 - Read Holding Registers
- 04 - Read Input Registers
- 05 - Force Single Coil
- 06 - Preset Single Register
- 15 - Force Multiple Coils
- 16 - Preset Multiple Registers

Tipo	IEC	Dado	Função
Usuário	%MX1 - %MX4096	BOOL	01, 05, 15
Usuário	%MD1 - %MD256	DINT, UDINT	03, 16
Usuário	%MW1 - MW2048	INT, UINT	03, 06, 16
Usuário	%MT1 - %MT512	DATE, TOD, TIME	03, 16
Usuário	%MR1 - %MR256	REAL	03, 16
Usuário	%MA1 - %MA8172	ARRAY, STRING e STRUCT	03, 16
Sistema	%SX1 - %SX256	BOOL	01, 05, 15
Sistema	%SW1 - %SX128	INT, UINT	03, 06, 16
Sistema	%SD1 - %SD32	DINT, UDINT	03,16
Sistema	%SR1 - %SR32	REAL	03,16
Sistema	%SA1 - %SA4	BYTE	03
Digital	%I0.0 - %I15.1F	BOOL	02
Digital	%Q0.0 - %Q15.1F	BOOL	01, 05, 15
Analógico	%IW1.1 - %IW15.16	INT, UINT	04
Analógico	%QW1.1 - %QW15.16	INT, UINT	03, 06, 16

Exception Response (ERROR)

Os erros de comunicação serão interpretados pela Slave de 2 maneiras:

1. **No reply** – Caso haja erro de **CRC**;
2. **Exception Response** – Caso haja erro de função inválida, endereço inválido ou dado inválido ([Ver Códigos de erro ou falhas](#)).

CRC - Cyclic Redundancy Check

Verificação de redundância cíclica é um código detector de erros, um tipo de função hash que gera um valor expresso em poucos bits que é função de um bloco maior de dados, como um pacote de dados, ou um arquivo, por forma a detectar erros de transmissão ou armazenamento.

O CRC é calculado e anexado à informação a transmitir (ou armazenar) e verificada após a recepção ou acesso, para confirmar se não ocorreram alterações. O CRC é popular por ser simples de implementar em hardware binário, simples de ser analisado matematicamente, e pela eficiência em detectar erros típicos causados por ruído em canais de transmissão.

O cálculo do CRC parte de um polinômio base de grau N, o polinômio gerador, que deve ser o mesmo sempre no caso da transmissão, quer no emissor quer no receptor. Teoricamente, o grau pode ser qualquer um, embora seja tipicamente especificado em função da máxima eficiência possível na detecção de erros.

Verificando configurações de comunicação via IHM

Este recurso permite a visualização das configurações de comunicação do CLP via IHM.

Para acessar essas informações, os seguintes passos devem ser seguidos:

- Coloque o CLP em modo PROG;
- Pressione a tecla  , da IHM;
- Para navegar pelas telas utilize as teclas  e  ;

Pode-se visualizar as seguintes telas:

- **Configurações de Comunicação**

```
[APR03 RS232] M:1  
T:57600 D:8 S:1 P:N
```

```
[APR03 RS485] M:1  
T:57600 D:8 S:1 P:N
```

```
[MODBUS RS232] M:1  
T:57600 D:8 S:1 P:N
```

```
[MODBUS RS485] M:1  
T:57600 D:8 S:1 P:N
```

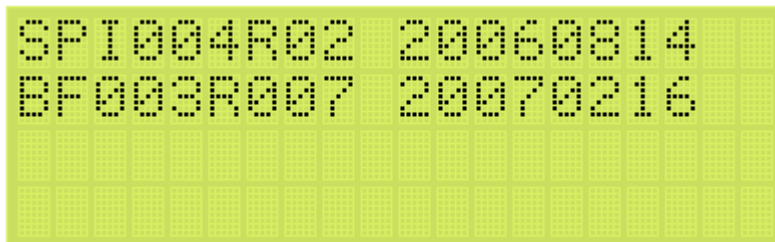
Descrição dos campos:

- [Protocolo e Porta]
- **M:** Número da estação
- **T:** Taxa de comunicação (Baud Rate);
- **D:** Data Bits
- **S:** Stop Bits
- **P:** Paridade
- **Configuração de envio e recebimento de caracteres pela serial**

```
[PRINT RS485] M:0  
T:57600 D:8 S:1 P:N
```

Descrição dos campos:

- [Função e Porta]
- **M:** Número da estação
- **T:** Taxa de comunicação (Baud Rate);
- **D:** Data Bits
- **S:** Stop Bits
- **P:** Paridade
- **Visualização de versão de SPI e FIRMWARE**



Descrição dos campos:

- **SPI:** Nome e data do arquivo;
- **FIRMWARE:** Nome e data do arquivo.

Módulo de comunicação

Através do software Atos A1 Soft, é possível configurar os modelos de um módulo de comunicação disponíveis para os drivers Atos MPC4004BF e MPC6006, mostrados na tabela abaixo:

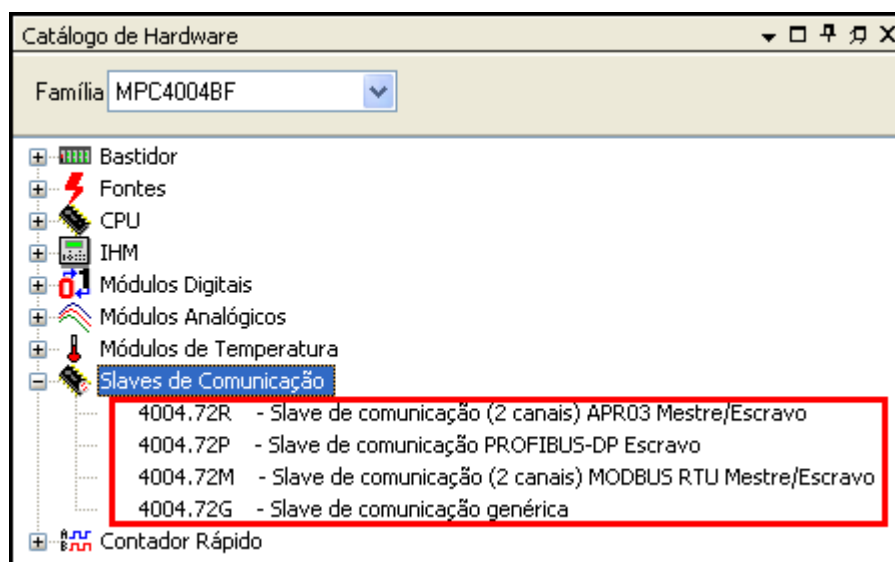
Referência	Descrição°
4004.72M	Slave de comunicação (2 canais) Modbus RTU Mestre/Escravo
4004.72R	Slave de comunicação (2 canais) Apr03 Mestre/Escravo
4004.72P	Slave de comunicação Profibus DP
6006.72	Slave de comunicação (2 canais) Modbus/APR - Mestre/Escravo
6006.72C	Módulo de comunicação (1 canal) Mestre CANopen

Inserindo uma Slave de comunicação no projeto

Para inserir um novo módulo na configuração de hardware, siga o procedimento abaixo:

- Clique com o botão direito do mouse sobre o item "Hardware" no Gerenciador de Projetos;
- No menu pop-up que abrir, clique sobre a opção "[Ver catálogo de hardware](#)".

No catálogo de Hardware mostrado abaixo, as referências da tabela acima podem ser visualizados no item "Slaves de comunicação".

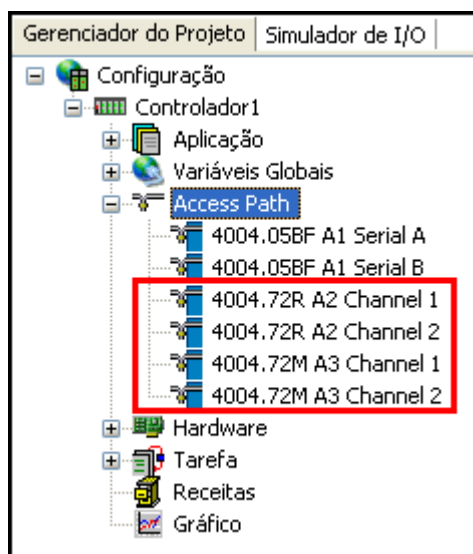


Para inserir uma slave é necessário ter um bastidor com pelo menos 1 slot livre configurado.

Depois de inserida a slave de comunicação, seus canais podem ser visualizados no item "Acess Path" do Gerenciador de Projeto.

Para configuração de frames de comunicação é necessário que o canal seja configurado como [modo mestre](#).

Cada referência de slave de comunicação possui regras específicas para sua configuração.



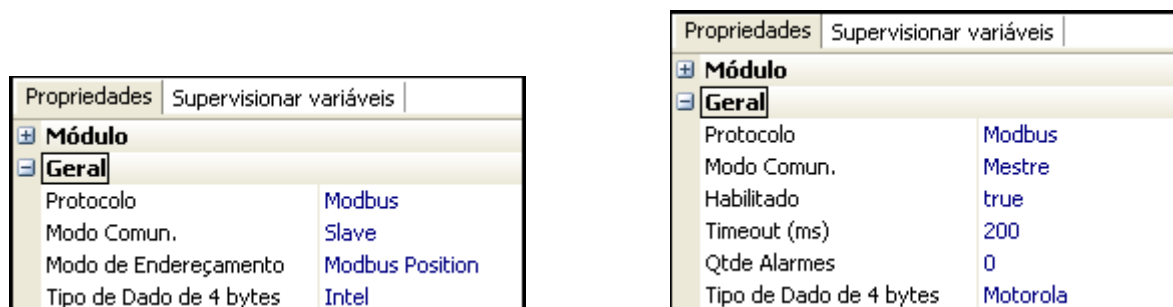
Quando configurados como Mestre, os canais da Slave podem comunicar até 1Kbyte de dados. Caso essa quantidade de bytes nos frames (Canal 1 e 2) ultrapasse o limite, uma mensagem de erro é mostrada na compilação do projeto.

Slave de comunicação 4004.72M

A Slave de comunicação **4004.72M** possui dois canais de comunicação isolados oticamente, que podem ser configurados como mestre ou escravo, utilizando o protocolo **Modbus RTU**.

Cada canal possui o padrão elétrico configurável individualmente como RS-232 ou RS-485 através de jumper no interior da slave de comunicação (Para mais informações, ver Catálogo de Hardware).

Através da guia Propriedades do Canal da Slave devem ser configurados alguns parâmetros tanto para o módulo definido como mestre ou como escravo, conforme mostra as figuras abaixo.



Descrição dos campos:

- **Modo Comun.:** Define a estação como Mestre de rede ou escravo (Slave);
- **Modo de Endereçamento:** Define o modo de endereçamento utilizado na comunicação podendo ser através do Endereço Físico ou da Posição Modbus;

Quando o módulo for configurado como Mestre, os seguintes campos serão habilitados para configuração:

- **Habilitado:** Em TRUE habilita o Access Path sem apagar os frames configurados;
- **Timeout (ms):** Tempo em milisegundos que a CPU aguarda a resposta de um frame antes de sinalizar algum tipo de falha;
- **Qtde Alarmes:** Variável do tipo UINT, que relaciona o número de estações na rede com a quantidade de alarmes, ou seja, com 25 slaves na rede deverão ser colocados 25 alarmes. O uso de alarmes é opcional, porém promove agilidade e praticidade na identificação de falhas em uma ou mais estações;
- **Variável inicial de falha:** Variável do tipo UINT, neste campo pode ser inserido o endereço (Exemplo %MW1) ou o nome dado à variável (Exemplo FALHA_1).

Esta variável está relacionada com o campo Qtde Alarmes, ou seja, com 25 alarmes configurados serão fixadas 25 variáveis a partir do endereço inicial de falha.

Quando a primeira estação apresentar algum erro a primeira variável inicial de falha irá exibir um código que relaciona o tipo de erro. Se a décima estação apresentar algum erro a décima variável de falha irá exibir qual o tipo de falha ([Ver Exemplo](#)).

- **Tipo de Dado de 4bytes:** Para as variáveis de 4 bytes é necessário configurar o formato que o dado será enviado e recebido, podendo ser no formato Intel, Motorola ou Motorola Invertido.



Os dois canais devem ter a mesma configuração, por isso, ao alterar as configurações de um dos canais o outro automaticamente será modificado.

A imagem a seguir mostra a configuração de frames no módulo configurado como Mestre.

Descrição dos campos:

- **Variável Inicial:** Endereço (Formato %M...) ou nome dado à variável na área de *Variáveis Globais*;
- **Qtde de Variáveis:** Número de variáveis transportadas;

Cada frame pode transportar até 64 bytes, ou seja, até 32 variáveis numéricas de 2 bytes, 16 variáveis de 4 bytes ou 64 variáveis do tipo Coil/Input (Booleana).

- **Endereço remota:** Endereço da variável (posição Modbus) em hexadecimal ou decimal;
- **N° estação da remota:** Número de estação da unidade remota;
- **Função:** Função Modbus a ser executada;



Quando o módulo da Slave for comunicar com as CPU ATOS MPC4004 deverá ser utilizado o endereço da remota em Hexa. Cada frame pode transportar até 126 variáveis numéricas de 2 bytes ou 256 variáveis do tipo Coil/Input (Booleana).



Se atualizada o firmware da Slave 4004.72M da versão **V4** para **V5**, as configurações de endereçamento devem ser alteradas para **Posição Modbus**.

Slave de comunicação 4004.72R

A Slave de comunicação **4004.72R** possui dois canais de comunicação isolados oticamente, que podem ser configurados como mestre ou escravo, utilizando o protocolo **Apr03**.

Cada canal possui o padrão elétrico configurável individualmente como RS-232 ou RS-485 através de jumper no interior da slave de comunicação (Para mais informações, ver Catálogo de Hardware).

A configuração do item **Endereço da remota (hex)**, da guia Propriedades do frame, aceita somente endereço em hexadecimal ([Endereço físico](#)), conforme mostra a figura abaixo, tanto para modo mestre quanto modo escravo, não importando o dispositivo com o qual irá comunicar.

Descrição dos campos:

- **Variável Inicial:** Endereço (**Formato %M...**) ou nome dado à variável no gerenciador de **Variáveis Globais**;
- **Qtde Bytes:** Número de bytes transportados; Cada frame pode transportar 255 bytes para variáveis numéricas e 255 variáveis booleanas.
- **Endereço remota (hex):** Endereço da variável em hexadecimal do dispositivo remoto;
- **Nº estação da remota:** Número de estação do dispositivo remoto;
- **Direção:** Define o fluxo de dados entre a Slave de comunicação (Mestre) e o dispositivo remoto (Slave);
- **Receber da Remota:** Leitura de informação no dispositivo (dispositivo -> slave de comunicação);
- **Enviar para Remota:** Escrita de informação no dispositivo (slave de comunicação -> dispositivo).

A quantidade máxima de bytes transportados também depende do dispositivo com o qual a slave se comunica. A tabela abaixo mostra a máxima quantidade de bytes transportados com alguns dispositivos.

Quantidade de bytes transportados	
Dispositivos	Bytes
Slave <-> Atos MPC4004BF	128 bytes
Slave <-> Atos MPC4004/L/G	8 bytes
Slave <-> Atos MPC4004R/T	24 bytes
Slave <-> TaprWin	128 bytes



Quando utilizado um sistema supervisor a máxima quantidade irá depender do sistema utilizado, sendo no máximo 128 bytes.
A quantidade de bytes configurados deve ser múltipla do tipo de dado que se deseja comunicar.

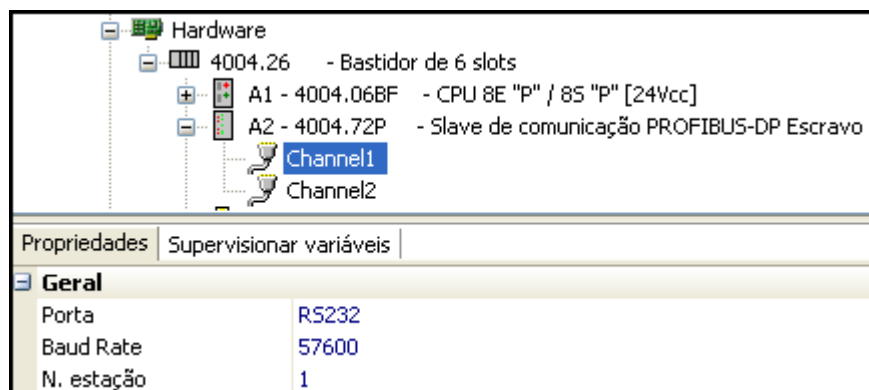
Por exemplo: Variáveis do tipo INT (2 bytes) só são atualizadas se configurados frames com a quantidade de bytes múltipla de 2.

Slave de comunicação 4004.72P

A slave de comunicação Profibus DP 4004.72/P é um adaptador de comunicação que pode ser acessado para leitura e escrita através de um mestre Profibus DP.

Ela possui dois canais, sendo canal 1 RS232 e canal 2 RS485, para configuração de frames de comunicação entre ela e a CPU principal. Através da guia Propriedades do Canal da Slave devem ser configurados alguns parâmetros conforme mostra a figura abaixo.

Canal 1



Descrição dos campos:

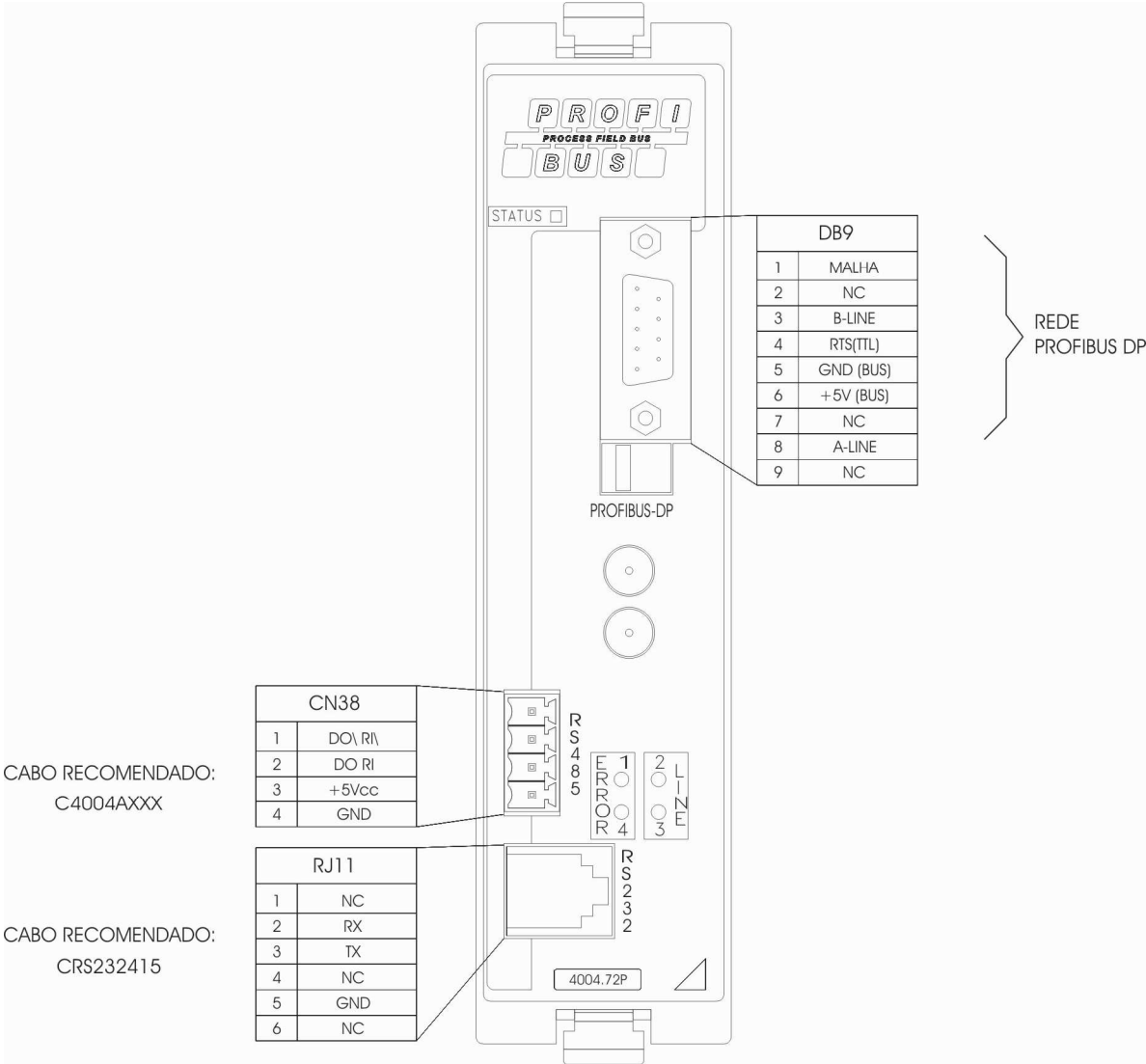
- **Porta:** Padrão elétrico utilizado (Canal 1 RS232/ Canal 2 RS485);
- **Baud Rate:** Define a taxa de comunicação;
- **Número da estação:** Define o número da estação;



Para demais configurações é necessário o uso do Software Atos Fieldbus.

Conexões do módulo Slave:

As conexões deverão ser realizadas conforme a figura abaixo:



RS232 e RS485

Estas conexões serão utilizadas para configuração do módulo mestre 4004.72MP, através do Atos Fieldbus. Devem ser utilizados os cabos recomendados.

Taxa de transmissão: 9,6KB a 12 MB

Indicadores de diagnóstico:

Através dos LEDs localizados no frontal é possível verificar o diagnóstico do módulo.

LED	Sinalização	Status
Diagnóstico	Piscar em vermelho na cadência de 1 HZ	Erro de configuração.
	Piscar em vermelho na cadência de 2 HZ	Erros nos dados do parâmetro do usuário.
	Piscar em vermelho na cadência de 2 HZ	Erro na inicialização da comunicação Profibus ASIC.
	Apagado	Sem diagnóstico presente.
On - Line	Aceso verde	Módulo esta Off-line pronto para troca de dados.
	Apagado	Módulo não esta On-line
Off - Line	Aceso vermelho	Módulo esta On-line impossível a troca de dados
	Apagado	Modulo não esta Off-line

O comprimento máximo do cabo depende da velocidade de transmissão (Veja tabela abaixo). As especificações de comprimento de cabo na tabela abaixo, são baseadas em um cabo, com as seguintes especificações:

- Impedância: 135 a 165 Ohms
- Capacidade: < 30 pf/m
- Resistência: 110 Ohms/Km
- Medida do cabo: 0.64mm
- Área do condutor: >0.34mm

Baud rate (kbits/s)	9.6	19.2	93.75	187.5	500	1500	1200
Distancia / segmento (m)	1200	1200	1200	1000	400	200	100

Observações:

1 - Durante a instalação observe atentamente a polaridade dos sinais de dados (A e B).

2 - O uso da blindagem é absolutamente essencial para se obter alta imunidade contra interferências eletromagnéticas. A blindagem por sua vez deve ser conectada ao sistema de aterramento em ambos os lados através de bornes de aterramento adequados. Adicionalmente recomenda-se que os cabos de comunicação sejam mantidos separados dos cabos de alta voltagem.

3 - O uso de cabos de derivação deve ser evitados para taxas de transmissão acima de 1,5Mbits/s.

Observação: As informações referentes ao Software Atos Fieldbus estão disponíveis no manual M400472P110 "Slave de comunicação Profibus-DP"

Slave de comunicação 6006.72

A Slave de comunicação 6006.72 possui dois canais de comunicação isolados opticamente, que podem ser configurados como mestre ou escravo, utilizando os protocolos Modbus RTU e Apr03 e podem ser configurados de maneiras diferentes, segue tabela abaixo com as possíveis combinações.

Canal 1 Escravo (Modbus)	Canal 2 Escravo (APR03)
Canal 1 Escravo (Modbus)	Canal 2 Mestre (APR03)
Canal 1 Mestre (Modbus)	Canal 2 Escravo (APR03)
Canal 1 Mestre (Modbus)	Canal 2 Mestre (APR03)
Canal 1 Escravo (APR03)	Canal 2 Escravo (Modbus)
Canal 1 Escravo (APR03)	Canal 2 Mestre (Modbus)
Canal 1 Mestre (APR03)	Canal 2 Escravo (Modbus)
Canal 1 Mestre (APR03)	Canal 2 Mestre (Modbus)
Canal 1 Escravo (Modbus)	Canal 2 Escravo (Modbus)
Canal 1 Mestre (Modbus)	Canal 2 Escravo (Modbus)
Canal 1 Escravo (Modbus)	Canal 2 Mestre (Modbus)
Canal 1 Mestre (Modbus)	Canal 2 Mestre (Modbus)
Canal 1 Escravo (APR03)	Canal 2 Escravo (APR03)
Canal 1 Mestre (APR03)	Canal 2 Escravo (APR03)
Canal 1 Escravo (APR03)	Canal 2 Mestre (APR03)
Canal 1 Mestre (APR03)	Canal 2 Mestre (APR03)



Cada canal possui o padrão elétrico configurável individualmente como RS-232 ou RS-485 através do Software de programação Atos A1 Soft

Através da guia Propriedades do Canal da Slave em Access Path devem ser configurados alguns parâmetros tanto para o módulo definido como mestre como escravo, conforme as figuras abaixo.

Modo Escravo



Escravo APR



Escravo Modbus

Descrição dos campos:

- **Protocolo:** Define o protocolo utilizado (APR03 ou Modbus)
- **Modo Comun.:** Define o módulo como Mestre de rede ou escravo (Slave);

Caso o protocolo seja configurado Modbus, teremos os seguintes campos habilitados para a configuração:

- **Modo de endereçamento:** Define o modo do endereçamento (Endereço Físico ou Posição Modbus).
- **Tipo de Dado de 4 bytes:** Define o tipo de dado (Intel, Motorola ou Motorola invertido).

Modo Mestre



Mestre APR



Mestre Modbus

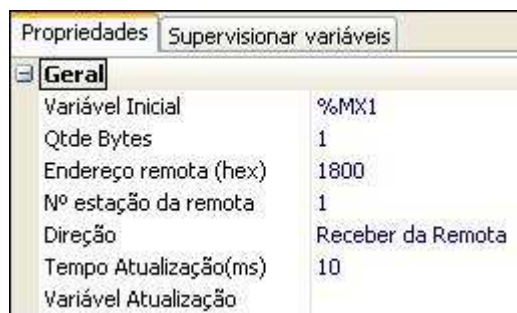
Descrição dos campos:

- **Protocolo:** Define o protocolo utilizado (APR03 ou Modbus)
- **Modo Comun.:** Define o módulo como Mestre de rede ou escravo (Slave);
- **Habilitado:** Em TRUE habilita o Access Path, em FALSE desabilita o Access Path sem apagar os frames configurados;
- **Timeout (ms):** Tempo em milisegundos que a CPU aguarda a resposta de um frame antes de sinalizar algum tipo de falha;
- **Tempo entre frames:** Tempo em milisegundos que a CPU aguarda após enviar um frame para enviar o próximo.
- **Qtde Alarmes:** Variável do tipo UINT, que relaciona o número de estações na rede com a quantidade de alarmes, ou seja, com 25 slaves na rede deverão ser colocados 25 alarmes. O uso de alarmes é opcional, porém promove agilidade e praticidade na identificação de falhas em uma ou mais estações;
- **Quantidade de variáveis de atualização:** Define a quantidade de variáveis (Booleanas) que serão utilizadas para a ativar a comunicação dos frames.

Configuração de Frames de Comunicação:

Quando o Canal estiver configurado como Mestre, podemos inserir um novo frame de comunicação clicando com o botão direito do mouse no canal configurado como mestre no access path e escolhendo a opção "Novo Frame", deve-se então, ser configurados alguns parâmetros, conforme mostram os exemplos abaixo

Exemplo de configuração para um frame APR03:



Propriedades	
Supervisionar variáveis	
Geral	
Variável Inicial	%MX1
Qtde Bytes	1
Endereço remota (hex)	1800
Nº estação da remota	1
Direção	Receber da Remota
Tempo Atualização(ms)	10
Variável Atualização	

Descrição dos campos:

- **Variável Inicial:** Endereço (Formato %M...) ou nome dado à variável na área de Variáveis Globais;
- **Qtde de bytes:** Número de bytes transportados;

Cada frame pode transportar até 255 bytes, ou seja, até 126 variáveis numéricas de 2 bytes, 64 variáveis de 4 bytes ou 255 variáveis do tipo (Booleana).

- **Endereço remota:** Endereço da variável (endereço físico) em hexadecimal;
- **Nº estação da remota:** Número de estação da unidade remota;
- **Direção:** Direção a ser executada (Enviar ou Receber);
- **Tempo de atualização (ms):** Tempo em milisegundos em que o frame será atualizado;
- **Variável Atualização:** Variável do tipo bool, usada para habilitar a comunicação do frame

Exemplo de configuração para um frame MODBUS:

Propriedades	
Supervisionar variáveis	
Geral	
Variável Inicial	ENVIA_REAL
Qtde de variáveis	1
Endereço remota	32019 decimal
Nº estação da remota	1
Funcao	Preset Multiple Registers
Tipo de Dado de 4 bytes	Motorola
Tempo Atualização(ms)	0
Variável Atualização	

Descrição dos campos:

- **Variável Inicial:** Endereço (Formato %M...) ou nome dado à variável na área de Variáveis Globais;
- **Qtde de variáveis:** Número de variáveis transportadas;

Cada frame pode transportar até 255 bytes, ou seja, até 126 variáveis numéricas de 2 bytes, 64 variáveis de 4 bytes ou 255 variáveis do tipo (Booleana).

- **Endereço remota:** Endereço da variável (posição Modbus);
- **Nº estação da remota:** Número de estação da unidade remota;
- **Função:** Função a ser executada;
- **Tipo de Dado de 4 bytes:** Define o tipo de dado (Intel, Motorola ou Motorola invertido). Obs: Este campo somente é mostrado se a variável for de 4 bytes.
- **Tempo de atualização (ms):** Tempo em milisegundos em que o frame será atualizado;
- **Variável Atualização:** Variável do tipo bool, usada para habilitar a comunicação.

Módulo de comunicação 6006.72C

O módulo de comunicação **6006.72C** possui um canal Mestre CANopen que permite trocar dados com até 32 dispositivos na mesma rede.

Através da guia Propriedades do canal do módulo devem ser configurados alguns parâmetros conforme mostra a figura abaixo



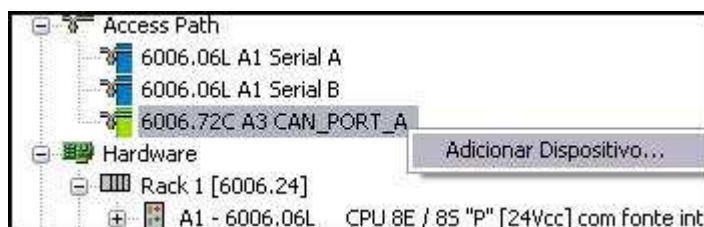
Descrição dos campos:

- **Protocolo:** Define o protocolo utilizado (CANopen);
- **ID do nó:** Define o ID do mestre CANopen;
- **Baud Rate:** Define a taxa de comunicação;
- **Habilita Heartbeat:** Habilita / Desabilita o Heartbeat;
- **ID do Nó do Heartbeat:** Define o ID do Heartbeat no mestre CANopen;
- **Tempo do Heartbeat:** Define o tempo de geração do Heartbeat;
- **Habilita Geração de Sincronismo:** Habilita / Desabilita a geração de Sincronismo

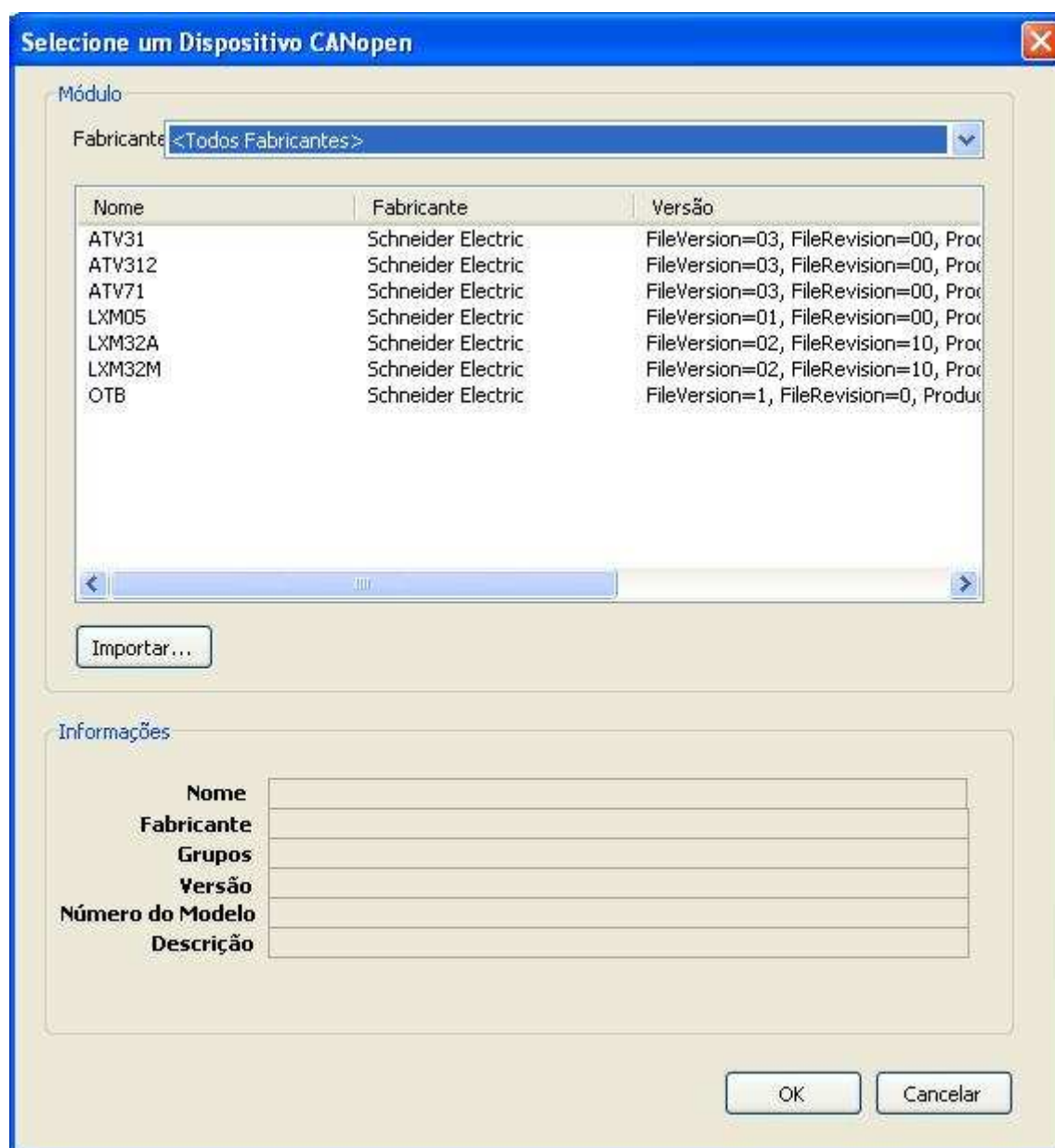
Configuração dos dispositivos na rede CANopen

Adicionando um dispositivo na rede:

Para adicionar um dispositivo na rede CANopen selecione com o cursor do mouse o canal de comunicação na guia Access Path, clique com o botão direito e escolha a opção "Adicionar Dispositivo..." conforme a figura abaixo:

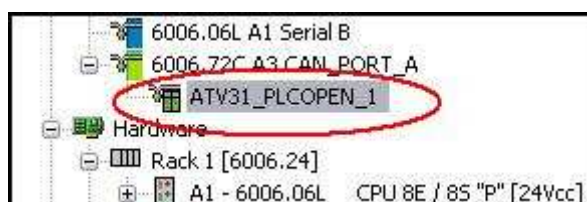


Ao seleccionar a opção "Adicionar Dispositivo...", a janela abaixo será exibida para a escolha do dispositivo.



Selecione o dispositivo e clique no botão OK para adicioná-lo à rede.

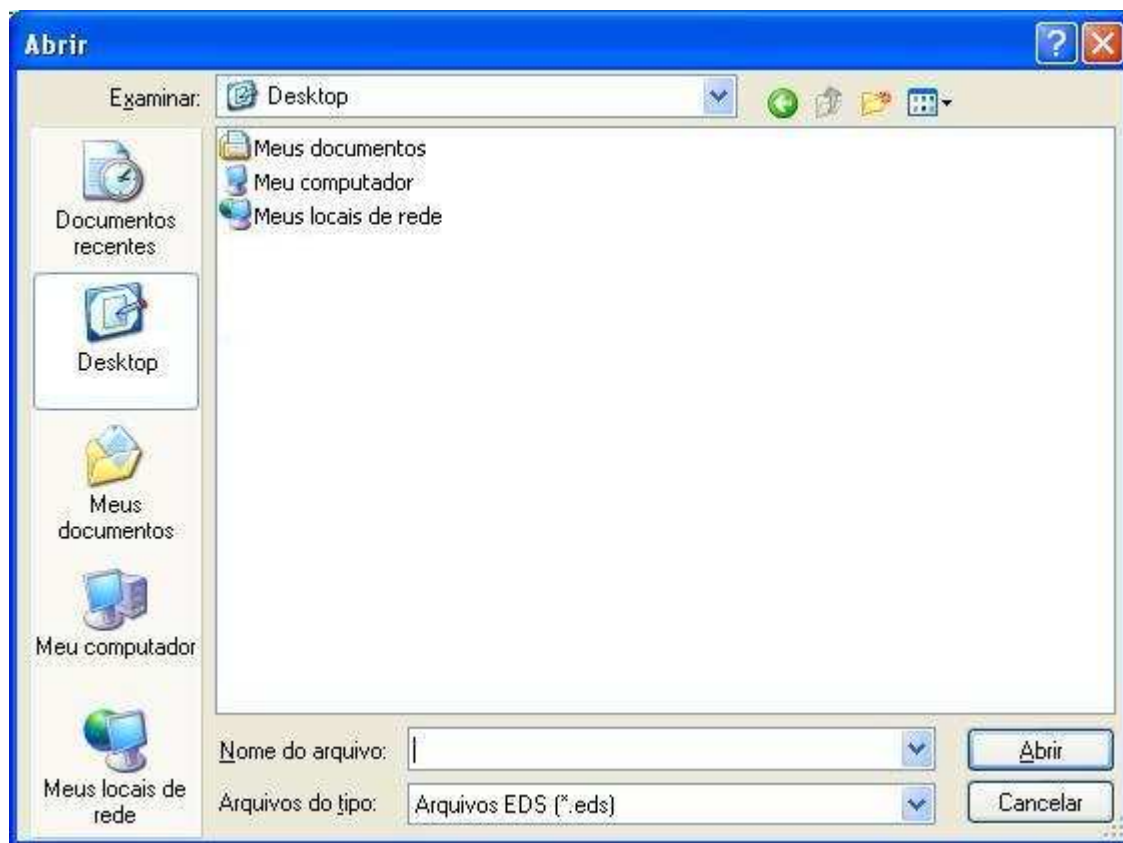
Após clicar no botão OK, o dispositivo selecionado estará disponível no access path conforme a figura abaixo:



O Software A1 contém os seguintes arquivos .EDS disponíveis:

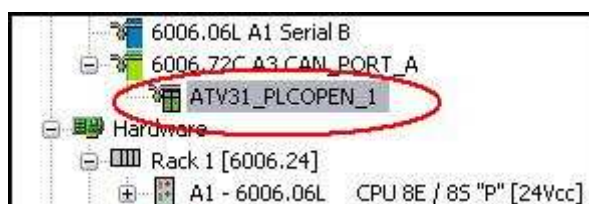
ATV31	LXM05	OTB
ATV312	LXM32A	
ATV71	LXM32M	

Caso deseje inserir novos arquivos .EDS, clique no botão "Importar...", após isso a janela abaixo será exibida para indicar o caminho do arquivo a ser importado.



Configurando um dispositivo

Na janela Gerenciador do Projeto, dê um duplo-clique no dispositivo a ser configurado na guia Access path, conforme a figura abaixo:



Após isso a janela abaixo será exibida para a configuração do dispositivo.

The screenshot shows the 'Mapeamento CANopen' configuration window. It has three tabs: 'Dispositivo Remoto CANopen', 'Mapeamento do PDO', and 'Configuração do SDO'. The 'Dispositivo Remoto CANopen' tab is active. The window is divided into several sections:

- Geral**:
 - ID do Nó: 1
 - ☐ Habilita Produtor de Sincronismo
 - ☒ Esse dispositivo é opcional na rede
- Node Guard**:
 - ☐ Habilita NodeGuard
 - Tempo de NodeGuard (ms): 0
 - Fator Tempo de vida: 0
- Emergência**:
 - ☒ Habilita Emergência
 - COB-ID: \$NODEID+16#80
- Heartbeat**:
 - ☒ Habilita geração de Heartbeat
 - Tempo de Heartbeat (ms): 200
 - Table with 3 columns: Habilita consumidor heart..., ID do nó do produtor de hear..., Tempo do Heartb...
 - Row 1: ☒ 127 300

At the bottom are 'OK' and 'Cancelar' buttons.

Descrição dos campos:

Geral:

- **ID do Nó:** O Node ID serve para identificar cada módulo CAN e corresponde ao número fixado no próprio módulo que é entre 1 e 127. Cada dispositivo da rede deve ter um Node ID exclusivo;

Node Guard

- **Habilita NodeGuard:** Se esta opção for ativada, uma mensagem será enviada para o módulo de acordo com o tempo do NodeGuard;
- **Tempo de NodeGuard (ms):** Define o tempo para envio da mensagem;
- **Fator tempo de vida:** Número de tentativas de envio da mensagem;

Emergência

- **COB-ID:** Identificador da mensagem de emergência;

Heartbeat

- **Habilita geração de Heartbeat:** Quando a opção estiver ativa, o módulo irá enviar o heartbeat de acordo com o intervalo definido no Heartbeat Producer Time;
- **Tempo de Heartbeat (ms):** Define o tempo para envio do [Heartbeat](#);
- **Habilita consumidor Heartbeat:** Quando a opção estiver ativa, verifica-se a geração do heartbeat que é enviado pelo dispositivo definido em "ID do nó do produtor de heartbeat", caso não verifique, o dispositivo entra em falha

Heartbeat

O protocolo heartbeat é um protocolo para detecção de erros.

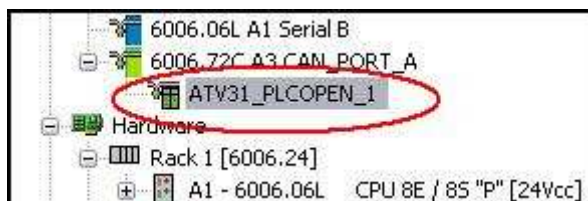
O produtor de heartbeat envia uma mensagem cíclica com baixa prioridade.

Esta mensagem é recebida e avaliada pelos consumidores de heartbeat.

Se o consumidor não receber os heartbeats, ele irá enviar uma mensagem apropriada.

Configurando um PDO

Na janela Gerenciador do Projeto, dê um duplo-clique no dispositivo a ser configurado na guia Access path, conforme a figura abaixo:



Após isso a janela abaixo será exibida para a configuração do dispositivo, selecione a guia Mapeamento do PDO para configurar os PDOs.

Mapeamento CANopen paraATV

Dispositivo Remoto CANopen | **Mapeamento do PDO** | Configuração do SDO

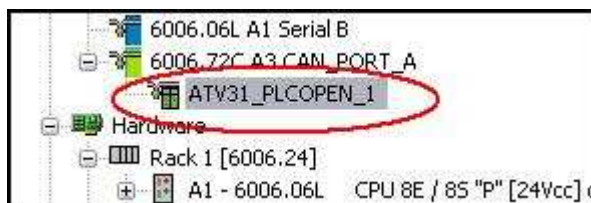
PDO de recepção				PDO de Transmissão			
Nome	Índice	Subíndice	Nº de Bits	Nome	Índice	Subíndice	Nº de Bits
☐ 1st Receive PDO	16#1400			☐ 1st Transmit PDO	16#1800		
Drivecom command reg.	16#6040	16#00	16	Drivecom status register	16#6041	16#00	16
☒ 6th Receive PDO	16#1405			☒ 6th Transmit PDO	16#1805		
Drivecom command reg.	16#6040	16#00	16	Drivecom status register	16#6041	16#00	16
Target velocity	16#6042	16#00	16	Extended status register	16#2002	16#07	16
				Control effort	16#6044	16#00	16
				Logic Input/Output image	16#2016	16#29	16

Padrão... Padrão...

OK Cancelar

Configurando um SDO

Na janela Gerenciador do Projeto, dê um duplo-clique no dispositivo a ser configurado na guia Access path, conforme a figura abaixo



Após isso a janela abaixo será exibida para a configuração do dispositivo, selecione a guia "Configuração do SDO" para configurar os SDOs.

Mapeamento CANopen paraATV

Dispositivo Remoto CANopen | Mapeamento do PDO | **Configuração do SDO**

Configuração Geral SDO

Linha	Índice:Subíndice	Nome	Valor	Nº de Bits	Tipo de Dado	Tipo de Ace...	Cancela se ...
1	16#1005:16#00	COB-ID SYNC	16#00000080	32	UDINT	RW	☐
2	16#100C:16#00	Guard Time	16#0000	16	UINT	RW	☐
3	16#100D:16#00	Life Time Factor	16#00	8	BYTE	RW	☐
4	16#1016:16#01	Consumer Heartbeat Time	16#007F01F4	32	UDINT	RW	☐
5	16#1017:16#00	Producer Heartbeat Time	16#01F4	16	UINT	RW	☐

Configuração da Comm. SDO

Linha	Índice:Subíndice	Nome	Valor	Nº de Bits	Tipo de Dado	Tipo de Ace...	Cancela se ...
1	16#1400:16#01	Disable PDO	16#80000202	32	UDINT	RW	☐
2	16#1405:16#01	Disable PDO	16#80000302	32	UDINT	RW	☐
3	16#1405:16#02	Set Transmission Type	16#01	8	BYTE	RW	☐
4	16#1605:16#00	Clear PDO Mapping	16#00	8	BYTE	RW	☐
5	16#1605:16#01	Set Mapping	16#60400010	32	UDINT	RW	☐
6	16#1605:16#02	Set Mapping	16#60420010	32	UDINT	RW	☐
7	16#1605:16#00	Set Number of PDOs	16#02	8	BYTE	RW	☐
8	16#1405:16#01	Set and Enable COB-ID	16#00000202	32	UDINT	RW	☐

SDOs s

Linha	Índice:Subíndice	Nome	Valor	Nº de Bits	Tipo de Dado	Tipo de Acesso	Cancela se hou...

Novo... Apagar Mover para cima Mover para baixo

OK Cancelar

