

LogicView for FFB



DEZ / 13
LogicView for FFB
Versão 3



smar
www.smar.com.br

Especificações e informações estão sujeitas a modificações sem prévia consulta.
Informações atualizadas dos endereços estão disponíveis em nosso site.

web: www.smar.com/brasil2/faleconosco.asp

INTRODUÇÃO

O manual de configuração, programação e aplicações do software LogicView for FFB, para os controladores DF62, DF63, DF73, DF75, DF79, DF81, DF89, DF95 e DF97 está dividido em 3 partes.

1. A Lógica Ladder: nesta parte serão descritos os elementos de uma estratégia de controle disponíveis no LogicView for FFB que usa símbolos e notações de acordo com o padrão IEC-61131-3.
2. Blocos de Função: serão apresentadas descrições detalhadas de todos os blocos de função disponíveis no LogicView for FFB.
3. O LogicView for FFB: neste capítulo o software da Smar LogicView for FFB será apresentado. Este programa é o aplicativo utilizado para implementar as lógicas ladder para controle (incluindo os elementos lógicos ladder e os blocos de função) além de configurar o hardware envolvido no controle discreto (módulos de I/O, fontes, controladores, etc.).

Sugere-se que o usuário leia inicialmente os Capítulos 1 e 2 para em seguida passar para o Capítulo 3, que descreve, de maneira clara, como implementar os elementos descritos nos dois primeiros capítulos. No entanto, nada impede que o usuário inicie a leitura do Capítulo 3 antes dos anteriores e os consulte toda vez que precisar.

O capítulo 4 tem um exemplo passo a passo de como elaborar um projeto usando o LogicView for FFB.

NOTA

Este documento é uma descrição dos blocos de função e os elementos lógicos (Elementos Ladder) que estão implementados nesta versão para os controladores DF62, DF63, DF73, DF75, DF79, DF81, DF89, DF95 e DF97. Além disso, este documento apresenta uma descrição de como configurar e editar redes lógicas Ladder através do programa aplicativo LogicView for FFB da Smar. Este documento também descreve em detalhes este software.

A Smar se reserva ao direito de alterar qualquer parte deste documento sem aviso prévio.

Lembre-se que diferentes versões desses controladores possuem diferentes tipos de dados, blocos de funções e características genéricas.

ÍNDICE

| | |
|--|----------------|
| CAPÍTULO 1 - ELEMENTOS DA REDE (ELEMENTOS LADDER)..... | 1.1 |
| OS ELEMENTOS DA REDE | 1.1 |
| DEFINIÇÕES DOS ELEMENTOS DA CAIXA DE FERRAMENTAS DA REDE (PADRÃO IEC-61131-3 - LADDER)..... | 1.1 |
| CONTATO NORMALMENTE ABERTO | 1.1 |
| CONTATO NORMALMENTE FECHADO | 1.1 |
| CONTATO SENSÍVEL À TRANSIÇÃO DE SUBIDA..... | 1.1 |
| CONTATO SENSÍVEL À TRANSIÇÃO DE DESCIDA | 1.1 |
| BOBINA | 1.1 |
| BOBINA INVERTIDA | 1.1 |
| SET DE BOBINA..... | 1.1 |
| RESET DE BOBINA..... | 1.1 |
| BOBINA SENSÍVEL À TRANSIÇÃO DE SUBIDA..... | 1.1 |
| BOBINA SENSÍVEL À TRANSIÇÃO DE DESCIDA | 1.2 |
| BOBINA COM RESET RETENTIVO..... | 1.2 |
| BOBINA COM SET RETENTIVO (DE MEMÓRIA) | 1.2 |
| CONEXÃO HORIZONTAL | 1.2 |
| CONEXÃO VERTICAL..... | 1.2 |
| ELIMINA CONEXÃO VERTICAL | 1.2 |
| ELIMINA OBJETO | 1.2 |
| SELEÇÃO | 1.2 |
| INSERE NOTA..... | 1.2 |
| DEFINIÇÕES DOS ELEMENTOS DA CAIXA DE FERRAMENTAS DA REDE (PADRÃO IEC-61131-3 - OUTRAS LINGUAGENS)..... | 1.3 |
| CONTATO NORMALMENTE ABERTO | 1.3 |
| BOBINA | 1.3 |
| LÓGICA BOOLEANA | 1.4 |
| CONTATO NORMALMENTE ABERTO | 1.4 |
| CONTATO NORMALMENTE FECHADO | 1.4 |
| FUNÇÃO LÓGICA OR (OU) | 1.4 |
| FUNÇÃO LÓGICA AND (E)..... | 1.4 |
| EQUAÇÕES BOOLEANAS..... | 1.5 |
| ÁLGEBRA DE BOOLE | 1.5 |
| CAPÍTULO 2 - BLOCOS FUNCIONAIS | 2.1 |
| INTRODUÇÃO | 2.1 |
| A ENTRADA EN E A SAÍDA EO | 2.1 |
| BLOCOS FUNCIONAIS DISPONÍVEIS EM ORDEM ALFABÉTICA | 2.2 |
| BLOCOS FUNCIONAIS LISTADOS POR GRUPOS FUNCIONAIS..... | 2.4 |
| FUNÇÕES RELACIONADAS POR TEMPO/CONTADOR..... | 2.4 |
| FUNÇÕES DE MANIPULAÇÃO DE DADOS | 2.4 |
| FUNÇÕES MATEMÁTICAS..... | 2.5 |
| FUNÇÕES DE COMPARAÇÃO..... | 2.5 |
| FUNÇÕES DE CONTROLÉ DE PROCESSO | 2.5 |
| FUNÇÕES DE ENTRADAS E SAÍDAS..... | 2.6 |
| FUNÇÕES RELACIONADAS POR TEMPO/CONTADOR | 2.7 |
| ACUMULADOR DE TEMPO (ACMT) | 2.7 |
| ACUMULADOR DE TEMPO REDUZIDO (ACMTR) | 2.8 |
| ACUMULADOR DE TEMPO REDUZIDO (ACMTH) | 2.9 |
| CONTADOR DECRESCENTE DE PULSOS (CDN)..... | 2.10 |
| CONTADOR DECRESCENTE DE PULSOS REDUZIDO (CDNR)..... | 2.11 |
| CONTADOR CRESCENTE/ DECRESCENTE DE PULSOS (CTUD)..... | 2.12 |
| CONTADOR CRESCENTE/ DECRESCENTE DE PULSOS REDUZIDO (CUDR)..... | 2.13 |
| CONTADOR CRESCENTE DE PULSOS (CUP) | 2.14 |
| CONTADOR CRESCENTE DE PULSOS REDUZIDO (CUPR)..... | 2.15 |
| CONTADOR CRESCENTE DE PULSOS REDUZIDO 2 (CTUR) | 2.16 |
| RESET SET (RS)..... | 2.17 |
| RESET SET REDUZIDO (RSR)..... | 2.18 |
| ALARME EM TEMPO REAL (RTA) | 2.19 |
| SET RESET (SR)..... | 2.21 |

| | |
|--|-------------|
| SET RESET REDUZIDO (SRR)..... | 2.22 |
| TEMPO DE ATRASO PARA DESLIGAR (TOF) | 2.23 |
| TEMPO DE ATRASO PARA DESLIGAR REDUZIDO (TOFR) | 2.24 |
| TEMPO DE ATRASO PARA LIGAR (TON) | 2.25 |
| TEMPO DE ATRASO PARA LIGAR REDUZIDO (TONR) | 2.26 |
| TEMPORIZADOR DE PULSO (TP) | 2.27 |
| TEMPORIZADOR DE PULSOS REDUZIDO (TPR)..... | 2.28 |
| FUNÇÕES DE MANIPULAÇÃO DE DADOS..... | 2.29 |
| CONVERSÃO DE BYTE PARA INTEIRO (BINT) | 2.29 |
| CONVERSÃO DE BYTE PARA BITS (BTB)..... | 2.30 |
| CONVERSÃO DE ENTRADA BOOLEANA PARA INTEIRO (BT1) | 2.31 |
| CONVERSÃO DE BCD PARA INTEIRO (BTI2)..... | 2.32 |
| OPERAÇÃO LÓGICA BINÁRIA (BWL1)..... | 2.33 |
| OPERAÇÃO LÓGICA BINÁRIA REDUZIDA (BWL1R) | 2.35 |
| OPERAÇÃO LÓGICA BIT A BIT (BWL2)..... | 2.37 |
| OPERAÇÃO LÓGICA BIT A BIT REDUZIDA (BWL2R)..... | 2.39 |
| CONSTANTES (CONST)..... | 2.41 |
| CONVERSÃO DE INTEIRO PARA BOOLEANO (ITB1) | 2.42 |
| CONVERSÃO DE INTEIRO PARA BCD (ITB2)..... | 2.43 |
| CONVERSOR DE FLOAT/LONG PARA LONG (LONG)..... | 2.44 |
| MULTIPLEXADOR PARA ENTRADAS BOOLEANAS (MUX1) | 2.45 |
| MULTIPLEXADOR PARA ENTRADAS BOOLEANAS REDUZIDO (MUX1R) | 2.46 |
| MULTIPLEXADOR PARA ENTRADAS FLOAT (MUX2)..... | 2.47 |
| MULTIPLEXADOR PARA ENTRADAS FLOAT REDUZIDO (MUX2R)..... | 2.48 |
| OPERAÇÃO NOT PARA UMA ENTRADA BOOLEANA (NOT1) | 2.49 |
| OPERAÇÃO NOT BIT A BIT (NOT2)..... | 2.50 |
| SELEÇÃO BINÁRIA DAS SAÍDAS (OSEL) | 2.51 |
| SELEÇÃO BINÁRIA PARA ENTRADAS BOOLEANAS (SEL1)..... | 2.52 |
| SELEÇÃO BINÁRIA PARA ENTRADAS FLOAT (SEL2)..... | 2.53 |
| TRUNCAGEM (TRC) | 2.54 |
| FUNÇÕES MATEMÁTICAS..... | 2.55 |
| VALOR ABSOLUTO (ABS)..... | 2.55 |
| ADIÇÃO (ADD) | 2.56 |
| ADIÇÃO REDUZIDA (ADDR)..... | 2.57 |
| OPERAÇÃO LÓGICA AND DE 2 A 8 ENTRADAS (AND2-AND8) | 2.58 |
| DIVISÃO (DIV) | 2.59 |
| MÓDULO (MDL) | 2.60 |
| MULTIPLICAÇÃO (MUL) | 2.61 |
| MULTIPLICAÇÃO REDUZIDA (MULR) | 2.62 |
| INVERSOR BOOLEANO (NOT) | 2.63 |
| OPERAÇÃO LÓGICA OR DE 2 A 8 ENTRADAS (OR2-OR8)..... | 2.64 |
| SUBTRAÇÃO (SBT) | 2.65 |
| RAIZ QUADRADA (SQR) | 2.66 |
| FUNÇÕES DE COMPARAÇÃO..... | 2.67 |
| QUADRUPLO ALARME (AI-SETA) | 2.67 |
| ALARME DUPLO (ALM) | 2.69 |
| DESIGUALDADE (DIF)..... | 2.71 |
| IGUALDADE (EQ)..... | 2.72 |
| IGUALDADE REDUZIDA (EQR)..... | 2.74 |
| SEQUÊNCIA DECRESCENTE (GT)..... | 2.75 |
| SEQUÊNCIA DECRESCENTE REDUZIDA (GTR)..... | 2.76 |
| SEQUÊNCIA MONOTÔNICA DECRESCENTE (GTE)..... | 2.77 |
| SEQUÊNCIA MONOTÔNICA DECRESCENTE REDUZIDA (GTER)..... | 2.79 |
| LIMITADOR (LMT) | 2.80 |
| SEQUÊNCIA CRESCENTE (LT) | 2.81 |
| SEQUÊNCIA CRESCENTE REDUZIDO (LTR) | 2.82 |
| SEQUÊNCIA MONOTÔNICA CRESCENTE (LTE) | 2.83 |
| SEQUÊNCIA MONOTÔNICA CRESCENTE REDUZIDA (LTER)..... | 2.85 |
| MÁXIMO (MAX) | 2.86 |
| MÁXIMO REDUZIDO (MAXR) | 2.87 |
| MÍNIMO (MIN)..... | 2.88 |
| MÍNIMO REDUZIDO (MINR)..... | 2.89 |
| FUNÇÕES DE CONTROLE DE PROCESSOS..... | 2.90 |
| CONTROLADOR PID AVANÇADO (APID) | 2.90 |
| RAMPA AUTOMÁTICA COM INCREMENTO E DECREMENTO (ARAMP)..... | 2.100 |

| | |
|---|--------------|
| CONTROLADOR PID OTIMIZADO (EPID)..... | 2.102 |
| TOTALIZAÇÃO OTIMIZADA (ETOT)..... | 2.110 |
| LINEARIZAÇÃO (LIN)..... | 2.113 |
| LEAD LAG (LLAG)..... | 2.115 |
| EQUAÇÃO MATEMÁTICA PARA PROCESSAMENTO DE SINAIS (MATH)..... | 2.118 |
| CONTROLADOR PID..... | 2.120 |
| COMPENSAÇÃO DE TEMPERATURA E PRESSÃO (PTC)..... | 2.123 |
| SAMPLE HOLD COM INCREMENTO E DECREMENTO (SMPL)..... | 2.124 |
| GERADOR DE SET POINT (SPG)..... | 2.125 |
| CONTROLE STEP (STP)..... | 2.128 |
| TOTALIZAÇÃO (TOT)..... | 2.130 |
| CONTROLE DE ABERTURA E FECHAMENTO DE VÁLVULAS (VDA-OC)..... | 2.132 |
| LIMITE CRUZADO E VELOCIDADE DE VARIAÇÃO (XLIM)..... | 2.134 |
| FUNÇÕES DE ENTRADAS E SAÍDAS..... | 2.136 |
| ACUMULADOR DE PULSOS (ACC)..... | 2.136 |
| ACUMULADOR DE PULSOS MÚLTIPLO (ACC_N)..... | 2.138 |
| ENTRADA ANALÓGICA SIMPLES (AI)..... | 2.140 |
| ENTRADAS ANALÓGICAS PARA EQUIPAMENTOS HART (AIH)..... | 2.142 |
| SAÍDAS ANALÓGICAS PARA EQUIPAMENTO HART (AOH)..... | 2.143 |
| MÚLTIPLAS ENTRADAS ANALÓGICAS (MAI)..... | 2.144 |
| MÚLTIPLAS ENTRADAS ANALÓGICAS PARA IOR OU HART (MAIX)..... | 2.145 |
| MÚLTIPLAS SAÍDAS ANALÓGICAS (MAO)..... | 2.146 |
| MÚLTIPLAS SAÍDAS ANALÓGICAS PARA IOR OU HART (MAOX)..... | 2.147 |
| STATUS DO SISTEMA (STATUS)..... | 2.148 |
| STATUS PARA VARIÁVEIS HART (STSH)..... | 2.150 |
| TEMPERATURA (TEMP)..... | 2.152 |
| | |
| CAPÍTULO 3 - O LOGICVIEW FOR FFB..... | 3.1 |
| INTRODUÇÃO..... | 3.1 |
| INSTALAÇÃO..... | 3.1 |
| LICENÇA..... | 3.1 |
| USANDO O LOGICVIEW FOR FFB..... | 3.3 |
| INICIANDO UMA APLICAÇÃO..... | 3.3 |
| MODO INSTANCE..... | 3.4 |
| MODO TEMPLATE..... | 3.9 |
| MODO SUPERVISION ONLY..... | 3.11 |
| MODO SIMULATION..... | 3.12 |
| MODO VIEW..... | 3.13 |
| AVALIAÇÃO DA REDE LADDER..... | 3.13 |
| CONHECENDO A ÁREA DE TRABALHO..... | 3.14 |
| MENU PRINCIPAL..... | 3.15 |
| MENU FILE..... | 3.15 |
| CREATE TEMPLATE..... | 3.16 |
| EXPORT TAGS FOR OPC BROWSING..... | 3.17 |
| EXPORT TAGS FOR SUPERVISION..... | 3.17 |
| OPÇÕES DE IMPRESSÃO..... | 3.18 |
| MENU EDIT..... | 3.24 |
| METAPARÂMETROS..... | 3.33 |
| MENU VIEW..... | 3.40 |
| MENU LADDER..... | 3.42 |
| MENU TOOLS..... | 3.44 |
| ATRIBUIÇÃO DOS ENDEREÇOS MODBUS..... | 3.47 |
| BIBLIOTECA DE LÓGICAS..... | 3.55 |
| MENU HELP..... | 3.62 |
| BARRAS DE FERRAMENTAS..... | 3.63 |
| MAIN BAR..... | 3.63 |
| ZOOM BAR..... | 3.63 |
| TOOLBOX..... | 3.64 |
| ABA COMMUNICATION..... | 3.74 |
| HIERARCHY..... | 3.84 |
| INFORMAÇÕES SOBRE O PROJETO..... | 3.84 |
| HARDWARE..... | 3.85 |
| PROGRAMS..... | 3.107 |
| VIRTUALS..... | 3.109 |

| | |
|--|------------|
| FF BLOCK DEFINITION | 3.110 |
| OBJECT PROPERTIES | 3.111 |
| ÁREA DE DESENHO DA LADDER | 3.113 |
| INSERT/REMOVE BLANK LINE | 3.113 |
| OUTPUT | 3.115 |
| STATUS BAR | 3.116 |
| CAPÍTULO 4 - EXEMPLO DE UTILIZAÇÃO DO LOGICVIEW FOR FFB | 4.1 |
| DESCRIÇÃO DO PROCESSO | 4.1 |
| IMPLEMENTANDO A LÓGICA | 4.1 |
| CONFIGURANDO O HARDWARE | 4.1 |
| DESENHANDO A LADDER | 4.3 |
| SIMULAÇÃO DO ALARME COM A OPÇÃO SIMULATION | 4.4 |

ELEMENTOS DA REDE (ELEMENTOS LADDER)

Esta seção apresentará o significado dos elementos Ladder da rede.

Os Elementos da Rede

O LogicView for FFB usa símbolos e notações como definido no padrão IEC - 61131-3 e alguns adicionais usados em outras linguagens além de Ladder.

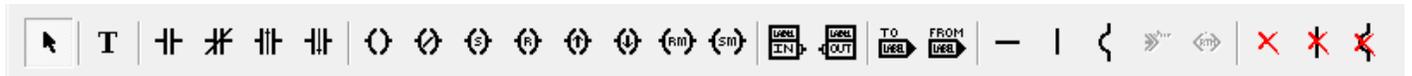


Fig 1.1 - Caixa de Ferramentas da Rede

Definições dos Elementos da Caixa de Ferramentas da Rede (Padrão IEC-61131-3 - Ladder)



Contato Normalmente Aberto

O estado do link esquerdo será copiado para o link direito se o estado da variável estiver ON (Ligado). Caso contrário, o link direito será OFF (Desligado).



Contato Normalmente Fechado

O estado do link esquerdo será copiado para o link direito se o estado da variável estiver OFF. Caso contrário, o link direito será OFF.



Contato Sensível à Transição de Subida

O estado do link direito será ON se o link esquerdo estiver ON e ocorrer uma subida do nível lógico (de OFF para ON). O estado do link direito será OFF para qualquer outra situação.



Contato Sensível à Transição de Descida

O estado do link direito será ON se o link esquerdo estiver ON e ocorrer uma descida do nível lógico (de ON para OFF). O estado do link direito será OFF para qualquer outra situação.



Bobina

O estado do link esquerdo será copiado para a variável booleana e para o link direito.



Bobina Invertida

O estado do link esquerdo será copiado para o link da direita. O inverso do estado do link esquerdo é copiado para a variável booleana associada, isto é, se o estado do link esquerdo estiver OFF, então o estado da variável será ON, e vice-versa.



Set de Bobina

A variável booleana será colocada em ON quando o link esquerdo estiver em ON, e permanecerá assim até que se tenha um *reset* através de uma bobina RESET.



Reset de Bobina

A variável booleana associada é colocada em OFF quando o link esquerdo está em ON, e permanece assim até que seja ajustada através de uma bobina SET.



Bobina Sensível à Transição de Subida

O estado da variável booleana será ON se o link esquerdo sofrer uma variação de OFF para ON. O estado do link esquerdo sempre será copiado para o link direito.



Bobina Sensível à Transição de Descida

O estado da variável booleana será ON se o link esquerdo sofrer uma variação de ON para OFF. O estado do link esquerdo sempre será copiado para o link direito.



Bobina com Reset Retentivo

O estado da variável booleana associada será colocado em OFF quando o link esquerdo estiver em ON e permanecerá assim até que seja ajustada através de uma bobina SET. A variável booleana associada será guardada na memória.

Observação: A ação desta bobina é idêntica à da bobina RESET (Destrava), exceto que a variável booleana associada é automaticamente gravada na memória.



Bobina com Set Retentivo (de Memória)

A variável booleana associada será ajustada para ON quando o link esquerdo estiver em ON, e permanece assim até que seja feito um *reset* através de uma bobina RESET. A variável booleana associada será guardada na memória.

Observação: A ação desta bobina retentiva de ajuste é idêntica à da bobina de ajuste (SET), exceto que a variável booleana associada é automaticamente gravada na memória.



Conexão Horizontal

Use esta ferramenta para traçar uma linha de ligação da esquerda para a direita na célula marcada.



Conexão Vertical

Use esta ferramenta para traçar uma linha de ligação (segmento para baixo) do lado direito da célula marcada.



Elimina Conexão Vertical

Elimina conexão vertical. Para efetuar esta ação é necessário posicionar o quadro de seleção no elemento o qual possui a linha vertical.



Elimina Objeto

Use esta ferramenta para eliminar um objeto inserido na célula. Tem a mesma função do botão "Delete" do teclado.



Seleção

Use esta ferramenta para selecionar um elemento de rede – contato ou bobina. O elemento selecionado ficará vermelho.



Inserir Nota

Use esta ferramenta para inserir uma nota (texto) na célula. Para selecionar a nota, basta clicar nela com o botão "Seleção" descrito acima e ela ficará vermelha. Depois de selecionada, ela poderá ser removida através do botão "Delete" do teclado ou poderá ser movida, clicando e arrastando o mouse. A nota inserida ficará com o texto na definida em **Tools**→**Options**→**Interface**.

Definições dos Elementos da Caixa de Ferramentas da Rede (Padrão IEC-61131-3 - Outras linguagens)



Contato Normalmente Aberto

O estado do link esquerdo será copiado para o link direito se o estado da variável estiver ON (Ligado). Caso contrário, o link direito será OFF (Desligado).



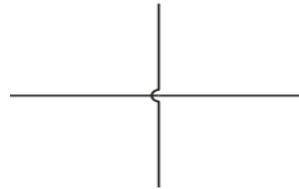
Bobina

O estado do link esquerdo será copiado para a variável booleana e para o link direito.



Gap Wire

Use esta ferramenta para traçar uma linha de ligação *gap wire* do lado direito da célula marcada, uma linha vertical que passa sobre uma horizontal, sem que ocorra um cruzamento entre elas, ou seja, o fluxo na vertical, não influencia o fluxo na horizontal e vice-versa. Veja o exemplo a seguir:



NOTA

Para que esta função tenha efeito, é necessário que exista uma linha vertical previamente no local em que se deseja inserir o *gap wire*.



Elimina Gap Wire

Para eliminar o *gap wire* é necessário posicionar o quadro de seleção no elemento que o possui.

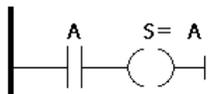
NOTA

Ao eliminar o *gap wire*, ele se transforma automaticamente numa linha vertical normal.

Lógica Booleana

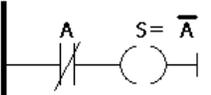
A associação de contatos e bobinas gera funções booleanas. Abaixo, apresenta-se um breve resumo sobre estas funções e álgebra de Boole.

Contato Normalmente Aberto

| Esquema | Tabela de estados | |
|---|-------------------|---|
|  | A | S |
| | 0 | 0 |
| | 1 | 1 |

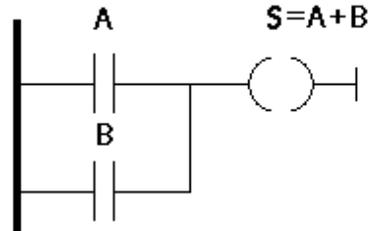
Quando o estado de A muda de 0 para 1 o contato A é fechado e o fluxo de força passa do Power Rail, que está na esquerda, para a direita energizando a bobina S.

Contato Normalmente Fechado

| Esquema | Tabela de estados | |
|---|-------------------|---|
|  | A | S |
| | 0 | 1 |
| | 1 | 0 |

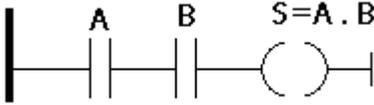
O contato A é normalmente fechado. Isto é, o fluxo de força passará por A energizando S até que o valor de A mude de 0 para 1. Desta forma a bobina S tem comportamento inverso comparada à bobina do item anterior (contato normalmente aberto).

Função Lógica OR (OU)

| Esquema | Tabela de estados | | |
|---|-------------------|---|---|
|  | A | B | S |
| | 0 | 0 | 0 |
| | 0 | 1 | 1 |
| | 1 | 0 | 1 |
| | 1 | 1 | 1 |

Os contatos A e B são normalmente abertos. Associando dois contatos em paralelo implementa-se a função OR (Ou Lógico). A bobina será energizada quando qualquer um dos dois contatos estiver fechado.

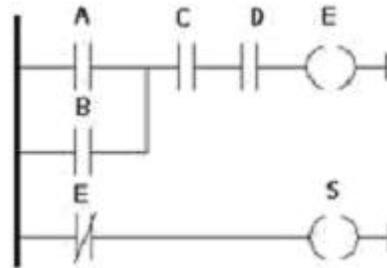
Função Lógica AND (E)

| Esquema | Tabela de estados | | |
|---|-------------------|---|---|
|  | A | B | S |
| | 0 | 0 | 0 |
| | 0 | 1 | 0 |
| | 1 | 0 | 0 |
| | 1 | 1 | 1 |

Os contatos A e B são normalmente abertos. A bobina S apenas será energizada quando A e B ao mesmo tempo forem iguais a 1. Caso contrário o fluxo de força não passará do lado esquerdo (Power Rail) até o lado direito.

Equações Booleanas

Utilizando contatos e bobinas pode-se implementar funções booleanas. Por exemplo, considere o diagrama ladder abaixo:



A saída S dependerá do estado dos contatos A, B, C, D e do estado da bobina E. O valor do estado C depende de A e B. Assim a função lógica que descreve o circuito acima é:

$$E = (A + B).C.D$$

$$S = \bar{E}$$

Álgebra de Boole

Equações booleanas, como mostradas acima, podem se tornar bastante complexas, porém o resultado pode ser simplificado utilizando a álgebra de boole. Abaixo é mostrado um resumo das propriedades da álgebra de Boole.

| | |
|----|--------------------------------------|
| 1 | $A.1 = A$ |
| 2 | $A.0 = 0$ |
| 3a | $A.A = A$ |
| 3b | $A\bar{A} = 0$ |
| 4a | $A + \bar{A} = 1$ |
| 4b | $A + A = A$ |
| 5 | $A + 1 = 1$ |
| 6 | $A.B + A.C = A.(B + C)$ |
| 7 | $A + A.B = A$ |
| 8 | $A.(B + C) = A.B + A.C$ |
| 9a | $\overline{A + B} = \bar{A}.\bar{B}$ |
| 9b | $\overline{A.B} = \bar{A} + \bar{B}$ |

Quando as expressões se tornarem muito complexas sugere-se utilizar o mapa de Karnaugh para simplificá-las. Esta informação pode ser encontrada em qualquer livro de eletrônica digital.

BLOCOS FUNCIONAIS

Introdução

Esta é uma referência completa e atualizada dos blocos funcionais. Aqui são apresentados diagramas de blocos com entradas, saídas e parâmetros de configuração. Inclui também, explicações detalhadas, funcionamento e configuração de cada bloco. Alguns exemplos são apresentados para facilitar o entendimento e utilização por parte do usuário.

Os tipos de dados que são utilizados pelo **LogicView for FFB** estão na tabela abaixo:

| Referência | Tipo de dados | Número de bits |
|------------|---------------|--------------------|
| BOOL | Booleano | 1 |
| LONG | Inteiro | 32 não sinalizados |
| FLOAT | Número Real | 32 |

Cada bloco funcional possui uma tabela onde são apresentados as entradas de cada bloco, as saídas e os parâmetros.

- I - Entradas: Podem ser uma variável de um outro bloco, ser proveniente de um cartão de E/S ou ser configurada pelo usuário manualmente.
- P- Parâmetros: são valores usados internamente pelos blocos funcionais.
- O - Saídas: são resultados do processamento do bloco.

ATENÇÃO

A vírgula “,” não é aceita na indicação de números com casas decimais (valores float). O usuário deve sempre utilizar o ponto “.” nesses casos. Por exemplo, para representar 9/5 deve-se escrever 1.8 e NÃO 1,8. Se for escrito 1,8 o software entenderá 18.

A entrada EN e a saída EO

Todas as funções têm entrada **EN** e saída **EO** com exceção daquelas com subíndice “r”, por exemplo, TPr, e da função CTUr que tem somente a entrada **EN**.

A entrada **EN** é usada para habilitar a função a ser processada. Se a entrada **EN** é falsa, todas as saídas irão para zero (falso) e a função não será executada.

A saída **EO** vai para um (verdadeiro) para indicar que a função foi executada com êxito, sem nenhum problema.

Blocos Funcionais Disponíveis em Ordem Alfabética

Abaixo estão listadas todas as funções.

| NOME DA FUNÇÃO | DESCRIÇÃO |
|----------------|--|
| ABS | Valor Absoluto |
| ACC | Acumulador de Pulsos |
| ACC_N | Acumulador de Pulsos Múltiplo |
| ACMT | Acumulador de Tempo |
| ACMTh | Acumulador de Tempo Reduzido |
| ACMTr | Acumulador de Tempo Reduzido |
| ADD | Adição |
| ADDr | Adição Reduzida |
| AI | Entrada Analógica Simples |
| AIh | Entradas Analógicas para Equipamento HART |
| AI-Seta | Quadruplo Alarme |
| ALM | Alarme Duplo |
| AND2-AND8 | Operação Lógica AND de 2 a 8 entradas |
| AOh | Saídas Analógicas para Equipamento HART |
| APID | Controlador PID Avançado |
| ARAMP | Rampa Automática com Incremento e Decremento |
| BINT | Conversão de Byte para Inteiro |
| BTB | Conversão de Byte para Bits |
| BTI1 | Conversão de Entrada Booleana para Inteiro |
| BTI2 | Conversão de BCD para Inteiro |
| BWL1 | Operação Lógica Binária |
| BWL1r | Operação Lógica Binária Reduzida |
| BWL2 | Operação Lógica Bit a Bit |
| BWL2r | Operação Lógica Bit a Bit Reduzida |
| CDN | Contador Decrescente |
| CDNr | Contador Decrescente Reduzido |
| CONST | Constantes |
| CUDr | Contador Crescente/Decrescente Reduzido |
| CUP | Contador Crescente |
| CUPr | Contador Crescente Reduzido |
| CTUD | Contador Crescente/Decrescente |
| CTUr | Contador Crescente de Pulsos Reduzido 2 |
| DIF | Desigualdade |
| DIV | Divisão |
| EPID | Controlador PID Otimizado |
| EQ | Igualdade |
| EQr | Igualdade Reduzida |
| ETOT | Totalização Otimizada |
| GT | Sequência Decrescente |
| GTr | Sequência Decrescente Reduzida |
| GTE | Sequência Monotônica Decrescente |
| GTEr | Sequência Monotônica Decrescente Reduzida |
| ITB1 | Conversão de Inteiro para Booleano |
| ITB2 | Conversão de Inteiro para BCD |
| LIN | Linearização |
| LLAG | Lead Lag |
| LMT | Limitador |
| LONG | Conversor de Float/Long para Long |
| LT | Sequência Crescente |
| LTr | Sequência Crescente Reduzida |
| LTE | Sequência Monotônica Crescente |
| LTEr | Sequência Monotônica Crescente Reduzida |
| MAI | Múltiplas Entradas Analógicas |
| MAIx | Múltiplas Entradas Analógicas para IOR ou HART |
| MAO | Múltiplas Saídas Analógicas |
| MAOx | Múltiplas Saídas Analógicas para IOR ou HART |

| NOME DA FUNÇÃO | DESCRIÇÃO |
|----------------|---|
| MATH | Equação Matemática para Processamento de Sinais |
| MAX | Máximo |
| MAXr | Máximo Reduzido |
| MDL | Módulo |
| MIN | Mínimo |
| MINr | Mínimo Reduzido |
| MUL | Multiplicação |
| MULr | Multiplicação Reduzida |
| MUX1 | Multiplexador para Entradas Booleanas |
| MUX1r | Multiplexador para Entradas Booleanas Reduzido |
| MUX2 | Multiplexador para Entradas Float |
| MUX2r | Multiplexador para Entradas Float Reduzido |
| NOT | Inversor Booleano |
| NOT1 | Operação NOT para Entrada Booleana |
| NOT2 | Operação NOT Bit a Bit |
| OR2-OR8 | Operação Lógica OR de 2 a 8 entradas |
| OSEL | Seleção Binária das Saídas |
| PID | Controlador PID |
| PTC | Compensação de Pressão e Temperatura |
| RS | Reset Set |
| RSr | Reset Set Reduzido |
| RTA | Alarme de Relógio em Tempo Real |
| SBT | Subtração |
| SEL1 | Seleção Binária para Entradas Booleanas |
| SEL2 | Seleção Binária para Entradas Float |
| SMPL | Sample Hold com Incremento e Decremento |
| SPG | Gerador de Set Point |
| SQR | Raiz Quadrada |
| SR | Set Reset |
| SRr | Set Reset Reduzido |
| STATUS | Status do Sistema |
| STP | Controle Step |
| STSh | Status para variáveis HART |
| TEMP | Temperatura |
| TOF | Tempo de Atraso para Desligar |
| TOFr | Tempo de Atraso para Desligar Reduzido |
| TON | Tempo de Atraso para Ligar |
| TONr | Tempo de Atraso para Ligar Reduzido |
| TOT | Totalização |
| TP | Pulso do Timer |
| TPr | Pulso do Timer Reduzido |
| TRC | Truncagem |
| VDA-OC | Controle de abertura e fechamento de Válvula |
| XLIM | Limite Cruzado e Velocidade de Variação |

Blocos Funcionais Listados por Grupos Funcionais

Funções Relacionadas por Tempo/Contador

| MNEMÔNICO | DESCRIÇÃO |
|-----------|---|
| ACMT | Acumulador de Tempo |
| ACMTr | Acumulador de Tempo Reduzido |
| ACMTh | Acumulador de Tempo Reduzido |
| CDN | Contador Decrescente |
| CDNr | Contador Decrescente Reduzido |
| CUP | Contador Crescente |
| CUPr | Contador Crescente Reduzido |
| CUDr | Contador Crescente/Decrescente Reduzido |
| CTUD | Contador Crescente/Decrescente |
| CTUr | Contador Crescente de Pulsos Reduzido 2 |
| RS | Reset Set |
| RSr | Reset Set Reduzido |
| RTA | Alarme de Relógio em Tempo Real |
| SR | Set Reset |
| SRr | Set Reset Reduzido |
| TOF | Tempo de Atraso para Desligar |
| TOFr | Tempo de Atraso para Desligar Reduzido |
| TON | Tempo de Atraso para Ligar |
| TONr | Tempo de Atraso para Ligar Reduzido |
| TP | Pulso do Timer |
| TPr | Pulso do Timer Reduzido |

Funções de Manipulação de Dados

| MNEMÔNICO | DESCRIÇÃO |
|-----------|--|
| BINT | Conversão de Byte para Inteiro |
| BTB | Conversão de Byte para Bits |
| BTI1 | Conversão de Entrada Booleana para Inteiro |
| BTI2 | Conversão de BCD para Inteiro |
| BWL1 | Operação Lógica Binária |
| BWL1r | Operação Lógica Binária Reduzida |
| BWL2 | Operação Lógica Bit a Bit |
| BWL2r | Operação Lógica Bit a Bit Reduzida |
| CONST | Constantes |
| ITB1 | Conversão de Inteiro para Booleano |
| ITB2 | Conversão de Inteiro para BCD |
| LONG | Conversor de Float/Long para Long |
| MUX1 | Multiplexador para Entradas Booleanas |
| MUX1r | Multiplexador para Entradas Booleanas Reduzido |
| MUX2 | Multiplexador para Entradas Float |
| MUX2r | Multiplexador para Entradas Float Reduzido |
| NOT1 | Operação NOT para Entrada Booleana |
| NOT2 | Operação NOT Bit a Bit |
| OSEL | Seleção Binária das Saídas |
| SEL1 | Seleção Binária para Entradas Booleanas |
| SEL2 | Seleção Binária para Entradas Float |
| TRC | Truncagem |

Funções Matemáticas

| MNEMÔNICO | DESCRIÇÃO |
|-----------|---------------------------------------|
| ABS | Valor Absoluto |
| ADD | Adição |
| ADDr | Adição Reduzida |
| AND2-AND8 | Operação Lógica AND de 2 a 8 entradas |
| DIV | Divisão |
| MDL | Módulo |
| MUL | Multiplicação |
| MULr | Multiplicação Reduzida |
| NOT | Inversor Booleano |
| OR2-OR8 | Operação Lógica OR de 2 a 8 entradas |
| SBT | Subtração |
| SQR | Raiz Quadrada |

Funções de Comparação

| MNEMÔNICO | DESCRIÇÃO |
|-----------|---|
| AI-Seta | Quadruplo Alarme |
| ALM | Alarme Duplo |
| DIF | Desigualdade |
| EQ | Igualdade |
| EQr | Igualdade Reduzida |
| GT | Sequência Decrescente |
| GTr | Sequência Decrescente Reduzida |
| GTE | Sequência Monotônica Decrescente |
| GTEr | Sequência Monotônica Decrescente Reduzida |
| LMT | Limitador |
| LT | Sequência Crescente |
| LTr | Sequência Crescente Reduzida |
| LTE | Sequência Monotônica Crescente |
| LTEr | Sequência Monotônica Crescente Reduzida |
| MAX | Máximo |
| MAXr | Máximo Reduzido |
| MIN | Mínimo |
| MINr | Mínimo Reduzido |

Funções de Controle de Processo

| MNEMÔNICO | DESCRIÇÃO |
|-----------|---|
| APID | Controlador PID Avançado |
| ARAMP | Rampa Automática com Incremento e Decremento |
| EPID | Controlador PID Otimizado |
| ETOT | Totalização Otimizada |
| LIN | Linearização |
| LLAG | Lead Lag |
| MATH | Equação Matemática para Processamento de Sinais |
| PID | Controlador PID |
| PTC | Compensação de Pressão e Temperatura |
| SMPL | Sample Hold com Incremento e Decremento |
| SPG | Gerador de Set Point |
| STP | Controle Step |
| TOT | Totalização |
| VDA-OC | Controle de abertura e fechamento de Válvula |
| XLIM | Limite Cruzado e Velocidade de Variação |

Funções de Entradas e Saídas

| MNEMÔNICO | DESCRIÇÃO |
|-----------|--|
| ACC | Acumulador de Pulsos |
| ACC_N | Acumulador de Pulsos Múltiplo |
| AI | Entrada Analógica Simples |
| AIh | Entradas Analógicas para Equipamentos HART |
| AOh | Saídas Analógicas para Equipamentos HART |
| MAI | Múltiplas Entradas Analógicas |
| MAIx | Múltiplas Entradas Analógicas para IOR ou HART |
| MAO | Múltiplas Saídas Analógicas |
| MAOx | Múltiplas Saídas Analógicas para IOR ou HART |
| STATUS | Status do Sistema |
| STSh | Status para variáveis HART |
| TEMP | Temperatura |

Funções Relacionadas por Tempo/Contador

Acumulador de Tempo (ACMT)

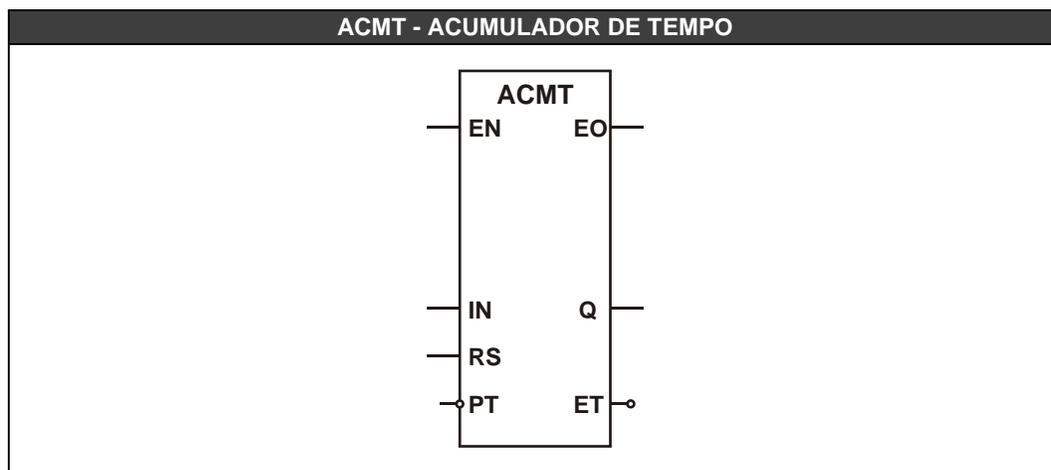
Descrição

Este bloco funcional, quando a entrada **EN** está verdadeira, e com a entrada **IN** em verdadeiro, acumula o tempo em que **IN** permanece verdadeiro.

Se **IN** muda para falso, a contagem do tempo é congelada até que **IN** volte para verdadeiro. Alcançado o tempo definido em **PT**, a saída **Q** muda para verdadeiro. O tempo é dado em milissegundos.

Se a entrada **RS** muda para verdadeiro as saídas **Q** e **ET** são zeradas. O valor em **RS** prevalece sobre o valor de **IN**.

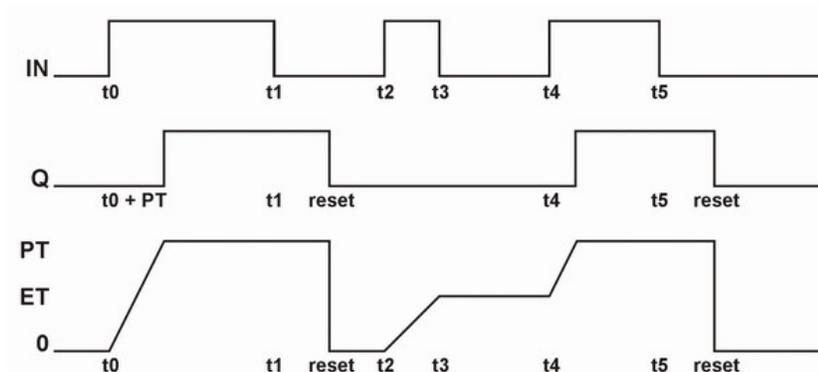
Se a entrada **EN** está falsa, todas as saídas são mantidas em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--------------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA DE PULSOS | BOOL |
| | RS | RESET DO BLOCO | BOOL |
| | PT | TEMPO PROGRAMADO | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | Q | SAÍDA DO BLOCO | BOOL |
| | ET | TEMPO TRANSCORRIDO DA CONTAGEM | LONG |

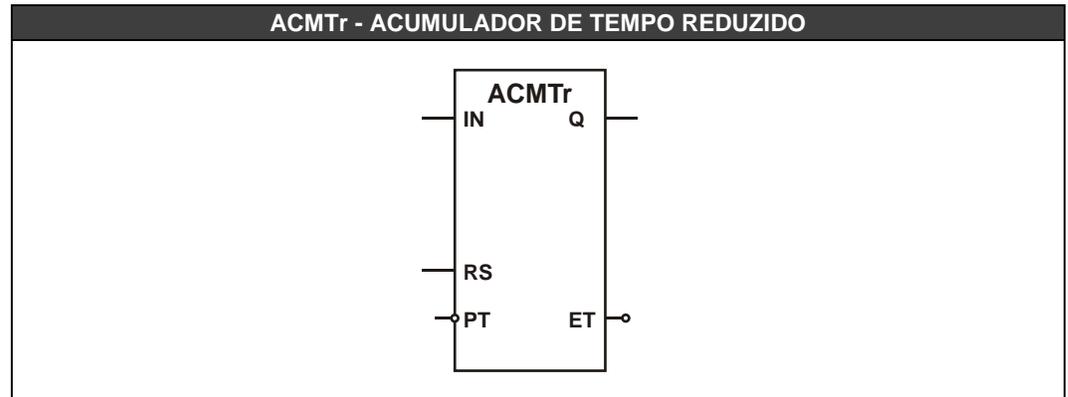
I: Entrada O: Saída P: Parâmetro Interno

Função do Acumulador de Tempo - Diagramas Temporais



Acumulador de Tempo Reduzido (ACMTr)

Apresenta funcionamento igual ao bloco ACMT, mas não possui a entrada **EN** e a saída **EO**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--|------|
| I | IN | ENTRADA DE PULSOS | BOOL |
| | RS | RESET DO BLOCO | BOOL |
| | PT | TEMPO PROGRAMADO (MILISSEGUNDOS) | LONG |
| O | Q | SAÍDA DO BLOCO | BOOL |
| | ET | TEMPO TRANSCORRIDO DA CONTAGEM (MILISSEGUNDOS) | LONG |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Acumulador de Tempo Reduzido (ACMTh)

Apresenta funcionamento igual ao bloco ACMTr, mas os tempos **PT** e **ET** são definidos em horas..



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|----------|------|---|------|
| I | IN | ENTRADA DE PULSOS | BOOL |
| | RS | RESET DO BLOCO | BOOL |
| | PT | TEMPO PROGRAMADO (EM HORAS) | LONG |
| O | Q | SAÍDA DO BLOCO | BOOL |
| | ET | TEMPO TRANSCORRIDO DA CONTAGEM (EM HORAS) | LONG |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Contador Decrescente de Pulsos (CDN)

Descrição

O bloco funcional CDN faz a contagem de transições do estado lógico 0 (falso) para estado lógico 1 (verdadeiro). Quando a entrada **EN** está verdadeira, é realizada a contagem de transições de falso para verdadeiro na entrada **IN**, decrementando o valor **CV**.

Quando **CV** chegar ao valor zero, a saída **Q** é mudada para verdadeiro, permanecendo neste valor até que a entrada **LD** vá para verdadeiro. Neste momento a saída **Q** retorna para falso e **CV** é carregado com o valor de **PV**.

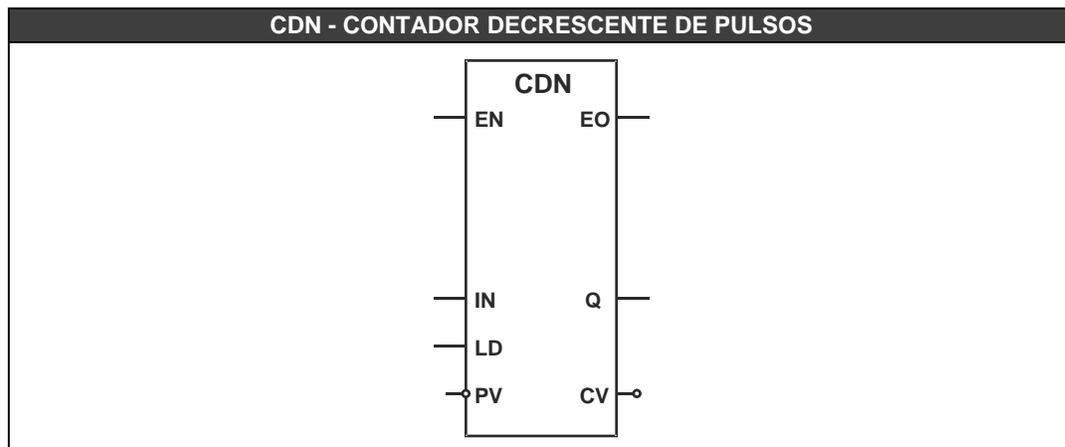
Se a entrada **EN** está falsa, todas as saídas booleanas são mantidas em zero e **CV** é carregado com o valor de **PV**.

Contador Interno CV

Na entrada deste bloco conecta-se uma entrada digital. Toda vez que ocorrer uma transição de subida, o **CV** decresce de uma unidade. Quando o contador interno chegar a zero, a saída **Q** irá para verdadeiro.

LD (Load)

Fazendo **LD** igual a verdadeiro a saída **Q** retorna para falso e **CV** é carregado com o valor de **PV**.

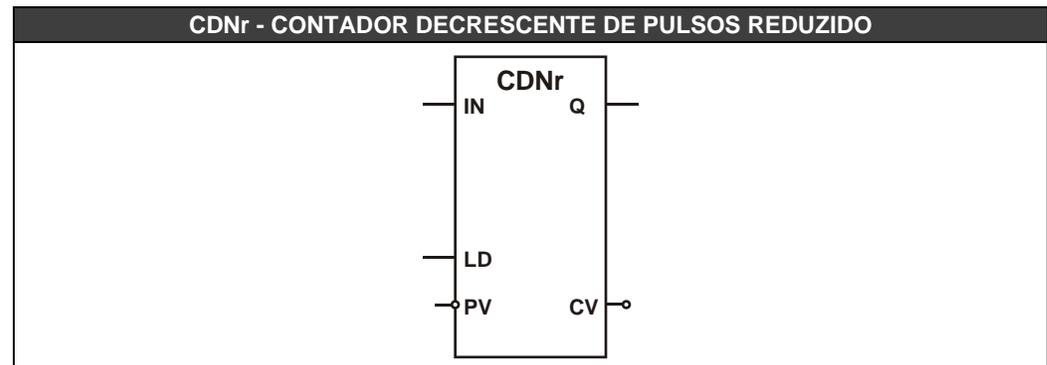


| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|----------|------|----------------------------|------|
| I | EN | HABILITAÇÃO DE ENTRADA | BOOL |
| | IN | ENTRADA DOS PULSOS | BOOL |
| | LD | LOAD | BOOL |
| | PV | VALOR PROGRAMADO | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | Q | SAÍDA DO BLOCO | BOOL |
| | CV | VALOR CORRENTE DA CONTAGEM | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Contador Decrescente de Pulsos Reduzido (CDNr)

Tem funcionamento igual ao bloco CDN, mas não possui a entrada **EN** e a saída **EO**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|----------------------------|------|
| I | IN | ENTRADA DOS PULSOS | BOOL |
| | LD | LOAD | BOOL |
| | PV | VALOR PROGRAMADO | LONG |
| O | Q | SÁIDA DO BLOCO | BOOL |
| | CV | VALOR CORRENTE DA CONTAGEM | LONG |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Contador Crescente/ Decrescente de Pulsos (CTUD)

Descrição

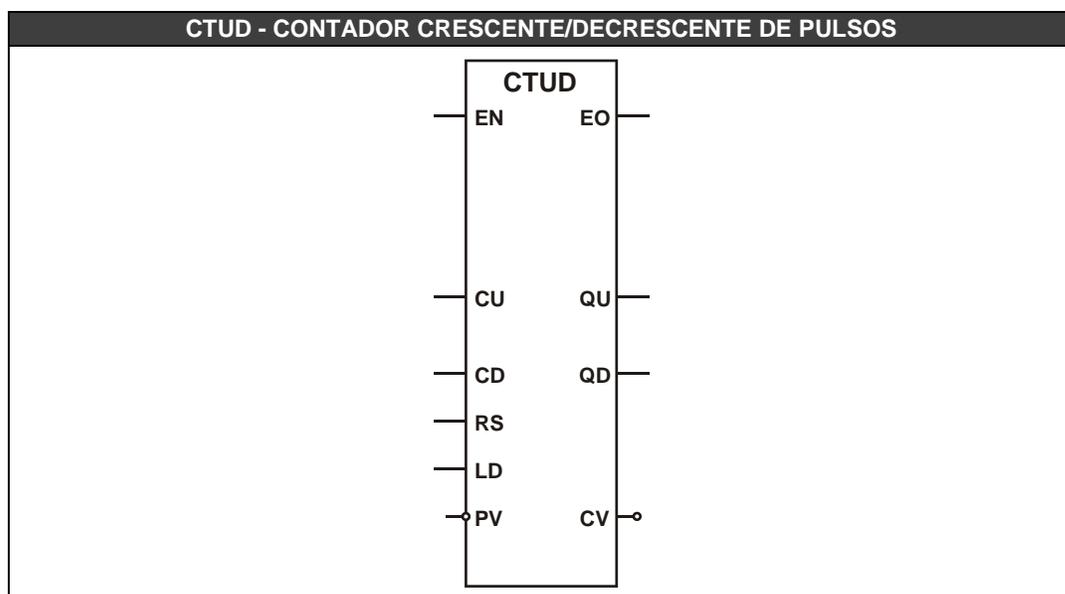
Este bloco funcional, quando a entrada **EN** está verdadeira, realiza a contagem de transições de falso para verdadeiro na entrada **CU**, incrementando o valor **CV**. Se ocorrerem transições de verdadeiro para falso na entrada **CD**, o valor **CV** será decrementado.

Se o valor em **CV** atinge o valor de **PV**, a saída **QU** é mantida em verdadeiro. Se o valor em **CV** atinge zero, a saída **QD** é mantida em verdadeiro.

A entrada **RS** indo para verdadeiro, mantém **QU** em falso, **QD** em verdadeiro e **CV** em zero.

A entrada **LD** indo para verdadeiro, mantém **QD** em falso, **QU** em verdadeiro e **CV** carregado com o valor de **PV**.

A entrada **LD** prevalece sobre a entrada **RS**. Enquanto **RS** ou **LD** estiver verdadeiro, a contagem é bloqueada. Se a entrada **EN** está falsa, todas as saídas booleanas são mantidas em zero e **CV** é carregado com o valor de **PV**.

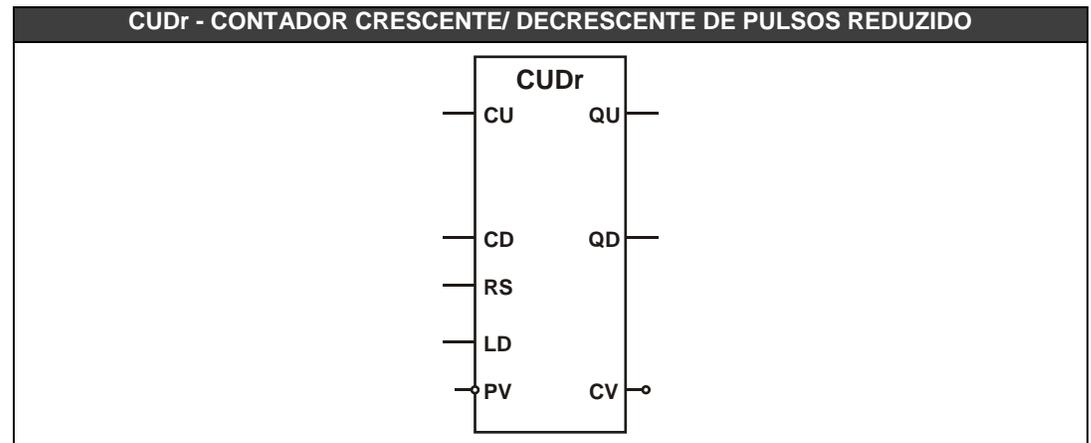


| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|----------|------|----------------------------|------|
| I | EN | HABILITAÇÃO DE ENTRADA | BOOL |
| | CU | ENTRADA DE PULSOS | BOOL |
| | CD | ENTRADA DE PULSOS | BOOL |
| | RS | RESET DO BLOCO | BOOL |
| | LD | LOAD | BOOL |
| | PV | VALOR PROGRAMADO | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | QU | SAÍDA DO BLOCO | BOOL |
| | QD | SAÍDA DO BLOCO | BOOL |
| | CV | VALOR CORRENTE DA CONTAGEM | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Contador Crescente/ Decrescente de Pulsos Reduzido (CUDr)

Tem funcionamento igual ao bloco CTUD, mas não possui a entrada EN e a saída EO.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|----------|------|----------------------------|------|
| I | CU | ENTRADA DE PULSOS | BOOL |
| | CD | ENTRADA DE PULSOS | BOOL |
| | RS | RESET DO BLOCO | BOOL |
| | LD | LOAD | BOOL |
| | PV | VALOR PROGRAMADO | LONG |
| O | QU | SAÍDA DO BLOCO | BOOL |
| | QD | SAÍDA DO BLOCO | BOOL |
| | CV | VALOR CORRENTE DA CONTAGEM | LONG |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Contador Crescente de Pulsos (CUP)

Descrição

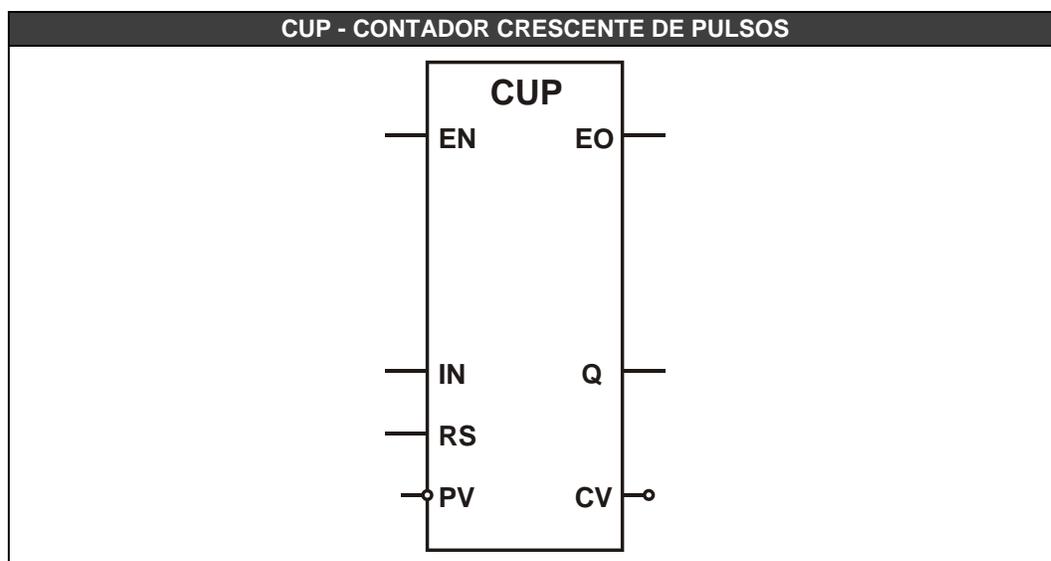
A função CUP realiza a contagem de transições do estado lógico 0 (falso) para estado lógico 1 (verdadeiro). Quando a entrada **EN** está verdadeira, é realizada a contagem de transições de falso para verdadeiro na entrada **IN**, incrementando o valor **CV**. Quando **CV** atinge o valor definido em **PV**, a saída **Q** é mudada para verdadeiro, permanecendo neste valor até que a entrada **RS** vá para verdadeiro, quando a saída **Q** retorna para falso. Se a entrada **EN** está falsa, todas as saídas são mantidas em zero.

Contador Interno CV

Toda vez que ocorrer uma transição de subida, o **CV** é incrementado de uma unidade. Esta operação é feita até que o contador interno chegue ao valor prefixado no parâmetro **PV**. Quando isso ocorrer a saída **Q** muda para verdadeiro, ou seja, de estado zero para estado 1.

RS (Reset)

Caso a entrada **RS** for verdadeira esta contagem será zerada. Enquanto **RS** estiver verdadeiro, a contagem é bloqueada.

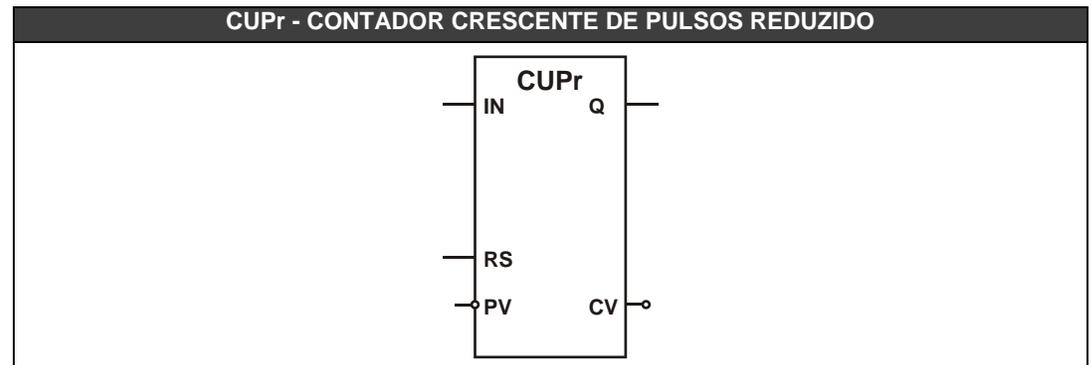


| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|----------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA DOS PULSOS | BOOL |
| | RS | RESET DO BLOCO | BOOL |
| | PV | VALOR PROGRAMADO | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | Q | SAÍDA DO BLOCO | BOOL |
| | CV | VALOR CORRENTE DA CONTAGEM | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Contador Crescente de Pulsos Reduzido (CUPr)

Tem funcionamento igual ao bloco CUP, mas não possui a entrada EN e a saída EO.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|----------------------------|------|
| I | IN | ENTRADA DOS PULSOS | BOOL |
| | RS | RESET DO BLOCO | BOOL |
| | PV | VALOR PROGRAMADO | LONG |
| O | Q | SAÍDA DO BLOCO | BOOL |
| | CV | VALOR CORRENTE DA CONTAGEM | LONG |

I: Entrada O: Saída P: Parâmetro Interno

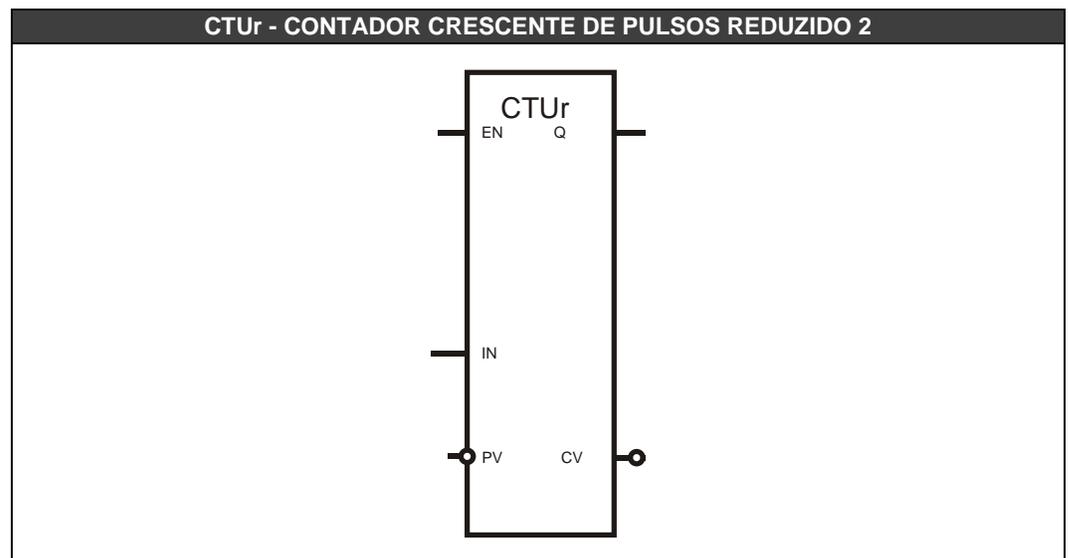
Contador Crescente de Pulsos Reduzido 2 (CTUr)

Descrição

A função CTUr realiza a contagem de transições do estado lógico 0 (falso) para estado lógico 1 (verdadeiro). Quando a entrada **EN** está verdadeira, é realizada a contagem de transições de falso para verdadeiro na entrada **IN**, incrementando o valor **CV**. Quando **CV** atinge o valor definido em **PV**, a saída **Q** é mudada para verdadeiro, permanecendo neste valor até que a entrada **EN** vá para falso, quando a saída **Q** retorna para falso. Se a entrada **EN** está falsa, todas as saídas são mantidas em zero.

Contador Interno CV

Toda vez que ocorrer uma transição de subida, o **CV** é incrementado de uma unidade. Esta operação é feita até que o contador interno chegue ao valor prefixado no parâmetro **PV**. Quando isso ocorrer a saída **Q** muda para verdadeiro, ou seja, de estado zero para estado 1.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|----------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA DOS PULSOS | BOOL |
| | PV | VALOR PROGRAMADO | LONG |
| O | Q | SAÍDA DO BLOCO | BOOL |
| | CV | VALOR CORRENTE DA CONTAGEM | LONG |

I: Entrada O: Saída P: Parâmetro Interno

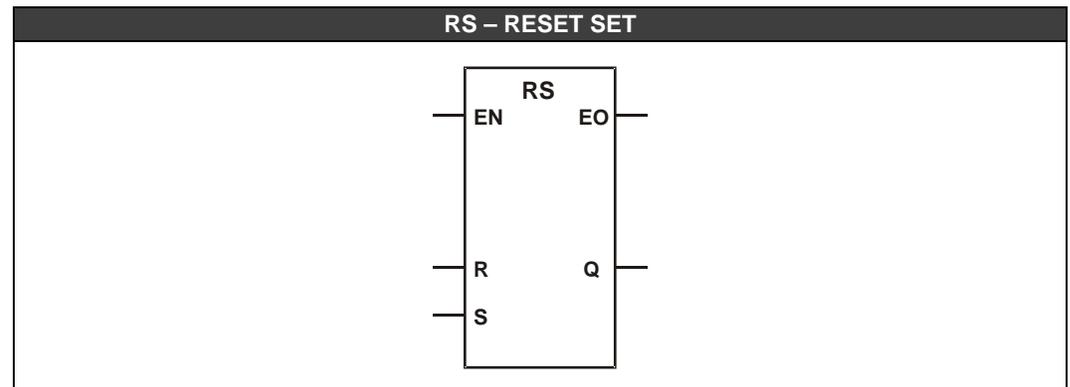
Reset Set (RS)

Descrição

Este bloco funcional, quando a entrada **EN** está verdadeira, funciona da seguinte forma:

Se a entrada **R** for verdadeira, a saída **Q** irá para falso. Se a entrada **S** for verdadeira, **Q** irá para verdadeiro. Se as duas entradas forem verdadeiras, **Q** se mantém em falso.

Se a entrada **EN** está falsa, todas as saídas são mantidas em zero.

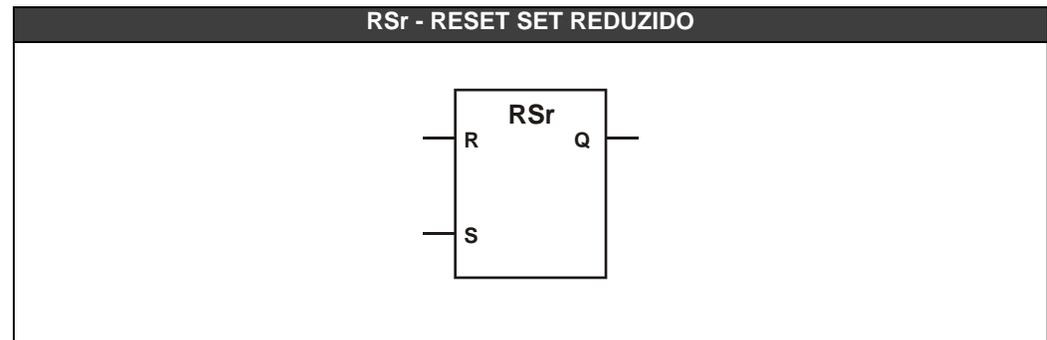


| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | R | RESET | BOOL |
| | S | SET | BOOL |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | Q | SAÍDA DO BLOCO | BOOL |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Reset Set Reduzido (RSr)

Apresenta funcionamento igual ao bloco RS, mas não possui a entrada **EN** e a saída **EO**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|----------------|------|
| I | R | RESET | BOOL |
| | S | SET | BOOL |
| O | Q | SAÍDA DO BLOCO | BOOL |

I: Entrada O: Saída P: Parâmetro Interno

Alarme em Tempo Real (RTA)

Descrição

Este bloco funcional, quando a entrada **EN** está verdadeira, funciona como um alarme de relógio. Uma data (**DT**) e um horário (**HR**) são especificados pelo usuário para que ocorra o disparo do alarme (**ALM**) nesta data e hora especificadas.

Quando o tempo especificado for atingido pela hora local, a qual está no fuso do Sistema Operacional Windows em que foi configurado o bloco, então a saída **ALM** irá para verdadeiro (estado lógico 1) caso estivesse em falso (estado lógico 0), o que configura o disparo do alarme.

Se for aplicado um sinal verdadeiro em **RS (RESET)**, o **ALM** retornará para falso. Se o sinal de **RESET** permanecer em verdadeiro, novos disparos do bloco estarão desabilitados.

Se a entrada **EN** está falsa, todas as saídas são mantidas em zero.

O bloco possui três parâmetros de configuração, que indicam a data, hora e a frequência de disparo do alarme. Os parâmetros são:

Parâmetro Data (DT)

O usuário, ao selecionar este parâmetro, escolherá uma data específica para o acionamento do alarme. O usuário deverá configurar a data desejada seguindo o formato: Ano/Mês/Dia, sendo que o ano deverá ter 4 dígitos. O ano escolhido deverá estar dentro do intervalo 2005 até 2037.

Parâmetro Periodicidade (PER)

Neste parâmetro o usuário poderá configurar a periodicidade de disparo do alarme. As opções são:

One-Shot: o bloco dispara apenas uma vez na data e hora determinadas;

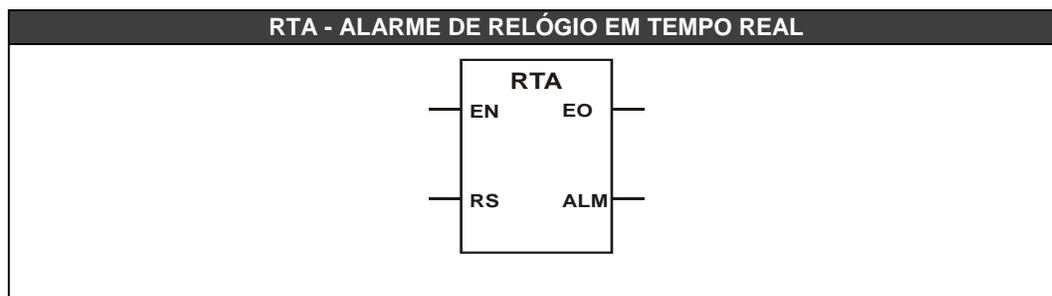
Daily: o bloco dispara todos os dias no mesmo horário especificado em **HR**;

Weekly: o bloco dispara toda semana no dia da semana do primeiro disparo;

Monthly: o bloco dispara todo mês no mesmo dia do mês do primeiro disparo.

Parâmetro Hora (HR)

O usuário deverá configurar a hora desejada para o acionamento do alarme. Esta hora deverá ser informada no formato **HR:MIN:SEC**, onde os parâmetros **HR**, **MIN** e **SEC** são respectivamente relacionados com hora, minuto e segundo.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|-------------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | RS | RESET DO BLOCO | BOOL |
| P | DT | DATA | DATE |
| | HR | HORA | hour |
| | PER | PERIODICIDADE | PERIODICITY |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | ALM | SAÍDA DO ALARME | BOOL |

I: Entrada **O:** Saída **P:** Parâmetro Interno

IMPORTANTE

1. O RTC (Real Time Clock) do controlador onde será executado o RTA deve estar de acordo com o horário oficial local.
2. O RTC do controlador do DF1302 pode ser configurado manualmente, via opção Batch Download do FBTools, e quando disponível, mantido sincronizado automaticamente via SNTP. Para mais informações veja o help do FBTools e o apêndice Server Manager no manual do Studio302, respectivamente.
3. Deve-se tomar cuidado com as mudanças no início e final do horário de verão. O importante é que, mudando o horário, para mais ou para menos, deve-se fazer a mesma mudança no controlador.

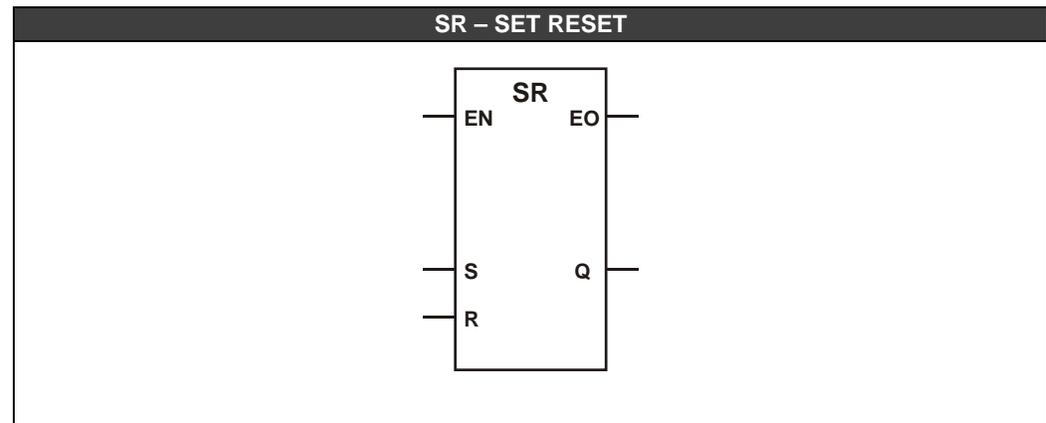
Set Reset (SR)

Descrição

Este bloco funcional, quando a entrada **EN** está verdadeira, funciona da seguinte forma:

Se a entrada **S** for verdadeira, a saída **Q** irá para verdadeiro. Se a entrada **R** for verdadeira, **Q** irá para falso. Se as duas entradas forem verdadeiras, **Q** se mantém em verdadeiro.

Se a entrada **EN** está falsa, todas as saídas são mantidas em zero.

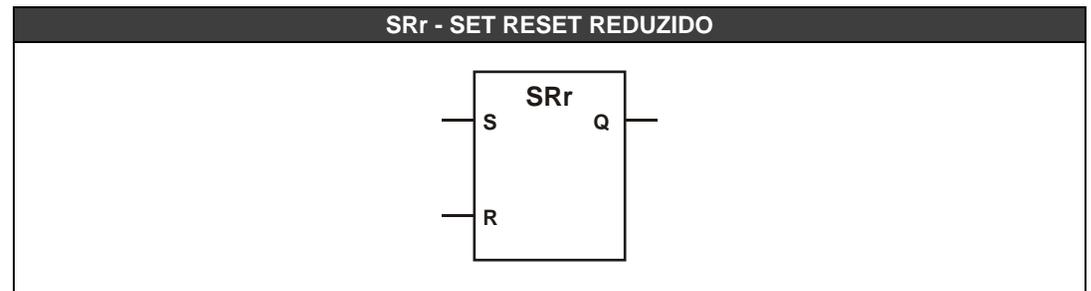


| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|----------|------|------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | R | RESET | BOOL |
| | S | SET | BOOL |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | Q | SAÍDA DO BLOCO | BOOL |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Set Reset Reduzido (SRr)

Apresenta funcionamento igual ao bloco **SR**, mas não possui a entrada **EN** e a saída **EO**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|----------------|------|
| I | R | RESET | BOOL |
| | S | SET | BOOL |
| O | Q | SAÍDA DO BLOCO | BOOL |

I: Entrada O: Saída P: Parâmetro Interno

Tempo de Atraso para Desligar (TOF)

Descrição

Quando a entrada **EN** está verdadeira, esta função mantém o estado verdadeiro da entrada **IN** na saída **Q** por um período de tempo, previamente definido, após a entrada **IN** mudar para falso. Este tempo é definido em **PT** e é dado em milissegundos.

Se **IN** muda para verdadeiro, antes de **Q** ir para falso, **Q** permanecerá no estado verdadeiro e o período de tempo iniciará novamente no momento em que **IN** for para falso.

Se a entrada **EN** está falsa, todas as saídas são mantidas em zero.

Entrada PT

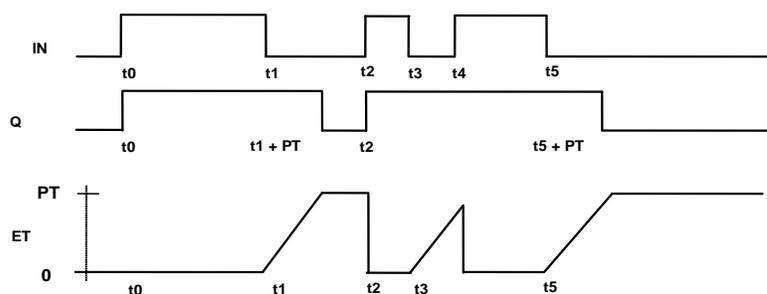
A entrada **PT** pode ser conectada à saída de um bloco funcional, um FFB ou um valor fixo.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|----------|------|--------------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA DE PULSOS | BOOL |
| | PT | TEMPO PROGRAMADO | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | Q | SAÍDA DO BLOCO | BOOL |
| | ET | TEMPO TRANSCORRIDO DA CONTAGEM | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Função Atraso no Tempo para Desligar - Diagramas temporais



Tempo de Atraso para Desligar Reduzido (TOFr)

Apresenta funcionamento igual ao bloco TOF, mas não possui a entrada **EN** e a saída **EO**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--------------------------------|------|
| I | IN | ENTRADA DE PULSOS | BOOL |
| | PT | TEMPO PROGRAMADO | LONG |
| O | Q | SAÍDA DO BLOCO | BOOL |
| | ET | TEMPO TRANSCORRIDO DA CONTAGEM | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Tempo de Atraso para Ligar (TON)

Descrição

Esta função, quando a entrada **EN** está verdadeira, causa um atraso na transição de falso para verdadeiro na saída **Q** por um tempo previamente definido, após a entrada **IN** ter mudado para verdadeiro. Este tempo é definido em **PT** e é dado em milissegundos.

Se a entrada **IN** mudar para falso, antes que **Q** mude para verdadeiro, **Q** permanecerá no estado falso e o período de tempo começará novamente no momento em que **IN** for para verdadeiro.

Se a entrada **EN** está falsa, todas as saídas são mantidas em zero.

Entrada PT

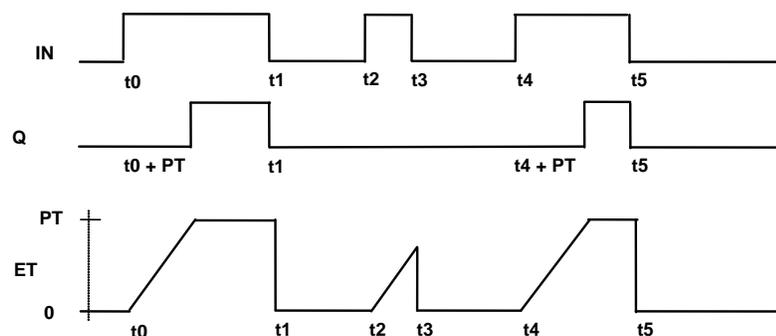
A entrada **PT** pode ser conectada à saída de um bloco funcional, um FFB ou um valor fixo.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--------------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA DE PULSOS | BOOL |
| | PT | TEMPO PROGRAMADO | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | Q | SAÍDA DO BLOCO | BOOL |
| | ET | TEMPO TRANSCORRIDO DA CONTAGEM | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Função Atraso no Tempo para Ligar – Diagramas Temporais



Tempo de Atraso para Ligar Reduzido (TONr)

Apresenta funcionamento igual ao bloco TON, mas não possui a entrada **EN** e a saída **EO**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--------------------------------|------|
| I | IN | ENTRADA DE PULSOS | BOOL |
| | PT | TEMPO PROGRAMADO | LONG |
| O | Q | SAÍDA DO BLOCO | BOOL |
| | ET | TEMPO TRANSCORRIDO DA CONTAGEM | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Temporizador de Pulso (TP)

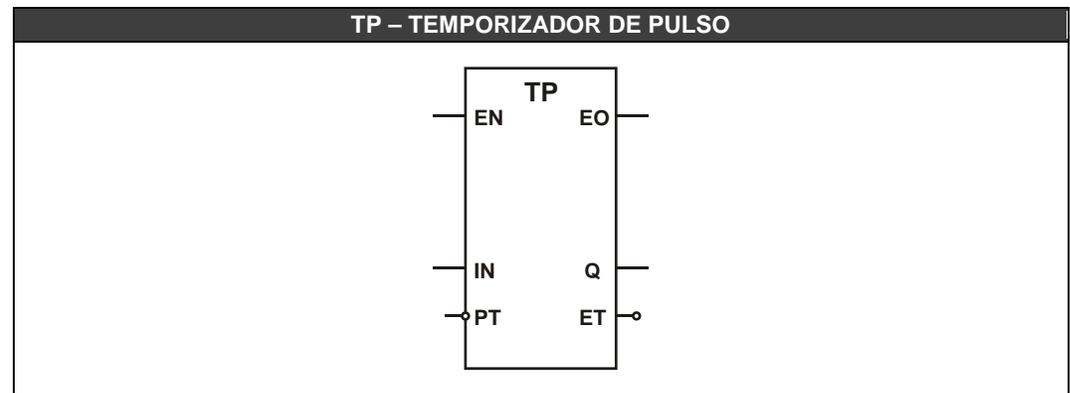
Descrição

Esta função, quando a entrada **EN** está verdadeira, gera um pulso de tempo com duração fixa na saída **Q** para cada transição de falso para verdadeiro na entrada **IN**. Esta duração é definida em **PT** e é dada em milissegundos.

Transições na entrada **IN** serão ignoradas enquanto a saída estiver ativa. Se a entrada **EN** está falsa, todas as saídas são mantidas em zero.

Entrada PT

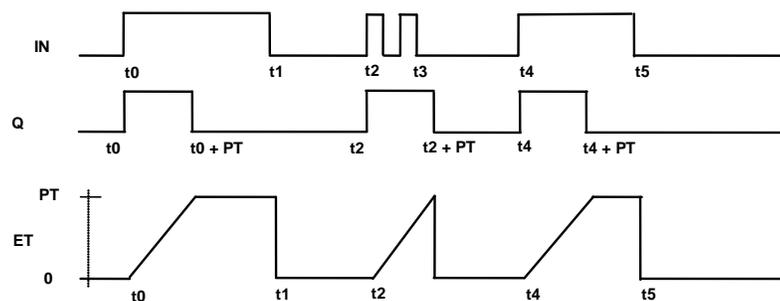
A entrada **PT** pode ser conectada à saída de um bloco funcional, um FFB ou um valor fixo.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--------------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA DE PULSOS | BOOL |
| | PT | TEMPO PROGRAMADO | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | Q | SAÍDA DO BLOCO | BOOL |
| | ET | TEMPO TRANSCORRIDO DA CONTAGEM | LONG |

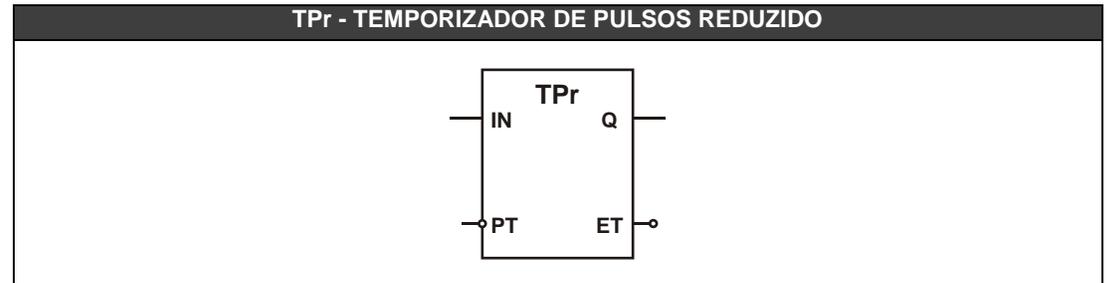
I: Entrada O: Saída P: Parâmetro Interno

Função do Temporizador de Pulso - Diagramas Temporais



Temporizador de Pulsos Reduzido (TPr)

Apresenta funcionamento igual ao bloco TP, mas não possui a entrada **EN** e a saída **EO**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--------------------------------|------|
| I | IN | ENTRADA DE PULSOS | BOOL |
| | PT | TEMPO PROGRAMADO | LONG |
| O | Q | SAÍDA DO BLOCO | BOOL |
| | ET | TEMPO TRANSCORRIDO DA CONTAGEM | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Funções de Manipulação de Dados

Conversão de byte para inteiro (BINT)

Descrição

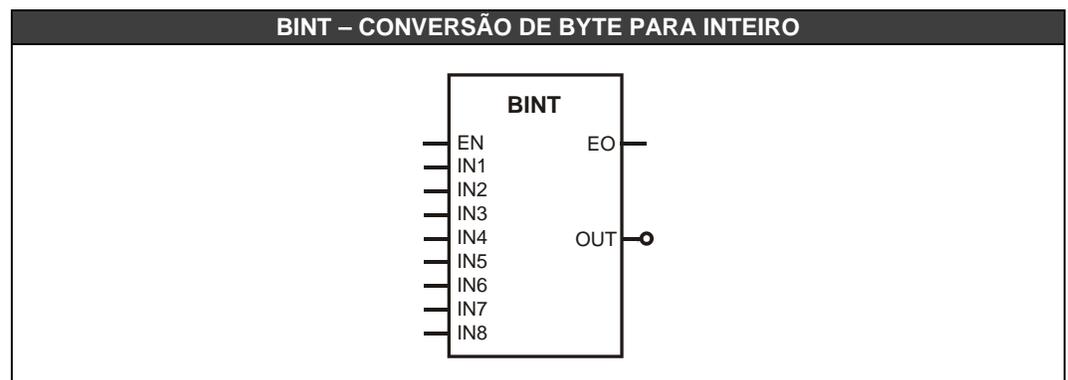
Esta função, quando **EN** é verdadeira, converte um byte formado pelas 8 entradas booleanas (**IN8-IN7-IN6-IN5-IN4-IN3-IN2-IN1**) para um número inteiro e coloca-o na saída **OUT**.

Conversão

O byte formado pelas entradas é convertido para um número inteiro. Por exemplo: tem-se as seguintes entradas:

IN1 = 1
 IN2 = 1
 IN3 = 1
 IN4 = 0
 IN5 = 0
 IN6 = 0
 IN7 = 1
 IN8 = 1

Ou seja, a entrada é 11000111 (em binário) ou C7 (em hexadecimal). A saída do bloco terá o valor 199.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 (BIT MENOS SIGNIFICATIVO) | BOOL |
| | IN2 | ENTRADA 2 | BOOL |
| | IN3 | ENTRADA 3 | BOOL |
| | IN4 | ENTRADA 4 | BOOL |
| | IN5 | ENTRADA 5 | BOOL |
| | IN6 | ENTRADA 6 | BOOL |
| | IN7 | ENTRADA 7 | BOOL |
| O | IN8 | ENTRADA 8 (BIT MAIS SIGNIFICATIVO) | BOOL |
| | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA (VALOR DAS ENTRADAS CONVERTIDO PARA INTEIRO) | LONG |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Conversão de Byte para Bits (BTB)

Descrição

A função BTB, quando **EN** é verdadeira, converte o primeiro byte de um dado tipo LONG em 8 saídas paralelas, cada uma delas representando um bit.

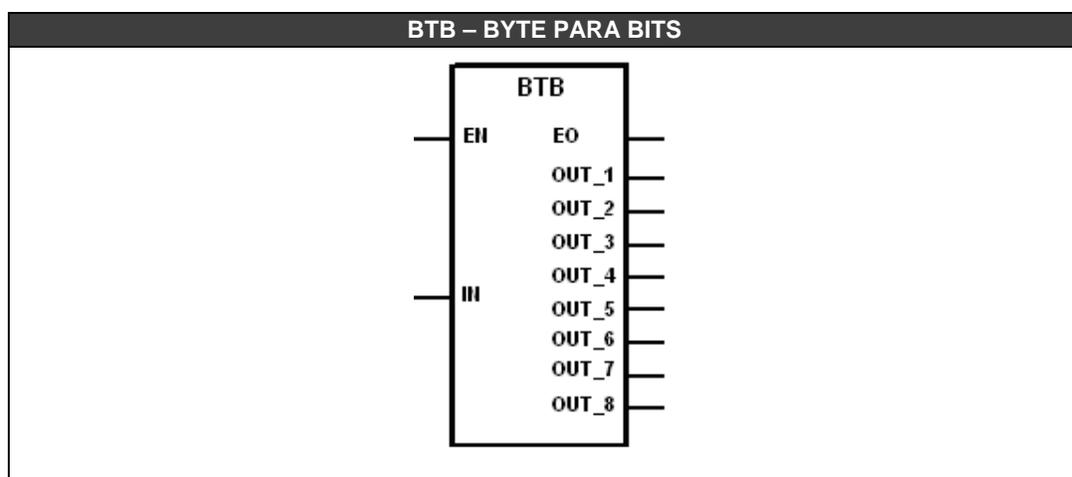
Se **EN** for falso, as saídas são zeradas.

Conversão

A entrada do bloco é um dado tipo LONG, em que, para efeito de conversão, é considerado apenas o byte menos significativo que é decomposto nas saídas **OUT_1** até **OUT_8**.

O dado de entrada pode ser proveniente da saída de um outro bloco funcional, por exemplo, a saída BROUT do bloco TEMP (DF45 – módulo de temperatura). Nesse caso, as saídas **OUT_1** até **OUT_8** representarão o status de *burnout* de cada uma das entradas do módulo de temperatura.

As saídas **OUT_1** até **OUT_8** podem ser utilizadas como entradas de outros blocos funcionais, como, por exemplo, o bloco BWL1 ou o bloco NOT1.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA DO BLOCO | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT_1 | BIT 0 (LSB) | BOOL |
| | OUT_2 | BIT 1 | BOOL |
| | OUT_3 | BIT 2 | BOOL |
| | OUT_4 | BIT 3 | BOOL |
| | OUT_5 | BIT 4 | BOOL |
| | OUT_6 | BIT 5 | BOOL |
| | OUT_7 | BIT 6 | BOOL |
| | OUT_8 | BIT 7 (MSB) | BOOL |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Conversão de Entrada Booleana para Inteiro (BTI1)

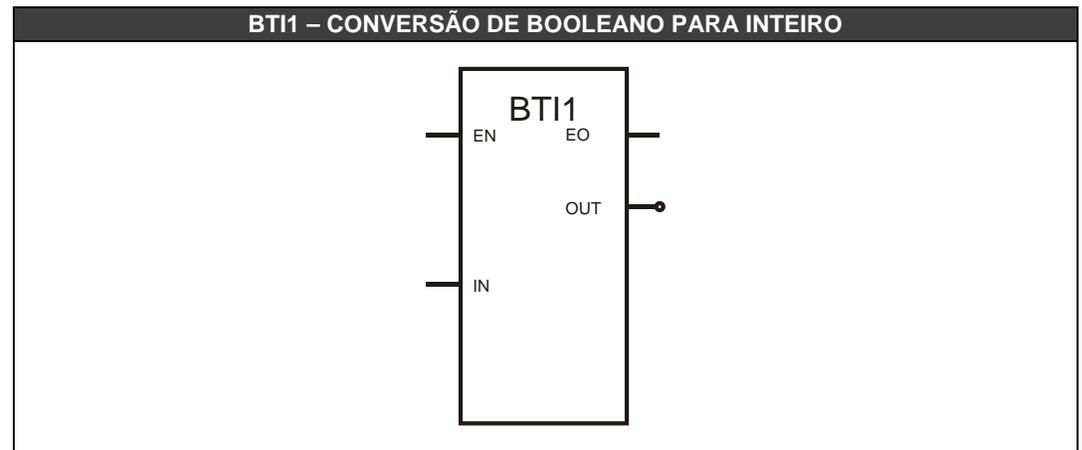
Descrição

Esta função, quando **EN** é verdadeira, converte o estado da entrada booleana **IN** para número inteiro e coloca-o na saída **OUT**.

Conversão

Se o estado lógico de **IN** for falso, a saída **OUT** terá o valor "0".

Se o estado lógico de **IN** for verdadeiro, a saída **OUT** terá o valor "1".



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|---|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA | BOOL |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA (ESTADO DA ENTRADA CONVERTIDO PARA INTEIRO) | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Conversão de BCD para Inteiro (BTI2)

Descrição

Esta função, quando **EN** é verdadeira, converte um valor de BCD na entrada **IN** para número inteiro e coloca-o na saída **OUT**.

Conversão

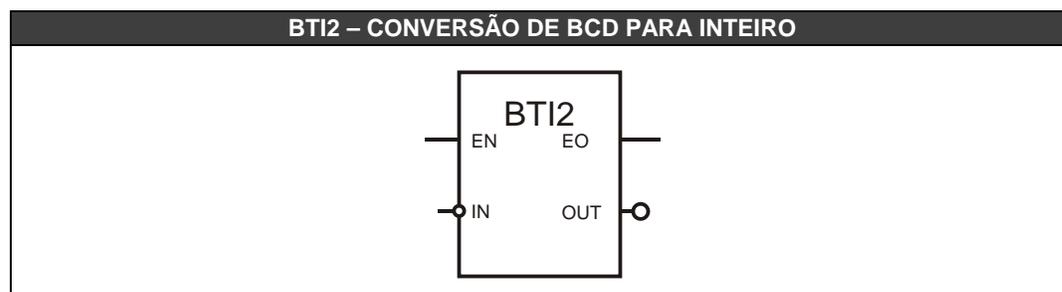
Um número de dois dígitos em BCD está na forma:

BIT7-BIT6-BIT5-BIT4 _____ BIT3-BIT2-BIT1-BIT0

Cada conjunto de quatro bits compõe um dígito. Por exemplo: o número 10. Em código BCD, a representação é 10. O primeiro dígito da esquerda para a direita pode ser escrito na forma binária como 0001. O segundo dígito pode ser escrito na forma binária como 0000. Assim 10BCD = 0001 0000 Binário.

Costuma-se confundir o código BCD com a numeração binária. Porém, cada grupo de quatro bits somente representa um dígito, que varia de 0 até 9. Por exemplo, não pode haver representação em código BCD do tipo 12 9BCD, mesmo que o número 12 possa ser representado por 4 bits.

O código BCD é bastante utilizado em displays de sete segmentos. Cada display representa um dígito BCD. A representação mostrada acima pode ser estendida para N dígitos, sempre notando que cada dígito varia apenas de 0 até 9.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA (VALOR DA ENTRADA CONVERTIDO PARA INTEIRO) | LONG |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Operação Lógica Binária (BWL1)

Descrição

Permite que funções lógicas sejam implementadas utilizando um bloco funcional. Seis funções lógicas diferentes podem ser configuradas: **AND**, **NAND**, **OR**, **NOR**, **XOR** e **NXOR**. O usuário escolhe o tipo de operação lógica durante a configuração e o bloco BWL1 passa a ter a função especificada neste procedimento. O número de entradas do bloco é configurado pelo parâmetro **N_IN**, sendo o mínimo de 2 e o máximo de 14. O bloco realiza as operações entre os bits representados por cada entrada digital.

Prm = "0": Função **AND**

A função lógica **AND** para duas entradas **IN1** e **IN2** tem uma saída **OUT** dada pela equação booleana: $OUT=IN1 \cdot IN2$. Transpondo esta equação lógica para a tabela de estados:

| IN1 | IN2 | OUT |
|-----|-----|-----|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Prm = "1": Função **OR**

A função lógica **OR** para duas entradas **IN1** e **IN2** tem uma saída **OUT** dada pela equação booleana: $OUT=IN1+IN2$. Transpondo esta equação lógica para a tabela de estados:

| IN1 | IN2 | OUT |
|-----|-----|-----|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Prm = "2": Função **XOR**

A função lógica **XOR** para duas entradas **IN1** e **IN2** tem uma saída **OUT** dada pela equação booleana:

$$OUT = \overline{IN1} \cdot IN2 + IN1 \cdot \overline{IN2}$$

Transpondo esta equação lógica para a tabela de estados:

| IN1 | IN2 | OUT |
|-----|-----|-----|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Prm = "3": Função **NAND**

Esta função é associação da função **AND** com a função **NOT** inversora. Logo, a saída lógica é a função **AND invertida**.

Prm = "4": Função **NOR**

Esta função é associação da função **OR** com a função **NOT** inversora. Logo, a saída lógica é a função **OR invertida**.

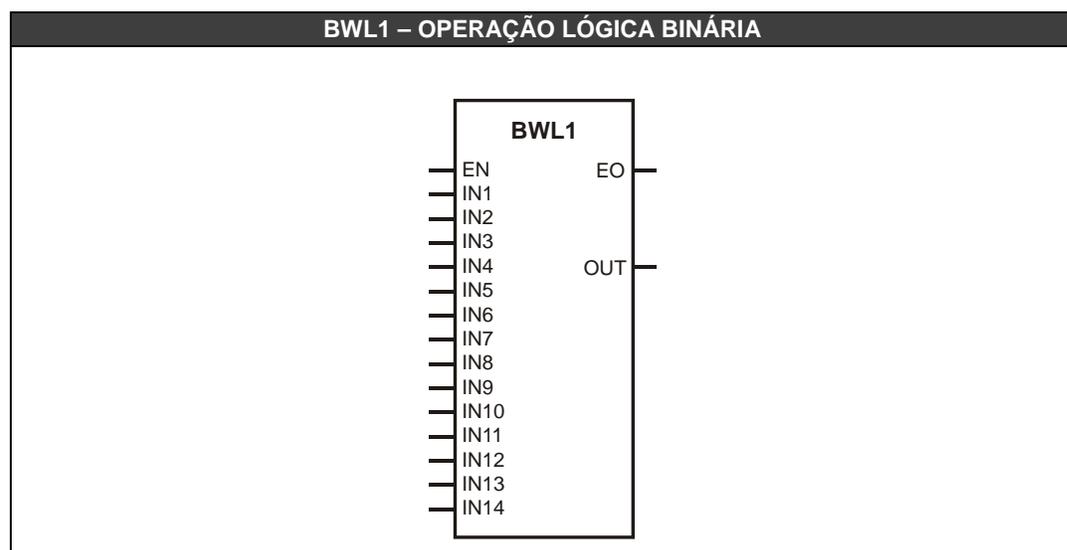
Prm = "5": Função **NXOR**

Esta função é associação da função **XOR** com a função **NOT** inversora. Logo, a saída lógica é a função **NXOR invertida**.

O bloco BWL permite que esta lógica seja expansível para 14 entradas. Na tabela abaixo mostramos as funções lógicas para mais de duas entradas e suas respectivas saídas.

| ENTRADAS | | | | | SAIDAS | | | | | |
|----------|-----|-----|-------|-----|--------|------|----|-----|-----|------|
| IN1 | IN2 | ... | INn-1 | INn | AND | NAND | OR | NOR | XOR | NXOR |
| 0 | 0 | | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 0 | | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |
| 0 | 0 | | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| 0 | 0 | | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| | | ... | | | | | 1 | 0 | | |
| 1 | 1 | | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |

Se a entrada **EN** está falsa, a saída será mantida em zero. Se o valor Prm for maior do que 5, as saídas **EO** e **OUT** ficarão em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | BOOL |
| | IN2 | ENTRADA 2 | BOOL |
| | IN3 | ENTRADA 3 | BOOL |
| | IN4 | ENTRADA 4 | BOOL |
| | IN5 | ENTRADA 5 | BOOL |
| | IN6 | ENTRADA 6 | BOOL |
| | IN7 | ENTRADA 7 | BOOL |
| | IN8 | ENTRADA 8 | BOOL |
| | IN9 | ENTRADA 9 | BOOL |
| | IN10 | ENTRADA 10 | BOOL |
| | IN11 | ENTRADA 11 | BOOL |
| | IN12 | ENTRADA 12 | BOOL |
| | IN13 | ENTRADA 13 | BOOL |
| IN14 | ENTRADA 14 | BOOL | |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | BOOL |
| P | Prm | OPERAÇÃO LÓGICA | LONG |
| | N_IN | NÚMERO DE ENTRADAS | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Operação Lógica Binária Reduzida (BWL1r)

Descrição

Permite que funções lógicas sejam implementadas utilizando um bloco funcional. Seis funções lógicas diferentes podem ser configuradas: **AND**, **NAND**, **OR**, **NOR**, **XOR** e **NXOR**. O usuário escolhe o tipo de operação lógica durante a configuração e o bloco BWL1r passa a ter a função especificada neste procedimento. O bloco realiza as operações entre os bits representados por cada uma das duas entradas digitais.

Prm = "0": Função **AND**

A função lógica **AND** para as entradas **IN1** e **IN2** tem uma saída **OUT** dada pela equação booleana: $OUT=IN1 \cdot IN2$. Transpondo esta equação lógica para a tabela de estados:

| IN1 | IN2 | OUT |
|-----|-----|-----|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Prm = "1": Função **OR**

A função lógica **OR** para as entradas **IN1** e **IN2** tem uma saída **OUT** dada pela equação booleana: $OUT=IN1+IN2$. Transpondo esta equação lógica para a tabela de estados:

| IN1 | IN2 | OUT |
|-----|-----|-----|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Prm = "2": Função **XOR**

A função lógica **XOR** para as entradas **IN1** e **IN2** tem uma saída **OUT** dada pela equação booleana:

$$OUT = IN1\overline{IN2} + \overline{IN1}IN2$$

Transpondo esta equação lógica para a tabela de estados:

| IN1 | IN2 | OUT |
|-----|-----|-----|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Prm = "3": Função **NAND**

Esta função é associação da função **AND** com a função **NOT** inversora. Logo, a saída lógica é a função **AND invertida**.

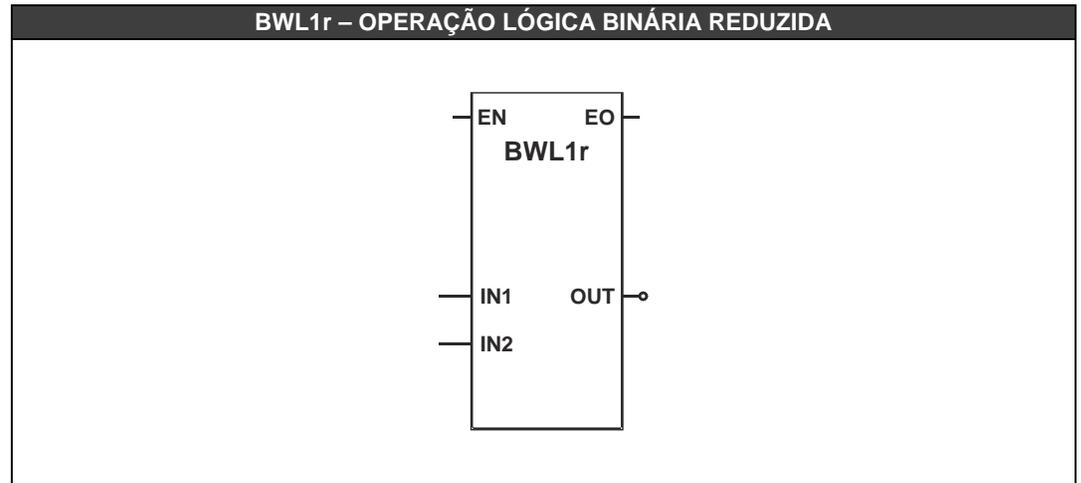
Prm = "4": Função **NOR**

Esta função é associação da função **OR** com a função **NOT** inversora. Logo, a saída lógica é a função **OR invertida**.

Prm = "5": Função **NXOR**

Esta função é associação da função **XOR** com a função **NOT** inversora. Logo, a saída lógica é a função **NXOR invertida**.

Se a entrada **EN** está falsa, a saída será mantida em zero. Se o valor Prm for maior do que 5, as saídas **EO** e **OUT** ficarão em 0.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | BOOL |
| | IN2 | ENTRADA 2 | BOOL |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | BOOL |
| P | Prm | OPERAÇÃO LÓGICA | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Operação Lógica Bit a Bit (BWL2)

Descrição

Permite que funções lógicas sejam implementadas utilizando um bloco funcional. Seis funções lógicas diferentes podem ser configuradas: **AND**, **NAND**, **OR**, **NOR**, **XOR** e **NXOR**. O usuário escolhe o tipo de operação lógica durante a configuração e o bloco BWL2 passa a ter a função especificada neste procedimento. O número de entradas do bloco é configurado pelo parâmetro **N_IN**, sendo o mínimo de 2 e o máximo de 14. O bloco realiza as operações lógicas bit a bit entre as entradas.

Prm = "0": Função **AND**

A função lógica **AND** para duas entradas **IN1** e **IN2** tem uma saída **OUT** dada pela equação booleana: $OUT=IN1.IN2$. Transpondo esta equação lógica para a tabela de estados:

```
IN1= (BIT17)(BIT16)(BIT15)(BIT14)(BIT13)(BIT12)(BIT11)(BIT10)
IN2= (BIT27)(BIT26)(BIT25)(BIT24)(BIT23)(BIT22)(BIT21)(BIT20)
OUT= (BIT17ANDBIT27).....(BIT10ANDBIT20)
Exemplo: IN1= 00001111
IN2= 11110000
OUT= 00000000
```

Prm = "1": Função **OR**

A função lógica **OR** para duas entradas **IN1** e **IN2** tem uma saída **OUT** dada pela equação booleana: $OUT=IN1+IN2$. Transpondo esta equação lógica para a tabela de estados:

```
IN1= (BIT17)(BIT16)(BIT15)(BIT14)(BIT13)(BIT12)(BIT11)(BIT10)
IN2= (BIT27)(BIT26)(BIT25)(BIT24)(BIT23)(BIT22)(BIT21)(BIT20)
OUT= (BIT17ORBIT27).....(BIT10ORBIT20)
Exemplo: IN1= 00001111
IN2= 11110000
OUT= 11111111
```

Prm = "2": Função **XOR**

A função lógica **XOR** para duas entradas **IN1** e **IN2** tem uma saída **OUT** dada pela equação booleana:

$$OUT=IN1.\overline{IN2} + \overline{IN1}.IN2$$

Transpondo esta equação lógica para a tabela de estados:

```
IN1= (BIT17)(BIT16)(BIT15)(BIT14)(BIT13)(BIT12)(BIT11)(BIT10)
IN2= (BIT27)(BIT26)(BIT25)(BIT24)(BIT23)(BIT22)(BIT21)(BIT20)
OUT= (BIT17XORBIT27).....(BIT10XORBIT20)

Exemplo: IN1= 01011100
          IN2= 11110000
          OUT= 10101100
```

Prm = "3": Função **NAND**

Esta função é associação da função **AND** com a função **NOT** inversora. Logo, a saída lógica é a função **AND invertida**.

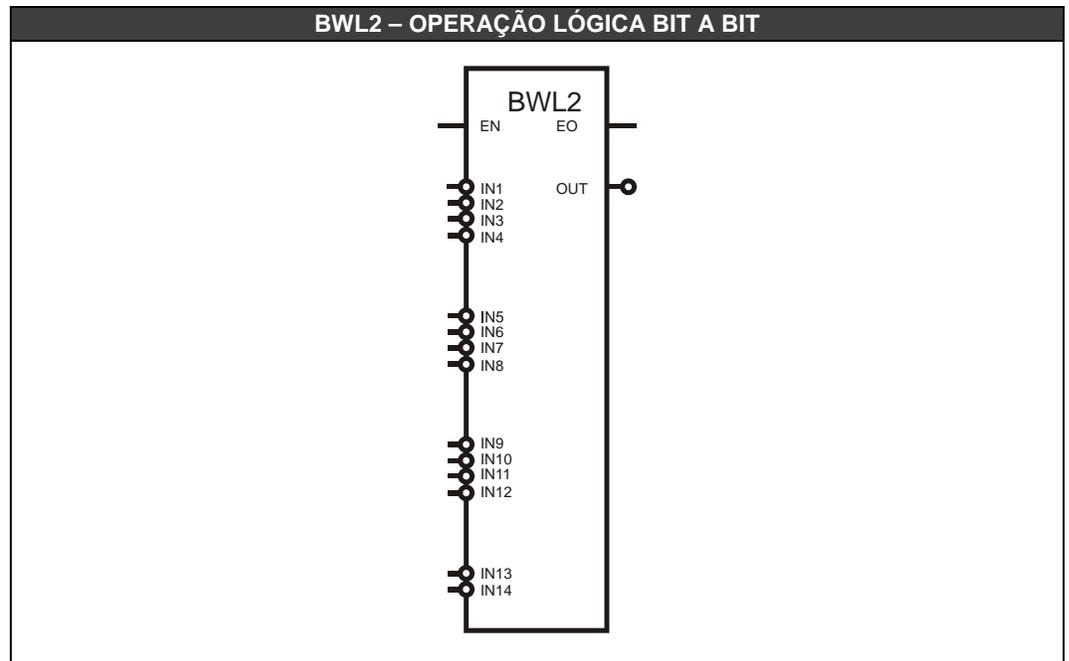
Prm = "4": Função **NOR**

Esta função é associação da função **OR** com a função **NOT** inversora. Logo, a saída lógica é a função **OR invertida**.

Prm = "5": Função **NXOR**

Esta função é associação da função **XOR** com a função **NOT** inversora. Logo, a saída lógica é a função **NXOR invertida**.

Se a entrada **EN** está falsa, a saída será mantida em zero. Se o valor Prm for maior do que 5, as saídas **EO** e **OUT** ficarão em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | LONG |
| | IN2 | ENTRADA 2 | LONG |
| | IN3 | ENTRADA 3 | LONG |
| | IN4 | ENTRADA 4 | LONG |
| | IN5 | ENTRADA 5 | LONG |
| | IN6 | ENTRADA 6 | LONG |
| | IN7 | ENTRADA 7 | LONG |
| | IN8 | ENTRADA 8 | LONG |
| | IN9 | ENTRADA 9 | LONG |
| | IN10 | ENTRADA 10 | LONG |
| | IN11 | ENTRADA 11 | LONG |
| | IN12 | ENTRADA 12 | LONG |
| | IN13 | ENTRADA 13 | LONG |
| IN14 | ENTRADA 14 | LONG | |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | LONG |
| P | Prm | OPERAÇÃO LÓGICA | LONG |
| | N_IN | NÚMERO DE ENTRADAS | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Operação Lógica Bit a Bit Reduzida (BWL2r)

Descrição

Permite que funções lógicas sejam implementadas utilizando um bloco funcional. Seis funções lógicas diferentes podem ser configuradas: **AND**, **NAND**, **OR**, **NOR**, **XOR** e **NXOR**. O usuário escolhe o tipo de operação lógica durante a configuração e o bloco BWL2r passa a ter a função especificada neste procedimento. O bloco realiza as operações lógicas bit a bit entre as duas entradas.

Prm = "0": Função **AND**

A função lógica **AND** para as entradas **IN1** e **IN2** tem uma saída **OUT** dada pela equação booleana: $OUT=IN1.IN2$. Transpondo esta equação lógica para a tabela de estados:

```
IN1= (BIT17)(BIT16)(BIT15)(BIT14)(BIT13)(BIT12)(BIT11)(BIT10)
IN2= (BIT27)(BIT26)(BIT25)(BIT24)(BIT23)(BIT22)(BIT21)(BIT20)
OUT= (BIT17ANDBIT27).....(BIT10ANDBIT20)
Exemplo: IN1= 00001111
IN2= 11110000
OUT= 00000000
```

Prm = "1": Função **OR**

A função lógica **OR** para as entradas **IN1** e **IN2** tem uma saída **OUT** dada pela equação booleana: $OUT=IN1+IN2$. Transpondo esta equação lógica para a tabela de estados:

```
IN1= (BIT17)(BIT16)(BIT15)(BIT14)(BIT13)(BIT12)(BIT11)(BIT10)
IN2= (BIT27)(BIT26)(BIT25)(BIT24)(BIT23)(BIT22)(BIT21)(BIT20)
OUT= (BIT17ORBIT27).....(BIT10ORBIT20)
Exemplo: IN1= 00001111
IN2= 11110000
OUT= 11111111
```

Prm = "2": Função **XOR**

A função lógica **XOR** para as entradas **IN1** e **IN2** tem uma saída **OUT** dada pela equação booleana: $OUT=IN1.IN2 + \overline{IN1}.IN2$

Transpondo esta equação lógica para a tabela de estados:

```
IN1= (BIT17)(BIT16)(BIT15)(BIT14)(BIT13)(BIT12)(BIT11)(BIT10)
IN2= (BIT27)(BIT26)(BIT25)(BIT24)(BIT23)(BIT22)(BIT21)(BIT20)
OUT= (BIT17XORBIT27).....(BIT10XORBIT20)

Exemplo: IN1= 01011100
IN2= 11110000
OUT= 10101100
```

Prm = "3": Função **NAND**

Esta função é associação da função **AND** com a função **NOT** inversora. Logo, a saída lógica é a função **AND invertida**.

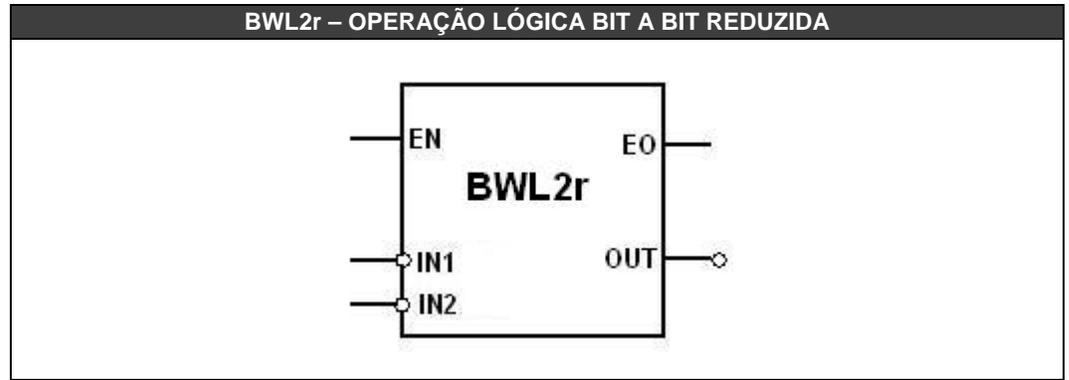
Prm = "4": Função **NOR**

Esta função é associação da função **OR** com a função **NOT** inversora. Logo, a saída lógica é a função **OR invertida**.

Prm = "5": Função **NXOR**

Esta função é associação da função **XOR** com a função **NOT** inversora. Logo, a saída lógica é a função **NXOR invertida**.

Se a entrada **EN** está falsa, a saída será mantida em zero. Se o valor Prm for maior do que 5, as saídas **EO** e **OUT** ficarão em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | LONG |
| | IN2 | ENTRADA 2 | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | LONG |
| P | Prm | OPERAÇÃO LÓGICA | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Constantes (CONST)

Descrição

Esta função, quando **EN** é verdadeira, envia valores de constantes para as saídas **OUT1**, **OUT2** e **OUT3**. Estas constantes são informadas na configuração do bloco no LogicView for FFB. Deve-se notar que estas constantes somente serão enviadas às saídas do bloco quando a entrada **EN** for verdadeira.

Se **EN** for falso, as saídas são zeradas.

Parâmetros P1, P2 e P3

Nestes parâmetros o usuário deverá inserir o valor das constantes que deseja gerar.

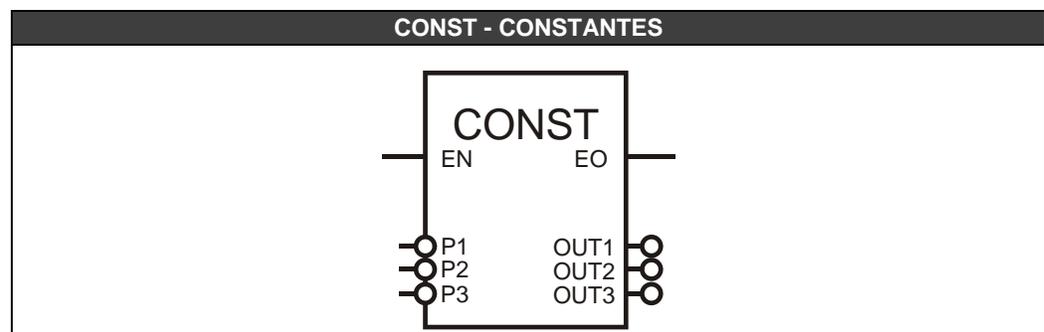
Por exemplo:

P1= 32

P2=346,87

P3= -456,5

Quando **EN** for verdadeiro, as saídas **OUT1**, **OUT2** e **OUT3** indicarão: 32, 346,87 e -456,5.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|-----------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | P1 | VALOR DA CONSTANTE 1 | FLOAT |
| | P2 | VALOR DA CONSTANTE 2 | FLOAT |
| | P3 | VALOR DA CONSTANTE 3 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT1 | SAÍDA DO VALOR ESTABELECIDO EM P1 | FLOAT |
| | OUT2 | SAÍDA DO VALOR ESTABELECIDO EM P2 | FLOAT |
| | OUT3 | SAÍDA DO VALOR ESTABELECIDO EM P3 | FLOAT |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Conversão de inteiro para booleano (ITB1)

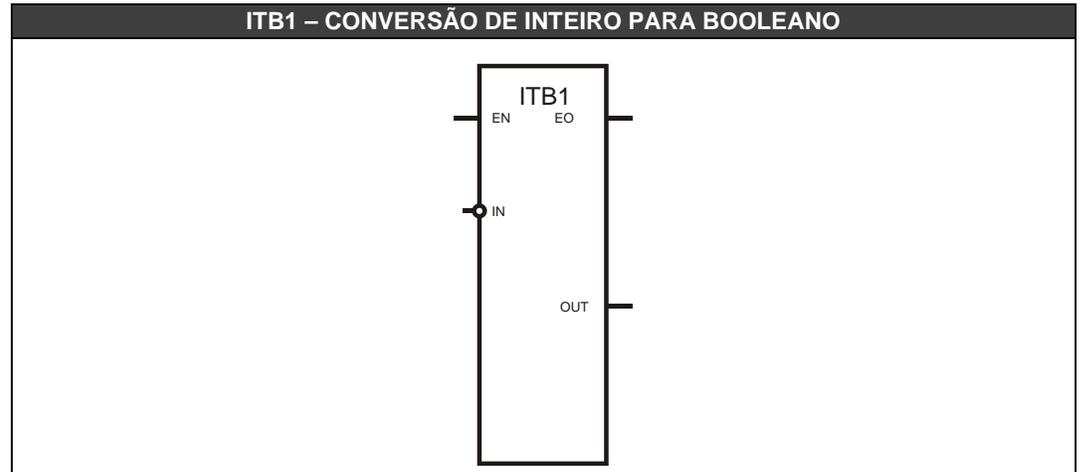
Descrição

Esta função, quando **EN** é verdadeira, converte um número inteiro na entrada **IN** para um estado booleano e coloca-o na saída **OUT**.

Conversão

Se o bit menos significativo da entrada **IN** for "0", a saída **OUT** terá o estado lógico falso.

Se o bit menos significativo da entrada **IN** for "1", a saída **OUT** terá o estado lógico verdadeiro.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA (BIT MENOS SIGNIFICATIVO DA ENTRADA CONVERTIDO PARA ESTADO LÓGICO) | BOOL |

I: Entrada O: Saída P: Parâmetro Interno

Conversão de inteiro para BCD (ITB2)

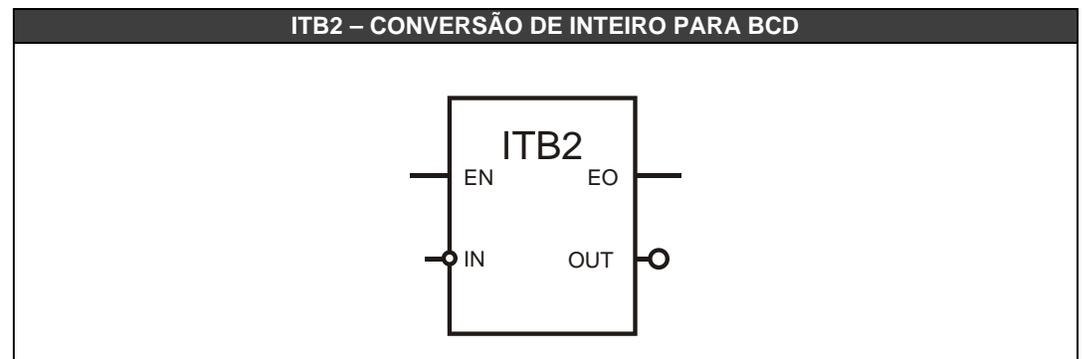
Descrição

Esta função, quando **EN** é verdadeira converte um número inteiro na entrada **IN** para o formato BCD e coloca-o na saída **OUT**.

Conversão

O dado inteiro da entrada **IN** será convertido para BCD, caso seja menor que 99. Caso a entrada seja maior que 99, a saída será colocada em 99BCD (1001 1001).

Por exemplo: na entrada **IN** tem-se a leitura 12. Na saída do bloco tem-se 12BCD ou 0001 0010.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA (VALOR DA ENTRADA CONVERTIDO PARA BCD) | LONG |

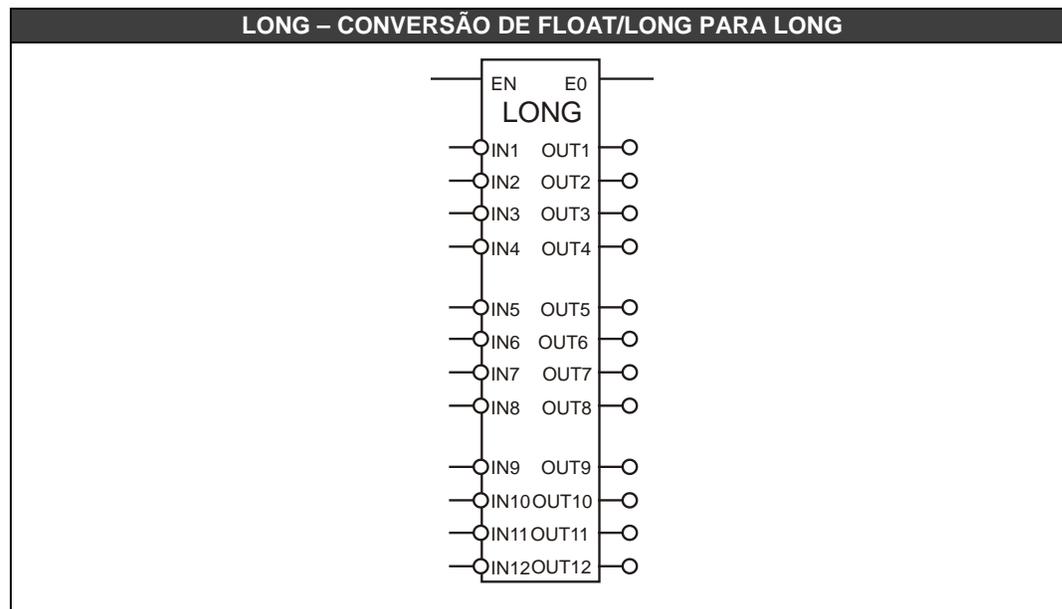
I: Entrada O: Saída P: Parâmetro Interno

Conversor de Float/Long para Long (LONG)

Descrição

Esta função, quando **EN** é verdadeira, multiplica os valores inteiros ou reais das entradas **INx** pelo valor definido no parâmetro **MUL**, converte em LONG e coloca os respectivos resultados nas saídas **OUTx**.

Se a entrada **EN** está falsa, as saídas serão mantidas em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|------------------------|------------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | LONG/FLOAT |
| | IN2 | ENTRADA 2 | LONG/FLOAT |
| | IN3 | ENTRADA 3 | LONG/FLOAT |
| | IN4 | ENTRADA 4 | LONG/FLOAT |
| | IN5 | ENTRADA 5 | LONG/FLOAT |
| | IN6 | ENTRADA 6 | LONG/FLOAT |
| | IN7 | ENTRADA 7 | LONG/FLOAT |
| | IN8 | ENTRADA 8 | LONG/FLOAT |
| | IN9 | ENTRADA 9 | LONG/FLOAT |
| | IN10 | ENTRADA 10 | LONG/FLOAT |
| | IN11 | ENTRADA 11 | LONG/FLOAT |
| IN12 | ENTRADA 12 | LONG/FLOAT | |
| P | MUL | VALOR DO MULTIPLICADOR | FLOAT |
| O | E0 | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT1 | SAÍDA 1 | LONG |
| | OUT2 | SAÍDA 2 | LONG |
| | OUT3 | SAÍDA 3 | LONG |
| | OUT4 | SAÍDA 4 | LONG |
| | OUT5 | SAÍDA 5 | LONG |
| | OUT6 | SAÍDA 6 | LONG |
| | OUT7 | SAÍDA 7 | LONG |
| | OUT8 | SAÍDA 8 | LONG |
| | OUT9 | SAÍDA 9 | LONG |
| | OUT10 | SAÍDA 10 | LONG |
| | OUT11 | SAÍDA 11 | LONG |
| OUT12 | SAÍDA 12 | LONG | |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Multiplexador para entradas booleanas (MUX1)

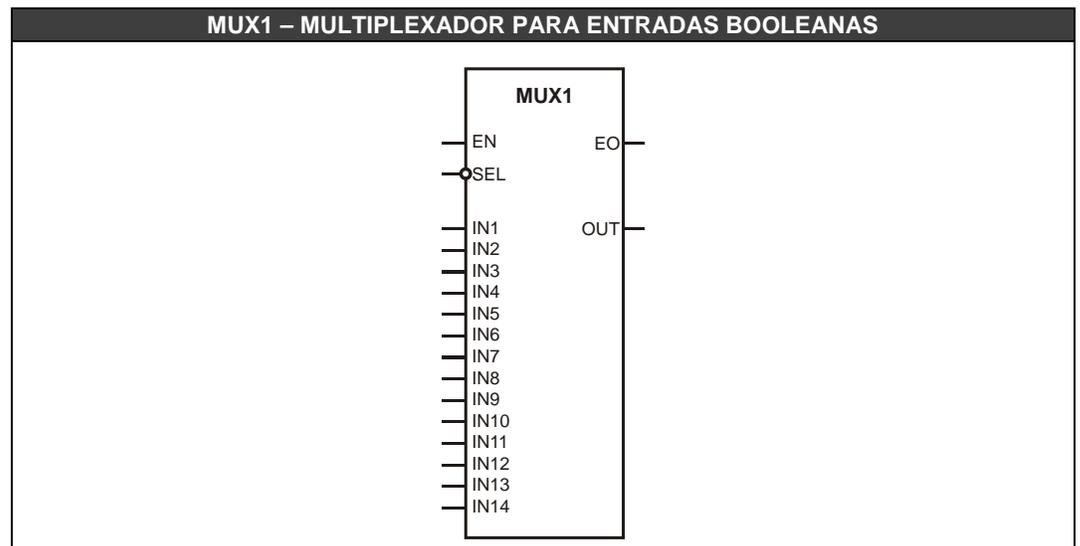
Descrição

Esta função, quando **EN** é verdadeira, seleciona uma das entradas **IN** e coloca o seu valor na saída **OUT**. A seleção é feita de acordo com o valor da entrada **SEL**.

Seleção da Saída

Se **SEL** for igual a 0, a saída selecionada será **IN1**, caso **SEL**= 1 a saída selecionada será **IN2** e assim sucessivamente. Caso **SEL** for maior do que o número de entradas possíveis (n-1) a saída será igual a **INn**. Neste caso, a saída **EO** irá para Falso mostrando que a entrada **SEL** está fora da faixa. Se o número **N_IN** for maior do que 14 ou menor do que 2, as saídas **EO** e **OUT** irão para zero.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | SEL | SELEÇÃO DA ENTRADA | LONG |
| | IN1 | ENTRADA 1 | BOOL |
| | IN2 | ENTRADA 2 | BOOL |
| | IN3 | ENTRADA 3 | BOOL |
| | IN4 | ENTRADA 4 | BOOL |
| | IN5 | ENTRADA 5 | BOOL |
| | IN6 | ENTRADA 6 | BOOL |
| | IN7 | ENTRADA 7 | BOOL |
| | IN8 | ENTRADA 8 | BOOL |
| | IN9 | ENTRADA 9 | BOOL |
| | IN10 | ENTRADA 10 | BOOL |
| | IN11 | ENTRADA 11 | BOOL |
| | IN12 | ENTRADA 12 | BOOL |
| IN13 | ENTRADA 13 | BOOL | |
| IN14 | ENTRADA 14 | BOOL | |
| P | N_IN | NÚMERO DE ENTRADAS | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | BOOL |

I: Entrada O: Saída P: Parâmetro Interno

Multiplexador para entradas booleanas reduzido (MUX1r)

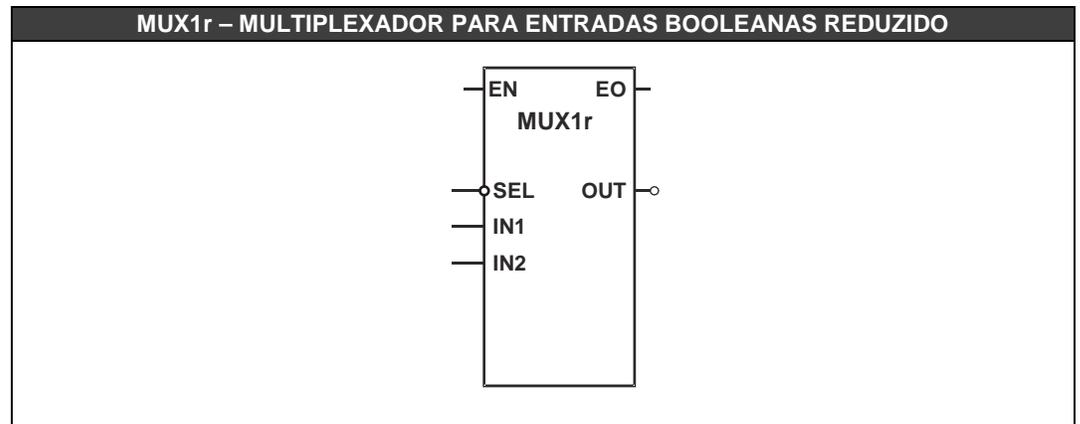
Descrição

Esta função, quando **EN** é verdadeira, seleciona uma das duas entradas **IN** e coloca o seu valor na saída **OUT**. A seleção é feita de acordo com o valor da entrada **SEL**.

Seleção da Saída

Se **SEL** for igual a 0, a saída selecionada será **IN1**, para qualquer outro valor de **SEL** a saída selecionada será **IN2**.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|----------|------|------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | SEL | SELEÇÃO DA ENTRADA | LONG |
| | IN1 | ENTRADA 1 | BOOL |
| | IN2 | ENTRADA 2 | BOOL |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | BOOL |

I: Entrada O: Saída P: Parâmetro Interno

Multiplexador para entradas float (MUX2)

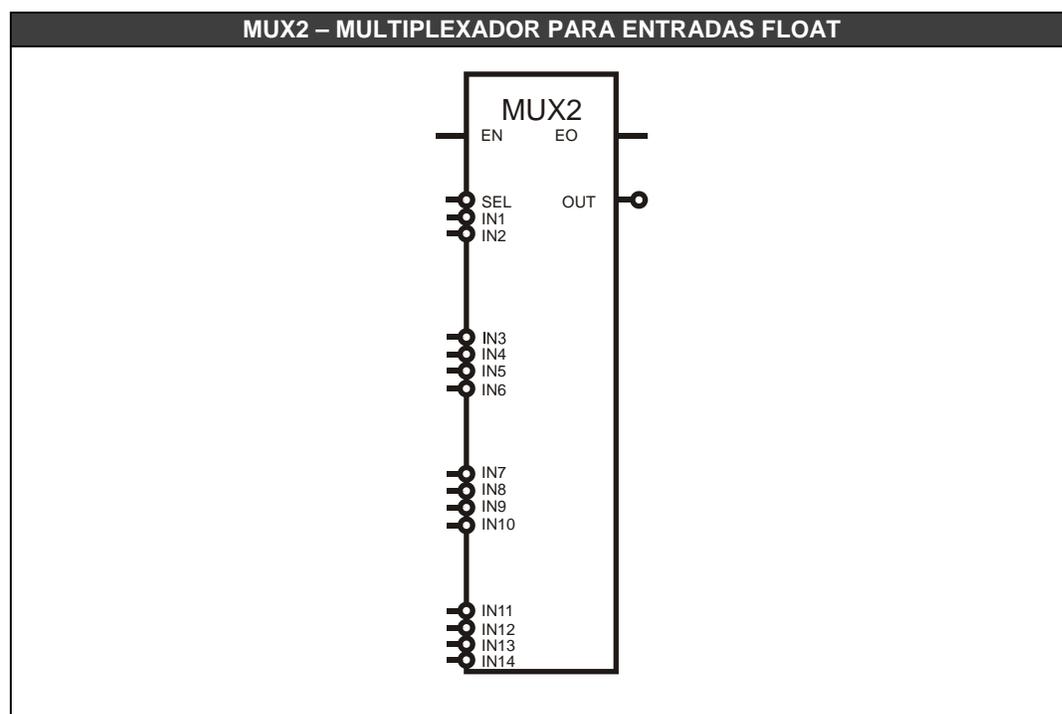
Descrição

Esta função, quando **EN** é verdadeira, seleciona uma das entradas **IN** e coloca o seu valor na saída **OUT**. A seleção é feita de acordo com o valor da entrada **SEL**.

Seleção da Saída

Se **SEL** for igual a 0, a saída selecionada será **IN1**, caso **SEL**= 1 a saída selecionada será **IN2** e assim sucessivamente. Caso **SEL** for maior do que o número de entradas possíveis (n-1) a saída será igual a **INn**. Neste caso, a saída **EO** irá para Falso mostrando que a entrada **SEL** está fora da faixa. Se o número **N_IN** for maior do que 14 ou menor do que 2, as saídas **EO** e **OUT** irão para zero.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | SEL | SELEÇÃO DA ENTRADA | LONG |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| | IN3 | ENTRADA 3 | FLOAT |
| | IN4 | ENTRADA 4 | FLOAT |
| | IN5 | ENTRADA 5 | FLOAT |
| | IN6 | ENTRADA 6 | FLOAT |
| | IN7 | ENTRADA 7 | FLOAT |
| | IN8 | ENTRADA 8 | FLOAT |
| | IN9 | ENTRADA 9 | FLOAT |
| | IN10 | ENTRADA 10 | FLOAT |
| | IN11 | ENTRADA 11 | FLOAT |
| | IN12 | ENTRADA 12 | FLOAT |
| IN13 | ENTRADA 13 | FLOAT | |
| IN14 | ENTRADA 14 | FLOAT | |
| P | N_IN | NÚMERO DE ENTRADAS | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Multiplexador para entradas float reduzido (MUX2r)

Descrição

Esta função, quando **EN** é verdadeira, seleciona uma das duas entradas **IN** e coloca o seu valor na saída **OUT**. A seleção é feita de acordo com o valor da entrada **SEL**.

Seleção da Saída

Se **SEL** for igual a 0, a saída selecionada será **IN1**, para qualquer outro valor de **SEL** a saída selecionada será **IN2**.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | SEL | SELEÇÃO DA ENTRADA | LONG |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Operação NOT para uma entrada booleana (NOT1)

Descrição

Esta função, quando **EN** é verdadeira, é usada para fazer a inversão de um dado booleano. Se a entrada for “verdadeira”, isto é, nível lógico “1”, o bloco NOT1 fará a saída igual a “falso” (nível lógico “0”). E vice-versa.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | FLOAT |

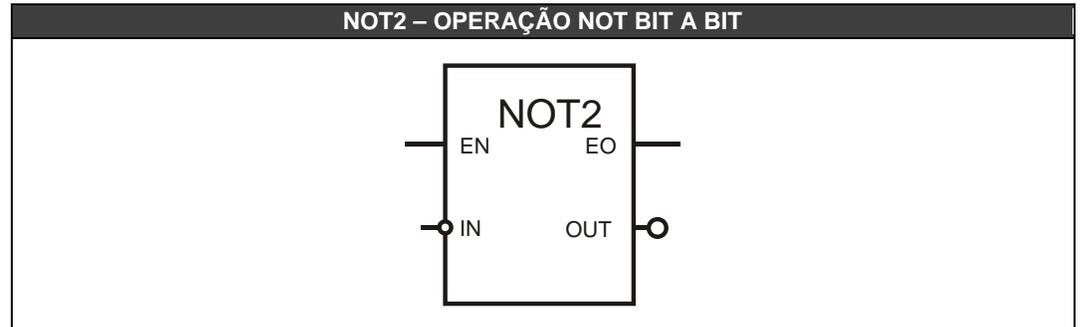
I: Entrada O: Saída P: Parâmetro Interno

Operação NOT Bit a Bit (NOT2)

Descrição

Esta função, quando **EN** é verdadeira, é usada para fazer a inversão do nível lógico da entrada. O byte menos significativo da entrada terá cada um de seus bits invertidos logicamente. A operação é feita bit a bit, por exemplo: se a entrada possui o byte menos significativo igual a "10011000" (binário), a saída será "01100111" (binário).

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Seleção Binária das Saídas (OSEL)

Descrição

Esta função, quando **EN** é verdadeira, é usada para selecionar a saída para onde será enviado o valor da entrada **IN**. Se a entrada **SEL** for igual a zero a saída **OUT1** será selecionada. Se **SEL** for igual 1 será selecionada a saída **OUT2**.

Caso a saída **OUT1** seja selecionada, o parâmetro **Prm2** define o valor desejado para a saída **OUT2**, conforme segue:

Prm2 = verdadeiro: envia zero para a saída **OUT2**.

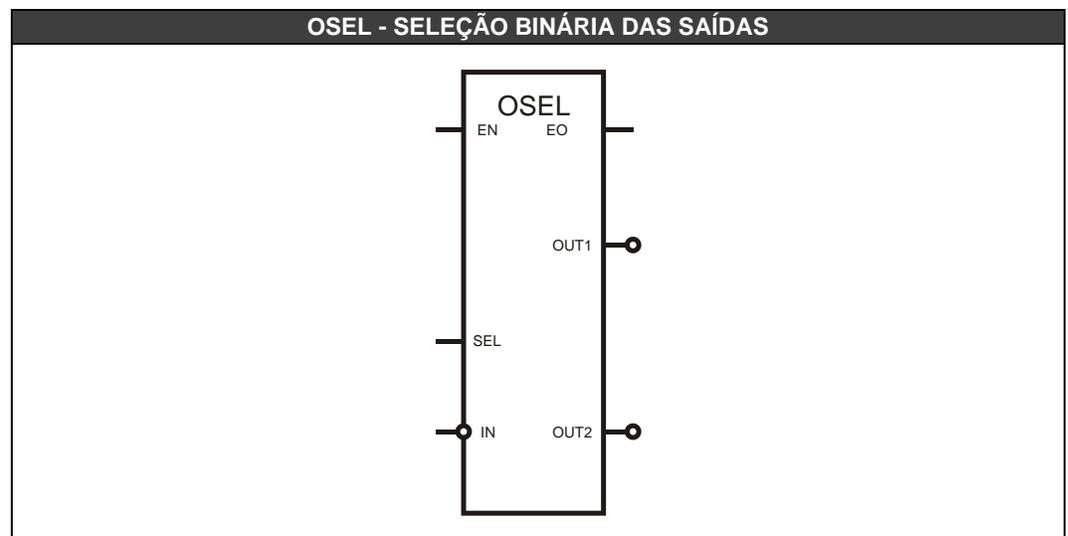
Prm2 = falso: mantém o último valor da saída **OUT2**.

Caso a saída **OUT2** seja selecionada, o parâmetro **Prm1** define o valor desejado para a saída **OUT1**, conforme segue:

Prm1 = verdadeiro: envia zero para a saída **OUT1**.

Prm1 = falso: mantém o último valor da saída **OUT1**.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | SEL | SELEÇÃO DA SAÍDA | BOOL |
| | IN | ENTRADA | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT1 | SAÍDA 1 | FLOAT |
| | OUT2 | SAÍDA 2 | FLOAT |
| P | Prm1 | SELEÇÃO DO VALOR PARA OUT1 NÃO SELECIONADA | BOOL |
| | Prm2 | SELEÇÃO DO VALOR PARA OUT2 NÃO SELECIONADA | BOOL |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Seleção Binária para Entradas Booleanas (SEL1)

Descrição

Esta função, quando **EN** é verdadeira, é usada para selecionar entre as duas entradas **IN1** e **IN2** e redirecioná-las para a saída **OUT**. A entrada **SEL** funciona como uma chave. Se **SEL** é Falso, o estado de **IN1** será transcrito para **OUT**, caso contrário, o valor de **IN2** é que será transcrito.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | SEL | SELEÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | BOOL |
| | IN2 | ENTRADA 2 | BOOL |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA DO BLOCO | BOOL |

I: Entrada O: Saída P: Parâmetro Interno

Seleção Binária para Entradas Float (SEL2)

Descrição

Esta função, quando **EN** é verdadeira, é usada para selecionar entre as duas entradas **P1** e **P2** e redirecioná-las para a saída **OUT**. A entrada **SEL** funciona como uma chave. Se **SEL** é falso, **P1** será transcrito para **OUT**, caso contrário, o valor de **P2** é que será transcrito.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|----------|------|------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | SEL | SELEÇÃO DA ENTRADA | BOOL |
| | P1 | ENTRADA 1 | FLOAT |
| | P2 | ENTRADA 2 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA DO BLOCO | FLOAT |

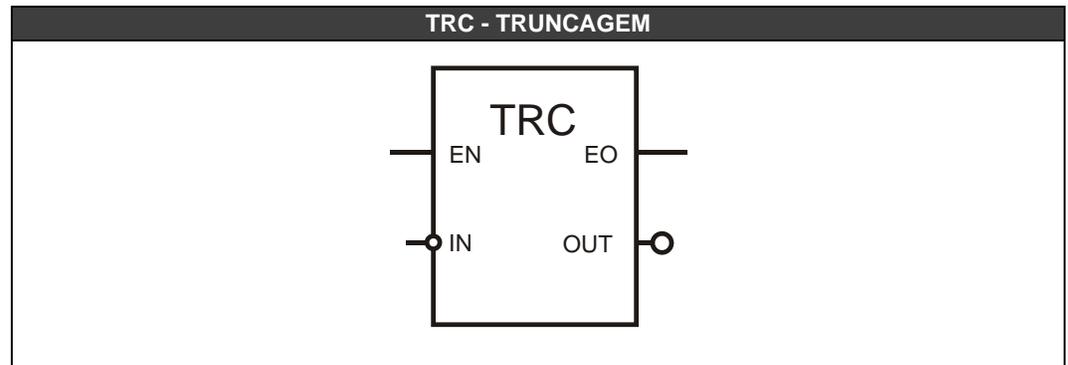
I: Entrada **O:** Saída **P:** Parâmetro Interno

Truncagem (TRC)

Descrição

Esta função, quando **EN** é verdadeira, é usada para fazer a truncagem de um número real, ou seja, a saída terá apenas a parte inteira do número de entrada.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

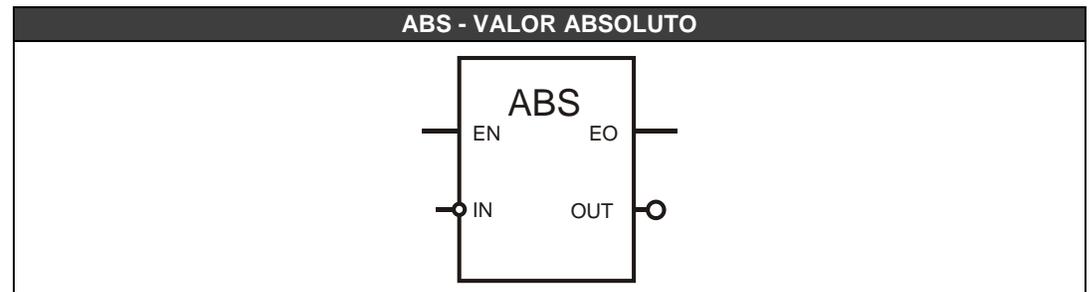
Funções Matemáticas

Valor Absoluto (ABS)

Descrição

Esta função, quando **EN** é verdadeira, coloca o valor absoluto da entrada **IN** na saída **OUT**. Por exemplo: se a leitura for $-0,78987$ na entrada **IN**, a saída **OUT** será $0,78987$.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA DO BLOCO | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA DO BLOCO. VALOR ABSOLUTO DA ENTRADA. | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Adição (ADD)

Descrição

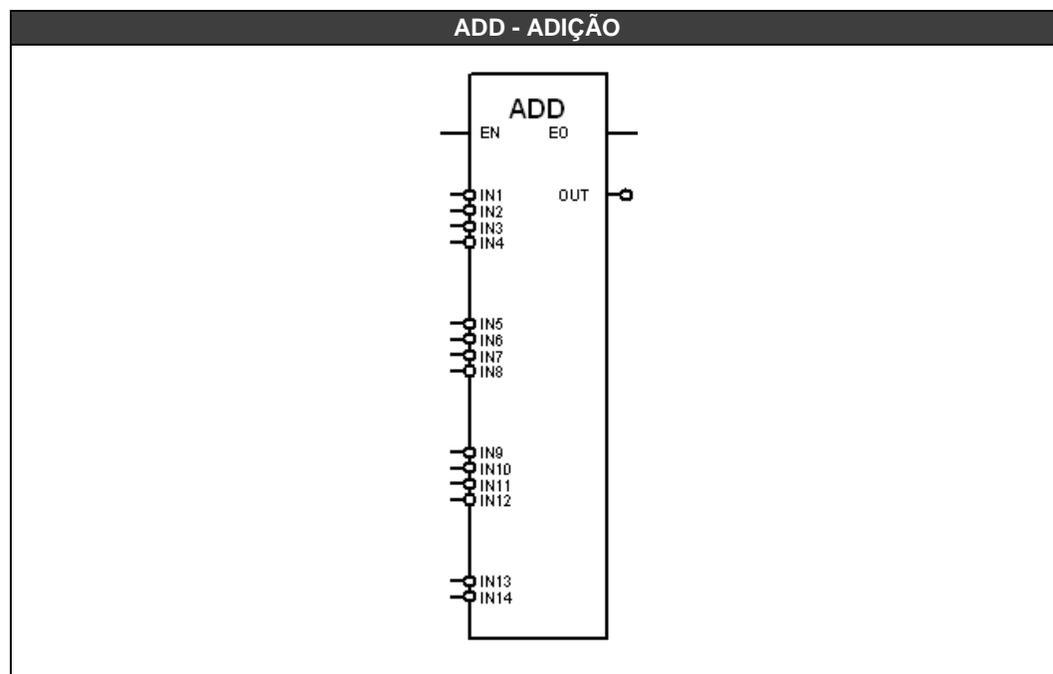
Esta função, quando **EN** é verdadeiro, soma os valores das entradas utilizadas e disponibiliza o resultado na saída **OUT**. A definição das entradas utilizadas é feita através do parâmetro **N_IN**. Por exemplo:

N_IN = 5

OUT receberá o valor de **IN1 + IN2 + IN3 + IN4 + IN5**

O número mínimo de entradas utilizadas é 2 e o número máximo é 14.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|---------------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| | IN3 | ENTRADA 3 | FLOAT |
| | IN4 | ENTRADA 4 | FLOAT |
| | IN5 | ENTRADA 5 | FLOAT |
| | IN6 | ENTRADA 6 | FLOAT |
| | IN7 | ENTRADA 7 | FLOAT |
| | IN8 | ENTRADA 8 | FLOAT |
| | IN9 | ENTRADA 9 | FLOAT |
| | IN10 | ENTRADA 10 | FLOAT |
| | IN11 | ENTRADA 11 | FLOAT |
| | IN12 | ENTRADA 12 | FLOAT |
| | IN13 | ENTRADA 13 | FLOAT |
| IN14 | ENTRADA 14 | FLOAT | |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO DA SOMA | FLOAT |
| P | N_IN | NÚMERO DE ENTRADAS A SEREM UTILIZADAS | LONG |

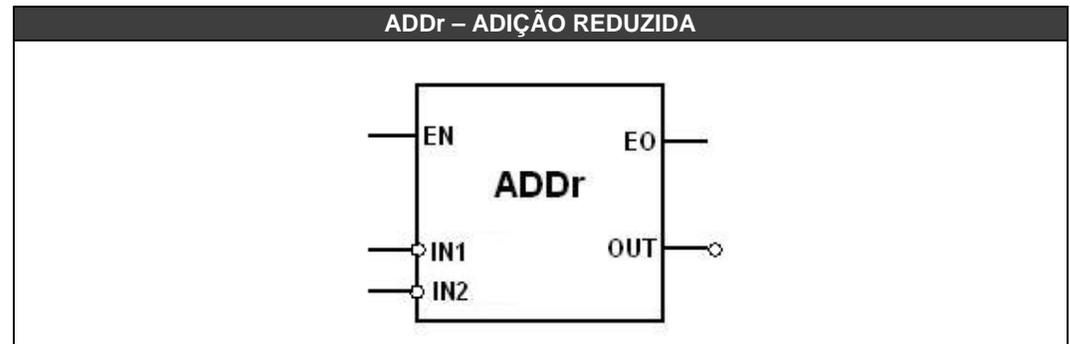
I: Entrada O: Saída P: Parâmetro Interno

Adição Reduzida (ADDr)

Descrição

Esta função, quando **EN** é verdadeiro, soma os valores das entradas **IN1** e **IN2** e disponibiliza o resultado na saída **OUT**

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| O | E0 | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO DA SOMA | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

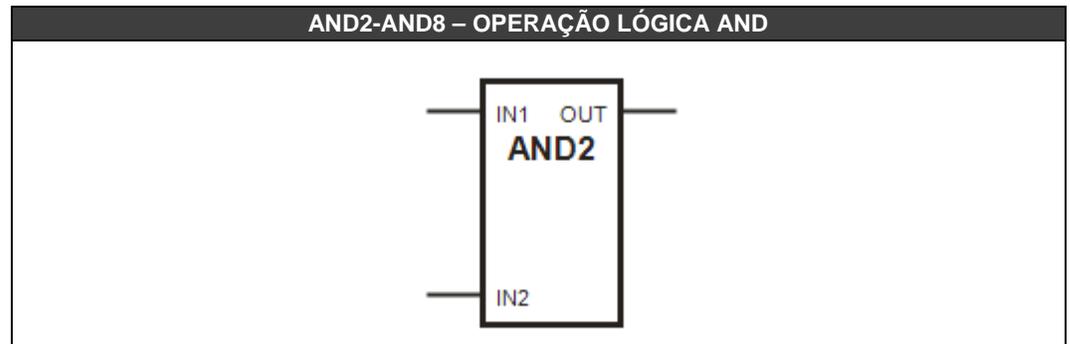
Operação Lógica AND de 2 a 8 entradas (AND2-AND8)

Descrição

Esta função realiza a função lógica **AND** para as entradas **IN1** e **IN2** até **IN8** disponibilizando o resultado em **OUT**.

Tabela verdade:

Se **IN1** até **INn** igual a 1, **OUT** será igual a 1, caso contrário será igual a 0.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|-----------------------|------|
| I | IN1 | ENTRADA 1 | BOOL |
| | INn | ENTRADA n | BOOL |
| O | OUT | RESULTADO DA OPERAÇÃO | BOOL |

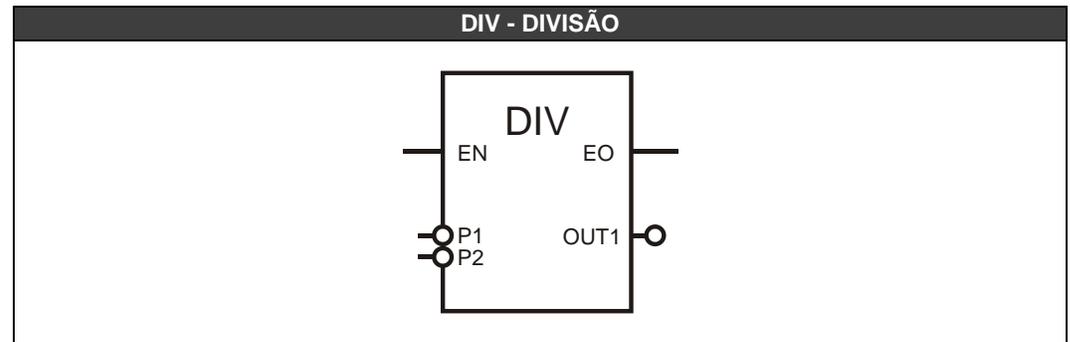
I: Entrada O: Saída P: Parâmetro Interno

Divisão (DIV)

Descrição

Esta função, quando **EN** é verdadeiro, divide os valores das entradas **P1** e **P2** e disponibiliza o resultado na saída **OUT**.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | P1 | ENTRADA DO DIVIDENDO | FLOAT |
| | P2 | ENTRADA DO DIVISOR | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO DA DIVISÃO | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Módulo (MDL)

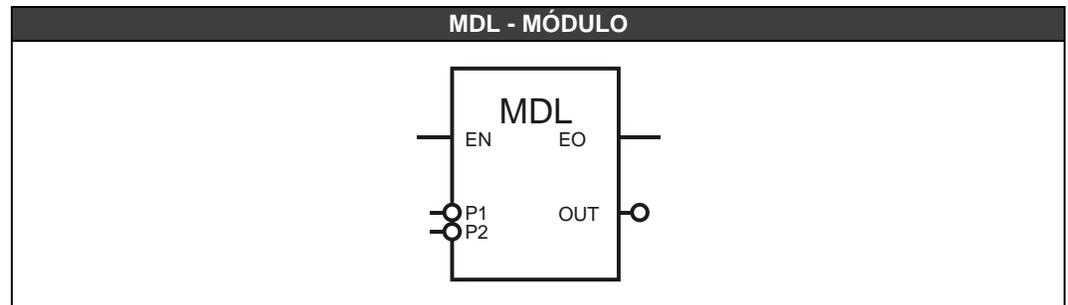
Descrição

Esta função toma o resto da divisão de **P1** por **P2** e coloca este valor na saída **OUT**, se a entrada **EN** for verdadeira.

Operação

Por exemplo: P1= 25 e P2= 7, a saída será OUT= 4. Pois:

$$\begin{array}{r} 25 \quad | \quad 7 \\ 4 \leftarrow 3 \end{array}$$



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | P1 | ENTRADA DO DIVIDENDO | FLOAT |
| | P2 | ENTRADA DO DIVISOR | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESTO DA DIVISÃO | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Multiplicação (MUL)

Descrição

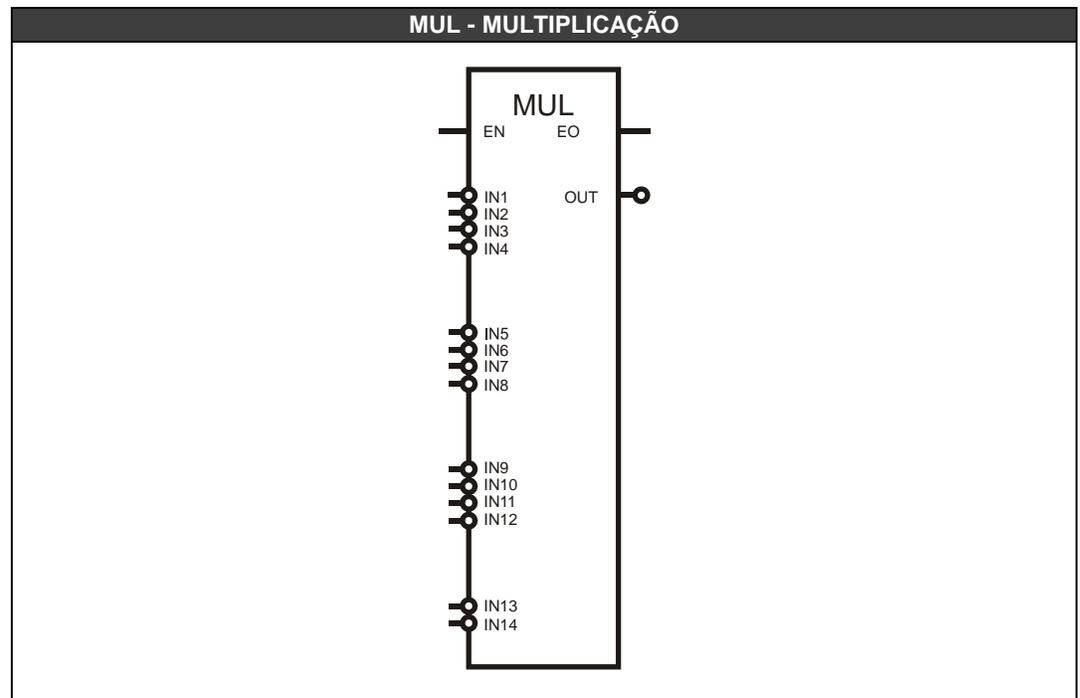
Esta função, quando **EN** é verdadeiro, multiplica os valores das entradas utilizadas e disponibiliza o resultado na saída **OUT**. A definição das entradas utilizadas é feita através do parâmetro **N_IN**. Por exemplo:

N_IN = 5

OUT receberá o valor de **IN1 * IN2 * IN3 * IN4 * IN5**

O número mínimo de entradas utilizadas é 2 e o número máximo é 14.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|---------------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| | IN3 | ENTRADA 3 | FLOAT |
| | IN4 | ENTRADA 4 | FLOAT |
| | IN5 | ENTRADA 5 | FLOAT |
| | IN6 | ENTRADA 6 | FLOAT |
| | IN7 | ENTRADA 7 | FLOAT |
| | IN8 | ENTRADA 8 | FLOAT |
| | IN9 | ENTRADA 9 | FLOAT |
| | IN10 | ENTRADA 10 | FLOAT |
| | IN11 | ENTRADA 11 | FLOAT |
| | IN12 | ENTRADA 12 | FLOAT |
| | IN13 | ENTRADA 13 | FLOAT |
| IN14 | ENTRADA 14 | FLOAT | |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO DA MULTIPLICAÇÃO | FLOAT |
| P | N_IN | NÚMERO DE ENTRADAS A SEREM UTILIZADAS | LONG |

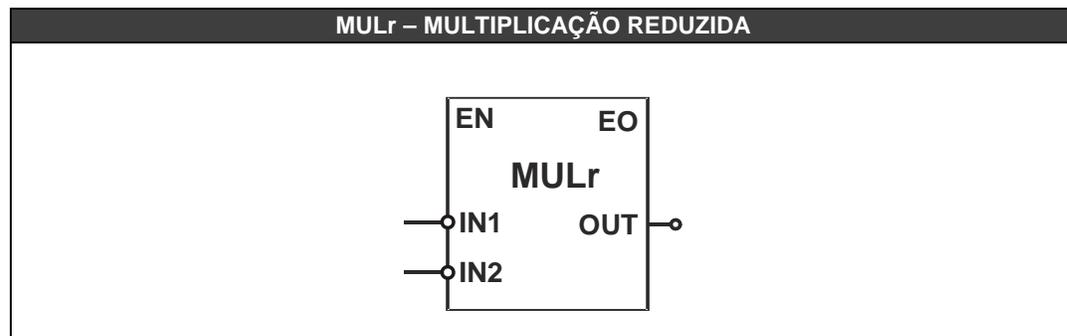
I: Entrada **O:** Saída **P:** Parâmetro Interno

Multiplicação Reduzida (MULr)

Descrição

Esta função, quando **EN** é verdadeiro, multiplica os valores das entradas **IN1** e **IN2** e disponibiliza o resultado na saída **OUT**.

Se a entrada **EN** está falsa, a saída será mantida em zero.



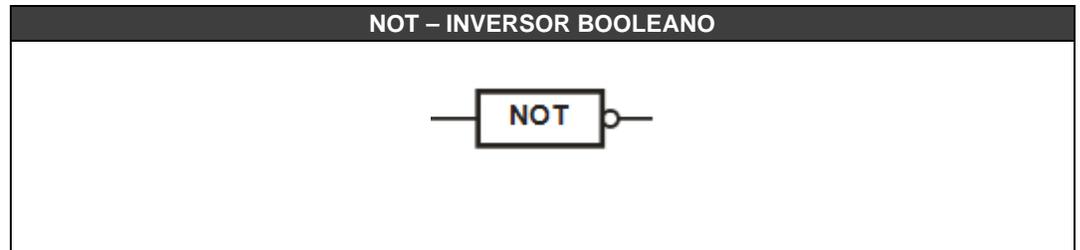
| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|----------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO DA MULTIPLICAÇÃO | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Inversor Booleano (NOT)

Descrição

Esta função é usada para fazer a inversão de um dado booleano. Se a entrada for “verdadeira”, isto é, nível lógico “1”, o bloco NOT fará a saída igual a “falso” (nível lógico “0”). E vice-versa.



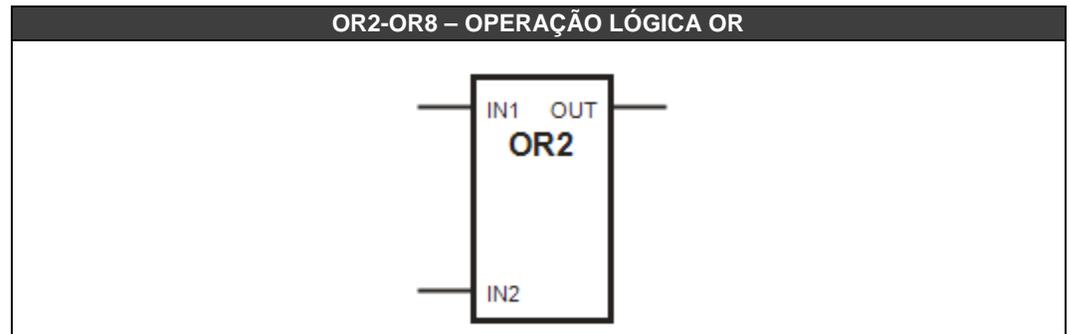
Operação Lógica OR de 2 a 8 entradas (OR2-OR8)

Descrição

Esta função realiza a função lógica **OR** para as entradas **IN1** e **IN2** até **IN8** disponibilizando o resultado em **OUT**.

Tabela verdade:

Se IN1 até INn igual a 0, OUT será igual a 0, caso contrário será igual a 1.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|-----------------------|------|
| I | IN1 | ENTRADA 1 | BOOL |
| | INn | ENTRADA n | BOOL |
| O | OUT | RESULTADO DA OPERAÇÃO | BOOL |

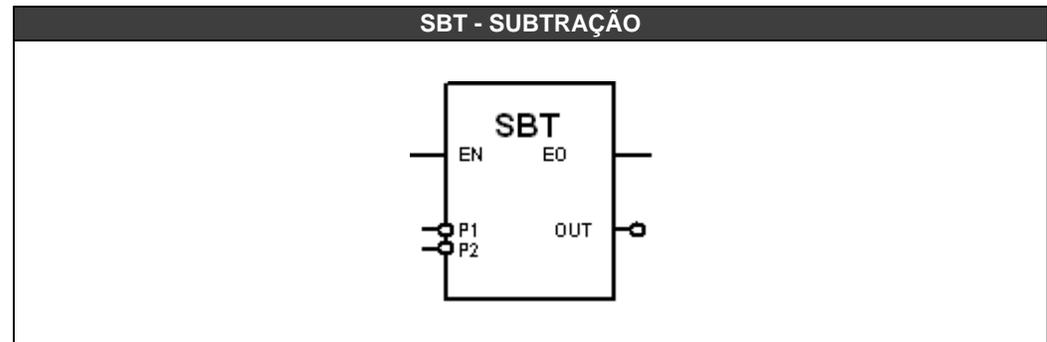
I: Entrada O: Saída P: Parâmetro Interno

Subtração (SBT)

Descrição

Esta função, quando **EN** é verdadeiro, subtrai os valores das entradas **P1** e **P2** e disponibiliza o resultado na saída **OUT**.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|---------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | P1 | 1.o ELEMENTO DA SUBTRAÇÃO | FLOAT |
| | P2 | 2.o ELEMENTO DA SUBTRAÇÃO | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO DA SUBTRAÇÃO | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Raiz Quadrada (SQR)

Descrição

Esta função, quando **EN** é verdadeiro, obtém a raiz quadrada do valor na entrada **IN** e coloca o resultado na saída **OUT**. Se a entrada **IN** é negativa, o resultado irá para zero e a saída **EO** irá para falso.

Seleção do tipo de dado

O tipo de dado na entrada e na saída (Regular ou Porcentagem) é selecionado pelo parâmetro **Prm1**.

Para a opção Regular (**Prm1** = falso), o bloco extrai a raiz quadrada da entrada.

Para a opção Porcentagem (**Prm1** = verdadeiro), tem-se duas subopções:

-parâmetro **PERC** = falso:

$$OUT = 10 * \sqrt{IN}$$

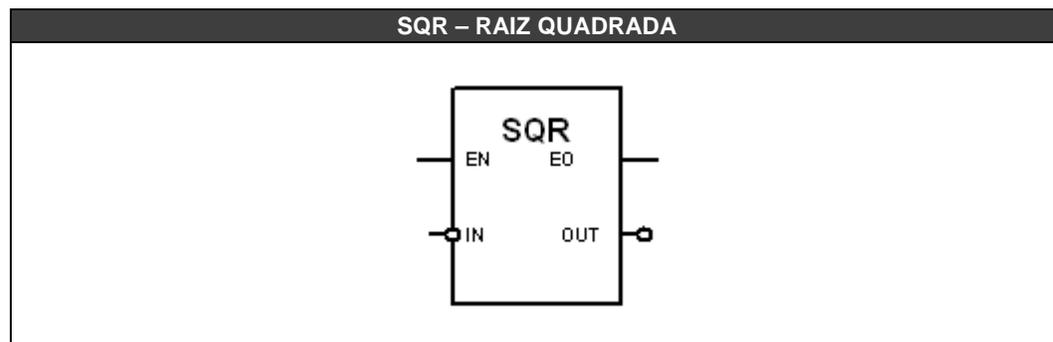
-parâmetro **PERC** = verdadeiro:

$$OUT = 100 * \sqrt{IN}$$

Nivelamento

Caso a entrada **IN** tenha um valor menor que o valor especificado no parâmetro **CTO**, a saída receberá o valor zero. Caso seja especificado um valor negativo para **CTO**, será assumido o valor zero.

Se a entrada **EN** está falsa, a saída será mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|---|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | ENTRADA DO BLOCO | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA DO BLOCO. VALOR ABSOLUTO DA ENTRADA | FLOAT |
| P | Prm1 | TIPO DE DADO NA ENTRADA E NA SAÍDA (REGULAR OU PORCENTAGEM) | BOOL |
| | PERC | SELECIONA MÉTODO DE CÁLCULO PARA ENTRADA PORCENTAGEM | BOOL |
| | CTO | NIVELAMENTO (CUT-OFF) | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Funções de Comparação

Quádruplo Alarme (AI-Seta)

Descrição

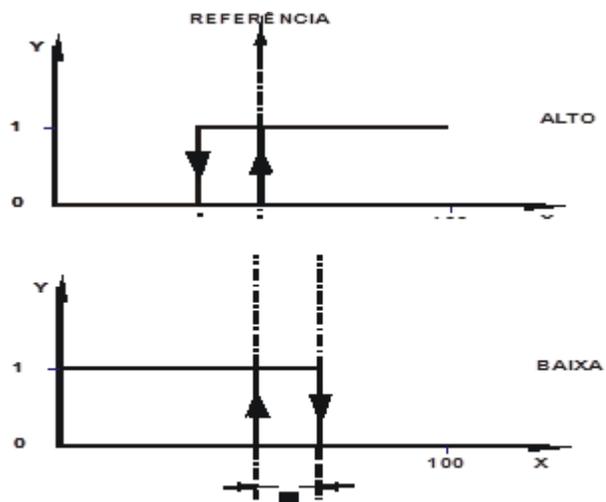
Esta função, quando a entrada **EN** está verdadeira, atua como um quádruplo alarme duplo, ou seja, ela compara um sinal de entrada em **IN** com quatro valores de referências: **LL**, **L**, **H** e **HH**.

A variável a ser comparada é conectada à entrada **IN1** e o sinal de referência nas entradas **LL**, **L**, **H** e **HH** é somado aos valores dos parâmetros internos **AGL**, **AGLL**, **AGH** e **AGHH**, respectivamente. Estas comparações dispararão as saídas **LLow**, **Low**, **High** e **HHigh** se elas forem menor, menor, maior e maior, respectivamente.

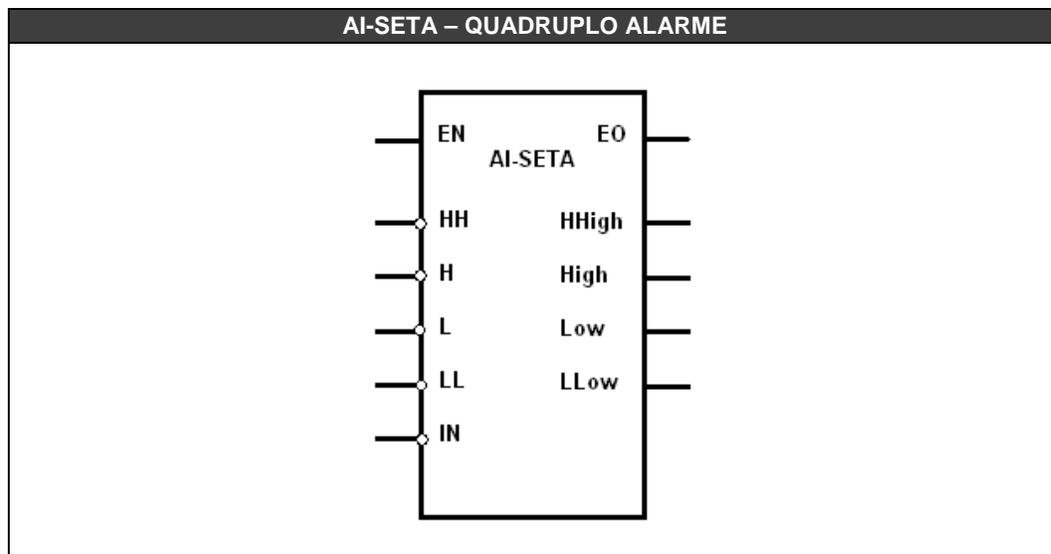
O parâmetro interno **State** indica se as saídas estarão normais em 0, e em alarme em 1, ou vice-versa.

Para se evitar a oscilação do sinal de saída quando a variável está muito próxima da referência, pode ser ajustado um valor de histerese pelo parâmetro **DBL**, **DBLL**, **DBH**, **DBHH**.

O bloco trabalha de acordo com a figura a seguir:



Ação de alarme com histerese



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|--------|---|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA | FLOAT |
| | LL | REFERÊNCIA PARA O ALARME DE BAIXA-BAIXA | FLOAT |
| | L | REFERÊNCIA PARA O ALARME DE BAIXA | FLOAT |
| | H | REFERÊNCIA PARA O ALARME DE ALTA | FLOAT |
| | HH | REFERÊNCIA PARA O ALARME DE ALTA-ALTA | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | LLow | SAÍDA ALARME BAIXA-BAIXA | BOOL |
| | Low | SAÍDA ALARME BAIXA | BOOL |
| | High | SAÍDA ALARME ALTA | BOOL |
| | HHHigh | SAÍDA ALARME ALTA-ALTA | BOOL |
| P | STATE | ESTADO DO DISPARO DO ALARME | LONG |
| | DBLL | HISTERESE ALARME LL | FLOAT |
| | AGLL | VALOR SOMADO A REFERÊNCIA PARA CÁLCULO DO ALARME LL | FLOAT |
| | DBL | HISTERESE ALARME L | FLOAT |
| | AGL | VALOR SOMADO A REFERÊNCIA PARA CÁLCULO DO ALARME L | FLOAT |
| | DBH | HISTERESE ALARME H | FLOAT |
| | AGH | VALOR SOMADO A REFERÊNCIA PARA CÁLCULO DO ALARME H | FLOAT |
| | DBHH | HISTERESE ALARME HH | FLOAT |
| | AGHH | VALOR SOMADO A REFERÊNCIA PARA CÁLCULO DO ALARME HH | FLOAT |

Alarme Duplo (ALM)

Descrição

Esta função, quando a entrada **EN** está verdadeira, atua como um alarme duplo, ou seja, ele possui dois comparadores de alarme independentes.

No primeiro comparador a variável a ser comparada é conectada à entrada **IN1** e o sinal de referência na entrada **REF1** é somado ao valor do parâmetro interno **ARG1**. O segundo comparador é igual ao primeiro, isto é, as entradas **IN2**, **REF2** e **ARG2** são usadas do mesmo modo que as entradas **IN1**, **REF1** e **ARG1**.

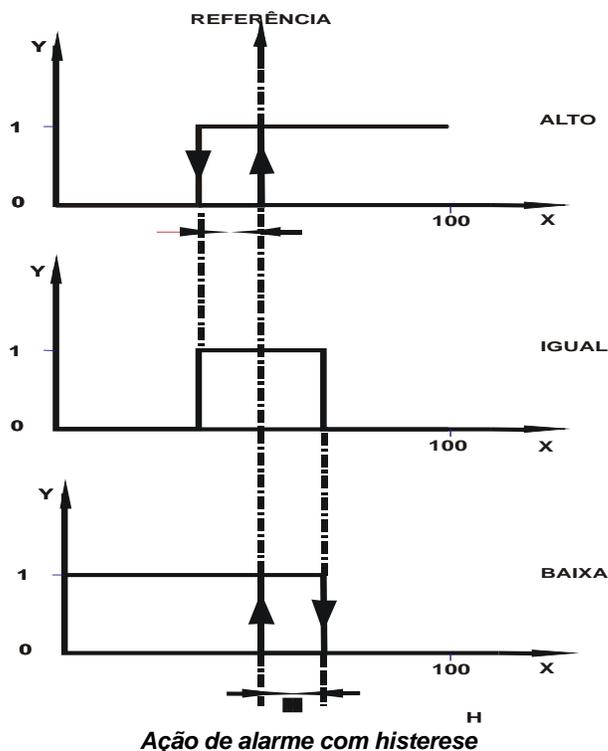
Cada comparador pode ser configurado independentemente para gerar saída de alarme de acordo com as opções:

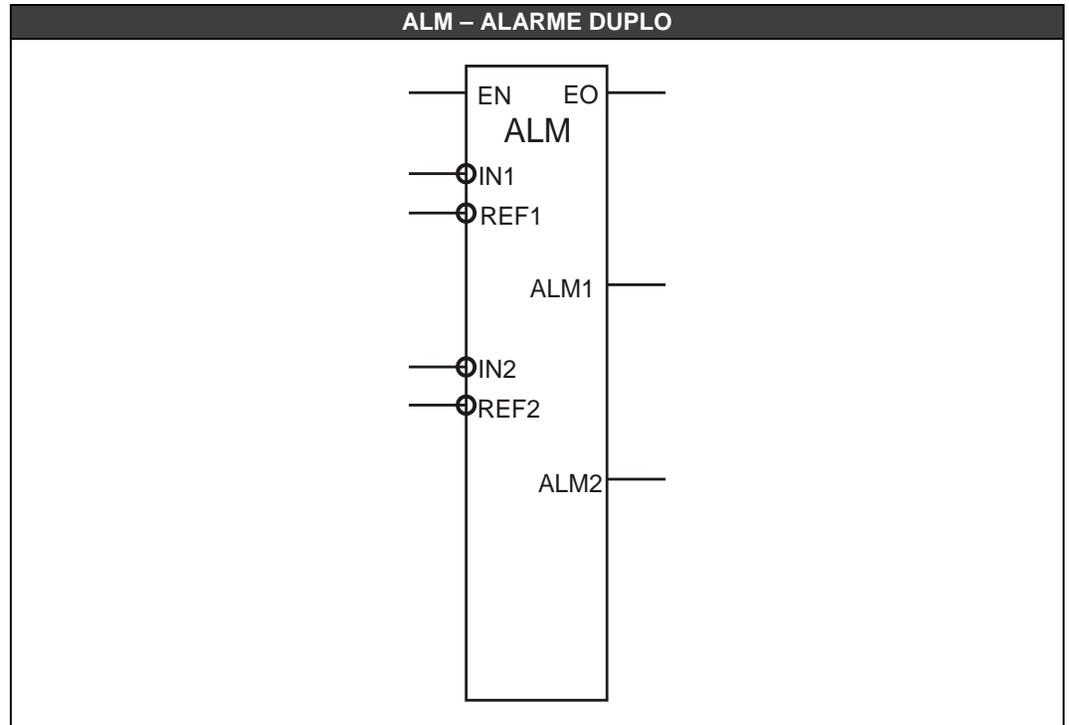
- Variável \leq Referência \rightarrow Alarme de baixa
- Variável \geq Referência \rightarrow Alarme de alta
- Variável = Referência \rightarrow Alarme de igualdade

A referência é a soma da entrada **REF1** (ou **REF2**) e o valor do parâmetro **ARG1** (ou **ARG2**).

Para se evitar a oscilação do sinal de saída quando a variável está muito próxima da referência, pode ser ajustado um valor de histerese pelo parâmetro **DBN1** (ou **DBN2**).

O bloco trabalha de acordo com a figura a seguir:





| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|---|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA PARA O ALARME1 | FLOAT |
| | REF1 | REFERÊNCIA PARA O ALARME1 | FLOAT |
| | IN2 | ENTRADA PARA O ALARME2 | FLOAT |
| | REF2 | REFERÊNCIA PARA O ALARME2 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | ALM1 | SAÍDA ALARME1 | BOOL |
| | ALM2 | SAÍDA ALARME2 | BOOL |
| P | TYPE1 | TIPO DO ALARME 1 | LONG |
| | DBN1 | HISTERESE ALARME 1 | FLOAT |
| | ARG1 | VALOR SOMADO A REFERÊNCIA PARA CÁLCULO DO ALARME1 | FLOAT |
| | TYPE2 | TIPO DO ALARME 2 | LONG |
| | DBN2 | HISTERESE ALARME 2 | FLOAT |
| | ARG2 | VALOR SOMADO A REFERÊNCIA PARA CÁLCULO DO ALARME2 | FLOAT |

Desigualdade (DIF)

Descrição

Esta função, quando a entrada **EN** está verdadeira, mantém a saída **OUT** em verdadeiro se o valor da diferença entre **IN1** e **IN2** ($IN1 - IN2$) for maior que **DBN** (zona morta). Caso contrário, a saída **OUT** se mantém em falso.

O parâmetro **DBN** é configurado pelo usuário.

Se a entrada **EN** está falsa, a saída é mantida em falso.

Exemplo:

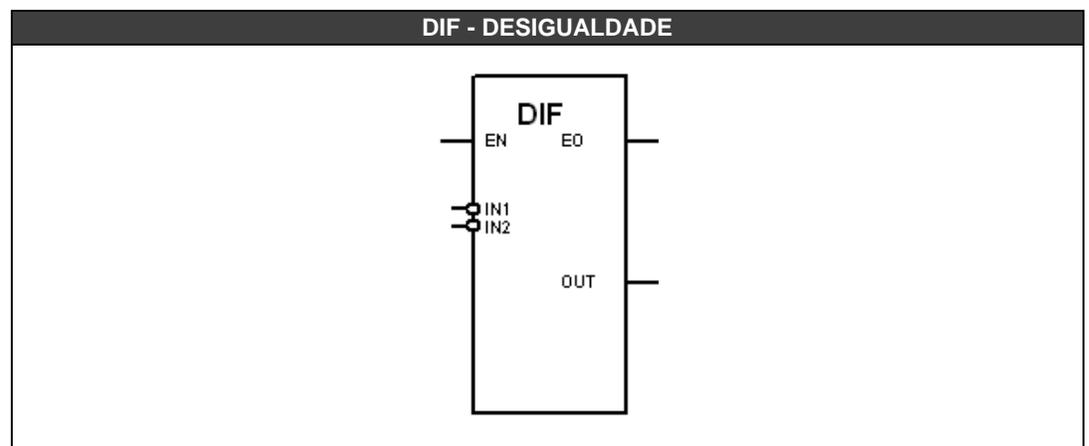
IN1= 0,78

IN2= 0,70

IN1 - IN2=0,08

DBN= 0,05

Neste caso a saída **OUT** = 1 (verdadeiro), pois o valor configurado para **DBN** (0,05) indica que na aplicação acima **IN1** é diferente de **IN2**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--------------------------------|-------|
| I | IN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO LÓGICO DA COMPARAÇÃO | BOOL |
| P | DBN | ZONA MORTA | FLOAT |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Igualdade (EQ)

Descrição

Esta função, quando a entrada **EN** está verdadeira, mantém a saída **OUT** em verdadeiro se os valores das entradas utilizadas não sofrerem desvio maior do que o valor da Zona-Morta (**DBN**) da entrada **IN1**. Caso contrário, se os valores das entradas utilizadas forem diferentes, a saída **OUT** se mantém em falso. A definição das entradas utilizadas é feita através do parâmetro **N_IN**.

O bloco EQ é indicado quando deseja-se comparar variáveis em termos de igualdade. O parâmetro **DBN** fornece ao usuário uma ferramenta para determinar o ajuste de quão próximas precisam ser cada uma dessas medidas para que estas sejam consideradas iguais.

Parâmetro DBN e Operação

No caso de usar somente 2 entradas (**IN1** e **IN2**) a função comporta-se como igual-com-zona morta, portanto tornando **OUT** Verdadeiro se $ABS(IN1 - IN2) \leq DBN$.

Por exemplo: tem-se 3 entradas e o parâmetro **DBN** foi configurado com o valor default 10.

N_IN = 3

IN1 = 12, **IN2** = 21 e **IN3** = 5. Ou seja:

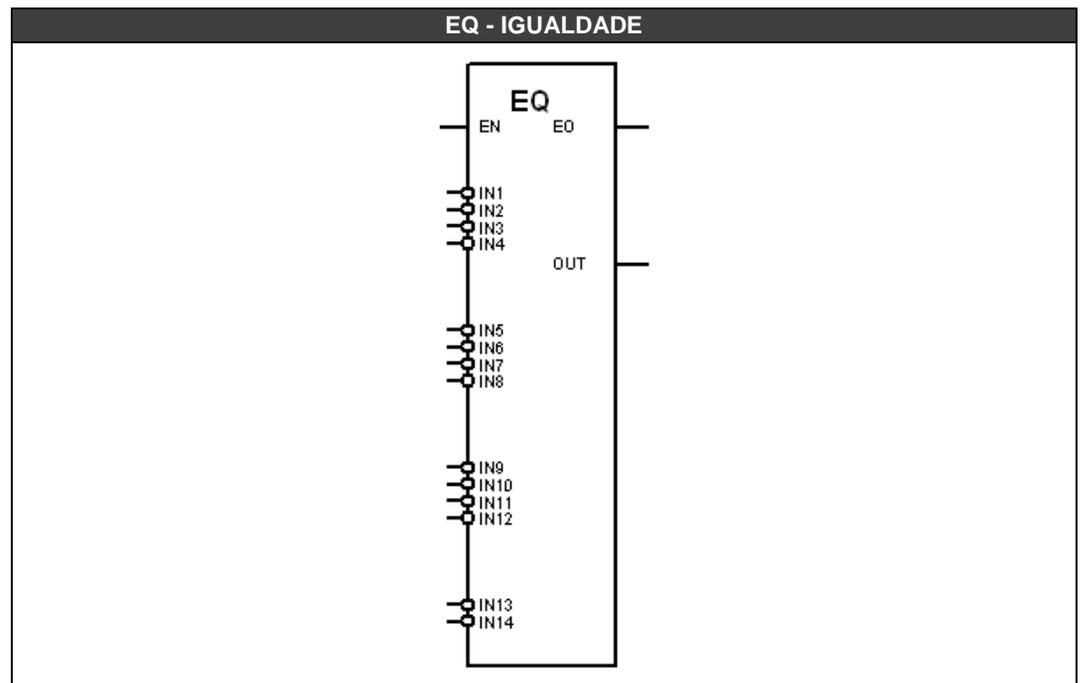
$ABS(IN1 - IN2) = 9 < 10$

$ABS(IN1 - IN3) = 7 < 10$

Portanto, como **DBN** = 10, a saída **OUT** será igual a “verdadeiro”.

O número mínimo de entradas utilizadas é 2 e o número máximo é 14.

Se a entrada **EN** está falsa, a saída é mantida em falso.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|---------------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| | IN3 | ENTRADA 3 | FLOAT |
| | IN4 | ENTRADA 4 | FLOAT |
| | IN5 | ENTRADA 5 | FLOAT |
| | IN6 | ENTRADA 6 | FLOAT |
| | IN7 | ENTRADA 7 | FLOAT |
| | IN8 | ENTRADA 8 | FLOAT |
| | IN9 | ENTRADA 9 | FLOAT |
| | IN10 | ENTRADA 10 | FLOAT |
| | IN11 | ENTRADA 11 | FLOAT |
| | IN12 | ENTRADA 12 | FLOAT |
| | IN13 | ENTRADA 13 | FLOAT |
| IN14 | ENTRADA 14 | FLOAT | |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO LÓGICO DA COMPARAÇÃO | BOOL |
| P | N_IN | NÚMERO DE ENTRADAS A SEREM UTILIZADAS | LONG |

I: Entrada O: Saída P: Parâmetro Interno

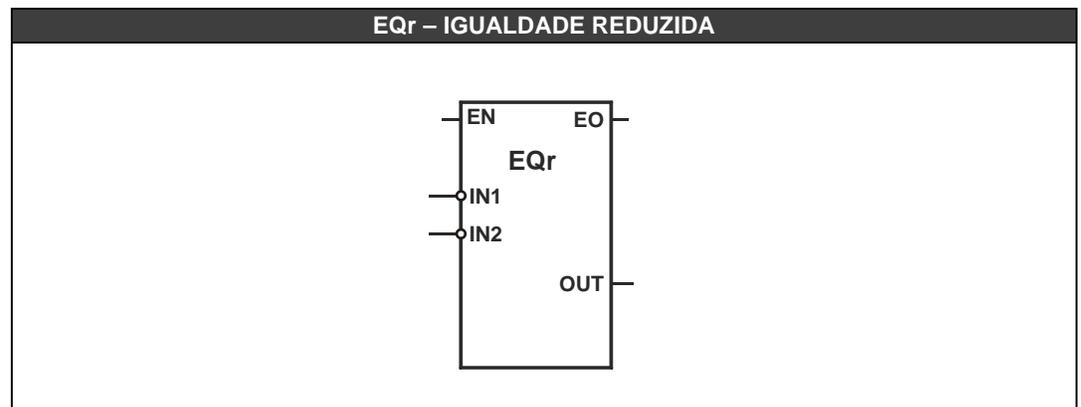
Igualdade Reduzida (EQr)

Descrição

Esta função, quando a entrada **EN** está verdadeira, mantém a saída **OUT** em verdadeiro se os valores das entradas não sofrerem desvio maior do que o valor da Zona-Morta (**DBN**) da entrada **IN1**. Caso contrário, se os valores das entradas utilizadas forem diferentes, a saída **OUT** se mantém em falso.

O bloco EQ é indicado quando deseja-se comparar variáveis em termos de igualdade. O parâmetro **DBN** fornece ao usuário uma ferramenta para determinar o ajuste de quão próximas precisam ser cada uma dessas medidas para que estas sejam consideradas iguais.

Se a entrada **EN** está falsa, a saída é mantida em falso.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|----------|------|--------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO LÓGICO DA COMPARAÇÃO | BOOL |

I: Entrada O: Saída P: Parâmetro Interno

Sequência Decrescente (GT)

Descrição

Esta função, quando a entrada **EN** está verdadeira, indicará verdadeiro na saída **OUT** se as entradas utilizadas (**IN1** a **INn**) estiverem em ordem decrescente. Isto é:

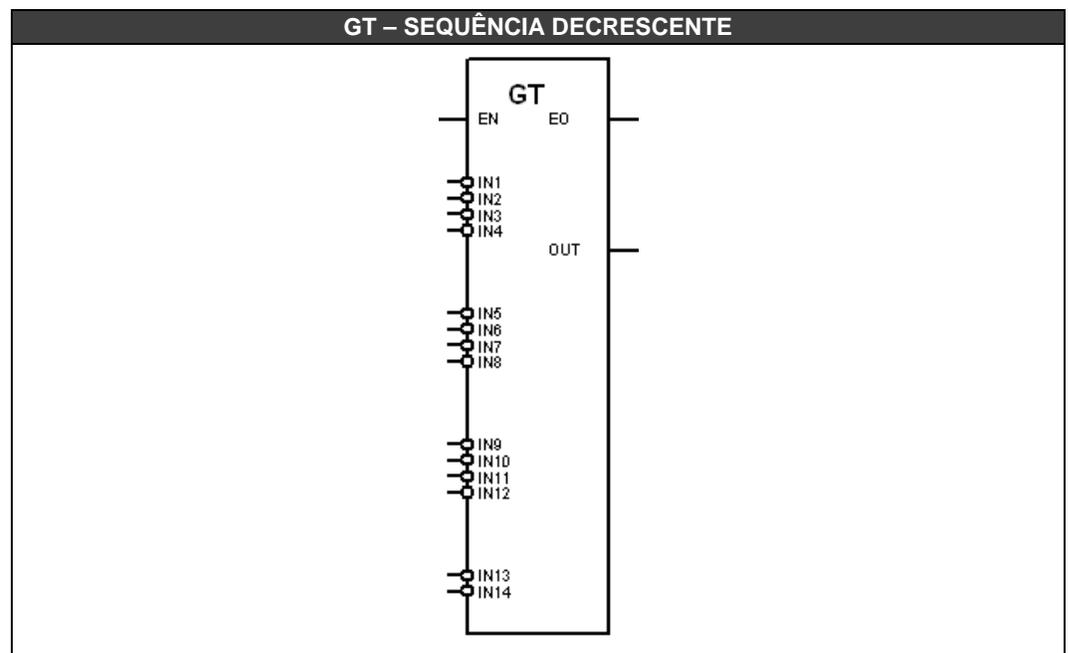
$$IN1 > IN2 > IN3 > IN4 \dots \dots \dots INn-1 > INn.$$

O número de entradas utilizadas é configurado pelo parâmetro **N_IN**. O número mínimo de entradas utilizadas é 2 e o número máximo é 14.

No caso de usar somente 2 entradas (**IN1** e **IN2**), a função comporta-se como uma comparação de maior que, tornando **OUT** verdadeiro se **IN1 > IN2**.

Se a entrada **EN** está falsa, a saída é mantida em falso.

Pode-se usar esta equação para implementar blocos condicionais que comparam duas entradas e tomam uma decisão (o estado da saída muda para 1 e habilita um outro bloco).



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|---------------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| | IN3 | ENTRADA 3 | FLOAT |
| | IN4 | ENTRADA 4 | FLOAT |
| | IN5 | ENTRADA 5 | FLOAT |
| | IN6 | ENTRADA 6 | FLOAT |
| | IN7 | ENTRADA 7 | FLOAT |
| | IN8 | ENTRADA 8 | FLOAT |
| | IN9 | ENTRADA 9 | FLOAT |
| | IN10 | ENTRADA 10 | FLOAT |
| | IN11 | ENTRADA 11 | FLOAT |
| | IN12 | ENTRADA 12 | FLOAT |
| | IN13 | ENTRADA 13 | FLOAT |
| IN14 | ENTRADA 14 | FLOAT | |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO LÓGICO DA COMPARAÇÃO | BOOL |
| P | N_IN | NÚMERO DE ENTRADAS A SEREM UTILIZADAS | LONG |

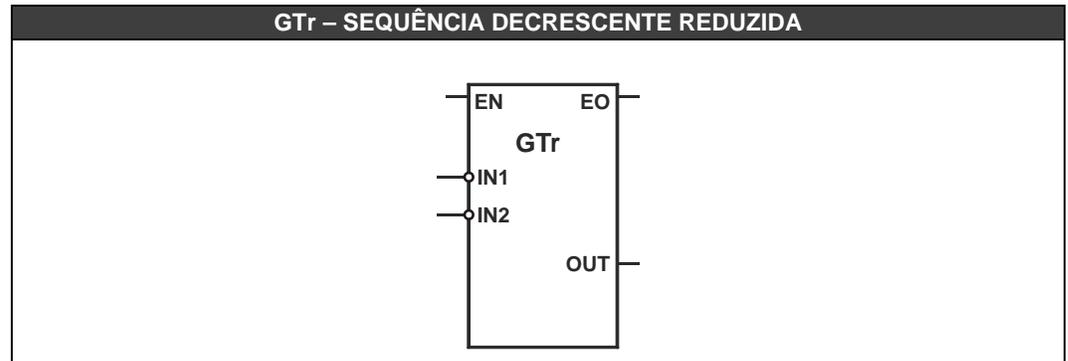
I: Entrada O: Saída P: Parâmetro Interno

Sequência Decrescente Reduzida (GTr)

Descrição

Esta função, quando a entrada **EN** está verdadeira, comporta-se como uma comparação de maior que, tornando **OUT** verdadeiro se **IN1 > IN2**.

Se a entrada **EN** está falsa, a saída é mantida em falso.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO LÓGICO DA COMPARAÇÃO | BOOL |

I: Entrada O: Saída P: Parâmetro Interno

Sequência Monotônica Decrescente (GTE)

Descrição

Esta função, quando a entrada **EN** está verdadeira, indicará verdadeiro na saída **OUT** se as entradas utilizadas (**IN1** a **INn**) estiverem em uma ordem monotônica decrescente. Uma sequência monotônica decrescente é definida como uma sequência de números em que dois elementos adjacentes estão relacionados por $INn-1 \geq INn$. Ou seja:

IN1, IN2, IN3, ..., INn-2, INn-1, INn

Onde:

IN1 \geq IN2

IN2 \geq IN3

...

INn-2 \geq INn-1

INn-1 \geq INn

O número de entradas utilizadas é configurado pelo parâmetro **N_IN**. O número mínimo de entradas utilizadas é 2 e o número máximo é 14.

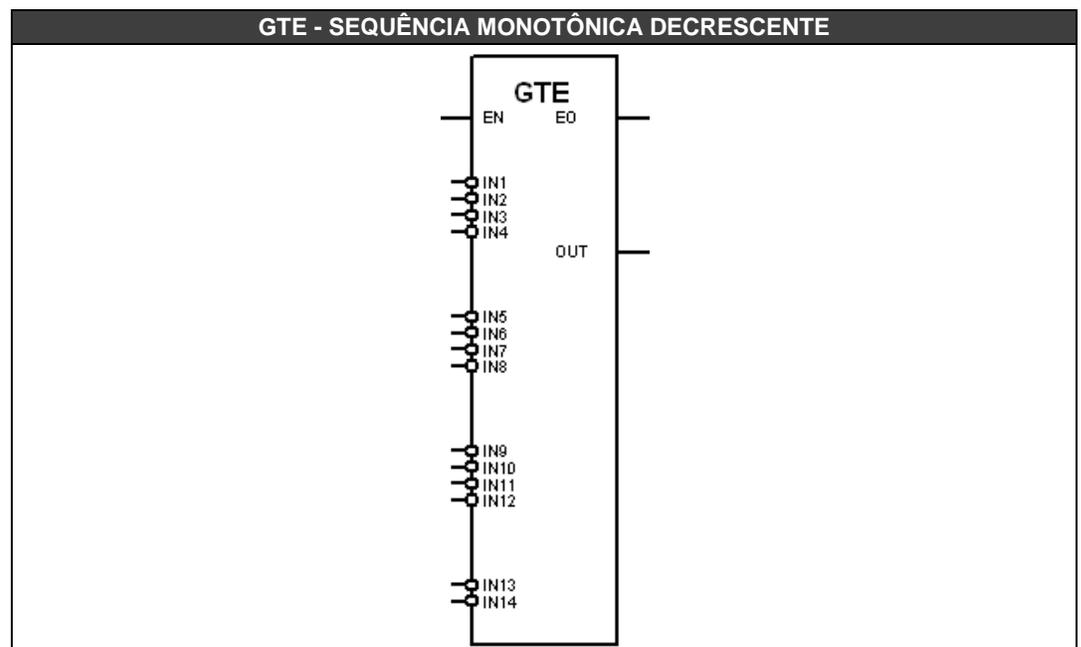
Operação

Uma sequência monotônica decrescente pode ser exemplificada como: **12,8,5,3,1**.

Caso somente 2 entradas (**IN1** e **IN2**) forem usadas, a função comporta-se como uma comparação de maior ou igual, tornando **OUT** verdadeiro se **IN1 \geq IN2**.

Se a entrada **EN** está falsa, a saída é mantida em falso.

Pode-se usar esta equação para implementar blocos condicionais que comparam duas entradas e tomam uma decisão (o estado da saída muda para 1 e habilita um outro bloco).



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|---------------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| | IN3 | ENTRADA 3 | FLOAT |
| | IN4 | ENTRADA 4 | FLOAT |
| | IN5 | ENTRADA 5 | FLOAT |
| | IN6 | ENTRADA 6 | FLOAT |
| | IN7 | ENTRADA 7 | FLOAT |
| | IN8 | ENTRADA 8 | FLOAT |
| | IN9 | ENTRADA 9 | FLOAT |
| | IN10 | ENTRADA 10 | FLOAT |
| | IN11 | ENTRADA 11 | FLOAT |
| | IN12 | ENTRADA 12 | FLOAT |
| | IN13 | ENTRADA 13 | FLOAT |
| IN14 | ENTRADA 14 | FLOAT | |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO LÓGICO DA COMPARAÇÃO | BOOL |
| P | N_IN | NÚMERO DE ENTRADAS A SEREM UTILIZADAS | LONG |

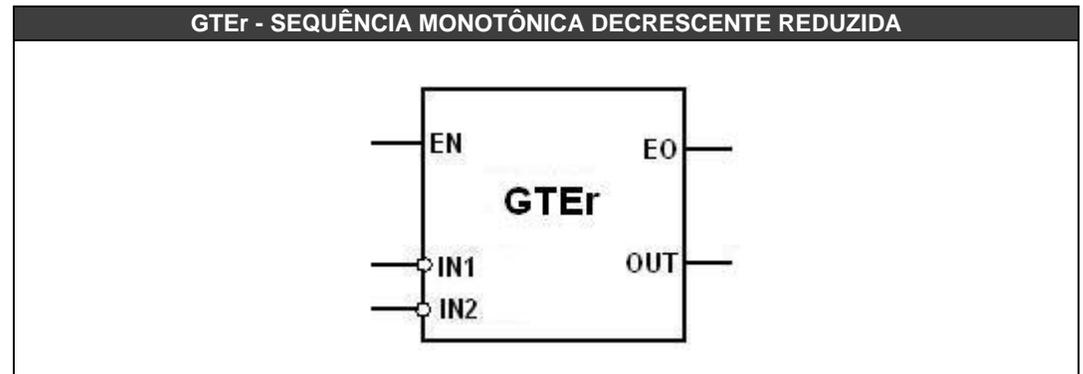
I: Entrada O: Saída P: Parâmetro Interno

Sequência Monotônica Decrescente Reduzida (GTer)

Descrição

Esta função, quando a entrada **EN** está verdadeira, comporta-se como uma comparação de maior ou igual, tornando **OUT** verdadeiro se **IN1** \geq **IN2**.

Pode-se usar esta equação para implementar blocos condicionais que comparam duas entradas e tomam uma decisão (o estado da saída muda para 1 e habilita um outro bloco).



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| O | E0 | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO LÓGICO DA COMPARAÇÃO | BOOL |

I: Entrada O: Saída P: Parâmetro Interno

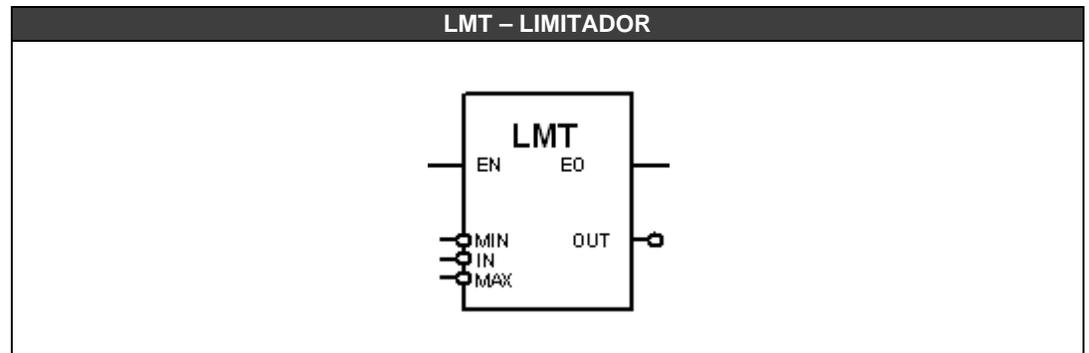
Limitador (LMT)

Descrição

Esta função, quando a entrada **EN** está verdadeira, limita a entrada **IN** entre o valor das entradas **MIN** e **MAX** e coloca o resultado na saída **OUT**.

Suponha que deseja-se limitar a entrada de um sinal entre 1 e 10. Neste caso, deve-se configurar **MIN** com o valor 1 e **MAX** com o valor 10 e o sinal a ser limitado é ligado na entrada **IN**. A saída será igual a 1 ou 10 quando os limites inferior ou superior, respectivamente, forem excedidos.

Se a entrada **EN** está falsa, a saída é mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|----------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | MIN | LIMITE MÍNIMO DO LIMITADOR | FLOAT |
| | IN | ENTRADA A SER LIMITADA | FLOAT |
| | MAX | LIMITE MÁXIMO DO LIMITADOR | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA DO BLOCO LIMITADA | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Sequência Crescente (LT)

Descrição

Esta função, quando a entrada **EN** está verdadeira, indicará verdadeiro na saída **OUT** se as entradas utilizadas (**IN1** a **INn**) estiverem em ordem crescente. Isto é:

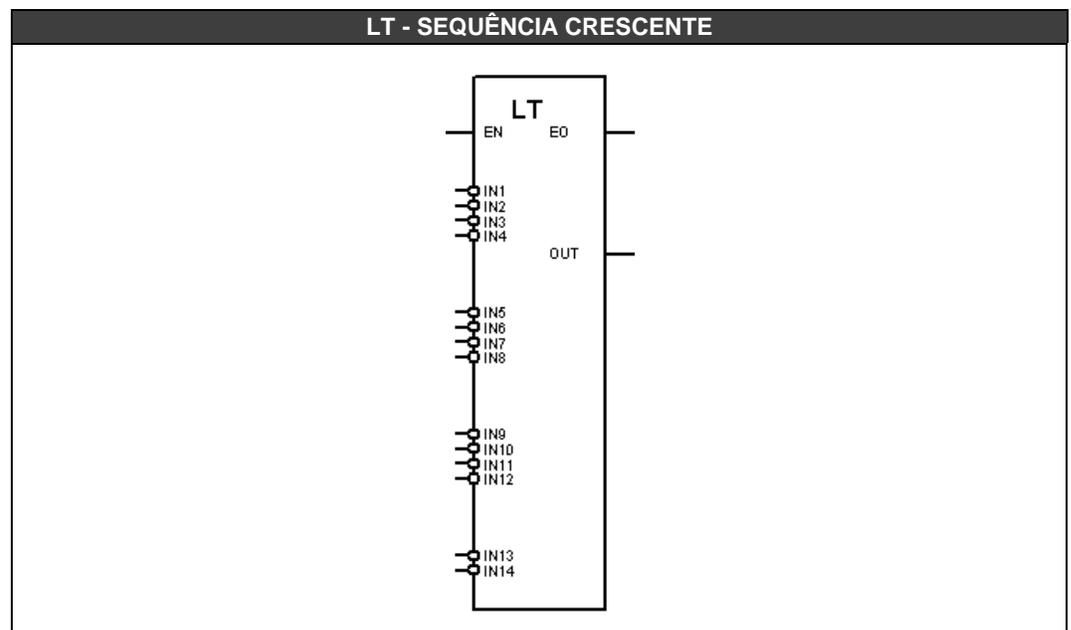
$$IN1 < IN2 < IN3 < IN4 \dots \dots \dots INn-1 < INn.$$

O número de entradas utilizadas é configurado pelo parâmetro **N_IN**. O número mínimo de entradas utilizadas é 2 e o número máximo é 14.

No caso de usar somente 2 entradas (**IN1** e **IN2**), a função comporta-se como uma comparação de menor que, tornando **OUT** verdadeiro se **IN1 < IN2**.

Se a entrada **EN** está falsa, a saída é mantida em falso.

Pode-se usar esta equação para implementar blocos condicionais que comparam duas entradas e tomam uma decisão (o estado da saída muda para 1 e habilita um outro bloco).



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|---------------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| | IN3 | ENTRADA 3 | FLOAT |
| | IN4 | ENTRADA 4 | FLOAT |
| | IN5 | ENTRADA 5 | FLOAT |
| | IN6 | ENTRADA 6 | FLOAT |
| | IN7 | ENTRADA 7 | FLOAT |
| | IN8 | ENTRADA 8 | FLOAT |
| | IN9 | ENTRADA 9 | FLOAT |
| | IN10 | ENTRADA 10 | FLOAT |
| | IN11 | ENTRADA 11 | FLOAT |
| | IN12 | ENTRADA 12 | FLOAT |
| | IN13 | ENTRADA 13 | FLOAT |
| IN14 | ENTRADA 14 | FLOAT | |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO LÓGICO DA COMPARAÇÃO | BOOL |
| P | N_IN | NÚMERO DE ENTRADAS A SEREM UTILIZADAS | LONG |

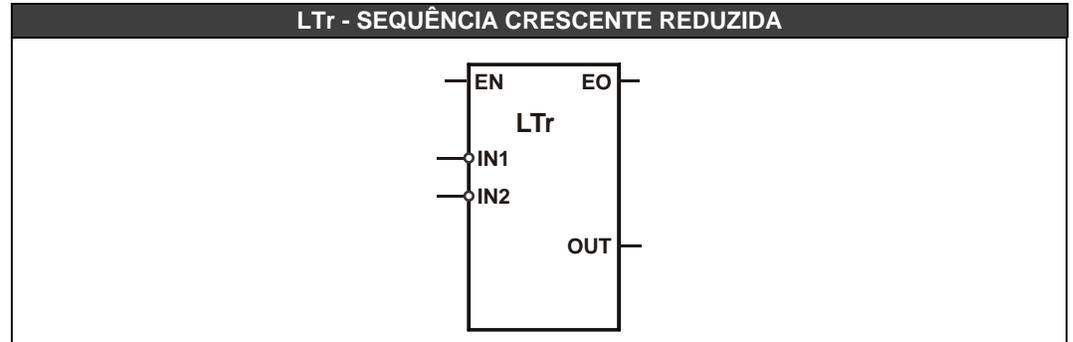
I: Entrada **O:** Saída **P:** Parâmetro Interno

Sequência Crescente Reduzido (LTr)

Descrição

Esta função, quando a entrada **EN** está verdadeira, comporta-se como uma comparação de menor que, tornando **OUT** verdadeiro se $IN1 < IN2$.

Se a entrada **EN** está falsa, a saída é mantida em falso.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO LÓGICO DA COMPARAÇÃO | BOOL |

I: Entrada O: Saída P: Parâmetro Interno

Sequência Monotônica Crescente (LTE)

Descrição

Esta função, quando a entrada **EN** está verdadeira, indicará verdadeiro na saída **OUT** se as entradas utilizadas (**IN1** a **INn**) estiverem em uma ordem monotônica crescente. Uma sequência monotônica crescente é definida como uma sequência de números em que dois elementos adjacentes estão relacionados por $INn-1 \leq INn$. Ou seja:

IN1, IN2, IN3, ..., INn-2, INn-1, INn

Onde:

IN1 \leq IN2

IN2 \leq IN3

...

INn-2 \leq INn-1

INn-1 \leq INn

O número de entradas utilizadas é configurado pelo parâmetro **N_IN**.

O número mínimo de entradas utilizadas é 2 e o número máximo é 14.

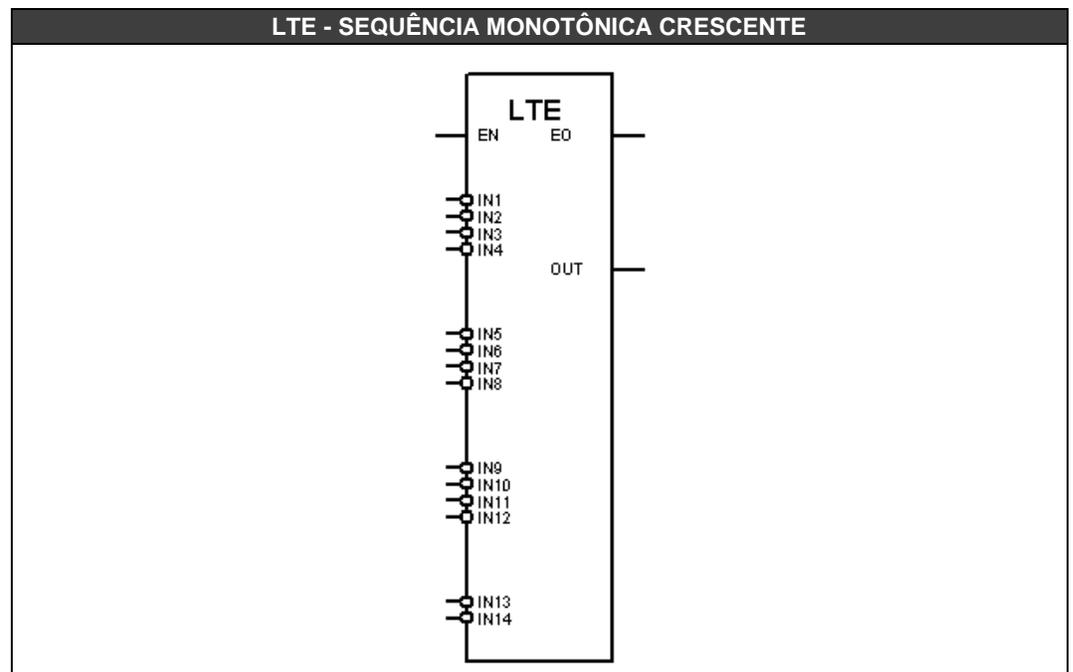
Operação

Uma sequência monotônica decrescente pode ser exemplificada como: **1,1,3,3,4,5,6,78,78**

No caso de usar somente 2 entradas (**IN1** e **IN2**), a função comporta-se como uma comparação de menor ou igual, tornando **OUT** verdadeiro se **IN1 \leq IN2**.

Se a entrada **EN** está falsa, a saída é mantida em falso.

Pode-se usar esta equação para implementar blocos condicionais que comparam duas entradas e tomam uma decisão (o estado da saída muda para 1 e habilita um outro bloco).



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|---------------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| | IN3 | ENTRADA 3 | FLOAT |
| | IN4 | ENTRADA 4 | FLOAT |
| | IN5 | ENTRADA 5 | FLOAT |
| | IN6 | ENTRADA 6 | FLOAT |
| | IN7 | ENTRADA 7 | FLOAT |
| | IN8 | ENTRADA 8 | FLOAT |
| | IN9 | ENTRADA 9 | FLOAT |
| | IN10 | ENTRADA 10 | FLOAT |
| | IN11 | ENTRADA 11 | FLOAT |
| | IN12 | ENTRADA 12 | FLOAT |
| | IN13 | ENTRADA 13 | FLOAT |
| IN14 | ENTRADA 14 | FLOAT | |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO LÓGICO DA COMPARAÇÃO | BOOL |
| P | N_IN | NÚMERO DE ENTRADAS A SEREM UTILIZADAS | LONG |

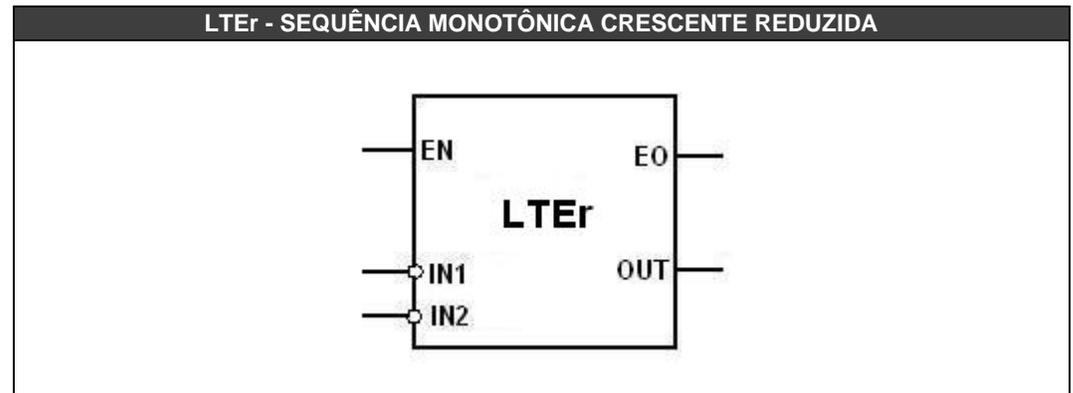
I: Entrada O: Saída P: Parâmetro Interno

Sequência Monotônica Crescente Reduzida (LTER)

Descrição

Esta função, quando a entrada **EN** está verdadeira, comporta-se como uma comparação de menor ou igual, tornando **OUT** verdadeiro se $IN1 \leq IN2$.

Pode-se usar esta equação para implementar blocos condicionais que comparam duas entradas e tomam uma decisão (o estado da saída muda para 1 e habilita um outro bloco).



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | RESULTADO LÓGICO DA COMPARAÇÃO | BOOL |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Máximo (MAX)

Descrição

Esta função, quando a entrada **EN** está verdadeira, seleciona o valor máximo entre as entradas utilizadas (**IN1** a **INn**) e coloca-o na saída **OUT**.

O número de entradas utilizadas é configurado pelo parâmetro **N_IN**. O número mínimo de entradas utilizadas é 2 e o número máximo é 14.

Operação

Suponha que esteja-se utilizando 4 entradas e as respectivas leituras sejam:

IN1 = 5,899

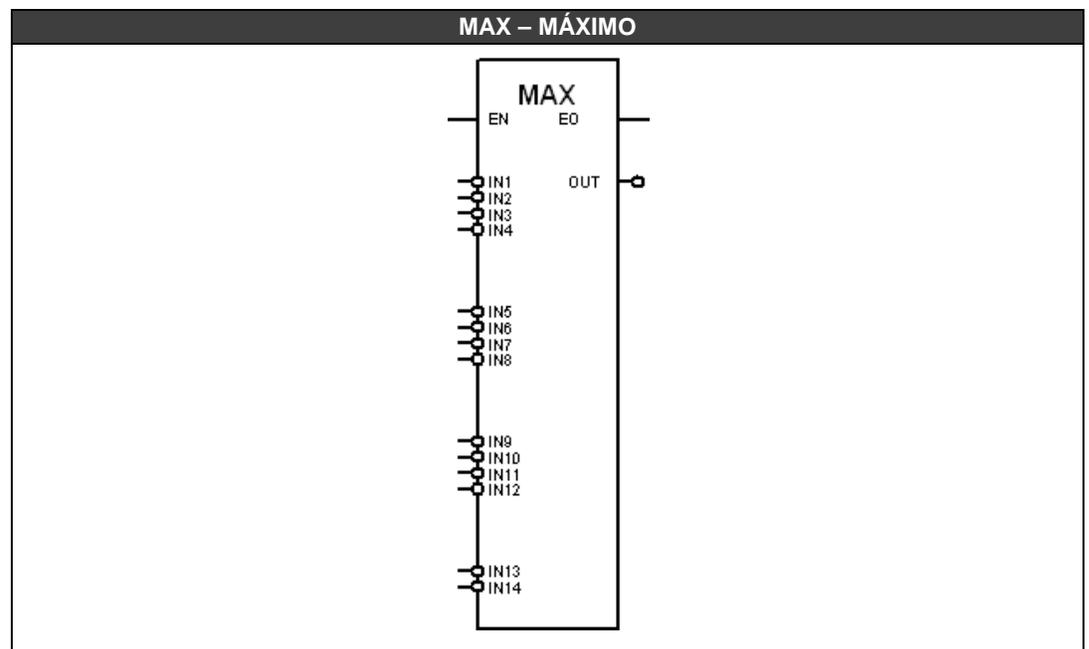
IN2 = 7,9000

IN3 = 10,899

IN4 = 23,90

A função MAX envia para saída o valor da saída **IN4**, ou seja, 23,90.

Se a entrada **EN** está falsa, a saída é mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|---------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA NÚMERO 1 | FLOAT |
| | IN2 | ENTRADA NÚMERO 2 | FLOAT |
| | IN3 | ENTRADA NÚMERO 3 | FLOAT |
| | ... | ... | |
| | ... | ... | |
| | IN13 | ENTRADA NÚMERO 13 | FLOAT |
| O | IN14 | ENTRADA NÚMERO 14 | FLOAT |
| | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | MÁXIMO VALOR DAS ENTRADAS | FLOAT |

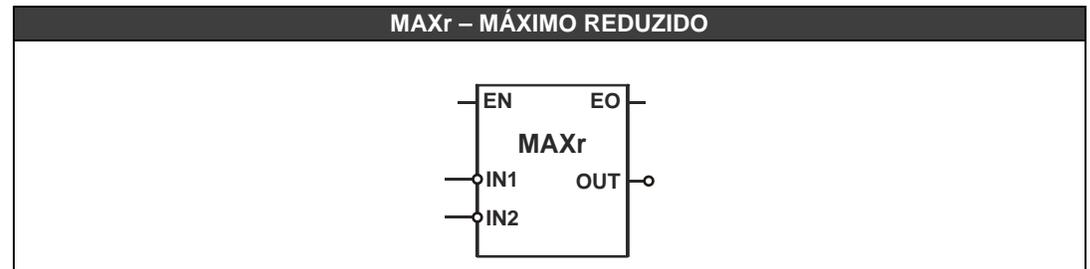
I: Entrada **O:** Saída **P:** Parâmetro Interno

Máximo Reduzido (MAXr)

Descrição

Esta função, quando a entrada **EN** está verdadeira, seleciona o valor máximo entre as entradas coloca-o na saída **OUT**.

Se a entrada **EN** está falsa, a saída é mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|---------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA NÚMERO 1 | FLOAT |
| | IN2 | ENTRADA NÚMERO 2 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | MÁXIMO VALOR DAS ENTRADAS | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Mínimo (MIN)

Descrição

Esta função, quando a entrada **EN** está verdadeira, seleciona o valor mínimo entre as entradas utilizadas (**IN1** a **INn**) e coloca-o na saída **OUT**.

O número de entradas utilizadas é configurado pelo parâmetro **N_IN**. O número mínimo de entradas utilizadas é 2 e o número máximo é 14.

Operação

Suponha que esteja-se utilizando 4 entradas e as respectivas leituras sejam:

IN1 = 5,899

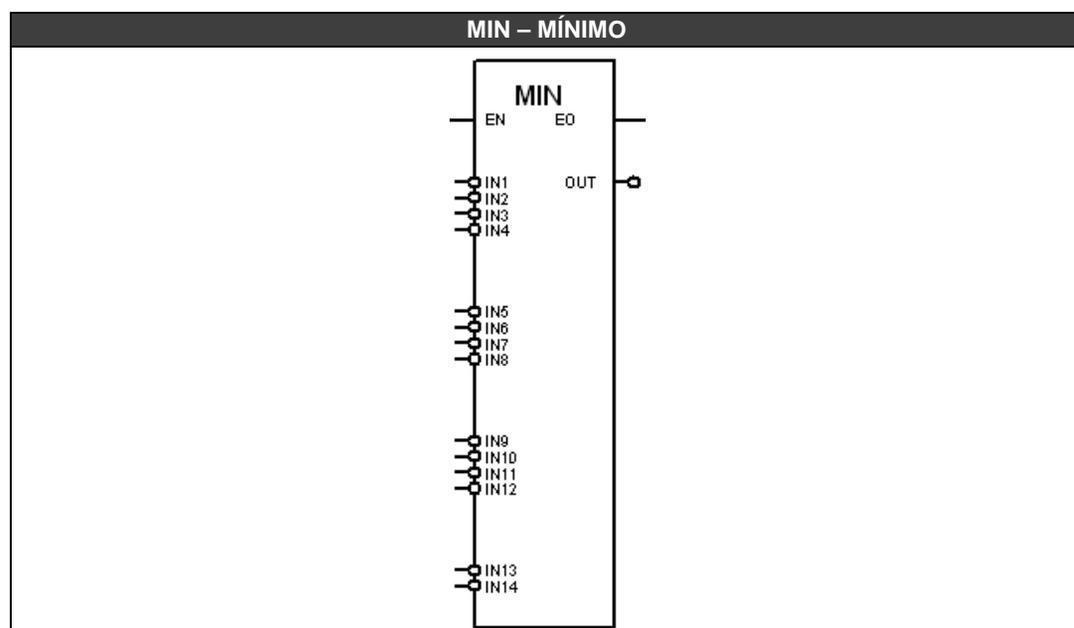
IN2 = 7,9000

IN3 = 10,899

IN4 = 23,90

A função MIN envia para saída o valor da saída **IN1**, ou seja, 5,899.

Se a entrada **EN** está falsa, a saída é mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------------|---------------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| | IN3 | ENTRADA 3 | FLOAT |
| | IN4 | ENTRADA 4 | FLOAT |
| | IN5 | ENTRADA 5 | FLOAT |
| | IN6 | ENTRADA 6 | FLOAT |
| | IN7 | ENTRADA 7 | FLOAT |
| | IN8 | ENTRADA 8 | FLOAT |
| | IN9 | ENTRADA 9 | FLOAT |
| | IN10 | ENTRADA 10 | FLOAT |
| | IN11 | ENTRADA 11 | FLOAT |
| | IN12 | ENTRADA 12 | FLOAT |
| | IN13 | ENTRADA 13 | FLOAT |
| IN14 | ENTRADA 14 | FLOAT | |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | MÍNIMO VALOR DAS ENTRADAS | FLOAT |
| P | N_IN | NÚMERO DE ENTRADAS A SEREM UTILIZADAS | LONG |

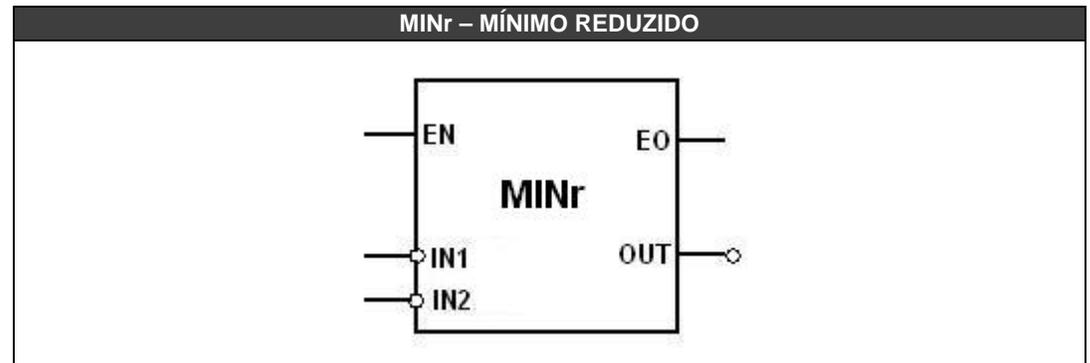
I: Entrada O: Saída P: Parâmetro Interno

Mínimo Reduzido (MINr)

Descrição

Esta função, quando a entrada **EN** está verdadeira, seleciona o valor mínimo entre as entradas e coloca-o na saída **OUT**.

Se a entrada **EN** está falsa, a saída é mantida em zero.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|---------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN1 | ENTRADA 1 | FLOAT |
| | IN2 | ENTRADA 2 | FLOAT |
| O | E0 | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | MÍNIMO VALOR DAS ENTRADAS | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Funções de Controle de Processos

Controlador PID Avançado (APID)

Descrição:

Este bloco funcional, quando a entrada **EN** está verdadeira realiza o controle PID. O consagrado algoritmo PID para controle de processos contínuos, associado à flexibilidade de configuração de características de operação através de parametrização, permite a utilização deste bloco à uma grande variedade de aplicações e estratégias de controle.

Este bloco oferece várias opções de configuração do algoritmo de controle tendo como base os termos Proporcional (P), Integral (I) e Derivativo (D), que podem ser aplicados ao erro ou apenas à variável de processo (PV). O bloco também oferece três saídas referentes a alarme, sendo uma saída para alarme de desvio e duas saídas configuráveis.

Este bloco permite a seleção dos seguintes tipos de controle: PI-Sampling, Erro Quadrático, GAP e Ganho Adaptativo.

Os limites de anti-reset windup (limites aplicados apenas ao termo integral) podem ser configurados pelo usuário. Além disto, tem-se opção de seleção entre: algoritmo ISA ou paralelo, ação direta ou reversa, transferência de controle de manual para automático, bumpless ou hard.

Tipo de PID

É definido pelo parâmetro **PID**.

PID = 0: tipo PI.D.

PID = 1: tipo PID.

PID = 2: tipo I.PD.

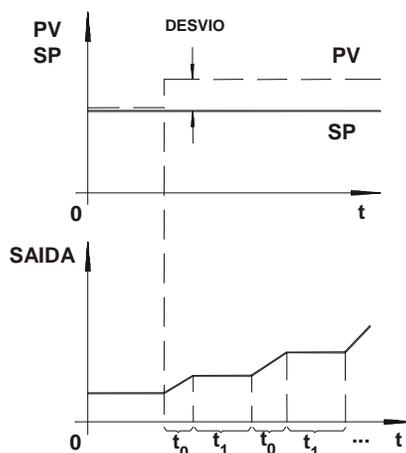
PID = 3: tipo PI-SAMPLING.

PI.D : As ações P e I atuam sobre o erro e a ação D sobre a variável de processo. Desta forma, o sinal de saída acompanha as mudanças de set point segundo as ações proporcional e integral, mas não dá uma variação indesejável devido à ação derivativa. É o mais recomendado para a maioria das aplicações com o set point ajustável pelo operador.

PID: As ações P, I e D atuam sobre o erro. Desta forma o sinal de saída é alterado quando há mudanças na variável de processo ou no set point. É recomendado para controle de relação ou para controle escravo de um cascata.

I.PD: Neste tipo somente a integral atua sobre o erro. Mudanças no set point provocam a variação no sinal de saída de maneira suave. É recomendado para processos que não podem ter variações bruscas na MV (variável manipulada) em função da mudança no set point. É o caso de processos de aquecimento com ganho muito alto.

PI-SAMPLING: Neste tipo, quando há um desvio, o sinal de saída muda de acordo com o algoritmo PI durante um tempo t_0 , ajustável através do parâmetro **ATRIS** (em segundos) e permanecendo constante durante um tempo t_1 , onde o período total deste ciclo (t_0+t_1) é ajustável através do parâmetro **PPIS** (em segundos). Se o desvio persistir, o sinal de saída continuará variando novamente durante o tempo t_0 e permanecendo constante durante o tempo t_1 . Este tipo de controle é recomendado para processos com alto tempo morto.



$t_0 + t_1$ = Período de Amostra (PPIS) – em segundos

t_0 = Tempo de Atuação (ATRIS) – em segundos

PI Sampling

Tipo de Algoritmo (ALG)

O tipo de algoritmo é definido pelo parâmetro **ALG**.

ALG = falso: Algoritmo Paralelo ou Ideal.

ALG = verdadeiro: Algoritmo ISA ou Não-Interativo.

$$\text{PARALELO} : MV(t) = K_p e(t) + \frac{1}{T_R} \int e(t) dt + T_D \frac{de(t)}{dt}$$

$$\text{ISA} : MV(t) = K_p \left[e(t) + \frac{1}{T_R} \int e(t) dt + T_D \frac{de(t)}{dt} \right]$$

Tipo de Ação (ACT)

Existem processos que requerem que o sinal de saída (variável manipulada – MV) aumente quando a variável de processo (PV) aumenta, enquanto que outros requerem o contrário. A escolha do tipo de ação é feita através do parâmetro **ACT**.

| PARÂMETRO | TIPO DE AÇÃO | ERRO (em porcentagem) | EFEITO |
|-------------------------|--------------|--------------------------|---------------------------------|
| ACT = falso | Reversa | $e = SP - PV$ | Saída diminui com aumento da PV |
| ACT = verdadeiro | Direta | $e = PV - SP$ | Saída aumenta com aumento da PV |

Tipo de Erro – Linear/Quadrático (TYERR)

No desvio ou erro linear (**TYERR** = 0), o erro (em porcentagem) considerado nos cálculos será:

Ação Reversa: $e = SP - PV$ "Saída diminui com o aumento da PV"

Ação Direta: $e = PV - SP$ "Saída aumenta com o aumento da PV"

Existem processos em que o desvio em relação ao set point é preferível aos distúrbios causados pelo controlador no processo. Portanto, a atuação do controle deve ser pequena para pequenos distúrbios e aumentar gradualmente com o aumento do desvio. Um exemplo típico deste tipo de processo é o controle de nível de um tanque em que o set point não é tão importante quanto a estabilidade da vazão de descarga. Este tipo de processo pode ser controlado com ganho adaptativo, controle com gap ou erro quadrático.

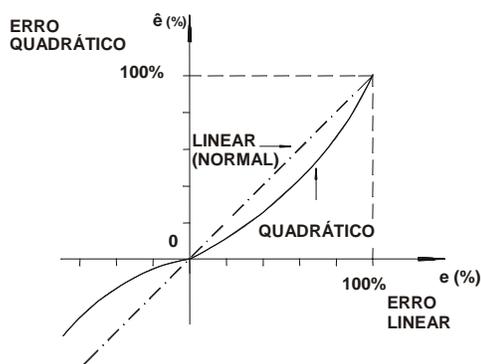
No erro quadrático ($TYERR = 1$), o erro considerado nos cálculos será:

Ação Reversa: $e = SP - PV$ "Saída diminui com o aumento da PV"

Ação Direta: $e = PV - SP$ "Saída aumenta com o aumento da PV"

$$\hat{e} = \frac{e. / e |}{100}$$

\hat{e} : Erro (em porcentagem) a ser considerado nos cálculos do APID.



Erro Quadrático x Erro Linear

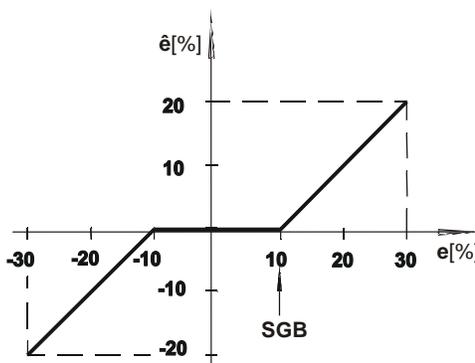
Controle Tipo GAP (SGB e SGGAP)

Existem aplicações em que o controle é instável dentro de um valor, em torno do set point, devido à faixa morta do atuador, ruído ou outras razões. Neste caso é aconselhável que o controlador tenha uma ação diferenciada nesta faixa.

O controle tipo GAP ou GAP com ganho adaptativo pode ser utilizado para resolver este problema.

EXEMPLO:

Considerar erro (\hat{e}), em porcentagem, para um controle tipo GAP com uma banda igual a $\pm 10\%$ ($SGB=10$) e ganho diferenciado igual a zero ($SGGAP=0$).

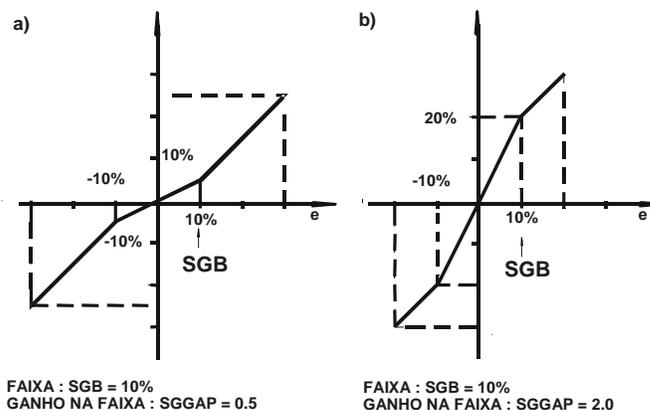


Controle de GAP com Ganho Especial = 0

Alguns processos podem necessitar de um ganho especial dentro do GAP. Em tais casos, é possível selecionar um fator para o parâmetro $SGGAP$ o qual multiplica o erro, fazendo com que o erro a ser considerado no cálculo do APID seja:

$$\hat{e} = e . SGGAP$$

Dentro do GAP, a ação de controle será mais rápida quando $SGGAP > 1$ e mais lenta quando $SGGAP < 1$. Para $SGGAP = 0$ (banda nula) o controle de tipo GAP não é ativado.



Controle de GAP com Ganho Especial (a) Ganho < 1, (b) Ganho > 1

Controle com Ganho Adaptativo (INVAG, ADAPG, Coordenadas X/Y)

O ganho adaptativo faz com que as constantes do PID sejam modificadas por um fator **G**. Este fator **G** é estabelecido em uma curva de 10 pontos (x,y) em função do tipo de variável definido através do parâmetro **INVAG**, onde os valores intermediários da curva serão calculados usando o método de interpolação linear. Os tipos de variáveis possíveis são:

- INVAG = 0 : SP** (set point)
- INVAG = 1 : PV** (variável de processo)
- INVAG = 2 : DEV** (desvio ou erro)
- INVAG = 3 : OUT** (sinal de saída)
- INVAG = 4 : EXT** (variável externa).

Os pontos da curva do ganho adaptativo no eixo das abscissas (X) são dados de acordo com a variável selecionada e no eixo das ordenadas (Y) pelo ganho **G**. O ganho modifica as constantes de sintonia **KP**, **TR** e **TD** para:

$$K_P' = G \cdot K_P$$

$$T_R' = \frac{T_R}{G}$$

$$T_D' = G \cdot T_D$$

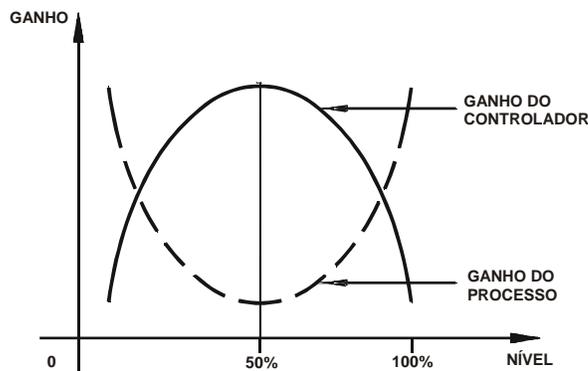
O Ganho **G** pode afetar as ações **PID**, **PI**, **P**, **I** ou **D**. A seleção da ação do ganho adaptativo é feita pelo parâmetro **ADAPG** o qual também pode inibir o ganho adaptativo quando **ADAPG=0** (not used).

- ADAPG = 0 : not used**
- ADAPG = 1 : PID**
- ADAPG = 2 : PI**
- ADAPG = 3 : P**
- ADAPG = 4 : I**
- ADAPG = 5 : D**

O ganho adaptativo é recomendado para controles excessivamente não lineares. Um exemplo clássico de ganho adaptativo é o controle de nível de uma tubulação de caldeira.

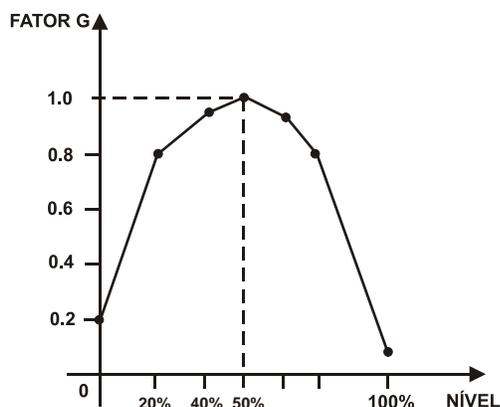
A variação de volume não é linear com a variação de nível. A linha pontilhada da figura abaixo mostra a variação de volume com o nível.

Note que o nível varia lentamente em torno de 50% e varia rapidamente nos extremos. As ações de controle devem ter um ganho inverso ao ganho do processo. Isto é mostrado pela linha contínua da figura abaixo.



Ganho do Processo x Ganho do Controlador

O ganho adaptativo pode ser configurado como mostra a figura a seguir. Esta curva pode se representada pelos seguintes pontos da curva:



X1 = 0 / Y1 = 0,2
 X2 = 20 / Y2 = 0,8
 X3 = 40 / Y3 = 0,96
 .
 .
 .

Curva de Ganho em Função de PV

Observe o seguinte:

- Os pares (x,y) devem ser inseridos em ordem crescente dos valores de x, iniciando-se no par (x1,y1) e sem saltar índices.
- Não é necessário utilizar todos os 10 pontos fornecidos pelo bloco para a geração da curva, porém será necessário repetir os valores de X e Y do último ponto da curva desejada nos demais pontos não utilizados. Por exemplo, a curva desejada utilizará de x1,y1 até os pares x5,y5, então os demais pares deverão ser configurados com os mesmos valores de x5,y5.
- Os valores inseridos para a coordenada X devem ser da mesma grandeza (valores em porcentagem ou unidade de engenharia) do valor definido para a variável selecionada em **INVAG** e devem respeitar os limites inferior e superior definidos em **SPLL** e **SPLH** quando utilizado as variáveis **SP**, **PV** ou **EXT**.

INVAG = 0 : SP (valores em unidade de engenharia)
INVAG = 1 : PV (valores em unidade de engenharia)
INVAG = 2 : DEV (valores em porcentagem)
INVAG = 3 : OUT (valores em porcentagem)
INVAG = 4 : EXT (valores em unidade de engenharia)

Os valores inseridos para a coordenada Y (fator G), devem estar em porcentagem. Por exemplo, se y2=0,5, então um KP=10 seria multiplicado por 0,5 resultando em KP'=5 e se y2=2, então um KP=10 seria multiplicado por 2 resultando em KP'=20.

- É recomendável programar a variável até 102% do seu valor, desde que a variável possa estar acima de 100%.

5. Sintonia normalmente é feita para $G = 1$. No exemplo, o controle torna-se mais lento acima ou abaixo de 50% do nível.
6. Ganho adaptativo também é muito utilizado em controle de pH.

Anti-Reset Windup pelo Termo Integral (AWL e AWH)

Usualmente o algoritmo de controle pára automaticamente a contribuição do termo integral, quando o sinal de saída atinge os limites inferior ou superior configurados através dos parâmetros **AWL** e **AWH**. As contribuições dos termos proporcional e derivativo não são afetadas. Uma característica diferenciadora do algoritmo deste bloco é a possibilidade de configuração desses limites. Quando o limite **AWH** é maior que o limite **OUTH**, a saída do bloco **OUT** é travada no valor **OUTH**, mas internamente, o algoritmo continua o cálculo integral até o limite **AWH**. O usuário pode evitar este caso configurando o limite **AWH** a um valor menor ou igual a **OUTH**, assim obtém-se respostas mais rápidas evitando-se assim, por exemplo, um overshoot em processos de aquecimento. A mesma idéia se aplica aos limites baixos (**AWL** e **OUTL**).

Limites para a saída OUT (CLIM, OUTL e OUTH)

Os limites para a saída **OUT** são definidos nos parâmetros **OUTL** e **OUTH**.

Os valores aplicados nestes dois parâmetros devem estar entre **-2%** e **102%**.

O parâmetro **CLIM** define em quais modos (automático/manual) serão aplicados os limites **OUTL** e **OUTH**.

CLIM = 0 : AUTO/MAN (em ambos os modos)
CLIM = 1 : AUTO (somente no automático)

Em qualquer modo de operação que seja possível realizar escrita na saída **OUT**, se o valor escrito estiver fora dos limites configurados, a saída se manterá no valor anterior.

Alarme de Desvio (DEVAL, MTDA, ALM)

O alarme pode ser programado para qual desvio (em porcentagem) se deseja um alarme (**DEVAL**) e quanto tempo (em segundos) este desvio pode ocorrer sem ativar o alarme (**MTDA**). Por exemplo, se **DEVAL=5** e **MTDA=30**, então a saída **ALM** será ativada (nível lógico 1) se um desvio de 5% persistir por um tempo maior que 30 segundos.

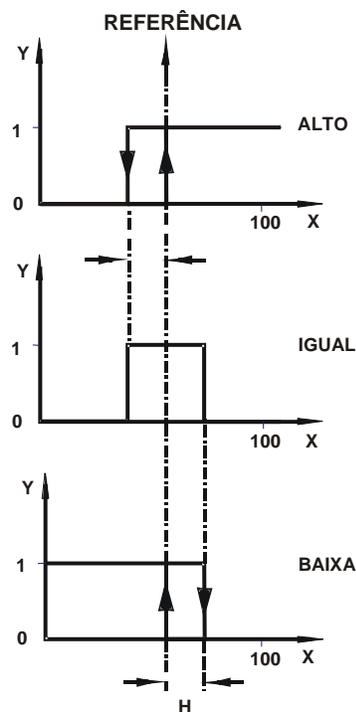
Para **MTDA=0** (tempo infinito) o alarme de desvio não é ativado (sem alarme).

Alarme Configurável (ALM1, INAL1, TYPE1, DBN1, REF1, ALM2, INAL2, TYPE2, DBN2, REF2)

Os alarmes configuráveis, independentes, são **ALM1** e **ALM2**. Eles são ativados através de seus respectivos parâmetros, comparando o valor de referência **REFx** e a variável selecionada em **INALx** (SP ou MV), sendo possível selecionar o tipo de comparação **TYPEx**, checando assim se **INALx** está acima (High), abaixo (Low) ou igual (Equal) a **REFx**.

Para evitar a oscilação do sinal de saída quando a variável está muito próxima da referência, pode ser ajustado um valor de histerese através do parâmetro **DBNx**.

Os alarmes trabalham de acordo com a figura a seguir:



Ação do Alarme com Histerese

Constantes do PID (parâmetros KP, TR, TD, BIAS e entrada FB)

KP – Ganho Proporcional.

TR – Tempo da Integral em minutos/repetição, portanto, quanto maior este parâmetro menor é a ação integral. Pode ser interpretado como sendo o tempo necessário para a saída ser incrementada/decrementada do valor do erro (no PID paralelo), mantendo-se o mesmo constante.

TD – Tempo Derivativo em minutos. O termo derivativo é calculado usando uma pseudoderivada, isto é, uma ação semelhante a um *lead/lag*, na qual a constante de *lag* é $\alpha \cdot TD$. Na implementação deste bloco o fator α é igual a 0,13.

BIAS – Neste parâmetro é possível ajustar o valor inicial da saída quando o controle é transferido de manual para automático. O valor aplicado neste parâmetro deve estar entre **0%** e **100%**. A utilização desta entrada pode ser feita através da seleção do parâmetro **TRS**.

FB – Através desta entrada é possível ajustar o valor inicial da saída quando o controle é transferido para manual. O valor aplicado nesta entrada deve estar entre **0%** e **100%**. A utilização desta entrada pode ser feita através da seleção do parâmetro **TRS**.

Entrada A/M (Automático/Manual)

Se **A/M** for verdadeiro, o APID estará em controle automático e se **A/M** for falso o APID estará em controle manual.

Tipos de transferência de Manual para Automático (TRS)

O valor da saída do bloco APID é definido através do parâmetro **TRS**.

TRS = 0 (Bumpless) :

Em modo manual, o valor da saída do bloco é igual ao último valor da saída em modo automático. Neste caso pode se escrever na saída **OUT**.

No chaveamento do modo manual para o automático, o bloco inicia os cálculos partindo do último valor da saída em modo manual.

TRS = 1 (Bumpless + BIAS) :

Em modo manual, o valor da saída do bloco é igual ao último valor da saída em modo automático. Neste caso pode se escrever na saída **OUT**.

No chaveamento do modo manual para o automático, o bloco inicia os cálculos partindo do valor do parâmetro **BIAS**.

TRS = 2 (Bumpless + FB) :

Em modo manual, o valor da saída do bloco é igual ao valor inserido na entrada **FB**. Neste caso não pode se escrever na saída **OUT**.

No chaveamento do modo manual para o automático, o bloco inicia os cálculos partindo do valor da entrada **FB**.

TRS = 3 (Hard) :

Em modo manual, o valor da saída do bloco é igual ao último valor da saída em modo automático. Neste caso pode se escrever na saída **OUT**.

No chaveamento do modo manual para o automático, o bloco inicia os cálculos partindo do último valor da saída em modo manual + o termo proporcional (**KP x erro**).

TRS = 4 (Hard + BIAS) :

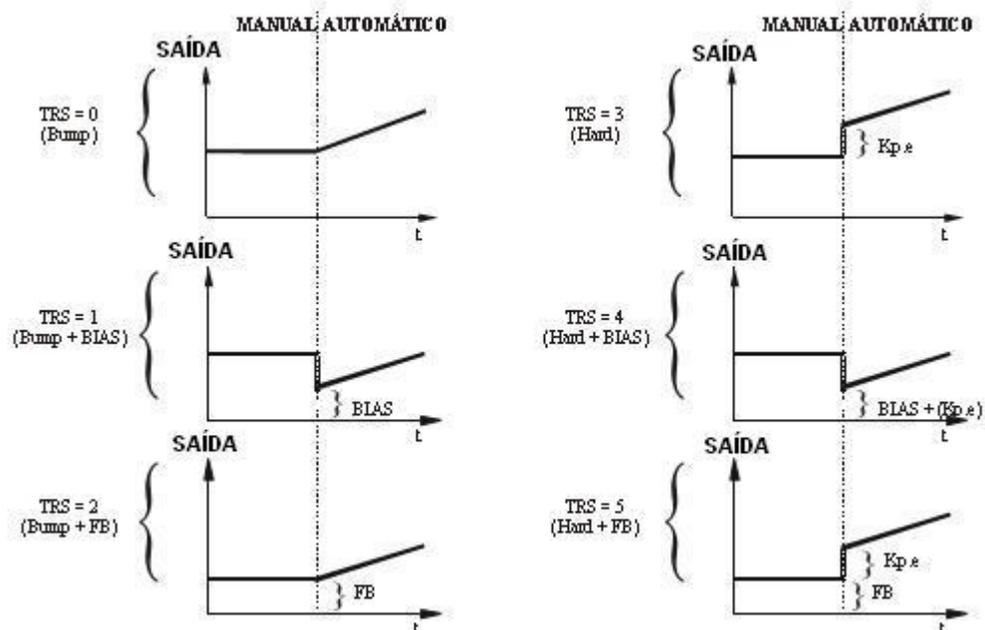
Em modo manual, o valor da saída do bloco é igual ao último valor da saída em modo automático. Neste caso pode se escrever na saída **OUT**.

No chaveamento do modo manual para o automático, o bloco inicia os cálculos partindo do valor do parâmetro **BIAS** + o termo proporcional (**KP x erro**).

TRS = 5 (Hard + FB) :

Em modo manual, o valor da saída do bloco é igual ao valor inserido na entrada **FB**. Neste caso não pode se escrever na saída **OUT**.

No chaveamento do modo manual para o automático, o bloco inicia os cálculos partindo do valor da entrada **FB** + o termo proporcional (**KP x erro**).



Transferência de Manual para Automático.

NOTA

Antes da mudança de estado do bloco, de **Manual** para **Automático**, é recomendável zerar o erro deixando o **SP** igual ao valor da **PV**.

Valor de Segurança (SEC_V, SEC, SECL, SECH e PRIOR)

Se **SEC** for verdadeiro, o valor definido na entrada **SEC_V** será repassado à saída **OUT**. Os parâmetros **SECL** e **SECH** são utilizados para a definição dos valores limites (inferior e superior) possíveis de configuração para a entrada **SEC_V**. Se a entrada possuir valores fora da faixa definida por **SECL** e **SECH**, o valor da saída ficará travada nos valores limites. Os valores aplicados aos limites **SECL** e **SECH** devem estar entre **0%** e **100%**.

O valor de segurança sempre atuará sobre o modo automático e para o modo manual, o parâmetro **PRIOR** define a prioridade da segurança sobre este modo manual.

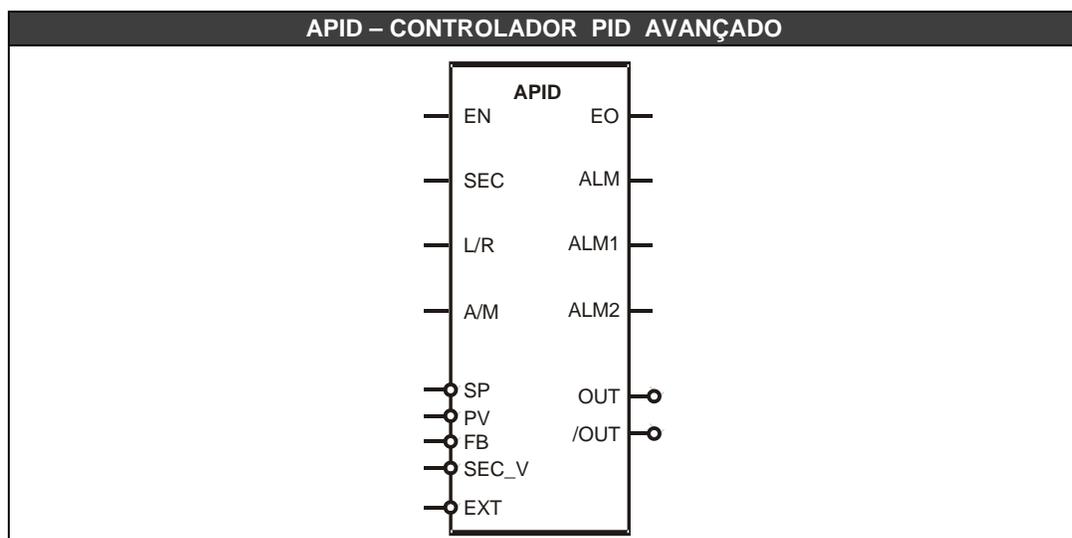
PRIOR = 0 : Man/Sec/Auto (a segurança não atuará sobre o modo manual)

PRIOR = 1 : Sec/Man/Auto (a segurança atuará sobre ambos os modos)

Set Point Local (L/R, SPL, SPLL e SPLH)

A entrada **L/R** define se o set point usado na integração será o remoto (entrada **SP**) ou o local (parâmetro interno **SPL**). Se **L/R** for verdadeiro, será usado o local, se for falso usará o remoto. O valor do set point local é limitado pelos parâmetros **SPLL** e **SPLH**. Se for escrito em **SPL** valores fora da faixa definida por **SPLL** e **SPLH**, o valor de **SPL** ficará travado em um dos dois limites. Este limite também vale para o **SP** e a **PV**, com a mesma forma de atuação que ocorre em **SPL**.

Estes limites também possuem uma segunda função, que é definir a faixa de valores para atuação das variáveis **SP** e **PV**, ou seja, utilização de valores de engenharia. O default para estes limites são 0 e 100, ou seja, neste caso considera **PV** e **SP** em porcentagem. Porém mudando os valores default de **SPLL** e **SPLH**, será possível trabalhar com qualquer faixa de valores para **PV** e **SP**, ou seja, considera-se **PV** e **SP** em unidades de engenharia.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|---|--|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | SEC | HABILITAÇÃO DO MODO DE SEGURANÇA | BOOL |
| | L/R | SELEÇÃO DO SET POINT LOCAL (1) OU REMOTO (0) | BOOL |
| | A/M | SELEÇÃO DE FUNCIONAMENTO: MANUAL (0) OU AUTOMÁTICO (1) | BOOL |
| | SP | SET POINT | FLOAT |
| | PV | VARIÁVEL DO PROCESSO | FLOAT |
| | FB | SE A/M FALSO, A ENTRADA CONECTADA EM FB É REPASSADA À SAÍDA OUT | FLOAT |
| | SEC_V | SE SEC É VERDADEIRO, A ENTRADA CONECTADA EM SEC_V SERÁ REPASSADA À SAÍDA OUT | FLOAT |
| | EXT | VARIÁVEL EXTERNA PARA DETERMINAR O GANHO ADAPTATIVO | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | ALM | ALARME DE DESVIO | BOOL |
| | ALM1 | ALARME CONFIGURÁVEL 1 | BOOL |
| | ALM2 | ALARME CONFIGURÁVEL 2 | BOOL |
| | OUT | SAÍDA | FLOAT |
| P | /OUT | SAÍDA INVERTIDA | FLOAT |
| | KP | GANHO PROPORCIONAL | FLOAT |
| | TR | TEMPO INTEGRATIVO (MIN/REP) | FLOAT |
| | TD | TERMO CONSTANTE DERIVATIVO (MIN) | FLOAT |
| | AWL | LIMITE INFERIOR ANTI-RESET WINDUP | FLOAT |
| | AWH | LIMITE SUPERIOR ANTI-RESET WINDUP | FLOAT |
| | OUTL | LIMITE INFERIOR PARA SAÍDA OUT | FLOAT |
| | OUTH | LIMITE SUPERIOR PARA SAÍDA OUT | FLOAT |
| BIAS | BIAS | FLOAT | |
| PID | DEFINE O TIPO DE CONTROLE PID SOBRE O ERRO E A VARIÁVEL DE PROCESSO | LONG | |

| | | |
|----------|--|-------|
| ALG | DEFINE O TIPO DE ALGORITMO UTILIZADO | BOOL |
| ACT | DEFINE O TIPO DE AÇÃO DIRETA/REVERSA | BOOL |
| TRS | DEFINE O TIPO DE TRANSFERÊNCIA DE AUTOMÁTICO PARA MANUAL | LONG |
| CLIM | DEFINE SE OS LIMITES DE OUTL E OUTH SÃO VÁLIDOS PARA OS MODOS AUTO/MAN OU SOMENTE AUTO | LONG |
| PRIOR | DEFINE A PRIORIDADE DO VALOR DE SEGURANÇA | LONG |
| SPL | DEFINE O VALOR DO SET POINT LOCAL | FLOAT |
| SPLL | LIMITE INFERIOR DO SET POINT LOCAL E DO PV/SP | FLOAT |
| SPLH | LIMITE SUPERIOR DO SET POINT LOCAL E DO PV/SP | FLOAT |
| SECL | LIMITE INFERIOR DO VALOR DE SEGURANÇA | FLOAT |
| SECH | LIMITE SUPERIOR DO VALOR DE SEGURANÇA | FLOAT |
| SGB | BANDA (GAP) A SER CONSIDERADO PARA O CONTROLE TIPO GAP | FLOAT |
| SGGAP | GANHO ESPECIAL DENTRO DO GAP | FLOAT |
| TYERR | TIPO DE ERRO LINEAR/QUADRÁTICO | LONG |
| DEVAL | LIMITE DO ALARME DE DESVIO | FLOAT |
| MTDA | TEMPO MÁXIMO PARA O ALARME DE DESVIO | FLOAT |
| ADAPG | AÇÃO DO GANHO ADAPTATIVO | LONG |
| INVAG | TIPO DE VARIÁVEL DE ENTRADA PARA O GANHO ADAPTATIVO | LONG |
| PPIS | PERÍODO DE AMOSTRAGEM DO PI-SAMPLING | FLOAT |
| ATPIS | TEMPO DE ATUAÇÃO DO PI-SAMPLING | FLOAT |
| INAL1 | ENTRADA DO ALARME 1 | LONG |
| TYPE1 | TIPO DO ALARME 1 | LONG |
| DBN1 | HISTERESE DO ALARME 1 | FLOAT |
| REF1 | VALOR DE REFERÊNCIA DO ALARME 1 | FLOAT |
| INAL2 | ENTRADA DO ALARME 2 | LONG |
| TYPE2 | TIPO DO ALARME 2 | LONG |
| DBN2 | HISTERESE DO ALARME 2 | FLOAT |
| REF2 | VALOR DE REFERÊNCIA DO ALARME 2 | FLOAT |
| X1...X10 | COORDENADAS X PARA CURVA DO GANHO ADAPTATIVO | FLOAT |
| Y1...Y10 | COORDENADAS Y PARA CURVA DO GANHO ADAPTATIVO | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Rampa Automática com Incremento e Decremento (ARAMP)

Descrição

Este bloco funcional, quando a entrada **EN** está verdadeira, incrementa ou decrementa a saída **OUT** de modo linear baseado num período de tempo estabelecido. Este bloco funcional pode ser utilizado para criar uma base de tempo para um gerador de um set point automático quando combinado com o bloco funcional de linearização ou uma rampa simples.

Numa aplicação de geração de set point, o ARAMP é preparado para gerar uma saída de 0 a 100% num período de tempo tal que acompanhe a curva de set point. A saída do ARAMP será conectada à entrada do bloco funcional LIN (linearização) configurado com a curva de perfil do set point.

Seleção do formato da entrada IN e da saída OUT (parâmetro PERC)

PERC = falso: os valores da entrada **IN** e da saída **OUT** são dados em percentagem (0 – 100%).

PERC = verdadeiro: os valores da entrada **IN** e da saída **OUT** são dados no formato 0 - 10000.

Seleção da base de tempo

Através do parâmetro **T_SEL**, a base de tempo do bloco pode ser seleccionada em segundos, minutos ou horas conforme a necessidade da aplicação. Este parâmetro possui os seguintes valores: 0: segundos, 1: minutos, 2: horas. Esta seleção influi diretamente no valor escolhido para o parâmetro **FTIME**.

Parâmetros FTIME e IC_DC

FTIME é o tempo que a saída leva para mudar de 0 a 100 %. A direção da mudança é dada pela entrada **IC_DC**. Se esta entrada for verdadeira, a saída **OUT** será gradualmente decrementada com velocidade definida pelo parâmetro **FTIME**, caso contrário, a saída será incrementada com a velocidade definida no parâmetro **FTIME**.

Comando de Pausa (PAUSE)

O **PAUSE** congela a saída **OUT**. Nesse instante, a saída pode ser incrementada ou decrementada através da seleção das entradas **UP** e **DOWN**.

Comandos UP e DOWN, parâmetro ASPD

O **UP** e o **DOWN** avançarão ou reverterão a saída **OUT** para um valor desejado usando o ajuste de velocidade manual pelo parâmetro **ASPD**. Este parâmetro configura a velocidade de atuação manual.

Parâmetros LOWL e HIGHL

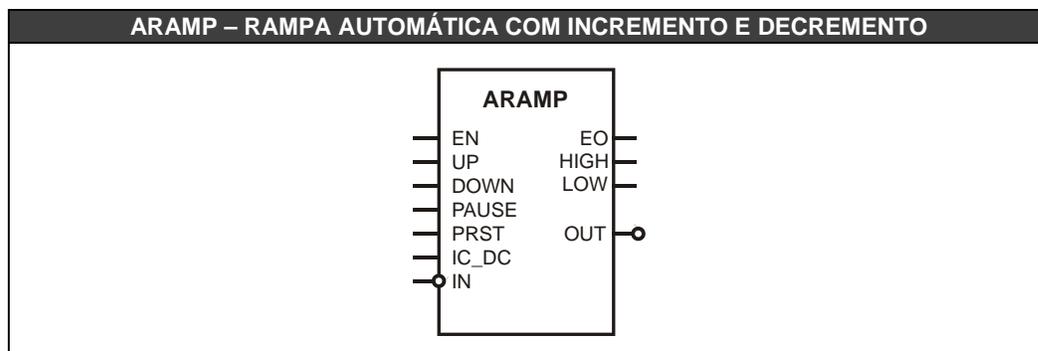
O parâmetro **LOWL** configura o limite inferior da rampa gerada pelo bloco ARAMP, enquanto que o parâmetro **HIGHL** configura o limite superior da rampa de saída. A rampa parte do valor da entrada **IN** até o valor máximo configurado no parâmetro **HIGHL**. Se o valor da entrada for menor do que **LOWL**, o valor inicial da rampa será igual **LOWL**.

Alarmes HIGH e LOW

Quando a rampa de saída atingir o limite inferior (**LOWL**) ou superior (**HIGHL**), os alarmes **LOW** e **HIGH** serão acionados. Isto é, a saída **LOW** vai para nível alto se o limite inferior é atingido. Similarmente, se o limite superior é atingido, a saída **HIGH** muda para verdadeiro.

Parâmetro ACCEL

É a aceleração manual de atuação. Quando a saída do bloco é uma parábola, o parâmetro **ACCEL** permite ajuste fino da saída, proporcionando maior definição à taxa de mudança da saída.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|---|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | UP | AVANÇA A SAÍDA OUT CONFORME ASPD | BOOL |
| | DOWN | REVERTE A SAÍDA OUT CONFORME ASPD | BOOL |
| | PAUSE | CONGELA A SAÍDA OUT | BOOL |
| | PRST | ZERA A RAMP A | BOOL |
| | IC_DC | INCREMENTO OU DECREMENTO DA SAÍDA OUT | BOOL |
| | IN | ENTRADA DO BLOCO (%) | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | HIGH | ALARME DE LIMITE SUPERIOR DA RAMP A | BOOL |
| | LOW | ALARME DE LIMITE INFERIOR DA RAMP A | BOOL |
| | OUT | RAMP A DE SAÍDA | FLOAT |
| P | T_SEL | SELEÇÃO DA BASE TEMPO (HORAS, MINUTOS OU SEGUNDOS) | LONG |
| | FTIME | TEMPO (EM SEGUNDOS) PARA MUDAR DE 0 A 100% A SAÍDA OUT | LONG |
| | ASPD | VELOCIDADE DE ATUAÇÃO MANUAL EM % POR SEG. | LONG |
| | ACCEL | ACELERAÇÃO INICIAL MANUAL DE ATUAÇÃO | LONG |
| | LOWL | LIMITE INFERIOR DO REGISTRADOR | FLOAT |
| | HIGHL | LIMITE SUPERIOR DO REGISTRADOR | FLOAT |
| | PERC | SELEÇÃO DO FORMATO DA ENTRADA E DA SAÍDA ENTRE "0 - 10000" E "0 - 100%" | BOOL |

I: Entrada O: Saída P: Parâmetro Interno

Controlador PID Otimizado (EPID)

Descrição:

Este bloco funcional, quando a entrada **EN** está verdadeira realiza o controle PID. O consagrado algoritmo PID para controle de processos contínuos, associado à flexibilidade de configuração de características de operação através de parametrização, permite a utilização deste bloco à uma grande variedade de aplicações e estratégias de controle.

Este bloco oferece várias opções de configuração do algoritmo de controle tendo como base os termos Proporcional (P), Integral (I) e Derivativo (D), que podem ser aplicados ao erro ou apenas à variável de processo (PV). O bloco também oferece três saídas referentes a alarme, sendo uma saída para alarme de desvio e duas saídas configuráveis.

Este bloco permite a seleção dos seguintes tipos de controle: PI-Sampling, Erro Quadrático e GAP.

Os limites de *anti-reset windup* (limites aplicados apenas ao termo integral) podem ser configurados pelo usuário. Além disto, tem-se opção de seleção entre: algoritmo ISA ou paralelo, ação direta ou reversa, transferência de controle de manual para automático, *bumpless* ou *hard*.

Tipo de PID

É definido pelo parâmetro **PID**.

PID = 0: tipo PI.D.

PID = 1: tipo PID.

PID = 2: tipo I.PD.

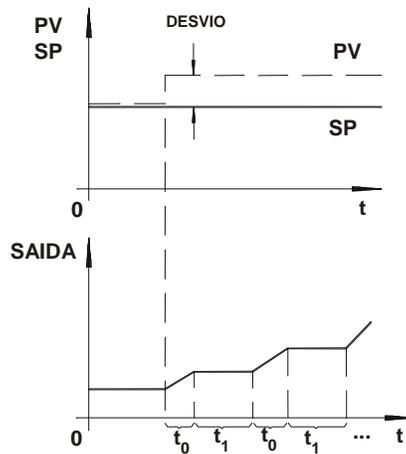
PID = 3: tipo PI-SAMPLING.

PI.D : As ações **P** e **I** atuam sobre o erro e a ação **D** sobre a variável de processo. Desta forma, o sinal de saída acompanha as mudanças de *set point* segundo as ações proporcional e integral, mas não dá uma variação indesejável devido à ação derivativa. É o mais recomendado para a maioria das aplicações com o *set point* ajustável pelo operador.

PID: As ações **P**, **I** e **D** atuam sobre o erro. Desta forma o sinal de saída é alterado quando há mudanças na variável de processo ou no *set point*. É recomendado para controle de relação ou para controle escravo de um cascata.

I.PD: Neste tipo somente a integral atua sobre o erro. Mudanças no *set point* provocam a variação no sinal de saída de maneira suave. É recomendado para processos que não podem ter variações bruscas na **MV** (variável manipulada) em função da mudança no *set point*. É o caso de processos de aquecimento com ganho muito alto.

PI-SAMPLING: Neste tipo, quando há um desvio, o sinal de saída muda de acordo com o algoritmo **PI** durante um tempo t_0 , ajustável através do parâmetro **ATRIS** (em segundos) e permanecendo constante durante um tempo t_1 , em que o período total deste ciclo (t_0+t_1) é ajustável através do parâmetro **PPIS** (em segundos). Se o desvio persistir, o sinal de saída continuará variando novamente durante o tempo t_0 e permanecendo constante durante o tempo t_1 . Este tipo de controle é recomendado para processos com alto tempo morto.



$t_0 + t_1$ = Período de Amostra (PPIS) – em segundos

t_0 = Tempo de Atuação (ATRIS) – em segundos

PI-Sampling

Tipo de Algoritmo (ALG)

O tipo de algoritmo é definido pelo parâmetro **ALG**.

ALG = falso: Algoritmo Paralelo ou Ideal.

ALG = verdadeiro: Algoritmo ISA ou Não-Interativo.

$$\text{PARALELO} : MV(t) = K_p e(t) + \frac{1}{T_R} \int e(t) dt + T_D \frac{de(t)}{dt}$$

$$\text{ISA} : MV(t) = K_p \left[e(t) + \frac{1}{T_R} \int e(t) dt + T_D \frac{de(t)}{dt} \right]$$

Tipo de Ação (ACT)

Existem processos que requerem que o sinal de saída (variável manipulada – **MV**) aumente quando a variável de processo (**PV**) aumenta, enquanto que outros requerem o contrário. A escolha do tipo de ação é feita através do parâmetro **ACT**.

| PARÂMETRO | TIPO DE AÇÃO | ERRO (em porcentagem) | EFEITO |
|-------------------------|--------------|-----------------------|---------------------------------|
| ACT = falso | Reversa | $e = SP - PV$ | Saída diminui com aumento da PV |
| ACT = verdadeiro | Direta | $e = PV - SP$ | Saída aumenta com aumento da PV |

Tipo de Erro – Linear/Quadrático (TYERR)

No desvio ou erro linear (**TYERR = 0**), o erro (em porcentagem) considerado nos cálculos será:

Ação Reversa: $e = SP - PV$ "Saída diminui com o aumento da PV"

Ação Direta: $e = PV - SP$ "Saída aumenta com o aumento da PV"

Existem processos em que o desvio em relação ao *set point* é preferível aos distúrbios causados pelo controlador no processo. Portanto, a atuação do controle deve ser pequena para pequenos distúrbios e aumentar gradualmente com o aumento do desvio. Um exemplo típico deste tipo de processo é o controle de nível de um tanque em que o *set point* não é tão importante quanto a estabilidade da vazão de descarga. Este tipo de processo pode ser controlado com ganho adaptativo, controle com gap ou erro quadrático.

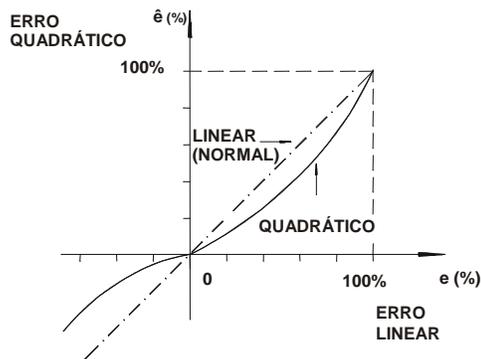
No erro quadrático (**TYERR = 1**), o erro considerado nos cálculos será:

Ação Reversa: $e = SP - PV$ "Saída diminui com o aumento da PV"

Ação Direta: $e = PV - SP$ "Saída aumenta com o aumento da PV"

$$\hat{e} = \frac{e \cdot |e|}{100}$$

\hat{e} : Erro (em porcentagem) a ser considerado nos cálculos do EPID.



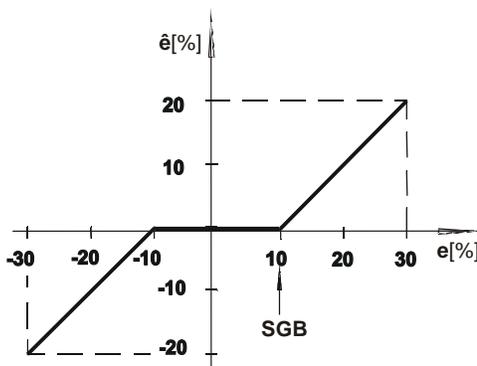
Erro Quadrático x Erro Linear

Controle Tipo GAP (SGB e SGGAP)

Existem aplicações em que o controle é instável dentro de um valor, em torno do *set point*, devido à faixa morta do atuador, ruído ou outras razões. Neste caso, é aconselhável que o controlador tenha uma ação diferenciada nesta faixa. O controle tipo GAP ou GAP com ganho adaptativo pode ser utilizado para resolver este problema.

EXEMPLO:

Considerar erro (\hat{e}), em porcentagem, para um controle tipo GAP com uma banda igual a $\pm 10\%$ (**SGB=10**) e ganho diferenciado igual a zero (**SGGAP=0**).

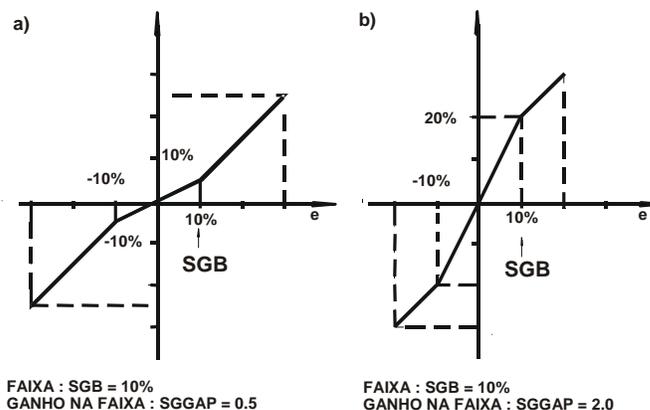


Controle de GAP com Ganho Especial = 0

Alguns processos podem necessitar de um ganho especial dentro do GAP. Em tais casos, é possível selecionar um fator para o parâmetro **SGGAP**, o qual multiplica o erro, fazendo com que o erro a ser considerado no cálculo do EPID seja:

$$\hat{e} = e \cdot \text{SGGAP}$$

Dentro do GAP, a ação de controle será mais rápida quando **SGGAP>1** e mais lenta quando **SGGAP<1**. Para **SGGAP=0** (banda nula) o controle de tipo GAP não é ativado.



Controle de GAP com Ganho Especial (a) Ganho < 1, (b) Ganho > 1

Anti-Reset Windup pelo Termo Integral (AWL e AWH)

Usualmente o algoritmo de controle pára automaticamente a contribuição do termo integral, quando o sinal de saída atinge os limites inferior ou superior configurados através dos parâmetros **AWL** e **AWH**. As contribuições dos termos proporcional e derivativo não são afetadas. Uma característica diferenciadora do algoritmo deste bloco é a possibilidade de configuração desses limites. Quando o limite **AWH** é maior que o limite **OUTH**, a saída do bloco OUT é travada no valor **OUTH**, mas internamente, o algoritmo continua o cálculo integral até o limite **AWH**. O usuário pode evitar este caso configurando o limite **AWH** a um valor menor ou igual a **OUTH**, assim obtém-se respostas mais rápidas evitando-se assim, por exemplo, um *overshoot* em processos de aquecimento. A mesma idéia se aplica aos limites baixos (**AWL** e **OUTL**).

Limites para a saída OUT (CLIM, OUTL e OUTH)

Os limites para a saída **OUT** são definidos nos parâmetros **OUTL** e **OUTH**.

Os valores aplicados nestes dois parâmetros devem estar entre **-2%** e **102%**.

O parâmetro **CLIM** define em quais modos (automático/manual) serão aplicados os limites **OUTL** e **OUTH**.

CLIM = 0 : AUTO/MAN (em ambos os modos)
CLIM = 1 : AUTO (somente no automático)

Em qualquer modo de operação que seja possível realizar escrita na saída **OUT**, se o valor escrito estiver fora dos limites configurados, a saída se manterá no valor anterior.

Alarme de Desvio (DEVAL, MTDA, ALM)

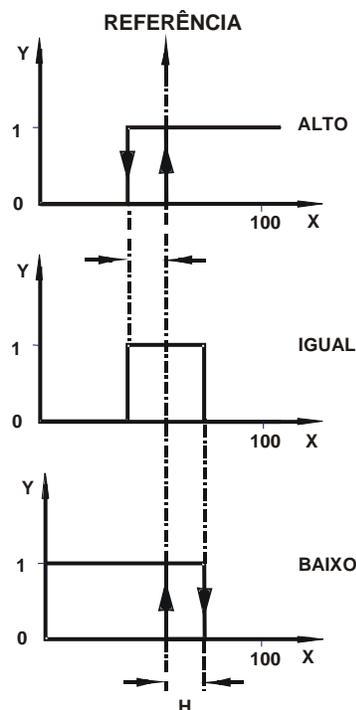
O alarme pode ser programado para qual desvio (em porcentagem) se deseja um alarme (**DEVAL**) e quanto tempo (em segundos) este desvio pode ocorrer sem ativar o alarme (**MTDA**). Por exemplo, se **DEVAL=5** e **MTDA=30**, então a saída **ALM** será ativada (nível lógico 1) se um desvio de 5% persistir por um tempo maior que 30 segundos.

Para **MTDA=0** (tempo infinito) o alarme de desvio não é ativado (sem alarme).

Alarme Configurável (ALM1, INAL1, TYPE1, DBN1, REF1, ALM2, INAL2, TYPE2, DBN2, REF2)

Os alarmes configuráveis, independentes, são **ALM1** e **ALM2**. Eles são ativados através de seus respectivos parâmetros, comparando o valor de referência **REFx** e a variável selecionada em **INALx** (SP ou MV), sendo possível selecionar o tipo de comparação **TYPEx**, checando assim se **INALx** está acima (*High*), abaixo (*Low*) ou igual (*Equal*) a **REFx**.

Para evitar a oscilação do sinal de saída quando a variável está muito próxima da referência, pode ser ajustado um valor de histerese através do parâmetro **DBNx**. Os alarmes trabalham de acordo com a figura a seguir:



Ação do Alarme com Histerese

Constantes do PID (parâmetros KP, TR, TD, BIAS e entrada FB)

KP – Ganho Proporcional.

TR – Tempo da Integral em minutos/repetição, portanto, quanto maior este parâmetro menor é a ação integral. Pode ser interpretado como sendo o tempo necessário para a saída ser incrementada/decrementada do valor do erro (no PID paralelo), mantendo-se o mesmo constante.

TD – Tempo Derivativo em minutos. O termo derivativo é calculado usando uma pseudoderivada, isto é, uma ação semelhante a um *lead/lag*, na qual a constante de *lag* é $\alpha \cdot TD$. Na implementação deste bloco o fator α é igual a 0,13.

BIAS – Neste parâmetro é possível ajustar o valor inicial da saída quando o controle é transferido de manual para automático. O valor aplicado neste parâmetro deve estar entre **0%** e **100%**. A utilização desta entrada pode ser feita através da seleção do parâmetro **TRS**.

FB – Através desta entrada é possível ajustar o valor inicial da saída quando o controle é transferido para manual. O valor aplicado nesta entrada deve estar entre **0%** e **100%**. A utilização desta entrada pode ser feita através da seleção do parâmetro **TRS**.

Entrada A/M (Automático/Manual)

Se **A/M** for verdadeiro, o EPID estará em controle automático e se **A/M** for falso o EPID estará em controle manual.

Tipos de transferência de Manual para Automático (TRS)

O valor da saída do bloco EPID é definido através do parâmetro **TRS**.

TRS = 0 (Bumpless) :

Em modo manual, o valor da saída do bloco é igual ao último valor da saída em modo automático. Neste caso pode se escrever na saída **OUT**.

No chaveamento do modo manual para o automático, o bloco inicia os cálculos partindo do último valor da saída em modo manual.

TRS = 1 (Bumpless + BIAS) :

Em modo manual, o valor da saída do bloco é igual ao último valor da saída em modo automático. Neste caso pode se escrever na saída **OUT**.

No chaveamento do modo manual para o automático, o bloco inicia os cálculos partindo do valor do parâmetro **BIAS**.

TRS = 2 (Bumpless + FB) :

Em modo manual, o valor da saída do bloco é igual ao valor inserido na entrada **FB**. Neste caso não pode se escrever na saída **OUT**.

No chaveamento do modo manual para o automático, o bloco inicia os cálculos partindo do valor da entrada **FB**.

TRS = 3 (Hard) :

Em modo manual, o valor da saída do bloco é igual ao último valor da saída em modo automático. Neste caso pode se escrever na saída **OUT**.

No chaveamento do modo manual para o automático, o bloco inicia os cálculos partindo do último valor da saída em modo manual + o termo proporcional (**KP x erro**).

TRS = 4 (Hard + BIAS) :

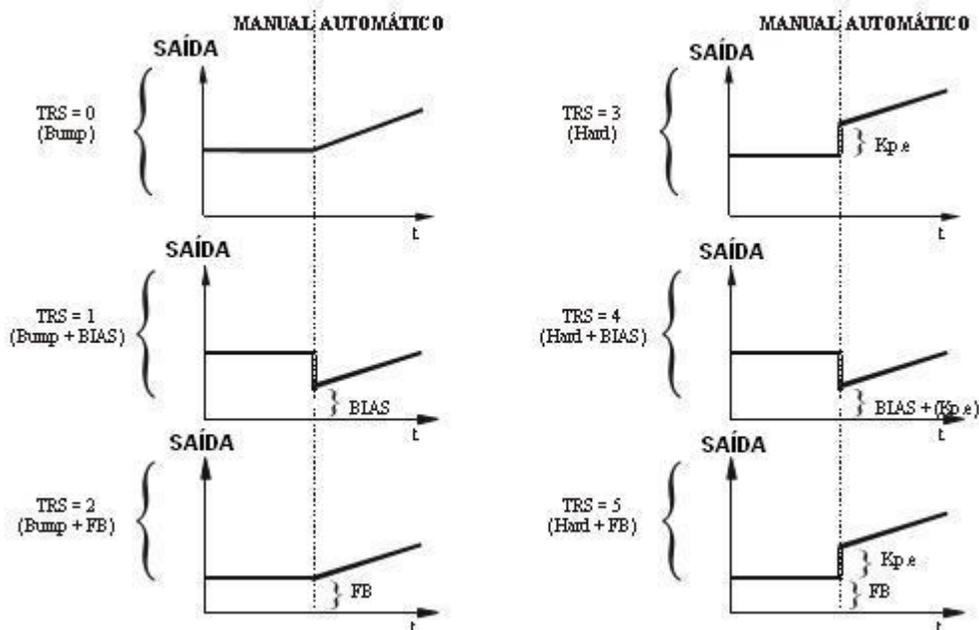
Em modo manual, o valor da saída do bloco é igual ao último valor da saída em modo automático. Neste caso pode se escrever na saída **OUT**.

No chaveamento do modo manual para o automático, o bloco inicia os cálculos partindo do valor do parâmetro **BIAS** + o termo proporcional (**KP x erro**).

TRS = 5 (Hard + FB) :

Em modo manual, o valor da saída do bloco é igual ao valor inserido na entrada **FB**. Neste caso não pode se escrever na saída **OUT**.

No chaveamento do modo manual para o automático, o bloco inicia os cálculos partindo do valor da entrada **FB** + o termo proporcional (**KP x erro**).



Transferência de Manual para Automático.

NOTA

Antes da mudança de estado do bloco, de **Manual** para **Automático**, é recomendável zerar o erro deixando o **SP** igual ao valor da **PV**.

Valor de Segurança (SEC_V, SEC, SECL, SECH e PRIOR)

Se **SEC** for verdadeiro, o valor definido na entrada **SEC_V** será repassado à saída **OUT**. Os parâmetros **SECL** e **SECH** são utilizados para a definição dos valores limites (inferior e superior) possíveis de configuração para a entrada **SEC_V**. Se a entrada possuir valores fora da faixa definida por **SECL** e **SECH**, o valor da saída ficará travada nos valores limites. Os valores aplicados aos limites **SECL** e **SECH** devem estar entre **0%** e **100%**.

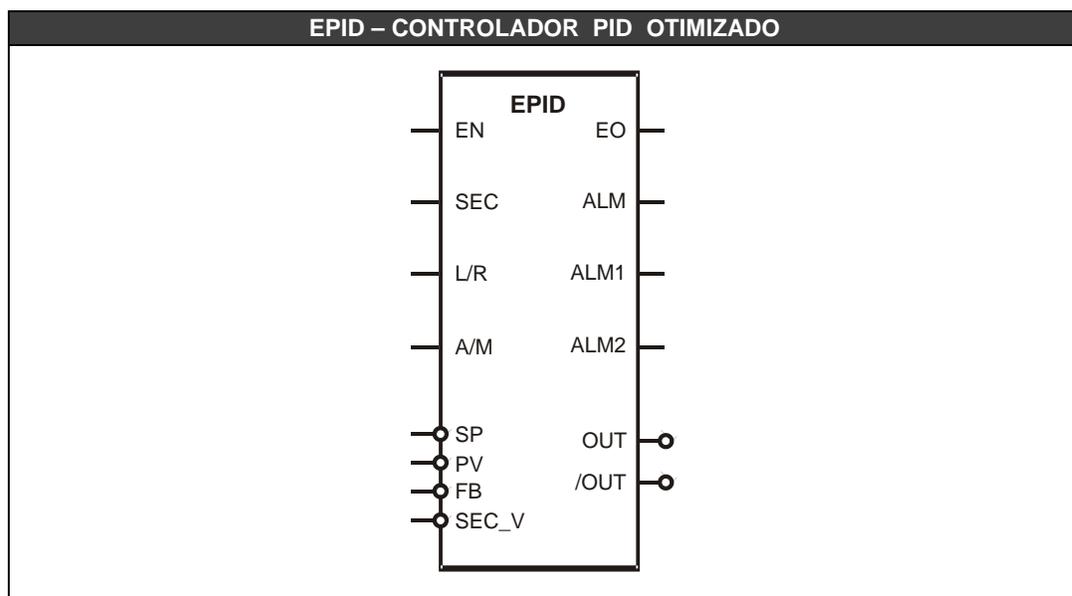
O valor de segurança sempre atuará sobre o modo automático e para o modo manual, o parâmetro **PRIOR** define a prioridade da segurança sobre este modo manual.

PRIOR = 0 : Man/Sec/Auto (a segurança não atuará sobre o modo manual)
PRIOR = 1 : Sec/Man/Auto (a segurança atuará sobre ambos os modos)

Set Point Local (L/R, SPL, SPLL e SPLH)

A entrada **L/R** define se o *set point* usado na integração será o remoto (entrada **SP**) ou o local (parâmetro interno **SPL**). Se **L/R** for verdadeiro, será usado o local, se for falsa usará o remoto. O valor do *set point* local é limitado pelos parâmetros **SPLL** e **SPLH**. Se for escrito em **SPL** valores fora da faixa definida por **SPLL** e **SPLH**, o valor de **SPL** ficará travado em um dos dois limites. Este limite também vale para o **SP** e a **PV**, com a mesma forma de atuação que ocorre em **SPL**.

Estes limites também possuem uma segunda função, que é definir a faixa de valores para atuação das variáveis **SP** e **PV**, ou seja, utilização de valores de engenharia. O default para estes limites são 0 e 100, ou seja, neste caso considera **PV** e **SP** em porcentagem. Porém mudando os valores default de **SPLL** e **SPLH**, será possível trabalhar com qualquer faixa de valores para **PV** e **SP**, ou seja, considera-se **PV** e **SP** em unidades de engenharia.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|--------------------------------|--|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | SEC | HABILITAÇÃO DO MODO DE SEGURANÇA | BOOL |
| | L/R | SELEÇÃO DO SET POINT LOCAL (1) OU REMOTO (0) | BOOL |
| | A/M | SELEÇÃO DE FUNCIONAMENTO: MANUAL (0) OU AUTOMÁTICO (1) | BOOL |
| | SP | SET POINT | FLOAT |
| | PV | VARIÁVEL DO PROCESSO | FLOAT |
| | FB | SE A/M FALSO, A ENTRADA CONECTADA EM FB É REPASSADA À SAÍDA OUT | FLOAT |
| | SEC_V | SE SEC É VERDADEIRO, A ENTRADA CONECTADA EM SEC_V SERÁ REPASSADA À SAÍDA OUT | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | ALM | ALARME DE DESVIO | BOOL |
| | ALM1 | ALARME CONFIGURÁVEL 1 | BOOL |
| | ALM2 | ALARME CONFIGURÁVEL 2 | BOOL |
| | OUT | SAÍDA | FLOAT |
| P | /OUT | SAÍDA INVERTIDA | FLOAT |
| | KP | GANHO PROPORCIONAL | FLOAT |
| | TR | TEMPO INTEGRATIVO (MIN/REP) | FLOAT |
| | TD | TERMO CONSTANTE DERIVATIVO (MIN) | FLOAT |
| | AWL | LIMITE INFERIOR DO ANTI-RESET WINDUP | FLOAT |
| | AWH | LIMITE SUPERIOR DO ANTI-RESET WINDUP | FLOAT |
| OUTL | LIMITE INFERIOR PARA SAÍDA OUT | FLOAT | |

| | | | |
|--|-------|--|-------|
| | OUTH | LIMITE SUPERIOR PARA SAÍDA OUT | FLOAT |
| | BIAS | BIAS | FLOAT |
| | PID | DEFINE O TIPO DE CONTROLE PID SOBRE O ERRO E A VARIÁVEL DE PROCESSO | LONG |
| | ALG | DEFINE O TIPO DE ALGORITMO UTILIZADO | BOOL |
| | ACT | DEFINE O TIPO DE AÇÃO DIRETA/REVERSA | BOOL |
| | TRS | DEFINE O TIPO DE TRANSFERÊNCIA DE AUTOMÁTICO PARA MANUAL | LONG |
| | CLIM | DEFINE SE OS LIMITES DE OUTL E OUTH SÃO VÁLIDOS PARA OS MODOS AUTO/MAN OU SOMENTE AUTO | LONG |
| | PRIOR | DEFINE A PRIORIDADE DO VALOR DE SEGURANÇA | LONG |
| | SPL | DEFINE O VALOR DO SET POINT LOCAL | FLOAT |
| | SPLL | LIMITE INFERIOR DO SET POINT LOCAL E DO PV/SP | FLOAT |
| | SPLH | LIMITE SUPERIOR DO SET POINT LOCAL E DO PV/SP | FLOAT |
| | SECL | LIMITE INFERIOR DO VALOR DE SEGURANÇA | FLOAT |
| | SECH | LIMITE SUPERIOR DO VALOR DE SEGURANÇA | FLOAT |
| | SGB | BANDA (GAP) A SER CONSIDERADO PARA O CONTROLE TIPO GAP | FLOAT |
| | SGGAP | GANHO ESPECIAL DENTRO DO GAP | FLOAT |
| | TYERR | TIPO DE ERRO LINEAR/QUADRÁTICO | LONG |
| | DEVAL | LIMITE DO ALARME DE DESVIO | FLOAT |
| | MTDA | TEMPO MÁXIMO PARA O ALARME DE DESVIO | FLOAT |
| | PPIS | PERÍODO DE AMOSTRAGEM DO PI-SAMPLING | FLOAT |
| | ATPIS | TEMPO DE ATUAÇÃO DO PI-SAMPLING | FLOAT |
| | INAL1 | ENTRADA DO ALARME 1 | LONG |
| | TYPE1 | TIPO DO ALARME 1 | LONG |
| | DBN1 | HISTERESE DO ALARME 1 | FLOAT |
| | REF1 | VALOR DE REFERÊNCIA DO ALARME 1 | FLOAT |
| | INAL2 | ENTRADA DO ALARME 2 | LONG |
| | TYPE2 | TIPO DO ALARME 2 | LONG |
| | DBN2 | HISTERESE DO ALARME 2 | FLOAT |
| | REF2 | VALOR DE REFERÊNCIA DO ALARME 2 | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Totalização Otimizada (ETOT)

Descrição:

Este bloco fornece a totalização da entrada. Esta totalização é a integral da entrada multiplicada por um fator de escala, **FCF**. Este fator de escala permite que o usuário configure a totalização em 3 modos de operação. Se a aplicação requer o cálculo instantâneo do volume totalizado, basta utilizar o bloco ETOT para integrar a vazão, que é a derivada temporal do volume. A base de tempo do bloco é segundos.

Uma vazão é dada, geralmente, em unidades de engenharia (EU) por unidade de tempo. Por exemplo:

Uma vazão de 1 m³/segundo como entrada do bloco ETOT terá como saída a totalização em m³. Considere que a aplicação necessite do valor da energia de um dispositivo elétrico. O bloco ETOT permite que se calcule o valor desta energia através da potência instantânea. Pois:

$$Energia = \int Pot(t)dt$$

e ainda $Pot(t) = V(t) \cdot I(t)$, onde $V(t)$ é a tensão instantânea e $I(t)$ a corrente instantânea.

Saída **OUT** e parâmetro **TU**

O intervalo de tempo em que a saída é totalizada está de acordo com o valor configurado em **TU**. A saída **OUT** é o valor da totalização.

Saída **MEM**

Guarda o valor da totalização antes do último *reset*.

Saídas **HIGH** e **PHIGH**

Se a totalização se torna maior ou igual aos valores configurados em **TRIP** e **PTRIP**, as saídas **HIGH** e **PHIGH** são ativadas respectivamente.

Parâmetro **CutOff**

Valor no qual se a vazão de entrada for menor ou igual a **CutOff**, a totalização não se realiza.

Parâmetro **FCF**

O parâmetro **FCF** permite ao bloco ETOT operar em 4 modos diferentes:

a) **IN** é FLOAT e representa a vazão em unidades de engenharia:

FCF deve ser igual a 1 para ter totalização sem qualquer fator de escala em unidades de engenharia (ou ajuste o fator que você queira usar). Por exemplo:

Vazão **Q** é medida em m³/horas. 1 hora possui 3600 segundos. Portanto, o valor de **TU** deve ser igual a 3600. Supondo uma vazão constante de 60 m³/hora, a totalização será dada pela expressão:

$$TOT(t) = \int_0^{t(\text{segundos})} \frac{FCF}{TU} * IN(t)dt = \int_0^{t(\text{segundos})} \frac{1}{3600} * 60dt = \int_0^{t(\text{segundos})} \frac{1}{60}dt [m^3]$$

Portanto, após 1 minuto ou 1/60 horas ou 60 segundos o valor de TOT será:

$$TOT[m^3] = \int_0^{60} \frac{1}{60}dt = 1m^3$$

A cada 1/60 horas ou a cada 1 minuto o bloco totaliza a entrada e mostra este valor na saída. Pois:

60 m³ _____ 1 hora

1 m³ _____ t (intervalo de tempo em que a totalização é mostrada)

Então, t= 1/60 horas ou 1 minuto.

b) **IN** é FLOAT e representa a vazão em porcentagem:

Neste caso, a entrada será interpretada como uma porcentagem representada por um número real entre 0 e 100 (0% e 100%, respectivamente). **FCF** deve ser igual à vazão máxima em unidades de engenharia (Vazão a 100%) para ter a totalização em unidades de engenharia. A configuração do parâmetro **TU** é similar à entrada real em unidades de engenharia. A totalização será mostrada na unidade de engenharia configurada.

c) **IN** é INTEIRO:

Neste caso, a entrada será interpretada como um número inteiro entre 0 e 10000 (0% e 100.00% respectivamente). **FCF** deve ser igual à vazão máxima em unidades de engenharia dividido por 10000. Supondo uma vazão máxima de 1 m³/segundo e uma vazão de 0,5 m³/segundo. O valor de **FCF** é igual à vazão máxima dividida por 10000, isto é, 0,0001. O valor de **TU** é neste caso 1, pois a totalização é dada em m³. Uma entrada de 0,5 m³/segundo equivale a 5000 (ou 50 % da escala). Portanto:

$$OUT = \int_0^t \frac{FCF}{TU} * IN\%(t) dt = \int_0^t 0.0001 * 5000 dt = 0.5t(m^3)$$

Logo, em 1 minuto (ou 60 segundos) o valor totalizado será de 30 m³.

d) Quando **FCF** é menor que zero:

Quando o bloco estiver totalizando uma vazão negativa, a totalização é decrementada, enquanto que quando a vazão é positiva a totalização é incrementada. Quando **FCF** for maior do que zero, isto é, positivo, o bloco totalizador só aceitará vazões positivas.

Entrada **RST**

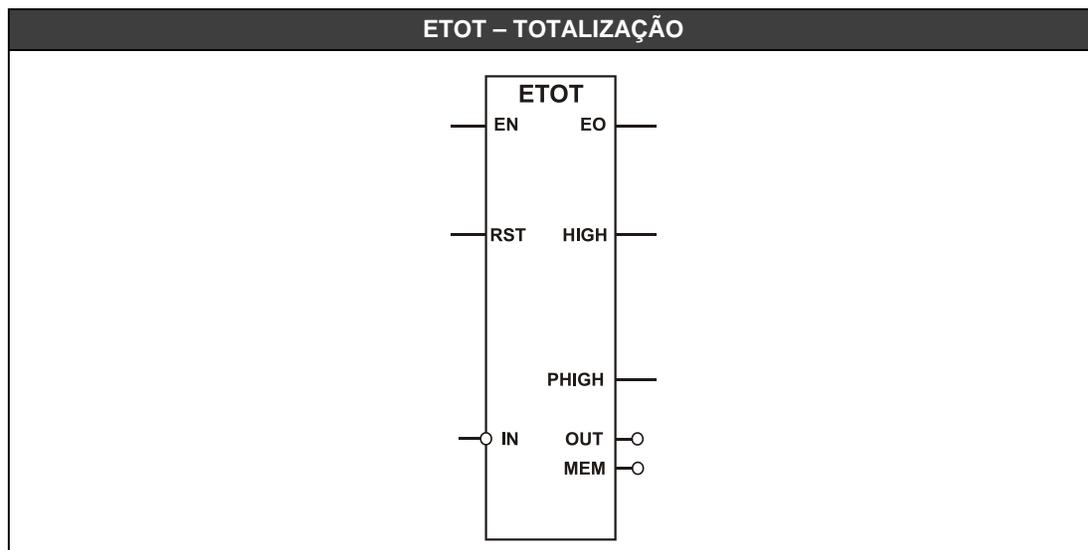
Se a entrada **RST** for alterada para verdadeiro, a totalização será reiniciada e os registradores internos do bloco ETOT serão zerados.

Parâmetro **OpMode**

Indica o modo de operação:

AUTO/DEMAND: neste modo o ETOT é reiniciado por um valor verdadeiro na entrada **RST** ou quando o valor da totalização atinge o valor de **TRIP**.

DEMAND: neste modo o ETOT só é reiniciado pela entrada **RST**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|--|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | RST | ZERA O TOTALIZADOR | BOOL |
| | IN | ENTRADA | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | HIGH | ALARME QUE INDICA QUANDO A TOTALIZAÇÃO ATINGIU O VALOR DE TRIP. | BOOL |
| | PHIGH | ALARME QUE INDICA QUANDO A TOTALIZAÇÃO ATINGIU O VALOR DE PTRIP. | FLOAT |
| | MEM | ARMAZENA O VALOR DA TOTALIZAÇÃO QUANDO OCORRE UM RESET. | FLOAT |
| | OUT | SAÍDA TOTALIZADA | FLOAT |

| | | | |
|----------|--------|--|-------|
| | TU | VALOR DE TOTALIZAÇÃO PARA UMA UNIDADE DE CONTAGEM | FLOAT |
| | FCF | FATOR DE TAXA DE VAZÃO | FLOAT |
| P | OpMode | INDICA O MODO DE OPERAÇÃO, SE O RESET É POR DEMANDA OU QUANDO ATINGE O VALOR DE TRIP | FLOAT |
| | TRIP | VALOR QUE GERA O ALARME HIGH | FLOAT |
| | PTRIP | VALOR QUE GERA O ALARME PHIGH | FLOAT |
| | CutOff | SE A ENTRADA IN FOR MENOR QUE ESTE VALOR A SAÍDA NÃO TOTALIZA | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Linearização (LIN)

Descrição:

Quando a entrada **EN** está verdadeiro, este bloco simula uma função usando uma tabela de pontos (coordenadas **x,y**). Valores intermediários são calculados usando o método de interpolação linear. Em cada bloco será possível implementar curvas com até 10 pontos, caso haja a necessidade de mais pontos, basta arranjar em série outros blocos LIN de forma a obter as curvas desejadas. Uma tabela de pontos, pares **X** e **Y**, deverá ser preenchida de modo a representar a função. Para cada valor de entrada **Xn** existe uma correspondente saída **Yn**, isto é, o bloco implementa uma função $f(x)$.

Seleção do formato da saída OUT (parâmetro PERC)

PERC = 0 : falso

Utilizado em operações de porcentagem com números reais, por exemplo: 21.56%.

Neste caso, o valor fornecido pela saída (**OUT**) será um número real. Ex.: Se o valor obtido através do processamento do bloco for 20.45, então o valor fornecido pela saída será 20.45 ou se o valor obtido através do processamento do bloco for 20.55, então o valor fornecido para a saída será 20.55.

PERC = 1 : verdadeiro

Utilizado em operações de porcentagem com valores inteiro (0 a 10000), onde 0 é a representação de 0%, 2156 a de 21.56% e 10000 a de 100.00% .

Neste caso, o valor fornecido pela saída (**OUT**) será um número inteiro. Ex.: Se o valor obtido através do processamento do bloco for 20.45, então o valor fornecido para a saída será 20 ou se o valor obtido através do processamento do bloco for 20.55, então o valor fornecido pela saída será 21.

Bypass

Quando a entrada **PASS** está verdadeiro, o bloco **LIN** repassa o valor da entrada do bloco para a saída, considerando a definição do parâmetro **PERC**.

Comportamento Serial

Quando uma aplicação requerer mais de 10 pontos, vários blocos LIN podem ser colocados em série. O comportamento serial do bloco é definido pelo parâmetro **TYPE**, conforme segue:

TYPE = 0 : ALONE (bloco único)

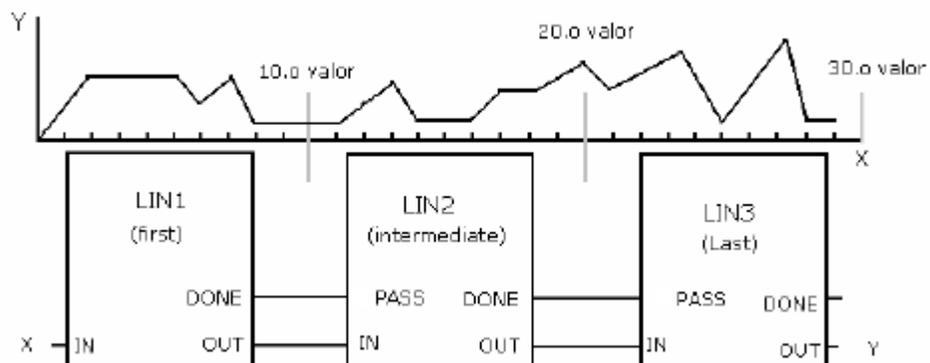
TYPE = 1 : FIRST (primeiro bloco)

TYPE = 2 : INTERMEDIATE (bloco intermediário)

TYPE = 3 : LAST (último bloco)

A saída **DONE** deve ser ligada à entrada **PASS** do próximo bloco **LIN**. O primeiro bloco do arranjo deve ser configurado como **FIRST**, todos os intermediários como **INTERMEDIATE** e o último como **LAST**.

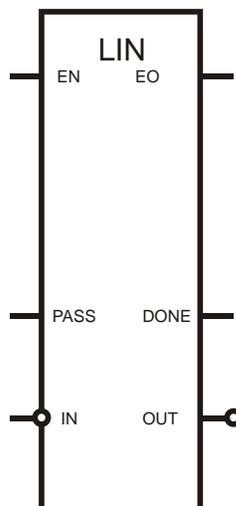
Por exemplo, uma aplicação que requer 30 pontos para representar uma função, tem a seguinte configuração:



NOTAS

- Os pares (x,y) devem ser inseridos em ordem crescente dos valores de "X", iniciando-se no par (x1,y1) e sem saltar índices.
- Não é necessário utilizar todos os 10 pontos fornecidos pelo bloco para a geração da curva, porém será necessário repetir os valores de X e Y do último ponto da curva desejada nos demais pontos não utilizados. Por exemplo, a curva desejada utilizará de x1,y1 até os pares x5,y5, então os demais pares deverão ser configurados com os mesmos valores de x5,y5.
- Os valores inseridos para a coordenada X devem ser da mesma grandeza (valores em porcentagem ou unidade de engenharia) do valor da entrada **IN** e os valores inseridos para a coordenada Y devem ser da mesma grandeza do valor esperado na saída **OUT**.

LIN – LINEARIZAÇÃO



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|---|--|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | PASS | REPASSA A ENTRADA ATÉ A SAÍDA SEM NENHUM PROCESSAMENTO | BOOL |
| | IN | ENTRADA DO BLOCO | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | DONE | HABILITA O BLOCO LIN SEGUINTE EM UMA APLICAÇÃO EM SÉRIE. | BOOL |
| | OUT | SAÍDA DO BLOCO | FLOAT |
| P | TYPE | DEFINE TIPO DE COMPORTAMENTO SERIAL | LONG |
| | X1 | X PARA O PRIMEIRO PONTO | FLOAT |
| | Y1 | Y PARA O PRIMEIRO PONTO | FLOAT |
| | X2 | X PARA O SEGUNDO PONTO | FLOAT |
| | Y2 | Y PARA O SEGUNDO PONTO | FLOAT |
| | : | | |
| | X9 | X PARA O NONO PONTO | FLOAT |
| | Y9 | Y PARA O NONO PONTO | FLOAT |
| | X10 | X PARA O ÚLTIMO PONTO | FLOAT |
| | Y10 | Y PARA O ÚLTIMO PONTO | FLOAT |
| PERC | SELEÇÃO DO FORMATO DA SAÍDA (NÚMEROS REAIS OU INTEIROS) | BOOL | |

I: Entrada O: Saída P: Parâmetro Interno

LEAD LAG (LLAG)

Descrição:

Este é um bloco de compensação dinâmica o qual pode operar como função derivativa e como uma função de compensação "lead-lag". A seleção entre ambas as funções é feita pelo parâmetro **DER**.

O bloco **LLAG** fornece compensação dinâmica do parâmetro **IN**. O usuário deve configurar os parâmetros **K1** e **K2** para obter o relacionamento desejado de entrada/saída.

FUNÇÃO DERIVATIVA (DER = true)

No modo derivativo, o bloco efetua a seguinte função de transferência:

$$O(s) = \frac{T_D s}{1 + T s} I(s)$$

Onde

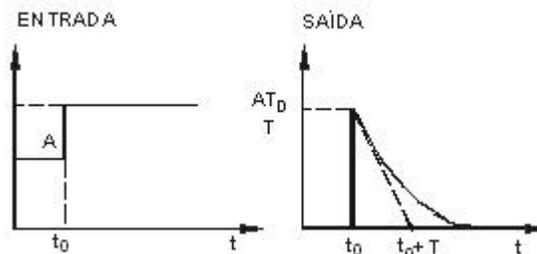
$I(s)$ e $O(s)$ - Transformadas de Laplace dos sinais de entrada e saída.

T_D - Constante derivativa, ajustada pelo parâmetro **K2** (segundos)

T - Constante "Lag" ajustada pelo parâmetro **K1** (segundos)

Quando $T=0$, o sinal de saída representa a taxa de variação do sinal de entrada no período determinado por T_D . Por exemplo, se o sinal de entrada varia numa taxa de 15% por segundo e $T_D=6$ seg., o sinal de saída será $15 * 6 = 90\%$ enquanto o sinal da entrada mantiver sua taxa de variação. A saída retorna a zero quando a entrada fica constante.

Quando $T>0$ o sinal de saída é submetido a um atraso (**LAG**). A resposta a um sinal de entrada em degrau com amplitude A é mostrada na figura abaixo:



Resposta da Função Derivada com um atraso na entrada IN

Esta função é usada quando é desejado mudança na taxa da variável.

FUNÇÃO LEAD-LAG E CONSTANTE DE TEMPO (DER=false)

Quando operando como lead-lag o bloco implementa a seguinte função de transferência:

$$O(s) = \frac{1 + T_D s}{1 + T s} I(s)$$

Onde

T_D - Constante "Lead", ajustada pelo parâmetro **K2** (segundos)

T - Constante "Lag", ajustada pelo parâmetro **K1** (segundos)

O parâmetro **K1** especifica o tempo de atraso para o bloco. Baseado numa mudança degrau na entrada este é o tempo para alcançar 63,2% do valor do degrau.

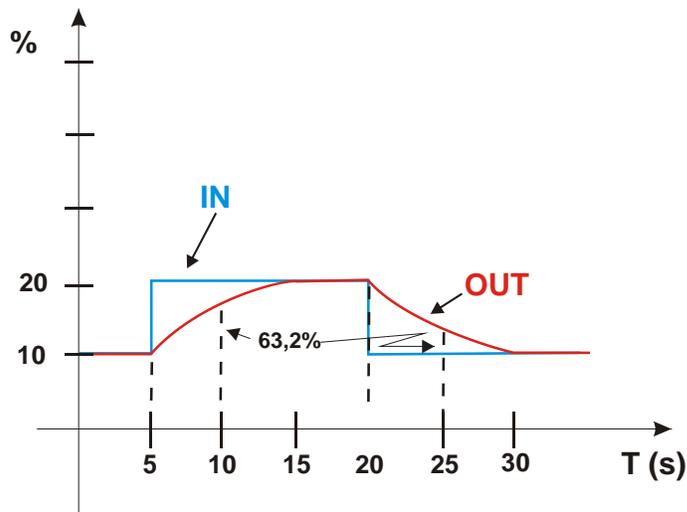
O parâmetro **K2** especifica o ganho ou impulso aplicado à entrada.

Para ambos os casos, o parâmetro **FLW** é usado para fazer com que a saída siga a entrada. Desta forma, quando **FLW** é verdadeiro, a saída **OUT** terá o valor da entrada e o algoritmo não será executado.

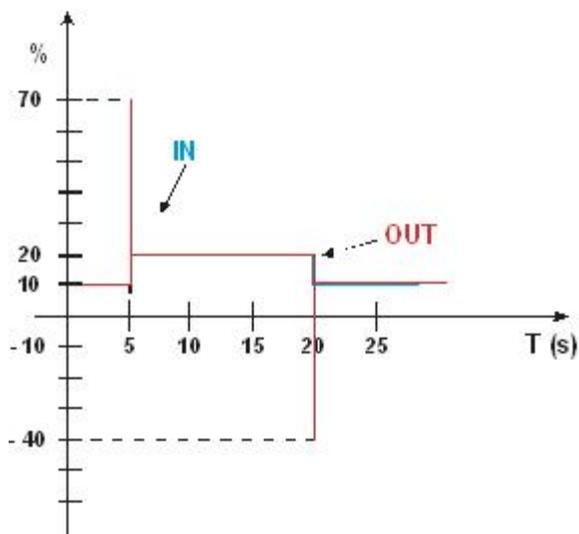
Exemplo de Aplicação:

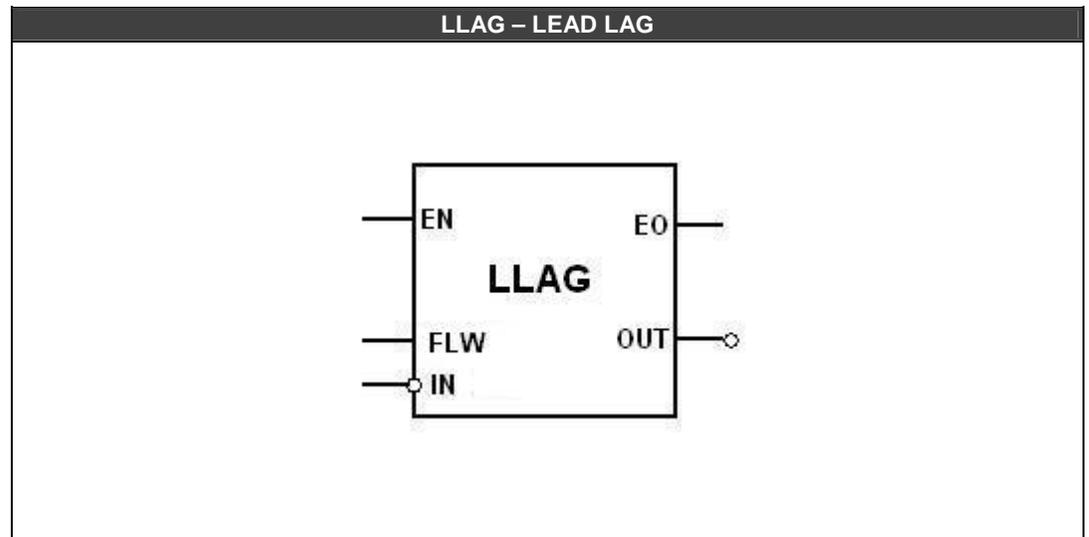
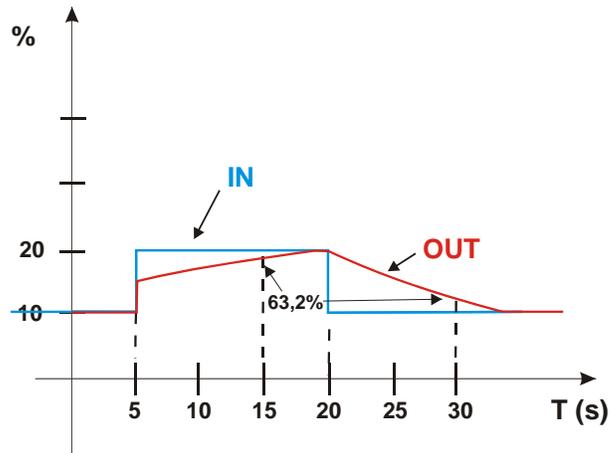
Considerando, inicialmente, um sinal de entrada $IN = 10$. A entrada sofre um degrau positivo de 10% em $t = 5$ s. Em $t = 20$ s sofre um degrau negativo de 10%. O funcionamento do bloco **LLAG** pode ser observado para os seguintes casos:

1) $K_2 = 0$ e $K_1 = 5$



2) $K_2 = 5$ e $K_1 = 0$



3) $K2= 5$ e $K1 = 10$ 

| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | VALOR DE ENTRADA | FLOAT |
| | FLW | SAÍDA SEGUE A ENTRADA | BOOL |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | FLOAT |
| P | K1 | TEMPO CARACTERÍSTICO DO FILTRO EM SEGUNDOS. É UM FILTRO EXPONENCIAL DE PRIMEIRA ORDEM (EM SEGUNDOS). | FLOAT |
| | K2 | CONSTANTE DE TEMPO LEAD (EM SEGUNDOS) | FLOAT |
| | DER | INDICA O TIPO DE ATUAÇÃO DO BLOCO (FALSE: LEAD LAG, TRUE: DERIVATIVO) | BOOL |

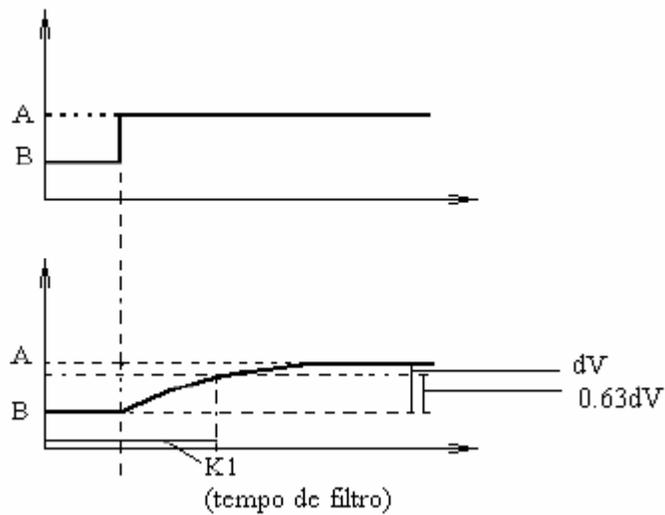
Equação Matemática para Processamento de Sinais (MATH)

Descrição

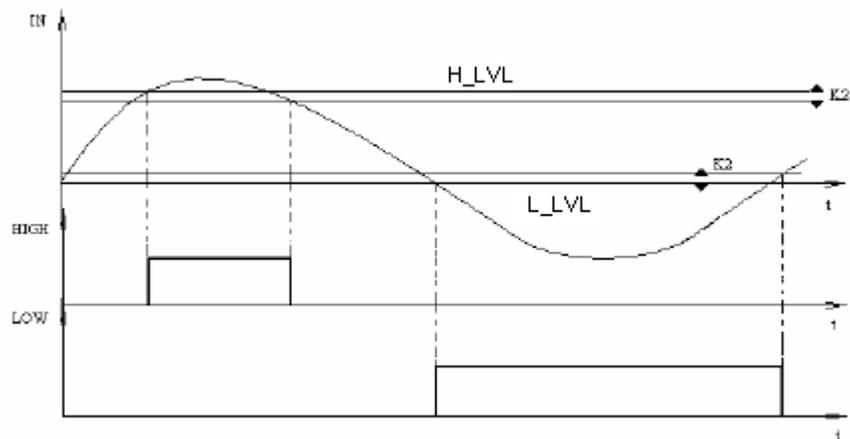
Este bloco funcional, quando a entrada **EN** está verdadeira utiliza uma equação que filtra o sinal de entrada. O filtro é exponencial de primeira ordem. A entrada **IN** recebe este sinal de entrada.

Tempo característico do filtro (K1)

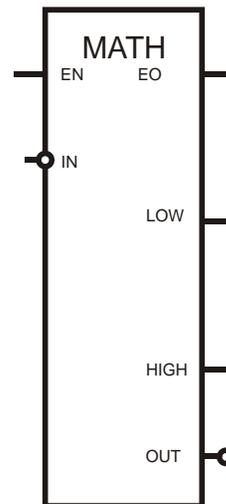
K1 é o tempo característico do filtro em segundos. Considere uma entrada degrau, quando o valor da saída atingir 63 % do valor do degrau, o tempo decorrido até este instante é definido como tempo característico do filtro.



Histerese K2 e alarmes HIGH e LOW



Quando a entrada atingir o valor configurado em H_LVL, a saída **HIGH** irá para nível alto até que a entrada ultrapasse ($H_LVL - K2$). De maneira semelhante, quando a entrada atingir o valor L_LVL, a saída **LOW** vai para nível alto até a que entrada ultrapasse o valor ($L_LVL + K2$).

MATH – EQUAÇÃO MATEMÁTICA PARA PROCESSAMENTO DE SINAIS

| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|---|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | IN | SINAL A SER PROCESSADO | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | LOW | ALARME DE LIMITE INFERIOR | BOOL |
| | HIGH | ALARME DE LIMITE SUPERIOR | BOOL |
| | OUT | SAÍDA APÓS CÁLCULO DO FILTRO | FLOAT |
| P | K1 | TEMPO CARACTERÍSTICO DO FILTRO EM SEGUNDOS. É UM FILTRO EXPONENCIAL DE PRIMEIRA ORDEM. | FLOAT |
| | K2 | HISTERESE PARA O PROCESSAMENTO O PROCESSAMENTO DE ALARME HIGH E LOW. DEVERÁ SER UMA VÁLVULA NÃO NEGATIVA. | FLOAT |
| | L_LVL | LIMITE INFERIOR PARA O PROCESSAMENTO DE ALARME APÓS O FILTRO DIGITAL. | FLOAT |
| | H_LVL | LIMITE SUPERIOR PARA O PROCESSAMENTO DE ALARME APÓS O FILTRO DIGITAL. | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Controlador PID

Descrição:

Este bloco funcional, quando a entrada **EN** está verdadeira realiza o controle PID. O consagrado algoritmo PID para controle de processos contínuos, associado à flexibilidade de configuração de características de operação através de parametrização, permite a utilização deste bloco à uma grande variedade de aplicações e estratégias de controle.

Este bloco oferece várias opções de configuração do algoritmo de controle tendo como base os termos Proporcional(P), Integral(I) e Derivativo(D), que podem ser aplicados ao erro ou apenas à variável de processo (PV).

Os limites de anti-reset windup (limites aplicados apenas ao termo integral) podem ser configurados pelo usuário. Além disto, tem-se opção de seleção entre: algoritmo ISA ou paralelo, ação direta ou reversa, transferência de controle de manual para automático, bumpless ou hard.

Seleção do formato das entradas SP, PV e FB e da saída OUT (parâmetro PERC)

PERC = falso: os valores das entradas **SP**, **PV** e **FB** e da saída **OUT** são dados em porcentagem (0 – 100%).

PERC = verdadeiro: os valores da entrada **SP**, **PV** e **FB** e da saída **OUT** são dados no formato 0 - 10000.

Tipo de PID

É definido pelo parâmetro **PID**.

PID = 0: tipo PI.D.

PID = 1: tipo PID.

PID = 2: tipo I.PD.

PI.D : As ações P e I atuam sobre o erro e a ação D sobre a variável de processo. Desta forma, o sinal de saída acompanha as mudanças de set point segundo as ações proporcional e integral, mas não dá uma variação indesejável devido à ação derivativa. É o mais recomendado para a maioria das aplicações com o set point ajustável pelo operador.

PID: As ações P, I e D atuam sobre o erro. Desta forma o sinal de saída é alterado quando há mudanças na variável de processo ou no set point. É recomendado para controle de relação ou para controle escravo de um cascata.

I.PD: Neste tipo somente a integral atua sobre o erro. Mudanças no set point provocam a variação no sinal de saída de maneira suave. É recomendado para processos que não podem ter variações bruscas na variável em função da mudança no set point. É o caso de processos de aquecimento com ganho muito alto.

Tipo de Algoritmo

É definido pelo parâmetro **ALG**.

ALG = falso: algoritmo paralelo

ALG = verdadeiro: algoritmo ISA

$$PARALELO : MV(t) = K_p e(t) + \frac{1}{T_R} \int e(t) dt + T_D \frac{de(t)}{dt}$$

$$ISA : MV(t) = K_p [e(t) + \frac{1}{T_R} \int e(t) dt + T_D \frac{de(t)}{dt}]$$

Tipo de Ação

Existem processos que requerem que o sinal de saída (variável manipulada – MV) aumente quando a variável de processo aumenta, enquanto que outros requerem o contrário.

| PARÂMETRO | TIPO DE AÇÃO | ERRO | EFEITO |
|------------------|--------------|---------------|---------------------------------|
| ACT = falso | Reversa | $e = SP - PV$ | Saída diminui com aumento da PV |
| ACT = verdadeiro | Direta | $e = PV - SP$ | Saída aumenta com aumento da PV |

Entrada TRF

Se **TRF** for verdadeiro, o PID estará em controle manual. Se **TRF** for falso o PID estará em controle automático.

Tipo de transferência de Manual para Automático

É definido pelo parâmetro **TRS**.

TRS = 0: bumpless & entrada **FB** não conectada.

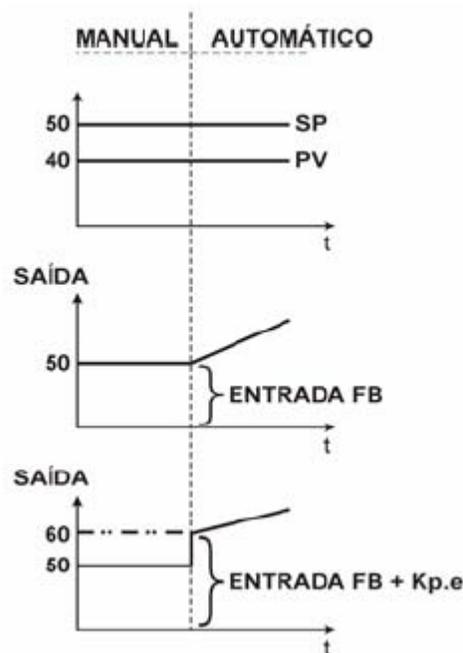
TRS = 1: bumpless & entrada **FB** conectada.

TRS = 2: hard & entrada **FB** não conectada.

TRS = 3: hard & entrada **FB** conectada.

Bumpless : No chaveamento do modo manual para o automático, o bloco PID inicia os cálculos partindo do último valor em manual, isto é, não ocorre um salto na saída do bloco.

Hard : No chaveamento do modo manual para o automático, o bloco PID fornecerá como primeiro valor em automático o último valor em modo manual mais o termo proporcional.

**Anti-saturação pelo termo integral (parâmetros AWL e AWU)**

Usualmente o algoritmo de controle pára automaticamente a contribuição do modo integral, quando o sinal de saída atinge os limites de 0% ou 100%. As contribuições dos modos proporcional e derivativo não são afetadas.

Uma característica diferenciadora do algoritmo deste bloco é a possibilidade de configuração desses limites. Estreitando-se tais limites através dos parâmetros **AWL** e **AWU**, obtém-se respostas mais rápidas e evita-se *overshoot* em processos de aquecimento, por exemplo.

Constantes do PID (parâmetros KP, TR, TD e BIAS)

KP – Ganho proporcional.

TR – Tempo da integral em minutos/repetição, portanto, quanto maior este parâmetro menor é a ação integral. Pode ser interpretado como sendo o tempo necessário para a saída ser incrementada/decrementada do valor do erro (no PID paralelo), mantendo-se o mesmo constante.

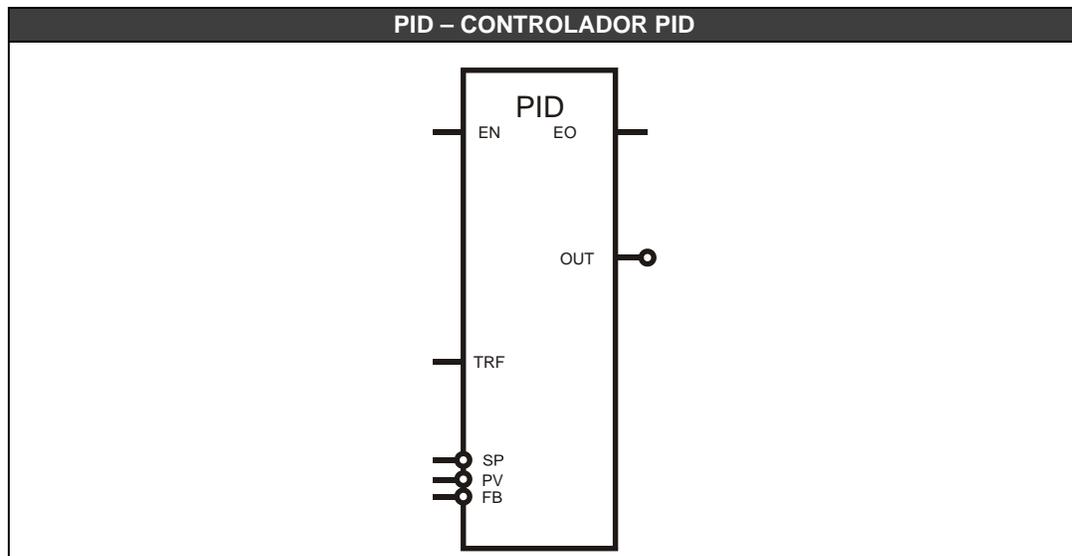
TD – Tempo derivativo em minutos. O termo derivativo é calculado usando uma pseudoderivada, isto é, uma ação semelhante a um *lead/lag*, na qual a constante de *lag* é $\alpha \cdot TD$. Na implementação deste bloco o fator α é igual a 0,13.

BIAS – Neste parâmetro é possível ajustar o valor inicial do sinal de saída quando o controle é transferido de manual para automático. Isto pode ser feito somente se a entrada **FB** não estiver conectada.

NOTA

BIAS, AWL e AWU são valores em porcentagem.

PID – CONTROLADOR PID



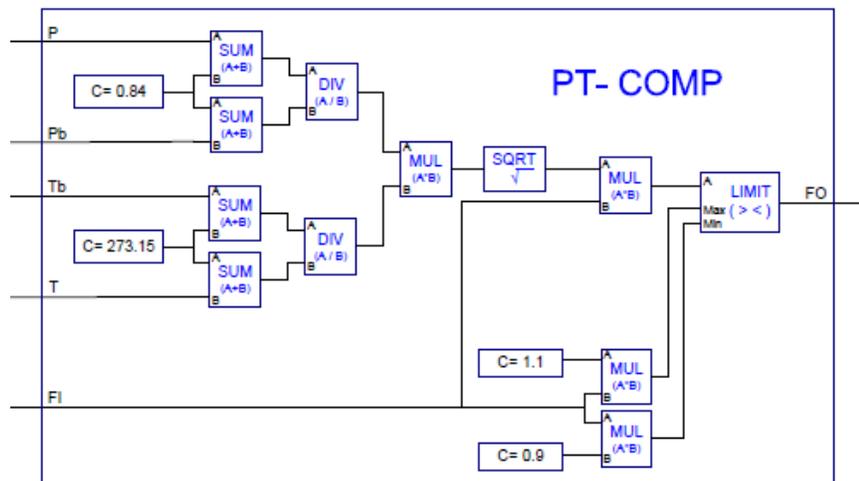
| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|----------|------|---|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | TRF | SELEÇÃO FUNCIONAMENTO MANUAL OU AUTOMÁTICO | BOOL |
| | SP | SETPOINT | FLOAT |
| | PV | VARIÁVEL DO PROCESSO | FLOAT |
| | FB | SE TRF VERDADEIRO, A ENTRADA CONECTADA EM FB É REPASSADA ATÉ A SAÍDA | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA (VARIÁVEL MANIPULADA) | FLOAT |
| P | KP | GANHO PROPORCIONAL | FLOAT |
| | BIAS | BIAS | FLOAT |
| | AWL | LIMITE INFERIOR FINAL DO ANTI-RESET | FLOAT |
| | AWU | LIMITE SUPERIOR FINAL DO ANTI-RESET | FLOAT |
| | TR | TEMPO INTEGRATIVO (Min/Rep) | FLOAT |
| | TD | TERMO CONSTANTE DERIVATIVO (Min) | FLOAT |
| | PERC | SELEÇÃO DO FORMATO DAS ENTRADAS E DA SAÍDA ENTRE "0 - 10000" E "0 - 100%" | BOOL |
| | TRS | DEFINE O TIPO DE TRANSFERÊNCIA DE AUTOMÁTICO PARA MANUAL | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Compensação de Temperatura e Pressão (PTC)

Descrição:

Este bloco funcional, quando a entrada **EN** está verdadeira, realiza a compensação de pressão e temperatura definida pelos diagrama em blocos e equação a seguir:



$$FO = \text{LIMIT} [\text{SQRT}((P + 0.84) / (Pb + 0.84) * (Tb + 273.15) / (T + 273.15)) * FI]$$

FI = Measured Flow Rate
 P = Measured Pressure [Bar G]
 T = Measured Temperature (Deg C)
 FO = Compensated Flow Rate
 Pb = Base Pressure [Bar G] : To be defined per each case
 Tb = Base Temperature (Deg C) : To be defined per each case



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | P | | FLOAT |
| | Pb | | FLOAT |
| | T | | FLOAT |
| | Tb | | FLOAT |
| | T | ENTRADA | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | FO | SAÍDA | FLOAT |
| P | K1 | CONSTANTE | FLOAT |
| | K2 | CONSTANTE | FLOAT |
| | K3 | CONSTANTE | FLOAT |
| | K4 | CONSTANTE | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Sample Hold com Incremento e Decremento (SMPL)

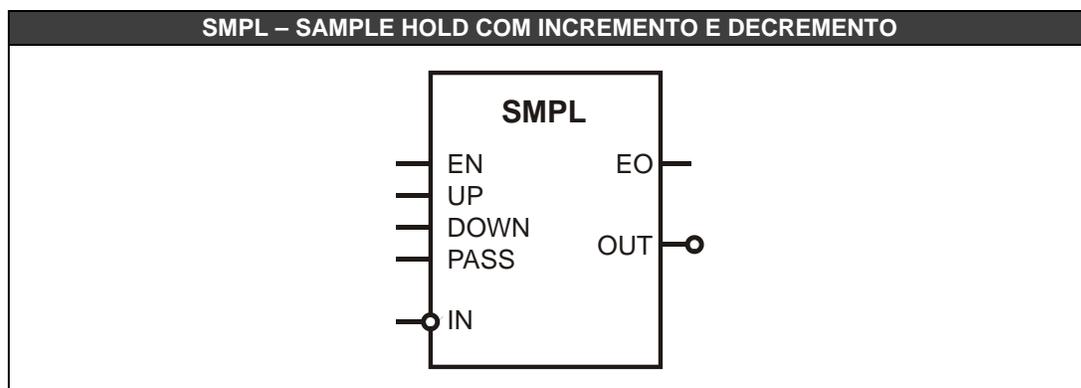
Descrição:

Este bloco funcional, quando a entrada **EN** está verdadeira amostra o valor na entrada **IN** e o coloca no REGISTRADOR quando a entrada **PASS** muda de verdadeiro para falso. O valor do REGISTRADOR pode ser incrementado ou decrementado usando a entrada **UP** ou **DOWN**. A velocidade deste incremento ou decremento é definida pelo parâmetro **ASPD**. Este bloco pode ser utilizado em conjunto com um bloco PID.

Seleção do formato da entrada IN e da saída OUT (parâmetro PERC)

PERC = falso: os valores da entrada **IN** e da saída **OUT** são dados em porcentagem (0 – 100%).

PERC = verdadeiro: os valores da entrada **IN** e da saída **OUT** são dados no formato 0 - 10000.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|---|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | UP | INCREMENTAR O CONTADOR | BOOL |
| | DOWN | DECREMENTAR O CONTADOR | BOOL |
| | PASS | COLOCA O VALOR DO REGISTRADOR NA SAÍDA | BOOL |
| | IN | ENTRADA | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | OUT | SAÍDA | FLOAT |
| P | ACCEL | FATOR DE ACELERAÇÃO – INCREMENTO E DECREMENTO | LONG |
| | ASPD | VELOCIDADE DE ATUAÇÃO EM % POR SEGUNDO | LONG |
| | L_LMT | LIMITE INFERIOR | FLOAT |
| | H_LMT | LIMITE SUPERIOR | FLOAT |
| | PERC | SELEÇÃO DO FORMATO DA ENTRADA E DA SAÍDA ENTRE "0 - 10000" E "0 - 100%" | BOOL |

I: Entrada **O:** Saída **P:** Parâmetro Interno

Gerador de Set Point (SPG)

Descrição:

O bloco Gerador de Set Point, **SPG**, é normalmente usado para gerar set point para o bloco PID em aplicações como controle de temperatura, reatores de batelada, etc. Nessas aplicações, o set point deve seguir um certo modelo em função do tempo.

Quando a entrada **EN** é verdadeira, o algoritmo é habilitado.

A curva é determinada por dez segmentos ou passos. Cada segmento é definido por um valor inicial [**VALx**] e um tempo de duração [**DURx**]. O valor inicial do próximo segmento determina se o segmento anterior aumenta, diminui ou mantém-se constante. A curva é dada por:

VALx (Valor Inicial) – Formado por um arranjo de até 11 pontos analógicos que definem o valor inicial de cada segmento, em Unidades de Engenharia.

DURx (Tempo de Duração) – Formado por um arranjo de até 10 pontos analógicos que definem a duração, em segundos, de cada segmento. Um valor zero define o último segmento.

Os dois arranjos definem o valor de set point (eixo y) em função do tempo (eixo t). Entre dois pontos dados, o set point é calculado por interpolação. Como cada segmento é definido por [**VALx**]_i, [**DURx**]_i e [**VALx**]_{i+1}, um modelo com "n" segmentos necessitará **n+1** valores iniciais e **n** tempos de duração. Como exemplo, os dois arranjos definem a curva mostrada no gráfico abaixo.

| | 1 | 2 | 3 | 4 | 5 | 6 |
|------|----|----|-----|-----|-----|----|
| VALx | 25 | 50 | 50 | 100 | 100 | 25 |
| DURx | 60 | 60 | 120 | 60 | 60 | 0 |

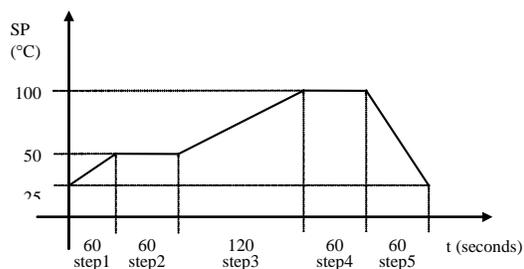


Fig.1 –Curva de Setpoint

A posição no eixo do tempo (eixo t) é controlado por um temporizador interno. O temporizador é iniciado por uma transição de falso para verdadeiro na entrada **STR**. Uma vez iniciado, ele executa até atingir a soma das durações determinadas no parâmetro **DURx**.

O temporizador é zerado (ou seja, posicionado no início da curva), se a entrada **RESET** for igual a 1. Após o reset, é esperado um novo *start* para reiniciar o temporizador. Quando é utilizada a entrada **RESET**, enquanto ela estiver com o valor 1 o bloco permanecerá em *reset*, portanto, estará disponível para iniciar somente depois que este parâmetro for para 0.

Toda vez que ocorre o disparo do SPG via **STR**, obrigatoriamente, para se fazer um novo disparo, é necessário primeiramente que ocorra um **RESET**.

O temporizador pode ser interrompido por uma mudança no sinal discreto **PAUSE** de falso para verdadeiro. Recomeçará a executar quando **PAUSE** for configurado para falso e nenhuma outra condição interromper o temporizador.

O temporizador é também interrompido por um **PAUSE** provocado por um desvio entre a entrada **BKIN** e o set point gerado ($DES\text{VIO} = BKIN - OUT$). Sendo que o desvio é programado em **ADEV** (em porcentagem %). O temporizador é parado e retoma à operação normal quando o desvio está dentro do limites pré-escritos. Se o valor de **ADEV** for 0%, ele não é considerado. Um outro modo deste valor não ser levado em conta é conectar a saída **OUT** à entrada **BKIN**.

Em ambos os casos em que o temporizador está em **PAUSE**, a saída **P_ST** é levada para valor 1. O set point está na coordenada “y”, enquanto o tempo está na coordenada “t”. O valor do set point está disponível para a saída **OUT**.

Três saídas informam o ponto atual da curva:

ST_PS – Informa o segmento atual ou passo.

TM_PN – Informa o tempo decorrido desde o início do passo atual.

TM_PT - Informa o tempo decorrido desde o início da curva.

Ao atingir o final do ciclo, a saída **END** vai para 1, permanecendo neste estado até que o temporizador seja reiniciado.

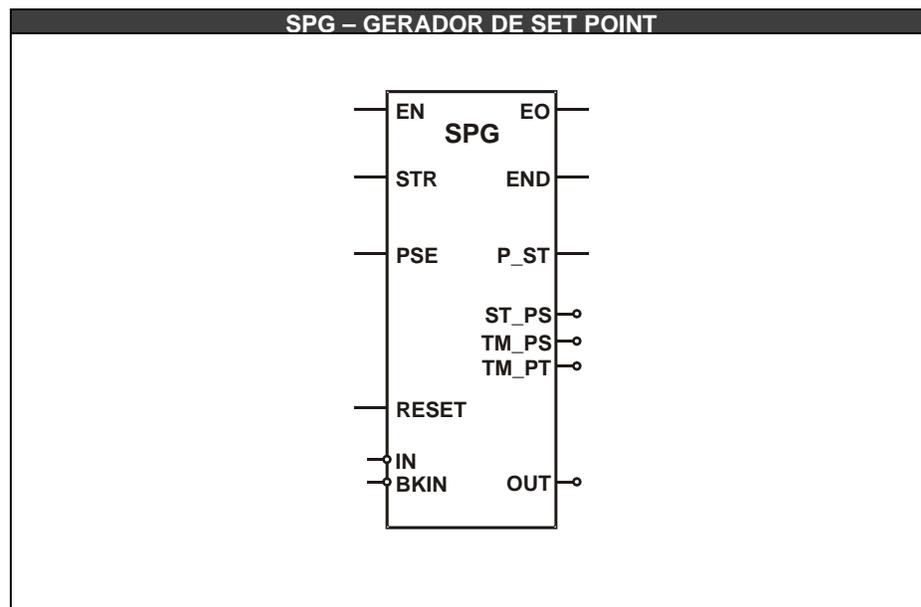
O parâmetro **AT_CY**, estando em verdadeiro, faz com que o ciclo seja repetido continuamente, até que o temporizador entre no estado de *reset* ou pausa.

Através do parâmetro **TIME** é configurada a escala de tempo em segundos, minutos ou horas. Através do parâmetro **TRK**, é indicado ao temporizador se ele começará do início da tabela (None), de um valor específico de tempo (**Time**) ou de um valor específico em unidade de engenharia (**Value** ou **Segm**).

No caso de **Value**, iniciará do primeiro ponto encontrado na curva, independente do segmento em que esteja. Já no caso de **Segm** iniciará de um valor específico em unidade de engenharia, dentro de um segmento específico se este foi definido no parâmetro **SEGM**. No caso de configurado **Time**, **Value** ou **Segm**, o valor inicial é definido na entrada **IN**.

Sendo configurado **Time**, em **IN** estará o tempo (valor inicial) em que começará a geração. Se for configurado **Value** ou **Segm**, em **IN** estará o valor inicial da curva em que começará a geração, sendo que no caso de **Segm**, também será levado em conta em qual segmento estará o valor inicial, que é configurado no parâmetro **SEGM**.

Ou seja, em **IN** estará sempre o valor inicial, que pode ser um tempo ou um valor em unidade de engenharia. E se **Time**, **Value** ou **Segm** não for configurado, estarão **None**, a geração começará no tempo = 0 s.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|---|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | STR | INICIA O TEMPORARIZADOR | BOOL |
| | PSE | PAUSA O TEMPORARIZADOR | BOOL |
| | RESET | RESETA O TEMPORARIZADOR | BOOL |
| | IN | ENTRADA DO BLOCO PARA DEFINIR O INÍCIO DA CURVA | FLOAT |
| O | BKIN | VALOR DE ENTRADA COMPARADO COM A SAÍDA PARA CÁLCULO DO DESVIO | FLOAT |
| | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | END | INDICA FINAL DO CICLO | BOOL |
| | P_ST | INDICA SE O ALGORITMO ESTÁ EM PAUSE | BOOL |
| | ST_PS | SEGMENTO ATUAL OU PASSO | FLOAT |
| | TM_PS | TEMPO DECORRIDO DESDE O INÍCIO DO PASSO ATUAL (SEGUNDOS) | FLOAT |
| | TM_PT | TEMPO DECORRIDO DESDE O INÍCIO DA CURVA (SEGUNDOS) | FLOAT |
| | OUT | SAÍDA DO BLOCO | FLOAT |
| P | VAL1 | VALOR INICIAL 1 | FLOAT |
| | VAL2 | VALOR INICIAL 2 | FLOAT |
| | : | | |
| | VAL10 | VALOR INICIAL 10 | FLOAT |
| | VAL11 | VALOR FINAL | FLOAT |
| | DUR1 | TEMPO DO PRIMEIRO SEGMENTO | FLOAT |
| | DUR2 | TEMPO DO SEGUNDO SEGMENTO | FLOAT |
| | : | | |
| | DUR10 | TEMPO DO DÉCIMO SEGMENTO | FLOAT |
| | AT_CY | CICLO CONTÍNUO | BOOL |
| | TIME | ESCALA DE TEMPO | LISTA |
| | TRK | TIPO DE INÍCIO DA CURVA | LISTA |
| | SEGM | DEFINE EM QUAL SEGMENTO A CURVA INICIARÁ | LONG |
| | ADEV | DESVIO MÁXIMO PERMITIDO (PORCENTAGEM) | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Controle Step (STP)

Descrição:

Este bloco funcional é usado em combinação com o bloco PID. A saída do PID é conectada à entrada **DMV** para executar um controle **ON_OFF** ou **ON_NONE_OFF**. O controle **ON_OFF** estabelece um controle de abertura e fechamento de válvulas durante um intervalo de tempo predefinido. O controle **ON_NONE_OFF** proporciona o controle da abertura ou fechamento das válvulas levando em conta a taxa de variação da saída do PID ou a entrada **DMV**.

Seleção do formato da entrada DMV (parâmetro PERC)

PERC = falso: o valor da entrada **DMV** é dado em porcentagem (0 – 100%).

PERC = verdadeiro: o valor da entrada **DMV** é dado no formato 0 - 10000.

Tempo de Abertura das Válvulas (VOT)

Este parâmetro deve ser ajustado com o tempo aproximado necessário para a válvula ir de totalmente fechada para totalmente aberta.

Largura de pulso mínima (WPL)

O usuário deverá configurar a largura deste pulso mínima por 0,1s no parâmetro **WPL** e o tempo para excursão total do elemento de controle.

Tipo de Controle (CTRL)

O usuário deverá selecionar o tipo de controle, isto é, **ON_OFF** ou **ON_None_OFF**.

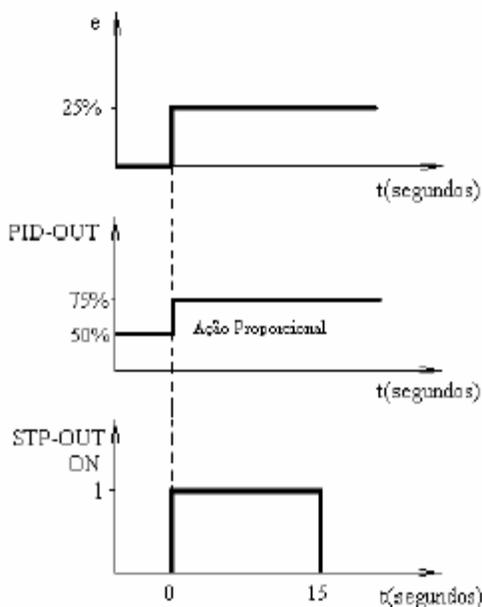
- Controle **ON_OFF** (CTRL = 1)

Neste modo de controle, o bloco compara a entrada **DMV** com os parâmetros **ON_T** e **OFF_T**:

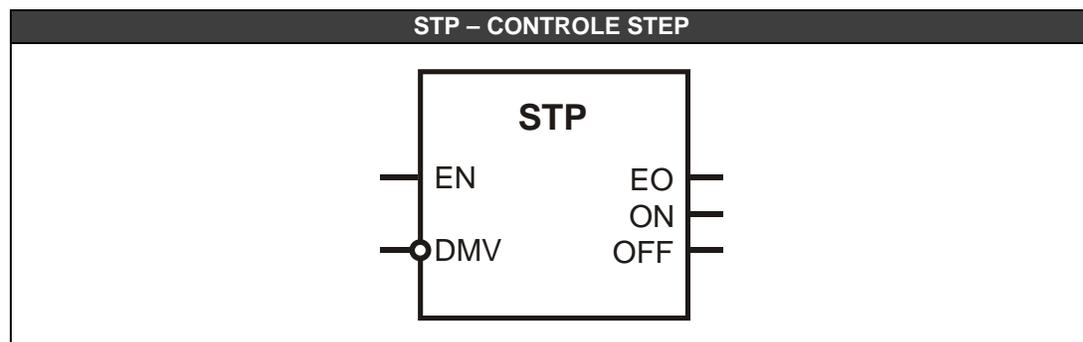
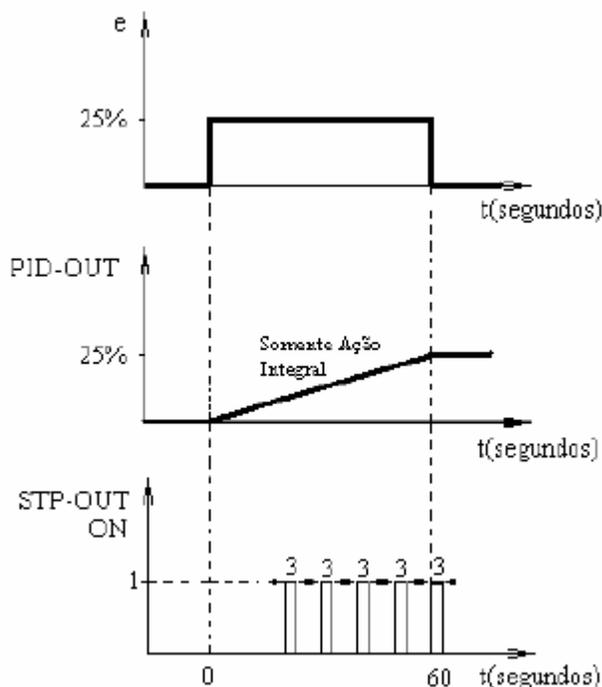
- se a entrada **DMV** for maior que **ON_T**, a saída **ON** ficará em nível alto e a saída **OFF** em nível baixo.
- se a entrada **DMV** for menor que **OFF_T**, a saída **ON** ficará em nível baixo e a saída **OFF** em nível alto.
- se a entrada **DMV** estiver entre **OFF_T** e **ON_T**, as saídas **ON** e **OFF** permanecerão no último estado.

- Controle **ON_None_OFF** (CTRL = 0)

Um PID apenas com ação proporcional com ganho $K_P=1$ e **VOT** igual a 1 minuto. Supondo que no instante $t=0$ um degrau de erro igual a 25% é aplicado. Portanto, a abertura das válvulas é de 25% de 1 minuto, isto é, $0.25 \cdot TR = 15$ segundos. A figura abaixo mostra este exemplo em maiores detalhes.



A ação integral do PID equivale a uma série de pulsos de tamanho mínimo **WPL** com frequência determinada pelo tempo integral do bloco PID (**TR**) e pelo desvio do controle. A frequência dos pulsos é dada pelo valor de **TR**. O valor de **WPL** é fixo e determinado na configuração do bloco. Supondo que **TR**= 1 minuto e que **WPL** = 3 segundos e que um degrau de erro de 25 % é aplicado na entrada. Um controlador padrão aumentaria ou diminuiria a saída em 25 % em 1 minuto (**TR**). Para fazer a válvula ter tempo de abertura (**VOT**) igual a 1 minuto são necessários 15 segundos (25% de 60 segundos), pois **WPL**= 3 segundos. Assim 5 pulsos de tamanho 3 segundos são necessários. A saída permanece neste modo de funcionamento, enquanto a saída do PID mantiver a mesma taxa de mudança.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|---|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | DMV | ENTRADA DO BLOCO | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | ON | SAÍDA NÍVEL ALTO (ABRIR) | BOOL |
| | OFF | SAÍDA NÍVEL BAIXO (FECHAR) | BOOL |
| P | VOT | TEMPO DE ABERTURA DA VÁLVULA EM 0.1s | FLOAT |
| | WPL | LARGURA DE PULSO MÍNIMA POR 0.1s | FLOAT |
| | CTRL | TIPO DE CONTROLE | LONG |
| | PERC | SELEÇÃO DO FORMATO DA ENTRADA E DA SAÍDA ENTRE "0 - 10000" E "0 - 100%" | BOOL |
| | ON_T | LIMIAR (%) PARA ACIONAR SAÍDA ON | FLOAT |
| | OFF_T | LIMIAR (%) PARA ACIONAR SAÍDA OFF | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Totalização (TOT)

Descrição:

Este bloco fornece a totalização da entrada. Esta totalização é a integral da entrada multiplicada por um fator de escala, **FCF**. Este fator de escala permite que o usuário configure a totalização em 3 modos de operação. Se a aplicação requer o cálculo instantâneo do volume totalizado, basta utilizar o bloco TOT para integrar a vazão, que é a derivada temporal do volume. A base de tempo do bloco é segundos.

Uma vazão é dada, geralmente, em unidades de engenharia (EU) por unidade de tempo. Por exemplo:

Uma vazão de 1 m³/segundo como entrada do bloco TOT terá como saída a totalização em m³. Considere que a aplicação necessite do valor da energia de um dispositivo elétrico. O bloco TOT permite que se calcule o valor desta energia através da potência instantânea. Pois:

$$Energia = \int Pot(t)dt$$

e ainda $Pot(t) = V(t) \cdot I(t)$, onde $V(t)$ é a tensão instantânea e $I(t)$ a corrente instantânea.

Saída **OUT** e parâmetro **TU**

O intervalo de tempo em que a saída é totalizada está de acordo com o valor configurado em **TU**. A integração (totalização) é mantida em um registrador interno que pode ir até 8.000.000 unidades. A saída **OUT** é o valor da totalização.

Saída **dI**

O valor máximo da totalização é de 8.000.000 e o mínimo de -8.000.000. Sempre que a saída do totalizador atingir estes valores, a saída **dI** muda de 0 para 1 durante um intervalo de tempo. A saída **dI** é portanto um contador de viradas de contagem.

Parâmetro **FCF**

O parâmetro **FCF** permite ao bloco TOT operar em 4 modos diferentes:

a) **IN** é FLOAT e representa a vazão em unidades de engenharia:

FCF deve ser igual a 1 para ter totalização sem qualquer fator de escala em unidades de engenharia (ou ajuste o fator que você quiser usar). Por exemplo:

Vazão Q é medida em m³/horas. 1 hora possui 3600 segundos. Portanto, o valor de **TU** deve ser igual a 3600. Supondo uma vazão constante de 60 m³/hora, a totalização será dada pela expressão:

$$TOT(t) = \int_0^{t(\text{segundos})} \frac{FCF}{TU} * IN(t)dt = \int_0^{t(\text{segundos})} \frac{1}{3600} * 60dt = \int_0^{t(\text{segundos})} \frac{1}{60} dt [m^3]$$

Portanto, após 1 minuto ou 1/60 horas ou 60 segundos o valor de TOT será:

$$TOT[m^3] = \int_0^{60} \frac{1}{60} dt = 1m^3$$

A cada 1/60 horas ou a cada 1 minuto o bloco totaliza a entrada e mostra este valor na saída. Pois:

60 m³ _____ 1 hora

1 m³ _____ t (intervalo de tempo em que a totalização é mostrada)

Então, t= 1/60 horas ou 1 minuto.

b) **IN** é FLOAT e representa a vazão em porcentagem:

Neste caso, a entrada será interpretada como uma porcentagem representada por um número real entre 0 e 100 (0% e 100%, respectivamente). **FCF** deve ser igual à vazão máxima em unidades de engenharia (Vazão a 100%) para ter a totalização em unidades de engenharia. A configuração do parâmetro **TU** é similar à entrada real em unidades de engenharia. A totalização será mostrada na unidade de engenharia configurada.

c) **IN** é INTEIRO:

Neste caso, a entrada será interpretada como um número inteiro entre 0 e 10000 (0% e 100.00% respectivamente). **FCF** deve ser igual à vazão máxima em unidades de engenharia dividido por 10000. Supondo uma vazão máxima de 1 m³/segundo e uma vazão de 0,5 m³/segundo. O valor de **FCF** é igual à vazão máxima dividida por 10000, isto é, 0,0001. O valor de **TU** é neste caso 1, pois a totalização é dada em m³. Uma entrada de 0,5 m³/segundo equivale a 5000 (ou 50 % da escala). Portanto:

$$OUT = \int_0^t \frac{FCF}{TU} * IN\%(t) dt = \int_0^t 0.0001 * 5000 dt = 0.5t(m^3)$$

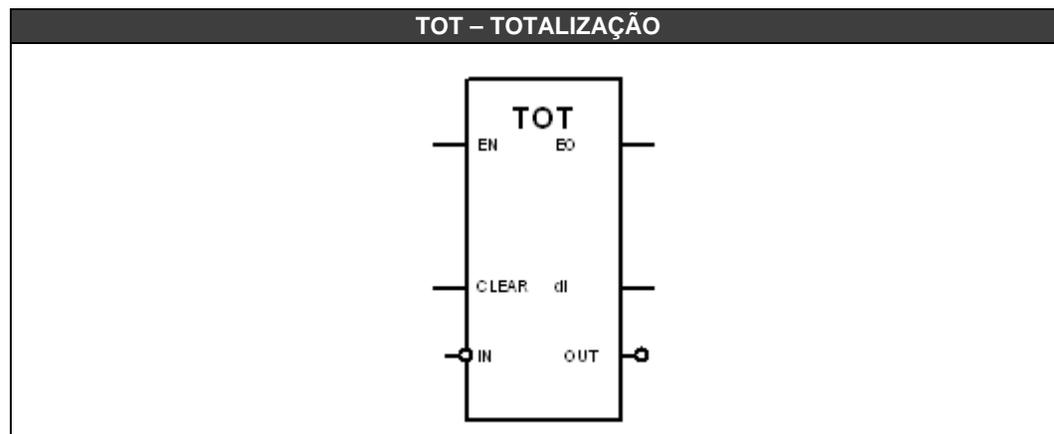
Logo, em 1 minuto (ou 60 segundos) o valor totalizado será de 30 m³.

d) Quando **FCF** é menor que zero:

Quando o bloco estiver totalizando uma vazão negativa, a totalização é decrementada, enquanto que quando a vazão é positiva a totalização é incrementada. Quando **FCF** for maior do que zero, isto é, positivo, o bloco totalizador só aceitará vazões positivas.

Entrada CLEAR

Se a entrada **CLEAR** for alterada para verdadeiro, a totalização será reiniciada e os registradores internos do bloco TOT serão zerados.



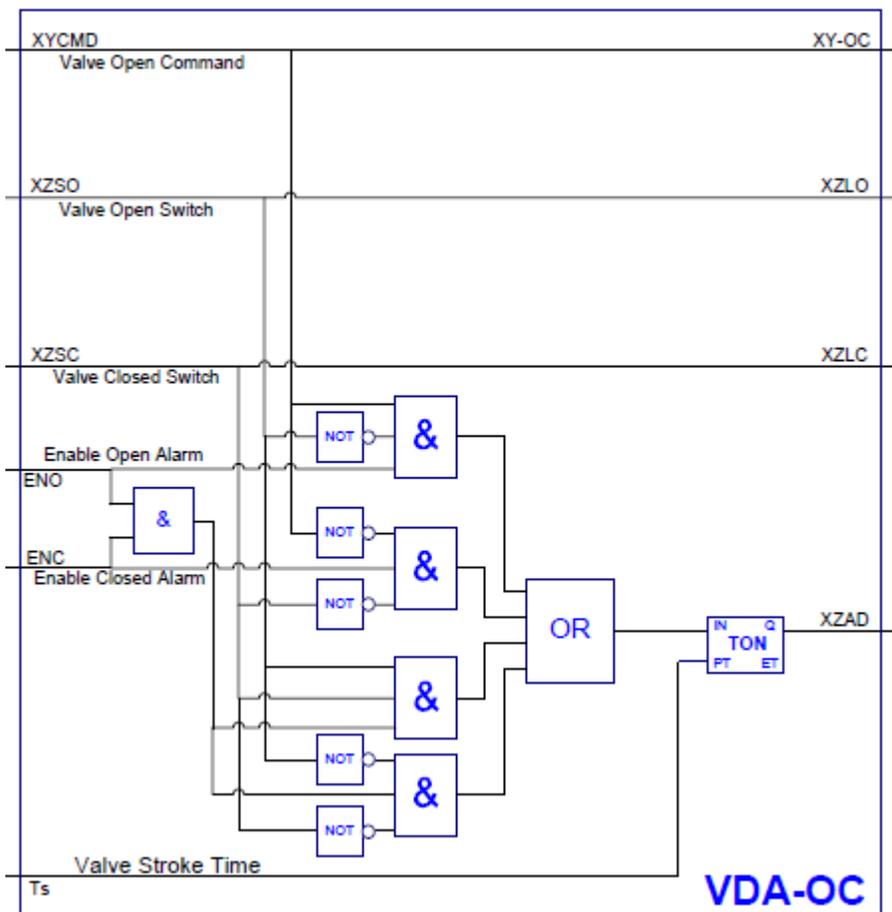
| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|--|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | CLEAR | ZERA O TOTALIZADOR | BOOL |
| | IN | ENTRADA | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | dl | ALARME QUE INDICA QUANDO A TOTALIZAÇÃO ATINGIU O VALOR -8.000.000 OU 8.000.000. NESTE CASO dl= ON. | BOOL |
| | OUT | SAÍDA TOTALIZADA | FLOAT |
| P | TU | VALOR DE TOTALIZAÇÃO PARA UMA UNIDADE DE CONTAGEM | FLOAT |
| | FCF | FATOR DE TAXA DE VAZÃO | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

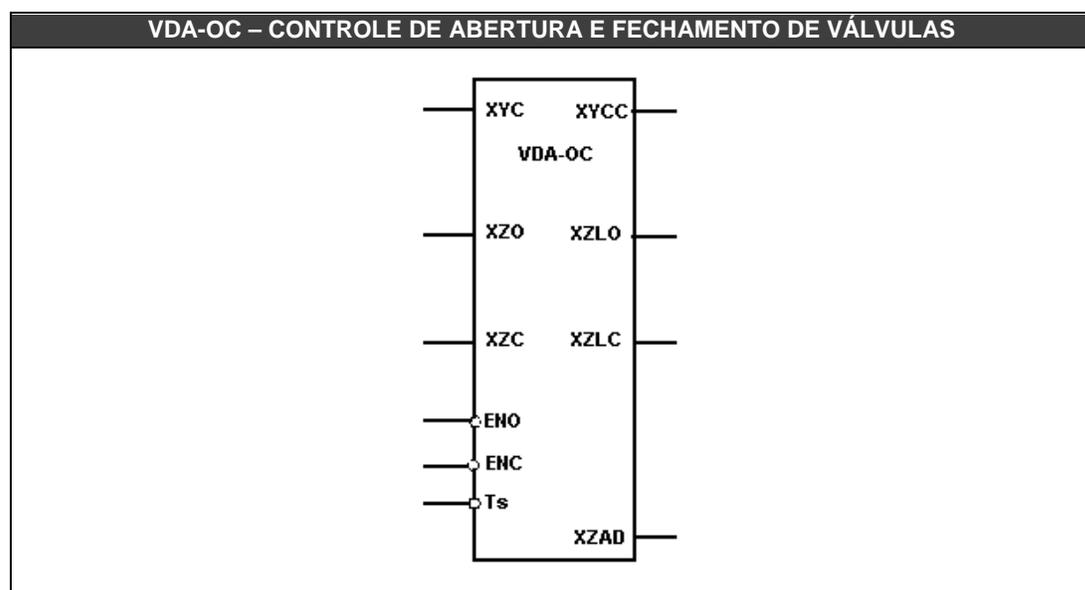
Controle de Abertura e Fechamento de Válvulas (VDA-OC)

Descrição:

Este bloco funcional, realiza a compensação o controle de abertura e fechamento de válvulas, definida pelo diagrama em blocos a seguir:



VDA-OC – CONTROLE DE ABERTURA E FECHAMENTO DE VÁLVULAS



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------------|-------|
| I | XYC | HABILITAÇÃO DA ENTRADA | BOOL |
| | XZO | | BOOL |
| | XZC | | BOOL |
| | ENO | | BOOL |
| | ENC | | BOOL |
| | Ts | ENTRADA PARA O TIMER INTERNO | LONG |
| O | XYCC | HABILITAÇÃO DA SAÍDA | BOOL |
| | XZLO | | BOOL |
| | XZLC | | BOOL |
| | XZAD | SAÍDA | FLOAT |
| P | K1 | CONSTANTE | FLOAT |
| | K2 | CONSTANTE | FLOAT |
| | K3 | CONSTANTE | FLOAT |
| | K4 | CONSTANTE | FLOAT |

I: Entrada O: Saída P: Parâmetro Interno

Limite Cruzado e Velocidade de Variação (XLIM)

Descrição:

Esta função limita um sinal dentro de valores estáticos ou dinâmicos e também controla sua velocidade de variação. A saída **OUT** é o resultado filtrado da entrada **A**.

Seleção do formato das entradas **A** e **B** e da saída **OUT** (parâmetro **PERC**)

PERC = falso: os valores das entradas **A** e **B** e da saída **OUT** são dados em porcentagem (0 - 100%).

PERC = verdadeiro: os valores das entradas **A** e **B** e da saída **OUT** são dados no formato 0 - 10000.

Limitação estática e dinâmica

- Estática

Para limitar estaticamente um sinal, desconecta-se a entrada **B**. O sinal **A** é limitado entre **BL** e **BH** (ajustados pelo usuário).

- Dinâmica

Se a entrada **B** for conectada é possível limitar dinamicamente a entrada **A** através da entrada **B**. Para maior flexibilidade, os limites são modificáveis com ganhos e bias (deslocamentos) individuais.

Rate of Change (parâmetro **MODE**)

O limite de velocidade de variação pode ser aplicado nos dois sentidos, aumentando ou diminuindo, ou para uma direção específica. Existem 4 tipos selecionáveis:

MODE = 0 : nenhum sentido é verificado.

MODE = 1: verifica apenas o sentido negativo.

MODE = 2: verifica apenas o sentido positivo.

MODE = 4: verifica ambos os sentidos.

Parâmetros **BL** e **BH**

Se $A \leq BL$ a saída **OUT** será igual a **BL**.

Se $BL < A < BH$ a saída **OUT** será igual a **A**.

Se $A \geq BH$ a saída **OUT** será igual a **BH**.

Parâmetros **GH** e **GL**

Se $A \leq B \cdot GL + BL$, a saída **OUT** será igual **B.GL+BL**.

Se $B \cdot GL + BL < A < B \cdot GH + BH$, a saída **OUT** será igual a **A**.

Se $A \geq B \cdot GH + BH$, a saída **OUT** será igual a **B.GH+BH**.

Parâmetro **DB** e as saídas **LOW** e **HIGH**

O bloco possui saídas para indicar se os limites inferior (**LOW**) e superior (**HIGH**) foram alcançados. O parâmetro **DB** pode ser ajustado para gerar uma histerese, evitando que a saída oscile quando a variável esta próxima do valor limite.

Parâmetro **RAT** e a saída **ROC**

A saída **ROC** vai para nível lógico 1 quando a velocidade de variação do sinal alcançar o valor configurado no valor **RAT**. Quando a entrada **A** muda mais rápido do que **RAT**, a variação na saída é mantida dentro do valor fixado por **RAT** até que o sinal de entrada **A** fique abaixo deste valor. O alarme **ROC** neste período está em nível alto.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | A | ENTRADA A | FLOAT |
| | B | ENTRADA B | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | HIGH | ALARME DE LIMITE SUPERIOR | BOOL |
| | LOW | ALARME DE LIMITE INFERIOR | BOOL |
| | ROC | ALARME DA TAXA DE ATUAÇÃO | BOOL |
| | OUT | SAÍDA | FLOAT |
| P | MODE | VERIFICA AMBOS SENTIDOS/APENAS SENTIDO POSITIVO/APENAS SENTIDO NEGATIVO/NENHUM | LONG |
| | GL | LIMITE DE GANHO INFERIOR | FLOAT |
| | BL | LIMITE DE BIAS INFERIOR | FLOAT |
| | GH | LIMITE DE GANHO ALTO | FLOAT |
| | BH | LIMITE DE BIAS ALTO | FLOAT |
| | DB | ZONA MORTA (HISTERESE) EM % | FLOAT |
| | RAT | VELOCIDADE DE VARIAÇÃO MÁXIMA EM % POR SEGUNDO | FLOAT |
| | PERC | SELEÇÃO DO FORMATO DAS ENTRADAS E DA SAÍDA ENTRE "0 - 10000" E "0 - 100%" | BOOL |

I: Entrada O: Saída P: Parâmetro Interno

Funções de Entradas e Saídas

Acumulador de Pulsos (ACC)

Descrição

O bloco do acumulador de pulsos trabalha junto com os módulos DF41, DF42 e DF67 (Módulos de entradas de pulsos) acumulando os pulsos de entrada provenientes de uma fonte externa. A entrada de pulsos é configurada no parâmetro de entrada **IN**.

IMPORTANTE

O parâmetro **IN** deve ser configurado obrigatoriamente com o ponto específico do slot em que o módulo está inserido. A regra para o preenchimento é **RRSGP** sendo RR: rack, S: slot, G: grupo (0 ou 1) e P: ponto (0 a 7). Exemplos:

- 214 – Rack 0, slot 2, grupo 1 e ponto 4.
- 12307 – Rack 12, slot 3, grupo 0 e ponto 7.

Com a entrada **EN** em nível lógico 1, a cada ciclo de controle, são lidos os pulsos acumulados no módulo e acrescentados ao acumulador **TOT**. Após a leitura de pulsos, o acumulador de pulsos do módulo é zerado e é iniciada uma nova contagem. Para zerar o contador de pulsos **TOT** é necessário uma transição de subida (nível lógico 0 para nível lógico 1) na entrada **CLRA**.

A saída Q

A saída **Q** é utilizada como um indicador de vazão, em que o tempo base é configurado no parâmetro **MP** (configuração do módulo). A saída **Q** será atualizada com o número de pulsos acumulados a cada intervalo de tempo **MP**.

A saída MEM

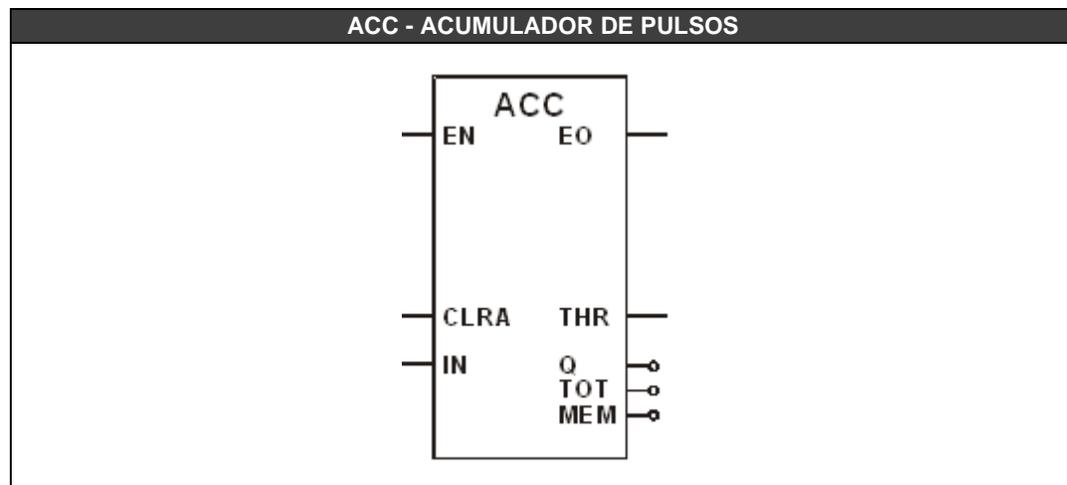
A saída **MEM** é atualizada com o número de pulsos acumulados em **TOT** após o reset dos contadores, ou seja, em uma transição de subida na entrada **CLRA**, o contador **TOT** é zerado e seu valor atual é atribuído à saída **MEM**.

A saída THR

A saída **THR** (Threshold) é controlada pelos parâmetros **TR_ON** e **TR_OFF** (parâmetros de configuração do módulo – DF41, DF42 ou DF67). A cada período de tempo **MP** é verificado se o número de pulsos acumulados é superior a **TR_ON** ou inferior a **TR_OFF**. Caso seja superior a **TR_ON** a saída **THR** será colocada em nível lógico 1 e somente será colocada em nível lógico 0 se o valor acumulado for inferior a **TR_OFF**.

Modo Acumulador

O bloco funcional ACC acumula os pulsos no registrador **TOT**. A contagem de **TOT** está dentro do intervalo 0 a $(2^{32} - 1)$



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|---------------------------------|------|
| I | EN | HABILITAÇÃO DE ENTRADA | BOOL |
| | CLRA | APAGA O ACUMULADOR TOT | BOOL |
| O | EO | HABILITAÇÃO DE SAÍDA | BOOL |
| | THR | SAÍDA THR | BOOL |
| | Q | PULSOS ACUMULADOS NO PERÍODO MP | LONG |
| | TOT | VALOR DO ACUMULADOR DE PULSOS | LONG |
| P | MEM | VALOR DO ACUMULADOR DA MEMÓRIA | LONG |
| | IN | CANAL | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Acumulador de Pulsos Múltiplo (ACC_N)

Descrição

Este bloco funcional é semelhante ao bloco anterior, exceto pelo número de entradas e acumuladores de pulso e também por não haver indicação de vazão. Ele pode trabalhar com até quatro entradas de pulsos, configuradas nos parâmetros **IN1**, **IN2**, **IN3** e **IN4**.

IMPORTANTE

Os parâmetros **IN1** a **IN4** devem ser configurados obrigatoriamente com os pontos específicos do slot em que o módulo está inserido. A regra para o preenchimento é **RRSGP** sendo RR: rack, S: slot, G: grupo (0 ou 1) e P: ponto (0 a 7). Exemplos:

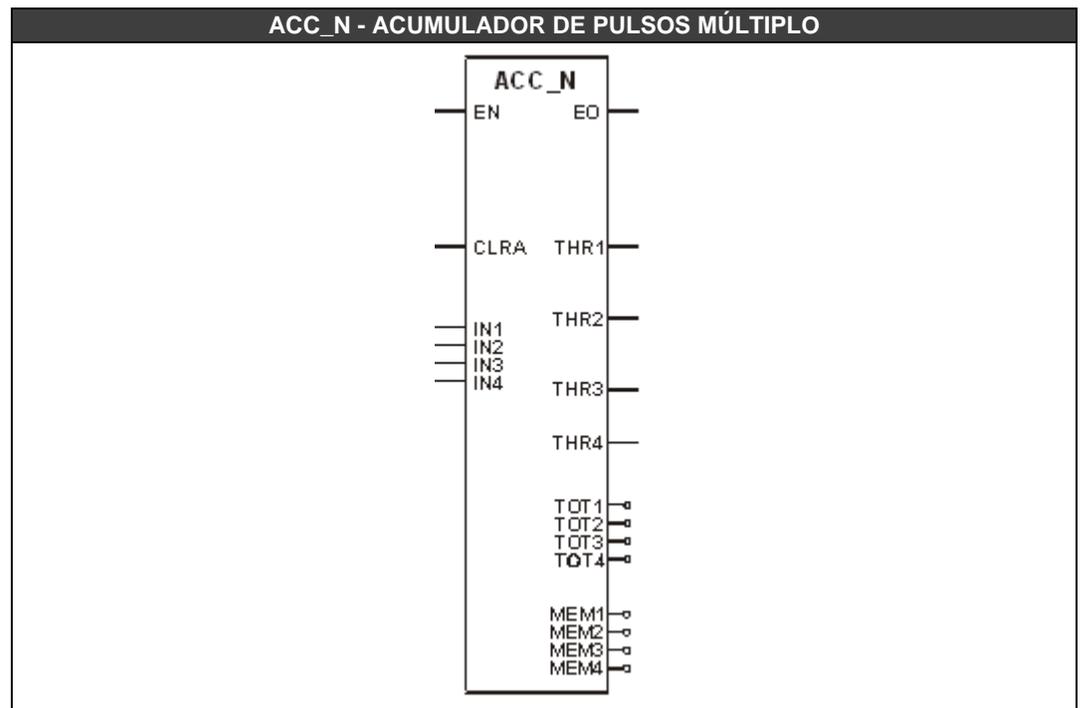
- 214 – Rack 0, slot 2, grupo 1 e ponto 4.
- 12307 – Rack 12, slot 3, grupo 0 e ponto 7.

Uma transição de subida na entrada **CLRA** irá apagar todos os acumuladores de pulsos simultaneamente, transferindo os valores acumulados em **TOT1**, **TOT2**, **TOT3** e **TOT4** para as saídas **MEM1**, **MEM2**, **MEM3** e **MEM4** respectivamente.

Os valores de Threshold das saídas **THR1**, **THR2**, **THR3** e **THR4** são verdadeiros ou falsos após configuração dos parâmetros **TR_ON**, **TR_OFF** e **MP** (parâmetros de configuração de cada ponto do módulo – DF41, DF42 ou DF67). Seu funcionamento é semelhante ao **THR** do bloco ACC.

Entrada CLRA

Toda vez que houver uma transição da entrada **CLRA** de verdadeiro para falso, as saídas **TOT** serão zeradas e os respectivos valores serão repassados às saídas **MEM**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--|---------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | CLRA | ZERA A TOTALIZAÇÃO E ENVIA OS VALORES PARA AS SAÍDAS MEM | BOOL |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | MEM1 | VALOR ACUMULADO DE PULSOS NA MEMÓRIA | LONG |
| | MEM2 | VALOR ACUMULADO DE PULSOS NA MEMÓRIA | LONG |
| | MEM3 | VALOR ACUMULADO DE PULSOS NA MEMÓRIA | LONG |
| | MEM4 | VALOR ACUMULADO DE PULSOS NA MEMÓRIA | LONG |
| | TOT1 | VALOR ACUMULADO NA SAÍDA 1 | LONG |
| | TOT2 | VALOR ACUMULADO NA SAÍDA 2 | LONG |
| | TOT3 | VALOR ACUMULADO NA SAÍDA 3 | LONG |
| | TOT4 | VALOR ACUMULADO NA SAÍDA 4 | LONG |
| | THR1 | SAÍDA THR1 | BOOL |
| | THR2 | SAÍDA THR2 | BOOL |
| | THR3 | SAÍDA THR3 | BOOL |
| | THR4 | SAÍDA THR4 | BOOL |
| | P | IN1 | CANAL 1 |
| IN2 | | CANAL 2 | LONG |
| IN3 | | CANAL 3 | LONG |
| IN4 | | CANAL 4 | LONG |

I: Entrada O: Saída P: Parâmetro Interno

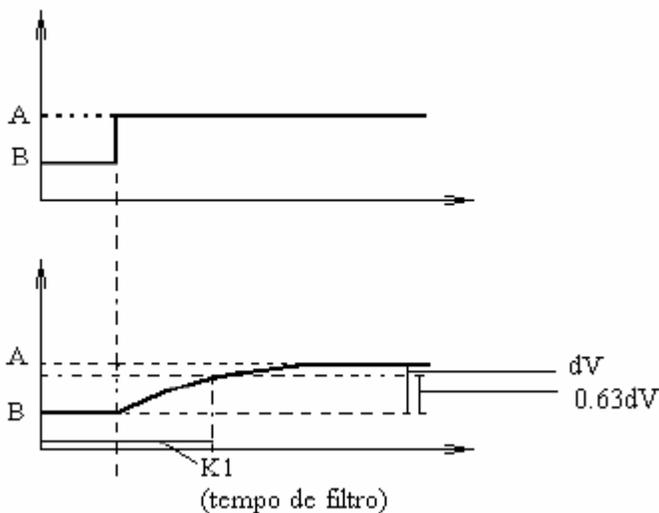
Entrada Analógica Simples (AI)

Descrição:

Este bloco funcional, quando a entrada **EN** está verdadeira, lê o valor do módulo de entrada analógica associado ao **CN** (Channel/Canal) e o atribui à saída **OUT**. Além disto, o bloco possui algumas funcionalidades a mais. A entrada **CN** deve ser do tipo RRS GP, onde R=rack, S=slot, G=grupo e P=ponto.

Tempo característico do filtro (K1)

K1 é o tempo característico do filtro, em segundos. Considere uma entrada degrau, quando o valor da saída atingir 63 % do valor do degrau, o tempo decorrido até este instante é definido como tempo característico do filtro.



Raiz Quadrada:

Se o parâmetro **SQR** for verdadeiro, obtém-se a raiz quadrada do valor da entrada analógica. Se a entrada for negativa, o resultado será zero.

Com **SQR** igual a verdadeiro, se o parâmetro **MUL** for verdadeiro, aplica-se a seguinte equação:

$$OUT = 10 * \sqrt{IN}$$

Caso a entrada analógica tenha um valor menor que o valor especificado no parâmetro **CTO**, a saída receberá o valor zero (nivelamento). Caso seja especificado um valor negativo para **CTO**, será assumido o valor zero.

Offset:

O parâmetro **Off** define um valor de offset que será adicionado ao valor convertido pelo bloco **AI**.

Saída de burnout:

A saída **BRT**, estando em 1, indica que a entrada está em *burnout*, ou seja, a entrada está num valor 2% maior do que o limite superior da escala ou num valor 2% menor do que o limite inferior da escala.

Estando em burnout, o parâmetro **BRTY** indica qual o tipo de atuação ocorrerá na saída **OUT**:

None: na saída estará o valor real da entrada.

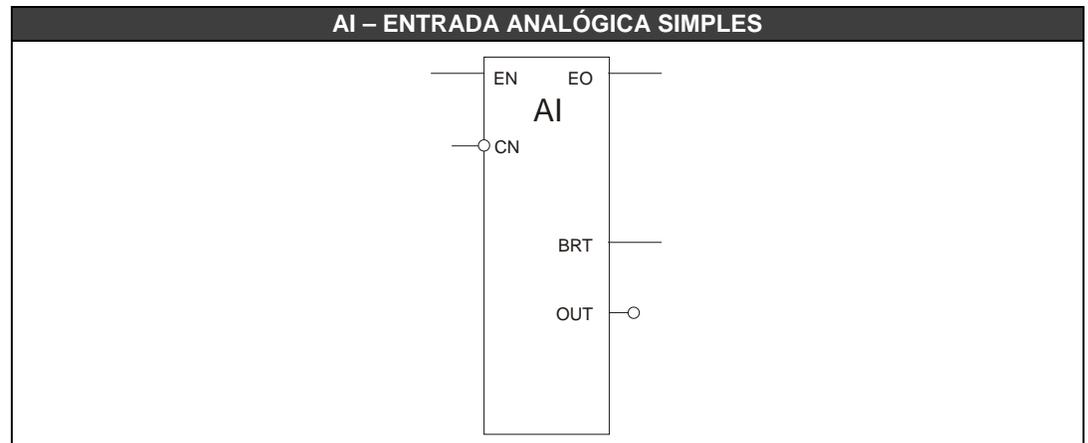
Low: a saída estará travada no valor 2% menor do que o limite inferior da escala.

High: a saída estará travada no valor 2% maior do que o limite superior da escala.

Caso o módulo de entrada analógica não exista, ou a CPU não consiga lê-lo, a saída depende do parâmetro **BRTY**:

None/High: a saída estará travada no valor 125% maior do que o limite superior da escala..

Low: a saída estará travada no valor 125% menor do que o limite inferior da escala.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|----------|------|--|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | CN | CANAL | LONG |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | BRT | SAÍDA DE BURNOUT | BOOL |
| | OUT | SAÍDA CONVERTIDA | FLOAT |
| P | SQR | HABILITA A EXTRAÇÃO DA RAIZ QUADRADA | BOOL |
| | CTO | NIVELAMENTO (CUT-OFF) | FLOAT |
| | MUL | MULTIPLICA A RAIZ QUADRADA POR 10 | FLOAT |
| | K1 | TEMPO CARACTERÍSTICO DO FILTRO EM SEGUNDOS. É UM FILTRO EXPONENCIAL DE PRIMEIRA ORDEM. | FLOAT |
| | OFF | VALOR DE OFFSET | FLOAT |
| | BRTY | INDICAÇÃO DO TIPO DE BURNOUT | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Entradas Analógicas para Equipamentos HART (Alh)

Descrição

Este bloco funcional, quando a entrada **EN** está verdadeira, lê os valores provenientes do equipamento HART de entrada, que esteja associado a **CN** (Channel), e os atribui às saídas **PRI_V**, **PV**, **SV**, **TV**, **QV**, **5V**, **6V**, **7V** e **8V**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|--------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | PRI_V | SAÍDA PARA O VALOR DE CORRENTE | FLOAT |
| O | PV | SAÍDA PARA O VALOR DE PV | FLOAT |
| | SV | SAÍDA PARA O VALOR DE SV | FLOAT |
| | TV | SAÍDA PARA O VALOR DE TV | FLOAT |
| | QV | SAÍDA PARA O VALOR DE QV | FLOAT |
| | 5V | SAÍDA PARA O VALOR DE 5V | FLOAT |
| | 6V | SAÍDA PARA O VALOR DE 6V | FLOAT |
| | 7V | SAÍDA PARA O VALOR DE 7V | FLOAT |
| | 8V | SAÍDA PARA O VALOR DE 8V | FLOAT |
| | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| P | CN | CANAL | LONG |

I: Entrada O: Saída P: Parâmetro Interno

IMPORTANTE

O parâmetro CN deve ser configurado obrigatoriamente com o canal base do slot em que o módulo está inserido. A regra para o preenchimento é **RRSx0** sendo RR: rack, S: slot e x representa o equipamento conectado ao DF116, de 0 a 7.

Exemplos:

- 200 – Rack 0, slot 2, equipamento 0
- 12350 – Rack 12, slot 3, equipamento 5

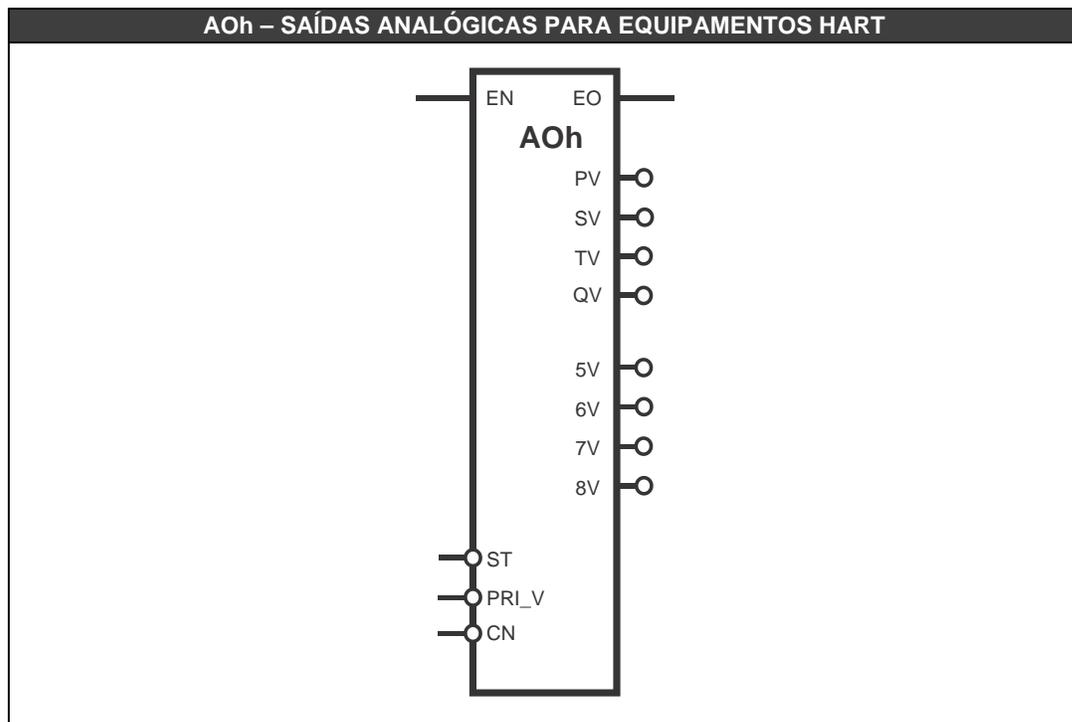
O bloco **Alh** acessa apenas um equipamento HART.

Saídas Analógicas para Equipamento HART (AOh)

Descrição

Este bloco funcional, quando a entrada **EN** está verdadeira, lê os valores do equipamento HART de saída (atuador), que esteja associado a **CN** (Channel), os atribui às saídas **PV**, **SV**, **TV**, **QV**, **5V**, **6V**, **7V** e **8V** e escreve em **PRI_V** o valor da corrente.

Se a entrada **EN** for falsa, o valor da corrente será escrito na entrada **ST**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|---------------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | ST | VALOR DA CORRENTE QUANDO EN FOR FALSA | FLOAT |
| | PRI_V | ENTRADA PARA O VALOR DE CORRENTE | FLOAT |
| O | PV | SAÍDA PARA O VALOR DE PV | FLOAT |
| | SV | SAÍDA PARA O VALOR DE SV | FLOAT |
| | TV | SAÍDA PARA O VALOR DE TV | FLOAT |
| | QV | SAÍDA PARA O VALOR DE QV | FLOAT |
| | 5V | SAÍDA PARA O VALOR DE 5V | FLOAT |
| | 6V | SAÍDA PARA O VALOR DE 6V | FLOAT |
| | 7V | SAÍDA PARA O VALOR DE 7V | FLOAT |
| | 8V | SAÍDA PARA O VALOR DE 8V | FLOAT |
| | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| P | CN | CANAL | LONG |

I: Entrada O: Saída P: Parâmetro Interno

IMPORTANTE

O parâmetro **CN** deve ser configurado obrigatoriamente com o canal base do slot em que o módulo está inserido. A regra para o preenchimento é **RRSx0** sendo RR: rack, S: slot e x representa o equipamento conectado ao DF117, de 0 a 7. Exemplos:

- 200 – Rack 0, slot 2, equipamento 0
- 12350 – Rack 12, slot 3, equipamento 5

O bloco **AOh** acessa apenas um equipamento HART.

Múltiplas Entradas Analógicas (MAI)

Descrição:

Este bloco funcional, quando a entrada **EN** está verdadeira, lê os valores do módulo de entrada analógica associado em **CN** (Channel) e os atribui às saídas **AI0**, **AI1**, **AI2**, **AI3**, **AI4**, **AI5**, **AI6** e **AI7**.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| O | AI0 | SAÍDA 0 | FLOAT |
| | AI1 | SAÍDA 1 | FLOAT |
| | AI2 | SAÍDA 2 | FLOAT |
| | AI3 | SAÍDA 3 | FLOAT |
| | AI4 | SAÍDA 4 | FLOAT |
| | AI5 | SAÍDA 5 | FLOAT |
| | AI6 | SAÍDA 6 | FLOAT |
| | AI7 | SAÍDA 7 | FLOAT |
| | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| P | CN | CANAL | LONG |

II: Entrada O: Saída P: Parâmetro Interno

IMPORTANTE

O parâmetro **CN** deve ser configurado obrigatoriamente com o canal base do slot em que o módulo está inserido. A regra para o preenchimento é **RRS00** sendo RR: rack e S: slot. Exemplos:

- 200 – Rack 0, slot 2.
- 12300 – Rack 12, slot 3.

Múltiplas Entradas Analógicas para IOR ou HART (MAIx)

Descrição:

Este bloco funcional, quando a entrada **EN** está verdadeira, lê os valores do módulo de entrada analógica associado a **CN** (Channel) e os atribui às saídas **AI0**, **AI1**, **AI2**, **AI3**, **AI4**, **AI5**, **AI6** e **AI7**.

Para os módulos de IOR (E/S Redundantes), as 8 entradas no módulo correspondem aos valores de saída do bloco. Já para os módulos HART, as saídas do bloco correspondem aos valores da corrente primária de entrada dos 8 equipamentos ligados aos canais do módulo.

E ainda gera na saída **STS** o estado das entradas, cada bit correspondendo a uma entrada, sendo que 0 indica "good" e 1 "bad".



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|--------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| O | AI0 | SAÍDA 0 | FLOAT |
| | AI1 | SAÍDA 1 | FLOAT |
| | AI2 | SAÍDA 2 | FLOAT |
| | AI3 | SAÍDA 3 | FLOAT |
| | AI4 | SAÍDA 4 | FLOAT |
| | AI5 | SAÍDA 5 | FLOAT |
| | AI6 | SAÍDA 6 | FLOAT |
| | AI7 | SAÍDA 7 | FLOAT |
| | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | STS | ESTADO INDIVIDUAL DAS ENTRADAS | LONG |
| P | CN | CANAL | LONG |

I: Entrada O: Saída P: Parâmetro Interno

IMPORTANTE

O parâmetro **CN** deve ser configurado obrigatoriamente com o canal base do slot em que o módulo está inserido. A regra para o preenchimento é **RRS00** sendo RR: rack e S: slot. Exemplos:

- 200 – Rack 0, slot 2.
- 12300 – Rack 12, slot 3.

Múltiplas Saídas Analógicas (MAO)

Descrição:

Este bloco funcional, quando a entrada **EN** está verdadeira, atribui os valores ligados ou configurados nas entradas **A0**, **A1**, **A2** e **A3** às respectivas saídas do módulo de saída analógica associado em **CN** (channel).

As entradas **ST0**, **ST1**, **ST2** e **ST3** são os valores de segurança (*Fault State Value*) a serem atribuídos às saídas do módulo caso a entrada **EN** seja falsa.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | A00 | ENTRADA 0 | FLOAT |
| | A01 | ENTRADA 1 | FLOAT |
| | A02 | ENTRADA 2 | FLOAT |
| | A03 | ENTRADA 3 | FLOAT |
| | ST0 | VALOR DE SEGURANÇA 0 | FLOAT |
| | ST1 | VALOR DE SEGURANÇA 1 | FLOAT |
| | ST2 | VALOR DE SEGURANÇA 2 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| P | CN | CANAL | LONG |

I: Entrada **O:** Saída **P:** Parâmetro Interno

IMPORTANTE

O parâmetro **CN** deve ser configurado obrigatoriamente com o canal base do slot em que o módulo está inserido. A regra para o preenchimento é **RRS00** sendo RR: rack e S: slot. Exemplos:

- 200 – Rack 0, slot 2.
- 12300 – Rack 12, slot 3.

Múltiplas Saídas Analógicas para IOR ou HART (MAOx)

Descrição

Este bloco funcional, quando a entrada **EN** está verdadeira, escreve os valores das entradas **AO0**, **AO1**, **AO2**, **AO3**, **AO4**, **AO5**, **AO6** e **AO7** no módulo de saída analógica associado a **CN** (Channel).

Para os módulos de IOR (E/S Redundantes), as saídas do módulo correspondem aos valores de entrada dos 8 canais do bloco. Já para os módulos HART, as entradas do bloco correspondem aos valores da corrente primária de saída dos 8 equipamentos ligados aos canais do módulo.

E ainda gera na saída **STS** o estado das saídas, cada bit correspondendo a uma saída, sendo que 0 indica "good" e 1 "bad".



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|------------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | AO0 | SAÍDA 0 | FLOAT |
| | AO1 | SAÍDA 1 | FLOAT |
| | AO2 | SAÍDA 2 | FLOAT |
| | AO3 | SAÍDA 3 | FLOAT |
| | AO4 | SAÍDA 4 | FLOAT |
| | AO5 | SAÍDA 5 | FLOAT |
| | AO6 | SAÍDA 6 | FLOAT |
| O | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| | STS | ESTADO INDIVIDUAL DAS SAÍDAS | LONG |
| P | CN | CANAL | LONG |

I: Entrada O: Saída P: Parâmetro Interno

IMPORTANTE

O parâmetro **CN** deve ser configurado obrigatoriamente com o canal base do slot em que o módulo está inserido. A regra para o preenchimento é **RRS00** sendo RR: rack e S: slot. Exemplos:

- 200 – Rack 0, slot 2.
- 12300 – Rack 12, slot 3.

Status do Sistema (STATUS)

Descrição:

Este bloco, quando **EN** é verdadeira, permite configurar 8 variáveis booleanas que informam o status de um módulo de E/S. Este bloco é recomendado para que se tenha um melhor monitoramento do estado funcional de cada módulo de E/S utilizado, assim o sistema poderá ser informado se algum módulo de E/S falhar. Desta forma fica mais fácil encontrar o módulo danificado.

NOTA

O número GLL que está impresso na placa eletrônica deve ser superior a 1100, caso contrário o módulo não suportará identificação através do bloco Status.

Parâmetros:

A programação do módulo de E/S a ter o status monitorado é feita definindo-se um par de parâmetros **RACKi** e **SLOTi**.

RACK1: define o rack do modulo de E/S a ter o status monitorado na saída **OUT1** do bloco.
SLOT1: define o slot do modulo de E/S a ter o status monitorado na saída **OUT1** do bloco.

RACK2: define o rack do modulo de E/S a ter o status monitorado na saída **OUT2** do bloco.
SLOT2: define o slot do modulo de E/S a ter o status monitorado na saída **OUT2** do bloco.

RACK3: define o rack do modulo de E/S a ter o status monitorado na saída **OUT3** do bloco.
SLOT3: define o slot do modulo de E/S a ter o status monitorado na saída **OUT3** do bloco.

RACK4: define o rack do modulo de E/S a ter o status monitorado na saída **OUT4** do bloco.
SLOT4: define o slot do modulo de E/S a ter o status monitorado na saída **OUT4** do bloco.

RACK5: define o rack do modulo de E/S a ter o status monitorado na saída **OUT5** do bloco.
SLOT5: define o slot do modulo de E/S a ter o status monitorado na saída **OUT5** do bloco.

RACK6: define o rack do modulo de E/S a ter o status monitorado na saída **OUT6** do bloco.
SLOT6: define o slot do modulo de E/S a ter o status monitorado na saída **OUT6** do bloco.

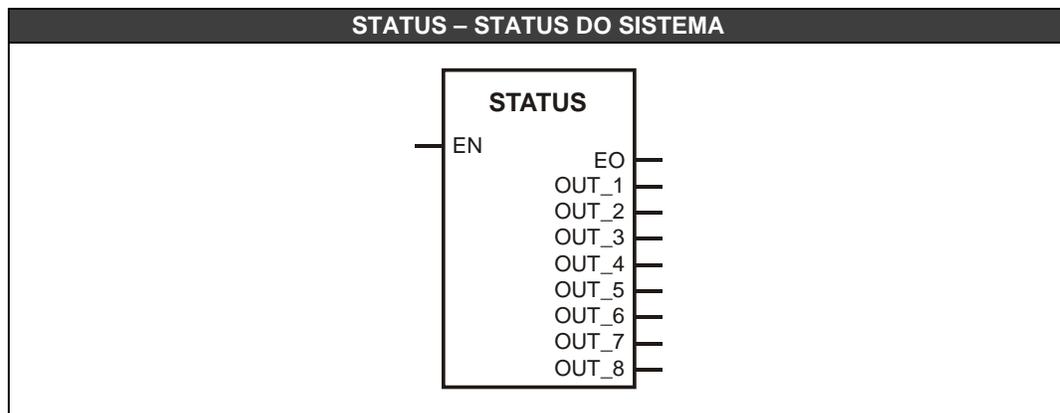
RACK7: define o rack do modulo de E/S a ter o status monitorado na saída **OUT7** do bloco.
SLOT7: define o slot do modulo de E/S a ter o status monitorado na saída **OUT7** do bloco.

RACK8: define o rack do modulo de E/S a ter o status monitorado na saída **OUT8** do bloco.
SLOT8: define o slot do modulo de E/S a ter o status monitorado na saída **OUT8** do bloco.

Significado do Status e saídas:

0: Status= módulo de E/S "bom".

1: Status= módulo de E/S "ruim".



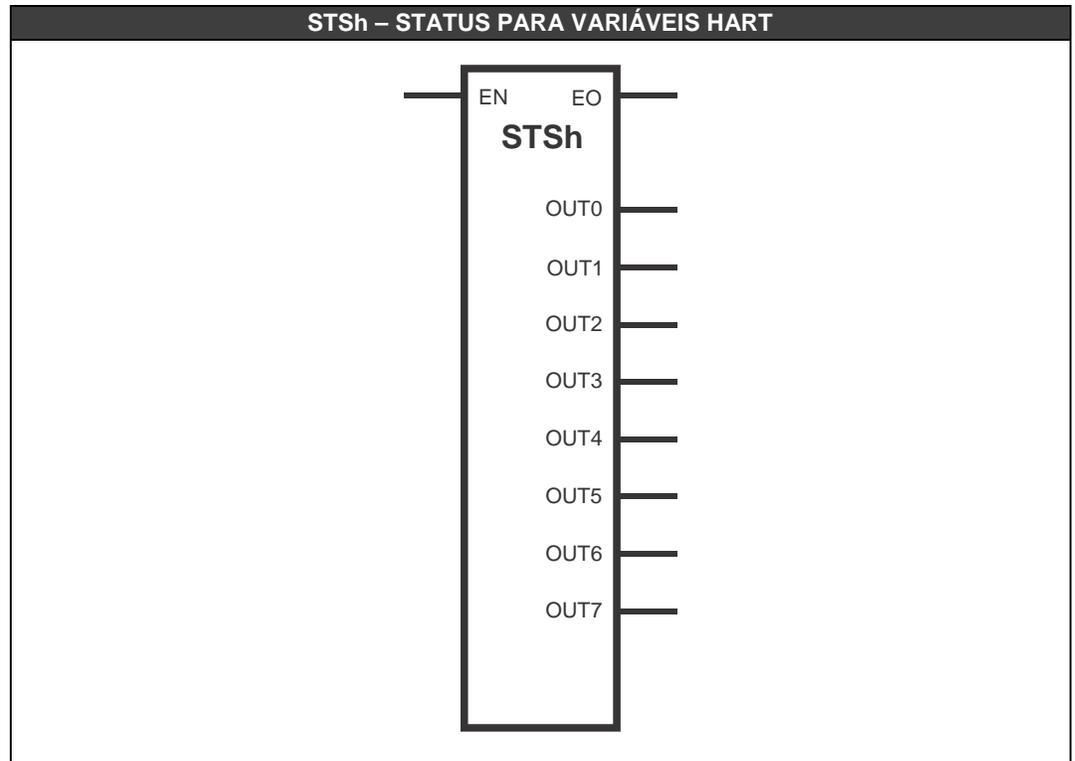
| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|---|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| O | OUT1 | STATUS DO MÓDULO DEFINIDO EM RACK1/SLOT1 | BOOL |
| | OUT2 | STATUS DO MÓDULO DEFINIDO EM RACK2/SLOT2 | BOOL |
| | OUT3 | STATUS DO MÓDULO DEFINIDO EM RACK3/SLOT3 | BOOL |
| | OUT4 | STATUS DO MÓDULO DEFINIDO EM RACK4/SLOT4 | BOOL |
| | OUT5 | STATUS DO MÓDULO DEFINIDO EM RACK5/SLOT5 | BOOL |
| | OUT6 | STATUS DO MÓDULO DEFINIDO EM RACK6/SLOT6 | BOOL |
| | OUT7 | STATUS DO MÓDULO DEFINIDO EM RACK7/SLOT7 | BOOL |
| | OUT8 | STATUS DO MÓDULO DEFINIDO EM RACK8/SLOT8 | BOOL |
| P | RACK1 | RACK DO MÓDULO CUJO STATUS ESTARÁ EM OUT1 | LONG |
| | SLOT1 | SLOT DO MÓDULO CUJO STATUS ESTARÁ EM OUT1 | LONG |
| | RACK2 | RACK DO MÓDULO CUJO STATUS ESTARÁ EM OUT2 | LONG |
| | SLOT2 | SLOT DO MÓDULO CUJO STATUS ESTARÁ EM OUT2 | LONG |
| | RACK3 | RACK DO MÓDULO CUJO STATUS ESTARÁ EM OUT3 | LONG |
| | SLOT3 | SLOT DO MÓDULO CUJO STATUS ESTARÁ EM OUT3 | LONG |
| | RACK4 | RACK DO MÓDULO CUJO STATUS ESTARÁ EM OUT4 | LONG |
| | SLOT4 | SLOT DO MÓDULO CUJO STATUS ESTARÁ EM OUT4 | LONG |
| | RACK5 | RACK DO MÓDULO CUJO STATUS ESTARÁ EM OUT5 | LONG |
| | SLOT5 | SLOT DO MÓDULO CUJO STATUS ESTARÁ EM OUT5 | LONG |
| | RACK6 | RACK DO MÓDULO CUJO STATUS ESTARÁ EM OUT6 | LONG |
| | SLOT6 | SLOT DO MÓDULO CUJO STATUS ESTARÁ EM OUT6 | LONG |
| | RACK7 | RACK DO MÓDULO CUJO STATUS ESTARÁ EM OUT7 | LONG |
| | SLOT7 | SLOT DO MÓDULO CUJO STATUS ESTARÁ EM OUT7 | LONG |
| | RACK8 | RACK DO MÓDULO CUJO STATUS ESTARÁ EM OUT8 | LONG |
| | SLOT8 | SLOT DO MÓDULO CUJO STATUS ESTARÁ EM OUT8 | LONG |

I: Entrada O: Saída P: Parâmetro Interno

Status para Variáveis HART (STSh)

Descrição:

Este bloco, quando **EN** é verdadeira, permite visualizar o status de até 8 variáveis dos equipamentos HART ligados aos canais dos módulos HART, sendo que 0 indica “good” e 1 “bad”.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|------|---|------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| O | OUT0 | STATUS DA VARIÁVEL DEFINIDA EM CN0 | BOOL |
| | OUT1 | STATUS DA VARIÁVEL DEFINIDA EM CN1 | BOOL |
| | OUT2 | STATUS DA VARIÁVEL DEFINIDA EM CN2 | BOOL |
| | OUT3 | STATUS DA VARIÁVEL DEFINIDA EM CN3 | BOOL |
| | OUT4 | STATUS DA VARIÁVEL DEFINIDA EM CN4 | BOOL |
| | OUT5 | STATUS DA VARIÁVEL DEFINIDA EM CN5 | BOOL |
| | OUT6 | STATUS DA VARIÁVEL DEFINIDA EM CN6 | BOOL |
| | OUT7 | STATUS DA VARIÁVEL DEFINIDA EM CN7 | BOOL |
| P | CN0 | CANAL DA VARIÁVEL A TER STATUS MONITORADO | LONG |
| | CN1 | CANAL DA VARIÁVEL A TER STATUS MONITORADO | LONG |
| | CN2 | CANAL DA VARIÁVEL A TER STATUS MONITORADO | LONG |
| | CN3 | CANAL DA VARIÁVEL A TER STATUS MONITORADO | LONG |
| | CN4 | CANAL DA VARIÁVEL A TER STATUS MONITORADO | LONG |
| | CN5 | CANAL DA VARIÁVEL A TER STATUS MONITORADO | LONG |
| | CN6 | CANAL DA VARIÁVEL A TER STATUS MONITORADO | LONG |
| | CN7 | CANAL DA VARIÁVEL A TER STATUS MONITORADO | LONG |

I: Entrada O: Saída P: Parâmetro Interno

IMPORTANTE

Os parâmetros **CN0...7** devem ser configurados obrigatoriamente com o canal da variável HART a ter o status monitorado. O canal segue o padrão **RRSGP**, sendo **RR**: rack; **S**: slot do módulo HART; **G**: grupo (posição do equipamento HART no módulo) e **P**: ponto indicando a variável a ser monitorada, de acordo com a tabela seguinte:

0: PV
1: SV
2: TV
3: QV
4: 5V
5: 6V
6: 7V
7: 8V
8: corrente primária

Exemplos:

Monitorar o status da TV do equipamento HART ligado no segundo canal do módulo HART que está no rack 3, slot 1:

CNx= 3112

Monitorar o status da PV do equipamento HART ligado no quinto canal do módulo HART que está no rack 12, slot 2:

CNx= 12240

Monitorar o status da corrente primária do equipamento HART ligado no primeiro canal do módulo HART que está no rack 1, slot 0:

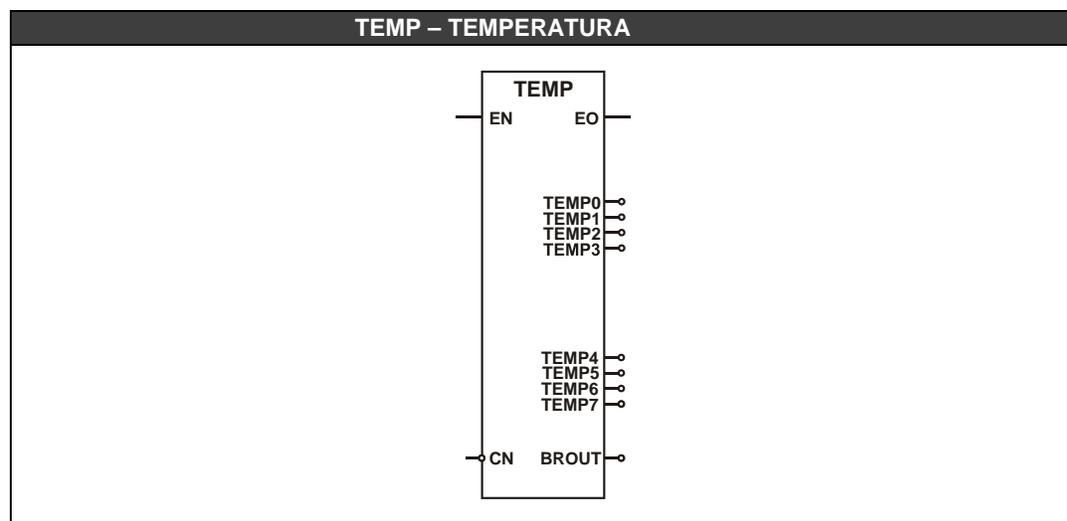
CNx= 1008

Temperatura (TEMP)

Descrição:

Este bloco funcional, quando a entrada **EN** está verdadeira, lê os valores do módulo de temperatura associado em **CN** (Channel/ Canal) e os atribui às saídas **TEMP0**, **TEMP1**, **TEMP2**, **TEMP3**, **TEMP4**, **TEMP5**, **TEMP6** e **TEMP7**.

O parâmetro de saída **BROUT** indica se há falha em cada uma das entradas de temperatura do módulo. Cada entrada corresponde a um bit, num total de 8 bits, em que o nível lógico 0 indica operação normal e o nível lógico 1 indica falha. Essa saída deverá ser utilizada em conjunto com o bloco BTB que separa cada um dos bits do valor apresentado.



| CLASSE | MNEM | DESCRIÇÃO | TIPO |
|--------|-------|------------------------|-------|
| I | EN | HABILITAÇÃO DA ENTRADA | BOOL |
| | EO | HABILITAÇÃO DA SAÍDA | BOOL |
| O | TEMP0 | SAÍDA DE TEMPERATURA 0 | FLOAT |
| | TEMP1 | SAÍDA DE TEMPERATURA 1 | FLOAT |
| | TEMP2 | SAÍDA DE TEMPERATURA 2 | FLOAT |
| | TEMP3 | SAÍDA DE TEMPERATURA 3 | FLOAT |
| | TEMP4 | SAÍDA DE TEMPERATURA 4 | FLOAT |
| | TEMP5 | SAÍDA DE TEMPERATURA 5 | FLOAT |
| | TEMP6 | SAÍDA DE TEMPERATURA 6 | FLOAT |
| | TEMP7 | SAÍDA DE TEMPERATURA 7 | FLOAT |
| P | BROUT | BURN OUT | LONG |
| | CN | CANAL | LONG |

I: Entrada **O:** Saída **P:** Parâmetro Interno

IMPORTANTE

O parâmetro **CN** deve ser configurado obrigatoriamente com o canal base do slot em que o módulo está inserido. A regra para o preenchimento é **RRS00** sendo RR: rack e S: slot. Exemplos:

- 200 – Rack 0, slot 2.
- 12300 – Rack 12, slot 3.

O LOGICVIEW FOR FFB

Introdução

Este capítulo apresenta os fundamentos do uso do software **LogicView for FFB** para os controladores da Smar - DF62, DF63, DF73, DF75, DF79, DF81, DF89, DF95 e DF97. Será mostrado como criar, enviar e corrigir erros eventuais na configuração das lógicas ladder que serão executadas nesses controladores.

Antes de ler este capítulo o usuário deverá ler os capítulos 1 e 2 deste manual para se familiarizar com os elementos de ladder e blocos funcionais.

O software **LogicView for FFB** é baseado no Microsoft Windows e, portanto, é operado da mesma maneira básica que outras aplicações Windows, isto é, através de menus, browsing, cortar e colar, botões e drop down lists, etc. Assume-se que o usuário já possua familiaridade com a interface Windows.

Instalação

Sistema Operacional

O **LogicView for FFB**, como qualquer outra parte integrante do **SYSTEM302**, opera em sistema operacional Windows . Veja mais detalhes no **Guia de Instalação do SYSTEM302**.

Antes de Iniciar a Instalação

Verifique as características mínimas para instalação no **Guia de Instalação do SYSTEM302**. É recomendado e, às vezes, obrigatório, que o usuário feche todas as aplicações antes de iniciar a instalação do **SYSTEM302**.

Instalando

A instalação deve autoiniciar poucos segundos após o DVD de instalação ser inserido no respectivo drive. Se após inserir o DVD no drive a instalação não iniciar automaticamente, vá até o diretório que contém a aplicação e execute o arquivo AUTORUN. A instalação do programa vai iniciar e guiá-lo durante todo procedimento de instalação. Para maiores detalhes refira-se ao **Guia de Instalação do SYSTEM302**.

Licença

Após instalar o **SYSTEM302** é preciso validar as licenças de uso para executar seus aplicativos, o **LogicView for FFB** é um deles. Pra maiores informações sobre como obter a licença de uso refira-se ao **Guia de Instalação do SYSTEM302**.

O usuário pode optar por trabalhar em modo Demo, no entanto, existem algumas restrições de uso:

- Ao começar uma configuração nova será permitido trabalhar com apenas um diagrama ladder. Todas as funcionalidades serão mantidas.
- Se o usuário tentar abrir uma configuração com mais de um diagrama aparecerá uma mensagem informando-o que em modo Demo só é possível trabalhar com arquivos que tenham apenas um diagrama ladder e o arquivo não será aberto.



Fig 3. 1 – Erro Demo Mode



Retirando a Hardkey

Se o **LogicView for FFB** estiver licenciado através de uma *hardkey* e durante a sua operação ela for retirada em 9 minutos o **LogicView for FFB** será fechado. Antes de fechar perguntará ao usuário se deseja salvar a configuração. No momento em que a *hardkey* for retirada aparecerá a seguinte mensagem.



Fig 3. 2 – Alerta sobre a falta de licença

Clicando em **Ok**, entrará em ação a contagem regressiva.

Se a *hardkey* for recolocada antes de expirar os nove minutos o *timeout* será cancelado. Caso não seja recolocada o **LogicView for FFB** será fechado, mas antes o usuário será alertado. Veja figura abaixo.



Fig 3. 3 – Alerta sobre o tempo de licença expirado

O usuário poderá acompanhar a contagem regressiva para o fechamento do **LogicView for FFB** na barra de Status, no canto inferior esquerdo. A cada minuto aparecerá a mensagem: “*This application will shutdown in x minute(s)!*”, indicando ao usuário o tempo restante.

Usando o LogicView for FFB

Iniciando uma Aplicação

Para iniciar uma aplicação, o usuário deve clicar em **Iniciar** → **Programas** → **System302** → **Studio302** → **Studio302**.

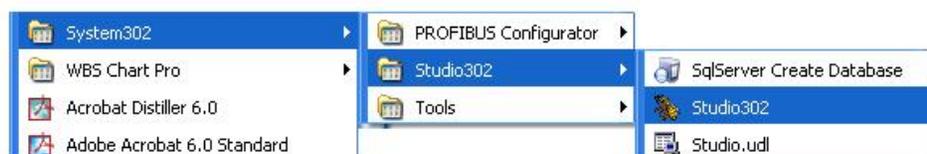


Fig 3. 4 - Iniciando uma aplicação Studio302

A seguinte janela abrirá e o **LogicView for FFB** poderá ser executado a partir desta janela no modo **Template**. Para isto basta clicar na barra de ferramentas, que está logo abaixo do menu principal, no ícone mostrado na figura abaixo.

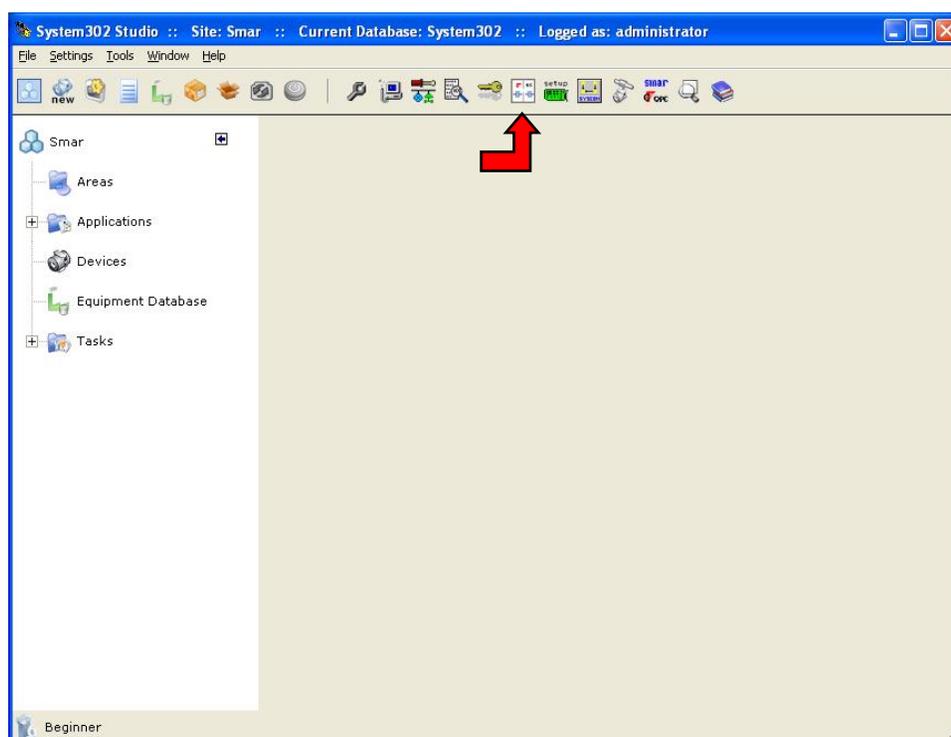


Fig 3. 5 - Iniciando uma aplicação no LogicView for FFB

Depois o usuário deverá escolher a opção **New FFB Logic Template**. O **LogicView for FFB** será executado no modo **Template**. Veja a figura a seguir.

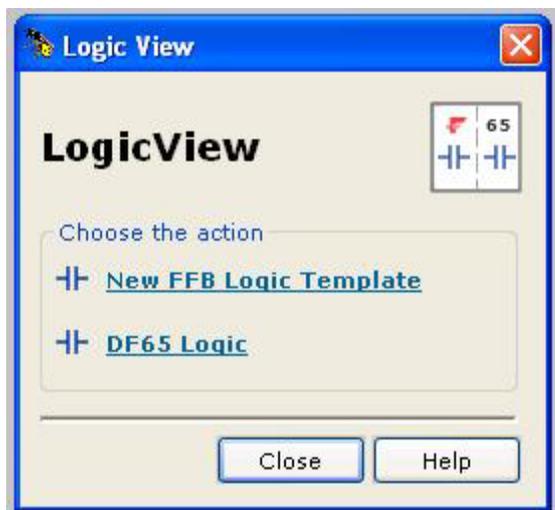


Fig 3. 6 - Iniciando uma aplicação no LogicView for FFB

No modo **Instance** o usuário deverá executar o **Syscon** e de lá ao editar a lógica o **LogicView for FFB** abrirá.

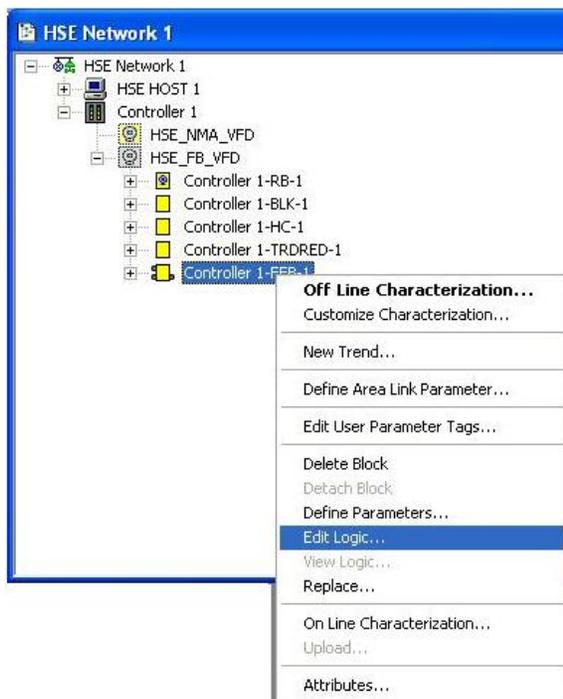


Fig 3. 7 - Editando a lógica - Modo Instance

Modo Instance

Tipicamente, o acesso ao **LogicView for FFB** para criação ou modificação da lógica discreta de um bloco flexível (FFB), será feita a partir da instância de um FFB criada no **Syscon**, como mostra a figura seguinte. Uma Instância de FFB (FFB Instance) pode ser entendida como um bloco “real” que pode ser transferido via download para um *linking device*. A instância POSSUI informações de “descritor de bloco” (*Device Descriptor*) e por este motivo é EXCLUSIVO de uma configuração que o contém.

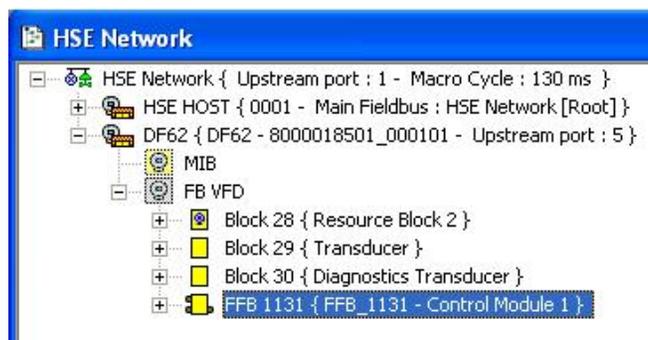


Fig 3. 8 - Bloco FFB no Syscon

Após inserir o bloco FFB no **Syscon** o usuário deverá definir primeiramente seus parâmetros e só depois disto é que a lógica discreta poderá ser editada. Clicando com o botão direito do mouse sobre o **bloco FFB** e depois em **Define Parameters**, a seguinte janela se abrirá:

Fig 3. 9 - Definição dos parâmetros FFB

NOTA

A partir da versão 7.3 do **SYSTEM302**, o FFB é criado automaticamente, com a seguinte quantidade de parâmetros: 32 DO, 32 DI, 32 AO, 32 AI, 4 DO64, 4 DI64, 4 AI16 e 4 AO16, estes 4 últimos tipos se for criado o FFB2.

Aqui o usuário poderá configurar a quantidade de entradas e saídas analógicas e digitais: Analog Inputs, Analog Outputs, Digital Inputs, Digital Outputs, Analog Inputs16, Analog Outputs16, Digital Inputs64 ou Digital Outputs64, respectivamente. Ao clicar **OK** os pontos DI, DO, AI, AO, DI64, DO64, AI16 e AO16 são gerados. Em **I/O Type** são escolhidos quantos e quais parâmetros serão configurados. Em **Single I/O** são configurados DI, DO, AI e AO. Em **Multiple I/O** são configurados DI64, DO64, AI16 e AO16. Esses pontos possibilitarão a troca de informações entre o controle contínuo, que utiliza a tecnologia FOUNDATION™ fieldbus, e o controle discreto. Para maiores detalhes sobre o **FFB Parameters Definition** veja o manual do **Syscon**. Veja figura seguinte:

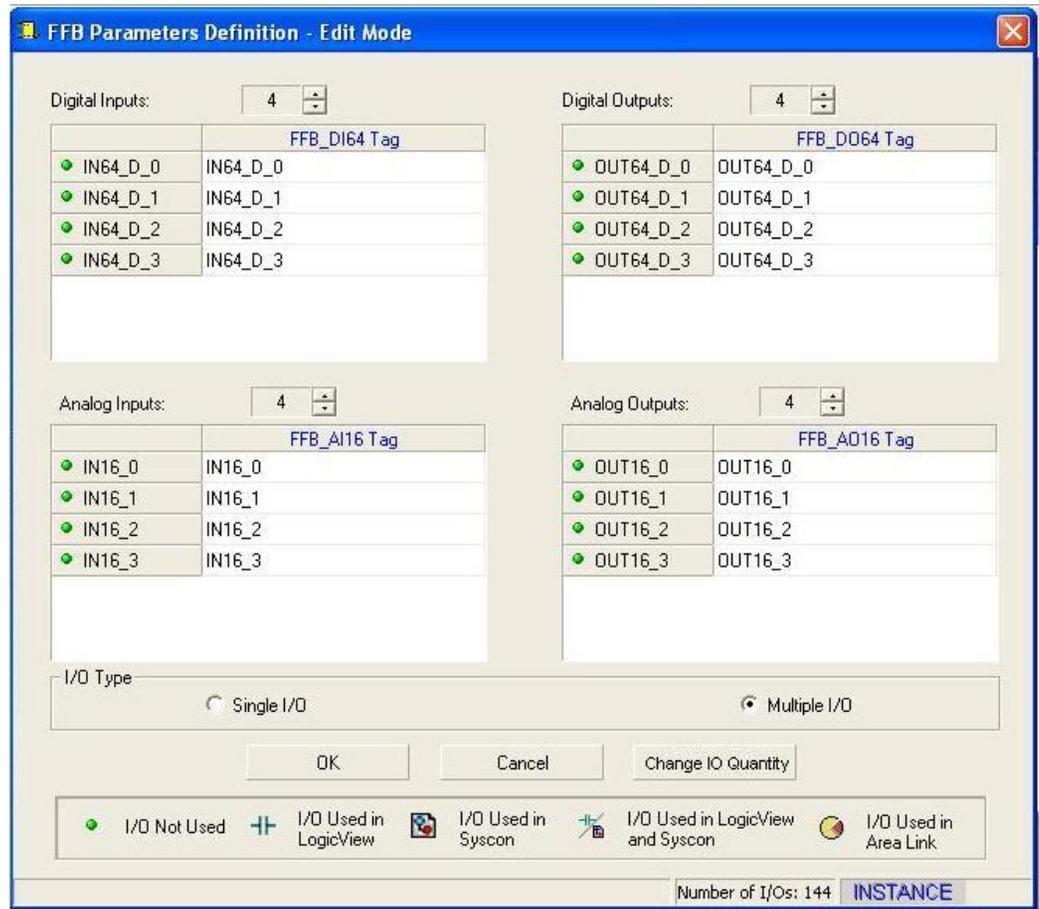


Fig 3. 10 - Definição dos parâmetros FFB

Quando os parâmetros do bloco FFB já estiverem definidos o usuário deverá editar a lógica. Clicando com o botão direito do mouse sobre o bloco **FFB** e depois em **Edit Logic** o **LogicView for FFB** será lançado em modo de Edição de Instância.

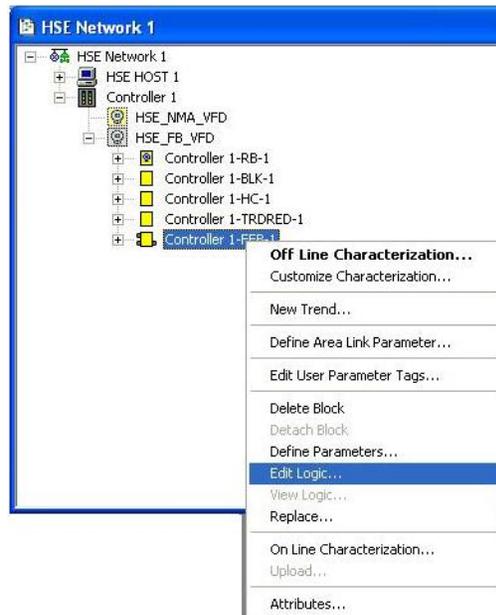


Fig 3. 11 - Editando a lógica - Modo Instance

Este modo de operação pode ser identificado por um tag **FFB LOGIC** no canto inferior esquerdo da ferramenta, conforme a figura abaixo:

FFB LOGIC

Fig 3. 12 – Identificação do Modo Instance

O tag do bloco FFB ao qual a instância está ligada aparecerá na barra de título, na parte superior da janela do **LogicView for FFB**, entre **LogicView for FFB** e o nome do arquivo gerado em modo Instance.

Neste modo o **LogicView for FFB** estará apto a EDITAR apenas a instância de Bloco Flexível (FFB) a partir da qual o comando **Edit Logic** foi processado. Por esta razão, operações como "New", "Open" ou "Save As" não estarão habilitadas, apenas o "Save" será permitido.

As operações de lógica ladder estarão totalmente habilitadas, possibilitando a criação e/ou modificação da lógica discreta. Por se tratar de um bloco real, é possível fazer o download da lógica para o dispositivo correspondente.

NOTA

Os arquivos gerados neste modo terão extensão "pgi".

Caso seja necessário alterar a definição de parâmetros do FFB no **Syscon**, através do procedimento descrito acima, ou seja, clicando com o botão direito do mouse sobre o bloco FFB, a tela da **Define Parameters Tool** poderá se apresentar da seguinte forma:

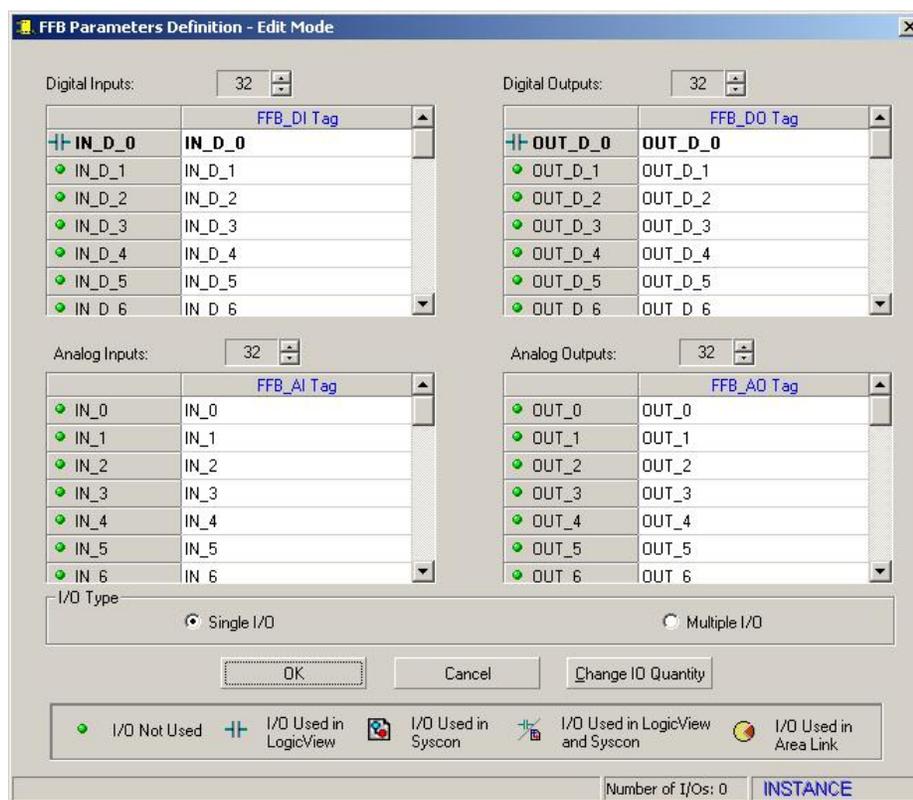


Fig 3. 13 – Alteração de parâmetros de E/S do FFB

Na figura anterior, a ferramenta informa o *status* das entradas/saídas do FFB, ou seja, se determinada entrada ou saída está sendo usada no **Syscon** e/ou no **LogicView for FFB**. As condições de uma entrada/saída são as seguintes:

 : significa que o estado atual da entrada/saída do FFB é "**Não Usado**". Portanto o tag deste ponto estará disponível para alterações e poderá ser editado;

 : significa que o estado atual da entrada/saída do FFB é "**Usado no LogicView**". Portanto, este ponto já foi associado dentro da lógica interna do FFB que está sendo editado e seu tag não estará disponível para alteração, está bloqueado para edição;

 : significa que o estado atual da entrada/saída do FFB é "**Usado no Syscon**". Portanto, este ponto já foi utilizado dentro de uma estratégia do **Syscon** que contém o FFB que está sendo editado e seu tag não estará disponível para alteração, está bloqueado para edição;

 : significa que o estado atual da entrada/saída do FFB é "**Usado no LogicView e Syscon**". Portanto, este ponto já foi utilizado da forma descrita nos itens acima, nas duas ferramentas simultaneamente e seu tag não estará disponível para alteração, está bloqueado para edição;

 : significa que o estado atual da entrada/saída do FFB é "**Definido pelo usuário como um parâmetro a ser usado pelo Area Link**". Portanto seu tag não estará disponível para alteração, está bloqueado para edição.



Ao forçar a edição dos tags a seguinte mensagem será exibida.



Fig 3. 14 – Aviso de desbloqueio de edição de pontos usados

Clicando em **Sim**, a ferramenta irá desbloquear todas as entradas/saídas que estavam com a edição protegida, levando automaticamente todos os pontos E/S ao status "**Não Usado**".

Cada parâmetro do bloco é exibido na janela com seu respectivo User Tag (definido no **Syscon**) ou com um tag default, caso não tenha um User Tag definido para ele. Para alterar os tags clique com o botão direito no ícone do bloco FFB no **Syscon** (na janela *Process Cell*, *Fieldbus* ou de estratégia) e clique **Edit User Parameter Tags**. A caixa de diálogo **User Parameter Tag** abrirá. Para mais detalhes sobre alterações de tags veja o manual do **Syscon**.



O FFB permite a utilização de tipos de dados mais avançados e o bloco é criado inicialmente com uma quantidade pré-definida destes pontos identificados como **Multiple I/O** na janela **Define Parameters**. Selecionando-se esta opção os pontos múltiplos serão exibidos na janela e sua edição é idêntica aos pontos simples (**Single IO**).

Após modificar uma definição de parâmetros de um FFB previamente criado, clique **OK** e a seguinte mensagem aparecerá:



Fig 3. 15 – Aviso de alteração de FFB já criado

Na caixa de mensagem acima, clicando em:

Sim: confirma as alterações e o FFB será modificado de acordo com as operações efetuadas na ferramenta;

Não: fecha a caixa de mensagens e volta à ferramenta para continuar a edição;

Cancelar: cancela todas as alterações realizadas e fecha a ferramenta.



Modificando um FFB já definido

Modificar um FFB que já foi previamente definido e utilizado em uma configuração de controle pode resultar no apagamento do bloco durante o processo de download, criando inconsistências na planta. É recomendado fazer um *download* completo na *bridge* ou *gateway* que contém o FFB modificado.

Modo Template

Outra maneira de se trabalhar com o Bloco Flexível de Lógica e que permite o seu “reúso” em diferentes configurações é a definição de “FFB Templates”. O template de um FFB é na verdade um “modelo” de bloco flexível que pode ser reaproveitado em configurações fieldbus distintas.

Por se tratar de um “modelo” apenas, um FFB Template NÃO PODE ser transferido via download para um *linking device*. A principal característica técnica do FFB Template é que ele NÃO POSSUI informações de “descriptor de bloco” (*Device Descriptor*). No entanto, essa característica contribui para a flexibilidade e reutilização do bloco nas aplicações.

O procedimento para lançar o **LogicView for FFB** em modo de edição de “FFB Templates” é a partir do ícone **LogicView** localizado na Barra de Ferramentas ou no item **Tasks** do **Studio302**, conforme figura 3.5. Outra forma é a partir da janela **Logics** do **Studio302**, com a opção via botão direito do mouse **New Logic Template**.

O procedimento acima descrito irá lançar o **LogicView for FFB** em modo de Edição de Template. Este modo de operação pode ser identificado por um tag **LOGIC** no canto inferior esquerdo da ferramenta, conforme a figura seguinte:

LOGIC

Fig 3. 16 - Modo Template

Ao se criar um novo template, a primeira ação do usuário será definir um tag para o mesmo, conforme figura seguinte.

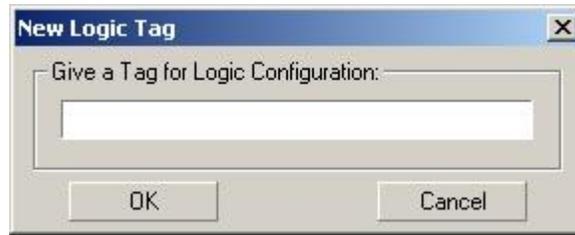


Fig 3. 17 – Informando o tag do novo template

Neste modo o **LogicView for FFB** estará apto a CRIAR e MODIFICAR apenas modelos (*Templates*) de Blocos Flexíveis. Por esta razão, operações como "New", "Open" ou "Save As" e "Save" estarão habilitadas. As operações de lógica ladder e a definição de parâmetros do FFB estarão totalmente habilitadas, possibilitando o desenvolvimento, sem restrições, da configuração discreta. No entanto, por se tratar de um "modelo" de FFB, o usuário não poderá fazer o download da lógica para o dispositivo correspondente.

NOTA

Os arquivos gerados neste modo terão extensão "pgt".

Para utilizar no **Syscon** um Template de Bloco Flexível de Lógica criado no **LogicView for FFB** é necessário criar uma **Instância de Bloco** baseada no **Template**, para que as informações de *Device Descriptor* sejam criadas pelo sistema e o bloco esteja pronto para ser transferido para um device via download. Uma instância de FFB criada a partir de um Template herda todas as suas características (Definição de Parâmetros e Lógica Interna).

A criação de uma Instância de FFB a partir de um Template é feita no **Syscon** a partir da opção **Replace** no menu do FFB, mostrada na figura seguinte:

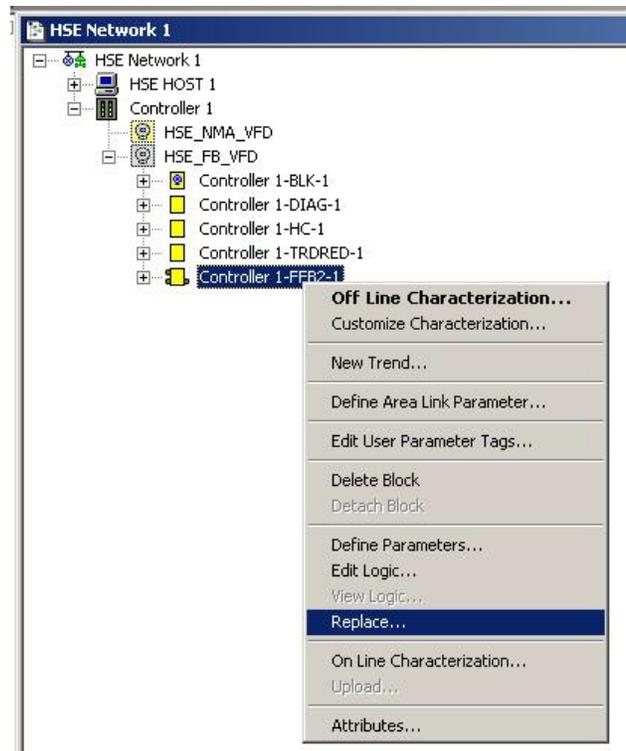


Fig 3. 18 - Inserção de um FFB via template

Será exibida uma lista dos FFB Templates criados pelo usuário. Basta selecionar um deles e clicar **OK**, para que uma Instância de FFB seja automaticamente gerada pelo sistema e incorporada ao Módulo de Controle (*Control Module*) do **Syscon**:

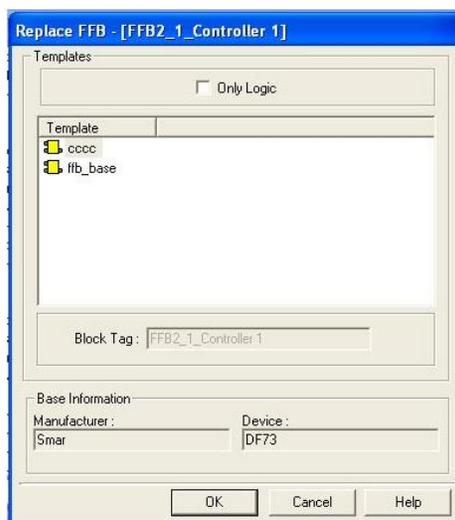


Fig 3. 19 - Novo FFB via template

Caso queira substituir apenas o algoritmo interno (lógica) do FFB, selecione a opção **Only Logic** na tela acima e clique **OK** para concluir.

IMPORTANTE

Algumas observações sobre **Instâncias** e **Templates** de FFB:

- As modificações feitas sobre um template de FFB **SÓ TERÃO EFEITO** sobre as Instâncias de FFB criadas **APÓS** a modificação realizada. Qualquer instância criada **ANTES** da modificação do template de FFB não será afetada pela alteração;
- Duas configurações distintas podem ter Instâncias de FFB criadas a partir do **MESMO TEMPLATE DE FFB**, porém essas Instâncias serão **DISTINTAS** (devido aos *Device Descriptors* apresentarem características diferentes). A criação de uma Instância de FFB é automática e gerenciada pelo sistema;
- As modificações feitas numa Instância não têm efeito sobre o Template no qual ela se baseou ao ser criada.

Modo Supervision Only

Neste modo, é possível a abertura de vários “LogicViews” simultaneamente na mesma máquina. Todas as operações de mudança de configuração da lógica estão bloqueadas. O usuário pode apenas visualizar a configuração, entrar em supervisão, monitorar as variáveis, tanto discretas quanto analógicas, como também atuar (escrever) sobre as mesmas. Mas estando em modo **SUPERVISION ONLY**, as mudanças nas variáveis não são persistidas em arquivos, apenas na CPU atuada.

O **LogicView for FFB** automaticamente entra em modo **SUPERVISION ONLY** ao abrir uma instância caso uma outra lógica já esteja aberta. Este modo só atua sobre instância, não tendo efeito sobre templates.

O modo **SUPERVISION ONLY** é indicado na barra de título e pela cor de fundo, que pode ser modificada pelo usuário. Veja figura a seguir.

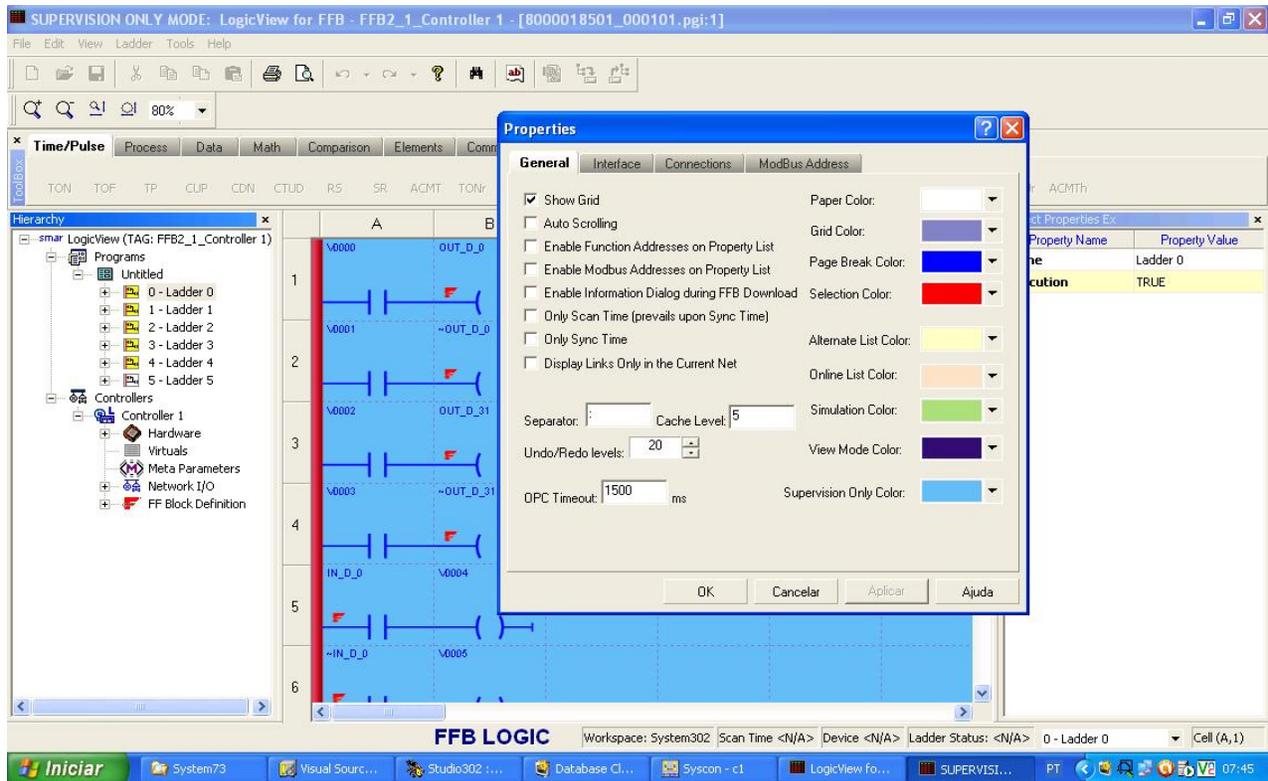


Fig 3. 20 – Modo Supervision Only

Modo Simulation

Este modo é usado para simulação de estratégias do **SYSTEM302** através do aplicativo **SimulationView**. Tanto o **Syscon** e o **LogicView for FFB** devem estar **Online** para que as estratégias possam ser simuladas. Para mais detalhes sobre como configurar e operar no modo de simulação veja o manual do **SimulationView**.

O modo **Simulation** é indicado na barra de título e pela cor de fundo da área de trabalho, que pode ser modificada pelo usuário. Veja figura a seguir.

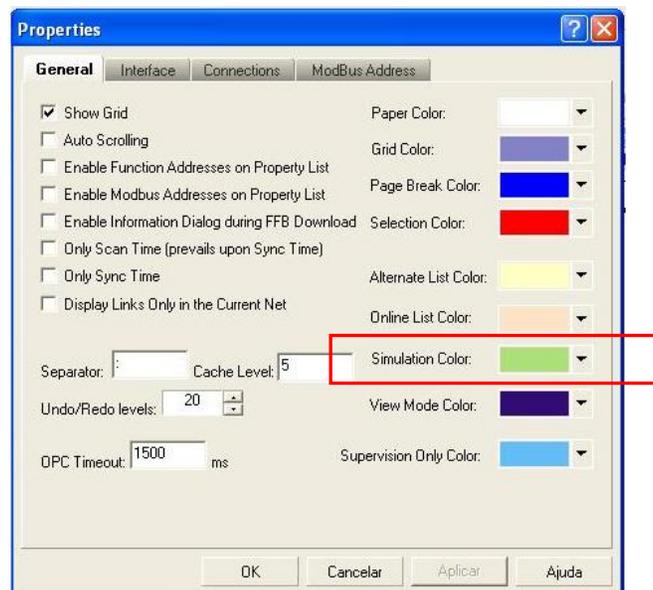


Fig 3. 21 – Opção de mudança de cor do Modo Simulation

Modo View

Este modo serve apenas para visualizar a lógica ladder associada ao bloco FFB. No **Syscon**, clique com o botão direito no ícone do FFB e selecione **View Logic**. O **LogicView for FFB** será executado no modo **View** e nenhuma alteração poderá ser feita.



Avaliação da rede Ladder

A avaliação da rede ladder pelo **LogicView for FFB** é feita da esquerda para direita (por linhas). Esta é uma característica que o usuário ao fazer a sua configuração deve ter sempre em mente. Veja figura abaixo.

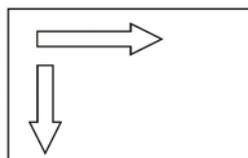


Fig 3. 22 - Avaliação da rede ladder

No **LogicView for DF65** a avaliação da ladder é feita de cima pra baixo (por colunas), por isso ao fazer novas configurações para o **LogicView for FFB** o usuário tem que tomar este cuidado, como exemplificado a seguir.

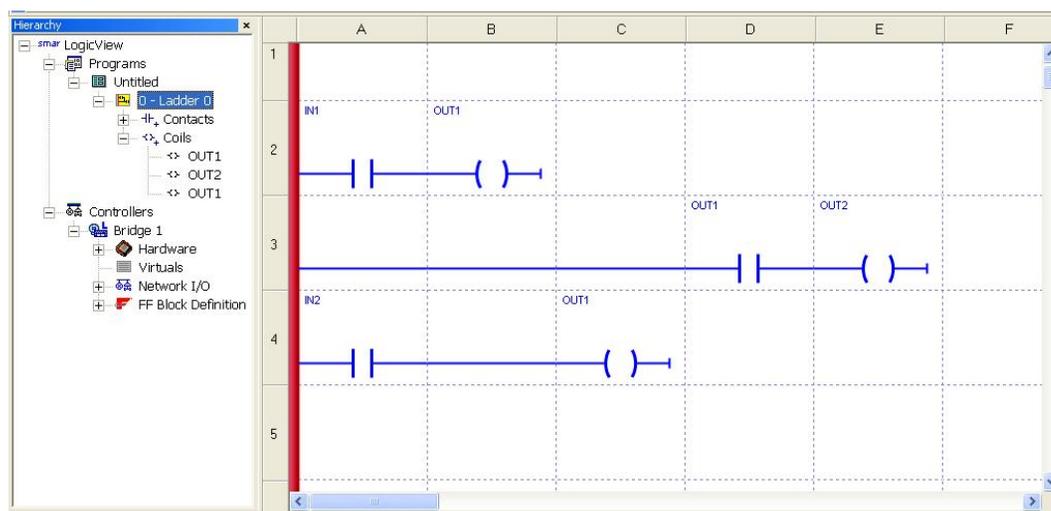


Fig 3. 23 - Exemplo de avaliação da ladder

No **DF65**, OUT2 será ativada quando IN2 for ativada, já que a ordem de execução da ladder é por coluna. Já nos novos controladores, OUT2 será ativada quando IN1 for ativada, já que a execução da ladder é por linha.

Conhecendo a área de trabalho

Ao abrir o **LogicView for FFB**, seja no modo **Template** ou modo **Instance**, a seguinte janela se abrirá. Neste exemplo, já está configurada uma lógica ladder simples.

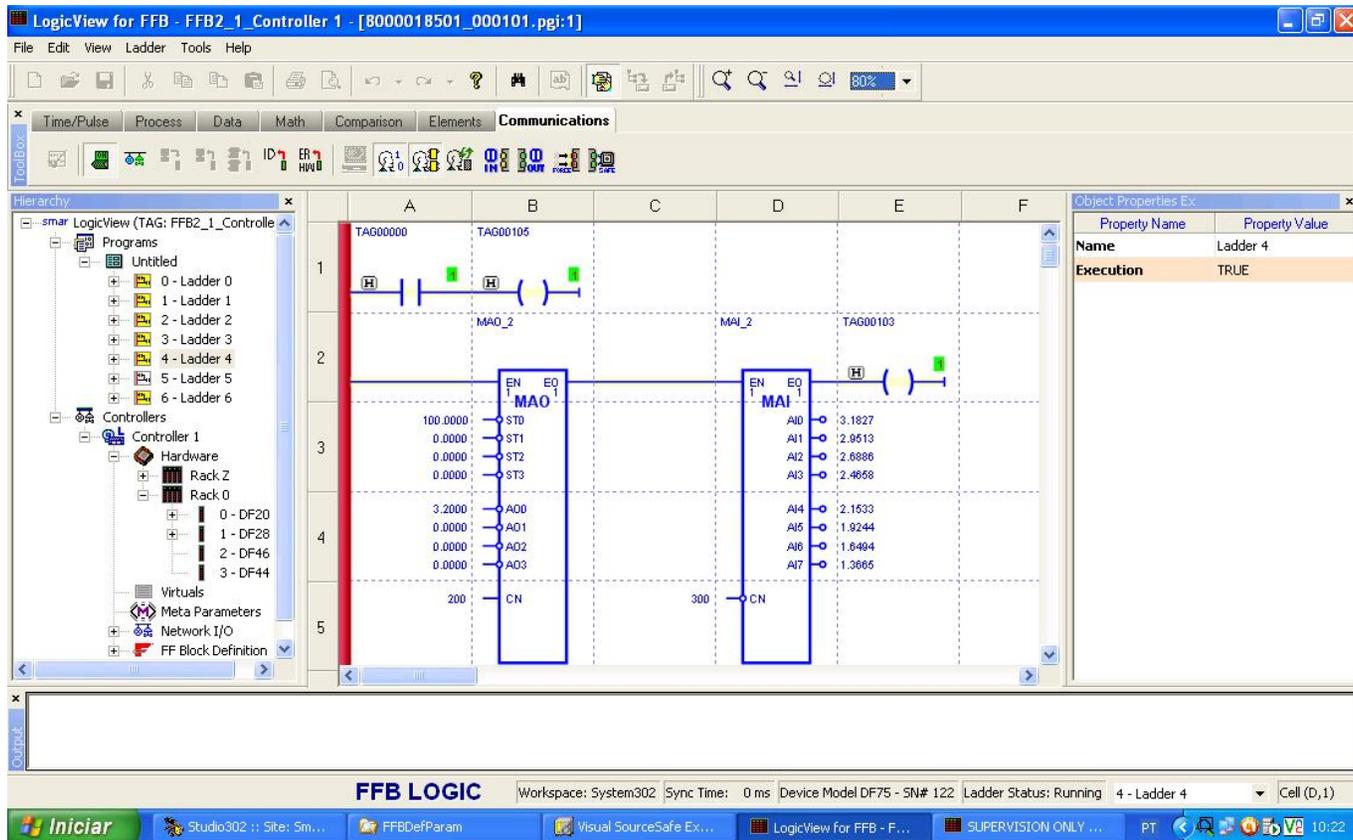


Fig 3. 24 - Área de trabalho

A área de trabalho do **LogicView for FFB** tem basicamente 7 subáreas:

- Menu principal
- Barras de Ferramentas
- Hierarchy
- Object Properties
- Área de Desenho da Ladder
- Output
- Status Bar

A seguir, cada uma delas será descrita em detalhes. Por default, todas elas estarão abertas ao iniciar o **LogicView for FFB**. Apenas o Menu Principal, a Status Bar e a Área de Desenho da Ladder não poderão ser fechadas.

Menu principal

No menu principal estão as funções básicas do software. Cada um deles pode ser aberto com a tecla ALT + a primeira letra do item desejado. A seguir todos os submenus serão detalhados.



Fig 3. 25 - Menu Principal

Menu File

Clicando em **File**, ou através do atalho ALT+ F, o seguinte menu se abrirá:

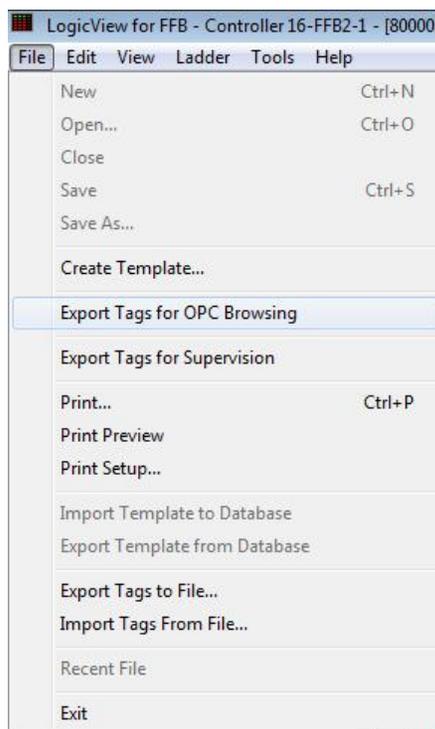


Fig 3. 26 - Menu File

Aqui estão as opções default do Windows **New**, **Open**, **Close**, **Print**, etc que funcionarão como em qualquer outro tipo de aplicativo Windows. Se o usuário estiver no **Modo Template** e escolher a opção **New**, um novo arquivo será criado com o Rack Z vazio e o Rack 0 com o slot 0 preenchido pela fonte de alimentação DF50 e o slot 1 preenchido pelo controlador DF75.

O usuário poderá escolher se deseja ou não utilizar o Rack Z (DF78 ou DF92) em sua configuração. Este rack deverá ser usado para redundância de fontes e controladores. Para maiores detalhes veja o manual do DFI302.

Maiores detalhes sobre a configuração do hardware serão apresentados no tópico **Hierarchy – Hardware configuration**.

Procedimento de Save/Save As

Ao escolher a opção **Save As**, será pedido um novo tag para a configuração, conforme figura a seguir.

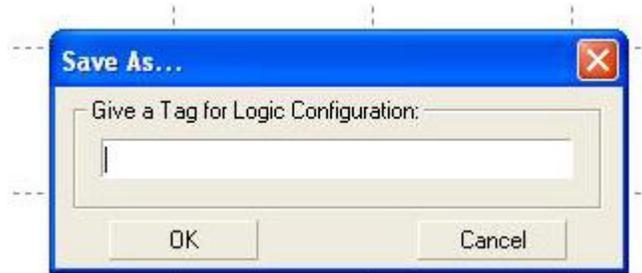


Fig 3. 27 – Salvando Templates de FFB

Se o tag já existir no sistema, o usuário será informado e deverá escolher outro tag.



Fig 3. 28 – Tag já existe

NOTA

A opção **Save As** está disponível apenas para Templates de FFB.

Create Template

Esta opção é usada para se criar um template de uma instância e, deste modo, poder replicar este template em qualquer outra CPU, independente do tipo de CPU definido na instância original. Ao clicar nesta opção a seguinte mensagem aparecerá:

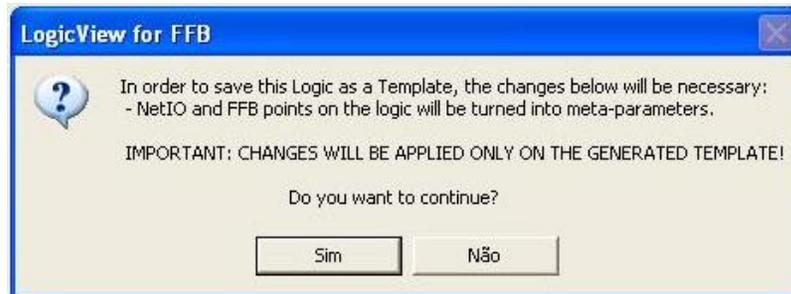


Fig 3. 29 – Confirmando a criação do template

O usuário é informado que todos os pontos de NetIO e FFB, que são dependentes da CPU utilizada, serão transformados em metaparâmetros. Em seguida, é pedido um tag para o template em questão. Automaticamente o template é criado e a instância original continua aberta. O template criado pode ser acessado via janela **Logics** do **Studio302**.

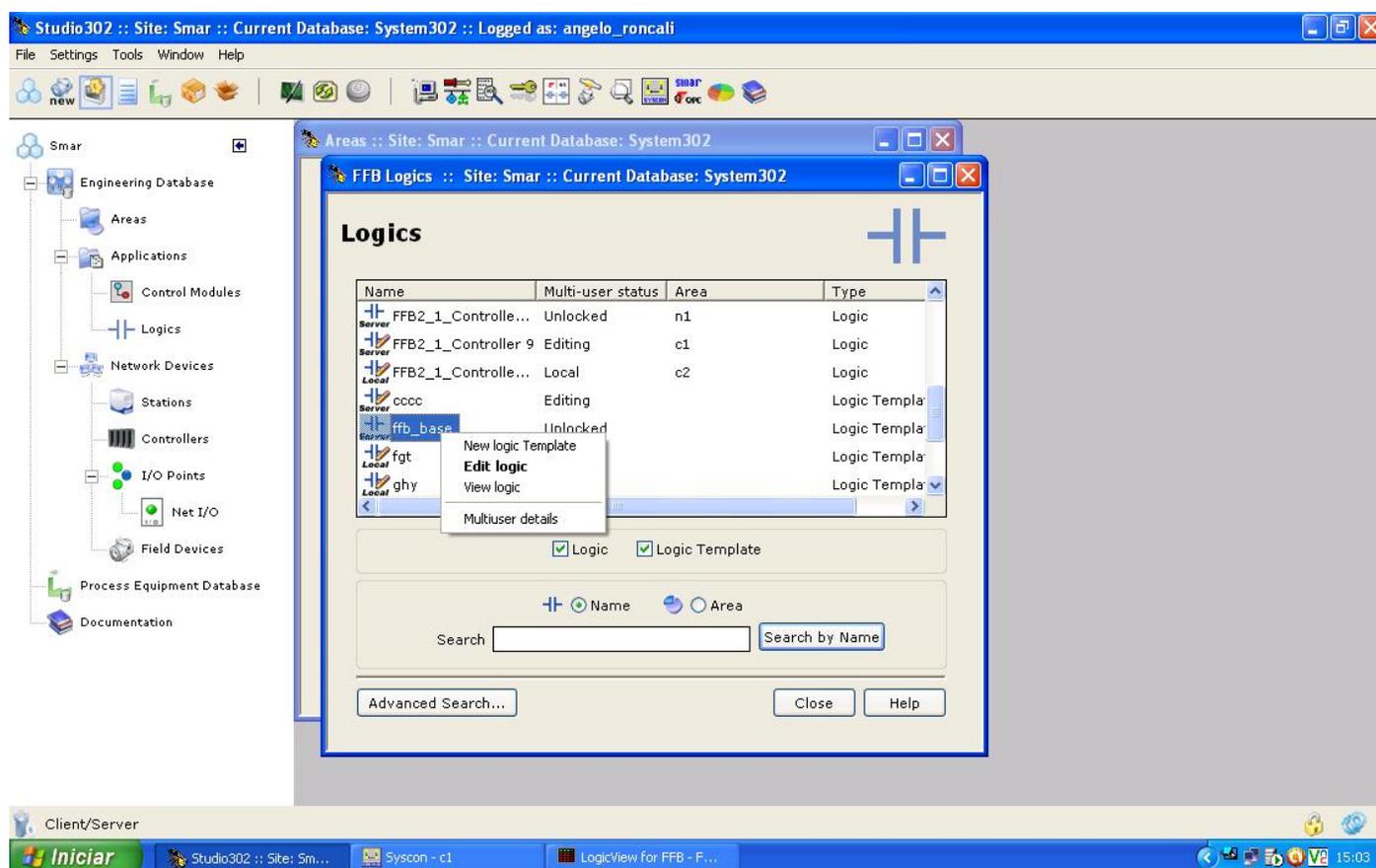


Fig 3. 30 – Janela Logics do Studio302

Export Tags for OPC Browsing

A opção **Export Tags for OPC Browsing** atualiza o arquivo Taginfo.ini com todos os tags da lógica aberta, habilitando-os para busca sem fazer o download da configuração no controlador.

NOTA

Esta é a mesma operação **Export Tags for OPC Browsing** do menu do **Syscon**. A diferença é o escopo, i.e., enquanto o **LogicView for FFB** realiza a exportação de tags somente da lógica aberta, a mesma operação no **Syscon** exporta todos os tags de todas as lógicas usadas na configuração (i.e. Area).

Esta operação também pode ser feita clicando no botão  da barra de ferramentas principal.

Export Tags for Supervision

Estando em **Modo Instance**, ao fazer uma mudança de tag na configuração da rede ladder o usuário pode fazer a exportação do tag sem precisar ir ao **Syscon**. Este faz a exportação da configuração completa portanto, é um processo mais lento. O usuário deve ir em **File** → **Export Tags for Supervision** e será solicitado o caminho onde se encontra o arquivo taginfo.ini. Este arquivo pode estar na máquina local ou numa máquina acessada remotamente via dcom pelo DFIOleServer.

A operação **Export Tags for Supervision** deve ser usada para atualizar a base de dados do OPC Server com os tags usados na lógica para o supervisor acessar esses tags OPC.

NOTA

Após um download da lógica sempre é realizado um **Export Tags for Supervision** automaticamente.



Fig 3. 31 – Localização do arquivo TagInfo.ini

A exportação será feita e a seguinte mensagem aparecerá.



Fig 3. 32 – Export Tags

Opções de Impressão

Print Setup

Clicando nesta opção o usuário poderá configurar a impressora e suas propriedades como em qualquer outro aplicativo Windows.

Print Preview

Clicando em **File** → **Print Preview**, ou em  na barra de ferramentas Main, a seguinte janela se abrirá.

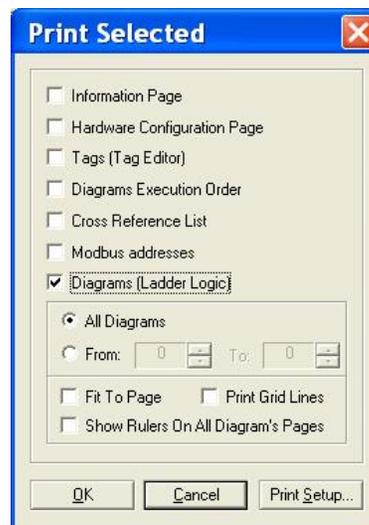


Fig 3. 33 – Opções de impressão

O usuário pode escolher quais informações deseja imprimir e como será a impressão. As opções são:

Information page: As informações sobre o projeto em questão cadastradas na janela **Object Properties**, como por exemplo nome da empresa, da planta, do projeto, controlador, etc serão impressas se este item for selecionado.

Hardware configuration page

Selecionando esta opção será impressa uma lista com toda a configuração do hardware como na figura abaixo.

| Rack | Slot | Module | Description |
|------|------|--------|---|
| 2 | 0 | DF50 | Power Supply Module 90-264VAC - Redundant |
| 2 | 1 | DF62 | DFI302 Processor 1x100Mbps, 4xH1 |
| 2 | 2 | DF20 | 1 Group of 8 On/Off Switches |
| 2 | 3 | DF24 | 2 Groups of 8 120/240VAC Outputs |

Fig 3. 34 – Lista da configuração de hardware

Tags (Tag Editor)

Selecionando esta opção será impressa uma lista com todos os tags da configuração. Veja a figura seguinte.

| Tag | Direction | Device | Channel | Safe | Description |
|----------|-----------|--------|---------|-------|-------------|
| TAG01200 | Input | DF20 | 01200 | ----- | |
| TAG01201 | Input | DF20 | 01201 | ----- | |
| TAG01202 | Input | DF20 | 01202 | ----- | |
| TAG01203 | Input | DF20 | 01203 | ----- | |
| TAG01204 | Input | DF20 | 01204 | ----- | |
| TAG01205 | Input | DF20 | 01205 | ----- | |
| TAG01206 | Input | DF20 | 01206 | ----- | |
| TAG01207 | Input | DF20 | 01207 | ----- | |
| TAG01300 | Output | DF24 | 01300 | Off | |
| TAG01301 | Output | DF24 | 01301 | Off | |
| TAG01302 | Output | DF24 | 01302 | Off | |
| TAG01303 | Output | DF24 | 01303 | Off | |
| TAG01304 | Output | DF24 | 01304 | Off | |
| TAG01305 | Output | DF24 | 01305 | Off | |
| TAG01306 | Output | DF24 | 01306 | Off | |
| TAG01307 | Output | DF24 | 01307 | Off | |
| TAG01310 | Output | DF24 | 01310 | Off | |
| TAG01311 | Output | DF24 | 01311 | Off | |
| TAG01312 | Output | DF24 | 01312 | Off | |
| TAG01313 | Output | DF24 | 01313 | Off | |

Fig 3. 35 – Lista dos tags da configuração

Diagrams Execution Order

Selecionando esta opção uma lista com a ordem de execução dos diagramas ladder será impressa.

Cross Reference List

Selecionando esta opção será impressa uma lista com todos os tags da configuração indicando os respectivos diagramas nos quais eles são utilizados. Veja a figura seguinte.

| Tag | Used in the diagram | Tag | Used in the diagram |
|----------|---------------------|-----|---------------------|
| TAG01200 | 0 | | |
| TAG01201 | 0 | | |
| TAG01202 | 0 | | |
| TAG01203 | 0 | | |
| TAG01204 | 0 | | |
| TAG01205 | 0 | | |
| TAG01206 | 0 | | |
| TAG01207 | 0 | | |

Fig 3. 36 – Lista dos tags – Referência cruzada

Modbus addresses

Selecionando esta opção será impressa uma lista com todos os grupos e todos os tags da configuração indicando os respectivos endereços Modbus utilizados. Veja a figura seguinte.

| Group | Start Address | End Address | Description |
|-------|---------------|-------------|-------------|
| 001 | 10001 | 10008 | |
| 002 | 00001 | 00016 | |

| Tag | In/Out Type | Group | MB Address |
|----------|-------------|-------|------------|
| TAG00300 | OUTPUT | 002 | 00001 |
| TAG00301 | OUTPUT | 002 | 00002 |
| TAG00302 | OUTPUT | 002 | 00003 |
| TAG00303 | OUTPUT | 002 | 00004 |
| TAG00304 | OUTPUT | 002 | 00005 |

Fig 3. 37 – Lista dos tags – Referência cruzada

Diagrams

Os diagramas com a lógica ladder podem ser impressos de várias formas. Ao selecionar Diagrams (Ladder Logic) as seguintes opções serão habilitadas e devem ser selecionadas conforme a necessidade do usuário:

- All Diagrams – Indica que serão impressos todos os diagramas.
- From xx to yy – Indica quais os diagramas serão impressos. Por exemplo, “From 0 to 4” indicando que serão impressos os diagramas 0, 1, 2, 3 e 4.
- Fit to page – O diagrama será impresso em uma única página.
- Print grid lines – As linhas de grid serão impressas juntamente com os diagramas.
- Show rulers on all diagram's pages – A régua que indica a numeração das células da Área de Desenho da Ladder será impressa em todas as páginas. Caso esta opção não seja selecionada, a régua será impressa somente nas páginas em que os diagramas estejam diretamente ligados a ela.

Print

Ao clicar em **File**→**Print**, ou no ícone  da barra de ferramentas Main, será aberto uma janela igual à da figura 3.33 e o usuário deverá selecionar aquelas que lhe convier conforme descrito no tópico anterior. A diferença é que após selecionar suas opções e clicar **OK** o arquivo com a configuração será impresso imediatamente.

Page Order

No canto superior direito da página impressa existe a informação **Page Order**. Veja figura seguinte.

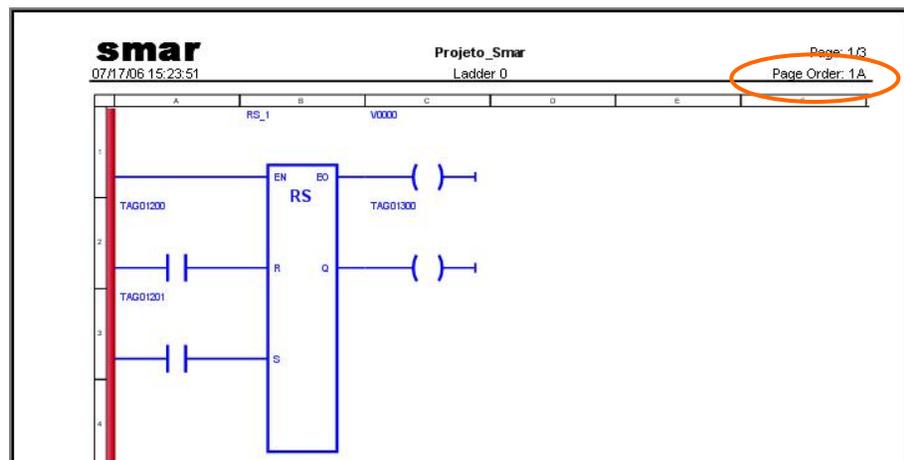


Fig 3. 38 – Ordem de montagem da impressão

O código indica a linha e a coluna da ordem de montagem da impressão quando a opção **Fit to page** não tiver sido escolhida. Por exemplo, o diagrama Ladder 0 mostrado acima será impresso em 3 páginas. Para que o usuário visualize todo o diagrama deverá montá-lo da seguinte forma:

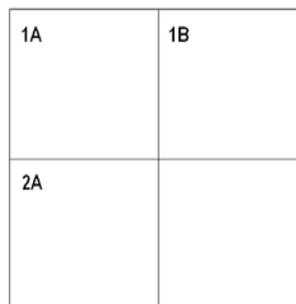


Fig 3. 39 – Montagem da impressão

Opções Import Template to Database e Export Template from Database

O **LogicView for FFB** disponibiliza um mecanismo de Importação/Exportação de Templates de FFB, a fim de permitir o intercâmbio destes arquivos entre estações de trabalho.

Exportando um Template de FFB

Para exportar um Template de FFB escolha a opção **Export Template from Database** no menu **File** e a seguinte janela será mostrada:

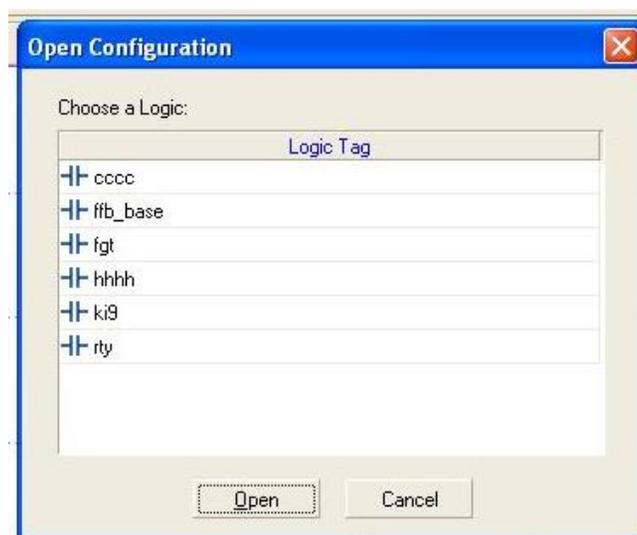


Fig 3. 40 – Selecionando um arquivo template

Basta escolher o tag da lógica a ser exportada e clicar no botão **Open**.

Em seguida será mostrada a caixa de diálogo abaixo:

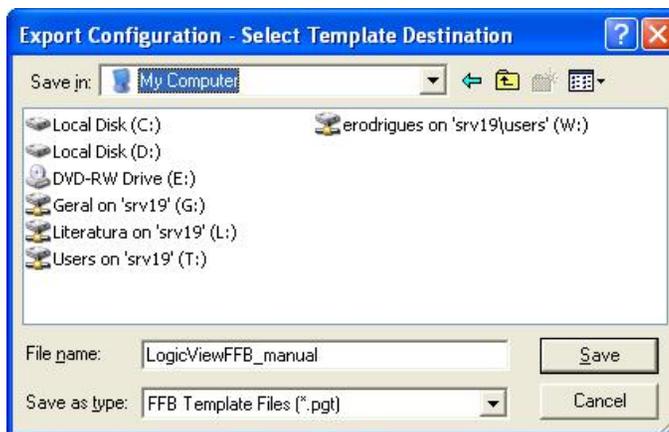


Fig 3. 41 – Selecionando o destino do template exportado

Nesta caixa de diálogo, o usuário deverá escolher o local de destino dos templates a serem exportados e por este motivo a seleção de diretórios e pastas é **livre**. Após escolher uma pasta de destino (ou até mesmo criar uma a partir das ferramentas disponíveis nesta caixa de diálogo), basta clicar em **Save**.

O **LogicView for FFB** irá exportar os arquivos referentes ao template escolhido (com extensões .dpt e .pgt) na pasta de destino selecionada.

Importando um Template de FFB

Para importar um Template de FFB escolha a opção **Import Template to Database** no menu **File** e a seguinte janela será mostrada:

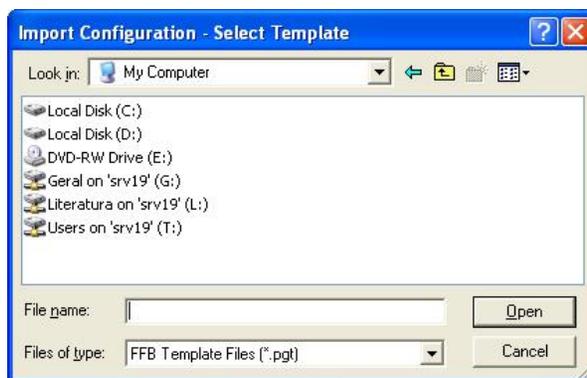


Fig 3. 42 – Selecionando um arquivo template

Nesta caixa de diálogo, o usuário deverá escolher o arquivo de template a ser importado e por este motivo a seleção de diretórios e pastas é **livre**. Após escolher o arquivo a ser importado, basta clicar em **Open**.

Após a importação, será pedido ao usuário um tag para o template em questão.

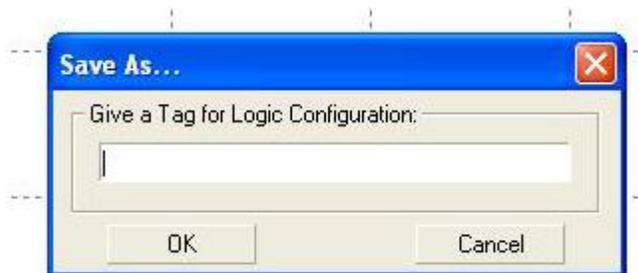


Fig 3. 43 – Informando um tag para o template

O **LogicView for FFB** irá importar arquivos referentes ao template escolhido (com extensões .dpt e .pgt) da pasta de origem selecionada e criará todas as referências necessárias ao gerenciamento do Sistema Integrado. Após o término do processo de importação, o **LogicView for FFB** abrirá o Template de FFB recém-importado.

NOTA

As opções **Import Template to Database** e **Export Template from Database** estão disponíveis apenas quando o **LogicView for FFB** estiver aberto em modo template. Para mais detalhes veja a seção Modo Template.

Menu Edit

Clicando em **Edit**, ou através do atalho ALT+ E, o seguinte menu se abrirá:

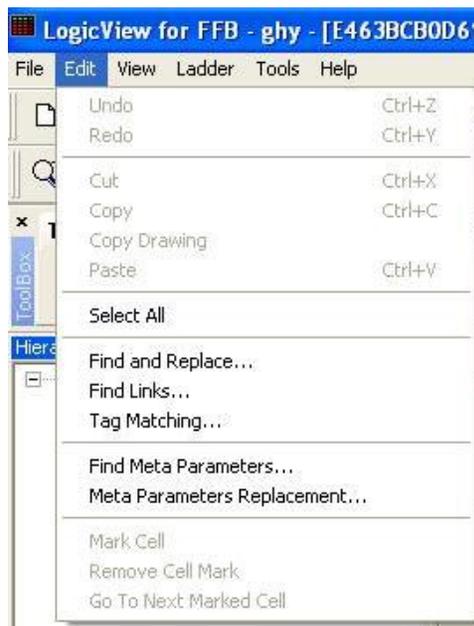


Fig 3. 44 - Menu Edit

Copy/Paste “Inteligente”

O **LogicView for FFB** possui um mecanismo de cópia inteligente de parâmetros dos diagramas lógicos.

As operações comuns no Windows como Copiar, Recortar e Colar (Copy/Cut/Paste) são aplicadas

a um grupo de elementos selecionados em um diagrama lógico (através do botão ) e estão disponíveis no **LogicView for FFB** da seguinte forma:

- **Copy Drawing:** esta opção copia somente o “desenho” da ladder (contatos bobinas, blocos funcionais, etc) removendo tags e links;
- **Cut:** este comando “recorta” o grupo selecionado, removendo-o da área de desenho;
- **Copy:** este comando copia o grupo selecionado, mantendo, além do desenho, todas as suas características **válidas** (veja mais abaixo);
- **Paste:** use este comando para colar o grupo que foi copiado ou recortado.

Ao executar o **Paste**, o **LogicView for FFB** irá avaliar as condições para manter as características válidas do grupo de elementos que será inserido no diagrama:

- **1º Paste executado após um Cut:** neste caso, será mantido, além do desenho, todas as características do grupo (tags e links de blocos funcionais);
- **2º Paste (em diante) após um Cut:** neste caso, o comportamento será idêntico ao **Paste após um Copy** (ver abaixo).
- **Paste após um Copy:** neste caso, o desenho será mantido e o **LogicView for FFB** poderá efetuar algumas das modificações abaixo, dependendo do caso:
 - Contatos e Bobinas: são mantidos exatamente como estavam, com seus tags associados caso eles existam;
 - Blocos Funcionais: links de blocos funcionais com pontos de saída analógicos (FFB ou NetIO) serão removidos (cada ponto de saída só pode ser utilizado uma vez em um link de bloco funcional);
 - Blocos funcionais (links internos): links internos entre blocos funcionais serão mantidos.

IMPORTANTE

Os tags dos blocos funcionais são modificados a cada *paste* para manter a unicidade dos tags. O **LogicView for FFB** automaticamente refaz todas as conexões entre os blocos em função dessa mudança de tag.

Exemplo de comportamento com blocos funcionais:

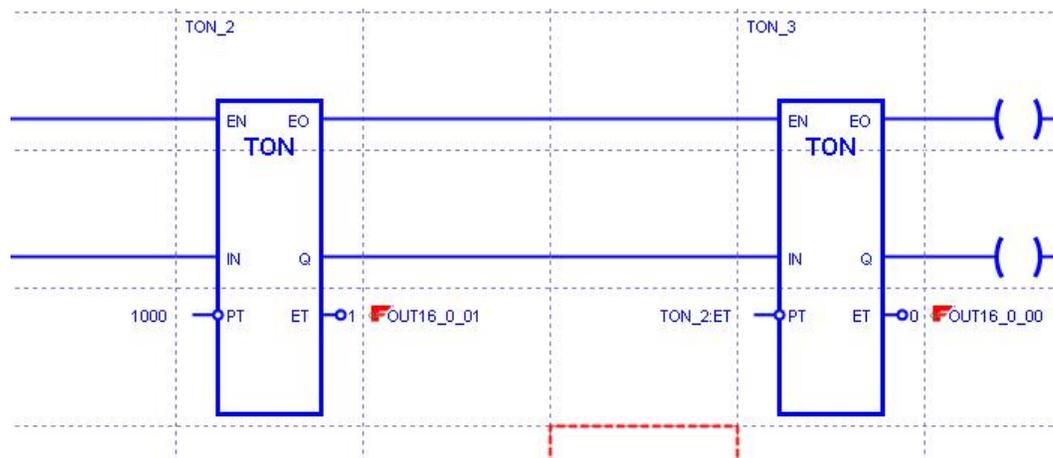


Fig 3. 45 - Exemplo de Copy/Paste inteligente

No diagrama da figura, temos dois blocos funcionais, com um link entre si (**TON2:ET** ligado a **TON3:PT**) e os pontos **ET** dos dois blocos estão ligados a pontos de saída analógicos FFB.

De acordo com o que já foi descrito, a operação de **Cut**, selecionando-se todos os elementos do diagrama acima, seguida de **Paste** irá manter todas as características e o resultado será o mesmo da figura acima.

No entanto, a partir do segundo **Paste** (e portanto os blocos **TON_2** e **TON_3** já estão no diagrama lógico), o resultado será o da figura abaixo:

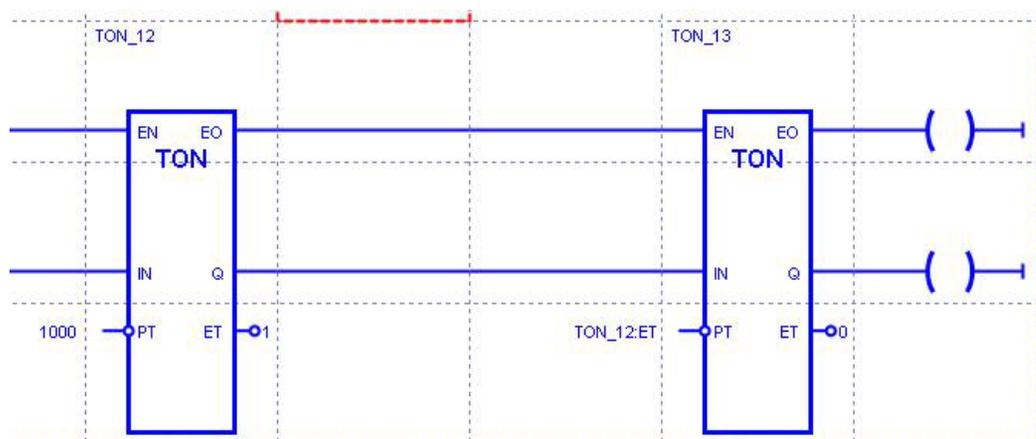


Fig 3. 46 - Exemplo de Copy/Paste inteligente usando blocos funcionais

O **LogicView for FFB** removeu os links dos blocos funcionais com os pontos analógicos de FFB, porque eles já estão sendo utilizados nos blocos **TON_2** e **TON_3**, e renomeou os blocos funcionais para manter a integridade de tags (**TON_12** e **TON_13**). O link entre os blocos foi mantido, agora é **TON_12:ET** ligado a **TON_13:PT**, a mudança de tags dos blocos funcionais foi refletida nos links.

IMPORTANTE

Este mesmo comportamento teria ocorrido com apenas um comando **Copy** seguido de **Paste**.

Num outro exemplo se apenas o bloco **TON_3** da figura 3.45 for selecionado para uma operação de **Copy** seguida de **Paste**, o resultado será o da figura abaixo:

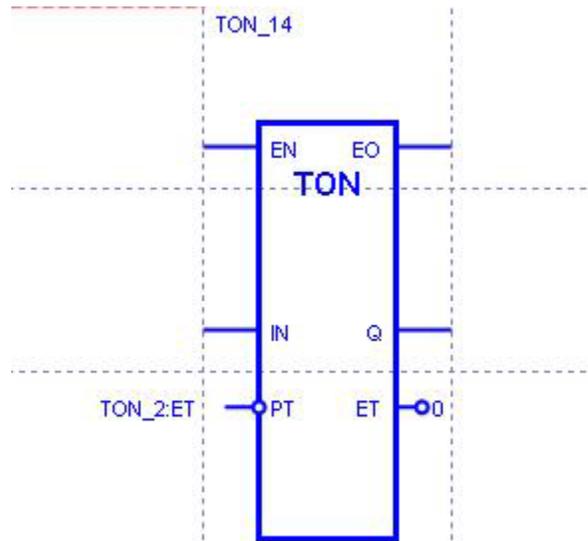


Fig 3. 47 - Exemplo de Copy/Paste inteligente

Assim como no exemplo anterior, o **LogicView for FFB** removeu os links dos blocos funcionais com os pontos analógicos de FFB e renomeou o bloco funcional para manter a integridade de tags.

No entanto, como apenas o bloco **TON_3** foi copiado, o link entre ele e o bloco **TON_2** foi mantido, agora é **TON_2:ET** ligado a **TON_14:PT**.

O usuário poderá selecionar todos os elementos que estão na área de desenho da ladder de uma só vez. Para isto basta clicar em **Edit** → **Select All**.

Find e Replace

O usuário poderá buscar e alterar tags dos elementos da rede ladder com comando **Edit** → **Find and Replace**. A seguinte janela aparecerá:

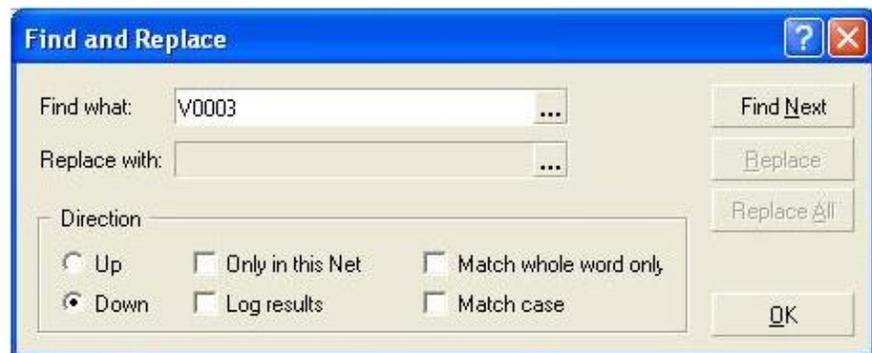


Fig 3. 48 – Comando Find and Replace

O **Find** atua sobre os tags de contatos, bobinas e blocos funcionais e o **Replace** atua apenas sobre os tags de contatos e bobinas. Se o usuário quiser encontrar um tag na rede basta escrevê-lo em **Find what** e depois clicar no botão **Find Next**. O **LogicView for FFB** vai encontrá-lo e a célula onde ele se encontra será selecionada.

Se o usuário quiser substituir um tag deverá selecionar o tag que será substituído, clicar em . Uma janela com os tags disponíveis aparecerá. O usuário deverá escolher aquele necessário à sua aplicação e clicar em **Select**. Veja figura a seguir.

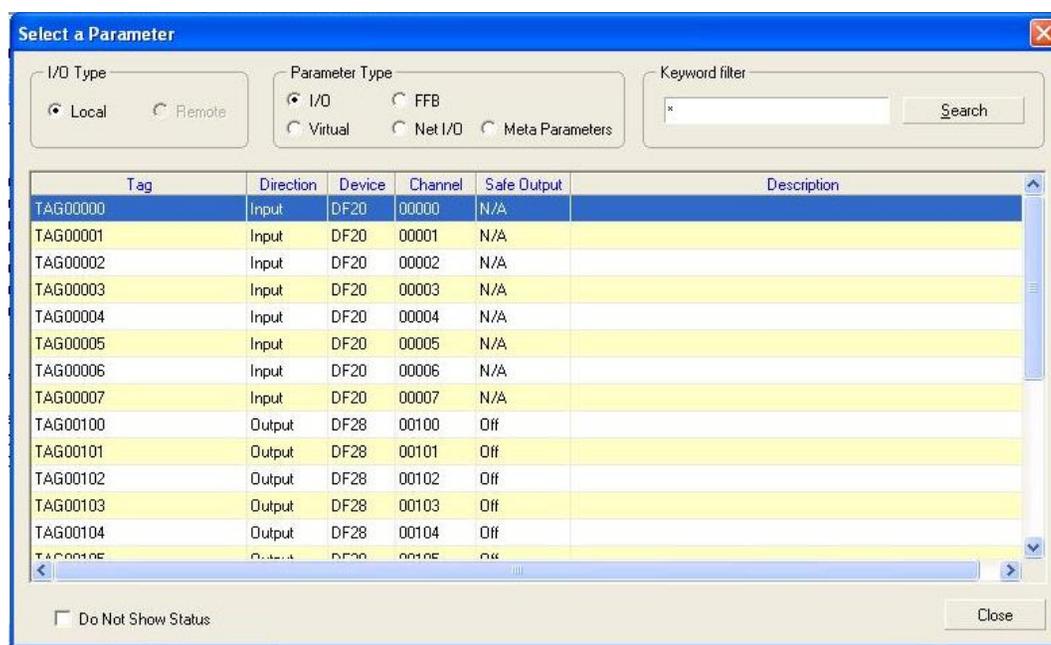


Fig 3. 49 - Selecionando o parâmetro

Assim que o tag for selecionado, o software retornará à janela **Find and Replace** e o usuário poderá substituir o tag clicando em **Replace** (para um elemento) ou **Replace All** (para todos os elementos de mesmo tipo). Para cancelar o comando clique em **Cancel**.

O usuário tem a opção de escolher a direção de busca – **Up** ou **Down**, e se a busca será realizada apenas na net selecionada – **Only in this net**. Ao escolher a opção **Log results** na janela **Output** aparecerá uma lista informando as células em que o tag escolhido foi encontrado. Veja o exemplo na figura abaixo:

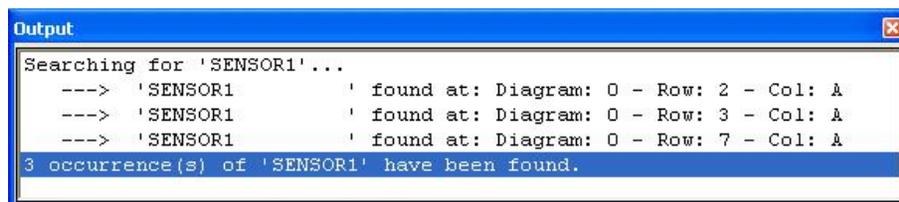


Fig 3. 50 – Lista informando as células em que o tag escolhido foi encontrado

NOTAS

- Os comandos **Find** e **Replace** diferenciam letras maiúsculas de minúsculas e palavras inteiras.
- Os comandos **Undo** e **Redo** só têm ação sobre inserção, remoção e movimentação de elementos da ladder.

Find Links: esta opção permite localizar blocos funcionais que possuam links com o parâmetro selecionado em uma lista.

Ela pode ser acessada a partir do menu **Edit**, de acordo com a figura abaixo:

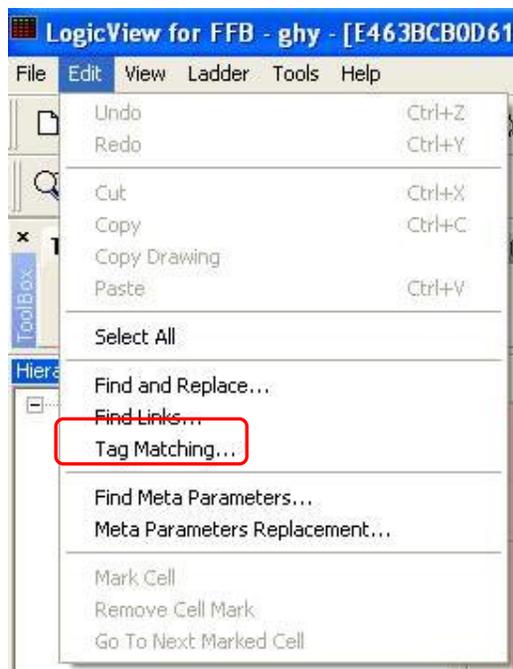


Fig 3. 51 – Opção Find links no menu Edit

Escolhendo essa opção, será exibida a seguinte janela:

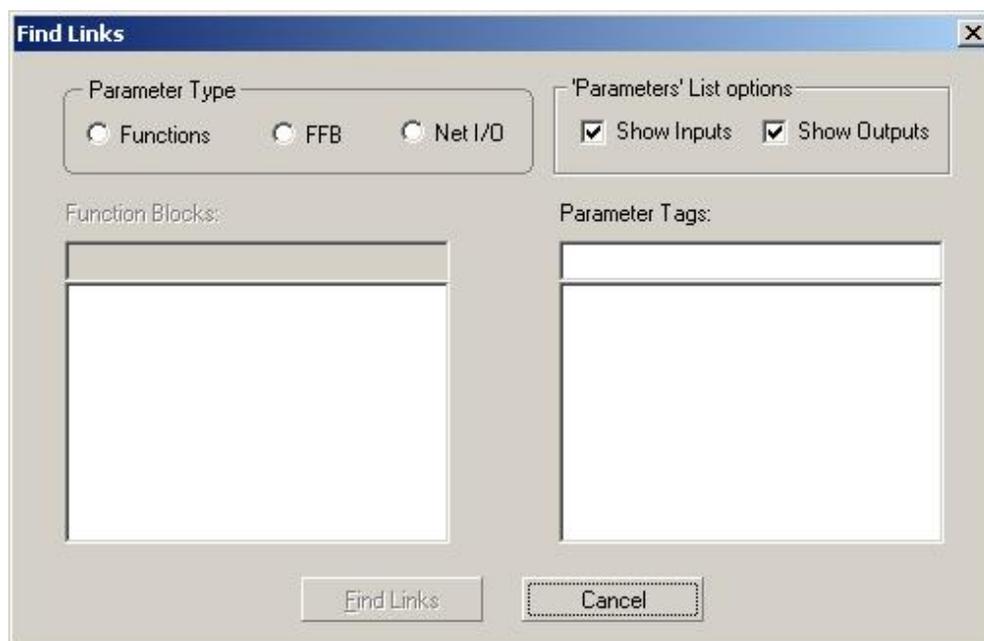


Fig 3. 52 – Janela “Find Links”

Os elementos desta janela são os seguintes:

Parameter Type: nesta caixa, é possível selecionar o tipo dos parâmetros que serão disponibilizados para realizar a busca: parâmetros de blocos funcionais (*Functions*), parâmetros de blocos funcionais flexíveis (*FFB*) e parâmetros de I/O de rede (*NetI/O*);

'Parameters' List options: permite selecionar se a lista de parâmetros irá exibir apenas pontos de entrada (*Show Inputs*) ou de saída (*Show outputs*);

Usando as opções de **Parameter Type**:

- Ao clicar na opção **Functions**, o conjunto de tags de blocos funcionais disponíveis será apresentado na lista do lado esquerdo da tela (*Function Blocks*). Ao selecionar um bloco funcional dessa lista, os tags de seus pontos analógicos serão exibidos na lista do lado direito (*Parameter Tags*).
- Ao clicar nas opções FFB ou NetIO, o conjunto de tags disponíveis será apresentado na lista do lado direito da tela (*Parameter Tags*). Este conjunto pode ser filtrado de acordo com a caixa **'Parameters' List options** (descrita mais acima).

Para efetuar a busca de links, o usuário deve selecionar um tag da lista da direita e clicar em **Find Links**. O mecanismo de busca irá localizar todos os blocos funcionais que contêm links para o ponto selecionado, exibindo os resultados na janela de log de resultados (*Output View*), de acordo com a figura seguinte.

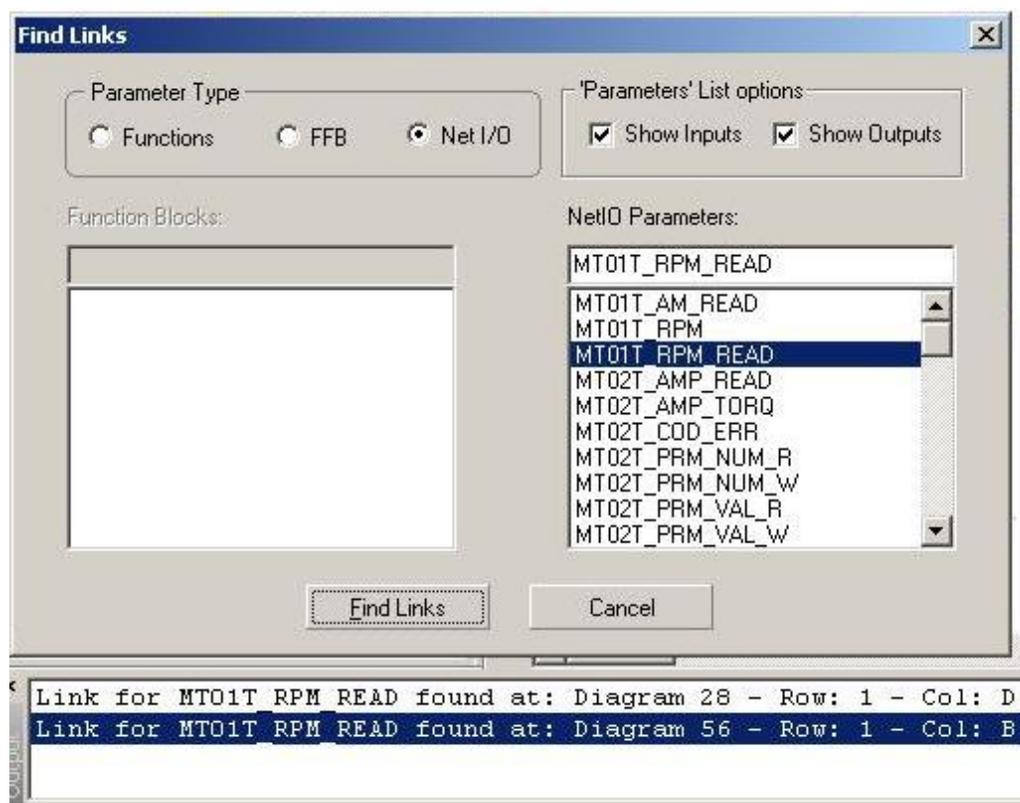


Fig 3. 53 – Janela de resultados de um processo de “Find links”

Na lista de resultados, para navegar até um bloco funcional específico, basta efetuar um duplo clique na linha de interesse. O **Logicview for FFB** irá sinalizar o bloco na janela principal do programa. Caso não sejam encontrados links que utilizam o ponto selecionado, uma janela com a mensagem **"No link(s) found!"** será exibida na tela.

NOTA

Caso o usuário escolha um ponto de SAÍDA (Output), o comportamento do mecanismo de **Find Links** será o descrito acima, localizando todos os blocos funcionais que fazem links com o ponto selecionado.

Entretanto, se o usuário escolher um ponto de ENTRADA (Input), o mecanismo de busca irá localizar o único bloco funcional cujo ponto de saída está “linkado” ao ponto selecionado. Este comportamento é idêntico à função **Go To Out** no menu de contexto de um diagrama ladder.

A opção **Tag Matching** permite **substituir** um conjunto de variáveis nos elementos de um diagrama ladder de forma rápida e eficiente por outro conjunto de variáveis **previamente** definidos no **LogicView for FFB**.

A operação de **Tag Matching** é feita somente para o diagrama selecionado na lista de diagramas e exibido na tela do **Logicview for FFB**.

Este recurso está localizado no menu **Edit** → **Tag Matching**.

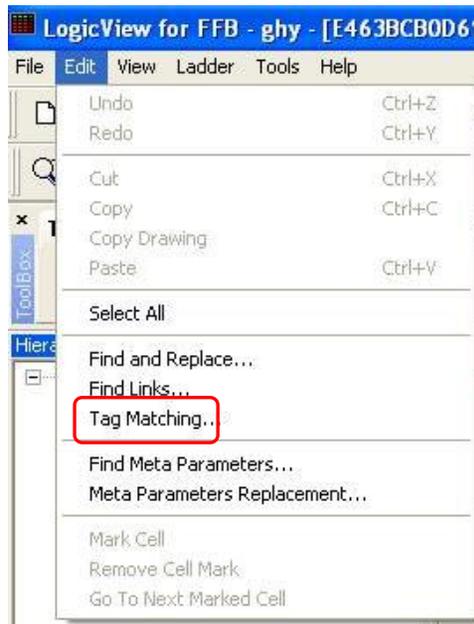


Fig 3. 54 – Opção Tag Matching

Ao selecionar **Tag Matching**, a seguinte tela será exibida:

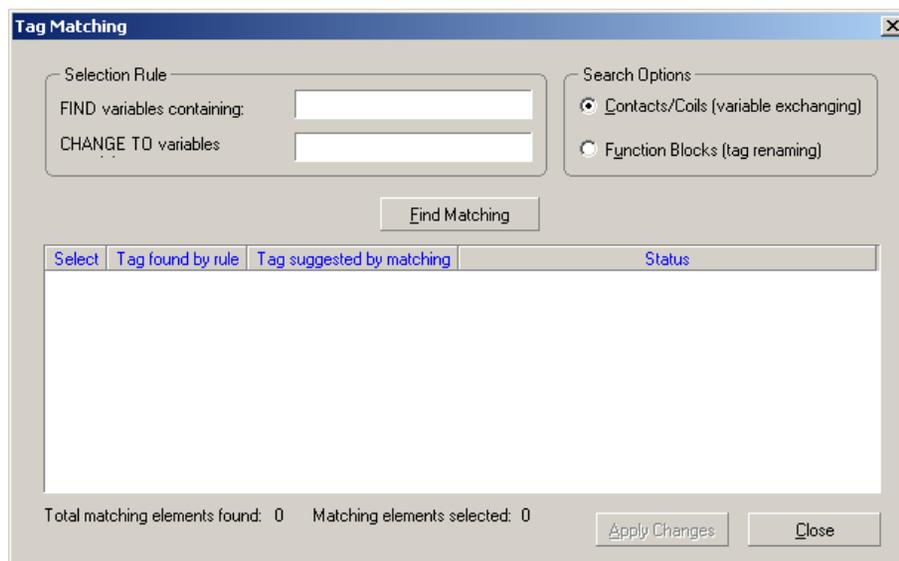


Fig 3. 55 – Caixa de diálogo Tag Matching

Os elementos desta janela são os seguintes:

Search Options: as opções disponíveis para o mecanismo de matching são:

- **Contacts/Coils (variable exchanging):** para contatos e bobinas o mecanismo de *matching* irá localizar aqueles cujos tags estiverem de acordo com a **Selection Rule** (Regra de Seleção) descrita adiante, verificando se é possível a substituição de variáveis em cada caso.
- **Function Blocks (tag renaming):** para blocos funcionais não é possível realizar substituição (troca) de variáveis. A operação disponível permite renomear os tags de um conjunto de blocos funcionais que estiverem de acordo com a **Selection Rule**.

Selection Rule: os campos para edição da regra de seleção permitem que o usuário digite os caracteres dos tags que ele quer localizar e substituir no diagrama, de forma similar a um procedimento de *Find/Replace*.

O primeiro campo de edição **FIND variables containing:** (ou **FIND Functions' Tags containing**, caso a opção **Function Blocks (tag renaming)** esteja ativa) deve ser preenchido com os caracteres referentes a tags que deverão ser localizados para uma possível substituição.

O segundo campo de edição, **CHANGE TO variables**, deve ser preenchido com os caracteres referentes a tags de elementos que deverão **SUBSTITUIR** aqueles localizados pelo primeiro campo (**FIND variables containing**). Caso a opção **Function Blocks (tag renaming)** esteja ativa, este campo de edição apresentará o texto **RENAME Function's Tag to:**, indicando que a operação não será de troca de variáveis e sim de mudança de tag.

Para entender melhor o **Tag Matching**, veja o exemplo a seguir.

O usuário possui um conjunto de variáveis previamente definidas e precisa trocar rapidamente um conjunto de variáveis em contatos e/ou bobinas cujos tags terminarem em '1' por outras variáveis cujos tags terminarem em '10'. Na tela de **Tag Matching** ele preenche da seguinte forma:

| Select | Tag found by rule | Tag suggested by matching | Status |
|-------------------------------------|-------------------|---------------------------|-----------|
| | V0001 | V00010 | NOT FOUND |
| <input checked="" type="checkbox"/> | INBOMB1 | INBOMB10 | FOUND |
| | V0001 | V00010 | NOT FOUND |
| <input checked="" type="checkbox"/> | OUTBOMB1 | OUTBOMB10 | FOUND |
| <input checked="" type="checkbox"/> | OUTBOMB1 | OUTBOMB10 | FOUND |

Total matching elements found: 3 Matching elements selected: 3

Fig 3. 56 – Exemplo de Tag Matching

Clicando em **Find Matching**, o que ocorre é que o mecanismo de associação irá procurar por variáveis em contatos ou bobinas que contenham o caractere '1' em seu tag. Ao encontrar variáveis com essa característica, o mecanismo de matching substitui o caractere '1' por '10' (digitado no campo **CHANGE TO variables**), gerando um novo tag.

A seguir, o mecanismo procura por variáveis com este novo tag gerado e verifica se é possível substituir uma variável pela outra (análise de compatibilidade). As variáveis encontradas serão mostradas na lista, com o status de compatibilidade entre elas.

No exemplo acima na primeira linha, o mecanismo de associação localizou uma variável V0001 (mostrada na coluna "**Tag found by rule**") já definida no **Logicview for FFB**. Pela regra de seleção definida no exemplo, deve-se trocar '1' por '10' o que forma um novo tag V00010. O mecanismo de busca não localizou nenhuma variável definida no **Logicview for FFB** com este tag (para que a troca de variáveis pudesse ser executada), indicado pelo status **NOT FOUND** na última coluna da tabela.

Na segunda linha, o mecanismo de associação encontrou uma variável INBOMB1 e pela regra de seleção, deveria trocar a variável por outra chamada INBOMB10. Como esta variável já havia sido definida no **Logicview for FFB**, o mecanismo de associação localiza a variável e verifica sua compatibilidade para a substituição de variáveis. Neste caso ocorre a compatibilidade e a operação é permitida, indicada pelo status **FOUND** na tabela.

Nos casos em que a variável existir, mas não houver compatibilidade entre elas, por exemplo, se uma variável estiver associada a uma bobina – que só permite variáveis de ‘saída’ – e o mecanismo encontrar uma variável de ‘entrada’ com o tag de acordo com a regra de seleção, esta não é uma substituição válida. A operação não será habilitada e o status será indicado como **FOUND BUT NOT COMPATIBLE** na tabela, como no exemplo a seguir.

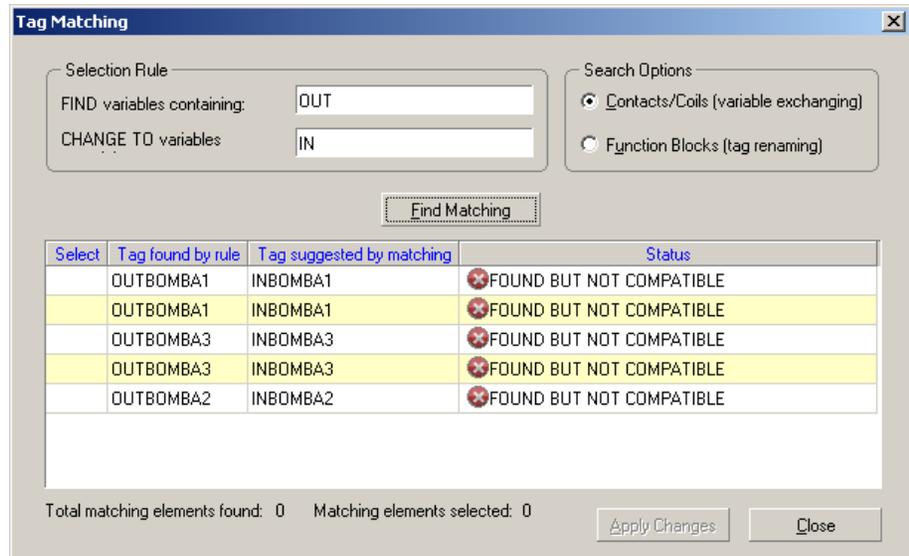


Fig 3. 57 – Exemplo de Tag Matching – verificando a compatibilidade

O usuário poderá selecionar na tabela quais operações de substituição de variáveis devem ser feitas através da coluna **Select**, conforme mostra a figura acima. Ao clicar no botão **Apply Changes**, todas as trocas serão efetivadas no diagrama ladder selecionado.

Exemplo para Blocos Funcionais:

Se o usuário quiser apenas renomear um conjunto de blocos funcionais também é possível através do **Tag Matching**.

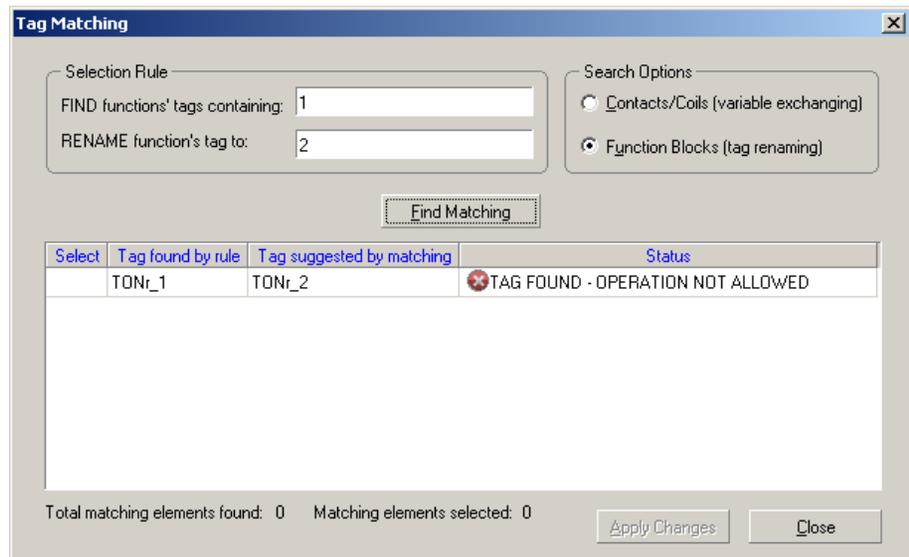


Fig 3. 58 – Exemplo de Tag Matching – renomeando o bloco funcional

No exemplo acima, o usuário tentou localizar todos os blocos funcionais cujo tag contenha ‘1’ para, a seguir, renomeá-los trocando o ‘1’ por ‘2’. No caso do exemplo, de TONr_1 para TONr_2. O mecanismo de associação localizou um tag TONr_2, já existente na configuração e por este motivo não é possível renomear o tag, indicado pelo status **TAG FOUND – OPERATION NOT ALLOWED**.

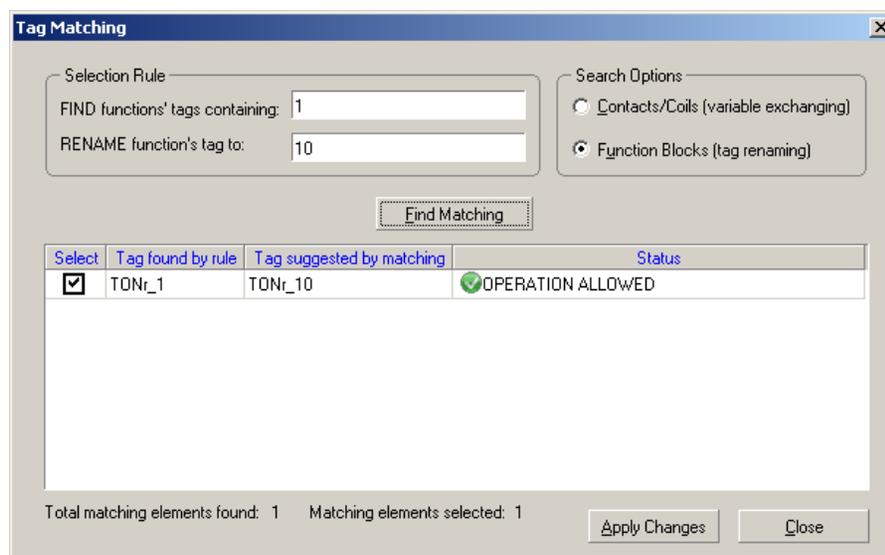


Fig 3. 59 – Exemplo de Tag Matching – renomeando o bloco funcional

No exemplo acima, o usuário tentou localizar todos os blocos funcionais cujo tag contenha '1' para, a seguir, renomeá-los trocando o '1' por '10'. No caso do exemplo, de TONr_1 para TONr_10. O mecanismo de associação não localizou nenhum TONr_10 e, por este motivo, é possível renomear o tag, indicado pelo status **OPERATION ALLOWED**.

Metaparâmetros

O **LogicView for FFB** possui os seguintes tipos de parâmetros convencionais:

- **Reais (I/O):** parâmetros associados a hardware;
- **Virtuais:** variáveis auxiliares para implementação de lógica discreta. São criados no **LogicView for FFB** e pertencem exclusivamente à configuração lógica onde foram definidos.
- **FFB:** parâmetros de entrada e saída de um Bloco Funcional Flexível (FFB), criado através da **Define Parameters Tool (DPT)**;
- **NetIO:** parâmetros de entrada e saída resultantes do mapeamento de uma rede (Profibus, AS-i ou DeviceNet) através da ferramenta **Mapping Tool**.

O metaparâmetro é um elemento especial do **LogicView for FFB** que tem como objetivo facilitar a reutilização de lógicas através de mecanismos especiais de substituição descritos mais adiante. Por esta razão é uma variável de uso temporário e sem um tipo específico que pode ser associado a um elemento discreto (contato/bobina) ou então ser utilizado em links nos blocos funcionais.

Um metaparâmetro é identificado pelo prefixo **#** e pode ser criado manualmente pelo usuário ou automaticamente pelo **LogicView for FFB** na criação de templates e na importação de lógicas da Biblioteca de Lógicas.

Assim como os pontos de FFB e NetIO, os metaparâmetros possuem valor e status, e são divididos da seguinte forma:

- Digital Input (DI);
- Digital Output (DO);
- Analog Input (AI);
- Analog Output (AO).

Ou seja, um metaparâmetro é definido por valor, status e tag e não possui endereço de memória. Por exemplo, para um metaparâmetro com tag BOMBA_1, temos:

- **#BOMBA_1** (valor do metaparâmetro BOMBA_1)
- **#~BOMBA_1** (status do metaparâmetro BOMBA_1)

É possível ter parâmetros convencionais definidos e usados na lógica juntamente com metaparâmetros, caracterizando uma *lógica híbrida*, o que maximiza o desenvolvimento incremental de lógicas e reaproveitamento parcial e total das mesmas.

NOTA

A presença de pelo menos 1 metaparâmetro num diagrama ladder irá bloquear as seguintes operações:

- Simulação;
- Download pelo **Syscon**
- Entrar Online no **LogicView for FFB**

O **LogicView for FFB** irá compilar normalmente uma configuração que contenha metaparâmetros, permitindo que o usuário possa localizar e corrigir erros comuns de estruturação do intertravamento discreto.

No entanto, ao tentar executar as operações não permitidas numa lógica híbrida, mensagens de erro como as abaixo serão exibidas:

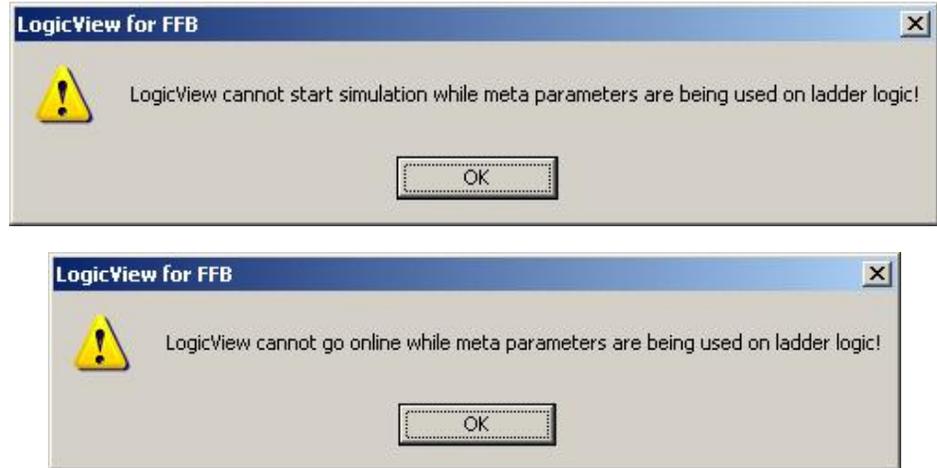


Fig 3. 60 – Mensagens de erro: operações não permitidas em lógicas híbridas

Criando metaparâmetros

Para criar metaparâmetros, basta selecionar na janela **Hierarchy** o item  **Meta Parameters** e definir sua quantidade desejada na janela **Object Properties**, como na figura abaixo:

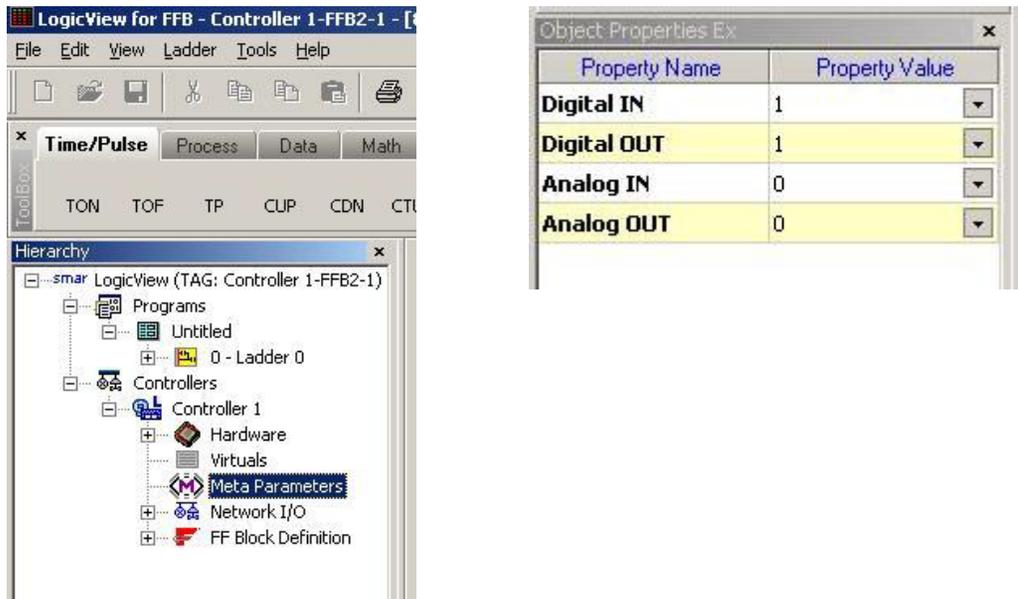


Fig 3. 61 – Criando metaparâmetros

Editando metaparâmetros

Para editar metaparâmetros, pode-se utilizar o **Properties Editor**. Veja figura seguinte.

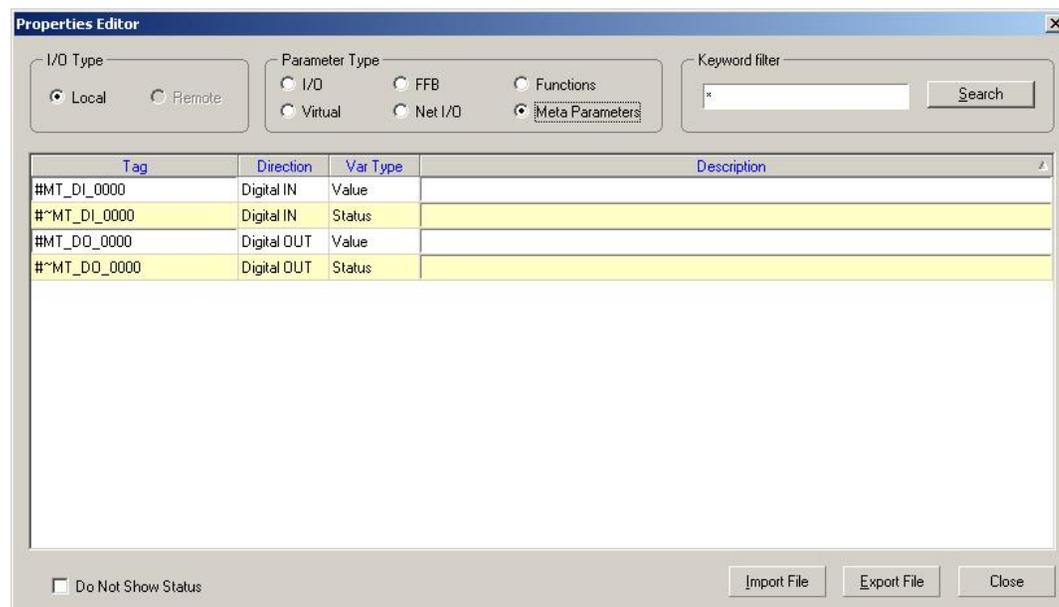


Fig 3. 62 – Editando metaparâmetros

A associação de metaparâmetros a contatos, bobinas ou blocos funcionais se dá exatamente da mesma forma utilizada para os outros tipos (Pontos de I/O, Virtuais, FFB ou NetIO). A lógica resultante da combinação de parâmetros comuns com metaparâmetros pode ser vista na figura abaixo:

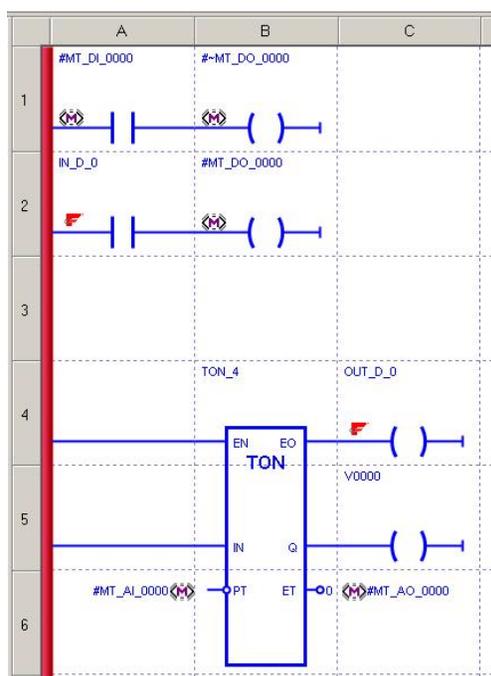


Fig 3. 63 – Exemplo de lógica híbrida, com metaparâmetros

Uma configuração lógica inteira pode ser criada apenas com metaparâmetros, sem a necessidade de especificar informações de hardware, definir pontos de FFB ou mesmo mapeamento de pontos de rede (NetIO). Esta lógica híbrida pode ser "convertida" em uma lógica convencional, através de um mecanismo automático do **LogicView for FFB**, denominado **Meta-Tag Replacement** que será descrito mais adiante.

Localizando Metaparâmetros

Para localizar os metaparâmetros utilizados na sua configuração lógica, o **LogicView for FFB** disponibiliza a opção **Find Meta Parameters...**, no menu **Edit**:

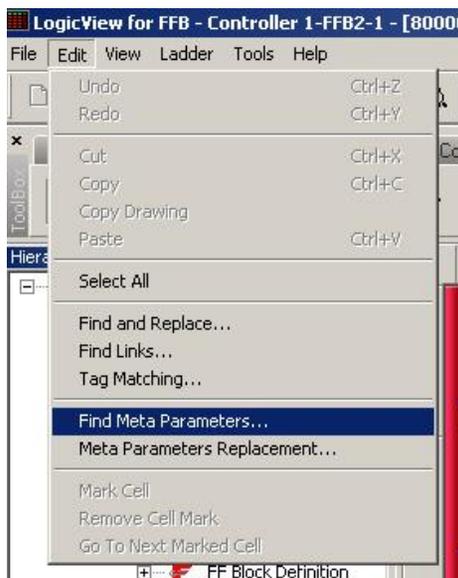


Fig 3. 64 – Localizando metaparâmetros

Ao selecionar esta opção a janela da figura seguinte abrirá. Basta escolher um diagrama específico **Only on this diagram** ou então uma busca completa **All Diagrams**:

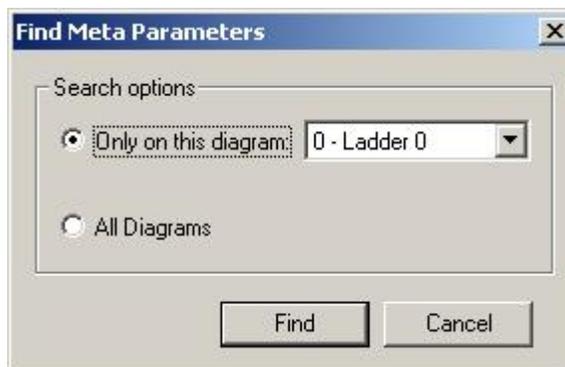


Fig 3. 65 – Janela Find Meta Parameters

Após escolher a opção de busca, basta clicar em **Find** e os resultados da busca serão exibidos na janela **Output**:

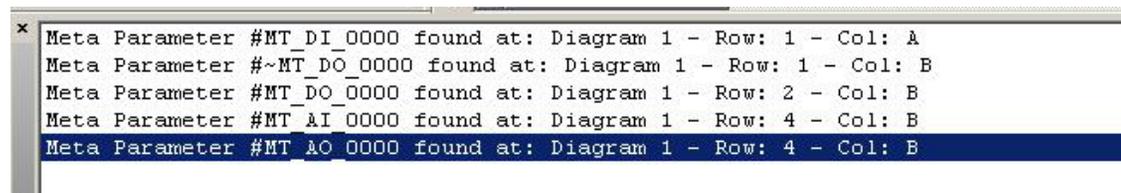


Fig 3. 66 – Resultados da busca por metaparâmetros

Substituindo metaparâmetros por variáveis convencionais

Uma lógica pode conter uma quantidade muito grande de metaparâmetros e em algum momento será necessário substituir os metaparâmetros por parâmetros (ou variáveis) convencionais, para tornar a lógica híbrida numa lógica convencional, em outras palavras, para que a lógica possa ser aplicada normalmente em uma planta de automação.

Para realizar esta operação de substituição, o **LogicView for FFB** possui a opção **Meta Parameters Replacement** no Menu **Edit**:

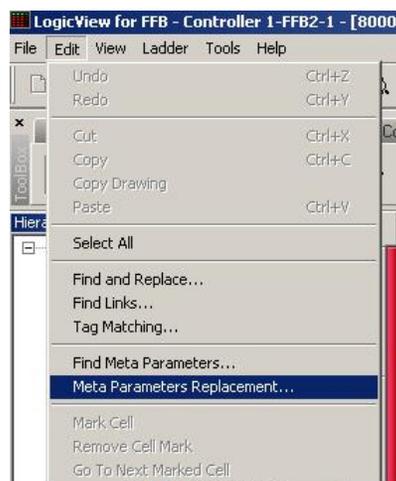


Fig 3. 67 – Opção de substituição de metaparâmetros

Ao escolher esta opção irá abrir a seguinte janela:

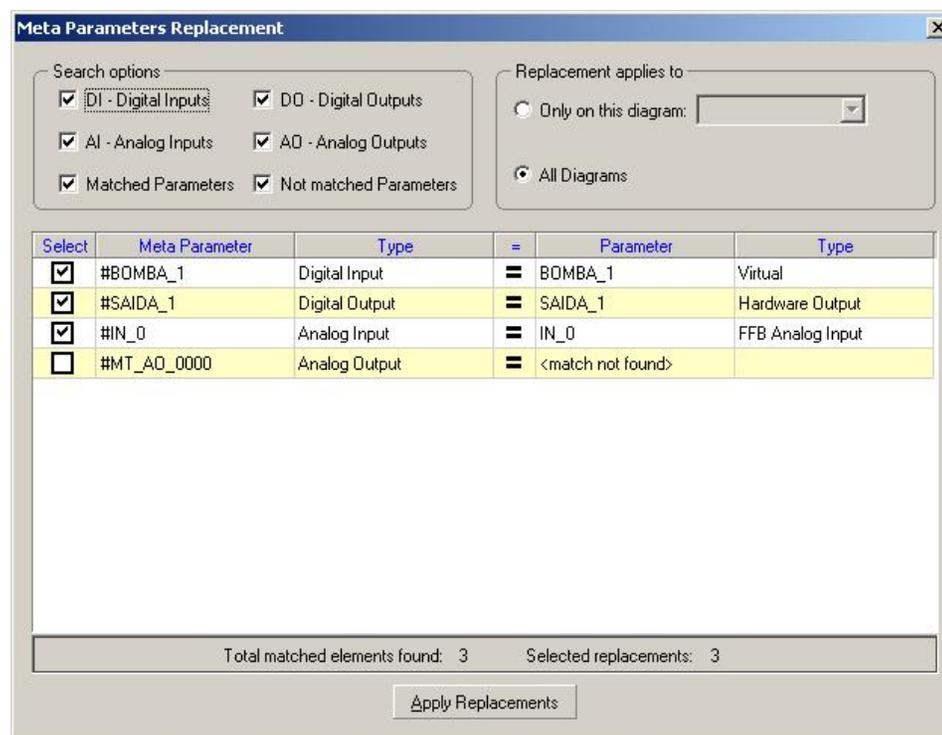


Fig 3. 68 – Janela Meta Parameters Replacement

Ao exibir a janela, o **LogicView for FFB** realiza o processo de **correspondência de tags** em todos os metaparâmetros que estão sendo utilizados na configuração, podendo obter os seguintes resultados:

- **Matched Parameters:** São variáveis (I/O, Virtuais, FFB ou NetIO) cujo tag seja equivalente ao do metaparâmetro. No exemplo da figura, "#BOMBA_1" e "BOMBA_1" são tags equivalentes. Além da correspondência de tags, é feita uma avaliação de compatibilidade de tipos, entre o metaparâmetro e a variável equivalente. Por exemplo, se houver correspondência de tags, mas o metaparâmetro for do tipo AI enquanto a variável correspondente for do tipo DO, a substituição não é válida. Quando ocorre correspondência de tags com variáveis cujos tipos são compatíveis, a tabela da janela é automaticamente preenchida com a "sugestão" de substituição.

- **Not Matched Parameters:** Quando o **LogicView for FFB** não encontra uma variável convencional que possa ser sugerida na tabela, seja porque não encontrou uma correspondência de tags ou porque não existe compatibilidade com a variável encontrada, será exibido **<match not found>** na coluna **Parameter** da tabela.

Nesta janela também estão disponíveis as seguintes opções:

- **Search Options:** fornece opções de filtragem dos metaparâmetros que são exibidos na tabela, pelos seguintes critérios: DI, DO, AI ou AO;
- **Matched Parameters:** exibe apenas os metaparâmetros que possuem uma variável correspondente para substituição;
- **Not Matched Parameters:** exibe apenas metaparâmetros que não possuem correspondência de tag para substituição.
- **Replacement Applies to:** esta caixa permite escolher se as substituições de metaparâmetros por parâmetros convencionais serão aplicadas a todos os diagramas da configuração lógica ou a um diagrama específico.

As colunas da tabela têm os seguintes significados:

- **Select:** Permite marcar ou desmarcar um determinado metaparâmetro para substituição na configuração lógica. Os que não possuem uma variável convencional atribuída para substituição não poderão ser marcados nesta coluna.
- **Meta Parameter:** exibe o tag dos metaparâmetros usados na configuração lógica;
- **Type:** identifica o tipo do metaparâmetro (DI, DO, AI, AO);
- **Parameter:** exibe os tags dos parâmetros convencionais que irão substituir os respectivos metaparâmetros;
- **Type:** identifica o tipo do parâmetro convencional (Virtual, Hardware, FFB DI, etc).

Na tabela é possível ainda fazer a escolha do parâmetro convencional que irá substituir o metaparâmetro na lógica. Portanto, é possível tanto mudar uma sugestão oferecida pelo **LogicView for FFB** quanto preencher as atribuições cujo status seja **<match not found>**.

Para escolher uma variável basta dar um clique direito do mouse na linha desejada da tabela e clicar na opção **Assign a parameter...**:

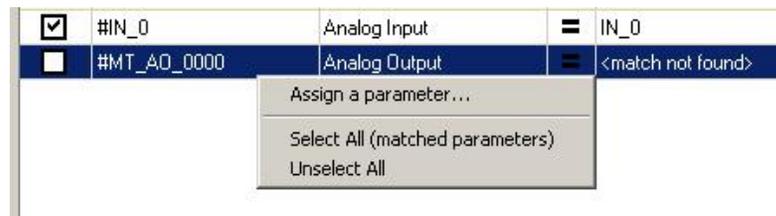


Fig 3. 69 – Atribuição manual de metaparâmetros

Ao clicar em **Assign a parameter...**, a janela de seleção de variáveis será exibida:

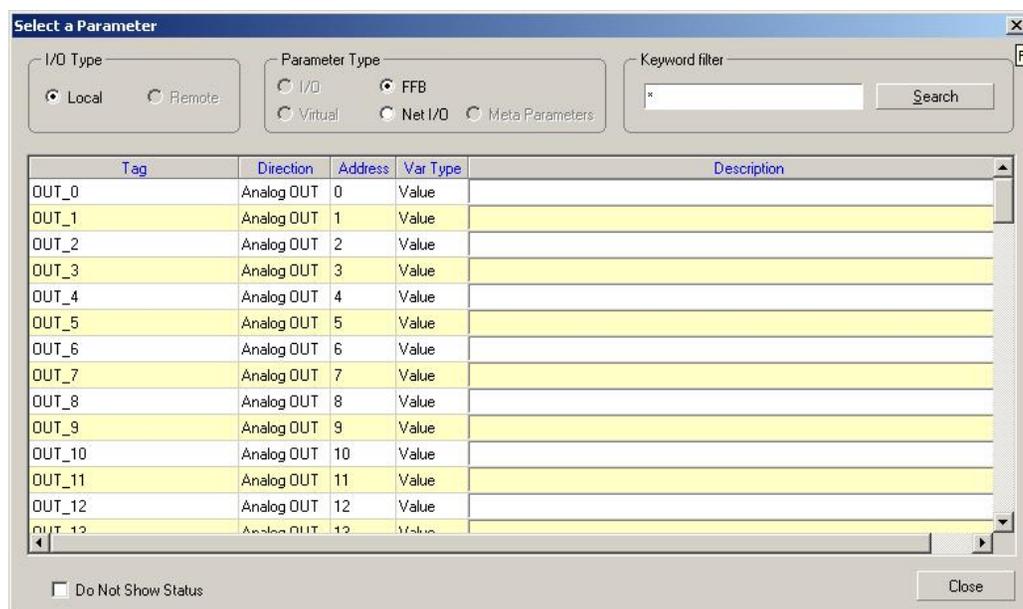


Fig 3. 70 – Seleção de parâmetros

O **LogicView for FFB** filtrará os tipos de parâmetros compatíveis com o metaparâmetro a ser substituído. Para escolher a variável desejada basta efetuar um duplo clique sobre ela.

Para aplicar as substituições selecionadas, clique no botão **Apply Replacements** e o **LogicView for FFB** irá executar as substituições automaticamente na configuração lógica.

NOTA

Todos os metaparâmetros usados na configuração lógica precisam ser substituídos para que a configuração possa ser utilizada normalmente na planta!

Os metaparâmetros constituem um recurso muito útil no **LogicView for FFB**, que expande e flexibiliza a criação e reuso de configurações lógicas permitindo ao usuário um nível de abstração mais alto quando estiver desenvolvendo seu projeto de automação.

Para mais detalhes sobre a utilização de metaparâmetros, consulte o tópico **Bibliotecas de Lógica**.

Se o usuário quiser marcar uma célula basta selecioná-la e ir em **Edit**→ **Mark Cell**. A célula marcada terá um símbolo azul, no canto direito superior, como na figura a seguir (célula B,1):

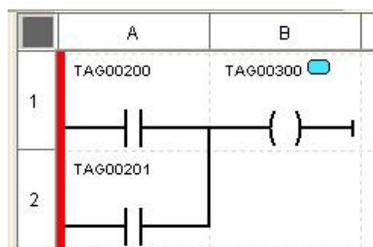


Fig 3. 71 – Marcando uma célula

Para retirar a marca basta ir em **Edit**→ **Remove Cell Mark**.

NOTA

A marca não pode ser salva no arquivo, só estará ativa enquanto o LogicView for FFB estiver sendo executado.

Quando o desenho tem mais de uma célula marcada pode-se usar a opção **Edit**→ **Go to Next Marked Cell**. Para utilizar esta função, deve-se primeiro selecionar uma célula **marcada**. Clicando em **Edit**→ **Go to Next Marked Cell** imediatamente a próxima célula marcada, na seqüência de execução da ladder, será selecionada e ficará piscando com o fundo preenchido de amarelo como na figura a seguir.

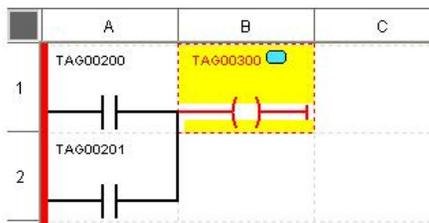


Fig 3. 72 - Go to next cell marked

Menu View

Clicando em **View**, ou através do atalho ALT+ V, o seguinte menu se abrirá:



Fig 3. 73 - Menu View

O menu **View** tem opções de visualização da configuração de hardware e do código gerado. Estes itens serão descritos detalhadamente logo abaixo.

O menu **View** também oferece opções de habilitar ou desabilitar os diversos tipos de barras de ferramentas: Main, Hierarchy, Object Properties, Output, Zoom e Toolbox. Para habilitá-las ou desabilitá-las basta clicar na opção desejada. Estas opções serão descritas com detalhes no tópico **Barras de Ferramentas**.

Hardware configuration

Ao clicar em **View**→**Hardware Configuration** o **LogicView for FFB** exibirá uma janela mostrando a ocupação dos racks e seus slots, e também, quais estão disponíveis. A configuração dos racks poderá ser alterada. Ao clicar no slot desejado uma lista de opções será aberta como mostrado na figura abaixo. A configuração de hardware será mais detalhada no tópico **Hierarchy**→ **Hardware**.

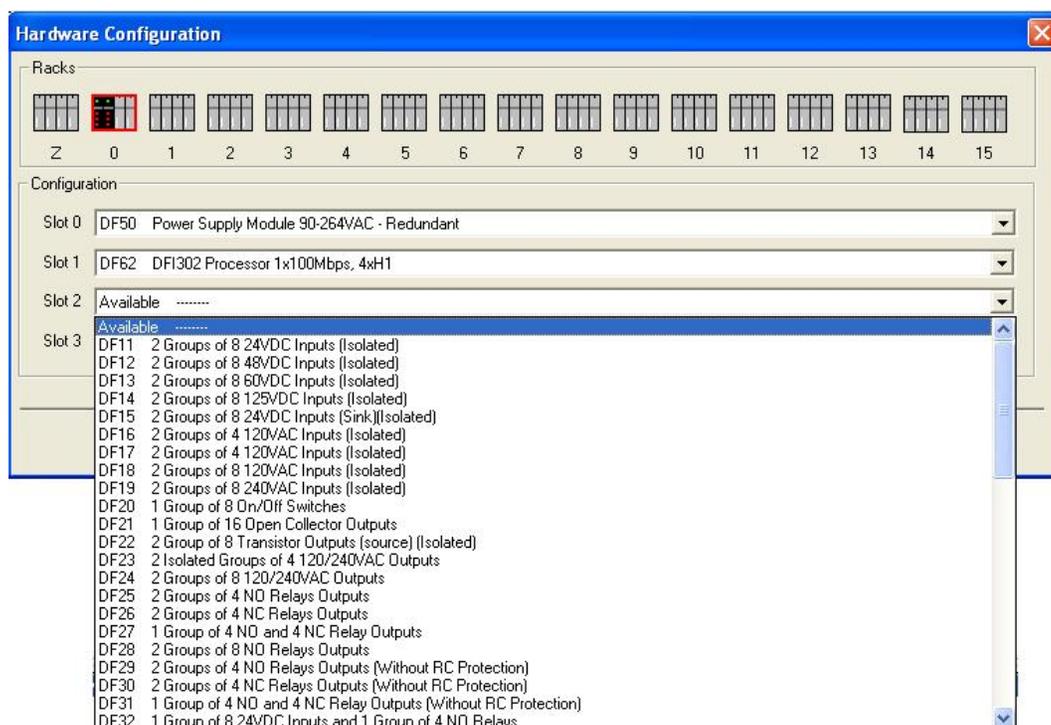


Fig 3. 74 - Janela de Configuração do Hardware

Code Generated

Code Generated é o pseudocódigo que é gerado pelo **LogicView for FFB** e enviado por ele para o device, via download, ou para o próprio **LogicView for FFB**, no caso da simulação, sendo executado pela máquina virtual 1131. Normalmente esta informação só será usada para debug. Em caso de falha, o usuário pode salvar esta informação em um arquivo e enviar ao suporte técnico da Smar.

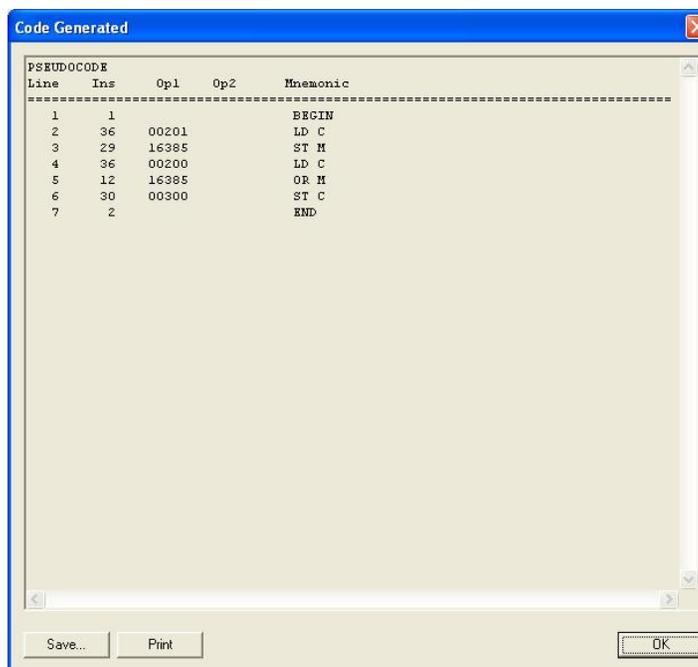


Fig 3. 75 - Janela Code Generated

Menu Ladder

Clicando em **Ladder**, ou através do atalho ALT+ L, o seguinte menu se abrirá:

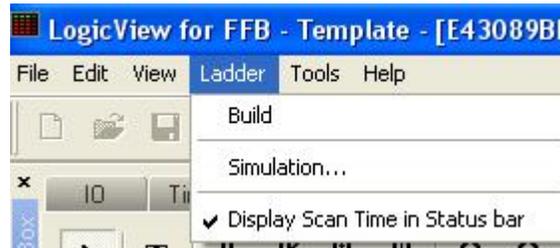


Fig 3. 76 - Menu Ladder

O **Build** é o comando para geração do pseudocódigo que será executado pela máquina virtual 1131. O build é quem gera o código mostrado em **View** → **Code generated**.

A opção **Simulation** está disponível apenas quando se está **offline**. Ao clicar em **Simulation** uma janela se abrirá mostrando a configuração dos racks. Os contatos aparecerão zerados na área de desenho da ladder. Veja a figura abaixo. Neste exemplo apenas o Rack 0 está sendo utilizado.

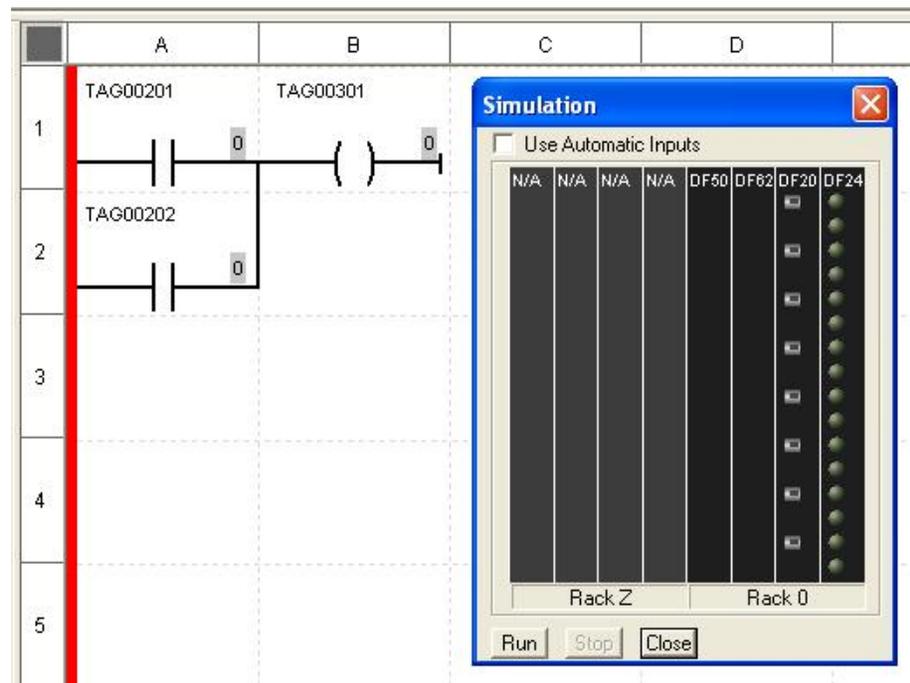


Fig 3. 77 - Janela Simulation

Ao clicar em **Run**, na parte inferior da janela, a simulação começará. Para os I/O reais os valores podem ser alterados na própria tela. Para isto basta clicar na entrada desejada e lhe será atribuído o nível alto (1). Na janela **Simulation** o nível alto das entradas é representado pela cor vermelha e nas saídas o nível alto é representado pela cor verde.

Na área de desenho da Ladder a simulação será apresentada com nível alto (1) em verde e nível baixo (0) em vermelho. Veja figura a seguir.

NOTA

Qualquer módulo pode ser usado na simulação com exceção dos módulos de temperatura, pulso e entrada analógica que não podem ser simulados nesta versão do **LogicView for FFB**.

Se na simulação houver alguma função que acesse esses módulos (MAI, TEMP, ACC e ACC_N) as saídas analógicas dessas funções se manterão sempre em zero.

A opção **Display Scan Time in Status bar** estará sempre ativa e não poderá ser desabilitada. O **Scan Time** é mostrado na barra inferior do **LogicView for FFB** (Status Bar) e reflete o tempo de varredura (scan time) da lógica, que é o tempo que um ciclo da lógica demora para ser executada no equipamento. Para escolher o equipamento do qual o scan time será monitorado, deve-se clicar

no ícone **Stop/Run** , ao lado do ícone usado para se conectar ao **Server**. Além de servir para disparar ou parar a execução da ladder no equipamento, o ícone **Stop/Run** serve para habilitar o pedido de tempo de scan, basta apenas definir o equipamento. Sempre aparecerá apenas um equipamento, que é aquele no qual a lógica foi descarregada.

Para obter a informação sobre o tempo de scan (**Scan Time**) basta estar conectado ao **Server**. O tempo aparecerá na barra de Status.

Menu Tools

Clicando em **Tools**, ou através do atalho ALT+ T, o seguinte menu se abrirá:

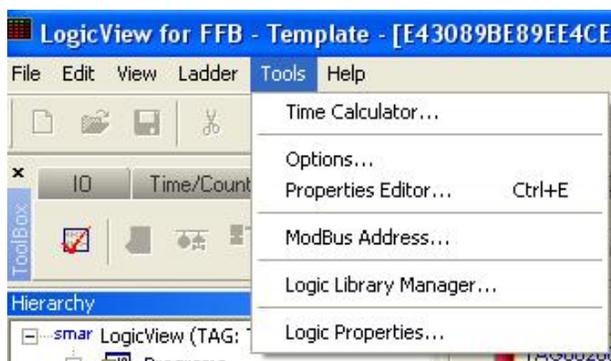


Fig 3. 80 - Menu Tools

Na opção **Time Calculator** o usuário poderá converter valores de tempo dados em horas, minutos, segundos e milissegundos para milissegundos, segundos, minutos ou horas. O usuário deverá entrar com o valor a ser convertido em **From**, escolher a unidade de conversão em **To** e depois clicar em **Convert**. O resultado da conversão será dado em **Result**.

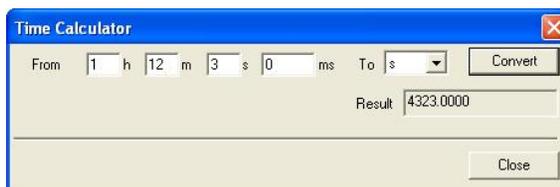


Fig 3. 81 - Time Calculator

Na opção **Options** o usuário poderá configurar a aparência da área de trabalho e das conexões. Na figura abaixo, na aba **General**, o usuário configurará a cor do fundo da área de desenho, a cor do grid, das quebras de página, da seleção de objetos e das células da **Object properties**. Além disso, nesta aba poderão ser configurados se o grid será mostrado, o rolamento automático, se os endereços dos parâmetros dos blocos funcionais e os endereços Modbus serão visualizados na **Object Properties** e o número de operações Undo/Redo. O usuário poderá desabilitar a confirmação sobre parar ou não a execução da CPU quando o download da configuração for feito via **Syscon**. É possível ao usuário decidir se será mostrado, só o Scan Time ou só o Sync Time ou ambos alternadamente na barra de Status.

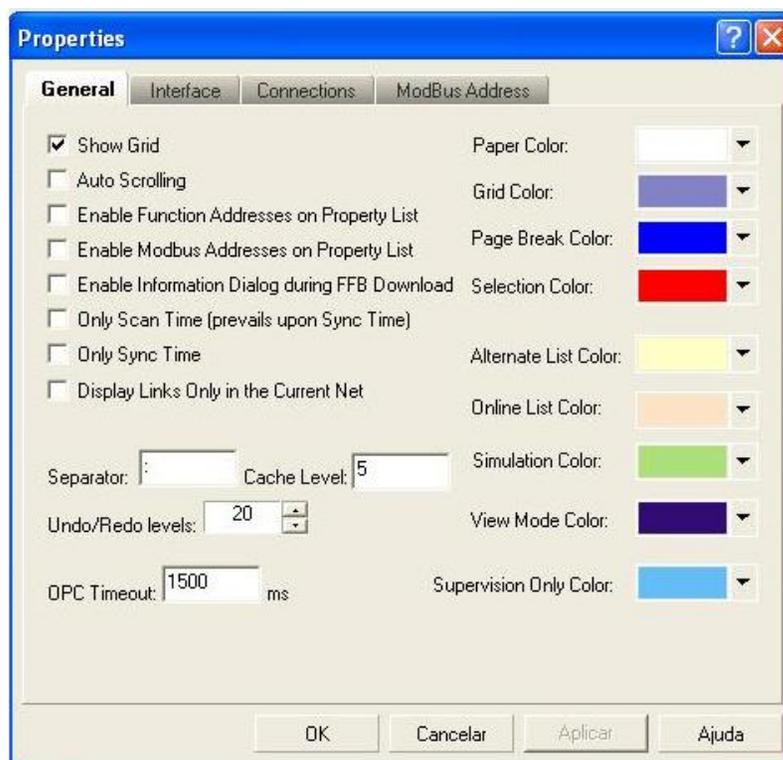


Fig 3. 82 - Configurando a aparência da área de trabalho (1)

O usuário poderá definir o símbolo de separação de tags quando estes são visualizados no **TagView**, por exemplo. O símbolo default é : (dois pontos). Alguns caracteres não são aceitos. Caso o usuário tente colocar um símbolo não aceito a mensagem **Invalid Char** aparecerá. O parâmetro **OPCTimeout** indica o tempo necessário que o LogicView for FFB deve esperar pela resposta do OPC Server a um *request* feito para a CPU na qual o LogicView for FFB esteja conectado. É especialmente útil nos casos em que a comunicação com a CPU é *wireless*.

O usuário poderá escolher o “Cache Level” que define o número máximo de ladders que estarão em cache durante a supervisão. Este valor deve estar entre 1 e 9.

Caso a opção **Enable Information Dialog during FFB Download** esteja marcada, durante o processo de download (descrito mais adiante neste capítulo) será exibida uma tela de mensagem, de caráter informativo, de acordo com a imagem abaixo:

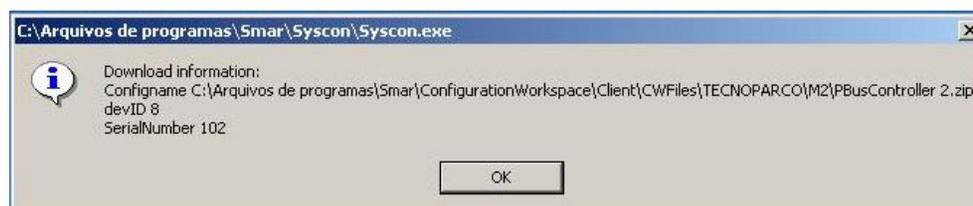


Fig 3. 83 – Informação durante download

Na aba **Interface** poderá ser configurada a fonte dos textos que aparecem na área de desenho assim como a sua cor. A cor dos elementos da ladder também poderá ser alterada. No lado direito da janela estão as opções de configuração das cores usadas na simulação e no modo online. Além disto pode ser configurado se os ícones que representam os tipos de parâmetros ou tipos de links utilizados serão visualizados ou não na área de desenho. Veja figura seguinte.

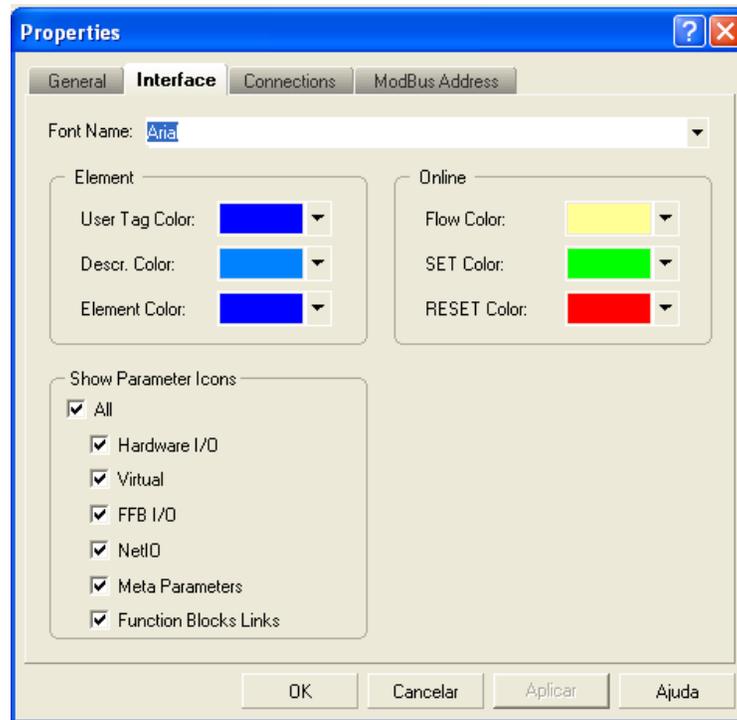


Fig 3. 84 - Configurando a aparência da área de trabalho (2)

Cada tipo de parâmetro ou link possui um ícone específico. Para a representação do NetIO, há um ícone para cada tipo de protocolo suportado: ProfiBus, DeviceNet e AS-i.

- V - ponto virtual
- H - ponto de E/S
- F <azul> - ponto de função
- F <vermelho> - ponto de FFB
- P - ponto Profibus
- D - ponto DeviceNet
- A - ponto AS-i

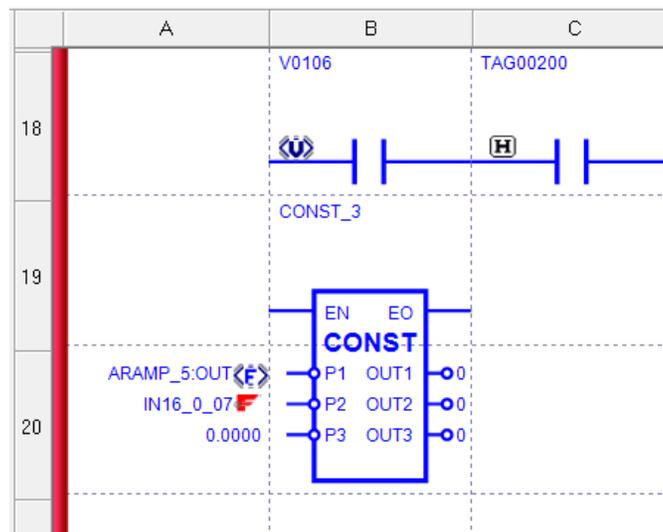


Fig 3. 85 – Ícones informando os tipos dos parâmetros e links

Na figura seguinte, na aba **Connections**, o usuário poderá configurar o **Scan Time** em segundos. Esta opção indica de quanto em quanto tempo será requisitado o **Scan Time** e este deverá estar entre 10 e 60 segundos.

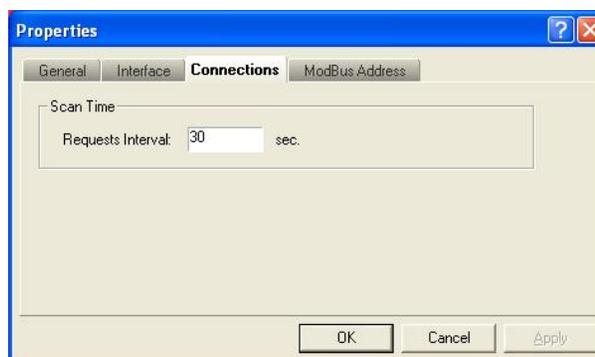


Fig 3. 86 - Configurando o Scan Time

Atribuição dos endereços Modbus

IMPORTANTE

- As funções descritas neste tópico só estão disponíveis para os controladores DF73, DF75, DF79, DF81, DF89, DF95 e DF97.
- Para que as funções relativas ao Modbus funcionem no DF73, DF75, DF79, DF81, DF89, DF95 e DF97 é necessário que o bloco **MBCF** seja configurado no Syscon com os parâmetros corretos da comunicação serial, com o **DEVICE_ADDRESS** correto e que o parâmetro **ON_APPLY** esteja configurado como **Apply**. Esses controladores sempre serão um Slave serial/TCP simultaneamente. Para maiores detalhes consulte o manual de Blocos Funcionais.
- As funções relativas ao Modbus também estão disponíveis para os controladores DF62 e DF63, funcionando exatamente como para os outros controladores. No entanto, é necessário um firmware específico, pois tanto o DF62, como o DF63, também pode ter suas funções Modbus funcionando via blocos funcionais do Syscon. As duas maneiras são mutuamente exclusivas, o firmware define com qual modo o controlador trabalha.

Modbus Address

Nesta opção (**Tools** → **Options** → **ModBus Address**) o usuário poderá escolher o modo de endereçamento Modbus. A opção default é **Automatic**. Se o usuário estava em modo **Automatic** e muda para **Manual** o **LogicView for FFB** mantém os endereços gerados pelo modo **Automatic**, mas estes podem ser alterados de acordo com a necessidade da aplicação.

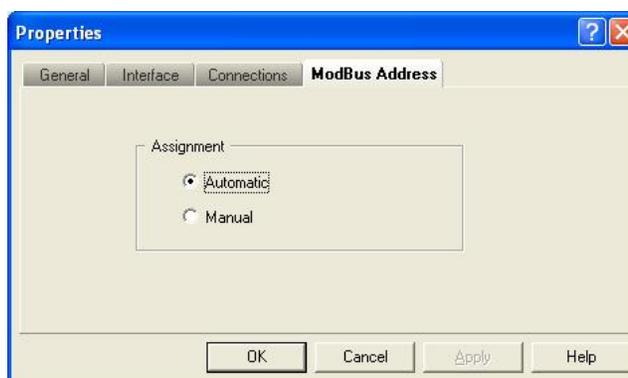


Fig 3. 87 - Configurando o tipo de endereçamento Modbus

Caso esteja em **Manual** e volte para **Automatic** os endereços cadastrados no modo anterior serão perdidos e ficarão no padrão do modo **Automatic**. O usuário será alertado pela seguinte mensagem.

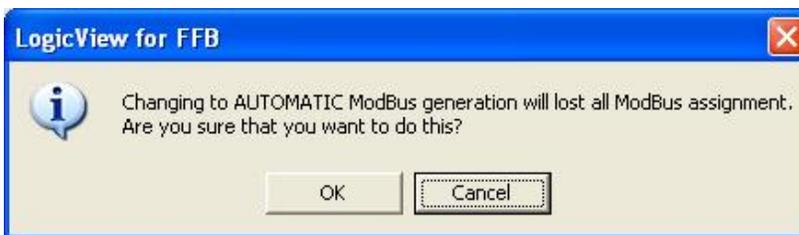


Fig 3. 88 – Mudando o endereçamento Modbus de Manual para Automatic

Na tabela a seguir estão as faixas de endereços Modbus que são utilizados para cada tipo de elemento. Os valores analógicos FLOAT ou LONG (formados por 4 bytes) utilizam dois endereços Modbus consecutivos.

| Item | Endereço inicial | Endereço final |
|---|------------------|----------------|
| Entrada Discreta de IO | 10001 | 11024 |
| NetIO DI (Value), DI, DO, AI, AO (Status) Entrada | 10001 | 11024 |
| Saída Discreta de IO | 1 | 1024 |
| NetIO DO (Value) Saída | 1 | 1024 |
| FFB_DI, DI64 (Value/Status) AI, AI16 (status) Entrada | 11025 | 11536 |
| FFB_DO, DO64 (Value/Status) AO, AO16 (status) Saída | 1025 | 1536 |
| FFB_AI, AI64 (Value) Entrada | 30001 | 30511 |
| NetIO AI (Value) Entrada | 30001 | 30511 |
| FFB_AO, AO16 (Value) Saída | 40001 | 40511 |
| NetIO AO (Value) Saída | 40001 | 40511 |
| Variáveis Virtuais Saída | 1537 | 5999 |
| Blocos Funcionais Entrada | 40513 | 44997 |
| Blocos Funcionais Saída | 40513 | 44997 |
| Blocos Funcionais Internos | 40513 | 44997 |

Em modo **Manual** se forem retirados elementos da ladder ficarão gaps, que são intervalos de endereços vazios. Qualquer elemento novo inserido estará com endereço Modbus zerado e o usuário deverá fazer a inserção.

Em modo **Automatic**, se existirem gaps, o **LogicView for FFB** irá preenchendo-os à medida que novos elementos são inseridos na configuração.

| DICA |
|--|
| Uma forma de eliminar os gaps no modo Automatic é fazer o seguinte procedimento: Em Tools → Options → ModBus Address mudar de Manual , clicar Ok , voltar para Automatic e clicar Ok novamente. |

Quando o usuário clicar em **Tools**→**ModBus Address** e escolher o tipo de elemento uma janela como a seguinte irá aparecer.

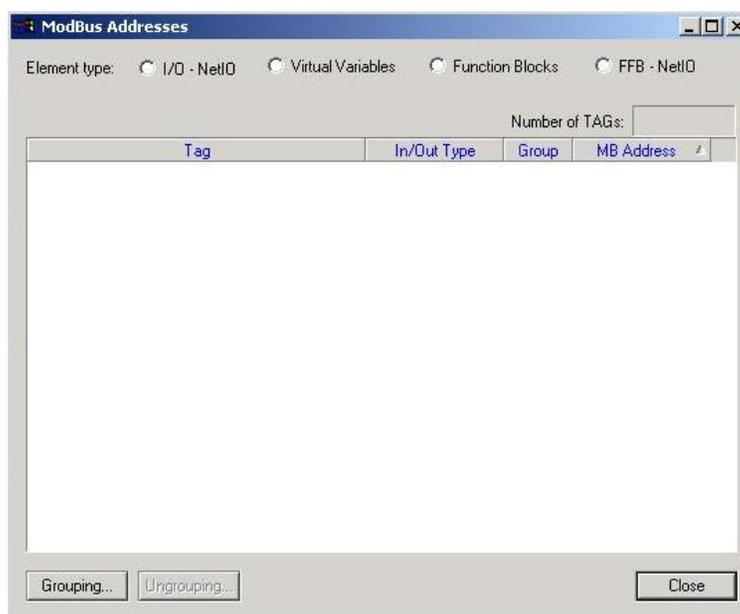


Fig 3. 89 – Visualizando o endereçamento Modbus

Na figura anterior é possível ver os tags, os tipos, a que grupo pertencem e os endereços Modbus dos elementos da configuração.

Atribuindo endereços no modo Manual

Grouping e Ungrouping

Por default é criado um grupo para cada tipo de elemento (Entradas ou Saídas). Caso o usuário necessite podem ser criados novos grupos, também é possível removê-los.

Para criar um novo grupo é preciso primeiro disponibilizar endereços Modbus. Selecione os tags que deverão formar o novo grupo (a seleção pode ser feita no modo padrão Windows com as teclas Shift ou Ctrl) e clique em **Ungrouping**. A seguinte mensagem aparecerá.



Fig 3. 90 – Removendo grupos

Confirme ou cancele a operação. Caso os endereços não estejam livres a seguinte mensagem aparecerá.



Fig 3. 91 – Erro ao criar grupos (1)

Os tags selecionados devem ser do mesmo tipo (Entradas ou Saídas), caso contrário a seguinte mensagem aparecerá.

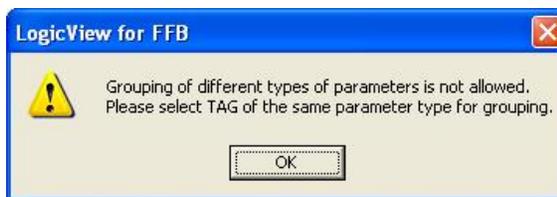


Fig 3. 92 – Erro ao criar grupos (2)

Com os endereços livres novos grupos podem ser criados, basta clicar em **Grouping** e a seguinte janela aparecerá.

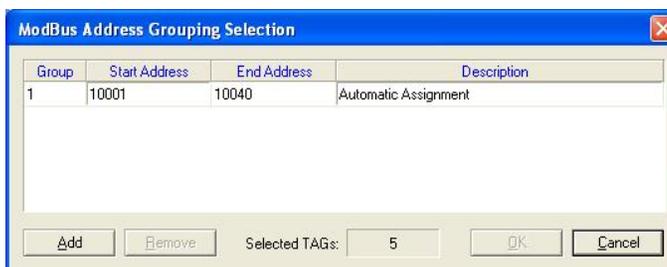


Fig 3. 93 – Criando ou atribuindo grupos

Clique em **Add** e um novo grupo será criado. O usuário poderá definir o endereço inicial da faixa respeitando os valores predefinidos na tabela de endereços Modbus. Caso o endereço inicial esteja fora da faixa determinada, mensagens como a seguinte aparecerão.

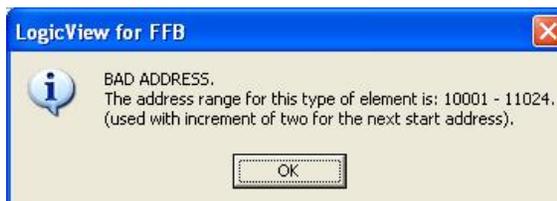


Fig 3. 94 – Erro ao atribuir endereços aos grupos (1)

Além de respeitar as faixas pré-definidas, o usuário deverá estar atento para não atribuir endereços iniciais pares aos grupos de elementos com entradas ou saídas analógicas. Se fizer isso a seguinte mensagem aparecerá.

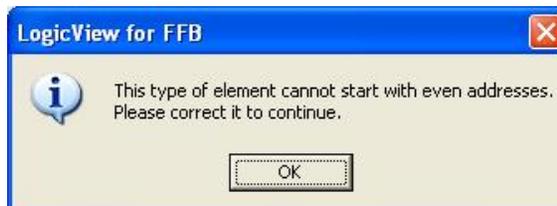


Fig 3. 95 – Erro ao atribuir endereços aos grupos (2)

Se a quantidade de elementos selecionados ultrapassar a faixa livre de endereços a seguinte mensagem aparecerá. Redefina os endereços ou os grupos.



Fig 3. 96 – Erro ao atribuir endereços aos elementos

Para atribuir um grupo a um elemento, com endereço livre, basta selecioná-lo, clicar em **Grouping**, depois no grupo desejado e dar um duplo clique ou clicar em **Ok**.

Se um grupo já está definido e mais elementos forem adicionados e não couberem na faixa, aparecerá a seguinte mensagem.

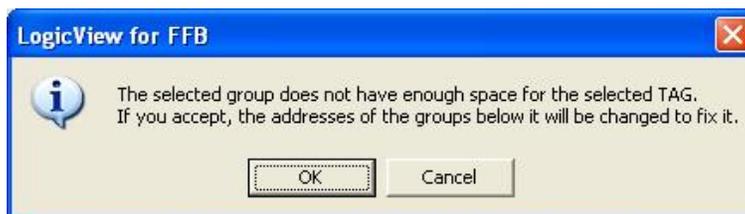


Fig 3. 97 – Espaço insuficiente no grupo

Clicando em **Ok** os endereços serão reorganizados e em **Cancel** a operação é cancelada.

Os grupos podem ser removidos, no entanto, todos os elementos associados a estes grupos ficarão sem atribuição de endereços, ou seja, com endereços vazios. Para remover um grupo basta selecioná-lo e clicar em **Remove**. Veja figura seguinte.

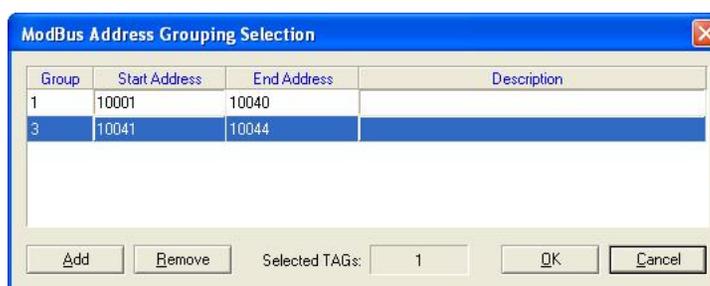


Fig 3. 98 – Removendo um grupo

Uma mensagem de alerta aparecerá confirmando a operação. Veja figura seguinte.



Fig 3. 99 – Alerta ao remover um grupo

Os endereços associados a um grupo podem ser alterados, no entanto, isto implica que os endereços dos grupos subsequentes serão alterados também. Uma mensagem de alerta aparecerá para que o usuário confirme a operação.



Fig 3. 100 – Alerta ao modificar endereço de um grupo

NOTAS

- Quando módulos de hardware são retirados da configuração os tags associados às suas entradas e saídas terão seus endereços Modbus liberados.
- Os tags das entradas e saídas dos blocos funcionais só aparecerão na lista de endereços Modbus se os sinais de E/S forem analógicos.
- O limite de endereçamento Modbus para as funções é 44997. Se houver mais parâmetros de funções, passando deste limite, e o usuário necessite supervisionar pontos via Modbus que não apareçam na tabela gerada automaticamente, deve-se colocar o endereçamento Modbus em modo manual e adicionar o ponto no lugar de algum outro ponto não utilizado.

Na opção **Properties Editor** do menu **Tools**, mostrada na figura abaixo, o usuário poderá alterar o tag das entradas, saídas, das variáveis virtuais, dos blocos funcionais, entradas e saídas do FFB, NetIO e metaparâmetros assim como alterar suas respectivas descrições. Os valores de Safe Output das variáveis reais poderão ser alterados e os blocos funcionais poderão ser configurados como se o usuário estivesse na **Área de Trabalho**, na janela **Object Properties**.

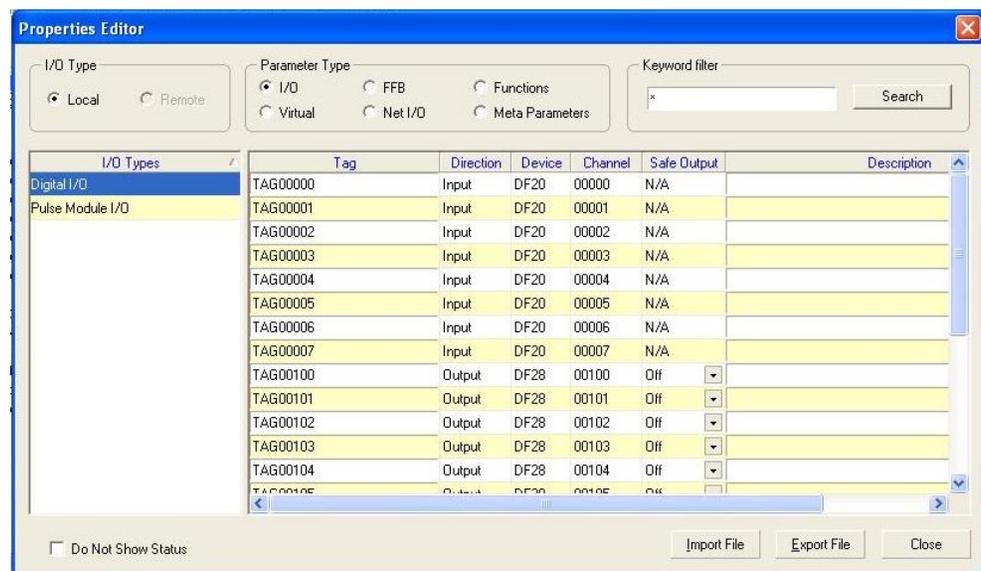


Fig 3. 101 - Alterando os Tags

Basta escolher no filtro **Parameter Type** o tipo de tags a serem visualizados e a lista será atualizada para refletir a escolha do filtro.

Ao clicar duas vezes sobre o parâmetro desejado habilita-se o modo de edição e assim o tag pode ser modificado. O mesmo procedimento aplica-se clicando sobre as descrições.

As alterações efetuadas neste editor são refletidas a todos os objetos da lógica que as utilizam, em todos os diagramas do projeto, independente de seu modo de execução (mesmo os diagramas desabilitados para execução serão atualizados).

Os tags só podem ter caracteres alfanuméricos e "underscore". Os tags também não podem conter espaços. Os caracteres inválidos automaticamente não serão permitidos nos tags.

NOTA

Os tags das variáveis virtuais e das entradas e saídas podem ter no máximo 16 caracteres.

Além dos tags as descrições dos elementos ficarão visíveis na área de desenho.

Outra importante característica presente em **Properties Editor** são as opções de **Import File** e

Export File. Os tags e descrições dos parâmetros tipo I/O e Virtual podem ser exportados para um arquivo txt que depois podem ser manipulados no Microsoft Excel. Este arquivo pode ser importado pelo **LogicView for FFB** com a opção **Import File**. O arquivo importado/exportado é do tipo txt.

Se nenhuma linha for marcada serão exportados todos os tags e suas descrições. Caso uma linha seja selecionada os dados serão exportados a partir da linha selecionada. Clicando em **Export File**, uma janela abrirá solicitando o nome e o local onde será salvo o arquivo. Clique **Save**.

Para fazer as alterações necessárias abra o arquivo exportado no Excel. A seguinte janela abrirá:

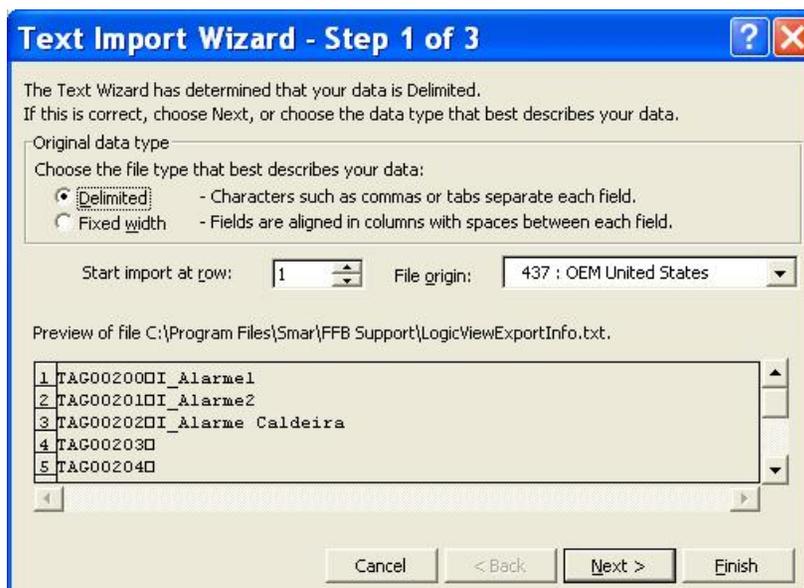


Fig 3. 102 – Abrindo arquivo txt no Excel (1)

Clique **Next**. A seguinte janela aparecerá:



Fig 3. 103 – Abrindo arquivo txt no Excel (2)

Selecione **Tab** e clique **Next**. A seguinte janela abrirá:



Fig 3. 104 – Abrindo arquivo txt no Excel (3)

Selecione **General**, clique **Finish** e faça as alterações necessárias nos tags e descrições. Ao salvar o arquivo a seguinte mensagem aparecerá.



Fig 3. 105 – Salvando o arquivo txt no Excel

Clique **Yes**. Para importar o arquivo basta clicar em **Import File** e escolher o arquivo txt. Os dados serão importados automaticamente. Se nenhuma linha for marcada serão importados os tags e descrições para os parâmetros a partir da primeira linha. Caso uma linha seja selecionada os parâmetros serão substituídos a partir da linha selecionada. O usuário será alertado sobre operações que possam causar problemas. Veja o exemplo a seguir. O alerta é mostrado, pois a quantidade de linhas importadas é maior que as linhas disponíveis em **Properties Editor**.

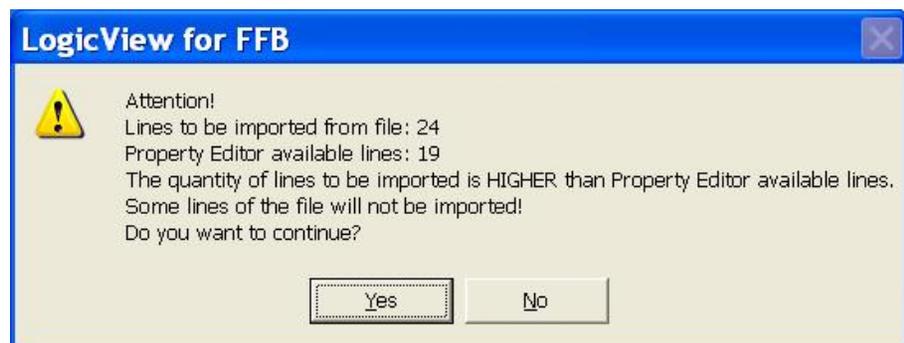


Fig 3. 106 – Importando o arquivo txt

Biblioteca de Lógicas

A Biblioteca de Lógicas é uma forma eficiente de reaproveitamento de lógicas e criação de “modelos” para tratamento de processos bem conhecidos na área de automação discreta.

Para entender a biblioteca de lógicas do **LogicView for FFB** é necessário entender os seguintes conceitos:

- **Library Logic Element:** uma “lógica da biblioteca” corresponde a um diagrama lógico (ladder) do **LogicView for FFB** ou parte dele, que pode ser exportado para reuso posterior. Cada “lógica da biblioteca” recebe um nome e uma descrição fornecidos pelo usuário no momento de sua criação.
- **Library File:** um “arquivo de biblioteca” pode ser entendido como um container ou “banco de lógicas”. Cada arquivo de biblioteca pode conter uma ou mais “lógicas de biblioteca” (ver acima). Os arquivos de biblioteca podem ser criados durante o processo de exportação (descrito mais abaixo) e recebem um nome e uma descrição apropriados, definidos pelo usuário.

Para criar uma lógica de biblioteca deve-se primeiro selecionar a lógica a ser copiada e com botão direito do mouse escolher a opção **Export to Library**. A seguinte janela será aberta.

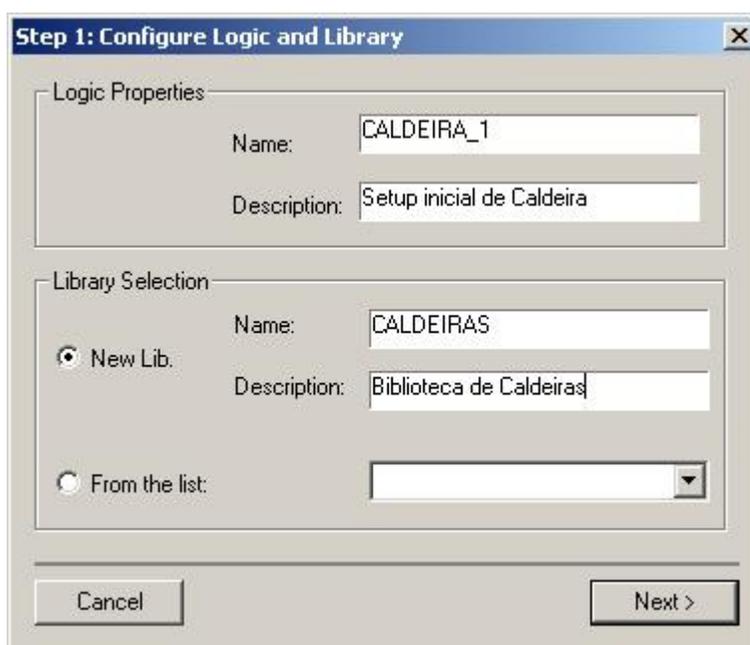


Fig 3. 107 –Opção Export to Logic Library

Na caixa **Logic Properties** o usuário irá definir um nome para a lógica que está exportando, com uma descrição (opcional).

Na caixa **Library Selection** existem duas opções:

- **New Lib.:** nesta opção pode-se criar uma nova biblioteca com uma descrição (opcional). Se já existir uma biblioteca, a seguinte mensagem será exibida:

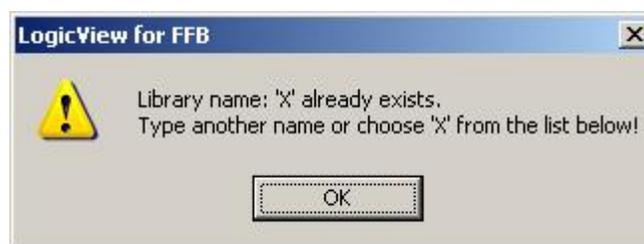


Fig 3. 108 –Mensagem de erro

Neste caso, escolha a opção **From the list**.

- **From the List:** nesta opção você irá selecionar uma biblioteca disponível na lista de bibliotecas de lógica.

É possível exportar mais de uma lógica para a mesma biblioteca, mas apenas uma de cada vez. Caso já exista na biblioteca selecionada uma lógica com o mesmo nome daquela que está sendo exportada, a mensagem abaixo será exibida:

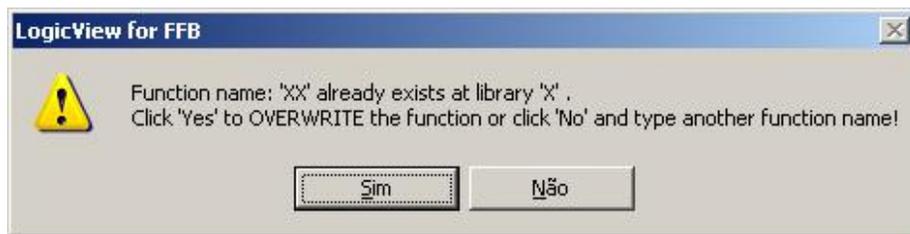


Fig 3. 109 –Mensagem de aviso

O usuário tem a opção de substituir a lógica mais antiga por aquela que está sendo exportada clicando em **Sim**. Caso contrário, clique em **Não** e modifique as informações na janela da figura 3.107. Clique **Next** e a lógica será exportada.

NOTA

Lógicas selecionadas para exportação não podem conter metaparâmetros. Caso uma lógica selecionada contenha pelo menos um metaparâmetro, uma mensagem de erro semelhante a da figura abaixo será exibida:

Após definir os dados da janela da figura 3.107, a seguinte janela será exibida.

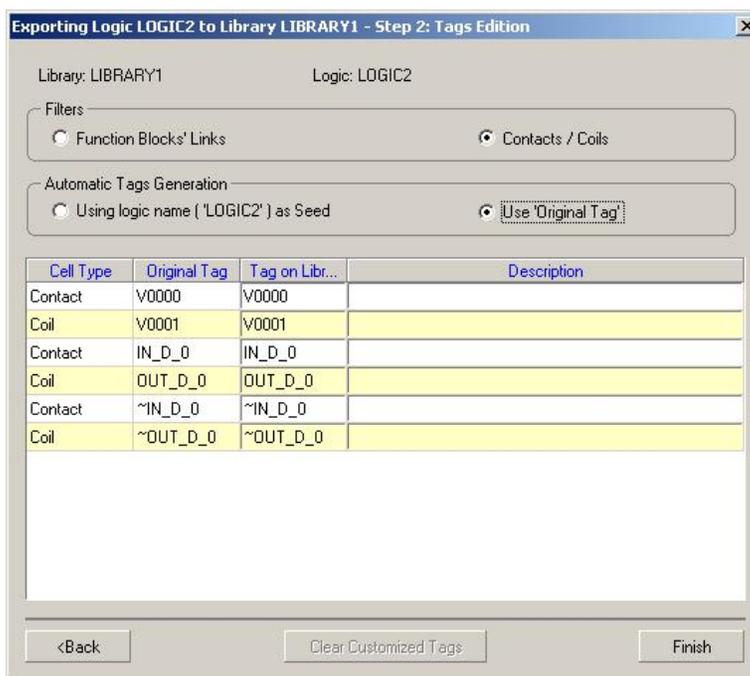


Fig 3. 110 –Edição de Tags durante exportação para Biblioteca de Lógicas (1)

Nesta janela os tags podem ser caracterizados, bem como as descrições dos parâmetros que serão exportados. A caracterização, ou personalização, de tags envolve a edição do conteúdo das colunas **Tag on Library** e **Description** da figura anterior.

A caixa **Filters** permite alternar a visualização da tabela entre elementos discretos, contatos e bobinas, e links de blocos funcionais.

Para Contatos e Bobinas: a tabela irá exibir os elementos exportados, o tag que possuem no diagrama (**Original Tag**), o tag do elemento na biblioteca de lógica (**Tag on Library**) e a descrição do elemento (**Description**).

NOTA

Apenas contatos e bobinas que possuem associação com alguma variável (I/O, Virtual, FFB, NetIO) serão exibidas na lista. Contatos e bobinas sem associação (sem tag) não serão exibidos e apenas seu desenho será exportado.

Para Blocos Funcionais: a tabela irá exibir na coluna **F. Parameter** os parâmetros dos blocos funcionais que possuem link com pontos de FFB ou NetIO (valores numéricos ou links entre blocos funcionais não serão exibidos). A coluna **Linked To** exibe o ponto com o qual o parâmetro está ligado e a coluna **Tag On Library** exibe o tag que será exportado para a biblioteca. Veja exemplo na figura seguinte.

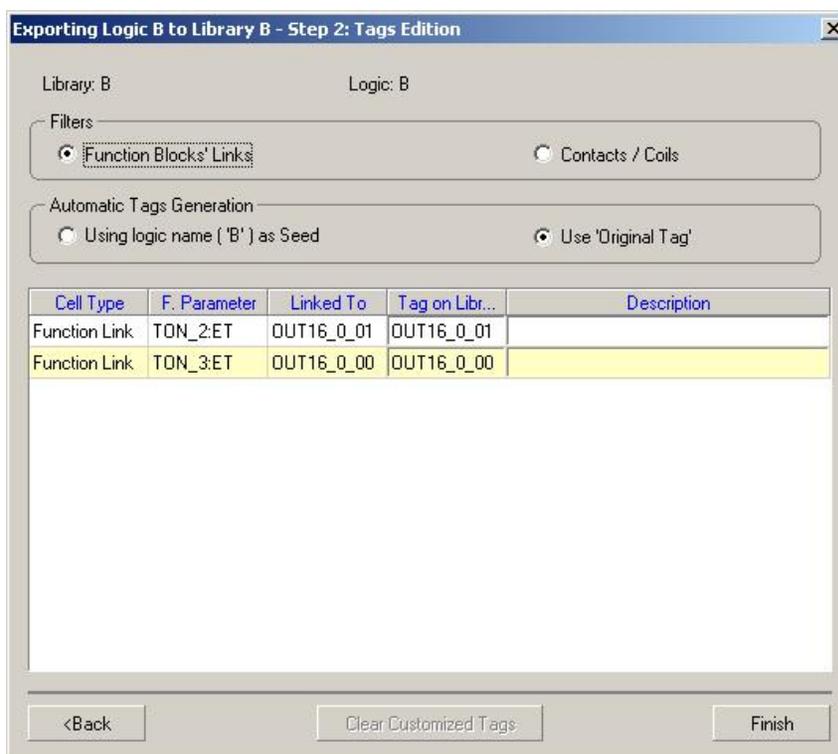


Fig 3. 111 –Edição de Tags durante exportação para Biblioteca de Lógicas (2)

NOTA

O LogicView for FFB irá realizar uma verificação de consistência para os links entre blocos funcionais. Links inconsistentes serão removidos na exportação. Por exemplo: o bloco funcional **TT** possui um link para o bloco **YY**, mas este não foi selecionado para ser exportado. Neste caso o link de **TT** será excluído no processo de exportação.

A caixa **Automatic Tags Generation** oferece duas opções de caracterização automáticas:

- **Use 'Original Tag'** (default): esta opção repete na coluna **Tag On Library** o mesmo tag da coluna **Original Tag** (para contatos e bobinas) ou da coluna **Linked To** (no caso de blocos funcionais).

- **Use Logic name ('xxxx') as Seed:** esta opção utiliza o nome da lógica de biblioteca (definido na janela da figura 3.107) como base para geração automática de todos os tags que serão exportados para a biblioteca, com indexação numérica. Por exemplo se o nome da lógica de biblioteca a ser exportada for **L1** os parâmetros gerados serão **L1_001**, **L1_002**, etc.

Caso queira personalizar os tags, é possível editá-los um a um. Basta clicar no tag da tabela e modificá-lo.

| NOTAS |
|---|
| <ul style="list-style-type: none">• A edição manual de tags tem precedência sobre a geração automática de tags, ou seja, tags que forem modificados manualmente pelo usuário não são mais afetados pelas opções disponíveis na caixa Automatic Tags Generation. Caso queira cancelar todas as edições manuais e voltar a utilizar as opções automáticas, clique no botão Clear Customized Tags.• Alterações em tags de pontos que possuem valor e status serão automaticamente refletidas na sua contraparte, isto é, uma edição no tag de valor será atualizada no respectivo tag de status e vice-versa. O LogicView for FFB mantém a consistência entre os parâmetros exportados.• Todos os tags utilizados na biblioteca são convertidos em metaparâmetros durante o processo de exportação. Para saber mais sobre metaparâmetros ou como utilizá-los, veja "Metaparâmetros" neste manual. |

Quando tiver terminado a edição de tags e descrições, clique em **Finish** e a lógica será exportada para a biblioteca.



Qualquer modificação, como por exemplo *reset* de link ou personalização de tags, será aplicada SOMENTE à lógica de biblioteca que está sendo exportada. O diagrama lógico da configuração não sofre qualquer tipo de modificação durante a exportação para a biblioteca de lógicas.

Gerenciando e Importando Lógicas da Biblioteca

Para importar uma lógica clique em **Tools**→**Logic Library Manager** ou clique com botão direito sobre a área de desenho e selecione **Logic Library Manager**. A seguinte janela abrirá:

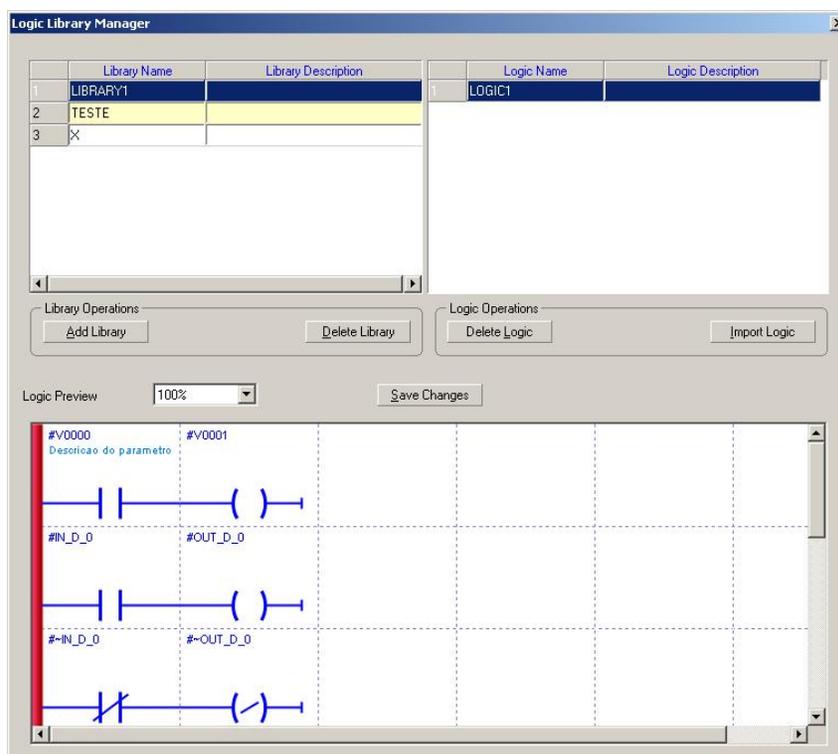


Fig 3. 112 – Importando uma lógica

Esta janela exibe as bibliotecas de lógica disponíveis na sua estação de trabalho. Ao selecionar uma biblioteca, serão exibidas as lógicas que aquela biblioteca contém.

Algumas operações de gerenciamento estão disponíveis:

- **Add Library:** cria uma nova biblioteca de lógicas (vazia).
- **Delete Library:** remove uma biblioteca de lógicas.
- **Delete Logic:** remove a lógica selecionada.

Ao realizar a remoção de uma biblioteca ou de uma lógica, uma mensagem de confirmação semelhante a da figura abaixo, será exibida:



Fig 3. 113 – Aviso de remoção

Também é possível modificar o nome das bibliotecas e das lógicas, bem como suas respectivas descrições.



Após realizar todas as modificações desejadas, é necessário clicar no botão **Save Changes** para efetivar as operações (edição, criação ou remoção de bibliotecas e/ou lógicas). Uma mensagem de confirmação semelhante a da figura abaixo será exibida:

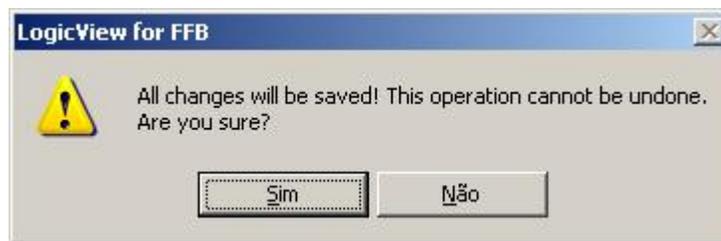


Fig 3. 114 – Confirmação de operação de alterações

Importação de lógica de uma biblioteca para o LogicView for FFB

Escolha a biblioteca desejada e a seguir a lógica. O desenho aparecerá na parte de baixo da janela. Em **Logic Preview** pode-se fazer o zoom na área de visualização. Clique em **Import Logic** e a seguinte janela aparecerá.

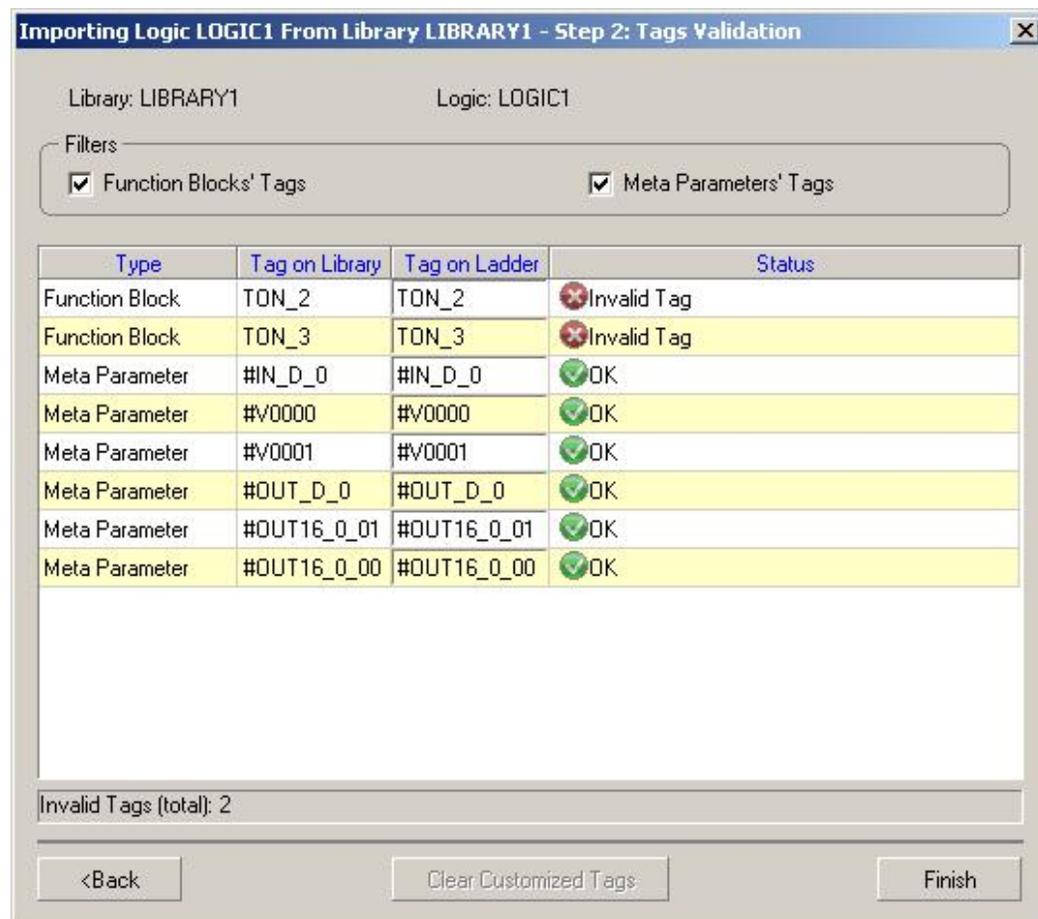


Fig 3. 115 – Importação de lógica da biblioteca

Nesta janela serão exibidos os tags dos metaparâmetros da lógica que será importada da biblioteca para o **LogicView for FFB**. Os itens da tabela são:

- **Type:** esta coluna identifica o parâmetro como tag de bloco funcional ou metaparâmetro;
- **Tag on Library:** esta coluna exibe o tag que foi exportado para a biblioteca;
- **Tag on Ladder:** esta coluna exibe o tag do elemento que será importado para o **LogicView for FFB** e ele pode ser editado e modificado;
- **Status:** esta coluna mostra se o tag definido na coluna **Tag on Ladder** pode ser importado ou não, dependendo de critérios de compatibilidade e integridade de tags da ferramenta.

A caixa **Filters** permite a visualização dos tags de blocos funcionais, metaparâmetros ou ambos.

Durante o processo de importação, o **LogicView for FFB** realiza uma verificação de compatibilidade entre os tags da lógica a ser importada e aqueles já definidos e utilizados na lógica discreta (ladder). Caso haja algum problema de compatibilidade, o tag será sinalizado com **Invalid Tag** na coluna de **Status** e deverá ser modificado.

| NOTAS |
|---|
| <p>Exemplos de incompatibilidade de tags:</p> <ul style="list-style-type: none"> • Blocos Funcionais: se já houver algum bloco funcional na ladder com o mesmo tag de um bloco funcional da lógica que está sendo importada, este tag será sinalizado como Invalid Tag. • Metaparâmetros: se houver algum metaparâmetro na ladder com tag igual a um metaparâmetro definido na lógica que está sendo importada, eles devem ser do mesmo tipo (Digital Input, Digital Output, Analog Input, Analog Output). Se eles forem de tipos diferentes (DI X DO) o tag do metaparâmetro será sinalizado como Invalid Tag. |

Todas as modificações de tags podem ser desfeitas clicando-se no botão **Clear Customized Tags**. Quando tiver terminado de editar os tags, clique em **Finish** para terminar o processo de importação da lógica. A seguinte mensagem será exibida.



Fig 3. 116 – Importação efetuada com sucesso



A importação só poderá ser realizada se todos os tags estiverem com o status **OK**. Caso contrário, a seguinte mensagem de erro será exibida e o processo de importação irá retornar para a tela da figura 3.115.

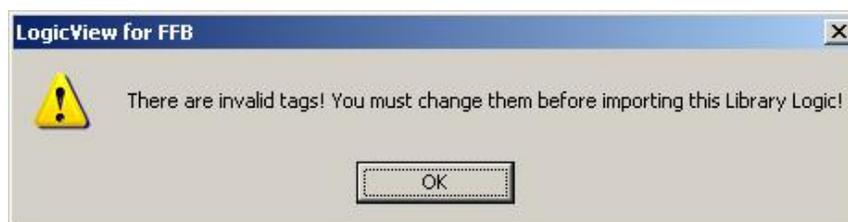


Fig 3. 117 – Erro de Importação

Clique **Ok** e cole a lógica copiada no local desejado.

| NOTA |
|--|
| <p>É possível colar a lógica importada mais de uma vez na configuração ladder. No entanto, a partir do 2º <i>paste</i>, o LogicView for FFB irá verificar a consistência dos dados e aplicar os procedimentos descritos no item Copy/Paste inteligente deste manual.</p> |

Menu Help

Clicando em **Help**, ou através do atalho ALT+ H, o seguinte menu se abrirá:

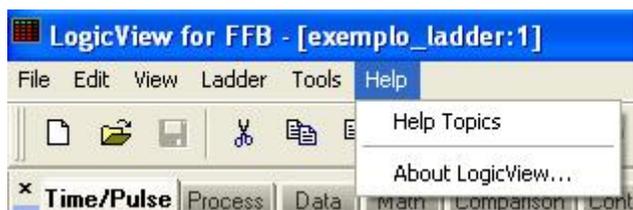


Fig 3. 118 - Menu Help

Neste menu o usuário poderá verificar a versão do software que foi instalada e também seu help.

Barras de Ferramentas

Aqui serão apresentados os detalhes das barras de ferramentas do **LogicView for FFB**. Elas podem ser habilitadas ou desabilitadas conforme foi visto no tópico **Menu View**.

Main Bar



Fig 3. 119 - Barra de Ferramentas Principal

Nesta barra de ferramentas além das opções básicas do Windows (**New, Open, Save, Cut, Copy, Paste, Print, Print Preview e Help**) existem algumas novas: **Copy Drawing, Commit, Export Tags for OPC Browsing, Find and Replace e Revert**.

O comando **Copy Drawing**  funciona como o comando Copy do Windows. O desenho selecionado será copiado, mas o tag associado não será copiado.

O comando **Export Tags for OPC Browsing**  atualiza o arquivo Taginfo.ini com todos os tags da lógica aberta, habilitando-os para busca sem fazer o download da configuração no controlador.

O comando **Commit**  funciona da seguinte forma: ao clicar nele é realizada a operação que envia os arquivos criados ou alterados localmente para o servidor de multiusuário.

Para descartar as alterações feitas no arquivo local e restaurar o arquivo de configuração original

clique no ícone **Revert**, , na barra de ferramentas **Main**. Para maiores detalhes sobre o modo multiusuário referira-se ao manual do Syscon.

Zoom Bar



Fig 3. 120 - Barra de Ferramentas Zoom

O botão  permite que o usuário, ao clicar na área de desenho da ladder, aumente o zoom em 10% a cada clique na janela.

O botão  permite o movimento inverso. A cada clique o zoom diminui em 10%.

O botão  permite que o usuário visualize a página inteira (Full Page).

O botão  permite que o usuário ajuste a visualização de forma que ela se ajusta à página.

NOTA

Para desabilitar as funções de zoom basta clicar em ESC ou no botão de Seleção 

Toolbox

A barra de ferramentas **Toolbox** é formada por oito abas com os diferentes tipos de blocos de funções: IO, Time/Counter, Process, Data, Math, Comparison e Elements. Também tem uma aba **Communication** com funções relacionadas ao modo online, simulação, supervisão dentre outras. Esta aba será explicada separadamente mais adiante.

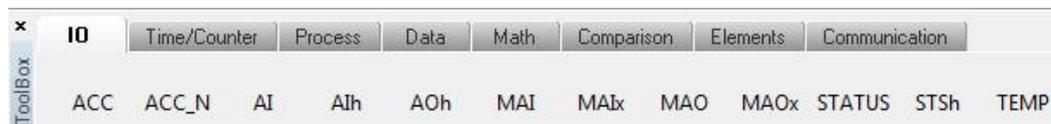


Fig 3. 121 - Toolbox

Cada um desses blocos funcionais, distribuídos em seis abas – **IO**, **Time/Counter**, **Process**, **Data**, **Math** e **Comparison** foram descritos em detalhes no capítulo 2. Neste tópico será tratado apenas a inserção e configuração das funções na rede ladder. A aba **Elements** será descrita separadamente.

Usando as teclas de atalho de F5 a F8 o usuário tem o controle sobre a toolbox. Clicando na tecla **F5 move-se à esquerda** nas abas dos tipos de blocos funcionais (Time/Pulse, Process, Data, Math, Comparison e Elements) da barra de ferramentas. Com a tecla **F6 move-se à direita** nas abas dos blocos funcionais.

Escolhida uma aba, por exemplo Math, clicando na tecla **F7 move-se à esquerda entre os elementos** desta aba. Com a tecla **F8 move-se à direita entre os elementos de uma aba**.

Para inserir um bloco funcional clique no bloco desejado na barra de ferramentas. Mova o mouse na rede. Note que o ícone do mouse muda para . Clique em qualquer célula para inserir o bloco funcional.

Existem restrições para onde os blocos podem ser inseridos. Isto está relacionado com o tamanho do bloco e elementos na vizinhança, assim, às vezes o usuário deverá selecionar outro local para inserir o bloco funcional. A seguinte mensagem aparecerá:



Fig 3. 122 - Alerta sobre inserção de elemento na célula

Após o bloco ser inserido, ele deverá ser configurado na janela **Object Properties**. Os itens que aparecem em cinza claro não podem ser alterados pelo usuário.



Fig 3. 123 - Janela Object Properties

Neste caso podem ser alterados os itens **TAG**, **PT Link Type**, **PT Value**, **ET Link Type** e **ET Value**.

Para alterar o TAG do bloco de função clique duas vezes na célula à direita de **TAG**. O modo de edição será habilitado e o usuário poderá escrever o TAG desejado.

Em **PT Link Type** as opções são:

- **Value** – Valor numérico que deve ser inserido pelo usuário e que será feito download.
- **Address** – Indica que a entrada do bloco funcional está ligada a uma saída de algum bloco.
- **FF Address** – Indica que a entrada do bloco funcional está ligada a uma saída analógica do FFB.
- **NetIO Address** - Indica que a entrada do bloco funcional está ligada a uma saída analógica do NetIO.
- **Meta Parameter** – Indica que a entrada do bloco funcional está ligada a um metaparâmetro do tipo saída analógica.

Em **PT Value** as opções disponíveis dependerão do que foi escolhido em **PT Link Type**. Se for escolhido **Value** o usuário deverá entrar com um valor numérico. Caso o valor inserido esteja fora do intervalo permitido a seguinte mensagem aparecerá:



Fig 3. 124 - Erro – Valor inválido

Se em **PT Link Type** for escolhido **Address** as opções disponíveis em **PT Link** serão **Not Connected** ou as saídas dos blocos funcionais.

Se em **PT Link Type** for escolhido **FF Address** as opções disponíveis em **PT Link** serão **Not Connected** ou as saídas analógicas do FFB.

Se em **PT Link Type** for escolhido **NetIO Address** as opções disponíveis em **PT Link** serão **Not Connected** ou as saídas analógicas do NetIO.

Se em **PT Link Type** for escolhido **Meta Parameter** as opções disponíveis em **PT Link** serão **Not Connected** ou as saídas analógicas de metaparâmetros.

Em **ET Link type** estão as opções de link para a saída do bloco funcional – **FF Address**, **NetIO Address** e **Meta Parameter**. Dependendo do tipo escolhido a saída do bloco estará ligada a uma entrada analógica do FFB, de NetIO ou de metaparâmetro.

Os itens a serem configurados vão variar de acordo com o bloco de função escolhido. Se o usuário inserir, por exemplo, o bloco **SEL2** os itens a serem configurados, na **Object Properties**, serão:

| Property Name | Property Value |
|---------------|----------------|
| Name | SEL2 |
| Tag | SEL2_1 |
| Description | |
| P1 Link Type | Value |
| P1 Value | 0.0000 |
| P2 Link Type | Value |
| P2 Value | 0.0000 |
| OUT Link Type | FF Address |
| OUT Value | Not connected |
| OUT Alias | |

Fig 3. 125 - Janela Object Properties- Bloco de Função SEL2

Neste caso podem ser alterados os itens **TAG**, **P1 Link Type**, **P1 Value**, **P2 Link Type**, **P2 Value**, **OUT Link Type** e **OUT Value**.

Para alterar o TAG do bloco de função clique duas vezes na célula à direita de **TAG**. O modo de edição será habilitado e o usuário poderá escrever o TAG desejado.

Em **P1** e **P2 Link Type** as opções são:

- **Value** – Valor numérico que deve ser inserido pelo usuário e que será feito download.
- **Address** – Indica que a entrada do bloco funcional está ligada a uma saída de algum bloco.
- **FF Address** – Indica que a entrada do bloco funcional está ligada a uma saída analógica do FFB.
- **NetIO Address** - Indica que a entrada do bloco funcional está ligada a uma saída analógica do NetIO.
- **Meta Parameter** – Indica que a entrada do bloco funcional está ligada a um metaparâmetro do tipo saída analógica.

Em **P1** e **P2 Value** as opções disponíveis dependerão do que foi escolhido em **P1** e **P2 Link Type**, respectivamente. Se for escolhido **Value** o usuário deverá entrar com um valor numérico.

Se em **P1** e **P2 Link Type** for escolhido **Address** as opções disponíveis em **P1** e **P2 Value** serão **Not Connected** ou as saídas dos blocos funcionais.

Se em **P1** e **P2 Link Type** for escolhido **FF Address** as opções disponíveis em **P1** e **P2 Value** serão **Not Connected** ou as saídas analógicas do FFB.

Se em **P1** e **P2 Link Type** for escolhido **NetIO Address** as opções disponíveis em **P1** e **P2 Value** serão **Not Connected** ou as saídas analógicas do NetIO.

Se em **P1** e **P2 Link Type** for escolhido **Meta Parameter** as opções disponíveis em **P1** e **P2 Value** serão **Not Connected** ou as saídas analógicas de metaparâmetro.

OUT Link Type poderá ser um **FF Address**, **NetIO Address** ou **Meta Parameter** e, neste caso, **OUT Value** poderá ser **Not Connected** ou os links disponíveis no FFB (FB Address), no NetIO ou nos metaparâmetros.

Se o usuário fizer uma configuração no **Syscon**, com um FFB que possui DI, DO, AI e AO, e se no **LogicView for FFB** colocar uma função, por exemplo um timer, ao definir o tipo dos links de PT e ET como FF Address, verá que os AOs e AIs, que foram criados no FFB estarão disponíveis para link.

Links entre entradas e saídas dos blocos funcionais

É possível fazer links entre entradas e saídas. Selecione o bloco funcional e na janela **Object Properties** configure o tipo de link necessário para a entrada – **Address**, **FF Address**, **NetIO Address** ou **Meta Parameter**.

Escolha a saída com a qual a entrada estará linkada. Nesta situação, clicando com o botão direito do mouse aparece um menu. O penúltimo item deste menu é o **Output Link**, que mostra todas as entradas linkadas naquela saída, conforme o exemplo mostrado na figura a seguir.

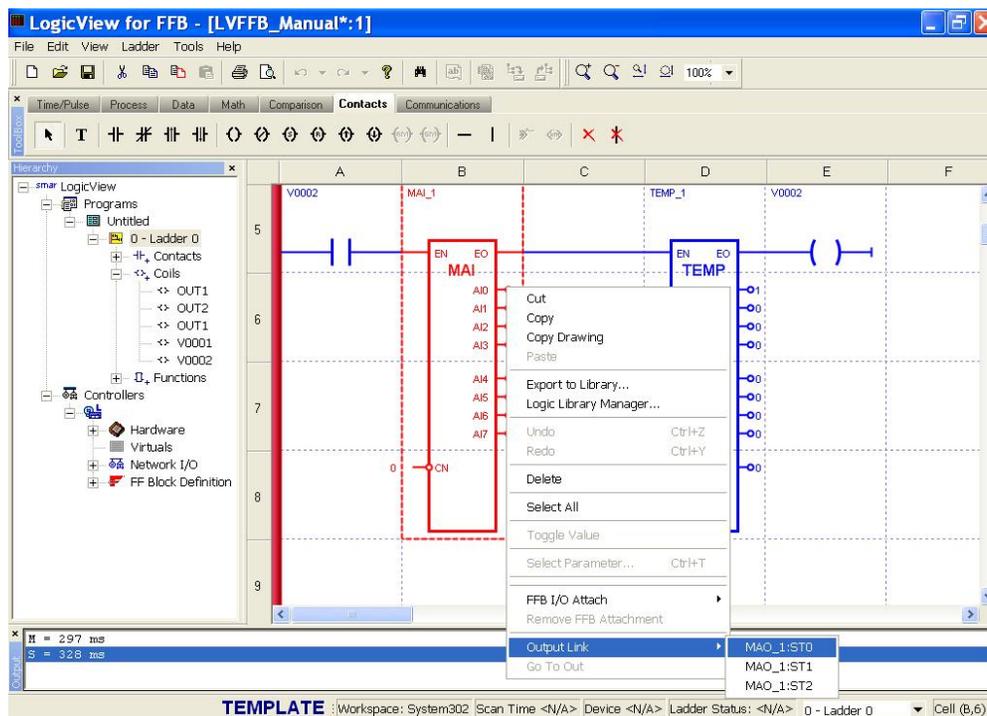


Fig 3. 126 – Opção Output Link

Um modo mais prático para fazer o link entre uma saída de uma função com uma entrada é posicionar o mouse sobre a saída que se deseja fazer o link (neste momento o cursor do mouse vira uma "mãozinha" indicando que está sobre um parâmetro analógico de saída de uma função) e pressionar a tecla **SHIFT**. Desta forma, o parâmetro de saída é armazenado. Ao colocar o mouse sobre a entrada na qual se deseja fazer o link (o cursor novamente tem a forma de uma "mãozinha") e pressionando a tecla **SHIFT**, o link é automaticamente realizado.

IMPORTANTE

Para realizar esta operação, o foco tem que estar na área de desenho da ladder

Se existir link em uma entrada e clicando com o botão direito sobre a mesma a seguinte janela aparecerá:

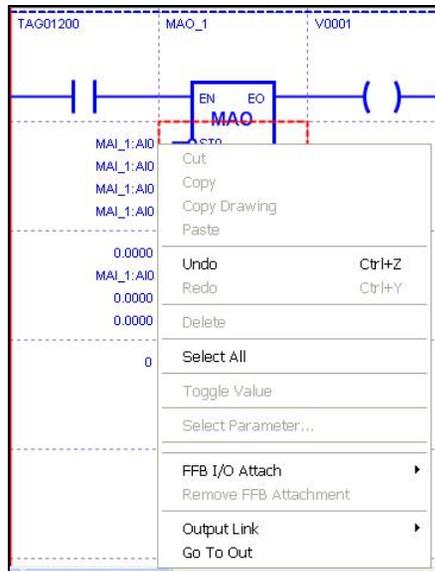


Fig 3. 127 – Opção Go to Out

A última opção - comando **Go To Out**, leva à função que está linkada nesta entrada. Se existirem links na saída, ao se clicar em alguma das entradas “linkadas” dentro de **Output Link**, também se é levado à função que possui o link.

Com o **LogicView for FFB** fora do modo de supervisão mostra-se o número de entradas que estão “linkadas” a uma saída, como na figura abaixo. Para saber quais são estas entradas, o procedimento é o mesmo descrito anteriormente.

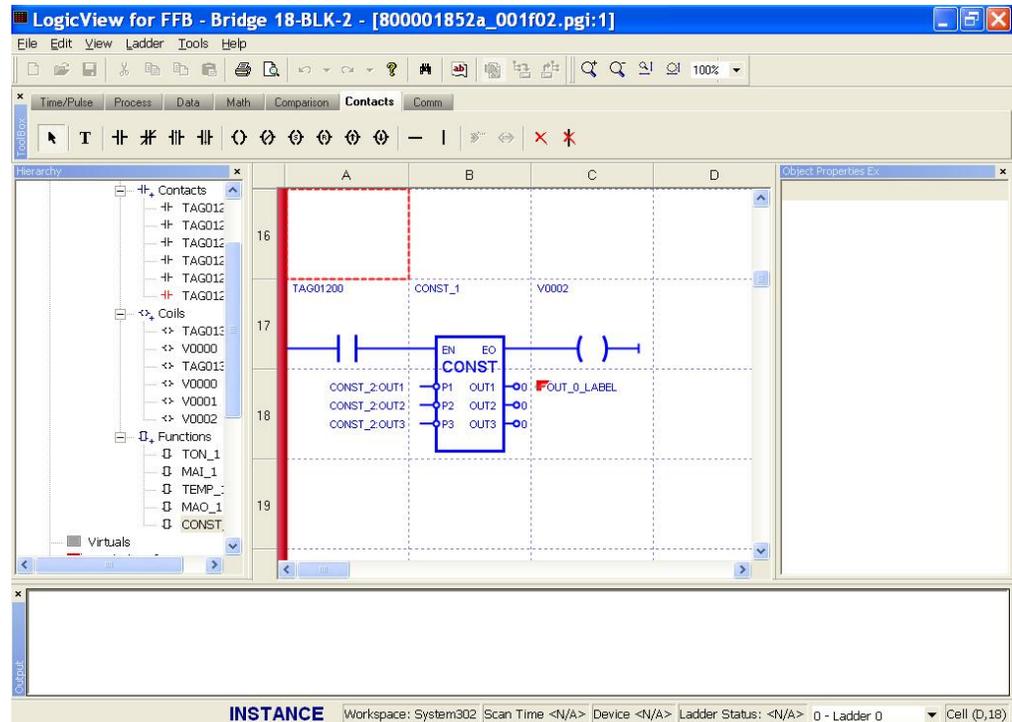


Fig 3. 128 – Entradas “linkadas” às saídas

NOTA

Em todas as saídas de funções, o usuário pode usar um *Alias* (tag do usuário) para elas, ou seja, se for definido um *Alias*, será este o tag que será supervisionado e que poderá ser “linkado” com entradas de funções. Na ausência do *Alias*, ficará valendo o tag default da saída - Nome da função:Nome da saída.

Link de pontos analógicos do FFB com os Blocos Funcionais

Existe outro modo de ligar pontos analógicos do FFB com as funções. Numa função, ao se passar o cursor sobre uma entrada/saída analógica, o cursor fica com a forma de uma mãozinha, indicando que é uma entrada/saída analógica.

Com o cursor em forma de seta, clicando no botão direito do mouse no elemento, aparece habilitada a opção **FFB I/O Attach**, indicando quais pontos analógicos do FFB estão disponíveis para link. Veja figura seguinte.

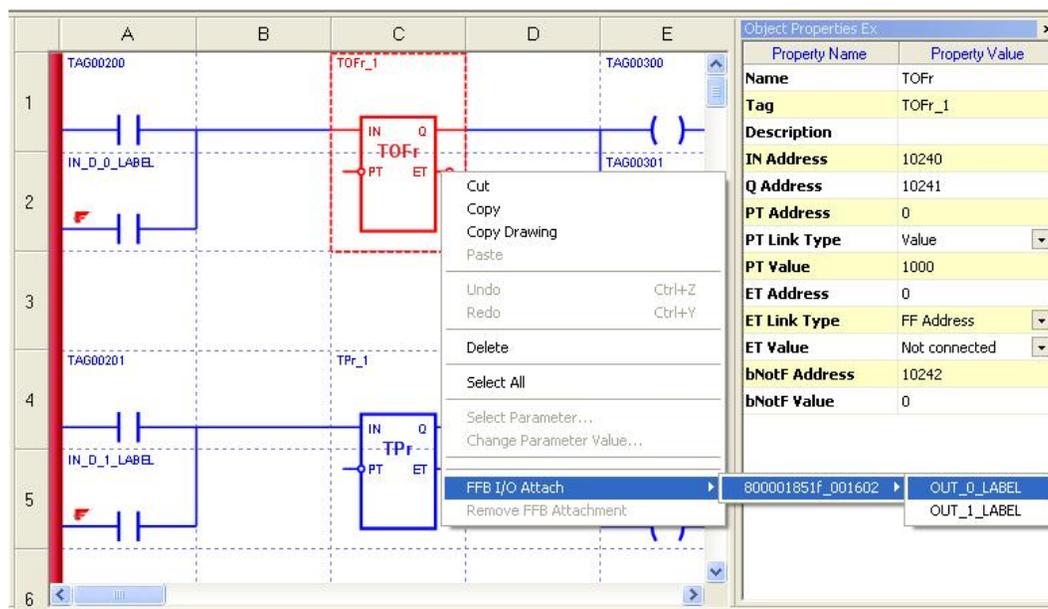


Fig 3. 129 - Comando FFB I/O Attach

Dada uma entrada de um bloco funcional, por exemplo PT, estarão disponíveis para link, os pontos IN_x do FFB, ou seja, pontos que vêm do FFB, e em uma saída de um bloco funcional, por exemplo ET, estarão disponíveis para link, os pontos OUT_x do FFB, ou seja, pontos que vão para o FFB.

Os pontos IN_x podem ser ligados a diversas entradas de blocos funcionais. Já os pontos OUT_x só podem receber um único link vindo de um bloco de função, por isso, à medida que OUT_x são ligados, o número de pontos disponíveis para link de OUT_x são diminuídos.

E se um ponto de um bloco funcional está ligado a um ponto analógico do FFB, no botão direito do mouse existe a opção **Remove FFB Attachment** habilitada que remove o link do bloco.

Para adicionar um elemento ladder na rede, selecione um elemento (contatos, bobinas, linhas verticais ou horizontais) na aba **Elements** do **LogicView for FFB**. Clique no elemento a ser adicionado e posicione-o na célula desejada dentro da área de desenho da Ladder. O **LogicView for FFB** automaticamente insere este elemento.

O **LogicView for FFB** tem uma característica “check-as-you-go” que previne o usuário de inserir elementos que não sejam aplicáveis a uma célula específica. Neste caso, a seguinte mensagem aparecerá.



Fig 3. 133 - Alerta sobre inserção de elemento na célula

O elemento escolhido pode ser inserido quantas vezes forem necessárias sem que se tenha que clicar novamente na aba **Elements**. Para cancelar o comando clique em ESC, no botão de seleção ou em outro elemento, caso deseje inseri-lo.

Assim que um elemento lógico tenha sido colocado ele poderá ser referido por seu tag padrão ou pelo tag do usuário.

NOTA

Após inserir elementos tipo contato ou bobina e associar um tag a ele, o elemento pode ser substituído por outro do mesmo tipo sem que seja preciso apagar primeiro o elemento a ser substituído. A troca é imediata, basta escolher o novo elemento na aba **Elements** e posicioná-lo na célula desejada. O tag não será alterado.

Após inserir os elementos é preciso configurá-los. Isto pode ser feito das seguintes maneiras:

- Clicando duas vezes sobre o elemento

A seguinte janela abrirá:

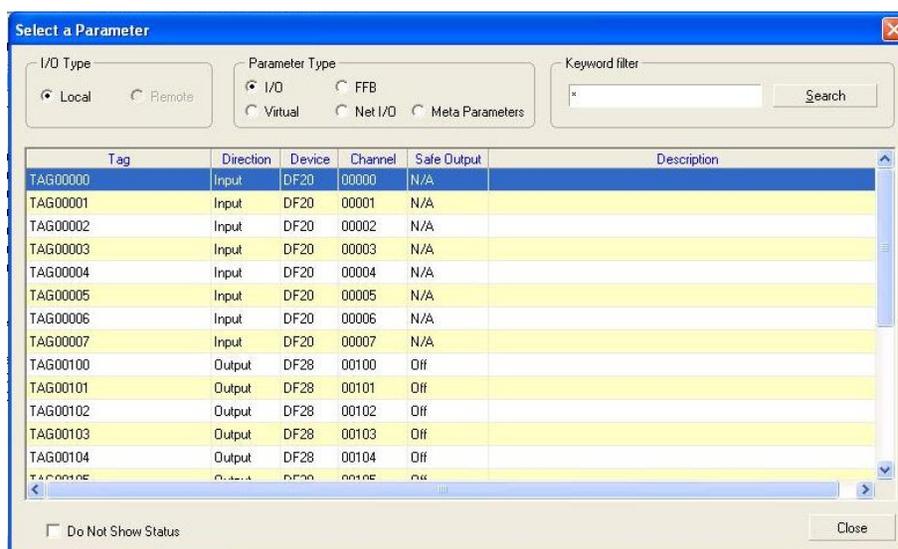


Fig 3. 134 - Selecionando o parâmetro

Nesta janela o usuário deverá configurar o tipo do parâmetro – I/O, Virtual, FFB, Meta Parameter ou Net I/O. No caso de pontos discretos do FFB, o valor do status está representado pelo mesmo tag do valor do ponto, acrescentado de um til (~) na frente do tag.

IMPORTANTE

Dentro da lógica qualquer valor de status, será sempre 1 (verdadeiro) se o valor do status for **Bad** ou **Uncertain** e 0 (falso) se o valor for **Good**.

Quando o tipo é selecionado aparece uma lista dos itens possíveis, seus tags padronizados, local onde estão instalados e os valores de Safe Output. O usuário deverá escolher aquele que lhe convém. Não é possível editar o tag e os valores de Safe Output nesta janela.

Se após configurar o parâmetro, o usuário pressionar a tecla **ALT** mais o botão esquerdo do mouse, será ativada a função carimbo, na qual o cursor do mouse assume a forma de um carimbo. Com esta função é possível replicar o mesmo tag em outros elementos, basta manter a tecla **ALT** pressionada e clicar no botão esquerdo do mouse.

A função carimbo para contatos e bobinas também é ativada com um clique simples sobre o elemento. Desta forma o elemento é armazenado. Pressionando a tecla **ALT** mais simples clique sobre qualquer contato ou bobina, o elemento é replicado.

A função carimbo também pode ser usada da mesma forma para funções. Os parâmetros internos e os de entrada, sem links, são armazenados e podem ser replicados em outra função de mesmo tipo da original.

IMPORTANTE

Para o carimbo poder atuar, o foco tem que estar na área de desenho da ladder

- Clicando com o botão direito sobre o elemento
A seguinte janela se abrirá:

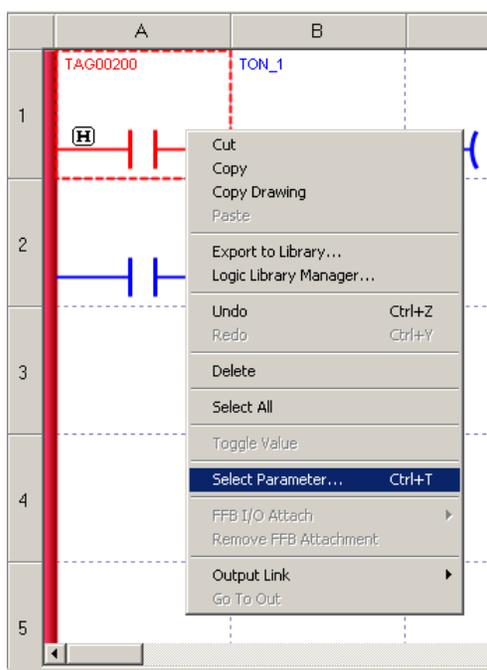


Fig 3. 135 - Selecionando o parâmetro

Ao selecionar **Select Parameter** uma janela igual à da figura 3.134 será aberta e o procedimento é o mesmo do anterior.

Inserindo tags nos elementos

Ao inserir contatos ou bobinas pode-se dar um tag a ele via grid. Ao clicar sobre um contato ou bobina sem tag a seguinte opção aparece na janela **Object Properties**.



Fig 3. 136 – Inserindo novo tag

Escreva o tag do parâmetro. Ao confirmar a operação, a seguinte tela será exibida:



Fig 3. 137 – Janela Tag Options

A janela da figura acima irá mostrar as opções disponíveis ao usuário que estão relacionadas ao tag editado. As opções para edição de tags são as seguintes:

- **Rename Tag: XXXXX to YYYY:** quando disponível, esta opção irá trocar o tag anterior pelo tag digitado pelo usuário, atualizando automaticamente todos os elementos que tenham o tag anterior para o novo tag;
- **Assign variable: YYYY to the selected Contact/Coil:** se o **LogicView for FFB** encontrar um parâmetro com o mesmo tag digitado pelo usuário, que seja compatível com o elemento discreto (contato/bobina) editado, esta opção irá associar o parâmetro encontrado ao contato ou bobina da ladder;
- **Create a new Virtual variable with Tag YYYY:** esta opção irá criar um parâmetro do tipo Virtual e a seguir irá associá-lo ao elemento discreto da ladder (contato/bobina) em edição. Este novo parâmetro virtual será inserido na lista de tags virtuais e pode ser visto em **Tools** → **Properties Editor** → **Virtual**;
- **Create a new Meta Parameter with Tag YYYY** esta opção irá criar um parâmetro do tipo metaparameter e a seguir irá associá-lo ao elemento discreto da ladder em edição.

Ao exibir a janela da figura anterior, o **LogicView for FFB** irá desabilitar automaticamente as opções que não forem compatíveis e/ou não estiverem disponíveis no momento. Por exemplo, as opções 1 e 2 são mutuamente exclusivas, ou seja, quando existir um parâmetro com o tag digitado pelo usuário não é possível apenas renomear o tag; só é possível atribuir contato ou bobina a variável existente.

Para selecionar uma opção, basta clicar com o mouse no item desejado ou digitar o número correspondente a opção (de 1 a 4). Pressionar a tecla ESC tem o mesmo efeito da opção 4.

Caso o tag digitado pelo usuário pertença a um bloco funcional, o **LogicView for FFB** exibirá a seguinte mensagem de erro:



Fig 3. 138 – Erro – Operação Indisponível

Esta mensagem indica que não é possível realizar nenhuma operação com o tag digitado porque

ele pertence a um bloco funcional utilizado no(s) diagrama(s). Ao clicar **OK**, a edição do tag é cancelada e o valor anterior é restaurado.

Caso o tag digitado pelo usuário pertença a um parâmetro não compatível com o contato/bobina, (por exemplo, o elemento editado é uma bobina e o tag digitado pertence a um ponto real de entrada), a seguinte mensagem de erro será exibida:



Fig 3. 139 – Erro – Operação Indisponível

Esta mensagem indica que não é possível realizar nenhuma operação com o tag digitado porque ele pertence a um parâmetro já criado no **LogicView for FFB** e que não é compatível com o contato ou bobina que está sendo editado. Ao clicar **OK**, a edição do tag é cancelada e o valor anterior é restaurado.

Aba Communication

Se o usuário estiver **Offline** a aba **Communication** aparecerá de seguinte forma:



Fig 3. 140 - Aba Comm Offline

Build

O botão **Build**  ativa o comando para geração do pseudocódigo que será executado pela máquina virtual 1131, conforme explicado anteriormente. Se existe algum erro na lógica ladder, como por exemplo falta de conexão no elemento, a seguinte mensagem aparecerá.



Fig 3. 141 - Erro ao executar o comando Build

Se aparecer a mensagem acima, na **Output bar** aparecerá a especificação do erro e ao se clicar nele, o usuário irá direto para o ponto de erro na área de desenho da ladder.

Simulation

O botão **Simulation**  foi explicado no tópico **Menu ladder**.

Online

Ao clicar no botão **Online**  a seguinte janela se abrirá:

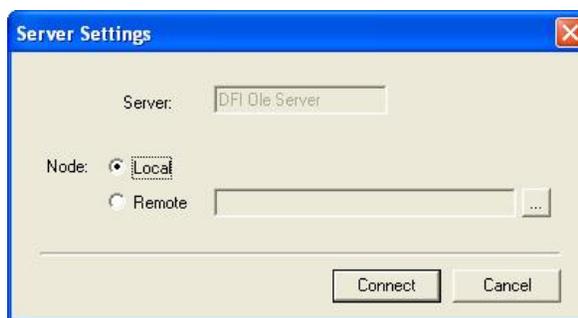


Fig 3. 142 - Janela de configuração do servidor

O usuário deverá escolher entre **Local** e **Remote** e clicar no botão **Connect**. Se o device desejado não for encontrado ao se conectar tanto o **Scan time** e o **Ladder Status**, na Barra de Status, ficarão **N/A** (*Not available*).

Se o usuário estiver no modo **Online** a aba **Comm** aparecerá da seguinte forma:



Fig 3. 143 - Aba Comm Online

O tipo de controlador (com o qual o **LogicView for FFB** deve se conectar) e seu número de série são obtidos do *Database*. Esta informação é salva no *Database* pelo comando **Save do Syscon** após o comissionamento. Caso isto não tenha sido feito a seguinte mensagem aparecerá:



ATENÇÃO

Se o **LogicView for FFB** reportar mensagem de erro, dizendo que há falha na conexão com o controlador, siga os passos seguintes para analisar esta condição:

- 1) Use o **FBTools** para tentar uma conexão com o controlador.
- 2) No *prompt* do DOS:
 - Fazer **ping** com o controlador.
 - Fazer **netstat -n**. Como resposta receber *TCP ip_pc:4988 ip_df:porta_aleatória ESTABLISHED*

Provavelmente uma destas duas condições falhará. Neste caso, as possíveis causas da falha são:

- 1) Configuração do IP no **Server Manager** errada (Verifique em Settings→Network→General).
- 2) Firewall ou antivírus estão bloqueando a conexão do controlador com o computador. Desativando-os, a conexão se estabelece.
- 3) Diferença na configuração da máscara de sub-rede das placas de rede. Normalmente ela é configurada como 255.255.255.0. Tanto no computador como no controlador, a máscara deve ter a mesma configuração para que a conexão se estabeleça. Para verificar esta configuração no computador, no *prompt* do DOS, faça **ipconfig**. No controlador, use o **FBTools** ou o **webserver**. Para maiores detalhes veja o manual do **DFI302**.

Fazendo o download da configuração

O botão **Download Configuration**  permite que o usuário faça o download da configuração de uma instância, a configuração 1131.



ATENÇÃO

O primeiro download de uma configuração sempre deve ser feito pelo **Syscon**.

Se o usuário tentar fazer o download da configuração sem antes ter feito o comissionamento dos devices ou não salvou a configuração a seguinte mensagem aparecerá:

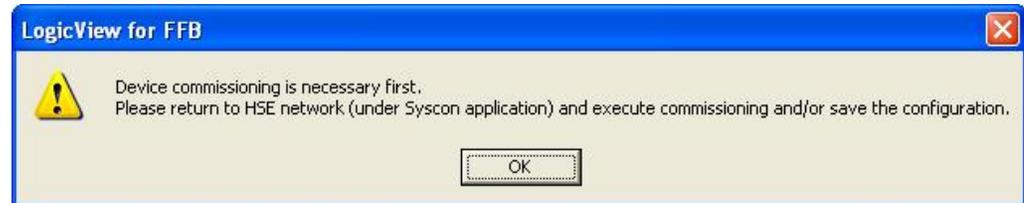


Fig 3. 144 – Erro ao fazer o download (1)

O usuário deverá retornar ao **Syscon** para comissionar os devices e salvar a configuração.

Se o primeiro download não foi feito via **Syscon** ou se o usuário mudar o tag do FFB ou mudar a revisão, fazendo um novo **Define parameters**, o **LogicView for FFB** vai retornar a seguinte mensagem quando lhe for solicitado um download.



Fig 3. 145 – Erro ao fazer o download (2)

O usuário deverá retornar ao **Syscon** para realizar um download do equipamento.

Após ter feito o primeiro download via **Syscon**, pode-se fazer outros via **LogicView for FFB**. No entanto, o usuário deverá escolher se deseja ou não manter a CPU em modo de execução ou se quer cancelar o processo de download.

Ao clicar em  a seguinte janela se abrirá:



Fig 3. 146 – Mantendo a execução da CPU



ATENÇÃO

Ao executar o comando **Stop**:

- As saídas discretas vão para *Safe Mode*.
- As saídas analógicas do bloco funcional MAO assumirão os valores definidos em ST0, ST1, ST2 e ST3.

Durante o download via **LogicView for FFB** é mostrada a animação representada na figura seguinte.

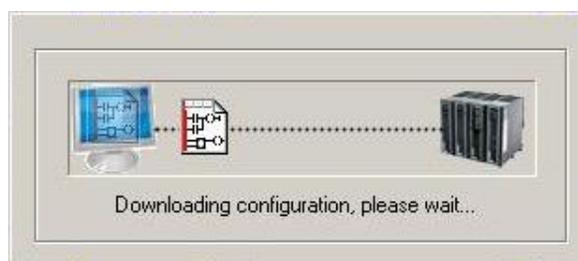


Fig 3. 147 – Download em execução

Outras mensagens de erro relacionadas ao download da configuração

Download da lógica via Syscon



Fig 3. 148 – Erro ao fazer o download via Syscon

Possível causa: Falha na comunicação Ethernet entre o equipamento e o OPC Server ou problema com o firmware.

Solução: Verificar a comunicação Ethernet. Falha persistindo, entrar em contato com suporte técnico da Smar.

Download da lógica via LogicView for FFB



Fig 3. 149 – Erro ao fazer o download via LogicView for FFB

Possível causa: Falha na comunicação Ethernet entre o equipamento e o OPC Server ou problema com o firmware.

Solução: Verificar a comunicação Ethernet. Falha persistindo, entrar em contato com suporte técnico da Smar.

Download da lógica via LogicView for FFB ou via Syscon

Todos os casos abaixo se referem a problemas ao tentar fazer o download da lógica, seja via **Syscon** ou via **LogicView for FFB**. As mensagens aparecerão em janelas.

1) **1131 Build Error:** Erro na lógica. Entrar no **LogicView for FFB** (Edit Logic) e realizar o build na lógica, para assim poder verificar onde está o erro. Na janela **Output** aparecerão os erros, clicando sobre eles o **LogicView for FFB** indicará o erro na **Área de Desenho da Ladder**.

2) **Data is too large to be compiled, please refer to LogicView User Guide to check about limitations Code: Tags: Funct:** Será mostrado o número de bytes gerados pelo código da lógica, o número de tags usados e o número de funções. Veja abaixo os limites para os controladores Smar (para maiores detalhes refira-se ao manual do DFI302)

DF75: 120000 bytes, 2000 funções;
DF73, DF79, DF81, DF95, DF97: 120000 bytes, 1200 funções;
DF62 e DF63: 20000 bytes, 300 funções;
HFC302: 5000 bytes, 300 funções;
DF89: 60000 bytes, 1200 funções.

O build da lógica no **LogicView for FFB** gera estas informações para o usuário.

3) Todos os erros abaixo indicam falha em um dos comandos de download. Repita a operação. Persistindo o erro, entrar em contato com o suporte técnico da Smar.

- 1131 Code Download Error
- 1131 Disc. Cfg Download Error
- 1131 Ana Cfg Download Error
- 1131 Disc. Points Download Error
- 1131 Real Points Download Error
- 1131 Disc. Safe Download Error
- 1131 Pulse Download Error
- 1131 Extra Long Download Error
- 1131 Extra Float Download Error
- 1131 Long Download Error
- 1131 Float Download Error
- 1131 Tags Download Error
- 1131 Cfg File Download Error
- 1131 FFB Download Error
- 1131 FFB Link Download Error
- 1131 ID Modules Error
- 1131 Internal Bool Parameters Error
- 1131 MB AI Error
- 1131 MB AO Error
- 1131 MB DI Error
- 1131 MB DO Error
- 1131 Num Net IO Error
- 1131 Net IO DO Error
- 1131 Back/Fore Times Error

4) **1131 Temp Download Error:** Falha no comando de download ou falha na configuração dos módulos de temperatura.

Fazendo o upload dos parâmetros das funções

O botão **Upload Function Parameters**  permite que o usuário atualize os parâmetros das funções. Ao clicar nele a seguinte mensagem aparecerá:



Fig 3. 150 – Confirmando o upload

Após confirmar o comando o **LogicView for FFB** atualizará os parâmetros das funções e a seguinte mensagem aparecerá informando o sucesso da operação.



Fig 3. 151 – Upload finalizado

Na **Output Window** aparecerá uma lista de parâmetros cujos valores mudaram – **NewValue** e **OldValue**. Veja figura seguinte.

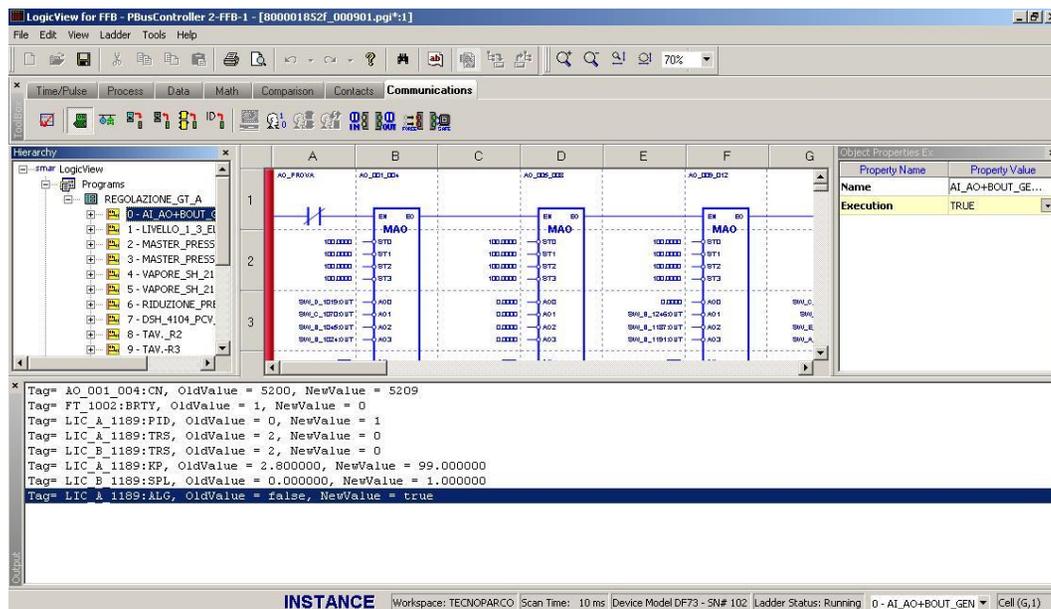


Fig 3. 152 – Lista de parâmetros atualizados

Em **OldValue** é mostrado o valor que está no arquivo da configuração. Em **NewValue** é mostrado o valor que veio do controlador. Ao clicar, na **Output Window**, na linha do parâmetro alterado, o **LogicView for FFB** mostra a função em destaque na ladder.

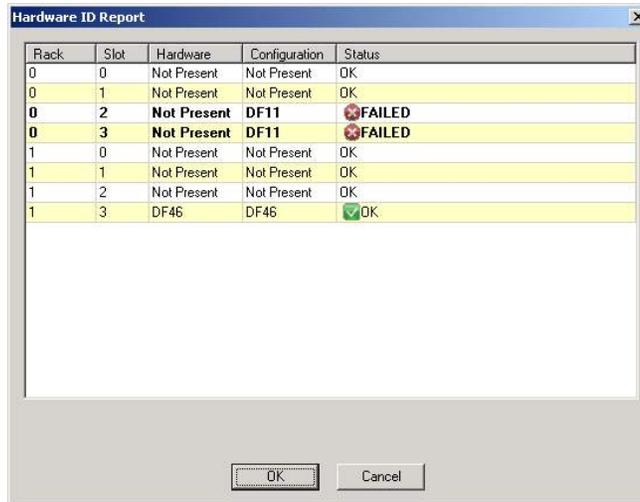
NOTA

Se o usuário não quiser mudar o arquivo da configuração, basta sair sem salvar.

Get Hardware IDs



Ao clicar no botão **Get Hardware IDs** o **LogicView for FFB** busca a instalação real do hardware de E/S e compara com a configuração feita via software. Aparecerá uma janela como a figura seguinte:



| Rack | Slot | Hardware | Configuration | Status |
|------|------|-------------|---------------|--------|
| 0 | 0 | Not Present | Not Present | OK |
| 0 | 1 | Not Present | Not Present | OK |
| 0 | 2 | Not Present | DF11 | FAILED |
| 0 | 3 | Not Present | DF11 | FAILED |
| 1 | 0 | Not Present | Not Present | OK |
| 1 | 1 | Not Present | Not Present | OK |
| 1 | 2 | Not Present | Not Present | OK |
| 1 | 3 | DF46 | DF46 | OK |

Fig 3. 153 – Comparando o hardware com o Get Hardware IDs

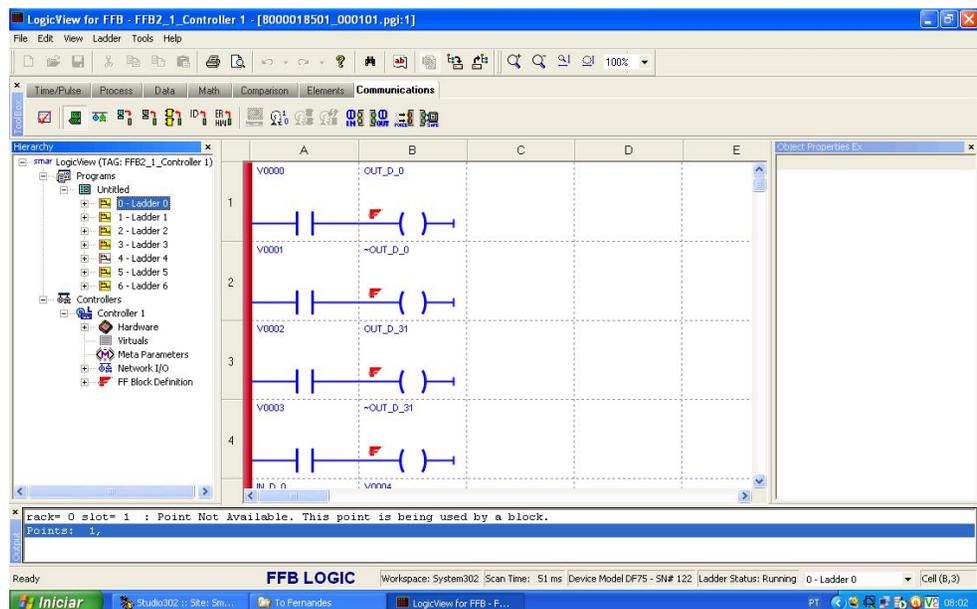
NOTA

A função **Get Hardware IDs** só reconhece a instalação física dos módulos de E/S cujos hardwares tenham número de GLL, que está impresso na placa eletrônica, superior a 1100.

Get Hardware Errors

Ao clicar no botão **Get Hardware Errors** , o usuário obtém informações sobre conflitos de hardware que tenham ocorrido ao configurar acesso ao hardware simultaneamente no **Syscon** e no **LogicView for FFB**. Possíveis conflitos podem ocorrer ao acessar uma mesma saída via blocos ou via ladder, ou ainda se houver configurações conflitantes para o módulo de temperatura DF45.

As informações sobre possíveis conflitos são disponibilizadas na **Output Window**, veja figura seguinte. No exemplo em questão, está indicado que para o módulo localizado no rack 0, slot 1, o ponto 1 está sendo usado tanto pela ladder como pelos blocos. Nesta situação o valor do ponto será definido pelos blocos. Para solucionar o problema o usuário deve retirar o ponto conflitante da ladder ou dos blocos.



The screenshot shows the LogicView for FFB interface. The main window displays a ladder logic diagram with rungs 1 through 4. Rung 1 shows a coil V0000 connected to output OUT_D_0. Rung 2 shows a coil V0001 connected to output -OUT_D_0. Rung 3 shows a coil V0002 connected to output OUT_D_31. Rung 4 shows a coil V0003 connected to output -OUT_D_31. The Output Window at the bottom displays the following message:

```

rack= 0 slot= 1 : Point Not Available. This point is being used by a block.
Points: 1,

```

Fig 3. 154 – Comparando o hardware com o Get Hardware Errors

Caso não ocorram conflitos, aparecerá a mensagem **No Errors**.

IMPORTANTE

- Caso haja conflito de uso ou de configuração de ponto pela ladder e pelos blocos, a preferência será sempre dos blocos.
- Ao entrar em **Supervisão** é realizado automaticamente um **Get Hardware Errors**.

Upload Configuration

Ao clicar no botão **Upload configuration** o **LogicView for FFB** faz o upload completo da configuração que está sendo executada no controlador. Esta opção só está disponível para os controladores DF73, DF75, DF79, DF81, DF89, DF95 e DF97.

Supervisão

O **LogicView for FFB** faz a supervisão de duas maneiras: dos pontos discretos (default) e dos pontos analógicos dos blocos funcionais.



O botão **Discrete Supervision** permite ao usuário monitorar os pontos discretos durante a execução da lógica Ladder no modo online. Para supervisionar, primeiro é preciso que seja feito o Export Tags no Syscon. Se estiver cinza não está supervisionando. Se apenas o botão **Discrete Supervision** estiver pressionado os pontos analógicos aparecerão como cinco pontos de interrogação (?????).



Para supervisionar os pontos analógicos clique no botão **Function Blocks Supervision**. Não há como supervisionar apenas os pontos analógicos. Este botão habilita a supervisão dos pontos de entrada e saída das funções. Para habilitar a supervisão dos parâmetros internos à



função, e conseqüentemente o grid, é necessário clicar no botão **Get Internal Function Parameters**. Esta ação realiza um upload dos parâmetros internos de todas as funções e, nesta condição, é possível atualizar qualquer um destes parâmetros, entrando com um novo valor no grid e teclando <Enter>.

Ao entrar em supervisão, é feita uma comparação entre a configuração que está no controlador e a que está no **LogicView for FFB**. Se forem iguais, os pontos são supervisionados normalmente. Se forem diferentes, aparece a mensagem a seguir, com as opções de **Upload**, **Download** e **Online**. Para não fazer nada feche a caixa de diálogo. Esta mensagem aparece para os controladores DF73, DF75, DF79, DF81, DF89, DF95 e DF97. Para os controladores HFC302, DF62 e DF63 só é perguntado se o usuário quer fazer o download.



Fig 3. 155 – Opções antes da supervisão

NOTA

Se o **LogicView for FFB** não conseguir ler os módulos (o hardware correspondente) das funções analógicas MAI e TEMP o valor da saída vai para 125% do fundo de escala.

Modos de execução da ladder**Stop/Run**

O botão **Stop/ Run** permite que o usuário execute a ladder ou pare sua execução. Ao clicar sobre ele a seguinte janela se abrirá:

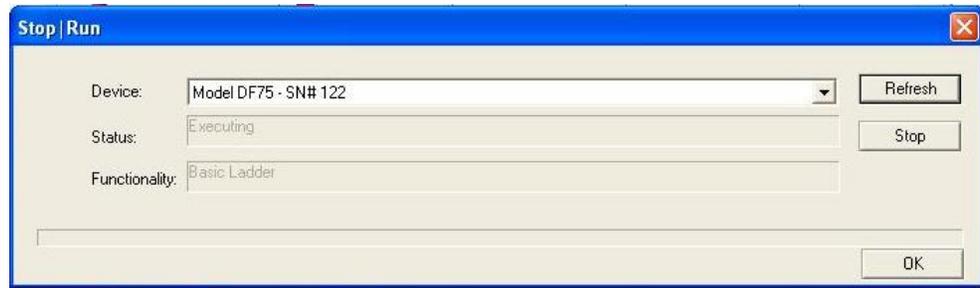


Fig 3. 156 - Janela Stop/Run

Além de servir para disparar ou parar a execução da ladder no device, o ícone **Stop/Run** serve para habilitar o pedido de tempo de scan, basta apenas definir o equipamento. Sempre aparecerá apenas um equipamento, que é aquele no qual a lógica foi descarregada. Em **Run** as entradas são escaneadas, a ladder é executada e as saídas são atualizadas.

Em **Stop** a ladder não é executada, as entradas não são escaneadas e as saídas não são atualizadas. Se a ladder estiver em **Stop**, o Toggle Value está habilitado e o usuário consegue manualmente modificar as saídas (em supervisão). Basta clicar com o botão direito na saída selecionada, depois em Toggle Value e o valor da saída será invertido.

**ATENÇÃO**

Ao executar o comando **Stop**:

- As saídas discretas vão para *Safe Mode*.
- As saídas analógicas do bloco funcional MAO assumirão os valores definidos em ST0, ST1, ST2 e ST3.

Freeze in  – Nesse modo as entradas não são escaneadas, a ladder é executada e as saídas são atualizadas. Ao clicar sobre o ícone a seguinte mensagem aparecerá confirmando a operação:



Fig 3. 157 – Confirmando o modo Freeze In

É possível alterar o Toggle Value das entradas discretas e do FFB. Basta clicar com o botão direito na entrada, depois em Toggle Value e o valor da entrada será invertido. A CPU pode estar em modo **Run** ou **Stop**.

Freeze out  – Nesse modo as entradas são escaneadas, a ladder é executada e as saídas não são atualizadas mantendo-se no último valor. Ao clicar sobre o ícone a seguinte mensagem aparecerá confirmando a operação:



Fig 3. 158 – Confirmando o modo Freeze Out

É possível alterar o Toggle Value das saídas discretas. Basta clicar com o botão direito na saída, depois em Toggle Value e o valor da saída será invertido. A CPU deve estar em modo **Stop**.

Safe Mode  - Nesse modo as entradas são escaneadas, a ladder é executada, mas as saídas se mantêm nos valores de segurança configurados pelo usuário. Ao clicar sobre o ícone a seguinte mensagem aparecerá confirmando a operação:



Fig 3. 159 – Confirmando o modo Safe

Force Mode  - Nesse modo as entradas existentes no hardware são escaneadas, a ladder é executada, as saídas são atualizadas e o usuário consegue atuar sobre as entradas não existentes no hardware. Ao clicar sobre o ícone a seguinte mensagem aparecerá confirmando a operação:



Fig 3. 160 – Confirmando o modo Force

É possível alterar o Toggle Value das entradas discretas e do FFB. Basta clicar com o botão direito na entrada, depois em Toggle Value e o valor da entrada será invertido. A CPU pode estar em modo **Run** ou **Stop**.



ATENÇÃO

- Os modos de execução podem ser simultâneos.
- Se o **LogicView for FFB** estiver com alguns dos modos de execução ativado ao ir para offline, quando retornar para online o modo de execução será mantido. Por exemplo, estando em online e Freeze In é o modo de execução vigente, ao ir para offline e retornar para online, o modo de execução voltará a ser Freeze In automaticamente.

Hierarchy

Esta janela poderá ser habilitada ou desabilitada através do **Menu View**. Nela o usuário poderá verificar toda a estrutura do projeto.

A seguir cada um dos itens que compõem esta janela serão descritos em detalhes.



Fig 3. 161 - Janela Hierarchy

Informações sobre o projeto

O **LogicView for FFB** permite que o usuário cadastre algumas informações sobre o projeto. Ao clicar sobre **LogicView** na aba **Hierarchy** será habilitada na janela **Object Properties** uma série de itens em que serão colocadas as informações sobre o projeto em questão, como por exemplo nome da empresa, da planta, do projeto, controlador, etc.



Fig 3. 162 - Janela Informações de Projeto

Priority - Define a propriedade na qual o controlador executará a lógica em relação às outras tarefas do sistema. Veja a tabela seguinte.

| Prioridade | | DF62 | DF63 | DF73 | DF75 | DF79 | DF81 | DF89 | DF95 | DF97 |
|------------|-----------------|------|------|------|------|------|------|------|------|------|
| 0 | Very High | | | | X | | | | | |
| 0,6 | Intermediate .6 | | | | X | | | | | |
| 0,7 | Intermediate .7 | X | X | X | X | X | X | | X | X |
| 0,8 | Intermediate .8 | X | X | X | X | X | X | | X | X |
| 0,9 | Intermediate .9 | X | X | X | X | X | X | | X | X |
| 1 | High | X | X | X | X | X | X | | X | X |
| 2 | Average | X | X | X | X | X | X | X | X | X |
| 3 | Low | X | X | X | X | X | X | X | X | X |

Foreground/Background - Visualização das taxas de execução da lógica em relação às outras tarefas do sistema. Os valores são alterados de acordo com a definição da prioridade escolhida.

Hardware

Nesta janela pode ser configurado o hardware que executará a lógica ladder. Aqui os racks poderão ser inseridos, excluídos e configurados. O **LogicView for FFB** exibirá uma janela mostrando a ocupação dos racks e seus slots, e também, quais estão disponíveis. A configuração dos racks poderá ser alterada.

Ao clicar com o botão direito do mouse em **Hierarchy** → **Hardware** as seguintes opções estarão disponíveis:

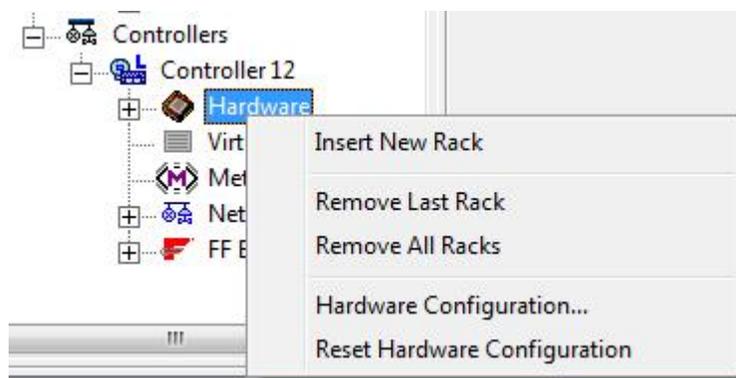


Fig 3. 163 - Opções de Hardware da Janela Hierarchy

Insert New Rack – Com esta opção o usuário poderá incluir quantos racks forem necessários para sua aplicação. Além do Rack Z (DF78 ou DF92) podem ser incluídos 16 racks, numerados de 0 a 15. À medida que foram sendo incluídos aqueles que estão vagos aparecerão em cinza claro. Veja figura a seguir:

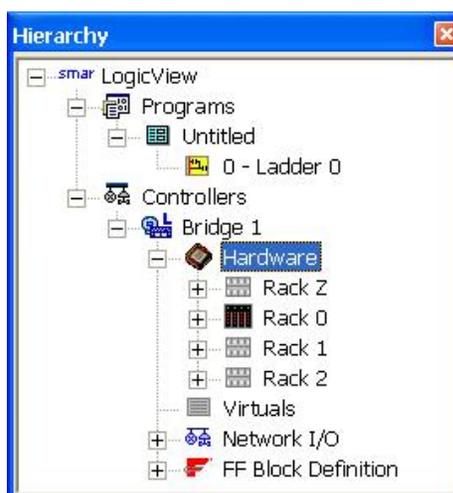


Fig 3. 164 - Inserindo racks

Remove Last Rack – Ao escolher esta opção o último rack será removido da aplicação independente se ele estiver vazio ou não. Esta opção será desabilitada se o último rack disponível for o Rack 0.

Remove All Racks – Com esta opção o usuário poderá excluir todos os racks inseridos, exceto os racks Z e 0, de uma só vez independente se eles estiverem vazios ou não.

ATENÇÃO
As operações **Remove Last Rack** e **Remove All Racks** não poderão ser desfeitas.

Reset Hardware Configuration - Com esta opção o usuário poderá mudar de E/S normal para E/S redundante (IOR) ou vice-versa, mas toda a configuração de hardware original será perdida e todos os pontos de E/S digitais serão convertidos para metaparâmetros.

Hardware Configuration – Ao escolher esta opção o usuário poderá escolher quais módulos participarão da lógica ladder. Inicialmente, o usuário deve escolher qual o tipo de módulo de E/S será utilizado, convencional ou redundante. Esta operação não pode ser desfeita. Veja figura seguinte.

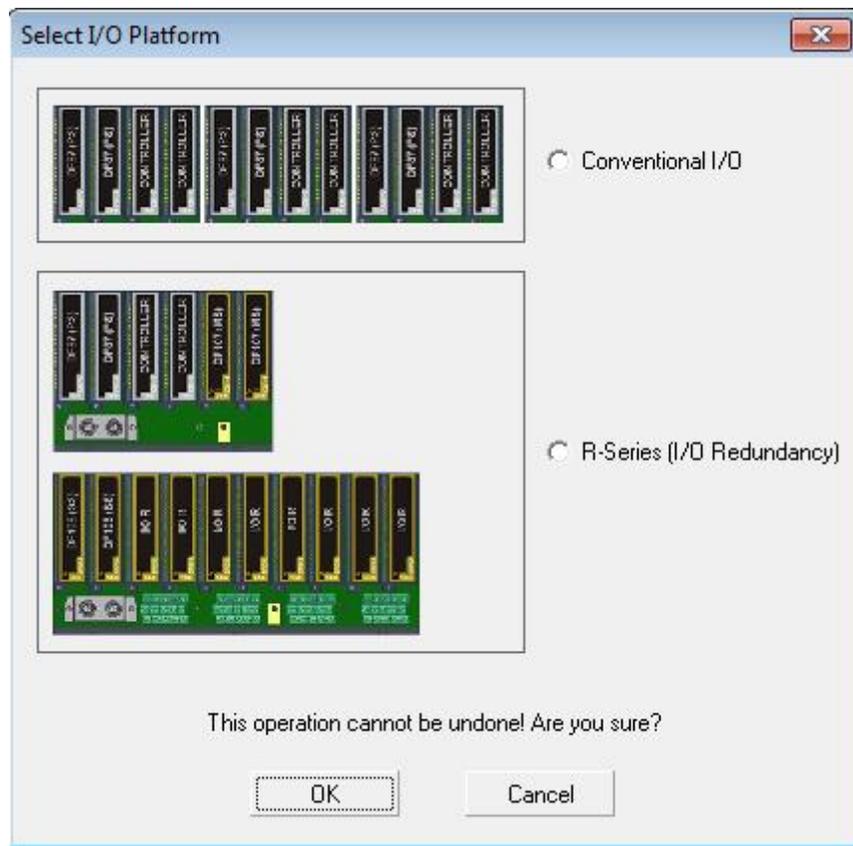


Fig 3. 165 – Escolhendo o tipo de módulo de E/S

Caso o usuário escolha usar módulos de E/S convencionais será apresentada a figura seguinte.

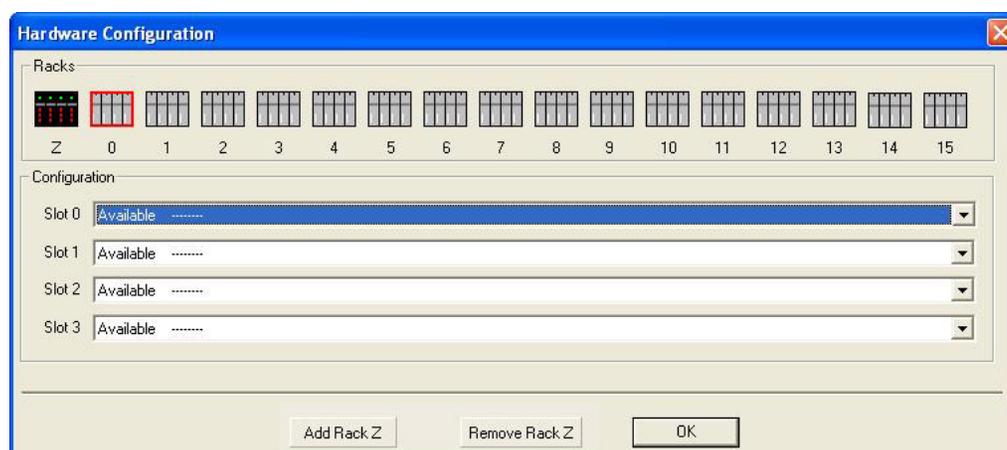


Fig 3. 166 - Configurando o Hardware

Caso o usuário escolha usar módulos redundantes a janela que será exibida é a seguinte:

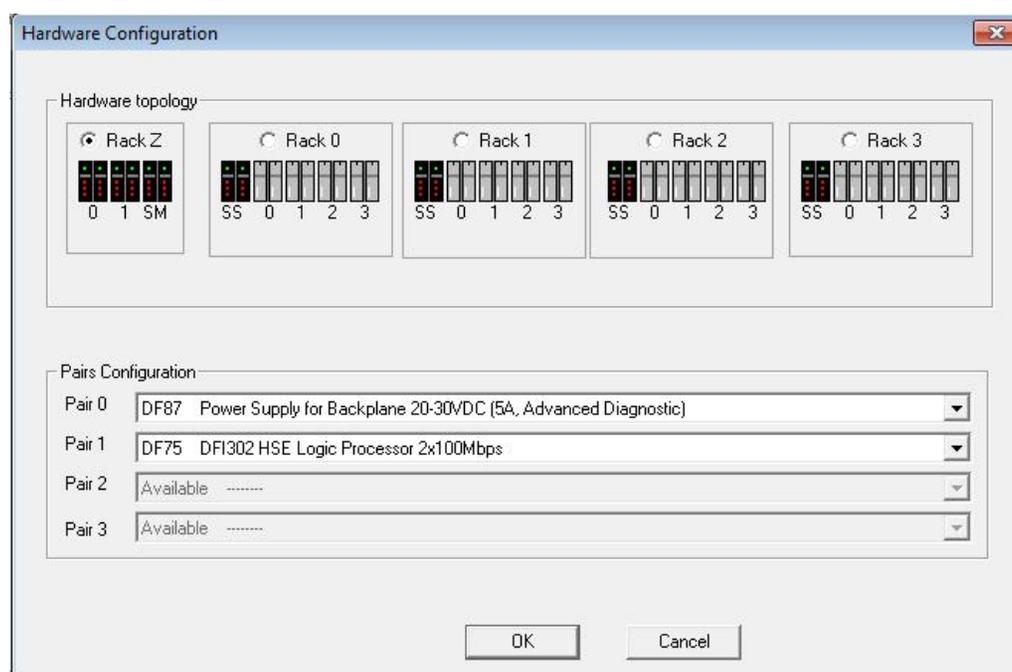


Fig 3. 167 - Configurando o hardware para módulos redundantes

Conforme apresentado no tópico **Menu File**, se o usuário estiver no **Modo Template** e criar um novo projeto com a opção **File** → **New** o **LogicView for FFB** criará o novo arquivo com o Rack Z vazio e o Rack 0 com o slot 0 preenchido pela fonte de alimentação DF50 e o slot 1 preenchido pelo controlador DF62.

O usuário poderá escolher se necessita ou não utilizar o Rack Z (DF78 ou DF92) em sua configuração. Este rack só poderá ser usado para redundância de fontes e controladores (CPUs). Para maiores detalhes veja o manual do DFI302.

O Rack Z poderá ser inserido a qualquer momento tanto no **Modo Instance** quanto no **Modo Template**. Clique em **Add Rack Z** e ele será inserido. Automaticamente a fonte de alimentação e a CPU que estavam nos slots 0 e 1 do Rack 0 serão transferidas para os respectivos slots no Rack Z. Desta forma os slots 0 e 1 do Rack 0 ficarão disponíveis e no Rack Z existirá a redundância de fonte e CPU. Veja a figura a seguir.

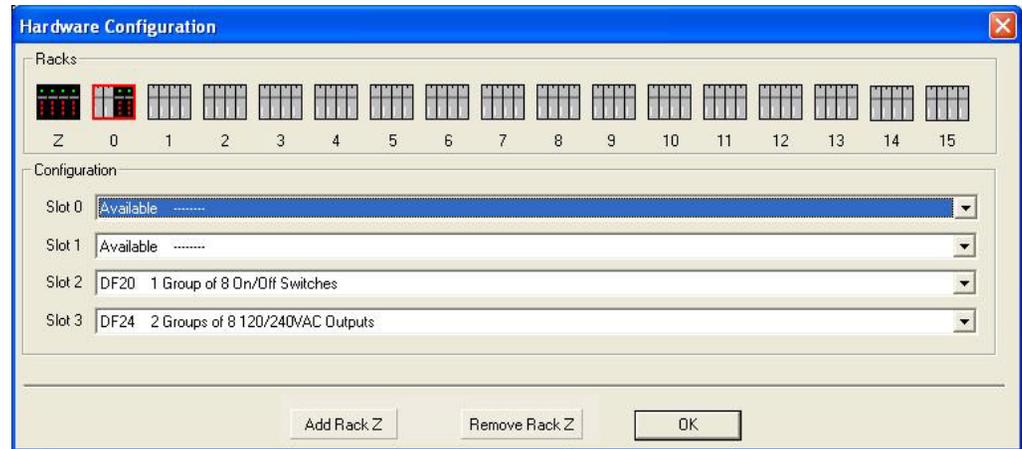


Fig 3. 168 – Incluindo o Rack Z

Se o usuário tentar acessar o Rack Z sem antes inseri-lo a seguinte mensagem aparecerá:



Fig 3. 169 – Erro ao acessar o Rack Z

Nos slots 0 e 1 do Rack Z só poderão ser colocadas fontes de alimentação e elas podem ser de tipos diferentes. Se o usuário tentar colocar algum módulo que não seja uma fonte de alimentação a seguinte mensagem aparecerá:



Fig 3. 170 – Erro de inserção de módulos no Rack Z (1)

Nos slots 2 e 3 do Rack Z só poderão ser colocados os controladores (CPUs) e eles necessariamente devem ser do mesmo tipo. Se o usuário inserir no slot 2 o DF73, automaticamente, o slot 3 será configurado também com o DF73 e vice-versa. Se o usuário tentar inserir algum outro tipo de módulo, que não seja uma CPU, a seguinte mensagem será exibida:



Fig 3. 171 – Erro de inserção de módulos no Rack Z (2)

Ao iniciar um projeto no **LogicView for FFB** o Rack Z será criado, mas estará vazio. O Rack 0 já virá com os dois primeiros slots preenchidos. No slot 0 estará a fonte de alimentação DF50 e no slot 1 estará o controlador DF62. Os slots 0 e 1 só poderão ser configurados com fonte de alimentação e controlador, respectivamente, neste caso em que o Rack Z não está sendo utilizado.

Os slots 2 e 3 do Rack 0 e todos os outros slots dos racks 1 a 15 poderão ser configurados com qualquer tipo de módulo, exceto controladores. Caso o usuário tente fazê-lo a seguinte mensagem aparecerá:



Fig 3. 172 – Erro - Inserção de CPU em slot errado

Se o usuário estiver trabalhando no **Modo Template** poderá alterar o tipo de controlador – DF62, DF63, DF73, DF75, DF79, DF81, DF89, DF95 e DF97. Esta escolha dependerá da aplicação do usuário.

Se o usuário estiver no **Modo Instance** o controlador já virá configurado pelo **Syscon** e, neste caso, não poderá ser alterado. Caso o usuário tente fazê-lo a seguinte mensagem aparecerá:



Fig 3. 173 – Erro - Mudando CPU no Modo Instance

O usuário poderá remover o Rack Z através do botão **Remove Rack Z**. Automaticamente a fonte de alimentação, que estava nos slots 0 e 1 do Rack Z, e o controlador, que estava nos slots 2 e 3 do Rack Z, serão transferidos para os respectivos slots no Rack 0.

Caso os slots 0 e 1 do Rack 0 já estiverem preenchidos, não será possível remover o Rack Z. As seguintes mensagens aparecerão:



Fig 3. 174 – Erro - Remoção do Rack Z (1)



Fig 3. 175 – Erro - Remoção do Rack Z (2)

Escolhendo os módulos - Ao clicar no slot desejado uma lista de opções será aberta como mostrado na figura seguinte.

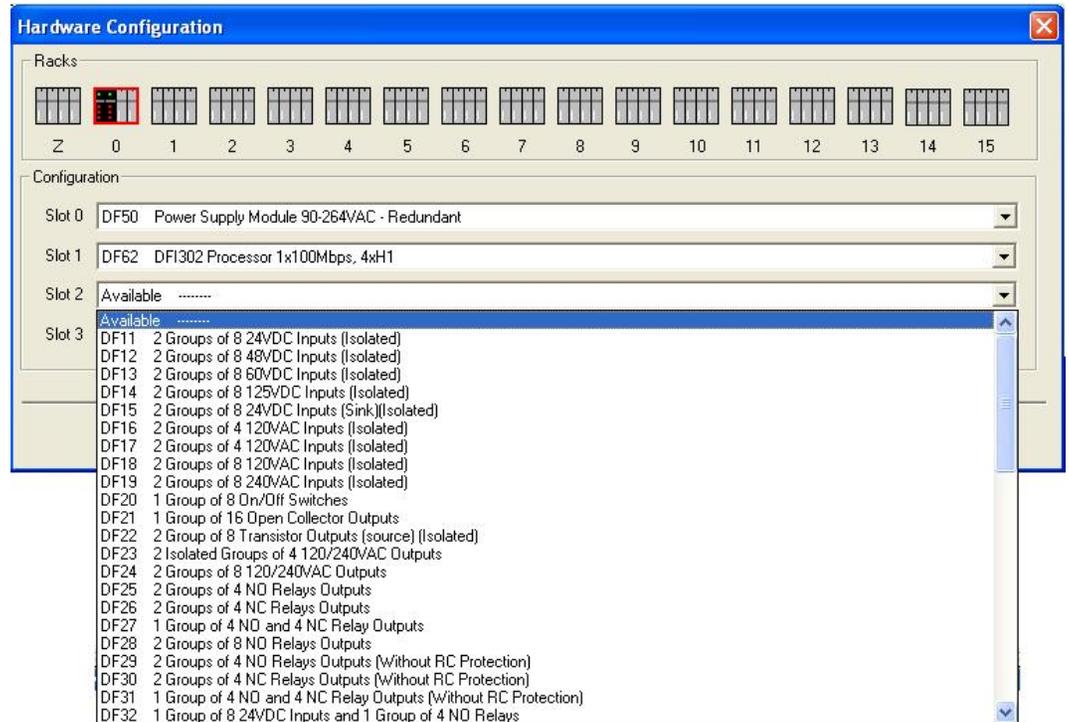


Fig 3. 176 - Configurando o Hardware

O usuário deverá escolher qual o módulo necessário à sua aplicação. Basta clicar sobre ele e automaticamente estará disponibilizado no rack. Clique **OK**.

A ocupação dos racks poderá ser vista diretamente na janela **Hierarchy**. Veja a figura seguinte.

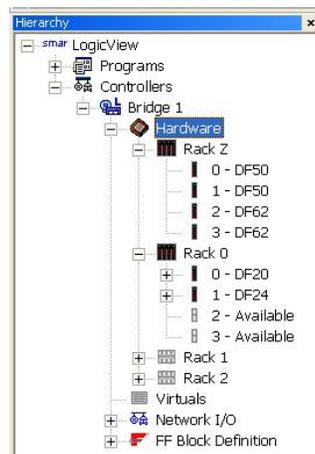


Fig 3. 177 - Ocupação dos Racks

Se o usuário desejar pode copiar a configuração do rack e colá-la em outro rack. Para isso ele deve clicar com o botão direito sobre o rack origem e escolher **Copy Configuration**. Para colar a configuração copiada basta ir ao rack destino e escolher **Paste Configuration** e o **LogicView for FFB** substituirá a configuração que porventura estiver no rack destino.

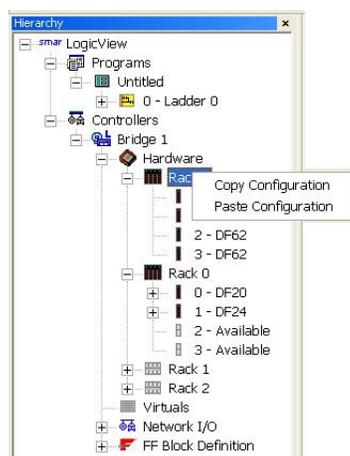


Fig 3. 178 - Copiando e Colando a Configuração dos Racks

Se o usuário quiser copiar e colar apenas um módulo realizará procedimento semelhante ao anterior. Ao clicar com o botão direito sobre o módulo origem e escolhendo **Copy Module**, o módulo será copiado e poderá ser colado em outro slot. Para isso basta escolher o slot destino e após acionar o botão direito do mouse escolher **Paste Module**. Veja a figura abaixo.

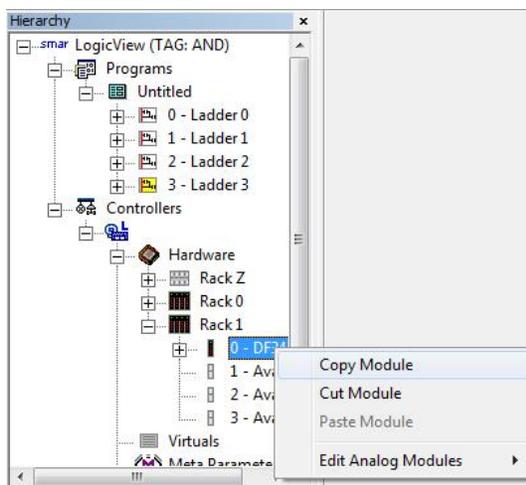


Fig 3. 179 - Copiando e Colando a Configuração dos Módulos

Também é possível movimentar os módulos, fazendo primeiramente a operação **Cut Module** e a seguir **Paste Module**. Ao se realizar estas operações, os elementos que estão com tags diferentes do default manterão os mesmos tags e se referenciarão ao ponto na nova posição do módulo. Se o elemento tiver um tag default, por exemplo, TAG01000, o tag será renomeado de acordo com a nova posição ocupada pelo ponto, seguindo a notação TAGRRSGP, onde *R* = Rack, *S* = Slot, *G* = Grupo e *P* = Ponto.

Configurando Safe Output Values

O usuário poderá configurar os valores de segurança das saídas dos módulos em caso de falha. No exemplo mostrado abaixo o Rack 0 tem um módulo DF24. O usuário deverá clicar com o botão direito na saída a ser configurada e escolher o valor desejado – **On** ou **Off**. Por default todas as saídas estão configuradas com o valor **Off** ao iniciar um projeto.

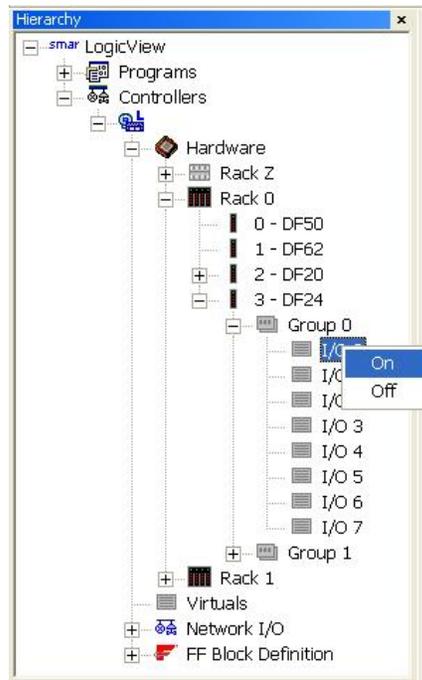


Fig 3. 180 – Configurando os valores “Safe Output”(1)

O valor de segurança poderá ser configurado também na janela **Object Properties**. Após selecionar a saída a ser configurada, dê um duplo clique na célula à direita de Safe Output Value e escolha o valor desejado – **On** ou **Off**. Veja a figura abaixo.

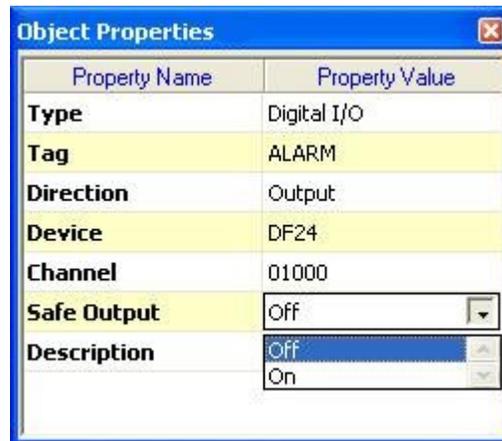


Fig 3. 181 – Configurando os valores “Safe Output” (2)

Em **Tools** → **Properties Editor** os valores de segurança poderão ser configurados. Clique na saída a ser configurada e escolha On ou Off. Veja figura a seguir.

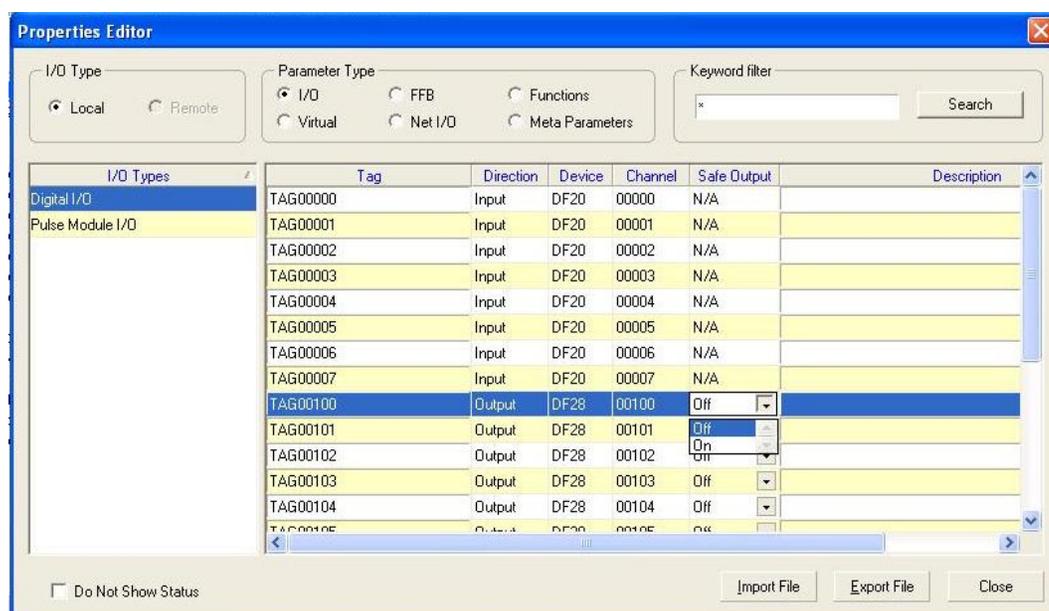


Fig 3. 182 – Configurando os valores “Safe Output” (3)

Alterando os tags

Os tags podem ser alterados na janela **Object Properties** se o usuário clicar sobre o elemento desejado – variáveis virtuais, I/O ou blocos funcionais – na janela **Hierarchy**. Após selecionar o elemento basta clicar duas vezes na célula à direita de Tag na janela **Object Properties** e escrever o novo tag.

Os tags só podem ter caracteres alfanuméricos e “underscore”. Os tags também não podem conter espaços. Caso contrário as seguintes mensagens aparecerão.

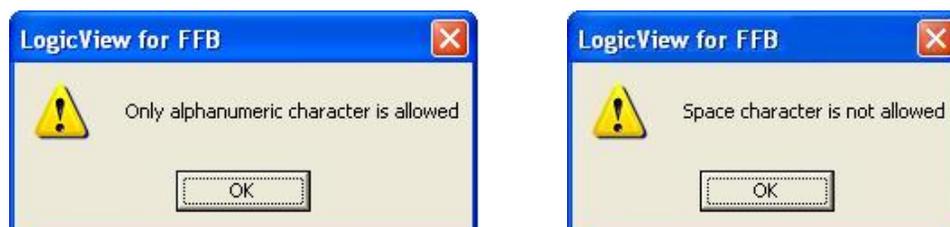


Fig 3. 183 – Erro - Alterando os Tags com caracteres não permitidos

O usuário será notificado se colocar um tag em branco no elemento selecionado. Veja figura seguinte.



Fig 3. 184 – Erro – Tag em branco

NOTA

Os tags das variáveis virtuais, das entradas e saídas podem ter no máximo 16 caracteres. Os tags dos blocos funcionais podem ter no máximo 10 caracteres.

Os tags dos blocos funcionais são únicos. Se o usuário colocar um tag que já é utilizado por algum outro bloco funcional a seguinte mensagem aparecerá.



Fig 3. 185 – Erro – Tag já existe

NOTAOs tags também podem ser alterados em **Tools** → **Properties editor**.**Inserindo uma descrição**

O usuário poderá inserir uma descrição do elemento selecionado para facilitar sua identificação. Após selecionar o elemento basta clicar duas vezes na célula à direita de **Description** na janela **Object Properties** e escrever a descrição, que pode ter até 64 caracteres.



Fig 3. 186 – Inserindo uma descrição

Módulos Especiais

Alguns tipos de módulos analógicos podem ser editados – entrada analógica, saída analógica, entrada de pulso e temperatura.

Módulo de Entradas Analógicas

Os módulos de entradas analógicas disponíveis são DF43, DF44 e DF57. Após inseri-los conforme descrito anteriormente, eles devem ser configurados.

Com o botão direito do mouse, abra a janela de configuração do módulo. Para tal selecione as opções **Edit Analog Modules** e em seguida **Edit Module Inputs**.

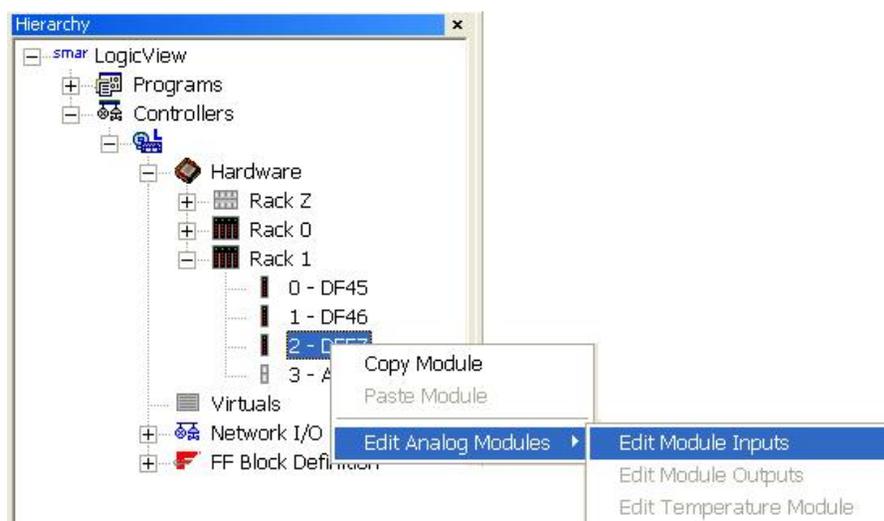


Fig 3. 187 - Configurando Módulos de Entradas Analógicas

Irá aparecer a janela mostrada na figura a seguir, onde poderão ser alteradas as configurações do módulo. Cada canal, ou ponto, possui configuração independente dos demais (num total de 8). Selecione o canal desejado através da opção **Select Channel**. A configuração padrão para todos os canais é a mostrada na figura a seguir.

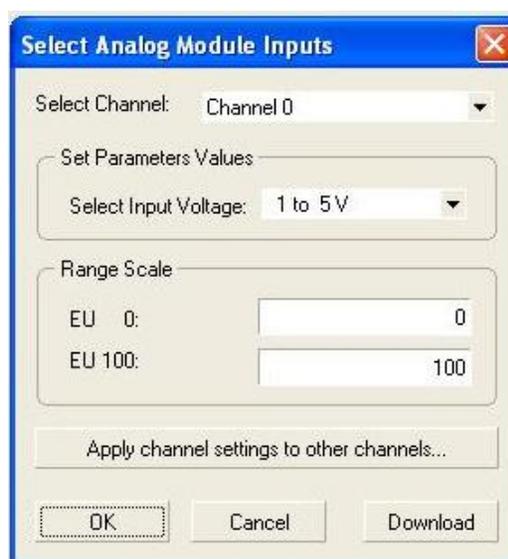


Fig 3. 188 - Alterando a configuração dos Módulos de Entradas Analógicas

A opção **Select Input Voltage** corresponde à faixa dos valores na entrada do canal. Os tipos permitidos são:

- 1 a 5 V (default);
- 0 a 5 V (default);
- -10 a 10 V (default);
- 0 a 10 V (default);

ATENÇÃO

Observe a configuração física do módulo e jumpers a serem colocados no manual do respectivo módulo.

A escala de engenharia para apresentação dos dados pode ser modificada, seguindo a seguinte regra:

- Eng. Unit 0 (EU0) é o valor mínimo.
- Eng. Unit 100 (EU100) é o valor máximo.

Caso $EU0 = 0$ e $EU100 = 1$, o valor apresentado será na faixa de 0 a 10000 (discretos) da faixa de entrada, ou seja, se **Select Input Voltage** igual a “1 a 5 V”, o valor 0 na saída do bloco corresponderá à entrada de 1 V no canal e o valor 10000 na saída do bloco corresponderá à entrada de 5 V no canal. Valores intermediários na entrada de tensão apresentarão na saída valores interpolados entre 0 e 10000.

Para outros valores de $EU0$ e $EU100$, o valor apresentado será na faixa especificada em $EU0$ e $EU100$ da faixa de entrada, ou seja, se **Select Input Voltage** igual a “1 a 5 V”, o valor de $EU0$ igual a 10 e o valor de $EU100$ igual a 50, na saída do bloco teremos para uma entrada de 1 V no canal, a saída terá valor 10 e, para uma entrada de 5 V no canal, teremos a saída do bloco com valor 50. Valores intermediários na entrada de tensão apresentarão na saída valores interpolados entre $EU0$ e $EU100$.

O botão **Apply channel settings to other channels**, serve para o usuário, após configurar um canal, replicar esta mesma configuração para outros. Basta selecionar os canais desejados conforme a figura seguinte.

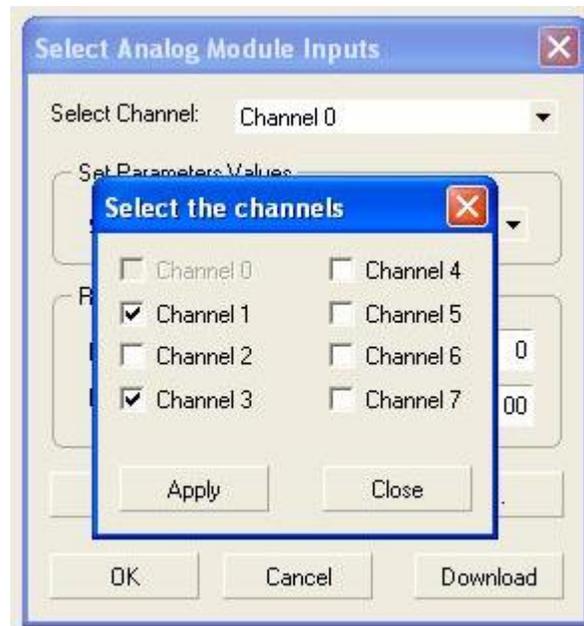


Fig 3. 189 – Selecionando os canais para replicar a configuração

NOTAS

- Configurar sempre $EU0 < EU100$.
- A opção **Download** pode ser usada, estando o **LogicView for FFB** online, para fazer o download apenas das escalas, caso elas tenham sido alteradas.
- Será feito o download da configuração de todos os módulos do mesmo tipo e não apenas do módulo alterado.

Módulo de Saídas Analógicas

O módulo de saída analógica disponível é o DF46. Após inseri-lo conforme descrito anteriormente, ele deve ser configurado.

Com o botão direito do mouse, abra a janela de configuração do módulo. Para tal selecione as opções **Edit Analog Modules** e em seguida **Edit Module Outputs**.

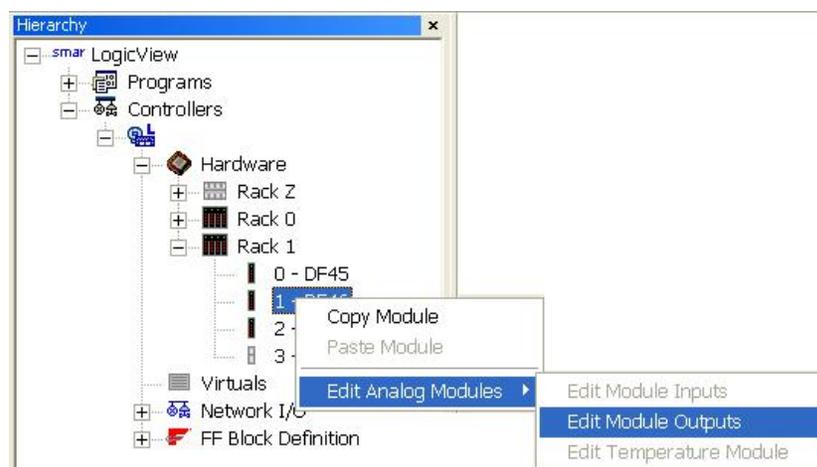


Fig 3. 190 - Configurando Módulos de Saídas Analógicas

Irá aparecer a janela mostrada na figura a seguir, onde poderão ser alteradas as configurações do módulo. Cada canal, ou ponto, possui configuração independente dos demais (num total de 4). Selecione o canal desejado através da opção **Select Channel**. A configuração padrão para todos os canais é a mostrada na figura a seguir.

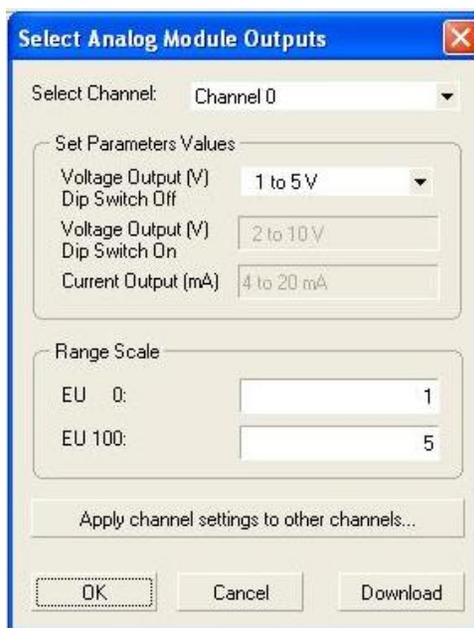


Fig 3. 191 - Alterando a configuração dos Módulos de Saídas Analógicas

A opção **Voltage Output (V)** corresponde à faixa dos valores na saída do canal. Caso a saída seja em corrente, o valor correspondente selecionado está em **Current Output (mA)**. Os tipos permitidos são:

- 1 to 5 V (default)
- 0 to 5 V (default)
- -5 to 5 V (default)

A escala de engenharia para entrada do bloco pode ser modificada, seguindo a seguinte regra:

- Eng. Unit 0 (EU0) é o valor mínimo.
- Eng. Unit 100 (EU100) é o valor máximo.

Caso $EU0 = 0$ e $EU100 = 1$, o valor da entrada deverá ter valores entre 0 e 10000 que serão convertidos na faixa de saída, ou seja, se **Voltage Output (V)** igual a "1 a 5 V", o valor 0 na entrada do bloco corresponderá à saída de 1 V no canal e o valor 10000 na entrada do bloco corresponderá à saída de 5 V no canal. Valores intermediários na entrada serão interpolados na saída com valores entre 1 e 5V.

Para outros valores de EU0 e EU100, o valor de entrada será convertido para a faixa especificada em EU0 e EU100 configuradas, ou seja, se **Voltage Output (V)** igual a “1 a 5 V”, o valor de EU0 igual a 10 e o valor de EU100 igual a 50, se na entrada do bloco temos 10, corresponderá a uma saída de 1 V no canal, caso a entrada seja 50, corresponderá a uma saída de 5 V no canal. Valores intermediários na entrada do bloco apresentarão na saída valores interpolados entre 1 e 5 V.

O botão **Apply channel settings to other channels**, serve para o usuário, após configurar um canal, replicar esta mesma configuração para outros. Basta selecionar os canais desejados conforme mostrado na figura seguinte.

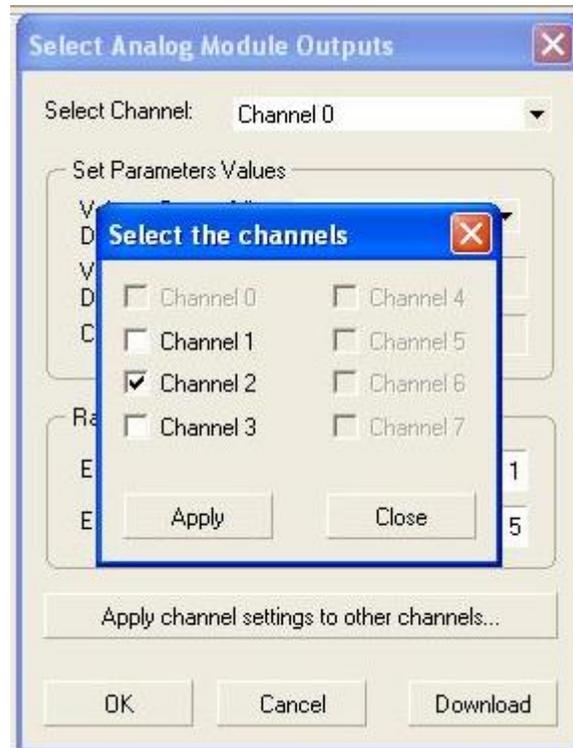


Fig 3. 192 – Selecionando os canais para replicar a configuração

NOTAS

- Configurar sempre EU0 < EU100.
- A opção **Download** pode ser usada, estando o **LogicView for FFB** online, para fazer o download apenas das escalas, caso elas tenham sido alteradas.
- Será feito o download da configuração de todos os módulos do mesmo tipo e não apenas do módulo alterado.

Módulos de Entradas de Pulso

Os módulos de entrada de pulso disponíveis são DF41, DF42 e DF67. Após inseri-los conforme descrito anteriormente, eles devem ser configurados.

A configuração de cada ponto no módulo de entrada de pulsos é feita individualmente. Para tal, expanda a “árvore” de I/Os como na figura seguinte.

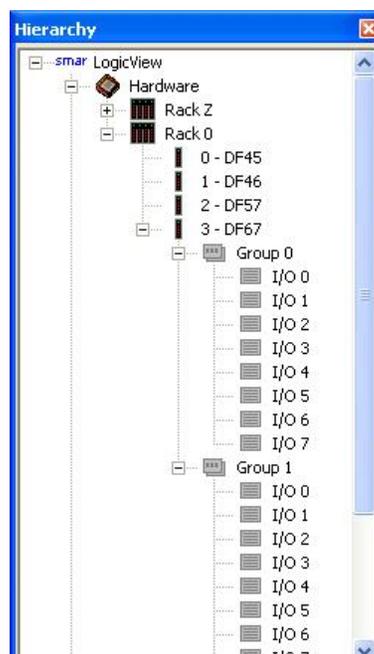


Fig 3. 193 - Configurando Módulos de Entradas de Pulso

Cada entrada deve ser configurada pela janela **Object Properties**, onde são disponibilizados os seguintes parâmetros de configuração:

- Description – Descrição do ponto.
- Q Value for THR On – Limite superior de histerese de vazão para que seja configurada a saída THR dos blocos ACC e ACC_N.
- Q Value for THR Off – Limite inferior de vazão para que seja zerada a saída THR dos blocos ACC e ACC_N.

OBSERVAÇÃO

Configurar sempre Q Value for THR Off < Q Value for THR On.

- Pulse Count Period – Período (em ms) para cálculo da vazão, por exemplo, quando se deseja saber a vazão para um intervalo de tempo de 2 em 2 segundos configura-se esse parâmetro com 2000 (2 s é igual a 2000 ms).



Fig 3. 194 - Configurando o Pulse Count Period

Módulo de Temperatura

O módulo de temperatura disponível é o DF45. Após inseri-lo conforme descrito anteriormente, ele deve ser configurado.

Com o botão direito do mouse, abra a janela de configuração do módulo. Para tal selecione as opções **Edit Analog Modules** e em seguida **Edit Temperature Module**.

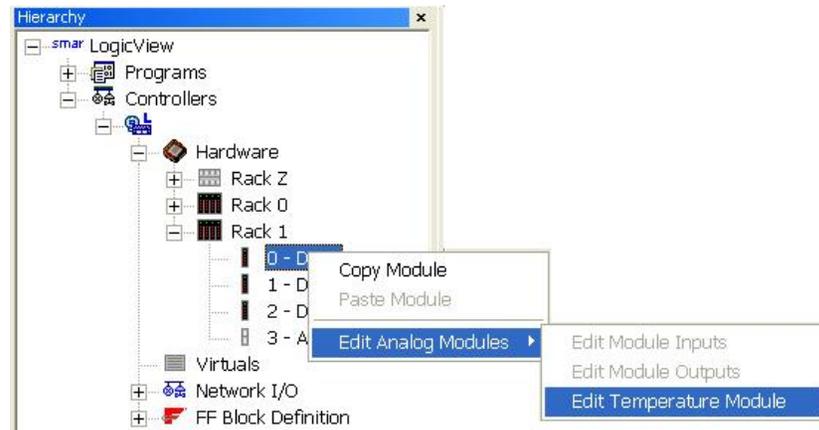


Fig 3. 195 - Configurando Módulo de Temperatura

Irá aparecer a janela mostrada na figura a seguir, onde poderão ser alteradas as configurações do módulo. Cada canal, ou ponto, possui configuração independente dos demais (num total de 8). Selecione o canal desejado através da opção **Select Channel**. A configuração padrão para todos os canais é a mostrada na figura abaixo.

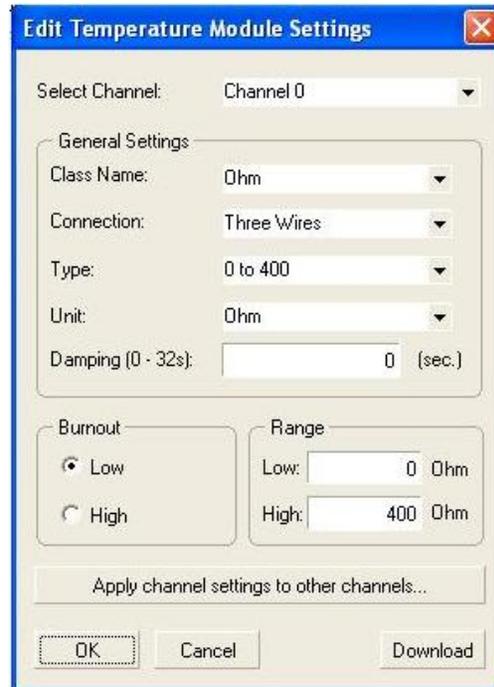


Fig 3. 196 - Alterando a configuração do Módulo de Temperatura

O botão **Apply channel settings to other channels**, serve para o usuário, após configurar um canal, replicar esta mesma configuração para outros. Basta selecionar os canais desejados conforme mostrado na figura seguinte.

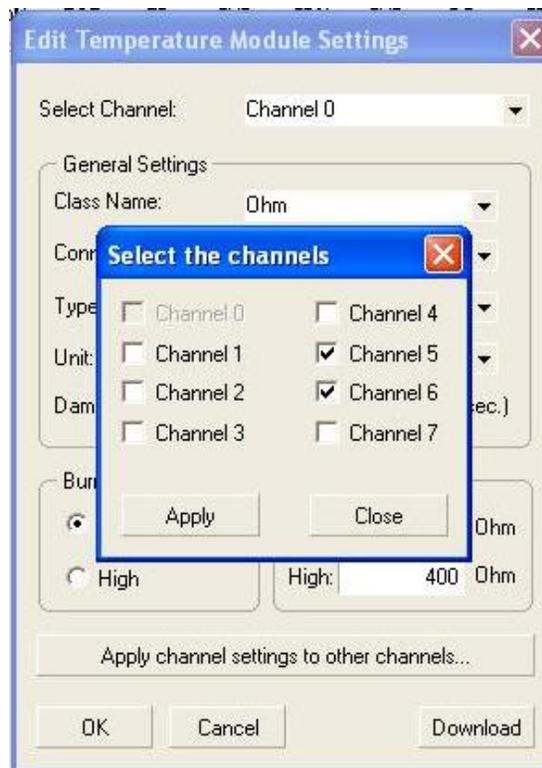


Fig 3. 197 – Seleccionando os canais para replicar a configuração

NOTAS

- A opção **Download** pode ser usada, estando o **LogicView for FFB** online, para fazer o download apenas das escalas, caso elas tenham sido alteradas.
- Será feito o download da configuração de todos os módulos do mesmo tipo e não apenas do módulo alterado.

A checagem da unidade de engenharia e do tipo de conexão está relacionada a classe do sensor. A tabela a seguir mostra quais os relacionamentos possíveis.

| Classe | Nome da Classe | Conexão | Nome da Conexão | Tipo | Nome do Tipo | Min | Max |
|--------|----------------|---------|-----------------|------|---------------|-------|------|
| 1 | RTD | 1 | Differential | 1 | Cu10 GE | -270 | 270 |
| 1 | RTD | 1 | Differential | 2 | Ni120 DIN | -320 | 320 |
| 1 | RTD | 1 | Differential | 3 | Pt50 IEC | -1050 | 1050 |
| 1 | RTD | 1 | Differential | 4 | Pt100 IEC | -1050 | 1050 |
| 1 | RTD | 1 | Differential | 5 | Pt500 IEC | -1050 | 1050 |
| 1 | RTD | 1 | Differential | 6 | Pt50 JIS | -850 | 850 |
| 1 | RTD | 1 | Differential | 7 | Pt100 JIS | -800 | 800 |
| 1 | RTD | 2 | 2 Wires | 1 | Cu10 GE | -20 | 250 |
| 1 | RTD | 2 | 2 Wires | 2 | Ni120 DIN | -50 | 270 |
| 1 | RTD | 2 | 2 Wires | 3 | Pt50 IEC | -200 | 850 |
| 1 | RTD | 2 | 2 Wires | 4 | Pt100 IEC | -200 | 850 |
| 1 | RTD | 2 | 2 Wires | 5 | Pt500 IEC | -200 | 450 |
| 1 | RTD | 2 | 2 Wires | 6 | Pt50 JIS | -200 | 600 |
| 1 | RTD | 2 | 2 Wires | 7 | Pt100 JIS | -200 | 600 |
| 1 | RTD | 3 | 3 Wires | 1 | Cu10 GE | -20 | 250 |
| 1 | RTD | 3 | 3 Wires | 2 | Ni120 DIN | -50 | 270 |
| 1 | RTD | 3 | 3 Wires | 3 | Pt50 IEC | -200 | 850 |
| 1 | RTD | 3 | 3 Wires | 4 | Pt100 IEC | -200 | 850 |
| 1 | RTD | 3 | 3 Wires | 5 | Pt500 IEC | -200 | 450 |
| 1 | RTD | 3 | 3 Wires | 6 | Pt50 JIS | -200 | 600 |
| 1 | RTD | 3 | 3 Wires | 7 | Pt100 JIS | -200 | 600 |
| 2 | TC | 1 | Differential | 151 | B NBS | -1600 | 1600 |
| 2 | TC | 1 | Differential | 152 | E NBS | -1100 | 1100 |
| 2 | TC | 1 | Differential | 153 | J NBS | -600 | 900 |
| 2 | TC | 1 | Differential | 154 | K NBS | -1550 | 1550 |
| 2 | TC | 1 | Differential | 155 | N NBS | -1400 | 1400 |
| 2 | TC | 1 | Differential | 156 | R NBS | -1750 | 1750 |
| 2 | TC | 1 | Differential | 157 | S NBS | -1750 | 1750 |
| 2 | TC | 1 | Differential | 158 | T NBS | -600 | 600 |
| 2 | TC | 1 | Differential | 159 | L DIN | -1100 | 1100 |
| 2 | TC | 1 | Differential | 160 | U DIN | -800 | 800 |
| 2 | TC | 2 | 2 Wires | 151 | B NBS | 100 | 1800 |
| 2 | TC | 2 | 2 Wires | 152 | E NBS | -100 | 1000 |
| 2 | TC | 2 | 2 Wires | 153 | J NBS | -150 | 750 |
| 2 | TC | 2 | 2 Wires | 154 | K NBS | -200 | 1350 |
| 2 | TC | 2 | 2 Wires | 155 | N NBS | -100 | 1300 |
| 2 | TC | 2 | 2 Wires | 156 | R NBS | 0 | 1750 |
| 2 | TC | 2 | 2 Wires | 157 | S NBS | 0 | 1750 |
| 2 | TC | 2 | 2 Wires | 158 | T NBS | -200 | 400 |
| 2 | TC | 2 | 2 Wires | 159 | L DIN | -200 | 900 |
| 2 | TC | 2 | 2 Wires | 160 | U DIN | -200 | 600 |
| 3 | mV | 1 | Differential | 213 | -500 to 500 | -500 | 500 |
| 3 | mV | 1 | Differential | 214 | -5000 to 5000 | -5000 | 5000 |
| 3 | mV | 2 | 2 Wires | 201 | -6 to 22 | -6 | 22 |
| 3 | mV | 2 | 2 Wires | 202 | -10 to 100 | -10 | 100 |
| 3 | mV | 2 | 2 Wires | 203 | -50 to 500 | -50 | 500 |
| 4 | Ohm | 1 | Differential | | -100 to 100 | -100 | 100 |
| 4 | Ohm | 1 | Differential | | -400 to 400 | -400 | 400 |
| 4 | Ohm | 2 | 2 Wires | 51 | 0 to 100 | 0 | 100 |
| 4 | Ohm | 2 | 2 Wires | 52 | 0 to 400 | 0 | 400 |
| 4 | Ohm | 2 | 2 Wires | 53 | 0 to 2000 | 0 | 2000 |
| 4 | Ohm | 3 | 3 Wires | 51 | 0 to 100 | 0 | 100 |
| 4 | Ohm | 3 | 3 Wires | 52 | 0 to 400 | 0 | 400 |
| 4 | Ohm | 3 | 3 Wires | 53 | 0 to 2000 | 0 | 2000 |

A faixa pode ser configurada dentro da faixa máxima especificada na tabela. São esses valores que serão utilizados em **Burnout**.

Configuração dos módulos HART

Os módulos HART disponíveis são DF116 (entrada) e DF117 (saída). Após inseri-los conforme descrito anteriormente, eles devem ser configurados.

Clique no módulo com o botão direito do mouse para abrir sua janela de configuração. Selecione a opção **Edit Analog Modules** e, em seguida, **Edit Module Inputs** (para o DF116) ou **Edit Module Outputs** (para o DF117). A seguinte janela abrirá:

Fig 3. 198 – Configurando os módulos HART

Desta forma poderão ser alteradas as configurações do módulo. Cada módulo possui 8 canais, sendo que cada um deles corresponde a um equipamento conectado ao respectivo canal. Selecione o canal desejado através da opção **Select Channel**. A configuração padrão para todos os canais é a mostrada na figura anterior. Para cada equipamento, os respectivos **VAR_CODES** das variáveis **PV**, **SV**, **TV**, **QV**, **5V**, **6V**, **7V** e **8V** poderão ser configurados, valores válidos de 0 a 255. A escala de engenharia para o bloco, referente ao valor da corrente, pode ser modificada, de acordo com a seguinte regra:

- Eng. Unit 0 (EU0) é o valor mínimo, correspondendo ao valor de 4 mA para a corrente.
- Eng. Unit 100 (EU100) é o valor máximo, correspondendo ao valor de 20 mA para a corrente.

O botão **Apply channel settings to other channels**, serve para o usuário, após configurar um canal, replicar esta mesma configuração para outros. Basta selecionar os canais desejados. E para o DF117 está habilitada a opção para configurar o comportamento em segurança, **Safe Behavior**, que indica para qual valor irá a corrente primária do equipamento HART caso ele entre em modo de segurança: 3,6 mA ou 21 mA.

NOTAS

- A opção **Download** pode ser usada, estando o **LogicView for FFB** online, para fazer o download apenas dos “var codes” e do “safe behavior”, no caso do DF117.
- Será feito o download da configuração de todos os módulos do mesmo tipo e não apenas do módulo alterado.

Configuração dos módulos de E/S redundantes

- **Módulo de entrada digital - DF111**

Todos os pontos de entrada digital redundantes possuem, além de uma variável representando o valor do ponto (0 ou 1), uma variável representando o status do mesmo (0 - good ou 1 - bad). O tag do status é o mesmo do valor, com o acréscimo do ~ (til) antes do tag. Os status são *read only*.

| Property Name | Property Value |
|---------------|----------------|
| Variable | Digital I/O |
| Tag | TAG00000 |
| Direction | Input |
| Device | DF111 |
| Channel | 00000 |
| Safe Output | N/A |
| Description | |
| ===== | ===== |
| Variable | Digital I/O |
| Tag | ~TAG00000 |
| Direction | Input |
| Device | DF111 |
| Channel | 00020 |
| Description | |

Fig 3. 199 – Valor e status módulo DF111

- **Módulo de saída digital - DF112**

Todos os pontos de saída digital redundantes possuem, além de uma variável representando o valor do ponto (0 ou 1), uma variável representando o status do mesmo (0 - good ou 1 – bad). O tag do status é o mesmo do valor, com o acréscimo do ~ (til) antes do tag. Os status são *read only*.

Cada valor possui um valor de segurança respectivo, que pode ser **On** (1) ou **Off** (0). E cada ponto também possui um comportamento de segurança, que no caso das saídas digitais podem ser último valor (*last value*) ou o valor configurado como *Safe Value*. Veja figura seguinte.

| Property Name | Property Value |
|----------------|----------------|
| Variable | Digital I/O |
| Tag | TAG00000 |
| Direction | Output |
| Device | DF112 |
| Channel | 00000 |
| Safe Behaviour | Safe Value |
| Safe Output | Off |
| Description | |
| ===== | ===== |
| Variable | Digital I/O |
| Tag | ~TAG00000 |
| Direction | Output |
| Device | DF112 |
| Channel | 00020 |
| Description | |

Fig 3. 200 – Valor e status módulo DF112

- **Módulo de entrada analógica - DF113**

Após inseri-lo conforme descrito anteriormente, ele deve ser configurado. Clique no módulo com o botão direito do mouse para abrir sua janela de configuração. Selecione a opção **Edit Analog Modules** e, em seguida, **Edit Module Inputs**. A seguinte janela abrirá:

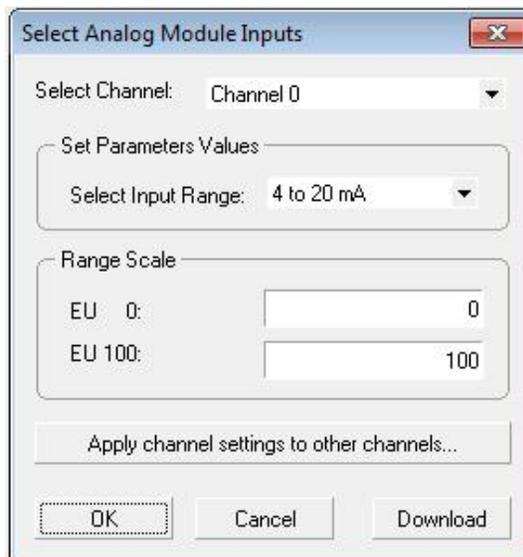


Fig 3. 201 – Configurando os módulos de entrada analógica

Nos campos mostrados na figura anterior poderão ser alteradas as configurações do módulo. Cada canal, ou ponto, possui configuração independente dos demais (num total de 8). Selecione o canal desejado através da opção **Select Channel**. A configuração padrão para todos os canais é a mostrada na figura anterior.

A opção **Select Input Range** corresponde à faixa dos valores na entrada do canal. Os tipos permitidos são:

- 4 a 20 mA
- 0 a 20 mA

A escala de unidade de engenharia para apresentação dos dados pode ser modificada, seguindo a seguinte regra:

- Eng. Unit 0 (EU0) é o valor mínimo.
- Eng. Unit 100 (EU100) é o valor máximo.

O valor apresentado na saída do bloco **MAIx** será na faixa especificada em EU0 e EU100 relativa à faixa de entrada. Por exemplo, se **Select Input Range** for igual a **4 a 20 mA**, o valor de EU0 for igual a **10** e o valor de EU100 igual a **50**, na saída do bloco teremos para uma entrada de 4 mA no canal, uma saída de valor 10. Para uma entrada de 20 mA no canal, teremos a saída do bloco com valor 50. Valores intermediários na entrada de corrente apresentarão na saída valores interpolados entre EU0 e EU100.

O botão **Apply channel settings to other channels**, serve para o usuário, após configurar um canal, replicar esta mesma configuração para outros.

NOTAS

- Configurar sempre $EU0 < EU100$.
- A opção **Download** pode ser usada, estando o **LogicView for FFB** online, para fazer o download apenas das escalas, caso elas tenham sido alteradas.
- Será feito o download da configuração de todos os módulos do mesmo tipo e não apenas do módulo alterado.

- **Módulo de saída analógica - DF114**

Após inseri-lo conforme descrito anteriormente, ele deve ser configurado. Clique no módulo com o botão direito do mouse para abrir sua janela de configuração. Selecione a opção **Edit Analog Modules** e, em seguida, **Edit Module Outputs**. A seguinte janela abrirá:

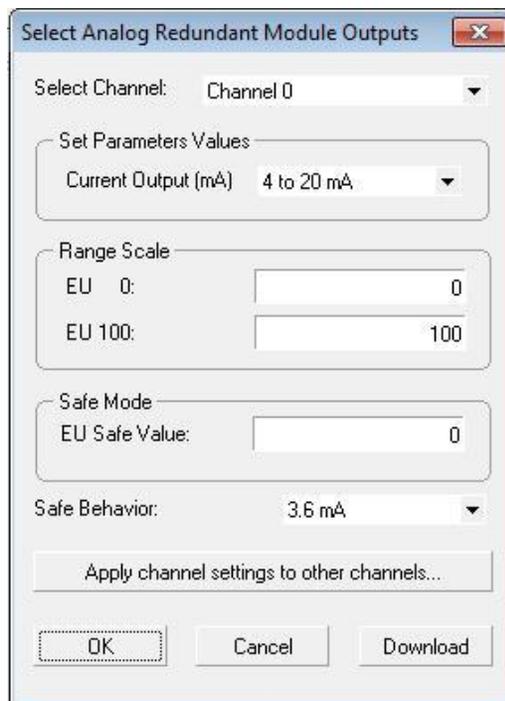


Fig 3. 202 – Configurando os módulos redundantes de saída analógica

Nos campos mostrados na figura anterior poderão ser alteradas as configurações do módulo. Cada canal, ou ponto, possui configuração independente dos demais (num total de 8). Selecione o canal desejado através da opção **Select Channel**. A configuração padrão para todos os canais é a mostrada na figura anterior.

A opção **Current Output (mA)** corresponde à faixa dos valores na saída do canal. Os tipos permitidos são:

- 4 a 20 mA
- 0 a 20 mA
- 0 a 21 mA

A escala de unidade engenharia para apresentação dos dados pode ser modificada, seguindo a seguinte regra:

- Eng. Unit 0 (EU0) é o valor mínimo.
- Eng. Unit 100 (EU100) é o valor máximo.

O valor apresentado na entrada do bloco **MAOx** será na faixa especificada em EU0 e EU100 relativa à faixa de saída. Por exemplo, se **Current Output (mA)** for igual a **4 a 20 mA**, o valor de EU0 for igual a **10** e o valor de EU100 for igual a **50**, se na entrada do bloco tivermos um valor de 10, na saída do canal teremos 4 mA. Para uma entrada igual 50, na saída teremos 20 mA. Valores intermediários na entrada entre EU0 e EU100, gerarão na saída de corrente valores interpolados entre 4 e 20 mA.

Para cada ponto de saída pode ser configurado separadamente um valor de segurança, no parâmetro **EU Safe Value**, e um comportamento de segurança em **Safe Behavior**, do qual existem 4 modos:

- **3,6 mA**: o ponto gerará 3,6 mA na saída;
- **21 mA**: o ponto gerará 21 mA na saída;
- **Safe Value**: o ponto irá para o valor de segurança configurado para ele;
- **Last Value**: o ponto permanecerá no último valor antes de ter entrado em segurança.

O botão **Apply channel settings to other channels**, serve para o usuário, após configurar um canal, replicar esta mesma configuração para outros.

NOTAS

- Configurar sempre EU0 < EU100.
- A opção **Download** pode ser usada, estando o **LogicView for FFB** online, para fazer o download apenas das escalas, caso elas tenham sido alteradas.
- Será feito o download da configuração de todos os módulos do mesmo tipo e não apenas do módulo alterado.

Programs

Um programa é um conjunto de ladders. O número de ladders que podem ser implementadas vai depender da quantidade de elementos de cada ladder e da capacidade do controlador. Na janela **Programs** o usuário pode gerenciar as redes ladder da aplicação.

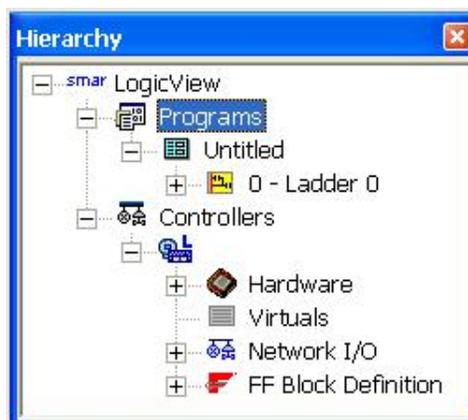


Fig 3. 203 - O Item Programs da Janela Hierarchy

Se o usuário desejar pode dar um nome ao seu programa, basta clicar em **Untitled** e na janela **Object Properties** clicar duas vezes na célula à direita de **Name**. Neste local deve escrever o nome do programa.



Fig 3. 204 - Alterando o nome do programa

O usuário poderá inserir diagramas ladder clicando com o botão direito no nome do programa e em seguida escolher **Insert New Diagram**. Quando um novo diagrama é inserido é realizado um build automaticamente.

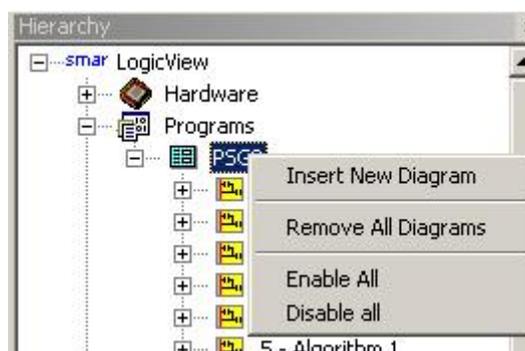


Fig 3. 205 - Inserindo diagramas Ladder

Se o usuário diminuir o número de diagramas (ou excluir algum diagrama) o **LogicView for FFB** exibirá a seguinte mensagem. O usuário deverá confirmar ou não sua operação.



Fig 3. 206 - Confirmando a mudança do número de diagramas ladder

O usuário poderá excluir todos os diagramas ladder de uma só vez. Clicando no nome do programa e depois em **Remove All** todos os diagramas serão removidos. O usuário também precisará confirmar esta operação.

O usuário poderá habilitar ou desabilitar todos os diagramas clicando com o botão direito no nome do programa e depois em **Enable All** ou **Disable All**, respectivamente. Veja a figura 3.205. Se a ladder estiver habilitada ficará com o símbolo preenchido de amarelo. Caso contrário ficará sem preenchimento.

Se o usuário quiser alterar a posição do diagrama no programa deverá clicar com o botão direito sobre ele e escolher o movimento desejado – **Move Up** ou **Move Down**. Isto mudará a ordem de execução deste diagrama no programa.

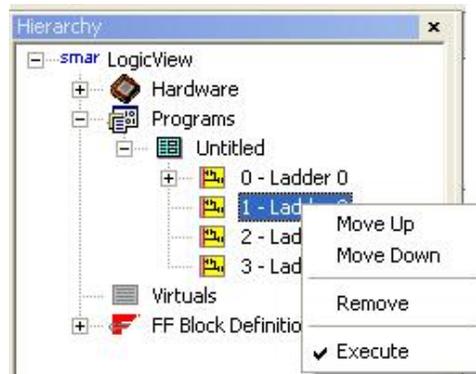


Fig 3. 207 - Alterando a posição dos diagramas ladder

O diagrama ladder pode ser removido facilmente. Clique com o botão direito sobre ele e depois em **Remove**. O **LogicView for FFB** solicitará ao usuário a confirmação ou não da operação.

Os diagramas poderão ser habilitados ou desabilitados individualmente. Na figura acima o diagrama 2 está habilitado. Se o usuário quiser desabilitá-lo basta retirar o símbolo ✓ de **Execute**. Quando esta opção é realizada um build é feito automaticamente.

A execução ou não do diagrama também poderá ser definida na janela **Object Properties**. Veja figura seguinte.



Fig 3. 208 – Definindo a execução dos diagramas ladder

O usuário poderá alterar o nome do diagrama ladder, basta clicar nele, por exemplo em **0 – Ladder 0** e na janela **Object Properties** clicar duas vezes na célula à direita de **Name**. Neste local deve escrever o novo nome da ladder. Este nome pode ter até 64 caracteres.

Virtuals

Este item serve para definir o número de parâmetros virtuais.

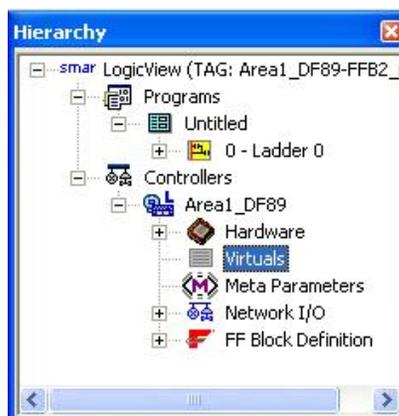


Fig 3. 209 - Definindo os parâmetros virtuais

Clicando nele, a janela **Object Properties**, aparecerá da seguinte forma:



Fig 3. 210 – Object Properties - Definindo a quantidade de variáveis virtuais

Em **Virtuals**, o usuário pode definir a quantidade de parâmetros virtuais do diagrama ladder. O valor default é 10 e pode ser mudado pelo usuário para até 4096.

Em escolhendo uma variável virtual na **Área de Desenho da Ladder** o usuário pode mudar o tag dele na janela **Object Properties**. Basta um duplo clique na célula à direita de **Tag** e o modo de edição será habilitado. Com um duplo clique na célula à direita de **Description**, o usuário pode alterar ou inserir uma descrição que melhor identifique o parâmetro virtual em questão. Veja figura seguinte.

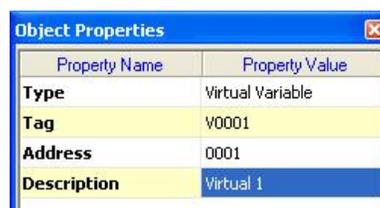


Fig 3. 211 – Object Properties - Definindo os parâmetros das variáveis virtuais

Diminuindo o número de parâmetros virtuais - Por exemplo tem-se um diagrama ladder e a quantidade de parâmetros virtuais definida é 15. No diagrama, os parâmetros virtuais que ocupam os endereços 0013 e 0014 estão sendo utilizados. Se o usuário diminuir a quantidade de parâmetros para 10, automaticamente os parâmetros dos endereços 0013 e 0014 serão excluídos

do diagrama. O usuário só será avisado do problema quando executar um **Build** . Na janela **Output** aparecerão os erros detectados. Veja a figura seguinte.

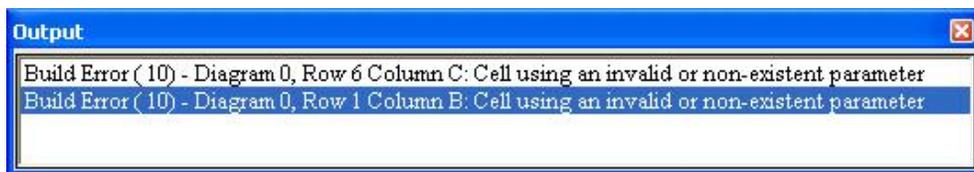


Fig 3. 212 - Erro ao diminuir a quantidade de parâmetros virtuais

FF Block Definition

Este item serve para definir os parâmetros FFB que podem ser: Digital Inputs, Digital Outputs, Analog Inputs ou Analog Outputs.

Ao clicar com o botão direito sobre o nome do projeto aparecerá a opção **Define FFB Parameters**.

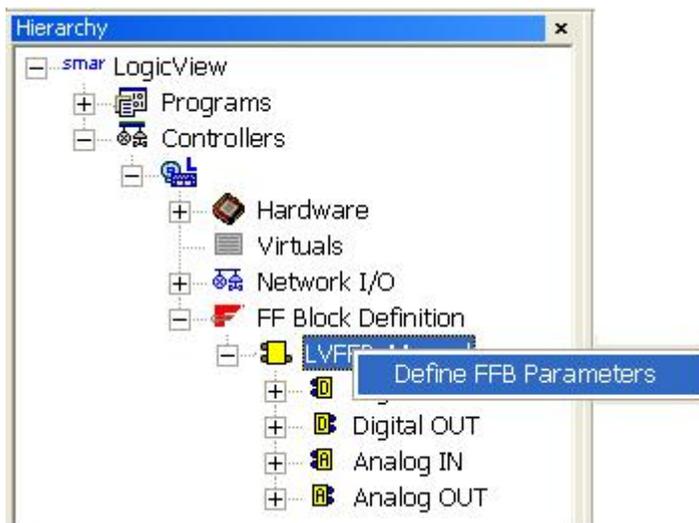


Fig 3. 213 - O Item FF Block Definition da Janela Hierarchy

Escolhendo-a a seguinte janela aparecerá e nela poderá ser configurada a quantidade de entradas e saídas analógicas e digitais. Os tags destes podem ser editados. Dê um duplo clique na célula IN_D_0_LABEL, por exemplo, e o modo de edição será habilitado. Veja a figura seguinte.

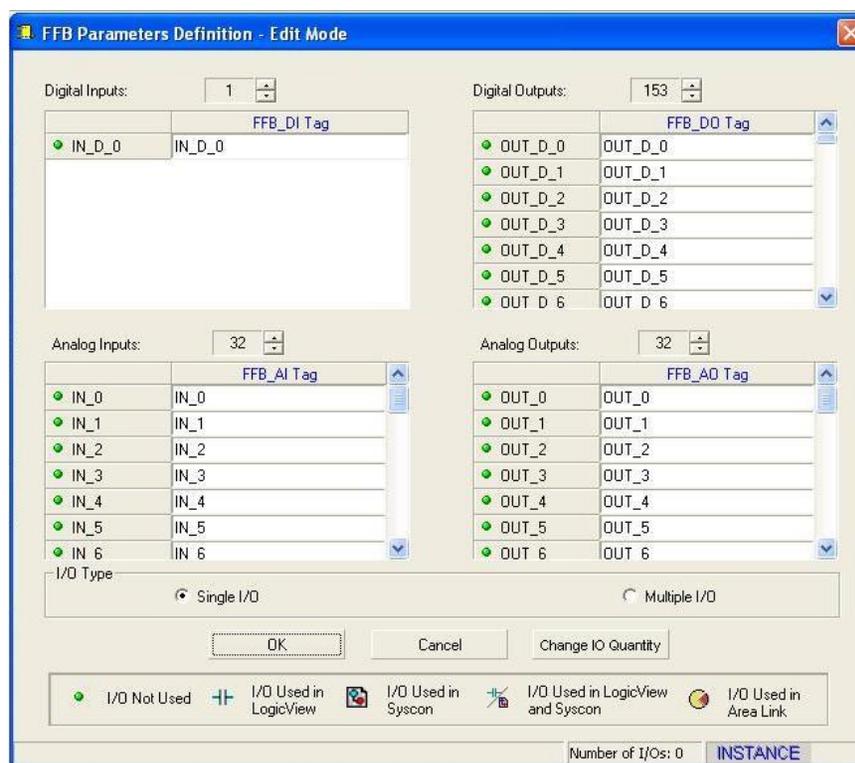


Fig 3. 214 - Definição dos parâmetros FFB

Este procedimento pode ser feito via **Syscon**, no modo Instance, antes de editar a lógica. Para maiores detalhes veja o manual do **Syscon**.

Os tags das entradas e saídas do FFB são únicos. Se o usuário colocar um tag que já é utilizado por algum outro elemento a seguinte mensagem aparecerá.



Fig 3. 215 – Erro ao mudar um tag dos parâmetros FFB

NOTA

Quando o bloco FFB for utilizado na estratégia de controle, é recomendado prever parâmetros reservas para uso futuro evitando assim um impacto de parada do controle durante um download incremental, o qual será necessário em inclusão de nova estratégia com novos parâmetros. É sabido que a inclusão de novos parâmetros no FFB, assim como mudança de nome do parâmetro, redefinirá as DDs do equipamento, e isto exigirá um download mais amplo culminando com exclusão de links e blocos e restabelecimento destes. A utilização dos parâmetros reservas já previstos, não redefinirá novas DDs e exigirá apenas estabelecimento dos links novos, utilizando os parâmetros reservados já existentes.

Object properties

Esta é a janela em que o usuário pode verificar as propriedades do elemento selecionado e alterá-las se for necessário.

A janela poderá ser habilitada através do **Menu View**, como foi visto anteriormente.



Fig 3. 216 - A Janela Object Properties

Os itens que aparecem em cinza claro não podem ser alterados pelo usuário. Aqueles que permitem edição aparecem em preto. As opções vão variar dependendo do tipo de elemento selecionado.

Naqueles que estão disponíveis basta clicar duas vezes e uma lista *drop-down* aparecerá. Assim o usuário poderá escolher a opção que lhe convier.

Área de desenho da Ladder

Esta é a área onde a lógica ladder é montada. Ela tem 80 linhas (de 1 a 80) e 32 colunas (de A a FF), ou seja, 2560 células.

Os elementos inseridos (contatos, bobinas e blocos de funções) podem ser excluídos com a tecla

DEL do teclado ou através do botão .

As conexões verticais podem ser excluídas usando o botão .

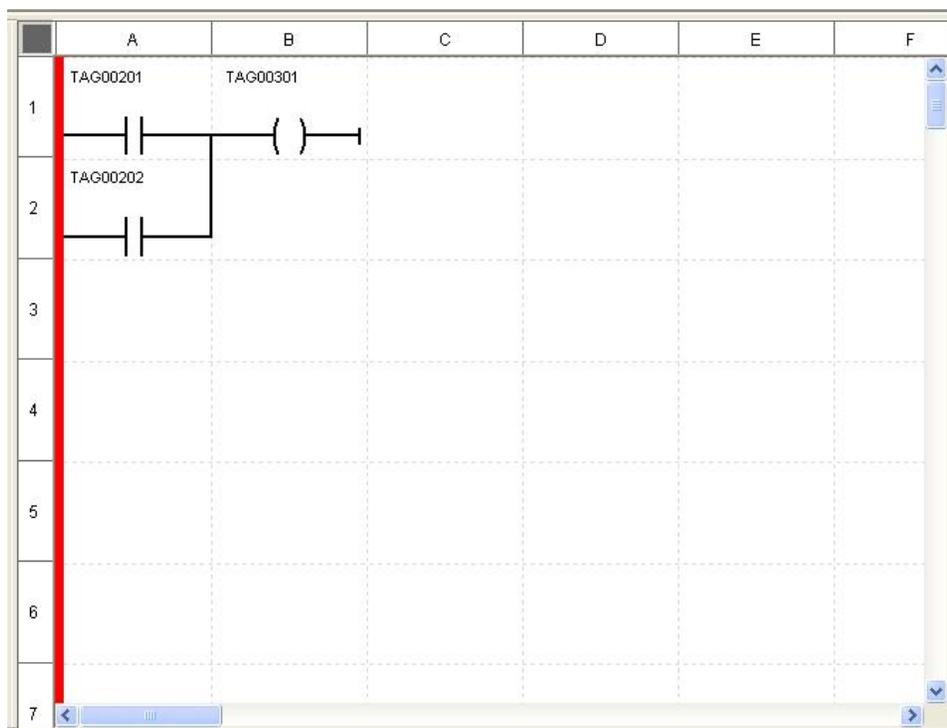


Fig 3. 217 - A Área de Desenho da Ladder

Existem restrições para onde os blocos podem ser inseridos. Isto está relacionado com o tamanho do bloco e elementos na vizinhança, assim, às vezes o usuário deverá selecionar outro local para inserir o elemento ou bloco de função. A seguinte mensagem aparecerá:



Fig 3. 218 - Alerta sobre inserção de elemento na célula

Insert/Remove Blank Line

O **LogicView for FFB** tem o recurso de inserir ou remover linhas vazias em um diagrama ladder para facilitar a edição e/ou modificação de lógicas.

Para inserir ou remover uma linha no diagrama, selecione uma célula, clique com o botão direito. O seguinte o menu abrirá.

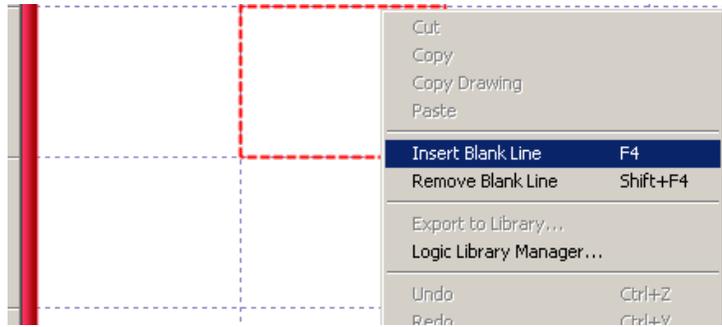


Fig 3. 219 – Menu Insert/Remove Blank Line

As teclas de atalho – F4 para inserir uma linha e Shift+F4 para remover uma linha também podem ser utilizadas.

Ao inserir uma linha vazia todos os elementos do diagrama serão deslocados para baixo, a partir da linha selecionada. Caso não seja possível deslocar os elementos por falta de espaço disponível na parte inferior do diagrama, a seguinte mensagem será exibida:



Fig 3. 220 – Erro – Inserindo linhas vazias

Remover uma linha vazia irá deslocar todos os elementos do diagrama para cima, a partir da linha selecionada. Caso a linha selecionada não esteja vazia, a seguinte mensagem de erro será exibida:



Fig 3. 221 – Erro – Removendo linhas vazias

Output

Esta é a janela em que o usuário pode ver a estatística da Ladder e os erros da configuração. A janela poderá ser habilitada ou desabilitada através do **Menu View**, como foi visto anteriormente, e também será habilitada clicando em **Build** ou **Simulation**. A seguinte janela será aberta:

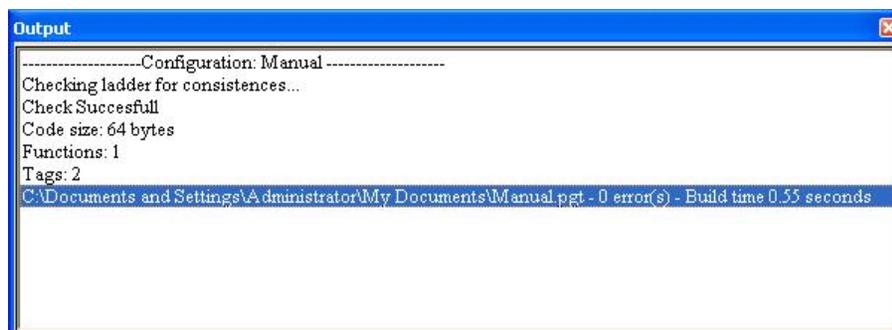


Fig 3. 222 - A Janela Output

Se ao clicar em **Build** o **LogicView for FFB** detectar algum erro, este será mostrado na janela **Output**. Veja a figura a seguir:

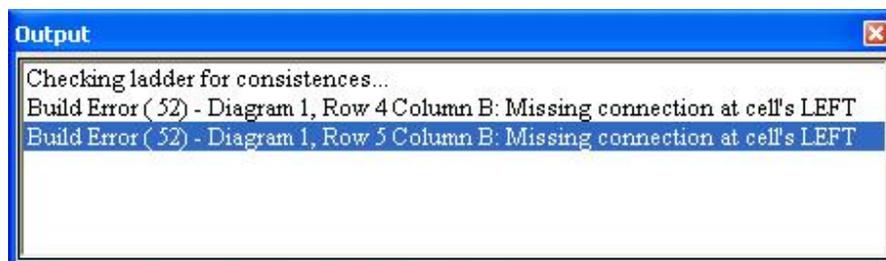


Fig 3. 223 - Erros de Build na Janela Output

No exemplo acima foram detectados dois erros. Ao clicar sobre algum dos erros na janela **Output** o **LogicView for FFB** levará o usuário direto para o erro. O ponto do erro estará piscando em amarelo na Área de Desenho da Ladder.

Status Bar

Esta é a parte da área de trabalho que dá ao usuário informações importantes sobre o status da aplicação.

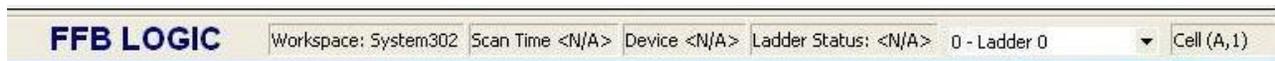


Fig 3. 224 - Barra de Status

- Workspace: indica sobre qual workspace ativo a instância está trabalhando.
- Scan time já foi explicado no item **Menu Ladder**.
- Sync Time: Tempo para sincronismo da lógica entre controladores redundantes. Zero significa que não há sincronismo ou que o sincronismo não está pronto. Quando o Sync Time é igual a Scan Time, os controladores estão sendo mantidos sincronizados a cada ciclo de execução. No item **Options** (menu **Tools**), é possível ao usuário decidir se serão mostrados ambos alternadamente, só o Scan Time ou só o Sync Time.
- Device: mostra de qual equipamento está sendo requisitado o scan time. Ex. Device Model DF62 - SN # 100
- Ladder Status: informa se a ladder está sendo executada (**Running**), se está parada (**Stopped**) ou se está em **Standby**. Este último status indica que a CPU é a secundária do par redundante.
- Ladder: indica qual ladder está sendo visualizada.
- Cell: indica em qual célula o cursor está.

NOTAS

- As informações sobre o Scan Time e sobre a execução da ladder podem ser obtidas diretamente na barra de **Status**.
- Se o equipamento desejado não for encontrado ao se conectar tanto o **Scan time** e o **Ladder Status**, na Barra de **Status**, ficarão **N/A** (*Not available*).

EXEMPLO DE UTILIZAÇÃO DO LOGICVIEW FOR FFB

Descrição do processo

A figura a seguir é um exemplo de uma parte de um sistema de alarme de incêndio usando a lógica Ladder. A área **Fire Area** é monitorada por 3 detectores de fumaça – **SENSOR1**, **SENSOR2** e **SENSOR3**. Existe uma chave **SW1** que pode ser usada para ativar o alarme manualmente.

Certos detectores de fumaça podem gerar sinais que indicam fogo de forma errônea. Para prevenir falsos alarmes o sistema só acionará o alarme se dois ou mais detectores forem acionados. Isto é possível fazendo a combinação dos detectores segundo uma lógica simples. Se dois detectores estão ligados o alarme será acionado. Este é representado pelo bloco funcional Alarm_SR.

O alarme só poderá ser desativado pela chave **Clear_Alarm**. Se qualquer um dos detectores estiver ligado um LED acenderá na sala de controle como forma de alerta aos operadores.

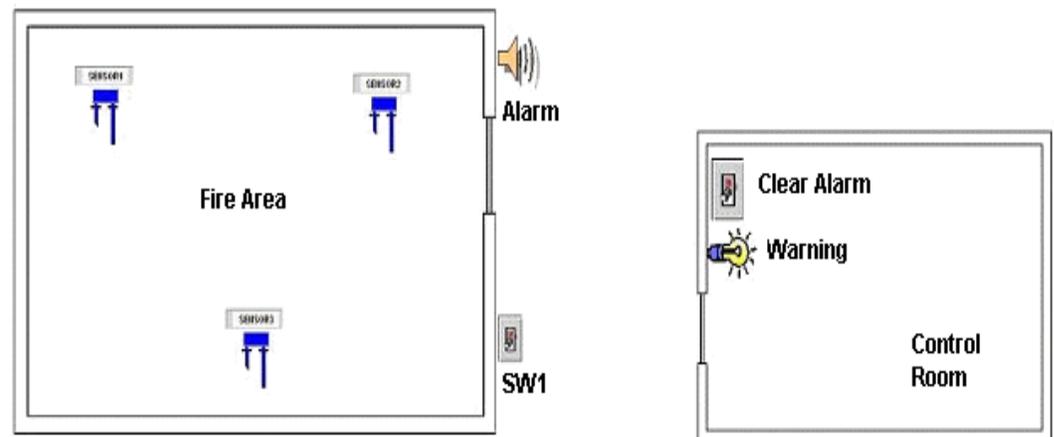


Fig 4. 1 – Sistema de Alarme de Incêndio

Implementando a lógica

Iniciar o System302 e na tela do Studio302, escolher o ícone do LogicView



Uma janela vai abrir e o usuário deverá escolher **New FFB Logic Template**. O LogicView for FFB será iniciado no modo template.

Crie um projeto novo, dê-lhe um nome e salve.

Se desejar preencha os dados do projeto clicando em **Smar LogicView**, na janela **Hierarchy**, e na janela **Object Properties** preencha os campos. Este passo não é obrigatório e pode ser feito a qualquer momento.

Configurando o Hardware

Clicar com o botão direito em **Hardware**, na janela **Hierarchy** e em seguida em **Hardware Configuration**. A seguinte janela irá aparecer:

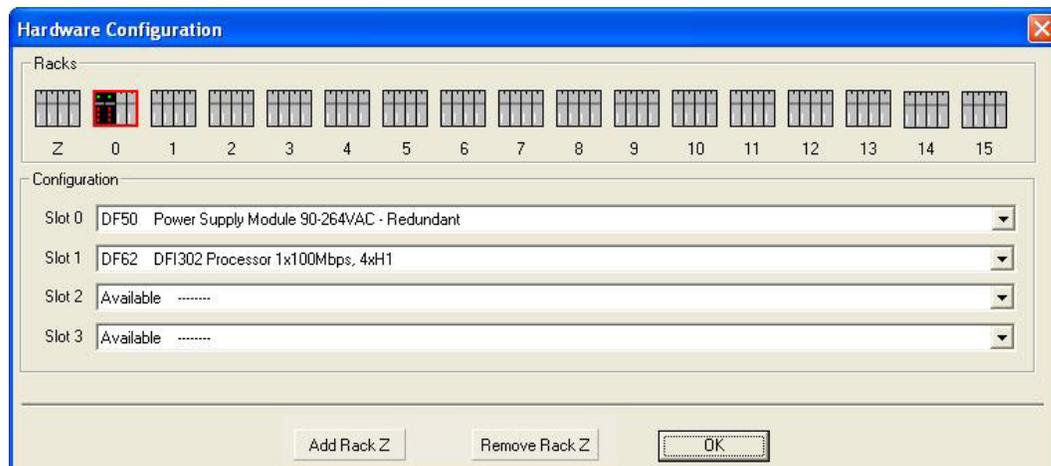


Fig 4. 2 – Configuração do Hardware (1)

No rack 0 já estarão o DF50 e o DF62 que são, respectivamente, a fonte de alimentação e o controlador. No slot 2 coloque o DF11, onde serão ligados os sensores, e no slot 3 coloque o DF20, onde serão ligadas as chaves SW1 e Clear_Alarm.

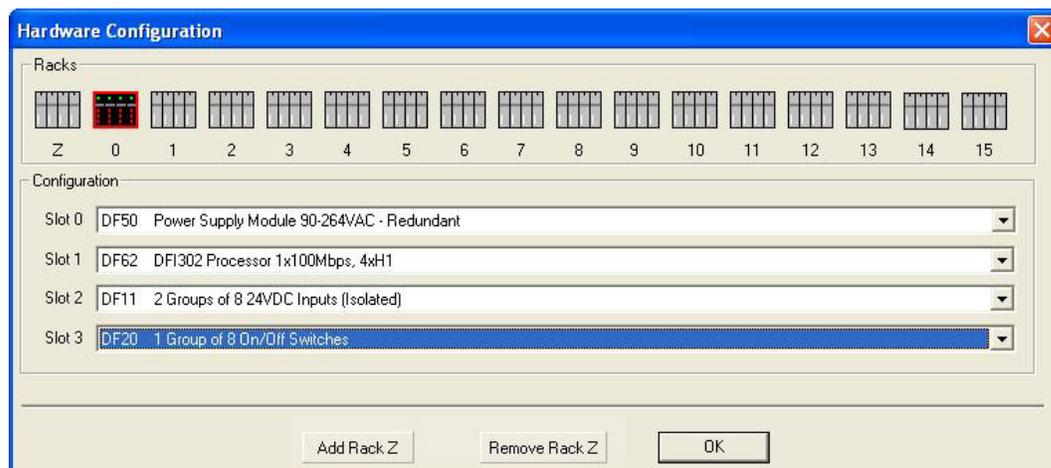


Fig 4. 3 – Configuração do Hardware (2)

Clique no rack 1, slot 0 e coloque o DF24, onde serão ligados a sirene do alarme e o LED de *warning*. Clique **OK**. Agora o hardware está configurado, o próximo passo é desenhar a lógica ladder.

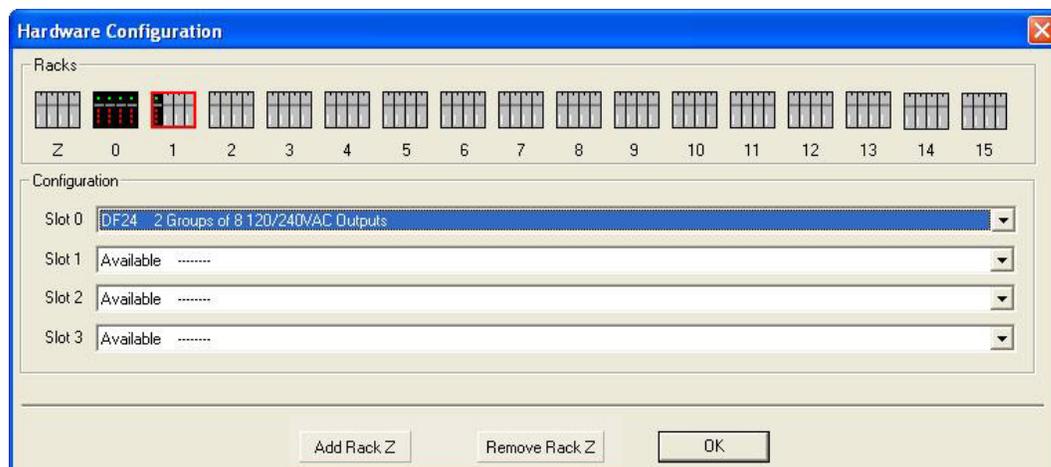


Fig 4. 4 – Configuração do Hardware (3)

Desenhando a Ladder

Insira os elementos na área de desenho da ladder. Refira-se ao tópico “Toolbox” para saber como é o procedimento de inserção dos elementos na ladder.

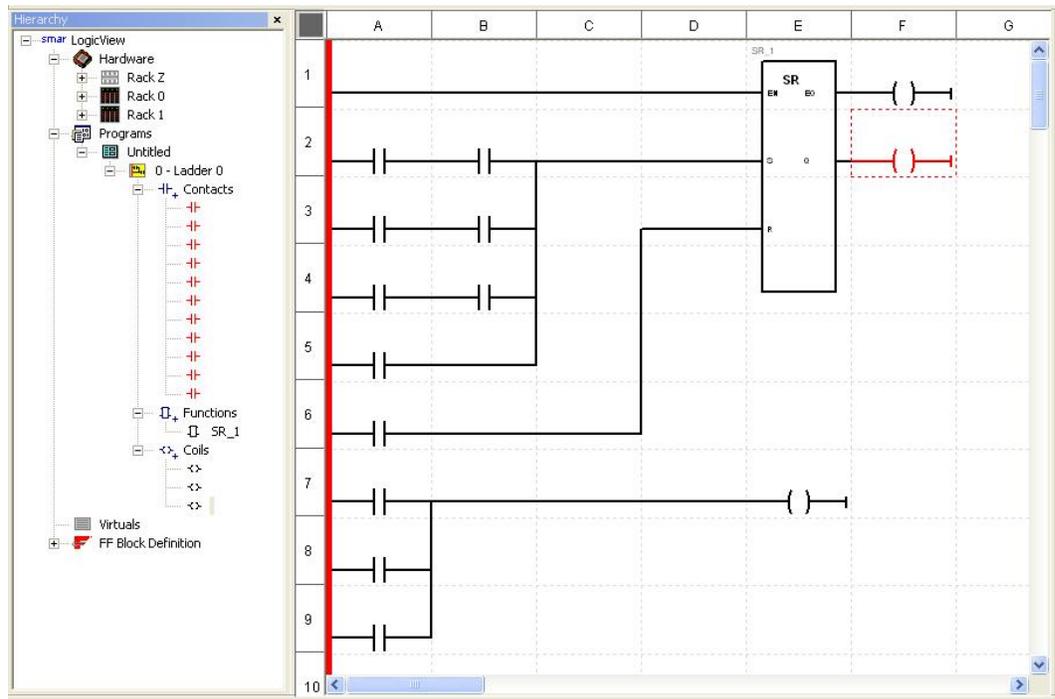


Fig 4.5 – Desenhando a Lógica Ladder

Note que a medida que os elementos são inseridos eles aparecem na janela **Hierarchy**.

O próximo passo é definir os tags e selecionar os parâmetros. Para melhor visualização e compreensão da ladder, edite os tags dos contatos e bobinas. Para isto vá ao menu **Tools** e escolha a opção **Properties editor**. A seguinte janela aparecerá:

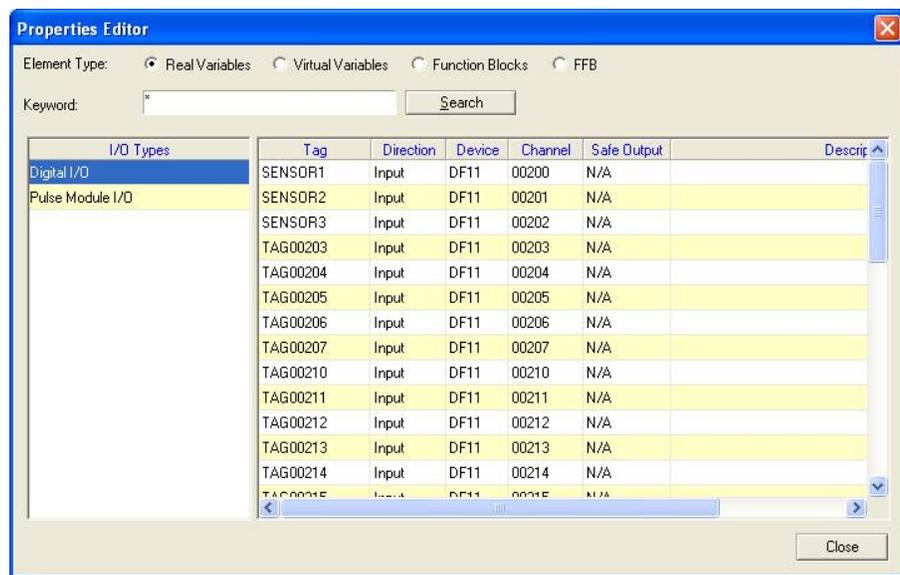


Fig 4.6 – Editando os Tags

Dê um duplo clique em TAG00200 e o modo de edição será habilitado. Escreva SENSOR1. Repita o procedimento para os seguintes tags:

TAG00201 - SENSOR2
 TAG00202 - SENSOR3
 TAG00300 – SW1
 TAG00301 – CLEAR_ALARM
 TAG01000 – ALARM
 TAG01001 – FIRE_WARNING
 V0000 – EO

Agora é preciso associar os elementos do hardware, e seus respectivos tags, aos elementos da ladder. Para isso selecione o elemento, clique com botão direito e depois em **Select parameter**. Escolha os parâmetros conforme a figura abaixo.

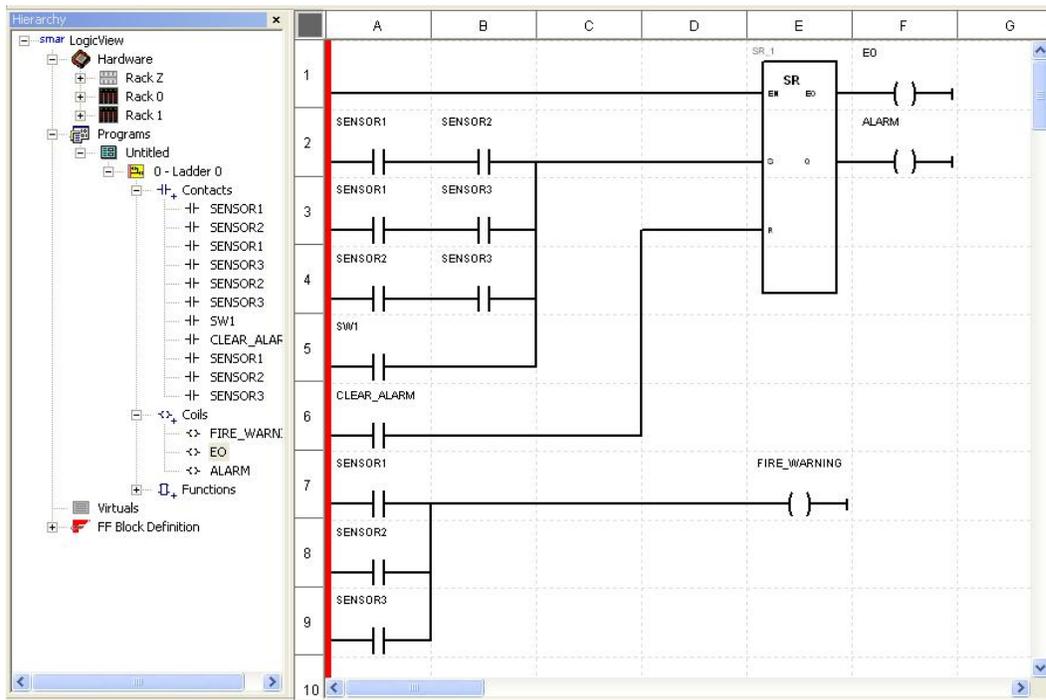


Fig 4. 7 – Escolhendo os Parâmetros

Salve o arquivo e depois execute o **Build** clicando em .

O usuário pode clicar no item desejado na janela **Hierarchy** ou no elemento na área de desenho da ladder e na janela **Object Properties** aparecerão suas propriedades.

Funcionamento do bloco funcional SR

Este bloco funcional, quando a entrada **EN** está verdadeira, funciona da seguinte forma:

Se a entrada **S** for verdadeira, a saída **Q** irá para verdadeiro. Se a entrada **R** for verdadeira, **Q** irá para falso. Se as duas entradas forem verdadeiras, **Q** se mantém em verdadeiro. Se a entrada **EN** está falsa, todas as saídas são mantidas em zero.

Simulação do alarme com a opção Simulation

Clique no botão **Simulation** , acione o SENSOR1, clique em **Run**.

Supondo que o SENSOR 1 detecte fumaça, o alarme não será acionado, apenas a lâmpada de alerta representada na lógica ladder pelo contato FIRE_WARNING é que será acionada. Veja a figura seguinte.

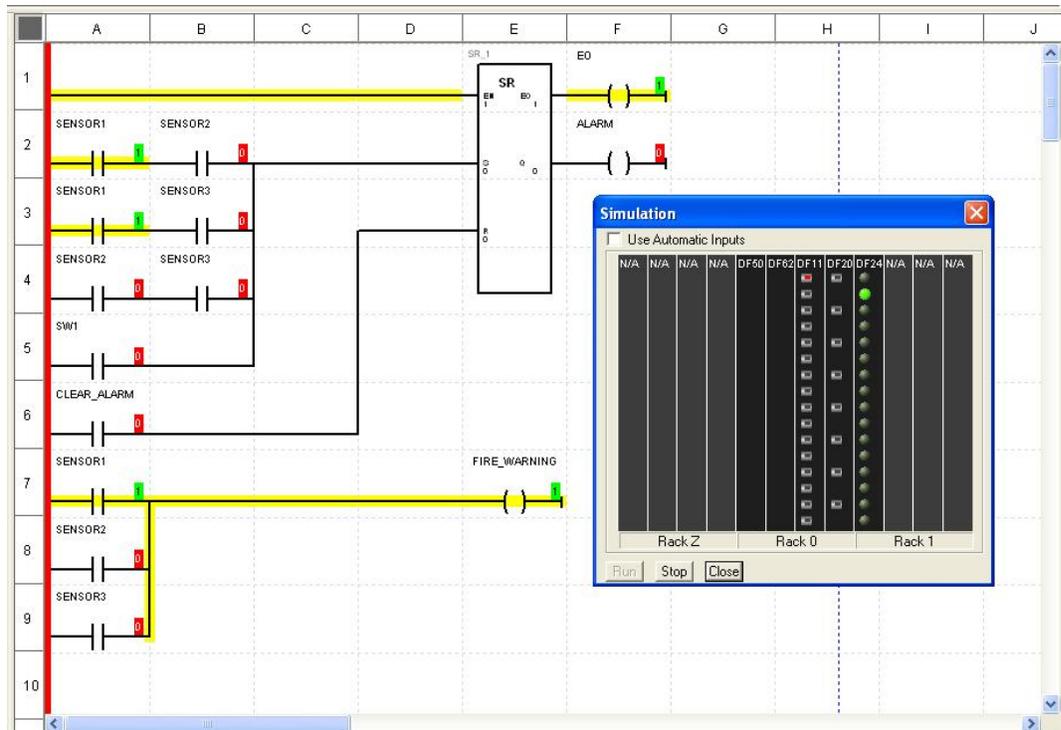


Fig 4.8 – Simulação – Sensor 1 Ativado

Supondo que o SENSOR3 também detecte a fumaça. Acione o SENSOR3 do rack 0 e veja que o alarme será acionado.

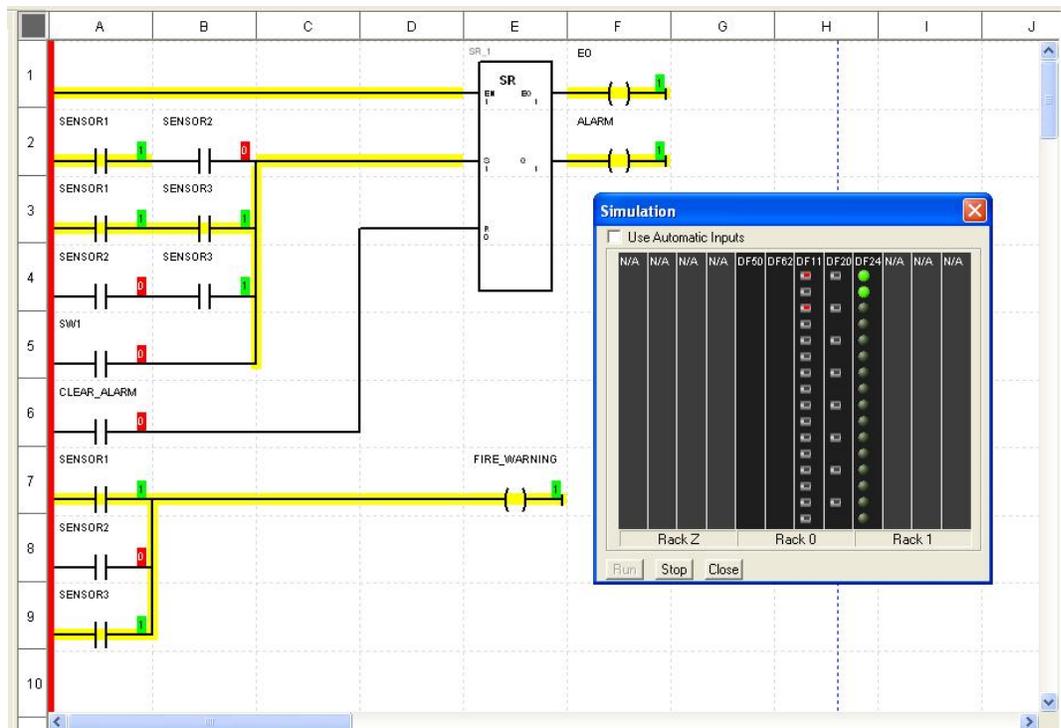


Fig 4.9 – Simulação – Sensores 1 e 3 Ativados

O alarme continuará acionado mesmo após os sensores não detectarem mais a presença de fumaça. O alarme só será desativado se a chave manual CLEAR_ALARM for acionada.

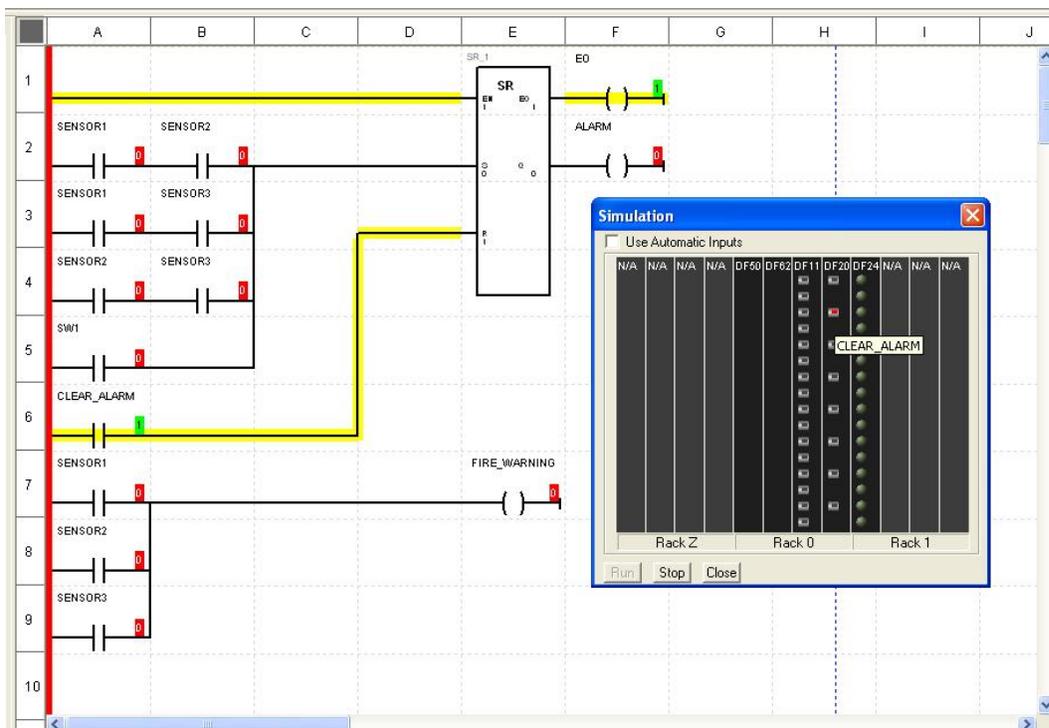


Fig 4. 10 – Simulação – Acionando o Clear_Alarm

O alarme poderá ser acionado manualmente através da chave SW1. Observe que a entrada S tem prioridade sobre a entrada R.

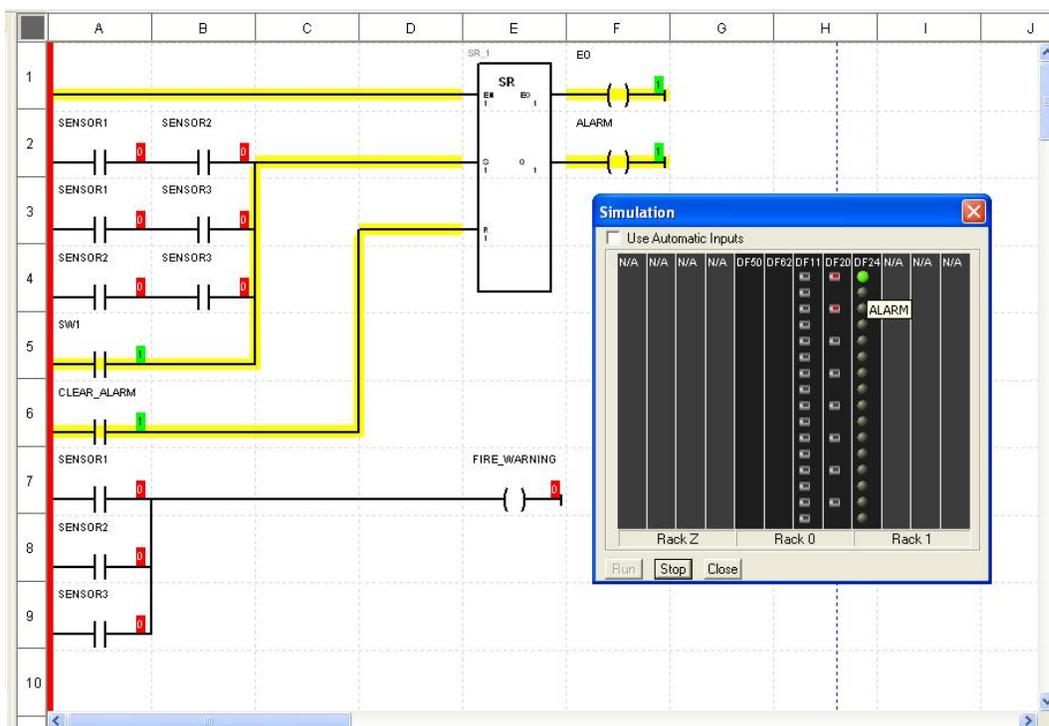


Fig 4. 11 – Simulação – Acionando o Alarme com a Chave SW1

Para finalizar clique em **Close**.