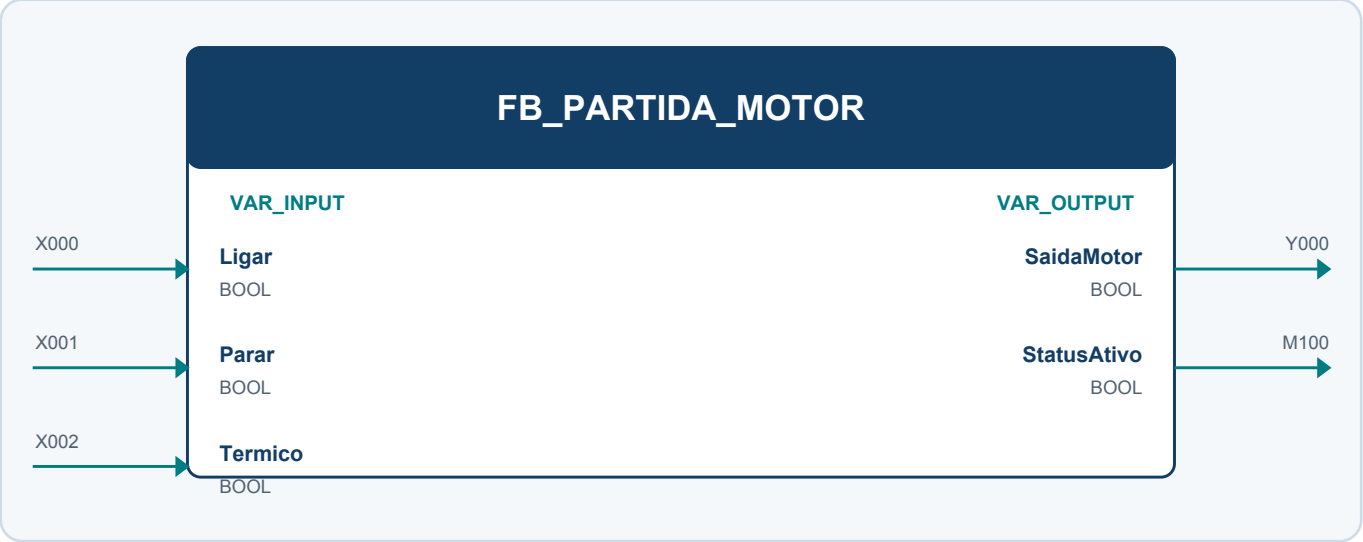
COOLMAY PLC STUDIO

Manual ilustrado dos blocos

43 interfaces reais. Pinos, lógica, exemplos e ensaios.



Edição técnica da IDE | Português (Brasil)

Os diagramas são interfaces de software reconstruídas das definições atuais, não desenhos de instalação elétrica. Os exemplos são guias de ensaio e não resultados de operação em CLP.

Como usar este manual

1. Abra +FB / FB_Pool, escolha o bloco e consulte entradas, saídas e a lógica IL.

2. Insira a instância no programa; use um nome único e associe cada pino a uma tag ou dispositivo compatível. Os endereços padrão são exemplos, não alocação exclusiva.

3. Defina VAR_INPUT e VAR_OUTPUT como interface lógica; associe os pinos físicos no programa chamador. VAR contém as tags internas e sua retenção. Instâncias que compartilham D/M/T/C também compartilham estado.

4. Use Inserir e Compilar; confira o IL expandido, ordem das redes e todos os endereços escritos. Para atualizar uma instância existente, use seu editor de pinos e confira a fonte resultante.

5. Simule casos normal, desabilitado, reinício e limites. Para comparar inteiros, teste menor/igual/maior. Para seletores, teste ambas as posições. Para memória SET/RST, teste comandos simultâneos.

6. Para uso físico, confira o modelo de CPU, o módulo, a polaridade e os limites no manual correspondente. Simulação e compilação não comprovam comunicação ou execução no CLP.

ATENÇÃO: o nome IEC de um bloco não garante semântica IEC. O IL mostrado é a implementação atual. Corpos ST e IL podem divergir; vários corpos ST são apenas comentários. Não os considere equivalentes sem compilar e testar.

Reserve a extensão dos operandos: DINT ocupa dois D; MUL/DIV podem escrever pares D; CMP/ZCP escrevem três bits a partir do destino; TRD usa sete D; buffers de comunicação e parâmetros PID exigem reserva própria. Evite divisão por zero e sobreposição entre instâncias.

Blocos de intertravamento são exemplos de lógica de processo e não funções de segurança certificadas.

Leitura dos diagramas

À esquerda: VAR_INPUT. À direita: VAR_OUTPUT. O nome dentro do bloco é o pino lógico; o tipo fica abaixo. O endereço fora do bloco é apenas a associação padrão da biblioteca. Um pino de saída pode representar uma área de trabalho, um temporizador, um contador ou uma saída física; consulte a tabela, não suponha que todo VAR_OUTPUT é Y.

Índice dos 43 blocos

01 FB_PARTIDA_MOTOR - Partida direta com selo, parada e proteção térmica

02 COMPARADOR_INT - Comparador inteiro estruturado (ATIVAR, S1_W, S2_W -> SAIDA_Y)

03 CALCULO99 - Cálculo proporcional percentual ((VALOR_ENTRADA * PORCENTAGEM) / 100)

04 SOMA_PESOS - Totalizador de balança / dosagem (SAL + FOSFATO + CALCARIO -> PESO_TOTAL)

05 FB_PULSO_EIXO - Acionamento relativo de eixo de pulso 16-bit (DRVI estruturado)

06 FB_POS_ABSOLUTO_DRVA - Posicionamento absoluto de eixo 16-bit (DRVA estruturado)

07 FB_POS_RELATIVO_32B - Posicionamento relativo 32-bit de alta resolução (DDRVI estruturado)

08 FB_POS_ABSOLUTO_32B - Posicionamento absoluto 32-bit para servo/mesa linear (DDRVA estruturado)

09 FB_HOME_ORIGEM_ZRN - Retorno à origem (Homing 16-bit ZRN) com busca rápida, creep e sensor Dog

10 FB_HOME_ORIGEM_DZRN - Retorno à origem 32-bit de alta velocidade (DZRN estruturado)

11 FB_JOG_VELOCIDADE_PLSV - Rotação contínua JOG / velocidade variável com direção (PLSV estruturado)

12 FB_TREM_PULSOS_PLSY - Gerador de trem de pulsos com frequência e contagem fixas (PLSY 16-bit)

13 FB_TREM_PULSOS_DPLSY - Gerador de trem de pulsos 32-bit de longo curso (DPLSY estruturado)

14 FB_INTERTRAV_EIXO - Supervisão de limites de curso (+/-), Servo Ready e parada de eixo

15 FB_TON_DELAY - Temporizador de retardo na energização base 100ms (TON / TIMER_100_FB_M)

16 FB_TOF_OFF_DELAY - Temporizador de retardo na desenergização (TOF estruturado)

17 FB_TP_PULSO - Temporizador de pulso monostável de largura fixa (TP IEC 61131-3)

18 TIMER_CONT_FB_M - Oscilador cíclico / Pisca-pisca contínuo (Blink / Clock Generator)

19 FB_CTU_CONTADOR - Contador incremental estruturado (CTU / COUNTER_FB_M) com entrada de Reset

20 FB_CTD_DECREMENTAL - Contador decrescente por registrador (CTD IEC 61131-3)

21 FB_CTUD_BIDIRECIONAL - Contador bidirecional Up/Down em registrador (CTUD IEC 61131-3)

22 FB_SR_SET_DOMINANTE - Flip-Flop biestável SR com prioridade para Set (IEC 61131-3 SR)

23 FB_RS_RESET_DOMINANTE - Flip-Flop biestável RS com prioridade para Reset (IEC 61131-3 RS)

24 FB_R_TRIG_BORDA_SUB - Detector de borda de subida de 1 scan (R_TRIG IEC 61131-3)

25 FB_F_TRIG_BORDA_DESC - Detector de borda de descida de 1 scan (F_TRIG IEC 61131-3)

26 FB_SEL_SELETOR - Seletor binário de 2 entradas inteiras (IEC 61131-3 SEL: G=0 -> IN0, G=1 -> IN1)

27 FB_LIMIT_FAIXA - Limitador de valor mínimo e máximo (IEC 61131-3 LIMIT: MN <= IN <= MX)

28 FB_MAX_VALOR - Seleciona o maior valor entre duas entradas (IEC 61131-3 MAX)

29 FB_MIN_VALOR - Seleciona o menor valor entre duas entradas (IEC 61131-3 MIN)

30 FB_HISTERESE_ZONA - Controle On/Off com banda morta / histerese (liga abaixo de Low, desliga acima de High)

31 FB_REGRA_DE_TRES - Conversão proporcional por regra de três ((Entrada * EscalaMax) / BaseMax)

32 FB_MEDIA_2_CANAIS - Cálculo de média aritmética entre dois sensores ((CH1 + CH2) / 2)

33 FB_COPIA_RECEITA_3W - Carrega 3 parâmetros de receita para os registradores de trabalho

34 FB_CLEAR_FAIXA_BITS - Reset geral em zona de bits/alarmes (ZRST estruturado)

35 FB_ESCALA_ANALOG - Escalonamento linear de entrada analógica AD (Divisão + Offset)

36 FB_DESESCALA_DA - Converte valor de engenharia (0..100%) para saída analógica DA (0..4000)

37 FB_PID_CONTROLE - Malha fechada de controle PID (SV / PV / Parâmetros -> MV)

38 FB_ALARME_ANALOG_HH_LL - Supervisão de limites analógicos Alto (High) e Baixo (Low) via ZCP

39 FB_INVERSAO_MOTOR - Partida reversora Avanço/Recuo com intertravamento elétrico cruzado

40 FB_VALVULA_SOLENOIDE - Controle de válvula solenoide com confirmação de sensor aberto/fechado

41 FB_MODBUS_READ_H03 - Mestre Modbus RTU Função 03 — Leitura de Registradores Holding (ADPRW)

42 FB_MODBUS_WRITE_H06 - Mestre Modbus RTU Função 06 — Escrita de Registrador Único (ADPRW)

43 FB_RTC_LEITURA_RELOGIO - Leitura do Relógio de Tempo Real (TRD D8013..D8019) para bloco de registradores

01 FB_PARTIDA_MOTOR

Partida direta com selo, parada e proteção térmica

8.1 Partida e Reversão de Motores (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Ligar	BOOL	X000	Botão de partida NA
IN	Parar	BOOL	X001	Botão de parada NF
IN	Termico	BOOL	X002	Relé térmico de sobrecarga
OUT	SaidaMotor	BOOL	Y000	Contator principal do motor
OUT	StatusAtivo	BOOL	M100	Bit auxiliar de motor ligado

Associar na IDE

Crie a instância Inst_FB_PARTIDA_MOTOR. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

01 FB_PARTIDA_MOTOR

2. Funcionamento, instrução por instrução

01	LD {Ligar}	Inicia uma condição lendo {Ligar}.
02	OR {SaidaMotor}	Combina em paralelo: condição anterior OR {SaidaMotor}.
03	ANI {Parar}	Exige também {Parar} falso; confira a polaridade do sinal real.
04	ANI {Termico}	Exige também {Termico} falso; confira a polaridade do sinal real.
05	OUT {SaidaMotor}	Escreve a condição atual em {SaidaMotor} a cada scan.
06	OUT {StatusAtivo}	Escreve a condição atual em {StatusAtivo} a cada scan.

Exemplo IL com as associações padrão

```
LD X000
OR Y000
ANI X001
ANI X002
OUT Y000
OUT M100
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

01 FB_PARTIDA_MOTOR

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Ligar=1; Parar=0; Termico=0	SaidaMotor=1; StatusAtivo=1
Ligar=0; Parar=0; Termico=0; SaidaMotor(t-1)=1	SaidaMotor=1; StatusAtivo=1
Parar=1; Termico=0	SaidaMotor=0; StatusAtivo=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

ANI exige bit 0 para liberar. Um botão NF pode produzir bit 1 em repouso dependendo da ligação: normalize o sinal e teste cabo rompido e comando simultâneo. A lógica apresentada não substitui circuito de segurança certificado.

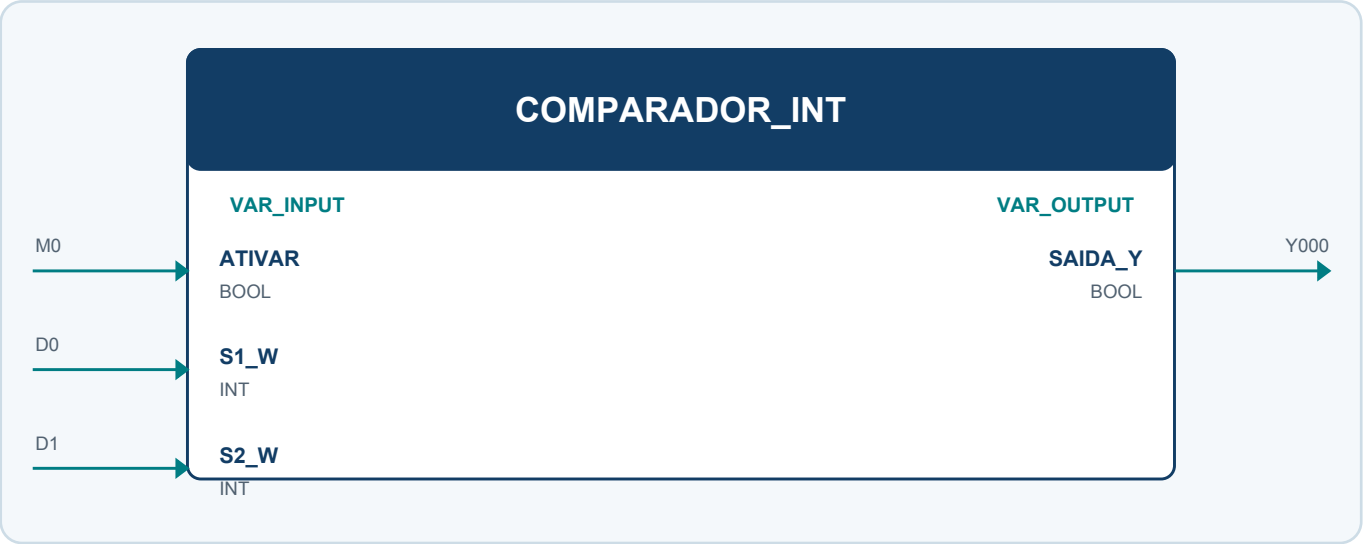
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

02 COMPARADOR_INT

Comparador inteiro estruturado (ATIVAR, S1_W, S2_W -> SAIDA_Y)

4.5 Blocos de Comparação e Seleção IEC (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	ATIVAR	BOOL	M0	Habilita comparador
IN	S1_W	INT	D0	Valor inteiro 1 (Word[Signed])
IN	S2_W	INT	D1	Valor inteiro 2 (Word[Signed])
OUT	SAIDA_Y	BOOL	Y000	Saída verdadeira quando S1_W = S2_W

Associar na IDE

Crie a instância Inst_COMPARADOR_INT. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

02 COMPARADOR_INT

2. Funcionamento, instrução por instrução

01	LD {ATIVAR}	Inicia uma condição lendo {ATIVAR}.
02	AND= {S1_W} {S2_W}	Exige também {S1_W} = {S2_W}.
03	OUT {SAIDA_Y}	Escreve a condição atual em {SAIDA_Y} a cada scan.

Exemplo IL com as associações padrão

```
LD M0
AND= D0 D1
OUT Y000
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

02 COMPARADOR_INT

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
ATIVAR=1; S1_W=10; S2_W=10	SAIDA_Y=1
ATIVAR=1; S1_W=9; S2_W=10	SAIDA_Y=0
ATIVAR=0; S1_W=10; S2_W=10	SAIDA_Y=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

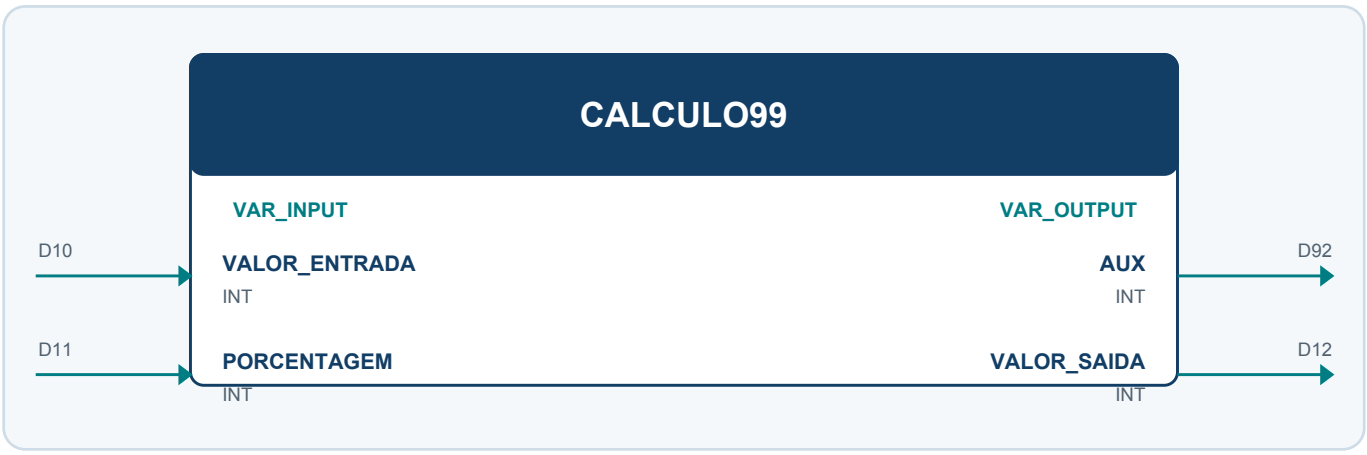
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

03 CALCULO99

Cálculo proporcional percentual ((VALOR_ENTRADA * PORCENTAGEM) / 100)

5.3 Blocos Matemáticos e Totalizadores (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	VALOR_ENTRADA	INT	D10	Valor base de entrada
IN	PORCENTAGEM	INT	D11	Fator percentual (0..100)
OUT	AUX	INT	D92	Acumulador auxiliar
OUT	VALOR_SAIDA	INT	D12	Resultado calculado

Associar na IDE

Crie a instância Inst_CALCULO99. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

03 CALCULO99

2. Funcionamento, instrução por instrução

01	LD M8000	Inicia uma condição lendo M8000.
02	MUL {VALOR_ENTRADA} {PORCENTAGEM} {AUX}	Se habilitado, multiplica {VALOR_ENTRADA} por {PORCENTAGEM} e grava o par iniciado em {AUX}.
03	DIV {AUX} K100 {VALOR_SAIDA}	Se habilitado, divide {AUX} por K100: quociente em {VALOR_SAIDA} e resto no D seguinte. Divisor zero é inválido.

Exemplo IL com as associações padrão

```
LD M8000
MUL D10 D11 D92
DIV D92 K100 D12
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

03 CALCULO99

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
VALOR_ENTRADA=100; PORCENTAGEM=25	IL: AUX=2500; VALOR_SAIDA=25
VALOR_ENTRADA=200; PORCENTAGEM=10	IL: AUX=2000; VALOR_SAIDA=20
VALOR_ENTRADA=0; PORCENTAGEM=100	AUX=0; VALOR_SAIDA=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

O produto de MUL ocupa dois D, mas DIV sem prefixo D usa o operando de 16 bits. Os exemplos pequenos evitam truncamento; não extrapole para toda a escala. Para produtos fora de INT16, revise a sequência com operações de 32 bits.

Reserve também o D seguinte a cada resultado de MUL/DIV. Um resultado dirigido a DA pode escrever o canal analógico adjacente; confira o mapa antes de usar saídas reais.

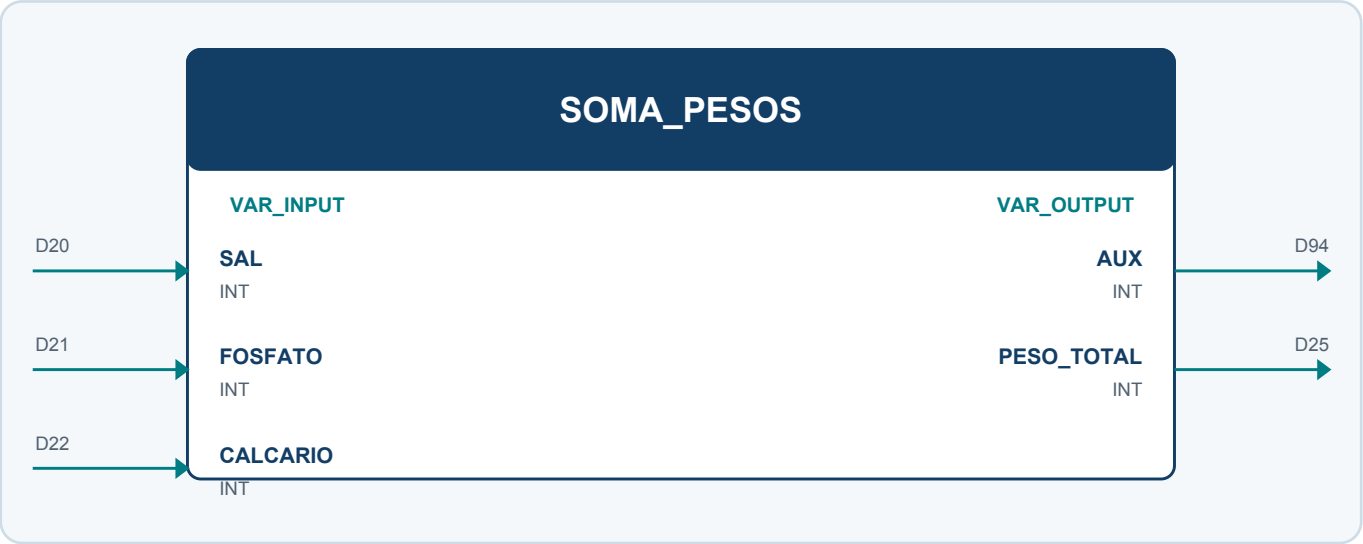
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

04 SOMA_PESOS

Totalizador de balança / dosagem (SAL + FOSFATO + CALCARIO -> PESO_TOTAL)

5.3 Blocos Matemáticos e Totalizadores (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	SAL	INT	D20	Peso dosado Sal
IN	FOSFATO	INT	D21	Peso dosado Fosfato
IN	CALCARIO	INT	D22	Peso dosado Calcário
OUT	AUX	INT	D94	Soma parcial
OUT	PESO_TOTAL	INT	D25	Peso total acumulado

Associar na IDE

Crie a instância Inst_SOMA_PESOS. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

04 SOMA_PESOS

2. Funcionamento, instrução por instrução

01	LD M8000	Inicia uma condição lendo M8000.
02	ADD {SAL} {FOSFATO} {AUX}	Se habilitado, calcula {SAL} + {FOSFATO} em {AUX}; confira overflow de 16 bits.
03	ADD {AUX} {CALCARIO} {PESO_TOTAL}	Se habilitado, calcula {AUX} + {CALCARIO} em {PESO_TOTAL}; confira overflow de 16 bits.

Exemplo IL com as associações padrão

```
LD M8000
ADD D20 D21 D94
ADD D94 D22 D25
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

04 SOMA_PESOS

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
SAL=10; FOSFATO=20; CALCARIO=30	AUX=30; PESO_TOTAL=60
SAL=0; FOSFATO=0; CALCARIO=0	AUX=0; PESO_TOTAL=0
SAL=-10; FOSFATO=20; CALCARIO=30	AUX=10; PESO_TOTAL=40

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

05 FB_PULSO_EIXO

Acionamento relativo de eixo de pulso 16-bit (DRVI estruturado)

2.4 Blocos de Função de Posicionamento e Motion (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Execute	BOOL	M0	Comando de disparo
IN	Pulsos	INT	K8000	Distância relativa em pulsos
IN	Frequencia	INT	K4000	Frequência de pulso em Hz
OUT	SaidaPulso	BOOL	Y000	Saída rápida de pulso (Y000..Y007)
OUT	SaidaDir	BOOL	Y004	Saída de sinal de direção

Associar na IDE

Crie a instância Inst_FB_PULSO_EIXO. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

05 FB_PULSO_EIXO

2. Funcionamento, instrução por instrução

01	LD {Execute}	Inicia uma condição lendo {Execute}.
02	DRVI {Pulsos} {Frequencia} {SaidaPulso} {SaidaDir}	Se habilitado, emite DRVI com operandos na ordem mostrada. Valide saída, direção, sinal e frequência no manual da CPU e em bancada.

Exemplo IL com as associações padrão

```
LD M0
DRVI K8000 K4000 Y000 Y004
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

05 FB_PULSO_EIXO

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Execute=0	DRVI: 0
Execute=1; Pulsos=100; Frequencia=100	DRVI(Pulsos,Frequencia,SaidaPulso,SaidaDir)
Execute=1; Pulsos=-100; Frequencia=100	DRVI(-100,100,SaidaPulso,SaidaDir)

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

A tabela de ensaio descreve o comando emitido, não uma posição ou frequência já medida. Use carga desacoplada, limites e instrumentação no comissionamento. Saídas permitidas, frequência, reaplicação do comando e parada dependem da CPU.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

06 FB_POS_ABSOLUTO_DRVA

Posicionamento absoluto de eixo 16-bit (DRVA estruturado)

2.4 Blocos de Função de Posicionamento e Motion (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Execute	BOOL	M1	Disparo do posicionamento absoluto
IN	PosicaoAlvo	INT	K15000	Coordenada absoluta alvo (pulsos)
IN	VelocidadeHz	INT	K5000	Frequência de deslocamento (Hz)
OUT	EixoPulso	BOOL	Y000	Saída de pulso do eixo
OUT	EixoDirecao	BOOL	Y004	Saída de direção do eixo

Associar na IDE

Crie a instância Inst_FB_POS_ABSOLUTO_DRVA. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

06 FB_POS_ABSOLUTO_DRVA

2. Funcionamento, instrução por instrução

01	LD {Execute}	Inicia uma condição lendo {Execute}.
02	DRVA {PosicaoAlvo} {VelocidadeHz} {EixoPulso} {EixoDirecao}	Se habilitado, emite DRVA com operandos na ordem mostrada. Valide saída, direção, sinal e frequência no manual da CPU e em bancada.

Exemplo IL com as associações padrão

```
LD M1
DRVA K15000 K5000 Y000 Y004
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

06 FB_POS_ABSOLUTO_DRVA

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Execute=0	DRVA: 0
Execute=1; PosicaoAlvo=100; VelocidadeHz=100	DRVA(100,100,EixoPulso,EixoDirecao)
Execute=1; PosicaoAlvo=0; VelocidadeHz=100	DRVA(0,100,EixoPulso,EixoDirecao)

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

A tabela de ensaio descreve o comando emitido, não uma posição ou frequência já medida. Use carga desacoplada, limites e instrumentação no comissionamento. Saídas permitidas, frequência, reaplicação do comando e parada dependem da CPU.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

07 FB_POS_RELATIVO_32B

Posicionamento relativo 32-bit de alta resolução (DDRVI estruturado)

2.4 Blocos de Função de Posicionamento e Motion (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Execute	BOOL	M2	Disparo do movimento relativo 32-bit
IN	Pulsos32b	DINT	K25000	Incremento de posição 32-bit
IN	Freq32b	DINT	K12000	Frequência em Hz (32-bit)
OUT	SaidaPulso	BOOL	Y000	Saída rápida de pulso
OUT	SaidaDir	BOOL	Y004	Saída de direção

Associar na IDE

Crie a instância Inst_FB_POS_RELATIVO_32B. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

07 FB_POS_RELATIVO_32B

2. Funcionamento, instrução por instrução

01	LD {Execute}	Inicia uma condição lendo {Execute}.
02	DDRVI {Pulsos32b} {Freq32b} {SaidaPulso} {SaidaDir}	Se habilitado, emite DDRVI com operandos na ordem mostrada. Valide saída, direção, sinal e frequência no manual da CPU e em bancada.

Exemplo IL com as associações padrão

```
LD M2
DDRVI K25000 K12000 Y000 Y004
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

07 FB_POS_RELATIVO_32B

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Execute=0	DDRVI: 0
Execute=1; Pulsos32b=100; Freq32b=100	DDRVI(Pulsos32b,Freq32b,SaidaPulso,SaidaDir)
Execute=1; Pulsos32b=-100; Freq32b=100	DDRVI(-100,100,SaidaPulso,SaidaDir)

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

A tabela de ensaio descreve o comando emitido, não uma posição ou frequência já medida. Use carga desacoplada, limites e instrumentação no comissionamento. Saídas permitidas, frequência, reaplicação do comando e parada dependem da CPU.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

08 FB_POS_ABSOLUTO_32B

Posicionamento absoluto 32-bit para servo/mesa linear (DDRVA estruturado)

2.4 Blocos de Função de Posicionamento e Motion (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Execute	BOOL	M3	Disparo do movimento absoluto 32-bit
IN	Alvo32b	DINT	K30000	Posição absoluta alvo 32-bit
IN	Freq32b	DINT	K15000	Frequência máxima em Hz
OUT	SaidaPulso	BOOL	Y001	Saída de pulso Eixo 2 (Y001)
OUT	SaidaDir	BOOL	Y005	Saída de direção Eixo 2 (Y005)

Associar na IDE

Crie a instância Inst_FB_POS_ABSOLUTO_32B. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

08 FB_POS_ABSOLUTO_32B

2. Funcionamento, instrução por instrução

01	LD {Execute}	Inicia uma condição lendo {Execute}.
02	DDRVA {Alvo32b} {Freq32b} {SaidaPulso} {SaidaDir}	Se habilitado, emite DDRVA com operandos na ordem mostrada. Valide saída, direção, sinal e frequência no manual da CPU e em bancada.

Exemplo IL com as associações padrão

```
LD M3
DDRVA K30000 K15000 Y001 Y005
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

08 FB_POS_ABSOLUTO_32B

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Execute=0	DDRVA: 0
Execute=1; Alvo32b=100; Freq32b=100	DDRVA(Alvo32b,Freq32b,SaidaPulso,SaidaDir)
Execute=1; Alvo32b=0; Freq32b=100	DDRVA(0,100,SaidaPulso,SaidaDir)

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

A tabela de ensaio descreve o comando emitido, não uma posição ou frequência já medida. Use carga desacoplada, limites e instrumentação no comissionamento. Saídas permitidas, frequência, reaplicação do comando e parada dependem da CPU.

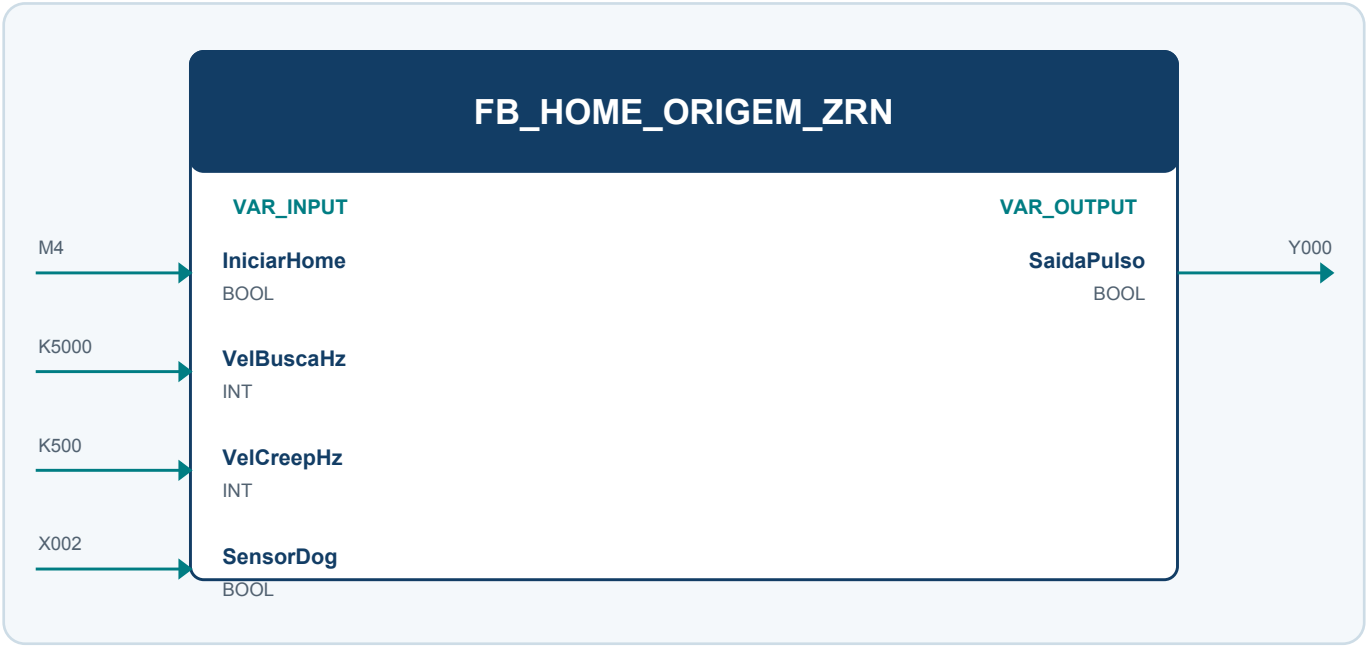
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

09 FB_HOME_ORIGEM_ZRN

Retorno à origem (Homing 16-bit ZRN) com busca rápida, creep e sensor Dog

2.4 Blocos de Função de Posicionamento e Motion (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	IniciarHome	BOOL	M4	Inicia busca de referência Home
IN	VelBuscaHz	INT	K5000	Velocidade rápida de busca (Hz)
IN	VelCreepHz	INT	K500	Velocidade lenta de aproximação Creep (Hz)
IN	SensorDog	BOOL	X002	Entrada física do sensor de origem (Near-Point Dog)
OUT	SaidaPulso	BOOL	Y000	Saída de pulso do eixo para Homing

Associar na IDE

Crie a instância Inst_FB_HOME_ORIGEM_ZRN. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

09 FB_HOME_ORIGEM_ZRN

2. Funcionamento, instrução por instrução

01	LD {IniciarHome}	Inicia uma condição lendo {IniciarHome}.
02	ZRN {VelBuscaHz} {VelCreepHz} {SensorDog} {SaidaPulso}	Se habilitado, emite ZRN com operandos na ordem mostrada. Valide saída, direção, sinal e frequência no manual da CPU e em bancada.

Exemplo IL com as associações padrão

```
LD M4
ZRN K5000 K500 X002 Y000
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

09 FB_HOME_ORIGEM_ZRN

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
IniciarHome=0	ZRN: 0
IniciarHome=1; SensorDog=0	ZRN(VelBuscaHz,VelCreepHz,SensorDog,SaidaPulso)
SensorDog: 0 -> 1 -> 0	ZRN: SensorDog; VelCreepHz

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

A tabela de ensaio descreve o comando emitido, não uma posição ou frequência já medida. Use carga desacoplada, limites e instrumentação no comissionamento. Saídas permitidas, frequência, reaplicação do comando e parada dependem da CPU.

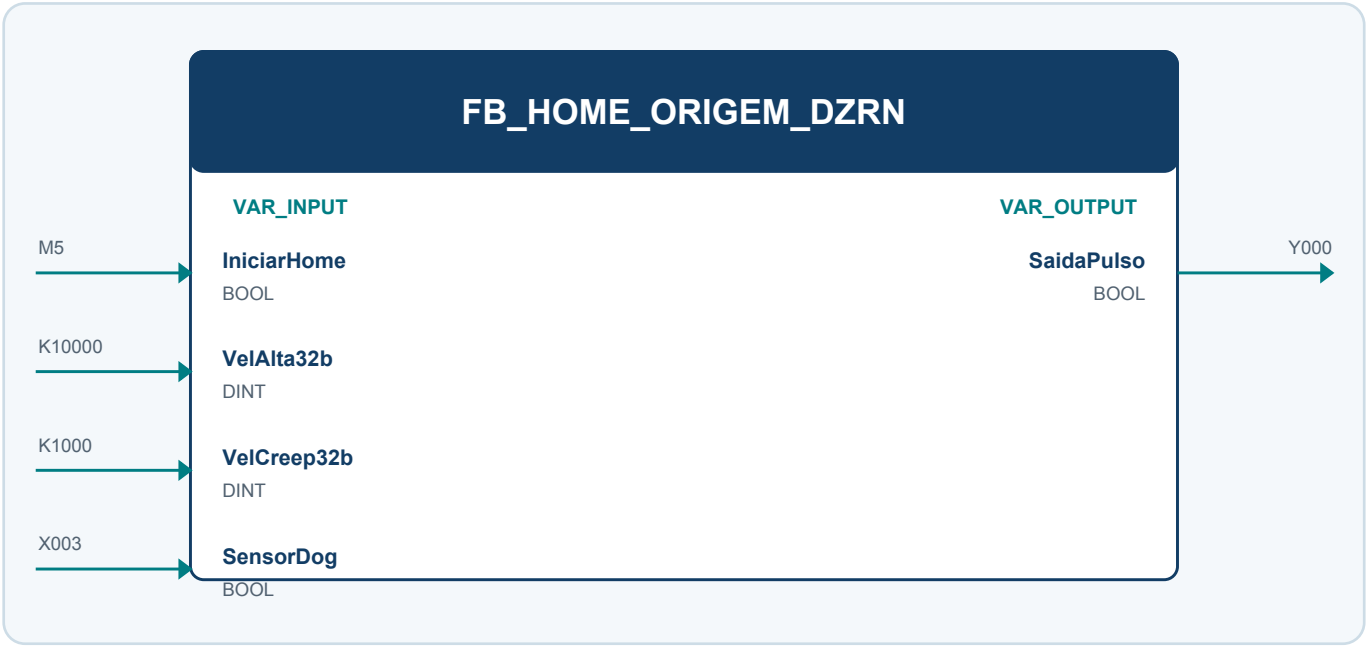
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

10 FB_HOME_ORIGEM_DZRN

Retorno à origem 32-bit de alta velocidade (DZRN estruturado)

2.4 Blocos de Função de Posicionamento e Motion (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	IniciarHome	BOOL	M5	Comando de Homing 32-bit
IN	VelAlta32b	DINT	K10000	Velocidade de busca 32-bit (Hz)
IN	VelCreep32b	DINT	K1000	Velocidade de aproximação Creep (Hz)
IN	SensorDog	BOOL	X003	Sensor de referência de origem
OUT	SaidaPulso	BOOL	Y000	Saída rápida de pulso

Associar na IDE

Crie a instância Inst_FB_HOME_ORIGEM_DZRN. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

10 FB_HOME_ORIGEM_DZRN

2. Funcionamento, instrução por instrução

01	LD {IniciarHome}	Inicia uma condição lendo {IniciarHome}.
02	DZRN {VelAlta32b} {VelCreep32b} {SensorDog} {SaidaPulso}	Se habilitado, emite DZRN com operandos na ordem mostrada. Valide saída, direção, sinal e frequência no manual da CPU e em bancada.

Exemplo IL com as associações padrão

```
LD M5
DZRN K10000 K1000 X003 Y000
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

10 FB_HOME_ORIGEM_DZRN

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
IniciarHome=0	DZRN: 0
IniciarHome=1; SensorDog=0	DZRN(VelAlta32b,VelCreep32b,SensorDog,SaidaPulso)
SensorDog: 0 -> 1 -> 0	DZRN: SensorDog; VelCreep32b

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

A tabela de ensaio descreve o comando emitido, não uma posição ou frequência já medida. Use carga desacoplada, limites e instrumentação no comissionamento. Saídas permitidas, frequência, reaplicação do comando e parada dependem da CPU.

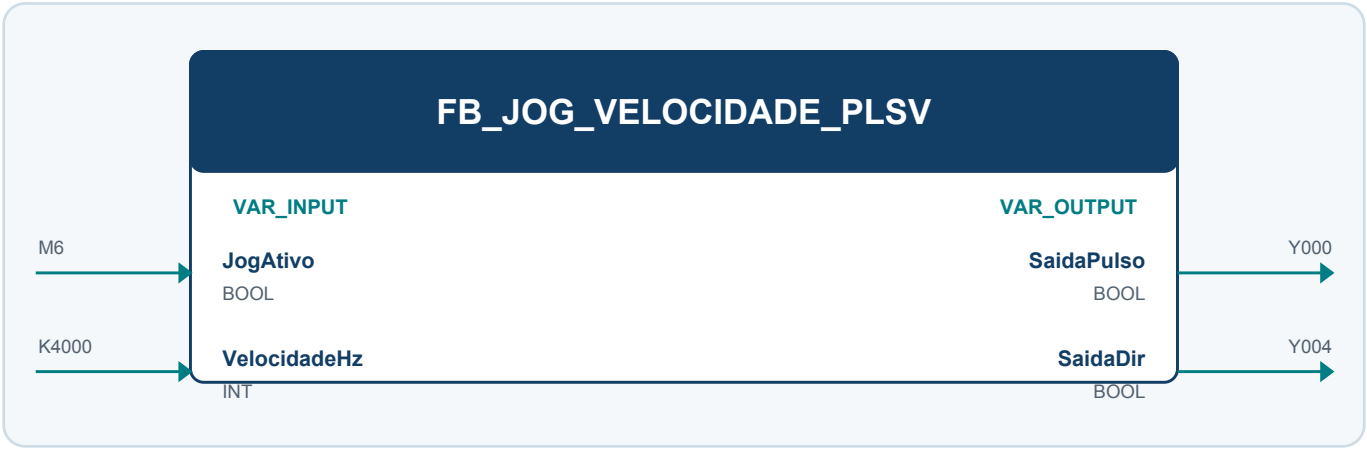
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

11 FB_JOG_VELOCIDADE_PLSV

Rotação contínua JOG / velocidade variável com direção (PLSV estruturado)

2.4 Blocos de Função de Posicionamento e Motion (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	JogAtivo	BOOL	M6	Enquanto ON, mantém eixo em movimento contínuo
IN	VelocidadeHz	INT	K4000	Frequência de pulso com sinal (+/- Hz)
OUT	SaidaPulso	BOOL	Y000	Saída de pulso contínuo
OUT	SaidaDir	BOOL	Y004	Saída de direção

Associar na IDE

Crie a instância Inst_FB_JOG_VELOCIDADE_PLSV. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

11 FB_JOG_VELOCIDADE_PLSV

2. Funcionamento, instrução por instrução

01	LD {JogAtivo}	Inicia uma condição lendo {JogAtivo}.
02	PLSV {VelocidadeHz} {SaidaPulso} {SaidaDir}	Se habilitado, emite PLSV com operandos na ordem mostrada. Valide saída, direção, sinal e frequência no manual da CPU e em bancada.

Exemplo IL com as associações padrão

```
LD M6
PLSV K4000 Y000 Y004
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

11 FB_JOG_VELOCIDADE_PLSV

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
JogAtivo=0	PLSV: 0
JogAtivo=1; VelocidadeHz=100	PLSV(100,SaidaPulso,SaidaDir)
JogAtivo=1; VelocidadeHz=-100	PLSV(-100,SaidaPulso,SaidaDir)

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

A tabela de ensaio descreve o comando emitido, não uma posição ou frequência já medida. Use carga desacoplada, limites e instrumentação no comissionamento. Saídas permitidas, frequência, reaplicação do comando e parada dependem da CPU.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

12 FB_TREM_PULSOS_PLSY

Gerador de trem de pulsos com frequência e contagem fixas (PLSY 16-bit)

2.4 Blocos de Função de Posicionamento e Motion (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Disparar	BOOL	M7	Habilita emissão de pulsos
IN	FreqHz	INT	K5000	Frequência de saída (Hz)
IN	QtdPulsos	INT	K12000	Quantidade total de pulsos (K0 = contínuo)
OUT	SaidaPulso	BOOL	Y000	Saída rápida Y000..Y007

Associar na IDE

Crie a instância Inst_FB_TREM_PULSOS_PLSY. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

12 FB_TREM_PULSOS_PLSY

2. Funcionamento, instrução por instrução

01	LD {Disparar}	Inicia uma condição lendo {Disparar}.
02	PLSY {FreqHz} {QtdPulsos} {SaidaPulso}	Se habilitado, emite PLSY com operandos na ordem mostrada. Valide saída, direção, sinal e frequência no manual da CPU e em bancada.

Exemplo IL com as associações padrão

```
LD M7
PLSY K5000 K12000 Y000
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

12 FB_TREM_PULSOS_PLSY

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Disparar=0	PLSY: 0
Disparar=1; FreqHz=100; QtdPulsos=10	PLSY(100,10,SaidaPulso)
Disparar=1; FreqHz=100; QtdPulsos=0	PLSY(100,0,SaidaPulso)

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

A tabela de ensaio descreve o comando emitido, não uma posição ou frequência já medida. Use carga desacoplada, limites e instrumentação no comissionamento. Saídas permitidas, frequência, reaplicação do comando e parada dependem da CPU.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

13 FB_TREM_PULSOS_DPLSY

Gerador de trem de pulsos 32-bit de longo curso (DPLSY estruturado)

2.4 Blocos de Função de Posicionamento e Motion (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Disparar	BOOL	M8	Habilita emissão 32-bit
IN	Freq32b	DINT	K10000	Frequência 32-bit (Hz)
IN	QtdPulsos32b	DINT	K50000	Total de pulsos 32-bit
OUT	SaidaPulso	BOOL	Y000	Saída rápida de pulso

Associar na IDE

Crie a instância Inst_FB_TREM_PULSOS_DPLSY. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

13 FB_TREM_PULSOS_DPLSY

2. Funcionamento, instrução por instrução

01	LD {Disparar}	Inicia uma condição lendo {Disparar}.
02	DPLSY {Freq32b} {QtdPulsos32b} {SaidaPulso}	Se habilitado, emite DPLSY com operandos na ordem mostrada. Valide saída, direção, sinal e frequência no manual da CPU e em bancada.

Exemplo IL com as associações padrão

```
LD M8
DPLSY K10000 K50000 Y000
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

13 FB_TREM_PULSOS_DPLSY

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Disparar=0	DPLSY: 0
Disparar=1; Freq32b=100; QtdPulsos32b=10	DPLSY(100,10,SaidaPulso)
Disparar=1; Freq32b=100; QtdPulsos32b=0	DPLSY(100,0,SaidaPulso)

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

A tabela de ensaio descreve o comando emitido, não uma posição ou frequência já medida. Use carga desacoplada, limites e instrumentação no comissionamento. Saídas permitidas, frequência, reaplicação do comando e parada dependem da CPU.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

14 FB_INTERTRAV_EIXO

Supervisão de limites de curso (+/-), Servo Ready e parada de eixo



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	ServoReady	BOOL	X004	Sinal Servo Pronto / Sem Alarme
IN	FimCursoPos	BOOL	X005	Sensor limite positivo (NF)
IN	FimCursoNeg	BOOL	X006	Sensor limite negativo (NF)
IN	Emergencia	BOOL	X007	Botão de emergência (NF)
OUT	EixoLiberado	BOOL	M110	Permissão global para movimento do eixo
OUT	ServoEnableY	BOOL	Y006	Saída Servo ON

Associar na IDE

Crie a instância Inst_FB_INTERTRAV_EIXO. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

14 FB_INTERTRAV_EIXO

2. Funcionamento, instrução por instrução

01	LD {ServoReady}	Inicia uma condição lendo {ServoReady}.
02	ANI {FimCursoPos}	Exige também {FimCursoPos} falso; confira a polaridade do sinal real.
03	ANI {FimCursoNeg}	Exige também {FimCursoNeg} falso; confira a polaridade do sinal real.
04	ANI {Emergencia}	Exige também {Emergencia} falso; confira a polaridade do sinal real.
05	OUT {EixoLiberado}	Escreve a condição atual em {EixoLiberado} a cada scan.
06	OUT {ServoEnableY}	Escreve a condição atual em {ServoEnableY} a cada scan.

Exemplo IL com as associações padrão

```
LD X004
ANI X005
ANI X006
ANI X007
OUT M110
OUT Y006
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

14 FB_INTERTRAV_EIXO

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
ServoReady=1; FimCursoPos=0; FimCursoNeg=0; Emergencia=0	EixoLiberado=1; ServoEnableY=1
FimCursoPos=1	EixoLiberado=0; ServoEnableY=0
Emergencia=1	EixoLiberado=0; ServoEnableY=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

ANI exige bit 0 para liberar. Um botão NF pode produzir bit 1 em repouso dependendo da ligação: normalize o sinal e teste cabo rompido e comando simultâneo. A lógica apresentada não substitui circuito de segurança certificado.

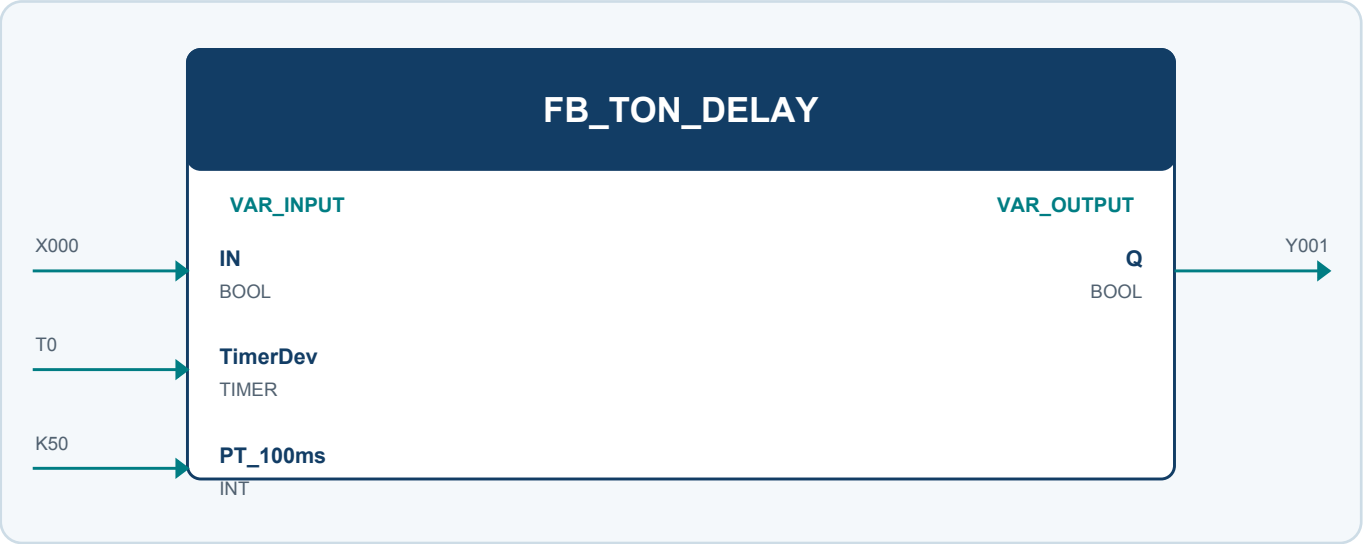
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

15 FB_TON_DELAY

Temporizador de retardo na energização base 100ms (TON / TIMER_100_FB_M)

3.2 Blocos Temporizadores IEC (TON / TOF / TP)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	IN	BOOL	X000	Entrada de habilitação do timer
IN	TimerDev	TIMER	T0	Dispositivo Timer (T0..T199)
IN	PT_100ms	INT	K50	Preset em base 100ms (ex: K50 = 5.0s)
OUT	Q	BOOL	Y001	Saída acionada após o tempo PT

Associar na IDE

Crie a instância Inst_FB_TON_DELAY. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

15 FB_TON_DELAY

2. Funcionamento, instrução por instrução

01	LD {IN}	Inicia uma condição lendo {IN}.
02	TMR {TimerDev} {PT_100ms}	Atualiza o temporizador {TimerDev} com preset {PT_100ms}. A base de tempo depende do dispositivo T.
03	OUT {Q}	Escreve a condição atual em {Q} a cada scan.

Exemplo IL com as associações padrão

```
LD X000
TMR T0 K50
OUT Y001
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

15 FB_TON_DELAY

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
IN=0	TimerDev=0; Q=0
IN=1; t=0	Q=1; TimerDev=0
IN=1; t>=PT_100ms	Q=1; TimerDev=1

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Limite atual: o modelo IL executa TMR e depois OUT Q sem carregar o contato TimerDev; não garante Q após PT. Para atraso em IL use LD IN / OUT T0 K30 / LD T0 / OUT Q. Teste antes de adotar este bloco.

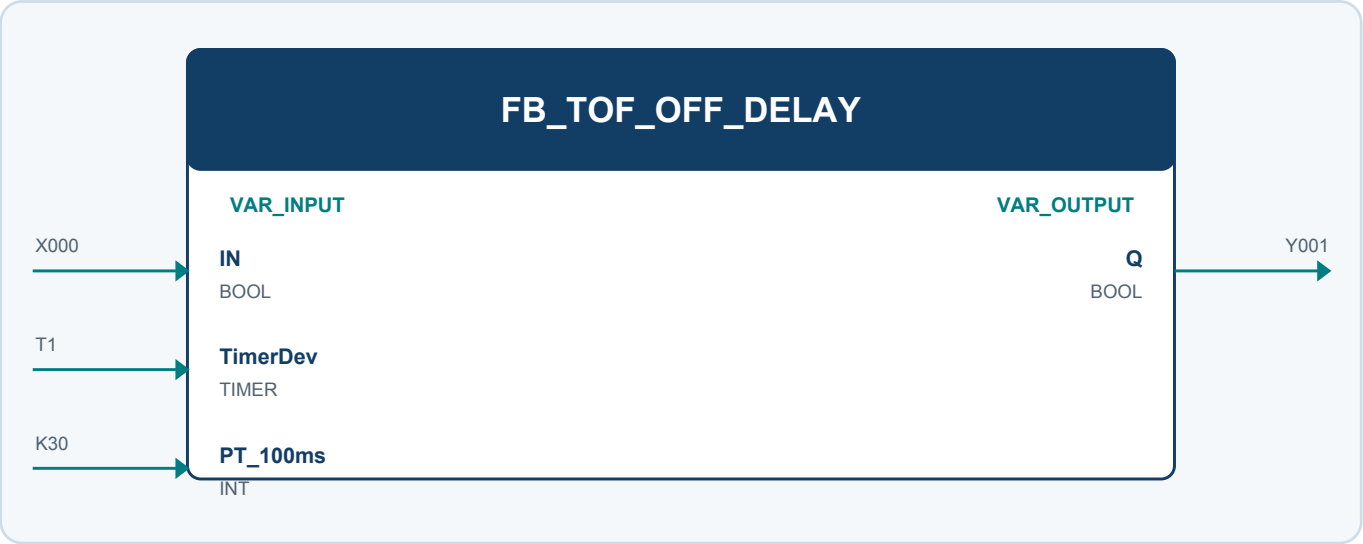
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

16 FB_TOF_OFF_DELAY

Temporizador de retardo na desenergização (TOF estruturado)

3.2 Blocos Temporizadores IEC (TON / TOF / TP)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	IN	BOOL	X000	Sinal de entrada
IN	TimerDev	TIMER	T1	Timer auxiliar de desligamento
IN	PT_100ms	INT	K30	Tempo de retenção após queda de IN
OUT	Q	BOOL	Y001	Saída mantida ligada durante o retardo

Associar na IDE

Crie a instância Inst_FB_TOF_OFF_DELAY. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

16 FB_TOF_OFF_DELAY

2. Funcionamento, instrução por instrução

01	LDI {IN}	Inicia uma condição com NOT {IN}.
02	TMR {TimerDev} {PT_100ms}	Atualiza o temporizador {TimerDev} com preset {PT_100ms}. A base de tempo depende do dispositivo T.
03	LD {IN}	Inicia uma condição lendo {IN}.
04	ANI {TimerDev}	Exige também {TimerDev} falso; confira a polaridade do sinal real.
05	OUT {Q}	Escreve a condição atual em {Q} a cada scan.

Exemplo IL com as associações padrão

```
LDI X000
TMR T1 K30
LD X000
ANI T1
OUT Y001
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

16 FB_TOF_OFF_DELAY

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
IN=1	Q=1
IN: 1 -> 0; t=0	Q=0
IN=0; t>=PT_100ms	Q=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Limite atual: OUT Q depende de IN; ao desligar IN a saída cai imediatamente no modelo IL. Este modelo não implementa retenção TOF IEC durante PT.

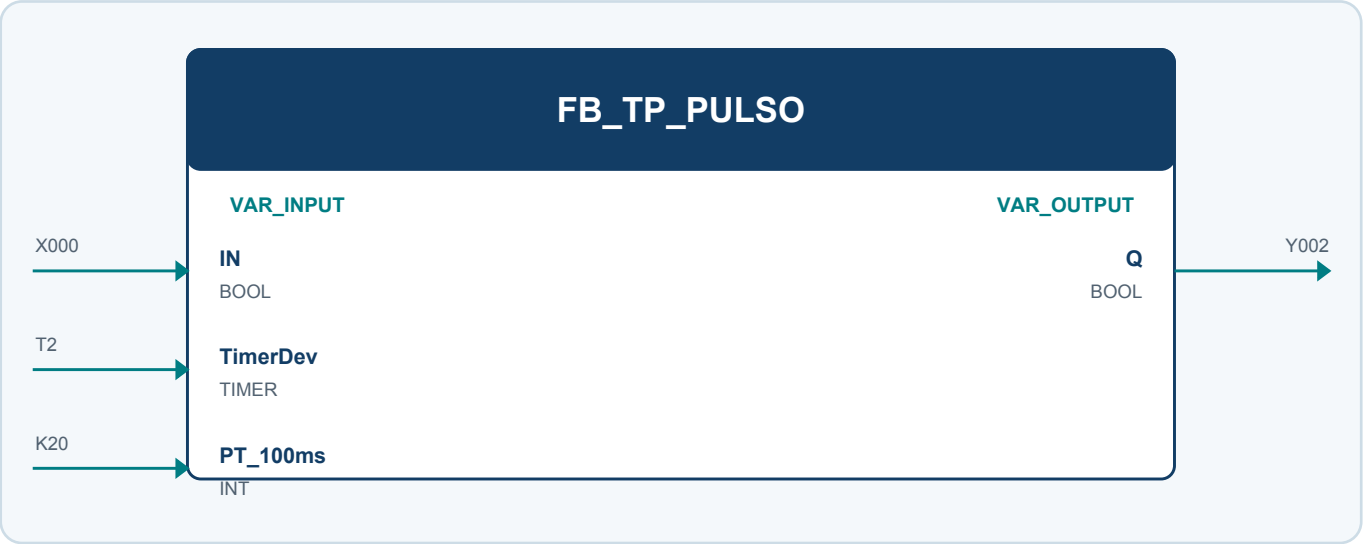
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

17 FB_TP_PULSO

Temporizador de pulso monostável de largura fixa (TP IEC 61131-3)

3.2 Blocos Temporizadores IEC (TON / TOF / TP)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	IN	BOOL	X000	Gatilho de disparo do pulso
IN	TimerDev	TIMER	T2	Dispositivo Timer (T0..T199)
IN	PT_100ms	INT	K20	Duração fixa do pulso (100ms)
OUT	Q	BOOL	Y002	Saída de pulso temporizado

Associar na IDE

Crie a instância Inst_FB_TP_PULSO. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

17 FB_TP_PULSO

2. Funcionamento, instrução por instrução

01	LD {IN}	Inicia uma condição lendo {IN}.
02	TMR {TimerDev} {PT_100ms}	Atualiza o temporizador {TimerDev} com preset {PT_100ms}. A base de tempo depende do dispositivo T.
03	LD {IN}	Inicia uma condição lendo {IN}.
04	ANI {TimerDev}	Exige também {TimerDev} falso; confira a polaridade do sinal real.
05	OUT {Q}	Escreve a condição atual em {Q} a cada scan.

Exemplo IL com as associações padrão

```
LD X000
TMR T2 K20
LD X000
ANI T2
OUT Y002
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

17 FB_TP_PULSO

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
IN: 0 -> 1; t<PT_100ms	Q=1
IN=1; t>=PT_100ms	Q=0
IN: 1 -> 0; t<PT_100ms	Q=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Limite atual: Q = IN AND NOT TimerDev; retirar IN encerra o pulso. Não equivale a um monoestável IEC independente da duração de IN.

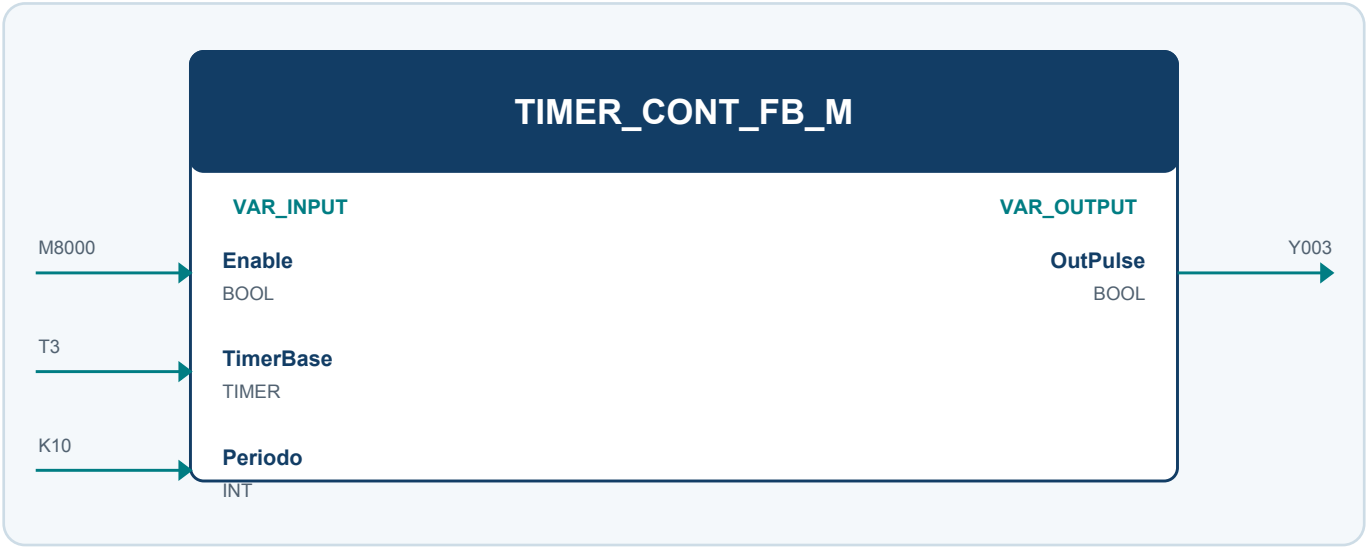
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

18 TIMER_CONT_FB_M

Oscilador cíclico / Pisca-pisca contínuo (Blink / Clock Generator)

3.2 Blocos Temporizadores IEC (TON / TOF / TP)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Enable	BOOL	M8000	Habilita gerador de pulsos
IN	TimerBase	TIMER	T3	Timer de período
IN	Periodo	INT	K10	Semi-período em 100ms
OUT	OutPulse	BOOL	Y003	Saída osciladora

Associar na IDE

Crie a instância Inst_TIMER_CONT_FB_M. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

18 TIMER_CONT_FB_M

2. Funcionamento, instrução por instrução

01	LD {Enable}	Inicia uma condição lendo {Enable}.
02	ANI {TimerBase}	Exige também {TimerBase} falso; confira a polaridade do sinal real.
03	TMR {TimerBase} {Periodo}	Atualiza o temporizador {TimerBase} com preset {Periodo}. A base de tempo depende do dispositivo T.
04	LD {TimerBase}	Inicia uma condição lendo {TimerBase}.
05	OUT {OutPulse}	Escreve a condição atual em {OutPulse} a cada scan.

Exemplo IL com as associações padrão

```
LD M8000
ANI T3
TMR T3 K10
LD T3
OUT Y003
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

18 TIMER_CONT_FB_M

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Enable=0	TimerBase=0; OutPulse=0
Enable=1; t<Periodo	TimerBase=0; OutPulse=0
Enable=1; t>=Periodo	TimerBase=1; OutPulse=1 -> 0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

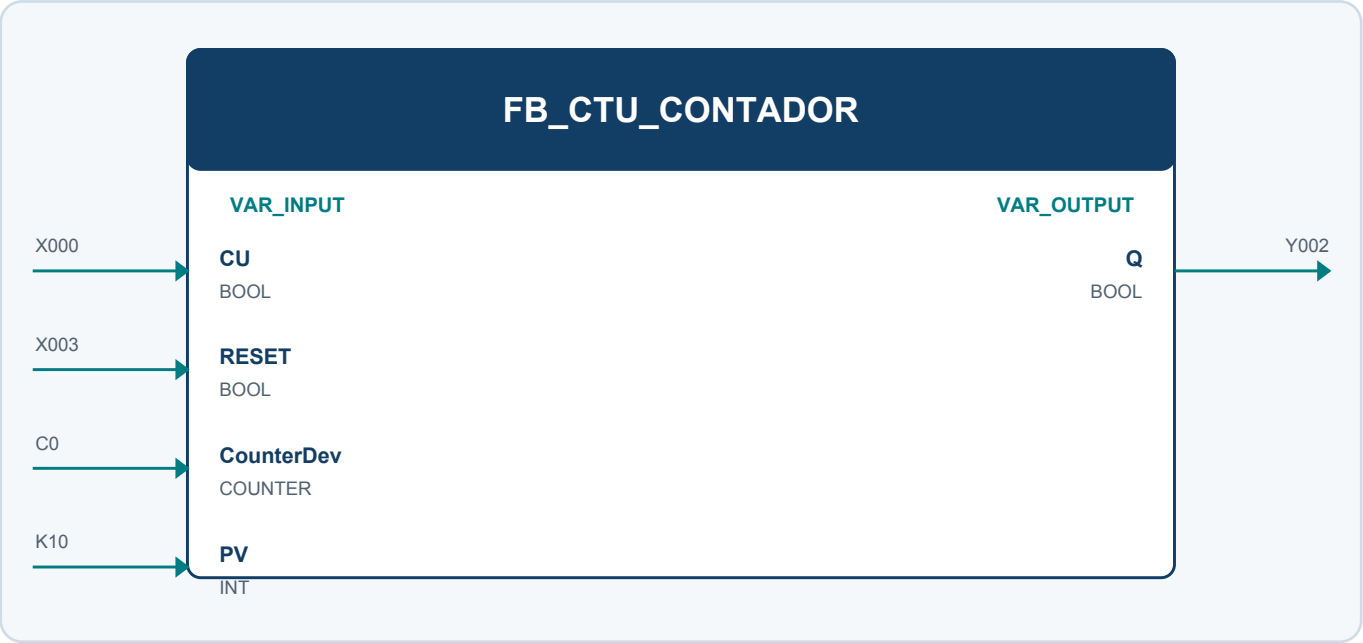
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

19 FB_CTU_CONTADOR

Contador incremental estruturado (CTU / COUNTER_FB_M) com entrada de Reset

3.3 Blocos Contadores, Biestáveis e Borda (CTU / SR / R_TRIG)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	CU	BOOL	X000	Pulso de contagem crescente
IN	RESET	BOOL	X003	Reset do contador
IN	CounterDev	COUNTER	C0	Dispositivo Contador (C0..C255)
IN	PV	INT	K10	Valor alvo de contagem
OUT	Q	BOOL	Y002	Saída de meta atingida

Associar na IDE

Crie a instância Inst_FB_CTU_CONTADOR. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

19 FB_CTU_CONTADOR

2. Funcionamento, instrução por instrução

01	LD {CU}	Inicia uma condição lendo {CU}.
02	ANI {RESET}	Exige também {RESET} falso; confira a polaridade do sinal real.
03	CNT {CounterDev} {PV}	Atualiza o contador {CounterDev} com preset {PV}. Verifique reset e contato de fim separadamente.
04	OUT {Q}	Escreve a condição atual em {Q} a cada scan.

Exemplo IL com as associações padrão

```
LD X000
ANI X003
CNT C0 K10
OUT Y002
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

19 FB_CTU_CONTADOR

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
CU=0; RESET=0	Q=0
CU: 0 -> 1; RESET=0; CV=0; PV>1	CV=1; Q=1
CU=1; RESET=1; CV=1	CV=1; Q=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Limite atual: RESET apenas inibe CNT; não há RST CounterDev no modelo IL. Q não é carregado do contato CounterDev. Adicione e teste reset e indicação de fim de contagem explicitamente.

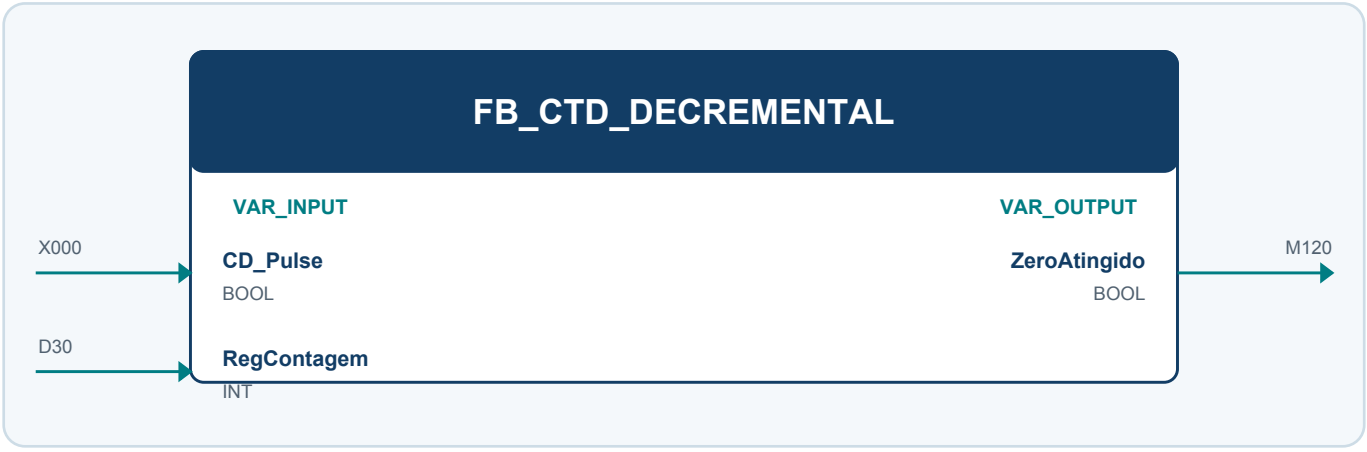
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

20 FB_CTD_DECREMENTAL

Contador decrescente por registrador (CTD IEC 61131-3)

3.3 Blocos Contadores, Biestáveis e Borda (CTU / SR / R_TRIG)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	CD_Pulse	BOOL	X000	Pulso de decremento
IN	RegContagem	INT	D30	Registrador de contagem atual
OUT	ZeroAtingido	BOOL	M120	Indica contagem <= 0

Associar na IDE

Crie a instância Inst_FB_CTD_DECREMENTAL. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

20 FB_CTD_DECREMENTAL

2. Funcionamento, instrução por instrução

01	LDP {CD_Pulse}	Condição verdadeira por um scan na borda 0 -> 1 de {CD_Pulse}.
02	DEC {RegContagem}	Se habilitado, decrementa {RegContagem} em 1.
03	LD<= {RegContagem} K0	Inicia a condição com {RegContagem} <= K0.
04	OUT {ZeroAtingido}	Escreve a condição atual em {ZeroAtingido} a cada scan.

Exemplo IL com as associações padrão

```
LDP X000
DEC D30
LD<= D30 K0
OUT M120
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

20 FB_CTD_DECREMENTAL

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
RegContagem=2; CD_Pulse: 0 -> 1	RegContagem=1; ZeroAtingido=0
RegContagem=1; CD_Pulse: 0 -> 1	RegContagem=0; ZeroAtingido=1
RegContagem=0; CD_Pulse: 0 -> 1	RegContagem=-1; ZeroAtingido=1

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

21 FB_CTUD_BIDIRECIONAL

Contador bidirecional Up/Down em registrador (CTUD IEC 61131-3)

3.3 Blocos Contadores, Biestáveis e Borda (CTU / SR / R_TRIG)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	CountUp	BOOL	X000	Pulso incrementa +1
IN	CountDown	BOOL	X001	Pulso decrementa -1
IN	PresetPV	INT	K50	Meta superior PV
OUT	ValorCV	INT	D32	Contagem atual CV
OUT	MetaQU	BOOL	M121	Saída CV >= PV

Associar na IDE

Crie a instância Inst_FB_CTUD_BIDIRECIONAL. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

21 FB_CTUD_BIDIRECIONAL

2. Funcionamento, instrução por instrução

01	LDP {CountUp}	Condição verdadeira por um scan na borda 0 -> 1 de {CountUp}.
02	INC {ValorCV}	Se habilitado, incrementa {ValorCV} em 1.
03	LDP {CountDown}	Condição verdadeira por um scan na borda 0 -> 1 de {CountDown}.
04	DEC {ValorCV}	Se habilitado, decrementa {ValorCV} em 1.
05	LD>= {ValorCV} {PresetPV}	Inicia a condição com {ValorCV} >= {PresetPV}.
06	OUT {MetaQU}	Escreve a condição atual em {MetaQU} a cada scan.

Exemplo IL com as associações padrão

```
LDP X000
INC D32
LDP X001
DEC D32
LD>= D32 K50
OUT M121
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

21 FB_CTUD_BIDIRECIONAL

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
ValorCV=0; CountUp: 0 -> 1; CountDown=0	ValorCV=1
ValorCV=1; CountUp=0; CountDown: 0 -> 1	ValorCV=0
ValorCV=0; CountUp: 0 -> 1; CountDown: 0 -> 1	ValorCV=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

22 FB_SR_SET_DOMINANTE

Flip-Flop biestável SR com prioridade para Set (IEC 61131-3 SR)

3.3 Blocos Contadores, Biestáveis e Borda (CTU / SR / R_TRIG)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	SET1	BOOL	X000	Entrada Set dominante
IN	RESET	BOOL	X001	Entrada Reset
OUT	Q1	BOOL	M130	Estado retido do biestável

Associar na IDE

Crie a instância Inst_FB_SR_SET_DOMINANTE. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

22 FB_SR_SET_DOMINANTE

2. Funcionamento, instrução por instrução

01	LD {RESET}	Inicia uma condição lendo {RESET}.
02	RST {Q1}	Se habilitado, zera {Q1}; para T/C também reinicia o valor acumulado.
03	LD {SET1}	Inicia uma condição lendo {SET1}.
04	SET {Q1}	Se habilitado, memoriza 1 em {Q1}; não desliga ao retirar o comando.

Exemplo IL com as associações padrão

```
LD X001
RST M130
LD X000
SET M130
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

22 FB_SR_SET_DOMINANTE

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
SET1=1; RESET=0	Q1=1
SET1=0; RESET=1	Q1=0
SET1=1; RESET=1	Q1=1

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

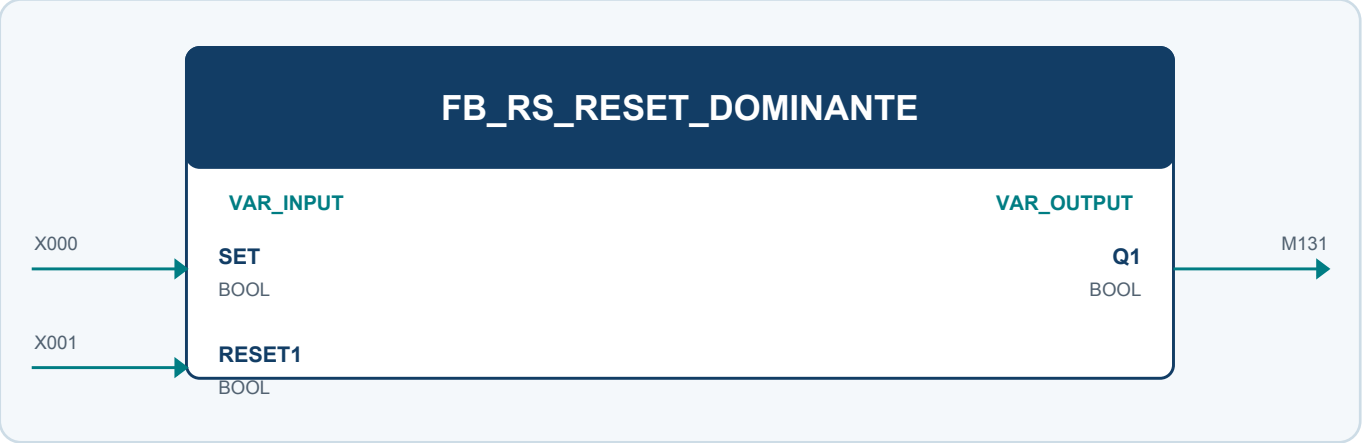
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

23 FB_RS_RESET_DOMINANTE

Flip-Flop biestável RS com prioridade para Reset (IEC 61131-3 RS)

3.3 Blocos Contadores, Biestáveis e Borda (CTU / SR / R_TRIG)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	SET	BOOL	X000	Entrada Set
IN	RESET1	BOOL	X001	Entrada Reset dominante
OUT	Q1	BOOL	M131	Estado retido do biestável

Associar na IDE

Crie a instância Inst_FB_RS_RESET_DOMINANTE. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

23 FB_RS_RESET_DOMINANTE

2. Funcionamento, instrução por instrução

01	LD {SET}	Inicia uma condição lendo {SET}.
02	SET {Q1}	Se habilitado, memoriza 1 em {Q1}; não desliga ao retirar o comando.
03	LD {RESET1}	Inicia uma condição lendo {RESET1}.
04	RST {Q1}	Se habilitado, zera {Q1}; para T/C também reinicia o valor acumulado.

Exemplo IL com as associações padrão

```
LD X000
SET M131
LD X001
RST M131
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

23 FB_RS_RESET_DOMINANTE

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
SET=1; RESET1=0	Q1=1
SET=0; RESET1=1	Q1=0
SET=1; RESET1=1	Q1=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

24 FB_R_TRIG_BORDA_SUB

Detector de borda de subida de 1 scan (R_TRIG IEC 61131-3)

3.3 Blocos Contadores, Biestáveis e Borda (CTU / SR / R_TRIG)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	CLK	BOOL	X000	Sinal monitorado
OUT	Q	BOOL	M132	Pulso de 1 scan na subida de CLK

Associar na IDE

Crie a instância Inst_FB_R_TRIG_BORDA_SUB. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

24 FB_R_TRIG_BORDA_SUB

2. Funcionamento, instrução por instrução

01	LDP {CLK}	Condição verdadeira por um scan na borda 0 -> 1 de {CLK}.
02	OUT {Q}	Escreve a condição atual em {Q} a cada scan.

Exemplo IL com as associações padrão

```
LDP X000
OUT M132
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

24 FB_R_TRIG_BORDA_SUB

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
CLK: 0 -> 1	Q=1
CLK: 1 -> 1	Q=0
CLK: 1 -> 0	Q=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

25 FB_F_TRIG_BORDA_DESC

Detector de borda de descida de 1 scan (F_TRIG IEC 61131-3)

3.3 Blocos Contadores, Biestáveis e Borda (CTU / SR / R_TRIG)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	CLK	BOOL	X000	Sinal monitorado
OUT	Q	BOOL	M133	Pulso de 1 scan na descida de CLK

Associar na IDE

Crie a instância Inst_FB_F_TRIG_BORDA_DESC. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

25 FB_F_TRIG_BORDA_DESC

2. Funcionamento, instrução por instrução

01	LDF {CLK}	Condição verdadeira por um scan na borda 1 -> 0 de {CLK}.
02	OUT {Q}	Escreve a condição atual em {Q} a cada scan.

Exemplo IL com as associações padrão

```
LDF X000
OUT M133
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

25 FB_F_TRIG_BORDA_DESC

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
CLK: 1 -> 0	Q=1
CLK: 0 -> 0	Q=0
CLK: 0 -> 1	Q=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

26 FB_SEL_SELETOR

Seletor binário de 2 entradas inteiras (IEC 61131-3 SEL: G=0 -> IN0, G=1 -> IN1)

4.5 Blocos de Comparação e Seleção IEC (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	G_Select	BOOL	M0	Chave seletora (OFF=IN0, ON=IN1)
IN	IN0	INT	D10	Valor selecionado quando G=OFF
IN	IN1	INT	D11	Valor selecionado quando G=ON
OUT	OUT_Val	INT	D12	Registrador de saída selecionado

Associar na IDE

Crie a instância Inst_FB_SEL_SELETOR. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

26 FB_SEL_SELETOR

2. Funcionamento, instrução por instrução

01	LDI {G_Select}	Inicia uma condição com NOT {G_Select}.
02	MOV {IN0} {OUT_Val}	Se habilitado, copia {IN0} para {OUT_Val}.
03	LD {G_Select}	Inicia uma condição lendo {G_Select}.
04	MOV {IN1} {OUT_Val}	Se habilitado, copia {IN1} para {OUT_Val}.

Exemplo IL com as associações padrão

```
LDI M0
MOV D10 D12
LD M0
MOV D11 D12
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

26 FB_SEL_SELETOR

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
IN0=10; IN1=20; G_Select=0	OUT_Val=10
IN0=10; IN1=20; G_Select=1	OUT_Val=20
IN0=-5; IN1=-5; G_Select=1	OUT_Val=-5

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

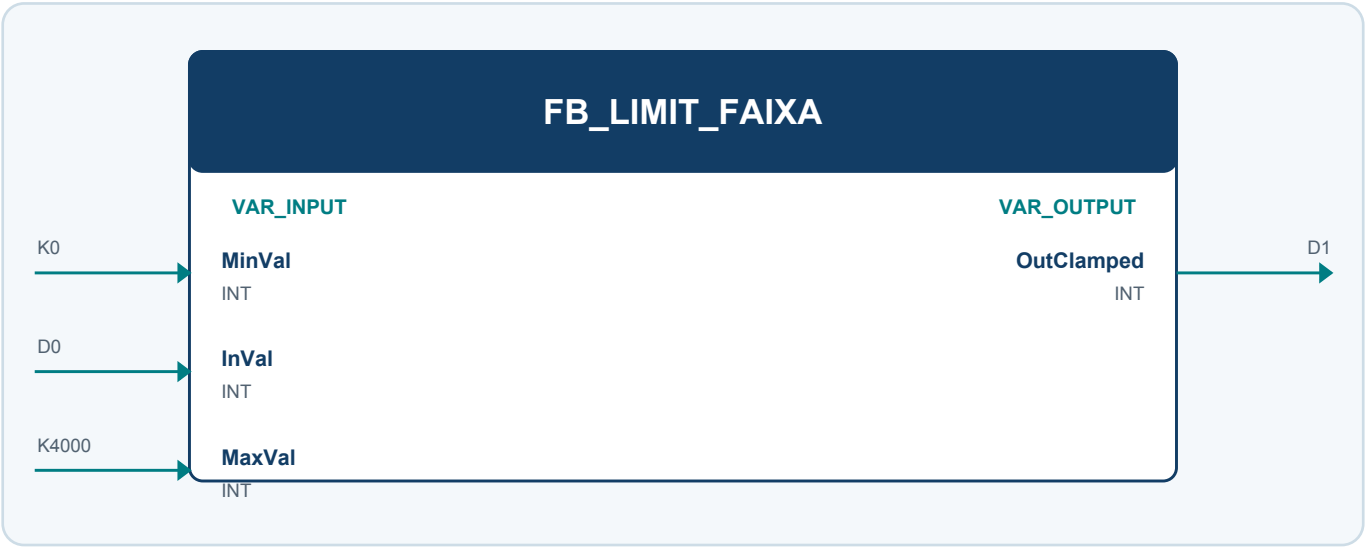
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

27 FB_LIMIT_FAIXA

Limitador de valor mínimo e máximo (IEC 61131-3 LIMIT: MN <= IN <= MX)

4.5 Blocos de Comparação e Seleção IEC (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	MinVal	INT	K0	Limite inferior MN
IN	InVal	INT	D0	Variável de entrada IN
IN	MaxVal	INT	K4000	Limite superior MX
OUT	OutClamped	INT	D1	Saída limitada à faixa MN..MX

Associar na IDE

Crie a instância Inst_FB_LIMIT_FAIXA. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

27 FB_LIMIT_FAIXA

2. Funcionamento, instrução por instrução

01	LD M8000	Inicia uma condição lendo M8000.
02	MOV {InVal} {OutClamped}	Se habilitado, copia {InVal} para {OutClamped}.
03	LD< {InVal} {MinVal}	Inicia a condição com {InVal} < {MinVal}.
04	MOV {MinVal} {OutClamped}	Se habilitado, copia {MinVal} para {OutClamped}.
05	LD> {InVal} {MaxVal}	Inicia a condição com {InVal} > {MaxVal}.
06	MOV {MaxVal} {OutClamped}	Se habilitado, copia {MaxVal} para {OutClamped}.

Exemplo IL com as associações padrão

```
LD M8000
MOV D0 D1
LD< D0 K0
MOV K0 D1
LD> D0 K4000
MOV K4000 D1
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

27 FB_LIMIT_FAIXA

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
MinVal=0; MaxVal=100; InVal=-5	OutClamped=0
MinVal=0; MaxVal=100; InVal=50	OutClamped=50
MinVal=0; MaxVal=100; InVal=150	OutClamped=100

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

28 FB_MAX_VALOR

Seleciona o maior valor entre duas entradas (IEC 61131-3 MAX)

4.5 Blocos de Comparação e Seleção IEC (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	IN1	INT	D0	Entrada 1
IN	IN2	INT	D1	Entrada 2
OUT	MaxOut	INT	D2	Maior valor entre IN1 e IN2

Associar na IDE

Crie a instância Inst_FB_MAX_VALOR. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

28 FB_MAX_VALOR

2. Funcionamento, instrução por instrução

01	LD>= {IN1} {IN2}	Inicia a condição com {IN1} >= {IN2}.
02	MOV {IN1} {MaxOut}	Se habilitado, copia {IN1} para {MaxOut}.
03	LD< {IN1} {IN2}	Inicia a condição com {IN1} < {IN2}.
04	MOV {IN2} {MaxOut}	Se habilitado, copia {IN2} para {MaxOut}.

Exemplo IL com as associações padrão

```
LD>= D0 D1
MOV D0 D2
LD< D0 D1
MOV D1 D2
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

28 FB_MAX_VALOR

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
IN1=10; IN2=20	MaxOut=20
IN1=20; IN2=10	MaxOut=20
IN1=-5; IN2=-5	MaxOut=-5

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

29 FB_MIN_VALOR

Seleciona o menor valor entre duas entradas (IEC 61131-3 MIN)

4.5 Blocos de Comparação e Seleção IEC (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	IN1	INT	D0	Entrada 1
IN	IN2	INT	D1	Entrada 2
OUT	MinOut	INT	D3	Menor valor entre IN1 e IN2

Associar na IDE

Crie a instância Inst_FB_MIN_VALOR. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

29 FB_MIN_VALOR

2. Funcionamento, instrução por instrução

01	LD<= {IN1} {IN2}	Inicia a condição com {IN1} <= {IN2}.
02	MOV {IN1} {MinOut}	Se habilitado, copia {IN1} para {MinOut}.
03	LD> {IN1} {IN2}	Inicia a condição com {IN1} > {IN2}.
04	MOV {IN2} {MinOut}	Se habilitado, copia {IN2} para {MinOut}.

Exemplo IL com as associações padrão

```
LD<= D0 D1
MOV D0 D3
LD> D0 D1
MOV D1 D3
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

29 FB_MIN_VALOR

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
IN1=10; IN2=20	MinOut=10
IN1=20; IN2=10	MinOut=10
IN1=-5; IN2=-5	MinOut=-5

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

30 FB_HISTERESE_ZONA

Controle On/Off com banda morta / histerese (liga abaixo de Low, desliga acima de High)

4.5 Blocos de Comparação e Seleção IEC (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	PV	INT	D8030	Variável medida
IN	LimiteBaixo	INT	K1000	Ponto inferior (Liga)
IN	LimiteAlto	INT	K3000	Ponto superior (Desliga)
OUT	SaidaControle	BOOL	Y001	Saída com histerese

Associar na IDE

Crie a instância Inst_FB_HISTERESE_ZONA. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

30 FB_HISTERESE_ZONA

2. Funcionamento, instrução por instrução

01	LD< {PV} {LimiteBaixo}	Inicia a condição com {PV} < {LimiteBaixo}.
02	SET {SaidaControle}	Se habilitado, memoriza 1 em {SaidaControle}; não desliga ao retirar o comando.
03	LD> {PV} {LimiteAlto}	Inicia a condição com {PV} > {LimiteAlto}.
04	RST {SaidaControle}	Se habilitado, zera {SaidaControle}; para T/C também reinicia o valor acumulado.

Exemplo IL com as associações padrão

```
LD< D8030 K1000
SET Y001
LD> D8030 K3000
RST Y001
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

30 FB_HISTERESE_ZONA

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
LimiteBaixo=20; LimiteAlto=30; PV=10	SaidaControle=1
LimiteBaixo=20; LimiteAlto=30; PV=25	SaidaControle(t)=SaidaControle(t-1)
LimiteBaixo=20; LimiteAlto=30; PV=40	SaidaControle=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

31 FB_REGRA_DE_TRES

Conversão proporcional por regra de três ((Entrada * EscalaMax) / BaseMax)

5.3 Blocos Matemáticos e Totalizadores (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Entrada	INT	D0	Valor lido
IN	EscalaMax	INT	K1000	Fundo de escala desejado
IN	BaseMax	INT	K4000	Resolução máxima de entrada
OUT	TempMul	INT	D96	Produto intermediário
OUT	SaidaEscala	INT	D1	Resultado proporcional

Associar na IDE

Crie a instância Inst_FB_REGRA_DE_TRES. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

31 FB_REGRA_DE_TRES

2. Funcionamento, instrução por instrução

01	LD M8000	Inicia uma condição lendo M8000.
02	MUL {Entrada} {EscalaMax} {TempMul}	Se habilitado, multiplica {Entrada} por {EscalaMax} e grava o par iniciado em {TempMul}.
03	DIV {TempMul} {BaseMax} {SaidaEscala}	Se habilitado, divide {TempMul} por {BaseMax}: quociente em {SaidaEscala} e resto no D seguinte. Divisor zero é inválido.

Exemplo IL com as associações padrão

```
LD M8000
MUL D0 K1000 D96
DIV D96 K4000 D1
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

31 FB_REGRA_DE_TRES

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Entrada=100; EscalaMax=100; BaseMax=1000	TempMul=10000; SaidaEscala=10
Entrada=0; EscalaMax=100; BaseMax=1000	SaidaEscala=0
Entrada=-100; EscalaMax=100; BaseMax=1000	TempMul=-10000; SaidaEscala=-10

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

O produto de MUL ocupa dois D, mas DIV sem prefixo D usa o operando de 16 bits. Os exemplos pequenos evitam truncamento; não extrapole para toda a escala. Para produtos fora de INT16, revise a sequência com operações de 32 bits.

Reserve também o D seguinte a cada resultado de MUL/DIV. Um resultado dirigido a DA pode escrever o canal analógico adjacente; confira o mapa antes de usar saídas reais.

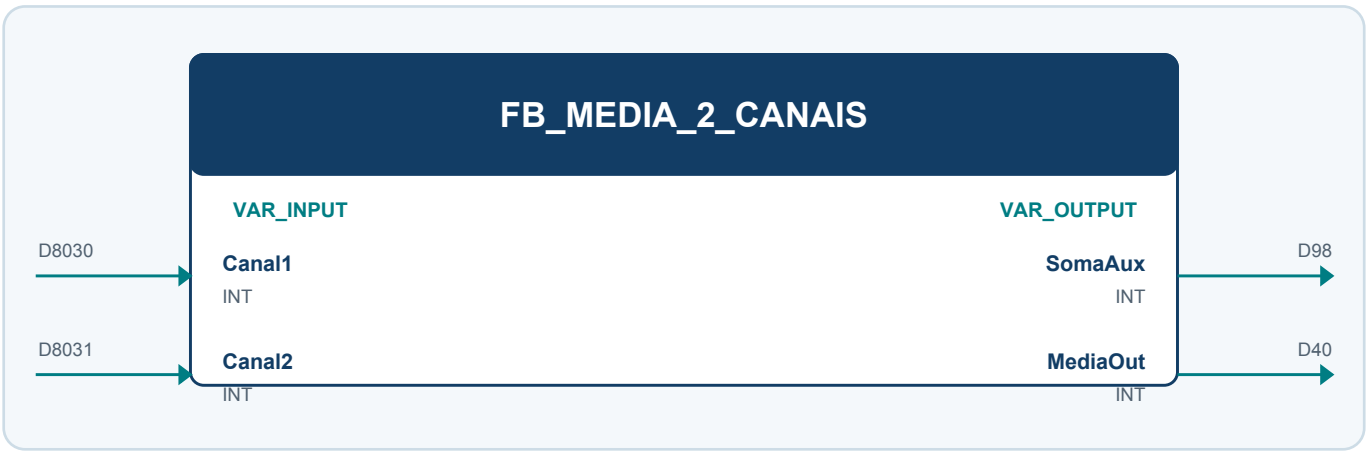
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

32 FB_MEDIA_2_CANAIS

Cálculo de média aritmética entre dois sensores ((CH1 + CH2) / 2)

5.3 Blocos Matemáticos e Totalizadores (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Canal1	INT	D8030	Leitura sensor 1
IN	Canal2	INT	D8031	Leitura sensor 2
OUT	SomaAux	INT	D98	Soma dos canais
OUT	MediaOut	INT	D40	Média calculada

Associar na IDE

Crie a instância Inst_FB_MEDIA_2_CANAIS. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

32 FB_MEDIA_2_CANAIS

2. Funcionamento, instrução por instrução

01	LD M8000	Inicia uma condição lendo M8000.
02	ADD {Canal1} {Canal2} {SomaAux}	Se habilitado, calcula {Canal1} + {Canal2} em {SomaAux}; confira overflow de 16 bits.
03	DIV {SomaAux} K2 {MediaOut}	Se habilitado, divide {SomaAux} por K2: quociente em {MediaOut} e resto no D seguinte. Divisor zero é inválido.

Exemplo IL com as associações padrão

```
LD M8000
ADD D8030 D8031 D98
DIV D98 K2 D40
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

32 FB_MEDIA_2_CANAIS

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Canal1=100; Canal2=200	SomaAux=300; MediaOut=150
Canal1=101; Canal2=200	SomaAux=301; MediaOut=150; R=1
Canal1=-100; Canal2=100	MediaOut=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Reserve também o D seguinte a cada resultado de MUL/DIV. Um resultado dirigido a DA pode escrever o canal analógico adjacente; confira o mapa antes de usar saídas reais.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

33 FB_COPIA_RECEITA_3W

Carrega 3 parâmetros de receita para os registradores de trabalho

6.3 Blocos de Transferência e Receitas (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Carregar	BOOL	M10	Comando para carregar receita
IN	RecVel	INT	K1500	Velocidade da receita
IN	RecTemp	INT	K180	Temperatura da receita
IN	RecTempo	INT	K60	Tempo de ciclo da receita
OUT	TrabVel	INT	D50	Velocidade ativa
OUT	TrabTemp	INT	D51	Temperatura ativa
OUT	TrabTempo	INT	D52	Tempo ativo

Associar na IDE

Crie a instância Inst_FB_COPIA_RECEITA_3W. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

33 FB_COPIA_RECEITA_3W

2. Funcionamento, instrução por instrução

01	LD {Carregar}	Inicia uma condição lendo {Carregar}.
02	MOV {RecVel} {TrabVel}	Se habilitado, copia {RecVel} para {TrabVel}.
03	MOV {RecTemp} {TrabTemp}	Se habilitado, copia {RecTemp} para {TrabTemp}.
04	MOV {RecTempo} {TrabTempo}	Se habilitado, copia {RecTempo} para {TrabTempo}.

Exemplo IL com as associações padrão

```
LD M10
MOV K1500 D50
MOV K180 D51
MOV K60 D52
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

33 FB_COPIA_RECEITA_3W

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Carregar=1; RecVel=10; RecTemp=20; RecTempo=30	TrabVel=10; TrabTemp=20; TrabTempo=30
Carregar=0	TrabVel(t)=TrabVel(t-1)
Carregar=1; RecVel=40	TrabVel=40

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

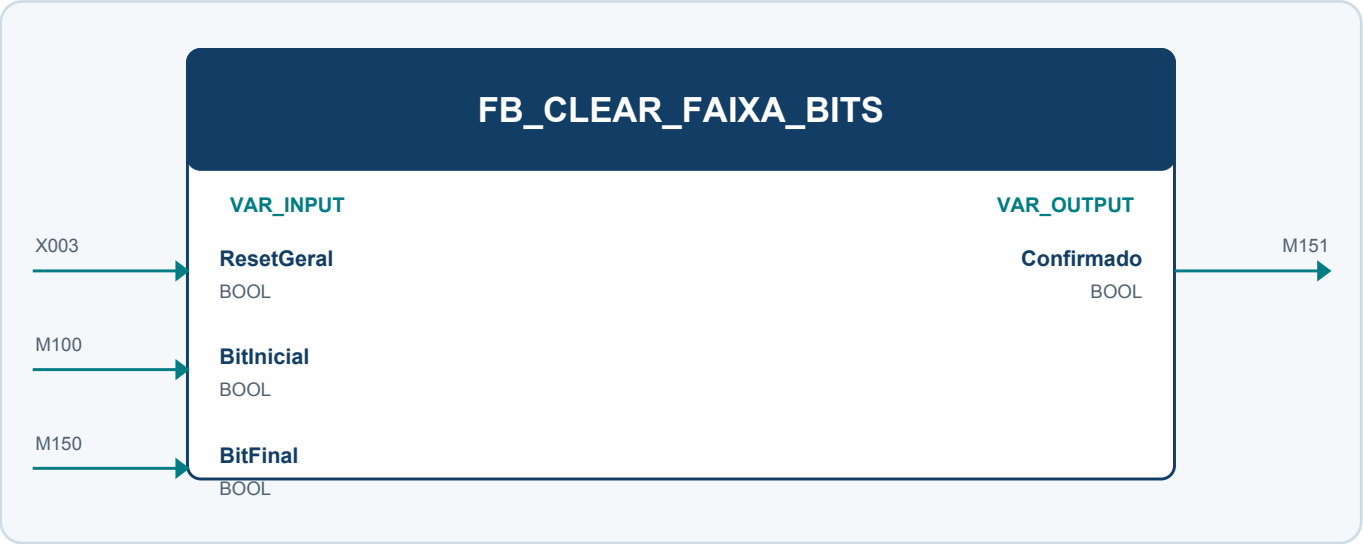
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

34 FB_CLEAR_FAIXA_BITS

Reset geral em zona de bits/alarmes (ZRST estruturado)

6.3 Blocos de Transferência e Receitas (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	ResetGeral	BOOL	X003	Botão de reconhecimento/reset
IN	BitInicial	BOOL	M100	Primeiro bit da faixa
IN	BitFinal	BOOL	M150	Último bit da faixa
OUT	Confirmado	BOOL	M151	Indica reset executado

Associar na IDE

Crie a instância Inst_FB_CLEAR_FAIXA_BITS. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

34 FB_CLEAR_FAIXA_BITS

2. Funcionamento, instrução por instrução

01	LD {ResetGeral}	Inicia uma condição lendo {ResetGeral}.
02	ZRST {BitInicial} {BitFinal}	Se habilitado, zera a faixa inclusiva {BitInicial}..{BitFinal}; confira cada endereço afetado.
03	OUT {Confirmado}	Escreve a condição atual em {Confirmado} a cada scan.

Exemplo IL com as associações padrão

```
LD X003
ZRST M100 M150
OUT M151
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

34 FB_CLEAR_FAIXA_BITS

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
ResetGeral=0	Confirmado=0
ResetGeral=1; BitInicial=M20; BitFinal=M22	M20=0; M21=0; M22=0; Confirmado=1
ResetGeral: 1 -> 0	Confirmado=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

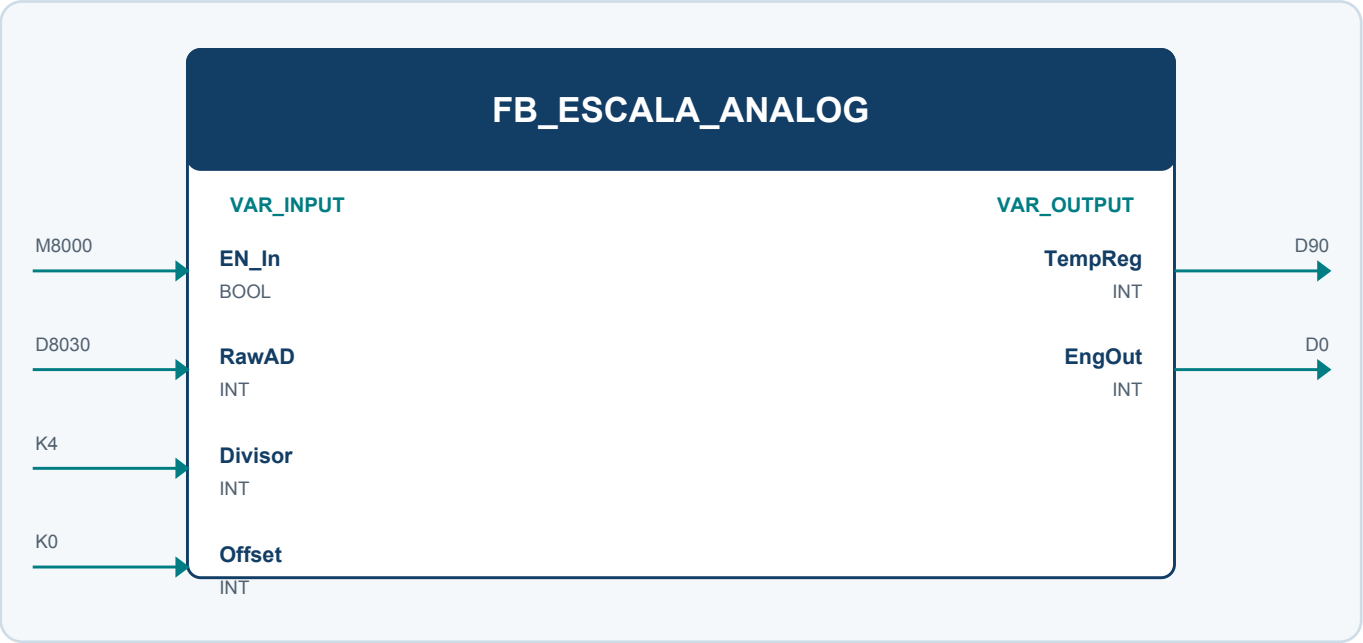
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

35 FB_ESCALA_ANALOG

Escalonamento linear de entrada analógica AD (Divisão + Offset)

7.2 Blocos de Função Analógicos e PID (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	EN_In	BOOL	M8000	Habilitação do bloco
IN	RawAD	INT	D8030	Registrador analógico bruto
IN	Divisor	INT	K4	Fator divisor de escala
IN	Offset	INT	K0	Offset de engenharia
OUT	TempReg	INT	D90	Registrador intermediário
OUT	EngOut	INT	D0	Valor escalonado final

Associar na IDE

Crie a instância Inst_FB_ESCALA_ANALOG. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

35 FB_ESCALA_ANALOG

2. Funcionamento, instrução por instrução

01	LD {EN_In}	Inicia uma condição lendo {EN_In}.
02	DIV {RawAD} {Divisor} {TempReg}	Se habilitado, divide {RawAD} por {Divisor}: quociente em {TempReg} e resto no D seguinte. Divisor zero é inválido.
03	ADD {TempReg} {Offset} {EngOut}	Se habilitado, calcula {TempReg} + {Offset} em {EngOut}; confira overflow de 16 bits.

Exemplo IL com as associações padrão

```
LD M8000
DIV D8030 K4 D90
ADD D90 K0 D0
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

35 FB_ESCALA_ANALOG

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
EN_In=1; RawAD=2000; Divisor=4; Offset=100	TempReg=500; EngOut=600
EN_In=1; RawAD=2001; Divisor=4; Offset=0	TempReg=500; EngOut=500; R=1
EN_In=0	EngOut(t)=EngOut(t-1)

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Reserve também o D seguinte a cada resultado de MUL/DIV. Um resultado dirigido a DA pode escrever o canal analógico adjacente; confira o mapa antes de usar saídas reais.

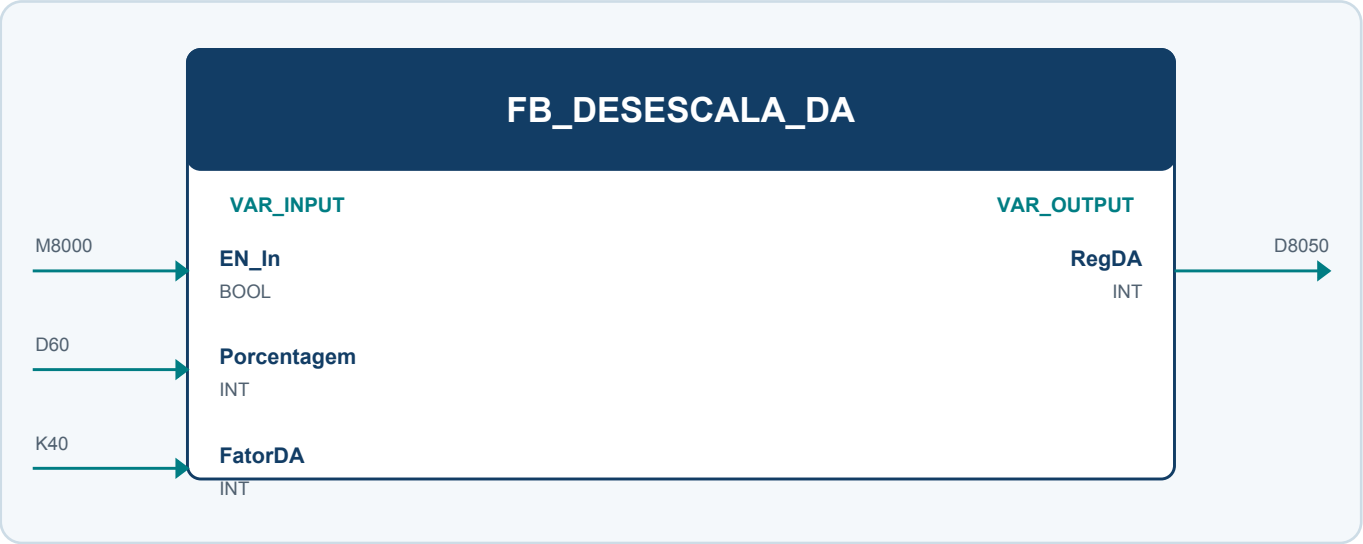
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

36 FB_DESESCALA_DA

Converte valor de engenharia (0..100%) para saída analógica DA (0..4000)

7.2 Blocos de Função Analógicos e PID (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	EN_In	BOOL	M8000	Habilita saída analógica
IN	Porcentagem	INT	D60	Comando 0..100%
IN	FatorDA	INT	K40	Multiplicador para 0..4000
OUT	RegDA	INT	D8050	Registrador analógico DA0 (D8050)

Associar na IDE

Crie a instância Inst_FB_DESESCALA_DA. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

36 FB_DESESCALA_DA

2. Funcionamento, instrução por instrução

01	LD {EN_In}	Inicia uma condição lendo {EN_In}.
02	MUL {Porcentagem} {FatorDA} {RegDA}	Se habilitado, multiplica {Porcentagem} por {FatorDA} e grava o par iniciado em {RegDA}.

Exemplo IL com as associações padrão

```
LD M8000
MUL D60 K40 D8050
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

36 FB_DESESCALA_DA

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
EN_In=1; Porcentagem=25; FatorDA=40	RegDA=1000; D+1=0
EN_In=1; Porcentagem=100; FatorDA=40	RegDA=4000; D+1=0
EN_In=0	RegDA(t)=RegDA(t-1)

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Reserve também o D seguinte a cada resultado de MUL/DIV. Um resultado dirigido a DA pode escrever o canal analógico adjacente; confira o mapa antes de usar saídas reais.

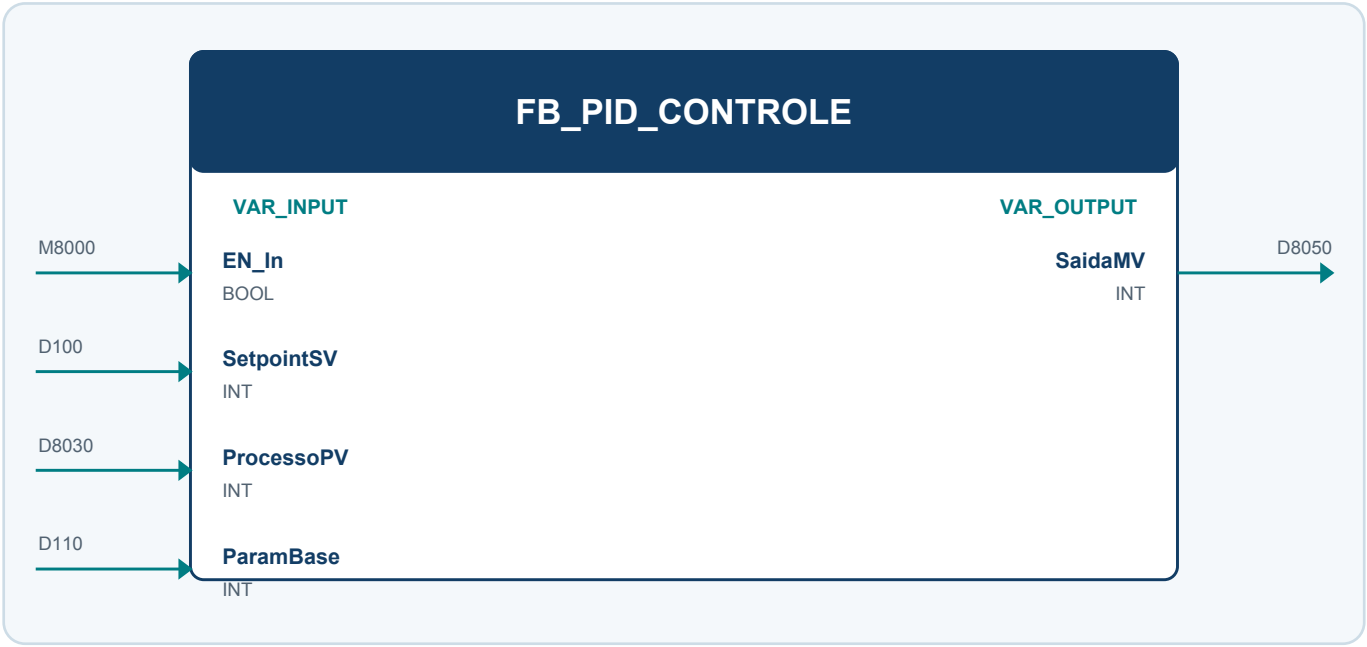
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

37 FB_PID_CONTROLE

Malha fechada de controle PID (SV / PV / Parâmetros -> MV)

7.2 Blocos de Função Analógicos e PID (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	EN_In	BOOL	M8000	Habilita malha PID
IN	SetpointSV	INT	D100	Valor desejado (Setpoint)
IN	ProcessoPV	INT	D8030	Variável de processo lida (AD)
IN	ParamBase	INT	D110	Bloco de parâmetros PID
OUT	SaidaMV	INT	D8050	Saída manipulada (DA)

Associar na IDE

Crie a instância Inst_FB_PID_CONTROLE. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

37 FB_PID_CONTROLE

2. Funcionamento, instrução por instrução

01	LD {EN_In}	Inicia uma condição lendo {EN_In}.
02	PID {SetpointSV} {ProcessoPV} {ParamBase} {SaidaMV}	Executa PID: SV={SetpointSV}, PV={ProcessoPV}, parâmetros={ParamBase}, MV={SaidaMV}. Dimensione parâmetros e limites antes do ensaio físico.

Exemplo IL com as associações padrão

```
LD M8000
PID D100 D8030 D110 D8050
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

37 FB_PID_CONTROLE

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
EN_In=0	PID: 0
EN_In=1; SetpointSV=ProcessoPV	PID(SetpointSV,ProcessoPV,ParamBase,SaidaMV)
EN_In=1; SetpointSV>ProcessoPV	PID: SV>PV; ParamBase; SaidaMV

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Não se pode deduzir MV apenas do sinal SV-PV: ganho, período, limites e estado interno fazem parte de ParamBase. Consulte a tabela PID da CPU antes de reservar a área e sintonizar a malha.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

38 FB_ALARME_ANALOG_HH_LL

Supervisão de limites analógicos Alto (High) e Baixo (Low) via ZCP

7.2 Blocos de Função Analógicos e PID (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	LimiteMin	INT	K500	Limite mínimo aceitável
IN	LimiteMax	INT	K3500	Limite máximo aceitável
IN	LeituraAD	INT	D8030	Canal analógico monitorado
OUT	FlagZonaBase	BOOL	M160	M160=Abaixo, M161=Normal, M162=Acima

Associar na IDE

Crie a instância Inst_FB_ALARME_ANALOG_HH_LL. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

38 FB_ALARME_ANALOG_HH_LL

2. Funcionamento, instrução por instrução

01	LD M8000	Inicia uma condição lendo M8000.
02	ZCP {LimiteMin} {LimiteMax} {LeituraAD} {FlagZonaBase}	Compara {LeituraAD} com [{LimiteMin}, {LimiteMax}]; grava três bits a partir de {FlagZonaBase}: abaixo, dentro, acima.

Exemplo IL com as associações padrão

```
LD M8000
ZCP K500 K3500 D8030 M160
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

38 FB_ALARME_ANALOG_HH_LL

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
LimiteMin=10; LimiteMax=20; LeituraAD=5	FlagZonaBase=1; +1=0; +2=0
LimiteMin=10; LimiteMax=20; LeituraAD=15	FlagZonaBase=0; +1=1; +2=0
LimiteMin=10; LimiteMax=20; LeituraAD=25	FlagZonaBase=0; +1=0; +2=1

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

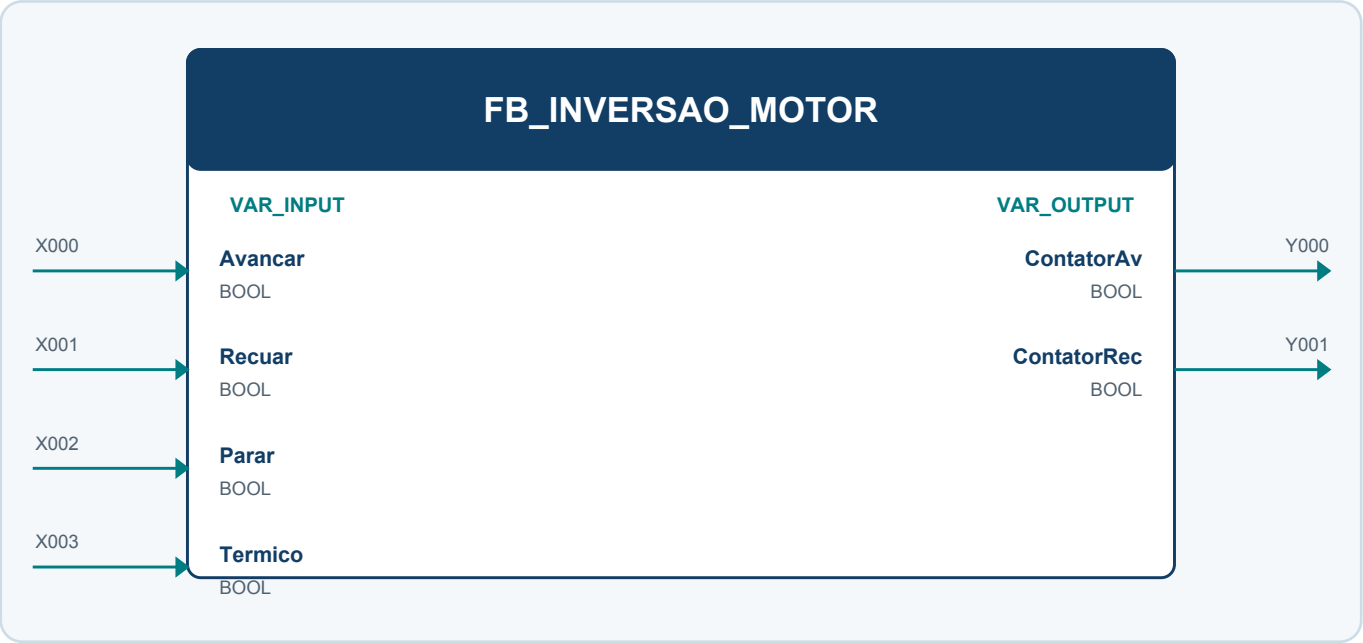
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

39 FB_INVERSAO_MOTOR

Partida reversora Avanço/Recuo com intertravamento elétrico cruzado

8.1 Partida e Reversão de Motores (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Avancar	BOOL	X000	Comando sentido horário
IN	Recuar	BOOL	X001	Comando sentido anti-horário
IN	Parar	BOOL	X002	Botão de parada
IN	Termico	BOOL	X003	Proteção térmica
OUT	ContatorAv	BOOL	Y000	Contator Avanço (K1)
OUT	ContatorRec	BOOL	Y001	Contator Recuo (K2)

Associar na IDE

Crie a instância Inst_FB_INVERSAO_MOTOR. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

39 FB_INVERSAO_MOTOR

2. Funcionamento, instrução por instrução

01	LD {Avancar}	Inicia uma condição lendo {Avancar}.
02	OR {ContatorAv}	Combina em paralelo: condição anterior OR {ContatorAv}.
03	ANI {Parar}	Exige também {Parar} falso; confira a polaridade do sinal real.
04	ANI {Termico}	Exige também {Termico} falso; confira a polaridade do sinal real.
05	ANI {ContatorRec}	Exige também {ContatorRec} falso; confira a polaridade do sinal real.
06	OUT {ContatorAv}	Escreve a condição atual em {ContatorAv} a cada scan.
07	LD {Recuar}	Inicia uma condição lendo {Recuar}.
08	OR {ContatorRec}	Combina em paralelo: condição anterior OR {ContatorRec}.
09	ANI {Parar}	Exige também {Parar} falso; confira a polaridade do sinal real.
10	ANI {Termico}	Exige também {Termico} falso; confira a polaridade do sinal real.
11	ANI {ContatorAv}	Exige também {ContatorAv} falso; confira a polaridade do sinal real.
12	OUT {ContatorRec}	Escreve a condição atual em {ContatorRec} a cada scan.

Exemplo IL com as associações padrão

```
LD X000
OR Y000
ANI X002
ANI X003
ANI Y001
OUT Y000
LD X001
OR Y001
ANI X002
ANI X003
ANI Y000
OUT Y001
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

39 FB_INVERSAO_MOTOR

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Avancar=1; Recuar=0; Parar=0; Termico=0	ContatorAv=1; ContatorRec=0
Avancar=0; Recuar=1; Parar=0; Termico=0; ContatorAv(t-1)=0	ContatorAv=0; ContatorRec=1
Parar=1	ContatorAv=0; ContatorRec=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

ANI exige bit 0 para liberar. Um botão NF pode produzir bit 1 em repouso dependendo da ligação: normalize o sinal e teste cabo rompido e comando simultâneo. A lógica apresentada não substitui circuito de segurança certificado.

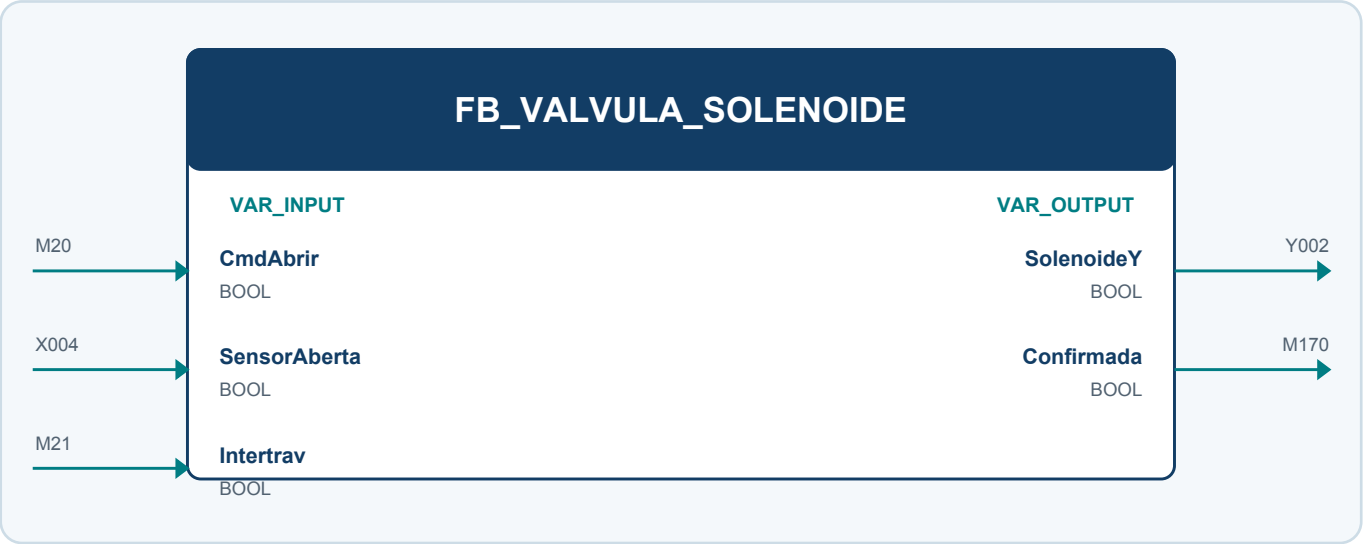
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

40 FB_VALVULA_SOLENOIDE

Controle de válvula solenoide com confirmação de sensor aberto/fechado

8.2 Válvulas, Cilindros e Segurança (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	CmdAbrir	BOOL	M20	Comando de abertura
IN	SensorAberta	BOOL	X004	Fim de curso válvula aberta
IN	Intertrav	BOOL	M21	Bloqueio de segurança
OUT	SolenoideY	BOOL	Y002	Saída da bobina solenoide
OUT	Confirmada	BOOL	M170	Válvula aberta e confirmada

Associar na IDE

Crie a instância Inst_FB_VALVULA_SOLENOIDE. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

40 FB_VALVULA_SOLENOIDE

2. Funcionamento, instrução por instrução

01	LD {CmdAbrir}	Inicia uma condição lendo {CmdAbrir}.
02	ANI {Intertrav}	Exige também {Intertrav} falso; confira a polaridade do sinal real.
03	OUT {SolenoideY}	Escreve a condição atual em {SolenoideY} a cada scan.
04	LD {SolenoideY}	Inicia uma condição lendo {SolenoideY}.
05	AND {SensorAberta}	Exige também {SensorAberta} verdadeiro.
06	OUT {Confirmada}	Escreve a condição atual em {Confirmada} a cada scan.

Exemplo IL com as associações padrão

```
LD M20
ANI M21
OUT Y002
LD Y002
AND X004
OUT M170
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

40 FB_VALVULA_SOLENOIDE

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
CmdAbrir=1; Intertrav=0; SensorAberta=0	SolenoideY=1; Confirmada=0
CmdAbrir=1; Intertrav=0; SensorAberta=1	SolenoideY=1; Confirmada=1
Intertrav=1	SolenoideY=0; Confirmada=0

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

ANI exige bit 0 para liberar. Um botão NF pode produzir bit 1 em repouso dependendo da ligação: normalize o sinal e teste cabo rompido e comando simultâneo. A lógica apresentada não substitui circuito de segurança certificado.

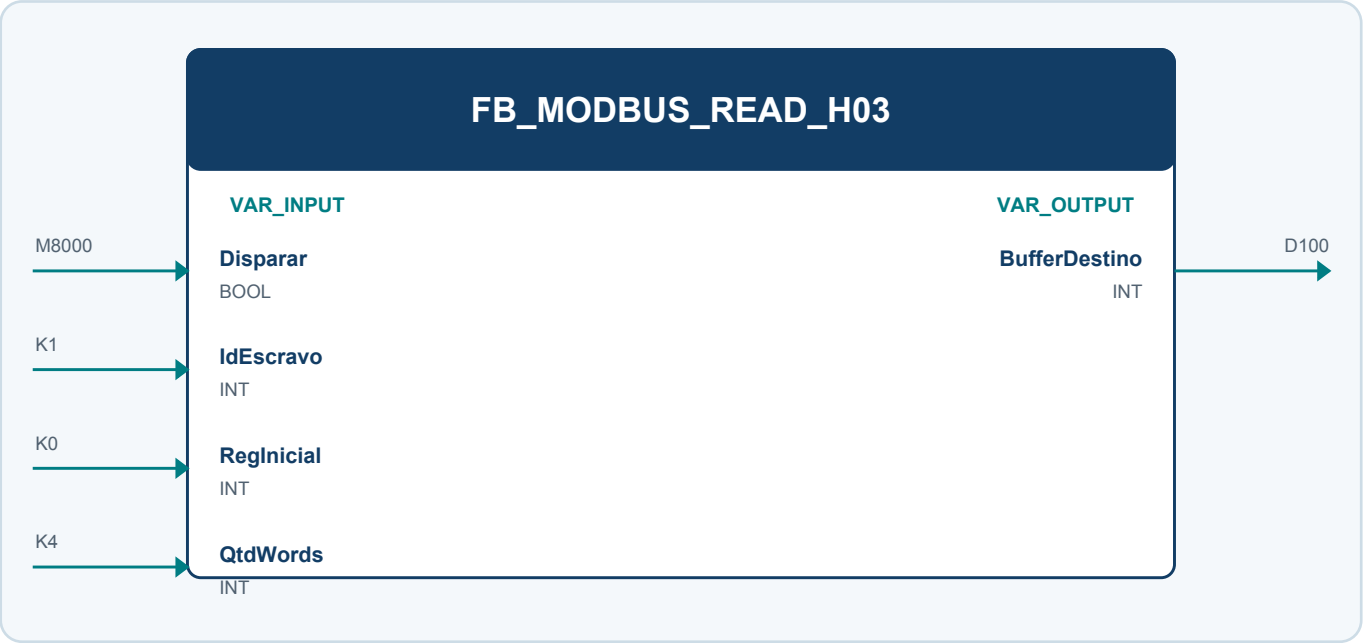
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

41 FB_MODBUS_READ_H03

Mestre Modbus RTU Função 03 — Leitura de Registradores Holding (ADPRW)

9.3 Blocos de Função Modbus e RTC (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Disparar	BOOL	M8000	Habilita leitura Modbus
IN	IdEscravo	INT	K1	Endereço do escravo (1..247)
IN	RegInicial	INT	K0	Endereço inicial no escravo
IN	QtdWords	INT	K4	Quantidade de registradores
OUT	BufferDestino	INT	D100	Primeiro registrador D de recepção

Associar na IDE

Crie a instância Inst_FB_MODBUS_READ_H03. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

41 FB_MODBUS_READ_H03

2. Funcionamento, instrução por instrução

01	LD {Disparar}	Inicia uma condição lendo {Disparar}.
02	ADPRW {IdEscravo} H3 {RegInicial} {QtdWords} {BufferDestino}	Comanda uma transação Modbus: estação={IdEscravo}, função=H3, endereço={RegInicial}, quantidade={QtdWords}, buffer={BufferDestino}. Disparo não equivale a resposta recebida.

Exemplo IL com as associações padrão

```
LD M8000
ADPRW K1 H3 K0 K4 D100
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

41 FB_MODBUS_READ_H03

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Disparar=0	ADPRW: 0
Disparar=1; IdEscravo=1; RegInicial=0; QtdWords=2	ADPRW(1,H3,0,2,BufferDestino)
IdEscravo=1; QtdWords=2	BufferDestino; D+1

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Confira baud rate, paridade, estação, terminação e mapa remoto. RegInicial/EndRemoto são operandos de protocolo, não números de tela presumidos. Leia o estado de conclusão/erro da CPU e confira resposta ou releitura; OUT StatusEnvio não é ACK.

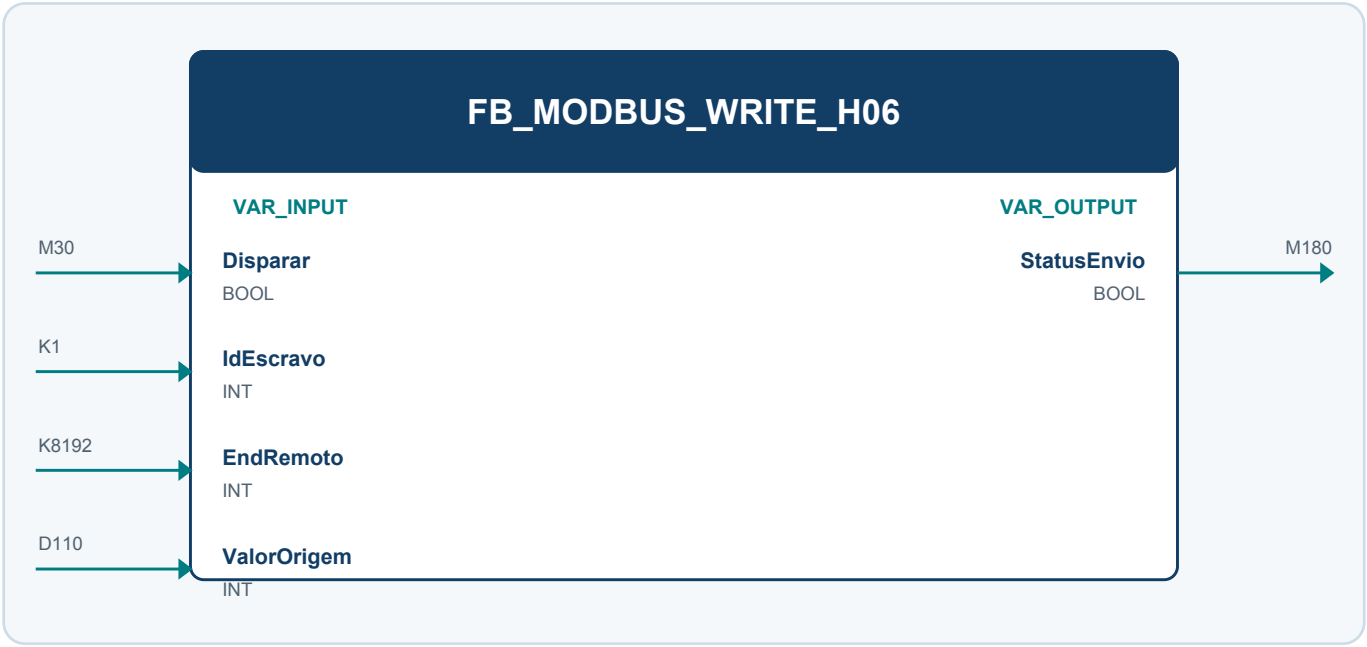
Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

42 FB_MODBUS_WRITE_H06

Mestre Modbus RTU Função 06 — Escrita de Registrador Único (ADPRW)

9.3 Blocos de Função Modbus e RTC (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Disparar	BOOL	M30	Comando de escrita Modbus
IN	IdEscravo	INT	K1	Endereço do escravo (1..247)
IN	EndRemoto	INT	K8192	Endereço do registrador no escravo
IN	ValorOrigem	INT	D110	Registrador local contendo o valor
OUT	StatusEnvio	BOOL	M180	Flag de disparo realizado

Associar na IDE

Crie a instância Inst_FB_MODBUS_WRITE_H06. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

42 FB_MODBUS_WRITE_H06

2. Funcionamento, instrução por instrução

01	LD {Disparar}	Inicia uma condição lendo {Disparar}.
02	ADPRW {IdEscravo} H6 {EndRemoto} K1 {ValorOrigem}	Comanda uma transação Modbus: estação={IdEscravo}, função=H6, endereço={EndRemoto}, quantidade=K1, buffer={ValorOrigem}. Disparo não equivale a resposta recebida.
03	OUT {StatusEnvio}	Escreve a condição atual em {StatusEnvio} a cada scan.

Exemplo IL com as associações padrão

```
LD M30
ADPRW K1 H6 K8192 K1 D110
OUT M180
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

42 FB_MODBUS_WRITE_H06

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Disparar=0	StatusEnvio=0
Disparar=1; IdEscravo=1; EndRemoto=0; ValorOrigem=123	ADPRW(1,H6,0,K1,ValorOrigem); StatusEnvio=1
Disparar=1; ACK=0	StatusEnvio=1

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

StatusEnvio é uma bobina comandada por Disparar; não comprova ACK nem conclusão da transação Modbus. Verifique os estados de comunicação aplicáveis.

Confira baud rate, paridade, estação, terminação e mapa remoto. RegInicial/EndRemoto são operandos de protocolo, não números de tela presumidos. Leia o estado de conclusão/erro da CPU e confira resposta ou releitura; OUT StatusEnvio não é ACK.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.

43 FB_RTC_LEITURA_RELOGIO

Leitura do Relógio de Tempo Real (TRD D8013..D8019) para bloco de registradores

9.3 Blocos de Função Modbus e RTC (FB)



1. Interface e associação dos pinos

Classe	Pin	Tipo	Padrão	Função do pino
IN	Habilitar	BOOL	M8000	Atualiza leitura do RTC a cada scan
OUT	BaseRtcD	INT	D200	Destino D..D+6 (Seg, Min, Hora, Dia, Mês, Ano, Sem)

Associar na IDE

Crie a instância Inst_FB_RTC_LEITURA_RELOGIO. Associe os pinos lógicos às tags do programa e altere os endereços padrão para uma faixa livre. Confira todas as saídas e áreas auxiliares, inclusive palavras adjacentes, antes de inserir e compilar.

43 FB_RTC_LEITURA_RELOGIO

2. Funcionamento, instrução por instrução

01	LD {Habilitar}	Inicia uma condição lendo {Habilitar}.
02	TRD {BaseRtcD}	Se habilitado, lê o RTC para sete D consecutivos desde {BaseRtcD}; confira formato e ano no equipamento.

Exemplo IL com as associações padrão

```
LD M8000
TRD D200
END
```

Exemplo isolado: END encerra este programa de estudo. Ao inserir um bloco em um programa existente, preserve a ordem das redes e o END único do programa principal. Não grave os endereços de exemplo sem revisar a instalação.

43 FB_RTC_LEITURA_RELOGIO

3. Roteiro de ensaio e resultados a conferir

1. Use uma cópia do projeto, sem comandos físicos habilitados. Reserve os dispositivos de cada pino e os auxiliares. Registre o estado inicial antes de cada caso independente.

2. Compile o IL expandido e confira a tabela de instruções. No simulador, altere entradas manualmente e execute scans controlados; mantenha a memória entre casos apenas quando indicado por (t-1).

Estímulo / condição inicial	Conferir no IL / monitor
Habilitar=0	TRD: 0
Habilitar=1	TRD(BaseRtcD); D..D+6
Habilitar=1; t: 1 s -> 2 s	BaseRtcD; D+1..D+6

0 e 1 representam estados lógicos; t é tempo ou scan conforme o caso, (t-1) é o estado anterior e R é o resto da divisão. Nos blocos físicos, a coluna indica a instrução comandada: não é um resultado medido. Vetores são roteiros de ensaio derivados do IL atual.

Limites da implementação atual

Verificação sugerida: aplique os valores padrão somente em projeto de ensaio; alterne cada entrada, observe as saídas e confira os dispositivos auxiliares no IL. Para blocos de movimento, PID, analógicas e comunicação, a prévia não valida operação física.

Se o resultado divergir

Confira primeiro associação de pinos e polaridade; depois, a ordem LD/AND/OR e os escritores posteriores do mesmo dispositivo. Para temporizadores/contadores, confira dispositivo, preset, base de tempo e reset. Para números, confira sinal, largura de palavra, divisor e memória adjacente. Preserve o diagnóstico e o IL realmente compilado.