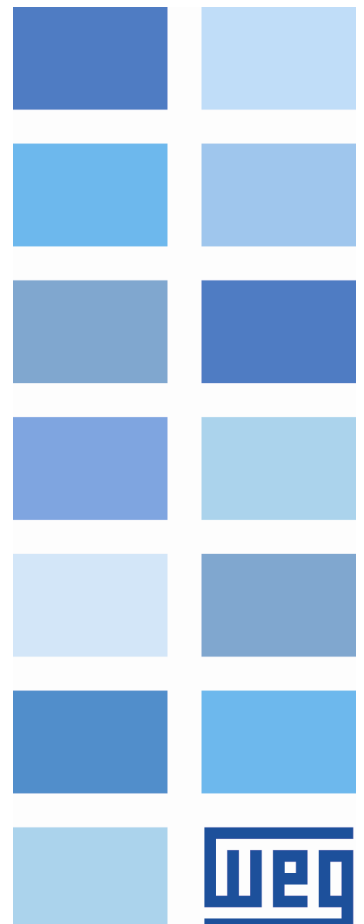


SoftPLC

CFW700

Manual do Usuário

Idioma: Português





Manual do Usuário SoftPLC

Série: CFW700

Idioma: Português

Nº do Documento: 10000823401 / 00

Data da Publicação: 04/2010

Sumário

ABREVIações E DEFINIções.....	5
REPRESENTAção NUMÉRICA.....	5
1 INTRODUÇÃO À SOFTPLC.....	6
1.1 SÍMBOLO DOS TIPOS DE DADOS	6
2 MEMÓRIA DA SOFTPLC.....	7
2.1 MEMÓRIA	7
2.2 MEMÓRIA DE DADOS	7
2.2.1 Constantes	7
2.2.2 Entradas e Saídas Físicas (Hardware).....	7
2.2.3 Marcadores Voláteis (Variáveis).....	8
2.2.4 Marcadores do Sistema.....	8
2.2.5 Parâmetros.....	10
2.3 MODBUS.....	11
2.3.1 Endereços SoftPLC no protocolo Modbus	11
2.3.2 Protocolo	11
3 RESUMO DOS BLOCOS DE FUNÇÃO.....	12
3.1 CONTATOS.....	12
3.1.1 Contato Normalmente Aberto – NO CONTACT.....	12
3.1.2 Contato Normalmente Fechado – NC CONTACT.....	12
3.1.3 Lógicas “E (AND)” com Contatos	12
3.1.4 Lógicas “OU (OR)” com Contatos.....	12
3.2 BOBINAS.....	13
3.2.1 Bobina Normal – COIL.....	13
3.2.2 Bobina Negada – NEG COIL	13
3.2.3 Seta Bobina – SET COIL.....	13
3.2.4 Reseta Bobina – RESET COIL	13
3.2.5 Bobina de Transição Positiva – PTS COIL	13
3.2.6 Bobina de Transição Negativa – NTS COIL.....	13
3.3 BLOCOS DE MOVIMENTO	13
3.3.1 Referência de Velocidade e/ou Torque – REF	13
3.4 BLOCOS DE CLP.....	14
3.4.1 Temporizador – TON	14
3.4.2 Contador Incremental – CTU.....	14
3.4.3 Controlador Proporcional-Integral-Derivativo – PID	14
3.4.4 Filtro Passa-Baixa ou Passa-Alta – FILTER	15
3.5 BLOCOS DE CÁLCULO	15
3.5.1 Comparador – COMP	15
3.5.2 Operação Matemática – MATH	16
3.5.3 Função Matemática – FUNC.....	16
3.5.4 Saturador – SAT	17
3.6 BLOCOS DE TRANSFERÊNCIA	17
3.6.1 Transfere Dados – TRANSFER	17
3.6.2 Converte de Inteiro (16 bits) para Ponto Flutuante – INT2FL	17
3.6.3 Gerador de Falha ou Alarme do Usuário – USERERR	18
3.6.4 Converte de Ponto Flutuante para Inteiro (16 bits) – FL2INT	18
3.6.5 Transfere Dados Indireta – IDATA	18
3.6.6 Multiplexador – MUX	19
3.6.7 Demultiplexador – DMUX.....	19
4 PARAMETRIZAÇÃO DO INVERSOR.....	20
4.1 SÍMBOLOS PARA DESCRIÇÃO DAS PROPRIEDADES.....	20
4.2 PARÂMETROS DE CONFIGURAÇÃO DO CFW700.....	20
4.3 PARÂMETROS EXCLUSIVOS DA SOFTPLC.....	21
P1000 – ESTADO DA SOFTPLC	21

Sumário

P1001 – COMANDO PARA SOFTPLC	21
P1002 – TEMPO CICLO DE SCAN	21
P1003 – APLICAÇÃO NA SOFTPLC	22
P1010 ATÉ P1059 – PARÂMETROS SOFTPLC	22
5 RESUMO DAS PRINCIPAIS FUNÇÕES DO WLP.....	23
5.1 PROJETO – NOVO	23
5.2 PROJETO – ABRIR	23
5.3 PROJETO – PROPRIEDADES.....	23
5.4 EXIBIR – INFORMAÇÕES DA COMPILAÇÃO	24
5.5 EXIBIR – CONFIGURAÇÃO DOS PARÂMETROS DO USUÁRIO	24
5.6 CONSTRUIR – COMPILAR.....	25
5.7 COMUNICAÇÃO – CONFIGURAÇÃO	25
5.8 COMUNICAÇÃO – DOWNLOAD.....	26
6 FALHAS, ALARMES E POSSÍVEIS CAUSAS.....	27

SOBRE O MANUAL

Este manual fornece a descrição necessária para a operação do inversor de frequência CFW700 utilizando o módulo de programação do usuário, denominado SoftPLC. Este manual deve ser utilizado em conjunto com manual do usuário do CFW700 e do software WLP.

ABREVIações E DEFINIções

CLP	Controlador Lógico Programável
CRC	Cycling Redundancy Check
RAM	Random Access Memory
WLP	Software de Programação em Linguagem Ladder
USB	Universal Serial Bus

REPRESENTAÇÃO NUMÉRICA

Números decimais são representados através de dígitos sem sufixo. Números hexadecimais são representados com a letra 'h' depois do número.

1 INTRODUÇÃO À SOFTPLC

A SoftPLC é um recurso que incorpora ao CFW700 as funcionalidades de um CLP, agregando flexibilidade ao produto e permitindo que o usuário desenvolva seus próprios aplicativos (programas do usuário).

As principais características da SoftPLC são:

- Programação em “Linguagem Ladder” utilizando o software WLP;
- Acesso a todos os Parâmetros e I/O's do CFW700;
- 50 parâmetros configuráveis para uso do usuário;
- Blocos de CLP, Matemáticos e de Controle;
- Transferência e monitoração *on-line* do software aplicativo via RS-232;
- Transferência do software aplicativo instalado ao PC dependendo de senha;
- Armazenamento do software aplicativo em memória FLASH interna;
- Execução diretamente em RAM.

1.1 SÍMBOLO DOS TIPOS DE DADOS

%KW	constantes do tipo word (16 bits)
%KF	constantes do tipo float (32 bits, ponto flutuante)
%MX	marcadores de bit
%MW	marcadores de word (16 bits)
%MF	marcadores de float (32 bits, ponto flutuante)
%SX	marcadores de bit de sistema
%SW	marcadores de word do sistema (16 bits)
%IX	entradas digitais
%IW	entradas analógicas (16 bits)
%QX	saídas digitais
%QW	saídas analógicas (16 bits)

2 MEMÓRIA DA SOFTPLC

2.1 MEMÓRIA

- RAM SoftPLC: 4536 bytes
- FLASH SoftPLC: 32768 bytes



NOTA!

O aplicativo SoftPLC armazenado em FLASH é executado em RAM. Por este motivo, sempre que o aplicativo for maior do que 4536 bytes, o ciclo de scan fica mais lento, devido aos carregamentos da FLASH para a RAM.

2.2 MEMÓRIA DE DADOS

Na SoftPLC, a área de memória de dados (variáveis do usuário) e de programa é compartilhada. Por isso um aplicativo pode variar o tamanho total em função da quantidade de variáveis utilizadas pelo usuário.

Os marcadores de bit, word e float são alocados de acordo com o **ÚLTIMO** endereço utilizado no aplicativo, ou seja, quanto maior for este último endereço, maior será a área alocada. Por isso, é recomendado ao usuário utilizar os marcadores de maneira **SEQUENCIAL**.

As constantes word e float também utilizam espaço de programa.

2.2.1 Constantes

Tabela 2.1: Mapa de memória das constantes

Símb.	Descrição	Bytes
%KW	Constantes Word (16 bits)	Depende da quantidade de constantes word diferentes. Ex: Se foram utilizados as: - %KW: 327 = 2 bytes - %KW: 5; 67 = 4 bytes - %KW: 13; 1000; 43; 4 = 6 bytes
%KF	Constantes Float (32 bits – IEEE)	Depende da quantidade de constantes float diferentes. Ex: Se foram utilizados as: - %KF: -0,335 = 4 bytes - %KF: 5,1; 114,2 = 8 bytes - %KF: 0,0; 115,3; 0,0; 13,333 = 12 bytes

2.2.2 Entradas e Saídas Físicas (Hardware)

Tabela 2.2: Mapa de memória dos I/O's

Símb.	Descrição	Faixa	Bytes
%IX	Entradas Digitais	1 ... 8	2
%QX	Saídas Digitais	1 ... 5	2
%IW	Entradas Analógicas	1 ... 2	4
%QW	Saídas Analógicas	1 ... 2	4



NOTA!

Os valores das Entradas Analógicas (%IW) e das Saídas Analógicas (%QW) lidos e escritos respectivamente via SoftPLC, respeitam os seus ganhos (P232, P237: %IW1–%IW2 e P252, P255: %QW1–%QW2) e offsets (P234, P239: %IW1–%IW2).



NOTA!

Os valores lidos ou escritos via SoftPLC obedecem as seguintes regras, respeitando-se os parâmetros relativos aos sinais das entradas e saídas analógicas (P233, P238: %IW1–%IW2 e P253, P256: %QW1–%QW2):

- Opção: 0 a 10V/20mA
 - 0V ou 0mA = 0
 - 10V ou 20mA = 32767
- Opção: 4 a 20mA
 - 4mA = 0
 - 20mA = 32767
- Opção: 10V/20mA a 0
 - 10V ou 20mA = 0
 - 0V ou 0mA = 32767
- Opção: 20 a 4mA
 - 20mA = 0
 - 4mA = 32767
- Opção: -10 a +10V
 - -10V = -32768 (ou 32768 para parâmetro sem sinal)
 - -5V = -16384 (ou 49152 para parâmetro sem sinal)
 - 0 = 0
 - +10V = 32767

2.2.3 Marcadores Voláteis (Variáveis)

Consistem em variáveis que podem ser utilizadas pelo usuário para executar as lógicas do aplicativo. Podem ser marcadores de bit (1 bit), marcadores de word (16 bits) ou marcadores de float (32 bits – IEEE).

Tabela 2.3: Mapa de memória dos marcadores voláteis

Símb.	Descrição	Faixa	Quantidade de Bytes Alocados
%MX	Marcadores de Bit	5000 ... 6099	Depende do último marcador utilizado. São organizados de 2 em 2 bytes. Ex: - último marcador: %MX5000 = 2 bytes - último marcador: %MX5014 = 2 bytes - último marcador: %MX5016 = 4 bytes - último marcador: %MX5039 = 6 bytes
%MW	Marcadores de Word	8000 ... 8199	Depende do último marcador utilizado. Ex: - último marcador: %MX8000 = 2 bytes - último marcador: %MX8001 = 4 bytes - último marcador: %MX8007 = 16 bytes
%MF	Marcadores de Float	9000 ... 9199	Depende do último marcador utilizado. Ex: - último marcador: %MX9000 = 4 bytes - último marcador: %MX9001 = 8 bytes - último marcador: %MX9007 = 32 bytes



NOTA!

Para minimizar o tamanho do aplicativo, utilizar marcadores de forma seqüencial.

Ex:

- Marcadores de bit: %MX5000, %MX5001, %MX5002, ...
- Marcadores de word: %MW8000, %MW8001, %MW8002, ...
- Marcadores de float: %MF9000, %MF9001, %MF9002, ...

2.2.4 Marcadores do Sistema

Consistem em variáveis especiais que permitem ao usuário ler e alterar dados do inversor que podem ou não estar disponíveis nos parâmetros. Podem ser: marcadores de bit do sistema (1 bit) ou marcadores de word do sistema (16 bits).

Tabela 2.4.a: Mapa de Memória dos Marcadores de Bit do Sistema - Ímpares

Símb.	Descrição	Faixa	Bytes
Tipo	Bits do Sistema	3000 ... 3040	4 bytes
%SX	<i>Escrita/Comando (Ímpares)</i>		
3001	Habilita Geral	0: Desabilita geral o inversor, interrompendo a alimentação para o motor. 1: Habilita geral o inversor, permitindo a operação do motor.	
3003	Gira/Pára	0: Pára motor por rampa de desaceleração. 1: Gira motor de acordo com a rampa de aceleração até atingir o valor da referência de velocidade.	
3005	Sentido de Giro	0: Gira motor no sentido anti-horário. 1: Gira motor no sentido horário.	
3007	JOG	0: Desabilita a função JOG. 1: Habilita a função JOG.	
3009	LOC/REM	0: Inversor vai para o modo local. 1: Inversor vai para o modo remoto.	
3011	Reset de Falhas	0: Sem função. 1: Se em estado de falha, executa o reset do inversor. NOTA: Ao ser executado este comando, o inversor e o Aplicativo SoftPLC serão reinicializados. O mesmo vale para o comando de Reset via HMI.	
3021	Ativa 2ª Rampa	0: Valores para aceleração e desaceleração do motor são os da 1ª Rampa (P0100 e P0101). 1: Valores para aceleração e desaceleração do motor são os da 2ª Rampa (P0102 e P0103). Obs.: Programar P0105 em 6 para habilitar a seleção via SoftPLC.	

Tabela 2.4.b: Mapa de Memória dos Marcadores de Bit do Sistema – Pares

Símb.	Descrição	Faixa	Bytes
Tipo	Bits do Sistema	3000 ... 3040	4 bytes
%SX	<i>Leitura/Estado (Pares)</i>		
3000	Habilitado Geral	0: Inversor está desabilitado geral. 1: Inversor está habilitado geral e pronto para girar motor.	
3002	Motor Girando (RUN)	0: Motor está parado. 1: Inversor está girando o motor à velocidade de referência, ou executando rampa de aceleração ou desaceleração.	
3004	Sentido de Giro	0: Motor girando no sentido anti-horário. 1: Motor girando no sentido horário.	
3006	JOG	0: Função JOG inativa. 1: Função JOG ativa.	
3008	LOC/REM	0: Inversor em modo local. 1: Inversor em modo remoto.	
3010	Em Falha	0: Inversor não está no estado de falha. 1: Alguma falha registrada pelo inversor. Obs.: O número da falha pode ser lido através do parâmetro P0049 – Falha Atual.	
3012	Em Subtensão	0: Sem subtensão. 1: Com subtensão.	
3014	Modo de Operação do PID	0: Em modo manual (função PID). 1: Em modo automático (função PID).	
3016	Em Alarme	0: Inversor não está no estado de alarme. 1: Inversor está no estado de alarme. Obs.: O número do alarme pode ser lido através do parâmetro P0048 – Alarme Atual.	
3018	Em Modo de Configuração	0: Inversor operando normalmente. 1: Inversor em modo de configuração. Indica uma condição especial na qual o inversor não pode ser habilitado: <ul style="list-style-type: none"> • Executando rotina de auto-ajuste. • Executando rotina de <i>start-up</i> orientado. • Executando função <i>copy</i> da HMI. • Executando rotina auto-guiada do cartão de memória flash. • Possui incompatibilidade de parametrização. Obs.: É possível obter a descrição exata do modo especial de operação no parâmetro P0692.	
3020	Rampa Ativa	0: Indica que a 1ª Rampa está ativa. 1: Indica que a 2ª Rampa está ativa.	
3032	Tecla Start (I)	0: Não pressionada.	
3034	Tecla Stop (O)	1: Pressionada por 1 ciclo de scan	
3036	Tecla Sentido de Giro (U)		
3038	Tecla Local/Remoto		
3040	Tecla JOG	0: Não pressionada. 1: Pressionada	

Tabela 2.5: Mapa de Memória dos Marcadores de Word do Sistema

Símb.	Descrição	Faixa	Bytes
%SW	Words do Sistema	3300 ... 3324	26 bytes
	<i>Marcadores de Leitura/Status (Pares)</i>		
	3300	Velocidade do motor [13 bits]	
	3302	Velocidade síncrona do motor [rpm]	
	3304	Velocidade do motor [rpm]	
	3306	Referência de velocidade [rpm]	
	3308	Alarme	
	3310	Falha	
	3312	Corrente de fluxo Id [13 bits]	
	3314	Corrente de torque Iq [13 bits]	
	3316	Referência da corrente de fluxo Id* [13 bits]	
	3318	Referência da corrente de torque Iq* [13 bits]	
	3320	Corrente nominal (HD) do inversor [A x10]	
	3322	Corrente do motor sem filtro (P003) [A x10]	
	3324	Torque do motor sem filtro [% x10]	


NOTA!

Os marcadores de word do sistema %SW3300 e %SW3301 utilizam uma resolução de 13 bits (8192 → 0 à 8191), que representa a velocidade síncrona do motor. Assim, para um motor de VI pólos (isto significa uma velocidade síncrona de 1200rpm) se a referência de velocidade via SoftPLC (%SW3301) for de 4096, o motor irá girar em 600 rpm.


NOTA!

Equação para o cálculo do valor da velocidade do motor em rpm:

$$\text{Velocidade em rpm} = \frac{\text{velocidade síncrona em rpm} \times \text{velocidade em 13 bits}}{8192}$$

2.2.5 Parâmetros

Os parâmetros P1011 a P1059 somente aparecem na HMI do CFW700 quando existe algum aplicativo (programa do usuário) válido contido na memória, ou seja, P1000 > 0.

Tabela 2.6: Mapa de Memória dos Parâmetros

Símb.	Descrição	Faixa	Bytes
%PD	Parâmetros do Drive (ver manual do CFW700)	0... 999	
%PW	Parâmetros do Sistema	1000 ... 1003	8 bytes
	P1000: Estado da SoftPLC [Parâmetro de Leitura]	0: Sem Aplicativo 1: Instal. Aplic. 2: Aplic. Incomp. 3: Aplic. Parado 4: Aplic. Rodando	
	P1001: Comando para a SoftPLC	0: Pára Aplic. 1: Executa Aplic. 2: Exclui Aplic.	
	P1002: Tempo Ciclo de Scan [ms] [Parâmetro de Leitura]		
	P1003: Aplicação na SoftPLC	0: Usuário 1: Regulador PID 2: Potenciômetro Eletrônico (PE) 3: Multispeed 4: Comando a Três Fios (Start/Stop) 5: Comando Avanço e Retorno	
%UW	Parâmetros do Usuário	P1010 ... P1059	100 bytes

2.3 MODBUS

2.3.1 Endereços SoftPLC no protocolo Modbus

Tabela 2.7: Faixa de Endereços SoftPLC x Modbus

Símb.	Descrição	SoftPLC	Modbus
%IX	Entradas Digitais	1 ... 8	2201...2208
%QX	Saídas Digitais	1 ... 5	2401...2405
%IW	Entradas Analógicas	1 ... 2	2601...2602
%QW	Saídas Analógicas	1 ... 2	2801...2802



NOTA!

Todos os demais tipos de dados possuem endereços do usuário (SoftPLC) iguais aos endereços Modbus. Ex: %PW100 = endereço Modbus 100; %MX5000 = endereço Modbus 5000; %SW3308 = endereço Modbus 3308.

2.3.2 Protocolo

Consultar Manual da Comunicação Serial RS232 / RS485, capítulo referente ao Protocolo Modbus.

3 RESUMO DOS BLOCOS DE FUNÇÃO

Neste capítulo será apresentado um resumo dos blocos de funções que estão disponíveis para a programação do usuário.

3.1 CONTATOS

Carregam para a pilha o conteúdo de um dado programado (0 ou 1), que pode ser do tipo:

- %MX: Marcador de Bit
- %IX: Entrada Digital
- %QX: Saída Digital
- %UW: Parâmetro do Usuário
- %SX: Marcador de Bit do Sistema – Leitura

3.1.1 Contato Normalmente Aberto – NO CONTACT



Menu: *Inserir-Contatos-NO CONTACT.*

Ex: Envia para a pilha o conteúdo do marcador de bit 5000.

3.1.2 Contato Normalmente Fechado – NC CONTACT

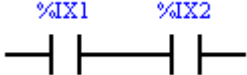
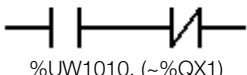


Menu: *Inserir-Contatos-NC CONTACT.*

Ex: Envia para a pilha o conteúdo negado da saída digital 1.

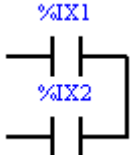

3.1.3 Lógicas “E (AND)” com Contatos

Quando os contatos estão em série, uma lógica “E” é executada entre eles armazenando o resultado na pilha. Exemplos:

Exemplo	Tabela Verdade		
 $%IX1 \cdot %IX2$	%IX1	%IX2	Pilha
	0	0	0
	0	1	0
	1	0	0
	1	1	1
 $%UW1010 \cdot (\sim \%QX1)$	%UW1010	%QX1	Pilha
	0	0	0
	0	1	0
	1	0	1
	1	1	0

3.1.4 Lógicas “OU (OR)” com Contatos

Quando os contatos estão em paralelo, uma lógica “OU” é executada entre eles armazenando o resultado na pilha. Exemplos:

Exemplo	Operação	Tabela Verdade		
 $%IX1 + %IX2$		%IX1	%IX2	Pilha
		0	0	0
		0	1	1
		1	0	1
		1	1	1
 $%UW1010 + (\sim \%QX1)$		%UW1010	%QX1	Pilha
		0	0	1
		0	1	0
		1	0	1
		1	1	1

Resumo dos Blocos de Função


3.2 BOBINAS

Salvam o conteúdo da pilha no dado programado (0 ou 1), que pode ser do tipo:

- %MX: Marcador de Bit
- %QX: Saída Digital
- %UW: Parâmetro do Usuário
- %SX: Marcador de Bit do Sistema – Escrita

É permitido adicionar bobinas em paralelo na última coluna.

3.2.1 Bobina Normal – COIL

%MX5001

Menu: *Inserir-Bobinas-COIL*
Ex: Seta o marcador de bit 5001 com o conteúdo da pilha

3.2.2 Bobina Negada – NEG COIL

%QX2


Menu: *Inserir-Bobinas-NEG COIL*
Ex: Seta a saída digital 2 com o conteúdo negado da pilha

3.2.3 Seta Bobina – SET COIL


%UW1011

Menu: *Inserir-Bobinas-SET COIL*
Ex: Seta o parâmetro do usuário **P1011** se o conteúdo da pilha não for 0


3.2.4 Reseta Bobina – RESET COIL

%UW1011

Menu: *Inserir-Bobinas-RESET COIL*
Ex: Reseta o parâmetro do usuário **P1011** se o conteúdo da pilha não for 0

3.2.5 Bobina de Transição Positiva – PTS COIL

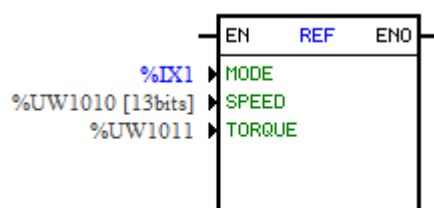
%MX5002

Menu: *Inserir-Bobinas-PTS COIL*
Ex: Seta o marcador de bit 5002 durante 1 ciclo de varredura, se for detectado uma transição de 0 para 1 no conteúdo da pilha

3.2.6 Bobina de Transição Negativa – NTS COIL

%SX3011

Menu: *Inserir-Bobinas-NTS COIL*
Ex: Seta o marcador de bit do sistema 3011 durante 1 ciclo de varredura, se for detectado uma transição de 1 para 0 no conteúdo da pilha

3.3 BLOCOS DE MOVIMENTO

3.3.1 Referência de Velocidade e/ou Torque – REF



Menu: *Inserir-Blocos de Função-Movimento-REF*

Entrada:

EN: Habilita o bloco

Saída:

ENO: Vai para 1 quando EN ≠ 0 e Sem erro

Propriedades:

MODE: 0=Modo Velocidade, 1=Modo Torque

SPEED: Referência de velocidade [RPM ou 13 Bits]

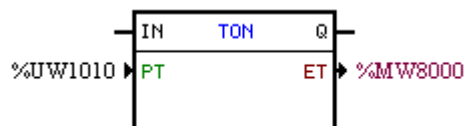
TORQUE: Referência de torque [13 Bits]

Resumo dos Blocos de Função

No exemplo acima, se a entrada EN estiver ativa, a entrada digital 1 estiver desligada, o bloco irá gerar uma referência de velocidade conforme o parâmetro do usuário P1010 na unidade 13 Bits. Se não **houver** erro (exemplo: inversor desabilitado), a saída ENO vai para 1.

3.4 BLOCOS DE CLP

3.4.1 Temporizador – TON



Menu: *Inserir-Blocos de Função-CLP-TON*

Entrada:

IN: Habilita o bloco

Saída:

Q: Vai para 1 quando $IN \neq 0$ e $ET \geq PT$

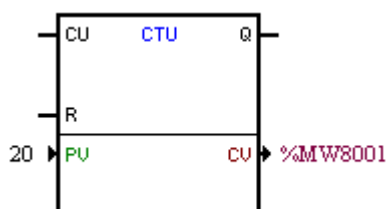
Propriedades:

PT: Tempo programado (*Preset Time*)

ET: Tempo decorrido (*Elapsed Time*)

No exemplo acima, se a entrada IN estiver ativa e o conteúdo do marcador de word 8000 for maior ou igual ao conteúdo do parâmetro do usuário P1010, a saída Q é setada.

3.4.2 Contador Incremental – CTU



Menu: *Inserir-Blocos de Função-CLP-CTU*

Entradas:

CU: Captura as transições de 0 para 1 nesta entrada (*Counter Up*)

R: Reseta CV

Saída:

Q: Vai para 1 quando $CV \geq PV$

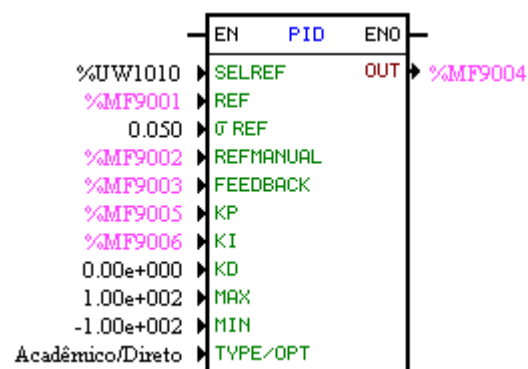
Propriedades:

PV: Valor programado (*Preset Value*)

CV: Valor de Contagem (*Counter Value*)

No exemplo acima, se o conteúdo do marcador de word 8001 for maior ou igual a 20, a saída Q é setada.

3.4.3 Controlador Proporcional-Integral-Derivativo – PID



Menu: *Inserir-Blocos de Função-CLP-PID*

Entradas:

EN: Habilita o bloco

Saída:

ENO: Imagem da entrada EN

Propriedades:

TS: Período de amostragem

SELREF: Referência automática/manual

REF: Referência automática

δREF: Constante de tempo do filtro da referência automática

REFMANUAL: Referência manual

FEEDBACK: Realimentação do processo

KP: Ganho proporcional

KI: Ganho integral

KD: Ganho derivativo

MAX: Valor máximo da saída

MIN: Valor mínimo da saída

TYPE: Acadêmico/paralelo

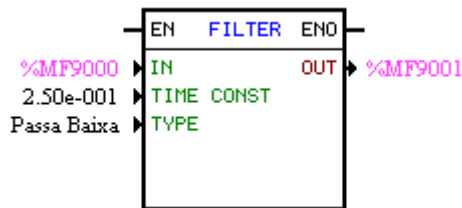
OPT: Direto/reverso

OUT: Saída do controlador

No exemplo acima, se a entrada EN estiver ativa, o controlador começa o seu trabalho. O conteúdo do parâmetro do usuário P1010 seleciona a referência que está ativa, ou seja, se é o marcador de float 9001 (referência automática) ou 9003 (referência manual). Para a referência automática há um filtro de 0.05s. Como o

ganho derivativo está fixo em 0, isto indica que o PID foi transformado para um PI. O valor da saída de controle OUT, representado pelo marcador de float 9004, possui os limites máximo e mínimo de 100 e -100.

3.4.4 Filtro Passa-Baixa ou Passa-Alta – FILTER



Menu: *Inserir-Blocos de Função-CLP-FILTER*

Entradas:

EN: Habilita o bloco

Saída:

ENO: Imagem da entrada EN

Propriedades:

TS: Período de amostragem

IN: Dado de entrada

TIMECONST: Constante de tempo do filtro

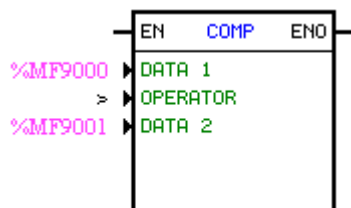
TYPE: Passa-baixa/Passa-alta

OUT: Valor filtrado do dado de entrada

No exemplo acima, se a entrada EN estiver ativa, o conteúdo do marcador de float 9000 será filtrado com uma constante de tempo de 0,25s por um filtro passa-baixa e será transferido marcador de float 9001.

3.5 BLOCOS DE CÁLCULO

3.5.1 Comparador – COMP



Menu: *Inserir-Blocos de Função-Cálculo-COMP*

Entrada:

EN: Habilita o bloco

Saída:

ENO: Vai para 1 quando a condição de comparação for satisfeita

Propriedades:

FORMAT: Inteiro ou ponto flutuante

DATA 1: Dado 1 de comparação

OPERATOR: Operador de comparação

DATA 2: Dado 2 de comparação

No exemplo acima, se a entrada EN estiver ativa e o conteúdo do marcador de float 9000 for maior que o do marcador de float 9001, então seta a saída ENO.

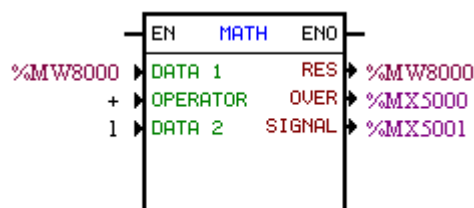


NOTA!

Se FORMAT for inteiro, todos os dados numéricos são considerados words de 15 bits + sinal (-32768 a 32767).

Resumo dos Blocos de Função

3.5.2 Operação Matemática – MATH



Menu: *Inserir-Blocos de Função-Cálculo-MATH*

Entrada:

EN: Habilita o bloco

Saída:

ENO: Indica se o cálculo foi executado

Propriedades:

FORMAT: Inteiro ou ponto flutuante

DATA1: Dado 1 do cálculo. Também pode aparecer como DATA1H e DATA1L (representando as partes alta e baixa do dado 1)

OPERATOR: Operador matemático (+, -, *, etc)

DATA2: Dado 2 do cálculo. Também pode aparecer como DATA2H e DATA2L (representando as partes alta e baixa do dado 2)

RES: Resultado do cálculo. Também pode aparecer como RESH e RESL (representando as partes alta e baixa do resultado) e também como QUOC e REM (representando o quociente e o resto de uma divisão)

OVER: Indica se o resultado ultrapassou o seu limite.

SIGNAL: Sinal do resultado

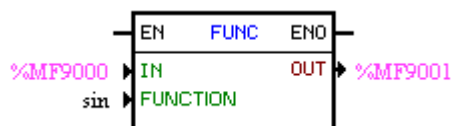
No exemplo acima, quando a entrada EN está ativa, o valor do marcador de word 8000 é incrementado a cada ciclo de scan. Quando o marcador de bit 5000 vai para 1, indica que houve um estouro de limite e o marcador de word 8000 permanece em 32767.



NOTA!

Se FORMAT for inteiro, todos os dados numéricos são considerados words de 15 bits + sinal (-32768 a 32767).

3.5.3 Função Matemática – FUNC



Menu: *Inserir-Blocos de Função-Cálculo-FUNC*

Entrada:

EN: Habilita o bloco

Saída:

ENO: Indica se o cálculo foi executado

Propriedades:

FORMAT: Inteiro ou ponto flutuante

IN: Dado a ser cálculo

FUNCTION: Função matemática (sen, cos, etc)

OUT: Resultado do cálculo

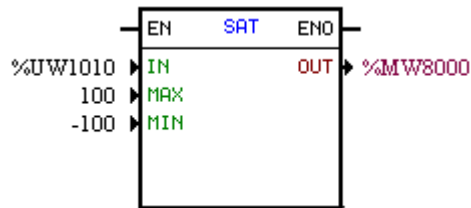
No exemplo acima, quando a entrada EN está ativa, o marcador de float 9001 apresenta o resultado do cálculo do seno do marcador de float 9000.



NOTA!

Se FORMAT for inteiro, todos os dados numéricos são considerados words de 15 bits + sinal (-32768 a 32767).

3.5.4 Saturador – SAT



Menu: *Inserir-Blocos de Função-Cálculo-SAT*

Entrada:

EN: Habilita o bloco

Saída:

ENO: Indica se houve saturação, se EN ≠ 0

Propriedades:

FORMAT: Inteiro ou ponto flutuante

IN: Dado de entrada

MAX: Valor máximo permitido

MIN: Valor mínimo permitido

OUT: Dado de saída

No exemplo acima, quando a entrada EN está ativa, o marcador de word 8000 conterá o valor do parâmetro do usuário P1010, porém limitado entre o máximo de 100 e o mínimo de -100.



NOTA!

Se FORMAT for inteiro, todos os dados numéricos são considerados words de 15 bits + sinal (-32768 à 32767).

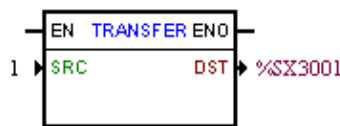


NOTA!

Caso o valor de MIN seja maior que o MAX as saídas OUT e ENO são zeradas.

3.6 BLOCOS DE TRANSFERÊNCIA

3.6.1 Transfere Dados – TRANSFER



Menu: *Inserir-Blocos de Função-Transferência-TRANSFER*

Entrada:

EN: Habilita o bloco

Saída:

ENO: Indica que a transferência foi feita

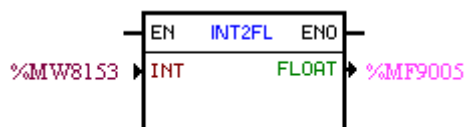
Propriedades:

SRC: Dado fonte

DST: Dado destino

No exemplo acima, se a entrada EN estiver ativa, a constante word 1 é transferida ao marcador de bit do sistema 3001 (habilita geral).

3.6.2 Converte de Inteiro (16 bits) para Ponto Flutuante – INT2FL



Menu: *Inserir-Blocos de Função-Transferência-INT2FL*

Entrada:

EN: Habilita o bloco

Saída:

ENO: Indica que a transferência foi feita

Propriedades:

INT: Dado inteiro

FLOAT: Dado convertido em ponto flutuante

No exemplo acima, se a entrada EN estiver ativa, o conteúdo do marcador de word 8153 (levando em conta o seu sinal) é convertido para ponto flutuante ao marcador de float 9005.

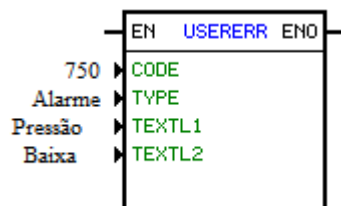


NOTA!

INT é tratado como word de 15 bits + sinal (-32768 à 32767).

Resumo dos Blocos de Função

3.6.3 Gerador de Falha ou Alarme do Usuário – USERERR



Menu: *Inserir-Blocos de Função-Transferência-USERERR*

Entrada:

EN: Habilita o bloco

Saída:

ENO: Indica 1 quando EN = 1 e o alarme ou erro foi efetivamente gerado.

Propriedades:

CODE: Código do alarme ou falha.

TYPE: 0: Gera alarme, 1: Gera falha

TEXTL1: Texto da linha 1 da HMI

TEXTL2: Texto da linha 2 da HMI

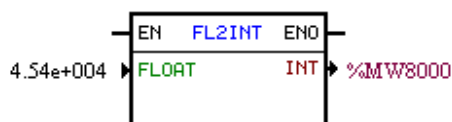
No exemplo acima, se a entrada EN estiver ativa, aparecerá A750 com o texto “Pressão Baixa” na HMI.



NOTA!

Se este bloco for configurado como Falha, é necessário resetar o drive, para o texto desaparecer da HMI e poder habilitar o drive novamente.

3.6.4 Converte de Ponto Flutuante para Inteiro (16 bits) – FL2INT



Menu: *Inserir-Blocos de Função-Transferência-FL2INT*

Entrada:

EN: Habilita o bloco

Saída:

ENO: Indica que a transferência foi feita

Propriedades:

FLOAT: Dado em ponto flutuante

INT: Dado convertido para inteiro

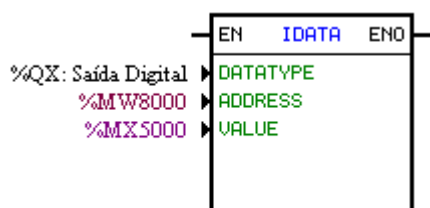
No exemplo acima, se a entrada EN estiver ativa, a constante float $4,54 \times 10^4$ é convertida para um inteiro com sinal via marcador de word 8000. Todavia, após a conversão, o marcador de word 8000 ficará com o valor de 32767, pois este é o limite positivo de uma word.



NOTA!

INT é tratado como word de 15 bits + sinal (-32768 à 32767).

3.6.5 Transfere Dados Indireta – IDATA



Menu: *Inserir-Blocos de Função-Transferência-IDATA.*

Entrada:

EN: Habilita o bloco.

Saída:

ENO: Indica que a transferência foi feita.

Propriedades:

CMD: Comando de Leitura/Escrita

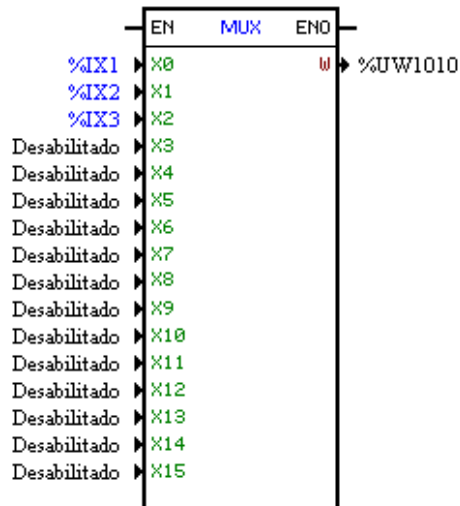
DATATYPE: Tipo de dado

ADDRESS: Endereço do usuário.

VALUE: Conteúdo lido/Valor a ser escrito

No exemplo acima, se a entrada EN estiver ativa, o conteúdo do marcador de bit 5000 é escrito para a saída digital cujo endereço é o conteúdo do marcador de word 8000.

3.6.6 Multiplexador – MUX



Menu: *Inserir-Blocos de Função-Transferência-MUX*

Entrada:

EN: Habilita a operação matemática

Saída:

ENO: Indica que a transferência foi feita

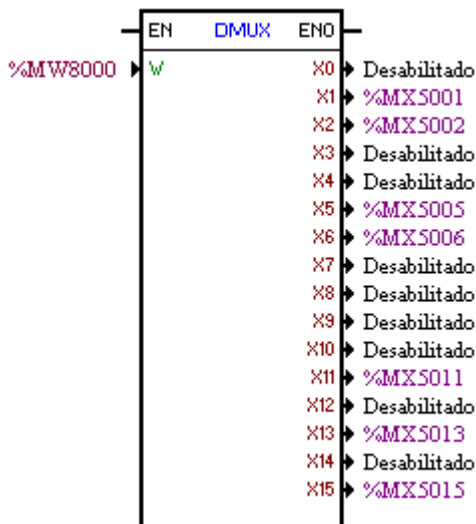
Propriedades:

X0-X15: Vetor de dados binários

W: Word resultante

No exemplo acima, quando a entrada EN está ativa, as entradas digitais 1, 2 e 3 transferem o seu conteúdo aos bits 0, 1 e 2 do parâmetros do usuário P1010.

3.6.7 Demultiplexador – DMUX



Menu: *Inserir-Blocos de Função-Transferência-DMUX*

Entrada:

EN: Habilita a operação matemática

Saída:

ENO: Indica que a transferência foi feita

Propriedades:

W: Word fonte

X0-X15: Vetor de dados binários resultante

No exemplo acima, quando a entrada EN está ativa, os bits 1, 2, 5, 6, 11, 13 e 15 do marcador de word 8000 são transferidos respectivamente aos marcadores de bit 5001, 5002, 5005, 5006, 5011, 5013 e 5015.

4 PARAMETRIZAÇÃO DO INVERSOR

A seguir serão apresentados apenas os parâmetros do inversor de frequência CFW700 que possuem relação com a SoftPLC.

4.1 SÍMBOLOS PARA DESCRIÇÃO DAS PROPRIEDADES

RO	Parâmetro somente de leitura
CFG	Parâmetro somente pode ser alterado com motor parado
Net	Parâmetro visível através da HMI se o inversor possuir interface de rede instalada – RS232, RS485, CAN, Anybus-CC, Profibus – ou se a interface USB for conectada
Serial	Parâmetro visível através da HMI se o inversor possuir interface RS232 ou RS485 instalada

4.2 PARÂMETROS DE CONFIGURAÇÃO DO CFW700

P0100 – Tempo de Aceleração

P0101 – Tempo de Desaceleração

P0220 – Seleção da Fonte LOCAL/REMOTO

P0221 – Seleção da Referência de Velocidade - Situação LOCAL

P0222 – Seleção da Referência de Velocidade - Situação REMOTO

P0223 – Seleção do Sentido de Giro - Situação LOCAL

P0226 – Seleção do Sentido de Giro - Situação REMOTO

P0224 – Seleção de Gira / Pára - Situação LOCAL

P0227 – Seleção de Gira / Pára - Situação REMOTO

P0225 – Seleção de JOG - Situação LOCAL

P0228 – Seleção de JOG - Situação REMOTO

P0251 – Função da Saída AO1

P0254 – Função da Saída AO2

P0275 – Função da Saída DO1 (RL1)

P0276 – Função da Saída DO2 (RL2)

P0277 – Função da Saída DO3 (RL3)

P0278 – Função da Saída DO4

P0279 – Função da Saída DO5



NOTA!

Maiores informações, consultar o Manual de Programação do CFW700.

Parametrização do Inversor

4.3 PARÂMETROS EXCLUSIVOS DA SOFTPLC

P1000 – Estado da SoftPLC

Faixa de	0 = Sem Aplicativo	Padrão: 0
Valores:	1 = Instal. Aplic. 2 = Aplic. Incomp. 3 = Aplic. Parado 4 = Aplic. Rodando	
Propriedades:	RO	
Grupos de acesso via HMI:	01 GRUPOS PARÂMETROS L 50 SoftPLC	

Descrição:

Permite ao usuário visualizar o status em que a SoftPLC se encontra. Se não há aplicativo instalado, os parâmetros P1001 a P1049 não serão mostrados na HMI.

Se este parâmetro apresentar a opção 2 (“Aplic. Incomp.”), indica que a versão que foi carregada do cartão de memória flash, não é compatível com o firmware atual do CFW700.

Neste caso, é necessário que o usuário recompile o seu projeto no WLP, considerando a nova versão do CFW700 e refazer o “download”. Caso isto não seja possível, pode-se fazer o “upload” deste aplicativo com o WLP, desde que a senha do aplicativo seja conhecida ou a senha não esteja habilitada.

P1001 – Comando para SoftPLC

Faixa de	0 = Pára Aplic.	Padrão: 0
Valores:	1 = Roda Aplic. 2 = Exclui Aplic.	
Propriedades:	CFG	
Grupos de acesso via HMI:	01 GRUPOS PARÂMETROS L 50 SoftPLC	

Descrição:

Permite parar, rodar ou excluir um aplicativo instalado, mas para isto, o motor deve estar desabilitado.

P1002 – Tempo Ciclo de Scan

Faixa de	0.0 a 999.9 ms	Padrão: não há
Valores:		
Propriedades:	CFG	
Grupos de acesso via HMI:	01 GRUPOS PARÂMETROS L 50 SoftPLC	

Descrição:

Consiste no tempo de varredura do aplicativo. Quanto maior o aplicativo, maior tende a ficar o tempo de varredura.

Parametrização do Inversor

P1003 – Aplicação na SoftPLC

Faixa de Valores: 0 = Usuário
1 = Regulador PID
2 = Potenciômetro Eletrônico (PE)
3 = Multispeed
4 = Comando a Três Fios (Start/Stop)
5 = Comando Avanço e Retorno

Padrão: 0

Propriedades: CFG

Grupos de acesso via HMI: 01 GRUPOS PARÂMETROS
L 50 SoftPLC

Descrição:

Permite ao usuário selecionar aplicações incorporadas no CFW700.

P1003	Descrição
0	Define que a aplicação que irá rodar na SoftPLC é aquela carregada pelo usuário através da ferramenta de programação ladder.
1	Define que a aplicação que irá rodar na SoftPLC é o regulador PID; pode ser utilizada para fazer o controle de um processo em malha fechada. Essa aplicação coloca um regulador proporcional, integral e derivativo superposto ao controle normal de velocidade do CFW700.
2	Define que a aplicação que irá rodar na SoftPLC é o potenciômetro eletrônico; permite o ajuste da referência de velocidade do motor através de duas entradas digitais, sendo uma para acelerar o motor e outra para desacelerar o motor.
3	Define que a aplicação que irá rodar na SoftPLC é o multispeed; permite o ajuste da referência de velocidade relacionando os valores definidos por parâmetros (P1011 a P1018) através da combinação lógica das entradas digitais DI4, DI5 e DI6, tendo como limite máximo 8 referências de velocidade pré-programadas. Traz como vantagens a estabilidade das referências fixas pré-programadas, e a imunidade contra ruídos elétricos (entradas digitais DIx isoladas).
4	Define que a aplicação que irá rodar na SoftPLC é o comando a três fios (start/stop); permite o comando do inversor de maneira análoga a uma partida direta com botão de emergência e contato de retenção.
5	Define que a aplicação que irá rodar na SoftPLC é o comando avanço e retorno; proporciona ao usuário a combinação de dois comandos do inversor (Sentido de Giro e Gira/Pára) em um só comando via entrada digital.



NOTA!

Para mais informações sobre as aplicações no CFW700, consultar o capítulo XX no manual do inversor de frequência CFW700.

P1010 até P1059 – Parâmetros SoftPLC

Faixa de Valores: 0 a 65535

Padrão: 0

Propriedades: CFG

Grupos de acesso via HMI: 01 GRUPOS PARÂMETROS
L 50 SoftPLC

Descrição:

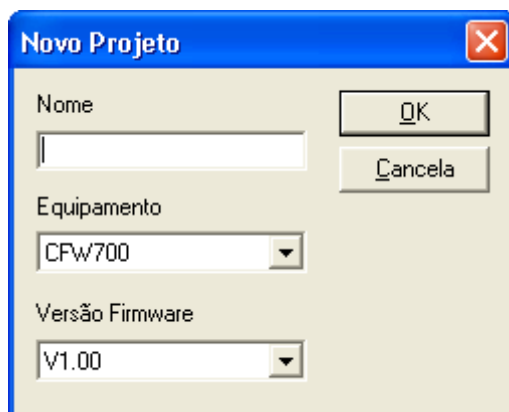
Consistem em parâmetros de uso definido pelo usuário via software WLP. Também é possível ao usuário configurar estes parâmetros, conforme descrito no item 5.5.

5 RESUMO DAS PRINCIPAIS FUNÇÕES DO WLP

Este capítulo traz informações básicas sobre as operações feitas com o software WLP para programação do inversor CFW700. Maiores informações podem ser obtidas na ajuda (help) do software WLP.

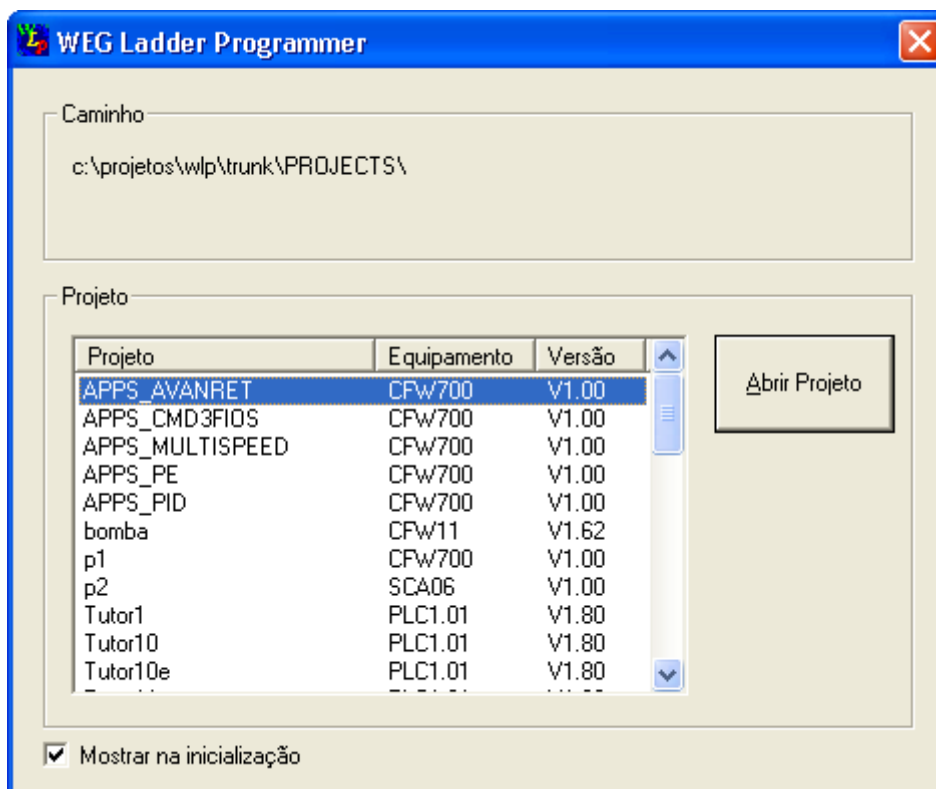
5.1 PROJETO – NOVO

Cria um novo projeto. Além de definir o nome do projeto, é necessário configurar o equipamento e a respectiva versão de firmware.



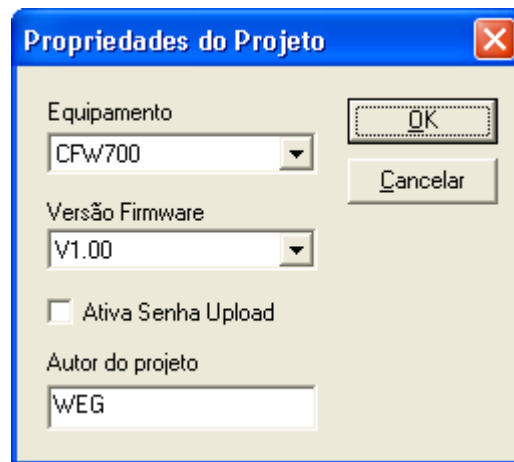
5.2 PROJETO – ABRIR

Abre o projeto selecionado.



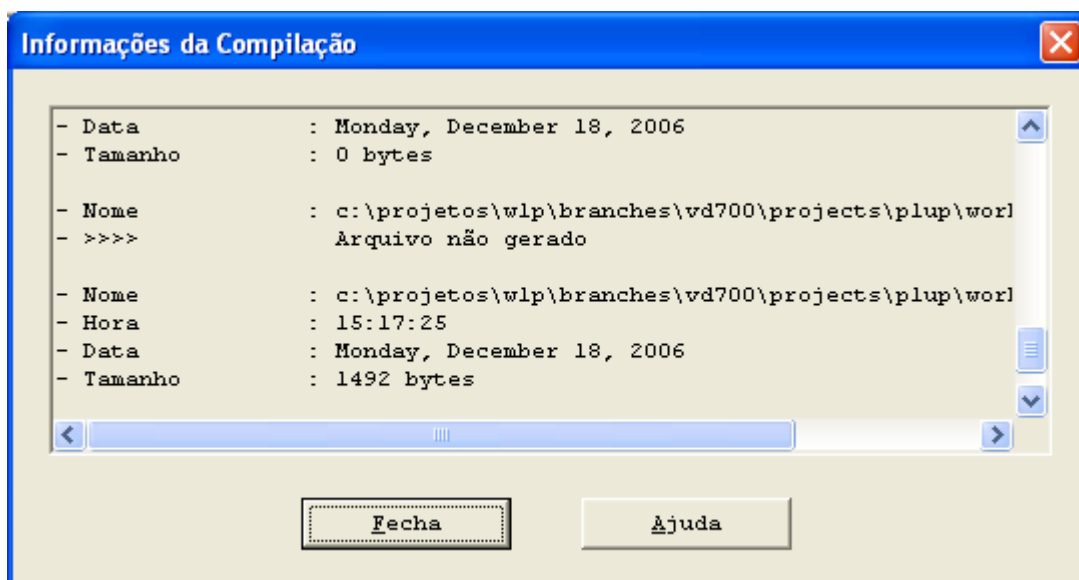
5.3 PROJETO – PROPRIEDADES

Permite ao usuário redefinir o equipamento e a versão de firmware. Nesta janela, também se configura se o projeto terá senha para upload.



5.4 EXIBIR – INFORMAÇÕES DA COMPILAÇÃO

Permite ao usuário saber o tamanho em bytes do aplicativo compilado (<nomedoprojeto>.bin) a ser enviado ao equipamento.

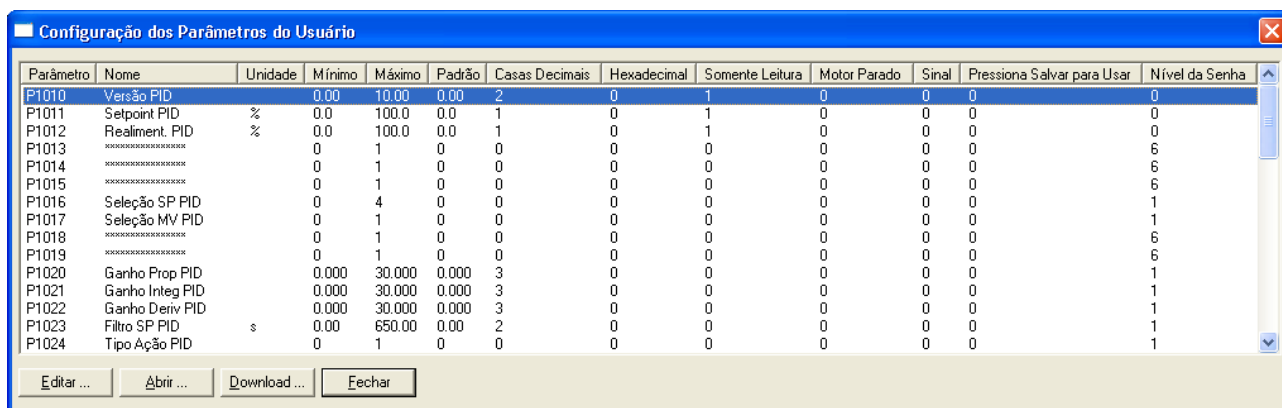


5.5 EXIBIR – CONFIGURAÇÃO DOS PARÂMETROS DO USUÁRIO

Abre uma janela de visualização dos atributos de todos os parâmetros do usuário. Com um duplo clique sobre o parâmetro selecionado, é permitida a configuração destes atributos, que incluem:

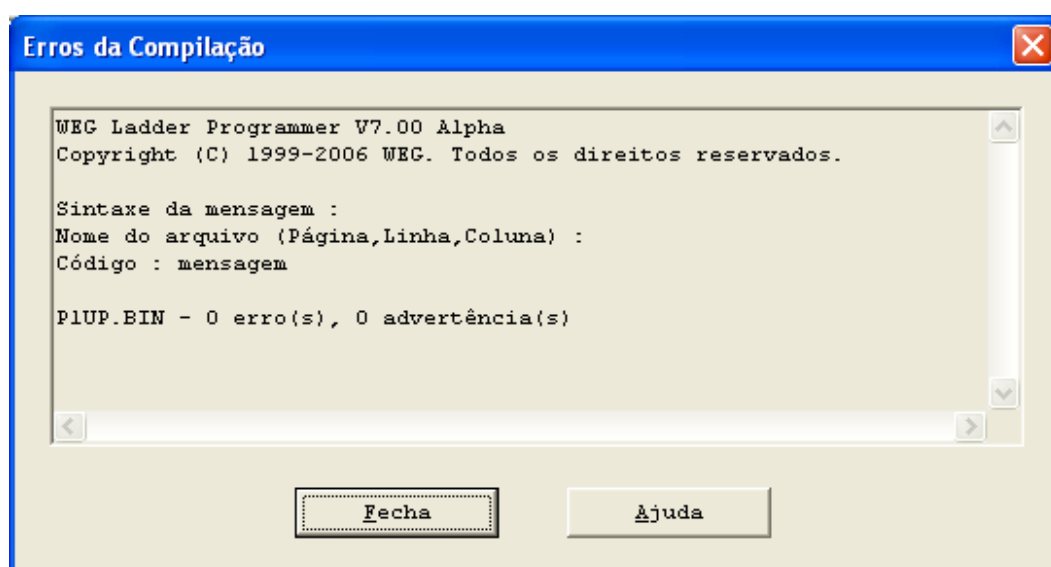
- Unidades;
- Limite mínimo e máximo;
- Número de casas decimais;
- Formato hexadecimal ou normal;
- Somente leitura ou escrita;
- Alteração do parâmetro: sem confirmação, motor parado ou motor parado+salvar;
- Com sinal ou sem sinal;
- Nível de senha: sempre visualiza e ignora senha, sempre visualiza e habilita a senha, somente visualiza ou nunca visualiza;
- Permite salvar o valor do parâmetro (retentivo), quando o mesmo é utilizado em blocos (CLP, Cálculos e Transferências) na desenergização;

Estas configurações podem ser transmitidas ao CFW700 pelo botão "Transmitir".



5.6 CONSTRUIR – COMPILAR

Analisa o aplicativo e gera o código para o equipamento especificado.



5.7 COMUNICAÇÃO – CONFIGURAÇÃO

Para o CFW700 se utiliza a comunicação RS-232.



5.8 COMUNICAÇÃO – DOWNLOAD

Este comando permite enviar ao CFW700 o aplicativo e/ou as configurações dos parâmetros do usuário.



6 FALHAS, ALARMES E POSSÍVEIS CAUSAS

Tabela 6.1: “Falhas”, “Alarmes” e causas mais prováveis

Falha/Alarme	Descrição	Causas mais Prováveis
A702: Inversor Desabilitado	Ocorre quando um bloco de movimento (Bloco REF) é ativo e o comando de habilita geral do drive não está ativo.	■ Verificar se o comando de habilita geral do drive está ativo.
A704: Dois Movim. Habilitados	Ocorre quando 2 ou mais blocos de movimento (Bloco REF) estão habilitados ao mesmo tempo.	■ Verificar lógica do programa do usuário.
A706: Ref. Não Progr. SPLC	Ocorre quando um bloco de movimento é habilitado e a referência de velocidade não está programada para a SoftPLC.	■ Verificar a programação das referências no modo local e/ou remoto (P221 e P222).