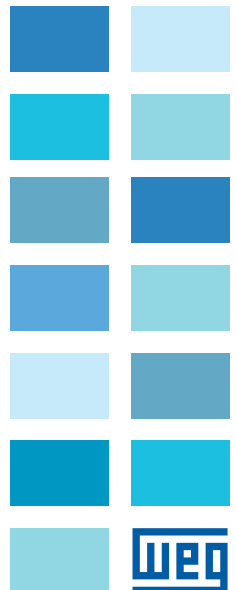
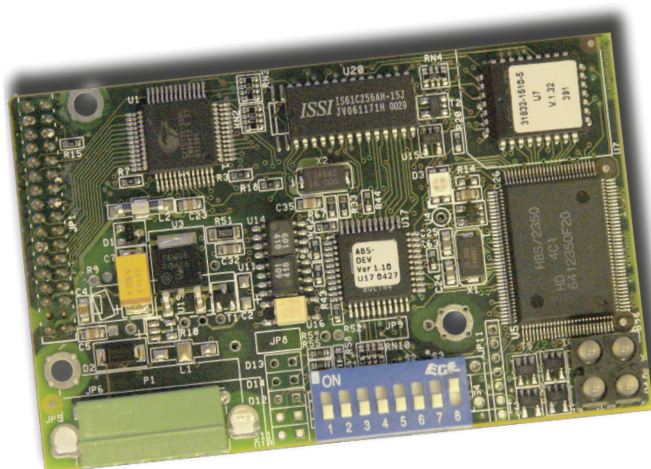


DeviceNet Drive Profile

MVW-01

Guia de Instalação, Operação e Configuração

Idioma: Português



ÍNDICE

1 DEVICENET DRIVE PROFILE	5
1.1 Lista de Parâmetros DeviceNet Drive Profile.....	5
1.2 INSTALAÇÃO DO KIT FIELDBUS	6
1.3 INTRODUÇÃO À REDE DEVICENET	6
1.3.1 Cabo e Conector.....	6
1.3.2 Terminação da Linha	7
1.3.3 Taxa de Transmissão (Baud rate) e Endereço do Nó	7
1.3.4 LEDs de Indicação	7
1.3.5 Arquivo de Configuração (EDS File)	8
1.4 PARAMETRIZAÇÃO DO INVERSOR	8
1.5 CONTEÚDO DOS DADOS PARA I/O	9
1.5.1 Conteúdo dos Dados para Instâncias 20 / 70.....	9
1.5.1.1 Words de Escrita (Instância 20).....	10
1.5.1.2 Words de Leitura (Instância 70).....	10
1.5.2 Conteúdo dos Dados para Instâncias 21 / 71	11
1.5.2.1 Words de Escrita (Instância 21)	11
1.5.2.2 Words de Leitura (Instância 71)	12
1.5.2.3 Referência de Velocidade para Instâncias 20 e 21	13
1.5.2.4 Velocidade do Motor para Instâncias 70 e 71	14
1.5.3 Conteúdo dos Dados para Instâncias 100 / 101	14
1.5.3.1 Words de Escrita (Instância 100)	14
1.5.3.2 Words de Leitura (Instância 101)	15
1.5.4 Conteúdo dos Dados para Instâncias 102 / 103	15
1.5.4.1 Seleção do Número de Words para a Área de I/O	15
1.5.4.2 Seleção dos Parâmetros para a Área de I/O.....	15
1.5.4.3 Parâmetros Específicos para a Área de I/O.....	16
1.5.4.3.1 P370: Comando Lógico WEG	16
1.5.4.3.2 P371: Referência de Velocidade	17
1.5.4.3.3 P372: Comando para as Saídas Digitais	17
1.5.4.3.4 P373: Estado Lógico WEG.....	18
1.5.4.3.5 P374: Velocidade do Motor	18
1.5.4.3.6 P375: Estado das Entradas Digitais.....	19
1.5.4.4 Erros de Software	19
1.6 TEMPOS DE ATUALIZAÇÃO DAS VARIÁVEIS	19
1.7 ERROS DE COMUNICAÇÃO	20
1.7.1 E29: Comunicação Fieldbus Inativa	20
1.7.2 E30: Cartão Fieldbus Inativo	20
1.8 CLASSES DE OBJETOS PARA A REDE DEVICENET	21
1.8.1 Identity Object, Class 01 _{hex}	21
1.8.2 Message Router Object, Class 02 _{hex}	21
1.8.3 DeviceNet Object, Class 03 _{hex}	21
1.8.4 Assembly Object, Class 04 _{hex}	22
1.8.5 DeviceNet Connection Object, Class 05 _{hex}	23
1.8.6 Acknowledge Handler Object, Class 2B _{hex}	26
1.8.7 Motor Data Object, Class 28 _{hex}	26
1.8.8 Control Supervisor Object, Class 29 _{hex}	27
1.8.9 AC/DC Drive Object, Class 2A _{hex}	28
1.8.10 Vendor Specific Object, Class 90 _{hex}	28

GUIA DEVICENET DRIVE PROFILE MVW-01

1 DEVICENET DRIVE PROFILE

Foi desenvolvida com o objetivo de tornar disponível no produto uma interface de comunicação para rede DeviceNet que possua as seguintes características:

- Possibilite a parametrização do inversor através da rede, com o acesso direto aos parâmetros com mensagens enviadas pelo mestre.
- Siga o padrão Device Profile for AC and DC Drives, especificado pela ODVA (Open DeviceNet Vendor Association), que define um conjunto comum de objetos para drives que operam em rede DeviceNet.

Este guia fornece uma visão geral sobre o funcionamento da rede DeviceNet, se atendo principalmente à parametrização e operação do inversor nesta rede. A descrição detalhada do protocolo pode ser obtido junto à ODVA.

1.1 LISTA DE PARÂMETROS DEVICENET DRIVE PROFILE

Parâmetro	Descrição	Faixa de Valores	Ajuste de Fábrica
P309	Fieldbus	0 a 10 10 = DeviceNet Drive Profile	0 = Inativo
P340	Instâncias de I/O	0 = 20 / 70 1 = 21 / 71 2 = 100 / 101 3 = 102 / 103	0 = 20 / 70
P348	Parâmetro de Escrita #1	0 a 999	370
P349	Parâmetro de Escrita #2	0 a 999	371
P350	Parâmetro de Escrita #3	0 a 999	372
P351	Parâmetro de Escrita #4	0 a 999	0
P352	Parâmetro de Leitura #1	0 a 999	373
P353	Parâmetro de Leitura #2	0 a 999	374
P354	Parâmetro de Leitura #3	0 a 999	375
P355	Parâmetro de Leitura #4	0 a 999	0
P356	Parâmetro de Escrita #5	0 a 999	0
P357	Parâmetro de Escrita #6	0 a 999	0
P358	Parâmetro de Escrita #7	0 a 999	0
P359	Parâmetro de Escrita #8	0 a 999	0
P360	Parâmetro de Escrita #9	0 a 999	0
P361	Parâmetro de Leitura #5	0 a 999	0
P362	Parâmetro de Leitura #6	0 a 999	0
P363	Parâmetro de Leitura #7	0 a 999	0
P364	Parâmetro de Leitura #8	0 a 999	0
P365	Parâmetro de Leitura #9	0 a 999	0
P366	Número de Word de I/O	1 a 9	2

1.2 INSTALAÇÃO DO KIT FIELDBUS

Para a comunicação com a rede DeviceNet é fornecido um kit KFB-DD contendo:

- Cartão de comunicação DeviceNet Drive Profile;
- Cabo com conector para a rede;
- Arquivo de configuração de rede EDS;
- Guia DeviceNet Drive Profile.

Para a instalação do kit (caso o kit seja fornecido separadamente do inversor), consulte o item 8.4.1 - Instalação do Kit Fieldbus do manual do usuário MVW-01, que descreve os passos para a instalação.

1.3 INTRODUÇÃO À REDE DEVICENET

A comunicação DeviceNet é utilizada para automação industrial, normalmente para o controle de válvulas, sensores, unidades de entradas/saídas e equipamentos de automação. O link de comunicação DeviceNet é baseado em um protocolo de comunicação CAN (Controller Area Network). A figura 1.1 fornece uma visão geral de uma rede DeviceNet.

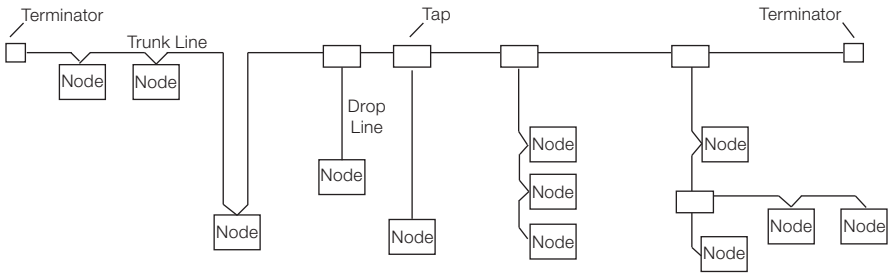
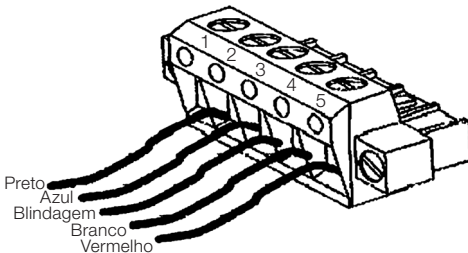


Figura 1.1: Rede DeviceNet

1.3.1 Cabo e Conector

A rede DeviceNet utiliza um cabo de cobre blindado com 2 pares trançados, sendo um deles responsável pela distribuição da alimentação 24 Vcc nos diversos nós, e o outro utilizado para o sinal de comunicação. Na figura 1.2 é apresentado um exemplo de conector para o MVW-01, juntamente com o esquema de cores utilizadas para a conexão:



1	Preto	V-
2	Azul	CANH
3	Blindagem	SHIELD
4	Branco	CANL
5	Vermelho	V+

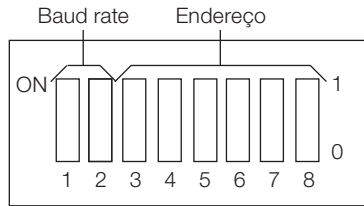
Figura 1.2: Conector para a rede DeviceNet

1.3.2 Terminação da Linha

Os pontos iniciais e finais da rede devem ser terminados na impedância característica para evitar reflexões. Para tanto, um resistor de 120 ohms/0,5 W deve ser conectado entre os pinos 2 e 4 do conector.

1.3.3 Taxa de Transmissão (Baudrate) e Endereço do Nó

Existem três diferentes taxas de transmissão (baud rate) para DeviceNet: 125 k, 250 k ou 500 kbits/s. A seleção é feita através de chaves DIP existentes no cartão de comunicação. O endereço do nó é selecionado através de seis chaves DIP presentes no cartão eletrônico, permitindo endereçamentos de 0 a 63.



Baud rate [bits/s]	DIPs 1 e 2
125 k	00
250 k	01
500 k	10
Reservado	11

Endereço	DIP3 a DIP8
0	000000
1	000001
2	000010
...
61	111101
62	111110
63	111111

Figura 1.3: Configuração do endereço e taxa de transmissão



NOTA!

A taxa de transmissão e o endereço do inversor na rede somente são atualizados durante a energização do inversor.

1.3.4 LEDs de Indicação

Para diagnóstico da rede, o cartão de comunicação possui quatro LEDs de indicação, com as seguintes funções:

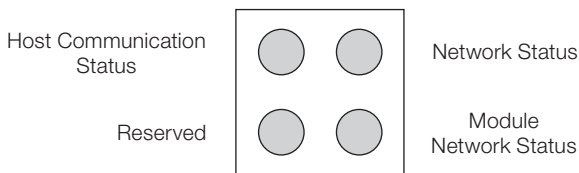


Figura 1.4: Indicação dos LEDs de status do cartão de comunicação

Tabela 1.1: Sinalização dos LEDs de status do cartão de comunicação

LED	Cor	Descrição
Host Communication Status	Verde	Troca de dados cartão - inversor OK
	Vermelho	Falha na troca de dados cartão - inversor (permanente)
	Vermelho Piscante	Falha na troca de dados cartão - inversor (temporário)
Network Status	Desligado	Sem alimentação/off line
	Verde	Link operante, conectado
	Vermelho	Falha crítica do link
	Verde Piscante	On line não conectado
	Vermelho Piscante	Time out da conexão
Module Network Status	Desligado	Sem alimentação
	Vermelho	Falta não recuperável
	Verde	Cartão operacional
	Vermelho Piscante	Falta menor

1.3.5 Arquivo de Configuração (EDS File)

Cada elemento de uma rede DeviceNet está associado a um arquivo EDS, que contém todas as informações referente ao elemento. Este arquivo é utilizado pelo programa de configuração da rede durante a configuração da mesma. Utilize o arquivo com extensão .EDS fornecido juntamente com o kit Fieldbus.



NOTA!

Para esta versão de firmware em conjunto com o cartão de comunicação, é possível programar o mestre para a comunicação em dois diferentes tipos de conexão:

- Polled; ou
- Change of State & Ciclic.

1.4 PARAMETRIZAÇÃO DO INVERSOR

Existe um conjunto de parâmetros que habilita e configura a operação do inversor na rede DeviceNet. Antes de iniciar a operação em rede, é necessário configurar estes parâmetros para que o inversor opere de acordo com o desejado.

- **P220... P228:** Estes parâmetros definem a fonte dos comandos para o inversor nos modos Local e Remoto. Para os comandos que se deseja operar via rede, é necessário programar estes parâmetros para a opção “Fieldbus”. Consulte o manual do usuário para a descrição detalhada destes parâmetros.
- **P309:** Uma vez instalado o kit Fieldbus, o parâmetro que habilita a operação do cartão de comunicação é o P309. Para a operação do inversor com o cartão DeviceNet Drives Profile, é necessário configurar P309 = 10.
- **P313:** Caso o inversor esteja sendo controlado pelo mestre da rede, mas ocorra alguma falha de comunicação, o inversor irá indicar E29 (para mais detalhes consulte o item 1.7.1 deste guia), neste caso é possível configurar o inversor para que ele tome alguma ação (por exemplo, parar o motor). Esta ação é controlada pelo P313.

- **P340***: Este parâmetro define a forma como serão disponibilizados os dados na área de I/O do mestre da rede. Existem quatro opções: duas que seguem o modelo Drive Profile definido pela ODVA, e duas que são específicas WEG.
- **P348... P365***: Caso seja selecionado P340 = 3 (102 / 103), é possível selecionar, através do P348 até P365, o número dos parâmetros devem ocupar a área de I/O do mestre. Existe também um grupo de parâmetros especiais (P370 até P375), disponíveis somente via Fieldbus, para controle e monitoração do estado do inversor.
- **P366***: Quando P340 = 3, este parâmetro define o número de words (1 word = 2 bytes) mapeados para a área de I/O (cada word equivale a um parâmetro). O conteúdo de cada Word é selecionado nos parâmetros P348 até P365. É possível mapear até 9 parâmetros de leitura (input) e 9 parâmetros de escrita (output).

* Alterações nestes parâmetros somente terão efeito na próxima vez que o inversor for energizado. A explicação detalhada destes parâmetros é feita a seguir.

1.5 CONTEÚDO DOS DADOS PARA I/O

O inversor MVW-01 em conjunto com o cartão de comunicação para DeviceNet Drive Profile contem quatro diferentes formatos de dados para serem mapeados para a área de I/O do inversor (assembly instances). Estas instâncias são definidas por P340, sendo que as duas primeiras opções seguem o modelo Drive Profile definido pela ODVA, e as próximas duas são específicas WEG.

Instâncias (P340)	Nº de words de input / output	Dados da área de output	Dados da área de input	Drive Profile
20 / 70	2	Word 1 = controle Word 2 = referência de velocidade	Word 1 = estado Word 2 = velocidade atual	ODVA
21 / 71	2	Word 1 = controle Word 2 = referência de velocidade	Word 1 = estado Word 2 = velocidade atual	ODVA
100 / 101	2	Word 1 = controle (WEG) Word 2 = referência de velocidade (WEG)	Word 1 = estado (WEG) Word 2 = velocidade atual (WEG)	WEG
102 / 103	1 ... 9 Definido através do P366	Definidos pelos parâmetros P348, P349, P350, P351, P356, P357, P358, P359, P360	Definidos pelos parâmetros P352, P353, P354, P355, P361, P362, P363, P364, P365	WEG

1.5.1 Conteúdo dos Dados para Instâncias 20 / 70

Programando P340 = 0 (20 / 70), o inversor automaticamente irá disponibilizar para a área de I/O duas words de escrita (output) e duas words de leitura (input), com o conteúdo descrito nos itens a seguir.

1.5.1.1 Words de Escrita (Instância 20)

Word 1 = Control word

Palavra de controle, formada por 16 bits onde cada bit possui a seguinte função:

Nº do Bit	Bit = 0	Bit = 1
0	Parar	Girar
1	Reservado	
2	Sem função	Reset de erros
3 ... 15	Reservado	

■ **Bit 0: Girar**

Bit 0 = 0: envia o comando de parada por rampa para o inversor.

Bit 0 = 1: envia o comando para habilitação por rampa (girar motor).

Observação: este comando só atua caso o inversor esteja programado para ser comandado via Fieldbus (consulte os parâmetros P224 e P227).

■ **Bit 2: Reset de erros**

Bit 2 = 0: não faz reset de erros.

Bit 2 = 1: envia o comando de reset para o inversor.

Word 2 = Referência de velocidade

Consulte o item 1.5.2.3 - Referência de velocidade para instâncias 20 e 21.

1.5.1.2 Words de Leitura (Instância 70)

Word 1 = Status Word

Palavra de estado, formada por 16 bits onde cada bit possui a seguinte função:

Nº do Bit	Bit = 0	Bit = 1
0	Sem erro	Com erro
1	Reservado	
2	Parado	Girando
3 ... 15	Reservado	

■ **Bit 0: Erro**

Bit 0 = 0: inversor não está em estado de erro.

Bit 0 = 1: inversor está com algum erro atuando.

■ **Bit 2: Girando**

Bit 2 = 0: inversor está parado.

Bit 2 = 1: inversor está girando o motor.

Word 2 = Velocidade do motor

Consulte o item 1.5.2.4 - Velocidade do motor para instâncias 70 e 71.

1.5.2 Conteúdo dos Dados para Instâncias 21 / 71

Programando P340 = 1 (21 / 71), o inversor automaticamente irá disponibilizar para a área de I/O duas words de escrita (output) e duas words de leitura (input), com o seguinte conteúdo:

1.5.2.1 Words de Escrita (Instância 21)

Word 1 = Control Word

Palavra de controle, formada por 16 bits onde cada bit possui a seguinte função:

Nº do Bit	Bit = 0	Bit = 1
0	Parar	Girar horário
1	Parar	Girar anti-horário
2	Sem função	Reset de erros
3	Reservado	
4	Reservado	
5	Controle local	Controle via rede
6	Referência local	Referência via rede
7 ... 15	Reservado	

■ Bit 0: Girar horário

Bit 0 = 0: envia o comando de parada por rampa para o inversor. Ao final da parada por rampa, é feito também a desabilitação geral do inversor. .

Bit 0 = 1: envia o comando para habilitação geral e por rampa no sentido horário.

Observação: este comando só atua caso o inversor esteja programado para ser comandado via Fieldbus (consulte os parâmetros P224 e P227). Ao final da parada por rampa, é feito também a desabilitação geral do inversor.

■ Bit 1: Girar anti-horário

Bit 1 = 0: envia o comando de parada por rampa para o inversor. Ao final da parada por rampa, é feito também a desabilitação geral do inversor.

Bit 1 = 1: envia o comando para habilitação geral e por rampa no sentido anti-horário.

Observação: este comando só atua caso o inversor esteja programado para ser comandado via Fieldbus (consulte os parâmetros P224 e P227).

■ Bit 2: Reset de erros

Bit 2 = 0: não faz reset de erros.

Bit 2 = 1: envia o comando de reset para o inversor.

Observação: depois de um reset de erros, o inversor perderá o controle e a referência via rede (bits 5 e 6), sendo necessário zerar estes bits para escrever novamente os valores desejados.

■ Bit 5: Controle via rede

Bit 5 = 0: envia comando para o inversor ser controlado localmente.

Bit 5 = 1: envia comando para o inversor ser controlado remotamente.

Observação: este comando atua diretamente sobre o modo de operação local/remoto. Desta forma, para que ele seja válido, é necessário deixar o comando local/remoto habilitado via Fieldbus (P220 = 7 ou 8), e configurar os comandos do modo remoto também para Fieldbus (P226, P227 e P228).

■ **Bit 6: Referência via rede**

Bit 6 = 0: envia comando para o inversor utilizar a referência local.

Bit 6 = 1: envia comando para o inversor utilizar a referência recebida via rede.

Observação: este comando altera os valores programados nos parâmetros P221 e P222. Quando a referência for via rede, tanto P221 quanto P222 são programados para 10 (Fieldbus). Quando a referência for via local, o inversor restaura os valores inicialmente programados, que foram salvos durante a energização do inversor.

Word 2 = Referência de velocidade

Consulte o item 1.5.2.3 - Referência de velocidade para instâncias 20 e 21.

1.5.2.2 Words de Leitura (Instância 71)

Word 1 = Status Word

Palavra de estado, formada por 16 bits sendo que cada bit possui a seguinte função:

Nº do Bit	Bit = 0	Bit = 1
0	Sem erro	Com erro
1	Sem função	Warning
2	Parado	Girando horário
3	Parado	Girando anti-horário
4	Sem função	Pronto
5	Controle local	Controle via rede
6	Referência local	Referência via rede
7	Não atingiu a referência	Girando no valor da referência
8 ... 15	Estado do inversor	

■ **Bit 0: Erro**

Bit 0 = 0: inversor não está em estado de erro.

Bit 0 = 1: inversor está com algum erro atuando.

■ **Bit 1: Warning**

O MVW-01 não possui indicações de warning.

■ **Bit 2: Girando horário**

Bit 2 = 0: inversor está parado.

Bit 2 = 1: inversor está girando o motor no sentido horário.

■ **Bit 3: Girando anti-horário**

Bit 3 = 0: inversor está parado.

Bit 3 = 1: inversor está girando o motor no sentido anti-horário.

■ **Bit 4: Pronto**

Bit 4 = 0: inversor está fazendo inicialização ou está em erro.

Bit 4 = 1: inversor está inicializado e sem erro.

■ Bit 5: Controle via rede

Bit 5 = 0: inversor está no modo local.

Bit 5 = 1: inversor está no modo remoto.

Observação: programar P220, P226, P227 e P228 para Fieldbus, de forma que este bit realmente represente que o controle está sendo feito via Fieldbus.

■ Bit 6: Referência via rede

Bit 6 = 0: inversor não está usando a referência recebida pela rede.

Bit 6 = 1: inversor está usando a referência recebida pela rede.

■ Bit 7: Girando no valor de referência

Bit 7 = 0: inversor não está girando o motor no valor de referência indicada.

Bit 7 = 1: inversor está girando o motor no valor de referência.

Observação: o inversor considera um erro de 0,5 % da velocidade síncrona do motor para analisar se o motor está girando no valor da referência.

Estado do inversor: byte que pode assumir os seguintes valores:

Valor do byte	Significado	Observação
0	Específico fabricante	Não utilizado
1	Inicializando	
2	Não está pronto	
3	Pronto	
4	Habilitado	
5	Parando	Detecta somente se o comando for dado via Fieldbus
6	Parando por erro	Inversor não possui este estado
7	Com erro	

Word 2 = Velocidade do motor

Consulte o item 1.5.2.4 - Velocidade do motor para instâncias 70 e 71.

1.5.2.3 Referência de Velocidade para Instâncias 20 e 21

A referência de velocidade nas instâncias 20 e 21 (output) é recebida pelo inversor na forma de um número inteiro com sinal (em complemento de 2). Cada unidade representa 1 rpm, sendo que valores negativos são interpretados pelo inversor como referência para o inversor girar no sentido anti-horário. Desta forma, temos como exemplo:

1200 = 04B0_{hex} = referência de 1200 rpm com sentido de giro horário

-1200 = FB50_{hex} = referência de 1200 rpm com sentido de giro anti-horário

**NOTA!**

- O valor de referência só será utilizado pelo inversor caso ele esteja programado para receber a referência via Fieldbus (consulte os parâmetros P221 e P222).
- Valores negativos somente irão alterar o sentido de giro se este estiver programado para ser comandado via Fieldbus (consulte os parâmetros P223 e P226).
- É necessário que os valores enviados respeitem a faixa de valores mínimo e máximo permitidos para referência, que são programados nos parâmetros P133 e P134. Caso contrário o inversor não irá considerar o valor recebido.
- Para que a referência em rpm corresponda corretamente à referência para o motor, é necessário ajustar o valor da escala de velocidade no P208.
- Se um valor negativo de referência for enviado em conjunto com um comando de girar no sentido anti-horário, o inversor irá girar o motor no sentido horário.

1.5.2.4 Velocidade do Motor para Instâncias 70 e 71

A velocidade que o inversor está girando o motor é transmitida pelo inversor na forma de um número inteiro com sinal (em complemento de 2). Cada unidade representa 1 rpm, sendo que valores positivos representam que o motor está girando no sentido horário, e valores negativos representam que o motor está girando no sentido anti-horário. Por exemplo:

1800 = 0708_{hex} = motor girando à velocidade de 1800 rpm com sentido de giro horário.
-1800 = F8F8_{hex} = motor girando à velocidade de 1800 rpm com sentido de giro anti-horário.

**NOTA!**

Para que a velocidade em rpm corresponda corretamente à velocidade do motor, é necessário ajustar o valor da escala de velocidade no P208.

1.5.3 Conteúdo dos Dados para Instâncias 100 / 101

Programando P340 = 2 (100 / 101), o inversor estará operando em um modo específico WEG. Também serão disponibilizadas para a área de I/O duas words de escrita (output) e duas words de leitura (input), com o seguinte conteúdo:

1.5.3.1 Words de Escrita (Instância 100)

Word 1 = Comando lógico WEG

Palavra de comando lógico, através da qual são enviados comandos para o inversor via rede. Para saber a estrutura desta palavra, consulte o item 1.5.4.3.1 - P370: Comando lógico WEG.

Word 2 = Referência de velocidade

Palavra com o valor da referência de velocidade para o inversor via Fieldbus. Para saber a estrutura desta palavra, consulte o item 1.5.4.3.2 - P371: Referência de velocidade.

1.5.3.2 Words de Leitura (Instância 101)

Word 1 = Estado lógico WEG

Palavra de estado lógico, pela qual o inversor indica seu estado para a rede. Para saber a estrutura desta palavra, consulte o item 1.5.4.3.4 - P373: Estado lógico WEG.

Word 2 = Velocidade do motor

Palavra com o valor da velocidade atual do motor. Para saber a estrutura desta palavra, consulte o item 1.5.4.3.5 - P374: Velocidade do motor.

1.5.4 Conteúdo dos Dados para Instâncias 102 / 103

O número de words e o conteúdo dos dados na área de I/O para as instâncias 102 (output) e 103 (input) é configurável pelo usuário, através dos parâmetros 348 até 366.

1.5.4.1 Seleção do Número de Words para a Área de I/O

Para as instâncias 102 / 103, o número de words de I/O é selecionado através do P366. Cada word representa um parâmetro, e os parâmetros que serão disponibilizados na área de I/O são selecionados em P348 até P365. É possível mapear de 1 até 9 parâmetros, sendo que o número de parâmetros de leitura (input) será sempre igual ao número de parâmetros de escrita (output).

1.5.4.2 Seleção dos Parâmetros para a Área de I/O

Uma vez selecionado o número de words a serem mapeados para I/O, é necessário selecionar que informações estas words representam. Isto é feito configurando os parâmetros P348 até P365, através dos quais é possível configurar 9 parâmetros para a área de leitura, mais 9 parâmetros para a área de escrita.

Por exemplo, digamos que se deseja ler informações de corrente (P003) e torque (P009), e escrever informações sobre rampa de aceleração (P100) e desaceleração (P101). Neste caso, basta fazer a seguinte programação do inversor:

P366 = 2: indicando que serão mapeados 2 parâmetros de leitura e 2 de escrita

P348 = 100: indicando que a primeira word de escrita será o conteúdo para o P100

P349 = 101: indicando que a segunda word de escrita será o conteúdo para o P101

P352 = 3: indicando que a primeira word de leitura representa o conteúdo do P003

P353 = 9: indicando que a segunda word de leitura representa o conteúdo do P009

Como só estão sendo utilizadas duas words de leitura e duas de escrita, os demais parâmetros utilizados para mapear as words são desconsiderados.

O conteúdo de cada parâmetro é transmitido como sendo uma word, e para interpretar corretamente os valores enviados e recebidos, é necessário observar o número de casas decimais utilizados no parâmetro. Por exemplo, para um tempo de aceleração (P100) de 5,0 segundos, como temos uma casa decimal de resolução, o valor real a ser transmitido na word é 50 (0032_{hex}). Para a lista de parâmetros existentes no inversor, consulte o manual do usuário MVW-01.



NOTA!

- Quase todos os parâmetros do inversor descritos no manual estão disponíveis para mapeamento na área de I/O, com exceção dos parâmetros P000, P001, P215 e P408.
- Os parâmetros mapeados para escrita possuem seus valores atualizados constantemente pela rede, porém não são gravados na memória não volátil (EEPROM), de forma que são restaurados para o valor anterior no caso de um reset do inversor.

1.5.4.3 Parâmetros Específicos para a Área de I/O

Além dos parâmetros do inversor (descritos na lista de parâmetros do manual do usuário), existem outros parâmetros acessíveis somente pela área de I/O, que foram criados de forma a possibilitar que comandos e estados do inversor possam ser acessados. Estes parâmetros não são visíveis pela HMI do inversor, porém podem ser indicados nos parâmetros P348 até P365, de forma a serem acessíveis pela área de I/O. São eles:

1.5.4.3.1 P370: Comando Lógico WEG

Parâmetro de leitura/escrita, acessível somente via Fieldbus, através do qual são enviados comandos para o inversor via rede. Formada por 16 bits onde cada bit possui a seguinte função:

Nº do Bit	Bit = 0	Bit = 1
0	Desabilita rampa	Habilita rampa
1	Desabilita geral	Habilita geral
2	Sentido de giro horário	Sentido de giro anti-horário
3	Desabilita JOG	Habilita JOG
4	Local	Remoto
5		Reservado
6		Reservado
7	Não faz reset	Faz reset de erros
8	Comando inativo	Máscara da habilitação da rampa
9	Comando inativo	Máscara da habilitação geral
10	Comando inativo	Máscara do sentido de giro
11	Comando inativo	Máscara do comando JOG
12	Comando inativo	Máscara do comando local/remoto
13		Reservado
14		Reservado
15	Comando inativo	Máscara do comando de reset

O comando lógico é dividido em 8 bits superiores, responsáveis por habilitar cada um dos comandos enviados nos 8 bits inferiores. Caso a máscara (nos bits superiores) esteja habilitada, o inversor irá executar o comando indicado no bit inferior correspondente. Caso a máscara esteja desabilitada, o inversor irá desprezar o valor enviado no bit inferior correspondente.

Para controlar as funções do Comando Lógico deve-se ajustar os respectivos parâmetros do inversor com a opção “Fieldbus”:

- Seleção local/remoto - P220;
- Sentido de giro - P223 e/ou P226;
- Habilita Geral, Gira/Pára - P224 e/ou P227;
- Seleção Jog - P225 e/ou P228.

1.5.4.3.2 P371: Referência de Velocidade

Parâmetro de leitura/escrita, através do qual é enviado para o inversor o valor da referência de velocidade via rede. Essa variável é representada usando resolução de 13 bits. Portanto, o valor da referência igual a 8191 (1FFF_{hex}) corresponderá a velocidade síncrona do motor (que equivale a 1800 rpm para motor de 4 pólos e rede 60 Hz). É possível enviar valores superiores ao valor da velocidade síncrona (valores maiores que 13 bits), desde que o valor enviado ao inversor, convertido para rpm, esteja dentro da faixa de valores de referência mínimo e máximo programada no inversor (P133 e P134).

O valor da referência é sempre positivo. Para inverter o sentido de giro usar os bits 2 e 10 do Comando Lógico. Para que o valor da referência seja aceito pelo inversor, é necessário programar os parâmetros P221 e/ou P222 para a opção “Fieldbus”.

1.5.4.3.3 P372: Comando para as Saídas Digitais

Parâmetro de leitura/escrita, para acionamento das saídas digitais do inversor via rede. Palavra de 16 bits, divididos em 8 bits superiores e 8 bits inferiores com a seguinte estrutura:

Nº do Bit	Bit = 0	Bit = 1
0	Saída DO1 inativa	Saída DO1 ativa
1	Saída DO2 inativa	Saída DO2 ativa
2	Saída RL1 inativa	Saída RL1 ativa
3	Saída RL2 inativa	Saída RL2 ativa
4	Saída RL3 inativa	Saída RL3 ativa
5	Saída RL4 inativa	Saída RL4 ativa
6	Saída RL5 inativa	Saída RL5 ativa
7	Reservado	
8	Comando inativo	Máscara de comando para DO1
9	Comando inativo	Máscara de comando para DO2
10	Comando inativo	Máscara de comando para RL1
11	Comando inativo	Máscara de comando para RL2
12	Comando inativo	Máscara de comando para RL3
13	Comando inativo	Máscara de comando para RL4
14	Comando inativo	Máscara de comando para RL5
15	Reservado	

Da mesma forma que para o comando lógico, o acionamento das saídas digitais também é dividido em máscara (bits superiores) e valor das saídas (bits inferiores). O valor para a saída somente será atualizado caso a máscara correspondente nos bits superiores esteja ativa, caso contrário o valor será desprezado.

Para que a saída possa ser acionada via rede, é necessário programar os parâmetros relativos a estas saídas (P275...P280) para a opção "Fieldbus".

1.5.4.3.4 P373: Estado Lógico WEG

Parâmetro de leitura, onde são indicados os estados do inversor, acessível somente via Fieldbus. Formada por 16 bits, divididos em 8 bits inferiores, indicando o código do erro, e 8 bits superiores, indicando os estados do inversor:

Nº do Bit	Bit = 0	Bit = 1
0 ... 7	Código do erro	
8	Rampa desabilitada	Rampa habilitada
9	Desabilitado geral	Habilitado geral
10	Sentido de giro anti-horário	Sentido de giro horário
11	JOG desabilitado	JOG habilitado
12	Local	Remoto
13	Sem subtensão	Com subtensão
14	Regulador PID - Manual	Regulador PID - Automático
15	Sem erro	Com erro

Quando o bit 15 do estado lógico está ativo (indicando inversor com erro), os 8 bits inferiores do estado irão indicar o código do erro, que pode ser um erro de hardware (consulte o manual do usuário no item 7.1 - Erros e Possíveis Causas), ou então um erro software (consulte o item 1.5.4.4 - Erros de software).

1.5.4.3.5 P374: Velocidade do Motor

Parâmetro de leitura, pelo qual o inversor possibilita a leitura da velocidade do motor. Essa variável é mostrada usando resolução de 13 bits mais sinal (em complemento de 2). Portanto o valor nominal será igual a 8191 (1FFF_{hex}) (giro horário) ou -8191 (E001_{hex}) (giro anti-horário) quando o motor estiver girando na velocidade síncrona (ou velocidade base, por exemplo 1800 rpm para motor 4 pólos, 60 Hz). O valor de 13 bits é utilizado apenas como base para a representação, valores de velocidade superiores a 13 bits também poderão ser indicados.

1.5.4.3.6 P375: Estado das Entradas Digitais

Parâmetro de leitura, que permite o monitoramento das entradas digitais do inversor via rede. Palavra de 16 bits, que possui exatamente a mesma estrutura do parâmetro P012. Cada bit possui o seguinte significado:

N° do Bit	Bit = 0	Bit = 1
0	Entrada DI8 inativa	Entrada DI8 ativa
1	Entrada DI7 inativa	Entrada DI7 ativa
2	Entrada DI6 inativa	Entrada DI6 ativa
3	Entrada DI5 inativa	Entrada DI5 ativa
4	Entrada DI4 inativa	Entrada DI4 ativa
5	Entrada DI3 inativa	Entrada DI3 ativa
6	Entrada DI2 inativa	Entrada DI2 ativa
7	Entrada DI1 inativa	Entrada DI1 ativa
8 ... 15	Reservado	

1.5.4.4 Erros de Software

Quando o inversor recebe algum comando indevido via rede, ele indica alguns erros específicos para o mestre, informando qual a causa deste erro. Estas indicações são feitas apenas na palavra de estado lógico (consulte o item 1.5.4.3.4 - P373: Estado lógico WEG), e não são mostradas na HMI do inversor. Podem ocorrer as seguintes indicações de erros:

A124 - Alteração de parâmetro permitida apenas com inversor desabilitado.

- Erro de parametrização.

A125 - provocado por:

- Leitura de parâmetro inexistente, ou
- Escrita em parâmetro inexistente.

A126 - Valor desejado de conteúdo fora da faixa permitida.

A127 - provocado por:

- Função selecionada no Comando Lógico não habilitada para Fieldbus, ou
- Comando de Saída Digital não habilitada para Fieldbus, ou
- Escrita em parâmetro apenas para leitura.

1.6 TEMPOS DE ATUALIZAÇÃO DAS VARIÁVEIS

Os tempos para atualização das variáveis através da rede DeviceNet pode ser dividido em duas partes:

- Tempo para envio e recebimento dos dados via rede: este tempo varia de acordo com uma série de fatores que são dependentes da aplicação que está sendo realizada. Dentre os fatores destacam-se a taxa de comunicação da rede DeviceNet, número de elementos da rede, número de dados trocados com cada elemento.
- Tempo para atualização dos dados recebidos: para o tempo de atualização dos dados,

o inversor acessa e atualiza as informações do cartão de comunicação a cada 20 ms. Os dados mapeados na área de I/O são atualizados de acordo com esta taxa, que deve ser incluída no cálculo do tempo total para a atualização das variáveis.

1.7 ERROS DE COMUNICAÇÃO

Com relação à comunicação do inversor com a rede, podem ocorrer dois tipos de erros: E29 ou E30.

1.7.1 E29: Comunicação Fieldbus Inativa

O erro 29 indica que ocorreu algum tipo de problema de comunicação entre o mestre da rede e o cartão de comunicação. As principais causas deste tipo de erro são:

- Problemas com o cabo de comunicação: o cabo que faz a ligação do mestre com o escravo da rede pode estar rompido, algum ponto pode ter problema de contato, ou a fiação do cabo pode estar invertida.
- Falta de alimentação da rede: a rede DeviceNet possui um par de fios para levar alimentação de 24 Vcc para os escravos, e esta alimentação precisa estar ligada.
- Problemas de configuração com o mestre: o mestre da rede precisa estar ligado e configurado para se comunicar com o inversor.
- Número de words de I/O incorreto: além do mestre estar configurado para se comunicar com o inversor, é necessário também que o número de words de entrada e saída do mestre esteja de acordo com o que foi programado para o inversor. Consulte o item 1.5 - Conteúdo dos Dados, para saber o número de words utilizadas para a comunicação.
- Mestre em IDLE: uma das condições previstas para a indicação de E29 é quando o mestre da rede vai para o estado de IDLE. Neste caso, apesar do cartão de comunicação continuar indicando que está on-line, a indicação de E29 será feita.

1.7.2 E30: Cartão Fieldbus Inativo

O erro 30 indica algum problema de transferência de dados entre o cartão de comunicação e o cartão de controle do inversor. Este erro é indicado principalmente durante a inicialização do inversor. Porém se depois de inicializado, for detectado um intervalo de 1 segundo sem troca de dados entre cartão e inversor (que é feito ciclicamente, independente da comunicação com o mestre), este erro também será indicado. As principais causas deste erro são:

- Problema de configuração do inversor: é necessário configurar corretamente o P309 para a opção Fieldbus desejada, de acordo com o tipo de cartão de comunicação. Para DeviceNet Drive Profile, é necessário programar P309 = 10.
- Problema no posicionamento do cartão: caso o cartão de comunicação não esteja conectado, ou então esteja com problemas na conexão do barramento (mau contato, algum pino torto), o inversor poderá indicar este erro.

**NOTA!**

- Na ocorrência de E29 ou E30, caso o inversor esteja sendo controlado pela rede Fieldbus, será executada a ação programada no P313 - Tipo de bloqueio com E29/E30.
- Para executar o procedimento de auto-ajuste para o modo vetorial, é necessário desabilitar a comunicação, caso contrário poderão ocorrer problemas durante a comunicação.

1.8 CLASSES DE OBJETOS PARA A REDE DEVICENET

Um nó na rede DeviceNet possui uma série de atributos, que estão agrupados em instâncias e classes, através dos quais é possível ter acesso a uma variedade de informações sobre o equipamento. A seguir é descrita a lista de classes e atributos acessíveis via rede para o MVW-01 com cartão de comunicação DeviceNet Drive Profile.

1.8.1 Identity Object, Class 01_{hex}

Atributos da Instância 1

Atributo	Acesso	Nome	Descrição	Padrão	Tipo
1	Get	Vendor ID	Vendor identification - WEG	853	UINT16
2	Get	Device Type	AC/DC Motor	2	UINT16
3	Get	Product Code	Code assigned by the vendor	1	UINT16
4	Get	Revision	Revision of the communication card	101	Struct of UINT8 UINT8
5	Get	Status			UINT16
6	Get	Serial Number	Serial number of the device		UINT32
7	Get	Product Name		MVW-01	SHORT_STRING
9	Get	Configuration Consist. Value	Contents identify configuration of device		UINT6

1.8.2 Message Router Object, Class 02_{hex}

Atributos da Classe

Atributo	Acesso	Nome	Descrição	Padrão	Tipo
1	Get	Revision			Array of UINT8

1.8.3 DeviceNet Object, Class 03_{hex}

Atributos da Instância 1

Atributo	Acesso	Nome	Descrição	Padrão	Tipo
1	Get	MAC ID	Node address		UINT8
2	Get	Baud Rate	Baud Rate of the device		UINT8
5	Get	Allocation Information	Allocation Choice Master's Mac ID		Struct of UINT8 UINT8

1.8.4 Assembly Object, Class 04_{hex}

Instâncias Suportadas

Instância	Tipo	Nome
20	Output	Basic Speed Control Output
70	Input	Basic Speed Control Input
21	Output	Extended Speed Control Output
71	Input	Extended Speed Control Input
100	Output	WEG Basic Speed Control Output
101	Input	WEG Basic Speed Control Input
102	Output	User Specific Output
103	Input	User Specific Input

1.8.5 DeviceNet Connection Object, Class 05_{hex}

Atributos da Instância 1: Explicit Connection Instance

Atributo	Acesso	Nome	Descrição	Padrão	Tipo
1	Get	State	State of the object	1	UINT8
2	Get	Instance Type	Indicates either I/O or Messages Connection	0	UINT8
3	Get	Transport Class Trigger	Defines behaviour of the Connection	83 _{hex}	UINT8
4	Get	Produced Connection ID	Placed in CAN Identifier Field when the Connection transmits		UINT16
5	Get	Consumed Connection ID	CAN Identifier Field value that denotes message to be received		UINT16
6	Get	Initial Communication Characteristics	Defines the message group(s) across which productions and consumptions associated with this Connection occur		UINT8
7	Get	Produced Connection size	Maximum number of bytes transmitted across this Connection	512	UINT16
8	Get	Consumed Connection size	Maximum number of bytes received across this Connection	512	UINT16
9	Get/ Set	Expected Package Rate	Defines timing associated with this Connection		UINT16
12	Get/ Set	Watchdog timeout action	Defines how to handle Inactivity/ Watchdog timeout		UINT8
13	Get	Produced Connection Path Length	Number of Bytes in the produced connection path attribute	256	UINT16
14	Get	Produced Connection Path	Specifies the Application Object(s) whose data is to be produced by these Connection Objects		Array of UINT8
15	Get	Consumed Connection Path Length	Number of bytes in the consumed connection path attribute	256	UINT16
16	Get	Consumed Connection Path	Specifies the Application object(s) that are to receive the data consumed by this Connection object		Array of UINT8
17	Get	Production Inhibit Time	Defines minimum time between new data production.	0	UINT16

Atributos da Instância 2: Polled I/O Connection Instance

Atributo	Acesso	Nome	Descrição	Padrão	Tipo
1	Get	State	State of the object	1	UINT8
2	Get	Instance Type	Indicates either I/O or Messages Connection	0	UINT8
3	Get	Transport Class Trigger	Defines behaviour of the Connection		UINT8
4	Get	Produced Connection ID	Placed in CAN Identifier Field when the Connection transmits		UINT16
5	Get	Consumed Connection ID	CAN Identifier Field value that denotes message to be received		UINT16
6	Get	Initial Communication Characteristics	Defines the message group(s) across which productions and consumptions associated with this Connection occur		UINT8
7	Get	Produced Connection size	Maximum number of bytes transmitted across this Connection		UINT16
8	Get	Consumed Connection size	Maximum number of bytes received across this Connection		UINT16
9	Get/ Set	Expected Package Rate	Defines timing associated with this Connection		UINT16
12	Get	Watchdog timeout action	Defines how to handle Inactivity/ Watchdog timeout		UINT8
13	Get	Produced Connection Path Length	Number of Bytes in the produced connection path attribute	3	UINT16
14	Get	Produced Connection Path	Specifies the Application Object(s) whose data is to be produced by these Connection Objects		Array of UINT8
15	Get	Consumed Connection Path Length	Number of bytes in the consumed connection path attribute	3	UINT16
16	Get	Consumed Connection Path	Specifies the Application object(s) that are to receive the data consumed by this Connection object		Array of UINT8
17	Get	Production Inhibit Time	Defines minimum time between new data production. This attribute is required for I/O Client connection	0	UINT16

Atributos da Instância 2: Change of state/Cyclic Connection Instance

Atributo	Acesso	Nome	Descrição	Padrão	Tipo
1	Get	State	State of the object	1	UINT8
2	Get	Instance Type	Indicates either I/O or Messages Connection	1	UINT8
3	Get	Transport Class Trigger	Defines behaviour of the Connection		UINT8
4	Get	Produced Connection ID	Placed in CAN Identifier Field when the Connection transmits		UINT16
5	Get	Consumed Connection ID	CAN Identifier Field value that denotes message to be received		UINT16
6	Get	Initial Communication Characteristics	Defines the message group(s) across which productions and consumptions associated with this Connection occur		UINT8
7	Get	Produced Connection size	Maximum number of bytes transmitted across this Connection	0	UINT16
8	Get	Consumed Connection size	Maximum number of bytes received across this Connection	0	UINT16
9	Get/ Set	Expected Package Rate	Defines timing associated with this Connection	0	UINT16
12	Get	Watchdog timeout action	Defines how to handle Inactivity/ Watchdog timeout		UINT8
13	Get	Produced Connection Path Length	Number of Bytes in the produced connection path attribute	3	UINT16
14	Get	Produced Connection Path	Specifies the Application Object(s) whose data is to be produced by these Connection Objects		Array of UINT8
15	Get	Consumed Connection Path Length	Number of bytes in the consumed connection path attribute	5	UINT16
16	Get	Consumed Connection Path	Specifies the Application object(s) that are to receive the data consumed by this Connection object		Array of UINT8
17	Get	Production Inhibit Time	Defines minimum time between new data production. This attribute is required for I/O Client connection	0	UINT16

1.8.6 Acknowledge Handler Object, Class 2B_{hex}

Atributos da Instância 1

Atributo	Acesso	Nome	Descrição	Padrão	Tipo
1	Get/ Set	Acknowledge Timer	Time to wait for acknowledge before resending	16	UINT16
2	Get/ Set	Retry Limit	Number of Ack Timeouts to wait before informing the producing application of a Retry_Limit_ Reached event	1	UINT8
3	Get	COS Producing Connection Instance	Connection Instance which contains the path of the producing I/O application object a which will be notified of Ack Handler events		UINT16

1.8.7 Motor Data Object, Class 28_{hex}

Atributos da Instância 1

Atributo	Acesso	Nome	Descrição	Padrão	Tipo
3	Get/ Set	MotorType	0 = Non Standard Motor 1 = PM DC Motor 2 = FC DC Motor 3 = PM Synchronous Motor 4 = FC Synchronous Motor 5 = Switched Reluctance Motor 6 = Wound Rotor Induction Motor 7 = Squirrel Cage Induction Motor 8 = Stepper Motor 9 = Sinusoidal PM BL Motor 10 = Trapezoidal PM BL Motor	7	UINT8
6	Get/ Set	Rated Current	Rates Stator Current from Motor nameplate		UINT16
7	Get/ Set	Rated Voltage	Rated Base Voltage from Motor nameplate		UINT16
9	Get/ Set	Rated Frequency	Rated Electrical Frequency		UINT16
15	Get/ Set	Base Speed	Nominal speed at rated frequency from Motor nameplate		UINT16

1.8.8 Control Supervisor Object, Class 29_{hex}

Atributos da Instância 1

Atributo	Acesso	Nome	Descrição	Padrão	Tipo
3	Get/Set	Run 1	Run forward		BOOL
4	Get/Set	Run 2	Run reverse		BOOL
5	Get/Set	NetCtrl	0 = Local Control 1 = Control from Network		BOOL
6	Get	State	0 = Vendor Specific 1 = Startup 2 = Not ready 3 = Ready 4 = Enabled 5 = Stopping 6 = Fault Stop 7 = Fault		UINT8
7	Get	Running 1	Running forward		BOOL
8	Get	Running 2	Running reverse		BOOL
9	Get	Ready	0 = Other State 1 = Ready or Enabled or Stopping		BOOL
10	Get	Fault	0 = No Faults Present 1 = Fault Occured		BOOL
12	Get/Set	Fault Reset	0 = No Action 0 → 1 = Reset Fault		BOOL
13	Get	Fault Code	If fault is active, this attribute indicates the code for the fault. If fault is not active, it indicates the last error code.		UINT16
15	Get	Crt From Net	0 = Control is local 1 = Control is from Network		BOOL

1.8.9 AC/DC Drive Object, Class 2A_{hex}

Atributos da Instância 1

Atributo	Acesso	Nome	Descrição	Padrão	Tipo
3	Get	At Reference	Frequency arrival		BOOL
4	Get/Set	Net Ref	0 = Set reference not DN Control 1 = Set Reference at DN Control		BOOL
6	Get/Set	Drive Mode	0 = Vendor specific mode 1 = Open loop speed (Frequency) 2 = Closed loop speed control 3 = Torque control 4 = Process control (e.g. PI) 5 = Position control	1	UINT8
7	Get	Speed Actual	Actual drive speed		SINT16
8	Get/Set	Speed Ref	Speed Reference		SINT16
9	Get	Current Actual	Actual current		UINT16
15	Get	Power actual	Actual power		UINT16
16	Get	Input voltage	Input voltage		UINT16
17	Get	Output voltage	Output voltage		UINT16
18	Get/Set	Acceleration time	Acceleration time		UINT16
19	Get/Set	Deceleration time	Deceleration time		UINT16
22	Get/Set	Speed Scale	Speed scaling factor		UINT8
23	Get/Set	Current Scale	Current scaling factor		UINT8
26	Get/Set	Power Scale	Power scaling factor		UINT8
27	Get/Set	Voltage Scale	Voltage scaling factor		UINT8
28	Get/Set	Time Scale	Time scaling factor		UINT8
29	Get	Ref from Net	0 = Local speed reference 1 = DeviceNet speed reference		BOOL

1.8.10 Vendor Specific Object, Class 90_{hex}

Nesta classe estão disponíveis praticamente toda a lista de parâmetros do MVW-01. A classe está dividida em diversas instâncias, e em cada instância é possível acessar um grupo de parâmetros de acordo com a tabela a seguir:

Parâmetros	Número da Instância
P002 ... P099	Instância 1
P100 ... P199	Instância 2
P200 ... P299	Instância 3
P300 ... P399	Instância 4
P400 ... P499	Instância 5
P500 ... P599	Instância 6

Através dos atributos das instâncias, é possível então acessar cada parâmetro do inversor.

Número do parâmetro	Número do atributo correspondente	Número da Instância à qual pertence o atributo
P002	102	Instância 1
P003	103	Instância 1
...
P100	100	Instância 2
P101	101	Instância 2
P102	102	Instância 2
...
P535	135	Instância 6
P536	136	Instância 6

O acesso para leitura ou escrita depende do número do parâmetro acessado. Estes atributos estão mapeados no arquivo EDS de configuração fornecido juntamente com o cartão de comunicação. Através deste arquivo, é possível informar ao software de configuração os endereços para acesso aos parâmetros, e com isso fazer toda a parametrização do inversor.

**NOTA!**

Vale lembrar que não estão disponíveis para acesso via rede os parâmetros P000, P001, P215 e P408.



WEG Equipamentos Elétricos S.A.
Jaraguá do Sul - SC - Brasil
Fone 55 (47) 3276-4000 - Fax 55 (47) 3276-4020
São Paulo - SP - Brasil
Fone 55 (11) 5053-2300 - Fax 55 (11) 5052-4212
automacao@weg.net
www.weg.net



11224126

Documento: 10000598027 / 00