

Inversor de Frequência CFW-08 Manual da Comunicação CANopen

05/2014

Série: CFW-08
Versão: 4.5X
0899.5601 P/3



10413172

IMPORTANTE!

Este manual adicional descreve as funções do protocolo CANopen no inversor de frequência CFW-08. Deve-se atentar que os inversores com esse protocolo devem ter no código inteligente a versão “A3” do cartão de controle conforme descrição exemplo:

MOD.: CFW080040B2024P0**A3Z**

As informações contidas no manual do usuário do CFW-08 para o cartão de controle A1 (CFW-08 Plus) podem ser usadas para essa versão do cartão de controle (A3), porém deve-se atentar para as seguintes modificações:

- Incluído o protocolo de comunicação CANopen.
- Excluídos os protocolos de comunicação seriais (protocolos WEG, Modbus-RTU e IHM remota serial).

A descrição detalhada destas alterações é feita nos itens seguintes.

Sumário

SOBRE O MANUAL	7
ABREVIÇÕES E DEFINIÇÕES	7
REPRESENTAÇÃO NUMÉRICA.....	7
DOCUMENTOS	7
1. COMUNICAÇÃO CANOPEN PARA CFW-08	8
1.1. KIT DE COMUNICAÇÃO CANOPEN PARA CFW-08.....	8
1.1.1. Interface do módulo de comunicação CANopen	8
2. INTRODUÇÃO AO PROTOCOLO CANOPEN	9
2.1. CAN	9
2.1.1. Frame de dados	9
2.1.2. Frame remoto	9
2.1.3. Acesso à rede.....	9
2.1.4. Controle de erros	9
2.1.5. CAN e CANopen	10
2.2. CARACTERÍSTICAS DA REDE CANOPEN.....	10
2.3. MEIO FÍSICO.....	11
2.4. ENDEREÇO NA REDE CANOPEN	11
2.5. ACESSO AOS DADOS	11
2.6. TRANSMISSÃO DE DADOS	11
2.7. OBJETOS RESPONSÁVEIS PELA COMUNICAÇÃO - COBS.....	11
2.8. COB-ID	13
2.9. ARQUIVO EDS.....	14
3. INSTALAÇÃO EM REDE CANOPEN	15
3.1. CONECTOR CANOPEN.....	15
3.2. FONTE DE ALIMENTAÇÃO.....	15
3.3. LIGAÇÃO DO DRIVE COM A REDE.....	16
3.4. RESISTOR DE TERMINAÇÃO.....	16
3.5. TAXA DE COMUNICAÇÃO	17
4. PARÂMETROS DA COMUNICAÇÃO CANOPEN	18
4.1. NOVOS PARÂMETROS DA COMUNICAÇÃO CANOPEN.....	18
4.1.1. P070 – Estado do controlador CAN	18
4.1.2. P074 – Contador de telegramas perdidos	18
4.1.3. P700 – Protocolo CAN	19
4.1.4. P701 – Endereço CAN	19
4.1.5. P702 – Taxa de comunicação CAN	19
4.1.6. P703 – Reset de bus off	20
4.2. PARÂMETROS ALTERADOS NO CFW-08.....	20
4.2.1. P220 – Seleção da fonte local/remoto	20
4.2.2. P221 – Seleção da referência de velocidade – Situação local	21
4.2.3. P222 – Seleção da referência de velocidade – Situação remoto	21
4.2.4. P229 – Seleção de comandos – Situação local.....	22
4.2.5. P230 – Seleção de comandos – Situação remoto.....	22
4.2.6. P231 – Seleção do sentido de giro – Situação local e remoto	22
4.2.7. P313 – Ação para erro de comunicação	23
4.3. VARIÁVEIS BÁSICAS DA COMUNICAÇÃO CANOPEN	23
4.3.1. VB02 – Estado do inversor.....	23
4.3.2. VB03 – Comando para o inversor	24
4.3.3. VB04 – Referência de velocidade	25
5. LEDS DE INDICAÇÃO	26
5.1. TIPOS DE INDICAÇÃO.....	26
5.2. ERROR LED (ERR).....	26
5.3. RUN LED (RUN).....	26
6. DICIONÁRIO DE OBJETOS	28

6.1.	ESTRUTURA DO DICIONÁRIO	28
6.2.	TIPOS DE DADOS	28
6.2.1.	<i>Tipos básicos</i>	29
6.2.2.	<i>Tipos compostos</i>	29
6.2.3.	<i>Tipos estendidos</i>	30
6.3.	COMMUNICATION PROFILE - OBJETOS PARA COMUNICAÇÃO	30
6.4.	MANUFACTURER SPECIFIC - OBJETOS ESPECÍFICOS DO CFW-08	31
7.	DESCRIÇÃO DOS OBJETOS DE COMUNICAÇÃO	32
7.1.	OBJETOS DE IDENTIFICAÇÃO	32
7.1.1.	<i>Objeto 1000h - Device Type</i>	32
7.1.2.	<i>Objeto 1001h - Error Register</i>	32
7.1.3.	<i>Objeto 1018h - Identity object</i>	33
7.2.	SERVICE DATA OBJECTS - SDOs	34
7.2.1.	<i>Objeto 1200h - Servidor SDO</i>	34
7.2.2.	<i>Funcionamento dos SDOs</i>	35
7.3.	PROCESS DATA OBJECTS - PDOs	36
7.3.1.	<i>Objetos mapeáveis para os PDOs</i>	37
7.3.2.	<i>PDOs de recepção</i>	38
7.3.3.	<i>PDOs de transmissão</i>	41
7.4.	NETWORK MANAGEMENT - NMT	44
7.4.1.	<i>Controle dos estados do escravo</i>	44
7.4.2.	<i>Controle de Erros – Node Guarding</i>	46
7.4.3.	<i>Procedimento de inicialização</i>	48
8.	ERROS RELACIONADOS À COMUNICAÇÃO CANOPEN.....	50

Sobre o manual

Este manual fornece a descrição necessária para a operação do inversor de frequência CFW-08 em rede CANopen, utilizando o kit de comunicação CANopen para CFW-08. Este manual deve ser utilizado em conjunto com manual do usuário do CFW-08.

Abreviações e Definições

CAN	Controller Area Network
CiA	CAN in Automation
COB	Communication Object
COB-ID	Communication Object Identifier
SDO	Service Data Object
PDO	Process Data Object
RPDO	Receive PDO
TPDO	Transmit PDO
NMT	Network Management Object
ASCII	American Standard Code for Information Interchange
ro	Read only (somente leitura)
rw	Read/write (leitura e escrita)

Representação numérica

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

Documentos

O protocolo CANopen para CFW-08 foi desenvolvido baseado nas seguintes especificações e documentos:

Documento	Versão	Fonte
CAN Specification	2.0	CiA
CiA DS 301 CANopen Application Layer and Communication Profile	4.02	CiA
CiA DRP 303-1 Cabling and Connector Pin Assignment	1.1.1	CiA
CiA DSP 306 Electronic Data Sheet Specification for CANopen	1.1	CiA

Tabela 1 - Documentação técnica sobre CANopen

Para obter esta documentação, deve-se consultar a CiA (CAN in Automation), que atualmente é a organização que mantém, divulga e atualiza as informações relativas à rede CANopen.

1. Comunicação CANopen para CFW-08

Para que o inversor de frequência CFW-08 possa se comunicar na rede CANopen, é necessária a utilização dos seguintes componentes:

- Inversor de frequência CFW-08 com cartão de controle versão A3.
- Kit de comunicação CANopen para CFW-08.

1.1. Kit de comunicação CANopen para CFW-08

O kit de comunicação CANopen para CFW-08 (KFB-CO-CFW-08, código 417118221) é composto por uma bula de instalação, mais uma IHM com interface CANopen, que deve ser montada no lugar da IHM padrão do inversor.

Caso o kit seja fornecido separadamente, é necessário fazer sua instalação, de acordo com o descrito na bula de instalação presente no kit. Caso o inversor seja fornecido com o cartão de comunicação instalado, basta seguir o descrito neste manual para configuração e operação do equipamento em rede.

1.1.1. Interface do módulo de comunicação CANopen

A interface CANopen disponibiliza os seguintes conectores e LEDs de interface para o CFW-08:

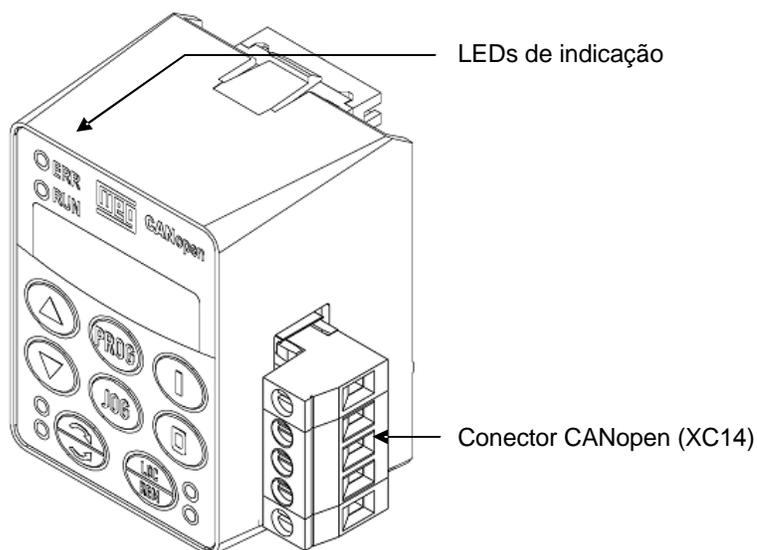


Figura 1 – Módulo de interface CANopen para CFW-08

A explicação detalhada de cada um destes componentes é feita nos itens seguintes.

2. Introdução ao protocolo CANopen

Para a operação do inversor de frequência CFW-08 em rede CANopen, é necessário conhecer a forma como a comunicação é feita. Para isto, este item traz uma descrição geral do funcionamento do protocolo CANopen, contendo as funções utilizadas pelo CFW-08. Para uma descrição detalhada do protocolo, consulte a documentação CANopen indicada na Tabela 1.

2.1. CAN

A rede CANopen é uma rede baseada em CAN, o que significa dizer que ela utiliza telegramas CAN para troca de dados na rede.

O protocolo CAN é um protocolo de comunicação serial que descreve os serviços da camada 2 do modelo ISO/OSI (camada de enlace de dados)¹. Nesta camada, são definidos os diferentes tipos de telegramas (frames), a forma de detecção de erros, validação e arbitragem de mensagens.

2.1.1. Frame de dados

Os dados em uma rede CAN são transmitidos através de um frame (telegrama) de dados. Este tipo de frame é composto principalmente por um campo identificador de 11 bits² (arbitration field), e um campo de dados (data field), que pode conter até 8 bytes de dados.

<i>Identificador</i>	<i>8 bytes de dados</i>							
11 bits	byte 0	byte 1	byte 2	byte 3	byte 4	byte 5	byte 6	byte 7

2.1.2. Frame remoto

Além do frame de dados, existe também o frame remoto (RTR frame). Este tipo de frame não possui campo de dados, apenas o identificador. Ele funciona como uma requisição para que outro dispositivo da rede transmita o frame de dados desejado.

2.1.3. Acesso à rede

Em uma rede CAN, qualquer elemento da rede pode tentar transmitir um frame para a rede em um determinado instante. Caso dois elementos tentem acessar a rede ao mesmo tempo, conseguirá transmitir aquele que enviar a mensagem mais prioritária. A prioridade da mensagem é definida pelo identificador do frame CAN, quanto menor o valor deste identificador, maior a prioridade da mensagem. O telegrama com o identificador 0 (zero) corresponde ao telegrama mais prioritário.

2.1.4. Controle de erros

A especificação CAN define diversos mecanismos para controle de erros, o que a torna uma rede muito confiável e com um índice muito baixo de erros de transmissão que

¹ Na especificação do protocolo CAN, é referenciada a norma ISO 11898 como definição da camada 1 deste modelo (camada física)

² A especificação CAN 2.0 define dois tipos de frames de dados: *standard* (11bits) e *extended* (29 bits). Para o protocolo CANopen do CFW-08, somente frames *standard* são aceitos.

não são detectados. Cada dispositivo da rede deve ser capaz de identificar a ocorrência destes erros, e informar os demais elementos que um erro foi detectado.

Um dispositivo da rede CAN possui contadores internos que são incrementados toda vez que um erro de transmissão ou recepção é detectado, e decrementado quando um telegrama é enviado ou recebido com sucesso. Caso ocorra uma quantidade considerável de erros, o dispositivo pode ser levado para os seguintes estados:

- **Warning:** quando esse contador passa de um determinado limite, o dispositivo entra no estado de *warning*, significando a ocorrência de uma elevada taxa de erros.
- **Error Passive:** quando este valor ultrapassa um limite maior, ele entra no estado de *error passive*, onde ele pára de atuar na rede ao detectar que um outro dispositivo enviou um telegrama com erro.
- **Bus Off :** por último, temos o estado de *bus off*, no qual o dispositivo não irá mais enviar ou receber telegramas.

2.1.5. CAN e CANopen

Somente a definição de como detectar erros, criar e transmitir um frame não é suficiente para definir um significado para os dados que são enviados via rede. É necessário que haja uma especificação que indique como o identificador e os dados devem ser montados e como as informações devem ser trocadas. Desta forma os elementos da rede podem interpretar corretamente os dados que são transmitidos. Neste sentido, a especificação CANopen define justamente como trocar dados entre os equipamentos e como cada dispositivo deve interpretar estes dados.

Existem diversos outros protocolos baseados em CAN, como DeviceNet, J1939, etc., que também utilizam frames CAN para a comunicação. Porém estes protocolos não podem operar em conjunto na mesma rede.

2.2. Características da rede CANopen

Por utilizar um barramento CAN como forma de transmissão de telegramas, todos os dispositivos da rede CANopen têm os mesmos direitos de acesso à rede, onde a prioridade do identificador é responsável por resolver problemas de conflito quando acessos simultâneos ocorrem. Isto traz o benefício de possibilitar a comunicação direta entre escravos da rede, além do fato de que os dados podem ser disponibilizados de maneira mais otimizada, sem a necessidade de um mestre que controle toda a comunicação fazendo acesso cíclico a todos os dispositivos da rede para atualização dos dados.

Outra característica importante é a utilização do modelo produtor / consumidor para a transmissão de dados. Isto significa dizer que uma mensagem que trafega na rede não possui um endereço fixo na rede como destino. Esta mensagem possui um identificador que indica qual o dado que ela está transportando. Qualquer elemento da rede que necessite utilizar desta informação para a sua lógica de operação, poderá consumi-la, e portanto uma mesma mensagem pode ser utilizada por vários elementos da rede ao mesmo tempo.

2.3. Meio físico

O meio físico para a transmissão de sinais em uma rede CANopen é especificado pela norma ISO 11898. Ela define como barramento de transmissão um par trançado com sinal elétrico diferencial.

O inversor de frequência CFW-08 utiliza ainda um circuito de interface com a rede isolado. A fonte de alimentação para a interface CANopen é compartilhada com as entradas e saídas digitais e analógicas presente no cartão de controle do CFW-08. O componente responsável pela transmissão e recepção de sinais é denominado transceiver, que obedece ao especificado pela ISO 11898.

2.4. Endereço na rede CANopen

Toda a rede CANopen deve possuir um mestre, responsável por serviços de gerenciamento da rede, e também pode possuir um conjunto de até 127 escravos. Cada dispositivo da rede também pode ser chamado de nó. Todo escravo em uma rede CANopen é identificado na rede através de seu endereço, ou Node-ID, que deve ser único para cada escravo da rede, e pode variar de 1 até 127.

O CFW-08 não possui funções que implementam os serviços de gerenciamento de rede, e portanto ele deve ser utilizado em conjunto com algum equipamento que possua tais serviços, em geral um mestre da rede CANopen.

2.5. Acesso aos dados

Cada escravo da rede CANopen possui uma lista, denominada dicionário de objetos, que contém todos os dados que são acessíveis via rede. Cada objeto desta lista é identificado através de um índice, e durante a configuração do equipamento e troca de mensagens, este índice é utilizado para identificar o que está sendo transmitido.

Uma descrição mais detalhada de como o dicionário de objetos está estruturado é fornecida no item 6.

2.6. Transmissão de dados

A transmissão de dados numéricos através de telegramas CANopen é feita utilizando a representação hexadecimal do número, e enviando o byte menos significativo do dado primeiro.

Exemplo: transmissão de um inteiro com sinal de 32 bits (12345678h = 305419896 decimal), mais um inteiro com sinal de 16 bits (FF00h = -256 decimal), em um frame CAN.

<i>Identificador</i>	<i>6 bytes de dados</i>					
11 bits	Inteiro 32 bits				Inteiro 16 bits	
	byte 0	byte 1	byte 2	byte 3	byte 4	Byte 5
	78h	56h	34h	12h	00h	FFh

2.7. Objetos responsáveis pela comunicação - COBs

Existe um determinado conjunto de objetos que são responsáveis pela comunicação entre os dispositivos da rede. Estes objetos estão divididos de acordo com os tipos de dados e a forma como são enviados ou recebidos por um dispositivo. O CFW-08 suporta os seguintes objetos de comunicação (COBs):

Tipo de Objeto	Descrição
Service Data Object (SDO)	Os SDOs são objetos responsáveis pelo acesso direto ao dicionário de objetos de um dispositivo. Através de mensagens utilizando os SDOs, é possível indicar explicitamente (através do índice do objeto), qual o dado que está sendo manipulado. Existem dois tipos de SDOs: Cliente SDO, responsável por fazer uma requisição leitura ou escrita para um dispositivo da rede, e o Servidor SDO, responsável por atender esta requisição. Como os SDOs são utilizados geralmente para configuração de um nó da rede, são menos prioritários que outros tipos de mensagens. Somente um SDO do tipo servidor está disponível para o CFW-08.
Process Data Object (PDO)	Os PDOs são utilizados para acessar dados do equipamento sem a necessidade de indicar explicitamente qual o objeto do dicionário está sendo acessado. Para isso, é necessário configurar previamente quais os dados que o PDO estará transmitindo (mapeamento dos dados). Também existem dois tipos de PDOs: PDO de recepção e PDO de transmissão. PDOs usualmente são utilizados para transmissão e recepção de dados utilizados durante a operação do dispositivo, e por isso são mais prioritários que os SDOs.
Emergency Object (EMCY)	Não disponível para o CFW-08.
Synchronization Object (SYNC)	Não disponível para o CFW-08
Network Management (NMT)	Toda a rede CANopen precisa ter um mestre que controle os demais dispositivos da rede (escravos). Este mestre será responsável por um conjunto de serviços que controlam a comunicação dos escravos e seu estado na rede CANopen. Os escravos são responsáveis por receber os comandos enviados pelo mestre e executar as ações solicitadas. O CFW-08 opera como um escravo da rede CANopen, e disponibiliza dois tipos de serviços que o mestre pode utilizar: serviços de controle do dispositivo, onde o mestre controla o estado de cada escravo na rede, e serviços de controle de erros (<i>Node Guarding</i>), onde o escravo envia mensagens periódicas para o mestre para informar que a conexão está ativa.

Tabela 2 - Tipos de Objetos de Comunicação - COBs

Toda a comunicação do inversor com a rede é feita utilizando-se estes objetos, e os dados que podem ser acessados são os existentes no dicionário de objetos do dispositivo. A descrição do funcionamento de cada COB é feita no item 7. O modelo do funcionamento do inversor de frequência CFW-08, do ponto de vista da interface com a rede CANopen, pode ser descrito pela seguinte figura:

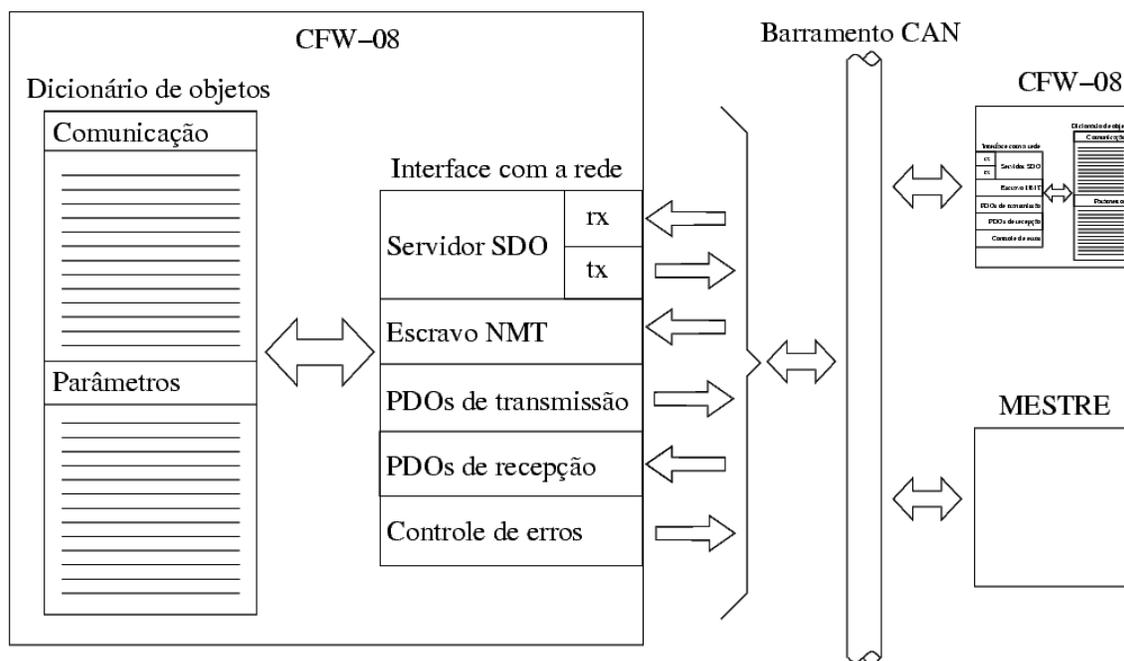


Figura 2 - Modelo de funcionamento do CFW-08 em rede CANopen

2.8. COB-ID

Um telegrama da rede CANopen sempre é transmitido por um objeto de comunicação (COB). Todo COB possui um identificador que indica o tipo de dado que está sendo transportado. Este identificador, chamado de COB-ID, possui um tamanho de 11 bits, e é transmitido no campo identificador de um telegrama CAN. Ele pode ser subdividido em duas partes:

Código da Função				Endereço do nó						
bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0

- *Código da função*: indica o tipo de objeto que está sendo transmitido.
- *Endereço do nó*: indica com qual dispositivo da rede o telegrama está vinculado.

A seguir é apresentada uma tabela com os valores padrão para os diferentes objetos de comunicação disponíveis no CFW-08. É necessário observar que o valor padrão do objeto depende do endereço do escravo, com exceção dos COB-IDs para NMT e SYNC, que são comuns para todos os elementos da rede. Estes valores também podem ser alterados durante a etapa de configuração do dispositivo.

COB	Código da Função (bits 10 - 7)	COB-ID Resultante (função + endereço)
NMT	0000	0
SYNC	0001	128 (80h)
EMCY	0001	129 - 255 (81h - FFh)
PDO1 (tx)	0011	385 - 511 (181h - 1FFh)
PDO1 (rx)	0100	513 - 639 (201h - 27Fh)
PDO2 (tx)	0101	641 - 767 (281h - 2FFh)
PDO2 (rx)	0110	769 - 895 (301h - 37Fh)
PDO3 (tx)	0111	897 - 1023 (381h - 3FFh)

PDO3 (rx)	1000	1025 - 1151 (401h - 47Fh)
PDO4 (tx)	1001	1153 - 1279 (481h - 4FFh)
PDO4 (rx)	1010	1281 - 1407 (501h - 57Fh)
SDO (tx)	1011	1409 - 1535 (581h - 5FFh)
SDO (rx)	1100	1537 - 1663 (601h - 67Fh)
Node Guarding	1110	1793 - 1919 (701h - 77Fh)

Tabela 3 - COB-ID para os diferentes objetos

2.9. Arquivo EDS

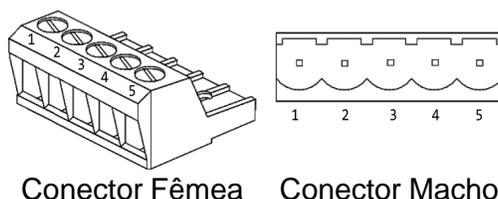
Cada dispositivo em uma rede CANopen possui um arquivo de configuração EDS, que contém diversas informações sobre o funcionamento do dispositivo na rede CANopen, bem como a descrição de todos os objetos existentes para comunicação. Em geral este arquivo é utilizado por um mestre ou software de configuração, para programação dos dispositivos presentes na rede CANopen.

O arquivo de configuração EDS para o CFW-08 é fornecido juntamente com o produto, e também pode ser obtido através do site <http://www.weg.com.br>. É necessário observar a versão de software do inversor, para utilizar um arquivo EDS que seja compatível com esta versão.

3. Instalação em rede CANopen

3.1. Conector CANopen

O módulo de interface CANopen possui um conector de 5 vias localizado na lateral do mesmo (XC14). A tabela a seguir descreve a função de cada pino deste conector:



Conector Fêmea Conector Macho
Figura 3 - Conector para rede CANopen

Conector CANopen	
1	CAN_GND (V-)
2	CAN_L
3	Shield
4	CAN_H
5	CAN_VCC (V+)

Tabela 4 - Pinagem do conector CANopen para CFW-08

- CAN_GND e CAN_VCC: conectores para fonte de alimentação externa de 24V, utilizada para alimentar o circuito de interface CANopen.
- CAN_L e CAN_H: são utilizados para transmissão dos dados, e devem ser conectados em todos os dispositivos da rede (CAN_L conectado ao CAN_L e CAN_H conectado ao CAN_H).
- Shield: neste pino deve-se conectar a blindagem do cabo.



NOTA!

Para melhorar o sinal de referência e a imunidade a ruídos da rede, deve-se também aterrar o borne 5 do cartão de controle.

3.2. Fonte de Alimentação

Os módulos de interface CAN necessitam de uma tensão de alimentação externa entre os pinos 1 e 5 do conector da rede. Os dados para consumo individual e tensão de entrada são apresentados na tabela a seguir.

Tensão de alimentação (V_{CC})		
<i>Mínimo</i>	<i>Máximo</i>	<i>Recomendado</i>
11	30	24
Corrente (mA)		
<i>Típico</i>		<i>Máximo</i>
30		50

Tabela 5 - Características da alimentação para interface CAN

3.3. Ligação do drive com a rede

Para interligar os diversos nós da rede, recomenda-se a conexão do equipamento diretamente a partir da linha principal, sem a utilização de derivações. Durante a instalação dos cabos, deve-se evitar sua passagem próximo a cabos de potência, pois devido a interferência eletromagnética, isto facilita a ocorrência de erros durante a transmissão. Para evitar problemas de circulação de corrente por diferença de potencial entre diferentes aterramentos, é necessário que todos os dispositivos estejam conectados no mesmo ponto de terra.

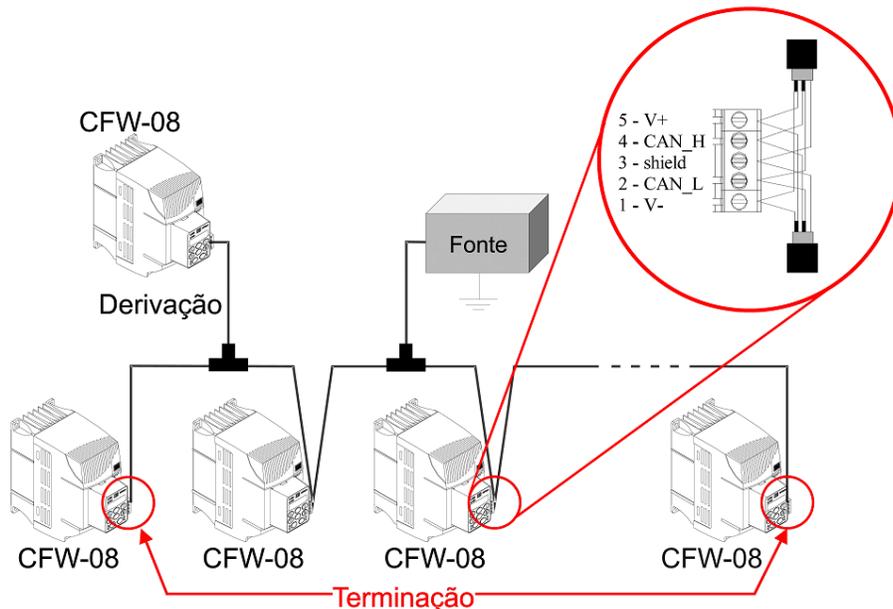


Figura 4 - CFW-08 em rede CANopen

O cabo para a ligação dos sinais CAN_L e CAN_H deve ter impedância característica de aproximadamente 120 Ohm, e um atraso máximo de propagação do sinal de 5ns/m. Outras características dependem do comprimento do cabo, que deve estar de acordo com a tabela a seguir.

Comprimento do cabo (m)	Resistência por metro (mOhm/m)	Área do condutor (mm²)
0 ... 40	70	0.25 ... 0.34
40 ... 300	<60	0.34 ... 0.60
300 ... 600	<40	0.50 ... 0.60
600 ... 1000	<26	0.75 ... 0.80

O número máximo de dispositivos conectados em um único segmento da rede é limitado em 64. Repetidores podem ser utilizados para conectar um número maior de dispositivos.

3.4. Resistor de terminação

As extremidades do barramento CAN devem possuir um resistor de terminação no valor de $120 \cdot / 0.25W$, conectando os sinais CAN_H e CAN_L.

3.5. Taxa de comunicação

A taxa de comunicação (*baud rate*) que pode ser utilizada por um equipamento na rede CANopen depende do comprimento do cabo utilizado na instalação. A tabela a seguir mostra as taxas de comunicação disponíveis para o CFW-08, e o comprimento máximo de cabo que pode ser utilizado na instalação, de acordo com o recomendado pela CiA.

<i>Taxa de comunicação</i>	<i>Comprimento do cabo</i>
1 Mbit/s (não recomendado)	40 m
500 Kbit/s (não recomendado)	100 m
250 Kbit/s	250 m
125 Kbit/s	500 m
100 Kbit/s	600 m
50 Kbit/s	1000 m
20 Kbit/s	1000 m
10 Kbit/s	1000 m

Tabela 6 - Taxas de comunicação e tamanho da instalação

4. Parâmetros da comunicação CANopen

A seguir serão descritos os parâmetros do CFW-08 que possuem relação com a comunicação CANopen. Os demais parâmetros não citados aqui não possuem relação direta com esta função, porém são importantes para a operação do inversor via rede. Deve-se consultar o manual do usuário do CFW-08 para a lista completa de parâmetros e sua descrição.

4.1. Novos parâmetros da comunicação CANopen

O inversor de frequência CFW-08 possui um conjunto de parâmetros, descritos a seguir, para a configuração do dispositivo em rede CANopen. Estes parâmetros somente são acessíveis caso utilize-se o cartão de controle A3, juntamente com o módulo de comunicação CANopen para o CFW-08.

4.1.1. P070 – Estado do controlador CAN

Fornece a informação do estado do dispositivo com relação ao barramento CAN. Ele indicará se o controlador está operando corretamente, ou então informará o tipo de erro que o inversor apresenta com relação à comunicação.

<i>Parâmetro</i>	<i>Descrição</i>	<i>Faixa</i>	<i>Padrão</i>	<i>Acesso</i>
P070	Estado do controlador CAN	0 = Interface CAN desabilitada 1 = Reservado 2 = Interface CAN ativa 3 = Warning 4 = Error Passive 5 = Bus off 6 = Sem alimentação	-	Somente Leitura

- **0 – Interface CAN desabilitada:** o protocolo CANopen não foi programado no parâmetro P700. É importante lembrar a alteração do P700 somente é válida após realizar o reset do inversor.
- **2 – Interface CAN ativa:** a comunicação via interface CAN está ativa e funcionando sem erros.
- **3 – Warning:** uma quantidade razoável de erros de transmissão ou recepção foi detectada via interface CAN.
- **4 – Error Passive:** uma grande quantidade de erros de transmissão ou recepção foi detectada via interface CAN. Pode atingir este estado, por exemplo, se o CFW-08 for o único equipamento ativo conectado na rede.
- **5 – Bus off:** a quantidade de erros de transmissão ou recepção ultrapassou o limite máximo, desabilitando a interface CAN. É necessário fazer o reset do equipamento para reiniciar a comunicação. Pode ocorrer por instalação incorreta dos cabos, curto circuito entre os sinais de comunicação, falta de resistor de terminação, etc.
- **6 – Sem alimentação:** a alimentação para a interface CAN é fornecida pelo cabo do módulo CANopen conectado ao CFW-08 durante sua instalação. Caso o parâmetro apresente este estado, remova o módulo CAN e verifique instalação do mesmo.

4.1.2. P074 – Contador de telegramas perdidos

Contador que indica quantos telegramas CAN foram perdidos, devido ao fato do inversor não conseguir tratar o telegrama antes que um outro telegrama fosse enviado para a rede.

Parâmetro	Descrição	Faixa	Padrão	Acesso
P074	Contador de telegramas perdidos	0 ... 9999	-	Somente leitura

Em condições normais de operação, este contador deve permanecer com o valor 0 (zero). Porém em condições onde se exige muito processamento do inversor – tipicamente em frequências de chaveamento a partir de 10KHz e em modo vetorial – caso a taxa de comunicação seja elevada, podem ocorrer perdas de telegramas. Neste caso, pode ser necessário diminuir a taxa de comunicação ou diminuir a frequência com que os telegramas são enviados para a rede.

4.1.3. P700 – Protocolo CAN

Permite selecionar o protocolo da camada de aplicação desejada para a interface CAN. É necessário programar a opção '1' para habilitar o protocolo CANopen para o CFW-08.

Parâmetro	Descrição	Faixa	Padrão	Acesso
P700	Protocolo CAN	0 = Desabilitado 1 = CANopen 2 = Reservado	-	Leitura / escrita



NOTA!

A alteração deste parâmetro somente terá efeito caso o inversor seja desligado e ligado novamente.

4.1.4. P701 – Endereço CAN

Permite selecionar o endereço (Node ID) do CFW-08 na rede CANopen.

Parâmetro	Descrição	Faixa	Padrão	Acesso
P701	Endereço CAN	1 ... 127	63	Leitura / escrita

Cada dispositivo da rede precisa ter um endereço diferente dos demais, e portanto podem existir até 127 equipamentos conectados em uma mesma rede CANopen.



NOTA!

A alteração deste parâmetro somente terá efeito caso o inversor seja desligado e ligado novamente.

4.1.5. P702 – Taxa de comunicação CAN

Permite selecionar a taxa de comunicação (*baud rate*) para a interface CAN do CFW-08.

Parâmetro	Descrição	Faixa	Padrão	Acesso
P702	Taxa de comunicação CAN	0 = 1 Mbit/s (não recomendado) 1 = Reservado 2 = 500 Kbit/s (não recomendado) 3 = 250 Kbit/s 4 = 125 Kbit/s 5 = 100 Kbit/s 6 = 50 Kbit/s 7 = 20 Kbit/s 8 = 10 Kbit/s	0	Leitura / escrita

Para que os dispositivos da rede possam se comunicar, é necessário que todos possuam a mesma taxa de comunicação configurada. Não esquecer também que existe uma limitação da taxa de comunicação de acordo com o comprimento do cabo usado na instalação (ver Tabela 6).



NOTA!

A alteração deste parâmetro somente terá efeito caso o inversor seja desligado e ligado novamente.

4.1.6. P703 – Reset de bus off

Permite escolher qual deve ser a ação do inversor caso ocorra erro de bus off.

Parâmetro	Descrição	Faixa	Padrão	Acesso
P703	Reset de bus off	0 = manual 1 = automático	0	Leitura / escrita

- **0 – Manual:** caso o inversor detecte erro de bus off, será indicado E34 na IHM do produto, e executada a ação programada no parâmetro P313. O inversor somente voltará a comunicar após o operador fazer o reset manual do inversor.
- **1 – Automático:** caso seja detectado erro de bus off, o inversor reiniciará automaticamente a comunicação. Neste caso, nenhuma ação ou indicação na IHM é feita.



NOTA!

Os parâmetros P070, P074, P700, P701, P702 e P703 somente são acessíveis caso utilize-se o cartão de controle A3, juntamente com o módulo de comunicação CANopen para o CFW-08.

4.2. Parâmetros alterados no CFW-08

Além dos parâmetros descritos no item anterior, outros parâmetros que possuem relação direta com a comunicação CANopen do CFW-08 tiveram suas funções alteradas, conforme descrito a seguir.

4.2.1. P220 – Seleção da fonte local/remoto

Permite definir qual a fonte de seleção entre o modo de operação local e remoto do CFW-08.

Parâmetro	Descrição	Faixa	Padrão	Acesso
P220	Seleção da fonte Local/Remoto	0 = Sempre Local 1 = Sempre Remoto 2 = Tecla HMI-CFW08-P (default: local) 3 = Tecla HMI-CFW08-P (default: remoto) 4 = DI2 a DI4 5 = CANopen (default: local) 6 = CANopen (default: remoto)	2	Leitura / escrita

Para que seja possível alterar o modo de operação via rede CANopen, é necessário programar este parâmetro com a opção 5 ou 6. O modo de operação *default* indica qual o modo de operação após a energização ou reset do inversor.



NOTA!

As opções 5 e 6 possuem funções diferentes das descritas no manual do usuário, devido ao fato do cartão de controle A3 não possuir funções relativas aos protocolos seriais (protocolos WEG, Modbus-RTU e IHM remota serial). Alterações semelhantes foram feitas nos parâmetros P221, P222, P229 e P230.

4.2.2. P221 – Seleção da referência de velocidade – Situação local

Permite definir, para o modo de operação local, qual deve ser a fonte para a referência de velocidade do inversor.

Parâmetro	Descrição	Faixa	Padrão	Acesso
P221	Seleção da referência de velocidade – Situação local	0 = Teclas HMI 1 = AI1 2, 3 = AI2 4 = E.P. 5 = CANopen 6 = Multispeed 7 = Soma AI ≥ 0 8 = Soma AI	0	Leitura / escrita

Para que seja possível indicar a referência de velocidade via rede CANopen em modo local, deve-se programar este parâmetro com a opção 5.

4.2.3. P222 – Seleção da referência de velocidade – Situação remoto

Permite definir, para o modo de operação remoto, qual deve ser a fonte para a referência de velocidade do inversor.

Parâmetro	Descrição	Faixa	Padrão	Acesso
P222	Seleção da referência de velocidade – Situação remoto	0 = Teclas HMI 1 = AI1 2, 3 = AI2 4 = E.P. 5 = CANopen 6 = Multispeed 7 = Soma AI ≥ 0 8 = Soma AI	0	Leitura / escrita

Para que seja possível indicar a referência de velocidade via rede CANopen em modo remoto, deve-se programar este parâmetro com a opção 5.

4.2.4. P229 – Seleção de comandos – Situação local

Permite definir, para o modo de operação local, qual a fonte de comandos do inversor.

Parâmetro	Descrição	Faixa	Padrão	Acesso
P229	Seleção de comandos – Situação local	0 = Tecla HMI-CFW08-P 1 = Bornes 2 = CANopen	0	Leitura / escrita

Para que seja possível comandar o inversor via rede CANopen em modo local, deve-se programar este parâmetro com a opção 2.

4.2.5. P230 – Seleção de comandos – Situação remoto

Permite definir, para o modo de operação remoto, qual a fonte de comandos do inversor.

Parâmetro	Descrição	Faixa	Padrão	Acesso
P230	Seleção de comandos – Situação remoto	0 = Tecla HMI-CFW08-P 1 = Bornes 2 = CANopen	0	Leitura / escrita

Para que seja possível comandar o inversor via rede CANopen em modo remoto, deve-se programar este parâmetro com a opção 2.

4.2.6. P231 – Seleção do sentido de giro – Situação local e remoto

Permite definir como é controlado o sentido de giro.

Parâmetro	Descrição	Faixa	Padrão	Acesso
P231	Seleção do sentido de giro – Situação local e remoto	0 = Horário 1 = Anti-horário 2 = Comandos, conforme P229 e P230	2	Leitura / escrita

Para que seja possível controlar o sentido de giro via rede CANopen, deve-se programar este parâmetro com a opção 2.

4.2.7. P313 – Ação para erro de comunicação

Quando o CFW-08 está sendo operado via rede, caso um erro de comunicação ocorra é possível programar no P313 uma ação que o inversor executará automaticamente no caso de falha da comunicação.

Parâmetro	Descrição	Faixa	Padrão	Acesso
P313	Ação para erro de comunicação	0 = Desabilita por rampa 1 = Desabilita geral 2 = Somente indica erro 3 = Vai para modo local	2	Leitura / escrita

Para a comunicação CANopen, são considerados erros de comunicação os erros 33 (sem alimentação), erro 34 (bus off) e erro 35 (*Node Guarding*).

- **0 – Desabilita por rampa:** o inversor para o motor de acordo com a rampa de desaceleração programada
- **1 – Desabilita geral:** o inversor é desabilitado e o motor para por inércia.
- **2 – Somente indica erro:** caso ocorra um dos erros citados, o inversor indica erro no display e permanece no estado atual.
- **3 – Vai para modo local:** caso o inversor esteja em modo remoto, este passa automaticamente para modo local.



NOTA!

- O inversor somente executará a ação indicada se esta ação estiver programada para ser executada via rede CANopen.
- Para o erro de bus off (E34), a ação somente será executada se o parâmetro P703 estiver programado com a opção 0 (zero).

4.3. Variáveis básicas da comunicação CANopen

Além dos parâmetros, também foram disponibilizadas algumas variáveis básicas, acessíveis somente via interface CANopen. Utilizando estas variáveis é possível monitorar os estados e enviar comandos e referência de velocidade via rede CANopen. Estas variáveis são mapeadas como objetos da comunicação CANopen, e devem ser acessadas utilizando os mecanismos de comunicação descritos no capítulo seguinte.

4.3.1. VB02 – Estado do inversor

A variável básica 2 é uma variável de leitura que representa o estado do inversor de frequência CFW-08. Possui 16 bits, que pode ser dividido em dois bytes:

- **Byte inferior:** representa o código de erros do inversor. Caso o drive esteja no estado de erro (bit 15 = 1), os bits 0 até 7 indicarão o código do erro atual. Exemplos:
 - Byte inferior = 2, CFW-08 está com E02 – subtensão no circuito intermediário.
 - Byte inferior = 6, CFW-08 está com E06 – erro externo.
- **Byte superior:** representa o estado do inversor. Cada bit possui um significado, de acordo com a tabela a seguir:

Bit	Função	Descrição
8	Habilitado por rampa	Indica se a rampa do inversor está habilitada ou não. Bit 8 = 0: habilita por rampa (gira/pára) inativo. Bit 8 = 1: habilita por rampa ativo.
9	Habilitado geral	Indica se o inversor está habilitado geral ou não. Bit 9 = 0: habilita geral inativo. Bit 9 = 1: habilita geral ativo.
10	Sentido de giro	Indica o sentido de giro do motor. Bit 10 = 0: sentido de giro anti-horário. Bit 10 = 1: sentido de giro horário.
11	JOG	Indica se a função JOG está habilitada. Bit 11 = 0: JOG inativo. Bit 11 = 1: JOG ativo.
12	Local/remoto	Indica qual o modo de operação do inversor. Bit 12 = 0: em modo local. Bit 12 = 1: em modo remoto.
13	Subtensão	Indica se o inversor possui ou não subtensão. Bit 13 = 0: sem subtensão. Bit 13 = 1: com subtensão.
14	Reservado	
15	Em erro	Indica se o drive está em estado de erro ou não. Bit 15 = 0: sem erro. Bit 15 = 1: com erro. O código do erro pode ser lido através do byte inferior desta mesma palavra.

4.3.2. VB03 – Comando para o inversor

A variável básica 3 permite enviar comandos (partir/parar, alterar sentido de giro, etc.) para o CFW-08. Possui 16 bits com as seguintes funções:

Bit	Função	Descrição
0	Habilita rampa	Permite habilitar a rampa do inversor. Bit 0 = 0: desabilita por rampa (pára). Bit 0 = 1: habilita a rampa (gira). Obs.: para executar esta função, é necessário que o inversor esteja habilitado geral.
1	Habilita geral	Permite executar o comando habilita geral. Bit 1 = 0: desabilita geral (pára por inércia). Bit 1 = 1: habilita geral.
2	Sentido de giro	Permite selecionar o sentido de giro do drive. Bit 2 = 0: sentido anti-horário. Bit 2 = 1: sentido horário.
3	JOG	Possibilita acionar a função JOG. Bit 3 = 0: JOG inativo. Bit 3 = 1: JOG ativo.
4	Local/remoto	Permite selecionar o modo de operação. Bit 4 = 0: local. Bit 4 = 1: remoto.
5 ... 6	Reservado	

7	Reset de erros	Caso o inversor esteja em estado de erro, este bit permite fazer o reset de erros do dispositivo. Bit 7 = 0: sem ação. Bit 7 = 1: faz reset de erros do inversor.
---	----------------	---



NOTA!

- Os comandos acima somente serão executados se os mesmos estiverem programados para operar via CANOpen.
- O reset de erros do drive fará com que o drive seja reiniciado, incluindo o reset da comunicação CANOpen.
- Para a comunicação CANOpen, não é necessária a utilização dos bits da máscara, como ocorria para a comunicação serial.

4.3.3. VB04 – Referência de velocidade

A variável básica 4 permite programar a referência de frequência desejada para o CFW-08 via rede CANOpen. Esta variável utiliza duas casas decimais de resolução para indicar a referência de frequência.

Exemplos:

- Referência de velocidade desejada: 5,00Hz è valor enviado via CANOpen: 500.
- Referência de velocidade desejada: 32,50Hz è valor enviado via CANOpen: 3250.

5. LEDs de Indicação

O módulo de comunicação CANopen possui dois LEDs de indicação do estado da rede:

- Error LED (ERR): faz a indicação de estados de erro da comunicação CANopen.
- Run LED (RUN): indica o estado do escravo na rede CANopen.

Estes LEDs fornecem informações importantes sobre o funcionamento do CFW-08 em rede CANopen. Durante a inicialização do inversor, caso o protocolo CANopen esteja habilitado, ambos os LEDs são acesos para teste por um período de aproximadamente 200 ms. Após este período, eles farão indicações conforme mostrado a seguir.

5.1. Tipos de indicação

Além dos estados aceso e apagado, os seguintes comportamentos também podem ser visualizados:

- **Intermitente:** o LED fica um período de 200 ms aceso, seguido de um período de 200 ms apagado.
- **Uma piscada:** o LED fica um período de 200 ms aceso, seguido de um período de 1 segundo apagado.
- **Duas piscadas:** o LED acende duas vezes por um período de 200 ms (com um período de 200 ms apagado entre estas indicações), seguido de um período de 1 segundo apagado.

5.2. Error LED (ERR)

O LED de erro CANopen indica erros da camada física do barramento CAN, além de erros da comunicação CANopen. Possui cor vermelha.

Indicação	Estado	Descrição
Apagado	Sem erro	O inversor está operando normalmente
Uma piscada	Atingido o estado de <i>warning</i>	Os contadores internos de erro do controlador CAN atingiram o estado de <i>warning</i> , devido a erros da comunicação CAN. Esta indicação também é válida caso o inversor esteja no estado <i>error passive</i> .
Duas piscadas	Falha no serviço de controle de erros – <i>Node Guarding</i>	Após o serviço de <i>Node Guarding</i> ter sido inicializado pelo mestre, ocorreu <i>timeout</i> na troca de telegramas entre o mestre e o escravo, ocasionando este erro.
Aceso	Bus off	O controlador CAN atingiu o estado de bus off.

Tabela 7- Indicações do LED de erro

5.3. Run LED (RUN)

O LED de operação CANopen indica o estado do escravo na comunicação CANopen, conforme indicado no item 7.4.1. Possui cor verde.

Indicação	Estado	Descrição
Uma piscada	Estado parado (STOPPED)	O inversor encontra-se no estado parado.
Intermitente	Pré-operacional (PRE- OPERATIONAL)	O inversor encontra-se no estado pré-operacional.
Aceso	Operacional (OPERATIONAL)	O inversor encontra-se no estado operacional.

Tabela 8- Indicações do LED de erro

6. Dicionário de Objetos

O dicionário de objetos é uma lista com os diversos dados do equipamento que são acessíveis via rede CANopen. Um objeto desta lista é identificado através de um índice de 16 bits, e é baseado nesta lista que toda a troca de dados entre os dispositivos é efetuada.

O documento CiA DS 301 define um conjunto mínimo de objetos que todo o escravo da rede CANopen deve possuir. Os objetos disponíveis nesta lista são agrupados de acordo com o tipo de função que ele executa. Os objetos são dispostos no dicionário da seguinte maneira:

Índice	Objetos	Descrição
0001h - 0360h	Definição dos tipos de dados	Utilizado como referência para os tipos de dados suportados pelo sistema.
1000h - 1FFFh	Objetos de comunicação	São objetos comuns a todos os dispositivos CANopen. Contêm informações gerais sobre o equipamento e também dados para a configuração da comunicação.
2000h - 5FFFh	Objetos específicos do fabricante	Nesta faixa, cada fabricante de equipamentos CANopen é livre para definir quais dados estes objetos irão representar.
6000h - 9FFFh	Objetos padronizados para dispositivos	Esta faixa é reservada para objetos que descrevem o comportamento de equipamentos similares, independente do fabricante. Não utilizada pelo CFW-08.

Tabela 9 - Agrupamentos do dicionário de objetos

Demais índices não referenciados nesta lista são reservados para uso futuro.

6.1. Estrutura do dicionário

A estrutura geral do dicionário de objetos possui o seguinte formato:

Índice	Objeto	Nome	Tipo	Acesso
--------	--------	------	------	--------

- **Índice:** indica diretamente o índice do objeto no dicionário.
- **Objeto:** descreve que informação o índice armazena (variável simples, array, record, etc.)
- **Nome:** contém o nome do objeto para facilitar sua identificação.
- **Tipo:** indica diretamente o tipo de dado armazenado. Para variáveis simples, este tipo pode ser um inteiro, um float, etc. Para arrays, ele indica o tipo do dado contido no array. Para records, ele indica o formato do record, de acordo com os tipos descritos na primeira parte do dicionário de objetos (índices 0001h - 0360h).
- **Acesso:** informa se o objeto em questão está acessível somente para leitura (ro), para leitura e escrita (rw), ou é uma constante (const). Para objetos do tipo array ou records, ainda é necessário um sub-índice, que não é descrito na estrutura do dicionário.

6.2. Tipos de dados

A primeira parte do dicionário de objetos (índices 0001h - 0360h) descreve os tipos de dados que podem ser acessados em um dispositivo na rede CANopen. Estes podem ser tipos básicos, como inteiros e floats, ou tipos compostos, formados por um conjunto de entradas, como records e arrays. A seguir são apresentados os tipos de objetos utilizados pelo CFW-08.

6.2.1. Tipos básicos

Os tipos básicos de dados suportados são os seguintes:

- Inteiros com sinal : existem três tipos de inteiros com sinal suportados pelo CFW-08, INTEGER8, INTEGER16 e INTEGER32, que representam, respectivamente, inteiros com 8, 16 e 32 bits de dados. Inteiros com sinal são calculados utilizando complemento de dois, e durante a transmissão, sempre o byte menos significativo é transmitido primeiro em um telegrama CAN.
- Inteiros sem sinal : existem três tipos de inteiros sem sinal suportados pelo CFW-08, UNSIGNED8, UNSIGNED16 e UNSIGNED32, que representam, respectivamente, inteiros com 8, 16 e 32 bits de dados. Também durante a transmissão, sempre o byte menos significativo é transmitido primeiro.

6.2.2. Tipos compostos

É possível formar novos tipos de dados através do agrupamento de tipos básicos em listas (arrays - formados por um único tipo de dado) e estruturas (records - formado por diversos tipos de dados). Neste caso, cada item deste tipo é identificado através de um sub-índice. Os tipos compostos utilizados pelo inversor são listados abaixo.

- **PDO COMM PARAMETER:** este record define as informações necessárias para configurar um PDO para a comunicação CANopen. O conteúdo e forma como cada campo é utilizado são detalhados no item 7.3.

Sub-índice	Descrição da entrada	Tipo
00h	Número de entradas suportadas neste record	UNSIGNED8
01h	COB-ID	UNSIGNED32
02h	Transmission type	UNSIGNED8
03h	Inhibit time	UNSIGNED16
04h	Reservado	UNSIGNED8
05h	Event timer	UNSIGNED16

Tabela 10 - Record para configuração dos PDOs

- **PDO MAPPING:** este record define como mapear os dados que serão transmitidos por um PDO durante a comunicação CANopen. O conteúdo e forma como cada campo é utilizado são detalhados no item 7.3.

Sub-índice	Descrição da entrada	Tipo
00h	Número de objetos mapeados no PDO	UNSIGNED8
01h	1o objeto mapeado	UNSIGNED32
02h	2o objeto mapeado	UNSIGNED32
...
40h	64o objeto mapeado	UNSIGNED32

Tabela 11 - Record para mapeamento dos dados de um PDO

- **SDO PARAMETER**: este record define as informações necessárias para configurar um SDO para a comunicação CANOpen. O conteúdo e forma como cada campo é utilizado são detalhados no item 7.2.

Sub-índice	Descrição da entrada	Tipo
00h	Número de entradas suportadas neste record	UNSIGNED8
01h	COB-ID cliente → servidor	UNSIGNED32
02h	COB-ID servidor → cliente	UNSIGNED32
03h	Node-ID do cliente/servidor	UNSIGNED8

Tabela 12 - Record para configuração dos SDOs

- **IDENTITY** : este record é utilizado para descrever o tipo de dispositivo presente na rede.

Sub-índice	Descrição da entrada	Tipo
00h	Número de entradas suportadas neste record	UNSIGNED8
01h	Vendor-ID	UNSIGNED32
02h	Product Code	UNSIGNED32
03h	Revision Number	UNSIGNED32
04h	Serial Number	UNSIGNED32

Tabela 13 - Record para identificação do dispositivo

6.2.3. Tipos estendidos

O CFW-08 não possui tipos estendidos.

6.3. Communication Profile - Objetos para comunicação

Os índices de 1000h até 1FFFh correspondem, no dicionário de objetos, à parte responsável pelas configurações da comunicação na rede CANOpen. Estes objetos são comuns a todos os dispositivos, porém somente alguns são obrigatórios. A seguir é apresentada uma lista com os objetos desta faixa suportados pelo CFW-08.

Índice	Objeto	Nome	Tipo	Acesso
1000h	VAR	Device Type	UNSIGNED32	ro
1001h	VAR	Error Register	UNSIGNED8	ro
100Ch	VAR	Guard Time	UNSIGNED16	rw
100Dh	VAR	Life Time Factor	UNSIGNED8	rw
1018h	RECORD	Identity Object	Identity	ro
Server SDO Parameter				

1200h	RECORD	1st Server SDO Parameter	SDO Parameter	ro
Receive PDO Communication Parameter				
1400h	RECORD	1st receive PDO Parameter	PDO CommPar	rw
Receive PDO Mapping Parameter				
1600h	RECORD	1st receive PDO Mapping	PDO Mapping	rw
Transmit PDO Communication Parameter				
1800h	RECORD	1st transmit PDO Parameter	PDO CommPar	rw
Transmit PDO Mapping Parameter				
1A00h	RECORD	1st transmit PDO Mapping	PDO Mapping	rw

Tabela 14 - Lista de objetos do CFW-08 - Communication Profile

Demais objetos não mostrados nesta lista não são utilizados pelo CFW-08, ou então estão em faixas reservadas do dicionário.

6.4. Manufacturer Specific - Objetos específicos do CFW -08

Nos índices de 2000h até 5FFFh, cada fabricante é livre para definir quais objetos estarão presentes, o tipo e a função de cada objeto. No caso do CFW-08, nesta faixa de objetos foi disponibilizada a lista de parâmetros. Através destes parâmetros é possível operar o CFW-08, executando qualquer função que o inversor possa realizar.

Os parâmetros foram disponibilizados a partir do índice 2000h, e com o número do parâmetro somado a este índice para obter sua posição no dicionário. As variáveis básicas foram disponibilizadas a partir do índice 3388h, com o número da variável somado a este índice para obter sua posição no dicionário. A tabela a seguir ilustra como estão distribuídos os parâmetros no dicionário de objetos.

Índice	Objeto	Nome	Tipo	Acesso
2000h	VAR	P000 – Parâmetro de acesso	INTEGER16	rw
2003h	VAR	P003 – Corrente do motor	INTEGER16	ro
2004h	VAR	P004 – Tensão CC	INTEGER16	ro
2005h	VAR	P005 – Frequência de saída	INTEGER16	ro
...
2064h	VAR	P100 – Tempo de aceleração	INTEGER16	rw
2065h	VAR	P101 – Tempo de desaceleração	INTEGER16	rw
...
338Ah	VAR	VB02 – Estado do inversor	UNSIGNED16	ro
338Bh	VAR	VB03 – Comando lógico	UNSIGNED16	rw
338Ch	VAR	VB04 – Referência de velocidade	INTEGER16	rw

Tabela 15 - Lista de objetos do CFW-08 - Manufacturer Specific

O arquivo eds que acompanha o produto traz a lista de todos os parâmetros acessíveis via interface CANopen. Para a lista completa e uma descrição detalhada dos parâmetros, consulte o manual do usuário do CFW-08. É necessário saber operar o inversor através dos parâmetros para poder programar corretamente sua operação via rede CANopen.

7. Descrição dos objetos de comunicação

Nesta seção são descritos detalhadamente cada um dos objetos citados na Tabela 14, além de descrever também o funcionamento dos objetos de comunicação (COBs) referenciados no item 2.7. é necessário conhecer como estes objetos são operados para utilizar as funções disponíveis para a comunicação do CFW-08.

7.1. Objetos de identificação

Existe um conjunto de objetos no dicionário utilizados para identificação do equipamento, porém não possuem influência no seu comportamento na rede CANopen.

7.1.1. Objeto 1000h - Device Type

Este objeto fornece um código em 32 bits que descreve o tipo de objeto e sua funcionalidade.

Índice	1000h
Nome	Device type
Objeto	VAR
Tipo	UNSIGNED32

Acesso	ro
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	0001.0192h

Este código pode ser dividido em duas partes: 16 bits inferiores, descrevendo o tipo de perfil (*profile*) que o dispositivo utiliza, e 16 bits superiores, indicando uma função específica, de acordo com o perfil especificado. Para o CFW-08, estes valores são respectivamente 0192h (segue o especificado pelo documento Device Profile Drives and Motion Control), e 0001h (possui funcionalidades de inversor de frequência).

7.1.2. Objeto 1001h - Error Register

Este objeto indica a ocorrência ou não de erro no dispositivo. O tipo de erro registrado para o CFW-08 segue o descrito pela Tabela 16.

Índice	1001h
Nome	Error register
Objeto	VAR
Tipo	UNSIGNED8

Acesso	ro
Mapeável	Sim
Faixa	UNSIGNED8
Valor Padrão	0

Bit	Significado
0	Erro genérico
1	Corrente

2	Tensão
3	Temperatura
4	Comunicação
5	Reservado (sempre 0)
6	Reservado (sempre 0)
7	Específico do fabricante

Tabela 16 - Estrutura do objeto Error Register

Caso o dispositivo apresente algum erro, o bit equivalente deve ser ativado. O primeiro bit (erro genérico) deverá ser ativado em qualquer situação de erro.

7.1.3. Objeto 1018h - Identity object

Traz informações gerais sobre o dispositivo.

Índice	1018h
Nome	Identity object
Objeto	Record
Tipo	Identity

Sub-índice	0
Descrição	Número do último sub-índice
Acesso	ro
Mapeável	Não
Faixa	UNSIGNED8
Valor Padrão	4

Sub-índice	1
Descrição	Vendor ID
Acesso	ro
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	0000.0123h

Sub-índice	2
Descrição	Código do produto
Acesso	ro
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	De acordo com o modelo do inversor. O arquivo EDS informa o código do produto para cada modelo existente

Sub-índice	3
Descrição	Número da revisão
Acesso	ro
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	De acordo com a versão de firmware do equipamento

Sub-índice	4
Descrição	Número serial
Acesso	ro
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	Diferente para cada CFW-08

O Vendor ID é um número que identifica o fabricante junto à CiA. Neste caso, WEG Indústrias S.A. - Divisão Automação é representada pelo número 0000.00123h. O código do produto é definido pelo fabricante, e varia de acordo com o modelo do inversor. O número da revisão representa a versão de firmware do equipamento. O sub-índice 4 é um número serial único para cada inversor de frequência CFW-08 em rede CANopen.

7.2. Service Data Objects - SDOs

Os SDOs são responsáveis pelo acesso direto ao dicionário de objetos de um determinado dispositivo na rede. Eles são utilizados para a configuração, e portanto possuem baixa prioridade, já que não devem ser utilizados para comunicar dados necessários para a operação do dispositivo.

Existem dois tipos de SDOs: cliente e servidor. Basicamente, a comunicação inicia com o cliente (usualmente o mestre da rede) fazendo uma requisição de leitura (*upload*) ou escrita (*download*) para um servidor, e este responde ao que foi requisitado.

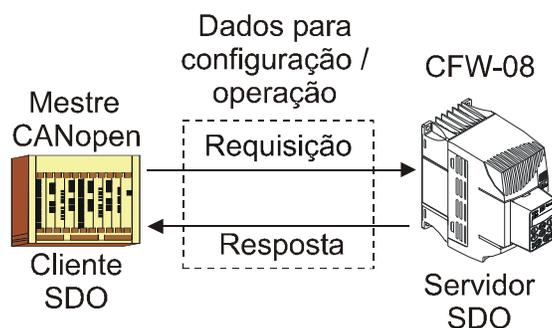


Figura 5 - Comunicação entre Cliente e Servidor SDO

7.2.1. Objeto 1200h - Servidor SDO

O CFW-08 possui um único SDO do tipo servidor, que possibilita o acesso a todo o seu dicionário de objetos. Através dele, um cliente SDO pode configurar a comunicação, parâmetros e modos de operação do inversor. Todo o servidor SDO possui um objeto, do tipo SDO PARAMETER (ver item 6.2.2), para a sua configuração, possuindo a seguinte estrutura:

Índice	1200h
Nome	Server SDO Parameter
Objeto	Record
Tipo	SDO Parameter

Sub-índice	0
Descrição	Número do último sub-índice

Acesso	ro
Mapeável	Não
Faixa	UNSIGNED8
Valor Padrão	2

Sub-índice	1
Descrição	COB-ID Cliente - Servidor (rx)
Acesso	ro
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	600h + Node-ID

Sub-índice	2
Descrição	COB-ID Servidor - Cliente (tx)
Acesso	ro
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	580h + Node-ID

7.2.2. Funcionamento dos SDOs

Um telegrama enviado por um SDO possui 8 bytes de tamanho, com a seguinte estrutura:

Identificador	8 bytes de dados							
	Comando	Índice		Sub-índice	Dados do objeto			
11 bits	byte 0	byte 1	byte 2	byte 3	byte 4	byte 5	byte 6	byte 7

O identificador depende do sentido da transmissão (rx ou tx) e do endereço (ou Node-ID) do servidor destino. Por exemplo, um cliente que faz uma requisição para um servidor cujo Node-ID é 1, deve enviar uma mensagem com o identificador igual a 601h. O servidor irá receber esta mensagem e responder com um telegrama cujo COB-ID é igual a 581h.

O código do comando depende do tipo de função utilizada. Para as transmissões de um cliente para um servidor, podem ser utilizados os seguintes comandos:

Comando	Função	Descrição	Dados do objeto
22h	Download	Escrita em objeto	Indefinido
23h	Download	Escrita em objeto	4 bytes
2Bh	Download	Escrita em objeto	2 bytes
2Fh	Download	Escrita em objeto	1 byte
40h	Upload	Leitura de objeto	Não utilizado
60h ou 70h	Upload segment	Leitura segmentada	Não utilizado

Tabela 17 - Código dos comandos para cliente SDO

Ao fazer a requisição, o cliente indicará através de seu COB-ID, qual o endereço do escravo para o qual esta requisição se destina. Somente um escravo (usando seu respectivo servidor SDO) poderá responder para o cliente o telegrama recebido. O

telegrama de resposta possuirá também a mesma estrutura do telegrama de requisição, porém os comandos serão diferentes:

Comando	Função	Descrição	Dados do objeto
60h	Download	Resposta para escrita em objeto	Não utilizado
43h	Upload	Resposta para leitura de objeto	4 bytes
4Bh	Upload	Resposta para leitura de objeto	2 bytes
4Fh	Upload	Resposta para leitura de objeto	1 byte
41h	Upload segment	Inicia resposta segmentada para leitura	4 bytes
01h ... 0Dh	Upload segment	Último segmento de dados para leitura	8 ... 2 bytes

Tabela 18 - Código dos comandos para servidor SDO

Para leituras que envolvem até quatro bytes de dados, uma única mensagem pode ser transmitida pelo servidor; para leitura de uma quantidade maior de bytes, é necessário que cliente e servidor troquem múltiplos telegramas.

Um telegrama somente é completo após a confirmação do servidor para a requisição feita pelo cliente. Caso algum erro seja detectado durante a troca de telegramas (por exemplo, não há resposta do servidor), o cliente poderá abortar o processo com uma mensagem de aviso com o código do comando igual a 80h.



NOTA!

Quando o SDO é utilizado para escrita nos objetos que representam os parâmetros do CFW-08 (objetos a partir do índice 2000h), este valor é salvo na memória não volátil do inversor de frequência. Desta forma, depois de desligado ou feito o reset do equipamento, os valores configurados não são perdidos. Para os demais objetos, estes valores não são salvos automaticamente, de maneira que é necessário reescrever os valores desejados.

Exemplo: um cliente SDO solicita para um CFW-08 no endereço 1, a leitura do objeto identificado pelo índice 2000h, sub-índice 0 (zero), que representa um inteiro de 16 bits. O telegrama do mestre possui a seguinte forma:

<i>Identificador</i>	<i>Comando</i>	<i>Índice</i>		<i>Sub-índice</i>	<i>Dados</i>			
601h	40h	00h	20h	00h	00h	00h	00h	00h

O CFW-08 responde à requisição, indicando que o valor para o referido objeto é igual a 999³:

<i>Identificador</i>	<i>Comando</i>	<i>Índice</i>		<i>Sub-índice</i>	<i>Dados</i>			
581h	4Bh	00h	20h	00h	E7	03h	00h	00h

7.3. Process Data Objects - PDOs

Os PDOs são utilizados para enviar e receber dados utilizados durante a operação do dispositivo, que muitas vezes precisam ser transmitidos de forma rápida e eficiente. Por isso, eles possuem uma prioridade maior do que os SDOs.

³ Não esquecer que qualquer dado do tipo inteiro, a ordem de transferência dos bytes vai do menos significativo até o mais significativo

Nos PDOs, apenas os dados são transmitidos no telegrama (índices e sub-índices são omitidos), e desta forma é possível fazer uma transmissão mais eficiente, com maior volume de dados em um único telegrama. Porém é necessário configurar previamente o que está sendo transmitido pelo PDO, de forma que, mesmo sem a indicação do índice e sub-índice, seja possível saber o conteúdo do telegrama.

Existem dois tipos de PDOs, os PDOs de recepção e os PDOs de transmissão. Os PDOs de transmissão são responsáveis por enviar dados para a rede, enquanto que os PDOs de recepção ficam responsáveis por receber e tratar estes dados. Desta forma é possível que haja comunicação entre escravos da rede CANopen, basta configurar um escravo para transmitir uma informação, e um ou mais escravos para receber esta informação.

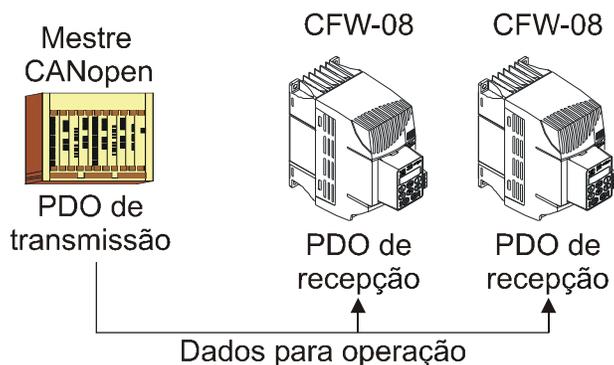


Figura 6 - Comunicação utilizando PDOs



NOTA!

PDOs somente podem ser transmitidos ou recebidos quando o dispositivo está no estado operacional. A Figura 7 ilustra os estados disponíveis para um nó da rede CANopen.

7.3.1. Objetos mapeáveis para os PDOs

Para um objeto poder ser transmitido através de um PDO, é necessário que ele seja mapeável para o conteúdo do PDO. Na descrição dos objetos de comunicação (1000h - 1FFFh), o campo *Mapeável* informa se ele é ou não mapeável. Usualmente, apenas informações necessárias para a operação do dispositivo são mapeáveis, como comandos para habilitação, status do dispositivo, referências, etc. Informações para configuração do dispositivo não são acessíveis através de PDOs, e caso seja necessário acessá-las via rede deve-se utilizar os SDOs.

Para os objetos específicos do CFW-08 (2000h - 5FFFh), a tabela a seguir apresenta alguns objetos mapeáveis para os PDOs. Parâmetros com acesso apenas para leitura (ro) podem ser utilizados apenas por PDOs de transmissão, enquanto que os demais parâmetros podem ser utilizados apenas por PDOs de recepção. O arquivo EDS do CFW-08 traz a lista de todos os objetos disponíveis para o inversor, informando se o objeto é mapeável ou não.

Índice	Objeto	Nome	Tipo	Acesso
2003h	VAR	P003 – Corrente do motor	INTEGER16	ro
2004h	VAR	P004 – Tensão do circuito intermediário	INTEGER16	ro

2005h	VAR	P005 – Frequência de saída	INTEGER16	ro
2007h	VAR	P007 – Tensão de saída	INTEGER16	ro
2008h	VAR	P008 – Temperatura do motor	INTEGER16	ro
2009h	VAR	P009 – Torque do motor	INTEGER16	ro
200Eh	VAR	P014 – Último erro ocorrido	INTEGER16	ro
2028h	VAR	P040 – Variável de processo	UNSIGNED16	ro
2064h	VAR	P100 – Tempo de aceleração	INTEGER16	rw
2065h	VAR	P101 – Tempo de desaceleração	INTEGER16	rw
2079h	VAR	P121 – Referência de velocidade via teclas	INTEGER16	rw
220Dh	VAR	P525 – Setpoint do regulador PID via teclas	INTEGER16	rw
338Ah	VAR	VB02 – Estado do inversor	UNSIGNED16	ro
338Bh	VAR	VB03 – Comando do inversor	UNSIGNED16	rw
338Ch	VAR	VB04 – Referência de velocidade CANopen	INTEGER16	rw

Tabela 19 - Lista de parâmetros mapeáveis para PDOs

7.3.2. PDOs de recepção

Os PDOs de recepção, ou RPDOs, são responsáveis por receber dados que outros dispositivos enviam para a rede CANopen. O CFW-08 possui apenas um PDO de recepção, podendo receber até 8 bytes de dados. O RPDO possui dois parâmetros para sua configuração, um PDO_COMM_PARAMETER e um PDO_MAPPING, conforme descrito a seguir.

PDO_COMM_PARAMETER

Índice	1400h
Nome	Receive PDO communication parameter
Objeto	Record
Tipo	PDO COMM PARAMETER

Sub-índice	0
Descrição	Número do último sub-índice
Acesso	ro
Mapeável	Não
Faixa	UNSIGNED8
Valor Padrão	2

Sub-índice	1
Descrição	COB-ID usado pelo PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	200h + Node-ID

Sub-índice	2
Descrição	Tipo de transmissão
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED8
Valor Padrão	254

O sub-índice 1 contém o COB-ID do PDO de recepção. Sempre que uma mensagem for enviada para a rede, este objeto irá ler qual o COB-ID desta mensagem, e caso ele seja igual ao valor deste campo, a mensagem será recebida pelo dispositivo. Este campo é formado por um UNSIGNED32 com a seguinte estrutura:

Bit	Valor	Descrição
31 (MSB)	0	PDO está habilitado
	1	PDO está desabilitado
30	0	RTR permitido
29	0	Tamanho do identificador = 11 bits
28 – 11	0	Não utilizado pelo CFW-08, sempre 0
10 – 0 (LSB)	X	COB-ID de 11 bits

Tabela 20 - Descrição do COB-ID

O bit 31 permite habilitar ou desabilitar o PDO. Os bits 30 e 29, que devem ser mantidos em 0 (zero), indicam respectivamente que o PDO aceita frames remotos (RTR frames) e que utiliza identificador de 11 bits. Como o inversor de frequência CFW-08 não utiliza identificadores de 29 bits, os bits de 28 - 11 devem ser mantidos em 0 (zero), enquanto que os bits de 10 até 0 (zero) são usados para configurar o COB-ID para o PDO.

O sub-índice 2 indica o tipo de transmissão deste objeto, de acordo com a tabela a seguir.

Tipo de transmissão	Transmissão de PDOs				
	<i>Cíclico</i>	<i>Acíclico</i>	<i>Síncrono</i>	<i>Assíncrono</i>	<i>RTR</i>
0		•	•		
1 – 240	•		•		
241 – 251	Reservado				
252			•		•
253				•	•
254				•	
255				•	

Tabela 21 - Descrição do tipo de transmissão

- **Valores 0 – 240:** relacionado com o telegrama SYNC. Não utilizado pelo CFW-08, uma vez que este não suporta telegramas do tipo SYNC.
- **Valores 252 e 253:** não permitido para PDOs de recepção.
- **Valores 254 e 255:** indica que não possui relação com o objeto de sincronização. Ao receber uma mensagem, seus valores serão atualizados imediatamente.

PDO_MAPPING

Índice	1600h
Nome	Receive PDO mapping
Objeto	Record
Tipo	PDO MAPPING

Sub-índice	0
Descrição	Número de objetos mapeados
Acesso	ro
Mapeável	Não
Faixa	0 = desabilitado 1 ... 4 = número de objetos mapeados
Valor Padrão	2

Sub-índice	1
Descrição	1º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	338B.0010h

Sub-índice	2
Descrição	2º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	338C.0010h

Sub-índice	3
Descrição	3º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	0

Sub-índice	4
Descrição	4º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	0

Este parâmetro indica os objetos mapeados no PDO de recepção do CFW-08. Para este RPDO, é possível mapear até quatro objetos diferentes, desde que o tamanho total não ultrapasse oito bytes. O mapeamento de um objeto é feito indicando o seu índice, sub-índice⁴ e tamanho (em bits) em um campo UNSIGNED32, com o seguinte formato:

UNSIGNED32		
Índice (16 bits)	Sub-índice (8 bits)	Tamanho do objeto (8 bits)

Por exemplo, analisando o mapeamento padrão do PDO de recepção, temos:

- **Sub-índice 0 = 2:** o RPDO possui dois objetos mapeados.

⁴ Caso o objeto seja do tipo VAR e não possua sub-índice, deve ser indicado o valor 0 (zero) para o sub-índice

- **Sub-índice 1 = 338B.0010h:** o primeiro objeto mapeado possui índice igual a 338Bh, sub-índice igual a 0 (zero) e tamanho de 16 bits. Este objeto corresponde à palavra de controle do inversor (ver item 4.3.2).
- **Sub-índice 2 = 338C.0010h:** o segundo objeto mapeado possui índice igual a 338Ch, sub-índice igual a 0 (zero) e tamanho de 16 bits. Este objeto corresponde à referência de velocidade do inversor (ver item 4.3.3).

Desta forma, sempre que este PDO receber um telegrama, ele vai saber que o telegrama deverá conter quatro bytes de dados, com o conteúdo para a palavra de controle e referência de velocidade para o CFW-08. É possível modificar este mapeamento, alterando a quantidade ou o número dos objetos mapeados. Lembrar que no máximo podem ser mapeados 4 objetos ou 8 bytes.



NOTA!

- Para poder alterar os objetos mapeados em um PDO, primeiro é necessário escrever o valor 0 (zero) no sub-índice 0 (zero). Desta forma, os valores dos sub-índices 1 até 4 podem ser alterados. Depois de feito o mapeamento desejado, deve-se escrever novamente no sub-índice 0 (zero) o número de objetos que foram mapeados, habilitando novamente o PDO.
- Para agilizar a atualização dos dados via PDO, os valores recebidos através destes objetos não são salvos na memória não volátil do inversor. Desta forma, após um desligamento ou reset do equipamento, os objetos modificados por um RPDO voltam para o seu valor padrão.
- Não esquecer que os PDOs somente podem ser recebidos caso o CFW-08 esteja no estado operacional.

7.3.3. PDOs de transmissão

Os PDOs de transmissão, ou TPDOs, como o nome diz, são responsáveis por transmitir dados para a rede CANopen. O CFW-08 possui apenas um PDO de transmissão, podendo transmitir até 8 bytes de dados. De forma semelhante aos RPDOs, cada TPDO possui dois parâmetros para sua configuração, um PDO_COMM_PARAMETER e um PDO_MAPPING, conforme descrito a seguir.

PDO_COMM_PARAMETER

Índice	1800h
Nome	Transmit PDO Parameter
Objeto	Record
Tipo	PDO COMM PARAMETER

Sub-índice	0
Descrição	Número do último sub-índice
Acesso	ro
Mapeável	Não
Faixa	UNSIGNED8
Valor Padrão	5

Sub-índice	1
Descrição	COB-ID usado pelo PDO
Acesso	rw

Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	180h + Node-ID

Sub-índice	2
Descrição	Tipo de transmissão
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED8
Valor Padrão	254

Sub-índice	3
Descrição	Tempo entre transmissões
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED16
Valor Padrão	-

Sub-índice	4
Descrição	Reservado
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED8
Valor Padrão	-

Sub-índice	5
Descrição	Temporizador de eventos
Acesso	rw
Mapeável	Não
Faixa	0 = desabilitado UNSIGNED16
Valor Padrão	0

O sub-índice 1 contém o COB-ID do PDO de transmissão. Sempre que este PDO enviar uma mensagem para a rede, o identificador desta mensagem será este COB-ID. A estrutura deste campo é descrita na Tabela 20.

O sub-índice 2 indica o tipo de transmissão deste objeto, que segue o descrito pela Tabela 21. Porém seu funcionamento é diferente para PDOs de transmissão:

- **Valores 0 – 240, 252:** relacionado com o telegrama SYNC. Não utilizado pelo CFW-08, uma vez que este não suporta telegramas do tipo SYNC.
- **Valor 253:** o PDO deve atualizar e enviar uma mensagem assim que receber um frame remoto.
- **Valores 254 e 255:** o objeto deve ser transmitido de acordo com o timer programado no sub-índice 5.

No sub-índice 3 é possível programar um tempo mínimo (em múltiplos de 100us) que deve transcorrer para que, depois de transmitido um telegrama, um novo telegrama possa ser enviado por este PDO. O valor 0 (zero) desabilita esta função.

O sub-índice 5 contém um valor para habilitar um temporizador para a envio automático de um PDO. Desta forma, sempre que um PDO for configurado para o tipo assíncrono, é possível programar o valor deste temporizador (em múltiplos de 1ms), para que o PDO seja transmitido periodicamente no tempo programado.



NOTA!

- Deve-se observar o tempo programado neste temporizador, de acordo com a taxa de transmissão utilizada. Tempos muito pequenos (próximos ao tempo de transmissão do telegrama) podem monopolizar o barramento, causando a retransmissão indefinida do PDO e impedindo que outros objetos menos prioritários possam transmitir seus dados.
- O tempo mínimo permitido para esta função no CFW-08 é 2ms.
- Não esquecer que os PDOs somente podem ser transmitidos caso o escravo esteja no estado operacional.

PDO_MAPPING

Índice	1A00h
Nome	Transmit PDO mapping
Objeto	Record
Tipo	PDO MAPPING

Sub-índice	0
Descrição	Número do último sub-índice
Acesso	ro
Mapeável	Não
Faixa	0 = desabilitado 1 ... 4 = número de objetos mapeados
Valor Padrão	2

Sub-índice	1
Descrição	1º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	338A.0010h

Sub-índice	2
Descrição	2º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	2005.0010h

Sub-índice	3
Descrição	3º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	0

Sub-índice	4
Descrição	4º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	0

O PDO MAPPING para a transmissão funciona de forma semelhante que para a recepção, porém neste caso são definidos os dados a serem transmitidos pelo PDO. Cada objeto mapeado deve ser colocado na lista de acordo com o descrito a seguir:

UNSIGNED32		
Índice (16 bits)	Sub-índice (8 bits)	Tamanho do objeto (8 bits)

Por exemplo, analisando o mapeamento padrão do PDO de transmissão, temos:

- **Sub-índice 0 = 2:** este PDO possui dois objetos mapeados.
- **Sub-índice 1 = 338A.0010h:** o primeiro objeto mapeado possui índice igual a 338Ah, sub-índice 0 (zero), e tamanho igual a 16 bits. Este objeto corresponde à palavra de estado do CFW-08 (ver item 4.3.1).
- **Sub-índice 2 = 2005.0010h:** o segundo objeto mapeado possui índice igual a 2005h, sub-índice 0 (zero), e tamanho igual a 16 bits. Este objeto corresponde ao parâmetro P005 do inversor.

Desta forma, sempre que este PDO for transmitir seus dados, ele vai elaborar o seu telegrama contendo quatro bytes de dados, com os valores do estado do inversor e do parâmetro P002 (valor proporcional à frequência). É possível modificar este mapeamento, alterando a quantidade ou o número dos parâmetros mapeados. Lembrar que no máximo podem ser mapeados 4 objetos ou 8 bytes.



NOTA!

Para poder alterar os objetos mapeados em um PDO, primeiro é necessário escrever o valor 0 (zero) no sub-índice 0 (zero). Desta forma, os valores dos sub-índices 1 até 4 podem ser alterados. Depois de feito o mapeamento desejado, deve-se escrever novamente no sub-índice 0 (zero) o número de objetos que foram mapeados, habilitando novamente o PDO.

7.4. Network Management - NMT

O objeto de gerenciamento da rede é responsável por um conjunto de serviços que controlam a comunicação do dispositivo na rede CANopen. Para o CFW-08 estão disponíveis os serviços de controle do nó e de controle de erros (utilizando *Node Guarding*).

7.4.1. Controle dos estados do escravo

Com relação à comunicação, um dispositivo da rede CANopen pode ser descrito pela seguinte máquina de estados:

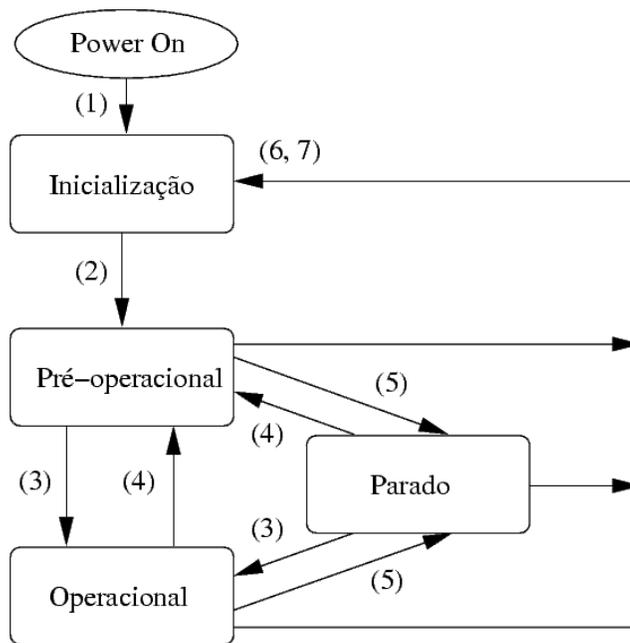


Figura 7 - Diagrama de estados do nó CANopen

Transição	Descrição
1	Dispositivo é ligado e começa a inicialização (automático)
2	Inicialização concluída, vai para o estado pré-operacional (automático)
3	Recebe comando Start Node para entrar no estado operacional
4	Recebe comando Enter Pre-Operational, e vai para o estado pré-operacional
5	Recebe comando Stop Node para entrar no estado parado
6	Recebe comando Reset Node, onde executa o reset completo do dispositivo
7	Recebo comando Reset Communication, onde reinicializa o valor dos objetos e a comunicação CANopen do dispositivo

Tabela 22 - Descrição das transições

Durante a inicialização, é definido o Node-ID, criados os objetos e configurada a interface com a rede CAN. Não é possível comunicar-se com o dispositivo nesta etapa, que é concluída automaticamente. No final desta etapa, o escravo envia para rede um telegrama do objeto Boot-up, utilizado apenas para indicar que a inicialização foi concluída e que o escravo entrou no estado pré-operacional. Este telegrama possui identificador 700h + Node-ID, e apenas um byte de dados com valor igual a 0 (zero).

No estado pré-operacional, já é possível comunicar-se com o escravo, porém os PDOs ainda não estão disponíveis para operação. No estado operacional, todos os objetos estão disponíveis, enquanto que no estado parado, apenas o objeto NMT pode receber ou transmitir telegramas para a rede. A tabela a seguir mostra os objetos disponíveis para cada estado.

	Inicialização	Pré-operacional	Operacional	Parado
PDO			•	
SDO		•	•	
SYNC		•	•	
EMCY		•	•	
Boot-up	•			
NMT		•	•	•

Tabela 23 - Objetos acessíveis em cada estado

O LED de indicação RUN mostra em qual estado o dispositivo encontra-se no momento, de acordo com o descrito no item 5.3.

Esta máquina de estados é controlada pelo mestre da rede, que envia para cada escravo, comandos para que seja executada a transição de estados desejada. Estes telegramas não possuem confirmação, o que significa que o escravo apenas recebe o telegrama sem retornar resposta para o mestre. Os telegramas recebidos possuem a seguinte estrutura:

<i>Identificador</i>	<i>byte 1</i>	<i>byte 2</i>
00h	Código do comando	Node-ID destino

Código do comando	Node-ID destino
1 = START node (transição 3)	0 = Todos os escravos
2 = STOP node (transição 4)	1 ... 127 = Escravo específico
128 = Enter pre-operational (transição 5)	
129 = Reset node (transição 6)	
130 = Reset communication (transição 7)	

Tabela 24 - Comandos para a transição de estados

As transições indicadas no código do comando equivalem às transições de estado executadas pelo nó após receber o comando (conforme Figura 7). O comando *Reset node* faz com que o CFW-08 execute um reset completo do dispositivo, enquanto que o comando *Reset communication* faz com que o dispositivo reinicialize apenas os objetos relativos à comunicação CANopen.

7.4.2. Controle de Erros – Node Guarding

Existem dois serviços para controle de erros do dispositivo: *heartbeat* e *guarding*. Destes, apenas o serviço de *node guarding* está implementado no CFW-08.

Este serviço é utilizado para possibilitar a monitoração da comunicação com a rede CANopen, tanto pelo mestre quanto pelo escravo. Neste tipo de serviço, o mestre envia telegramas periódicos para o escravo, que responde o telegrama recebido. Caso ocorra algum erro que interrompa a comunicação, será possível identificar este erro, pois tanto o mestre quanto o escravo serão notificados pelo *timeout* na execução deste serviço. Os eventos de erro são chamados de *Node Guarding* para o mestre, e de *Life Guarding* para o escravo.

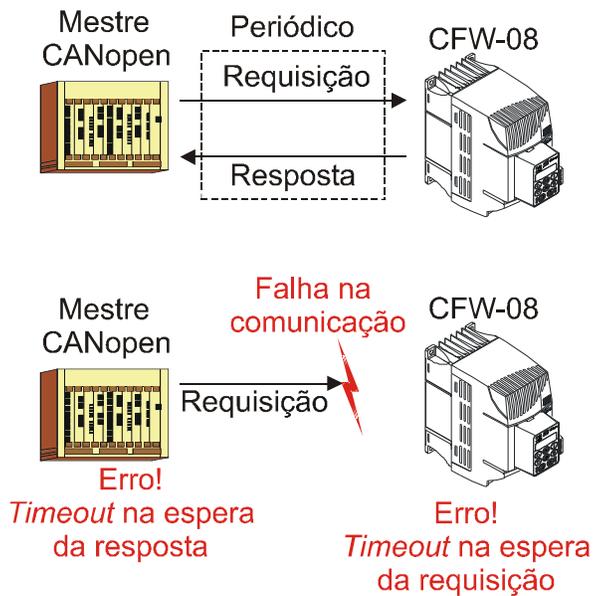


Figura 8 - Serviço de controle de erros – Node Guarding

Para o serviço de *Node Guarding*, existem dois objetos do dicionário para configuração dos tempos para detecção de erros de comunicação:

Índice	100Ch
Nome	Guard Time
Objeto	VAR
Tipo	UNSIGNED16

Acesso	rw
Mapeável	Não
Faixa	UNSIGNED16
Valor Padrão	0

Índice	100Dh
Nome	Life Time Factor
Objeto	VAR
Tipo	UNSIGNED8

Acesso	rw
Mapeável	Não
Faixa	UNSIGNED8
Valor Padrão	0

O objeto 100Ch permite programar o tempo necessário (em milissegundos) para que uma ocorrência de falha seja detectada, caso o CFW-08 não receba nenhum telegrama do mestre. O objeto 100Dh indica quantas falhas em seqüência são necessárias até que se considere que houve realmente erro de comunicação. Portanto, a multiplicação destes dois valores fornecerá o tempo total necessário para detecção de erros de comunicação utilizando este objeto. O valor 0 (zero) desabilita esta função.

Uma vez configurado, o CFW-08 começa a contar estes tempos a partir do primeiro telegrama *Node Guarding* recebido do mestre da rede. O telegrama do mestre é do tipo

remoto, não possuindo bytes de dados. O identificador é igual a 700h + Node-ID do escravo destino. Já o telegrama de resposta do escravo possui 1 byte de dados com a seguinte estrutura:

Identificador	byte 1	
	bit 7	bit 6 ... bit 0
700h + Node-ID	Toggle	Estado do escravo

Este telegrama possui um único byte dados. Este byte contém, nos sete bits menos significativos, um valor para indicar o estado do escravo (4 = Parado, 5 = Operacional e 127 = Pré-operacional), e no oitavo bit, um valor que deve ser alterado a cada telegrama de enviado pelo escravo (*toggle bit*).

Caso o CFW-08 detecte um erro utilizando este mecanismo, ele automaticamente irá para o estado pré-operacional e indicará E35 na sua IHM. A ocorrência deste erro também pode ser observada através do LED de erro do CFW-08. É possível também programar o inversor para tomar uma ação quando este erro ocorrer, através do parâmetro P313. Consulte o item 4 para a descrição detalhada dos parâmetros.



NOTA!

Com relação à utilização deste serviço, deve-se observar os seguintes pontos:

- Este objeto está ativo mesmo no estado parado (ver Tabela 23).
- O valor 0 (zero) em um dos dois objetos desabilita esta função.
- Depois de detectado o erro, caso o serviço seja habilitado mais uma vez, a indicação do erro é retirada da IHM e dos LEDs de indicação.
- O valor mínimo aceito para o CFW-08 é de 2 ms, mas levando-se em conta a taxa de transmissão e o número de pontos na rede, os tempos programados para essa função devem ser coerentes, de maneira que haja tempo suficiente para transmissão dos telegramas e também para que o resto da comunicação possa ser processada.

7.4.3. Procedimento de inicialização

Uma vez conhecido o funcionamento dos objetos disponíveis para o inversor de frequência CFW-08, é necessário agora programar os diferentes objetos para operarem em conjunto na rede. De forma geral, o procedimento para inicialização dos objetos em uma rede CANopen segue o descrito pelo fluxograma a seguir:

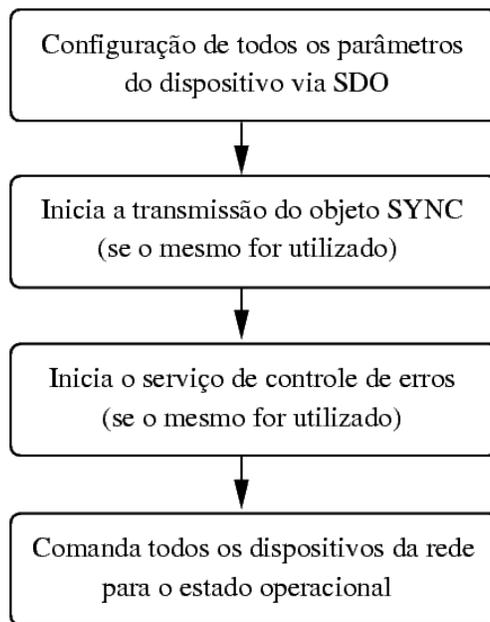


Figura 9 - Fluxograma do processo de inicialização

É necessário observar que os objetos de comunicação do CFW-08 (1000h até 1FFFh) não são armazenados na memória não volátil. Desta forma, sempre que for feito o reset ou desligado o equipamento, é necessário refazer a parametrização dos objetos de comunicação. Para os objetos específicos do fabricante (a partir de 2000h, que representam os parâmetros), estes são armazenados na memória não volátil, e portanto podem ser parametrizados uma única vez.

8. Erros relacionados à comunicação CANopen

A seguir são apresentados erros do CFW-08 relativos a problemas com a comunicação CANopen.

Erro	Descrição da atuação	Prováveis causas e soluções
E41 Erro de auto-diagnose	Após energizado, o inversor apresenta E41 na IHM e não responde a ação alguma do operador.	Além das causas descritas no manual do usuário do CFW-08, a indicação de E41 pode ocorrer também caso o módulo de comunicação CANopen seja conectado a um inversor que não possui o cartão de controle adequado a esta comunicação. Deve-se consultar o manual do usuário do inversor CFW-08 para identificar a versão do cartão de controle adequada para operar o inversor juntamente com o módulo CANopen.
E33 Sem alimentação	Quando o drive detectar falha na alimentação da interface CANopen, indicando na IHM do produto E33.	Conferir se a fonte de alimentação externa de 24V está devidamente conectada entre os pinos 1 e 5 da interface CANopen.
E34 Bus off	Após a inicialização ou durante a comunicação, caso o número de erros de comunicação seja muito elevado, o dispositivo indica E34 na IHM e o LED ERR indica vermelho sólido.	A ocorrência do erro de bus off deve-se principalmente por problemas na instalação ou parametrização dos equipamentos conectados na rede. Problemas na configuração da rede: <ul style="list-style-type: none"> • Verifique se todos os equipamentos conectados na rede CANopen estão programados para operar na mesma taxa de comunicação. • Verifique se não existem dois equipamentos programados com o mesmo endereço na rede. Problemas na instalação: <ul style="list-style-type: none"> • Verifique se a passagem dos cabos está feita corretamente, com os drives e a blindagem do cabo devidamente aterrada. • Verifique se não há problemas de mau contato, oxidação ou curto-circuito entre os sinais da rede. Verifique também se os cabos para a transmissão do sinal não estão rompidos ou invertidos. • Veja se os resistores de terminação estão ativos somente os dois extremos do segmento. • Veja se a taxa de comunicação utilizada não está muito alta, e se o comprimento do cabo não está acima do permitido. Em geral, a rede fica mais suscetível a problemas de comunicação quanto maior a taxa de comunicação utilizada.
E35 Controle de erros - <i>Node Guarding</i>	Depois de iniciada a comunicação, o escravo detecta falha no serviço de controle de erros. É indicado E35 na IHM e o LED de erro faz a indicação de duas piscadas.	Esta função é dependente das configurações feitas pelo mestre da rede. Eventuais erros podem ser decorrentes de problemas na comunicação, ou de programação inadequada do mestre ou dos objetos de comunicação através dos quais esta função é habilitada. Para a descrição detalhada desta função deve-se consultar o item 7.4.2.