



# Manual da Comunicação CANopen

Cartão PLC11

Série: PLC11

Idioma: Português

Documento: 10002134020 / 00



# Sumário

<b>SOBRE O MANUAL</b> .....	<b>5</b>
ABREVIÇÕES E DEFINIÇÕES .....	5
REPRESENTAÇÃO NUMÉRICA .....	5
DOCUMENTOS.....	5
<b>1 INTRODUÇÃO À COMUNICAÇÃO CANOPEN</b> .....	<b>6</b>
1.1 CAN.....	6
1.1.1 <i>Frame de dados</i> .....	6
1.1.2 <i>Frame remoto</i> .....	6
1.1.3 <i>Acesso à rede</i> .....	6
1.1.4 <i>Controle de erros</i> .....	6
1.1.5 <i>CAN e CANopen</i> .....	6
1.2 CARACTERÍSTICAS DA REDE CANOPEN .....	7
1.3 MEIO FÍSICO .....	7
1.4 ENDEREÇO NA REDE CANOPEN .....	7
1.5 ACESSO AOS DADOS.....	7
1.6 TRANSMISSÃO DE DADOS .....	7
1.7 OBJETOS RESPONSÁVEIS PELA COMUNICAÇÃO – COBs.....	8
1.8 COB-ID.....	8
1.9 ARQUIVO EDS.....	9
<b>2 KITS OPCIONAIS</b> .....	<b>10</b>
2.1 INTERFACE CAN.....	10
2.1.1 <i>Pinagem do conector</i> .....	10
2.1.2 <i>Fonte de alimentação</i> .....	10
2.1.3 <i>Resistor de terminação</i> .....	10
2.1.4 <i>Taxa de comunicação</i> .....	10
2.1.5 <i>Ligação do equipamento na rede CAN</i> .....	11
<b>3 PARAMETRIZAÇÃO DO CARTÃO PLC11</b> .....	<b>12</b>
3.1 SÍMBOLOS PARA DESCRIÇÃO DAS PROPRIEDADES .....	12
P1220 – ESTADO DO CONTROLADOR CAN .....	12
P1221 – CONTADOR DE TELEGRAMAS CAN RECEBIDOS .....	12
P1222 – CONTADOR DE TELEGRAMAS CAN TRANSMITIDOS .....	13
P1223 – CONTADOR DE ERROS DE BUS OFF .....	13
P1224 – CONTADOR DE MENSAGENS CAN PERDIDAS .....	13
P1225 – ESTADO DA CONFIGURAÇÃO CANOPEN.....	13
P1226 – ESTADO DA COMUNICAÇÃO CANOPEN .....	14
P1227 – ESTADO DO NÓ CANOPEN.....	14
P1285 – PROTOCOLO CAN .....	14
P1286 – ENDEREÇO CAN.....	15
P1287 – TAXA DE COMUNICAÇÃO CAN .....	15
P1288 – RESET DE BUS OFF .....	15
P1289 – AÇÃO PARA ERRO DE COMUNICAÇÃO .....	16
<b>4 DICIONÁRIO DE OBJETOS</b> .....	<b>17</b>
4.1 ESTRUTURA DO DICIONÁRIO .....	17
4.2 TIPOS DE DADOS.....	17
4.2.1 <i>Tipos básicos</i> .....	17
4.2.2 <i>Tipos compostos</i> .....	17
4.2.3 <i>Tipos estendidos</i> .....	18
4.3 COMMUNICATION PROFILE – OBJETOS PARA COMUNICAÇÃO .....	18
4.4 MANUFACTURER SPECIFIC – OBJETOS ESPECÍFICOS DO CARTÃO PLC11 .....	19
<b>5 DESCRIÇÃO DOS OBJETOS DE COMUNICAÇÃO</b> .....	<b>21</b>
5.1 OBJETOS DE IDENTIFICAÇÃO .....	21
5.1.1 <i>Objeto 1000h – Device Type</i> .....	21

5.1.2	Objeto 1001h – Error Register.....	21
5.1.3	Objeto 1018h – Identity Object.....	21
5.2	SERVICE DATA OBJECTS – SDOs.....	22
5.2.1	Objeto 1200h – Servidor SDO.....	23
5.2.2	Funcionamento dos SDOs.....	23
5.3	PROCESS DATA OBJECTS – PDOs.....	24
5.3.1	Objetos mapeáveis para os PDOs.....	25
5.3.2	PDOs de recepção.....	25
5.3.3	PDOs de transmissão.....	28
5.4	EMERGENCY OBJECT – EMCY.....	31
5.5	SYNCHRONIZATION OBJECT – SYNC.....	32
5.6	NETWORK MANAGEMENT – NMT.....	32
5.6.1	Controle dos estados do escravo.....	32
5.6.2	Controle de Erros – Node Guarding.....	34
5.6.3	Controle de Erros – Heartbeat.....	35
5.7	PROCEDIMENTO DE INICIALIZAÇÃO.....	37
<b>6</b>	<b>FALHAS E ALARMES RELACIONADOS COM A COMUNICAÇÃO CANOPEN.....</b>	<b>39</b>
	A833/F833 – SEM ALIMENTAÇÃO NA INTERFACE CAN.....	39
	A834/F834 – BUS OFF.....	39
	A835/F835 – ERRO DE NODE GUARDING/HEARTBEAT.....	39

## Sobre o manual

Este manual fornece a descrição necessária para a operação do cartão PLC11 – incorporado ao inversor de frequência CFW-11 – utilizando o protocolo CANopen operando no modo escravo<sup>1</sup>. Este manual deve ser utilizado em conjunto com manual do usuário da PLC11, software WLP e software WSCAN.

### 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
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 PLC11 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

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

---

<sup>1</sup> Para informações de como operar o cartão PLC11 como mestre da rede CANopen, consulte a documentação do software WLP e software WSCAN.

# 1 Introdução à Comunicação CANopen

Para a operação do cartão PLC11 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 cartão. Para uma descrição detalhada do protocolo, consulte a documentação CANopen indicada no item anterior.

## 1.1 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)<sup>2</sup>. Nesta camada, são definidos os diferentes tipos de telegramas (*frames*), a forma de detecção de erros, validação e arbitragem de mensagens.

### 1.1.1 Frame de dados

Os dados em uma rede CAN são transmitidos através de um frame de dados. Este tipo de frame é composto principalmente por um campo identificador de 11 bits<sup>3</sup> (*arbitration field*), e um campo de dados (*data field*), que pode conter até 8 bytes de dados.

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

### 1.1.2 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.

### 1.1.3 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.

### 1.1.4 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 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.

### 1.1.5 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

---

<sup>2</sup> Na especificação do protocolo CAN, é referenciada a norma ISO 11898 como definição da camada 1 deste modelo (camada física).

<sup>3</sup> A especificação CAN 2.0 define dois tipos de frames de dados: *standard* (11 bits) e *extended* (29 bits). Para o protocolo CANopen do cartão PLC11, somente frames *standard* são aceitos.

## Introdução à Comunicação CANopen

---

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. Estes protocolos, porém, não podem operar em conjunto na mesma rede.

### 1.2 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.

### 1.3 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 cartão PLC11 utiliza ainda um circuito de interface com a rede isolado, com a alimentação para esta interface sendo fornecida por uma fonte externa ao equipamento. O componente responsável pela transmissão e recepção de sinais é denominado transceiver, que obedece ao especificado pela ISO 11898.

### 1.4 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 cartão PLC11, operando no modo escravo, 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.

### 1.5 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 4.

### 1.6 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.

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

## 1.7 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 cartão PLC11 suporta os seguintes objetos de comunicação (COBs):

Tipo de Objeto	Descrição
<i>Service Data Object</i> (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 a PLC11.
<i>Process Data Object</i> (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.
<i>Emergency Object</i> (EMCY)	Este objeto é responsável pelo envio de mensagens para indicar a ocorrência de erros no dispositivo. Quando um erro ocorre em um determinado dispositivo (Produtor EMCY), este pode enviar uma mensagem para a rede. Caso algum dispositivo da rede esteja monitorando esta mensagem (Consumidor EMCY), é possível programar para que uma ação seja tomada (desabilitar demais dispositivos da rede, reset de erros, etc.). A PLC11 possui apenas a funcionalidade de produtor EMCY.
<i>Synchronization Object</i> (SYNC)	Na rede CANopen é possível programar um dispositivo (Produtor SYNC) para enviar, periodicamente, uma mensagem de sincronização para todos os dispositivos da rede. Estes dispositivos (Consumidores SYNC) podem então, por exemplo, enviar um determinado dado que necessita ser disponibilizado periodicamente. A PLC11 possui a função de consumidor SYNC.
<i>Network Management</i> (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. A PLC11 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> e <i>Heartbeat</i> ), onde o escravo envia mensagens periódicas para o mestre para informar que a conexão está ativa.

*Tabela 1.1 – Tipos de Objetos de Comunicação (COBs)*

Toda a comunicação do cartão 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 5.

## 1.8 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 cartão PLC11. 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/ Heartbeat	1110	1793 – 1919 (701h – 77Fh)

*Tabela 1.2 – COB-ID para os diferentes objetos*

## 1.9 Arquivo EDS

Cada dispositivo em uma rede CANopen possui um arquivo de configuração EDS, que contém 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 a PLC11 é fornecido juntamente com o produto, e também pode ser obtido através do site <http://www.weg.net>. É necessário observar a versão de software do cartão para utilizar um arquivo EDS que seja compatível com esta versão.

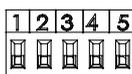
## 2 Kits Opcionais

O cartão PLC11 já possui uma interface de comunicação CAN disponível no produto, sendo possível utilizar o protocolo CANopen sem a necessidade de utilizar um cartão de interface adicional.

### 2.1 Interface CAN

#### 2.1.1 Pinagem do conector

O módulo para comunicação CAN possui um conector *plug-in* de 5 vias (XC36) com a seguinte pinagem:



Pino	Nome	Função
1	V-	Pólo negativo da fonte de alimentação
2	CAN L	Sinal de comunicação CAN_L
3	Shield	Blindagem o cabo
4	CAN H	Sinal de comunicação CAN_H
5	V+	Pólo positivo da fonte de alimentação

*Tabela 2.1 – Pinagem do conector XC36 para interface CAN*

#### 2.1.2 Fonte de alimentação

A interface CAN para a PLC11 necessita de uma tensão de alimentação externa entre os pinos 1 e 5 do conector da rede. Para evitar problemas de diferença de tensão entre os dispositivos da rede, é recomendado que a rede seja alimentada em apenas um ponto, e o sinal de alimentação seja levado a todos os dispositivos através do cabo. Caso seja necessária mais de uma fonte de alimentação, estas devem estar referenciadas ao mesmo ponto. Os dados para consumo individual e tensão de entrada são apresentados na tabela a seguir.

Tensão de alimentação ( $V_{CC}$ )		
Mínimo	Máximo	Recomendado
11	30	24
Corrente (mA)		
Mínimo	Máximo	Médio
20	50	30

*Tabela 2.2 – Características da alimentação para interface CAN*

O cartão PLC11 possui um LED na cor verde ao lado do conector para indicar que a interface está alimentada.

#### 2.1.3 Resistor de terminação

As extremidades do barramento CAN devem possuir um resistor de terminação no valor de  $120\Omega / 0.25W$ , conectando os sinais CAN\_H e CAN\_L caso o drive seja o primeiro ou último elemento do segmento.

#### 2.1.4 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 a PLC11, e o comprimento máximo de cabo que pode ser utilizado na instalação, de acordo com o recomendado pela CiA.

Taxa de comunicação	Comprimento do cabo
1 Mbit/s	40 m
500 Kbit/s	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 2.3 – Taxas de comunicação suportadas e tamanho da instalação*

### 2.1.5 Ligação do equipamento na rede CAN

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 próxima a cabos de potência, pois devido à 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.

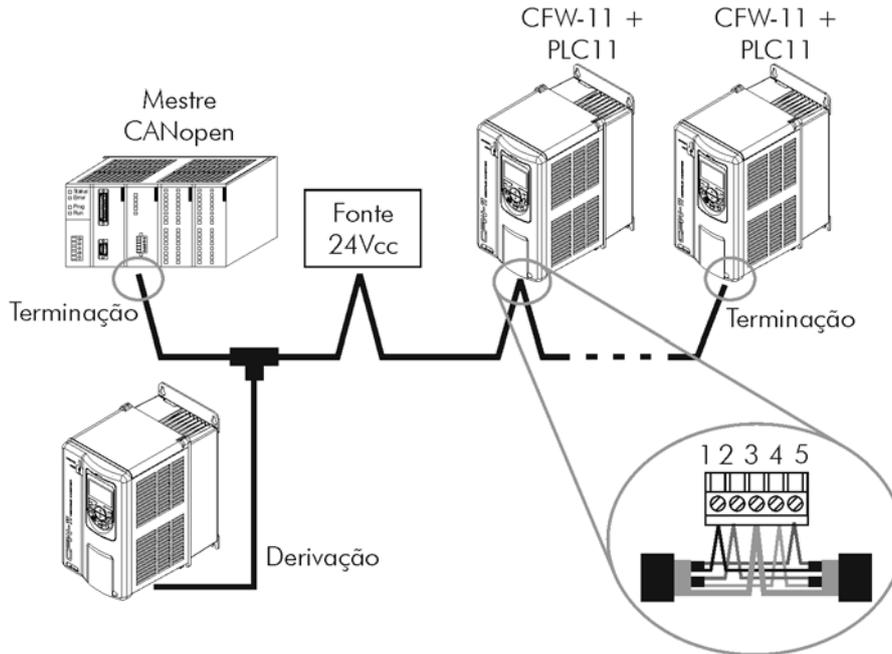


Figura 2.1 – PLC11 em rede CANopen

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 (mm <sup>2</sup> )
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 Parametrização do cartão PLC11

A seguir serão apresentados apenas os parâmetros do cartão PLC11 que possuem relação com a comunicação CANopen.

#### 3.1 Símbolos para Descrição das Propriedades

RO	Parâmetro somente de leitura
CFG	Parâmetro somente alterado com o motor parado

#### P1220 – Estado do Controlador CAN

Faixa de Valores:	0 = Desabilitado 1 = Reservado 2 = Interface CAN ativa 3 = <i>Warning</i> 4 = <i>Error Passive</i> 5 = <i>Bus Off</i> 6 = Sem alimentação	Padrão: -
-------------------	---	-----------

Propriedades: RO

Grupos de acesso via HMI:

01 GRUPOS PARÂMETROS
L 51 PLC
L 130 Parâm. Sistema

Descrição:

Permite identificar se o cartão de interface CAN está devidamente instalado, e se a comunicação apresenta erros.

Opções	Descrição
0 = Inativo	Interface CAN inativa. Ocorre quando a comunicação CAN não está habilitada no parâmetro P1285.
1 = Reservado	
2 = Interface CAN ativa	Interface CAN ativa e sem erros.
3 = <i>Warning</i>	Controlador CAN atingiu o estado de <i>warning</i> <sup>4</sup> .
4 = <i>Error Passive</i>	Controlador CAN atingiu o estado de <i>error passive</i> <sup>4</sup> .
5 = <i>Bus Off</i>	Controlador CAN atingiu o estado de <i>bus off</i> <sup>4</sup> .
6 = Sem alimentação	Interface CAN não possui alimentação entre os pinos 1 e 5 do conector.

Tabela 3.1 – Valores do parâmetro P1220

#### P1221 – Contador de Telegramas CAN Recebidos

Faixa de Valores:	0 a 65535	Padrão: -
-------------------	-----------	-----------

Propriedades: RO

Grupos de acesso via HMI:

01 GRUPOS PARÂMETROS
L 51 PLC
L 130 Parâm. Sistema

Descrição:

Este parâmetro funciona como um contador cíclico, que é incrementado toda vez que um telegrama CAN é recebido. Fornece um retorno para o operador se o dispositivo está conseguindo comunicar-se com a rede. Este contador é zerado sempre que o equipamento for desligado, feito o reset ou ao atingir o limite máximo do parâmetro.

<sup>4</sup> Ver descrição no item 1.1.4.

### P1222 – Contador de Telegramas CAN Transmitidos

Faixa de Valores: 0 a 65535 Padrão: -

Propriedades: RO

Grupos de acesso via HMI:

01 GRUPOS PARÂMETROS

L 51 PLC

L 130 Parâm. Sistema

Descrição:

Este parâmetro funciona como um contador cíclico, que é incrementado toda vez que um telegrama CAN é transmitido. Fornece um retorno para o operador se o dispositivo está conseguindo comunicar-se com a rede. Este contador é zerado sempre que o equipamento for desligado, feito o reset ou ao atingir o limite máximo do parâmetro.

### P1223 – Contador de Erros de Bus Off

Faixa de Valores: 0 a 65535 Padrão: -

Propriedades: RO

Grupos de acesso via HMI:

01 GRUPOS PARÂMETROS

L 51 PLC

L 130 Parâm. Sistema

Descrição:

Contador cíclico que indica o número vezes que o cartão entrou em estado de *bus off* na rede CAN. Este contador é zerado sempre que o equipamento for desligado, feito o reset ou ao atingir o limite máximo do parâmetro.

### P1224 – Contador de Mensagens CAN Perdidas

Faixa de Valores: 0 a 65535 Padrão: -

Propriedades: RO

Grupos de acesso via HMI:

01 GRUPOS PARÂMETROS

L 51 PLC

L 130 Parâm. Sistema

Descrição:

Contador cíclico que indica o número de mensagens recebidas pela interface CAN, mas que não puderam ser processadas pelo cartão PLC11. Caso o número de mensagens perdidas seja incrementado com frequência, recomenda-se diminuir a taxa de comunicação utilizada para a rede CAN. Este contador é zerado sempre que o equipamento for desligado, feito o reset ou ao atingir o limite máximo do parâmetro.

### P1225 – Estado da Configuração CANopen

Faixa de Valores: 0 = Escravo 1 = Mestre Padrão: -

Propriedades: RO

## Parametrização do cartão PLC11

### Grupos de acesso via HMI:

01 GRUPOS PARAMETROS

L 51 PLC

L 130 Parâm. Sistema

### Descrição:

Parâmetro de leitura para indicação da configuração ativa para rede CANopen. O cartão PLC11 pode operar como escravo (opção padrão) ou mestre (utilizando o software WSCAN) da rede CANopen. Este parâmetro permite observar em qual modo o cartão PLC11 está operando.

Para maiores informações de como operar o cartão PLC11 como mestre da rede CANopen, consulte a documentação do software WLP e software WSCAN.

### P1226 – Estado da Comunicação CANopen

<b>Faixa de</b>	0 = Desabilitado	<b>Padrão:</b> -
<b>Valores:</b>	1 = Reservado	
	2 = Comunicação Habilitada	
	3 = Controle de Erros Habilitado	
	4 = Erro de <i>Guarding</i>	
	5 = Erro de <i>Heartbeat</i>	

**Propriedades:** RO

### Grupos de acesso via HMI:

01 GRUPOS PARÂMETROS

L 51 PLC

L 130 Parâm. Sistema

### Descrição:

Indica o estado do cartão com relação à rede CANopen, informando se o protocolo foi habilitado e se o serviço de controle de erros está ativo (*Node Guarding* ou *Heartbeat*).

### P1227 – Estado do Nó CANopen

<b>Faixa de</b>	0 = Desabilitado	<b>Padrão:</b> -
<b>Valores:</b>	1 = Inicialização	
	2 = Parado	
	3 = Operacional	
	4 = Pré-Operacional	

**Propriedades:** RO

### Grupos de acesso via HMI:

01 GRUPOS PARÂMETROS

L 51 PLC

L 130 Parâm. Sistema

### Descrição:

O cartão PLC11 operando como escravo da rede CANopen possui uma máquina de estados que controla o seu comportamento com relação à comunicação. Este parâmetro indica em qual estado encontra-se o dispositivo.

### P1285 – Protocolo CAN

<b>Faixa de</b>	0 = Desabilitado	<b>Padrão:</b> 0
<b>Valores:</b>	1 = CANopen	
	2 = DeviceNet	

**Propriedades:** CFG

## Parametrização do cartão PLC11

### Grupos de acesso via HMI:

01 GRUPOS PARAMETROS

L 51 PLC

L 130 Parâm. Sistema

### Descrição:

Permite selecionar o protocolo desejado para a interface CAN. Para habilitar o protocolo CANopen, é necessário programar este parâmetro com a opção '1 = CANopen'.

Caso este parâmetro seja alterado, ele somente é válido após o cartão PLC11 ser desligado e ligado novamente.

### P1286 – Endereço CAN

Faixa de 0 a 127

Padrão: 63

Valores:

Propriedades: CFG

### Grupos de acesso via HMI:

01 GRUPOS PARAMETROS

L 51 PLC

L 130 Parâm. Sistema

### Descrição:

Permite programar o endereço utilizado para comunicação CAN do cartão PLC11. É necessário que cada equipamento da rede possua um endereço diferente dos demais. Os endereços válidos para este parâmetro dependem do protocolo programado no P1285:

- P1285 = 1 (CANopen) → endereços válidos: 1 a 127.
- P1285 = 2 (DeviceNet) → endereços válidos: 0 a 63.

Caso este parâmetro seja alterado, ele somente é válido após o cartão PLC11 ser desligado e ligado novamente.

### P1287 – Taxa de Comunicação CAN

Faixa de 0 = 1 Mbit/s

Padrão: 0

Valores:

- 1 = Reservado
- 2 = 500 Kbit/s
- 3 = 250 Kbit/s
- 4 = 125 Kbit/s
- 5 = 100 Kbit/s
- 6 = 50 Kbit/s
- 7 = 20 Kbit/s
- 8 = 10 Kbit/s

Propriedades: CFG

### Grupos de acesso via HMI:

01 GRUPOS PARAMETROS

L 51 PLC

L 130 Parâm. Sistema

### Descrição:

Permite programar o valor desejado para a taxa de comunicação da interface CAN, em bits por segundo. Esta taxa deve ser a mesma para todos os equipamentos conectados na rede.

### P1288 – Reset de Bus Off

Faixa de 0 = Manual

Padrão: 1

Valores:

- 1 = Automático

Propriedades: CFG

**Grupos de acesso via HMI:**

01 GRUPOS PARAMETROS
L 51 PLC
L 130 Parâm. Sistema

**Descrição:**

Permite programar qual o comportamento do cartão PLC11 ao detectar um erro de *bus off* na interface CAN:

Opções	Descrição
0 = Reset Manual	Caso ocorra <i>bus off</i> , será indicado na HMI o alarme/falha 133, a ação programada no parâmetro P1289 será executada e a comunicação será desabilitada. Para que o cartão PLC11 volte a se comunicar através da interface CAN, será necessário desligar e ligar novamente o cartão.
1 = Reset Automático	Caso ocorra <i>bus off</i> , a comunicação será reiniciada automaticamente e o erro será ignorado. Neste caso, não será feita a indicação de alarme/falha na HMI e o cartão PLC11 não executará a ação descrita no P1289.

*Tabela 3.2 – Valores para o parâmetro P1288*

**P1289 – Ação para Erro de Comunicação**

Faixa de Valores:	0 = Indica Alarme 1 = Causa Falha	Padrão: 0
-------------------	--------------------------------------	-----------

Propriedades: CFG

**Grupos de acesso via HMI:**

01 GRUPOS PARAMETROS
L 51 PLC
L 130 Parâm. Sistema

**Descrição:**

Este parâmetro permite selecionar qual a ação deve ser executada pelo cartão PLC11, caso um erro de comunicação seja detectado.

Opções	Descrição
0 = Indica Alarme	Em caso de erro de comunicação, o cartão irá gerar um alarme que será mostrado na HMI do inversor, mas a operação do equipamento não será afetada.
1 = Causa Falha	Em caso de erro de comunicação, o cartão irá gerar uma falha que será mostrado na HMI do inversor interrompendo o controle do motor. Neste caso será necessário fazer o reset do equipamento para que este retorne para operação normal.

*Tabela 3.3 – Valores para o parâmetro P1289*

Para a interface CAN utilizando o protocolo CANopen, são considerados erros de comunicação os seguintes eventos:

- Alarme A833/Falha F933: sem alimentação na interface CAN.
- Alarme A834/Falha F934: *bus off*
- Alarme A835/Falha F935: erro de comunicação CANopen (*Node Guarding*)

A descrição destes alarmes é feita no item 6.

## 4 Dicionário de Objetos

O dicionário de objetos é uma lista com os diversos dados do equipamento que são acessíveis através 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. Contém 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 é utilizada pelo cartão PLC11

Tabela 4.1 – Agrupamentos do dicionário de objetos

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

### 4.1 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.

### 4.2 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.

#### 4.2.1 Tipos básicos

Os seguintes tipos básicos de dados são suportados:

- ☑ **Inteiros com sinal:** existem três tipos de inteiros com sinal suportados pela PLC11, 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 pela PLC11, 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.

#### 4.2.2 Tipos compostos

## Dicionário de Objetos

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 cartão PLC11 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 5.3.

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

Tabela 4.2 – 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 5.3.

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 4.3 – 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 5.2.

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

Tabela 4.4 – 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 <i>record</i>	UNSIGNED8
01h	<i>Vendor-ID</i>	UNSIGNED32
02h	<i>Product Code</i>	UNSIGNED32
03h	<i>Revision Number</i>	UNSIGNED32
04h	<i>Serial Number</i>	UNSIGNED32

Tabela 4.5 – Record para identificação do dispositivo

### 4.2.3 Tipos estendidos

O cartão PLC11 não possui tipos estendidos.

## 4.3 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 cartão PLC11.

Índice	Objeto	Nome	Tipo	Acesso
1000h	VAR	device type	UNSIGNED32	ro

1001h	VAR	error register	UNSIGNED8	ro
1005h	VAR	COB-ID SYNC	UNSIGNED32	rw
100Ch	VAR	guard time	UNSIGNED16	rw
100Dh	VAR	life time factor	UNSIGNED8	rw
1014h	VAR	COB-ID EMCY	UNSIGNED32	ro
1016h	ARRAY	Consumer heartbeat time	UNSIGNED32	rw
1017h	VAR	Producer heartbeat time	UNSIGNED16	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
1401h	RECORD	2nd receive PDO Parameter	PDO CommPar	rw
1402h	RECORD	3rd receive PDO Parameter	PDO CommPar	rw
1403h	RECORD	4th receive PDO Parameter	PDO CommPar	rw
Receive PDO Mapping Parameter				
1600h	RECORD	1st receive PDO mapping	PDO Mapping	rw
1601h	RECORD	2nd receive PDO mapping	PDO Mapping	rw
1602h	RECORD	3rd receive PDO mapping	PDO Mapping	rw
1603h	RECORD	4th receive PDO mapping	PDO Mapping	rw
Transmit PDO Communication Parameter				
1800h	RECORD	1st transmit PDO Parameter	PDO CommPar	rw
1801h	RECORD	2nd transmit PDO Parameter	PDO CommPar	rw
1802h	RECORD	3rd transmit PDO Parameter	PDO CommPar	rw
1803h	RECORD	4th transmit PDO Parameter	PDO CommPar	rw
Transmit PDO Mapping Parameter				
1A00h	RECORD	1st transmit PDO mapping	PDO Mapping	rw
1A01h	RECORD	2nd transmit PDO mapping	PDO Mapping	rw
1A02h	RECORD	3rd transmit PDO mapping	PDO Mapping	rw
1A03h	RECORD	4th transmit PDO mapping	PDO Mapping	rw

**Tabela 4.6** – Lista de objetos da PLC11 – Communication Profile

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

#### 4.4 Manufacturer Specific – Objetos específicos do cartão PLC11

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 cartão PLC11, nesta faixa de objetos foi disponibilizada toda a lista de parâmetros e marcadores da comunicação CAN. Através destes parâmetros e marcadores é possível enviar e receber dados através da rede para operação do equipamento. 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. A tabela a seguir ilustra como estão distribuídos os parâmetros no dicionário de objetos.

Índice	Objeto	Nome	Tipo	Acesso
24B0h	VAR	P1200 – Versão Firmware PLC11	UNSIGNED16	ro
24B1h	VAR	P1201 – Estado da PLC11	UNSIGNED16	ro
24B2h	VAR	P1202 – Ciclo de Scan	UNSIGNED16	ro
...	...	...	...	...
2514h	VAR	P1300 – Parâmetro do Usuário	UNSIGNED16	rw
2515h	VAR	P1301 – Parâmetro do Usuário	UNSIGNED16	rw
...	...	...	...	...
3068h	VAR	%RW0 – Word de leitura da CAN	UNSIGNED16	ro
3069h	VAR	%RW1 – Word de leitura da CAN	UNSIGNED16	ro
...	...	...	...	...
3130h	VAR	%RB0 – Byte de leitura da CAN	UNSIGNED16	ro
3131h	VAR	%RB1 – Byte de leitura da CAN	UNSIGNED16	ro
...	...	...	...	...
31F8h	VAR	%WW0 – Word de escrita da CAN	UNSIGNED16	rw
31F9h	VAR	%WW1 – Word de escrita da CAN	UNSIGNED16	rw
...	...	...	...	...
32C0h	VAR	%WB0 – Byte de escrita da CAN	UNSIGNED16	rw
32C1h	VAR	%WB1 – Byte de escrita da CAN	UNSIGNED16	rw

...	...	...	...	...
-----	-----	-----	-----	-----

**Tabela 4.7** – Lista de objetos da PLC11 – Manufacturer Specific

Para a lista completa e uma descrição detalhada dos parâmetros, consulte o manual do cartão PLC11. É necessário conhecer como programar e operar o cartão PLC11 para poder operar corretamente o equipamento via rede CANopen.

## 5 Descrição dos objetos de comunicação

Nesta seção são descritos detalhadamente cada um dos objetos citados na Tabela 4.6, além de descrever também o funcionamento dos objetos de comunicação (COBs) referenciados no item 1.7. É necessário conhecer as funções disponíveis e como programar o cartão PLC11 para utilização destes objetos.

### 5.1 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.

#### 5.1.1 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	0000.0000h

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. O cartão PLC11 não segue um *Profile* definido pela especificação CAN, logo este objeto apresenta valor 0 (zero).

#### 5.1.2 Objeto 1001h – Error Register

Este objeto indica a ocorrência ou não de erro no dispositivo. O tipo de erro registrado para o cartão PLC11 segue o descrito pela Tabela 5.1.

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 5.1 – 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.

#### 5.1.3 Objeto 1018h – Identity Object

Traz informações gerais sobre o dispositivo.

## Descrição dos objetos de comunicação

---

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	0000.0202h

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 cartão

O Vendor ID é um número que identifica o fabricante junto à CiA. O código do produto é definido pelo fabricante de acordo com o tipo de produto. O número da revisão representa a versão de firmware do equipamento. O sub-índice 4 é um número serial único para cada cartão PLC11 em rede CANopen.

### 5.2 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.

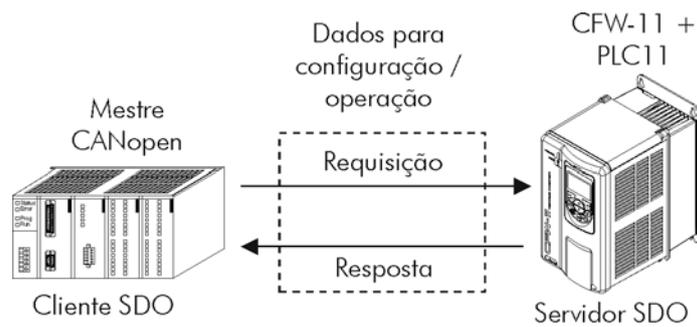


Figura 5.1 – Comunicação entre Cliente e Servidor SDO

### 5.2.1 Objeto 1200h – Servidor SDO

O cartão PLC11 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 marcadores do equipamento. Todo o servidor SDO possui um objeto, do tipo SDO\_PARAMETER (ver item 4.2.2), 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

### 5.2.2 Funcionamento dos SDOs

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

Identificador	8 bytes de dados							
11 bits	Comando	Índice		Sub-índice	Dados do objeto			
	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:

## Descrição dos objetos de comunicação

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 5.2** – 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 5.3** – 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 cartão PLC11 (objetos a partir do índice 2000h), este valor é salvo na memória não volátil do cartão. 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 cartão PLC11 no endereço 1, a leitura do objeto identificado pelo índice 24B0h, sub-índice 0 (zero) (equivale ao parâmetro P1200), que representa um inteiro sem sinal de 16 bits. O telegrama do mestre possui a seguinte forma:

Identificador	Comando	Índice	Sub-índice	Dados			
601h	40h	B0h	24h	00h	00h	00h	00h

O cartão responde à requisição, indicando que o valor para o referido objeto é igual a 100<sup>5</sup>:

Identificador	Comando	Índice	Sub-índice	Dados			
581h	4Bh	B0h	24h	64h	00h	00h	00h

## 5.3 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.

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.

<sup>5</sup> Não esquecer que qualquer dado do tipo inteiro, a ordem de transferência dos bytes vai do menos significativo até o mais significativo.

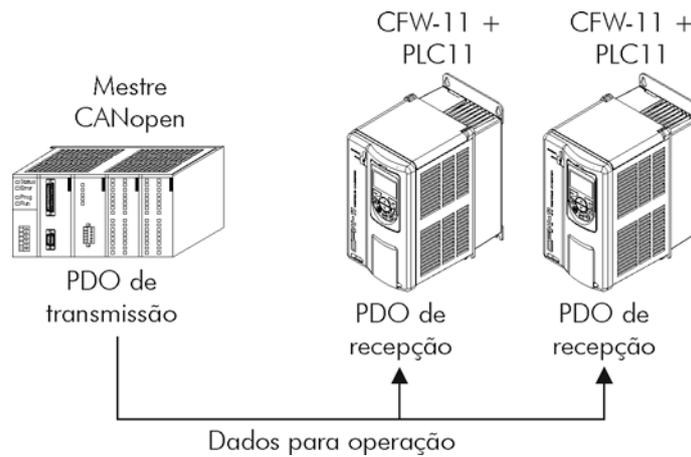


Figura 5.2 – Comunicação utilizando PDOs



**NOTA!**

PDOs somente podem ser transmitidos ou recebidos quando o dispositivo está no estado operacional. A Figura 5.5 ilustra os estados disponíveis para um nó da rede CANopen.

**5.3.1 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 esta condição. Usualmente, apenas informações necessárias para a operação do dispositivo são mapeáveis, como comandos e, 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 o cartão PLC11, os marcadores de leitura da CAN (Word e Byte) são mapeáveis nos PDOs de transmissão, enquanto os marcadores de escrita da CAN (Word e Byte) são mapeáveis nos PDOs de recepção.

Índice	Objeto	Nome	Tipo	Acesso
3068h	VAR	%RW0 – Word de leitura da CAN	UNSIGNED16	Ro
3069h	VAR	%RW1 – Word de leitura da CAN	UNSIGNED16	Ro
...	...	...	...	...
3130h	VAR	%RB0 – Byte de leitura da CAN	UNSIGNED16	Ro
3131h	VAR	%RB1 – Byte de leitura da CAN	UNSIGNED16	Ro
...	...	...	...	...
31F8h	VAR	%WW0 – Word de escrita da CAN	UNSIGNED16	Rw
31F9h	VAR	%WW1 – Word de escrita da CAN	UNSIGNED16	Rw
...	...	...	...	...
32C0h	VAR	%WB0 – Byte de escrita da CAN	UNSIGNED16	Rw
32C1h	VAR	%WB1 – Byte de escrita da CAN	UNSIGNED16	Rw
...	...	...	...	...

Tabela 5.4 – Exemplos de marcadores mapeáveis para PDOs

**5.3.2 PDOs de recepção**

Os PDOs de recepção, ou RPDOs, são responsáveis por receber dados que outros dispositivos enviam para a rede CANopen. O cartão PLC11 possui 4 PDOs de recepção, cada um podendo receber até 8 bytes de dados. Cada 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

## Descrição dos objetos de comunicação

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	1400h: 200h + Node-ID 1401h: 300h + Node-ID 1402h: 400h + Node-ID 1403h: 500h + 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, sempre 0
10 – 0 (LSB)	X	COB-ID de 11 bits

**Tabela 5.5** – 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 cartão PLC11 não utiliza identificadores de 29 bits, os bits de 28 até 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				
	Cíclico	Acíclico	Síncrono	Assíncrono	RTR
0		•	•		
1 – 240	•		•		
241 – 251	Reservado				
252			•		•
253				•	•
254				•	
255				•	

**Tabela 5.6** – Descrição do tipo de transmissão

- ☑ **Valores 0 – 240:** qualquer RPDOs programado nesta faixa possui o mesmo funcionamento. Ao detectar uma mensagem, ele irá receber os dados, porém não atualizará os valores recebidos até detectar o próximo telegrama 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

## Descrição dos objetos de comunicação

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	1600h: 4 1601h: 4 1602h: 4 1603h: 4

Sub-índice	1
Descrição	1º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	1600h: 3068.0010h 1601h: 306C.0010h 1602h: 3070.0010h 1603h: 3074.0010h

Sub-índice	2
Descrição	2º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	1600h: 3069.0010h 1601h: 306D.0010h 1602h: 3071.0010h 1603h: 3075.0010h

Sub-índice	3
Descrição	3º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	1600h: 306A.0010h 1601h: 306E.0010h 1602h: 3072.0010h 1603h: 3076.0010h

Sub-índice	4
Descrição	4º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	1600h: 306B.0010h 1601h: 306F.0010h 1602h: 3073.0010h 1603h: 3077.0010h

Este parâmetro indica os objetos mapeados nos PDOs de recepção da PLC11. Para cada 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-índice<sup>6</sup> e tamanho (em bits) em um campo UNSIGNED32, com o seguinte formato:

<sup>6</sup> Caso o objeto seja do tipo VAR e não possua sub-índice, deve ser indicado o valor 0 (zero) para o sub-índice.

## Descrição dos objetos de comunicação

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 = 4:** o RPDO possui quatro objetos mapeados.
- ☑ **Sub-índices 1 = 3068.0010h:** representa o marcador de word de leitura da CAN 0.
- ☑ **Sub-índices 1 = 3069.0010h:** representa o marcador de word de leitura da CAN 1.
- ☑ **Sub-índices 1 = 306A.0010h:** representa o marcador de word de leitura da CAN 2.
- ☑ **Sub-índices 1 = 306B.0010h:** representa o marcador de word de leitura da CAN 3.

Desta forma, sempre que este PDO receber um telegrama, ele vai saber que o telegrama deverá conter oito bytes de dados, com o conteúdo para os marcadores citados. O programa aplicativo elaborado para a PLC11 deve utilizar estes marcadores como fonte de dados recebidos pela rede.

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 cartão. 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 cartão PLC11 esteja no estado operacional.

### 5.3.3 PDOs de transmissão

Os PDOs de transmissão, ou TPDOs, como o nome diz, são responsáveis por transmitir dados para a rede CANopen. O cartão PLC11 possui 4 PDOs de transmissão, cada um 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	1800h: 180h + Node-ID 1801h: 280h + Node-ID 1802h: 380h + Node-ID 1803h: 480h + Node-ID

## Descrição dos objetos de comunicação

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 5.5*.

O sub-índice 2 indica o tipo de transmissão deste objeto, que segue o descrito pela *Tabela 5.6*. Porém seu funcionamento é diferente para PDOs de transmissão:

- ☑ **Valor 0:** indica que a transmissão deve ocorrer imediatamente após a recepção de um telegrama SYNC, mas não periodicamente.
- ☑ **Valores 1 – 240:** o PDO deve ser transmitido a cada telegrama SYNC detectado (ou ocorrências múltiplas de SYNC, de acordo com o número escolhido entre 1 e 240).
- ☑ **Valor 252:** indica que o conteúdo da mensagem deve ser atualizado (mas não enviado), após a recepção de um telegrama SYNC. O envio da mensagem deve ser feito após a recepção de um frame remoto (RTR frame).
- ☑ **Valor 253:** o PDO deve atualizar e enviar uma mensagem assim que receber um frame remoto.
- ☑ **Valores 254:** o objeto deve ser transmitido de acordo com o timer programado no sub-índice 5.
- ☑ **Valores 255:** o objeto é transmitido automaticamente quando o valor de algum dos objetos mapeados neste PDO for alterado. Funciona por alteração de estado (*Change Of State*). Este tipo também permite que o PDO seja 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 o 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 cartão PLC11 é 1ms.
- ☑ É importante observar o tempo entre transmissões programado no sub-índice 3 principalmente

## Descrição dos objetos de comunicação

quando o PDO for programado com o valor 255 no sub-índice 2 (*Change Of State*).

- 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	1A00h: 4 1A01h: 4 1A02h: 4 1A03h: 4

Sub-índice	1
Descrição	1º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	1A00h: 31F8.0010h 1A01h: 31FC.0010h 1A02h: 3200.0010h 1A03h: 3204.0010h

Sub-índice	2
Descrição	2º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	1A00h: 31F9.0010h 1A01h: 31FD.0010h 1A02h: 3201.0010h 1A03h: 3205.0010h

Sub-índice	3
Descrição	3º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	1A00h: 31FA.0010h 1A01h: 31FE.0010h 1A02h: 3202.0010h 1A03h: 3206.0010h

Sub-índice	4
Descrição	4º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	1A00h: 31FB.0010h 1A01h: 31FF.0010h 1A02h: 3203.0010h 1A03h: 3207.0010h

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 quarto PDO de transmissão, temos:

- ☑ **Sub-índice 0 = 2:** este PDO possui dois objetos mapeados.
- ☑ **Sub-índice 1 = 30F8.0010h:** representa o marcador de word de escrita da CAN 0.
- ☑ **Sub-índice 1 = 30F9.0010h:** representa o marcador de word de escrita da CAN 1.
- ☑ **Sub-índice 1 = 30FA.0010h:** representa o marcador de word de escrita da CAN 2.
- ☑ **Sub-índice 1 = 30FB.0010h:** representa o marcador de word de escrita da CAN 3.

Desta forma, sempre que este PDO for transmitir seus dados, ele vai elaborar o seu telegrama contendo oito bytes de dados,, com o conteúdo para os marcadores citados. O programa aplicativo elaborado para a PLC11 deve utilizar estes marcadores como fonte de dados enviados para a rede.

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.

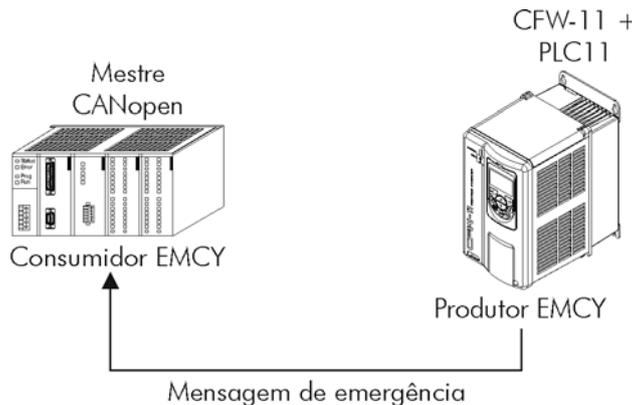


**NOTA!**

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.

**5.4 Emergency Object – EMCY**

O objeto de emergência (EMCY) é utilizado para sinalizar a ocorrência de um erro no dispositivo. Sempre que uma falha ocorrer no cartão PLC11, este objeto irá enviar uma mensagem de emergência para a rede. Esta mensagem poderá ser interpretada por um consumidor EMCY (usualmente o mestre da rede), que poderá tomar uma ação de acordo com o programado para a aplicação, como fazer o reset de erro ou desabilitar os demais dispositivos da rede.



*Figura 5.3 - EMCY*

Ao transmitir uma mensagem, três informações são enviadas nos oito bytes de dados do telegrama: o código de erro CiA, o objeto 1001h (*error register*) e o código da falha do cartão PLC11. O telegrama possui então a seguinte estrutura:

byte 0	byte 1	byte 2	byte 3	byte 4	byte 5	byte 6	byte 7
Código do erro CIA		Objeto 1001h Error Register	Código da falha PLC11	Reservado			

Na PLC11, este objeto está sempre ativo e, portanto, sempre reportará para a rede a ocorrência de erros no cartão. Existe um único parâmetro que possibilita ler qual o COB-ID do objeto, ou seja, qual o identificador do telegrama para as mensagens de erro enviadas.

Índice	1014h
Nome	COB-ID emergency message
Objeto	VAR
Tipo	UNSIGNED32

Acesso	ro
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	80h + Node-ID

### 5.5 Synchronization Object – SYNC

Este objeto é transmitido com o objetivo de permitir a sincronização de eventos entre os dispositivos da rede CANopen. Ele é transmitido por um produtor SYNC, e os dispositivos que detectam a sua transmissão são denominados consumidores SYNC.

O cartão PLC11 possui a função de consumidor SYNC e, portanto, pode programar seus PDOs para serem síncronos. Como descrito na Tabela 5.6, PDOs síncronos são aqueles relacionados com o objeto de sincronização, e portanto podem ser programados para serem transmitidos ou atualizados com base neste objeto.

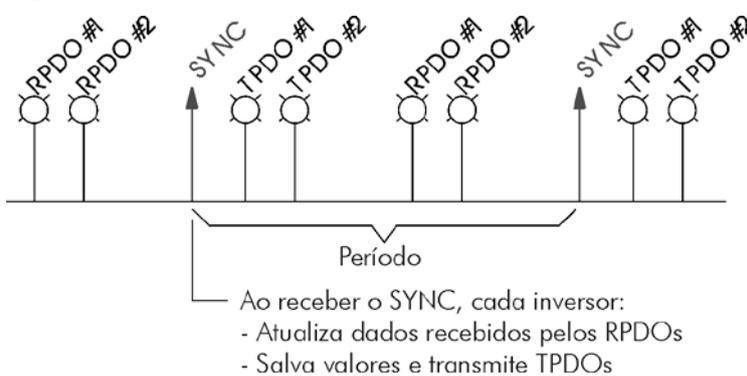


Figura 5.4 – SYNC

A mensagem SYNC transmitida pelo produtor não possui dado algum em seu campo de dados, pois seu objetivo é fornecer uma base de tempo para os demais objetos. Na PLC11, existe um objeto para a configuração do COB-ID do consumidor SYNC.

Índice	1015h
Nome	COB-ID SYNC
Objeto	VAR
Tipo	UNSIGNED32

Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	80h



**NOTA!**

Deve-se observar o tempo programado no produtor para o período dos telegramas SYNC, de acordo com a taxa de transmissão utilizada e o número de PDOs síncronos a serem transmitidos. É necessário que haja tempo suficiente para a transmissão destes objetos, e também é recomendado que haja folga para possibilitar o envio de mensagens assíncronas, como EMCY, PDOs assíncronos e SDOs.

### 5.6 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 cartão PLC11 estão disponíveis os serviços de controle do nó e de controle de erros (utilizando Node Guarding ou Heartbeat).

#### 5.6.1 Controle dos estados do escravo

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

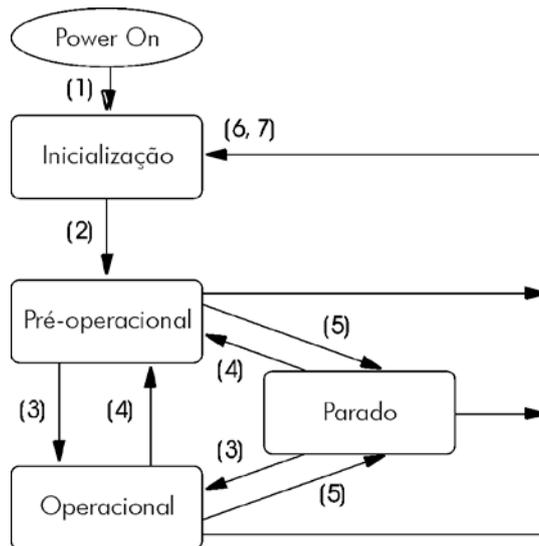


Figura 5.5 – Diagrama de estados do nó CANopen

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 5.7 – 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 5.8 – Objetos acessíveis em cada estado

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:

Identificador	byte 1	byte 2
00h	Código do comando	Node-ID destino

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

Tabela 5.9 – 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 5.5). O comando *Reset node* faz com que o cartão PLC11 execute um reset completo do dispositivo, enquanto que o comando *Reset comunicação* faz com que o dispositivo reinicialize apenas os objetos relativos à comunicação CANopen.

### 5.6.2 Controle de Erros – Node Guarding

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.

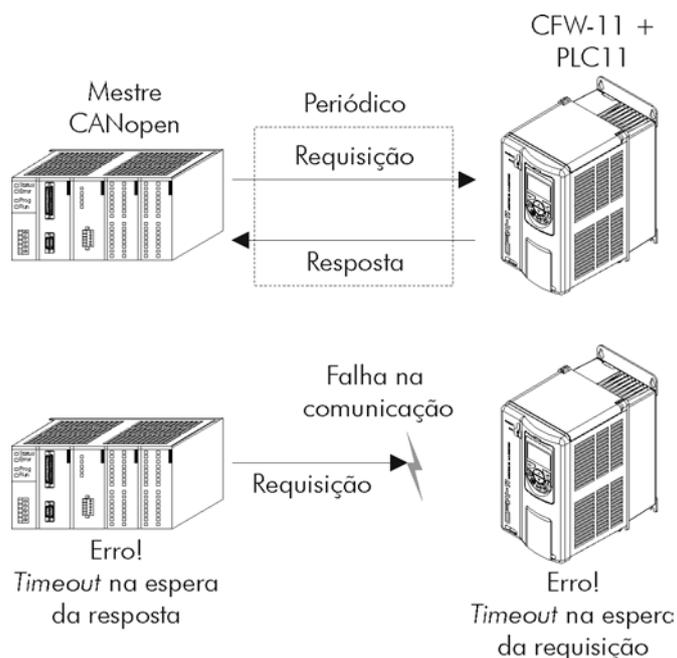


Figura 5.6 – Serviço de controle de erros – Node Guarding

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

## Descrição dos objetos de comunicação

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 cartão PLC11 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, a PLC11 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 cartão PLC11 detecte um erro utilizando este mecanismo, ele irá automaticamente para o estado pré-operacional e indicará A835/F835 na HMI do inversor. A ocorrência deste erro também pode ser observada através do parâmetro P1226. É possível também programar o cartão para tomar uma ação quando este erro ocorrer, através do parâmetro P1289. Consulte o item 3 para a descrição detalhada dos parâmetros.



### NOTA!

- ☑ Este objeto está ativo mesmo no estado parado (consulte a Tabela 5.8).
- ☑ 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.
- ☑ O valor mínimo aceito para o cartão PLC11 é de 1 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.
- ☑ Para cada escravo, somente um dos serviços – *Heartbeat* ou *Node Guarding* – pode ser habilitado.

### 5.6.3 Controle de Erros – *Heartbeat*

A detecção de erros através do mecanismo de *heartbeat* é feita utilizando dois tipos de objetos: o produtor *heartbeat* e o consumidor *heartbeat*. O produtor é responsável por enviar telegramas periódicos para a rede, simulando uma batida do coração, indicando que a comunicação está ativa e sem erros. Um ou mais consumidores podem monitorar estes telegramas periódicos e, caso estes telegramas deixem de ocorrer, significa que algum problema de comunicação ocorreu.

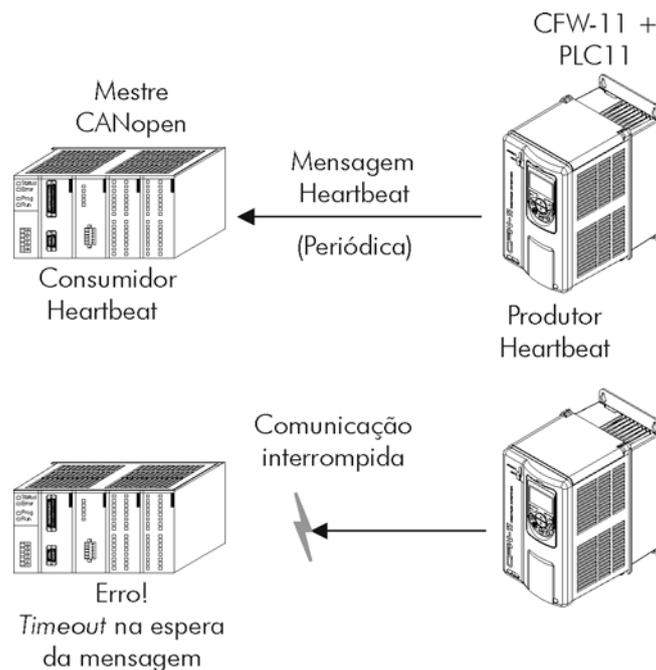


Figura 5.7 – Serviço de controle de erros – Heartbeat

Um mesmo dispositivo da rede pode ser produtor e consumidor de mensagens *heartbeat*. Por exemplo, o mestre da rede pode consumir mensagens enviadas por um dispositivo da rede, permitindo detectar problemas de comunicação com o escravo, e ao mesmo tempo o escravo pode consumir mensagens *heartbeat* enviadas pelo mestre, também possibilitando ao escravo detectar erros na comunicação com o mestre.

O cartão PLC11 possui os serviços de produtor e consumidor *heartbeat*. Como consumidor, é possível programar até 4 diferentes produtores para serem monitorados pelo cartão:

Índice	1016h
Nome	Consumer Heartbeat Time
Objeto	ARRAY
Tipo	UNSIGNED32

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

Sub-índices	1 – 4
Descrição	Consumer Heartbeat Time 1 – 4
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	0

Nos sub-índices de 1 até quatro, é possível programar o consumidor escrevendo um valor no seguinte formato:

UNSIGNED32		
Reservado (8 bits)	Node-ID (8 bits)	Heartbeat time (16 bits)

- Node-ID:** permite programar o Node-ID do produtor *heartbeat* o qual se deseja monitorar.
- Heartbeat time:** permite programar o tempo, em múltiplos de 1 milissegundo, até a detecção de erro, caso nenhuma mensagem do produtor seja recebida. O valor 0 (zero) neste campo desabilita o consumidor.

Depois de configurado, o consumidor *heartbeat* inicia a monitoração após o primeiro telegrama enviado pelo produtor. Caso seja detectado erro pelo fato do consumidor deixar de receber mensagens do produtor *heartbeat*,

## Descrição dos objetos de comunicação

o cartão PLC11 irá automaticamente para o estado pré-operacional e indicará A835/F835 na HMI do inversor. A ocorrência deste erro também pode ser observada através do parâmetro P1226.

Como produtor, o cartão PLC11 possui um objeto para configuração deste serviço:

Índice	1017h
Nome	Producer Heartbeat Time
Objeto	VAR
Tipo	UNSIGNED16

Acesso	rw
Mapeável	Não
Faixa	UNSIGNED8
Valor Padrão	0

O objeto 1017h permite programar o tempo em milissegundos no qual o produtor envie um telegrama *heartbeat* para a rede. Uma vez programado, o cartão inicia a transmissão de mensagens com o seguinte formato:

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



### NOTA!

- ☑ Este objeto está ativo mesmo no estado parado (ver Tabela 5.8).
- ☑ O valor 0 (zero) em um dos objetos desabilita esta função.
- ☑ Depois de detectado o erro, caso o serviço seja habilitado mais uma vez, a indicação de alarme é retirada da HMI. No caso do cartão estar programa para indicação de falha, é necessário fazer o reset do equipamento.
- ☑ O valor de tempo programado para o consumidor deve ser maior do que o programado para o respectivo produtor, preferencialmente programar o consumidor com valores múltiplos do programado para o consumidor.
- ☑ Para cada escravo, somente um dos serviços – *Heartbeat* ou *Node Guarding* – pode ser habilitado.

## 5.7 Procedimento de inicialização

Uma vez conhecido o funcionamento dos objetos disponíveis para o cartão PLC11, 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:

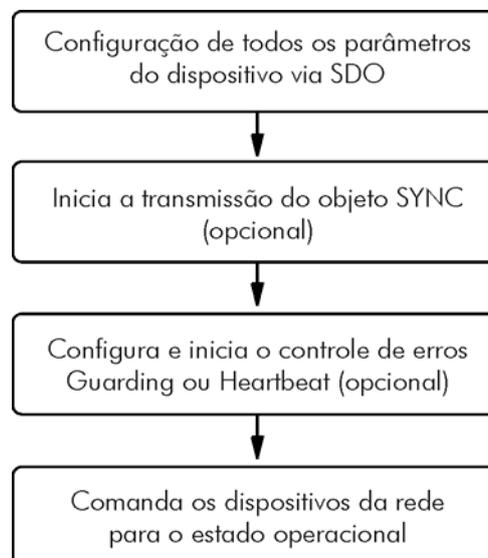


Figura 5.8 – Fluxograma do processo de inicialização

É necessário observar que os objetos de comunicação da PLC11 (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

## Descrição dos objetos de comunicação

---

representam os parâmetros), estes são armazenados na memória não volátil e, portanto, podem ser parametrizados uma única vez.

## 6 Falhas e Alarmes Relacionados com a Comunicação CANopen

Os erros relacionados à comunicação CANopen podem ser programados para serem interpretados como alarmes ou falhas através do parâmetro P1289. Alarmes apenas fazem a indicação do evento na HMI do inversor, enquanto falhas causam o bloqueio do motor e requerem reset do equipamento.

### A833/F833 – Sem alimentação na Interface CAN

#### Descrição:

Indica que a interface CAN não possui alimentação entre os pinos 1 e 5 do conector.

#### Atuação:

Para que seja possível enviar e receber telegramas através da interface CAN, é necessário fornecer alimentação externa para o circuito de interface.

Se for detectada a falta de alimentação na interface CAN, a comunicação é desabilitada e será mostrado A133/F133 na HMI do inversor. Para alarmes, caso a alimentação do circuito seja restabelecida, a indicação de alarme será retirada da HMI e a comunicação CANopen será reiniciada.

#### Possíveis Causas/Correção:

- Medir se existe tensão entre os pinos 1 e 5 do conector da interface CAN, conforme indicado na Tabela 2.2.
- Verificar se os cabos de alimentação e comunicação não estão trocados ou invertidos.
- Verificar problemas de contato no cabo ou no conector da interface CAN.

### A834/F834 – Bus Off

#### Descrição:

Detectado erro de *bus off* na interface CAN.

#### Atuação:

Caso o número de erros de recepção ou transmissão detectados pela interface CAN seja muito elevado<sup>7</sup>, o controlador CAN pode ser levado ao estado de *bus off*, onde ele interrompe a comunicação e desabilita a interface CAN.

Caso ocorra erro de *bus off*, a comunicação CAN é desabilitada e será mostrado A134/F134 na HMI do inversor. Para que a comunicação seja restabelecida, é necessário desligar e ligar novamente o inversor, ou retirar e ligar novamente a alimentação da interface CAN, para que a comunicação seja reiniciada.

#### Possíveis Causas/Correção:

- Verificar curto-circuito nos cabos de transmissão do circuito CAN.
- Verificar se os cabos não estão trocados ou invertidos.
- Verificar se resistores de terminação com valores corretos foram colocados somente nos extremos do barramento principal.
- Verificar se a instalação da rede CAN foi feita de maneira adequada.
- Verificar a taxa de comunicação é a mesma para todos os participantes da rede CAN.

### A835/F835 – Erro de Node Guarding/Heartbeat

#### Descrição:

Controle de erros da comunicação CANopen detectou erro de comunicação utilizando o mecanismo de *guarding* ou no consumidor *heartbeat*.

#### Atuação:

O controle de erros deve ser habilitado pelo mestre da rede CANopen. Utilizando o *Node Guarding*, o mestre troca mensagens periódicas com o escravo (neste caso, a PLC11), em um período pré-determinado. Caso a comunicação seja interrompida por algum motivo, tanto mestre quanto escravo poderão detectar erro na

<sup>7</sup> Para maiores informações sobre detecção de erros, consultar especificação CAN.

## Falhas e Alarmes Relacionados com a Comunicação CANopen

---

comunicação pelo *timeout* na troca destas mensagens. Utilizando o consumidor *heartbeat*, o escravo recebe mensagens periódicas de outro dispositivo da rede. Caso a comunicação se interrompida por algum motivo o escravo poderá detectar erro na comunicação pelo *timeout* na troca destas mensagens.

Caso ocorra erro de *guarding* ou *heartbeat*, será mostrado A135/F135 na HMI do inversor. Para alarmes, caso este controle de erros seja habilitado novamente, a indicação de alarme será retirada da HMI.

### Possíveis Causas/Correção:

- Verificar os tempos programados no mestre e no escravo para troca de mensagens. Para evitar problemas devido a atrasos na transmissão e diferenças na contagem dos tempos, recomenda-se que os valores programados para troca de mensagens no mestre seja um pouco menor que os tempos programados para detecção de erros pelo escravo.
- Verificar se o mestre está enviando os telegramas de *guarding* ou *heartbeat* no tempo programado.
- Verificar problemas na comunicação que possam ocasionar perda de telegramas ou atrasos na transmissão.