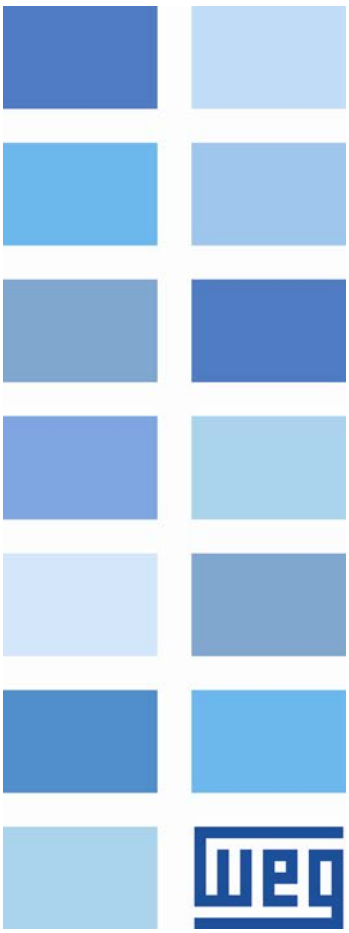
CANopen

RUW06

Manual do Usuário



Manual do Usuário CANopen

Série: RUW06

Idioma: Português

N ° do Documento: 10002213831 / 01

Data da Publicação: 07/2017

SUMÁRIO

SUMÁRIO 3

SOBRE O MANUAL 5

ABREVIAÇÕES E DEFINIÇÕES 5
REPRESENTAÇÃO NUMÉRICA 5
DOCUMENTOS 5

1 CONEXÕES E INSTALAÇÃO 6

1.1 INSTALAÇÃO MECÂNICA 6
1.1.1 Condições Ambientais 6
1.1.2 Posicionamento e Fixação 6
1.2 INSTALAÇÃO ELÉTRICA 7
1.3 ENERGIZAÇÃO 7
1.4 CONEXÃO NA REDE CANOPEN 8
1.5 CONEXÃO DAS ENTRADAS ANALÓGICAS 8

2 INTRODUÇÃO À COMUNICAÇÃO 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 10
2.4 ENDEREÇO NA REDE CANOPEN 10
2.5 ACESSO AOS DADOS 10
2.6 TRANSMISSÃO DE DADOS 10
2.7 OBJETOS RESPONSÁVEIS PELA COMUNICAÇÃO – COBS 11
2.8 COB-ID 11
2.9 ARQUIVO EDS 12

3 INTERFACE DE COMUNICAÇÃO CANOPEN 13

3.1 CARACTERÍSTICAS DA INTERFACE CAN 13
3.2 PINAGEM DO CONECTOR 13
3.3 FONTE DE ALIMENTAÇÃO 13
3.4 INDICAÇÕES 13

4 INSTALAÇÃO DA REDE CANOPEN 15

4.1 TAXA DE COMUNICAÇÃO 15
4.2 ENDEREÇO NA REDE CANOPEN 15
4.3 RESISTORES DE TERMINAÇÃO 15
4.4 CABO 16
4.5 LIGAÇÃO NA REDE 16

5 CONFIGURAÇÃO 17

5.1 ENDEREÇO REDE CANOPEN 17
5.2 TAXA DE COMUNICAÇÃO – BAUD RATE 18
5.3 CONFIGURADOR WPSCAN 19

6 DICIONÁRIO DE OBJETOS 20

6.1 ESTRUTURA DO DICIONÁRIO 20
6.2 TIPOS DE DADOS 20
6.3 COMMUNICATION PROFILE – OBJETOS PARA COMUNICAÇÃO 20
6.4 MANUFACTURER SPECIFIC – OBJETOS ESPECÍFICOS DO FABRICANTE 21
6.4.1 Objeto 2003h – Bus Off autoreset 21
6.4.2 Objeto 2004h – Analogue Channel Status 22
6.4.3 Objeto 2005h – Read Analogue Input 24bits 22
6.4.4 Objeto 2006h – Read Analogue Input 24bits process 23
6.4.5 Objeto 2007h – Read Analogue Input 16 MSB process 23
6.4.6 Objeto 2008h – Analogue Channel Enable 24
6.4.7 Objeto 200Ah – Analogue Channel Unit 24
6.4.8 Objeto 200Ch – Analogue Channel Filter 25
6.4.9 Objeto 200Dh – Analogue Channel Gain 26
6.4.10 Objeto 200Eh – Analogue Channel Offset 26
6.4.11 Objeto 200Fh – Strain Gage Full Scale 27
6.4.12 Objeto 2010h – Strain Gage Sensibility 28
6.4.13 Objeto 2011h – Aquisition Sample Rate 28
6.4.14 Objeto 2012h – Sample Maximum Variation 29
6.4.15 Objeto 2013h – Discard Value 29
6.4.16 Objeto 2014h – Filter TAU Constant 30
6.4.17 Objeto 2015h – Variation Step 30
6.5 DEVICE PROFILE – OBJETOS COMUNS PARA MODULOS I/O 31

7 DESCRIÇÃO DOS OBJETOS DE COMUNICAÇÃO 32

7.1 OBJETOS DE IDENTIFICAÇÃO 32
7.1.1 Objeto 1000h – Device Type 32
7.1.2 Objeto 1001h – Error Register 32
7.1.3 Objeto 1018h – Identity Object 32
7.2 SERVICE DATA OBJECTS – SDOS 33
7.2.1 Objeto 1200h – Servidor SDO 33
7.2.2 Funcionamento dos SDOs 34
7.3 PROCESS DATA OBJECTS – PDOS 35
7.3.1 Objetos Mapeáveis para os PDOs 35
7.3.2 PDOs de Recepção 36
7.3.3 PDOs de Transmissão 36
7.4 SYNCHRONIZATION OBJECT – SYNC 38
7.5 NETWORK MANAGEMENT – NMT 39
7.5.1 Controle dos Estados do Escravo 39
7.5.2 Controle de Erros – Node Guarding 40
7.5.3 Controle de Erros – Heartbeat 41
7.6 PROCEDIMENTO DE INICIALIZAÇÃO 43

8 DESCRIÇÃO DOS OBJETOS PARA MÓDULO I/O 45

8.1 MODULO DE ENTRADA ANALÓCIA 45
8.1.1 Objeto 6401h - Read Analogue Input 16-Bit 45
8.1.2 Objeto 6423h – Analogue Input Global Interrupt Enable 45

9 FALHAS E ALARMES RELACIONADOS COM A COMUNICAÇÃO CANOPEN 47

SEM ALIMENTAÇÃO NA INTERFACE CAN 47
BUS OFF 47
NODE GUARDING/HEARTBEAT 47

SOBRE O MANUAL

Este manual fornece a descrição necessária para a operação da remota RUW06 utilizando o protocolo CANopen. Este manual deve ser utilizado em conjunto com o Guia de Instalação, Configuração e Operação da RUW06.

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 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
CiA DSP 401 Device Profile for Generic I/O Modules	2.1	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.

1 CONEXÕES E INSTALAÇÃO

Este capítulo descreve os procedimentos de instalação elétrica e mecânica da remota RUW06. As orientações e sugestões devem ser seguidas visando a segurança de pessoas, equipamentos e o correto funcionamento.

1.1 INSTALAÇÃO MECÂNICA

1.1.1 Condições Ambientais

Evitar:

- Exposição direta a raios solares, chuva, umidade excessiva ou maresia;
- Gases ou líquidos explosivos ou corrosivos;
- Vibração excessiva;
- Poeira, partículas metálicas ou óleos suspensos no ar.

Condições ambientais permitidas para funcionamento:

- Temperatura: -10 °C a 50 °C;
- Umidade relativa do ar: 5 % a 90 % sem condensação;
- Altitude máxima: até 1000m – condições nominais.

1.1.2 Posicionamento e Fixação

A remota RUW06 pode ser instalada em qualquer posição. Pode ser montada em trilho DIN 35mm ou através de parafusos, conforme Figura 1.1

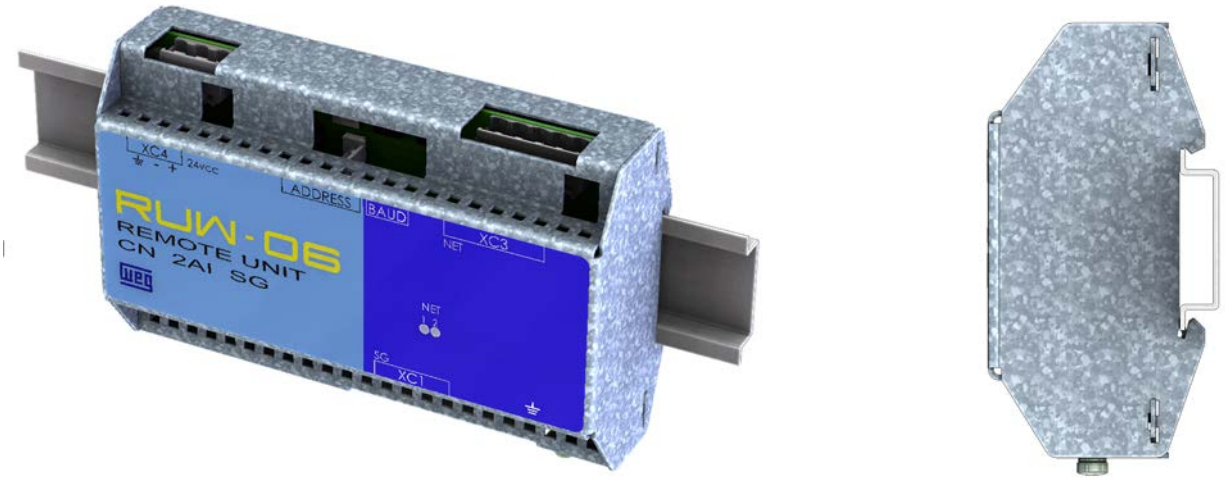


Figura 1.1: *Fixação da RUW06*

1.2 INSTALAÇÃO ELÉTRICA

NOTA!
As informações a seguir têm a intenção de servir como guia para se obter uma instalação correta. Siga também as normas de instalações elétricas aplicáveis.

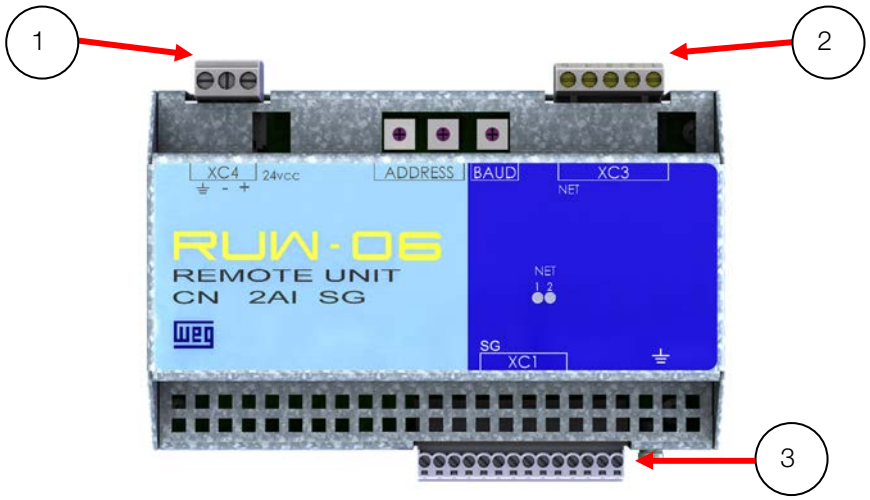


Figura 1.2: *Conexões da RUW06*

1 – Conector XC4 alimentação;
2 – Conector XC3 para rede CANopen;
3 – Conector XC1 entradas analógicas.

1.3 ENERGIZAÇÃO

NOTA!
A tensão de alimentação da RUW06 deve ser 24 Vcc +/- 15 %.

A RUW06 é energizada através do conector XC4, conforme Figura 1.3.

- Tensão de alimentação: 24 Vcc +/- 15%;

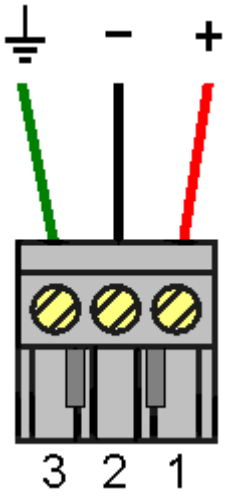


Figura 1.3: *Conector de alimentação da remotaRUW06*

- Terminal 1: (+) Vcc;
- Terminal 2: (-) GND;
- Terminal 3: PE (Terra).

1.4 CONEXÃO NA REDE CANOPEN

A RUW06 possui um LED na cor verde para indicar que a interface está alimentada. Este led está posicionado ao lado do conector CAN.

NOTA!
Observar o esquema de ligação do conector CAN conforme item 3.2.

1.5 CONEXÃO DAS ENTRADAS ANALÓGICAS

A RUW06 apresenta 2 entradas para células de carga, sendo possível ligações a 6 fios e a 4 fios:

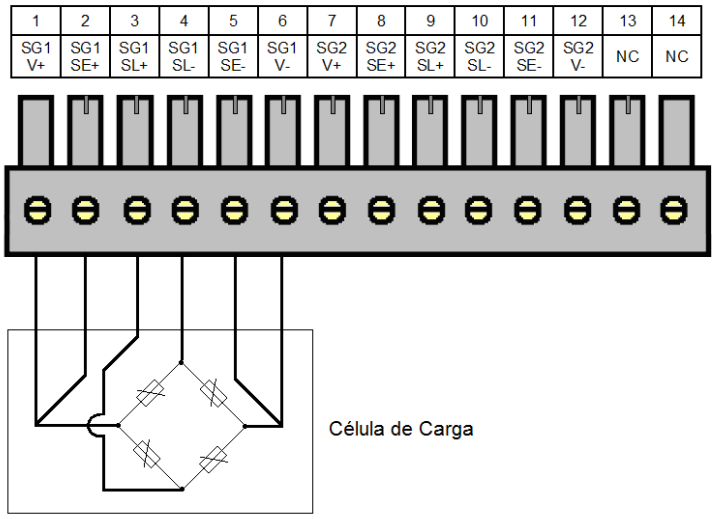


Figura 1.4: *Esquema de ligação da célula de carga a 6 fios*

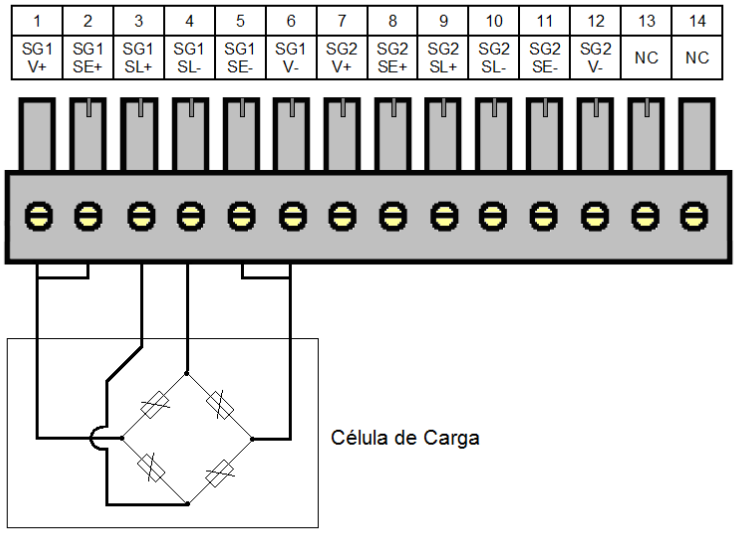


Figura 1.5: *Esquema de ligação da célula de carga a 4 fios (jumpers entre V e SE)*

2 INTRODUÇÃO À COMUNICAÇÃO CANOPEN

Para a operação de um equipamento em rede CANopen, é necessário conhecer a forma como a comunicação é feita. Este item traz uma descrição geral do funcionamento do protocolo CANopen, contendo as funções utilizadas pela RUW06. Para uma descrição mais detalhada pode-se consultar a especificação do protocolo.

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 arbitração de mensagens.

2.1.1 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[2] (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

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 não são detectados. Cada dispositivo da rede deve ser capaz de identificar a ocorrência destes erros, e informar aos 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. Cada dispositivo na rede CAN pode ser levado para os seguintes estados, de acordo com a quantidade de erros de transmissão ou recepção detectados:

- *Error Active:* os contadores internos de erro estão em um nível baixo e o dispositivo opera normalmente na rede CAN. Pode enviar e receber telegramas e atuar na rede CAN caso detecte algum erro na transmissão de telegramas.
- *Warning:* quando algum destes contadores passa de um determinado limite, o dispositivo entra no estado de *warning*, significando a ocorrência de uma elevada taxa de erros de comunicação.
- *Error Passive:* quando este valor ultrapassa um limite maior, ele entra no estado de *error passive*, onde ele para de atuar na rede ao detectar que outro dispositivo enviou um telegrama com erro.

[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* (11 bits) e *extended* (29 bits). Para esta implementação, somente frames *standard* são aceitos.

- *Bus Off:* por último, temos o estado de *bus off*, no qual o dispositivo não irá mais enviar ou receber telegramas. O dispositivo opera como se estivesse desconectado da rede.

2.1.5 CAN e CANopen

Somente a definição de como detectar erros, criar e transmitir um frame não são suficientes 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 protocolos baseados em CAN, como DeviceNet, CANopen, J1939, etc., que 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.

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.

Para a remota RUW06, o endereço do escravo é programado através das chaves hexadecimais S1 e S2.

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.

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.

Identificador	6 bytes de dados					
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. Os seguintes objetos de comunicação (COBs) são descritos pela especificação:

Tabela 2.1: *Tipos de 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.
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)	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.).
Synchronization Object (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.
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. Dentre os serviços descritos pelo protocolo estão: serviços de controle do dispositivo, onde o mestre controla o estado de cada escravo na rede, e serviços de controle de erros (Node Guarding e Heartbeat), onde o dispositivo envia mensagens periódicas para informar que a conexão está ativa.

Toda a comunicação do escravo com a rede é feita utilizando estes objetos, e os dados que podem ser acessados são os existentes no dicionário de objetos do dispositivo.

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 descritos no protocolo. 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.

Tabela 2.2: *COB-ID para os diferentes objetos*

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)

2.9 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 é fornecido em um CD juntamente com o produto, e também pode ser obtido através do site http://www.weg.net. É necessário observar a versão de software do equipamento, para utilizar um arquivo EDS que seja compatível com esta versão.

3 INTERFACE DE COMUNICAÇÃO CANOPEN

A remota RUW06 apresenta diagnóstico através de Leds que indicam o estado da comunicação.

3.1 CARACTERÍSTICAS DA INTERFACE CAN

- Interface isolada galvanicamente e com sinal diferencial, conferindo maior robustez contra interferência eletromagnética.
- Alimentação externa de 24 V.
- Permite a conexão de até 64 dispositivos no mesmo segmento. Uma quantidade maior de dispositivos pode ser conectada e com o uso de repetidores[3].
- Comprimento máximo do barramento de 1000 metros.

3.2 PINAGEM DO CONECTOR

A interface CAN possui um conector *plug-in* de 5 vias (X4) com a seguinte pinagem:

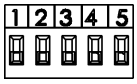


Tabela 3.1: *Pinagem do conector X4 para interface CAN*

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

3.3 FONTE DE ALIMENTAÇÃO

As interfaces 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.

Tabela 3.2: *Características da alimentação para interface CAN*

Tensão de alimentação (VCC)		
Mínimo	Máximo	Recomendado
11	30	24
Corrente (mA)		
Típico		Máximo
30		50

3.4 INDICAÇÕES

A comunicação apresenta dois Leds: CAN (NET 1) e ERROR (NET 2).

Led CAN (NET 1): sinalização do estado do protocolo CANopen;

Led ERROR (NET 2): sinalização do estado da interface CAN;

As tabelas abaixo mostram o comportamento destes leds em função do estado da unidade remota:

Tabela 3.3: *sinalização do estado de operação da unidade remota*

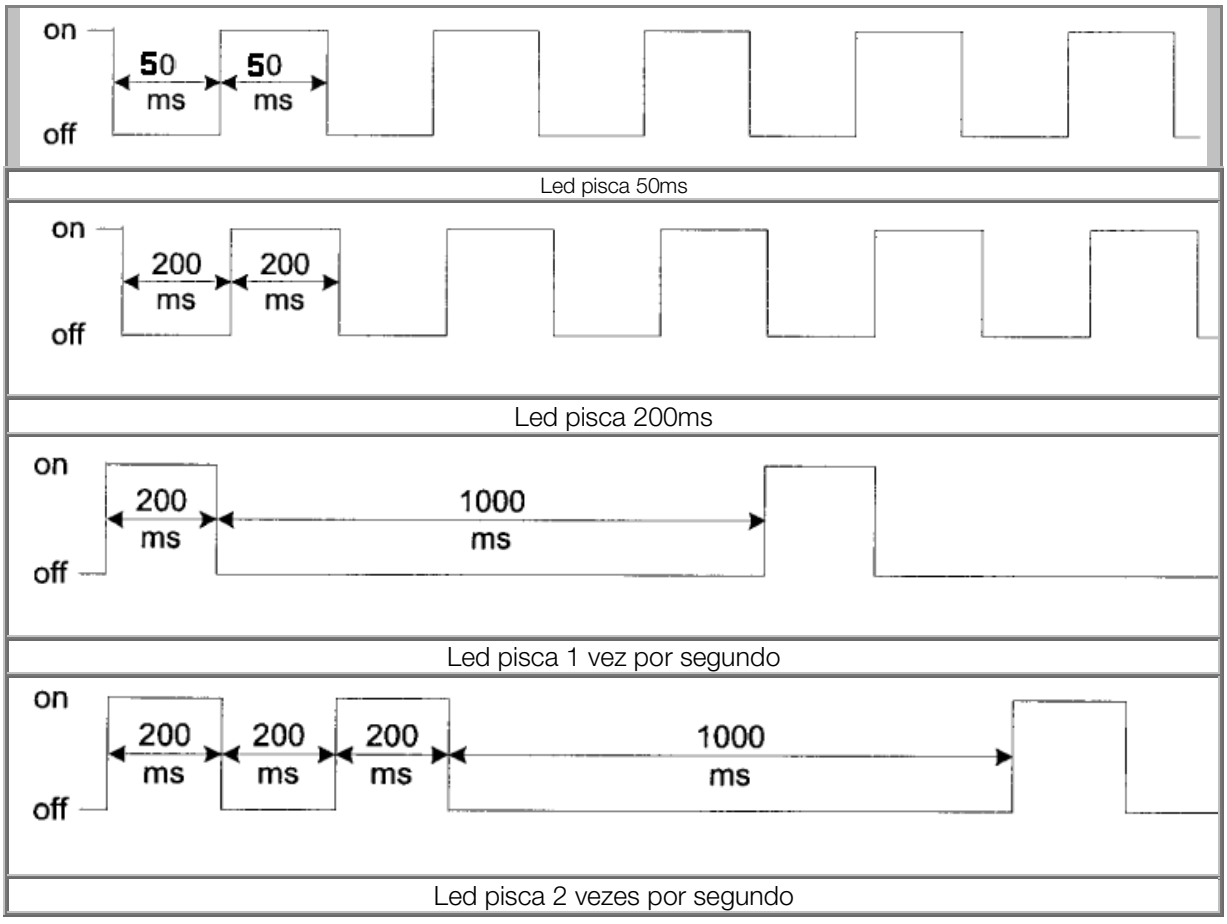
Estado	LED CAN	Descrição
STOPPED	Verde pisca uma vez por segundo	Dispositivo está no estado parado, de acordo com a especificação do protocolo CANopen.
PRE-OPERATIONAL	Verde pisca 200ms	Dispositivo está no estado Pré-operacional. PDOSs não estão disponiveis para comunicação.
OPERATIONAL	Verde	Dispositivo está no estado Operacional. Todos os objetos de comunicação rstão disponíveis.

[3] O número limite de equipamentos que podem ser conectados na rede também depende do protocolo utilizado.

Tabela 3.4: *sinalização de erro da unidade remota*

Estado	LED ERROR	Descrição
Sem ERRO	Apagado	Dispositivo está em condições normais de operação, on-line, e com conexões estabelecidas.
Warning	Vermelho pisca 1 vez por segundo	Indica que o controlador CAN está em estado de Warning ou Error Passive (ver item 6.1.4). Pode ocorrer, por exemplo, se for o único equipamento ligado à rede CANopen.
Erro de Node Guarding ou Heartbeat	Vermelho pisca 2 vez por segundo	Controle de erros da comunicação CANopen detectou erro de comunicação utilizando o mecanismo de guarding ou heartbeat.
Erro de BUS OFF	Vermelho	Indica que o controlador CAN está no stado de BUS OFF (ver item 6.1.4). Indica uma condição crítica de operação na rede CANopen, em geral associada a problemas na instalação ou configuração incorreta da taxa de comunicação. É necessário desligar e ligar novamente a RUW01.
Erro de BUS POWER OFF	Vermelho	Indica que a interface CAN não possui alimentação entre os pinos 1 e 5 do conector.
CANopen não Inicializado	Vermelho pisca 50ms	Indica que o Protocolo CANopen não foi inicializado. Verificar se o endereço está ajustado em um valor válido (01h – 7Fh).

Tabela 3.5: *Diagrama de tempo para o comportamento da sinalização*



4 INSTALAÇÃO DA REDE CANOPEN

A rede CANopen, como várias redes de comunicação industriais, pelo fato de ser aplicada muitas vezes em ambientes agressivos e com alta exposição a interferência eletromagnética, exige certos cuidados que devem ser tomados para garantir uma baixa taxa de erros de comunicação durante a sua operação. A seguir são apresentadas recomendações para realizar a instalação do produto na rede.

4.1 TAXA DE COMUNICAÇÃO

Equipamentos com interface CANopen em geral permitem configurar a taxa de comunicação desejada, podendo variar de 10Kbit/s até 1Mbit/s. A taxa de comunicação (*baud rate*) que pode ser utilizada por um equipamento também depende do comprimento do cabo utilizado na instalação. A tabela a seguir apresenta a relação entre as taxas de comunicação e o comprimento máximo de cabo que pode ser utilizado na instalação, de acordo com o recomendado pela CiA[4].

Tabela 4.1: *Taxas de comunicação suportadas e comprimento do cabo*

Taxa de comunicação	Comprimento do cabo
1 Mbit/s	25 m
800 Kbit/s	50 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

Todos os equipamentos da rede devem ser programados para utilizar a mesma taxa de comunicação. Para a remota RUW06, a taxa de comunicação CAN é programada através da chaves hexadecimal S3.

4.2 ENDEREÇO NA REDE CANOPEN

Todo dispositivo na rede CANopen deve possuir um endereço, ou Node ID, entre 1 e 127. Este endereço precisa ser diferente para cada equipamento. Para a remota RUW06, o endereço do equipamento é programado através das chaves hexadecimais S1 e S2.

4.3 RESISTORES DE TERMINAÇÃO

A utilização de resistores de terminação nas extremidades do barramento CAN é fundamental para evitar reflexão de linha, que pode prejudicar o sinal transmitido e ocasionar erros na comunicação. Resistores de terminação no valor de 120 Ω / 0.25 W devem ser conectados entre os sinais CAN_H e CAN_L nas extremidades do barramento principal.

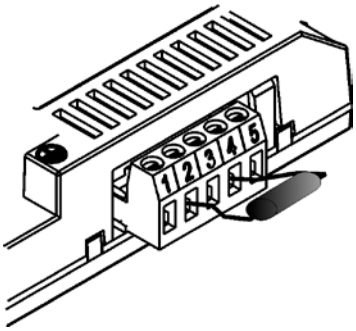


Figura 4.1: *Exemplo de instalação do resistor de terminação*

[4] Diferentes produtos podem apresentar variações no comprimento máximo do cabo suportado para a instalação.

4.4 CABO

Para a ligação dos sinais CAN_L e CAN_H deve-se utilizar par trançado com blindagem. A tabela a seguir apresenta as características recomendadas para o cabo.

Tabela 4.2: *Características do cabo para rede CANopen*

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

A utilização de um par trançado adicional para levar a alimentação de 24Vcc para os equipamentos que necessitam deste sinal também é recomendada.

4.5 LIGAÇÃO NA 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 próxima a cabos de potência, pois isto facilita a ocorrência de erros durante a transmissão devido à interferência eletromagnética. 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.

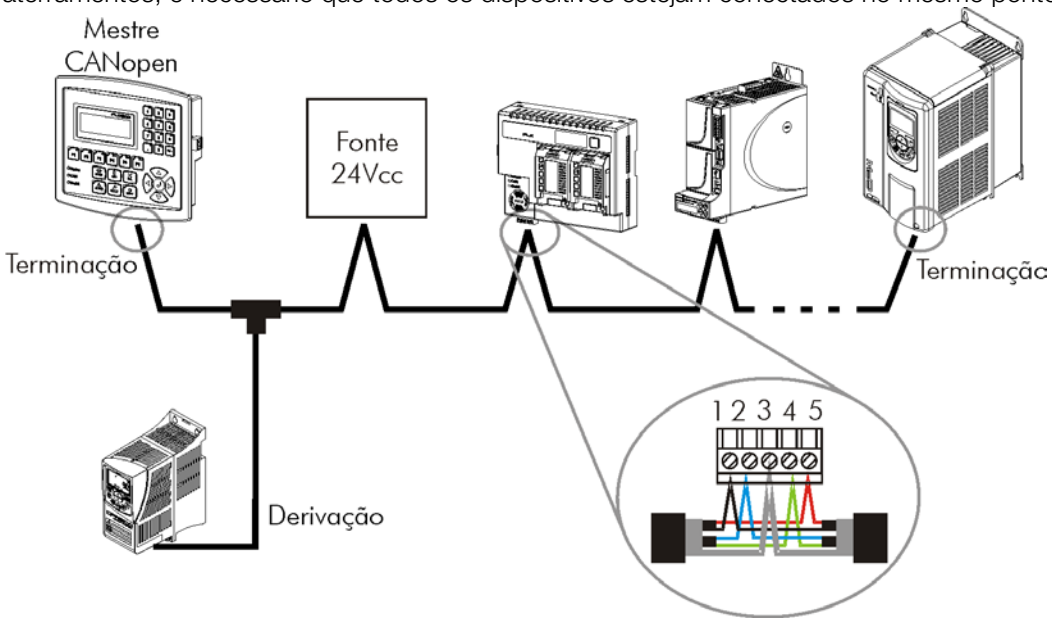


Figura 4.2: *Exemplo de instalação em rede CANopen*

Para evitar problemas de diferença de tensão na alimentação 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.

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.

5 CONFIGURAÇÃO

A configuração da remota RUW06 é feita através de chaves hexadecimal com as seguintes funções:

S1 e S2: Configuração do endereço na rede CANopen;
S3: Seleção da taxa de comunicação.

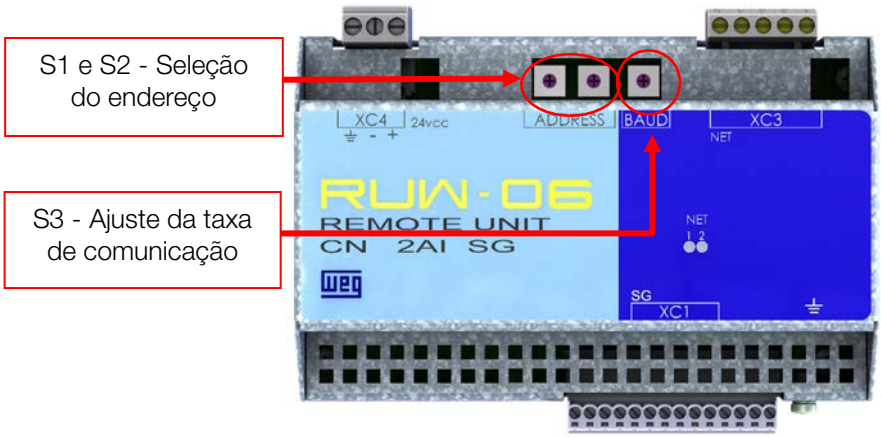


Figura 5.1: *Chaves hexadecimal para configuração da RUW06*

5.1 ENDEREÇO REDE CANOPEN

O endereço da unidade remota é configurado através de duas chaves hexadecimal S1 e S2 como ilustra a Figura 5.1. Este endereço é utilizado para identificar a remota na rede CANopen. É necessário que cada equipamento da rede possua um endereço diferente dos demais.

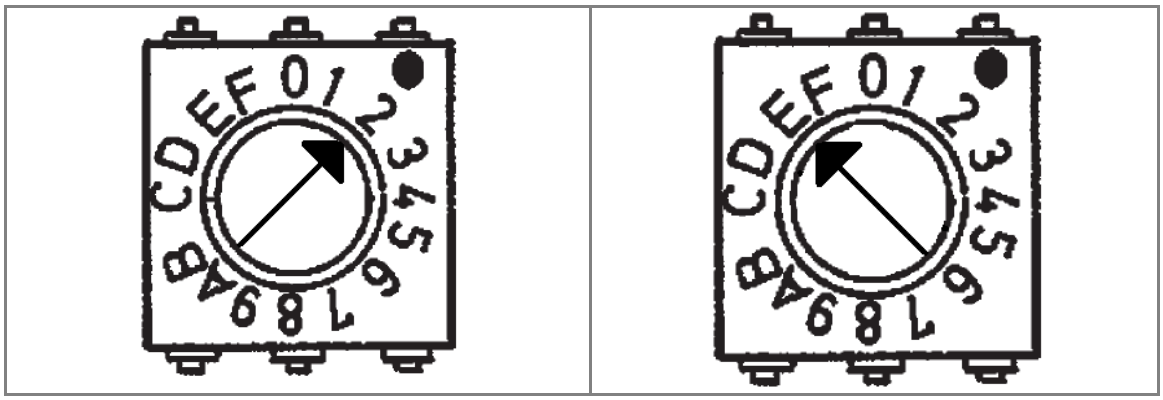
Endereços válidos: 1 a 127 (01h a 7Fh).

NOTA!
Caso o endereço seja alterado, ele somente será válido após a remota RUW06 ser desligada e ligada novamente.

O endereço selecionado através das chaves representa um valor hexadecimal. Portando, o mesmo deve ser convertido para decimal, quando necessário, na parametrização do mestre CANopen. As chaves que configuram o endereço CAN da unidade remota são ilustradas com mais detalhes na Figura 5.1, onde a chave S1 é a mais significativa e a chave S2 é a menos significativa.
Os valores das chaves conforme a Figura 5.1.

Chave S1 = 02h;
Chave S2 = 0Eh.

Como a chave S1 é a mais significativa o endereço formado pelas chaves é 2Eh, ou 46 decimal.



Chave hexadecimal S1 – mais significativa ***Chave hexadecimal S2 – menos significativa***

Figura 5.2: *Chaves hexadecimal para configuração da RUW06*

5.2 TAXA DE COMUNICAÇÃO – BAUD RATE

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. A remota RUW06 apresenta uma chave hexadecimal para a seleção da taxa de comunicação, como ilustra a Figura 5.2. A unidade remota apresenta as seguintes taxas de comunicação:

0 = 1 Mbit/s
1 = 800 Kbit/s
2 = 500 Kbit/s
3 = 250 Kbit/s
4 = 125 Kbit/s
5 = 100 Kbit/s
6 = 50 Kbit/s
7 = 20 Kbit/s

NOTA!
Os demais valores selecionados na chave apresentam taxa de comunicação de 1 Mbit/s.

NOTA!
Caso a taxa de comunicação seja alterada, ela somente será válida após a RUW06 ser desligada e ligada novamente.

5.3 CONFIGURADOR WPSCAN

Na configuração do escravo no WPSCAN esta disponível na aba SETUP uma interface para configurar os principais objetos para a unidade remota, conforme pode ser visto na Figura 5.3. A descrição completa de cada objeto é feita no capítulo 6.

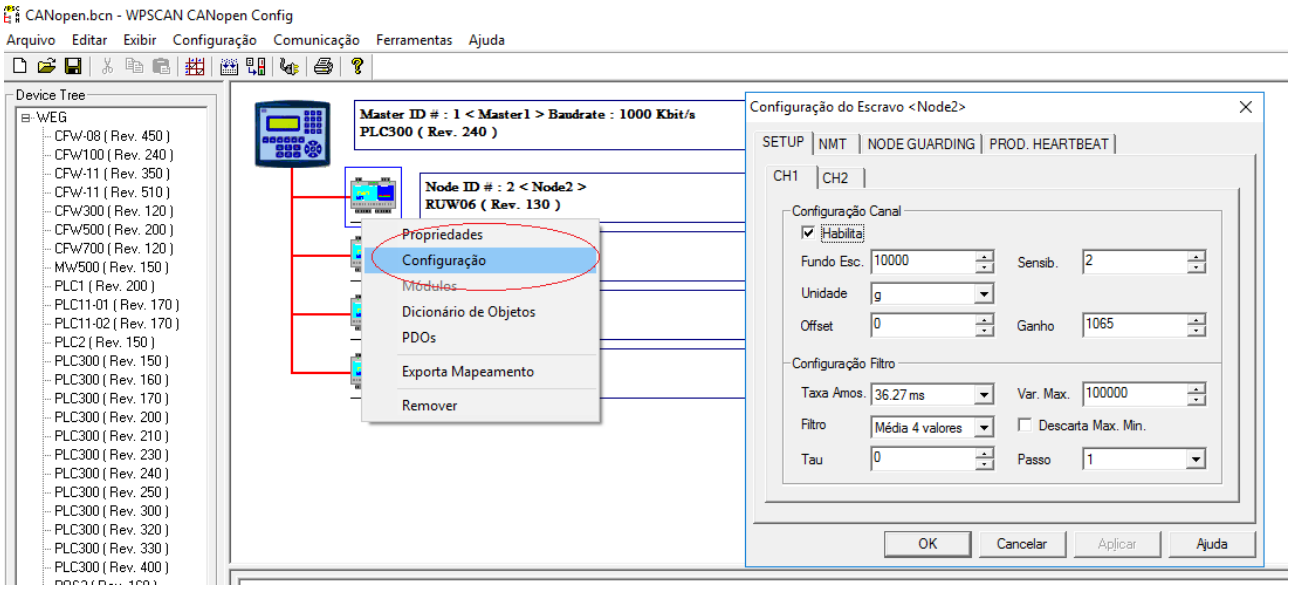


Figura 5.3: *Configurador WPSCAN para RUW06*

6 DICIONÁRIO DE OBJETOS

O dicionário de objetos é uma lista com os diversos dados do equipamento que são acessíveis através da 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:

Tabela 6.1: *Agrupamentos do dicionário de objetos*

Índice	Objetos	Descrição
0001h – 025Fh	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 representarão.
6000h – 9FFFh	Objetos padronizados para dispositivos	Esta faixa é reservada para objetos que descrevem o comportamento de equipamentos similares, independente do fabricante.

Demais índices não referenciados nesta lista são reservados.

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 – 025Fh).
- *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 – 025Fh) 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.

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, mas somente alguns são obrigatórios. A seguir é apresentada uma lista com alguns dos objetos desta faixa suportados pela remota RUW06.

Tabela 6.2: *Lista de objetos – Communication Profile*

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
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
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

Estes objetos somente podem ser lidos e escritos através da rede CANopen, não estão disponíveis via HMI ou outra interface de rede. O mestre da rede, em geral, é o equipamento responsável pela configuração do equipamento antes de iniciar a operação. O arquivo de configuração EDS traz a lista de todos os objetos de comunicação suportados.

Para uma descrição detalhada de quais objetos estão disponíveis nesta faixa do dicionário de objetos, consulte o item 7.

6.4 MANUFACTURER SPECIFIC – OBJETOS ESPECÍFICOS DO FABRICANTE

Nos índices de 2000h até 5FFFh, cada fabricante é livre para definir quais objetos estarão presentes, o tipo e a função de cada objeto. Para o drive, nesta faixa de objetos foi disponibilizada toda a lista de parâmetros. Através destes parâmetros é possível operar o equipamento, executando qualquer função que o drive 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. A tabela a seguir ilustra o dicionário de objetos.

Tabela 6.3: *Lista de objetos – Manufacturer Specific*

Índice	Objeto	Nome	Tipo	Acesso
2001h	VAR	Read Node Address	UNSIGNED8	ro
2002h	VAR	Read Baud rate	UNSIGNED16	ro
2003h	VAR	Bus Off AutoReset	UNSIGNED8	rw
2004h	ARRAY	Analogue Channel Status	UNSIGNED8	ro
2005h	ARRAY	Read Analogue Input 24bits	UNSIGNED32	ro
2006h	ARRAY	Read Analogue Input 24bits process	INTEGER32	ro
2007h	ARRAY	Read Analogue Input 16MSB process	INTEGER16	ro
2008h	ARRAY	Analogue Channel Enable	USIGNED8	rw
200Ah	ARRAY	Analogue Channel Unit	USIGNED8	rw
200Ch	ARRAY	Analogue Channel Filter	UNSIGNED16	rw
200Dh	ARRAY	Analogue Channel Gain	INTEGER16	rw
200Eh	ARRAY	Analogue Channel Offset	INTEGER32	rw
200Fh	ARRAY	Strain Gage Full Scale	INTEGER16	rw
2010h	ARRAY	Strain Gage Sensibility	UNSIGNED8	rw
2011h	ARRAY	Aquisition Sample Rate	UNSIGNED8	rw
2012h	ARRAY	Sample Maximum Variation	UNSIGNED32	rw
2013h	ARRAY	Discard Value	UNSIGNED8	rw
2014h	ARRAY	Filter TAU Const	UNSIGNED16	rw
2015h	ARRAY	Variation Step	UNSIGNED8	rw

6.4.1 Objeto 2003h – Bus Off autoreset

Através deste objeto é possível desabilitar o autoreset para o erro de Bus Off. O valor padrão do objeto é 0, autoreset habilitado.

Índice	2003h
Nome	BUS OFF autoreset
Objeto	VAR
Tipo	SDO Parameter

Acesso	rw
Mapeável	Não
Faixa	UNSIGNED8
Valor Padrão	0

0 -> autoreset do BUS OFF
1 -> sem auto reset de BUS OFF

6.4.2 Objeto 2004h – Analogue Channel Status

Através deste objeto é possível verificar o estado do canal, sendo:

0 -> desabilitado
1 -> habilitado

Índice	2004h
Nome	Analogue Channel Status
Objeto	ARRAY
Tipo	UNSIGNED8

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1-2
Valor Padrão	2

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	ro
Mapeável	Sim
Faixa	UNSIGNED8
Valor Padrão	Não

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	ro
Mapeável	Sim
Faixa	UNSIGNED8
Valor Padrão	Não

6.4.3 Objeto 2005h – Read Analogue Input 24bits

Objeto com os valores dos canais analógicos lidos diretamente dos conversores A/D conectados à cada célula de carga. O valor lido de cada conversor é apresentado com uma resolução de 24 bits com sinal, e sua faixa poderá ser -8.388.608 (FF800000h) a 8.388.607 (007FFFFFh).

- O valor 0 (zero) representa o valor lido sem carga (a célula de carga fornece a tensão de referência do A/D).
- O valor 007FFFFFh é o valor lido caso se a célula de fornecer a tensão máxima do A/D.
- O valor FF800000h é o valor lido caso se a célula de fornecer a tensão mínima do A/D.

A faixa efetiva deste objeto vai depender da célula de carga utilizada, que normalmente varia poucos milivolts com carga máxima.

Índice	2005h
Nome	Read Analogue Input 24Bits
Objeto	ARRAY
Tipo	UNSIGNED32

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	Ro
Mapeável	Não
Faixa	1 – 2h
Valor Padrão	Não

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	ro
Mapeável	Sim
Faixa	UNSIGNED32
Valor Padrão	Não

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	Ro
Mapeável	Sim
Faixa	UNSIGNED32
Valor Padrão	Não

6.4.4 Objeto 2006h – Read Analogue Input 24bits process

Objeto com os valores dos canais analógicos, similar ao objeto 2005h, mas utilizando o filtro programado no objeto 200Ch.

Este valor deverá ser multiplicado pela sensibilidade (objeto 2010h) e dividido pelo fundo de escala (objeto 200Fh) da célula de carga, para obter o valor efetivo da carga. Pode ser necessário ainda ajustar um ganho (objeto 200Dh) e um offset (objeto 200Eh) para obter os valores desejados. O valor com estes cálculos realizados é apresentado no objeto 6401h.

Índice	2006h
Nome	Read Analogue Input 24 Bits process
Objeto	ARRAY
Tipo	INTEGER32

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	Ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	Não

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	ro
Mapeável	Sim
Faixa	INTEGER32
Valor Padrão	Não

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	Ro
Mapeável	Sim
Faixa	INTEGER32
Valor Padrão	Não

6.4.5 Objeto 2007h – Read Analogue Input 16 MSB process

Objeto de 16 bits com sinal, representando o valor do objeto 2006h, desconsiderando-se os 8 bits menos significativos da resolução do A/D.

Índice	2007h
Nome	Read Analogue Input 16MSB process
Objeto	ARRAY
Tipo	INTEGER16

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	Ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	Não

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	ro
Mapeável	Sim
Faixa	INTEGER16
Valor Padrão	Não

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	ro
Mapeável	Sim
Faixa	INTEGER16
Valor Padrão	Não

6.4.6 Objeto 2008h – Analogue Channel Enable

Habilitação da leitura do canal correspondente. Caso o canal esteja desabilitado, os objetos de leitura estarão com o valor zero.

Índice	2008h
Nome	Analogue Channel Enable
Objeto	ARRAY
Tipo	USIGNED8

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	rw
Mapeável	Não
Faixa	1 – 2
Valor Padrão	2

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	rw
Mapeável	não
Faixa	USIGNED8
Valor Padrão	1

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	rw
Mapeável	não
Faixa	USIGNED8
Valor Padrão	1

6.4.7 Objeto 200Ah – Analogue Channel Unit

Configura a unidade do sinal processado, sendo:

0 – g (grama)
1 – kg (quilograma)
2 – t (tonelada)

Nota: A configuração desse objeto não é necessária, sendo que o mesmo não é utilizado para apresentação dos valores lidos.

Índice	200Ah
Nome	Analogue Channel Unit
Objeto	ARRAY
Tipo	USIGNED8

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	2

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	rw
Mapeável	não
Faixa	USIGNED8
Valor Padrão	0

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	rw
Mapeável	não
Faixa	USIGNED8
Valor Padrão	0

6.4.8 Objeto 200Ch – Analogue Channel Filter

Filtro de média móvel dos últimos valores lidos, de acordo com o configurado no objeto.

0 – sem filtro
1 – média dos últimos 2 valores
2 – média dos últimos 4 valores
3 – média dos últimos 8 valores
4 – média dos últimos 16 valores
5 – média dos últimos 32 valores

Este filtro de média móvel armazena os últimos X valores lidos (2, 4, 8,16 ou 32) e faz a média deles. Na próxima amostra o primeiro valor armazenado no buffer é descartado, o novo valor adicionado ao final do buffer e a média é novamente calculada.
A cada amostra, os valores máximo e mínimo podem ser descartados para o cálculo da média, conforme objeto 2013h

Índice	200Ch
Nome	Analogue Channel Filter
Objeto	ARRAY
Tipo	UNSIGNED16

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	2

Sub-índice	1
Descrição	Filtro da entrada analógica 1
Acesso	Rw
Mapeável	não
Faixa	UNSIGNED8
Valor Padrão	2

Sub-índice	2
Descrição	Filtro da entrada analógica 2
Acesso	rw
Mapeável	não
Faixa	UNSIGNED8
Valor Padrão	2

6.4.9 Objeto 200Dh – Analogue Channel Gain

Ganho aplicado ao sinal processado após a adição do offset. O ganho deve ser multiplicado por 1000, ou seja, para um ganho 1, o objeto deve receber o valor 1000, para um ganho 0,5, o objeto deve receber 500.

Índice	200Dh
Nome	Analogue Channel Gain
Objeto	ARRAY
Tipo	INTEGER16

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	2

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	rw
Mapeável	não
Faixa	INTEGER16
Valor Padrão	1000

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	rw
Mapeável	não
Faixa	INTEGER16
Valor Padrão	1000

6.4.10 Objeto 200Eh – Analogue Channel Offset

Valor de offset a ser somado no valor processado, podendo ser positivo ou negativo. O offset está na unidade configurada no objeto 200Ah e de acordo com o fundo de escala.

Índice	200Eh
Nome	Analogue Channel Offset
Objeto	ARRAY
Tipo	INTEGER32

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	2

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	rw
Mapeável	não
Faixa	INTEGER32
Valor Padrão	0

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	rw
Mapeável	não
Faixa	INTEGER32
Valor Padrão	0

Ajuste de offset e ganho: o sinal de entrada é somado ao offset (na unidade do usuário: g, kg ou t). O valor resultante é multiplicado pelo Ganho.

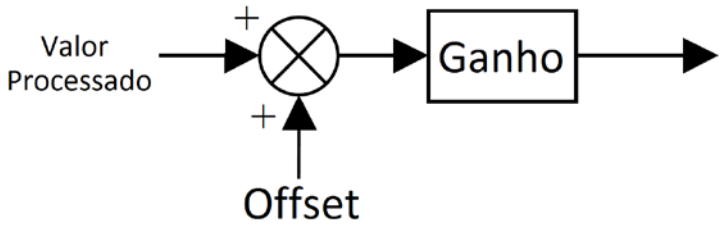


Figura 6.1: *Diagrama da entrada analógica.*

Para efetuar o ajuste inicial, deixe o módulo sem carga, corrija o offset e transfira a configuração. Na sequência, após verificar que o offset foi corrigido (balança indicando zero), coloque um peso conhecido, recomenda-se pelo menos 70% do fundo de escala. O ganho é dado pela equação:

$$Ganho = \frac{Valor\ Conhecido}{Valor\ Medido} \times 1000$$

6.4.11 Objeto 200Fh – Strain Gage Full Scale

Objeto com o fundo de escala da célula de carga (carga máxima). Por exemplo, para uma célula de carga de até 10 Kg, configurando o fundo de escala com o valor 10.000 (10.000 gramas), o objeto com o valor lido (6401h) terá o valor da carga em gramas.

Índice	200Fh
Nome	Strain Gage Full Scale
Objeto	ARRAY
Tipo	INTEGER16

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	2

Sub-índice	1
Descrição	Strain Gage 1
Acesso	rw
Mapeável	não
Faixa	INTEGER16
Valor Padrão	10000

Sub-índice	2
Descrição	Strain Gage 2
Acesso	rw
Mapeável	não
Faixa	INTEGER16
Valor Padrão	10000

6.4.12 Objeto 2010h – Strain Gage Sensibility

Objeto com a sensibilidade da célula de carga em mV/V.

Índice	2010h
Nome	Strain Gage Sensibility
Objeto	ARRAY
Tipo	UNSIGNED8

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	Não

Sub-índice	1
Descrição	Strain Gage 1
Acesso	rw
Mapeável	não
Faixa	UNSIGNED8
Valor Padrão	1

Sub-índice	2
Descrição	Strain Gage 2
Acesso	rw
Mapeável	não
Faixa	UNSIGNED8
Valor Padrão	1

6.4.13 Objeto 2011h – Aquisition Sample Rate

Configura a taxa de amostragem de cada canal conforme o valor do objeto:
0 – 1,68 SPS* (596,12 ms)
1 – 3,35 SPS (298,06 ms)
2 – 6,71 SPS (149,03 ms)
3 – 13,42 SPS (74,52 ms)
4 – 26,83 SPS (36,27 ms)
5 – 53,66 SPS (18,64 ms)
6 – 107,32 SPS (9,32 ms)

SPS: amostras por segundo (samples per second).

OBS: Se ambos os canais estiveram habilitados o tempo de amostragem será a soma dos tempos dos dois canais.
Exemplo: para o valor default, a leitura do canal será feita a cada 36,27ms quando apenas um deles estiver habilitado. Se ambos os canais forem habilitados a leitura será feita a cada 72,54ms.

Índice	2011h
Nome	Acquisition Sample Rate
Objeto	ARRAY
Tipo	UNSIGNED8

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	Não

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	rw
Mapeável	não
Faixa	0 - 6
Valor Padrão	4

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	rw
Mapeável	não
Faixa	0 - 6
Valor Padrão	4

6.4.14 Objeto 2012h – Sample Maximum Variation

Máxima variação permitida da leitura atual com relação à leitura anterior. Objeto na unidade de medida configurada.
Exemplo: pode ser configurado para evitar variações bruscas na leitura devido a cargas móveis, etc.
Quanto menor o valor, mais tempo o sistema levará para estabilizar.

Índice	2012h
Nome	Sample Maximum Variation
Objeto	ARRAY
Tipo	UNSIGNED32

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	Não

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	Rw
Mapeável	Não
Faixa	0 – 100000
Valor Padrão	100000

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	Rw
Mapeável	não
Faixa	0 - 100000
Valor Padrão	100000

6.4.15 Objeto 2013h – Discard Value

Possibilita descartar os valores máximo e mínimo do buffer da média móvel configurada no filtro do objeto 200Ch, eliminando possíveis variações indesejadas.

0 – Mantem máximo e mínimo
1 – Descarta o valor máximo e o mínimo

Exemplo: Caso configurado para descartar, o buffer é percorrido por inteiro a cada nova amostra, o valor máximo e o mínimo são descartados e a média é feita com os valores restantes.

Índice	2013h
Nome	Discard Value
Objeto	ARRAY
Tipo	UNSIGNED8

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	Não

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	Rw
Mapeável	Não
Faixa	0 – 1
Valor Padrão	0

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	Rw
Mapeável	não
Faixa	0 – 1
Valor Padrão	0

6.4.16 Objeto 2014h – Filter TAU Constant

Constante de tempo em milissegundos do filtro passa baixas de primeira ordem.

Índice	2014h
Nome	Filter TAU Constant
Objeto	ARRAY
Tipo	UNSIGNED16

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	Não

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	Rw
Mapeável	Não
Faixa	0 – 10000
Valor Padrão	0

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	Rw
Mapeável	não
Faixa	0 – 10000
Valor Padrão	0

6.4.17 Objeto 2015h – Variation Step

Passo da variação do último digito transmitido no objeto 6401h

0 – passo 1 (000, 001, 002, 003...)
1 – passo 2: (000, 002, 004, 006...)
2 – passo 5: (000, 005, 010, 015...)
3 – passo 10: (000, 010, 020, 030...)
5 – passo 50: (000, 050, 100, 150...)

Índice	2015h
Nome	Variation Step
Objeto	ARRAY
Tipo	UNSIGNED8

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	Não

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	Rw
Mapeável	Não
Faixa	0 – 5
Valor Padrão	0

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	Rw
Mapeável	Não
Faixa	0 – 5
Valor Padrão	0

O fluxograma abaixo mostra a sequência de execução da RUW06. Os blocos amarelos representam objetos configuráveis e os verdes objetos referentes à pesagem que podem ser lidos pelo usuário.

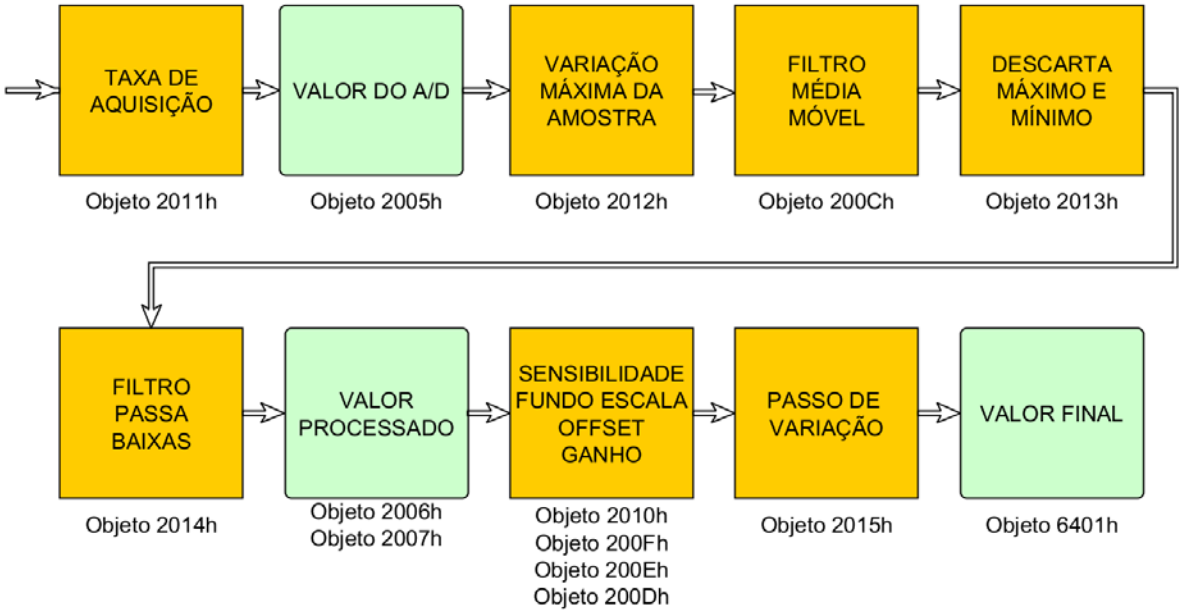


Figura 6.2: *Fluxograma da sequencia de execução da medida.*

6.5 DEVICE PROFILE – OBJETOS COMUNS PARA MODULOS I/O

A documentação CANopen também inclui propostas para padronização de determinados tipos de dispositivos. A remota RUW06 segue o descrito pela *CiA DPS 401 – Device Profile for Generic I/O Modules*. Este documento descreve um conjunto de objetos que devem ser comuns para módulos de I/O, independente do fabricante. Isto facilita a interoperabilidade entre dispositivos com a mesma função, pois tanto os dados quanto o comportamento do dispositivo são disponibilizados de uma forma padronizada.

Para estes objetos foram reservados os índices de 6000h até 9FFFh. É possível operar o inversor através da rede CANopen, tanto através dos parâmetros (localizados a partir do índice 2000h) quanto através destes objetos padronizados.

Para uma descrição detalhada de quais objetos estão disponíveis nesta faixa do dicionário de objetos, consulte o item 8.

7 DESCRIÇÃO DOS OBJETOS DE COMUNICAÇÃO

Neste item são descritos detalhadamente cada um dos objetos de comunicação disponíveis para a remota RUW06. É necessário conhecer como estes objetos são operados para utilizar as funções disponíveis para a comunicação do drive.

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	0004.0191h.

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.

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 equipamento segue o descrito pela tabela a seguir.

Índice	1001h
Nome	Error register
Objeto	VAR
Tipo	UNSIGNED8

Acesso	ro
Mapeável	Sim
Faixa	UNSIGNED8
Valor Padrão	0

Tabela 7.1: *Estrutura do objeto Error Register*

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

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 objetct
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	0000.0C00h

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

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 remota RUW06 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.

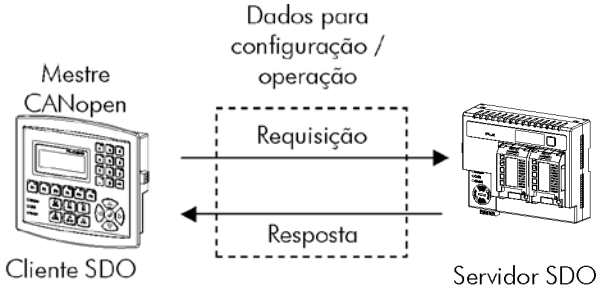


Figura 7.1: *Comunicação entre cliente e servidor SDO*

7.2.1 Objeto 1200h – Servidor SDO

A remota RUW06 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 drive. Todo o servidor SDO possui um objeto, do tipo SDO_PARAMETER, 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							
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:

Tabela 7.2: *Código dos comandos para cliente SDO*

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

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, mas os comandos serão diferentes:

Tabela 7.3: *Código dos comandos para servidor SDO*

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

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.

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.

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. É necessário, porém, 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.

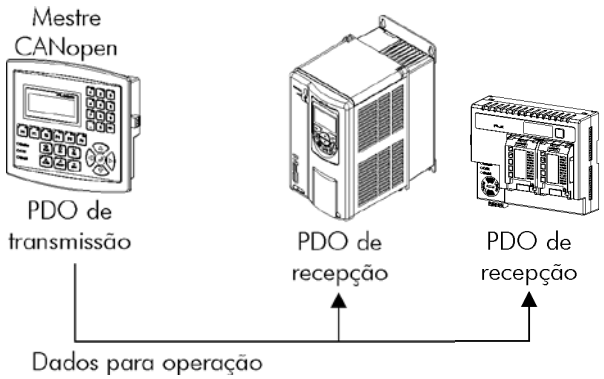


Figura 7.2: *Comunicação utilizando PDOs*

NOTA!
PDOs somente podem ser transmitidos ou recebidos quando o dispositivo está no estado operacional.

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 esta condição. 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.

O arquivo EDS do equipamento traz a lista de todos os objetos disponíveis, informando se o objeto é mapeável ou não.

7.3.2 PDOs de Recepção

A remota RUW06 não possui PDOs de recepção.

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. A remota RUW06 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 até 1803h
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

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 Erro! Fonte de referência não encontrada..

O sub-índice 2 indica o tipo de transmissão deste objeto, que segue o descrito pela Erro! Fonte de referência não encontrada.. 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 na remota RUW06 é 2ms.
- É importante observar o tempo entre transmissões programado no sub-índice 3 principalmente 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 até 1A03h
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	0

Sub-índice	1 até
Descrição	1º até 4º objeto mapeado no PDO
Acesso	rw
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	0

O PDO MAPPING para a transmissão funciona de forma semelhante ao de 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, supondo um mapeamento para um PDO de recepção com os seguintes valores configurados, temos:

- *Sub-índice 0 = 2:* este PDO possui dois objetos mapeados.
- *Sub-índice 1 = 6401.0110h:* o primeiro objeto mapeado possui índice igual a 6401h, sub-índice 1 (um), e tamanho igual a 16 bits. Este objeto corresponde a leitura da entrada analógica 1.
- *Sub-índice 2 = 6401.0210h:* o segundo objeto mapeado possui índice igual a 6401h, sub-índice 2 (dois), e tamanho igual a 16 bits. Este objeto corresponde a leitura da entrada analógica 2.

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 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 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.

A remota RUW06 possui a função de consumidor SYNC e, portanto, pode programar seus PDOs para serem síncronos. PDOs síncronos são aqueles relacionados com o objeto de sincronização e, portanto, podem ser programados para serem transmitidos ou atualizados com base neste objeto.

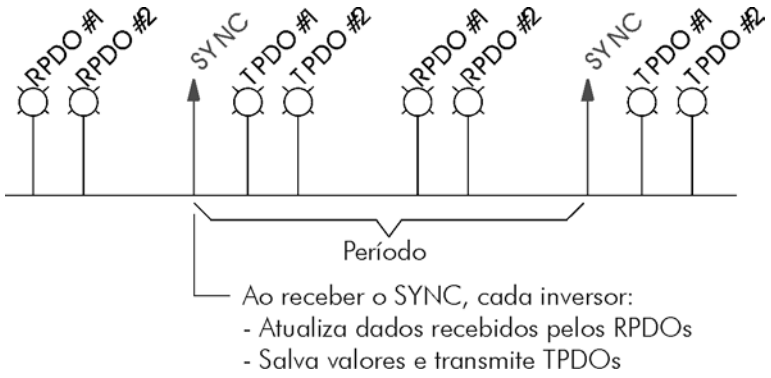


Figura 7.3: *SYNC*

A mensagem SYNC transmitida pelo produtor não possui dado algum em seu campo de dados, pois seu objetivo é fornecer um evento sincronizado entre os dispositivos da rede. O seguinte objeto está disponível para configuração 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.

7.5 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 este objeto estão disponíveis os serviços de controle do nó e de controle de erros (utilizando *Node Guarding* ou *Heartbeat*).

7.5.1 Controle dos Estados do Escravo

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

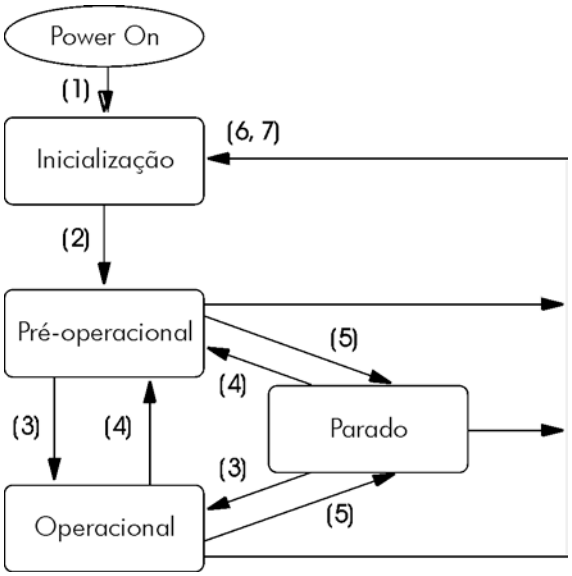


Figura 7.4: *Diagrama de estados do nó CANopen*

Tabela 7.4: *Descrição das transições*

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

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. Os PDOs, porém, 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.

Tabela 7.5: *Objetos acessíveis em cada estado*

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

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

Tabela 7.6: *Comandos para a transição de estados*

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

As transições indicadas no código do comando equivalem às transições de estado executadas pelo nó após receber o comando (conforme figura 7.4). O comando *Reset node* faz com que o escravo execute um reset completo do dispositivo, enquanto que o comando *Reset communication* faz com que o escravo reinicialize apenas os objetos relativos à comunicação CANopen.

7.5.2 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.

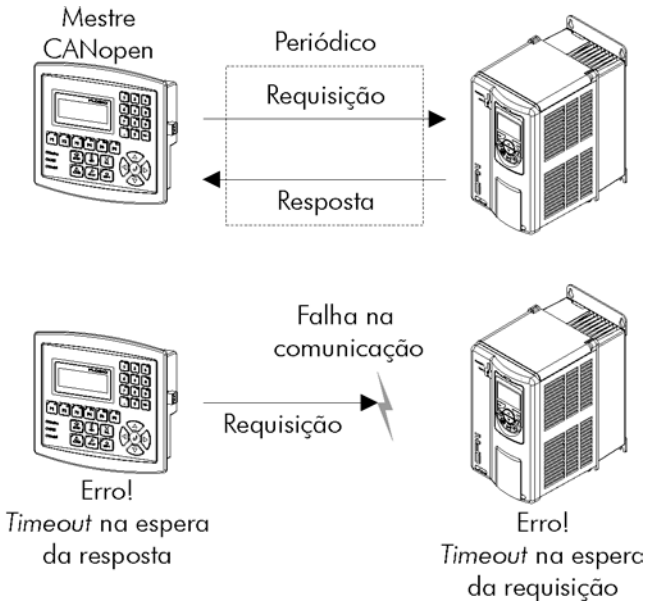


Figura 7.5: *Serviço de controle de erros – Node Guarding*

Para o serviço de *Node Guarging*, 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 milisegundos) para que uma ocorrência de falha seja detectada, caso o escravo não receba nenhum telegrama do mestre. O objeto 100Dh indica quantas falhas em sequência são necessárias até que se considere que houve realmente perda da 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 escravo 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 enviado pelo escravo (*toggle bit*).

Caso a remota RUW06 operando no modo escravo detecte um erro utilizando este mecanismo, ele irá automaticamente para o estado pré-operacional e indicará esta condição no LED CAN.

NOTA!

- Este objeto está ativo mesmo no estado parado (consulte a tabela 7.5).
- 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 HMI.
- O valor mínimo aceito para a remota RUW06 é de 2ms. 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.

7.5.3 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.

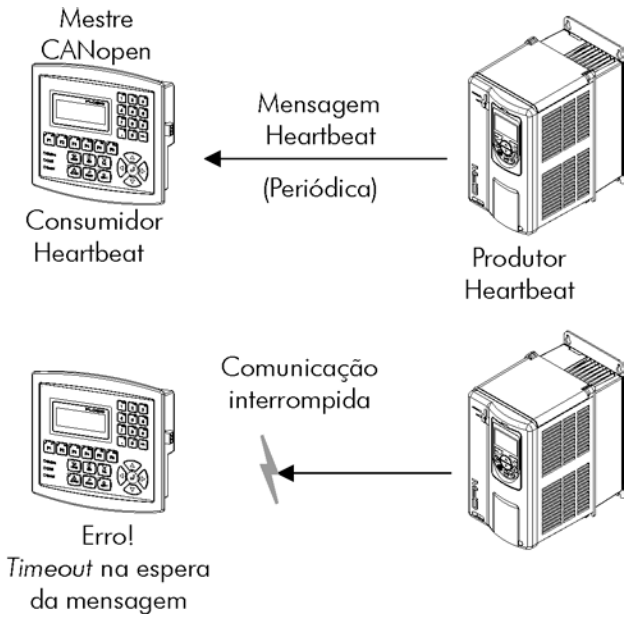


Figura 7.6: *Serviço de controle de erros – Heartbeat*

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 escravo, 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 falhas na comunicação com o mestre.

A remota RUW06 possui os serviços de produtor e consumidor *heartbeat*. Como consumidor, é possível programar diferentes produtores para serem monitorados pelo equipamento:

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 4, 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 milisegundo, 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*, este irá automaticamente para o estado pré-operacional e indicará esta condição no LED CAN.

Como produtor, a remota RUW06 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 dispositivo 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 (consulte a tabela 7.5).
- 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 do erro é retirada da HMI.
- O valor de tempo programado para o consumidor deve ser maior do que o programado para o respectivo produtor. Recomenda-se programar o consumidor com valores múltiplos do utilizado para o produtor.
- Para cada escravo, somente um dos serviços – Heartbeat ou Node Guarding – pode ser habilitado.

7.6 PROCEDIMENTO DE INICIALIZAÇÃO

Uma vez conhecido o funcionamento dos objetos disponíveis para a remota RUW06 operando no modo escravo, 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:

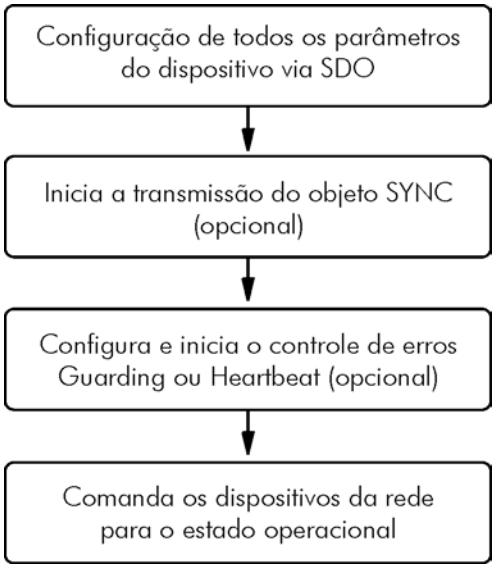


Figura 7.7: *Fluxograma do processo de inicialização*

É necessário observar que os objetos de comunicação da remota RUW06 (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.

8 DESCRIÇÃO DOS OBJETOS PARA MÓDULO I/O

Neste item serão descritos os objetos comuns para módulos genéricos de I/O, definidos pela especificação CANopen, no documento CiA DS 401. Os objetos citados aqui possuem descrição e operação semelhantes, independente do fabricante do módulo de I/O. Isto facilita a interoperabilidade e intercambiabilidade entre diferentes dispositivos.

A tabela a seguir mostra a lista dos objetos disponíveis para a remota RUW06, divididos de acordo com os diferentes modos de operação do equipamento.

Tabela 8.1: *Lista de objetos da RUW06 – Drive Profile*

Índice	Objeto	Nome	Tipo	Acesso	Mapeável
6401h	ARRAY	Read Analogue Input 16-Bit	INTEGER16	ro	Sim
6423h	VAR	Analogue Input Global Interrupt Enable	BOOLAEN	rw	Sim

Sempre que um objeto desta lista for lido ou escrito, o drive irá mapear suas funções nos parâmetros do usuário. Desta forma, ao operar o sistema através destes objetos, os valores dos parâmetros podem ser alterados de acordo com a função utilizada. Nos itens seguintes, é feita uma descrição detalhada de cada um destes objetos, onde são indicados os parâmetros que são utilizados para executar as funções destes objetos.

8.1 MODULO DE ENTRADA ANALÓCIA

8.1.1 Objeto 6401h - Read Analogue Input 16-Bit

Objeto com a carga medida de acordo com a sensibilidade e fundo de escala configurados nos objetos 200Fh e 2010h, além do ganho e offset programados nos objetos 200Dh e 200Eh, como ilustra a Figura 6.2.

Índice	6401h
Nome	Read Anolohue Input 16-Bit
Objeto	ARRAY
Tipo	INTEGER16

Sub-índice	0
Descrição	Número de entradas analógicas
Acesso	ro
Mapeável	Não
Faixa	1 – 2
Valor Padrão	2

Sub-índice	1
Descrição	Entrada analógica 1
Acesso	ro
Mapeável	Sim
Faixa	INTEGER16
Valor Padrão	Não

Sub-índice	2
Descrição	Entrada analógica 2
Acesso	ro
Mapeável	Sim
Faixa	INTEGER16
Valor Padrão	Não

8.1.2 Objeto 6423h – Analogue Input Global Interrupt Enable

Este objeto deve habilitar/desabilitar a interrupção global. Como padrão nenhuma entrada analógica tem a interrupção ativa.
TRUE = interrupção global habilitada (Change of state habilitado);
FALSE = interrupção global desabilitada (Change of state desabilitado).

Índice	6423h
Nome	Analogue Input Global Interrupt Enable
Objeto	VAR
Tipo	Boolean
Parâmetros utilizados	-

Acesso	rw
Mapeável	Sim
Faixa	0 - 1
Valor Padrão	0

9 FALHAS E ALARMES RELACIONADOS COM A COMUNICAÇÃO CANOPEN

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 a interface CAN estiver alimentada e for detectada a falta de alimentação na interface CAN, será sinalizada através do led vermelho ERROR na frontal da unidade remota. Se a alimentação do circuito for restabelecida, a comunicação CAN será reiniciada.

Possíveis Causas/Correção:
- Medir se existe tensão dentro da faixa permitida entre os pinos 1 e 5 do conector da interface CAN.
- Verificar se os cabos de alimentação não estão trocados ou invertidos.
- Verificar problemas de contato no cabo ou no conector da interface CAN.

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[5], o controlador CAN pode ser levado ao estado de *bus off*, onde ele interrompe a comunicação e desabilita a interface CAN.

Caso o número de erros de recepção ou transmissão detectados pela interface CAN seja muito elevado[6], o controlador CAN pode ser levado ao estado de bus off, onde ele interrompe a comunicação e desabilita a interface CAN.
Neste caso será sinalizada através do led vermelho ERROR na frontal da unidade remota.
A comunicação é restabelecida automaticamente se o objeto 2003h estiver com valor 0. Se o valor do objeto 2003h for 1, será necessário desligar e ligar a unidade remota para restabelecer a comunicação.

Possíveis Causas/Correção:
- Verificar curto-circuito nos cabos de transmissão do circuito CAN.
- Verificar se os cabos não estão trocados ou invertidos.
- Verificar se todos os dispositivos da rede utilizam a mesma taxa de comunicação.
- 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.

NODE GUARDING/HEARTBEAT

Descrição:
Controle de erros da comunicação CANopen detectou erro de comunicação utilizando o mecanismo de *guarding*.

Atuação:
Utilizando os mecanismos de controle de erro – *Node Guarding* ou *Heratbeat* – o mestre e o escravo podem trocar telegramas periódicos, em um período pré-determinado. Caso a comunicação seja interrompida por

[5] Para mais informações sobre detecção de erros, consultar especificação CAN.
[6] Para maiores informações sobre detecção de erros, consultar especificação CAN.

algum motivo, tanto mestre quanto escravo poderão detectar erro na comunicação pelo *timeout* na troca destas mensagens.

Neste caso será sinalizada através do led vermelho ERROR na frontal da unidade remota. O erro é eliminado automaticamente quando um dos mecanismos de guarding é restabelecido.

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 detecção de erros pelo escravo sejam múltiplos dos tempos programados para a troca de mensagens no mestre.
- Verificar se o mestre está enviando os telegramas de *guarding* no tempo programado.
- Verificar problemas na comunicação que possam ocasionar perda de telegramas ou atrasos na transmissão.

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