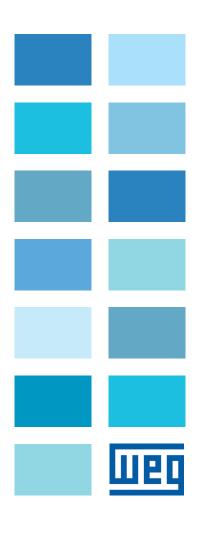
Unidade Remota CANopen

RUW-01

Manual do Usuário







MANUAL DA UNIDADE REMOTA

Série: RUW-01

Idioma: Português

N ° do Documento: 10000301331 / 01

Modelos: CANopen

Data da Publicação: 02/2010



Revisão	Descrição	Capítulo
0	Primeira Edição	-
1	Incluído obieto 2003h - BUS OFF autoreset	7



ÍNDICE

	~		
INIOTRI	JÇOES	-0110	
INSIRI	$\mathbf{I}(\mathbf{I}(\mathbf{I}) = \mathbf{S}$	-(= /	
1140 I I <i>I</i> (-WUI 1/	71100
	5		

1.1 AVISOS DE SEGURANÇA NO MANUAL	
INFORMAÇÕES GERAIS	
2.1 TERMOS E DEFINIÇÕES UTILIZADOS NO MANUAL	2-1 2-1
CONEXÕES E INSTALAÇÃO	
3.1 INSTALAÇÃO MECÂNICA	3-13-23-23-33-3
CONFIGURAÇÃO	
4.1 ENDEREÇO REDE CANOPEN4.2 TAXA DE COMUNICAÇÃO – BAUDRATE	4-1 4-2
DIAGNÓSTICO	
5.1 DIAGNÓSTICO DA COMUNICAÇÃO 5.2 DIAGNÓSTICO DAS ENTRADAS E SAÍDAS DIGITAISINTRODUÇÃO À COMUNICAÇÃO CANOPEN	5-1 5-2
6.1 CAN	6-1 6-1 6-1
6.1.5 CAN e CANopen	6-2 6-2 6-2
6.5 ACESSO AOS DADOS	6-2 6-3
6.9 ARQUIVO EDS	6-4 6-4 6-5
6.14 LIGAÇÃO DO INVERSOR NA REDE CAN	



DICIONÁRIO DE OBJETOS

7.1 ESTRUTURA DO DICIONÁRIO	
7.2 TIPOS DE DADOS	
7.2.1 Tipos Básicos	
7.2.2 Tipos Compostos	
7.2.3 Tipos Estendidos	
7.3 COMMUNICATION PROFILE - OBJETOS PARA COMUNICAÇÃO	
7.4.1 Objeto 2000h - Filter for Digital Inputs 1 to 14	
7.4.2 Objeto 2000h - Filter for Digital Inputs 1 to 14	
7.5 DEVICE PROFILE – OBJETOS COMUNS PARA MÓDULOS GENÉRICOS DE I/O	7-4 7-4
DESCRIÇÃO DOS OBJETOS DE COMUNICAÇÃO	
8.1 OBJETOS DE IDENTIFICAÇÃO	0.4
8.1.1 Objeto 1000h - Device Type	
8.1.2 Objeto 1000h – Bevice Type	
8.1.3 Objeto 1018h – Identity Object	
8.2 SERVICE DATA OBJECTS – SDOS	
8.2.1 Objeto 1200h – Servidor SDO	
8.2.2 Funcionamento dos SDOs	
8.3 PROCESS DATA OBJECTS - PDOS	8-4
8.3.1 Objetos Mapeáveis para os PDOs	8-5
8.3.2 PDOs de Recepção	8-5
8.3.3 PDOs de Transmissão	
8.4 SYNCHRONIZATION OBJECT – SYNC	
8.5 NETWORK MANAGEMENT – NMT	
8.5.1 Controle dos Estados do Escravo	
8.5.2 Controle de Erros – Node Guarding	
8.5.3 Controle de Erros – Heartbeat	
8.6 PROCEDIMENTO DE INICIALIZAÇÃO	8-15
DESCRIÇÃO DOS OBJETOS PARA MODULO I/O	
9.1 OBJETO 6000H - READ INPUT 8-BIT	
9.2 OBJETO 6020H – READ INPUT BIT 1 TO 128	
9.3 OBJETO 6100H – READ INPUT 16-BIT	
9.4 OBJETO 6200H – WRITE OUTPUT 8-BIT	
9.5 OBJETO 6220H – WRITE OUTPUT BIT 1 TO 128	
9.6 OBJETO 6300H – WRITE OUTPUT 16-BIT	
9.7 OBJETO 6306 – ERROR MODE OUTPUT 16-BIT	
FALHAS E ALARMES RELACIONADOS COM A COMUNICAÇÃO CANOPEN	
10.1 SEM ALIMENTAÇÃO NA INTERFACE CAN (BUS POWER OFF)	10-1
10.2 BUS OFF	
10.3 NODE GUARDING/HEARTBEAT	

CARACTERÍSTICAS TÉCNICAS



INSTRUÇÕES DE SEGURANÇA

Este manual contém as informações necessárias para o uso da Unidade Remota WEG - RUW-01.

Ela foi desenvolvida para ser utilizado por pessoas com treinamento ou qualificação técnica adequados para operar este tipo de equipamento.

1.1 AVISOS DE SEGURANÇA NO MANUAL

Neste manual são utilizados os seguintes avisos de segurança:



PERIGO

Não considerar os procedimentos recomendados neste aviso podem levar à morte, ferimentos graves e danos materiais consideráveis.



ATENÇÃO!

Não considerar os procedimentos recomendados neste aviso podem levar a danos materiais.



NOTA!

O texto objetiva fornecer informações importantes para o correto entendimento e bom funcionamento do produto.

1.2 RECOMENDAÇÕES PRELIMINARES



PERIGO!

Somente pessoas com qualificação adequada e familiaridade com a unidade remota CANopen e equipamentos associados devem planejar ou implementar a instalação, partida, operação e manutenção deste equipamento.

Estas pessoas devem seguir todas as instruções de segurança contidas neste manual e/ou definidas por normas locais.

Não seguir as instruções de segurança pode resultar em risco de vida e/ou danos no equipamento.



NOTA!

Para os propósitos deste manual, pessoas qualificadas são aquelas treinadas de forma a estarem aptas para:

- 1. Instalar, aterrar, energizar e operar a RUW-01 de acordo com este manual e os procedimentos legais de segurança vigentes;
- 2. Utilize os equipamentos de proteção de acordo com as normas estabelecidas;
- 3. Prestar serviços de primeiros socorros.



ATENÇÃO!

Os cartões eletrônicos possuem componentes sensíveis a descarga eletrostática. Não toque diretamente sobre os componentes ou conectores.



NOTA!

Leia completamente este manual antes de instalar ou operar este relé.





INFORMAÇÕES GERAIS

Este manual apresenta informações de como instalar, colocar em funcionamento e as principais características da Unidade Remota CANopen RUW-01.

2.1 TERMOS E DEFINIÇÕES UTILIZADOS NO MANUAL

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.

Amp, A: Ampères.

°C: Graus celsius.

CA: Corrente alternada.

CC: Corrente contínua.

Hz: Hertz.

mA: Miliampère = 0.001 Ampère.

2.2 REPRESENTAÇÃO NUMÉRICA

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

2.3 DOCUMENTOS

O protocolo CANopen para a RUW-01 foi desenvolvido baseado nas seguintes especificações e documentos:

Tabela 2.1: Especificação e documentos CiA

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 (Can in Automation), que atualmente é a organização que mantém, divulga e atualiza as informações relativas à rede CANopen.



2.4 SOBRE A RUW-01

A unidade remota CANopen é um equipamento escravo, que permite a expansão de entradas e saídas digitais em um sistema de automação baseado no protocolo CANopen. A RUW-01 deve ser utilizada em conjunto com um mestre CANopen como os cartões PLC2 e PLC11. Apresenta conectores plug-in, caixa metálica e fixação em trilho DIN 35.



CONEXÕES E INSTALAÇÃO

Este capítulo descreve os procedimentos de instalação elétrica e mecânica da RUW-01. As orientações e sugestões devem ser seguidas visando a segurança de pessoas, equipamentos e o correto funcionamento.

3.1 INSTALAÇÃO MECÂNICA

3.1.1 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 1000 m condições nominais;

3.1.2 Posicionamento e Fixação

A RUW-01 pode ser instalada em qualquer posição. Instale em trilho DIN 35 mm, conforme figura 3.1.

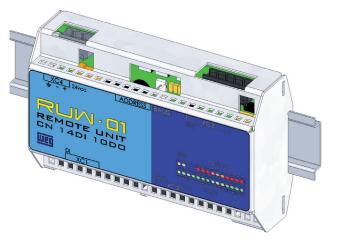




Figura 3.1: Fixação da RUW-01



3.2 INSTALAÇÃO ELÉTRICA

ATENÇÃO!

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.

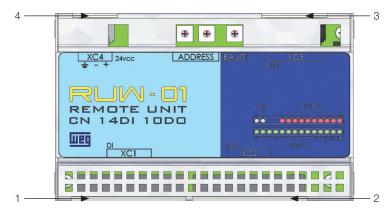


Figura 3.2: Conexões da RUW-01

- 1 Conector XC1 entradas digitais;
- 2 Conector XC2 saídas digitais;
- 3 Conector XC3 para rede CANopen; 4 Conector XC4 alimentação.

3.3 ENERGIZAÇÃO

A RUW-01 é energizada através do conector XC4, conforme figura 3.3.



NOTA!

A tensão de alimentação da RUW-01 deve ser 24 Vcc \pm 15 %.

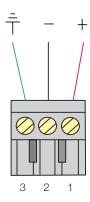


Figura 3.3: Conector de alimentação da RUW-01

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



3.4 CONEXÃO NA REDE CANOPEN

A RUW-01 possui um LED na cor verde para indicar que a interface está alimentada.



NOTA!

Para mais informações, consulte o item 6.10.

3.5 CONEXÃO DAS ENTRADAS DIGITAIS

A RUW-01 apresenta 14 entradas digitais bidirecionais que podem ser acionadas através de uma fonte de tensão externa de 24 Vcc, conforme figura 3.4.

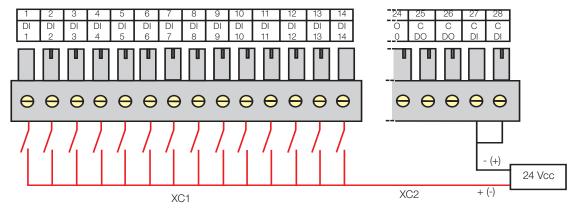


Figura 3.4: XC1: Acionamento das entradas digitais

3.6 CONEXÃO DAS SAÍDAS DIGITAIS

A RUW-01 apresenta 10 saídas digitas bidirecionais a transistor com tensão de trabalho de 24 Vcc e corrente máxima 0.5 A cada. A figura 3.5 ilustra o esquema de ligação das saídas digitais.

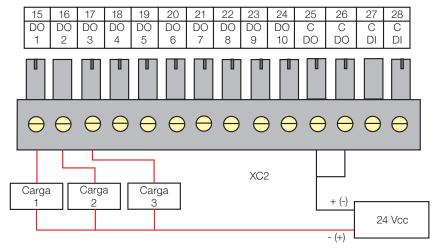


Figura 3.5: XC2: Esquema de ligação das saídas digitais





CONFIGURAÇÃO

A configuração da RUW-01 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.

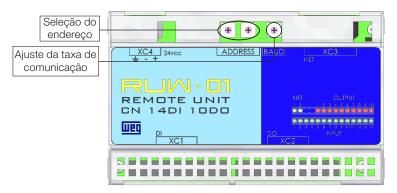


Figura 4.1: Chaves hexadecimal para configuração da RUW-01

4.1 ENDEREÇO REDE CANOPEN

O endereço da unidade remota é configurado através de duas chaves hexadecimal S1 e S2 como ilustra a figura 4.2. 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).



ATENÇÃO!

Caso o endereço seja alterado, ele somente será válido após a RUW-01 ser 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 4.2, onde a chave S1 é a mais significativa e a chave S2 é a menos significativa.

Os valores das chaves conforme a figura 4.2:

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

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

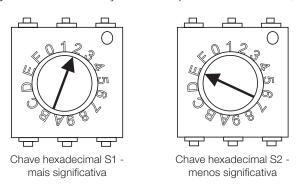


Figura 4.2: Chaves hexadecimal para configuração da RUW-01



4.2 TAXA DE COMUNICAÇÃO - BAUDRATE

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 RUW-01 apresenta a chave hexadecimal S3 para a seleção da taxa de comunicação, como ilustra a figura 4.2. A unidade remota apresenta as seguintes taxas de comunicação:

- 0 = 1 Mbit/s;
- \blacksquare 1 = 800 Kbit/s;
- = 2 = 500 Kbit/s;
- = 3 = 250 Kbit/s;
- = 4 = 125 Kbit/s;
- = 5 = 100 Kbit/s;
- \bullet 6 = 50 Kbit/s;
- = 7 = 20 Kbit/s;
- 8 = 10 Kbit/s.



NOTA!

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



ATENÇÃO!

Caso a taxa de comunicação seja alterada, ela somente será válida após a RUW-01 ser ligada novamente.



DIAGNÓSTICO

A RUW-01 apresenta diagnóstico através de LEDs que indicam o estado de cada entrada e saída digital e o estado da comunicação.

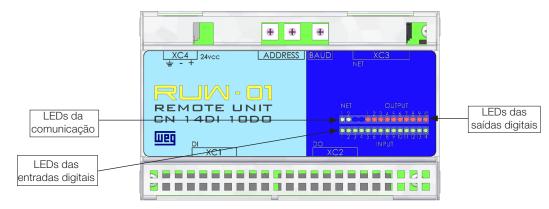


Figura 5.1: LED de indicação da RUW-01

5.1 DIAGNÓSTICO DA COMUNICAÇÃO

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 5.1: 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 200 ms	Dispositivo está no estado Pré-operacional. PDOSs não estão disponiveis para comunicação.				
OPERATIONAL	Verde	Dispositivo está no estado Operacional. Todos os objetos de comunicação estão disponíveis.				

Tabela 5.2: 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 vez por segund		 Indica que o controlador CAN está em estado de Warning ou Error Passive (consulte o 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 vezes 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 (consulte 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 ligar novamente a RUW-01.
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 50 ms	Indica que o Protocolo CANopen não foi inicializado. Verifique se o endereço está ajustado em um valor válido (01h-7Fh).



5.2 DIAGNÓSTICO DAS ENTRADAS E SAÍDAS DIGITAIS

Tabela 5.3: Sinalização do estado das entradas e saídas digitais

Estado	LED I/O	Descrição				
Saída ligada	Vermelho	Saída correspondente está ligada.				
Saída desligada Apagado		Saída correspondente está desligada.				
Entrada ativada	Verde	Entrada correspondente está acionada.				
Entrada desativada	Apagado	Entrada correspondente não está acionada.				

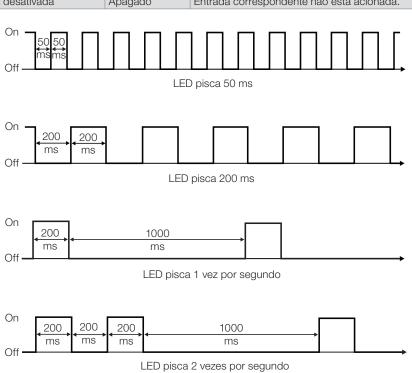


Figura 5.2: Diagrama de tempo para o comportamento da sinalização



INTRODUÇÃO À COMUNICAÇÃO CANOPEN

Para a operação da unidade remota em rede CANopen, é necessário conhecer a forma como a comunicação é feita. Para isto, este capítulo traz uma descrição geral do funcionamento do protocolo CANopen, contendo as funções utilizadas pela RUW-01. Para uma descrição detalhada do protocolo, consulte a documentação CANopen indicada no item anterior.

6.1 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.

6.1.1 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 bits² (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	

6.1.2 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.

6.1.3 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.

6.1.4 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 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 o protocolo CANopen da RUW-01, somente frames standard são aceitos.



6.1.5 CAN e CANopen

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

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

6.2 CARACTERÍSTICAS DA REDE CANOPEN

Por utilizar um barramento CAN como forma de transmissão de telegramas, todos os dispositivos da rede CAN open 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 significar 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.

6.3 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.

A RUW-01 utiliza ainda um circuito de interface isolada com a rede. O componente responsável pela transmissão e recepção de sinais é denominado transceiver, que obedece ao especificado pela ISO 11898.

6.4 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.

A RUW-01 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.

6.5 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 capítulo 7.

6.6 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	Inteiro	16 bits					
11 bits	byte 0	byte 1	byte 2	byte 3	byte 4	byte 5			
	78h	56h	34h	12h	00h	FFh			

6.7 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. A RUW-01 suporta os seguintes objetos de comunicação (COBs):

Tabela 6.1: 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. Somente um SDO do tipo servidor está disponível para a RUW-01.
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.
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. A RUW-01 possui a função de consumidor SYNC.
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. A RUW-01 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 (Node Guarding ou Heartbeat), onde o escravo envia mensagens periódicas para o mestre para informar que a conexão está ativa.

Toda a comunicação da unidade remota 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 capítulo 7.

6.8 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 tabela 6.2 apresenta os valores padrão para os diferentes objetos de comunicação disponíveis na RUW-01. 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 6.2: COB-ID para os diferentes objetos

СОВ	Código da Função (bits 10 – 7)	COB-ID Resultante (função + endereço)
NMT	0000	0
SYNC	0001	128 (80h)
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)
SDO (tx)	1011	1409 – 1535 (581h – 5FFh)
SDO (rx)	1100	1537 – 1663 (601h – 67Fh)
Node Guarding/Heartbeat	1110	1793 – 1919 (701h – 77Fh)

6.9 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 RUW-01 pode ser obtido através do site <u>www.weg.net</u>. É necessário observar a versão de software da unidade remota, para utilizar um arquivo EDS que seja compatível com esta versão.

6.10 PINAGEM DO CONECTOR

O módulo para comunicação CAN possui um conector plug-in de 5 vias (XC3) com a seguinte pinagem:

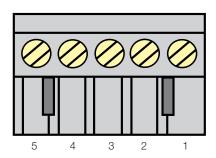


Tabela 6.3: Pinagem do conector XC3 para interface CAN

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

6.11 FONTE DE ALIMENTAÇÃO

A interface CAN da RUW-01 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.

Tabela 6.4: Características da alimentação para interface CAN

Tensão de alimentação (Vcc)		
Mínimo	Máximo	Recomendado
11	30	24
Consumo em 24 V (mA)		
Mínimo	Máximo	Médio
20	50	30



Os módulos de interface CAN possuem um LED na cor verde para indicar que a interface está alimentada.

6.12 RESISTOR DE TERMINAÇÃO

As extremidades do barramento CAN devem possuir um resistor de terminação no valor de 120 Ω / 0.25 W, conectando os sinais CAN_H e CAN_L caso a RUW-01 seja o primeiro ou último elemento do segmento.

6.13 TAXA DE COMUNICAÇÃO

A taxa de comunicação (baudrate) que pode ser utilizada por um equipamento na rede CANopen depende do comprimento do cabo utilizado na instalação. A tabela 6.5 mostra as taxas de comunicação disponíveis para a RUW-01 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 Khit/a	1000 m

Tabela 6.5: Taxas de comunicação suportadas e tamanho da instalação

6.14 LIGAÇÃO DO INVERSOR 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.

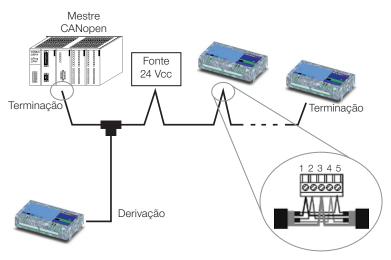


Figura 6.1: RUW-01 em rede CANopen

O cabo para a ligação dos sinais CAN_L e CAN_H deve ter impedância característica de aproximadamente 120 Ω , e um atraso máximo de propagação do sinal de 5 ns/m. Outras características dependem do comprimento do cabo, que deve estar de acordo com a tabela 6.6.



Tabela 6.6: Características do cabo

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



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 7.1: Agrupamentos do dicionário de objetos

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

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

7.1 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.

7.2 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.

7.2.1 Tipos Básicos

Os seguintes tipos básicos de dados são suportados:

- Inteiros com sinal: existem três tipos de inteiros com sinal suportados pela RUW-01, 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 RUW-01, 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.



UNSIGNED16

7.2.2 Tipos Compostos

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

PDO_COMM_PARAMETER: este record define as informações necessárias para configurar um PDO para a comunicação CANopen. O conteúdo e forma como cada campo é utilizado são detalhados no item 8.3.

Sub-índice Descrição da entrada Tipo Número de entradas suportadas neste record **UNSIGNED8** 00hCOB-ID **UNSIGNED32** 01h 02h **UNSIGNED8** Transmission type Inhibit time **UNSIGNED16** 03h 04h **UNSIGNED8**

Reservado

Event timer

05h

Tabela 7.2: 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 8.3.

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

Tabela 7.3: 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 8.2.

Tabela 7.4: Record para configuração dos SDOs

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

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

Tabela 7.5: Record para identificação do dispositivo

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

7.2.3 Tipos Estendidos

A RUW-01 não possui tipos estendidos.

7.3 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 tabela 7.6 apresenta uma lista com os objetos desta faixa suportados pela unidade remota.



Tabela 7.6: Lista de objetos da RUW-01 - 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
		Receive PDO Communication Paramete	r	
1400h	RECORD	1st receive PDO Parameter	PDO CommPar	RW
1401h	RECORD	2nd 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
Transmit PDO Communication Parameter				
1800h	RECORD	1st transmit PDO Parameter	PDO CommPar	RW
1801h	RECORD	2nd 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

Demais objetos não mostrados nesta lista não são utilizados pela RUW-01, ou então estão em faixas reservadas do dicionário.

7.4 MANUFACTURER SPECIFIC - OBJETOS ESPECÍFICOS DA RUW-01

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 da RUW-01, nesta faixa de objetos foi disponibilizado um objeto para filtro das entradas digitais e outro objeto para configuração do BUS OFF autoreset.

7.4.1 Objeto 2000h - Filter for Digital Inputs 1 to 14

Através deste objeto é possível parametrizar o tempo, em milisegundos (ms), do filtro para cada entrada digital. A estrutura deste objeto é apresentada abaixo.

O valor programado em cada um dos sub-índices representa o tempo que a respectiva entrada digital deve permanecer estável (sem transições) para que o novo valor seja assumido pela RUW-01.

Índice	2000h
Nome	Filter for digital inputs 1 to 14
Objeto	Array
Tipo	SDO Parameter

Sub-índice	0
Descrição	Number of Entries
Acesso	RO
Mapeável	Não
Faixa	UNSIGNED16
Valor Padrão	14

Sub-índice	1 a 14
Descrição	Filter for digital input 1 to 14
Acesso	RW
Mapeável	Não
Faixa	UNSIGNED16
Valor Padrão	1ms (máximo = 65535ms)



7.4.2 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

7.5 DEVICE PROFILE - OBJETOS COMUNS PARA MÓDULOS GENÉRICOS DE I/O

A documentação CANopen também inclui propostas para padronização de determinados tipos de dispositivos. No caso da RUW-01, ele segue o descrito pela CiA DS 401 – Device Profile for Generic I/O Modules. Este documento descreve um conjunto de objetos que devem ser comuns para unidades remotas, 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. A tabela 7.6 apresenta os objetos do Device Profile que estão implementados na RUW-01. Para uma descrição detalhada de quais objetos estão disponíveis nesta faixa do dicionário de objetos, consulte a seção 9.

Tabela 7.7: Lista de objetos da RUW-01 - Device Profile

Índice	Objeto	Nome	Tipo	Acesso
6000h	ARRAY	Read Input 8-bit	UNSIGNED8	ro
6020h	ARRAY	Read Input Bit 1 to 128	BOOLEAN	ro
6100h	ARRAY	Read Input 16-bit	UNSIGNED16	ro
6200h	ARRAY	Write Output 8-bit	UNSIGNED8	rw
6220h	ARRAY	Write Output Bit 1 to 128	BOOLEAN	rw
6300h	ARRAY	Write Output 16-bit	UNSIGNED16	rw
6306h	ARRAY	Error Mode Output 16-Bit	UNSIGNED16	rw
6307h	ARRAY	Error Value Output 16-Bit	UNSIGNED16	rw



DESCRIÇÃO DOS OBJETOS DE COMUNICAÇÃO

Nesta seção são descritos detalhadamente cada um dos objetos citados na tabela 7.5, além de descrever também o funcionamento dos objetos de comunicação (COBs) referenciados no item 6.7. É necessário conhecer como estes objetos são operados para utilizar as funções disponíveis para a comunicação da RUW-01.

8.1 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.

8.1.1 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	0003.0191h

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. Para a RUW-01, estes valores são respectivamente 0191h (segue o especificado pelo documento CiA 401 - Device Profile for Generic I/O Modules), e 0003h (possui funcionalidades das entradas e saídas do módulo).

8.1.2 Objeto 1001h - Error Register

Este objeto indica a ocorrência ou não de erro no dispositivo. O documento CiA 401 - Device Profile for Generic I/O Modules não especifica este objeto.

8.1.3 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	0000.0400h

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	Não implementado

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 não é implementado na RUW-01.

8.2 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.

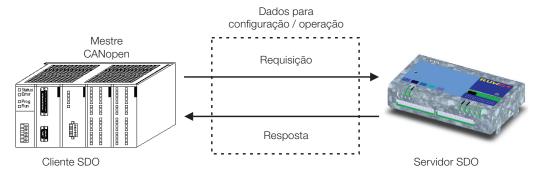


Figura 8.1: Comunicação entre Cliente e Servidor SDO

8.2.1 Objeto 1200h - Servidor SDO

A RUW-01 possui um único SDO do tipo servidor, que possibilita o acesso a todo o seu dicionário de objetos. Através dele, um cliente SDO pode configurar a comunicação, parâmetros e modos de operação do inversor. Todo o servidor SDO possui um objeto, do tipo SDO_PARAMETER (consulte o item 7.2.2), 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

8.2.2 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			
11 DILS	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 8.1: Lista de objetos da RUW-01 – Device Profile

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, porém os comandos serão diferentes:

Tabela 8.2: 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.

Exemplo: um cliente SDO solicita a RUW-01 no endereço 1, a leitura do objeto identificado pelo índice 2000h, sub-índice 1, que representa um inteiro de 16 bits. O telegrama do mestre possui a seguinte forma:

Identificador	Comando	Índice		Sub-índice	Dados			
601h	40h	00h	20h	00h	00h	00h	00h	00h

A RUW-01 responde à requisição, indicando que o valor para o referido objeto é igual a 999 3:

Identificador	Comando	Índice		Sub-índice	Dados			
581h	4Bh	00h	20h	00h	E7	03h	00h	00h

8.3 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.

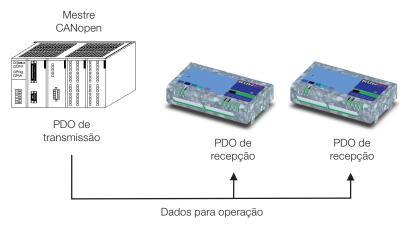


Figura 8.2: Comunicação utilizando PDOs



NOTA!

PDOs somente podem ser transmitidos ou recebidos quando o dispositivo está no estado operacional. A figura 8.2 ilustra os estados disponíveis para um nó da rede CANopen.

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



8.3.1 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, saídas, 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 da RUW-01 traz a lista de todos os objetos disponíveis para o inversor, informando se o objeto é mapeável ou não.

8.3.2 PDOs de Recepção

Os PDOs de recepção, ou RPDOs, são responsáveis por receber dados que outros dispositivos enviam para a rede CANopen. A RUW-01 possui 2 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

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

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:

Tabela 8.3: Descrição do COB-ID

Bit	Valor	Descrição	
Od (MCD)	0	PDO está habilitado	
31 (MSB)	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	



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 a RUW-01 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 8.4.

Tabela 8.4: Descrição do tipo de transmissão

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

- 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

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 10 = número de objetos mapeados
Valor Padrão	1600h: 2 1601h: 1

Sub-índice	1	
Descrição	1º objeto mapeado no PDO	
Acesso	RW	
Mapeável	Não	
Faixa	UNSIGNED32	
Valor Padrão	1600h: 6200.0108h 1601h: 6300.0110h	

Sub-índice	2	
Descrição	2° objeto mapeado no PDO	
Acesso	RW	
Mapeável	Não	
Faixa	UNSIGNED32	
Valor Padrão	1600h: 6200.0208h 1601h: 0	

Este parâmetro indica os objetos mapeados nos PDOs de recepção da RUW-01. 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 ⁴ e tamanho (em bits) em um campo UNSIGNED32, com o seguinte formato:

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



	UNSIGNED32	
Índice (16 bits)	Índice (16 bits)	Tamanho do objeto (8 bits)

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

- Sub-índice 0 = 10: o RPDO possui dez objetos mapeados. Cada saída digital pode representar um objeto.
- Sub-índice 1 = 6200.0108h: o primeiro objeto mapeado possui índice igual a 6200h, sub-índice 1, e tamanho igual a 8 bits. Este objeto corresponde a escrita das 8 saídas digitais menos significativas da RUW-01.
- Sub-índice 2 = 6200.0208h: o segundo objeto mapeado possui índice igual a 6000h, sub-índice 2, e tamanho igual a 8 bits. Este objeto corresponde a escrita das 3 saídas digitais mais significativas da RUW-01.

Desta forma, sempre que este PDO receber um telegrama, ele vai saber que o telegrama deverá conter dois bytes de dados, com o conteúdo para as saídas digitais da RUW-01. É possível modificar este mapeamento, alterando a quantidade ou o número dos objetos mapeados. Lembrar que no máximo podem ser mapeados 4 objetos ou 8 bytes.



NOTA!

- Para poder alterar os objetos mapeados em um PDO, primeiro é necessário escrever o valor 0 (zero) no sub-índice 0 (zero). Desta forma, os valores dos sub-índices 1 até 4 podem ser alterados. Depois de feito o mapeamento desejado, deve-se escrever novamente no sub-índice 0 (zero) o número de objetos que foram mapeados, habilitando novamente o PDO.
- Para agilizar a atualização dos dados via PDO, os valores recebidos através destes objetos não são salvos na memória não volátil do inversor. Desta forma, após um desligamento ou reset do equipamento, os objetos modificados por um RPDO voltam para o seu valor padrão.
- Não esquecer que os PDOs somente podem ser recebidos caso a RUW-01 esteja no estado operacional.

8.3.3 PDOs de Transmissão

Os PDOs de transmissão, ou TPDOs, como o nome diz, são responsáveis por transmitir dados para a rede CANopen. A RUW-01 possui 2 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



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 8.3.

O sub-índice 2 indica o tipo de transmissão deste objeto, que segue o descrito pela tabela 8.4. 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 100 µs) 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 1 ms), 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 RUW-01 é 2 ms.
- 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	
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 14 = número de objetos mapeados	
Valor Padrão	1A00h: 2 1A01h: 1	

Sub-índice	1
Descrição	1º objeto mapeado no PDO
Acesso	RW
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	1A00h: 6000.0108h 1A01h: 6100.0110h

Sub-índice	2
Descrição	2º objeto mapeado no PDO
Acesso	RW
Mapeável	Não
Faixa	UNSIGNED32
Valor Padrão	1A00h: 6000.0208h 1A01h: 0

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

	UNSIGNED32		
ĺ	ndice (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 = 14: este PDO possui quatorze objetos mapeados. Cada entrada digital pode representar um objeto.
- Sub-índice 1 = 6000.0108h: o primeiro objeto mapeado possui índice igual a 6000h, sub-índice 1, e tamanho igual a 8 bits. Este objeto corresponde ao estado das entradas digitais menos significativas da RUW-01.
- Sub-índice 2 = 6000.0208h: o segundo objeto mapeado possui índice igual a 6000h, sub-índice 2, e tamanho igual a 8 bits. Este objeto corresponde ao estado das entradas digitais mais significativas da RUW-01.

Desta forma, sempre que este PDO for transmitir seus dados, ele vai elaborar o seu telegrama contendo dois bytes de dados, com os valores das entradas digitais da RUW-01. É possível modificar este mapeamento, alterando a quantidade ou o número dos parâmetros mapeados. Lembrar que no máximo podem ser mapeados 4 objetos ou 8 bytes.



NOTA!

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

8.4 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 RUW-01 possui a função de consumidor SYNC e, portanto, pode programar seus PDOs para serem síncronos. Como descrito na tabela 8.4, PDOs síncronos são aqueles relacionados com o objeto de sincronização, e portanto podem ser programados para serem transmitidos ou atualizados com base neste objeto.

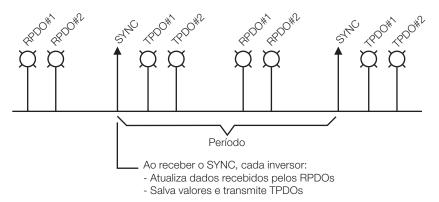


Figura 8.3: SYNC

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 RUW-01, existe um objeto para a configuração do COB-ID do consumidor SYNC.

Índice	1005h	
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 PDOs assíncronos e SDOs.

8



8.5 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 a RUW-01 estão disponíveis os serviços de controle do nó e de controle de erros (utilizando Node Guarding ou Heartbeat).

8.5.1 Controle dos Estados do Escravo

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

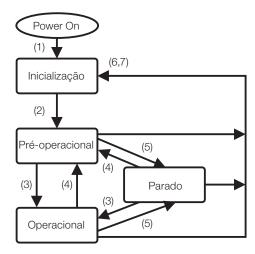


Figura 8.4: Diagrama de estados do nó CANopen

Tabela 8.5: 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, 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 8.6 mostra os objetos disponíveis para cada estado.

Tabela 8.6: 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 8.7: 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 8.2). O comando Reset node faz com que a RUW-01 execute um reset completo do dispositivo, enquanto que o comando Reset communication faz com que o dispositivo reinicialize apenas os objetos relativos à comunicação CANopen.

8.5.2 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.

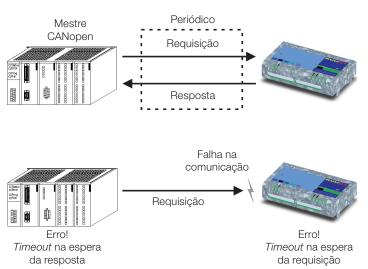


Figura 8.5: Serviço de controle de erros - Node Guarding

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

8



O objeto 100Ch permite programar o tempo necessário (em milisegundos) para que uma ocorrência de falha seja detectada, caso a RUW-01 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 RUW-01 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	
identificador	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 a RUW-01 detecte um erro utilizando este mecanismo, ele irá automaticamente para o estado pré-operacional sinalizando seu estado no LED CAN e indicando erro de node guarding no LED ERROR.



NOTA!

- Este objeto está ativo mesmo no estado parado (consulte a tabela 23).
- O valor 0 (zero) em um dos dois objetos desabilita esta função.
- Depois de detectado o erro, caso o serviço seja habilitado mais uma vez, a indicação do erro é retirada dos LEDs de diagnóstico.
- O valor mínimo aceito para o RUW-01 é de 2 ms, mas levando-se em conta a taxa de transmissão e o número de pontos na rede, os tempos programados para essa função devem ser coerentes, de maneira que haja tempo suficiente para transmissão dos telegramas e também para que o resto da comunicação possa ser processada.
- Para cada escravo, somente um dos serviços Heartbeat ou Node Guarding pode ser habilitado.

8.5.3 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.

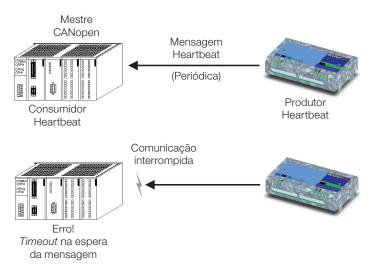


Figura 8.6: Serviço de controle de erros - Heartbeat



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 RUW-01 possui os serviços de produtor e consumidor heartbeat. Como consumidor, é possível programar até 4 diferentes produtores para serem monitorados pelo inversor:

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 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 produtos heartbeat, o inversor irá automaticamente para o estado pré-operacional sinalizando seu estado no LED CAN e indicando erro de Heartbeat no LED ERROR.

Como produtor, a RUW-01 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 milisegundos no qual o produtor envie um telegrama heartbeat



para a rede. Uma vez programado, o inversor inicia a transmissão de mensagens com o seguinte formato:

Identificador	by	rte 1
identificador	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 8.6).
- 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 dos LEDs de diagnóstico.
- O valor de tempo programado para o consumidor deve ser maior do que o programado para o respectivo produtor. Recomenda-se inclusive 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.

8.6 PROCEDIMENTO DE INICIALIZAÇÃO

Uma vez conhecido o funcionamento dos objetos disponíveis para a unidade remota RUW-01, 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:

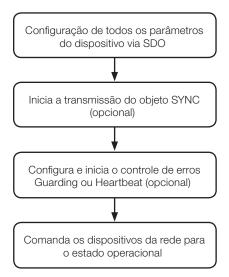


Figura 8.7: Fluxograma do processo de inicialização

É necessário observar que os objetos de comunicação da RUW-01 (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. Os objetos específicos do fabricante (a partir de 2000h), também devem ser parametrizados novamente





DESCRIÇÃO DOS OBJETOS PARA MODULO I/O

Neste capítulo 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.

9.1 OBJETO 6000H - READ INPUT 8-BIT

Possibilita a leitura das entradas digitais em dois bytes, onde o sub-índice 1 é o byte menos significativo e sub-indice 2 é o byte mais significativo, conforme figura 9.1.

	-	-	14	13	12	11	10	9	8	7	6	5	4	3	2	1
	Sub-indice 2							Sub-indice 1								
Byte mais significativo											Byte	e menos	significa	ativo		

Figura 9.1: Mapa das entradas digitais para o objeto 6000h

Índice	6000h
Nome	Read Input 8-bit
Objeto	ARRAY
Tipo	UNSIGNED8

Sub-índice	0
Descrição	Number of Inputs 8-Bit
Acesso	RO
Mapeável	Não
Faixa	UNSIGNED8
Valor Padrão	2

Sub-índice	1
Descrição	Digital Input 8-Bit
Acesso	RO
Mapeável	Sim
Faixa	UNSIGNED8
Valor Padrão	-

Sub-índice	2
Descrição	Digital Input 8-Bit
Acesso	RO
Mapeável	Sim
Faixa	UNSIGNED8
Valor Padrão	-

9.2 OBJETO 6020H - READ INPUT BIT 1 TO 128

Possibilita a leitura individual das entradas digitais, onde cada sub-índice equivale a uma entrada digital, conforme figura 9.2.

-	-	14	13	12	11	10	9	8	7	6	5	4	3	2	1
		Sub-	Sub-	Sub-	Sub-	Sub-	Sub-	Sub-	Sub-	Sub-	Sub-	Sub-	Sub-	Sub-	Sub-
-	-	indice 14	indice 13	indice 12	índice 11	índice 10	índice 9	índice 8	indice 7	índice 6	índice 5	índice 4	indice 3	indice 2	índice 1

Figura 9.2: Mapa das entradas digitais para o objeto 6020h

Índice	6020h
Nome	Read Input Bit 1 to 128
Objeto	Array
Tipo	BOOLEAN



Sub-índice	0
Descrição	Number of Inputs 1-Bit
Acesso	RO
Mapeável	Não
Faixa	BOOLEAN
Valor Padrão	14

Sub-índice	1 a 14
Descrição	Read Single Input
Acesso	RO
Mapeável	Sim
Faixa	BOOLEAN
Valor Padrão	-

9.3 OBJETO 6100H - READ INPUT 16-BIT

Possibilita a leitura das entradas digitais através de um WORD (16 bit), conforme figura 9.3.



Figura 9.3: Mapa das entradas digitais para o objeto 6100h

Índice	6100h				
Nome	Read Input 16-bit				
Objeto	ARRAY				
Tipo	UNSIGNED16				

Sub-índice	0
Descrição	Number of Inputs 16-Bit
Acesso	RO
Mapeável	Não
Faixa	UNSIGNED16
Valor Padrão	1

Sub-índice	1
Descrição	Digital Input 16-Bit
Acesso	RO
Mapeável	Sim
Faixa	UNSIGNED16
Valor Padrão	-

9.4 OBJETO 6200H - WRITE OUTPUT 8-BIT

Possibilita a escrita das saídas digitais em dois bytes, onde o sub-índice 1 é o byte menos significativo e sub-indice 2 é o byte mais significativo, conforme figura 9.4.

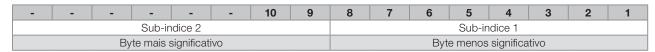


Figura 9.4: Mapa das entradas digitais para o objeto 6200h

Índice	6200h
Nome	Write Output 8-bit
Objeto	ARRAY
Tipo	UNSIGNED8

Sub-índice	0
Descrição	Number of Outputs 8-Bit
Acesso	RO
Mapeável	Não
Faixa	UNSIGNED8
Valor Padrão	2

9



Sub-índice	1
Descrição	Digital Output 8-Bit
Acesso	RO
Mapeável	Sim
Faixa	UNSIGNED8
Valor Padrão	0

Sub-índice	2
Descrição	Digital Output 8-Bit
Acesso	RO
Mapeável	Sim
Faixa	UNSIGNED8
Valor Padrão	0

9.5 OBJETO 6220H - WRITE OUTPUT BIT 1 TO 128

Possibilita a escrita individual das saídas digitais, onde cada sub-índice equivale a uma saída digital, conforme figura 9.5.

-	-	-	-	-	-	10	9	8	7	6	5	4	3	2	1
-	-	Sub- índice 14	Sub- índice 13	Sub- índice 12	Sub- índice 11	Sub- índice 10	Sub- índice 9	Sub- índice 8	Sub- índice 7	Sub- índice 6	Sub- índice 5	Sub- índice 4	Sub- índice 3	Sub- índice 2	Sub- índice 1

Figura 9.5: Mapa das entradas digitais para o objeto 6220h

Índice	6220h
Nome	Write Output Bit 1 to 128
Objeto	Array
Tipo	BOOLEAN

Sub-índice	0
Descrição	Number of Outputs 1-Bit
Acesso	RO
Mapeável	Não
Faixa	BOOLEAN
Valor Padrão	14

Sub-índice	1 a 14
Descrição	Write Output
Acesso	RO
Mapeável	Sim
Faixa	BOOLEAN
Valor Padrão	0

9.6 OBJETO 6300H - WRITE OUTPUT 16-BIT

Possibilita a escrita das saídas digitais através de um WORD (16 bit), conforme figura 9.6.



Figura 9.6: Mapa das entradas digitais para o objeto 6100h

Índice	6100h
Nome	Read Input 16-bit
Objeto	ARRAY
Tipo	UNSIGNED16



Sub-índice	0
Descrição	Number of Outputs 16-Bit
Acesso	RO
Mapeável	Não
Faixa	UNSIGNED16
Valor Padrão	1

Sub-índice	1
Descrição	Digital Output 16-Bit
Acesso	RO
Mapeável	Sim
Faixa	UNSIGNED16
Valor Padrão	0

9.7 OBJETO 6306 - ERROR MODE OUTPUT 16-BIT

Este objeto define se uma saída digital recebe um valor pré-definido (objeto 6307h) em caso de erro interno ou quando a RUW-01 vai para STOP MODE. O sub-índice 1 define uma WORD (16 Bit) onde cada bit desta WORD configura a ação para uma saída, conforme figura 9.7.

Saídas	-	-	-	-	-	-	10	9	8	7	6	5	4	3	2	1
WORD	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Valor definido	-	_	-	-	-	-	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Figura 9.7: Mapa das entradas digitais para o objeto 6200h

- 1 valor da saída é pré-definido no objeto 6307h;
- 0 valor da saída mantido em caso de erro.

Índice	6306h
Nome	Error Mode Output 16-Bit
Objeto	Array
Tipo	UNSIGNED16

Sub-índice	0
Descrição	Number of Output 16-Bit
Acesso	RO
Mapeável	Não
Faixa	UNSIGNED16
Valor Padrão	1

Sub-índice	1
Descrição	Digital Output Error Mode 16-Bit
Acesso	RO
Mapeável	Sim
Faixa	UNSIGNED16
Valor Padrão	03FFh

9.8 OBJETO 6307 - ERROR VALUE OUTPUT 16-BIT

Neste objeto é parametrizado o valor que a saída digital deverá apresentar em caso de erro interno ou quando a RUW-01 vai para STOP MODE. O sub-índice 1 define uma WORD (16 Bit) onde cada bit desta WORD configura a ação para uma saída, conforme figura 9.8.

Saídas	-	-	-	-	-	-	10	9	8	7	6	5	4	3	2	1
WORD	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Valor definido	-	-	-	-	-	-	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Figura 9.8: Mapa das entradas digitais para o objeto 6200h



- 1 valor da saída é 1 (ligada) se habilitado no objeto 6306h; 0 valor da saída é 0 (desligada) se habilitado no objeto 6306h.

Índice	6307h
Nome	Error Value Output 16-Bit
Objeto	Array
Tipo	UNSIGNED16

Sub-índice	0
Descrição	Number of Output 16-Bit
Acesso	RO
Mapeável	Não
Faixa	UNSIGNED16
Valor Padrão	1

Sub-índice	1
Descrição	Digital Output 16-Bit
Acesso	RO
Mapeável	Sim
Faixa	UNSIGNED16
Valor Padrão	03FFh





FALHAS E ALARMES RELACIONADOS COM A COMUNICAÇÃO CANOPEN

10.1 SEM ALIMENTAÇÃO NA INTERFACE CAN (BUS POWER OFF)

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, 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 entre os pinos 1 e 5 do conector da interface CAN, conforme indicado na tabela 6.3.
- Verificar se os cabos de alimentação não estão trocados ou invertidos.
- Verificar problemas de contato no cabo ou no conector da interface CAN.

10.2 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.

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.

10.3 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 algum motivo, tanto mestre quanto escravo poderão detectar o 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 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 no tempo programado.
- Verificar problemas na comunicação que possam ocasionar perda de telegramas ou atrasos na transmissão.





CARACTERÍSTICAS TÉCNICAS

Geral

Alimentação: 24 Vcc Tolerância: -15 % a +15 %

Consumo máximo (24 Vcc): 120 mA

I/O

Entradas Digitais:

14 entradas digitais isoladas, bidirecionais

Consumo: 12 mA @ 24 Vcc

Tensão mínima para nível lógico alto: 15 Vcc
 Tensão máxima para nível lógico baixo: 3 Vcc

Tensão máxima: 30 Vcc

Saídas Digitais:

10 saídas bidirecionais a transistor

Capacidade: 0.5 A @ 24 VccTensão máxima: 30 Vcc

Rede de Comunicação

Protocolo CANopen

Interface CAN:

Tensão nominal: 24 Vcc
Tensão máxima: 30 Vcc
Tensão mínima: 11 Vcc
Corrente típica: 30 mA
Corrente máxima: 60 mA

Programação

Endereço (Node ID):

DIP switch hexadecimal

Taxa de comunicação - Baudrate:

DIP switch hexadecimal

LEDs

- 2 LEDs tipo bicolor para status da rede (CAN (NET 1), ERROR (NET 2))
- 1 LED laranja de power on
- 1 LED verde para alimentação da CAN
- 14 LEDs verdes para sinalização das entradas digitais
- 10 LEDs vermelhos para sinalização das saídas digitais



Dimensões Mecânicas

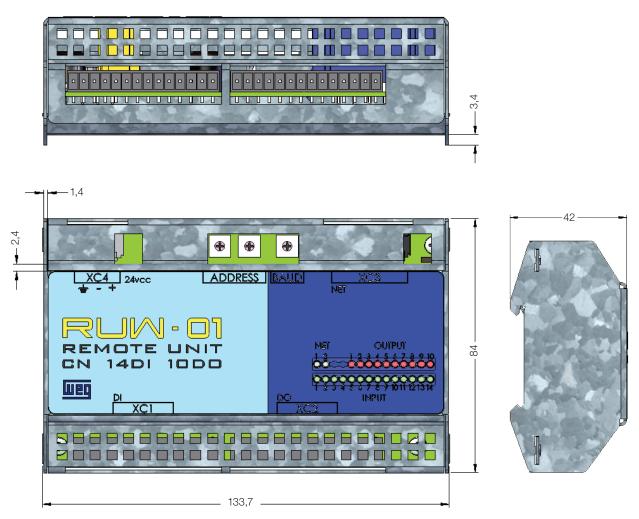


Figura 11.1: Dimensões da RUW-01 (mm)