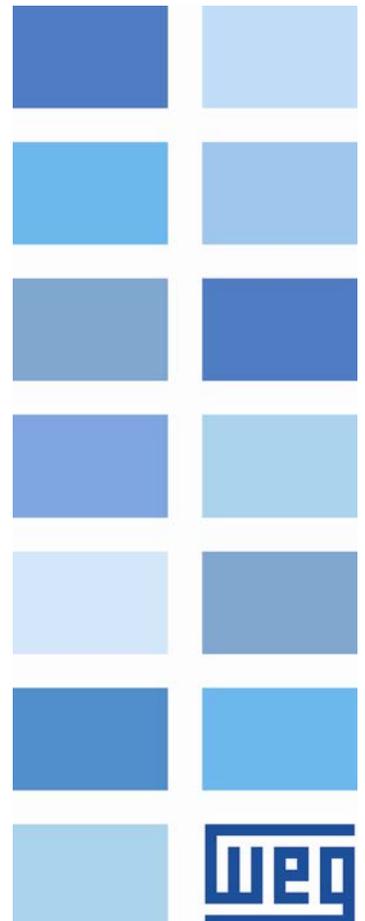


WEGBus

SCA06

Manual do Usuário





Manual do Usuário WEGBus

Série: SCA06

Idioma: Português

N ° do Documento: 10003501638 / 00

Data da Publicação: 04/2015

SUMÁRIO

SUMÁRIO.....	3
SOBRE O MANUAL.....	5
ABREVIações E DEFINIções.....	5
REPRESENTAÇÃO NUMÉRICA.....	5
1 INTRODUÇÃO À COMUNICAÇÃO SERIAL.....	6
2 DESCRIÇÃO DAS INTERFACES.....	7
2.1 MÓDULO DE EXPANÇÃO E COMUNICAÇÃO RS232 E RS485 ECO1.....	7
2.2 RS232.....	7
2.2.1 Indicações.....	7
2.2.2 Conexão com a Rede RS232.....	7
2.2.3 Cabos para Ligação em RS232.....	7
2.2.4 Pinagem do Conector.....	7
2.3 RS485.....	8
2.3.1 Indicações.....	8
2.3.2 Características da interface RS485.....	8
2.3.3 Pinagem do Conector.....	8
2.3.4 Resistor de terminação.....	9
2.3.5 Conexão com a Rede RS485.....	9
3 PARAMETRIZAÇÃO.....	10
3.1 SÍMBOLOS PARA DESCRIÇÃO DAS PROPRIEDADES.....	10
P00650 – ENDEREÇO DO SERVOCONVERSOR NA COMUNICAÇÃO SERIAL 1 – RS232.....	10
P00656 – ENDEREÇO DO SERVOCONVERSOR NA COMUNICAÇÃO SERIAL 2 – RS485.....	10
P00652 – BIT RATE SERIAL 1 – RS232.....	10
P00658 – BIT RATE SERIAL 2 – RS485.....	10
P00653 – CONFIGURAÇÃO SERIAL 1 – RS232.....	11
P00659 – CONFIGURAÇÃO SERIAL 2 – RS485.....	11
P00654 – SELECIONA PROTOCOLO SERIAL 1 – RS232.....	11
P00660 – SELECIONA PROTOCOLO SERIAL 2 – RS485.....	11
P00662 – AÇÃO PARA ERRO DE COMUNICAÇÃO.....	11
P00663 – WATCHDOG SERIAL.....	12
P00664 – SALVA PARÂMETROS EM MEMÓRIA NÃO VOLÁTIL.....	12
P00667 – SALVA EM MARCADORES.....	12
4 PROTOCOLO WEGBUS.....	14
4.1 ENDEREÇAMENTO NO PROTOCOLO WEGBUS.....	14
4.2 CAMPOS DO PROTOCOLO.....	14
4.3 FORMATO DOS TELEGRAMAS.....	15
4.3.1 Telegrama de leitura.....	15
4.3.2 Telegrama de escrita.....	16
4.4 EXEMPLO DE TELEGRAMAS UTILIZANDO O PROTOCOLO WEGBUS.....	17
5 FALHAS E ALARMES RELACIONADOS COM A COMUNICAÇÃO WEGBUS.....	19
A00128/F00028 – TIMEOUT NA RECEPÇÃO DE TELEGRAMAS.....	19
I. APÊNDICES.....	20
APÊNDICE A. TABELA ASCII.....	20

SOBRE O MANUAL

Este manual fornece a descrição necessária para a operação do servoconversor SCA06 utilizando o protocolo WEGBus. Este manual deve ser utilizado em conjunto com manual do usuário do SCA06.

ABREVIações E DEFINIções

ASCII	American Standard Code for Information Interchange
CRC	Cycling Redundancy Check
EIA	Electronic Industries Alliance
TIA	Telecommunications Industry Association
RTU	Remote Terminal Unit

REPRESENTAÇÃO NUMÉRICA

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

1 INTRODUÇÃO À COMUNICAÇÃO SERIAL

Em uma interface serial os bits de dados são enviados sequencialmente através de um canal de comunicação ou barramento. Diversas tecnologias utilizam comunicação serial para transferência de dados, incluindo as interfaces RS232 e RS485.

As normas que especificam os padrões RS232 e RS485, no entanto, não especificam o formato nem a sequência de caracteres para a transmissão e recepção de dados. Neste sentido, além da interface, é necessário identificar também o protocolo utilizado para comunicação. Dentre os diversos protocolos existentes, um protocolo muito utilizado na indústria é o protocolo WEGBus.

A seguir serão apresentadas características das interfaces seriais RS232 e RS485 disponíveis para o produto, bem como a descrição detalhada dos protocolos para utilização destas interfaces.

2 DESCRIÇÃO DAS INTERFACES

Para possibilitar a comunicação serial no servoconversor SCA06, é necessário utilizar o módulo de expansão e comunicação ECO1 descrito a seguir. Informações sobre a instalação deste módulo podem ser obtidas no guia que acompanha o acessório.

2.1 MÓDULO DE EXPANÇÃO E COMUNICAÇÃO RS232 E RS485 ECO1



- Item WEG: 11330271.
- Composto pelo módulo de comunicação ECO1 (figura ao lado) e um guia de montagem.
- Interface isolada galvanicamente e com sinal diferencial, conferindo maior robustez contra interferência eletromagnética.
- Os sinais RS232 e RS485 são canais independentes, podendo ser utilizados simultaneamente.

2.2 RS232

2.2.1 Indicações

O LED LA121 indica (acende) quando há transmissão de dados através da comunicação RS232.

2.2.2 Conexão com a Rede RS232

- Os sinais RX e TX do inversor devem ser ligados respectivamente aos sinais TX e RX do mestre, além da conexão do sinal de referência (GND).
- A interface RS232 é muito susceptível a interferências. Por este motivo, o cabo utilizado para comunicação deve ser o mais curto possível – sempre menor que 10 metros – e deve ser colocado em separado da fiação de potência que alimenta o inversor e motor.

2.2.3 Cabos para Ligação em RS232

Caso seja desejado, estão disponíveis itens dos seguintes cabos para ligação em RS232 entre o servoconversor e um mestre da rede, como um PC:

Tabela 2.1: Cabos RS232

<i>Cabo</i>	<i>Item</i>
Cabo RS232 blindado com conectores DB9 Comprimento: 3 metros	10050328
Cabo RS232 blindado com conectores DB9 fêmea Comprimento: 10 metros	10191117

Outros cabos, porém, podem ser encontrados no mercado – em geral denominados null-modem – ou montados de acordo com o desejado para a instalação.

2.2.4 Pinagem do Conector

A conexão para a interface RS232 está disponível através dos conectores XA121 e XA122 utilizando a seguinte pinagem:

Tabela 2.1: Pinagem do conector para RS232 XA121

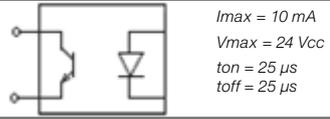
Pino	Função
1	Terra
2	RX_232
3	TX_232
4	Reservado ¹
5	GND
6	
7	
8	A
9	B

Tabela 2.2: Pinagem do conector para RS232 XA122

Pino	Função
1	NC
2	RX_232
3	TX_232
4	Reservado <small>Erro! Indicador não definido.</small>
5	GND
6	NC
7	232 RTS
8	NC
9	NC
Caraça	Terra

2.3 RS485

2.3.1 Indicações

O LED LA122 indica (acende) quando há transmissão de dados através da comunicação RS485.

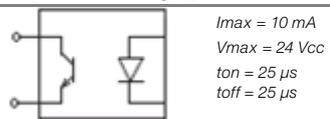
2.3.2 Características da interface RS485

- Interface segue o padrão EIA/TIA-485.
- Pode operar como escravo da rede WEGBus.
- Possibilita comunicação utilizando taxas de 9600 até 57600 Kbit/s.
- Interface isolada galvanicamente e com sinal diferencial, conferindo maior robustez contra interferência eletromagnética.
- Permite a conexão de até 32 dispositivos no mesmo segmento. Uma quantidade maior de dispositivos pode ser conectada com o uso de repetidores.²
- Comprimento máximo do barramento de 1000 metros.

2.3.3 Pinagem do Conector

A conexão para a interface RS485 está disponível através do conector XC1 utilizando a seguinte pinagem:

Tabela 2.3: Pinagem do conector para RS485 XA121

Pino	Função
1	Terra
2	RX_232
3	TX_232
4	Reservado <small>Erro! Indicador não definido.</small>
5	GND
6	
7	
8	A
9	B

¹ Não conectar os pinos reservados.

² O número limite de equipamentos que podem ser conectados na rede também depende do protocolo utilizado.

Tabela 2.4: Pinagem do conector para RS485 XA123

Pino	Função
1	NC
2	NC
3	NC
4	NC
5	GND
6	Reservado <small>Erro! Indicador não definido.</small>
7	NC
8	A (data -)
9	B (data +)
Caraça	Terra

2.3.4 Resistor de terminação

Para cada segmento da rede RS485, é necessário habilitar um resistor de terminação nos pontos extremos do barramento principal. O acessório ECO1 possui duas chaves DIP Switch que podem ser ativadas (colocando ambas as chaves SA121 na posição ON) para habilitar o resistor de terminação.


Figura 2.1: Resistor de terminação

2.3.5 Conexão com a Rede RS485

Para a ligação do servoconversor SCA06 utilizando a interface RS485, os seguintes pontos devem ser observados:

- É recomendado o uso de um cabo com par trançado blindado.
- Recomenda-se também que o cabo possua mais um fio para ligação do sinal de referência (GND). Caso o cabo não possua o fio adicional, deve-se deixar o sinal GND desconectado.
- A passagem do cabo deve ser feita separadamente (e se possível distante) dos cabos para alimentação de potência.
- Todos os dispositivos da rede devem estar devidamente aterrados, preferencialmente na mesma ligação com o terra. A blindagem do cabo também deve ser aterrada.
- Habilitar os resistores de terminação apenas em dois pontos, nos extremos do barramento principal, mesmo que existam derivações a partir do barramento.

3 PARAMETRIZAÇÃO

A seguir serão apresentados apenas os parâmetros do servoconversor SCA06 que possuem relação direta com a comunicação WEGBus.

3.1 SÍMBOLOS PARA DESCRIÇÃO DAS PROPRIEDADES

RW	Parâmetro de leitura e escrita
AC	Parâmetro visível na HMI somente quando o acessório correspondente esta conectado

P00650 – ENDEREÇO DO SERVOCONVERSOR NA COMUNICAÇÃO SERIAL 1 – RS232

P00656 – ENDEREÇO DO SERVOCONVERSOR NA COMUNICAÇÃO SERIAL 2 – RS485

Faixa de	1 a 247	Padrão: 1
Valores:		
Propriedades:	RW, AC	

Descrição:

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

- P00654/P00660 = 0 ou 1 (WEGBus ou WEGTP) → endereços válidos: 1 a 30.
- P00654/P00660 = 2 (Modbus RTU) → endereços válidos: 1 a 247.

P00652 – BIT RATE SERIAL 1 – RS232

P00658 – BIT RATE SERIAL 2 – RS485

Faixa de	0 = 4800 bits/s	Padrão: 1
Valores:	1 = 9600 bits/s	
	2 = 14400 bits/s	
	3 = 19200 bits/s	
	4 = 24000 bits/s	
	5 = 28800 bits/s	
	6 = 33600 bits/s	
	7 = 38400 bits/s	
	8 = 43200 bits/s	
	9 = 48000 bits/s	
	10 = 52800 bits/s	
	11 = 57600 bits/s	
Propriedades:	RW, AC	

Descrição:

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

P00653 – CONFIGURAÇÃO SERIAL 1 – RS232
P00659 – CONFIGURAÇÃO SERIAL 2 – RS485

Faixa de Valores:	0 = 8 bits de dados, sem paridade, 1 stop bit 1 = 8 bits de dados, paridade par, 1 stop bit 2 = 8 bits de dados, paridade ímpar, 1 stop bit 3 = 8 bits de dados, sem paridade, 2 stop bits 4 = 8 bits de dados, paridade par, 2 stop bits 5 = 8 bits de dados, paridade ímpar, 2 stop bits 6 = 7 bits de dados, sem paridade, 1 stop bit 7 = 7 bits de dados, paridade par, 1 stop bit 8 = 7 bits de dados, paridade ímpar, 1 stop bit 9 = 7 bits de dados, sem paridade, 2 stop bits 10 = 7 bits de dados, paridade par, 2 stop bits 11 = 7 bits de dados, paridade ímpar, 2 stop bits	Padrão: 3
Propriedades: RW, AC		

Descrição:

Permite a configuração do número de bits de dados, paridade e *stop* bits nos bytes da interface serial. Esta configuração deve ser a mesma para todos os equipamentos conectados na rede.

P00654 – SELECIONA PROTOCOLO SERIAL 1 – RS232
P00660 – SELECIONA PROTOCOLO SERIAL 2 – RS485

Faixa de Valores:	0 = WEGBus 1 = WEGTP 2 = Modbus RTU	Padrão: 2
Propriedades: RW		

Descrição:

Permite selecionar o protocolo desejado para a interface serial. A descrição detalhada do protocolo é feita no item 4 deste manual.

P00662 – AÇÃO PARA ERRO DE COMUNICAÇÃO

Faixa de Valores:	0 = Indica Alarme 1 = Causa Falha 2 = Causa alarme e Stop 3 = Causa alarme e desabilita	Padrão: 0
Propriedades: RW		

Descrição:

Este parâmetro permite selecionar qual a ação deve ser executada pelo equipamento, caso ele seja controlado via rede e um erro de comunicação seja detectado.

Tabela 3.1: Opções para o parâmetro P00662

Opção	Descrição
0 = Apenas indica alarme	Apenas mostra o código de alarme na HMI do servoconversor.
1 = Causa Falha	Causa falha e o servoconversor só volta a operar caso seja feito reset de falhas.
2 = Indica alarme e executa STOP	Será feita a indicação de alarme juntamente com a execução do comando STOP. Para que o servo saia desta condição, será necessário realizar o reset de falhas ou desabilitar o drive.
3 = Indica alarme e desabilita drive	Será feita a indicação de alarme juntamente com a execução do comando desabilita.

São considerados erros de comunicação os seguintes eventos:

Comunicação Serial (RS232/RS485):

- Alarme A00128/Falha F00028: *timeout* da interface serial.

P00663 – WATCHDOG SERIAL

Faixa de	0.0 a 999.0s	Padrão: 0.0
Valores:		
Propriedades:	RW	

Descrição:

Permite programar um tempo para a detecção de erro de comunicação via interface serial. Caso o inversor fique sem receber telegramas válidos por um tempo maior do que o programado neste parâmetro, será considerado que ocorreu um erro de comunicação, mostrado o alarme A00128 na HMI (ou falha F00028, dependendo da programação feita no P00662) e a ação programada no P00662 será executada.

Após energizado, o inversor começará a contar este tempo a partir do primeiro telegrama válido recebido. O valor 0,0 desabilita esta função.

P00664 – SALVA PARÂMETROS EM MEMÓRIA NÃO VOLÁTIL

Faixa de	0 = Não salva parâmetro na memória não volátil	Padrão: 1
Valores:	1 = Salva parâmetro na memória volátil	
Propriedades:	RW	

Descrição:

Permite selecionar se a escrita de parâmetros via serial deve ou não salvar o conteúdo dos parâmetros em memória não volátil (EEPROM). Quando utilizado o protocolo WEGBus ou Modbus é apenas esse parâmetro que determina se os parâmetros escritos via serial serão ou não salvos na memória não volátil. Porém, quando utilizado o protocolo WEGTP, deve-se observar que no byte de código do telegrama consta a informação sobre salvar ou não o parâmetro na EEPROM. Para que via WEGTP os mesmos sejam salvos em memória não volátil, é necessário que as duas informações, o byte de código do telegrama e o parâmetro P00664, sejam verdadeiras.



NOTA!

Este tipo de memória possui um numero limite de escritas (100.000 vezes). Dependendo da aplicação, este limite pode ser ultrapassado, caso alguns parâmetros sejam escritos ciclicamente via serial (referência de velocidade, torque, etc.). Nestes casos, pode ser desejado que, durante a operação do servoconversor, a escrita via serial não salve o conteúdo dos parâmetros em memória não volátil, para não ultrapassar o limite de escritas no servoconversor.



NOTA!

Esse parâmetro não se aplica quando a escrita e feita utilizando a interface USB.

P00667 – SALVA EM MARCADORES

Faixa de	0 = Lê e escreve normalmente o conteúdo no parâmetro correspondente	Padrão: 0
Valores:	1 = Lê e escreve conteúdo em marcadores de WORD volátil a partir do MW13000	
Propriedades:	RW	

Descrição:

Propriedade verificada quando parâmetro e escrito e lido via serial. Seleciona se quer que conteúdo a ser escrito/lido seja salvo em parâmetro ou em marcador de Word volátil.

**NOTA!**

Sendo este parâmetro $P00667 = 1$, ao escrever via serial no parâmetro $P00105 = 30$, o conteúdo do parâmetro será armazenado no marcador de Word 13105 ($MW_{inicial} + \text{Numero_par} \Rightarrow 13000 + 105$). Portanto, $MW13105 = 30$.

Observação: Uma vez que $P00667 = 1$, o mesmo não poderá ser alterado via serial. Pois na tentativa de escrever no parâmetro $P00667$ estará escrevendo no marcador de Word $P13667$.

4 PROTOCOLO WEGBUS

O protocolo WEGBus segue a norma ISO 1745 para transmissão de dados em código. São usadas somente seqüências de caracteres de texto sem cabeçalho.

São usados dois tipos de mensagens pelo mestre:

- Telegrama de leitura: para consulta do conteúdo dos parâmetros dos servoconversores;
- Telegrama de escrita: para alterar conteúdo dos parâmetros dos servoconversores.

Não é possível uma transmissão entre dois servoconversores. O mestre tem o controle do acesso ao barramento.

4.1 ENDEREÇAMENTO NO PROTOCOLO WEGBUS

Para o protocolo WEGBus, durante a transmissão de telegramas, o endereço selecionado no parâmetro endereço do servoconversor na comunicação serial é representado por um caractere ASCII, de acordo com a tabela a seguir:

Tabela 4.1: Endereço para o protocolo WEGBus

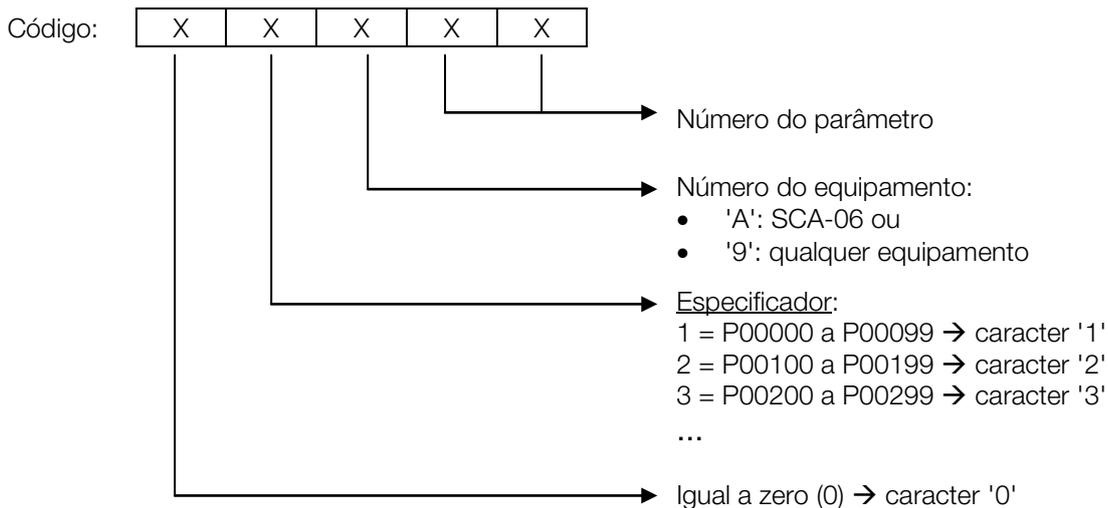
Endereço	ASCII	hexadecimal	Endereço	ASCII	hexadecimal
0	@	0x40	16	P	0x50
1	A	0x41	17	Q	0x51
2	B	0x42	18	R	0x52
3	C	0x43	19	S	0x53
4	D	0x44	20	T	0x54
5	E	0x45	21	U	0x55
6	F	0x46	22	V	0x56
7	G	0x47	23	W	0x57
8	H	0x48	24	X	0x58
9	I	0x49	25	Y	0x59
10	J	0x4A	26	Z	0x5A
11	K	0x4B	27	[0x5B
12	L	0x4C	28	\	0x5C
13	M	0x4D	29]	0x5D
14	N	0x4E	30	^	0x5E
15	O	0x4F	31	_	0x5F

Endereços para executar tarefas especiais:

- Endereço 0: qualquer servoconversor da rede é consultado, independentemente de seu endereço. Deve-se ter apenas um servoconversor ligado a rede (ponto-a-ponto) para que não ocorram curto-circuitos nas linhas de interface.
- Endereço 31: um comando pode ser transmitido simultaneamente para todos os servoconversores da rede, sem reconhecimento de aceitação.

4.2 CAMPOS DO PROTOCOLO

- **STX**: Byte de "Start of Transmission": Valor: 0x02 (hexadecimal); 2 (decimal).
- **ETX**: Byte de "End of Transmission": Valor: 0x03 (hexadecimal); 3 (decimal).
- **ADR**: Byte do endereço do servoconversor na rede
Faixa de Valores: 0x41 (hexadecimal); 65 (decimal); 'A' (ASCII) ... 0x5E (hexadecimal); 94 (decimal); '^' (ASCII) → Representam os valores de 1 ... 30 programados no parâmetro do endereço do servoconversor.
Especial 1: 0x40 (hexadecimal); 64 (decimal); '@' (ASCII) → Permite escrita ou leitura de todos os equipamentos conectados a rede.
Especial 2: 0x5F (hexadecimal); 95 (decimal); '_' (ASCII) → Permite SOMENTE escrita em todos os equipamentos conectados a rede sem resposta de aceitação ou rejeição.
- **CÓDIGO**: Contém o endereço do parâmetro com 5 dígitos (todos em ASCII) de acordo com o seguinte:



- **VAL:** Valor em 4 dígitos hexadecimais (cada um dos 4 dígitos corresponde a um número de 0x00 a 0x0F, que concatenados formam o valor em hexadecimal que é transmitido via serial).
- **BCC:** Byte de Checksum longitudinal do telegrama, ou seja, OU EXCLUSIVO entre todos os bytes do telegrama. Tamanho de 1 byte (0x00 ... 0xFF hexadecimal).
- **ACK:** Byte de aceitação do escravo após uma escrita do mestre
Valor: 0x06 (hexadecimal); 6 (decimal).
- **NAK:** Byte de rejeição do escravo após uma leitura ou escrita do mestre. Pode ocorrer quando o mestre solicita uma escrita ou leitura de um parâmetro inexistente ou o valor a ser escrito no parâmetro está fora da faixa de valores permitida,
Valor: 0x15 (hexadecimal); 21 (decimal).

4.3 FORMATO DOS TELEGRAMAS

A seguir serão apresentados os formatos dos telegramas de leitura e escrita em parâmetros. Telegramas que possuem erro no formato serão ignorados pelo servoconversor, que não enviará resposta para o mestre.



NOTA!

O tempo de escrita na EEPROM é de 10ms por parâmetro, portanto é necessário cuidar-se para não sobrecarregar o servoconversor com muitos telegramas seguidos pois isto pode fazer com que o servoconversor ignore os últimos telegramas de modo a ter tempo de escrever todos parâmetros na EEPROM (quando isto ocorre o servoconversor indica alarme 107).

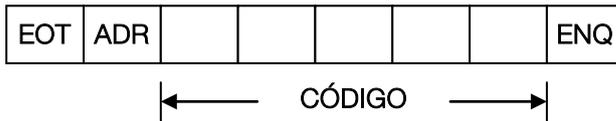
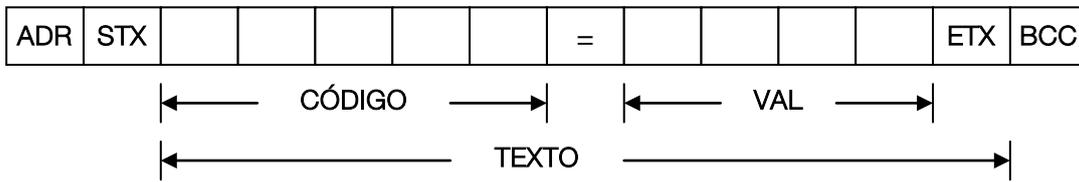


NOTA!

O número de escritas na memória EEPROM limita a vida útil desta, portanto é recomendável não se salvar na EEPROM parâmetros que são escritos muitas vezes por dia, o usuário deve salvar na EEPROM apenas aqueles parâmetros em que isto é realmente necessário.

4.3.1 Telegrama de leitura

Este telegrama permite que o mestre receba do servoconversor o conteúdo correspondente ao código da solicitação. No telegrama de resposta o servoconversor transmite os dados solicitados pelo mestre.

MESTRE:

SERVOCONVERSOR:


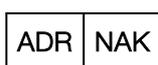
Formato do telegrama de leitura

- **EOT**: caracter de controle - End Of Transmission (valor 0x04);
- **ADR**: endereço do servoconversor - 1 caracter ASCII (ver Tabela 4.1);
- **CÓDIGO**: endereço da variável - 5 caracteres ASCII;
- **ENQ**: caracter de controle ENQuiry (solicitação) (valor 0x05);

Formato do telegrama de resposta do servoconversor

- **ADR**: endereço do servoconversor - 1 caracter ASCII (ver Tabela 4.1);
- **STX**: caracter de controle - Start of TeXt (valor 0x02);
- **TEXTO**: consiste em:
 - **CÓDIGO**: endereço da variável - 5 caracteres;
 - **"=**": caracter da separação (caracter em ASCII - valor 0x3E);
 - **VAL**: valor em 4 dígitos HEXADECIMAIS (cada um dos 4 dígitos corresponde a um número de 0x00 a 0x0F, que concatenados formam o valor em hexadecimal que é transmitido via serial);
 - **ETX**: caracter de controle - End of TeXt (valor 0x03);
- **BCC**: Byte de checksum - EXCLUSIVE OR de todos os bytes entre STX (excluído) e ETX (incluído), ou seja, de todos os bytes pertencentes ao TEXTO.

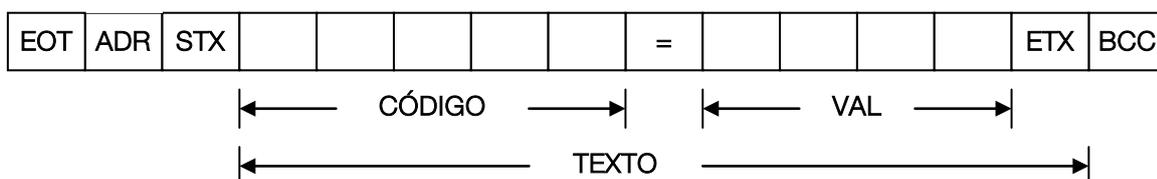
Em caso de erro no tratamento do telegrama, poderá haver uma resposta do servoconversor com:



NOTA: Se o mestre enviar o endereço "@" (ver item 4.1) no telegrama ADR, o servoconversor responderá o valor do endereço que o servoconversor se encontra na rede no telegrama ADR.

4.3.2 Telegrama de escrita

Este telegrama envia dados para os parâmetros dos servoconversores. O servoconversor irá responder indicando se os dados foram aceitos ou não.

MESTRE:

SERVOCONVERSOR:


Formato do telegrama de escrita

- **EOT**: caracter de controle - End Of Transmission (valor 0x04);
- **ADR**: endereço do servoconversor - 1 caracter ASCII (ver Tabela 4.1);
- **STX**: caracter de controle - Start of TeXt (valor 0x02);
- **TEXTO**: consiste em:
 - **CÓDIGO**: endereço da variável - 5 caracteres ASCII;
 - **"="**: caracter da separação (caracter em ASCII - valor 0x3E);
 - **VAL**: valor em 4 dígitos HEXADECIMAIS (cada um dos 4 dígitos corresponde a um número de 0x00 a 0x0F, que concatenados formam o valor em hexadecimal que é transmitido via serial);
 - **ETX**: caracter de controle - End of TeXt (valor 0x03);
- **BCC**: Byte de checksum - EXCLUSIVE OR de todos os bytes entre STX (excluído) e ETX (incluído), ou seja, de todos os bytes pertencentes ao TEXTO.

Formato do telegrama de resposta do servoconversor

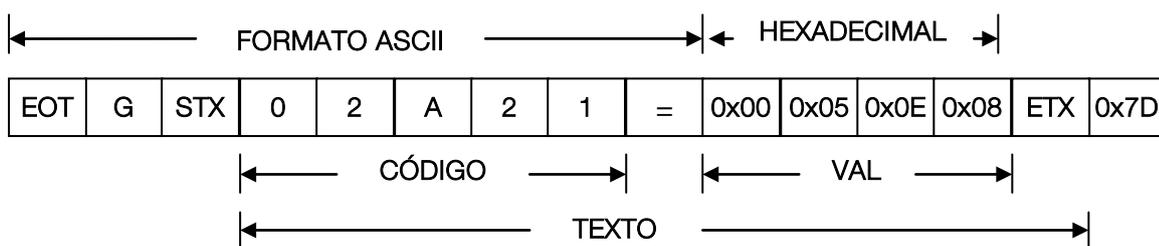
- **Telegrama aceito pelo servoconversor**:
 - **ADR**: endereço do servoconversor - 1 caracter ASCII (ver Tabela 4.1);
 - **ACK**: caracter de controle - ACKnowledge (valor 0x06);
- **Telegrama não aceito pelo servoconversor**:
 - **ADR**: endereço do servoconversor - 1 caracter ASCII (ver Tabela 4.1);
 - **NAK**: caracter de controle - Not Acknowledge (valor 0x15). Isso significa que os dados não foram aceitos e a variável endereçada permanece com o seu valor antigo.

NOTA: Se o mestre enviar o endereço "@" (ver item 4.1) no telegrama ADR, o servoconversor responderá o valor do endereço que o servoconversor se encontra na rede no telegrama ADR.

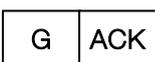
4.4 EXEMPLO DE TELEGRAMAS UTILIZANDO O PROTOCOLO WEGBUS

Exemplo 1: alteração de P00121, supondo que quero P00121 = 1512 rpm (1512 = 0x05E8) e que o endereço do servoconversor na rede é 7 (P00650 = 7).

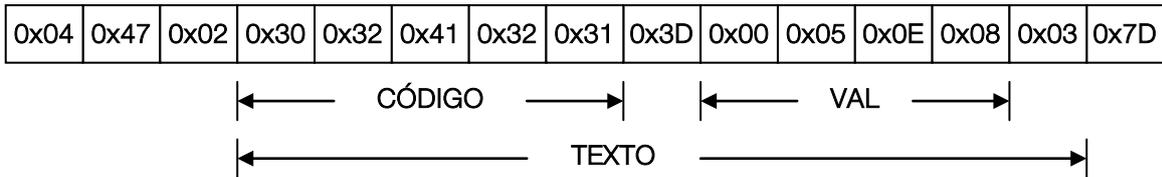
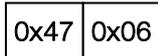
MESTRE:



SERVOCONVERSOR:

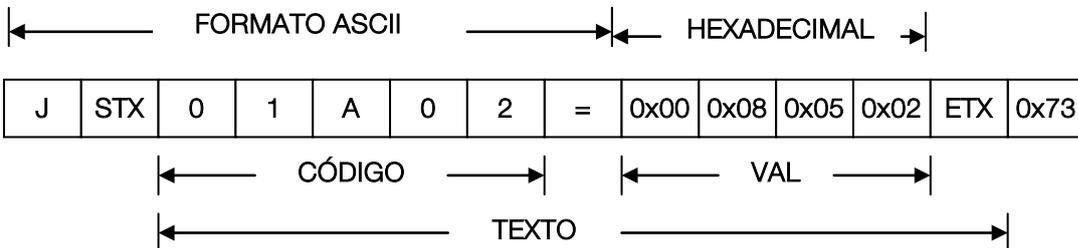


Ou o mesmo telegrama, mas representado todo no formato hexadecimal:

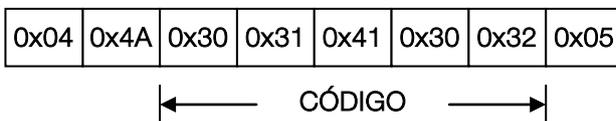
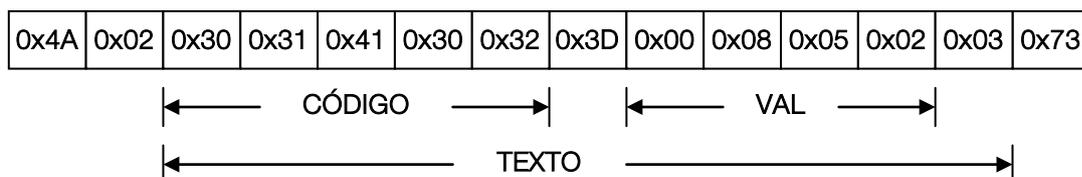
MESTRE:

SERVOCONVERSOR:


Exemplo 2: leitura de P00002, supondo P00002 = 2130 rpm (2130 = 0x0852) no momento da consulta, e que o endereço do servoconversor na rede seja o 10 (P00650 = 10).

MESTRE:

SERVOCONVERSOR:


Ou o mesmo telegrama, mas representado todo no formato hexadecimal:

MESTRE:

SERVOCONVERSOR:


5 FALHAS E ALARMES RELACIONADOS COM A COMUNICAÇÃO WEGBUS

A00128/F00028 – TIMEOUT NA RECEPÇÃO DE TELEGRAMAS

Descrição:

Alarme que indica falha na comunicação serial. Indica que o equipamento parou de receber telegramas seriais válidos por um período maior do que o programado no P00663.

Atuação:

O parâmetro P00663 permite programar um tempo dentro do qual o servoconversor deverá receber ao menos um telegrama válido via interface serial RS485 – com endereço e campo de checagem de erros corretos – caso contrário será considerado que houve algum problema na comunicação serial. A contagem do tempo é iniciada após a recepção do primeiro telegrama válido. Esta função pode ser utilizada para qualquer protocolo serial suportado pelo servoconversor.

Depois de identificado o timeout na comunicação serial, será sinalizada através da HMI a mensagem de alarme A00128 – ou falha F00028, dependendo da programação feita no P00662. Para alarmes, caso a comunicação seja restabelecida, a indicação do alarme será retirada da HMI.

Possíveis Causas/Correção:

- Verificar instalação da rede, cabo rompido ou falha/mal contato nas conexões com a rede, aterramento.
- Garantir que o mestre envie telegramas para o equipamento sempre em um tempo menor que o programado no P00663.
- Desabilitar esta função no P00663.

I. APÊNDICES

APÊNDICE A. TABELA ASCII

Tabela I.1: Caracteres ASCII

Dec	Hex	Chr	Dec	Hex	Chr	Dec	Hex	Chr	Dec	Hex	Chr
0	00	NUL (Null char.)	32	20	Sp	64	40	@	96	60	`
1	01	SOH (Start of Header)	33	21	!	65	41	A	97	61	a
2	02	STX (Start of Text)	34	22	"	66	42	B	98	62	b
3	03	ETX (End of Text)	35	23	#	67	43	C	99	63	c
4	04	EOT (End of Transmission)	36	24	\$	68	44	D	100	64	d
5	05	ENQ (Enquiry)	37	25	%	69	45	E	101	65	e
6	06	ACK (Acknowledgment)	38	26	&	70	46	F	102	66	f
7	07	BEL (Bell)	39	27	'	71	47	G	103	67	g
8	08	BS (Backspace)	40	28	(72	48	H	104	68	h
9	09	HT (Horizontal Tab)	41	29)	73	49	I	105	69	i
10	0A	LF (Line Feed)	42	2A	*	74	4A	J	106	6A	j
11	0B	VT (Vertical Tab)	43	2B	+	75	4B	K	107	6B	k
12	0C	FF (Form Feed)	44	2C	,	76	4C	L	108	6C	l
13	0D	CR (Carriage Return)	45	2D	-	77	4D	M	109	6D	m
14	0E	SO (Shift Out)	46	2E	.	78	4E	N	110	6E	n
15	0F	SI (Shift In)	47	2F	/	79	4F	O	111	6F	o
16	10	DLE (Data Link Escape)	48	30	0	80	50	P	112	70	p
17	11	DC1 (Device Control 1)	49	31	1	81	51	Q	113	71	q
18	12	DC2 (Device Control 2)	50	32	2	82	52	R	114	72	r
19	13	DC3 (Device Control 3)	51	33	3	83	53	S	115	73	s
20	14	DC4 (Device Control 4)	52	34	4	84	54	T	116	74	t
21	15	NAK (Negative Acknowledgement)	53	35	5	85	55	U	117	75	u
22	16	SYN (Synchronous Idle)	54	36	6	86	56	V	118	76	v
23	17	ETB (End of Trans. Block)	55	37	7	87	57	W	119	77	w
24	18	CAN (Cancel)	56	38	8	88	58	X	120	78	x
25	19	EM (End of Medium)	57	39	9	89	59	Y	121	79	y
26	1A	SUB (Substitute)	58	3A	:	90	5A	Z	122	7A	z
27	1B	ESC (Escape)	59	3B	;	91	5B	[123	7B	{
28	1C	FS (File Separator)	60	3C	<	92	5C	\	124	7C	
29	1D	GS (Group Separator)	61	3D	=	93	5D]	125	7D	}
30	1E	RS (Record Separator)	62	3E	>	94	5E	^	126	7E	~
31	1F	US (Unit Separator)	63	3F	?	95	5F	_	127	7F	DEL



WEG Drives & Controls - Automação LTDA.
Jaraguá do Sul – SC – Brazil
Phone 55 (47) 3276-4000 – Fax 55 (47) 3276-4020
São Paulo – SP – Brazil
Phone 55 (11) 5053-2300 – Fax 55 (11) 5052-4212
automacao@weg.net
www.weg.net