

Driver Ablerex EnerSolis

Nome do Arquivo	EnerSolis.dll
Fabricante	Ablerex Electronics Co., Ltd.
Equipamentos	EnerSolis ES Séries 6000, 8000, 10000, 12000 e 25600
Protocolo	JBus
Versão	1.0.1
Última Atualização	24/08/2022
Plataforma	Win32
Dependências	IOKit v2.00 ou superior
Leitura com Superblocos	Não
Nível	0

Introdução

Este Driver implementa o protocolo JBus, permitindo a uma aplicação desenvolvida pela **Elipse Software** comunicar com os equipamentos EnerSolis ES Séries 6000, 8000, 10000, 12000 e 25600 da Ablerex Electronics Co., Ltd.

Configuração do Driver

Os parâmetros [P] de configuração deste Driver não são utilizados.

Todas as configurações devem ser realizadas na caixa de diálogo de propriedades deste Driver no **E3** ou pela opção **Extra Settings** no **Elipse SCADA**.

Nesta janela de propriedades, a aba **EnerSolis** contém as configurações específicas deste Driver. As demais abas se referem às configurações de comunicação da biblioteca **IOKit** da **Elipse Software**.

Para mais informações sobre a configuração da biblioteca **IOKit**, consulte o tópico **Documentação das Interfaces de Comunicação**.

Configuração das Propriedades

Este tópico contém informações sobre as propriedades disponíveis na aba **EnerSolis**, já incluindo o valor das **Strings** de propriedades *offline*, que podem ser programadas pelo usuário ao iniciar uma aplicação em modo **Offline**.

Para aplicações **E3** ou **Elipse Power**, o valor destas configurações também pode ser definido em tempo de execução. Para isto, selecione a opção **Start driver OFFLINE** na aba **Setup** da janela de propriedades para iniciar este Driver em modo **Offline**.

As opções de configuração deste Driver estão listadas na tabela a seguir.

Opções de configuração do Driver Ablerex EnerSolis

ABA	PARÂMETRO	STRING OFFLINE	TIPO DE DADOS	DESCRIÇÃO
Setup	Physical Layer	IO.Type	Texto	Por padrão, utilize a opção Ethernet
	Timeout	IO.TimeoutMs	Número	Limite de tempo, em milissegundos, para

ABA	PARÂMETRO	STRING OFFLINE	TIPO DE DADOS	DESCRIÇÃO
				recepção dos dados na resposta do equipamento. Por exemplo, o valor 1000 define um limite de um segundo
Serial	Port	IO.Serial.Port	Número	Por padrão, utilize o mesmo valor configurado no dispositivo. Os valores possíveis são 1 : COM1, 2 : COM2, 3 : COM3, ..., n : COMn
	Baud rate	IO.Serial.Baudrate	Número	Por padrão, utilize o mesmo valor configurado no dispositivo. Por exemplo, o valor 9600 define a taxa de <i>baud rate</i> da comunicação serial
	Data bits	IO.Serial.DataBits	Número	Por padrão, utilize o mesmo valor configurado no dispositivo. Os valores possíveis são 7 : 7 bits ou 8 : 8 bits
	Parity	IO.Serial.Parity	Número	Por padrão, utilize o mesmo valor configurado no dispositivo. Os valores possíveis são 0 : No Parity, 1 : Odd Parity, 2 : Even Parity, 3 : Mark Parity ou 4 : Space Parity
	Stop bits	IO.Serial.StopBits	Número	Por padrão, utilize o mesmo valor configurado no dispositivo. Os valores possíveis são 0 : Stop Bit 1, 1 : Stop Bit 1.5 ou 2 : Stop Bit 2
EnerSolis	Use Default Slave Address	EnerSolis.UseDefaultSlaveAddress	Número	Ativa ou desativa o uso de <i>Default Slave Address</i> . Os valores possíveis são 0 : Desativado ou 1 : Ativado
	Default Slave Address	EnerSolis.DefaultSlaveAddress	Número	Endereço (ID) do equipamento a ser utilizado

ABA	PARÂMETRO	STRING OFFLINE	TIPO DE DADOS	DESCRIÇÃO
				automaticamente por este Driver. Os valores válidos estão no intervalo entre 1 (um) e 255

Todas as propriedades *offline* devem ser configuradas via Tags PLC no formato **String** usando os parâmetros *N1* igual a -1 (menos um), *N2* igual a 0 (zero), *N3* igual a 0 (zero) e *N4* igual a 3 (três). Para mais detalhes e exemplos, consulte o tópico **Documentação das Interfaces de Comunicação**.

Referência de Tags

Esta seção contém informações sobre a configuração dos Tags **[N/B]** deste Driver.

Leitura de Alarmes

Somente Leitura

Utilize um Tag PLC ou Bloco para leitura dos alarmes de um equipamento, utilizando os parâmetros descritos na tabela a seguir.

Parâmetros para leitura de alarmes

N1/B1	Endereço (ID) de um equipamento
N2/B2	1000
N3/B3	Não utilizado
N4/B4	Número do alarme, entre 0 (zero) e 31

Um equipamento disponibiliza a leitura de 32 alarmes, numerados de 0 (zero) a 31, com valor de status em **1** (um) para Ligado ou **0** (zero) para Desligado.

Leitura de Múltiplos Alarmes Utilizando um Tag Bloco

Utilize um Tag Bloco para ler múltiplos alarmes, em que cada Elemento representa, sequencialmente, o status de um alarme, iniciando pelo número do alarme definido no parâmetro **B4**. A tabela a seguir contém alguns exemplos de utilização de Tags Bloco.

Exemplos de utilização de Tags Bloco

OBJETIVO	ELEMENTOS	B1	B2	B3	B4 (ALARME INICIAL)
Leitura de todos os 32 alarmes	32	ID do equipamento	1000	Não utilizado	0 (zero)
Leitura dos 10 primeiros alarmes	10	ID do equipamento	1000	Não utilizado	0 (zero)

OBJETIVO	ELEMENTOS	B1	B2	B3	B4 (ALARME INICIAL)
Leitura dos 10 últimos alarmes	10	ID do equipamento	1000	Não utilizado	22
Leitura de 5 (cinco) alarmes, a partir do alarme 10	5 (cinco)	ID do equipamento	1000	Não utilizado	10

Leitura de um Alarme Simples Utilizando um Tag PLC

Utilize um Tag PLC para ler, individualmente, o status do número de alarme, entre 0 (zero) e 31, definido pelo parâmetro **N4**. A tabela a seguir contém alguns exemplos de utilização de Tags PLC.

Exemplos de utilização de Tags PLC

OBJETIVO	N1	N2	N3	N4 (NÚMERO DO ALARME)
Leitura do alarme número 14	ID do equipamento	1000	Não utilizado	14
Leitura do alarme número 30	ID do equipamento	1000	Não utilizado	30

NOTA

Todos os alarmes estão armazenados em dois **Words**, nos registradores **0xC000** e **0xC001** da memória de um equipamento.

Leitura de Erros

Somente Leitura

Utilize um Tag PLC ou Bloco para leitura de erros de um equipamento, utilizando os parâmetros descritos na tabela a seguir.

Parâmetros para leitura de erros

N1/B1	Endereço (ID) de um equipamento
N2/B2	2000
N3/B3	Não utilizado
N4/B4	Número do erro, entre (0) e 47

Um equipamento disponibiliza a leitura de 48 erros, numerados entre 0 (zero) e 47, com valor de status em **1** (um) para Ligado ou **0** (zero) para Desligado.

Leitura de Múltiplos Erros Utilizando um Tag Bloco

Utilize um Tag Bloco para ler múltiplos erros, em que cada Elemento representa, sequencialmente, o status de um erro, iniciando pelo número do erro definido no parâmetro **B4**. A tabela a seguir contém alguns exemplos de utilização de Tags Bloco.

Exemplos de utilização de Tags Bloco

OBJETIVO	ELEMENTOS	B1	B2	B3	B4 (ERRO INICIAL)
Leitura de todos os 48 erros	48	ID do equipamento	2000	Não utilizado	0 (zero)
Leitura dos 10 primeiros erros	10	ID do equipamento	2000	Não utilizado	0 (zero)
Leitura dos 10 últimos erros	10	ID do equipamento	2000	Não utilizado	38
Leitura de 5 (cinco) erros, a partir do erro 10	5 (cinco)	ID do equipamento	2000	Não utilizado	10

Leitura de um Erro Simples Utilizando um Tag PLC

Utilize um Tag PLC para ler, individualmente, o status do número de erro, entre 0 (zero) e 47, definido pelo parâmetro **N4**. A tabela a seguir contém alguns exemplos de utilização de Tags PLC.

Exemplos de utilização de Tags PLC

OBJETIVO	N1	N2	N3	N4 (NÚMERO DO ERRO)
Leitura do erro número 18	ID do equipamento	2000	Não utilizado	18
Leitura do erro número 45	ID do equipamento	2000	Não utilizado	45

NOTA

Todos os erros estão armazenados em três **Words**, nos registradores **0xC010**, **0xC011** e **0xC012** da memória de um equipamento.

Leitura dos Valores de Energia

Somente Leitura

Utilize um Tag PLC ou Bloco para leitura dos valores de energia de um equipamento, utilizando os parâmetros descritos na tabela a seguir.

Parâmetros para leitura dos valores de energia

N1/B1	Endereço (ID) de um equipamento
N2/B2	3000
N3/B3	Não utilizado
N4/B4	Número do valor de energia, entre 0 (zero) e 31

Um equipamento disponibiliza a leitura de 32 valores de energia, numerados entre 0 (zero) e 31, conforme descrito na tabela a seguir.

Valores de energia

N4/B4	VALOR DE ENERGIA
0 (zero)	Output power (kW)
1 (um)	AC voltage phase L1 (V)
2 (dois)	AC voltage phase L2 (V)
3 (três)	AC voltage phase L1-L2 (V)
4 (quatro)	AC output current L1 (A)
5 (cinco)	AC output current L2 (A)
6 (seis)	AC frequency L1 (Hz)
7 (sete)	DC-Bus Positive-voltage (V)
8 (oito)	DC-Bus Negative-voltage (V)
9 (nove)	Inverter internal temperature (°C)
10	Inverter Heat sink temperature (°C)
11	DC1 input voltage (V)
12	DC2 input voltage (V)
13	DC1 input current (A)
14	DC2 input current (A)
15	Input Power A (kW)
16	Input Power B (kW)
17	Total Output Power (kWh)
18	Reserve
19	Reserve
20	Reserve
21	Reserve
22	Self Test Vmin
23	Self Test Vmax
24	Self Test Fmin
25	Self Test Fmax
26	AC voltage phase L2-L3 (V)

N4/B4	VALOR DE ENERGIA
27	AC frequency L2 (Hz)
28	AC voltage phase L3 (V)
29	AC voltage phase L3-L1 (V)
30	AC frequency L3 (Hz)
31	AC output current L3 (A)

Leitura de Múltiplos Valores de Energia Utilizando Tags Bloco

Utilize um Tag Bloco para ler múltiplos valores de energia, em que cada Elemento representa, sequencialmente, os valores de medições de energia, iniciando pelo número do valor de energia definido pelo parâmetro **B4**. A tabela a seguir contém alguns exemplos de utilização de Tags Bloco.

Exemplos de utilização de Tags Bloco

OBJETIVO	ELEMENTOS	B1	B2	B3	B4 (VALOR DE ENERGIA INICIAL)
Leitura de todos os 32 valores de energia	32	ID do equipamento	3000	Não utilizado	0 (zero)
Leitura dos 10 primeiros valores de energia	10	ID do equipamento	3000	Não utilizado	0 (zero)
Leitura dos 10 últimos valores de energia	10	ID do equipamento	3000	Não utilizado	22
Leitura de 5 (cinco) valores de energia, a partir do valor de energia 10	5 (cinco)	ID do equipamento	3000	Não utilizado	10

Leitura de um Valor Simples de Energia utilizando Tags PLC

Utilize um Tag PLC para ler, individualmente, o número do valor de energia, entre 0 (zero) e 31, definido pelo parâmetro **N4**. A tabela a seguir contém alguns exemplos de utilização de Tags PLC.

Exemplos de utilização de Tags PLC

OBJETIVO	N1	N2	N3	N4 (NÚMERO DO VALOR DE ENERGIA)
Leitura do valor de energia número 14	ID do equipamento	3000	Não utilizado	14
Leitura do valor de energia número 30	ID do equipamento	3000	Não utilizado	30

NOTA

Todos os valores de energia estão armazenados em 96 **Words**, do registrador **0xC020** ao **0xC07F** da memória de um equipamento.

Configurações de um Equipamento

Leitura e Escrita

Utilize Tags PLC ou Bloco para leitura ou escrita dos valores de configuração de um equipamento, utilizando os parâmetros descritos na tabela a seguir.

Parâmetros para leitura ou escrita dos valores de configuração

N1/B1	Endereço (ID) de um equipamento
N2/B2	Consulte a tabela Comandos de configuração
N3/B3	Não utilizado
N4/B4	Não utilizado

A tabela a seguir contém os comandos que se pode utilizar para configurar um equipamento.

Comandos de configuração

N2/B2	COMANDO	REGISTER	CODE
6019	Power Factor P1-P2	0xCF13	C19
6020	Power Factor PF0-PF1	0xCF14	C20
6021	Power Factor PF2-PF3	0xCF15	C21
6024	Vpset	0xC098	C24
6025	Reduce Power Rate	0xCF16	C25
6029	Enable or Disable Vpset	0xC09D	C29

NOTA

Para escrever dados via Tags Bloco, todos os Elementos devem ser enviados em uma única escrita em bloco.

Leitura ou Escrita para Power Factor P1-P2

Utilize um Tag Bloco com 2 (dois) Elementos para ler ou escrever valores, conforme os dados a seguir:

- **TagBloco.Elemento1:** Power Factor P1 (valores de 0~100, referente a 0%~100%)
- **TagBloco.Elemento2:** Power Factor P2 (valores de 0~100, referente a 0%~100%)

Leitura ou Escrita para Power Factor PF0-PF1

Utilize um Tag Bloco com 2 (dois) Elementos para ler ou escrever valores, conforme os dados a seguir:

- **TagBloco.Elemento1:** Power Factor PF0, conforme a tabela de exemplos a seguir
- **TagBloco.Elemento2:** Power Factor PF1, conforme a tabela de exemplos a seguir

Exemplos para Power Factor PF0-PF1

VALOR	POWER FACTOR
125	0.75 (adiantado)
110	0.90 (adiantado)
100	1 (um)
90	0.90 (atrasado)
75	0.75 (atrasado)

Leitura ou Escrita para Power Factor PF2-PF3

Utilize um Tag Bloco com 2 (dois) Elementos para ler ou escrever valores, conforme os dados a seguir:

- **TagBloco.Elemento1:** Power Factor PF2, conforme a tabela de exemplos a seguir
- **TagBloco.Elemento2:** Power Factor PF3, conforme a tabela de exemplos a seguir

Exemplos para Power Factor PF2-PF3

VALOR	POWER FACTOR
125	0.75 (adiantado)
110	0.90 (adiantado)
100	1 (um)
90	0.90 (atrasado)
75	0.75 (atrasado)

Leitura ou Escrita para Vpset

Utilize um Tag PLC para ler ou escrever valores, conforme os dados a seguir:

- **TagPLC:** Vpset (valores de 231~240, referente a 231Vac~240Vac)

Leitura ou Escrita para Reduce Power Rate

Utilize um Tag PLC para ler ou escrever valores, conforme os dados a seguir:

- **TagPLC:** Reduce Power Rate (valores de 10~100, referente a 10%~100%)

Leitura ou Escrita para Enable ou Disable Vpset

Utilize um Tag PLC para ler ou escrever valores, conforme os dados a seguir:

- **TagPLC:** Enable/Disable Vpset (**1:** Enable ou **0:** Disable)

Tag Genérico

Leitura e Escrita

Utilize Tags PLC ou Bloco para leitura ou escrita de valores diretamente no endereço de memória dos registradores de um equipamento, utilizando os parâmetros descritos na tabela a seguir.

Parâmetros para leitura ou escrita nos registradores de memória

N1/B1	Endereço (ID) de um equipamento
N2/B2	Consulte a tabela Comandos para manipular dados em registradores de memória
N3/B3	Não utilizado
N4/B4	Memória do registrador

A tabela a seguir contém os comandos que se pode utilizar para manipular dados nos registradores de memória de um equipamento.

Comandos para manipular dados em registradores de memória

N2/B2	COMANDO	REGISTRADOR
310	Leituras com FCode 0x03 e escritas com FCode 0x10	N4/B4
364	Leituras com FCode 0x03 e escritas com FCode 0x64	N4/B4

Qualquer registrador de memória de um equipamento pode ser lido via FCode 0x03. Alguns registradores podem ser modificados pelo FCode 0x10 e outros, mais restritos, pelo FCode 0x64.

Para leitura ou escrita via Tags Bloco, cada um dos Elementos representa um registrador de memória sequencial, partindo do endereço inicial definido no parâmetro **B4**, conforme os exemplos da tabela a seguir.

Exemplos de utilização de Tags Bloco

OBJETIVO	ELEMENTOS	B1	B2	B3	B4 (REGISTRADOR INICIAL)	RESULTADO
Leitura com Tag Bloco (FCode 0x03)	5 (cinco)	ID do equipamento	310	Não utilizado	10	Leitura de 5 (cinco) registradores, do 10 ao 14
Escrita com Tag Bloco (FCode 0x10)	3 (três)	ID do equipamento	310	Não utilizado	32	Escrita de 3 (três) registradores, do 32 ao 34

Cada registrador de memória pode armazenar um valor de 16 bits (**Word**). Consulte o manual do usuário do equipamento para verificar a tabela de registradores disponíveis para leitura e escrita de valores.


Documentação das Interfaces de Comunicação

Esta seção contém a documentação das Interfaces de Comunicação referente ao Driver **EnerSolis**.

Configurações do Driver

A configuração das Interfaces de Comunicação é realizada na caixa de diálogo de configuração do Driver. Para acessar a configuração da caixa de diálogo no **E3** (versão 1.0), siga estes passos:

1. Clique com o botão direito do mouse no objeto Driver (IODriver).
2. Selecione o item **Propriedades** no menu contextual.
3. Selecione a aba **Driver**.
4. Clique em **Outros parâmetros**.

No **E3** versão 2.0 ou posterior, clique em **Configurar o driver**  na barra de ferramentas do Driver. No **Elipse SCADA**, siga estes passos:

1. Abra o Organizer.
2. Selecione o Driver na árvore do Organizer.
3. Clique em **Extras** na aba **Driver**.

Atualmente, as Interfaces de Comunicação permitem que apenas uma conexão seja aberta para cada Driver. Isto significa que, se for necessário o acesso a duas portas seriais, é preciso adicionar dois Drivers na aplicação e configurar cada um destes Drivers para cada porta serial.

Caixa de Diálogo de Configuração

A caixa de diálogo das Interfaces de Configuração permitem configurar a conexão de I/O que é utilizada pelo Driver. Esta caixa de diálogo contém as abas **Setup**, **Serial**, **Ethernet**, **Modem** e **RAS**, descritas nos tópicos a seguir. Se um Driver não implementa uma conexão de I/O específica, a respectiva aba não está disponível para configuração. Alguns Drivers podem conter abas adicionais, específicas para cada Driver, na caixa de diálogo de configuração.

Aba Setup

A aba **Setup** contém a configuração geral de um Driver. A aba é dividida nos seguintes grupos:

- **Configurações gerais:** Configurações da camada física do Driver, *time-out* e modo de inicialização
- **Connection management:** Configurações de como a Interface de Comunicação mantém a conexão e qual a política de recuperação em caso de falha
- **Logging options:** Controla a geração dos arquivos de log

The screenshot shows the 'Setup' tab of a driver configuration window. It contains several sections:

- Physical Layer:** A dropdown menu set to 'Serial'.
- Start driver OFFLINE:** An unchecked checkbox.
- Timeout:** A text input field containing '1000' followed by 'ms'.
- Connection management:** A sub-section containing:
 - Mode:** A dropdown menu set to 'Automatic (managed by the driver)'.
 - Retry failed connection every:** A checked checkbox followed by a text input '20' and the word 'seconds'.
 - Give up after:** An unchecked checkbox followed by a text input '1' and the words 'failed retries'.
 - Disconnect if non-responsive for:** An unchecked checkbox followed by a text input '0' and the word 'seconds'.
- Logging Options:** A sub-section containing:
 - Log to File:** An unchecked checkbox followed by a text input field containing the path 'C:\eeLogs\Modbus_%.DATE%.log'.

Aba Setup

Opções gerais da aba Setup

OPÇÃO	DESCRIÇÃO
Physical Layer	Selecione a interface física em uma lista. As opções são Serial, Ethernet, Modem e RAS . A interface selecionada deve ser configurada na sua aba específica
Timeout	Configure o <i>time-out</i> , em milissegundos, para a camada física. Esta é a medida de tempo que a interface de I/O aguarda para a recepção de um byte qualquer do <i>buffer</i> de recepção
Start driver OFFLINE	Selecione esta opção para que o Driver inicie em modo Offline ou parado. Isto significa que a interface de I/O não é criada até que se configure o Driver em modo Online utilizando-se um Tag na aplicação. Este modo possibilita a configuração dinâmica da interface de I/O em tempo de execução

Opções para o grupo Connection management

OPÇÃO	DESCRIÇÃO
Mode	Seleciona o modo de gerenciamento de conexão. Selecionar a opção Automatic permite que o Driver gerencie a conexão automaticamente, como especificado nas opções seguintes. Selecionar a opção Manual permite que a aplicação gerencie a conexão completamente. Consulte o tópico Estados do Driver para mais informações

OPÇÃO	DESCRIÇÃO
Retry failed connection every ... seconds	Selecione esta opção para habilitar a retentativa de conexão do Driver em um determinado intervalo, em segundos. Se a opção Give up after failed retries não está selecionada, o Driver continua retentando até que a conexão seja efetuada, ou que a aplicação seja parada
Give up after ... failed retries	Habilite esta opção para definir um número máximo de retentativas de conexão. Quando o número especificado de tentativas consecutivas de reconexão for atingido, o Driver vai para o modo Offline , assumindo que um problema de hardware foi detectado. Se o Driver estabelece uma conexão com sucesso, o número de retentativas sem sucesso é zerado. Se esta nova conexão é perdida, então o contador de retentativas inicia do zero
Disconnect if non-responsive for ... seconds	Habilite esta opção para forçar o Driver a se desconectar se nenhum byte chegou à interface de I/O no <i>time-out</i> especificado, em segundos. Este <i>time-out</i> deve ser maior que o <i>time-out</i> configurado na opção Timeout

Opções para o grupo Logging Options

OPÇÃO	DESCRIÇÃO
Log to File	<p>Habilite esta opção e configure o nome do arquivo onde o log é escrito. Arquivos de log podem ser bem extensos, portanto utilize esta opção por curtos períodos de tempo, apenas para o propósito de testes e depurações.</p> <p>Caso se utilize a macro %PROCESS% no nome do arquivo de log, esta é substituída pelo ID do processo atual. Esta opção é particularmente útil ao se utilizar várias instâncias do mesmo Driver no E3, permitindo assim que cada instância gere um arquivo separado de log. Por exemplo, ao configurar esta opção como c:\e3logs\drivers\sim_%PROCESS%.log, gera-se o arquivo c:\e3logs\drivers\sim_00000FDA.log para o processo 0FDAh.</p> <p>Pode-se também utilizar a macro %DATE% no nome do arquivo. Neste caso é gerado um arquivo de log por dia (no formato aaaa_mm_dd). Por exemplo, ao configurar esta opção como c:\e3logs\drivers\sim_%DATE%.log, gera-se o arquivo c:\e3logs\drivers\sim_2005_12_31.log no dia 31 de dezembro de 2005 e o arquivo c:\e3logs\drivers\sim_2006_01_01.log no dia primeiro de janeiro de 2006</p>

Aba Serial

Utilize esta aba para configurar os parâmetros da Interface **Serial**.

Serial

Port:

Baud rate:

Data bits:

Parity:

Stop bits:

Enable 'ECHO' suppression

Handshaking

DTR control:

RTS control:

Wait for CTS before send

CTS timeout: ms

Delay before send: ms

Delay after send: ms

Inter-byte delay (microseconds): μ s

Inter-frame delay (milliseconds): ms

Aba Serial

Opções gerais da aba Serial

OPÇÃO	DESCRIÇÃO
Port	Selecione a porta serial a partir da lista, de COM1 até COM4 , ou digite o nome da porta serial no formato COMn , como por exemplo "COM15". Ao digitar o nome da porta manualmente, a caixa de diálogo aceita apenas nomes de portas começando com a expressão "COM"
Baud rate	Selecione o <i>baud rate</i> a partir da lista (1200, 2400, 4800, 9600, 19200, 38400, 57600 ou 115200) ou digite o <i>baud rate</i> desejado, como por exemplo 600
Data bits	Selecione 7 (sete) ou 8 (oito) bits de dados a partir da lista
Parity	Selecione a paridade a partir da lista. As opções disponíveis são None, Even, Odd, Mark ou List
Stop bits	Selecione o número de stop bits a partir da lista. As opções disponíveis são 1, 1.5 ou 2 stop bits
Enable 'ECHO' suppression	Habilite esta opção para remover o eco recebido após a Interface de Comunicação enviar dados pela porta serial. Se o eco não é igual aos bytes recém enviados, a Interface de Comunicação aborta a comunicação
Inter-byte delay (microseconds)	Define uma espera entre cada byte transmitido pela Interface de Comunicação, em milionésimos de segundo, ou seja, 1000000 é igual a um segundo. Esta opção deve ser utilizada com esperas pequenas de menos de um milissegundo
Inter-frame delay (milliseconds)	Define uma espera entre pacotes enviados ou recebidos pela Interface de Comunicação, em milésimos de segundo, ou seja, 1000 é igual a um segundo. Esta espera é

OPÇÃO	DESCRIÇÃO
	aplicada caso a Interface de Comunicação envie dois pacotes consecutivos, ou entre um pacote recebido e o próximo envio

O grupo **Handshaking** configura o uso dos sinais **RTS**, **CTS** e **DTR** no processo de *handshaking* ou seja, controlar quando o dado pode ser enviado ou recebido através da linha serial. Na maioria das vezes, configurar a opção **DTR control** para **ON** e a opção **RTS control** para **Toggle** funciona tanto com linhas seriais RS232 quanto com linhas seriais RS485.

Opções disponíveis no grupo Handshaking

OPÇÃO	DESCRIÇÃO
DTR control	Selecione ON para deixar o sinal DTR sempre ligado enquanto a porta serial está aberta. Selecione OFF para desligar o sinal DTR enquanto a porta serial está aberta. Alguns equipamentos exigem que o sinal DTR esteja ligado para permitir a comunicação
RTS control	Selecione ON para deixar o sinal RTS sempre ligado enquanto a porta serial está aberta. Selecione OFF para desligar o sinal RTS enquanto a porta serial está aberta. Selecione Toggle para ligar o sinal RTS enquanto se envia os bytes através da porta serial, e desligá-lo quando não se está enviando bytes e, portanto, habilitando a recepção
Wait for CTS before send	Disponível apenas quando a opção RTS control está configurada para Toggle . Utilize esta opção para forçar o Driver a verificar o sinal CTS antes de enviar os bytes através da porta serial, após ligar o sinal de RTS . Neste modo o sinal CTS é tratado como um <i>flag</i> de permissão para envio
CTS timeout	Determina o tempo máximo, em milissegundos, que o Driver aguarda pelo sinal de CTS depois de ligar o sinal de RTS . Se o sinal de CTS não é levantado dentro deste <i>time-out</i> , o Driver falha a comunicação atual e retorna erro
Delay before send	Alguns hardwares de porta serial demoram a habilitar o circuito de envio de dados depois que o sinal RTS é ligado. Configure esta opção para aguardar uma determinada quantidade de milissegundos depois de ligar o sinal RTS e antes de enviar o primeiro byte. IMPORTANTE: Esta espera deve ser utilizada com muito cuidado, pois consome 100% dos recursos da CPU enquanto aguarda. A performance geral do sistema se degrada conforme este valor aumenta
Delay after send	Tem o mesmo efeito que a opção Delay before send , mas neste caso a espera é efetuada depois que o último byte é enviado, antes de desligar o sinal RTS

Aba Ethernet

Utilize esta aba para configurar os parâmetros da Interface **Ethernet**. Estes parâmetros, exceto as configurações da porta, devem ser também configurados para uso na Interface **RAS**.

Ethernet

Transport: TCP/IP ▾

PING before connecting
 Timeout: 4000 ms
 Retries: 1

Listen for connections on port: 0

Share listen port with other processes

Interface: (All Interfaces) ▾

Use IPv6 Use SSL SSL Settings

Enable 'ECHO' suppression

IP Filter:

Connect to

<input type="checkbox"/> Main IP:		Port:	502	<input type="checkbox"/> Local port:	0
<input type="checkbox"/> Backup IP 1:		Port:	0	<input type="checkbox"/> Local port:	0
<input type="checkbox"/> Backup IP 2:		Port:	0	<input type="checkbox"/> Local port:	0
<input type="checkbox"/> Backup IP 3:		Port:	0	<input type="checkbox"/> Local port:	0

Aba Ethernet

Opções disponíveis na aba Ethernet

OPÇÃO	DESCRIÇÃO
Transport	Selecione TCP/IP para um <i>socket</i> TCP (<i>stream</i>) ou selecione UDP/IP para utilizar um <i>socket</i> UDP (<i>connectionless datagram</i>)
Listen for connections on port	Utilize esta opção para aguardar por novas conexões em uma porta IP específica, comum em Drivers Escravos. Caso esta opção permaneça desmarcada, o Driver se conecta ao endereço e porta especificados na opção Connect to
Share listen port with other processes	Selecione esta opção para compartilhar a porta de escuta com outros Drivers e processos
Interface	Selecione a interface de rede local, identificada por seu endereço IP, que o Driver utiliza para efetuar e receber conexões, ou selecione o item (All Interfaces) para permitir conexões em qualquer interface de rede
Use IPv6	Selecione esta opção para forçar o Driver a utilizar endereços no formato IPv6 em todas as conexões Ethernet. Deixe esta opção desmarcada para utilizar o formato IPv4
Enable 'ECHO' suppression	Habilite esta opção para eliminar o <i>eco</i> dos dados recebidos. O <i>eco</i> é uma cópia dos dados que foram enviados, que pode ser retornada antes da mensagem de resposta
IP Filter	Lista de endereços IP restringidos ou permitidos de onde o Driver aceita conexões (<i>Firewall</i>). Consulte a propriedade IO.Ethernet.IPFilter para mais informações
PING before connecting	Habilite esta opção para executar um comando ping , ou seja, verificar se o dispositivo pode ser encontrado na rede,

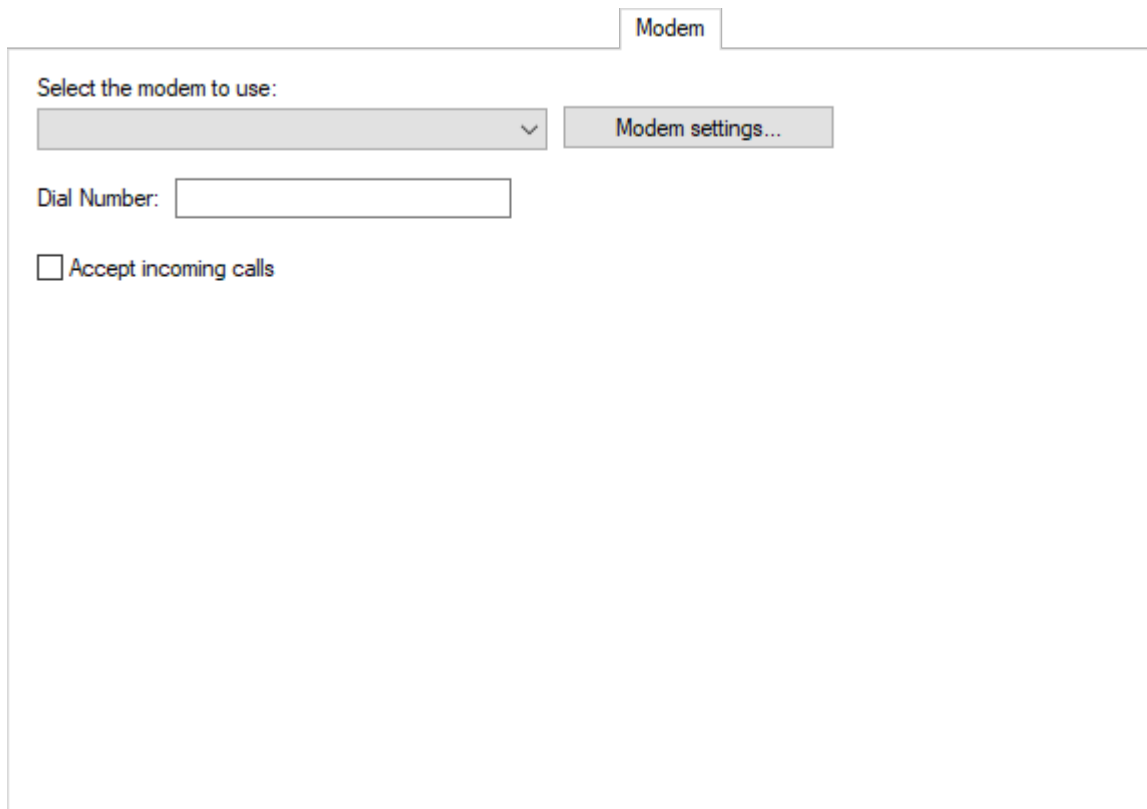
OPÇÃO	DESCRIÇÃO
	<p>no dispositivo antes de tentar uma conexão com o <i>socket</i>. Esta é uma maneira rápida de determinar uma conexão bem sucedida antes de tentar abrir um <i>socket</i> com o dispositivo. O <i>time-out</i> de uma conexão com um <i>socket</i> pode ser bem alto. As opções disponíveis são:</p> <ul style="list-style-type: none"> • Timeout: Especifique o número de milissegundos de espera por uma resposta do comando ping. Deve-se usar o comando ping para verificar o tempo normal de resposta, configurando esta opção para um valor acima desta média. Normalmente pode-se configurar um valor entre 1000 e 4000 milissegundos, ou seja entre um e quatro segundos • Retries: Número de retentativas do comando ping, sem contar a tentativa inicial. Se todas as tentativas falharem, então a conexão com o <i>socket</i> é abortada

Opções disponíveis no grupo **Connect to**

OPÇÃO	DESCRIÇÃO
Main IP	<p>Digite o endereço IP do dispositivo remoto. Pode-se usar tanto o endereço IP separado por pontos quanto uma URL. No caso de uma URL, o Driver usa o serviço de DNS disponível para mapear a URL para um endereço IP, como por exemplo "192.168.0.13" ou "Server1"</p>
Port	<p>Digite a porta IP do dispositivo remoto, entre 0 (zero) e 65535</p>
Local port	<p>Selecione esta opção para utilizar uma porta IP local fixa ao conectar ao dispositivo remoto</p>
Backup IP 1, 2 e 3	<p>Indique o endereço IP, a porta IP e a porta IP local fixa de até 3 (três) endereços de <i>backup</i> do dispositivo remoto</p>

Aba Modem

Utilize esta aba para configurar os parâmetros da Interface **Modem**. Algumas opções da aba **Serial** afetam a configuração do modem, portanto é interessante não esquecer de configurar a Interface **Serial**.



Aba Modem

A Interface **Modem** utiliza os modems TAPI instalados no computador.

Opções disponíveis na aba Modem

OPÇÃO	DESCRIÇÃO
Select the modem to use	Selecione um modem a partir da lista de modems disponíveis no computador. Caso selecione-se a opção Default modem , então o primeiro modem disponível é utilizado. Selecionar esta opção é recomendado especialmente quando a aplicação é utilizada em outro computador
Modem settings	Clique para abrir a janela de configuração do modem selecionado
Dial Number	Digite o número padrão para discagem. Este valor pode ser modificado em tempo de execução. Pode-se utilizar o caractere w para representar uma pausa ou espera pelo tom de discagem. Por exemplo, "0w33313456" disca o número 0 (zero), espera e então disca o número "33313456"
Accept incoming calls	Habilite esta opção para que o Driver atenda o telefone quando receber uma chamada externa. Para utilizar esta opção é necessário configurar a opção Connection management na aba Setup para Manual

Aba RAS

Use esta aba para configurar os parâmetros da Interface **RAS**. É necessário também configurar a aba **Ethernet**.

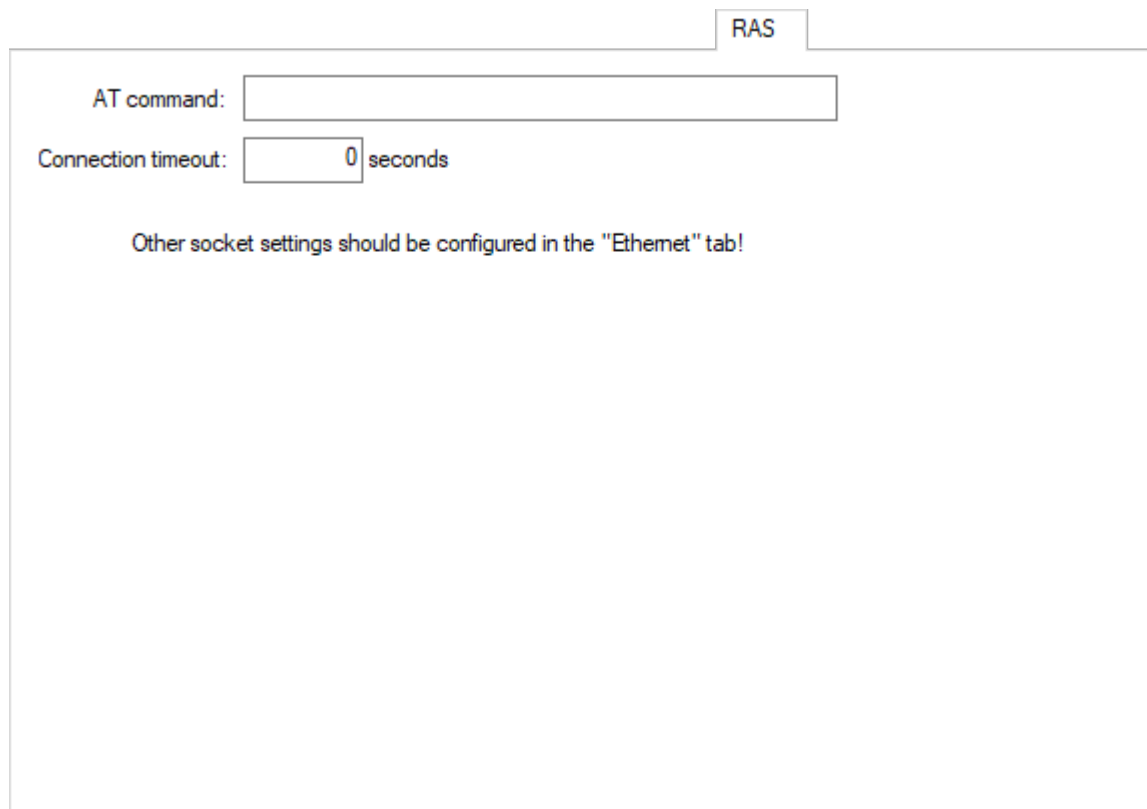
A Interface **RAS** abre uma conexão *socket* com um dispositivo RAS. O dispositivo RAS é um servidor de modems acessível através de TCP/IP, aguardando por conexões *socket* em uma porta IP. Para cada conexão aceita nesta porta tem-se acesso a um modem.

Ao conectar-se a um dispositivo RAS, primeiramente a Interface de Comunicação conecta ao *socket* no endereço IP e na porta configurados na aba **Ethernet**. Depois que o *socket* é aberto, os seguintes passos de inicialização ou de conexão são efetuados:

1. Limpeza do *socket*, ou seja, remove qualquer mensagem de saudação **TELNET** recebida do dispositivo RAS.
2. Envio do comando de discagem **AT** (em ASCII) no *socket*.
3. Aguarda pela recepção de uma resposta **CONNECT**.
4. Caso o *time-out* expire, a conexão é abortada.
5. Se a resposta **CONNECT** é recebida dentro do *time-out*, o *socket* está disponível para comunicação com o dispositivo, ou seja, a conexão foi estabelecida.

Se o passo 5 é efetuado com sucesso, então o *socket* comporta-se como um *socket* normal, com o dispositivo RAS funcionando como um roteador entre o Driver e o dispositivo. Os bytes enviados pelo Driver são recebidos pelo dispositivo RAS e enviados para o dispositivo de destino utilizando um modem. Os bytes recebidos pelo dispositivo RAS do modem são enviados de volta ao Driver utilizando o mesmo *socket*.

Depois que a conexão é estabelecida, a Interface **RAS** monitora os dados recebidos pelo Driver. Caso uma **String** "NO CARRIER" seja encontrada, o *socket* é fechado. Se o dispositivo RAS não envia o sinal **NO CARRIER**, a Interface **RAS** não consegue detectar quando a conexão modem entre o dispositivo RAS e o dispositivo final de I/O falha. Para recuperação de tal falha é fortemente recomendado que seja habilitada a opção **Disconnect if non-responsive** na aba **Setup**.



The image shows a screenshot of a software interface with a tab labeled "RAS". Inside the tab, there are two input fields: "AT command:" followed by an empty text box, and "Connection timeout:" followed by a text box containing the number "0" and the word "seconds". Below these fields, there is a text instruction: "Other socket settings should be configured in the 'Ethernet' tab!".

Aba RAS

Opções disponíveis na aba RAS

OPÇÃO	DESCRIÇÃO
AT command	Uma String com o comando AT completo usado para discar para o dispositivo de destino. Por exemplo, "ATDT33313456" disca por tom para o número "33313456"
Connection timeout	Número de segundos a aguardar por uma resposta CONNECT do modem, após o envio do comando AT

Configurações Gerais

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e as **Propriedades** gerais das Interfaces de Comunicação.

Tags de Comunicação

Tags gerais das Interfaces de Comunicação (N2/B2 = 0)

Os Tags descritos a seguir são fornecidos para todas as Interfaces de I/O suportadas.

IO.IOKitEvent

Tipo de Tag	Tag Bloco
Tipo de Acesso	Somente Leitura
Parâmetro B1	-1 (menos um)
Parâmetro B2	0 (zero)
Parâmetro B3	0 (zero)
Parâmetro B4	1 (um)
Propriedade Size	4 (quatro)
Propriedade ParamItem	IO.IOKitEvent

Este Bloco retorna eventos de Driver gerados por várias fontes nas Interfaces de Comunicação. A propriedade **TimeStamp** do Bloco representa o momento em que o evento ocorreu. Os Elementos de Bloco são os seguintes:

- **Elemento 0:** Tipo de evento. Os valores possíveis são **0:** Informação, **1:** Advertência ou **2:** Erro
- **Elemento 1:** Fonte do evento. Os valores possíveis são **0:** Driver (específico do Driver), **-1:** IOKit (eventos genéricos da Interface de Comunicação), **-2:** Interface **Serial**, **-3:** Interface **Modem**, **-4:** Interface **Ethernet** ou **-5:** Interface **RAS**
- **Elemento 2:** Número do erro, específico de cada fonte de evento
- **Elemento 3:** Mensagem do evento, uma **String** específica de cada evento

NOTA

O Driver mantém um número máximo de 100 eventos internamente. Se eventos adicionais são reportados, os eventos mais antigos são descartados.

IO.PhysicalLayerStatus

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	2 (dois)
Configuração por String	IO.PhysicalLayerStatus

Este Tag indica o estado da camada física. Os possíveis valores são os seguintes:

- **0:** Camada física parada, ou seja, o Driver está em modo **Offline**, a camada física falhou ao inicializar ou excedeu o número máximo de tentativas de reconexão
- **1:** Camada física iniciada mas não conectada, ou seja, o Driver está em modo **Online**, mas a camada física não está conectada. Se a opção **Connection management** estiver configurada para **Automatic**, a camada física pode estar conectando, desconectando ou esperando por uma tentativa de reconexão. Se a opção **Connection management** estiver configurada para **Manual**, então a camada física permanece neste estado até ser forçada a conectar
- **2:** Camada física conectada, ou seja, a camada física está pronta para ser usada. Isto **NÃO** significa que o equipamento esteja conectado, apenas que a camada de acesso está funcionando

IO.SetConfigurationParameters

Tipo de Tag	Tag Bloco
Tipo de Acesso	Somente Leitura
Parâmetro B1	-1 (menos um)
Parâmetro B2	0 (zero)
Parâmetro B3	0 (zero)
Parâmetro B4	3 (três)
Propriedade Size	2 (dois)
Propriedade ParamItem	IO.SetConfigurationParameters

Use este Tag para modificar qualquer propriedade da caixa de diálogo de configuração do Driver em tempo de execução.

Este Tag funciona somente enquanto o Driver estiver em modo **Offline**. Para iniciar o Driver em modo **Offline**, selecione a opção **Start driver OFFLINE** na caixa de diálogo de configuração do Driver. Pode-se tanto escrever em um Tag PLC ou em um Tag Bloco contendo os parâmetros a serem modificados. As escritas de Elementos de Bloco individuais não são suportadas, o Bloco inteiro precisa ser escrito de uma vez só.

No **Eclipse SCADA** é necessário usar um Tag Bloco. Cada parâmetro a ser configurado utiliza dois Elementos de Bloco. Por exemplo, caso seja necessário configurar três parâmetros, então o tamanho do Bloco deve ser 6 (3 × 2). O primeiro Elemento é o nome da propriedade, como uma **String**, e o segundo Elemento é o valor da propriedade, conforme o exemplo a seguir.

```
// 'Block' deve ser um Tag Bloco com leitura automática,
// leitura por varredura e escrita automática desabilitadas.
// Configura os parâmetros
Block.element001 = "IO.Type" // Parâmetro 1
Block.element002 = "Serial"
Block.element003 = "IO.Serial.Port" // Parâmetro 2
Block.element004 = 1
Block.element005 = "IO.Serial.BaudRate" // Parâmetro 3
Block.element006 = 19200
// Escreve o Bloco inteiro
Block.Write()
```

Ao usar o **E3**, a habilidade de criar *arrays* em tempo de execução permite o uso tanto de um Tag de Comunicação quanto de um Tag Bloco. Pode-se utilizar o método **Write** do Driver para enviar os parâmetros diretamente para o Driver, sem a necessidade de criar um Tag, conforme o exemplo a seguir.

```
Dim arr(6)
' Configura os elementos do array
arr(1) = "IO.Type"
arr(2) = "Serial"
arr(3) = "IO.Serial.Port"
arr(4) = 1
arr(5) = "IO.Serial.BaudRate"
arr(6) = 19200
' Há dois métodos de enviar os parâmetros
' Método 1: Usando um Tag de Comunicação
tag.WriteEx arr
' Método 2: Sem utilizar um Tag
Driver.Write -1, 0, 0, 3, arr
```

Uma variação do exemplo anterior usa um *array* bidimensional.

```
Dim arr(10)
' Configura os elementos do array. Note que o array foi redimensionado
' para 10 elementos. Elementos vazios são ignorados pelo Driver
arr(1) = Array("IO.Type", "Serial")
arr(2) = Array("IO.Serial.Port", 1)
arr(3) = Array("IO.Serial.BaudRate", 19200)
Driver.Write -1, 0, 0, 3, arr
```

O Driver não valida nomes de parâmetros ou valores passados, por isto tenha cuidado ao escrever parâmetros e valores. O comando **Write** falha se o *array* de configuração é criado incorretamente. Pode-se consultar o log do Driver ou usar o parâmetro *writeStatus* do método **WriteEx** para descobrir a causa exata do erro.

```
Dim arr(10), strError
arr(1) = Array("IO.Type", "Serial")
arr(2) = Array("IO.Serial.Port", 1)
arr(3) = Array("IO.Serial.BaudRate", 19200)
If Not Driver.WriteEx -1, 0, 0, 3, arr, , , strError Then
    MsgBox "Falha ao configurar os parâmetros do Driver: " + strError
End If
```

IO.WorkOnline

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Leitura ou Escrita
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	4 (quatro)
Configuração por String	IO.WorkOnline

Este Tag informa o estado atual do Driver e permite iniciar ou parar a camada física. Os valores possíveis são os seguintes:

- **0 - Driver Offline:** A camada física está fechada ou parada. Este modo permite uma configuração dinâmica dos parâmetros do Driver através do Tag **IO.SetConfigurationParameters**
- **1 - Driver Online:** A camada física está aberta ou em execução. Enquanto estiver em modo **Online**, a camada física pode ser conectada ou desconectada e seu estado atual pode ser conferido no Tag **IO.PhysicalLayerStatus**

No exemplo a seguir, utilizando o **E3**, o Driver é colocado em modo **Offline**, a porta COM é modificada e então é colocado em modo **Online** novamente.

```
'Configura o Driver em modo Offline
Driver.Write -1, 0, 0, 4, 0
'Muda a porta para COM2
Driver.Write -1, 0, 0, 3, Array("IO.Serial.Port", 2)
'Configura o Driver em modo Online
Driver.Write -1, 0, 0, 4, 1
```

O método **Write** pode falhar quando estiver configurando o Driver em modo **Online**, ou seja, escrevendo o valor 1 (um). Neste caso, o Driver permanece em modo **Offline**. A causa da falha pode ser:

- Tipo de camada física configurada incorretamente, provavelmente um valor inválido foi configurado para a propriedade **IO.Type**
- O Driver pode ter ficado sem memória
- A camada física pode ter deixado de criar seu *thread* de trabalho. Procure no arquivo de log pela mensagem "Failed to create physical layer thread!"
- A camada física não conseguiu inicializar. A causa da falha depende do tipo de camada física. Pode ser um número de porta serial inválida, falha ao inicializar o Windows Sockets ou falha ao inicializar o TAPI (modem), entre outras. A causa é gravada no arquivo de log

IMPORTANTE

Mesmo que a configuração do Driver para o modo **Online** seja bem sucedida, isto não significa necessariamente que a camada física esteja pronta para uso, ou seja, pronta para executar operações de entrada e saída com um equipamento externo. O Tag **IO.PhysicalLayerStatus** deve ser verificado para assegurar que a camada física esteja conectada e preparada para a comunicação.

Propriedades

Estas são as propriedades gerais de todas as Interfaces de I/O suportadas.

IO.ConnectionMode

9 Controla o modo de gerenciamento da Conexão. Os valores possíveis são **0**: Modo automático, em que o Driver gerencia a conexão ou **1**: Modo manual, em que a aplicação gerencia a conexão.

IO.GiveUpEnable

■ Quando configurada para Verdadeiro, define um número máximo de tentativas de reconexão. Se todas as reconexões falharem, o Driver entra em modo **Offline**. Se configurada para Falso, o Driver tenta até que uma reconexão seja bem-sucedida.

IO.GiveUpTries

9 Número de tentativas de reconexão antes que esta seja abortada. Por exemplo, se o valor desta propriedade é igual a 1 (um), o Driver tenta apenas uma reconexão quando a conexão é perdida. Se esta falhar, o Driver entra em modo **Offline**.

IO.InactivityEnable

▣ Configure em Verdadeiro para habilitar e em Falso para desabilitar a detecção de inatividade. A camada física é desconectada se estiver inativa por um certo período de tempo. A camada física é considerada inativa apenas se é capaz de enviar dados mas não de recebê-los de volta.

IO.InactivityPeriodSec

9 Número de segundos para a verificação de inatividade. Se a camada física estiver inativa por este período de tempo, então é desconectada.

IO.RecoverEnable

▣ Configure em Verdadeiro para habilitar o Driver a recuperar conexões perdidas e em Falso para deixar o Driver em modo **Offline** quando uma conexão é perdida.

IO.RecoverPeriodSec

9 Tempo de espera entre duas tentativas de conexão, em segundos.

NOTA

A primeira reconexão é executada imediatamente após a conexão ser perdida.

IO.StartOffline

▣ Configure em Verdadeiro para iniciar o Driver em modo **Offline** e em Falso para iniciar o Driver em modo **Online**.

NOTA

Não faz sentido modificar esta propriedade em tempo de execução, já que esta só pode ser modificada quando o Driver já estiver em modo **Offline**. Para configurar o Driver em modo **Online** em tempo de execução, escreva o valor 1 (um) no Tag **IO.WorkOnline**.

IO.TimeoutMs

9 Define o *time-out* da camada física, em milissegundos. Um segundo equivale a 1000 milissegundos.

IO.Type

A Define o tipo de interface física utilizada pelo Driver. Os valores possíveis são os seguintes:

- **N ou None:** Não utiliza uma interface física, ou seja, o Driver deve fornecer uma interface personalizada
- **S ou Serial:** Utiliza uma porta serial local (COM n)
- **M ou Modem:** Utiliza um modem local, interno ou externo, acessado via TAPI (*Telephony Application Programming Interface*)

- **E ou Ethernet:** Utiliza um *socket* TCP/IP ou UDP/IP
- **R ou RAS:** Utiliza uma Interface **RAS** (*Remote Access Server*). O Driver conecta-se ao equipamento RAS através da Interface **Ethernet** e então emite um comando **AT** (*dial*)

Configuração de Estatísticas

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e as **Propriedades** das estatísticas das Interfaces de Comunicação.

Tags de Comunicação

Tags de estatísticas das Interfaces de Comunicação (N2/B2 = 0)

Os Tags descritos a seguir mostram estatísticas para todas as Interfaces de Comunicação.

IO.Stats.Partial.BytesRecv

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1101
Configuração por String	IO.Stats.Partial.BytesRecv

Este Tag retorna a quantidade de bytes recebidos na conexão atual.

IO.Stats.Partial.BytesSent

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1100
Configuração por String	IO.Stats.Partial.BytesSent

Este Tag retorna a quantidade de bytes enviados na conexão atual.

IO.Stats.Partial.TimeConnectedSeconds

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1102
Configuração por String	IO.Stats.Partial.TimeConnectedSeconds

Este Tag retorna o número de segundos que o Driver está conectado na conexão atual ou 0 (zero) se o Driver está desconectado.

IO.Stats.Partial.TimeDisconnectedSeconds

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1103
Configuração por String	IO.Stats.Partial.TimeDisconnectedSeconds

Este Tag retorna o número de segundos que o Driver está desconectado desde o término da última conexão ou 0 (zero) se o Driver está conectado.

IO.Stats.Total.BytesRecv

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1001
Configuração por String	IO.Stats.Total.BytesRecv

Este Tag retorna a quantidade de bytes recebidos desde que o Driver foi carregado.

IO.Stats.Total.BytesSent

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1000
Configuração por String	IO.Stats.Total.BytesSent

Este Tag retorna a quantidade de bytes enviados desde que o Driver foi carregado.

IO.Stats.Total.ConnectionCount

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1004
Configuração por String	IO.Stats.Total.ConnectionCount

Este Tag retorna a quantidade de conexões que o Driver já estabeleceu, com sucesso, desde que foi carregado.

IO.Stats.Total.TimeConnectedSeconds

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1002
Configuração por String	IO.Stats.Total.TimeConnectedSeconds

Este Tag retorna o número de segundos que o Driver permaneceu conectado desde que foi carregado.

IO.Stats.Total.TimeDisconnectedSeconds

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1003
Configuração por String	IO.Stats.Total.TimeDisconnectedSeconds

Este Tag retorna o número de segundos que o Driver permaneceu desconectado desde que foi carregado.

Propriedades

Atualmente, não existem propriedades definidas especificamente para mostrar as estatísticas das Interfaces de Comunicação em tempo de execução.

Configuração da Interface Ethernet

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e as **Propriedades** da Interface Ethernet.

Tags de Comunicação

Tags da Interface Ethernet (N2/B2 = 4)

Os Tags descritos a seguir permitem controlar e identificar a Interface **Ethernet** em tempo de execução e também são válidos quando a Interface **RAS** estiver selecionada.

IMPORTANTE

Estes Tags estão disponíveis **SOMENTE** enquanto o Driver estiver em modo **Online**.

IO.Ethernet.IPSelect

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Leitura ou Escrita
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	4 (quatro)
Parâmetro N4	0 (zero)
Configuração por String	IO.Ethernet.IPSelect

Indica o endereço IP ativo. Os valores possíveis são **0**: O endereço principal de IP está selecionado, **1**: O primeiro endereço IP alternativo ou de *backup* está selecionado, **2**: O segundo endereço IP alternativo ou de *backup* está selecionado ou **3**: O terceiro endereço IP alternativo ou de *backup* está selecionado.

Se a Interface **Ethernet** ou **RAS** está conectada, este Tag indica qual dos quatro endereços IP configurados está em uso. Se a Interface está desconectada, este Tag indica qual endereço IP é usado primeiro na próxima tentativa de conexão.

Durante o processo de conexão, se o endereço IP ativo não está disponível, a Interface de Comunicação tenta conectar-se usando o outro endereço IP. Se a conexão com o endereço IP alternativo funcionar, este é configurado como o endereço IP ativo (*switchover* automático).

Para forçar um *switchover* manual, escreva valores de 0 (zero) a três (3) neste Tag. Isto força a reconexão com o endereço IP especificado (**0**: Endereço principal, **1, 2, 3**: Endereços alternativos) se o Driver está atualmente conectado. Se o Driver está desconectado, este Tag configura o endereço IP ativo para a próxima tentativa de conexão.

IO.Ethernet.IPSwitch

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Escrita
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	4 (quatro)
Parâmetro N4	1 (um)
Configuração por String	IO.Ethernet.IPSwitch

Qualquer valor escrito neste Tag força um *switchover* manual. Se o endereço principal de IP está ativo, então o primeiro endereço IP alternativo ou de *backup* é ativado, e assim por diante passando por todos os endereços IP alternativos e voltando para o endereço principal até estabelecer uma conexão.

Se o Driver estiver desconectado, este Tag configura o endereço IP ativo para a próxima tentativa de conexão.

Propriedades

Estas propriedades controlam a configuração da Interface **Ethernet**.

NOTA

A Interface **Ethernet** também é usada pela Interface **RAS**.

IO.Ethernet.AcceptConnection

Configure em Falso se o Driver não deve aceitar conexões externas, ou seja, o Driver se comporta como mestre, ou configure em Verdadeiro para habilitar a recepção de conexões, ou seja, o Driver se comporta como escravo.

IO.Ethernet.BackupEnable[2,3]

Configure em Verdadeiro para habilitar o endereço IP alternativo ou de *backup*. Se a tentativa de reconectar com o endereço IP principal falhar, o Driver tenta utilizar um endereço IP alternativo ou de *backup*. Configure em Falso para desabilitar sua utilização.

IO.Ethernet.BackupIP[2,3]

A Endereço IP alternativo ou de *backup* do equipamento remoto. Pode-se utilizar tanto o endereço numérico como o nome de *host* do equipamento, como por exemplo "192.168.0.7" ou "SERVER2".

IO.Ethernet.BackupLocalPort[2,3]

9 Número da porta local a ser utilizada na conexão ao endereço IP alternativo ou de *backup* do equipamento remoto. Usado apenas se a propriedade **IO.Ethernet.BackupLocalPortEnable** está configurada para Verdadeiro.

IO.Ethernet.BackupLocalPortEnable[2,3]

■ Configure em Verdadeiro para forçar o uso de uma porta local específica ao conectar ao endereço IP alternativo ou de *backup* ou configure em Falso para utilizar qualquer porta local disponível.

IO.Ethernet.BackupPort[2,3]

9 Número da porta do endereço IP alternativo ou de *backup* do equipamento remoto, usado juntamente com a propriedade **IO.Ethernet.BackupIP**.

IO.Ethernet.IPFilter

A Lista de endereços IPv4 ou IPv6 separados por vírgula, que define de quais endereços o Driver aceita ou bloqueia conexões. Pode-se utilizar asteriscos, como por exemplo "192.168.*.*", ou intervalos, como por exemplo "192.168.0.41-50", em qualquer parte dos endereços IP. Para bloquear um endereço IP ou um intervalos de endereços IP, use o caractere til ("~") no início do endereço. Exemplos:

- **192.168.0.24**: Aceita apenas conexões do endereço IPv4 192.168.0.24
- **192.168.0.41-50**: Aceita conexões dos endereços IPv4 no intervalo entre 192.168.0.41 e 192.168.0.50
- **192.168.0.***: Aceita conexões dos endereços IPv4 no intervalo entre 192.168.0.0 e 192.168.0.255
- **fe80:3bf:877::** (expande para fe80:03bf:0877:0000:0000:0000:0000:0000:****)**: Aceita conexões de endereços IPv6 no intervalo entre fe80:03bf:0877:0000:0000:0000:0000:0000 e fe80:03bf:0877:0000:0000:0000:0000:ffff:ffff
- **192.168.0.10, 192.168.0.15, 192.168.0.20**: Aceita conexões dos endereços IPv4 192.168.0.10, 192.168.0.15 e 192.168.0.20
- **~192.168.0.95, 192.168.0.***: Aceita conexões dos endereços IPv4 no intervalo entre 192.168.0.0 e 192.168.0.255, exceto o endereço IPv4 192.168.0.95

Quando o Driver recebe uma tentativa de conexão, a lista de filtros é percorrida sequencialmente da esquerda para a direita, procurando por uma autorização ou bloqueio específico para o endereço IP de onde veio a conexão. Se nenhum elemento da lista corresponder ao endereço IP, a autorização ou bloqueio são ditados pelo último elemento da lista:

- Se o último elemento da lista é uma autorização, como por exemplo "192.168.0.24", então todos os endereços IP não encontrados na lista são bloqueados
- Se o último elemento da lista é um bloqueio, como por exemplo "~192.168.0.24", então todos os endereços IP não encontrados na lista são autorizados

Se um endereço IP aparecer em mais de um filtro da lista, o filtro mais à esquerda tem precedência. Por exemplo, no caso de "~192.168.0.95, 192.168.0.*", o endereço IP 192.168.0.95 se encaixa nas duas regras, mas a regra que vale é a mais à esquerda, "~192.168.0.95", e portanto o endereço IP é bloqueado.

Quando o **IOKit** bloqueia uma conexão, a mensagem "Blocked incoming socket connection from {IP}!" é logada.

No caso de conexões UDP em modo escuta em *broadcast*, onde o Driver pode receber pacotes de diferentes endereços IP, o bloqueio ou permissão é realizado a cada pacote recebido. Se um pacote é recebido de um endereço IP bloqueado, a mensagem "Blocked incoming packet from {IP} (discarding {N} bytes)!" é logada.

IO.Ethernet.ListenIP

A Endereço IP da interface local de rede por onde o Driver efetua e aceita conexões. Deixe esta propriedade vazia para efetuar e aceitar conexões por qualquer interface local de rede.

IO.Ethernet.ListenPort

9 Número da porta IP utilizada pelo Driver para escutar conexões.

IO.Ethernet.MainIP

A Endereço IP do equipamento remoto. Pode-se utilizar tanto o endereço numérico como o nome de *host* do equipamento, como por exemplo "192.168.0.7" ou "SERVER2".

IO.Ethernet.MainLocalPort

9 Número da porta local a ser utilizada na conexão ao endereço IP principal do equipamento remoto. Este valor é usado apenas se a propriedade **IO.Ethernet.MainLocalPortEnable** é igual a Verdadeiro.

IO.Ethernet.MainLocalPortEnable

■ Configure em Verdadeiro para forçar o uso de uma porta local específica ao conectar ao endereço IP principal ou configure em Falso para utilizar qualquer porta local disponível.

IO.Ethernet.MainPort

9 Número da porta IP no equipamento remoto, usado em conjunto com a propriedade **IO.Ethernet.MainIP**.

IO.Ethernet.PingEnable

■ Configure em Verdadeiro para habilitar o envio de um comando **ping** para o endereço IP do equipamento remoto, antes de tentar conectar-se ao *socket*. O *time-out* de conexão do *socket* não pode ser controlado, por isto o envio de um comando **ping** antes de conectar-se é uma maneira rápida de detectar se a conexão vai falhar. Configure em Falso para desabilitar o comando **ping**.

IO.Ethernet.PingTimeoutMs

9 Tempo de espera por uma resposta de um comando **ping**, em milissegundos.

IO.Ethernet.PingTries

9 Número máximo de tentativas de comandos **ping**. O valor mínimo é 1 (um), incluindo o primeiro comando **ping**.

IO.Ethernet.ShareListenPort

■ Configure em Verdadeiro para compartilhar a porta de escuta com outros Drivers e processos ou Falso para abrir a porta de escuta em modo exclusivo. Para compartilhar uma porta de escuta com sucesso, todos os Drivers e processos envolvidos devem abrir esta porta em modo compartilhado. Quando uma porta de escuta é compartilhada, cada nova conexão é distribuída para um dos processos que estão escutando. Desta forma, se um Driver Escravo só suporta uma

conexão por vez, pode-se utilizar várias instâncias deste Driver escutando na mesma porta, portanto simulando um Driver com suporte a múltiplas conexões.

IO.Ethernet.SuppressEcho

☑ Configure em Verdadeiro para eliminar o eco presente em uma comunicação. O eco é a recepção indesejada de uma cópia exata de todos os pacotes de dados que o Driver enviou para o equipamento.

IO.Ethernet.Transport

⚠ Define o protocolo de transporte. Os valores possíveis são **T ou TCP**: Utiliza o protocolo TCP/IP ou **U ou UDP**: Utiliza o protocolo UDP/IP.

IO.Ethernet.UseIPv6

☑ Configure em Verdadeiro para utilizar endereços IPv6 em todas as conexões Ethernet ou configure em Falso para utilizar endereços IPv4 (padrão).

Configuração da Interface Modem

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e as **Propriedades** da Interface **Modem** (TAPI).

Tags de Comunicação

Tags da Interface Modem (N2/B2 = 3)

Os Tags descritos a seguir permitem controlar e diagnosticar a Interface **Modem** (TAPI) em tempo de execução.

IMPORTANTE

Estes Tags estão disponíveis **SOMENTE** enquanto o Driver estiver em modo **Online**.

IO.TAPI.ConnectionBaudRate

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	3 (três)
Parâmetro N4	5 (cinco)
Configuração por String	IO.TAPI.ConnectionBaudRate

Indica o valor de *baud rate* da conexão atual. Se o modem não estiver conectado, retorna o valor 0 (zero).

IO.TAPI.Dial

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Escrita
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	3 (três)
Parâmetro N4	1 (um)
Configuração por String	IO.TAPI.Dial

Escreva qualquer valor neste Tag para forçar a Interface **Modem** a iniciar uma chamada. Este comando é assíncrono, apenas iniciando o processo de chamada. Pode-se monitorar o Tag **IO.TAPI.IsModemConnected** para detectar quando a chamada é estabelecida.

IO.TAPI.HangUp

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Escrita
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	3 (três)
Parâmetro N4	4 (quatro)
Configuração por String	IO.TAPI.HangUp

Qualquer valor escrito neste Tag desliga a chamada atual.

NOTA

Use este comando apenas quando estiver gerenciando a camada física manualmente ou se estiver explicitamente tentando forçar um Driver a reiniciar a comunicação. Se a camada física estiver configurada para reconexão automática, o Driver imediatamente tenta restabelecer a conexão.

IO.TAPI.IsModemConnected

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	3 (três)
Parâmetro N4	3 (três)
Configuração por String	IO.TAPI.IsModemConnected

Este Tag indica o estado da conexão do modem. Os valores possíveis são **0**: O modem não está conectado, mas pode estar realizando ou recebendo uma chamada externa ou **1**: O modem está conectado e o Driver completou ou recebeu uma chamada externa com sucesso. Enquanto estiver neste estado, a camada física consegue enviar ou receber dados.

IO.TAPI.IsModemConnecting

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	3 (três)
Parâmetro N4	6 (seis)
Configuração por String	IO.TAPI.IsModemConnecting

Este Tag indica o estado de conexão do modem, com mais detalhes do que o Tag **IO.TAPI.IsModemConnected**. Os valores possíveis são **0**: O modem não está conectado, **1**: O modem está conectando, ou seja, realizando ou recebendo uma chamada externa, **2**: O modem está conectado. Enquanto estiver neste estado, a camada física consegue enviar ou receber dados ou **3**: O modem está desconectando a chamada atual.

IO.TAPI.ModemStatus

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	3 (três)
Parâmetro N4	2 (dois)
Configuração por String	IO.TAPI.ModemStatus

Retorna uma **String** com o estado atual do modem. Os valores possíveis são os seguintes:

- **"No status!"**: A Interface **Modem** ainda não foi aberta ou já foi fechada
- **"Modem initialized OK!"**: A Interface **Modem** foi inicializada com sucesso
- **"Modem error at initialization!"**: O Driver não conseguiu inicializar a linha do modem. Confira o arquivo de log do Driver para maiores detalhes
- **"Modem error at dial!"**: O Driver não conseguiu começar ou aceitar uma chamada
- **"Connecting..."**: O Driver iniciou uma chamada com sucesso, e está atualmente processando esta chamada
- **"Ringing..."**: Indica que o modem está recebendo uma chamada externa, mas ainda não a aceitou
- **"Connected!"**: O Driver conectou-se com sucesso, ou seja, completou ou aceitou uma chamada externa

- **"Disconnecting..."**: O Driver está desligando a chamada atual
- **"Disconnected OK!"**: O Driver desligou a chamada atual
- **"Error: no dial tone!"**: O Driver abortou a chamada porque o sinal de linha disponível não foi detectado
- **"Error: busy!"**: O Driver abortou a ligação porque a linha estava ocupada
- **"Error: no answer!"**: O Driver abortou a chamada porque não recebeu resposta do outro modem
- **"Error: unknown!"**: A chamada atual foi abortada por um erro desconhecido

IO.TAPI.PhoneNumber

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Leitura ou Escrita
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	3 (três)
Parâmetro N4	0 (zero)
Configuração por String	IO.TAPI.PhoneNumber

Este Tag é uma **String** que lê ou modifica o número do telefone utilizado pelo Tag **IO.TAPI.Dial**. Ao modificar este Tag, o novo valor é usado apenas no próximo comando **Dial**.

Propriedades

Estas propriedades controlam a configuração da Interface **Modem** (TAPI).

IO.TAPI.AcceptIncoming

9 Configure em Falso se o modem não pode aceitar chamadas externas, ou seja, o Driver se comporta como mestre, e configure em Verdadeiro para habilitar a recepção de chamadas, ou seja, o Driver se comporta como escravo.

IO.TAPI.ModemID

9 É o número de identificação do modem. Este ID é criado pelo Windows e é usado internamente para identificar o modem dentro de uma lista de equipamentos instalados no computador. Este ID pode não permanecer válido caso o modem seja reinstalado ou a aplicação seja executada em outro computador.

NOTA

Recomenda-se que esta propriedade seja configurada em 0 (zero), indicando que o Driver deve utilizar o primeiro modem disponível.

IO.TAPI.PhoneNumber

A O número de telefone utilizado em comandos **Dial**, como por exemplo "0w01234566", em que o caractere "w" força o modem a esperar por um sinal de chamada.

Configuração da Interface RAS

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e as **Propriedades** da Interface **RAS**.

Tags de Comunicação

Tags da Interface RAS (N2/B2 = 5)

Atualmente, não existem Tags definidos especificamente para gerenciar a Interface **RAS** em tempo de execução.

Propriedades

Estas propriedades controlam a configuração da Interface **RAS**.

NOTA

A Interface **RAS** utiliza a Interface **Ethernet**, que por este motivo também deve ser configurada.

IO.RAS.ATCommand

🚩 Comando **AT** a ser enviado através do *socket* para forçar o equipamento RAS a realizar uma ligação usando o canal RAS atual, como por exemplo "ATDT6265545".

IO.RAS.CommandTimeoutSec

🚩 Tempo de espera pela mensagem **CONNECT** em resposta ao comando **AT**, em segundos.

Configuração da Interface Serial

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e as **Propriedades** da Interface **Serial**.

Tags de Comunicação

Tags da Interface Serial (N2/B2 = 2)

Atualmente, não existem Tags definidos especificamente para gerenciar a Interface **Serial** em tempo de execução.

Propriedades

Estas propriedades controlam a configuração da Interface **Serial**.

IO.Serial.Baudrate

🚩 Especifica a taxa de *bauds* da porta serial, como por exemplo 9600.

IO.Serial.CTSTimeoutMs

🚩 Tempo de espera pelo sinal **CTS**, em milissegundos. Após o sinal **RTS** ser ligado (**ON**), um temporizador é iniciado para esperar pelo sinal **CTS**. Se este temporizador expirar, o Driver aborta o envio de bytes através da porta serial. Disponível apenas quando a propriedade **IO.Serial.RTS** está configurada em **Toggle** e a propriedade **IO.Serial.WaitCTS** está configurada em Verdadeiro.

IO.Serial.DataBits

9 Especifica o número de bits de dados para a configuração da porta serial. Os valores possíveis são **5**: Cinco bits de dados, **6**: Seis bits de dados, **7**: Sete bits de dados ou **8**: Oito bits de dados.

IO.Serial.DelayAfterMs

9 Número de milissegundos de atraso após o último byte ter sido enviado através da porta serial, mas antes de desligar (**OFF**) o sinal **RTS**. Disponível apenas quando a propriedade **IO.Serial.RTS** está configurada em **Toggle** e a propriedade **IO.Serial.WaitCTS** está configurada em Falso.

IO.Serial.DelayBeforeMs

9 Número de milissegundos de atraso após o sinal **RTS** ter sido ligado (**ON**), mas antes dos dados serem enviados. Disponível apenas quando a propriedade **IO.Serial.RTS** está configurada em **Toggle** e a propriedade **IO.Serial.WaitCTS** está configurada em Falso.

IO.Serial.DTR

A Indica o modo como o Driver lida com o sinal **DTR**. Os valores possíveis são **OFF**: Sinal **DTR** sempre desligado ou **ON**: Sinal **DTR** sempre ligado.

IO.Serial.InterbyteDelayUs

9 Tempo de espera, em milissegundos (1/1000000 de um segundo), para cada dois bytes enviados pela Interface **Serial**.

IO.Serial.InterframeDelayMs

9 Tempo de espera, em milissegundos, antes de enviar um pacote após o último pacote enviado ou recebido.

IO.Serial.Parity

A Especifica a paridade para a configuração da porta serial. Os valores possíveis são **E ou Even**: Paridade par, **N ou None**: Sem paridade, **O ou Odd**: Paridade ímpar, **M ou Mark**: Paridade de marca ou **S ou Space**: Paridade de espaço.

IO.Serial.Port

9 Número da porta serial local. Os valores possíveis são **1**: Utiliza a porta COM1, **2**: Utiliza a porta COM2, **3**: Utiliza a porta COM3 ou **n**: Utiliza a porta COMn.

IO.Serial.RTS

A Indica como o Driver lida com o sinal **RTS**. Os valores possíveis são **OFF**: Sinal **RTS** sempre desligado, **ON**: Sinal **RTS** sempre ligado ou **Toggle**: Liga (**ON**) o sinal **RTS** quando estiver transmitindo dados e desliga (**OFF**) o sinal **RTS** quando não estiver transmitindo dados.

IO.Serial.StopBits

9 Especifica o número de bits de parada para a configuração da porta serial. Os valores possíveis são **1**: Um bit de parada, **2**: Um bit e meio de parada ou **3**: Dois bits de parada.

IO.Serial.SuppressEcho

9 Utilize um valor diferente de 0 (zero) para habilitar a supressão de eco ou 0 (zero) para desabilitá-la.

IO.Serial.WaitCTS

Configure em Verdadeiro para forçar o Driver a esperar pelo sinal **CTS** antes de enviar bytes quando o sinal **RTS** estiver ligado (**ON**). Disponível apenas quando a propriedade **IO.Serial.RTS** está configurada em **Toggle**.

Histórico de Revisões do Driver

VERSÃO	DATA	AUTOR	COMENTÁRIOS
1.0.1	24/08/2022	C. Mello	<ul style="list-style-type: none">Versão inicial deste Driver.

Matriz

Rua Mostardeiro, 322/Cj. 902, 1001 e
1002
90430-000 — Porto Alegre — RS
Fone: (+55 51) 3346-4699
Fax: (+55 51) 3222-6226
E-mail: elipse-rs@elipse.com.br

Filial no Paraná

Av. Sete de Setembro, 4698/1708
80240-000 — Curitiba — PR
Fone: (+55 41) 4062-5824
E-mail: elipse-pr@elipse.com.br

Filial no Rio de Janeiro

Av. José Silva de A. Neto, 200/Bl. 4/Sl.
109B
22250-044 — Rio de Janeiro — RJ
Fone: (+55 21) 2430-5912
Suporte Técnico: (+55 21) 2430-5963
E-mail: elipse-rj@elipse.com.br

Filial em São Paulo

Rua dos Pinheiros, 870/Cj. 141 e 142
05422-001 — São Paulo — SP
Fone: (+55 11) 3061-2828
Fax: (+55 11) 3086-2338
E-mail: elipse-sp@elipse.com.br

Filial em Minas Gerais

Rua Antônio de Albuquerque, 156/705
30112-010 — Belo Horizonte — MG
Fone: (+55 31) 4062-5824
E-mail: elipse-mg@elipse.com.br

Filial em Taiwan

9F., No.12, Beiping 2nd St., Sanmin Dist.
807 — Kaohsiung City — Taiwan
Fone: (+886 7) 323-8468
Fax: (+886 7) 323-9656
E-mail: evan@elipse.com.br

Consulte nosso website para informações sobre o representante do seu estado.

www.elipse.com.br

kb.elipse.com.br

forum.elipse.com.br

www.youtube.com/elipsesoftware

elipse@elipse.com.br



Gartner, Cool Vendors in Brazil 2014, April 2014.

Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability of fitness for a particular purpose.

Microsoft Partner
Gold Independent Software Vendor (ISV)