

# Driver Protec Protec

Nome do Arquivo	Protec.dll
Fabricante	Protec
Equipamentos	Protec Fire Detection PLC
Protocolo	Protec Control Panel RS232 To PC Protocol
Versão	2.0.2
Última Atualização	15/10/2024
Plataforma	Win32
Dependências	Nenhuma
Leitura com Superblocos	Não
Nível	0

## Introdução

O Driver Protec permite a aquisição de eventos gerados por um painel de controle Protec, e também possibilita o envio de requisições a este painel, dependendo do modelo.

## Configuração do Driver

Esta seção contém informações sobre a configuração deste Driver.

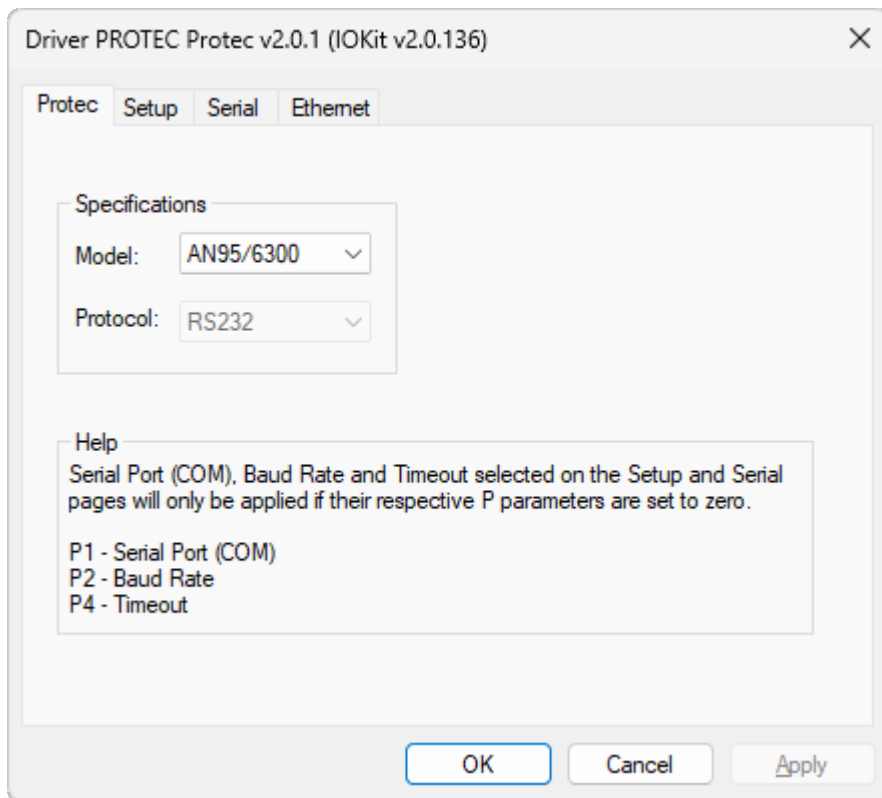
### Parâmetros [P]

P1	Número da porta serial. Os valores possíveis são <b>1</b> : COM1, <b>2</b> : COM2, ou <b>n</b> : COMn
P2	Taxa de transferência ( <i>baud rate</i> ). Os valores possíveis são 1200, 2400, 4800 ou 9600, entre outros
P3	Não utilizado
P4	<i>Time-out</i> em milissegundos. Recomenda-se utilizar o <i>time-out</i> padrão, já que o fabricante sugere que os tempos de resposta de um painel são de até 1 (um) segundo

Caso um parâmetro esteja zerado, usa-se a configuração presente na janela de propriedades deste Driver.

## Configuração das Propriedades do Driver

A aba **Protec** contém configurações específicas para este Driver.



**Aba Protec**

As opções disponíveis nesta aba estão descritas na tabela a seguir.

**Opções disponíveis na aba Protec**

OPÇÃO	DESCRIÇÃO
<b>Model</b>	Modelo do equipamento. Os valores possíveis para esta opção são <b>AN95/6300</b> , <b>X400</b> ou <b>6500/6600</b>
<b>Protocol</b>	Protocolo de comunicação. Os valores possíveis para esta opção são <b>RS232</b> , <b>CST Pager</b> , <b>ASCOM Pager</b> ou <b>Scope Pager</b> . O protocolo <b>RS232</b> está disponível para todos os modelos e os protocolos <b>CST Pager</b> , <b>ASCOM Pager</b> e <b>Scope Pager</b> estão disponíveis apenas para o modelo <b>6500/6600</b>

## Referência de Tags por Protocolo

Esta seção contém informações sobre a configuração dos Tags [N/B] deste Driver para cada protocolo suportado.

### Protocolo RS232

Esta seção contém informações sobre a configuração dos Tags [N/B] deste Driver ao selecionar o protocolo **RS232**.

### Parâmetros [N]

Tags PLC são utilizados apenas para enviar requisições a um painel, através de operações de escrita de um Tag. Os Tags PLC deste Driver não aceitam operações de leitura. As requisições podem ser **Simples** ou **Normais**.

Para enviar uma requisição **Simples**, utilize um Tag PLC com a configuração indicada na tabela a seguir.

### Configuração para requisições Simples

PARÂMETRO	DESCRIÇÃO
<b>N1</b>	0 (zero, identificador de requisições <b>Simples</b> )
<b>N2</b>	Código da requisição. Os valores possíveis são <b>0</b> : Requisita a retransmissão de todos os eventos correntes ("#E*"), <b>1</b> : Requisita que o painel silencie ("#X*"), <b>2</b> : Requisita a evacuação ("#Y*"), <b>3</b> : Requisita o <i>reset</i> do painel ("#Z*") ou <b>4</b> : Requisita que o painel aceite todos os eventos ("#W*", apenas para o modelo <b>X400</b> )
<b>N3</b>	Não utilizado
<b>N4</b>	Não utilizado

Para enviar uma requisição **Normal** ("#1eepplaaazzccc\*"), utilize um Tag PLC com a configuração indicada na tabela a seguir.

### Configuração para requisições Normais

PARÂMETRO	DESCRIÇÃO
<b>N1</b>	Número do evento, entre 1 (um) e 99
<b>N2</b>	Número do painel ou nó, entre 0 (zero) e 999
<b>N3</b>	Número do <i>loop</i> (l, entre zero e 99) mais o número da zona (entre zero e 9999) multiplicado por 10
<b>N4</b>	Número do endereço, entre 0 (zero) e 999

Para enviar uma requisição, utilize a escrita de qualquer valor neste Tag PLC ou utilize o método **Write** do Tag PLC.

## Parâmetros [B]

Este Driver permite a utilização de um Tag Bloco para leitura dos eventos. Este Tag Bloco deve ter todos os parâmetros [B] zerados e com um tamanho de 169 Elementos.

<b>B1</b>	0 (zero)
<b>B2</b>	0 (zero)
<b>B3</b>	0 (zero)
<b>B4</b>	0 (zero)
<b>Size</b>	169

Este Tag Bloco deve estar configurado com a escrita desabilitada. Os Tags Bloco retornam os eventos em lista, logo podem retornar 0 (zero) ou mais eventos. Para tratar corretamente estes eventos, deve-se utilizar o evento **OnRead** de um Tag Bloco, que é chamado para cada evento enviado pelo painel. No script deste evento, os Elementos de um Tag Bloco podem ser acessados para determinar que tipo de evento foi gerado, conforme a tabela a seguir.

## Tipos de eventos

ELEMENTO	DESCRIÇÃO
0 (zero)	1 (um) para novo evento ou 0 (zero) para <i>event cleared</i>
1 (um)	Um Inteiro com o número de um evento, entre 0 (zero) e 99 ou entre 160 e 172 ou entre 176 e 183
2 (dois)	Um Inteiro com o número do painel ou nó, entre 0 (zero) e 999
3 (três)	Um Inteiro com o número do <i>loop</i> , entre 0 (zero) e 99
4 (quatro)	Um Inteiro com o endereço, entre 0 (zero) e 999
5 (cinco)	Um Inteiro com o número da zona, entre 0 (zero) e 9999
6 (seis)	<b>String</b> com o texto enviado em uma mensagem
7 (sete)	<b>String</b> com o texto explicativo de um evento
8 (oito)	0 (zero) se não possui dados extras ou 1 (um) se possui dados extras, nos Elementos entre 9 (nove) e 169
9 (nove)	Um Inteiro com o código ASCII do primeiro byte de dados extras
10	Um Inteiro com o código ASCII do segundo byte de dados extras
11	Um Inteiro com o código ASCII do terceiro byte de dados extras
...	...
169	Um Inteiro com o código ASCII do centésimo sexagésimo byte de dados extras

## Protocolo CST

### Parâmetros [B] de Endereçamento para Tags Bloco

Este Driver permite a utilização de um Tag Bloco para leitura dos eventos. Este Tag Bloco deve ter todos os parâmetros [B] zerados e com um tamanho de 4 (quatro) Elementos.

B1	0 (zero)
B2	0 (zero)
B3	0 (zero)
B4	0 (zero)
Size	4 (quatro)

Este Tag Bloco deve estar configurado com a escrita desabilitada. Os Tags Bloco retornam os eventos em lista, logo podem retornar 0 (zero) ou mais eventos. Para tratar corretamente estes eventos, deve-se utilizar o evento **OnRead** de um Tag Bloco, que é chamado para cada evento enviado pelo painel. No script deste evento, os Elementos de um Tag Bloco podem ser acessados para determinar que tipo de evento foi gerado, conforme a tabela a seguir.

## Tipos de eventos

ELEMENTO	DESCRIÇÃO
0 (zero)	Um Inteiro indicando o <i>cap code</i>
1 (um)	Uma <b>String</b> indicando o evento
2 (dois)	Um Inteiro indicando o número da zona
3 (três)	Uma <b>String</b> enviada na mensagem

## Protocolo ASCOM

### Parâmetros [B] de Endereçamento para Tags Bloco

Este Driver permite a utilização de um Tag Bloco para leitura dos eventos. Este Tag Bloco deve ter todos os parâmetros [B] zerados e com um tamanho de 3 (três) Elementos.

B1	0 (zero)
B2	0 (zero)
B3	0 (zero)
B4	0 (zero)
Size	3 (três)

Este Tag Bloco deve estar configurado com a escrita desabilitada. Os Tags Bloco retornam os eventos em lista, logo podem retornar 0 (zero) ou mais eventos. Para tratar corretamente estes eventos, deve-se utilizar o evento **OnRead** de um Tag Bloco, que é chamado para cada evento enviado pelo painel. No script deste evento, os Elementos do Tag Bloco podem ser acessados para determinar que tipo de evento foi gerado, conforme a tabela a seguir.

## Tipos de eventos

ELEMENTO	DESCRIÇÃO
0 (zero)	Uma <b>String</b> indicando o evento
1 (um)	Um Inteiro indicando o número da zona
2 (dois)	Uma <b>String</b> enviada na mensagem

## Protocolo Scope

### Parâmetros [B] de Endereçamento para Tags Bloco

Este Driver permite a utilização de um Tag Bloco para leitura dos eventos. Este Tag Bloco deve ter todos os parâmetros [B] zerados e com um tamanho de 4 (quatro) Elementos.

<b>B1</b>	0 (zero)
<b>B2</b>	0 (zero)
<b>B3</b>	0 (zero)
<b>B4</b>	0 (zero)
<b>Size</b>	4 (quatro)

Este Tag Bloco deve estar configurado com a escrita desabilitada. Os Tags Bloco retornam os eventos em lista, logo podem retornar 0 (zero) ou mais eventos. Para tratar corretamente estes eventos, deve-se utilizar o evento **OnRead** de um Tag Bloco, que é chamado para cada evento enviado pelo painel. No script deste evento, os Elementos do Bloco podem ser acessados para determinar que tipo de evento foi gerado, conforme a tabela a seguir.

#### Tipos de eventos

ELEMENTO	DESCRIÇÃO
<b>0 (zero)</b>	Um Inteiro indicando o identificador
<b>1 (um)</b>	Uma <b>String</b> indicando o evento
<b>2 (dois)</b>	Um Inteiro indicando o número da zona
<b>3 (três)</b>	Uma <b>String</b> enviada na mensagem


## Documentação das Interfaces de Comunicação

Esta seção contém a documentação das Interfaces de Comunicação referentes ao Driver **Protec**.

### 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 (IODevice).
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

Aba Setup

#### Opções gerais da aba Setup

OPÇÃO	DESCRIÇÃO
<b>Physical Layer</b>	Selecione a interface física em uma lista. As opções são <b>Serial</b> , <b>Ethernet</b> , <b>Modem</b> e <b>RAS</b> . A interface selecionada deve ser configurada na sua aba específica
<b>Timeout</b>	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
<b>Start driver OFFLINE</b>	Selecione esta opção para que o Driver inicie em modo <b>Offline</b> ou parado. Isto significa que a interface de I/O não é criada até que se configure o Driver em modo <b>Online</b>

OPÇÃO	DESCRIÇÃO
	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
<b>Mode</b>	Seleciona o modo de gerenciamento de conexão. Selecionar a opção <b>Automatic</b> permite que o Driver gerencie a conexão automaticamente, como especificado nas opções seguintes. Selecionar a opção <b>Manual</b> permite que a aplicação gerencie a conexão completamente. Consulte o tópico <b>Estados do Driver</b> para mais informações
<b>Retry failed connection every ... seconds</b>	Selecione esta opção para habilitar a retentativa de conexão do Driver em um determinado intervalo, em segundos. Se a opção <b>Give up after failed retries</b> não está selecionada, o Driver continua retentando até que a conexão seja efetuada, ou que a aplicação seja parada
<b>Give up after ... failed retries</b>	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 <b>Offline</b> , 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
<b>Disconnect if non-responsive for ... seconds</b>	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 <b>Timeout</b>

Opções para o grupo Logging Options

OPÇÃO	DESCRIÇÃO
<p><b>Log to File</b></p>	<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 <b>%PROCESS%</b> 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 <b>E3</b>, permitindo assim que cada instância gere um arquivo separado de log. Por exemplo, ao configurar esta opção como <b>c:\e3logs\drivers\sim_%PROCESS%.log</b>, gera-se o arquivo <b>c:\e3logs\drivers\sim_00000FDA.log</b> para o processo <b>0FDAh</b>.</p> <p>Pode-se também utilizar a macro <b>%DATE%</b> no nome do arquivo. Neste caso é gerado um arquivo de log por dia (no formato <b>aaaa_mm_dd</b>). Por exemplo, ao configurar esta opção como <b>c:\e3logs\drivers\sim_%DATE%.log</b>, gera-se o arquivo <b>c:\e3logs\drivers\sim_2005_12_31.log</b> no dia 31 de dezembro de 2005 e o arquivo <b>c:\e3logs\drivers\sim_2006_01_01.log</b> 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):   $\mu$ s

Inter-frame delay (milliseconds):  ms

**Aba Serial**

## Opções gerais da aba Serial

OPÇÃO	DESCRIÇÃO
<b>Port</b>	Selecione a porta serial a partir da lista, de <b>COM1</b> até <b>COM4</b> , ou digite o nome da porta serial no formato <b>COMn</b> , 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"
<b>Baud rate</b>	Selecione o <i>baud rate</i> a partir da lista ( <b>1200, 2400, 4800, 9600, 19200, 38400, 57600</b> ou <b>115200</b> ) ou digite o <i>baud rate</i> desejado, como por exemplo 600
<b>Data bits</b>	Selecione 7 (sete) ou 8 (oito) bits de dados a partir da lista
<b>Parity</b>	Selecione a paridade a partir da lista. As opções disponíveis são <b>None, Even, Odd, Mark</b> ou <b>List</b>
<b>Stop bits</b>	Selecione o número de stop bits a partir da lista. As opções disponíveis são <b>1, 1.5</b> ou <b>2</b> stop bits
<b>Enable 'ECHO' supression</b>	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
<b>Inter-byte delay (microseconds)</b>	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
<b>Inter-frame delay (milliseconds)</b>	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 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
<b>DTR control</b>	Selecione <b>ON</b> para deixar o sinal <b>DTR</b> sempre ligado enquanto a porta serial está aberta. Selecione <b>OFF</b> para desligar o sinal <b>DTR</b> enquanto a porta serial está aberta. Alguns equipamentos exigem que o sinal <b>DTR</b> esteja ligado para permitir a comunicação
<b>RTS control</b>	Selecione <b>ON</b> para deixar o sinal <b>RTS</b> sempre ligado enquanto a porta serial está aberta. Selecione <b>OFF</b> para desligar o sinal <b>RTS</b> enquanto a porta serial está aberta. Selecione <b>Toggle</b> para ligar o sinal <b>RTS</b> enquanto se envia os bytes através da porta serial, e desligá-lo quando não se está enviando bytes e, portanto, habilitando a recepção

OPÇÃO	DESCRIÇÃO
<b>Wait for CTS before send</b>	Disponível apenas quando a opção <b>RTS control</b> está configurada para <b>Toggle</b> . Utilize esta opção para forçar o Driver a verificar o sinal <b>CTS</b> antes de enviar os bytes através da porta serial, após ligar o sinal de <b>RTS</b> . Neste modo o sinal <b>CTS</b> é tratado como um <i>flag</i> de permissão para envio
<b>CTS timeout</b>	Determina o tempo máximo, em milissegundos, que o Driver aguarda pelo sinal de <b>CTS</b> depois de ligar o sinal de <b>RTS</b> . Se o sinal de <b>CTS</b> não é levantado dentro deste <i>time-out</i> , o Driver falha a comunicação atual e retorna erro
<b>Delay before send</b>	Alguns hardwares de porta serial demoram a habilitar o circuito de envio de dados depois que o sinal <b>RTS</b> é ligado. Configure esta opção para aguardar uma determinada quantidade de milissegundos depois de ligar o sinal <b>RTS</b> e antes de enviar o primeiro byte. <b>IMPORTANTE:</b> 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
<b>Delay after send</b>	Tem o mesmo efeito que a opção <b>Delay before send</b> , mas neste caso a espera é efetuada depois que o último byte é enviado, antes de desligar o sinal <b>RTS</b>

## 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' supression

IP Filter:

Connect to

<input type="checkbox"/> Main IP: <span style="border: 1px solid gray; padding: 2px;"> </span>	Port: <span style="border: 1px solid gray; padding: 2px;">502</span>	<input type="checkbox"/> Local port: <span style="border: 1px solid gray; padding: 2px;">0</span>
<input type="checkbox"/> Backup IP 1: <span style="border: 1px solid gray; padding: 2px;"> </span>	Port: <span style="border: 1px solid gray; padding: 2px;">0</span>	<input type="checkbox"/> Local port: <span style="border: 1px solid gray; padding: 2px;">0</span>
<input type="checkbox"/> Backup IP 2: <span style="border: 1px solid gray; padding: 2px;"> </span>	Port: <span style="border: 1px solid gray; padding: 2px;">0</span>	<input type="checkbox"/> Local port: <span style="border: 1px solid gray; padding: 2px;">0</span>
<input type="checkbox"/> Backup IP 3: <span style="border: 1px solid gray; padding: 2px;"> </span>	Port: <span style="border: 1px solid gray; padding: 2px;">0</span>	<input type="checkbox"/> Local port: <span style="border: 1px solid gray; padding: 2px;">0</span>

**Aba Ethernet**

### Opções disponíveis na aba Ethernet

OPÇÃO	DESCRIÇÃO
<b>Transport</b>	Selecione <b>TCP/IP</b> para um <i>socket</i> TCP ( <i>stream</i> ) ou selecione <b>UDP/IP</b> para utilizar um <i>socket</i> UDP ( <i>connectionless datagram</i> )
<b>Listen for connections on port</b>	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 <b>Connect to</b>
<b>Share listen port with other processes</b>	Selecione esta opção para compartilhar a porta de escuta com outros Drivers e processos
<b>Interface</b>	Selecione a interface de rede local, identificada por seu endereço IP, que o Driver utiliza para efetuar e receber conexões, ou selecione o item ( <b>All Interfaces</b> ) para permitir conexões em qualquer interface de rede
<b>Use IPv6</b>	Selecione esta opção para forçar o Driver a utilizar endereços no formato <b>IPv6</b> em todas as conexões Ethernet. Deixe esta opção desmarcada para utilizar o formato <b>IPv4</b>
<b>Enable 'ECHO' suppression</b>	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
<b>IP Filter</b>	Lista de endereços IP restringidos ou permitidos de onde o Driver aceita conexões ( <i>Firewall</i> ). Consulte a propriedade <b>IO.Ethernet.IPFilter</b> para mais informações
<b>PING before connecting</b>	Habilite esta opção para executar um comando <b>ping</b> , ou seja, verificar se o dispositivo pode ser encontrado na rede, 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: <ul style="list-style-type: none"> <li>• <b>Timeout:</b> Especifique o número de milissegundos de espera por uma resposta do comando <b>ping</b>. Deve-se usar o comando <b>ping</b> 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</li> <li>• <b>Retries:</b> Número de tentativas do comando <b>ping</b>, sem contar a tentativa inicial. Se todas as tentativas falharem, então a conexão com o <i>socket</i> é abortada</li> </ul>

### Opções disponíveis no grupo Connect to

OPÇÃO	DESCRIÇÃO
<b>Main IP</b>	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

OPÇÃO	DESCRIÇÃO
	disponível para mapear a URL para um endereço IP, como por exemplo "192.168.0.13" ou "Server1"
<b>Port</b>	Digite a porta IP do dispositivo remoto, entre 0 (zero) e 65535
<b>Local port</b>	Selecione esta opção para utilizar uma porta IP local fixa ao conectar ao dispositivo remoto
<b>Backup IP 1, 2 e 3</b>	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

## 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

<b>Tipo de Tag</b>	Tag Bloco
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro B1</b>	-1 (menos um)
<b>Parâmetro B2</b>	0 (zero)
<b>Parâmetro B3</b>	0 (zero)
<b>Parâmetro B4</b>	1 (um)
<b>Propriedade Size</b>	4 (quatro)
<b>Propriedade ParamItem</b>	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

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	2 (dois)
<b>Configuração por String</b>	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

<b>Tipo de Tag</b>	Tag Bloco
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro B1</b>	-1 (menos um)
<b>Parâmetro B2</b>	0 (zero)
<b>Parâmetro B3</b>	0 (zero)
<b>Parâmetro B4</b>	3 (três)
<b>Propriedade Size</b>	2 (dois)
<b>Propriedade ParamItem</b>	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

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Leitura ou Escrita
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	4 (quatro)
<b>Configuração por String</b>	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

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

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

#### IO.Stats.Partial.BytesSent

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

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

## IO.Stats.Partial.TimeConnectedSeconds

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1102
<b>Configuração por String</b>	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

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1103
<b>Configuração por String</b>	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

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

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

## IO.Stats.Total.BytesSent

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

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

## IO.Stats.Total.ConnectionCount

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1004
<b>Configuração por String</b>	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

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

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

## IO.Stats.Total.TimeDisconnectedSeconds

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Leitura
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	0 (zero)
<b>Parâmetro N4</b>	1003
<b>Configuração por String</b>	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

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Leitura ou Escrita
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	4 (quatro)
<b>Parâmetro N4</b>	0 (zero)
<b>Configuração por String</b>	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

<b>Tipo de Tag</b>	Tag de Comunicação
<b>Tipo de Acesso</b>	Somente Escrita
<b>Parâmetro N1</b>	-1 (menos um)
<b>Parâmetro N2</b>	0 (zero)
<b>Parâmetro N3</b>	4 (quatro)
<b>Parâmetro N4</b>	1 (um)
<b>Configuração por String</b>	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 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

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

### IO.Serial.CTSTimeoutMs

9 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
2.0.2	15/10/2024	A. Fetzner	<ul style="list-style-type: none"> <li>Driver atualizado conforme a nova revisão 21, de julho</li> </ul>

VERSÃO	DATA	AUTOR	COMENTÁRIOS
			de 2020, da especificação do protocolo ( <i>Case 36613</i> ).
<b>2.0.1</b>	18/08/2023	A. Fetzner	<ul style="list-style-type: none"><li>• Adicionado suporte para os protocolos de comunicação <b>CST, ASCOM e Scope</b> (<i>Case 34591</i>).</li><li>• Driver atualizado de acordo com o padrão <b>IOKit</b> (<i>Case 34592</i>).</li></ul>
<b>1.1.1</b>	16/12/2010	A. Quites	<ul style="list-style-type: none"><li>• Driver atualizado conforme a nova revisão 13, de junho de 2009, da especificação do protocolo (<i>Case 11652</i>).</li></ul>
<b>1.0.1</b>	04/03/2001	F. Englert	<ul style="list-style-type: none"><li>• Todas as versões anteriores ao controle de revisões.</li></ul>

**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](mailto: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](mailto: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](mailto: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](mailto: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](mailto: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](mailto:evan@elipse.com.br)

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

[www.elipse.com.br](http://www.elipse.com.br)

[kb.elipse.com.br](http://kb.elipse.com.br)

[forum.elipse.com.br](http://forum.elipse.com.br)

[www.youtube.com/elipsesoftware](http://www.youtube.com/elipsesoftware)

[elipse@elipse.com.br](mailto: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)