

Driver B. Braun AutoProgramming

| | |
|--------------------------------|---------------------|
| Nome do Arquivo | AutoProgramming.dll |
| Fabricante | B. Braun |
| Equipamentos | SpaceCom |
| Protocolo | AutoProgramming |
| Versão | 1.0.2 |
| Última Atualização | 12/06/2025 |
| Plataforma | Win32 |
| Dependências | IOKit versão 2.0 |
| Leitura com Superblocos | Não |
| Nível | 0 |

Introdução

Este Driver implementa a funcionalidade AutoProgramming, permitindo a uma aplicação desenvolvida pela **Elipse Software** comunicar com os equipamentos SpaceCom da B. Braun.

Funcionamento do Driver

Para que o Driver B. Braun AutoProgramming seja utilizado adequadamente em uma aplicação, é importante entender como funciona, porque este Driver não tem as características mais comuns aos demais Drivers de Comunicação.

O equipamento B. Braun SpaceCom fornece uma funcionalidade onde um sistema externo pode receber a capacidade de enviar conjuntos de dados de configuração de terapias de pacientes. Para tanto, o equipamento possui uma porta TCP/IP dedicada a recepção de uma lista de dados no formato **XML** (*Extensible Markup Language*), ao qual pode-se referir como **Documento XML**. Este documento XML pode ser constituído de uma lista de até 24 configurações terapêuticas, e cada um dos itens desta lista pode receber um certo conjunto de parâmetros que determina esta configuração. Para constituir um documento XML de configurações de terapias é necessário, portanto, que um certo número de valores seja configurado por alguma forma de edição de formulários. Uma aplicação da **Elipse Software** atuando conjuntamente com este Driver pode se tornar portanto um sistema de gestão de terapias de pacientes, editando formulários e aplicando o preenchimento de valores na formação de um documento XML a ser enviado a uma determinada unidade de terapia.

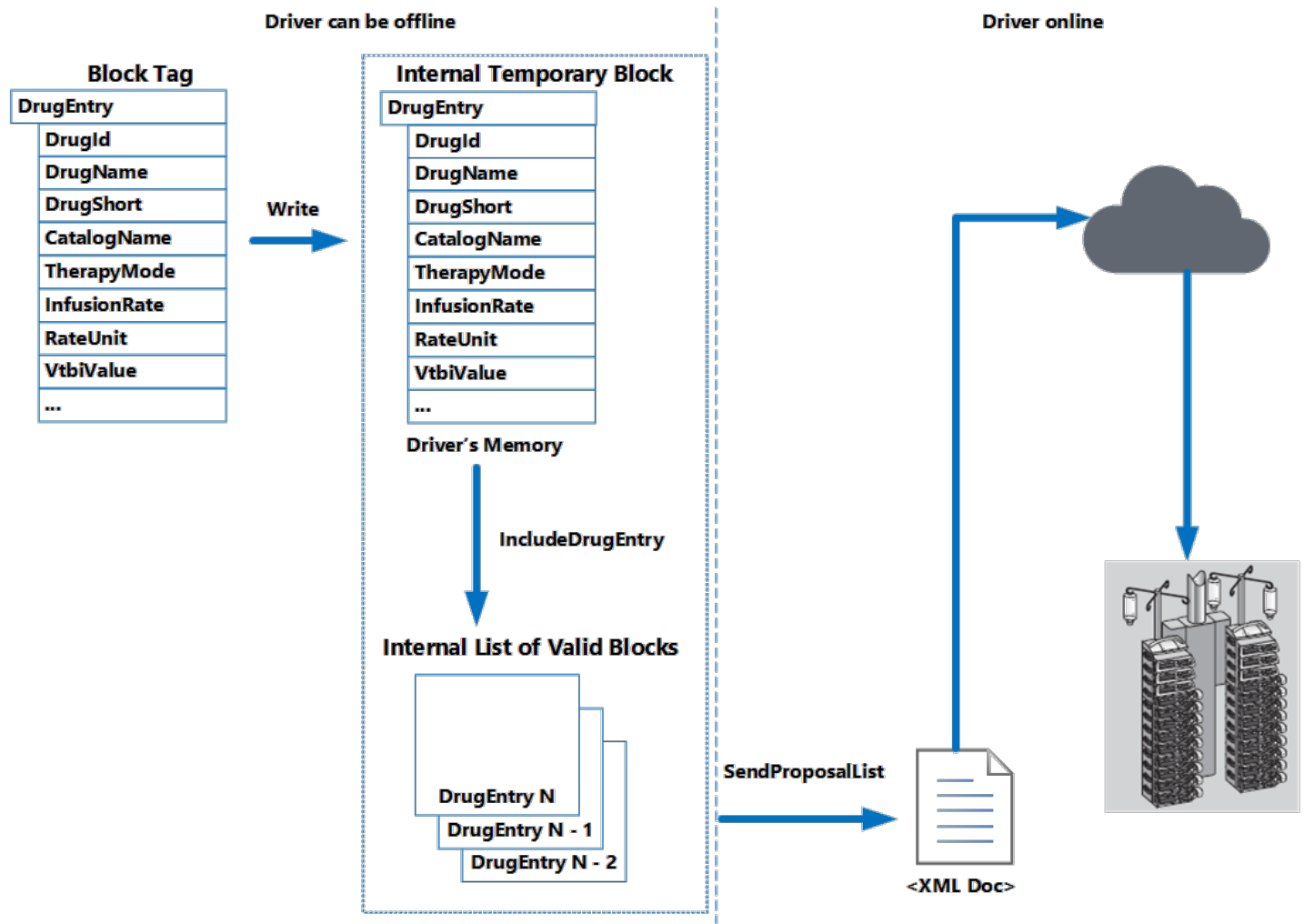
Pela natureza desta funcionalidade de AutoProgramming da máquina do fabricante B. Braun, basicamente a maior parte da atividade de processamento deste Driver concentra-se na formação do documento XML a ser enviado do que propriamente no processo de comunicação.

O funcionamento deste Driver pode ser reduzido à operação da sequência de escritas ordenadas essenciais, ou seja, a escrita do Tag Bloco **DrugEntry**, a escrita da operação **IncludeDrugEntry**, usando o Tag **DrugEntryOperation**, e a escrita do Tag **SendProposalList** para comandar a formação e envio do documento XML ao equipamento remoto. Até o momento anterior da chamada ao comando **SendProposalList**, este Driver pode ser iniciado em modo *offline*, isto é, sem iniciar a interface de comunicação e conectar com o dispositivo remoto, pois antes de enviar os dados este Driver trabalha apenas com dados em memória. No entanto, no momento em que o comando **SendProposalList** é chamado, este Driver já deve estar no estado *online*, ou seja, conectado ao dispositivo remoto.

Na memória interna deste Driver operam duas formas de contenção de dados, ou contêineres de dados, para a formação de um documento XML. Primeiramente, há um bloco temporário que é memorizado pela escrita do Tag Bloco **DrugEntry**. Supondo que o Tag Bloco esteja corretamente preenchido, este pode ser copiado para um segundo contêiner que guarda por ordem de escrita uma lista de blocos válidos após cada operação **IncludeDrugEntry**. Desta

forma um documento XML pode ser composto por até 24 blocos, realizando as operações de escrita de bloco **DrugEntry** e a operação **IncludeDrugEntry** tantas vezes quantas sejam necessárias até finalizar o documento XML e executar a escrita do Tag **SendProposalList**. Após a execução desta escrita, este Driver comunica o conjunto de configurações de terapia ao dispositivo remotamente ligado pela conexão Ethernet TCP/IP.

A figura a seguir ilustra estas operações elementares de funcionamento deste Driver. Outras operações mais secundárias podem ser executadas por outros Tags ainda não mencionados, mas que podem ser úteis de acordo com as intenções de quem projeta a aplicação.



Funcionamento do Driver B. Braun AutoProgramming

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. Para mais informações sobre as abas de configuração, consulte o tópico **Documentação das Interfaces de Comunicação**.

Configuração das Propriedades

Todas as configurações são executadas na caixa de diálogo de configurações deste Driver. Nesta caixa de diálogo, os parâmetros mais relevantes estão descritos na tabela a seguir.

Opções de configuração do Driver B. Braun AutoProgramming

| ABA | PARÂMETRO | STRING OFFLINE | TIPO DE DADOS | DESCRIÇÃO |
|-------|----------------|----------------|---------------|---|
| Setup | Physical Layer | IO.Type | Texto | Por padrão, utilize a opção Ethernet |

| ABA | PARÂMETRO | STRING OFFLINE | TIPO DE DADOS | DESCRIÇÃO |
|-----------------|-------------------------------|-----------------------|---------------|--|
| | Timeout | IO.TimeoutMs | Número | Limite de tempo, em milissegundos, para recepção dos dados na resposta de um equipamento. Por exemplo, o valor 1000 define um limite de 1 (um) segundo |
| | Logging options / Log to File | IO.Log.Enable | Número | Habilita a geração de log e nome do arquivo de log a ser gerado para depuração de problemas na comunicação. O valor 0 (zero) desabilita a geração de log e qualquer valor diferente de 0 (zero) habilita os logs |
| | | IO.Log.Filename | Texto | |
| Ethernet | Transport | IO.Ethernet.Transport | Texto | Por padrão, utilize a opção TCP |
| | Main IP | IO.Ethernet.MainIP | Texto | Endereço IP de um equipamento, no formato [0-255].[0-255].[0-255].[0-255] |
| | Port | IO.Ethernet.MainPort | Número | Por padrão, utilize o valor 4002 |

Além da Janela de Propriedades, estas configurações também podem ser definidas em tempo de execução em aplicações **Elipse E3**, **Elipse Power** ou **Elipse Water**. Para isto, inicialize este Driver em modo **Offline**, ou seja, execute a aplicação com a opção **Start driver OFFLINE** habilitada, configurável na aba **Setup** da Janela de Propriedades, descrita anteriormente.

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

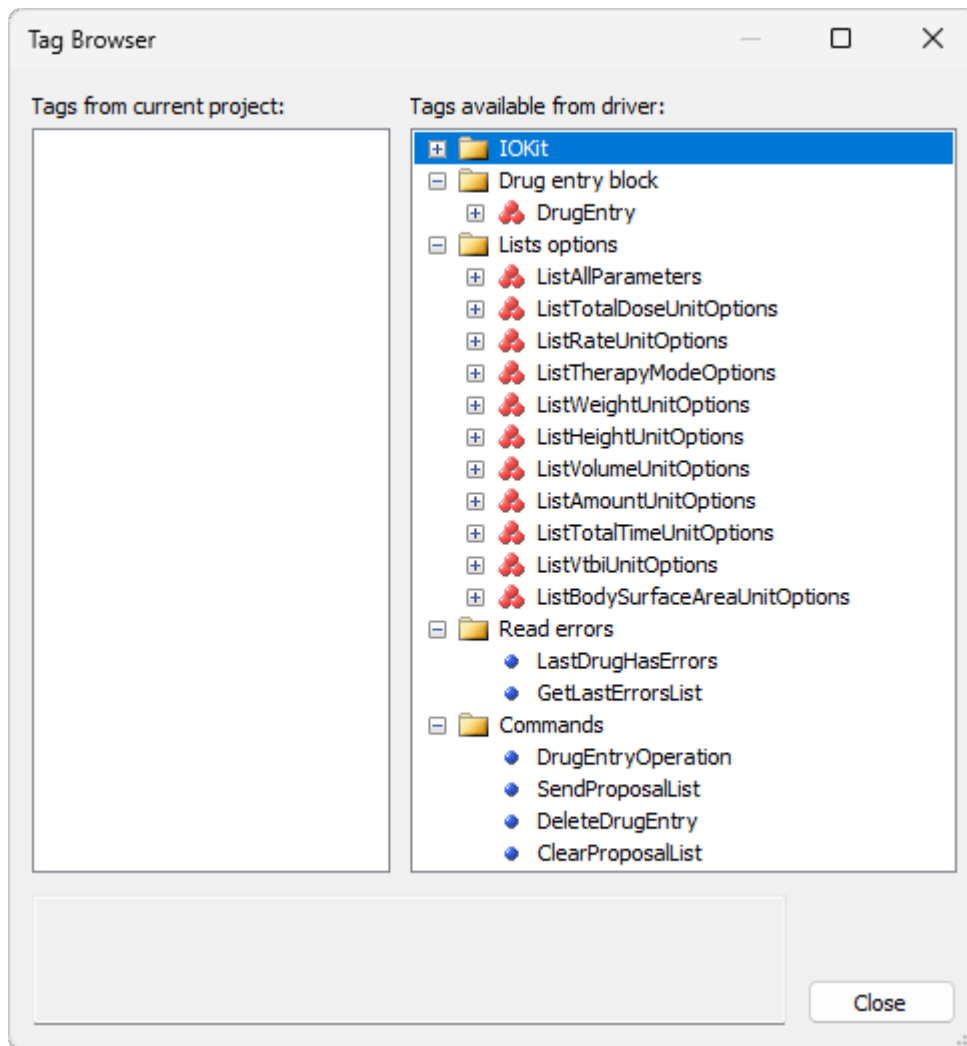
Esta seção contém informações sobre a configuração dos Tags deste Driver. Não são utilizados parâmetros *N* ou *B*. Apenas o parâmetro *Item* é usado para referir qual Tag é executado.

Qualquer palavra não prevista como nome de Tag no parâmetro *Item* gera um erro de falha de preenchimento. Parâmetros *N* ou *B* preenchidos com valores diferentes de 0 (zero) também geram um erro de falha de preenchimento.

Utilizando o Tag Browser

O **Elipse E3** versão **2.0**, o **Elipse Power** e o **Elipse Water** possuem uma ferramenta chamada **Tag Browser**, que permite que um Driver auxilie o usuário na criação e configuração de Tags.

O Tag Browser visa automatizar a configuração dos Tags preenchendo automaticamente o campo **Item** e o tamanho em Elementos adequado, caso seja um Tag Bloco.



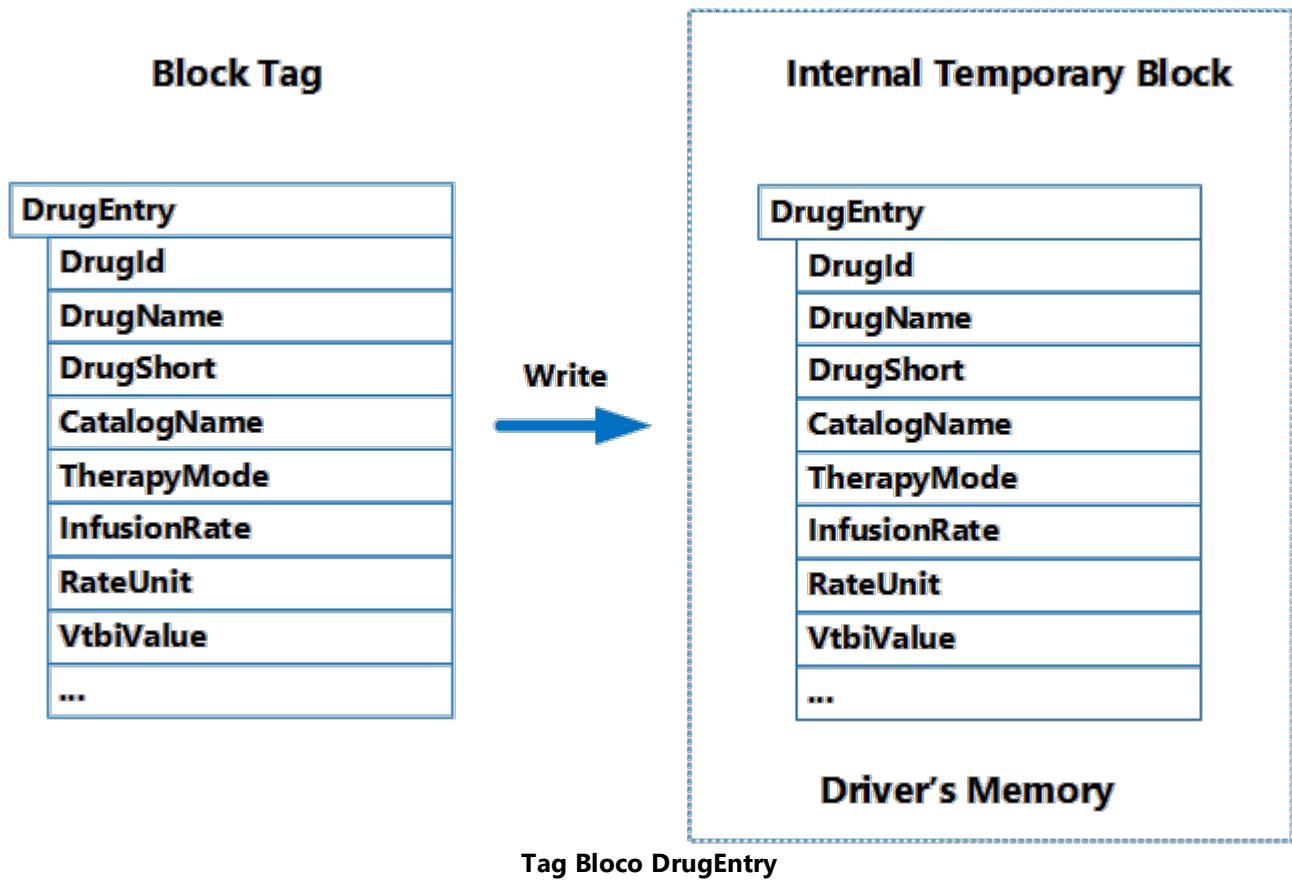
Janela Tag Browser

A listagem **Tags do projeto corrente** (*Tags from current project*) mostra os Tags e pastas que existem no projeto corrente. A listagem **Tags disponibilizados pelo driver** (*Tags available from driver*) mostra uma árvore com os Tags disponíveis no Driver. Para criar um Tag novo na aplicação, arraste um dos Tags definidos pelo Driver para a pasta desejada no diretório corrente. O Driver B. Braun AutoProgramming contém os seguintes nós nesta árvore:

- O nó **IOKit** mostra os Tags disponíveis na biblioteca **IOKit**, divididos nas seguintes categorias:
 - **General:** Tags de uso geral
 - **Modem:** Tags de manipulação da comunicação via modem
 - **Ethernet:** Tags de manipulação da comunicação via rede Ethernet
 - **Parameters:** Tags para configuração dos parâmetros da biblioteca **IOKit**
- O nó **Drug entry block** possui o Tag Bloco de escrita e leitura **DrugEntry**
- O nó **Lists options** possui todos os Tags Bloco de **listagem de opções**
- O nó **Read errors** possui todos os Tags PLC de **leitura de erros de configuração**
- O nó **Commands** possui todos os Tags PLC de **comandos de escrita**

Tag de Configuração de Terapia

O Tag Bloco **DrugEntry** realiza a configuração de terapia desejada e a escrita neste Tag insere os valores dentro da memória deste Driver como um novo candidato de configuração a ser colocado na lista interna de configurações de terapia.



DrugEntry

Leitura e Escrita

Este Tag Bloco deve conter 27 Elementos. Todos os nomes e regras de preenchimento de cada um destes Elementos estão descritos na tabela a seguir. Dependendo da operação realizada, leitura ou escrita, este Tag age em conformidade com os comportamentos descritos a seguir.

Escrita

A escrita deve ser realizada na operação de bloco inteiro, ou seja, a escrita de apenas Elementos de Bloco não é permitida. Esta operação armazena em memória interna um candidato a novo item **Drug**, que é processado conforme o comando selecionado nas operações do Tag **DrugEntryOperation**.

Leitura

Exibe todos os itens **Drug** armazenados na lista de memória interna deste Driver. Esta leitura retorna uma lista de N blocos de itens **Drug**, onde N é o número de itens memorizados.

Formatos do Tag Bloco DrugEntry

| ELEMENTO | PARÂMETRO | FORMATO | REGRA |
|-------------|----------------------|--|------------------------------------|
| Elemento 1 | DrugId | String | Opcional |
| Elemento 2 | DrugName | String | Consulte a Regra 1 (um) |
| Elemento 3 | DrugShort | String com um máximo de 8 (oito) caracteres | Consulte a Regra 1 (um) |
| Elemento 4 | CatalogName | String | Opcional |
| Elemento 5 | TherapyMode | ListTherapyModeOptions | Opcional |
| Elemento 6 | InfusionRate | Número com ponto decimal | Opcional |
| Elemento 7 | RateUnit | ListRateUnitOptions | Consulte a Regra 2 (dois) |
| Elemento 8 | VtbiValue | Número com ponto decimal | Consulte a Regra 3 (três) |
| Elemento 9 | VtbiUnit | ListVtbiUnitOptions | Consulte a Regra 2 (dois) |
| Elemento 10 | AmountValue | Número com ponto decimal | Opcional |
| Elemento 11 | AmountUnit | ListAmountUnitOptions | Consulte a Regra 2 (dois) |
| Elemento 12 | VolumeValue | Número inteiro positivo | Consulte a Regra 4 (quatro) |
| Elemento 13 | VolumeUnit | ListVolumeUnitOptions | Consulte a Regra 2 (dois) |
| Elemento 14 | TotalDose | Número com ponto decimal | Opcional |
| Elemento 15 | TotalDoseUnit | ListTotalDoseUnitOptions | Consulte a Regra 2 (dois) |
| Elemento 16 | TotalTime | Número inteiro positivo | Opcional |
| Elemento 17 | TotalTimeUnit | ListTotalTimeUnitOptions | Consulte a Regra 2 (dois) |
| Elemento 18 | PatientID | String | Opcional |
| Elemento 19 | NurseID | String | Opcional |
| Elemento 20 | HeightValue | Número com ponto decimal | Opcional |
| Elemento 21 | HeightUnit | ListHeightUnitOptions | Consulte a Regra 2 (dois) |
| Elemento 22 | WeightValue | Número com ponto decimal | Opcional |
| Elemento 23 | WeightUnit | ListWeightUnitOptions | Consulte a Regra 2 (dois) |
| Elemento 24 | BodySurfaceAreaValue | Número com ponto decimal | Opcional |
| Elemento 25 | BodySurfaceAreaUnit | ListBodySurfaceAreaUnitOptions | Consulte a Regra 2 (dois) |
| Elemento 26 | OrderNumber | String com um máximo de 19 caracteres numéricos | Opcional |
| Elemento 27 | PumpId | String | Consulte a Regra 5 (cinco) |

Tipos de Formato

Todos os Elementos devem ser preenchidos como **Strings**, obedecendo o formato indicado na tabela anterior.

- **Formato numérico com ponto decimal:** Pode ter casas decimais depois de um ponto. Por exemplo, 10.0 ou 10 são valores válidos
- **Formato numérico inteiro positivo:** Sem casas decimais. Por exemplo 10, 20 ou 1000 são valores válidos

Regras de Preenchimento

Não é preciso preencher todos os Elementos, pois vários são opcionais, mas deve-se observar as seguintes regras:

- **Regra 1:** Os Elementos **DrugName** ou **DrugShort** devem ser preenchidos. Apenas um destes Elementos deve ser preenchido e o outro deve ser omitido
- **Regra 2:** É obrigatório preencher este Elemento caso o Elemento anterior seja preenchido e deve ser omitido em caso contrário
- **Regra 3:** É opcional, porém se existe o preenchimento do Elemento **TotalTime**, deve ser omitido
- **Regra 4:** É obrigatório preencher se os Elementos **AmountValue** e **AmountUnit** estão preenchidos, senão deve ser omitido
- **Regra 5:** É opcional, mas ao preencher é necessário obedecer os prefixos ISP ou PSP, seguido de dígitos numéricos decimais

Escrita deste Tag Bloco por Script

O script a seguir serve como exemplo de como preencher um Array e escrevê-lo neste Tag Bloco.

```
Dim Arr(26) 'Array DrugList
Arr(1) = "ampicillin" 'DrugName
Arr(3) = "2749283" 'CatalogName
Arr(4) = "Intermittent" 'TherapyMode
Arr(5) = "200.0" 'InfusionRate
Arr(6) = "ml/h" 'RateUnit
Arr(7) = "100.0" 'VtbiValue
Arr(8) = "ml" 'VtbiUnit
Arr(9) = "1000.0" 'AmountValue
Arr(10) = "mg" 'AmountUnit
Arr(11) = "100.0" 'VolumeValue
Arr(12) = "ml" 'VolumeUnit
Arr(13) = "1000.0" 'TotalDose
Arr(14) = "mg" 'TotalDoseUnit
Arr(25) = "3488817" 'OrderNumber
Arr(26) = "ISP154133" 'PumpId
Set Driver = Application.GetObject("DriverAutoProgramming")
'Write DrugList
Driver.Item("DrugEntry").WriteEx Arr
```

Tags de Comandos de Escrita

Estes Tags PLC apenas de escrita realizam importantes operações de manipulação da memória interna, especialmente o Tag **DrugEntryOperation**, e de finalização e envio de um documento no formato **XML**, **SendProposalList**.

ClearProposalList

Somente Escrita

Este Tag PLC pode ser utilizado para limpar todas as inclusões de configurações de terapia, item **Drug**, que estão sendo memorizados internamente neste Driver antes do envio ao equipamento pelo Tag **SendProposalList**. Qualquer valor pode ser escrito neste Tag para executá-lo.

DeleteDrugEntry

Somente Escrita

Tag PLC que recebe um valor inteiro numérico que comanda o apagamento de apenas um dos itens armazenados na memória interna deste Driver. Este valor numérico deve corresponder à posição guardada em memória. O primeiro item incluído pelo comando **IncludeDrugEntry** é o de valor 0 (zero), o segundo é o de valor 1 (um), e assim por diante. Ao apagar um item da lista interna, a posição de todos os itens é alterada, portanto o valor pretendido por um novo apagamento deve ser revisto de acordo com a situação atual da lista armazenada.

DrugEntryOperation

Somente Escrita

Este Tag recebe um valor inteiro numérico que determina qual operação realizar com o último Tag Bloco **DrugEntry** armazenado na memória interna deste Driver. Valores fora da faixa de valores prevista retornam erro de escrita.

| VALOR | OPERAÇÃO | COMANDO |
|-------|-------------------|--|
| 1 | ValidateDrugEntry | Validação da nova configuração de terapia, item Drug , atribuída na última escrita do Tag Bloco DrugEntry . Se há erros de preenchimento, este Driver indica nos Tags LastDrugHasErrors e GetLastErrorsList e no parâmetro de retorno <i>WriteStatus</i> , descrito a seguir |
| 2 | IncludeDrugEntry | Inclusão na memória interna deste Driver da nova configuração de terapia, item Drug , atribuída na última escrita do Tag Bloco DrugEntry . Valores inválidos não podem ser adicionados e a tentativa de inclusão com preenchimento de valores inválidos provoca o retorno de erro de escrita |

Parâmetro WriteStatus

Se este Tag é chamado para realizar a operação **ValidateDrugEntry** através de um script, o parâmetro *WriteStatus* pode ser verificado para checar a validade do último Tag Bloco **DrugEntry**, ou seja, um valor 0 (zero) não tem erros e um valor 1 (um) contém erros. Se esta funcionalidade é utilizada, não é necessário executar a leitura do Tag **LastDrugHasErrors** para descobrir a validade do Tag Bloco. O exemplo a seguir demonstra como utilizar esta funcionalidade.

```
'Write DrugList for validation
Driver.Item("DrugEntryOperation").WriteEx 1, Now, 192, WriteStatus
'Checks whether there is a fill error
If WriteStatus Then
  MsgBox "Fill error"
Else
  MsgBox "No errors"
End If
```

SendProposalList

Somente Escrita

Utilize este Tag para comandar o processamento de um documento no formato **XML** com cada um dos itens **Drug** memorizados internamente por este Driver e enviar a um equipamento. Quaisquer problemas na montagem deste documento retorna um erro de escrita e não o envia. Este Tag aguarda o retorno de um equipamento por resposta e,

dependendo da resposta dada por aquele equipamento, este Tag pode retornar algum erro ou sucesso de escrita. Qualquer valor pode ser escrito no campo deste Tag para executá-lo.

NOTAS

- A escrita neste Tag sem nenhum item **Drug** na memória interna deste Driver envia um documento vazio que indica um comando de apagamento das **ProposalLists** de um equipamento SpaceCom.
- A escrita neste Tag não apaga automaticamente o conteúdo da lista de blocos válidos na memória deste Driver. Se há intenção de enviar novas configurações de terapia, é necessário executar a escrita do Tag **ClearProposalList**.
- Este é o único Tag deste Driver que estabelece comunicação com um equipamento. Isto significa que este Driver pode permanecer *offline* durante toda a execução até o momento de executar este comando. Ao executar a escrita neste Tag, o estado *online* já deve ter sido iniciado e a conexão com um dispositivo remoto já deve ter sido inicializada. Este Driver pode retornar ao estado *offline* assim que este Tag terminar a execução.

Tags de Leitura de Erros de Configuração

São Tags PLC úteis para validar um Tag Bloco **DrugEntry** e, caso não seja válido, descobrir quais os erros de preenchimento.

Os valores contidos nestes Tags PLC sempre referem-se ao último Tag Bloco **DrugEntry** escrito após uma operação do Tag **DrugEntryOperation**, ou seja, enquanto não é realizada alguma nova operação do Tag **DrugEntryOperation**, o conteúdo destes Tags se referem ao conteúdo da operação anterior. O conteúdo destes Tags PLC apontam inexistência de erros até o momento em que a operação do Tag **DrugEntryOperation** é executada pela primeira vez.

GetLastErrorsList

Somente Leitura

Retorna uma única **String** contendo todos os erros de preenchimento dos parâmetros ao executar a última escrita do Tag **DrugEntryOperation**.

LastDrugHasErrors

Somente Leitura

Retorna o valor 0 (zero) se não houve nenhum erro ao executar a última escrita do Tag **DrugEntryOperation**. Se houve algum erro, retorna um valor diferente de 0 (zero).

Tags de Listagem de Opções

Estes Tags apenas de leitura são úteis para saber quais **Strings** são válidas para alguns campos de um Tag **DrugEntry**. São Tags Bloco que dependem da quantidade de opções em cada parâmetro. Estes Tags não são fundamentais para o funcionamento deste Driver, porém podem servir de ajuda para o preenchimento dos campos do Tag **DrugEntry**, de acordo com a conveniência do projetista.

ListAllParameters

Somente Leitura

Tag Bloco para listar todos os nomes dos parâmetros a serem preenchidos em cada Elemento de um Tag Bloco **DrugEntry**.

| PARÂMETRO | VALOR |
|-----------|-------------------|
| Item | ListAllParameters |
| Tamanho | 27 |

ListAmountUnitOptions

Somente Leitura

Lista todas as opções de preenchimento do Elemento **AmountUnit**.

| PARÂMETRO | VALOR |
|-----------|-----------------------|
| Item | ListAmountUnitOptions |
| Tamanho | 12 |

ListBodySurfaceAreaUnitOptions

Somente Leitura

Lista todas as opções de preenchimento do Elemento **BodySurfaceAreaUnit**.

| PARÂMETRO | VALOR |
|-----------|--------------------------------|
| Item | ListBodySurfaceAreaUnitOptions |
| Tamanho | 1 (um) |

ListHeightUnitOptions

Somente Leitura

Lista todas as opções de preenchimento do Elemento **HeightUnit**.

| PARÂMETRO | VALOR |
|-----------|-----------------------|
| Item | ListHeightUnitOptions |
| Tamanho | 3 (três) |

ListRateUnitOptions

Somente Leitura

Lista todas as opções de preenchimento do Elemento **RateUnit**.

| PARÂMETRO | VALOR |
|-----------|---------------------|
| Item | ListRateUnitOptions |

| PARÂMETRO | VALOR |
|-----------|-------|
| Tamanho | 79 |

ListTherapyModeOptions

Somente Leitura

Lista todas as opções de preenchimento do Elemento **TherapyMode**.

| PARÂMETRO | VALOR |
|-----------|------------------------|
| Item | ListTherapyModeOptions |
| Tamanho | 2 (dois) |

ListTotalDoseUnitOptions

Somente Leitura

Lista todas as opções de preenchimento do Elemento **TotalDoseUnit**.

| PARÂMETRO | VALOR |
|-----------|--------------------------|
| Item | ListTotalDoseUnitOptions |
| Tamanho | 32 |

ListTotalTimeUnitOptions

Somente Leitura

Lista todas as opções de preenchimento do Elemento **TotalTimeUnit**.

| PARÂMETRO | VALOR |
|-----------|--------------------------|
| Item | ListTotalTimeUnitOptions |
| Tamanho | 1 (um) |

ListVolumeUnitOptions

Somente Leitura

Lista todas as opções de preenchimento do Elemento **VolumeUnit**.

| PARÂMETRO | VALOR |
|-----------|-----------------------|
| Item | ListVolumeUnitOptions |
| Tamanho | 1 (um) |

ListVtbiUnitOptions

Somente Leitura

Lista todas as opções de preenchimento do Elemento **VtbiUnit**.

| PARÂMETRO | VALOR |
|-----------|---------------------|
| Item | ListVtbiUnitOptions |
| Tamanho | 1 (um) |

ListWeightUnitOptions

Somente Leitura

Lista todas as opções de preenchimento do Elemento **WeightUnit**.

| PARÂMETRO | VALOR |
|-----------|-----------------------|
| Item | ListWeightUnitOptions |
| Tamanho | 3 (três) |

Documentação das Interfaces de Comunicação

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

Configurações de um Driver

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

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

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

1. Abra o Organizer.
2. Selecione um 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, no caso de acesso a duas portas seriais, é preciso adicionar dois Drivers em um aplicação e configurar cada um destes Drivers para cada porta serial.

Caixa de Diálogo de Configuração

A caixa de diálogo das Interfaces de Configuração permite configurar a conexão de I/O que é utilizada por um 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 aquele Driver, na caixa de diálogo de configuração.

Aba Setup

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

- **Configurações gerais:** Configurações da camada física de um 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 configuration dialog. It is divided into three main sections:

- Physical Layer:** A dropdown menu is set to 'Ethernet'. To the right is a checkbox labeled 'Start driver OFFLINE' which is unchecked.
- Timeout:** A text box contains '1000' followed by 'ms'.
- Communication check time:** A text box contains '5000' followed by 'ms'.
- Connection management:** A sub-section containing:
 - A dropdown menu set to 'Automatic (managed by the driver)'.
 - A checked checkbox 'Retry failed connection every' followed by a text box '20' and 'seconds'.
 - An unchecked checkbox 'Give up after' followed by a text box '1' and 'failed retries'.
 - An unchecked checkbox 'Disconnect if non-responsive for' followed by a text box '0' and 'seconds'.
- Logging Options:** A sub-section containing:
 - An unchecked checkbox 'Log to File:' followed by a text box containing the path 'C:\eeLogs\MicrolokII_%DATE%.log'.
 - A text box 'File size limit (MB):' followed by a text box '0' and the note '(0 is unlimited)'.

Aba Setup

Opções gerais da aba Setup

| OPÇÃO | DESCRIÇÃO |
|-----------------------|---|
| Physical Layer | Selecione a interface física em uma lista. As opções disponíveis são Serial , Ethernet , Modem e RAS . A interface selecionada deve ser configurada na 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 |

| OPÇÃO | DESCRIÇÃO |
|---------------------------------|--|
| | aguarda para a recepção de um byte qualquer do <i>buffer</i> de recepção |
| Communication check time | Configure o tempo, em milissegundos, para definir o intervalo em que a comunicação é considerada em estado inativo. Enquanto um Driver de Comunicação receber dados válidos, o estado de comunicação é considerado ativo. Porém, se durante o funcionamento um Driver de Comunicação não receber dados válidos neste período de tempo, o estado é considerado inativo. O estado de comunicação é mostrado no Tag IO.CommunicationStatus |
| Start driver OFFLINE | Selecione esta opção para que um Driver inicie em modo Offline ou parado. Isto significa que a interface de I/O não é criada até que se configure um Driver em modo Online utilizando-se um Tag em uma 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 um Driver gerencie a conexão automaticamente, como especificado nas opções seguintes. Selecionar a opção Manual permite que uma aplicação gerencie a conexão completamente |
| Retry failed connection every ... seconds | Selecione esta opção para habilitar a retentativa de conexão de um Driver em um determinado intervalo, em segundos. Se a opção Give up after failed retries não está selecionada, este 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 é atingido, um Driver vai para o modo Offline , assumindo que um problema de hardware foi detectado. Se um 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 um 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. Caso se utilize a macro %PROCESS% no nome do arquivo de log, esta é substituída pelo identificador do processo atual. Esta opção é particularmente útil ao se utilizar várias instâncias de um mesmo Driver no Elipse E3, permitindo assim que cada instância gere um arquivo separado de log. Por exemplo, ao configurar esta opção com o valor "c:\e3logs\drivers\sim_%PROCESS%.log", gera-se um arquivo c:\e3logs\drivers\sim_00000FDA.log para o processo 0FDAh. 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 com o valor "c:\e3logs\drivers\sim_%DATE%.log", gera-se o arquivo c:\e3logs\drivers\sim_2005_12_31.log em 31/12/2005 e o arquivo c:\e3logs\drivers\sim_2006_01_01.log em 01/01/2006. De forma semelhante, a macro %DATE_HOUR% gera um arquivo de log por hora, no formato aaaa_mm_dd_hh</p> |
| File size limit (MB) | <p>Configure o limite de tamanho do arquivo de log, em megabytes. Um valor igual a 0 (zero) significa que não há limite de tamanho para o arquivo de log</p> |

Aba Ethernet

Utilize esta aba para configurar os parâmetros da Interface **Ethernet**. Estes parâmetros, exceto as configurações de 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 o valor TCP/IP para um <i>socket</i> TCP (<i>stream</i>) ou selecione o valor 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, um Driver se conecta ao endereço e porta especificados no grupo 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 pelo endereço IP, que um Driver utiliza para efetuar e receber conexões, ou selecione o valor (All Interfaces) para permitir conexões em qualquer interface de rede |
| Use IPv6 | Selecione esta opção para forçar um 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 enviados, que pode ser retornada antes da mensagem de resposta |
| IP Filter | Lista de endereços IP restringidos ou permitidos de onde um 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, para verificar se um dispositivo pode ser encontrado na rede, em um dispositivo antes de tentar uma conexão |

| OPÇÃO | DESCRIÇÃO |
|-------|--|
| | <p>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 um 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 de um comando ping. Deve-se usar um 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 1 (um) e 4 (quatro) segundos • Retries: Número de retentativas de um 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 | Digite o endereço IP de um dispositivo remoto. Pode-se usar tanto o endereço IP separado por pontos quanto uma URL. No caso de uma URL, um Driver usa o serviço de DNS disponível para mapear a URL para um endereço IP, como por exemplo "192.168.0.13" ou "Server1" |
| Port | Digite a porta IP de um dispositivo remoto, entre 0 (zero) e 65535 |
| Local port | Selecione esta opção para utilizar uma porta IP local fixa ao conectar a um dispositivo remoto |
| Backup IP 1, 2 e 3 | Indique o endereço IP, a porta IP e a porta IP local fixa de até 3 (três) endereços de <i>backup</i> de um dispositivo remoto |

Configurações Gerais

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e das **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.CommunicationStatus

| | |
|--------------------------------|------------------------|
| Tipo de Tag | Tag de Comunicação |
| Tipo de Acesso | Leitura |
| Parâmetro N1 | -1 (menos um) |
| Parâmetro N2 | 0 (zero) |
| Parâmetro N3 | 0 (zero) |
| Parâmetro N4 | 6 (seis) |
| Configuração por String | IO.CommunicationStatus |

Este Tag informa o estado da comunicação de um Driver. Indica o funcionamento da comunicação em função do recebimento de dados válidos dentro de um período de tempo arbitrado na configuração. Para mais informações, consulte o tópico **Aba Setup**. Os valores possíveis são **0 - Comunicação inativa**: O Driver não recebeu dados válidos ou deixou de receber dados depois de n milissegundos, conforme configurado na janela de propriedades, ou **1 - Comunicação ativa**: O Driver está recebendo dados válidos.

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** de um Bloco representa o momento em que um evento ocorre. 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 de um evento. Os valores possíveis são **0**: Driver (específico de um 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 de um evento, uma **String** específica de cada evento

NOTA

Um 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 valores possíveis são os seguintes:

- **0:** Camada física parada, ou seja, um 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, um Driver está em modo **Online**, mas a camada física não está conectada. Se a opção **Connection management** está configurada com o valor **Automatic**, a camada física pode estar conectando, desconectando ou esperando por uma tentativa de reconexão. Se a opção **Connection management** está configurada com o valor **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 um 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 de um Driver em tempo de execução.

Este Tag funciona somente enquanto um Driver está em modo **Offline**. Para iniciar um Driver em modo **Offline**, selecione a opção **Start driver OFFLINE** na caixa de diálogo de configuração deste 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, um Bloco inteiro precisa ser escrito de uma vez só.

No **Elipse SCADA** é necessário usar um Tag Bloco. Cada parâmetro a ser configurado utiliza dois Elementos de Bloco. Por exemplo, caso seja necessário configurar 3 (três) parâmetros, então o tamanho do Bloco deve ser 6 (seis, 3×2). O primeiro Elemento é o nome da propriedade, como uma **String**, e o segundo Elemento é o valor desta 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 **Elipse 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** de um Driver para enviar os parâmetros diretamente para este 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
```

Um Driver não valida nomes de parâmetros ou valores passados, por isto tenha cuidado ao escrever parâmetros e valores. O método **Write** falha se o *array* de configuração é criado incorretamente. Pode-se consultar o log de um Driver ou usar o parâmetro *writeStatus* do método **WriteEx** para descobrir a causa exata de um 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 de um 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 de um Driver através do Tag **IO.SetConfigurationParameters**
- **1 - Driver Online:** A camada física está aberta ou em execução. Enquanto está em modo **Online**, a camada física pode ser conectada ou desconectada e o estado atual pode ser conferido no Tag **IO.PhysicalLayerStatus**

No exemplo a seguir, utilizando o **Elipse E3**, um 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 ao configurar um Driver em modo **Online**, ou seja, escrevendo o valor 1 (um). Neste caso, este 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**
- Este Driver pode ter ficado sem memória
- A camada física pode ter deixado de criar a *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 de um 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 um Driver gerencia a conexão ou **1**: Modo manual, em que uma aplicação gerencia a conexão.

IO.GiveUpEnable

■ Quando configurada para Verdadeiro, define um número máximo de tentativas de reconexão. Se todas as reconexões falharem, um Driver entra em modo **Offline**. Se configurada para Falso, um 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), um Driver tenta apenas uma reconexão quando a conexão é perdida. Se esta falhar, este 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 está 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 está inativa por este período de tempo, então é desconectada.

IO.RecoverEnable

☑ Configure em Verdadeiro para habilitar um Driver a recuperar conexões perdidas e em Falso para deixar um 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 um Driver em modo **Offline** e em Falso para iniciar um Driver em modo **Online**.

NOTA

Não faz sentido modificar esta propriedade em tempo de execução, já que esta só pode ser modificada quando um Driver já está em modo **Offline**. Para configurar um 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 por um Driver. Os valores possíveis são os seguintes:

- **N ou None:** Não utiliza uma interface física, ou seja, um 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*). Um Driver conecta-se a um 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 das **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 um Driver está conectado na conexão atual ou 0 (zero) se um 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 um Driver está desconectado desde o término da última conexão ou 0 (zero) se um 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 um 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 um 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 um 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 um 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 um 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 das **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** está selecionada.

IMPORTANTE

Estes Tags estão disponíveis **SOMENTE** enquanto um Driver está 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 um Driver está atualmente conectado. Se um 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 um Driver está desconectado, este Tag configura o endereço IP ativo para a próxima tentativa de conexão.

IO.Ethernet.SocketState

| | |
|--------------------------------|-------------------------|
| Tipo de Tag | Tag de Comunicação |
| Tipo de Acesso | Somente Leitura |
| Parâmetro N1 | -1 (menos um) |
| Parâmetro N2 | 0 (zero) |
| Parâmetro N3 | 4 (quatro) |
| Parâmetro N4 | 2 (dois) |
| Configuração por String | IO.Ethernet.SocketState |

A propriedade **Value** deste Tag corresponde a estados do *socket* em um mapa de bits:

- **Bit 0**: 0 (zero, não está em escuta) ou 1 (um, em escuta)

- **Bit 1:** 0 (zero, desconectado) ou 1 (um, conectado)

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 um Driver não deve aceitar conexões externas, ou seja, se um Driver se comporta como mestre, ou configure em Verdadeiro para habilitar a recepção de conexões, ou seja, se um 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, um Driver tenta utilizar um endereço IP alternativo ou de *backup*. Configure em Falso para desabilitar a utilização.

IO.Ethernet.BackupIP[2,3]

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

IO.Ethernet.BackupLocalPort[2,3]

📌 Número da porta local a ser utilizada na conexão ao endereço IP alternativo ou de *backup* de um 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]

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

IO.Ethernet.IPFilter

📌 Lista de endereços IPv4 ou IPv6 separados por vírgula, que define de quais endereços um 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, conforme os exemplos a seguir:

- **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:*:***): Aceita conexões de endereços IPv6 no intervalo entre `fe80:03bf:0877:0000:0000:0000:0000:0000` e `fe80:03bf:0877:0000:0000:0000: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 um 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 corresponde 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 aparece 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*, em que um 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 um 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 por um Driver para escutar conexões.

IO.Ethernet.MainIP

A Endereço IP de um equipamento remoto. Pode-se utilizar tanto o endereço numérico como o nome de *host* de um 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 de um 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 em um 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 de um 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.SupressEcho

☑ 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 um Driver enviou para um equipamento.

IO.Ethernet.Transport

A 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).

Histórico de Revisões do Driver

| VERSÃO | DATA | AUTOR | COMENTÁRIOS |
|--------|------------|-----------|--|
| 1.0.2 | 12/06/2025 | M. Ludwig | <ul style="list-style-type: none"> Driver atualizado para a biblioteca IOKit versão 3.0 e Visual Studio 2022 (<i>Case 37942</i>). |
| 1.0.1 | 05/06/2020 | M. Ludwig | <ul style="list-style-type: none"> Primeira versão 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)