

Driver SIEMENS MProt (MPI/PPI/ISO-TCP)

Nome do Arquivo	MProt.dll
Fabricante	Siemens, Vipa
Equipamentos	Modelos de CLPs S7-200, S7-300, S7-400 e S7-1200 da Siemens; Speed7 da Vipa e demais equipamentos compatíveis com algum protocolo do Driver
Protocolo	PPI e MPI (Serial); MPI encapsulado em Ethernet e ISO sobre TCP (RFC1006 ou S7-TCP/IP em interface Ethernet)
Versão	4.0.19
Última Atualização	05/08/2019
Plataforma	Win32
Dependências	IOMKit v2.00
Leitura com Superblocos	Sim
Nível	0

Introdução

O Driver Siemens multiprotocolo (M-Prot) comunica com os CLPs S7-200, S7-300, S7-400 e S7-1200 da Siemens e Speed7 da VIPA usando os protocolos Siemens PPI, MPI, ISOTCP e MPI encapsulados em Ethernet (IBHLink).

O protocolo **PPI** deve ser usado somente para a linha S7-200 usando o cabo conversor RS232-PPI/MPI fornecido pela Siemens.

O protocolo **MPI** pode ser usado para as linhas S7-300 e S7-400 através de cabo conversor RS232-PPI/MPI fornecido pela Siemens, ou também para a linha Speed7 da VIPA na porta MPI usando um cabo RS232 comum.

O protocolo **ISOTCP** (que também pode ser chamado por ISO sobre TCP, RFC1006 ou S7-TCP/IP em diversos materiais dos fabricantes de hardware) pode ser usado para os modelos S7-300 ou S7-400 da Siemens através do uso de um cartão Ethernet CP-3XX, CP-433 ou CP-443; para o modelo S7-1200, e também para a linha Speed7 da VIPA, diretamente na porta Ethernet da CPU. Para o modelo S7-200, há uma variação especial do protocolo **ISOTCP** para ser usado em conjunto com a interface CP-243. Este protocolo é denominado **ISOTCP243**.

Para os CLPs que não possuem porta Ethernet, uma alternativa pode ser o uso de um conversor Ethernet/MPI IBHLink, fornecido pelas empresas IBH Softec ou Hilscher, que atua no nível FDL. A vantagem deste conversor é que a velocidade nominal é mais alta, chegando a 187 kbps na rede MPI, ao passo que pelo conversor serial é de 38,4 kbps. O uso deste conversor é uma alternativa ao uso das placas CP5611 ou similares.

Outra alternativa semelhante é o cabo conversor NETLink PRO Eth da Softing, que converte de **ISOTCP** para **MPI**.

Este Driver não suporta o uso de adaptadores Siemens PPI/MPI por interface USB.

Este Driver não suporta o uso de interfaces CP5611 ou similares para o acesso à rede MPI. Para comunicar com estas placas deve ser usado o Driver S7Functions ou ainda o SIMATIC.NET da Siemens, através do servidor OPC já incluso.

NOTA

M-Prot é um nome criado pela Elipse Software com a finalidade de especificar um Driver que suporta múltiplos protocolos. Não existe relação alguma com nomes de equipamentos, protocolos ou padrões definidos pelos fabricantes supracitados.

Parâmetros de Configuração do Driver

Os parâmetros [P] de configuração do Driver não são utilizados. Todas as configurações são executadas na janela de configurações do Driver, mostrada na figura a seguir.

Aba MProt

As opções disponíveis para o grupo **General** estão descritas na tabela a seguir.

Opções disponíveis no grupo General

OPÇÃO	DESCRIÇÃO
Default Slave Address	Pode ser utilizado como o endereço padrão para qualquer Tag. Deixe o parâmetro <i>N1</i> em 0 (zero) para que seja substituído pelo endereço padrão
Network	Seleção do protocolo: PPI, MPI, ISOTCP, ISOTCP para CPU 243 ou MPI para conversor IBHLink . Os grupos PPI, MPI ou ISOTCP/ISOTCP243 nesta aba são habilitados ou desabilitados conforme a seleção do protocolo
Local Address	Endereço do Driver na rede. Pode ser selecionado arbitrariamente

As opções disponíveis para o grupo **PPI** estão descritas na tabela a seguir.

PPI

☐ PPI Multi Master Operation delay (ms):

Application Timeout (ms): ☒ only for write

Grupo PPI**Opções disponíveis no grupo PPI**

OPÇÃO	DESCRIÇÃO
PPI Multi Master	Informa ao Driver que existem outros Mestres na rede
Application Timeout (ms)	Tempo máximo de comunicação para cada Tag, em milissegundos. Disponível apenas quando é multimestre
Operation delay (ms)	Tempo de parada para o intervalo entre operações de comunicação, em milissegundos. Selecione a opção only for write para indicar a aplicação do intervalo apenas para operações de escrita (veja a nota a seguir)

NOTA

A opção **Operation delay (ms)** adiciona um tempo mínimo de espera que deve haver entre o fim de uma operação de leitura ou escrita e o início de outra operação. Utilize um valor diferente de 0 (zero) nesta configuração apenas se estiver enfrentando falhas de comunicação ocasionadas pela inércia de processamento do CLP. As operações de escrita são as mais prejudicadas, pois normalmente são aleatórias, e para estas operações existe a opção **only for write**. Se esta opção não está selecionada, a espera aplica-se às operações de leitura e de escrita. Se está selecionada, aplica-se apenas às operações de escrita (recomendado). Note que o acréscimo de um tempo de espera pode diminuir o desempenho da aplicação.

As opções disponíveis para o grupo **MPI** estão descritas na tabela a seguir.

MPI

Highest Station Address:

Profibus Speed:

Grupo MPI**Opções disponíveis no grupo MPI**

OPÇÃO	DESCRIÇÃO
Highest Station Address	Indica o maior endereço disponível na rede, para que nos modos PPI e MPI o Driver encontre outros possíveis Mestres na rede. Devem ser inseridas apenas as opções 15, 31 ou 63
Profibus Speed	Velocidade nominal da rede Profibus

As opções disponíveis para o grupo **ISOTCP / ISOTCP243** estão descritas na tabela a seguir.

ISOTCP / ISOTCP243

Extra Connections: Max Simult Req: Source TSAP (hex): Connection type:

Watchdog period (ms): ☐ Use Dest. TSAP ☒ Use default Source Ref. ☐ Use default TSAPs

Connect to:

Main Rack: Slot: Dest. TSAP (hex):

☐ Backup 1 Rack: Slot: Dest. TSAP (hex):

☐ Backup 2 Rack: Slot: Dest. TSAP (hex):

☐ Backup 3 Rack: Slot: Dest. TSAP (hex):

Grupo ISOTCP / ISOTCP243

Opções disponíveis no grupo ISOTCP / ISOTCP243

OPÇÃO	DESCRIÇÃO
Extra Connections	Número de conexões TCP adicionais que podem ser criadas para melhorar o desempenho da comunicação
Max Simult Req	Número máximo de variáveis simultâneas em um único pedido. Esta configuração pode ser usada para melhorar significativamente o desempenho da comunicação, desde que se utilize um valor maior que 0 (zero) na opção Extra Connections
Source TSAP (hex)	Número formado por um Word em hexadecimal que identifica o TSAP local no protocolo
Connection type	Tipo da conexão: PG , OP ou PC . Deve ser selecionada de acordo com o configurado na CPU
Watchdog period (ms)	Intervalo em milissegundos para expiração do Tag de Watchdog . Se este Tag existir e seu valor não variar neste intervalo, o Driver realiza a troca de endereço IP
Use Dest. TSAP	Se esta opção está selecionada, habilita a entrada de valores de Dest. TSAP . Se esta opção não está selecionada, habilita a entrada de valores de Rack , Slot e Connection type
Use default TSAPs	Se esta opção está selecionada, utiliza valores TSAP padrão e assim desabilita a entrada de valores de configuração de Source TSAP , Dest. TSAP , Connection Type , Rack e Slot
Source Ref. (hex)	Número formado por um Word em hexadecimal que identifica a referência local (<i>source reference</i>) no protocolo. Somente é liberada quando a opção Use default Source Ref não está selecionada
Rack	Número do <i>rack</i> da CPU principal de destino
Slot	Número do <i>slot</i> da CPU principal de destino
Dest. TSAP (hex)	Número formado por um Word em hexadecimal que identifica o TSAP de destino no protocolo

OPÇÃO	DESCRIÇÃO
Backup 1 / Backup 2 / Backup 3	Habilita a entrada de valores de <i>rack</i> , <i>slot</i> ou TSAP de destino da(s) respectivas CPU(s) de <i>backup</i> , para uso em sistemas com redundância que sejam de valores diferentes da CPU principal

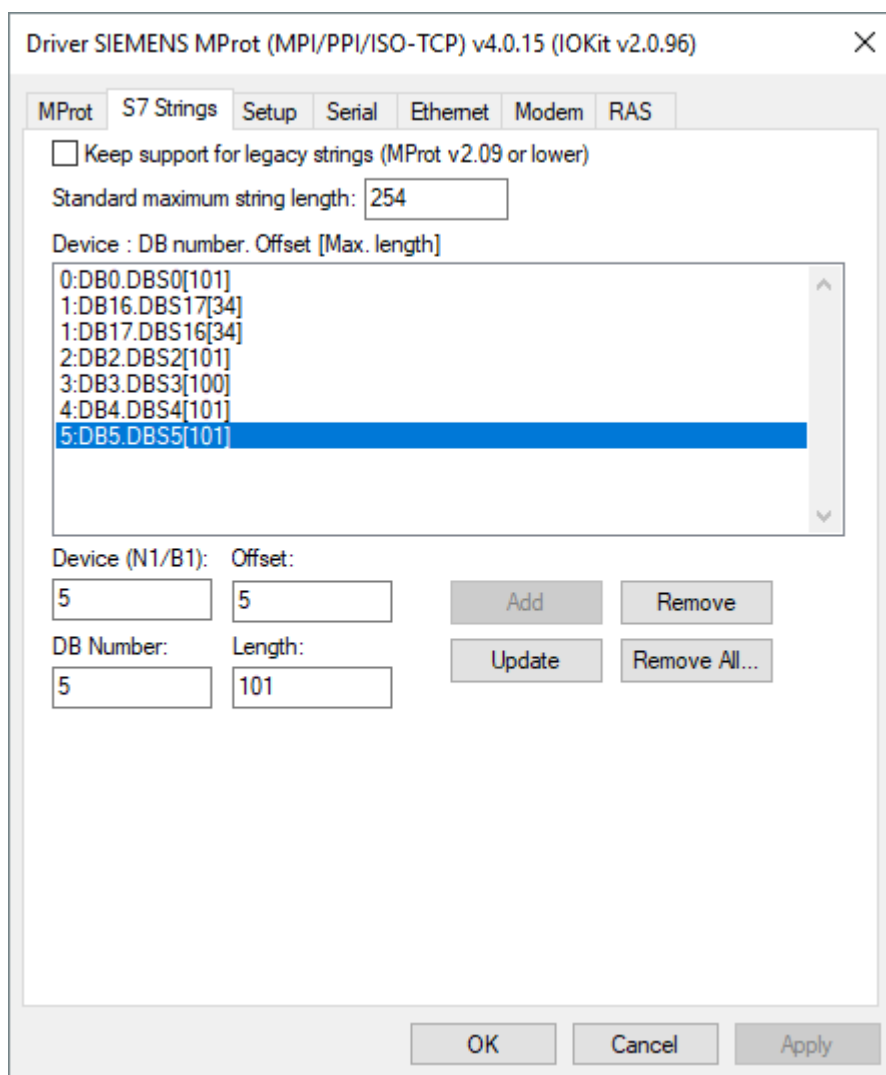
Para que a comunicação deste Driver funcione com o modelo de CLP Siemens S7-1200, é necessário selecionar a opção **ISOTCP**, desmarcar a opção **Use default TSAPs**, configurar a propriedade **Source TSAP (hex)** para o valor "0100" e definir as opções **Connection type** como **PG, Rack** com o valor 0 (zero) e **Slot** com o valor 1 (um).

NOTAS

- Ao selecionar os protocolos **ISOTCP** ou **ISOTCP243**, todos os Tags que estiverem no objeto Driver devem ter o parâmetro *NI* (ou *BI*) em 0 (zero) e o parâmetro **Default Slave Address** também em 0 (zero).
- Os parâmetros **Source Ref** e **Source TSAP** devem ser considerados apenas em casos muito particulares. Devido à execuções bem sucedidas em uma enorme gama de topologias, recomenda-se que se mantenha a opção **Use default Source Ref** sempre selecionada e o valor de **Source TSAP** sempre em "0100".
- Quando a opção **Use Default TSAPs** está selecionada em conjunto com o protocolo **ISOTCP**, o valor de **Source TSAP** é "0100" e o valor de **Dest. TSAP** utilizado é "0202".
- TSAP vem do inglês *Transport Service Access Point*, e é uma terminologia utilizada no protocolo ISO.
- Quando se utiliza adaptadores seriais PC - PPI/MPI, tem sido bastante usual a necessidade de configurar o *handshaking* na aba **Serial** da janela de configurações do Driver. Apenas o controle de **RTS** deve ser configurado em **ON**. Havendo insucesso de comunicação nos testes iniciais com este Driver, convém experimentar esta mudança (**RTS Control** configurado como **ON**) e refazer o teste.

Parâmetros de Configuração de Strings

Esta aba é útil caso haja necessidade de declarar **Strings** com tamanho máximo definido, individualmente ou genericamente.



Aba S7 Strings

As opções disponíveis na aba **S7 Strings** estão descritas na tabela a seguir.

Opções disponíveis na aba S7 Strings

OPÇÃO	DESCRIÇÃO
Keep support for legacy strings (MProt v2.09 or lower)	<p>Mantém o suporte para Strings antigas, anteriores à versão 2.10. A opção selecionada mantém o formato antigo de String implementado em versões anteriores, evitando transtornos em uma migração de versão do Driver.</p> <p>Recomenda-se selecionar esta opção apenas em caso de migração de um projeto cuja versão do Driver seja 2.09 ou anterior. Caso o projeto utilize Strings após uma migração de versão, os Tags de tipo String retornam erros de leitura do CLP.</p> <p>O formato de Strings legado possui uma reserva de 32 bytes de espaço a partir do <i>offset</i> configurado.</p> <p>Caso esteja trabalhando com um projeto novo, deixe esta opção desmarcada</p>
Standard maximum string length	<p>Tamanho máximo padrão das Strings. Preencha um valor padrão configurado na memória do CLP para Strings sem tamanho máximo declarado. Por exemplo, nos CLPs S7-200 este valor é igual a "254". Isto significa que os pedidos por Strings com tamanhos não declarados contêm e indicam uma extensão fixa de 254 caracteres</p>

Listagem de Tamanhos Máximos de Strings

Esta aba também mostra uma lista selecionável com **Strings** declaradas com tamanhos já determinados. A lista aparece em branco caso não existam **Strings** configuradas. As **Strings** podem ser declaradas na memória do CLP de duas formas:

- Sem especificar o tamanho máximo na declaração. Exemplo:

```
STRING var;
```

A **String** é alocada automaticamente com a extensão máxima padrão do CLP.

- Especificando o tamanho máximo na declaração. Exemplo:

```
STRING var[50];
```

No exemplo anterior, a **String** é alocada com tamanho máximo de "50". Devido a esta segunda forma é que a listagem de tamanho de **Strings** se mostra importante.

Para determinar o tamanho de uma nova **String** declarada, é preciso preencher corretamente todos os campos, conforme descrito na tabela a seguir.

Opções disponíveis para a configuração do tamanho máximo de Strings

OPÇÃO	DESCRIÇÃO
Device	Endereço do CLP. Preencha com o mesmo valor do parâmetro <i>N1/B1</i> do Tag. Para isto, veja o tópico Endereçamento Padrão
DB Number	Informe o valor do número do DB onde a String está localizada
Offset	Informe o valor do <i>offset</i> no DB onde a String está localizada

OPÇÃO	DESCRIÇÃO
Length	Preencha com o valor máximo de tamanho da String , conforme declarado na programação do CLP

Caso já exista alguma **String** declarada na lista com mesmo valor de **Device**, **DB Number** e **Offset**, esta é automaticamente apontada na tabela e seus valores são carregados em todos os campos de edição. Três opções destinam-se à manipulação dos dados das **Strings** da lista:

- **Add**: Adiciona novos parâmetros
- **Update**: Altera parâmetros já listados
- **Remove**: Remove totalmente uma linha de parâmetros

Clique em **OK** para confirmar todas as configurações listadas e fechar a janela. Clique em **Remove All** para remover todos os dados da lista.

NOTA

Ao declarar Tags com parâmetros de Endereçamento Simbólico, não há necessidade de preencher a lista com declarações de **Strings**. O tamanho pode ser especificado no próprio parâmetro de símbolo disponível no Tag.

Referência de Tags

Esta seção contém informações de configuração de Tags por **Endereçamento Simbólico** e por **Endereçamento Padrão** (parâmetros *N/B*). Também contém referências aos **Tags da Interface em Conexões ISOTCP Extras**.

Configuração por Parâmetros Sintáticos

Utilize a sintaxe a seguir para cada campo presente no **E3** ou **Elipse Power**:

- **Dispositivo**: Insira o endereço do equipamento na rede. Se igual a 0 (zero) e protocolo diferente de **ISOTCP** ou **ISOTCP243**, é substituído pelo **Default Slave Address**. Se o protocolo é **ISOTCP** ou **ISOTCP243**, este valor deve ser deixado em 0 (zero). O campo **Dispositivo** também pode permanecer vazio, desde que seja inserido no campo **Item** antes do símbolo de dois pontos
- **Item**: Este campo deve obedecer alguma das sintaxes a seguir

Use a sintaxe geral a seguir se a área não é igual a **DB**. Valores entre colchetes são opcionais:

```
<[Dispositivo:]><Área><[Tipo]><Endereço>[.Bit]
```

Onde:

- **Dispositivo**: Endereço do CLP conforme exposto no item **Dispositivo**, caso não tenha sido informado naquele campo
- **Área**: Área de dados dentro do CLP. As seguintes opções podem ser utilizadas:
 - **S**
 - **SM**
 - **AI** (*Analog Input*)
 - **AQ** (*Analog Output*)
 - **C** (*Counter*)
 - **T** (*Timer*)
 - **I** (*Digital Input*)
 - **Q** (*Digital Output*)
 - **M** (*Memory*)
 - **HC** (*High Speed Counter*)
- **Tipo**: Tipo de dados a ler. A tabela a seguir mostra os possíveis símbolos para os tipos de dados

Opções disponíveis para tipos de dados

TIPO DE DADOS	SIGNIFICADO
X	Usado para extrair um bit de um byte
B	Byte
W	Word
D	DWord
I	Int
LI	DInt
F	Float
S	String

TIPO DE DADOS	SIGNIFICADO
S5T	S5Time

- **Endereço:** Endereço numérico a ler
- **Bit:** Opcional que informa o bit de uma palavra a ler ou a escrever (entre zero e 31)

Exemplo:

```
(CLP 4, bit 1 da memória do endereço 10)
Device: Vazio - Item 4:M10.1
```

Se a área de dados é igual a **DB** (também conhecida como **V**), use a sintaxe a seguir. Valores entre colchetes são opcionais:

```
<[Dispositivo:]>DB<NumeroDB>:<Tipo><Endereço><[.Bit]>
```

- **Dispositivo:** Refere-se ao mesmo item opcional da sintaxe geral
- **NumeroDB:** Coloque o número do DB. Caso a memória contenha um bloco DB único ou não especificado, preencha com o valor 1 (um)
- **Endereço:** Endereço numérico (*offset*) a ler
- **Bit:** Valor opcional que informa o bit do tipo de dados a ler ou a escrever (entre zero e 31)

Opções disponíveis para tipos de dados de áreas de dados DB

TIPO DE DADOS	SIGNIFICADO
DBX	Usado para extrair um bit de um byte em um DB
DBB	Usado para ler ou escrever um byte em um DB
DBW ou DW	Usado para ler ou escrever um Word em um DB
DBD ou DD	Usado para ler ou escrever um Double Word em um DB
DBI ou DI	Usado para ler ou escrever um Int em um DB
DBLI ou DLI	Usado para ler ou escrever um DInt em um DB
DBF ou DF	Usado para ler ou escrever um Floating Point (real de 32 bits) em um DB
DBS ou DS	Usado para acessar uma String em um DB
DBS5T	Usado para acessar um <i>timer</i> do tipo de dados S5Time em um DB

Exemplos:

```
(CLP 2, Word começando no endereço 20 do DB1)
Device: 2 - Item: DB1:DW20
(Mesmo do anterior, porém Device foi informado no campo Item)
Device: Vazio - Item: 2:DB1:DW20
(CLP 7, DB 5, bit 2 do byte 7)
Device: Vazio - Item: 7:DB5:DBX7.2
```

A sintaxe para tipos de dados **String** em áreas de dados **DB** é a seguinte:

```
<[Dispositivo:]>DB<NumeroDB>:DBS<Endereço><[Tamanho máximo]>
```

Onde:

- **Device, NumeroDB e Endereço:** Referem-se aos mesmos itens da sintaxe geral
- **Tamanho máximo:** Opcional que informa o tamanho máximo declarado na **String**. Se não é informado, é considerado o tamanho máximo padrão da **String** como preenchido na janela de configuração de **Strings**

Exemplos de sintaxe para **Strings**:

```
(CLP 2, String começando no endereço 16 do DB17,
usando o tamanho padrão máximo do CLP)
Device: 2 - Item: DB17:DBS16
(mesmo do anterior, porém Device foi informado no campo Item
e com tamanho máximo alocado de 25 caracteres)
Device: Vazio - Item: 2:DB17:DBS16[25]
(CLP 4, String começando no endereço 100 do DB10,
com tamanho máximo alocado de 50 caracteres)
Device: Vazio - Item 4:DB10:DS100[50]
```

Configuração por Parâmetros Numéricos (N/B)

Use a sintaxe padrão descrita na tabela a seguir para todos os Tags e Blocos.

Sintaxe padrão para Tags e Blocos

PARÂMETRO	DESCRIÇÃO
N1/B1	Endereço do CLP. Se igual a 0 (zero) e protocolo diferente de ISOTCP ou ISOTCP243 , é substituído pelo Default Slave Address . Se o protocolo é ISOTCP ou ISOTCP243 , este valor deve ser deixado em 0 (zero)
N2/B2	Tipo de dados e área de dados (veja as tabelas a seguir). O valor deve ser composto pelo tipo de dados multiplicado por 100 mais a área de dados (a fórmula é $N2/B2 = TipoDados \times 100 + Área$)
N3/B3	Se a área de dados selecionada é V (DB) , preencha com o número do bloco DB. Caso contrário, deixe em 0 (zero). Caso a memória contenha um bloco DB único ou não especificado, preencha com o valor 1 (um)
N4/B4	Endereço na área de dados ou <i>offset</i> do bloco DB. Para usar tipos de dados que ocupam mais de um byte, devem ser colocados endereços múltiplos de dois para tipos de dados de dois bytes (16 bits com e sem sinal) e múltiplos de quatro para tipos de dados de quatro bytes (32 bits com e sem sinal e ponto flutuante de 32 bits)

Opções disponíveis para tipos de dados

TIPO DE DADOS	SIGNIFICADO
0	Padrão da área de dados
1	BOOL (Booleano)

TIPO DE DADOS	SIGNIFICADO
2	BYTE (oito bits sem sinal)
3	WORD (16 bits sem sinal)
4	INT (16 bits com sinal)
5	DWORD (32 bits sem sinal)
6	DINT (32 bits com sinal)
7	REAL (32 bits de ponto flutuante - IEEE 754)
8	STRING (ver nota a seguir)
12	S5TIME (tempo em segundos, 32 bits de ponto flutuante - IEEE 754, ver nota a seguir)

Opções disponíveis para áreas de dados

ÁREA DE DADOS	SIGNIFICADO
0	S
1	SM
2	AI (<i>Analog Input</i>)
3	AQ (<i>Analog Output</i>)
4	C (<i>Counter</i>)
5	T (<i>Timer</i>)
6	I (<i>Digital Input</i>)
7	Q (<i>Digital Output</i>)
8	M (<i>Memory</i>)
9	V (<i>DB</i>)
10	HC (<i>High Speed Counter</i>)

NOTAS

- Para tipos de dados **S5Time**, o valor a preencher é sempre em segundos, com ponto flutuante de 32 bits. A gama de valores diferentes de zero está entre 0,01 e 9990,0 segundos. A base de tempo é preenchida ou interpretada automaticamente.
- No protocolo **PPI** há uma limitação no Bloco de Comunicação para dados em bytes. Para leitura, o máximo permitido são 224 bytes, e para escrita são 218 bytes. Isto significa, respectivamente, que para tipos de dados **Word** (16 bits), o Bloco não pode ultrapassar 112 e 109 Elementos. Para tipos de dados **DWord** (32 bits), o Bloco não pode ultrapassar 56 e 54 Elementos, e assim por diante.
- Caso desconheça a definição de **Rack** e **Slot** para endereçamento dos Tags em protocolo **ISOTCP**, consulte o artigo KB-39019: *Configurações de Rack e Slot* no **Elipse Knowledgebase**.

Tags da Interface em Conexões ISOTCP Extras

Ao optar pelo uso de conexões ISOTCP extras com o parâmetro **Extra Connections** na **Janela de Configurações do Driver**, estas conexões podem ser controladas e monitoradas por três Tags específicos de Interface: **Physical Layer Status**, **IPSelect** e **IPSwitch**.

NOTA

Estes Tags não podem ser utilizados quando o parâmetro **Extra Connections** é igual a 0 (zero). Neste caso, utiliza-se os Tags do **IOKit** correspondentes, de mesmo nome, cujo uso pode ser consultado no tópico **Documentação das Interfaces de Comunicação**.

Physical Layer Status (MProt)

Somente Leitura

Configuração por parâmetros numéricos

PARÂMETRO	VALOR
N1	-2
N2	0 (zero)
N3	0 (zero)
N4	2

Configuração por parâmetros sintáticos

PARÂMETRO	VALOR
Item	MProt.IO.PhysicalLayerStatus

Este Tag indica o estado da conexão na camada física. Seus possíveis valores são os seguintes:

- **0**: Camada física desconectada
- **1**: Camada física conectada

IPSelect (MProt)

Leitura e Escrita

Configuração por parâmetros numéricos

PARÂMETRO	VALOR
N1	-2
N2	0 (zero)
N3	4
N4	0

Configuração por parâmetros sintáticos

PARÂMETRO	VALOR
Item	MProt.IO.Ethernet.IPSelect

Indica o endereço IP ativo. Seus possíveis valores são os seguintes:

- **0**: O endereço IP principal está selecionado (ativo)
- **1**: O endereço IP alternativo (*backup*) está selecionado (ativo)

Se a interface Ethernet estiver conectada, o Tag indica qual dos dois endereços IP configurados está sendo usado. Se a interface estiver desconectada, o Tag indica qual endereço IP é usado primeiro na próxima tentativa de conexão.

Durante o processo de conexão, se o endereço IP ativo não estiver disponível, o **IOKit** tenta conectar com 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 1 (um) ou 0 (zero) neste Tag. Isto força a reconexão com o endereço IP especificado (**0**: endereço IP principal e **1**: endereço IP de *backup*) se o Driver estiver atualmente conectado. Se o Driver estiver desconectado, isto configura o endereço IP ativo para a próxima tentativa de conexão.

IPSwitch (MProt)

Somente Escrita

Configuração por parâmetros numéricos

PARÂMETRO	VALOR
N1	-2
N2	0 (zero)
N3	4
N4	1

Configuração por parâmetros sintáticos

PARÂMETRO	VALOR
Item	MProt.IO.Ethernet.IPSwitch

Qualquer valor escrito neste Tag força o *switchover* manual. Se o endereço IP principal estiver ativo, então o endereço IP de *backup* é ativado, e vice-versa. Isto força a reconexão com o endereço IP especificado se o Driver estiver atualmente conectado. Se o Driver estiver desconectado, isto configura o endereço IP ativo para a próxima tentativa de conexão.

Tag Watchdog

Esta funcionalidade é útil apenas para arquiteturas de redundância que contam com uma variável no CLP a ser eleita para indicar o momento de troca do endereço IP de destino. Na grande maioria dos casos, este tipo de Tag não é necessário. Se for o caso de aplicar a um projeto, o Tag deve ser uma variável Booleana (**BIT**) que deve variar de valor antes que o tempo configurado expire, e logicamente deve ser lida com uma varredura de tempo razoavelmente inferior ao tempo de expiração (opção **Watchdog period (ms)** do grupo **ISOTCP / ISOTCP243** na aba **MProt** da janela de configuração). Sempre que a variável não variar por este período, a troca de endereço IP é executada. Para configurar um Tag como **Watchdog**, adicione a **String** `"/watchdog"` ao final do parâmetro sintático **Item**, como por exemplo `"DB2:DBX0.0/watchdog"`.

Coleta SOE

Esta seção contém informações específicas para Coleta SOE de eventos.

Preparando a Coleta SOE

Antes de utilizar os Tags de Coleta SOE, é preciso preparar o CLP com a construção de uma Tabela DB (área **V**) e o desenvolvimento de uma lógica programável compatível com os procedimentos de coleta SOE desenvolvida para este Driver.

Tabela de Eventos do SOE

Esta tabela tem como objetivo dimensionar o tamanho do *buffer* de eventos e gerenciar a entrada e a saída destes em uma rotina de *buffer* circular. Esta tabela é constantemente atualizada, tanto pelo CLP quanto pelo Driver Siemens MProt.

A Tabela de Eventos SOE deve conter registros de controle e de armazenamento dos eventos, com base na estrutura de dados descrita na tabela a seguir.

Estrutura de dados

ENDEREÇO	DESCRIÇÃO	TIPO DE DADOS
0.0		STRUCT
+0.0	Status da tabela	WORD (16 bits sem sinal)
+2.0	Ponteiro de gravação	WORD (16 bits sem sinal)
+4.0	Status da aquisição	WORD (16 bits sem sinal)
+6.0	Limite máximo de itens do <i>buffer</i> circular	WORD (16 bits sem sinal)
+8.0	<i>Buffer</i> circular	ARRAY[1..n] (limite de itens definido pelo usuário)
+0.0		STRUCT
+0.0	TIMESTAMP_LOLO (ano)	WORD (16 bits sem sinal)
+2.0	TIMESTAMP_LOHI (dia e mês)	WORD (16 bits sem sinal)
+4.0	TIMESTAMP_HILO (hora e minuto)	WORD (16 bits sem sinal)
+6.0	TIMESTAMP_HIHI (segundo e milissegundo)	WORD (16 bits sem sinal)
+8.0	Valor do tipo de evento 1	Tipo de dados do evento (definido pelo usuário)
+n.0	Valor do tipo de evento 2	Repete o mesmo tipo de dados
+n.0	Valor do tipo de evento 3	Repete o mesmo tipo de dados
+n.0	Valor do tipo de evento n	Repete o mesmo tipo de dados
=n.0		END_STRUCT
=n.0		END_STRUCT

Descrição dos Registros de Controle dos Eventos

- **Status da Tabela:** Deve ser mantido exclusivamente pelo CLP, indicando o número de eventos disponíveis para a leitura no *buffer* circular. Deve ser atualizado pelo CLP sempre que novos eventos são adicionados ao *buffer* circular, ou após a conclusão da coleta de eventos pela aplicação, o que pode ser detectado pela mudança no **Status da Aquisição**

- **Ponteiro de Gravação:** Deve ser mantido exclusivamente pelo CLP, indicando o índice, começando em zero, da posição onde deve ser inserido o próximo evento. O índice deve ser incrementado pelo CLP a cada nova inserção de eventos no *buffer* circular, voltando ao índice zero após alcançar o limite máximo do *buffer* circular
- **Status da Aquisição:** Deve ser mantido pelo CLP em conjunto com o Driver MProt, indicando o número de registros já lidos a cada transação. Após cada coleta, o Driver MProt escreve neste registro o número de eventos que conseguiu ler. Ao detectar esta modificação, o CLP deve imediatamente descontar este valor escrito pelo Driver MProt do **Status da Tabela**, e então zerar o **Status da Aquisição**
- **Limite Máximo de Itens do Buffer Circular:** Valor constante que especifica o limite máximo de eventos a armazenar no *buffer* circular antes do ponteiro voltar para o índice 0 (zero). Deve conter exatamente o valor limite do *Array* dimensionado para os eventos do *buffer* circular

Descrição dos Registros de Armazenagem dos Eventos

- **TIMESTAMP:** Horário da ocorrência do evento
- **Valor do Evento:** Valor do evento ocorrido, que pode ser composto por um ou n valores (todos com o mesmo tipo de dados), no qual são agrupados para o mesmo **TIMESTAMP** gerado na ocorrência de um evento

Formato do TIMESTAMP

O **TIMESTAMP** é representado por quatro **Words**, conforme a estrutura de dados descrita na tabela a seguir.

Estrutura de dados

WORD	CONTEÚDO	INTERVALO
0	Ano	Entre 0 (zero) e 65535
1	Dia e mês	dddddddddmmmmmmmm
2	Hora e minuto	hhhhhhhhmmmmmmmm
3	Segundo e milissegundo	ssssssmmmmmmmmmm

- O primeiro **Word** contém um valor inteiro referente ao ano
- O segundo **Word** está dividido em parte alta para representar o dia e em parte baixa para representar o mês
- O terceiro **Word** está dividido em parte alta para representar as horas e em parte baixa para representar os minutos
- O quarto **Word** usa os seis bits mais altos para representar os segundos e os 10 bits mais baixos para representar os milissegundos

Procedimento de Aquisição

O CLP deve iniciar a inserção dos eventos no sentido crescente, a partir do endereço base da tabela, referente ao início do *buffer* circular. A cada novo evento inserido, o ponteiro de gravação deve ser incrementado, passando a apontar para o próximo endereço disponível do *buffer*.

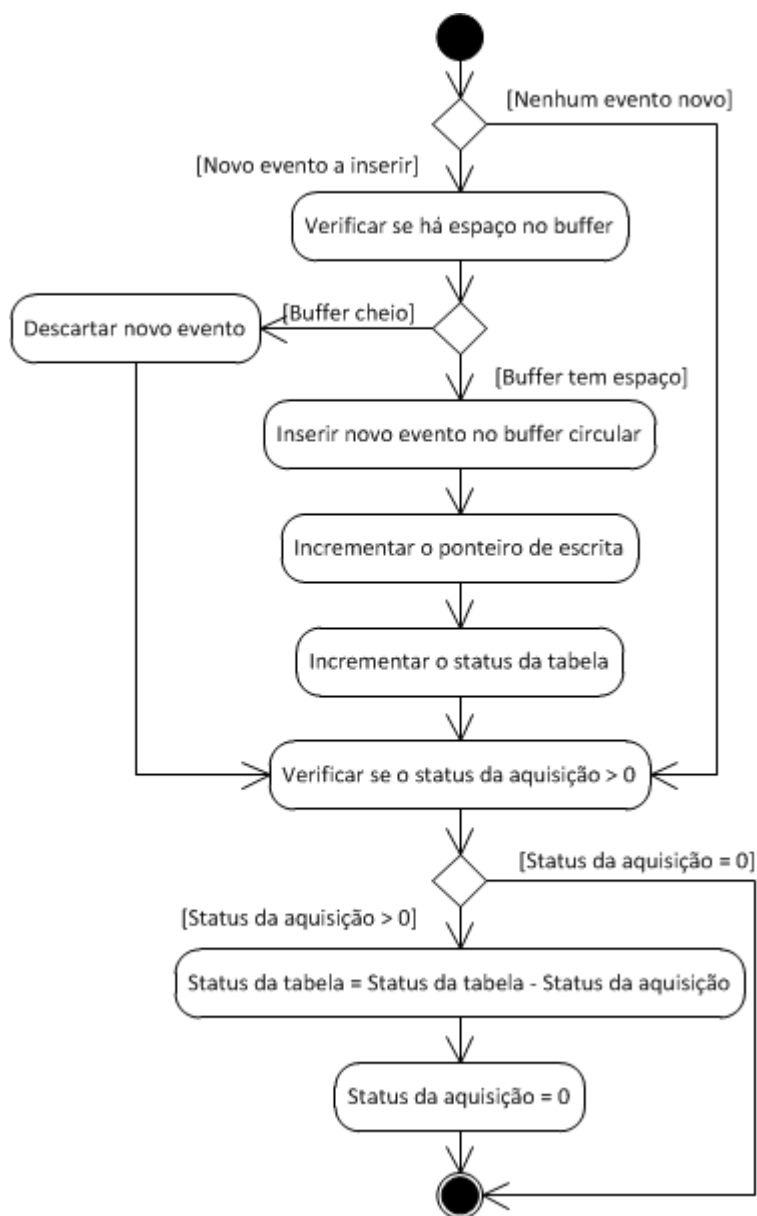
O Driver executa a leitura do evento mais antigo para o mais recente. O endereço do início da leitura é calculado pelo Driver através do valor do **Ponteiro de Gravação** e do **Status da Tabela**.

Se o número de eventos disponíveis é maior que o máximo permitido em um único *frame* de comunicação do protocolo, o Driver executa múltiplas leituras em bloco, atualizando o valor do **Status da Aquisição** no final do processo com o número total de eventos lidos.

Ao detectar que o Driver escreveu um valor maior que 0 (zero) no **Status da Aquisição**, o CLP deve imediatamente subtrair o valor do **Status da Aquisição** do valor do **Status da Tabela** e zerar o **Status da Aquisição**.

O CLP pode inserir novos eventos na tabela durante o processo de aquisição pelo CLP, desde que não ocorra *overflow* do *buffer* circular, incrementando o **Status da Tabela**.

A figura a seguir apresenta um pequeno fluxograma, em formato de Diagrama de Atividades UML, com uma sugestão de implementação para a lógica do CLP.



Fluxograma SOE

Tags de Coleta SOE

A coleta SOE de eventos é executada com a utilização dos Tags descritos a seguir, por meio de uma comunicação ISOTCP com o CLP.

Tag Bloco de Registro de Controle (Somente Leitura)

- **B1:** 0 (zero)
- **B2:** 391 (Tipo de dados = $3 \times 100 + \text{Área de dados} = 91$)
- **B3:** Número do bloco DB. Caso a memória contenha um bloco DB único ou não especificado, preencha com o valor 1 (um)
- **B4:** Não utilizado

O Tag Bloco de consulta dos Registros de Controle deve conter quatro Elementos para retornar os seguintes valores:

- **Elemento 1:** Status da tabela
- **Elemento 2:** Ponteiro de gravação
- **Elemento 3:** Status da aquisição
- **Elemento 4:** Limite máximo de itens do *buffer* circular

Para uma descrição de cada um destes Registros de Controle, consulte o tópico **Preparando a Coleta SOE**.

Tag Bloco de Coleta de Dados (Somente Leitura)

- **B1:** 0 (zero)
- **B2:** Tipo de dados $\times 100 + \text{Área de dados} = 90$
- **B3:** Número do bloco DB. Caso a memória contenha um bloco DB único ou não especificado, preencha com o valor 1 (um)
- **B4:** Não utilizado

O Tag Bloco de Coleta de Dados deve conter um número de Elementos correspondente ao número de valores de tipo de evento n que compõem um único evento. Se o evento é composto por um único valor, dimensione o Tag Bloco de Coleta de Dados com apenas um Elemento. Caso o evento seja composto por dois valores, o Tag Bloco deve ser dimensionado para dois Elementos, e assim por diante. Utilize o parâmetro **B2** do Tag Bloco para indicar o tipo de dados associado aos valores do evento.

NOTA

- Todos os valores que compõem um evento devem ser do mesmo tipo de dados, assim como cada tabela DB do CLP deve ser preenchida por um mesmo tipo de evento.
- As Áreas de Dados **90** e **91** existem para serem utilizadas somente no nível lógico de aplicação do usuário. Em nível de protocolo, ambos os Tags vão trabalhar na Área de Dados **9 (DA_V)** do CLP automaticamente.


Documentação das Interfaces de Comunicação

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

Configurações do Driver

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

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

No **E3** versão 2.0 ou posterior, clique em **Configurar o driver**  na barra de ferramentas do Driver. No **Eclipse 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 do Driver. A aba é dividida em três partes distintas:

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

The screenshot shows the 'Setup' tab of a configuration window. It contains three main sections: 'Physical Layer' with a dropdown set to 'Serial' and a 'Start driver OFFLINE' checkbox; 'Timeout' with a text box set to '1000' and 'ms'; and 'Connection management' with a 'Mode' dropdown set to 'Automatic (managed by the driver)', a checked 'Retry failed connection every' checkbox with a '20' second interval, an unchecked 'Give up after' checkbox with a '1' failed retries interval, and an unchecked 'Disconnect if non-responsive for' checkbox with a '0' second interval. Below these is a 'Logging Options' section with an unchecked 'Log to File' checkbox and an empty text box.

Aba Setup

Opções gerais da aba Setup

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

Opções para o grupo Connection management

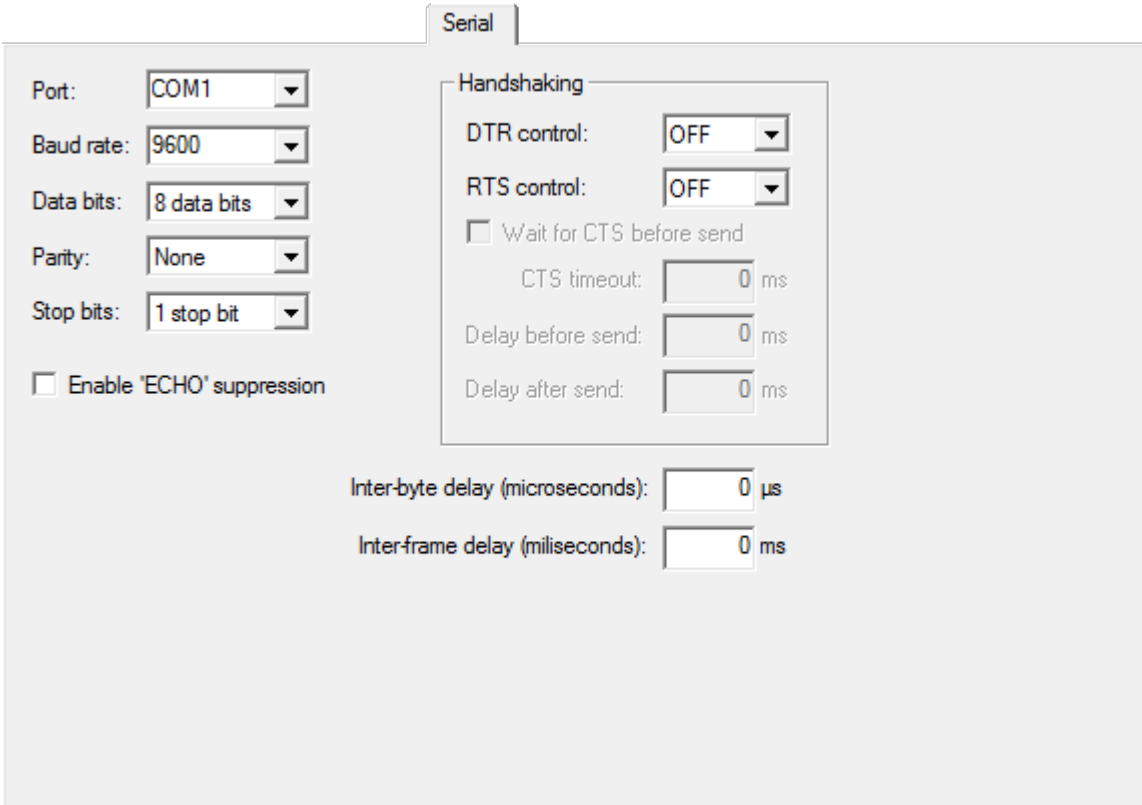
OPÇÃO	DESCRIÇÃO
Mode	Seleciona o modo de gerenciamento de conexão. Selecionar a opção Automatic permite que o Driver gerencie a conexão automaticamente, como especificado nas opções seguintes. Selecionar a opção Manual permite que a aplicação gerencie a conexão completamente. Veja o tópico Estados do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

Opções para o grupo Logging Options

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

Aba Serial

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



Aba Serial

Opções gerais da aba Serial

OPÇÃO	DESCRIÇÃO
Port	Selecione a porta serial a partir da lista (de COM1 até COM4) ou digite o nome da porta serial no formato COMn (por exemplo, "COM15"). Ao digitar o nome da porta manualmente, a caixa de diálogo aceita apenas nomes de portas começando com a expressão "COM".
Baud rate	Selecione o <i>baud rate</i> a partir da lista (1200, 2400, 4800, 9600, 19200, 38400, 57600 ou 115200) ou digite o <i>baud rate</i> desejado (por exemplo, 600).
Data bits	Selecione 7 ou 8 bits de dados a partir da lista.
Parity	Selecione a paridade a partir da lista (None, Even, Odd, Mark ou List).
Stop bits	Selecione o número de stop bits a partir da lista (1, 1.5 ou 2 stop bits).
Enable 'ECHO' suppression	Habilite esta opção para remover o eco recebido após a Interface de Comunicação enviar dados pela porta serial. Se o eco não for igual aos bytes recém enviados, a Interface de Comunicação aborta a comunicação.

OPÇÃO	DESCRIÇÃO
Inter-byte delay (microseconds)	Define uma espera entre cada byte transmitido pela Interface de Comunicação, em milionésimos de segundo (1000000 igual a um segundo). Esta opção deve ser utilizada com esperas pequenas (menos de um milissegundo).
Inter-frame delay (milliseconds)	Define uma espera entre pacotes enviados ou recebidos pela Interface de Comunicação, em milésimos de segundo (1000 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* (controlar quando o dado pode ser enviado ou recebido através da linha serial). Na maioria das vezes, configurar a opção **DTR control** para **ON** e a opção **RTS control** para **Toggle** funciona tanto com linhas seriais RS232 quanto com linhas seriais RS485.

Opções disponíveis no grupo Handshaking

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

Aba Ethernet

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

Aba Ethernet

Opções disponíveis na aba Ethernet

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

OPÇÃO	DESCRIÇÃO
IP Filter	Lista de endereços IP restringidos ou permitidos de onde o Driver aceita conexões (<i>Firewall</i>). Veja a propriedade IO.Ethernet.IPFilter para mais detalhes
Main IP Backup IP 1 Backup IP 2 Backup IP 3	<p>Estas opções permitem configurar até quatro endereços para o dispositivo remoto:</p> <ul style="list-style-type: none"> • IP: Digite o endereço IP do dispositivo remoto. Pode-se usar tanto o endereço IP separado por pontos quanto uma URL. No caso de uma URL, o Driver usa o serviço de DNS disponível para mapear a URL para um endereço IP. Por exemplo, "192.168.0.13" ou "Server1" • Port: Digite a porta IP do dispositivo remoto (de 0 até 65535) • Specify local port: Selecione esta opção para utilizar uma porta local fixa ao conectar ao dispositivo remoto
PING before connecting	<p>Habilite esta opção para executar um comando ping (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):</p> <ul style="list-style-type: none"> • Timeout: Especifique o número de milissegundos de espera por uma resposta do comando ping. Deve-se usar o comando ping para verificar o tempo normal de resposta, configurando esta opção para um valor acima desta média. Normalmente pode-se configurar um valor entre 1000 e 4000 milissegundos (entre um e quatro segundos) • Retries: Número de tentativas do comando ping (não conta a tentativa inicial). Se todas as tentativas falharem, então a conexão com o <i>socket</i> é abortada

Aba Modem

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

The screenshot shows a software window titled 'Modem'. It contains a tab labeled 'Modem'. Below the tab, there is a section 'Select the modem to use:' with a dropdown menu and a button labeled 'Modem settings...'. Below this, there is a 'Dial Number:' text box. At the bottom, there is a checkbox labeled 'Accept incoming calls'.

Aba Modem

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

Opções disponíveis na aba Modem

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

Aba RAS

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

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

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

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

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

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

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

Aba RAS

Opções disponíveis na aba RAS

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

Configurações Gerais

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

Tags de Comunicação

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

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

IO.IOKitEvent

Tipo de Tag	Tag Bloco
Tipo de Acesso	Somente Leitura
Parâmetro B1	-1
Parâmetro B2	0
Parâmetro B3	0
Parâmetro B4	1
Propriedade Size	4
Propriedade ParamItem	IO.IOKitEvent

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

- **Elemento 0:** Tipo de evento
 - **0:** Informação
 - **1:** Advertência
 - **2:** Erro
- **Elemento 1:** Fonte do evento
 - **0:** Driver (específico do Driver)
 - **-1:** IOKit (eventos genéricos da Interface de Comunicação)
 - **-2:** Interface **Serial**

- -3: Interface **Modem**
- -4: Interface **Ethernet**
- -5: Interface **RAS**
- **Elemento 2:** Número do erro (específico de cada fonte de evento)
- **Elemento 3:** Mensagem do evento (**String**, específica de cada evento)

NOTA

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

IO.PhysicalLayerStatus

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1
Parâmetro N2	0
Parâmetro N3	0
Parâmetro N4	2
Configuração por String	IO.PhysicalLayerStatus

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

- **0:** Camada física parada (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 (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 (a camada física está pronta para ser usada). Isto **NÃO** significa que o equipamento esteja conectado, apenas que a camada de acesso está funcionando

IO.SetConfigurationParameters

Tipo de Tag	Tag Bloco
Tipo de Acesso	Somente Leitura
Parâmetro B1	-1
Parâmetro B2	0
Parâmetro B3	0
Parâmetro B4	3
Propriedade Size	2
Propriedade ParamItem	IO.SetConfigurationParameters

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

Este Tag funciona somente enquanto o Driver estiver em modo **Offline**. Para iniciar o Driver em modo **Offline**, selecione a opção **Start driver OFFLINE** na caixa de diálogo de configuração do Driver. Pode-se tanto escrever em um Tag PLC ou em um Tag Bloco contendo os parâmetros a serem modificados (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. Veja este script no **Eclipse SCADA**:

```
// 'Block' deve ser um Tag Block 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 Block 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. Veja estes exemplos:

```
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 do array são ignorados pelo Driver.
arr(1) = Array("IO.Type", "Serial")
arr(2) = Array("IO.Serial.Port", 1)
arr(3) = Array("IO.serial.BaudRate", 19200)
Driver.Write -1, 0, 0, 3, arr
```

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

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

IO.WorkOnline

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Leitura ou Escrita
Parâmetro N1	-1
Parâmetro N2	0
Parâmetro N3	0
Parâmetro N4	4
Configuração por String	IO.WorkOnline

Este Tag informa o estado atual do Driver e permite iniciar ou parar a camada física.

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

No exemplo a seguir (utilizando o **E3**), o Driver é colocado em modo **Offline**, sua 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** (escrevendo o valor 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, falha ao inicializar o TAPI (modem), etc. 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 (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:

- **0**: Modo automático (o Driver gerencia a conexão)
- **1**: Modo manual (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, ela 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 ela 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 é igual 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 (o Driver deve fornecer uma interface personalizada)
- **S ou Serial:** Utiliza uma porta serial local (COMn)
- **M ou Modem:** Utiliza um modem local (interno ou externo) acessado via TAPI (*Telephony Application Programming Interface*)
- **E ou Ethernet:** Utiliza um *socket* TCP/IP ou UDP/IP
- **R ou RAS:** Utiliza uma Interface **RAS** (*Remote Access Server*). O Driver conecta-se ao equipamento RAS através da Interface **Ethernet** e então emite um comando **AT** (*dial*)

Configuração de Estatísticas

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

Tags de Comunicação

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

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

IO.Stats.Partial.BytesRecv

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

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

IO.Stats.Partial.TimeDisconnectedSeconds

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

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

IO.Stats.Total.BytesRecv

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

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

IO.Stats.Total.BytesSent

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

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

IO.Stats.Total.ConnectionCount

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

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

IO.Stats.Total.TimeConnectedSeconds

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

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

IO.Stats.Total.TimeDisconnectedSeconds

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

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

Propriedades

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

Configuração da Interface Ethernet

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

Tags de Comunicação

Tags da Interface Ethernet (N2/B2 = 4)

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

IMPORTANTE

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

IO.Ethernet.IPSelect

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

Indica o endereço IP ativo. Os valores possíveis são os seguintes:

- **0:** O endereço principal de IP está selecionado
- **1:** O endereço alternativo (*backup*) de IP está selecionado
- **2:** O endereço alternativo (*backup*) 2 de IP está selecionado
- **3:** O endereço alternativo (*backup*) 3 de IP está selecionado

Se a Interface **Ethernet** (ou **RAS**) estiver conectada, este Tag indica qual dos quatro endereços IP configurados está em uso. Se a Interface estiver 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 estiver 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 estiver atualmente conectado. Se o Driver estiver 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
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 estiver ativo, então o primeiro endereço alternativo de IP (*backup*) é ativado, e assim por diante passando por todos os endereços alternativos de IP e voltando para o endereço principal até conseguir 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 (o Driver se comporta como mestre) ou configure em Verdadeiro para habilitar a recepção de conexões (o Driver se comporta como escravo).

IO.Ethernet.BackupEnable

■ Configure em Verdadeiro para habilitar o endereço IP de reserva (*backup*). Se a tentativa de reconectar com o endereço IP principal falhar, o Driver tenta utilizar o endereço IP de reserva. Configure em Falso para desabilitar sua utilização.

IO.Ethernet.BackupIP

📌 Endereço IP alternativo (*backup*) do equipamento de destino. 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

9 Número da porta local a ser utilizada na conexão ao endereço IP alternativo do equipamento de destino. Usado apenas se **IO.Ethernet.BackupLocalPortEnable** for Verdadeiro.

IO.Ethernet.BackupLocalPortEnable

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

IO.Ethernet.BackupPort

9 Número da porta do endereço IP alternativo do equipamento de destino (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 e fe80:03bf:0877:0000:0000:0000:ffff:ffff)**: 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 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 que não forem 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 que não forem 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, é logada a mensagem "Blocked incoming socket connection from {IP}!".

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, é logada a mensagem "Blocked incoming packet from {IP} (discarding {N} bytes)!".

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 de destino. 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 de destino. 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. Configure em Falso para utilizar qualquer porta local disponível.

IO.Ethernet.MainPort

9 Número da porta IP no equipamento de destino (usado juntamente 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 de destino, 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

A Define o protocolo de transporte. Os valores possíveis são os seguintes:

- **T ou TCP**: Utiliza o protocolo TCP/IP
- **U ou UDP**: Utiliza o protocolo UDP/IP

IO.Ethernet.UseIPv6

■ Configure em Verdadeiro para utilizar endereços IPv6 em todas as conexões Ethernet. Configure em Falso para utilizar endereços IPv4 (este é valor padrão).

Configuração da Interface Modem

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

Tags de Comunicação

Tags da Interface Modem (N2/B2 = 3)

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

IMPORTANTE

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

IO.TAPI.ConnectionBaudRate

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1
Parâmetro N2	0
Parâmetro N3	3
Parâmetro N4	5
Configuração por String	IO.TAPI.ConnectionBaudRate

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

IO.TAPI.Dial

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Escrita
Parâmetro N1	-1
Parâmetro N2	0
Parâmetro N3	3
Parâmetro N4	1
Configuração por String	IO.TAPI.Dial

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

IO.TAPI.ModemStatus

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1
Parâmetro N2	0
Parâmetro N3	3
Parâmetro N4	2
Configuração por String	IO.TAPI.ModemStatus

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

- **"No status!"**: A Interface **Modem** ainda não foi aberta ou já foi fechada
- **"Modem initialized OK!"**: A Interface **Modem** foi inicializada com sucesso
- **"Modem error at initialization!"**: O Driver não conseguiu inicializar a linha do modem. Confira o arquivo de log do Driver para maiores detalhes
- **"Modem error at dial!"**: O Driver não conseguiu começar ou aceitar uma chamada
- **"Connecting..."**: O Driver iniciou uma chamada com sucesso, e está atualmente processando esta chamada
- **"Ringing..."**: Indica que o modem está recebendo uma chamada externa, mas ainda não a aceitou
- **"Connected!"**: O Driver conectou-se com sucesso (completou ou aceitou uma chamada externa)
- **"Disconnecting..."**: O Driver está desligando a chamada atual
- **"Disconnected OK!"**: O Driver desligou a chamada atual
- **"Error: no dial tone!"**: O Driver abortou a chamada porque o sinal de linha disponível não foi detectado
- **"Error: busy!"**: O Driver abortou a ligação porque a linha estava ocupada
- **"Error: no answer!"**: O Driver abortou a chamada porque não recebeu resposta do outro modem
- **"Error: unknown!"**: A chamada atual foi abortada por um erro desconhecido

IO.TAPI.PhoneNumber

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

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

IO.TAPI.HangUp

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Escrita
Parâmetro N1	-1
Parâmetro N2	0
Parâmetro N3	3
Parâmetro N4	4
Configuração por String	IO.TAPI.HangUp

Qualquer valor escrito neste Tag desliga a chamada atual.

NOTA

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

IO.TAPI.IsModemConnected

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1
Parâmetro N2	0
Parâmetro N3	3
Parâmetro N4	3
Configuração por String	IO.TAPI.IsModemConnected

Este Tag indica o estado da conexão do modem. Os valores possíveis são os seguintes:

- **0:** O modem não está conectado, mas pode estar realizando ou recebendo uma chamada externa
- **1:** O modem está conectado e o Driver completou ou recebeu uma chamada externa com sucesso. Enquanto estiver neste estado, a camada física consegue enviar ou receber dados

IO.TAPI.IsModemConnecting

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1
Parâmetro N2	0
Parâmetro N3	3
Parâmetro N4	6
Configuração por String	IO.TAPI.IsModemConnecting

Este Tag indica o estado de conexão do modem, com mais detalhes do que o Tag **IO.TAPI.IsModemConnected**. Os valores possíveis são os seguintes:

- **0**: O modem não está conectado
- **1**: O modem está conectando (realizando ou recebendo uma chamada externa)
- **2**: O modem está conectado. Enquanto estiver neste estado, a camada física consegue enviar ou receber dados
- **3**: O modem está desconectando a chamada atual

Propriedades

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

IO.TAPI.AcceptIncoming

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

IO.TAPI.ModemID

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

NOTA

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

IO.TAPI.PhoneNumber

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

Configuração da Interface RAS

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

Tags de Comunicação

Tags da Interface RAS (N2/B2 = 5)

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


Propriedades

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

NOTA

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

IO.RAS.ATCommand

 Comando **AT** a ser enviado através do *socket* para forçar o equipamento RAS a realizar uma ligação usando o canal RAS atual. Exemplo: "ATDT6265545".

IO.RAS.CommandTimeoutSec

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

Configuração da Interface Serial

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

Tags de Comunicação

Tags da Interface Serial (N2/B2 = 2)

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


Propriedades

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

IO.Serial.Baudrate

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

IO.Serial.CTSTimeoutMs

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

IO.Serial.DataBits

9 Especifica o número de bits de dados para a configuração da porta serial. Os valores possíveis são os seguintes:

- **5:** Cinco bits de dados
- **6:** Seis bits de dados
- **7:** Sete bits de dados
- **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**:

- **OFF:** Sinal **DTR** sempre desligado
- **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 os seguintes:

- **E ou Even:** Paridade par
- **N ou None:** Sem paridade
- **O ou Odd:** Paridade ímpar
- **M ou Mark:** Paridade de marca
- **S ou Space:** Paridade de espaço

IO.Serial.Port

9 Número da porta serial local:

- **1:** Utiliza a porta COM1
- **2:** Utiliza a porta COM2
- **3:** Utiliza a porta COM3
- **n:** Utiliza a porta COM n

IO.Serial.RTS

A Indica como o Driver lida com o sinal **RTS**:

- **OFF:** Sinal **RTS** sempre desligado
- **ON:** Sinal **RTS** sempre ligado
- **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 os seguintes:

- **1:** Um bit de parada
- **2:** Um bit e meio de parada
- **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
4.0.19	05/08/2019	M. Ludwig	<ul style="list-style-type: none"> • Driver portado para o Visual Studio 2017 (Case 27092).
4.0.17	08/02/2018	M. Ludwig	<ul style="list-style-type: none"> • Implementados os tipos de dados Int e DInt em parâmetros sintáticos (Case 23837).

VERSÃO	DATA	AUTOR	COMENTÁRIOS
4.0.16	28/11/2017	M. Ludwig	<ul style="list-style-type: none"> Implementadas as configurações de <i>rack</i>, <i>slot</i> e TSAP de destino para os endereços de <i>backup</i> 2 e 3 (<i>Case</i> 23428).
4.0.13	11/09/2017	M. Ludwig	<ul style="list-style-type: none"> Corrigido um congelamento em leituras usando conexões simultâneas após um tempo indeterminado (<i>Case</i> 23203). Implementado um mecanismo de <i>watchdog</i> para disparar a troca de endereços IP (<i>Case</i> 23270).
4.0.12	06/07/2017	M. Ludwig	<ul style="list-style-type: none"> Corrigido um problema com a leitura de Strings em protocolos ISOTCP e ISOTCP243 com conexões extras (<i>Case</i> 22950).
4.0.11	20/06/2017	M. Ludwig	<ul style="list-style-type: none"> Desabilitada a otimização de leitura agrupada quando configurados os protocolos ISOTCP e ISOTCP243 com otimização de Pedidos Simultâneos (<i>Case</i> 22897).
4.0.10	12/06/2017	C. Mello	<ul style="list-style-type: none"> Ajustes para isolar a Coleta de SOE dos serviços de Superblocos e conexões simultâneas orientadas por <i>callbacks</i> (<i>Case</i> 22785).
4.0.9	29/05/2017	F. Englert	<ul style="list-style-type: none"> Durante uma reconexão em uma CPU de <i>backup</i>, agora há uma verificação adicional se o endereço TSAP de destino corresponde ao novo endereço IP. Caso não seja o endereço IP esperado, o Driver não envia o pedido de conexão, e o refaz imediatamente considerando o novo endereço IP e seu respectivo TSAP (<i>Case</i> 22020).
4.0.7	15/05/2017	M. Ludwig	<ul style="list-style-type: none"> Adicionadas configurações de TSAP de destino em formato numérico hexadecimal (<i>Case</i> 22432).

VERSÃO	DATA	AUTOR	COMENTÁRIOS
4.0.4	10/04/2017	M. Ludwig	<ul style="list-style-type: none"> Implementada a leitura e escrita de <i>database gateway</i> (Case 22249). Corrigido um erro de leitura de BOOL e BYTE unitário (Case 22451).
4.0.1	16/03/2017	M. Salvador M. Ludwig	<ul style="list-style-type: none"> Melhorias de performance no protocolo ISOTCP (Case 22246).
3.1.2	12/05/2015	M. Ludwig	<ul style="list-style-type: none"> Corrigida a negação de funcionamento da opção de seleção do protocolo ISOTCP243 (Case 18675).
3.1.1	19/09/2014	M. Ludwig	<ul style="list-style-type: none"> Implementada a redundância de CPU (seleção automática de CPU de <i>backup</i>, <i>Rack</i> ou <i>Slot</i> alternativos, com a conexão ao endereço IP de <i>backup</i>, Case 15782). Implementada a configuração de <i>Rack</i>, <i>Slot</i> e tipo de conexão na janela de propriedades do Driver (Case 15911). Adicionados os Tags de interface específicos para opção de conexões extras (Case 17221).
3.0.1	20/12/2013	M. Salvador M. Ludwig	<ul style="list-style-type: none"> Implementados os Superblocos internos em conexões TCP extras (Case 14025). Driver portado para o IOKit 2.00 (Case 14019).
2.13.1	21/08/2012	M. Ludwig	<ul style="list-style-type: none"> Implementada a funcionalidade do campo PDU REF no protocolo ISOTCP (Case 13299).
2.12.1	30/05/2012	C. Mello	<ul style="list-style-type: none"> Adicionado suporte para Coleta SOE de eventos em tabelas DB (Case 12483).
2.11.1	04/08/2011	M. Ludwig	<ul style="list-style-type: none"> Incluída a consistência prevista no protocolo MPI e melhorias de codificação (Case 12392). Adicionadas informações sobre suporte a PLC Siemens modelo S7-1200 (Case 12292).

VERSÃO	DATA	AUTOR	COMENTÁRIOS
2.10.1	25/03/2011	M. Ludwig	<ul style="list-style-type: none"> Implementado o formato S7 String e a nova janela de propriedades para configuração de Strings (Case 12005).
2.9.1	25/08/2009	M. Ludwig	<ul style="list-style-type: none"> Corrigida uma falha ao ler variáveis de tipo Contador (Case 10701). Implementadas as configurações avançadas para protocolos ISOTCP e ISOTCP243 (Case 10717).
2.8.1	19/06/2009	M. Ludwig	<ul style="list-style-type: none"> Corrigida uma falha na desconexão endereçando múltiplos escravos no protocolo MPI (Case 10595).
2.7.1	03/06/2009	M. Ludwig	<ul style="list-style-type: none"> Implementado o tipo de dados S5Time (Case 10413).
2.6.1	07/01/2009	M. Ludwig	<ul style="list-style-type: none"> Corrigida uma falha na conexão sob protocolo ISOTCP (Case 10138).
2.5.1	04/11/2008	M. Ludwig	<ul style="list-style-type: none"> Melhorias na apresentação da janela de propriedades (Case 9994). Implementado o <i>delay</i> de operação no protocolo PPI (Case 9968).

VERSÃO	DATA	AUTOR	COMENTÁRIOS
2.4.1	01/04/2008	M. Ludwig	<ul style="list-style-type: none"> Corrigido um problema que ocorria ao endereçar entradas e saídas analógicas combinadas com a propriedade EnableReadGrouping em Verdadeiro (<i>Case 8927</i>). Melhoria e consistência para evitar problema de desconexões do CLP, relatado no <i>Case 8968</i> (recebimento de valores aleatórios em variáveis de alarme no protocolo ISOTCP). Corrigida a falta de tratamento de erro em recebimento de caracteres NAK no protocolo MPI, que ocasionava travamento em recepção de dados (<i>Case 8981</i>). Melhoria de consistência em recepções do protocolo MPI (<i>Case 8981</i>). Retirado um byte desnecessário no <i>frame</i>, que ocasionava problemas em escritas de byte e bit sob protocolo ISOTCP e PLC S7-400 (<i>Case 9021</i>). Corrigida uma falha de reconexão automática em desconexão física no protocolo ISOTCP (<i>Case 9030</i>). Corrigida a implementação de recepção de um <i>frame</i> longo de ACK no protocolo PPI (<i>Case 9118</i>). Implementada a condição de dados indisponíveis no protocolo PPI. Quando a condição é encontrada, retorna uma lista vazia e OK ao invés de falha (<i>Case 9232</i>).

VERSÃO	DATA	AUTOR	COMENTÁRIOS
			<ul style="list-style-type: none"> Corrigida a atribuição errada de <i>Service Access Point</i> em protocolo MPI, que ocasionava falhas de comunicação com adaptadores Tecnatron (<i>Case 9238</i>).
2.3.1	13/09/2007	M. Ludwig	<ul style="list-style-type: none"> Corrigido um problema de reconexão com adaptador serial quando há desligamento de CLP (<i>Case 8069</i>). Implementado o endereçamento a múltiplos escravos no protocolo MPI (<i>Case 8625</i>). Porta Ethernet livremente configurável (<i>Case 8683</i>). Driver compilado no IOKitLib v1.14, de forma a corrigir erros de leitura e escrita antes da primeira conexão (<i>Case 7614</i>). Documentação atualizada com informações sobre tamanho de Strings, protocolos e equipamentos compatíveis (<i>Case 8206</i>).
2.2.1	28/03/2007	M. Ludwig	<ul style="list-style-type: none"> Corrigida a falta de criação de <i>blob</i> que ocasionava erros em tempo de execução (<i>Case 8015</i>). Corrigido um problema de troca de endereços IP em tempo de execução (<i>Case 8026</i>). Adicionado suporte a Windows CE (<i>Case 7504</i>). Adicionado suporte a conversores IBHLink (<i>Case 7994</i>). Corrigido um problema de escrita de Strings (<i>Case 7967</i>).
2.1.1	10/07/2006	M. Ludwig	<ul style="list-style-type: none"> Correção no <i>parsing</i> de variáveis DB (<i>Case 7172</i>).

VERSÃO	DATA	AUTOR	COMENTÁRIOS
2.0.1	13/04/2006	M. Salvador M. Ludwig	<ul style="list-style-type: none"> • Corrigida uma falha do protocolo PPI Error: Single DLE in data field (Case 6644). • Retirada a verificação de endereços. Não importa o tipo de dados, é permitida a entrada de qualquer valor para <i>N4</i> (Case 6644). • Corrigido um erro de interface de configuração, onde se misturavam configurações do conversor IBHLink com o protocolo ISOTCP. A porta 1099 era forçada ao invés da porta 102 (Case 6644). • Adicionado suporte a Superblocos e endereçamento simbólico (Case 6644).
1.1.1	03/11/2005	M. Ludwig	<ul style="list-style-type: none"> • Otimização, padronização e revisão do código fonte.
1.0.1	01/05/2005	M. Salvador	<ul style="list-style-type: none"> • Versão original do Driver.

Matriz

Rua 24 de Outubro, 353 - 10º andar
90510-002 Porto Alegre
Fone: (+55 51) 3346-4699
Fax: (+55 51) 3222-6226
E-mail: elipse-rs@elipse.com.br

Filial PR

Av. Sete de Setembro, 4698/1705
80240-000 Curitiba - PR
Fone: (+55 41) 4062-5824
E-mail: elipse-pr@elipse.com.br

Filial RJ

Praia de Botafogo, 300/525
22250-044 Rio de Janeiro - RJ
Fone: (+55 21) 2158-1015
Fax: (+55 21) 2158-1099
E-mail: elipse-rj@elipse.com.br

Filial SP

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

Filial MG

Rua Antônio de Albuquerque, 156
7º andar Sala 705
30112-010 Belo Horizonte - MG
Fone: (+55 31) 4062-5824
E-mail: elipse-mg@elipse.com.br

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)