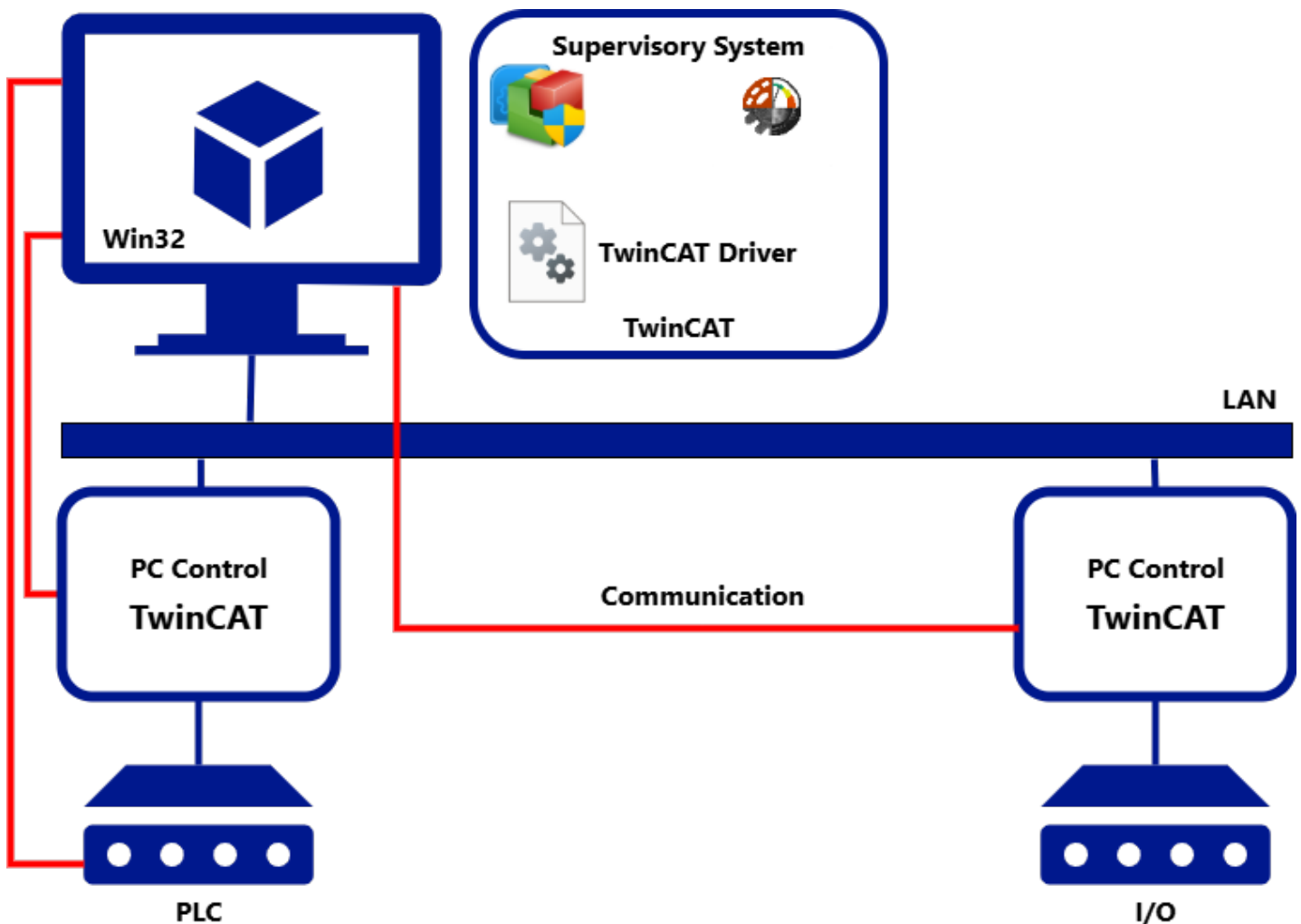


Driver Beckhoff TwinCAT

Nome do Arquivo	TwinCAT.dll
Fabricante	Beckhoff
Equipamentos	CX1000 Embedded PC e demais equipamentos Beckhoff comunicáveis com TwinCAT ADS
Protocolo	Automation Device Specification (ADS)
Versão	2.0.7
Última Atualização	12/06/2025
Plataforma	Win32
Dependências	Software TwinCAT System Control / CLP TwinCAT e Biblioteca TcAdsDll.dll (Win32)
Leitura por Superblocos	Não
Nível	0

Introdução

Este é o Driver Beckhoff TwinCAT para a comunicação entre sistemas da **Elipse Software** e os equipamentos da Beckhoff. A arquitetura do sistema TwinCAT permite que módulos individuais de software sejam tratados como equipamentos independentes, ou seja, para cada tarefa há um módulo de software, cliente ou servidor. Os servidores no sistema são *equipamentos* em execução na forma de software, cujo comportamento operacional é exatamente como o de um equipamento em hardware. Por esta razão pode-se falar em equipamentos *virtuais* implementados em software. Os *clientes* são programas que realizam pedidos de serviços dos *servidores*. As mensagens entre estes objetos são trocadas através de uma interface ADS (*Automation Device Specification*) consistente pelo *roteador de mensagens*. Esta entidade gerencia e distribui todas as mensagens no sistema e sobre as conexões TCP/IP. Os roteadores de mensagens existem em cada computador TwinCAT e em cada controlador de barramento Beckhoff BCxxx. Isto permite que servidores e clientes TwinCAT troquem comandos e dados, enviem mensagens e transfiram informações de status, entre outras tarefas.



Exemplo de topologia TwinCAT / Aplicação Elipse E3 ou Elipse SCADA

Em uma topologia típica de sistema de supervisão e controle, a aplicação de supervisão executando em uma aplicação da **Elipse Software** desempenha o papel de um cliente, realizando pedidos de leitura ou escrita de variáveis ou de memória de CLPs Beckhoff. Estes CLPs, por sua vez, desempenham o papel de servidores.

Preparação do Equipamento

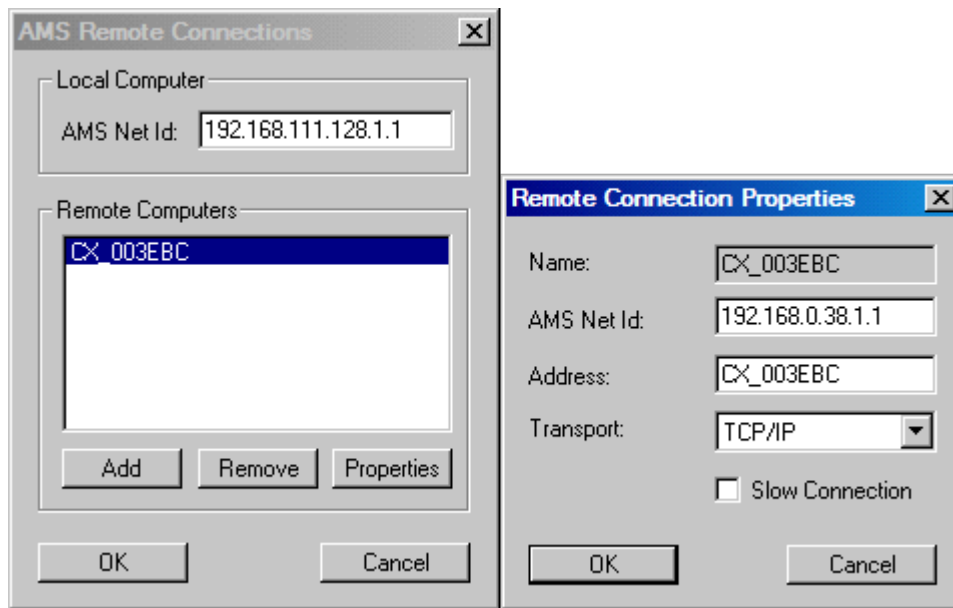
O pré-requisito para uma comunicação bem sucedida de um CLP com uma aplicação de supervisão executando em software da **Elipse Software** é a presença de um programa *Ladder* executando com as variáveis corretamente configuradas neste Driver. O CLP também deve possuir um *Ams Net Id* e uma porta de comunicação disponível conhecidos, a serem corretamente configurados neste Driver.

Já no computador executando a aplicação de supervisão, é essencial a configuração de roteamentos ADS utilizando algum software da Beckhoff.

Ao utilizar o **PLC CX1000 Embedded PC**, as conexões remotas podem ser configuradas por meio do software **AMS Remote Connections**. Este software encontra-se na pasta de instalação do **TwinCAT System Manager**, na subpasta **ADS Api/TcAdsDll**. O software apresenta o *AMS Net Id* local e uma lista de equipamentos remotos. As conexões remotas devem ser adicionadas clicando-se em **Add**, para abrir uma caixa de diálogo com as opções a seguir:

- **Name:** Nome da conexão
- **AMS Net Id:** *Net Id* do computador remoto
- **Address:** Endereço IP ou nome do *host*
- **Transport:** Protocolo de transporte, sempre "TCP/IP"

Este software deve ser executado no **PLC CX1000**, desta vez configurando o computador com a aplicação de supervisão como o computador remoto.



Configuração da aplicação AMS Remote Connections

Em outros equipamentos, a tarefa de roteamento pode ser mais fácil utilizando o software **TwinCAT System Manager**. Neste software é possível identificar todos os equipamentos ADS através de uma varredura em toda a rede e, depois de criada uma lista de equipamentos, seleciona-se os itens aos quais se deseja adicionar o roteamento. Neste caso, o roteamento é realizado bidirecionalmente, ou seja, Computador » Equipamento e Equipamento » Computador, e não somente em um sentido, como no caso do **AMS Remote Connections**.

Terminadas as configurações de roteamentos ADS, reinicie todos dispositivos configurados. Para dúvidas a respeito das aplicações e dos CLPs TwinCAT da Beckhoff, consulte os respectivos manuais.

Configuração do Driver

Os parâmetros **[P]** de configuração deste Driver não são utilizados. Todas as configurações são realizadas na caixa de diálogo de configurações extras do Driver, nas abas **TwinCAT** e **TwinCAT Names**.

NOTA

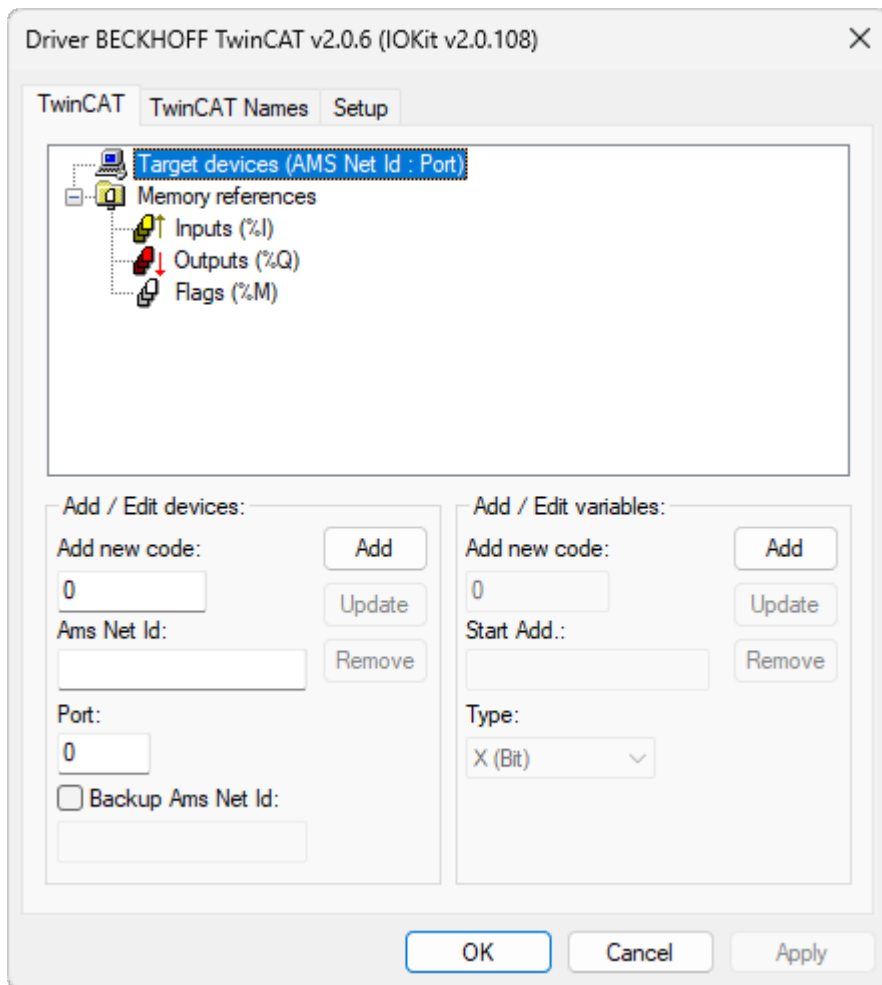
O **Elipse E3**, o **Elipse Power** ou o **Elipse Water** permitem a configuração de Tags por parâmetros sintáticos e, ao optar por este modo de configuração, não é preciso preencher as listas nas abas **TwinCAT** e **TwinCAT Names**. Estas abas somente se referem à configuração por parâmetros numéricos e podem ser completamente ignoradas ao utilizar parâmetros sintáticos.

Aba TwinCAT

Esta aba mostra uma árvore com dois grupos de itens a serem preenchidos. O item **Target Devices** aguarda a inserção de conexões com equipamentos remotos com os quais se deseja comunicar com este Driver, e é necessário configurar no mínimo uma conexão para que se estabeleça uma comunicação bem sucedida.

Já o item **Memory references** aguarda a adição de referências a endereços de memórias, que estão subdivididos entre entradas físicas (**Inputs %I**), saídas físicas (**Outputs %Q**) e memórias de uso geral (**Flags %M**).

As tabelas a seguir apresentam instruções de como preencher estes grupos de itens.



Aba TwinCAT

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

Opções disponíveis no grupo Target devices


OPÇÃO	DESCRIÇÃO
Add new code	Preencha com um número de índice, um código numérico entre 0 (zero) e 255. Caso já exista alguma conexão com o número desejado, este é automaticamente apontado na tabela e os valores são carregados em todos os campos de edição. Este código é a referência para uma conexão com um CLP ou outro equipamento Beckhoff nos Tags. Em outras palavras, o índice atribuído na lista é o que deve ser indicado nos Tags pelo parâmetro <i>N2/B2</i>
Ams Net Id	Configure aqui o <i>Ams Net Id</i> do equipamento de destino principal. Deve ser composto por oito números separados por ponto
Port	Porta do <i>Ams Net Id</i> na qual este Driver deve se conectar
Backup Ams Net Id	Marque a caixa de seleção e configure aqui o <i>Ams Net Id</i> do equipamento de destino de <i>backup</i> , caso haja uma arquitetura de redundância. Deve ser composto por oito números separados por ponto




Opções disponíveis no grupo Memory references

OPÇÃO	DESCRIÇÃO
Add new code	Preencha com um número de índice, um código numérico entre 0 (zero) e 65535. Caso já exista alguma memória com o número desejado, este é automaticamente apontado na lista e os valores são carregados em todos os campos de edição. Este código é a referência para um endereço de variável, ou seja, o índice atribuído na lista deve ser indicado nos Tags de leitura ou escrita de variáveis por endereço pelo parâmetro <i>N3/B3</i>
Start Add	Endereço inicial da variável. Somente é permitida a inserção de números em notação decimal. No caso do tipo de dados Bit , devem ser inseridos dois números separados por um ponto, em que o segundo número indica a posição do bit a ser lido, como por exemplo "100.1", ou seja, o bit 1 (um) da memória de endereço 100
Type	Formato da memória. Os valores possíveis são X (<i>Bit</i>), B (<i>Byte</i>), W (<i>Word</i>), D (<i>Double Word</i>) ou F (<i>Float</i>). As propriedades destes formatos podem ser consultadas na tabela Tipos de Variáveis . O tipo de memória ainda pode variar entre Entrada física (<i>%I</i>), Saída física (<i>%Q</i>) ou Memória interna (<i>%M</i>), dependendo do item selecionado na árvore

As opções que se destinam à manipulação dos dados das listas são as seguintes:

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

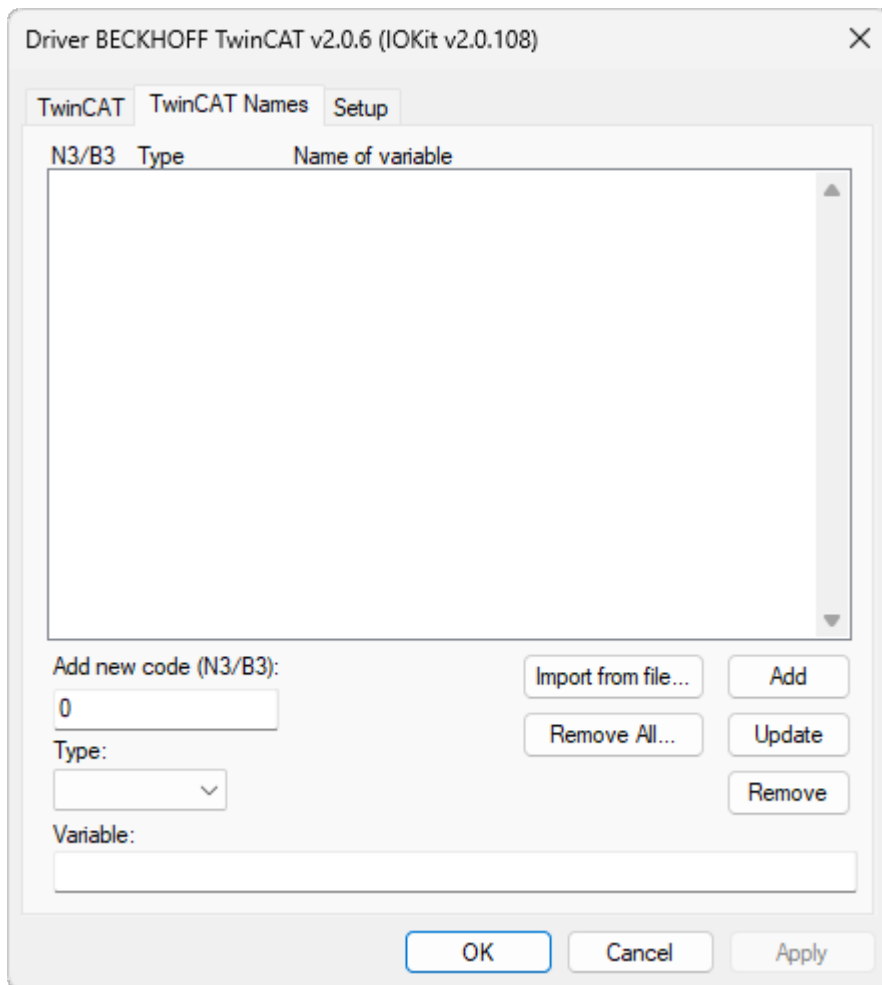
Para adicionar uma referência a um equipamento de destino, selecione o item **Target Devices** , preencha os campos conforme a tabela **Opções disponíveis no grupo Target devices** e clique em **Add**.

Para adicionar uma referência a um campo de memória, selecione o ícone do tipo desejado ( para uma entrada física,  para uma saída física ou  para uma memória interna), preencha os campos conforme indicado na tabela **Opções disponíveis no grupo Memory references** e clique em **Add**.

Para alterar ou remover qualquer das referências existentes, selecione o item que representa a referência e clique em **Update** ou **Remove**.

Aba TwinCAT Names

Similar à aba **TwinCAT**, esta aba apresenta uma lista de referências para nomes de variáveis. É preciso preencher esta lista para ler ou escrever variáveis com nomes e não endereços numéricos.



Aba TwinCAT Names

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

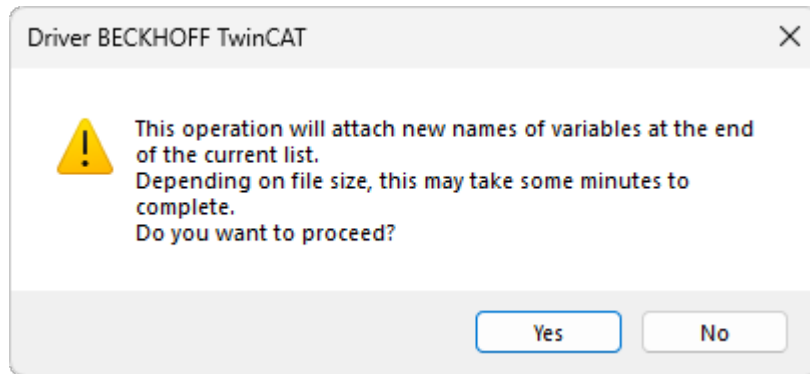
Opções disponíveis na aba TwinCAT Names

OPÇÃO	DESCRIÇÃO
Add new code (N3/B3)	Preencha com um número de índice, um código numérico entre 0 (zero) e 99999. Caso já exista alguma variável com o número desejado, este é automaticamente apontado na tabela e os valores são carregados em todos os campos de edição. Este código é a referência para um nome de variável, ou seja, o índice atribuído na lista deve ser indicado nos Tags de leitura ou escrita de variáveis por nome no parâmetro <i>N3/B3</i>
Type	Tipo de dados da variável. Todos os tipos de dados de variáveis estão listados na tabela Tipos de Variáveis
Variable	Nome da variável. O tamanho máximo é de 100 caracteres

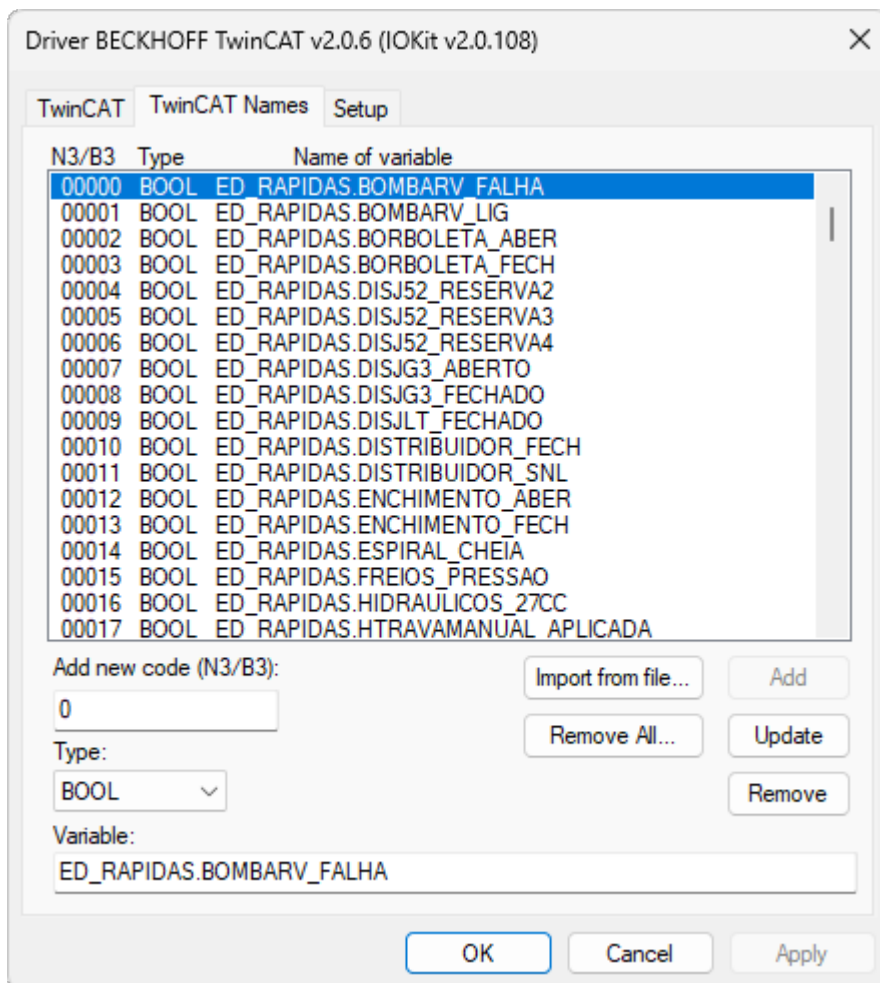
Além das opções **Add**, **Update** e **Remove** para manipulação da lista, a aba **TwinCAT Names** tem ainda opções dedicadas à importação e remoção de nomes de variáveis. A opção **Import from file** abre uma caixa de diálogo de abertura de arquivos de símbolos (SYM), que são arquivos gerados pelo software de design do programa *Ladder* do CLP, **TwinCAT PLC Control**. A operação de importação de um arquivo SYM é muito útil quando um projeto trabalha com muitas variáveis, reduzindo o trabalho de digitar nomes de variáveis para a configuração deste Driver.

Para gerar um arquivo SYM com nomes de variáveis através do software **TwinCAT PLC Control**, é necessário habilitar todos os itens contidos na caixa de diálogo aberta pela opção **Configure symbol file** do menu **Options - Symbol configuration**. Após habilitar os itens, execute a opção **Project - Rebuild all** para gerar um arquivo SYM com todos os nomes de variáveis em um projeto.

Para realizar a importação dos nomes de variáveis de um arquivo SYM, clique em **Import from file** e selecione o arquivo que deseja importar. Ao confirmar a operação, inicia-se o processo de leitura e conversão do arquivo que, dependendo do tamanho e dos recursos disponíveis de processamento do computador, pode ser demorado. Neste caso, uma barra indica o progresso da importação.



Mensagem de importação



Resultado da importação de um arquivo SYM

A opção **Remove all** destina-se à remoção de todos os nomes de variáveis da lista. Clique em **Yes** para confirmar esta operação de remoção.

Outras Configurações

Na aba **Setup** deve-se selecionar o item **None** na opção **Physical Layer**, indicando que nenhuma interface física é acionada diretamente por este Driver. Esta opção já vem configurada como padrão ao adicionar um novo objeto Driver em uma aplicação.

Quaisquer outras configurações além das já citadas até aqui são irrelevantes e não surtem qualquer efeito, por isto devem ser desconsideradas e deixadas com os respectivos valores padrão.

Clique em **OK** para salvar todas as configurações e fechar a janela de propriedades deste Driver.

Tipos de variáveis

TIPO DE DADOS	TAMANHO EM BITS	FORMATO	FAIXA DE VALORES
BIT	1	Bit	0 (zero) ou 1 (um)
BOOL	8	Booleano	FALSE ou TRUE
BYTE	8	Byte	0 (zero) a 255
WORD	16	Word	0 (zero) a 65535
DWORD	32	Double Word	0 (zero) a 4294967295
SINT	8	Inteiro de oito bits com sinal	-128 a 127
USINT	8	Inteiro de oito bits sem sinal	0 (zero) a 255
INT	16	Inteiro de 16 bits com sinal	-32768 a 32767
UINT	16	Inteiro de 16 bits sem sinal	0 (zero) a 65535
DINT	32	Inteiro de 32 bits com sinal	-2147483648 a 2147483647
UDINT	32	Inteiro de 32 bits sem sinal	0 (zero) a 4294967295
REAL ou FLOAT	32	Ponto flutuante de precisão simples no padrão IEEE 754	-3.402823e+38 a 3.402823e+38
LREAL ou DOUBLE	64	Ponto flutuante de precisão dupla no padrão IEEE 754	-1.79769313486231e308 a 1.79769313486232e308
STRING	-	Texto no formato ASCII	Máximo de 255 caracteres
TIME	32	Tempo em milissegundos	0ms a 71582m47s295ms
TIME_OF_DAY (TOD)	32	Hora do dia em milissegundos	00:00:00 a 1193:02:47.295
DATE	32	Data atual	1970-01-01 a 2106-02-06
DATE_AND_TIME (DT)	32	Data e hora atuais	1970-01-01-00:00 a 2106-02-06-06:28:15

Referência de Tags

Os Tags neste Driver podem ser referenciados no **Elipse E3**, no **Elipse Power** ou no **Elipse Water** através de **Strings** nos campos **Dispositivo** e **Item** ou então por parâmetros numéricos **[N/B]**.

Configuração por Parâmetros Sintáticos

O **Elipse E3**, a partir da versão **2.0**, o **Elipse Power** e o **Elipse Water** permitem a configuração de Tags por parâmetros sintáticos. Utilize a sintaxe a seguir para cada campo:

- O campo **Dispositivo** deve obedecer à sintaxe **<Ams Net Id principal>:<Número da Porta>[{Ams Net Id backup}]**:
 - **<Ams Net Id principal>** deve ser preenchido com o *Net Id* do equipamento de destino, composto por oito números separados por ponto
 - **<Número da porta>** deve ser preenchido com o valor da porta do *Ams Net Id* à qual este Driver deve se conectar
 - O item **{Ams Net Id backup}** é opcional e, caso selecionado, deve também ser composto por oito números separados por ponto
- O campo **Item** deve obedecer uma sintaxe característica para cada Tag:
 - **<Nome da variável>:<Tipo>** para **Variáveis por Nome**
 - **%<Área-Tipo><Endereço inicial>** para **Variáveis por Endereço**
 - Preencher com a **String** "TwinCAT.ADSSstatus" para o **Tag homônimo**
 - Idem para o Tag **TwinCAT.NetIdSelect**
 - Idem para o Tag **TwinCAT.NetIdSwitch**

Configuração por Parâmetros Numéricos (N/B)

Quando parâmetros *N/B* são utilizados, os Tags são referenciados a comandos através da configuração do parâmetro *N1/B1*. O parâmetro *N2/B2*, em qualquer Tag, é dedicado à configuração do terminal de destino ao qual um comando deve ser direcionado. Preencha-o com o índice numérico correspondente ao que foi atribuído na lista de conexões da caixa de diálogo de configurações, como explicado no item **Configuração do Driver**.

NOTA

Os parâmetros *N/B* não mencionados para preenchimento são irrelevantes na configuração dos Tags. Recomenda-se que estes valores sejam deixados em 0 (zero).

Tag de Variáveis por Nome

Leitura e Escrita

Configuração por parâmetros numéricos

PARÂMETRO	VALOR
N1/B1	0 (zero)
N2/B2	Terminal de destino
N3/B3	Referência de nome
N4/B4	0 (zero)

O parâmetro *N3/B3* deve ser configurado com o índice de uma das variáveis preenchidas na lista dos parâmetros extras. Para isto, consulte a aba **TwinCAT Names**.

Configuração por parâmetros sintáticos

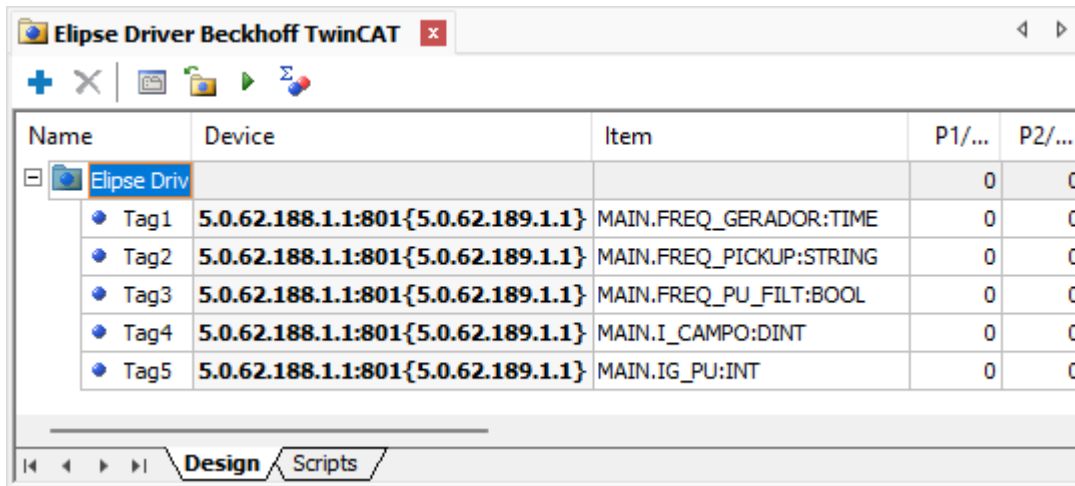
PARÂMETRO	VALOR
Dispositivo	<Ams Net Id principal>:<Número da Porta>[{{Ams Net Id backup}}]
Item	<Nome da variável>:<Tipo>

- **Campo Valor:** O campo **Valor** apresenta o conteúdo da variável, independente do tipo de dados que a variável representa.

Este Tag pode ser tanto um Tag de Comunicação quanto um Tag Bloco. Utilize um Tag de Comunicação ou um Tag Bloco de um Elemento para representar uma variável simples. Utilize um Tag Bloco com mais de um Elemento para representar um vetor (*array*). As **Strings** para tipos de dados são as seguintes:

- **BOOL**
- **BYTE**
- **WORD**
- **DWORD**
- **SINT**
- **USINT**
- **INT**
- **UINT**
- **DINT**
- **UDINT**
- **REAL**
- **LREAL**
- **STRING**
- **TIME**
- **TOD**
- **DATE**
- **DT**

O significado destas **Strings** de tipos de dados pode ser consultado na tabela **Tipos de Variáveis**.



Exemplo de configuração sintática de um Tag de Variáveis por Nome

Tag de Variáveis por Endereço

Leitura e Escrita

Configuração por parâmetros numéricos

PARÂMETRO	VALOR
N1/B1	1 (um)
N2/B2	Terminal de destino
N3/B3	Referência de endereço
N4/B4	0 (zero)

O parâmetro *N3/B3* deve ser configurado com o índice de uma das variáveis adicionadas na lista de referências à memórias dos parâmetros extras. Para isto, consulte a aba **TwinCAT**.

Configuração por parâmetros sintáticos

PARÂMETRO	VALOR
Dispositivo	<Ams Net Id principal>:<Número da Porta>[{Ams Net Id backup}]
Item	%<Área-Tipo><Endereço inicial>

- **Campo Valor:** Este campo apresenta o conteúdo da variável, independente do tipo de dados que esta variável representa.

Este Tag pode ser tanto um Tag de Comunicação quanto um Tag Bloco. Utilize um Tag de Comunicação ou um Tag Bloco de um Elemento para representar uma variável simples lida na posição de memória referenciada. Utilize um Tag Bloco com mais de um Elemento para representar a leitura de blocos de memória em sequência, onde cada Elemento reflete uma posição de memória diferente, intercalada com a do Elemento vizinho pelo espaço ocupado pelo tipo de dados da variável. Isto se torna valioso quando se deseja ler de forma mais otimizada muitos elementos de memória em um CLP que estão próximos entre si.

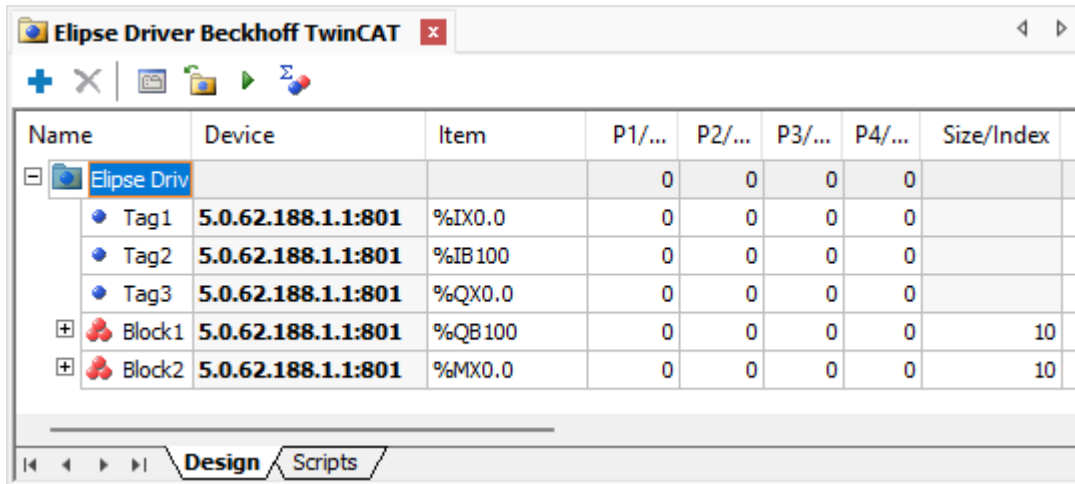
Por exemplo, os campos de memória **%MW100** e **%MW104** podem ser lidos ao mesmo tempo em um Tag Bloco de três Elementos. Para isto, configure o endereço inicial como 100. O primeiro Elemento contém o valor de **%MW100** e o

terceiro contém o valor de **%MW104**. Isto já apresenta vantagem sobre a leitura destes mesmos campos se referenciados separadamente em Tags de Comunicação, mesmo considerando que **%MW102**, o campo localizado entre ambos, seja lido sem utilidade.

A exceção fica por conta de memórias do tipo de dados **X (Bit)**, que só podem ser lidas ou escritas por Tags de um Elemento, de Comunicação ou Bloco.

Caracteres de área e tipo de dados

STRING DE ÁREA E TIPO DE DADOS	SIGNIFICADO
IX	Entrada física individual do tipo de dados Bit
IB	Entradas físicas interpretadas como um tipo de dados Byte de 8 (oito) bits
IW	Entradas físicas interpretadas como um tipo de dados Word de 16 bits
ID	Entradas físicas interpretadas como um tipo de dados Double Word de 32 bits
IF	Entradas físicas interpretadas como um tipo de dados Float de 32 bits
QX	Saída física individual do tipo de dados Bit
QB	Saídas físicas interpretadas como um tipo de dados Byte de 8 (oito) bits
QW	Saídas físicas interpretadas como um tipo de dados Word de 16 bits
QD	Saídas físicas interpretadas como um tipo de dados Double Word de 32 bits
QF	Saídas físicas interpretadas como um tipo de dados Float de 32 bits
MX	Memória interna do tipo de dados Bit
MB	Memória interna do tipo de dados Byte de 8 (oito) bits
MW	Memória interna do tipo de dados Word de 16 bits
MD	Memória interna do tipo de dados Double Word de 32 bits
MF	Memória interna do tipo de dados Float de 32 bits



Tags de Variáveis por Endereço

Tag ADSStatus

Leitura e Escrita

Configuração por parâmetros numéricos

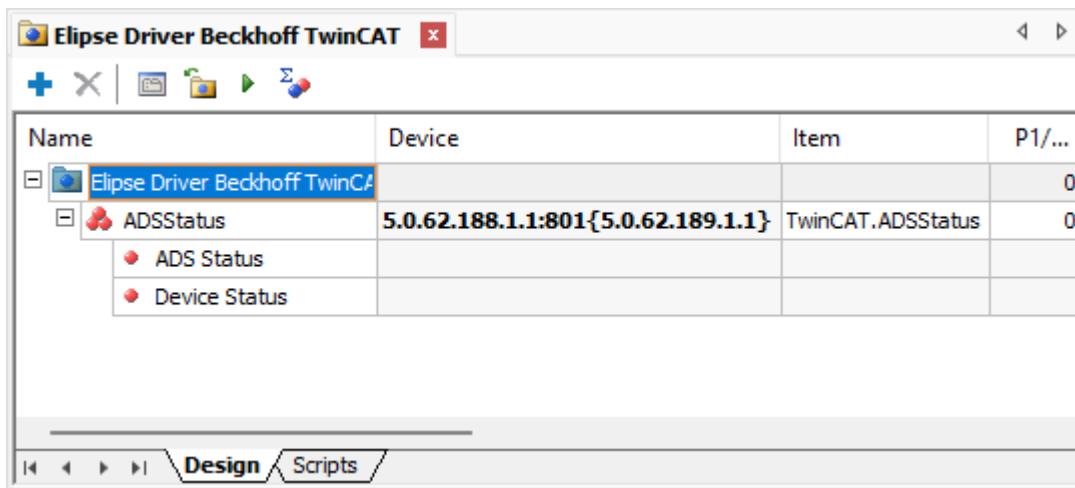
PARÂMETRO	VALOR
N1/B1	2 (dois)
N2/B2	Terminal de destino
N3/B3	0 (zero)
N4/B4	0 (zero)

Configuração por parâmetros sintáticos

PARÂMETRO	VALOR
Dispositivo	<Ams Net Id principal>:<Número da Porta>[<Ams Net Id backup>]
Item	TwinCAT.ADSStatus

Lê ou escreve o status **ADS** e **Device**. Necessita de um Tag Bloco com dois Elementos:

- **Elemento 1:** ADS Status
- **Elemento 2:** Device Status



Exemplo de configuração sintática de um Tag ADSstatus

Tag NetIdSelect

Leitura e Escrita

Configuração por parâmetros numéricos

PARÂMETRO	VALOR
N1/B1	10
N2/B2	Terminal de destino
N3/B3	0 (zero)
N4/B4	0 (zero)

Configuração por parâmetros sintáticos

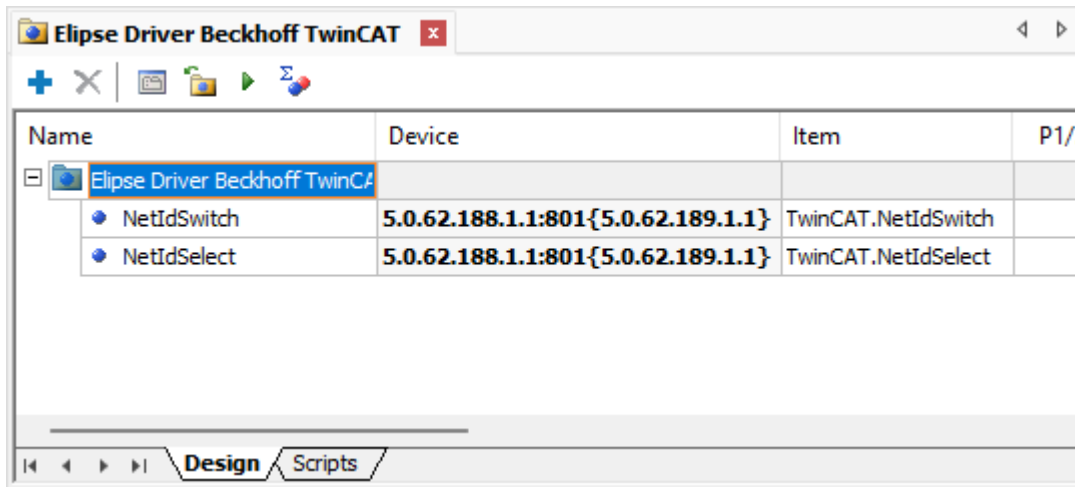
PARÂMETRO	VALOR
Dispositivo	<Ams Net Id principal>:<Número da Porta>[<Ams Net Id backup>]
Item	TwinCAT.NetIdSelect

- **Campo Valor:** Indica qual dos *Ams Net Id* está ativo:
 - **0:** O *Net Id* principal está selecionado (ativo)
 - **1:** O *Net Id backup* está selecionado (ativo)

Se a conexão está ativa, este Tag indica qual dos dois *Net Id* configurados está em uso. Se está desconectado, este Tag indica qual dos *Net Id* é usado primeiro na próxima tentativa de conexão.

Se durante um processo de conexão o *Net Id* ativo não está disponível, então este Driver tenta conectar-se com o outro *Net Id*. Se a conexão ao *Net Id* alternativo é bem-sucedida, então este é o *Net Id* ativo, ou seja, ocorre uma troca automática.

Para forçar uma troca manual, escreva o valor 0 (zero) ou 1 (um) neste Tag. Isto força com que a comunicação seja direcionada a outro *Net Id* (0: *Net Id* principal ou 1: *Net Id backup*) se este Driver está atualmente conectado. Se este Driver está desconectado, então apenas configura o *Net Id* ativo para a próxima tentativa de conexão.



Exemplo de configuração sintática de Tags NetId

Tag NetIdSwitch

Somente Escrita

Configuração por parâmetros numéricos

PARÂMETRO	VALOR
N1/B1	11
N2/B2	Terminal de destino
N3/B3	0 (zero)
N4/B4	0 (zero)

Configuração por parâmetros sintáticos

PARÂMETRO	VALOR
Dispositivo	<Ams Net Id principal>:<Número da Porta>[<Ams Net Id backup>]
Item	TwinCAT.NetIdSwitch

Escreva qualquer valor neste Tag para enviar um comando a este Driver para realizar uma troca manual de *Ams Net Id* de destino. Se o *Net Id* principal estava ativo, então o *Ams Net Id* de backup é ativado e vice-versa.

Suporte à Redundância

O suporte à redundância entre dois equipamentos remotos, um principal e outro de *backup*, é automaticamente selecionado toda vez que, para um dado dispositivo, é também configurado um dispositivo de *backup*.

Existindo um endereço de *backup* configurado, este Driver cria uma linha de execução exclusiva para verificação do estado da conexão, que é periodicamente executada a cada segundo. Em caso de falha na conexão do *Ams Net Id* principal, o *backup* é selecionado automaticamente. Da mesma forma, no momento em que o endereço de *backup* está selecionado e apresenta falha, o *Ams Net Id* principal é novamente selecionado.

Para controlar manualmente as conexões com os dispositivos principal e *backup*, utilize os Tags **NetIdSelect** ou **NetIdSwitch**, explicados anteriormente.

Documentação das Interfaces de Comunicação

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

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

Setup

Physical Layer: Ethernet v
 Start driver OFFLINE

Timeout: 1000 ms
Communication check time: 5000 ms

Connection management

Mode: Automatic (managed by the driver) v

Retry failed connection every 20 seconds

Give up after 1 failed retries

Disconnect if non-responsive for 0 seconds

Logging Options

Log to File: C:\eeLogs\MicrolokII_%DATE%.log

File size limit (MB): 0 ('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 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	Selecione 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	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
File size limit (MB)	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

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.

Histórico de Revisões do Driver

VERSÃO	DATA	AUTOR	COMENTÁRIOS
2.0.7	12/06/2025	M. Ludwig	<ul style="list-style-type: none"> • Driver atualizado para a biblioteca IOKit versão 3.0 e para o Visual Studio 2022 (Case 37496).
2.0.6	09/08/2019	M. Ludwig	<ul style="list-style-type: none"> • Driver portado para o Visual Studio 2017 (Case 27293).
2.0.5	25/08/2016	M. Ludwig	<ul style="list-style-type: none"> • Corrigida uma falha na leitura de variáveis do tipo de dados String (Case 21390).
2.0.4	16/08/2016	M. Ludwig	<ul style="list-style-type: none"> • Aumentado o tamanho máximo de caracteres no tipo de dados String (Case 21324).
2.0.3	27/08/2014	M. Ludwig	<ul style="list-style-type: none"> • Corrigida uma falha na análise sintática dos nomes de variáveis (Case 16869).
2.0.2	26/12/2013	M. Ludwig	<ul style="list-style-type: none"> • Driver portado para a biblioteca IOKit versão 2.0 (Case 14630). • Implementados os endereços de <i>backup</i> e de redundância (Case 14841). • Corrigida a incorreta permissão de configurar uma variável com um nome vazio (Case 14872). • Implementada a leitura em funções de <i>callback</i> (Case 14876). • Implementado um impedimento interno à seleção da camada física de comunicação (Case 14982). • Atualização da biblioteca TcAdsDII da Beckhoff (Case 14998). • Implementado o carregamento dinâmico da biblioteca TcAdsDII (Case 15036).

VERSÃO	DATA	AUTOR	COMENTÁRIOS
1.3.1	10/08/2012	G. Taschetto	<ul style="list-style-type: none"> • Corrigida uma falha ao realizar uma operação em um bit superior a sete (<i>Case 12989</i>).
1.2.1	03/12/2007	M. Ludwig	<ul style="list-style-type: none"> • Corrigido o acesso individual a Elementos de Bloco, ou seja, variáveis do tipo Array (<i>Case 8734</i>). • Alterada a limitação do tamanho da lista de variáveis para evitar perda de dados (<i>Case 8906</i>).
1.1.1	13/12/2006	M. Ludwig	<ul style="list-style-type: none"> • Corrigida uma retenção de símbolos inválidos quando um PLC é reiniciado (<i>Case 7690</i>).
1.0.1	14/09/2006	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)