

# Driver MQTT

Nome do Arquivo	MQTT.dll
Fabricante	
Equipamentos	Clientes e Brokers compatíveis com as especificações 3.1 e 3.1.1 do MQTT
Protocolo	MQTT sobre Ethernet TCP/IP e TLS/SSL
Versão	1.0.14
Última Atualização	04/05/2020
Plataforma	Win32
Dependências	IOKit v2.00 e a biblioteca OpenSSL
Leitura com Superblocos	Não
Nível	0

## Introdução

O Driver MQTT comunica com qualquer cliente que implemente a versão 3.1.X do protocolo MQTT, através de um *Broker* MQTT, não fornecido.

Este Driver é capaz de receber e extrair valores de mensagens recebidas, bem como enviar mensagens que são processadas por outros clientes.

As conexões entre os clientes não são realizadas diretamente, somente através de um *Broker*.

# Parâmetros de Configuração do Driver

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

The screenshot shows a configuration window titled "Driver MQTT v1.0.13 [BETA Apr 14 2020 09:16:32] (IOLite v2.0.116)". It has five tabs: General, Publish, Templates, Setup, and Ethernet. The General tab is active. It contains the following fields and options:

- Version:** A dropdown menu showing "3.1.1".
- KeepAlive (s):** A text input field containing "5".
- Client ID:** An empty text input field.
- User:** An empty text input field.
- Password:** An empty text input field.
- Read QoS:** A dropdown menu showing "0 (No Confirm)".
- Resend Unanswered Subscriptions After (s):** A text input field containing "0".
- ☐ Use Session Control
- ☐ Use Last Will Msg for Hierarchical Quality

At the bottom right, there are three buttons: OK, Cancel, and Apply.

**Aba General**

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

**Opções disponíveis na aba General**

OPÇÃO	DESCRIÇÃO
<b>Version</b>	A versão do protocolo. As opções disponíveis são <b>3.1</b> e <b>3.1.1</b>
<b>KeepAlive (s)</b>	Tempo, em segundos, que o cliente envia uma mensagem de <b>ping</b> no protocolo MQTT ( <b>PingReq</b> ) para verificar se a conexão com o <i>Broker</i> ainda está ativa. O valor padrão para esta opção é 5 (cinco) segundos
<b>Client ID</b>	Nome que este cliente utiliza como identificador para os demais clientes. Não deve haver outro cliente conectado no mesmo <i>Broker</i> com este nome
<b>User</b>	Nome de usuário usado nas mensagens de conexão ( <b>ConnAck</b> ) caso desejado e permitido pelo <i>Broker</i>
<b>Password</b>	Senha usada juntamente com o nome de usuário

OPÇÃO	DESCRIÇÃO
<b>Read QoS</b>	Indica que QoS é usado para solicitar as assinaturas dos Tags (QoS 0, 1 or 2). O <i>Broker</i> compara o QoS solicitado e o QoS originalmente publicado, enviando um dado no mínimo compatível
<b>Resend Unanswered Subscriptions After (s)</b>	Informe o número de segundos para reenviar uma assinatura de um item se este não recebeu o primeiro valor dentro do tempo informado
<b>Use Session Control</b>	Informa se este Driver deve manter o estado da sessão ativa, ou seja, caso ocorra uma desconexão e posterior reconexão, as mensagens <b>QoS 1 e 2</b> enviadas por outros clientes podem ser recuperadas
<b>Use Last Will Msg for Hierarchical Quality</b>	Ao selecionar esta opção, se este Driver receber a mensagem configurada na opção <b>Message</b> do grupo <b>Last Will and Testament</b> da aba <b>Publish</b> , então o tópico da mensagem é utilizado para definir hierarquicamente outros Tags que são configurados com qualidade ruim

Supondo que a mensagem de *Last Will* de um equipamento é configurada com a expressão "Error" para o item **Device527**, ao receber esta mensagem todos os Tags endereçados que contiverem a expressão "Device527" a partir do início do endereço têm sua qualidade configurada como ruim, ou seja, a propriedade **Quality** tem seu valor configurado para 20, 24 ou 28 (inválida). Exemplos de endereçamentos de equipamentos:

```
Device527/Temperature
Device527/AnalogInput01
Device527
```

Para que estes Tags retornem à qualidade boa (192), o tópico deve receber algum valor diferente da mensagem configurada em **Last Will and Testament**.

Driver MQTT v1.0.13 [BETA Apr 17 2020 16:25:33] (IOMKit v2.0.116)

General Publish Templates Setup Ethernet

QoS: 0 (No Confirm) ▼

☐ Retain Messages on Server

Last Will and Testament

☐ Enable Last Will      Topic:

Message:

OK Cancel Apply

#### Aba Publish

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

#### Opções disponíveis na aba Publish

OPÇÃO	DESCRIÇÃO
<b>QoS</b>	Indica o nível de serviço das mensagens publicadas por este Driver. As opções disponíveis são <b>QoS 0</b> : Quase sempre uma entrega (pode haver zero ou mais entregas), <b>QoS 1</b> : Pelo menos uma ou mais entregas ou <b>QoS 2</b> : Exatamente uma entrega
<b>Retain messages on Server</b>	Selecionar esta opção instrui ao <i>Broker</i> para manter as mensagens deste cliente juntamente com o QoS para que clientes que se inscreverem no futuro possam receber mensagens acumuladas. Quando um novo cliente se inscreve, a última mensagem de cada tópico, se houver, deve ser enviada ao cliente pelo <i>Broker</i>
<b>Enable Last Will</b>	Ao habilitar esta opção, o servidor deve manter em memória um tópico e mensagem para o cliente, que são enviados para todos os demais clientes quando houver uma desconexão não ordeira, ou seja, sem o envio da mensagem de <i>Disconnect</i>
<b>Topic</b>	Tópico ou endereço que é enviado na desconexão ( <i>Last Will</i> ), como por exemplo "ELIPSE\goodbye"
<b>Message</b>	Mensagem associada ao tópico, como por exemplo "Adeus"

Na aba **Templates** é possível definir diferentes padrões para os dados da mensagem MQTT, cujos elementos podem ser automaticamente extraídos ou preenchidos por este Driver através de Associações com Tags ou Blocos de Comunicação.

**Aba Templates**

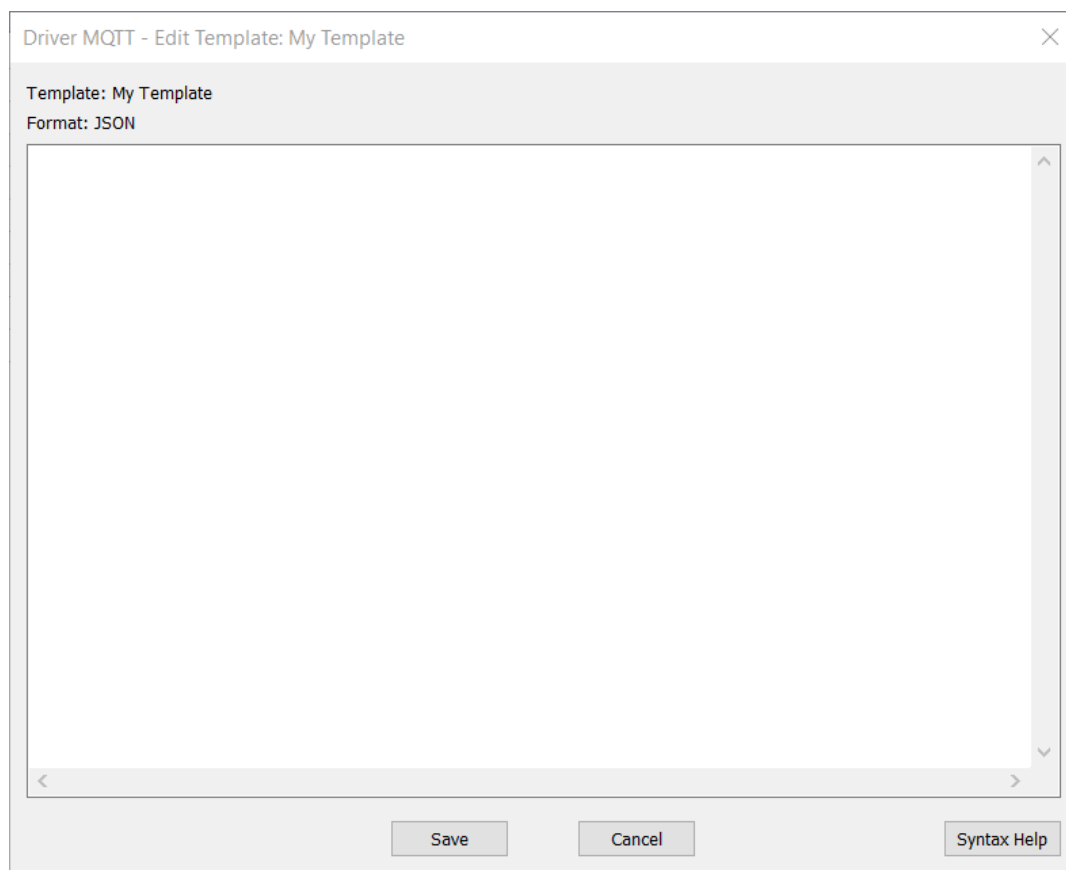
As opções disponíveis na aba **Templates** estão descritas na tabela a seguir.

**Opções disponíveis na aba Templates**

OPÇÃO	DESCRIÇÃO
<b>Apply local timezone to timestamps</b>	Quando esta opção está selecionada, as estampas de tempo recebidas são convertidas para o fuso horário utilizado pelo sistema operacional. Do contrário, as estampas de tempo recebidas são exibidas no fuso horário UTC
<b>Template Name</b>	Informa o nome do <i>Template</i> desejado. Não podem existir dois <i>Templates</i> com o mesmo nome
<b>Data Format</b>	Declara o formato da mensagem do <i>Template</i> . As opções disponíveis são <b>JSON</b> , <b>CSV</b> ou <b>XML</b> . O formato <b>JSON</b> é o preferido para uso pelo MQTT, pois permite a definição de qualquer tipo de estrutura de dados com o menor número de bytes
<b>Add</b>	Adiciona um <i>Template</i> na lista de <i>Templates</i>
<b>Edit</b>	Abre uma janela para edição do <i>Template</i> selecionado

OPÇÃO	DESCRIÇÃO
<b>Delete</b>	Apaga o <i>Template</i> selecionado da lista de <i>Templates</i>

Clique em **Edit** para abrir a janela para edição do *Template* selecionado. O nome e o formato do *Template* são exibidos na parte superior. O *Template* propriamente dito deve ser inserido na caixa de edição.



**Janela para edição de um Template**

As opções disponíveis na janela para edição do *Template* estão descritas na tabela a seguir.

#### Opções disponíveis na janela para edição do Template

OPÇÃO	DESCRIÇÃO
<b>Save</b>	Registra as modificações realizadas e fecha esta janela
<b>Cancel</b>	Descarta as modificações realizadas e fecha esta janela
<b>Syntax Help</b>	Informa a sintaxe da mensagem, no formato selecionado ( <b>JSON</b> , <b>CSV</b> ou <b>XML</b> ) juntamente com palavras-chave que são usadas durante o processamento para substituição por estampas de tempo, qualidades ou valores. Para mais informações, consulte o tópico <b>Sintaxe dos Templates</b>

Na aba **Ethernet** deve ser configurado o endereço IP e a porta TCP/IP do *Broker* MQTT. A norma MQTT estabelece a porta TCP/IP 1883 como padrão para conexões diretas sem criptografia. A porta TCP/IP 8883 é definida para conexões criptografadas TLS/SSL sem certificado e a porta TCP/IP 8884 é definida para conexões TLS/SSL com certificado.

# Sintaxe dos Templates

Para extrair o conteúdo de uma mensagem é necessário declarar um *Template*, que permite informar o formato da mensagem e quais partes devem ser transformadas em dados. Os *Templates* podem ser do tipo **JSON**, **CSV** ou **XML**. Cada *Template* deve utilizar palavras-chave, que devem ser colocadas no lugar dos valores a serem extraídos. As palavras-chave disponíveis estão descritas na tabela a seguir.

**Palavras-chave disponíveis para Templates**

PALAVRA-CHAVE	DESCRIÇÃO
<b>TS_TEXT(format)</b>	Estampa de tempo, em formato textual, que é usada como estampa de tempo do Bloco ou Tag de Comunicação. O significado de cada campo está descrito na tabela <b>Opções disponíveis para a palavra-chave TS_TEXT</b>
<b>TS_UNIX</b>	Estampa de tempo, no formato de segundos, desde 1970 (formato <b>UNIX</b> ). Este valor pode ser numérico ou textual, e é usado como estampa de tempo do Bloco ou Tag de Comunicação, como por exemplo 1504198675 ou "1504198675"
<b>TS_DAYS1900</b>	Estampa de tempo, em formato numérico cujo valor é o número de dias desde 01-jan-1900 (também conhecido como calendário gregoriano), e será usado como estampa de tempo do Bloco ou Tag de comunicação. Ex: 43595.37373843
<b>TS_UNIX_TZ_SECONDS+</b> <b>TS_UNIX_TZ_SECONDS-</b>	Fuso horário, em segundos, informado separadamente da palavra-chave <b>TS_UNIX</b> . Deve ser informado um símbolo + ou - para indicar se os segundos devem ser adicionados ou removidos, respectivamente, da palavra chave <b>TS_UNIX</b> para representar a estampa de tempo da mensagem
<b>QL_OPC</b>	Qualidade, no padrão OPC DA, utilizando um byte. Este valor é usado diretamente como valor de qualidade no Bloco ou Tag de Comunicação, sem transformações. Consulte a tabela <b>Padrão OPC DA</b> para mais informações
<b>QL_BOOL</b>	Qualidade no padrão Booleano. Se o valor é maior que 0 (zero) ou a expressão "true" ou "TRUE", a qualidade é boa. Se o valor é igual a 0 (zero) ou a expressão "false" ou "FALSE", a qualidade é ruim
<b>V1, V2, ... V50</b>	Declara um ou mais valores, que podem ser extraídos individualmente para Tags de Comunicação, através da especificação do terceiro parâmetro de endereçamento ( <i>param</i> ). Se todos os valores de uma mensagem são mapeados para um Bloco único, então é possível usar a palavra-chave <b>E3VAL</b> para qualquer valor a ser extraído, indistintamente
<b>E3VAL</b>	Especifica um valor que é extraído na sequência que ocorre para um Bloco, cada valor em um Elemento. Se houver apenas um valor <b>E3VAL</b> na mensagem, o Template também pode ser usado com um Tag de Comunicação
<b>DUMMY</b>	Campo que é variável, mas cujo valor não deve ser enviado para Tags de Comunicação

### Opções disponíveis para a palavra-chave TS\_TEXT

OPÇÃO	DESCRIÇÃO
%a	Dia da semana abreviado, em inglês
%A	Dia da semana completo, em inglês
%b	Nome do mês abreviado, em inglês
%B	Nome do mês completo, em inglês
%C	Século
%d	Dia do mês, começando com 0 (zero)
%e	Dia do mês, começando com um espaço
%f	Milissegundos, de 0 (zero) a 999
%h	Hora, no formato de 12 horas
%H	Hora, no formato de 24 horas
%m	Mês
%M	Minuto
%p	AM ou PM
%S	Segundos
%y	Ano com dois dígitos
%Y	Ano com quatro dígitos
%Z	Nome do fuso horário, um código internacional que transforma o horário para GMT
%+	Deslocamento (offset) GMT no formato +-HH:MM

Exemplos de estampas de tempo formatados pela palavra-chave **TS\_TEXT**:

```
"2014-07-11T15:26:37Z" -> "TS_TEXT(%y-%m-%dT%H:%M:%SZ)"
"Mon Jul 10 11:04:47 BRT 2017" -> "TS_TEXT(%a %b %d %H:%M:%S %Z %Y)"
"2018-05-02T10:29:28.622-02:00" -> "TS_TEXT(%Y-%m-%dT%H:%M:%S.%f%+)"
```

### Padrão OPC DA

BIT	7	6	5	4	3	2	1	0
Descrição	Q	Q	S	S	S	S	L	L

### Opções disponíveis para o padrão OPC DA

OPÇÃO	DESCRIÇÃO
QQ	Dois bits de qualidade
SSSS	Quatro bits de substatus
LL	Dois bits de limite
QQ	<b>0</b> : BAD, <b>1</b> : UNCERTAIN ou <b>3</b> : GOOD



OPÇÃO	DESCRIÇÃO
SSSS	<b>0:</b> BAD_NONSPECIFIC, <b>1:</b> BAD_CONFIGERROR, <b>2:</b> BAD_NOTCONNECTED, <b>3:</b> BAD_DEVICEFAILURE ou <b>4:</b> BAD_SENSORFAILURE
SSSS	<b>0:</b> UNCERT_NONSPECIFIC, <b>1:</b> UNCERT_LASTUSABLEVALUE ou <b>4:</b> UNCERT_SENSORNOTACCURATE
SSSS	<b>0:</b> GOOD_NONSPECIFIC, <b>1:</b> GOOD_LOCALOVERRIDE ou <b>6:</b> GOOD_NONSPECIFICLOCALTIMESTAMP
LL	<b>0:</b> FREE, <b>1:</b> LOW, <b>2:</b> HIGH ou <b>3:</b> CONST

## Template JSON

**JSON** (*JavaScript Object Notation*) é um formato de troca de dados de fácil compreensão para humanos, e também simples para que programas realizem seu processamento e extração de informações. O formato **JSON** é construído sobre duas estruturas, uma coleção de pares contendo nome e valor e uma lista ordenada destes pares.

Um **Objeto** é uma lista não ordenada de pares de nomes e valores. Um objeto é delimitado por chaves, cada nome é definido dentro de aspas duplas seguido por dois pontos e o respectivo valor, e cada par é separado por vírgulas.

Um **Array** é uma lista ordenada de valores. Um *array* é delimitado por colchetes e os valores são separados por vírgulas. Valores podem ser números ou textos, que devem ser representados entre aspas duplas. Exemplo de uma mensagem no formato **JSON**:

```
{ "s":1, "t":"2014-07-11T15:26:37Z", "q":192,"c":1, "x":-1.234,"y":0.234, "z":-0.234 }
```

Para extrair o conteúdo da mensagem anterior, um possível *Template* seria o seguinte:

```
{ "s":"DUMMY", "t":"TS_TEXT(%y-%m-%dT%H:%M:%SZ)", "q":"QL_OPC", "c":"DUMMY", "x":"E3VAL", "y":"E3VAL", "z":"E3VAL" }
```

Neste caso, deve ser criado um Tag Bloco com três Elementos, e cada Elemento recebe um dos valores **x**, **y** e **z**, respectivamente.

O código a seguir contém outro exemplo de mensagem no formato **JSON**:

```
{
  "n_channels":2,
  "timestamp":1504198675,
  "hash":"1842E0F97392F08BDF996961A8333832AB06D113",
  "battery":5.13,
  "gmt":-3,
  "tag_channels":["Analog1","Analog2"],
  "value_channels":[28.100,27.200],
  "tag_units":["°C","°C"],
  "alarm_low":[0,0],
  "alarm_high":[0,0],
  "buzzer_state":0
}
```

Um possível *Template* para o exemplo anterior seria o seguinte:

```
{
  "n_channels":"DUMMY",
  "timestamp":"TS_UNIX",
  "hash":"DUMMY",
  "battery":"V1",
  "gmt":"V2",
  "tag_channels":["V3","V4"],
  "value_channels":["V5","V6"],
  "tag_units":["V7","V8"],
  "alarm_low":["V9","V10"],
  "alarm_high":["V11","V12"],
  "buzzer_state":"V13"
}
```

## Template CSV

Um **Template CSV** permite processar valores que estejam separados por ponto e vírgula. Os valores, estampas de tempo e qualidades a ser extraídos da mensagem devem ser substituídos pelas palavras-chave. Um exemplo de mensagem no formato **CSV** é o seguinte:

```
18/10/2016 22:30:45; 22BAC2300P10; "Nível Alto";98,3;m
```

Um exemplo de *Template* para este formato, para ser associado a um Tag Bloco com quatro Elementos, seria o seguinte:

```
TS_TEXT(%d/%m/%y %H:%M:%S);E3VAL;E3VAL;E3VAL;E3VAL
```

## Template XML

Um **Template XML** permite processar valores em uma mensagem MQTT que sejam enviados no formato **XML**. Deve-se declarar o formato da mensagem e substituir os campos em nós a serem transformados em valores pelas palavras-chave. Um exemplo de mensagem no formato **XML** é o seguinte:

```
<updates>
  <update>
    <id>TEMP23A</id>
    <value>34.5</value>
    <quality>true</quality>
    <timestamp>18/10/2016 22:30:45</timestamp>
  </update>
</updates>
```

Um possível *Template* para a mensagem anterior seria o seguinte:

```
<updates>
  <update>
    <id>V1</id>
    <value>V2</value>
    <quality>QL_BOOL</quality>
    <timestamp>TS_TEXT(%d/%m/%y %H:%M:%S)</timestamp>
  </update>
</updates>
```

# Configuração dos Tags

Utilize a sintaxe a seguir para a configuração de Tags no **E3** ou **Elipse Power**:

- **Dispositivo**: Não utilizado
- **Item**: Este campo deve obedecer a sintaxe a seguir

```
Topico[;Template][;Param]
```

Onde:

- **Topico**: Endereço do item, ou tópico, no banco de dados do cliente. De acordo com a norma MQTT, cada item no banco de dados de um cliente pode corresponder a um conjunto de dados livre, transmitido em formato **Texto**. Os tópicos podem estar organizados em árvore, e neste caso é utilizada uma barra para a separação de níveis. Exemplo:

```
"station12/pump01/pressure1"
"station12/pump01/pressure2"
```

Um cliente também pode utilizar o sinal numérico (#) e o sinal de mais (+) para endereçar tópicos. No primeiro caso devem ser incluídos na resposta do pedido todos os itens que contiverem no nome o item atual ou seus filhos. Por exemplo, para o valor "station12#" são retornados os seguintes valores:

```
"station12/pump01"
"station12/pump01/pressure1"
"station12/pump01/pressure2"
"station12/pump02"
"station12/pump02/pressure1"
"station12/pump02/pressure2"
```

No segundo caso devem ser incluídos na resposta do pedido todos os itens no mesmo nível. Por exemplo, para o valor "station12/pump01/+" são retornados os seguintes valores:

```
"station12/pump01/pressure1"
"station12/pump01/pressure2"
```

- **Template (opcional)**: Informa o nome de um *Template* usado como modelo de interpretação dos dados da mensagem, com o objetivo de mapear campos dentro da mensagem para Tags (estampa de tempo, qualidade e valores). Este *Template* deve estar cadastrado na aba **Templates**
- **Param (opcional)**: Informa o nome de um dos parâmetros no *Template*, caso este Tag deva receber apenas um dos parâmetros

A tabela a seguir contém exemplos de endereçamento de mensagens.

## Exemplos de endereçamento

ITEM	DESCRIÇÃO
"station12/pump01"	Neste caso a mensagem lida ou escrita para o item é um Tag do tipo <b>Texto (String)</b> sem qualquer processamento
"station12/pump01;pumpdata"	Aplica-se o <i>Template pumpdata</i> ao conteúdo de <b>station12/pump01</b> . Se o <i>Template</i> descreve mais de um valor, deve-se utilizar um Bloco em que cada Elemento é um valor do <i>Template</i>

ITEM	DESCRIÇÃO
"station12/pump01;pumpdata;V1"	Caso idêntico ao anterior, porém recuperando somente o primeiro valor do <i>Template</i> e associando-o ao Tag, que não precisa ser um Bloco. Neste caso o <i>Template</i> precisa descrever o valor <b>V1</b> . Para mais informações, consulte a aba <b>Template</b> no tópico <b>Parâmetros de Configuração do Driver</b>

Caso uma informação apareça em mais de uma mensagem e seja necessário que um Tag receba valores de todas estas mensagens, pode-se especificar um *array* de endereços no formato **JSON**, conforme a sintaxe a seguir.

```
[ "Topico1;Template1;Param1", "Topico2;Template2;Param2", "Topico3;Template3;Param3" ]
```

Por exemplo, suponha que um equipamento envie dois tipos de mensagens. A mensagem **1** contém os valores **A**, **B** e **C** e a mensagem **2** contém os valores **D**, **E**, **F** e **B** novamente, nesta ordem. Caso deseje que um Tag receba os valores de **B** vindos de qualquer das duas mensagens, configure o *array* no formato a seguir.

```
[ "device023/msg1;template1;V2", "device023/msg2;template2;V4" ]
```

Neste caso, o segundo valor configurado em **template1**, recebido no tópico **device023/msg1**, e o quarto valor configurado em **template2**, recebido no tópico **device023/msg2**, são enviados para este Tag.

#### NOTA

Pode-se repetir tópicos, como por exemplo dois ou mais Tags usando o mesmo tópico, mas um dos demais parâmetros, *template* ou *param*, deve ser diferente.

Exemplos de mensagens:

```
Tag1: "device023/msg1" - OK
Tag2: "device023/msg1;T1" - OK
Tag3: "device023/msg1;T1;V2" - OK
Tag4: ["device023/msg1;T1;V2", "device023/msg1;T1;V3"] - Não aceito. O primeiro elemento repete Tag3
```

## Documentação das Interfaces de Comunicação

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

## 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 the Eclipse MQTT Driver configuration window. It is divided into three main sections: 'Physical Layer', 'Connection management', and 'Logging Options'. In the 'Physical Layer' section, 'Serial' is selected from a dropdown menu, and 'Start driver OFFLINE' is an unchecked checkbox. The 'Timeout' is set to 1000 ms. The 'Connection management' section has a 'Mode' dropdown set to 'Automatic (managed by the driver)'. It includes three checkboxes: 'Retry failed connection every 20 seconds' (checked), 'Give up after 1 failed retries' (unchecked), and 'Disconnect if non-responsive for 0 seconds' (unchecked). The 'Logging Options' section has a 'Log to File' checkbox which is unchecked.

Aba Setup

### Opções gerais da aba Setup

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

### Opções para o grupo Connection management

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

#### Opções para o grupo Logging Options

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



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

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

## 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 **Elipse 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 **Elipse 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

 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

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

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

## IO.Stats.Partial.TimeDisconnectedSeconds

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

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

## IO.Stats.Total.BytesRecv

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

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

## IO.Stats.Total.BytesSent

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

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

## IO.Stats.Total.ConnectionCount

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

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

## IO.Stats.Total.TimeConnectedSeconds

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

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

## IO.Stats.Total.TimeDisconnectedSeconds

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

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

Qualquer valor escrito neste Tag força um *switchover* manual. Se o endereço principal de IP 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).

## Histórico de Revisões do Driver

VERSÃO	DATA	AUTOR	COMENTÁRIOS
1.0.14	04/05/2020	G. Beal	<ul style="list-style-type: none"><li>Implementada uma nova janela para edição dos <i>Templates</i> (Case 28650).</li><li>Modificada a comparação entre mensagens e <i>Templates</i> em formato <b>JSON</b> para que a ordem dos pares nome e valor dentro de um objeto não afete a correspondência (Case 28678).</li></ul>
1.0.8	05/12/2019	M. Ludwig	<ul style="list-style-type: none"><li>Adicionada uma interpretação sintática no formato <b>JSON</b> somente com nomes de atributo que sejam idênticos com <i>Template</i>. Atributos inexistentes são ignorados (Case 26316).</li></ul>
1.0.7	01/02/2019	M. Salvador	<ul style="list-style-type: none"><li>Correções de problemas de desempenho e verificação de <i>time-outs</i>.</li><li>Atualização de estampa de tempo quando o item não está associado a um <i>Template</i>.</li><li>Adicionado suporte a diversas referências para um mesmo item com <i>Templates</i> diferentes.</li></ul>
1.0.1	07/11/2018	M. Salvador	<ul style="list-style-type: none"><li>Versão inicial deste Driver.</li></ul>



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

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

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

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

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

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

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



Gartner, Cool Vendors in Brazil 2014, April 2014.

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

**Microsoft Partner**  
Gold Independent Software Vendor (ISV)