



AltusTCPIP

v2.0.1 · Português

Conteúdo

Driver ALTUS AltusTCPIP

Configuração do Driver

Parâmetros [P]

Configuração das Propriedades

Referência de Tags

Parâmetros [N]

Parâmetros [B]

Endereçamento de Tags usando o Elipse Drivers OPC

Leitura de Eventos (SOE)

Mensagens Não Solicitadas (MNS)

Threads de Read-Ahead

Otimização de Leitura

Driver Hot-Standby

Dicas de Performance

Documentação das Interfaces de Comunicação

Histórico de Revisões do Driver

Driver ALTUS AltusTCPIP

Nome do Arquivo	AltusTCPIP.dll
Fabricante	Altus Sistemas de Automação S.A.
Equipamentos	UCPs Altus conectadas às interfaces AL-3405, PO7091 e WebGates PO9900 ou PO9901
Protocolo	Alnet II via TCP/IP
Versão	2.0.1
Última Atualização	01/07/2026
Plataforma	Win32
Dependências	IOKit versão 3.0 ou superior
Leitura com Superblocos	Não
Nível	0

Introdução

O Driver ALTUS AltusTCPIP efetua a comunicação entre sistemas da **Elipse Software** e qualquer UCP da Altus Sistemas de Automação S.A. conectada à uma rede Ethernet através das interfaces AL-3405, PO7091 ou WebGates PO9900 ou PO9901. Este Driver também permite receber e tratar mensagens não solicitadas (MNS), isto é, decodificar comandos enviados através de instruções ECR ou ECH por uma aplicação que executa em um controlador programável.

Configuração do Driver

Esta seção contém informações sobre as configurações deste Driver.

Parâmetros [P]

P1	Não usado, deixe em 0 (zero)
P2	Não usado, deixe em 0 (zero)
P3	Não usado, deixe em 0 (zero)
P4	Número de canais de comunicação <i>read-ahead</i> para cada controlador programável

O parâmetro *P4* indica quantas tarefas adicionais de comunicação (*threads* de *read-ahead*) são criadas para comunicar com um controlador programável. Durante a execução, este Driver identifica quais Tags uma aplicação está lendo ao disparar leituras destes Tags nestas *threads* de *read-ahead*, de forma que o valor de um Tag já esteja disponível quando a próxima leitura de um Tag seja requisitada para este Driver.

NOTA

Recomenda-se um máximo de 4 (quatro) canais de comunicação. Para mais informações, consulte o tópico Threads de Read-Ahead.

Configuração das Propriedades

Os parâmetros de configuração para efetuar comunicação com um equipamento estão disponíveis na janela de propriedades deste Driver, mostrada na figura a seguir.

Extra Configuration

Other Settings

Time-Out (ms): 5000

☐ Handle MNS

☐ Share tags

Computer IP:

Computer Port: 405

☒ Use ALNET II

Reconnection settings...

CHESF directories settings...

LOG File

File path (client): c:\AltusTCPIP.log

File path (MNS): ItusTCP_MNS.log

File type: No LOG

☒ Log Alnet I Telegrams

☒ Log Alnet II Telegrams

Connections

0: 0405 - 195.4.32.29

Add

Remove

Edit

OK

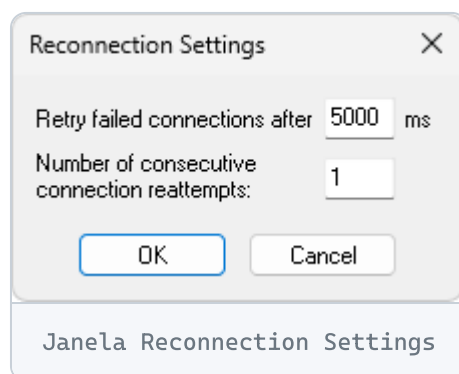
Cancel

Janela Extra Configuration

As opções disponíveis no grupo **Other Settings** estão descritas na tabela a seguir.

Opções disponíveis no grupo Other Settings

OPÇÃO	DESCRIÇÃO
Time-Out (ms)	Define o tempo, em milissegundos, que este Driver aguarda por uma resposta de um controlador programável de uma requisição realizada por este Driver
Handle MNS	Define se este Driver trata mensagens não solicitadas (MNS), mas esta opção só é habilitada quando a opção Use ALNET II está selecionada
Share tags	Se esta opção não está selecionada, este Driver cria um endereçamento de memória para cada conexão de mensagens não solicitadas (MNS). Do contrário, todas as mensagens não solicitadas são guardadas em uma mesma tabela interna. Esta opção é muito útil quando utilizada com controladores programáveis redundantes
Computer IP	Endereço IP de um computador. Esta opção só é utilizada com mensagens não solicitadas
Computer Port	Porta TCP/IP de um computador. Esta opção só é utilizada com mensagens não solicitadas
Use ALNET II	Por definição, uma rede de controladores programáveis via TCP/IP utiliza o protocolo ALNET II, mas em algumas arquiteturas pode-se comunicar usando o protocolo ALNET I via TCP/IP. Nestes casos, desmarque esta opção
Reconnection settings	Permite configurar as opções de reconexão automática, conforme a figura a seguir



As opções disponíveis na janela Reconnection Settings estão descritas na tabela a seguir.

Opções disponíveis na janela Reconnection Settings

OPÇÃO	DESCRIÇÃO
Retry failed connections after ms	Define de quanto em quanto tempo, em milissegundos, este Driver deve reiniciar as tentativas de reconexão
Number of consecutive connection reattempts	Número de retentativas de conexão sucessivas. Esta opção configura o número de operações de leitura ou escrita em Tags de um equipamento que tenha sido desconectado mas que ainda tentam estabelecer uma comunicação. Cada uma destas operações, se não é bem-sucedida na reconexão, retorna um erro após o tempo configurado. Após este número de retentativas, novas operações retornam erro imediatamente, sem uma tentativa de reconexão, até que expire o intervalo de tempo configurado. Após este tempo, a sequência de retentativas recomeça

Opções disponíveis no grupo **LOG File** estão descritas na tabela a seguir.

Opções disponíveis no grupo LOG File

OPÇÃO	DESCRIÇÃO
File path (client)	Nome de um arquivo de log onde são gravadas as comunicações com os controladores programáveis configurados no grupo Connections. Um arquivo é criado para cada <i>thread</i> de comunicação. O nome configurado nesta opção define o nome do arquivo de log da <i>thread</i> principal de cada Escravo . Cada <i>thread</i> de <i>read-ahead</i> cria um arquivo com o mesmo nome da <i>thread</i> principal mais um caractere de sublinhado e o número de ordem daquela <i>thread</i> , iniciando em 1 (um). Todas as <i>threads</i> de mesma ordem de cada Escravo compartilham o mesmo arquivo
File path (MNS)	Nome de um arquivo de log para mensagens não solicitadas (MNS)
File type	Define o tipo de log a ser criado. As opções disponíveis são No LOG : Não cria arquivos de log, Create new : Cria um arquivo de log, apagando o arquivo antigo cada vez que este Driver é reiniciado ou Append data : Um arquivo de log cresce indefinidamente, mesmo que este Driver seja reiniciado
Log Alnet I Telegrams	Escreve no arquivo de log os telegramas ALNET I
Log Alnet II Telegrams	Escreve no arquivo de log os telegramas ALNET II

As opções disponíveis no grupo **Connections** estão descritas na tabela a seguir.

Opções disponíveis no grupo Connections

OPÇÃO	DESCRIÇÃO
Add	Abre a janela Add Remote Settings, mostrada a seguir
Remove	Remove a conexão selecionada
Edit	Edita a conexão selecionada

Add Remote Settings

Main Connection:

IP address:

195 . 4 . 32 . 29

Port:

405

Time-Out:

0

ms

☐ Customize Max. Frame Size

☒ AL-3405, AL-3412 and P07091 (default option)

☐ Use appropriate limit for WebGates P09900 and P09901

☐ Customized limit:

220

 bytes

☐ Use CHESF Config (from CSV files)

CP Name:

Redundancy

☐ Use Backup IP address

Backup IP:

Port:

0

☐ Automatically connect to alternative IP address in case of comm. error

☐ Use driver hot-standby logic

Test period:

2000

ms

Monitoring operand:

M009.0

☐ Automatically activate backup PLC in case of disconnection

Set state operand:

M98

Activation value:

8

OK

Cancel

Janela Add Remote Settings

As opções disponíveis na janela Add Remote Settings estão descritas na tabela a seguir.

Opções disponíveis na janela Add Remote Settings

OPÇÃO	DESCRIÇÃO
IP address	Endereço IP do controlador programável principal
Port	Porta TCP/IP do controlador programável principal. O valor padrão desta opção é 405
Time-Out	Tempo, em milissegundos, que este Driver aguarda por uma resposta de um controlador programável principal de uma requisição realizada por este Driver
Customize Max. Frame Size	Esta opção configura o tamanho máximo da área de dados de cada <i>frame</i> de comunicação. Se alguma comunicação, normalmente uma operação de escrita ou leitura de blocos, exigir uma área de dados maior que este limite, este Driver automaticamente divide esta operação em quantas operações sejam necessárias para respeitar o limite de bytes de dados para cada <i>frame</i> isolado. As opções disponíveis são AL-3405, AL-3412 and P07091 (default option): Esta opção equivale a um <i>frame</i> com no máximo 220 bytes de dados, Use appropriate limit for WebGates PO9900 and PO9901: Esta opção limita a área de dados em 128 bytes ou Customized limit: Esta opção permite personalizar o tamanho de um <i>frame</i>. NOTA: A opção Customized limit deve ser usada apenas se absolutamente necessário, pois configurar esta opção com um valor inadequado pode ter impactos sérios em termos de perda de desempenho. Em caso de dúvida, deixe esta opção com o valor padrão ou entre em contato com o fabricante
Use Backup IP address	Selecionar esta opção permite especificar um endereço IP secundário de um controlador programável redundante. NOTA: Ao desmarcar esta opção, o endereço IP secundário configurado é perdido
Backup IP	Se a opção Use Backup IP address está selecionada, defina nesta opção o endereço IP de um controlador programável redundante

OPÇÃO	DESCRIÇÃO
Port	Porta TCP/IP do controlador programável redundante. O valor padrão desta opção é 0 (zero), ou seja, usa a mesma porta TCP/IP do controlador programável principal
Automatically connect to alternative IP address in case of comm. error	Quando esta opção está selecionada, este Driver automaticamente tenta comunicar com um endereço IP alternativo em caso de falha com o endereço IP em uso, ou seja, muda do endereço principal para o secundário e vice-versa
Use driver hot-standby logic	Quando esta opção está selecionada, este Driver gerencia automaticamente o processo de redundância (<i>Hot-Standby</i>) entre controladores programáveis. A opção Automatically connect to alternative IP address in case of comm. error deve estar selecionada para que este recurso funcione corretamente
Test period	Define um período de monitoramento do operando que informa se um controlador programável com o qual este Driver está comunicando está ativo, ou seja, a opção Monitoring Operand
Monitoring operand	Esta opção deve conter uma String informando o operando de monitoramento a ser usado. Este operando informa se um controlador programável está ativo. O valor padrão desta opção é "M009.0" e o valor de retorno é 1 (um) se um controlador programável está ativo ou 0 (zero) se um controlador programável está inativo. NOTA: Este operando pode ser definido como um Tag Bloco, mas deve necessariamente ter apenas 1 (um) Elemento

OPÇÃO	DESCRIÇÃO
Automatically activate backup PLC in case of disconnection	Esta opção só tem sentido em ser usada na lógica deste Driver em <i>Hot-Standby</i> . Se esta opção está selecionada, este Driver detecta a desconexão de um Escravo e conecta-se a um controlador programável que estava inativo e tenta ativar este controlador se o operando de monitoramento está com o valor 0 (zero, inativo). A ativação é realizada escrevendo um valor de ativação em um operando também configurável, conforme as opções a seguir. NOTA: Este recurso ainda não está padronizado no processo de redundância, e pode não estar implementado em uma aplicação. Antes de ativar, verifique se a aplicação de um controlador está preparada para este recurso e se não existem implicações em deixar a ativação em modo automático
Set state operand	Esta opção indica uma String usada para informar o operando de ativação de um controlador programável, se existir. Trata-se de um operando que permite a escrita de um valor pré-determinado para ativar um controlador programável inativo, definido na opção Activation value . NOTA: Este operando pode ser um Tag Bloco, mas deve necessariamente ter apenas 1 (um) Elemento
Activation value	Valor a ser escrito em um operando para ativar um controlador programável inativo

Além desta janela de propriedades, estas configurações também podem ser definidas em tempo de execução em aplicações **E3**, **Power** ou **Water**. Para isto, inicialize este Driver em modo **Offline**, ou seja, execute uma aplicação com a opção **Start driver OFFLINE** habilitada, configurável na aba **Setup** da janela de propriedades. As opções de configuração deste Driver estão descritas na tabela a seguir.

Opções de configuração do Driver ALTUS AltusTCPIP

ABA	PARÂMETRO	STRING OFFLINE	TIPO DE DADOS	DESCRIÇÃO
Setup	Physical Layer	IO.Type	Texto	Por padrão, utilize a opção Ethernet
	Timeout	IO.TimeoutMs	Número	Limite de tempo, em milissegundos, para recepção de dados na resposta de um equipamento. Por exemplo, o valor 1000 define um limite de 1 (um) segundo
Ethernet	Transport	IO.Ethernet.Transport	Texto	Por padrão, utilize a opção TCP
	Main IP	IO.Ethernet.MainIP	Texto	Endereço IP de um equipamento, no formato [0-255].[0-255].[0-255].[0-255]
	Port	IO.Ethernet.MainPort	Número	Por padrão, utilize o valor 405

ABA	PARÂMETRO	STRING OFFLINE	TIPO DE DADOS	DESCRIÇÃO
AltusTCPIP	MNS server TCP port	AltusTCPIP.MnsPort	Número	Por padrão, utilize o valor 0 (zero) para desabilitar mensagens não solicitadas
	Max. data bytes	AltusTCPIP.MaxData	Número	Por padrão, utilize o valor 230. Use o valor 128 para compatibilidade com equipamentos WebGate. Os limites válidos variam entre 4 (quatro, mínimo útil) e 230 (máximo do protocolo)
	Direct mode	AltusTCPIP.Direct	Número	Utilize o valor 0 (zero) para desabilitar e um valor diferente de 0 (zero) para habilitar. Quando esta opção está habilitada, este Driver envia <i>frames</i> ALNET I brutos sobre TCP/IP, sem o envoltório ALNET II, para dispositivos ou <i>gateways</i> que transmitem <i>frames</i> ALNET I nativos

Todas as propriedades no modo **Offline** devem ser configuradas via Tags PLC no formato **String** usando o parâmetro *N1* igual a -1 (menos um), o parâmetro *N2* igual a 0 (zero), o parâmetro *N3* igual a 0 (zero) e o parâmetro *N4* igual a 3 (três). Para mais detalhes e exemplos, consulte o tópico Documentação das Interfaces de Comunicação.

Referência de Tags

Esta seção contém informações sobre a configuração dos Tags **[N/B]** deste Driver.

Parâmetros [N]

Utilize um Tag PLC para execução de comandos comuns, utilizando os parâmetros descritos na tabela a seguir.

N1	Deixe em 0 (zero) para mensagens solicitadas ou em 1000 para mensagens não solicitadas (MNS)
N2	Tipo de dados. Para mais informações, consulte a tabela a seguir
N3	Endereço. Para mais informações, consulte a tabela a seguir
N4	Bit ou posição de tabela. Para mais informações, consulte a tabela a seguir

O parâmetro *N1* contém o número da conexão na lista de controladores programáveis, o grupo Connections na janela de propriedades, começando em 0 (zero). Para mensagens não solicitadas (MNS), use o valor 1000 + IPPLC. A tabela a seguir contém a lista de comandos comuns suportados por este Driver.

Tipos de Tags PLC de dados

TIPO DE TAG	N1	N2	N3	N4	MODO	COMENTÁRIOS
Memória	Deixe em 0 (zero) para mensagens solicitadas ou em 1000 para mensagens não solicitadas	0 (zero)	Endereço	(*)	Leitura ou escrita	
Tabela de memória		1	Endereço	Posição	Leitura ou escrita	
Bit de memória		2	Endereço	n	Leitura ou escrita	Em desuso
Decimal		3	Endereço	(*)	Leitura ou escrita	
Tabela de decimais		4	Endereço	Posição	Leitura ou escrita	
Auxiliar		5	Endereço	(*)	Leitura ou escrita	
Entradas e saídas		6	Endereço	(*)	Leitura ou escrita	
Inteiro		7	Endereço	(*)	Leitura ou escrita	
Bit em octeto (entrada e saída)		8	Endereço	Bit, entre 0 (zero) e 7 (sete)	Somente escrita	Em desuso

TIPO DE TAG	N1	N2	N3	N4	MODO	COMENTÁRIOS
Bit em auxiliar		9	Endereço	Bit, entre 0 (zero) e 7 (sete)	Somente escrita	Em desuso
Real (float)		10	Endereço	(*)	Leitura ou escrita	
Tabela de reais (float)		11	Endereço	Posição	Leitura ou escrita	
Tabela de inteiros		12	Endereço	Posição	Leitura ou escrita	
Captura de eventos		13	Endereço TM	Aux × 10 + bit	Somente leitura	
Evento		14	Módulo, entre 0 (zero) e 15	Ponto, entre 0 (zero) e 31	Somente leitura	Requer um Tag com o parâmetro N2 igual a 13
Erro de módulo		15	0 (zero)	0 (zero)	Somente leitura	Requer um Tag com o parâmetro N2 igual a 13
Overflow de memória		16	0 (zero)	0 (zero)	Somente leitura	Requer um Tag com o parâmetro N2 igual a 13
Mudança de data e hora		17	0 (zero)	0 (zero)	Somente leitura	Requer um Tag com o parâmetro N2 igual a 13

NOTAS

- Adicione o valor 10000 ao parâmetro *N2* para tratar os valores lidos como inteiros de 16 bits sem sinal, portanto os valores variam entre 0 (zero) e 65535.
- Os parâmetros *N4* marcados com asterisco (*) na tabela anterior indicam que, se este parâmetro é igual a 0 (zero), todo o operando é lido. Se este parâmetro é maior que 0 (zero), um comando é executado sobre o bit especificado neste parâmetro. Os valores variam entre 1 (um) e 32, dependendo do tamanho do operando, em que o bit menos significativo é indicado pelo valor 1 (um).
- Quando o valor do parâmetro *N1* está configurado para mensagens não solicitadas, ou seja, igual a 1000, os valores são inicializados e permanecem em 0 (zero) até que um controlador programável atualize os operandos correspondentes.

Parâmetros [B]

Utilize um Tag Bloco para execução de comandos comuns, utilizando os parâmetros descritos na tabela a seguir.

B1	Deixe em 0 (zero) para mensagens solicitadas ou em 1000 para mensagens não solicitadas (MNS)
B2	Tipo de dados. Para mais informações, consulte a tabela a seguir
B3	Endereço. Para mais informações, consulte a tabela a seguir
B4	Não utilizado, deixe em 0 (zero)

A tabela a seguir contém a lista de comandos comuns suportados por este Driver.

Tipos de Tags Bloco de dados

TIPO DE TAG	B1	B2	B3	MODO
Tabela de memória	Deixe em 0 (zero)	Número na tabela	Posição inicial na tabela	Leitura ou escrita
Tabela de decimais	Deixe em 0 (zero)	1000 + número na tabela	Posição inicial na tabela	Leitura ou escrita
Tabela de reais (float)	Deixe em 0 (zero)	4000 + número na tabela	Posição inicial na tabela	Leitura ou escrita
Tabela de inteiros	Deixe em 0 (zero)	6000 + número na tabela	Posição inicial na tabela	Leitura ou escrita
Tabela de memória de MNS	1000	Número na tabela	Posição inicial na tabela	Leitura ou escrita
Tabela de decimais de MNS	1000	1000 + número na tabela	Posição inicial na tabela	Leitura ou escrita
Tabela de reais (float) de MNS	1000	4000 + número na tabela	Posição inicial na tabela	Leitura ou escrita
Status barramentos de entrada e saída	Deixe em 0 (zero)	2000	Número do barramento	Leitura ou escrita
Status comunicação	Deixe em 0 (zero)	2001	Reset de variáveis (*)	Leitura ou escrita
Status equipamento	Deixe em 0 (zero)	2002	-	Leitura ou escrita
Bloco de memória	Deixe em 0 (zero)	3000	Endereço inicial	Leitura ou escrita
Bloco de real (float)	Deixe em 0 (zero)	5000	Endereço inicial	Leitura ou escrita
Bloco de entradas	Deixe em 0 (zero)	5001	Endereço inicial	Leitura ou escrita

TIPO DE TAG	B1	B2	B3	MODO
Bloco de saídas	Deixe em 0 (zero)	5001	Endereço inicial	Somente escrita
Bloco de auxiliares	Deixe em 0 (zero)	5002	Endereço inicial	Leitura ou escrita
Bloco de inteiros	Deixe em 0 (zero)	5003	Endereço inicial	Leitura ou escrita
Bloco de decimais	Deixe em 0 (zero)	5004	Endereço inicial	Leitura ou escrita

NOTAS

- Adicione o valor 10000 ao parâmetro *B2* para tratar os valores lidos como inteiros de 16 bits sem sinal, portanto os valores variam entre 0 (zero) e 65535.
- Quando o valor do parâmetro *B1* está configurado para mensagens não solicitadas, ou seja, igual a 1000, os valores são inicializados e permanecem em 0 (zero) até que um controlador programável atualize os operandos correspondentes.
- Ao reiniciar as variáveis, os valores válidos são **0**: Os valores dos dados sobre recepção e transmissão não são alterados ou **1**: Os valores dos dados sobre recepção e transmissão são zerados.

Endereçamento de Tags usando o Elipse Drivers OPC

Caso este Driver seja usado com o **Elipse Drivers OPC**, utilize a tabela a seguir para acessar as memórias de um controlador programável.

Mnemônicos para uso do Elipse Drivers OPC

MNEMÔNICO	DESCRIÇÃO	MODO
<con:> A x	Auxiliar	Leitura ou escrita
<con:> A x.b	Bit de auxiliar	Leitura ou escrita
<con:> A x_n	Bloco de auxiliares	Leitura ou escrita
<con:> D x	Decimal	Leitura ou escrita
<con:> D x.b	Bit de decimal	Leitura ou escrita
<con:> D x_n	Bloco de decimais	Leitura ou escrita
<con:> E x	Entrada	Somente leitura
<con:> E x.b	Bit de entrada	Somente leitura
<con:> E x_n	Bloco de entradas	Somente leitura
<con:> F x	Real (ponto flutuante	Leitura ou escrita
<con:> F x_n	Bloco de reais (ponto flutuante	Leitura ou escrita
<con:> I x	Inteiro de 32 bits	Leitura ou escrita
<con:> I x.b	Bit de inteiro de 32 bits	Leitura ou escrita
<con:> I x_n	Bloco de inteiros de 32 bits	Leitura ou escrita
<con:> M x	Memória	Leitura ou escrita
<con:> M x.b	Bit de memória	Leitura ou escrita
<con:> M x_n	Bloco de memórias	Leitura ou escrita
<con:> S x	Saída	Somente escrita
<con:> S x.b	Bit de saída	Somente escrita
<con:> S x_n	Bloco de saídas	Somente escrita

MNEMÔNICO	DESCRIÇÃO	MODO
IP plc: TDUM x.y_n	Tabela de decimais de mensagens não solicitadas	Somente leitura
<con:> TD x.y	Posição da tabela de decimais	Leitura ou escrita
<con:> TD x.y_n	Tabela de decimais	Leitura ou escrita
IP plc: TFUM x.y_n	Tabela de reais de mensagens não solicitadas	Somente leitura
<con:> TF x.y	Posição da tabela de reais (ponto flutuante)	Leitura ou escrita
<con:> TF x.y_n	Tabela de reais (ponto flutuante)	Leitura ou escrita
<con:> TI x.y	Posição da tabela de inteiros de 32 bits	Leitura ou escrita
<con:> TI x.y_n	Tabela de inteiros de 32 bits	Leitura ou escrita
IP plc: TMUM x.y_n	Tabela de memórias de mensagens não solicitadas	Somente leitura
<con:> TM x.y	Posição da tabela de memórias	Leitura ou escrita
<con:> TM x.y_n	Tabela de memórias	Leitura ou escrita
<con:> BS x_n	Status do barramento de entrada e saída	Somente leitura
<con:> CS x	Status da comunicação	Somente leitura
<con:> ES	Status de um equipamento	Somente leitura

Em que:

- Caracteres em negrito representam mnemônicos fixos que identificam comandos. Caracteres que não estão em negrito representam valores que devem ser substituídos
- O valor <con:> é um prefixo opcional que define o número de uma conexão na lista de controladores programáveis, começando em 0 (zero). Se este prefixo não é informado, este Driver utiliza a conexão 0 (zero). Consulte os exemplos a seguir para mais informações

- O valor *plc*, no caso de mensagens não solicitadas, corresponde ao número da conexão, ou valor IPPLC usando em Tags PLC ou Bloco, e coincide com último octeto do endereço IP de um controlador programável. Portanto, para um endereço IP igual a 195.4.32.29, o valor de *plc* é igual a 29. O valor *x* representa o endereço de um operando ou o número de uma tabela, dependendo do tipo de dados solicitado. Para os tipos CS e BS, consulte a tabela Tipos de Tags Bloco de dados e as respectivas notas
- O valor *y* representa a posição inicial em uma tabela
- O valor *n* representa o total de itens a serem lidos
- O valor *b* representa um bit, entre 0 (zero, bit menos significativo) e 31 (bit mais significativo). Entretanto, este número máximo depende do número de bits presentes no tipo de dados

A tabela a seguir contém alguns exemplos de mnemônicos.

Exemplos de mnemônicos

MNEMÔNICO	DESCRIÇÃO
A20	Auxiliar 20 e conexão 0 (zero)
1:M2400	Memória 2400 e conexão 1 (um)
D10_4	Bloco de decimais 10 com 4 (quatro) elementos e conexão 0 (zero)
E5.2	Bit 2 (dois) da entrada 5 (cinco) e conexão 0 (zero)
TM17.0_20	21 posições da tabela 17 de memória a partir da posição 0 (zero) e conexão 0 (zero)
IP29:TFUM3.1_30	30 posições da tabela 3 (três) de reais de mensagens não solicitadas a partir da posição 1 (um) do controlador programável com o octeto final do endereço IP igual a 29
1:M2400	Memória 2400 e conexão 1 (um)

Leitura de Eventos (SOE)

Este Driver segue a nota de aplicação NAE-040 da Altus Sistemas de Automação S.A., seção 2.3.2, no que se refere ao tratamento de eventos. Para isto são necessários os seguintes procedimentos:

1. Crie um Tag de captura de eventos, ou seja, com o parâmetro *N2* igual a 13, com um ciclo de varredura recomendado de 1000 ms. O parâmetro *N3* define qual tabela de memória de 63 posições possui os eventos, pois o *ladder* deve gravar os eventos nesta tabela. O parâmetro *N4* especifica o número do Auxiliar e o bit que é reiniciado após a leitura dos eventos. Da mesma forma, o *ladder* deve configurar este bit auxiliar para 1 (um) quando há eventos. É importante configurar corretamente os parâmetros *N3* e *N4* para um correto funcionamento da captura dos eventos. A utilização deste Tag é obrigatório para a leitura de eventos. A propriedade **Value** deste Tag contém o número de eventos encontrados na tabela de memória durante a última leitura.
2. Após a criação e leitura do Tag criado anteriormente, este Driver começa a guardar os eventos internamente. Para a leitura destes eventos, é necessário criar um Tag especificando o número da conexão, o parâmetro *N1*, o número do módulo ou placa, o parâmetro *N3*, e o número do ponto, o parâmetro *N4*. Os valores válidos são **0**: Estado 0 (zero) da entrada ou **1**: Estado 1 (um) da entrada.

Deve-se, portanto, criar um Tag para cada bit de cada módulo do qual se deseja ler os eventos. Cada evento traz na propriedade **TimeStamp** a data e a hora que um evento ocorreu em um controlador programável. Desta forma, é desejável que um controlador programável esteja com a data e hora ajustada com a data e hora do computador.

Mensagens Não Solicitadas (MNS)

O uso de mensagens não solicitadas permite melhorar significativamente a performance de uma aplicação, pois não há necessidade de realizar uma operação de *polling* nas variáveis de um controlador programável constantemente, mesmo quando estas variáveis não sofrem alteração. Ao

contrário, usando mensagens não solicitadas, sempre que uma variável sofre alteração, o próprio controlador programável envia uma mensagem avisando este Driver que, por sua vez, executa a alteração na base de dados.

Uma aplicação, por sua vez, deve configurar os Tags PLC e Bloco de forma muito semelhante à configuração de Tags normais, por *polling*, mudando apenas o parâmetro *N1* ou *B1* para o valor 1000. Este número indica a este Driver que os valores destes Tags devem ser lidos não de um controlador programável, mas diretamente da base de dados deste Driver, atualizada pelas mensagens não solicitadas. Os Tags de uma aplicação, portanto, continuam funcionando por *polling*, exceto que a leitura é realizada diretamente de uma memória imagem contida neste Driver, que não gera comunicação com um controlador programável. Como esta operação é extremamente rápida, sugere-se usar um período de varredura (*scan*) muito pequeno, da ordem de milissegundos, o que permite uma atualização imediata dos Tags em caso de alteração de valor por mensagens não solicitadas. A tabela a seguir contém um exemplo de parâmetros para mensagens não solicitadas. O valor de varredura (*scan*) deve ser configurado em 10 milissegundos.

N1	1000, leitura da memória imagem
N2	0 (zero, memória)
N3	1 (um, memória M1)
N4	Não usado, deixe em 0 (zero)

Threads de Read-Ahead

Para habilitar uma *thread* de *read-ahead*, configure o parâmetro *P4* com o número de *threads* que este Driver deve criar para cada controlador programável, além da *thread* principal de comunicação. Portanto, se o valor do parâmetro *P4* é igual a 5 (cinco), este Driver cria 6 (seis) conexões com um controlador programável, ou seja, uma *thread* principal e 5 (cinco) *threads* de *read-ahead*. Configure o parâmetro *P4* com o valor 0 (zero) para que este Driver crie apenas uma *thread*, a principal, que é o modo normal de funcionamento deste Driver.

NOTA

Dimensione corretamente o parâmetro *P4*, pois o número de conexões não pode ser maior que a capacidade de um controlador programável de tratar múltiplas requisições. Recomenda-se que o parâmetro *P4* seja configurado com um valor 3 (três) ou 4 (quatro), no máximo. Versões mais recentes de *firmware* de controladores programáveis podem suportar até 9 (nove) conexões.

Otimização de Leitura

Este Driver suporta agrupamento automático de Tags Bloco, quando suportado pelo programa que usa este Driver, o que permite ganhos de performance de monitoramento em aplicações que não definem Tags Bloco. Este agrupamento é realizado automaticamente por uma aplicação na medida que os Tags entram em varredura (*scan*), permitindo minimizar requisições de leitura de um controlador programável.

NOTA

Atualmente esta otimização está disponível somente no **Elipse Drivers OPC** versão **1.1** ou superior.

Quando este Driver é utilizado com uma aplicação compatível com esta otimização, não é necessário criar Tags Bloco, apenas tenha cuidado em definir Tags em endereços consecutivos em um controlador programável. O critério para justificar a criação de Tags Bloco neste caso é apenas a necessidade da escrita simultânea em vários endereços. Inclusive, recomenda-se criar Tags PLC se não há necessidade de escritas em bloco.

Driver Hot-Standby

A lógica de redundância **Hot-Standby** implementada neste Driver segue as especificações da Altus Sistemas de Automação S.A. Considera-se que existe um controlador principal e um reserva para cada conexão definida na janela de propriedades deste Driver, cada controlador com um endereço IP específico. Em um dado momento, sempre há um controlador ativo e um inativo. Em condições normais, este Driver está normalmente conectado e comunicando com o controlador ativo, ao mesmo tempo em que monitora, a intervalos regulares, o operando de monitoramento. O intervalo de tempo entre as leituras do operando de monitoramento é chamado de período de teste, ou Test period. O operando de monitoramento é usado para indicar se um controlador está ativo ou inativo. O valor 1 (um) indica que um controlador está ativo e o valor 0 (zero) indica que um controlador está inativo.

De acordo com esta lógica, se o operando de monitoramento vai do valor 1 (um) para o valor 0 (zero), indicando que o controlador passou para a condição de inativo, este Driver deve se desconectar do endereço IP antes ativo e conectar-se ao endereço IP alternativo, realizando então a leitura do operando de monitoramento neste novo endereço IP. Se este operando está configurado com o valor 1 (um), o controlador alternativo está ativo e a comunicação pode prosseguir com este novo controlador. Se este operando também está com o valor 0 (zero) neste novo endereço IP, este Driver novamente se desconecta e tenta conectar ao endereço IP anterior. Este processo continua até que este Driver tenha sucesso na conexão com algum dos controladores que retorne o valor 1 (um) na leitura do operando de monitoramento.

A opção Automatically activate backup PLC in case of disconnection na janela de propriedades permite a este Driver forçar a ativação de um controlador inativo em caso de perda de conexão com o controlador ativo. Se esta opção está selecionada, este Driver tenta se conectar ao controlador inativo ao perder a conexão com o controlador ativo. Após a conexão, é realizada a leitura do operador de monitoramento do controlador inativo. Se este operando retorna o valor 0 (zero), isto indica que este controlador continua no estado inativo, portanto este Driver escreve um valor pré-configurado, definido na opção Activation value, em um operando também pré-configurado, definido na opção Set state operand. Se o programa residente do controlador está preparado para interpretar este valor neste operando como um comando de ativação, o controlador passa para a

condição de ativo, mudando o valor do operando de monitoramento para 1 (um). Este recurso, no entanto, não é padronizado, e depende da implementação no programa residente do controlador para funcionar corretamente.

Dicas de Performance

Existem algumas configurações a serem realizadas no conjunto Driver-Controlador que podem gerar um ganho de performance:

- Equipamentos vêm de fábrica configurados com um número de retentativas na camada TCP/IP igual a 5 (cinco) e cada retentativa possui um *time-out* que começa em 200 ms e, a cada nova tentativa, este valor dobra. Na prática, concluiu-se que o número de retentativas ótimo é igual a 3 (três), resultando em um *time-out* total de 3 (três) segundos. Se ao final deste tempo, somando ao tempo de resposta de um controlador, não é recebida nenhuma respostas, isto significa que uma requisição não chegou a um controlador, ou que um controlador não pôde responder por qualquer motivo. Portanto, o *time-out* configurado neste Driver deve ser maior que este valor
- Ainda em relação a equipamentos, durante o desenvolvimento da programação *ladder*, é interessante utilizar um operando chamado **LAI**, que permite que um controlador tenha um tempo extra para responder requisições deste Driver, o que permite que um controlador responda mais de uma requisição entre ciclos de programa. O número máximo de operando que podem ser adicionados é 7 (sete), dependendo do ciclo de programação de um controlador. Para mais informações sobre estas configurações, consulte o documento **Características Técnicas do Módulo AL3412**
- Em relação a este Driver, configure as *threads* de comunicação, ou seja, quantas conexões deste Driver são estabelecidas para comunicação instantânea. O valor normalmente utilizado gira em torno de 3 (três) a 4 (quatro) *threads*. O uso de um valor maior não garante um performance melhor na comunicação, podendo inclusive resultar em perda de performance. Estes valores foram definidos como ideais em testes realizados
- O valor de *time-out* deste Driver deve ser configurado para um valor maior que o tempo máximo de resposta de um controlador. Para a configuração indicada anteriormente, este tempo corresponde a 3 (três) segundo de *time-out* da camada TCP/IP somado com o tempo de resposta de um controlador. Um valor indicado em testes é de 4 (quatro) segundos

Documentação das Interfaces de Comunicação

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

Esta documentação é compartilhada e mantida no manual do IOKit:

- Configurações de um Driver
- Configurações Gerais
- Configuração de Estatísticas
- Configuração da Interface Ethernet
- Configuração da Interface Modem
- Configuração da Interface RAS
- Configuração da Interface Serial

Histórico de Revisões do Driver

VERSÃO	DATA	AUTOR	COMENTÁRIOS
2.0.1	01/07/2026	M. Ludwig	• Driver atualizado para a biblioteca IOKit versão 3 e Visual Studio 2022 (<i>Case 17154</i>).
1.13.1	26/01/2006	A. Corrêa A. Quites	• Implementado o compartilhamento para mensagens não solicitadas (<i>Case 6126</i>). • Implementada a leitura de eventos (<i>Case 6143</i>). • Corrigido um problema na leitura de operandos com o parâmetro <i>N4</i> diferente de zero (<i>Case 6127</i>). • Corrigido um problema na comunicação com controladores programáveis antigos (<i>Case 6613</i>).
1.12.1	05/07/2005	A. Quites	• Implementada um lógica completa de redundância, ou <i>Hot-Standby</i> (<i>Case 4344</i>). • Melhorada a rotina de importação de configurações de versões antigas, que agora contempla todas as versões anteriores deste Driver (<i>Case 5477</i>). • Criadas opções de configuração para o processo de reconexão automática deste Driver (<i>Case 5759</i>). • Melhorado o desempenho da escrita de bits para os operandos Auxiliar, Octeto e Memória, que antes usavam mascaramento, para usar o recurso de subdivisão do protocolo (<i>Case 5833</i>). • Substituído o comando 129 (<i>Força Operando Simples</i>) pelo comando 133 (<i>Escreve Operando Simples</i>) na escrita de operandos simples, conforme orientação da Altus (<i>Case 5832</i>).
1.11.1	11/04/2005	L. Silva	• Adicionado suporte a otimização de leituras, ou agrupamento automático de Tags em Blocos (<i>Case 5521</i>). • Adicionado suporte a leituras síncronas reais mesmo quando usando <i>threads</i> de <i>read-ahead</i> (<i>Case 4977</i>).
1.10.1	31/03/2005	A. Quites	• Corrigida a importação das configurações extras de versões anteriores deste Driver. A importação continua não funcionando para Drivers publicados antes de junho de 2002 (<i>Case 5477</i>).

VERSÃO	DATA	AUTOR	COMENTÁRIOS
1.9.1	07/03/2005	L. Silva	• Melhorada a lógica de <i>read-head</i> para Tags com tempo de varredura (<i>scan</i>) alto, ou seja, maior que 10 segundos (<i>Case 4969</i>). • Corrigida a lógica da troca de endereço IP, manual ou automática, quando são usados os canais de <i>read-head</i> (<i>Case 5089</i>). • Criada uma opção para desabilitar a troca de endereço IP automática quando é usada redundância de controladores programáveis (<i>Case 5089</i>).
1.8.1	08/11/2004	A. Corrêa A. Quites	• Pequenas modificações apenas na documentação, sem alteração neste Driver. • Adicionada uma opção que desabilita a conversão para o protocolo ALNET II, tornando este Driver um Driver ALNET I comum, portanto desabilitando o recurso de mensagens não solicitadas (<i>Case 3190</i>). • Este Driver agora não retorna mais erro no procedimento StartCom caso ocorra um erro na abertura do arquivo de log (<i>Case 3323</i>). • Adicionadas leituras de bloco para os operandos Auxiliares, Decimais, Entradas e Saídas (<i>Cases 4455 e 3932</i>). • Implementado o operando I (inteiro) (<i>Case 4456</i>). • Adicionado suporte ao Eclipse Drivers OPC (<i>Case 4457</i>). • Removida a opção Log Date & Time. • Adicionado o número da conexão na janela de edição (<i>Case 4458</i>). • Este Driver agora permite escrever blocos de operandos de qualquer tamanho, mesmo maiores que o tamanho máximo permitido pelo protocolo, situação em que este Driver particiona a escrita em mais de um bloco (<i>Case 4754</i>). • A escrita de blocos de mais de 64 operandos agora funciona devido a uma alteração no código de conversão ALNET I para ALNET II, que antes não estava convertendo para a instrução ALNET II mais adequada (<i>Case 4751</i>). • Implementada a redundância de conexões (<i>Case 4344</i>).
1.7.1	09/01/2004	F. Englert	• Corrigido um GPF (<i>General Protection Failure</i>) que ocorria na segunda vez que uma aplicação executava se as mensagens não solicitadas estavam habilitadas (<i>Case 3193</i>). • O padrão deste Driver agora é não gerar logs (<i>Case 3193</i>).

VERSÃO	DATA	AUTOR	COMENTÁRIOS
1.5.1	31/10/2003	F. Englert	• Habilitado o suporte a mensagens não solicitadas (MNS) (<i>Case 2805</i>). • Resolvido um problema na memória <i>cache</i> de pré-leitura, em que este Driver não descartava valores antigos de um Tag Bloco se algum Elemento era escrito (<i>Case 2838</i>).
1.4.1	29/09/2003	F. Englert	• Melhorado o log de mensagens NACK recebidas pelo protocolo.
1.3.1	26/09/2003	F. Englert	• Este Driver agora converte Strings para números nas escritas. Ao escrever um valor "1234", o número 1234 é enviado. Se a String contém algum caractere não numérico ou um número mal-formatado, a escrita retorna erro.
1.2.1	05/09/2003	F. Englert	• Criada uma rotina para minimizar descarte de dados <i>read-ahead</i>. Um Tag só é lido quando falta menos de 8 (oito) segundos para a próxima varredura (<i>scan</i>).
1.1.1	02/09/2003	F. Englert	• Criadas <i>threads</i> de <i>read-ahead</i> e retentativas de leitura e escrita.
1.0.1		R. Haetinger	• Todas as publicações anteriores ao controle de revisões.