
Manual de Utilização Driver DNP Escravo para Remota Hadron **AL-2741**

Rev. B 09/2004
Cód. Doc.: MU204844



altus

Nenhuma parte deste documento pode ser copiada ou reproduzida sem o consentimento prévio e por escrito da Altus Sistemas de Informática S.A., que se reserva o direito de efetuar alterações sem prévio comunicado.

Conforme o Código de Defesa do Consumidor vigente no Brasil, informamos a seguir, aos clientes que utilizam nossos produtos, aspectos relacionados com a segurança de pessoas e instalações.

Os equipamentos de automação industrial fabricados pela Altus são robustos e confiáveis devido ao rígido controle de qualidade a que são submetidos. No entanto, equipamentos eletrônicos de controle industrial (controladores programáveis, comandos numéricos, etc.) podem causar danos às máquinas ou processos por eles controlados em caso de defeito em suas partes e peças ou de erros de programação ou instalação, podendo inclusive colocar em risco vidas humanas.

O usuário deve analisar as possíveis conseqüências destes defeitos e providenciar instalações adicionais externas de segurança que, em caso de necessidade, sirvam para preservar a segurança do sistema, principalmente nos casos da instalação inicial e de testes.

É imprescindível a leitura completa dos manuais e/ou características técnicas do produto antes da instalação ou utilização do mesmo.

A Altus garante os seus equipamentos conforme descrito nas Condições Gerais de Fornecimento, anexada às propostas comerciais.

A Altus garante que seus equipamentos funcionam de acordo com as descrições contidas explicitamente em seus manuais e/ou características técnicas, não garantindo a satisfação de algum tipo particular de aplicação dos equipamentos.

A Altus desconsiderará qualquer outra garantia, direta ou implícita, principalmente quando se tratar de fornecimento de terceiros.

Pedidos de informações adicionais sobre o fornecimento e/ou características dos equipamentos e serviços Altus devem ser feitos por escrito. A Altus não se responsabiliza por informações fornecidas sobre seus equipamentos sem registro formal.

DIREITOS AUTORAIS

Série Ponto, MasterTool, Quark, ALNET e WebPlc são marcas registradas da Altus Sistemas de Informática S.A.

IBM é marca registrada da International Business Machines Corporation.

Sumário

1. INTRODUÇÃO.....	1
Remota HD3001	1
Aplicações da UTR Hadron	1
Documentos Relacionados a este Manual.....	2
Visual.....	2
Suporte Técnico.....	2
Mensagens de Advertência Utilizadas neste Manual	2
2. VISÃO GERAL SOBRE DNP V3.00	4
Convenções e Exemplo de Topologia de Comunicação.....	4
Mensagens Não Solicitadas	5
Objetos de Dados DNP.....	5
Cabeçalho de um Objeto (Object Header): Object, Variation, Qualifier e Range.....	5
Tipos de Dados DNP	6
Grupos de Objetos DNP.....	6
Binary Input	7
Binary Output	7
Counter.....	7
Analog Input.....	7
Analog Output.....	7
Time.....	7
Class.....	7
Funções DNP 3.0	7
Read (Function Code 1)	7
Write (Function Code 2)	7
Select (Function Code 3)	8
Operate (Function Code 4).....	8
Direct Operate (Function Code 5)	8
Direct Operate - No Acknowledgement (Function Code 6)	8
Immediate Freeze (Function Code 7).....	8
Freeze and Clear (Function Code 9).....	8
Freeze and Clear - No Acknowledgement (Function Code 10).....	8
Cold Restart (Function Code 13)	8
Warm Restart (Function Code 14)	9
Enable Spontaneous Messages (Function Code 20).....	9
Disable Spontaneous Messages (Function Code 21).....	9
Assign Class (Function Code 22).....	9
Delay Measurement (Function Code 23)	9
Descrição Detalhada dos Objetos	9
Single-Bit Binary Input.....	9
Binary Input with Status.....	10
Binary Input Change without Time.....	10
Binary Input Change with Time.....	11
Binary Output	11
Binary Output Status	11
Control Relay Output Block	12
32-Bit Binary Counter.....	13
16-Bit Binary Counter.....	14

32-Bit Counter without Flag	14
16-Bit Counter without Flag	15
32-Bit Frozen Counter	15
16-Bit Frozen Counter	15
32-Bit Frozen Counter without Flag	16
16-Bit Frozen Counter without Flag	16
32-Bit Counter Change Event without Time	16
16-Bit Counter Change Event without Time	16
32-Bit Counter Change Event with Time	17
16-Bit Counter Change Event with Time	17
32-Bit Frozen Counter Event without Time	17
16-Bit Frozen Counter Event without Time	18
32-Bit Frozen Counter Event with Time	18
16-Bit Frozen Counter Event with Time	18
16-Bit Analog Input.....	18
16-Bit Analog Input without Flag	19
16-Bit Analog Input Change Event without Time.....	19
16-Bit Analog Input Change Event with Time	20
16-Bit Analog Output Status.....	20
16-Bit Analog Output Block.....	20
Time and Date	21
Time Delay Fine	21
Class 0 Data	21
Class 1 Data	21
Class 2 Data	22
Class 3 Data	22
Regras para Numeração (Endereçamento) de Pontos	22
3. INTEROPERABILIDADE DNP	25
Perfil do Driver AL-2741	25
Tabela de Implementação para o AL-2741	28
4. CONFIGURAÇÃO	32
Telas do ProHadron.....	32
Tela de Dados Gerais do Driver (Driver General Data)	33
Guia UTR (<i>RTU</i>)	33
Tipo do driver (Driver Type).....	33
Posição no Barramento 0 do AL-2005 (AL-2005 Bus 0 Position).....	33
Guia Camada Física (<i>Physical Layer</i>)	33
Porta Serial do AL-2005 (AL-2005 Serial Port)	33
Taxa de Bits do Canal (Baudrate).....	34
Paridade (Parity)	34
Bits Adicionais Finais do Quadro Físico (Stop Bits).....	34
Sinais de Modem (Handshake - Modem Signals)	34
Tempo Mínimo para Transmissão (Transmission Delay).....	34
Guia Camada de Enlace (<i>Data Link Layer</i>)	34
Endereço de Enlace DNP (DNP Link Address)	35
Modo de Confirmação de Enlace (Link Confirm Mode).....	35
Timeout para Confirmação do Enlace (Link Confirm Timeout).....	35
Número Máximo de Retentativas de Enlace (Link Maximum Retries)	35
Guia Camada de Aplicação (<i>Application Layer</i>).....	36
Timeout de Confirmação da Aplicação (Application Confirm Timeout).....	36

Intervalo entre Requisições para Sincronismo (Delay between Sync Requests)	37
Intervalo Máximo para Indicar Fora de Sincronismo (Maximun Delay between Syncs)	37
Tamanho dos Fragmentos de Resposta (Response Fragment Size)	37
Guia de Variação Default dos Objetos (<i>Object Default Variation</i>)	37
Variação Default para Objetos 01 (Default Variation Object 01)	38
Variação Default para Objetos 02 (Default Variation Object 02)	38
Variação Default para Objetos 20 (Default Variation Object 20)	38
Variação Default para Objetos 21 (Default Variation Object 21)	38
Variação Default para Objetos 22 (Default Variation Object 22)	39
Variação Default para Objetos 23 (Default Variation Object 23)	39
Variação Default para Objetos 30 (Default Variation Object 30)	39
Variação Default para Objetos 32 (Default Variation Object 32)	39
Variação Default para Objetos 40 (Default Variation Object 40)	39
Guia de Envio por Exceção (<i>Report By Exception - RBE</i>)	40
Período de Varredura de Entradas Binárias (RBE Binary Input Scan Period)	40
Período de Varredura de Contadores (RBE Running Counter Scan Period)	41
Período de Varredura de Contadores Congelados (RBE Frozen Counter Scan Period)	41
Período de Varredura de Entradas Analógicas (RBE Analog Input Scan Period)	41
Guia de Respostas Não Solicitadas (<i>Unsolicited Responses</i>)	42
Habilitação de Respostas Não Solicitadas (Unsolicited Responses Permitted)	42
Endereço Destino para Respostas Não Solicitadas (Unsolicited Responses Destination Address)	42
Mínimo de Eventos para Transmissão da Classe 1 (Unsolicited Minimum Events for Class 1)	42
Mínimo de Eventos para Transmissão da Classe 2 (Unsolicited Minimum Events for Class 2)	43
Mínimo de Eventos para Transmissão da Classe 3 (Unsolicited Minimum Events for Class 3)	43
Período Mínimo para Transmissão da Classe 1 (Unsolicited Notification Delay for Class 1)	43
Período Mínimo para Transmissão da Classe 2 (Unsolicited Notification Delay for Class 2)	43
Período Mínimo para Transmissão da Classe 3 (Unsolicited Notification Delay for Class 3)	43
Período entre Retentativas de Mensagens Não Solicitadas (Unsolicited Retry Delay)	43
Número de Retentativas de Mensagens Não Solicitadas (Unsolicited Maximum Retries) ..	43
Tela de Pontos de E/S da Remota (I/O Points)	44
Coluna Grupo de Objetos DNP (DNP Object Grouping)	45
Coluna Índice DNP (DNP Index)	45
Coluna Classe de Eventos (Event Class)	45
Coluna Método para Geração de Eventos (Event Method)	45
Coluna Banda Morta (Dead Band)	46
Tela de Pontos de E/S de IED (IED I/O Points)	46
Coluna Grupo de Objetos DNP (DNP Object Grouping)	47
Coluna Índice DNP (DNP Index)	47
Coluna Classe de Eventos (Event Class)	47
Coluna Método para Geração de Eventos (Event Method)	48
Coluna Banda Morta (Dead Band)	48
5. DIAGNÓSTICOS	49
Tabela de Configuração	49
Tabela de Diagnósticos	52

6. GLOSSÁRIO 54

1. Introdução

O driver AL-2741 implementa o protocolo DNP escravo para utilização na remota Hadron (HD3001). Ele serve com a interface entre um centro de operação ou um sistema SCADA e a remota. É executado no coprocessador AL-2005. Uma remota pode executar até 4 drivers AL-2741, cada um num AL-2005 exclusivo, ou seja, cada um deles deve possuir apenas uma aplicação carregada.

Implementa o protocolo DNP 3.00 nível 2 (DNP-L2) e algumas características além deste nível. O “Device Profile” e a “Implementation Table” são apresentadas em capítulo posterior neste manual. Na seção abaixo apresenta-se uma descrição geral sobre a remota Hadron. Maiores detalhes podem ser encontrados no Manual de Utilização da Remota Hadron, HD3001, Cód. Doc.: MU208301.

Remota HD3001

A Unidade Terminal Remota HD3001, ou simplesmente UTR HD3001, pertencente à Série Hadron, é uma moderna solução para supervisão e controle de sistemas de geração, transmissão e distribuição de energia elétrica. É um sistema modular, com configuração flexível, composto de múltiplos processadores, várias opções de módulos de entrada e saída e de protocolos de comunicação.

Entre as suas características, destacam-se:

- Multiprocessadores com sistema operacional multitarefa
- Protocolo DNP escravo
- Comunicação com até 4 centros de operação
- Base de dados com configuração individual para cada centro de operação
- Protocolos mestre para comunicação com IEDs (Intelligent Electronic Devices), tais como MODBUS RTU, PROFIBUS DP e DNP
- Configuração amigável de todos os parâmetros via software para sistemas operacionais Windows
- Possibilidade de execução de intertravamentos e lógicas de controle
- Sincronização através de receptores GPS (Global Positioning System) ou através de Centros de Operação e Sistemas de Supervisão via protocolo
- Duas portas de comunicação serial incorporadas na remota para realizar configuração, manutenção, interface com IHM local ou ajuste de horário por receptor GPS
- Porta Ethernet TCP/IP com nível de aplicação ALNET II através de módulo de interface opcional
- Aquisição e tratamento de entradas digitais e analógicas com várias opções de interfaces elétricas para os sinais de campo
- Registro de eventos de variações em entradas digitais com data e horário associado com resolução de 1 ms
- Comandos digitais com ou sem seleção - SBO (Select Before Operate)
- Comandos digitais com ou sem e verificação de hardware - CBO (Check Before Operate)
- Comandos analógicos
- Entradas analógicas de 16 bits com calibração e parametrização digital



Aplicações da UTR Hadron

- Sistemas de geração de energia elétrica
- Sistemas de distribuição de energia elétrica
- Supervisão e controle de energia elétrica
- Aquisição de dados e registro de eventos

- Intertravamentos de segurança

Documentos Relacionados a este Manual

Para obter informações adicionais sobre a Série Hadron podem ser consultados outros documentos (manuais e características técnicas) além deste. Estes documentos encontram-se disponíveis em sua última revisão em www.altus.com.br.

Cada produto possui um documento denominado Característica Técnica (CT), onde encontram-se as características do produto em questão. Adicionalmente o produto pode possuir Manuais de Utilização (o código do manuais são citados na CT).

Por exemplo, o módulo AL-3138 tem todas as informações de características de utilização e de compra na sua CT. Por outro lado, o AL-3202 possui, além da CT, um manual de utilização.

Aconselha-se os seguintes documentos como fonte de informação adicional:

- Características Técnicas de Cada Produto
- MasterTool Programming Manual de Programação
- Manual de Utilização da Remota Hadron, HD3001

Visual

Antes de proceder à instalação é recomendável fazer uma inspeção visual cuidadosa dos equipamentos, verificando se não há danos causados pelo transporte. Verifique se todos os componentes de seu pedido estão em perfeito estado. Em caso de defeitos, informe a companhia transportadora e o representante ou distribuidor Altus mais próximo.

CUIDADO:

Antes de retirar os módulos da embalagem, é importante descarregar eventuais potenciais estáticos acumulados no corpo. Para isso, toque (com as mãos nuas) em uma superfície metálica aterrada qualquer antes de manipular os módulos. Tal procedimento garante que os níveis de eletricidade estática suportados pelo módulo não serão ultrapassados.

É importante registrar o número de série de cada equipamento recebido, bem como as revisões de software, caso existentes. Essas informações serão necessárias na eventualidade de contatar o Suporte Técnico da Altus.

Suporte Técnico

Para entrar em contato com o Suporte Técnico da Altus em São Leopoldo, RS, ligue para +55 51 589-9500. Para conhecer os centros de Suporte Técnico da Altus existentes em outras localidades, consulte nosso site (www.altus.com.br) ou envie um email para altus@altus.com.br.

Se o equipamento já estiver instalado, tenha em mãos as seguintes informações ao solicitar assistência:

- os modelos dos equipamentos utilizados e a configuração do sistema instalado;
- o número de série da UCP;
- a revisão do equipamento e a versão do software executivo, constantes na etiqueta afixada na lateral do produto;
- informações sobre o modo de operação da UCP, obtidas através do programador MasterTool.
- o conteúdo do programa aplicativo (módulos), obtido através do programador MasterTool;
- a versão do programador utilizado.

Mensagens de Advertência Utilizadas neste Manual

Neste manual, as mensagens de advertência apresentarão os seguintes formatos e significados:

PERIGO:

Relatam causas potenciais, que se não observadas, *levam* a danos à integridade física e saúde, patrimônio, meio ambiente e perda da produção.

CUIDADO:

Relatam detalhes de configuração, aplicação e instalação que *devem* ser seguidos para evitar condições que possam levar a falha do sistema e suas conseqüências relacionadas.

ATENÇÃO:

Indicam detalhes importantes de configuração, aplicação ou instalação para obtenção da máxima performance operacional do sistema.

2. Visão Geral sobre DNP V3.00

Este capítulo apresenta alguns conceitos sobre o protocolo DNP3.00 para auxiliar o entendimento dos capítulos seguintes. Numa seção específica serão apresentados os objetos implementados na remota. Por questão de precisão da informação, algumas informações são apresentadas em inglês, tal como apresentadas nas próprias normas que definem o protocolo.

Convenções e Exemplo de Topologia de Comunicação

Os termos abaixo possuem os seguintes significados neste manual:

- SCADA: o mestre num barramento DNP 3.00.
- UTR (Unidade Terminal Remota, ou simplesmente remota): um equipamento escravo num barramento DNP e, eventualmente, mestre em outros barramentos.
- IED (Intelligent Electronic Device): um equipamento escravo num barramento com uma remota como mestre.

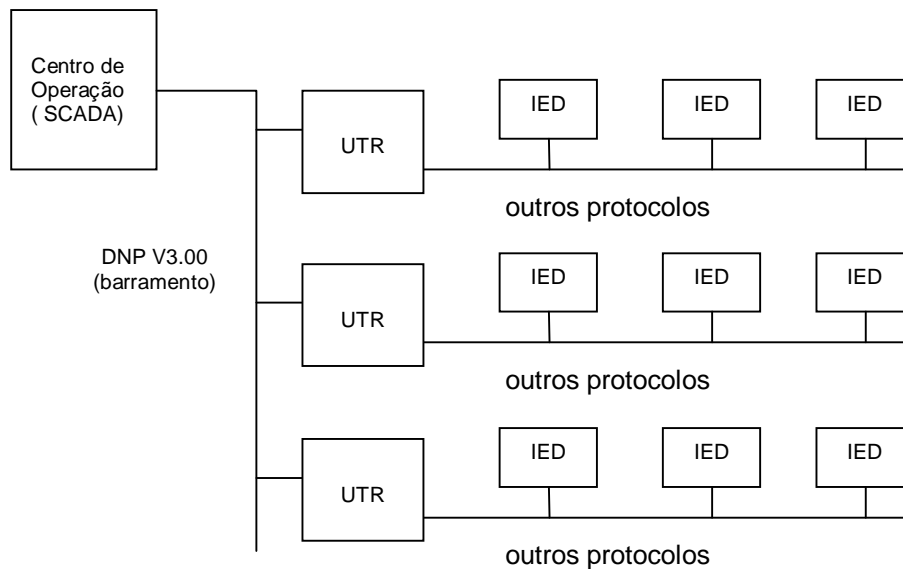


Figura 2-1. Exemplo de Topologia de Comunicação

A figura acima mostra um barramento que conecta um Centro de Operação ou um sistema SCADA a várias remotas, usando o protocolo DNP.

O protocolo DNP permite uma comunicação ponto-multiponto num barramento com um mestre (SCADA) e vários escravos (UTR). Cada estação ou equipamento ligado ao barramento DNP possui um endereço DNP único de 16 bits, que deve estar na faixa de 0 até 65519. Os endereços 65520 até 65535 estão reservados para mensagens em broadcast do mestre para todos os escravos.

Uma UTR pode ainda ser mestre de outra rede (ou outras redes) para comunicação com IED. Estas outras redes podem também utilizar o protocolo DNP ou outros protocolos como Modbus, por exemplo.

Os pontos configurados (solicitados ou enviados) no SCADA podem estar presentes nos próprios módulos de entrada e saída da remota ou então nos IED, que são escravos na remota.

Mensagens Não Solicitadas

Mensagens não solicitadas podem ser usadas por drivers escravos para enviar dados ao mestre, sem que tenha havido uma requisição do mestre. Este recurso é importante para obter tempos de resposta curtos e otimização da utilização da largura de banda disponível.

O driver AL-2741 permite a utilização de mensagens não solicitadas, mas esta opção deve estar normalmente desabilitada. O driver AL-2741 não implementa detecção e prevenção de colisões na comunicação. Portanto, somente quando o meio físico fornece recursos para contornar problemas de colisão é que está opção deve ser utilizada.

Alguns exemplos de quando esta opção pode ser utilizada são dadas a seguir:

- Comunicação ponto-a-ponto com enlaces full duplex com somente um mestre e um escravo.
- Ethernet TCP/IP gerencia colisões. Embora a remota Hadron não implemente o protocolo DNP com interface Ethernet, existem conversores entre meio físico serial RS232 e Ethernet que podem ser utilizados para encapsular os frames DNP 3.00 serial em TCP/IP sobre Ethernet.

Objetos de Dados DNP

Os dispositivos que utilizam o protocolo DNP são capazes de monitorar, controlar e produzir uma grande quantidade de informação, tanto em software como em hardware. Estas informações são chamadas de objetos de dados (**data objects**); são processadas e armazenadas e podem ser empacotadas para transmissão ou recepção como unidades de dados da camada de aplicação. Seja qual for o dispositivo ou o fabricante, todos que se comunicam através do protocolo DNP devem fornecer e receber dados com os mesmos formatos.

Os formatos para os objetos de dados implementados na remota Hadron serão descritas neste documento

A camada de aplicação do DNP possui dois campos de 8 bits chamados de **object** e de **variation**, utilizados para identificar os objetos empacotados. O campo **object** define um objeto de dados genérico, tal como uma informação binária (**static binary data**). Já o campo **variation** define diferentes representações para um mesmo objeto de dados genérico, como o tamanho do objeto e se ele possui ou não **flags** que complementam a informação sobre um determinado objeto.

Cabeçalho de um Objeto (Object Header): Object, Variation, Qualifier e Range

A camada de aplicação DNP tem capacidade de transferir vários objetos em uma única mensagem. No campo **object header** é especificado um conjunto de objetos de dados. Ele possui três campos, a saber:

Object specification: campo de 16 bits com 8 bits para **object** e 8 bits para **variation** (ver seção anterior).

Qualifier: campo de 8 bits que define o significado do campo **range**, quando ele existir.

Range: é um campo de comprimento variável cujo formato depende do campo **qualifier**. Ele pode especificar, por exemplo, o **start index** e o **stop index** de um conjunto de objetos, o número de objetos a partir do índice 0 ou a lista de índices de objetos que não estão em seqüência. Para maiores detalhes sobre o endereçamento de pontos no DNP ver seção “Regras para Numeração (Endereçamento) de Pontos”.

Um ou mais **Object Header** podem estar contidos numa única mensagem.

Os seguintes valores (hexadecimal) podem definir o campo **qualifier** numa mensagem do driver AL-2741.

00: O campo **field** contém dois índices de 8 bits (**start index** e **stop index**) que definem o primeiro e o último objeto do tipo especificado.

- 01: O campo **field** contém dois índices de 16 bits (**start index** e **stop index**) que definem o primeiro e o último objeto do tipo especificado.
- 06: O campo **field** não está presente. Todos os objetos do tipo especificado possuem uma identificação própria do seu endereço.
- 07: O campo **field** contém um valor de 8 bits (q = 1 até 255) que informa a quantidade de objetos. Também fica definido que o índice do primeiro objeto é 0 e o índice do último objeto é “q-1”.
- 08: O campo **field** contém um valor de 16 bits (q = 1 até 65535) que informa a quantidade de objetos. Também fica definido que o índice do primeiro objeto é 0 e o índice do último objeto é “q-1”.
- 17: O campo **field** contém um valor de 8 bits (q = 1 até 255) que informa a quantidade de pares índice/objeto que seguem este campo. Também fica definido que cada par índice/objeto que segue este campo contém o seu próprio índice de 8 bits e um objeto do tipo especificado.
- 28: O campo **field** contém um valor de 16 bits (q = 1 até 65535) que informa a quantidade de pares índice/objeto que seguem este campo. Também fica definido que cada par índice/objeto que segue este campo contém o seu próprio índice de 16 bits e um objeto do tipo especificado.

Como visto acima, o campo **qualifier** permite grande flexibilidade na montagem de um frame DNP com uma inerente complexidade, de forma que as opções para este campo são limitadas. A tabela abaixo expõem o subconjunto de qualificadores utilizados pela remota Hadron bem como o uso de cada um.

Qualifier (hex)	Uso em Requisições	Uso em Respostas
00, 01	Uma faixa de pontos estáticos ou um único ponto com um apontador.	Objetos estáticos.
06	Todos os pontos.	Não utilizado.
07, 08	Uma quantidade limitada de eventos. Um único ponto sem valor (p. ex. apenas tempo e data)	Um único ponto sem valor (p. ex. apenas tempo e data)
17, 28	Controles (normalmente um ou mais pontos sem relação um com outro).	Objetos de eventos (normalmente um ou mais pontos sem relação um com outro).

Table 2-1 Códigos do campo Qualifier com seus respectivos usos.

Tipos de Dados DNP

Os dados DNP são classificados em quatro categorias, a saber:

Static Objects: objetos que refletem o valor atual de um determinado ponto.

Event Objects: objetos cujo valor é atualizado como resultado de uma mudança no valor de algum ponto. Portanto, são registros históricos que refletem o valor do ponto em algum instante anterior ao atual.

Frozen Static Objects: objetos que refletem o valor atual congelado de algum ponto. Os valores dos pontos são congelados como resultado de uma requisição de congelamento para aquele ponto.

Frozen Event Objects: objetos cujo valor é atualizado como resultado de uma mudança em um ponto congelado. Portanto, são registros históricos que refletem o valor do ponto congelado em algum instante anterior ao atual.

Grupos de Objetos DNP

Alguns grupos de objetos são previstos pelo protocolo DNP. Cada grupo pode conter vários objetos. E cada endereço de objeto pode referenciar uma série de **variation**.

Para cada grupo a norma DNP reservou uma faixa de códigos de objetos. Os grupos e as respectivas faixas de códigos de objetos são as seguintes:

Binary Input

Este grupo contém todos os objetos que representam informações de entrada com dois estados (binária ou booleano). Os códigos de objetos 1 até 9 são reservados para estes objetos.

Binary Output

Este grupo contém todos os objetos que representam informações de saída com dois estados ou então saídas a relé. Os códigos de objetos 10 até 19 são reservados para estes objetos.

Counter

Este grupo contém todos os objetos que representam valores acumulados ao longo do tempo. Os códigos de objetos 20 até 29 são reservados para estes objetos.

Analog Input

Este grupo contém todos os objetos que representam informações de entradas analógicas. Os códigos de objetos 30 até 39 são reservados para estes objetos.

Analog Output

Este grupo contém todos os objetos que representam informações de saídas analógicas. Os códigos de objetos 40 até 49 são reservados para estes objetos.

Time

Este grupo contém todos os objetos que representam o tempo de forma absoluta ou relativa com qualquer resolução. Os códigos de objetos 50 até 59 são reservados para estes objetos.

Class

Este grupo contém todos os objetos que representam classes de dados ou dados prioritários. Os códigos de objetos 60 até 69 são reservados para estes objetos. Classe 0 representa todos os objetos estáticos. Classes 1, 2 e 3 representam todos os eventos de objetos de dados.

Funções DNP 3.0

A camada de aplicação DNP 3.0 fornece um conjunto de funções para realizar operações com dados ou na própria remota.

Read (Function Code 1)

Esta função é utilizada para requisitar objetos de dados de um equipamento. Os campos **object header** (**object**, **qualifier** e **range**) são codificados de tal maneira que o tamanho dos campos com objetos que os sucedem no frame pode ser calculados, permitindo a requisição de múltiplos objetos ou faixas numa única mensagem. O número de múltiplas requisições permitidas numa única mensagem é definida no **device profile** de cada equipamento no qual o DNP esteja implementado.

Write (Function Code 2)

Esta função é utilizada para enviar objetos de uma estação mestre para um equipamento escravo. Os campos **object header** são definidos da mesma maneira que na função **read**.

Select (Function Code 3)

Esta função é utilizada para selecionar um ou mais pontos de controle num equipamento, que podem ser saídas a relé ou saídas analógicas. Esta função não tem a capacidade de enviar um valor para uma saída ou alterar o seu estado, mas sim de prepará-la ou armá-la, se for o caso, e reportar o seu estado. Uma outra mensagem de operação deve ser posteriormente enviada para o equipamento para efetivar a alteração na saída. A mensagem de operação enviada posteriormente deve referenciar o mesmo objeto que a mensagem de seleção que a precedeu para que a alteração na saída seja efetivada. A mensagem de seleção provoca o disparo de um temporizador regressivo no equipamento. A mensagem de operação deve ser recebida corretamente antes que a contagem do temporizador termine para que o comando seja realmente executado.

Operate (Function Code 4)

Esta função é utilizada acionar um ou mais pontos de saídas a relé ou saídas analógicas num equipamento. Uma mensagem anterior de seleção referenciando o mesmo objeto deve ter sido recebida pelo equipamento, e o temporizador de seleção deve ainda estar ativo antes que a saída seja acionada.

Direct Operate (Function Code 5)

Esta função é utilizada para ativar uma ou mais saídas ou setpoints no equipamento sem a necessidade de ser precedido por uma mensagem de seleção.

Direct Operate - No Acknowledgement (Function Code 6)

Esta função é utilizada para ativar uma ou mais saídas ou setpoints no equipamento sem a necessidade de ser precedido por uma mensagem de seleção. Além disto o equipamento não responderá a estação mestre com uma mensagem (no nível da camada de aplicação).

Immediate Freeze (Function Code 7)

Esta função é utilizada para copiar o objeto de dado especificado para uma memória de congelamento. Ao receber esta mensagem, o equipamento copia os valores atuais dos objetos referenciados para as suas memória apropriadas. Os objetos que foram congelados podem ser requisitados em outra mensagem específica para tal função.

Freeze and Clear (Function Code 9)

Esta função é utilizada para copiar o objeto de dado especificado para uma memória de congelamento, tal qual a função **Immediate Freeze**. A diferença é que além disto o equipamento também realiza o zeramento o objeto de dado especificado.

Freeze and Clear - No Acknowledgement (Function Code 10)

Esta função realiza o mesmo procedimento anterior (**Freeze and Clear**) com a diferença de que o equipamento não envia nenhuma resposta (no nível da camada de aplicação). Esta função é usada tipicamente para realizar um congelamento e zeramento global em todos os equipamentos sob supervisão do mestre (endereçamento em broadcasting).

Cold Restart (Function Code 13)

Esta função provoca uma reinicialização completa da aplicação do usuário, como se ela tivesse sido energizada pela primeira vez.

Warm Restart (Function Code 14)

Esta função provoca uma reinicialização parcial do equipamento, ou seja, ele não passa por todos os passos da seqüência de inicialização (somente parte da aplicação é reinicializada). A aplicação DNP pode reiniciar-se sem reinicializar outros subsistemas e processos. Tipicamente, esta função realiza uma inicialização da configuração do equipamento e apaga os eventos armazenados na sua memória local.

Enable Spontaneous Messages (Function Code 20)

Esta função é utilizada pelo mestre para informar ao equipamento para habilitar o envio espontâneo dos objetos especificados no **object header (unsolicited responses)**

Disable Spontaneous Messages (Function Code 21)

Esta função é utilizada pelo mestre para informar ao equipamento para desabilitar o envio espontâneo dos objetos especificados no **object header (unsolicited responses)**

Assign Class (Function Code 22)

Esta função é utilizada para atribuir objetos de dados a determinadas classes a partir da estação mestre.

Normalmente, uma classe é configurada para cada objeto de dados na remota Hadron na sua inicialização, mas esta classe pode ser alterada posteriormente pela estação mestre.

Por exemplo, se o mestre altera a classe de alguns objetos para classe 0, não serão mais gerados eventos para estes objetos.

Delay Measurement (Function Code 23)

Esta função é utilizada para calcular o atraso da comunicação para um equipamento em particular. Ela geralmente é utilizada no processo de sincronização de tempo dos equipamentos.

Descrição Detalhada dos Objetos

Nesta seção todos os objetos de dados são descritos. Como é uma seção apenas de referência à norma DNP, o texto foi retirado da própria norma e foi mantido no original em inglês por uma questão de fidelidade da informação tal qual é definida na norma.

Single-Bit Binary Input

Data Object: 01

Variation: 01

Type: Static

Grouping: Binary Input

The *single-bit binary input* object is used to represent the state of a digital input point (hardware or software).

A single bit is used to store this data object.

This variation contains no point status information. For example, *on-line*, *restart*, etc. bits which are part of the *binary input with status* variation, are not part of this variation. The use of the *single-bit binary input* variation implies that the point is on-line and all other status bits are clear. (i.e. *restart*, *communication lost*, etc. bits are cleared).

Binary Input with Status

Data Object: 01

Variation: 02

Type: Static

Grouping: Binary Input

The *binary input with status* object is used to represent the state of a digital input point (hardware or software), and also indicates the status of the point.

An 8-bit field is used to store this data object, as follows:

The *on-line* bit (0) indicates that the binary input point has been read successfully. If this field is set to *off-line*, the state of the digital point may not be correct.

The *restart* bit (1) indicates that the field device that originated the data object is currently restarting. This device may be the device reporting this data object.

The *communication lost* bit (2) indicates that the device reporting this data object has lost communication with the originator of the data object.

The *remote forced data* bit (3) indicates that the state of the binary input has been forced to its current state at a device other than the end device.

The *local forced data* bit (4) indicates that the state of the binary input has been forced to its current state at the end device.

The *chatter filter* (5) bit indicates that the binary input point has been filtered in order to remove unneeded transitions in the state of the point.

The *state* bit (7) indicates the current state of the binary input point.

Binary Input Change without Time

Data Object: 02

Variation: 01

Type: Event

Grouping: Binary Input

The *binary input change without time* object is used to represent the changed state of a digital input point (hardware or software) and also indicates the status of the point.

An 8-bit field is used to store this data object, as follows:

The *on-line* bit (0) indicates that the binary input point has been read successfully. If this field is set to *off-line*, the state of the digital point may not be correct.

The *restart* bit (1) indicates that the field device that originated the data object has been re-started. This device may be the device reporting this data object.

The *communication lost* bit (2) indicates that the device reporting this data object has lost communication with the originator of the data object.

The *remote forced data* bit (3) indicates that the state of the binary input has been forced to its current state at the originating device.

The *local forced data* bit (4) indicates that the state of the binary input has been forced to its current state at the device reporting this data object.

The *chatter filter* bit (5) indicates that the binary input point has been filtered in order to remove unneeded transitions in the state of the point.

The *state* bit (7) indicates the current changed state of the binary input point.

This object is only reported when the current value is different than the last recorded or measured value. If the chatter filter is on, this object may only be reported when the new state has remained constant for a certain period of time.

Binary Input Change with Time

Data Object: 02

Variation: 02

Type: Event

Grouping: Binary Input

The *binary input change with time* object is used to represent the changed state of a digital input point (hardware or software) and the time at which the state changed. It also indicates the status of the point.

An 8-bit field is used to store the state and status of the data object, as follows:

The *on-line* bit (0) indicates that the binary input point has been read successfully. If this field is set to *off-line*, the state of the digital point may not be correct.

The *restart* bit (1) indicates that the field device that originated the data object has been re-started. This device may be the device reporting this data object.

The *communication lost* bit (2) indicates that the device reporting this data object has lost communication with the originator of the data object.

The *remote forced data* bit (3) indicates that the state of the binary input has been forced to its current state at the originating device.

The *local forced data* bit (4) indicates that the state of the binary input has been forced to its current state at the device reporting this data object.

The *chatter filter* bit (5) indicates that the binary input point has been filtered in order to remove unneeded transitions in the state of the point.

The *state* bit (7) indicates the current changed state of the binary input point.

An additional 48-bit field is used to store the time. This field indicates the absolute time at which the telecontrol device detected the change of state. The accuracy of this time will depend on the accuracy of the individual device. *Time of occurrence* is recorded as milliseconds since midnight, January 1st, 1970, at zero hours, zero minutes, seconds, and milliseconds.

Binary Output

Data Object: 10

Variation: 01

Type: Static

Grouping: Binary Output

The *binary output* object is used to control a digital output point (hardware or software). The *state* bit indicates the desired logic state of the digital output point.

A 1-bit field is used to store this data object.

Binary Output Status

Data Object: 10

Variation: 02

Type: Static

Grouping: Binary Output

The *binary output status* object is used to indicate the current state of a controlled digital point and the status of that point.

An 8-bit field is used to store this data object, as follows:

The *on-line* bit (0) indicates that the device controlling the binary output point is operating correctly. A binary output command to this point should work correctly. If this field were set to *off-line*, a binary output command to this point would be unsuccessful.

The *restart* bit (1) indicates that the field device that originated the data object has been re-started. This device may be the device reporting this data object.

The *communication lost* bit (2) indicates that the digital output point could not be controlled because communications have been lost with the controlled device.

The *remote forced data* bit (3) indicates that the digital output point has been controlled at the originating device and the current state is in the *state* bit.

The *local forced data* bit (4) indicates that the digital output point has been controlled at this device and the current state is in the *state* bit.

The *state* bit (7) indicates the current state of the binary output point.

Control Relay Output Block

Data Object: 12

Variation: 01

Type: Static

Grouping: Binary Output

The *control relay output block information* object contains 4 digital output control parameters fields, and 1 status field.

The 4 parameter fields define the type and duration of the digital output.

The *control code* field (8 bits) indicates the control function to perform. The applicability of this code will depend on the type of hardware used in the end device.

The *count* field (8 bits) indicates the number of times that the control operation should be performed in succession (0 ... 255 repetitions). If the *count* is 0, the control will not be executed. Hadron RTU will consider any value bigger than 0 as 1.

The *on-time* field (32 bits) specifies the amount of time the digital output is to be turned on, and may not apply to all control types (0... 2^{32} -1 ms).

The *off-time* field (32 bits) specifies the amount of time the digital output is to be turned off, and may not apply to all control types (0... 2^{32} -1 ms). Hadron RTU will ignore this field. It will assume the *off-time* field to be 0.

The *control code* field (8 bits) contains 4 sub-fields:

- *Trip/Close (bits 6 and 7)*: This sub-field determines which control relay to activate in a system where a trip and close relay pair is used to energize and de-energize the field points. The *NUL* value in this field can be used to activate the field point select relay only, without activating the trip or close relays. In a system without field point select relays, the *NUL* value would not perform any control operation. In a system without trip/close relays, this field should always be *NUL* to indicate a normal digital control operation where the exact control point is inherently known. This field does not support having both the trip and close attributes simultaneously, as this is an illegal operation for the system. Sub-field's valid values are:

00 = NUL
 01 = Close
 10 = Trip

- *Code (bits 0 to 3)*: The control relay output block data object can be used with devices which support control queuing on a point per point basis or devices which have other control mechanisms. In the former, any control command should be queued for the particular point in question. In the latter, each control is performed until completion before the next control is accepted for that point. The meaning of the *code* sub-field and the operation to perform is determined by the following:
 - 0: NUL operation. No operation specified. May be used to read the output state (bit 7 of status field).
 - 1: Pulse On. The point is turned on for the specified *on-time*, turned off for the specified *off-time* and left in the *off* state.
 - 2: Pulse Off. The point is turned off for the specified *off-time*, then turned on for the specified *on-time* and left in the *on* state. Hadron RTU will return the code 4 in *status field*, if this bit is set.
 - 3: Latch On. This latches the point on.
 - 4: Latch Off. This latches the point off.
 - 5 - 15: Undefined.
- *Queue (bit 4)*: If the *code* sub-field is NUL then no control operation is queued, and the queue is cleared of all controls including the presently running control if the *clear* attribute is set. When the *control function* is executed and completed, it is removed from the queue. If the control block for that operation had the *queue* attribute set, the control operation is re-queued (to the back of the queue) for that point. Hadron RTU will return the code 4 in *status field*, if this bit is set.
- *Clear (bit 5)*: If the control operation has the *clear* attribute set, all control operations are removed from the queue including the presently running control, and this new control operation is activated and performed. Hadron RTU will return the code 4 in *status field*, if this bit is set.

The *status* field (7-bits) indicate the success or failure of the requested control operation. The meaning of this field is determined as follows:

- 0: Request accepted, initiated, or queued.
- 1: Request not accepted as the *operate* message was received after the *arm* timer timed out. The *arm* timer was started when the *select* operation for the same point was received.
- 2: No previous matching *select* message (i.e. an *operate* message was sent to activate a control point that was not previously armed with the *select* message.)
- 3: Request not accepted as there were formatting errors in the *control* request (either *select*, *operate*, or *direct operate*).
- 4: Control operation not supported for this point.
- 5: Request not accepted, as the control queue is full or the point is already active.
- 6: Request not accepted because of control hardware problems.
- 7 - 127: Undefined.

32-Bit Binary Counter

Data Object: 20

Variation: 01

Type: Static

Grouping: Counter

The *32-bit binary counter* represents an accumulated value. This can be accumulated pulses or transitions from a hardware or software point.

The *value* field (32 bits) shows the current value of the counter at the time of reporting or last reported value from the originating device. This value is incremented indefinitely until a counter clear operation is performed in which case the value is reset to 0.

The *flag* field has 8 bits, as follows:

The *on-line* bit (0) indicates that the counter point has been read successfully. If this field is set to *off-line*, the state of the counter point may not be correct.

The *restart* bit (1) indicates that the field device that originated the data object has been re-started. This device may be the device reporting this data object.

The *communication lost* bit (2) indicates that the device reporting this data object has lost communication with the originator of the data object.

The *remote forced data* bit (3) indicates that the state of the counter has been forced to its current state at the originating device.

The *local forced data* bit (4) indicates that the state of the counter has been forced to its current state at the device reporting this data object.

When set, the *roll-over* bit (5) indicates that the accumulated value has exceeded the maximum possible recordable value ($2^{32}-1$). The counter value has been reset to 0 upon roll-over, and counting has resumed as normal. This bit is cleared when the counter value (plus the *roll-over* state) is reported.

Quando o contador é associado com um ponto de entrada binário, o *remote forced data bit flag* do contador terá o mesmo estado do flag do ponto binário correspondente.

16-Bit Binary Counter

Data Object: 20

Variation: 02

Type: Static

Grouping: Counter

The *16-bit binary counter* represents an accumulated value. This can be accumulated pulses or transitions from a hardware or software point.

The *value* field (16 bits) shows the current value of the counter at the time of reporting or last reported value from the originating device. This value is incremented indefinitely until a counter clear operation is performed in which case the value is reset to 0.

The *flag* field (8 bits) has the same meaning as in previous counter objects, with the following difference:

When set, the *roll-over* bit (5) indicates that the accumulated value has exceeded the maximum possible recordable value ($2^{16}-1$). The counter value has been reset to 0 upon roll-over, and counting has resumed as normal. This bit is cleared when the counter value (plus the *roll-over* state) is reported.

32-Bit Counter without Flag

Data Object: 20

Variation: 05

Type: Static

Grouping: Counter

The *32-bit binary counter* represents an accumulated value. This can be accumulated pulses or transitions from a hardware or software point.

The *value* field (32 bits) shows the current value of the counter at the time of reporting or last reported value from the originating device. This value is incremented indefinitely until a counter clear operation is performed in which case the value is reset to 0.

The use of this variation implies that the input point is on-line and that all other flag bits are normal (i.e. this variation implies that flag = 1).

16-Bit Counter without Flag**Data Object: 20****Variation: 06****Type: Static****Grouping: Counter**

The *16-bit binary counter* represents an accumulated value. This can be accumulated pulses or transitions from a hardware or software point.

The *value* field (16 bits) shows the current value of the counter at the time of reporting or last reported value from the originating device. This value is incremented indefinitely until a counter clear operation is performed in which case the value is reset to 0.

The use of this variation implies that the input point is on-line and that all other flag bits are normal (i.e. this variation implies that flag = 1).

32-Bit Frozen Counter**Data Object: 21****Variation: 01****Type: Frozen Static****Grouping: Counter**

The *32-bit frozen counter* is a compound information object that contains information about a counter that was previously frozen. The counter accumulates pulses or transitions from a hardware or software point.

The *frozen value* field (32 bits) shows the value of the counter when the last counter freeze was performed at the originating device.

The *flag* field (8 bits) has the same meaning as in previous counter objects.

16-Bit Frozen Counter**Data Object: 21****Variation: 02****Type: Frozen Static****Grouping: Counter**

The *16-bit frozen counter* is a compound information object that contains information about a counter that was previously frozen. The counter accumulates pulses or transitions from a hardware or software point.

The *frozen value* field (16 bits) shows the value of the counter when the last counter freeze was performed at the originating device.

The *flag* field (8 bits) has the same meaning as in previous counter objects.

32-Bit Frozen Counter without Flag

Data Object: 21

Variation: 09

Type: Frozen Static

Grouping: Counter

The *32-bit frozen counter* is a compound information object, which contains information about a counter, which was previously frozen. The counter accumulates pulses or transitions from a hardware or software point.

The *frozen value* field (32 bits) shows the value of the counter when the last counter freeze was performed at the originating device.

The use of this variation implies that the input point is on-line and that all other flag bits are normal (i.e. this variation implies that flag = 1).

16-Bit Frozen Counter without Flag

Data Object: 21

Variation: 10

Type: Frozen Static

Grouping: Counter

The *16-bit frozen counter* is a compound information object, which contains information about a counter, which was previously frozen. The counter accumulates pulses or transitions from a hardware or software point.

The *frozen value* field (16 bits) shows the value of the counter when the last counter freeze was performed at the originating device.

The use of this variation implies that the input point is on-line and that all other flag bits are normal (i.e. this variation implies that flag = 1).

32-Bit Counter Change Event without Time

Data Object: 22

Variation: 01

Type: Event

Grouping: Counter

The *32-bit counter change event without time* represents a counter value that has exceeded a configured dead band. This can be accumulated pulses or transitions from a hardware or software point.

The *current value* field (32 bits) shows the value of the counter, which generated the event.

The *flag* field (8 bits) has the same meaning as in previous counter objects.

16-Bit Counter Change Event without Time

Data Object: 22

Variation: 02

Type: Event

Grouping: Counter

The *16-bit counter change event without time* represents a counter value that has exceeded a configured dead band. This can be accumulated pulses or transitions from a hardware or software point.

The *current value* field (16 bits) shows the value of the counter, which generated the event.

The *flag* field (8 bits) has the same meaning as in previous counter objects.

32-Bit Counter Change Event with Time

Data Object: 22

Variation: 05

Type: Event

Grouping: Counter

The *32-bit counter change event with time* represents a counter value that has exceeded a configured dead band. This can be accumulated pulses or transitions from a hardware or software point.

The *current value* field (32 bits) shows the value of the counter, which generated the event.

The *time* field (48 bits) shows the time at which the processing caused the event.

The *flag* field (8 bits) has the same meaning as in previous counter objects.

16-Bit Counter Change Event with Time

Data Object: 22

Variation: 06

Type: Event

Grouping: Counter

The *16-bit counter change event with time* represents a counter value that has exceeded a configured dead band. This can be accumulated pulses or transitions from a hardware or software point.

The *current value* field (16 bits) shows the value of the counter, which generated the event.

The *time* field (48 bits) shows the time at which the processing caused the event.

The *flag* field (8 bits) has the same meaning as in previous counter objects.

32-Bit Frozen Counter Event without Time

Data Object: 23

Variation: 01

Type: Frozen Event

Grouping: Counter

The *32-bit frozen counter event without time* represents a frozen counter value that is reported as an event. This can be accumulated pulses or transitions from a hardware or software point.

The *frozen value* field (32 bits) shows the value at the time of freezing.

The *flag* field (8 bits) has the same meaning as in previous counter objects.

16-Bit Frozen Counter Event without Time**Data Object: 23****Variation: 02****Type: Frozen Event****Grouping: Counter**

The *16-bit frozen counter event without time* represents a frozen counter value that is reported as an event. This can be accumulated pulses or transitions from a hardware or software point.

The *frozen value* field (16 bits) shows the value at the time of freezing.

The *flag* field (8 bits) has the same meaning as in previous counter objects.

32-Bit Frozen Counter Event with Time**Data Object: 23****Variation: 05****Type: Frozen Event****Grouping: Counter**

The *32-bit frozen counter event with time* represents a frozen counter value that is reported as an event. This can be accumulated pulses or transitions from a hardware or software point.

The *frozen value* field (32 bits) shows the value at the time of freezing.

The *time* field (48 bits) shows the time at which the processing caused the event.

The *flag* field (8 bits) has the same meaning as in previous counter objects.

16-Bit Frozen Counter Event with Time**Data Object: 23****Variation: 06****Type: Frozen Event****Grouping: Counter**

The *16-bit frozen counter event with time* represents a frozen counter value that is reported as an event. This can be accumulated pulses or transitions from a hardware or software point.

The *frozen value* field (16 bits) shows the value at the time of freezing.

The *time* field (48 bits) shows the time at which the processing caused the event.

The *flag* field (8 bits) has the same meaning as in previous counter objects.

16-Bit Analog Input**Data Object: 30****Variation: 02****Type: Static****Grouping: Analog Input**

The *16-bit analog input* is an information object used to represent a hardware or software analog point. The 16-bit signed value could represent a digitized signal or calculated value.

The *value* field (16 bits) shows the current value of the analog input at the time of reporting, or the last reported value from the originating device.

The *flag* field is stored in 8 bits, as follows:

The *on-line* bit (0) indicates that the analog input point has been read successfully. If this field is set to *off-line*, the state of the analog point may not be correct.

The *restart* bit (1) indicates that the field device that originated the data object has been re-started. This device may be the device reporting this data object.

The *communication lost* bit (2) indicates that the device reporting this data object has lost communication with the originator of the data object.

The *remote forced data* bit (3) indicates that the state of the analog input has been forced to its current value at the originating device.

The *local forced data* bit (4) indicates that the state of the analog input has been forced to its current value at the device reporting this data object.

The *over-range* bit (5) indicates that the digitized signal or calculation has exceeded $+2^{15} - 1$ positively, or -2^{15} negatively. The actual *value* field can be ignored, as its value is not defined.

The *reference check* bit (6) indicates that the reference signal used to digitize the analog input is not stable and the resulting digitized value may not be correct.

16-Bit Analog Input without Flag

Data Object: 30

Variation: 04

Type: Static

Grouping: Analog Input

The *16-bit analog input* is an information object used to represent a hardware or software analog point. The 16-bit signed value could represent a digitized signal or calculated value.

The *current value* field (16 bits) shows the current value of the analog input at the time of reporting, or the last reported value from the originating device.

The use of this variation implies that the input point is on-line and that all other flag bits are normal (i.e. this variation implies that flag = 1).

16-Bit Analog Input Change Event without Time

Data Object: 32

Variation: 02

Type: Event

Grouping: Analog Input

The *16-bit analog change event without time* is an information object used to represent a changed hardware or software analog point. The 16-bit signed value could represent a digitized signal or calculated value.

The *current value* field (16 bits) shows the current value of the analog input at the time of reporting, or the last reported value from the originating device. This object will only be reported if the *current value* and the last reported value differs by a configurable dead band value. This filtering is commonly known as dead banding.

The *flag* field (8 bits) has the same meaning as previous analog input objects.

16-Bit Analog Input Change Event with Time

Data Object: 32

Variation: 04

Type: Event

Grouping: Analog Input

The *16-bit analog change event with time* is an information object used to represent a changed hardware or software analog point. The 16-bit signed value could represent a digitized signal or calculated value.

The *current value* field (16 bits) shows the value of the analog input at the time specified in *time*.

The *time* field (48 bits) shows the time at which the processing caused the event.

The *flag* field (8 bits) has the same meaning as previous analog input objects.

16-Bit Analog Output Status

Data Object: 40

Variation: 02

Type: Static

Grouping: Analog Output

The *16-bit analog output status* information object represents the actual value of a hardware DAC analog output or software point and associated status.

This object can be returned after an analog output operation is performed in order to determine the success of the operation.

The *actual value* field (16 bits) contains the current value of the analog output.

The *flag* field is stored in 8 bits, as follows:

The *on-line* bit (0) indicates that the analog output point has been read successfully. If this field is set to *off-line*, the state of the analog point may not be correct.

The *restart* bit (1) indicates that the field device that originated the data object has been re-started. This device may be the device reporting this data object.

The *communication lost* bit (2) indicates that the device reporting this data object has lost communication with the originator of the data object.

The *remote forced data* bit (3) indicates that the state of the analog output has been forced to its current state at the originating device.

16-Bit Analog Output Block

Data Object: 41

Variation: 02

Type: Static

Grouping: Analog Output

The *16-bit analog output block* information object represents the desired value of a hardware DAC analog output or software point. The value represented is merely logical, as the value may be scaled and/or manipulated before any output level is set.

The *requested value* field (16 bits) contains the desired value of the analog output. The *actual value* of the analog output is returned in the *analog output status* object.

The *control status* field (8 bits) indicates the status of the analog control operation in the same way as the *control relay output block*. The definition of this field is the same as the *control relay output block*.

Time and Date

Data Object: 50

Variation: 01

The *time and date* object is an information object that represents the absolute time of day and date. This object should be used for time-synchronization.

Absolute Time field (48 bits) is recorded as milliseconds since midnight, January 1st, 1970, at zero hours, zero minutes, zero seconds, and milliseconds.

Time Delay Fine

Data Object: 52

Variation: 02

The *time delay fine* information object (16 bits) represents a relative time (ms) that indicates a time period starting from the time of message reception. This object can be used in time-synchronization to perform path delay measurement calculations or other functions that require time-based calibration.

Class 0 Data

Data Object: 60

Variation: 01

The class 0 data object is an object place-holder that specifies a class of zero or more information elements. These elements can be entire object types, a specific variation, certain points of the variation, or any combination of these. The data specified by this object type is configurable within the responding station.

Class 0 data is potentially any information object not assigned to class 1, 2, or 3. That is, class 0 data is non-priority data.

The class 0 data object does not carry any information in itself, and therefore does not have an object coding. Class 0 is a null class to which all data objects not assigned to other classes can belong by default.

In a typical minimum implementation, it is expected that a Master device will poll frequently for Class 1, 2 or 3 Data, interspersed with infrequent Class 0 Data polls. Every event object reported by the Slave device therefore belongs to Class 1, Class 2 or Class 3. Every static object reported by the Slave device belongs to Class 0.

A Slave device will not report event objects for a particular data point by assigning event objects from the data point to Class 0. Since Class 0 is the class of static data and event objects cannot be Class 0, the Slave will not produce event objects for the data point.

Class 1 Data

Data Object: 60

Variation: 02

The *class 1 data* object is an object place-holder that specifies a class of zero or more information elements. These elements can be entire object types, a specific variation, certain points of the

variation, or any combination of these. The data specified by this object type is configurable within the responding station.

The responding station does not send the *class 1 data* object, as it does not contain any actual information, but is simply an identifier for other objects.

The *class 1 data* object is used to request a configured group, usually changes, of information objects from a responding station. This data object does not carry any information in itself, and therefore does not have an object coding.

Typically, *class 1 data* has higher priority than *class 2* and *class 3* data.

Class 2 Data

Data Object: 60

Variation: 03

The *class 2 data* object is an object place-holder that specifies a class of zero or more information elements. These elements can be entire object types, a specific variation, certain points of the variation, or any combination of these. The data specified by this object type is configurable within the responding station.

The responding station does not send the *class 2 data* object, as it does not contain any actual information, but is simply an identifier for other objects.

The *class 2 data* object is used to request a configured group, usually changes, of information objects from a responding station. This data object does not carry any information in itself, and therefore does not have an object coding.

Class 3 Data

Data Object: 60

Variation: 04

The *class 3 data* object is an object place-holder that specifies a class of zero or more information elements. These elements can be entire object types, a specific variation, certain points of the variation, or any combination of these. The data specified by this object type is configurable within the responding station.

The responding station does not send the *class 3 data* object, as it does not contain any actual information, but is simply an identifier for other objects.

The *class 3 data* object is used to request a configured group, usually an event, of information objects from a responding station. The data object does not carry any information in itself, and therefore does not have an object coding.

Regras para Numeração (Endereçamento) de Pontos

Cada ponto no protocolo DNP possui um número associado para referência ao mesmo nas mensagens. Este número equivale a um endereço do ponto e no protocolo DNP ele é denominado de *índice (index)*. As seguintes regras devem ser levadas em conta na interpretação do número (endereço) dos objetos DNP (camada de aplicação, campo **range**) em conjunto com tipos de objetos e suas variações. Elas são importantes para que fazer a associação entre os pontos físicos e virtuais e os endereços DNP para objetos de dados.

Regra 1:

Cada ponto físico ou virtual está associado a um ou mais tipos de objeto, e cada tipo de objeto, por sua vez, pode ter mais de uma variação associada ao mesmo. Por exemplo, se a remota possui 16

contadores (objetos 20) numerados de 0 a 15, o ponto 5 (ou qualquer destes pontos) pode ser referenciado numa mensagem de quatro maneiras diferentes:

- **Object 20, variation 1, range 5** refere-se ao valor atual do ponto 5 no formato de 32 bits.
- **Object 20, variation 2, range 5** refere-se ao valor atual do ponto 5 no formato de 16 bits.
- **Object 20, variation 3, range 5** refere-se à variação do ponto 5 em relação a última vez que ele foi enviado, no formato de 32 bits.
- **Object 20, variation 4, range 5** refere-se à variação do ponto 5 em relação a última vez que ele foi enviado, no formato de 16 bits.

Regra 2:

Quando um objeto é enviado como um evento também existem muitas variações possíveis para o mesmo. Qual variação será utilizada para reportar o evento é uma opção definida pela aplicação. De qualquer forma apenas um evento deve ser enviado, na variação que estiver definida para tal.

Quando a remota responde a uma requisição de dados de uma determinada **class** ou a uma requisição por um objeto com variação 0, não está especificado na requisição qual a variação que deverá ser utilizada no envio do(s) objeto(s) solicitado(s), mas também nestes casos apenas uma variação deverá ser enviada, e também é a aplicação que define a variação utilizada.

A remota Hadron permite configurar qual variação de cada tipo de objeto será enviada quando a variação não está definida na requisição ou no caso de eventos, através do parâmetro “variação default”.

Regra 3:

Como já foi destacado na regra 1 cada ponto está associado a mais de um tipo de objeto. Estes objetos sempre pertencem ao mesmo grupo (ver seção “Grupos de Objetos DNP”). Para cada um destes grupos, aplica-se as seguintes regras:

- Os mesmos objetos de dados do tipo estático e do tipo evento, que possuem a mesma numeração (endereço) DNP referem-se ao mesmo ponto físico ou virtual da remota.
- Os objetos de dados do tipo estático congelado e do tipo evento congelado, que possuem a mesma numeração (endereço) DNP referem-se ao mesmo ponto físico ou virtual da remota.

Por exemplo:

- as mesmas entradas binárias representadas por **Object 1** com **variation 1** e 2 com determinada numeração (endereço) também são representados por **Object 2** com **variation 1, 2** e 3 com a mesma numeração (relação entre tipos estático e evento);
- os mesmos contadores representados por **Object 20** com **variation 1, 2, 3** e 4 com determinada numeração também são representados por **Object 22** com **variation 1, 2, 3, 4, 5, 6, 7** e 8 com a mesma numeração (relação entre tipos estático e evento);
- os mesmos contadores de bit representados por **Object 21** com **variation 1, 2, 3, 4, 5, 6, 7** e 8 com determinada numeração também são representados por **Object 23** com **variation 1, 2, 3, 4, 5, 6, 7** e 8 com a mesma numeração (relação entre tipos congelado e estático congelado);

Objetos 20 e 21 com mesma numeração não são necessariamente o mesmo ponto. Não há uma relação direta entre objetos congelados e não congelados.

Na remota Hadron contadores e contadores congelados com a mesma numeração (índice) fazem referência ao mesmo ponto físico.

Regra 4:

Grupos de objetos que tenham apenas um ponto por objeto ou nos casos em que o número do ponto não for necessário o **range** reportado informa índice do ponto igual a 0 ou então quantidade igual a 1, de acordo com o frame, quando inserido numa mensagem que requer o número do ponto ou a quantidade de pontos.

3. Interoperabilidade DNP

O driver AL-2741 implementa a norma DNP-L2 (nível 2). Este capítulo descreve a interoperabilidade do driver, através da tabela de implementação (**Implementation Table**) e do perfil do dispositivo (**device profiles**).

Ambos os documentos acima citados são formatados pela própria norma DNP, e são mantidos na sua forma original, em inglês, para manter a fidelidade das informações e facilitar a comparação com os documentos correspondentes de outros equipamentos que tenham que comunicar-se ou compartilhar o meio físico com a remota Hadron.

Perfil do Driver AL-2741

A tabela abaixo apresenta o “**Device Profile Document**” no formato padrão apresentado no documento “DNP V3.00 Subset Definitions Document”. Esta tabela é apenas um dos componentes que definem a interoperabilidade da remota. Ele deve ser utilizado em conjunto com as seguintes orientações para definir completamente a interoperabilidade da remota Hadron:

- A tabela de implementação (**Implementation Table**) apresentada na próxima seção.
- Os métodos e recursos disponíveis através do ProHadron para configuração, descritos no próximo capítulo.

DNP V3.00	
DEVICE PROFILE DOCUMENT	
OBS: Ver na próxima seção Implementation Table para o AL-2741	
Vendor Name: ALTUS SISTEMAS DE INFORMATICA SA	
AL-2741 (driver escravo)	
Highest DNP Level Supported:	Device Function:
For Requests: Level 2	<input type="checkbox"/> Master
For Responses: Level 2	<input checked="" type="checkbox"/> Slave
Notable objects, functions, and/or qualifiers supported in addition to the Highest DNP Levels Supported (the complete list is described in the attached table):	
<p>For static (non-change-event) object requests, request qualifier codes 00 and 01 (start-stop), 07 and 08 (limited quantity), and 17 and 28 (index) are supported in addition to request qualifier code 06 (no range – or all points). Static object requests received with qualifiers 00, 01, 06, 07, or 08, will be responded with qualifiers 00 or 01. Static object requests received with qualifiers 17 or 28 will be responded with qualifiers 17 or 28. For change-event object requests, qualifiers 17 or 28 are always responded.</p> <p>16-bit and 32-bit Analog Change Events with Time may be requested.</p> <p>The read function code for Object 50 (Time and Date), variation 1 is supported.</p>	
Maximum Data Link Frame Size (octets):	Maximum Application Fragment Size (octets):
Transmitted: 292	Transmitted: Configurable – See parameter “Application Response Fragment Size”
Received: 292	Received: 2048

Sends/Executes Control Operations:	
WRITE Binary Outputs	<input checked="" type="checkbox"/> Never <input type="checkbox"/> Always <input type="checkbox"/> Sometimes <input type="checkbox"/> Configurable
SELECT/OPERATE	<input type="checkbox"/> Never <input checked="" type="checkbox"/> Always <input type="checkbox"/> Sometimes <input type="checkbox"/> Configurable
DIRECT OPERATE	<input type="checkbox"/> Never <input checked="" type="checkbox"/> Always <input type="checkbox"/> Sometimes <input type="checkbox"/> Configurable
DIRECT OPERATE – NO ACK	<input type="checkbox"/> Never <input checked="" type="checkbox"/> Always <input type="checkbox"/> Sometimes <input type="checkbox"/> Configurable
Count > 1	<input type="checkbox"/> Never <input type="checkbox"/> Always <input checked="" type="checkbox"/> Sometimes <input type="checkbox"/> Configurable
Pulse On	(may be available for some IED points, but is not available for all internal RTU IO points) <input type="checkbox"/> Never <input type="checkbox"/> Always <input checked="" type="checkbox"/> Sometimes <input type="checkbox"/> Configurable
Pulse Off	(may not be available for some IED points, but is available for all internal RTUs IO points) <input type="checkbox"/> Never <input type="checkbox"/> Always <input checked="" type="checkbox"/> Sometimes <input type="checkbox"/> Configurable
Latch On	(may not be available for some IED points, and is not available for all internal RTUs IO points) <input type="checkbox"/> Never <input type="checkbox"/> Always <input checked="" type="checkbox"/> Sometimes <input type="checkbox"/> Configurable
Latch Off	(may not be available for some IED points, but is available for all internal RTUs IO points) <input type="checkbox"/> Never <input type="checkbox"/> Always <input checked="" type="checkbox"/> Sometimes <input type="checkbox"/> Configurable
Queue	<input checked="" type="checkbox"/> Never <input type="checkbox"/> Always <input type="checkbox"/> Sometimes <input type="checkbox"/> Configurable
Clear Queue	<input checked="" type="checkbox"/> Never <input type="checkbox"/> Always <input type="checkbox"/> Sometimes <input type="checkbox"/> Configurable
Reports Binary Input Change Events when no specific variation requested:	Reports time-tagged Binary Input Change Events when no specific variation requested:
<input type="checkbox"/> Never <input type="checkbox"/> Only time-tagged <input type="checkbox"/> Only non-time-tagged <input checked="" type="checkbox"/> Configurable , see parameter Default Object 02 Variation	<input type="checkbox"/> Never <input checked="" type="checkbox"/> Binary Input Change With Time <input type="checkbox"/> Binary Input Change With Relative Time <input type="checkbox"/> Configurable (attach explanation)
Sends Unsolicited Responses:	Sends Static Data in Unsolicited Responses:
<input type="checkbox"/> Never <input checked="" type="checkbox"/> Configurable , see parameter Unsolicited Responses Permitted <input type="checkbox"/> Only certain objects <input type="checkbox"/> Sometimes (attach explanation) <input checked="" type="checkbox"/> ENABLE/DISABLE UNSOLICITED Function codes supported	<input checked="" type="checkbox"/> Never <input type="checkbox"/> When Device Restarts <input type="checkbox"/> When Status Flags Change No other options are permitted.

Default Counter Object/Variation: <input type="checkbox"/> No Counters Reported <input checked="" type="checkbox"/> Configurable: see parameters Default Object 20 Variation and Default Object 21 Variation	Counters Roll Over at: <input type="checkbox"/> No Counters Reported <input type="checkbox"/> Configurable (attach explanation) <input type="checkbox"/> 16 Bits <input type="checkbox"/> 32 Bits <input checked="" type="checkbox"/> Other Value: All counters are internally stored with 32 bits, and both 16 and 32-bit roll over bits are generated. The roll over bit will be set accordingly to the request from the master (16 or 32 bits), in static data polling. For events, the roll over bit will depend on the default variation chosen for the object (a 16 or 32 bit variation).
Sends Multi-Fragment Responses: <input checked="" type="checkbox"/> Yes <input type="checkbox"/> No	

Tabela 3-1 Tabela com Perfil da Remota

Tabela de Implementação para o AL-2741

A tabela a seguir apresenta as variações, códigos de função e qualificadores suportados pelo driver AL-2741, tanto em mensagens requisitadas quanto em mensagens respondidas.

Para objetos do tipo estáticos, as requisições enviadas com qualificadores 00, 01, 06, 07 ou 08 serão respondidas com qualificadores 00 ou 01.

Para objetos do tipo estáticos, as requisições enviadas com qualificadores 17 ou 28 serão respondidas com qualificadores 17 ou 28.

Para objetos do tipo eventos, requisições com qualificadores 17 ou 28 serão sempre respondidas.

A coluna **REQUEST** contém códigos de funções e qualificadores que serão interpretados (**parsed**) pelo driver AL-2741 para uma dada dupla **object/variation**. Entretanto, isto não quer dizer que o driver responderá a estas requisições com objetos de dados. Se a coluna **RESPONSE** está vazia, isto significa que o driver responderá uma informação de erro (como por exemplo objeto não implementado).

A coluna **RESPONSE** contém códigos de funções e qualificadores que serão utilizados nas respostas do drivers para uma dada dupla **object/variation**. Quando esta coluna está vazia, isto significa que o driver não responderá ou então responderá uma informação de erro (como por exemplo objeto não implementado).

Os textos sublinhados na próxima tabela indicam funcionalidades além do DNP-L2, ou seja, funcionalidades além DNP-L3 ou até mesmo além destas.

OBJECT			REQUEST (will be parsed)		RESPONSE (will be responded with)	
Object Number	Variation Number	Description	Function Codes (dec)	Qualifier Codes (hex)	Function Codes (dec)	Qualifier Codes (hex)
1	0	Binary Input (Variation 0 is used to request default variation)	1 (read)	<u>00, 01</u> (start-stop) 06 (no range, or all) <u>07, 08</u> (limited qty) <u>17, 28</u> (index)		
1	1	Binary Input	<u>1</u> (read) <u>22</u> (assign class)	<u>00, 01</u> (start-stop) 06 (no range, or all) <u>07, 08</u> (limited qty) <u>17, 28</u> (index)	129 (response)	00, 01 (start-stop) <u>17, 28</u> (index – see note 1)

OBJECT			REQUEST (will be parsed)		RESPONSE (will be responded with)	
Object Number	Variation Number	Description	Function Codes (dec)	Qualifier Codes (hex)	Function Codes (dec)	Qualifier Codes (hex)
1	2	Binary Input with Status	1 (read) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
2	0	Binary Input Change (Variation 0 is used to request default variation)	1 (read)	06 (no range, or all) 07, 08 (limited qty)		
2	1	Binary Input Change without Time	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
2	2	Binary Input Change with Time	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
2	3 (parse only)	Binary Input Change with Relative Time	1 (read)	06 (no range, or all) 07, 08 (limited qty)		
10	0	Binary Output Status (Variation 0 is used to request default variation)	1 (read)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)		
10	2	Binary Output Status	1 (read)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
12	1	Control Relay Output Block	3 (select) 4 (operate) 5 (direct op) 6 (dir. op, noack)	00, 01 (start-stop) 07, 08 (limited qty) 17, 28 (index)	129 (response)	echo of request
20	0	Binary Counter (Variation 0 is used to request default variation)	1 (read) 7 (freeze) 8 (freeze noack) 9 (freeze clear) 10 (frz. cl. noack) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)		
20	1	32-Bit Binary Counter	1 (read) 7 (freeze) 8 (freeze noack) 9 (freeze clear) 10 (frz. cl. noack) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
20	2	16-Bit Binary Counter	1 (read) 7 (freeze) 8 (freeze noack) 9 (freeze clear) 10 (frz. cl. noack) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
20	5	32-Bit Binary Counter without Flag	1 (read) 7 (freeze) 8 (freeze noack) 9 (freeze clear) 10 (frz. cl. noack) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
20	6	16-Bit Binary Counter without Flag	1 (read) 7 (freeze) 8 (freeze noack) 9 (freeze clear) 10 (frz. cl. noack) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
21	0	Frozen Counter (Variation 0 is used to request default variation)	1 (read) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)		
21	1	32-Bit Frozen Counter	1 (read) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
21	2	16-Bit Frozen Counter	1 (read) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)

OBJECT			REQUEST (will be parsed)		RESPONSE (will be responded with)	
Object Number	Variation Number	Description	Function Codes (dec)	Qualifier Codes (hex)	Function Codes (dec)	Qualifier Codes (hex)
21	9	32-Bit Frozen Counter without Flag	1 (read) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
21	10	16-Bit Frozen Counter without Flag	1 (read) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
22	0	Counter Change Event (Variation 0 is used to request default variation)	1 (read)	06 (no range, or all) 07, 08 (limited qty)		
22	1	32-Bit Counter Change Event	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
22	2	16-Bit Counter Change Event	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
22	5	32-Bit Counter Change Event with Time	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
22	6	16-Bit Counter Change Event with Time	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
23	0	Frozen Counter Event (Variation 0 is used to request default variation)	1 (read)	06 (no range, or all) 07, 08 (limited qty)		
23	1	32-Bit Frozen Counter Event	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
23	2	16-Bit Frozen Counter Event	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
23	5	32-Bit Frozen Counter Event with Time	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
23	6	16-Bit Frozen Counter Event with Time	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
30	0	Analog Input (Variation 0 is used to request default variation)	1 (read) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)		
30	1	32-Bit Analog Input	1 (read) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
30	2	16-Bit Analog Input	1 (read) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
30	3	32-Bit Analog Input without Flag	1 (read) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
30	4	16-Bit Analog Input without Flag	1 (read) 22 (assign class)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
32	0	Analog Change Event (Variation 0 is used to request default variation)	1 (read)	06 (no range, or all) 07, 08 (limited qty)		
32	1	32-Bit Analog Change Event without Time	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
32	2	16-Bit Analog Change Event without Time	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
32	3	32-Bit Analog Change Event with Time	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
32	4	16-Bit Analog Change Event with Time	1 (read)	06 (no range, or all) 07, 08 (limited qty)	129 (response) 130 (unsol. resp)	17, 28 (index)
40	0	Analog Output Status (Variation 0 is used to request default variation)	1 (read)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)		
40	1	32-Bit Analog Output Status	1 (read)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)

OBJECT			REQUEST (will be parsed)		RESPONSE (will be responded with)	
Object Number	Variation Number	Description	Function Codes (dec)	Qualifier Codes (hex)	Function Codes (dec)	Qualifier Codes (hex)
40	2	16-Bit Analog Output Status	1 (read)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
41	1	32-Bit Analog Output Block	3 (select) 4 (operate) 5 (direct op) 6 (dir. op, noack)	00, 01 (start-stop) 07, 08 (limited qty) 17, 28 (index)	129 (response)	echo of request
41	2	16-Bit Analog Output Block	3 (select) 4 (operate) 5 (direct op) 6 (dir. op, noack)	00, 01 (start-stop) 07, 08 (limited qty) 17, 28 (index)	129 (response)	echo of request
50	0	Time and Date	1 (read)	00, 01 (start-stop) 06 (no range, or all) 07, 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
50	1	Time and Date	1 (read) 2 (write)	00, 01 (start-stop) 06 (no range, or all) 07 (limited qty=1) 08 (limited qty) 17, 28 (index) 00, 01 (start-stop) 07 (limited qty=1) 08 (limited qty) 17, 28 (index)	129 (response)	00, 01 (start-stop) 17, 28 (index – see note 1)
52	2	Time Delay Fine			129 (response)	07 (limited qty) (qty = 1)
60	0	Class 0, 1, 2, and 3 Data	1 (read) 20 (enbl. unsol.) 21 (dsbl. unsol.)	06 (no range, or all)		
60	1	Class 0 Data	1 (read) 22 (assign class)	06 (no range, or all)		
60	2	Class 1 Data	1 (read) 20 (enbl. unsol.) 21 (dsbl. unsol.) 22 (assign class)	06 (no range, or all) 07, 08 (limited qty) 06 (no range, or all)		
60	3	Class 2 Data	1 (read) 20 (enbl. unsol.) 21 (dsbl. unsol.) 22 (assign class)	06 (no range, or all) 07, 08 (limited qty) 06 (no range, or all)		
60	4	Class 3 Data	1 (read) 20 (enbl. unsol.) 21 (dsbl. unsol.) 22 (assign class)	06 (no range, or all) 07, 08 (limited qty) 06 (no range, or all)		
No Object (function code only)			13 (cold restart)			
No Object (function code only)			14 (warm restart)			
No Object (function code only)			23 (delay meas.)			

Tabela 3-2 Tabela de Implementação do AL-2741

Para objetos do tipo estático, respostas com qualificadores 17 ou 28 são geradas quando uma requisição com qualificadores 17 ou 28, respectivamente, for recebida. Caso contrário, requisições de objetos estáticos com qualificadores 00, 01, 06, 07 ou 08 serão respondidas com qualificadores 00 ou 01. Para objetos do tipo evento, requisições com qualificadores 17 ou 28 são sempre respondidos.

4. Configuração

A configuração da remota Hadron é feita através do configurador ProHadron. A primeira parte da configuração não depende do driver escravo utilizado (neste caso DNP), devendo ser preenchida conforme orientações contidas no Manual de Utilização da Remota Hadron, HD3001, Cód. Doc. MU208301.

A segunda parte contém definições específicas de cada driver. Neste capítulo é explicada a configuração do driver escravo AL-2741. Como uma remota Hadron pode possuir até quatro drivers escravos, para cada driver presente na remota existem telas que deverão ser preenchidas com informações específicas. Desta forma, todas as telas e opções aqui explicadas podem ser repetidas até quatro vezes no caso de utilização máxima de drivers escravos. Existe um conjunto de telas para cada driver, todas elas idênticas caso os drivers implementem o mesmo protocolo.

Telas do ProHadron

Existem três telas de configuração para cada driver AL-2741. Como já foi frisado, uma remota pode possuir até quatro drivers escravos. Estes drivers podem implementar o mesmo protocolo ou não. Portanto a remota tem a capacidade de se comunicar com mais de um tipo de protocolo simultaneamente.



Figura 4-1. Tela com menu para edição de parâmetros do driver

Tela de Dados Gerais do Driver (Driver General Data)

Esta Tela contém dados gerais sobre o driver escravo, como canal de comunicação utilizado, taxa de bits do canal, endereço da remota entre outros. Através dela estão disponíveis sete guias descritas a seguir.

Guia UTR (RTU)

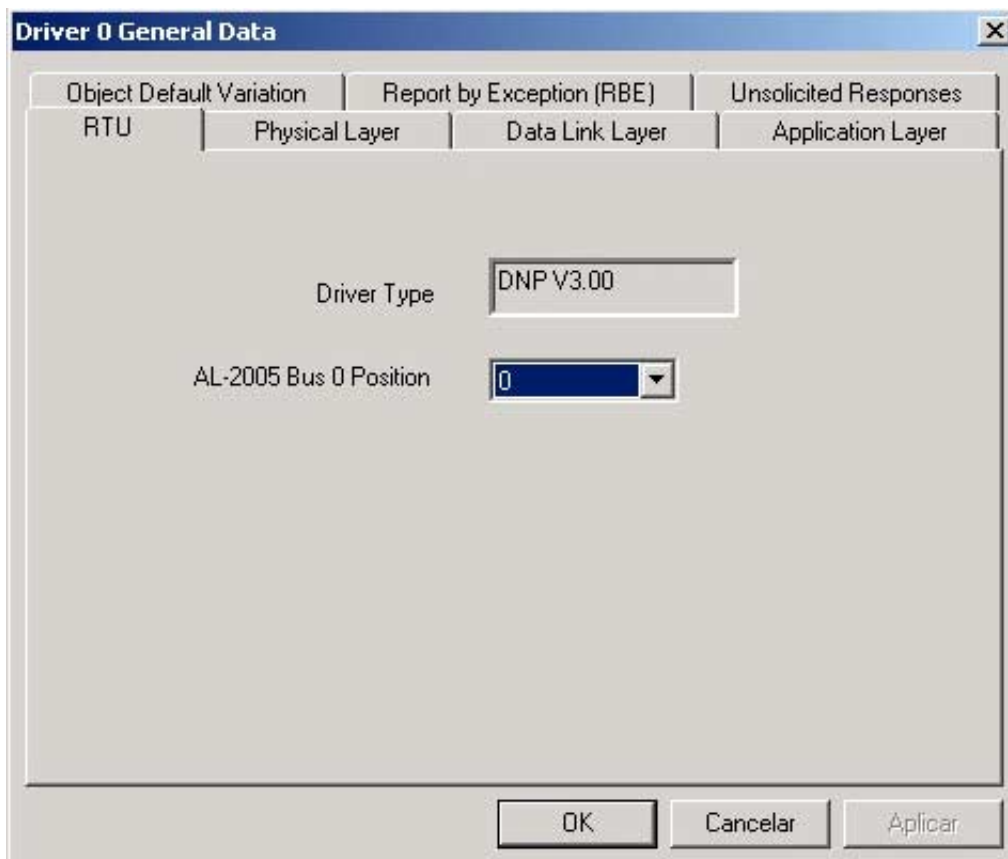


Figura 4-2. Guia UTR

Tipo do driver (Driver Type)

Contém um campo não editável que informa o protocolo implementado pelo driver que está sendo configurado, no caso DNP V3.00.

Posição no Barramento 0 do AL-2005 (AL-2005 Bus 0 Position)

Informa a posição do barramento 0 onde está o AL-2005 no qual este driver será executado. As posições disponíveis dependem do tipo de bastidor utilizado.

A primeira posição em qualquer barramento é sempre a de número 0.

Guia Camada Física (Physical Layer)

Nesta guia estão todas as definições da camada física do protocolo.

Porta Serial do AL-2005 (AL-2005 Serial Port)

O coprocessador AL-2005 possui duas portas seriais (A e B). Este parâmetro define qual a porta serial será utilizada pelo driver. Os seguintes valores podem ser designados:

- 1: A
- 2: B

- 3: ambas portas seriais A e B. Este caso é utilizado quando existe redundância de meio físico. A remota sempre responde pela porta que recebeu a última requisição do mestre. Ambas as portas possuem o mesmo baudrate.

Taxa de Bits do Canal (Baudrate)

Define a taxa de bits de transmissão pelo canal serial em bps (bits por segundo). Os seguintes valores podem ser configurados:

- 50
- 75
- 110
- 150
- 300
- 600
- 1200
- 2400
- 4800
- 9600
- 14400
- 19200
- 28800
- 33600
- 38400

Paridade (Parity)

Este parâmetro é definido no protocolo DNP como sendo fixo, com paridade ímpar. Contudo, na remota Hadron ele pode ser configurado com os seguintes valores:

- 0: sem paridade
- 1: paridade ímpar
- 2: paridade par

Bits Adicionais Finais do Quadro Físico (Stop Bits)

Este parâmetro é definido no protocolo DNP como sendo fixo, com valor 1. Contudo, na remota Hadron ele pode ser configurado com os seguintes valores:

- 1: um stop bit
- 2: dois stop bit

Sinais de Modem (Handshake - Modem Signals)

Este parâmetro como os sinais de modem são utilizados. Ele pode ser configurado com os seguintes valores:

- 0: none (sem sinais)
- 1: RTS signal always on (RTS sempre ligado)
- 2: RTS/CTS modem handshake (utilização dos sinais RTS e CTS)

Tempo Mínimo para Transmissão (Transmission Delay)

Tempo mínimo, em ms, que o driver irá aguardar após receber algum caracter na sua porta serial, antes que ele transmita uma resposta. Valores possíveis estão entre 0 e 1000 ms.

Este parâmetro é particularmente importante em redes RS485 ou em comunicações utilizando rádio-modem, nos quais deve transcorrer um certo tempo, após o mestre transmitir sua mensagem, para que ele

Guia Camada de Enlace (Data Link Layer)

Nesta guia estão as definições para a camada de enlace do protocolo.

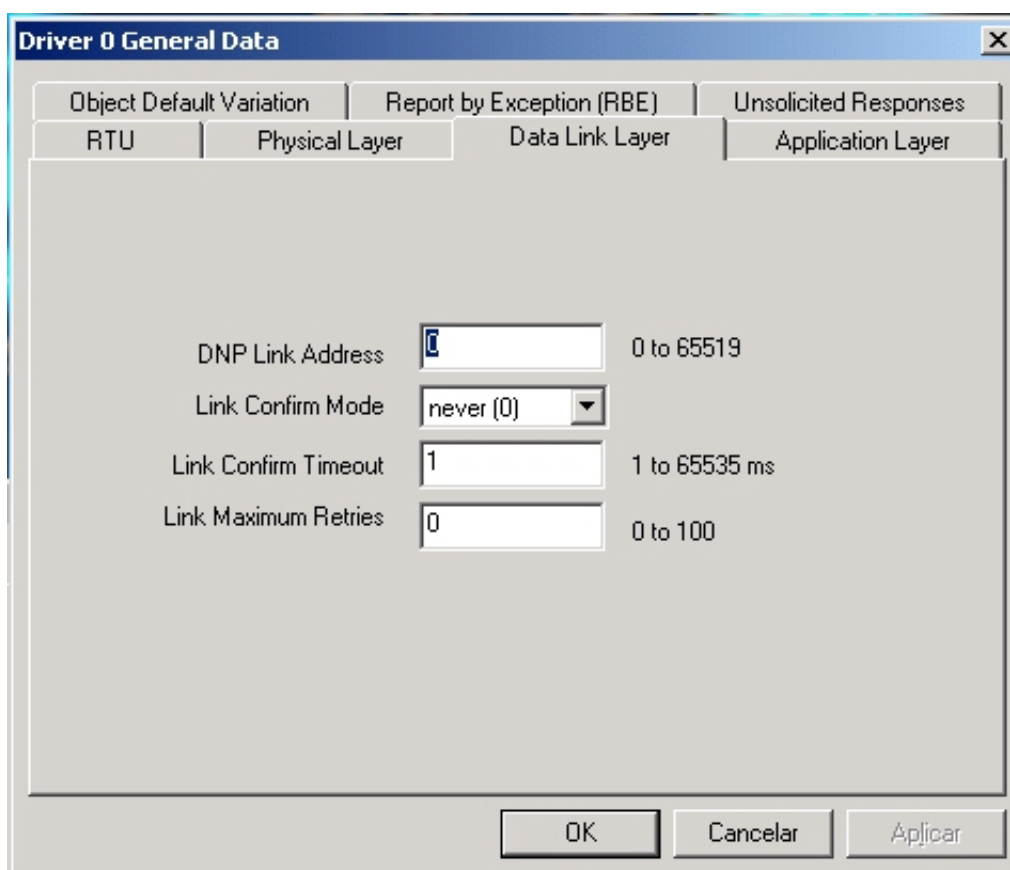


Figura 4-3. Guia Camada de Enlace

Endereço de Enlace DNP (DNP Link Address)

Define o endereço da remota num barramento DNP. Frames enviados com este endereço serão recebidos pela remota. Outros endereços serão ignorados.

Os valores possíveis estão na faixa de 0 a 65519. Endereços de 65520 até 65535 são reservados para broadcast.

Modo de Confirmação de Enlace (Link Confirm Mode)

Este parâmetro é utilizado para determinar se a remota deve solicitar ou não confirmações, do mestre, no nível de enlace sobre o recebimento de frames enviados pela remota. Também é possível estabelecer quando estas solicitações de confirmações devem ocorrer. Os seguintes valores podem ser designados pela remota:

- 0 (never): a remota nunca solicita confirmações de enlace.
- 1 (sometimes): a remota solicita confirmações de enlace somente em frames intermediários, isto é, que não seja o último de uma mensagem multi-frame fragmentada.
- 2 (always): a remota requisita confirmações de enlace para todos os frames transmitidos.

Timeout para Confirmação do Enlace (Link Confirm Timeout)

É utilizado para especificar, em ms, o timeout que a camada de enlace irá aguardar por uma confirmação do mestre para um frame anteriormente enviado com solicitação de confirmação. Este valor é independente do Timeout de Confirmação da Aplicação. Se a opção de confirmação de enlace está habilitada, o temporizador do Timeout de Confirmação da Aplicação não inicia até que uma confirmação de enlace seja recebida.

Este parâmetro pode assumir valores entre 1 até 65535 ms.

Número Máximo de Retentativas de Enlace (Link Maximum Retries)

É a quantidade de vezes que um frame será retransmitido sempre que uma confirmação foi requisitada mas não foi recebida dentro de um período de Timeout para Confirmação do Link. Observe que este

valor é um adicional à primeira tentativa de envio de um frame. Por exemplo, se este parâmetro for 2, então uma tentativa fracassada de transmissão do frame resultará no envio de três frames idênticos até que uma falha seja reportado para ao usuário da camada de enlace.

Este parâmetro pode assumir valores entre 0 até 100.

Guia Camada de Aplicação (*Application Layer*)

Nesta guia estão as definições para a camada de aplicação do protocolo.

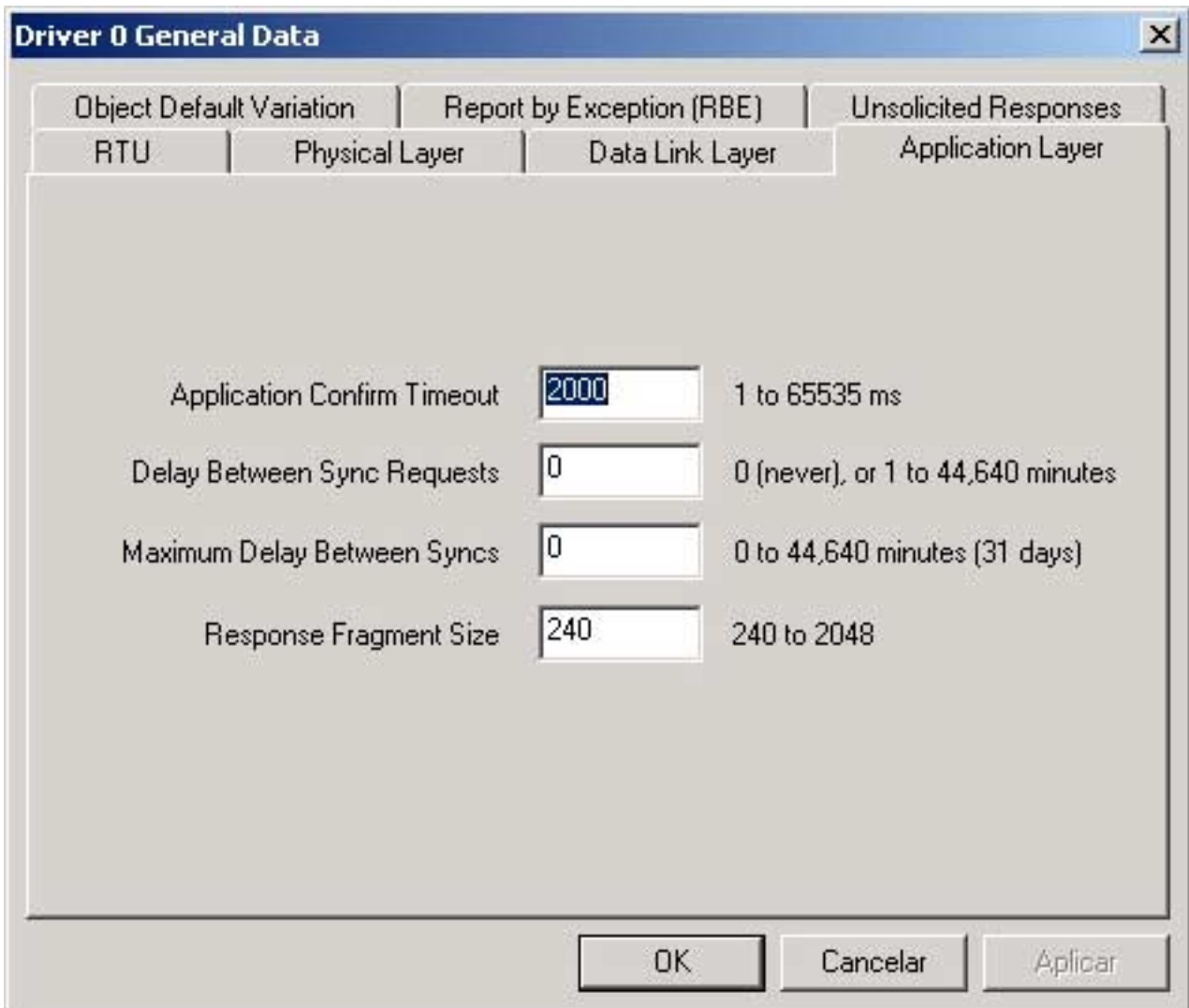


Figura 4-4. Guia Camada de Aplicação

Timeout de Confirmação da Aplicação (Application Confirm Timeout)

É utilizado para especificar, em ms, o timeout que a camada de aplicação irá aguardar por uma confirmação do mestre para uma mensagem anteriormente enviada com solicitação de confirmação. Este valor é independente do Timeout de Confirmação do Enlace. Se a opção de confirmação de enlace está habilitada, o temporizador do Timeout de Confirmação da Aplicação não inicia até que uma confirmação de enlace seja recebida.

Este parâmetro também é utilizado como parte do controle de retransmissão quando uma mensagem não solicitada sem confirmação pode ser regenerada e retransmitida. Veja o parâmetro Período entre Retentativas de Mensagens Não Solicitadas

Este parâmetro pode assumir valores entre 1 até 65535 ms.

Intervalo entre Requisições para Sincronismo (Delay between Sync Requests)

É o intervalo de tempo, em minutos, após o mestre DNP executar um comando de sincronização na remota, no qual o bit de “necessidade de tempo” no campo Internal Indication (IIN) não será ligado mesmo que neste intervalo não seja recebida nenhuma outra mensagem de sincronização. Após este período sem receber comandos de sincronização, este bit será ligado para requisitar ao mestre uma nova sincronização.

Este valor está relacionado com a taxa na qual o relógio interno da remota torna-se dessincronizado com relação ao relógio do mestre. Se este parâmetro for ajustado para **zero**, isto quer dizer que o bit “necessidade de tempo” no campo Internal Indication (IIN) **nunca será ligado**. Isto pode ser aceitável desde que a remota possua uma fonte de tempo precisa, tal como um dispositivo IRIG-B, ou então quando a remota não necessita registrar eventos nos pontos do seu próprio barramento com estampa de tempo. Observe-se que esta última afirmação é válida mesmo que a remota esteja enviando (repassando, de fato) estampas de tempo de eventos recebidos de IED.

Este parâmetro pode assumir os valores

- 0 (nunca requisita sincronização) ou
- 1 até 44640 minutos (31 dias).

Intervalo Máximo para Indicar Fora de Sincronismo (Maximun Delay between Syncs)

Depois de transcorrido o tempo fixado por este parâmetro sem que a remota tenha recebido uma sincronização do mestre, ela assume o estado “fora de sincronismo com o mestre” e indica isto através da posição 33 da tabela de diagnósticos (ver capítulo 5), escrevendo 0 nesta posição. Uma vez neste estado, a UCP (AL-2003 ou AL-2004) é quem assume a responsabilidade de ajustar o relógio do driver do AL-2005.

Quando a remota recebe um comando sincronismo, este temporizador é reinicializado e a posição 33 da tabela de diagnósticos possui valor 1, informando que a remota está em sincronismo com o mestre. Este deve ser o valor normal desta posição quando a remota é sincronizada pelo mestre.

Se este parâmetro for ajustado para zero, isto quer dizer que o estado “fora de sincronismo com o mestre” nunca será assumido. Este é o valor que deve ser utilizado neste parâmetro quando a remota é ajustada por um GPS em vez de ser ajustada pelo mestre.

Este parâmetro pode assumir os valores

- 0 (nunca assume nem indica o estado “fora de sincronismo com o mestre”) ou
- 1 até 44640 minutos (31 dias).

Tamanho dos Fragmentos de Resposta (Response Fragment Size)

Esta informação é utilizada quando a remota necessita fragmentar sua resposta de aplicação. Informa em quantos bytes, no máximo, cada fragmento da aplicação deve conter.

Este parâmetro é útil quando o buffer de mensagens da aplicação da estação que receberá a resposta possui tamanho limitado. Neste caso este valor deve ser configurado de acordo com este limite.

Este parâmetro deve ser configurado entre 240 e 2048 bytes.

Guia de Variação Default dos Objetos (Object Default Variation)

Nesta guia são especificadas as variações default para todos os objetos. Estas variações são utilizadas quando não estiver definido, por outros meios, a variação dos objetos contidos em mensagens transmitidas pela remota.

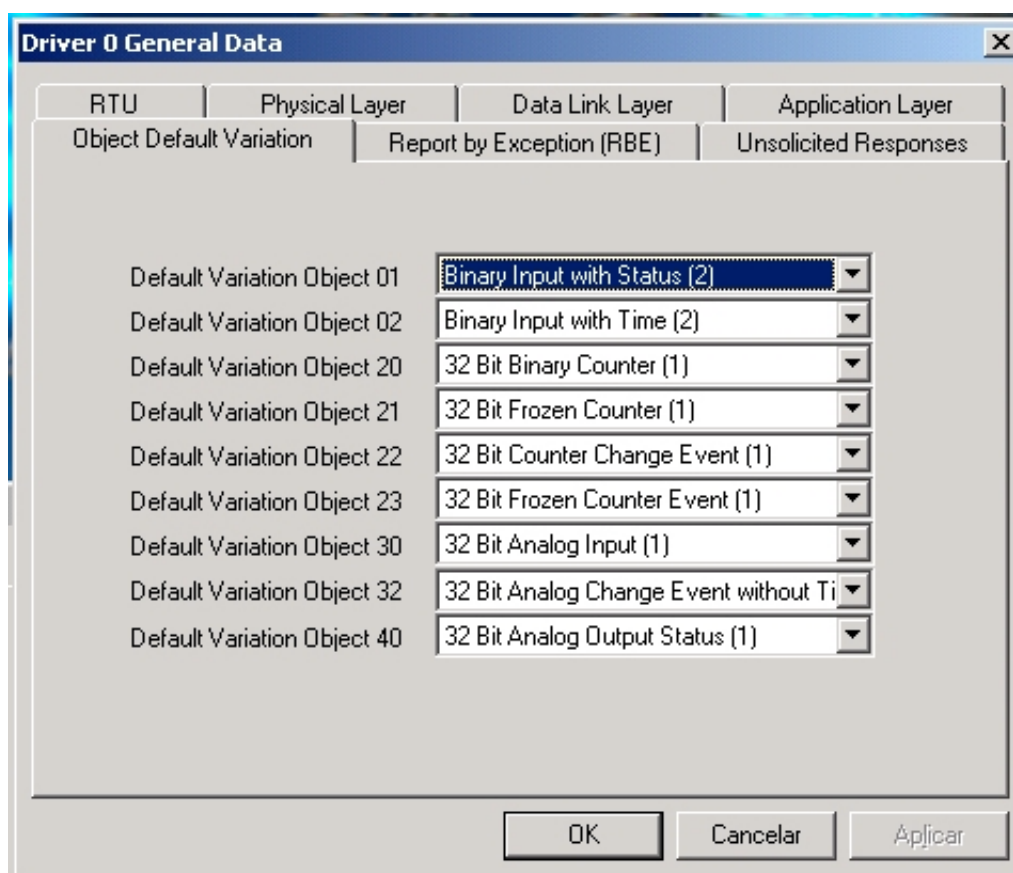


Figura 4-5. Guia de Variação Default dos Objetos

Variação Default para Objetos 01 (Default Variation Object 01)

Especifica a variação default para todos os objetos 01 (entradas binárias). A variação default é utilizada sempre que a variação 0 for requisitada ou então em respostas a polling de classes.

Este parâmetro pode assumir os valores

- 1: entradas binárias sem status
- 2: entradas binárias com status

Variação Default para Objetos 02 (Default Variation Object 02)

Especifica a variação default para todos os objetos 02 (eventos de mudanças em entradas binárias). A variação default é utilizada sempre que a variação 0 for requisitada ou então em respostas a polling de classes.

Este parâmetro pode assumir os valores

- 1: mudança em entradas binárias sem estampa de tempo
- 2: mudança em entradas binárias com estampa de tempo

Variação Default para Objetos 20 (Default Variation Object 20)

Especifica a variação default para todos os objetos 20 (contadores binários). A variação default é utilizada sempre que a variação 0 for requisitada ou então em respostas a polling de classes.

Este parâmetro pode assumir os valores

- 1: contadores binários de 32 bits com status
- 2: contadores binários de 16 bits com status
- 5: contadores binários de 32 bits sem status
- 6: contadores binários de 16 bits sem status

Variação Default para Objetos 21 (Default Variation Object 21)

Especifica a variação default para todos os objetos 21 (contadores congelados). A variação default é utilizada sempre que a variação 0 for requisitada ou então em respostas a polling de classes.

Este parâmetro pode assumir os valores

- 1: contadores congelados de 32 bits com status
- 2: contadores congelados de 16 bits com status
- 9: contadores congelados de 32 bits sem status
- 10: contadores congelados de 16 bits sem status

Variação Default para Objetos 22 (Default Variation Object 22)

Especifica a variação default para todos os objetos 22 (eventos de mudança em contadores binários). A variação default é utilizada sempre que a variação 0 for requisitada ou então em respostas a polling de classes.

Este parâmetro pode assumir os valores

- 1: mudança em contadores binários de 32 bits sem estampa de tempo
- 2: mudança em contadores binários de 16 bits sem estampa de tempo
- 5: mudança em contadores binários de 32 bits com estampa de tempo
- 6: mudança em contadores binários de 16 bits com estampa de tempo

Variação Default para Objetos 23 (Default Variation Object 23)

Especifica a variação default para todos os objetos 23 (eventos de mudança em contadores congelados). A variação default é utilizada sempre que a variação 0 for requisitada ou então em respostas a polling de classes.

Este parâmetro pode assumir os valores

- 1: mudança em contadores congelados de 32 bits sem estampa de tempo
- 2: mudança em contadores congelados de 16 bits sem estampa de tempo
- 5: mudança em contadores congelados de 32 bits com estampa de tempo
- 6: mudança em contadores congelados de 16 bits com estampa de tempo

Variação Default para Objetos 30 (Default Variation Object 30)

Especifica a variação default para todos os objetos 30 (entradas analógicas). A variação default é utilizada sempre que a variação 0 for requisitada ou então em respostas a polling de classes.

Este parâmetro pode assumir os valores

- 1: entradas analógicas de 32 bits com status
- 2: entradas analógicas de 16 bits com status
- 3: entradas analógicas de 32 bits sem status
- 4: entradas analógicas de 16 bits sem status

Variação Default para Objetos 32 (Default Variation Object 32)

Especifica a variação default para todos os objetos 32 (evento de mudança em entradas analógicas). A variação default é utilizada sempre que a variação 0 for requisitada ou então em respostas a polling de classes.

Este parâmetro pode assumir os valores

- 1: mudança em entradas analógicas de 32 bits sem estampa de tempo
- 2: mudança em entradas analógicas de 16 bits sem estampa de tempo
- 3: mudança em entradas analógicas de 32 bits com estampa de tempo
- 4: mudança em entradas analógicas de 16 bits com estampa de tempo

Variação Default para Objetos 40 (Default Variation Object 40)

Especifica a variação default para todos os objetos 40 (estado de saídas analógicas). A variação default é utilizada sempre que a variação 0 for requisitada ou então em respostas a polling de classes.

Este parâmetro pode assumir os valores

- 1: estado das saídas analógicas de 32 bits
- 2: estado das saídas analógicas de 16 bits

Guia de Envio por Exceção (*Report By Exception - RBE*)

Nesta guia estão as definições para a camada de enlace do protocolo.

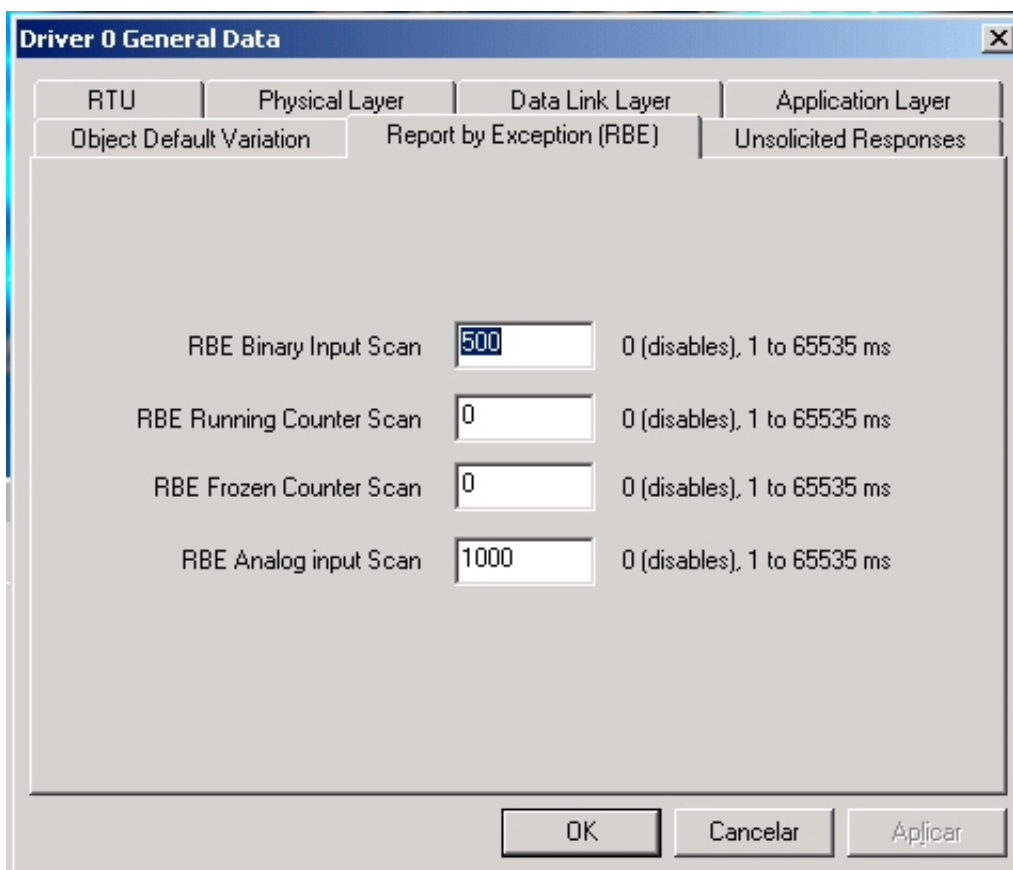


Figura 4-6. Guia de Envio por Exceção

Período de Varredura de Entradas Binárias (RBE Binary Input Scan Period)

Intervalo de tempo, em ms, entre varreduras para detectar eventos de mudanças em entradas binárias (**Report By Exception**). Quando ajustado para zero, as varreduras automáticas para detecção de mudanças em entradas binárias estará desabilitada. Se este valor for muito pequeno, as varreduras ocorrerão tão frequentes quanto for possível, podendo se constituir numa maneira ineficiente de utilizar os recursos do sistema ou até mesmo sobrecarregar o processamento da remota. Se este valor for muito grande, algumas mudanças rápidas em entradas binárias podem não ser detectadas.

Este parâmetro pode assumir valores entre 0 (sem varredura automática) e 65535 ms.

Este parâmetro é irrelevante para a detecção e transmissão de eventos de IED, pois para estes casos o mecanismo utilizado não depende da varredura aqui ajustada.

A partir da versão 1.13.6 do driver AL-2741, este parâmetro pode ser deixado em zero quando todas as mudanças nos pontos de entradas binárias são detectadas pelos módulos AL-313x, pois a transferência deste tipo de evento ocorre a cada varredura do AL-2003 ou AL-2004 de forma otimizada. Portanto, deve ser utilizado apenas no caso de detecção de eventos em entradas binárias através do AL-2005.

Período de Varredura de Contadores (RBE Running Counter Scan Period)

Intervalo de tempo, em ms, entre varreduras para detectar eventos de mudanças em contadores (**Report By Exception**). Quando ajustado para zero, as varreduras automáticas para detecção de mudanças em contadores estará desabilitada. Se este valor for muito pequeno, as varreduras ocorrerão tão freqüentes quanto for possível, podendo se constituir numa maneira ineficiente de utilizar os recursos do sistema ou até mesmo sobrecarregar o processamento da remota. Se este valor for muito grande, algumas mudanças rápidas em contadores podem não ser detectadas.

Este parâmetro pode assumir valores entre 0 (sem varredura automática) e 65535 ms.

Este parâmetro é irrelevante para a detecção e transmissão de eventos de IED, pois para estes casos o mecanismo utilizado não depende da varredura aqui ajustada.

Período de Varredura de Contadores Congelados (RBE Frozen Counter Scan Period)

Intervalo de tempo, em ms, entre varreduras para detectar eventos de mudanças em contadores congelados (**Report By Exception**). Quando ajustado para zero, as varreduras automáticas para detecção de mudanças em contadores congelados estará desabilitada. Se este valor for muito pequeno, as varreduras ocorrerão tão freqüentes quanto for possível, podendo se constituir numa maneira ineficiente de utilizar os recursos do sistema ou até mesmo sobrecarregar o processamento da remota. Se este valor for muito grande, algumas mudanças rápidas em contadores congelados podem não ser detectadas.

Este parâmetro pode assumir valores entre 0 (sem varredura automática) e 65535 ms.

Este parâmetro é irrelevante para a detecção e transmissão de eventos de IED, pois para estes casos o mecanismo utilizado não depende da varredura aqui ajustada.

Período de Varredura de Entradas Analógicas (RBE Analog Input Scan Period)

Intervalo de tempo, em ms, entre varreduras para detectar eventos de mudanças em entradas analógicas (**Report By Exception**). Quando ajustado para zero, as varreduras automáticas para detecção de mudanças em entradas analógicas estará desabilitada. Se este valor for muito pequeno, as varreduras ocorrerão tão freqüentes quanto for possível, podendo se constituir numa maneira ineficiente de utilizar os recursos do sistema ou até mesmo sobrecarregar o processamento da remota. Se este valor for muito grande, algumas mudanças rápidas em entradas analógicas podem não ser detectadas.

Este parâmetro pode assumir valores entre 0 (sem varredura automática) e 65535 ms.

Este parâmetro é irrelevante para a detecção e transmissão de eventos de IED, pois para estes casos o mecanismo utilizado não depende da varredura aqui ajustada.

Guia de Respostas Não Solicitadas (*Unsolicited Responses*)

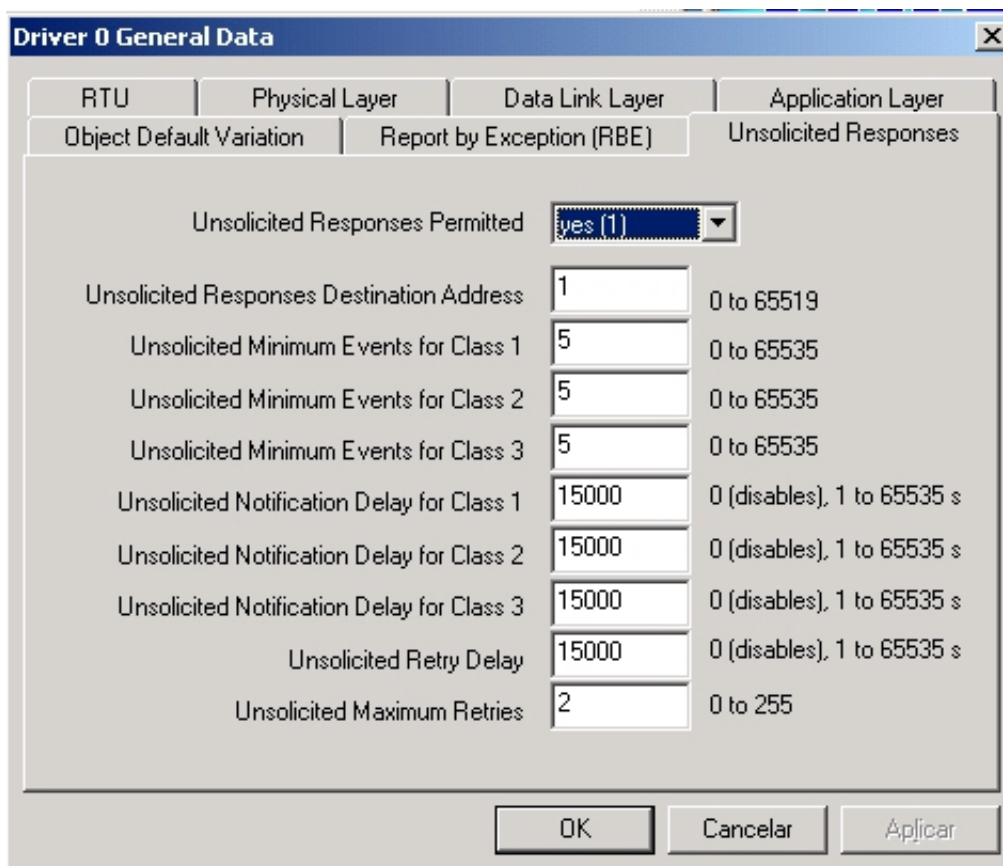


Figura 4-7. Guia de Respostas Não Solicitadas

Habilitação de Respostas Não Solicitadas (*Unsolicited Responses Permitted*)

Se este parâmetro estiver habilitado (**yes**) a remota poderá enviar mensagens não solicitadas para o mestre. Caso contrário (**no**) a só envia mensagens mediante requisição do mestre.

Para o primeiro caso, a remota transmitirá inicialmente uma mensagem não solicitada *nula*, e continuará enviado-a até que seja recebida uma confirmação no nível da camada de aplicação enviada pelo mestre. A partir de então, o mestre DNP deverá enviar uma mensagem de “habilitação de não solicitadas” para uma ou mais das três classes de eventos de dados (classes 1, 2 ou 3) antes que qualquer outra mensagem não solicitada seja enviada pela remota.

Para o segundo caso, mensagens de “habilitação de não solicitadas” e de “desabilitação de não solicitadas” do mestre DNP não serão aceitas e uma resposta com a indicação do bit “**BAD FUNCTION**” Internal Indication (IIN) será enviada como resposta para estas mensagens.

Neste caso todos os demais parâmetros estarão com sua edição desabilitada.

Endereço Destino para Respostas Não Solicitadas (*Unsolicited Responses Destination Address*)

Especifica o endereço destino na camada de aplicação para as mensagens não solicitadas.

Tipicamente é o endereço do mestre de uma comunicação DNP. Todas as mensagens não solicitadas, inclusive as mensagens de inicialização, serão transmitidas para este endereço.

Este parâmetro pode assumir valores entre 0 e 65519.

Mínimo de Eventos para Transmissão da Classe 1 (*Unsolicited Minimum Events for Class 1*)

Este parâmetro é utilizado para designar em que condições a remota enviará mensagens não solicitadas. Se a quantidade de eventos de classe 1 que ocorreram desde a última transmissão ultrapassar este valor, então a partir deste momento, e desde que o mestre tenha habilitado as mensagens não solicitadas para esta classe, a remota irá transmitir-los através de mensagens não solicitadas.

Este parâmetro pode assumir valores entre 0 e 65535.

Esta não é a única condição de disparo de mensagens não solicitadas. Ver parâmetro “Período Mínimo para Transmissão da Classe 1”.

Mínimo de Eventos para Transmissão da Classe 2 (Unsolicited Minimum Events for Class 2)

Similar ao parâmetro “Mínimo de Eventos para Transmissão da Classe 1”, mas refere-se a classe 2.

Mínimo de Eventos para Transmissão da Classe 3 (Unsolicited Minimum Events for Class 3)

Similar ao parâmetro “Mínimo de Eventos para Transmissão da Classe 1”, mas refere-se a classe 3.

Período Mínimo para Transmissão da Classe 1 (Unsolicited Notification Delay for Class 1)

Este parâmetro é utilizado para designar em que condições a remota enviará mensagens não solicitadas. Se o tempo transcorrido após a geração de algum evento de classe 1 for igual ou maior que ele, a remota irá transmitir os eventos desta classe através de mensagens não solicitadas, desde que o mestre tenha habilitado as mensagens não solicitadas para esta classe.

Se este parâmetro for zero, então não haverá envio por estouro de tempo para esta classe, e a única maneira da remota enviar uma mensagem não solicitada desta classe é através da superação do parâmetro “Mínimo de Eventos para Transmissão da Classe 1” .

Este parâmetro pode assumir os valores :

- 0 (quer dizer que não haverá envio por estouro de tempo para esta classe) ou
- 1 até 65535 segundos.

Esta não é a única condição de disparo de mensagens não solicitadas. Ver parâmetro “Mínimo de Eventos para Transmissão da Classe 1”.

Período Mínimo para Transmissão da Classe 2 (Unsolicited Notification Delay for Class 2)

Similar ao parâmetro “Período Mínimo para Transmissão da Classe 1”, mas refere-se a classe 2.

Período Mínimo para Transmissão da Classe 3 (Unsolicited Notification Delay for Class 3)

Similar ao parâmetro “Período Mínimo para Transmissão da Classe 1”, mas refere-se a classe 3.

Período entre Retentativas de Mensagens Não Solicitadas (Unsolicited Retry Delay)

Especifica o período mínimo que deve transcorrer entre a falha no recebimento de confirmação de uma mensagem não solicitada e a retransmissão desta mensagem.

Se o temporizador de timeout definido pelo parâmetro “Timeout de Confirmação da Aplicação” esgotar antes que a remota receba a confirmação esperada, este parâmetro define quando a próxima retentativa de mensagem não solicitada será enviada. Se este parâmetro for zero ou menor do que o “Timeout de Confirmação da Aplicação”, a retransmissão ocorrerá tão logo encerre-se a contagem deste temporizador.

Enquanto a remota estiver aguardando a confirmação de uma mensagem não solicitada, por um período definido por “Timeout de Confirmação da Aplicação”, qualquer solicitação de leitura recebida do mestre será postergada até que o referido período tenha transcorrido. Por outro lado, se durante este mesmo período a remota receber mais de uma solicitação de leitura, apenas a última delas será postergada e as demais serão totalmente ignoradas.

Este parâmetro pode assumir valores entre 0 e 65535 ms.

Caso seu valor seja 0, a retransmissão ocorrerá tão logo encerre-se a contagem do tempo “Timeout de Confirmação da Aplicação”.

Número de Retentativas de Mensagens Não Solicitadas (Unsolicited Maximum Retries)

É o número máximo de vezes que a remota irá retransmitir uma mensagem não solicitada cuja confirmação do mestre não tenha sido recebida. O período entre estas retentativas é definido por o “Período entre Retentativas de Mensagens Não Solicitadas”.

Após esgotar-se esta quantidade de retentativas, outras irão ocorrer mas com um intervalo dez vezes maior do que o anterior entre duas mensagens sem confirmação de recebimento do mestre.

Se este parâmetro for configurado com 255, um número ilimitado de retentativas, com o “Período entre Retentativas de Mensagens Não Solicitadas” entre uma e outra, irá ocorrer até que haja uma confirmação de recebimento por parte do mestre.

Tela de Pontos de E/S da Remota (I/O Points)

Na Tela Pontos de E/S (**I/O Points**) descrito no manual da remota HD3001, são definidos grupos de pontos pertencentes aos módulos de E/S do barramento da remota. Cada linha desta tela corresponde a um grupo, que possui características independentes do driver escravo utilizado.

Já na tela que está sendo descrita agora, os mesmos grupos definidos anteriormente são mantidos. Observe-se que não é possível criar novos grupos a partir desta tela, pois os grupos são definidos na Tela Pontos de E/S (**I/O Points**) e apenas reapresentados aqui. As colunas definidas na outra tela estão bloqueadas para edição nesta.

Por outro lado, esta tela adiciona novas colunas para edição de novos parâmetros para cada grupo existente, que são dependentes do driver utilizado, no presente caso o AL-2741, que implementa o protocolo DNP V3.00.

Vale relembrar também que para cada driver escravo instalado, e podem existir até quatro numa remota HD3001, existe uma Tela de Pontos de E/S da Remota (**I/O Points**).

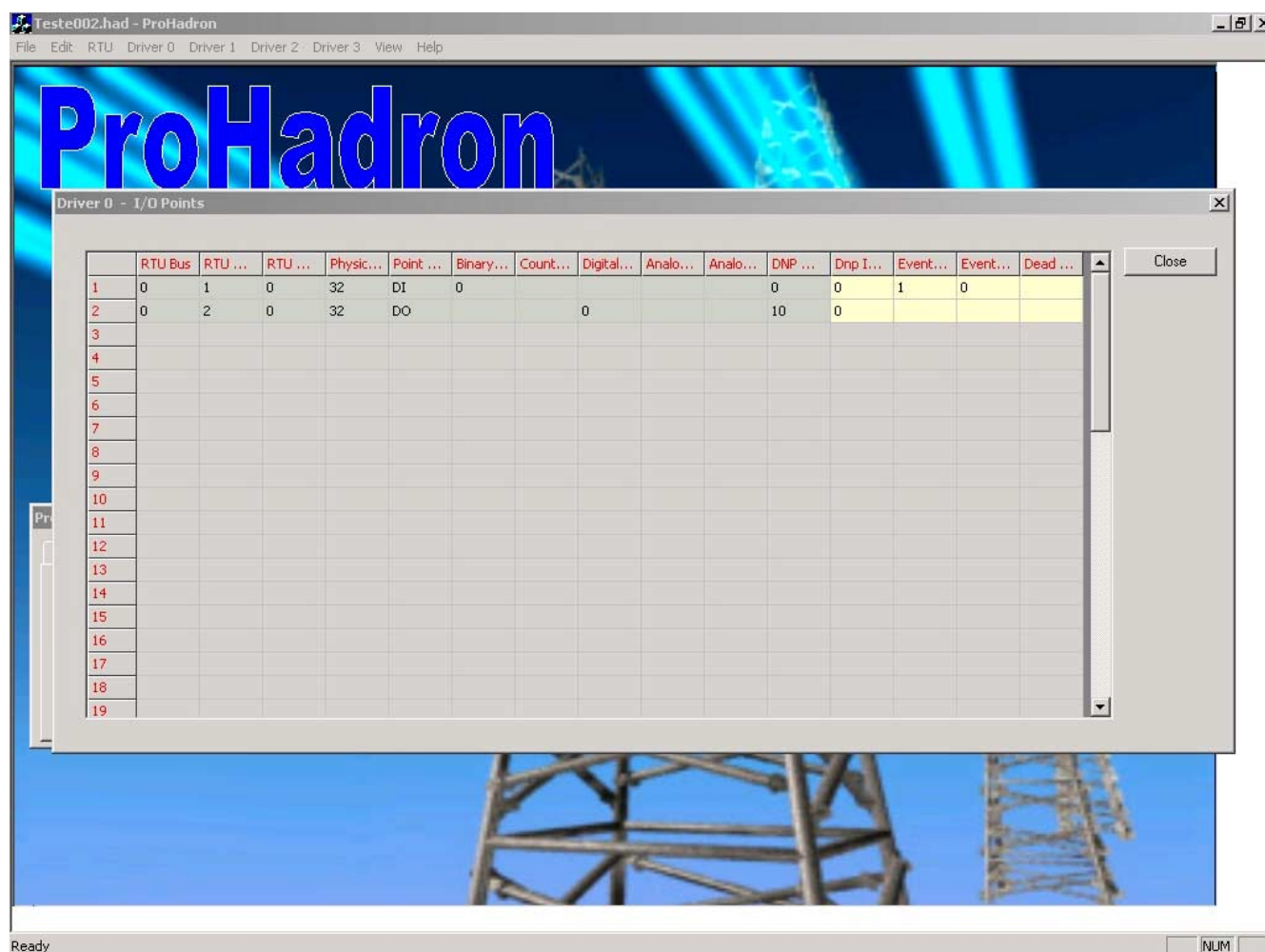


Figura 4-8. Tela de Pontos de E/S da Remota

Coluna Grupo de Objetos DNP (DNP Object Grouping)

Esta coluna não é editável, pois ela é calculada automaticamente das colunas anteriores. Ela é utilizada para fins de documentação, informando a qual Grupo de Objetos DNP pertence o grupo de pontos definido nesta linha.

Coluna Índice DNP (DNP Index)

Esta coluna define o índice DNP do primeiro objeto do grupo. Não pode haver sobreposição de índices entre dois objetos do mesmo grupo de objetos DNP. Por outro lado, esta limitação não existe entre pontos de diferentes grupos de objetos DNP.

Pode assumir valores entre 0 e 65535.

Algumas vezes um índice DNP pode corresponder a mais de um ponto físico de E/S. Para saídas digitais do tipo trip/close por exemplo, dois pontos físicos possuem o mesmo índice.

Coluna Classe de Eventos (Event Class)

Esta coluna define em qual classe os eventos gerados serão enviados. Caso os eventos não devam ser reportados, deve-se atribuir a classe 0.

A recomendação de atribuição para cada classe é a seguinte:

- Classe 1: eventos de entradas binárias
- Classe 2: eventos de entradas analógicas
- Classe 3: eventos de contadores

Pode assumir valores entre 0 e 3.

Coluna Método para Geração de Eventos (Event Method)

Quando o objeto DNP associado ao grupo de pontos for do tipo evento, define o método pelo qual novos eventos serão reconhecidos e como serão enfileirados.

- 0: utilizado para **entradas binárias** (única opção para este tipo de ponto).
Todas as mudanças no ponto (0 para 1 e 1 para 0) são detectadas pelo driver e enfileiradas como eventos para posterior transmissão (ver parâmetros para envio de eventos para as classes 1, 2 e 3). Quando o mestre requisita os eventos da classe nos quais os eventos destes pontos estão enfileirados, poderão ser enviado vários eventos de um mesmo ponto, na mesma requisição.
- 1: utilizado para **entradas analógicas, contadores e contadores congelados**.
Todas as mudanças no ponto (que excedam a banda morta) são detectadas pelo driver e enfileiradas como eventos para posterior transmissão (ver parâmetros para envio de eventos para as classes 1, 2 e 3).
Quando o mestre requisita os eventos da classe nos quais os eventos destes pontos estão enfileirados, poderão ser enviado vários eventos de um mesmo ponto, na mesma requisição.
- 2: utilizado para **entradas analógicas, contadores e contadores congelados**.
Todas as mudanças no ponto (que excedam a banda morta) são detectadas pelo driver, mas apenas o último evento ocorrido para um ponto é enfileirado para posterior transmissão (ver parâmetros para envio de eventos para as classes 1, 2 e 3).
Quando o mestre requisita os eventos da classe nos quais os eventos destes pontos estão enfileirados, ele receberá no máximo um evento de cada ponto na mesma requisição, caso existam. Eles serão reportados com o valor que possuíam quando ocorreu a requisição de polling da classe de dados, e o tempo informado, para aqueles que possuem estampa de tempo, também será o tempo em que ocorreu a requisição de polling da classe de dados, e não o tempo no qual o evento realmente ocorreu.

Coluna Banda Morta (Dead Band)

Utilizado para detectar eventos em **entradas analógicas, contadores e contadores congelados**. O mesmo parâmetro válido para um **contador** também é válido para o seu respectivo **contador congelado**.

A cada varredura (ver parâmetros RBE) a remota avalia a diferença absoluta entre o último valor amostrado de uma entrada e o último valor enviado para o mestre daquela entrada (valor de referência). Este valor de referência é zerado na inicialização.

Se o valor absoluto daquela diferença é **maior** do que a banda morta daquela entrada, então:

- um evento é detectado para aquele ponto;
- o valor de referência é igualado ao último valor lido na varredura da variável.

Pode assumir valores entre 0 e 32767.

Parâmetro válido apenas para eventos detectados no driver AL-2741 executado no coprocessador AL-2005.

Tela de Pontos de E/S de IED (IED I/O Points)

Na Tela Pontos de IED (**IED I/O Points**) descrito no manual da remota HD3001, são definidos grupos de pontos pertencentes A IED conectados à remota Hadron. Cada linha desta tela corresponde a um grupo, que possui características independentes do driver escravo utilizado.

Já na tela que está sendo descrita agora, os mesmos grupos definidos anteriormente são mantidos. Observe-se que não é possível criar novos grupos a partir desta tela, pois os grupos são definidos na Tela Pontos de IED (**IED I/O Points**) e apenas reapresentados aqui. As colunas definidas na outra tela estão bloqueadas para edição nesta.

Por outro lado, esta tela adiciona novas colunas para edição de novos parâmetros para cada grupo existente, que são dependentes do driver utilizado, no presente caso o AL-2741, que implementa o protocolo DNP V3.00.

Vale lembrar também que para cada driver escravo instalado, e podem existir até quatro numa remota HD3001, existe uma Tela de Pontos de E/S da Remota (**IO Points**).

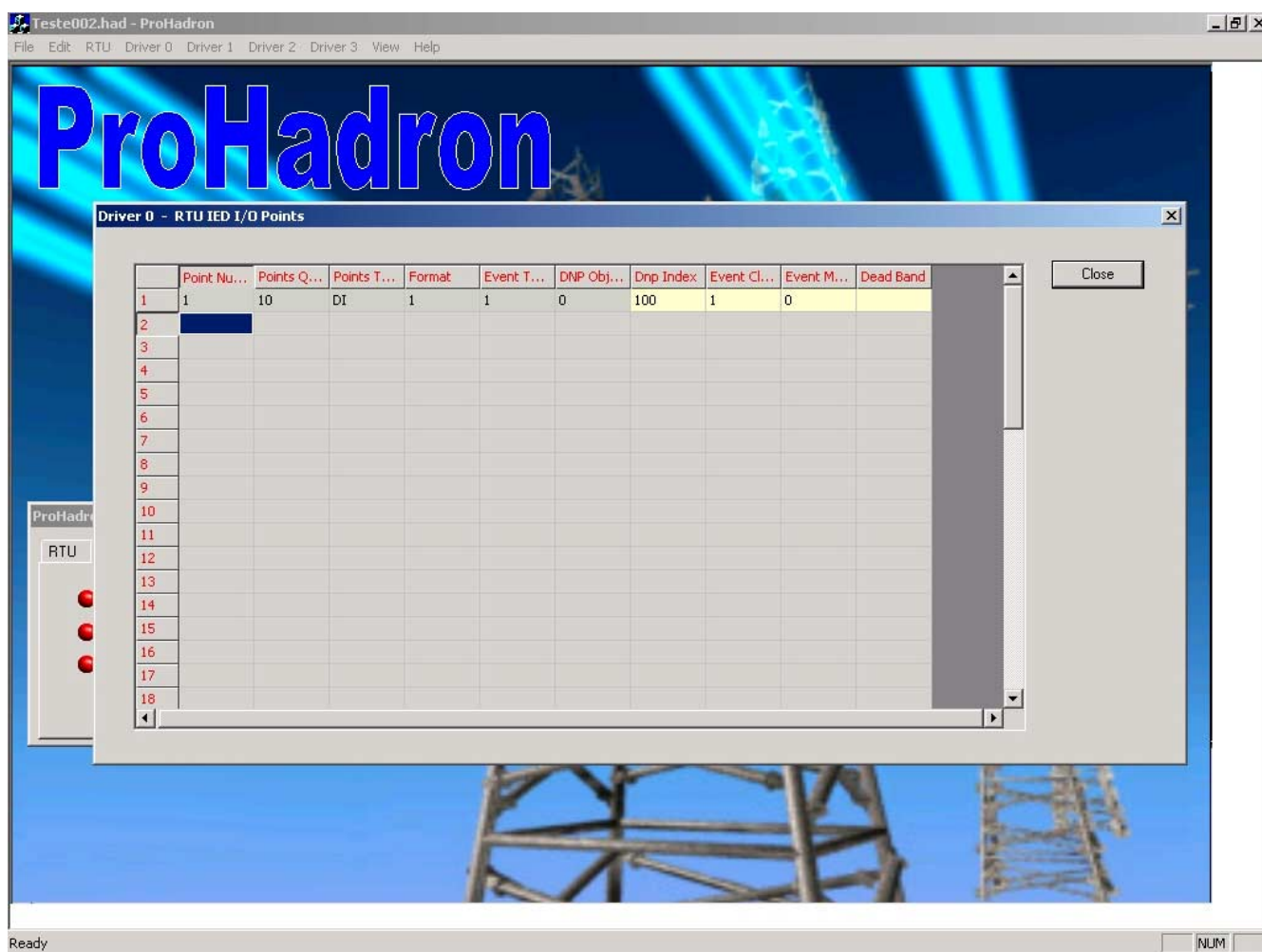


Figura 4-9. Tela de Pontos de E/S de IED

Coluna Grupo de Objetos DNP (DNP Object Grouping)

Esta coluna não é editável, pois ela é calculada automaticamente das colunas anteriores. Ela é utilizada para fins de documentação, informando a qual Grupo de Objetos DNP pertence o grupo de pontos definido nesta linha.

Coluna Índice DNP (DNP Index)

Esta coluna define o índice DNP do primeiro objeto do grupo. Não pode haver sobreposição de índices entre dois objetos do mesmo grupo de objetos DNP. Por outro lado, esta limitação não existe entre pontos de diferentes grupos de objetos DNP.

Pode assumir valores entre 0 e 65535.

Coluna Classe de Eventos (Event Class)

Esta coluna define em qual classe os eventos gerados serão enviados. Caso os eventos não devam ser reportados, deve-se atribuir a classe 0.

A recomendação de atribuição para cada classe é a seguinte:

- Classe 1: eventos de entradas binárias
- Classe 2: eventos de entradas analógicas
- Classe 3: eventos de contadores

Pode assumir valores entre 0 e 3.

Coluna Método para Geração de Eventos (Event Method)

Quando o objeto DNP associado ao grupo de pontos for do tipo evento, define o método pelo qual novos eventos serão reconhecidos e como serão enfileirados.

- 0: utilizado para **entradas binárias** (única opção para este tipo de ponto).
Todas as mudanças no ponto (0 para 1 e 1 para 0) são detectadas pelo driver e enfileiradas como eventos para posterior transmissão (ver parâmetros para envio de eventos para as classes 1, 2 e 3).
Quando o mestre requisita os eventos da classe nos quais os eventos destes pontos estão enfileirados, poderão ser enviado vários eventos de um mesmo ponto, na mesma requisição.
- 1: utilizado para **entradas analógicas, contadores e contadores congelados**.
Todas as mudanças no ponto (que excedam a banda morta) são detectadas pelo driver e enfileiradas como eventos para posterior transmissão (ver parâmetros para envio de eventos para as classes 1, 2 e 3).
Quando o mestre requisita os eventos da classe nos quais os eventos destes pontos estão enfileirados, poderão ser enviado vários eventos de um mesmo ponto, na mesma requisição.
- 2: utilizado para **entradas analógicas, contadores e contadores congelados**.
Todas as mudanças no ponto (que excedam a banda morta) são detectadas pelo driver, mas apenas o último evento ocorrido para um ponto é enfileirado para posterior transmissão (ver parâmetros para envio de eventos para as classes 1, 2 e 3).
Quando o mestre requisita os eventos da classe nos quais os eventos destes pontos estão enfileirados, ele receberá no máximo um evento de cada ponto na mesma requisição, caso existam. Eles serão reportados com o valor que possuíam quando ocorreu a requisição de polling da classe de dados, e o tempo informado, para aqueles que possuem estampa de tempo, também será o tempo em que ocorreu a requisição de polling da classe de dados, e não o tempo no qual o evento realmente ocorreu.

Coluna Banda Morta (Dead Band)

Utilizado para detectar eventos em **entradas analógicas, contadores e contadores congelados**. O mesmo parâmetro válido para um **contador** também é válido para o seu respectivo **contador congelado**.

A cada varredura (ver parâmetros RBE) a remota avalia a diferença absoluta entre o último valor amostrado de uma entrada e o último valor enviado para o mestre daquela entrada (valor de referência). Este valor de referência é zerado na inicialização.

Se o valor absoluto daquela diferença é **maior** do que a banda morta daquela entrada, então:

- um evento é detectado para aquele ponto;
- o valor de referência é igualado ao último valor lido na varredura da variável.

Pode assumir valores entre 0 e 32767.

Parâmetro válido apenas para eventos detectados no driver AL-2741 executado no coprocessador AL-2005.

5. Diagnósticos

A configuração da remota Hadron feita através do configurador ProHadron aloca uma série de operandos da UCP AL-2003 ou AL-2004. Neste capítulo, alguns destes operandos tabelas são descritos para referência para solução de problemas.

Tabela de Configuração

Esta tabela normalmente não necessita ser editada. Pode ser consultada para confirmação de alguns parâmetros de configuração que nela são armazenados.

Posição	Descrição
0	Reserva
1	Tipo do modulo processador
2	Canal de comunicação do modulo processador
3	Tipo do driver
4	Identificador de compatibilidade com o gerador de ladder.
5	Modo de operação
6	Endereço DNP da HADRON neste canal
7	Numero de chamadas F-2005.016 por ciclo do CP
8	Velocidade do canal de comunicação
9	Paridade do canal de comunicação
10	Número de stop bits
11	Sinais de modem
12	Modo de confirmação da comunicação
13	Numero de tentativas de confirmação da comunicação
14	Modo de sincronismo
15	Primeiro octeto de saída
16	Memória de status da configuração
17	Timeout de confirmação de comunicação
18	Timeout de barramento
19	Atraso na transmissão
20	Timeout processamento DNP
21	Timeout de confirmação da aplicação
22	Atraso entre solicitações de sincronismo
23	Variação default objeto 01
24	Variação default objeto 02
25	Variação default objeto 20
26	Variação default objeto 21
27	Variação default objeto 22
28	Variação default objeto 23
29	Variação default objeto 30
30	Variação default objeto 32
31	Variação default objeto 40
32	Período de varredura de entradas binárias RBE
33	Período de varredura de contadores ativos RBE
34	Período de varredura de contadores congelados RBE
35	Período de varredura de entradas analógicas RBE
36	Respostas não solicitadas autorizadas
37	Endereço destino para respostas não solicitadas
38	Mínimo de eventos não solicitados para classe 1
39	Mínimo de eventos não solicitados para classe 2
40	Mínimo de eventos não solicitados para classe 3

41	Atraso para notificação de não solicitadas para classe 1
42	Atraso para notificação de não solicitadas para classe 2
43	Atraso para notificação de não solicitadas para classe 3
44	Atraso para retentativas de não solicitadas
45	Maximo de retentativas de não solicitadas
46	Período de acionamento do modulo E-.018
47	Timeout de seleção dos pontos de saída digital/analógico
48	Maximo de atraso entre comandos de sincronismo
49	Timeout de execução de comandos IED (DO e AO)
50	Número da tabela de eventos
51	Numero da tabela de eventos dos Ieds.
52	Número da tabela de comandos CBO
53	Tamanho da tabela de eventos
54	Reserva
55	Reserva
56	Reserva
57	Reserva
58	Buffer de transferência
59	Tamanho do buffer de transferência
60	Buffer de transferência dos Ieds
61	Tamanho do buffer de transferência dos Ieds
62	Imagem de entradas e saídas digitais
63	Tamanho das entradas e saídas digitais
64	Imagem forçamentos
65	Tamanho forçamentos
66	Imagem estados dos módulos
67	Tamanho estados dos módulos
68	Imagem CBO
69	Tamanho CBO
70	Imagem estados AL3202
71	Tamanho estados AL3202
72	Imagem analógicos AL315X
73	Tamanho analógicos AL315X
74	Imagem entradas analógicas comuns
75	Tamanho entradas analógicas comuns
76	Imagem contadores
77	Tamanho contadores
78	Imagem relógio AL-2003
79	Tamanho relógio AL-2003
80	Imagem entradas digitais virtuais
81	Tamanho entradas digitais virtuais
82	Imagem entradas analógicas virtuais
83	Tamanho entradas analógicas virtuais
84	Imagem contadores virtuais
85	Tamanho contadores virtuais
86	Imagem saídas digitais virtuais
87	Tamanho saídas digitais virtuais
88	Imagem saídas analógicas virtuais
89	Tamanho saídas analógicas virtuais
90	Reserva
91	Reserva
92	Reserva
93	Reserva
94	Reserva
95	Reserva

96	Reserva
97	Reserva
98	Reserva
99	Reserva
100	Reserva
101	Reserva
102	Reserva
103	Reserva
104	Reserva
105	Reserva
106	Reserva
107	Reserva
108	Reserva
109	Reserva
110	Indicação de eventos
111	Memória de status do sincronismo
112	Indicação de eventos ieds
113	Reserva
114	Reserva
115	Reserva
116	Reserva
117	Reserva
118	Reserva
119	Reserva
120	Registro de configuração
121	Memória de comandos gerais
122	Reserva
123	Reserva
124	Reserva
125	Reserva
126	Reserva
127	Reserva
128	Reserva
129	Reserva
130	Acerto do relógio AL-2003 ou AL-2004
131	Tamanho acerto do relógio AL-2003 ou AL-2004
132	Comando saídas virtuais (DO, CN e AO)
133	Tamanho comando saídas virtuais (DO, CN e AO)
134	Imagem saídas analógicas.
135	Tamanho saídas analógicas.
136	Ultimo operando memória configurado para os ieds.
137	Reserva
138	Reserva
139	Reserva
140	Reserva
141	Reserva
142	Reserva
143	Reserva
144	Reserva
145	Reserva
146	Reserva
147	Reserva
148	Reserva
149	Reserva
150	Reserva

151	Reserva
152	Reserva
153	Reserva
154	Reserva
155	Reserva
156	Reserva
157	Reserva
158	Reserva
159	Reserva
160	Reserva
161	Reserva
162	Reserva
163	Reserva
164	Reserva
165	Reserva
166	Reserva
167	Reserva
168	Reserva
169	Reserva

Tabela de Diagnósticos

Esta tabela deve ser consultada para obter informações sobre o funcionamento do driver AL-2741.

Posição	Descrição
0	Reserva
1	Indicação de atividade
2	Contagem do numero de chamadas da CHF para configuração
3	Contagem do numero de inicializações do driver da Triangle
4	Estado da inicialização do driver da DNP
5	Tempo sem recepções validas para este endereço
6	Contagem do numero de recepções validas para este endereço
7	Contagem do numero de transmissões por este endereço
8	Reserva
9	Versão do driver
10	Status de leitura/consistência da F-ARQ31.042
11	Quantidade de Binary Inputs configurados
12	Quantidade de Binary Outputs configurados
13	Quantidade de contadores configurados
14	Quantidade de Analog Inputs configurados
15	Quantidade de Analog Outputs configurados
16	Reserva
17	Reserva
18	Reserva
19	Reserva
20	Tempo de ciclo instantâneo do AL-2005 (ms).
21	Tempo de ciclo médio do AL-2005 (ms).
22	Tempo de ciclo Maximo do AL-2005 (ms).
23	Tempo de ciclo mínimo do AL-2005 (ms).
24	Diferença de tempo entre CPU e BIOS, em milisegundos.
25	Informa o numero de posições livres que a lista de eventos contem no momento.
26	Informa o numero de posições livres que a lista de eventos IED contem no momento.
27	Reserva
28	Reserva

29	Reserva
30	Reserva
31	Reserva
32	Resultado da abertura do canal serial
33	Estado de funcionamento do sincronismo
34	Reserva
35	Reserva
36	Reserva
37	Reserva
38	Reserva
39	Reserva
40	Reserva
41	Reserva
42	Reserva
43	Reserva
44	Reserva
45	Reserva
46	Reserva
47	Reserva
48	Reserva
49	Reserva

6. Glossário

Acesso ao meio	Método utilizado por todos os nós de uma rede de comunicação para sincronizar as transmissões de dados e resolver possíveis conflitos de transmissões simultâneas.
Algoritmo	Seqüência finita de instruções bem definidas, objetivando à resolução de problemas.
Arrestor	Dispositivo de proteção contra raios carregado com gás inerte.
Autoclear	Em redes PROFIBUS, é o parâmetro que, quando ativado, muda o estado do mestre para Clear ao ocorrer um erro na rede.
Backoff	Tempo que o nó de uma rede tipo CSMA/CD aguarda antes de voltar a transmitir dados após a ocorrência de colisão no meio físico.
Barramento	Conjunto de sinais elétricos agrupados logicamente com a função de transferir informação e controle entre diferentes elementos de um subsistema.
Baud rate	Taxa com que os bits de informação são transmitidos através de uma interface serial ou rede de comunicação (medido em bits/segundo).
Bit	Unidade básica de informação, podendo estar no estado 0 ou 1.
Bridge (ponte)	Equipamento para conexão de duas redes de comunicação dentro de um mesmo protocolo.
Broadcast	Disseminação simultânea de informação a todos os nós interligados a uma rede de comunicação.
BT	Sigla para teste de bateria em inglês (battery test).
Byte	Unidade de informação composta por oito bits.
Canal serial	Interface de um equipamento que transfere dados no modo serial.
Ciclo de varredura	Uma execução completa do programa aplicativo de um controlador programável.
Circuito de cão de guarda	Circuito eletrônico destinado a verificar a integridade do funcionamento de um equipamento.
Código comercial	Código do produto, formado pelas letras PO, seguidas por quatro números.
Controlador programável	Também chamado de CP. Equipamento que realiza controle sob o comando de um programa aplicativo. É composto de uma UCP, uma fonte de alimentação e uma estrutura de E/S.
CP	Veja controlador programável.
CSMA/CD	Disciplina de acesso ao meio físico, baseada na colisão de dados, utilizada pelas redes Ethernet.
Database	Banco de dados.
Default	Valor predefinido para uma variável, utilizado em caso de não haver definição.
Diagnóstico	Procedimento utilizado para detectar e isolar falhas. É também o conjunto de dados usados para tal determinação, que serve para a análise e correção de problemas.
Download	Carga de programa ou configuração no CP.
E/S	Veja entrada/saída.
E2PROM	Memória não-volátil, que pode ser apagada eletricamente.
EIA RS-485	Padrão industrial (nível físico) para comunicação de dados.
EN 50170	Em redes PROFIBUS, é a norma que define a rede de campo.
Encoder	Transdutor para medidas de posição.
Endereço de módulo	Endereço pelo qual o CP realiza acessos a um determinado módulo de E/S.
Entrada/saída	Também chamado de E/S. Dispositivos de E/S de dados de um sistema. No caso de CPs, correspondem tipicamente a módulos digitais ou analógicos de entrada ou saída que monitoram ou acionam o dispositivo controlado.
EPROM	Significa Erasable Programmable Read Only Memory. É uma memória somente de leitura, apagável e programável. Não perde seu conteúdo quando desenergizada.
ER	Sigla usada para indicar erro nos LEDs.
Escravo	Equipamento ligado a uma rede de comunicação que só transmite dados se for solicitado por outro equipamento denominado mestre.
ESD	Sigla para descarga devida a eletricidade estática em inglês (electrostatic discharge).
Estação de supervisão	Equipamento ligado a uma rede de CPs ou instrumentação com a finalidade de monitorar ou controlar variáveis de um processo.
Flash EPROM	Memória não-volátil, que pode ser apagada eletricamente.
FMS	Sigla para Fieldbus Message System.
Frame	Uma unidade de informação transmitida na rede.
Freeze	Em redes PROFIBUS, é o estado da rede quando os dados das entrada são congelados.
Gateway	Equipamento para a conexão de duas redes de comunicação com diferentes protocolos.
Hardkey	Conector normalmente ligado à interface paralela do microcomputador com a finalidade de impedir a execução de cópias ilegais de um software.
Hardware	Equipamentos físicos usados em processamento de dados onde normalmente são executados programas

	(software).
IEC 1131	Norma genérica para operação e utilização de CPs.
IEC Pub. 144 (1963)	Norma para proteção contra acessos incidentais e vedação contra água, pó ou outros objetos estranhos ao equipamento.
IEC-536-1976	Norma para proteção contra choque elétrico.
IEC-801-4	Norma para testes de imunidade a interferências por trem de pulsos.
IEEE C37.90.1 (SWC)	SWC significa Surge Withstand Capability. Esta norma trata da proteção do equipamento contra ruídos tipo onda oscilatória.
Interface	Dispositivo que adapta elétrica e/ou logicamente a transferência de sinais entre dois equipamentos.
Interrupção	Evento com atendimento prioritário que temporariamente suspende a execução de um programa e desvia para uma rotina de atendimento específica
ISOL.	Sigla usada para indicar isolado ou isolamento.
kbytes	Unidade representativa de quantidade de memória. Representa 1024 bytes.
LED	Sigla para light emitting diode. É um tipo de diodo semicondutor que emite luz quando estimulado por eletricidade. Utilizado como indicador luminoso.
Linguagem Assembly	Linguagem de programação do microprocessador, também conhecida como linguagem de máquina.
Linguagem de programação	Um conjunto de regras e convenções utilizado para a elaboração de um programa.
Linguagem de relés e blocos Altus	Conjunto de instruções e operandos que permitem a edição de um programa aplicativo para ser utilizado em um CP.
Lógica	Matriz gráfica onde são inseridas as instruções de linguagem de um diagrama de relés que compõe um programa aplicativo. Um conjunto de lógicas ordenadas seqüencialmente constitui um módulo de programa.
MasterTool	Identifica o programa Altus para microcomputador, executável em ambiente WINDOWS®, que permite o desenvolvimento de aplicativos para os CPs das séries Ponto, Piccolo, AL-2000, AL-3000 e Quark. Ao longo do manual, este programa é referido pela própria sigla ou como programador MasterTool.
Menu	Conjunto de opções disponíveis e exibidas por um programa no vídeo e que podem ser selecionadas pelo usuário a fim de ativar ou executar uma determinada tarefa.
Mestre	Equipamento ligado a uma rede de comunicação de onde se originam solicitações de comandos para outros equipamentos da rede.
Módulo (referindo-se a hardware)	Elemento básico de um sistema completo que possui funções bem definidas. Normalmente é ligado ao sistema por conectores, podendo ser facilmente substituído.
Módulo (referindo-se a software)	Parte de um programa aplicativo capaz de realizar uma função específica. Pode ser executado independentemente ou em conjunto com outros módulos, trocando informações através da passagem de parâmetros.
Módulo C	Veja módulo de configuração.
Módulo de configuração	Também chamado de módulo C. É um módulo único em um programa de CP que contém diversos parâmetros necessários ao funcionamento do controlador, tais como a quantidade de operandos e a disposição dos módulos de E/S no barramento.
Módulo de E/S	Módulo pertencente ao subsistema de entradas e saídas.
Módulo E	Veja módulo execução.
Módulo execução	Módulo que contém o programa aplicativo, podendo ser de três tipos: E000, E001 e E018. O módulo E000 é executado uma única vez, na energização do CP ou na passagem de programação para execução. O módulo E001 contém o trecho principal do programa que é executado ciclicamente, enquanto que o módulo E018 é acionado por interrupção de tempo.
Módulo F	Veja módulo função.
Módulo função	Módulo de um programa de CP que é chamado a partir do módulo principal (módulo E) ou a partir de outro módulo função ou procedimento, com passagem de parâmetros e retorno de valores. Atua como uma sub-rotina.
Módulo P	Veja módulo procedimento.
Módulo procedimento	Módulo de um programa de CP que é chamado a partir do módulo principal (módulo E) ou a partir de outro módulo procedimento ou função, sem a passagem de parâmetros.
Monomaster	Em redes PROFIBUS, é a rede com apenas um mestre.
Multicast	Disseminação simultânea de informação a um determinado grupo de nós interligados a uma rede de comunicação.
Multimaster	Em redes PROFIBUS, é a rede com mais de um mestre.
Nibble	Unidade de informação composta por quatro bits.
Nó	Qualquer estação de uma rede com capacidade de comunicação utilizando um protocolo estabelecido.
Octeto	Conjunto de oito bits numerados de 0 a 7.
Operandos	Elementos sobre os quais as instruções atuam. Podem representar constantes, variáveis ou um conjunto de variáveis.
PA	Ver pontes de ajuste.
PC	Sigla para programmable controller. É a abreviatura de controlador programável em inglês.
Peer to peer	Tipo de comunicação onde dois nós de uma rede trocam dados e/ou avisos sem depender de um mestre.

Ponte de ajuste	Chave de seleção de endereços ou configuração composta por pinos presentes na placa do circuito e um pequeno conector removível, utilizado para a seleção.
Posta em marcha	Procedimento de depuração final do sistema de controle, quando os programas de todas as estações remotas e UCPs são executados em conjunto, após terem sido desenvolvidos e verificados individualmente.
PROFIBUS PA	Significa protocolo PROFIBUS Process Automation.
Programa aplicativo	É o programa carregado em um CP, que determina o funcionamento de uma máquina ou processo.
Programa executivo	Sistema operacional de um controlador programável. Controla as funções básicas do controlador e a execução de programas aplicativos.
Protocolo	Regras de procedimentos e formatos convencionais que, mediante sinais de controle, permitem o estabelecimento de uma transmissão de dados e a recuperação de erros entre equipamentos.
RAM	Sigla para random access memory. É a memória onde todos os endereços podem ser acessados diretamente de forma aleatória e com a mesma velocidade. É volátil, ou seja, seu conteúdo é perdido quando o equipamento é desenergizado, a menos que se possua uma bateria para a retenção dos valores.
Rede de comunicação	Conjunto de equipamentos (nós) interconectados por canais de comunicação.
Rede de comunicação determinística	Rede de comunicação onde a transmissão e a recepção de informações entre os diversos nós é garantida com um tempo máximo conhecido.
Rede de comunicação mestre-escravo	Rede de comunicação onde as transferências de informações são iniciadas somente a partir de um único nó (mestre da rede) ligado ao barramento de dados. Os demais nós da rede (escravos) apenas respondem quando solicitados.
Rede de comunicação multimestre	Rede de comunicação onde as transferências de informações são iniciadas por qualquer nó ligado ao barramento de dados.
Ripple	Ondulação presente em tensão de alimentação contínua.
RX	Sigla usada para indicar recepção serial.
Sistema redundante	Sistema que contém elementos de reserva ou duplicados para executar determinada tarefa, que podem tolerar determinados tipos de falha sem que execução da tarefa seja comprometida.
Software	Programas de computador, procedimentos e regras relacionadas à operação de um sistema de processamento de dados.
Soquete	Dispositivo no qual se encaixam circuitos integrados ou outros componentes, facilitando a substituição dos mesmos e simplificando a manutenção.
Sub-rede	Segmento de uma rede de comunicação que interliga um grupo de equipamentos (nós) com o objetivo de isolar o tráfego local ou utilizar diferentes protocolos ou meio físicos.
Subsistema de E/S	Conjunto de módulos de E/S digitais ou analógicos e interfaces de um controlador programável.
Tag	Nome associado a um operando ou a uma lógica que permite uma identificação resumida de seu conteúdo.
Time-out	Tempo preestabelecido máximo para que uma comunicação seja completada. Se for excedido procedimentos de retentiva ou diagnóstico serão ativados.
Toggle	Elemento que possui dois estados estáveis, trocados alternadamente a cada ativação.
Token	É uma marca que indica quem é o mestre do barramento no momento.
Troca a quente	Procedimento de substituição de módulos de um sistema sem a necessidade de desenergização do mesmo. Normalmente utilizado em trocas de módulos de E/S.
TX	Sigla usada para indicar transmissão serial.
UCP	Sigla para unidade central de processamento. Controla o fluxo de informações, interpreta e executa as instruções do programa e monitora os dispositivos do sistema.
UCP ativa	Em um sistema redundante, a UCP ativa realiza o controle do sistema, lendo os valores dos pontos de entrada, executando o programa aplicativo e acionando os valores das saídas.
UCP inoperante	É a UCP que não está no estado ativo (controlando o sistema) nem no estado reserva (supervisionando a UCP ativa). Não pode assumir o controle do sistema.
UCP redundante	Corresponde à outra UCP do sistema, como, por exemplo, a UCP2 em relação à UCP1 e vice-versa.
UCP reserva	Em um sistema redundante, é a UCP que supervisiona a UCP ativa, não realizando o controle do sistema, mas estando pronta para assumir o controle em caso de falha na UCP ativa.
Upload	Leitura do programa ou configuração do CP.
Varistor	Dispositivo de proteção contra surto de tensão.
WD	Sigla para cão de guarda em inglês (watchdog). Veja circuito de cão de guarda.
Word	Unidade de informação composta por 16 bits.