

Action.NET

DNP 3.0 - Slave Communication Protocol

Versão 1.0.1

Manual de Referência

00052.01
Outubro, 2013

Action.NET – DNP 3.0
Slave (Server) Communication Protocol

Versão 1.0.1

Manual de Referência

00052.01
outubro, 2013

Copyright 2013[®]
Spin Engenharia de Automação Ltda
Todos os Direitos Reservados

Nenhuma parte deste documento pode ser reproduzida, copiada, fotocopiada, distribuída ou alterada sem a prévia e expressa autorização da Spin Engenharia de Automação Ltda.

NOTA

Em virtude do contínuo desenvolvimento de seus produtos, a informação contida neste documento está sujeita a alterações e/ou modificações sem prévia notificação. A Spin não se considera responsável por erros de digitação ou interpretação das informações aqui contidas; e/ou por danos e prejuízos causados / gerados a terceiros. O conteúdo desta publicação poderá ser alterado a qualquer momento sem que exista a obrigação de notificar qualquer parte envolvida; isto não implicará, em nenhuma hipótese, em alterações, reclamações, ou extensão de O alinhamento vertical dessa página é inferior, para tentar manter o texto sobre as alterações sempre alinhado com a margem inferior.

Nesta página estão exemplificados os estilos para serem utilizados com os ícones de atenção do texto. O estilo é **Atenção – ícones**.



Cuidado! Indica que o usuário deverá proceder exatamente como descrito neste manual, sob pena de danificar ou configurar errado o equipamento.



Dica. Indica informações úteis e rápidas para solução de pequenos problemas.



Perigo! Indica que o usuário deverá proceder exatamente como descrito neste manual, sob risco de choque ou descarga elétrica.

ActionNET – DNP 3.0

Slave Communication Protocol

Sumário

1. INFORMAÇÃO GERAL.....	1
1.1 Sumário	1
1.2 Objetos de dados suportados.....	1
1.3 Funcionamento geral	4
2. CONFIGURAÇÃO DO CANAL.....	5
2.1 Opções de protocolo.....	5
2.2 Settings	5
2.3 Timeout	6
2.4 Estado Inicial.....	6
3. CONFIGURAÇÃO DE “NODES”	7
3.1 Dados dos Canais (primário e backup)	7
4. CONFIGURAÇÃO DE PONTOS	8
4.1 Geral.....	8
4.2 Tipos de pontos	8
4.3 Endereço do ponto.....	11
4.4 Access Type	13

1. INFORMAÇÃO GERAL

1.1 Sumário

Communication Driver Name: DNP3oS

Current Version: 1.0

Implementation DLL: T.ProtocolDriver.DNP3oS.dll

Protocol: DNP3.0 Slave standard protocol

Interface: TCP/IP ou Serial

Description: O driver é utilizado para a comunicação em modo escravo (ou Servidor) com supervisórios ou outros equipamentos que utilizam o protocolo DNP 3.0 Nível 2 em Modo MESTRE (ou Cliente). A comunicação pode ser feita através de canal serial ponto a ponto ou utilizando rede local Ethernet e protocolo TCP-IP como escravo único de um endereço IP.

Masters types supported: Qualquer equipamento em Modo MESTRE compatível DNP 3.0

Communication block size: Maximum 250 bytes, formato FT 1.2

Protocol Options: Modo "LinkConfirm" e endereço da estação mestre.

Multi-threading: Configurável pelo usuário, default é cinco threads para cda nó da rede.

Max number of nodes: user defined

PC Hardware requirements: Standard PC Ethernet interface board, RS485 or RS232 port

PC Software requirements: ActionNET system.

1.2 Objetos de dados suportados

O quadro abaixo apresenta os objetos DNP e suas variantes, suportados por esta implementação.

Objeto			Requisição (Mestre)		Resposta (Escravo)	
Obj.	Var	Descrição	Func. Code (decimal)	Qual. Code (Hex)	Func. Codes (decimal)	Qual. Code (Hex)
1	1	Single Bit Binary Input			129	00
1	2	Binary Input with status			129	00
2	1	Binary Input change without time			129	17
2	2	Binary Input change with time			129	17
12	1	Control relay output block	3,4,5	0x17	129	17
20	0	Binary Counter – all variation				
21	0	Frozen counter – all variations				
30	0	Analog Input – all variations				
30	1	32 Bits Analog Input			129	00
30	2	16 Bit Analog input with flag			129	00

30	3	32 Bits Analog Input without flag			129	00
30	4	16 Bit Analog input without flag			129	00
30	5	Short Floating Point (32bits)				
32	0	Analog Input change event – all variations				
32	1	32 Bits Analog Input change event			129	17
32	2	16 Bit Analog change event without flag			129	17
32	3	32 Bit Analog change event with flag				
32	4	16 Bit Analog change event with flag				
40	2	16 Bit Analog output status			129	00
41	1	32Bit Analog output block	5	0x17	129	17
41	2	16 Bit Analog output block	5	0x17	129	17
50	1	Time and Data	1,2	0x07		
60	1	Class 0 data	1	0x06		
60	2	Class 1 data	1	0x06		
60	3	Class 2 data	1	0x06		
60	4	Class 3 data	1	0x06		
80	1	Internal indications	2	0x06		

Na implementação do protocolo o mestre só executa as requisições destacadas em azul. O equipamento servidor, responde utilizando as respostas destacadas em amarelo. Observe-se que cabe ao equipamento servidor decidir como será a resposta e o mestre deve suportar todas as funções do nível 2 possíveis de serem utilizadas como resposta.

Os objetos, variações de objeto, códigos de função e qualificadores tem seus significados padronizados no DNP. Abaixo são mostradas as tabelas de código de função e qualificador:

Código de Função	Descrição	Origem
1	Ler	Mestre
2	Escrever	Mestre
3	Selecionar	Mestre
4	Operar	Mestre
5	Operar direto (sem seleção)	Mestre

6	Operar direto (sem ack)	Mestre
7	Congela Imediatamente	Mestre
8	Congela Imediatamente (sem ack)	Mestre
9	Congela e limpa	Mestre
10	Congela e lê	Mestre
13	Restart (Cold)	Mestre
14	Restart (Warm)	Mestre
20	Habilita mensagem não solicitada	Mestre
21	Desabilita mensagem não solicitada	Mestre
22	Assinala classe a objeto	Mestre
23	Medida com delay	Mestre
129	Resposta	Escravo
130	Resposta não solicitada (não existe no nível 2)	Escravo

Código de Qualificador	Uso em requisição	Uso em resposta
00,01	Um intervalo de pontos estáticos (classe 0) ou um único ponto com um número	Objeto estático
06	Todos os pontos	Inválido
07,08	Uma quantidade limitada de eventos. Um ponto simples sem número (isto é uma data / hora)	Um ponto simples sem número (isto é uma data / hora)
17, 28	Controles (usualmente um ou mais pontos não relacionados)	Objetos evento (usualmente um ou mais pontos não relacionados)

O DNP tem o conceito de classes de dados, sendo definidas quatro classes:

Classe 0: Corresponde ao ponto estático, analógico ou digital. Seu conteúdo é o valor de uma variável analógica ou digital, de entrada ou saída, em um dado momento;

Classe 1, 2 e 3: Corresponde a eventos de transição de estados ou de variáveis de classe 0 ou situações internas à remota / relé que causam o evento.

O que ocorre normalmente nos IED's ao utilizar-se o DNP, é associar à variação de estado de variáveis digitais ou valores de banda morta de variáveis analógicas as classes 1, 2 e 3. Dessa forma, a modificação do estado / valor dessas variáveis causará eventos que serão transmitidos através do pedido de eventos das respectivas classes (60/2, 60/3 e 60/4). Periodicamente, pode

ser feita uma leitura cíclica para verificação de integridade. Essa leitura, corresponde a um pedido de classe 0 (60/1).

Observações:

- 1) Nesta implementação é considerado automaticamente que variáveis digitais Tipo BI, quando sofrerem alteração serão enviadas como de Classe 1, na forma do Objeto 2 com variação 2 (Binary input with time stamp).
- 2) Nesta implementação é considerado automaticamente que variáveis analógicas AI, AIF, quando sofrerem alteração serão enviadas como de Classe 2., na forma dos Objetos 30 variação 3, para AI e como 30 variação 5 para o AIF.

1.3 Funcionamento geral

A sequência normal de operação do escravo é:

- Caso seja necessário na instalação o mestre deverá enviar mensagem de sincronismo (50,1) periodicamente. A data e horário recebido será entendido como UCT (ou GMT), e será utilizada para alterar o relógio do computador “host”.
- Ao iniciar a execução, nas primeiras mensagens de resposta com, enviará flags de IIN indicando que IED reiniciou (restart). O mestre deverá então executar uma escrita de “clear device flag” (80/1) para limpar esta indicação;
- Sempre que houver alteração do estado de digitais ou analógicas, que sejam configurados na tabela POINTS, com *AccessType* como *ReadWrite*, será enviada mensagem de evento como Classe 1 ou Classe 2. Caso não se desejar eventos em alguns pontos usar para estes *AccessType = Read*;
- Quando o modo escravo recebe um pedido de comando de saída digital ou saída analógica, através software envia a respectiva requisição para o IED (12/1 ou 41/2);

2. CONFIGURAÇÃO DO CANAL

2.1 Opções de protocolo

LinkConfirmMode – Modo do protocolo em que na camada de enlace todas as solicitações recebem resposta de confirmação de seu recebimento. Por default este campo é “enable”.

MasterStationID - Número entre 1 e 65534, unívoco e corresponde ao endereço da estação mestre. Assim, um mestre quando envia mensagem a este escravo deve informar seu endereço. Por default este campo é “65534”.

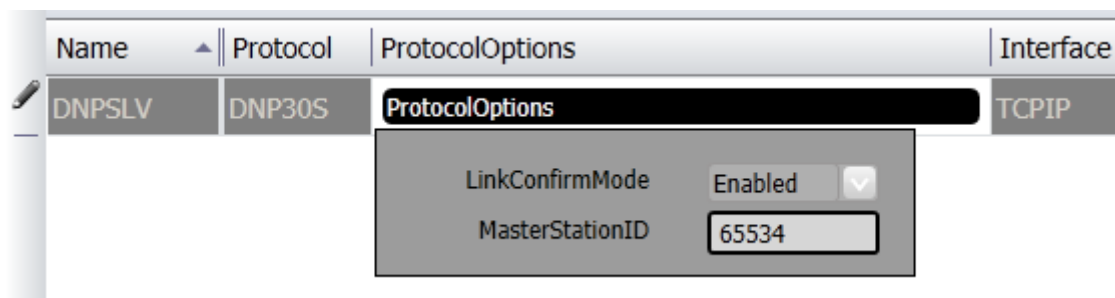


Figura 1 – Opções do Protocolo

2.2 Settings

Canais Serial:

Com: Porta de comunicação do computador utilizada;

BaudRate

DataBits: 8

StopBits: 1ou 2

Parity: None, Even, Odd;

DTR: on, off

RTS: on,off

CTS: on, off;

Configurações da porta devem ser compatíveis no mestre e escravo.

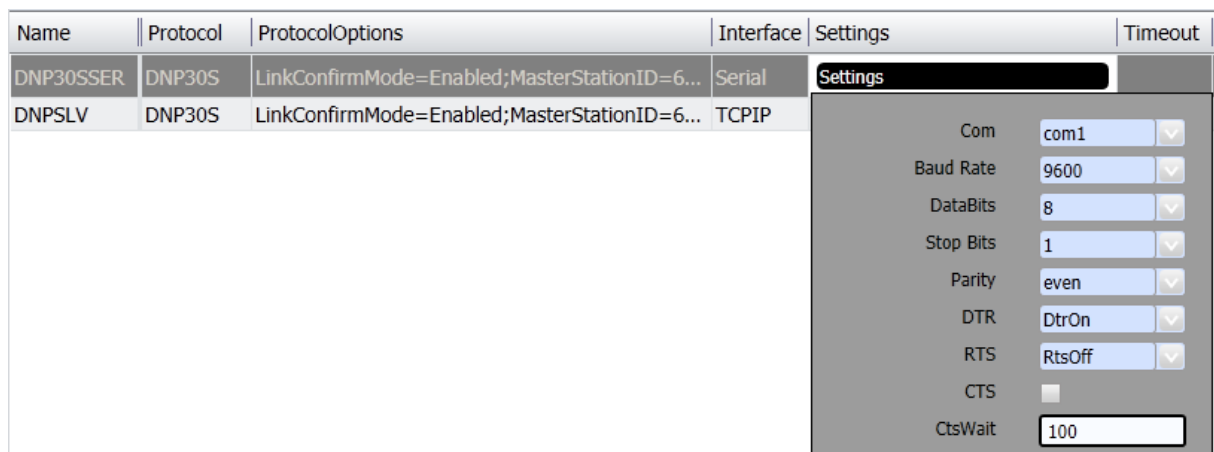


Figura 2 – Settings de Canal Serial

TCP/IP channels:

- **ServerMode e AcceptUnsolicited** são pré marcados e não podem ser alterados;
- **ListeningPort** - Deve-se definir a porta no micro onde este canal escravo vai aceitar conexões.
- **MaxSimultaneousConnections**- Deve ser 1.
- **ShareNodeSameIP**: Aceita IED's com o mesmo IP.

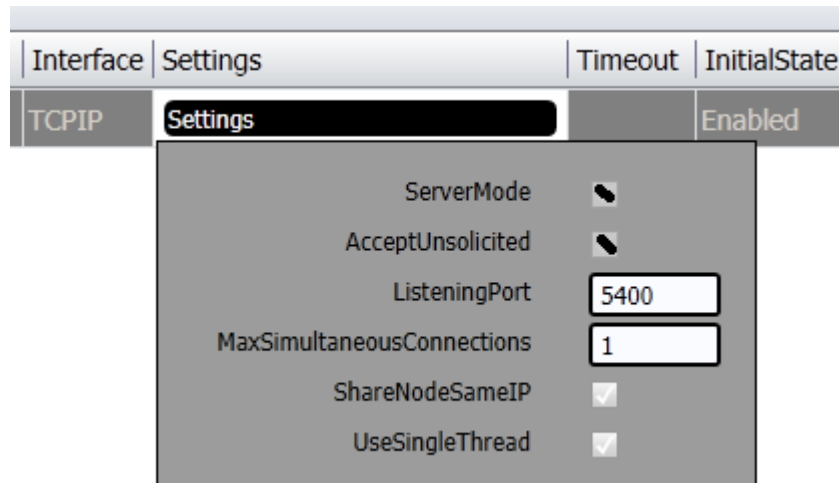


Figura 3 – Settings de TCP/IP

2.3 Timeout

Define tempos limites para a transmissão e recepção de caracteres de mensagens e o número de retries.

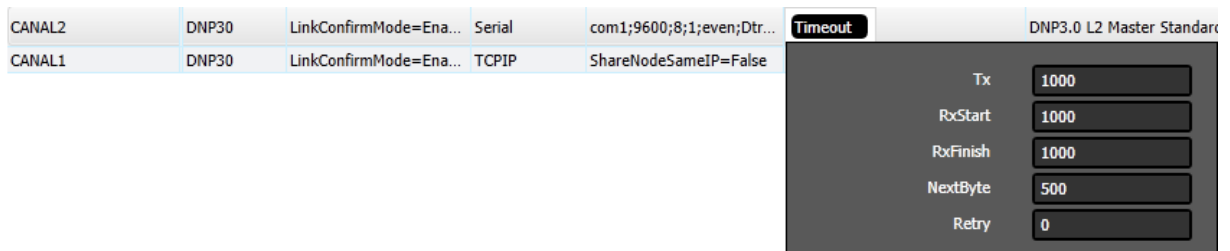


Figura 4 – Atributos de Timeout

2.4 Estado Inicial

Define como o canal iniciará: habilitado ou Desabilitado. Se o canal estiver desabilitado, não serão enviadas nem recebidas mensagens através do mesmo, isto é o canal ficará inativo.

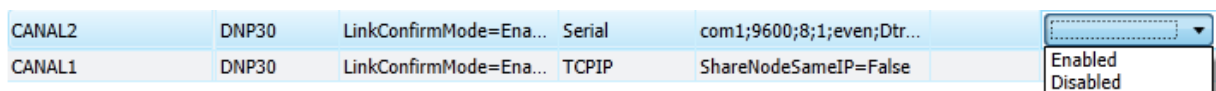


Figura 5 – Estado inicial do Canal

3. CONFIGURAÇÃO DE “NODES”

Deve ser configurado um único nó como estação servidora em cada canal. Caso seja necessário servir a mais de um cliente, devem ser configurados tantos canais quantos são os clientes.

3.1 Dados dos Canais (primário e backup)

Conjunto de atributos associados ao nodo (canal), referentes ao seu endereço e outros atributos apresentados abaixo:

NDNP1	DNPSLV	1;5400;1;Enabled;Enabled										
<div style="border: 1px solid gray; padding: 5px;"> <table> <tr> <td>IP</td> <td><input type="text" value="1"/></td> </tr> <tr> <td>Port</td> <td><input type="text" value="5400"/></td> </tr> <tr> <td>SlaveID</td> <td><input type="text" value="1"/></td> </tr> <tr> <td>IgnoreBitOnLine</td> <td><input type="button" value="Enabled"/> ▼</td> </tr> <tr> <td>WaitForIdleToCmd</td> <td><input type="button" value="Enabled"/> ▼</td> </tr> </table> </div>			IP	<input type="text" value="1"/>	Port	<input type="text" value="5400"/>	SlaveID	<input type="text" value="1"/>	IgnoreBitOnLine	<input type="button" value="Enabled"/> ▼	WaitForIdleToCmd	<input type="button" value="Enabled"/> ▼
IP	<input type="text" value="1"/>											
Port	<input type="text" value="5400"/>											
SlaveID	<input type="text" value="1"/>											
IgnoreBitOnLine	<input type="button" value="Enabled"/> ▼											
WaitForIdleToCmd	<input type="button" value="Enabled"/> ▼											

Figura 6 – Dados dos Canais

Para o caso de comunicação via Tcp-Ip:

Endereço IP – Não utilizado.

Port - Não utilizado

Para ambos TCP-IP e Serial:

SlaveID –Endereço da estação Escrava, definida por este canal.

IgnoreBitOnLine - Se “enabled” indica que o driver deve ignorar a indicação de “BitOnLine” que é inserida pelo IED quando ocorre falha ou normalidade em um ponto, segundo o critério do IED.

WaitForIdleToCmd – Se “enabled” indica que um comando somente será enviado quando a comunicação de amostragem estiver em repouso, isto é, não estiver ocorrendo.

4. CONFIGURAÇÃO DE PONTOS

4.1 Geral

Os pontos podem ser de entrada ou de saída.

Os pontos de entrada, isto é os pontos que são servidos pelo protocolo tem basicamente dois parâmetros principais: O tipo de ponto e o seu endereço.

No mapa de endereços de um IED os endereços reiniciam para cada tipo de ponto.

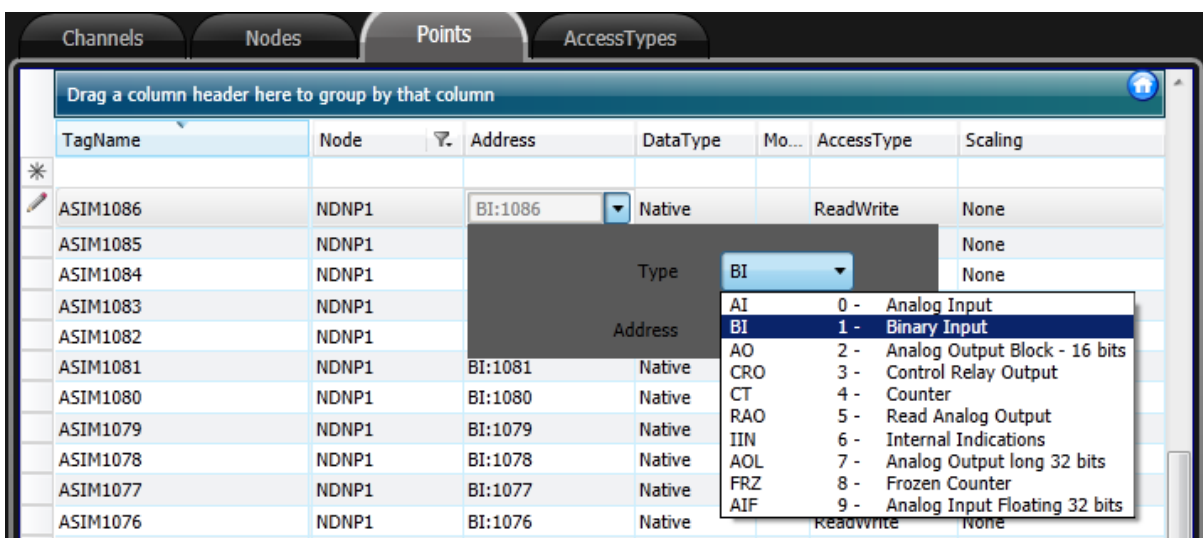
Os estados ou valores dos pontos são reportados ao cliente por Objetos de Informação definidos no protocolo. Estes objetos possuem variações como com ou sem “timestamp”. Sempre que o escravo reportar mudanças em classes 1 ou 2 , utilizará a variação com “timestamp” utilizando o ultimo “timestamps” correspondente de um ponto no ActionNET. Quando o escravo reportar leituras gerais de analógicos, contadores ou digitais este protocolo utilizará variação sem “timestamp”.

O módulo de comunicação no modo Escravo, implementa as seguintes operações:

- Respostas a pedidos de leitura de pontos digitais;
- Respostas a pedidos de leitura de variáveis analógicas;
- Respostas a pedidos de Leitura de contadores;
- Execução de solicitação de comandos de Pontos digitais e analógicos;

4.2 Tipos de pontos

Os tipos de pontos implementados são definidos utilizando os objetos de dados definidos na norma, a seguir apresentados. Para cada tipo de ponto quaisquer que sejam as variações de objetos recebidos do IED , com “flag” ou sem, com timestamp ou sem, os valores adquiridos serão colocados em pontos com o tipo entre os abaixo listados. Na tabela **pontos**, no campo “**address**” escolhe-se o tipo do ponto e especifica-se seu endereço.



TagName	Node	Address	DataType	Mo...	AccessType	Scaling
ASIM1086	NDNP1	BI:1086	Native		ReadWrite	None
ASIM1085	NDNP1					None
ASIM1084	NDNP1					None
ASIM1083	NDNP1					
ASIM1082	NDNP1					
ASIM1081	NDNP1	BI:1081	Native			
ASIM1080	NDNP1	BI:1080	Native			
ASIM1079	NDNP1	BI:1079	Native			
ASIM1078	NDNP1	BI:1078	Native			
ASIM1077	NDNP1	BI:1077	Native			
ASIM1076	NDNP1	BI:1076	Native		Readwrite	None

AI - Analog Input

Medição analógica escalar utilizado para transmissão de grandezas analógicas. Utilizado para enviar dados através do objeto 32 variação 1. São numeros de 32 bits inteiros.

BI - Binary Input

Ponto de entrada binario simples, valor 0 ou 1. Utilizado para enviar dados através do objetos 2 variação 2.

RAO – Read Analog Output

Não utilizado na implementação do modo escravo.

CRO - Control Relay Output

Ponto de saída digital utilizado para receber comandos de chaves e disjuntores. É esperado o objeto DNP 12, incluído o *Control Code* (8 bits) para indicação do tipo de comando e detalhes de execução. No cadastramento destes pontos em modo escravo, não é preciso definir o control code, já que somente é executada a alteração do valor do ponto do Action.NET , para que esta alteração sirva como um disparador de um comando no protocolo cliente que executará o comando em um IED verdadeiro.

No caso de controle de chaves e disjuntores recomenda-se a utilização de dois Tags: um para o Trip e outro para o Close.

Os valores possíveis recebidos do cliente são apresentados na tabela abaixo com as respectivas ações associadas:

Control Code	Ação protocolo	Ação no tag Action.NET executada no escravo
1	Saída Pulse ON	Altera para 1 e a seguir para Zero
2	Saída Pulse OFF	Altera para Zero e a seguir para 1
3	Saída Latch ON	Altera para 1
4	Saída Latch OFF	Altera para ZERO
65	Saída Pulse ON + Close	Altera para 1 e a seguir para Zero
66	Saída Pulse OFF + Close	Altera para Zero e a seguir para 1
67	Saída Latch ON + Close	Altera para 1
68	Saída Latch OFF + Close	Altera para ZERO
129	Saída Pulse ON + Trip	Altera para 1 e a seguir para Zero
130	Saída Pulse OFF + Trip	Altera para Zero e a seguir para 1
131	Saída Latch ON + Trip	Altera para 1
132	Saída Latch OFF + Trip	Altera para ZERO
+ 16	Enfileirar (Queue) + Trip	Nada faz
+ 32	Limpar (Clear) + Trip	Nada faz

CT – Counter

Contador binário de 16 ou 32bits, enviado através do objetos 20 variação 1. Este numero possui o ultimo estado do tag de contagem, no instante em que é enviado.

FRZ - Frozen Counter

Contador binário de 32 bits, enviado através do objeto 21 variação 1, que contém a informação de um contador utilizado como um tag.

AO - Analog Output Status or Block (16bits)

Quando este modo escravo recebe mensagem com Function Code de escrita (2 = Write ou 4=Operate, ou 5= DirectOperate) e este objeto ou 41 (valor requerido para ser alcançado na saída analógica) na suas variações 2 (16 bits), este valor é escrito no tag definido na tabela POINTS para o endereço recebido no objeto..

AOL - Analog Output long (32 bits)

Quando este modo escravo recebe mensagem com Function Code de escrita (2 = Write) e este objeto 41 (valor requerido para ser alcançado na saída analógica) na suas variações 1 (32 bits), este valor é escrito no tag definido na tabela POINTS para o endereço recebido no objeto..

AIF - Analog Input Floating (32 bits)

Medição analógica utilizada para transmissão de grandezas analógicas. Tags com este tipo devem ser tipificados no Action.Net como reais Seu valor é enviado através do objeto 30 variação 5. São numeros de ponto flutuante de 32 ou 64 bits, formato IEEE-754.

IIN - Internal Indications

Como é o padrão do protocolo os status dos Internal Indications ou status de resposta de comandos são enviados em todas as mensagens de resposta do escravo, para utilização pelo cliente. A tabela mostra o formato destas duas palavras de 16 bits.

BIT	Origem	Descrição	Conteúdo
0	IIN	Broadcast	Retorna 1 quando escravo recebe uma mensagem de broadcast (endereço = FFFF)
1	IIN	Classe 1	Retorna 1 se escravo tem eventos classe 1
2	IIN	Classe 2	Retorna 1 se escravo tem eventos classe 2
3	IIN	Classe 3	Retorna 1 se escravo tem eventos classe 3
4	IIN	Sincronismo de relógio	Retorna 1 se escravo pede sincronização de relógio
5	IIN	Saídas em local	Retorna 1 se algumas ou todas as saídas estão em local
6	IIN	Problema	Retorna 1 se escravo com problema
7	IIN	Reinício	Retorna 1 se escravo reiniciou (restart)
8	IIN	Função não implementada	Retorna 1 se função enviada pelo mestre não está implementada no escravo
9	IIN	Objeto desconhecido	Retorna 1 se escravo não tem determinado objeto ou classe não tem esse objeto

10	IIN	Dado inválido	Retorna 1 se tem parâmetro inválido no qualificador ou intervalo de endereços está inválido
11	IIN	Overflow	Retorna 1 se buffer do escravo teve overflow
12	IIN	Ocupado	Retorna 1 se requisição foi entendida mas já está sendo executada
13	IIN	Dado corrompido	Retorna 1 se dados de parametrização estão corrompidos
14	IIN	Reservado	Sempre 0
15	IIN	Reservado	Sempre 0

Status field: 8 bits enviados como resposta de um comando.

BIT	Origem	Descrição	Conteúdo
0	Status	Comando aceito	Retorna 1 após comando correto
1	Status	Comando não aceito	Retorna 1 se ocorreu time out entre tempo de select e operate.
2	Status	Falta select	Retorna 1 se foi feito um operate sem prévio select
3	Status	Erro formato	Retorna 1 se comando tem erro de formato
4	Status	Controle não suportado	Retorna 1 se operação não é suportada
5	Status	Fila cheia	Retorna 1 se fila de requisição do escravo está cheia ou ponto já está ativo
6	Status	Erro hardware	Retorna 1 se ocorreu um erro de hardware do dispositivo ao tentar executar o comando
7		Não utilizado	

4.3 Endereço do ponto

O campo **Address** a ser preenchido no cadastramento de pontos é o que a norma chama de "Index". Trata-se de um número de 16 bits que é o índice [0 a n-1] indicativo de cada um dos pontos do mesmo tipo mapeados dentro do IED.

Abaixo, a título de exemplo é apresentada uma tabela de pontos preenchida com diferentes tipos de ponto. Os pontos tipo saída digital (CRO), conforme citado acima, além do endereço possuem o código de controle.

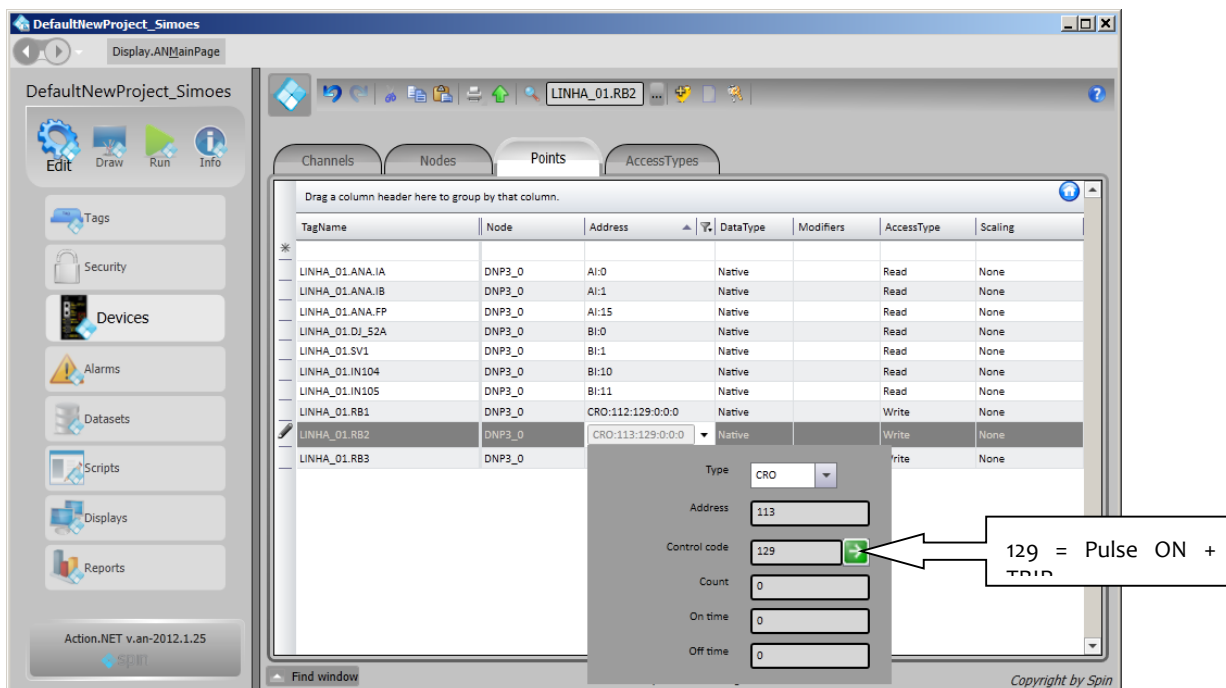
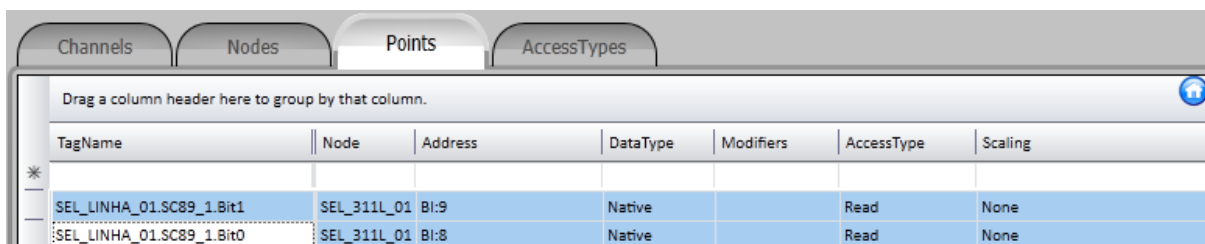


Figura 12 – Exemplo da tabela de Points com diferentes tipos de pontos

Para implementar pontos de entrada digital discretos, basta usar o atributo “Bit” de um Tag, para cada um dos pontos que definem o valor da entrada digital discreta. Assim, por exemplo, uma seccionadora que tem dois contatos para definir seu estado:

Tag	ENDEREÇO	Complemento
SEL_LINHA_01.SC89_1	8	Seccionadora aberta
SEL_LINHA_01.SC89_1	9	Seccionadora Fechada

É definida como um Tag AnalogInt e na tabela node se usa o atributo Bit desta variável de 16 bits (AnalogInt) para endereçar os dois pontos, como na tabela abaixo:



Os valores assumidos pela variável SEL_LINHA_01.SC89_1 serão:

Tag	Bit 0	Bit 1	VALOR	SIGNIFICADO
SEL_LINHA_01.SC89_1	0	0	0	INDEFINIDO
SEL_LINHA_01.SC89_1	1	0	1	ABERTA
SEL_LINHA_01.SC89_1	0	1	2	FECHADA
SEL_LINHA_01.SC89_1	1	1	3	ERRO

4.4 Access Type

Por ser um módulo de comunicação em modo servidor há a necessidade de algumas características próprias para a parametrização do campo **Access Type** da tabela **Points**:

Para pontos com tipos que são de leitura ou receptores de comandos (CRO, AO e AOL):

O Access Type deve ser definido com:

```
ReadOnStartup= On;  
ReadPooling= Never;  
WriteEvent= Changed;  
AccepUnsolicited = On;
```

Para pontos com tipos de leitura , para os quais não se deseja geração de eventos

O Access Type deve ser definido da mesma forma acima porém com com:

```
WriteEnable = off;  
WriteEvent= none;
```