

# CRIANDO UMA CONFIGURAÇÃO MODBUS USANDO O DF89

## Introdução

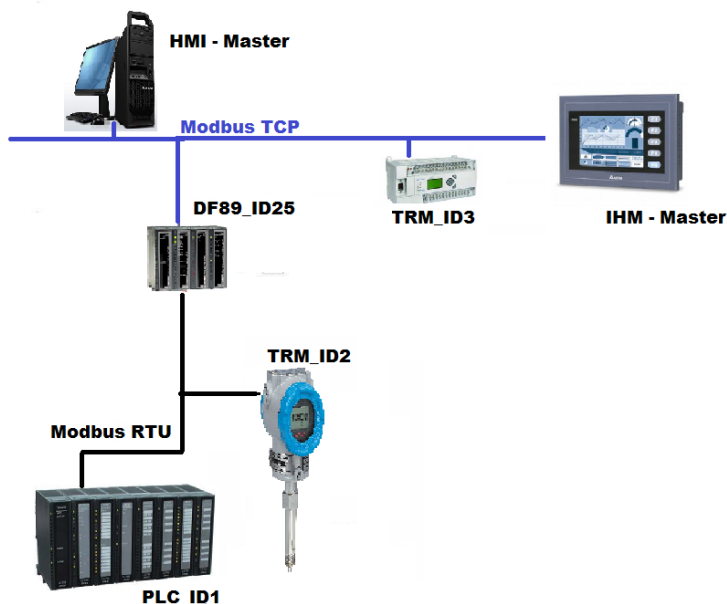
Nesta seção, será mostrada a configuração de uma estratégia utilizando como controlador o DF89. O controlador DF89 pode trabalhar em vários cenários Modbus. Portanto serão abordados diferentes cenários de aplicação ao longo dos tópicos desta seção.

### NOTA

- As características do DF89 descritas nesta seção somente estão disponíveis a partir do firmware versão 4 (V4.x.x) com *Device Description* 4.xx.
- As características de configuração do Modbus das versões anteriores ao firmware 4 são as mesmas descritas na seção “Adicionando Modbus” deste manual.
- O DF89 somente possui porta RS-485 a partir do número de série SN623.
- Os módulos com número de série inferior ao SN623 suportam o firmware versão 4 e as características descritas neste capítulo somente para a porta serial RS-232.

Para ilustrar a configuração considere como exemplo um processo que possui dois transmissores, um PLC escravo, uma IHM e uma estação de supervisão (HMI). A arquitetura da rede é mostrada na figura abaixo. Os cenários existentes nesse processo usado como exemplo são:

- 1) O DF89 faz o papel de mestre Modbus RTU do PLC escravo (PLC\_ID1) e do transmissor escravo (TRM\_ID2) na porta serial RS-485.
- 2) O DF89 faz o papel de mestre Modbus TCP do transmissor escravo (TRM\_ID3) na Ethernet.
- 3) O DF89 é escravo Modbus TCP da IHM (IHM Master).
- 4) A estação de supervisão (HMI Master) pode ler os dados do DF89 (concentrador) ou ler os dados diretamente dos escravos na porta serial do DF89 (*Bypass*).



**Figura 23. 1 – Exemplo de processo usando vários cenários Modbus com o DF89**

A configuração de cada um destes cenários é explicada passo a passo, nos tópicos que se seguem. No início, os passos de 1 a 4 são comuns para todos os cenários. A partir do passo 5 existirão diferenças de acordo com o cenário.

No **SYSTEM302** é possível fazer a lógica de controle de duas formas: ladder ou blocos funcionais. Os passos de 5 a 8 são necessários para a configuração da rede Modbus. O passo 9 mostra como mapear os pontos da rede Modbus em pontos disponíveis para utilização em Lógica Ladder ou em lógica de Blocos Funcionais. O passo 10 mostra como fazer a lógica usando ladder e o passo 10' mostra como fazer a lógica utilizando blocos funcionais. Também é possível misturar as duas configurações, ou seja, fazer parte em ladder e parte em blocos funcionais.

#### IMPORTANTE

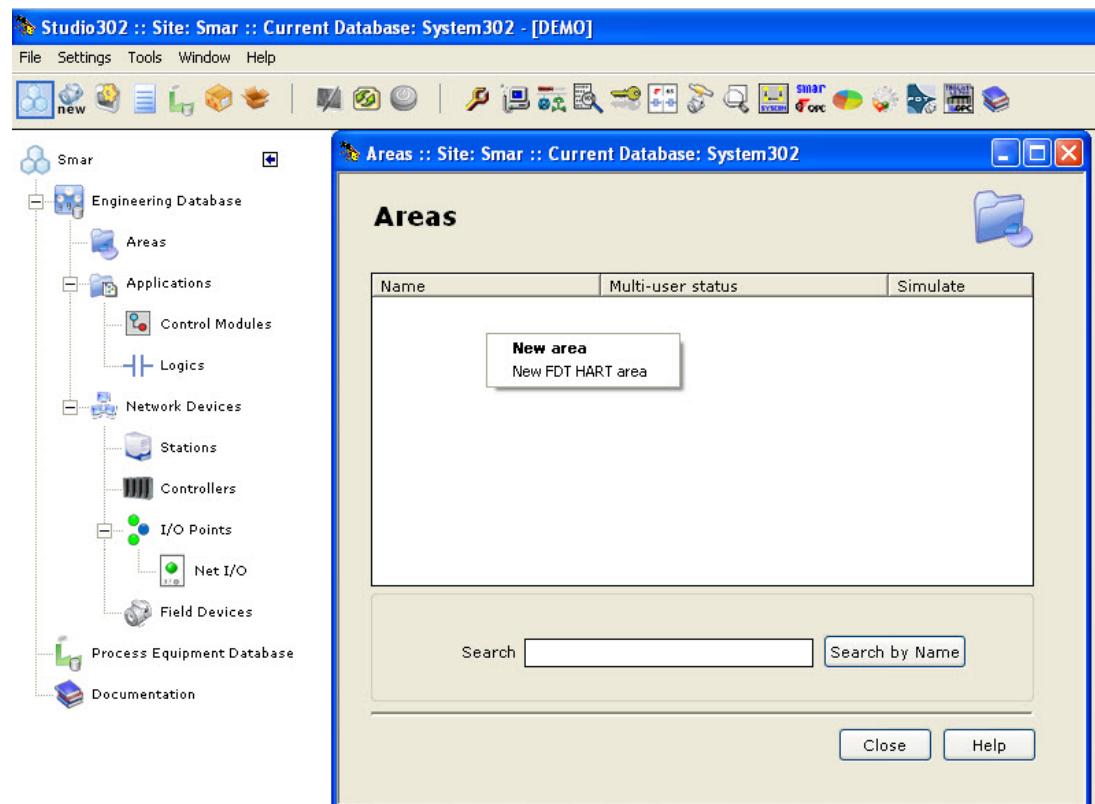
Antes da configuração da rede Modbus é necessário previamente obter informações de configuração de cada equipamento escravo como taxa de comunicação, tipo de dados, endereçamento dos pontos e funções suportadas. Estas informações são obtidas com o fabricante do respectivo equipamento Modbus.

## Criando uma Área Básica

Neste tópico será mostrado como criar uma área básica de DF89 que poderá ser utilizada em qualquer um dos cenários.

### Passo 1

Para criar uma área (na linguagem do **SYSTEM302**), clique com o botão direito dentro da janela **Areas** do **Studio302** e escolha **New Area** como mostrado na figura seguinte.



**Figura 23. 2 - Opções para criação de nova área no Studio302**

Então será mostrada uma janela para o usuário escolher o tipo de configuração que deseja. Como o controlador DF89 é um equipamento HSE, pode ser escolhidos **HSE Area** ou **Predefined**, que é a forma que utiliza um *template*.

Na janela seguinte são mostrados os tipos de *templates* existentes. Para todos os controladores existem as opções com bloco FFB1131 ou sem ele. Aqui será utilizado o *template* de um DF89 usando o FFB1131.

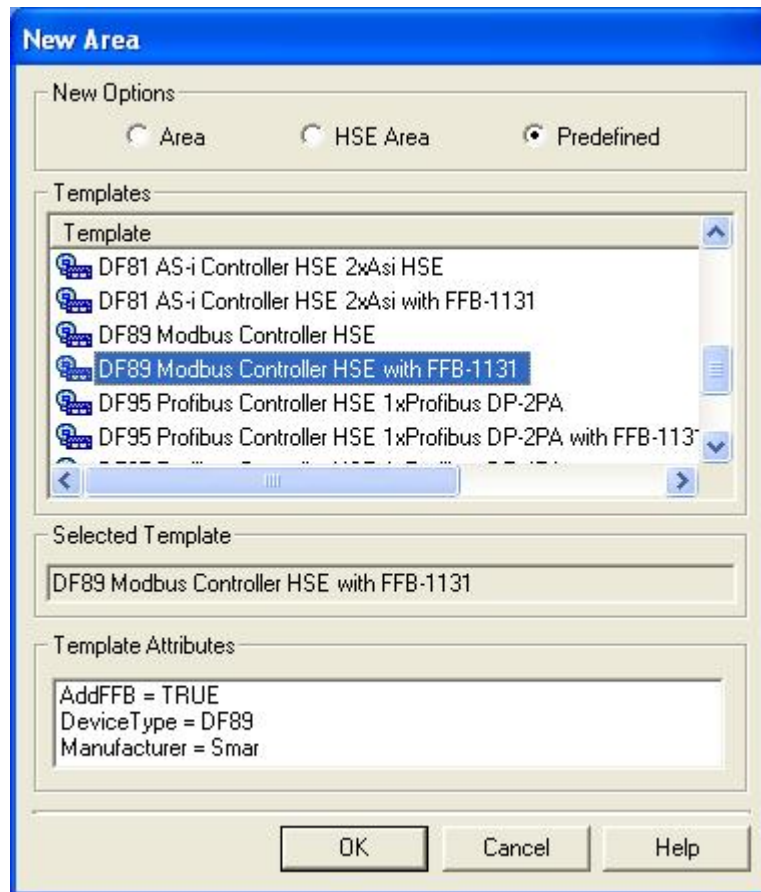


Figura 23. 3 - Janela de configurações pré-definidas

Escolha o *template* e clique **OK**.

#### Passo 2

O usuário deve entrar com o nome da área como mostrado na figura abaixo.



Figura 23. 4 – Nome da nova área

Digite o nome **PROJ\_DF89** para o exemplo proposto no campo **Area Name** e clique **OK**. Então a nova área será criada com o controlador DF89. Veja figura seguinte.

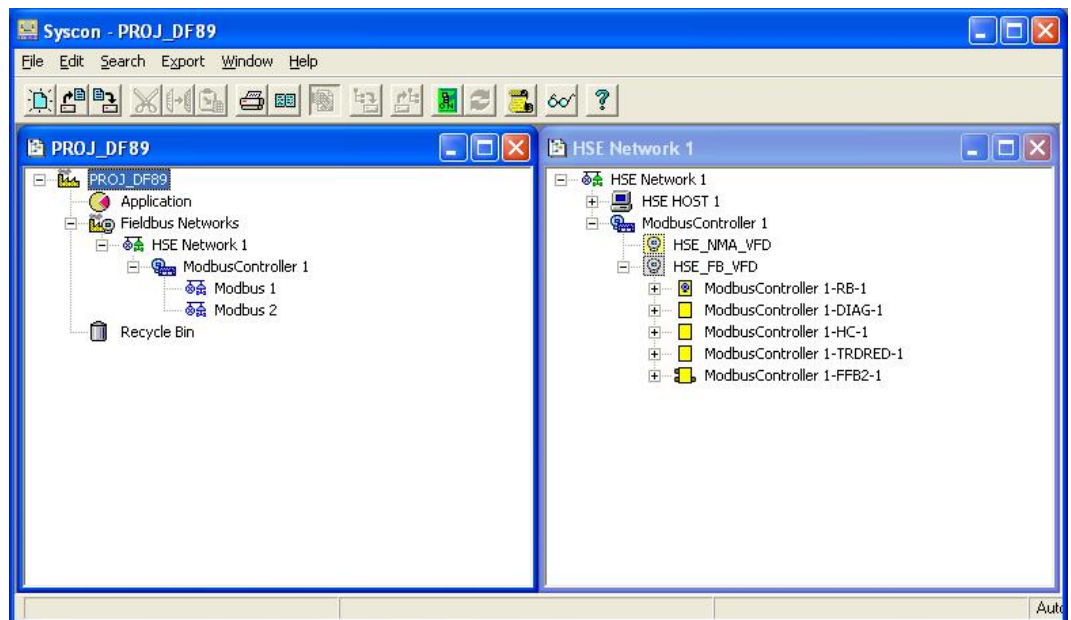


Figura 23. 5 – Syscon com as janelas da área e do canal HSE

Na figura acima é mostrada a janela do **Syscon**, sendo que a da esquerda mostra a rede fieldbus HSE (**HSE Network 1**) e abaixo dela existe somente um controlador, DF89, com tag default (**ModbusController 1**). Abaixo do controlador DF89 já foram criados os dois canais suportados por este equipamento: **Modbus 1**, que corresponde ao canal serial, e **Modbus 2** que corresponde ao canal TCP.

A segunda janela da direita representa a expansão (ou a visão) do canal HSE que tem somente o **ModbusController 1**. Outros controladores HSE podem ser adicionados bastando clicar em cima do canal e pedir para adicionar novos equipamentos.


### Passo 3

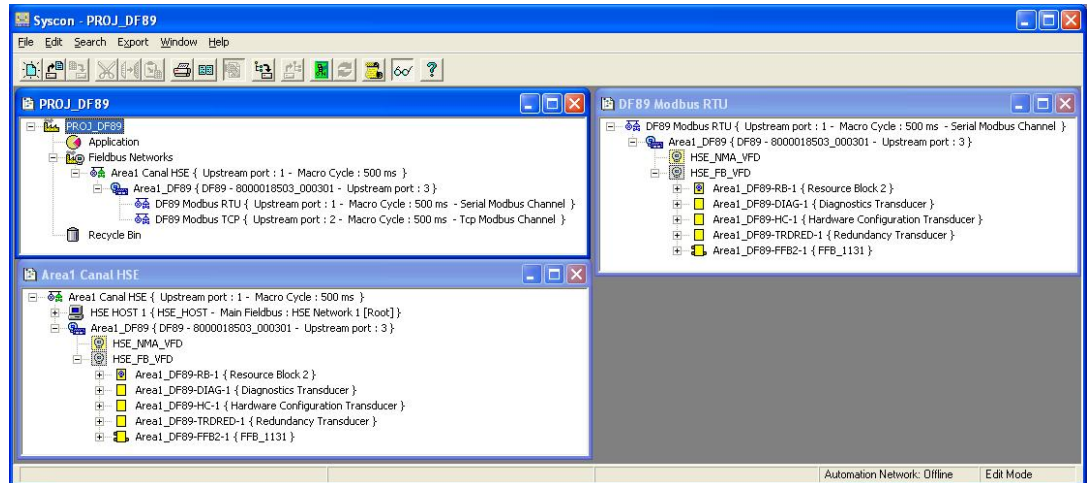
Estes tags default podem ser alterados, para isto basta clicar no correspondente tag na janela principal com o botão direito e escolher **Attributes**. O exemplo da figura abaixo foi alterado o tag do DF89 para **Area1\_DF89**.



Figura 23. 6 – Janela de atributos do controlador

Para alterar o tag dos canais basta, na janela principal, clicar sobre o canal (por exemplo no Modbus 1) e pedir para expandir a janela. Uma nova janela abrirá. No alto da janela clique no tag do canal como o botão direito do mouse e escolha **Attributes**. Na figura abaixo são mostradas as configurações do controlador e dos canais com os novos tags. Foram alterados os tags dos canais: HSE – de **HSE Network 1** para **Area1 Canal HSE**  
 DF89 Modbus Serial – de **Modbus 1** para **DF89 Modbus RTU**  
 DF89 Modbus TCP – de **Modbus 2** para **DF89 Modbus TCP**

Para facilitar a visualização, o botão **Details**  na *toolbar* mostra o tipo do objeto entre chaves { }; canal, controlador, bloco . Na figura abaixo são mostrados os novos tags com os respectivos tipos.



**Figura 23. 7 – Syscon com as Janelas da área (acima à esquerda), do canal HSE (abaixo à esquerda) e do canal serial (acima à direita)**

**IMPORTANTE**

É importante ressaltar que nem todos os caracteres são válidos para a atribuição de *tags*.

Os caracteres válidos são:

A-Z a-z 0-9 # { } [ ] ( ) + -

Os caracteres inválidos são:

~ ` ! @ # \$ % ^ & \* = | : ; , . < > ? / ' " \

**ATENÇÃO**

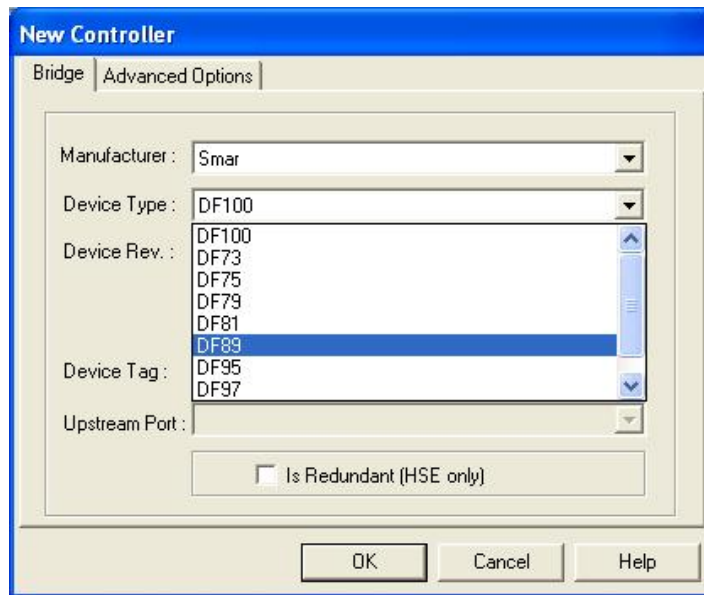
Aqui foi usado o método mais fácil de criar uma configuração, baseada no *template*. Neste caso, a configuração é criada com todos os canais necessários. Caso o usuário não queira um determinado canal ele pode ser excluído.

Existe outra forma de criar a configuração em que o controlador e o canal são criados manualmente. Para isso, no passo 1 da figura 23.3, o tipo de área a ser escolhida deve ser a opção **HSE Area**. Neste caso será criada somente uma configuração HSE vazia. O passo 4 descrito a seguir mostra como criar um controlador a partir da janela HSE. Para mais detalhes refira-se ao manual do **Syscon**.

A opção de criação a partir de um *template* somente está disponível para novas áreas.

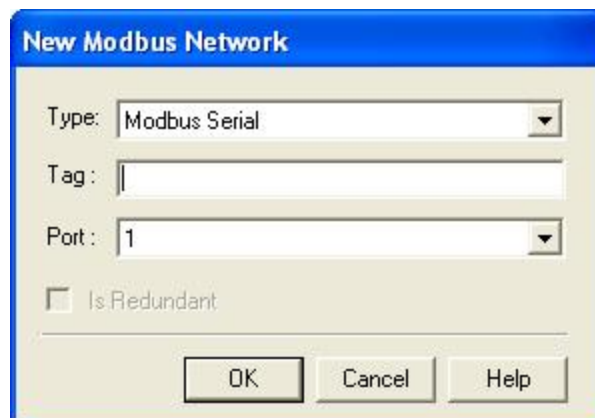
**Passo 4**

Este passo somente servirá para adicionar um novo controlador ou quando for criada a configuração somente com o HSE Host. No exemplo deste seção não é necessário este passo. Para adicionar um controlador DF89, na janela do canal HSE, clique com o botão direito sobre o **Area1 Canal HSE**, escolha **New Controller** e, em seguida, **DF89** como mostrado na figura seguinte.



**Figura 23. 8 – Criando o Controlador DF89 Manualmente**

O novo controlador será adicionado. Então ainda na janela da área clique com o botão direito do mouse e escolha **New Network**. Aparecerá a janela mostrada na figura abaixo. Preencha o tipo e o tag do canal. Clique **OK** e automaticamente abrirá o **SmarNetConf**.



**Figura 23. 9 – Criando o canal Modbus**

## Configuração do DF89 como Mestre Modbus Serial

Nesta seção serão detalhados todos os passos para fazer a seguinte configuração:

- Adicionar equipamentos no canal serial do DF89
- Adicionar pontos Modbus nos equipamentos
- Mapear os pontos Modbus para lógica do **SYSTEM302**

Para o exemplo do processo da figura 23.1 são utilizados um PLC escravo (PLC\_ID1) e um transmissor (TRM\_ID2) que são ligados no canal RS-485 do DF89 usando Modbus RTU.

### IMPORTANTE

No cenário de mestre Modbus serial o DF89 possui dois meios físicos RS-232 e RS-485, porém o DF89 somente possui uma porta serial. Ou seja, somente poderá trabalhar com a mesma configuração (*baudrate*, *stop bit*, paridade e *timeout*) para ambos os meios (RS-232 e RS-485). O DF89 suporta até 32 equipamentos no canal serial. Por isso somente existe um canal serial.

## Passo 5

## Configuração da rede Modbus

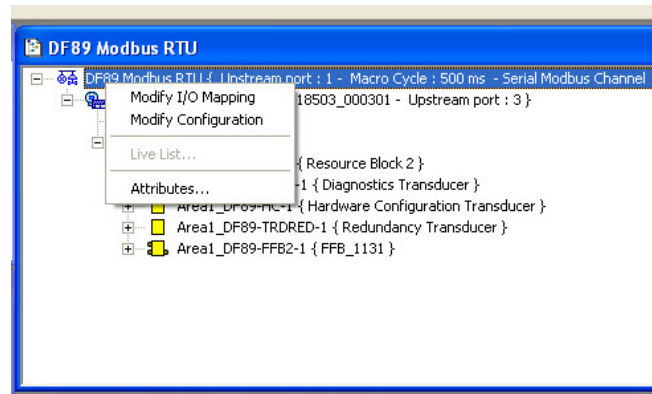
O **SmarNetConf** ou **Network Configurator** é a ferramenta de configuração da rede Modbus. Nela, o usuário definirá os parâmetros da rede, todos os escravos que a compõem e todos os parâmetros necessários para comunicação Modbus. O **SmarNetConf** somente tem a visão do respectivo canal (Serial ou TCP), ou seja, na visão do canal serial ele somente mostrará os equipamentos correspondentes daquele canal escolhido.

Toda vez que o usuário precisar configurar o canal do DF89 como **Mestre Modbus** ele deve usar o **SmarNetConf**. Isto será possível nos seguintes casos :

- O DF89 é mestre de escravos Modbus RTU no canal serial (RS-232/RS-485);
- O DF89 é *bypass* de equipamentos na porta serial;
- O DF89 é mestre de escravos Modbus TCP no canal TCP (ETH1);

Maiores detalhes serão dados nas seções respectivas dos cenários. Para outras informações sobre o **SmarNetConf** consulte o help da ferramenta.

Para abrir o **SmarNetConf** primeiramente expanda a janela do canal, neste exemplo o canal serial **DF89 Modbus RTU**. Selecione o canal e clique com o botão direito do mouse. Escolha a opção **Modify Configuration**. Veja figura seguinte.



**Figura 23. 10 – Abrindo SmarNetConf para configurar equipamento da rede Modbus Serial**

A figura abaixo mostra a janela principal do **SmarNetConf** do canal serial.

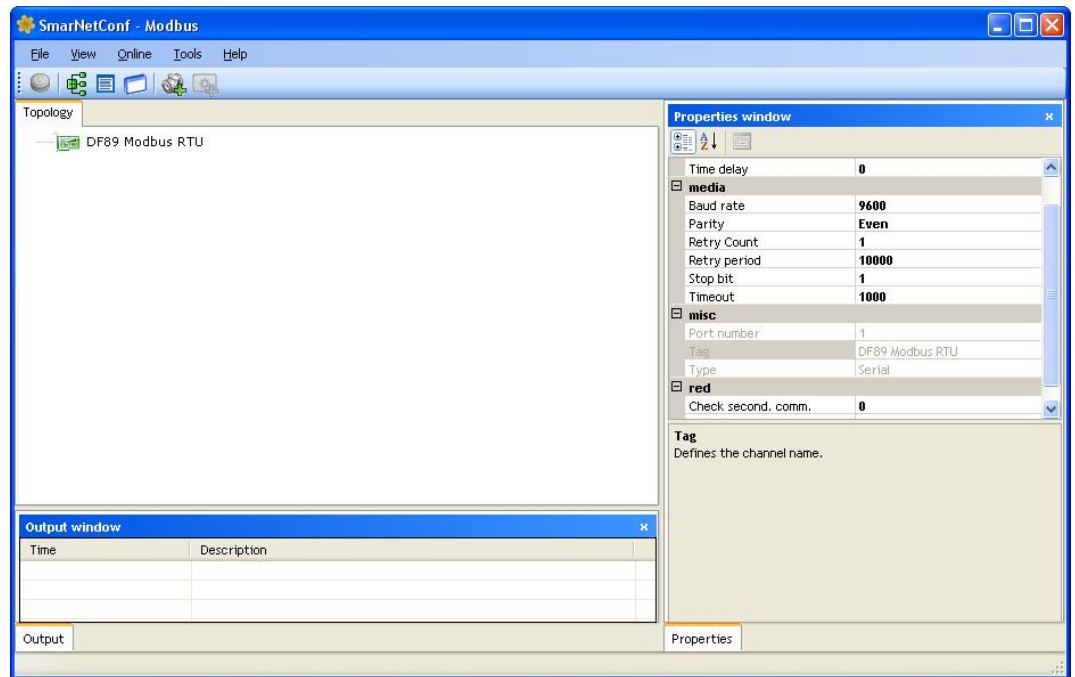


Figura 23. 11 – Janela principal do Configurator Modbus SmarNetConf

O configurador possui três janelas principais como mostrado na figura anterior: a janela de topologia (**Topology**) à esquerda, que mostra em estrutura de árvore os equipamentos da rede em determinado canal, a janela de propriedades (**Properties**) à direita mostra todas as propriedades do item selecionado na janela de topologia e, por fim, a janela de eventos e logs de erros (**Output Window**) na parte inferior da janela.

Na janela de propriedades do canal devem ser preenchidos os parâmetros do canal – *Baudrate*, *Stop Bit*, paridade, *timeout*, etc. Cada um destes parâmetros são detalhados na seção Parametrização do **SmarNetConf**.

#### NOTA

Todos os equipamentos devem ter a mesma parametrização de meio (*Baudrate*, *Stop Bit* e paridade). O DF89 considera o parâmetro *databits* sempre igual a 8.

## Passo 6

### Configurando os equipamentos Modbus

Após a criação da rede Modbus, os equipamentos deverão ser configurados. Para inserir um equipamento escravo, clique no botão **Insert Device** localizado na barra de ferramentas, ou através do menu **Insert**→**Device**. A figura seguinte mostra a janela de inserção de equipamentos escravos, nela o usuário pode adicionar um ou vários equipamentos do mesmo tipo.



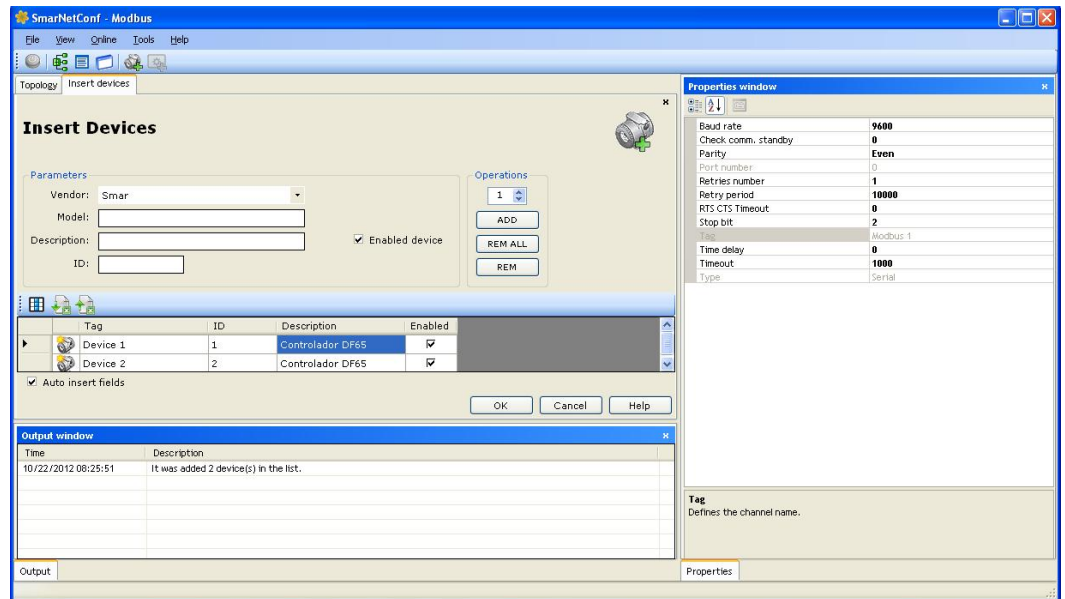


Figura 23. 12 – Adicionando equipamentos na rede Modbus

Nesta janela, deve ser selecionado o equipamento escravo que será inserido. Para o exemplo proposto, serão incluídos no canal serial dois equipamentos. Primeiramente obtenha as informações de configuração Modbus do equipamento escravo (veja seu manual). Siga os passos abaixo para a inserção do equipamento escravo:

- A. Preencha os campos **Model**, **Description** e **ID** do equipamento;
- B. Selecionar a quantidade de equipamentos de determinado tipo;
- C. Clique **Add**. Neste momento será mostrada uma tabela com a quantidade de equipamentos selecionados;
- D. No campo **ID**, atribua um endereço para este equipamento.
- E. No campo **Tag** é dado o tag do equipamento. Caso o usuário não atribua um *tag* ao equipamento, será atribuído um *tag default*;
- F. Clique **OK**. Então na janela principal serão incluídos os equipamentos adicionados.

#### NOTA

Os endereços atribuídos assim como os tags dos equipamentos deverão ser únicos na rede Modbus e também no *Database* do **Studio302** (Workspace). Para maiores detalhes refira-se ao manual do **Studio302**.

Os dois equipamentos (PLC\_ID1 e TRM\_ID2) do exemplo proposto, no canal serial, são mostrados na figura seguinte na tela do **SmarNetconf**.

### Passo 7

#### Editando as propriedades do equipamento

Após a adição do equipamento, as propriedades dele podem ser alteradas. As principais propriedades são tag, ID e se o equipamento está ou não habilitado. Outras propriedades podem ser alteradas nesta janela. Uma explicação de cada propriedade é mostrada na seção **Parâmetros do SmarNetConf**.

Por exemplo, para editar as propriedades do **PLC\_ID1**, clique sobre o equipamento Modbus correspondente, com o botão direito do *mouse*. Na *pop up* aberta, escolha a opção **Edit Device Properties** para configurar os parâmetros do equipamento escravo.

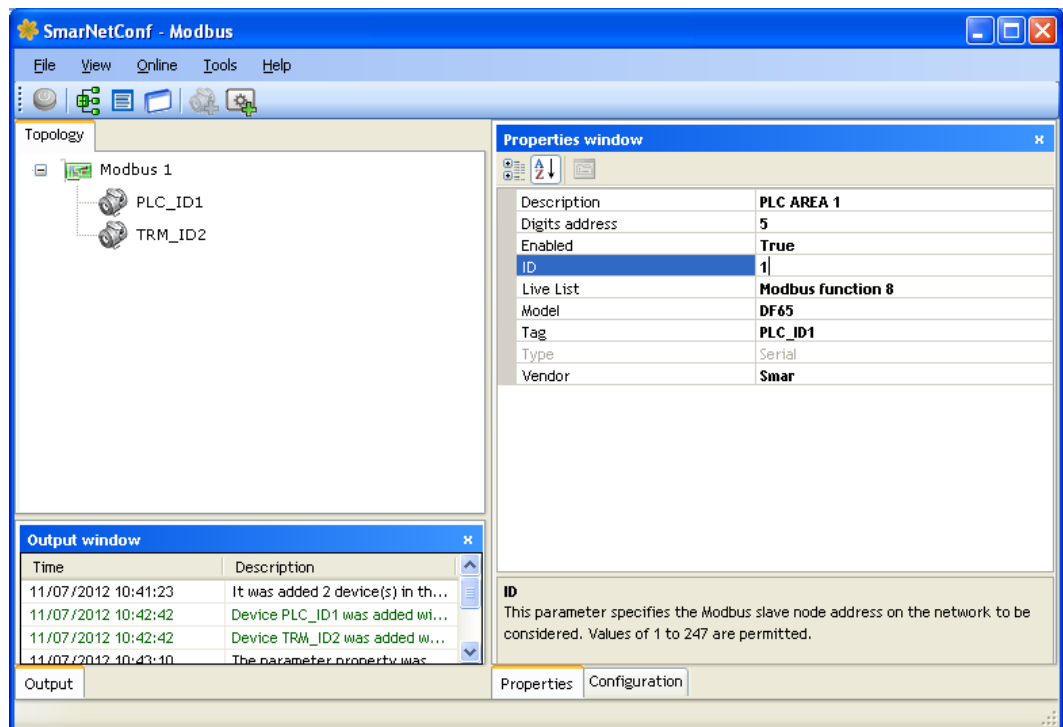


Figura 23. 13 – Editando as propriedades do equipamento

## Passo 8

### Adicionando comandos ao equipamento

As variáveis do escravo que serão supervisionadas ou escritas são mapeadas em comandos. Estes comandos podem ser individuais ou múltiplos. Por exemplo, deseja-se supervisionar os pontos analógicos nos endereços 40001 e 40010. Esta supervisão pode ser através de dois comandos individuais (cmd1 para leitura de 40001 e cmd2 para leitura de 40010) ou um comando múltiplo (cmd1- leitura a partir do ponto 40001 com 10 registros). Esta ultima forma é a mais comum e a mais otimizada.

A janela de configuração dos comandos é acessada clicando sobre o equipamento Modbus correspondente, por exemplo no **PLC\_ID1**, com o botão direito do *mouse*. Na *pop up* aberta, escolha a opção **Insert Commands** para configurar os comandos Modbus do equipamento. A figura seguinte mostra um exemplo da janela de adição de comandos.

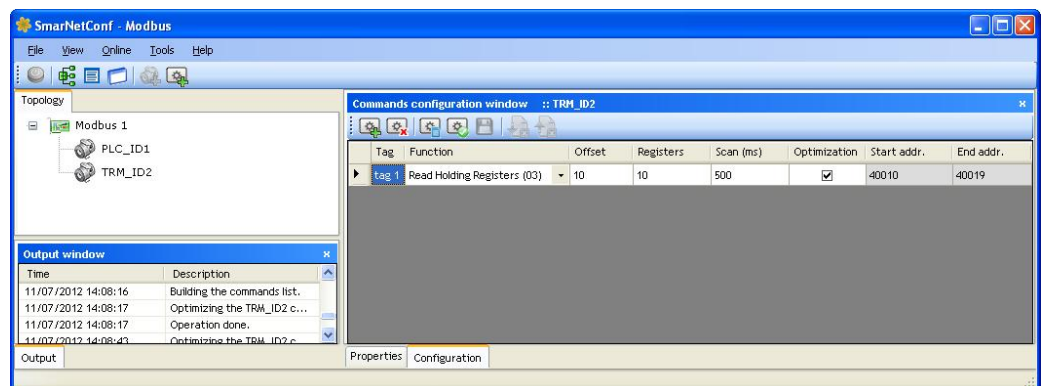


Figura 23. 14 – Configurando os comandos Modbus do equipamento

Na janela de configuração dos comandos mostrado na figura acima existe uma barra de ferramentas em que é possível adicionar, apagar, checar se a configuração está correta, salvar a configuração, importar e exportar para uma planilha no formato xls. Mais informações sobre estas características refira-se ao help do **SmarNetConf**.

Para cada comando criado, é criada uma linha na tabela até o limite de 30 comandos por equipamento. No protocolo Modbus, o endereçamento leva em conta as funções, o *offset* e o número de registros. Então para cada comando o usuário deve se atentar ao mapeamento destes três parâmetros. A descrição detalhada de cada um destes comandos e mostrada na próxima seção Parâmetros do **SmarNetConf**.

O DF89 trabalhando como mestre admite as funções Modbus mostradas na tabela abaixo. Na coluna “tipo de dado” RW significa que ele suporta leitura e escrita e RO somente suporta leitura.

Função Modbus	Descrição	Tipo de dado
1	Read Coils (0x)	Discrete Input (RW)
2	Read Discrete Inputs (1x)	Discrete Input (RO)
3	Read Holding registers (4x)	Analog Input (RW)
4	Read Input registers (3x)	Analog Input (RO)
5	Write Single Coil (0x)	Discrete Output
6	Write Single Register (4x)	Analog Output
15	Write Multiple Coils (0x)	Discrete Output
16	Write Multiple Registers (4x)	Analog Output

**Tabela 23. 1 – Funções Modbus Suportadas**

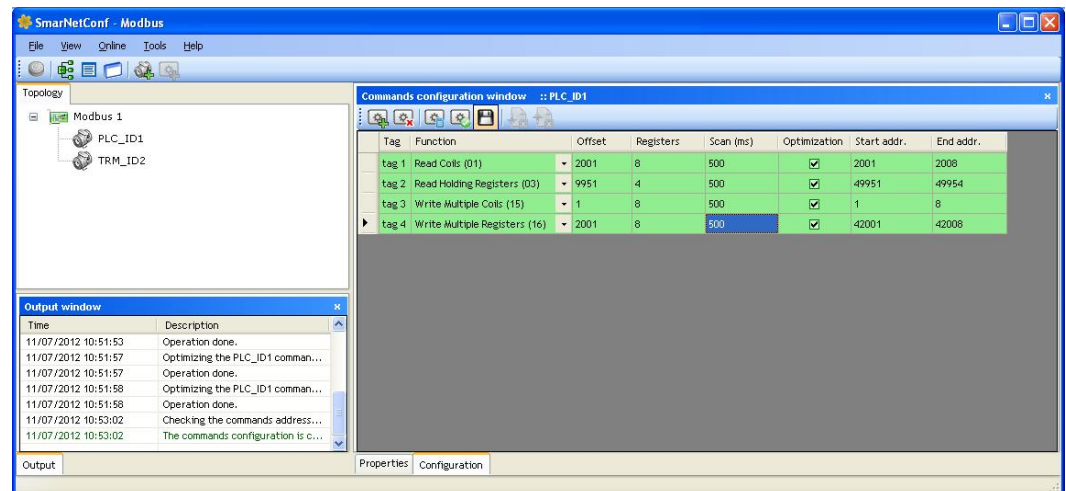
Após a inclusão dos comandos é necessário checar a configuração e então salvá-la. Este salvamento indica que será efetivada toda a configuração de comandos do equipamento na base de dados do **SYSTEM302**. Para isso não pode haver inconsistências nos comandos montados (como a inclusão de pontos repetidos nos comandos ou um comando maior que o limite. Isto é feito

pelos botões de **Check**  e **Save**  na barra de ferramentas do **SmarNetConf**.

Os campos **Scan** e **Optimization** são parâmetros avançados que podem ser deixados no Default. O campo **Scan** indica o tempo mínimo desejado entre o *scan* do respectivo comando. O campo **Optimization** indica que se deseja ou não que o comando Modbus seja otimizado. Neste último, no caso do usuário escolher não otimizar, então aquele comando será enviado da forma que foi configurado.

Por fim os campos **Start Address** e **End Address** são formas de visualizar o endereço Modbus dos pontos. O **Start Address** representa o início da faixa Modbus endereçada, enquanto que o **End Address** representa o fim naquele comando. Esta forma de visualização pode ser de 5 dígitos (segundo a especificação Modbus Modcon) ou de 6 dígitos (segundo a especificação Modbus 2.0). A propriedade **Digits Address** do equipamento indica se a visualização será com 5 ou 6 dígitos.

Por exemplo, na figura 23.13 utilizou-se a visualização de 5 dígitos. O primeiro comando foi mapeado da seguinte forma: função de leitura analógica “03”, *offset* igual a 10 e número de registros igual a 10. Isto representa o endereçamento dos registros Modbus 40010 a 40019. Caso fosse escolhido a visualização de 6 dígitos então os correspondentes endereços Modbus StartAddress seria 400010 e o EndAddress seria 400019.



**Figura 23. 15 – Configurando os comandos do PLC\_ID1**

Finalizados os passos anteriores, a configuração do canal Modbus pode ser salva e, então, a **Smar Network Configurator** pode ser fechada. Automaticamente a ferramenta **Mapping Tool** é aberta.

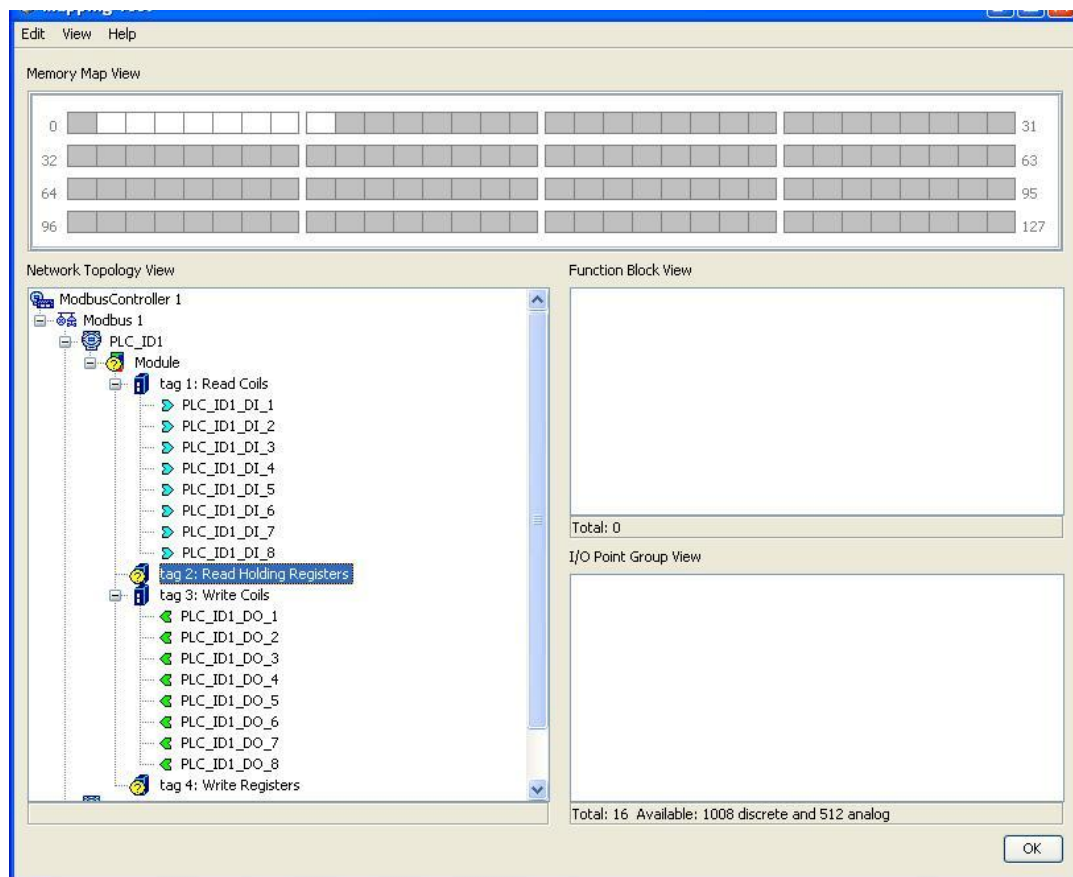


Não se esqueça de salvar a configuração de comandos antes de sair do **Smar Network Configurator**. Este é o único ponto que precisa de salvar manualmente. As mudanças de propriedades do canal ou do equipamento são salvas automaticamente.

### Passo 09


## Mapeamento dos pontos Modbus para lógica do SYSTEM302

A ferramenta **Mapping tool** é responsável por mapear os pontos da rede Modbus em pontos disponíveis para utilização em Lógica Ladder ou em lógica de Blocos Funcionais. Nesta ferramenta também são configuradas todas as características dos pontos da rede Modbus (tags de usuário, tipo de dados, escalas, etc.). Observe a figura a seguir:



**Figura 23. 16 – Janela da Mapping Tool**

Como pode ser visto na figura anterior, são exibidos na janela **Network Topology View** os equipamentos que foram inseridos na rede Modbus com seus respectivos comandos. Na janela **Function Block Label** são mostrados os pontos que serão visualizados no **Syscon** e na janela **IO Point Group View** são exibidos os pontos que serão mostrados no **LogicView for FFB**. O propósito aqui é configurar cada “ponto” (ou *byte*) da rede nos seus respectivos tipos de dados.

Por default a **Mapping Tool** automaticamente mapeia todos os registros criados no **SmarNetConf** com funções discretas (funções Modbus 1, 2, 5, 15) como tipo booleano. O usuário pode alterar somente os tags dos pontos. Os pontos analógicos, que aparecem com o símbolo , precisam ser configurados.

Para a configuração do ponto, clique sobre o comando a ser configurado, na figura seguinte, por exemplo, será no **Tag2**. Lembrando que no **SmarNetConf** o **Tag2** foi mapeado na faixa de

registros Modbus de 49951 a 49954.

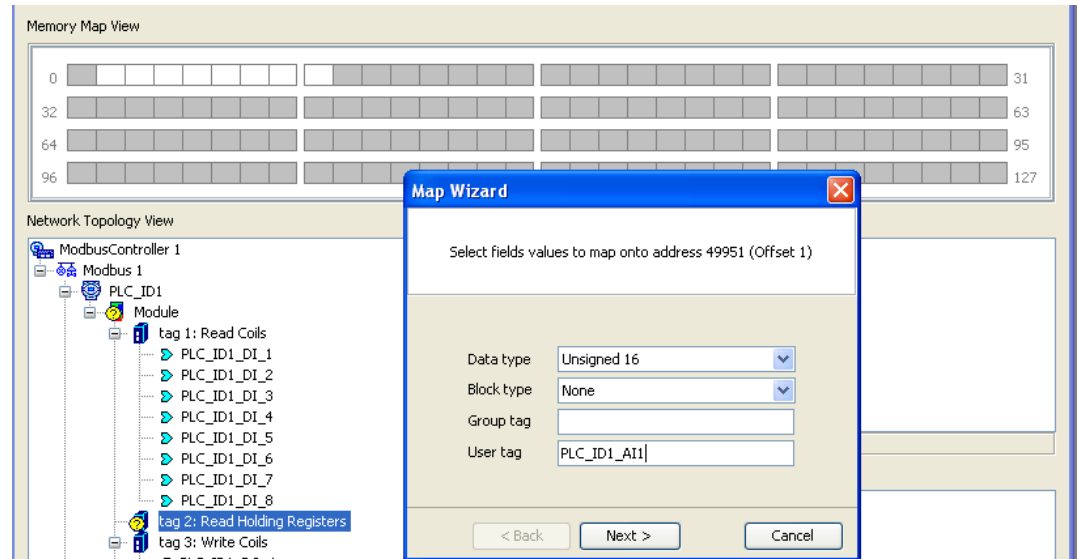


Figura 23. 17 – Mapeamento dos pontos analógicos na Mapping Tool

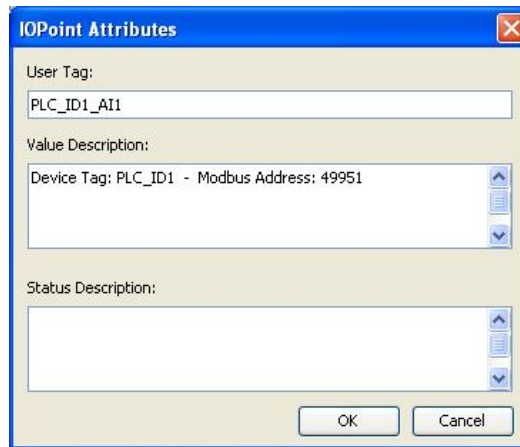
Na parte de cima da janela serão mostrados com a cor branca os *bytes* disponíveis. Clicando no *byte* aparecerá uma janela para configuração do ponto. Para o nosso exemplo do **Tag2** foi clicado no primeiro *byte* disponível. Segundo a janela de mapeamento ele corresponde ao endereço 49951 (Byte Offset 1).

Escolha o tipo de dados (**Data type**) da variável do equipamento escravo. Os tipos disponíveis dependerão do equipamento escolhido e deve ser obtido no manual do equipamento escravo. Para o nosso exemplo o tipo de dado do ponto 49951 é **Unsigned16**. Para pontos analógicos também é possível configurar escala para cada ponto. Mais informações sobre o **Data type** serão mostradas na seção Tópicos Avançados. A figura abaixo mostra a configuração de escala do ponto.



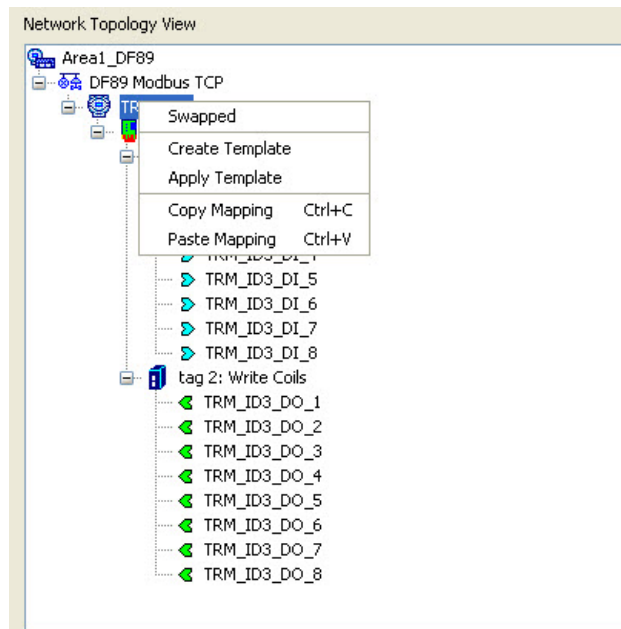
Figura 23. 18 – Configurando os parâmetros da escala

Clicando com o botão direito do mouse sobre o ponto pode-se excluí-lo, através da opção **Delete**, ou definir alguns atributos que o identifiquem melhor, através da opção **Attributes**. A **Mapping tool** cria automaticamente a descrição com o endereço Modbus. Isto é útil para identificar o ponto através do endereço Modbus dentro da lógica ladder. Preencha os campos necessários e clique **OK**.



**Figura 23. 19 – Alterando os atributos dos pontos mapeados**

Para os tipos analógicos com *data types* de 4 bytes (float, uns32 e signed32) é possível ainda escolher entre o dado no formato direto ou invertido (*swapped*). Esta informação do equipamento diz se ele usa dado no formato *swapped* ou não. Para configurar o *swapped* basta clicar com botão direito do mouse no equipamento na **Mapping tool** e escolher **Swapped** como mostrado na figura abaixo. Na seção Tópicos Avançados é explicado em detalhes o tipo de dado *swapped*.



**Figura 23. 20 – Exemplo de configuração**

Após mapear e configurar todos os pontos clique **OK** para fechar a **Mapping Tool**.



De volta ao **Syscon** salve a configuração.

## Configuração do DF89 como Mestre Modbus TCP

Nesta seção serão detalhados todos os passos para fazer a seguinte configuração:

- Adicionar equipamentos no canal TCP do DF89
- Adicionar pontos Modbus nos escravos
- Mapear os pontos Modbus para o **SYSTEM302**

Para o exemplo do processo da figura 23.1, na configuração como Mestre Modbus TCP é utilizado um transmissor escravo (PLC\_ID3).

Para configurar o canal Modbus TCP o procedimento é o mesmo usado no canal serial, partir do passo 5, em que a ferramenta de configuração dos escravos Modbus TCP também é o **SmarNetConf**. Para isso, na janela mostrada na figura 23.7 escolha expandir o canal TCP. Partindo da janela da área expanda o canal TCP (DF89 Modbus TCP). Selecione o canal (no alto da janela) e clique com o botão direito do mouse. Escolher a opção **Modify Configuration**.

A figura abaixo mostra a janela principal do **SmarNetConf** do canal TCP.

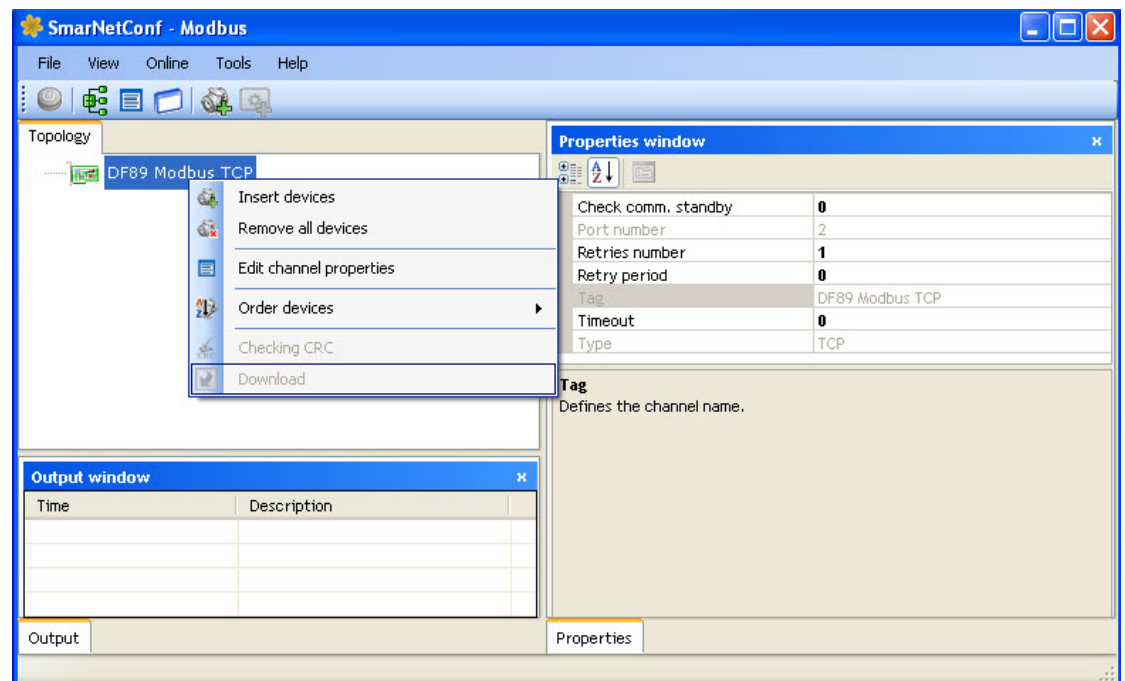
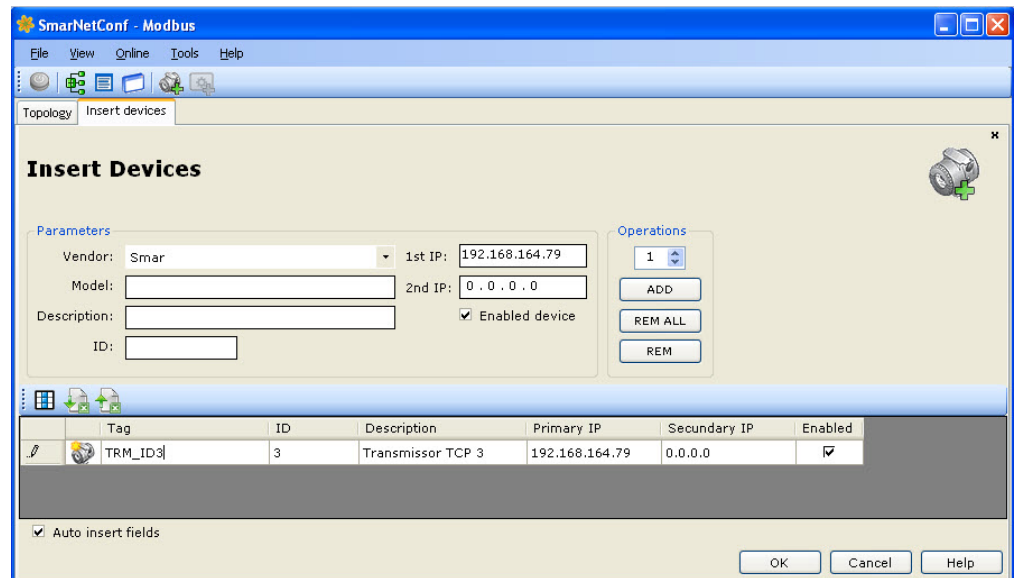


Figura 23. 21 – Janela principal do Configurador Modbus SmarNetConf

O parâmetro CHECK SECOND. COMM. possui funcionalidade semelhante ao parâmetro CHECK\_COMM\_STANDBY do bloco MBCF utilizado em outros controladores da linha DF1302. Assim o DF89 secundário (*standby*) realizará um teste de comunicação com os equipamentos escravos na periodicidade configurada neste parâmetro.

## Configurando os equipamentos Modbus

Para inserir um equipamento escravo, clique no botão **Insert Device** localizado na barra de ferramentas, ou através do menu **Insert→Device** ou no canal TCP com o botão direito do mouse como mostrado na figura acima. A janela mostrada na figura seguinte é a janela de inserção de equipamentos escravos, em que o usuário pode adicionar um ou vários equipamentos do mesmo tipo.



**Figura 23. 22 – Adicionando equipamentos na rede Modbus**

Nesta janela, deve ser selecionado o equipamento escravo que será inserido. Para o exemplo proposto, será incluído no canal somente um equipamento. Primeiramente obtenha as informações de configuração Modbus do equipamento escravo através de seu manual. Siga os passos abaixo para a inserção do equipamento escravo:

- A. Preencha **Model**, **Description** e **ID** do equipamento;
- B. Preencha o endereço IP do equipamento no campo **1st IP** (*first IP*). No caso do equipamento ser redundante será preenchido o **2nd IP** (*Second IP*) como o caminho redundante. Para mais detalhes veja a seção de redundância.
- C. Selecionar a quantidade de equipamentos de um determinado tipo. No nosso caso será somente 1.
- D. Clique **Add**. Neste momento será mostrada uma tabela com a quantidade de equipamentos selecionados.
- E. No campo **ID**, atribua um endereço para este equipamento.
- F. No campo **Tag** é dado o tag do equipamento. Caso o usuário não atribua um tag ao equipamento, será atribuindo um tag *default*;
- G. Clique **OK**. Então na janela principal será incluído o equipamento adicionado.

#### NOTA

Os endereços atribuídos assim como os tags dos equipamentos deverão ser únicos na rede Modbus e também no *Database* do **Studio302** (*Workspace*). Para maiores detalhes refira-se ao manual do **Studio302**.

## Editando as propriedades e os comandos do equipamento

Após a adição do equipamento, as propriedades dele podem ser alteradas. Para alterar as propriedades do equipamento, adicionar e apagar comandos siga os mesmos procedimentos descritos para o canal serial. As propriedades dos equipamentos e dos comandos são descritas em detalhes na seção Parâmetros do **SmarNetConf**.

Para o exemplo no canal TCP tem-se um equipamento (TRM\_ID3). A figura seguinte mostra a tela do **SmarNetconf** com o equipamento **TRM\_ID3** do exemplo proposto.



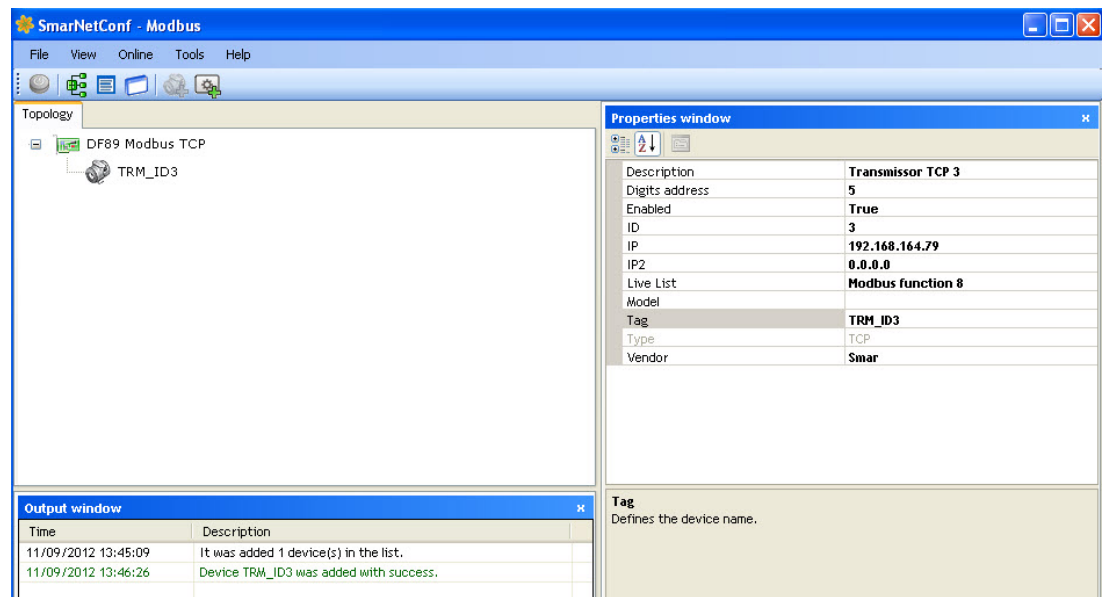


Figura 23. 23 – Exemplo das propriedades do canal TCP

## Fazendo lógica com os Pontos de E/S Modbus

Nesta seção serão detalhados todos os passos para fazer a seguinte configuração:

- Fazer uma lógica em blocos ou em ladder com os pontos Modbus do escravo
- Fazer download da configuração

Até agora foram criados os equipamentos escravos Modbus TCP e Serial e também mapeados os pontos. Os próximos passos mostram como utilizar os pontos mapeados na **Network Configurator** na lógica de controle. Como foi mencionado anteriormente existem duas formas para mapear as entradas e saídas – lógica ladder ou blocos funcionais. Cada ferramenta possui sua biblioteca de blocos específica. É possível misturar as duas lógicas através do bloco FFB 1131.

Inicialmente serão mostrados os passos para utilizar somente a Ladder. Após isso será explicado como montar a lógica usando blocos funcionais. A escolha de qual ferramenta usar é do usuário.

## Mapeando os pontos Modbus para serem utilizados na Ladder

### Passo 10

Como os pontos serão mapeados para utilização na ladder, voltando à figura 23.17, não será necessário utilizar a opção **Block Type** (None). Pode ser dado um tag para o grupo na opção **Group tag** ou deixar o tag *default* (*Group Tag* é importante para a utilização em blocos). Por fim é escolhido o **User Tag** que será o tag mostrado na lógica ladder. No nosso exemplo o **User Tag** do ponto 49951 será **PLC\_ID1\_AI1** mostrado na figura 23.25.

Algumas vezes (principalmente quando o **LogicView for FFB** já estava aberto) os pontos mapeados na **Mapping Tool** não aparecem no **LogicView for FFB**. Neste caso é necessário executar o comando **Refresh Data** para que os pontos de E/S configurados na **Mapping Tool** sejam atualizados na ladder. Clicando com o botão direito em **Network I/O** na janela **Hierarchy** do **LogicView for FFB**. Veja figura seguinte.

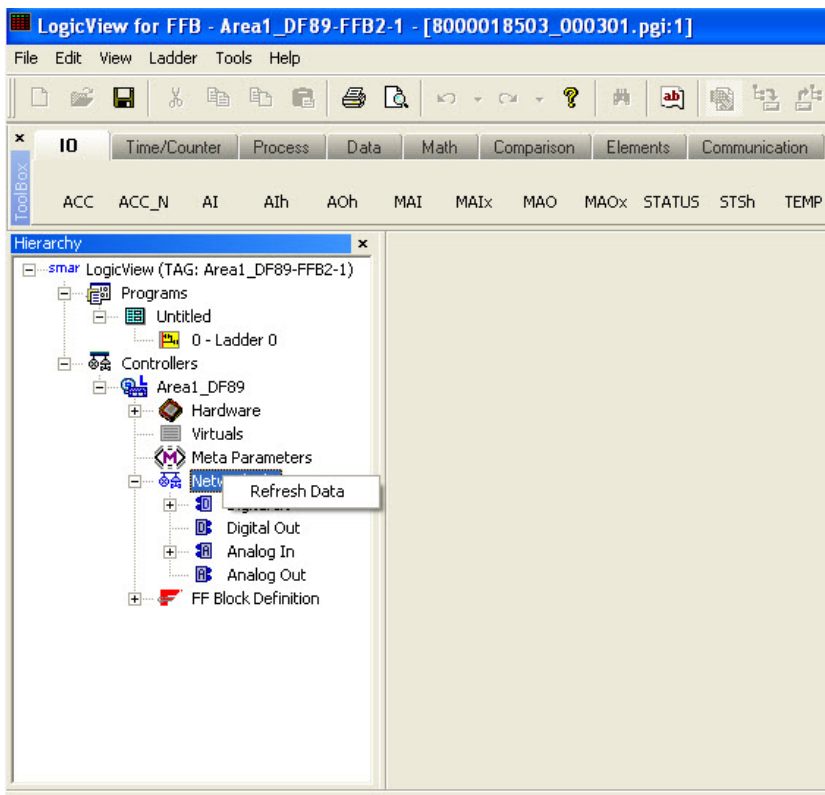


Figura 23. 24 – Realizando o Refresh Data

Os pontos de **NetIO** podem ser vistos na janela **Hierarchy** do **LogicView for FFB**. Na Janela **Object Properties** é gerado inicialmente uma descrição que contém o endereço Modbus da variável. Na figura abaixo, a janela do lado esquerdo é a **Hierarchy** com todos os pontos Modbus mapeados na **Mapping tool**. No **LogicView for FFB** são mostrados todos os pontos dos dois canais. Do lado direito da figura é mostrada a janela **Object properties** do ponto selecionado **PLC\_ID1\_AI\_1** que foi configurado no endereço Modbus 49951.

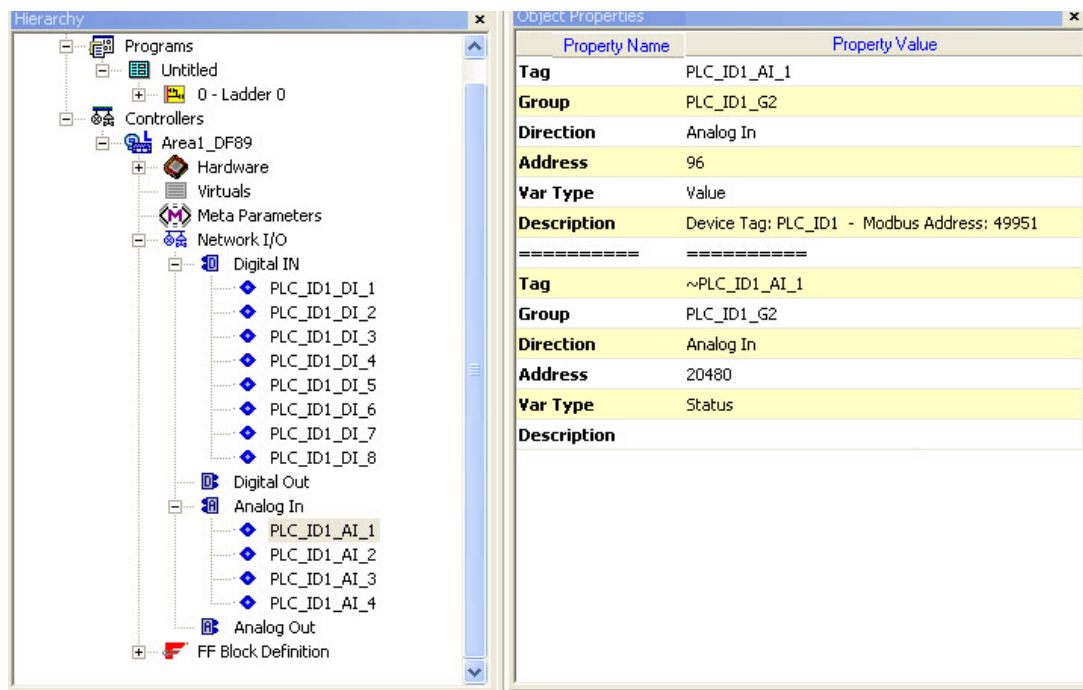


Figura 23. 25 – Network I/O no LogicView for FFB

Após este procedimento os pontos de E/S da rede Modbus já estão disponíveis para serem

utilizados na lógica. Para isso o usuário conta com uma biblioteca completa de funções que poderá utilizar na lógica. Para maiores detalhes veja o manual do **LogicView for FFB**.

Na figura abaixo são mostrados dois pontos Modbus utilizados na lógica ladder. O ponto discreto **PLC\_ID1\_D1\_1** e o ponto analógico **PLC\_ID1\_AI\_1**. Todos os pontos de E/S Modbus aparecem com o símbolo **(MB)** na lógica ladder.

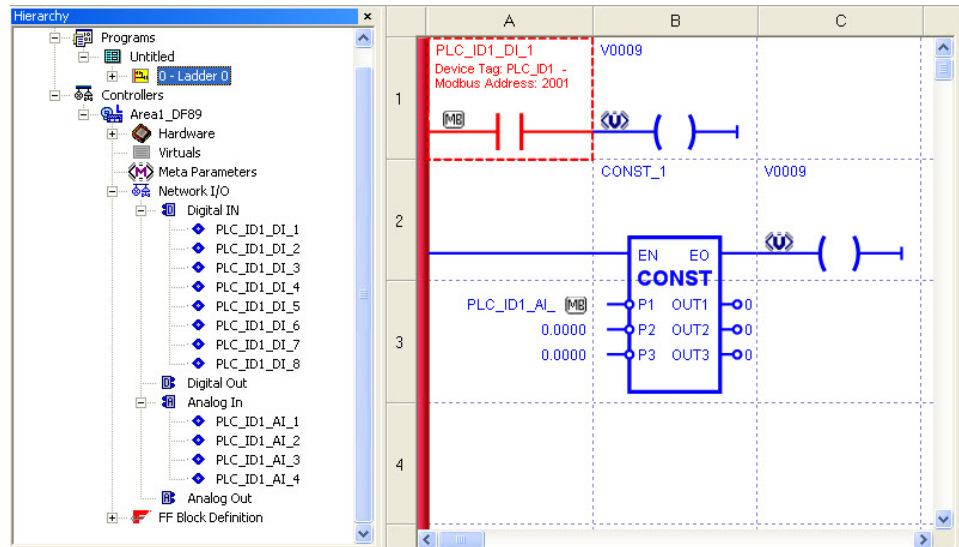


Figura 23. 26 – Editando a lógica

Aqui termina a configuração somente do canal serial usando ladder. Salve a configuração do **LogicView for FFB** e retorne para o **Syscon**.

Após a configuração da ladder o usuário poderá descarregar a configuração para o equipamento ou então escolher configurar blocos funcionais no **Syscon**. Caso queira trabalhar também com blocos funcionais é necessário seguir o passo 11 para a edição de blocos funcionais.

Seguindo o exemplo proposto ainda é necessário configurar também o canal TCP. Se desejar configurar o canal TCP sem utilizar blocos funcionais pule para o passo **Configurando Canal TCP**. Se desejar fazer o download somente da configuração até este ponto passe para o passo **Download de configuração**.

## Passo 11

### Mapeando os pontos Modbus para serem utilizados em Blocos Funcionais

Da mesma forma que foi feita a configuração através da ladder poderá ser feita a configuração utilizando blocos funcionais. Nesta seção será mostrado o mapeamento dos mesmos pontos de E/S Modbus do passo 10, mas agora utilizando blocos funcionais.

Para mapear os pontos de E/S em blocos, na janela da **Mapping Tool** como na figura 23.17, é necessário preencher o campo **Block Type**. Quando for escolhido **None**, como mostrado no passo 10, indica que o ponto somente será mapeado na ladder.

O usuário somente poderá escolher mapear os pontos de E/S em blocos funcionais do tipo múltiplas entradas analógicas ou discretas (MAI e MDI) ou múltiplas saídas analógica ou discreta (MDO e MAO).

Neste ponto o usuário está pronto para desenvolver a estratégia de controle. Neste caso ele poderá utilizar outros blocos funcionais para montar a estratégia de controle desejada. Para interligar os blocos ele precisa fazer links entre eles. Para abrir uma estratégia de controle no **Syscon** é necessário criar um módulo de controle (*Control Module*) e então as estratégias (*Strategy*). Para saber como montar uma lógica de controle no **Syscon** recomenda-se verificar seu manual.

Um parâmetro importante que deve ser configurado em todos os blocos funcionais é o parâmetro **MODE\_BLK** que determina a operação do bloco - Fora de Serviço (OOS), Manual (MAN) ou Automático (AUTO). Todos os blocos para esta área devem estar com parâmetro **MODE\_BLK.TARGET** em Auto. Cada bloco tem uma configuração específica dos parâmetros. Para maiores informações sobre cada bloco veja o manual de Blocos Funcionais.

#### NOTA

Para os blocos de E/S Modbus não é necessário configurar o parâmetro **CHANNEL**. Este parâmetro somente será utilizado quando forem usados os blocos de E/S para mapear pontos de E/S local (IMB).

## Comissionamento e download da configuração para o controlador

Após a configuração dos parâmetros, o usuário pode iniciar a comunicação com os equipamentos. É necessário fazer o comissionamento dos equipamentos para que os *tags*, IDs e endereços de cada equipamento sejam atribuídos adequadamente. Se este procedimento não for realizado, o **Syscon** detectará o equipamento não comissionado e o *download* deste equipamento será abortado. O comissionamento consiste em associar um controlador DF89 da rede com o configurador da configuração. E também será atribuído o mesmo tag da configuração ao controlador.

Para comissionar basta habilitar a comunicação online com os equipamentos da rede através do botão **ON** da barra de ferramentas do **Syscon** mostrado na figura abaixo.



Figura 23. 27 – Habilitando o modo Online

Então, clique no controlador ou no canal e escolha **Commission**. Mais informações sobre como fazer comissionamento refira-se ao manual do **Syscon**.

Finalizado o comissionamento dos equipamentos, pode-se iniciar o processo de *download*. O processo de *download* pode ser feito de vários pontos do **Syscon** e dependerá do que se deseja fazer - download de todos os controladores, download de somente de um controlador, etc.. Porém o download do **Syscon** é o mais recomendado por ser o mais completo, pois ele controla as alterações de todas as ferramentas.

No **Syscon**, há dois procedimentos de *download*: o modo convencional (não incremental) e o download incremental.

- O **download convencional** ou **não incremental**, a configuração inteira armazenada em cada equipamento é apagada e então a nova configuração é descarregada nos equipamentos.
- O **download incremental** inclui uma fase que compara a configuração armazenada no equipamento físico, na planta, com a configuração do **Syscon**, armazenado no *Database* do **Studio302**, e somente as diferenças são descarregadas, evitando que informações desnecessárias sejam enviadas para o equipamento. Para esta opção, o usuário pode limitar o escopo do download.

A figura abaixo mostra o download do equipamento, onde existe a opção de fazer o download incremental ou não (que neste caso seria o download total).



Figura 23. 28 – Fazendo o download

A operação de download pode afetar uma rede de equipamentos (quando seleciona uma rede Fieldbus, uma rede HSE ou um canal de um *Linking Device*), ou o download pode afetar um equipamento em particular, que pode ser um equipamento HSE (*Bridge*, *Linking Device*, *Controlador*) ou um equipamento H1. Neste último caso, a operação também é chamada de **download parcial**.

A operação de download só está disponível quando o **Syscon** está no modo **on-line**, comunicando com a planta e com os equipamentos que já foram comissionados.

#### NOTA

Se o usuário acabou de realizar um comissionamento o Syscon já sugere um *download*.

Às vezes, após o comissionamento é necessário clicar no botão **Refresh** ao lado direito do botão de online no *toolbar* do Windows.

Caso o usuário opte pelo *download* incremental ele deve selecionar a opção na janela acima. Então aparecerá uma janela com as opções de configuração do *download* incremental. Caso exista alguma das opções selecionadas significa que ele vai comparar inicialmente aquela opção com o controlador e somente vai fazer o *download* se existirem diferenças. Antes do *download*, o **Syscon** ainda vai mostrar uma janela com as diferenças para o usuário. Somente quando o usuário verificar as diferenças é que realmente será feito o *download* da nova configuração.

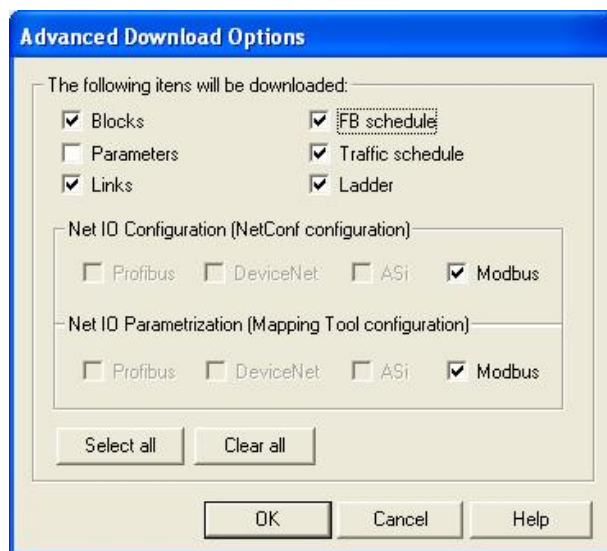


Figura 23. 29 – Selecionando as opções de download

No exemplo em questão deseja-se fazer o *download* somente do **PROJ\_DF89**. Então, clique sobre o controlador, com o botão direito do *mouse*, selecione a opção **Download**. Na janela de *download* selecionar **Start**.

Após a finalização do *download* o usuário já pode verificar no seu equipamento a supervisão dos

parâmetros da lógica. Para supervisão dos parâmetros (tanto da ladder quanto de blocos) ele primeiramente deve fazer um **Export tags**. Este procedimento indica que vai atualizar o banco de dados com as informações de supervisão desejada. O **Export tags** é feito na janela da área clicando com o botão direito no seu nome.

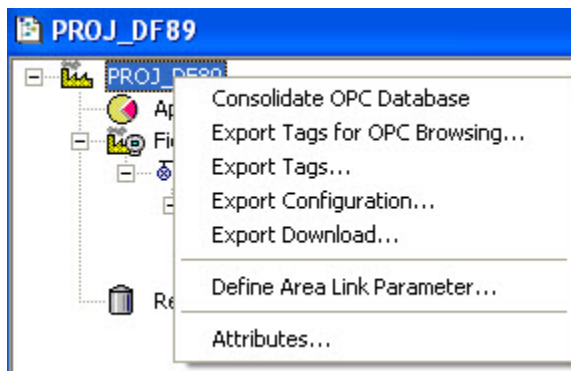


Figura 23. 30 – Fazendo o Export Tags

Após esta operação o usuário pode supervisionar os parâmetros da configuração. Novas alterações da lógica ladder ou de blocos poderão ser feitas. Somente quando for feita alguma nova inclusão de pontos de E/S ou de blocos funcionais deve ser repetido este procedimento de **Export Tags**.

## Configuração do DF89 como Escravo Modbus

O DF89 pode trabalhar como escravo Modbus Serial ou TCP. Neste caso qualquer equipamento Mestre Modbus pode ler pontos do próprio controlador DF89, sejam pontos de E/S convencionais, pontos de E/S Modbus de escravos “escaneados” pelo próprio DF89 ou outros pontos internos do DF89. No exemplo da figura 23.1 o DF89 é escravo Modbus de uma IHM Modbus TCP.

Para configurar o DF89 como escravo Modbus, primeiramente deve ser criado um bloco no **Syscon - Modbus Slave Configuration**. Para isso, clique com o botão direito do mouse sobre o **HSE\_FB\_VFB** do controlador DF89 como mostrado na figura seguinte.

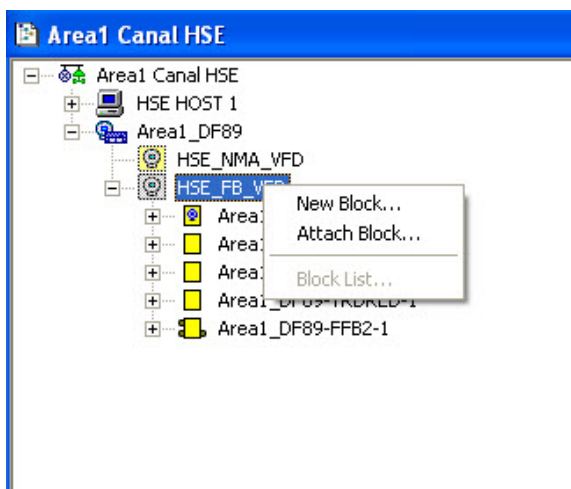


Figura 23. 31 – Incluindo um novo bloco no controlador

Então deve ser criado o bloco **Modbus Slave Configuration**. Este bloco representa toda a configuração do DF89 como escravo Modbus (Serial ou TCP).

Após a criação do bloco o usuário poderá configurar os parâmetros. Como escravo Modbus TCP os parâmetros principais são **MODE\_BLK** e o **DEVICE\_ADDRESS**. No exemplo da figura 23.1 o DF89 tem o endereço Modbus 25.

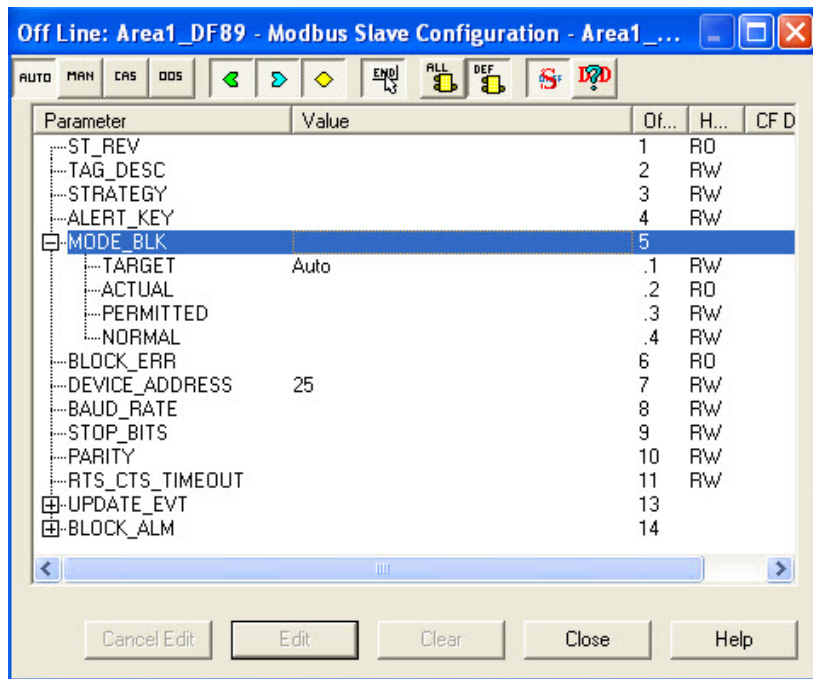


Figura 23. 32 – Parâmetros de configuração como Escravo Modbus

Para a configuração como escravo Modbus Serial siga o seguinte procedimento:

- 1) Não pode haver canal serial criado no **Syscon**. Isso porque o DF89 somente tem um canal serial. O canal serial como mestre tem prioridade sobre a função escravo.
- 2) Devem ser configurados os parâmetros **MODE\_BLK** e **DEVICE\_ADDRESS** no bloco **MBSCF**:
- 3) Devem ser configurados os parâmetros da mídia serial no bloco **MBSCF**: *Baudrate*, *StopBits* e *Paridade*.
- 4) Pode ser verificado se a serial está habilitada como escravo ou como mestre no parâmetro **SERIAL\_CHANNEL** do bloco **Modbus Transducer**.

Somente os pontos mapeados na lógica ladder podem ser mapeados como escravos Modbus. Cada ponto Modbus mapeado na ladder tem um respectivo endereço Modbus. Para obter informações do ponto clique nele e verifique a janela **Object Properties** do **LogicView for FFB** como mostrado na figura seguinte.

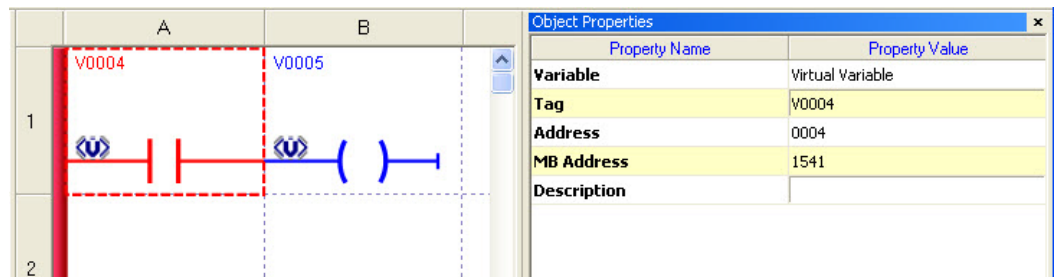


Figura 23. 33 – Exemplo de obtenção do endereço Modbus do ponto da Ladder

No exemplo da figura acima, o contato V0004 tem um endereço Modbus MB Address = 1541.

#### NOTAS

- 1) Todos os pontos analógicos são do tipo ponto flutuante (*float*).
- 2) Somente são admitidos funções Modbus de escrita múltiplas, ou seja, função 15 e 16.
- 3) O DF89 já sai de fábrica como escravo Modbus TCP e Serial no endereço 247 e com *baudrate* 9600/1/Even.

## Configuração do DF89 somente como *Bypass Modbus*

Outra característica do DF89 é trabalhar como **Bypass Modbus**. Ou seja, um equipamento mestre Modbus TCP pode acessar um equipamento escravo Modbus serial que esta abaixo do DF89.

A configuração do DF89 como *Bypass* é feito criando um canal serial. Se o usuário desejar somente trabalhar como *Bypass* basta criar o canal serial no **SmarNetConf** e configurar a mídia serial.

Para o exemplo da figura 23.1, o canal serial com equipamentos já foi criado, portanto ele já esta suportando o *bypass* automaticamente. Um exemplo seria a IHM Master TCP acessar qualquer ponto do escravo **TRM\_ID2** ou **PLC\_ID1** somente colocando o **ID** do respectivo equipamento (**ID 1** ou **ID2**) e o IP do DF89.

### NOTAS

- Para trabalhar como *Bypass* o endereço do escravo Modbus não pode ser o mesmo do **DEVICE\_ADDRESS** do bloco **MBSCF**.
- A configuração da mídia somente é feita no **SmarNetConf**. No caso do *bypass* pode ser utilizado o *download* pelo próprio **SmarNetConf** na modificação de qualquer parâmetro da mídia.
- O *Bypass* é suportado pelas duas portas Ethernet do DF89 - ETH1 e ETH2.
- No caso da redundância também é possível acessar a porta secundária como *Bypass*, bastando configurar o parâmetro do equipamento **BYPASS\_SECONDARY** no **SmarNetConf**.

## Tópicos avançados do Modbus

Existem duas especificações principais de Modbus: uma antiga, que foi criada pela empresa Modcon precursora do Modbus, segue o guia de referência **Modcon PI-MBUS-300** de 1996. A versão atual do Modbus segue a especificação da Modbus Organization - **Application Protocol Specification** - a partir de 2002. As duas especificações são totalmente compatíveis, somente existe uma diferença de visualização e tamanho da faixa de endereçamento.

O registro Modbus é um número de 2 bytes (Word) que indica um determinado parâmetro do equipamento escravo. Por exemplo, o endereço 1 indica um ponto de E/S, enquanto o endereço 2, indica outro ponto de E/S. Este endereçamento para o Modbus Modcon possui faixas específicas dependendo da função mostrada abaixo. Lembrando que a função no protocolo Modbus indica se é desejado trabalhar com leitura ou escrita de registro. As funções suportadas e a descrição de cada função esta na tabela 23.1.

- Função 1, 5, 15 – faixa endereçamento Modbus Modcon 0xxxx (0001 – 10000)
- Função 2 – faixa endereçamento Modbus Modcon 1xxxx (10001 – 20000)
- A Função 3, 6, 16 – faixa endereçamento Modbus Modcon 4xxxx (40001 – 50000)
- Função 4 – faixa endereçamento Modbus Modcon 3xxxx (30001 – 40000)

Para a versão atual também chamado de Modbus estendido, esta faixa não precisa mais existir já que o formato do *frame* já desconsiderava esta faixa. Isto fez com que a faixa de endereçamento não ficasse restrita aos 10000 registros da faixa.

Uma forma, muito adotada para referenciar estas faixas é pela quantidade de bytes no endereço Modbus, ou seja, se ele usa 5 ou 6 dígitos. Na tabela abaixo é mostrado uma comparação com os endereçamentos antigos e novos para alguns exemplos de endereços.

EXEMPLO DE ENTRADA DO USUÁRIO		VISÃO 5 DÍGITOS	VISÃO 6 DÍGITOS	MODBUS NO BARRAMENTO	
Função	Endereço Inicial	Endereço Inicial	Endereço Inicial	Função	Endereço inicial
1	2000	2000	2000	1	1999
3	10	40010	400010	3	9
2	2500	12500	102500	2	2499
4	1533	31533	301533	4	1532
3	15200	Não tem como visualizar	115200	3	15199

**Tabela 23. 2 – Exemplo de endereço Modbus no protocolo antigo (5 dígitos) ou Atual (6 dígitos) e como fica o protocolo internamente (Modbus na linha)**



A tabela anterior mostra exemplos de endereçamento Modbus usando as duas especificações Modbus: Modcon (com 5 dígitos) e atual (com 6 dígitos).

As colunas de entrada do usuário mostram um exemplo de determinada função e registro Modbus.

A coluna Modbus no barramento mostra como seria o endereçamento do protocolo Modbus. Esta coluna o usuário geralmente não vê, pois é algo interno do protocolo, para comunicação entre os equipamentos. Neste caso, o endereço inicial no barramento é sempre um endereço relativo descontado a base da função e é única para as duas representações.

O que pode ser observado na tabela 23.2 é que as duas visões do protocolo não têm diferenças para a comunicação, ou seja, elas geram o mesmo comando no barramento. Então o usuário escolhe qual a forma que ele quer trabalhar, na forma antiga ou na atual. A única diferença será quando o equipamento tiver uma faixa de endereços muito grande (maior que 10000 registros) e então somente será possível utilizar a forma de 6 dígitos.

#### NOTA

O DF89 suporta o protocolo Modbus TCP e o Modbus RTU serial. Para os dois protocolos o padrão dos comandos Modbus, segundo a especificação antiga ou atual, é o mesmo e são totalmente suportados pelo controlador. A visualização default para o DF89 é com 5 dígitos que é a forma ainda mais usada.

Existem outras extensões não padronizadas do protocolo Modbus como Enron e Daniel que não são suportadas pelo DF89.

## Tipos de dados suportados

No Modbus os pontos são mapeados em registros. Estes registros dependendo do tipo da variável diferem. Para tipos discretos (funções 1, 2, 5 ou 15), cada registro ocupa 1 bit de dados. No caso dos tipos analógicos (funções 3, 4, 6 ou 16), cada registro ocupa uma *word* (2 bytes). Além desta informação é necessário saber quais os tipos de dados que correspondem a estes registros. As informações dos tipos de dados (*data types*) suportados pelo Modbus do DF89 são os seguintes:

Função Modbus	Data types Suportados	Tamanho em Bytes	Nr de Registros Modbus Necessários
1, 2, 5 ou 15	BIT	1/8	1
3, 4, 6 ou 16	UNSIGNED 8	1	1
3, 4, 6 ou 16	UNSIGNED 16	2	1
3, 4, 6 ou 16	UNSIGNED 32	4	2
3, 4, 6 ou 16	FLOAT_SINGLE	4	2
3, 4, 6 ou 16	SIGNED 8	1	1
3, 4, 6 ou 16	SIGNED 16	2	1
3, 4, 6 ou 16	SIGNED 32	4	2

**Tabela 23.3 – Tipos de dados suportados**

Além destes tipos de dados, o dado analógico de tamanho de 4 bytes pode ser *swapped* ou não. Neste caso, o tipo de dados *swapped* tem uma inversão do dado (*littleendian*) com relação ao tipo de dado não *swapped* (*big-endian*). Esta informação é selecionada no equipamento dentro da **Mapping tool** e é válida para todo o equipamento.

Por exemplo, considerando um número de 4 bytes 12345678 alocados nos endereços Modbus 40001 e 40002:

Datatype	Registro 40001	Registro 40002
Unsigned32	5678	1234
Swapped Unsigned32	1234	5678

## Supervisão e Controle usando DF89

As variáveis Modbus no equipamento escravo podem ser mapeadas no DF89 pelo usuário nos seguintes cenários de uso: supervisão ou controle. Como supervisão, as variáveis são lidas ciclicamente via Modbus do equipamento escravo e armazenadas na memória do DF89 e podem ser escritas por demanda. Como controle, as variáveis são lidas ciclicamente (e não admitem escritas) ou escritas ciclicamente (e nunca são lidas).

Característica	Cenário de uso: Supervisão	Cenário de uso: Controle
Configuração de comando Modbus no SmarNetConf	Apenas comando Modbus de leitura: - Read Coil (comando 1) - Read Holding Register (comando 3)	Comando Modbus de leitura ou escrita: - Read Discrete Input (comando 2) - Read Input Register (comando 4) - Read Coil (comando 1) - Read Holding Register (comando 3) - Write Single Coil (comando 5) - Write Single Register (comando 6) - Write Multiple Coil (comando 15) - Write Multiple Register (comando 16)
Tipo de variável Modbus no equipamento escravo e tipo de comando Modbus possível.	Coil (endereço 0x.xxx ou 0xx.xxx) e Holding Register (endereço 4x.xxx ou 4xx.xxx) : Comando Modbus de leitura	Discrete Input (endereço 1x.xxx ou 1xx.xxx) e Input Register (endereço 3x.xxx ou 3xx.xxx) : Comando Modbus de leitura  Coil (endereço 0x.xxx ou 0xx.xxx) e Holding Register (endereço 4x.xxx ou 4xx.xxx) : Comando Modbus de leitura ou escrita
Variável modbus no DF89 (cache)	Discrete Input (endereço 1x.xxx ou 1xx.xxx) ou Input Register (endereço 3x.xxx ou 3xx.xxx) com tipo de dado float.	Se comando Modbus de leitura: Discrete Input (endereço 1x.xxx ou 1xx.xxx) ou Input Register (endereço 3x.xxx ou 3xx.xxx) com tipo de dado float.  Se comando Modbus de escrita: Coil (endereço 0x.xxx ou 0xx.xxx) ou Holding Register (endereço 4x.xxx ou 4xx.xxx)
Freqüência do envio do comando Modbus para leitura	Cíclico, conforme configuração do SmarNetConf.	Cíclico, conforme configuração do SmarNetConf.
Freqüência do envio do comando Modbus para escrita	Sob demanda do usuário. Quando ocorrer a escrita via Smar.DfiOleServer	Cíclico, conforme configuração do SmarNetConf.
Aplicação	DF89 funciona como <b>concentradora para um OPC Server/Client</b> (faz scan dos equipamentos escravos e guarda em memória do DF89 para acesso por um host). Apesar da variável modbus no DF89 ser um Discrete Input ou Input Register, a escrita via Smar.DfiOleServer será convertida em um comando Modbus de escrita para o equipamento escravo.	DF89 transfere ciclicamente uma variável : - de um equipamento Modbus para outro equipamento Modbus - de um equipamento Modbus para um equipamento FF - de um equipamento FF para um equipamento Modbus.

### Supervisão

Por exemplo, na IHM do exemplo da figura 23.1 deseja-se mapear o ponto 40020, que seria um *setpoint* de um PID, e um contato 2001 que seria um botão liga/Desliga de uma bomba. O ponto será criado no **SmarNetConf** e na **Mapping tool** como mostrado nos passos de 5 a 9 e ficaria da seguinte forma.

Função Modbus (SmarNetConf)	Start Address (SmarNetConf)	Tag Ponto (Mapping Tool)	Descrição
Read Coils (01)	2001	PLC_ID1_DI_1	Liga/Desliga Bomba
Read Holding Registers (03)	40020	PLC_ID1_AI_1	SP do PID

Estes pontos foram utilizados na ladder em uma determinada estratégia de controle e também poderiam ser mapeados no supervisor via Smar.DfiOleServer. No supervisor, o valor dos pontos **PLC\_ID1\_DI\_1** e **PLC\_ID1\_AI\_1** podem ser alterados pelo usuário, a alteração será mapeada em um comando Modbus de escrita (função 5, 6, 15 ou 16) e será enviada para o escravo. O valor da próxima leitura já deverá ser o valor realmente efetivado no escravo. Se o ponto continuar com o

valor anterior significa que a escrita não foi reconhecida pelo escravo.

Na ladder não é possível alterar o valor do *setpoint*, pois a alteração online da ladder não admite escrever em parâmetros de entrada como este.

### **Peer-to-Peer de ponto de E/S**

Um ponto mapeado como *peer-to-peer* deve ser lido de algum equipamento escravo e escrito em outro equipamento escravo.

Por exemplo, se deseja ler o valor do *setpoint* de **PLC\_ID1** e escrever no *setpoint* de **TRM\_ID2**. Então, neste caso, deve ser mapeado o *setpoint* de **PLC\_ID1** com uma função de leitura qualquer (função 1, 2, 3 ou 4) e mapear uma função de escrita do **TRM\_ID2** qualquer (função 5, 6, 15 ou 16). Desta forma, pode-se usar lógica ladder ou lógica de blocos funcionais para fazer com que o ponto mapeado como **PLC\_ID1.PID1\_SP** seja escrito em **TRM\_ID2.PID2\_SP**.

## **Tempo de scan entre comandos**

O DF89 possui a característica adicional de escolha do *scan time* por comando. Por *default* este parâmetro vem com o valor 500 ms e deve atender a maioria das aplicações. Este valor pode ser ajustado de 0 a 65500 ms (65,5 s).

O *scan time* Modbus permite o usuário colocar tempos diferentes para cada comando, sendo eles múltiplos de 100 ms. Para variáveis em que o processo precisa com uma periodicidade menor, como variáveis analógicas, podem ser colocadas em um *scan time* Modbus mais lento, por exemplo, em relação a pontos discretos.

O *scan time* também é útil para priorizar o *bypass*. O *scan time* somente se refere aos comandos determinados para o DF89 executar como mestre Modbus e não influencia em nada o *bypass* Modbus. Então, se for colocado um *scan time* de 1000 ms e se tem somente um comando, o DF89 fará o *scan* do comando a cada um segundo, o restante do tempo ele poderá atender o *bypass*. Ou seja, isso faz com que a supervisão usando *bypass* seja muito mais rápida, pois a banda para *bypass* é maior neste caso.

Este tempo é o tempo desejado pelo usuário. Porém, como o *scan* depende de outros comandos para ser executado às vezes este tempo não consegue ser efetivado, tendo um *scan* muito maior do que o desejado. Por exemplo, imagine que o usuário tem 5 comandos para realizar para um dado equipamento (5 comandos já contando a otimização). E suponha que o usuário deixou o tempo de 200 ms para todos os comandos. Se o tempo para cada comando for de 100 ms então o tempo para fazer todo o *scan* seria de 500 ms. Como o usuário configurou 200 ms (tempo desejado) o tempo nunca conseguirá ser realizado, pois o *scan* ocorre no mínimo a cada 500 ms. Agora se o usuário alterar o tempo para 1000 ms então existirá uma folga entre o *scan* de 500 ms.

## **Otimização de comandos**

O DF89 possui a característica adicional de otimização de comandos. Esta otimização consiste em diminuir o número de comandos enviados na linha e otimizar a comunicação Modbus. Por *default*, esta opção é sempre selecionada. Recomenda-se somente utilizar a opção não selecionada quando o escravo Modbus realmente não suportar os comandos otimizados.

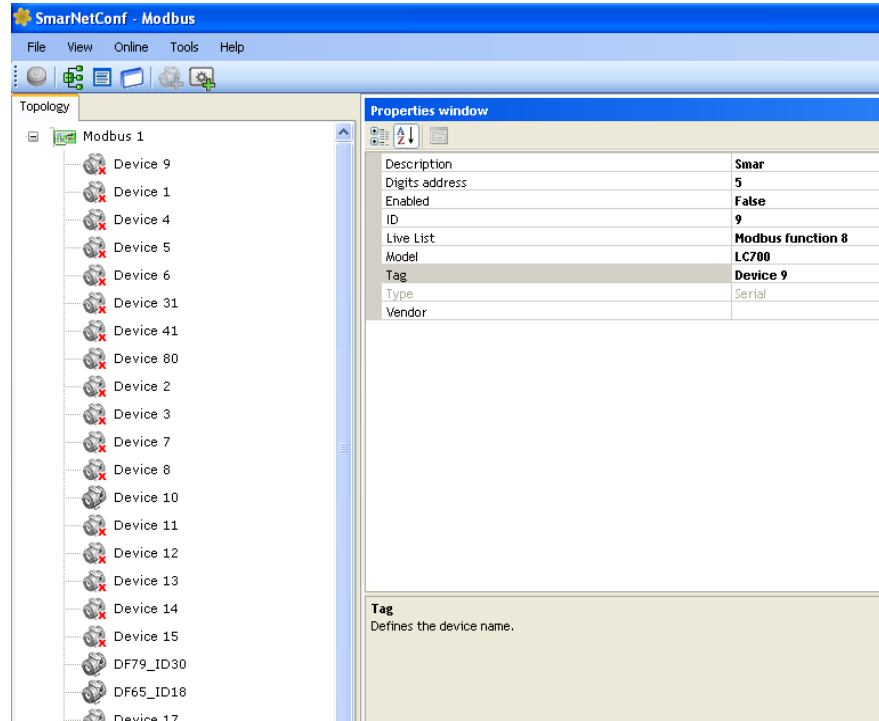
## **Habilitando e desabilitando um equipamento da configuração**

Outra característica interessante do **Network Configurator** é a possibilidade de habilitar e desabilitar um equipamento escravo da configuração sem afetar qualquer configuração de mapeamento ou lógica. Esta operação é realizada na janela de propriedades do equipamento em que o usuário habilita ou não o *scan* do equipamento.

Quando o equipamento está desabilitado, apesar de não serem alterados a configuração de lógica e mapeamento do equipamento, o controlador não realiza mais nenhuma operação com este equipamento, ou seja, não envia nenhum comando ao escravo.

A figura seguinte mostra um exemplo de configuração com equipamentos desabilitados. Quando o

equipamento está desabilitado aparece um (x) vermelho no desenho do equipamento. Na figura seguinte somente os equipamentos 10, 18 e 30 estão habilitados. Esta opção é muito útil quando existem equipamentos em manutenção.



**Figura 23. 34 – Exemplo de configuração com equipamentos habilitados e desabilitados**

Esta alteração de habilitação precisa de um download para que o controlador a reconheça. Este download pode ser o do próprio **Smar Network Configuration** ou um download do **Syscon**.

Para habilitar novamente o equipamento, retorne à janela de propriedades do equipamento e faça novo download.

## Comunicação online do Network Configurator com o controlador

É possível comunicar diretamente com o DF89 através do **SmarNetConf**. Esta comunicação pode facilitar algumas funções como download de parâmetros do canal, do escravo, entre outras. Esta comunicação é realizada através da rede Ethernet.

Para habilitar a comunicação Modbus pelo **SmarNetConf** primeiramente habilite a comunicação com o servidor OPC.

## Procedimento de Manutenção do SYSTEM302

Nesta seção é apresentado um resumo dos procedimentos de download que são recomendados nos casos de manutenção de uma configuração usando o **SYSTEM302**.

Como no **SYSTEM302** existem várias formas de download aqui é recomendada uma forma geral para atender a maioria dos casos. Porém outras formas podem existir para proceder a mesma alteração, consulte o manual do **Syscon**.

Na coluna **Download Recomendado** é referenciado o download incremental do **Syscon** na planta ou no controlador. Estas operações são explicadas em detalhes no manual do **Syscon**.

EVENTO MANUTENÇÃO	FERRAMENTA ALTERAÇÃO	DOWNLOAD RECOMENDADO
Fazer uma nova configuração do SYSTEM302 com controlador Modbus e com lógica ladder. Bloco FFB não tem links HSE para outros controladores.	Syscon, Network Configurator, Mapping Tool, LogicView for FFB	No Syscon fazer download incremental da planta com todas as opções selecionadas ou um download no controlador. (*)
Incluir um novo equipamento ou incluir um novo comando na rede Modbus.	Network Configurator, Mapping Tool	No Syscon fazer download incremental da planta com todas as opções selecionadas. (*)
Alterar configuração do equipamento da rede Modbus (endereço do equipamento, endereçamento dos pontos Modbus)	Network Configurator	No Syscon fazer download incremental no controlador. (*)
Mudar parâmetros específicos do canal Modbus ( <i>baud rate, timeout</i> ou outro parâmetro).	Network Configurator	No Network Configurator fazer download da rede Modbus. (*)
Habilitar ou desabilitar equipamento Modbus mantendo toda a configuração Modbus.	Network Configurator	No Network Configurator fazer download da rede Modbus. (*)
Acrescentar parâmetros ao bloco FFB e construção de links HSE ou somente acrescentar parâmetros FFB para um FFB que já possui link HSE.	Syscon	No Syscon fazer download incremental da planta. (*)
Incluir ou remover um link FFB para blocos internos dentro do mesmo controlador.	Syscon	No Syscon fazer download incremental no controlador sem as opções NetIO habilitadas.
Mudar parâmetros de mapeamento dos pontos de E/S (escala, tags do ponto, alteração de bytes acessados ou <i>swapped</i> ).	Mapping Tool	No Syscon fazer download incremental no controlador com opção Net IO Parametrization habilitada.
Mudar parâmetros de configuração de equipamento escravo Modbus (tag, modelo ou número de dígitos de visualização) ou habilitar ou desabilitar equipamento escravo.	Network Configurator	No Network Configurator fazer download da rede Modbus.
Alteração somente da lógica ladder	LogicView for FFB	No LogicView for FFB fazer download da lógica.

**Tabela 23.4 – Procedimentos de download recomendados**

\* Estas operações irão parar o canal de rede Modbus.

## Parâmetros do Smar Network Configurator

### Parâmetros do Canal Serial

Categoria	Parâmetro	Faixa Válida	Default	Unidade	Descrição
Advanced					
	RTS CTS Timeout	0-65535	0	ms	Especifica o tempo máximo de espera para CTS se tornar ativo após enviar RTS nas portas seriais
	Time delay	0-65535	0	ms	Tempo de espera entre uma recepção (Rx) e a próxima transmissão (Tx).
Media					
	Baud rate	0:110, 1:300, 2:600, 3:1200, 4:2400, 5:4800, 6:9600, 7:19200, 8:38400, 9:57600, 10:115200	9600	E	Define o baud rate das portas seriais.
	Parity	0:Even, 1:Odd, 2:None.	Even	E	Define a paridade das portas seriais.
	Retry Count	0-5	1		Número de retransmissões se o mestre não receber uma resposta do escravo.
	Retry period	0-65536	10000		Tempo para fazer nova verificação (livelist) para os equipamentos que falharam o scan.
	Stop bit	0:1, 1:2	1	E	Define o número de stop bits para as portas seriais.
	Timeout	0-65536	1000	ms	Tempo a se esperar por uma resposta do escravo após enviar um comando para as portas seriais.
Misc					
	Port number				1
	Tag				Tag do Canal
	Type				Serial
Redundancy					
	Check second. comm.	0-Desabilita teste 1-65535 tempo de intervalo entre cada teste	0	ms	Parâmetro configurado para a porta secundária (redundância) se usuário desejar o teste da comunicação entre os equipamentos escravos. Este será o tempo em MS da checagem da livelist.
	Secondary Bypass	True/False	False		Parâmetro configurado para a porta secundária (redundância) se desejar utilizar o caminho como bypass. (um mestre TCP acessar o escravo serial abaixo do DF89).

### Parâmetros do Canal TCP

Categoria	Parâmetro	Faixa Válida	Default	Unidade	Descrição
Media					
	Retry Count	0-5	1		Número de retransmissões se o mestre não receber uma resposta do escravo.
	Retry period	0-65536	0	ms	Tempo para fazer nova verificação (livelist) para os equipamentos que falharam o scan.
	Timeout	0-65536	1000	ms	Tempo a se esperar por uma resposta do escravo após enviar um comando para as portas TCP.
Misc					
	Port number			(read only)	2
	Tag			(read only)	Tag do Canal
	Type			(read only)	TCP
Redundancy					
	Check second. comm.	0-Desabilita teste 1-65535 tempo de intervalo entre cada teste	0	ms	Parâmetro configurado para o DF89 secundário. Se o usuário desejar o teste da comunicação periódico com os equipamentos escravos para avaliar a comunicação.
	Secondary Bypass	0 – False 1 – True	False		

## Parâmetros do Equipamento Serial

Categoria	Parâmetro	Faixa Válida	Default	Unidade	Descrição
Advanced					
	Digits address	5 ou 6	5		Indica se o campo "Modbus_Address" mostrará o número com 5 ou 6 dígitos indicando se é o padrão antigo (Modbus Modicon) ou novo.
	Live List	1 – Use Modbus function 8. 2 – Use the First command	1		Este parâmetro indica qual comando será utilizado pelo mestre para checar se o equipamento está vivo (livelist). Por default é utilizado o comando de loop back (função 8), porém pode ser que o equipamento não suporte este comando. Neste caso o usuário poderá optar pela opção 2, então, será enviado o primeiro comando configurado.
Main					
	Enabled	0 – False 1 – True	1		Este campo define se o comando deve ser executado e em que condições. Se o parâmetro estiver configurado como 0, o comando está desabilitado e não será executado na sequência normal de polling. Se o parâmetro estiver configurado como 1, o comando será executado a cada scan da lista de comandos.
	Slave address	1-247			Este parâmetro especifica o endereço do nó, na rede, do escravo Modbus que será considerado. Valores 1 a 247 são permitidos.
Misc					
	Description				
	Model				
	Tag				Tag do equipamento.
	Type				Serial
	Vendor				

## Parâmetros do Equipamento TCP

Categoria	Parâmetro	Faixa Válida	Default	Unidade	Descrição
Advanced					
	Digits address	5 ou 6	5		Indica se o campo "Modbus_Address" mostrará o número com 5 ou 6 dígitos, indicando se é o padrão antigo (Modbus Modicon) ou novo.
	Live List	1 – Use Modbus function 8. 2 – Use the First command	1		Este parâmetro indica qual comando será utilizado pelo mestre para checar se o equipamento está vivo (livelist). Por default é utilizado o comando de loop back (função 8), porém pode ser que o equipamento não suporte este comando. Neste caso, o usuário poderá optar pela opção 2, então, será enviado o primeiro comando configurado.
Main					
	Enabled	0 – False 1 – True	1		Este campo define se o comando deve ser executado e em que condições. Se o parâmetro estiver configurado como 0, o comando está desabilitado e não será executado na sequência normal de polling. Se o parâmetro estiver configurado como 1, o comando será executado a cada scan da lista de comandos.
	Slave address	1-247			Este parâmetro especifica o endereço do nó, na rede, do escravo Modbus que será considerado. Valores 1 a 247 são permitidos.
	IP		Spaces		Endereço IP Principal do equipamento
	IP2		Spaces		Endereço IP Redundante do equipamento. Ver casos de redundância.
Misc					
	Description				
	Model				
	Tag				Tag do equipamento.
	Type				TCP
	Vendor				

## Parâmetros dos Comandos do equipamento (serial ou TCP)

Parâmetro	Faixa Válida	Default	Unidade	Descrição
Tag				Tag do equipamento.
Function	As opções de acordo com tabela de funções Modbus suportadas.	1		Este parâmetro especifica a função Modbus que será executada pelo comando (leitura, escrita de pontos analógicos ou discretos). Estes códigos de funções são definidos pelo protocolo Modbus.
Offset	0 a 65535	0		Este parâmetro especifica o registro inicial Modbus ou o endereço do ponto digital a ser considerado pelo comando no equipamento escravo Modbus. Refira-se à documentação de cada equipamento escravo na rede para atribuição de seus registros e endereços de pontos digitais.
Registers	0 a 65535	0		Este parâmetro especifica o número de registros e pontos digitais a serem associados ao comando. Funções 5 e 6 ignoram este campo e elas somente se aplicam a um único ponto de dados. Para funções 1, 2 e 15, configure o número de registros a serem associados ao comando. Para funções 3, 4 e 16, configure o número de registros analógicos a serem associados ao comando. Obs.: registro que pode ser associado a ponto depende do tipo de dado. Ver mais sobre associação de registros e pontos no item tipo de dados suportados descrito nesta seção.
Scan	0 a 65500 (múltiplos de 100)	500	ms	Este parâmetro especifica o intervalo mínimo para executar comandos periodicamente. Este parâmetro deve ser múltiplo de 100. Portanto, se o scan time é 500, o comando será executado no MÍNIMO a cada 500 ms.
Optimization	0 – False 1 – True	1		Quando o parâmetro é true (selecionado), indica que será utilizada a otimização do comando Modbus, em que o algoritmo interno tentará agrupar vários pontos do mesmo tipo e do mesmo equipamento no mesmo comando.
Start addr.				Este campo é uma composição da função e do Relative_Address, que será o endereço Modbus na linguagem do usuário. Ele dependerá também do parâmetro Nr_digits_addr. Veja montagem do Modbus_Address.

### Funções Modbus Suportadas

A tabela abaixo mostra todas as funções Modbus suportadas pelo DF89 e sua respectiva descrição. Este código de função segue a especificação Modbus atual da Modbus Organization:- **ApplicationProtocolSpecification**.

Função Modbus	Parâmetro MODBUS_FUNCTION	Classe	Descrição
1	Read Coils (0x)	RW	Leitura de Pontos Discretos de Entrada/Saída.
2	Read Discrete Inputs (1x)	RO	Leitura de Pontos Discretos de Entrada
3	Read Holding registers (4x)	RW	Leitura de Pontos Analógicos de Entrada/Saída.
4	Read Input registers (3x)	RO	Leitura de Pontos Analógicos de Entrada
5	Write Single Coil (0x)	W	Escrita de Ponto Discreto (Somente 1 registro)
6	Write Single Register (4x)	W	Escrita de Ponto Analógico (Somente 1 registro)
15	Write Multiple Coils (0x)	W	Escrita de Pontos Discretos (1 ou mais registros)
16	Write Multiple Registers (4x)	W	Escrita de Pontos Analógicos (1 ou mais registros)

RW – Read and Write – suporta leitura e escrita; RO; Read Only – Somente leitura; W –Somente Escrita.

Além destas funções também é suportada a função de checagem do equipamento (função 8 – check). Esta função é utilizada na inicialização do equipamento para saber se ele está vivo ou não (livelist). No parâmetro **Livelist** do equipamento o usuário poderá escolher entre usar esta função ou a primeira função definida na tabela de comandos.



## Blocos específicos do Controlador DF89

### Modbus Communication Transducer

Este bloco provê as seguintes funcionalidades:

- Live list dos equipamentos escravos;
- Diagnóstico dos equipamentos escravos;
- Configuração online de parâmetros de configuração dos escravos.

Veja na tabela a seguir a descrição dos parâmetros.

IDX	Parâmetro	Tipo de dados	Faixa Válida	Default	Memória	Modo	Unidade	Descrição
1	ST_REV	Uns16			S	RO		
2	TAG_DESC	Octet			S	RW		
3	STRATEGY	Uns16			S	RW		
4	ALERT_KEY	UNS8			S	RW		
5	MODE_BLK	DS_69			S	RW		
6	BLOCK_ERR	Bitstr			D	RO		
7	SERIAL_CHANNEL	Uns8	0 : Slave 1 : Master	0	S	RO		Mostra se a porta serial está sendo usada como mestre ou escravo Modbus.
8	BAUD_RATE	uns8	3:1200, 4:2400, 5:4800, 6:9600, 7:19200, 8:38400, 9:57600, 10:115200	9600	S	RO		Mostra o baud rate atual usado na porta serial mestre ou escrava.
9	STOP_BITS	uns8	0:1; 1:2	1	S	RO		Mostra o stop bit atual usado na porta serial mestre ou escrava.
10	PARITY	uns8	0:Even 1:Odd 2:None	Even	S	RO		Mostra a paridade atual usada na porta serial mestre ou escrava.
11	SERIAL_TIMEOUT	Uns16	Positive	1000	S	RO	ms	Tempo de espera pela resposta do escravo usando a porta serial.
12	SERIAL_NR_RETRIES	Uns8	Positive	2	S	RO		Número de tentativas de reenvio do comando que falhou na porta serial.
13	SERIAL_RETRY_PERIOD	Uns16	Positive	5000	S	RO	ms	Período de tempo para reenviar o comando para o escravo que não está respondendo.
14	RTS_CTS_TIMEOUT	uns16	Positive	0	S	RO	ms	Especifica o tempo máximo de espera para CTS se tornar ativo depois de configurado RTS nas portas seriais.
15	TIME_DELAY	uns16	Positive		S	RO	ms	Tempo de espera entre a recepção (Rx) e a próxima transmissão (Tx).
16	SERIAL_CHK_COM_SEC	uns16	Positive		S	RO	sec	Indica para a redundância se foi configurado que será feito checagem do caminho secundário.
17	BYPASS_SECONDARY	uns8	0=False; 1=True		S	RO		Indica para Redundância se o usuário pode usar ou não o caminho secundário como bypass.
18	SERIAL_ENABLE_DEVS	Uns8			D	RO		Lista dos equipamentos habilitados na rede serial. O número representa o device_address do escravo.
19	SERIAL_LIVE_LIST	Uns8			D	RO		Lista dos equipamentos conectados na rede serial. O número representa o device_address do escravo.

20	SERIAL_ERR_CNT	Uns32		0	D	RO		Contador do número de erros seriais. Use o RESET_COUNTERS para reiniciar este contador.
21	SERIAL_SCAN_TIME	Uns32		0	D	RO		Tempo usado pela porta serial para realizar o scan das variáveis Modbus.
22	TCP_TIMEOUT	uns16	Positive		S	RO		Tempo de espera pela resposta de um escravo depois de enviar um comando na porta TCP.
23	TCP_NR_RETRIES	uns8	Positive		S	RO		Número de retransmissões se não for recebida a resposta do escravo.
24	TCP_RETRY_PERIOD	uns16	Positive		S	RO		Período de tempo para reenviar o comando para o escravo que não está respondendo.
25	TCP_CHK_COM_SEC	uns16	Positive		S	RO	s	Indica para a redundância se foi configurado que será feito checagem do caminho secundário.
26	TCP_ENABLE_DEVS	Uns8			D	RO		Lista dos equipamentos habilitados na rede TCP. O número representa o device_address do escravo.
27	TCP_LIVE_LIST	Uns8			D	RO		Lista dos equipamentos conectados na rede TCP.
28	TCP_ERR_CNT	Uns32		0	D	RO		Contador do número de erros TCP. Use o RESET_COUNTERS para reiniciar este contador.
29	TCP_SCAN_TIME	Uns32		0	D	RO		Tempo usado pela porta TCP para realizar o scan das variáveis Modbus.
30	RESET_COUNTERS	Uns8			D	RW		Reinicia os contadores de erros TCP e Serial.
31	NUM_LINKS	Uns8		1	S	RO		Define o número de barramentos suportados. Uso interno da LiveList.
32	LINK_SEL	Uns8	0: First ; 1: Next ; 2: Previous ; 3: Last	First	D	RW		Seleciona qual barramento terá suas informações mostradas. Uso interno da LiveList.
33	LINK_ID	uns16		0	D	RW		Identifica o barramento atualmente selecionado. Uso interno da LiveList.
34	LINK_ID_REV	Uns32		0	S	RO		Revisão da Live list do barramento atual. Uso interno.
35	NUM_DEV	Uns16		0	D	RO		Número de equipamentos no barramento atual para uso interno da livelist.
36	SLAVE_SELECTOR	Uns8	0: First ; 1: Next ; 2: Previous ; 3: Last	First	D	RW		Seletor do endereço escravo para ler a informação da LiveList. Uso interno.
37	SLAVE_ADDR	uns8	1 to 247	0	D	RW		Endereço do parâmetro do escravo para Live List. Uso interno.
38	UPDATE_EVT	DS-73			D			
39	BLOCK_ALM	DS-72			D			

## Modbus Slave Configuration

Veja na tabela a seguir a descrição dos parâmetros.

Idx	Parâmetro	Tipo de dados	Faixa Válida	Default	Memória	Modo	Unidade	Descrição
1	ST_REV	UNS16		0	S	RO		
2	TAG_DESC	OCT(32)		Spaces	S	RW		
3	STRATEGY	UNS16		0	S	RW		
4	ALERT_KEY	UNS8	1 to 255	0	S	RW		
5	MODE_BLK	DS_69	OOS , AUTO	OOS	S	RW		
6	BLOCK_ERR	Bitstr(2)			D	RO		
7	DEVICE_ADDRESS	uns8	1 to 247	1	S	RW		Define o endereço Modbus DF1302 quando trabalhando como escravo Modbus.
8	BAUD_RATE	uns8	0:110, 1:300, 2:600, 3:1200, 4:2400, 5:4800, 6:9600, 7:19200, 8:38400, 9:57600, 10:115200	9600	S	RW		Define o baudrate da porta serial do DF1302 quando trabalhando como escravo Modbus
9	STOP_BITS	uns8	0:1, 1:2	1	S	RW		Define o stop bit da porta serial do DF1302 quando trabalhando como escravo Modbus
10	PARITY	uns8	0:Even, 1:Odd, 2:None	Even	S	RW		Define a paridade da porta serial do DF1302 quando trabalhando como escravo Modbus
11	RTS_CTS_TIMEOUT	uns16		0	S	RW	ms	Quando o valor é diferente de zero indica que a função RTS_CTS está habilitada e o valor é o timeout da conexão. Usado quando o DF1302 está trabalhando como escravo Modbus.
12	ENABLE_CHK_COM_SEC	Uns8		0-False ; 1-True	S	RW		Quando o DF1302 está trabalhando como escravo Modbus, este flag habilita a redundância, checka a comunicação no secundário.
13	UPDATE_EVT	DS-73			D			
14	BLOCK_ALM	DS-72			D			

## Redundância e Modbus

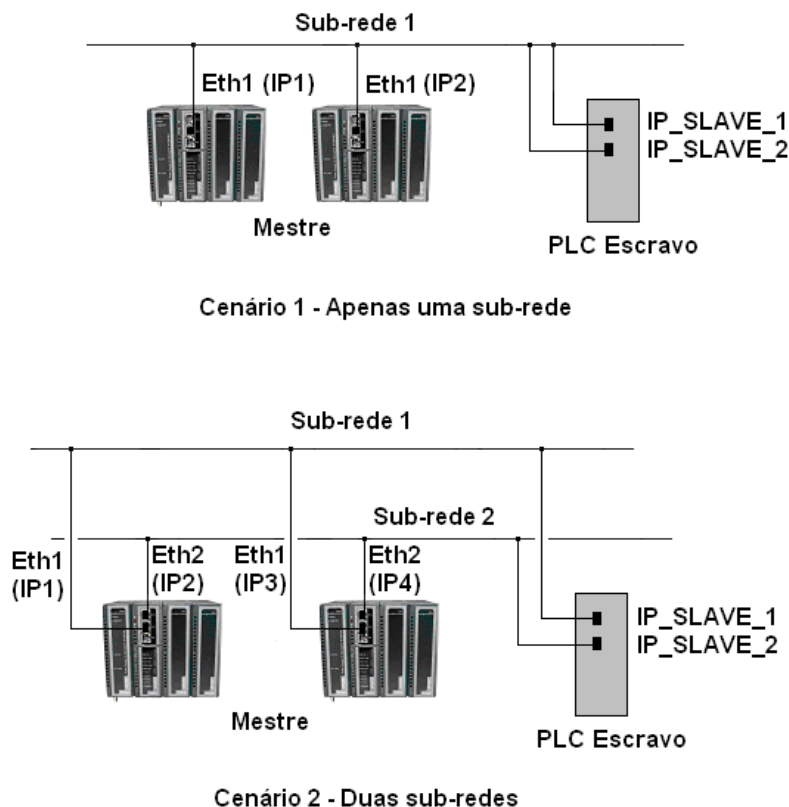
A redundância no Modbus pressupõe um grande número de cenários possíveis. Alguns são suportados outros não. As restrições que devem ser consideradas são:

1. A comunicação Modbus TCP no DF89 se estabelece em ambas as portas de comunicação (ETH1 e ETH2). Porém, recomenda-se o uso das duas portas apenas nos cenários em que o DF89 atue como escravo. O DF89 atuando como mestre não deveria ser configurado para utilizar a ETH2 sob pena de duplicar o tráfego da comunicação e prejudicar sua performance.
2. Quando o DF89 operar somente como escravo, existirá resposta às requisições Modbus nas duas portas Ethernet (ETH1 e ETH2).
3. As condições de *switchover* são relacionadas às “*Bad conditions*” no controlador primário (veja a descrição dessas “*Bad conditions*” na seção Adicionando Redundância aos controladores DFI302 HSE). Como escravos Modbus, os controladores não vão gerar uma condição de *switchover* por causa de falhas de comunicação. É sempre o mestre que alterna em tais condições. Quando usando TCP e todos os IPs na mesma sub-rede o mestre deve decidir quando chavear a requisição para outra conexão Ethernet escrava.
4. Como mestre Modbus, o DF89 (tanto TCP quanto serial RTU) somente chavearão se o primário não tiver nenhuma resposta de todos os módulos escravos.

### DF89 Redundante como mestre e PLC com duas portas Ethernet como escravo, usando Modbus TCP

Há dois cenários possíveis, no primeiro apenas uma sub-rede é empregada e todos os IPs dos equipamentos encontram-se na mesma sub-rede.

No segundo cenário, duas sub-redes são empregadas sendo que cada equipamento possui uma porta Ethernet conectada a uma das sub-redes, veja figura seguinte. O segundo cenário é o mais recomendado caso a disponibilidade da comunicação Modbus seja um fator importante, pois neste caso havendo a falha de um *switch* a outra rede/sub-rede cobrirá esta falha.



**Figura 23. 35 - Topologias de rede para DF89 Mestre redundante e PLC escravo com duas portas Ethernet**

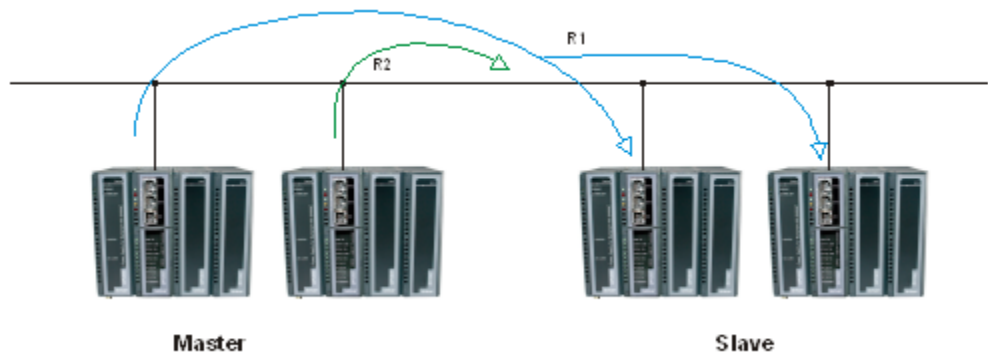
O PLC redundante tanto pode utilizar dois cartões Ethernet ou um cartão com duas portas Ethernet.

Uma vez que o DF89 primário executa as requisições, assume-se que qualquer uma das conexões com o PLC escravo pode receber e processar a requisição. Se uma conexão não está operando adequadamente, o DF89 primário usará outra conexão disponível (chaveamento de conexão).

Caso o DF89 primário não tenha resposta para nenhuma das conexões e o DF89 secundário tenha resposta para pelo menos uma das conexões, o par DF89 chaveará (cenário de *Bad Condition*/ chaveamento de controlador).

**DF89 Redundante como mestre e PLC redundante como escravo, usando Modbus RTU**

Neste cenário, o DF89 é redundante, usa a porta serial RS-485 e está conectado via *multidrop* ao PLC redundante (ambas as portas). Uma vez ativo, o DF89 executa as requisições (veja R1 na figura seguinte). Assume-se que somente um escravo processará a requisição. Em caso de alguma “*Bad condition*” o DF89 chaveará (veja R2 na figura seguinte).



**Figura 23. 36 - DF89 Mestre redundante e PLC redundante escravo, usando Modbus RTU**

**DF89 Redundante como escravo e PLC redundante como mestre, usando Modbus TCP**

Neste cenário, os DF89s primário e secundário respondem a todas as leituras requisitadas pelo PLC mestre. A escrita será executada somente pelo DF89 primário. Se por qualquer motivo o DF89 secundário recebe uma escrita, então esta requisição será encaminhada via caminho redundante do DF89 secundário para o DF89 primário.

**DF89 Redundante como escravo e PLC redundante como mestre, usando Modbus RTU**

Neste cenário, apenas o DF89 primário responderá às leituras e escritas requisitadas pelo PLC mestre.

