MANUAL DO USUÁRIO

Processador Modbus RTU e TCP/IP



MB700MP

MAR / 07

MB-700

FOUNDATION

Smar - MB-700

First in Fieldbus

smar

www.smar.com.br

Especificações e informações estão sujeitas a modificações sem prévia consulta.
Informações atualizadas dos endereços estão disponíveis em nosso site.

web: www.smar.com/brasil2/faleconosco.asp

INTRODUÇÃO

O MB-700 é um módulo multifunção que pode ser usado isolado ou integrado ao *SYSTEM302*. O módulo pode realizar diversas funções utilizando o protocolo MODBUS, como *gateway* MODBUS TCP/IP e MODBUS RTU, concentrador de dados MODBUS, e comunicação *Peer-to-peer* entre equipamentos MODBUS escravos.

Estas são algumas características do MB-700:

- Integração com equipamentos inteligentes e *softwares* de outros fabricantes devido ao uso de padrões abertos como OPC *Server* e Modbus TCP/IP e RTU;

- Unidade totalmente integrada, tendo as seguintes funções: interface, *gateway*, *linking device*, *bridge* e concentrador de dados MODBUS;

- Como MODBUS *gateway*, o MB-700 pode trabalhar de duas formas: como *Gateway* TCP/IP para Serial ou Serial para TCP;

- Como concentrador de dados, o módulo pode concentrar os dados dos equipamentos escravos na Serial e fornecer os dados ao TCP/IP via OPC ou MODBUS TCP/IP;

- Como *Peer-to-peer* pode trocar dados MODBUS entre escravos conectados no TCP/IP, Serial ou ambos os Meios;

- Possui arquitetura limpa baseada em tecnologia de componentes; e

- Redundância total e isolação a falhas para segurança e operação ininterrupta.

ÍNDICE

INTRODUÇÃO ..... ***III***

GLOSSÁRIO ..... ***IX***

REFERÊNCIAS ..... ***XIII***

Capítulo 1 – VISÃO GERAL ***1.1***
Características Principais **1.2**
Integração do Sistema **1.3**
Módulo Fonte – PS-AC-R 1.3
Módulo Processador – MB-700 ..... 1.3
Protocolos Abertos 1.3
Configuração 1.3
Supervisão 1.3

Capítulo 2 - INSTALAÇÃO ***2.1***
Fixando os Racks e os Módulos **2.1**
Encaixe do Módulo ao Rack **2.2**
Encaixe do Rack ao Trilho DIN **2.3**
Encaixe entre os Racks **2.3**
Dicas para a Montagem 2.3
Instalando o *Hardware* **2.3**
Utilizando o Relé de Falha 2.4
Instalando o *System302* **2.4**
Obtendo a Licença para o DFI OLE *Server* **2.5**
Conectando o MB-700 à Sub-Rede **2.5**

Capítulo 3 - CONFIGURAÇÃO ***3.1***
Atualizando o *Firmware* **3.1**
Configurando o MB-700 via *Software* **3.6**
Criando uma Nova Planta 3.7

Capítulo 4 – BLOCOS FUNCIONAIS DO MB-700 ..... ***4.1***
Bloco CCCF- *Concentrate Configuration* **4.1**
Visão Geral 4.1
Descrição 4.1
Direção e Sentido do Fluxo de Dados 4.1
Endereços MODBUS 4.2
Configuração da Mídia Serial 4.2
Configuração da Mídia TCP 4.3
Ciclo de *Scan* 4.4
Parâmetros 4.5
Bloco CCSM - *Concentrate Supervision Master* **4.7**
Visão Geral 4.7
Descrição 4.7
A - Configurando os Pontos a serem Supervisionados 4.7
B - Supervisão dos Dados 4.8
C - Status de Supervisão 4.9
Parâmetros 4.9
Bloco CCCM – *Concentrate Control Master* **4.11**
Visão Geral 4.11
Esquemático 4.11
Descrição 4.11
A - Modo de Funcionamento da Comunicação “*Peer-to-peer*” em Dispositivo MODBUS 4.11
B - Endereçamento 4.13
C - Monitoração dos Dados 4.14
D - Status da Supervisão 4.15
Parâmetros 4.16

Bloco CCDL – *Concentrate Data Logger* **4.18**
Visão Geral 4.18
Descrição 4.18
Parâmetros 4.19

Capítulo 5 – ADICIONANDO BLOCOS AO MB-700 ..... ***5.1***
MB-700 como Mestre Serial MODBUS **5.1**
Cenário 5.1
Descrição 5.1
Configuração *Off Line* 5.2
Configuração *On Line* 5.12
MB-700 como Mestre TCP/IP **5.12**
Descrição 5.12
1 - MB-700 trabalhando como *Bypass* (Serial para TCP) 5.12
Configuração 5.13
2 - MB-700 trabalhando como *Peer-to-peer* 5.15
MB-700 como Escravo Modbus TCP **5.16**
1 - MB-700 trabalhando como Concentradora 5.16
Usando o *Data Logger* **5.17**
Descrição 5.17
Configuração 5.17
Blocos *DATA LOGGER* em Cascata 5.19

Capítulo 6 - CENÁRIOS ***6.1***
Múltiplos Mestres TCP/IP Comunicando Diretamente com vários LC700s, inclusive via Rádio **6.1**
Múltiplos Mestres TCP/IP Comunicando Diretamente com vários LC700s e "*Peer-to-peer*" entre LC700s na mesma Rede Modbus RTU **6.2**
Múltiplos Mestres TCP/IP Comunicando Diretamente com vários LC700s e "*Peer-to-peer*" entre LC700s de diferentes Redes Modbus RTU **6.3**
Múltiplos Mestres TCP/IP Comunicando Diretamente com os LC700s, "*Peer-to-peer*" e Redundância de Rede **6.4**
MB-700 como Concentrador e Supervisão via DFI OPC *Server* **6.5**
MB-700 como Concentrator, Supervisão via DFI OPC *Server* e "*Peer-to-peer*" **6.6**
MB-700 como Concentrador, Supervisão via DFI OPC *Server*, "*Peer-to-peer*" e Redundância de Redes **6.7**
Múltiplos Mestres Modbus RTU acessando um LC700 por uma única Porta (P2) **6.8**

Capítulo 7 – REDUNDÂNCIA HOT STANDBY ***7.1***
Arquitetura de um Sistema Redundante Completo **7.1**
Redundância do Módulo MB-700 ..... **7.2**
Terminologia 7.2
Pré-requisitos do Sistema **7.2**
Configurando a Redundância de Rede **7.3**
Configurando a *Workstation* 7.3
Configurando o DFI OLE *Server* 7.3
Configurando Redundância *Hot Standby* **7.4**
Configurando o Sistema pela Primeira Vez 7.6
Trocando a Configuração 7.7
Substituição de um Módulo com Falha 7.7
Correção de uma Falha no Canal de Sincronismo 7.7
Atualização de *Firmware* sem Interrupção do Processo **7.7**
Adicionando Redundância a um Sistema Existente **7.8**
Conexão Física SYNC_CABLE **7.8**
Tabela de Parâmetros Adicionais **7.8**
Descrição do Significado dos Bits de RED_BAD_CONDITIONS_L / R **7.10**

Apêndice A – SOLUCIONANDO PROBLEMAS ***A.1***
Reset **A.1**
Factory Init **A.1**
Modo HOLD **A.1**
Quando Usar os Procedimentos de *Factory Init/Reset* **A.2**

Apêndice B - CABLAGEM B.1
Especificação do Cabo Ethernet B.1
Especificação do Cabo Serial B.1
Interface EIA-232 ..... B.1
Conectando MB-700 ao LC700 ..... B.4
Interface EIA-485 ..... B.6
Dimensões B.7

Apêndice C – ESPECIFICAÇÕES PARA O MB-700 ..... C.1
Especificações Técnicas C.1
Tabela de Configuração Mínima de Blocos para o MB-700 ..... C.2
Tabela de Tipos de Dados Disponíveis para o Parâmetro DATA_TYPE C.2
Conversão de Escala C.3
Estrutura de Dados para o MB-700 ..... C.4
Estrutura de Bloco – DS-64 ..... C.4
Valor & Status –Estrutura Float – DS-65 ..... C.4
Valor & Status – Estrutura Discreta – DS-66 ..... C.4
Modo de Estrutura – DS-69 ..... C.4
Estrutura de Alarme Discreto – DS-72 ..... C.5
Estrutura de Evento de Atualização – DS-73 ..... C.5
Estrutura de Endereço Escravo - DS-263 ..... C.5
Estrutura de Endereço - DS-264 ..... C.6
Estrutura de Endereço - DS-265 ..... C.6
Estrutura de Endereço - DS-266 ..... C.6
Estrutura de Endereço - DS-267 ..... C.6

Apêndice D – FSR – FORMULÁRIO PARA SOLICITAÇÃO DE REVISÃO D.1

GLOSSÁRIO

***Baudrate*:**
Taxa de transferência de dados, dada em bits por segundo.

***Bridge*:**
Uma bridge isola duas redes. Ela tem a função de apenas repassar para o outro lado os dados endereçados a ele. Atua na camada de enlace.

Blocos Funcionais:
Linguagem de configuração de equipamentos de automação industrial. Para cada bloco funcional associam-se algoritmos e parâmetros de configuração. A troca de informações entre os blocos ocorre através de links entre entradas e saídas de blocos funcionais.

***Bypass*:**
Quando uma mensagem Modbus é recebida pelo MB-700 e repassada a outra rede no qual o módulo está conectado.

CCCF:
Bloco que configura a comunicação entre MB-700 e dispositivos escravos.

CCCM:
Bloco que configura a comunicação "*Peer-to-peer*".

CCSM:
Bloco que configura o funcionamento da função concentradora de dados do MB-700.

CCDL:
Bloco que configura *data Logger*.

Concentrador de Dados:
O MB-700 é um concentrador de dados. Ele pode armazenar em sua memória as variáveis lidas de equipamentos Modbus escravos e disponibilizá-las ao sistema supervisório e/ou monitoração, evitando que estes sistemas precisem acessar escravos da rede diretamente.

Comunicação "*Peer-to-peer*":
Forma de comunicação que permite a troca de informações entre equipamentos de mesmo nível hierárquico. Nas aplicações utilizando o MB-700 "peer-to-peer", significa que há troca de informações entre equipamentos Modbus escravos.

Comunicação Serial:
Os dados estão dispostos na forma serial, isto é, em uma sucessão de pulsos, resultando em uma forma de onda que compõe o dado a ser transmitido. Cada bit é enviado por vez em um único canal ou caminho**.**

Configuração:
Estabelecer os parâmetros para o funcionamento de um dispositivo.

Controlador Lógico Programável:
Controladores lógicos programáveis são dispositivos microprocessados que possibilitam criar lógicas de controle para automação de processos.

Download:
Envio da configuração ou do *firmware* de/para um dispositivo.

EIA-232/EIA-485:
Padrões que estabelecem as normas para os cabos (conexão física) usados na comunicação serial.

Endereços IP:
O protocolo IP cuida do encaminhamento dos dados na rede. Cada elemento da rede possui um endereço IP que é utilizado na comunicação dos dados. Um endereço IP tem 4 bytes. Por exemplo: 196.198.100.001.

Endereço Modbus:
Endereço do dispositivo em uma rede Modbus.

Endereço da Variável Modbus:
Trata-se do endereço da variável Modbus lida ou escrita no bloco de E/S de um escravo.

E/S:
Referem-se às entradas e saídas físicas do controlador lógico, através do qual obtém-se e enviam-se informações do/para o processo, respectivamente.

Escravo Modbus:
Equipamento que suporta o protocolo Modbus, desempenhando o papel de escravo. Este tipo de equipamento apenas responde aos comandos enviados a ele, isto é, não tem a iniciativa de requisitar algo a outro equipamento da rede.

Ethernet:
É normatizada pelo padrão 802.3 do IEEE. Refere-se ao meio físico onde está instalada a rede local. A taxa de transferência de dados pode chegar a 10 Mbits/s.

***Fast*-Ethernet:**
Opera em 100 Mbits/s. É semelhante à rede ethernet, porém a distância entre o *Hub* e a estação de trabalho é limitada em 100 metros.

***Gateway*:**
São conversores de protocolo. Atuam na camada de transporte (camada 4) do modelo OSI.

HSE:
High Speed Ethernet. Meio físico do protocolo TCP/IP.

***Hub*:**
Concentrador. É um dispositivo que une linhas de comunicação em um local central, fornecendo uma conexão comum a todos os dispositivos da rede.
É um concentrador de conexões. Cada elemento de uma rede local é conectado a um *Hub*, isolando e possibilitando a detecção de erros.

***Local Area Network* (LAN):**
Trata-se de uma rede local instalada dentro de uma empresa ou instituição, onde as estações de trabalho trocam informações entre si através desta rede.

Meios de Transmissão:
Meio físico utilizado nas redes industriais por onde o dado é transmitido.

Mestre Modbus:
Equipamento que suporta o protocolo Modbus, desempenhando o papel de mestre. Este tipo de equipamento pode ter a iniciativa de enviar comandos aos equipamentos escravos Modbus.

Modelo OSI:
Modelo da ISO que padroniza as fases da transferência de dados em redes. Existem 7 níveis neste modelo:
Nível 7 - Aplicação: Programas
Nível 6 - Apresentação: Conversão de dados.
Nível 5 - Sessão: Estabelecimento da conexão.
Nível 4 - Transporte: Controla a transferência de dados.
Nível 3 - Rede: Encaminhamento de pacotes, contabilização e transferência de dados.
Nível 2 - Controle de Linha: Gerenciamento (detecção e correção) dos erros.
Nível 1 - Físico: Especifica a conexão física dos elementos de rede.

***Modem*:**
Modem é uma sigla para modulador e demulador. O dispositivo efetua a modulação FSK (*Frequency Shift Keying*), ASK (*Amplitude Shift Keying*) e PSK (*Phase Shift Keying*) utilizadas para envio dos dados digitais através de uma linha telefônica ou através da irradiação eletromagnética. O *Modem* também efetua o processo inverso, a demodulação.

Modbus:
Protocolo de comunicação para equipamentos de uso industrial.

Modbus RTU:
Padrão Modbus de comunicação que utiliza como meio físico EIA-232, EIA-485 ou via rádio. A principal característica deste padrão é suportar um único mestre.

Modbus TCP:
Protocolo Modbus que utiliza as camadas TCP/IP suportando multi-mestre.

Monitorar/Monitoração:
Visualizar e alterar as variáveis lidas.

Parâmetro:
Variável cujo valor pode ser atribuído pelo usuário ou sistema para o funcionamento, comunicação e armazenamento do MB-700.

Protocolos de Comunicação:
São os padrões que controlam a comunicação de dados. O protocolo é o *software* que controla e garante a comunicação, que formata a forma de envio dos dados, define o caminho, o envio e a recepção dos dados, e ainda verifica erros de transmissão.

Redes:
Conjunto de dispositivos conectados de modo que eles possam trocar informações entre si.

Roteador:
O roteador é responsável pelo encaminhamento dos pacotes de dados por uma rede externa até que eles atinjam o destino. Atua na camada de rede (camada 3) do modelo OSI.

Sistema Supervisório:
Sistema localizado na estação de trabalho, onde o usuário pode monitorar e alterar as variáveis de campo e controlar os processos.

***Switch*:**
Análogo à *bridge*, porém permite que várias sub-redes conversem entre si duas a duas.

TCP/IP:
Protocolo de comunicação para redes locais e redes externas de empresas. Possui 4 níveis:
1) Nível de Rede: Protocolos dos níveis 2 e 3 do modelo OSI.
2) Nível de Roteamento: O protocolo IP roteia os dados pelas redes, mas não garante que a comunicação tenha sido feita.
3) Nível de Serviço: Os protocolos TCP e UDP transmitem os dados roteados do nível anterior para o nível acima.
4) Nível de Aplicação: Equivale às camadas 5, 6 e 7 do modelo OSI. Neste nível incluem-se ftp, SMTP, Telnet, etc.

Transmissor:
Dispositivo que inclui a placa eletrônica e transdutor/sensor que envia o valor da varíavel de campo lida neste dispositivo.

Variável Modbus:
Variáveis de equipamentos Modbus escravos disponíveis via comunicação.

REFERÊNCIAS

- **Manual de Instruções dos Blocos Funcionais**
- **Manual Syscon**
- **LC700: Guia do Usuário**
- **LC700: Manual de Configuração**

VISÃO GERAL

Este Manual do Usuário apresenta instruções sobre como configurar e instalar o **MB-700**.

O **MB-700** é um módulo microprocessado. Ele possui portas Ethernet, EIA-232/EIA-485, além de uma porta de sincronismo que pode ser utilizada para redundância. O MB-700 utiliza uma fonte de alimentação PS-AC-R. O sistema MB-700 consiste de um módulo CPU e uma fonte de alimentação.

Hardware:
- ✓ *Backplane* R-700-4A – Rack com 4 slots;
- ✓ Módulo Processador MB-700 - Processador 1x10 Mbps, 1xSYNC, Portas Seriais EIA-232/EIA-485;
- ✓ Fonte de Alimentação PS-AC-R – Fonte de Alimentação AC com capacidade para redundância;
- ✓ Cabo padrão Ethernet DF54 – Cabo Par trançado (100 Base Tx).

Software
- ✓ DFI OLE *Server* (utilizado na fase de configuração do MB-700 e quando configurada a característica de concentrador de dados);
- ✓ System302;
- ✓ Servidor DHCP (opcional);
- ✓ Windows NT *WorkStation* (*Service Pack* 3 ou superior).

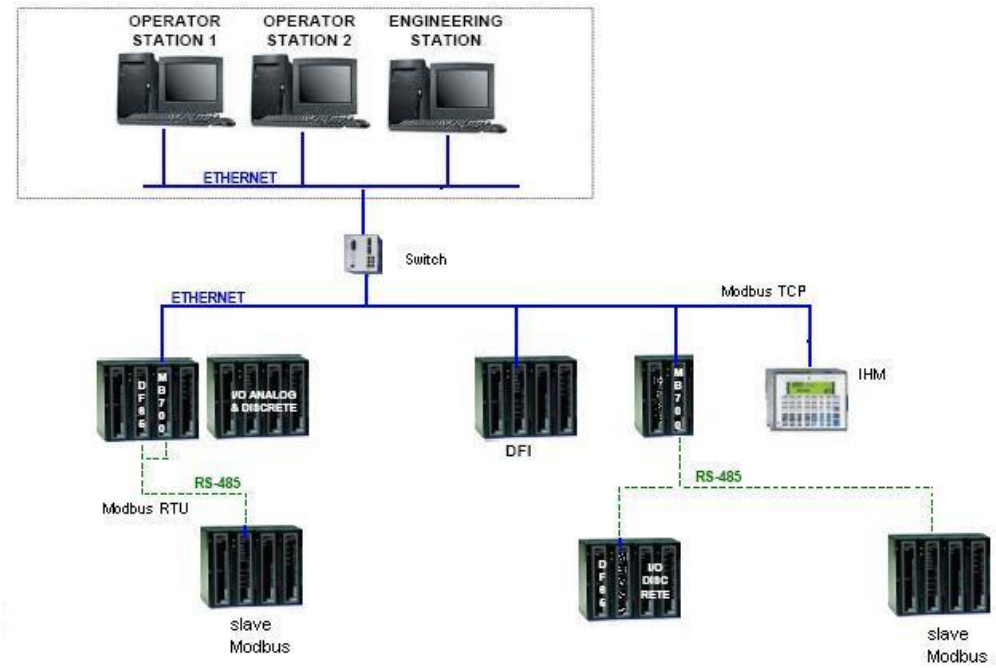


Figura 1.1- Visão Geral do MB-700

Características Principais

O MB-700 possui dupla função: é tanto uma ponte entre redes Ethernet e redes Modbus, como um dispositivo concentrador de dados.

A instalação do MB-700 pode ser realizada dentro de painéis localizados na sala de controle ou em caixas seladas no campo. É indicado para aplicações remotas evitando que o usuário se desloque até o local do aplicação para configurar um dispositivo escravo MODBUS. O usuário conecta a estação de trabalho ou interface homem máquina à rede e realiza a configuração dos controladores lógicos. O MB-700 faz da comunicação entre protocolos.

O MB-700 também pode ser configurado para atuar como um concentrador de dados. Ele coleta estes dados dos instrumentos de campo do processo conectados aos dispositivos escravos MODBUS. Desta forma, o sistema de monitoração ou supervisório pode acessar estes dados coletados diretamente na memória do MB-700, ao invés de acessá-los diretamente no dispositivo de campo.

O MB-700 é um módulo multifunção montado em um *backplane*, conectado em um trilho DIN onde também é conectada a fonte PS-AC-R. A modularidade é a chave da flexibilidade do sistema MB-700.

As conexões da fonte e do canal de Sincronismo (função de redundância) são feitas usando-se conectores *plug-in*, fazendo com que a remoção e inserção seja fácil e segura. Os conectores têm a vantagem de não poderem ser conectados de maneira errada, prevenindo assim uma aplicação de alta voltagem em um terminal de baixa voltagem. O módulo da fonte possui LEDs de diagnóstico que indicam operação normal e condições de falhas, o que faz com que a solução de problemas e diagnóstico seja simples, especialmente em um sistema com muitas unidades. É possível trocar o fusível (acessível externamente e localizado na entrada) sem a necessidade de se remover o módulo da fonte ou desconectar qualquer fio.

É importante observar que:
- Um *Backplane* é requerido para cada 4 módulos;
- Um *Flat Cable* é requerido para conexão de *Backplanes*; e
- O MB-700 utiliza como servidor OLE *Server* o servidor DFI OLE*Server*. A Licença para o DFI OLE*Server* está disponível em diferentes níveis, com diferentes capacidades para supervisão de blocos funcionais.

Integração do Sistema

Características avançadas de comunicação implementadas no **MB-700** garantem alta integração do sistema:

- Porta Ethernet;
- Modbus TCP/IP;
- Portas Serial EIA-232 / EIA-485;
- Modbus RTU; e
- Suporta até 31 escravos conectados no barramento EIA-485.

Módulo Fonte – PS-AC-R

Fonte para o sistema MB-700 e LC700. Ela fornece os 5V necessários para alimentação do barramento.

O módulo tem funções de diagnóstico e LEDs dedicados que indicam operação normal e condições de falha, o que faz com que os possíveis problemas sejam facilmente detectados, especialmente em sistemas com muitas unidades. Torna-se fácil verificar um módulo de fonte defeituoso em um painel com centenas de módulos.

É possível trocar o fusível (localizado na entrada com acesso externo) sem a necessidade de remover o módulo da fonte ou desconectar qualquer fio. A saída é protegida contra curto-circuito e não é danificada mesmo com curtos prolongados.

Módulo Processador – MB-700

Baseado em um processador 32-bit RISC e programa armazenado em *Flash*, este módulo manipula comunicação e tarefas de controle.

- 1 Porta Ethernet @ 10Mbps
- 1 Porta SYNC @ 31.25Kbps
- 1 Porta EIA-232/EIA-485 @ 9.6 Kbps – 115.2 Kbps
- CPU *Clock* @ 25MHz, 2MB NVRAM

Protocolos Abertos

Ethernet

Implementa os protocolos Smar Ethernet (SE) e MOBBUS baseados em TCP/IP e pode coexistir com outros protocolos Ethernet.

Modbus RTU: EIA-232 e EIA-485

Usando estas portas, o protocolo Modbus conecta dados da rede MODBUS com a sub rede local (Ethernet). As portas podem ser conectadas a uma rede de dispositivos (EIA-485) ou conectadas a controladores através de um modem. Suporta até 31 escravos conectados em cada porta serial, se usando EIA-485.

Configuração

O **MB-700** é completamente configurado por uma ferramenta de configuração (por exemplo SYSCON) através dos Blocos Funcionais disponíveis no padrão *Foundation Fieldbus*.

Supervisão

O **MB-700** é projetado com as tecnologias mais recentes. O uso destas tecnologias como OPC (OLE *for Process Control*) faz do **MB-700** uma interface flexível TCP/IP / MODBUS.

O servidor OPC permite que o **MB-700** seja conectado a qualquer pacote de supervisão. O único requisito é a existência de um cliente OPC para o pacote, assim o **MB-700** pode ser conectado às melhores interfaces de supervisão disponíveis.

Capítulo 2

INSTALAÇÃO

ATENÇÃO: A não observância de qualquer etapa descrita neste capítulo poderá ser a causa de um mal funcionamento do sistema.

Fixando os Racks e os Módulos

Observe as Figuras 2.1 e 2.2, e proceda conforme as instruções.

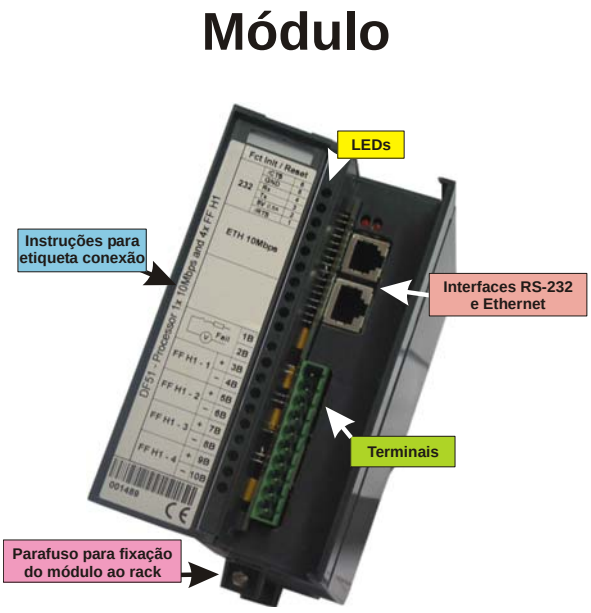


Figura 2.1 – Módulo

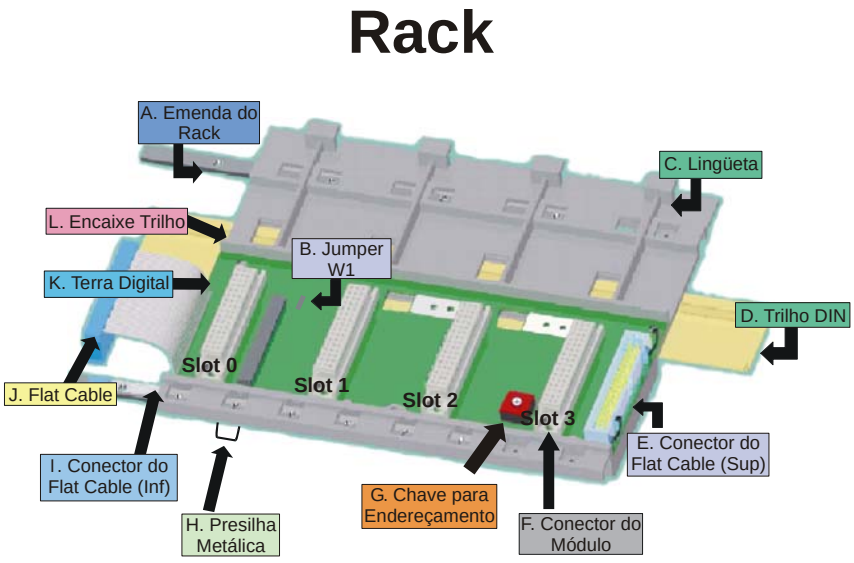


Figura 2.2 – Rack

A. Emenda do *Rack* - Ao montar mais de um *rack* em um mesmo trilho DIN, utilize a emenda do *rack* para fixar um *rack* ao outro. O uso da emenda dará mais firmeza ao conjunto e possibilitará a conexão do terra digital (K);

B. *Jumper* W1 – Quando conectado, permite que o rack seja alimentado pela fonte DC do *rack* precedente;

C. Lingüeta – Encaixe localizado na parte superior do *rack*;
D. Trilho DIN – Base para fixação do *rack*, deve estar firmemente fixado ao local de montagem do *rack*;
E. Conector do *Flat Cable* Superior – Permite que dois *racks* sejam interligados através do *flat cable* (J). Quando existir mais de um *rack* no mesmo trilho DIN, deve-se usar um *flat cable* (J) ligado ao conector do *flat cable* (I) e (E) para interligar os *racks*;
F. Conector do Módulo – Encaixe inferior do módulo ao *rack*;
G. Chave de Endereçamento – Quando houver mais de um *rack* no mesmo trilho DIN, as chaves de endereçamento permitem que seja atribuído um endereço distinto para cada *rack*;
H. Presilhas Metálicas - As presilhas metálicas, situadas na parte inferior do *rack*, permitem a fixação deste no trilho DIN. Devem ser puxadas antes de se encaixar o *rack* no trilho DIN e depois empurradas para a fixação das peças;
I. Conector do *Flat Cable* Inferior - Permite que dois *racks* sejam interligados através do *flat cable* (J). Quando existir mais de um *rack* no mesmo trilho DIN, deve-se usar um *flat cable* (j) ligado ao conector do *flat cable* (BUS) (I) e (E) para interligar os *racks*;
J. *Flat Cable* – Cabo usado para conexão do barramento de dados entre os *racks*;
K. Terra Digital – Quando houver mais de um *rack* em um mesmo trilho DIN, a conexão entre os terras digitais (K) deve ser reforçada através do encaixe metálico apropriado; e
L. Encaixe do Trilho – Suporte que faz o encaixe entre o *rack* e o trilho DIN (D).

Encaixe do Módulo ao Rack

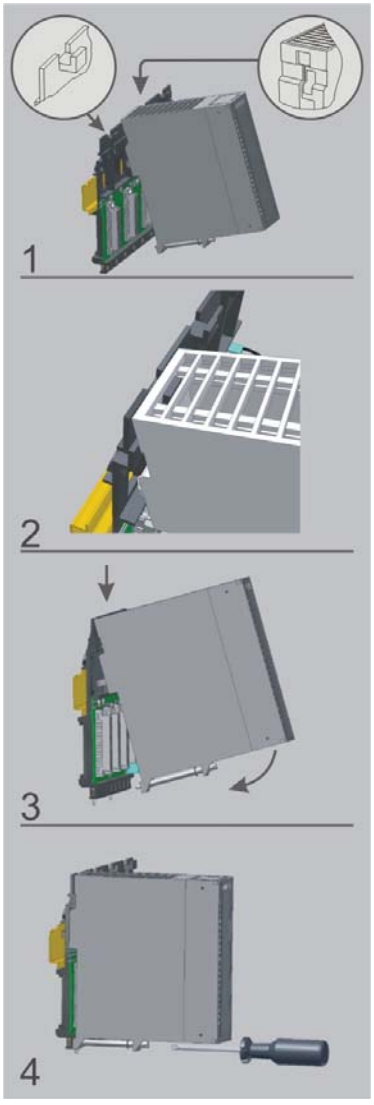
	Encaixando um módulo no rack: o Localize no rack a lingüeta que fica no topo do slot livre. o Encaixe o furo, localizado no topo e na parte traseira do módulo, na lingüeta.
	Detalhe do encaixe.
	Trave o módulo no conector (slot) do IMB pressionando-o contra o rack.
	Para finalizar, fixe o módulo no rack apertando com uma chave de fenda o parafuso de travamento localizado no fundo da caixa do módulo.

Tabela 2.1 – Encaixe do Módulo ao Rack

Encaixe do Rack ao Trilho DIN

1. Caso exista somente um rack, esta fixação pode ser feita como primeira etapa, mesmo antes de encaixar qualquer módulo ao rack;
2. Posicione (puxe) as presilhas metálicas (H) do rack;
3. Incline o rack e encaixe sua parte superior ao trilho DIN;
4. Dirija o rack à parte inferior do trilho até obter o contato das partes; e
5. Fixe o rack ao trilho, empurrando as presilhas metálicas (H).

Encaixe entre os Racks

1. Para o caso de existir mais de um rack no mesmo trilho, observe as conexões do *flat cable* (J) no conector superior do primeiro rack e no conector inferior do segundo rack, antes de encaixar o módulo do slot 3 do primeiro rack;
2. Fixe um rack ao outro através da emenda do rack (A). Passe o encaixe metálico de um rack ao outro e fixe através de parafusos;
3. Faça a conexão do terra digital (K), usando uma conexão metálica fixada por parafusos;
4. Observe a colocação do ***terminador*** para o último rack da montagem. O ***terminador*** deve ser plugado no conector do *flat cable* superior (E).

Dicas para a Montagem

Caso esteja trabalhando com mais de um rack, siga as seguintes instruções:

- Deixe para fazer a fixação no trilho DIN ao final da montagem;
- Mantenha o slot 3 do rack **livre** para poder interligar ao módulo seguinte pelo conector do *flat cable*;
- Verifique atentamente a configuração dos endereços (chave de endereçamento), bem como o *Jumper* W1 e o cabo do barramento;
- Lembre-se que para dar continuidade à alimentação DC do rack anterior é preciso que o *Jumper* W1 esteja conectado; e
- Faça a emenda dos racks e reforce o terra digital do conjunto.

Instalando o Hardware

Observe os detalhes da vista frontal dos módulos, como mostra a Figura 2.3 abaixo.

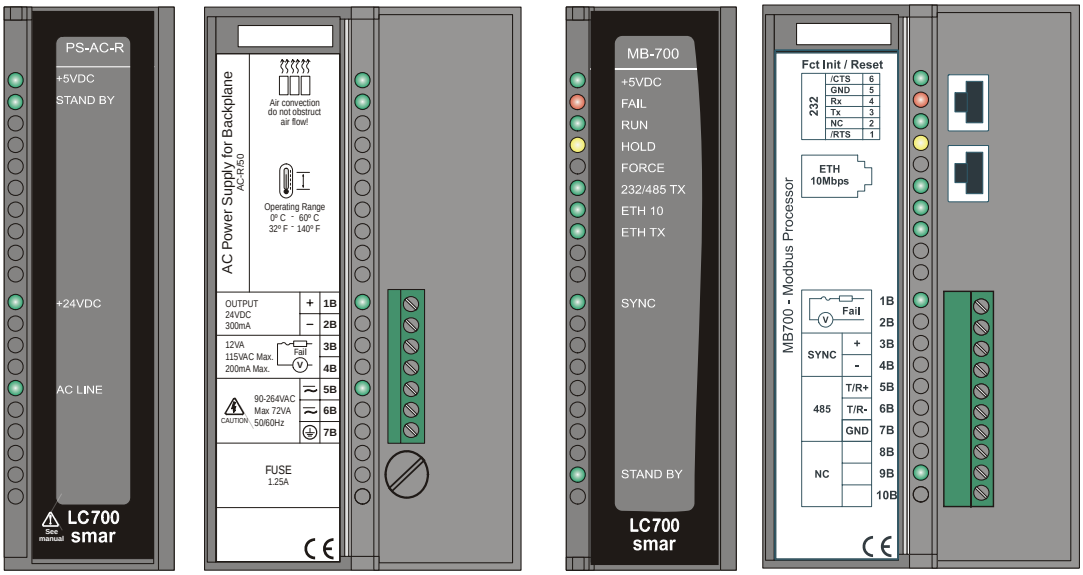


Figura 2.3 – Sistema Básico de Hardware do MB-700

Um cabo de par-trançado blindado é utilizado para conectar o **MB-700** ao *HUB*. O **MB-700** tem conectores RJ-45 simples. Não é requerida nenhuma ferramenta especial ou habilidades para a conexão. A instalação é simples e muito rápida.

O **MB-700** possui LEDs que indicam comunicação ativa ou falha. O módulo pode ser conectado ou desconectado sem ser desligado. Com o uso de *hub/switches*, pode-se desconectar dispositivos sem interromper o controle ou a comunicação de outros nós.

Há dois tipos de cabos que viabilizam a conexão **MB-700**/*HUB* / *Switch* (cabo DF54) ou conexão direta **MB-700**/PC (cabo DF55). Ver o Apêndice B para maiores detalhes.

Para uma instalação básica, execute os seguintes passos:
1. Conecte os dois módulos (PS-AC-R, **MB-700**);
2. Conecte a tensão de alimentação na entrada do PS-AC-R;
3. Utilizando o cabo Ethernet (DF54), ligue o **MB-700** ao *HUB* / *Switch*; e
4. O **MB-700** obterá automaticamente um endereço IP do ***DHCP Server***, mas se este servidor **não** estiver disponível, então inicialmente terá um IP fixo (este endereço IP fixo inicial poderá ser alterado através do FBTools – veja o Tópico "Conectando o **MB-700** à Sub-Rede").

Utilizando o Relé de Falha

Os terminais 1B e 2B disponíveis no **MB-700** podem ser utilizados em aplicações que exijam indicações de falha. Na verdade, estes terminais são os contatos de um Relé NC.

O Relé NC suporta:

0,5 A @ 125 VAC
0,25 A @ 250VAC
2 A @ 30VDC

Normalmente o **MB-700** força este relé a permanecer em aberto, mas se o processador entrar em qualquer condição de falha, o *hardware* fechará o relé. Esta indicação de falha pode ser utilizada em situações de redundância, no qual o Processador *Backup* lê estes contatos e notifica a falha. Outra possibilidade é utilizar estes contatos para acionar um alarme.

Instalando o System302

Execute a instalação dos aplicativos a partir do CD de instalação do *System302*. O acesso a todos aplicativos pode ser feito pelo atalho *System302 Browser* .

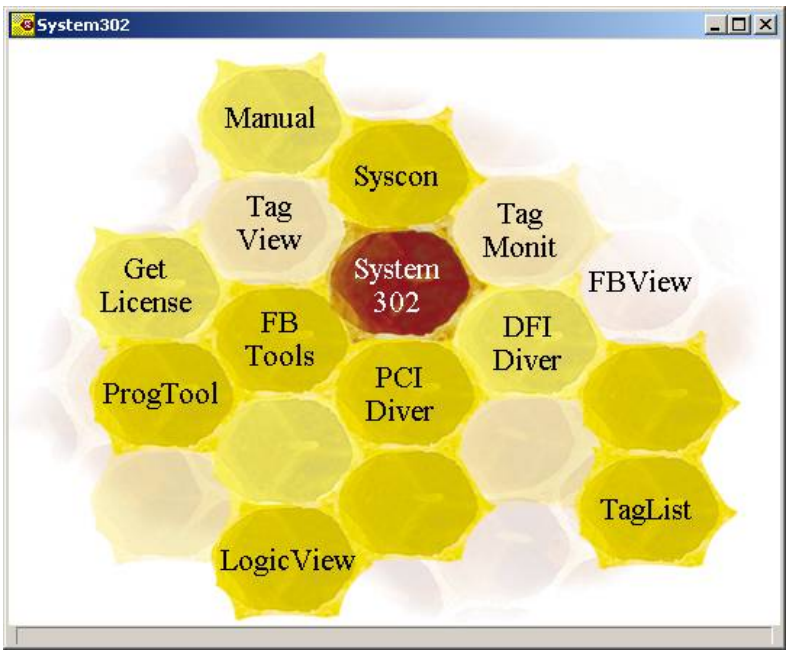


Figura 2.4 – Tela de Atalho do System302

Obtendo a Licença para o DFI OLE Server

Há duas maneiras de se obter a licença para o uso do DFI OLE*Server*. Existe a versão com proteção via *Hard Lock* (*HardKey*) e outra via *software* (*SoftKey*).

- A versão *HardKey* já vem pronta para uso, bastando conectar o dispositivo na porta paralela do computador; e

- Para uso da proteção via *software* é necessário obter a *License Key* entrando em contato com a SMAR. Para tanto use o aplicativo GetLicense.exe localizado no diretório de trabalho da Smar (geralmente "drive:\Program Files\Smar\OLEServers\GetLicense.exe"), ou diretamente pelo atalho "*Get License*" no *System302 Browser* (colméia).

A partir das informações geradas deste aplicativo, preencha o formulário FaxBack.txt e envie a Smar para obter a licença de uso do *Syscon*, PCI OLE*Server* e/ou DFI OLE*Server*. Veja a Figura 2.5.

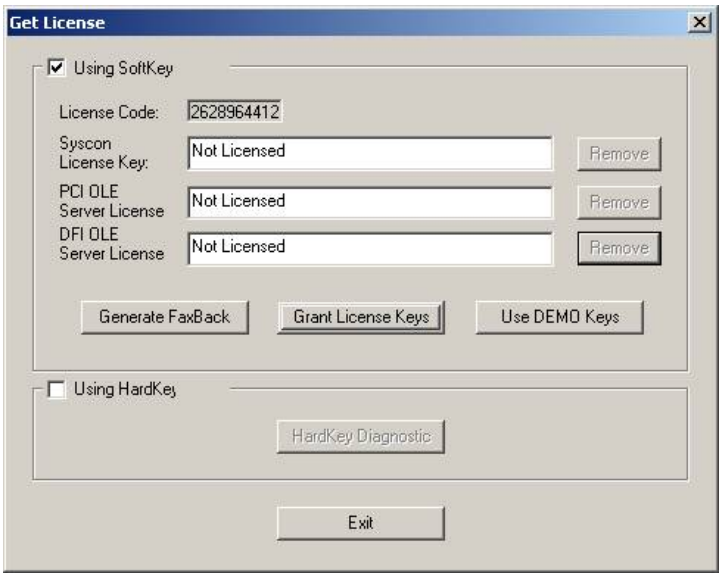


Figura 2.5 – Tela para Obtenção da License Key (Get License)

Ao obter o retorno da Smar com as *License Keys*, digite os códigos nos campos em branco (veja a figura anterior). Pressione a botão "*Grant License Keys*". Caso os códigos tenham sido aceitos, serão geradas mensagens confirmando o sucesso da operação. Assim o DFI OLE*Server* e/ou PCI OLE*Server* e/ou *Syscon* estarão prontos para serem usados.

Conectando o MB-700 à Sub-Rede

O ambiente para trabalhar com o **MB-700** envolve uma rede (Sub-Rede) onde deve-se ter os endereços IP para cada equipamento conectado.

A solução automática para atribuição destes endereços consiste em ter um servidor DHCP ***(Dynamic Host Configuration Protocol).*** Este Servidor DHCP fará a atribuição destes endereços IP dinamicamente para cada equipamento, evitando assim qualquer problema como a atribuição de endereços iguais para dois equipamentos distintos.

ATENÇÃO
Para conectar mais de um **MB-700**, os passos seguintes devem ser executados completamente para um **MB-700** de cada vez.

1. Conecte o cabo Ethernet DF54 entre o módulo **MB-700** e o *Switch* (ou *HUB*) da sub-rede a qual o **MB-700** fará parte.
OBS: Para conexão ponto a ponto (o módulo **MB-700** ligado diretamente ao microcomputador) utilize o cabo cruzado DF55.

2. Ligue o módulo **MB-700**. Assegure-se que o LED ETH10 e o LED RUN estejam acesos.

3. Mantenha pressionado firmemente o *Push-Button* (*Factory Init* / *Reset*) da esquerda e, em seguida, clique no *Push-Button* da direita três vezes, garantindo que o LED FORCE esteja piscando três vezes a cada segundo.
OBS.: Se o usuário perder a conta do número de vezes que o *Push-Bottom* da direita foi pressionado, basta verificar o número de vezes que o LED FORCE está piscando a cada segundo. Ele voltará a piscar uma vez por segundo depois do quarto toque (esta função é rotativa).

4. Libere o *Push-Bottom* da esquerda, o sistema executará o RESET e passará à execução do *firmware* com os valores padrões para o endereço IP e máscara de Sub-Rede.

5. Se a rede possuir um servidor DHCP (consulte o administrador da sua rede) o **MB-700** já estará conectado à sub-rede. Caso contrário, ele estará com o endereço IP 192.168.164.100 e os próximos passos deverão ser executados.

6. Se o usuário está neste passo é porque a rede não possui um servidor DHCP. Sendo assim, o endereço IP do microcomputador deverá, momentaneamente, ser alterado (para isso é aconselhável conhecimentos de administração de rede). Entre no Painel de Controle (*Control Panel - Windows*) e escolha a opção Conexões (*Network*).
OBS:. Caso a opção Conexões (*Network*) no Painel de Controle não possua o protocolo TCP/IP, use o Windows para proceder a instalação.

7. Escolha a opção Protocolo de Internet (*Internet Protocol*) e clique em Propriedades (*Properties*). A seguinte janela será exibida, conforme mostra a Figura 2.6..

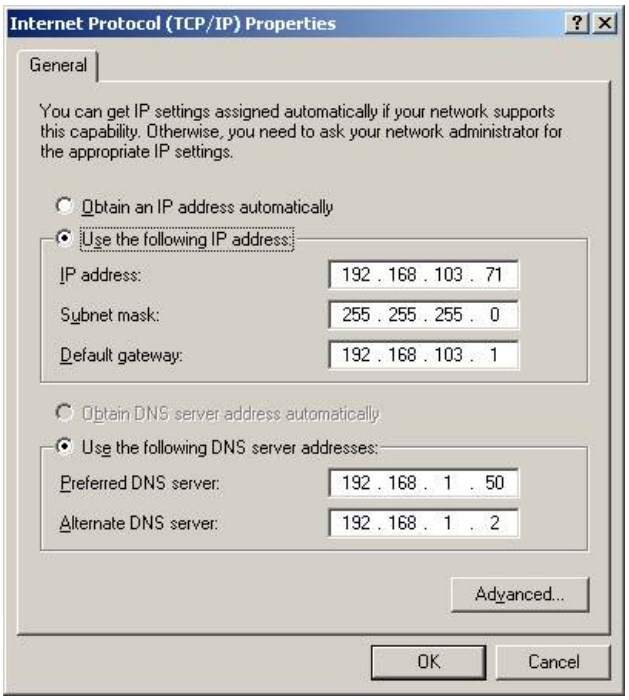


Figura 2.6 – Tela de Alteração do Endereço IP

8. Anote os valores originais de endereço IP e máscara de Sub-Rede de seu microcomputador para poder restaurá-los ao final da operação.

9. Altere o endereço IP e a máscara de Sub-Rede de seu computador, para que ele esteja na mesma Sub-Rede do **MB-700**. Preferencialmente os endereços IP que vão ser usados devem ser fornecidos pelo administrador da rede.
OBS.: Os valores deverão ser algo do tipo: Endereço IP (*IP Address*) 192.168.164.XXX e Máscara da Sub-Rede (*Subnet Mask*) 255.255.255.0. Mantenha o valor do *Gateway* padrão (*Default Gateway*).

ATENÇÃO
Não use o endereço 192.168.164.100 uma vez que este é o endereço padrão usado pelo **MB-700**. Certifique-se que o endereço escolhido não está em uso.

10. Clique no botão *OK*.

11. Execute o FBToolsWizard.exe, (localizado no diretório de trabalho da Smar, geralmente em "drive:\Program Files\Smar\FBTools\FBToolsWizard.exe", ou diretamente pelo atalho "*FBTools Wizard*".

12. Selecione o device **MB-700** e clique no botão *Next*. Veja a Figura 2.7 a seguir.

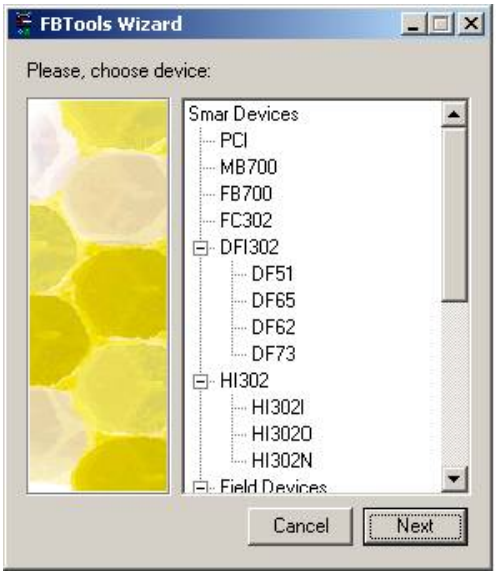


Figura 2.7 – FBTools Wizard – Tela para Escolha do Device

13. A tela abaixo, representada pela Figura 2.8, será mostrada. Clique no botão *Connect* para que os módulos que estejam disponíveis sejam reconhecidos.

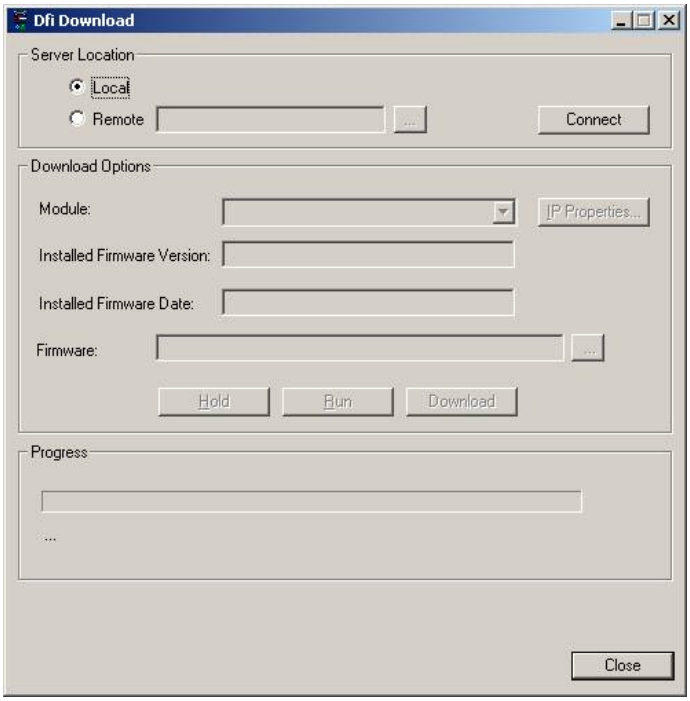


Figura 2.8 – Dfi Download - Tela para Conexão do Módulo (1)

14. Selecione o módulo **MB-700** desejado na opção *Module* usando como referência o número de série (verifique na etiqueta lateral, no próprio **MB-700**).

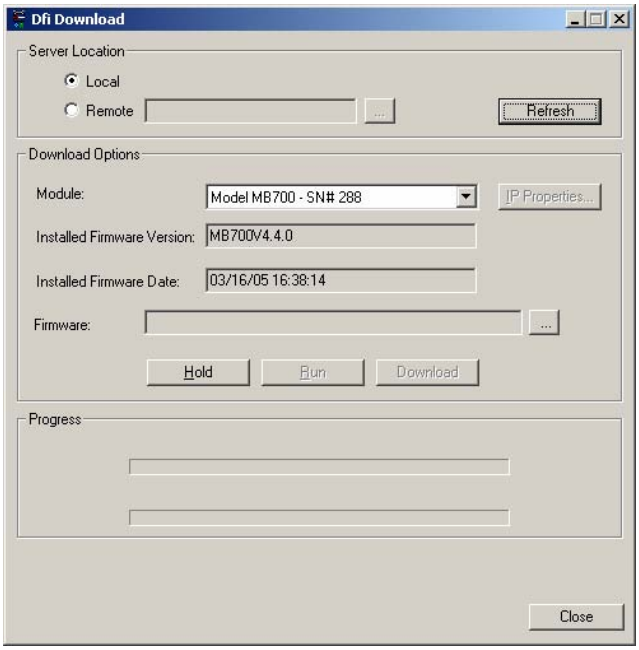


Figura 2.9 – Dfi Download - Tela para Escolha do Módulo

ATENÇÃO
A não observação deste passo pode implicar em conseqüências graves.

15. Para prosseguir será necessário interromper o *Firmware* que está sendo executado no módulo **MB-700**. Pressione o botão *Hold*.

16. Após o *Hold* do **MB-700**, o módulo não estará mais executando o *Firmware* e portanto irá parar toda a sua atividade na linha *Fieldbus*. Confirme a operação.



Figura 2.10 – Dfi Download – Tela de Confirmação de Entrada em Modo Hold

17. Certifique-se que o LED HOLD esteja aceso. Após interrromper a execução do *Firmware* no Módulo (*Hold*) a tela seguinte, representada pela Figura 2.11, será exibida novamente.

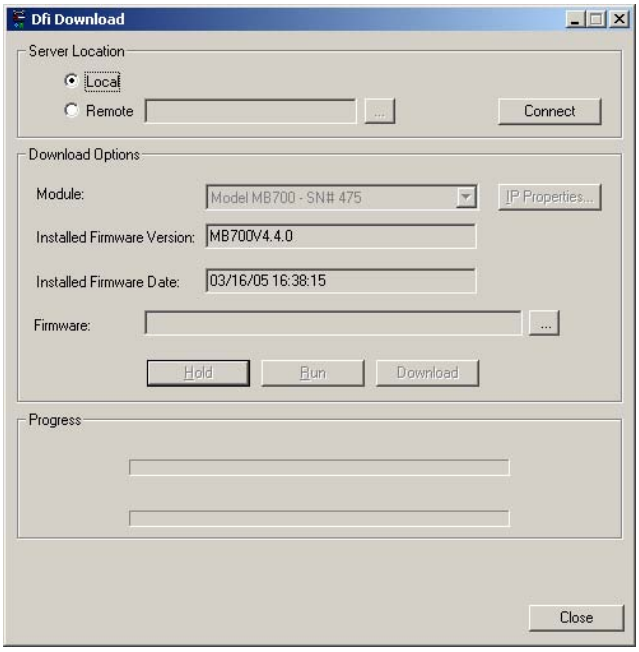


Figura 2.11 – Dfi Download – Tela para Conexão do Módulo (2)

18. Clique no botão *Connect* para voltar à conexão. Selecione o módulo **MB-700** desejado na opção *Module* usando como referência o número de série (verifique na etiqueta lateral, no próprio **MB-700**).

ATENÇÃO
A não observação deste passo pode implicar em consequências graves.

19. A opção padrão (*default*) é a atribuição através de um Servidor DHCP. Clique na opção *IP Properties***. Veja a Figura 2.12.**

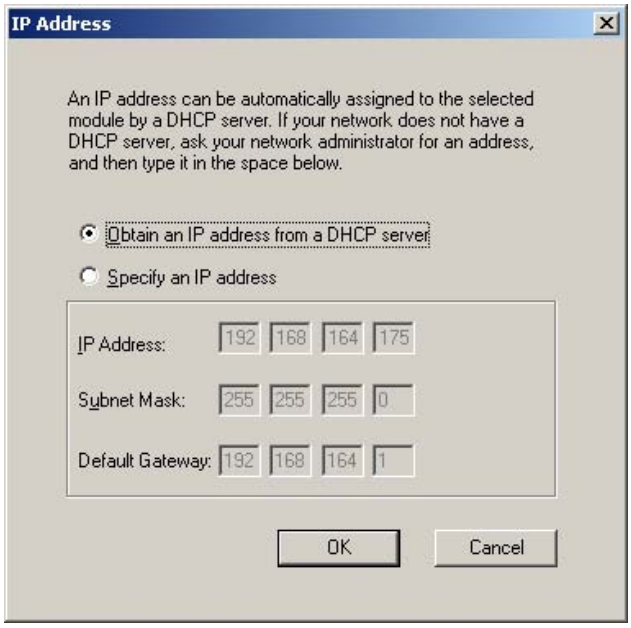


Figura 2.12 – Tela do Endereço IP - Atribuição do Endereço IP à partir de um DHCP Server

20. Clique na opção *Specify an IP Address*. Digite o Endereço IP e a Máscara da Sub-Rede que serão atribuídos ao **MB-700**.

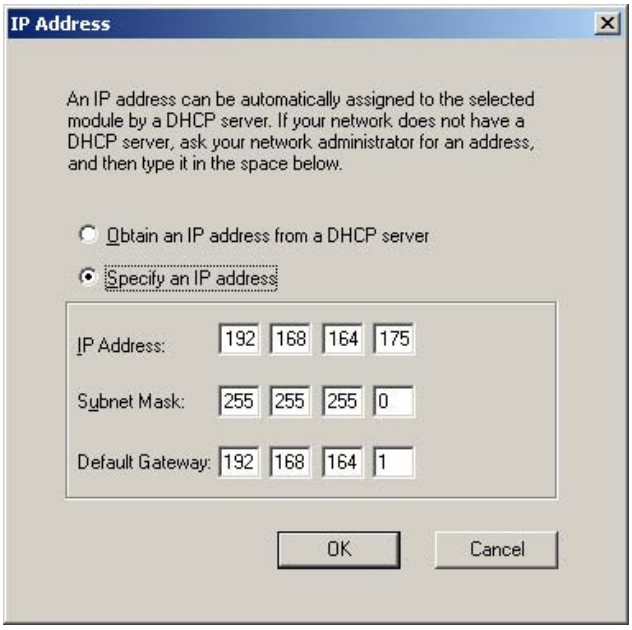


Figura 2.13 – Tela do Endereço IP - Especificação de um Endereço IP

ATENÇÃO
Não use o endereço 192.168.164.100 uma vez que este é o endereço padrão usado pelo **MB-700**. Assegure-se que o endereço escolhido não está em uso.

DICA
Anote os endereços de IP que serão atribuídos e relacione aos Números de Série de cada módulo **MB-700**, isso ajudará bastante na identificação e diagnóstico de possíveis falhas.

21. Clique *OK* para finalizar a operação.

22. Agora retorne à tela de propriedades TCP/IP do computador e restaure os valores originais de endereço IP e máscara de Sub-Rede.

23. Após atribuir o novo endereço de IP, o processo retornará à tela *DfiDownload* (veja a Figura 2.14).

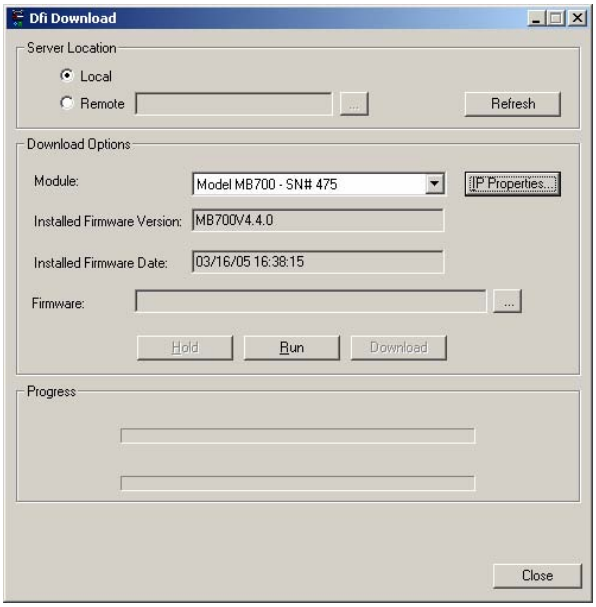


Figura 2.14 – Dfi Download – Tela para Conexão do Módulo (3)

24. Clique no botão *Run* para colocar o *Firmware* novamente em execução no **MB-700**.

25. Clique em *Close* na tela *DfiDownload* para encerrar a operação de Atribuição de IP.

26. No prompt do DOS, digite C:\>arp –d 192.168.164.100 <enter> (ver observação abaixo).

OBSERVAÇÃO
Caso tenha que configurar mais de um **MB-700,** execute o seguinte comando para **limpar a tabela ARP**, antes de passar para a configuração do próximo **MB-700**. C:\>arp –d 192.168.164.100 <enter>

27. Fim do procedimento de Conexão do **MB-700** na sua Sub-Rede para este módulo, para outros módulos repita este procedimento.

Capítulo 3

CONFIGURAÇÃO

Atualizando o Firmware

1. Conecte o cabo Ethernet DF54 do módulo **MB-700** ao *Switch* (ou HUB) da sub-rede a qual o **MB-700** fará parte.
OBS.: Para conexão ponto a ponto (módulo **MB-700** ligado diretamente ao microcomputador) utilize o cabo cruzado DF55.

2. Ligue o módulo **MB-700**. Assegure-se que o LED ETH10 e o LED RUN estejam acesos.

3. Mantenha pressionado firmemente o *Push-Button* (*Factory Init / Reset*) da esquerda e, em seguida, clique no *Push-Button* da direita três vezes, garantindo que o LED FORCE esteja piscando 3 vezes a cada segundo.
OBS.: Se o usuário perder a conta do número de vezes que o *Push-Button* da direita foi pressionado, basta verificar o número de vezes que o LED FORCE está piscando a cada segundo. Ele voltará a piscar uma vez por segundo depois do quarto toque (esta função é rotativa).

4. Libere o *Push-Button* da esquerda, o sistema executará o RESET e passará à execução do *firmware* com os valores padrões para o endereço IP e máscara de Sub-Rede.

5. Se a rede possuir um servidor DHCP (consulte o administrador da sua rede), o **MB-700** já estará conectado à sua sub-rede. Caso contrário ele estará com o endereço IP 192.168.164.100 e os próximos passos deverão ser executados.

6. Se o usuário está neste passo é porque a rede não possui um servidor DHCP. Sendo assim, o endereço IP do microcomputador deverá, momentaneamente, ser alterado (para isso é aconselhável conhecimentos de administração de rede). Entre no Painel de Controle (*Control Panel - Windows*) e escolha a opção Conexões (*Network*).
OBS.: Caso a opção Conexões (*Network*) em seu Painel de Controle não possua o protocolo TCP/IP, use o Windows para proceder a instalação.

7. Escolha a opção Protocolo de Internet (*Internet Protocol*) e clique em Propriedades (*Properties*). A seguinte janela será exibida, conforme mostra a Figura 3.1.

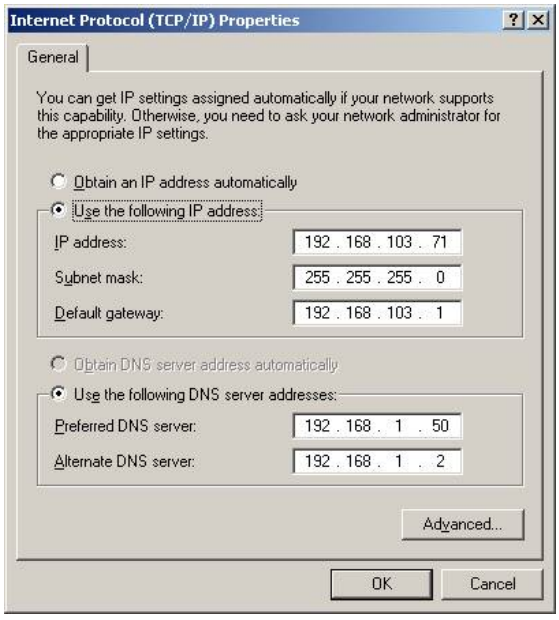


Figura 3.1 - Tela de Alteração do Endereço IP

8. Anote os valores originais de endereço IP e máscara de Sub-Rede de seu microcomputador para poder restaurá-los ao final da operação.

9. Altere o endereço IP e a máscara de Sub-Rede de seu computador, para que ele esteja na mesma Sub-Rede do **MB-700**. Preferencialmente os endereços IP que vão ser usados devem ser fornecidos pelo administrador da rede.
OBS.: Os valores deverão ser algo do tipo: Endereço IP (IP *Address*) 192.168.164.XXX e Máscara da Sub-Rede (*Subnet Mask*) 255.255.255.0. Mantenha o valor do *Gateway* padrão (*Default Gateway*).

ATENÇÃO
Não use o endereço 192.168.164.100 uma vez que este é o endereço padrão usado pelo **MB-700**. Assegure-se que o endereço escolhido não está em uso.

10. Clique no botão *OK*.

11. Execute o FBToolsWizard.exe, (localizado no diretório de trabalho da Smar, geralmente "drive:\Program Files\Smar\FBTools\FBToolsWizard.exe", ou diretamente pelo atalho "FBTools Wizard".

OBSERVAÇÃO
Para maiores informações de como utilizar o FBTools, consulte o Manual do Produto.

12. Selecione o device **MB-700** e pressione *Next*.

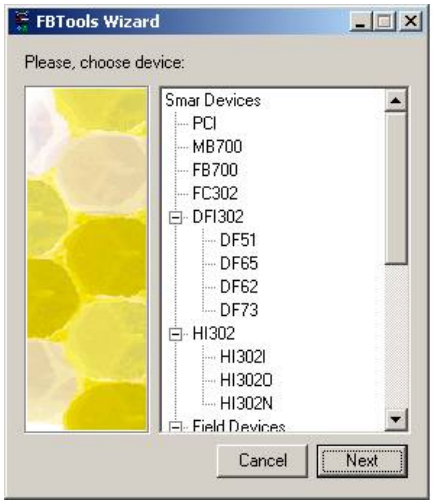


Figura 3.2 – FBTools Wizard – Tela para Escolha do Device

13. A janela abaixo será mostrada. Clique no botão *Connect* para que os módulos que estejam disponíveis sejam reconhecidos.

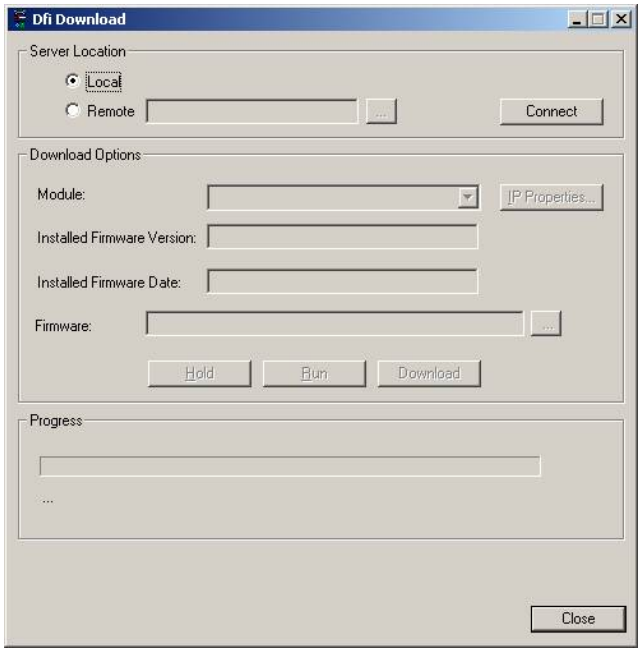


Figura 3.3 – Dfi Download - Tela para Conexão do Módulo (1)

14. Selecione o módulo **MB-700** desejado na opção *Module* usando como referência o número de série (verifique na etiqueta lateral, no próprio **MB-700**).

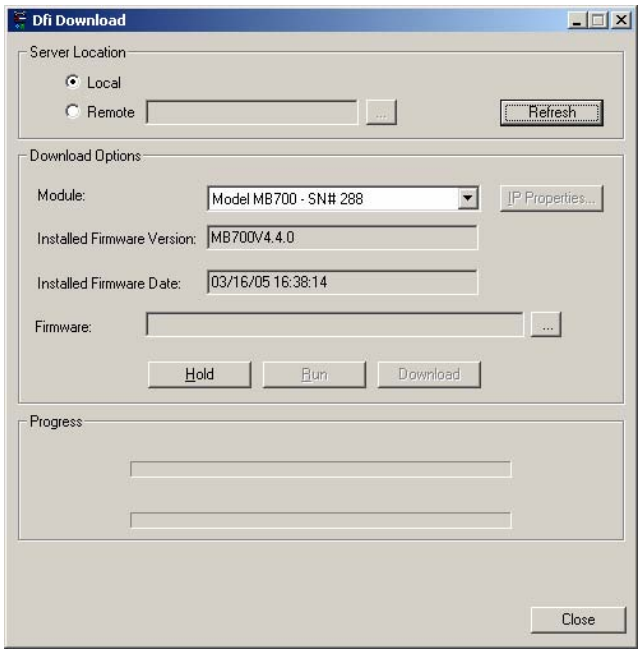


Figura 3.4 – Dfi Download - Tela para Escolha do Módulo

ATENÇÃO
A não observação deste passo pode implicar em consequências graves.

15. Para prosseguir será necessário interromper o *Firmware* que está sendo executado no módulo **MB-700**. Pressione o botão *Hold*.

16. Após o Hold do **MB-700**, o módulo não estará mais executando o *Firmware* e portanto irá parar toda a sua atividade na linha *Fieldbus*. Confirme a operação.



Figura 3.5 – Dfi Download – Tela para Interromper as Atividades Fieldbus

17. Certifique-se que o Led HOLD esteja aceso. Após interrromper a execução do *Firmware* no Módulo (*Hold*) a tela seguinte será exibida novamente (Ver a Figura 3.6).

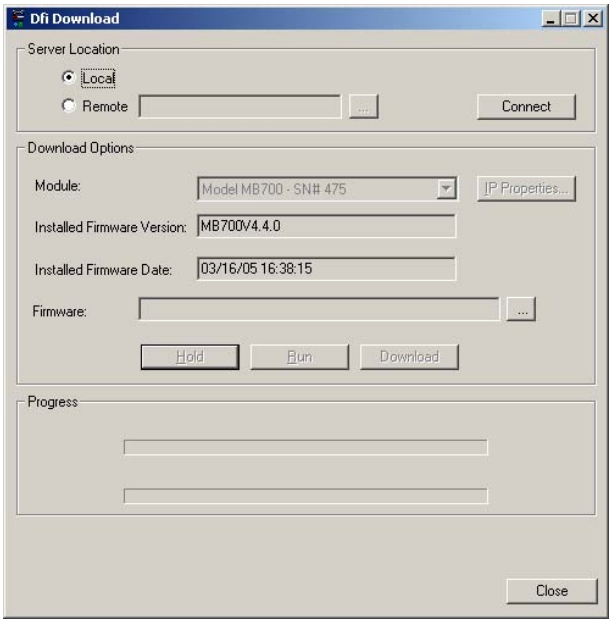


Figura 3.6 – Dfi Download – Tela para Conexão do Módulo (2)

18. Clique sobre o botão . Escolha o *firmware* desejado para fazer o download. Após ter escolhido o *firmware*, a janela *Dfi Download* aparecerá.

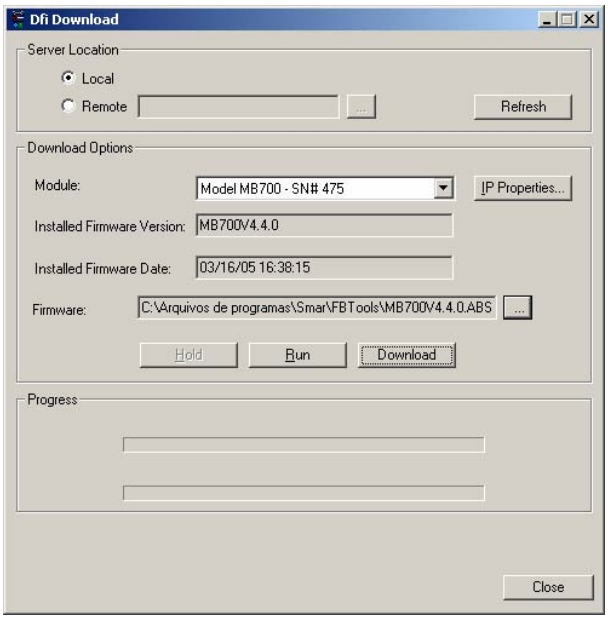


Figura 3.7 – Dfi Download – Tela para Escolha do Firmware para Download

19. Clique sobre o botão *Download*. A seguinte caixa de diálogo será exibida, conforme a Figura 3.8.



Figura 3.8 – Dfi Download – Tela de Confirmação do Download

20. Se a opção *Sim* for escolhida, o download do *firmware* será inicializado.

21. Durante o download, será apresentada a tela de progresso da operação. Veja a Figura 3.9.

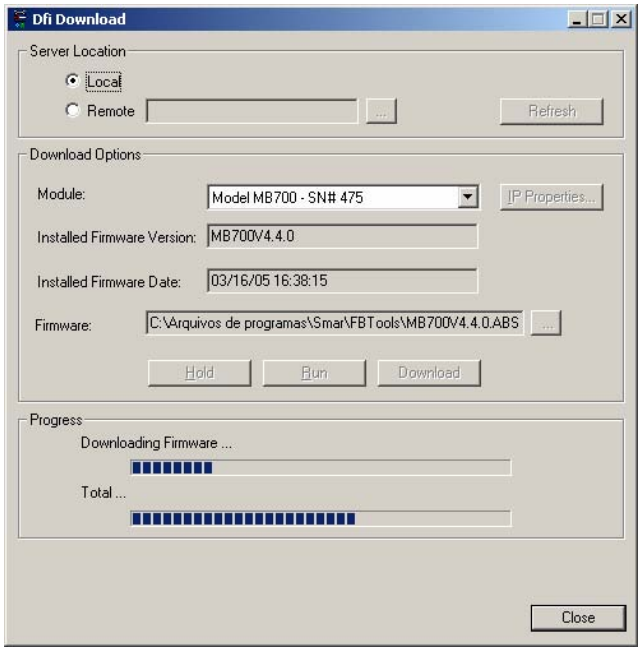


Figura 3.9 – Dfi Download – Tela de Progresso do Download

22. Ao final da operação será apresentada uma mensagem de *status* da operação de download. Neste momento o **MB-700** já estará no "*Modo Run*". Clique no botão *OK* (assegure-se que o LED RUN esteja aceso).



Figura 3.10 – Dfi Download – Tela de Finalização do Download

23. Para encerrar, clique na opção *Close, conforme a Figura 3.11.*

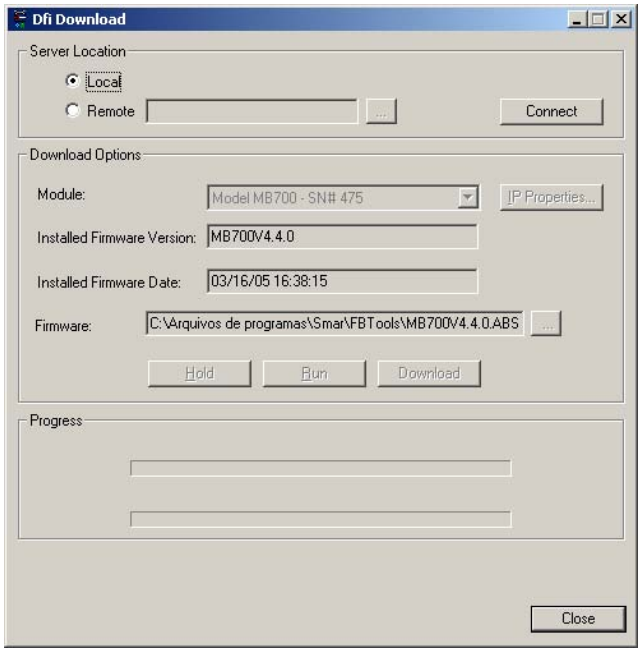


Figura 3.11 – Dfi Download – Tela para Encerrar a Operação

Configurando o MB-700 via Software

ATENÇÃO
Para que o **MB-700** possa ser configurado pelo *Syscon*, deve-se certificar que o procedimento "Conectando o **MB-700** à Sub-Rede" tenha sido concluído com sucesso.

O **MB-700** é completamente configurado através dos Blocos Funcionais *Fieldbus Foundation*.

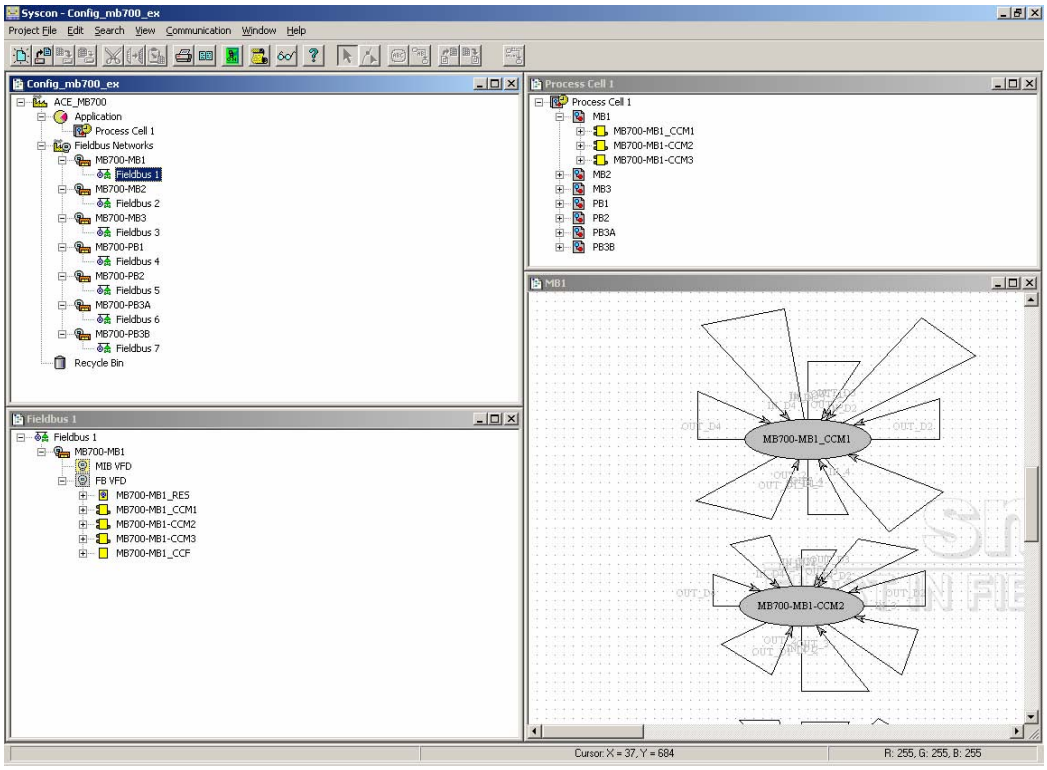


Figura 3.12 – Exemplo de Configuração - Syscon

Navegue pelos equipamentos da rede e estratégias de controle através do *software Syscon*.

O **MB-700** trabalha junto com o *Syscon*, *software* de configuração e manutenção, para se utilizar a característica de operação *plug-'n'-play* podendo então detectar, identificar e atribuir os endereços dos dispositivos conectados, removidos ou que tenham tido problemas. Uma vez conectado ao barramento Ethernet ou a uma estação de trabalho, o **MB-700** é detectado sendo então atribuído um endereço IP fixo ou variável dependendo do procedimento setado via *FBTools*, eliminando qualquer problema com micro-chaves (*Dip-switches*) ou duplicação de endereços.

Criando uma Nova Planta

Certifique-se de ter instalado o pacote do *System302* que contém o *Syscon*.

1. Uma vez instalado o *Syscon*, execute o aplicativo.



Figura 3.13 – Software Syscon

2. Na janela principal escolha *Project File → New*. Veja a Figura 3.14.

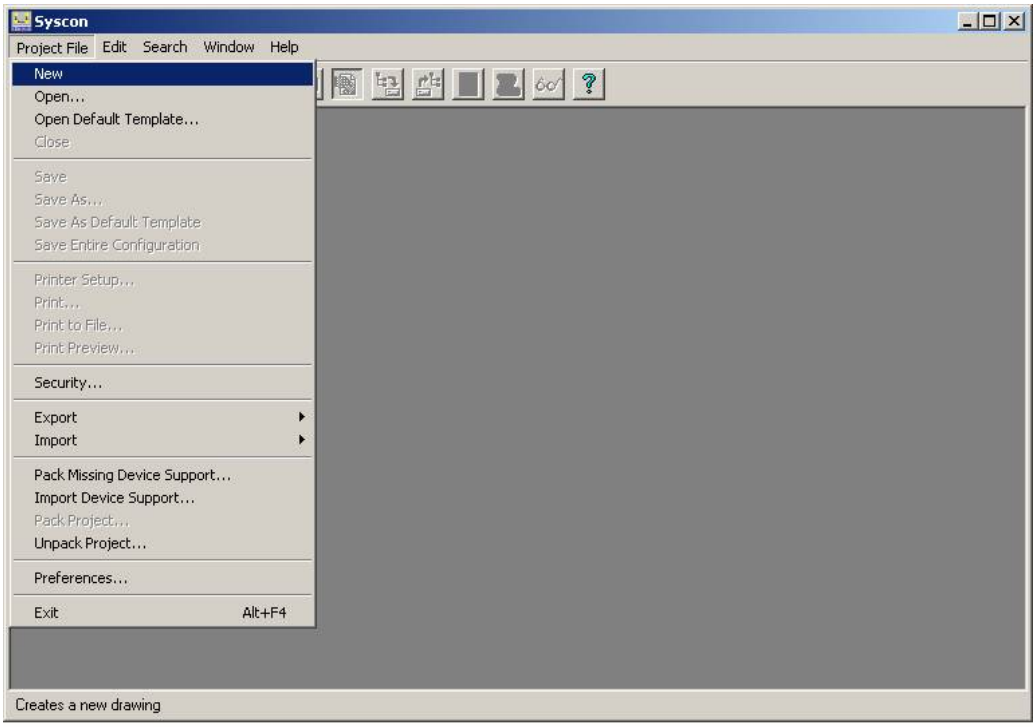


Figura 3.14 – Tela Principal - Software Syscon

3. Escolha *Projects* e atribua um nome a nova planta.

4. Primeiramente, deve-se configurar o aplicativo para usar o DFI *OLEServer*.

5. Na janela do projeto, clique sobre a opção *Fieldbus Networks* para que se possa configurar o *server*. Na barra de ferramentas principal, escolha *Communication → Settings*.

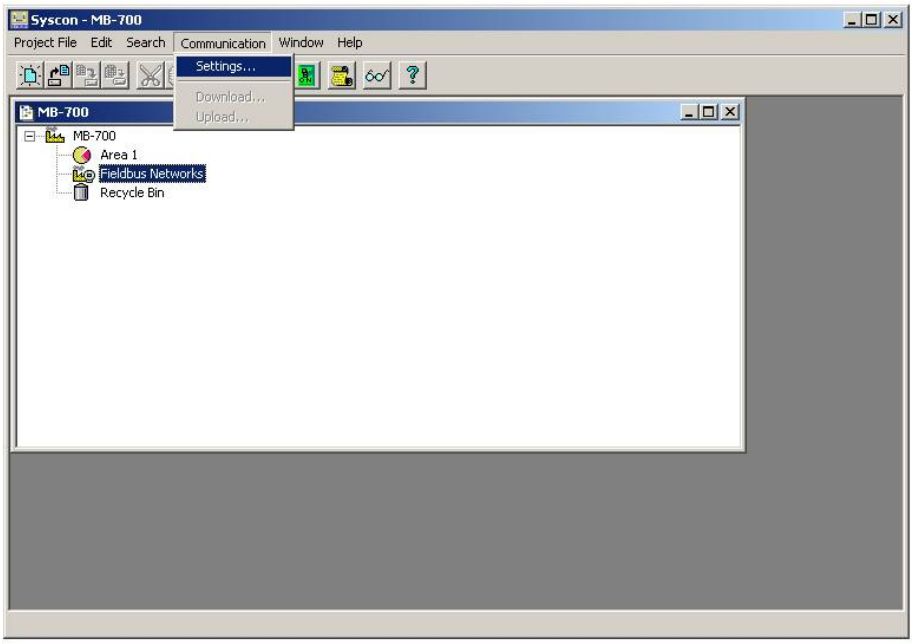


Figura 3.15 – Tela de Configuração do Server (1) - Syscon

6. Selecione o nome Smar.*DFIOLEServer*.0 no parâmetro *Server Id* e clique no botão *OK*.

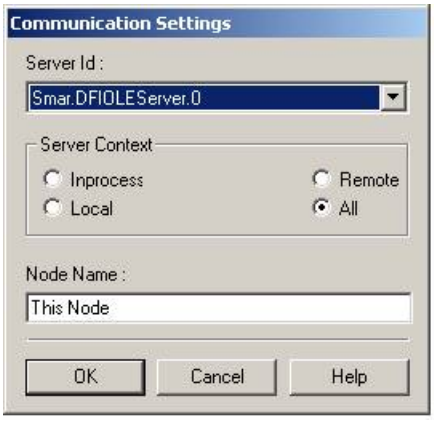


Figura 3.16 – Tela de Configuração do Server (2) - Syscon

7. Adicione todos os canais *Fieldbus* a serem utilizados.
8. No device **MB-700**, adicione os blocos CCCF (configuração) e *Resource*.
9. Faça a configuração *Off Line* do *Device*.
10. De acordo com o seu processo, adicione os blocos CCSM e CCCM. Veja o Capítulo 4 “Blocos Funcionais do **MB-700**” para mais informações. Os demais procedimentos são abordados com maiores detalhes no manual do *Syscon*.

Capítulo 4

BLOCOS FUNCIONAIS DO MB-700

Bloco CCCF - Concentrate Configuration

Visão Geral

Este bloco permite a configuração de vários parâmetros de comunicação do protocolo do MODBUS.

Descrição

Este bloco permite configurar os parâmetros de comunicação entre **MB-700** e equipamentos escravos Modbus através das portas Ethernet e serial (EIA-232/EIA-485). O usuário define a velocidade de transferência de dados das portas seriais, paridade, *timeout*, número de retransmissões e direção do *bypass*.

IMPORTANTE
O usuário deve configurar apenas 1 Bloco CCCF por dispositivo MB-700.

Direção e Sentido do Fluxo de Dados

O usuário deve também configurar o parâmetro BYPASS_DIRECTION, de modo estabelecer a direção do *bypass*. Existem duas opções:

- ✓ 0: TCP to serial (default); e
- ✓ 1: Serial to TCP.

A seguir são explicados os dois cenários de *bypass*:

- ***TCP to Serial:*** O **MB-700** trabalha como um *gateway* Modbus TCP para Modbus RTU. Quando um comando MODBUS chega ao **MB-700** via TCP/IP e o comando endereça um dispositivo diferente do que está configurado no parâmetro DEVICE_ADDRESS, então este comando é transmitido na serial (EIA-232/EIA-485). Havendo resposta do dispositivo endereçado, a resposta será transmitida na porta TCP/IP.

Aplicação: Mestre MODBUS TCP/IP comunicando com equipamento MODBUS RTU escravo, para supervisão, configuração; ou MB700 trabalhando como concentrador de dados ou *Peer-to-peer*.

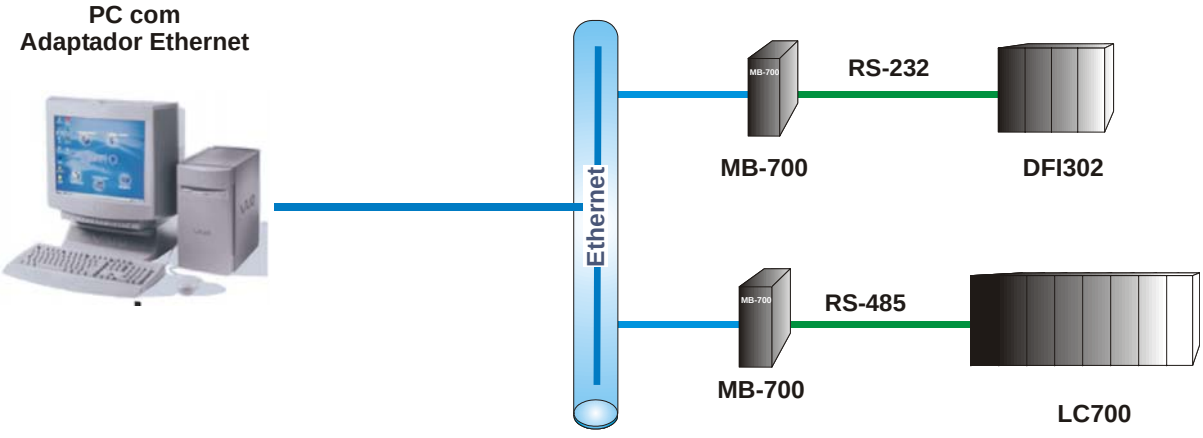


Figura 4.1 –Exemplo Configuração do MB-700 como Mestre Serial

- ***Serial to TCP:*** O MB-700 trabalha como um *gateway* Modbus RTU para Modbus TCP. Neste caso, o **MB-700** se comporta como um equipamento escravo na serial (EIA-232/EIA-485) e como mestre no TCP/IP. Um comando MODBUS RTU que chegar ao **MB-700** via serial (EIA-232/EIA-485) será transmitido no porta TCP/IP (endereçada nos parâmetros SLAVE_ADDRESSES e/ou IP_SLAVE_x). Havendo a resposta no TCP/IP, a mesma será transmitida na serial.

Aplicação: Equipamento mestre MODBUS RTU comunicando com equipamento típico MODBUS TCP/IP escravo.

Trabalhando em conjunto com outros **MB-700**s, permite-se que vários equipamentos mestre MODBUS RTU comuniquem com um mesmo equipamento escravo MODBUS RTU por uma única porta. Isto é, transformar o protocolo MODBUS RTU em multi-mestre.

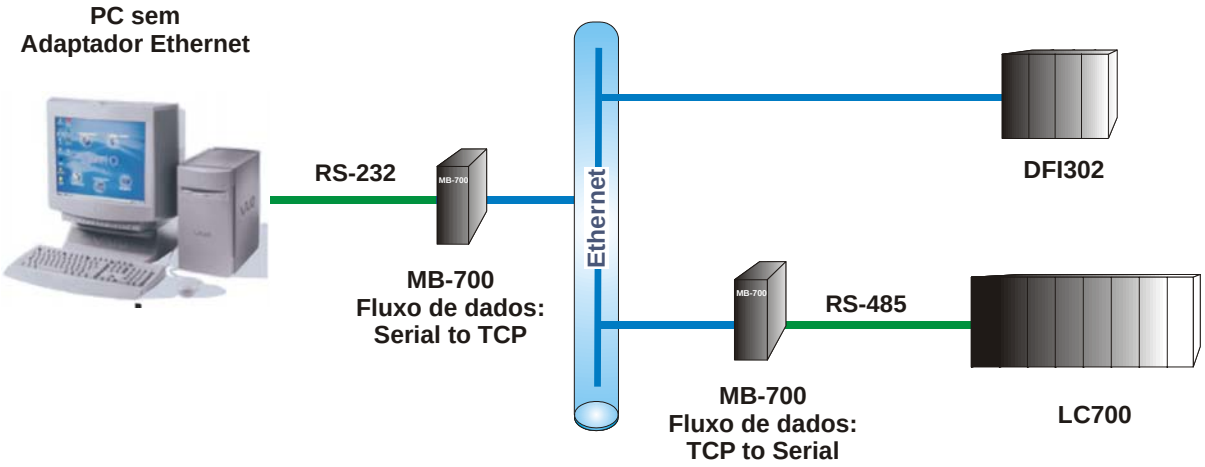


Figura 4.2 – Exemplo Configuração do MB-700 como Escravo Serial (Comunicação MB-700 com Estação de Trabalho) e Mestre Serial (Comunicação MB com LC700)

NOTA
Para este cenário, o **MB-700** não suporta os blocos CCSM e CCCM.

Endereços MODBUS

O usuário deve atribuir um endereço MODBUS ao **MB-700**. Porém deve se certificar de que este endereço não seja o mesmo de outro dispositivo das redes Modbus a qual está conectado na ethernet e serial. Neste caso o usuário deve configurar o parâmetro DEVICE_ADDRESS. O valor default deste parâmetro é 247.

Configuração da Mídia Serial

Em aplicações, onde utiliza-se a porta serial 232/485 do **MB-700,** é necessário configurar os seguintes parâmetros:

1 - Velocidade de Transferência de Dados das Portas Seriais

É possível selecionar a taxa de transferência de dados das portas seriais. Esta pode ser selecionada através do parâmetro BAUD_RATE. Ele permite a seleção entre as seguintes taxas de transferência:

- ✓ 0: 100 bps
- ✓ 1: 300 bps
- ✓ 2: 600 bps
- ✓ 3: 1200 bps
- ✓ 4: 2400 bps
- ✓ 5: 4800 bps
- ✓ 6: 9600 bps(*default*)
- ✓ 7: 19200 bps
- ✓ 8: 38400 bps
- ✓ 9: 57600 bps
- ✓ 10: 115200 bps

2 - Paridade

O parâmetro PARITY define o tipo de paridade para as portas seriais, que podem ser:

- ✓ 0: Sem paridade
- ✓ 1: Paridade Par (*default*)
- ✓ 2: Paridade Ìmpar

3 - *Timeout* e Número de Retransmissões

Timeout: tempo de espera pela resposta de um escravo, após uma mensagem ter sido enviada através de uma das portas seriais (P1 e P2) ou ethernet. O parâmetro TIME_OUT permite que o usuário escolha entre valores de 0 até 65535 milisegundos. O valor *default* é 1000. Este parâmetro tem ligação direta com o parâmetro NUMBER_RETRANSMISSIONS .

Número de Retransmissões: é o número de vezes que o **MB-700** tentará novamente estabelecer comunicação com o dispositivo escravo MODBUS após não receber resposta. O tempo esperado por esta resposta é configurado pelo parâmetro TIME_OUT. O número de retransmissões é escolhido através do parâmetro NUMBER_RETRANSMISSIONS. O usuário pode selecionar um valor de 0 a 255 para este parâmetro. O valor *default* é 1.

Na função *bypass*, não se aplica a funcionalidade do parâmetro NUMBER_RETRANSMISSIONS.

4 - RTS / CTS

Para os equipamentos que necessitam dos sinais RTS e CTS, o usuário pode habilitar ou desabilitar esta opção. No parâmetro RTS_CTS_TIMEOUT, o usuário configura o tempo máximo de espera pelo CTS após o RTS ter sido enviado. Caso o parâmetro RTS_CTS_TIMEOUT for igual a zero, esta função estará desabilitada. O valor *default* é zero (desabilitado).

5 - Tempo de Espera entre Rx e Tx

Opcionalmente, pode-se inserir um tempo de espera entre o final de uma recepção (Rx) e o início da próxima transmissão serial (Tx). O parâmetro TIME_DELAY determina o tempo em millisegundos do tempo de espera. Este tempo de espera é utilizado quando o equipamento (mestre ou escravo) apresenta lentidão no processamento da serial.

No caso do **MB-700** trabalhar como Mestre Serial, o atraso será entre uma resposta do escravo e a próxima pergunta do **MB-700**.

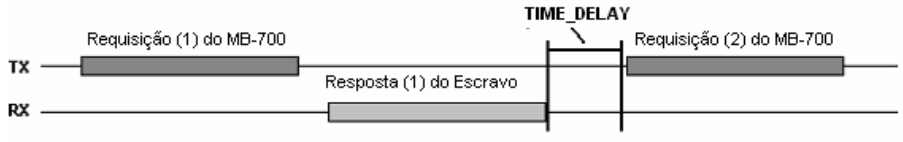


Figura 4.3 – Tempo de Espera entre Rx e Tx (MB-700 como Mestre Serial)

No caso do **MB-700** trabalhar como Escravo Serial, o atraso será entre uma pergunta do Mestre e a resposta do **MB-700** para o Mestre.

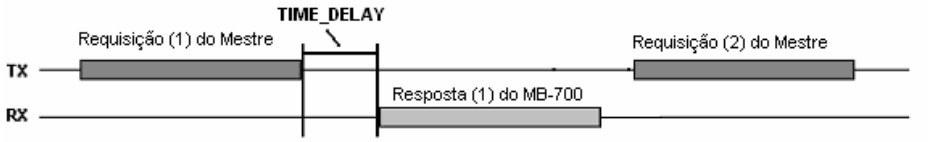


Figura 4.4 – Tempo de Espera entre Rx e Tx (MB-700 como Escravo Serial)

Configuração da Mídia TCP

Nas aplicações em que o **MB-700** é utilizado como Mestre Modbus TCP, os seguintes parâmetros devem ser configurados:

- No caso de ter um escravo TCP com um único endereço Modbus, utiliza-se o parâmetro SLAVE_ADDRESSES, onde é configurado o endereço IP do Escravo (IP_SLAVE_1 até IP_SLAVE_6) e o respectivo endereço Modbus do Escravo (MODBUS_ADDRESS_SLAVE_1 até MODBUS_ADDRESS_SLAVE_ 6);
- No caso de ter vários escravos Modbus para o mesmo endereço IP, utiliza-se os parâmetros IP_SLAVE_x com os respectivos endereços Modbus dos escravos DEVICE_IDS_IP_x (onde x indica os parâmetros de 7 ate 12).

Maiores detalhes sobre esta configuração são mostrados no Capítulo 5 “Adicionando Blocos ao **MB-700**”.

Ciclo de *Scan*

O clico de *scan* dos blocos de supervisão e controle podem ser ajustados através dos parâmetros SUPERVISION_OFF_DUTY e CONTROL_OFF_DUTY. O parâmetro SUPERVISION_OFF_DUTY indica o tempo do atraso em milisegundos (atraso entre leituras) durante o *scan* da supervisão (Bloco *Concentrate Supervision Master*). Da mesma forma, o parâmetro CONTROL_OFF_DUTY indica atraso durante o *scan* do controle (Bloco *Concentrate Control Master*).

OBSERVAÇÃO
Quando o **MB-700** está trabalhando como *bypass*, ou quando existe comandos de escritas de Supervisão, os comandos têm prioridade sobre o *Scan* de supervisão e controle e não segue os tempos de OFF_DUTY.

Parâmetros

Idx	*Parâmetro*	*Tipo de Dado (Tamanho)*	*Faixa Válida/ Opções*	*Valor Default*	*Unid*	*Store / Mode*	*Descrição*
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	Veja Parâmetro de Modo.
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	REDUNDANCY_ROLE	Unsigned8	Não utilizado	Main	E	S	Especifica o MB-700 preferencialmente ativo, quando há redundância.
8	REDUNDANCY_STATE	Unsigned8	Não utilizado	Standby	E	D/RO	Mostra qual MB-700 está ativo e qual MB-700 está no modo Standby.
9	BAD_COMM	Boolean	Não utilizado	TRUE	E	D / RO	Indica se qualquer dipositivo escravo Modbus acessado está comunicando.
10	BAD_COMM_STANDBY	Unsigned8	Não utilizado	TRUE	E	D / RO	Indica se Standby não é capaz de comunicar com o dispositivo escravo.
11	CHECK_COMM_STANDBY	Unsigned8	0 ~ 255	0	NA	S	Parâmetro configurado para Standby se for realizado o teste de comunicação entre os Equipamentos escravos no TCP. 0: Desabilita o teste. 1 – 255: Habilita o teste definindo o tempo de intervalo entre cada teste (s).
12	DEVICE_ADDRESS	Unsigned8	1-247	247	E	S	Define o endereço do dispositivo na rede Modbus quando este atua como escravo.
13	SLAVE_ADDRESSES	DS-263				S	O número IP e o endereço correspondente do dispositivo escravo quando acessado através de ethernet TCP/IP
14	BAUD_RATE	Unsigned8	0:110, 1:300, 2:600, 3:1200, 4:2400, 5:4800, 6:9600, 7:19200, 8:38400, 9:57600, 10:115200	9600	E	S	Define o baudrate das portas seriais.
15	STOP_BITS	Unsigned8	0:1, 1:2	1	E	S	Define o número de stop bits para as portas seriais.
16	PARITY	Unsigned8	0:None, 1:Even, 2:Odd.	Even	E	S	Define a paridade das portas seriais.
17	TIMEOUT	Unsigned16	0-65535	1000	ms	S	Tempo a se esperar por uma resposta do escravo após enviar um comando para as portas seriais.
18	RTS_CTS_TIMEOUT	Unsigned16	0-65535	0	ms	S	Especifica o tempo máximo de espera para CTS se tornar ativo após enviar RTS nas portas seriais.
19	NUMBER_RETRANSMISSIONS	Unsigned8	0-255	1		S	Número de retransmissões se o MB-700 não receber uma resposta do escravo.
20	SUPERVISION_OFF_DUTY	Unsigned16	0-65535	0	ms	S	Tempo entre comandos de supervisão. O Valor 0 indica que OFF_DUTY está desabilitado.
21	CONTROL_OFF_DUTY	Unsigned16	0-65535	0	ms	S	Tempo entre Comandos de Controle. O valor 0 indica que OFF_DUTY está desabilitado.

Idx	Parâmetro	Tipo de Dado (Tamanho)	Faixa Válida/ Opções	Valor Default	Unid	Store / Mode	Descrição
22	BYPASS_DIRECTION	Unsigned8	0: TCP to Serial 1: Serial to TCP	0	na	S	Define a direção dos dados. Se a direção é "Serial to TCP" o dispositivo tem apenas a funcionabilidade de Bypass.
23	ON_APPLY	Unsigned8	0:None, 1: Apply	None	E	S	Aplica as mudanças feitas nos blocos Modbus.
24	SCAN_TIME	FloatingPoint				D/RO	Tempo utilizado pela porta serial para fazer a varredura de todas as variáveis Modbus.
25	UPDATE_EVT	DS-73			Na	D	Este alarme é gerado por qualquer mudança nos dados estáticos.
26	BLOCK_ALM	DS-72			Na	D	O bloco de alarme é utilizado para toda configuração de hardware, falha na conexão ou problemas de sistema nos blocos. A causa do alerta é mostrada no campo subcode. O primeiro alarme a se tornar ativo mudará o status ativo no Status Attribute. Assim que o status Unreported é zerado pela ferramenta de anúncio de alerta, outro bloco pode ser informado sem apagar o status Active, se subcode mudou.
27	IP_SLAVE_7	VisibleString(32)		Spaces		S	Endereço IP do Escravo 7.
28	IP_SLAVE_8	VisibleString(32)		Spaces		S	Endereço IP do Escravo 8.
29	IP_SLAVE_9	VisibleString(32)		Spaces		S	Endereço IP do Escravo 9.
30	IP_SLAVE_10	VisibleString(32)		Spaces		S	Endereço IP do Escravo 10.
31	IP_SLAVE_11	VisibleString(32)		Spaces		S	Endereço IP do Escravo 11.
32	IP_SLAVE_12	VisibleString(32)		Spaces		S	Endereço IP do Escravo 12.
33	DEVICE_IDS_7	Array Unsigned8 [32]		0		S	Lista de IDs Modbus dos Equipamentos para o respectivo ID do escravo 7.
34	DEVICE_IDS_8	Array Unsigned8 [32]		0		S	Lista de IDs Modbus dos Equipamentos para o respectivo ID do escravo 8.
35	DEVICE_IDS_9	Array Unsigned8 [32]		0		S	Lista de IDs Modbus dos Equipamentos para o respectivo ID do escravo 9.
36	DEVICE_IDS_10	Array Unsigned8 [32]		0		S	Lista de IDs Modbus dos Equipamentos para o respectivo ID do escravo 10.
37	DEVICE_IDS_11	Array Unsigned8 [32]		0		S	Lista de IDs Modbus dos Equipamentos para o respectivo ID do escravo 11.
38	DEVICE_IDS_12	Array Unsigned8 [32]		0		S	Lista de IDs Modbus dos Equipamentos para o respectivo ID do escravo 12.
39	TIME_DELAY	Unsigned 16		0	millisec	S	Tempo de espera em millisegundos entre uma recepção (Rx) e a próxima transmissão (Tx) do MB-700.
40	TCP_SCAN_TIME	Float				D/RO	Tempo utilizado pela porta Ethernet para fazer a varredura de todas as variáveis Modbus.

Tabela 4.1 - Legenda: Coluna Store / Mode – Store: armazenamento de dados (D – Dinâmico; N – Não volátil; S – Estático) / Mode: mínimo modo necessário para o usuário conseguir modificar o parâmetro (OOS – Out of Service ou MAN - Manual); Se não tiver Mode, significa que não depende do modo para ser alterado. RO – Somente leitura; Coluna Unid – E: parâmetro Enumerado; Na – Parâmetro adimensional.

Bloco CCSM - Concentrate Supervision Master

Visão Geral

Este bloco fornece informação para monitorar um dispositivo escravo MODBUS conectado à porta serial do **MB-700** com função de supervisão. Esta funcionalidade é obtida fazendo o remapeamento das variáveis Modbus do dispositivo escravo em parâmetros deste bloco.

Descrição

Neste bloco, o usuário deve informar ao **MB-700** os endereços MODBUS dos dispositivos conectados à rede MODBUS.

NOTA
O padrão do protocolo Modbus especifica a seguinte divisão de range de endereço Modbus para as variáveis: ▪ 0001 a 9999: Saídas Digitais ▪ 10001 a 19999: Entradas Digitais ▪ 30001 a 39999: Entradas Analógicas ▪ 40001 a 49999: Saídas Analógicas

O Bloco CCSM permite que o usuário visualize e altere as variáveis dos endereços MODBUS configurados.

A - Configurando os Pontos a serem Supervisionados

MODO DE OPERAÇÃO

O parâmetro MODE_BLK configura o modo de operação do bloco. Os modos suportados pelo Bloco CCSM são “Auto” (Automático, ou seja, operação normal) e “OOS” (“*Out Of Service*”, fora de serviço). O MODE_BLOCK é composto por *Target*, *Actual*, *Permitted* e *Normal*. O modo *Target* é o modo de operação do bloco selecionado pelo usuário. *Actual* é o modo real de operação do bloco, pois apresenta o modo atual em que o bloco está funcionando.

Os parâmetros de configuração do bloco como: SLAVE_ADDRESS, B_ADDRESS, F_ADDRESS não poderão ser alterados se o bloco estiver funcionando em modo automático. Para alterar os parâmetros, o usuário deverá alterar o modo *Target* para OOS e, somente depois disso, alterar os parâmetros de configuração para em seguida alterar o modo de operação *Target* para “Auto”. Após alterar qualquer parâmetro, o usuário deverá novamente alterar o modo ON_APPLY (no bloco de configuração CCCF) para “*Apply*” para que essas mudanças sejam efetivadas. O bloco continuará em OOS (MODE_BLK.ACTUAL = OOS), enquanto o parâmetro ON_APPLY não for alterado para “*Apply*”.

NOTA
A coluna Store/Modo da tabela de parâmetros do bloco CCSM contém uma lista de parâmetros que precisam que o modo OOS seja selecionado antes que estes parâmetros sejam alterados. Para saber mais sobre o modo, consulte o manual dos Blocos Funcionais.

FORMA DE SUPERVISÃO

1 - Parâmetro SCAN_BEHAVIOR:

O usuário pode selecionar este parâmetro de dois modos:

1- Usando *Config View*, ou
2- Não usando *Config View*

Se o usuário selecionar o modo “1”, o **MB-700** utilizará uma forma otimizada de obter os dados dos dispositivos escravos. Isto aumentará a freqüência de atualização dos parâmetros destes blocos, que estão mapeados a variáveis em equipamentos escravos Modbus.

NOTA
O modo “Usando *Config View*” é suportado apenas por dispositivos da SMAR.

IMPORTANTE
Quando o parâmetro SCAN_BEHAVIOR= “Usando *Config View*”, deve-se atentar que os parâmetros do tipo Porcentagem (P_EU_ADDRESS_Ai/Bi.DATATYPE, por exemplo) suportarão somente os formatos dos tipos *Integer16* e *Unsigned16*.

ENDEREÇAMENTO

1 - Parâmetro SLAVE_ADRR:
Neste parâmetro, o usuário deverá informar o endereço do dispositivo escravo de sua rede MODBUS RTU.

2 - Parâmetro B_ADDRESSi:
Dados *booleanos*. O usuário deve digitar os endereços MODBUS das variáveis discretas que deseja monitorar. Até 96 pontos *booleanos* podem ser supervisionados.

3 - Parâmetro I_ADDRESSi:
Dados Inteiros. O usuário deve digitar os endereços MODBUS das variáveis inteiras que deseja monitorar. Este parâmetro permite monitorar dados inteiros de 1, 2 ou 4 *bytes*. Uma leitura de 1 ou 2 *bytes* possui apenas um endereço MODBUS correspondente. Se o formato de 4 *bytes* for selecionado, o usuário deverá apenas selecionar o primeiro endereço MODBUS. Até 8 variáveis inteiras podem ser supervisionadas.

4 - Parâmetros P_EU_ADDRESS_Ai/ P_EU_ADDRESS_Bi:
Dados em porcentagem. O usuário deve configurar os parâmetros mostrados abaixo. O Apêndice C mostra detalhes de conversão de escala. Até 56 variáveis em porcentagem podem ser supervisionadas.

- FROM_EU_100%
- FROM_EU_0%
- TO_EU_100%
- TO_EU_0%

- DATATYPE

Corresponde ao formato do dado no dispositivo escravo Modbus. Para mais detalhes sobre os formatos disponíveis, veja a tabela no Apêndice C.

- MODBUS_ADDRESS_OF_VALUE

Os endereços Modbus da variável que será monitorada devem ser inseridos neste parâmetro.

IMPORTANTE
Caso seja selecionado um formato de dados inteiro de 4 bytes ou ponto flutuante, o usuário deverá apenas configurar o primeiro endereço MODBUS do dispositivo conectado ao cartão de I/O.

5 - Parâmetro F_ADDRESS_i:
Dados no formato ponto flutuante. O usuário deve inserir o endereço MODBUS da variável Modbus em ponto flutuante. Até 16 variáveis em ponto flutuante podem ser supervisionadas.

EFETIVANDO A NOVA CONFIGURAÇÃO

O usuário deve alterar o parâmetro ON_APPLY para "*Apply*" para validar a nova configuração estabelecida. Para isso, deve-se acessar o Bloco CCCF e proceder como indicado.

IMPORTANTE
Se este procedimento não for realizado, a configuração selecionada não será atualizada. A configuração foi enviada após a escrita, mas o bloco só vai executar novamente após ON_APPLY ter sido colocado em "*Apply*".

B - Supervisão dos Dados

1 - Parâmetro BVALUE:
O parâmetro BVALUEi supervisiona as variáveis *booleanas* endereçadas pelos parâmetros B_ADDRESSi. O bloco suporta até 96 *Booleanos* que poderão ser mapeados para supervisão do dispositivo Modbus escravo.

2 - Parâmetro IVALUE:
O parâmetro IVALUEi supervisiona as variáveis inteiras endereçadas pelos parâmetros I_ADDRESSi. O bloco suporta até 8 inteiros que poderão ser mapeados para supervisão do dispositivo Modbus escravo.

3 - Parâmetros P_EU_VALUE_A/ P_EU_VALUE_B:
O parâmetro P_EU_VALUE_A/P_EU_VALUE_B supervisiona as variáveis analógicas endereçadas pelos parâmetros P_EU_ADDRESS_i. O bloco suporta até 56 analógicos que poderão ser mapeados para supervisão do dispositivo Modbus escravo.

4 - Parâmetro F_VALUE:
O parâmetro F_VALUE supervisiona as variáveis em ponto flutuante endereçadas pelos parâmetros F_ADDRESS_i. O bloco suporta até 16 pontos flutuantes que poderão ser mapeados para supervisão do dispositivo Modbus escravo.

C - Status de Supervisão

1 - Parâmetro SCAN_STATUS:
Parâmetro de status do *scan* realizado pelo **MB-700** na comunicação com os dispositivos escravos seriais.

BIT	*Mensagem*	*Descrição*
0	Last write message Failed	Quando o valor selecionado em NUMBER_RETRANSMISSIONS for atingido, e o **MB-700** não conseguiu escrever o dado corretamente, o valor de SCAN_STATUS indicará "Last write message Failed".
1	Scan failed	Quando o valor selecionado em NUMBER_RETRANSMISSIONS for atingido, e o **MB-700** não receber resposta do escravo serial, o valor de SCAN_STATUS indicará "Scan Failed".
2	Using configurable view	Se o usuário selecionou SCAN_BEHAVIOR igual a 1 e a comunicação com o escravo foi feita utilizando View, o SCAN_STATUS será "Using Configurable view".
3	Scan Stopped	Indica que o Scan parou por algum motivo.

Tabela 4.2 – Descrição dos Bits de Status

Parâmetros

Idx	*Parâmetro*	*Tipo de Dado (Tamanho)*	*Faixa Válida/ Opções*	*Valor Default*	*Unid*	*Store / Mode*	*Descrição*
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	Veja parâmetro de modo.
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	SLAVE_ADDRESS	Unsigned8	0 to 255	2		S / OOS	Endereço do dipositivo escravo a ser acessado.
8	SERIAL_PORT	Unsigned8	1 : P1 2 : P2	1	1	S / OOS	Número da porta serial onde o dispositivo escravo é conectado.
9	SCAN_BEHAVIOR	Unsigned8	0 : Use Config View 1 : Not Use Config View	0	0	S / OOS	Define o uso ou não de Config View para acessar o dispositivo.
10	SCAN_STATUS	Bitstring(2)	See Scan_Status bitstring description	0	E	D / RO	Indica o status de comunicação como dispositivo escravo.
11	BVALUE	Boolean96		FALSE		N	Dado booleano lido de um dispositivo escravo.
12	IVALUE	8 Integer32		0		N	Valor inteiro lido de um dipositivo escravo.
13	P_EU_VALUE_A	FloatingPoint28		0		N	Porcentagem para o valor EU lido do dispositivo escravo.
14	P_EU_VALUE_B	FloatingPoint28		0		N	Porcentagem para o valor EU lido do dispositivo escravo.

Idx	Parâmetro	Tipo de Dado (Tamanho)	Faixa Válida/ Opções	Valor Default	Unid	Store / Mode	Descrição
15	FVALUE	FloatingPoint16		0.0		N	Valor ponte flutuante lido de um dispositivo escravo..
16	B_ADDRESS1	Unsigned16		65535		S / OOS	Endereço Modbus para localizar uma variável booleana.
…	…	…	…	…	…	…	…
111	B_ADDRESS96	Unsigned16		65535		S / OOS	Endereço Modbus para localizar uma variável booleana.
112	I_ADDRESS1	DS-264				S / OOS	Endereço Modbus para localizar uma variável inteira e tamanho.
…	…	…	…	…	…	…	…
119	I_ADDRESS8	DS-264				S / OOS	Endereço Modbus para localizar uma variável inteira e tamanho.
120	P_EU_ADDRESS_A1	DS-265				S / OOS	Informação para localizar a variável porcentagem e também a conversão de escala.
…	…	…	…	…	…	…	…
147	P_EU_ADDRESS_A28	DS-265				S / OOS	Informação para localizar a variável porcentagem e também a conversão de escala.
148	P_EU_ADDRESS_B1	DS-265				S / OOS	Informação para localizar a variável porcentagem e também a conversão de escala.
…	…	…	…	…	…	…	…
175	P_EU_ADDRESS_B28	DS-265				S / OOS	Informação para localizar a variável porcentagem e também a conversão de escala.
176	F_ADDRESS1	Unsigned16		65535		S / OOS	Endereço Modbus para localizar uma variável floatingpoint.
…	…	…	…	…	…	…	…
191	F_ADDRESS16	Unsigned16		65535		S / OOS	Endereço Modbus para localizar uma variável floatingpoint.
192	UPDATE_EVT	DS-73			Na	D	Este alarme é gerado por qualquer mudança nos dados estáticos.
193	BLOCK_ALM	DS-72			Na	D	O bloco de alarme é utilizado para toda configuração de hardware, falha na conexão ou problemas de sistema nos blocos. A causa do alerta é mostrada no campo subcode. O primeiro alarme a se tornar ativo mudará o status ativo no Status Attribute. Assim que o status Unreported é zerado pela ferramenta de anúncio de alerta, outro bloco pode ser informado sem apagar o status Active, se subcode mudou.

Tabela 4.3 - Legenda: Coluna Store / Mode – Store: armazenamento de dados (D – Dinâmico; N – Não volátil; S – Estático) / Mode: mínimo modo necessário para o usuário conseguir modificar o parâmetro (OOS – Out of Service ou MAN - Manual); Se não tiver Mode, significa que não depende do modo para ser alterado. RO – Somente leitura; Coluna Unid – E: parâmetro Enumerado; Na – Parâmetro adimensional.

Bloco CCCM – Concentrate Control Master

Visão Geral

O Bloco CCCM fornece um modo de troca de dados entre um dispositivo escravo MODBUS e um outro dispositivo escravo MODBUS, para aplicações de controle de processo.

Esquemático

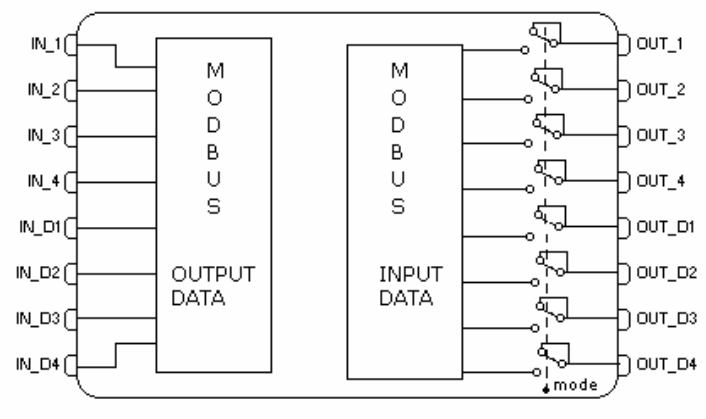


Figura 4.5 – Esquemático do Bloco CCCM

Descrição

O bloco CCCM permite a troca de dados entre equipamentos escravo modbus RTU/TCP. Como cada ponto é configurado individualmente, não é necessário que os equipamentos estejam na mesma rede. Por exemplo, um escravo pode estar conectado a porta serial do **MB-700,** e o outro conectado na rede ethernet ou mesmo em uma outra porta serial de outro **MB-700.** O usuário precisa apenas informar os endereços MODBUS de cada equipamento, endereços das variáveis MODBUS e um "link" interno ao **MB-700,** ligando o parâmetro de entrada ao parâmetro de saída do mesmo bloco CCCM, para que o **MB-700** estabeleça a comunicação "*Peer-to-peer*" entre os dois dispositivos.

O bloco dispõe de status para cada ponto indicando se a comunicação está ok. Este status é informado no parâmetro OUT_xx.Status e IN_xx.Status.

O parâmetro COMM_STATUS é um bitstring com um resumo de todos os status bom, ruim ou não configurado do bloco. Se o bit estiver 1 indica que o ponto esta ruim ou não configurado.

Opcionalmente pode ser configurado para cada ponto um parâmetro de status que vem de uma variável do equipamento escravo Modbus (este status é associado ao parâmetro OUT_xx IN_xx através do argumento MODBUS_ADDRESS_OF_STATUS e segue o padrão ***FOUNDATION™ Fieldbus***). Ou seja, se desejar que uma variável de indicação de status interna do escravo Modbus seja tranferida para um outro escravo, ela pode ser configurada neste status.

NOTA
O parâmetro MODE_BLK do Bloco CCCM deve ser setado para "Auto" para que este bloco funcione corretamente. Veja a descrição deste parâmetro no item anterior, Bloco CCSM.

A - Modo de Funcionamento da Comunicação "*Peer-to-peer*" em Dispositivo MODBUS

O Bloco CCCM deve ser configurado para estabelecer a comunicação de dados entre dois escravos através do **MB-700**. Portanto, o usuário deve conectar os endereços MODBUS das entradas e saídas.

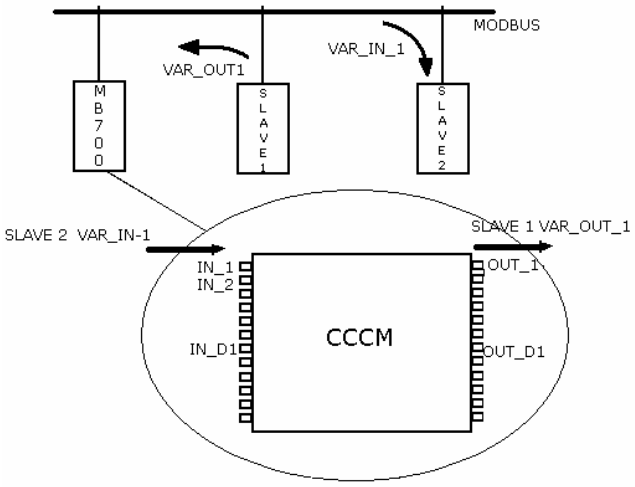


Figura 4.6 –Funcionamento da Comunicação Peer-to-peer em Dispositivo Modbus

Um exemplo de Peer-to-Peer é mostrado na figura 4.6. Suponha que deseja-se ler uma variável do escravo 1 (slave 1) e escrever no escravo 2 (slave 2). Cada variável (de leitura do escravo 1 e de escrita do escravo 2) tem um endereço Modbus, que deve ser obtido do equipamento escravo (consultar manual do fabricante).

O primeiro passo seria identificar quais os endereços dos equipamentos escravo modbus, e identificar as variáveis que se deseja ler e escrever. A variável de Leitura (VAR_OUT_1) é associada à **saída** (OUT_xx) do bloco CCCM. A variável de escrita (VAR_IN_1), é associada à **entrada** (IN_xx) do bloco CCCM.

Após identificar as variáveis, é necessário relacionar a variável de entrada com a variável de saida. Existem duas formas de fazer este relacionamento:

- **(*)Automático -** onde cada entrada e saída já esta relacionada, ou seja, OUT_1 → IN_1, OUT_2→ IN_2, ..., OUT_D1→ IN_D1, OUT_D2 → IN_D2. Para habilitar esta opção basta colocar o parâmetro ALERT_KEY = 5. Qualquer valor no parâmetro ALERT_KEY diferente de 5, desabilita a opção.
- **Através de "Link" –** neste caso deve ser feito um link entre a saída desejada e a entrada desejada. A Figura 4.7 exemplifica esta conexão usando "Link".

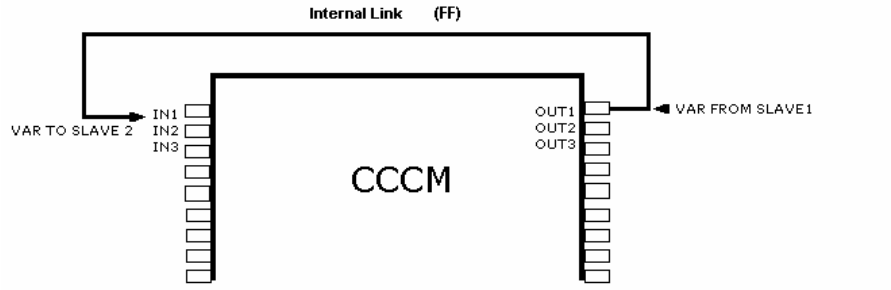


Figura 4.7 – Conexão Peer-to-peer no Bloco CCCM

Quando a opção de Relacionamento for Através de "Link", segue as seguinte observações:

- Para verificar se um ponto esta sendo escrito ou não, deve ser observado o parâmetro COMM_STATUS, pois o status do parâmetro IN_xx indica o status da variável OUT_xx que está linkado a ele. Por exemplo, no exemplo acima um status "good" no parâmetro IN_1 indica que o valor que esta sendo lido pelo parâmetro OUT_1 do slave 1 está Ok, mas não indica que o slave 2 esta recebendo o valor. A escrita deve ser checada no parâmetro COMM_STATUS.
- O Modo do bloco em Manual significa que as entradas estão sendo lidas mas não estão sendo repassadas para o bloco. O operador poderá alterar o valor da variável OUT_xx que este valor repassará para IN_x e para o equipamento escravo relacionado a ele.

(*) Quando a opção de Relacionamento for Automático, segue as seguinte observações:

- Nesta opção o modo Manual não está disponivel.
- O status da saida (OUT_x) reflete se o ponto (VALUE) está sendo lido ou não, ou corresponde ao valor do status se o MODBUS_ADDRESS_OF_STATUS foi configurado. O status da entrada (IN_x) reflete se o ponto conseguiu ser escrito ou não. Se for configurado MODBUS_ADDRESS_OF_STATUS ele reflete o valor lido do status do OUT_x correspondente.
- Caso o ponto não consiga ser lido do OUT_x, ele continuará sendo escrito do último comando válido de IN_x.
- Este relacionamento é mais rápido que o relacionamento por "Link", pois ele atualiza automaticamente o comando de saida assim que receber o comando de entrada, enquanto que o outro tem de esperar o processamento do bloco.

Nota
(*) Esta opção está disponivel à partir da versão de firmware 4.5.5 do MB-700.

Maiores detalhes sobre como configurar uma conexão "*peer-to-peer*" são apresentados no Capítulo 5 deste Manual.

B - Endereçamento

1. Parâmetros EU_ADDRESS_IN1 a EU_ADDRESS_IN4:

Nestes parâmetros o usuário deve informar os endereços e parâmetros Modbus do escravo que recebe o valor da variável analógica, ou seja, o escravo para onde será enviado o valor no parâmetro de entrada do bloco CCCM. Para maiores detalhes sobre conversão de escala, deve-se consultar o Apêndice C. Os seguintes parâmetros devem ser informados:

- FROM_EU_100%
- FROM_EU_0%
- TO_EU_100%
- TO_EU_0%

- DATATYPE

O formato para o dado desejado entre os diversos tipos disponíveis pode ser escolhido neste parâmetro. A tabela que mostra os tipos disponíveis se encontra no Apêndice C.

- PORT_NUMBER

O usuário deve informar a porta do **MB-700** a ser utilizada para a comunicação entre os escravos: Ethernet, P1 ou P2.

- SLAVE_ADDRESS

Neste parâmetro, deve-se informar o endereço Modbus do dispositivo escravo.

- MODBUS_ADDRESS_OF_VALUE

Neste parâmetro, deve-se informar o endereço Modbus da variável a ser escrita, cujo valor é "y".

- MODBUS_ADDRESS_OF_STATUS

Neste parâmetro, deve-se informar o endereço do Modbus da variável a ser escrita no dispositivo escravo.

2. Parâmetros EU_ADDRESS_OUT1 a EU_ADDRESS_OUT4

Nestes parâmetros, o usuário deve informar endereços e parâmetros Modbus do escravo que indicam o valor da variável analógica, ou seja, a variável a ser lida. O usuário deve configurar os parâmetros mostrados a seguir. Para maiores detalhes sobre conversão de escala, consulte o Apêndice C.

- FROM_EU_100%
- FROM_EU_0%
- TO_EU_100%
- TO_EU_0%

- DATATYPE

O formato para o dado desejado entre os diversos tipos disponíveis pode ser escolhido neste parâmetro. A tabela que mostra os tipos disponíveis se encontra no Apêndice C.

- PORT_NUMBER

O usuário deve informar a porta do **MB-700** a ser utilizada para a comunicação entre os escravos:

Ethernet, P1 ou P2.

- SLAVE_ADDRESS

O endereço Modbus do dispositivo escravo deve ser informado neste parâmetro.

- MODBUS_ADDRESS_OF_VALUE

O endereço Modbus da variável lida do escravo deve ser informado neste parâmetro.

- MODBUS_ADDRESS_OF_STATUS

O endereço do status da variável lida do escravo deve ser informado neste parâmetro.

3. Parâmetros EU_ADDRESS_IN_D1 a EU_ADDRESS_IN_D2:

Nestes parâmetros o usuário deverá informar endereços e parâmetros Modbus do escravo onde o valor booleano será escrito. O usuário deverá informar os seguintes parâmetros:

- ✓ PORT_NUMBER

O usuário deve informar a porta do **MB-700** a ser utilizada para a comunicação entre os escravos: Ethernet, P1 ou P2.

- ✓ SLAVE_ADDRESS

O endereço Modbus do dispositivo escravo deve ser informado neste parâmetro.

- ✓ MODBUS_ADDRESS_OF_VALUE

O endereço Modbus onde será escrito o valor da variável deve ser inserido neste parâmetro. Informe o endereço da variável MODBUS do dispositivo escravo a ser escrito, cujo valor é o valor do parâmetro de entrada IN_DX

- ✓ MODBUS_ADDRESS_OF_STATUS

O endereço do status da variável a ser escrita deve ser informado neste parâmetro.

4. Parâmetros EU_ADDRESS_OUT_D1 a EU_ADDRESS_OUT_D2

Nestes parâmetros o usuário deverá informar endereços e parâmetros Modbus do escravo que informa o valor da variável digital, ou seja, a variável a ser lida. Usuário deverá informar os seguintes parâmetros:

- ✓ PORT_NUMBER

O usuário deve informar a porta do **MB-700** a ser utilizada para a comunicação entre os escravos: Ethernet, P1 ou P2.

- ✓ SLAVE_ADDRESS

Informe o endereço Modbus do dispositivo escravo.

- ✓ MODBUS_ADDRESS_OF_VALUE

O endereço Modbus da variável lida do escravo deve ser inserido neste parâmetro, e que se tornará disponível no parâmetro de saída OUT_DX deste bloco.

- ✓ MODBUS_ADDRESS_OF_STATUS

O endereço do status da variável a ser escrita deve ser informado neste parâmetro.

Enviar a configuração para o MB-700

O usuário deve alterar o parâmetro ON_APPLY para “APPLY” para atualizar a nova configuração estabelecida. Para isso, deve acessar o Bloco CCCF e proceder como indicado.

IMPORTANTE
Se o usuário não realizar este procedimento para a configuração selecionada, as configurações dos parâmetros não serão efetivadas no **MB-700**. A configuração foi atualizada após a escrita, mas o bloco só vai executar novamente após ON_APPLY ter sido colocado em “Apply”.

C - Monitoração dos Dados

- Parâmetros de monitoração de entrada analógica do Bloco CCCM: IN1, IN2, IN3, IN4.
- Parâmetros de monitoração de saída analógica do Bloco CCCM: OUT1, OUT2, OUT3, OUT 4.
- Parâmetros de monitoração de entrada digital do Bloco CCCM: IN_D1, IN_D2, IN_D3, IN_D4.

- Parâmetros de monitoração de saída digital do Bloco CCCM: OUT_D1, OUT _D2, OUT _D3, OUT_D4.

D - Status da Supervisão

1 - Parâmetro COMM_STATUS
Este parâmetro indica se a comunicação entre os escravos foi estabelecida corretamente. Se o bit correspondente estiver em nível lógico 1, isto significa que houve um erro na leitura/escrita do respectivo parâmetro.

Bit	*Descrição*	*Significado*
0	IN_1	Falha de Escrita do parâmetro IN_1 no modbus ou parâmetro não configurado.
1	IN_2	Falha de Escrita do parâmetro IN_2 no modbus ou parâmetro não configurado.
2	IN_3	Falha de Escrita do parâmetro IN_3 no modbus ou parâmetro não configurado.
3	IN_4	Falha de Escrita do parâmetro IN_4 no modbus ou parâmetro não configurado.
4	IN_D1	Falha de Escrita do parâmetro IN_D1 no modbus ou parâmetro não configurado.
5	IN_D2	Falha de Escrita do parâmetro IN_D2 no modbus ou parâmetro não configurado.
6	IN_D3	Falha de Escrita do parâmetro IN_D3 no modbus ou parâmetro não configurado.
7	IN_D4	Falha de Escrita do parâmetro IN_D4 no modbus ou parâmetro não configurado.
8	OUT_1	Falha de Leitura do parâmetro OUT_1 no modbus ou parâmetro não configurado.
9	OUT_2	Falha de Leitura do parâmetro OUT_2 no modbus ou parâmetro não configurado.
10	OUT_3	Falha de Leitura do parâmetro OUT_3 no modbus ou parâmetro não configurado.
11	OUT_4	Falha de Leitura do parâmetro OUT_4 no modbus ou parâmetro não configurado.
12	OUT_D1	Falha de Leitura do parâmetro OUT_D1 no modbus ou parâmetro não configurado.
13	OUT_D2	Falha de Leitura do parâmetro OUT_D2 no modbus ou parâmetro não configurado.
14	OUT_D3	Falha de Leitura do parâmetro OUT_D3 no modbus ou parâmetro não configurado.
15	OUT_D4	Falha de Leitura do parâmetro OUT_D4 no modbus ou parâmetro não configurado.

Tabela 4.4 – Descrição do Parâmetro de Status

Status do parâmetro
Cada parâmetro de entrada ou saída possui um status correspondente. Este status pode ser atualizado pelo equipamento escravo através da configuração do MODBUS_ADRESS_OF_STATUS. Isso ocorre somente se o equipamento escravo possuir um status (este status segue o padrão ***FOUNDATION™ Fieldbus***). Veja maiores detalhes de como configurar um status do escravo no Manual dos Blocos Funcionais. Caso o parâmetro MODBUS_ADDRESS_OF_STATUS não esteja configurado, o Bloco CCCM automaticamente atualizará o status para "*Good Non Cascade*", quando a comunicação com o escravo estiver OK, ou "*Bad No Communication with last Value*" e "*Non-Specific*, se a comunicação falhar.

Parâmetros

Idx	*Parâmetro*	*Tipo de Dado (Tamanho)*	*Faixa Válida/ Opções*	*Valor Default*	*Unid*	*Store / Mode*	*Descrição*
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S / OOS	
5	MODE_BLK	DS-69		O/S	Na	S	Veja parâmetro de Modo.
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	COMM_STATUS	Bitstring(2)		0	E	D / RO	Indica se a comunicação com o escravo é boa ou não. Cada bit corresponde a uma variável Modbus.
8	IN1	DS-65				N	Entrada Analógica 1.
9	EU_ADDRESS_IN1	DS-266				S / OOS	Informação para gerar constantes A e B na equação Y=A*X+B mais os endereços em um dispositivo escravo.
10	IN2	DS-65				N	Entrada Analógica 2.
11	EU_ADDRESS _IN2	DS-266				S / OOS	Informação para gerar constantes A e B na equação Y=A*X+B mais os endereços em um dispositivo escravo
12	IN3	DS-65				N	Entrada Analógica 3.
13	EU_ADDRESS _IN3	DS-266				S / OOS	Informação para gerar constantes A e B na equação Y=A*X+B mais os endereços em um dispositivo escravo
14	IN4	DS-65				N	Entrada analógica 4.
15	EU_ADDRESS _IN4	DS-266				S / OOS	Informação para gerar constantes A e B na equação Y=A*X+B mais os endereços em um dispositivo escravo
16	IN_D1	DS-66				N	Entrada discreta 1.
17	ADDRESS _IN_D1	DS-267				S / OOS	Endereços no dispositivo escravo.
18	IN_D2	DS-66				N	Entrada discreta 2.
19	ADDRESS _IN_D2	DS-267				S / OOS	Endereços no dispositivo escravo.
20	IN_D3	DS-66				N	Entrada discreta 3.
21	ADDRESS _IN_D3	DS-267				S / OOS	Endereços no dispositivo escravo.
22	IN_D4	DS-66				N	Entrada discreta 4.
23	ADDRESS _IN_D4	DS-267				S / OOS	Endereços no dispositivo escravo.
24	OUT1	DS-65				N / Man	Saída analógica 1.
25	EU_ADDRESS _OUT1	DS-266				S / OOS	Informação para gerar constantes A e B na equação Y=A*X+B mais os endereços em um dispositivo escravo.
26	OUT2	DS-65				N / Man	Saída analógica 2.
27	EU_ADDRESS _OUT2	DS-266				S / OOS	Informação para gerar constantes A e B na equação Y=A*X+B mais os endereços em um dispositivo escravo
28	OUT3	DS-65				N / Man	Saída analógica 3.
29	EU_ADDRESS _OUT3	DS-266				S / OOS	Informação para gerar constantes A e B na equação Y=A*X+B mais os endereços em um dispositivo escravo
30	OUT4	DS-65				N / Man	Saída analógica 4.
31	EU_ADDRESS _OUT4	DS-266				S / OOS	Informação para gerar constantes A e B na equação Y=A*X+B mais os endereços em um dispositivo escravo
32	OUT_D1	DS-66				N / Man	Saída discreta 1.
33	ADDRESS _OUT_D1	DS-267				S / OOS	Endereços em um dispositivo escravo.
34	OUT2_D2	DS-66				N / Man	Saída discreta 2.
35	ADDRESS _OUT_D2	DS-267				S / OOS	Endereços em um dispositivo escravo.
36	OUT_D3	DS-66				N / Man	Saída discreta 3.

Idx	Parâmetro	Tipo de Dado (Tamanho)	Faixa Válida/ Opções	Valor Default	Unid	Store / Mode	Descrição
37	ADDRESS _OUT_D3	DS-267				S / OOS	Endereços em um dispositivo escravo.
38	OUT_D4	DS-66				N / Man	Saída discreta 4.
39	ADDRESS _OUT_D4	DS-267				S / OOS	Endereços em um dispositivo escravo.
40	UPDATE_EVT	DS-73			Na	D	Este alarme é gerado por qualquer mudança nos dados estáticos
41	BLOCK_ALM	DS-72			Na	D	O bloco de alarme é utilizado para toda configuração de hardware, falha na conexão ou problemas de sistema nos blocos. A causa do alerta é mostrada no campo subcode. O primeiro alarme a se tornar ativo mudará o status ativo no Status Attribute. Assim que o status Unreported é zerado pela ferramenta de anúncio de alerta, outro bloco pode ser informado sem apagar o status Active, se subcode mudou..

Tabela 4.5 - Legenda: Coluna Store / Mode – Store: armazenamento de dados (D – Dinâmico; N – Não volátil; S – Estático) / Mode: mínimo modo necessário para o usuário conseguir modificar o parâmetro (OOS – Out of Service ou MAN - Manual); Se não tiver Mode, significa que não depende do modo para ser alterado. RO – Somente leitura; Coluna Unid – E: parâmetro Enumerado; Na – Parâmetro adimensional.

Bloco CCDL – Concentrate Data Logger

Visão Geral

Este bloco fornece um mecanismo para buscar dados armazenados de dispositivos escravos Modbus de uma maneira otimizada. Uma variável Modbus especificada na configuração será usada para indicar se é necessário escanear os dados armazenados, cujos endereços Modbus são fornecidos pelo usuário.

Descrição

Mecanismo otimizado para trazer os dados armazenados do LC700:

- Para iniciar o Bloco FIFO *Data Logger* no LC700 quando a comunicação com o mestre do **MB-700** (por exemplo, um supervisório) falhar, o Bloco CCDL envia um comando modbus para a variável no LC700 apontada por STOP_SCAN_ADDRESS. O bloco escreve TRUE na variável quando deseja habilitar, e FALSE quando deseja desabilitar o FIFO *Data Logger*.

- Se a supervisão dos dados está normal (por exemplo, se um supervisório está comunicando normalmente com um bloco de supervisão (CCSM)), no scan normal do Bloco CCDL ele supervisiona a variável modbus apontada por NEED_SCAN_ADDRESS para saber se existe dado armazenado no Bloco FIFO *Data Logger*. Se a variável for diferente de zero, isto indica que existe dado no Bloco *Data Logger*. O parâmetro LOGGER_ADDRESS aponta para o endereço de início do Bloco *Data Logger* (FIFO.CTW). O parâmetro NUM_REGISTER mostra a quantidade de registros armazenados no Bloco CCDL.

1 – Parâmetro TYPE_OF_LOG
O parâmetro TYPE_OF_LOG configura o *time stamp* do *Data Logger*.

2 – Parâmetro P_EU_ADDRESS_A1
O parâmetro P_EU_ADDRESS_A1 configura a escala de conversão e o formato dos dados do Bloco FIFO Data Logger (estes parâmetros devem ter a mesma configuração do respectivo Bloco FIFO *Data Logger* do LC700). Para maiores detalhes sobre conversão de escala, deve-se referir ao Apêndice C. O usuário deve configurar os seguintes parâmetros:

- ✓ FROM_EU_100%
- ✓ FROM_EU_0%
- ✓ TO_EU_100%
- ✓ To_EU_0%

- ✓ DATATYPE

Escolha do formato do dado lido do dispositivo. Os tipos de dados suportados pelo Bloco *Data Logger* (que serão convertidos para *Float*) são *Float* e *Integer*16.

- ✓ MODBUS_ADDRESS_VALUE

Não é utilizado.

- Blocos em Cascata – O Bloco CCDL pode armazenar até 280 registros. Se um Bloco CCDL não for suficientemente longo para armazenar todo os dados logados no Bloco FIFO, é possível dispor blocos em cascata tantas vezes quantas forem necessárias. Para fazer isso, o usuário deverá criar quantas instâncias do Bloco CCDL achar necessário e configurar todos eles com as mesmas informações (MODE_BLK, SLAVE_ADDRESS, NEED_SCAN_ADDRESS, LOGGER_ADDRESS, etc.). Então quando os dados de *log* trazidos do Bloco FIFO não couberem no Bloco CCDL, e se existir um outro Bloco CCDL configurado para aquela FIFO, este bloco receberá o restante dos dados de *log*. O Bloco CCDL indica no parâmetro NEXT_BLOCK_LOG, qual é o próximo bloco com o restante dos dados de *log*.

Parâmetros

Idx	*Parâmetro*	*Tipo de dados (tamanho)*	*Faixa Válida/ Opções*	*Valor Default*	*Unid*	*Store / Mode*	*Descrição*
1	ST_REV	Unsigned16		0	None	S	
2	TAG_DESC	OctString(32)		Spaces	Na	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	Na	S	Veja parâmetros de modo.
6	BLOCK_ERR	Bitstring(2)			E	D / RO	
7	SLAVE_ADDRESS	Unsigned8	0 to 255	2		S / OOS	Endereço do dispositivo escravo a ser escaneado.
8	SERIAL_PORT	Unsigned8	1 : P1 2 : P2	1	1	S / OOS	Número da porta serial onde o dispositivo escravo será conectado.
9	SCAN_STATUS	Bitstring(2)	Veja status de Scan	0	E	D/RO	Indica o status da comunicação com o dispositivo escravo.
10	NEED_SCAN_ADDRESS	Unsigned16	1-49999	0	E	S / OOS	Este é o endereço Modbus da flag que indica se é necessário escanear os dados armazenados.
11	CLEAR_LOG_ADDRESS	Unsigned16	1-9999	0	E	S / OOS	Endereço da variável no escravo Modbus para apagar a data logger. Pode ser usado para apagar a data logger após a leitura. Se o endereço é dafault (0), então o mecanismo é desabilitado.
12	NEED_SCAN_COUNTER	Unsigned16		0	E	D / RO	Indica o número de scans para os dados armazenados. Esta variável pode ser utilizada pelo OPC Server para o trigger de um comando de scan para ler LOG_FVALUEx e TIME_STAMP_x.
13	TYPE_OF_LOG	Unsigned8	0:Only last time stamp 1:Every sample 2:No time stamp	0	E	S / OOS	Seleção do tipo de armazenamento (log).
14	P_EU_ADDRESS_A1	DS-265				S / OOS	Informação para localizar um variável float/percentage e também a escala de conversão.
15	LOGGER_ADDRESS	Unsigned16		0	E	S / OOS	Endereço Modbus do começo da data logger.
16	NUM_REGISTER	Unsigned16		0	E	D / RO	Número de registradores FIFO armazenados.
17	STOP_SCAN_ADDRESS	Unsigned16	0 - 9999	0	E	S	O MB-700 pode forçar o trigger da data logger escrevendo TRUE para uma variável Modbus no dispositivo Modbus escravo.
18	NEXT_BLOCK_LOG	Visiblestring(32)				S	Tag do próximo bloco CCDL associado a este data logger.
19	LOG_FVALUE_1	28 Float		0.0		N/RO	Valor Float escaneado do dispositivo escravo.
...	...	...	...	...	...	...	...
28	LOG_FVALUE_10	28 Float		0.0		N/RO	Valor Float escaneado do dispositivo escravo.
29	TIME_STAMP_1	28 Time Value				N/RO	Time stamp das amostras em LOG_FVALUE1
...	...	...	...	...	...	...	...
38	TIME_STAMP_10	28 Time Value				N/RO	Time stamp das amostras em _FVALUE10
39	UPDATE_EVT	DS-73			Na	D	Este alerta é gerado por cada mudança nos dados estáticos.

Idx	Parâmetro	Tipo de dados (tamanho)	Faixa Válida/ Opções	Valor Default	Unid	Store / Mode	Descrição
40	BLOCK_ALM	DS-72			Na	D	O alarme do bloco é usado para toda configuração, hardware, falha de conexão ou problemas de sistemas no bloco. A causa do alerta é mostrada no campo subcode. O primeiro alerta a se tornar ativo irá alterar o status Active no atributo status. Tão logo Unreported status é apagado pelo alerta de aviso de tarefa, outro alerta de bloco pode ser mostrado sem apagar o status Active, se subcode mudou.

Tabela 4.6 - Legenda: Coluna Store / Mode – Store: armazenamento de dados (D – Dinâmico; N – Não volátil; S – Estático) / Mode: mínimo modo necessário para o usuário conseguir modificar o parâmetro (OOS – Out of Service ou MAN - Manual); Se não tiver Mode, significa que não depende do modo para ser alterado. RO – Somente leitura; Coluna Unid – E: parâmetro Enumerado; Na – Parâmetro adimensional.

ADICIONANDO BLOCOS AO MB-700

MB-700 como Mestre Serial MODBUS

Cenário

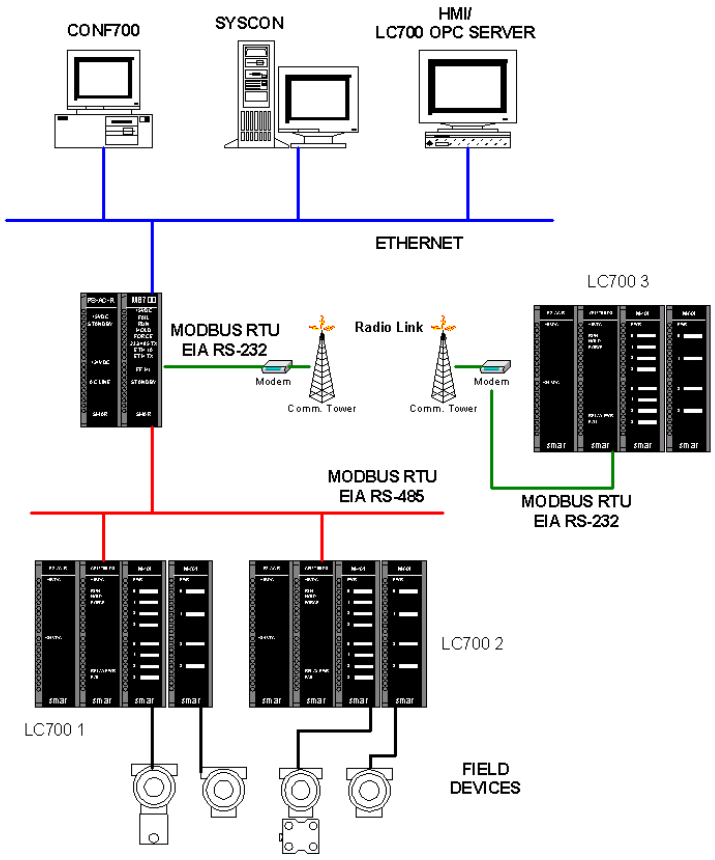


Figura 5.1 – Exemplo Arquitetura do MB-700 como Mestre Serial MODBUS

Descrição

Neste cenário, o **MB-700** pode realizar as seguintes funções, simultaneamente, quando o parâmetro *Bypass Direction* estiver configurado com "TCP to Serial":

- ✓ *Bypass* (TCP/IP to Serial): O MB-700 repassa a mensagem Modbus transmitindo a um nível inferior da rede. O MB-700 atua como um conversor TCP/IP Modbus para Modbus RTU (485 ou 232).
- ✓ Concentrador: As variáveis Modbus dos dispositivos escravos são lidas, processadas (conversão de *Data Type* e Escala) e armazenadas em parâmetros do Bloco CCSM. Ao invés do supervisório se comunicar diretamente com os dispositivos escravos, comunicará somente com o MB-700.
- ✓ *Peer-to-Peer*: Dois escravos Modbus trocam dados entre si. A função "*Peer-to-peer*" possibilita esta troca. Os escravos podem estar em redes Modbus diferentes e mesmo assim podem trocar dados entre si.

O sistema da Figura 5.1 acima é indicado para operações de controle remoto, no qual o processo a ser controlado está localizado numa área distante do sistema supervisório. O usuário não precisa se deslocar até a área, basta que um MB-700 esteja conectado a uma rede Ethernet.

O MB-700 possui duas portas seriais, EIA-232/EIA-485, as quais permitem a conexão com controladores lógicos através de uma rede MODBUS ou através de um modem ou rádio. Utilizando

a porta EIA-485 pode-se criar uma sub rede de controladores lógicos, ou integrar uma sub rede já existente. Para controle remoto, em localidades distantes, um modem pode ser utilizado na porta EIA-232 para estabelecer a comunicação entre o sistema supervisório e sistemas de configuração com os controladores na planta de processo.

Nesta aplicação, o MB-700 possui duas funções básicas: "*Bypass*" MODBUS e Concentradora de Dados. O "*Bypass*" MODBUS ocorre quando um dado MODBUS chega até o MB-700, isto é., quando o CONF700 (ou qualquer outro sistema de configuração MODBUS) está enviando comandos para configuração de algum dispositivo escravo na sub rede MODBUS. Neste caso, o MB-700 repassa a mensagem e a transmite para o escravo solicitado. O MB-700 também possui a função Concentrador de Dados. Ao invés do sistema de supervisão acessar um dispositivo de campo em particular, ele apenas procura os dados monitorados na memória do MB-700. Os dados coletados são enviados dentro de uma base de tempo determinada para o MB-700, e o sistema supervisório busca esses dados diretamente no MB-700 aumentando a velocidade de acesso.

O MB-700 é configurado através do SYSCON. O Bloco CCSM configura a função concentrador do MB-700, e o Bloco CCCM configura a comunicação "*Peer-to-peer*". O Bloco CCCF é utilizado para configurar os parâmetros de comunicação do MB-700.

O MB-700 possui um IP dentro da rede, utilizado para comunicação entre o sistema de operação e o MB-700. Cada escravo da rede possui um endereço MODBUS que deve ser informado ao SYSCON. Cada E/S e os respectivos endereços também devem ser informados ao SYSCON. Na configuração mostrada na Figura 5.1, a função "*Peer-to-peer*" pode ser utilizada.

Configuração *OFF LINE*

IMPORTANTE
Antes de iniciar a configuração do MB-700, o usuário deve referir-se ao Capítulo 4 deste manual.

A - Configurando o IP do MB-700

O IP do MB-700 deve ser configurado. Para isto veja o Capítulo 2 deste manual "Conectando o MB-700 à Sub Rede".

B - Inicializando o SYSCON

- Adicionar o MB-700 ao SYSCON. Inicie o SYSCON;
- Clique em *Project →New → Project*. Em seguida clique em *Fieldbus Networks → New Fieldbus*;
- Selecione *Expand* para abrir uma janela auxiliar. Nesta nova janela, clique com o botão da direita do mouse em *New→Bridge*. Selecione o fabricante "Smar", e em seguida o dispositivo "MB-700"; e
- Salve.

Para maiores detalhes, veja o manual do SYSCON e o Capítulo 3 deste manual.

IMPORTANTE
O usuário deverá se certificar de que Smar.DFIOLEServer.0 esteja configurado no menu *Communication Settings*. Caso contrário, a comunicação entre MB-700 e SYSCON não será efetuada.

No item abaixo, é explicada a configuração mínima exigida para o MB-700 trabalhar nas funções "*Bypass*", "Concentrador" e "*Peer-to-peer*".

C - Configurando Blocos

1) MB-700 trabalhando como *Bypass*

Para o MB-700 trabalhar como *Bypass,* é necessário configurar, apenas os Blocos *Resource* e CCCF.

Bloco *RESOURCE*

Passos para configurar o Bloco *Resource*:

- O único parâmetro a ser configurado é o MODE_BLK. Certifique-se de MODE_BLK.TARGET="Auto";
- Clique com o botão direito do mouse sobre <FB_VFD>. Uma janela aparecerá. Clicar em *New Block*;

- Selecione o Bloco *RESOURCE*. Clique com o botão da direita no bloco criado, e em seguida selecione *Off Line Characterization*;
- Localize o parâmetro MODE_BLK.TARGET e coloque-o em “Auto”.

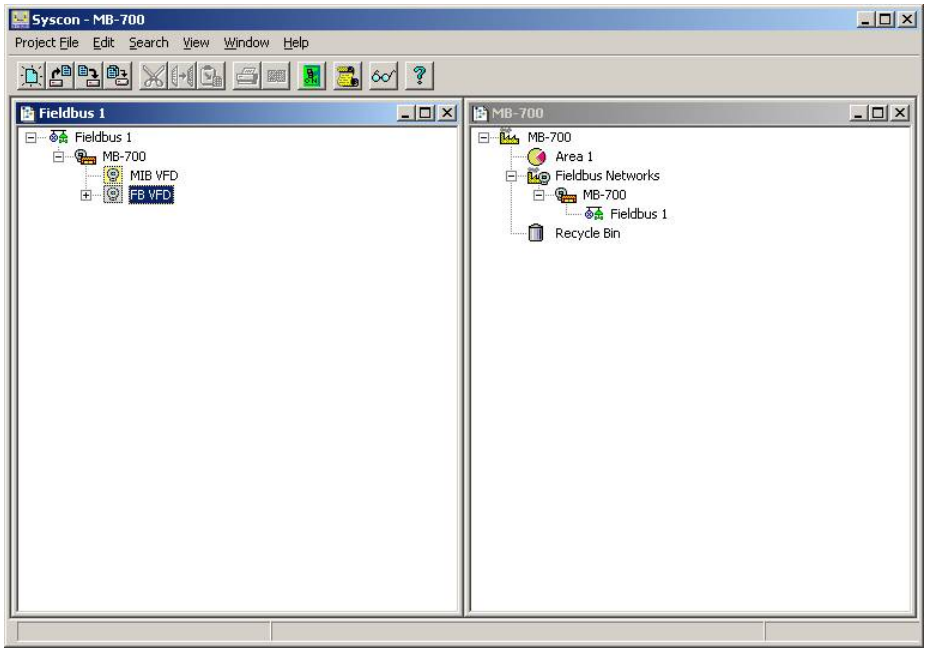


Figura 5.2 – Janelas do Syscon

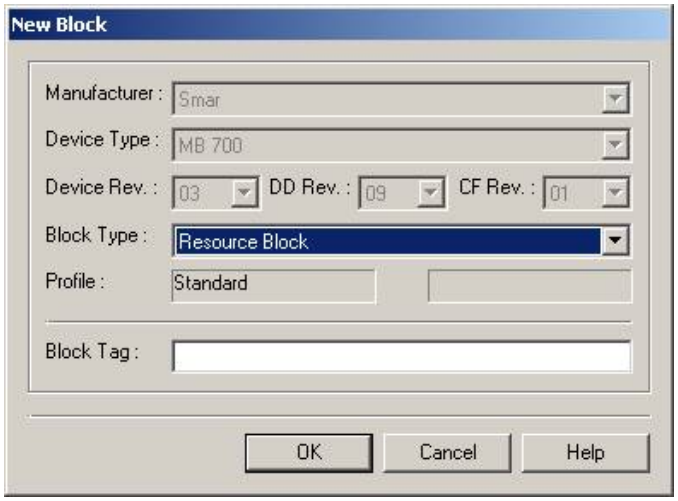


Figura 5.3 – Tela para Inserir Bloco

Bloco CCCF

Passos para configurar o Bloco CCCF:

- No SYSCON localize o MB-700;
- Clique com o botão direito do mouse sobre <FB_VFD>. Uma janela aparecerá, clique em *New Block*;
- Selecione o Bloco CCCF. Clique com o botão da direita no bloco criado e depois selecione *Off Line Characterization*;

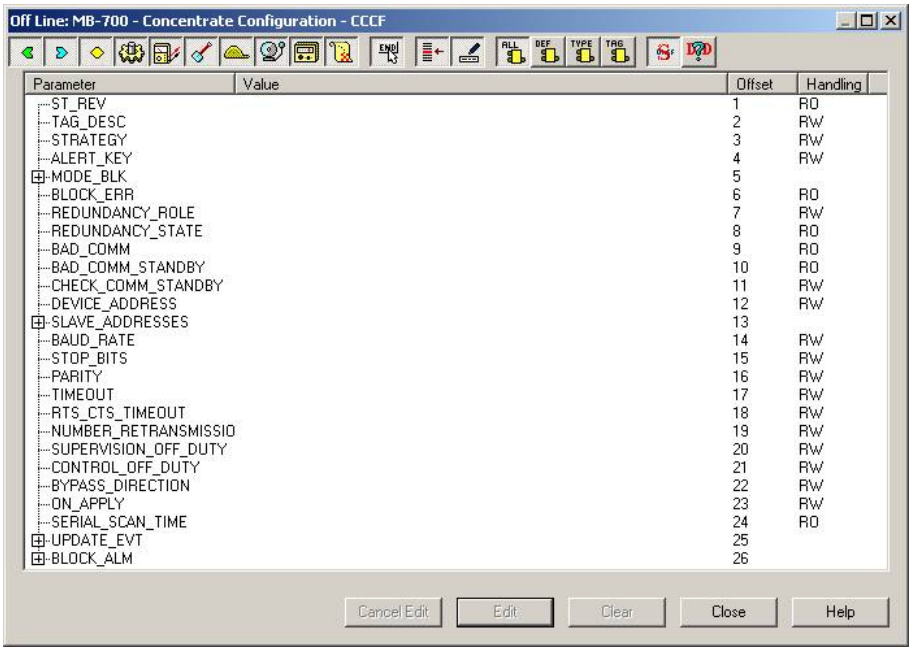


Figura 5.4 –Tela Configuração do Bloco CCCF

Ajuste os seguintes parâmetros para:

- ✓ BAUD_RATE: 57600;
- ✓ BYPASS_DIRECTION: TCP to Serial;
- ✓ NUMBER_RETRANSMISSIONS: Escrever “3";
- ✓ TIMEOUT: "1000" (1000 ms);
- ✓ MODE_BLK: Selecione "Auto";
- ✓ DEVICE_ADDRESS: Certifique-se na sua rede MODBUS não haja nenhum outro endereço com o número mostrado nesse parâmetro. O default é 247;
- ✓ ON_APPLY: Selecione este parâmetro para “Apply”.

Para melhor visualização, renomeie os blocos recém criados. Clique sobre o botão  para visualizar a descrição dos blocos. Clicando com o botão direito sobre Block 1, um menu será aberto.

- O usuário deve clicar em “*Attributes*” e alterar o *Block Tag* segundo a sua conveniência. Este procedimento deve ser repetido para os demais blocos;
- Ao final, uma tela como a Figura 5.5 abaixo deve aparecer no SYSCON.

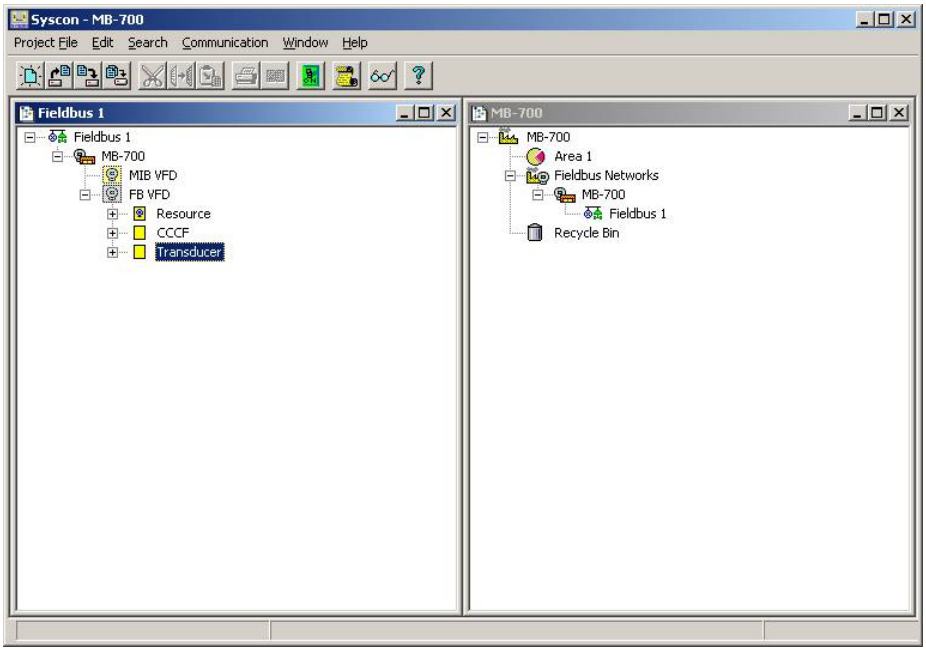


Figura 5.5 – Exemplo Configuração de Blocos do MB-700

2) MB-700 trabalhando como Concentrador

Para o MB-700 trabalhar como Concentrador, é necessário, além de configurar os blocos anteriores, configurar o Bloco CCSM.

Bloco CCSM:

NOTA
Os endereços MODBUS mostrados apresentam o seguinte endereçamento: - 0001 a 9999 ➔ Saídas Digitais - 10001 a 19999 ➔ Entradas Digitais - 30001 a 39999 ➔ Entradas Analógicas - 40001 a 49999 ➔ Saídas Analógicas

Em determinados momentos, o usuário deverá informar ao SYSCON os endereços Modbus de alguns dispositivos. No caso de dados no formato inteiro de 4 *bytes* e *float*, o usuário sempre anotará APENAS o primeiro endereço, já que estes tipos de dados ocupam 2 posições.

- No SYSCON localize o MB-700;
- Clique com o botão direito do mouse sobre <FB_VFD>. Uma janela aparecerá, clique em *New Block*;
- Selecione o Bloco CCSM. Clique com o botão da direita no bloco criado e depois selecione *Off Line Characterization*;
- Coloque MODE_BLK .TARGET em “Auto”;
- O parâmetro SLAVE_ADDR deverá conter o endereço do LC700 1 ou o endereço Modbus do escravo. Neste caso o endereço é 1;
- O parâmetro SERIAL_PORT deverá ser configurado para “P1”, para selecionar a comunicação através das portas EIA-232 e EIA-485.

NOTA
Se o usuário estiver utilizando dispositivos escravos MODBUS da SMAR, poderá alterar o parâmetro SCAN_BEHAVIOR para “*Using Config View*”, e assim poderá obter comunicação mais rápida entre MB-700 e os dispositivos escravos.

No cenário proposto da Figura 5.1, o LC700 1 possui um cartão de E/S conectado a oito chaves discretas digitais, oito saídas digitais representadas no diagrama por LEDs, e duas entradas para dois transmissores.

Considerando que o LC7001 tenha endereço 1 na rede Modbus, e desejando-se configurar os pontos da Tabela 5.1 a seguir, deve-se observar que os endereços MODBUS das entradas das variáveis a serem monitoradas deverão também ser informados.

Entrada/Saída	*Tipo de Dado*	*Endereço MODBUS*
LED 1	Digital	00001
LED 2	Digital	00002
LED 3	Digital	00003
LED 4	Digital	00004
LED 5	Digital	00005
LED 6	Digital	00006
LED 7	Digital	00007
LED 8	Digital	00008
Chave 1	Digital	10001
Chave 2	Digital	10002
Chave 3	Digital	10003
Chave 4	Digital	10004
Chave 5	Digital	10005
Chave 6	Digital	10006
Chave 7	Digital	10007
Chave 8	Digital	10008
Transmissor 1	Porcentagem	30001-30002
Transmissor 2	Porcentagem	30003-30004

Tabela 5.1 – Exemplo de Configuração de Pontos Modbus de um Escravo Modbus

- Na janela do SYSCON clique com o botão direito do mouse sobre o item recém criado denominado CCSM.
- Clique em *Off Line Characterization*.

Nesta janela o usuário deverá localizar os parâmetros B_ADDRESS1 até B_ADDRESS96.

- Clique em B_ADDRESS1 e digite no campo em branco o endereço MODBUS correspondente, ou seja, 10001.
- Após digitar este valor clique no botão *End Edit*.
- Repita o mesmo procedimento para B_ADDRESS2 até B_ADDRESS8, adicionando os endereços MODBUS das chaves *ON/OFF* conforme a Tabela 5.1 estabelecida.

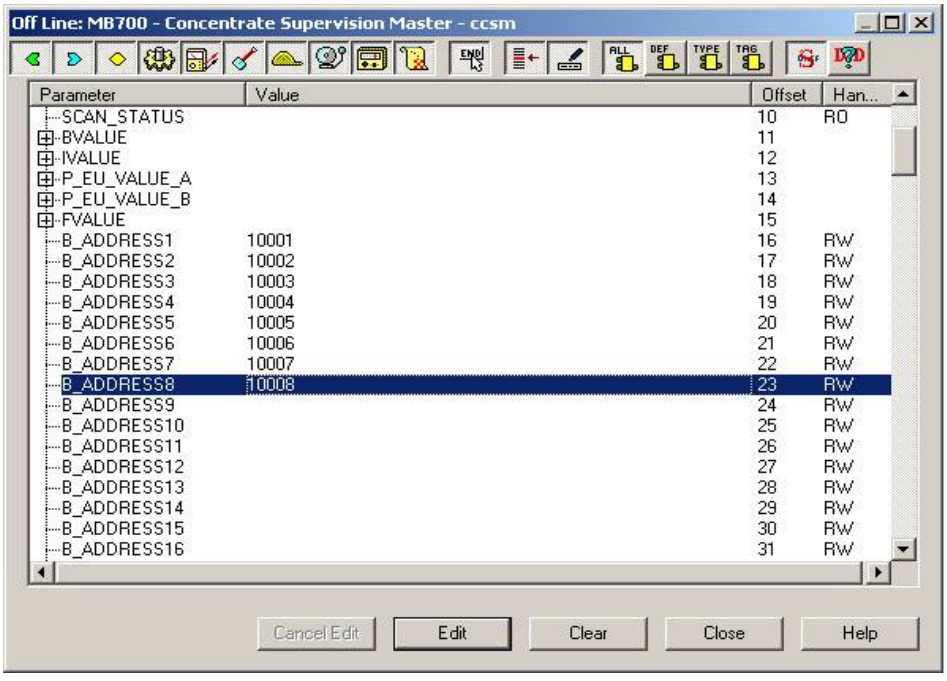


Figura 5.6 - Configurando Bloco CCSM (1)

O usuário deverá agora configurar as entradas dos transmissores. Na janela representada pela Figura 5.6,, procure pelos parâmetros P_EU_ADDRESS_A1 até P_EU_ADDRESS_A28. Clique sobre o parâmetro P_EU_ADDRESS_A1. O usuário deverá configurar os seguintes parâmetros:

⇨ FROM_EU_100%
Informe o valor máximo da variável de entrada lida no transmissor 1.

⇨ FROM_EU_0%
Informe o valor mínimo da variável de entrada lida no transmissor 1.

⇨ TO_EU_100%
Informe o valor máximo do parâmetro P_EU_VALUE_A[0].

⇨ TO_EU_0%
Informe o valor mínimo do parâmetro P_EU_VALUE_A[0].

⇨ DATATYPE
Escolha o formato da variável MODBUS lida do dispositivo escravo. Para maiores detalhes sobre os tipos disponíveis de dados, veja a tabela no Apêndice C.

⇨ MODBUS_ADDRESS_OF_VALUE.
Digite o endereço 30001, onde está o valor lido do transmissor 1. Conforme o tipo de dado utilizado na leitura desta variável, o transmissor ocupará um ou dois endereços. Apenas o primeiro endereço deverá ser informado

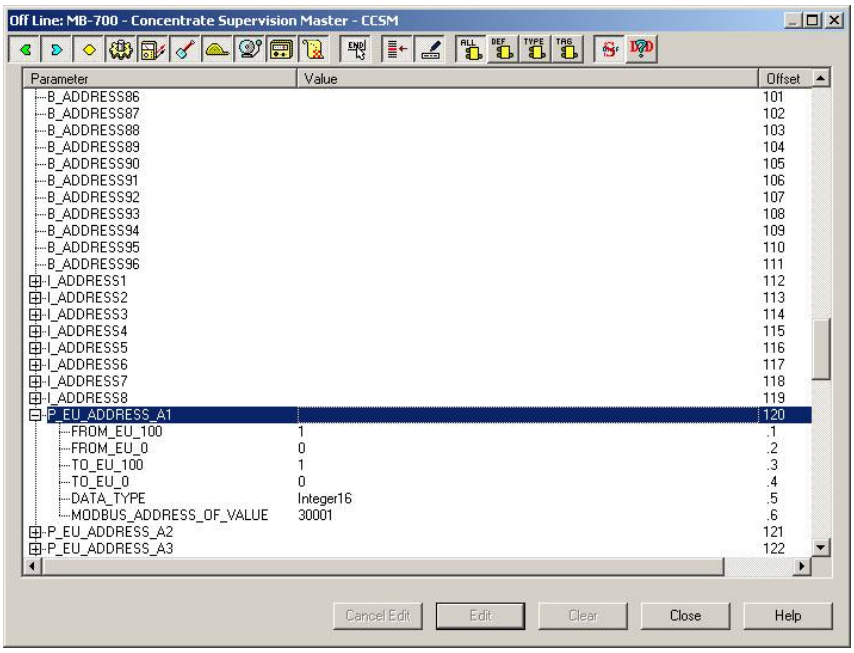


Figura 5.7 - Configurando Bloco CCSM (2)

Terminada a configuração, a seguinte tela deve aparecer (ver Figura 5.8):

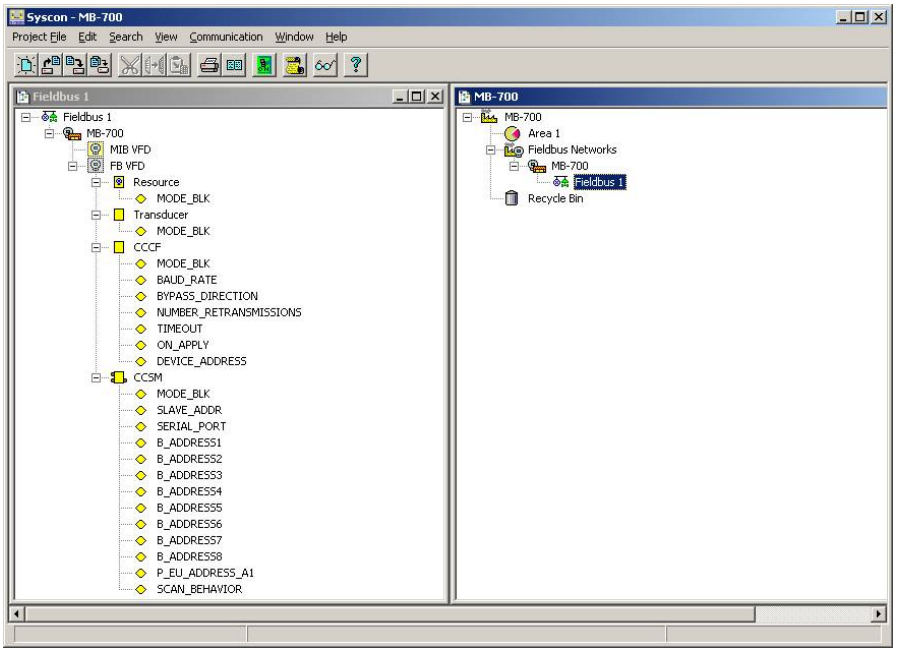


Figura 5.8 – Configuração de Blocos Concluída

3) MB-700 trabalhando como *Peer-to-peer*

Para o MB-700 trabalhar como *Peer-to-peer* entre dispositivos MODBUS na mesma rede serial, é necessário configurar, além dos Blocos *Resource* e CCCF, o Bloco CCCM.

Bloco CCCM

Insira este bloco à configuração.

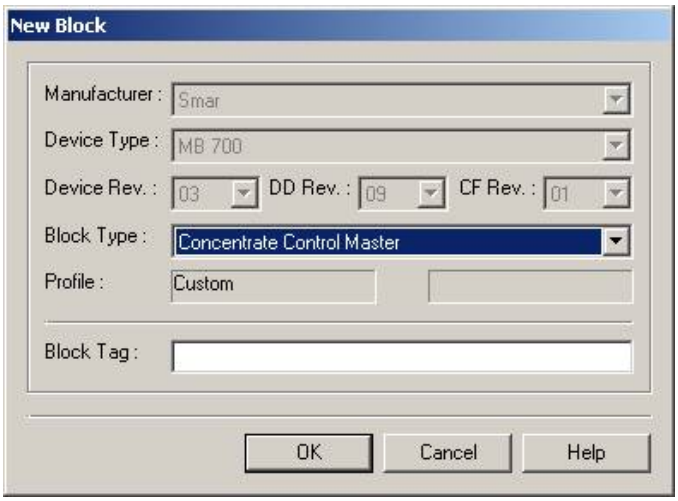


Figura 5.9 – Inserindo Bloco CCCM

Após inserir este novo bloco, a janela *Fieldbus* deve ser semelhante à Figura 5.10 a seguir:

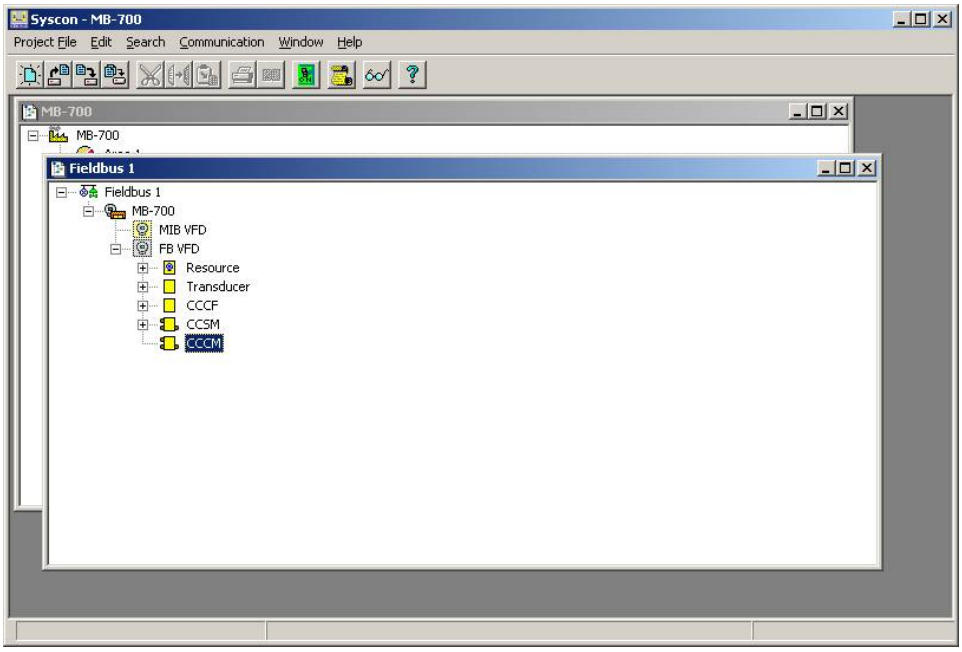


Figura 5.10 – Bloco CCCM Inserido no Projeto

Altere MODE_BLK.Target para “Auto”.

Inicialmente será associado as variáveis modbus dos escravos que se deseja ler e escrever valores. Por exemplo, considerando o exemplo da Figura 5.1 onde os equipamentos escravos modbus estão na porta serial do MB-700, e deseja-se transferir uma variável do escravo 1 para o escravo 2. Então, associa-se as variáveis de Leitura nos parâmetros OUT_xx e as variáveis de Escrita no parâmetro IN_xx do bloco CCCM.

Localize EU_ADDRESS_OUT1. Clique sobre este parâmetro e configure:
⇨ FROM_EU_100%
Informe o valor máximo da variável de entrada lida no transmissor.

⇨ FROM_EU_0%
Informe o valor mínimo da variável de entrada lida no transmissor.

⇨ TO_EU_100%
Informe o valor máximo da variável a ser monitorada.

⇨ TO_EU_0%
Informe o valor mínimo da variável a ser monitorada.

⇨ DATATYPE
Escolha o formato para o dado desejado. Para maiores detalhes sobre os tipos de dados disponíveis, veja a tabela no Apêndice C.

⇨ PORT_NUMBER
Selecione P1. A porta P1 deve ser selecionada, pois a comunicação é feita através da porta serial do MB-700.

⇨ SLAVE_ADDRESS
Insira o número do escravo Modbus.

⇨ MODBUS_ADDRESS_OF_VALUE
Insira o endereço Modbus da variável de leitura do escravo 1.

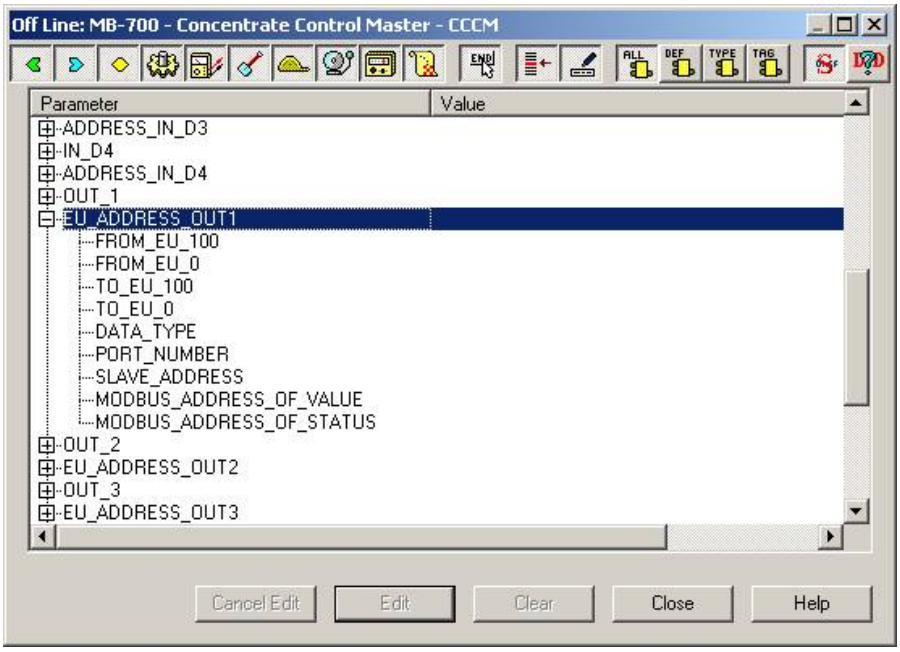


Figura 5.11 – Configurando Bloco CCCM

Para configurar as variáveis de escrita, localize o parâmetro EU_ADDRESS_IN1. Informe os parâmetros requeridos. Eles são semelhantes aos parâmetros configurados em EU_ADDRESS_OUT1. Porém SLAVE_ADDRESS deve ser igual ao endereço do segundo escravo na rede Modbus.

No parâmetro MODBUS_ADDRESS_OF_VALUE insira o endereço Modbus da variável de escrita do escravo 2.

Agora falta relacionar as variáveis de entrada e saída para indicar onde são os pontos de leitura e onde são os pontos de escrita do "*Peer-to-peer*". Caso o usuário desejar o relacionamento através de "link" deve proceder da seguinte forma:

Configure o Bloco CCCM no diagrama lógico:
- No SYSCON, clique com o botão direito do mouse em *Area1*;
- Selecione *New Process Cell*;
- Em *Process Cell 1*, clique com o botão direito do mouse em *Expand*;
- Na nova janela, clique com o botão direito em *Process Cell 1*, selecione *New Control Module*;
- Em *Control Module 1*, clique com o botão direito do mouse e selecione *Strategy*. Redimensione estas janelas conforme a conveniência. Ao final o usuário deverá ter 4 janelas abertas:

- ✓ Uma janela principal com o nome do arquivo;
- ✓ Uma janela *Process Cell;*
- ✓ Uma janela *Fieldbus; e*
- ✓ Uma janela *Control Module.*

- Na janela *Fieldbus*, arraste o ícone CCCM para dentro da janela *Control Module*. O Syscon insere um Bloco CCCM nesta janela.
- Clique no botão de link. Com o botão direito do mouse clique no Bloco CCCM na janela *Control Module*.
- Selecione **OUT_1**. Crie um link com a entrada **IN_1** desse mesmo Bloco CCCM. Com isso o SYSCON efetua o link entre OUT_1 (que significa o ponto de leitura de dados) e o IN_1 (que significa o ponto de escrita de dados).

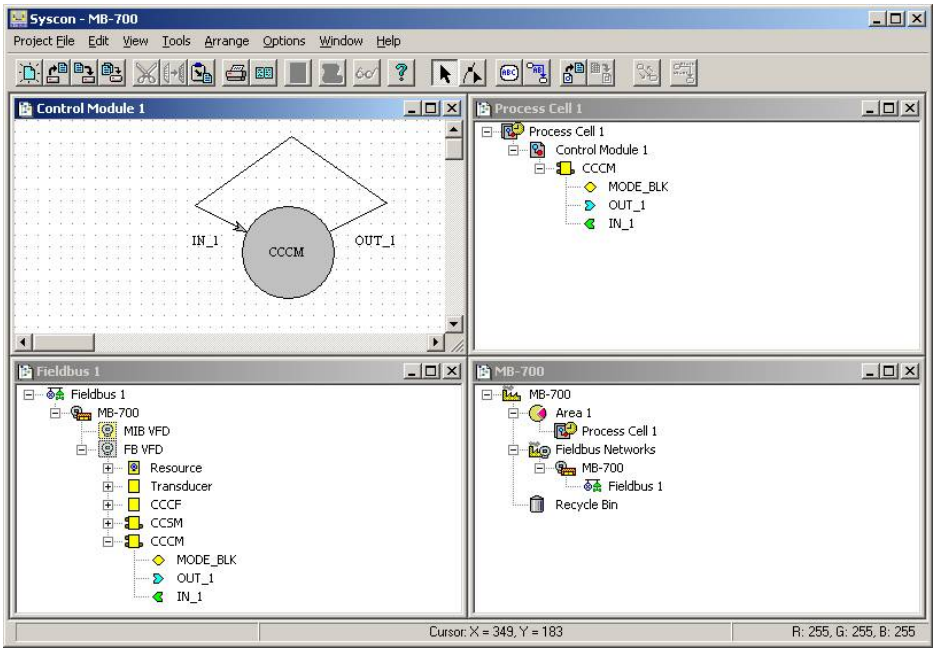


Figura 5.12 – Exemplo de Configuração de um Link OUT_1 para IN_1(Comunicacão Ponto a Ponto Modbus)

D - Download da Configuração

Antes de iniciar a comunicação, o usuário deve realizar os seguintes procedimentos:

- Na janela principal do projeto aberto no SYSCON, clicar com o botão direito do mouse sobre o primeiro item desta janela. No exemplo apresentado neste manual, o projeto foi chamado de MB-700;
- Selecionar "*Export Tags*". Uma janela será aberta pedindo que o usuário escolha a pasta onde os *Tags* será exportado e salve o arquivo "TagInfo.ini";
- Clique no botão *OK.*

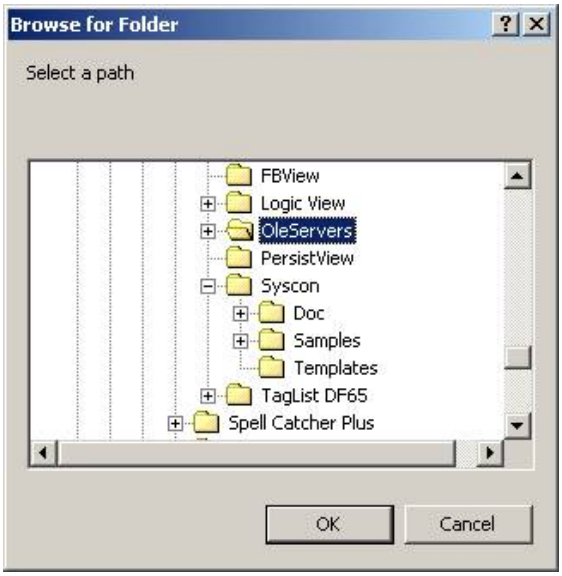


Figura 5.13 – Escolhendo a Pasta para Export Tags

No botão *On*, inicie a comunicação com o MB-700.



- Caso a conexão não for estabelecida, clique com o botão da direita sobre B*ridge*, neste projeto chamada de MB-700, e selecione *Attributes*.
- No campo *Device ID*, verificar se o dispositivo é reconhecido. Selecionar o número correspondente ao dispositivo. Veja o número de série localizado no cartão MB-700.
- Faça agora o download da configuração estabelecida. Clique em *Fieldbus1* ou em (MB-700) *bridge* com o botão direito do mouse. Selecione *Download.* A configuração será enviada ao MB-700
- Após ter sido enviada a configuração, o usuário deverá alterar o parâmetro ON_APPLY do Bloco CCCF para "Apply". Isto atualizará e habilitará a configuração.

Configuração *On Line*

É possível configurar os Blocos do MB-700 *On Line*, isto é, com o dispositivo já conectado à rede e funcionando, basta que o usuário clique sobre o bloco com o botão direito do mouse e selecione "*On Line Characterization*".

IMPORTANTE
Para efetivar cada nova configuração, o usuário deve sempre alterar o parâmetro ON_APPLY para "Apply", caso contrário esta configuração não será atualizada no MB-700. O usuário também deve verificar quais parâmetros podem ser configurados e alterados no modo "auto", pois alguns parâmetros só podem ser alterados no modo OOS (*Out of Service*).

MB-700 como Mestre TCP/IP

Descrição

Neste cenário, o MB-700 não possui função concentradora. Portanto os dois únicos blocos a serem configurados são os Blocos CCCF e *RESOURCE*. Para configurá-los veja o item Configuração *Off Line*.

Funções desempenhadas pelo MB-700 neste cenário:

- ✓ *Bypass* (Serial to TCP): O MB-700 funciona como um conversor de protocolos. Nesta função ele converte uma mensagem serial Modbus RTU para TCP/IP e TCP/IP para Modbus RTU.

- ✓ *Peer-to-Peer*: Dois escravos Modbus trocam dados entre si. A função "*peer-to-peer*" possibilita esta função. Os escravos podem estar em redes Modbus diferentes e mesmo assim podem trocar dados entre si.

1 - MB-700 trabalhando como *Bypass* (Serial para TCP)

Vários MB-700 estão conectados em uma rede local ethernet. Em cada MB-700 são conectados vários dispositivos escravos. O usuário pode configurar qualquer um destes PLCs através de uma mesma estação de trabalho. Dois MB-700 são conectados a suas respectivas estações de trabalho através da porta serial EIA-232 de cada MB-700. As portas ethernet conectam estes MB-700 a rede corporativa. Conectado a esta rede está um terceiro MB-700 e nele são conectados os dispositivos MODBUS escravos. O MB-700 atua como mestre TCP/IP estabelecendo uma ponte entre as estações de trabalho, a rede local e a rede Modbus. A grande vantagem proporcionada pelo MB-700 nesta configuração é permitir a conexão de um escravo Modbus com vários mestres seriais. Isso possibilita que um escravo seja acessível a diversos pontos da rede ethernet através do MB-700.

NOTA
Para este cenário, o MB-700 não suporta os blocos CCSM e CCCM.

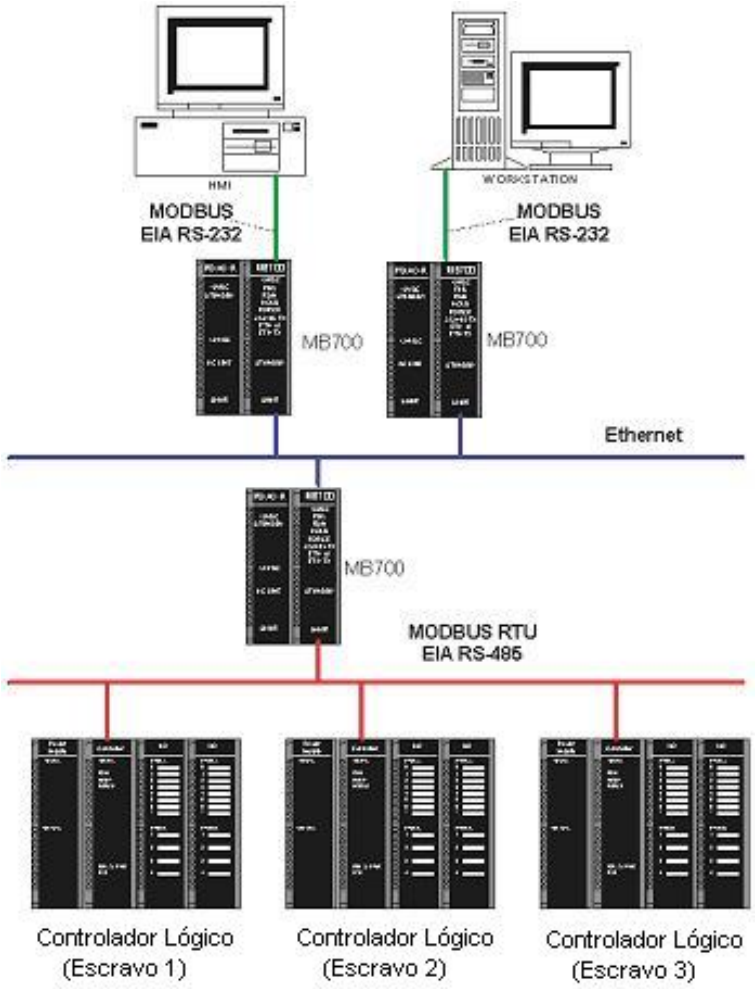


Figura 5.14 – Exemplo Arquitetura do MB-700 trabalhando como Bypass Serial para TCP

Configuração

O Bloco *RESOURCE* deve ter o parâmetro MODE_BLK colocado em “Auto”.

Os procedimentos para configuração do Bloco CCCF são equivalentes à aplicação anterior, exceto pelo parâmetro BYPASS_DIRECTION que deve ser selecionado para “Serial To TCP”.

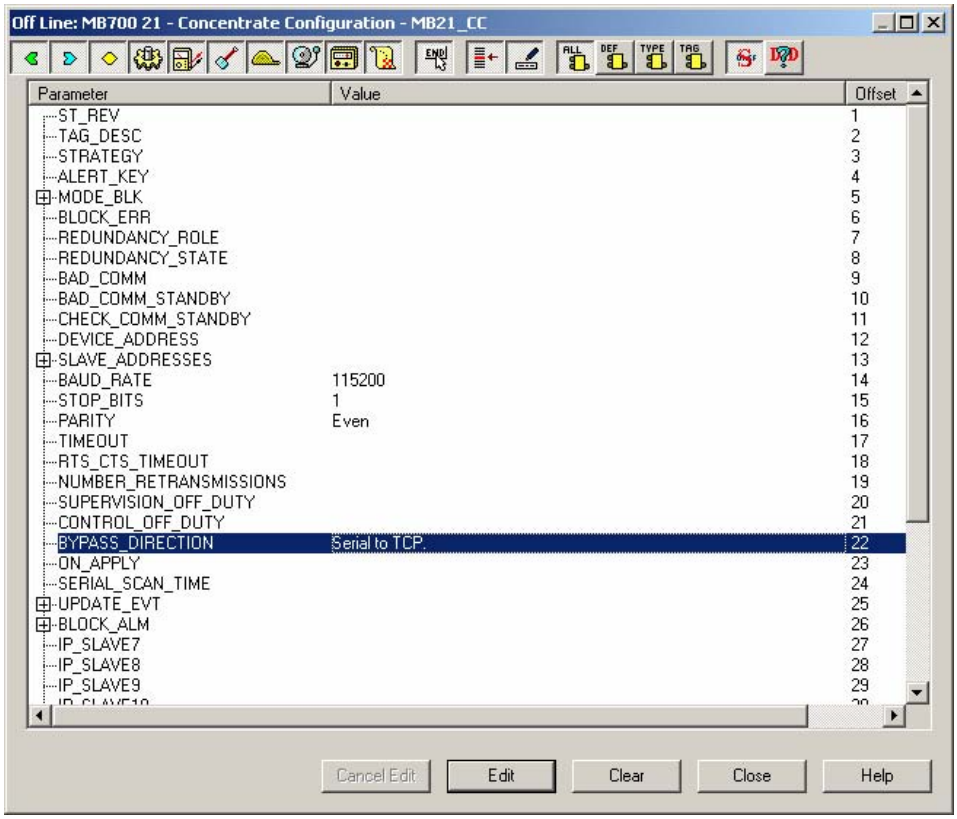


Figura 5.15 – Configurando o Parâmeto BYPASS_DIRECTION no Bloco CCCF

A configuração dos Escravos Modbus no Bloco CCCF segue a seguinte forma:

- Caso a configuração contenha apenas um escravo Modbus para um determinado endereço IP, utiliza-se o parâmetro SLAVE_ADRESSES. Neste caso, para cada IP_SLAVE tem-se um único MODBUS_ADDRESS_SLAVE correspondente.

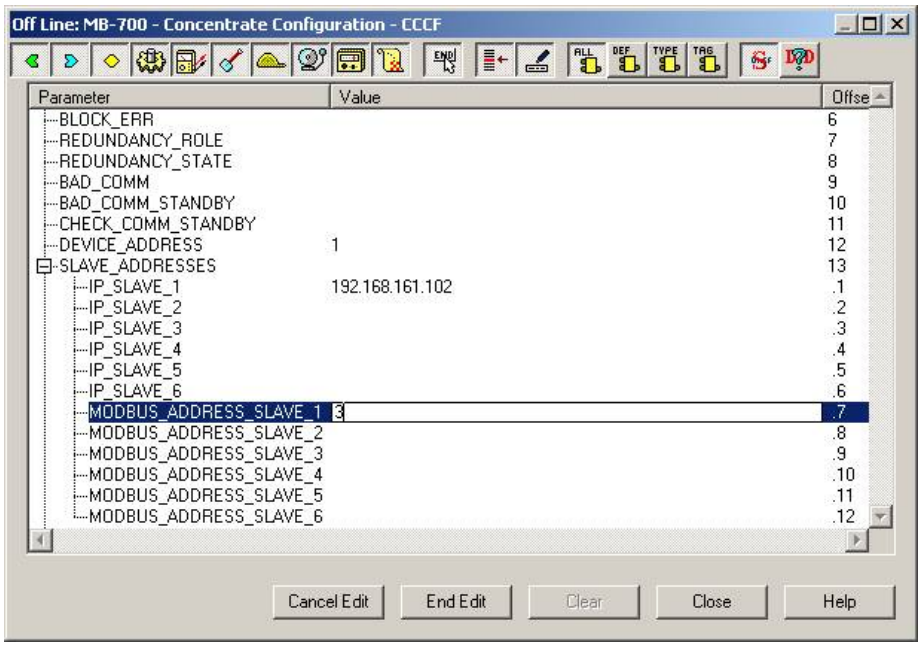


Figura 5.16 – Configurando os Escravos Modbus no Bloco CCCF

- Caso a configuração tenha mais de um escravo Modbus para um mesmo endereço IP, utiliza-se os parâmetros IP_SLAVE_x e DEVICE_IDS_IP_x. Neste caso para cada IP_SLAVE pode-se configurar até 32 endereços Modbus de escravos.

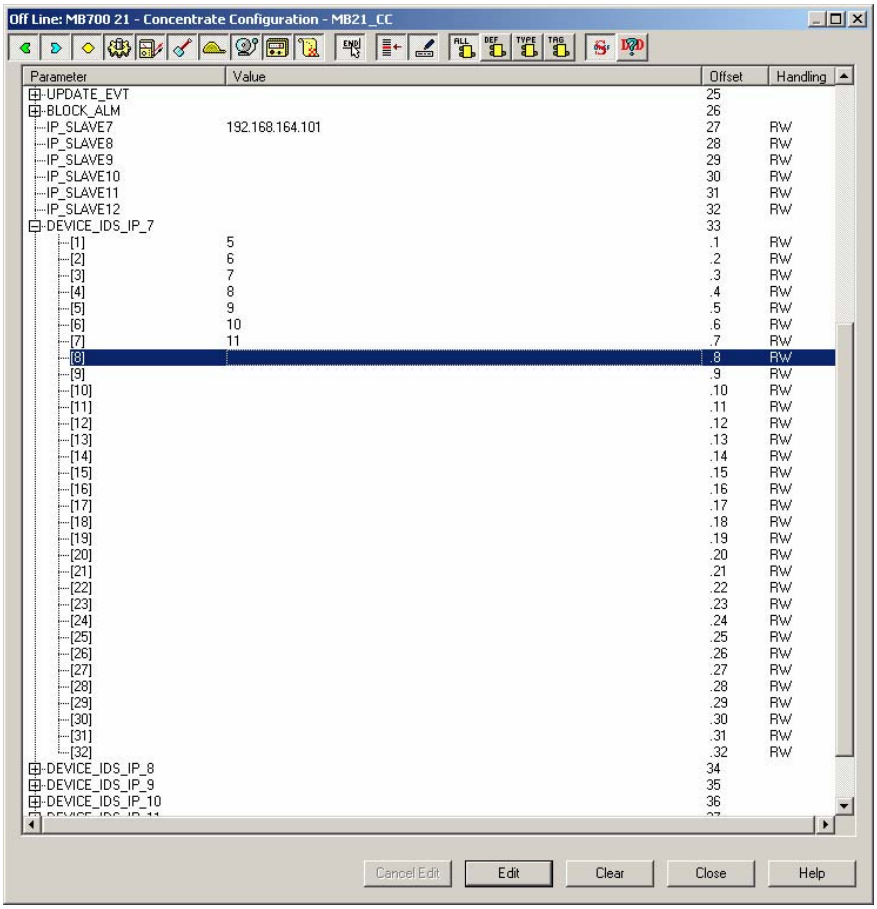
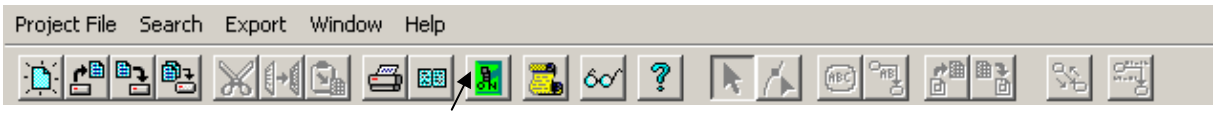


Figura 5.17 – Configurando os IPs dos Escravos Modbus no Bloco CCCF

No Bloco CCCF, mude o parâmetro ON_APPLY para “Apply”.

No botão *ON*, estabeleça a comunicação com o MB-700.



Faça o download da configuração. Após este passo ter sido realizado, a comunicação é estabelecida e o MB-700 está *On Line*.

2 - MB-700 trabalhando como *Peer-to-peer*

Para trabalhar como *Peer-to-peer* no TCP/IP, deve-se seguir os passos definidos na Configuração *Off Line* no item MB-700 trabalhando como *Peer-to-peer* Serial e, além disso, os seguintes parâmetros deverão ser configurados:

- No Bloco CCCF, o parâmetro SLAVE_ADDRESSES e/ou o parâmetro IP_SLAVE deverá ser configurado com o IP e o endereço Modbus dos escravos TCP/IP (como descrito no item anterior) até o limite de 12 endereços TCP/IP.

- No Bloco CCCM, no parâmetro EU_ADDRESS_XXXX.PORT_NUMBER, selecionar “Ethernet” indicando que para este ponto (XXXX) a comunicação será feita através do TCP/IP. (Onde XXXX pode ser IN_1...IN_4, IN_D1...IN_D4, OUT_1...OUT_4, OUT_D1...OUT_D4).

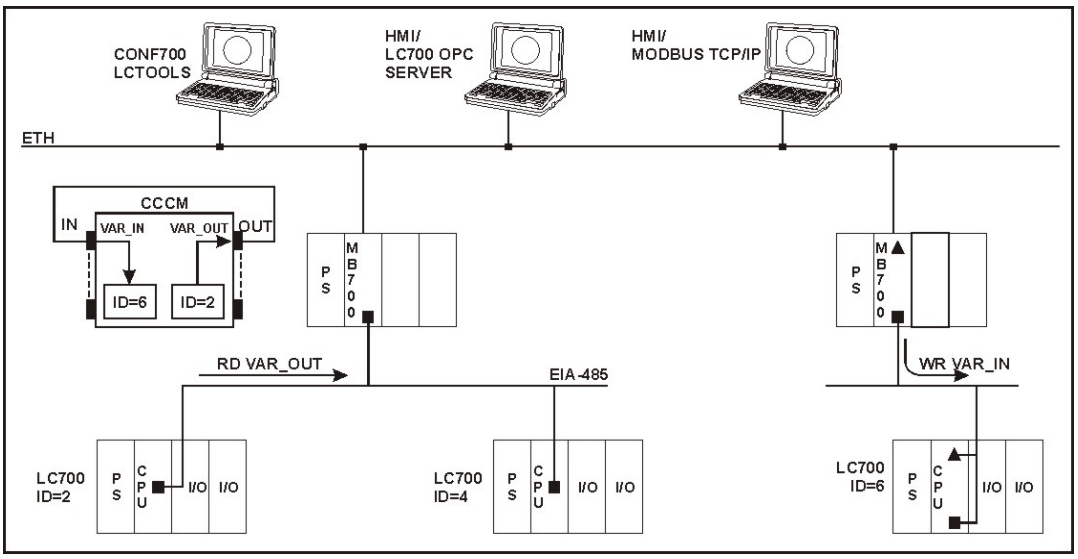


Figura 5.18 – MB-700 trabalhando como Peer-to-peer Serial e TCP/IP

MB-700 como Escravo Modbus TCP

1- MB-700 trabalhando como Concentradora

O MB-700 pode trabalhar como um concentrador Modbus Serial / TCP e, também como Escravo TCP dos dados dinâmicos concentrados dentro dele (Bloco CCSM).

O endereço do equipamento escravo (ID Modbus) é definido no parâmetro DEVICE_ADDRESS do Bloco *Concentrate Configuration* (CCCF).

O endereço Modbus do parâmetro a ser monitorado é fixo para cada Bloco CCSM e dependente da instância do bloco (indicado no parâmetro *Strategy*).

Para atender a redundância, o MB-700 deve responder como Escravo Modbus tanto em Ativo quanto em *Standby*.

Cálculo do Endereço Modbus

Cada bloco tem uma faixa fixa de endereços Modbus, definida seguindo a fórmula:

Modbus_Addr = Modbus_start_addr + (offset * n)

Onde:
n – parametro STRATEGY do Bloco CCSM (indica a instância do bloco).

Por exemplo, para a seguinte configuração :

- MB-700
- CCSM1
- CCSM2
- CCSM3
- CCSM4

O endereço Modbus do parâmetro BVALUE[1] do Bloco CCSM4 seria:

Modbus_start_addr – BVALUE[1] = 1001
Offset Bloco CCSM = 200
CCSM.Strategy = 3 (quarto bloco SM na configuração)
Modbus_Addr = 1001 + (200 * 3)
Modbus_Addr = 1601

Bloco CCSM

Index	Parâmetro	DataType	Modbus Address (Strategy = x) Offset_SM = 200 * x x = 0 ~ 24	Modbus Address Strategy= 0 Offset_SM = 0	Modbus Address Strategy= 1 Offset_SM = 200	Modbus Address Strategy = 2 Offset_SM = 400
10	SCAN_STATUS	UInteger16	42001 + Offset_SM	42001	42201	42401
11.1	BVALUE[1]	Bool	2001 + Offset_SM	2001	2201	2401
11.2	BVALUE[2]	Bool	2002 + Offset_SM	2002	2202	2402
11.96	BVALUE[96]	Bool	2096 + Offset_SM	2096	2296	2496
12.1	IVALUE[1]	Integer32	42002-42003 + Offset_SM	42002-42003	42202-42203	42402-42403
12.2	IVALUE[2]	Integer32	42004-42005 + Offset_SM	42004-42005	42204-42205	42404-42405
12.8	IVALUE[8]	Integer32	42016-42017 + Offset_SM	42016-42017	42216-42217	42416-42417
13.1	P_EU_VALUE_A[1]	Float	42018-42019 + Offset_SM	42018-42019	42218-42219	42418-42419
13.2	P_EU_VALUE_A[2]	Float	42020-42021 + Offset_SM	42020-42021	42220-42221	42420-42421
13.28	P_EU_VALUE_A[28]	Float	42072-42073 + Offset_SM	42072-42073	42272-42273	42472-42473
14.1	P_EU_VALUE_B[1]	Float	42074-42075 + Offset_SM	42074-42075	42274-42275	42474-42475
14.2	P_EU_VALUE_B[2]	Float	42076-42077 + Offset_SM	42076-42077	42276-42277	42476-42477
14.28	P_EU_VALUE_B[28]	Float	42128-42129 + Offset_SM	42128-42129	42328-42329	42528-42529
15.1	FVALUE[1]	Float	42130-42131 + Offset_SM	42130-42131	42330-42331	42530-42531
15.2	FVALUE[2]	Float	42132-42133 + Offset_SM	42132-42133	42332-42333	42532-42533
15.16	FVALUE[16]	Float	42160-42161 + Offset_SM	42160-42161	42360-42361	42560-42161

Tabela 5.2 – Tabela de Endereços Modbus do MB-700 trabalhando como Escravo Modbus

Usando o Data Logger

Descrição

O MB-700 pode armazenar os dados de *Log* de um Bloco *Data Logger* do LC700. Então, na leitura normal do Bloco *Concentrate Data Logger*, ele verifica se existem dados armazenados no Escravo correspondente. Caso existam dados armazenados no escravo, ele faz a leitura desses dados e limpa o *buffer* de *log* do escravo.

Configuração

Primeiramente, deve ter sido criado o Bloco FIFO no LC700.
Na configuração do Syscon podem ser incluídos tantos Blocos "*Concentrate Data Logger*" quantos forem necessários. Para inclusão do Bloco CCDL na configuração, basta proceder como nos exemplos anteriores.

Obs.: A quantidade de Blocos *Concentrate Data Logger* depende do tamanho da FIFO definida no LC700 (FIFO *Size*). Cada Bloco CCDL comporta até 280 amostras de dados. Por exemplo, se o

tamanho da FIFO definido for de 1000 registros e o tipo de dado for Inteiro (Porcentagem) então vão ser necessários: 1000 / 280 = 4 Blocos CCDL.
Caso o tipo de dados definido for REAL, utilizando os mesmos 1000 registros e sabendo que um REAL precisa de 2 registros para cada amostra, então o cálculo seria feito da seguinte forma:
500/ 280 = 2 Blocos CCDL

O Bloco *Data Logger* trabalha junto com os Blocos de Supervisão. Portanto deverá haver blocos de supervisão apontando para o mesmo endereço do escravo, para que o Bloco CCDL funcione corretamente.

Um exemplo de configuração é mostrado na Figura 5.19 a seguir. Considerando uma configuração onde o LC700 tem os seguintes parâmetros:

- SLAVE_ADDR = 1 (LC700 Address)
- STOP_SCAN_ADDRESS = 2009 (FIFO_LOAD)
- CLEAR_LOG_ADDRESS = 2011 (FIFO_CLEAR)
- LOGGER_ADDRESS = 42506 (FIFO_CTW)
- NEED_SCAN_ADDRESS = 2 (FIFO_EMPTY)
- DATATYPE BLOCO FIFO = REAL

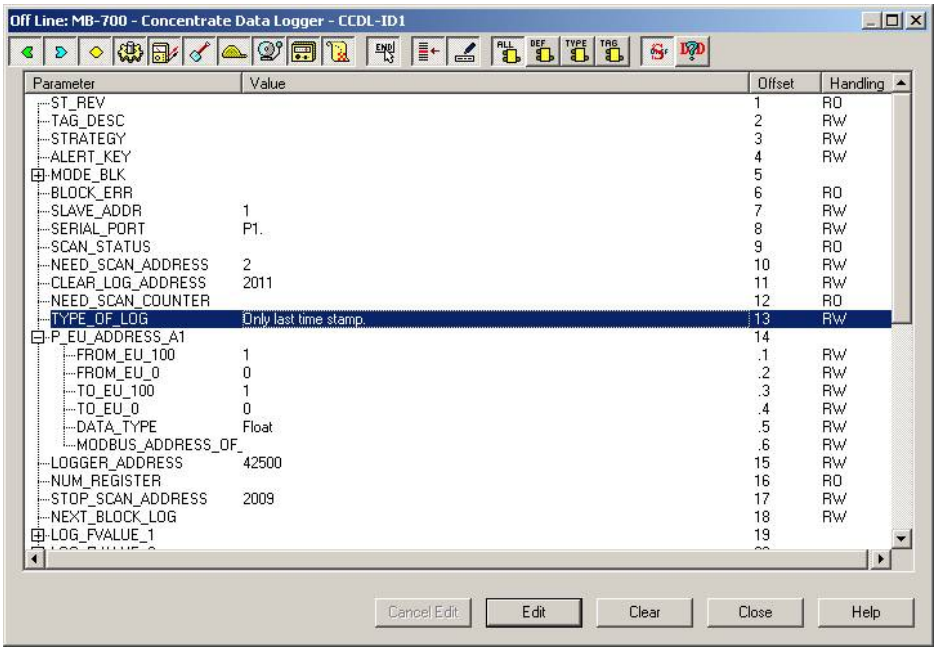


Figura 5.19 – Configurando o Bloco CCDL

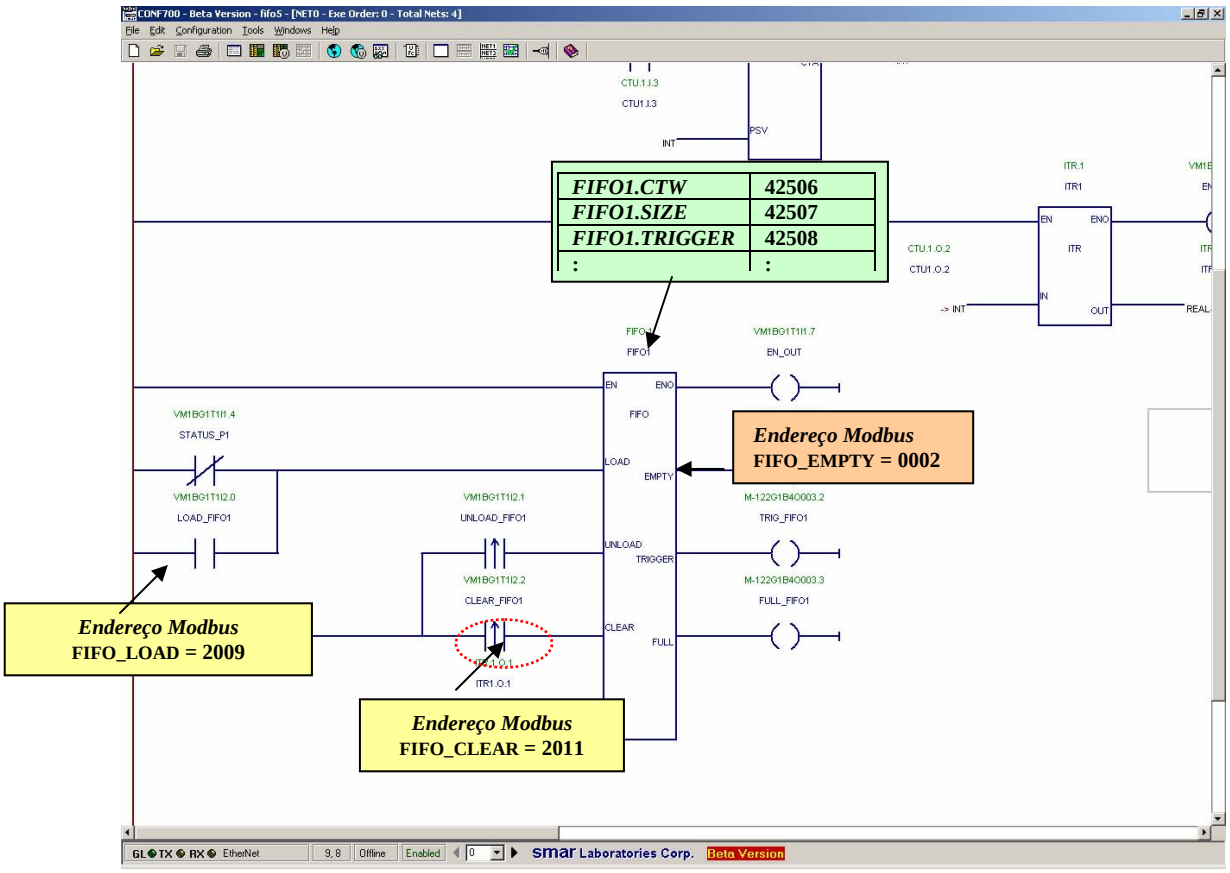


Figura 5.20 – Interação do CCDL do MB-700 com LC700

Blocos *DATA LOGGER* em Cascata

Caso a quantidade de dados armazenados for maior que a suportada pelo Bloco CCDL, o usuário poderá colocar Blocos CCDL em cascata, devendo apenas repetir a mesma configuração para todos os Blocos *Data Logger* instanciados. Desta forma, o **MB-700** começará a usar os blocos pela ordem de instanciação indicando qual é o próximo bloco armazenado através do parâmetro NEXT_BLOCK_LOG.

Capítulo 6

CENÁRIOS

Os vários cenários de aplicação do MB-700 apresentados a seguir utilizam o LC700 da SMAR como exemplo de dispositivo Modbus RTU escravo, porém o mesmo se aplica a qualquer dispositivo Modbus RTU escravo.

Múltiplos Mestres TCP/IP Comunicando Diretamente com vários LC700s, inclusive via Rádio

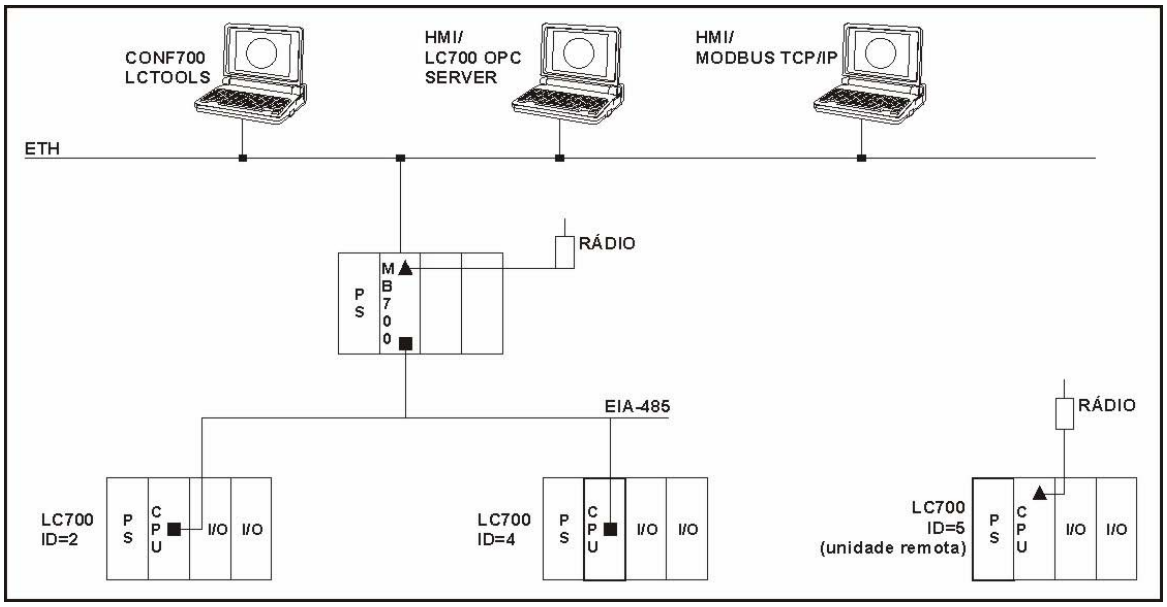


Figura 6.1 – Exemplo de Aplicação do MB-700 como Gateway MODBUS (Mestre Serial)

Nesta aplicação o MB-700 suporta as seguintes funções simultaneamente:

- ✓ CONF700 faz download/upload via TCP/IP inclusive remotamente.
- ✓ Supervisão do LC700 via OPC e/ou driver Modbus TCP/IP.
- ✓ Download do firmware do LC700 via TCP/IP inclusive remotamente.

O MB-700 é conectado à rede corporativa através de sua porta ethernet. Na rede estão conectados as estações de trabalho para configuração e monitoração dos dispositivos escravos. As portas seriais EIA-232 e EIA-485 permitem que 3 LC700s sejam conectados ao MB-700. Dois controladores LC700s estão ligados ao barramento da porta EIA-485, e um terceiro é conectado através de um link de rádio. Este terceiro LC700 é uma unidade remota pois se localiza distante do MB-700 mestre.

O primeiro LC700 tem Modbus ID= 2, o segundo Modbus ID= 4 e o LC700 remoto tem Modbus ID=5.

O MB-700 possui um endereço IP dentro da rede ethernet, desta forma as ferramentas de configuração e supervisão "enxergam" o MB-700 dentro da rede. No Bloco CCCF, os seguintes parâmetros de comunicação devem ser configurados: TIMEOUT, BAUDRATE, BYPASS_DIRECTION (TCP to Serial), PARITY and NUMBER_RETRANSMISSIONS.

Múltiplos Mestres TCP/IP Comunicando Diretamente com vários LC700s e "Peer-to-peer" entre LC700s na mesma Rede Modbus RTU

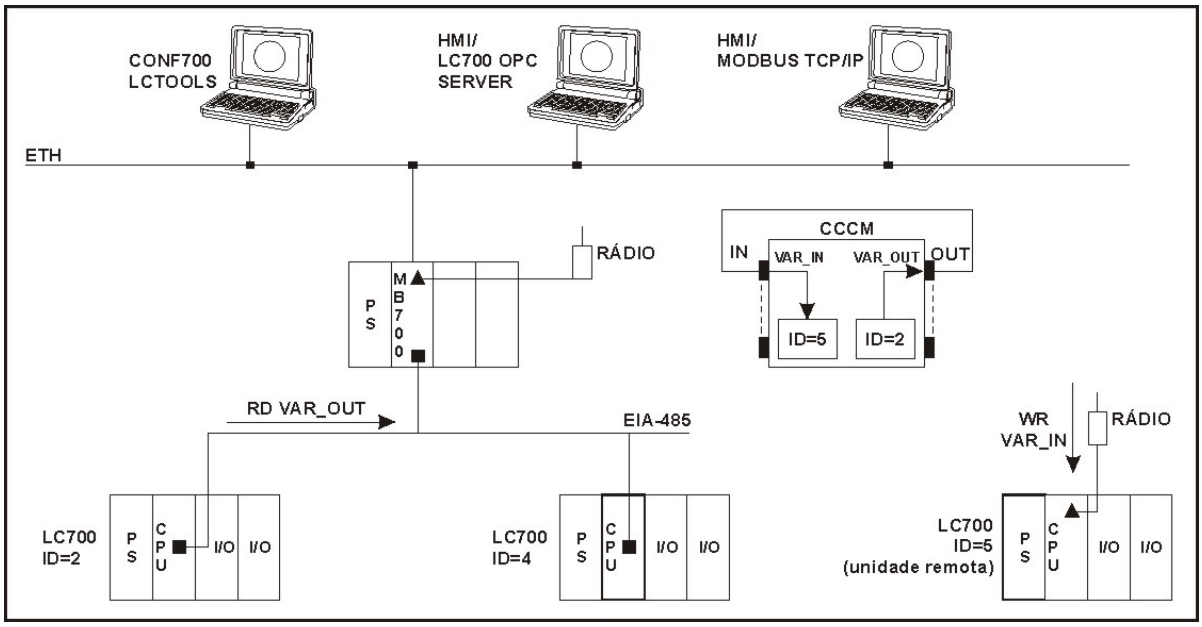


Figura 6.2 – Exemplo de Aplicação do MB-700 como Gateway MODBUS e Ligação de Equipamentos Ponto a Ponto (Mestre Serial)

Nesta aplicação, o MB-700 suporta as seguintes funções simultaneamente:

- ✓ CONF700 faz download/upload via TCP/IP inclusive remotamente.
- ✓ Supervisão do LC700 via OPC e/ou driver Modbus TCP/IP.
- ✓ Download do firmware do LC700 via TCP/IP inclusive remotamente
- ✓ Comunicação "*Peer-to-peer*" entre LC700s na mesma rede Modbus RTU, inclusive com unidades remotas (Do LC700 ID=2 para o LC700 ID= 5).

Este cenário é semelhante ao anterior, porém a função "*Peer-to-peer*" foi implementada. Há comunicação *Peer-to-peer* entre os LC700s da mesma rede Modbus RTU, inclusive com unidades remotas. Por exemplo, do LC700 (Modbus ID=2) para o LC700 (Modbus ID=5).

Esta funcionalidade é adicionada através do Bloco CCCM. Neste bloco, o parâmetro PORT_NUMBER é configurado. A comunicação entre os escravos é realizada através das portas seriais EIA-232 e EIA-485 (porta P1 do MB-700).

A comunicação *Peer-to-peer* entre os LC700s com Modbus IDs 2 e 5, respectivamente, requer um Bloco CCCM adicionado. Neste bloco informam-se os endereços Modbus das variáveis que devem ser lidas e os endereços Modbus onde estas variáveis são escritas. A comunicação é feita dentro da rede Modbus RTU em meio físico serial EIA-485 e via rádio.

Assim, os escravos podem trocar dados diretamente entre eles, sem que a mensagem passe pelos mestres TCP/IP. O MB-700 possibilta esta troca, conectando os dois pontos da rede Modbus. Dentro de cada Bloco CCCM, até 8 links *Peer-to-peer* podem ser estabelecidos, pois ele possui 4 entradas digitais, 4 entradas analógicas, 4 saídas digitais e 4 saídas analógicas.

Múltiplos Mestres TCP/IP Comunicando Diretamente com vários LC700s e "Peer-to-peer" entre LC700s de diferentes Redes Modbus RTU

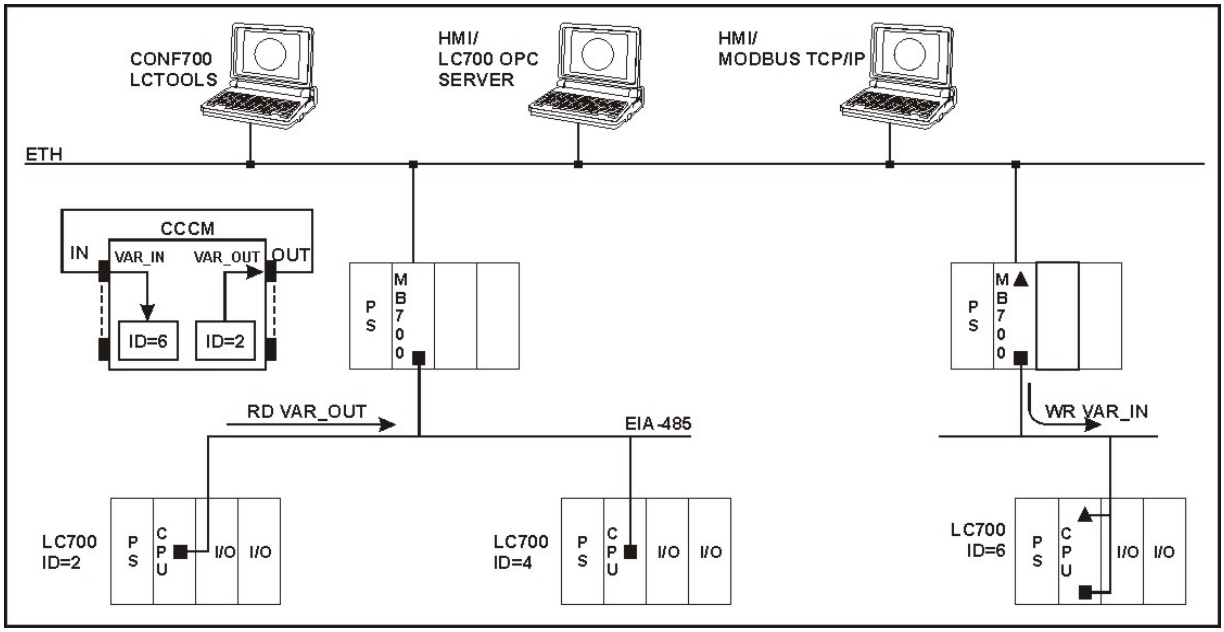


Figura 6.3 – Exemplo de Aplicação do MB-700 ligando Equipamentos Ponto a Ponto Local e Remoto (atraves do TCP) e também Gateway Modbus

Nesta aplicação, o MB-700 suporta as seguintes funções simultaneamente:

- ✓ CONF700 faz download/upload via TCP/IP inclusive remotamente.
- ✓ Supervisão do LC700 via OPC e/ou driver Modbus TCP/IP.
- ✓ Download do firmware do LC700 via TCP/IP, inclusive remotamente.
- ✓ Comunicação "*Peer-to-peer*" entre LC700's de diferentes redes Modbus RTU (do LC700 ID=2 e para o LC700 ID=6).

Diferentemente dos cenários anteriores, tem-se um segundo MB-700 conectado à rede Etnernet (esta aplicação também é válida a mais de dois MB-700). A comunicação "*Peer-to-peer*" é estabelecida entre o escravo de ID= 2 e o escravo de ID= 6. Como ambos estão conectados em duas redes Modbus RTU diferentes, tem-se dois MB-700 que realizam a conexão destes escravos com a rede ethernet. O MB-700 onde se conecta o escravo de ID= 2 possui um endereço IP, da mesma forma que o segundo MB-700 também possui um endereço IP. Somente um Bloco CCCM é necessário para realizar a comunicação "*Peer-to-peer*".

Para cada MB-700 deverá ser configurado um Bloco CCCF para estabelecer os parâmetros de comunicação. O *bypass* continua sendo no sentido "TCP to serial". No cenário, o Bloco CCCF tem papel crucial no endereçamento das redes ethernet e Modbus RTU. Nos parâmetros SLAVE_ADDRESSES são informados os endereços IP de cada MB-700, nos campos IP_SLAVE_1 e IP_SLAVE_2. Cada um desses tem um correspondente MODBUS_ADDRESS_SLAVE, onde são informados os IDs de cada escravo Modbus RTU. O endereço de onde é lida a variável tem a seguinte configuração apresentada a seguir.

A variável é lida no endereço:

IP do 1º MB-700	Modbus ID do escravo (LC700)	Endereço Modbus da variável
192.168.164.101	02	30001

A variável será escrita no endereço:

IP do 2º MB-700	Modbus ID do escravo (LC700)	Endereço Modbus
192.168.164.102	06	40001

Múltiplos Mestres TCP/IP Comunicando Diretamente com os LC700s, "Peer-to-peer" e Redundância de Rede

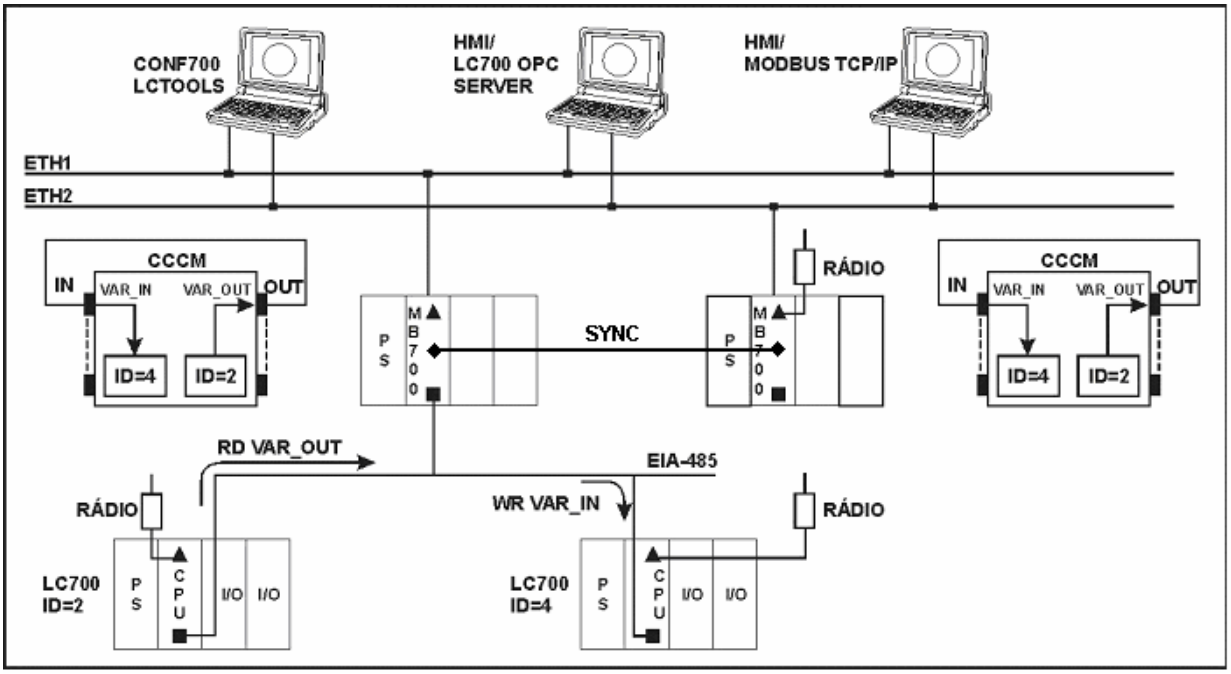


Figura 6.4 – Exemplo de Aplicação do MB-700 ligando Equipamentos Ponto a Ponto Local e com Redundancia de CPU

Nesta aplicação, o MB-700 suporta as seguintes funções simultaneamente:

- ✓CONF700 faz download/upload via TCP/IP, inclusive remotamente.
- ✓Supervisão do LC700 via OPC e/ou driver Modbus TCP/IP.
- ✓Download do firmware do LC700 via TCP/IP, inclusive remotamente.
- ✓Comunicação "*Peer-to-peer*" entre LC700's (do LC700 ID=2 e para o LC700 ID=4).
- ✓Redundância de redes de comunicação entre os múltiplos mestres TCP/IP e os LC700s, inclusive por diferentes meios físicos (EIA-485 e rádio).

A rede redundante permite que no caso de falha da rede principal, o segundo MB-700 assuma a supervisão e controle da comunicação entre os mestres TCP/IP e o escravos Modbus RTU.

O primeiro MB-700 conecta os mestres TCP/IP aos escravos Modbus RTU através da rede ethernet ETH1 e seu canal serial EIA-485. O segundo MB-700 faz a conexão dos mestres TCP/IP através da rede redundante ethernet ETH2 e o canal serial EIA-232 onde é conectado um link de rádio entre MB-700 e as CPUs dos escravos.

MB-700 como Concentrador e Supervisão via DFI OPC Server

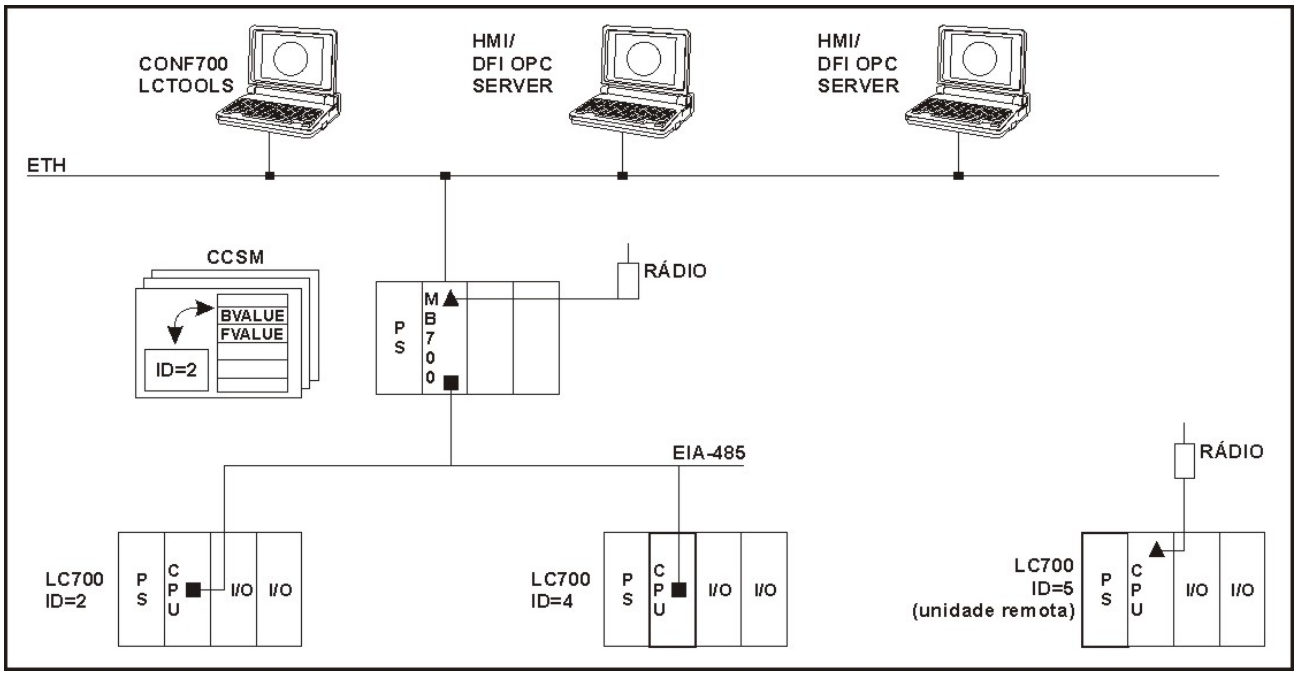


Figura 6.5 – Exemplo de Aplicação do MB-700 como Concentrador de Dados Serial e Gateway Modbus

Nesta aplicação, o MB-700 suporta as seguintes funções simultaneamente:

- ✓ CONF700 faz download/upload via TCP/IP inclusive remotamente.
- ✓ Download do firmware do LC700 via TCP/IP inclusive remotamente.
- ✓ Supervisão dos LC700's indiretamente através do MB-700, isto é, via DFI OPC *Server* supervisionam-se parâmetros do Bloco CCSM que são cópias das variáveis Modbus dos LC700's.
- ✓ Possibilidade de múltiplos DFI OPC *Server* ou trabalhar com um único *server* e de forma remota. O OPC.Client acessa os parâmetros do CCSM.

A configuração do Bloco CCSM permite que as variáveis Modbus sejam lidas e copiadas dentro da memória do MB-700. Para cada LC700 existe um Bloco CCSM.

No Bloco CCSM está informado o ID do LC700 (No cenário acima ID=2), e o MB-700 faz o scan das variáveis Modbus configuradas dentro do Bloco CCSM e as registra em sua memória como parâmetro do Bloco CCSM. Através do DFI OPC *Server*, estas variáveis são lidas por um OPC *Client*.

Desta forma, quando o usuário monitora uma variável, o acesso é mais rápido, pois o MB-700 armazena o valor desta variável Modbus em sua memória de parâmetros do Bloco CCSM.

MB-700 como Concentrator, Supervisão via DFI OPC Server e "Peer-to-peer"

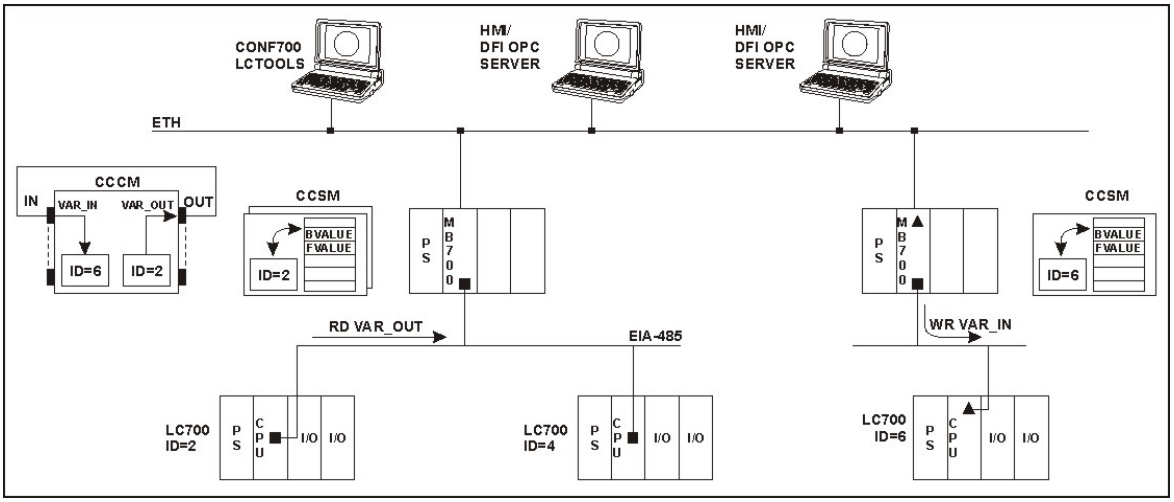


Figura 6.6 – Exemplo de Aplicação do MB-700 como Concentrador de Dados Serial e ligação de Escravos Modbus

Nesta aplicação, o MB-700 suporta as seguintes funções simultaneamente:

- ✓ CONF700 faz download/upload via TCP/IP inclusive remotamente.
- ✓ Download do firmware do LC700 via TCP/IP inclusive remotamente.
- ✓ Supervisão dos LC700s indiretamente através do MB-700, isto é, via DFI OPC *Server* supervisionam-se parâmetros do Bloco CCSM que são cópias das variáveis Modbus dos LC700s.
- ✓ Possibilidade de múltiplos DFI OPC *Server* ou trabalhar com um único *server* e de forma remota. O OPC.Client acessa os parâmetros do CCSM.
- ✓ Comunicação "*Peer-to-peer*" entre LC700s de diferentes redes Modbus RTU (do LC700 ID= 2 para o LC700 ID=6).

MB-700 como Concentrador, Supervisão via DFI OPC Server, "Peer-to-peer" e Redundância de Redes

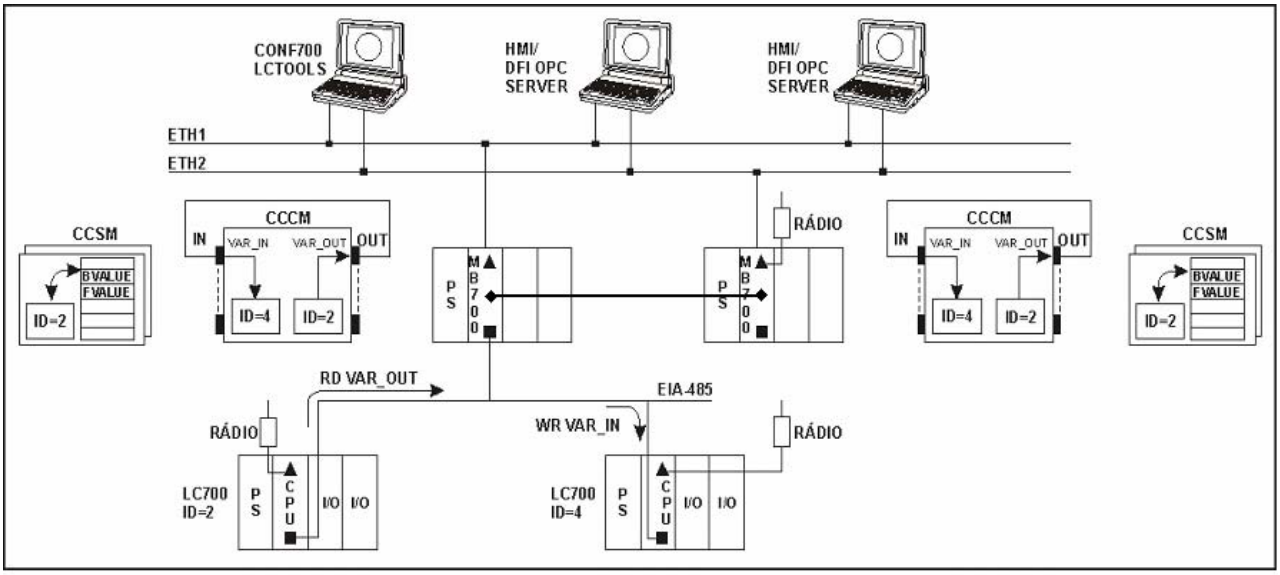


Figura 6.7 – Exemplo de Aplicação do MB-700 como Concentrador e "Peer-to-peer" com Redundância de CPU

Nesta aplicação, o MB-700 suporta as seguintes funções simultaneamente:

- ✓ CONF700 faz download/upload via TCP/IP inclusive remotamente.
- ✓ Download do firmware do LC700 via TCP/IP inclusive remotamente.
- ✓ Supervisão dos LC700s indiretamente através do MB-700, isto é, via DFI OPC *Server* supervisionam-se parâmetros do Bloco CCSM que são cópias das variáveis Modbus dos LC700s.
- ✓ Possibilidade de múltiplos DFI OPC *Server* ou trabalhar com um único *server* e de forma remota. O OPC Client acessa os parâmetros do CCSM.
- ✓ Redundância de redes de comunicação entre os mestres Modbus TCP/IP (CONF700 e LCTools) e DFI OPC *Server* (supervisão) com os LC700s, inclusive por diferentes meios físicos (EIA-485 e rádio).

Múltiplos Mestres Modbus RTU acessando um LC700 por uma única Porta (P2)

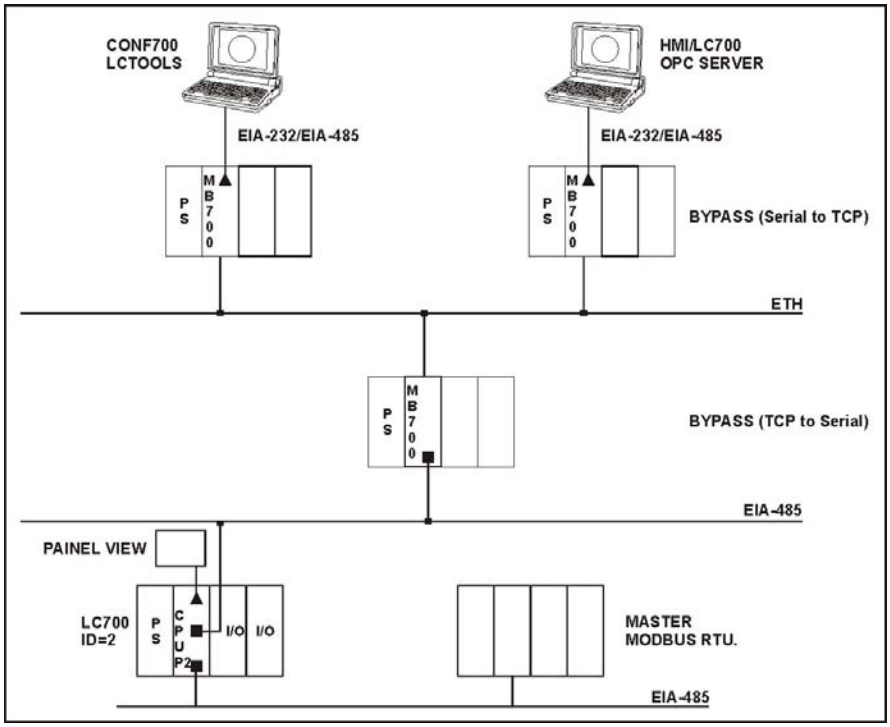


Figura 6.8 – Exemplo de Aplicação do MB-700 como Gateway Invertido (Serial para TCP)

Nesta aplicação, o MB-700 suporta as seguintes funções simultaneamente:

- ✓ Obtém-se a funcionalidade de multi-mestre no Modbus RTU.
- ✓ Dois MB-700s fazendo bypass (Serial to TCP).
- ✓ Um MB-700 fazendo *bypass* (TCP to serial).

No cenário acima, os mestres Modbus RTU (CONF700, LCTOOLS, LC700 OPC Server) são conectados cada um deles a um MB-700 através do canal serial EIA-232. Cada um desses MB-700s conecta-se a um ponto da rede ethernet. Nesta rede ethernet está conectado um MB-700 cujo canal serial EIA-485 forma um barramento de uma rede Modbus RTU. Nesta rede conecta-se um LC700. Na porta P2 deste LC700 conecta-se um mestre Modbus RTU através do canal serial EIA-485.

Capítulo 7

REDUNDÂNCIA HOT STANDBY

Arquitetura de um Sistema Redundante Completo

O sistema redundante abaixo, mostrado pela Figura 7.1, apresenta diferentes funcionalidades com redundância, como será visto a seguir.

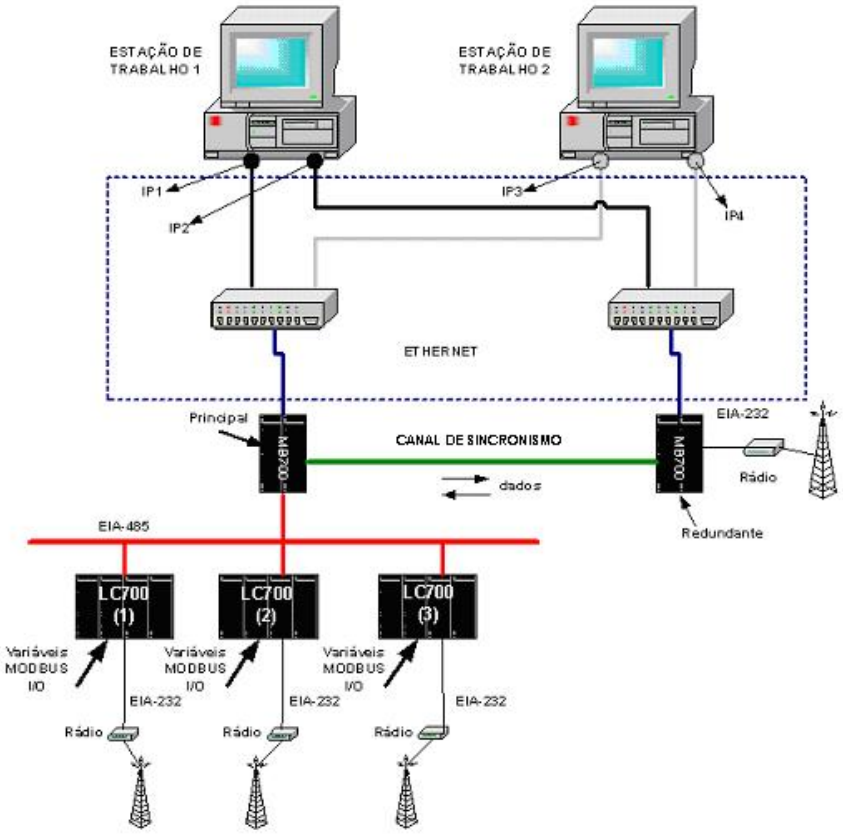


Figura 7.1 – Exemplo de Arquitetura de Redundância do MB-700

As funcionalidades com redundância são :

- Estações de Supervisão: A aplicação apresenta duas estações de supervisão, que podem estar monitorando as mesmas variáveis/tags do mesmo módulo MB-700;
- Rede Ethernet: Cada estação de supervisão possui duas placas de rede, cada qual em uma rede ethernet, utilizando *switches* distintos. Portanto, mesmo havendo problema numa placa de rede ou cabeamento de rede, ou *switch* de um estação de supervisão, a mesma continuará em condições de manter a supervisão, além de estar disponível a outra estação de supervisão;
- Redundância de Fonte e *Backplane*: Cada MB-700 possui a sua fonte e *backplane* e, embora a aplicação acima não mostre, ainda é possível fazer redundância de fontes de alimentação para cada MB-700;
- Redundância do Módulo MB-700: Sendo o módulo responsável por todo o armazenamento e controle de troca de informações entre escravos Modbus, a redundância deste módulo garante a disponibilidade de funcionamento do sistema de supervisão e controle sem interrupções;
- Diagnóstico: Através da monitoração dos parâmetros do *transducer* do MB-700, pode-se rastrear as possíveis falhas que estejam ocorrendo no sistema.

Redundância do Módulo MB-700

Terminologia

***Role*:** Configuração do usuário definindo o módulo como *Main* ou *Backup*.

State: Estado do módulo que depende das condições em que ambos se encontram, e também da configuração Role. Os possíveis estados são *Active* e *Standby*.

***Active*:** Estado que indica basicamente se o módulo está executando os Blocos Funcionais, que cuidará do Scan da Rede e atualização dos dados.

***Standby*:** Estado no qual o MB-700 monitora o comportamento e desempenho do outro módulo, para uma eventual necessidade de assumir como *Active*. Mantém-se constantemente atualizado em relação aos dados processados pela *Active*, através do canal de comunicação de redundância.

***Main*:** É uma configuração do usuário que define qual módulo deve ser preferencialmente o Ativo. Portanto, se ambos MB-700s estiverem em boas condições para executar os Blocos Funcionais (incluindo-se a comunicação Modbus), o módulo configurado como *Main* será o módulo *Active*.

***Backup*:** É uma configuração do usuário que define qual o módulo deve ceder o controle ao outro quando em condições iguais de operação. E deverá assumir como *Active* quando ocorrer falha no outro módulo MB-700.

***Hot Standby*:** Esquema de redundância em que se tem um elemento ativo e outro constantemente em alerta monitorando a condição do ativo, mantendo-se constantemente atualizado para que possa assumir o controle quando há falha no módulo *Main*.

***Local*:** É o módulo MB-700 com o qual se está comunicando e supervisionando. Lembrando que o MB-700 *Standby* pode ser supervisionado, isto é, os parâmetros dos blocos no MB-700 *Standby*, que são cópias dos parâmetros dos blocos do MB-700 *Active*, podem ser monitorados.

***Remote*:** Praticamente quase toda base de dados do MB-700 *Active* é transferida para o MB-700 *Standby*, porém algumas informações não se enquadram neste caso, por exemplo, a configuração (*Role*) e o próprio estado (*State*) da redundância. Através do Bloco *Transducer IDShell* é possível saber não apenas a configuração (*Role*) e estado (*State*) do MB-700, que está sendo supervisionado (*Local*), mas também o outro MB-700 (*Remote*), pois esta informação é trocada entre ambos módulos pelo canal de comunicação de redundância.

Canal de Comunicação para Redundância: Canal de comunicação entre os módulos MB-700s para que ambos se mantenham sincronizadas, isto é, a configuração de ambas seja sempre igual, e as variáveis dinâmicas do processo sejam constantemente transferidas.

***Switchover*:** Chaveamento do controle de um módulo para outro, isto é, um MB-700 que está *Standby* passa a *Active*.

Pré-requisitos do Sistema

Os requisitos listados aplicam-se a ambos os modos de redundância.

A versão de firmware para sistemas redundantes possui a terminação "R". Isto designa um firmware próprio para aplicações em redundância. Com o firmware redundante, o módulo inicializa no modo *Hot Standby* por default, em um estado de segurança chamado "*Sync_Idle*". O usuário poderá mudar o modo de redundância, conforme será visto posteriormente.

A configuração no *SYSCON* deverá ser criada do mesmo modo que para um sistema não-redundante (em caso de dúvidas, veja o Capítulo 3 deste manual). A única diferença é a necessidade de acrescentar um bloco transdutor à *bridge*. Este transdutor será usado para inicializar a redundância.

Na configuração do *SYSCON*, o *Tag* para o bloco transdutor pode ter qualquer nome, preferencialmente, que seja relacionado ao *Tag* do MB-700 ou à planta. Deve-se precaver a não usar *Tags* que já estejam em uso na mesma planta.

Para mais infomações sobre o *SYSCON*, referir-se ao manual de operação e instalação do *Syscon*.

Para qualquer modo de redundância é necessário, antes de tudo, configurar a redundância de rede. A próxima seção explica como isto dever ser feito.

Configurando a Redundância de Rede

Para que todas as ferramentas OPC-Client possam funcionar com redundância de rede, é necessário configurar as *workstations* e o DFI OLE Server.

Configurando a *Workstation*

É possível ter uma ou duas *workstations* (redundância de *workstation*). Seguem os passos para configuração.

1. Cada *workstation* deve ter uma HMI instalada.
2. Cada *workstation* deverá ter duas Placas de Interface de Rede (*Network Interface Card* – NIC).
3. Cada NIC deverá ser configurada em uma faixa diferente de sub-rede (ex.: NIC1, IP=**192.168.164**.50 / Subnet Mask 255.255.255.0 e NIC2, IP=**192.168.163**.50 / Subnet Mask 255.255.255.0).
4. Configure também um *default gateway* de acordo com suas necessidades.
5. Instale dois HUBs ou *switches*. Cada NIC deve ser conectada a um deles de forma que duas Redes Locais (LAN) sejam montadas isoladas uma da outra.
6. Desta forma, cada um dos módulos MB-700 pode ser conectado a um dos HUBs obedecendo as faixas pré-definidas de sub-rede (ex.: Primeiro MB-700, IP=**192.168.164**.51 / Subnet Mask 255.255.255.0 e segundo MB-700, IP=**192.168.163**.51 / Subnet Mask 255.255.255.0).
7. Para testar a rede, use o comando *ping* no prompt do DOS, chamando pelos IPs de cada módulo MB-700 para confirmar a comunicação.

Configurando o DFI OLE *Server*

Existem duas maneiras de configurar o OLE Server para redundância de rede. A seguir, os passos para a configuração em cada caso.

O HMI Client seleciona o DFI OLE Server (local e remoto).

Para este caso, o *Server* local possui um *NIC adapter* específico, e o cliente seleciona qual servidor será usado.

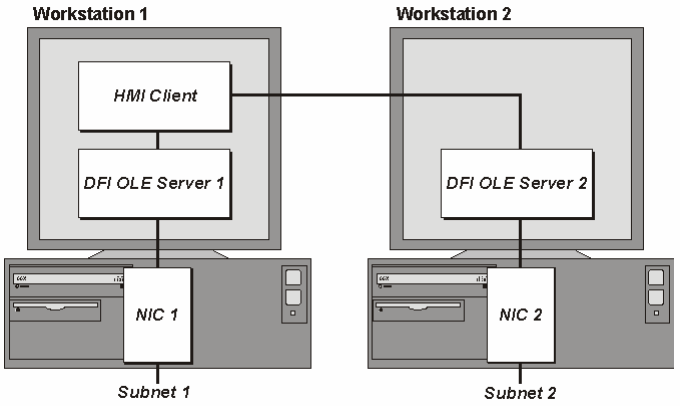


Figura 7.2 – Exemplo de Supervisão de Dados com Gerenciamento da Redundancia pelo OPC com uma única Placa de Rede em cada Estação de Trabalho

1. Configure no arquivo SmarOleServer.ini, o *NIC adapter* que será usado em cada workstation (por exemplo: Primeira workstation, NIC=192.168.164.50 e a Segunda workstation, NIC=192.168.163.50).
2. Desta forma, cada DFI OLE *Server* selecionará o *NIC adapter* especificado.

3. Na hora de configurar a IHM, configure cada *TAG* a ser monitorado, o que pode ser feito de duas formas:

    a) Primeira: usando o *DFI OLE Server Local*; e

    b) Segunda: usando o *DFI OLE Server Remoto* (algumas IHM não permitem este tipo de configuração, e será necessária a ajuda de algum outro *software* auxiliar).

4. Para validar a conexão remota entre o *Client* e o *Server*, certifique-se de configurar o *DCOM e NT Security*.

O DFI OLE Server está conectado a ambas sub-redes onde os módulos redundantes estão.

Neste caso, o cliente emprega apenas um *Server*. O *Server* escolhe dinamicamente qual *NIC adapter* será usado (ver figura 15.3).

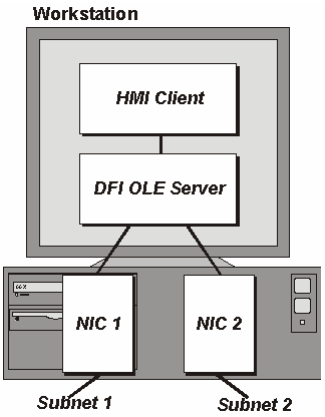


Figura 7.3 – Exemplo de Supervisão de Dados com Gerenciamento da Redundancia pelo OPC com duas Placa de Rede em cada Estação de Trabalho

1 – Configure no arquivo SmarOleServer.ini os *NIC adapters* desejados. Por exemplo:

NIC = 192.168.164.50

NIC2 = 192.168.163.50

2 – Desta forma, o DFI OLE Server terá informação através de ambos os *NIC adapters*.

O último dado atualizado será selecionado pelo DFI OLE *Server* para ser encaminhado para o *client*. Quando o MB-700 está em modo *Hot Standby*, o DFI OLE *Server* selecionará preferivelmente o dado que vem do módulo *Active*, para ser encaminhado para o *client*.

Configurando Redundância Hot Standby

Para habilitar redundância *Hot Standby* e monitorar seu *status*, alguns parâmetros disponíveis no *MB-700 transducer block* devem ser usados.

A maioria dos parâmetros de redundância possui um sufixo. O sufixo "*L*" significa *Local*, ou que o parâmetro traz informações do módulo que está sendo monitorado diretamente através do DFI OLE *Server*. O sufixo "*R*" significa *Remote*, ou que o parâmetro traz informações sobre o outro módulo, conhecidas pelo módulo *Local*, através do canal de sincronismo.

A seguir é apresentada uma descrição funcional destes parâmetros para entender como a redundância *Hot Standby* funciona. Para maiores informações sobre estes parâmetros, veja a tabela descritiva do Bloco Transdutor (Manual de Blocos Funcionais).

FUNCTION_IDS

Este é o único parâmetro a ser configurado. O usuário deve designar um módulo para ser o *Main*, escolhendo o valor *Sync_Main.* Através do canal de sincronismo, o outro módulo automaticamente será inicializado como *Backup*. Isto designa fisicamente o módulo processador Preferencial e o

Redundante, respectivamente. Desta forma, *Main* e *Backup* podem ser entendidos simplesmente como *labels*.

- **RED_ROLE_L / RED_ROLE_R**

Reflete a configuração feita em FUNCTION_IDS, identificando o *Role* (papel) do módulo, *Sync_Main* ou *Sync_Backup*.

- **RED_STATE_L / RED_STATE_R**

Active – executa todas as tarefas e gera toda a informação.

Standby – não executa as tarefas, apenas recebe toda a informação gerada pelo *Active* e permanece pronto para assumir, se necessário.

Not Ready – redundância não disponível.

As diferentes falhas que podem ocorrer no sistema levam-no a um *switch over*, quando o *Standby* assume o estado de *Active* e vice-versa numa forma sem sobressaltos. A seguir, são apresentadas as possíveis causas de *switch over*, divididas em dois tipos:

Falhas Gerais

Quando todo um módulo falha, as possíveis falhas são:

- Falha de *Hardware*
- Falha na Alimentação
- Remoção do Módulo Processador do *Backplane*.

Falhas de Má Condição

Quando uma das interfaces de um módulo falha:

- Falha na Comunicação Modbus

O sistema é capaz de checar qual módulo está em melhores condições, elegendo-o como *Active*.

É assegurada a recuperação de uma falha por vez. Ou seja, uma vez ocorrida uma falha, uma segunda falha só poderá ser recuperada pela redundância caso a primeira falha já tenha sido corrigida. Enquanto a falha não for corrigida, a redundância não estará totalmente disponível (em caso de *Falha de Má Condição*), ou mesmo indisponível (no caso de *Falha Geral*).

Para o caso de *Falha Geral*, assim que o módulo em falha se recupera, ou é substituído, os módulos se tornam automaticamente um par redundante, ou seja, o sistema reconhece automaticamente um novo módulo inserido.

- **RED_SYNC_STATUS_L / RED_SYNC_STATUS_R**

Este parâmetro informa todos os estados possíveis de sincronismo entre os módulos.

SYNC STATUS	DESCRIÇÃO
Stand Alone	Há apenas um módulo operando. Se o sistema sincronizou pelo menos uma vez, e este valor aparece, indica que o outro módulo teve uma *Falha Geral*.
Synchronizing	Os módulos estão checando as configurações entre si para chegarem ao estado de *Synchronized*. Pode levar um tempo de até 9 min. no máximo (enquanto o sistema aguarda que o módulo em "Not Ready" complete suas *Live Lists*).
Updating Remote	Logo após um download de configuração, o módulo transfere toda a configuração para o outro através do canal de sincronismo.
Maintenance	O módulo está sendo configurado pelo outro módulo através do canal de sincronismo ou pelo SYSCON. Caso apareça para ambos o parâmetro "*L*" e "*R*", indica que nenhum dos módulos foi configurado ainda.
Synchronized	Os módulos estão em perfeito sincronismo. O módulo *Active* atualiza continuamente as bases de dados do *Standby*.

SYNC STATUS	DESCRIÇÃO
Warning: Role Conflict	Se um segundo módulo é conectado no painel, com o mesmo *Role* (papel) daquele que já está em operação, essa mensagem é mostrada. O procedimento para corrigir este conflito é executar um *Factory Init* no novo módulo e aguardar que o sincronismo do sistema.
Warning: Sync Cable Fail	Caso ocorra uma falha no cabo de sincronismo, essa mensagem é exibida. O sistema não terá a redundância até que a falha no cabo de sincronismo seja corrigida.
Warning: Updating Remote Fail	Se uma falha ocorre na transferência de configuração do *Active* para o *Standby*, essa mensagem é mostrada. O procedimento é executar um *Factory Init* no módulo que não está em *Active* e aguardar até que a transferência seja completada com sucesso.

- **RED_BAD_CONDITIONS_L / RED_BAD_CONDITIONS_R**

Pode apresentar o valor abaixo:

BIT	BAD CONDITION	DESCRIÇÃO
0	Modbus	Quando trabalhando como mestre e não houver resposta do escravo Modbus, significa que a comunicação Modbus está em más condições. A causa pode ser uma falha no caminho de comunicação, ou até mesmo falha no módulo escravo.

O valor desejável e mais provável é <*none*> para ambos os módulos (*L* e *R*), o que assegura boas condições para ambos, ou seja, redundância completamente disponível. Este parâmetro pode ter duas funções:

- Uma *falha de má condição* para o módulo em *Active* leva o sistema a um *switch over*. Neste caso, este parâmetro atua como um registro da causa do último *switch over*.

- Quando uma *falha de má condição* ocorre para o módulo em *Standby* este parâmetro mostra esta condição como um alarme. Assim, advertindo o operador que o módulo em *Standby* apresenta um determinado problema, permite manutenção pró-ativa para que se tenha a redundância completamente disponível.

- **RED_MAIN_WDG / RED_BACKUP_WDG**

São *watchdogs* que indicam o status de comunicação entre a IHM e os módulos processadores. Enquanto seus valores estiverem incrementando num intervalo de 2 segundos, as respectivas conexões com a rede (*Main* e *Backup*) estarão funcionando bem.

Em resumo, a redundância está completamente disponível, SOMENTE se os módulos estão em *Synchronized* e têm o valor <*none*> nos parâmetros de *Bad Conditions* (*L* e *R*).

As seguintes operações podem ser realizadas sem interrupção do processo da planta: substituição de um módulo com falha, correção de uma falha de cabo SYNC, atualização do firmware e adição de redundância em um sistema em operação.

NOTA
Quando o LED *STAND BY* está aceso, significa que o módulo está em *Standby.* Quando o LED está apagado, o módulo pode estar em *Active* ou *Not Ready*. Se um dos módulos está em *Standby*, o outro estará em *Active*.

Seguem os passos para a configuração e manutenção da redundância Hot Standby. Recomenda-se que os passos sejam todos lidos e entendidos antes de serem executados.

Configurando o Sistema pela Primeira Vez

Este é o procedimento para configurar o sistema pela primeira vez com redundância *Hot Standby*,

no *start up* da planta.

1 – Com o conector dos canais de sincronismo desconectado, execute um *Factory Init* em ambos módulos para garantir o estado default.

2 – Conecte ambos módulos através do canal SYNC.

3 – Abra a configuração desejada no *SYSCON* e coloque-o em modo *On-line*. Clique com o botão direito no ícone *bridge* e na opção *Attributes* escolha um dos módulos listados no campo *Device Id*. O módulo escolhido será aquele a ser configurado como *Main*. No menu principal do *SYSCON* vá ao menu *Export* e clique *Tags*.

4 – Ainda no ícone da *bridge,* clique com o botão direito em *FB VFD* e então clique em *Block List*. Uma nova janela será aberta mostrando todos os blocos que estão pré-instanciados no módulo. Então, nesta janela, clique com o botão direito no *transdutor* realizando um *Assign Tag* com o *tag* que está previsto na configuração. Feche a janela *Block List*.

5 – Clique com o botão direito no ícone do *transdutor* da *bridge* e escolha *On Line Characterization*. Configure o parâmetro FUNCTION_IDS como *Sync_Main*. Através do canal de sincronismo o outro módulo será automaticamente inicializado como *Backup*. Após isto, ambos os parâmetros RED_SYNC_STATUS (*L* e *R*) devem indicar *Maintenance*, o que significa que nenhum dos módulos foi configurado ainda.

6 – Caso necessário, realize *Assign Tag* para todos os *field devices*. Aguarde até que as *Live Lists* de todos os canais estejam completas. Então, configure o sistema a partir do módulo *Active* executando todos os downloads de configuração necessários, da mesma forma que para um sistema MB-700 não-redundante.

7 – Assim que os downloads forem completados com sucesso, o transdutor apresentará as seguintes fases:

- O *Active* irá transferir toda a configuração para o outro módulo (RED_SYNC_STATUS_L como *Updating Remote* e RED_SYNC_STATUS_R como *Maintenance*).
- Após a configuração ter sido transferida, os módulos podem levar algum tempo para sincronizar (parâmetros RED_SYNC_STATUS (*L* e *R*) como *Synchronizing*). Este é o tempo necessário para que os módulos chequem a configuração um com o outro.
- Finalmente, os módulos irão sincronizar (parâmetros RED_SYNC_STATUS (*L* e *R*) como *Synchronized* e RED_STATE_R como *Standby*. Com o sistema nestas condições, o *Active* estará atualizando constantemente o *Standby*.

Trocando a Configuração

Apenas siga os passos 6 e 7 da seção "*Configurando o Sistema pela Primeira Vez*".

Substituição de um Módulo com Falha

1 – Com o conector do canal de sincronismo desconectado, insira o novo módulo no *backplane*.

2 – Atualize o firmware do novo módulo, caso seja necessário. Execute um *Factory Init* no novo módulo para garantir o estado default.

3 – Conecte o conector SYNC ao novo módulo.

4 – O novo módulo será automaticamente reconhecido pelo *Active* e ambos irão permanecer em *Synchronizing* por algum tempo. Assim que o sistema tiver o status *Synchronized* e <*none*> nos parâmetros *Bad Conditions*, a redundância estará totalmente disponível e simulações de falhas podem ser feitas.

Correção de uma Falha no Canal de Sincronismo

Se ocorrer uma falha na conexão do cabo SYNC (retirada do cabo, por exemplo), o sincronismo será momentaneamente perdido, assim os dois módulos se tornarão ativos. Quando a falha na conexão do cabo SYNC for corrigida (reconectado o cabo SYNC), o módulo Main assumirá como Active novamente. Desta forma, ambos permanecerão em *Synchronizing* por algum tempo. Quando o sistema tiver o status *Synchronized* e <none> nos parâmetros *Bad Conditions*, a redundância estará totalmente disponível e simulações de falhas podem ser feitas.

Atualização de Firmware sem Interrupção do Processo

Este procedimento descreve como atualizar o firmware de ambos os módulos sem interromper o

processo da planta.

1 – Certifique-se de que o sistema tenha o status *Synchronized* e *<none>* nos parâmetros *Bad Conditions*. Então, usando o FBTools atualize o firmware do módulo em *Active*. Neste momento, o outro módulo irá assumir a planta.

2 – Após a atualização do firmware ter sido finalizada, os módulos irão sincronizar um com o outro, com o *Active* transferindo toda a configuração para o outro. Aguarde o sistema ter o status *Synchronized* e *<none>* nos parâmetros *Bad Conditions*.

3 – Usando o FBTools, atualize o firmware do módulo em *Backup*. Neste momento, o outro módulo irá assumir a planta.

4 – Após a atualização do firmware ter sido finalizada, os módulos irão sincronizar um com o outro, com o *Active* transferindo toda a configuração para o outro. Assim que o sistema tiver o status *Synchronized* e *<none>* nos parâmetros *Bad Conditions*, a redundância estará totalmente disponível e simulações de falhas podem ser feitas.

Adicionando Redundância a um Sistema Existente

Se um sistema não redundante é intencionado para ser redundante no futuro, no *startup* da planta as seguintes condições devem ser obedecidas:

- A arquitetura LAN poderá ser expandida para atender o que está descrito no item "Redundância com Supervisão via OPC".
- O único módulo deve usar um firmware redundante que suporta redundância *Hot Standby*. O parâmetro *FUNCTION_IDS* deve ser setado como *Sync_Main*. Então, o módulo pode ser configurado e enquanto há somente um módulo, ele trabalhará no status *Stand Alone*.

Obedecendo estas condições, a redundância pode ser adicionada a qualquer momento sem interromper o processo.

Para adicionar redundância a um sistema, siga alguns passos descritos na seção "Substituindo um Módulo com Falha".

Conexão Física SYNC_CABLE

A conexão física é simples, como mostra a figura abaixo:

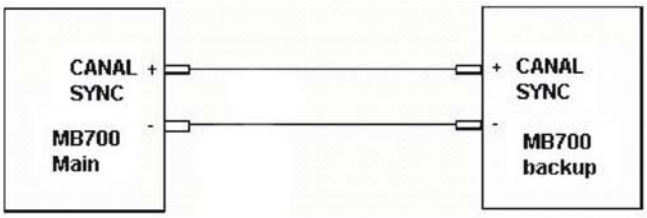


Figura 7.4 - Conexão Física do Canal de Redundancia do MB-700

Tabela de Parâmetros Adicionais

Novos Parâmetros e Valores do Transdutor IDShell para a Redundância do MB-700.

Index	Parâmetro	Tipo de Dado (comprimento)	Faixa Válida/ Opções	Valor Default	Unidades	Store/ Modo	Descrição
13	FUNCTION_IDS	Unsigned8	1:Passive 2:Active 3:Backup 4:Active_Not_Link_Master 7:Sync_Idle 8:Sync_Main 9:Sync_Backup	7	E	D / RW	Função do dispositivo local na redundância. **Passive**, **Active, Backup** e **Active_Not_Link_Master** não são funções sincronizadas e são válidas somente para supervisão e redundância LAS. A redundância Hot Standby é setada via as seguintes funções: **Sync_Idle** é a função default, após a inicialização de fábrica. A 4ª porta é usada para sincronizar dois MB-700s diferentes. **Sync_Main** inidica o processador preferencial para assumir as tarefas. **Sync_Backup** indica o processador que assume as tarefas.
169	RED_ROLE_L	Unsigned8	1:Passive 2:Active 3:Backup 4:Active_Not_Link_Master 7:Sync_Idle 8:Sync_Main 9:Sync_Backup	7	E	D / RO	Função de Redundância para o dispositivo local. Idem à descrição FUNCTION_IDS.
170	RED_STATE_L	Unsigned8	0:Not Ready 1:Standby 2:Active	0	E	D / RO	Estado de redundância para o dispositivo local **Not Ready** – Não está pronto para executar. **Standby** – Presente mas não executando. **Active** – Rodando as tarefas.
171	RED_SYNC_STATUS _L	Unsigned8	0: Not defined 1: Stand Alone 2: Synchronizing 3: Updating Remote 4: Maintenance 5: Synchronized 6: WARNING: Role Conflict 7: WARNING: Sync Cable Fail 8: WARNING: Updating Remote Fail 9: Warning 1 10: Warning 2	0	E	D / RO	Status de Sincronismo para o dispositivo local: 0: Valor Inicial 1: Operação Stand alone 2: Verificando a Configuração para sincronização 3: Transferência de toda a Configuração para o módulo remoto 4: Recebendo toda a Configuração do módulo remoto 5: Os módulos são completamente atualizados uns com os outros 6: O módulo extra tem a mesma função daquele que está executando 7: Falha no cabo de sincronismo 8: Falha na atualização do remoto 9: Uso futuro 10: Uso futuro
172	RED_ROLE_R	Unsigned8	7:Sync_Idle 8:Sync_Main 9:Sync_Backup	7	E	D / RO	Função de Redundância para o dispositivo remoto. Idem à descrição FUNCTION_IDS.
173	RED_STATE_R	Unsigned8	0:Not Ready 1:Standby 2:Active	0	E	D / RO	Função de Redundância para o dispositivo remoto. Idem à descrição RED_STATE_L.
174	RED_SYNC_STATUS_R	Unsigned8	0: Not defined 1: Stand Alone 2: Synchronizing 3: Updating Remote 4: Maintenance 5: Synchronized 6: WARNING: Role Conflict 7: WARNING: Sync Cable Fail 8: WARNING: Updating Remote Fail 9: Warning 1 10: Warning 2	0	E	D / RO	Status de Sincronismo para o dispositivo remoto: 0: Valor Inicial 1: Operação Stand alone 2: Verificando a Configuração para sincronização 3: Transferência de toda a Configuração para o módulo remoto 4: Recebendo toda a Configuração do módulo remoto 5: Os módulos são completamente atualizados uns com os outros 6: O módulo extra tem a mesma função daquele que está executando 7: Falha no cabo de sincronismo 8: Falha na atualização do remoto 9: Uso futuro 10: Uso futuro

Index	Parâmetro	Tipo de Dado (comprimento)	Faixa Válida/ Opções	Valor Default	Unidades	Store/ Modo	Descrição
175	RED_BAD_CONDITIONS_L	Bitstring(2)		0	E	D / RO	Condições ruins para o dispositivo local.
176	RED_BAD_CONDITIONS_R	Bitstring(2)		0	E	D / RO	Condições ruins para o dispositivo remoto.
177	RED_RESERVED1	Unsigned8	0 ~ 255	0	NA	D / RW	Reservado para uso futuro.
178	RED_RESERVED2	Unsigned8	0 ~ 255	0	NA	D / RW	Reservado para uso futuro.

Tabela 7.1 - Parâmetros Adicionais de Redundância

Descrição do Significado dos Bits de RED_BAD_CONDITIONS_L / R

Bit	Variable
0	Modbus
1	H1-1 (não usado)
2	H1-2 (não usado)
3	H1-3 (não usado)
4	LiveList
5	Res_1
6	Res_2
7	Res_3

Tabela 7.2 – Tabela de Descrição dos Bits

Apêndice A

SOLUCIONANDO PROBLEMAS

O MB-700 disponibiliza alguns recursos de inicialização para solucionar determinados problemas. Estes recursos são:

ATENÇÃO
Qualquer que seja a forma de inicialização selecionada causará um grave impacto no sistema !

Reset

Clique o *Push-Button* da direita (ver no detalhe da Figura A1 seguinte os dois pequenos botões localizados acima do conector de Modbus 232) e o sistema executará o RESET levando alguns segundos para a inicialização correta do sistema. De acordo com o procedimento via FBTools, neste momento um novo IP será atribuído automaticamente ou o último IP setado será aceito pelo sistema. Certifique-se de que o LED RUN e o LED ETH10 permaneçam acesos.

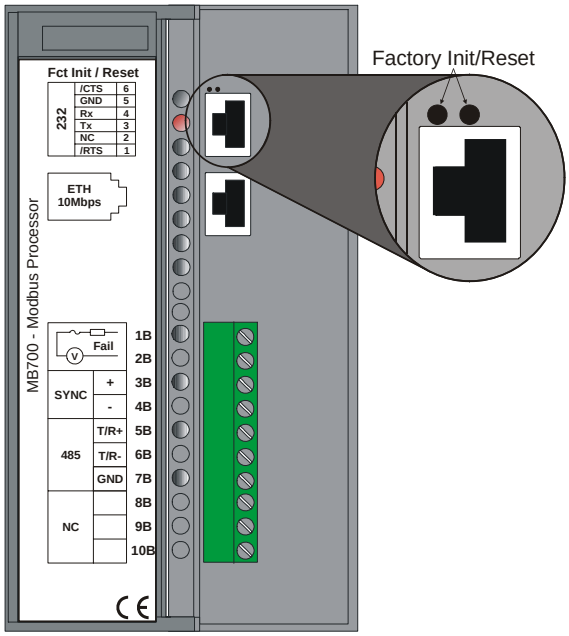


Figura A1 – MB-700

Factory Init

Mantenha pressionado o *Push-Button* da esquerda e, em seguida, clique o *Push-Button* da direita garantindo que o LED FORCE esteja piscando 1 vez a cada segundo. Libere o *Push-Button* da esquerda e o sistema executará o RESET apagando as configurações anteriores.

Modo HOLD

Mantenha pressionado o *Push-Button* da esquerda e, em seguida, clique o *Push-Bottom* da direita duas vezes garantindo que o LED FORCE esteja piscando 2 vezes a cada segundo. Libere o *Push-Button* da esquerda e o sistema executará o RESET e irá para o modo HOLD. Certifique-se de que o LED HOLD e o LED ETH10 permaneçam acesos.

Com o MB-700 neste modo, o FBTools Wizard pode ser utilizado para atualização do firmware ou alteração do endereço IP.

Utilize o *Reset* novamente caso queira retornar para o modo de execução (RUN).

DICAS	
1-	Qualquer um dos modos (Factory Init e Modo HOLD) pode ser evitado uma vez iniciados, mantendo-se pressionado o *Push-Bottom* da direita e liberando-se o *Push-Button* da esquerda primeiro.
2-	Se o usuário perder a conta do número de vezes que o *Push-Button* da direita foi pressionado, basta verificar o número de vezes que o LED FORCE está piscando a cada segundo. Ele voltará a piscar uma vez por segundo depois do quarto toque (ou seja a função é rotativa).
3-	Para pressionar o *Push-Button* do Factory Init/Reset é adequado o uso de algum instrumento pontiagudo (ex. caneta esferográfica).

Quando Usar os Procedimentos de Factory Init/Reset

1. Como resetar o MB-700 sem desligá-lo?
Use o procedimento de RESET.

2. O LED HOLD permanece aceso, como devo proceder?
Caso após o MB-700 ser ligado (ou resetado) e o LED HOLD permanecer aceso, é provável que o Firmware esteja corrompido. Deve-se proceder um Firmware Download para carregar um firmware novamente. Para isso, siga os seguintes passos:

2.1- Assegure-se que o MB-700 esteja ligado e que esteja conectado à Sub-Rede. Caso não esteja, use o procedimento "Conectando o MB-700 à Sub-Rede". Certifique-se que o LED HOLD esteja aceso;
2.2- Execute o FBTools Wizard, (localizado no diretório de trabalho da Smar, geralmente "drive:\Program Files\Smar\FBTools\FBTools Wizard.exe" ,diretamente pelo atalho "FBTools Wizard" na "colméia" do System302;
2.3- Na tela principal (Choose device type) selecione o **MB-700** e pressione o botão "**Next**";
2.4- Escolha o caminho para o DFI OLEServer a ser usado (default: Local) e pressione o botão "***Next***";
2.5- Selecione o módulo MB-700 desejado na opção "Module" usando como referência o número de série (verifique na etiqueta lateral, no próprio MB-700).
2.6- Pressione o botão "**Browse…**" para selecionar qual arquivo de firmware será carregado (arquivo *MB-700**.ABS);
2.7- Após selecionar o arquivo a ser carregado, pressione a botão "**Finish**" para iniciar o download do firmware;
2.8- Durante o download será apresentada a tela de progresso da operação;
2.9- Ao final da operação será apresentada uma mensagem de status da operação de download. Neste momento o MB-700 já estará no "Modo Run". Clique no botão "OK" (assegure-se que o LED RUN esteja aceso);
2.10- Para encerrar clique em "**Finish**" na tela seguinte.

3. O FBTools Wizard não consegue colocar o MB-700 em HOLD, como devo proceder?
Use o procedimento do Modo HOLD. Colocado o MB-700 em HOLD, execute o procedimento de atualização do firmware usando os passos descritos no item 2.
Se mesmo assim continuar havendo problema, é possível que esteja relacionado à conexão TCP/IP (verificar os cabos e o LED ETH10).

4. O firmware inicia a execução, mas depois de um certo tempo trava, como devo proceder?
Pode ser um problema da configuração, use o procedimento do Factory Init e reconfigure o MB-700.
Caso o problema persista, será necessário fazer um novo download de firmware no MB-700.

5. O LED ETH10 não acende, como devo proceder?
Verificar se o cabo foi conectado corretamente, ou se o cabo não está rompido. Lembre-se que da especificação do cabos:
DF54 – Cabo Padrão. Para ser usado em uma rede entre MB-700 e Switch/HUB.
DF55 – Cabo Cruzado (*Cross*). Para ser usado ponto a ponto entre PC e MB-700.

6. O LED FORCE está piscando, como devo proceder?
Use o procedimento de RESET.

7. O FBTools não mostra todos os MB-700s que estão na Sub-Rede, como devo proceder?
Provavelmente está havendo conflito de endereço IP nesta Sub-Rede. Para solucionar este tipo de problema deve-se desconectar todos os MB-700s desta Sub-Rede e executar o procedimento

"Conectando o MB-700 à Sub-Rede" para cada módulo, assegurando que o endereço a ser usado não esteja associado a outro equipamento da rede.

8. O FBTools não encontra o MB-700.

- Certifique-se que o procedimento inicial de conexão foi seguido, ou seja, foi inicialmente colocado o IP Default via Modo de Reset 3 e o computador foi colocado com IP 192.168.164.101.
- O cabo ethernet utilizado deve ser o DF54 quando usando HUB ou SWITCH. Use o cabo DF55 se for uma conexão direta entre computador e MB-700.
- Teste se a placa de rede do micro está OK executando o comando PING para o IP do próprio micro via o prompt do DOS.
- Teste se a conexão ethernet está OK executando o comando PING para o MB-700.

9. O MB-700 estava operando corretamente, desliguei e liguei novamente e agora nenhum tipo de reset funciona e o LED Hold fica constantemente aceso e/ou piscando.
Deve-se neste caso fazer o uso da *Boot Flash*.

10. Preciso usar a *Boot Flash* para recarregar o Programa Boot.
Utilize o procedimento de Fábrica "Carregando o programa Boot no MB-700".

11. Durante a operação do Syscon, no *On line Characterization* de alguns blocos, perdi a conexão com o MB-700.
Versões do System302 5.0 anteriores ao Service Pack 8 possuíam um bug que podiam gerar o efeito acima. Neste caso, somente fechando o Syscon e abrindo novamente e ou resetando o MB-700 este problema se resolverá.

12. A licença não é aceita pelo programa *Get License*.
Siga os procedimentos a seguir:
1. Tente registrar a licença DEMO. No Get License há um botão "Use DEMO keys". Caso funcione, o problema deve ser algum erro na digitação da chave.

2. Se ainda assim não funcionar, verifique a existência da variável SmarOlePath no ambiente. Use "Start\Programs\Administrative Tools\Windows NT Diagnostics" na pasta Environment (Ambiente) e verifique se existe uma variável SmarOlePath. Caso não exista, execute o programa "Interface Setup" da pasta de trabalho da Smar que ela será criada.
Obs: Use somente caracteres que sejam números e traços "-". NÃO use espaços e caracteres simbolos "! @ # $ % ^ & * ( ) _ + ~ < > , . / ? \ | { } [ ] :;"

3. Execute o registro do servers novamente. Na pasta de trabalho da Smar (Program Files\Smar\OleServers\) execute o programa Register.Bat.

4. Caso as opções anteriores tenham falhado, pode-se gerar o arquivo de licença manualmente:
- Use um editor de texto ASC (por exemplo, *Notepad*) pois o arquivo não pode conter caracteres de formatação. O nome de cada arquivo e seu conteúdo são apresentados a seguir:
Arquivo: Syscon.dat
SMAR-MaxBlocks-**55873-03243-22123-04737-10406**

Arquivo: OleServer.dat
#PCI OLE Server
SMAR-OPC_NBLOCKS**8-23105-23216-11827-2196**

Arquivo: DfiOleServer.dat
#DFI OLE Server
SMAR-DFIOPC_NBLOCKS**8-19137-32990-37787-24881-12787**

As chaves mostradas são para a licença DEMO, o usuário pode usar as chaves fornecidas pela Smar.

13.Tento mudar um valor estático de um bloco Modbus, mas o valor não é atualizado.
Para que um valor estático de um Bloco Modbus seja atualizado, primeiramente é necessário que o bloco seja colocado em "O/S", o que permite que os valores estáticos possam ser mudados.

14. Após mudar algum valor estático de um bloco, e colocar o MODE_BLK.Target para "Auto" o atual não vai para "Auto".
Se algum parâmetro estático de um Bloco Modbus for alterado, o bloco só irá para "Auto" após realizar o "ON_Apply" no Bloco CCCF.

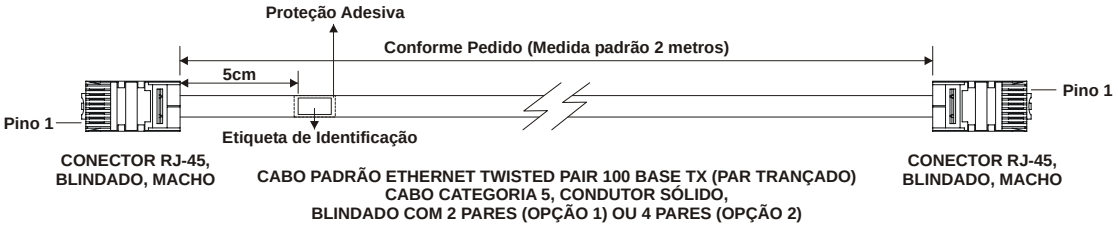
Apêndice B

CABLAGEM

Especificação do Cabo Ethernet

Caso seja necessário a montagem de um novo cabo Ethernet, abaixo estão as especificações do cabo Par Trançado, conforme Código de Pedido para DF54 ou DF55.

DF54 – Cabo Padrão. Para ser usado em uma rede entre MB-700 e Switch/HUB.
DF55 – Cabo Cruzado (*Cross*). Para ser usado ponto a ponto entre PC e MB-700.



DF54		
1	*Marrom*	1
2	*Branco / Marrom*	2
3	*Laranja*	3
6	*Branco / Laranja*	6

DF55 Cross		
1	*Marrom*	3
2	*Branco / Marrom*	6
3	*Laranja*	1
6	*Branco / Laranja*	2

Figura B.1 – Especificações do Cabo Par Trançado – DF54/DF55

OBS: As cores são apenas sugestões. O importante é usar os pares (cor XXX e branco/cor XXX).

Especificação do Cabo Serial

Interface EIA-232

Conectando MB-700 ao SI-700 ou DF58

Para conectar a porta serial RS232 do MB-700 ao módulo SI-700 (Conversor RS232/RS485) será necessário um cabo DF59 ou deverá ser montado um cabo conforme o esquema seguinte.

DF59 – Cabo usado para conectar MB-700 e SI-700

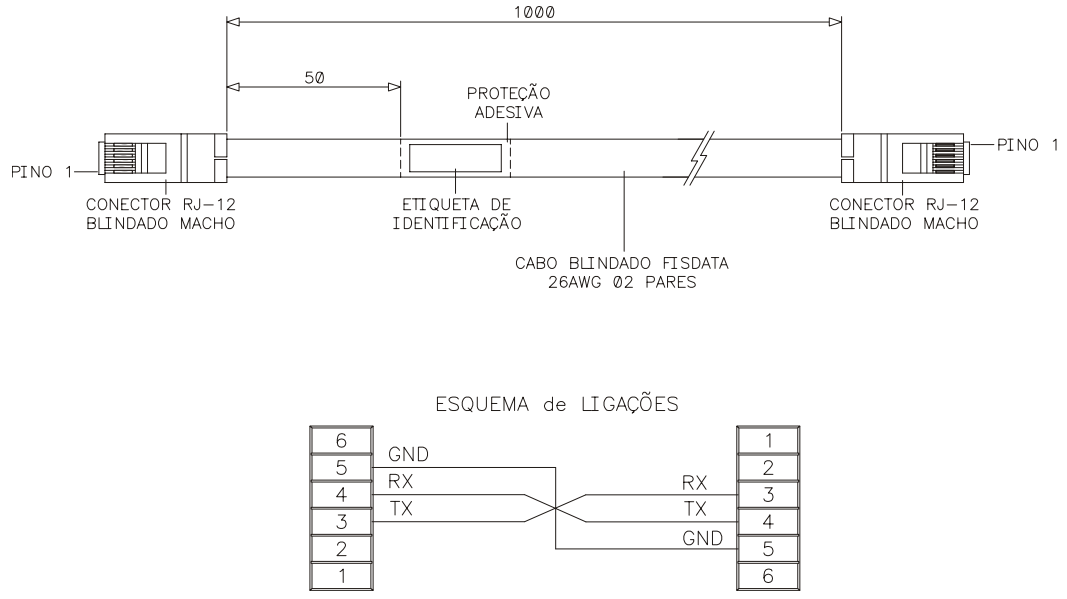


Figura B.2 –Conectando MB-700 ao SI-700 ou DF58

Conectando MB-700 ao DF51/FC302

Para conectar a porta serial RS232 do MB-700 a porta serial do DF51, FC302 ou um outro MB700, deverá ser montado um cabo com conexão entre RJ12 e RJ12, conforme o esquema seguinte:

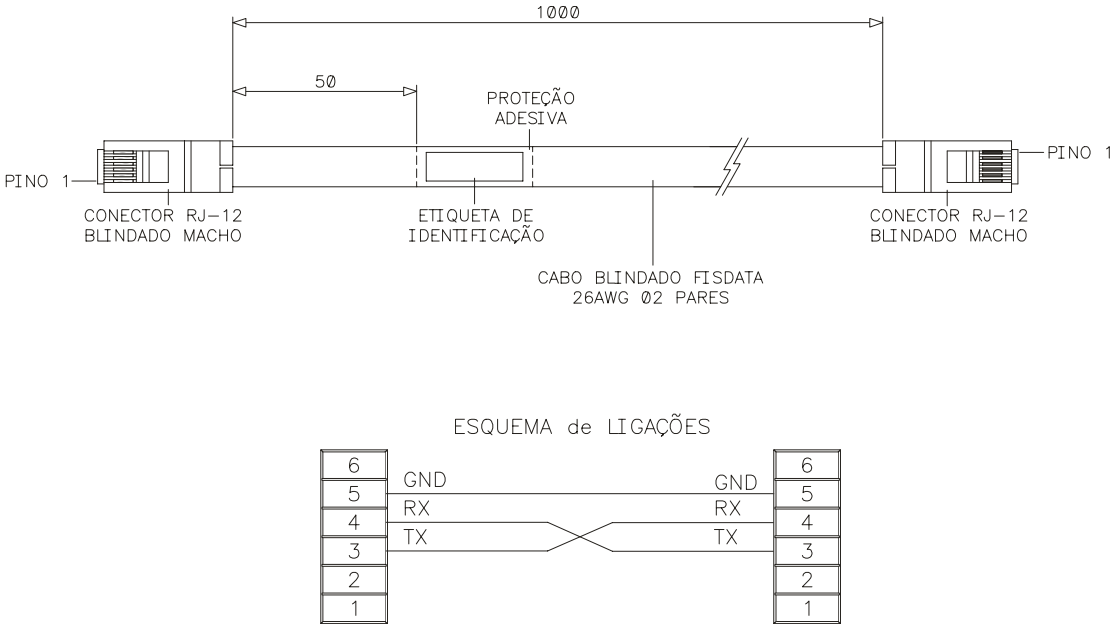


Figura B.3 – Conectando MB-700 ao DF51/FC302

Conectando MB-700 ao Computador

Para montar um cabo serial entre o **MB-700 (Processador) e o computador**, siga as instruções seguintes que mostra uma conexão entre RJ12 (usado no MB-700) e DB9 fêmea.

DIAGRAMA DE LIGAÇÃO DOS CABOS

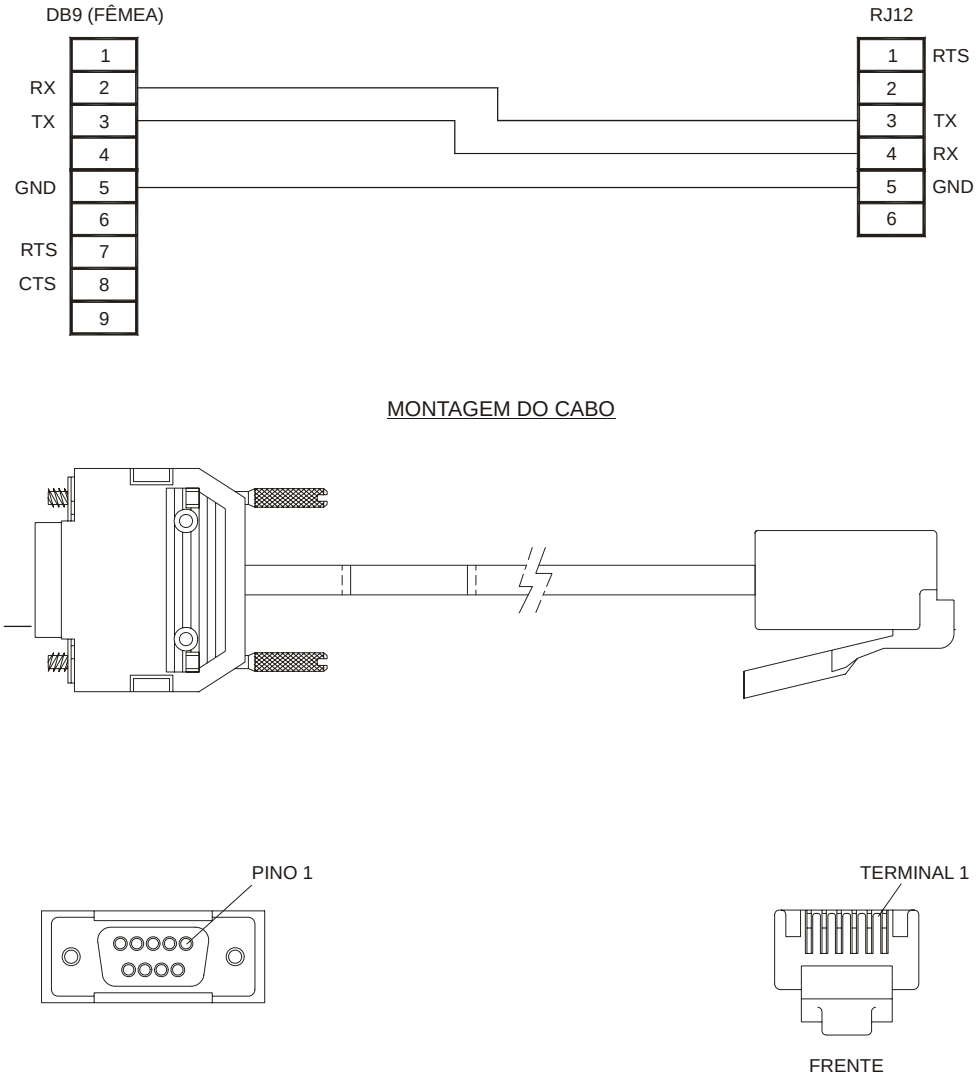


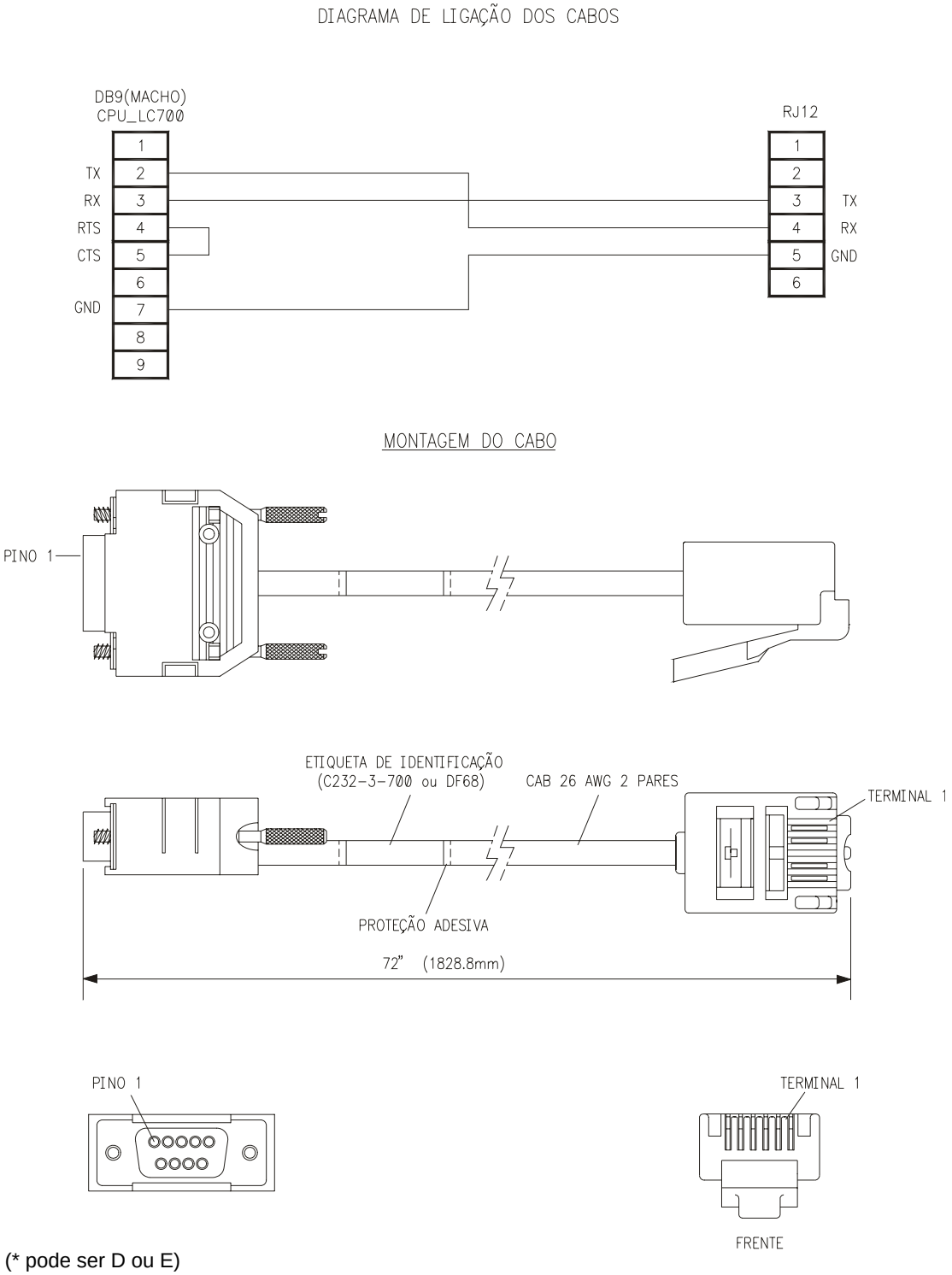
Figura B.4 – Conectando MB-700 ao Computador

Conectando MB-700 ao LC700

Há dois cabos para fazer a conexão do MB-700 ao LC700. Os esquemas de montagem são mostrados nos itens a seguir.

Cabo EIA-232 para conectar CPU-700-X3* e MB-700

Código de pedido
C232-3-700 – Cabo para conexão entre CPU-700 e MB-700

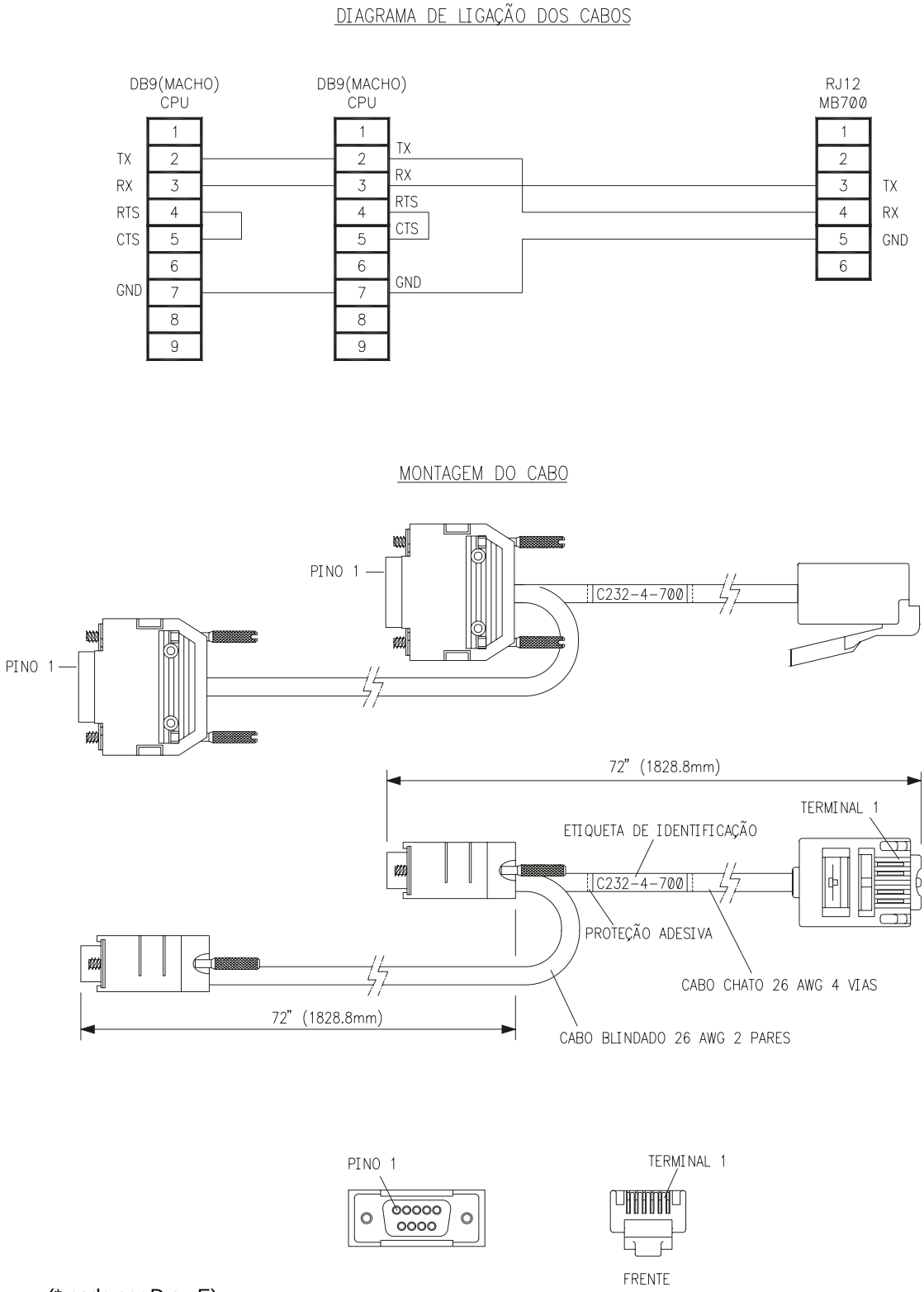


(* pode ser D ou E)

Figura B.5 – Conectando MB-700 ao LC-700 - (LC700 sem Redundância)

Cabo EIA-232 para conectar CPU-700-X3R* e MB-700

Código de pedido
C232-4-700 – Cabo para conexão entre CPU-700 –R e MB-700



(* pode ser D ou E)

Figura B.6 – Conectando MB-700 ao LC-700 - (LC700 com Redundância)

Interface EIA-485

Ver na parte frontal da borneira do MB-700 (Capítulo 2).

Pinos do Bloco Terminal

Pinos	Descrição
5	+: EIA 485
6	-: EIA 485
7	GND: Referência para sinal de comunicação RS485.

Tabela B.1 – Descrição dos Pinos do Bloco Terminal

Abaixo é mostrado um exemplo de ligação do MB-700 a uma rede 485. Para longas distâncias do cabo 485 é recomentado utilizar um terminador (resistor de 120ohms) em ambas as extremidades da rede. O MB-700 não possui este terminador interno.

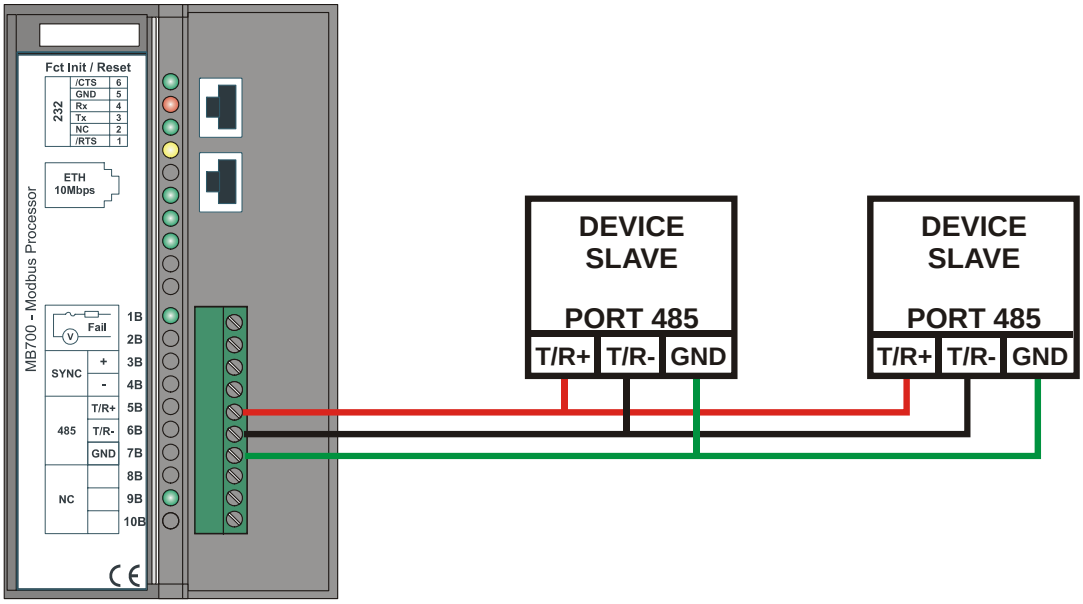


Figura B.7 – Exemplo de Ligação do MB-700 a uma Rede 485

Dimensões

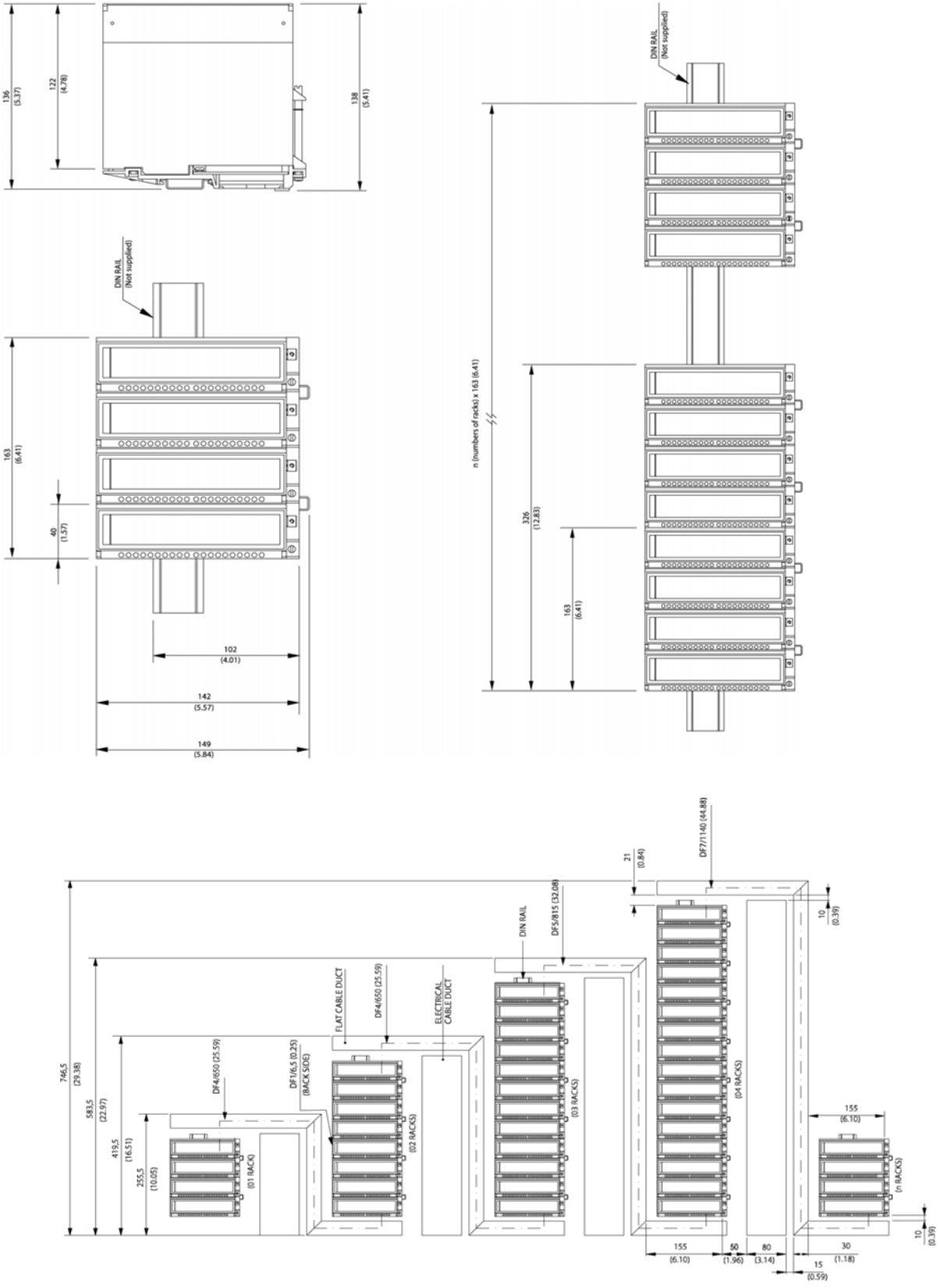


Figura B.8 – Dimensões

Apêndice C

ESPECIFICAÇÕES PARA O MB-700

Especificações Técnicas

CONTROLADOR	
Tipo	32 Bit RISC (CPU Clock @25 MHz)
Performance	50 MIPS
Memória para código	2 Mbytes, 32 Bits Memória Flash (Firmware Regravável)
Memória para dados	2 Mbytes, 32 Bits NVRAM (Armazenamento de dados e configuração)

PORTAS DE COMUNICAÇÃO	
Ethernet	1 Porta Ethernet @10 Mbits (Conector RJ-45)
Serial	1 Porta EIA-232@ 9.6 Kbps-115.2 Kbps (Conector RJ11) 1 Porta EIA-485 @ 9.6 Kbps-115.2 Kbps
Canal de Sincronismo	1 Porta Independente com DMA-Baudrate 31,25 Kbps (SYNC)

POTÊNCIA INTERNA	
Provida pelo IMB	5Vdc/ 950 mA
Total Dissipação	4,75 W

INDICAÇÃO DE STATUS	
+5VDC	LED verde indicando energização do módulo
Fail	LED vermelho indicando que existe falha no processador
Run	LED verde indicando que o programa está rodando
Hold	LED amarelo indicando que o programa está em modo de espera
Force	Não utilizado
232/485 TX	LED verde indicando a transmissão de dados EIA-232 e EIA-485
ETH 10	LED verde sinalizando cabo Ethernet conectado
ETH TX	LED verde indicando transmissão de dados pela Ethernet
SYNC	LED verde indicando sincronismo de dados
Standby	LED verde indicando que o módulo está em standby (estado de espera quando ele é redundante)

DIMENSÕES E PESO	
Dimensões (L x A x P)	39,9 x 137,0 x 141,5 (mm) 157 x 5,39 x. 5,57 (pol.)
Peso	0,340 kg

TEMPERATURA	
Operação	0 ºC a 60 ºC
Armazenamento	-30 ºC a 70 ºC

CABOS	
Um Fio	14 AWG (2 mm^2)
Dois Fios	20 AWG (0,5 mm^2)

Tabela de Configuração Mínima de Blocos para o MB-700

Função	Blocos Necessários	Número de Pontos Suportados por Bloco
BYPASS	**RESOURCE** **CCCF**	
CONCENTRADOR	**RESOURCE** **CCCF** **CCSM1** **: :** **CCSM25** **(Suporta até 25 Blocos SM)**	**1 CCSM = 96 PONTOS DISCRETOS** **08 PONTOS INTEIROS** **56 PORCENTAGEM** **16 FLOAT**
PEER-TO-PEER SERIAL, ETHERNET ou AMBOS	**RESOURCE** **CCCF** **CCCM1** **: :** **CCCM16** **(Suporta até 16 Blocos CM)**	**1 CCCM = 04 PONTOS ANALÓGICOS** **04 PONTOS DISCRETOS**

Observação: O MB-700 suporta combinação de funções como *Bypass* e Concentrador e *Peer-to-Peer*.

Tabela de Tipos de Dados Disponíveis para o Parâmetro DATA_TYPE

Nr Tipo de dado	Tipo de Dado	Descrição
1	Integer8	Inteiro sinalizado (1 Byte)
2	Integer16	Inteiro sinalizado (2 Bytes)
3	Integer32	Inteiro sinalizado (4 Bytes) - (big-*endian)*
4	Unsigned8	Inteiro sinalizado (1 Byte)
5	Unsigned16	Inteiro sinalizado (2 Bytes)
6	Unsigned32	Inteiro sinalizado (4 Bytes)) - (big-*endian)*
7	FloatingPoint	Ponto Flutuante - (big-*endian)*
8	Swapped Integer8	Inteiro sinalizado (1 Byte)
9	Swapped Integer16	Inteiro sinalizado (2 Bytes)
10	Swapped Integer32	Inteiro sinalizado (4 Bytes) –*(little-endian)*
11	Swapped Unsigned8	Inteiro sinalizado (1 Byte)
12	Swapped Unsigned16	Inteiro sinalizado (2 Bytes)
13	Swapped Unsigned32	Inteiro sinalizado (4 Bytes) –*(little-endian)*
14	Swapped FloatingPoint	Ponto Flutuante –*(little-endian)*

Obs.: Para tipo de dados de 4 bytes, o tipo de dado s*wapped* tem uma inversão do dado (*little-endian*) com relação ao tipo de dado não *swapped* (*big-endian*). Para tipo de dados de 1 e 2 bytes não há diferença entre o *swapped* e não *swapped*.
Por exemplo, considerando um número de 4 bytes 12345678 alocados nos endereços Modbus 40001 e 40002:

- Alocação dos Dados para o Tipo de Dado *Unsigned32* (*Big-endian*)

Endereço Modbus	Dado
40001	5678
40002	1234

- Alocação dos Dados para o Tipo de Dado *Swapped Unsigned32* (*little-endian*)

Endereço Modbus	Dado
40001	1234
40002	5678

Conversão de Escala

Os Blocos que possuem Conversão de Escala são listados abaixo:

- **CCSM**
  - EU_ADDRESS_A/EU_ADDRESS_B
- **CCCM**
  - EU_ADDRESS_IN1 a EU_ADDRESS_IN2
  - EU_ADDRESS_OUT1 a EU_ADDRESS_OUT2
- **CCDL**
  - EU_ADDRESS_A1

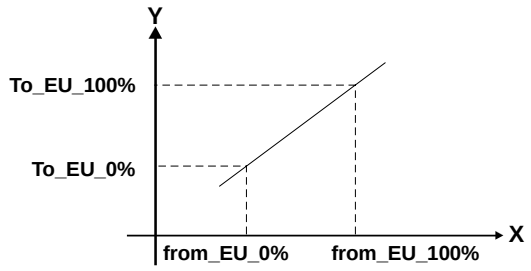
A conversão de escala para o protocolo Modbus tem dois propósitos:

- Conversão de um valor analógico Interno do Bloco para um valor Modbus expresso em unidades de engenharia;
- Conversão de um valor analógico do Modbus para o valor Interno do Bloco em unidades de engenharia.

Os parâmetros de escala são compostos pelos seguintes campos:
- FROM_EU_100% - define o maior valor da unidade de entrada (unidade atual do dado);
- FROM_EU_0% - define o menor valor da unidade de entrada (unidade atual do dado);
- TO_EU_100% - define o maior valor da unidade de saída (unidade desejada do dado);
- TO_EU_0% - define o menor valor da unidade de saída (unidade desejada do dado).

OBSERVAÇÃO
Apesar de existirem diferentes estruturas de dados que possuem conversão de escala, todas elas seguem o mesmo procedimento de conversão.

A = (TO_EU_100% - TO_EU_0%)/(FROM_EU_100% - FROM_EU_0%)
B = TO_EU_0% - A*FROM_EU_0%
Y=Ax+B



Onde x é a variável lida do dispositivo escravo de entrada e Y a variável de saída.

Estrutura de Dados para o MB-700

Estrutura de Bloco – DS-64

Esta estrutura de dados consiste em atributos de um bloco.

E	Nome do Elemento	Tipo de Dado	Tam.
1	Block Tag	VisibleString	32
2	DD MemberId	Unsigned32	4
3	DD ItemId	Unsigned32	4
4	DD Revision	Unsigned16	2
5	Profile	Unsigned16	2
6	Profile Revision	Unsigned16	2
7	Execution Time	Unsigned32	4
8	Period of Execution	Unsigned32	4
9	Number of Parameters	Unsigned16	2
10	Next FB to Execute	Unsigned16	2
11	Starting Index of Views	Unsigned16	2
12	NumberofVIEW_3	Unsigned8	1
13	NumberofVIEW_4	Unsigned8	1

Valor & Status –Estrutura Float – DS-65

Esta estrutura de dados consiste dos parâmetros de valor e status dos parâmetros float que são Entradas ou Saídas.

E	Nome do Elemento	Tipo de Dado	Tam.
1	Status	Unsigned8	1
2	Value	Float	4

Valor & Status – Estrutura Discreta – DS-66

Esta estrutura consiste de valor e status de parâmetros de valores discretos.

E	Nome do Elemento	Tipo de Dado	Tam.
1	Status	Unsigned8	1
2	Value	Unsigned8	1

Modo de Estrutura – DS-69

Esta estrutura de dados consiste em uma cadeia de bits para modos target, atual, permitido e normal.

E	Nome do Elemento	Tipo de Dado	Tam.
1	Target	BitString	1
2	Actual	BitString	1
3	Permitted	BitString	1
4	Normal	BitString	1

Estrutura de Alarme Discreto – DS-72

Esta estrutura de dados consiste nas descrições dos alarmes discretos.

E	Nome do Elemento	Tipo de Dado	Tam
1	Unacknowledged	Unsigned8	1
2	Alarm State	Unsigned8	1
3	Time Stamp	Time Value	8
4	Subcode	Unsigned16	2
5	Value	Unsigned8	1

Estrutura de Evento de Atualização – DS-73

Esta estrutura de dados consiste de dados que descrevem um alarme de revisão estático.

E	Nome do Elemento	Tipo de Dado	Tam
1	Unacknowledged	Unsigned8	1
2	Update State	Unsigned8	1
3	Time Stamp	Time Value	8
4	Static Revision	Unsigned16	2
5	Relative Index	Unsigned16	2

Estrutura de Endereço Escravo - DS-263

Esta estrutura de dados consiste em dados que informam o endereço IP e o endereço Modbus dos escravos.

E	Nome do Elemento	Tipo de Dado	Tam.
1	IP Slave1	VisibleString(16)	16
2	IP Slave2	VisibleString(16)	16
3	IP Slave3	VisibleString(16)	16
4	IP Slave4	VisibleString(16)	16
5	IP Slave5	VisibleString(16)	16
6	IP Slave6	VisibleString(16)	16
7	Slave Address1	Unsigned8	1
8	Slave Address2	Unsigned8	1
9	Slave Address3	Unsigned8	1
10	Slave Address4	Unsigned8	1
11	Slave Address5	Unsigned8	1
12	Slave Address6	Unsigned8	1

Estrutura de Endereço - DS-264

E	Nome do Elemento	Tipo de Dado	Tam.
1	Modbus Address of Value	Unsigned16	2
2	Number of Bytes	Unsigned8	1

Estrutura de Endereço - DS-265

E	Nome do Elemento	Tipo de Dado	Tam.
1	From EU 100%	Float	4
2	From EU 0%	Float	4
3	To EU 100%	Float	4
4	To EU 0%	Float	4
5	Data Type	Unsigned8	1
6	Modbus Address of Value	Unsigned16	2

Estrutura de Endereço - DS-266

E	Nome do Elemento	Tipo do Dado	Tam.
1	From EU 100%	Float	4
2	From EU 0%	Float	4
3	To EU 100%	Float	4
4	To EU 0%	Float	4
5	Data Type	Unsigned8	1
6	Port Number	Unsigned8	1
7	Slave Address	Unsigned8	1
8	Modbus Address of Value	Unsigned16	2
9	Modbus Address of Status	Unsigned16	2

Estrutura de Endereço - DS-267

E	Nome do Elemento	Tipo do Dado	Tam.
1	Port Number	Unsigned8	1
2	Slave Address	Unsigned8	1
3	Modbus Address of Value	Unsigned16	2
4	Modbus Address of Status	Unsigned16	2

Apêndice A

smar	FSR - Formulário para Solicitação de Revisão	
	MB-700	**Proposta Nº:**

DADOS DA EMPRESA

Empresa: _____

Unidade/Setor/Departamento: _____

Nota Fiscal de Remessa: _____

CONTATO COMERCIAL

Nome Completo: _____

Telefone: _____ _____ _____ _____ **Fax:** _____

Email: _____

CONTATO TÉCNICO

Nome Completo: _____

Telefone: _____ _____ _____ _____ **Ramal:** _____

Email: _____

DADOS DO EQUIPAMENTO

Modelo: _____

Número de Série: _____

INFORMAÇÕES DO PROCESSO

Tipo de processo (Ex. controle de caldeira)**:** _____

Tempo de Operação: _____

Data da Falha: _____

DESCRIÇÃO DA FALHA

(Por favor, descreva o comportamento observado, se é repetitivo, como se reproduz, etc. Quanto mais informações melhor)

_____

_____

_____

_____

OBSERVAÇÕES / SUGESTÃO DE SERVIÇO

_____

_____

_____

DADOS DO EMITENTE

Empresa: _____

Contato: _____

Identificação: _____

Setor: _____

Telefone: _____ _____ _____ _____ **Ramal:** _____

E-mail: _____ **Data:** _____/ _____/ _____

Verifique os dados para emissão de Nota Fiscal no Termo de Garantia disponível em: http://www.smar.com/brasil/suporte.asp