

Sistema Serial MachSys

- **Protocolo MODBUS RTU**
- **Comunicação em alta velocidade (115200 bps)**
- **Linha de comunicação Opto Isolada**
- **Sistema de Baixo Custo**

E²pro Engenharia

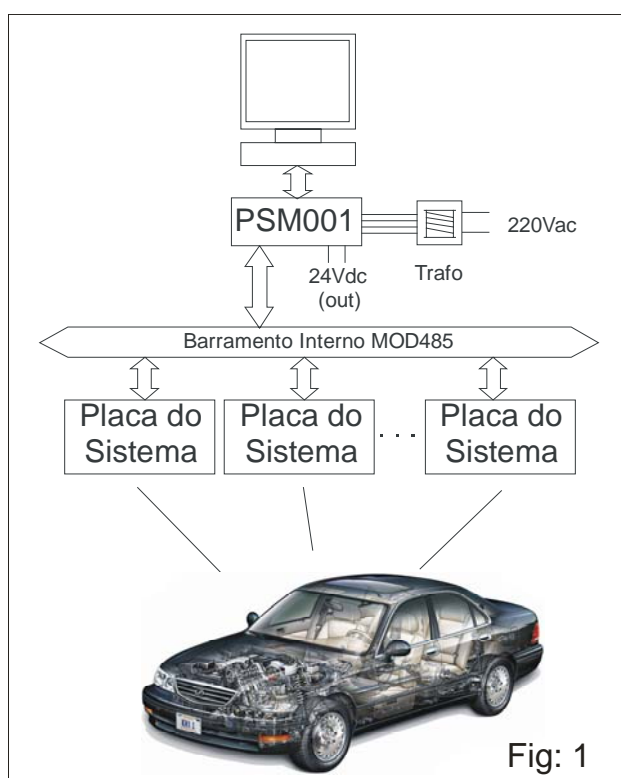
Elaborado por: Paulo Schaefer

Revisão 1.0

Introdução

O Sistema **MachSys** da E2pro é composto por uma série de Módulos padronizados que permitem a implementação de soluções diversas nas áreas de automação, aquisição de dados, testes computadorizados etc. Com tecnologia própria, os Módulos apresentam grande flexibilidade de aplicação, facilidade de uso e uma relação custo benefício bastante atraente.

Uma configuração típica de uso do **MachSys** é apresentada na figura 1:



A Placa PSM001 está presente em todas as aplicações do Sistema E2pro. Ela tem a função de alimentar as demais placas do Sistema, além de permitir a comunicação entre o Computador de controle e as placas. O sistema tem capacidade de endereçamento de até 64 placas, mas em função do consumo médio das placas, o número máximo de placas numa mesma aplicação fica limitado a 12. se for necessário um maior número de placas, pode-se expandir a capacidade com a utilização de uma segunda ou mais placas de alimentação.

Atualmente o Sistema conta com os seguintes Módulos:

- PSM001** – Alimentação e Comunicação com o PC
- DSM001** – I/O Digital 8 entradas 8 saídas
- DSM002** – I/O Digital 32 entradas 32 saídas
- DSM003** – 8 saídas Up, Down ou TriState 1A
- ASM001** – Módulo para Teste de Estanqueidade
- ASM002** – Aquisição 16 bits de resolução 16 canais ou 8 diferenciais
- ASM003** – Aquisição com memória 12 Bits 40Ks 4 canais
- FSM002** – Leitura de tempo com trigger por tensão 2 canais
- FSM006** – Sintetizador de Senóide 100Khz
- MSM001** – Driver para Motor de Passo
- LSM001** – Módulo de Resistência Programável DownSide
- PSM008** – Fonte Programável 0 a 30V 1mV de resolução 4A

As características básicas do Sistema são:

- **Protocolo de Comunicação:** o sistema adota o formato MODBUS RTU, com velocidade de comunicação em 115200bps.
- **Software de Controle:** cada placa possui um driver de controle que pode ser fornecido como objeto OCX ou como uma DLL. Isto permite o controle por qualquer tipo de linguagem que aceite estes dois recursos, como Visual Basic, LabView, Visual C entre outros.
- **Montagem:** Cada placa é fornecida montada em uma base de fixação, permitindo a montagem em trilhos de 35mm.

- **Conexão dos Módulos:** a conexão entre os Módulos é feita por um conjunto de cabos Flat de 14vias. Estes cabos providenciam a alimentação dos Módulos e sinal de comunicação. Desta forma a montagem de uma aplicação é bastante rápida pois após a conexão dos Módulos e alimentação com a PSM001 o sistema está praticamente pronto para uso.

1. Protocolo MODBUS – RTU e MOD485B

2.1 – Conceitos Básicos

O protocolo MODBUS é amplamente utilizado na indústria para a comunicação entre Módulos que suportam este protocolo.

A comunicação por este protocolo utiliza a técnica Master-Slave. Nesta técnica um Módulo denominado Master é o único habilitado a iniciar a comunicação com os demais Módulos do sistema, denominados Slave. O sistema só permite 1 Módulo Master e os Módulos Slave são diferenciados entre si por endereços próprios. O Módulo Master pode endereçar cada Módulo Slave individualmente ou pode endereçar todos simultaneamente, mandando uma mensagem do tipo BroadCast, entendida por todos, independente do endereço próprio.

O frame enviado pelo Módulo Master segue uma arquitetura padrão definida a seguir

Posição	Tipo	Valor
01	END	XXh
02	FUNÇÃO	XXh
03	DADOS 1	XXh
.	DADOS 2	XXh
.	.	XXh
.	DADOS n	XXh
.	CRC high	CALC
n + 4	CRC low	CALC

Sendo:

END: define o endereço do Módulo Slave acessado

FUNÇÃO: permite a definição de até 255 funções diferentes para cada Módulo Slave

DADOS: byte de dados necessários à execução da Função solicitada

CRC high e CRC low: dois bytes de verificação de erro calculados automaticamente de acordo com os bytes contidos no frame transmitido.

No Protocolo MOD485 da E2pro O frame de resposta, enviado pelo Módulo Slave para o Master, segue a seguinte arquitetura:

Posição	Tipo	Valor
01	END	XXh
02	FUNÇÃO	XXh
03	DADOS 1	XXh
.	DADOS 2	XXh
.	.	XXh
.	DADOS n	XXh
.	CRC high	CALC
n + 4	CRC low	CALC

Sendo:

END: repete o endereço do módulo para identificar quem está respondendo

FUNÇÃO: pode assumir dois valores:

- **em caso de resposta positiva:** função requisitada pelo master + 40h. Exemplo, se o Master requisitou a função 34h, o byte de função do frame de resposta valerá 74h

- **em caso de resposta negativa:** o byte de função assumirá o valor 3Fh;

DADOS: bytes contendo os dados de resposta. Em caso de resposta negativa, será enviado apenas 1 byte com o código do erro detectado;

CRC high e CRC low: dois bytes de verificação de erro calculados automaticamente de acordo com os bytes contidos no frame transmitido.

Dentro do Protocolo MODBUS, podem ser adotadas duas formas de transmissão de byte possíveis, ASCII ou RTU. Na opção ASCII, cada byte de 8 bits é transmitido como dois caracteres da tabela ASCII. Por exemplo o número 109 decimal, é representado como 6Dh e transmitido em duas etapas pela linha. Primeiro o “6” na tabela ASCII e em seguida “D”. No formato RTU ele é transmitido de uma só vez como “01101101” que é a forma binária para o número 109.

O MOD485B da E2pro adota o formato RTU.

Cada frame de comunicação no MODBUS deve ser transmitidos de forma contínua, sem intervalo entre os bytes. O protocolo MODBUS define um período chamado de Silent interval (intervalo de silêncio) correspondente a 3,5 caracter. Se o Módulo Slave detecta um intervalo de tempo superior a 3,5 caracter sem transmissão na linha, ele assume automaticamente que o próximo byte recebido é o de endereço de um novo frame. O intervalo de silêncio pode ser calculado pela fórmula:

$$T_{si} = (1 / \text{baudrate}) * 11 * 3,5$$

Como exemplo, para 115200bps:

$$T_{si} = (1 / 115200) * 11 * 3,5 = 334,2 \mu S$$

2.2 – Cálculo do CRC

Os dois últimos bytes transmitidos pela linha de comunicação formam um registro de 16 bits com o valor final calculado para o CRC (Cyclical Redundancy Check). O CRC é calculado a partir de todos os bytes transmitidos pelo frame, excluindo os próprios bytes de CRC, tomando como base apenas os 8 bits de informação de cada byte, excluindo start e stop bit e parity bit. A rotina de cálculo do CRC é a mesma nos dispositivos Master e Slave. Desta forma durante a transmissão, o dispositivo transmissor calcula os bytes de CRC e anexa o resultado ao final do frame, transmitindo primeiro o byte LSB. O dispositivo que está recebendo a mensagem, efetua o mesmo cálculo sobre os bytes recebidos e compara o resultado com os dois últimos bytes recebidos. Se os valores forem diferentes um erro é gerado.

2.3 – Sistema de Endereçamento dos Módulos

Os Módulos do Sistema MOD485B são fornecidos de fábrica com endereços pré definidos.

Entretanto, se for necessário definir um novo endereço para uma placa (por exemplo quando a aplicação necessitar de mais de uma placa do mesmo modelo),isto é possível com a rotina de configuração das placas.

O endereço do Módulo Slave pode assumir qualquer valor entre 01h e FEh.

Uma **Interface de Configuração** específica, desenvolvida em Visual Basic, permite a definição do endereço de uso do Módulo, bem como a velocidade da comunicação RS232. Para tanto deverão ser efetivadas as seguintes ações:

- Acionar a Interface de Configuração com apenas o Módulo a ter o endereço alterado, conectado ao sistema;
- A Interface de Configuração detectará a presença do Módulo e solicitará o novo endereço e velocidade de comunicação. Os valores serão testados e se OK o Módulo estará programado;
- Os passos acima deverão ser repetidos para cada Módulo novo cujo o endereço deva ser trocado antes do uso.

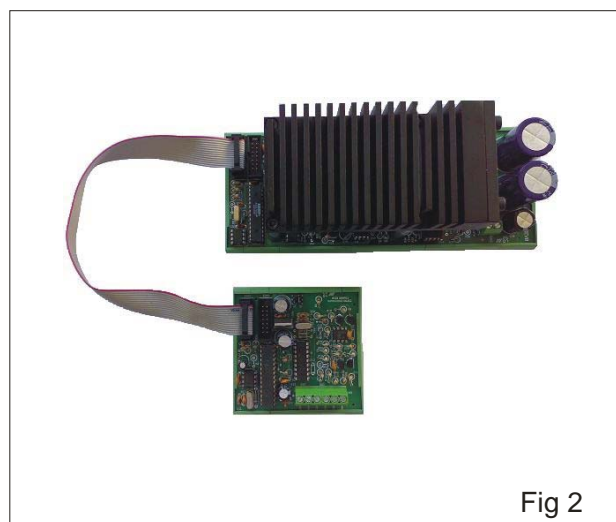


Fig 2

A figura 2 mostra a forma de conexão entre as placas do Sistema Modular da E2pro.

