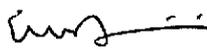


1. Publicação nº <i>INPE-2733-RPE/432</i>	2. Versão	3. Data <i>Maio, 1983</i>	5. Distribuição <input type="checkbox"/> Interna <input checked="" type="checkbox"/> Externa <input type="checkbox"/> Restrita
4. Origem <i>DCA</i>	Programa <i>REDACE/PSDA</i>		
6. Palavras chaves - selecionadas pelo(s) autor(es) <i>REDE DE COMUNICAÇÃO DE DADOS MICROPROCESSADOR MULTIPROCESSAMENTO</i>			
7. C.D.U.: <i>681.324:681.322:621.38</i>			
8. Título <i>MODELO DE UMA INTERFACE DE COMUNICAÇÃO UTILIZANDO MULTIMICROPROCESSAMENTO</i>		<i>INPE-2733-RPE/432</i>	10. Páginas: <i>26</i>
			11. Última página: <i>19</i>
9. Autoria <i>Mauro Hissao Hashioka</i>			12. Revisada por  <i>Eduardo W. Bergamini</i>
Assinatura responsável 			13. Autorizada por  <i>Nelson de Jesus Parada</i> Diretor
14. Resumo/Notas <i>Um modelo de interface de comunicação para aplicações em redes de comunicação de dados é apresentado. O modelo está baseado na utilização simultânea de vários microprocessadores, com arquitetura configurada para processamento distribuído.</i>			
15. Observações			

ABSTRACT

A model of communication interface to be applied in data communication networks is presented. It is based on the simultaneous utilization of multiple microprocessors, with an architecture configured for distributed processing.

SUMÁRIO

	<u>Pág.</u>
LISTA DE FIGURAS	<i>iii</i>
1 - <u>INTRODUÇÃO</u>	1
2 - <u>ARQUITETURA DO MODELO</u>	3
2.1 - Configuração inicial	3
2.2 - Configuração adotada	5
3 - <u>DESCRIÇÃO DOS MÓDULOS</u>	7
3.1 - Supervisor	7
3.2 - Porta externa	8
3.3 - Barramentos do MCR	11
3.4 - Porta interna	13
4 - <u>CONEXÃO ENTRE MCRS</u>	15
5 - <u>COMUNICAÇÃO INTERNA AO MCR</u>	17
6 - <u>CONCLUSÕES</u>	17
REFERÊNCIAS BIBLIOGRÁFICAS	18

LISTA DE FIGURAS

	<u>Pág.</u>
1 - Utilização do processador de comunicação	1
2 - Aplicação da interface de comunicação	3
3 - Arquitetura inicial da interface de comunicação	4
4 - Arquitetura do Multiprocessador de Comunicação em Rede (MCR) ..	6
5 - Estrutura do Supervisor (SV)	8
6 - Estrutura da Porta Externa (PE)	10
7 - Relacionamento entre MCRBUS, SV e portas	13
8 - Estrutura da Porta Interna (PI)	14
9 - Conexão entre os MCRs	16

•
• •

1. INTRODUÇÃO

Com a evolução da tecnologia nos campos da computação e das comunicações, é natural que surjam as redes de computadores para permitir que informações disponíveis em um centro possam ser utilizados por usuários de outros centros. Da mesma forma que as pessoas se comunicam através de uma linguagem que obedece a certas regras sintáticas e semânticas, os computadores também devem ter regras de comunicação que tornem possíveis as trocas de informações entre eles. Essas regras ou normas, chamadas protocolos de comunicação, têm as seguintes finalidades: inicializar, desconectar e manter um enlace físico entre dois nós; deixar esse enlace físico livre de erros; fazer o encaminhamento das mensagens; controlar o fluxo de dados, etc. (Kleinrock, 1976; Tanenbaum, 1981; Tarouco, 1979).

Portanto, é necessário que haja um processamento dedicado ao serviço de troca de informações. Para evitar que o computador hospedeiro de cada nó se encarregue de executar esse tipo de serviço, é desejável que haja um processador com essa finalidade. Este é comumente chamado processador de comunicação ("front-end") (Figura 1) (Tarouco, 1979).

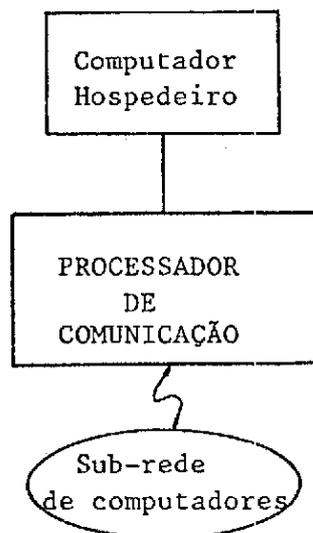


Fig. 1 - Utilização do processador de comunicação.

Devido a este tipo de problema, surgiu a idéia de desenvolver uma interface de comunicação com capacidade de processamento, a qual inclui as seguintes funções principais:

- a) Executar o programa que implementa o protocolo de comunicação nos níveis mais baixos (níveis 1, 2 e 3).
- b) Ter a capacidade de armazenamento temporário das mensagens ("buffering").
- c) Permitir a ligação entre vários nós da rede.
- d) Permitir a implementação de vários tipos de protocolos de comunicação (X.25, SDLC, HDLC, PRIME, NASCOM, etc.).
- e) Ter a capacidade de funcionar independentemente do computador hospedeiro.
- f) Fazer a estatística do fluxo de dados que fluem através dela.

A idéia inicial seria desenvolver uma interface de comunicação que pudesse ser também utilizada num nó de uma rede de comunicação, e não somente como uma interface de acesso à rede. Essa idéia foi um dos requisitos observados na configuração adotada na Seção 2.2. A Figura 2 mostra uma aplicação dessa interface.

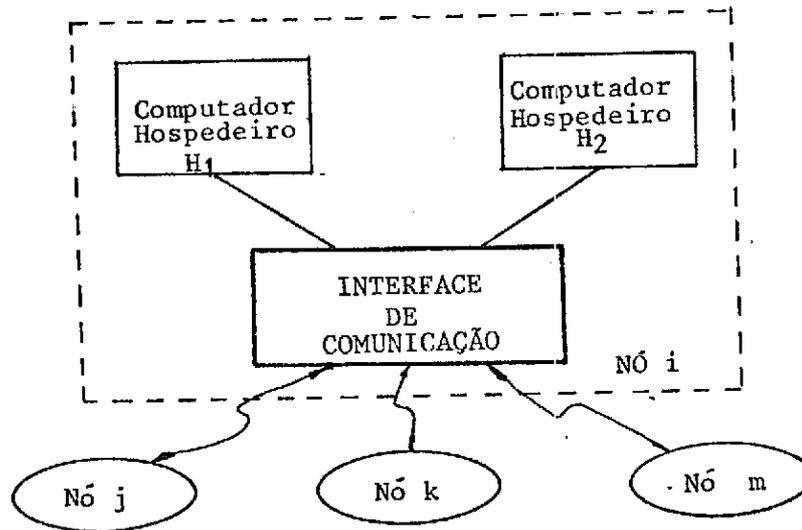


Fig. 2 - Aplicação da interface de comunicação.

2. ARQUITETURA DO MODELO

2.1 - CONFIGURAÇÃO INICIAL

Inicialmente pensou-se em uma interface de comunicação com um processador para atender várias linhas seriais e, além disso, fazer a conexão da interface paralela com o computador que controla o nó. Dessa forma a Figura 3 descreve a arquitetura inicialmente proposta.

As interfaces seriais não teriam processamento, porêm seriam compostas de memórias para armazenagem temporária de dados.

A comunicação entre as interfaces seriais seriam feitas através do barramento interno-BI e controladas por um único processador.

As transferências dos dados propriamente ditos seriam feitas por DMA (Acesso Direto a Memória) para permitir alta taxa de transferência de dados e menor tempo de processamento.

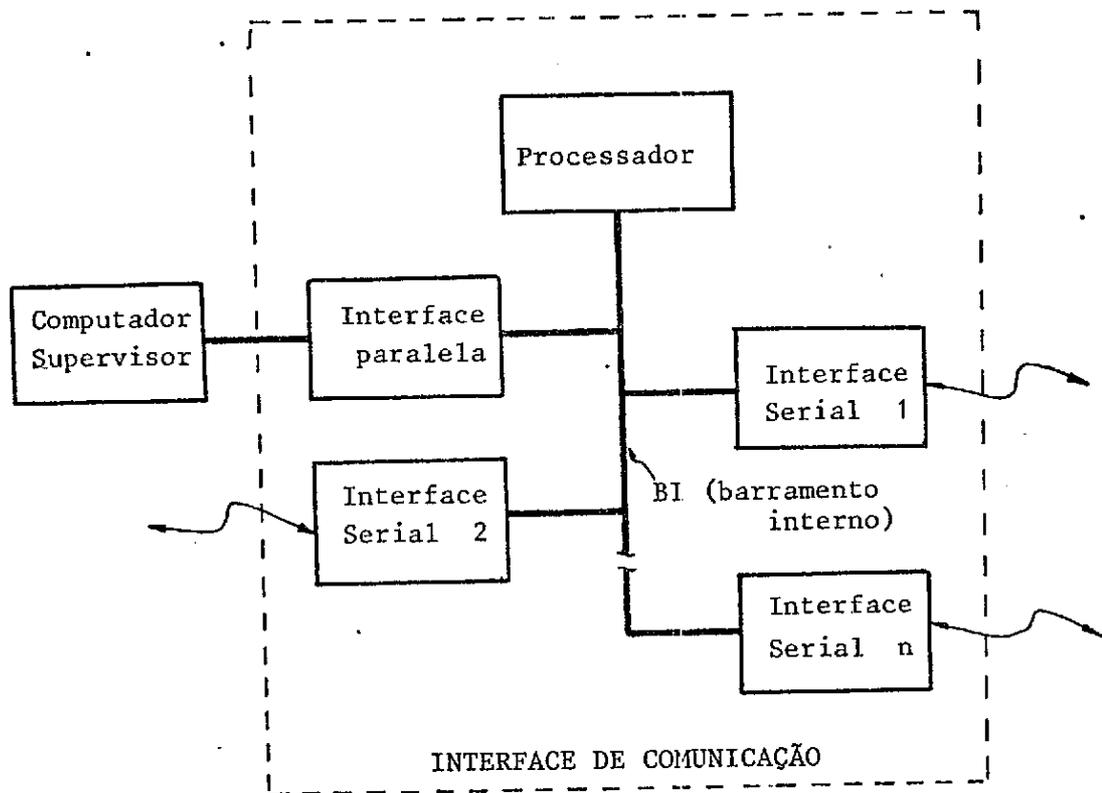


Fig. 3 - Arquitetura inicial da interface de comunicação.

A interface com o computador controlador do \bar{n} seria do tipo paralela para ser mais veloz que as interfaces seriais.

Desistiu-se de implementar essa configuração inicial devido aos inconvenientes abaixo relacionados:

- a) Deveria ser usado um processador com alta velocidade de processamento, ou ele deveria ter um número muito baixo de interfaces seriais.
- b) O interfaceamento com o barramento interno (BI) seria bastante complexo para haver troca eficiente de dados entre as interfaces seriais. Provavelmente deveria ser adotada uma estrutura microprogramada.

- c) Haveria necessidade de implementar um sofisticado sistema operacional para atender todas as interfaces em termos de protocolos de comunicação.

Pensou-se então em obter uma configuração análoga, mas com recurso de processamento mais poderoso. A seção seguinte descreve a configuração adotada.

2.2 - CONFIGURAÇÃO ADOTADA

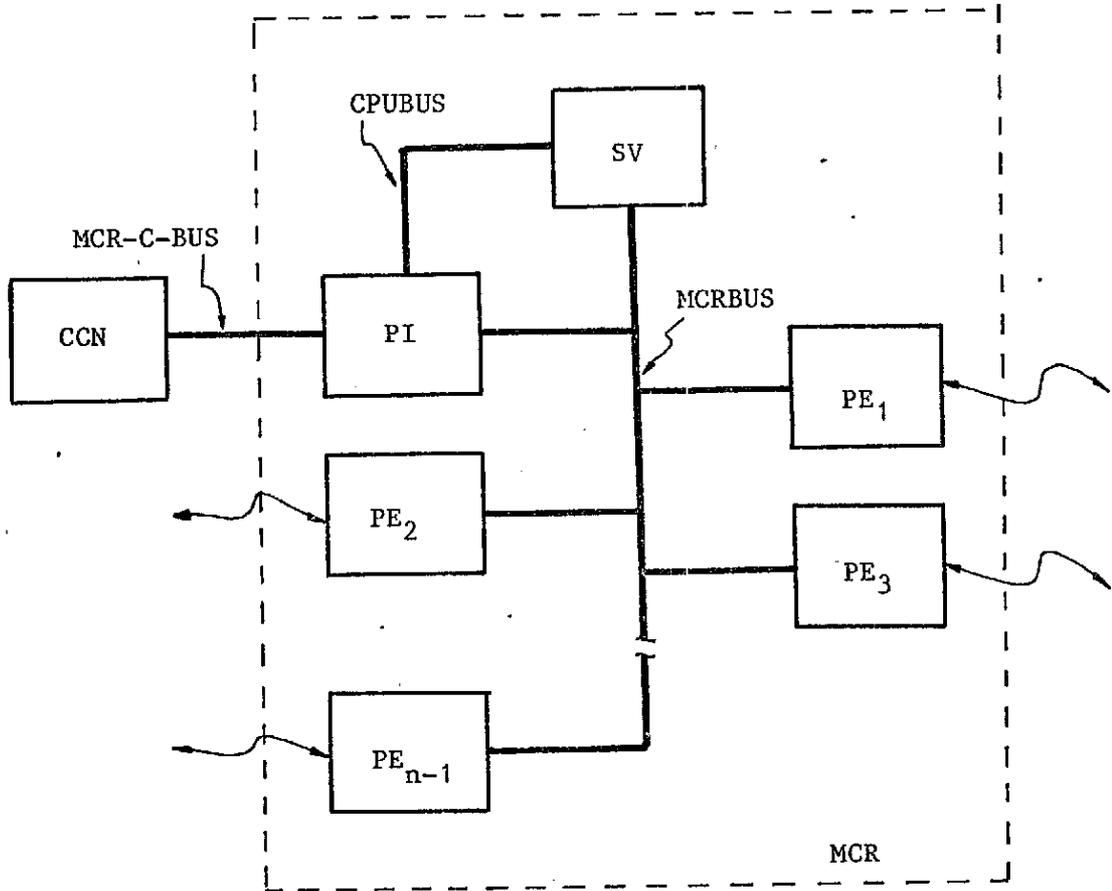
O propósito básico é utilizar vários microprocessadores, uma vez que são extremamente versáteis e têm custo relativamente baixo. Além disso, existe flexibilidade de expansão devido à modularidade de seu "hardware" e de sua programação, características importantes como requisitos de um projeto.

Dessa forma, a configuração adotada neste trabalho é composta de vários microprocessadores, cada um atendendo praticamente a uma das linhas seriais, e de um outro para fazer a supervisão e controle da troca de informações entre eles.

A essa interface de comunicação deu-se o nome de Multiprocessador de comunicação em Rede (MCR), que está apresentada na Figura 4.

A grande diferença em relação à configuração inicial está nas suas portas externas (PEs), pelo acréscimo de uma estrutura com microprocessador.

As vantagens que se destacam ao adotar essa filosofia são: circuitos mais simples para fazer o interfaceamento com o MCRBUS (barramento interno do MCR) para troca de informações e passagem de dados entre as portas; estruturada de programação mais simples em cada processador; maior capacidade de processamento e possibilidade de protocolos de comunicação diferentes em cada Porta Externa (PE).



- MCR: Multiprocessador de Comunicação em Rede
- CCN: Computador Controlador do Nó
- SV: Supervisor
- PI: Porta Interna
- PE: Porta Externa
- MCRBUS: barramento padrão do MCR.
- CPUBUS: barramento padrão para as CPUs
- MCR-C-BUS: barramento padrão para ligação de MCRs com CCN

Fig. 4 - Arquitetura do Multiprocessador de Comunicação em Rede (MCR)

3. DESCRIÇÃO DOS MÓDULOS

3.1 - SUPERVISOR

A configuração adotada prevê, como meio de comunicação entre portas, o barramento MCRBUS. O supervisor (SV) tem como função principal o controle desse barramento, devendo através dele se comunicar com as portas externas - PEs. Dessa maneira evita-se o problema de congestionamento e perda de informações.

A segunda tarefa do supervisor (SV) é o controle da Porta Interna (PI) para comunicação como o computador de controle do nó.

Além dessas funções principais, o supervisor deve fazer a diagnose e até mesmo a auto-diagnose do MCR, para aumentar a confiabilidade do sistema MCR.

A Figura 5 mostra a estrutura do supervisor (SV) o qual é composto de um microprocessador de 8 bits e de memórias EPROM e RAM somente para cumprir as tarefas acima mencionadas. Tais memórias não se destinam à armazenagem temporária de informação. O temporizador fornece a velocidade para a interface serial, além de gerar interrupção para "timeouts" ou mesmo para as rotinas de diagnose. A interface serial tem sua aplicação na fase de teste para ter acesso ao monitor através de um terminal de vídeo (esse monitor serve para auxiliar na depuração de programas). Na fase de funcionamento normal, essa interface serial pode ser utilizada como mecanismo para gerar o estado do MCR e se comunicar com o operador, caso necessite.

A interface para a PI é do tipo controlador por "3-state", pois o barramento CPUBUS também é padrão e os sinais são os mesmos utilizados em outros módulos que contêm o mesmo tipo de processador no MCR.

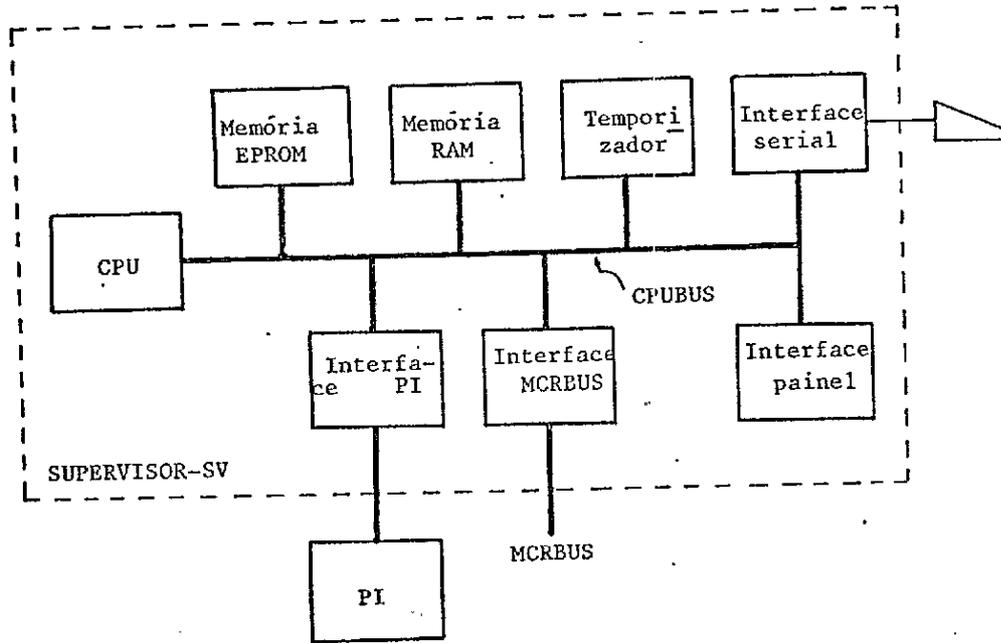


Fig. 5 - Estrutura do Supervisor (SV).

A interface para o MCRBUS também é do tipo "3-state" e será descrita melhor na Seção 3.3.

Para efeito de aumento da confiabilidade, poder-se-ia pensar num supervisor com dois processadores, cada um com suas memórias, já que o não-funcionamento deste módulo irá acarretar uma parada do sistema.

3.2 - PORTA EXTERNA

A porta externa (PE) é na verdade o módulo que satisfaz o objetivo final do MCR, que é implementar o protocolo de comunicação da rede.

Prevê-se a capacidade de armazenamento temporário de pacotes de informação, a qual é uma das características básicas dos protocolos para comutação por pacotes.

A rápida transferência de pacotes de uma porta para outra é um requisito básico para este módulo.

Pensando-se na modularidade de placas de circuitos integrados, a porta externa (PE) é dividida em duas partes: a CPU e a interface.

A parte da CPU tem uma estrutura igual à utilizada pelo supervisor, de tal forma que ao confeccionar a placa de circuitos integrados a CPU servirá tanto para o supervisor como para a porta externa.

A parte da interface prevê o interfaceamento da PE com o MCRBUS, bem como a interface serial para a linha. A Figura 6 descreve a porta externa.

As memórias do bloco da CPU servem também para o armazenamento temporário dos pacotes.

O temporizador da CPU atende ao mesmo objetivo apresentado para o caso do supervisor do MCR. A interface serial também serve para testes. Portanto quando a PE estiver em ação, poderá apresentar relatórios referentes a sua operação através de um terminal de vídeo.

O temporizador gera velocidade (base de tempo) para a interface serial de comunicação de dados, e ainda pode ser aproveitado para gerar interrupção caso a PE necessite de vários "timeouts".

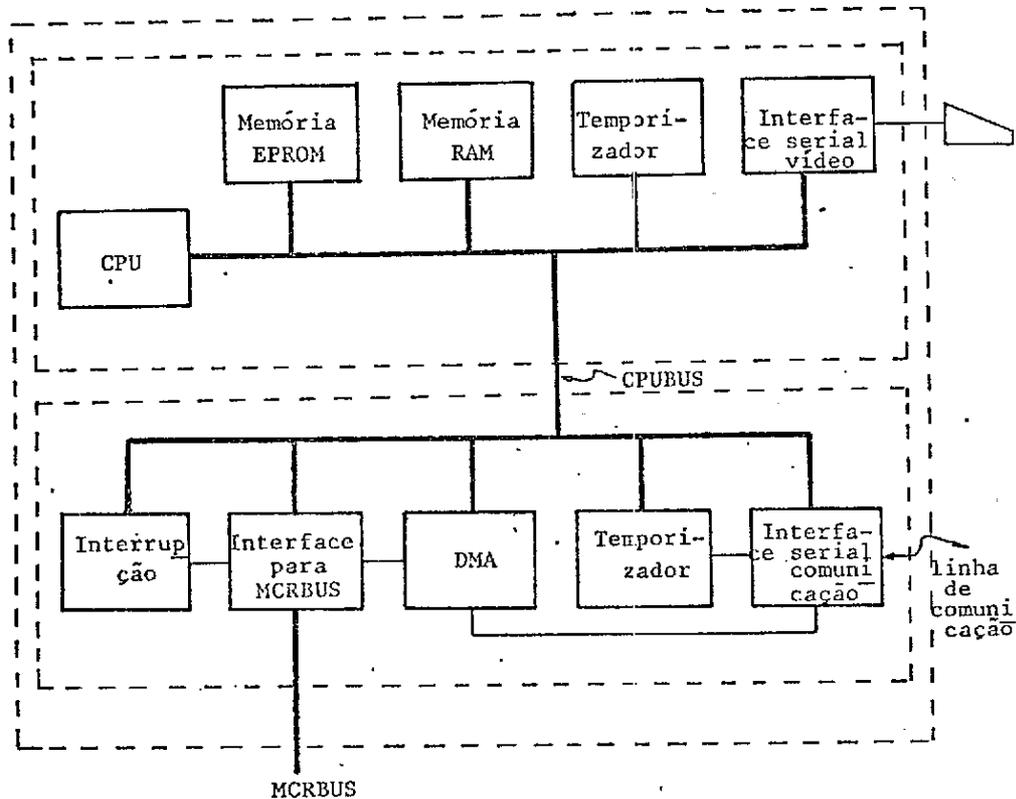


Fig. 6 - Estrutura da Porta Externa (PE).

A interface serial de comunicação pode ser de dois tipos básicos. A primeira atende as necessidades dos protocolos de comunicação SDLC, HDLC e X.25. Suas características básicas são: "full-duplex" ou "half-duplex até 64K de "baud-rate", geração e cheque automático de CRC ("Cyclic Redundancy Check"), comunicação síncrona ou assíncrona, e comandos a níveis de quadros (pacote ou pacotes e o cabeçalho). A segunda é a USART, normalmente utilizada como interface serial. Compõem-se também das características básicas como comunicação síncrona ou assíncrona, "full-duplex" ou "half-duplex", porém não tem circuito mais sofisticado para geração e cheque de CRC, assim como de comandos a níveis de quadros.

O DMA é o mecanismo de acesso direto à memória e está ligado com a interface serial de comunicação para que esta não sacrifique o seu recurso de processamento. Ou seja, o processo de transmissão ou recepção de pacotes é ativado para o controle do DMA, de tal forma que o acesso à memória é feito por roubo de ciclo. Além disso, a transmissão de dados pelo MCRBUS também é feita por DMA para obter maior velocidade de transferência.

A interface para o MCRBUS da PE cuida da sincronização de sinais e possibilita a correta transferência de dados por este barramento. Eletricamente ela é composta de "3-state" para isolar as PEs (ou a PI) que fazem parte do processo de transferência em dado momento. A troca de informações com o SV pelo MCRBUS é feita através de registros de escrita e de leitura. Na seção seguinte há uma descrição melhor do funcionamento desses registros. Um dos pontos importantes enfocados nessa interface é a sua modularidade. A placa da porta externa é padrão, ou seja, ela se identifica com PE₁, PE₂, etc.

O bloco de interrupção da PE atende as interrupções geradas pelo: recebimento e final de transmissão de um pacote controlado pela interface serial de comunicação; final de transferência de pacote pelo MCRBUS; troca de informações pelos registros da interface do MCRBUS; e temporizador, no caso de esse funcionar como "timeout".

3.3 - BARRAMENTOS DO MCR

A estrutura do modelo necessita de dois barramentos padrões: o primeiro, o CPUBUS, é aquele definido para qualquer configuração onde haja utilização de um microprocessador. Há necessidade desse barramento porque a porta externa, ou mesmo o supervisor, não cabe fisicamente num só placa de circuito. A vantagem da definição desse barramento está no fato de que em cada módulo pode haver expansão através da criação de placas de memória ou periféricos, sem necessidade de perder a modularidade do sistema. Esse barramento não serve para interligação entre CPUs.

O segundo, o MCRBUS, é um barramento definido especificamente para o MCR. É através dele que ocorrem trocas de informações entre processadores, além da transferência de pacotes de dados de uma porta para outra.

A idéia do MCRBUS é definir os sinais de tal forma que o posicionamento físico das placas do MCR não seja rígido.

Praticamente o MCRBUS é dividido em duas partes: uma onde o SV controla os sinais, a outra por onde se dá propriamente a transferência de dados.

O SV controla o MCRBUS através de sinais de endereço, dados, escrita leitura e interrupções, enquanto a transferência de dados propriamente dita é realizada pelos sinais de dados (diferentes dos anteriores), DMA, escrita e sincronização.

A Figura 7 mostra o relacionamento do MCRBUS com o SV e as portas.

Nota-se que há independência entre os dados para o SV e os dados para transferência entre portas. Dessa forma a comunicação entre SV e uma porta pode estar ocorrendo simultaneamente com a transferência de dados controlados por DMA.

É importante frisar que o DMA da porta fonte de dados é que controla o processo de transferência de dados pelo MCRBUS.

Esses sinais para transferência de dados são protegidos por "3-state" liberados para o MCRBUS, na direção correta, somente quando forem autorizados tanto pelas portas envolvidas no processo como pelo SV.

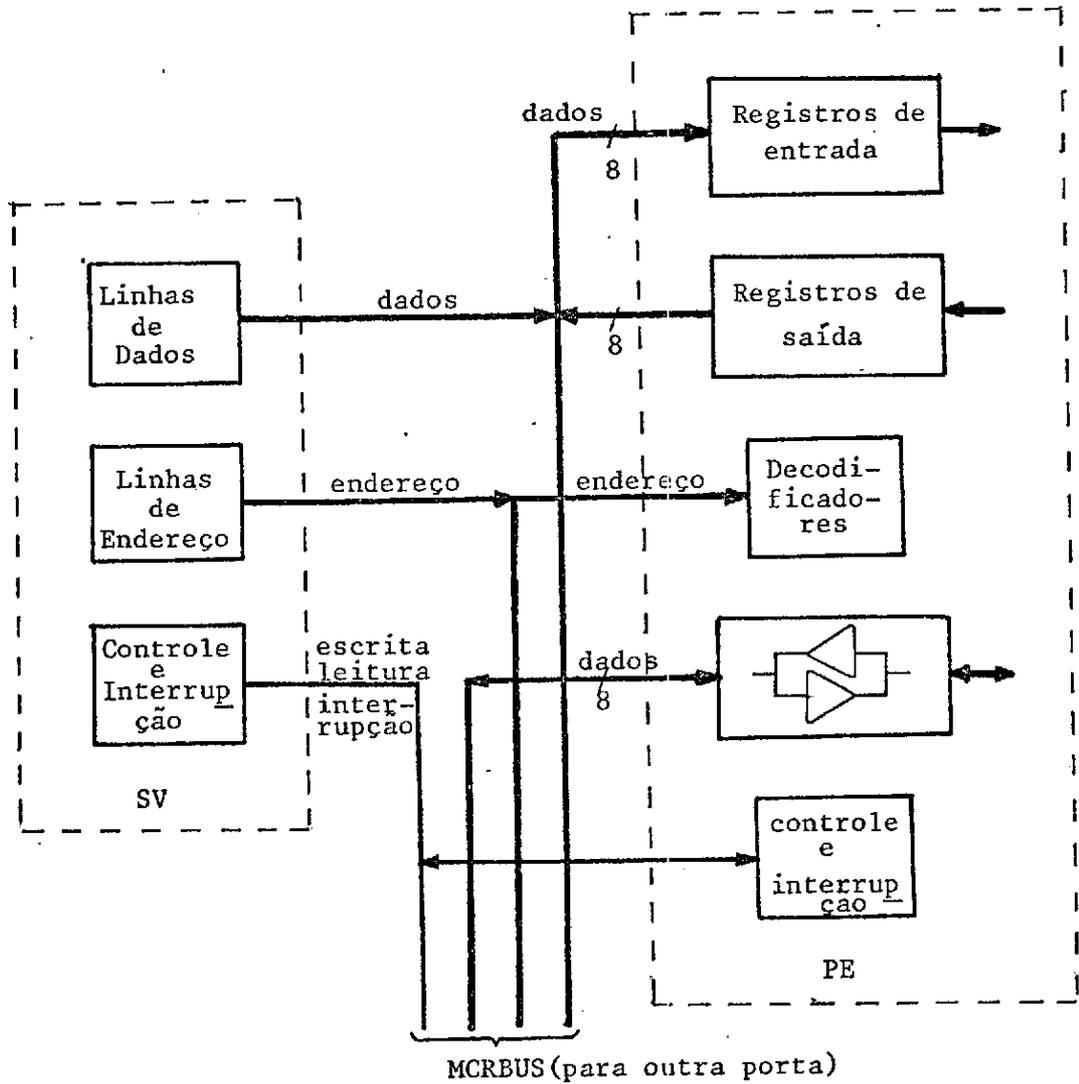


Fig. 7 - Relacionamento entre MCRBUS, SV e portas.

3.4 - PORTA INTERNA

A porta interna (PI) é uma interface que serve para fazer a conexão com o computador controlador do n^o e fazer a interconexão entre os MCRs.

A PI é composta basicamente de memória para armazenamento temporário dos pacotes de dados que fluem entre os MCRs ou entre o MCR e o computador controlador de n^o.

Para a PI também é desejável que a transferência de dados seja realizada com alta velocidade.

A PI não necessita muito de processamento, uma vez que o seu protocolo de comunicação é muito mais simples do que um protocolo de comunicação de rede do tipo X.25. Portanto fica justificada a utilização do supervisor como controlador da PI. Além disso, é através da PI que se dá conhecimento do estado ("status") do MCR ao computador controlador do n^o.

A Figura 8 mostra a estrutura da porta interna (PI).

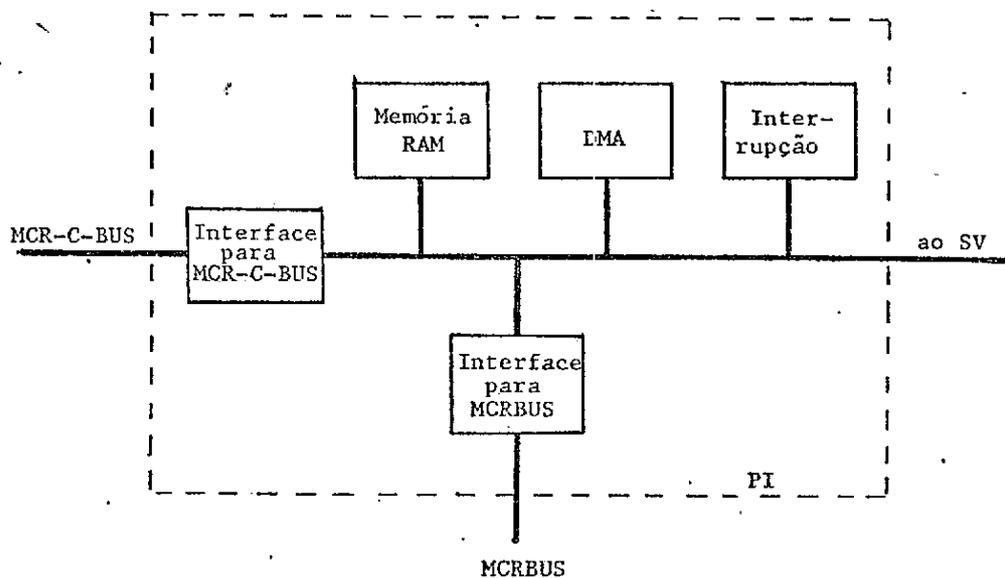


Fig. 8 - Estrutura da Porta Interna (PI).

A memória RAM atua como armazenador temporário de pacotes e é passível de acesso direto pelo SV, como se ela fizesse parte de sua memória principal.

O DMA tem função importante na transferência de dados através dos barramentos. Ele continua com a mesma função de atuar somente quando os dados são transmitidos da PI para qualquer outro destino.

A interrupção atende basicamente a interface com o MCR-C-BUS (barramento de interconexão entre os MCRs e o computador controlador do nó) durante a troca de informações com o computador controlador do nó.

A interface com o MCRBUS, nesse caso, é mais simples do que a utilizada nas PEs, pelo fato de o processador da PI ser o próprio do SV.

Com o objetivo de padronizar os barramentos, evitando duplicação no desenvolvimento do projeto de interfaces e de programação dos protocolos de comunicação, o barramento MCR-C-BUS é exatamente igual ao barramento MCRBUS. Dessa forma, a interface da PI com MCR-C-BUS é muito semelhante à interface da PE com o MCRBUS. A memória RAM da PI pode, portanto, ser operada por três barramentos.

4. CONEXÃO ENTRE MCRS

Uma das funções da PI é possibilitar a conexão entre os MCRs (Figura 9). Conforme já foi dito, os barramentos MCRBUS e MCR-C-BUS são iguais. Portanto, em termos de protocolo de comunicação há uma equivalência entre a PE e o SV, ou seja, a programação para atender o MCR-C-BUS pelo SV é a mesma utilizada pela PE para atender o MCRBUS. Além disso, a programação do controlador de MCRs (Figura 9) para controlar o MCR-C-BUS é a mesma utilizada pelo SV para controlar o MCRBUS.

A configuração mostrada na Figura 9 objetiva a troca de informações mais rápidas entre os MCRs; porém, é também possível interligar os MCRs utilizando as linhas seriais das PEs.

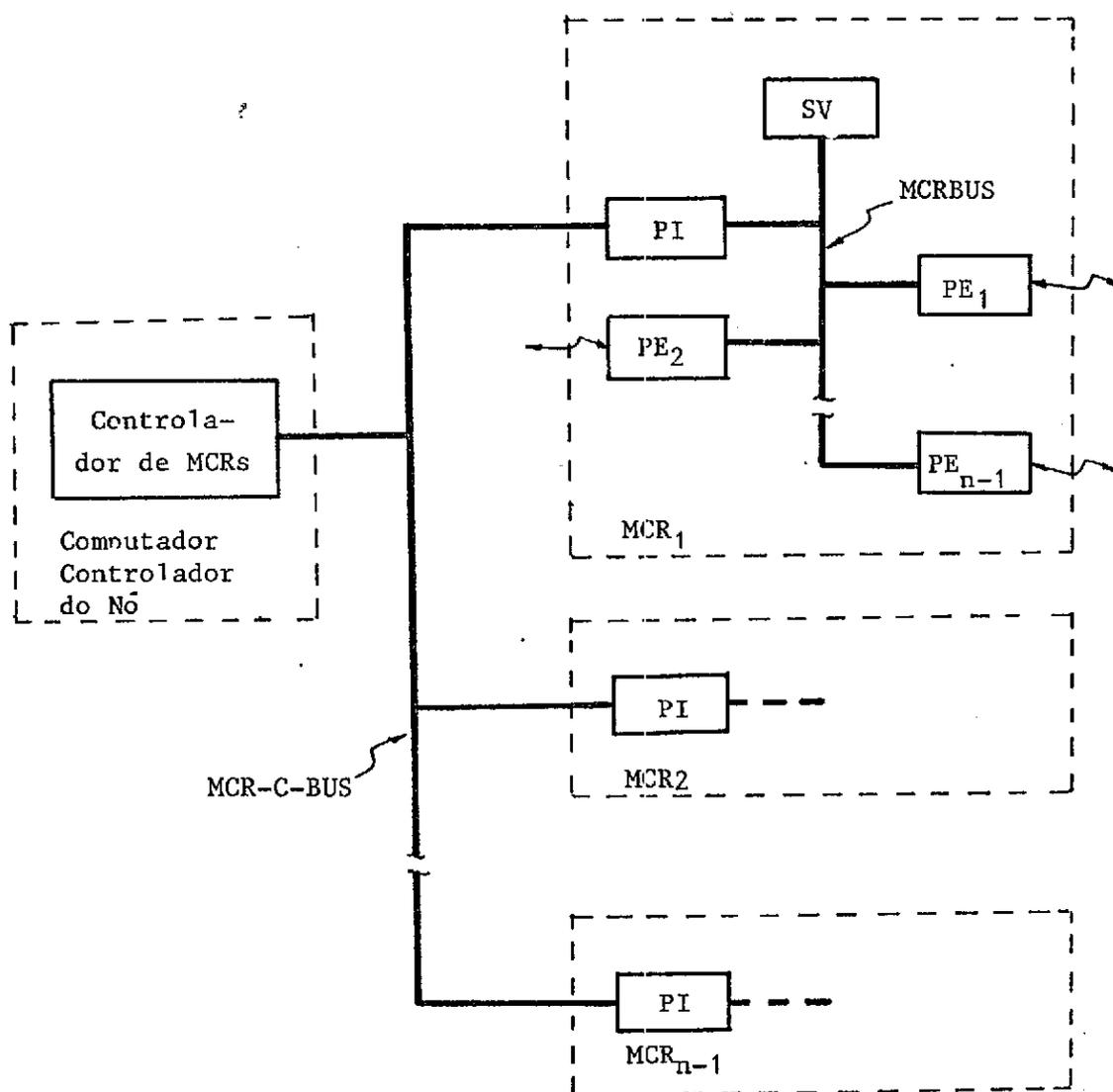


Fig. 9 - Conexão entre os MCRs

5. COMUNICAÇÃO INTERNA AO MCR

Toda comunicação interna entre os principais elementos funcionais do MCR é realizada com o auxílio do SV, que atua como intérprete para a conversa entre duas portas quaisquer do MCR. Todo processo de comunicação interna é baseado em interrupções.

Se uma PE quer enviar um pacote de dados a outra PE, o processo se inicia com a PE fonte escrevendo nos seus registros de saída para o MCRBUS. Isso gera uma interrupção que o SV atende verificando quem interrompeu e lendo os registros onde a PE fonte escreveu. Após o SV localizar a PE de destino, ele escreve nos registros de entrada da PE de destino e gera nesta uma interrupção. Essa PE atende a interrupção lendo esses registros e, então prepara-se para receber o pacote escrevendo nos seus registros de saída. Isso gera uma interrupção que o SV atende lendo esses registros. No caso mais geral, a PE de destino poderia informar que não seria possível realizar a transferência de dados. Em seguida o SV escreve em um registro de habilitação de cada PE, de uma só vez, o que torna disponível o caminho por onde fluirão os dados no MCRBUS. Em seguida, o SV avisa a porta fonte para disparar o processo de transferência do pacote controlado pelo DMA da PE fonte. No final dessa transferência são geradas as interrupções para o SV e para as PEs em questão, para avisar que a transferência de dados pelo MCRBUS foi concluída.

6. CONCLUSÕES

O modelo apresentado neste trabalho está sendo construído pelo Programa de Sistemas Digitais e Analógicos do INPE, para ser utilizado nos projetos RECODI e REDACE do mesmo programa. O projeto prevê a construção do MCR com até 8 portas. Atualmente um protótipo deste equipamento encontra-se em fase de teste.

A utilização de vários microprocessadores facilitou tanto a organização do "hardware" como a estruturação do "software". A modularidade é a maior característica conseguida na organização do equipamento.

Foi desenvolvido um modelo analítico baseado em teoria de filas para fazer estimativas do retardo médio (Hashioka, no prelo). Para aplicação inicial desejada ao sistema, com velocidade das linhas de comunicação normalmente com o MODEM (4800, 9600 Bps), o modelo analítico mostrou que o MCR irá operar com grande margem de segurança, ou seja, dificilmente ocorrerá congestionamento interno.

REFERÊNCIAS BIBLIOGRÁFICAS

- HASHIOKA. M. H. *Modelo e análise de uma interface de comunicação com processamento distribuído para aplicação em rede de comunicação por comutação de pacotes.* Dissertação de Mestrado. S.J.Campos, INPE no prelo.
- KLEINROCK. L. *Queueing systems.* New York, John Wiley, 1976. V.2.
- TANENBAUM. A. S. *Computer networks.* New Jersey, Prentice-hall, 1981.
- TAROUCO, L. M. R. *Redes de comunicação de dados.* Rio de Janeiro, Livros Técnicos e Científicos Editora, 1979.