

RELIME - Rede local do IME\*

Ivan Melo de Carvalho\*\*

Emmanuel Lopes Passos\*\*\*

## Sumário

Este trabalho mostra a evolução do projeto RELIME\* com relação ao desenvolvimento de seus níveis 1 (físico) e 2 (enlace). Especifica a rede quanto à topologia, ao método de controle de acesso e à estrutura dos nós no tocante ao controle de enlace, ao processamento do método de acesso e à forma de interface com o hospedeiro.

Apresenta também a forma como foi implementada uma versão simplificada da rede, utilizando-se como hospedeiros 2 microcomputadores Schumec da Seção de Informática do IME.

\* Trabalho financiado pela Finep - projeto 83

\*\* Engenheiro Eletrônico, o aluno de mestrado em Informática do IME

\*\*\* Professor Adjunto do Programa de Pós-Graduação em Informática do IME.

## INTRODUÇÃO

Rede local caso particular das grandes Redes, numa área limitada a 2,5 km e com recursos computacionais de uma única empresa.

Este trabalho tem como objetivo especificar uma rede local para microcomputadores e realizar uma implementação simplificada em "hardware" desta rede, desenvolvendo até o nível 2. A topologia é tipo "barra" e o método de acesso CSMA/CD<sup>1</sup>.

A estrutura dos nós é Hospedeiro, Interface, Controlador e Transceptor, com protocolos Hospedeiro-nó e nó-nó. Nó-nó é basicamente uma parte do HDLC em conjunto com o mecanismo de controle de acesso CSMA/CD. Isto quer dizer que durante a transmissão de quadros em que não hajam colisões prevalece essencialmente o estabelecido no protocolo HDLC.

Desenvolvimento baseado nos princípios da rede Ethernet, cujas características conhecidas são: 10 Mbps, cabo coaxial, até nível 2 (tipo HDLC), CSMA/CD, arquitetura em camadas separando níveis 1 e 2, simples, barato, flexibilidade de do endereçamento etc.

### 2 - Desenvolvimento

Com relação a constituição das redes observa-se

que os fabricantes<sup>2,3</sup> desenvolvem controladores (nível 2) do tipo padrão ou universais ("universal") que de um lado são ligados a um transceptor ("transceiver") que por sua vez liga-se ao meio físico da rede e por outro lado são ligados de forma paralela às interfaces com diversos tipos de "buses" existentes (QBUS, UNIBUS, MULTIBUS, S100, etc).

Este tipo de configuração é exemplificada na figura 1 onde se observa a aplicação do conceito de modularidade.

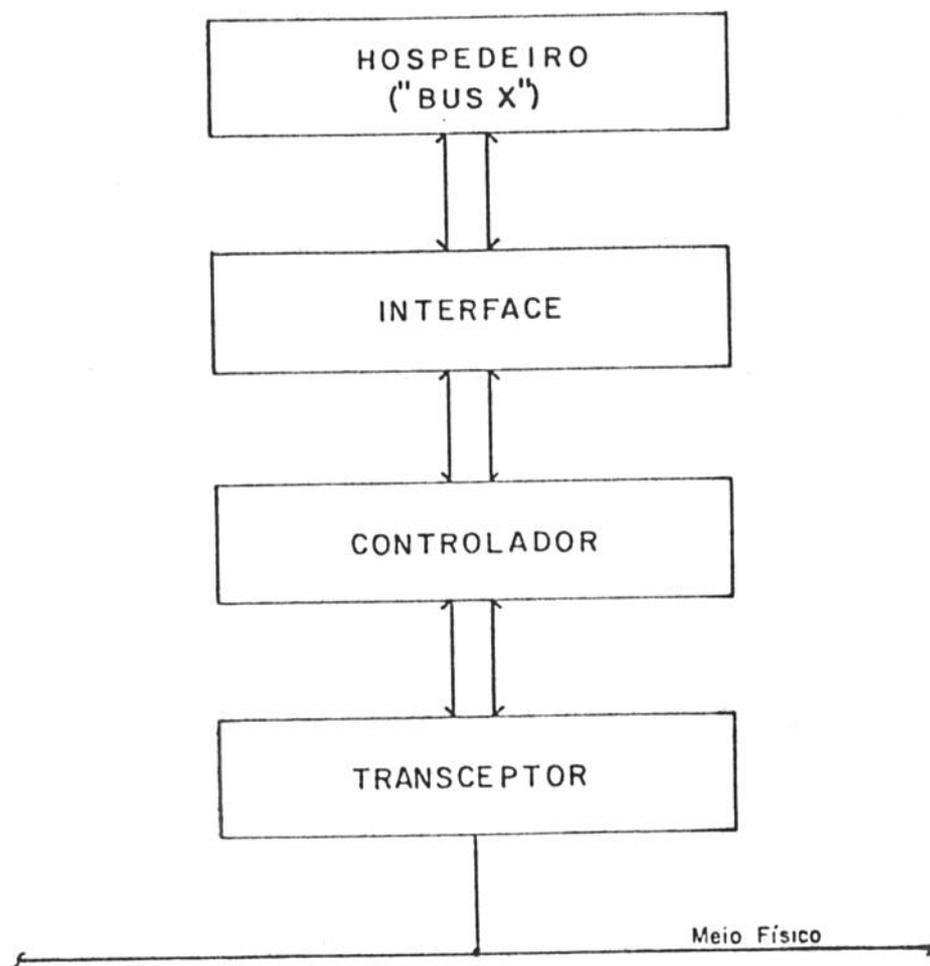


Figura 1 : Constituição de uma Rede Local.

A construção física de interface e do controlador é feita, na maioria dos casos, numa única placa com dimensões compatíveis com as placas do computador hospedeiro em questão, de forma que a mesma possa ser instalada dentro do computador, como mostrado na figura 2. Fica, assim, o nó da rede parcialmente embutido no hospedeiro permanecendo fora apenas o transceptor e os cabos de conexão do mesmo.

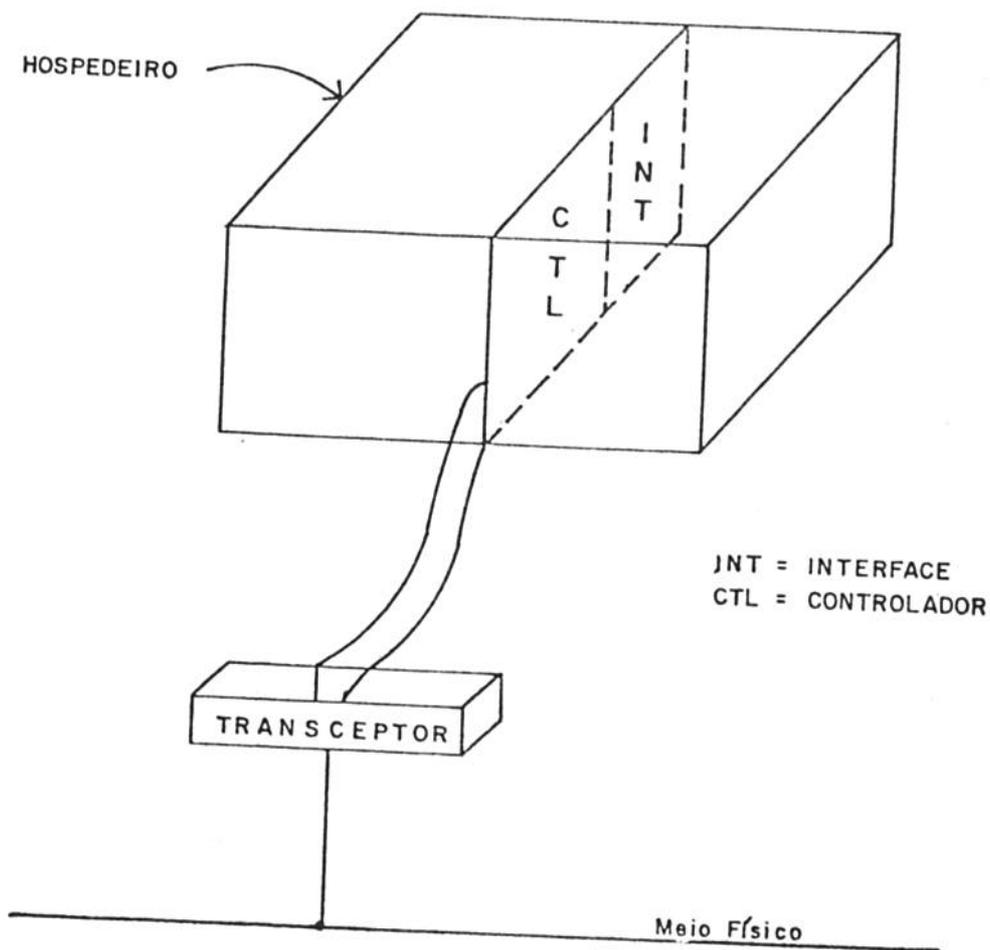


Figura 2 : Construção Física da Interface e do Controlador.

Aos controladores anteriormente citados são atribuídas principalmente funções do tipo.

a) Transmissão e recepção de quadros (nível de enlace), caracterizados por:

i ) Permitirem diversas formas de endereçamento: único, em grupos ("multicast") e a todos (difusão ou "broadcast").

ii ) Terem alguma capacidade de armazenamento temporário ("Buffers") tanto para recepção como para transmissão de dados, de forma a aliviarem o hospedeiro e o próprio meio físico em tráfego normal ou sob tráfego mais intenso.

b) Monitoração de ocorrência de erros.

Não há preocupação com recuperação de erros de transmissão/recepção até o nível 2, somente detecção, o que pode ser atribuído à projeção da rede Ethernet<sup>1</sup>.

Admite-se que protocolos de níveis superiores no hospedeiro, aplicando políticas de reconhecimento ("acknowledgment") e outras técnicas de controle farão a recuperação de erros ocorridos com pedidos de retransmissão, etc.

### 3 - Especificação da Rede

#### Topologia

Nossa proposição é a de uma rede que utilize topologia do tipo de barra ("bus") pois acreditamos que a mesma possui uma confiabilidade intrínseca maior do que as demais aplicáveis a redes locais.

### Método de Acesso

CSMA/CD - acreditamos que este método é o que permite a realização da gerência do contrôle de acesso, ao meio físico da rede, de forma mais distribuída.

### Estrutura dos Nós

A figura 3 apresenta a estrutura do sistema HOSPEDEIRO, INTERFACE, CONTROLADOR e TRANSEPTOR que compõem um dos sistemas que estarão conectados à rede.

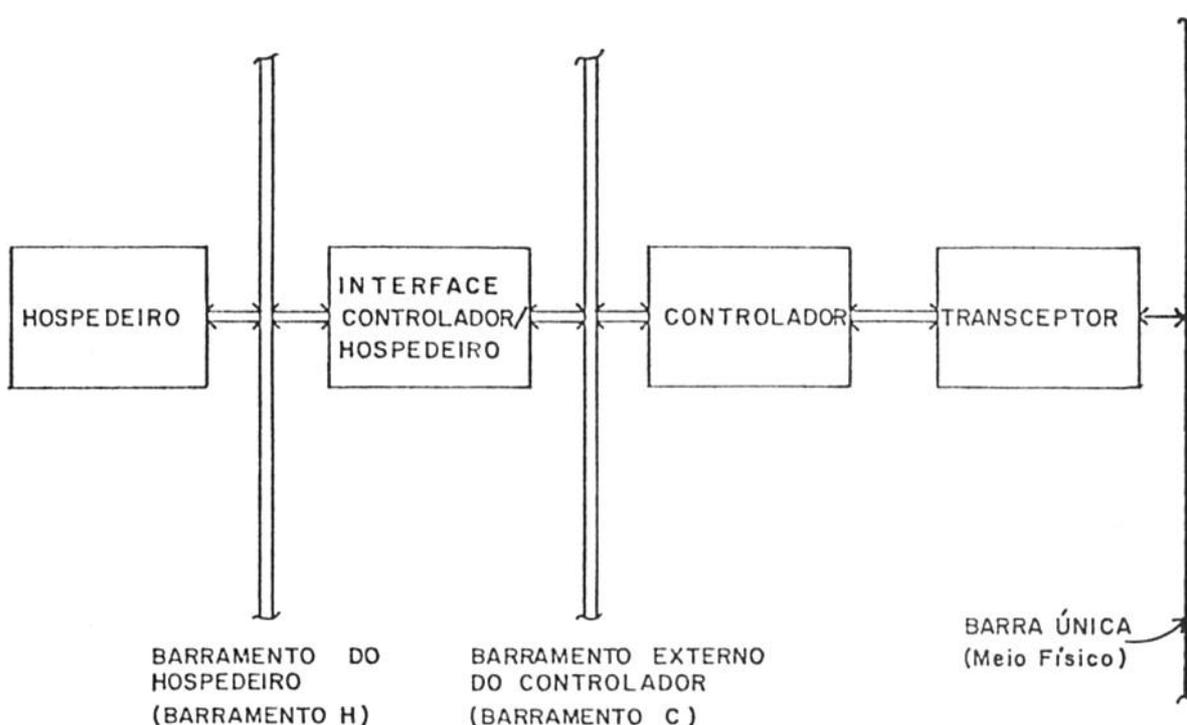


Figura 3 : A Estrutura da Rede Proposta.

Nos ateremos aqui a detalhar a estrutura e a operação do controlador.

a) Funções do Controlador

- i ) Receber quadros colocados na barra única que se destinem a ele de acordo com as modalidades de endereçamento possíveis.
- ii ) Informar ao Hospedeiro a recepção de quadros livres de erros.
- iii ) Ter capacidade de armazenamento para recepção de até 2 quadros de tamanho máximo.
- iv ) Transmitir quadros, a partir de dados enviados pelo Hospedeiro, fazendo a gerência necessária para evitar colisões e retransmiti-los caso estas ocorram.
- v ) Ter capacidade para armazenar dados correspondendo a 2 quadros de tamanho máximo, para transmissão.
- vi ) O CTL (Controlador) deve se apresentar ao hospedeiro como um dispositivo de E/S (I/O) que recebe comandos, fornece status, recebe e transmite dados de ou para o meio externo. A transferência de dados Hospedeiro/CTL poderá ser feita via instruções de E/S ou preferencialmente via DMA.

vii ) O CTL deve se apresentar ao Hospedeiro como um canal de comunicação contendo dois subcanais (transmissão e recepção) independentes sob o ponto de vista lógico (\*) embora interdependentes fisicamente em função da topologia BARRA da Rede.

viii) Gerência de filas na memória interna para grupos de dados correspondentes a quadros recebidos e/ou quadros a serem transmitidos ao meio externo.

(\*) Isto permite que o controlador receba quadros endereçados a ele mesmo.

#### b) Detalhamento do Controlador

##### i ) Diagrama em blocos

A estrutura interna do controlador foi concebida de forma que a mesma possuísse um conjunto de blocos tal, que permitisse, no caso de uma implementação, a utilização de pastilhas poderosas existentes no mercado, normalmente associados a uma determinada família de microprocessadores(\*).

O diagrama em blocos do controlador proposto é apresentado a seguir (figura 4) onde procuramos simplificar as ligações, mantendo apenas os sinais que possuem uma maior importância para o entendimento lógico do mesmo.

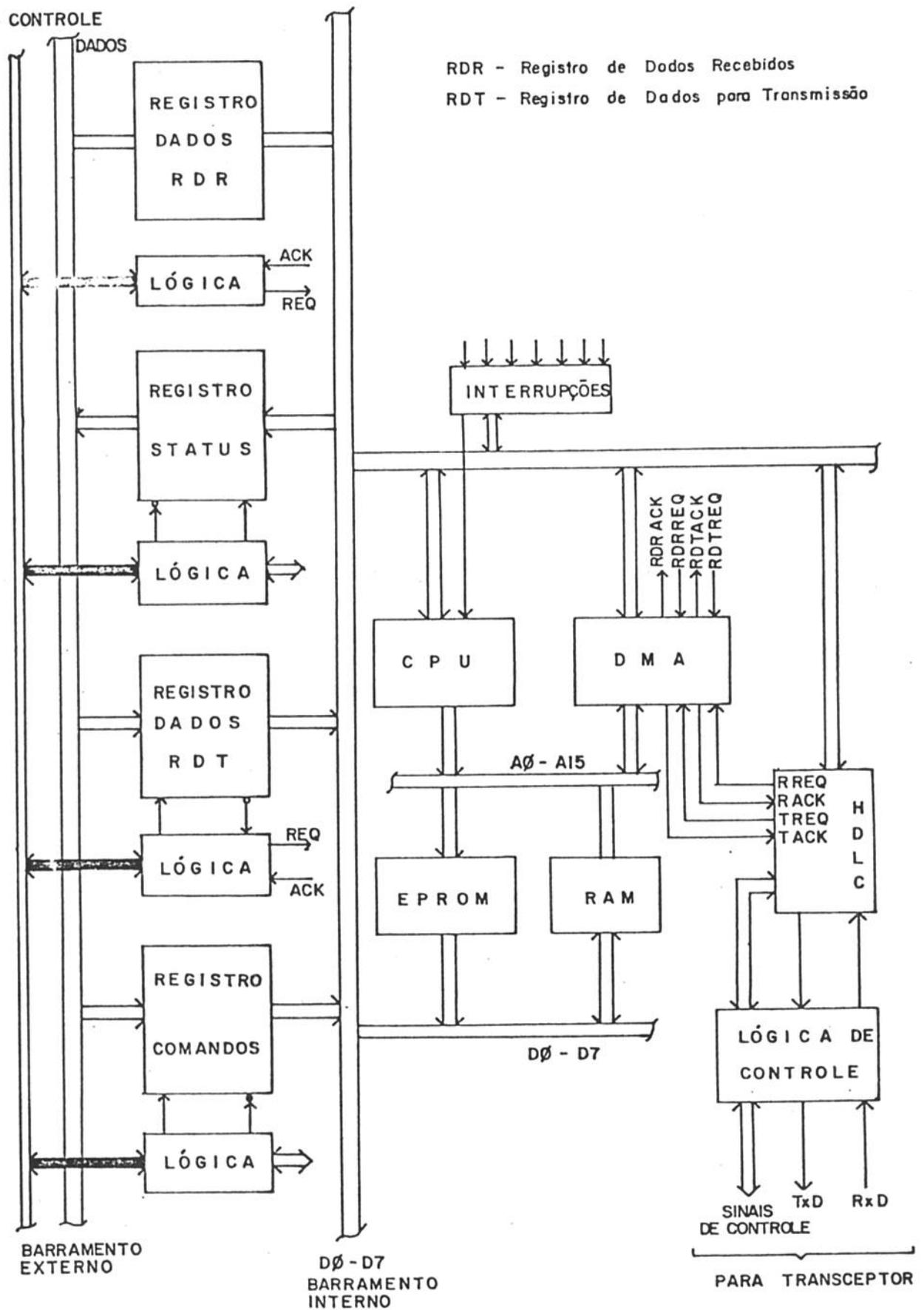


Figura 4 : Diagrama em Blocos do Controlador Proposto.

(\*) Os fabricantes de microprocessadores procuram fornecer aos usuários pastilhas chamadas microperiféricos que suportem uma série de aplicações (comunicação série, comunicação paralela, DMA, etc).

ii ) Descrição dos blocos principais

CPU - Gerencia a operação dos demais blocos do controlador. Executa um programa residente no módulo EPROM que permite:

- Inicializar os módulos DMA e HDLC.
- Interpretar e executar comandos enviados pelo Hospedeiro.
- Fornecer status ao Hospedeiro.
- Cumprir o protocolo de diálogo com o Hospedeiro para recebimento de comandos/dados e envio de dados/status.
- Preparar a operação do módulo DMA fornecendo ao mesmo os parâmetros necessários para que se realizem as transferências de dados via DMA (DADOS HOSPEDEIROS/MEM.CTL e MEM.CTL/HDLC).
- Preparar o módulo HDLC para envio/recepção de quadros, fornecimento de status e geração de sinais de controle.
- Gerencia da área de RAM, constituindo filas para recepção e transmissão de dados.

HDLC - Executa o protocolo HDLC.

- Endereço programável com 8 bits de comprimento.
- Recebe quadros de acordo com as diversas formas de endereçamento (único, "broadcast", etc).
- Transmite quadros no formato HDLC, inserindo seu endereço e todos os demais campos estabelecidos.
- Gera os sinais de controle necessários para o diálogo com a CPU e para a transferência de dados via DMA.
- Possui linhas de controle à disposição da CPU para uso em conjunto com o bloco LÓGICA DE CONTRÔLE.

DMA - Executa operações de DMA em 4 canais independentes.

Canal HDLC/MEM - transfere dados recebidos pelo módulo HDLC, vindos do meio externo, para a memória RAM.

Canal MEM/HDLC - Transfere dados da memória para o módulo HDLC que os transmitirá serialmente ao meio externo.

Canal RDT/MEM - Transfere dados enviados pelo hospedeiro registra

dos no RDT para a memória interna do CTL.

Canal MEM/RDR - Transfere dados da memória interna para o registro RDR que será posteriormente acessado pelo hospedeiro.

Todas as transferências são coordenadas pelo CPU que fornece ao módulo DMA os endereços de memória onde iniciam os "buffers" de dados, o número de bytes a serem transferidos, etc.

LÓGICA DE CONTROLE - Auxilia no controle de acesso à barra única.

- Detecta presença de sinais de transmissores na barra.
- Acusa a ocorrência de colisões.
- Auxilia a CPU no atendimento às interrupções possíveis dentro do controlador.

INTERRUPÇÕES

- x Recebimento de comando do Hospedeiro
- x Leitura de status pelo Hospedeiro
- x Término de transmissão para o meio externo

- x Ocorrência de colisões
- x Término de transferência de dados nos canais de DMA
- x Relógio ("TIME-OUTS").

## PROTOSCOLOS

- a) Nô/Nô - O protocolo de conversação Nô/Nô é basicamente uma parte do HDLC (\*) em conjunto com o mecanismo de controle de acesso CSMA/CD.

Isto significa que durante a transmissão de quadros em que não hajam colisões prevalece essencialmente o estabelecido no protocolo HDLC.

Adicionalmente, o mecanismo CSMA/CD estabelece que o início de uma transmissão seja vinculado a não existência de outro transmissor na barra (CD-"carrier detect").

Em caso de ocorrência de colisão, a mesma deve ser "reforçada" pelos nós que primeiro a perceberam, de forma que esta seja detectada também por todos os outros que porventura tenham participado desta transmissão simultânea.

(\*) Referente ao formato dos quadros, à verificação de endereço por parte do receptor e verificação de erros.

Novas tentativas de transmissão após a ocorrência de colisões devem aguardar um período de tempo gerado de forma pseudo-aleatória por parte de cada nó envolvido na colisão.

ão. Isto diminuirá a probabilidade de ocorrência de novas colisões.

### Protocolo Hospedeiro/Nó

Hospedeiro (Host) deseja transmitir comando para controlador (CTL).

- HOST escreve comando no registro de comandos do controlador, gerando uma interrupção no CTL.
- O recebimento do comando por parte do CTL é percebido (INTERRUPÇÃO) pelo HOST.
- CTL fornece status referente à aceitação do comando, interrompendo o HOST.

Em caso de desejo de transmissão de dados para o meio externo por parte do HOST, este escreve no registro de dados RDT, em dois ciclos consecutivos, o nº de BYTES que vão ser transferidos.

- CTL informa ao HOST a possibilidade ou não de receber estes dados (análise da área de RAM disponível).
- Em caso positivo, HOST transfere dados para memória interna do controlador (via RDT).
- CTL informa ao HOST o sucesso da transferência de dados para o meio externo referente ao primeiro quadro da fila de espera (caso vários quadros estejam no controlador aguardando transmissão para o meio externo).

- O formato dos dados passados do HOST ao CTL é mostrado a seguir (figura 5).

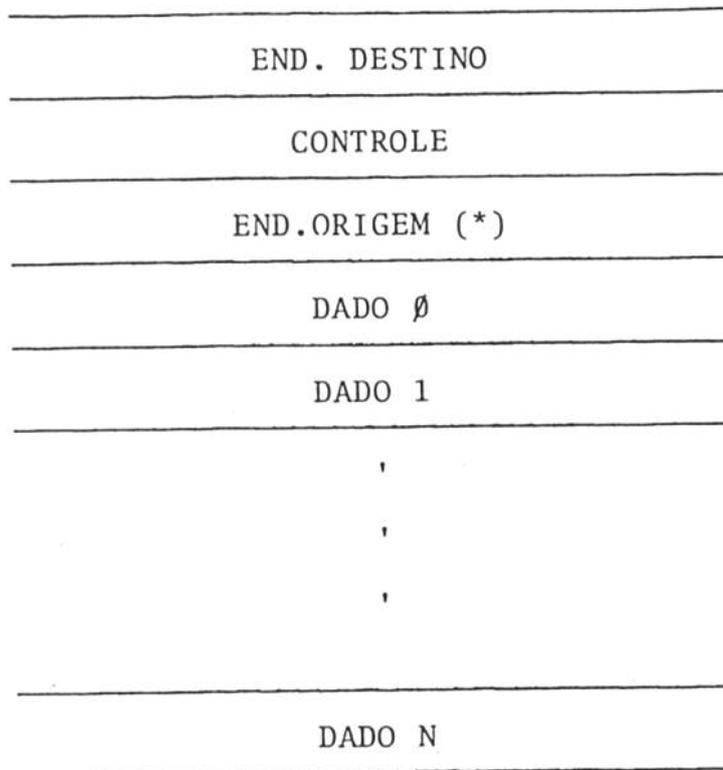


Figura 5

(\*) Inserido pelo CTL que possui um único endereço que o identifica.

#### RECEPÇÃO DE DADOS

- HOST, através de um comando, habilitou o CTL à recepção de dados do meio externo.
- CTL captura quadros a ele endereçado e após verificar a origem e o comprimento, informa ao HOST.
- HOST informa ao CTL o desejo de receber os dados aguardando nos 2 próximos ciclos de escrita no registro de dados (RDR) o nº de BYTES deste quadro.

- HOST informa o desejo ou não de continuar a operação.
- Os dados passados do CTL ao HOST terão o formato apresentado na figura 6.

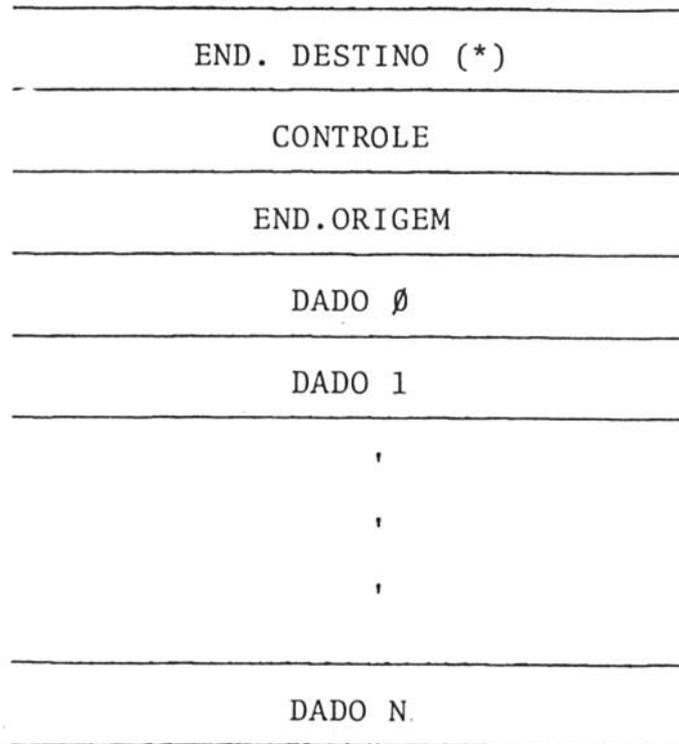


Figura 6

#### 4 - Implementação e Testes Realizados

Simplificando-se a estrutura de "hardware" proposta foram montados os circuitos necessários à realização da interconexão de dois micros shumec da Seção de Informática do IME. Os circuitos foram montados em duas placas s-100 com ligações em "wire-wrap".

As simplificações feitas em relação ao proposto

consistiram principalmente de:

- a) Um único canal de DMA (uma pastilha Z80-DMA) ao invés dos 4 propostos. Este canal é então partilhado entre a recepção e a transmissão serial de dados. A transferência de dados entre os registros de dados a transmitir e recebidos e a memória é feita sob a supervisão da CPU, fazendo uso de interrupções.
- b) Não foi implementado um mecanismo de relógio para geração de "time-outs".

#### Estrutura do controlador

A figura 7 apresenta um diagrama simplificado do controlador (CTL) implementado.

O controlador utiliza um microprocessador Z80 e alguns microperiféricos da família Z80 para suporte das operações de transmissão/recepção síncrona (SIO), DMA e interfaceamento paralelo (PIO).

A descrição de cada bloco assim como a operação do controlador se aproxima do que foi apresentado aqui. Devemos ressaltar o partilhamento de um único canal de DMA entre transmissão e recepção serial de dados e o fato de que o Z80 e seus microperiféricos trabalham com sistema de interrupção vetorizado. As prioridades das interrupções são determinadas intrinsecamente em função de interligação dos microperiféricos ("Daisy-chain"). Isto simplifica o "hardware"

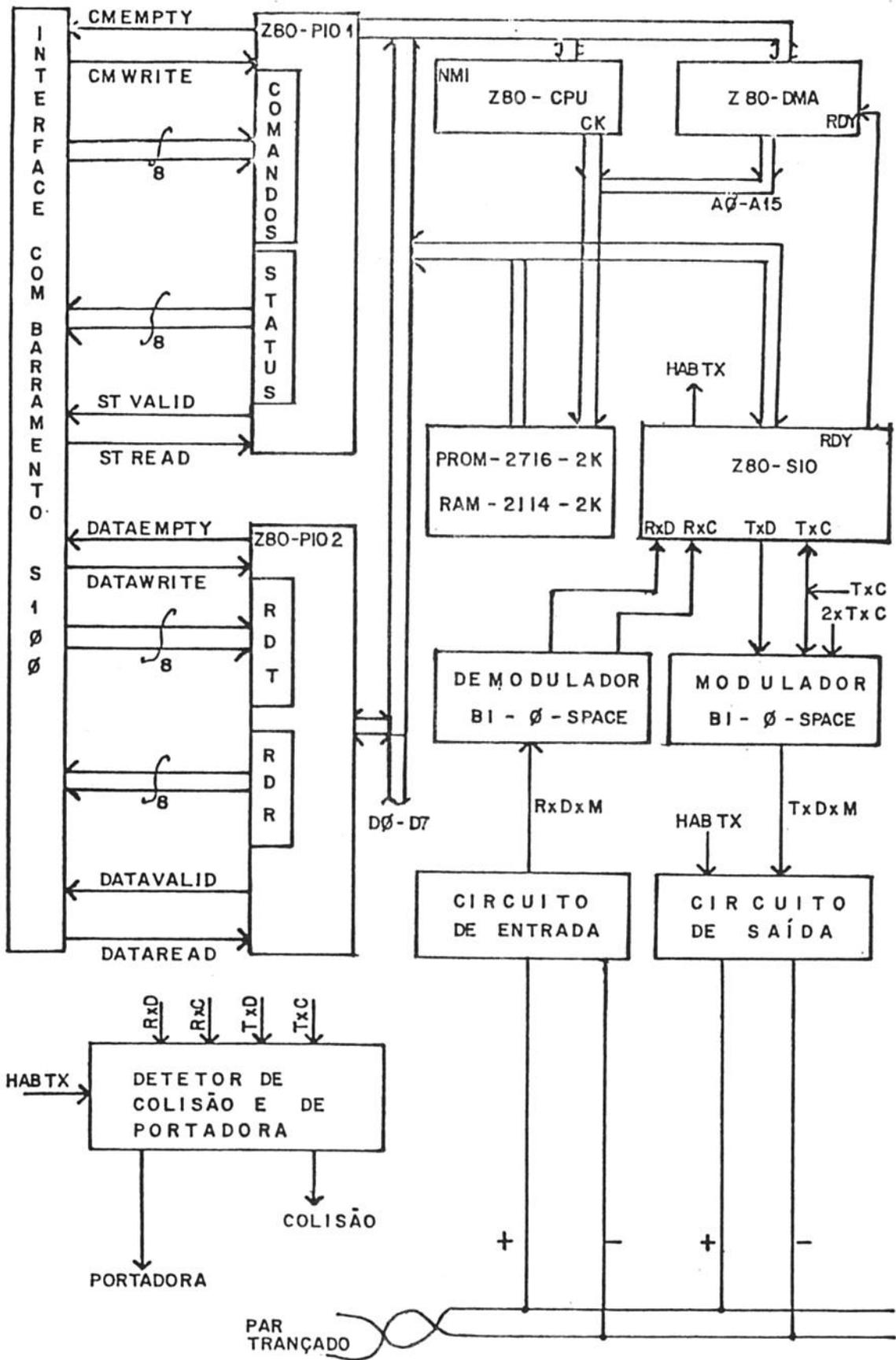


Figura 7 : Diagrama em Blocos do Controlador Implementado.

pois dispensa o uso de um controlador de interrupções.

#### Interface com hospedeiro

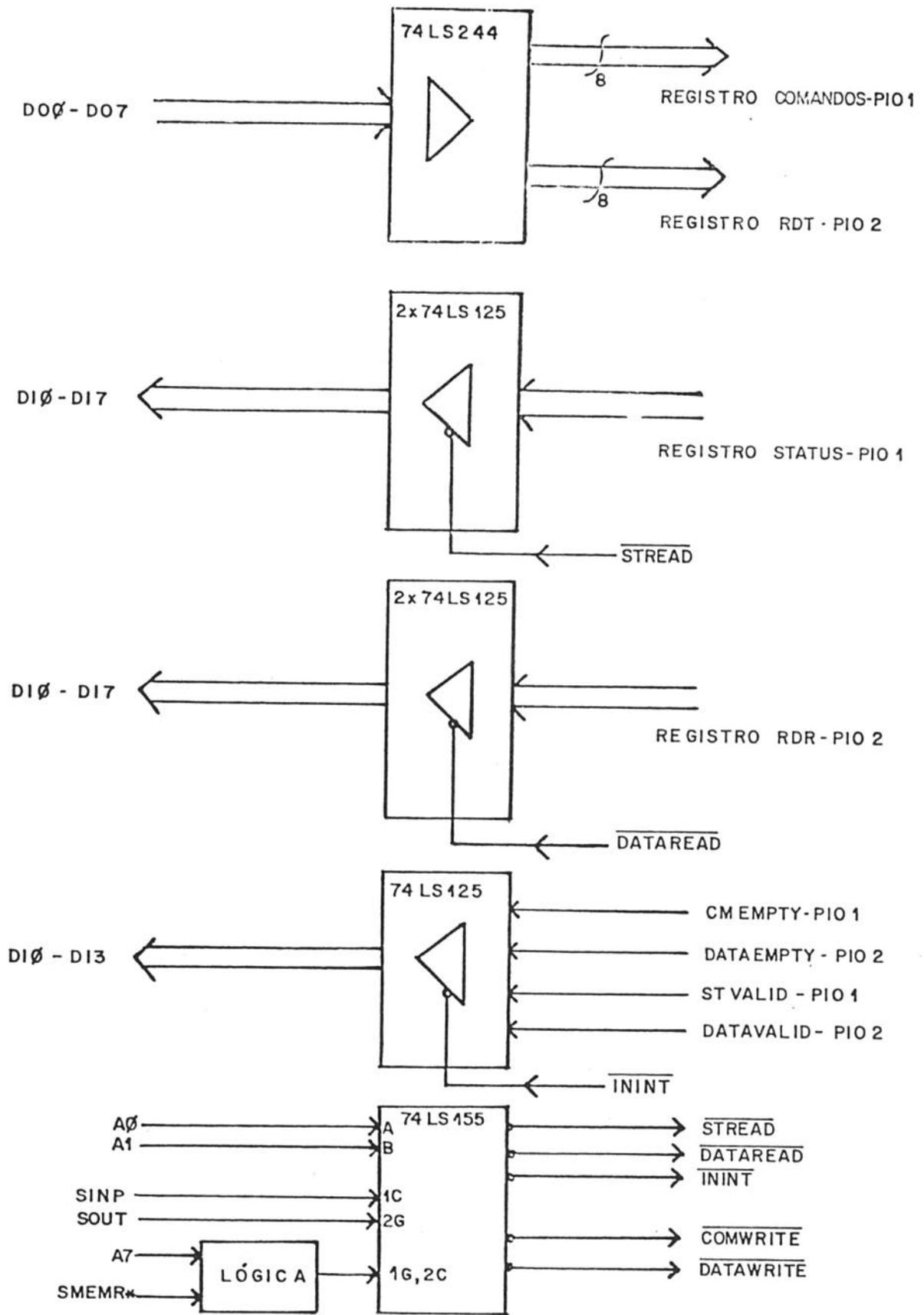
Os micros shumec possuem um barramento tipo S-100 (IEEE 696).

A interface implementada é mostrada na figura 8.

#### a) Descrição e Operação

Em função da interface, o controlador passa a ser visto pelo "host" (shumec) como um dispositivo de E/S contendo 3 canais de entrada e 2 canais de saída como descritos a seguir.

- Canal COMWRITE (8 bits) - destinado a escrita de comandos no controlador, é acionado através de uma instrução do tipo "OUT 00".
- Canal DATAWRITE(8 bits) - destinado a escrita de dados no controlador, é acionado através de uma instrução do tipo "OUT 0 1".
- Canal STREAD (8 bits) - através de uma instrução do tipo "IN 00" o host pode ler o registro do controlador que contém as informações de status do mesmo.
- Canal DATAREAD (8 bits) - através de uma instrução do tipo "IN 0 1" o host acessa o registro de dados do controlador.
- Canal ININT (4 bits) - através deste canal o



BARRAMENTO S I $\emptyset\emptyset$

Figura 8 : Diagrama em Blocos da Interface Implementada.

host toma conhecimento do status dos demais canais no tocante a presença de informações ainda não lidas nos canais de dados e/ou status por parte do host ou de informações referentes à ocorrência de leitura de um comando e/ou de um dado por parte do controlador. O significado dos bits constantes neste canal é mostrado a seguir. Este registrador ININT é lido através da instrução "IN 02".

REGISTRO ININT							
D7	D6	D5	D4	D3	D2	D1	D0
X	X	X	X	DATA VALID	ST. VALID	DATA EMPTY	CM EMPTY

- CM EMPTY - Quando igual a "1", indica que o último comando escrito pelo host, já foi lido pelo controlador.
- DATA EMPTY - Quando igual a "1" indica que o último dado escrito pelo host, já foi lido pelo controlador.
- STVALID - Este bit, quando igual a "1" informa ao host que o último Status fornecido pelo controlador é válido ou seja ainda não foi lido.

DATAVALID - Este bit, quando igual a "1" informa ao host que o último dado fornecido pelo controlador ainda não foi lido.

Conceitualmente o canal ININT deveria, ao nível do host, estar veinculado a algum mecanismo de interrupção, o que possibilitaria que o diálogo host-CTL pudesse ser feito de forma assíncrona. Achamos, entretanto, que este mecanismo poderia ser implementado posteriormente, admitindo que nesta primeira fase este diálogo fosse feito através de um "polling" do registro ININT.

### Transceptor

Não houve, em nossa implementação a preocupação de distinguirmos fisicamente os circuitos que em outras redes compõem os transceptores. O que fizemos foi incorporar estes circuitos à placa destinada ao controlador.

O processo de modulação que utilizamos para embutir o clock no sinal a ser transmitido denomina-se modulação "Bi- $\emptyset$ -space".

Este processo consiste em criar transições de níveis no sinal a ser transmitido numa taxa correspondente à taxa de transmissão. Adicionalmente, criam-se transições num ponto correspondente no tempo à metade de um bit quando o mesmo for igual a " $\emptyset$ ".

Na recepção deveremos distinguir as transições que correspondem a taxa de transmissão das inseridas no meio de cada bit "0". O diagrama de tempo destas duas operações é mostrado na figura 9 .

Detecção de transmissor presente e detecção de colisão.

A parte do protocolo CSMA/CD referente à detecção de portadora corresponde na verdade a detecção de um transmissor presente na barra. A realização deste mecanismo consistiu num circuito multivibrador monoestável com "re-trigger" ajustado para uma frequência tal que sempre que algum transmissor estivesse presente sua saída fosse igual a "1". Este circuito faz uso dos pulsos gerados a cada transição necessários ao circuito de recepção síncrona utilizado.

Sua saída é então conectada à entrada CTSB do circuito SIO de forma que a CPU lendo um dos registros de leitura do SIO (RR0) e testando a condição do bit correspondente à entrada CTSB tem meios de monitorar a situação da barra antes de habilitar uma transmissão.

O circuito utilizado para detecção de colisão tem como princípio de funcionamento, comparar os dados que estão sendo transmitidos com os que estariam sendo recebidos. A ocorrência de uma colisão determinará fatalmente uma degeneração nos dados transmitidos de forma que uma comparação simultânea entre o que está sendo recebido (vindo da barra) com

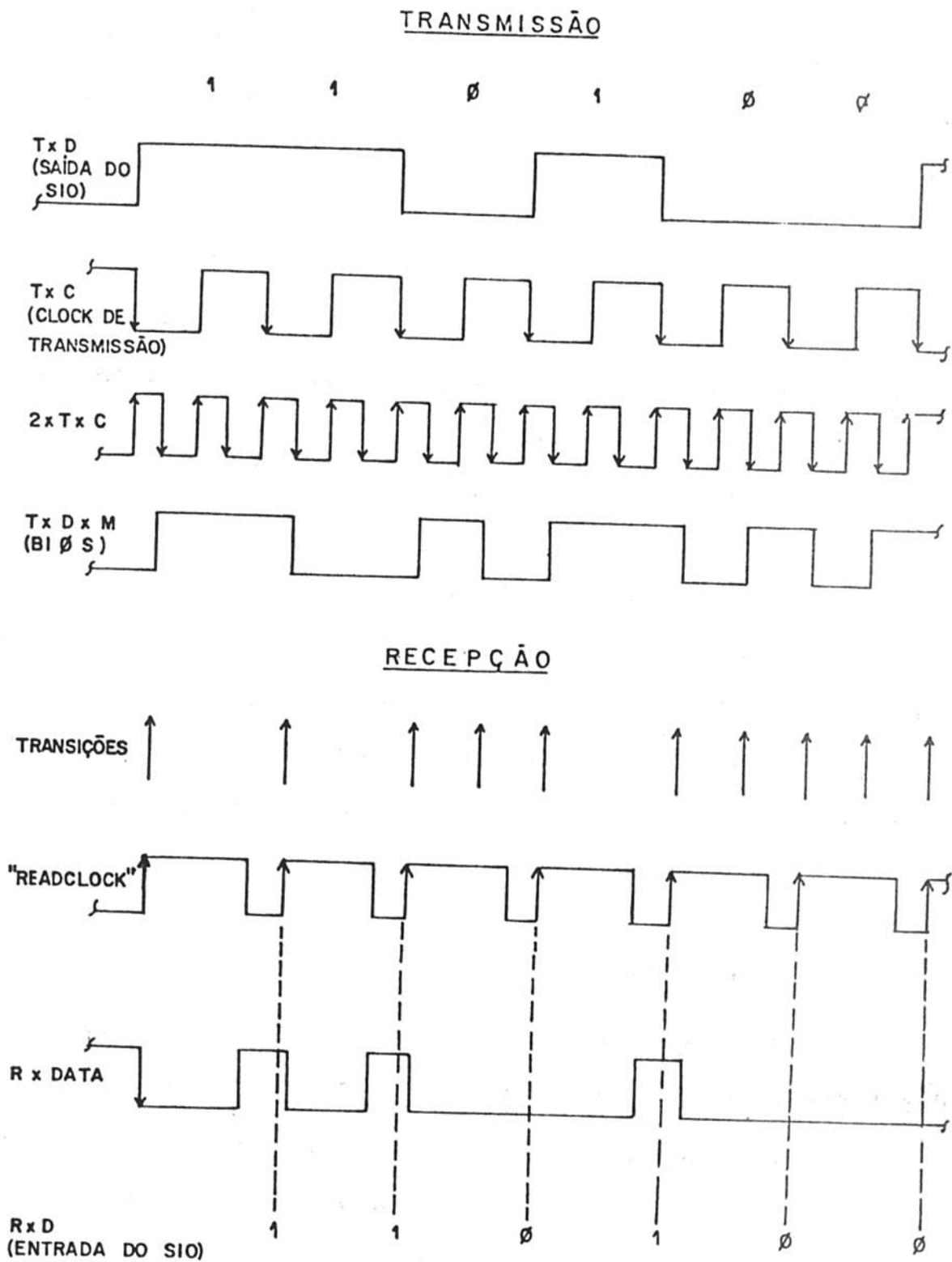


Figura 9 : Diagrama de Tempo do Processo de Modulação e De modulação..

o que está sendo transmitido (antes da barra) resultará numa desigualdade, indicativa de colisão. O resultado desta colisão é passado à CPU através de uma interrupção não mascarável para que devidas providências sejam tomadas.

Circuitos de entrada e saída e meio de transmissão

Decidimos utilizar como meio de transmissão um par trançado, sem blindagem, que atende às necessidades de taxa de transmissão desejada (400 kbps) e às possíveis distâncias que a rede venha a percorrer dentro do IME<sup>8</sup>.

Os circuitos de entrada ("line receivers") e saída ("line drivers") devem então possuir entrada diferencial e saída balanceada respectivamente. Estas características dão ao conjunto "driver/receiver" par trançado uma boa imunidade a ruído. Em nossa implementação optamos pelas pastilhas 75174 ("driver") e 75175 ("receiver").

#### Testes realizados

Os itens apresentados a seguir mostram a evolução dos testes, já realizados, em nossa implementação.

- i ) Cada placa foi testada no sentido de verificar o bom funcionamento dos circuitos de CPU, EPROM, RAM, SIO, PIO e DMA.
- ii ) Individualmente, foi testada cada interface com o hospedeiro. Foram feitas trocas de dados entre hospedeiro e controlador através dos canais COMWRITE, DATAWRITE, STREAD e DATAREAD.

- iii) Cada conjunto hospedeiro/controlador foi testado no tocante à comunicação serial com o SIO programado na modalidade SDLC. Isto é, foi feita uma conexão tipo "LOOP" onde verificou-se se os dados transmitidos, após passarem pelos circuitos de modulação e demodulação, eram recebidos corretamente.
- iv ) Foram realizados testes de comunicação entre os dois sistemas "Hospedeiro/Controlador". Isto é, dados entrados pelo terminal de um hospedeiro eram enviados ao outro "Hospedeiro/Controlador" com posterior apresentação no terminal deste último.

O programa final do controlador deverá ser capaz de aceitar comandos dentro do protocolo estabelecido, enviar status, fazer transmissão e recepção de quadros nas modalidades única e por difusão, etc.

#### Conclusão:

Os objetivos estão sendo alcançados, pois já conseguiu-se transferência de dados de um terminal ao outro (nos dois sentidos).

#### As continuações:

- a) Criação de protocolos de níveis superiores(3a7) até uma aplicação de automação de escritório.

- b) Desenvolvimento de outros controladores para os outros micros Shumeck existente no IME.
- c) Desenvolvimento de controladores específicos para suportarem o partilhamento de discos e impressoras.

BIBLIOGRAFIA (Resumo)

- 1 - Digital, Intel, Xerox, the Ethernet, A Local Area Network, Data link layer and Physical layer Specification, Version 1.0, set. 30, 1980.
- 2 - 3Com, 3C400 multibus Ethernet Controller, 3Com Corporation
- 3 - Interlan, NI2010 bus Ethernet Communications Controller, Interlan.
- 4 - MFE, Recording Techniques, MFE, Application Note NO.64B, out 1975.
- 5 - Kotelly, G., Local Area Networks, EDN, 109-150, fev. 1982.
- 6 - Zilog, Z80CPU, Z80DMA, Z80 SIO, Z80 PIO, Z80 CTC, Zilog, Product Specification Catalog, 1981.
- 7 - Osborne, McGraw-Hill, Interfacing to S.100/IEEE696 Micro computers, Osborne/McGraw-Hill, 1981.