

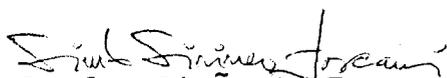
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

UM SISTEMA DE COMUNICAÇÃO PARA  
UMA REDE LOCAL DE MICROCOMPUTADORES

por

Luis Fernando Friedrich

Dissertação submetida como requisito parcial para  
a obtenção do grau de Mestre em  
Ciência da Computação



Prof. Simão S. Toscani e Martin Tornquist  
Orientador



Co-orientador

Porto Alegre, outubro de 1985

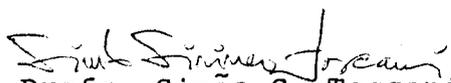
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

UM SISTEMA DE COMUNICAÇÃO PARA  
UMA REDE LOCAL DE MICROCOMPUTADORES

por

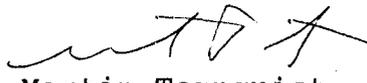
Luis Fernando Friedrich

Dissertação submetida como requisito parcial para  
a obtenção do grau de Mestre em  
Ciência da Computação



Profs. Simão S. Toscani e Martin Tornquist

Orientador



Co-orientador

Porto Alegre, outubro de 1985

Friedrich, Luis Fernando.

Um sistema de comunicação para uma rede local de micro-computadores. Porto Alegre, CPGCC da UFRGS, 1985.  
1 v.

Diss. (mestr. ci. comp.) UFRGS-CPGCC, Porto Alegre, BR-RS, 1985.

Dissertação: Redes Locais: Protocolo de Comunicação:  
Sistemas Operacionais: Mumps.

## AGRADECIMENTOS

Ao Corpo Docente do Curso de Pós-Graduação em Ciência da Computação da UFRGS, em especial ao Prof. Simão Sirineo Toscani pelo incentivo e apoio recebidos durante as etapas do curso.

À CAPES e CNPq pelo auxílio recebido, que possibilitou o desenvolvimento deste trabalho.

Aos meus colegas pelo agradável ambiente proporcionado durante a realização do curso; a Autran, Celso e Miriam pelo apoio e interesse dispensados.

Aos integrantes da linha de pesquisa de Comunicação de Dados pela execução do hardware da rede.

À

Bi e

Fernanda

## SUMÁRIO

LISTA DE FIGURAS .....	08
LISTA DE TABELAS .....	10
RESUMO .....	11
ABSTRACT .....	12
1. INTRODUÇÃO .....	13
2. REDES LOCAIS .....	15
2.1 O Modelo de Referência da ISO .....	15.
2.1.1 Nível de Aplicação .....	17
2.1.2 Nível de Apresentação .....	18
2.1.3 Nível de Sessão .....	18
2.1.4 Nível de Transporte .....	19
2.1.5 Nível de Rede .....	20
2.1.6 Nível de Ligação .....	21
2.1.7 Nível Físico .....	21
2.2 Considerações sobre Redes Locais .....	22
2.3 Padronização em Redes Locais .....	23
2.3.1 Nível Físico .....	26
2.3.2 Nível de Ligação .....	27
3. HARDWARE DE COMUNICAÇÃO .....	30
3.1 Descrição do Adaptador .....	31
4. NÍVEL DE ENLACE .....	33
4.1 Estrutura de dados .....	35
4.2 Método de Acesso .....	36
4.2.1 Tratamento de colisões .....	38
4.3 Primitivas .....	38
4.3.1 Transmissão .....	39
4.3.2 Recepção .....	40
5. NÍVEL DE TRANSPORTE: PRIMITIVAS .....	44

5.1	Considerações sobre o Nível de Transporte proposto .....	44
5.2	Primitivas .....	45
5.2.1	Estabelecimento da conexão .....	45
5.2.1.1	Primitiva CONECTA .....	45
5.2.1.2	Primitiva PERGUNTA .....	46
5.2.2	Transferência de Dados .....	47
5.2.2.1	Primitiva ENVIA .....	47
5.2.2.2	Primitiva RECEBE .....	48
5.2.3	Término da Conexão .....	49
5.2.3.1	Primitiva DESCONECTA .....	49
5.2.4	Difusão de Mensagens .....	50
5.3	Cenário de uma Conexão .....	50
5.4	Diagrama de Transição de Estados .....	50
6.	NÍVEL DE TRANSPORTE: DESCRIÇÃO DO PROTOCOLO .....	53
6.1	Endereçamento .....	54
6.2	Fase de Conexão .....	55
6.2.1	Unidades de Dados .....	55
6.2.2	Procedimentos .....	56
6.2.2.1	Estação com a Iniciativa ....	56
6.2.2.2	Estação Consultada .....	60
6.3	Fase de Transferência de Dados .....	60
6.3.1	Unidades de Dados .....	61
6.3.2	Procedimentos .....	63
6.3.2.1	Estação Transmissora .....	63
6.3.2.2	Estação Receptora .....	65
6.4	Fase de Encerramento de Conexão .....	70
6.4.1	Unidades de Dados .....	70
6.4.2	Procedimentos .....	70
6.4.2.1	Estação Solicitante .....	70

6.4.2.2 Estação Consultada .....	71
6.5 Difusão de Mensagens .....	71
6.6 Controle de Erros .....	74
6.7 Segmentação/Junção .....	74
7. UTILIZAÇÃO DO SOFTWARE DE COMUNICAÇÃO .....	75
8. CONCLUSÕES .....	80
ANEXO 1 Listagem de Software de Comunicação .....	82
APÊNDICE 1 Símbolos da Linguagem de Descrição e Especificação (SDL) .....	117
BIBLIOGRAFIA .....	119

## LISTA DE FIGURAS

Figura 2.1	Arquitetura em níveis para o modelo OSI ..	16
Figura 2.2	Esquema de uma rede local ligando dois prédios .....	24
Figura 2.3	Relação entre os níveis do ISO OSI e o IEEE 802 .....	25
Figura 2.4	O IEEE 802 considera três tipos de abordagem para os níveis abaixo do LCC em normas específicas .....	29
Figura 3.1	Hardware da Rede Local .....	30
Figura 3.2	Adaptador da Rede .....	32
Figura 4.1	Formato da mensagem do Nível de Enlace ...	35
Figura 4.2	Diagrama de funcionamento da rotina TRANSMISSÃO .....	41
Figura 4.3	Diagrama de funcionamento da primitiva RECEPÇÃO .....	43
Figura 5.1	Relacionamento do nível de transporte com os níveis adjacentes .....	44
Figura 5.2	Cenário de uma conexão de transporte ....	51
Figura 5.3	Diagrama de transição de estados .....	52
Figura 6.1	Diagrama de interação .....	53
Figura 6.2	Formato das unidades de dados .....	55
Figura 6.3	Diagrama de execução da primitiva CONECTA.	59
Figura 6.4	Diagrama de execução da primitiva PERGUNTA.	62
Figura 6.5	Formato da unidade de dados DADOS .....	63
Figura 6.6	Diagrama de execução da primitiva ENVIA ..	66
Figura 6.7	Diagrama de execução da primitiva RECEBE .	69
Figura 6.8	Diagrama da execução da primitiva DESCONECTA .....	72

Figura 6.9	Diagrama de execução da primitiva DIFUNDE .....	73
Figura 7.1	Exemplo de utilização da rede local .....	76
Figura 7.2	Situação do sistema de comunicação .....	79

## LISTA DE TABELAS

Tabela 4.1	Campos de Cabeçalho do NE .....	35
Tabela 6.1	Identificação das Unidades de Dados do Protocolo de Transporte .....	55

## RESUMO

Este trabalho descreve a implementação de um software de comunicação para uma rede local de microcomputadores, com base no modelo de referência da ISO.

São descritas as primitivas da interface do nível de transporte com o nível superior, as primitivas da interface do nível de enlace com o nível de transporte e as funções do protocolo de transporte.

## ABSTRACT

This work presents the implementation of a micro-computers communication software applied to local area networks (LAN), consistent with the Reference Model for Open System Interconnection proposed by International Standards Organization (ISO).

The primitives on the interface of the transport level with its upper level and the primitives on the interface of the link level with the transport level are described. The transport protocol functions are also described.

## 1 INTRODUÇÃO

Com o surgimento dos microcomputadores e sua grande proliferação, cada vez mais torna-se necessário compartilhar todos os recursos computacionais, sejam periféricos ou processadores centrais /FER 84/.

Esta necessidade de compartilhamento de recursos computacionais e de distribuição do processamento, com a interconexão de computadores e terminais, faz com que as Redes Locais de computadores apareçam como a grande solução. Estas tem como grande característica a rápida comunicação das informações, envolvendo equipamentos próximos uns dos outros /RAD 84/.

Baseado na arquitetura proposta pela ISO /ISO 80/, o presente trabalho objetiva apresentar a implementação de um software de comunicação para Redes Locais, composto por três níveis de protocolo equivalentes aos níveis 1, 2 e 4 da ISO, que fornece os serviços necessários para a interconexão de sistemas.

O capítulo 2 apresenta a conceituação básica de Redes Locais, mostrando inicialmente o modelo de referência para interconexão de sistemas abertos (OSI) apresentada pela ISO. É mostrado o que existe em termos de padronização em Redes Locais.

O capítulo 3 descreve o hardware da Rede Local, implementado pelo Grupo de Comunicação de Dados do CPGCC da UFRGS.

No capítulo 4 está a descrição do nível 2 do sistema, Nível de Enlace, suas primitivas e o protocolo de acesso à rede usado na implementação.

O capítulo 5 apresenta as primitivas do nível 3 do sistema, o Nível de Transporte, mostrando o cenário de uma conexão de transporte e o diagrama de estados do Nível de Transporte.

No capítulo 6 está descrito o protocolo de Transporte com os procedimentos e unidades de dados usados no Nível de Transporte.

O capítulo 7 apresenta uma proposta de utilização do software de comunicação, mostrando as facilidades que poderiam ser implementadas.

O capítulo 8 faz uma avaliação do sistema implementado, sugerindo também algumas possíveis extensões para o mesmo.

## 2 REDES LOCAIS

### 2.1 O Modelo de Referência da ISO

O modelo de referência para interconexão de sistemas abertos OSI/RM ("Open Systems Interconnection Reference Model"), fornece a base para interconexão de sistemas em aplicações com processamento distribuído.

O princípio escolhido pela ISO, como método de definição e modelo de referência, foi o de estruturação em níveis, com agrupamento lógico das funções necessárias. A arquitetura de níveis foi estruturada com base em algumas premissas básicas /ISO 80/.

- a. Número de níveis suficiente para dividir o trabalho em partes pequenas e de fácil compreensão;
- b. Criação de níveis separados para manusear funções, de forma independente da tecnologia envolvida;
- c. Reunir funções semelhantes no mesmo nível;
- d. Selecionar limites no ponto em que experiências anteriores demonstraram sucesso;
- e. Criar um limite onde possa ser útil, no futuro, ter a correspondente interface padronizada.

Com base nas premissas, um conjunto de sete níveis foi escolhido para constituir a arquitetura OSI/RM (figura 2.1).

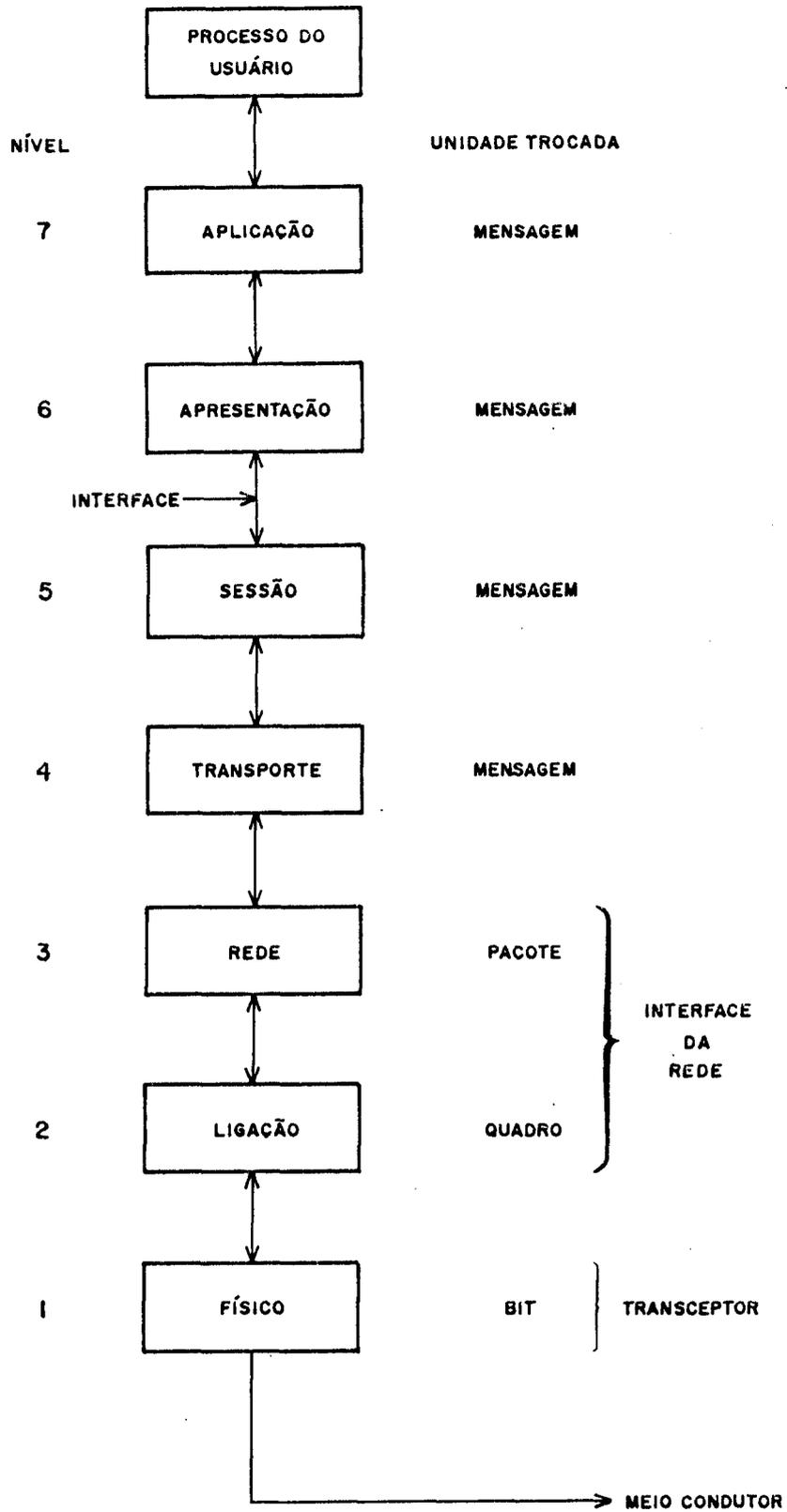


Figura 2.1 - Arquitetura em níveis para o modelo OSI.

### 2.1.1 Nível de Aplicação

O nível de aplicação fornece os serviços para os usuários do ambiente OSI. É através deste nível que os processos do usuário interagem com o ambiente OSI, trocando informações.

Os serviços oferecidos por este nível são os seguintes:

- a. Identificação dos processos que desejam se comunicar (por nome, por endereço, por descrição);
- b. Determinação da disponibilidade de comunicação entre os processos envolvidos;
- c. Estabelecimento de níveis de autorização para comunicação;
- d. Estabelecimento de acordo nos procedimentos para validação dos dados;
- e. Estabelecimento de acordos sobre mecanismos privados necessários;
- f. Estabelecimento de acordo na responsabilidade pela recuperação de erros;
- g. Autenticação dos processos comunicantes;
- h. Seleção da disciplina de diálogo, incluindo procedimentos de inicialização e finalização;
- i. Determinação da metodologia de alocação de custos;
- j. Sincronização das aplicações cooperantes;
- k. Determinação dos recursos necessários;
- l. Identificação de aberrações na sintaxe dos dados;
- m. Determinação da qualidade de serviço aceitável;
- n. Transferência de informações.

O nível de aplicação contém todas as funções necessárias para a comunicação entre sistemas abertos e que são executados pelos níveis inferiores; incluem-se as funções executadas pelos programas como também as executadas por pessoas. As funções, para obtenção dos serviços acima, são classificadas em administração do sistema, administração das aplicações e administração do nível de aplicação.

### 2.1.2 Nível de Apresentação

O objetivo do nível de apresentação é o de oferecer à camada de aplicação os serviços necessários para representar a informação, de maneira que preserve o seu significado e resolva diferenças sintáticas entre sistemas.

São oferecidos pelo nível de apresentação os seguintes serviços:

- a. Gerenciamento de formatação de dados;
- b. Transformação de dados;
- c. Seleção de sintaxe.

Existem três versões sintáticas de dados sendo transferidas: a sintaxe usada pela entidade - aplicação emissora dos dados, a sintaxe usada entre as entidades-apresentação para transferir os dados e a sintaxe usada pela entidade-aplicação receptora dos dados.

### 2.1.3 Nível de Sessão

O objetivo do nível de sessão é fornecer os meios necessários para que entidades-apresentação possam comunicar-se de forma organizada e sincronizada. O nível de sessão gerencia a troca de dados e, fornece serviços para estabelecer uma conexão de sessão entre duas entidades-apresen

tação. Os serviços oferecidos pelo nível de sessão são os seguintes:

- a. Estabelecimento de conexão de sessão entre duas entidades-apresentação;
- b. Liberação da conexão de sessão;
- c. Transferência de dados entre as entidades-apresentação;
- d. Serviço de quarentena, onde é permitida que a entidade-apresentação origem controle explicitamente as unidades de dados a serem entregues à entidade-apresentação destino;
- e. Transferência de dados urgentes que possuem prioridade sobre os outros transmitidos entre entidades-apresentação que possuam conexão de sessão;
- f. Gerenciamento da interação entre as entidades-apresentação, a qual pode ser simultânea ou alternada nos dois sentidos ou somente uma interação do tipo monólogo, em um sentido;
- g. Sincronização da conexão.

#### 2.1.4 Nível de Transporte

O nível de transporte existe para oferecer um serviço de transporte em associação com serviços oferecidos pelos níveis inferiores. O serviço de transporte tem como finalidade providenciar a transferência de dados entre entidades-sessão, de forma transparente.

O nível de transporte possibilita a otimização do uso dos recursos de comunicação disponíveis, para oferecer o desempenho necessário a cada usuário de transporte, mini-

mizando o custo. Os serviços do nível de transporte disponíveis para o nível de sessão, são os seguintes:

- a. Estabelecimento de uma conexão de transporte entre entidades-sessão;
- b. Seleção da classe de serviços de um conjunto de classes disponíveis;
- c. Transferência de informações de acordo com a qualidade do serviço solicitada;
- d. Encerramento da conexão de transporte por qualquer uma das entidades-sessão envolvidas na conexão.

Ao contrário dos níveis anteriores, o nível de transporte é um nível fim a fim, ou seja, preocupa-se com a chegada de suas informações no processo destino.

#### 2.1.5 Nível de Rede

O nível de rede oferece serviços para estabelecer, manter e terminar conexões de rede entre sistemas contendo entidades comunicantes e serviços para a troca de unidades de dados entre entidades-transporte conectadas.

O nível de rede tem como objetivo providenciar para as entidades-transporte, um caminho para que as mensagens sejam transferidas. Este caminho, ou rota, é transparente para a entidade-transporte.

Os serviços oferecidos pelo nível de rede ao nível de transporte são:

- a. Estabelecimento de conexão de rede entre entidades-transporte;

- b. Transferência transparente dos dados entre entidades-transporte;
- c. Estabelecimento e manutenção da qualidade de serviço durante a conexão;
- d. Notificação de erros às entidades-transporte;
- e. Sequenciamento das unidades de informação;
- f. Controle de fluxo.

#### 2.1.6 Nível de Ligação

O nível de ligação (ou enlace) tem como objetivo transformar a facilidade apresentada pelo nível físico em uma linha que pareça isenta de erros para o nível seguinte. Provê os meios funcionais e os procedimentos para estabelecer, manter e encerrar conexões de ligação de dados entre entidades-rede. Os serviços oferecidos ao nível de rede são:

- a. Estabelecimento de conexões entre entidades-rede;
- b. Transferência das unidades de informação na conexão de ligação;
- c. Sequenciamento das unidades de informação, se necessário;
- d. Notificação de erro de comunicação, quando estes forem irrecuperáveis;
- e. Controle de fluxo.

#### 2.1.7 Nível Físico

O nível físico estabelece as características mecânicas, elétricas, funcionais e os procedimentos para ativar, manter e encerrar conexões físicas para transmissão de bits

entre entidades-ligação. Os serviços oferecidos ao nível imediatamente superior são:

- a. Estabelecimento de uma conexão física para entidades-ligação;
- b. Transmissão de um bit em transmissão serial ou "n" bits em transmissão paralela;
- c. Notificação de condições de erro às entidades-ligação;
- d. Identificação dos circuitos de dados;
- e. Sequenciamento na transmissão, entregando os bits na mesma ordem recebida.

## 2.2 Considerações sobre Redes Locais

Uma crescente demanda de serviços e a necessidade de redução de custos nas aplicações de sistemas de computadores tem feito com que Redes Locais sejam consideradas como o grande avanço tecnológico dos anos 80. Pesquisas de mercado indicam que de 130.000 estações em 1983 existirão perto de 1,2 milhões de estações de rede local em 1990.

Uma Rede Local é um sistema de comunicação de dados, direcionada a organização simples, que permite a troca de informação entre dispositivos digitais, similares ou não, através de um meio de transmissão comum.

As comunicações se processam entre diversos tipos de equipamentos tais como:

- computadores de grande porte;
- minicomputadores;
- microcomputadores;
- computadores pessoais;

terminais inteligentes;  
impressoras e discos.

O termo Local determina que as comunicações tenham seu alcance limitado geograficamente. As distâncias envolvidas nas comunicações em redes locais podem variar entre alguns metros até cerca de 10 quilômetros.

O limite imposto permite aplicações como a mostrada na figura 2.2. As ligações entre as redes locais são feitas através de "gateways".

Se comparados às linhas telefônicas, as redes locais tem uma taxa de transmissão mais alta (de 0,25 Kbps a 50 M bps), no entanto ainda são mais baixas que os barramentos de dados dos computadores.

Algumas razões e benefícios para o uso de redes locais são apontadas em /EDN 82/ tais como:

- a. Taxa de transmissão alta;
- b. Baixa taxa de erros de transmissão;
- c. Baixo custo de conexão e instalação;
- d. Grande número de usuários;
- e. Segurança, através da restrição de acesso;
- f. Topologia flexível;
- g. Transmissão de dados, voz e imagem; no mesmo meio de comunicação;
- h. Compatibilidade entre fabricantes.

### 2.3 Padronização em Redes Locais

O desenvolvimento de padronização para redes locais tem sido buscada ultimamente e o seu estabelecimento será certamente uma grande conquista. Para gerenciar este tra

balho de desenvolvimento de uma padronização para redes locais existe o comite IEEE-802. Como base para o desenvolvimento desta padronização está sendo utilizado o trabalho desenvolvido pela ISO, descrito na seção 2.2 /ISO 80/.

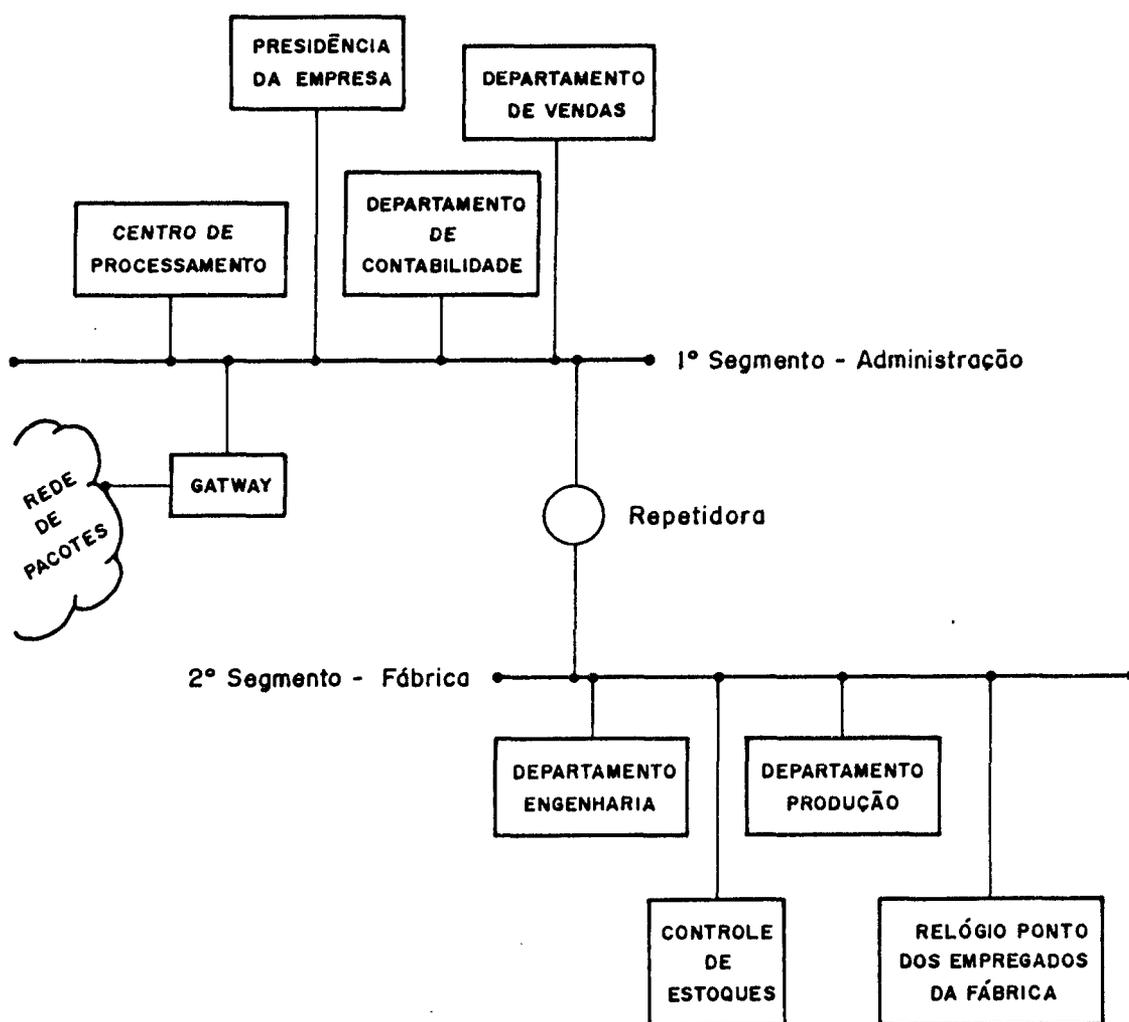


FIGURA 2.2 Esquema de uma rede local ligando dois prédios

O comite IEEE-802 planejou estabelecer padrões para os sete níveis da OSI/ISO, sendo que já existem padronização nos dois primeiros níveis, como mostra a figura 2.3 /SCO 85/.

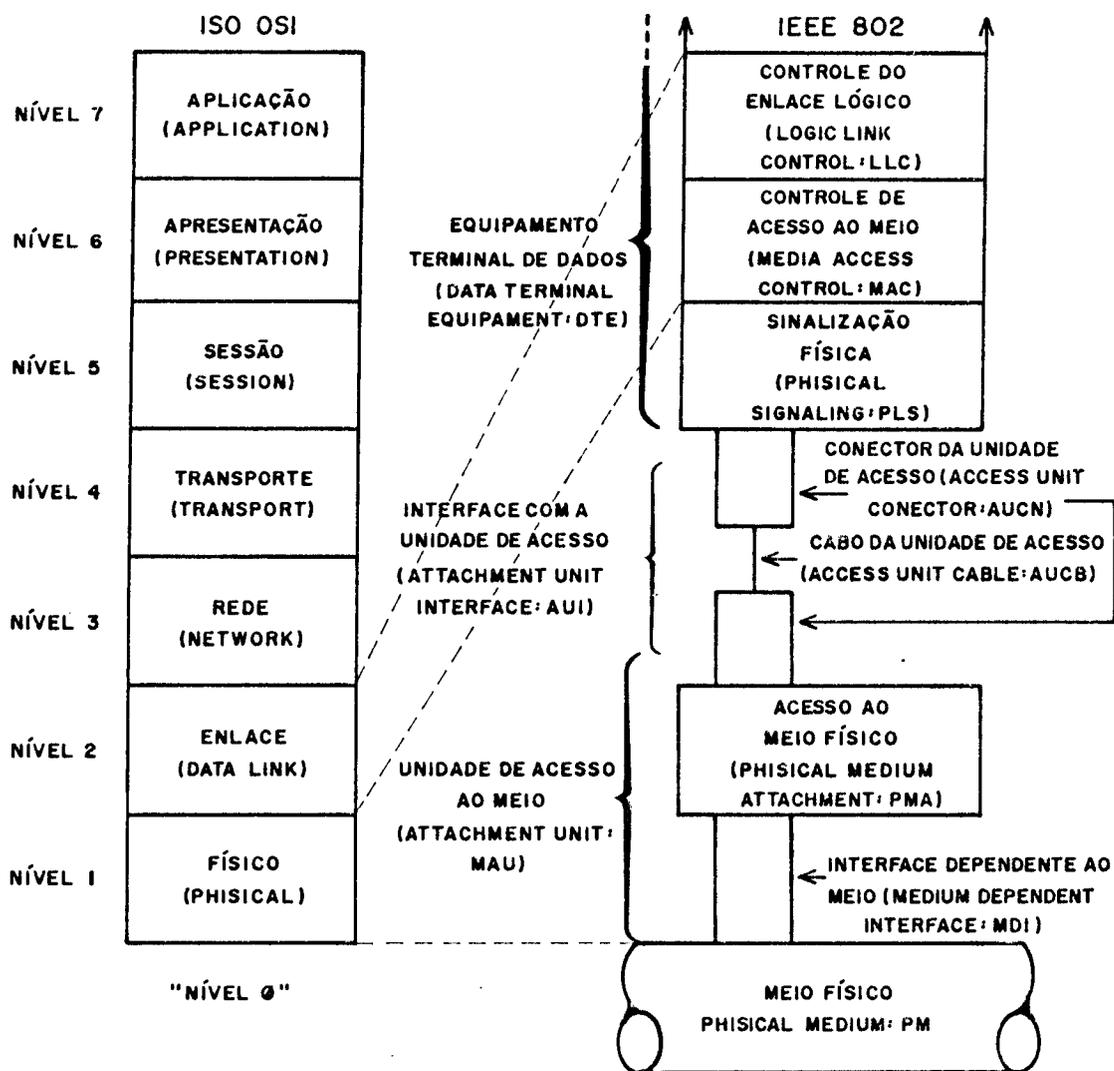


Figura 2.3 - Relação entre os níveis do ISO OSI e o IEEE 802.

### 2.3.1 Nível Físico

Para estabelecer padrões para a tecnologia de transmissão de sinais existente hoje em dia e ainda manter em aberto a possibilidade do surgimento de novas tecnologias, o padrão IEEE 802 define uma unidade chamada Unidade de Acesso ao Meio (UAM) que permite tanto o uso de cabos convencionais quanto de cabos fibra ótica.

Um dos lados da UAM tem um interface padrão para se conectar ao equipamento desejado tal como um computador ou um terminal. O outro lado providencia um interface para o tipo de linha de transmissão em particular. Três tipos estão padronizados, são eles:

- a. Cabo coaxial com sinalização em banda base
- b. Cabo coaxial com sinalização em banda larga
- c. Fibra ótica

Com qualquer um destes tipos de linha um dado pode ser transmitido e recebido serialmente. A UAM contém somente os dispositivos necessários para a particular linha de transmissão, sendo que no aparecimento de novas técnicas de transmissão somente a UAM torna-se obsoleta. Os padrões para taxas de transmissão estão entre 1M e 20 M bps.

O lado da UAM que se conecta com o equipamento realiza a maioria das funções do nível físico tais como: codificação e decodificação dos dados, sincronização e reconhecimento do início e fim de uma unidade de dados.

### 2.3.2 Nível de ligação

O nível de ligação é responsável pela transmissão e recepção de uma unidade de dados. O endereço destino de uma unidade de dados tipo "pacote" (frame) determina qual o dispositivo da rede deve receber os dados. O endereço fonte diz quem está mandando este pacote. Um outro campo "frame-check-sequence" (FCS) é um código gerado pelo transmissor e verificado pelo receptor, que serve para indicar a existência de erros durante a transmissão.

Para que o nível de ligação inicie a transmissão de unidades de dados é necessário que o mesmo saiba se o meio de transmissão está disponível. Isto porque o meio de transmissão é compartilhado por todos os dispositivos na rede. Existem alguns métodos para determinar os direitos de acesso ao meio de transmissão.

O comite IEEE 802 padronizou dois destes métodos de acesso que serão descritos a seguir. A figura 2.4 apresenta os três tipos de acesso considerados pelo IEEE 802.

1. Contenda (carrier-sense multiple access with collision detect-CSMA/CD). Neste método, o nível de ligação que deve transmitir, primeiro escuta a portadora para sentir se outro dispositivo está transmitindo. Se a linha de transmissão está livre, o dispositivo transmite. Devido ao atraso de propagação, na linha de transmissão, dois ou mais dispositivos podem iniciar a transmissão no mesmo instante. Neste caso haverá colisão e o protocolo dirá para dispositivo que detectou colisão, parar de transmitir por um período de tempo randômico. Após esta espera randômica o dispositivo realiza nova tentativa. Este método de acesso é especificado pela norma IEEE 802.3 /IEE 83/.

2 Passagem de permissão (Token Passing). Uma rede local pode usar um protocolo de acesso a linha que permi

te apenas um dispositivo por vez, ter permissão de acesso ao meio de transmissão. Quando o dispositivo está usando a linha de transmissão ele passa a permissão para outro dispositivo através de uma unidade de dados especial. Desta maneira a permissão passa pelos dispositivos da rede, garantindo o direito de acesso a linha de transmissão para cada um destes dispositivos.

Este método de acesso se divide em duas possibilidades:

Passagem de permissão em barramento, especificado pela norma IEEE 802.4 /IEE 83a/, e Passagem de permissão em anel, norma IEEE 802.5 /IEE 84/. Na primeira, o meio consta de um barramento (bus) que interconecta todas as estações, de forma que a escalação da sequência de acesso é puramente lógica, nada tem a ver com a posição física da estação em relação ao meio. Na segunda, as estações estão intercaladas no meio físico, de modo que uma estação genérica recebe mensagens de outra "fisicamente anterior" e as envia para outra "fisicamente posterior" por meio de conexões físicas exclusivas. Os sinais circulam por um Anel onde cada estação se comporta como um interceptador que recebe, consulta, transmite ou repete mensagens.

Cada um dos métodos descritos tem suas vantagens e desvantagens.

CSMA/CD é simples de implementar entretanto, o acesso a linha de transmissão é não determinístico, isto implica em um dispositivo poder ficar repetidamente colidindo com outros na tentativa de acesso ao meio de transmissão

O método de passagem de permissão, por outro lado, é determinístico. Entretanto é mais complexo para implementação. Por exemplo o protocolo deverá estabelecer como

um novo dispositivo que se conecta à rede irá receber a permissão, o que acontecerá quando o dispositivo que tem a permissão quebrar ou quando dois dispositivos estiverem com a permissão.

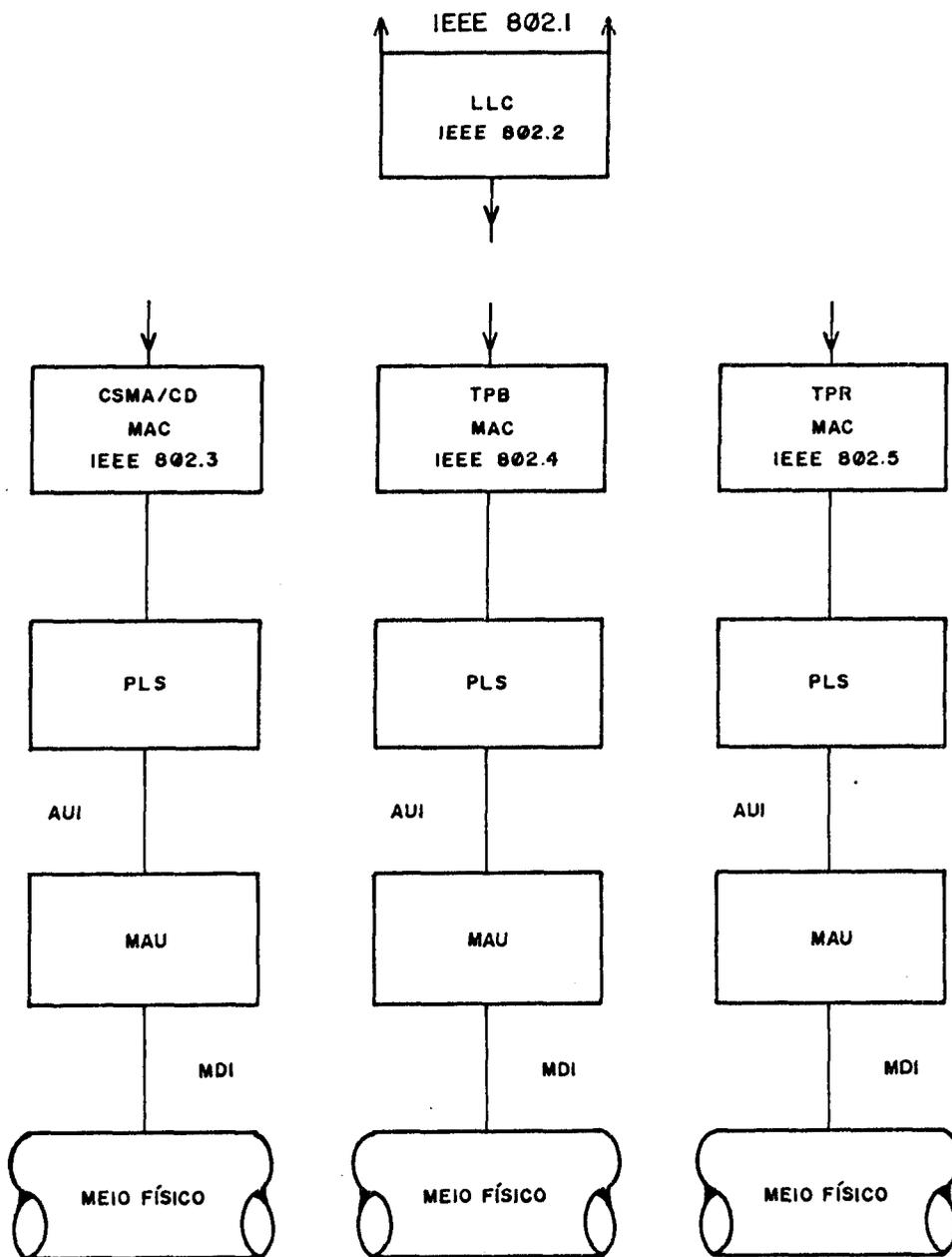


Figura 2.4 - O IEEE 802 considera tres tipos de abordagem para os níveis abaixo do LLC em normas específicas.

### 3 HARDWARE DE COMUNICAÇÃO

A necessidade do sistema no que diz respeito ao hardware, se resume em uma interface de comunicação serial, padrão RS232-C, e um adaptador para rede local em barra, como mostra a figura 3.1.

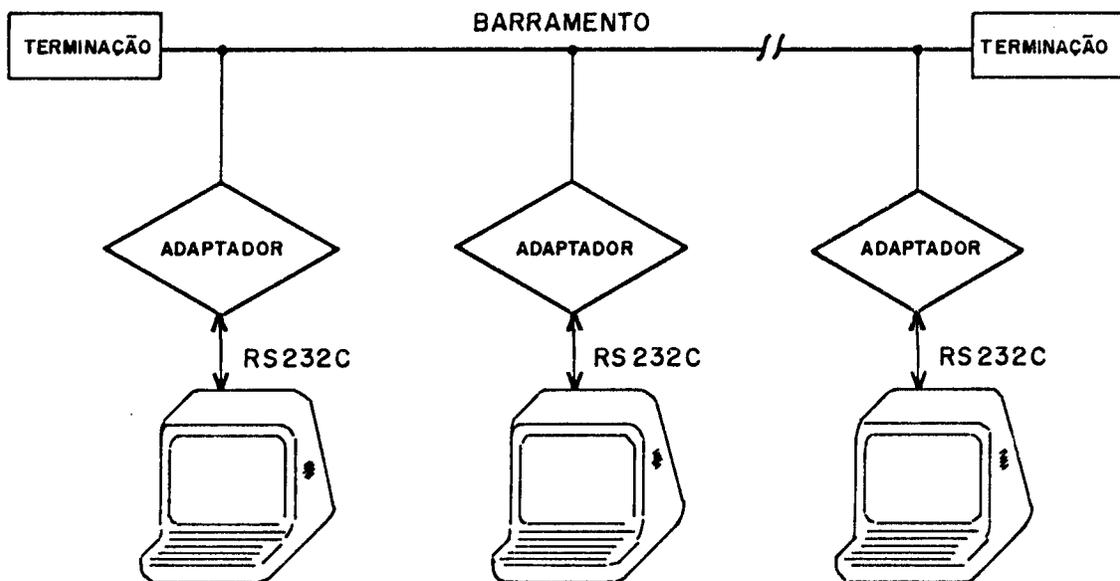


FIGURA 3.1 Hardware da Rede Local

O adaptador desenvolvido foi baseado na descrição de /KEN 81/. Este adaptador faz com que os dados que são enviados por uma estação sejam recebidos por todas as estações da rede inclusive pela estação transmissora. Isto tornou possível ao software detectar a existência de colisão na rede. A interface de comunicação serial usada, ACIA (Asynchronous Communications Interface Adapter), está descrita em /ROC 84/.

### 3.1 Descrição do Adaptador

O circuito mostrado na figura 3.2 foi embutido no conector DB25 da interface de comunicação serial RS232-C, formando assim o adaptador da rede local.

O canal de comunicação está composto por 3 fios: sinal de dados, sinal de controle e retorno elétrico (terra). Cada adaptador é composto basicamente por 4 amplificadores operacionais, realizando funções de "drivers" e "receivers" para os sinais de dados e controle.

Quando uma estação da rede deseja transmitir, faz com que o sinal "Request to Send" (RTS) seja acionado (pino 4), imediatamente o sinal "Clear to Send" (CTS-pino 5) também é acionado, habilitando desta forma a transmissão de dados. O acionamento do sinal RTS em uma estação da rede faz com que o sinal "Data Carrier Detect" (DCD-pino 8) seja ativado em todas as estações da rede, habilitando a recepção de dados. O sinal DCD é utilizado como indicação de linha ocupada. Após isto todos os dados que forem transmitidos, sinal "Transmitted Data" (TXD-pino 2), por uma estação esta rão sendo recebidos, sinal "Received Data" (RXD-pino 3), por todas as estações da rede.

O adaptador desenvolvido permite uma taxa de transmissão máxima de 100 Kbps, no entanto devido a limitações impostas pela interface de comunicação serial usada, a taxa de transmissão máxima da rede é de 19,2 Kbps.

O número máximo de estações que podem ser conectadas à rede está estimado em 10.

Adaptador para Rede Local

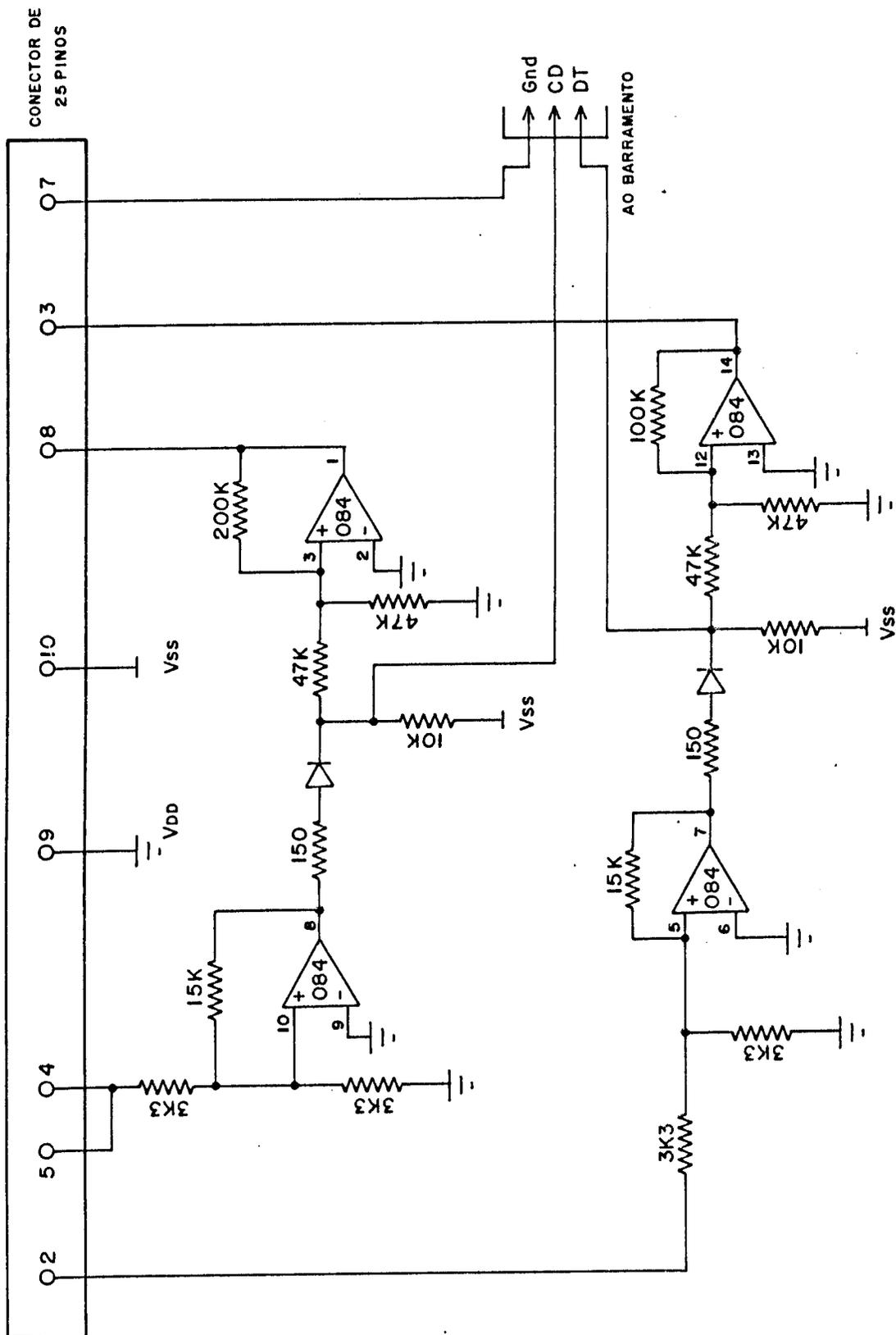


Figura 3.2 - Adaptador da Rede.

#### 4 NÍVEL DE ENLACE

Na tentativa de seguir padronizações, em especial da Organização de Padronização Internacional, ISO ("International Standards Organization"), o segundo nível de comunicação proposto baseia-se no modelo de referência para interconexão de sistemas abertos OSI/RM ("Open Systems Interconnection Reference Model"), mantendo equivalência aos níveis 2 e 3 do modelo referido.

O segundo nível de comunicação, que chamaremos Nível de Enlace (NE) tem como responsabilidade principal o controle de transferência das mensagens do nível imediatamente superior. Para que o NE possa enviar mensagens para a rede é necessário que exista uma política de acesso ao meio de comunicação. O método que foi usado é o CSMA/CD ("Carrier Sense Multiple Access/Colision Detect") /MET 76/. Neste método a portadora inicialmente é sentida e estando desocupada o NE inicia a transmissão. A seguir é verificado a existência de colisão (há colisão se o que uma estação está enviando está sendo recebido por ela mesma de forma não consistente). No caso em que há colisão a transmissão é interrompida e uma nova tentativa de transmissão é realizada.

O NE transmite os dados byte a byte para uma interface serial de comunicação com padrão RS-232C. A interação do NE com o hardware é feita através da monitoração dos sinais que são fornecidos por essa interface /ROC 85/. Estes sinais são registrados na posição de memória correspondente ao byte de estado da interface de comunicação. Uma outra posição de memória armazena os dados que são enviados ou retirados da interface.

No byte de estado são mantidas as seguintes informações necessárias ao NE:

RDRF - "Receive Data Register Full"

Indica o estado da recepção:

0 - dado recebido já foi lido, ou seja, não chegou  
dado novo

1 - dado recebido ainda não foi lido

TDRE - "Transmit Data Register Empty"

Indica o estado da transmissão:

0 - dado ainda não foi transmitido, nova transmissão não habilitada.

1 - dado já foi transmitido, habilitada nova transmissão.

DCD - "Data Carrier Detect"

Indica o estado da linha de transmissão:

0 - linha de transmissão ocupada

1 - linha de transmissão livre

OVRN - "Overrun"

Indica se existe erro de overrun

0 - não detectado

1 - erro detectado

PE - "Parity"

Indica se existe erro de paridade

0 - não detectado

1 - erro detectado

Sendo este nível o responsável pela interação com o hardware, o seu desenvolvimento realizou-se de forma simples, procurando facilitar, assim, a execução de alterações quando da modificação do hardware.

O Nível de Enlace oferece seus serviços para o nível imediatamente superior através de duas primitivas: (TRANSMISSÃO e RECEPÇÃO) que serão descritas adiante.

#### 4.1 Estrutura de Dados

Todas as mensagens que fluem no sistema implementado tem passagem pelo NE, o qual realiza a operação desejada (Transmissão ou Recepção). Para isto é utilizada uma estrutura de dados bem definida onde as mensagens são armazenadas.

A área de dados existente no NE serve de "buffer" tanto na recepção quanto na transmissão dos dados. Esta área do NE é composta por duas partes a saber: o cabeçalho da mensagem e o texto (dados) da mensagem.

O cabeçalho da mensagem é composto por seis bytes, dos quais quatro são vistos pelo NE. O restante do cabeçalho é de interesse do nível de transporte. Os campos do cabeçalho do NE são indicados na Tabela 4.1.

TABELA 4.1 Campos de cabeçalho do NE

CAMPO	TAMANHO	CONTEÚDO
Destino	1 byte	endereço destino da mensagem (0-255)
Origem	1 byte	endereço origem da mensagem (0-254)
Bytestx	2 bytes	número de bytes que compõe a mensagem (cabeçalho do nível de transporte mais dados).

O endereço destino (nº de estação) 255 é reservado para indicar mensagem de difusão (todas as estações podem receber a mensagem).

O endereço de origem é obtido na inicialização do sistema, quando informa-se o número da estação de trabalho.

O campo Bytestx, como indicado, controla a transmissão e/ou recepção do NE a partir do quinto byte do cabe-

çalho, onde estão informações que dizem respeito ao nível de transporte.

Todas as mensagens do sistema, independentemente do tipo (dados, controle), possuem um cabeçalho fixo que sempre é transmitido. Desta forma o NE transmite/recebe 4 bytes, mais os bytes indicados no campo Bytestx.

Os Dados da Mensagem compõe uma parte variável da mensagem, podendo ter de 0 à 255 bytes. Este tamanho foi escolhido em função das características da rede implementada, podendo ser mudado se necessário.

O NE tem ainda a função de anexar a cada mensagem a ele enviada o caracter de início de transmissão "SOH" (start of header).

A figura 4.1 mostra o formato de uma mensagem do nível de enlace.



FIGURA 4.1 Formato da mensagem do Nível de Enlace

#### 4.2 Método de Acesso

O sistema proposto tem como método de acesso, o protocolo de controle CSMA/CD. Este protocolo tem como características básicas, "ouvir antes de falar" e também "ouvir enquanto fala". Na primeira etapa do protocolo, antes de iniciar a transmissão, a estação verifica se não há ninguém usando o canal e só então inicia a transmissão (CSMA). Após o início da transmissão a estação tem condições de detectar

as colisões (CD), porque ao transmitir também está recebendo o que está no canal, podendo assim verificar se o que está na linha é o mesmo dado que foi transmitido.

No protocolo implementado a tentativa de detecção de colisão se dá nos cinco primeiros bytes do cabeçalho da mensagem.

Esta escolha decorre da existência do sinal DCD (Data Carrier Detect), que indica o estado do canal de transmissão. Este sinal é ativado por software através do registrador de comando da ACIA (Asynchronous Communications Interface Adapter).

O nível de enlace ativa este sinal quando detecta que o canal de transmissão está livre. O sinal ativado chega as outras estações indicando que o canal de transmissão está ocupado, sendo que qualquer tentativa de acesso à rede não é mais permitida, até que o canal de transmissão fique novamente livre. Desta forma as ocorrências de colisão são possíveis apenas quando duas ou mais estações recebem a informação de que o canal de transmissão está livre. Após a transmissão de uma mensagem o NE desativa o sinal DCD.

O NE quando inicia a transmissão do byte de início de transmissão "SOH", até o byte final do campo "Bytestx", verifica se o que foi recebido da linha é igual ao que foi transmitido. Quando nesta comparação ocorre uma diferença, a mesma é interpretada como colisão sendo a partir de então realizado o tratamento de colisão.

#### 4.2.1 Tratamento de colisões

Na ocorrência de colisão, quando da tentativa de transmissão de mensagens, as estações envolvidas entram em um período de espera randômico, implementado através de um busy wait. Este tempo randômico é conseguido através do registrador R ("REPRESH") da máquina Z-80.

O NE decrementa o valor do registrador R até chegar a zero e, então volta a tentativa de iniciar uma transmissão. À medida que as colisões vão ocorrendo o tempo de espera aumenta porque passa a ser o número de colisões ocorridos multiplicado pelo valor do registrador R, no instante da primeira colisão.

O tratamento de colisões prevê um número máximo de ocorrências durante a transmissão de uma mensagem do NE.

O registrador R tem o seu valor modificado constantemente, sendo isto uma característica do microprocessador Z-80.

#### 4.3 Primitivas

O Nível de Enlace fornece serviço ao nível de transporte, através de suas duas primitivas. Estas primitivas são controladas pelo próprio NE, o qual é chamado pelo nível de transporte que fornece como parâmetro de entrada, o código do serviço que deve ser executado. As primitivas do NE foram projetadas para prover os serviços básicos que são necessários para o nível de transporte poder executar a sua tarefa.

#### 4.3.1 Transmissão

A primitiva TRANSMISSÃO tem como função controlar o processo de transmissão de uma mensagem do nível de enlace e, retornar o resultado de sua execução.

Inicialmente a primitiva faz a verificação do estado da linha de transmissão através da leitura do estado da interface serial de comunicação. Enquanto o canal de transmissão estiver no estado ocupado e o tempo que a primitiva tiver para tentar a transmissão não se esgotar, irá permanecer verificando o estado da linha. Quando esgotar-se o tempo e a primitiva não conseguir a transmissão, voltará ao seu chamador indicando como código de retorno o valor 'FF'H (transmissão não realizada).

Após detectar o canal de transmissão livre, o nível de enlace envia um comando para a interface serial de comunicação. Este comando fará com que seja ativado o sinal RTS (Request to Send), cuja finalidade é sinalizar as outras estações que o canal de transmissão está ocupado.

A transmissão iniciará com o envio do caracter de início de transmissão (SOH) e dos quatro primeiros bytes do cabeçalho da mensagem. Nesta etapa são detectadas as colisões e tratadas da forma descrita anteriormente. Se a primitiva não conseguir executar de forma correta neste estágio, o código de retorno da primitiva será 'FF'H.

Após ter transmitido a parte referente ao cabeçalho da mensagem, a primitiva TRANSMISSÃO entrará na sua última etapa que é a transferência do texto da mensagem. O número de bytes que devem ser transferidos é indicado no campo "Bytestx". Durante esta etapa qualquer impossibilidade de transmissão que ocorrer implicará no término da execução da primitiva e no retorno do código de erro 'FF'H.

A primitiva terá sucesso quando todos os bytes da mensagem forem transmitidos de forma correta. O código de retorno para a execução correta é o valor '06'H (transmissão realizada).

Para a transmissão byte à byte é utilizada a rotina TXCOM. Esta rotina tenta a transmissão do byte especificado, um número determinado de vezes e no caso de não conseguir transmití-lo retorna um código de erro.

Após a transmissão de uma mensagem a primitiva TRANSMISSÃO espera pela confirmação de recebimento, que é enviada pela entidade de enlace remota.

A figura 4.2 mostra o diagrama de funcionamento da primitiva TRANSMISSÃO.

Antes de retornar ao nível de transporte o nível de enlace comanda a interface serial de comunicação para desativar o sinal RTS, liberando o canal de transmissão.

#### 4.3.2 Recepção

O objetivo desta primitiva é o de controlar a recepção de uma mensagem. Ela recebe como parâmetro de entrada o endereço da estação origem da mensagem a ser recebida e rejeita mensagens originadas por outras estações. A chamada da primitiva com o parâmetro de entrada igual a 'FF'H, indica que a mensagem a receber pode ter origem em qualquer estação.

Em uma primeira etapa, a primitiva RECEPÇÃO verifica o estado do canal de transmissão. Se estiver livre, tenta a recepção do primeiro byte de uma mensagem. Esta tentativa se repetirá enquanto não receber nada e o tempo de re-

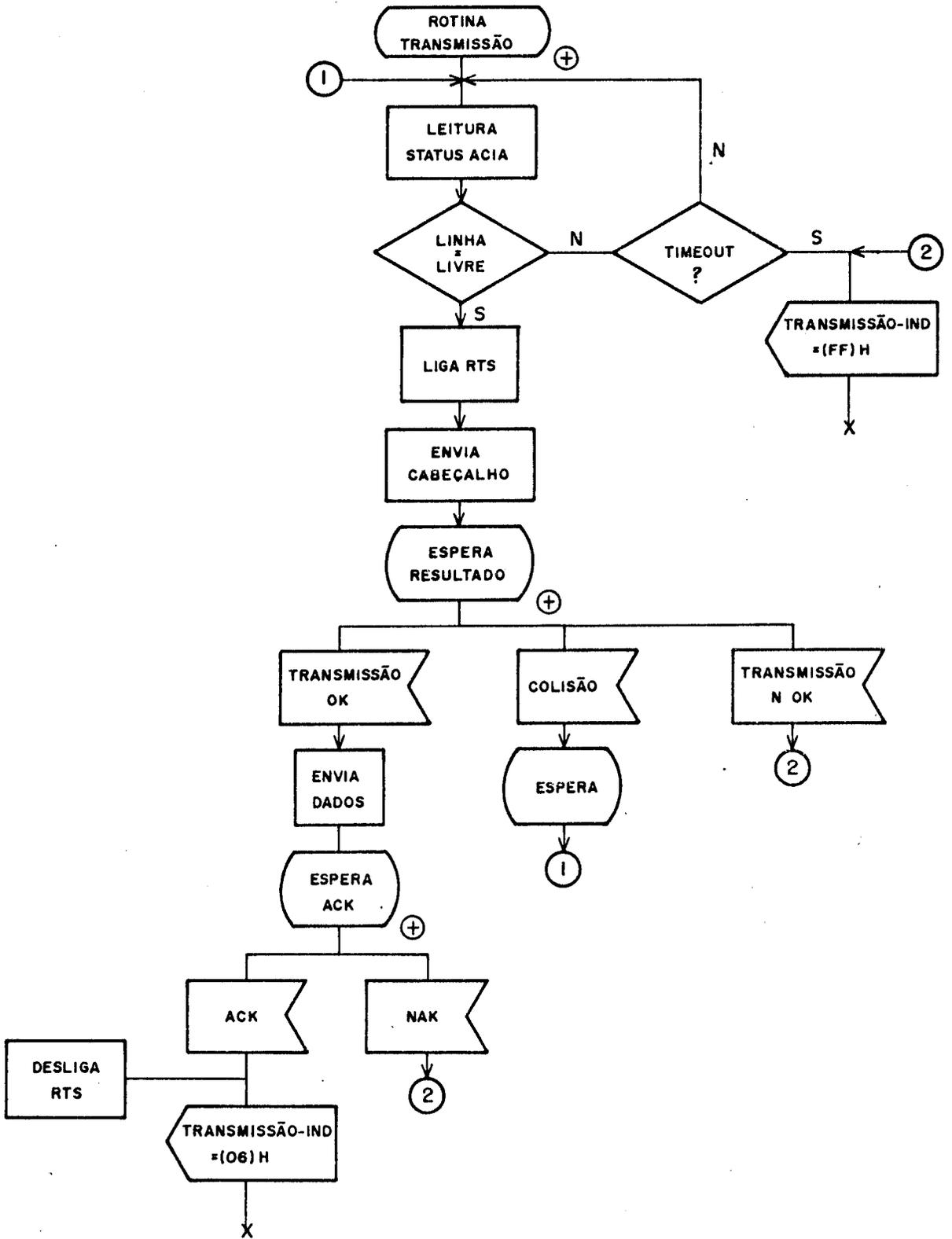


Figura 4.2 - Diagrama de funcionamento da rotina TRANSMISSÃO.

cepção não se esgotar. O esgotamento do tempo sem que a primitiva tenha concretizado o recebimento, ocasiona o término da execução e a devolução de um código de erro ('FF'H).

Na recepção do primeiro byte a primitiva verifica se o mesmo indica o início de uma mensagem ("SOH"). Não indicando, significa que o NE está interceptando uma mensagem que não deve ser capturada. A primitiva então volta a tentar a recepção de uma mensagem.

Quando ocorre o recebimento do byte "SOH", a primitiva tenta a recepção do cabeçalho da mensagem, que é composto pelos 4 bytes seguintes.

No cabeçalho é extraída a informação referente ao tamanho da mensagem a receber (campo "Bytestx") o qual controla, a execução da primitiva. Todos os bytes recebidos são colocados na área de dados do NE. No final da recepção é verificado se a mensagem recebida é endereçada a estação receptora, comparando o parâmetro de entrada da primitiva RECEPÇÃO e o campo "origem", da mensagem recebida. A execução correta desta primitiva retorna ao chamador o código 'Ø6'H, significando sucesso na recepção.

No nível de enlace são detectados os erros de comunicação tais como: "Overrun" e Paridade.

Estes erros são indicados no registrador de estado da interface serial de comunicação.

Após a recepção de uma mensagem a primitiva RECEPÇÃO envia um byte de confirmação de recebimento para a estação transmissora.

A figura 4.3 mostra o diagrama de funcionamento da primitiva RECEPÇÃO.

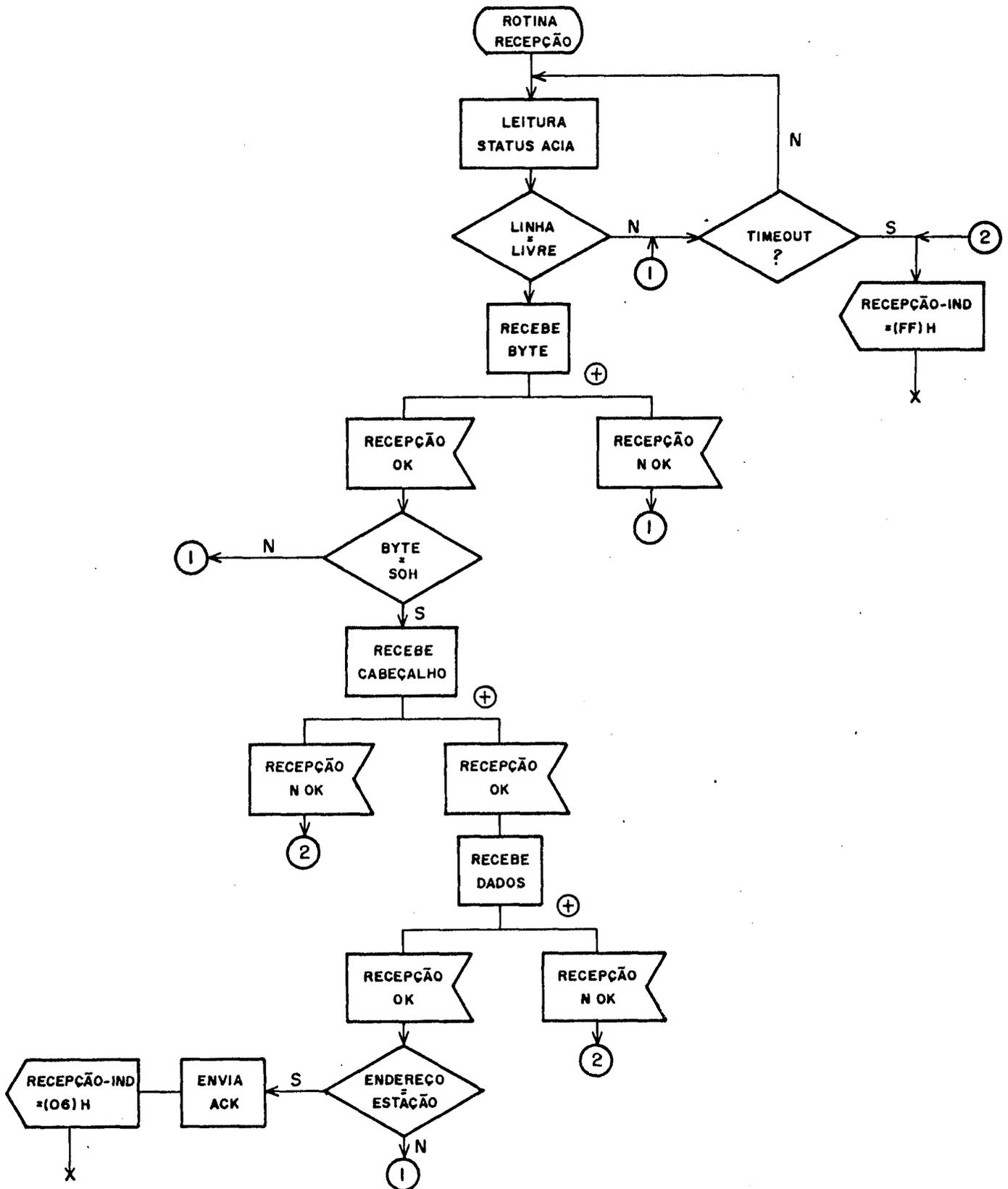


Figura 4.3 - Diagrama de funcionamento da primitiva RECEPTÃO.

## 5 NÍVEL DE TRANSPORTE: PRIMITIVAS

### 5.1 Considerações sobre o Nível de Transporte proposto

O Nível de Transporte (NT), visa fornecer um mecanismo confiável para a comunicação entre microcomputadores. Este nível tem a responsabilidade de permitir o estabelecimento de circuitos virtuais entre as estações comunicantes e possibilitar assim que a comunicação se dê sem erros e de forma ordenada.

Comparativamente com uma rede de longa distância, foram feitas algumas simplificações que se justificam devido ao enfoque do sistema ser redes locais. Estas mesmas simplificações são também face aos componentes da rede local implementada, microcomputadores monoprogamáveis, apesar disto o nível de transporte oferece um conjunto de serviços, que são representados pelas suas primitivas, permitindo abrir e fechar conexões, receber e enviar mensagens.

O NT é acionado quando o nível superior deseja se comunicar com outros sistemas remotos. A chamada é feita pelo nível superior que com isto atinge o ponto de entrada do TRANSPORTE.

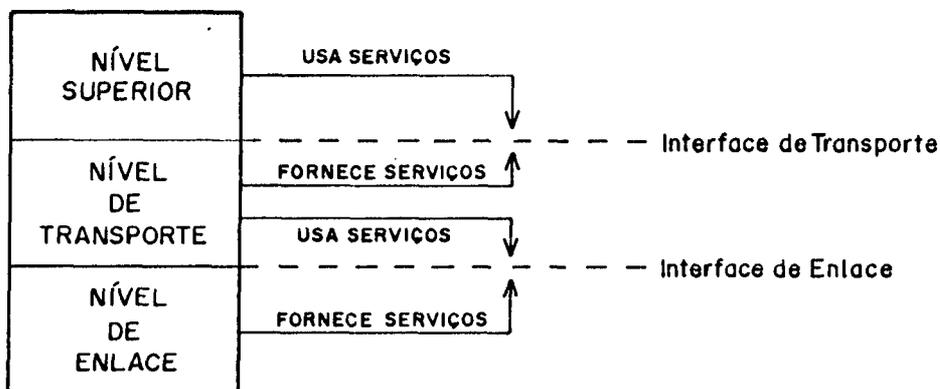


FIGURA 5.1 Relacionamento do nível de transporte com os níveis adjacentes

Como parâmetro da chamada o NT recebe um código indicando a primitiva que deve ser executada. Cada primitiva tem um conjunto de parâmetros que também são passados para o NT. O relacionamento do nível de transporte proposto, com os outros níveis adjacentes é mostrado na fig. 5.1.

## 5.2 Primitivas

A interface entre o nível superior (aplicação) e o sistema de comunicação é responsabilidade das primitivas do NT. Estas primitivas implementam os serviços que são oferecidos para o desenvolvimento de aplicações na rede.

### 5.2.1 Estabelecimento da Conexão

Nesta fase da comunicação as entidades envolvidas criam um canal virtual entre si, e estabelecem parâmetros para as fases posteriores da comunicação. O estabelecimento de conexão se dá de duas maneiras: o nível superior requisita ao NT uma conexão, ou a estação escutando a linha recebe um pedido de conexão.

#### 5.2.1.1 Primitiva CONECTA

Esta primitiva é requisitada pelo nível superior quando for necessário estabelecer um circuito virtual para troca de informações com outra estação remota. Para isto acontecer é necessário que a estação remota envolvida esteja em estado de ESCUTA. Esta condição é atingida quando o nível superior (na estação remota) ativa a primitiva PERGUNTA.

As conexões se realizam na relação uma para uma, ou seja, em um dado momento cada estação pode estar conecta

da com apenas uma outra estação remota. Desta forma para que o NT acione a primitiva CONECTA, é necessário que a estação envolvida esteja no estado LIVRE.

Como parâmetro de entrada a primitiva recebe o endereço da estação com a qual deve ser estabelecida a conexão. No retorno é devolvido ao nível superior o resultado da execução da primitiva CONECTA, o qual pode indicar:

- . sucesso: conexão estabelecida
- . insucesso: conexão não estabelecida
- . estado inicial da estação não é livre.

A ocorrência de um insucesso poderá se dar quando a estação remota não aceitar uma conexão quando não estiver ativa ou quando a estação origem não conseguir enviar o pedido de conexão.

#### 5.2.1.2 Primitiva PERGUNTA

Para que uma estação da rede local possa receber mensagens, é necessário (por falta de mecanismo de interrupção, no hardware) que a mesma esteja em estado de espera de mensagem. Isto corresponde a escutar a linha de comunicação a espera de pacotes. A primitiva PERGUNTA é ativada pelo nível superior para tentar a recepção de um pedido de conexão ou de uma mensagem de difusão (broadcast).

A execução da primitiva produz um resultado para o nível superior que pode ser:

- . nenhuma mensagem foi recebida
- . mensagem de difusão recebida
- . mensagem pedido de conexão recebida e confirmada.

No segundo e no terceiro caso, a primitiva devolve como parâmetro de saída o endereço da estação que enviou a mensagem. No caso de mensagem de difusão são informados ao nível superior, a posição do buffer onde está a mensagem e o tamanho da mesma.

Na mensagem de pedido de conexão, a primitiva informa ao nível superior que o pedido foi recebido e confirmado, ou seja, a conexão foi estabelecida com a estação remota.

### 5.2.2 Transferência de Dados

Após o estabelecimento da conexão, as estações envolvidas estão aptas a trocarem informações. Nesta fase de transferência de dados, as estações enviam e recebem mensagens.

#### 5.2.2.1 Primitiva ENVIA

A primitiva ENVIA é ativada pelo NT quando o nível superior deseja a transmissão de informações para a estação remota com a qual se conectou. O endereço da estação que deve receber a mensagem já foi informado quando o nível superior realizou o pedido de conexão. A estação que requisitou a primitiva CONECTA, tem a prioridade para enviar a mensagem inicial para a estação remota.

Toda mensagem que é enviada deve receber uma confirmação de recebimento (acknowledgment). Só após a confirmação do recebimento é que uma próxima mensagem pode ser enviada (ou seja, a janela de transmissão de mensagens é igual a 1).

Na chamada da primitiva o nível superior passa como parâmetros o endereço do buffer da mensagem e o tamanho da mensagem.

Recebe no retorno o resultado de execução da primitiva:

- . sucesso: mensagem transmitida e confirmada
- . insucesso: não recepção de confirmação ou problemas na transmissão.
- . sem conexão: estação não está conectada.

#### 5.2.2.2 Primitiva RECEBE

Para uma estação, que tenha estabelecido conexão com outra estação remota, poder receber dados enviados pela estação remota é necessário que o nível superior ative o NT, chamando a primitiva RECEBE.

As mensagens de dados seguem uma sequência rígida, que é verificada na execução desta primitiva. Para cada mensagem recebida corretamente é enviada uma mensagem de confirmação de recebimento, liberando assim a estação remota da necessidade de repetir a mensagem.

Como parâmetro de entrada, o nível superior, na chamada da primitiva, informa o endereço do buffer onde a mensagem deve ser colocada. Como parâmetro de saída a primitiva devolve o código do resultado de execução, o qual pode ser:

- . mensagem recebida
- . mensagem não recebida
- . desconexão pedida e realizada
- . estação não está conectada.

Quando a primitiva recebe alguma mensagem de dados, ela devolve para o nível superior o tamanho da mensagem que foi recebida.

### 5.2.3 Término da Conexão

Após a fase de transferência de dados entre as estações envolvidas em uma conexão, as mesmas devem se liberar para que novas conexões possam ser estabelecidas. Para isto o serviço de transporte oferece a primitiva DESCONECTA.

#### 5.2.3.1 Primitiva DESCONECTA

O objetivo da primitiva DESCONECTA é fazer com que as estações conectadas entre si sejam ordenadamente liberadas para novas conexões.

A estação que recebe o pedido de desconexão deve estar em um estado de recepção, para poder aceitar e confirmar a desconexão.

Como parâmetro de resposta a primitiva devolve para o nível superior o resultado da execução, o qual pode ser:

- . sucesso: a conexão com a estação remota foi encerrada.
- . insucesso: não foi possível encerrar a conexão (impossibilidade de transmissão)
- . a estação não está conectada.

#### 5.2.4 Difusão de Mensagens

Para possibilitar o envio de mensagem para todas as estações que estejam ativas e em um estado de escuta na rede, o NT oferece o serviço de difusão.

A primitiva DIFUNDE transmite uma mensagem para todos os receptores da rede. Este tipo de mensagem não necessita da fase de estabelecimento de conexão e, tampouco espera a recepção de mensagem de confirmação de recebimento.

Como parâmetro de entrada, a primitiva recebe o endereço do buffer onde está depositada a mensagem a ser difundida e o tamanho da mesma. O parâmetro de saída informa ao nível superior o resultado da execução da primitiva DIFUNDE.

- . sucesso: mensagem de difusão transmitida
- . insucesso: transmissão negativa.

#### 5.3 Cenário de uma Conexão

Para que as primitivas mostradas nos itens anteriores sejam melhor compreendidos no que diz respeito a sua sequência de ativação, é apresentado na figura 5.2 o cenário de uma conexão estabelecida e encerrada normalmente pelo usuário "1".

#### 5.4 Diagrama de Transição de Estados

Os estados de uma conexão de transporte sofrem modificações que são ocasionados pela ativação das primitivas de transporte.

Para finalizar este capítulo sobre as funções primitivas, é mostrado na figura 5.3 os estados de uma conexão de transporte e suas mudanças.

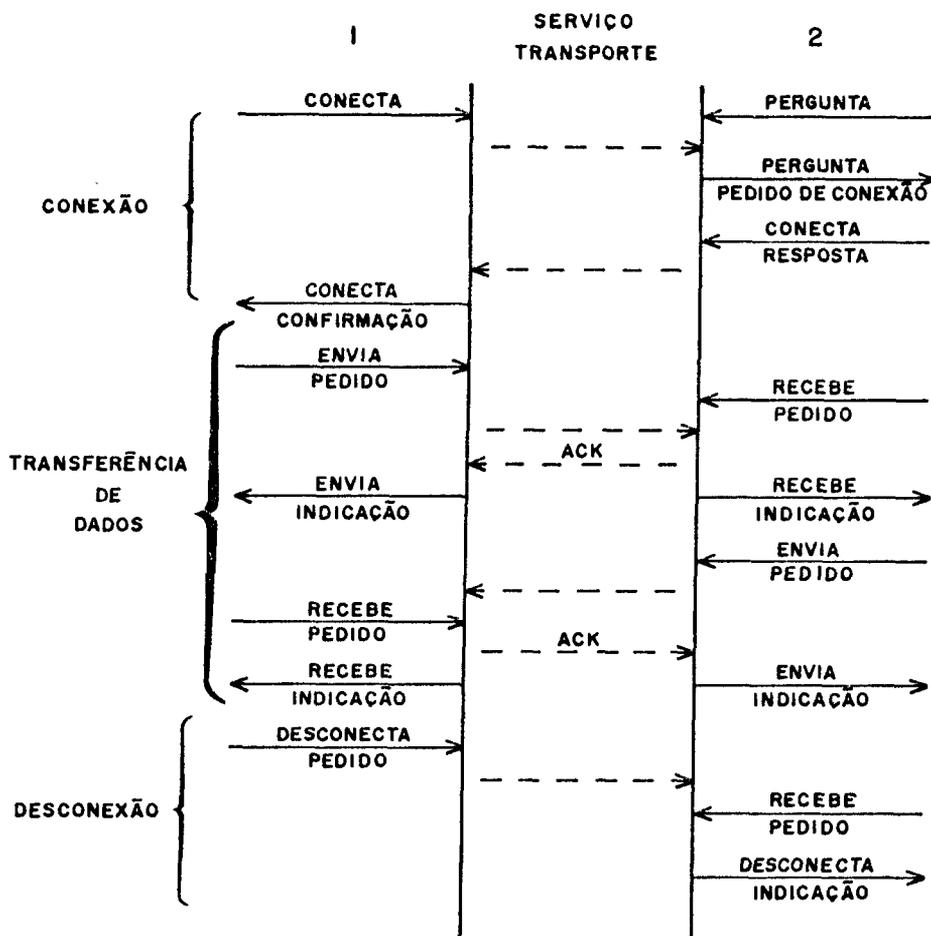


FIGURA 5.2 Cenário de uma conexão de transporte

A ausência de uma conexão é representada pelo estado LIVRE. A mudança deste estado é ocasionada quando existir um pedido de uma conexão.

Sempre que uma entidade de transporte requisita uma conexão com outra entidade de transporte, a primeira per-

manece no estado de CONEXÃO PENDENTE. Este estado é modificado pela recepção da confirmação da conexão ou por um "timeout" de pedido de conexão.

No estado CONECTADO a entidade de transporte realiza a sua fase de transferência de dados.

O encerramento de uma conexão de transporte é feito a partir do estado CONECTADO, fazendo com que as entidades envolvidas nesta conexão imigrem para o estado LIVRE.

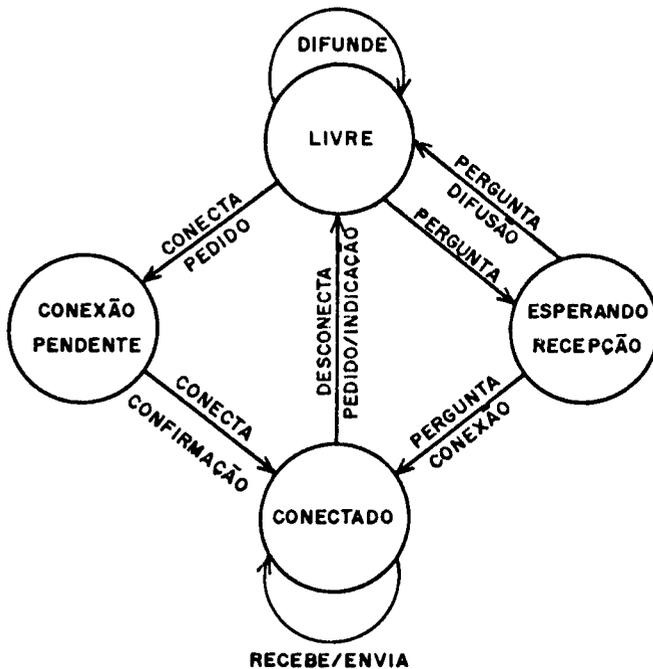


Figura 5.3 - Diagrama de transição de estados .

## 6 NÍVEL DE TRANSPORTE: DESCRIÇÃO DO PROTOCOLO

Neste capítulo descreve-se o funcionamento do protocolo de transporte, o qual é mostrado através das fases envolvidas na conexão de transporte.

São mostrados os procedimentos e as unidades de dados necessários para a implementação das funções primitivas.

O protocolo de transporte é apresentado através de uma Linguagem de Descrição e Especificação (SDL), descrita em /CHA 83/, adaptada para o ambiente da implementação do nível de transporte em questão. Os símbolos usados nos diagramas SDL estão no Anexo 1. A figura 6.1 mostra o diagrama de interação da entidade de transporte implementada, que se comunica com o ambiente através de dois canais.

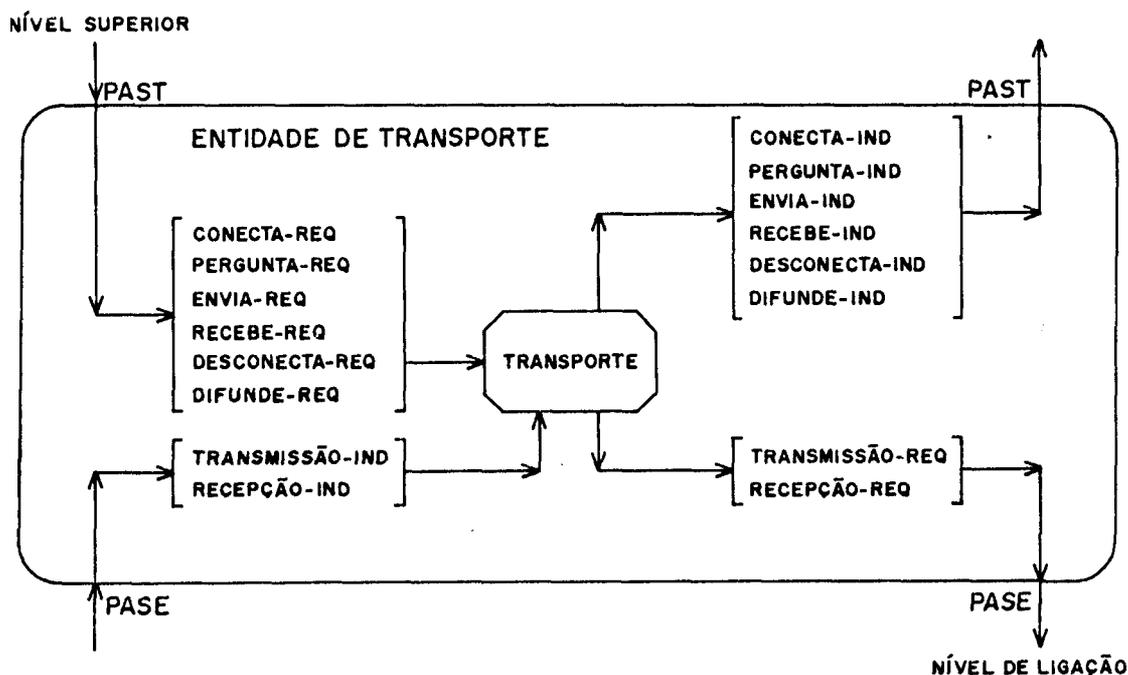


FIGURA 6.1 Diagrama de interação

Os canais são conhecidos como Ponto de acesso aos serviços de transporte (PAST) e Ponto de acesso aos serviços de enlace (PASE). São considerados como ambiente, o nível superior e o nível de enlace. As requisições e respostas existentes no ambiente são consideradas como sinais, os quais podem ser:

a. Sinais de transporte:

nível superior para nível transporte =

CONNECTA\_REQ, PERGUNTA\_REQ, ENVIA\_REQ,  
RECEBE\_REQ, DESCONNECTA\_REQ, DIFUNDE\_REQ

nível transporte para nível superior =

CONNECTA\_IND, PERGUNTA\_IND.  
ENVIA\_IND, RECEBE\_IND, DESCONNECTA\_IND.  
DIFUNDE\_IND

b. Sinais de enlace

nível de transporte para nível de enlace =

TRANSMISSÃO\_REQ, RECEPÇÃO\_REQ

nível de enlace para nível de transporte =

TRANSMISSÃO-IND, RECEPÇÃO\_IND

As requisições são atendidas pelo chamamento das rotinas que implementam os serviços, sendo que o retorno das rotinas causa os sinais de resposta.

## 6.1 Endereçamento

O nível de transporte (NT) implementado considera que as estações estão endereçados com valores entre (00)H a (FE)H. Quando chamado para a abertura de uma conexão de transporte a entidade de transporte deve receber um destes endereços. O Serviço de Transporte (ST) permite que uma estação

possa endereçar uma determinada estação ou todas as estações da rede (serviço de difusão), neste caso o endereço usado pelo NT é (FF)H.

## 6.2 Fase de Conexão

### 6.2.1 Unidades de Dados

Na fase de conexão são utilizadas unidades de dados que tem o formato indicado na figura 6.2.

END.DESTINO	END.ORIGEM	CONT. BYTES	TIPO
-------------	------------	----------------	------

FIGURA 6.2 Formato das unidades de dados

Os campos END.DESTINO, END.ORIGEM, CONT.BYTES e TIPO, são comuns a todas as unidades de dados. Os três primeiros estão descritos na seção 4.1. O campo TIPO serve para indicar o tipo da mensagem, conforme é mostrado na Tabela 6.1.

TABELA 6.1 Identificação das unidades de dados do protocolo de transporte

TIPO	SIGNIFICADO
(CØ)H	Pedido de Conexão (CR)
(CC)H	Confirmação de conexão (CC)
(00)H	Dados Ø (DØ)
(01)H	Dados 1 (D1)
(AØ)H	ACK Ø (AØ)
(A1)H	ACK 1 (A1)
(DC)H	Pedido de desconexão (DR)
(DF)H	Difusão (DF)

Para que uma determinada estação efetue uma transferência de dados é necessário que a mesma estabeleça uma conexão com a remota. Para isto a estação, através do ST, envia a unidade de dados do tipo CONECTA\_REQ (CR).

A entidade transporte remota que recebe esta unidade de dados e concorda com o pedido tem que enviar como resposta a unidade de dados confirmação de conexão (CC).

## 6.2.2 Procedimentos

### 6.2.2.1 Estação com a Iniciativa

O nível TRANSPORTE ativado pelo sinal CONECTA\_REQ, efetua a chamada da primitiva CONECTA. A primitiva CONECTA recebe na sua chamada o endereço da estação destino do pedido de conexão. Para que a estação possa efetuar um pedido de conexão é necessário que a mesma esteja no estado LIVRE (sem conexão); esta verificação é feita pela primitiva CONECTA e quando a estação não está livre o sinal de retorno CONECTA\_IND volta ao ambiente, indicando a condição da estação. Após esta verificação, CONECTA monta a mensagem de pedido de conexão (CR), e transfere esta para a área de dados do nível de enlace (NE). Neste instante CONECTA requisita a transmissão da mensagem através do sinal TRANSMISSÃO\_REQ que aciona o nível de enlace para executar uma transmissão. O estado de CONECTA neste momento é ESPERANDO TRANSMISSÃO.

O retorno do NE acontece pelo sinal TRANSMISSÃO\_IND e pode ter um dos significados:

- (FF)H - transmissão negativa
- (06)H - transmissão positiva

Se a indicação é negativa, a entidade transporte, através da primitiva CONECTA, continua na tentativa de estabelecimento de conexão, enquanto o "timeout" não expira. Acontecendo o término do tempo de tentativa, a primitiva atualiza o indicador de retorno CONECTA\_IND com a condição de insucesso.

Sendo a transmissão efetuada com sucesso a primitiva CONECTA prepara-se para requisitar a recepção de uma mensagem que deve ser a mensagem de confirmação do pedido de conexão e envia, através da chamada do NE, o sinal RECEPÇÃO\_REQ. O estado da primitiva passa a ser ESPERANDO RECEPÇÃO.

A indicação retornada pelo NE através do sinal RECEPÇÃO-IND pode indicar uma das seguintes condições:

- . (FF)H - recepção negativa
- . (06)H - recepção positiva

a condição (FF)H é interpretada como não recepção da mensagem transmitida para a estação remota, que significa a necessidade de retransmissão da mensagem de pedido de conexão. O número de retransmissões é controlado por um "timeout" que quando expirado faz com que CONECTA retorne no sinal CONECTA\_IND a condição de insucesso.

A condição (06)H faz com que a primitiva CONECTA verifique o tipo de mensagem recebida. A primitiva só aceita mensagem do tipo (CC)H, confirmação de conexão. Quando a mensagem recebida não for igual a esperada CONECTA interpreta isto como sendo pedido de retransmissão e passa a executar os mesmos procedimentos de retransmissão já descritos anteriormente.

O recebimento de mensagem do tipo (CC)H indica a primitiva CONECTA que devem ser inicializados os apontadores de controle da fase de transferência de dados e armazenado o endereço da estação remota. Neste momento a estação passa para o estado CONECTADO.

Cada estação possui dois apontadores de controle da fase de transferência de dados, descritos a seguir.

TXCTRL - apontador de controle de transmissão da fase de transferência de dados, indica o tipo da mensagem de dados a ser transmitida. (00)HR(01)H.

RXCTRL - apontador de controle de recepção da fase de transferência de dados: indica o tipo da mensagem de dados a ser recebida (00)HR(01)H.

O diagrama mostrado na figura 6.3 descreve os procedimentos executados pela estação com a iniciativa da conexão.

### 6.2.2.2 Estação Consultada

Para que uma conexão de transporte seja estabelecida entre duas entidades de transporte é necessário que uma destas esteja no estado de ESCUTA. Isto é conseguido quando a estação consultada ativa o processo TRANSPORTE através do sinal PERGUNTA\_REQ, que traduzirá em chamada da primitiva de escuta PERGUNTA. O estado da estação neste momento é LIVRE.

A primitiva PERGUNTA envia um sinal RECEPÇÃO-REQ, acionando o nível de enlace para a tentativa de recepção de uma mensagem originada por qualquer outra estação da rede. PERGUNTA neste instante entra no estado ESPERANDO RECEPÇÃO. No retorno o nível de enlace envia o sinal RECEPÇÃO\_IND que pode ter o seguinte significado:

- . (FF)H - recepção negativa
- . (06)H - recepção positiva

No primeiro caso, PERGUNTA tenta novas recepções até que expire o "timeout" associado a esta primitiva. A recepção de uma mensagem deve ser identificada pela primitiva PERGUNTA através da interpretação de tipo da mensagem recebida. PERGUNTA pode aceitar dois tipos de mensagem, a saber:

- . (DF)H - Difusão;
- . (CØ)H - Pedido de conexão

Uma mensagem de difusão é apenas indicada pelo sinal de retorno PERGUNTA\_IND que sai da entidade de transporte para o ambiente através do ponto de acesso PAST. Quando a primitiva PERGUNTA identifica uma mensagem de pedido de conexão, prepara a transmissão de uma mensagem de confirmação de conexão (CC)H, ativando o nível de enlace para a transmissão da mensagem e passando para o estado ESPERANDO TRANSMISSÃO. O resultado que o nível de enlace retorna no sinal TRANSMISSÃO\_IND não pode significar:

- . (FF)H - transmissão negativa,
- . (06)H - transmissão positiva.

Quando a transmissão não é efetuada a primitiva PERGUNTA deve voltar à tentativa de recepção de mensagens. Se a estação com iniciativa de conexão, quando não recebe a confirmação, volta a retransmitir o pedido de conexão é necessário que a estação consultada volte a estar no estado de escuta para que a retransmissão seja recebida.

A transmissão positiva de uma mensagem de confirmação de conexão, implica na inicialização das variáveis de controle da fase de transferência dos dados, já mencionados anteriormente, sendo que a estação entra no estado CONECTADO.

O retorno da primitiva PERGUNTA faz com que o sinal PERGUNTA\_IND indique um dos seguintes resultados:

- . Conexão
- . Difusão
- . Recebimento negativo

O diagrama da figura 6.4 mostra os procedimentos da estação consultada, no estabelecimento de conexão.

### 6.3 Fase de Transferência de Dados

#### 6.3.1 Unidades de Dados

Para esta fase do protocolo de transporte, duas unidades de dados são trocadas entre as entidades transporte participantes da conexão: Dados e Ack. A unidade de dados DADOS (DØ, D1), cujo formato é apresentado na figura 6.5 contém os dados que o usuário do serviço de transporte quer trans

mitir, através do pedido ENVIA\_REQ que ativa a primitiva ENVIA; a unidade de dados ACK, cujo formato é apresentado na figura 6.2, apenas contém o reconhecimento positivo da unidade de dados recebida da estação remota.

END.DEST.	END. ORIG.	CONT. BYTES	TIPO	CONTROLE	DADOS	CKS
-----------	------------	----------------	------	----------	-------	-----

FIGURA 6.5 Formato da unidade de dados DADOS

Os campos END.DEST, END.ORIG, CONT.BYTES são descritos na seção 4.1. O campo TIPO pode assumir os valores (00)H e (01)H (DADOSØ e DADOS1), dependendo do conteúdo da variável de controle TXCTRL. O campo CONTROLE indica se a mensagem tem continuação ou não. O campo CKS contém o checksum do campo DADOS. No campo DADOS estão os dados que devem ser recebidos pelo usuário do serviço de transporte na estação remota.

### 6.3.2 Procedimentos

#### 6.3.2.1 Estação Transmissora

Após a fase de estabelecimento de conexão, a estação com a iniciativa da conexão executa a fase de transferência de dados, iniciando como estação transmissora. O TRANSPORTE é ativado pelo pedido de transferência ENVIA\_REQ e efetua a chamada da primitiva ENVIA. Para que a primitiva ENVIA execute os procedimentos necessários à transmissão da mensagem e, a estação deve estar no estado CONECTADO. Esta verificação é realizada no início de ENVIA. Caso seja confirmada a não conexão, é retornada a indicação de sem conexão, no sinal ENVIA\_IND.

Confirmada a existência de conexão, a primitiva executa os procedimentos de preparação do envio da mensagem: prepara cabeçalho da mensagem; particiona a mensagem (quando maior que 256 bytes); transfere a mensagem para o buffer de transmissão do nível de enlace e calcula o checksum da mensagem. Uma vez preparada a mensagem a ser transmitida pelo nível de enlace, a primitiva ENVIA requisita a transmissão da mensagem através do sinal TRANSMISSÃO\_REQ que ativa o nível de enlace. ENVIA neste momento permanece no estado ESPERANDO TRANSMISSÃO.

O retorno desta requisição é indicado no sinal TRANSMISSÃO\_IND, que pode assumir dois valores:

(FF)H - transmissão negativa

(Ø6)H - transmissão positiva

O NT tentará a transmissão de uma mensagem até que esgote o timeout de envio de mensagens. Persistindo a não transmissão indica ao nível superior, através do sinal ENVIA\_IND, o insucesso na execução da primitiva. Após uma transmissão bem sucedida a primitiva ENVIA deve receber a confirmação de recebimento da mensagem enviada, que é transmitida pela estação remota. Para isto ENVIA ativa o nível de enlace com o sinal RECEPÇÃO\_REQ e fica no estado ESPERANDO RECEPÇÃO. O sinal RECEPÇÃO\_IND retorna do nível de enlace com o resultado da recepção: (FF)H ou (Ø6)H. Quando a primitiva ENVIA verifica a não recepção da mensagem de ACK, ela interpreta isto como um pedido de retransmissão. Passa então a retransmitir a mensagem até que receba o ACK correspondente, ou até que esgote o timeout associado.

A recepção da mensagem faz com que a primitiva identifique a mensagem recebida e verifique se a mesma é a esperada. Se o resultado da verificação é negativo ENVIA passa para a fase de retransmissão.

Na recepção do ACK correspondente a mensagem enviada pelo NT, a primitiva ENVIA acerta as variáveis de controle da fase de transferência de dados e verifica se a mensagem tem continuação (mensagem do NS maior do que 256 bytes). Quando a mensagem é maior que 256 bytes a primitiva ENVIA transmite-a em partes de 256 bytes, até completar a transferência de toda a mensagem.

O nível superior é informado do resultado da execução da primitiva através do sinal ENVIA\_IND.

O diagrama da figura 6.6 apresenta os procedimentos da primitiva ENVIA.

#### 6.3.2.2 Estação Receptora

Uma vez estabelecida a conexão, para que a entidade de Transporte possa receber uma mensagem da entidade remota é necessário que o nível superior ative o processo TRANSPORTE, através do envio do sinal RECEBE\_REQ\_ que resultará na chamada da rotina RECEBE. Esta rotina recebe a mensagem enviada pela estação remota. A estação deve estar no estado CONECTADO para que a primitiva RECEBE seja executada.

A primitiva RECEBE após verificar a existência de conexão ativa o nível de enlace, através do sinal RECEPÇÃO\_REQ\_, para indicar que deve ser executada uma recepção. Neste momento RECEBE passa para o estado ESPERANDO RECEPÇÃO.

O sinal de retorno, RECEPÇÃO\_IND, do nível de enlace, devolve o resultado da recepção de mensagens. Quando negativo, (FF)H, a primitiva RECEBE permanece na tentativa de recepção até que o timeout correspondente esgota.

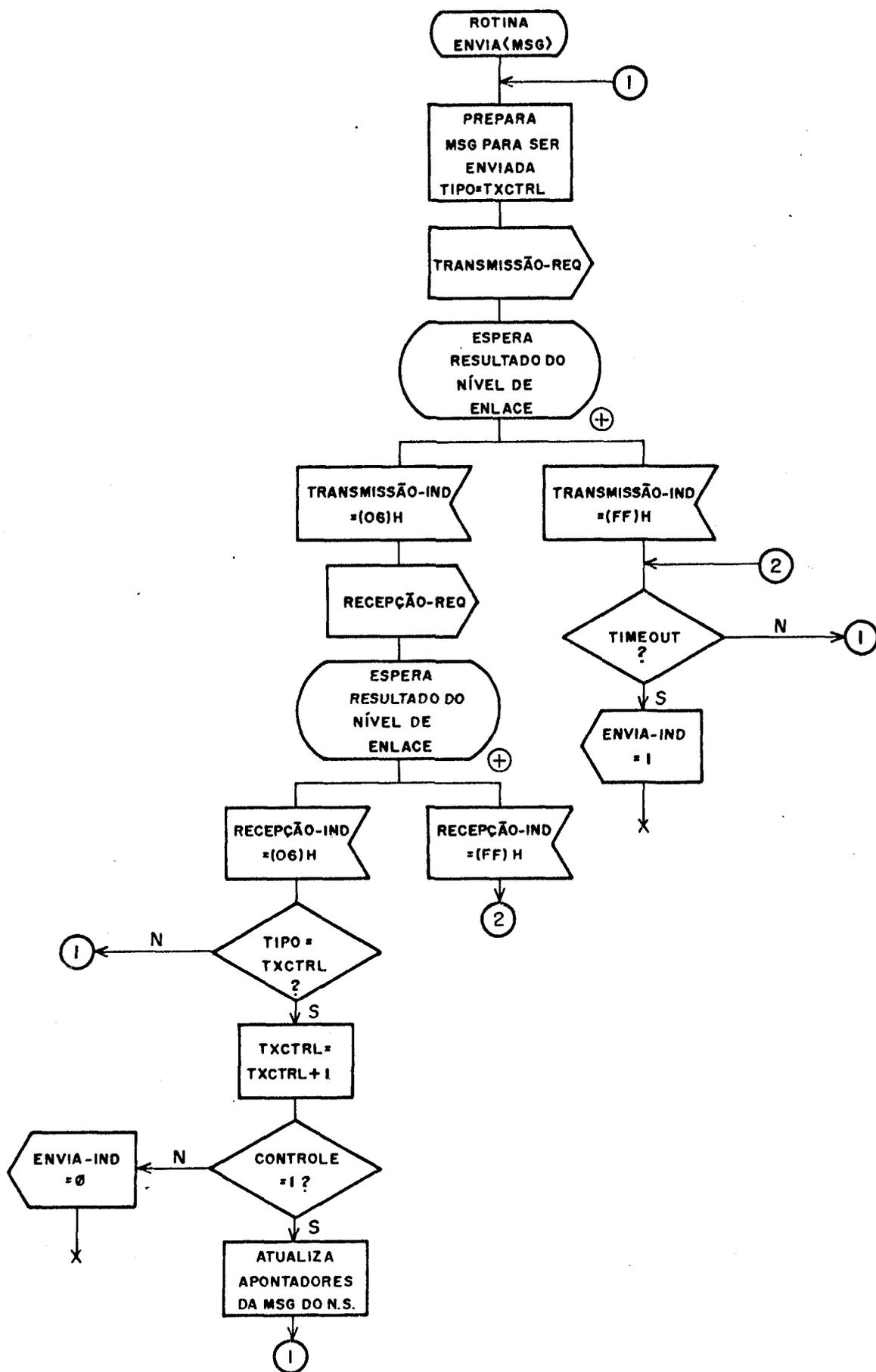


Figura 6.6 - Diagrama de execução da primitiva ENVIA .

Quando a recepção for positiva a primitiva deve executar o procedimento de verificação do tipo da mensagem recebida. O tipo da mensagem pode ser:

- 1 Mensagem recebida tem controle diferente do esperado:

Trata-se de uma mensagem duplicata retransmitida pela estação remota, visto que provavelmente não recebeu sua confirmação. A primitiva RECEBE prepara então uma mensagem de confirmação para ser retransmitida, e após isto comanda o nível de enlace para uma transmissão. Sendo a transmissão realizada com sucesso ou não a primitiva retorna a tentativa de recepção.

- 2 Mensagem recebida é pedido de conexão (CØ)H:

Trata-se da perda de uma mensagem de confirmação de conexão (CC)H pela estação com iniciativa de estabelecimento de conexão. A primitiva RECEBE deve repetir a confirmação de conexão. Após preparar a mensagem a ser repetida a primitiva ativa o nível de enlace, através do sinal TRANSMISSÃO\_REQ, para que seja transmitida a mensagem de confirmação de conexão. Para qualquer sinal de retorno TRANSMISSÃO\_IND (negativo ou positivo), a primitiva retorna para o estado ESPERANDO RECEPÇÃO, na tentativa de receber mensagem.

- 3 Mensagem recebida tem tipo diferente de (00)H, (01)H, (CØ)H, (DC)H

A primitiva permanece no estado ESPERANDO RECEPÇÃO até que o número de tentativas de recepção esgote.

- 4 Mensagem recebida é a esperada (TXCTRL=RXCTRL):

Neste caso a primitiva RECEBE verifica inicialmente se os dados estão corretos, ou seja, se não ocorreu erro de checksum. Se a mensagem recebida está incorreta, a

primitiva volta a tentar recepção de uma nova mensagem, sendo que a estação transmissora ao não receber a confirmação de recebimento irá retransmitir a mensagem que chegar incorreta.

Não existindo erros de checksum a primitiva prepara uma mensagem de confirmação de recebimento (ACK), e ativa o nível de enlace com o pedido de uma transmissão. Após completada a transmissão com resultado positivo, a primitiva verifica se a mensagem possui continuação, campo CONTROLE = 1, voltando ao estado de recepção da próxima mensagem da estação remota.

Após receber toda a mensagem de nível superior o sinal RECEBE\_IND retorna ao nível superior com a indicação de sucesso na execução da primitiva RECEBE.

Quando a transmissão da mensagem de confirmação de recebimento não for realizado com sucesso a primitiva volta a tentativa de recepção de mensagem. Neste caso a estação remota ao não receber a confirmação realizará a retransmissão da mensagem.

O não recebimento completo de uma mensagem, dentro do timeout estipulado para este procedimento, implica na indicação de insucesso através do sinal RECEBE\_IND.

A figura 6.7 apresenta o diagrama dos procedimentos executados pela primitiva RECEBE.

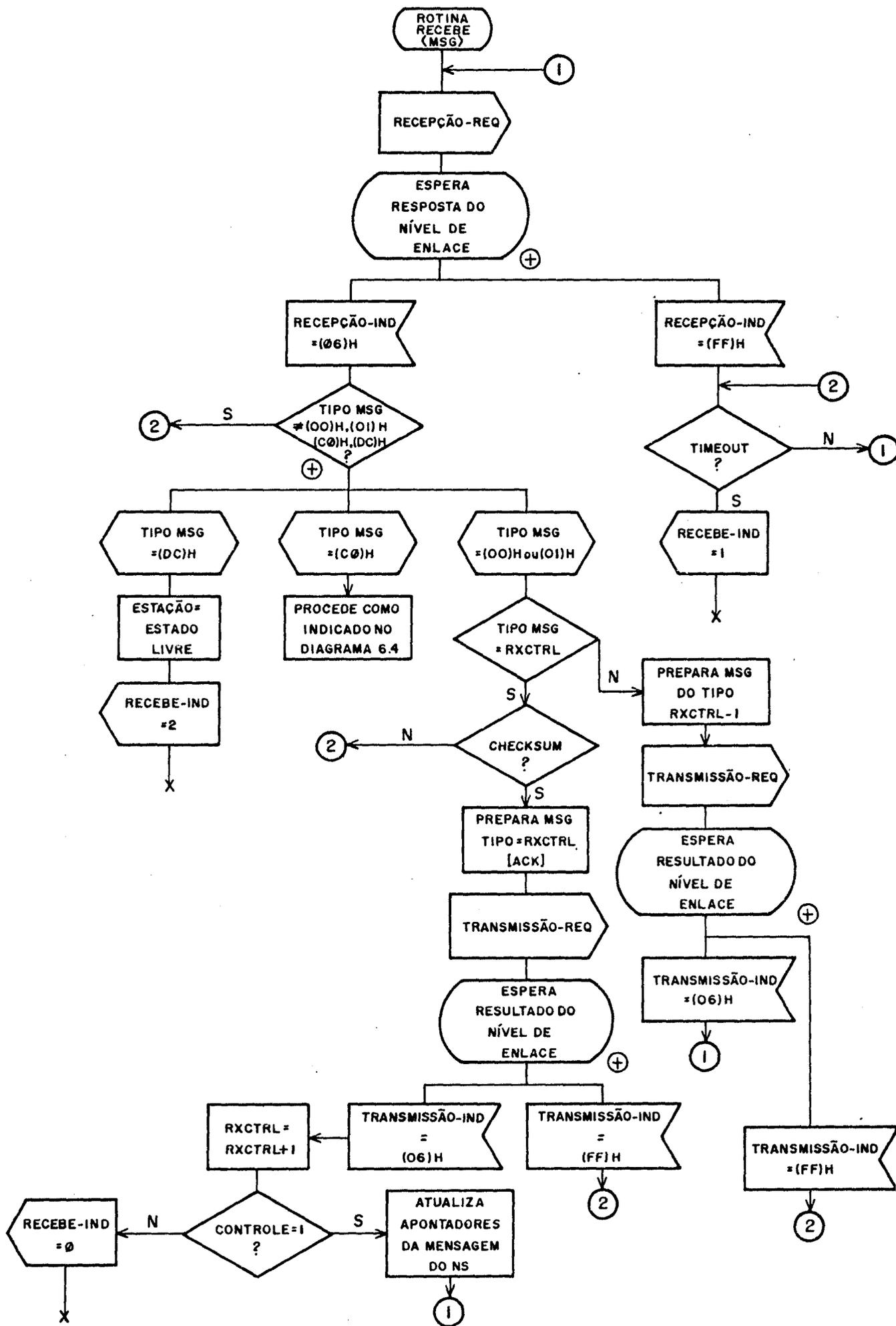


Figura 6.7 - Diagrama de execução da primitiva RECEBE.

## 6.4 Fase de Encerramento de Conexão

### 6.4.1 Unidades de Dados

Na fase de encerramento de conexão acontece o envio da unidade de dados do tipo DESCONECTA\_REQ, cujo formato é mostrado na figura 6.2.

### 6.4.2 Procedimentos

#### 6.4.2.1 Estação Solicitante

O nível de transporte é ativado pelo sinal DESCONECTA\_REQ e efetua a chamada do procedimento DESCONECTA.

A primitiva DESCONECTA verifica inicialmente se a solicitação é válida, ou seja, se existe conexão com alguma outra estação remota. Sendo a verificação negativa a solicitação é negada e o nível superior é informado disto através do sinal DESCONECTA\_IND.

Na existência de uma conexão a primitiva DESCONECTA prepara uma mensagem de pedido de desconexão, e requisita os serviços do nível de enlace através do sinal TRANSMISSÃO\_REQ. O retorno do nível de enlace traz em TRANSMISSÃO\_IND o resultado da execução do pedido feito ao nível de enlace, que pode ser:

$(FF)_H$  - transmissão negativa,

$(06)_H$  - transmissão positiva.

O resultado negativo implica em novas tentativas de envio da mensagem de desconexão, até que o timeout seja alcançado. Neste caso a primitiva retorna com a indicação de insucesso no sinal DESCONECTA\_IND.

Quando a primitiva `DESCONECTA` consegue realizar o envio da mensagem de desconexão ela deve reinicializar seus apontadores de estado. Isto é feito através da passagem do estado `CONECTADO` para o estado `LIVRE`. Deste modo novas conexões podem ser estabelecidas pela estação. O diagrama da figura 6.8 apresenta os procedimentos de encerramento de conexão para a estação solicitante.

#### 6.4.2.2 Estação Consultada

Para a estação receber um pedido de desconexão é necessário que a mesma esteja executando a primitiva `RECEBE` no estado `ESPERANDO RECEPÇÃO`. Com estas condições satisfeitos quando a primitiva `RECEBE` identificar um pedido de desconexão, ela fará a estação mudar do estado `CONECTADO` para o estado `LIVRE`, indicando ao nível superior a existência da desconexão. A descrição deste procedimento é mostrado na figura 6.7.

#### 6.5 Difusão de Mensagens

O nível de transporte oferece um serviço de difusão para que mensagens possam ser transmitidas de uma estação para outra, sem a necessidade da fase de abertura de conexão.

O serviço é realizado pela primitiva `DIFUNDE`, que tenta a transmissão da mensagem especificada até conseguir enviar ou até o time out de transmissão terminar. Após o envio da mensagem a primitiva retorne com o sinal `DIFUNDE_IND` informando o sucesso na execução da difusão da mensagem.

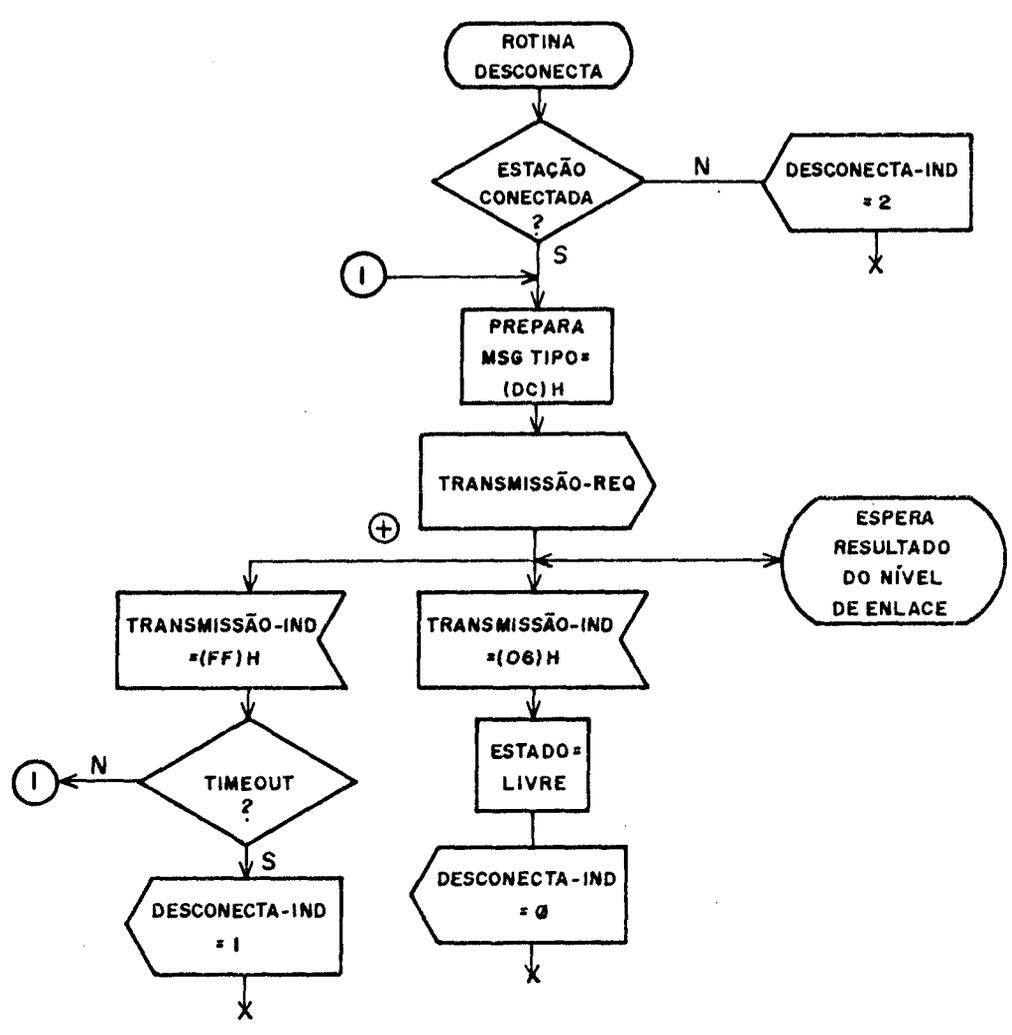


Figura 6.8 - Diagrama de execução da primitiva DESCONNECTA .

As estações que estiverem no estado LIVRE e executando a primitiva PERGUNTA estarão aptas a receber uma mensagem de difusão. A estação receptora da mensagem não tem necessidade de enviar a confirmação de recebimento (ACK).

O diagrama dos procedimentos da primitiva DIFUNDE é apresentado na figura 6.9. Os procedimentos da estação receptora são mostrados no diagrama da figura 6.4.

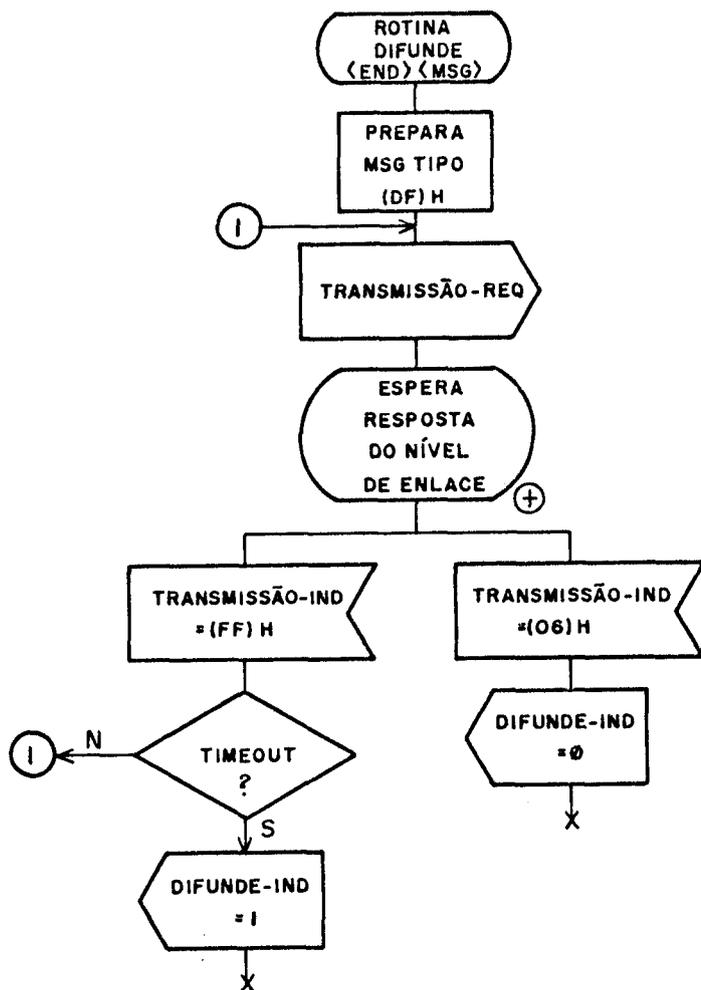


FIGURA 6.9 Diagrama de execução da primitiva DIFUNDE

## 6.6 Controle de Erros

Por ocasião da recepção de uma mensagem de dados o nível de transporte faz uma verificação na parte de dados da mensagem.

Para implementar o controle de erros é usado um "check sum", que é a soma (Módulo 8) de todos os bytes de dados da mensagem. Este check sum é anexado à mensagem na última posição dos dados, no momento da transmissão.

Na recepção é feita a soma de todos os bytes da mensagem de dados e comparada com o "check sum" recebido se coincidem a mensagem está correta. A existência de um erro de check-sum fará com que seja pedido à estação remota, a retransmissão da mensagem com erro.

## 6.7 Segmentação/Junção

As mensagens do nível superior podem ter um tamanho variável. Desta forma é necessário que o nível de transporte execute o serviço de segmentação das mensagens vindos do nível superior para enviá-las ao nível de enlace para a transmissão. As mensagens que o nível de enlace transmite tem um tamanho máximo de 256 bytes de dados. Cada invocação da primitiva ENVIA pode resultar em uma ou mais transmissões do nível de enlace. Na primitiva RECEBE do nível de transporte é realizado o serviço de junção das mensagens de transporte até formar a mensagem do nível superior. O nível de transporte não possui buffer de transmissão/recepção; as mensagens são transferidas do buffer do nível superior para o buffer do nível de enlace, na transmissão, e do buffer do nível de enlace para o buffer do nível superior, na recepção. O controle desta transferência é chamado aqui de segmentação/junção e é realizado pelo nível de transporte.

## 7 UTILIZAÇÃO DO SOFTWARE DE COMUNICAÇÃO

Muitas implementações de redes locais se restringem à rede física propriamente dita e aos primeiros dois níveis do software de comunicação, resultando em projetos de pouca aplicação prática. Para o desenvolvimento de aplicações seria muito interessante que o sistema implementado (hardware e software de comunicação) pudesse ser utilizado a partir de uma linguagem de alto nível.

Foi pensando nisto que resolveu-se juntar o hardware e o software de comunicação da rede local implementada com o sistema MUMPS já implementado em vários computadores. Este trabalho deverá ser realizado em uma próxima etapa, não como parte desta dissertação. Isto permitirá tirar proveito do domínio tecnológico já existente sobre o sistema MUMPS /TOR 82/, oferecendo aos usuários uma rede local MUMPS com as ferramentas necessárias para o desenvolvimento de aplicações. Desta forma a aplicação prática da rede seria muito facilitada, pois se disporia de recursos adequados para criação de programas, e armazenamento de dados.

Este capítulo tem a finalidade de ilustrar a utilização do sistema de comunicação implementado, descrevendo uma possível forma de incorporá-lo ao sistema MUMPS. A seguir será mostrado como poderia ficar o sistema resultante.

Basicamente, o usuário enxergaria um ambiente MUMPS, composto de seu pedaço (partição) de memória, suas variáveis locais, seu programa, seus recursos locais e o espaço no qual armazena os programas e arquivos (variáveis globais). Um exemplo deverá ajudar a esclarecer melhor a situação: supor uma rede local composta de três micros independentes, cada qual com seus discos próprios e periféricos próprios, conforme é mostrado na figura 7.1.

Cada micro constitui uma estação de trabalho independente, podendo-se ter um aplicativo diferente rodando em cada estação.

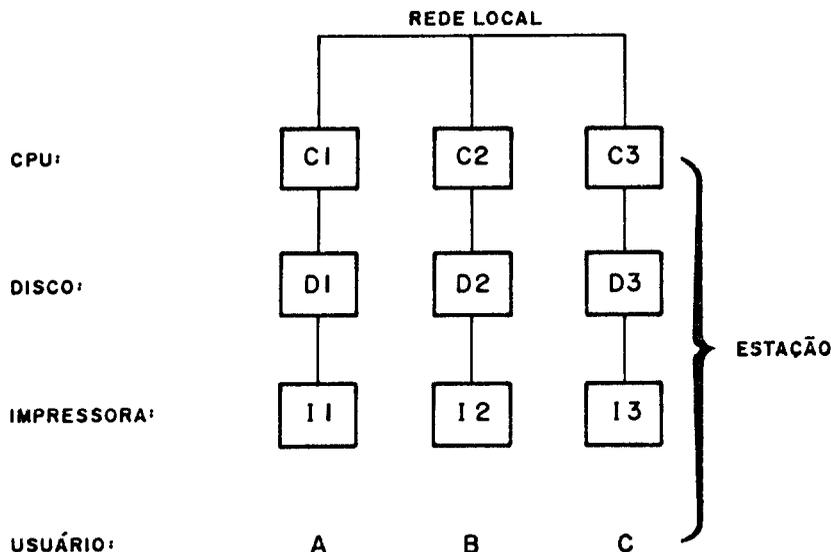


FIGURA 7.1 Exemplo de utilização da rede local

O sistema operacional MUMPS poderia oferecer várias opções de comunicação que serão exemplificadas a seguir.

A Comunicação direta sob controle do programador

Cada estação seria identificada por um número único. No exemplo que está sendo considerado os números das estações poderiam ser 1, 2 e 3.

Para que a estação 2 recebesse uma mensagem enviada pela estação 1, bastaria a estação 2 executar os comandos:

Use 1 Read MENSAGEM

A estação 1 enviaria a mensagem através dos seguintes comandos:

## Use 2 Write MENSAGEM

O limite de cada mensagem seria de 255 bytes.

Para transmitir várias mensagens, bastaria executar um programa de transmissão como o seguinte, que transmite 20 mensagens diferentes para a estação 2:

```
Use 2
```

```
For I=1:1:20 Write VETOR(I)
```

B Comunicação por compartilhamento de dados

Considere o seguinte comando MUMPS:

```
Set ↑ARQUIVO(CHAVE)=VALOR
```

Normalmente este comando cria um novo elemento (registro) no arquivo de nome ARQUIVO, sob chave de acesso CHAVE e com valor VALOR.

Para recuperar o valor, basta referenciar o nome como no exemplo abaixo:

```
Write ↑ARQUIVO(CHAVE)+1
```

Neste exemplo está sendo somando 1 ao conteúdo do registro com chave de acesso CHAVE, no arquivo de nome ARQUIVO, e exibindo (Write) o resultado na tela.

Esta independência da localização física do arquivo, característica do MUMPS, poderia ser aproveitada na rede local da maneira ilustrada a seguir.

O programa MUMPS a seguir cria 100 registros em um arquivo indexado pelas chaves 1, 2, 3, ..., 100, tendo cada registro o mesmo conteúdo que sua chave:

```
For I = 1:1:100 Set ↑ARQUIVO(I)=I
```

Este arquivo será gravado na unidade de disco local onde foram executados os comandos acima, por exemplo a estação 1.

O programador poderia especificar, no entanto, que deseja fazer acesso à unidade de disco D2 (isto é feito através do comando ZUse D2) e a partir daí toda e qualquer referência a arquivos se dirigiria à estação na qual se encontra a unidade de disco D2.

Por exemplo, para criar o arquivo anterior no disco D2 (que pertence à estação 2), bastaria executar a seguinte sequência de comandos:

```
Use D2 For I=1:1:100 Set ↑ARQUIVO(I)=I
```

Em caso de erro, o sistema operacional abortaria a tarefa em execução emitindo a mensagem de erro apropriada. Caso o programador assim o desejasse, também poderia interceptar o erro por programa e dar-lhe o tratamento que julgasse mais conveniente.

### C Comunicação por compartilhamento de programas

O usuário poderia de forma extremamente simplificada trocar de programas entre as diversas estações. Bastaria indicar qual a unidade de disco a utilizar (o sistema operacional deve conhecer a distribuição das unidades de disco pelas estações) e utilizar os comandos ZLOAD (carrega programa de disco para memória) e/ou ZFILE (salva programa em memória para disco). Por exemplo, supor que o usuário na estação 1 tenha acabado de digitar um programa que deseja armazenar no disco D2 da estação 2. Para isto seria necessário (e suficiente) executar os comandos:

```
Use D2 ZFILE <nome do programa>
```

### D Comunicação por transferência de controle

A quarta e última forma de comunicação entre estações S permitiria que um usuário em uma determinada estação enviase um programa (com seus respectivos dados) para ser executado em outra estação. O comando utilizado seria o comando JOB <nome do programa>, no qual deveria ser indicado, ainda, a estação na qual o programa deveria executar.

Desta maneira o usuário do sistema operacional poderia acionar outras tarefas que executariam assincronamente com tarefa principal (por exemplo, emissão de um relatório em paralelo com a entrada de dados).

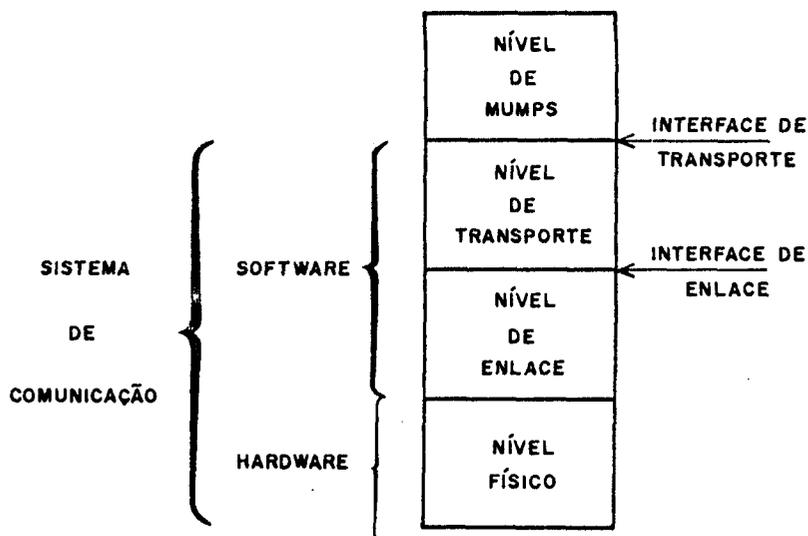


Figura 7.2 - Situação do Sistema de Comunicação.

## 8 CONCLUSÕES

Para o desenvolvimento do trabalho aqui descrito, estudou-se, inicialmente, o modelo de referência para interconexão de sistemas OSI/ISO, visando determinar quais seriam as necessidades para a construção de um software de comunicação para redes locais. Definiu-se uma rede local em barra com 2 níveis de software, que implementariam as funções básicas para a troca de informações.

A partir da definição do escopo do trabalho, passou-se à especificação dos níveis de software a serem implementados.

O software foi escrito em linguagem Assembly do microprocessador Z80. O código resultante ocupa cerca de 3K bytes de memória.

Para que o sistema pudesse ser testado foi desenvolvido um monitor, o qual realiza as chamadas das primitivas implementadas.

Os testes realizados permitiram concluir que os objetivos do trabalho foram alcançados.

No desenvolvimento do trabalho foi possível visualizar extensões que poderiam melhorar o desempenho do sistema.

A inexistência de interrupção para sinalizar a chegada de dados no microcomputador, faz com que seja necessário estar continuamente testando a interface de comunicação. A implementação de um mecanismo de interrupção retiraria do software a tarefa de verificar o estado da interface serial de comunicação.

Para tornar o sistema de comunicação um sistema de uso prático seria importante incorporá-lo a um sistema operacional existente, ou desenvolver o restante dos níveis de protocolo definidos pela ISO.

Para facilitar o transporte do software de comunicação para diferentes microcomputadores, seria necessário escrever, para cada micro, apenas o nível de enlace (dependente de hardware).

ANEXO 1

LISTAGEM DO SOFTWARE DE COMUNICAÇÃO



0069'	20 20 44 45			
006D'	20			
006E'	43 4F 4E 45	DB	'CONEXAO PARA UMA ESTACAO REMOTA'	
0072'	50 41 4F 20			
0076'	50 41 52 41			
007A'	20 55 4D 41			
007E'	20 45 53 54			
0082'	41 43 41 4F			
0086'	20 52 45 4D			
008A'	4F 54 41			
008D'	0A 0D 20 20	DB	LF,CR,' PERGUNTA: VERIFICA A CHEGADA'	
0091'	20 20 20 20			
0095'	20 20 50 45			
0099'	52 47 55 4E			
009D'	54 41 3A 20			
00A1'	56 45 52 49			
00A5'	46 49 43 41			
00A9'	20 41 20 43			
00AD'	48 45 47 41			
00B1'	44 41			
00B3'	20 44 45 20	DB	' DE MENSAGENS,DIFUSAO OU CONEXAO'	
00B7'	4D 45 4E 53			
00EB'	41 47 45 4E			
00EF'	53 2C 44 49			
00C3'	46 55 53 41			
00C7'	4F 20 4F 55			
00CB'	20 43 4F 4E			
00CF'	45 58 41 4F			
00D3'	0A 0D 09 20	DB	LF,CR,' ENVIA : MANDA MENSAGENS PA'	
00D7'	20 20 20 20			
00DB'	20 20 45 4E			
00DF'	56 49 41 20			
00E3'	20 20 3A 20			
00E7'	4D 41 4E 44			
00EB'	41 20 4D 45			
00EF'	4E 53 41 47			
00F3'	45 4E 53 20			
00F7'	50 41			
00F9'	52 41 20 45	DB	'RA ESTACAO REMOTA CONECTADA'	
00FD'	53 54 41 43			
0101'	41 4F 20 52			
0105'	45 4D 4F 54			
0109'	41 20 43 4F			
010D'	4E 45 43 54			
0111'	41 44 41			
0114'	0A 0D 20 20	DB	LF,CR,' RECEBE : ESPERA CHEGADA DE'	

011B' 20 20 20 20  
 011C' 20 20 52 45  
 0120' 43 45 42 45  
 0124' 20 20 3A 20  
 0128' 45 53 50 45  
 012C' 52 41 20 43  
 0130' 48 45 47 41  
 0134' 44 41 20 20  
 0138' 44 45  
 013A' 20 4D 45 4E  
 013E' 53 41 47 45  
 0142' 4D 20 44 41  
 0146' 20 45 53 54  
 014A' 41 43 41 4F  
 014E' 20 43 4F 4E  
 0152' 45 43 54 41  
 0156' 44 41  
 0158' 0A 0D 20 20  
 015C' 20 20 20 20  
 0160' 20 20 44 45  
 0164' 53 43 4F 4E  
 0168' 45 43 54 41  
 016C' 3A 44 45 53  
 0170' 46 41 5A 20  
 0174' 43 4F 4E 45  
 0178' 58 41 4F 20  
 017C' 43 4F 4D 20  
 0180' 45 53 54 41  
 0184' 43 41 4F 20  
 0188' 52 45 4D 4F  
 018C' 54 41 20 43  
 0190' 4F 4E 45 43  
 0194' 54 41 44 41  
 0198' 0A 0D 20 20  
 019C' 20 20 20 20  
 01A0' 20 20 44 4F  
 01A4' 45 55 4E 44  
 01A8' 45 20 3A 20  
 01AC' 45 4E 56 49  
 01B0' 41 20 4D 45  
 01B4' 4E 53 41 47  
 01B8' 45 4D 20 50  
 01BC' 41 52 41  
 01BF' 20 54 4F 44  
 01C3' 41 53 20 45  
 01C7' 53 54 41 43

DB ' MENSAGEN DA ESTACAO CONECTADA '

DB LF,CR,' DESCONECTA:DESFAZ CONEXAO '

DB 'COM ESTACAO REMOTA CONECTADA'

DB LF,CR,' DIFUNDE : ENVIA MENSAGEN PARA '

DB ' TODAS ESTACOES DA REDE '

```

01CB' 4F 45 53 20
01CF' 44 41 20 52
01D3' 45 44 45
01D6' 0A 0A 0A 0D
01DA' 2A 2A 2A 2A
01DE' 2A 2A 2A 2A
01E2' 2A 2A 2A 2A
01E6' 2A 2A 2A 2A
01EA' 2A 2A 2A 2A
01EE' 2A 2A 2A 2A
01F2' 2A 2A 2A 2A
01F6' 2A 2A 2A
01F9' 2A 2A 2A 2A
01FD' 2A 2A 2A 2A
0201' 2A 2A 2A 2A
0205' 2A 2A 2A 2A
0209' 2A 2A 2A 2A
020D' 2A 2A 2A 2A
0211' 2A 2A 2A 2A
0215' 2A 2A 2A 2A
0217' 2A 2A 2A 2A
021D' 0E 09
021F' 11 022B'
0222' CD 0005
0225' C3 024E'
0228' 0A 0D 4E 2E
022C' 20 44 41 20
0230' 45 53 54 41
0234' 43 41 4F 20
0238' 44 45 20 54
023C' 52 41 42 41
0240' 4C 46 4F 20
0244' 28 30 20 41
0248' 20 37 29 3A
024C' 3D 24
024E' 0E 01
0250' CD 0005
0253' 32 0C5E'
0256' CD 00DD'
0259' 0E 09
025B' 11 0264'
025E' CD 0005
0261' C3 031E'
0264' 0A 0A 0A 0A
0268' 0D 20 4D 4F
026C' 4E 49 54 4F
    
```

DB LF,LF,LF,CF, '\*\*\*\*\*'

DB '\*\*\*\*\*'

```

MNT00: LD C,OUTPUT
        LD DE,CBC2
        CALL CPH
        JP MNT01
CBC2: DB LF,CF,'M. DA ESTACAO DE TRABALHO (0 A 9):=6'
    
```

```

MNT01: LD C,INPUT
        CALL CPH
        LD (ESTACAO),A
        CALL ABERTURA
MNT03: LD C,OUTPUT
        LD DE,CBC3
        CALL CPH
        JP MNT02
CBC3: DB LF,LF,LF,LF,CF,'MONITOR - TRANSPORTE '
    
```

0270'	52 20 2D 20		
0274'	54 52 41 4E		
0278'	53 50 4F 52		
027C'	54 45 20		
027F'	0A 20 43 4F	DB	LF, ' COMANDOS = ',CR
0283'	4D 41 4E 44		
0287'	4F 53 20 3D		
028B'	20 0D		
028D'	0A 20 20 20	DB	LF, ' C : CONECTA ',CR
0291'	20 43 20 3A		
0295'	20 43 4F 4E		
0299'	45 43 54 41		
029D'	20 0D		
029F'	0A 09 50 20	DB	LF, ' P : PERGUNTA ',CR
02A3'	3A 20 50 45		
02A7'	52 47 55 4E		
02AB'	54 41 20 0D		
02AF'	0A 09 45 20	DB	LF, ' E : ENVIA ',CR
02B3'	3A 20 45 4E		
02B7'	56 49 41 20		
02BB'	0D		
02BC'	0A 09 52 20	DB	LF, ' R : RECEBE ',CR
02C0'	3A 20 52 45		
02C4'	43 45 42 45		
02C8'	20 0D		
02CA'	0A 09 44 20	DB	LF, ' D : DESCONECTA ',CR
02CE'	3A 20 44 45		
02D2'	53 43 4F 4E		
02D6'	45 43 54 41		
02DA'	20 0D		
02DC'	0A 09 42 20	DB	LF, ' B : DIFUNDE ',CR
02E0'	3A 20 44 49		
02E4'	46 55 4E 44		
02E8'	45 20 0D		
02EB'	0A 09 49 20	DB	LF, ' I : INICIALIZA ',CR
02EF'	3A 20 49 4E		
02F3'	49 43 49 41		
02F7'	4C 49 5A 41		
02FB'	20 0D		
02FD'	0A 4F 55 54	DB	LF, 'OUTRO P/CPM ',CR
0301'	52 4F 20 50		
0305'	2F 43 50 4D		
0309'	20 0D		
030B'	0A 0A 0D 20	DB	LF,LF,CR, ' COMANDO := \$'
030F'	20 20 43 4F		
0313'	4D 41 4E 44		

```

0317' 4F 20 3A 3D
0318' 20 20 24
031E' 0E 01          MNT02: LD C, INPUT
0320' CD 0005       CALL CPM
0323' FE 43        CP 'C'
0325' CA 0349'     JP Z,C00
0328' FE 50        CP 'P'
032A' CA 03F0'     JP Z,P00
032D' FE 45        CP 'E'
032F' CA 0453'     JP Z,E00
0332' FE 52        CP 'R'
0334' CA 04F3'     JP Z,R00
0337' FE 44        CP 'D'
0339' CA 0532'     JP Z,D00
033C' FE 42        CP 'B'
033E' CA 05B1'     JP Z,B00
0341' FE 49        CP 'I'
0343' CA 0000'     JP Z,MONITOR
0346' C3 0000     JP 0
0349' 0E 09          C00: LD C, OUTPUT ;CONEXAO
034B' 11 0354'     LD DE,CBC4
034E' CD 0005       CALL CPM
0351' C3 0373'     JP C01
0354' 0A 00 4E 2E   CBC4: DB LF,CR,'N.DA ESTACAO REMOTA(0 A 9)::-5'
0358' 44 41 20 45
035C' 53 54 41 43
0360' 41 4F 20 52
0364' 45 4D 4F 54
0368' 41 28 30 20
036C' 41 20 39 29
0370' 3A 3D 24
0373' 0E 01          C01: LD C, INFUT
0375' CD 0005       CALL CPM
0378' 47           LD B,A
0379' 3A 0C5E'     LD A,(ESTACAO)
037C' B8           CP B
037D' 20 06       JR NZ,C04
037F' 11 03D9'     LD DE,CBC17
0382' C3 0397'     JP C03
0385' 3E 00          C04: LD A,0
0387' CD 0812'     CALL TRANSPORTE
038A' FE 00        CP 0
038C' 20 06       JR NZ,C02
038E' 11 039F'     LD DE,CBC5
0391' C3 0397'     JP C03
0394' 11 03BA'     C02: LD DE,CBC6

```

0397'	0E 09	C03:	LD	C,OUTPUT
0399'	CD 0005		CALL	CPM
039C'	C3 0259'		JP	MNT03
039F'	0A 0D 3E 3E	CBC5:	DB	LF,CR,'>> CONEXAO ESTABELECID A %'
03A3'	20 43 4F 4E			
03A7'	45 50 41 4F			
03AB'	20 45 53 54			
03AF'	41 42 45 4C			
03B3'	45 43 49 44			
03B7'	41 20 24			
03BA'	0A 0D 3E 3E	CBC6:	DB	LF,CR,'>> CONEXAO NAO ESTABELECID A %'
03BE'	20 43 4F 4E			
03C2'	45 50 41 4F			
03C6'	20 4E 41 4F			
03CA'	20 45 53 54			
03CE'	41 42 45 4C			
03D2'	45 43 49 44			
03D6'	41 20 24			
03D9'	0A 0D 3E 3E	CBC17:	DB	LF,CR,'>> ORIGEM = DESTINO %'
03DD'	20 4F 52 49			
03E1'	47 45 4D 20			
03E5'	3D 20 44 45			
03E9'	53 54 49 4E			
03ED'	4F 20 24			
03F0'	3E 03	P00:	LD	A,3 ;PERGUNTA
03F2'	CD 0812'		CALL	TRANSPORTE
03F5'	FE 00		CP	0
03F7'	28 08		JR	Z,P01
03F9'	FE 02		CP	2
03FB'	28 0A		JR	Z,P02
03FD'	FE 01		CP	1
03FF'	28 22		JR	Z,P03
0401'	11 040D'	P01:	LD	DE,CBC7 ;SEM RECEBIMENTO
0404'	C3 0397'		JP	C03
0407'	11 039F'	P02:	LD	DE,CBC5 ;CONEXAO
040A'	C3 0397'		JP	C03
040D'	0A 0D 3E 3E	CBC7:	DB	LF,CR,'>> SEM RECEBIMENTO %'
0411'	20 53 45 4D			
0415'	20 52 45 43			
0419'	45 42 49 4D			
041D'	45 4E 54 4F			
0421'	20 24			
0423'	09	P03:	ADD	HL,BC ;DIFUSAO
0424'	36 24		LD	(HL),'%'
0426'	11 0431'		LD	DE,CBC8
0429'	0E 09		LD	C,OUTPUT

```

042B' CD 0005          CALL   CPM
042E' C3 044D'        JF     F04
0431' 0A 0D 3E 3E    CBCB:  DB     LF,CR,'>> RECEBIMENTO DIFUSAO '
0435' 20 52 45 43
0439' 45 42 49 4D
043D' 45 4E 54 4F
0441' 20 44 49 46
0445' 55 53 41 4F
0449' 20
044A' 0A 0D 24          DB     LF,CR,'&'
044D' 11 0DF3'        F04:  LD     DE,FACDAT+6
0450' C3 0397'        JF     C03
0453' CD 059C'        E00:  CALL   LER           ;ENVIA
0455' 21 0E0A'        LD     HL,BUFFER
0459' 3E 06          LD     A,6
045B' CD 0812'        CALL   TRANSPORTE
045E' FE 00          CP     0
0460' 20 28          JR     NZ,E01
0462' 11 046D'        LD     DE,CBC9
0465' 0E 09          E02:  LD     C,OUTPUT
0469' CD 0605          CALL   CPM
046A' C3 0257'        JF     INT03
046D' 0A 0D 3E 3E    CBC9:  DB     LF,CR,'>> SUCESSO NA TRANSMISSAO $'
0471' 20 53 55 43
0475' 45 53 53 4F
0479' 20 4E 41 20
047D' 54 52 41 4E
0481' 53 4D 49 53
0485' 53 41 4F 20
0489' 24
048A' FE 01          E01:  CP     1
048C' 20 06          JR     NZ,E03
048E' 11 0464'        LD     DE,CBC10
0491' C3 0465'        JF     E02
0494' FE 03          E03:  CP     3
0496' 28 06          JR     Z,E04
0498' 11 04C3'        LD     DE,CBC15
049B' C3 0465'        JF     E02
049E' 11 04D3'        E04:  LD     DE,CBC16
04A1' C3 0465'        JF     E02
04A4' 0A 0D 3E 3E    CBC10: DB     LF,CR,'>> INSUCESSO NA TRANSMISSAO $'
04A8' 20 49 4E 53
04AC' 55 43 45 53
04B0' 53 4F 20 4E
04B4' 41 20 54 52
04B8' 41 4E 53 4D

```

```

0400' 40 51 51 11
0406' 4F 20 24
0403' 0A 0D 3E 3E          CBC15:      DB      LF,CR,'>> S/CONEXAO 1'
0407' 20 53 2F 43
0408' 4F 4E 45 58
040F' 41 4F 20 24
04D3' 0A 0D 3E 3E          CBC15:      DB      LF,CR,'>> ESTACAO REMOTA E/RESPOSTA 1'
04D7' 20 45 53 54
04DB' 41 43 41 4F
04DF' 20 52 45 4D
04E3' 4F 54 41 20
04E7' 53 2F 52 45
04EB' 53 50 4F 53
04EF' 54 41 20 24
04F3' 21 0608'          R00:      LD      HL,BUFFER          ;FECEBE
04F6' 3E 07              LD      A,9
04F8' CD 0812'          CALL    TRANSPORTE
04FB' FE 00              CF      0
04FD' 28 0A              JR      Z,R01
04FF' FE 02              CF      2
0501' 28 0E              JR      Z,R02
0503' 11 040D'          LD      DE,CBC7
0506' C3 0465'          JF      E02
0507' 36 24              R01:      LD      (HL),'B'
0508' 11 0608'          LD      DE,BUFFER-2
050E' C3 0465'          JF      E02
0511' 11 0517'          R02:      LD      DE,CBC11
0514' C3 0465'          JF      E02
0517'                  CBC11:
0517' 0A 0D 3E 3E          DB      LF,CR,'>> PEDIDO DE DESCONEXAO 1'
051B' 20 50 45 44
051F' 49 44 4F 20
0523' 44 45 20 44
0527' 45 53 43 4F
052B' 4E 43 5B 41
052F' 4F 20 24
0532' 3E 0C          D00:      LD      A,12          ;PEDIDO DE DESCONEXAO
0534' CD 0812'          CALL    TRANSPORTE
0537' FE 00              CF      0
0539' 20 06              JR      NZ,D01
053B' 11 0547'          LD      DE,CBC12
053E' C3 0465'          JF      E02
0541' 11 0562'          D01:      LD      DE,CBC13
0544' C3 0465'          JF      E02
0547' 0A 0D 3E 3E          CBC12:      DB      LF,CR,'>> DESCONEXAO REALIZADA 1'
054B' 20 44 45 53

```

```

054F' 43 4F 4E 45
0553' 58 41 4F 20
0557' 52 45 41 40
055B' 49 5A 41 44
055F' 41 20 24
0562' 0A 0D 3E 3E      CBC13:      DB      LF,CR,' )) DESCONEXAO NAO REALIZADA $'
0566' 20 44 45 53
056A' 43 4F 4E 45
056E' 58 41 4F 20
0572' 4E 41 4F 20
0576' 52 45 41 40
057A' 49 5A 41 44
057E' 41 20 24
0581' CD 057C'      B00:      CALL      LEF          ;DIFUSAO
0584' 3E 0F          LD          A,15
0588' 21 0406'      LD          HL,BUFFER
0589' CD 0812'      CALL      TRANSPORTE
058C' FE 00          CF          0
058E' 20 06          JR          NZ,B01
0590' 11 046D'      LD          DE,CBC9
0593' C3 0485'      JF          E02
0596' 11 04A4'      B01:      LD          DE,CBC10
0599' C3 0485'      JF          E02
;
;
; ROTINAS AUXILIARES
;
;
059C' 0E 09      LER:      LD          C,OUTPUT
059E' 11 05A7'      LD          DE,CBC14
05A1' CD 0005      CALL      CPM
05A4' C3 05EC'      JF          LER01
05A7' 0A 0D 20 45      CBC14:      DB          LF,CR,' ENTRE COM MENSAGEN : '
05AB' 4E 54 52 45
05AF' 20 43 4F 40
05B3' 20 4D 45 4E
05B7' 53 41 47 45
05BB' 4D 29 3A 20
05BF' 0A 0D 20 44      DB          LF,CR,' DIFUSAO - 256 CHAR'
05C3' 49 43 55 53
05C7' 41 4F 20 20
05CB' 20 32 35 36
05CF' 20 43 48 41
05D3' 52
05D4' 0A 0D 20 45      DB          LF,CR,' ENVIA - 512 CHAR'
05D8' 4E 53 49 41

```

```

05DC' 20 20 20 20
05EE' 20 35 31 32
05E4' 20 43 43 41
05E8' 52
05E9' 0A 0D 24
05EC' 21 060A'
05EF' 11 0000
05F2' 0E 01
05F4' E5
05F5' D5
05F6' CD 0005
05F8' D1
05FA' E1
05FB' FE 00
05FD' 26 06
05FF' 13
0600' 77
0601' 23
0602' C3 05F2'
0605' D5
0606' C1
0607' C9

```

```

LER01: DB LF,CR,'$'
LD HL,BUFFER
LD DE,0
LER02: LD C,INPUT
PUSH HL
PUSH DE
CALL CPA
POP DE
POP HL
CF CR
JR Z,LER55
INC DE
LD (HL),A
INC HL
JP LER02
LER99: PUSH DE
POP BC
RET

```

```

;
;
;
; AREA DE DADOS DO MONITOR
;
;

```

```

0608' 0A DB LF
0609' 0D DB CR
060A' BUFFER: DS 520

```

\*\*\*\*\*

```

;
;
;
; N I V E L D E T R A N S P O R T E
;
;
; C O N S T A N T E S D O T R A N S P O R T E
;

```

```

0000 LIVRE EQU 0 ;ESTADO LIVRE
0001 CNTDTX EQU 1 ; " CONECTADO TRANSMISSAO
0002 CNTDRX EQU 2 ; " CONECTADO RECEPCAO
0003 MAXCNX EQU 03
0003 MAXREC EQU 03

```

```

0000          MOV          EQU          00          ;LIM. INICIALIZAO
0008          MARET          EQU          00
000C          JUMF          EQU          000H          ;CODIGO DO JUMF
;
;
; PONTO DE ENTRADA DO TRANSPORTE
;
;
0812'   E5          TRANSPORTE:   PUSH          HL          ;GUARDA PARAMETROS
0813'   C5          PUSH          BC
0814'   21 081C'   LD          HL,TFENTRY
0817'   4F          LD          C,A
0818'   06 00          LD          B,0
081A'   07          ADD          HL,BC
081B'   E9          JP          (HL)
;
; TABELA DE ENTRADAS DO TRANSPORTE
;
081C'   C3          TFENTRY:      DB          JUMF
081D'   082E'   DW          CONECTA
081F'   C3          DB          JUMF
0820'   0895'   DW          PERGUNTA
0822'   C3          DB          JUMF
0823'   09BD'   DW          ENVA
0825'   C3          DB          JUMF
0826'   0AAA'   DW          RECEBE
0828'   C3          DB          JUMF
0829'   0924'   DW          DESCONECTA
082B'   C3          DB          JUMF
082C'   0972'   DW          DIFUNDE
;
;*****
;
;          FRUITIVA CONECTA
;          -----
;          CHAMADA FEITA POR "CALL TRANSPORTE", COM VALOR 00 NO REGISTRA-
;          DOR A. O REGISTRADOR B DEVE CONTER O ENDEREÇO DA ESTACAO COM QUEM SE
;          PROCESSARA A CONEÇÃO.
;          COMO PARAMETRO DE SAIDA O REGISTRADOR A INDICA O RESULTADO
;          DA OPERACAO :
;          0 - SUCESSO      1 - INSUCESSO      2 - NAO ESTA LIVRE
;*****
;
;
082E'   C1          CONECTA:      POP          BC

```

```

002F' E1                                POP    HL
0030' 3A 003F'                          LD     A,(ESTADO) ;VERIFICA ESTADO DA ESTACAO
0033' FE 00                              CP     LIVRE
0035' C2 0000'                          JP     NZ,RETOR1
0038' 3E C0                                CHX00: LD     A,000H ;CABECALHO
003A' 11 0001                          LD     DE,1
003D' CD 00F0'                          CALL  CABEC
0040' CD 003B'                          CALL  INICIO
0043' CD 001A'                          CHX01: CALL  TRANSF
0044' 3E 00                              LD     A,0
0045' CD 0077'                          CALL  ENLACE ;CHAMADA DO ENLACE TRANSFERAO
;ESPERANDO TRANSFERAO
0048' FE FF                              CP     NAK
004C' 20 0F                              JR     NZ,CHX02
004F' 3A 0050'                          CHX03: LD     A,(CHTINT)
0052' FE 03                              CP     NACKCH
0054' CA 0092'                          JP     Z,RETOR2
0057' 3C                                    INC    A
0058' 32 0050'                          LD     (CHTINT),A
005B' C3 0043'                          JP     CHX01
005E' CD 0053'                          CHX02: CALL  DELAY
0061' 3A 0054'                          LD     A,(TXAREA)
0064' 5F                                    LD     E,A
0065' 3E 03                              LE     A,3
0067' CD 0079'                          CALL  ENLACE ;CHAMADA DO ENLACE RECEPCAO
;ESPERANDO RECEPCAO ---- CONEXAO
006A' FE FF                              CP     NAK
006C' 28 E1                              JR     Z,CHX03
006E' 3A 00F3'                          LD     A,(FACDAT+4)
0071' FE C0                              CP     000H ;NEG CONFIRM ?
0073' 20 DA                              JR     NZ,CHX03 ;NAO
0075' 3E 01                              LD     A,CHTDTX ;SIM
0077' 32 005F'                          LD     (ESTADO),A
007A' 3E 00                              LD     A,00 ;INICIALIZA CONTROLE
007C' 32 0061'                          LD     (TXCTRL),A
007F' 32 0062'                          LD     (RXCTRL),A
0082' 3A 00F0'                          LD     A,(FACDAT+1)
0085' 32 0063'                          LD     (ESTREIT),A ;ESTACAO SENDA CONECTADA
0088' 3E 00                              LD     A,0 ;SUCESSO
008A' C3 009A'                          JP     CHX05
008D' 3E 02                                RETOR1: LD     A,2 ;ESTACAO JA CONECTADA
008F' C3 0094'                          JP     CHX05
0092' 3E 01                                RETOR2: LD     A,1 ;INSUCESSO
0094' C9                                CHX05: RET ;RETORNO DA ROTINA CONECTA
;DO NIVEL DE TRANSPORTE

```

```

;
;*****
;
; PRIMITIVA PERGUNTA
;
; CHAMADA FEITA POR "CALL TRANSPORTE", COM VALOR 03 NO
;REGISTRADOR A. QUANDO ALGUMA MENSAGEM E' RECEBIDA O ENDEREÇO DA
;ESTACAO TRANSMISSORA E' GUARDADO EM ESTTX, SENDO QUE OS DADOS DA
;MENSAGEM,DIFUSAO, FICAM LOCALIZADOS A PARTIR DA POSICAO FACDAT+6
; E O TAMANHO DA MESSA NO PAR DE REGISTRADORES BC. O REGISTRADOR
;A INFORMA O RESULTADO DA OPERACAO:
; 0 - SEM RECEBIMENTO 1 - DIFUSAO 2 - CONEXAO
;
;*****

```

0895'	C1	PERGUNTA:	FOP	BC	
0896'	E1		FOP	HL	
0897'	CD 0C3B'		CALL	INICIO	
089A'	1E FF	PGT00:	LD	E,0FFH	
089C'	3E 03		LD	A,3	
089E'	CD 0C79'		CALL	ENLACE	;CHAMA RE PARA RECEPCAO
		;ESPERANDO RECEPCAO	-----	PERGUNTA	
08A1'	FE FF		CF	NAK	
08A3'	2B 0C		JR	Z,PGT01	
08A5'	3A 0DF3'		LD	A,(FACDAT+4)	
08A8'	FE C0		CF	0C0H	
08AA'	DA 080B'		JF	Z,PGT0X	
08AD'	FE DF		CF	0DFH	
08AF'	2B 0E		JR	Z,PGTDF3	
08B1'	3A 0C3B'	PGT01:	LD	A,(RECTHT)	
08B4'	FE 03		CF	NAAREC	
08B6'	2B 1B		JR	Z,PGT0E	
08B8'	3C		INC	A	
08B9'	32 0C3B'		LD	(RECTHT),A	
08BC'	C3 089A'		JF	PGT00	
08BF'	3A 0DF0'	PGTDF3:	LD	A,(FACDAT+1)	;MENSAGEM DE DIFUSAO
08C2'	32 0C3C'		LD	(ESTTX),A	
08C5'	ED 4B 0DF1'		LD	BC,(BTETX)	
08C9'	08		DEC	BC	
08CA'	0B		DEC	BC	
08CB'	21 0DF5'		LD	HL,FACDAT+6	
08CE'	3E 01		LD	A,1	
08D0'	C3 0723'		JF	PGT99	
08D3'	3E 06	PGT0E:	LD	A,0	;SEM RECEBIMENTO
08D5'	C3 0723'		JF	PGT99	
08D8'	3A 0C5F'	PGT0X:	LD	A,(ESTACO)	;PEDIDO DE CONEXAO



```

0922' 3A 0C5F' LD A,(ESTADO)
0927' FE 01 CP CNTDTZ ;VERIFICA ESTADO
0928' 28 07 JR Z,DCT00
092D' FE 02 CP CNTDRZ
092F' 29 03 JR Z,DCT00
0931' C3 093A' JF RETD01
0934' 3A 0C33' DCT00: LD A,(ESTRELA) ;PEDIDO DE DESCONEXAO
0937' 47 LD B,A
0938' 3E DC LD A,00CH
093A' 11 0001 LD DE,I
093D' CD 0BF0' CALL CABEC
0940' 3E 00 LD A,0
0942' 32 0C30' LD (CNTTNT),A
0945' CD 0C1A' DCT01: CALL TRANSF
0948' 3E 00 LD A,0
094A' CD 0C79' CALL ENLACE ;CHAMA RE PARA TRANSMISSAO
;ESPERANDO TRANSMISSAO ----- DESCONECTA
094D' FE FF CP NAK
094F' 20 0F JR NZ,DCT03
0951' 3A 0C30' DCT04: LD A,(CNTTNT)
0954' FE 03 CP NACONH
0956' CA 092F' JF Z,RETDE2
0959' 3C INC A
095A' 32 0C30' LD (CNTTNT),A
095D' C3 0945' JF DCT01
0960' DCT03: LD A,LIVRE ;DESDONEXAO REALIZADA
0962' 3E 0C5F' LD (ESTADO),A
0965' 3E 00 LD A,0
0967' C3 0971' JF DCT79
096A' 3E 02 RETD01: LD A,2 ;SEM CONEXAO
096C' C3 0971' JF DCT79
096F' 3E 01 RETD02: LD A,1 ;INSUCESSO
0971' C9 DCT79: RET ;RETORNO DA ROTINA DESCONECTA
; DO NIVEL DE TRANSPORTE.
;
;*****
;
; PRIMITIVA DIFUNDE
;
; CHAMADA FEITA POR "CALL TRANSPORTE", COM VALOR 15 NO
;REGISTRADOR A. O PAR HL DEVE CONTER O ENDEREÇO DA AREA DE DADOS DA
;MENSAGEM, SENDO QUE O TAMANHO DESTA E' INDICADO PELO PAR BC.
; O REGISTRADOR A RETORNA COM O RESULTADO DA OPERAÇÃO:
; 0 - SUCESSO 1 - INSUCESSO
;*****

```

```

;
;
0972' 01          DIFUNDE:  POF  BC
0973' 01          POF  HL
0974' 3E 0F      LD    A,0FFH
0975' ED 43 0C6E' LD    (SIZE),BC
0976' ED 5B 0C6E' LD    DE,(SIZE)
097E' 13         INC   DE
097F' 13         INC   DE
0980' 03 FF      LD    B,0FFH
0981' 0D 0BF0'   CALL  CABEC
0983' 3E 00      LD    A,0
0987' 32 0C6E'   LD    (TAREA+5),A
098A' 22 0D70'   LD    (INIBS),HL
098D' 11 0DF5'   LD    DE,FACTAT+5
0990' ED 4B 0C6E' LD    BC,(SIZE)
0994' ED B0      LDIR                    ;TRANSFERE DADOS DE NS PARA NE
0995' 3E 00      LD    A,0
0998' 32 0C6D'   LD    (TRSTNT),A
099B' 0D 0C1A'   DFD00: CALL  TRANSF
099E' 3E 00      LD    A,0
09A0' 0D 0D79'   CALL  ENLACE                    ;CHAMA NE PARA TRANSMISSAO
;ESPERANDO TRANSMISSAO ----- DIFUNDE
09A3' FE FF      CP    NAK                    ;FOI TRANSMITIDO?
09A5' 20 0E      JR    NZ,RETDF1        ;SIM
09A7' 3A 0C6D'   LD    A,(TRSTNT)                ;NAO
09AA' FE 0F      CP    MAXTRES
09AC' 2B 0C      JR    Z,RETDF2
09AE' 3C         INC   A
09AF' 32 0C6D'   LD    (TRSTNT),A
09B2' 03 099B'   JF    DFD00
09B5' 3E 00      RETDF1: LD    A,0                    ;TRANSMISSAO OK - SUCESSO
09B7' 03 09BC'   JF    DFD99
09BA' 3E 01      RETDF2: LD    A,1                    ;INSUCESSO
09BC' 0F          DFD99: RET                    ;RETORNO DA ROTINA DIFUNDE
;GO NIVEL DE TRANSPORTE.
;
;*****
;
; PRIMITIVA  ENVIA
;
;          CHAMADA FEITA POR "CALL TRANSPORTE", COM VALOR 06
;NO REGISTRADOR A. COMO PARAMETROS DE ENTRADA O SISTEMA RECEBE :
; PAR HL - ENDEREÇO DO INICIO DA MENSAGEM DO NS
; PAR BC - TAMANHO DA MENSAGEM
;
;          O REGISTRADOR A RETORNA COM O RESULTADO DA OPERAÇÃO:

```

; 0 - SUCESSO 1 - INSUCESSO 2 - SEM CONEXAO

;  
;\*\*\*\*\*

```

09BD' C1          ENVIAR:      POP      BC
09BE' E1          POP      HL
09BF' 3A 0C5F'    LD        A,(ESTADO) ;VERIFICA ESTADO
09C0' FE 00      CF        LIVRE
09C1' CA 0A90'    JF        Z,RETEVI
09C2' FE 01      CF        CNTDTX
09C3' 28 05      JR        Z,ENV01
09C4' 3E 01      LD        A,CNTDTX
09C5' 32 0C5F'    LD        (ESTADO),A
09C6' 3A 0C53'    ENVI01:   LD        A,(ESTREN) ;PREPARA CABECALHO
09C7' C5          PUSH     BC
09C8' 47          LD        B,A
09C9' 3A 0C11'    LD        A,(TXCTL)
09CA' 11 0000     LD        DE,0
09CB' CD 0BF0'    CALL    CABEC
09CC' C1          POP      BC
09CD' CD 0C5E'    ENVI02:   CALL    INICIO
09CE' 22 0C71'    LD        (NSGADDR),HL ;GUARDA AFONTAÇÕES NS
09CF' ED 43 0C75' LD        (NSGSETZ),BC

```

; TESTA N. DE BYTES A TRANSMITIR

; 255 TRANSMITE 255

; (<= 255 TRANSMITE BC

```

09E9' E5          PUSH     HL
09EA' 37          SCF
09EB' 3F          CCF
09EC' 11 00FF     LD        DE,255
09ED' C5          PUSH     BC
09EE' E1          POP      HL
09EF' ED 52      SRC      HL,DE
09F0' 28 11      JR        Z,ENV03
09F1' FA 0A05'    JF        N,ENV03
09F2' ED 53 0C75' LD        (TRFH0),DE ; ) 255
09F3' 3E 01      LD        A,1
09F4' 32 0C69'    LD        (TAREA+5),A
09F5' E5          PUSH     HL
09F6' C1          POP      BC
09F7' C3 0A1E'    JF        ENV04
09F8' ED 43 0C75' ENVI03:   LD        (TRFH0),BC ; (<= 255
09F9' 01 0000     LD        BC,0
09FA' 3E 00      LD        A,0

```

```

0A0F' 32 0C87' LD (TAREA+3),A
0A12' E1 ENV04: FOF HL
0A13' C5 FUSH BC
0A14' ED 58 0C75' LD DE,(TRFMSG)
0A18' 13 INC DE
0A19' 13 INC DE
0A1A' 13 INC DE
0A1B' ED 53 0C66' LD (TAREA+2),DE
0A1F' ENV08: CALL TRANSF
0A22' CD 0C1A' LD BC,(TRFMSG)
0A26' 11 0DF5' LD DE,FACDAT+6
0A29' ED 80 LDIR ;TRANSEFE DADOS DE NS PARA NE
0A2B' CD 0C02' CALL CHECKSUM
0A2E' 3E 00 ENV05: LD A,0
0A30' CD 0C77' CALL ENLACE ;CHAMA NE PARA TRANSMISSAO
;REFERANDO TRANSMISSAO ----- ENVA
0A33' FE FF CF NAK ;TRANSMITIDO?
0A35' 20 0F JR NZ,ENV06 ;SIM
0A37' 3A 0C3E' LD A,(TRETNT) ;NAO
0A3A' FE 0F CF NAKTRE
0A3C' CA 0A7E' JF Z,FETEVA
0A3F' 3C INC A
0A40' 32 0C2D' LD (TRETNT),A
0A43' C3 0A3E' JF ENV05
0A46' 3A 0C63' ENV06: LD A,(ETRET) ;TRANSMISSAO EFETUADA
0A47' 5F LD E,A
0A4A' 3E 03 LD A,3
0A4C' CD 0C79' CALL ENLACE ;CHAMA NE PARA RECEPCAO
0A4F' FE FF CF NAK ;OK?
0A51' 20 12 JR NZ,ENV07 ;SIM
0A53' 3A 0C6A' ENVRET: LD A,(RETTNT)
0A56' FE 03 CF NAKRET
0A5B' CA 0A7E' JF Z,RETEVA
0A5B' 3C INC A
0A5C' 32 0C6A' LD (RETTNT),A
0A5F' 2A 0C71' LD HL,(MSGDEF)
0A62' C3 0A1F' JF ENV06
0A65' 3A 0DF3' ENV07: LD A,(FACDAT+4) ;RECEPCAO REALIZADA
0A68' E3 A0 AND 0A0H
0A6A' FE A0 CF 0A0H ;E' CONFIRMACAO?
0A6C' 28 03 JR Z,ENV09 ;SIM
0A6E' C3 0A53' JF ENVRET
0A71' 3A 0DF3' ENV09: LD A,(FACDAT+4) ;MSG E' DE CONFIRMACAO
0A74' E3 0F AND 0FH
0A76' 47 LD B,A

```



```

0AB4' 31 005F'          LD      (ESTADO),A
0AB7' 01 0000          LD      BC,0
0ABA' ED 43 0073'          LD      (REGSIZE),BC
0ABE' CD 003B'          CALL    INICIO          ;INICIALIZA CONTADORES
0AC1' 3A 00E3'          LD      A,(ESTREM)
0AC4' 5F                LD      E,A
0AC5' 22 0071'          LD      (REGADDR),HL
0AC8' 3E 03             LD      A,3
0ACA' CD 0079'          CALL    ENLACE          ;CHAMA NE PARA RECEPCAO
0ACD' FE FF             CP      NAK             ;NEO RECEBIDA?
0ACF' 20 0F             JR      NZ,REC02        ;SIM
0AD1' 3A 00E8'          LD      A,(RECTHT)      ;NAO
0AD4' FE 03             CP      NAKREC
0AD6' CA 00CB'          JF      Z,RETREC
0AD9' 3C                INC     A
0ADA' 32 00E8'          LD      (RECTHT),A
0ADD' C3 0AC1'          JF      REC00
0AE0' 3A 00F3'          LD      A,(FACDAT+4)    ;NEO RECEBIDA
0AE3' 47                LD      B,A
0AE4' 3A 00E2'          LD      A,(RECTFL)
0AE7' 8B                CP      B               ;E' ESPERADA?
0AE8' C2 0B8C'          JF      NZ,REC10        ;NAO
0AEB' ED 4B 00F1'          LD      BC,(FACDAT+2)   ;SIM
0AEF' 00                DEC     BC
0AF0' 00                DEC     BC
0AF1' ED 43 0075'          LD      (TRFHSZ),BC     ;TAMANHO DA NEG
0AF5' CD 0042'          CALL    CHECKSUM
0AF8' FE 00             CP      0               ;ERRO DE TRANSMISSAO?
0AFA' C2 0B8D'          JF      NZ,REC09        ;SIM
0AFD' 3A 00E5'          LD      A,(ESTREM)      ;NAO
0B00' 47                LD      B,A             ;PREPARA CONFIGURACAO
0B01' 3A 00F4'          LD      A,(FACDAT+5)
0B04' 32 0077'          LD      (CONTIN),A
0B07' 3A 00F3'          LD      A,(FACDAT+4)
0B0A' F3 A0             OR      0A0H
0B0C' 11 0001          LD      DE,1
0B0F' CD 00F0'          CALL    CABEC
0B12' CD 001A'          CALL    TRNEF          ;REC03:
0B15' 3E 00             LD      A,0
0B17' CD 0079'          CALL    ENLACE          ;CHAMA NE PARA TRANSMISSAO
0B1A' FE FF             CP      NAK             ;TRANSMITIDO?
0B1C' 20 0F             JR      NZ,REC04        ;SIM
0B1E' 3A 00AD'          LD      A,(RECTHT)      ;NAO
0B21' FE 0F             CP      NAKTRES
0B23' CA 00CB'          JF      Z,RETREC
0B26' 3C                INC     A

```

```

0027' 32 0020' LD (TESTNT),A
002A' C3 00D1' JF RECO0
002D' 3A 0062' REC04: LD A,(RCTRL) ;MSG TRANSMITIDA
0030' EE 01 XOR 01H
0032' 32 0062' LD (RCTRL),A
0035' E5 PUSH HL
0038' 2A 0073' LD HL,(MSGSIZE)
0039' ED 4B 0075' LD BC,(TRFREQ)
003D' 09 ADD HL,BC ;ATUALIZA TAMANHO DA MSG
003E' 22 0073' LD (MSGSIZE),HL
0041' E1 POP HL
0042' E5 PUSH HL
0043' D1 POP DE
0044' 21 00F5' LD HL,FACDAT+6
0047' 0B DEC BC
0048' ED 09 LDIR ;TRANSFERE MSG PARA MS
004A' D5 PUSH DE
004B' E1 POP HL
004C' 3A 0077' LD A,(CONTIN)
004F' FE 01 CF 01 ;MSG CONTINUA?
0051' CA 00BE' JF Z,RECO6 ;SIM

0054' ED 4B 0073' LD BC,(MSGSIZE) ;NAO
0058' 3E 00 LD A,0
005A' C3 00C0' JF RECO9
005D' 3A 0063' REC09: LD A,(ESTRELA) ;PEDIDO DE RETRANSMISSAO
0059' 47 LD B,A
0061' 11 0001 LD DE,1
0064' 3A 0062' LD A,(RCTRL)
0067' EE 01 XOR 01H
0069' F8 A0 OR 0A0H
006B' CD 00F0' CALL CABEC
006E' CD 001A' REC05: CALL TRANSF
0071' 3E 00 LD A,0
0073' CD 0079' CALL ENLACE ;CHAMA ME PARA TRANSMISSAO
0076' FE FF CF 00 ;TRANSMITIDO?
0078' 20 0F JR NZ,RECO6 ;EIH
007A' 3A 006D' LD A,(TESTNT) ;NAO
007D' FE 0F CF 00
007F' CA 00CB' JF Z,RETC02
0082' 5C INC A
0083' 32 006D' LD (TESTNT),A
0086' C3 00C1' JF RECO0
0089' C3 00BE' REC03: JF RECO0
008C' 3A 00F3' REC10: LD A,(FACDAT+4)
008F' FE DC CF 00 ;MSG DEBOLHEXAO?

```

```

0B91' 00 00 JR Z,RETRO3 ;SIN
0B93' FE 00 CP 000H ;REQ CONEXAO?
0B95' CA 0BA0' JF Z,REC13 ;SIN
0B98' E6 D0 AND 000H ;NAO
0B9A' CA 0B5D' JF Z,REC09 ;REQ DADO: SEM SEQUENCIA
0B9D' C3 0AC1' JF REC00 ;REQ NAO INTERESSA
0BA0' 3A 0C83' RETRO3: LD A,(ESTREN) ;PEDIDO DE CONEXAO
0BA3' 47 LD B,A
0BA4' 11 0001 LD DE,1
0BA7' 3E CC LD A,0CCH ;CONFIRMACAO
0BA9' CD 0BFB' CALL CABEC
0BAC' CD 0C1A' RETRO4: CALL TRANSF
0BAF' 3E 00 LD A,0
0BB1' CD 0C75' CALL ENLACE ;CHAMA NE PARA TRANSMISSAO
0BB4' FE FF CP NAK ;TRANSMITIDO?
0BB6' 20 22 JR NZ,RETRO4 ;SIN
0BB8' 3A 0C8D' LD A,(TRSTHT) ;NAO
0BBE' FE 0F CP NAKTS
0BBD' 28 0C INC A
0BBF' 3C LD (TRSTHT),A
0BC0' 32 0C8D' JF REC00
0BC3' C3 0AC1' RETRO1: LD A,3 ;SEM CONEXAO
0BC8' C3 0BDC' JF REC99
0BCB' 3E 01 RETRO2: LD A,1 ;INSUCESSO
0BCD' C3 0BDC' JF REC99
0BD0' 3E 00 RETRO3: LD A,LIVRE ;PEDIDO DE DESCONEXAO
0BD2' 32 0C5F' LD (ESTADO),A ;ESTADO LIVRE
0BD5' 3E 00 LD A,2
0BD7' C3 0BDC' JF REC99
0BD9' 3E 04 RETRO4: LD A,4 ;PEDIDO DE CONEXAO
0BDC' C7 RETRO9: RET ;RETORNO DA ROTINA FEDEBE
;DO NIVEL DE TRANSPORTE.

```

;

\*\*\*\*\*

; ROTINA AUXILIARES

; ROTINA DE INICIALIZACAO DO TRANSPORTE

```

0BDD' 3E 00 ABERTURA: LD A,LIVRE
0BDF' 32 0C5F' LD (ESTADO),A
0BE2' 3E 01 RETCON: LD A,CONEX
0BE4' 32 E0AE LD (CONID),A

```

```

0BE7' 3E 9E          LD    A,INTAR:
0BE9' 30 E0AF       LD    (CONCON),A
0BEC' 3A E0AD       LD    A,(CONETS)
0BEF' 09           RET

;
; ROTINA DE FORMATAÇÃO DO CABEÇALHO
;
0BF0' 32 0C68'     CABEC:  LD    (TXAREA+4),A
0BF3' 7B          LD    A,B
0BF4' 32 0C64'     LD    (TXAREA),A
0BF7' 3A 0C5E'     LD    A,(ESTACAO)
0BFA' 32 0C65'     LD    (TXAREA+1),A
0BFD' ED 53 0C66'  LD    (TXAREA+2),DE
0C01' 09           RET

;
; ROTINA DE CÁLCULO DE CHECKSUM DOS DADOS
;
0C02' E5          CHECKSUM: PUSH  HL
0C03' 21 0DF5'     LD    HL,PACDAT+6
0C06' ED 5B 0C05'  LD    DE,(TRFHSG)
0C0A' 0E 00       LD    C,0
0C0C' 7E          CHK1:  LD    A,(HL)
0C0D' AF          XOR    C
0C0E' 4F          LD    C,A
0C0F' 1B          DEC    DE
0C10' 23          INC    HL
0C11' 7A          LD    A,D
0C13' B3          OR    E
0C13' 20 F7       JR    NZ,CHK1
0C15' 79          LD    A,C
0C16' 2F          CPL
0C17' 77          LD    (HL),A
0C18' E1          POP  HL
0C19' 09           RET

;
; ROTINA DE TRANSFERÊNCIA DO CABEÇALHO DA KERNIGEN, DA ÁREA DE
; CONTROLE DE TRANSPORTE PARA O NÍVEL DE ENLACE.
;
0C1A' 3A 0C64'     TRNEF:  LD    A,(TXAREA)
0C1D' 32 0BEF'     LD    (PACDAT),A
0C20' 3A 0C65'     LD    A,(TXAREA+1)
0C23' 32 0BF0'     LD    (PACDAT+1),A
0C26' ED 5B 0C66'  LD    DE,(TXAREA+2)
0C2A' ED 53 0BF1'  LD    (BYTETX),DE
0C2E' 3A 0C68'     LD    A,(TXAREA+4)
0C31' 32 0BF3'     LD    (PACDAT+4),A

```

```

0034' 3A 006A' LD A,(TXAREA+5)
0037' 3E 00F4' LD (FACDAT+5),A
003A' C9 RET

;
; ROTINA DE INICIALIZAÇÃO DAS VARIÁVEIS DE CONTROLE TX-RX
;
003B' 3E 00 INICIO: LD A,0
003D' 32 006D' LD (TRSTNT),A
0040' 3E 006E' LD (RECTNT),A
0043' 32 0077' LD (CONTIN),A
0046' 32 0069' LD (CRSTNT),A
0049' 32 006A' LD (RETTNT),A
004C' 3A E0AC LD A,(COMDAT)
004F' 3A E0AD LD A,(COMBYS)
0052' C9 RET

;
; ROTINA DELAY
;
;
0053' C5 DELAY: PUSH BC
0054' 01 3FF0 LD BC,3FF0H
0057' 0B DEL00: DEC BC
005B' 78 LD A,B
005F' 01 OR C
005A' 20 FB JR NZ,DEL00
005C' 01 POP BC
005D' C9 RET

;
;*****
;
;
; VARIÁVEIS DO NÍVEL DE TRANSPORTE
;
005E' ESTACAO: DS 01 ;ESTACAO ORIGEM
005F' ESTADO: DS 01 ;ESTADO DO NT
0060' CRSTNT: DS 01 ;CONTADOR DE CONEXAO
0061' TXCTRL: DS 01 ;MENSAGEN A TRANSMITIR
0062' RXCTRL: DS 01 ;MENSAGEN A RECEBER
0063' ESTREN: DS 01 ;ESTACAO CONECTADA
0064' TXAREA: DS 06 ;CABECALHO DE CONTROLE
;DA MENSAGEM
;DEST,ORIG,BYTES,TIPO,CONT.

006A' RETTNT: DS 01
006B' RECTNT: DS 01 ;CONTADOR RECEPCAO
006C' ESTTX: DS 01 ;ESTACAO ORIGEM DIFUSAO
006D' TRSTNT: DS 01 ;CONTADOR TRANSMISSAO

```

```

006E'      SIZE:      DS      02      ;TAMANHO DA MENSAGEM
0070'      INIHEE:    DS      01      ;INICIO DA MENSAGEM
0071'      INSHADR:   DS      02      ;ENDEREÇO
0073'      NEGHEE:    DS      02      ;TAMANHO
0075'      TEPHEG:    DS      02
0077'      CONTIN:    DS      01
0078'      LASTNO:    DS      01
;
;*****
;
;
;      N I V E L      D E      E N L A C E
;
;
;      ROTINAS DO NIVEL DE ENLACE
;
;      CONSTANTES USADAS PELAS ROTINAS
;
;
00C3      CSH          EQU      00C3H
00D6      DSH          EQU      00D6H
E0AF      CONCOH      EQU      0E0AFH ;DADOS DA INTERFACE SERIAL
E0AE      CONCHO      EQU      0E0AEH ;STATUS DA INTERFACE SERIAL
E0AC      COADAT      EQU      0E0ACH
E0AD      CONSTE      EQU      0E0ADH
00F0      NRCOLS      EQU      00F0H ;NÚMERO DE COLUNAS
00FF      NTVL        EQU      00FFH ;VERIFICAÇÃO DA LINHA
0010      YATENT      EQU      0010H ;TENTATIVA DE TRANSMISSÃO
0003      RXTENT      EQU      0003
0002      ACK         EQU      0002H ;RESPOSTA POSITIVA
00FF      NAK         EQU      00FFH ;NEGATIVA
009E      INTARX      EQU      009EH
0081      CHCF:       EQU      0081H
0080      CACTX       EQU      0080H
00FF      RXVEZES     EQU      00FFH ;RECEPCAO DE 1 BYTE
00FF      TXVEZES     EQU      00FFH ;TRANSMISSAO DE 1 BYTE
0001      SOH         EQU      0001H ;START OF HEADER
;
;
;
;      PONTO DE ENTRADA DO
;      NIVEL DE ENLACE
;
;
0079'      ES          EQU      ENLACE:   PUSH   HL      ; SALVA REGISTRADORES
007A'      CS          EQU      PUSH   BC

```

```

0076' 01 0003' LD HL,TABENTRY
007E' 4F LD C,A
007F' 08 00 LD B,0
0081' 09 ADD HL,BC
0082' E7 JF (HL)
;
; TABELA DE ENTRADA DAS ROTINAS DO
; NIVEL DE ENLACE
;
0083' 03 TABENTRY: DB 03H
0084' 0089' DW TRANSMISSAO
0085' 03 DB 03H
0087' 0D1E' DW REDEFCAO
;
; ROTINA DE TRANSMISSAO DE MENSAGENS DO
; NIVEL DE ENLACE
;
0089' 3E 00 TRANSMISSAO: LD A,0
008B' 3E 0DEB' LD (NOBLE),A
008E' 3E 0DEC' LD (NTTX),A
0091' DD 01 0DEE' LD IX,START
;
; VERIFICA O ESTADO DA LINHA DE TRANSMISSAO
;
0095' 0E FF INIC: LD C,NTVL
0097' 3A E0AD ULINHA: LD A,(CONSTE)
009A' 0B 2F BIT 5,A
009C' 20 14 JR NZ,TXCON
009E' 0D 0BEE' RETRAT: CALL RETCON
00A1' 0D DEC C
00A2' 20 F3 JR NZ,ULINHA
00A4' 3A 0DEC' LD A,(NTTX)
00A7' FE 10 CF TXENT
00A9' 20 6D JR Z,FIMNAK
00AB' 3C INC A
00AD' 3E 0DEC' LD (NTTX),A
00AF' 03 0095' JF INIC
;
; TRANSMITE CON --- LINHA LIVRE
;
00B2' 3E B9 TXCON: LD A,CHDTX
00B4' 3E E0AE LD (CONCMD),A
00B7' 0D 00B8' CALL TXCON
00BA' FE FF CF NAK
00BC' 0A 009E' JF Z,RETRAT
00BF' 0D 00D2' CALL RXCON

```

```

0CC2' FE FF          CF      NAK
0CC4' 28 52          JR      Z,FINNAK
0CC5' 7A             LD      A,D
0CC7' FE 01          CF      SOH
0CC9' 28 1A          JR      Z,TXCTFI

```

```

;
; TRATAMENTO DE COLISAO
;

```

```

0CCB' CD 0BE2'      COLISAO:  CALL   RETCON
0CCE' 3A 0DEB'      LD      A,(NDOLB)
0CD1' FE F0          CF      NACOLS
0CD3' 28 43          JR      Z,FINNAK
0CD5' 3C             INC     A
0CD6' 32 0DEB'      LD      (NDOLB),A
0CD7' 57             LD      D,A
0CDA' ED 5F          CLEIN:  LD      A,R
0CDC' 3D             DECFH:  DEC     A
0CDE' 20 F0          JR      NZ,DECFH
0CDF' 15             DEC     D
0CE0' 20 F8          JR      NZ,CLEIN
0CE2' C3 0C9E'      JF      RETRAT

```

```

;
; TRANSMISSAO DO CONTROLE DA MENSAGEM
;

```

```

0CE5' 01 0004       TXCTR1:  LD      BC,4
0CE8' DD 23         LOOFTX:  INC     IX
0CEA' CD 00B8'      CALL   TXCON
0CED' FE FF          CF      NAK
0CEF' CA 0C9E'      JP      Z,RETRAT
0CF2' 0B             DEC     BC
0CF3' 78             LD      A,B
0CF4' B1             OR      C
0CF5' 20 F1          JR      NZ,LOOFTX

```

```

;
; TRANSMISSAO DOS DADOS DA MENSAGEM
;

```

```

0CF7' ED 4E 0CF1'  TXMENS:  LD      BC,(BYTETX)
0CF8' DD 23         TXLOOP:  INC     IX
0CFD' CD 00B8'      CALL   TXCON
0D00' FE FF          CF      NAK
0D01' CA 0C9E'      JP      Z,RETRAT
0D05' 0B             DEC     BC
0D06' 78             LD      A,B
0D07' B1             OR      C
0D08' 20 F1          JR      NZ,TXLOOP
0D0A' 3A E0AC       FIHTX:  LD      A,(CONDAT)

```

```

0000' 3A E0AD          LD      A,(CONST)
0010' 3E 01          LD      A,CNDXK
;                  LD      (CONCND),A
;                  CALL    RXCON
;                  CP      NAK
;                  JR      Z,FINNAK
;                  LD      A,D
;                  CP      NAK
;                  JR      Z,FINNAK
0012' 3E 06          LD      A,ACK
0014' 47            LD      B,A
0015' C3 00B1'       JP      FIRENLACE
;
; FIN NEGATIVO
;
0018' 3E FF          FINNAK: LD      A,NAK
001A' 47            LD      B,A
001B' C3 00B1'       JP      FIRENLACE
;
; ROTINA DE RECEPCAO DE MENSAGENS
; DO NIVEL DE ENLACE
;
;
001E' D0 01 0DEF'    RECEPCAO: LD      IX,FACD1
0022' 3E 06          LD      A,0
0024' 32 0DED'       LD      (NTRX),A
;
; VERIFICA LINHA DE TRANSMISSAO
;
;
0027' 3A E0AD          LIFREE: LD      A,(CONST)
002A' C8 0F          BIT      5,A
002C' 28 1A          JR      Z,RCS1
002E' 0E FF          INICI:  LD      C,INTVL
;
; RECEPCAO DO INICIO DE MENSAGEM (SOH)
;
0030' C0 0DD2'       RXCON:  CALL    RXCON
0033' FE FF          CP      NAK
0035' 20 14          JR      NZ,FACD1
0037' 00            RYTRAN: DEC     C
0038' 20 F6          JR      NZ,RXCON
003A' 3A 0DED'       LD      A,(NTRX)
003D' FE 03          CP      EXTENT
003F' 28 07          JR      Z,FINNAK
0041' 3C            INC     A
0042' 32 0DED'       LD      (NTRX),A

```

```

0045' 03 00EE'                JF      INICI
;
; RECEPCAO DO CABECALHO DA MENSAGEM (CONTROLE)
0048' 00 00D2'      RECEB1:    CALL    RXCOM
;
0048' 7A      RXCTR1:    LD      A,D
004C' FE 01      CF      SOH
004E' 02 0D37'      JF      NZ,RXTRAN
0051' 01 0004      LD      BC,4
0054' 00 00D2'      RECEB1:    CALL    RXCOM
0057' FE FF      CF      NAK
0059' 0A 0D37'      JF      Z,RXTRAN
005C' DD 7E 00      LD      (IX+0),D
005F' DD 23      INC     IX
0061' 0E      DEC     BC
0062' 7B      LD      A,B
0063' B1      OR      C
0064' 20 EE      JR      NZ,RECEB1
;
; RECEPCAO DOS DADOS DA MENSAGEM
;
0066' ED 40 00F1'      RXDATA:    LD      BC,(BYTETA)
006A' 00 00D2'      RXLOOP:    CALL    RXCOM
006D' FE FF      CF      NAK
006F' 0A 0D37'      JF      Z,RXTRAN
0072' DD 7E 00      LD      (IX+0),D
0075' DD 23      INC     IX
0077' 0B      DEC     BC
0078' 7B      LD      A,B
0079' B1      OR      C
007A' 20 EE      JR      NZ,RXLOOP
;
; VERIFICA ENDEREÇO DESTINO , E MODO DE SER UMA RECEPCAO
; ESPECIFICA O ENDEPEÇO ORIGEM.
;
007C' 7B      LD      A,E
007D' FE FF      CF      0FFH
007F' 20 11      JR      NZ,RECEB
0081' 3A 00EF'      LD      A,(FACDAT)
0084' FE FF      CF      0FFH
0086' 20 1A      JR      Z,FIMRX
0088' 47      LD      B,A
0089' 3A 00CE'      LD      A,(ESTACAO)
009C' 88      CF      0
009D' 28 13      JR      Z,FIMRX
009F' 03 0D37'      JF      RXTRAN

```

```

0092' 48          RECEBF:  LD    B,C
0093' 3A 00F8'    LD    A,(FACDAT+1)
0094' 88          CF    B
0097' 28 07      JR    Z,FINRX
0099' C3 0037'    JF    RETRAN
009C' 3E FF      ENLACE:  LD    A,NAK
009E' 47          LD    B,A
009F' C3 00B1'    JF    FIMEMLACE
00A2' 3A E0AC    FINRX:  LD    A,(CONDAT)
00A5' 3A E0AD    LD    A,(CONSTB)
00A8' D0 01 00EA' LD    IX,ACRTO
00AC' 3E 8F      LD    A,CHDTX
;              LD    (CONCHD),A
;              CALL  TXCON
;              CF    NAK
;              JR    Z,FINAK
00AE' 3E 06      LD    A,ACK
00B0' 47          LD    B,A
;
; RETORNO DO NIVEL DE ENLACE
;
00B1' C0 00E2'    FIMEMLACE: CALL  RETCON
00B4' 78          LD    A,B
00B5' C1          POP  BC
00B6' E1          POP  HL
00B7' C7          RET
;
; ROTINAS AUXILIARES DO NIVEL DE ENLACE
;
;
; ROTINA DE TRANSMISSAO DE 1 BYTE
;
;
00B8' 16 FF      TXCON:  LD    D,TAVEZES
00BA' 3A E0AD    TX1:  LD    A,(CONETE)
00BD' C8 07      BIT  4,A
00BF' 28 08      JR    NZ,DATA
00C1' 15          DEC  D
00C2' 28 FA      JR    NZ,TX1
00C4' 3E FF      LD    A,NAK
00C6' C3 00D1'    JF    VOLTA
00C9' D0 7E 00    DATA: LD    A,(IX+0)
00CC' 3E E0AC    LD    (CONDAT),A
00CF' 3E 06      LD    A,ACK
00D1' C7          VOLTA: RET
;

```

```

; ROTINA DE RECEPCAO DE 1 BYTE
;
;
0DD2' 16 FF      RXCON:      LD      D,RXVEZES
0DD4' 3A E0AD    RX1:        LD      A,(COMETS)
0DD7' 0B 5F      BIT      3,A
0DD8' 20 08      JR      NZ,DATRX
0DDB' 15        DEC      D
0DDC' 20 F8      JR      NZ,RX1
0DDE' 3E FF      LD      A,DAK
0DE0' 03 0DEF'   JF      RETORNO
0DE3' 3A E0AD    DATRX:     LD      A,(CONDAT)
0DE5' 57        LD      D,A
0DE7' 3E 06      LD      A,ACK
0DE8' 07        RETORNO:    RET
;
;
; DEFINICAO DA AREA DE DADOS DO NIVEL DE ENLACE
;
;
0DEA' 06        ACNHO:     DB      ACK      ;RESPOSTA ACK
0DEB' 06        NCOLS:     DB      0        ;NUMERO DE COLISOES
0DEC' 06        NTRY:      DB      0        ;TENTATIVAS DE TRANSMISSAO
0DED' 06        NTRX:      DB      0        ;TENTATIVA DE RECEPCAO
0DEE' 01        START:     DB      SCH      ;INICIO DE MENSAGEM
0DEF' 00        FACDAT:    DB      0        ;ENDEREÇO DESTINO
0DF0' 00        PADAT:     DB      0        ;ENDEREÇO ORIGEM
0DF1' 0000      BYTETX:    DW      0        ;N.DE BYTES TRANSMITIR
0DF3' 00        DB:        DB      0        ;tipo da mensagem
0DF4' 00        DB:        DB      0        ;CONTROLE
0DF5'          DB:        DB      256     ;DADOS
END

```

Macros:

Symbols:

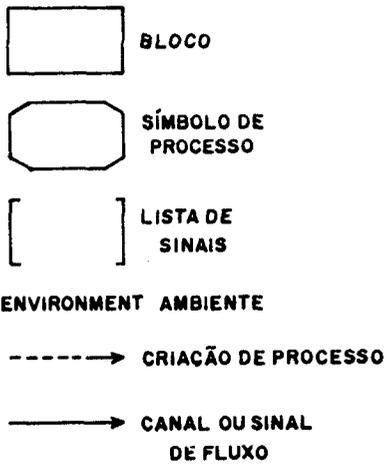
ABERTU	0BDD'	ACK	0205	ACKNO	0DEA'	B00	0581'
B01	0576'	BUFFER	040A'	BYTETX	0DF1'	C00	0349'
C01	0373'	C02	0354'	C03	0397'	C04	0385'
C0H	06C3	CABEC	0BF0'	CBC1	0906'	CBC10	0444'
CBC11	0517'	CBC12	0547'	CBC13	0552'	CBC14	05A7'
CBC15	04C3'	CBC16	04D3'	CBC17	03D9'	CBC2	0328'
CBC3	0264'	CBC4	0354'	CBC5	039F'	CBC6	03BA'
CBC7	040D'	CBC8	0431'	CBC9	045D'	CHK1	0C0C'
CHECK5	0C02'	CLSN	0CDA'	CNDRX	0901	CNDTX	0089'
CNTDEX	0602	CNTDTX	0901	CNX00	0830'	CNX01	0843'
CNX02	085E'	CNX03	084F'	CNX09	0854'	CNTINT	0C10'
COLISA	0C0B'	CONCHD	E9AE	CONCON	E9AF	CONDAT	E9AC
CONETS	E9AD	CONELT	082E'	CONTIN	0C77'	CFH	0005'
CR	000D	D00	0532'	D01	0541'	D0H	0000'
DATRX	0DE3'	DATA	0DC7'	DCT00	0934'	DCT01	0745'
DCT03	0750'	DCT04	0751'	DCT99	0971'	DECRH	0C0C'
DEL00	0C57'	DELAY	0C53'	DEED00	0924'	DFD00	0956'
DFD99	09BC'	DFUND	0772'	E00	0453'	E01	048A'
E02	0465'	E03	0474'	E04	049E'	ENLACE	0C75'
ENV01	09D0'	ENV02	09DF'	ENV03	0A06'	ENV04	0A12'
ENV05	0A2E'	ENV06	0A46'	ENV07	0A55'	ENV08	0A1F'
ENV09	0A71'	ENV10	0A80'	ENV99	0AAB'	ENVIA	09BD'
ENVRET	0A53'	ESTACH	0C5E'	ESTAD0	0C5F'	ESTREN	0C83'
ESTTX	0C3C'	FINENL	00B1'	FINNAK	0018'	FINRX	00A2'
FINTX	006A'	INIC	0C95'	INICI	0CCE'	INIC10	0C3B'
INRES3	0C70'	INFUT	0001	INTARX	009E	JUMP	00C3'
LASTH3	0C78'	LER	059C'	LER01	05EC'	LER02	05F2'
LER79	0625'	LF	006A	LIFREE	0027'	LIVRE	0000'
LOOPTX	0CE6'	MAXCNA	0003	MAXPBT	0003	MAXREC	0003'
MAXRET	0008	MAXTRS	000F	MNT00	021D'	MNT01	024E'
MNT02	031E'	MNT03	025F'	MNHITO	0000'	MSGADD	0C71'
MSGSZ	0C73'	NAK	00FF	NCOLS	0DEB'	NCOLS	00F0'
NTRX	0CE0'	NITX	0DEC'	NTVL	00FF	OUTPUT	000F'
P00	03F0'	P01	0401'	P02	0407'	P03	0423'
P04	044D'	FACDAT	0DEF'	PERCUN	087D'	PBT00	087A'
PBT01	08B1'	PBT02	08D3'	PBT03	08F1'	PBT04	090E'
PBT99	0923'	PBTCHK	08D8'	PBTDF0	08BF'	R00	04FC'
R01	0509'	R02	0511'	RC01	0D4B'	REC00	0A01'
REC01	0AB7'	REC02	0AED'	REC03	0B12'	REC04	0B2D'
REC05	0B3E'	REC06	0B89'	REC07	0B5D'	REC10	0B8C'
REC13	0BA0'	REC14	0BAC'	REC20	0ABE'	REC79	0BDC'
RECEB1	0D54'	RECEBE	0AA4'	RECEFC	0D1E'	RECEEF	0D92'
RECTNT	0C1E'	RETCX1	088D'	RETCX2	0892'	RETD01	09B5'
RETD02	09BA'	RETD01	076A'	RETD02	096F'	RETEV1	0A90'
RETEV2	0A95'	RETEV3	0AA1'	RETEV4	0A9B'	RETURN	0DE7'
RETRAT	0C9E'	RETRC1	0BC5'	RETRC2	0BCB'	RETRC3	0BD0'

RETRCA	0BDA'	RETTNT	0C6A'	RETCOH	06E2'	RXI	0DD4'
RXCCH	0DD2'	RXCTRI	0D4B'	RXCTRL	0C62'	RXDATA	0D66'
RXLOOP	0D6A'	RXNAK	0D9C'	RXSOH	0D30'	RXTENT	0993
RXTRAN	0D37'	RXVEZE	09FF	SIZE	0C6E'	SOH	0001
START	0DEE'	TABENT	0C80'	TPTENT	081C'	TRANSF	0C1A'
TRANSH	0C87'	TRANSP	0812'	TRFNSG	0C75'	TRSTNT	0C3D'
TXI	0DBA'	TXAKEA	0C34'	TXCOH	0DB8'	TXCTRI	0CE5'
TXCTRL	0C61'	TXLOOP	0CFB'	TXRENS	0CF7'	TXSOH	0CB2'
XTENT	0010	TXVEZE	09FF	VLINHA	0C77'	VOLTA	0DD1'

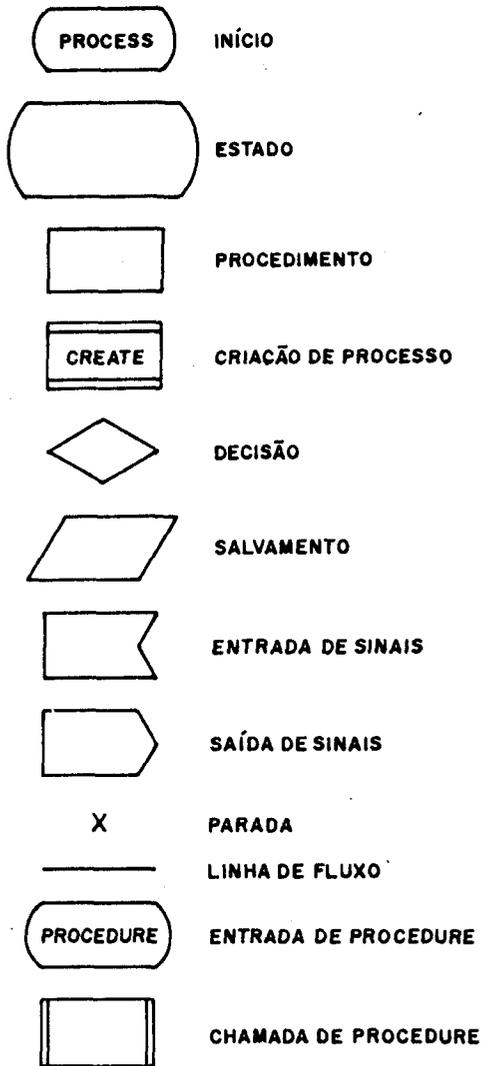
No Fatal error(s)

APÊNDICE 1

SÍMBOLOS DA LINGUAGEM DE DESCRIÇÃO  
E ESPECIFICAÇÃO (SDL)



Símbolos de diagrama de interação .



Símbolos de diagramas de processos .

## BIBLIOGRAFIA

- /ALL 81/ ALLAN, Roger. Local-net architecture, protocol issues heating up. Electronic Design, New Jersey, 29(8):91-102, Apr. 1981.
- /ASE 85/ ASENSIO PEREZ, Victor Ramon. Redes Locais: níveis "0" e "1/2". Boletim SCOPUS, São Paulo, v.8, n.85. jul. 1985.
- /CHA 83/ CHAZAL, Pierre E. de & DICKSON, Gary I. Status of CCITT description techniques and application to protocol specification. Proceedings of the IEEE, New York, 71(12):1346-55, Dec. 1983.
- /CLE 81/ CLEMENTS, Ken & DAUGHERTY, Dane. Ultra-low-cost network for personal computers. Byte, Peterborough, 6(10):50-66, Oct. 1981.
- /COL 82/ COLLINSON, R.P.A. The Cambridge ring and UNIX. Software-Practice & Experience, Sussex, 12(6):583-94, June 1982.
- /DAV 80/ DAVIDSON, John M. et alii. Local network gives new flexibility to distributed processing. Electronics, New York, 53(21):114-22, Sept. 1980.
- /DEL 82/ DELLAR, Carl N. R. A file server for a network of low cost personal microcomputers. Software-Practice & Experience, Sussex, 12(11):1051-68, Nov. 1982.
- /FER 84/ FERNANDES, Lúcio F. de Sã. Protocolo de transporte para redes de locais. São Paulo, IME, 1984. (Dissertação de Mestrado).
- /GIE 78/ GIEN, Michel. A file transfer protocol (FTP). Computer Networks, Amsterdam, 2(1):312-9, Feb. 1978.
- /GRA 82/ GRAUBE, Maris. IEEE-802: a standards update. EDN, Boston, 27(4):119-22, Feb. 1982.
- /IEE 83/ IEEE. Project 802, local area network standards; draft IEEE 802.3. New York, July 1983.
- /IEE 83a/ IEEE. Project 802, local area network standards; draft IEEE 802.4. New York, July 1983.
- /IEE 84/ IEEE. Project 802, local area network standards; draft IEEE 802.5. New York, Feb. 1984.

- /ISO 81/ ISO/TC97/SC16 data processing-open system inter-connection-basic reference model. Computer Networks, Amsterdam, 5(2):81-118, Apr. 1981.
- /MCL 81/ McLEOD, Jonah. Local-net terminals take on new looks for new function. Electronic Design, New Jersey, 29(8):133-40, Apr. 1981.
- /MET 76/ METCALFE, R.M. & BOGGS, R.D. ETHERNET: distributed packet switching for local computer networks. Communications of the ACM, New York, 19(7):395-404, July 1976.
- /OLS 82/ OLSEN, Robert P. Local-area networks: definitions and benefits. EDN, Boston, 27(4):110-1, Feb. 1982.
- /RAD 84/ RADDATZ, Eldo. Proposição de um protocolo de transporte para redes locais. Porto Alegre, PGCC da UFRGS, 1984. (Dissertação de Mestrado).
- /REI 81/ REINTJES, Peter B. Network tools ideas for intelligent network software. Byte, Peterborough, 6(10):140-4, 151, 163, 174, Oct. 1981.
- /ROC 82/ ROCHOL, Juergen et alii. Definição de uma tecnologia para um projeto de rede local. In: SEMINÁRIO INTEGRADO de SOFTWARE e HARDWARE, 9., Ouro Preto, jul. 12-16, 1982. Anais. Minas Gerais, SBC, 1982. p.155-71.
- /ROC 84/ ROCKWELL. R6551 asynchronous communications interface adapter (ACIA). Mar. 1984. (Product description order n.284, rev. 2).
- /SAA 81/ SAAL, Harry I. Local-area networks: possibilities for personal computers. Byte, Peterborough, 6(10):92-112, Oct. 1981.
- /SCH 81/ SCHINDLER, Max. Networks may look alike, but software makes the difference. Electronic Design, New Jersey, 29(8):121-6, Apr. 1981.
- /SUN 79/ SUNSHINE, Carl. Formal techniques for protocol specification and verification. Computer, Los Angeles, 12(9):20-7, Sept. 1979.
- /TAN 81/ TANENBAUM, Andrew S. Computer Networks. Englewood Cliffs, Prentice-Hall, 1981.
- /TOR 82/ TORNQUIST, Martin. Implementação do sistema MUMPS para o poly 101HS. Porto Alegre, PGCC da UFRGS, 1982. (Dissertação de Mestrado).

- /WAL 79/ WALDEN, David C. & MACKENZIE, Alexander A. The evolution of host-to-host protocol technology. Electronics, New York, 53(21):114-22, Sept. 1980.
- /YEN 81/ YENCHARIS, Len. Local-net communications improves at both ends. Electronics Design, New Jersey, 29(8):111-7, Apr. 1981.
- /ZIL 78/ ZILOG. Z80 technical reference manual. Cupertino, 1978.
- /ZIL 79/ ZILOG. Z80 assembler reference manual. Cupertino, 1979.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
Pós-Graduação em Ciência da Computação

Um sistema de comunicação  
para rede local de micros

Dissertação apresentada aos Srs.

Silvio Simões Torciani

---

Antônio Augusto

---

Francisco de Sales Silva

---

Jorge Rachel

---

Visto e permitida a impressão  
Porto Alegre, .16../.03../.87..

  
\_\_\_\_\_  
Coordenador do Curso de Pós-Graduação  
em Ciência da Computação