

ALTERNATIVAS DE ACESSO DE
COMPUTADORES A REDES PÚBLICAS

José da Silva Rodrigues Filho (*)

Paulo Roberto Freire Cunha (**)

GRUPO DE REDES DE COMPUTADORES E SISTEMAS DISTRIBUIDOS DA
UNIVERSIDADE FEDERAL DE PERNAMBUCO
REDIS/UFPE

RESUMO

O presente artigo apresenta algumas alternativas de conexão de computadores à uma rede pública de comutação por pacotes. Em particular, serão visto os detalhes da implementação de duas das alternativas mencionadas.

(*) Aluno de Mestrado do Departamento de Informática da UFPE.

(**) Professor Adjunto do Departamento de Informática da UFPE.

I. INTRODUÇÃO

Dentro do esforço de conectar computadores hospedeiros às redes públicas de comunicação de dados, uma grande parte do trabalho estará envolvida com a implementação de um protocolo de acesso à sub-rede de comunicação. Esse protocolo de acesso variará de acordo com os serviços oferecidos pela rede pública de pacotes. Dentre as diversas alternativas propostas, destacam-se a implementação direta do protocolo de acesso no hospedeiro, a utilização de comportas [MONT 84] e a utilização de conversores do protocolo básico de comunicação do hospedeiro (Start/Stop, BSC, etc.) para o protocolo de acesso à subrede.

No Brasil, com o aparecimento da Rede Nacional de Comutação por Pacotes (RENPAQ - Embratel), tornou-se de grande interesse na área de informática, o desenvolvimento de equipamentos e programas, que possibilitem o acesso de computadores à rede pública. Como a implementação direta do protocolo de acesso no hospedeiro, implica no desperdício da capacidade de processamento do processador principal, torna-se bastante interessante a utilização de processadores dedicados, de custo mais barato, que desempenharão as funções básicas de comunicação de dados.

Com o objetivo de conectar os recursos computacionais da UFPE aos de outras universidades, foi iniciado o estudo de alternativas de acesso à rede nacional de pacotes. Este artigo apresenta a experiência de implementação de duas das alternativas mencionadas acima. Uma delas envolveu o desenvolvimento de uma interface entre o computador DEC-10 da UFPE e um conversor "Start/Stop" - X.25, possibilitando o acesso à rede pública através do serviço X.25

[CCIT 79]. A outra consistiu na implementação de uma comporta X.25, como a descrita em [MONT 84].

II. ALTERNATIVAS DE ACESSO

Com o objetivo de criar uma padronização internacional a nível de protocolo de acesso às redes públicas comutadas, o CCITT (Comité Consultatif International Télégraphique et Téléphonique), entidade responsável pelo estabelecimento de padrões na área de comunicação de dados, criou grupos de estudo, cujo trabalho resultou na recomendação X.25, que especifica os procedimentos para interface entre um computador hospedeiro e o nó na sub-rede de comunicação.

Seguindo o padrão ISO [ZIMM 80] que hierarquizou os protocolos para redes de computadores em sete níveis, a recomendação X.25 do CCITT abrange os tres primeiros níveis do modelo ISO para interconexão de sistemas abertos.

O primeiro, o nível físico, define basicamente as características físicas, elétricas e mecânicas da interface terminal-rede. Especifica quais os tipos de conectores a serem utilizados, quais os sinais trocados com o modem, qual a velocidade de transmissão/recepção dos bits, enfim tudo que se refere a linha física de comunicação.

O segundo, o nível de enlace, garante uma confiabilidade dos dados transmitidos/recebidos pela linha de comunicação. É responsável principalmente, pelos procedimentos de detecção e correção de erros no circuito de acesso. A unidade de informação do nível de enlace é o quadro. Um quadro é um conjunto de bits, que além de conter algumas informações de controle do enlace,

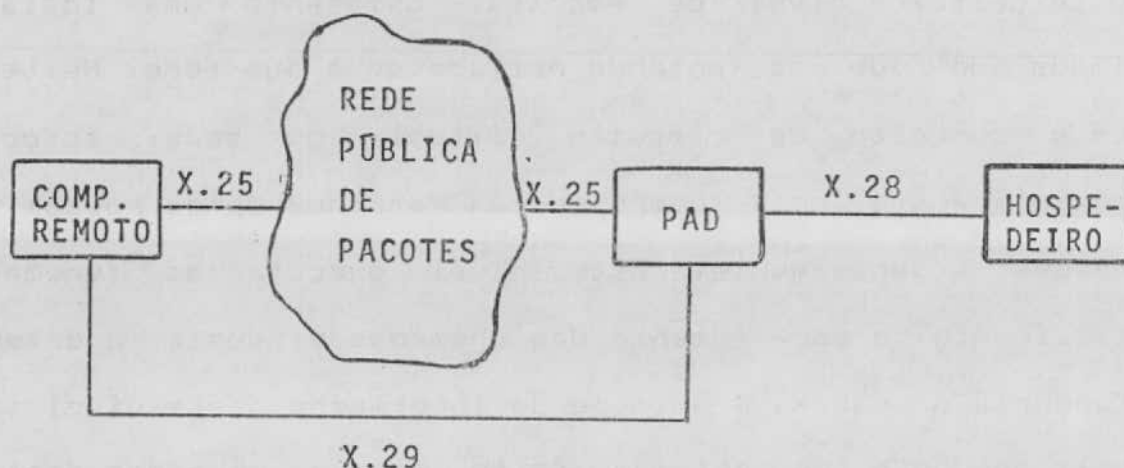
transporta informações dos níveis superiores.

O terceiro, o nível de pacotes, apresenta uma idéia mais aproximada do que se entende por acesso à sub-rede. Neste nível aparece o conceito de circuito virtual, ou seja, associações temporárias entre dois computadores para que estes possam trocar informações ordenadamente. Este nível executa as funções de estabelecimento, e encerramento das chamadas virtuais, e gerencia a transferência de dados. A unidade de informação deste nível são os pacotes, conjunto de octetos (oito bits), enviados dentro dos quadros de informação do nível de enlace.

Pelo fato de ser bastante elaborado, o protocolo X.25 implica em recursos normalmente não disponíveis em equipamentos de dados mais simples, como é o caso dos terminais assíncronos. Para permitir o acesso desse tipo de terminais, as redes públicas oferecem serviços de um PAD (Packet Assembler/Disassembler), cuja função principal é empacotar e desempacotar os dados. Em outras palavras, o PAD recebe os caracteres originados por um terminal "Start/Stop", e monta os pacotes X.25 para transmissão através da rede, executando também a operação inversa no sentido rede-terminal.

A recomendação X.28 [CCIT 80] define os procedimentos necessários para interface entre um terminal assíncrono e o PAD. Especifica o formato dos comandos que devem ser enviados para o PAD e o formato das respostas geradas pelo mesmo. A recomendação X.3 [CCIT 77], define as características de operação interna do PAD, e a recomendação X.29 [CCIT 79], define os procedimentos para interface entre o PAD e o computador remoto. A ausência destas

recomendações pode ser visualizada na figura 1 abaixo.



Fis. 1 - Contexto das recomendações X.28 e X.29

Redes públicas atuais costumam oferecer os serviços X.25 e X.28. No caso da RENPAC, os serviços oferecidos são os seguintes:

1. **Serviço 3025** - Destina-se aos terminais de dados que operam no modo pacote, isto é, terminais que se ligam à rede através do protocolo X.25.
2. **Serviço 3028** - Atende aos terminais de dados que operam no modo caracter, isto é, terminais que se ligam à rede através do protocolo X.28.
3. **Serviço 2000** - Destina-se à mesma categoria de terminais do serviço 3028, mantidos os critérios de conexão via PAD. A diferença é que o serviço 2000 caracteriza-se pelo uso de acessos comutados da Rede Pública de Telefonia.
4. **Serviço 1000** - Possibilita a conexão, via interface PAD, de terminais da Rede Nacional de Telex, utilizando acessos comutados.

Uma das alternativas de acesso abordadas neste artigo compreende a utilização de uma comporta X.25, ou seja, um processador dedicado com função de implementar o protocolo X.25. Uma comporta possui duas vias por onde fluem os dados. Uma que se relaciona com a sub-rede e a outra que se relaciona com o computador hospedeiro. A função básica da comporta será efetuar a tradução do protocolo de comunicação do sistema que desejamos conectar à rede, para o protocolo X.25, que é o protocolo de acesso à sub-rede. A figura 2 abaixo mostra o esquema de ligação da comporta.

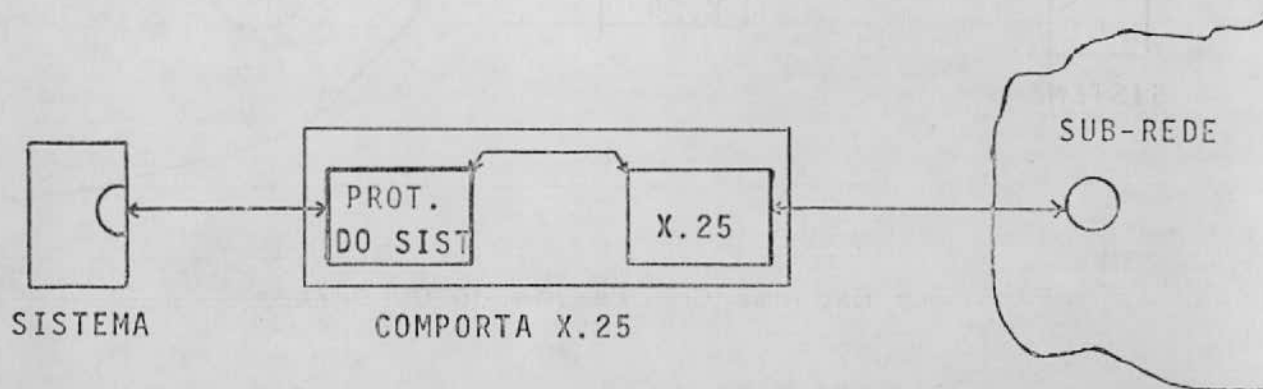


Fig. 2 - Esquema de Ligação da Comporta

Podemos notar que do lado do hospedeiro, a comporta deve executar o protocolo de comunicação do mesmo, e do lado da sub-rede deve executar o protocolo X.25. No caso de RENPAC a comporta seria conectada à rede através do serviço 3025, num acesso comutado síncrono.

Outra alternativa abordada neste artigo consiste na utilização de um conversor de protocolo. Este tem o objetivo de permitir o acesso de computadores à uma rede pública, liberando o processador principal das tarefas básicas de comunicação de dados. Em

particular o conversor X.25, é um processador encarregado do controle da comunicação de dados, como também da interface com o computador hospedeiro. Ele executa em sua interface local um protocolo que é uma simplificação da recomendação X.28 e na interface com a rede ele opera segundo a recomendação X.25. A figura 3 a seguir ilustra o esquema de ligação do conversor X.25.

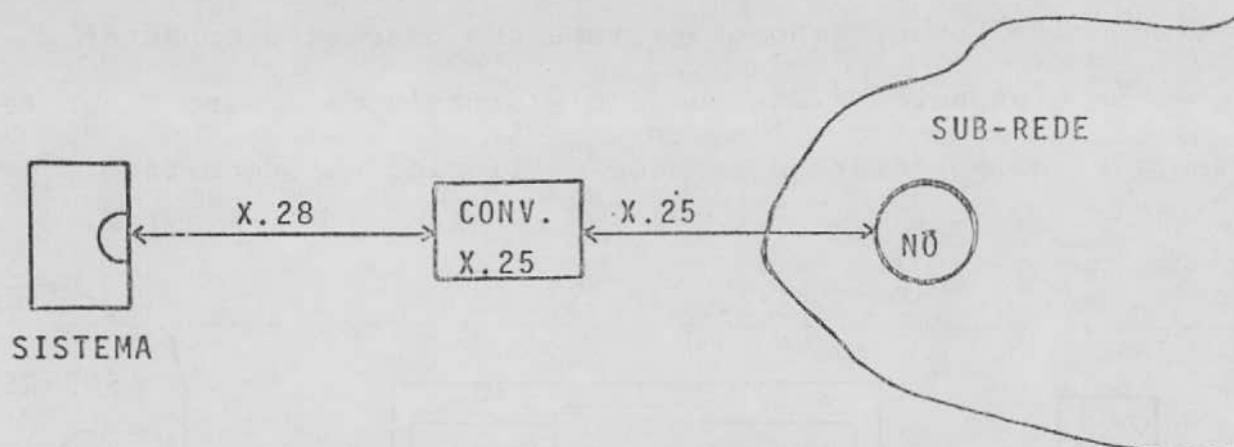
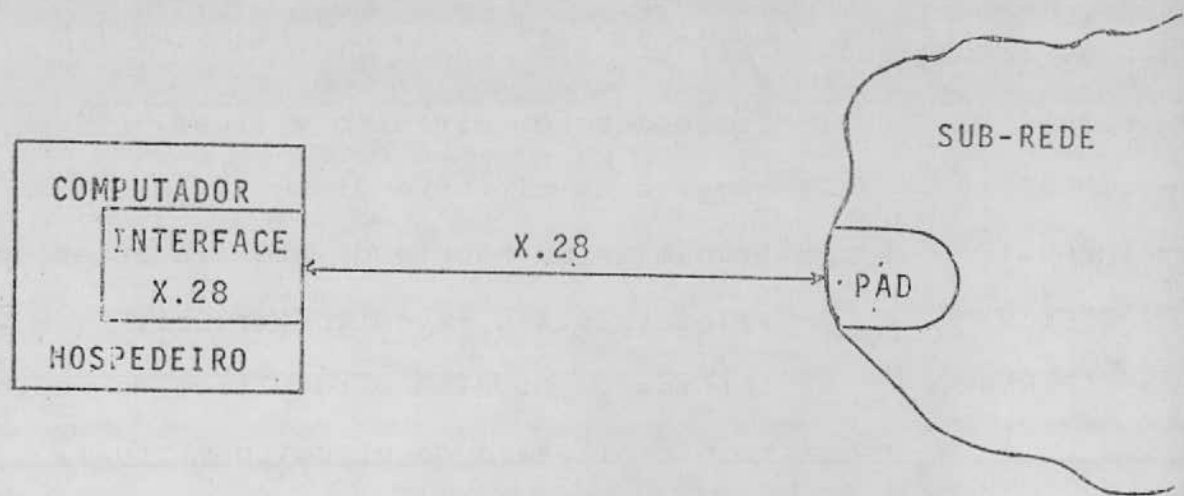


Fig. 3 - Esquema de Ligação do Conversor

Como a recomendação X.28 funciona através de uma troca de mensagens entre o terminal e o PAD, de acordo com procedimentos bem definidos, a obediência a estes procedimentos implica na necessidade de implementar no computador hospedeiro a "interface X.28", que terá a função de gerar e interpretar as mensagens de controle do conversor além de fornecer os serviços do nível de rede ao nível superior.

Deve-se notar que o desenvolvimento de uma interface X.28 em um computador com linhas assíncronas, viabiliza uma terceira alternativa de conexão. De fato, poderíamos, com uma pequena adaptação, conectar o computador hospedeiro ao serviço X.28 da rede pública. No caso da RENFAC seria utilizado o serviço 3028, sem a

necessidade de utilização do conversor X.25. Esta nova alternativa é ilustrada na figura 4 abaixo.



Fis. 4 - Utilização do Serviço X.28

Após esta abordagem inicial das alternativas existentes para conexão de computadores à redes públicas, apresentaremos nas seções seguintes, os detalhes de implementação da comporta X.25 em um microcomputador, e da interface X.28 no DEC-10 da UFPe.

III. SOLUÇÃO CONVERSOR X.25

A principal vantagem de utilização do conversor X.25 para o acesso de computadores à redes públicas, é que ele libera o computador hospedeiro das tarefas básicas de comunicação de dados. Assim, ao invés da necessidade de implementação de uma interface X.25, basta desenvolver no computador hospedeiro uma interface que manipule corretamente o conversor.

A função básica desta interface será a de fornecer os serviços do nível de rede ao nível 4 (transporte). Para isto ela utiliza os serviços oferecidos pelo conversor. A interação entre o nível 4 e a nossa interface (denominada interface X.28), será feita

utilizando-se de um conceito de "primitivas" que implementam os serviços necessários. As primitivas implementadas pela interface são as seguintes:

- ESTABCV - estabelece um circuito virtual.
- LIMPACV - encerra um circuito virtual.
- TRPACOTE - transmite um pacote de até 128 bytes de dados.
- RESETCV - reinicializa o circuito virtual.
- RESP_RESETCV - informa ao nível 4 o resultado do pedido de reinicialização do circuito virtual.
- RESP_ESTABCV - informa ao nível 4 o resultado do pedido de ESTABCV.
- IND_ESTABCV - informa ao nível 4 um pedido remoto de estabelecimento de uma chamada virtual.
- IND_RESET - informa ao nível 4 a ocorrência de uma reinicialização naquele circuito virtual.
- IND_FIMCV - informa uma desconexão remota naquele c.v.
- RESP_LIMPACV - informa o resultado do pedido de LIMPACV.

Desta forma podemos visualizar o software da interface como sendo composto por dois módulos principais:

1. O módulo que se encarrega da interação com o nível 4 (PTRN4).
2. O módulo que cuida da operação correta do conversor (PTRCX).

Além desses dois citados acima existem mais dois módulos com funções bem definidas. Um deles é o módulo (PDRIVER) que manipula as interfaces assíncronas do computador hospedeiro. Ele se encarrega de transmitir os caracteres para o conversor e de receber

os caracteres vindos do conversor. O outro é o módulo que tem como função inicializar as interfaces com o nível de transporte e com o conversor.

A figura a seguir ilustra a estrutura proposta para o software da interface X.28:

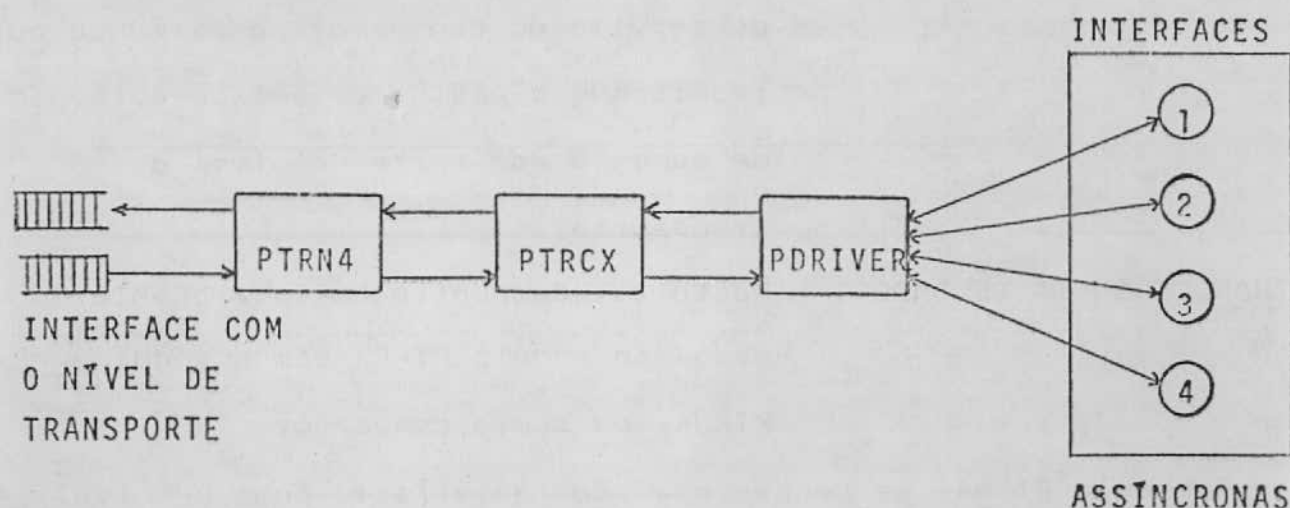


Fig. 5 - Estrutura de Software da Interface X.28

A seguir detalharemos os módulos que compõem a estrutura proposta para o software da interface.

III.1 ESTRUTURA DA INTERFACE X.28

O módulo PTRCX é quem opera o conversor CX.25. Ele é quem monta e desmonta as mensagens que são transmitidas e recebidas respectivamente. Ele interpreta os caracteres vindos do conversor indentificando a mensagem e o evento a ela associado.

O módulo possui uma tabela associada a conexão cujos campos são os seguintes: O estado da ligação com o conversor e a "UDX" da interface, que é um parâmetro do sistema identificando a porta física pela qual está sendo feito o intercâmbio de informações.

A esse nível de interação identificamos tres estados possíveis:

Para a conexão:

- ESPERA COMANDO - Neste estado o módulo aceita comandos de estabelecimento de conexão vindos do nível de transporte ou do conversor.
- SERVIÇO DO CONVERSOR - Neste estado o módulo aguarda um sinal de serviço do conversor confirmando ou rejeitando o pedido de estabelecimento de conexão por parte do nível de transporte.
- TRANSFERENCIA DE DADOS - Neste estado ocorre um intercâmbio de dados do módulo PTRCX com o módulo PTRN4, ou com o conversor.

O módulo PTRN4 se encarrega da interface com o nível 4 (transporte). Ele simula as funções de nível de rede, ficando portanto transparente para a estação de transporte a utilização do conversor como método de acesso à rede pública. Ele se encarrega do gerenciamento dos circuitos virtuais fornecendo um conjunto de serviços que tornam possível o estabelecimento/encerramento/reinicialização do canal virtual, como também da transferência de dados, que permite o intercâmbio de informações entre as estações de transporte envolvidas na conexão.

O módulo possui uma tabela associada ao circuito virtual contendo as seguintes informações:

- . ENDEREÇO REMOTO - Contém o endereço da entidade remota envolvida na conexão.
- . IND-BLOC-LIG - Contém uma indicação do bloco de ligação da conexão de transporte.

- . ESTADO - Informa o estado atual do canal lógico.
- Os estados por que passa o canal lógico durante a conexão são os seguintes:
- . LIVRE - Neste estado o canal está pronto para ser utilizado.
 - . ESPERA CONFIRMAÇÃO DE CONEXÃO - Neste estado o módulo espera uma resposta do pedido de conexão enviado.
 - . TRANSFERENCIA DE DADOS - Neste estado ocorre o intercâmbio de dados entre as estações de transporte envolvidas na conexão.
 - . PEDIDO DE DESCONEXÃO - Asguarda-se uma resposta ao pedido de desconexão enviado.

O módulo PDRIVER manipula as interfaces assíncronas do computador hospedeiro. Ele é composto de um submódulo de "recepção" que retira os caracteres das interfaces ativas e do submódulo de transmissão que se encarrega de enviar os caracteres para as interfaces.

O submódulo de recepção possui certa inteligência no sentido de detectar o final do pacote, como também distinguir se o pacote recebido é de dados ou de comando, já informando ao módulo PTRCX, que por sua vez analisará o conteúdo do pacote de comando e simplesmente passará adiante o conteúdo do pacote de dados. Este submódulo é ativado sempre que chegar algum caracter em qualquer das interfaces ativas. Sua execução compreende a retirada do caracter do buffer da interface para posterior análise. Neste ponto a análise é feita a nível de caracter, para identificação de certos eventos tais como: final do pacote, chegada de um prompt (caracter

enviado pelo conversor para indicar fase de comandos ou dados), chegada de um sinal de serviço, etc.

O submódulo de transmissão retira os pacotes (comandos ou dados) da fila de pacotes a transmitir de acordo com os PROMPT'S enviados pelo conversor. Cada interface possui sua própria fila de pacotes a transmitir, com a finalidade de evitar congestionamentos caso haja alguma interface a espera do PROMPT. Uma vez ativado este submódulo transmite um pacote de até 128 caracteres para cada interface ativa que contenha algum elemento na fila de pacotes a transmitir.

III.2 IMPLEMENTAÇÃO DA INTERFACE X.28 NO DEC-10

A interface entre o conversor e o protocolo de transporte foi implementada no sistema DEC - 1091 da Universidade Federal de Pernambuco. Este sistema possui um sistema operacional por partilha de tempo (TOPS10), e faz o chaveamento de contexto considerando um processo (Job) como unidade de processamento.

A interação entre a interface X.28 e o nível 4 é feita utilizando-se de um utilitário do sistema chamado IPCF-10 (InterProcess Communication Facility) [DEC 82]. Este software permite uma comunicação entre processos, a qual é feita através de filas de mensagens. Existem dois modos de transmissão disponíveis: o "Small Mode" que permite que a mensagem tenha um comprimento variando entre 1 (uma) e 10 (dez) palavras, e o modo "Normal" que permite que a mensagem tenha um tamanho entre 1 (uma) e 255 palavras. Um processo possui um identificador dentro do sistema denominado PID (Process Identifier). Assim, para enviar uma mensagem

para um determinado processo, é suficiente especificar sua PID e enviar a mensagem, que será colocada na fila de recepção do processo destino.

Para o acesso físico ao conversor X.25 foi utilizada a interface assíncrona do PDP-11, que é o processador "front-end" da configuração do DEC-10 da UFPe. Tal interface denominada "DZ-11" [DEC 79], pode controlar até dezesseis linhas assíncronas. Algumas alterações devem ser efetuadas no software do PDP-11 para adaptar as interfaces à conexão com o conversor, pois originalmente estas interfaces são usadas para conexão de terminais de vídeo ao computador.

A linguagem escolhida para a implementação foi o Pascal, que apesar de não ser uma linguagem suportada pela DEC (Digital Equipment Corporation) não apresentou problemas durante a implementação e testes. Toda a interface foi programada em alto nível, inclusive o driver que manipula as interfaces do PDP-11. O software do IPCF-10 também está disponível para ser usado através de uma linguagem de alto nível (Ex: Pascal, Fortran, Cobol, etc.).

IV. SOLUÇÃO COMPORTA X.25

Uma das alternativas de se conectar um computador hospedeiro à uma rede pública envolve a utilização de um processador dedicado, denominado "Comporta". Com o uso de tal equipamento conseguimos eliminar do computador, a tarefa de executar o protocolo X.25, liberando-o para outros tipos de processamento.

Como sabemos, entre as funções de uma comporta, está a de realizar os três níveis do protocolo X.25. Por outro lado, a

comporta deve se comunicar com o hospedeiro, através do protocolo de comunicação do mesmo, formando assim o elo de ligação desejado entre o sistema e a sub-rede.

A comporta proposta neste trabalho será utilizada para conectar o computador DEC-10 da UFPe à uma rede pública de pacotes. Para desempenhar tal função ela foi estruturada em dois módulos com funções bem definidas: o módulo X.25, que realiza a interface com a rede e o módulo hospedeiro que tem o objetivo de realizar a interface com o DEC-10. Ela será implementada em um microcomputador da Embracom (SDE-42), no qual foi inserida uma placa que executa o nível físico do X.25 e algumas funções do nível de enlace. Esta placa é baseada numa interface de link 8273 (Intel).

Conforme podemos observar na figura 6, a comporta possui duas portas de comunicação: uma com o nó da sub-rede e a outra com o sistema local. A primeira deve seguir a padronização imposta pela recomendação X.25. A segunda segue as características do protocolo de comunicação do sistema DEC-10, o DDCMP (Digital Data Communications Message Protocol) [DEC 74].

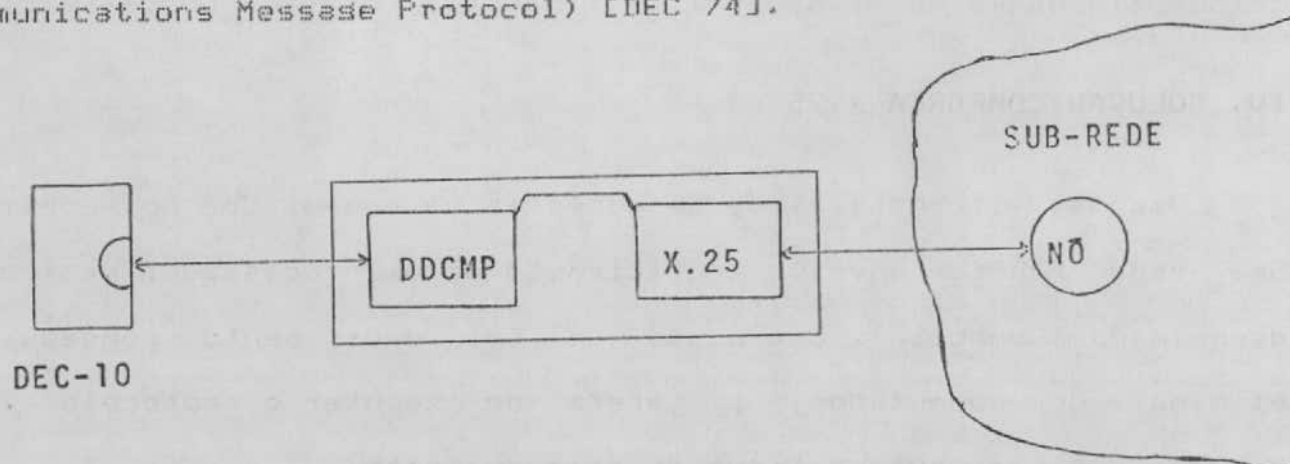


Fig. 6 - Estrutura da Comporta

IV.1 ESTRUTURA DO MÓDULO X.25

O software de cada nível do protocolo X.25 na comporta está estruturado em diversos módulos seguindo a proposta apresentada em [QUEI 85]. Cada módulo pode ser um processo ou um monitor, dependendo de suas características ativa ou passiva, respectivamente.

Dois processos são ditos concorrentes se as suas execuções se sobrepõem no tempo [HANS 73]. Se estes dois processos são não disjuntos, i.e., se em algum ponto eles precisam trocar informações para prosseguir em suas execuções, um mecanismo de sincronização faz-se necessário.

Dentre os mecanismos de sincronização propostos, escolhemos os monitores. Monitor é um conjunto de procedimentos que operam sobre variáveis comuns a vários processos. Um tipo de dados (ex: filas, tabelas) definido pelo monitor representa variáveis comuns a vários processos paralelos e só poderão ser acessadas pelos processos através de chamadas aos procedimentos do monitor.

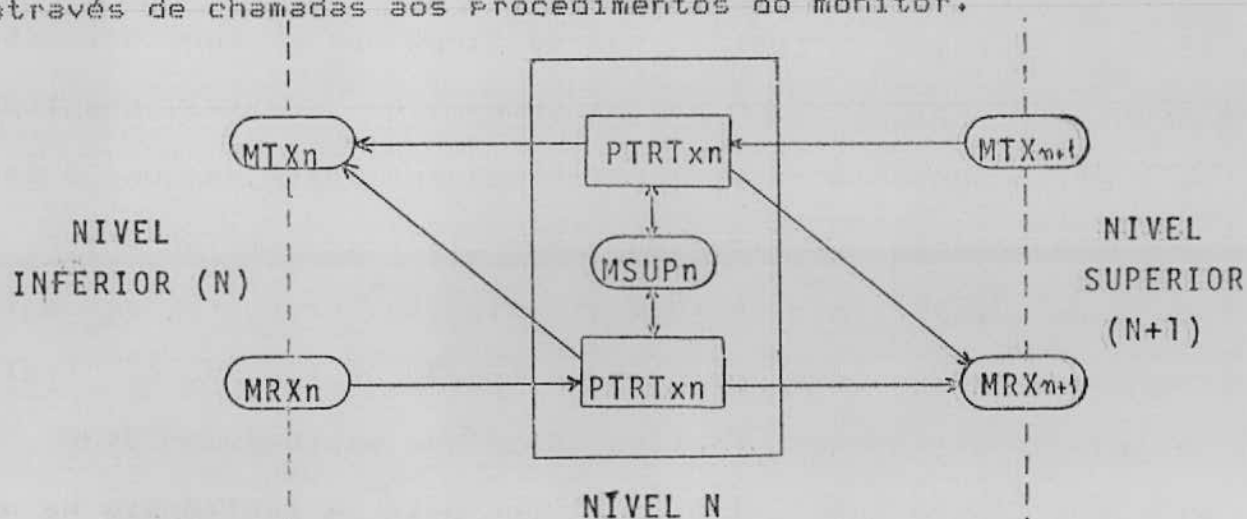


Fig. 7 - Estrutura de Software do Nível N

Na figura 7 acima podemos ver a estrutura de software de cada

nível da comporta (nível de enlace e nível de pacotes do X.25). Os arcos direcionados representam fluxo de dados e não de controle. Os monitores (envoltos em círculos) são sempre elementos passivos, enquanto que os processos são os elementos ativos, i.e., os processos tem execução própria, e invocam os procedimentos do monitor.

A interface entre os níveis é realizada através de quatro monitores. Cada um destes monitores consiste, basicamente, de uma fila de mensagens (comandos ou respostas) originadas em um nível e destinadas a um dos níveis adjacentes. Ou seja, temos um nível que "produz" mensagens, e outro que "consome" mensagens, de acordo com o sentido dos arcos. Os procedimentos de acesso a esta fila são os de inserção (colocar na fila) e de retirada de mensagens. Em alguns casos particulares, existem outros procedimentos e funções associadas.

No monitor de supervisão do nível "N" (MSUP_N) são guardadas informações referentes ao estado das conexões do nível. Os procedimentos de acesso as estruturas de dados deste monitor, são os de consulta e atualização destas informações. Certas consultas podem bloquear o acesso a estas informações por parte de um outro processo, até que a atualização seja feita. Este mecanismo é utilizado para evitar "corridas" de atualização por parte de dois processos, que podem resultar numa situação de impasse do nível. Assim procedendo garantimos a "atomicidade" da execução de ações e possível atualização do estado em resposta à ocorrência de um determinado evento.

Os processos de tratamento de mensagens enviadas do nível "x"

Para o nível "n" (PTRTxn) tem como funções: retirar mensagens destinadas ao nível "n" pelo nível "x"; identificar a mensagem (análise sintática da mesma); determinar qual evento foi disparado; consultar o estado da conexão ou do nível (em MSUPn) e executar as ações correspondentes que podem consistir, além da atualização do estado e/ou de outras informações, em transmissões de mensagens para o nível inferior e/ou para o nível superior.

Outra parte da nossa estrutura de software seria o módulo responsável pela temporização. Entende-se por temporização o mecanismo em que o transmissor guarda uma cópia da mensagem transmitida e aguarda uma confirmação de recebimento. Se esta confirmação não chegar dentro de um certo intervalo de tempo, ocorre um estouro do temporizador e a mensagem é retransmitida. Para implementar temporizadores é preciso usar um relógio de hardware, que gere pedidos de interrupção em intervalos regulares de tempo, sendo este intervalo determinado de acordo com o tipo de aplicação, correspondendo a uma unidade de temporização. O tratamento deste pedido de interrupção consistiria na decrementação de uma unidade das variáveis, que representam os diversos temporizadores do sistema. Se um destes temporizadores for decrementado até "0" (zero) correspondendo ao estouro de temporização, então, o nível correspondente deverá ser notificado da ocorrência deste evento.

Seguindo o esquema apresentado para a recepção de mensagens do nível inferior ou do superior, o processo de tratamento de interrupção do relógio (PTIMER), ao detectar o estouro de um determinado temporizador insere, numa fila do monitor de

temporização (MTMRn), a indicação correspondente. Faz parte da estrutura, um processo que retira da fila estas indicações, e faz o tratamento destes eventos. Este processo de tratamento de temporizações (PTMRn) do nível "n" possui funções análogas às do PTRTxn.

IV.2 ESTRUTURA DO MÓDULO HOSPEDEIRO

A comporta será conectada ao DEC-10 através do protocolo de comunicação do mesmo, ou seja, o DDCMP [DEC 74]. Este protocolo especifica a estrutura, o conteúdo e os procedimentos para a transmissão de dados entre computadores, incluindo as técnicas para detecção de erros. Ele opera sobre linhas assíncronas ou síncronas, full-duplex ou half-duplex, em configurações ponto-a-ponto ou multiponto.

A operação do DDCMP envolve as seguintes funções: enquadramento, serenciamento do link e troca de mensagens. Por enquadramento entende-se o processo de determinar o início e o final de uma mensagem. Este processo pode requerer uma sincronização a nível de bit, byte ou mensagem. Por serenciamento do link entende-se o processo de controlar a transmissão e recepção nos enlaces half-duplex ou multiponto, quando existem mais de um transmissor e mais de um receptor. A troca de mensagens relaciona-se com a transferência de dados do usuário através do link, mantendo a sequência original e sem erros de transmissão.

Entre as funções do módulo na comporta podemos citar: montagem das mensagens DDCMP, recepção de mensagens DDCMP, desmontagem das mensagens, cálculo do CRC, etc. O acesso à linha física será

realizado através de uma interface de link 8251 [Intel]. A configuração da ligação é full-duplex ponto-a-ponto, simplificando assim as funções a serem desempenhadas pelo módulo. No caso, a parte de gerenciamento do link não se faz necessária, uma vez que as estações estão sempre habilitadas nesta configuração.

A divisão do DDCMP em submódulos resulta na estrutura mostrada na figura 8 abaixo.

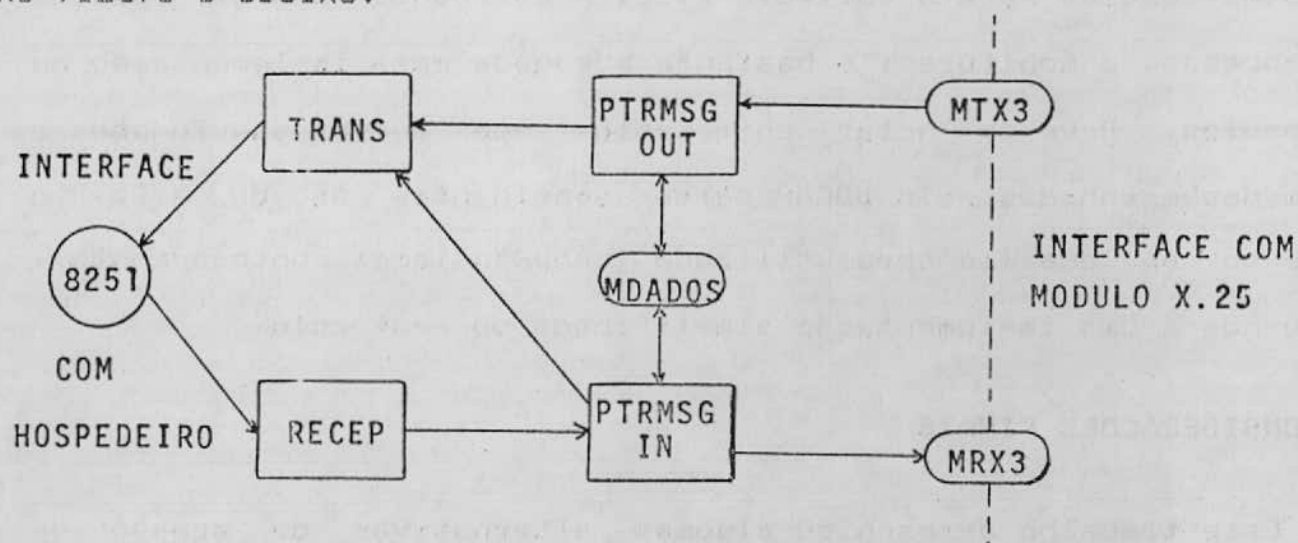


Fig. 8 - Estrutura do Módulo Hospedeiro

Os processos TRANS (Transmissão para o Enlace) e RECEP (Recepção do Enlace) manipulam a interface 8251 (Intel) transmitindo ou recebendo mensagens através da linha de comunicação. O submódulo PTRMSGIN (Processo de Tratamento de Mensagens Recebidas) realiza a identificação da mensagem, tomando ações correspondentes. O processo PTRMSGOUT (Processo de Tratamento de Mensagens a serem Transmitidas) se encarrega da parte de montagem das mensagens DDCMP, inserindo os campos que compõem o cabeçalho, e a parte do CRC. O MDADOS (Monitor de Dados) contém as informações necessárias à operação do protocolo, e as rotinas de acesso a estas informações. O processo PTIMER (temporizador) é o

mesmo do módulo X.25, e será utilizado para temporizar as mensagens transmitidas. Existe ainda uma rotina que terá a função de calcular o CRC associado à um bloco de informações. Ela será utilizada tanto na parte de checagem das mensagens recebidas, como na montagem de mensagens que devem ser transmitidas.

Podemos notar que a estratégia utilizada na implementação foi a mesma seguida para o software X.25. A estrutura modular, baseada em processos e monitores, é bastante adequada para implementação de protocolos. Deve-se notar entretanto, que apesar das funções a serem desempenhadas pelo DDCMP serem semelhantes às do X.25, o contexto em que ele será utilizado (conexão local ponto-a-ponto), levou-nos a uma implementação simplificada do protocolo.

V. CONSIDERAÇÕES FINAIS

Este trabalho apresentou algumas alternativas de acesso de computadores à redes públicas. Duas das possíveis alternativas foram detalhadas à nível de implementação. Dentro da solução "Conversor X.25", foi proposta uma estrutura para o software da interface entre o computador que desejamos conectar à rede e o referido conversor. Devemos lembrar que esta implementação viabiliza o acesso de um computador à rede pública através do serviço X.28. O software já está pronto, tendo sido testado localmente em conjunto com o módulo que implementa a estação de transporte [CUNH 84]. Restam apenas os testes integrados que dependem da aquisição do conversor X.25. Com relação à comporta X.25, o software do protocolo X.25 já está pronto, estando em fase de implementação final e testes o módulo DDCMP.

VI. REFERENCIAS

- [CCIT 77] CCITT RECOMENDATION X.3 - **Packet Assembly/Disassembly Facility (PAD) in a Public Data Network**, Geneva, 1977.
- [CCIT 79] CCITT RECOMENDATION X.25 - **Interface Between Data Terminal Equipment and Data Circuit-Terminating Equipment for Terminals Operating in the Packet Mode on Public Data Networks**, Geneva, 1979.
- [CCIT 80] CCITT RECOMENDATION X.28 - **DTE/DCE Interface for a start-stop Data Terminal Equipment Accessing the Packet Assembly/Disassembly Facility (PAD) in a Public Data Network situated in the same Country**, Geneva, 1980.
- [CCUNH 84] CUNHA, P.R.F., MONTEIRO, J.A.S., LUCENA, E.B. - **Implementação de um Protocolo de Transporte para a Rede CEPINNE**, submetido ao 3o. Simpósio Brasileiro de Redes de Computadores, Janeiro, 1985.
- [DAVI 79] DAVIES, D.W., BARBER, D.L.A., PRICE, W.L., SOLOMONIDES, C.M. - **Computer Networks and Their Protocols**, New York: John Wiley, 1979; pp. 344 à 347.
- [DEC 74] DIGITAL EQUIPMENT CORPORATION - **DDCMP, Digital Data Communications Message Protocol**, Edition: 3, Dec., 1974.
- [DEC 79] DIGITAL EQUIPMENT CORPORATION - **Terminals and Communications Handbook - 1979**, capítulo 7, p. 145-159.
- [DEC 82] DIGITAL EQUIPMENT CORPORATION - **Communication Between Processes: IPCF**, Software Notebook 4, July 1982, p. 8.1 - 8.24.
- [HANS 73] HANSEN, P.B. - **Concurrent Programming Concepts**, Computing Surveys, 5(4): 233-45, Dec., 1973.

- [MONT 84] MONTEIRO, J.A.S., JUREMA, M.A., CUNHA, P.R.F. - **Comporta
para Conexão de um Computador DEC-10 à uma Rede Pública
de comunicação de Dados**, in: 2. Simpósio Brasileiro so-
bre Redes de Computadores, Anais, Campina Grande, 1984,
P. 2.1 - 2.21.
- [QUEI 85] QUEIROZ, R.J.B., CUNHA, P.R.F., - **Um Modelo de Progra-
mação para Implementação de Protocolos**. in: Revista
Brasileira de Computação, Rio de Janeiro, V.4, No.1,
1984/1985, pp. 37 a 65.
- [ZIMM 80] ZIMMERMANN, H. - **OSI Reference Model - The ISO Model
of Architecture For Open Systems Interconnection**. IEEE
Transactions on Communications, 28(4):425-32, Apr., 1980.