

3º SIMPOSIO BRASILEIRO DE REDES DE COMPUTADORES (3º SBRC)

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

USO DE UM CONVERSOR "START/STOP" - X.25 PARA O
ACESSO DE COMPUTADORES A REDES PUBLICAS

RECIFE

DEZEMBRO 1984

USO DE UM CONVERSOR "START/STOP" - X.25 PARA O
ACESSO DE COMPUTADORES A REDES PÚBLICAS

José Augusto Suruagy Monteiro (*)

Paulo Roberto Freire Cunha (**)

José da Silva Rodrigues Filho (***)

RESUMO

O presente artigo apresenta a experiência de implementação da interface entre um computador hospedeiro e um conversor "start/stop" - X.25, viabilizando a utilização do referido conversor como alternativa de acesso à uma rede pública de comunicação por pacotes.

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

I. INTRODUÇÃO

Durante o processo de implantação de Redes de Computadores, uma das questões básicas diz respeito ao modo como serão conectados os computadores hospedeiros à sub-rede de comunicação. Diversas alternativas foram propostas [DEAT 83], dentre elas destacam-se a implementação direta do protocolo de acesso ao 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 rede.

Dentro do projeto CEPINNE (Centro Piloto de Serviços de Teleinformática para Aplicações em Ciência e Tecnologia), que prevê a interligação dos sistemas computacionais das universidades federais do Norte e Nordeste, foi firmado em Janeiro de 1983 um convênio envolvendo a UFPe, a UFPb e a EMBRATEL para o desenvolvimento do "software" do protocolo de transporte a ser utilizado na Rede CEPINNE [CUNH 84]. Neste convênio cabia a Embratel colocar à disposição de cada uma das universidades um conversor X.25 ("start/stop" - X.25).

Do ponto de vista da UFPe tínhamos como alternativa a utilização do referido conversor ou da comporta descrita em [MONT 84], mas que levaria ainda algum tempo até ser concluída. Devido à perspectiva de disponibilidade do conversor assim que necessário, além do interesse em analisarmos uma alternativa diferente para a conexão de computadores a redes é que optamos pelo uso do conversor.

O objetivo deste artigo consiste em apresentar a experiência de implementação da interface entre um computador hospedeiro e um conversor "start/stop" - X.25, de agora em diante denominado de conversor X.25, ou simplesmente conversor.

A seção II apresenta as funções básicas do Conversor X.25 e do protocolo de acesso ao mesmo, enquanto que a seção III descreve as características da interface entre o protocolo de transporte e o conversor, e, finalmente, a seção IV apresenta a experiência de implementação desta interface no DEC-10 da UFPe.

II. O CONVERSOR X.25

O conversor de protocolos tem o objetivo de permitir o acesso de Equipamentos Terminais de Dados (ETD) à uma rede pública comutada por pacotes, liberando o computador hospedeiro das tarefas básicas de comunicação de dados. Para tanto existe uma conversão do protocolo de comunicação do ETD para o protocolo de acesso à rede pública.

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 [CCIT 80] e na interface com a rede ele opera segundo a recomendação X.25 [CCIT 79].

A recomendação X.28 define os procedimentos necessários para interfaceamento entre um ETD assíncrono e o PAD (Packet Assembler/Disassembler). Podemos considerar o conversor como um PAD, pois ele executa as seguintes funções características de um

PAI [CCIT 77]:

- Montagem e desmontagem de pacotes
- Gerenciamento e execução de procedimentos para estabelecimento de chamadas virtuais, finalização, interrupção ou reinicialização dos mesmos.
- Geração de mensagens de serviço
- Edição
- Controle do fluxo de dados.

O esquema abaixo mostra um cenário típico do estabelecimento de um circuito virtual e transferência de dados através de um conversor X.25 seguindo a recomendação X.28. Os caracteres enviados pelo conversor estão em negrito, enquanto os caracteres normais são os teclados pelo usuário:

- | | |
|---------------------|--|
| >SEL:098<CR> | - Pedido de estabelecimento de um circuito virtual |
| CON<CR> | - Confirmação de conexão estabelecida |
| *<dados do usuário> | - Início da fase de transferência de dados |
| . | |
| . | |
| . | |
| <ESC> | - Sai da transferência de dados |
| >CLR<CR> | - Solicita encerramento do circuito virtual |

No exemplo de uso acima os comandos e sinais de serviço estão de acordo com o manual de operação do conversor CX.25 da Icatel.

Quando do início dos serviços o conversor envia o caracter '>' que indica que ele está apto a receber comandos. Após receber o comando 'SEL:098<CR>' ele sera um pacote de 'CALL REQUEST' [CCIT 79] para solicitar o estabelecimento de um circuito virtual. Quando da chegada do pacote 'CALL CONNECTED' o conversor envia para o usuário o sinal de serviço '<BEL>COM<CR>' confirmando o estabelecimento do circuito virtual. Se o ETD remoto enviar o pacote de 'CLEAR INDICATION' (indicando que não pode ou não deseja estabelecer o c.v.), o conversor envia para o usuário local um sinal de serviço de 'CLEAR', indicando a causa.

Após o estabelecimento de um circuito virtual, a conexão entra na fase de transferência de dados. A saída desta fase pode ser feita por meio de uma desconexão remota ou através de um 'Escape' (usuário envia caracter de escape (DLE)) da transferência de dados. Após enviar o escape e receber o prompt de comando o usuário pode enviar qualquer comando que desejar. A fase de transferência de dados é distinguida da fase de comando através de prompts distintos, onde o caracter '*' denota o prompt de dados, enquanto o caracter '>' representa o prompt de comando. Na fase de comando o usuário pode requerer uma desconexão do circuito virtual estabelecido, ou pode solicitar uma reinicialização da conexão. Para o usuário distinguir os dados recebidos dos sinais de serviço, o conversor envia antes destes sinais o caracter CTRL-G <BEL>.

O PAD possui certos parâmetros cujos valores formam um conjunto denominado de 'perfil' do PAD. Estes parâmetros e seus possíveis valores estão definidos na recomendação X.3 [CCIT 77]. Terminais com características diferentes requerem um determinado conjunto de valores que varia de acordo com o tipo de utilização do PAD. Os valores dos parâmetros podem ser consultados ou modificados através de comandos especiais, formando perfis distintos.

Como a recomendação X.28 funciona através de uma troca de mensagens entre o ETD 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 de nível de rede ao nível de transporte.

III. A INTERFACE X.28

A função básica de nossa interface é a de fornecer os serviços de nível de rede ao nível 4. Para isto ela utiliza os serviços oferecidos pelo conversor X.25. Esta interação entre os dois níveis é feita através 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

- RESP_ESTABCV - 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.

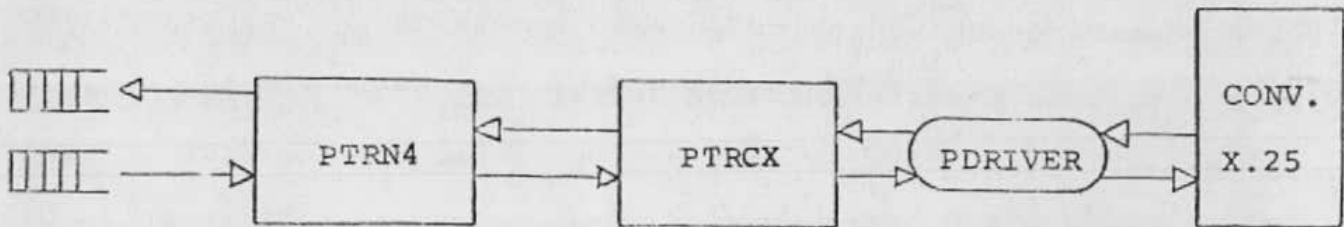
Outra função da interface é a de operar corretamente o conversor X.25, serando as mensagens de comando necessárias para o estabelecimento/reinicialização/encerramento da chamada virtual, e interpretando os sinais de serviço enviados.

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

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

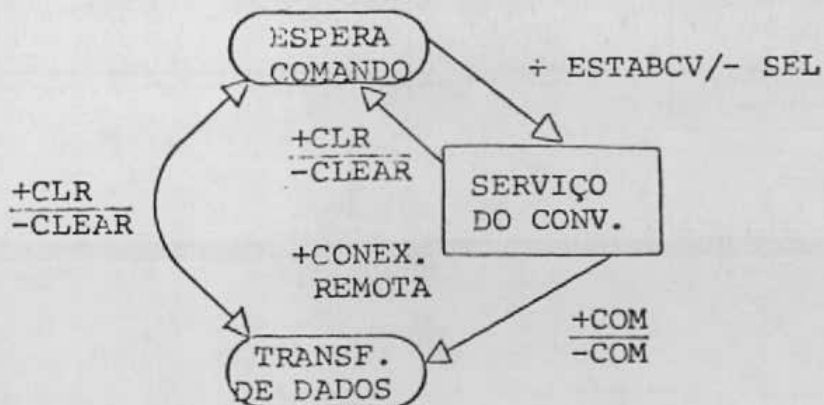
Além desses dois citados acima existem mais dois módulos com funções bem definidas. Um deles é o módulo (FDRIIVER) que manipula as interfaces assíncronas do computador hospedeiro. Ele se encarrega da transmissão e recepção dos caracteres vindos do conversor. O outro é o módulo que tem como função inicializar 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:



filas de interação com o nível 4

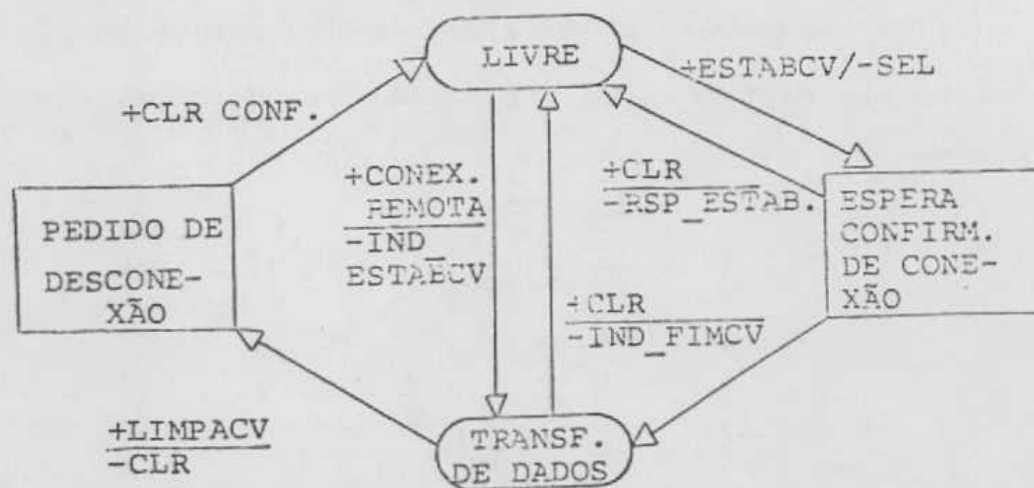
O módulo PTRCX é quem opera o conversor CX.25. Ele é quem monta e desmonta os pacotes que são transmitidos e recebidos respectivamente. Ele interpreta os caracteres vindos do conversor identificando a mensagem e o evento a ela associado. Ele opera segundo uma tabela de transição por estados cujo diagrama é o seguinte:



No diagrama acima as transições são disparadas de acordo com os eventos ocorridos. No estado de "Espera Comando" o módulo aceita comandos vindos do nível de transporte ou do conversor. Um pedido

de ESTABCV causa uma transição para o estado de "Serviço do Conversor". Neste estado a interface aguarda um sinal de serviço do conversor confirmando ou rejeitando o pedido de estabelecimento do c.v. A chegada de um pedido de conexão remota causa uma transição para o estado de transferência de dados, pois neste ponto o c.v. já está estabelecido. A chegada de um sinal de serviço "COM" (confirmação de conexão) causa uma transição para transferência de dados. A chegada de um "CLEAR" vindo do nível de transporte ou do conversor, faz com que a interface volte ao estado de "Espera Comando".

O módulo PTRN4 se encarrega da interface com o nível de transporte. Ele opera segundo uma tabela de transição por estados cujo diagrama é o seguinte:



No estado "Livre" o canal lógico está pronto para ser utilizado. A chegada de um pedido de ESTABCV do nível de transporte faz com que o canal passe para o estado "Espera Confirmação de Conexão", ao mesmo tempo em que um comando de seleção é enviado ao conversor. A chegada de um "COM" faz com que o canal passe para

o estado de "Transferência de Dados". A chegada de um sinal de conexão remota faz com que o canal passe do estado "Livre" para o de "Transferência de Dados". Estando neste estado a recepção de um LIMPAQU causa uma transição para o estado de "Pedido de Desconexão". Neste estado a interface aguarda a chegada de um "Clear Confirmation" para poder liberar o circuito virtual. Estando o canal lógico no estado de "Transferência de dados", a recepção de um "Clear" vindo do conversor faz com que o canal passe para o estado "Livre".

Durante a fase de transferência de dados os fragmentos (unidades de informação do nível 4) serão enviados para o lado remoto via conversor. Acontece que existem certos caracteres ASCII, que normalmente são usados como caracteres de controle pelo conversor. Por exemplo o caracter "CR" (Carriage Return) quando enviado após um conjunto de dados, pode servir como caracter de finalização do pacote, fazendo com que o conversor o envie para o ETD remoto. Ora, se o "CR" fizesse parte dos dados do nível de transporte haveria uma divisão do fragmento em dois pacotes distintos, o que não é desejável pois além de perdermos a informação correspondente ao "CR", o ETD remoto não teria condições de determinar que o conteúdo do segundo pacote não corresponde a um novo fragmento e sim à continuação do anterior.

O fato acima está relacionado com o problema da transparência na fase de transferência de dados. A utilização do "CR" como caracter de finalização do pacote é feita através do parâmetro 3, quando este é igual a 2. A idéia é fazer a escolha de um perfil

transparente. Por exemplo se fizermos o parâmetro 3 igual a 0, a montagem do pacote será feita de acordo com outros eventos (temporização ou tamanho máximo do pacote). A transmissão de um pacote após o estouro da temporização ou ao completar o tamanho máximo do pacote ainda pode provocar uma divisão do fragmento ou o agrupamento de fragmentos distintos. Se quisermos assegurar que cada pacote contenha apenas um fragmento devemos manter a existência de um carácter de controle para provocar o término da montagem do pacote. Neste caso um modo de se garantir a transparência seria codificar cada octeto em dois caracteres ASCII. A desvantagem principal deste esquema consiste na duplicação do número de octetos enviados.

Outra forma de garantir a transparência seria não utilizarmos nenhum carácter para indicar o término do pacote. Neste caso deveríamos colocar antes do primeiro octeto de cada fragmento um outro octeto indicando o comprimento do mesmo, e utilizaríamos o temporizador para comandar o envio de pacotes. A transição da fase de transferência de dados para a de comando seria feita através do sinal de "break" (parâmetro 7 igual a 8).

IV. IMPLEMENTAÇÃO DA INTERFACE X.2B NO DEC-10

A interface entre o conversor e o protocolo de transporte foi implementada no sistema DEC - 1091 da Universidade Federal de Pernambuco.

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 512 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.

Para que não houvesse a possibilidade de perda de caracteres na recepção, foi usado na implementação um sistema que gera interrupções toda vez que chega caracteres na interface. O tratamento da interrupção é feito recuperando o caracter do buffer da interface.

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 sistema de interrupções forma um conjunto de rotinas escritas em MACRO-10 (assembler do DEC-10), especialmente para serem usadas por programas escritos em Pascal. O software do IPCF-10 também está disponível para ser usado através de uma linguagem de alto nível (Ex: Pascal, Fortran, Cobol, etc.).

Após concluída a implementação, alguns testes isolados foram realizados. Foi testada a interação entre a interface X.28 e o protocolo de transporte para verificar o funcionamento do IPCF-10. Foi testada em diversos cenários o funcionamento da interface com relação ao protocolo de transporte, isto é, colocamos uma pessoa simulando o nível de aplicação em um terminal de vídeo e outra simulando o conversor em outro terminal de vídeo. O nível de aplicação requisitava a abertura de uma conexão forçando o nível 4 à enviar um "ESTABCV" para a interface, e ao mesmo tempo o conversor mandando prompts e respostas aos comandos enviados. Desta forma constatou-se que a interface funcionava satisfatoriamente. Restam portanto os testes finais que dependem da entrega do conversor X.25. Uma facilidade que foi implementada foi a de fazer com que a interface gravasse em um arquivo de ocorrências os eventos inesperados, acontecidos em uma determinada conexão, de forma que na fase de testes possamos analisar com mais detalhes o funcionamento do protocolo.

V. CONCLUSÃO

Este artigo retrata a experiência de implementação da interface entre um computador hospedeiro e um conversor *start/stop - X.25 no DEC-10 da UFPe. O trabalho encontra-se praticamente concluído, restando apenas os testes finais com o uso do conversor, que até a data de elaboração deste artigo ainda não havia sido colocado à disposição da UFPe. Este projeto faz parte de um estudo mais amplo que trata das alternativas de se conectar computadores à uma rede pública de pacotes.

Gostaríamos de agradecer ao pessoal do NPD (Núcleo de Processamento de Dados) da UFPe, pelo apoio prestado durante a fase de desenvolvimento do projeto, como também à Embratel e à Secretaria Especial de Informática (SEI) pelo apoio financeiro, que nos incentivou no sentido de concluirmos nosso trabalho.

VI. REFERENCIAS

- [CCIT 77] CCITT RECOMENDATION X.3 - Packet Assembly/Disassembly Facility (PAD) in a Public Data Network, Genebra, 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, Genebra, 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, Genebra, 1980.
- [CUNH 84] CUNHA, P.R.F., MONTEIRO, J.A.S., LUCENA, E.B. - Implemen-

tação de um Protocolo de Transporte para a Rede CEPINNE, submetido ao 3o. Simpósio Brasileiro de Redes de Computadores, Janeiro, 1985.

- [DEAT 83] DEATON Jr., G.A. - HIPPERT, Jr., R.O. - X.25 and Related Recommendations in IBM products, IBM Systems Journal, 22 (1/2): 11-29, 1983.
- [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.
- [MONT 84] MONTEIRO, J.A.S., JUREMA, M.A. - LUNHA, P.R.F. - Comporta para Conexão de um Computador DEC-10 à uma Rede Pública de comunicação de Dados, in: 2. Simpósio Brasileiro sobre Redes de Computadores, Anais, Campina Grande, 1984, p. 2.1 - 2.21.