Um Servidor de Processamento Paralelo Baseado em Transputers - Requisitos e Definição

Onofre Trindade Jr*
Marcos José Santana†

RESUMO

Neste trabalho definem-se os requisitos e aspectos de hardware e software de um servidor de processamento paralelo operando em um ambiente distribuído baseado em rede local de computadores. O servidor dispõe de uma certa quantidade de processadores interligados por chaves reconfiguráveis, possibilitando que cada usuário, a partir de sua estação de trabalho, possa requisitar a alocação de processadores, interligados numa topologia adequada para sua aplicação.

São discutidas e analisadas diversas alternativas na definição da arquitetura do servidor, bem como alguns detalhes de implementação.

Na configuração máxima, o servidor proposto pode atingir uma potência computacional de pico de 28.8 GIPS e 4.1 GFLOPS, permitindo que esse seja incluído entre os sistemas denominados supercomputadores.

ABSTRACT

This paper describes the requirements and general aspects of both hardware and software for a parallel processing server operating in a LAN-based distributed computing environment.

The proposed server is composed of a number of processors interconnected to each other by means of reconfigurable switches, allowing a system user to request (from his or her workstation) some of the processors to build up a suitable topology for the application involved.

Several architecture alternatives and implementation issues are discussed. At the top configuration the proposed server can reach a supercomputer peak power of about 28.8 GIPS and 4.1 GFLOPS.

*Mestre (IFQSC USP) – Áreas de Interesse: Arquiteturas Paralelas; Programação Concorrente – End: ICMSC USP - Cx Postal 668, CEP 13560 - São Carlos - SP – E-mail OTJUNIOR@BRUSP.BITNET

†Mestre (ICMSC USP), PhD (Southampton University, UK) – Áreas de Interesse: Sistemas Computacionais Distribuídos; Simulação; Programação Concorrente – End: ICMSC USP - Cx Postal 668, CEP 13560 - São Carlos - SP – E-mail MJSANTAN@BRUSP.BITNET

REDES LOCAIS DE COMPUTADORES

Redes locais de computadores (LANs) [CL78, HO86, ST84, CU88] compreendem um grupo especial de redes de computadores que teve desenvolvimento acelerado nas duas últimas décadas, dando margem ao aparecimento de diversas aplicações, principalmente na definição de sistemas computacionais distribuídos [TA85, CO88, MU89] baseados em redes locais. Esses sistemas apresentam inúmeras vantagens sobre as arquiteturas tradicionais que utilizam multiprogramação, destacando-se maior flexibilidade, confiabilidade, desempenho, vida útil, adaptabilidade e facilidade na contenção do custo de implementação e manutenção em um sistema computacional.

O uso de redes locais para a implementação de sistemas distribuídos, introduziu a possibilidade de se oferecer recursos especializados e de alto custo, de uma maneira relativamente econômica a uma população de usuários elevada e distribuída geograficamente.

PROCESSAMENTO PARALELO E TRANSPUTERS

Apesar do crescente aumento no poder computacional das máquinas sequenciais (arquitetura de von Neumann), restrições de ordem física têm imposto limites máximos que estão próximos de serem atingidos. O processamento paralelo representa uma alternativa importante na solução desse problema, favorecendo a construção de máquinas com elevado desempenho. Embora diversas arquiteturas dedicadas ao processamento paralelo tenham sido propostas ao longo dos últimos anos [HW84, AL89], o aparecimento do transputer da INMOS [IN89a] e de sua linguagem concorrente OCCAM [JO88, WE89] vieram dar novo impulso nessa área, permitindo a obtenção de alto desempenho e confiabilidade por meio de hardware e software flexíveis e de custo relativamente baixo.

O TRANSPUTER INMOS T800

O transputer, um microprocessador fabricado pela INMOS (SGS- THOMSON), possui algumas características que permitem a implementação de sistemas para processamento paralelo de uma maneira bastante simples. A nível de hardware, ele é composto por um processador de inteiros (30 MIPS), um processador de ponto flutuante (4.3 MFLOPS), 4 Kbytes de memória RAM interna e 4 canais de comunicação serial (20 Mb/s) [IN89a]. Os canais do transputer implementam um protocolo de comunicação assíncrona com "handshake" por hardware. Esses canais permitem a interligação de diversos transputers, definindo redes de processadores para computação paralela. A figura 1 exemplifica uma rede de 4 transputers formando um "pipeline". A facilidade em se alterar as interligações permite que o usuário implemente a topologia mais adequada para sua aplicação. Chaves programáveis podem tornar essa operação ainda mais fácil, bem como possibilitar a reconfiguração dinâmica das interligações. Para a inicialização, detecção de erros e depuração destes através da análise do conteúdo da memória e registradores do transputer, foram previstos três sinais, respectivamente RESET, ERROR e ANALYSE (RAE).

Juntamente com o desenvolvimento do hardware do transputer foi definida a linguagem OCCAM. Essa linguagem de alto nível suporta as características inovadoras do transputer, e posssibilita o desenvolvimento amigável de programas paralelos. Os canais físicos de comunicação do transputer são vistos na linguagem como um tipo especial de variável denominado *CHAN*. Estas variáveis permitem a comunicação entre processos concorrentes. A implementação de processos concorrentes em um único transputer é feita por tempo compartilhado. Os canais, nesse caso, são implementados como variáveis na memória interna do transputer. Quando uma rede de transputers está disponível, as

variáveis tipo canal podem ser associadas aos canais físicos e cada transputer executará um conjunto de processos que podem se comunicar com outros através desses canais, permitindo a exploração de processamento paralelo de média granularidade.

Existem compiladores das principais linguagens de alto nível para transputers. Mesmo usando linguagens convencionais, pode-se explorar concorrência utilizando-se a lingugem OCCAM para aglutinar os módulos escritos nessas linguagens. Isso facilita o transporte e adaptação de sistemas existentes para os transputers e processamento paralelo. Para a utilização de todos os recursos do transputer e máximo desempenho, deve-se programar na linguagem OCCAM. A INMOS fornece um sistema de desenvolvimento com editor, compilador OCCAM e depurador denominado TDS (Transputer Development System) [IN88]. Algumas outras linguagens de alto nível compatíveis com o TDS, tais como FORTRAN, C e ADA, também estão disponíveis. O TDS pode ser utilizado com um computador hospedeiro fazendo o papel de servidor de arquivos, teclado/display e impressão, conforme mostra a figura 2.

REQUISITOS DO SERVIDOR

A adoção de bancos de processadores tem sido sugerida como uma solução eficaz para a alocação de recursos a uma vasta população de usuários, a um custo efetivo [BA89]. Neste trabalho essa idéia é extendida, possibilitando aos usuários de um sistema distribuído baseado em rede local de computadores, recursos para processamento paralelo. O ambiente de um sistema distribuído desse tipo pode ser visto, de uma maneira sumária, como um grupo de estações de trabalho compartilhando recursos e informações providos por servidores especializados (modelo de estações de trabalho/servidores [TA85, CO88]) (fig 3).

O desenvolvimento e implementação de um servidor especializado em processamento paralelo baseado em Transputers (Banco de Transputers), deve ser o mais transparente possível, do ponto de vista do usuário, permitindo que os recursos disponíveis sejam utilizados facilmente, através de um conjunto coeso de comandos e ferramentas que escondam os detalhes de implementação, mapeamento físico e existência de múltiplos recursos. O usuário deve ter a ilusão de estar utilizando todos os recursos como se esses estivessem conectados direta e exclusivamente à sua estação de trabalho. Nesse sentido, o usuário pode requisitar ao servidor um determinado número de processadores (transputers), especificando a topologia da interconexão entre eles. O servidor analisa a solicitação baseando-se nos recursos ainda disponíveis, atendendo-a ou não em função dessa análise, permitindo ainda que o usuário permaneça numa fila aguardando a liberação dos recursos solicitados. Uma vez alocados os recursos, o usuário pode utilizar ferramentas e linguagens para o desenvolvimento de programas paralelos, tal como o TDS.

ARQUITETURA DO SERVIDOR

Em linhas gerais o servidor dispõe de um sistema computacional acoplado à rede, responsá-vel por toda a comunicação com o meio exterior, agindo como um processador de entrada (FEP Front-End Processor) (fig 3). Esse sistema computacional é responsável pela implementação de um sistema coordenador que gerencia todo o acesso ao Banco de Transputers. O Banco de Transputers, por sua vez, conta com um conjunto de placas processadoras, baseadas no Transputer T800 da INMOS [IN89a], com memória local (fig 4). Um sistema de chaves eletrônicas especializado é utilizado para promover a configuração das interconexões dos diversos Transputers requisitados para uma determinada tarefa.

Diversos usuários podem ter acesso ao Banco de Transputers concomitantemente,

sendo tarefa do coordenador manter canais virtuais entre as estações de trabalho e os respectivos lotes de transputers. Sob o ponto de vista do usuário, um lote de transputers representa um determinado número de transputers conectados numa topologia requisitada. O acesso a esses tranputers é feito por um canal ("link") de carga ("boot") e pelos sinais RESET, ERROR e ANALYSE, conforme sugere a figura 5. Essa condição é análoga à especificada para a utilização do sistema de desenvolvimento TDS.

Uma primeira tentativa de definição da configuração do banco leva ao esquema apresentado na figura 6. Nessa figura, uma chave "crossbar" é utilizada para prover a reconfiguração das interligações entre os processadores. Até um total de U usuários podem compartilhar N processadores através de uma chave com 4N+U canais.

A definição do número máximo de usuários do servidor (Nu) pode, numa aproximação inicial, ser baseada no número máximo de estações de trabalho (Net) e no número máximo de processadores (Np). A seguinte relação deve ser satisfeita:

$$Nu \leq MIN(Net; Np)$$

Procedendo-se dessa forma, corre-se um risco elevado de um super dimensionamento no número de usuários. Outros fatores, tais como número máximo de usuários do servidor e o número médio de processadores alocados por usuário, poderiam ser utilizados se seus valores numéricos fossem previamente conhecidos. Esses fatores, entretanto, dependem fortemente do ambiente de utilização do servidor. Pretende-se estudá-los no âmbito do SCE-ICMSC-USP a partir da coleta de dados reais de utilização do servidor, e posterior análise estatística e/ou simulação.

Devido às dificuldades acima apresentadas para a definição do número máximo de usuários, adotou-se a seguinte relação:

$$Nu = Npi/4$$

onde:

Npi --> número de processadores instalados

Apesar de heurística, esta relação introduz modularidade na definição do número de usuários. A análise de dados reais de utilização deverá permitir o julgamento da conveniência dessa decisão.

A modularização e expansibilidade da configuração 1, apresentada na figura 6, ficam prejudicadas pela adoção de uma única chave. Apesar dessa chave poder ser implementada a partir de outras com menor número de canais [IN89b], são exigidas muitas alterações a nível de interligações físicas das chaves para que o número de canais disponíveis seja aumentado. A implementação total da chave, num servidor com poucos processadores instalados em relação ao número máximo possível, pode tornar a chave várias vezes mais cara que todos os seus demais componentes. A figura 7 mostra uma outra configuração onde são considerados os aspectos discutidos sobre a modularização e expansibilidade do servidor. Considerações feitas ao nível de implementação, mostram entretanto que essa solução apresenta uma péssima relação entre o custo das chaves e o custo total do sistema.

Levando-se em consideração a similaridade entre os quatro canais de cada transputer, pode-se obter conectividade total em um grupo deles, utilizando-se uma chave mais simples [NI88, IN89b]. A figura 8 ilustra o esquema básico proposto. Nessa figura, cada chave indicada deve possibilitar a interligação de dois canais quaisquer em lados opostos. Não é

necessária, entretanto, a interligação de canais de um mesmo lado da chave, o que simplifica seu projeto em relação a uma chave "crossbar" convencional. Deve-se observar que a perda da identidade dos canais imposta por essa configuração deve ser compensada por um esforço adicional na implementação do software, como será discutido posteriormente. Um servidor baseado na configuração apresentada na figura 8 pode ser extendido hierarquicamente, como sugere a figura 9.

A próxima etapa da definição da arquitetura do servidor consiste em definir os meios para acesso dos sinais RESET, ERROR e ANALYSE dos processadores. Entre as diversas possibilidades existentes, figuram a utilização de "multiplexers/demultiplexers", diversos tipos de chaves e portas de E/S de controladores. No ítem seguinte é apresentada a solução proposta, a qual se baseia em um barramento controlado por um processador.

ASPECTOS DA IMPLEMENTAÇÃO DO HARDWARE

Na implementação do Banco de transputers foi adotada a configuração 3 apresentada nas figuras 8 e 9. O motivo dessa escolha foi fortemente calcado nos aspectos de modularização e expansibilidade discutidos no ítem anterior.

Na implementação das chaves está sendo utilizado o circuito integrado INMOS IMS C004 [IN89a]. Desenvolvido especificamente para a interligação de canais de transputers, esse CI é constituido por uma chave "crossbar" com 32 canais e é programável via software por um canal especial. Ele substitui várias dezenas de circuitos integrados MSI convencionais e provê lógica específica para a regeneração dos sinais que fluem através dele. Isso mantém a integridade dos sinais mesmo que vários CIs sejam ligados em cascata. A hierarquia do banco foi limitada em três níveis, baseada nos seguintes componentes:

processador – um dos processadores do banco. Consiste basicamente em um transputer T800 com 1 a 4 MBytes de memória RAM

módulo – conjunto de 1 a 30 processadores alojados em um bastidor padrão eurocard. O módulo consiste na implementação física do esquema proposto na figura 8

banco – conjunto de até 64 módulos. O banco corresponde à implementação física do esquema proposto na figura 9

Observe que um módulo pode ser utilizado isoladamente com até 30 processadores, uma vez que neste caso são destinados 8 canais para comunicação com os usuários. Se utilizado dentro de um banco, o módulo terá no máximo 15 processadores. Os canais correspondentes aos outros 15 processadores serão utilizados na comunicação do módulo com a chave do banco.

Considerando a utilização da chave IMS C004, a configuração adotada (configuração 3) apresenta os seguintes números máximos:

- até 64 módulos por banco
- até 256 usuários
- até 960 processadores

Para a implementação total dessa configuração são necessárias 640 chaves IMS C004. Para efeito de comparação, um sistema equivalente baseado na configuração 1 mostrada

na figura 6, apresenta os seguintes números:

> até 256 usuários
> até 960 processadores
> 1152 chaves IMS C004

E um baseado na configuração 2, mostrada na figura 7:

> até 64 módulos por banco
> até 256 usuários
> até 960 processadores
> 1920 chaves IMS C004

Esses números permitem calcular os seguintes valores para a relação Nc/Np (número de chaves/número de processadores):

conf 1	conf 2	conf 3
1.2	2.0	0.67

Se o número de processadores for restringido a 240, o que minimiza a relação Nc/Np para a configuração da figura 7, esses valores se tornam:

conf 1	conf 2	conf 3
0.4	1.2	0.67

Observa-se que a configuração adotada exibe uma boa relação Nc/Np. Além disso, essa relação permanece constante até o número máximo considerado de 960 processadores. A configuração 1, conforme mencionado, foi desconsiderada pela pequena modularidade e expansibilidade que apresenta. Se entretanto, o limite de até 240 processadores for razoável, torna-se uma boa opção, principalmente se o banco for definido com um número fixo de processadores. Para o acesso aos sinais RAE de cada lote de transputers, optou-se por um barramento em cada módulo, com as seguintes características:

- endereçamento de 32 processadores (normalmente 15 ou 30 são implementados)
- endereçamento de 16 usuários (normalmente 4 ou 8 são implementados)
- linha especifica para o sinal /ERROR dos processadores do módulo

O barramento em cada módulo é interligado com os de outros módulos através de "buffers". Existe no barramento um sinal de seleção de módulo que permite validar os demais sinais do mesmo. Um processador controla o barramento descrito, as chaves do módulo e as chaves do banco (caso o servidor contenha vários módulos). O diagrama de blocos desse controlador é mostrado na figura 10. Cada processador ligado ao barramento dispõe de um registrador de 4 bits que permite o armazenamento de um número correspondente ao lote (usuário) ao qual ele foi alocado. Esse número, comparado com as linhas de dados do barramento definido, permite que os sinais RAE sejam dirigidos para um lote em particular.

ASPECTOS DA IMPLEMENTAÇÃO DO SOFTWARE

O software básico, necessário para a operação do servidor, tem como requisito principal permitir a utilização do sistema de desenvolvimento TDS no ambiente descrito. Isso deve acontecer da forma mais transparente possível sob o ponto de vista do usuário final do sistema.

Outro requisito importante consiste em tornar o software do servidor o menos dependente possível do sistema operacional (SO) da rede local. Feito isso, o transporte desse

software será facilitado para quaisquer máquinas que possam executar o TDS e disponham de uma interface ETHERNET.

Levando-se em consideração os requisitos mencionados, optou-se por utilizar o meio físico da rede local concorrentemente com o SO. Isso equivale a tornar o mecanismo de transmissão de mensagens entre as estações de trabalho e o servidor de processamento paralelo dependente unicamente do padrão ETHERNET. A figura 11 delineia os aspectos mencionados. Note-se também que essa opção minimiza as alterações necessárias no sistema de desenvolvimento TDS.

Além do software básico para o funcionamento do servidor, alguns programas de suporte são desejáveis. O primeiro deles deve introduzir no programa fonte do usuário o mapeamento físico das interligações alocadas cada vez que o usuário requisite um certo número de transputers interconectados numa topologia definida. Isso eliminará a dificuldade introduzida com a adoção de uma chave que, apesar de permitir conectividade total, não mantém a identidade dos canais de cada transputer. Outros programas de suporte devem permitir a coleta e avaliação de dados sobre a utilização do servidor. Conforme mencionado, esses dados serão de grande valia na avaliação das decisões tomadas durante as fases de definição e projeto do servidor.

ASPECTOS DA UTILIZAÇÃO DO SERVIDOR

A potência computacional disponível no banco, caso os 960 processadores sejam implementados com transputers T800 @ 30 MHZ, atinge 28.8 GIPS e 4.1 GFLOPS. Essa potência computacional coloca o servidor proposto na classe dos denominados supercomputadores.

Deve-se observar que, como na maioria dessas máquinas, essa potência computacional de pico somente é atingida em poucos casos particulares. Além disso, em máquinas com vários processadores, como o servidor proposto, a potência mencionada somente será atingida se o problema a ser resolvido tiver um nível de paralelismo compatível com o número de processadores disponíveis. Esses aspectos abrem novas possibilidades de pesquisas relacionadas com o estudo de soluções e algoritmos otimizados para a arquitetura proposta.

CONCLUSÕES

Este artigo teve como objetivo a divulgação das idéias básicas que definiram o desenvolvimento de um servidor de processamento paralelo no SCE ICMSC USP, São Carlos (em andamento). A análise inicial mostra que a potência computacional alcançada, e os custos atrativamente baixos, tornam o servidor uma alternativa viável.

Diversos aspectos ainda estão em aberto, e suas definições finais, dependem das etapas de implementação e reavaliação do projeto. Pretende-se ter uma versão inicial do sistema descrito no início de 1991, o que determinará uma nova fase na pesquisa, permitindo que muitas das decisões tomadas possam ser melhor avaliadas.

O ambiente baseado em rede local e o servidor descrito fornecerão subsídios para o desenvolvimento de inúmeros trabalhos relacionados com computação paralela e sistemas distribuídos baseados em redes locais de computadores.

Na data em que este artigo foi escrito, iniciava-se a implementação do hardware e o projeto e definição do software do servidor.

AGRADECIMENTOS

Os autores, em nome do grupo de Hardware e Software Básico do SCE ICMSC USP, agradecem o apoio recebido da FAPESP para o desenvolvimento do trabalho descrito neste artigo, conforme processo 90/1445-2.

Referências

[AL89] Almasi, G.S., Gottlieb, A., Highly Parallel Computing, Benjamin/Cummings Publishing Company, Inc, USA 1989.

[BA89] Bacon, J.M., Leslie, I.M. and Needham, R.M., Distributed Computing With a Processor Bank, Thechnical Report, Computer Laboratory, Cambridge University, U.K., abril 1989.

[CL78] Clark, D.D., Pogran, K.T. e Reed, D.P., An Introduction to Local Area Networks, Proceedings of the IEEE, 66(11), pp.1497-1517, Nov.1978.

[CO88] Coulouris, G.F. e Dollimore, J., Distributed Systems - Concepts and Design, Addison-Wesley Publishers Limited,England 1988.

[CU88] Currie, W.S., LANs Explained - A Guide to Local Area Networks, Ellis Horwood Limited, England, 1988.

[HO86] Hopper, A., Temple, S. and Williamson, R., Local Area Network Design, Addison-Wesley Publishing Company, London, 1986.

[HW84] Hwang, K. and Briggs, F.A., Computer Architecture and Parallel Processing, McGraw-Hill, 1984.

[IN88] INMOS Limited, Transputer Development System, Prentice Hall, UK, 1988.

[IN89a] INMOS Limited, The Transputer Databook, INMOS Databook Series, INMOS document number 72 TRN 203 00, 989.

[IN89b] INMOS Limited, The Transputer Applications Notebook - Systems and Performance, INMOS Databook Series, INMOS document number 72 TRN 205 00, 1989.

[JO88] Jones, G. and Goldsmith, M., Programming in Occam2, Prentice Hall International Series in Computer Science, 1988.

[MU89] Mullender, S.J. (ed), Distributed Systems, ACM Press Frontier Series, Addison-Wesley, 1989.

[NI88] Nicole, D. A., Esprit Project 1085 - Reconfigurable Transputer Processor Architecture, private communication, Southampton University, UK, 1989.

[ST84] Stallings, W., Local Networks, ACM Computer Surveys, 17(1), pp. 3-41, Mar. 1984.

[TA85] Tanenbaum, A.S. and van Renesse, R., Distributed Operating Systems, ACM Computing Surveys, 17(4), pp. 419-470, Dez. 1985.

[WE89] Wexler, J., Concurrent Programming in Occam2, Ellis Horwood Limited, 1989.

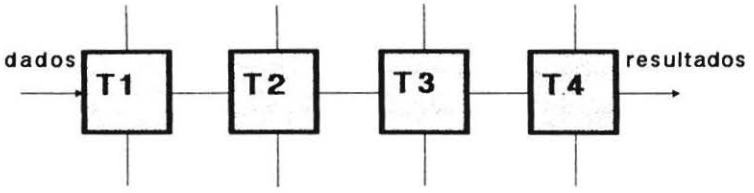

fig 1 "Pipeline" constituído por uma rede com 4 Transputers

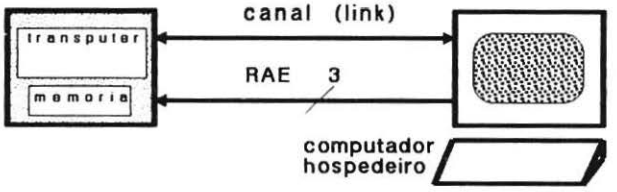

fig 2 Ambiente de utilização do TDS

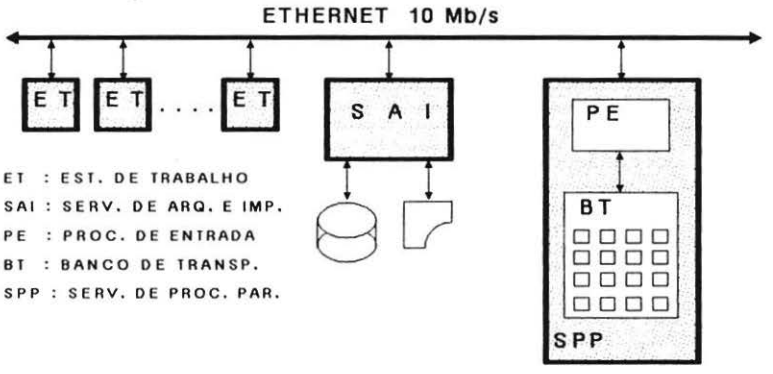

fig 3 Estações de trabalho e servidores em uma rede local

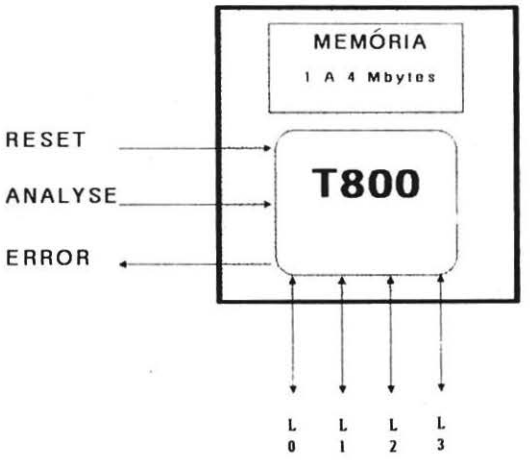

fig 4 Processador baseado no transputer T800

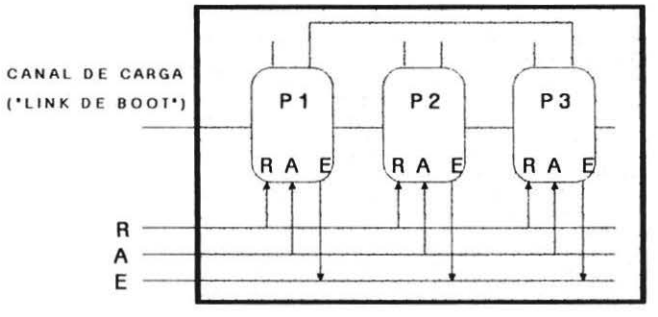

fig 5 Aspecto de um lote de transputers sob o ponto de vista do usuário

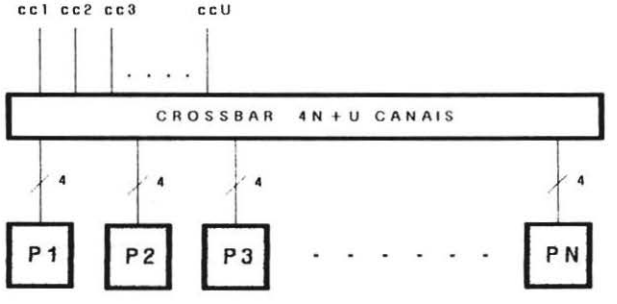

fig 6 Banco de Transputers - Configuração 1

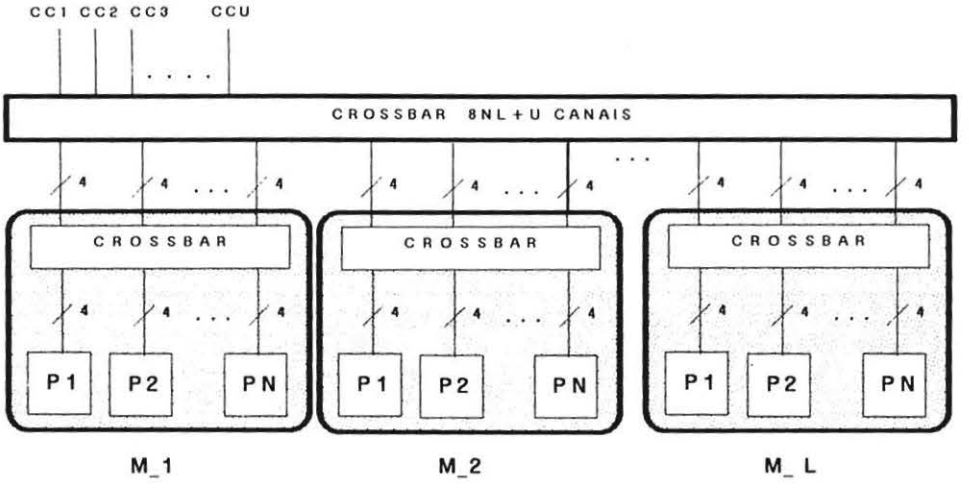

fig 7 Banco de Transputers - Configuração 2

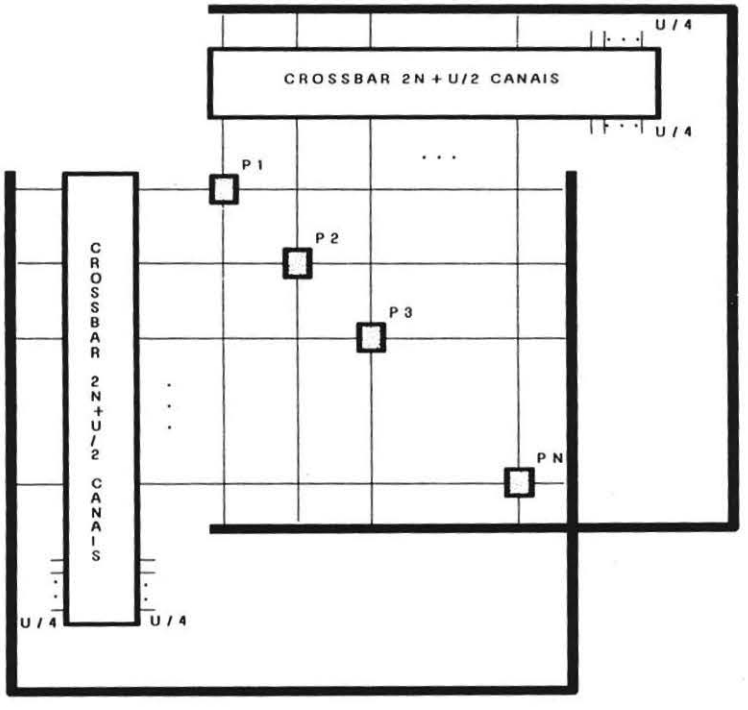

fig 8 Configuração 3 - Conectividade Total Entre N Processadores
Com Perda da Identidade Dos Canais

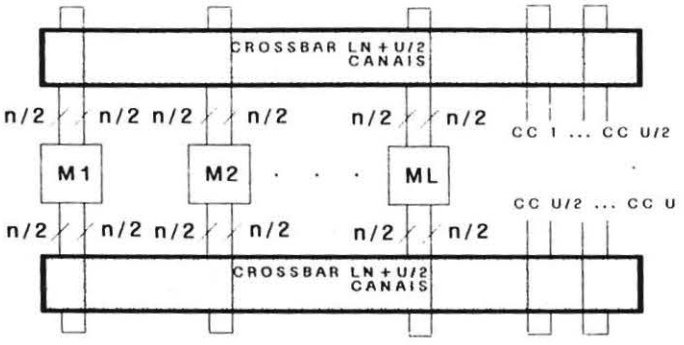

fig 9 Banco de Transputers Constituído por Módulos Baseados na Configuração 3

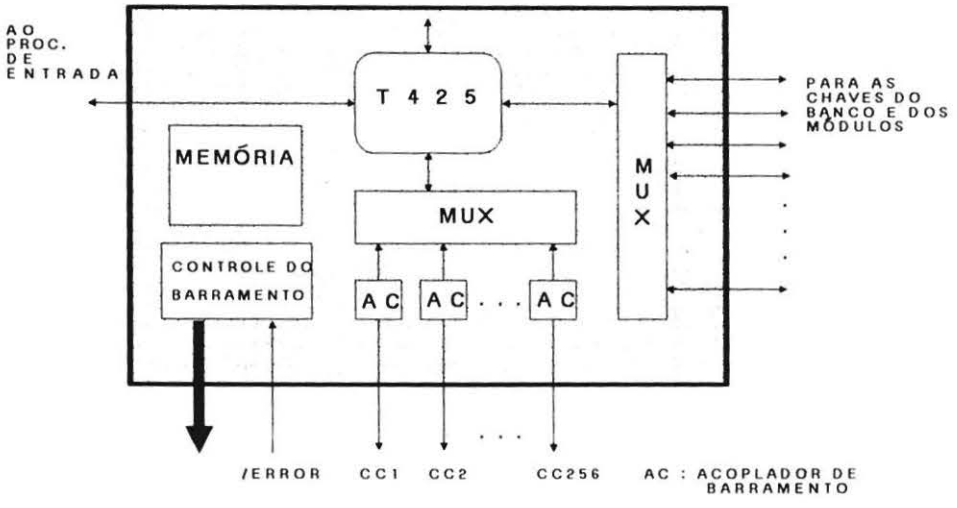

fig 10 Diagrama de Blocos do Controlador de Módulo/Banco

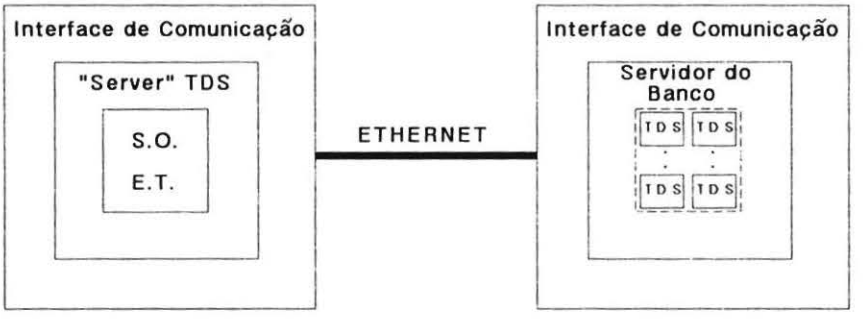

fig 11 Organização do Software Básico do Sistema