PROCESSADOR PARALELO P3

A. Pestana, E. Cavalli
CPqD - TELEBRÁS
Caixa Postal 1179, 13085 - Campinas - SP

RESUMO

Descrição de um sistema para processamento paralelo de alto desempenho (até 500Mflops) baseado na placa de processamento PP-U32 e no Sistema Operacional PP-SO/P desenvolvidos no CPqD Telebrás. O sistema vale-se da tecnologia mais avançada dos microprocessadores de 32 bits (80386 e Transputer) e de um sofisticado Sistema Operacional desenvolvido especificamente para aplicações de processamento distribuído em tempo real.

ABSTRACT

Description of a high performance parallel processing system (up to 500Mflops) based on computer board PP-U32 and on operating system PP-SO/P developped by CPqD - Telebrás. The system utilizes advanced 32 bits microprocessor technology (80386 and Transputer) and a sophisticated operating system especifically developped for real-time distributed processing applications.

1. INTRODUÇÃO

Este artigo apresenta, de forma resumida, um sistema para processamento paralelo com capacidade de processamento de até 500 Mflops e 2700 Mips.

Apesar dos componentes do sistema permitirem arquiteturas muito diferentes em tamanho e tipo de interconexão somente uma estrutura será representada. Em particular nenhuma considereção será feita sobre estruturas de processamento paralelo por memória comum, não por considerá-las impossíveis ou incompatíveis com os componentes básicos do sistema descrito, mas por simples auto limitação originada pela complexidade e extensão do problema. A unidade de processamento utilizada no sistema possue características que permitem um bom aproveitamento em arquiteturas de processamento paralelo do tipo **memória comum**. Estas arquiteturas poderão ser objeto de estudos posteriores.

Sobre as vantagens de se utilizar arquiteturas de processamento baseadas em múltiplos microprocessadores existe uma literatura extensa [?], [?], [?], rica também com relação as estruturas de conexão entre processadores [?], [?], [?].

O objetivo principal do artigo é descrever um sistema de processamento paralelo de alto desempenho e com um nível tecnológico atual, a ser utilizado em aplicações divisíveis em partes fracamente ou mediamente acopladas.

Uma primeira fase do sistema pode ser realizada, com tecnologia de projeto e construção inteiramente nacional, em um prazo inferior a dois anos considerando o estado avançado em que se encontra o desenvolvimento do hardware e a possibilidade de se utilizar, nesta primeira fase o Sistema Operacional PP-SO/P [?] com poucos acréscimos. Este prazo está relacionado com a atividade de adaptação de compiladores (C, Fortran), com o software básico para utilização do transputer e com a implementação de bibliotecas.

2. ESTRUTURA HARDWARE DO SISTEMA

O sistema é constituído de 16 **módulos de processamento** e um **módulo de gerência** (figura 1).

Os módulos de processamento são todos iguais e cada um deles contêm oito placas processadoras (PP-U32) e a fonte de alimentação.

Uma gaveta de 19 polegadas com altura de cerca de 26cm pode conter um módulo.

O módulo de gerência contém 8 placas processadoras U32 e as placas (3) controladoras dos periféricos básicos do sistema.

Teremos um total de 17 gavetas para os módulos de processamento (incluindo o módulo de gerência), uma para as fontes do módulo de gerência e dos periféricos e duas para acomodar os periféricos (discos flexíveis, discos rígidos e eventualmente fita magnética).

Estas 20 gavetas podem ser acomodadas em 4 bastidores de 19" com uma altura de cerca de 180 cm considerando a presença de ventiladores e planos defletores.

Todos os 17 módulos do sistema são interligados em dupla malha completa implementada via conexões seriais bidirecionais a 10-20 Mbits/seg controladas pelo transputer T800.

Cada módulo tem duas conexões diretas com cada um dos demais 16 módulos.

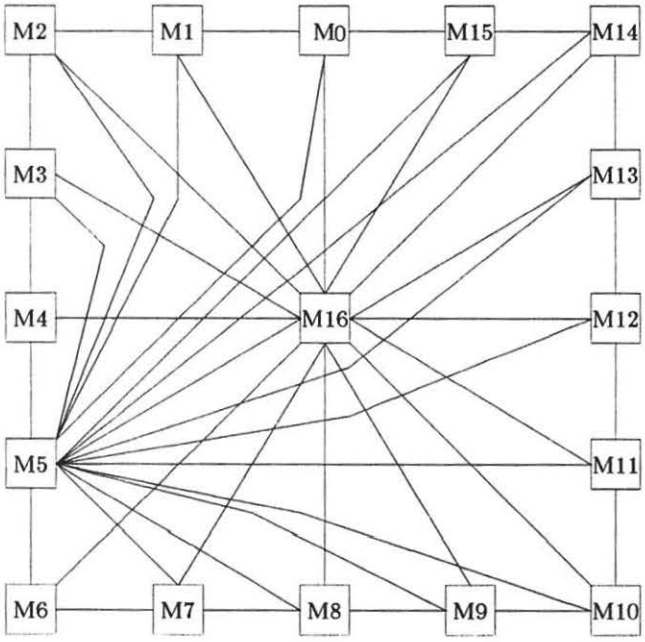

Figura 1: *Sistema S17M8*

No desenho da figura 1 os módulos de M00 a M15 são os de processamento e o módulo M16 é o de gerência. Na figura cada linha de interconexão representa duas vias bidirecionais de 10-20 Mbits/seg. São visualizadas todas as ligações do M16 e do M05.

2. 1. Estrutura do Módulo

A figura 2 apresenta a estrutura lógica do módulo de processamento que é constituído de 8 unidades U32 interligadas no barramento de 32 bits de alta velocidade.

Cada unidade U32 consegue acessar, além da própria memória interna, toda a memória de cada uma das demais unidades U32.

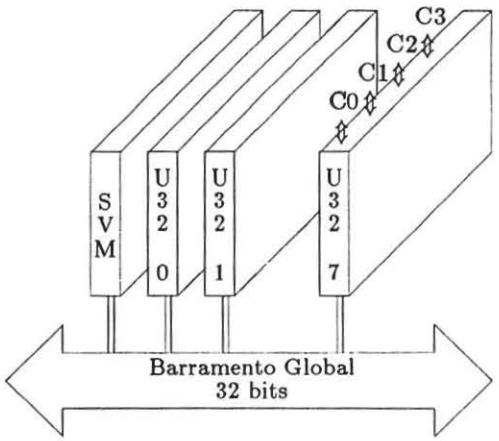

Figura 2: *Módulo de processamento*

O módulo possue 32 vias de comunicação bidirecionais de 10-20 Mbits/seg, 4 por unidade U32, utilizadas para as interconexões com os demais 16 módulos.

O acesso à memória das outras unidades é executada utilizando faixas de memória pré-definidas via barramento global.

Cada placa tem micro-chaves que definem o endereço da sua memória dinâmica para o acesso do lado do barramento (são definidas as linhas mais significativas A24-31 dos endereços que no barramento operam como identificadores de unidade).

A velocidade efetiva de comunicação entre as 8 unidades é maior que 12 Mbytes/s sendo suficiente para evitar, em grande parte das aplicações, a limitação de desempenho gerada pela contenção. Uma análise técnica que nos levou à escolha desta configuração do módulo encontra-se no **Sistema hardware para processamento paralelo** [8].

Além das unidades de processamento e da fonte de alimentação o módulo pode conter opcionalmente uma placa de supervisão de módulo cujas funções principais são:

- arbitração paralela do barramento (a arbitração normal das unidades U32 é do tipo daisy-chain com prioridade circular).
- memória EPROM comum a todas as placas para reduzir os custos e facilitar a manutenção.
- supervisão do hardware do barramento para facilitar a localização das falhas.
- suporte para análise de desempenho, contenção e comutação de mestre.

2. 1. 1. Unidade de Processamento PP-U32

As principais características da unidade de processamento U32, na sua configuração máxima, são:

- Microprocessador
  - 80386 de 25 MHz (5 Mips)
  - 80387 (0,85 Mflops) e WTL1167 (1.1 Mflops)
  - 128 Kbytes de memória cache com transferência de blocos
- Memória dinâmica
  - 8 Mbytes (integrados de 1M x 1 bits)
  - até 16 Mbytes com integrados de 4M x 1 bits
  - correção de erro simples, deteção de erro duplo
- Lógica de DMA
  - 8 canais de DMA de 8-16-32 bits implementados pelo integrado 82380.
  - Opera sem bloquear a operação do 80386 e T800.
- Módulo de comunicação
  - transputer T800 (15 Mips, 2,25 Mflops, 4 Kbytes de memória interna)
  - 128 Kbytes de memória cache
  - 128 Kbytes de memória local
  - 4 canais bidirecionais assíncronos de 10-20 Mbits/seg.
- Outras funções:
  - 7 temporizadores
  - 20 linhas de interrupção
  - 2 canais seriais RS-232
  - 128/512K EPROM
  - Lógica para tratamento das condições de falha

A figura 3 mostra o esquema lógico da U32. A memória dinâmica (DRAM) é acessada pelo 80386, pelo 82380, pelo T800 e pelo mestre do barramento global, isto é, outra placa U32.

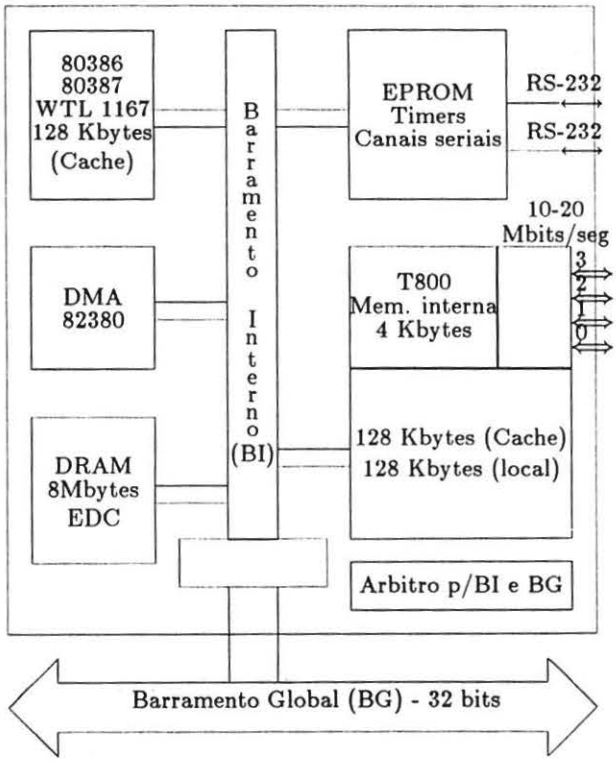

Figura 3: Placa U-32

As memórias cache do 80386 e do T800 operam somente sobre os acessos à memória DRAM interna a placa e são mantidas coerentes com o conteúdo da mesma, por circuitos que operam de forma completamente transparente ao software.

Os processadores de ponto flutuante (80387, WTL1167 e T800) podem operar contemporaneamente.

O barramento global pode ser acessado por 80386, 82380 (DMA) e T800, de forma idêntica.

Existem mecanismos hardware para implementação de semáforo, tanto para o 80386 quanto para o T800. Esses mecanismos podem ser utilizados tanto para acessos à DRAM interna como para acessos à memória externa.

Os integrados 80386, T800 e 82380 (DMA) podem executar escrita simultânea (broadcast) em todas as memórias DRAM das U32 conectados no barramento, inclusive a própria. Esta operação é executada utilizando uma determinada faixa de endereços reconhecida por todas as placas (0FCH nas linhas A24-31) e não pode ser interrompida por nenhum outro elemento (lock). Nesta faixa de endereços o bloqueio (lock) opera inclusive nos ciclos de leitura de forma a possibilitar uma operação indivisível de leitura-modificação-escrita em broadcast.

2. 2. Módulo de gerência

O módulo de gerência (figura 4) contém, além das 8 placas U32, algumas outras para controle de periféricos padrões. Em princípio elas são:

- CON controladora de terminais. Pode controlar até 32 estações de trabalho constituídas de:
  - vídeo e teclado PC - compatíveis
  - 2 linhas seriais (terminal, impressora, emulação de terminal)
  - interface paralela tipo centronics.
- DIS controladora de unidades de disco flexível e disco rígido winchester (ST506).
- SMD controladora de alto desempenho para unidades de disco rígido padrão SMD.

As placas de processamento executam as seguintes funções:

- servidor de arquivos
- controlador de terminais
- supervisão do sistema hardware
- balanceamento de carga

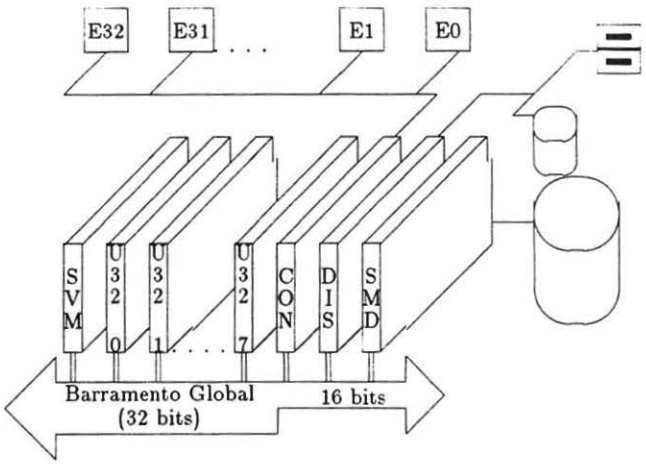

Figura 4: *Módulo de gerência*

- funções de suporte (depuração, rastreamento, etc)
- operações centralizadas dos algorítimos paralelos (distribuição dos programas e dos dados, coleta dos resultados, armazenamento em memória de massa, formatação, visualização, etc).

3. SISTEMA DE COMUNICAÇÃO

Para efeito de comunicação entre duas unidades genéricas do sistema é preciso considerar dois tipos de transferência:

- intramodular: a comunicação ocorre via barramento de alta velocidade utilizando a visibilidade global da memória das placas U32.
- intermodular: a comunição entre dois módulos utiliza dois canais seriais bidirecionais de 10-20 Mbits/seg.

Na comunicação entre duas unidades U32 do mesmo módulo teremos um único trecho de alta velocidade no barramento. Na comunicação entre duas unidades U32 de módulos diferentes teremos um trecho nos canais seriais e zero, um ou dois trechos intramodulares. Para melhorar o tempo de comunicação é conveniente alocar no mesmo módulo os programas com muita comunicação mútua.

A interconexão entre módulos utiliza 272 cabos cada um com dois pares trançados blindados. A interconexão ponto a ponto com pares balanceados e adequadamente terminados permite a transmissão dos pacotes assumindo a hipótese de ausência de erros. Isso evita a necessidade de controle a cada pacote, mas não dispensa a execução de testes iniciais e periódicos, que tornem a transmissão mais confiável, assim como mais rápida a deteção de falhas hardware.

3. 1. Análise de Comunicação

Este item contém uma rápida análise do desempenho das vias de comunicação assumindo a hipótese de tráfego uniforme entre os 136 processadores do sistema. Nesta condição cada processador comunica com qualquer outro a mesma quantidade de dados e a sua saída é igual a sua entrada.

As percentagens de tráfego interno e externo ao módulo são:

PTIN=7/135= 0,052

PTEX=1 PTIN=0,948

Do tráfego externo teremos uma parte que não usa o barramento (comunicação direta entre placas processadoras em módulos diferentes), uma parte que usa um barramento (no módulo de origem ou no módulo de destino) e uma parte que usa dois barramentos. Em relação ao tráfego total temos:

PTEX0=PTEX*((4/16)*(1/8)= 0,029

PTEX2=PTEX*((12/16)*(12/16)= 0,562

PTEX1=1-PTIN-PTEX0-PTEX2= 0,357

Para cada byte de tráfego na saída de um processador teremos os seguintes tráfegos no barramento e nas vias de comunicação (do processador):

PBAR= PTIN + PTEX1 + 2*PTEX2= 1,533

PVIAS= 2*PTEX= 1,896

No barramento o tráfego na entrada de um processador é ao mesmo tempo a saída de um outro e não precisa ser computado para a análise de carga do barramento. Nas vias de comunicação seriais é necessário considerar a soma da entrada e saída porque, mesmo que a comunicação seja fisicamente bidirecional, existem interações entre as duas direções de comunicação que podem diminuir a velocidade máxima do canal no seu conjunto.

Assumindo que a perda de desempenho dos processadores devido a problemas de comunicação não possa ser superior a 25% calcula-se o fator de utilização do barramento que provoca esta perda.

ep = 0,75=1/(1+fub(np-1))

onde:
ep = eficiência dos processadores
fub = fator de utilização do barramento de cada processador ligado no mesmo.
np = número de processadores no barramento (8).

Obtém-se fub= 0,0476 que, com uma velocidade de transferência no barramento de 12 Mbytes/seg, corresponde a 571/PBAR (367) Kbytes de saída para cada processador. Nas vias de comunicação seriais teremos 367*PVIAS=696 Kbytes/s o que dá 174 Kbytes/s para cada uma das 4 vias de comunicação da placa processadora.

Este trafego é muito inferior ao máximo possível (800 kbytes/s) com a configuração mais lenta das vias de comunicação (T414 e vias de 10 Mbytes/s) e pode-se supor que estas vias não constituem gargalo. Na configuração mais rápida (T800 e vias de 20 Mbits/s) a máxima taxa de transferência em cada via é de 2,3 Mbytes/s.

Esta ampla margem de velocidade permite a utilização do T800 para funções adicionais àquela de comunicação aumentando a capacidade de processamento da U32.

Em uma configuração de 5 módulos (40 U32) onde cada processador está interligado diretamente com cada um dos outros 4 módulos, teremos:

PTIN = 0,179 PTEX = 0,821

PTEX0= 0,103 PTEX1= 0,718 PTEX2= 0

PBAR = 0,897

PVIAS= 1,642

Isso corresponde na hipótese de 75% de eficiência dos processadores, a uma saída, isto é, a uma produção de 636Kbytes/s em cada placa U32 e a um tráfego de 271 Kbytes/s em cada via serial de comunicação.

4. SISTEMA OPERACIONAL

A utilização eficiente de um sistema de processamento paralelo do tipo aqui descrito requer que o usuário defina o seu algorítimo considerando a estrutura da máquina e que determine onde e quando carregar e executar as partes do seu problema. Uma alternativa a esta abordagem poderá ser oferecida por linguagens de quarta geração mas antes de se chegar nisso é oportuno possibilitar a execução do software aplicativo existente (Fortran na sua maioria) sem grandes dificuldades.

Um conjunto resumido de funções básicas que devem ser disponíveis para o Sistema Operacional e/ou para uma aplicação são apresentadas a seguir.

(a) carregamento do Sistema Operacional (Kernel) em todos os processadores.

(b) instalação de um usuário em uma unidade em um módulo ou em todos os módulos (localização definida).

(c) carregamento de um programa em uma unidade ou em um módulo (localização definida).

(d) carregamento de um programa em **n** unidades (localização não definida).

(e) partida dos programas carregados

(f) mecanismo para envio (SEND) e recepção (RECEIVE) de mensagens para sincronização e troca de dados.

O Sistema Operacional PP-SO/P -Processador Preferencial Sistema Operacional modo Protegido- (multiusuário, multiprogramação, tempo real) [7] foi desenvolvido especificamente para arquiteturas multiprocessadoras baseados na troca de mensagem e suporta facilmente o conjunto básico acima.

A função **f** é implementada diretamente por primitivas do Sistema Operacional. As funções **b**, **c** e **e** são facilmente implementados via chamadas a bibliotecas valendo-se do fato que no SO/P a instalação de usuário e a carga de um programa são executáveis remotamente enviando sinais pré-definidos.

A função **d** requer a intervenção do algorítimo de equalização de carga das unidades U32. A ativação deste algorítimo deverá ser requisitada via sinal e a função **d** poderá também ser implementada com uma simples chamada à biblioteca.

A função **a** é utilizada somente pelo Sistema Operacional na fase de iniciação do sistema.

O exemplo a seguir (na linguagem CHILL) ilustra a forma de implementação de uma linguagem para processamento paralelo valendo-se das facilidades fornecidas pelo PP-SOP.

```
1  NEWMODE    FLAG=SET  (PAR, SEQ)

.
.
2  I:LOADPROG ("MATINV","MTZA MTZB","P0250",PAR)

.
.
3  LOADPROG   ("MATINV","MTZC MTZD",NULL,SEQ)
```

A instrução 1 define um novo tipo de variável que pode assumir os valores PAR (0) e SEQ (1). A instrução 2 é expandida como uma simples chamada de biblioteca onde os parâmetros são o programa a ser executado (por exemplo MATINV pode ser um programa que inverte matrizes), os parâmetros para o programa (passados como cadeia de caracteres ''MTZC MTZD"), o processador onde o programa deve ser carregado (P0250) e a indicação de execução paralela. O programa é carregado (a sua execução é iniciada automaticamente) e o chamador continua a sua execução. Na variável I será colocada a identificação do programa carregado.
Na instrução 2 não é fornecido o processador onde deve ser feito a carga e a rotina de biblioteca assume que seja o próprio processador onde ela está executando. O parâmetro SEQ indica para a biblioteca que deve esperar o fim da execução de MATINV antes de continuar

a execução do chamador.

5. LINGUAGENS DE PROGRAMAÇÃO

Para que a aplicação possa ser desenvolvida sem dificuldades intransponíveis é necessário que sejam colocados na linguagem pelo menos as funções listadas no item anterior e que existam facilidades de depuração dos programas distribuídos nos diferentes processadores.

As linguagens escolhidas, além do CHILL que já suporta as chamadas para o SO/P, são C e **Fortran**. A primeira pela sua alta difusão e porque pode ser utilizada como base para o desenvolvimento de linguagens de quarta geração. A segunda pela existência de uma grande quantidade de software escrito nesta linguagem para aplicações que podem valer-se eficientemente do processamento paralelo.

Como alternativa a implementação de um novo compilador preferiu-se, por razões de prazo, colocar na linguagem comandos especiais a serem analizados sob o aspecto da sintaxe e da semântica por um pré-compilador.

O pré-compilador, além de espandir as chamadas ao Sistema Operacional e/ou à biblioteca, poderá gerar os fontes dos programas paralelos definidos em uma única listagem.

Além das listagens fontes o pré-compilador gera os comandos para a compilação e eventualmente os comandos para a ligação (ou parte dos parâmetros utilizados na ligação).

Além da introdução de uma fase de pré-compilação é necessário trocar parte das bibliotecas da linguagem (por exemplo toda a parte relacionada com E/S) e introduzir novas bibliotecas. Estas últimas podem ser residentes (não são ligadas com o programa e são chamadas via interrupção) de forma a gerar executáveis menores e melhorar a utilização da memória em um contexto de multiprogramação.

6. CONCLUSÕES

A utilização de uma estrutura fisicamente híbrida, isto é, de uma estrutura em que os processadores são em parte interligados via barramento paralelo, utilizando até memória comum, e em parte via canais seriais apresenta vantagens em relação as outras duas estruturas mais 'ortogonais". Vantagens restritas, naturalmente, ao modelo de processamento paralelo por troca de mensagens onde o roteamento das informações é executado inteiramente pelo sistema operacional ficando completamente escondida, para o software de aplicação, a complexidade da estrutura.

Com relação á estrutura puramente conectiva podemos comparar com o hipercubo de 7 dimensões que interconecta 128 processadores. Cada processador deve ter 7 vias de comunicação em lugar das 4 da estrutura híbrida do P3.

Supondo que as vias de comunicação sejam implementadas com 82586 (10 Mbits/s com uma eficiência do 50%) a produção dos processadores será limitada a cerca de 616 Kbytes/s. Isso supondo que o processador não seja sobrecarregado pela função de roteamento e que haja perdas somente quando as vias chegarem a 100% de utilização. Na realidade em cada canal teremos a concorrência da saída gerada pelo processador para aquele canal, da entrada dos outros 6 canais e que precisa ser retransmitida (o 71% do tráfego é retransmissão) e da entrada na mesma via (o 82586 transmite em uma direção por vez).

Na prática somente uma pequena parte desta velocidade poderá ser conseguida.

Um parâmetro importante para avaliação do sistema é a medida do tempo para efetivar uma comunicação. Este tempo é proporcional ao número de segmentos a serem percorridos. Enquanto que no hipercubo de 7 dimensões o número médio de segmentos é de 3,527 na estrutura do P3 alcança-se um número médio de 0,948 (considerando que a comunicação via barramento tem tempo desprezível em relação ao uso das vias seriais.)

A utilização do barramento pela U32 tem um custo baixo e vale-se de características desenvolvidos para acesso a placas controladoras de periféricos e para sistemas de processamento paralelo fortemente acoplados

(memória comum) com um número limitado de processadores (menos que 18).

A utilização de vias de comunicação por processador e não por módulos, como a estrutura parece sugerir, é motivada pela exigência de modularidade e pela redução de tráfego no barramento (25% a menos no sistema com 17 módulos e 55% a menos no sistema com 5 módulos).

Com relação as estruturas físicas baseadas em memória comum é preciso observar que a custo de uma matriz de comutação de altíssima velocidade não é compatível com um modelo de processamento paralelo baseado na troca de mensagens.

Uma vantagem desta estrutura é a utilização de uma única placa de processamento U32 (80386) compatível com a placa UPN de 16 bits (80286) e utilizável em aplicações diferentes como estação de trabalho e sistemas de processamento paralelo no modelo "memória comum".

A compatibilidade com o micro de 16 bits permite a utilização das placas controladoras de periféricos já desenvolvidos e, em uma primeira fase, a utilização do mesmo software.

Mas o que nós consideramos como o maior mérito deste sistema é a existência de um sistema operacional (PPSO/P) aberto e com facilidades para o processamento paralelo que dificilmente poderão ser encontrados em outros sistemas.

Referências

[1] SEITZ C, L. "The Cosmic Cube", **Communications of the ACM**, pag. 22-23, Janeiro/85.

[2] FATHI, E.T. et al, "Multiple Microprocessor Systems: What, Why and When", **Computer** Vol. 16, No. 3, pag. 23-22, Março/83.

[3] HILLIS W. D. et al, "Data parallel Algorithms", **Communication of the ACM** vol. 29 No 12, Dezembro/86

[4] MUDGE T. N, HAYES J. P et al, "Multiple Bus Arquitectures", **Computer** pag. 42-48, Junho/87

[5] BIANCHINI R. P et al, "Interprocessor Traffic Scheduling Algorithm for Multiple - Processor Networks", **IEEE Transactions on Computers**, pag. 396-409, Abril/87.

[6] SIEGEL H.J et al, "Large-scal parallel processing, System", **Microprocessor e Microsystems** Vol. 11 No. 1, Janeiro/Fevereiro 1987.

[7] CAVALLI E., ZABEU M. C., "Sistema operacional para processamento paralelo", **Anais 1o. SBACC**, pag. 101-114, Maio/87.

[8] CAVALLI E., ZABEU M. C., "Sistema hardware para processamento paralelo", **Anais 1o. SBACC**, pag. 91-100, Maio/87.