SEGUNDA GERAÇÃO DE PROCESSADORES PARALELOS DO "ADVANCED COMPUTER PROGRAM" (ACP) COLABORAÇÃO FERMILAB/URA/DOE & DRP/CBPF/CNPq /MCT

Apresentado por:
B.Schulze, R.Valois
Centro Brasileiro de Pesquisas Físicas - CNPq
Departamento de Relatividade e Partículas
Rua Dr. Xavier Sigaud, 150 - Urca, 22290 - Rio de Janeiro, RJ

RESUMO

O Departamento de Relatividade e Partículas do CBPF-CNPq participa de uma colaboração de Física Experimental de Altas Energias com o Fermi National Accelerator Laboratory - URA/DOE. Das experiências realizadas resulta uma enorme quantidade de dados a serem processador e portanto uma grande quantidade de cálculo a ser realizado. Visando atender estas necessidades computacionais surgiu o ACP no no FNAL com um projeto de processamento paralelo. O DRP desde então vem participando do projeto ACP visando suprir algumas de suas necessidades computacionais. No momento temos um sistema ACP de 20 nós com CPUs 68020 funcionando no departamento e estamos participando do desenvolvimento da segunda geração do sistema ACP. Este trabalho apresenta alguns aspectos e características deste projeto.

ABSTRACT

The Department of Relativistics and Particle Physics (DRP) at CBPF-CNPq is participating of a collaboration in High Energy Physics with the Fermi National Accelerator Laboratory - URA/DOE. The experiments generate a enormous quantity of data to be reconstructed off-line and consequently a lot of calculations to be done. To cover this leak of computational power the ACP group was formed to implement a parallel processing system. The DRP is participating since then on the project in a sense to solve some of its own computational needs. At the moment there is ACP system working at the department at CBPF, with 20 nodes of 68020 CPUs and at the moment me ar participating on the development of ACP second generation. This paper describes some of the aspects and characteristics of this project.

1. INTRODUÇÃO

Os sistemas ACP atualmente em funcionamento demonstram a validade da utilização em paralelo de uma grande quantidade de processadores de alta performance.

Com a disponibilidade de novos microprocessadores mais velozes e mais potentes, o plano atual é de construir uma segunda geração de ACPs mais poderosa [1]. Não será descrita em maiores detalhes a primeira geração [2,3].

Podemos classificar a implementação da segunda geração de ACPs da seguinte forma:

1) novas unidades de processamento;

2) nova comunicação inter-processadores;

3) eliminação da dependência de um hospedeiro do tipo uVAX II.

Na segunda geração do sistema será feita uma redição do "software", para tornar possível a utilização total das potencialidades dos novos processadores/comunicação mantendo a compatibilidade com a geração anterior. Pretende-se que os novos processadores possam ser utilizados também na primeira geração do ACP.

2. METAS DA SEGUNDA GERAÇÃO

1. O sistema não é mais configurado com um único hospedeiro mestre e vários nós escravos. Qualquer nó no sistema pode assumir algumas das funções do hospedeiro da primeira geração, ou seja, passa a existir um "mestre" gerenciando o sistema, enquanto as tarefas de entrada e saída estão diluídas pelos demais nós.

2. As tarefas de E/S são realizadas por vários nós de processamento simultaneamente, através de periféricos distribuídos como nós do sistema.

3. Qualquer nó no sistema pode comunicar-se com qualquer outro nó, sem necessidade de intervenção do "mestre".

4. Os nós podem ser (logicamente) configurados em grupos, com dados fluindo de um grupo para o próximo grupo.

5. Qualquer CPU utilizando o sistema VMS / UNIX pode ser um nó conectado ao "Branch Bus" ou ao "Ethernet", permitindo grande flexibilidade.

3. NOVAS UNIDADES DE PROCESSAMENTO

Estão sendo desenvolvidas duas novas unidades de processamento (UP). Uma UP utiliza microprocesadores MIPS atingindo uma performance 10x superior a das atuais UPs (68020/68881 com 2 MB de memória na placa). As placas serão em padrão VME, com gerenciamento de interrupções externas, 4 MB de memória, com gerenciamento de memória e capaz de suportar um sistema operacional do tipo "UNIX". Arquitetura RISC de 32 bits, 8 MHz e

performance de 8 mips [4]. Uma outra UP em desenvolvimento utiliza microprocessadores WEITEK ("array processors") e ao invés do barramento VME, utiliza o próprio protocolo do barramento tronco para intercomunicação.

4. NOVA COMUNICAÇÃO INTER-PROCESSADORES

A implementação de uma nova comunicação inter-processadores ocorre em dois estágios:

- uma estrutura de chaveamento do barramento tronco, a nível das UPs em um mesmo bastidor [7]:

- um módulo VBBC e um módulo BBBWI, a nível da comunicação entre bastidores onde residem as UPs [5,6].

A estrutura de chaveamento do barramento tronco estabelece uma comunicação direta nó-nó através de uma chave de 16 pólos/16 posições de 46 vias (figura A). Esta chave consiste de 13x CIs AS 8840 [9] (figura B). Para cada nó são copiadas as 46 linhas úteis do BT (figura C) . Para cada nó vão então 46 linhas da chave mais um barramento de 7 linhas de endereçamento e controle de cada nó (figura D). Entenda-se que nó pode ser uma UP ou um periférico conforme definição adiante.

Características da ECBT:

1. Possui 16 portas ("spigots") TTL bidirecionais compatíveis com o barramento tronco;
2. Banda de 20 Mbytes/s para cada porta;
3. Banda total da ECBT de 160 Mbytes/s;
4. A operação da ECBT é transparente para o barramento tronco;
5. A ROM de endereçamento ("roteamento") é capaz de endereçar até 2048 nós;
6. A ROM de "roteamento" possui uma rota primária e uma rota alternativa;
7. Módulos, gabinete e conector em padrão Euro card.

Um exemplo de uma sequência de operação da ECBT:

1. Um mestre do BT realiza a arbitragem para utilizar o seu BT local e sinaliza um BREQ com o endereço de um nó;
2. O árbitro da ECBT identifica o BREQ e armazena o endereço do nó;
3. Uma ROM interna é utilizada para determinar qual porta poderá ser usada para a conexão com o nó solicitado;
4. Se a porta requerida estiver disponível a ECBT realiza arbitragem necessária;
5. Se a arbitragem for bem sucedida a ECBT estabelecerá a comunicação com o nó ou BT remoto;
6. Se a arbitragem não for bem sucedida a ECBT sinalizará com um BERR no barramento solicitante;
7. Os passos 2 a 6 são repitidos para cada porta no percurso até o nó destino.

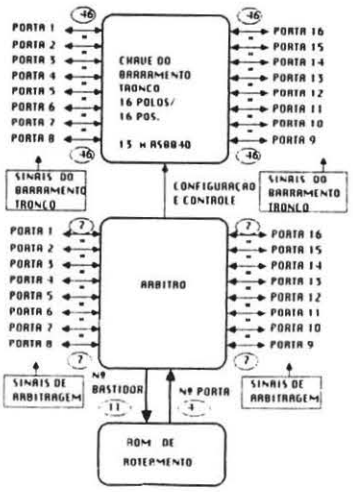

Figura A - Diagrama em Blocos da Estrutura de Chaveamento do Barramento Tronco. Utilizando 13 CIs AS 8840

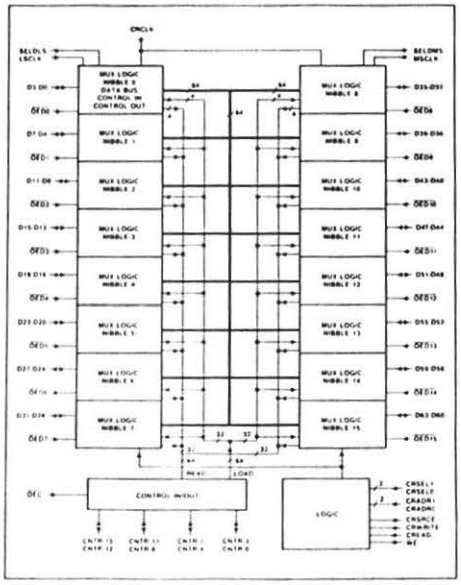

Figura B.1 - Diagrama em Blocos do CI AS 884 da Texa Instruments

Na figura E vemos uma interconexão típica utilizando a ECBT. Os próximos parágrafos descrevem os módulos de comunicação a nível da comunicação entre bastidores.

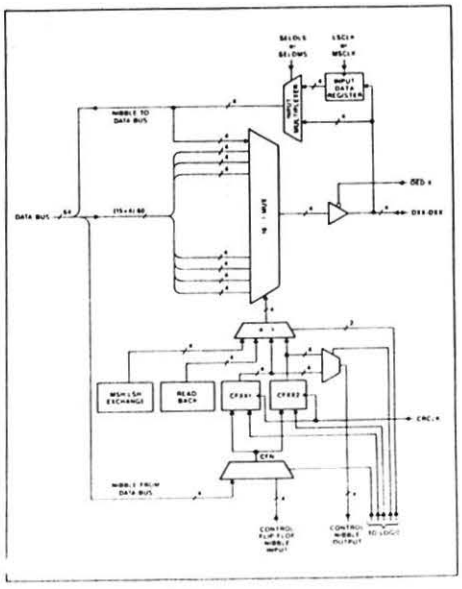

Figura B2 - Detalhe da Lógica de um "NIBBLE"

conector J1			conector J2		
1			1		
GND	. .	GND	GND	. .	GND
AD00-	. .	AD00+	RESET-	. .	RESET+
AD01-	. .	AD01+	BE-	. .	BE+
AD02-	. .	AD02+	BREQ-	. .	BREQ+
AD03-	. .	AD03+	BGNT-	. .	BGNT+
AD04-	. .	AD04+	SPARE1-	. .	SPARE1+
AD05-	. .	AD05+	SPARE2-	. .	SPARE2+
AD06-	. .	AD06+	PAR2-	. .	PAR2+
AD07-	. .	AD07+	AD16-	. .	AD16+
AD08-	. .	AD08+	AD17-	. .	AD17+
AD09-	. .	AD09+	AD18-	. .	AD18+
AD10-	. .	AD10+	AD19-	. .	AD19+
AD11-	. .	AD11+	AD20-	. .	AD20+
AD12-	. .	AD12+	AD21-	. .	AD21+
AD13-	. .	AD13+	AD22-	. .	AD22+
AD14-	. .	AD14+	AD23-	. .	AD23+
AD15-	. .	AD15+	AD24-	. .	AD24+
PAR1-	. .	PAR1+	AD25-	. .	AD25+
AS-	. .	AS+	AD26-	. .	AD26+
DS-	. .	DS+	AD27-	. .	AD27+
DSV-	. .	DSV+	AD28-	. .	AD28+
R/W-	. .	R/W+	AD29-	. .	AD29+
TC-	. .	TC+	AD30-	. .	AD30+
WAIT-	. .	WAIT+	AD31-	. .	AD31+

Figura C - Pinagem do Barramento Tronco em Padrão RS 485 (J1,J2).

Numero	coluna a	coluna b	coluna c
1	GND	GND	GND
2	+5V	+5V	+5V
3	-12V	-12V	-12V
4	AD00	AD01	AD02
5	AD03	AD04	AD05
6	AD06	AD07	PAR
7	AD08	AD09	AD10
8	AD11	AD12	AD13
9	GND	GND	GND
10	AD14	AD15	PAR1
11	AD16	AD17	AD18
12	AD19	AD20	AD21
13	AD22	AD23	PAR2
14	AD24	AD25	AD26
15	AD27	AD28	AD29
16	AD30	AD31	PAR3
17	+5V	+5V	+5V
18	GND	GND	GND
19	AS	DS	/DSV
20	W/R	/WAIT	/TC
21	/BE	/RESET	/BREQ
22	BGNT	GND	GND
23	SPIGOT BUSY	/ARB PASS	/ARB FAIL
24	/SPIGOT REQ	DEST SPIGOT BSY	DEST SPIG GNT
25	SUCH REQ SPIG		
26		GND	
27			
28		GND	
29			
30	+12V	GND	+12V
31	+5V	+5V	+5V
32	GND	GND	GND

Figura D - Descrição do Conector de cada porta (SPIGOT) da Estrutura de Chaveamento do Barramento Tronco.

O módulo VBBC (VME - Branch Bus Controller) consiste de um controlador do barramento tronco (barramento) definido para comunicação VME e realiza a interface VME - BT. O barramento tronco permite uma transferência de dados a uma taxa de 20 MB/s VME-BT. Este módulo é constituído da placa mãe de um módulo de interface (BVI) da primeira geração com um novo submódulo de interface com o BT que adiciona o recurso de multi-mestre a cada nó do barramento tronco (BT). Anteriormente o único mestre do BT era o hospedeiro.

Este módulo estabelece 16 níveis de prioridade para a arbitragem do barramento tronco utilizando as 16 linhas inferiores do barramento de endereço/dados do barramento. É adicionado um ciclo de arbitragem e uma linha de requisição CREQ.

O módulo de interface entre o BT e a estrutura de chaveamento do BT (ECBT), realiza a conversão de sinais em padrão RS-485 do BT para sinais TTL, para ECBT. Em adição a esta função esta interface realiza as funções de arbitragem do BT e da ECBT. Este módulo pode efetivamente tornar-se um mestre ou escravo no BT.

A comunicação com o sistema ACP poderá ser feita por "Ethernet" através do bastidor mestre (figura F), permitindo que o sistema fique ligado a uma rede de comunicação. Esta função já suportada pelo hospedeiro da primeira geração, continuará a ser suportada pela segunda geração.

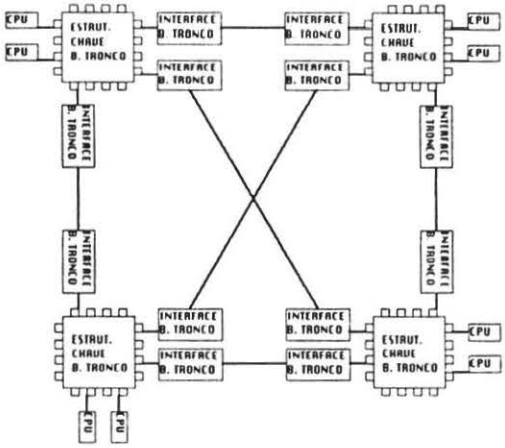

Figura E - Estrutura de Chaveamento do Barramento Tronco e uma Interconexão Típica de Nodos.

5. ELIMINAÇÃO DA DEPENDÊNCIA DO TIPO DO HOSPEDEIRO

Com as novas unidades de processamento operando com um sistema operacional do tipo "UNIX" , não será mais necessário um uVAX como hospedeiro (Front End) do sistema. Uma das UPs pode executar as funções de hospedeiro do sistema . As funções de hospedeiro, ou seja, nó mestre, consistem em gerenciar o sistema, isto é, ter controle sobre o "status" do sistema, conhecer sua configuração de nós, alocar/deslocar nós, etc...

6. IMPLEMENTAÇÃO DO NOVO "SOFTWARE"

O novo "software" deverá ser capaz de satisfazer as novas exigências e potencialidades do "hardware". Deverá possuir:

1. habilidade para executar programas do ACP em uma grande variedade de nós de processamento;

2. um conjunto de ferramentas de desenvolvimento de programas para as novas UPs;

3. um conjunto de ferramentas de comunicação interprocessadores que permitam comunicação rápida e bidirecional entre nós;

4. um conjunto de ferramentas de gerenciamento de recursos que permitam a um usuário de um sistema ACP estabelecer facilmente uma comunicação de envio-bloco-dados /recebo-bloco-dados com um membro do sistema, e que permitam usuários múltiplos compartilhando eficientemente o sistema.

6.1 Ferramentas de Desenvolvimento de Programas.

As ferramentas básicas consistem de um sistema operacional UNIX para o nó ACP "mestre" e nó ACP "normal", um conjunto de utilitários do UNIX para desenvolvimento de programas (compiladores Fortran, concatenadores, biblioteca e depurador) e uma biblioteca de rotinas ACP.

6.2 Ferramentas de Comunicação Inter— processadores

As ferramentas de intercomunicação suportadas consistem de um conjunto de módulos de "hardware" de intercomunicação (VBBC, QBBC, UBBC, Ethernet), um "driver" correspondente para cada módulo de intercomunicação e um conjunto de subrotinas de alto nível que permitam a um usuário conversar de um processador para outro. Quanto a unidade utilizada, deve ficar transparente ao usuário.

6.3 Ferramentas de Gerenciamento de Recursos

Para gerenciamento de recursos estão incluídos um processo Gerente do Sistema (GS) e um processo Gerente de Usuário (GU). O GS desempenha aproximadamente a mesma função na segunda geração que o HARDCON ("hardware configuration file") desempenha na primeira geração, ou seja, contém a informação dos recursos disponíveis no sistema. O processo gerente de usuário indaga ao GS as características do sistema e solicita a alocação de recursos para um usuário. O GS estará normalmente localizado no ACP mestre.

O GU tem duas funções principais:

. Solicitar a alocação dos recursos que um usuário necessita e iniciar todos os processos do usuário;

. gerenciar a comunicação entre os processos do usuário permitindo a comunicação ao nível de blocos comuns (chama acp-send e acp-get).

O GU informa sobre cada processo do usuário e responde as mensagens de cada processo. A informação mantida pelo GU inclui:

. a localização de cada processo. Qual a CPU em que o processo está rodando e o percurso de comunicação a ser tomado para a comunicação com o mesmo;

. o endereço e o tamanho de cada bloco comum ("common block") no processo, que tenha sido declarado disponível para comunicação por bloco comum;

. o "status" de cada processo.

7. CONFIGURAÇÃO DE UM SISTEMA ACP DE SEGUNDA GERAÇÃO.

7.1 Em linhas gerais, um sistema ACP de segunda geração contém o seguinte hardware (figura F):

- um "bastidor mestre (fig.G) " de nós ACP, assim denominada por conter o nó mestre do sistema. "Nó ACP mestre" é o nó que atua como anfitrião e que serve o disco para os demais nós, quando estes são inicializados ("booted").

O nó mestre deve rodar UNIX e deve estar conectado tanto ao BT quanto ao "Ethernet". O BT interconecta todos os bastidores com nós ACP, enquanto "Ethernet" é uma rede para interconexão de computadores. Portanto o mestre serve de via de acesso entre nós no BT e outros computadores conectados ao "Ethernet" (figura F, G).

- zero ou mais "bastidores normais (figura H)" de nós ACP, assim denominada por conter somente nós normais, ou seja, não contém o nó mestre. "Nó ACP normal" é uma UP sem disco que se inicializa através de conversação com o nó mestre via BT.Um Nó normal pode ser qualquer processador em padrão VME rodando UNIX.

- zero ou mais computadores distintos dos nós ACP, conectados ao "branch bus", rodando UNIX ou VMS;

- zero ou mais computadores distintos dos nós ACP conectados a "Ethernet", rodando UNOX ou VMS.

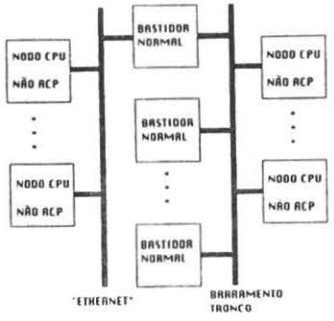

Figura F - Sistema ACP de Segunda Geração

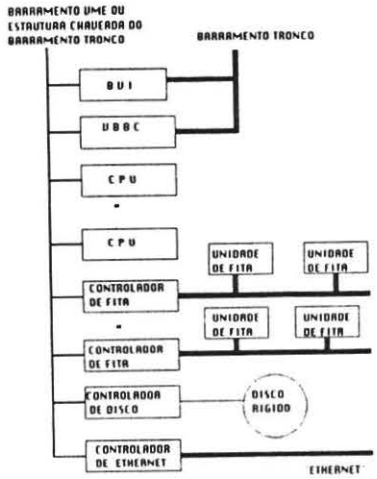

Figura G - Bastidor Mestre

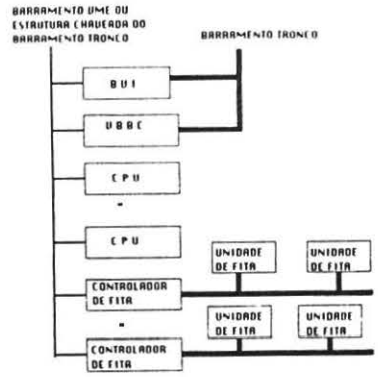

Figura H - Bastidor Mestre

7.2 Em linhas gerais o "software" da segunda geração tem a seguinte composição:

Cada CPU no sistema roda um ou mais processos. Cada processo pode comunicar-se com qualquer outro processo. A comunicação pode ocorrer em dois níveis - ao nível do "driver" da unidade e ao nível de blocos comuns ("common blocks").

A comunicação ao nível do "driver" da unidade é o nível de comunicação mais primitivo entre dois processos. O nível de comunicação de blocos comuns é do tipo envio-blocos/recebo-bloco que é parte da primeira geração. Um usuário do sistema ACP pode enviar dados para (ou receber de) um bloco comum em um outro processo sem precisar saber o endereço físico do bloco comum ou o tipo de barramento de comunicação através do qual os dados estão sendo transferidos.

O nível de comunicação por bloco comum é implementado com a ajuda do processo GU. Se o sistema ACP está utilizando este nível de comunicação deve haver um processo GU para cada usuário do sistema ACP. Cada processo GU acompanha todos os outros processos deste usuário, gerando informação a respeito. Esta informação consiste de:

. a localização dos blocos comuns ("common blocks") no processo, que serão utilizados na comunicação interprocessos;

. o tipo de computador em que o processo está sendo executado, por exemplo, um nó MIMPS ou VMS Vax;

. o percurso de comunicação que pode ser utilizado para transferir dados para/do processo;

. o "status" de cada processo, por exemplo, se o processo está pronto para enviar outro evento para processamento.

8. DESENVOLVIMENTO DE PROGRAMAS PELO USUÁRIO.

A primeira geração de ACP dispõe de um utilitário denominado MULTICOMP que facilita ao usuário compilar e concatenar programas nos nós e no hospedeiro. Na segunda geração não existe o multicomp. O usuário do sistema prepara o programa da mesma maneira que um programa é preparado em qualquer outro computador (edição, compilação e concatenação com uma biblioteca ACP). Em uma UP ACP, as ferramentas de desenvolvimento de programa serão aquelas utilizadas em um sistema operacional do tipo "UNIX".

9. EXECUÇÃO DE PROGRAMAS DO USUÁRIO.

No nível mais básico, um programa pode ser executado num sistema ACP de segunda geração através da alocação de um nó ACP (executando um comando "login" remoto pela rede) e emissão de um comando "run". Esta é a forma de execução durante o desenvolvimento de um programa. Quando da execução de um programa, digamos de produção, isto ocorre através da execução do programa GU. Um tal programa GU pode ser iniciado interativamente ou por submissão a um mecanismo de "batch".

Quando o GU começa a ser executado, ele lê informações de um Arquivo GU (UMF). Este arquivo contém todas as informações necessárias ao GU para iniciar a tarefa. Estas informações incluem:

. o número de grupos de processos;

. o tipo de computador para cada grupo de processos;

. o nome de cada arquivo contendo os programas de determinado grupo.

O GU solicita a alocação dos recursos solicitados (CPUS e unidades de fita) ao GS e inicia a execução de todos os processos. Quando a tarefa está completa, o GU desaloca todos os recursos de volta ao GS e finaliza.

10. GERENCIAMENTO DE ARQUIVOS.

Os compiladores e concatenadores ("linkers")da segunda geração de ACP rodam debaixo de "UNIX" e, portanto, necessitam dos arquivos em disco com os quais operam. Tais arquivos podem residir ou em um disco UNIX conectado diretamente (por exemplo no mesmo barramento VME) ao computador que está executando a compilação/concatenação ou então os arquivos podem ser acessados pela rede de comunicação (utilizando protocolo

apropriado e conversão de arquivo em disco não UNIX para UNIX).

11. "STATUS" DOS PROCESSOS.

. "ready" - O programa está pronto para ser carregado com um evento para processar.

. "running" - O processo está trabalhando em um evento que tenha sido enviado.

. "done" - O processo conclui o processamento de um evento e está aguardando que os resultados sejam lidos (por outro processo em outro grupo).

. "dead" - O processo foi declarado incapaz de executar qualquer outro trabalho devido um erro explícito identificado pelo usuário ou devido a uma falha do processo em proceder conforme determinado pelo GU.

. "allocated" - O processo está alocado por um outro usuário com o propósito de enviar/receber dados.

12. IDENTIFICAÇÃO DO PROCESSO.

Cada processo é identificado pelo GU por um número lógico. O número lógico do processo é atribuído após a mensagem de cada processo de que foi iniciado. Em seguida o GU retorna para cada processo: o número lógico do processo, o número do grupo do processo e uma lista de todos os números lógicos dos demais processos em cada grupo. A informação retorno do GU ao processo também contém a descrição de onde cada processo está localizado (por exemplo o endereço de BT, para processos que estão rodando em nós ACP).

13. CONCLUSÃO.

Na segunda geração almeja-se obter maior potência de cálculo, sistema operacional mais flexível ("UNIX"), eliminar a dependência de um hospedeiro específico (DEC - uVAX, VAX) e incrementar substancialmente a intercomunicação entre nós de processamento (VBBC, "crossbar switch"). Em consequência é esperado um aumento de aplicações científicas bastante significativas. Espera-se ter no início de 1989 um sistema destes em funcionamento no Fermilab. O CBPF participa de projeto, no momento, no desenvolvimento de "software" básico.

14. REFERÊNCIAS BIBLIOGRÁFICAS.

[1] Biel, J.; Mark, E.: "ACP Second Generation Design - Preliminary", FNAL - ACP Report 3/16/88.

[2] Documentação do sistema ACP de primeira geração, FNAL-ACP.

[3] "O Sistema ACP no CBPF", apresentado por C. Barros neste simpósio.

[4] "Performance Brief - MIMPS M/500 with UMIPS-BSD 1.0", MIMPS Computer System, INC..

[5] Atac, R., "MultiMaster Branch Bus Specification", FNAL-ACP, March 10,87.

[6] Atac, R., "Branch Bus to Switch Interface", FNAL-ACP April 1,87.

[7] Atac, R., "Bus Switch Specification - BSS", FNAL-ACP April 3,87.

[8] "Branch Bus Specification", FNAL-ACP, March 6, 1986.

[9] Texas Instruments AS8840 Specification.