

MARCO DIMAS GUBITOSO

SIMULAÇÃO DE THREADS EM AMBIENTE MULTI-CORE

MARCELO KENJI MATSUZAKA & MARCIO GUEDES HASEGAWA

Multi Core

A evolução dos processadores convencionais (single core) chegou ao seu ápice com o pentium 4 da intel, no qual o limite físico de processamento foi atingido. A solução foi criar um processador que simulasse o dobro de núcleos existentes. Então, surgiu a tecnologia hyper-threading da intel que aproveita a disponibilidade excedente de processamento para obter melhor desempenho.

A idéia do hyper-threading estimulou o desenvolvimento de processadores com mais de um núcleo (multi core). Um computador com processador multi-core é eficiente em situações nas quais existem várias threads concorrentes.

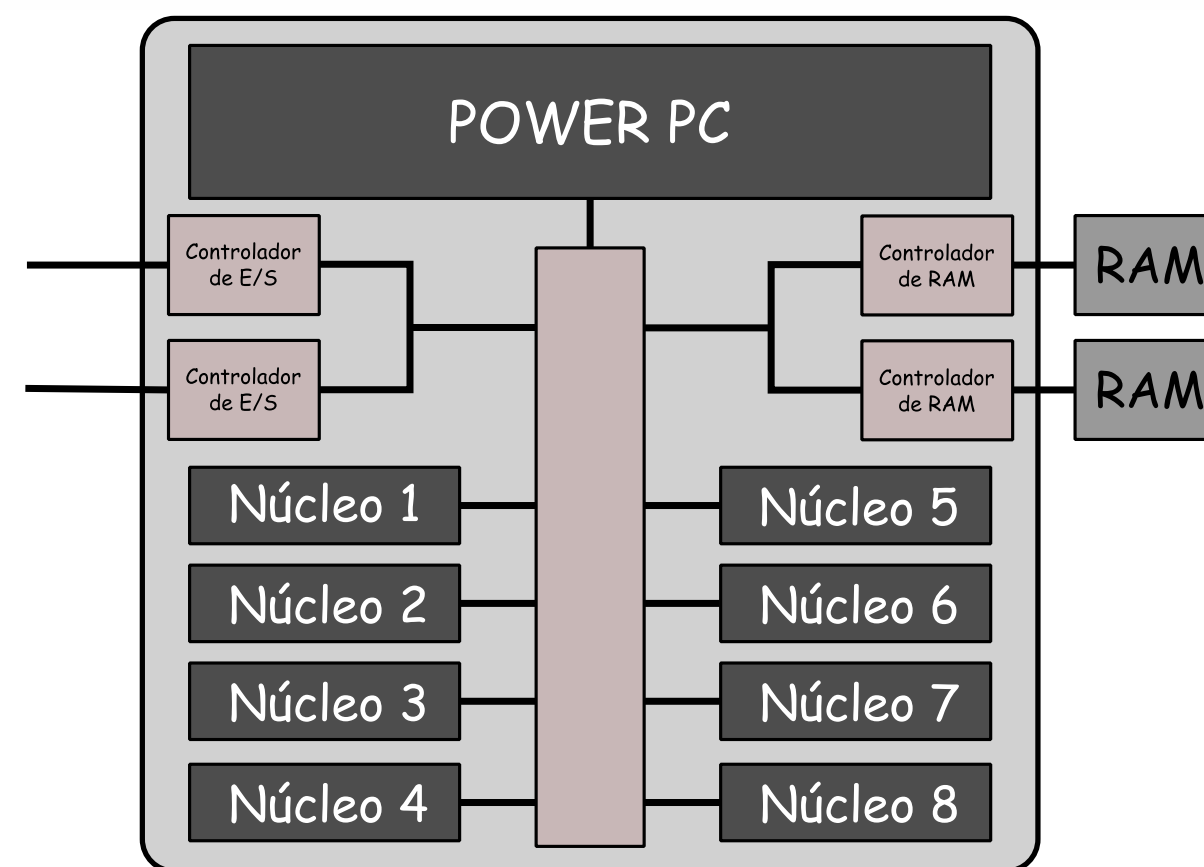
É importante salientar que um computador com múltiplos processadores não é a mesma coisa que um computador com um processador de múltiplos núcleos. A vantagem do segundo é que a interação feita entre os núcleos consegue ser mais rápida pois não há a necessidade de barramentos de comunicação.

CELL

CELL é um projeto de processador multi-core desenvolvido pelas empresas Sony, Toshiba e IBM. Foi criado como ponte entre os processadores convencionais e os de alta performance, como por exemplo os processadores gráficos.

Hoje, sua grande aplicação é nos consoles do video-game Sony PlayStation 3, porém num futuro próximo, é planejada sua utilização em sistemas que necessitam de grande poder de processamento em tempo real (processamento de vídeo do sinal de televisão digital e simulações de realidade virtual).

A execução de um processo no CELL é gerenciada por um núcleo PowerPC central, que delega aos 8 núcleos cooperativos, otimizados para cálculo vetorial, as execuções das threads existentes.



PThread

No decorrer do tempo, foram implementadas muitas versões de proprietárias de threads. A diferença entre elas criou grande dificuldade para a criação de aplicações portáteis.

Assim, para que o aproveitamento das funcionalidades das threads fosse aprimorado, foi desenvolvida uma interface padrão. Para o sistema UNIX, a interface escolhida foi a POSIX.

Portanto, todas as implementações referentes a essa interface foram denominadas como POSIX Threads ou PThreads.

Biblioteca de Simulação

A biblioteca foi desenvolvida para ser utilizada como camada intermediária entre a implementação do usuário e a biblioteca POSIX thread. Além das funções básicas da biblioteca, como por exemplo criar, destruir e esperar o fim da execução da thread, foi adicionado o monitoramento dos tempos das threads para, em posterior análise, simular a execução do processo em ambiente multi-core.

Para isso, foi desenvolvida uma função oculta que executa uma outra especificada, marcando os tempos de sistema e de usuário, além do tempo de parede num arquivo de saída.

A análise do arquivo de saída é realizada depois da execução completa do processo. De acordo com o ambiente desejado, ela "monta" os tempos de execução das threads no número de núcleos pré-definido, realizando uma aproximação de como seria a execução do dado processo numa máquina de características desejadas.

