

Múltiplos Núcleos

Multiple Cores

Melo Gustavo Henrique

Graduando em Ciência da Computação – IFSULDEMINAS - Muzambinho

Melo Marcelo Júnior

Graduando em Ciência da Computação – IFSULDEMINAS – Muzambinho

Vilhena, Perses de

Graduando em Ciência da Computação – IFSULDEMINAS – Muzambinho

RESUMO. *Multicore (múltiplos núcleos, do inglês multicore) consiste em colocar dois ou mais núcleos de processamento (cores) no interior de um único chip. Em processadores de múltiplos núcleos o sistema operacional trata cada um desses núcleos como um processador diferente. Na maioria dos casos, cada unidade possui seu próprio cache e pode processar várias instruções quase simultaneamente.*

ABSTRACT. *Multicore consists of placing two or more processing cores within a single chip. In multicore processors the operating system treats each core as a separate processor. In most cases, each unit has its own cache and can process multiple instructions almost simultaneously.*

1. INTRODUÇÃO

Desde a última década, a evolução dos processadores acontece de forma surpreendente. Isto se deve principalmente ao aumento na densidade de integração de transistores em áreas cada vez menores, a melhoria de desempenho na computação está relacionada aos vários métodos de paralelização de aplicações em arquiteturas com múltiplos núcleos.

Neste nível de integração, é necessário explorar técnicas de paralelismo para efetivamente utilizar ao máximo o desempenho dos processadores. *Pipeline*, *superescalaridade*, *multithreading*, são algumas técnicas aplicadas a exploração de paralelismo para aumento de desempenho. Pipeline superescalar, aplica o processamento das instruções divididos em estágios, possibilitando a execução de mais de uma instrução por ciclo, e para isso, aumenta o número de unidades funcionais e técnicas para solucionar falsas dependências entre instruções. Os processadores superescalares são capazes de aumentar o desempenho na execução de cargas de trabalho com alto paralelismo no nível de instruções. O suporte a múltiplos threads foca a exploração de paralelismo de maior granularidade, explorando o paralelismo além do nível de instruções, mas também no nível de fluxo de instruções (threads). Isto aumenta a vazão de threads (possibilitando que mais de uma thread possa ser executada ao mesmo tempo) diferente da superescalaridade onde a vazão é de instruções de uma única thread.

2. ARQUITETURA MULTICORE

O desenvolvimento de processadores multicore tem como motivação duas realidades: o grande consumo de potência e as limitações de manter a evolução de desempenho, como prevê a Lei de Moore. Com isso tornou-se necessário algumas adaptações a nível de arquitetura para o melhor aproveitamento dos seus núcleos, e as primeiras a serem feitas tem como objetivo melhorar o paralelismo a nível de threads e assim criar uma arquitetura que propicie o chamado paralelismo ao nível do chip. O aumento de desempenho alcançado pelos processadores se baseia na aplicação de técnicas que exploram o paralelismo de instruções, e também pelo aumento de frequência de operação. Entretanto, o consumo de potência também atinge níveis impraticáveis, já que é proporcional ao aumento da frequência de trabalho. Uma arquitetura multicore é geralmente um multiprocessamento simétrico (Simultaneous Multiprocessing - SMP, em que múltiplos núcleos de processador tipicamente compartilham um segundo ou terceiro nível de cache comum e interconectado), implementado em um único circuito VLSI (Very Large Scale Integration). O objetivo de um sistema multicore é permitir maior utilização de paralelismo no nível de threads, especialmente para aplicações que faça pouco uso do paralelismo em nível de instrução para fazer um bom uso de processadores superescalares. Em processadores single-core que executam vários programas, este define diferentes períodos para execução de cada programa, e isto pode gerar erros, conflitos ou queda de desempenho, quando precisar desempenhar muitas tarefas simultaneamente. As arquiteturas multicores apresentam a possibilidade de distribuir as diferentes tarefas pelos vários núcleos, obtendo maior eficácia do sistema e desempenho aprimorado de aplicativos, mesmo executando vários aplicativos simultaneamente. Surgem novas possibilidades de extração de paralelismo, e então, a possibilidade de extrair paralelismo inter-thread. Esta característica pode ser explorada por máquinas capazes de executar mais de um thread simultaneamente. Outras arquiteturas capazes de explorar essas características são as máquinas multiprocessadas e máquinas SMT (Simultaneous Multithreaded). O objetivo é suportar tanto paralelismo em nível de instrução quanto no nível de threads, mas estas somente são suportadas pela existência de mais de um núcleo, já que cada núcleo é responsável pela execução de apenas uma thread. Processadores multicore que suportam a execução multithreading são chamados de Chip Multithreading (CMT).

3. Arquiteturas de Redes em Chip (NoC – Network on Chip)

O projeto de redes de comunicação entre os elementos de um processador multicore enfrenta as limitações de modularidade e escalabilidade, com especificações mais rígidas e complexas. O desenvolvimento e aplicação de NoCs apresenta-se como uma ótima resposta para esta questão.

A premissa básica é simples: o desenvolvimento das interconexões em chip devem seguir os mesmos princípios que são aplicados em projetos de rede de comunicação em um nível macroscópico, que demonstram escalabilidade sustentável, aumento de desempenho exponencial, notável confiança e robustez. A natureza modular e escalável de NoCs e seu apoio para uma eficiente comunicação on-chip potencialmente levam-na para aplicação em processadores multicore caracterizados pela alta complexidade estrutural e diversidade funcional. Por um lado, esses recursos devem ser devidamente abordados por meio de nova concepção de metodologias, enquanto por outro lado, mais esforços devem ser dedicados a modelagem de arquiteturas de NoCs e integrá-los em um único ambiente de modelamento e simulação combinando ambos elementos de processamento e comunicação. O barramento é desenvolvido sob conceitos bem compreendidos e fácil de ser modelado. Em um sistema multicore altamente interconectado, porém, pode rapidamente tornar-se uma comunicação não eficiente. Quanto mais unidades são adicionados a ele, o consumo de potência na utilização da comunicação em cada evento cresce.

O uso de links dedicados ponto-a-ponto são ótimas em termos de disponibilidade de largura de banda, latência, potência e usos que são concebidos especialmente para este determinado fim. Mas o número de ligações necessárias aumenta exponencialmente como o número de núcleos aumenta. Assim, problemas acerca de área e roteamento surgem. Do ponto de vista do esforço de desenvolvimento, em sistemas com menos de 20 núcleos, uma estrutura de comunicação ad hoc é viável. Para a máxima flexibilidade e escalabilidade, é explorada a ideia de uma comunicação segmentada, dividida. Este conceito traduz-se em uma rede de roteamento de dados composto de links de comunicação e roteadores que são implementados no chip.

A NoCs é composta basicamente por três elementos:

- Roteador: responsável pela interconexão da rede, pela definição de rotas, pelo controle de fluxo, qualidade de serviço e pela garantia de entrega do pacote de dados. Por se tratar de uma rede de comunicação composta por roteadores, o mecanismo de entrega de dados é através de passagem de mensagem ou pacotes de rede;
- Links de comunicação: são os elementos que interligam os roteadores. Responsáveis pelo caminho entre fonte e destino dos pacotes a serem trafegados.
- Interface de rede: É necessária para garantir a correta comunicação entre a rede (roteadores da NoC) e os núcleos ou periféricos que estão interconectados. Esta interface garante que haja uma correta comunicação entre protocolos diferentes (NoC, núcleo, memória, etc).

É possível citar três tipos de topologias: fixas, sem fio e reconfiguráveis. As topologias fixas são alternativas clássicas de adoção de uma determinada forma de interconexão que privilegie um comportamento específico de uma determinada carga de trabalho. Topologias sem fio são alternativas recentes para eliminar as limitações do fio no projeto de NoCs através de uma tecnologia chamada de Radio-on-Chip. Dentre as topologias estudadas e desenvolvidas para arquiteturas NoC, as mais encontradas na literatura são baseadas em mesh e torus. A principal característica está relacionada à capacidade de suportar aplicações cujos problemas podem ser particionados (e.g., operações com matrizes e processamento de imagens).

4. CONCLUSÃO

Com a grande evolução dos processadores multicore, abre um grande campo para pesquisas e desenvolvimento visando a melhoria dos processadores existentes, os tornando-os com maior poder de processamento e menor consumo energético. Aliando as novas tecnologias de paralelismo pode tornar as processadores com um grande ganho na capacidade de execução de múltiplas tarefas simultaneamente, gerando um grande ganho no tempo de execução e evitando a sobrecarga de um único núcleo. A aplicação de Redes em Chips em arquiteturas de processadores multicore abre uma vasta área de pesquisa e desenvolvimento, devido ao aumento de elementos integrados dentro de um único chip, este tipo de solução deve buscar a melhor relação entre desempenho, área e consumo de potência.

REFERENCIAS

FOSTER, D. C., (2009) "Arquiteturas Multicore"

FREITAS, H. C., NAVAUX, P. O. A. (2006) "Chip Multithreading: Conceitos, Arquiteturas e Tendências." (Desenvolvimento de material didático ou instrucional), TI 1253, PPGC/UFRGS, 2006.

BENINI, L., MICHELI, G. D., (2005) "Network-on-chip architectures and design methods", IEEE Proceedings Computers & Digital Techniques, Vol. 152, Issue 2, pp. 261-272, 2005.

KOLODNY, Avinoam. (2006), "What is Network-on-Chip?", disponível em: http://www.sigda.org/newsletter/2006/eNews_060415.html, ACM/SIGDA ENewsletter, Vol. 36, No. 8, 15 de Abril 2006. Acessado em 25 de Junho de 2009.

OLUKOTUN, K. et al., (1996), "The Case for a Single-Chip Multiprocessor", 7th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), pp.2-11, 1996.