

ENSINO DE ARQUITETURA DE COMPUTADORES UTILIZANDO SIMULADORES COMPLETOS

André E. Lourenço - andre.lourenco@poli.usp.br

Edson T. Midorikawa - edson.midorikawa@poli.usp.br

Departamento de Engenharia da Computação e Sistemas Digitais

Escola Politécnica da Universidade de São Paulo

Av. Prof. Luciano Gualberto, Trav. 3, nº 158 - CEP 05580-900 - São Paulo/SP - Brasil

***Resumo:** Os currículos dos cursos de Engenharia da Computação, Ciência da Computação e afins ministrados nas universidades brasileiras e estrangeiras possuem, dentre outras, a disciplina de Arquitetura de Computadores. Esta disciplina tem como objetivo apresentar ao aluno o computador como um agregado de componentes eletrônicos, mostrando o seu funcionamento no nível de linguagem de máquina e de linguagem assembly.*

O curso de Arquitetura de Computadores possui um conteúdo altamente complexo e seja qual for o método adotado pelo docente, a assimilação por parte dos alunos é sempre difícil, pois o aluno é exposto a um certo nível de detalhes nem sempre visível. Procurando por soluções pedagógicas mais favoráveis, desenvolveram-se diversas metodologias a fim de sanar o déficit de aprendizagem dos alunos. Uma possível solução para este problema foi o desenvolvimento dos simuladores completos, ou seja, aplicações capazes de simular por completo todos os componentes de uma máquina real, além de ser uma alternativa simples e barata para a maioria das universidades que não podem dispor de laboratórios com vários sistemas de computadores com arquiteturas diferentes.

O objetivo deste trabalho é propor uma nova metodologia de ensino para os cursos de Arquitetura de Computadores, principalmente aqueles voltados para as novas arquiteturas de 64 bits. Nesta proposta se levará em consideração a utilização de simuladores completos como uma ferramenta de apoio pedagógico onde se apresentará, a um custo relativamente baixo, propostas para as aulas teóricas e práticas.

***Palavras Chaves:** Engenharia de Computação, Arquitetura de Computadores, Processo Pedagógico, Simulação, Diretrizes Curriculares.*

1. INTRODUÇÃO

O currículo dos cursos de Engenharia da Computação e afins que são ministrados nas universidades brasileiras e estrangeiras possuem dentre outras, disciplinas básicas, humanísticas e complementares. Estes cursos são geralmente parecidos e em muitos casos diferem-se apenas naquelas disciplinas que visam a formação com objetivos regionais. Dentro das disciplinas do grupo básico, tem-se a disciplina de **Arquitetura de Computadores**, cujo objetivo é apresentar ao aluno o computador como um componente eletrônico, mostrando o seu funcionamento no nível de linguagem de máquina e linguagem assembly. Isto permite ao aluno compreender melhor todos os níveis de abstração que se fazem presentes em um ambiente computacional. Porém devido ao alto grau de complexidade da Arquitetura de Computadores, o nível de compreensão dos alunos é bastante baixo, o que leva em muitas

vezes a uma deficiência de aprendizagem, e o mais grave, esta deficiência será propagada durante todo o curso.

Diante de tal situação, o que seria possível alterar ou modificar (na ementa da disciplina), sem que haja perda de abrangência e qualidade de conteúdo? Em outras palavras será possível um mecanismo, ou um método pedagógico, em que o aluno de Arquitetura de Computadores pudesse ver a execução das instruções no nível de registradores, ou ainda, visualizar a transferência de dados da memória principal para a memória cache?

2. ENSINO DE ARQUITETURA DE COMPUTADORES

2.1 Diretrizes Curriculares

No Brasil, o MEC – Ministério da Educação e Cultura e a SBC – Sociedade Brasileira de Computação são os órgãos que trabalham na estruturação dos cursos de Engenharia da Computação, Ciência da Computação e outros cursos, com o objetivo de manter um currículo de referência para a formação dos cursos. As Diretrizes Curriculares contêm uma estrutura curricular abstrata, organizada de tal forma que as Instituições de Ensino Superior possam, a partir desta, conceber currículos plenos diversificados que atendam as necessidades regionais onde se situam. Esta estrutura serve como base, para que através de detalhamentos sucessivos, chegue-se a um grupo de disciplinas distribuídas ao longo do curso, formando assim um currículo pleno que deve ser executado pelo corpo docente.

As Diretrizes Curriculares contêm também, orientações de como detalhar a estrutura abstrata, dependendo do perfil do curso desejado. Na Tabela 1 está reproduzido como a disciplina de Arquitetura de Computadores deve ser aplicada em cada um dos perfis de cursos da área de computação e informática e como ela deve ser detalhada.

Tabela 1 - Distribuição da Arquitetura de Computadores em vários perfis de curso [01]

Cursos	Tópico	Arquitetura de Computadores
Bacharelado em Engenharia de Computação		As disciplinas deste tópico devem cobrir os assuntos com abrangência e profundidade
Bacharelado em Ciência da Computação		As disciplinas devem cobrir esta matéria com abrangência e profundidade
Bacharelado em Sistemas de Informação		As disciplinas devem cobrir esta matéria de forma abrangente e geral
Licenciatura em Computação		As disciplinas devem cobrir esta matéria de forma abrangente e geral

Para manter uma referência internacional, dois órgãos internacionais procuram regulamentar os cursos da área de tecnologia. A ACM (*Association for Computing Machine*) e o IEEE (*Electrical and Electronic Engineers Computer Society*) procuram desenvolver currículos de referência para a computação/engenharia de um modo geral.

Assim como o MEC/SBC, a ACM/IEEE também procuram orientar na formação de cursos com a indicação de um currículo mínimo a ser seguido, não no sentido de disciplinas básicas e introdutórias, mas no sentido de disciplinas obrigatórias em todo e qualquer currículo das áreas de informática, fornecendo uma divisão entre disciplinas do núcleo básico e disciplinas fora do núcleo básico, assim como uma sugestão da quantidade de horas. Nesta classificação, a **Arquitetura de Computadores** aparece dentro do núcleo de disciplinas básicas e uma relação do conteúdo proposto pelos órgãos aparece Tabela 2. Observe-se que a quantidade de horas não deve ser vista como uma medida absoluta, mas sim para indicar a ênfase a ser dada em cada assunto. ,,

Tabela 2 - Curso de Arquitetura de Computadores proposto pela ACM/IEEE. ,

Assunto	Horas
---------	-------

Lógica digital e sistemas digitais	6
Representação de dados no nível de máquina	3
Arquitetura no nível de Assembly	9
Organização de memória	5
Interfaces e Comunicação	3
Organização funcional	7
Multiprocessamento e arquiteturas alternativas	3
Desempenho	opcional
Arquitetura de redes e sistemas distribuídos	opcional

2.2 Ensino Acadêmico

Muitos métodos pedagógicos do ensino de Arquitetura de Computadores trabalham de forma semelhante, ou seja, utiliza-se de pelo menos um dos seguintes itens: exposição, explicação, conferências, demonstração, textos escritos, audiovisuais, estudo de casos, trabalhos em grupo e simulação. A escolha do melhor método cabe somente ao docente que leva em consideração vários fatores dentre eles as características dos alunos, a situação em que se encontra o curso, os recursos disponíveis e por fim o estilo próprio.

Interessado em se mapear os métodos adotados nas universidades brasileiras, realizou-se uma pesquisa através dos sites de busca disponíveis na Internet (por exemplo www.google.com.br, www.altavista.com e www.radaruol.com.br). Pesquisou-se por cursos de Arquitetura de Computadores que pudesse fornecer, entre outras informações, detalhes sobre o método pedagógico adotado e as bibliografias recomendadas. Após uma análise, observou-se que atualmente, nas principais universidades brasileiras, utiliza-se o método tradicional de ensino, com aulas expositivas e o uso de livros textos e aplicações de exercícios. Com relação às referências bibliográficas, nota-se que mesmo existindo uma grande variedade de títulos disponíveis no mercado, há uma tendência de se seguir as referências bibliográficas utilizadas nas mais conceituadas universidades do mundo, ocasionando assim uma concentração por uma mesma bibliografia. A Figura 1 mostra as principais referências bibliográficas no Brasil

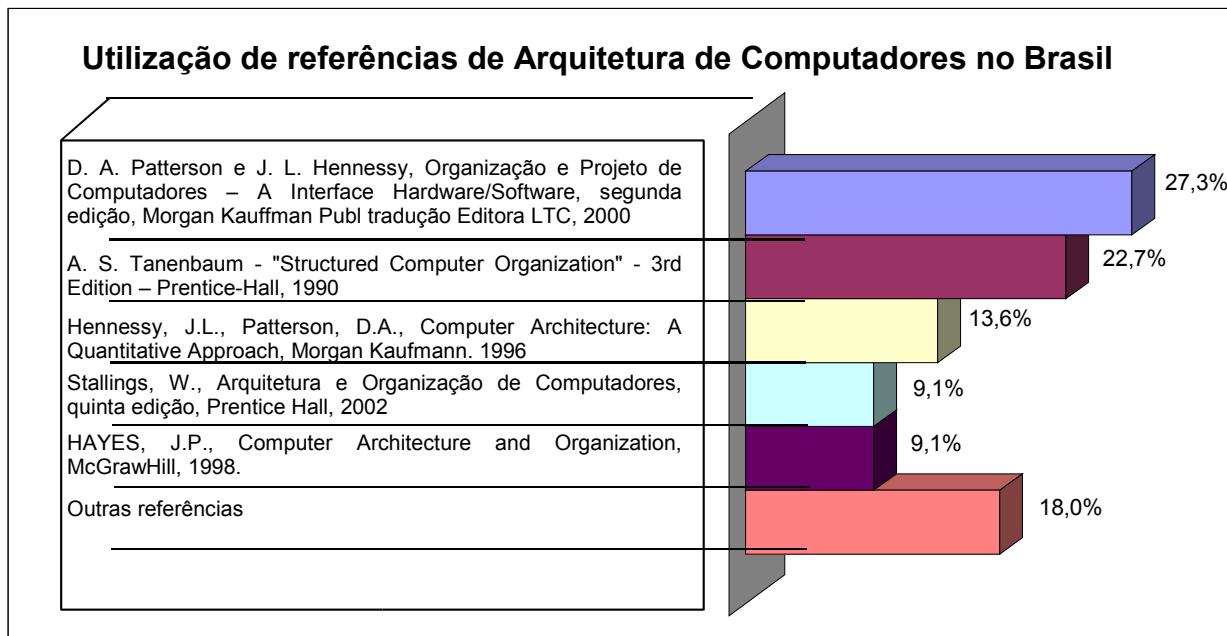


Figura 1 - Gráfico da utilização bibliográfica de Arquitetura de Computadores no Brasil

3. MÉTODO PROPOSTO

3.1 Métodos pedagógicos

No Dicionário Universal da Língua Portuguesa, a palavra **método** (Latim *methodu* - Grego *méthodos*) significa um caminho ou um processo racional para se atingir um determinado propósito. Já **pedagogia** (Latim *paedagogia* - Grego *paidagogia*) é a arte da educação e do ensino, ou ainda, é o estudo dos ideais de educação, segundo uma determinada concepção de vida, e dos processos mais eficientes para concretizar esses ideais. Para se trabalhar com um método específico é necessário antes, analisar os objetivos a serem alcançados, as situações a enfrentar, assim como os recursos e o tempo disponível, para por último observar as diversas alternativas possíveis. **Métodos Pedagógicos** pode então ser entendido como os diferentes modos de proporcionar uma determinada aprendizagem ou investigação científica. O método pedagógico não se refere ao conteúdo a ser transmitido e sim ao modo de como será a transmissão da informação.

A maioria dos métodos pedagógicos adotados para o ensino de Arquitetura de Computadores enfoca uma das seguintes formas:

1. **Tradicional:** Utilizando-se somente de livros textos;
2. **Máquinas Reais:** Focalizando em uma Arquitetura especial.

Em ambos os casos têm-se vantagens e desvantagens que devem ser analisadas de forma cuidadosa. No primeiro caso exige-se um nível bastante elevado de abstração por parte dos alunos, além de exigir conhecimento e sensibilidade do professor em aplicar o conteúdo de forma clara e precisa sem que haja perda da linha de raciocínio ou ambigüidade de conceitos. Uma vantagem neste caso é o custo, que é praticamente nulo. Já no segundo caso tem-se a necessidade de se possuir um laboratório específico onde são testados e aplicados os conceitos da arquitetura em estudo. A principal vantagem neste caso é a fácil absorção dos conceitos por parte dos alunos. Em contraste tem-se o custo que pode ser bastante elevado na fase de implantação e manutenção.

O curso de Arquitetura de Computadores possui um conteúdo altamente complexo e seja qual for o método adotado pelo docente, a assimilação por parte dos alunos é sempre difícil, pois o aluno é exposto a um certo nível de detalhes nem sempre visível. Como por exemplo, não se é possível abrir um processador e visualizar os seus circuitos e tão pouco analisar as instruções sendo executadas pela máquina, em tempo de execução. Procurando por soluções pedagógicas mais favoráveis, desenvolveu-se diversas metodologias a fim de sanar o déficit de aprendizagem dos alunos. Uma possível solução para este problema foi o desenvolvimento dos **simuladores completos**, ou seja, aplicações capazes de simular por completo todos os componentes de uma máquina real. O objetivo geral de um sistema simulador de arquitetura é fornecer ao professor uma ferramenta prática para o ensino de tópicos da disciplina de Arquitetura de computadores, facilitando o entendimento e o aprendizado do aluno, além de aproximar a teoria dada em sala de aula com experimentos práticos em laboratório. Uma outra vantagem é possibilidade de se testar arquiteturas novas sem mesmo que as mesmas existam, além de se verificar o comportamento de compiladores, sistemas operacionais, linguagem de máquina, circuitos integrados, etc. Como a maioria das universidades não pode dispor de laboratórios que contêm sistemas de computadores que representam várias arquiteturas diferentes, os simuladores tornam-se uma alternativa natural e barata. Estas são algumas das razões pelas quais existe um interesse crescente na utilização desta metodologia pedagógica nos cursos de Arquitetura de Computadores. A Tabela 3 mostra uma relação entre os alguns métodos utilizados por educadores e o novo método aqui proposto.

Tabela 3 - Comparativo entre os métodos de ensino de Arquitetura de Computadores

Método	Item	Análise
--------	------	---------

Tradicional	Metodologia	Aulas expositivas, leitura de livros e artigos, resolução de exercícios. Alguns utilizam exercícios de laboratório com máquinas reais ou pequenos simuladores específicos (Ex. simuladores de memória, simuladores de conjunto de instruções, etc).
	Vantagens	Baixo custo
	Desvantagens	Baixo índice de abstração por parte dos alunos, dificuldade para o professor lecionar, grande utilização de materiais de apoio (projedor, slides, etc).
Máquinas reais	Metodologia	Aulas expositivas com exercícios em laboratórios específicos para cada arquitetura estudada.
	Vantagens	Grande índice de abstração por parte dos alunos.
	Desvantagens	Alto custo de implantação e manutenção.
Simulação	Metodologia	Aulas expositivas com exercícios em laboratórios com simuladores completos de arquitetura podendo simular qualquer arquitetura específica.
	Vantagens	Baixo custo e níveis elevados de absorção do conhecimento (igual ao método com máquinas reais)
	Desvantagens	Aprendizagem do simulador, outras limitações (velocidade de execução reduzida, grande demanda de recursos da máquina hospedeira).

3.2 Simuladores completos e outras ferramentas

Várias ferramentas de software estão disponíveis para uso no ensino de Arquitetura de Computadores. De uma maneira geral, tem-se os simuladores e emuladores. A simulação e a emulação são as técnicas que reproduzem o ambiente de trabalho através de uma software ou ferramenta (emulador ou simulador), onde uma máquina simula um ambiente abstrato, sem mesmo que ele exista fisicamente, com o fim de proporcionar treino ou investigação. Porém deve-se ficar atento às particularidades de cada um dos termos emulador e simulador. Existem várias definições e os conceitos por muitas vezes se confundem, mas para este trabalho considera-se que **emulador** é um software ou hardware que possibilita a um equipamento (emulador) operar como um outro (emulado) com características distintas, trabalhando de tal forma a traduzir cada instrução do equipamento emulado para o equipamento emulador, lembrando que esta tradução pode acarretar em perda de desempenho. Não existe neste caso a possibilidade de se instalar novos aplicativos ou mesmo um novo sistema operacional. Já **simulador** prepara um ambiente para que se tenha a execução de todas as instruções simuladas sem a necessidade de traduzir instrução por instrução. Nos simuladores ditos completos, existe a possibilidade de se reiniciar o sistema de forma natural, da mesma maneira de quando se liga um equipamento real e observa-se o processo de boot. Outra característica dos simuladores completos é a possibilidade de se instalar aplicativos e até mesmo novos sistemas operacionais.

Dentre os diversos emuladores e simuladores disponíveis no mercado, pode se citar os seguintes:

1. Emuladores de Arquitetura:

- **Simularq:** Emulador simples de arquiteturas CISC; ,
- **Wine:** Emulador do Windows para Linux; e ,
- **Bochs:** Emulador de Arquitetura Intel x86.

2. Simuladores completos de Arquitetura:

- **VMWare:** Simulador completo para sistemas operacionais; e
- **SimICS:** Simulador completo de arquiteturas x86 e IA-64 da Intel, x86-64 da AMD, Sparc da SUN, PowerPC da IBM.

Neste trabalho se utilizará o simulador completo SimICS. A Figura 2 mostra a execução do SimICS no Linux, onde estão sendo simulados o Windows NT e Windows XP, várias versões do Linux, além de uma máquina Itanium I.

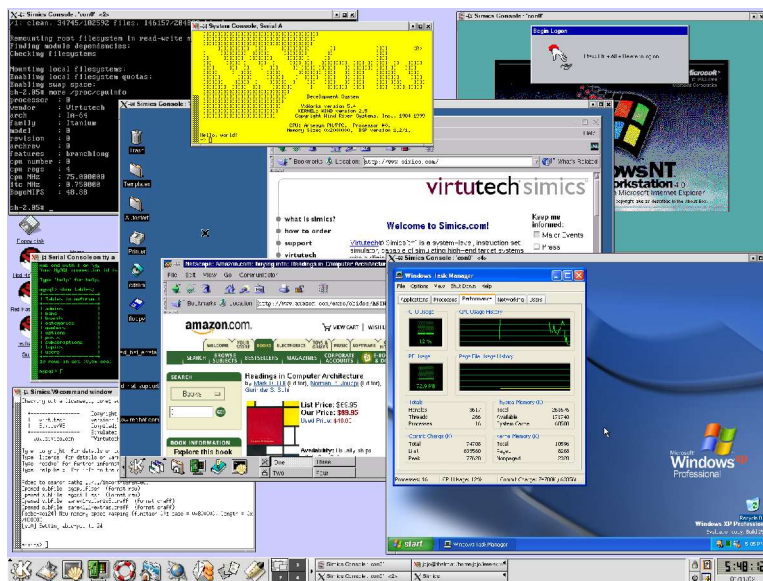


Figura 2 – Simulação de vários sistemas operacionais utilizando o SimICS

3.3 Análise de custos

Para que um curso de Arquitetura de Computadores de 64 bits seja melhor lecionado e ainda melhor recebido pelos alunos se faz necessário a utilização de um laboratório dedicado a este fim. Por outro lado implantar e manter um laboratório, por exemplo, com 20 máquinas Itanium é bastante custoso. Uma solução mais barata é a utilização dos simuladores completos de arquitetura. Outra vantagem dos simuladores completos é que se o docente resolver experimentar uma outra arquitetura em especial, isto se fará de forma natural sem a necessidade de se adquirir novos equipamentos e tampouco a aquisição de novos softwares.

A Virtutech possui uma licença acadêmica para a utilização do SimICS para fins educacionais e de pesquisa. Esta licença permite um número limitado de cópias simultâneas em funcionamento dentro do campus e dá acesso livre a todos os módulos do SimICS, porém existe uma taxa de manutenção anual que conforme o número de licenças pode ser gratuito no primeiro ano. Observe na Tabela 4 o custo de licenciamento do SimICS (em US\$) anuais.

Tabela 4 - Taxa de licenças do SimICS fornecidos pela Virtutech

Licenças	Primeiro ano (US\$)	Anos seguintes (US\$)
40	Gratuito	2.500
100	2.500	5.000
250	5.000	7.500
500	7.500	10.000

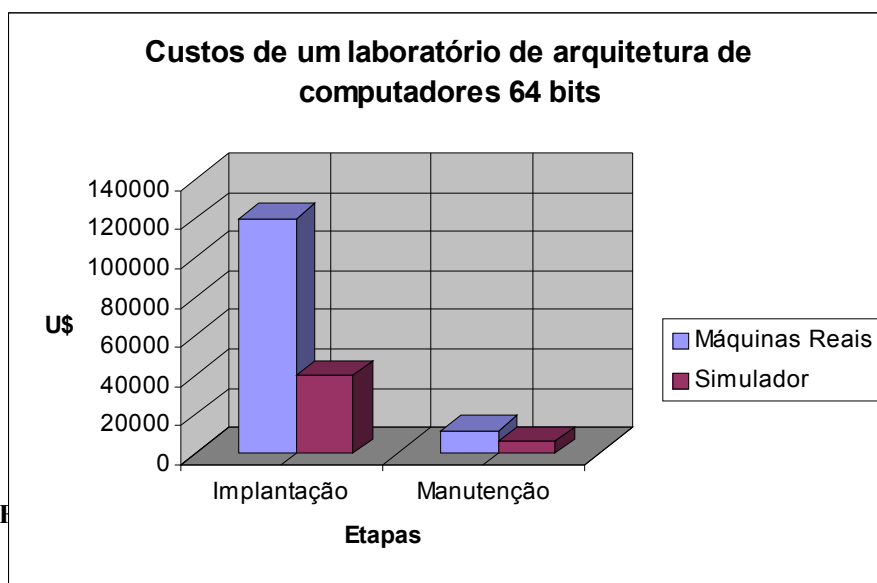
Na Tabela 5 pode ser observado o custo de implantação de um laboratório com 20 máquinas dedicado ao ensino de Arquitetura de Computadores para 64 bits, tanto para máquinas reais quando para um laboratório com o simulador SimICS. Pode ser observado que na utilização de simuladores há uma economia de mais ou menos 68% na fase de implantação. Caso se queira colocar os equipamentos em uma rede de computadores (acesso à Internet/Intranet), se deve orçar também US\$ 132 por cada ponto de rede (material do cabeamento e serviços de instalação; porta livre no switch).

Tabela 5 - Custo de implantação de um laboratório de Arquitetura de Computadores em US\$

Laboratório com 20 máq.	Descrição (hardware e softwares)	Valor Unit.	Valor Total
Máquinas reais	Dell – Poweredge 3250 c/ processador Itanium II	5.999	119.980
	Academic Microsoft Windows 2003 Server Client Access Licenses	7	140
	Total	6.006	120.120
Com Simulador	Pentium IV 2.4GHz 512MB 60G	2.000	40.000
	Red Hat 9.0	40	40
	Licença SimICS (1º ano)	0	0
	Total	2.020	40.040

Deve se considerar também o custo de manutenção do laboratório. Normalmente as próprias instituições mantêm funcionários responsáveis pela manutenção dos laboratórios. Em média para cada laboratório com 20 equipamentos são necessários pelo menos um técnico e um estagiário a um custo mensal de aproximadamente US\$ 970¹. Observe que em ambos os casos (máquinas reais e ambientes simulado) são necessários pessoal qualificado e treinado em cada tipo de solução. Outro fator importante é observar que as licenças para utilização do SimICS em um laboratório com no máximo 40 máquinas é de US\$ 2.500 anuais (\approx US\$ 209 mensais) a partir do segundo ano. Equipamentos novos geralmente possuem garantias por em média 3 anos, mas normalmente tem-se um custo anual de manutenção em torno de 10% do valor de um equipamento novo.

Observa-se que na manutenção dos equipamentos tem-se uma economia de mais ou menos 46%. Não se levou em consideração a aquisição e atualização dos softwares, porém note que os valores da Tabela 5 podem sofrer variações conforme a necessidades específicas como por exemplos: softwares dedicados, impressoras, espaço físico, etc. Observe ainda na Figura 3 um gráfico comparativo entre as duas soluções de laboratório dedicado a Arquitetura de Computadores.



3.4 Nova Metodologia

Os atuais métodos (tradicional e máquinas reais) exploram a transmissão oral do saber com resolução de exercícios, seja eles teóricos ou em pequenos simuladores. Utilizando-se

¹ Custos obtidos através de pesquisa entre os funcionários das instituições de ensino.

destes métodos, observa-se a baixa compreensão por parte dos alunos, já que os mesmos não visualizam de forma clara o que realmente está acontecendo na partes internas no sistema computacional. Pierre Goguelin classifica os métodos pedagógicos em três categorias: verbais, intuitivos e activos.

Os métodos intuitivos e activos exploram o “mostrar”, ou seja, para que o aluno possa intuir, aprender ou perceber o que se pretende transmitir, deve se mostrar o que está acontecendo. Portanto, o aluno consegue absorver um maior número de conhecimento se ele for exposto ao conteúdo por observação, reflexão e experimentação. Resumindo: é mais fácil entender vendo do que lendo.

Sendo assim, será possível um mecanismo, ou um método pedagógico, em que o aluno de Arquitetura de Computadores pudesse ver a execução das instruções ao nível de registradores, ou ainda, visualizar a transferência de dados da memória principal para a memória cache? Na verdade, o que se procura é fornecer ao graduando um meio dele "ver" a arquitetura do computador funcionando, e se possível, de alguma forma, interagir com essa arquitetura podendo acompanhar, passo-a-passo, a execução de um programa. Se isto fosse possível, a assimilação do conteúdo seria mais fácil do que apenas acompanhando aulas teóricas expositivas.

Uma alternativa para que se possa definitivamente visualizar o que está acontecendo dentro de uma máquina é a utilização de simuladores completos de arquitetura, ou seja, disponibilizar aos estudantes ferramentas de softwares que sirvam para aproximar o abstrato do concreto. A proposta é a utilização da simulação como uma ferramenta de apoio pedagógico e não de substituir a figura do professor. Dentre vários fatores, os simuladores são interessantes nos seguintes aspectos:

- Os estudantes visualizam detalhes da operação do computador em múltiplos níveis de abstração, conforme a ementa do curso;
- Os estudantes têm acesso a conteúdos, antes inacessíveis;
- Muitos tópicos (memória, registrador, etc) que antes estavam indisponíveis, podem ser explorados de forma natural;
- Muitos simuladores possuem uma correlação com livros textos utilizados em muitos cursos de Arquitetura de Computadores; e
- Pequeno ou nenhum custo de infra-estrutura é exigido.

Além disso, os simuladores permitem aos estudantes experimentar várias arquiteturas já existentes e até mesmo criar novas. Atualmente, criar um sistema por simulação é uma experiência educacional semelhante à de se construir um computador real em hardware, porém com **1** - custo menor, **2** - mais flexível, pois permite aos estudantes cometerem erros e recuperá-los e por fim **3** - mais extensível no sentido de se construir funcionalidades adicionais ao projeto original.

Assim como em qualquer curso de Arquitetura de Computadores tradicional, o curso de **Arquitetura de Computadores Auxiliada por Simulador** exige a necessidade de se ter uma ementa e o conteúdo muito bem elaborado e distribuído. Os estudantes podem aprender os fundamentos da disciplina e observar visualmente, além de interagindo com o fluxo de dados dentro de uma máquina particular hipotética como se estivesse realmente manipulando uma máquina real. E o mais importante, o custo de um curso de Arquitetura de Computadores Auxiliada por Simulador é bastante baixo.

4. ESTUDO DE CASO: ARQUITETURAS DE 64 BITS

4.1 Evolução de 32 para 64 bits

Devido ao aumento das complexidades computacionais e ao tamanho dos conjuntos de dados a serem processados, a evolução das arquiteturas de computadores é processo natural frente as novas pesquisas envolvendo softwares e hardwares que procuram aumentar a velocidade de processamento dos aplicativos. Um grande número de problemas relacionados, por exemplo, à previsão do tempo, pesquisas espaciais e outros, requerem um grande número de cálculos que utilizam vetores, matrizes e polinômios. Aumentando o tamanho do barramento não resolve o problema por completo, existe uma necessidade de se aumentar a capacidade processamento interno das CPUs, bem como a capacidades de armazenamento de informações junto aos registradores.

Dentre as soluções presentes no mercado, duas se fazem interessantes no sentido de manter a compatibilidade com as arquiteturas de 32 bits: Itanium-II (IA-64) da Intel , , e o Athlon 64 e Opteron (x86-64) da AMD . Essas arquiteturas apresentam um novo conceito no que se refere ao mecanismo de programação e compilação. No caso do Itanium, a grande diferença está no rompimento da natureza seqüencial das arquiteturas de processadores convencionais existentes até então, ao permitir que o software se comunique explicitamente com o processador, quando as operações puderem ser feitas em paralelo. O aumento de desempenho é obtido com a diminuição do número de desvios e prognósticos errados, e a redução dos efeitos de latência entre memória e processador. Já o Opteron introduz um novo mecanismo de gerência (modos de operação) de tal forma a garantir a compatibilidade com os aplicativos de 32 e 16 bits.

Sendo assim um curso de Arquitetura de Computadores voltado para 32 bits já está ultrapassado e a necessidade por atualização para 64 bits se faz necessário. Por outro lado é extremamente complicado ensinar uma nova Arquitetura de Computadores sem a experimentação prática dos conceitos teóricos, ainda mais quando estes conceitos, no caso do Itanium, são poucos convencionais. O ideal seria a montagem de um laboratório específico, onde os alunos pudessem verificar todos os componentes que formam a nova tecnologia.

Desta forma pretende-se desenvolver um plano de ensino voltado para arquiteturas 64 bits utilizando o simulador SimICS.

4.2 Exemplos de aulas

Em um curso de Arquitetura de Computadores Auxiliado por Simulador deve possuir aulas específicas, como por exemplo:

- Análise do código assembly gerado;
- Análise de desempenho;
- Mecanismos de acesso à memória;
- Estudo de arquiteturas paralelas;
- Estudo de arquiteturas distribuídas.

Por exemplo, para a aula de análise do código *assembly* gerado, utilizando o SimICS, o aluno deve gerar um código em linguagem C e compilá-lo dentro do ambiente simulado. Utilizando-se de breakpoints, editores de texto, visualizadores de registradores pode se verificar além do código assembly gerado, o status da máquina em um momento que achar coerente. Já uma aula de estudo de arquiteturas paralelas pode ser feita utilizando o SimICS para simular uma máquina paralela com n processadores, com mostra a Figura 4 onde é observado uma máquina paralela virtual com 10 processadores Itanium II em uma janela de console do simulador SimICS.

Para maiores informações recomenda-se consultar a referência .

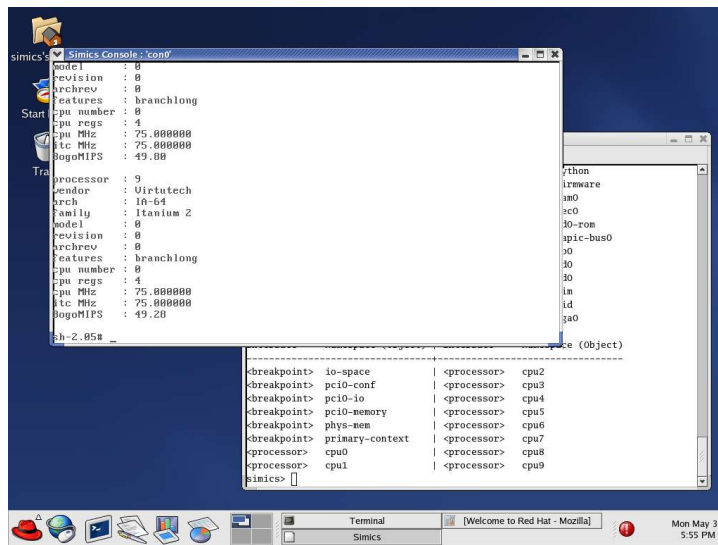


Figura 4 – Tela principal simulador com 10 processadores Itanium 2 utilizando SimICS

5. CONCLUSÕES E PERSPECTIVAS

Arquitetura de Computadores está se expandindo continuamente, conseqüentemente, o material de pesquisa só pode ser coberto parcialmente em um curso universitário típico. Desta forma existe a constante necessidade de se encontrar meios alternativos de aprimorar o ensino.

Pretende-se com o presente trabalho montar um curso de Arquitetura de Computadores que seja mais flexível e didático de forma a atingir o máximo de compreensão por parte dos alunos. Uma alternativa é a aplicação de ferramentas, como é o caso de simuladores completos, a fim de suprir esta necessidade. É claro que o docente tem que estar preparado para administrar esta nova proposta pedagógica, o que leva o professor a se atualizar e pesquisar novos meios didáticos afim de incluir as ferramentas disponíveis.

Por fim, outros cursos podem sofrer modificações a fim de se incorporar simulação como é o caso de sistemas operacionais, compiladores, circuitos digitais, etc. Estas disciplinas, assim como Arquitetura de Computadores, possuem muitos tópicos de difícil assimilação por parte dos alunos o que justifica plenamente a utilização de métodos pedagógicos mais coerentes no sentido de se obter o máximo de aproveitamento das aulas. No caso de sistemas operacionais pode-se testar novas instruções sem correr o risco de danificar realmente a máquina. Se em algum instante o teste for desastroso, pode-se simplesmente reiniciar a simulação em um momento antes da falha acontecer. Já no caso de compiladores, pode-se experimentar novas técnicas de compilação e analisar os resultados. Novos circuitos podem ser criados e analisados de forma simples e segura sem custo adicional.

O objetivo deste trabalho foi propor uma nova metodologia de ensino para os cursos de Arquitetura de Computadores, principalmente aqueles voltados para as arquiteturas 64 bits. Nesta proposta se levou em consideração a utilização de simuladores completos como uma ferramenta de apoio pedagógico onde se apresentou, a um custo relativamente baixo, propostas para as aulas teóricas e práticas. Esta proposta esta em fase de finalização e é tema de mestrado desenvolvido no LASB/PCS/EPUSP e possui planos de adoção para o ano de 2005 pela FASP – Faculdades Associadas de São Paulo.

REFERÊNCIAS

- [1]. MINISTÉRIO DA EDUCAÇÃO E CULTURA. *Diretrizes Curriculares de Cursos da Área de Computação e Informática*. Disponível em: <http://www.mec.gov.br/sesu/ftp/curdiretriz/computacao_co_diretriz.rtf>. Acesso em: 20 set. 2003.
- [2]. SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. *Currículo de Referência da SBC para cursos de Graduação em Computação*. Versão 1999. Disponível em: <<http://www.sbc.org.br/educacao/cr99.pdf>>. Acesso em: 20 set. 2003.
- [3]. ACM - ASSOCIATION FOR COMPUTING MACHINERY. *Curricula Recommendations*. Disponível em: <<http://www.acm.org/education/curricula.html>>. Acesso em: 29 set. 2003
- [4]. ELECTRICAL AND ELECTRONIC ENGINEERS (IEEE-CS) & ASSOCIATION FOR COMPUTING MACHINERY (ACM). *Computing Curricula 2001 - Computer Science*. Disponível em: <<http://www.acm.org/sigcse/cc2001/>>. Acesso em: 29 set. 2003
- [5]. MENEZES, P. B.; WEBER, R. F. *Proposta de plano pedagógico para cursos de ciência da computação*. Instituto de Informática da UFRGS. Porto Alegre: 2001. Disponível em: <www.inf.ufrgs.br/~teia/congr_comp/Qualidade2001.pdf>. Acesso em: 29 set. 2003.
- [6]. *Comissão de Especialistas de Ensino de Computação e Informática*. Disponível em: <<http://www.inf.ufrgs.br/mec/>>. Acesso em: 06 out. 2003.
- [7]. UNIREDE. *Universidade Virtual Pública do Brasil*. Disponível em: <www.unirede.com.br>. Acesso em: 21 set. 2003.
- [8]. LARC. *Laboratório de Arquitetura e Redes de Computadores da USP*. Disponível em: <<http://www.larc.usp.br/>>. Acesso em: 26 out. 2003.
- [9]. VIRTUS. *Cursos Virtuais da UFPE*. Disponível em: <<http://www.virtus.ufpe.br/>>. Acesso em: 26 out. 2003.
- [10]. UFPE. *Universidade Federal de Pernambuco*. Disponível em: <<http://www.ufpe.br/>>. Acesso em: 26 out. 2003.
- [11]. FAMATH. *Faculdades Integradas Maria Thereza*. Disponível em: <<http://www.famath.br>>. Acesso em: 26 out. 2003.
- [12]. *Intel Corporation*. Disponível em: <<http://www.intel.com>>. Acesso em: 28 set. 2003.
- [13]. *UCL - University College London - Graduate School*. Disponível em: <<http://www.cs.ucl.ac.uk/teaching/syllabus/curriculum/>>. Acesso em: 25 set. 2003
- [14]. *Stanford University - Graduate school at Stanford*. Disponível em: <<http://cse.stanford.edu/mscs/degree/CScourses.html>>. Acesso em: 15 out. 2003.
- [15]. *UCLA - University of California - Los Angeles. B.S. in Computer Science Curriculum*. Disponível em: <http://www.cs.ucla.edu/csd/cs_catalog.html>. Acesso em: 25 set. 2003
- [16]. *University of Harvard - Division of Engineering and Applied Sciences*. Disponível em: <<http://www.deas.harvard.edu/>>. Acesso em: 25 set. 2003.
- [17]. *MIT - Massachusetts Institute of Technology*. Disponível em: <<http://www.eecs.mit.edu/>>. Acesso em: 25 set. 2003
- [18]. *Navegando na Formação*. Disponível em: <<http://formar.do.sapo.pt/index.html>>. Acesso em: 06 out. 2003.
- [19]. YURCIK, W., et al. *A survey of simulators used in computer organization/architecture courses*. Summer Conference on Computer Simulation. Orlando: July 2001. Disponível em: <<http://www.sosresearch.org/caale/wcae2002webresources.pdf>>. Acesso em: 12 jun. 2003.

- [20]. _____. *Teaching computer organization/architecture with Limited resources using simulators*. Proc. 33rd SIGCSE Technical Symposium on Computer Science Education, Northern Kentucky, February, 2002. Disponível em: <<http://scom.hud.ac.uk/scomhro/Papers/SIGCSE02/csa-resources.pdf>>. Acesso em: 12 jun. 2003.
- [21]. *Universidade Federal de Santa Catarina*. Disponível em: <www.ufsc.br>. Acesso em 15 out. 2003.
- [22]. CANCIAN, R.L.; BARON, S. *Simularq II - Software educacional para o ensino de Arquitetura de Computadores*. Anais do 6º Congresso de Iniciação Científica. São Carlos-SP: 2001. Disponível em: <http://inf.univali.br/~csed/publicacao/sidnei_saocarlos.pdf>. Acesso em: 05 set. 2003
- [23]. *Documentação sobre Simularq - simulador de Arquitetura de computador CISC*. Disponível em <<http://www.inf.ufsc.br/~cancian/arquivos/simuldoc.zip>>. Acesso em: 05 out. 2003
- [24]. *Wine HQ – Site oficial do Wine*. Disponível em <<http://www.winehq.com/>>. Acesso em: 07 out. 2003.
- [25]. *Bochs*. Disponível em: <<http://bochs.sourceforge.net>>. Acesso em: 23 out. 2003.
- [26]. *Guia do Hardware - Tutorial sobre Bochs*. Disponível em: <<http://www.guiadohardware.info/tutoriais/079/>>. Acesso em: 23 out. 2003
- [27]. *The official web site for SimICS*. Disponível em <<http://www.simics.com>>. Acesso em 06 out. 2003.
- [28]. *Virtutech Company*. Disponível em: <<http://www.virtutech.com>>. Acesso em: 1 ago. 2003.
- [29]. *Swedish Institute of Computer Science*. Disponível em: <www.sics.se/SimICS/>. Acesso em: 06 out. 2003.
- [30]. *Dicionário Universal da Língua Portuguesa*. Disponível em: <<http://www.priberam.pt/dlpo/>>. Acesso em: 25 ago. 2003.
- [31]. MAGNUSSON, P., et al. *SimICS: A full system simulation platform*. In *IEEE Computer*, volume 35, n. 2, pages 50–58. 2002.
- [32]. MATTOS, N.P., RASKIN, S.F. *Aspectos da Arquitetura de processadores RISC*. Companhia de Informática do Paraná – CELEPAR. Disponível em: <<http://www.pr.gov.br/batebyte/edicoes/1994/bb35/aspectos.htm>>. Acesso em: 20 out. 2002.
- [33]. *IBM - International Business Machines Corporation*. Disponível em: <<http://www.ibm.com/us/>>. Acesso em: 28 set. 2003.
- [34]. *AMD - American Micro Devices*. Disponível em: <<http://www.amd.com>>. Acesso em: 28 set. 2003.
- [35]. *HP - Hewlett-Packard Company*. Disponível em: <<http://www.hp.com>>. Acesso em: 28 set. 2003.
- [36]. ZELENOVSKY, R.; MENDONÇA, A. *Arquitetura de 64 bits da AMD (x86-64)*. Disponível em: <<http://www.clubedohardware.com.br/x86-64.html>>. Acesso em: 14 jun. 2003.
- [37]. *Intel Itanium Processor & IA-64 Architecture*. Disponível em: <<http://developer.intel.com/design/ia64/>>. Acesso em: 25 fev. 2003
- [38]. OLIVEIRA, R.A. et al. *Arquitetura superescalar*. Universidade Estadual de Londrina. Londrina-PR. 2002. Disponível em <<http://www.superescalar.hpg.ig.com.br/final.htm>>. Acesso em: 14 out. 2003.
- [39]. ZELENOVSKY, R.; MENDONÇA, A. *Arquitetura de 64 bits da Intel*. Disponível em: <<http://www.clubedohardware.com.br/ia64.html>>. Acesso em: 14 jun. 2003.
- [40]. BEREZAGA, M.D. *Arquitetura IA-64 (EPIC)*. Disponível em: <<http://www.inf.ufrgs.br/procp/par/disc/inf1191/trabs/992/IA64/ia64.html>> Acesso em: 14 out. 2003.
- [41]. LOURENÇO, A.E. *Ensino de Arquitetura de Computadores Utilizando Simuladores Completos*. 2004. 101f. Dissertação (Mestrado em Engenharia Elétrica) – PCS/POLI. Universidade de São Paulo, São Paulo, 2004.

TEACHING COMPUTERS ARCHITECTURE USING COMPLETE SIMULATORS

Abstract: *The curriculum of the Engineering and Computer Science courses, taught in the universities have, among others the subject of Computer Architecture whose objective is to present to the student the computer a cluster of electronic components, showing its operation in the level of machine language and assembly language. This allows the student to understand all the abstraction levels that are made and are present in an computational environment better. However due to the high degree of complexity of the Computer Architecture, the level of the students' understanding is rather low, and it takes a lot of time to a learning deficiency, and the most serious, this deficiency will be spread during whole the course.*

Due to such situation, what would possible be to alter or to modify (in the curricula of the subject), without it there are inclusion loss and content quality? In other words, a mechanism, or a pedagogic method it will be possible, in what the student of Computers Architecture could see the execution of the instructions in the level of registers, or still, to visualize the transfer of data of the main memory to the cache memory?

The objective of this project is to propose a new teaching methodology for the Computer Architecture course, mainly to the 64-bit architectures. In this proposal it will be taken in consideration the use of complete simulators as a tool of pedagogic support where it will come, at a cost relatively low, proposed for the theoretical and practical classes.

Key-Word: *Computer Engineering, Computer Architecture, Pedagogical process, Simulation, Curriculum.*