

Prof.: Herbert Oliveira Rocha

Aluno: Carlos Alberto Vieira Marques Filho

Trabalho de Pesquisa sobre diferentes arquiteturas e quais as diferenças entre a arquitetura ARM e x86.

Na sua criação nem todas as arquiteturas são iguais e existem diversos fatores que influenciam nas suas peculiaridades. Algumas são pequenas e muito eficientes em termos energéticos, outras tem o baixo custo de produção alguns são mais adequados à fabricação com matérias resistentes a radiação, como arsenieto de gálio (em vez de silício). Existem também arquiteturas ótimas para suportar a velocidades de clock muito altas.

Por exemplo: O mnemônico NOP (sem operação) em um processador 80x86 é muito mais fácil de ler do que o opcode 0x90 armazenado na memória e que é lido e executado pelo processador. Um mnemônico é traduzido pela montadora em uma única instrução de máquina.

Devido às diferentes arquiteturas, os métodos utilizados nos mnemônicos (bem como no código da máquina) são, por exemplo, muito diferentes ao endereçar endereços de memória em processadores que possuem barramento de dados de diferentes larguras.

Para complicar as coisas, a Intel uma vez (no 8086, com registradores de 16 bits, mas uma largura de barramento de dados externa de 20 bits) introduziu o modelo de memória segmentada que precisa usar registros de processador combinados para indicar um endereço físico na memória (e que pode tem várias combinações que se referem ao mesmo endereço): você precisa de um registro para endereçar uma área de memória contígua de 65536 (2^{16}) bytes de comprimento, mas os outros 4 bits (deslocamento) precisam vir de outro registro para completar o endereço de 20 bits para o barramento de dados externo.

O layout de endereço de memória "segmentado" acima introduziu o formulário de endereço como 0x40: 0x7F, que não é encontrado nas instruções da máquina para os processadores onde os registros têm a mesma largura que o barramento de dados (como o Motorola 68000 antigo, todos os 32 bits).

ARM vs x86

Os responsáveis pela explosão de sucesso do ARM são os smartphones e os tablets. Com isso, algumas corporações estão suscitando a ideia de que a novidade pode ser interessante na substituição da atual arquitetura que reina no mercado dos desktops.

Os processadores x86 são baseados na arquitetura CISC (Complex Instruction Set Computers), que, como a própria sigla sugere, é voltada para instruções complexas. O termo “complexo” aqui serve tanto para designar a parte de trabalho do componente quanto a própria disposição dos elementos e a forma como tudo funciona ali dentro.

O hardware de um chip como o Intel Core i7 ou o AMD Ryzen 7 é composto por diversos itens: núcleos, controlador de memória, interface de entrada e saída, memória cache e outros.

No entanto, pensando basicamente em um esquema ilustrativo simplificado, conforme proposto pela Universidade de Stanford, uma CPU pode ser separada em memória, registros e unidade de execução. A memória contém os dados básicos, que, através de uma instrução, serão carregados de forma temporária no registro e computados na unidade de execução.

O grande detalhe desse tipo de processador é que o software não precisa detalhar instruções de forma separada e a unidade de execução pode realizar múltiplos processos com uma única instrução. Assim, pensando no exemplo de cálculos matemáticos, um processador CISC pode realizar várias contas através de uma única linha de comando e dar um retorno imediato.

Pensando em um sentido amplo, a arquitetura CISC permite realizar tarefas complexas sem que o programa tenha de se desdobrar em inúmeras partes, isso garante um menor consumo de memória RAM para armazenamento das funções. Vale notar que a arquitetura diretamente do hardware nesses chips já é voltada para essa alta complexidade.

É importante observar que boa parte do trabalho pesado aqui é feita diretamente no hardware, que pode trabalhar com diferentes configurações de clock, usar códigos menores (o que beneficia os desenvolvedores) e operar com ciclos maiores por cada segundo.

Os processadores ARM, por outro lado, são baseados na arquitetura RISC (Reduced Instruction Set Computer), que, conforme você pode imaginar, opta pela simplicidade. O hardware desses componentes também é bastante complexo e um olhar mais amplo pode até confundir os usuários menos atentos, pois há muitos itens similares aos dos chips x86.

Os processadores mais avançados, como os que são baseados na arquitetura Cortex-A75, trazem núcleos equipados com memória cache de diferentes níveis (com capacidades próximas às que temos em processadores x86), unidade de ponto flutuante, controlador de periféricos e outros dispositivos que garantem operações robustas em sistemas completos.

Novamente, baseado em um esquema mais simples — com memória, registro e unidade de execução —, a forma de operação é um pouco diferente. A memória ainda contém os dados básicos que serão carregados no registro e computados na unidade de execução, porém, em vez de usar uma única instrução, uma CPU RISC precisa de várias.

Ao contrário da CISC, a RISC requisita muitos comandos vindos do software, cada qual detalhando uma operação. Apesar de parecer ineficiente, isso permite que cada tarefa seja executada de forma mais rápida, já que são instruções mais simples.

Pensando em um mesmo cálculo para as duas unidades, uma RISC poderia dar um retorno quase que no mesmo tempo de uma CISC, com a diferença de que apenas necessitaria mais linhas de comando. Outra vantagem é que a CISC mantém as instruções no registro, as quais podem ser usadas posteriormente em outras tarefas, então isso pode gerar alguma economia de energia e processamento.

A desvantagem é que o programa precisa ser separado em várias partes para dar o mesmo resultado, o que gera um maior consumo de memória RAM. O hardware um pouco mais simples aqui deve ser compensado pelo software, que precisa ter mais sofisticação.

Assim, ao contrário da CISC, boa parte do trabalho é adiantada pelas linhas de código no software. Além disso, essas CPUs geralmente são programadas para trabalhar com configurações únicas de clock, usam códigos maiores e operam com menores ciclos por segundo.

É preciso entender que assim como existem veículos com funcionamentos distintos para atividades específicas. Não faz sentido usar um avião para se deslocar diariamente por 10 ou 20

quilômetros, assim como não é lógico apostar em um carro como meio de locomoção para ir do Brasil até a Europa.

Da mesma forma, há processadores com arquiteturas diferentes voltados para funções únicas. É claro que um processador Intel Core pode realizar as atividades básicas de um celular, mas o gasto de energia, as altas temperaturas e também o espaço necessário para instalar esse componente em um smartphone inviabiliza esse tipo de aplicação.

Por outro lado, usar um Snapdragon 810 em um PC também não faria sentido. Em teoria, com as devidas adaptações de código, ele até poderia renderizar um vídeo no Adobe Premiere, porém levaria uma eternidade para finalizar a tarefa. Durante todo o processo, esse componente usaria pouca energia e não aqueceria quase nada, mas não seria uma aplicação inteligente.

Basicamente, um processador ARM é programado para atividades focadas em tarefas simples e que necessitem de um resultado imediato. Controlar um player de música, fazer cálculos na calculadora, gerenciar os alarmes, cuidar de um sistema leve ou executar alguns poucos apps em paralelo pode ser moleza para um processador RISC.

Além disso, essa questão da energia e temperatura são importantíssimas em um celular, que precisa ter bateria para durar o dia todo e se manter frio para evitar danos ao usuário. Dessa forma, as fabricantes mantêm tais componentes pequenos, eficientes, focados e adicionam novidades que possam incrementar a experiência no mundo mobile.

Agora, quando tratamos de um processador x86, estamos falando de atividades bem mais complexas, mas que ainda necessitem de resultados imediatos (ou que demorem um mínimo de tempo possível, mesmo que com alto custo de energia).

No caso dos processadores CISC, estamos pensando em um uso muito mais intenso. Aqui, o usuário pode usar um navegador similar ao do celular, mas esse processador estará pronto para processar 30 abas simultaneamente se for necessário, com um layout maior, com divisão de tela, com gerenciamento de janelas, enquanto realiza sincronização de múltiplos apps em segundo plano e dá conta de processar jogos e apps muito mais pesados.

É claro que o consumo e a temperatura vão para as alturas, mas é bom pensar que estamos tratando de máquinas bem maiores, como laptops (que trazem baterias de alta capacidade) e desktops que são conectados diretamente à tomada. Não é por acaso que agora temos processadores com mais de dez núcleos, grandes, ainda eficientes (dentro dessa proposta), robustos para executar milhares de threads e cada vez mais poderosos.

Então, um x86 jamais poderá substituir um ARM e um ARM nunca estará apto para um computador? Bom, a verdade é que a Intel até já tentou emplacar processadores x86 no mundo mobile (como é o caso do Intel Atom z3580, que foi instalado no Zenfone 2), bem como já ouvimos planos de ARM para computadores.

O Android até tem uma versão para x86, sendo um sistema que pode ser perfeitamente escalado para laptops (o próprio Chromebook usa processadores ARM e entrega uma experiência bem completa e similar à do Windows). Todavia, são coisas que não pegam com facilidade e dificilmente vão virar regra em mercados já estabelecidos.

Em teoria, com as devidas adaptações de software, qualquer um desses processadores pode substituir o outro nas principais tarefas, mas isso nem sempre compensa. Assim, em teoria, um ARM pode fazer o mesmo que um x86, mas, em um comparativo, ele nunca terá o mesmo desempenho.

Referência bibliográfica

<https://www.quora.com/Why-dont-all-computers-use-the-same-architecture>

<https://www.tecmundo.com.br/produto/119693-diferenca-processador-arm-um-x86.htm>