

Família e Evolução dos Processadores

Objetivo da Aula

Abordar as características das diferentes famílias de processadores e as principais tecnologias inseridas para a aumentar sua performance.

Apresentação

Olá!

Agora que você já conhece os componentes do processador e sua participação na execução das instruções, é hora de vermos todas as inovações que vêm sendo agregadas aos processadores ao longo dos anos.

Vamos entender o que são diferentes famílias de processadores e aprender como a inserção de unidades de ponto flutuante, *pipelining*, predição de desvios e outras tecnologias permitiram o aumento do desempenho dos processadores.

O assunto é bastante interessante e você verá como os projetistas conseguem a cada geração melhorar o que já parecia ótimo.

Mãos à obra!

1. Introdução

A presença de computadores nos mínimos detalhes do nosso dia a dia é fato mais do que conhecido por todos nós. Além disso, a demanda por altas capacidades de computação em áreas como a indústria, medicina, agricultura, educação, entre outras, faz com que a evolução dos computadores, e em especial dos processadores, siga um ritmo muito rápido e difícil de acompanharmos.

Vamos abordar aqui a evolução dos processadores, apresentando algumas das principais tecnologias que influenciaram decisivamente em sua velocidade e capacidade de executar tarefas de altíssima complexidade, inimagináveis há algumas décadas.

2. ISA e Famílias de Processadores

Para começar, vamos entender o que é o ISA (*Instruction Set Architecture* – Arquitetura do conjunto de instruções).

O ISA é parte de um modelo abstrato de um computador que define como o processador é controlado pelo software. Essa arquitetura age como uma interface entre o software e o hardware, especificando o que o processador consegue interpretar e executar. Em outras palavras, o ISA define como as instruções básicas são codificadas (representadas na forma binária) para que possam ser entendidas pelo processador. Esse conjunto de instruções pode ser visto como o “vocabulário” que o processador entende.

Desta definição podemos entender que processadores com conjuntos de instruções diferentes fazem parte de *famílias* diferentes.

Todo processador é capaz de realizar as mesmas operações, por exemplo, somas, subtrações, comparações etc., porém, essas operações são codificadas de forma diferente em famílias diferentes.

Para facilitar o entendimento, vamos fazer uma analogia. As palavras “água” e “*water*” significam a mesma coisa, respectivamente em português e inglês. Porém, são *codificadas* de formas diferentes. O mesmo acontece com os processadores. Enquanto numa determinada família uma soma de inteiros é codificada por uma determinada sequência de bits, em outra família a mesma soma é codificada por outra sequência de bits. Mas o que importa é que em ambas o resultado de $2+2$ é igual a 4.

A definição do ISA por parte dos projetistas leva em consideração aspectos diversos, tais como o tipo de aplicação que tal família de processadores deve atender.

3. Evolução dos Processadores

Os processadores, desde seus primórdios, vêm evoluindo como qualquer outro equipamento. Essa evolução passa pela tecnologia de fabricação, matéria-prima utilizada, capacidade de clock, arquitetura do conjunto de instruções e otimização dos recursos presentes.

Uma das principais formas de melhorar o desempenho do processador é aumentar a frequência do clock, mas isso não é simples e provoca o aumento da geração de calor e consumo de energia.

Vamos a partir de agora analisar algumas das tecnologias que permitiram verdadeiros saltos na capacidade e velocidade de processamento.

4. Coprocessadores Aritméticos

Figura 1: Coprocessador aritmético



Fonte: <https://pt.wikipedia.org/wiki/Coprocessador>.

Os primeiros processadores fabricados pela Intel e que equipavam os computadores domésticos, que pertenciam à família x86, operavam basicamente com números inteiros. Porém, algumas funções matemáticas envolviam cálculos complexos envolvendo números não inteiros (ponto flutuante) e tais processadores efetuavam essas operações de forma lenta.

Surgiu então a figura do coprocessador aritmético que tinha a função de auxiliar o processador a executar esses cálculos de forma muito mais rápida.

Inicialmente o coprocessador era vendido como um acessório e era encaixado na placa mãe. Por ser muito caro, era pouco usado pelos usuários comuns.

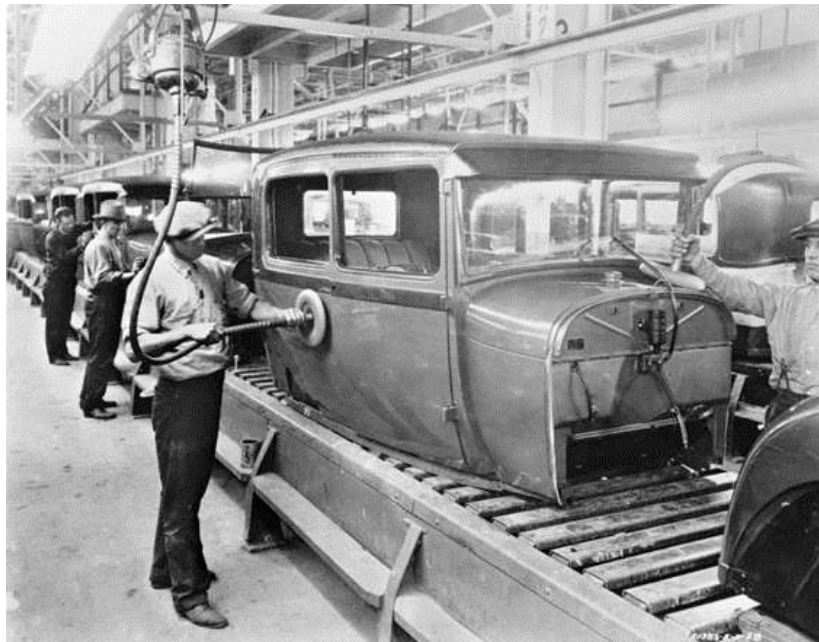
A partir da geração do 486DX o coprocessador foi incorporado no núcleo do processador, o que permitiu sua produção em maior escala e o barateamento dos preços.

5. Pipelining

Esta tecnologia é baseada na ideia de linha de montagem das indústrias, criada por Henry Ford.

De forma resumida, a linha de montagem fez com que os carros, que eram construídos artesanalmente, passassem a ser construídos numa sequência de passos, cada um deles executado por um grupo de profissionais, permitindo que vários carros estivessem sendo simultaneamente montados, cada um em um estágio diferente.

Figura 2: Linha de montagem



Fonte: <https://media.ford.com/content/fordmedia/fsa/br/pt/news/2016/02/11/ford-britanica-comemora-113-anos-com-mostra-vintage-de-fotos-e-s.html>.

Através da técnica de pipeline, os processadores passaram a executar instruções semelhantemente a uma linha de montagem. Cada instrução buscada começa seu ciclo e mesmo antes de acabar, outras instruções vão sendo buscadas, de forma que em um dado momento o processador está atendendo a mais de uma instrução, cada uma delas em um estágio diferente.

Figura 3: Execução de instruções sem pipeline e com pipeline

B	D	BO	EX	ER	B	D	BO	EX	ER	B	D	BO	EX	ER
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

(a) – execução sem pipeline

B	D	BO	EX	ER		
	B	D	BO	EX	ER	
		B	D	BO	EX	ER
1	2	3	4	5	6	7

(b) – execução com pipeline

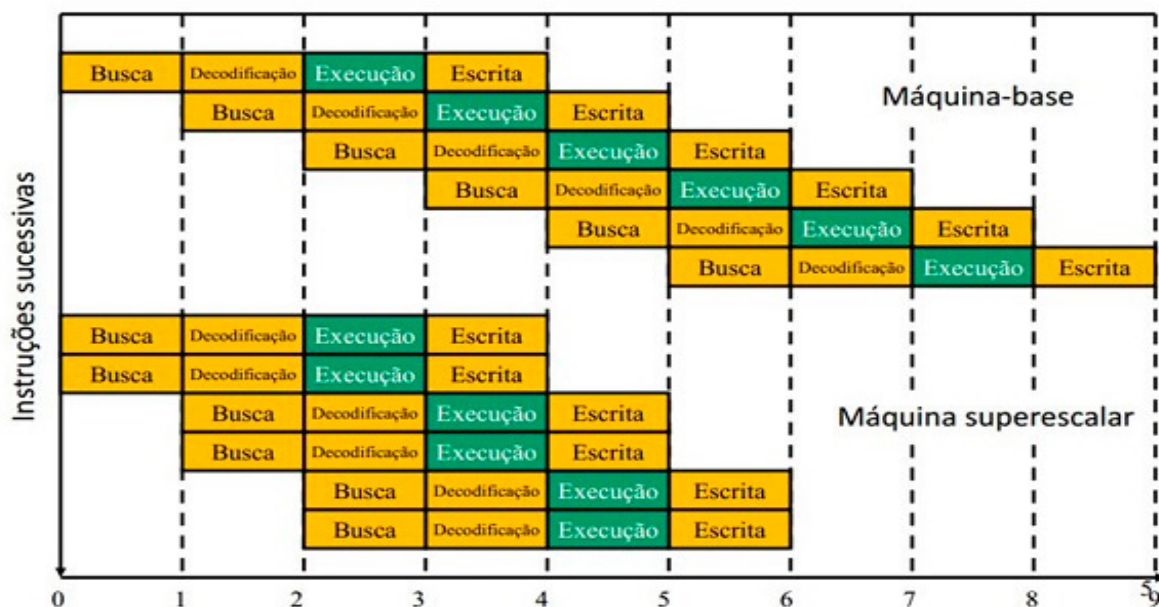
Fonte: elaborado pelo autor.

Como podemos ver na Figura 3, uma sequência com três instruções gastaria quinze ciclos para ser concluída sem a utilização do pipeline. Já a mesma sequência de instruções gastaria sete ciclos para ser concluída utilizando pipeline.

6. Arquitetura Superescalar

As arquiteturas superescalares possibilitam que diferentes instruções sejam executadas simultaneamente em diferentes linhas de pipeline, pois permitem que se iniciem múltiplas instruções no mesmo ciclo de clock.

Figura 4: Modelo de arquitetura superescalar



Fonte: https://wiki.ifsc.edu.br/mediawiki/index.php/AULA_15_-_Microprocessadores_-_Gradua%C3%A7%C3%A3o.

Na Figura 4 temos a comparação de uma arquitetura simples e uma superescalar, onde todos os estágios do pipeline são duplicados.

7. Processadores Multicore

Os processadores multicore (múltiplos núcleos) são aqueles que possuem mais de um núcleo de processamento. O sistema operacional enxerga esses processadores como se fossem vários processadores de um núcleo (singlecore). Isso torna possível a busca e execução de dois ou mais fluxos de instruções completamente independentes entre si.

A grande limitação que as arquiteturas multicore enfrentam é a questão do acesso à memória. Embora os diversos núcleos possam ter suas caches internas, o acesso à memória principal é feito pelo mesmo barramento, o que configura uma espécie de gargalo.

Figura 5: Arquitetura Multicore



Fonte: https://www.researchgate.net/figure/Figura-2-modelo-de-arquitetura-multicore_fig2_282367700.

Observando a Figura 5, vemos que o barramento frontal (*frontal bus*), que liga o processador à memória principal na placa mãe, é o mesmo para todos os núcleos e isso deve ser gerenciado na busca das instruções que não estiverem nas caches internas dos núcleos.

8. Predição de Desvios com Execução Especulativa de Instruções

Um dos principais motivos de atrasos no pipeline é a alta quantidade de ciclos que algumas instruções gastam. Entre essas estão as *instruções de desvio*.

Instruções de desvio são aquelas que testam uma condição e, dependendo do resultado deste teste, podem mudar o fluxo de instruções. Porém, esses testes são demorados e podem atrasar muito o início das instruções subsequentes.

A solução para minimizar esse problema é o processador executar especulativamente um dos fluxos de instruções possíveis se antecipando ao resultado do teste e guardando os resultados em registradores provisórios. Caso o fluxo executado seja aquele definido pelo resultado do teste, ótimo: o processador ganhou tempo e os resultados provisórios são tornados definitivos. Caso contrário, todos os resultados provisórios são descartados e o fluxo correto é executado.

Considerações Finais

Chegamos ao fim dessa aula em que abordamos as famílias de processadores e sua evolução.

A evolução dos processadores não se resume a essas tecnologias que apresentamos aqui.

Muitas outras ideias vêm sendo incorporadas e um estudo mais aprofundado delas fugiria ao escopo desse curso.

Em todas elas a principal diretriz é explorar ao máximo os recursos que o processador já tem, utilizando-os de forma mais eficiente.

Elas permitem que a performance seja melhorada sem encarecer tanto o processador.

Como vimos no início, o aumento do clock também é um fator de melhoria de desempenho, mas como a velocidade dos sistemas de armazenamento não acompanha a velocidade do processador, essa melhoria acaba sendo prejudicada.

Vimos que os processadores têm uma espécie de vocabulário, que é a forma como suas instruções básicas são codificadas. A esse vocabulário chamamos de Arquitetura do Conjunto de Instruções. Portanto, as diferentes famílias de processadores são caracterizadas por diferentes arquiteturas do conjunto de instruções.

Vimos também que diversas tecnologias vêm sendo incorporadas aos processadores para melhorar sua performance e abordamos os coprocessadores aritméticos, *pipelining*, arquiteturas superescalares, arquiteturas multicore e predição de desvios.

Agora que vimos os dois primeiros subsistemas do computador, a memória e o processador, é hora de estudarmos os dispositivos de entrada e saída, que facilitam nossa comunicação com o processador.

Até lá!

Material Complementar



Vídeo

Neste vídeo você poderá saber mais um pouco sobre os processadores multicore: <https://www.youtube.com/watch?v=aje1Kj-jXOY>.

Referências

STALLINGS, William. *Arquitetura e Organização de Computadores: projeto para o desempenho*. 8ª edição. Editora Pearson. Livro. (642 p.). ISBN 9788576055648.

TANENBAUM, Andrew S. *Sistemas Operacionais Modernos*. 3ª edição. Editora Pearson. Livro. (674 p.). ISBN 9788576052371.

TANENBAUM, Andrew S. *Organização estruturada de computadores*. 6ª edição. Editora Pearson. Livro. (628 p.). ISBN 9788581435398.