

Desempenho

Desempenho: uma analogia

- O que é desempenho ?

Airplane	Passenger capacity	Cruising range (miles)	Cruising speed (m.p.h)	Passenger throughput (passenger x m.p.h)
Boeing 777	375	4630	610	228.750
Boeing 747	470	4150	610	286.700
BAC/Sud Concorde	132	4000	1350	178.200
Douglas DC-8-50	146	8720	544	79.424

Desempenho: métricas

- **Comparação entre desktops**
 - Qual deles termina a tarefa primeiro ?
 - Tempo para realizar uma tarefa
 - » Tempo de execução ou tempo de resposta
- **Comparação entre servidores**
 - Qual deles completou mais tarefas ?
 - Número de tarefas na unidade de tempo
 - » Throughput

Desempenho relativo

- "X é n vezes mais rápido que Y" significa:

$$n = \frac{\text{TempoEx}(Y)}{\text{TempoEx}(X)} = \frac{\text{Desempenho}(X)}{\text{Desempenho}(Y)}$$

Medida de desempenho

- **Tempo de execução de um programa**
 - Medido em segundos
- **Tempo de resposta**
 - Tempo total para completar uma tarefa
 - » CPU + memória + HD + E/S
 - » Compartilhamento: programas simultâneos
 - Tempo de execução de CPU
 - » $\text{tempo}_{\text{CPU}} = \text{tempo}_{\text{usuário}} + \text{tempo}_{\text{sistema}}$

Medida de desempenho

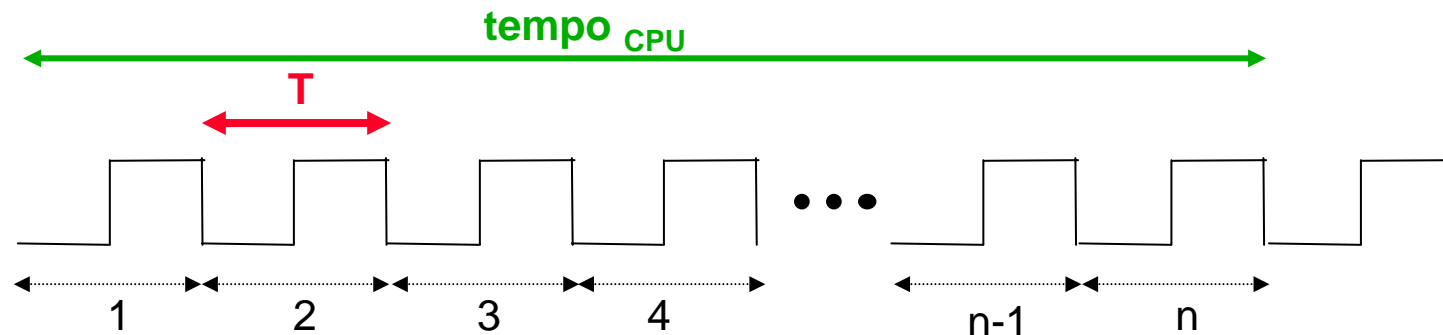
- **Desempenho do sistema**
 - Tempo **de resposta** de um sistema sem carga
- **Desempenho de CPU**
 - Tempo **de CPU** dedicado a um dado usuário
 - Foco deste curso

Desempenho: perspectivas

- **Perspectiva do usuário**
 - Quão rápido um programa executa ?
 - Métrica: tempo
- **Perspectiva do projetista**
 - Quão rápido o HW executa funções básicas
 - Métrica: ciclos de relógio

O relógio

- Computador é um sistema digital síncrono
- Relógio
 - Determina quando ocorrem eventos no HW
 - Período (T) ou frequência (f)
 - » Exemplo: $T = 0.25\text{ns} = 250\text{ ps}$ ou $f = 4\text{ GHz}$



$$\text{tempo}_{\text{CPU}} = n \times T = \frac{n}{f}$$

Relacionando as perspectivas

$$\text{tempo}_{\text{CPU}} = \text{ciclos}_{\text{CPU}} \times T = \frac{\text{ciclos}_{\text{CPU}}}{f}$$

Exemplo

- **Problema**

- 10s (comp. A, 4 GHz)
- 6s (comp. B, ?), mas $\text{ciclos}_B = 1,2 \times \text{ciclos}_A$

- **Solução**

$$\text{tempo}_{\text{CPU}} = \text{ciclos}_{\text{CPU}} \times T = \frac{\text{ciclos}_{\text{CPU}}}{f}$$

Exemplo

- **Problema**

- 10s (comp. A, 4 GHz)
- 6s (comp. B, ?), mas $\text{ciclos}_B = 1,2 \times \text{ciclos}_A$

- **Solução**

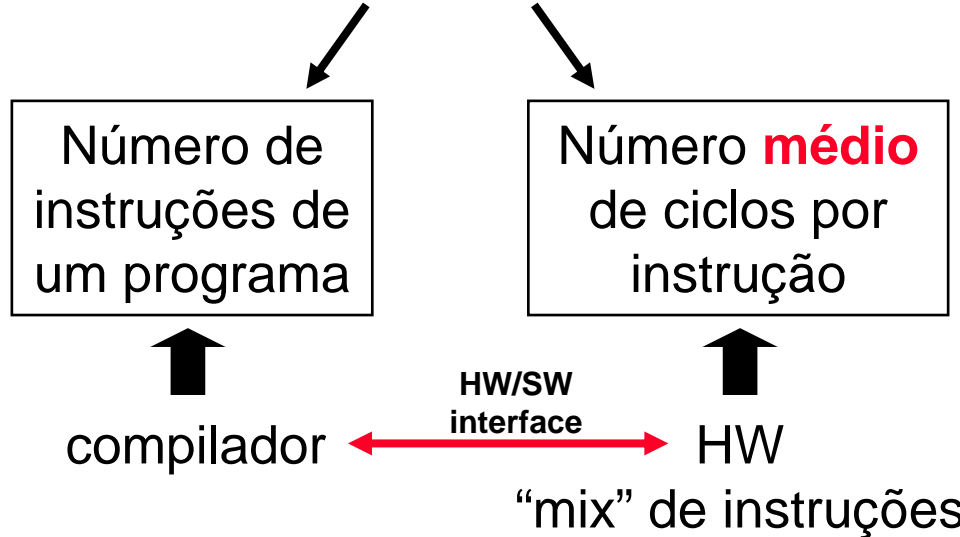
$$\text{ciclos}_{\text{CPU}}(A) = \text{tempo}_{\text{CPU}}(A) \times f(A) = 40 \times 10^9$$

$$\text{tempo}_{\text{CPU}}(B) = \frac{1.2 \times \text{ciclos}_{\text{CPU}}(A)}{f(B)} = \frac{1.2 \times 40 \times 10^9}{f(B)} = 6$$

$$f(B) = \frac{1.2 \times 40 \times 10^9}{6} = 8 \text{ GHz}$$

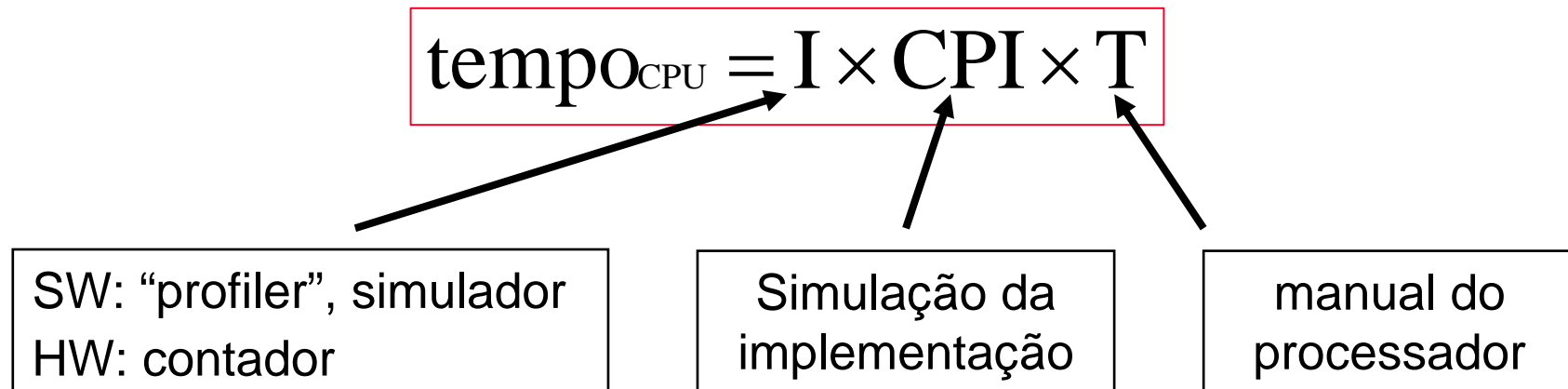
Refinando a modelagem

$$\text{ciclo}_{\text{CPU}} = I \times \text{CPI}$$



$$\text{tempo}_{\text{CPU}} = I \times \text{CPI} \times T = \frac{I \times \text{CPI}}{f}$$

Como determinar cada fator ?



$$\text{ciclos}_{\text{CPU}} = \sum_{i=1}^n (I_i \times \text{CPI}_i)$$

(Exemplo: p. 35-36)

Adapted from “Computer Organization & Design: The Hardware/Software Interface”, D. Patterson and J. Hennessy, Morgan Kaufmann Publishers. Copyright 1998 UCB.

Cálculo do CPI médio

$$\text{CPI} = \frac{\sum_{i=1}^n I_i \times \text{CPI}_i}{I} \quad (\text{média ponderada dos CPIs})$$

$$\text{CPI} = \sum_{i=1}^n \left(\frac{I_i}{I} \times \text{CPI}_i \right) = \sum_{i=1}^n (F_i \times \text{CPI}_i)$$

Classe	CPI	Qde.	Fração
A	5	20M	0,33
B	2	30M	0,50
C	4	6M	0,10
D	4	4M	0,07

$$\text{CPI} = \frac{5 \times 20 + 2 \times 30 + 4 \times 6 + 4 \times 4}{60} = \frac{10}{3}$$

$$\text{CPI} = 5 \times 0,33 + 2 \times 0,50 + 4 \times 0,10 + 4 \times 0,07 = 3,33$$

Impacto no desempenho

- **Algoritmo**
 - **Afeta o número de instruções executadas**
 - » **Determina o número de instruções do programa**
 - **Pode afetar o CPI**
 - » **Pode favorecer instruções mais lentas ou rápidas**
 - » **Exemplo: uso de instruções de ponto flutuante ao invés de instruções inteiras**

Impacto no desempenho

- **Linguagem**
 - **Afeta o número de instruções**
 - » Determina as instruções-fonte a serem traduzidas em instruções do processador-alvo
 - **Afeta o CPI**
 - » Forte suporte a **abstrações de dados** requer chamadas indiretas, que deterioram o CPI
 - **Exemplo: Invocação de métodos em Java**
 - » **Seção 2.15:**
 - Teste se ponteiro nulo (beq, bne)
 - Carga do endereço da tabela de métodos (lw)
 - Carga do endereço do método apropriado (lw)
 - Desvio para endereço em registrador (jr)

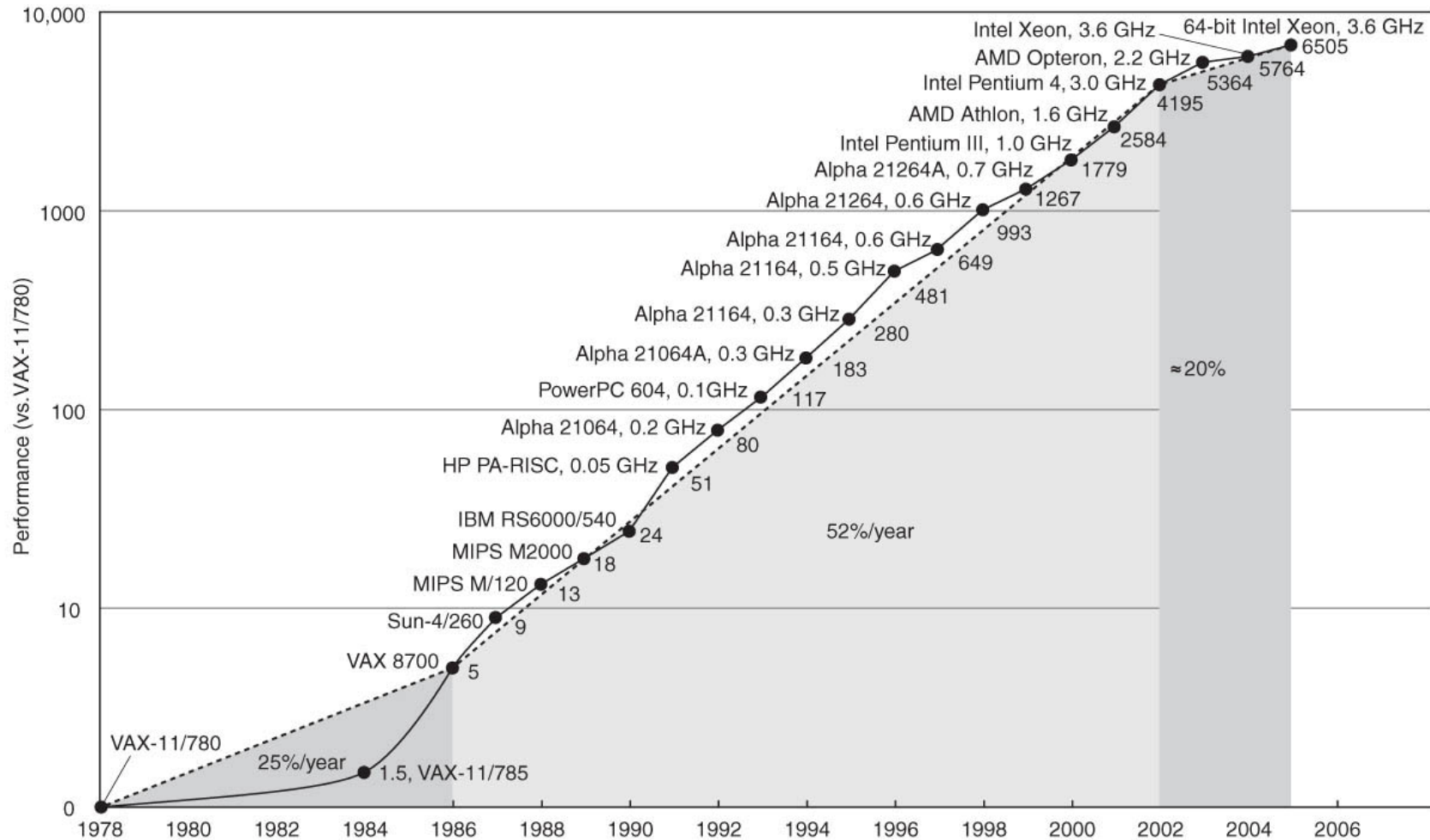
Impacto no desempenho

- **Compilador**
 - **Afeta o número de instruções executadas**
 - » **Determina a tradução de instruções-fonte em instruções do processador-alvo**
 - **Pode afetar o CPI**
 - » **Determina a proporção de instruções de cada tipo**
 - » **Determina a ordem das instruções (pipeline)**
 - » **Influencia a localidade de acesso à memória (cache)**

Impacto no desempenho

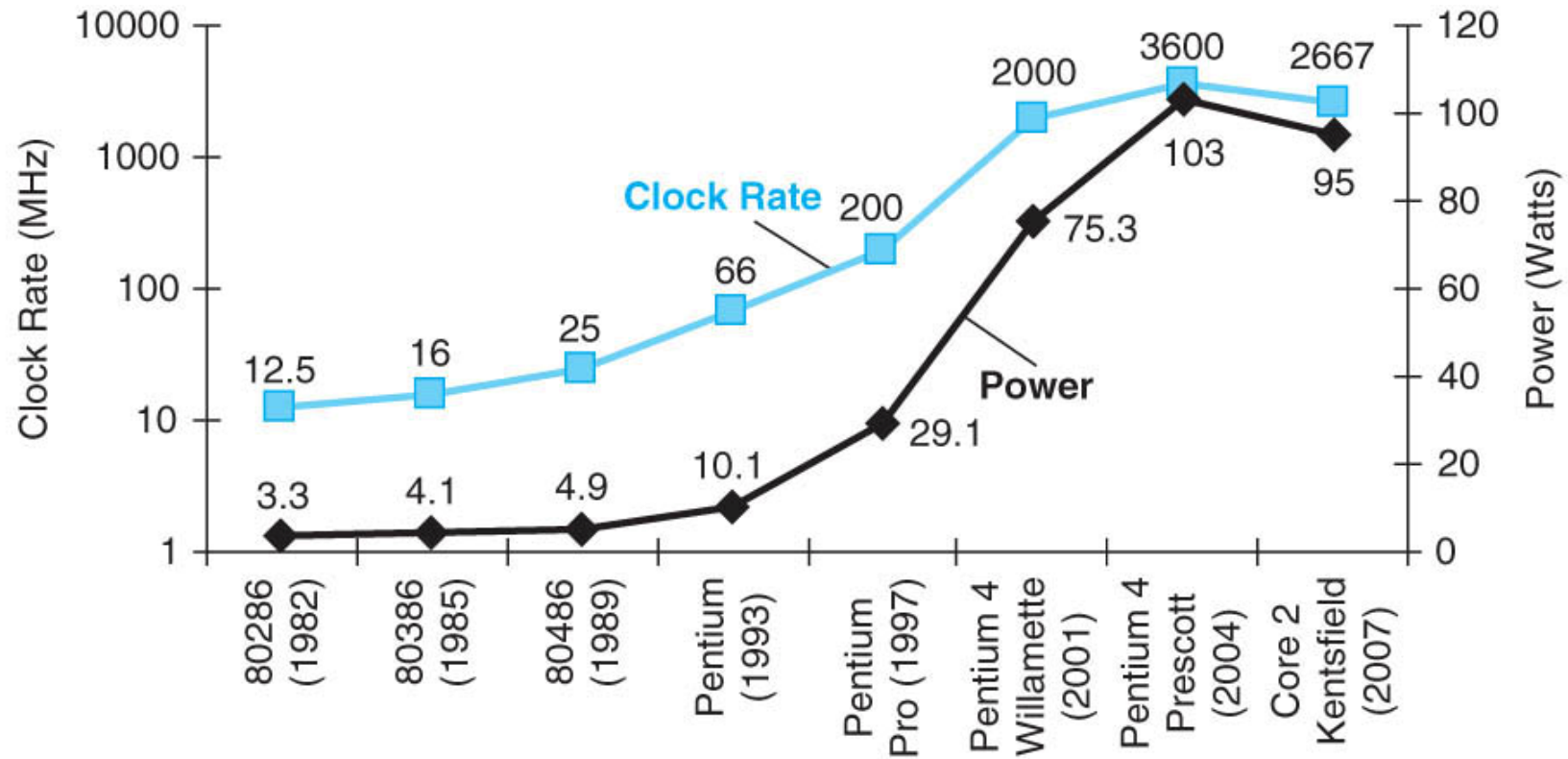
- **ISA**
 - **Afeta o número de instruções executadas**
 - » Afeta a seleção de instruções pelo compilador
 - » Que instruções do processador-alvo são necessárias para executar uma dada função ?
 - **Afeta o CPI**
 - » Determina o custo em ciclos de cada instrução
 - **Afeta a frequência**
 - » A simplicidade das instruções permite organizar o sistema digital com menores período de relógio
 - » Para uma dada tecnologia de fabricação

Limites ao crescimento do desempenho



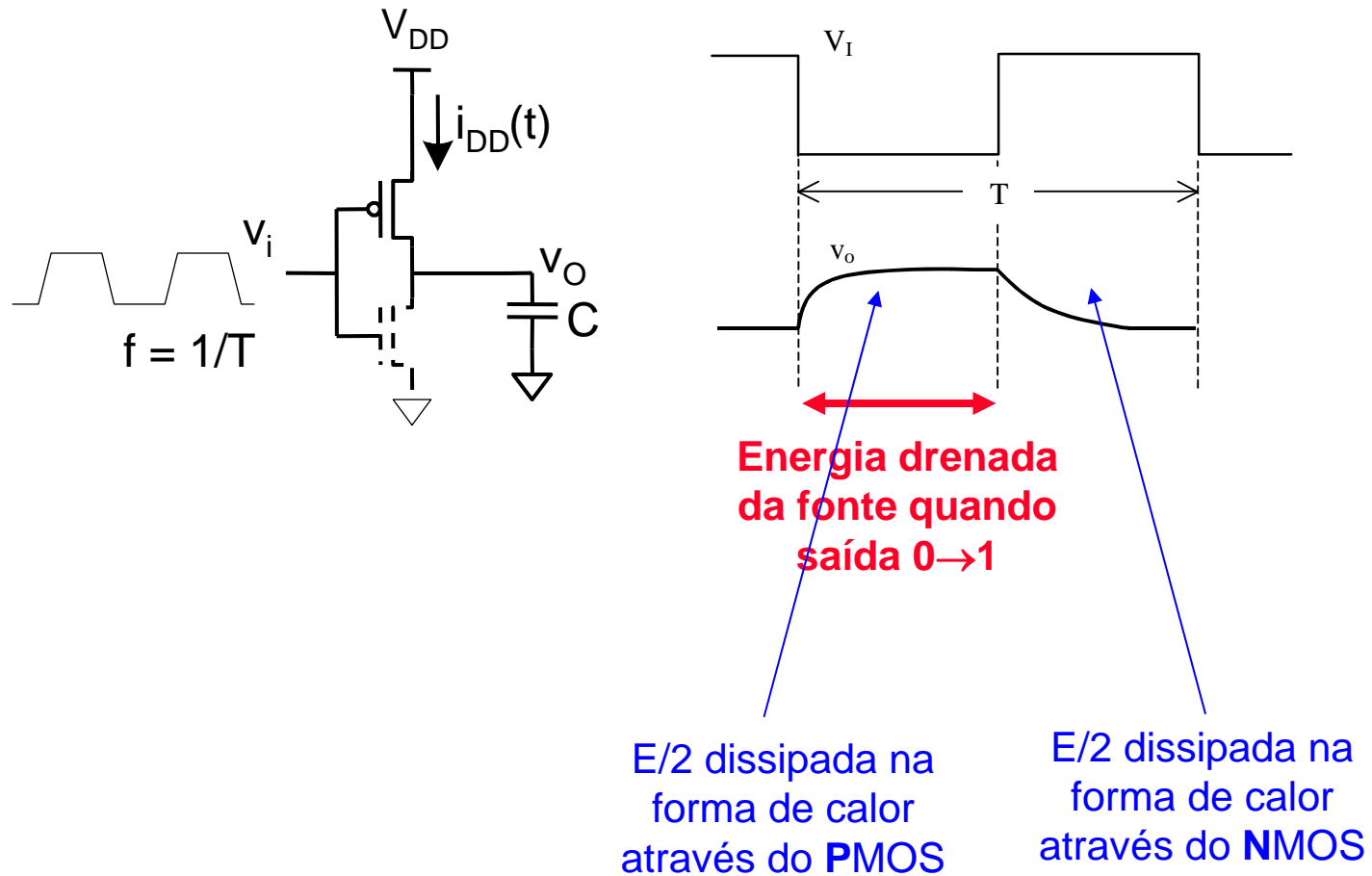
Redutores do crescimento: barreira de potência, limite de paralelismo entre instruções, latência no acesso à memória

A barreira da potência

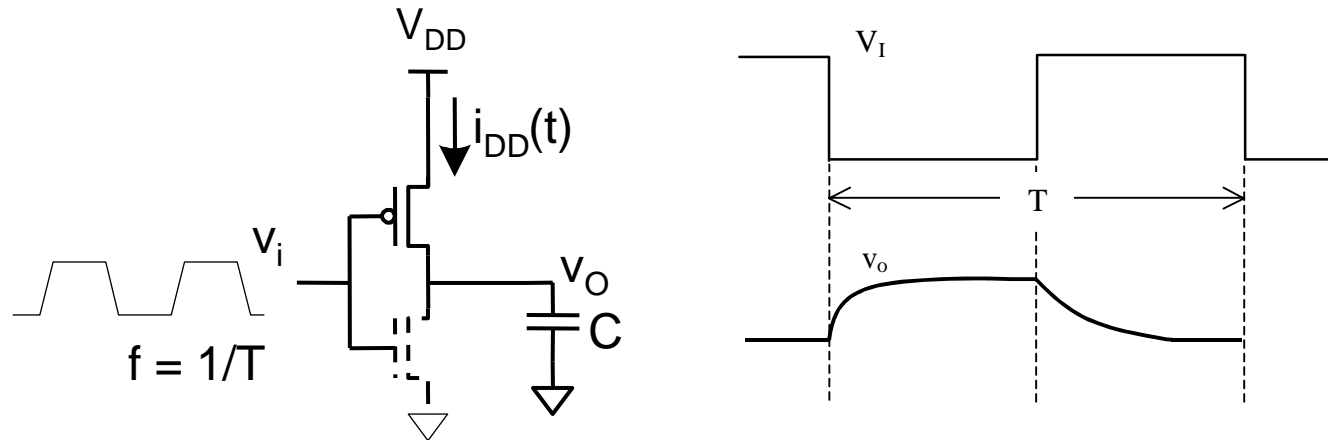


Qual a relação entre P e f ?

Revisão de tecnologia CMOS

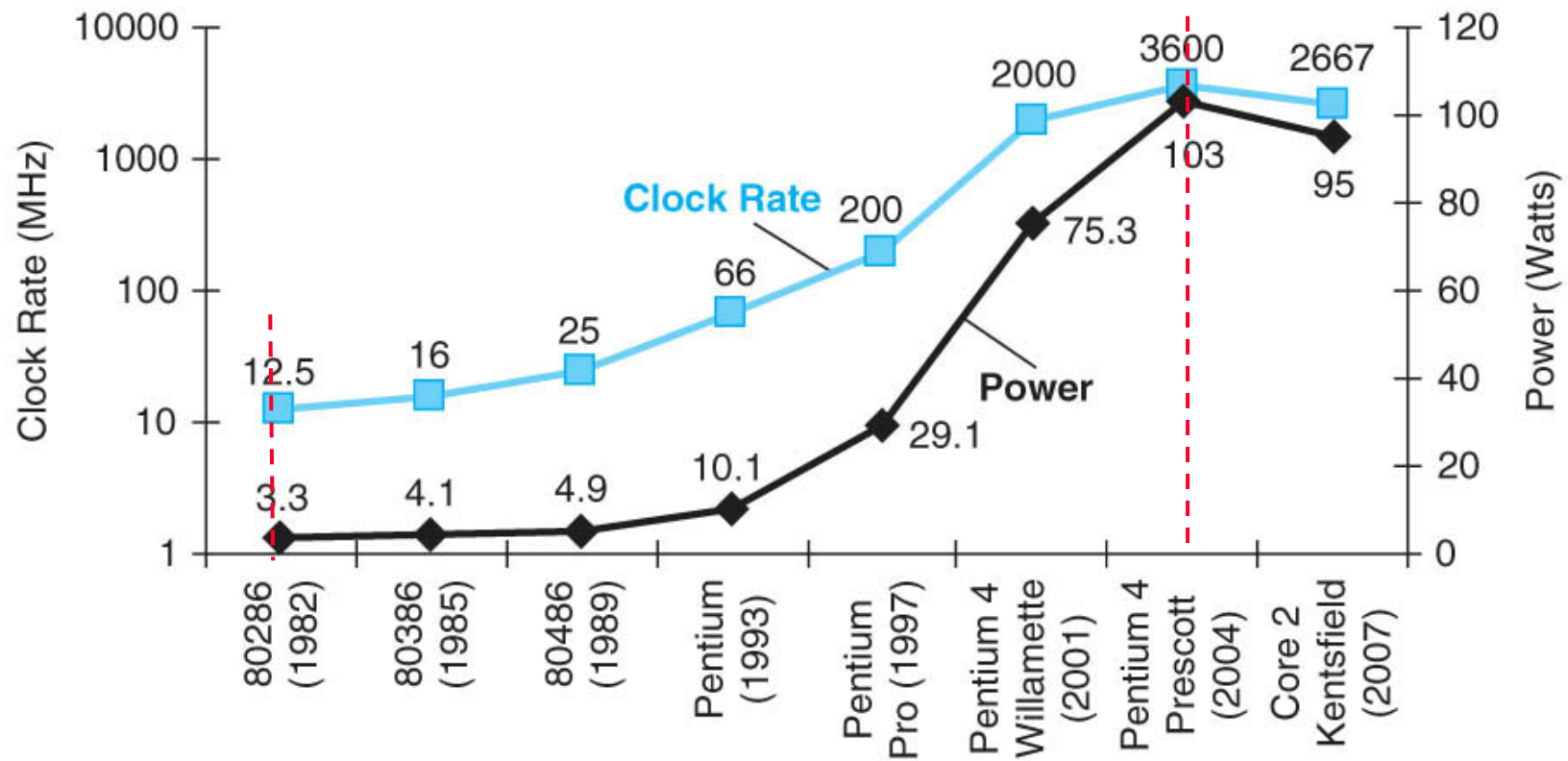


Revisão de tecnologia CMOS



$$P_{\text{din}} = C \times V_{DD}^2 \times f_{0 \rightarrow 1}$$

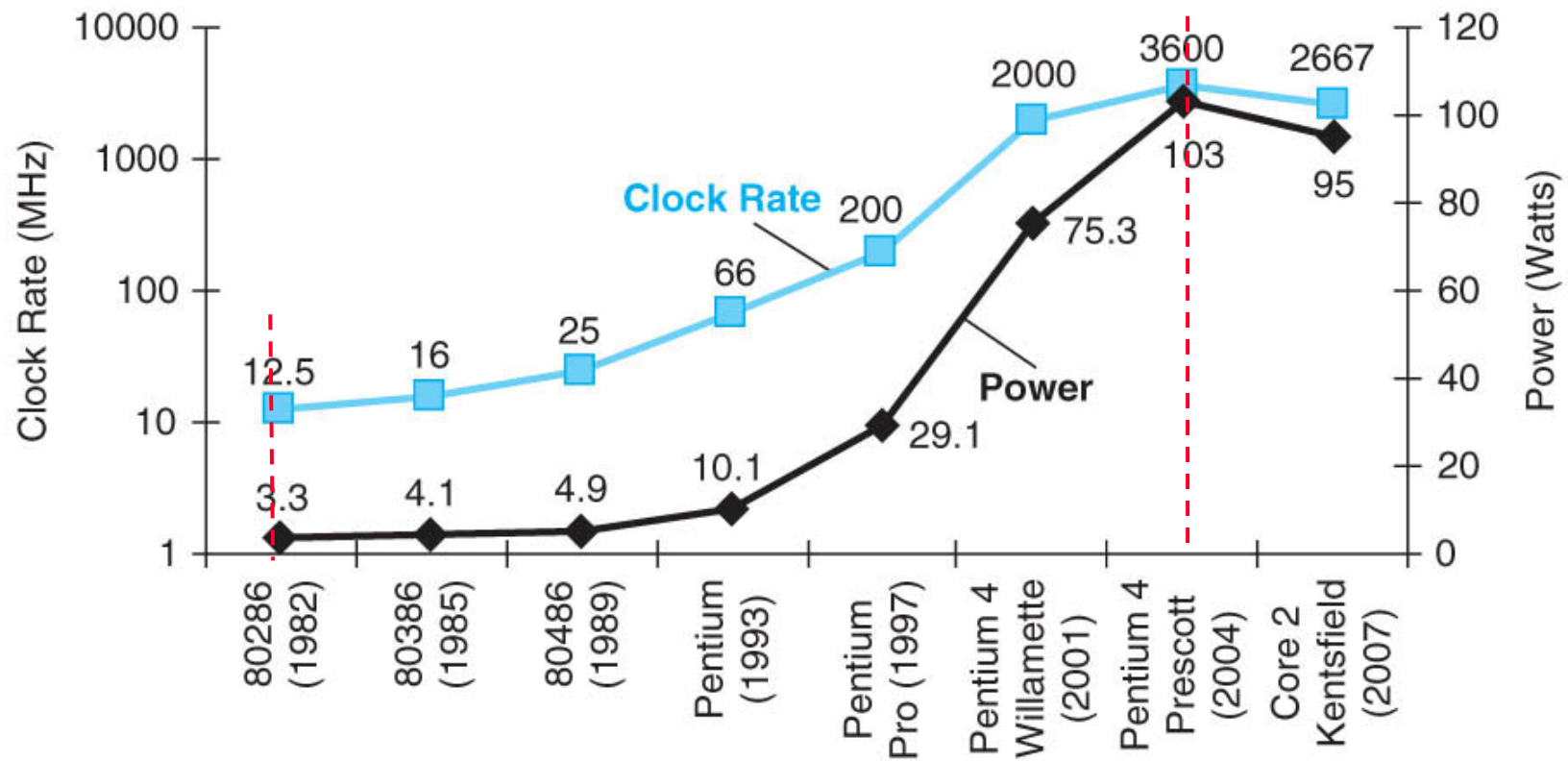
A barreira da potência



$$P_{\text{din}} = C \times V_{\text{DD}}^2 \times f_{0 \rightarrow 1}$$

Por que P cresceu só 30x enquanto f cresceu 300x ?

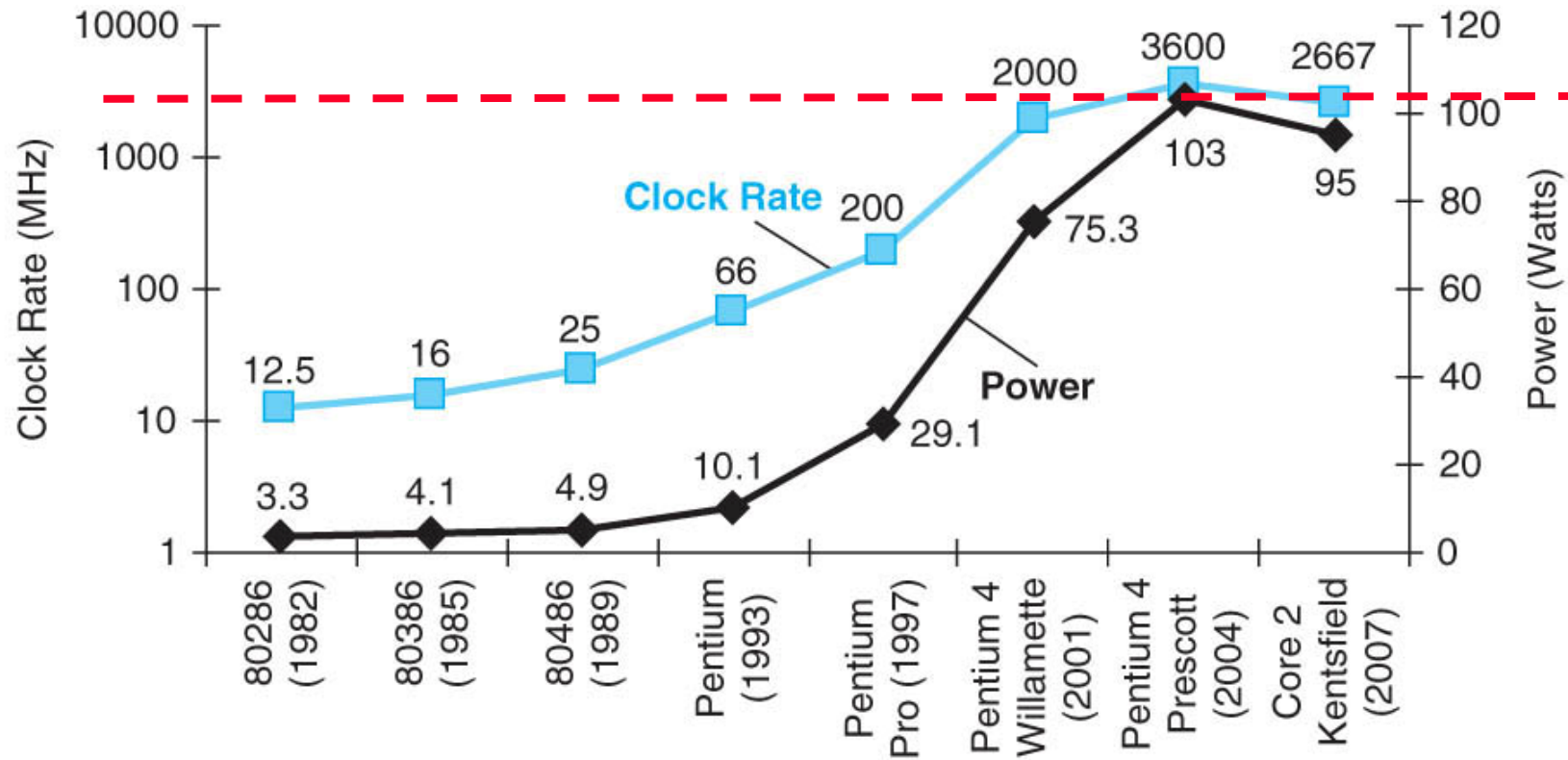
A barreira da potência



$$P_{\text{din}} = C \times V_{\text{DD}}^2 \times f_{0 \rightarrow 1}$$

Porque V_{DD} caiu de 5V para 1V no mesmo período!

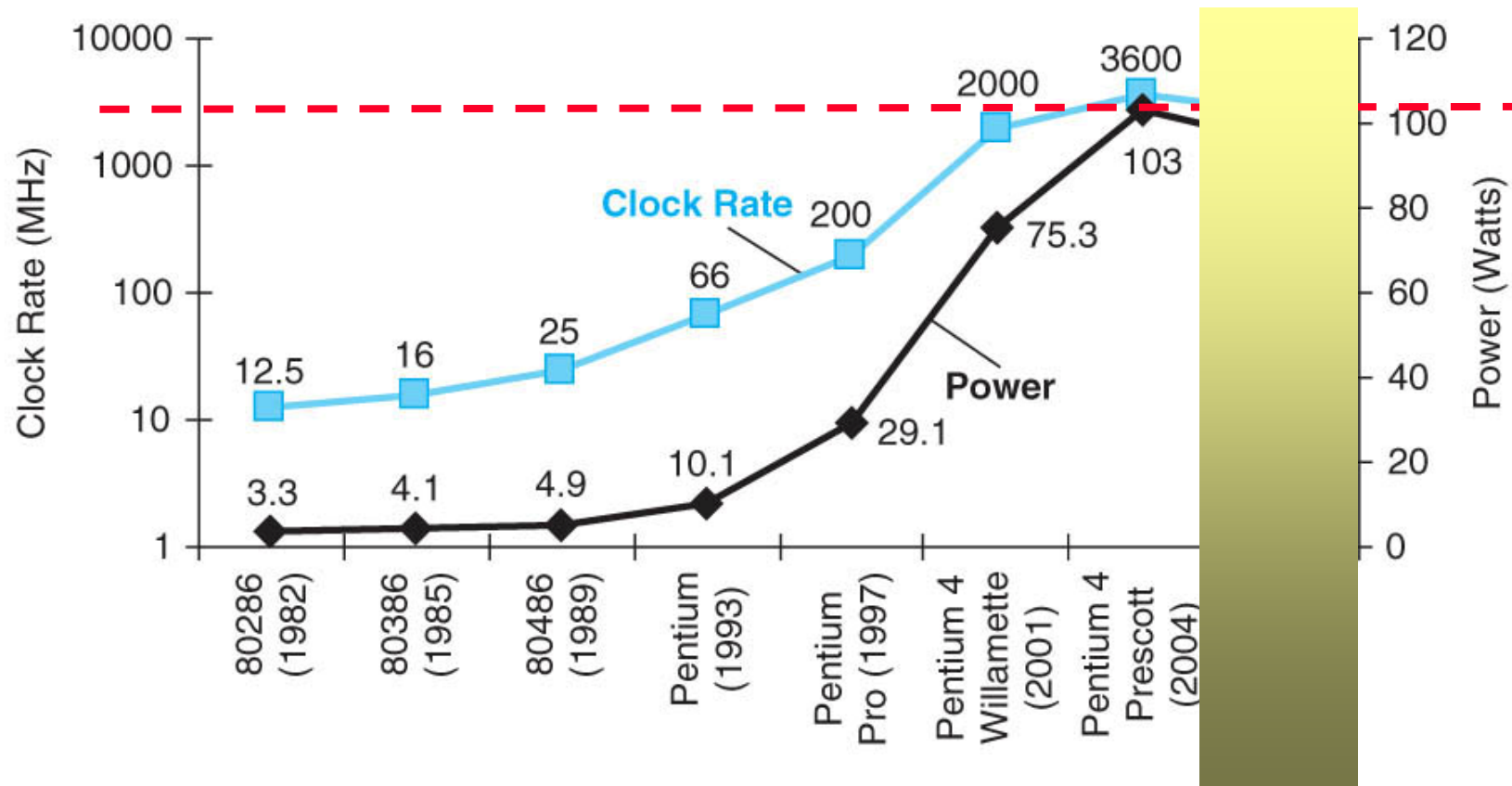
A barreira da potência



$$P_{\text{din}} = C \times V_{\text{DD}}^2 \times f_{0 \rightarrow 1}$$

Para desktops, há um limite prático de resfriamento que restringe a potência máxima a 100W

A barreira da potência



$$P_{\text{din}} = C \times V_{\text{DD}}^2 \times f_{0 \rightarrow 1}$$

Consequência: mudança de paradigma nos microprocessadores

Microprocessadores: mudança de paradigma

- Do uniprocessador ...
- Ao multiprocessador em um único chip
 - Microprocessador *multicore*

Microprocessadores: mudança de paradigma

- Do uniprocessador ...
- Ao multiprocessador em um único chip
 - Microprocessador *multicore*
- Panorama em 2008:

Product	AMD Opteron X4 (Barcelona)	Intel Nehalem	IBM Power 6	Sun Ultra SPARC T2 (Niagara 2)
Cores per chip	4	4	2	8
Clock rate	2.5 GHz	~ 2.5 GHz ?	4.7 GHz	1.4 GHz
Microprocessor power	120 W	~ 100 W ?	~ 100 W ?	94 W

Planejamento dos fabricantes:
dobrar o número de cores a cada 2 anos.

Consequência do novo paradigma

- **Requer programação paralela explícita**
 - **Ao contrário do paralelismo entre instruções**
 - » Hardware executa múltiplas instruções por vez
 - » Paralelismo escondido do programador
 - Compilador e HW faziam quase todo o trabalho
 - **Dificuldades do novo paradigma:**
 - » Programar para desempenho (mais difícil)
 - Convencional: deve-se garantir programa correto, resolução de problema importante, interface adequada
 - Agora tem que se programar para velocidade
 - » Balancear carga
 - » Otimizar comunicação e sincronização

Suporte ao paralelismo a ser estudado

- **Sincronização**
 - Entre *threads*
- **Paralelismo entre instruções**
 - De uma dada *thread*
- **Hierarquia de memória compartilhada**
 - Coerência e consistência

IEEE Computer Magazine, março 2010

