

# **William Stallings**

# **Computer Organization**

# **and Architecture**

# **8<sup>th</sup> Edition**

---

## **Chapter 3**

## **Avaliação de Desempenho**

# Clock do sistema

---



- Cristal de Quartzo envia onda de sinais constantes que são convertidas em sinais digitais
- 1GHz = 1 bilhão de pulsos
- Taxa de pulsos = taxa de clock
- Incremento de clock = ciclo de clocks
- Tempo entre pulsos = tempo de ciclos

# Taxa de Execução de Instrução

---

- Processador é controlado por um clock com frequência constante  $f$  ou, de modo equivalente, um tempo de ciclo constante  $\tau$ , onde  $\tau = 1/f$ .
- Contagem de instruções **Ic** é o número de instruções de máquinas, executadas, até que um determinado programa rode até o fim ou por algum intervalo de tempo específico. (Obs.: n° de execuções de instrução e não n° de instruções no código)

# **Taxa de Execução de Instrução**

---

- CPI → Média de ciclos por instrução
- Se todas as instruções exigissem o mesmo número de ciclos de clock, CPI seria valor constante para o processador.
- Na realidade, esse valor varia para cada tipo de instrução (load, store, etc.)

# Taxa de Execução de Instrução

---

- CPI<sub>i</sub> é o número de ciclos exigidos para a instrução tipo *i* e I<sub>i</sub> o número de instruções executadas do tipo *i* para determinado programa.

$$CPI = \frac{\sum_{i=1}^n (CPI_i \times I_i)}{I_c}$$

CPI - Média de Ciclos por instrução no conjunto

I<sub>c</sub> - Total de Instruções do conjunto

I<sub>i</sub> - Parcial de Instruções que levam a mesma quantidade de ciclos para executar

CPI<sub>i</sub> - Quantidade de ciclos que um grupo leva pra executar

# Taxa de Execução de Instrução

---

- O tempo de processador,  $T$ , necessário para executar determinado programa pode ser expresso como:

$$T = I_c \times CPI \times \tau^{(1/f)}$$

$T$  - Tempo de processador pra rodar o conjunto todo

$I_c$  - Total de Instruções

$CPI$  - Média de ciclos por instrução do conjunto

$\tau$  - Tempo do ciclo de clock do processador

$$f = 1 / \tau$$

$$\tau = 1 / f$$

# Fatores de Desempenho e Atributos do Sistema

- Como parte do trabalho é feito pelo processador e parte do trabalho é feito em transferências de dados para a memória, cujo tempo de ciclo pode ser maior que o do processador, a equação pode ser reescrita:

$$T = I_c \times [p + (m \times k)] \times \tau$$

- Onde,  $p$  é o número de ciclos de processador necessários para decodificar e executar a instrução,  $m$  é o número de referências de memória necessárias e  $k$  razão entre o tempo de ciclo de memória e o tempo de ciclo do processador.

	$I_c$	$p$	$m$	$k$	$\tau$
Arquitetura do conjunto de instruções	X	X			
Tecnologia do compilador	X	X	X		
Implementação do processador		X			X
Hierarquia da cache e da memória				X	X

\* Arquitetura do conjunto de instruções → Projeto do conjunto de instruções

# Milhões de Instruções por Segundo

---

$$MIPS = \frac{I_c}{T \times 10^6} = \frac{f}{CPI \times 10^6}$$

$$T = I_c \times CPI \times \tau$$

$$\frac{T}{I_c} = CPI \times \tau \quad (\text{Cruzando})$$

$$\frac{I_c}{T} = \frac{1}{CPI \times \tau} \quad f = 1 / \tau$$

$$\frac{I_c}{T} = \frac{f}{CPI}$$



# Bilhões de Instruções por Segundo

---

$$MIPS = \frac{I_c}{T \times 10^6} = \frac{f}{CPI \times 10^6}$$

Apresenta o resultado em Milhões

$$BIPS = \frac{I_c}{T \times 10^9} = \frac{f}{CPI \times 10^9}$$

Apresenta o resultado em Bilhões

# Exemplo

---

- Considere a execução de um programa que resulta na execução de 2 milhões de instruções num processador 400MHz. O programa consiste em quatro tipos principais de instruções, de acordo com a tabela abaixo:

Tipo de Instrução	CPI	Número de Instruções (%)
Aritmética e Lógica	1	60%
Load/Store com acerto de cache	2	18%
Desvio	4	12%
Referência de memória com falha de cache	8	10%

# Exemplo

$$I_c = 2000000 \text{ Instruções} - f = 400 \times 10^6 \text{ Hz}$$

$$CPI1 = 1 - I1 = 0,60 \times 2000000 = 1200000$$

$$CPI2 = 2 - I2 = 0,18 \times 2000000 = 360000$$

$$CPI3 = 4 - I3 = 0,12 \times 2000000 = 240000$$

$$CPI4 = 8 - I4 = 0,10 \times 2000000 = 200000$$

$$CPI = \frac{(1 \times 1200000) + (2 \times 360000) + (4 \times 240000) + (8 \times 200000)}{2000000}$$

$$CPI = 2,24 \text{ Ciclos por instrução}$$

$$T = I_c \times CPI \times \frac{1}{f}$$

$$T = 2000000 \times 2,24 \times \frac{1}{400 \times 10^6}$$

$$T = 0,0112 \text{ s}$$

$$T = 11,2 \text{ ms}$$

$$MIPS = \frac{f}{CPI \times 10^6}$$

$$MIPS = \frac{400 \times 10^6}{2,24 \times 10^6}$$

$$MIPS \cong 178 \text{ milhões de instruções por segundo}$$

$$BIPS = \frac{f}{CPI \times 10^9}$$

$$BIPS = \frac{400 \times 10^6}{2,24 \times 10^9}$$

$$BIPS \cong 0,18 \text{ bilhões de instruções por segundo}$$

# Exemplo

---

$I_c = 2000000$  Instruções –  $f = 400 \times 10^6$  Hz

$$CPI_1 = 1 - I_1 = 0,60$$

$$CPI_2 = 2 - I_2 = 0,18$$

$$CPI_3 = 4 - I_3 = 0,12$$

$$CPI_4 = 8 - I_4 = 0,10$$

$$CPI = (1 \times 0,60) + (2 \times 0,18) + (4 \times 0,12) + (8 \times 0,10)$$

$$CPI = 2,24 \text{ Ciclos por instrução}$$

$$T = I_c \times CPI \times \frac{1}{f}$$

$$T = 2000000 \times 2,24 \times \frac{1}{400 \times 10^6}$$

$$T = 0,0112 \text{ s}$$

$$T = 11,2 \text{ ms}$$

$$MIPS = \frac{f}{CPI \times 10^6}$$

$$MIPS = \frac{400 \times 10^6}{2,24 \times 10^6}$$

$$MIPS \cong 178 \text{ milhões} \\ \text{de instruções} \\ \text{por segundo}$$

$$BIPS = \frac{f}{CPI \times 10^9}$$

$$BIPS = \frac{400 \times 10^6}{2,24 \times 10^9}$$

$$BIPS \cong 0,18 \text{ bilhões} \\ \text{de instruções} \\ \text{por segundo}$$

# Lei de Amdahl

---

- Alguns exemplos de melhoria de desempenho de computadores passam pelo uso de processadores paralelos, hierarquia de cache e speedup no tempo de acesso da memória e na taxa de transferência de E/S
- A Lei de Amdahl, proposta por Gene Amdahl lida com o potencial de speedup de um programa usando múltiplos processadores em comparação com um único processador.

# Lei de Amdahl

---

- Considere um programa rodando em um único processador, de modo que uma fração  $(1-f)$  do tempo de execução envolva um Código inerentemente serial e uma fração  $f$  envolva Código infinitamente paralelizável sem overhead de escalonamento. Considere que  $T$  seja o tempo de execução total do programa usando um único processador. Então, o speedup usando um processador paralelo com  $N$  processadores que explora totalmente a parte paralela do programa é como segue:

$$\text{Speedup} = \frac{\text{Tempo para Executar Programa em um Único Processador}}{\text{Tempo para Executar Programa em } N \text{ Processadores Paralelos}}$$

$$\text{Speedup} = \frac{T(1-f) + Tf}{T(1-f) + \frac{Tf}{N}} = \frac{1}{(1-f) + \frac{f}{N}}$$

# Lei de Amdahl

---

$$Speedup = \frac{1}{(1 - f) + \frac{f}{N}}$$

N – Número de processadores

f - % das instruções que podem rodar em paralelo

(1 – f) - % das instruções seriais

# Lei de Amdahl

---

- Conclusões:
  - Se  $f$  for pequeno, o uso de processadores paralelos tem pouco efeito
  - Se  $N$  tende ao infinito, o speedup é limitado por  $1/(1-f)$ , de modo que existem retornos decrescentes para o uso de mais processadores.



# Lei de Amdahl

---

- A Lei de Amdahl ilustra os problemas enfrentados pela indústria no desenvolvimento de máquinas Multicore com um número cada vez maior de processadores: o software que roda nessas máquinas, deve ser adaptado para um ambiente de execução altamente paralelo, para explorar o poder do processamento paralelo.

# Lei de Amdahl

---

- Considere qualquer melhoria a um recurso de um Sistema que resulte num speedup o qual pode ser expresso por:

$$\text{Speedup} = \frac{\text{Desempenho após melhoria}}{\text{Desempenho antes da melhoria}} = \frac{\text{Tempo de execução antes da melhoria}}{\text{Tempo de execução após melhoria}}$$

- Suponha que o recurso do Sistema seja usado durante a execução de uma fração do tempo  $f$ , antes da melhoria, e que o speedup desse recurso, após a melhoria seja  $SU_f$ . Então, o speedup geral do sistema é:

$$\text{Speedup} = \frac{1}{(1 - f) + \frac{f}{SU_f}}$$

# Lei de Amdahl

---

$$Speedup = \frac{1}{(1 - f) + \frac{f}{SU_f}}$$

$f$  - % das instruções que podem rodar em paralelo

$(1 - f)$  - % das instruções seriais

$SU_f$  – Fator de melhoria computacional

Máquina Base  $\rightarrow SU_f = 1$

## Exemplo

---

- Suponha que uma tarefa utilize muitas operações de ponto flutuante com 40% do tempo sendo consumido por operações de ponto flutuante. Com um novo projeto de hardware, o modulo de ponto flutuante é agilizado por um fator  $K$ . Então, o speedup geral é:

$$Speedup = \frac{1}{(1 - 0,4) + \frac{0,4}{K}}$$

# Exercício 1

---

- Considere a execução de um programa que resulta na execução de 10 milhões de instruções num processador 1,6GHz. O programa consiste em quatro tipos principais de instruções, de acordo com a tabela abaixo. Qual o CPI, T e MIPS (ou BIPS) ?

Tipo de Instrução	CPI	Número de Instruções (%)
Aritmética e Lógica	2	66%
Load/Store	4	14%
Desvio de processamento	6	12%
Referência de cache	16	8%

# Exercício 1

---

$$I_c = 10000000 \text{ Instruções} - f = 1,6 \times 10^9 \text{ Hz}$$

$$CPI_1 = 2 - I_1 = 0,66 \times 10000000 = 6600000$$

$$CPI_2 = 4 - I_2 = 0,14 \times 10000000 = 1400000$$

$$CPI_3 = 6 - I_3 = 0,12 \times 10000000 = 1200000$$

$$CPI_4 = 16 - I_4 = 0,08 \times 10000000 = 800000$$

$$CPI = \frac{(2 \times 6600000) + (4 \times 1400000) + (6 \times 1200000) + (16 \times 800000)}{10000000}$$

$$CPI = 3,88 \text{ Ciclos por instrução}$$

$$T = I_c \times CPI \times \frac{1}{f}$$

$$T = 10000000 \times 3,88 \times \frac{1}{1,6 \times 10^9}$$

$$T = 0,02425 \text{ s}$$

$$T = 24,25 \text{ ms}$$

$$MIPS = \frac{f}{CPI \times 10^6}$$

$$MIPS = \frac{1,6 \times 10^9}{3,88 \times 10^6}$$

$$MIPS = 412,4 \text{ milhões de instruções por segundo}$$

$$BIPS = \frac{f}{CPI \times 10^9}$$

$$BIPS = \frac{1,6 \times 10^9}{3,88 \times 10^9}$$

$$BIPS \cong 0,4 \text{ bilhões de instruções por segundo}$$

# Exercício 1

---

$$I_c = 10000000 \text{ Instruções} - f = 1,6 \times 10^9 \text{ Hz}$$

$$CPI1 = 2 - I1 = 0,66$$

$$CPI2 = 4 - I2 = 0,14$$

$$CPI3 = 6 - I3 = 0,12$$

$$CPI4 = 16 - I4 = 0,08$$

$$CPI = (2 \times 0,66) + (4 \times 0,14) + (6 \times 0,12) + (16 \times 0,08)$$

$$CPI = 3,88 \text{ Ciclos por instrução}$$

$$T = I_c \times CPI \times \frac{1}{f}$$

$$T = 10000000 \times 3,88 \times \frac{1}{1,6 \times 10^9}$$

$$T = 0,02425 \text{ s}$$

$$T = 24,25 \text{ ms}$$

$$MIPS = \frac{f}{CPI \times 10^6}$$

$$MIPS = \frac{1,6 \times 10^9}{3,88 \times 10^6}$$

$$MIPS = 412,4 \text{ milhões de instruções por segundo}$$

$$BIPS = \frac{f}{CPI \times 10^9}$$

$$BIPS = \frac{1,6 \times 10^9}{3,88 \times 10^9}$$

$$BIPS \cong 0,4 \text{ bilhões de instruções por segundo}$$

## Exercício 2

---

- Considerando que o tempo de processador para executar o programa do exercício 1 seja  $10\mu\text{s}$ , qual a contagem de Instruções  $I_c$ ?

$$T = I_c \times CPI \times \frac{1}{f}$$

$$I_c = \frac{T \times f}{CPI}$$

$$I_c = \frac{10 \times 10^{-6} \times 1,6 \times 10^9}{3,88}$$

$$I_c \cong 4124 \text{ Instruções}$$



## Exercício 3

---

- Suponha que uma tarefa utilize muitas operações de ponto flutuante com 35% do tempo sendo consumido por operações de ponto flutuante. Com um novo projeto de hardware, o modulo de ponto flutuante é agilizado por um fator 100. Então, o speedup geral é? Qual o speedup máximo?

$$Speedup_{Geral} = \frac{1}{(1 - f) + \frac{f}{SU_f}}$$

$$Speedup_{Geral} = \frac{1}{(1 - 0,35) + \frac{0,35}{100}}$$

$$Speedup_{Geral} = 1,53$$

$$Speedup_{Max} = \frac{1}{(1 - f) + \frac{f}{\infty}}$$

$$Speedup_{Max} = \frac{1}{(1 - f)}$$

$$Speedup_{Max} = 1,54$$

## Exercício 4

- Suponha que uma tarefa utilize muitas operações de ponto flutuante com 44% do tempo sendo consumido por operações de ponto flutuante. Qual o fator de agilidade necessário para um aumento de speedup de 17%?

$$Speedup = \frac{1}{(1-f) + \frac{f}{SU_f}}$$

$$SU_f = \frac{f}{\frac{1}{Speedup} - (1-f)}$$

$$(1-f) + \frac{f}{SU_f} = \frac{1}{Speedup}$$

$$SU_f = \frac{0,44}{\frac{1}{1,17} - 0,56}$$

$$\frac{f}{SU_f} = \frac{1}{Speedup} - (1-f)$$

$$SU_f \sim 1,5$$

$$\frac{1}{SU_f} = \frac{\frac{1}{Speedup} - (1-f)}{f}$$