اصول سیستم‌های کامپیوتری

فصل ۱: فناوری و تجریدهای کامپیوتری

علی شکیبا – ali@shakiba.id.ir

زمان پاسخ و توان عملیاتی

- زمان پاسخ (Response Time)
  - زمان مورد نیاز برای انجام **یک کار**

زمان پاسخ و توان عملیاتی

- زمان پاسخ (Response Time)
  - زمان مورد نیاز برای انجام **یک کار**
- توان عملیاتی (Throughput)
  - جمع کل کار انجام شده در واحد زمان
    - مانند تعداد کار، وظیفه، تراکنش و ... در هر ساعت

عملکرد نسبی

- کارایی = ۱ / زمان‌اجرا
- «X، n بار از Y سریع‌تر است»

$$\text{Performance}_X / \text{Performance}_Y = \text{Execution time}_Y / \text{Execution time}_X = n$$

- مثال:
  - 10s برای A، 15s برای B

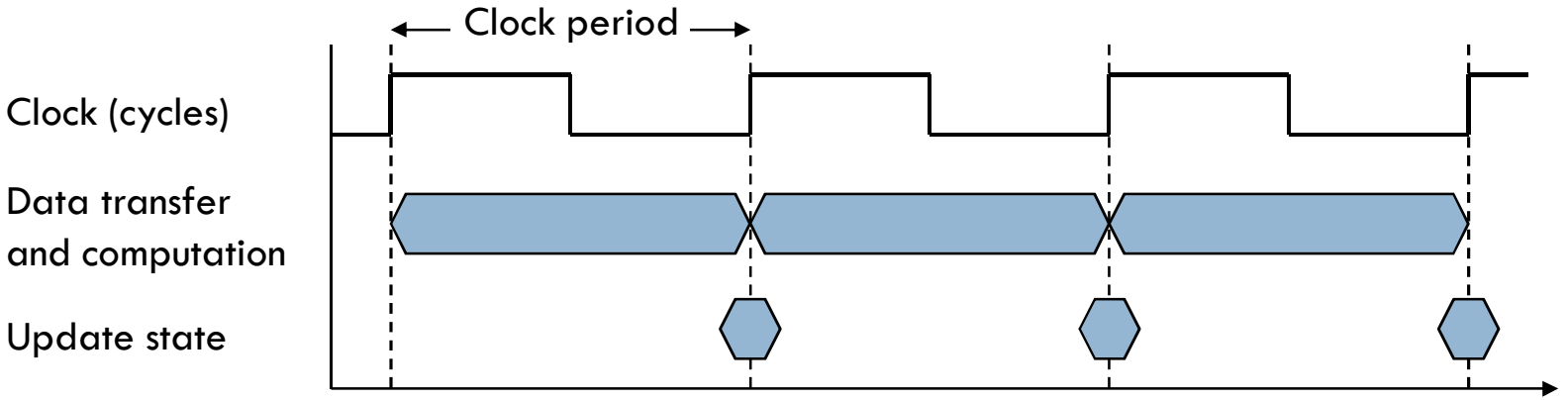
- Execution Time_B / Execution Time_A
  = 15s / 10s = 1.5
  - بنابراین، A، ۱.۵ برابر سریعتر از B است!

اندازه‌گیری زمان اجرا

- فاصله بین زمان شروع و زمان پایان
  - شامل
    - زمان پردازش، I/O، سربار سیستم‌عامل، زمان تلف شده و ...
  - تعیین کننده کارایی سیستم
- زمان CPU
  - زمانی که فقط صرف امور پردازشی برنامه شده
    - بدون درنظر گرفتن زمان I/O، زمان مصرف شده برای اجرای سایر برنامه‌ها، ...
  - شامل زمان CPU کاربر و OS
  - تاثیرپذیری برنامه‌های مختلف نسبت به OS و کارایی سیستم متفاوت است.

چرخه‌های ساعت (CPU Clocking)

- عملکرد سخت‌افزار دیجیتال تابع ساعتی با دوره تناوب ثابت است.



- دوره تناوب ساعت (Clock Period)
  - طول یک چرخه ساعت، برای نمونه، $250ps = 0.25ns = 250 \times 10^{-12}s$
- فرکانس ساعت (Clock Frequency (Rate))
  - تعداد چرخه در هر ثانیه، برای نمونه، $4.0GHz = 4000MHz = 4.0 \times 10^{9}Hz$

زمان CPU

$$\text{CPU Time} = \text{CPU Clock Cycles} \times \text{Clock Cycle Time}$$
$$= \frac{\text{CPU Clock Cycles}}{\text{Clock Rate}}$$

- بهبود کارایی با
  - کاهش تعداد چرخه‌های لازم
  - افزایش نرخ ساعت
- اما، در اغلب اوقات طراحان سخت‌افزار لازم است تا بین **نرخ ساعت** و **تعداد چرخه** مصالحه‌ای را اتخاذ کنند!

مثال

- کامپیوتر A: 2GHz clock, 10s CPU time داده شده است
- قصد داریم تا کامپیوتر B را به گونه‌ای طراحی کنیم تا
  - زمان CPU به 6s برسد
  - افزایش نرخ ساعت منجر به 1.2 × clock cycles می‌شود
- نرخ ساعت کامپیوتر B چقدر باید باشد؟

$$\text{CPU Time} = \text{CPU Clock Cycles} \times \text{Clock Cycle Time}$$

$$= \frac{\text{CPU Clock Cycles}}{\text{Clock Rate}}$$

مثال

- کامپیوتر A: 2GHz clock, 10s CPU time داده شده است
- قصد داریم تا کامپیوتر B را به گونه‌ای طراحی کنیم تا
  - زمان CPU به 6s برسد
  - افزایش نرخ ساعت منجر به 1.2 × clock cycles می‌شود
- نرخ ساعت کامپیوتر B چقدر باید باشد؟

$$\text{Clock Rate}_B = \frac{\text{Clock Cycles}_B}{\text{CPU Time}_B} = \frac{1.2 \times \text{Clock Cycles}_A}{6\text{s}}$$

$$\text{Clock Cycles}_A = \text{CPU Time}_A \times \text{Clock Rate}_A$$

$$= 10\text{s} \times 2\text{GHz} = 20 \times 10^9$$

$$\text{Clock Rate}_B = \frac{1.2 \times 20 \times 10^9}{6\text{s}} = \frac{24 \times 10^9}{6\text{s}} = 4\text{GHz}$$

تعداد دستورالعمل و CPI

$$\text{Clock Cycles} = \text{Instruction Count} \times \text{Cycles per Instruction}$$

$$\text{CPU Time} = \text{Instruction Count} \times \text{CPI} \times \text{Clock Cycle Time}$$

$$= \frac{\text{Instruction Count} \times \text{CPI}}{\text{Clock Rate}}$$

- تعداد دستورالعمل‌ها
  - تعیین توسط: برنامه، ISA و مترجم
- CPI یا متوسط چرخه‌ها برای هر دستور
  - تعیین شده توسط CPU
  - در صورتی که دستورات مختلف، CPI متفاوت داشته باشند،
    - ترکیب دستورات از رده‌های مختلف در CPI موثر است.

مثال از CPI

- Computer A: Cycle Time = 250ps, CPI = 2.0
- Computer B: Cycle Time = 500ps, CPI = 1.2
  - ISA یکسان است!
  - کدام‌یک سریعتر هستند؟ به چه نسبتی؟

$$\text{Clock Cycles} = \text{Instruction Count} \times \text{Cycles per Instruction}$$

$$\text{CPU Time} = \text{Instruction Count} \times \text{CPI} \times \text{Clock Cycle Time}$$

$$= \frac{\text{Instruction Count} \times \text{CPI}}{\text{Clock Rate}}$$

مثال از CPI

- Computer A: Cycle Time = 250ps, CPI = 2.0
- Computer B: Cycle Time = 500ps, CPI = 1.2
  - ISA یکسان است!
  - کدام‌یک سریعتر هستند؟ به چه نسبتی؟

$$\text{CPU Time}_A = \text{Instruction Count} \times \text{CPI}_A \times \text{Cycle Time}_A$$
$$= I \times 2.0 \times 250\text{ps} = I \times 500\text{ps}$$

← A سریعتر است

$$\text{CPU Time}_B = \text{Instruction Count} \times \text{CPI}_B \times \text{Cycle Time}_B$$
$$= I \times 1.2 \times 500\text{ps} = I \times 600\text{ps}$$

$$\frac{\text{CPU Time}_B}{\text{CPU Time}_A} = \frac{I \times 600\text{ps}}{I \times 500\text{ps}} = 1.2$$

← نسبت سرعت B به A

CPI از نگاه نزدیک‌تر!

- اگر رده‌های مختلف دستورات، تعداد چرخه‌های متفاوتی برای انجام نیاز داشته باشند

$$\text{Clock Cycles} = \sum_{i=1}^{n} (\text{CPI}_i \times \text{Instruction Count}_i)$$

- میانگین وزن‌دار CPI

$$\text{CPI} = \frac{\text{Clock Cycles}}{\text{Instruction Count}} = \sum_{i=1}^{n} \left(\text{CPI}_i \times \frac{\text{Instruction Count}_i}{\text{Instruction Count}} \right)$$

نرخ نسبی

مثال

- مترجم برای ترجمه یک برنامه، دو رویه مختلف را می‌تواند در پیش گیرد
- در هر یک از این رویه‌ها، تعداد دستورات از هر رده مختلف است

Class	A	B	C
CPI for class	1	2	3
IC in sequence 1	2	1	2
IC in sequence 2	4	1	1

مثال

- مترجم برای ترجمه یک برنامه، دو رویه مختلف را می‌تواند در پیش گیرد
- در هر یک از این رویه‌ها، تعداد دستورات از هر رده مختلف است

Class	A	B	C
CPI for class	1	2	3
IC in sequence 1	2	1	2
IC in sequence 2	4	1	1

- Sequence 1: IC = 5
  - Clock Cycles
    = 2×1 + 1×2 + 2×3
    = 10
  - Avg. CPI = 10/5 = 2.0

مثال

- مترجم برای ترجمه یک برنامه، دو رویه مختلف را می‌تواند در پیش گیرد
- در هر یک از این رویه‌ها، تعداد دستورات از هر رده مختلف است

Class	A	B	C
CPI for class	1	2	3
IC in sequence 1	2	1	2
IC in sequence 2	4	1	1

- Sequence 1: IC = 5
  - Clock Cycles
    = 2×1 + 1×2 + 2×3
    = 10
  - Avg. CPI = 10/5 = 2.0
- Sequence 2: IC = 6
  - Clock Cycles
    = 4×1 + 1×2 + 1×3
    = 9
  - Avg. CPI = 9/6 = 1.5

کارایی: جمع‌بندی

$$\text{CPU Time} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Clock cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Clock cycle}}$$

- کارایی وابسته است به
  - الگوریتم
  - زبان برنامه‌نویسی
  - مترجم
  - ISA

کارایی: جمع‌بندی

$$\text{CPU Time} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Clock cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Clock cycle}}$$

- کارایی وابسته است به
  - الگوریتم
    - موثر بر IC، تغییر احتمالی در CPI
  - زبان برنامه‌نویسی
  - مترجم
  - ISA

کارایی: جمع‌بندی

$$\text{CPU Time} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Clock cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Clock cycle}}$$

- کارایی وابسته است به
  - الگوریتم
    - موثر بر IC، تغییر احتمالی در CPI
  - زبان برنامه‌نویسی
    - موثر بر IC و CPI
  - مترجم
  - ISA

کارایی: جمع‌بندی

$$\text{CPU Time} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Clock cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Clock cycle}}$$

- کارایی وابسته است به
  - الگوریتم
    - موثر بر IC، تغییر احتمالی در CPI
  - زبان برنامه‌نویسی
    - موثر بر IC و CPI
  - مترجم
    - موثر بر IC و CPI
  - ISA

کارایی: جمع‌بندی

$$\text{CPU Time} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Clock cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Clock cycle}}$$

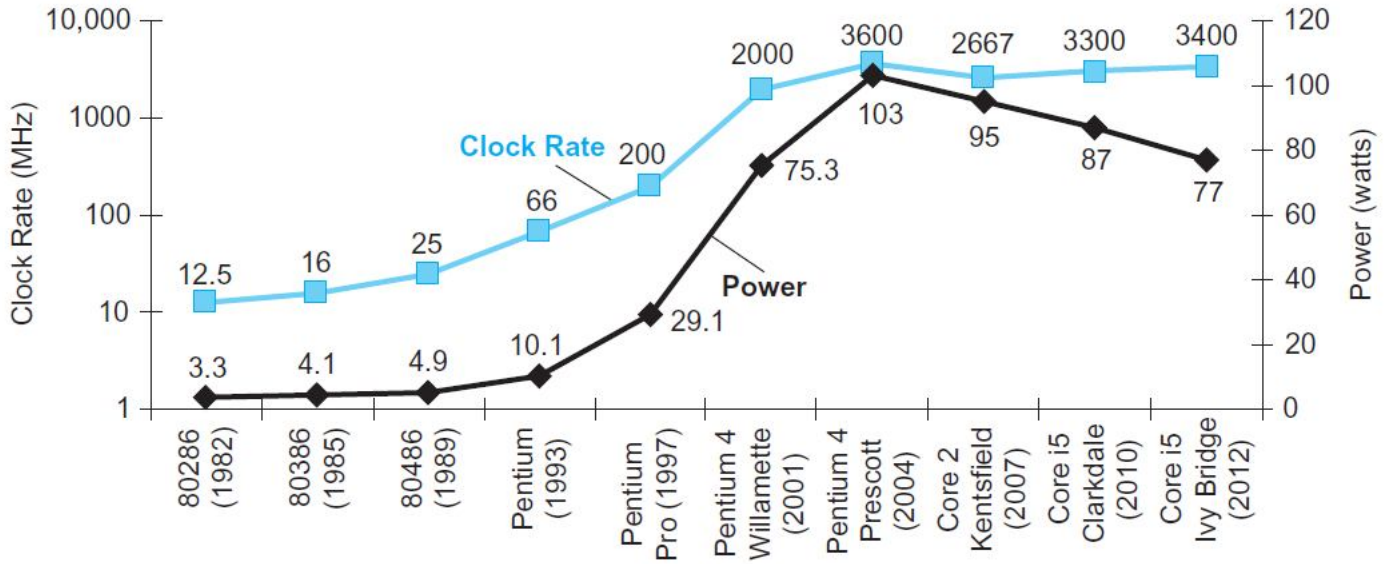
- کارایی وابسته است به
  - الگوریتم
    - موثر بر IC، تغییر احتمالی در CPI
  - زبان برنامه‌نویسی
    - موثر بر IC و CPI
  - مترجم
    - موثر بر IC و CPI
  - ISA
    - موثر بر IC و CPI و زمان چرخه ساعت (CP)

کارایی: جمع‌بندی

- عوامل موثر بر کارایی

Components of performance	Units of measure
CPU execution time for a program	Seconds for the program
Instruction count	Instructions executed for the program
Clock cycles per instruction (CPI)	Average number of clock cycles per instruction
Clock cycle time	Seconds per clock cycle

توان



- در فناوری طراحی CMOS IC

$$\text{Power} = \text{Capacitive load} \times \text{Voltage}^2 \times \text{Frequency}$$

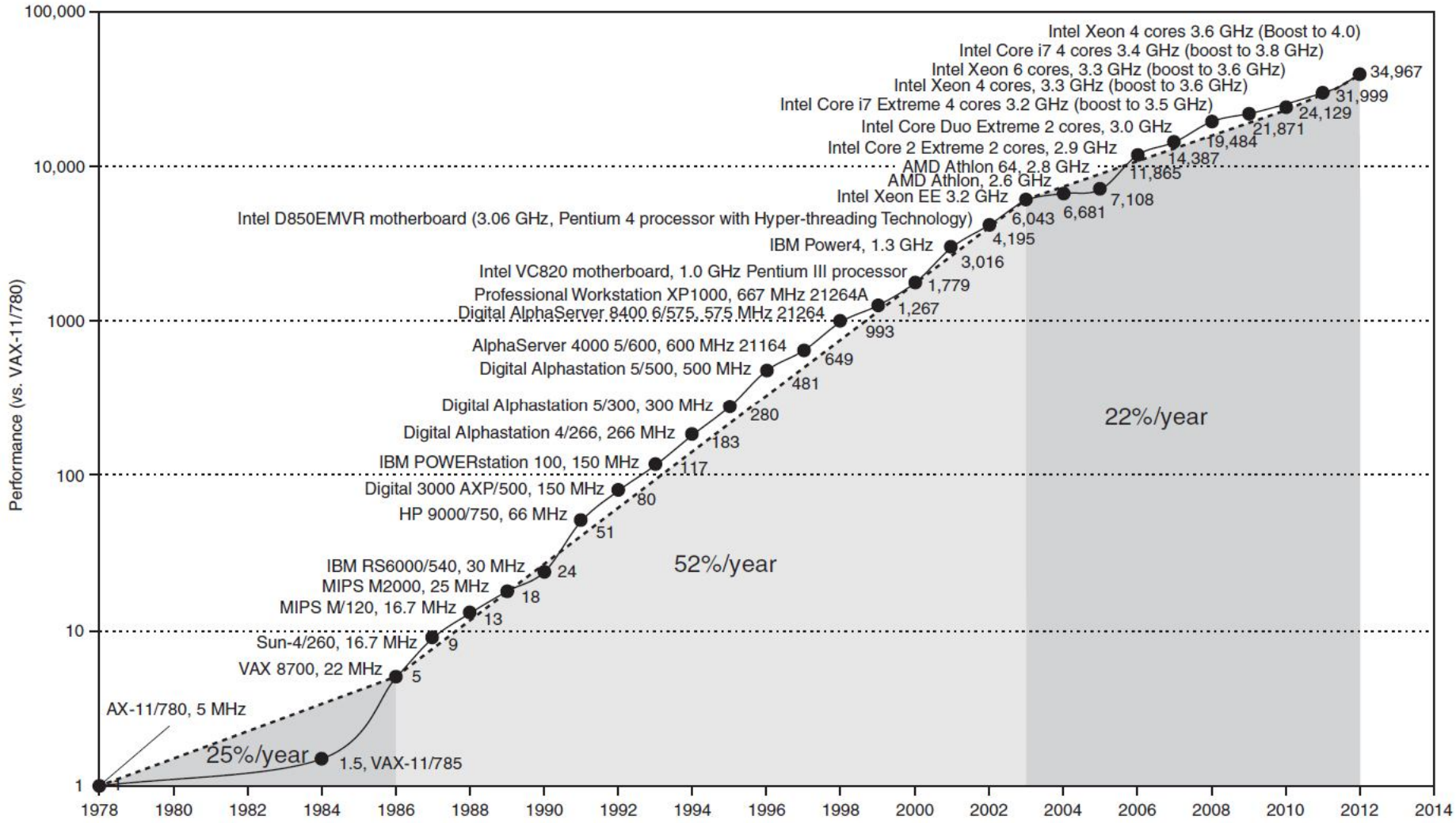
کاهش توان

- فرض کنید در پردازنده جدید
  - ظرفیت خازنی 85% پردازنده قدیمی است
  - و 15% در ولتاژ و 15% در فرکانس کاهش داشته‌ایم!

$$\frac{P_{new}}{P_{old}} = \frac{C_{old} \times 0.85 \times (V_{old} \times 0.85)^2 \times F_{old} \times 0.85}{C_{old} \times V_{old}^{\ 2} \times F_{old}} = 0.85^4 = 0.52$$

- دیوار توان!
  - نمی‌توانیم ولتاژ را از یک حدی بیشتر کم کنیم!
  - نمی‌توانیم بیش از این گرما را تحمل کنیم/کاهش دهیم!
- پس چگونه کارایی را افزایش دهیم؟

نرخ افزایش کارایی تک‌پردازنده



چندپردازنده

- میکروپروسسورهای چندهسته‌ای
  - بیش از یک پردازنده در هر چیپ
- الزام به برنامه‌نویسی موازی صریح!
  - مقایسه با توازی در سطح دستورات
    - توازی در سطح دستورات از برنامه نویس پنهان بوده است!
  - انجام ...
    - برنامه‌نویسی برای کارایی
    - توازن بار
    - بهینه‌سازی ارتباطات و بهنگام‌سازی
  - ... دشوار است