ارزیابی کارایی

معماری کامپیوتر

۱۳۹۱-۱۱-۱۳

جلسه‌ی دوم

دانشگاه شهید بهشتی

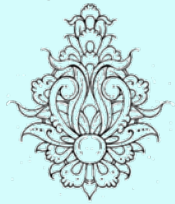
دانشکده‌ی مهندسی برق و کامپیوتر

زمستان ۱۳۹۱

احمد محمودی ازناوه

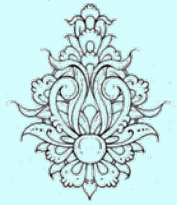
فهرست مطالب

- بخش‌های مختلف یک کامپیوتر
- چگونگی اجرای یک برنامه
- کارایی



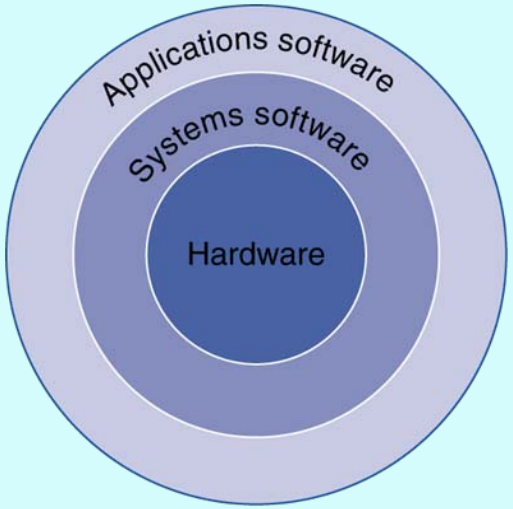
کارایی برنامه

- الگوریتم
  - تعداد دستورالعمل‌ها و تعداد عملیات I/O
- زبان برنامه‌نویسی، کامپایلر و معماری
  - تعداد دستورالعمل زبان ماشین به ازای دستورالعمل‌های زبان سطح بالا
- پردازنده و حافظه
  - سرعت اجرای هر دستور چقدر است؟
- سرعت انجام عملیات I/O



برنامه

- **برنامه‌های کاربردی**
  - به زبان‌های سطح بالا نوشته می‌شوند.
- **برنامه‌های سیستمی**
  - کامپایلر
  - سیستم‌عامل
    - مدیریت حافظه و ذخیره‌سازی
    - اشتراک منابع
    - مدیریت ورودی و خروجی



- **سخت‌افزار**
  - پردازنده، حافظه و ورودی-خروجی

سطوح زبان‌های برنامه‌نویسی

```
swap (int v[], int k){
   int temp;
   temp = v[k];
   v[k] = v[k+1];
   v[k+1] = temp;
}
```

one-to-many

C compiler

```
swap:       sll    $2, $5, 2
            add    $2, $4, $2
            lw     $15, 0($2)
            lw     $16, 4($2)
            sw     $16, 0($2)
            sw     $15, 4($2)
            jr     $31
```

one-to-one

assembler

```
0000000000000101
0001000010000000
0000000010000010
0001000000100000
       . . .
```

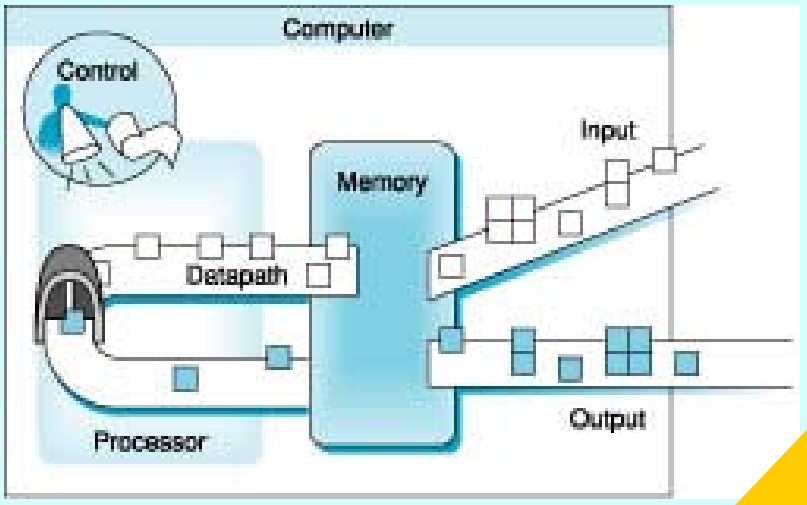
- **زبان‌های سطح بالا**
  - سطحی از تجرید که به زبان طبیعی نزدیک‌تر است
  - کارایی و قابلیت حمل برنامه را افزایش می‌دهد.
- **زبان اسمبلی**
  - نمادهایی که جایگزین زبان ماشین می‌شوند.
- **زبان ماشین**
  - دنباله‌ای از صفر و یک

instruction

دستوری که سخت‌افزار می‌فهمد و می‌تواند آن را اجرا کند

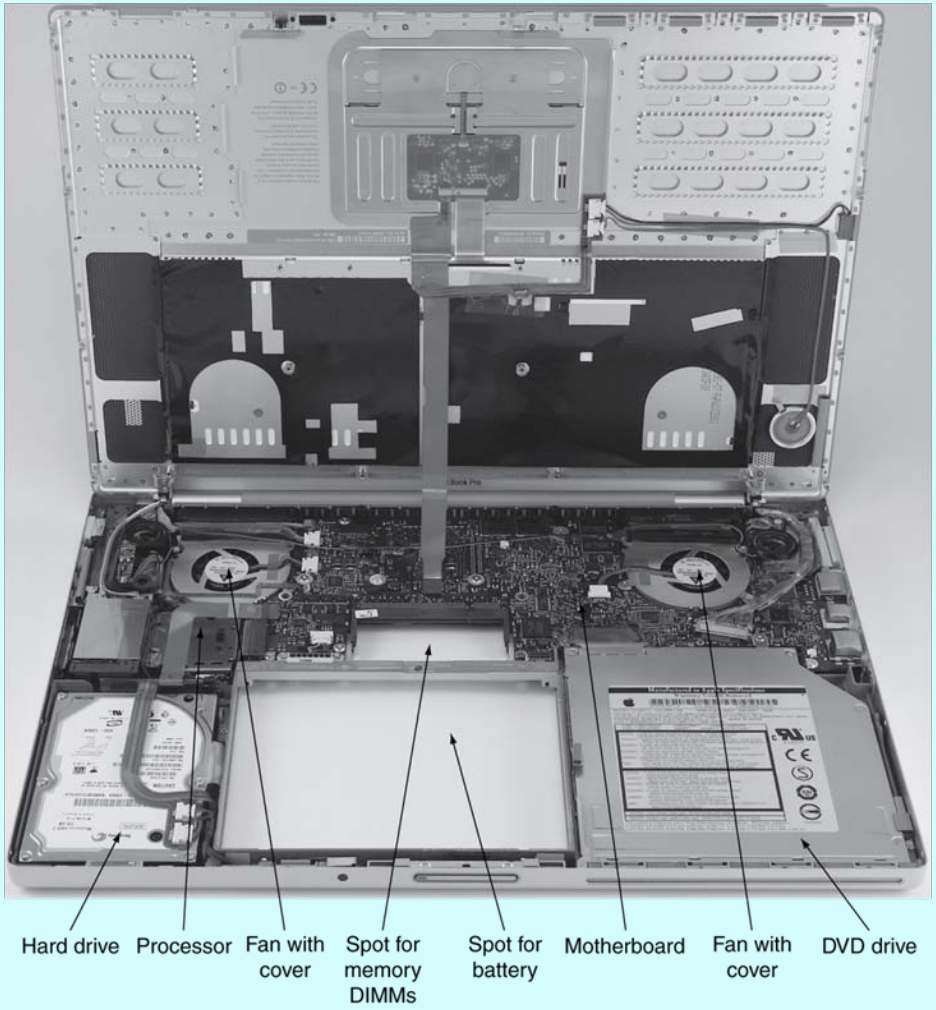
بخش‌های یک کامپیوتر

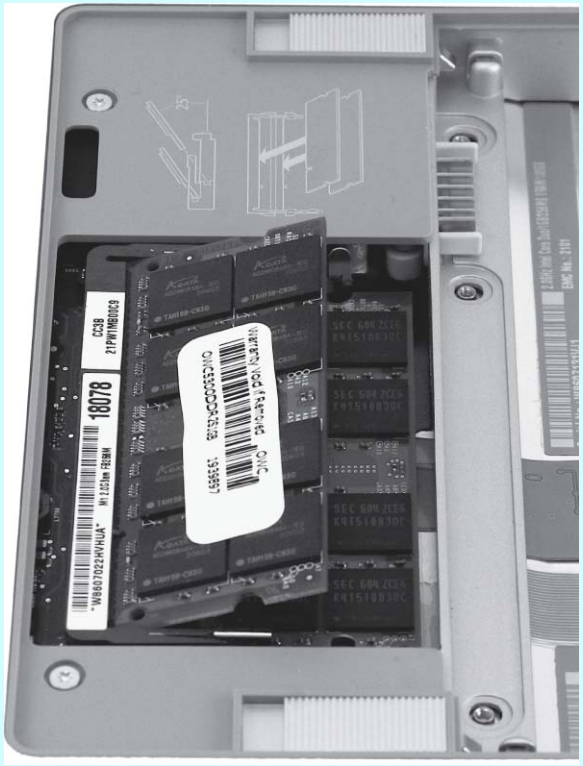
- گذشته از نوع معماری، هر کامپوتر دارای پنج بخش پایه است:
- ورودی
- خروجی
- حافظه
- مسیر گذار داده
- کنترل



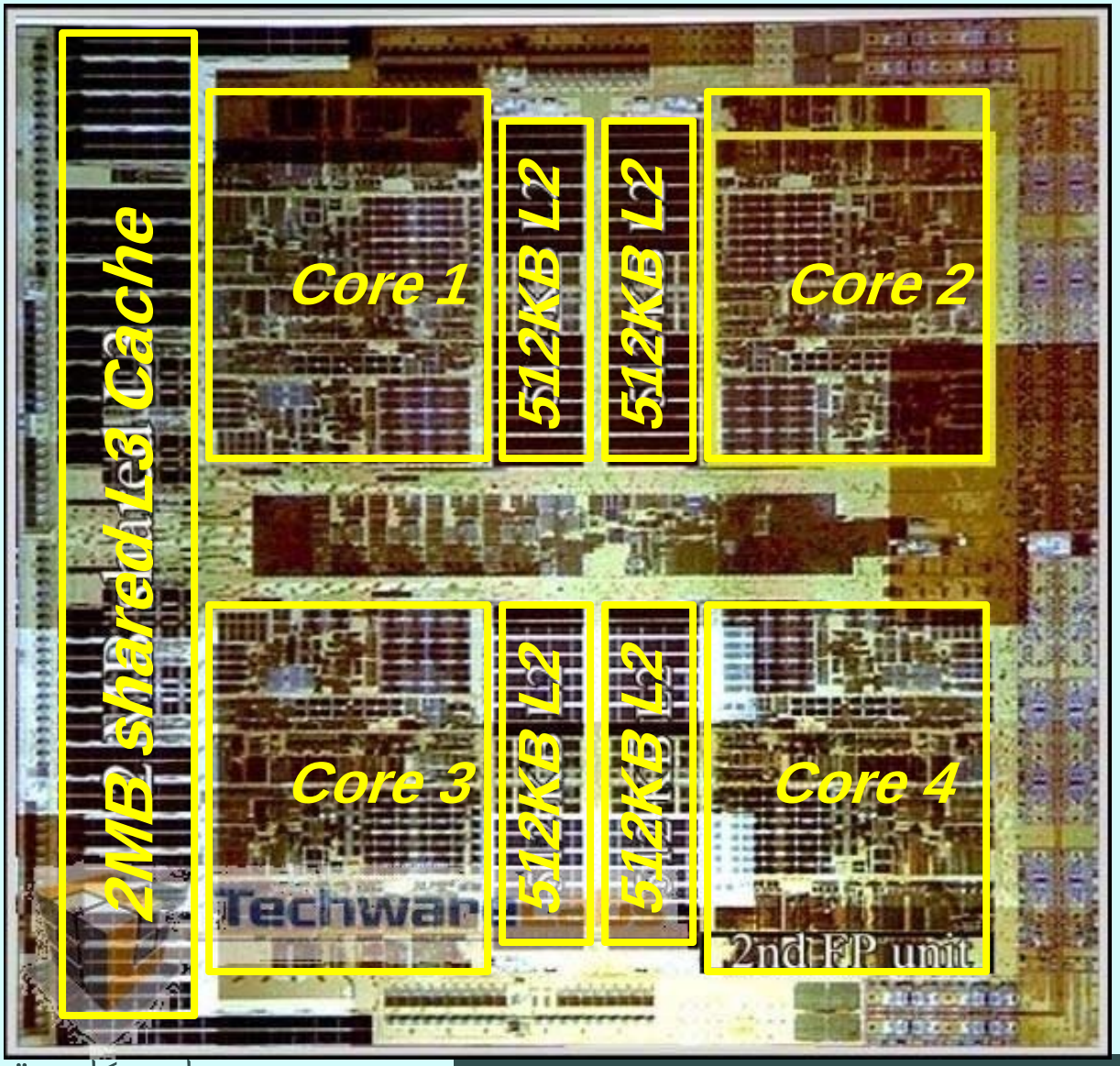
The BIG Picture

اجزای کامپیوتر





اجزای داخلی پردازنده



- مسیر گذار داده (انجام عملیات محاسباتی)
- واحد کنترل
- حافظه نهان (cache) (memory

AMD Barcelona: 4 processor cores

تجرید

- تجرید به برخورد با سیستم‌های پیچیده کمک می‌کند.
  - جزییات لایه‌های پایین را از دیده‌ها پنهان می‌کند.
- رابط انتزاعی میان سخت‌افزار و نرم‌افزار

Instruction Set Architecture (ISA)

- ISA همراه با رابط سیستم‌عامل

Application binary interface (ABI)

یکی از کلیدی‌ترین واسط‌های بین سطوح تجرید، معماری مجموعه‌ی دستورالعمل (ISA) یا همان واسط بین سخت‌افزار و نرم‌افزار سطح پایین است. چنین واسط مجردی است که این امکان را فراهم آورده تا پیاده‌سازی‌های متعدد با قیمت و کارایی متفاوت از یک سخت‌افزار خاص وجود داشته باشند و همه‌ی آن ها بتوانند نرم‌افزار واحدی را اجرا کنند.

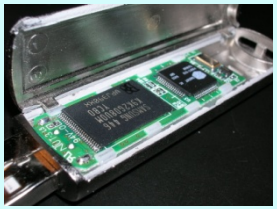
دوازده

Volatile main memory

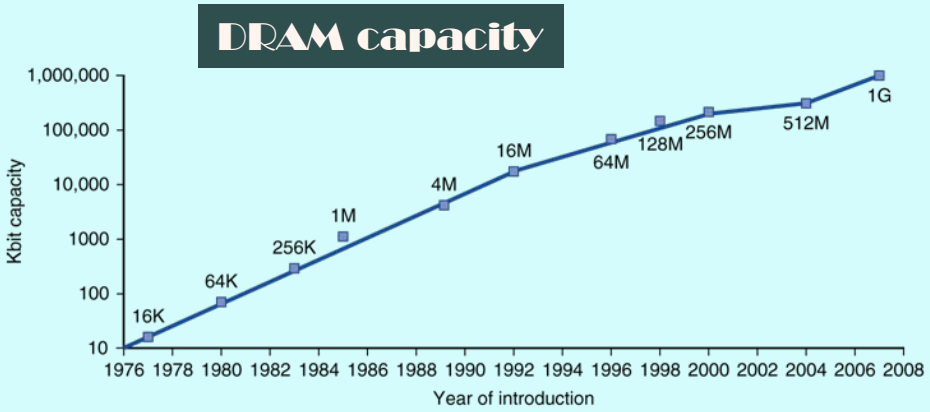
- حافظه‌ی اصلی
  - در صورت قطع منبع تغذیه، حافظه پاک می‌شود.

Non-volatile secondary memory

- حافظه‌ی ثانویه
  - دیسک‌های مغناطیسی
  - سی‌دی و دی‌وی‌دی
  - flash memory



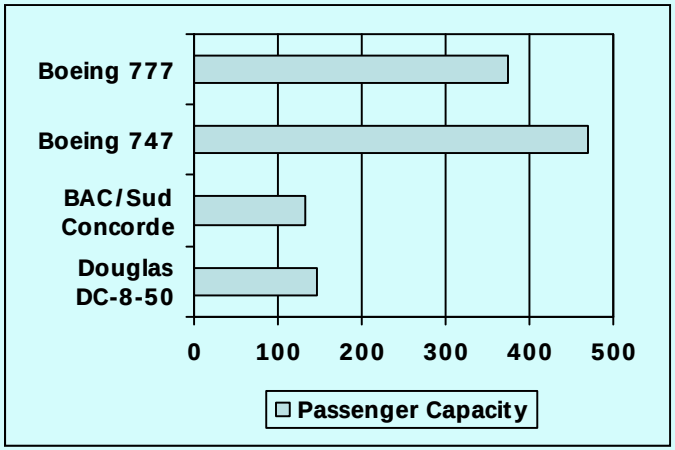
روند به کارگیری فناوری

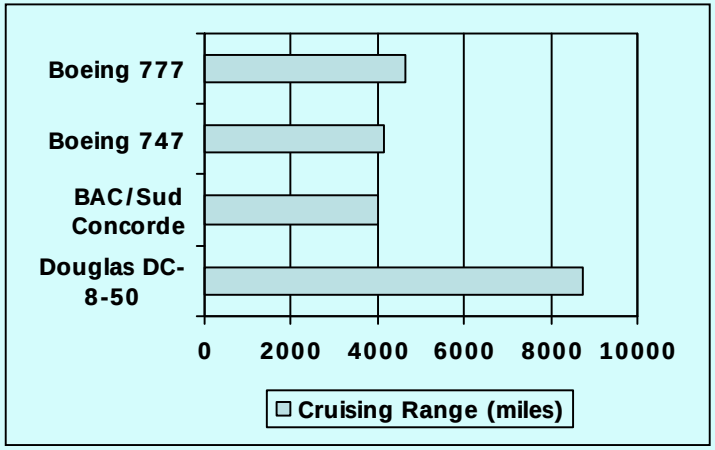


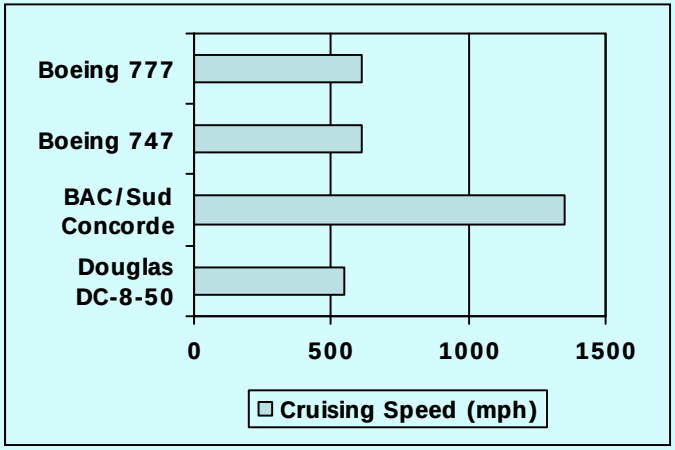
Year	Technology	Relative performance/cost
1951	Vacuum tube	1
1965	Transistor	35
1975	Integrated circuit (IC)	900
1995	Very large scale IC (VLSI)	2,400,000
2005	Ultra large scale IC	6,200,000,000

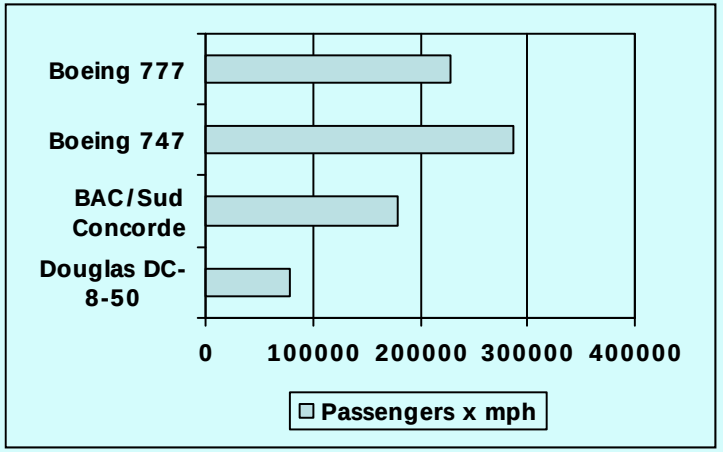
Year	2004	2006	2008	2010	2012
Feature size (nm)	90	65	45	32	22
Intg. Capacity (BT)	2	4	6	16	32

کارایی اجرا


Boeing 777
Boeing 747
BAC/Sud Concorde
Douglas DC-8-50
0
100
200
300
400
500
Passenger Capacity


Boeing 777
Boeing 747
BAC/Sud Concorde
Douglas DC-8-50
0
2000
4000
6000
8000
10000
Cruising Range (miles)


Boeing 777
Boeing 747
BAC/Sud Concorde
Douglas DC-8-50
0
500
1000
1500
Cruising Speed (mph)


Boeing 777
Boeing 747
BAC/Sud Concorde
Douglas DC-8-50
0
100000
200000
300000
400000
Passengers x mph

کارایی سیستم در برابر کارایی پردازنده

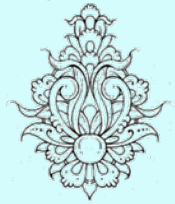
- **زمان پاسخ:**
  - بازه‌ی زمانی که برای تکمیل یک کار صرف می‌شود، شامل پردازش، عملیات I/O و ...
  - بیان‌گر کارایی سیستم می‌باشد.

wall clock time, response time, elapsed time

- **زمان اجرای cpu:**
  - زمانی که صرف پردازش می‌شود. زمان سایر فعالیت‌ها در نظر گرفته نمی‌شود.

cpu execution time

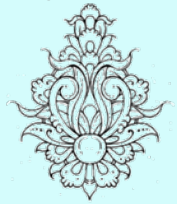
  - شامل user cpu time و system cpu time



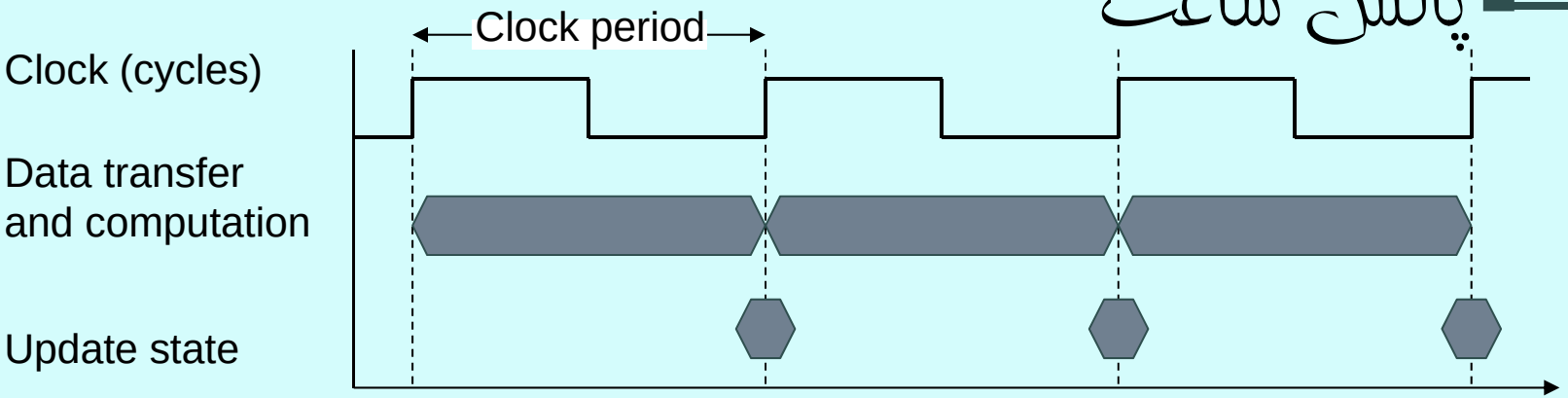
توان عملیاتی در برابر زمان پاسخ

- **زمان پاسخ(اجرا):**
  - زمانی که طول می‌کشد تا یک کامپیوتر کاری را تمام کند.
- **توان:**
  - تعداد کارهایی که در واحد زمان انجام می‌شوند.
- کارایی به صورت معکوس زمان پاسخ تعریف می‌شود.
- X از Y، nبار سریع‌تر است اگر:

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



پالس ساعت

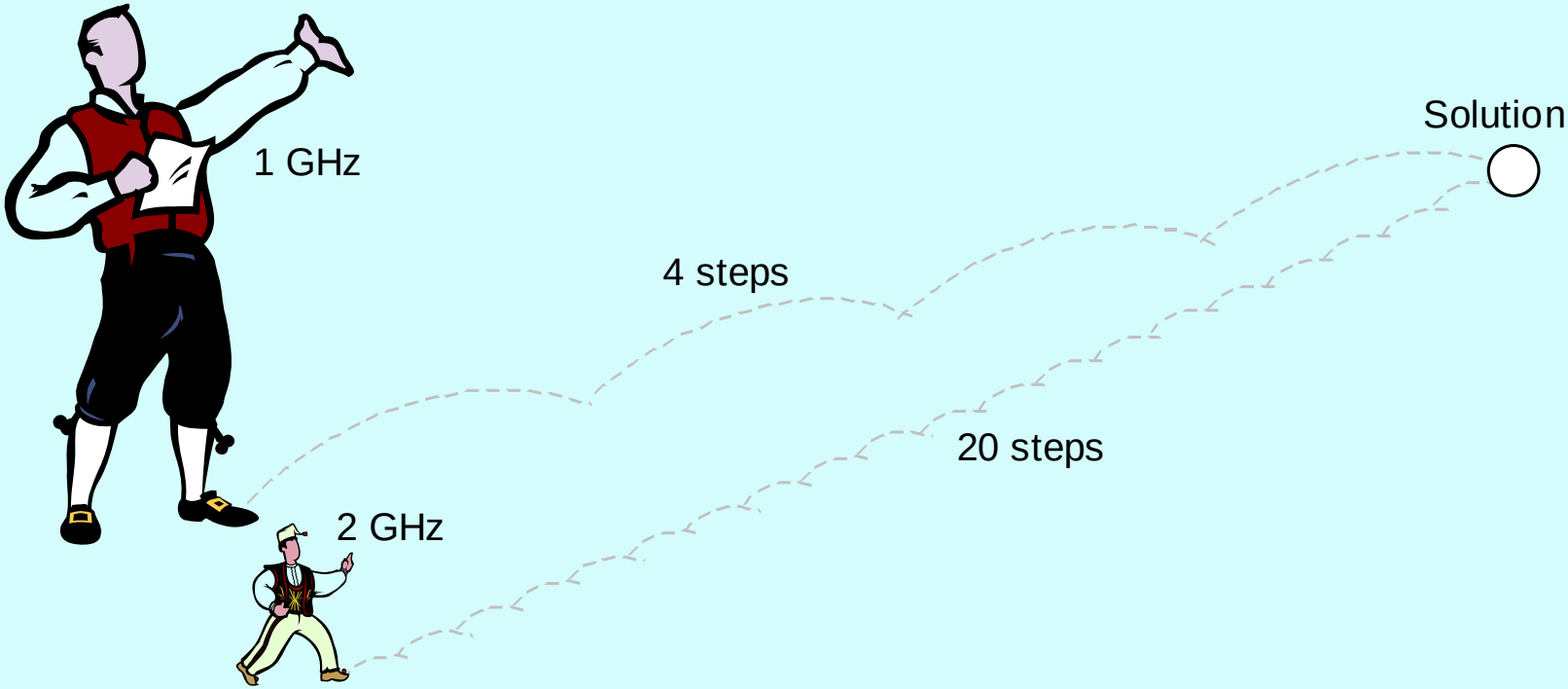


$$\text{CPU Time} = \text{CPU Clock Cycles} \times \text{Clock Cycle Time}$$
$$= \frac{\text{CPU Clock Cycles}}{\text{Clock Rate}}$$

- برای افزایش سرعت cpu
  - فرکانس پالس ساعت را افزایش داد.
  - تعداد پالس به ازای هر دستورالعمل را کاهش داد.

سرعت کامپیوترها

پالس ساعت سریع‌تر به معنای سرعت اجرای بیشتر نیست



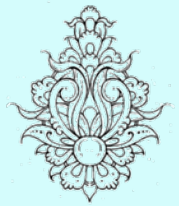
مثال

- کامپیوتر A با 2GHz clock به ده ثانیه برای اجرای برنامه‌ای خاص نیاز دارد. می‌خواهیم کامپیوتر B را به گونه‌ای طراحی کنیم که زمان اجرا را به شش ثانیه تقلیل دهد. با این فرض که در صورت افزایش سرعت پالس ساعت مجبور به تغییر طراحی خواهیم شد به طوری که تعداد سیکل‌های لازم برای اجرای دستورات ۱,۲ برابر می‌شود. فرکانس کامپیوتر B را حساب کنید؟

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

$$\text{Clock Cycles}_A = \text{CPU Time}_A \times \text{Clock Rate}_A$$
$$= 10\text{s} \times 2\text{GHz} = 20 \times 10^9$$

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



کارایی دستورالعمل

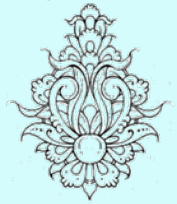
clock cycle per instruction

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

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

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

- تعداد متوسط دستورالعمل توسط
  - برنامه، ISA و کامپایلر تعیین می‌شود
- تعداد سکیل به ازای هر دستورالعمل توسط سخت‌افزار تعیین می‌شود.



مثالی از CPI

- دو کامپیوتر را در نظر بگیرید:
  - کامپیوتر A با پالس ساعت 250ps و CPI=2 برای برنامه‌ای خاص
  - کامپیوتر B با پالس ساعت 500ps و CPI=1.2 برای همان برنامه
- کدام سریع‌ترند؟

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

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

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

کامپیوتر A سریع‌تر است

این قدر