

# Lecture 2 程序的性能

---

## 1. 性能的定义

### 响应时间 **response time**

- 也叫执行时间 **execution time**
- 是计算机完成某任务所需的总时间，包括硬盘访问、内存访问、I/O访问、操作系统开销和CPU执行时间等

### 吞吐率 **throughput**

- 也叫带宽 **bandwidth**
- 是性能的另一度度量参数，表示单位时间内完成的任务数量

### 响应时间和吞吐量的改善

两者都可以改善

- 计算机中的处理器换成更高级的型号
- 在个人移动设备和云之间增加一条二外的网络信道

只改善吞吐率

- 增加多个CPU

只改善响应时间

- 改进网络软件，减少网络通讯的延迟

## 性能

对于某个计算机 X，我们定义性能为

$$Performance_X = \frac{1}{Execution\ Time_X}$$

如果有两台计算机 X 和 Y，X 比 Y 性能更好，则

$$\begin{aligned} Performance_X &> Performance_Y \\ Execution\ Time_Y &> Execution\ Time_X \end{aligned}$$

如果 X 比 Y 快 n 倍，那么 Y 上的执行时间是 X 的 n 倍，即

$$\frac{Performance_X}{Performance_Y} = \frac{Execution\ Time_Y}{Execution\ Time_X} = n$$

## 2. 性能的度量

我们往往把运行子集的任务的时间与一般的响应时间区分开来

### CPU执行时间 CPU execution time

CPU 执行时间， CPU execution time，简称 CPU 时间， CPU time

表示只在**CPU**上花费时间，而不包括 **I/O** 或运行其它程序的时间

### CPU时间的分类

- 用户**CPU**时间 **user CPU time**  
CPU花费在程序的时间
- 系统**CPU**时间 **system CPU time**  
CPU花费在操作系统的时间

### 系统性能 **System performance**

表示空载系统的响应时间

## CPU性能 CPU performance

表示用户CPU时间

### 时钟

几乎所有计算机都用时钟来驱动硬件中发生的各种时间

### 时钟周期 **clock cycle/tick/clock period/clock/cycle**

时钟间隔的时间称为时钟周期 clock cycle

计算机一个时钟周期的时间，通常是处理器时钟，一般为常数

### 时钟频率 **clock rate**

时钟周期的倒数为时钟频率

- $2GHz = 2 \times 10^9$  时钟周期数/秒

### 时钟周期数 **clock cycles/ clock cycle number**

执行的时钟周期数量

## CPU执行时间的计算

### CPU执行时间

$$CPU\ Execution\ Time_P = clock\ cycle\ number_P \times clock\ cycle$$

一个程序的CPU执行时间 = 一个程序时钟周期数 × 时钟周期

$$CPU\ Execution\ Time_P = \frac{clock\ cycles\ number_P}{clock\ rate}$$

$$\text{一个程序的CPU执行时间} = \frac{\text{一个程序时钟周期数}}{\text{时钟频率}}$$

## 例题1 CPU执行时间计算

某程序在一台时钟频率为2GHz的计算机A上运行需要10s。现在将设计一台计算机B，希望将运行时间缩短为6s。计算机的设计者采用的方法是提高时钟频率，但这会影响CPU其余部分的设计,使计算机B运行该程序时需要相当于计算机A的1.2倍的时钟周期数。那么计算机设计者应该将时钟频率提高到多少？

对于计算机A而言

$$10s = \frac{\text{clock cycle number}_A}{2GHz} = \frac{\text{clock cycle number}}{2 \times 10^9 (\text{clock number/s})}$$
$$\text{clock cycle number}_A = 2 \times 10^{10}$$

对于计算机B来说

$$6s = \frac{\text{clock cycle number}_A \times 1.2}{\text{clock rate}_B} = \frac{2 \times 10^{10} \times 1.2}{\text{clock rate}_B}$$
$$\text{clock rate}_B = 4 \times 10^9 (\text{clock number/s}) = 4GHz$$

故需要将B的时钟频率提高到A的2倍

## 指令的性能

计算机是通过执行指令来运行程序的，执行时间依赖于程序的指令数

## CPI clock cycle per instruction/clock cycle number per instructinon

表示执行每条指令所需要的时钟周期数的平均值

## CPU 时钟周期数 CPU clock cycles/CPU clock cycles number

$$CPU \text{ clock cycles number} = \text{Instruction number}_P \times CPI_P$$
$$CPU\text{时钟周期数} = \text{程序的指令数} \times \text{每条指令需要的平均时钟周期数}$$

## CPU 执行时间

$$CPU \text{ execution time}_P = \text{Instruction number}_P \times CPI_P \times \text{clock cycle}$$

一个程序的CPU时间 = 一个程序的指令数 × 每条指令需要的平均时钟周期数 × 时钟周期

$$CPU\ execution\ time_P = \frac{Instruction\ number_P \times CPI_P}{clock\ rate}$$

一个程序的CPU时间 =  $\frac{\text{一个程序的指令数} \times \text{每条指令需要的平均时钟周期数}}{\text{时钟频率}}$

## 例题2 CPI相关计算

假设我们有相同指令集的两种不同实现方式。计算机 A 的时钟周期为 **250ps**,对某程序的 CPI 为 **2.0**，计算机 B 的时钟周期为 **500ps** ,对同样程序的 CPI 为 **1.2**,对于该程序，请问哪台计算机执行的速度更快？快多少？

假设程序执行需要的指令数为  $I$

$$\begin{aligned}CPI_A &= 2.0 \\CPU\ clock\ cycles\ number_A &= I \times 2.0 \\CPU\ execution\ time_A &= I \times 2.0 \times 250ps \\CPI_B &= 1.2 \\CPU\ clock\ cycles\ number_B &= I \times 1.2 \\CPU\ execution\ time_B &= I \times 1.2 \times 500ps\end{aligned}$$

故计算机 A 的执行速度更快，是计算机 B 的1.2倍

## 3. 基准测试

### SPEC CPU基准测试

为了能测量计算机的性能，常见的方法是使用专门用于测量性能的**基准测试程序 benchmark**

Description	Name	Instruction Count x 10 <sup>9</sup>	CPI	Clock cycle time (seconds x 10 <sup>-9</sup> )	Execution Time (seconds)	Reference Time (seconds)	SPECratio
Interpreted string processing	perl	2252	0.60	0.376	508	9770	19.2
Block-sorting compression	bzip2	2390	0.70	0.376	629	9650	15.4
GNU C compiler	gcc	794	1.20	0.376	358	8050	22.5
Combinatorial optimization	mcf	221	2.66	0.376	221	9120	41.2
Go game (AI)	go	1274	1.10	0.376	527	10490	19.9
Search gene sequence	hmmer	2616	0.60	0.376	590	9330	15.8
Chess game (AI)	sjeng	1948	0.80	0.376	586	12100	20.7
Quantum computer simulation	libquantum	659	0.44	0.376	109	20720	190.0
Video compression	h264avc	3793	0.50	0.376	713	22130	31.0
Discrete event simulation library	omnetpp	367	2.10	0.376	290	6250	21.5
Games/path finding	astar	1250	1.00	0.376	470	7020	14.9
XML parsing	xalancbmk	1045	0.70	0.376	275	6900	25.1
Geometric mean	-	-	-	-	-	-	25.7

几何平均值的计算公式为

$$\sqrt[n]{\prod_{i=1}^n Execution\ time\ ratio}$$

## SPEC 功耗基准测试

Target Load %	Performance (ssj_ops)	Average Power (watts)
100%	865,618	258
90%	786,688	242
80%	698,051	224
70%	607,826	204
60%	521,391	185
50%	436,757	170
40%	345,919	157
30%	262,071	146
20%	176,061	135
10%	86,784	121
0%	0	80
Overall Sum	4,787,166	1922
$\sum ssj\_ops / \sum power =$		2490

性能采用吞吐率来测量，SPEC采用单个的指标来进行衡量

$$overall\ ssj\_ops\ per\ watt = \frac{\sum_{i=0}^{10} ssj\_ops_i}{\sum_{i=0}^{10} power_i}$$

- $ssj\_ops_i$ : 工作负载在每10%增量处的性能
- $power$ : 对应功耗