Computer Architecture
Performance

Moniruzzaman

Adjunct Lecturer
North Western University, Khulna-9000

Confession

- ❖ Most of the materials have been collected from Internet.
- ❖ Images are taken from Internet.
- ❖ Various books are used to make these slides.
- ❖ Various slides are also used.
- ❖ References & credit:
  - ➢ Atanu Shome, Assistant Professor, CSE, KU.
  - ➢ Computer Organization and Design: the Hardware/Software Interface - Textbook by David A Patterson and John L. Hennessy.
  - ➢ Computer Organization and Architecture - Book by William Stallings

Which Plane is Better?

Airplane	Passenger capacity	Cruising range (miles)	Cruising speed (m.p.h.)	Passenger throughput (passengers × 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

Which One is Better?

Two computers run the same program in 5 seconds and 7 seconds.

First program can satisfy 10 services at a time. Second one can provides service to 15.

Terms

Response Time / Execution Time:

Time between start to end of a task

Throughput:

Total amount of work complete in a given time

Performance vs Execution

$$\text{Performance}_X = \frac{1}{\text{Execution time}_X}$$

This means that for two computers X and Y, if the performance of X is greater than the performance of Y, we have

$$\text{Performance}_X > \text{Performance}_Y$$

$$\frac{1}{\text{Execution time}_X} > \frac{1}{\text{Execution time}_Y}$$

$$\text{Execution time}_Y > \text{Execution time}_X$$

$$\frac{\text{Performance}_X}{\text{Performance}_Y} = n$$

If X is n times faster than Y, then the execution time on Y is n times longer than it is on X:

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

Terms

If computer X runs a program in **10 seconds** and computer Y runs the same program in **15 seconds**, how much faster is X than Y?

Problem

If computer X runs a program in 10 seconds and computer Y runs the same program in 15 seconds, how much faster is X than Y?

- $$\frac{Performance_X}{Performance_Y} = \frac{Execution_Y}{Execution_X}$$

- $$\frac{Performance_X}{Performance_Y} = \frac{15}{10} = 1.5$$

$$Performance_X = Performance_Y * 1.5$$

Measuring Performance

Terms

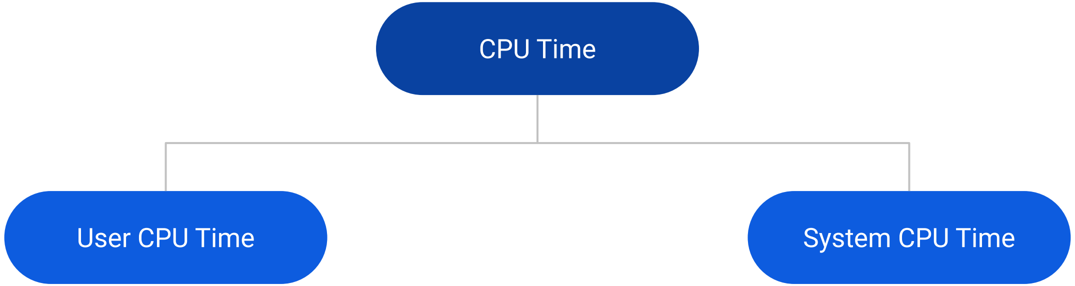
Execution Time = Seconds per Program
Again

Time(wall clock time, response time, or elapsed time) = total time to complete a task, including disk accesses, memory accesses, input/output (I/O) activities, operating system overhead—everything.

CPU Time

Multiple programs run at the same time

Throughput OR Less Execution Time



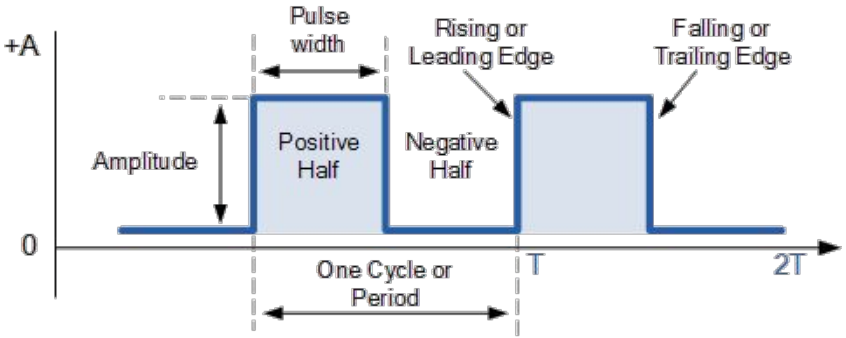
Clock Cycles

Clock cycles (ticks, clock ticks, clock periods, clocks, cycles)

Time for one clock period, usually of the processor clock, which runs at a constant rate.

Clock Period

The length of each clock cycle



Clock Rate

the frequency at which a processor's clock generator creates pulses.

Clock rate is also known as **clock speed** or **clock frequency**. It's measured in hertz (Hz) or gigahertz (GHz).

Clock rate is a measure of how quickly a computer's central processing unit (CPU) can execute instructions.

Clock Cycle
vs
Clock Rate

$$Clock\ Rate = \frac{1}{Clock\ Cycle}$$

1 GHz processor has a cycle time of **1.0 ns**
and
4 GHz processor has a cycle time of **0.25 ns**.

- A **4GHz** processor performs **4,000,000,000** clock cycles per second.
- Computer processors can execute one or more instructions per clock cycle, depending on the type of processor.

CPU Execution Time Formula

$$\text{CPU execution time for a program} = \text{CPU clock cycles for a program} \times \text{Clock cycle time}$$

Alternatively, because clock rate and clock cycle time are inverses,

$$\text{CPU execution time for a program} = \frac{\text{CPU clock cycles for a program}}{\text{Clock rate}}$$

Decimal term	Abbreviation	Value	Binary term	Abbreviation	Value	% Larger
kilobyte	KB	10^3	kibibyte	KiB	2^{10}	2%
megabyte	MB	10^6	mebibyte	MiB	2^{20}	5%
gigabyte	GB	10^9	gibibyte	GiB	2^{30}	7%
terabyte	TB	10^{12}	tebibyte	TiB	2^{40}	10%
petabyte	PB	10^{15}	pebibyte	PiB	2^{50}	13%
exabyte	EB	10^{18}	exbibyte	EiB	2^{60}	15%
zettabyte	ZB	10^{21}	zebibyte	ZiB	2^{70}	18%
yottabyte	YB	10^{24}	yobibyte	YiB	2^{80}	21%

Improving Performance ?

Our favourite program runs in **10 seconds** on computer A, which has a **2 GHz** clock. We are trying to help a computer designer build a computer, B, which will run this program in **6 seconds**. The designer has determined that a substantial increase in the clock rate is possible, but this increase will affect the rest of the CPU design, causing computer B to require **1.2 times** as many clock cycles as computer A for this program. What clock rate should we tell the designer to target?

Problem

Our favourite program runs in **10 seconds** on computer A, which has a **2 GHz** clock. We are trying to help a computer designer build a computer, B, which will run this program in **6 seconds**. The designer has determined that a substantial increase in the clock rate is possible, but this increase will affect the rest of the CPU design, causing computer B to require **1.2 times** as many clock cycles as computer A for this program. What clock rate should we tell the designer to target?

Let's first find the number of clock cycles required for the program on A:

$$\text{CPU time}_A = \frac{\text{CPU clock cycles}_A}{\text{Clock rate}_A}$$

$$10 \text{ seconds} = \frac{\text{CPU clock cycles}_A}{2 \times 10^9 \frac{\text{cycles}}{\text{second}}}$$

$$\text{CPU clock cycles}_A = 10 \text{ seconds} \times 2 \times 10^9 \frac{\text{cycles}}{\text{second}} = 20 \times 10^9 \text{ cycles}$$

CPU time for B can be found using this equation:

$$\text{CPU time}_B = \frac{1.2 \times \text{CPU clock cycles}_A}{\text{Clock rate}_B}$$

$$6 \text{ seconds} = \frac{1.2 \times 20 \times 10^9 \text{ cycles}}{\text{Clock rate}_B}$$

$$\text{Clock rate}_B = \frac{1.2 \times 20 \times 10^9 \text{ cycles}}{6 \text{ seconds}} = \frac{0.2 \times 20 \times 10^9 \text{ cycles}}{\text{second}} = \frac{4 \times 10^9 \text{ cycles}}{\text{second}} = 4 \text{ GHz}$$

To run the program in 6 seconds, B must have twice the clock rate of A.

Instruction Performance

CPI

Clock Cycles per Instruction

Average number of clock cycles per instruction for a program or program fragment.

CPU clock cycles = Instructions for a program X CPI

$$\text{CPU execution time for a program} = \text{CPU clock cycles for a program} \times \text{Clock cycle time}$$

Alternatively, because clock rate and clock cycle time are inverses,

$$\text{CPU execution time for a program} = \frac{\text{CPU clock cycles for a program}}{\text{Clock rate}}$$

Problem

Suppose we have two implementations of the same instruction set architecture. Computer **A** has a clock cycle time of **250 ps** and a CPI of **2.0** for some program, and computer **B** has a clock cycle time of **500 ps** and a CPI of **1.2** for the same program. Which computer is faster for this program and by how much?

Problem

Suppose we have two implementations of the same instruction set architecture. Computer **A** has a clock cycle time of **250 ps** and a CPI of **2.0** for some program, and computer **B** has a clock cycle time of **500 ps** and a CPI of **1.2** for the same program. Which computer is faster for this program and by how much?

We know that each computer executes the same number of instructions for the program; let's call this number I. First, find the number of processor clock cycles for each computer:

$$\text{CPU clock cycles}_A = I \times 2.0$$
$$\text{CPU clock cycles}_B = I \times 1.2$$

Now we can compute the CPU time for each computer:

$$\begin{aligned}\text{CPU time}_A &= \text{CPU clock cycles}_A \times \text{Clock cycle time} \\ &= I \times 2.0 \times 250\text{ ps} = 500 \times I\text{ ps}\end{aligned}$$

Likewise, for B:

$$\text{CPU time}_B = I \times 1.2 \times 500\text{ ps} = 600 \times I\text{ ps}$$

Clearly, computer A is faster. The amount faster is given by the ratio of the execution times:

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

We can conclude that computer A is 1.2 times as fast as computer B for this program.

Problem

A compiler designer is trying to decide between two code sequences for a computer. The hardware designers have supplied the following facts:

	CPI for each instruction class		
	A	**B**	**C**
CPI	1	2	3

For a particular high-level language statement, the compiler writer is considering two code sequences that require the following instruction counts:

	Instruction counts for each instruction class		
Code sequence	**A**	**B**	**C**
1	2	1	2
2	4	1	1

Which code sequence executes the most instructions? Which will be faster? What is the CPI for each sequence?

Comparing Code Segments

A compiler designer is trying to decide between two code sequences for a particular computer. The hardware designers have supplied the following facts:

	CPI for each instruction class		
	A	B	C
CPI	1	2	3

For a particular high-level language statement, the compiler writer is considering two code sequences that require the following instruction counts:

Code sequence	Instruction counts for each instruction class		
	A	B	C
1	2	1	2
2	4	1	1

Which code sequence executes the most instructions? Which will be faster? What is the CPI for each sequence?

Sequence 1 executes $2+1+2=5$ instructions. Sequence 2 executes $4+1+1=6$ instructions. Therefore, sequence 1 executes fewer instructions.

We can use the equation for CPU clock cycles based on instruction count and CPI to find the total number of clock cycles for each sequence:

$$\text{CPU clock cycles} = \sum_{i=1}^{n} (\text{CPI}_i \times \text{C}_i)$$

This yields

$$\text{CPU clock cycles}_1 = (2\times 1) + (1\times 2) + (2\times 3) = 2+2+6 = 10 \text{ cycles}$$

$$\text{CPU clock cycles}_2 = (4\times 1) + (1\times 2) + (1\times 3) = 4+2+3 = 9 \text{ cycles}$$

So code sequence 2 is faster, even though it executes one extra instruction. Since code sequence 2 takes fewer overall clock cycles but has more instructions, it must have a lower CPI. The CPI values can be computed by

$$\text{CPI} = \frac{\text{CPU clock cycles}}{\text{Instruction count}}$$

$$\text{CPI}_1 = \frac{\text{CPU clock cycles}_1}{\text{Instruction count}_1} = \frac{10}{5} = 2.0$$

$$\text{CPI}_2 = \frac{\text{CPU clock cycles}_2}{\text{Instruction count}_2} = \frac{9}{6} = 1.5$$

Do

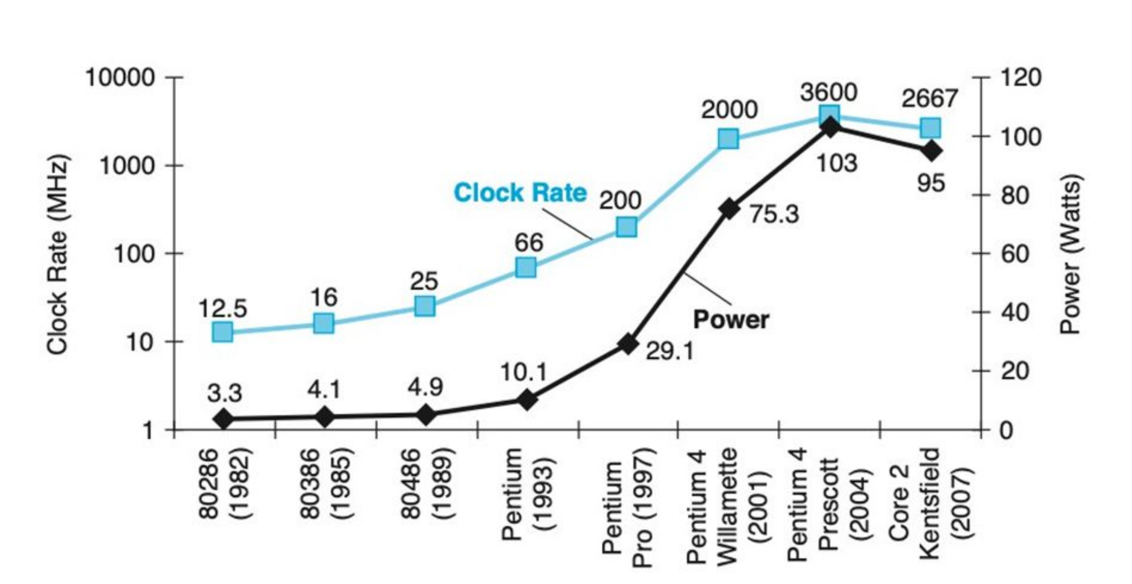
A given application written in Java runs **15 seconds** on a desktop processor. A new Java compiler is released that requires only **0.6** as many instructions as the old compiler. Unfortunately, it increases the **CPI** by **1.1**. How fast can we expect the application to run using this new compiler? Pick the right answer from the three choices below:

a. $\frac{15 \times 0.6}{1.1} = 8.2\,\text{sec}$

b. $15 \times 0.6 \times 1.1 = 9.9\,\text{sec}$

c. $\frac{1.5 \times 1.1}{0.6} = 27.5\,\text{sec}$

Power Wall



Clock Rate (MHz)
10000
1000
100
10
1
Power (Watts)
120
100
80
60
40
20
0
Clock Rate
Power
12.5
16
25
66
200
2000
3600
2667
3.3
4.1
4.9
10.1
29.1
75.3
103
95
80286 (1982)
80386 (1985)
80486 (1989)
Pentium (1993)
Pentium Pro (1997)
Pentium 4 Willamette (2001)
Pentium 4 Prescott (2004)
Core 2 Kentsfield (2007)

Pitfall: Expecting the improvement of one aspect of a computer to increase overall performance by an amount proportional to the size of the improvement.

Amdahl's law: A rule stating that the performance enhancement possible with a given **improvement is limited by the amount that the improved feature is used**.

Problem

Suppose a program runs in **100 seconds** on a computer, with multiply operations responsible for **80 seconds** of this time. How much do I have to improve the speed of multiplication if I want my program to run five times faster?

Problem

Suppose a program runs in **100 seconds** on a computer, with multiply operations responsible for **80 seconds** of this time. How much do I have to improve the speed of multiplication if I want my program to run five times faster?

$$\text{Execution time after improvement} = \frac{\text{Execution time affected by improvement}}{\text{Amount of improvement}} + \text{Execution time unaffected}$$

For this problem:

$$\text{Execution time after improvement} = \frac{80 \text{ seconds}}{n} + (100 - 80 \text{ seconds})$$

Since we want the performance to be five times faster, the new execution time should be 20 seconds, giving

$$20 \text{ seconds} = \frac{80 \text{ seconds}}{n} + 20 \text{ seconds}$$

$$0 = \frac{80 \text{ seconds}}{n}$$

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

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

Thank You