LO1: Computer Performance

(Hiệu năng máy tính)

- Possible measures:
  - response time (thời gian đáp ứng) – time elapsed between start and end of a program
  - throughput (thông lượng) – amount of work done in a fixed time
- The two measures are usually linked
  - A faster processor will improve both
  - More processors will likely only improve throughput
  - Some policies will improve throughput and worsen response time
- What influences performance?

Execution Time (thời gian thực thi)

Consider a system X executing a fixed workload W

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

Execution time = response time = wall clock time

- Note that this includes time to execute the workload as well as time spent by the operating system co-ordinating various events

The UNIX "time" command breaks up the wall clock time as user and system time

Speedup (tăng tốc độ) and Improvement (cải thiện hiệu năng)

- System X executes a program in 10 seconds, system Y executes the same program in 15 seconds

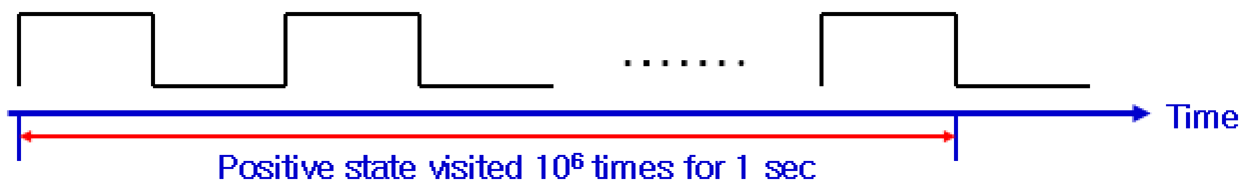
- System X is 1.5 times faster than system Y, it means the speed up of system X over system Y is 1.5 (the ratio) = perf X / perf Y = exectime Y / exectime X

- The performance improvement of X over Y is
  1.5 -1 = 0.5 = 50% = (perf X – perf Y) / perf Y = speedup – 1

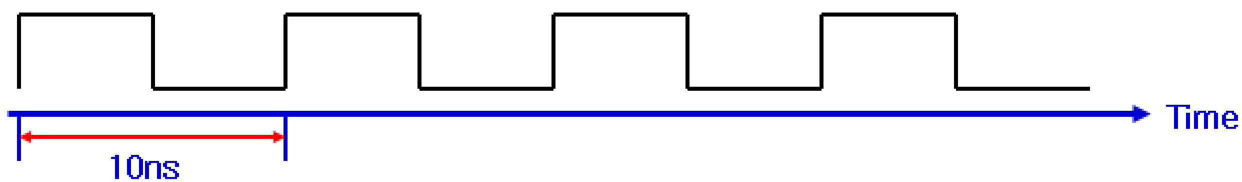
- The execution time reduction for system X, compared to Y is (15-10) / 15 = 33%
  The execution time increase for Y, compared to X is
  (15-10) / 10 = 50%

Clock cycle time (chu kỳ đồng hồ máy tính)

- **A machine is running at 100MHz**
  - Clock rate = 100MHz = 100 * 10^6 cycles / sec



  - Clock cycle time = 1/(100*10^6) cycles / sec = 10ns



Learning outcome 1 Performance Metrics

Application software
(Python, C languages)

Systems software
(OS, compiler)

Hardware

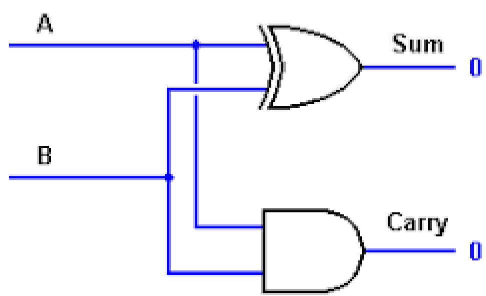
a = b + c;

↓ Compiler

add $16, $15, $14

↓ Assembler

000000101100000…



INPUTS		OUTPUTS	
A	B	CARRY	SUM
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Performance Equation - I

CPU execution time = CPU clock cycles x Clock cycle time

Clock cycle time = 1 / Clock speed

If a processor has a frequency of 3 GHz, the clock ticks 3 billion times in a second – as we'll soon see, with each clock tick, one or more/less instructions may complete

If a program runs for 10 seconds on a 3 GHz processor, how many clock cycles did it run for?

If a program runs for 2 billion clock cycles on a 1.5 GHz processor, what is the execution time in seconds?

Performance Equation

CPU clock cycles = number of instrs x avg clock cycles per instruction (CPI)

(CPI: trung bình tốc độ trên instruction)

Substituting in previous equation,

Execution time = clock cycle time x number of instrs x avg CPI

If a 2 GHz processor graduates an instruction every third cycle, how many instructions are there in a program that runs for 10 seconds?

Factors Influencing Performance

Execution time = clock cycle time x number of instrs x avg CPI

- Clock cycle time: manufacturing process (how fast is each transistor), how much work gets done in each pipeline stage (more on this later)
- Number of instrs: the quality of the compiler and the instruction set architecture
- CPI: the nature of each instruction and the quality of the architecture implementation

Example

Execution time = clock cycle time x number of instrs x avg CPI

Which of the following two systems is better?

- A program is converted into 4 billion MIPS instructions by a compiler ; the MIPS processor is implemented such that each instruction completes in an average of 1.5 cycles and the clock speed is 1 GHz

- The same program is converted into 2 billion x86 instructions; the x86 processor is implemented such that each instruction completes in an average of 6 cycles and the clock speed is 1.5 GHz

Power and Energy (công suất tiêu thụ và năng lượng của máy tính)

- Total power = dynamic power + leakage power
- Dynamic power α activity x capacitance x voltage^2 x frequency
- Leakage power α voltage
- Energy = power x time
  (joules) (watts) (sec)

Example Problem

- A 1 GHz processor takes 100 seconds to execute a program, while consuming 70 W of dynamic power and 30 W of leakage power. Does the program consume less energy in Turbo boost mode when the frequency is increased to 1.2 GHz?

Example Problem

- A 1 GHz processor takes 100 seconds to execute a program, while consuming 70 W of dynamic power and 30 W of leakage power. Does the program consume less energy in Turbo boost mode when the frequency is increased to 1.2 GHz?

  Normal mode energy = 100 W x 100 s = 10,000 J
  Turbo mode energy = (70 x 1.2 + 30) x 100/1.2 = 9,500 J

  Note: Frequency only impacts dynamic power, not leakage power. We assume that the program's CPI is unchanged when frequency is changed, i.e., exec time varies linearly with cycle time.

Questions & Answers