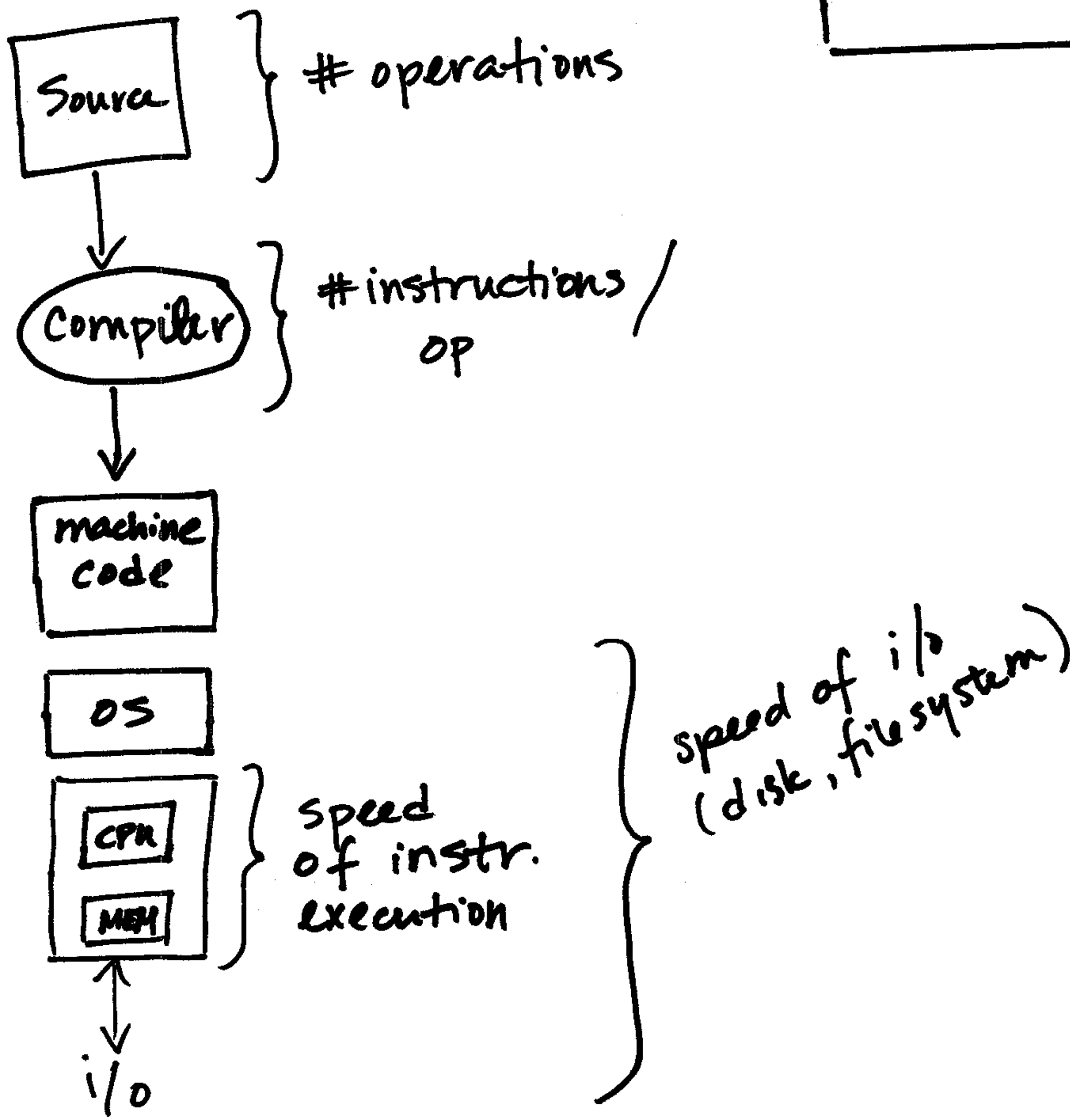
CPU Performance Notes

4/12/10



Latency
- how long does a particular task take?
- response time
- time / work

Throughput
- total work done per unit time
- work / time

swap for a faster processor?
L & T improve ($L\downarrow$, $T\uparrow$)

add more processors (same speed)?
only T will improve

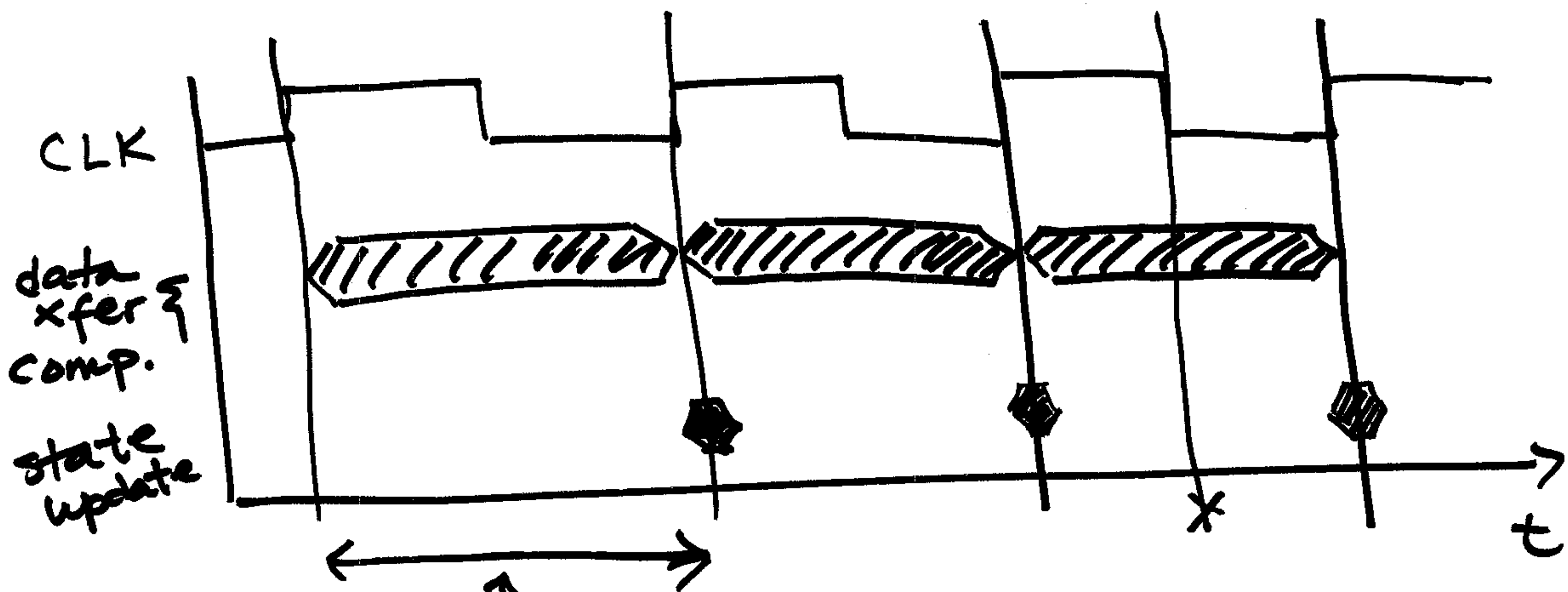
Elapsed Time: total response time for a job

→ this & other jobs on CPU
→ I/O
→ overhead
→ idle time

CPU Time: → total time spent executing this task

Elapsed $\geq$ CPU

CPU hardware governed by constant rate clock:



Clock period [s/cc] e.g. 250 ps

clock frequency [cc/s] e.g. 2 GHz

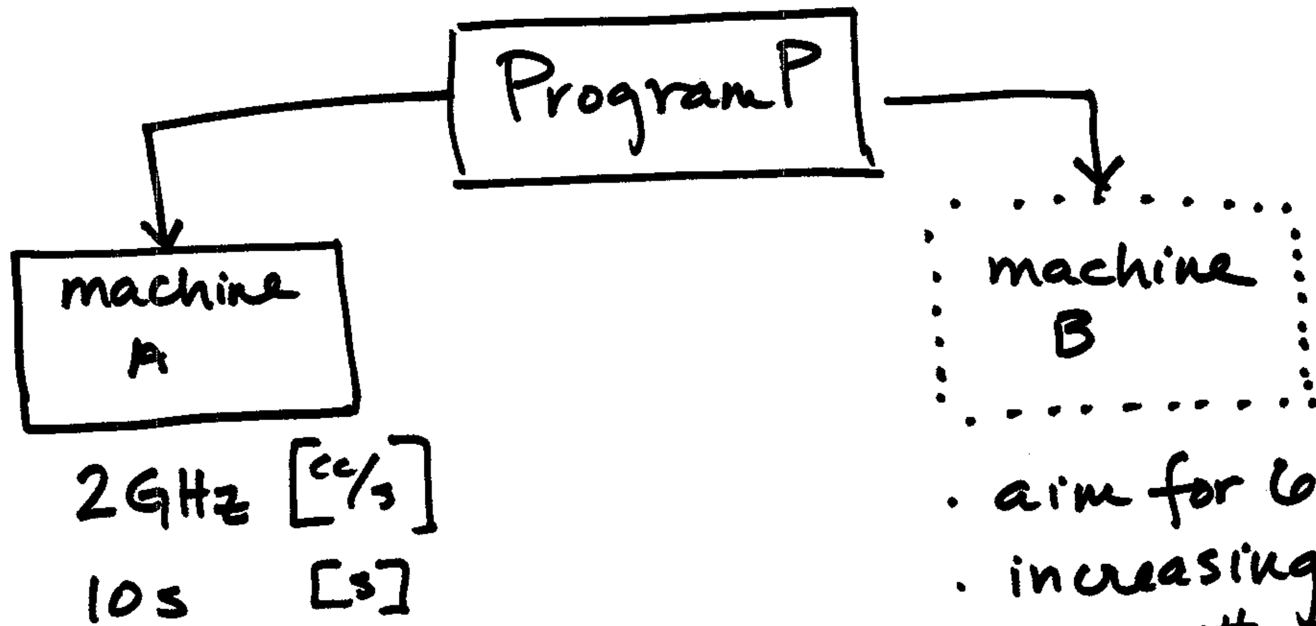
$$\underset{[s]}{\text{CPU Time}} = \underset{[cc]}{\text{CPU Clock Cycles}} \times \underset{[s/cc]}{\text{Clock Cycle Time}}$$

$$= \underset{[cc]}{\text{CPU Clock Cycles}} \Big/ \underset{[cc/s]}{\text{Frequency}}$$

To improve performance:

1. reduce # clk cycles
2. increase clock rate

Example problem:



- aim for 6s CPU time
- increasing clk rate → 1.2x # cycles

How fast must mach. B's clock be?

Example Solution

$$\underset{[cc/s]}{\text{Freq}_B} = \frac{\text{Cycles}_B \; [cc]}{\text{Time}_B \; [s]} = \frac{1.2 \times \text{Cycles}_A}{6s}$$

$$\underset{[cc]}{\text{Cycles}_A} = \underset{[s]}{\text{Time}_A} \cdot \underset{[cc/s]}{\text{Freq}_A}$$

$$= 10s \cdot 2 \times 10^9 = 2 \times 10^{10} \; [cc]$$

$$\text{Freq}_B = \frac{1.2 \times 2 \times 10^{10}}{6s} = \frac{2.4 \times 10^{10}}{6} = 4 \times 10^9$$

Instruction Count (dynamic) (IC)

Average Cycles Per Instruction (CPI)

- typically compute a weighted avg based on instruction mix

(beware: static v. dynamic)

$$\underset{[cc]}{\text{Clock Cycles}} = \underset{[ins]}{IC} \times \underset{[cc/ins]}{CPI}$$

$$\underset{[s]}{\text{CPU Time}} = \left(\underset{[ins]}{IC} \times \underset{[cc/ins]}{CPI}\right) \times \underset{[s/cc]}{\text{CycleTime}}$$

$$= \left(\underset{[ins]}{IC} \times \underset{[cc/ins]}{CPI}\right) \Big/ \underset{[cc/s]}{\text{Frequency}}$$

*CPi Problem

machine A	machine B
cycle time = 250ps	cycle time = 500 ps
CPi = 2.0	CPi = 1.2

(same ISA)

Which is faster?

Soln

$Time_A = \text{~~InsCount~~ IC} \times CPi_A \times CycleTime_A$

$= IC \times 2.0 \times 250\,ps = IC \times 500\,ps$

$Time_B = IC \times CPI_B \times CycleTime_B$

$= IC \times 1.2 \times 500\,ps = IC \times 600\,ps$

← A is faster

$$\frac{CPU\ Time_B}{CPU\ Time_A} = \frac{IC \times 600\,ps}{IC \times 500\,ps} = 1.2$$

← by this much "A is 1.2x faster than B"

Amdahl's Law

$$T_{improved} = \frac{T_{affected}}{\text{improvement factor}} + T_{unaffected}$$

* **Example**: On machine A, multiplication accounts for 80s out of 100s ~~total~~ CPU Time. How much improvement in multiplication to make the program 5 times as fast (5x speedup overall).

Soln: infinite improvement s.t. multiplication accounts for 0s of execution.

Corollary of Amdahl's: make the common case fast.

* Performance Summary

$$\text{CPUTime} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Cycle}}$$

- Algorithm, language & compiler affect these
- ISA affects all three
- Performance depends on all terms