# Hardware Technologies

Dr. Vincent C. Emeakaroha

18-01-2017

vc.emeakaroha@cs.ucc.ie

# Computer Components

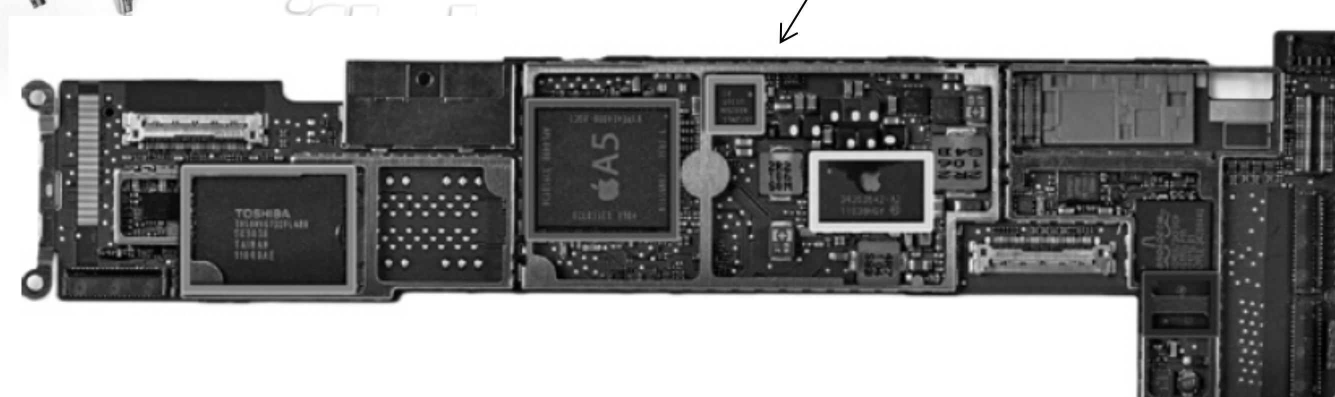
- Classic components
  - Inputs
  - Outputs
  - Memory
  - Processor (Datapath and Control units)
- Same components for all kinds of computer
  - Desktop, server, supercomputer, embedded

# Opening the Box

Capacitive multitouch LCD screen

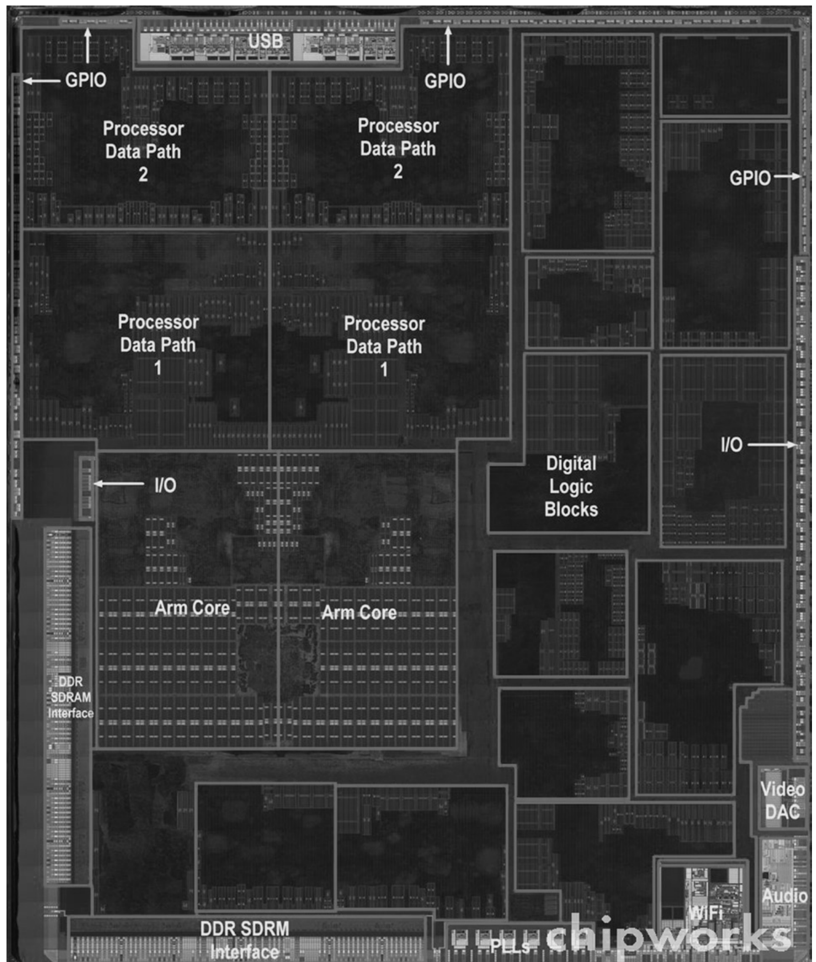
3.8 V, 25 Watt-hour battery

Computer board



# Microprocessor Package

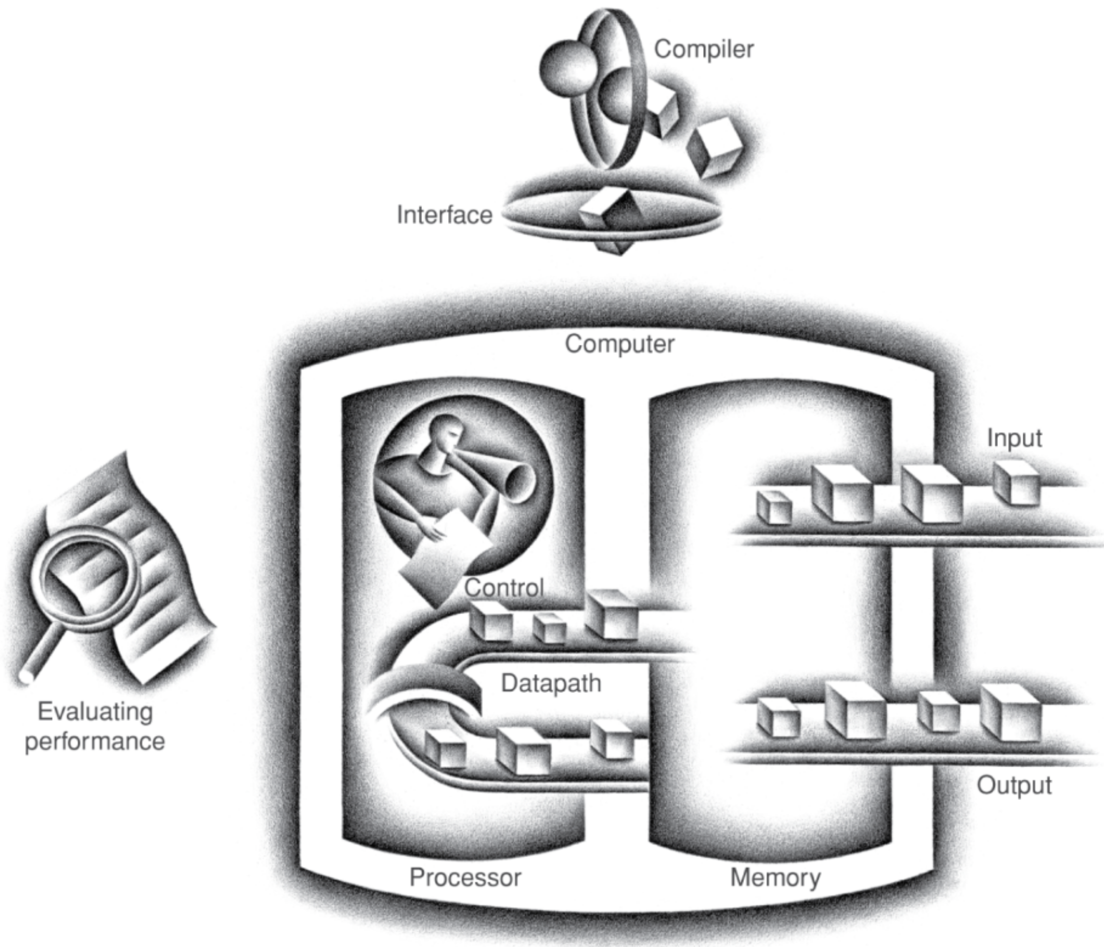
- Apple A5



# Inside the Processor

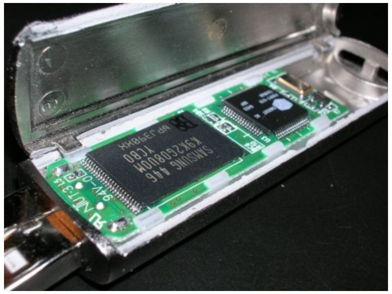
- Datapath:
  - performs operations on data
- Control:
  - sequences datapath, memory access
- Cache memory
  - Small fast SRAM memory for immediate access to data

# Hardware Operation Overview

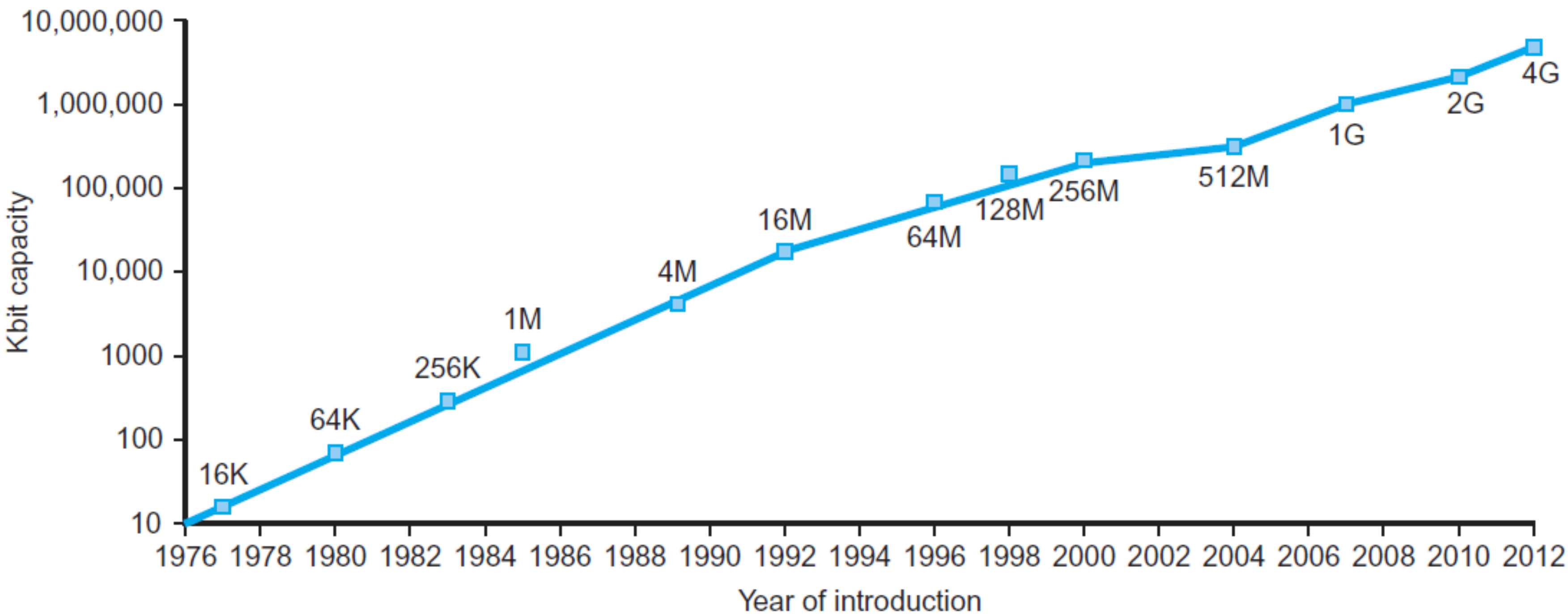


# Memory

- Volatile main memory
  - Loses instructions and data when power off
- Non-volatile secondary memory
  - Magnetic disk
  - Flash memory
  - Optical disk (CDROM, DVD)



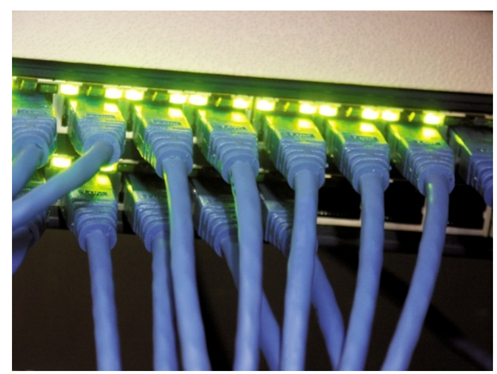

# DRAM Growth Trends




- Electronics technology continues to evolve
  - Increased capacity and performance
  - Reduced cost

# Networks

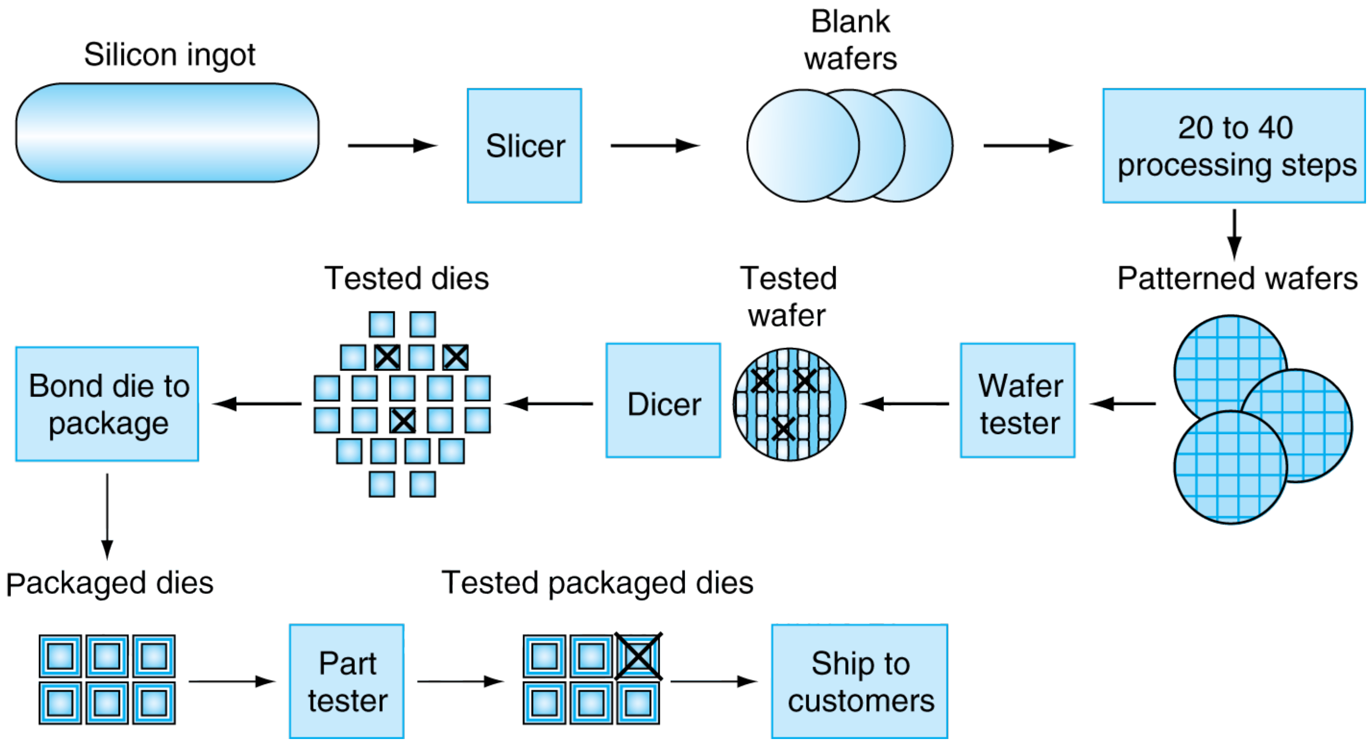
- Communication, resource sharing, nonlocal access
  - Printers, remote login
- Local area network (LAN): Ethernet
- Wide area network (WAN): the Internet
- Wireless network: WiFi, Bluetooth

# IC Manufacturing Technology

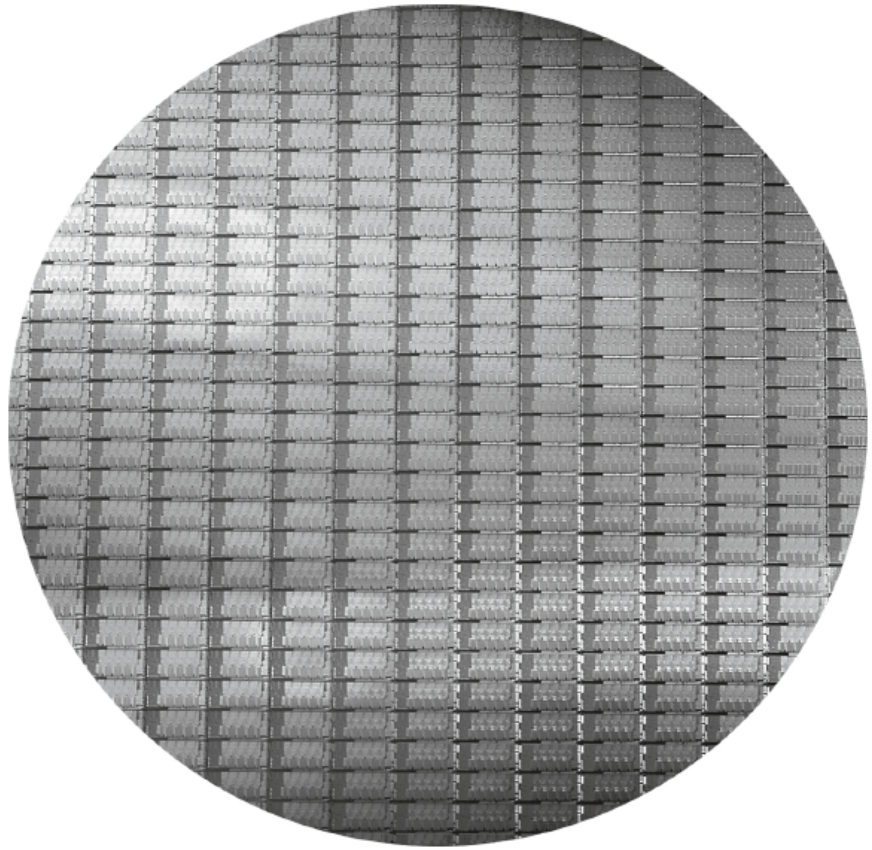
- Silicon:
  - Found in sand
  - Poor electricity conductor, thus the name "semiconductor"
- Add materials to transform properties:
  - Conductors
  - Insulators
  - Switch (transistors)

# Manufacturing ICs




- Yield: proportion of working dies per wafer
  - In 2012: 32 nanometer is the standard

# Intel Core i7 Wafer



- 300mm wafer, 280 chips, 32nm technology
- Each chip is 20.7 x 10.5 mm

# Integrated Circuit Cost

$$\text{Cost per die} = \frac{\text{Cost per wafer}}{\text{Dies per wafer} \times \text{Yield}}$$

$$\text{Dies per wafer} \approx \text{Wafer area} / \text{Die area}$$

$$\text{Yield} = \frac{1}{(1 + (\text{Defects per area} \times \text{Die area}/2))^2}$$

- Nonlinear relation to area and defect rate
  - Wafer cost and area are fixed
  - Defect rate determined by manufacturing process
  - Die area determined by architecture and circuit design

# System Performance

- Response time
  - How long it takes to do a task
  - Also referred to as execution time
  - Personal mobile device
- Throughput
  - Total work done per unit time
    - e.g., tasks/transactions/… per hour
  - Focus of servers
- How are response time and throughput affected by
  - Replacing the processor with a faster version?
  - Adding more processors?

# Relative Performance

- Define Performance = 1/Execution Time
- "X is $n$ time faster than Y"

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

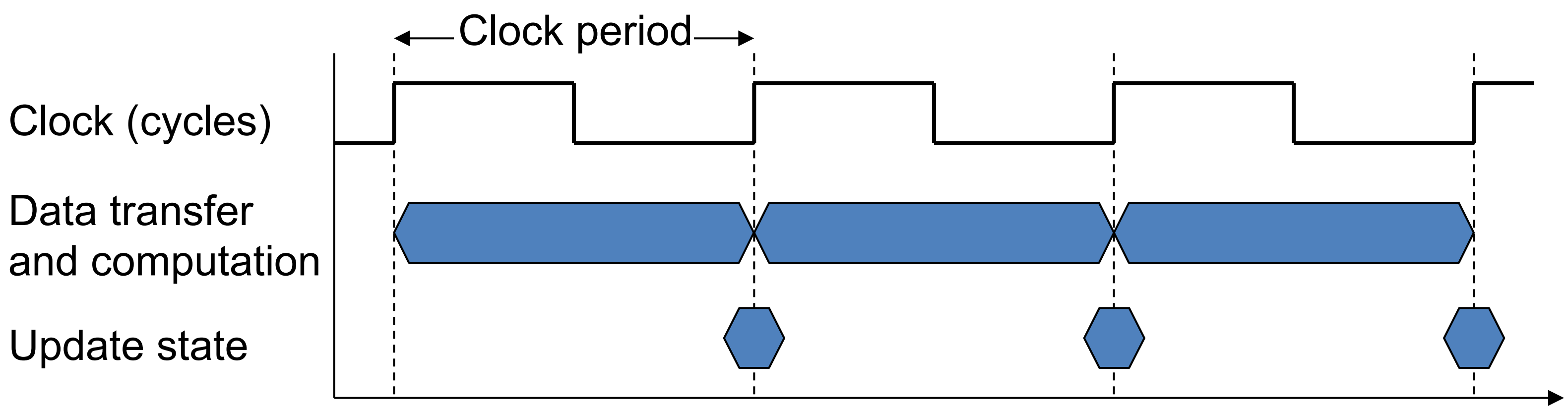
- Example: time taken to run a program
  - 10s on A, 15s on B. How much faster is A than B?
- Execution $\text{Time}_B$ / Execution $\text{Time}_A$ = 15s / 10s = 1.5s
  - So A is 1.5 times faster than B

# Measuring Performance

- Elapsed time
  - Total time to complete a task, including all aspects
    - Processing, I/O, OS overhead, idle time
  - Determines system performance
- CPU time
  - Time spent processing a given job
    - Does not include: I/O time, other jobs' shares
  - Comprises user CPU time and system CPU time
  - Different programs are affected differently by CPU and system performance

# CPU Clocking

- Operation of digital hardware governed by a constant-rate clock




- Clock period: duration of a clock cycle
  - E.g., 250ps = 0.25ns = $250 \times 10^{-12}$s
- Clock frequency (rate): cycles per second
  - E.g., 4.0GHz = 4000MHz = $4.0 \times 10^{9}$Hz

# CPU Performance

- CPU execution time for a program

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

- Performance improved by
  – Reducing number of clock cycles
  – Increasing clock rate
  – Hardware designer must often trade off clock rate against cycle count

# CPU Time Example

- Computer A: 2GHz clock, 10s CPU time
- Designing Computer B
  - Aim for 6s CPU time
  - Can do faster clock, but causes 1.2 × clock cycles
- How fast must Computer B clock rate be?

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

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

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

# Instruction Count and Cycles Per Instruction

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

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

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

- Instruction Count for a program
  - Determined by program, ISA and compiler
- Average cycles per instruction
  - Determined by CPU hardware
  - If different instructions have different CPI
    - Average CPI affected by instruction mix

# CPI Example

- Computer A: Cycle Time = 250ps, CPI = 2.0
- Computer B: Cycle Time = 500ps, CPI = 1.2
- Same ISA
- Which is faster, and by how much?

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

A is faster…

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

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

…by this much

# Performance in General

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

- Performance depends on
  - Algorithm: affects IC, possibly CPI
  - Programming language: affects IC, CPI
  - Compiler: affects IC, CPI
  - Instruction set architecture: affects IC, CPI, $T_c$
- Time is the only complete and reliable measure of performance