EE116C/CS151B
Fall 2017

Instructor: Professor Lei He
TA: Yuan Liang

Contact Information

- Yuan Liang
- **liangyuandg@g.ucla.edu**
    - Subject starting with [EE116C] or [CS151B]
- **Discussion Session:**
    - Thursday 12pm-1:50pm
    - Friday 2pm-3:50pm
- **Office Hour:**
    - TA does not have office hour for this course
- **Email Response:**
    - Whenever have time
    - Tuesday 1pm-2pm
    - Friday 1pm-2pm

Discussion Format

- 1. Quick review of subjects covered by professor during the week;

- 2. Solution and discussion for last week's homework;

- 3. Sample questions and **discussion**;

Course Syllabus

- 1. ISA and arithmetic: ~3 weeks
    - What is inside a processor; (hardware)
    - What are the most basic arithmetics make a processor; (software on processor)
    - How to convert a **simple** high-level programming code line into instructions on processor; (compile)
- 2. Data Path and Control Design: ~2 weeks
    - How to carry more complex operation (**more** high-level programming codes) on processor.
- 3. Data Pipeline: ~1 week
    - How it works since 1960s.
    - Hazard/branch/other situations.
- 4. Memory (cache) and I/O: ~2 weeks
    - How they work.
- 5. Fast networks and multiprocessors: ~1 week
    - GPU and advanced topics by professor

Why this course

How programs are translated into the machine language.

What determines program performance (trade-offs).

How to improve performance.

Review

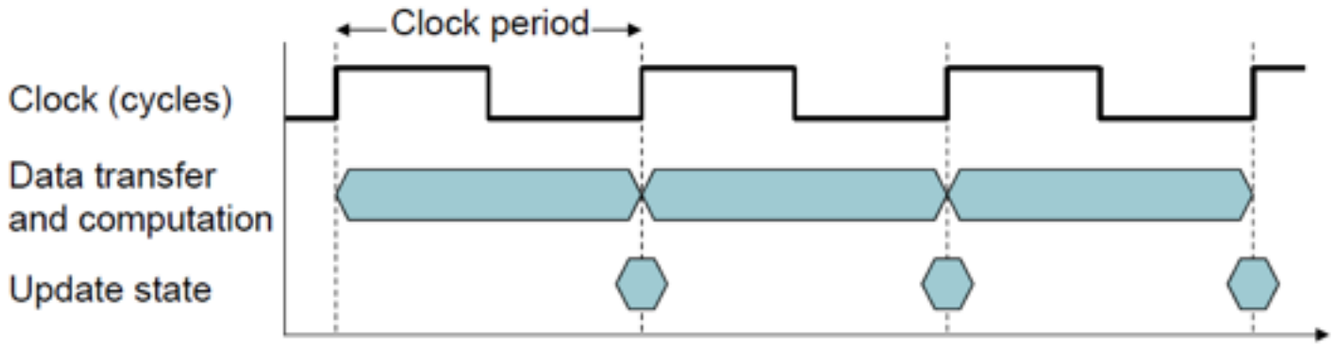
Eight Great Ideas

- Design for ***Moore's Law***
- Use ***abstraction*** to simplify design
- Make the ***common case fast***
- Performance *via* ***parallelism***
- Performance *via* ***pipelining***
- Performance *via* ***prediction***
- ***Hierarchy*** of memories
- ***Dependability*** *via* redundancy

Review

Definition of performance:

Response time
  Relative performance = 1 / execution time
  Elapsed time = total time = Toutput - Tinput
  CPU time = processing time only
    = CPU clock cycles * clock cycle time
    = CPU clock cycle / clock rate (frequency)



Throughput
  tasks/transactions/… per hour

Power

Review

Definition of Instruction set architecture (ISA) from Wiki:

An instruction set architecture (ISA) is a group of commands for a CPU in machine language.

It is an abstract model of a computer. A realization of an ISA is called an implementation.

An ISA permits multiple implementations that may vary in performance, physical size, and monetary cost;

The ISA serves as the interface between software and hardware.

# of clock cycles = Instruction Count * Cycles per Instruction (CPI)
CPU time = Instruction count * CPI * Clock Cycle Time
= Instruction Count * CPI / Clock Rate

Instruction Count for a program determined by :
- program
- ISA
- compiler

Average cycles per instruction determined by:
- CPU

Review

Definition of MIPS:
Million of instructions per second.

Doesn't account for:

Differences in ISAs between computers
One computer with two ISA can have different MIPS.

Differences in complexity between instructions
One computer executes two sets of instructions have different MIPS.

Difference in compilers
More instructions, less time for each instruction, MIPS can be the same.

Review

Power:

Power = Capacitive load * Voltage^2 * Frequency

Frequency and Capacitive load increase.

1. Power increase;
2. Voltage decrease -> electricity leakage

Limit to the performance: the power wall.

Sample Question 1

For one specific program.
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

Sample Question 1

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

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

$$= 10\text{s} \times 2\text{GHz} = 20 \times 10^9$$

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

Sample Question 2

Computer A: Cycle Time = 250ps, CPI =2.0
Computer B: Cycle Time = 500ps, CPI = 1.2
With the same ISA
Which computer faster for this program, and by how much?

Sample Question 2

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

$$= I \times 2.0 \times 250\text{ps} = I \times 500\text{ps}$$

A is faster…

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

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

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

…by this much

Sample Question 3

Alternative compiled code sequences using instructions in classes A, B, C

(one program is compiled to run based on two sets of ISA, each ISA has three **TYPE** of instructions)

Class	A	B	C
CPI for class	1	2	3
IC in sequence 1	2	1	2
IC in sequence 2	4	1	1

What is instruction count (IC), # of clock cycles (CC), cycle per instruction (CPI)?

Sample Question 3

Sequence 1: IC = 5

- Clock Cycles
  = 2×1 + 1×2 + 2×3
  = 10
- Avg. CPI = 10/5 = 2.0

Sequence 2: IC = 6

- Clock Cycles
  = 4×1 + 1×2 + 1×3
  = 9
- Avg. CPI = 9/6 = 1.5