CMP 1203

LECTURE 11

CISC

- Complex Instruction Set Computer
- Motivated by high cost of memory ➔ more complexity in each instruction ➔ smaller program ➔less space in memory
- CISC ISA's have variable length of instructions ➔keep simple instructions short but also allow for longer, more complicated ones
- Also many instructions can directly access memory
- So we have a varying number of clock cycles per instruction.
- Complex instructions need several clock cycles. At times needed to slow down clock to allow instructions to complete ➔ Longer execution time
- Execution of one instruction performs very many operations
- Solution was to return to less complicated architecture : hardwire small, but complete instruction set which would execute quickly
- Hence becomes compilers job to produce efficient code for the ISA

RISC

- “Reduced” Instruction Set Computer
- Provide a machine with minimal number of instructions but can carry out all **essential** operations – data movement, ALU operations and branching
- Only specific instructions allowed to access memory (e.g. load and store)
- Simplifies instructions so they can execute more quickly
- Each instruction performs one operation
- All instructions are same size, few differing layouts
- All arithmetic operations must be performed between registers
- Time for instructions is predictable and constant
- Simple instructions = shorter clock cycles
- Memory became cheaper so storage space less issue
- Most newer machines use RISC or combine CISC and RISC

CISC Vs RISC

- Less instructions = fewer transistors required on chip = cheaper manufacturing costs and more chip space for other uses
- Quantify performance for both machines with computer performance equation:
- Computer performance, measured by program execution time is directly proportional to clock cycle time, number of clock cycles per instruction and number of instructions in the program

$$\frac{\text{time}}{\text{program}} = \frac{\text{time}}{\text{cycle}} \times \frac{\text{cycles}}{\text{instruction}} \times \frac{\text{instructions}}{\text{program}}$$

CISC Vs. RISC

- Shorter clock cycle = improved performance for both RISC and CISC
- RISC minimizes number of cycles per instruction
- CISC reduce the number of instructions per program
- CISC rely on micro code to deal with instruction complexity
- Microcode tells processor how to execute each instruction
- Efficiency of microcode limited by variable length instructions = slower decoding and varying number of clock cycles per instruction = hard to use pipelining

CISC Vs. RISC

- Also microcode executes each instruction as its fetched from memory which requires additional time
- Most RISC instructions need one clock cycle
- RISC replaces micro programmed control by hardwired control which executes instructions faster
- Easier to do pipelining but more difficult to deal with complexity at hardware level
- So complexity removed from instruction set is pushed up to the compiler
- **Homework: VLIW**

CISC Vs RISC

- CISC **mul A,B**
- RISC **add A, B**

  loop
- Multiplication will take more clock cycles than add even if code is shorter
- Also RISC clock cycles are usually shorter than CISC
- So total execution time is less for RISC
- Most CPU architectures have evolved to a combination of the two

Flynn's Taxonomy

- Categorizes computer architectures
- Considers two factors: number of instructions and number of data streams into the processor
- Machine has one or more multiple data streams and one or more processors manipulating the data to give 4 possible options
- SISD (Single Instruction stream, single data stream)
- SIMD (Single instruction stream, multiple data streams)
- MISD (multiple instruction streams, single data streams)
- MIMD (multiple instruction streams, multiple data streams)

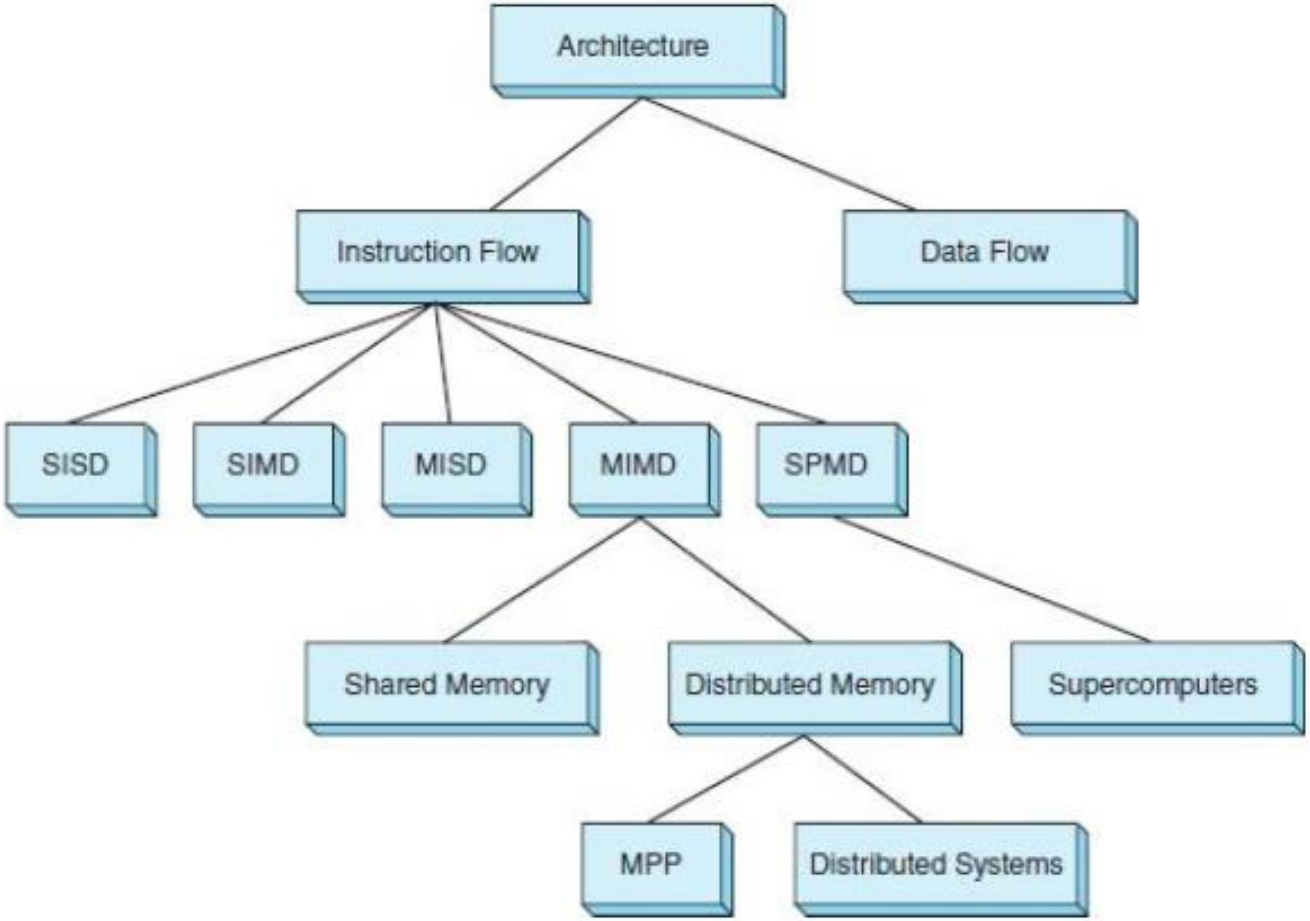
Flynn's Taxonomy

- SISD : uniprocessors
- SIMD: have single point of control; execute same instruction simultaneously on multiple data values
- E.g. array processors, vector processors and systolic arrays
- MISDs have multiple instruction streams operating the same data stream
- MIMDs have multiple control points, independent data and instruction streams e.g. multiprocessors and parallel systems
- SIMD are easier to design than MIMD but less flexible. All SIMD multiprocessors must execute the same instruction simultaneously

Flynn’s Taxonomy

- Very few applications of MISD machines
- Doesn't exhaustively define parallelism or MIMD
- **Homework:**
- Shortcomings of flynn’s taxonomy
- Examples/applications of the 4 classifications above
- SPMD (single program multiple data)

Flynn's Taxonomy



.2 A Taxonomy of Computer Architectures

Parallel and Multiprocessor Architectures

- Goal overtime : make machines solve problems better and faster
- Miniaturization led to improved circuits
- Faster clocks = faster CPUs
- BUT processor performance can only be improved within limit
- Heat and EM interference limit chip transistor density
- Even if we manage to control these limitations, constrained by speed of light
- Also economic limitations: how much will consumers be willing to pay for a processor?
- Eventually no way to improve processor performance than distribute computational load among several processors

Parallelism

- Parallelism improves throughput, better fault tolerance and better price/performance ratio
- Speedup however can never be perfect ➔ cannot necessarily reduce time by 1/n for n processors
- Recall Amdahl's law
- If two processing components run at different speeds, smaller speed will dominate
- There will also be some portion of the work that processor must do serially: other processes have to wait until this stage is complete

Comments on your CAT

- Performance was generally poor
- Read and understand the question. Ask for clarification if you do not understand the question
- Study the text ➔ it is not enough to read the slides
- Understand the material
- Review binary Arithmetic and number representations
- Do NOT confuse number of addresses with number of bits required to represent an address. Refrain from Cramming.
- Do several examples in your text books

About the Exam

- 5 questions, answer 4
- 1 compulsory question
- Be clear and concise
- Answer each question on a new page and write the number you have attempted at beginning of booklet
- Answers to the same question should be on the same pages
- Leave question blank if you are unable to answer it
- Good Luck

Handout

- Null and Lorbur , chapter 9