

# Computer architecture and design

## 1 Digital expression and calculation 04/05

- Digital expression
  - How to express negative number and real number
- Calculation
  - adder and n-bit adder
    - \* use Carry look ahead to prevent the latency in the carry
  - subtraction
    - \*  $X + (-Y)$
  - ALU
- Memorize
  - in order to memorize the result of calculation, we use the flip-flop
  - Register is a set of flip-flop
  - The combination of ALU and Register enables computers calculate

## 2 Branch instruction and Data flow 04/19

- Memory
  - composed of data line and decoder
  - decoder outputs an address and data line receives the data at the address
- Difference between ROM and RAM also SRAM and DRAM
- how to run the instruction sets, program
  - First, an instruction was loaded into the instruction register
  - Then, decode the instruction
  - Calculate and write the result into the register file.
- Sequencer

## 3 Instruction and architecture 04/26 05/10

- assembly language
  - operand and opcode, immediate

- each assembly code corresponds with the machine language
- instruction set
  - Arithmetic logic operation instruction
  - data move instruction
  - branch instruction
- Addressing
  - zero register is used in some architectures
- subroutine
  - function
  - CISC and RISC

## 4 Pipeline 05/24 05/31

- Pipeline
  - The elements are often executed in parallel or in time-sliced fashion, which is called pipeline. In computer it is composed of four elements, "fetch", "decode", "execute", and "write"
  - insert pipeline register between circuits for each elements
- Stall is a condition when execution is stopped by hazard
  - Two ways to realize this. Valid bit and field decode.
- Three kinds of hazard
  - structural hazard
    - \* it occurs when two different instruction uses same hardware sources(data memory etc)
    - \* It can be solved by doubling hardware sources
  - data hazard
    - \* it occurs when process b needs the process a's output
    - \* it can be solved compiler's optimization
  - control hazard
    - \* caused by binary instruction
    - \* predict the binary instruction's result
- Five ways to solve the hazard
  - Forwarding
    - \* Forwarding is to send the result of E step to the next E step without doing W step. (Wstep is a write step, E step is a execute step)
    - \* Branch predction

## 5 Cache 06/07

- Memory hierarchy
  - Combination between fast small memory and Slow large memory

- Transparency
  - Hardware deals with the problems of security and effectiveness
- Write through and write back
  - In the write through systems, CPU stores data in the cache and memory. In the write back systems, CPU stores data only in the cache
- Cache's circuits
  - Address is separated into two parts. One is index and the other is tags
  - Index is used for finding the place in the cache
  - Tags is used for judging the picked data is correct
  - Full associative and set associative
- CPU with cache
  - Construct two caches, instruction cache and data cache, preventing the conflict between instruction cache and load-store
- Cache's performance

## 6 Virtual memory 06/14

- Virtual addresses separate into two pieces, one is page address and the other is offset.
- TLB's movement
- It is important where to place cache and virtual memory
- In order to use virtual memory, the architecture is designed as write back styles

## 7 Presentation 06/21

My team made a presentation for the assembly language, powerPC

## 8 Instruction-level parallelism 06/28 07/05

- parallel processing
  - we need three things to realize this
    1. Hardware resources
    2. The number of ports of registers
    3. Forwarding
    4. The control of the parallel processing
- VLIW (very large instruction word)
  - compiler decides which instructions should be processed in parallel
  - an instruction is composed of hundreds of operations
  - merit and demerit of VLIW
- Superscalar

- Hardware decides which instructions should be processed in parallel, depending on which hazard can be observed
- Hardware would be too complicated

## 9 Other topics (later)

- how display works
- how disk works