OS 1/16

Reagan Shirk

January 16, 2020

Computer Organization

- We should know all of the basic elements of a computer
    - Processor
        * like the brain, controls the operations of the computer
        * performs data processing functions
        * nickname: CPU
    - Main memory
        * Stores data and programs
        * volatile (not permanent, lost after shutdown)
        * nickname: real memory
    - I/O modules
        * monitor
        * keyboard
        * mouse
        * microphone
        * camera
        * storage
        * communication
    - System bus
        * provides communication for all components

CPU

- *I have a picture of a CPU that we drew in class that I'll try to remember to include*
- PC: program counter
    - Keeps track of which instruction you're on; has pointer to next instruction
- IR: instruction register
    - The instruction pointed to by the PC is loaded into the IR
- ALU: Arithmetic logic unit
    - does math- arithmetic based on the instruction
- MAR: memory address register
- MBR: memory buffer register
- I/O address register
- I/O buffer register
- These are called processors mainly because of the ALU thing, each machine has multiple
- Microprocessor
    - General purpose processor on a single chip
- GPU: Graphics Processing Unit
    - What makes a GPU different?
        * Instead of having large system with memory on the outside, they have SIMD
    - SIMD: Single instruction multiple data
        * A grid where each cell in the grid has a piece of data associated with it, you can execute one instruction for each piece of data

* Allows for one instruction to be sent out across multiple pieces of data, similar to parallel programming
* ++MatrixMath is relavent here somehow...?
- DSP: Digital Signal Processing
  * On the rave in 90's telecom business
  * Allows small streaming operations specialized for a chip
  * Seen in digital cameras, motems, any type of speech translator/synthesis, encryption stuff
- 2016 TPU: Tensor Processing Unit
  * Grant's two cents: have some level of competence in machine learning and linear algebra for a successful career
  * tensor: n-dimensional matrix (matrix is a 2 dimensional tensor)
  * TPUs are similar to GPUs but they're optimized for bulk processing operations and matrix multiplication
- SoC: System on a Chip
  * You have *everything* one one chip
    · CPU, GPU, memory, I/O, cache (maybe not all of these but most)

Instructions

- Stages in a program:
  - Fetch: retreive next instruction, contains a start command
  - Execute: processor executes, contains a halt command
- Steps:
  - The processor fetches an instruction from memory, and at that moment:
  - PC counter is incremented
  - Fetched instruction is loaded onto the IR
    * processor-memory
    * processor I/O
    * data processing
    * control
- Quick aside:
  - $10011_2 = 19_{10}$
  - 0 & 1 = 0
  - 0 & 0 = 0
  - 1 & 0 = 0
  - 1 & 1 = 1
  - 0 or 0 = 0
  - 0 or 1 = 1
  - 1 or 0 = 1
  - 1 or 1 = 1
  - 0 xor 0 = 0
  - 0 xor 1 = 1
  - 1 xor 0 = 1
  - 1 xor 1 = 0
  - $2^1 = 2$
  - $2^{10} = 1024 = 1K$
  - $2^{20} = 1048576$
  - $2^{30} = 1$ billion something, G
  - $2^{40} = $ T
  - $2^{12} = 4096 = 2^2 \times 2^{10} = 4K$
  - We should know all powers of 2 up through 2^{10} as well as the ones listed above
- Working Instruction Set Examples
  - What is the format of an instruction in the IR?

* *I have a drawing of this I'll try to remember to add*
- In the integer format, what does the S mean?
  * It is the sign of the integer (+/-)
    · 1 is negative, 0 is positive
  * If you have an IR with [1|0|1|1], the sign is negative and the magnitude is 3
- Least significant bit is always the one all the way to the left (in a picture)
- 1's and 2's complements: what's the difference?
  * -1 in 1's complement: 1110
  * -1 in 2's complement: 1111
  * *Need to make sure we know 1's and 2's complements*
- Remember: we have the program counter (PC), instruction register (IR), and accumulator (AC)
- Operations:
  * 0001 → load from AC
  * 0010 → store AC memory
  * 0101 → add AC from memory
- Memory:
  * 300: 1|9|4|0
  * 301: 5|9|4|1
  * 302: 2|9|4|1
  * 940: 0|0|0|3
  * 941: 0|0|0|2
- How do we run the computer?
  * PC = 300, we need a fetch operation
    · We take the value of memory slot 300 and load it into the IR
    · IR = 1940
  * PC = 301, look at instruction register to find instruction from OpCode, in this case we need to lead from accumulator
    · AC = 0003
    · 0003 was taken from memory slot 940 but I don't think I understand why, I took comp org a year ago and I was bad at it then too
    · Fetch 301 from memory after the execute
    · IR = 5941
  * PC = 302, look at instruction register to find instruction from OpCode, we need to add AC from memory
    · Last three digits of IR are 941, so we go to that spot in memory and add it to what is in the accumulator
    · AC = 0003, 941 = 0002, together = 0005
    · AC = 0005
  * PC = 303
    · IR = 2941, store AC in memory
    · Store 0005 in 941