Homework 4

Katerina Chinnappan

kachinna@ucsc.edu

1) (10 pts) Name the five components of the von Neumann model. For each component, state its purpose
Answer: 1) Memory: capable 2^ (28) distinct memory locations, each capable of storing 8 bits of information. 2) Processing unit: actual info in the computer is carried out by the processing unit. Can perform many complicated functional units. 3) Input: basic devices that can take a user's input into the computer. Keyboard, mouse. 4) Output: display the user's input results. Monitor, printer, LED display. 5) Control unit: unit in charge to make all the other parts play together. Keeps track of instructions.

2) (8 pts) Briefly describe the interface between the memory and the processing unit. That is, describe the method by which the memory and the processing unit communicate.
Answer: Communication between memory and processing unit – 2 registers.
   1) MAR (memory address register)
   2) MDR (memory data register)
      - Read – address of location is in MAR. Then the value is put in MDR.
      - Write – address of location is in MAR. Then the value is put in MDR.

3) (4 pts) What is misleading about the name "program counter"? Why is the name "instruction pointer "more insightful?
Answer: Because program counter is like counting something whereas an instruction pointer is more accurate because a program counter is a process register that indicated where a computer is in the sequence of instructions.

4) (8 pts) What is the word length of a computer? How does the word length of a computer affect what the computer can compute?
Answer: Word length refers to the number of bits processed by the CPU. Word size can be 32 bits or 64 bits.

5) (8 pts) What are two components of an instruction? What information do these two components contain?
Answer: Opcode – the code for the instruction (ADD), what it does.
Operand – what the instruction must do with the operand, specifies what the data is to be manipulated operated on.

6) (8 pts) Suppose a 32-bit instruction takes the following format:

OPCODE SR DR IMM

If there are 60 opcodes and 32 registers, what is the range of values that can be represented by (IMM)? Assume IMM is a 2's complement value.

Answer: 60 opcodes: represented in binary we calculate ceiling[log(base2) (60)] = **6 bits**.

Total is 32 registers so to calculate SR and DR= ceiling[log(base2)(32)] = **5 bits.**

Now remaining IMM we calculate: 32-6-5-5 = **16 bits**.

A two's complement 16 bit binary number can be in the range from -2^(16-1) to 2^(16-1) -1

So: -32768 to 32767

7) (12 pts) State the phases of the instruction cycle and briefly describe what operations occur in each phase.

Answer: Instructions to order:

Fetch – fetch instruction from memory, load in PC.
Decode – decode, understand what the instructions are
Evaluate Address – calculate address
Fetch Operands – get operands from registers
Execute – run the instructions
Store address – store the result in the register specified

8) (4 pts) A memory's addressability is 64 bits. What does that tell you about the size of the MAR and MDR?

Answer: If a memory's addressability is 64 bits, then the location is 64 bits long. Basically, MDR is the same in size as the memory locations so MDR is 64 bits. MAR - the size of MAR is the number of locations, so I can't say the size MAR.

9) (12 pts) Say we have a memory consisting of 256 locations, and each location contains 16 bits.

a. How many bits are required in for the address?

b. If we use the PC-relative address mode, and want to allow control transfer between instructions 20 locations away, how many bits of a branch instruction are needed to specify the PC-relative offset?

c. If a control instruction is in location 3, what is the PC-relative offset of address 10. Assume that the control transfer instructions work the same way as in the LC-3

Answer: a) log(base2)(256) = **8 bits**

b) ceiling[(log(base2)(20+1)+1)] = 5 + 1 = **6**

c) Since (b) is 6m then 10 – x = 6, x = 4, **so PC is incremented to 4**.

10) (4 pts) What is the largest positive number we can represent literally (i.e. as an immediate value) within an LC-3 ADD instruction?

Answer: previously, we found out that IMM5 is 5 bits, so 2^(5-1) – 1 = 2^(4) – 1 = 16 -1 = **15.**