

CMPE 12 Homework #2

John Allard
Lab Section #2

October 29th, 2014

1. Refer to the table supplied in the textbook for the next few questions.
 - (a) What binary value does location 3 contain? What about location 6?
Answer : Location 3 contains the binary value 0000 0000 0000 0000, location 6 contains 000 0110 1101 1001.
 - (b) Binary values can be interpreted in different ways, interpret the values in the given table as instructed so by the problems below.
 - i. Interpret location 0 and location 1 in two's complement.
Answer : Location 0 contains 7,747, location 1 contains -4,059.
 - ii. Interpret location 4 as an ASCII value.
Answer : 'e'.
 - iii. Interpret locations 6 and 7 and an IEEE floating point number. (7 being the high bits, 6 being the low).
Answer : 0 00001101 10110011111111011010011 = 8.20007582580913043462315872541E-35
 - iv. Interpret locations 1 and 0 and unsigned integers. **Answer :** Location 0 contains 7,474, and location 1 contains 61,477.
 - (c) Interpret the data in location 0 as an instruction.
Answer : The opcode 0001 corresponds to the ADD operation. This operation has a 0 in the 5th bit place, which means it is operating on two source registers. The operation is $R7 = R1 + R3$.
 - (d) Interpret the data in location 5 as an address, then state the data that is at that address.
Answer : Location 5 contains the address 6. Location 6 contains the value 0xfed3.
2. Suppose a 32-bit opcode takes the format of

OPCODE	SR	DR	IMM
--------	----	----	-----

If there are 60 opcodes and 32 registers, what is the range of values that can be represented by the intermediate (IMM)?

Answer : To represent 60 opcodes, you would need 6 bits (this would give you 4 extra). To address 32 registers, you would need exactly 5 bits. Thus to state an opcode and 2 registers, you would need to use 16 bits. This would leave 16 bits for the IMM. $2^{16} = 65,536$. Thus any address within 65,536 of the current position can be accessed.
3. Using the same 32-bit instruction format as above, except there needs to be 225 opcodes and a 120 registers to address.
 - (a) Minimum number of bits to represent an opcode? **Answer :** 8 bits.
 - (b) Min. number of bits to represent the Destination register? **Answer :** 7 bits.
 - (c) What is the maximum number of unused bits in the instruction encoding? **Answer :** 3 unused bits.
4. Describe the execution of the JMP instruction if R3 contains x369C.
Answer : This instruction will be stored at x36A2. The IR is loading with the JMP instruction, and the PC is incremented to x36A3. Then, when the jump command is processed, the PC is made to point to x369C, when the RET command is found, it will jump back to x36A3 and continue execution.
5. If we have a memory consisting of 256 locations, and each location contains 16 bits.
 - (a) How many bits are required for the address? **Answer :** 8 bits
 - (b) If we use the PC-relative addressing 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? **Answer :** 5 bits, would allow us to jump up to 32 spaces away.

- (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.
6. Fill in the missing two sequences below to perform an OR operation. (Lines 1,3 are given, 2 and 4 need to be filled by the student). The operation is being performed on registers 1 and 2 and will be stored in register 3.
- (a) 1001 100 001 1111111 - NOT R1 into R4.
 (b) 1001 000 010 1111111 - NOT R2 into R5.
 (c) 0101 110 100 000 101 - AND R4 and R5 into R6
 (d) 1001 011 110 1111111 - NOT R6 into R3

The OR is accomplished using DeMorgans laws, which states that $\neg(\neg A \wedge \neg B) = A \vee B$

7. State the contents of the R1, R2, R3, and R4 registers after the program starting at x3100 halts.
8. An LC-3 program is located in memory locations x3000 to x3006. It starts executing at the former, if we keep track of all values loaded into the MAR as the program executed, we will get a sequence that starts as follows (See book). Your job is to fill in each blank space with either a 0 or 1 as appropriate.

x3000	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0
x3001	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1
x3002	1	0	1	1	0	0	0	0	0	0	0	0	0	0	1	1
x3003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
x3004	1	1	1	1	0	0	0	0	0	0	1	0	0	1	0	1
x3005	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
x3006	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1

It starts at x3000, gets a LD instruction from x3005 into R0, then it adds 1 to this value, then it stores this value.

9. The LC3 has no divide instruction. A programmer needing to divide two numbers would have to write a routine to handle it. Show the systematic decomp. of the process of dividing two positive integers. Write an LC-3 machine language program starting at location x3000 which divides the number in memory location x4000 by the number in memory location x4001 and stores the quotient at x5000 and the remainder at x5001.
10. For the program listed on page 191 of the book (Chapter 7, problem 5), what does the program do and what is the value contained in the RESULT (x0000) address after the program halts?
11. The program fragment shown on page 192 of the book has an error in it. Identify the error and state how to fix it.
12. Write a program that checks the initial value in memory location x4000 to see if it is a valid ASCII code. If it is, print the char. Else, exit the program.
13. What does the LC-3 program on page 217 (Chapter 8, problem 16) do?
- 14.
- 15.
- 16.