NAME:

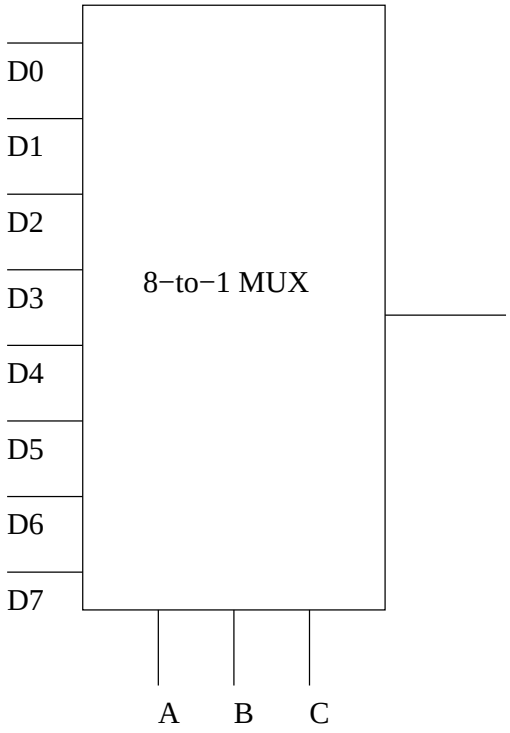
USERNAME:

**CS 2011**
**Exam 1**
D-Term 2008

| | | |
|---|---|---|
| Question 1: | _____ | (10) |
| Question 2: | _____ | (20) |
| Question 3: | _____ | (15) |
| Question 4: | _____ | (20) |
| Question 5: | _____ | (20) |
| Question 6: | _____ | (15) |
| TOTAL: | _____ | (100) |

Fill in your name and username. DO NOT OPEN THIS TEST UNTIL YOU ARE TOLD TO DO SO.

1. (10 points) In class, we saw how a function such as the majority function could be implemented by hard-wiring a multiplexer. Hard-wire the mutiplexer pictured below so that it computes the following Boolean function: $P(A, B, C) = \overline{A}B + C$




2. (20 points) An 8-bit byte contains the hexadecimal value xF2. What decimal number does the byte represent if the value in the byte is

   (a) an unsigned integer?

   (b) a signed, two's-complement integer?

3. (15 points) Here's a 32-bit quantity expressed in hexadecimal: x41F20000. The number is in IEEE single-precision floating point. What decimal value does it represent? (Show all work for partial credit.)

4. Given the following Karnaugh map:

| AB \ CD | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | | 1 | 1 | 1 |
| 01 | | 1 | 1 | |
| 11 | | 1 | | |
| 10 | | | 1 | |

(a) (5 points) Write the sum-of-products equation from which this Karnaugh map is derived. (Don't provide a minimal function yet, just give the function that would lead to the Karnaugh map pictured above.)

(b) (5 points) Put rectangles around the 1's in the given Karnaugh map to make a minimal mapping.

(c) (5 points) Use your rectangles to write a minimal sum-of-products equation for the function P.

(d) (5 points) Draw the minimal circuit that corresponds to your equation from part (c).

5. (20 points)

(a) Write machine code instructions (machine code, not assembly language!) that will implement the following pseudo-code:

```
DO
    R2 = R2 + R1
    R3 = R3 - 1
WHILE R3 != 0
```

(b) An LDR instruction, located at location x3100, uses R6 as its base register. The value currently in R6 is x5000.

i. What is the largest address that the LDR instruction can load from?

ii. Suppose the offset in an LDR instruction is zero-extended, rather than sign-extended. Then what would be the largest address the LDR instruction could load from?

6. (15 points) At some instant in time, the state of the LC-3 is partially described by these register values and memory location contents (all given in hex):

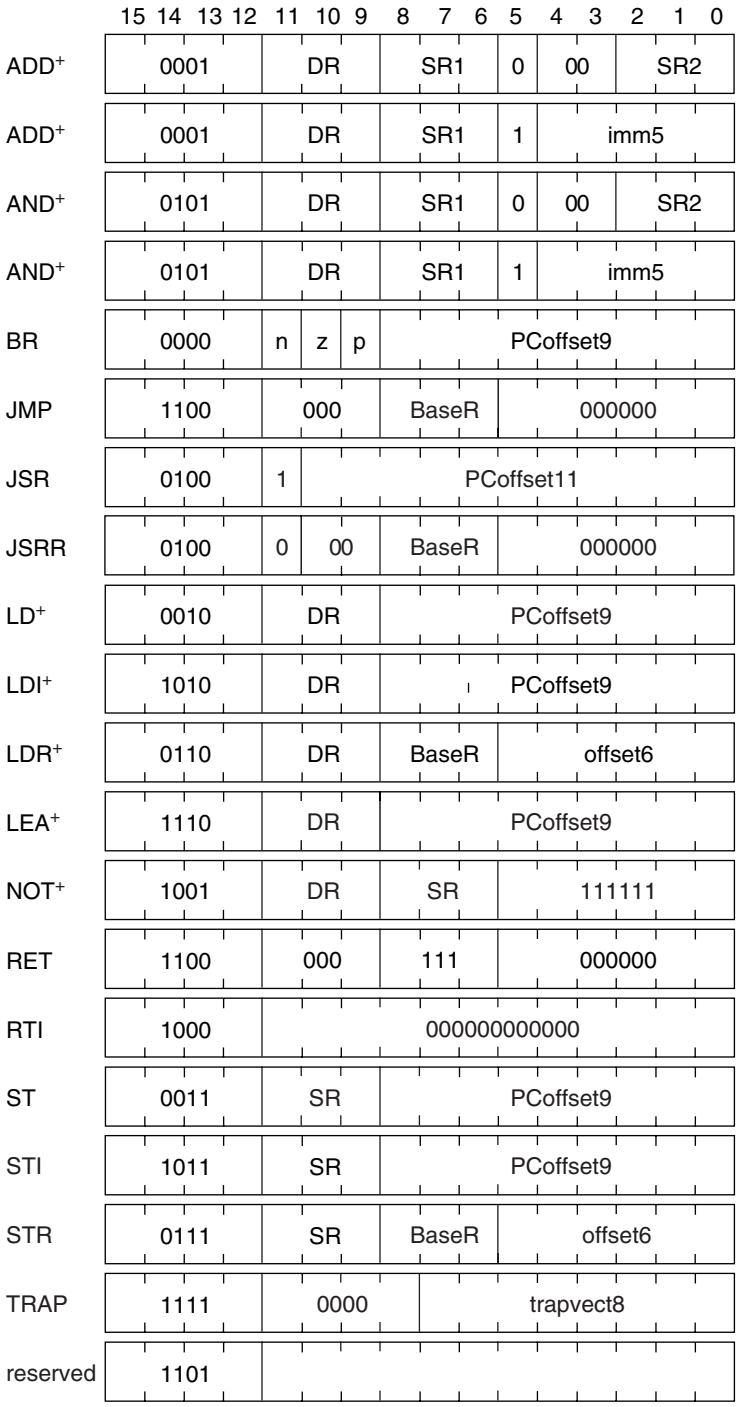
```
PC:  x3100                      Memory
IR:  x0000                 address  contents
                           -------  --------
                           x30FE:   x3103
R0:  x0000                 x30FF:   x3104
R1:  x1111                 x3100:   xB9FE
R2:  x2222                 x3101:   xC0C0
R3:  x3333                 x3102:   xF025
R4:  x4444                 x3103:   xABCD
R5:  x5555                 x3104:   x30FE
R6:  x6666                 x3105:   x2345
R7:  x7777

MAR: x4000
MDR: x5000
```

Assume the machine in this state is just about to fetch an instruction.

Give the series of micro-operations that show how the data above is transferred throughout the machine as a **single instruction** is fetched, decoded, and executed. Provide a comment for each step. For example, the first step is:

MAR $\leftarrow x3100$ `;; MAR gets the current PC value`


15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ADD+ 0001 DR SR1 0 00 SR2
ADD+ 0001 DR SR1 1 imm5
AND+ 0101 DR SR1 0 00 SR2
AND+ 0101 DR SR1 1 imm5
BR 0000 n z p PCoffset9
JMP 1100 000 BaseR 000000
JSR 0100 1 PCoffset11
JSRR 0100 0 00 BaseR 000000
LD+ 0010 DR PCoffset9
LDI+ 1010 DR PCoffset9
LDR+ 0110 DR BaseR offset6
LEA+ 1110 DR PCoffset9
NOT+ 1001 DR SR 111111
RET 1100 000 111 000000
RTI 1000 000000000000
ST 0011 SR PCoffset9
STI 1011 SR PCoffset9
STR 0111 SR BaseR offset6
TRAP 1111 0000 trapvect8
reserved 1101