READING: Chapter 5 (complete)

HOMEWORK SET (6 questions, 12 points): Complete the following exercises.

Questions 1-3:

I suggest that you start by printing copies of the ***LC-3 data path schematic***

Download LC-3 data path schematic

(a simplified version of this schematic is also available on pages 126 & 174 of the text, and under Resources -> LC-3 ISA Guides).
Use the schematic to trace the sequence of control signals, showing the resulting "flow" of data for each instruction. Do NOT submit these traces with your homework - they are just to help you visualize the instruction.

To prepare for the final exam, you will need to do this same analysis for all 14 instructions we study (all but RTI), so you might as well go ahead & print 14 copies of the schematic now :)
See:

Links to an external site.

LC-3 instruction datapaths & control signals.pdf

Download LC-3 instruction datapaths & control signals.pdf

==========

1. Given the data path of the LC-3 as per the above-linked schematic, give a complete description of the instruction:

x31A1: LDI R1, label ; **The instruction is stored at address x31A1. *label* corresponds to the address x3246**

a) *(1 point)* Assemble the instruction to ML (Machine Language)

The machine language instruction would be 1010 0001 0010 0100. The LDI's opcode is 1010, R1 is register 001, and the label's offset is 0x0246, or 0010 0100 0110 in binary.

b) *(1 point)* **Give the RT (Register Transfer) description of the instruction.**

R1 <- sign_extend(offset9)

PC <- PC + 1

c) *(1 point)* ***List, in the correct sequence, every control signal set by the FSM to implement this instruction.***

MAR <- R6 + offset6

PCMUX <- 00

MDR <- R0

DRMUX <- 0

SR1MUX <- 1

ALUK <- 00

BEN <- N/A

R6MUX <- 1

PC

==========

2. Do the same for the instruction *(1 + 1 + 1 points)*:

STR R0, R6, x0

*Note: the Store instructions all have to **FIRST** copy the target address into the MAR (STI requires 2 "passes" through memory to do this!), and **THEN** copy the contents of the source register to the MDR, before they can trigger Mem.En/W.*
The second step, MDR <= (SR), requires that (see schematic

Download schematic

)

a) the bitfield DR (IR[11:9]) be applied to the the SR1 decoder via the SR1.MUX selector; and

0111 000 110 000000

0111 for STR opcode

000 for binary of decimal 0 of R0

110 for binary of decimal 6 of R6

000000 for the offset

b) the contents of the SR1 bus being passed through to the ALU output via ALUK selecting "Pass through input A".
Make sure you understand this sequence fully!

Transfers the value stored in R0 to the memory address stored in R6

==========

3. Do the same for the instruction: *(1 + 1 + 1 points)*

x35D0: BRzp label ; where *label* corresponds to address x34EF
Notes:
a) the BRzp instruction assembles to memory address x35D0;
b) the labeled address comes before the instruction!
The assembler will calculate the 9-bit offset in the usual manner:
offset = target address - (PC) = -((PC) - target address)
where (PC) is of course the incremented PC.

==========

Questions 4-6: Chapter 5, exercises **5.40, 5.41, 5.42** - *pay special attention to 5.40 (and should seem familiar) & 5.41!*

(Study the BR instruction, both in the text 5.4.1 and in the datapath document

Download datapath document

for these two questions!)

5.40 The following logic diagram shows part of the control structure of the LC-3 machine.
What is the purpose of the signal labeled A?
The signal A determines if it will branch or not.

5.41 A part of the implementation of the LC-3 architecture is shown in the following diagram.
a. What information does Y provide?

Y is the P condition code.
b. The signal X is the control signal that gates the gated D latch. Is there an error in the logic that produces X?
Signal X produces 1.

5.42 The LC-3 macho-company has decided to use opcode 1101 to implement a new instruction. They need your help to pick the most useful one from the following:
a. MOVE Ri, Rj; The contents of Rj are copied into Ri.
b. NAND Ri, Rj, Rk; Ri is the bit-wise NAND of Rj, Rk
c. SHFL Ri, Rj, #2; The contents of Rj are shifted left 2 bits and stored into Ri.
d. MUL Ri, Rj, Rk; Ri is the product of 2's complement integers in Rj, Rk.
Justify your answer.
I picked c because the instruction has the most uses. This instruction can perform operations like addition, subtraction, and multiplication by shifting bits which is very versatile compared to the other instructions.