

Homework 5 Fall 2011 Solution

Problem 1 (4 Points):

PC – 32 bits

PC contains the address of the NEXT instruction to fetch.

MAR contains the address in memory that is to be read or written

These two registers work by having the contents of PC placed into the MAR so the address in memory containing an instruction can be loaded

IR – 64 bits

IR – contains the instruction that is currently being executed

MDR contains data that is to be read/written from/to memory

During a fetch cycle the contents of memory is placed into the MDR and then placed into the IR so it can be executed

Problem 2 (6 points):

1)

x4400 AND R3, R3, #0

x4401 ADD R3, R3, R1

x4402 ADD R2, R2, #-1

x4403 BRp 0x4401 <-3>

2)

Assume the contents of R1 are 2 and R2 is 2, the student can enter any value:

R1 = 2

R2 = 2

Start:

X4400: R1=2, R2=2, R3=0, CC=Z

X4401: R1=2, R2=2, R3=2, CC=P

X4402: R1=2, R2=1, R3=2, CC=P

X4401: R1=2, R2=1, R3=4, CC=P

X4402: R1=2, R2=0, R3=4, CC=Z

END

For a solution there will be R2 loops, or branches.

3)

This algorithm performs a multiply operation: $R3 \leftarrow R1 \times R2$

Problem 3 (5 points):

First convert the binary to assembly:

```

0x30FF      LEA R1, 0x1
0x3100      LDR R2, R1, 0x2
0x3101      TRAP 0x25          ;HALT, look at page 543
0x3102      ADD R2, R1, R1
0x3103      ADD R2, R2, R2

```

```

0x30FF      R1 ← PC + 0x1 = 0x3100 + 0x1 = 0x3101
0x3100      R2 ← mem[ R1 + 0x2 ] = mem[3103] – R2 ← 0001 0100 1000 0010
0x3101      Program halts, so R2 contain the data at 0x3103

```

R2 contains 0001 0100 1000 0010, or 0x1482, or ADD R2, R2, R2

Problem 4 (5 points):

First, write the assembly of this:

```

0x3100      LEA R1, 32(0x20)
0x3101      LD R2, 32(0x20)
0x3102      LDI R3, 32(0x20)
0x3103      LDR R4, R2, 1(0x1)
0x3104      TRAP 0x25          ;HALT, look at page 543
...
0x3122      0x4566
0x3123      0x4567
...
0x4567      0xABCD
0x4568      0xFED3

```

Look at contents:

```

0x3100      R1 ← 0x3101 + 0x20 = 0x3121
0x3101      R2 ← mem[PC + 0x20] = mem[0x3102 + 0x20] = mem[0x3122] = 0x4566
0x3102      R3 ← mem[mem[PC + 0x20]] = mem[mem[3123]] = mem[0x4567] = 0xABCD
0x3103      R4 ← mem[R2 + 0x1] = mem[0x4566 + 0x1] = mem[0x4567] = 0xABCD

```

So:

```

R1 ← 0x3121
R2 ← 0x4566
R3 ← 0xABCD
R4 ← 0xABCD

```

Problem 5 (4 points):

Students are free to pick which registers they want.

- a) Assume a move from R2 to R1:
AND R1, R2, 0xFFFF
or ADD R1, R2, 0x0
- b) Clear R1:
AND R1, R1, 0x0
- c) Shift contents of R1 to the left by one bit, same as multiplying by 2:
ADD R1, R1, R1
- d) (AND R1, R1, R1) or (AND R1, R1, 0xFFFF) or (ADD R1, R1, 0x0)
There could be other answers as well.

Problem 6 (6 points):

The screenshot shows the Penn Sim interface. The title bar is "Penn Sim - 1.2.5 \$Rev: 436 \$ - LC3 ISA". The menu bar has "File" and "About". The "Controls" section has buttons for "Next", "Step", "Continue", and "Stop", and a status indicator "Suspended". Below the controls is a large empty text area. The "Registers" section shows a table of registers R0 through R5, with R2 selected. The "Memory" section shows a table of memory addresses from 0x2FEE to 0x300A, with the instruction ".FILL x0000" for each. A red box highlights the memory addresses from 0x3000 to 0x300A, and the instruction for 0x3000 is ".FILL x524A".

BP	Address	Value	Instruction
<input type="checkbox"/>	x2FEE	x0000	.FILL x0000
<input type="checkbox"/>	x2FEF	x0000	.FILL x0000
<input type="checkbox"/>	x2FF0	x0000	.FILL x0000
<input type="checkbox"/>	x2FF1	x0000	.FILL x0000
<input type="checkbox"/>	x2FF2	x0000	.FILL x0000
<input type="checkbox"/>	x2FF3	x0000	.FILL x0000
<input type="checkbox"/>	x2FF4	x0000	.FILL x0000
<input type="checkbox"/>	x2FF5	x0000	.FILL x0000
<input type="checkbox"/>	x2FF6	x0000	.FILL x0000
<input type="checkbox"/>	x2FF7	x0000	.FILL x0000
<input type="checkbox"/>	x2FF8	x0000	.FILL x0000
<input type="checkbox"/>	x2FF9	x0000	.FILL x0000
<input type="checkbox"/>	x2FFA	x0000	.FILL x0000
<input type="checkbox"/>	x2FFB	x0000	.FILL x0000
<input type="checkbox"/>	x2FFC	x0000	.FILL x0000
<input type="checkbox"/>	x2FFD	x0000	.FILL x0000
<input type="checkbox"/>	x2FFE	x0000	.FILL x0000
<input type="checkbox"/>	x2FFF	x0000	.FILL x0000
<input type="checkbox"/>	x3000	x524A	.FILL x524A
<input type="checkbox"/>	x3001	x0000	.FILL x0000
<input type="checkbox"/>	x3002	x0000	.FILL x0000
<input type="checkbox"/>	x3003	x0000	.FILL x0000
<input type="checkbox"/>	x3004	x0000	.FILL x0000
<input type="checkbox"/>	x3005	x0000	.FILL x0000
<input type="checkbox"/>	x3006	x0000	.FILL x0000
<input type="checkbox"/>	x3007	x0000	.FILL x0000
<input type="checkbox"/>	x3008	x0000	.FILL x0000
<input type="checkbox"/>	x3009	x0000	.FILL x0000
<input type="checkbox"/>	x300A	x0000	.FILL x0000

Students should have the HEX value of their initials entered starting at address 0x3000.