Compulsory, Capacity, Conflict

Final Exam – Part Two –Written Part

Write your answers to the written part in clear, concise writing. You need to write with a #2 pencil or pen. If your writing is too light the scanner will not pick it up and you will not get credit for your answer. Feel free to ask for scratch paper.

Name: Andrew Janiszka

1. (5 pts) Translate the following into MIPS assembly. Assume that a, b, and c are in $s0, $s1, and $s2 respectively:

```
if (a < b)
  c = c + b;
else
  c = c + a;
```

```
bge $s0, $s1, else
Add $s2, $s2, $s1
j cont
else:
  Add $s2, $s2, $s0

cont:
```

2. (4 pts) Define "Von Neumann architecture" – remember, a definition should let me tell one thing from all other things and should not include unnecessary facts, so be specific but be concise.

A single shared memory path for data and programs.

3. (8 pts) Translate the following code into MIPS assembly:

```
int find_item(int *A, int N, int item)
{
    int i;
    for (i = 0; i < N; i++) {
      if (A[i] == item) return i;
    }
   return -1;
}
```

```
find_item:
li $t0, 0
li $v0, -1
loop:
bge $t0, $a2, end
lw $t1, 0($a1)
bne $t1, $a3, Cont
add $v0, $zero, $t0
j end
Cont:
addi $t0, $t0, 1
sll $a1, $a1, 2
j loop
end:
jr $ra
```

4. (6 pts) Show the encoding in 2's complement 16-bit binary, octal, and hex:
   a. 3423

      16-bit Binary: 0000_1101_0101_1111

      16-bit (actually, 18-bit) Octal: 000_000_000_000_006_537

      16-bit Hex: 0x0000 0000 0000 0D5F

   b. -54

      16-bit Binary: 0000_0000_0000_1100

      16-bit (actually, 18-bit) Octal: 000_000_000_000_000_014

      16-bit Hex: 0000_0000_0000_000C

5. (6 pts) Consider two processors, P1, and P2, executing the same instruction set (so same number of instructions). P1 has a 3GHz clock, and CPI of 1.5. P2 has a 2.6GHz clock and a CPI of 1.25.

   c. Which processor has the highest instructions per second?

   cycles / s

   Cycles / inst

   P1 ⇒ 3e9 / 1.5 ⇒ 2e9 instructions / second

   P2 = 2.6e9 / 1.25 ⇒ 2.08e9 instructions/second

   P2

   d. Processor P3 is a proposed "enhancement" of P1. In a given program, 20% of its instructions can use the *enhanced* instructions. Compute the CPI that would yield a speed-up of 13% for P3 over P1.

   3e9 / 2350400000 = 1.276 CPI

6. (4 pts) Show the IEEE 754 Single-Precision encoding of -11.25

   1

   0.25 × 2 = 50 = 0

   50 × 2 = 1 = 1

   1011

   10.1110

   1 10000 100 10111000000000000000000

7. (6 pts) Define "power wall" and explain why its so important to the future of computing.

   - We have reached the maximum power and efficiency per transistors
   - These are needed for computers
   - Future computers need to run faster, but can't because of the limitations.

Consider the following datapath diagram of the MIPS processor, and use it to answer the following questions:

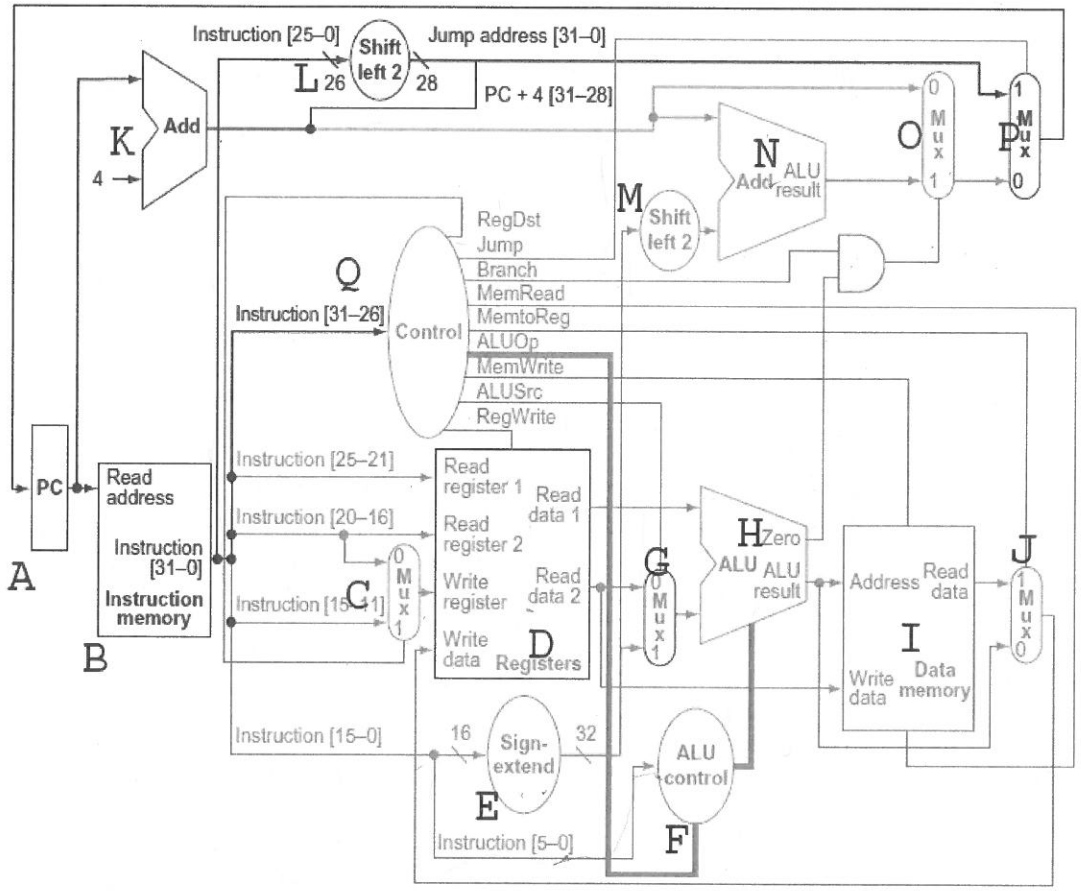

Figure 4.24 from Hennessy and Patterson [1]

Use the figure to answer questions 8 through 13.

8. When handling the load word (LW) instruction, the ALU control will select the Write Reg function.

9. When handling an I-type instruction which functional units will be used:
Sign extended, Read Reg 2, Read Reg 1, Control

10. The devices labeled C,G,J,O,P are all "muxes", what do they do?
they act as binary switches relying on input from other units to work

11. When would functional unit N be used?
when shifting left 2 from E or by the pc using it.

[1] Figure taken from Hennesy & Patterson, pg 263.

12. (3 pts) What instructions would cause the "RegDst" control line to be turned on?

instructions 11-15

13. (4 pts) Fill in the table for the control signals for an R-type instruction, using 1 or 0 based on the muxes of the figure on the previous page:

RegDst	Jump	Branch	MemRead
0	1	1	1
MemtoReg	**MemWrite**	**AluSrc**	**RegWrite**
1	1	1	0

14. (3 pts) Explain why a single-cycle implementation CPU is no longer used:

Because it relies on memory to do the work

15. (3 pts) Define pipelining

A diagram showing instructions working their way through the cpu

16. (8 points) A major concept of chapter 4 is that there are four types of hazards possible due to pipelining. List them, and then describe one way of "fixing" the problem:

Hazard Type	"Fix"
Structural	Forwarding
data	Seprate instruction and data memory