McGill University
ECSE 425
COMPUTER ORGANIZATION AND ARCHITECTURE
Winter 2009
Midterm Examination

February 17, 2009 4:05 PM – 5:35 PM

Duration: 1.5 hours

- **There are 4 questions for a total of 100 points. There are 12 pages. Please check that you have all 12 pages.**
- **This is an opened-book exam. Use of calculators and all documentation are permitted.**
- **Write your name and student number in the space below. Do the same on the top of each sheet of this exam.**
- **State any assumptions.**
- **Write your answers in the space provided.**

Name: _____

Student Number: _____

Q1: _____ **Q3:** _____

Q2: _____ **Q4:** _____

Total:

Name: **ID:**

Question 1. Short Answers (20 points).
There are 10 sub questions (2 points each).

1. List four implementation technologies critical to modern computer design.

2. What means the term “the learning curve”.

3. What are two methods to measure the overall system speedup?

4. What is called *addressing mode*?

5. Why alignment restrictions are used in a computer design?

6. What are the addressing modes for control flow instructions?

7. How many general-purpose registers must provide a new instruction set architecture and why?

8. What are the classes of hazards?

9. How many stages are there in the R4000?

10. What are the characteristics of integer arithmetic overflow exception?

Name: **ID:**

Question 2. (20 points):
There are two sub questions 10 points each

Three enhancements with the following speedups are proposed for a new architecture: S1=35, S2=30, S3=20. Only one enhancement is usable at a time.

1. What is the overall speedup gained by incorporating only the enhancement 1 if this enhancement is used 20% of time?

2. Amdahl's Law can be generalized to handle multiple enhancements in the following way:

   $S_{overall} = \left[1 - \sum_i FE_i + \sum_i \frac{FE_i}{S_i} \right]^{-1}$ where FE_i is the fraction of time that enhancement *i* can be used and S_i is the speedup of enhancement *i*.
   If enhancements 1 and 2 are each usable for 35% and 25% of the time correspondently, what fraction of the time must enhancement 3 be used to achieve an overall speedup of 10?

Name: **ID:**

Question 3 (20 points).
There are two parts, (a) and (b) 10 points each

a) Suppose that we use the classic RISC five-stage integer pipeline with a single-cycle delayed branch. Consider the following code fragments and schedule the branch delay slot using the most appropriate strategy, explain your choice.

1.

```
          LD        R5, 20(R6)
          DADDI     R5, R5,#14
          SD        20(R6),R5
Label:    LD        R1, 0(R2)
          DADDI     R1, R1,#1
          SD        0(R2),R1
          DADDI     R2,R2,#4
          DSUB      R4,R3,R2
          BNEZ      R4, Label
```

2.

```
Label:    LD        R1, 0(R2)
          DADDI     R1, R1,#1
          SD        0(R2),R1
          DADDI     R2,R2,#4
          DSUB      R4,R3,R2
          BNEZ      R4, Label
```

b) Assume the RISC pipeline having 7 stages (IF1, IF2, ID, EX, M1,M2, WB).The branch is resolved at the end of the third cycle for unconditional branches and at the end of the forth cycle for conditional branches. Suppose that 20% of all instructions are conditional branches (60% are taken) and 5% are unconditional branches or procedure calls.

1. How many stalls we have for

a. Conditional branch?

b. Unconditional branch?

2. What is the CPI for this computer if we don't use prediction, nor delay slot completion.

Name: **ID:**

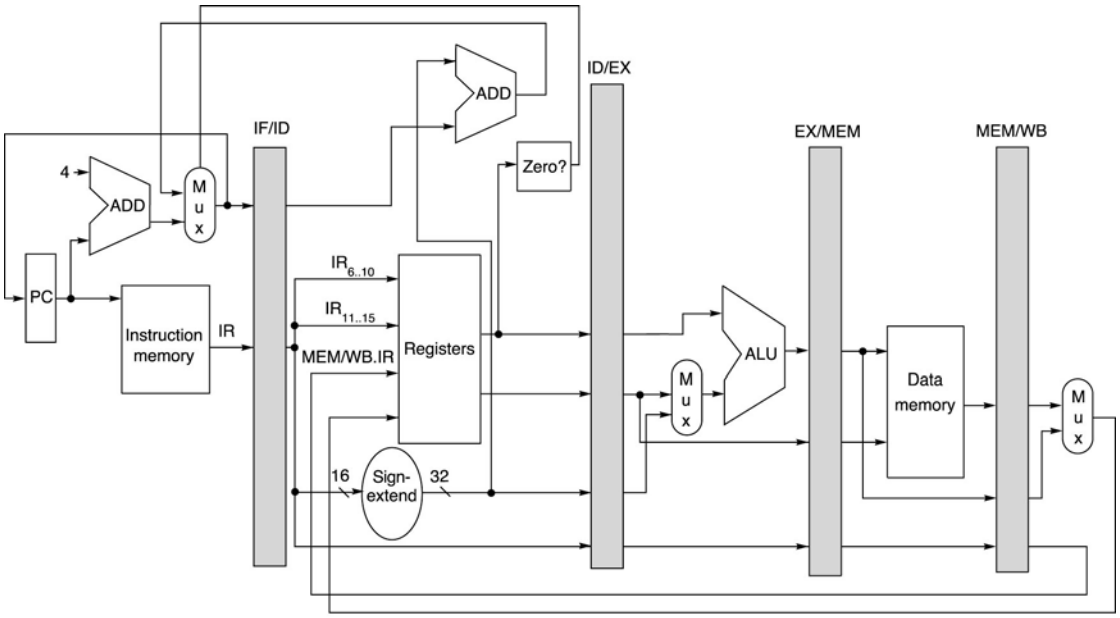
Question 4. Pipelining (40 points):

There are three parts, (a) and (b), and (c) to this question.

(a) **(15 points)** Consider the following code fragment:

```
Label:    LD       R1, 0(R2)
          DADDI    R1, R1, #1
          SD       R1, 0(R2)
          DADDI    R2, R2, #4
          DSUB     R4, R3, R2
          BNEZ     R4, Label
```

The code will run on the classic 5-stage (I, D, E, M, W) RISC pipeline shown in the figure below and assume all memory accesses take 1 clock cycle.



Fill in the chart on the next page to show the timing of this instruction sequence **without** any forwarding or bypassing hardware but assuming a register read and a write in the same clock cycle. Assume that the branch is handled by flushing the pipeline.

Name: **ID:**

```
Label:    LD        R1, 0(R2)
          DADDI     R1, R1, #1
          SD        R1, 0(R2)
          DADDI     R2, R2, #4
          DSUB      R4, R3, R2
          BNEZ      R4, Label
```

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
LD	**I**	**D**	**E**	**M**	**W**														

Name: **ID:**

(b) **(15 points)** Consider the same code fragment:

```
Label:    LD      R1, 0(R2)
          DADDI   R1, R1, #1
          SD      R1, 0(R2)
          DADDI   R2, R2, #4
          DSUB    R4, R3, R2
          BNEZ    R4, Label
```

Assume now that the pipeline has full hazard detection and forwarding hardware (the forwarding paths are not shown). Assume a register read and a write in the same clock cycle "forwards" through the register file. Assume all memory accesses take 1 clock cycle.
Assume the initial value of R3 is R2+36. Show the timing of this code fragment for the RISC pipeline. **Assume the branch is handled by a "predict-not-taken" strategy**. Fill in the chart to show the timing. Add arrows to indicate any forwarding. The first line is filled in for you. If all memory references take 1 cycle, how many cycles does this loop take to execute?
.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
LD	**I**	**D**	**E**	**M**	**W**														

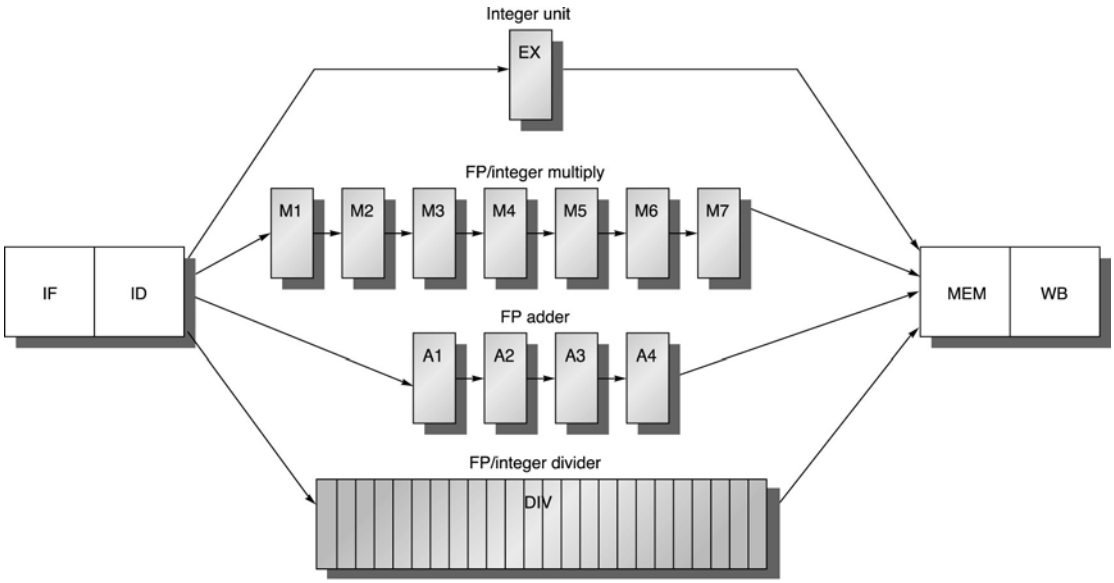
Name: **ID:**

(c) **(10 points)** Consider the following code fragment:

```
L.D       F0, 0(R2)
L.D       F4, 0(R3)
MUL.D     F0, F0, F4
SD        F0,0(R3)
ADD.D     F2, F0, F2
DADDUI    R2, R2, #8
DADDUI    R3, R3, #8
DSUBU     R5, R4, R2
```

Assume the MIPS floating point pipeline in the figure below. The latencies and initiation intervals are given in the table below.

	Integer unit (ALU)	FP add	FP and integer multiply	FP and integer divide
Latency	0	3	6	24
Initiation interval	1	1	1	25



Show the timing of this code assuming full forwarding hardware and assuming a register read and write in the same clock cycle “forwards” through the register file. Assume all memory references complete in one clock cycle. Fill in the chart on the next page to show the timing. When write-back contention occurs, the earliest instructions get priority.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30