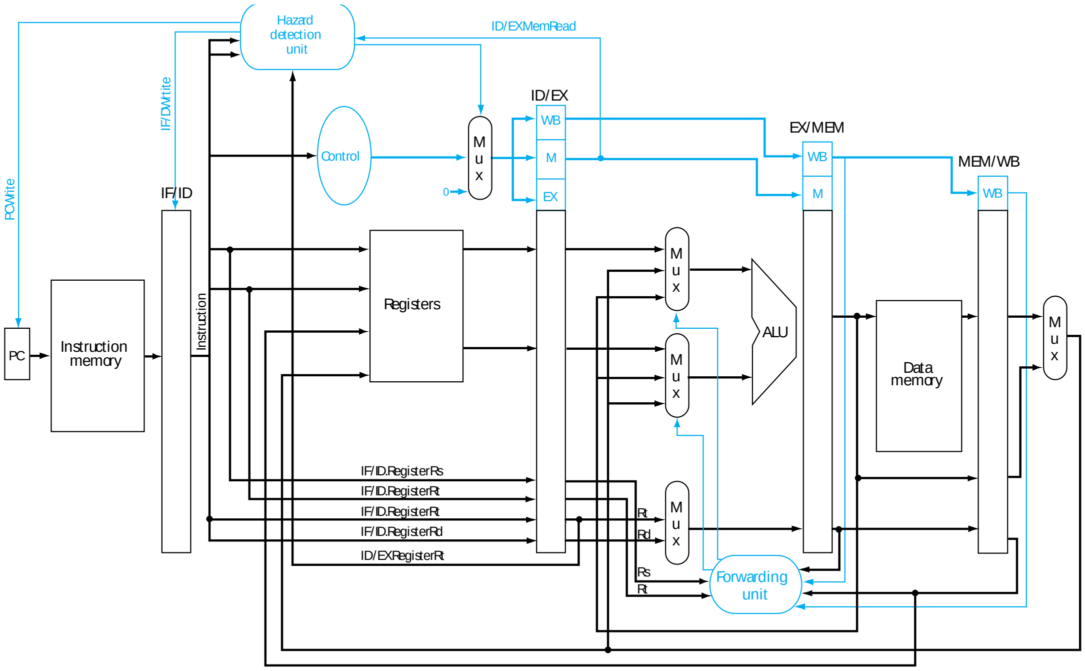
Computer Organization and Structure

Homework #4
Due: 2013/12/10

1. Consider executing the following code on the pipelined datapath shown as the following figure:

```
add  $2, $3, $1
sub  $4, $3, $5
add  $5, $3, $7
add  $7, $6, $1
add  $8, $2, $6
```



a. At the end of the fifth cycle of execution, which registers are being read and which register will be written?

b. Explain what the forwarding unit is doing during the fifth cycle of execution. If any comparisons are being made, mention them.

c. Explain what the hazard detection unit is doing during the fifth cycle of execution. If any comparisons are being made, mention them.

2. Assuming the following latencies for logic blocks in the datapath:

	I-Mem	Add	Mux	ALU	Regs	D-Mem	Sign-extend	Shift-left-2
a.	400ps	100ps	30ps	120ps	200ps	350ps	20ps	0ps
b.	500ps	150ps	100ps	180ps	220ps	1000ps	90ps	20ps

a. What is the clock cycle time if the only type of instructions we need to support are

ALU instructions (`add`, `and`, etc.)?

b. What is the clock cycle type if we only had to support `lw` instructions?

c. What is the clock cycle time if we must support `add`, `beq`, `lw`, and `sw` instructions?

For the remaining problems, assume that there are no pipeline stalls and that the breakdown of executed instructions is as follows:

	add	addi	not	beq	lw	sw
a.	30%	15%	5%	20%	20%	10%
b.	25%	5%	5%	15%	35%	15%

d. In what fraction of all cycles is the data memory used?

e. In what fraction of all cycles is the input of the sign-extend circuit needed? What is this circuit doing in cycles in which its input is not needed?

f. If we can improve the latency of one of the given datapath components by 10%, which component should it be? What is the speed-up from this improvement?

3. Consider a 5-stage (`IF`, `ID`, `EX`, `MEM`, `WB`) MIPS pipeline processor with hazard detection unit. Suppose the processor has instruction memory for `IF` stage, and data memory for `MEM` stage so that the structural hazard for memory references can be avoided.

a. Assume *no forwarding unit* is employed for the pipeline. We are given a code sequence shown below.

```
LD    R1, 10(R2);        #R1 ← MEM[R2 + 10]
SUB   R4, R1, R6;        #R4 ← R1 - R6
ADD   R5, R1, R6;        #R5 ← R1 + R6
```

Show the *timing of each instruction* of the code sequence. Your answer may be in the following form.

Instruction		Clock Cycle									
		1	2	3	4	5	6	7	8	9	10
LD	R1, 10(R2)	IF	ID	EX	MEM	WB					
SUB	R4, R1, R6										
ADD	R5, R1, R6										

b. Consider another code sequence shown below.

```
SUB   R1, R3, R8;        #R1 ← R3 - R8
SUB   R4, R1, R6;        #R4 ← R1 - R6
ADD   R5, R1, R6;        #R5 ← R1 + R6
```

Suppose both hazard detector and forwarding unit are employed. Show the timing of each instruction of the code sequence as the above table.

4. The following code contains a “read after write” data hazard that is resolved by forwarding:

```
add   $2, $3, $4
add   $5, $2, $6
```

Consider the similar situation in which a memory read occurs after a memory write:

```
sw    $7, 100($2)
lw    $8, 100($2)
```

Write a paragraph describing how this situation differs from the one involving registers, and describe how the potential “read after write” problem is resolved.

5. Suppose we have a floating-point (FP) unit that requires 400ps for a FP add and 600ps for a FP multiply, not including the time to get the instruction or read and write any registers, which take the same as for an integer instruction. Assume that the functional unit times are the following:

- Memory units: 200ps
- ALU and adders: 100ps
- Register file (read or write): 50ps

Assume the following:

- All loads take the same time and comprise 30% of the instructions.
- All stores take the same time and comprise 15% of the instructions.
- R-format instructions comprise 25% of the mix.
- Branches comprise 10% of the instructions, while jumps comprise 5%.
- FP add and subtract take the same time and together total 5% of the instructions.
- FP multiply and divide take the same time and together total 10% of the instructions.

Find

a. The time for the FP operations
b. The time for the processor with a single clock cycle length equal to the longest instruction