0907432 Computer Design (Summer 2013)

Quiz 1

الاسم: الرقم التسلسلي: رقم الشعبة: 1

==

Instructions: Time **30** minutes. Open book and notes exam. No electronics. Please answer all problems in the space provided and limit your answer to the space provided. **No questions are allowed**.

<Good Luck>

Q1. Assume that you have a typical 5-stage pipelined processor that uses forwarding and stalls to solve data hazards. Assume that the processor is executing the following code segment and that the first instruction has reached the write back stage. Also assume that when the first instruction was fetched, the contents of the registers were: R0 has 0, R1 has 4, R2 has 8, R3 has 12, R4 has 16, etc.

[5 marks]

```
sub      R1, R2, R3   F  D  E  M  W
lw       R2, 8(R3)       F  D  E  M  W
and      R4, R1, R2         F  D  D  E  M  W
or       R1, R2, R3            F  F  D  E  M  W
```

(a) In the following table specify what instruction has reached each of the four listed stages.

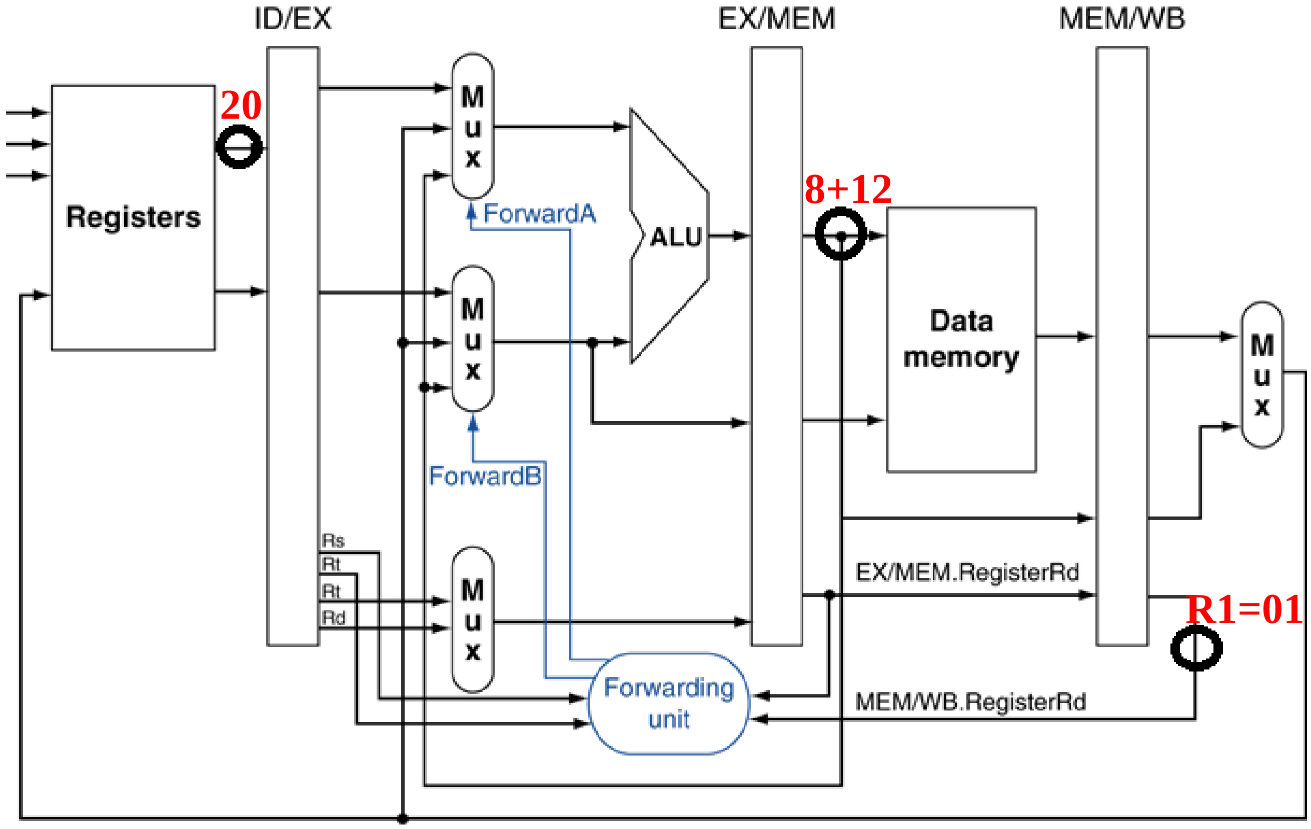
[half for each instruction]

Decode	Execute	Memory	Write back
and	bubble	lw	sub

0.5 mark each

(b) The figure below shows the last four stages of this pipeline. Specify the value of the circled busses.

[3 marks]



1 mark each

Q2. Unroll the following loop twice and schedule it for the two-way static scheduling processor studied in the class.

[5 marks]

```
Loop: lw        r5, 0(r1)
      lw        r4, 0(r2)
      sub       r5, r5, r4
      add       r6, r6, r5
      addi      r2, r2, #4
      addi      r1, r1, #4
      bne       r1, r3, Loop
```

```
Loop: lw      r5, 0(r1)
      lw      r4, 0(r2)
      sub     r5, r5, r4
      add     r6, r6, r5
      lw      r5, 4(r1)
      lw      r4, 4(r2)
      sub     r5, r5, r4
      add     r6, r6, r5
      addi    r2, r2, #8
      addi    r1, r1, #8
      bne     r1, r3, Loop
```

Renaming registers:

```
Loop: lw      r5, 0(r1)
      lw      r4, 0(r2)
      sub     r5, r5, r4
      add     r6, r6, r5
      lw      r7, 0(r1)
      lw      r8, 0(r2)
      sub     r7, r7, r8
      add     r6, r6, r7
      addi    r2, r2, #8
      addi    r1, r1, #8
      bne     r1, r3, Loop
```

Cycle	ALU/branch	Load/store
1	`nop`	`lw r5,0(r1)`
2	`addi r1,r1,#8`	`lw r4,0(r2)`
3	`addi r2,r2,#8`	`lw r7,-4(r1)`
4	`sub r5,r5,r4`	`lw r8,-4(r2)`
5	`add r6,r6,r5`	`nop`
6	`sub r7,r7,r8`	`nop`
7	`add r6,r6,r7`	`nop`
8	`bne r1,r3,loop`	`nop`
9		
10		
11		

- *1 Mark for correct replication and removed of loop overhead.*
- *1 Mark for modifying constants.*
- *1 Mark for correct distribution of instructions in two slots.*
- *1 mark for correct schedule.*
- *1 Mark for 8-cycle schedule.*