

Homework 6

Due: Nothing to turnin, for Quiz review only.

Turnin: Nothing.

1. — Using a drawing similar to Figure 6.29, page 408 (or Slide 22 of 6 — Pipelining), show the forwarding paths needed to execute each of the following sets of instructions:

a.)

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

b.)

```
add $4, $1, $7
lw  $6, 16($4)
slt $8, $1, $5
sub $5, $6, $6
```

2. — Identify all of the data dependencies in the following code. Which dependencies are data hazards that will be resolved via forwarding?

```
add $2, $5, $4
add $4, $2, $5
sw  $5, 100($2)
add $3, $2, $4
```

3. — For each pipeline register in Figure 6.22, page 400 (also Slide #14 of 6 — Pipelining), label each portion of the pipeline register with the name of the value that is loaded into the register. Include the number of bits needed for each value, and the total number of bits in each register. Note: this figure does not include the control bits that would also be in the registers.

To get you started: The IF/ID pipeline register contains two fields, one of which is an instruction field that is 32 bits wide. The second field is...

4. — For each pipeline register in Figure 6.32, page 411 (also Slide #27 of 6 — Pipelining), state the number of bits that are added to each pipeline register to handle data forwarding. State the purpose of each of the bits added. For register(s) that do not add bits to support forwarding, state that no bits are added to those registers.

5. — Consider executing the following code on the pipelined datapath of Figure 6.36, page 416 (also Slide 29 of 6 — Pipelining):

```
add $1, $2, $3
add $4, $5, $6
add $7, $8, $9
add $10, $11, $12
add $13, $14, $15
```

a. During the fifth cycle of execution, which general-purpose registers are being read and which general-purpose register will be written?

b.) During the fifth cycle of execution, what else is happening?

6. —

a.) Consider executing the following code on the pipelined datapath of Figure 6.36, page 416 (also Slide 23 of 6 — Pipelining):

```

add  $7, $6, $7
lw   $6, 100($7)
sub  $7, $6, $8
add  $6, $6, $7

```

How many cycles will it take to execute this code? Draw a diagram like that of Figure 6.29, page 408 (or Slide 22 of 6 — Pipelining) that illustrate the dependencies that need to be resolved, and provide another diagram like that of Figure 6.35, page 415 (Slide 24 of 6 — Pipelining) that illustrates how the code will actually be executed (incorporating any stalls or forwarding) so as to resolve the identified problems.

b.) Same question as a., but use the following code:

```

add  $2, $5, $4
add  $4, $2, $5
lw   $4, 100($4)
add  $3, $2, $4

```

c.) Same question as a., but use the following code:

```

add  $3, $5, $7
sw   $7, -32($3)
lw   $4, 16($3)
sub  $1, $17, $4
lw   $5, 16($1)
slt  $2, $5, $4

```

d.) Same question as a., but use the following code:

```

lw   $2, 0($7)
add  $3, $1, $4
sub  $4, $3, $2
lw   $10, 0($4)
slt  $5, $3, $10
sw   $5, 0($10)

```