JOINT INSTITUTE
交大密西根学院

ECE3700J Introduction to Computer Organization

Homework 4

Assigned: October 20, 2022

Due: 2:00pm on November 1, 2022

Submit a PDF file on Canvas

1. (15 points) Given this instruction:
   `sw x5, -4(x2)`

   As the instruction goes through the pipeline, what will be stored in the pipeline registers:
   IF: what's in PC
   Answer:
   - **address of the sw instruction (1 point)**

   ID: what's in IF/ID
   Answer:
   - **address of the sw instruction,**
   - **machine code of sw instruction: 1111111_00101_00010_010_11100_0100011 (0xFE51_2E23) (1 point)**

   EX: what's in ID/EX?
   Answer:
   - **address of the sw instruction**
   - **content of x2, (1 point)**
   - **content of x5, (1 point)**
   - **32-bit immediate number: 0xFFFFFFFC (1 point)**
   - **Instruction[30, 14-12]: 0b1010, (1 point)**
   - **Rd: 28 (1 point)**
   - **ALUSrc: 1 (1 point)**
   - **ALUOp: 00**
   - **Branch: 0**
   - **MemWrite: 1 (1 point)**
   - **MemRead: 0 (1 point)**
   - **MemtoReg: x (1 point)**
   - **RegWrite: 0 (1 point)**

   MEM: what's in EX/MEM
   Answer:
   - **Address of sw + (-8) (1 point)**
   - **Content of x2 – 4 (1 point)**
   - **Content of x5**
   - **Rd: 28**
   - **Branch: 0**
   - **MemWrite: 1**
   - **MemRead: 0**

- **MemtoReg: x**
- **RegWrite: 0**
- **Zero: unknown**

WB: what's in MEM/WB?
Answer:
- **Mem read data output: unknown (1 point)**
- **Content of x2 – 4**
- **Rd: 28**
- **MemtoReg: x**
- **RegWrite: 0**

2. (20 points) Assume that individual stages of the RISC-V pipelined datapath have the following latencies:

IF	ID	EX	MEM	WB
300 ps	150 ps	250 ps	300 ps	150 ps

Also, assume that instructions executed by the processor are broken down as follows:

ALU/Logic	Jump/Branch	Load	Store
45%	15%	20%	20%

(1) What is the clock cycle time? (2 points)
Answer: at least 300 ps

(2) What is the execution time of a sw instruction in the pipelined processor? (3 points)
Answer: clock cycle time * number of stages = 300 x 5 ps = 1,500 ps

(3) If we can split one stage of the pipelined datapath into two new stages, each with half the latency of the original stage, which stage would you split and what is the new clock cycle time of the processor? (5 points)
Answer: IF or MEM stage (3 points), 250ps (2 points)

(4) Using the processor to run a program of 1,000 instructions, what is the total execution time? What is the average CPI? (10 points)
Answer:
Total execution time = (1000+4) * 300 = 301,200 ps (5 points)
(using the answer from (3) is also OK)

Average CPI = (1000+4)/1000 = 1.004 (5 points)

3. (10 points) Assume that x11 is initialized to 11 and x12 is initialized to 12. Suppose you executed the code below on a pipelined processor that does not handle data hazards at all.

```
L1: addi x11, x12, 5
L2: add x13, x12, x11
L3: addi x14, x11, 15
```

(1) Indicate data dependencies, if any, in above instruction sequence. (which register between which instructions) (5 points)

Answer:
x11 between L1 and L2 (2.5 points)
x11 between L1 and L3 (2.5 points)

(2) What would the final values of registers x13 and x14 be? (5 points)

Answer:
x13 will be 11+12 = 25 (2.5 points)
x14 will be 11+15 = 16 (2.5 points)

4. (30 points) Given the following instructions:

```
L1: addi x8,x0,1000
L2: sw   x18,-12(x8)
L3: lw   x3,8(x18)
L4: add  x6,x3,x3
L5: or   x8,x9,x6
```

a) Assume there is no forwarding in this pipelined processor. Indicate hazards and add NOP instructions to eliminate them. How many clock cycles will it take to execute the instructions? (10 points)

Answer:
Hazards:
- **EX hazard on x8 between L1 and L2 (1 point)**
- **load-use hazard between L3 and L4 on x3 (1 points)**
- **EX hazard between L4 and L5 on x6 (1 point)**

Adding NOP:

```
L1: addi x8,x0,1000
NOP (2 points)
NOP
L2: sw  x18,-12(x8)
L3: lw  x3,8(x18)
NOP (2 points)
NOP
L4: add x6,x3,x3
NOP (2 points)
NOP
L5: or  x8,x9,x6
```

It will take 15 clock cycles to complete the instructions. (1 points)

b) Assume there is ALU-ALU forwarding. Indicate hazards and add NOP instructions to eliminate them. How many clock cycles will it take to execute the instructions? (10 points)

Answer:

Same hazard as above. If there is ALU-ALU forwarding, EX hazards and some MEM hazards involving ALU output can be eliminated, but not the load-use hazard. (3 points)

Adding NOP: (5 points)

```
L1: addi x8,x0,1000
L2: sw   x18,-12(x8)
L3: lw   x3,8(x18)
NOP
NOP
L4: add x6,x3,x3
L5: or  x8,x9,x6
```

It will take 11 clock cycles to complete the instructions. (2 points)

c) Assume there is full forwarding. Indicate hazards and add NOP instructions to eliminate them. How many clock cycles will it take to execute the instructions? (10 points)

If there is full forwarding, then EX hazard is removed. For load-use hazard, data coming out of memory can be forwarded to ALU (MEM-ALU forwarding). But load-use hazard cannot be completely eliminated. (3 points)

One NOP has to be added. (5 points)

```
L1: addi x8,x0,1000
L2: sw   x18,-12(x8)
L3: lw   x3,8(x18)
NOP
L4: add x6,x3,x3
L5: or  x8,x9,x6
```

It will take 10 clock cycles to complete. (2 points)

5. (25 points) Given this assembly instruction sequence executed by the pipelined processor:

```
L1: sub x6, x2, x1
L2: lw  x3, 8(x6)
L3: lw  x2, 0(x6)
L4: or  x3, x5, x3
L5: sw  x3, 0(x5)
```

Hazards:
- **EX hazard (ALU-ALU): L1 and L2 on x6**
- **MEM hazard (ALU-ALU): L1 and L3 on x6**
- **MEM hazard (MEM-ALU): L2 and L4 on x3**
- **New hazard (ALU-MEM): L4 and L5 on x3**

a) If the processor has forwarding, but we forgot to implement the hazard detection unit, what happens when this code executes? (5 points)

Answer:

The hazard detection unit is used to stall an instruction to insert a bubble to handle load-use hazard. There is no load-use hazard, thus no need to stall the pipeline, thus hazard detection unit has no effect to this instruction sequence.

b) If there is forwarding, for the first five cycles during the execution of this code, specify which signals are asserted in each cycle by hazard detection and forwarding units. (10 points)

Signals generated by hazard detection unit: PCWrite, IF/IDWrite, Hazard
Signals generated by forwarding unit: ForwardA, ForwardB

CC1: PCWrite=1, IF/IDWrite=1, Hazard=0; ForwardA=00, ForwardB=00
CC2: PCWrite=1, IF/IDWrite=1, Hazard=0; ForwardA=00, ForwardB=00
CC3: PCWrite=1, IF/IDWrite=1, Hazard=0; ForwardA=00, ForwardB=00
CC4: PCWrite=1, IF/IDWrite=1, Hazard=0; ForwardA=10, ForwardB=00
CC5: PCWrite=1, IF/IDWrite=1, Hazard=0; ForwardA=01, ForwardB=00

(2 points each)

c) If there is no forwarding, what new inputs and output signals do we need for the hazard detection unit? Using this instruction sequence as an example, explain why each signal is needed. (10 points)

Answer:
If there is no forwarding, data hazards can only be resolved by stalling an instruction

```
L1: sub x6, x2, x1
NOP
NOP
L2: lw  x3, 8(x6)
L3: lw  x2, 0(x6)
NOP
L4: or  x3, x5, x3
NOP
NOP
L5: sw  x3, 0(x5)
```

So in the hazard detection unit, we also need to detect:
- **ID/EX.RegisterRd == IF/ID.RegisterRs1**
- **ID/EX.RegisterRd == IF/ID.RegisterRs2**
- **ID/EX.RegWrite == 1**
- **EX/MEM.RegisterRd == IF/ID.RegisterRs1**
- **EX/MEM.RegisterRd == IF/ID.RegisterRs2**
- **EX/MEM.RegWrite == 1**

EX/MEM.RegisterRd, ID/EX.RegWrite, EX/MEM.RegWrite are the new inputs. (2 points each signal)

No new output signals are needed. (2 points)

Explanation: using above code or similar. (2 points)