


1. Assume 73 and 88 are signed 8-bit decimal integers stored in sign-magnitude format. Calculate 73 + 88. Is there overflow, underflow, or neither? (5 分)

2. 13

2. Assume 73 and 88 are signed 8-bit decimal integers stored in sign-magnitude format. Calculate 73 - 88. Is there overflow, underflow, or neither? (5 分)

3. Calculate the product of the octal unsigned 6-bit integers 62 and 12 using the hardware described in the following Figure. You should show the contents of each register on each step. (10 分)

| Step | Action | Multiplier | Multiplicand | Product |
|------|-----------------|------------|-----------------|-----------------|
| 0 | Initial Rals | 001 010 | 000 000 110 010 | 000 000 000 000 |
| | Isb=0, no op | 001 010 | 000 000 110 010 | 000 000 000 000 |
| | Lshift Mcond | 001 010 | 000 001 100 100 | 000 000 000 000 |
| | Rshift Mplier | 000 101 | 000 001 100 100 | 000 000 000 000 |
| 1 | Prod=Prod+Mcond | 000 101 | 000 001 100 100 | 000 001 100 100 |
| | Lshift Mcond | 000 101 | 000 011 001 000 | 000 001 100 100 |
| | Rshift Mplier | 000 010 | 000 011 001 000 | 000 001 100 100 |
| 2 | Isb=0, no op | 000 010 | 000 011 001 000 | 000 001 100 100 |
| | Lshift Mcond | 000 010 | 000 110 010 000 | 000 001 100 100 |
| | Rshift Mplier | 000 001 | 000 110 010 000 | 000 001 100 100 |
| 3 | Prod=Prod+Mcond | 000 001 | 000 110 010 000 | 000 111 110 100 |
| | Lshift Mcond | 000 001 | 001 100 100 000 | 000 111 110 100 |
| | Rshift Mplier | 000 000 | 001 100 100 000 | 000 111 110 100 |
| 4 | Isb=0, no op | 000 000 | 001 100 100 000 | 000 111 110 100 |
| | Lshift Mcond | 000 000 | 011 001 000 000 | 000 111 110 100 |
| | Rshift Mplier | 000 000 | 011 001 000 000 | 000 111 110 100 |
| 5 | Isb=0, no op | 000 000 | 011 001 000 000 | 000 111 110 100 |
| | Lshift Mcond | 000 000 | 110 010 000 000 | 000 111 110 100 |
| | Rshift Mplier | 000 000 | 110 010 000 000 | 000 111 110 100 |
| 6 | Isb=0, no op | 000 000 | 110 010 000 000 | 000 111 110 100 |
| | Lshift Mcond | 000 000 | 110 010 000 000 | 000 111 110 100 |
| | Rshift Mplier | 000 000 | 110 010 000 000 | 000 111 110 100 |

1. $73_{10} = 01001001_2$
 $88_{10} = 01011000_2$

$1001001 = 73$

$1011000 = 88$

$73 + 88 = 161$ (overflow)

最多 111111 只能表示到 127

2.

$73 - 88 = -15$ (neither)

3.

| | | |
|---|----------|--------|
| 1 | Rshift M | 000101 |
| 2 | Rshift M | 000010 |
| 3 | Rshift M | 000001 |
| 4 | Rshift M | 000000 |

000111 110 100

4. We examine how pipelining affects the clock cycle time of the processor. Problems in this exercise assume that individual stages of the datapath have the following latencies:

| | IF | ID | EX | MEM | WB |
|-------------|-------|-------|-------|-------|-------|
| Processor 1 | 300ps | 400ps | 350ps | 500ps | 100ps |
| Processor 2 | 200ps | 150ps | 120ps | 190ps | 140ps |

- 4.1 What is the clock cycle time in a pipelined and non-pipelined processor? (10 p)

500ps

- 4.2 What is the total latency of an LW instruction in a pipelined and non-pipelined processor? (10 p)

MEM 400ps

5. In this exercise, we examine how resource hazards, control hazards, and Instruction Set Architecture (ISA) design can affect pipelined execution. Problems in this exercise refer to the following fragment of MIPS code:

```
sw r16, 12(r6)
lw r16, 8(r6)
beq r5, r4, Label # Assume r5 != r4
add r5, r1, r4
slt r5, r15, r4
```

Assume that individual pipeline stages have the following latencies:

| IF | ID | EX | MEM | WB |
|-------|-------|-------|-------|-------|
| 200ps | 120ps | 150ps | 190ps | 100ps |

- 5.1 For this problem, assume that all branches are perfectly predicted (this eliminates all control hazards) and that no delay slots are used. If we only have one memory (for both instructions and data), there is a structural hazard every time we need to fetch an instruction in the same cycle in which another instruction accesses data. To guarantee forward progress, this hazard must always be resolved in favor of the instruction that accesses data. What is the total execution time of this instruction sequence in the 5-stage pipeline that only has one memory? (5 p)

| Instruction | Pipeline Stage | Cycles |
|-------------------|----------------|--------|
| sw r16, 12(r6) | IF | 100ps |
| lw r16, 8(r6) | | |
| beq r5, r4, Label | | |
| add r5, r1, r4 | | |
| slt r5, r15, r4 | | |

2

4.1 clock cycle
non-pipelined

$$\text{Processor 1} : 300 + 400 + 350 + 500 + 100 = 1650 \text{ ps}$$

$$\text{Processor 2} : 200 + 150 + 120 + 190 + 140 = 800 \text{ ps}$$

pipelined

$$\text{Processor 1} : 500 \text{ ps}$$

$$\text{Processor 2} : 200 \text{ ps}$$

time latency

non-pipelined

$$\text{Processor 1} : 1650 \text{ ps}$$

$$\text{Processor 2} : 800 \text{ ps}$$

pipelined

$$\text{Processor 1} : 500 \times 5 = 2500 \text{ ps}$$

$$\text{Processor 2} : 200 \times 5 = 1000 \text{ ps}$$

4.2

4.1 What is the clock cycle time in a pipelined and non-pipelined processor? (10 分)

500ps

4.2 What is the total latency of an LW instruction in a pipelined and non-pipelined processor? (10 分)

MEM 400ps

5. In this exercise, we examine how resource hazards, control hazards, and Instruction Set Architecture (ISA) design can affect pipelined execution. Problems in this exercise refer to the following fragment of MIPS code:

```
sw r16,12(r6)
lw r16,8(r6)
beq r5,r4,Label # Assume r5!=r4
add r5,r1,r4
slt r5,r15,r4
```

Assume that individual pipeline stages have the following latencies:

| IF | ID | EX | MEM | WB |
|-------|-------|-------|-------|-------|
| 200ps | 120ps | 150ps | 190ps | 100ps |

5.1 For this problem, assume that all branches are perfectly predicted (this eliminates all control hazards) and that no delay slots are used. If we only have one memory (for both instructions and data), there is a structural hazard every time we need to fetch an instruction in the same cycle in which another instruction accesses data. To guarantee forward progress, this hazard must always be resolved in favor of the instruction that accesses data. What is the total execution time of this instruction sequence in the 5-stage pipeline that only has one memory? (5 分)

| Instruction | Pipeline Stage | Cycles |
|---|----------------|--------|
| sw r16,12(r6) lw r16,8(r6) beq r5,r4,Label add r5,r1,r4 slt r5,r15,r4 | IF | 100ps |

2

5.1

IF ID EX Mem WB

IF ID EX Mem WB

IF ID EX Mem WB

xx xx IF ID EX Mem WB

IF ID EX Mem WB

11 cycles

11 x 200 = 2200 p

5.2 For this problem, assume that all branches perfectly predicted (this eliminates all control hazards) and that no delay slots are used. If we change load/store instructions to use a register (without an offset) as the address, these instructions no longer need to use the ALU. As a result, MEM and EX stages can be overlapped and the pipeline has only 4 stages. Change this code to accommodate this changed ISA. Assuming this change does not affect clock cycle time, what speedup is achieved in this instruction sequence? (10分)

6. Caches are important to providing a high-performance memory hierarchy to processors. Below is a list of 32-bit memory address references, given as word addresses.

3, 180, 43, 2, 191, 88, 190, 14, 181, 42, 186, 253

You are asked to optimize a cache design for the given references. There are three direct-mapped cache designs possible, all with a total of 8 words of data: C1 has 1-word blocks, C2 has 2-word blocks, and C3 has 4-word blocks. In terms of miss rate, which cache design is the best (total cycles)? If the miss stall time is 25 cycles, and C1 has an access time of 2 cycles, C2 takes 3 cycles, and C3 takes 5 cycles, which is the best cache design? (35分)

There are many different design parameters that are important to a cache's overall performance. Below are listed parameters for different direct-mapped cache designs.

Cache Data Size: 32 KiB
Cache Block Size: 2 words
Cache Access Time: 1 cycle

| Word Address | Binary Address | Tag | Cache 1 | | Cache 2 | | Cache 3 | |
|--------------|----------------|-----|---------|----------|---------|----------|---------|----------|
| | | | index | hit/miss | index | hit/miss | index | hit/miss |
| 3 | 000 0011 | 0 | 3 | ✓ | 1 | M | 0 | M |
| 180 | 101 0100 | 1 | 8 | M | 2 | M | 1 | M |
| 43 | 00 01011 | 2 | 1 | M | 5 | M | 2 | M |
| 2 | 0000 0010 | 0 | 2 | M | 1 | H | 0 | M |
| 191 | 101 1111 | 1 | 15 | M | 7 | M | 3 | M |
| 88 | 0101 1000 | 5 | 8 | M | 4 | M | 2 | M |
| 190 | 101 1110 | 1 | 14 | M | 7 | H | 3 | M |
| 14 | 0100 1110 | 0 | 14 | M | 7 | M | 3 | M |
| 181 | 101 0101 | 1 | 5 | M | 2 | H | 1 | M |
| 42 | 0010 1010 | 2 | 10 | M | 5 | M | 2 | M |
| 186 | 101 1010 | 1 | 10 | M | 5 | M | 2 | M |
| 253 | 1111 1101 | 15 | 13 | M | 6 | M | 3 | M |

5.2

IF ID EX Mem WB

IF ID EX Mem WB

IF ID EX Mem WB

IF ID EX Mem WB

(IF ID EX Mem WB)

5 stage \Rightarrow 9 cycles

4 stage \Rightarrow 8 cycles

speed up: $\frac{9}{8} = 1.13$

Cache 1 miss rate: 0
 Cache 1 total cycles: 0
 Cache 2 miss rate: 0
 Cache 2 total cycles: 0
 Cache 3 miss rate: 0
 Cache 3 total cycles: 0

Which cache design provides the best performance: *Cache 1*

7. For a direct-mapped cache design with a 32-bit address, the following bits of the address are used to access the cache.

| Tag | Index | Offset |
|-------|-------|--------|
| 31-10 | 9-4 | 3-0 |

7.1 What is the cache block size (in words)? (5 分)

16

7.2 How many entries does the cache have? (5 分)

64

| Word Address | Binary Address | Tag | Cache 1 | | Cache 2 | | Cache 3 | |
|--------------|----------------|-----|---------|----------|---------|----------|---------|----------|
| | | | index | hit/miss | index | hit/miss | index | hit/miss |
| 3 | 0000 0011 | 0 | 3 | M | 1 | M | 0 | M |
| 180 | 1011 0100 | 22 | 4 | M | 2 | M | 1 | M |
| 43 | 00101011 | 5 | 3 | M | 1 | M | 0 | M |
| 2 | 0000 0010 | 0 | 2 | M | 1 | M | 0 | M |
| 191 | 10111111 | 23 | 7 | M | 3 | M | 1 | M |
| 88 | 01011000 | 11 | 0 | M | 0 | M | 0 | M |
| 190 | 10111110 | 23 | 6 | M | 3 | H | 1 | H |
| 14 | 00001110 | 1 | 6 | M | 3 | M | 1 | M |
| 181 | 10110101 | 22 | 5 | M | 2 | H | 1 | M |
| 42 | 00101010 | 5 | 2 | M | 1 | M | 0 | M |
| 186 | 10111010 | 23 | 2 | M | 1 | M | 0 | M |
| 253 | 11111101 | 31 | 5 | M | 2 | M | 1 | M |

cycle = miss 數 * 停滯
 + 全部數量 * 快取時間

Cache 1 Miss rate: 100%

Cache 1 total cycle

$$= 12 \times 25 + 12 \times 2 = 324$$

Cache 1 Miss rate: $\frac{10}{12} = 0.83$

Cache 2 total cycle:

$$0 \times 25 + 12 \times 3 = 286$$

Cache 3 Miss rate: $\frac{11}{12} = 0.916$

Cache 3 total cycle

$$= 11 \times 25 + 12 \times 5 = 335$$

Cache 2 有最棒的 performance

Cache 1 miss rate : 0
 Cache 1 total cycles : 0
 Cache 2 miss rate : 0
 Cache 2 total cycles : 0
 Cache 3 miss rate : 0
 Cache 3 total cycles : 0
 Which cache design provides the best performance : Cache 1

7. For a direct-mapped cache design with a 32-bit address, the following bits of the address are used to access the cache.

| Tag | Index | Offset |
|-------|-------|--------|
| 31-10 | 9-4 | 3-0 |

7.1 What is the cache block size (in words)? (5 分)

$$2^4 = 16 \text{ bytes} / 4 = 4$$

7.2 How many entries does the cache have? (5 分)

0

7.1

用 3 4 個 offset

所以 用 $2^4 = 16 \text{ bytes}$

$16 \text{ bytes} / 4 = 4 \text{ words}$

7.2

用 6 個 Index

所以 $2^6 = 64 \text{ entries}$