Computer Architecture

Fall, 2024
Week 7
2024.10.14

組別：_____ 簽名：_____

[group 1]

1. Qustion:

Fill the blank below

Single precision		**Double precision**		**Object represented**
Exponent	**Fraction**	**Exponent**	**Fraction**	
?	?	?	?	±0
0	nonzero	0	nonzero	?
?	?	?	?	Normal floating-point number
255	0	2047	0	?
255	nonzero	2047	nonzero	?

Ans:

Single precision		**Double precision**		**Object represented**
Exponent	**Fraction**	**Exponent**	**Fraction**	
0	**0**	**0**	**0**	±0
0	nonzero	0	nonzero	**Denormalized number**
1~254	**any**	**1~2046**	**any**	Normal floating-point number
255	0	2047	0	**±Infinity**
255	nonzero	2047	nonzero	**Not a Number(NaN)**

[group 2]

2. Question:

which of the following statements are true?

(A) mul rd, rs, rt in MIPS can store 64 bits of product to rd.

(B) we don't need ALU to subtract in multiply hardware.

(C) if some value are divided by 0, MIPS will raise an exception.

(D) 1 10000110 01100000110000000000000 in IEEE 754 represents -176.375 in decimal

(E) In biased 15, 10110 represents 7.

Ans:

(D)(E)

(A) only 32 bits（least significant 32 bits）

(B) signed multiply need subtraction.

(C) MIPS don't check.

(D) True.

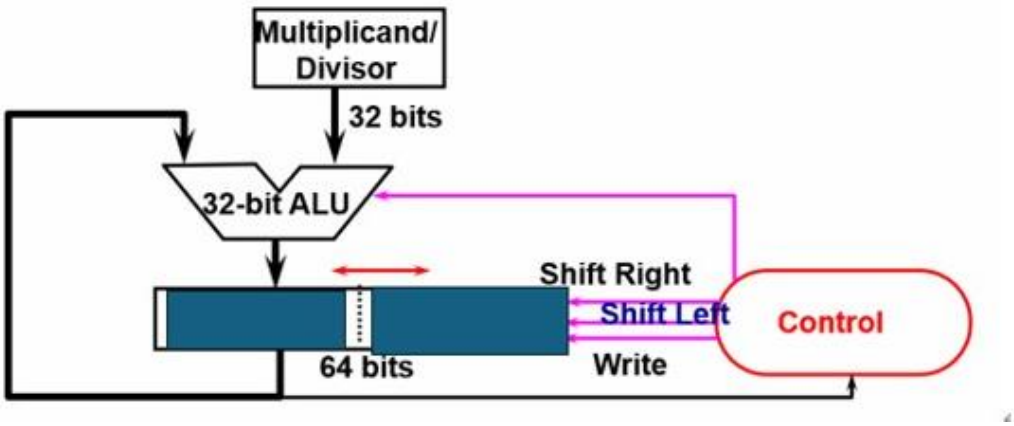
(E) True.

10110➞22

22-15=7

[group 3]

3. Question:



1.請問圖中 64bits 的暫存器在進行乘法與除法運算時會往哪個方向 shift?

2.請問 Multiplier 一開始會擺在最右邊還是最左邊而 Quotient 最後會出現在最右邊還最左邊?

Ans:

1.乘法向右邊 shift，除法向左邊 shift

2.都是右邊

[group 5]

4. Question:

Calculate A+B present the result by both decimal and single precision

A:0 | 1000 0110 | 0110 0000 0000 0000 0000 000

B:0 | 1000 0100 | 0011 1000 0000 0000 0000 000

Ans:

A 的轉換：

- 符號位：0（正數）
- 指數位：1000 0110 = 134
- 偏移：134 - 127 = 7
- 尾數位：1.0110 0000 0000 0000 0000 000
- 轉換公式：$1.0110 \times 2^7 = 1.375 \times 2^7 = 176$

B 的轉換：

- 符號位：0（正數）
- 指數位：1000 0100 = 132
- 偏移：132 - 127 = 5
- 尾數位：1.0011 1000 0000 0000 0000 000
- 轉換公式：$1.0011\ 1 \times 2^5 = 1.21875 \times 2^5 = 39$

Decimal: A+B= 176 + 39 = 215

Single precision:

$A+B=1.0110 \times 2^7+1.0011\ 1 \times 2^5=1.0110 \times 2^7+1.110011\ 1\times 2^7=1.1010111\times 2^7$

=>0｜10000110｜10101110000000000000000

[group 6]

5. Question:

請將以下 floating-point 轉換為十進位。

(1) 0 10011011 01 1001 0000 0000 0000 00000

(2) 1 00111001 10 0010 1000 0000 0000 00000

Ans:

(1)

Sign: 0 => positive

Exponent:

10011011(two)=155(ten)

Bias adjustment: 155-127=28

Significand: 01 1001 0000 0000 0000 0000 0(two) = 1.390625(ten)

Ans: 1.390625 * (2^28)

(2)

Sign: 1 => negative

Exponent:

00111001(two)=57(ten)

Bias adjustment: 57-127=-70

Significand: 10 0010 1000 0000 0000 0000 0(two) = 1.5390625 (ten)

Ans: -1.5390625 * (2^-70)

[group 7]

6. Question:

Consider IEEE 754 single precision floating-point format. Which of the following statements are correct? And where are wrong?

(a) The FP number representation is (-1)^sign * (significand) * 2^(exp-bias)

(b) The smallest positive normalized number is 1.0 * 2^(-125)

(c) The smallest positive denormalized number is 1.0 * 2^(-149)

(d) Converting an integer variable to a single precision FP number will lose precision.

Ans:

Ans: a, c, d

(b) 1.0 * 2^(-126)

[group 8]

7. Question:

是非題（錯誤請說明原因):

a. The mul instruction does not modify the hi and lo registers, but mult does.

b. Booth’ s algorithm (to rationale) only works if there’ s only one sequence of 1’ s.

c. The Hi and Lo registers in MIPS combine to act as a 64-bit register for multiply and divide

d. In Biased 7 notation, 0011 is 6

Ans:

a. True

b. False, it can also work if there are multiple sequences of 1’ s

c. True

d. False, 0011 is -4, 6 is 1101

[group 9]

8. Question:

1. True or False: If false, please explain why.

A. Multiplication and division can use the same hardware.

B. Overflow can be detected in MIPS division.

C. Overflow can be detected in MIPS multiplication.

D. When performing signed division, it can first be treated as unsigned (positive) division.

Ans:

A. True

B. False, software must perform checks if required

C. True

D. True

[group 10]

9. Question:

The following questions involve converting between hexadecimal and decimal representations of 32-bit floating-point numbers based on the IEEE 754 standard.

Questions:

1. Hex to Decimal: Convert the hexadecimal number 0x420A0000 to decimal.
2. Hex to Decimal: Convert the hexadecimal number 0xC1A80000 to decimal.
3. Decimal to Hex: Convert the decimal number 3.14 to hexadecimal.
4. Decimal to Hex: Convert the decimal number -100.5 to hexadecimal.

Ans:

1. 0x420A0000 => 34.5
2. 0xC1A80000 => -21
3. 3.14 => 0x4048F5C3
4. -100.5 => 0xC2C90000

[group 12]

10. Question:

Can computer represents all floating numbers? If not, what floating number will it pick to store for a non representable number?

Ans:

No. It will pick the nearest floating number of the non representable number to store it.

[group 14]

11. Question:

Write down two multiplication rules for 2's complement and calculate -11 * -13, showing the detailed steps of the process.

Ans:

Rule 1: Multiplicand sign extended

Rule 2: Sign bit (s) of Multiplier