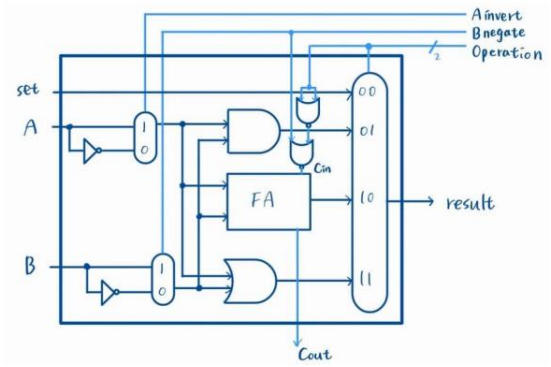
CS4100 Computer Architecture
Spring 2022 Homework 3
Answer

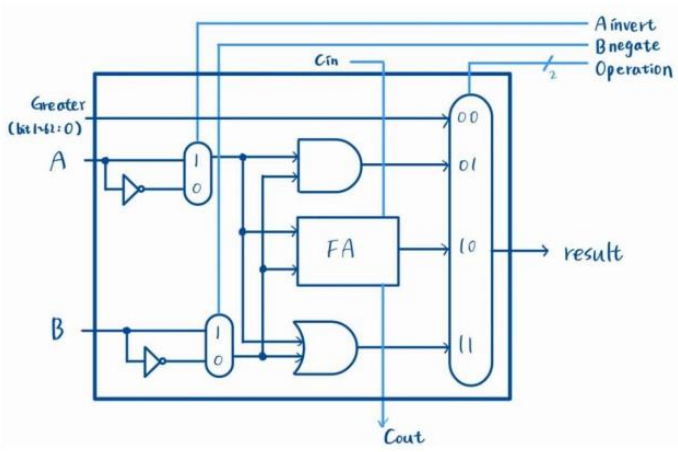
1. (15 points)

A denotes Ainvert and B denotes Bnegate

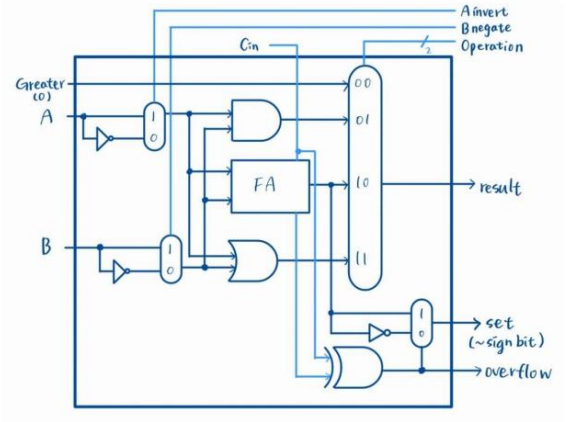
- ALU 0



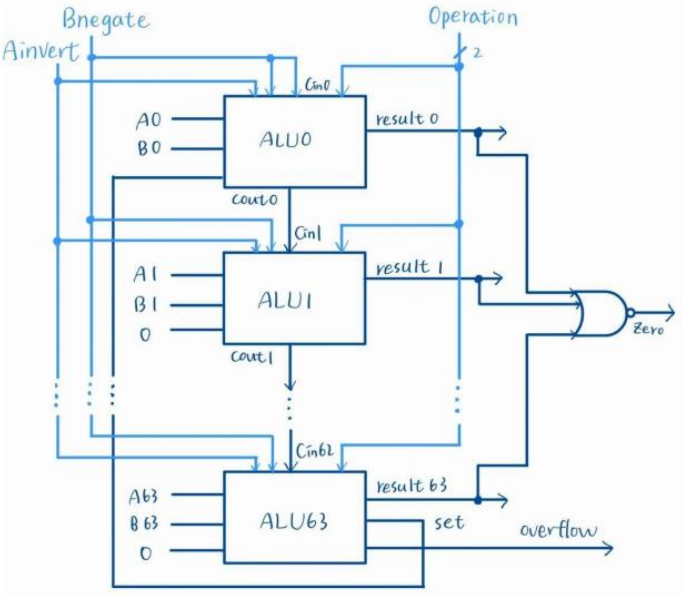
- ALU 1~62



- ALU 63



- 64-bit ALU



2. (36 points)

(a) version 1 of the multiply algorithm

Iteration	Step	Multiplier	Multiplicand	Product
0	Initial values	0101	00001010	00000000
1	1a: 1 => Prod = Prod + Mcand	0101	00001010	00001010
	2: Shift left Multiplicand	0101	00010100	00001010
	3: Shift right Multiplier	0010	00010100	00001010
2	1: 0 => No operation	0010	00010100	00001010
	2: Shift left Multiplicand	0010	00101000	00001010
	3: Shift right Multiplier	0001	00101000	00001010
3	1a: 1 => Prod = Prod + Mcand	0001	00101000	00110010
	2: Shift left Multiplicand	0001	01010000	00110010
	3: Shift right Multiplier	0000	01010000	00110010
4	1: 0 => No operation	0000	01010000	00110010
	2: Shift left Multiplicand	0000	10100000	00110010
	3: Shift right Multiplier	0000	10100000	00110010

(b) version 2 of the multiply algorithm

Iteration	Step	Multiplicand	Product
0	Initial values	1010	00000101
1	1a: 1 => Product[left] = Product[left] + Mcand	1010	10100101
	2: Shift right Product	1010	01010010
2	2: Shift right Product	1010	00101001
3	1a: 1 => Product[left] = Product[left] + Mcand	1010	11001001
	2: Shift right Product	1010	01100100
4	2: Shift right Product	1010	00110010

(c) version 1 of the divide algorithm

Iteration	Step	Quotient	Divisor	Remainder
0	Initial values	0000	01010000	00001010
1	1: Rem = Rem - Div	0000	01010000	10111010
	2b: Rem < 0 => +Div, LSL Q, Q0 = 0	0000	01010000	00001010
	3: Shift right Div	0000	00101000	00001010
2	1: Rem = Rem - Div	0000	00101000	11100010
	2b: Rem < 0 => +Div, LSL Q, Q0 = 0	0000	00101000	00001010
	3: Shift right Div	0000	00010100	00001010
3	1: Rem = Rem - Div	0000	00010100	11110110
	2b: Rem < 0 => +Div, LSL Q, Q0 = 0	0000	00010100	00001010
	3: Shift right Div	0000	00001010	00001010
4	1: Rem = Rem - Div	0000	00001010	00000000
	2a: Rem >= 0 => LSL Q, Q0 = 1	0001	00001010	00000000
	3: Shift right Div	0001	00000101	00000000
5	1: Rem = Rem - Div	0001	00000101	11111011
	2b: Rem < 0 => +Div, LSL Q, Q0 = 0	0010	00000101	00000000
	3: Shift right Div	0010	00000010	00000000

(d) version 2 of the divide algorithm

Iteration	Step	Remainder	Divisor
0	Initial values	00001010	0101
1	1: Shift left Remainder	00010100	0101
	2: Remainder[left] = Remainder[left] - Div	11000100	0101
	3b: +Div, Shift left Remainder, Set Remainder0 = 0	00101000	0101
2	2: Remainder[left] = Remainder[left] - Div	11011000	0101
	3b: +Div, Shift left Remainder, Set Remainder0 = 0	01010000	0101
3	2: Remainder[left] = Remainder[left] - Div	00000000	0101
	3a: Shift left Remainder, Set Remainder0 = 1	00000001	0101
4	2: Remainder[left] = Remainder[left] - Div	10110001	0101
	3b: +Div, Shift left Remainder, Set Remainder0 = 0	00000010	0101
Done	Shift right Remainder[left]	00000010	0101

3. (20 points)

(a)

783.3125_{10} =

S	E	F
0	10001000	10001000101010000000000

-13.125_{10} =

S	E	F
1	10000010	10100100000000000000000

(b) 加法步驟 (5 分, 每個步驟一分，答案錯全扣)

1 ：將指數較小的數的有效數字部分向右移位，直到它的指數與較大的數的指數部分一樣為止。

較大的數的指數部分 : 9 => -13.125 向右位移 6 位數

2 : 將有效數字相加。=>相加後 : 1.10000010000110000000000

3 : 將總和正規化，並檢查是否有溢位。=> 總和已符合正規化規定且無溢位

4 : 將結果四捨五入。=>結果不用四捨五入

答案 :

S	E	F
0	10001000	10000010000110000000000

(c) 乘法步驟 (5 分, 每個步驟一分，答案錯全扣)

1 : 將兩指數相加後減掉 bias(127)，計算乘積的指數。=>指數: 10001011

2 : 將兩個有效數字相乘。=>乘積 : 10.100001000011001110100000

3 : 正規化乘積應並檢查指數是否會導致 overflow 或 underflow。

=>正規化後的乘積 : 1.010000100001100111010000

指數: 10001100

=>沒有 overflow 或 underflow

4 : 四捨五入乘積。若四捨五入的結果會讓有效數字變成非正規格式，則再正規化一次，並且 y 再檢查指數的大小。

=>四捨五入後的乘積 : 1.01000010000110011101000

5 : 設定乘積正負號—如果兩運算元異號則乘積符號位元為 1，否則為 0。

=>乘積符號位元為 1

答案 :

S	E	F
1	10001100	01000010000110011101000

4. (9 points)

(a) -7467501

(b) NaN

(c) yes, addi x28, x28, -8

5. (20 points)

(a) a0 = $1.0000000000 * 2^{-14}$

(b) a1 = $0.1111111111 * 2^{-14}$, a2 = $0.1111111110 * 2^{-14}$

(c) Both are $0.0000000001 * 2^{-14}$，normalized 到 denormalized 的間距與 denormalized 內的間距是相同的。

(d) $-1\frac{423}{1024}$

(e) F = 0 01111 0011110110 = $1\frac{246}{1024}$