Computer Architecture

Fall, 2020

Week 7

2020.10.26

Group:

組員簽名：_____

[group1]

1.

a	$s0=0x70 000 000 (hex)	$s1=0x0F FFF FFF (hex)
b	$s0=0x40 000 000 (hex)	$s1=0x40 000 000 (hex)

For the contents of registers $s0 and $s1 as specified above, what is the value of $t0 for the following assembly code respectively?

A.

```
add $t0, $s0, $s1
```

B.

```
sub $t0 ,$s0,$s1
```

C.

```
add $t0,$s0,$s1
add $$t0,$s0,$t0
```

Are above results in $t0 the desired result or has there been overflow?

Ans :

	A	B	C
a	0x7F FFF FFF, no overflow	60 000 001,no overflow	EF FFF FFF, overflow
b	0x80 000 000, overflow	0 , no overflow	C0 000 000, overflow

[group11]

2. Which of the following statements are true?

(A) when overflow occurs, Carry into MSB = Carry out of MSB

(B) When adding operands with different signs, overflow cannot occur

(C) In carry look ahead, Propagate Carry via Bit i: pi = Ai nor Bi

(D) We can use a Nor gate (input are results of 1 bit ALU) to detect zero

Ans: (B) (D)

(A) not equal

(C)xor

[group2] (對抗賽)

3. Please find whether the followings are true or false, if false , correct it.

(A) According to Design trick 1, We should take the problem into simple pieces and solve them, then glue them together.

(B) In NOR operation, i.e. A nor B, We take 0 in A and B ,and take 1 in AND-gate, so the control signal is 0011.

(C) For 4-bit binary number, if it >8 or <-9, then overflow will occurs.

(D) In programming language, some will use addu, addui, subu, etc to require raising an exception; some will use add, addi, sub to ignore overflow.

(E) We define Generate Carry at bit i is Ai xor Bi, Propagate Carry via Bit i is Ai*Bi.

Ans :

(A) (C) true

(B) False, take 1 in A and B to get negate, and take 0 for ”and”, so the signal should be 1100

(D) False, using addu, addui, subu to ignore overflow and using add, addi, subi to require raising an exception.

(E) False, Generate Carry at bit i is Ai*Bi , Propagate Carry via Bit is Ai xor Bi.

[group 5] (對抗賽)

4. Which statements is incorrect?

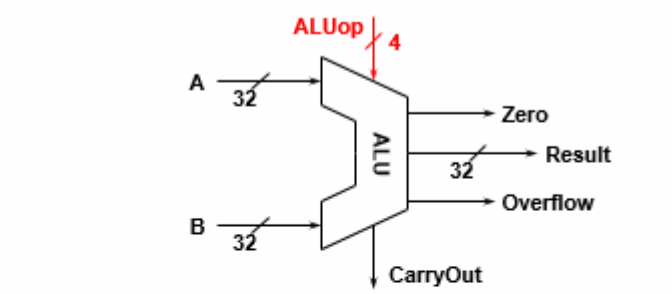
(A) If “Carry into MSB” doesn’t equal to “Carry out of MSB”, the sum of two sign numbers will occur overflow.

(B) According to the picture, when we choose “add” function, ALU only runs the add circuit.

(C) Zero Detection Logic is a one BIG(multiple input) NOR gate.

(D) We can build a “full” carry look-ahead adder because it will be fast.

(E) In ALU, it runs A-B ,and then choose the LSB of the result to determine whether A is less than B.



ALU Control (ALUop)	Function
0000	and
0001	or
0010	add
0110	subtract
0111	set-on-less-than
1100	nor

Ans:

(B) All the circuit will run, but only “add” result will be selected by the “add” function.

(D) We can’t build because it is expensive

(E) LSB->MSB

[group7] (對抗賽)

5. Carry lookahead is often used to speed up the addition operation in ALU. For a 4-bit addition with carry lookahead, assuming the two 4-bit inputs are a3a2a1a1 and b3b2b1b0, and the carry-in is c0,

(1) Fist derive the recursive equations of carry-out ci+1 in terms of ai and bi and ci, where i = 0, 1, 2…

(2) Then by defining the generate(gi) and propagate(pi) signals, express c1, c2, c3, and c4 in terms of only gi's, pi's and c0.

(3) Estimate the speed up for this 4-bit carry lookahead adder over the 4-bit ripple carry adder(assuming each logic operation introduces T delays)

Ans :

(1) ci+1 = aibi+aici+bici

(2) c1 = g0 + (p0c0)

c2 = g1 + (p1g0) + (p1p0c0)

c3 = g2 + (p2g1) + (p2p1g0) + (p2p1p0c0)

c4 = g3 + (p3g2) + (p3p2g1) + (p3p2p1g0) + (p3p2p1p0c0)

(3) RCA:2T*4 = 8T, CLA = 2T+T = 3T

speedup = 8T/3T = 2.67

[group10] (對抗賽)

6. 我們以 4 bit Multiple Level Carry Look-ahead Adder 計算下列 16bits 數字相加產生的 gi，pi，Pi，Gi，並用 Pi,Gi 計算 Carryout15 的值。

a. $1000\ 0111\ 0101\ 0011_2$

b. $0011\ 1001\ 1101\ 0110_2$

Ans : gi : $0000\ 0001\ 0101\ 0010_2$，pi : $1011\ 1110\ 1000\ 0101_2$

P3 = 1*0*1*1 = 0，P2 = 1*1*1*0 = 0，P1 = 1*0*0*0 = 0， P0 = 0*1*0*1 = 0

G0 = g3+(p3*g2)+(p3*p2*g1)+(p3*p2*p1*g0)

= 0+(0*0)+(0*1*1)+(0*1*0*0)

= 0

G1 = g7+(p7*g6)+(p7*p6*g5)+(p7*p6*p5*g4)

= 0+(1*1)+(1*0*0)+(1*0*0*1)

= 1

G2 = g11+(p11*g10)+(p11*p10*g9)+(p11*p10*p9*g8)

= 0+(1*0)+(1*1*0)+(1*1*1*1)

= 1

G3 = g15+(p15*g14)+(p15*p14*g13)+(p15*p14*p13*g12)

= 0+(1*0)+(1*0*0)+(1*0*1*0)

= 0

c15 = C4

= G3+(P3*G2)+(P3*P2*G1)+(P3*P2*P1*G0)

= 0+(0*1)+(0*0*1)+(0*0*0*1) =0

[group12] (對抗賽)

7. Consider two code sequences for detecting the overflow of addition on two numbers in registers $t1 and $t2. One addition is signed addition while the other is unsigned addition.

(1) The code sequence in bottom left figure is to detect signed addition overflow. Complete the following MIPS code and explain how it can detect signed addition overflow.

(2) The code sequence in bottom right figure is to detect unsigned addition overflow. Complete the following MIPS code and explain how it can detect unsigned addition overflow.

```
addu    $t0, $t1, $t2
_①_     $t3, $t1, $t2
slt     $t3, $t3, $zero
bne     $t3, $zero, __③___
_②_     $t3, $t0, $t1
slt     $t3, $t3, $zero
bne     $t3, $zero, __④___
```

```
addu    $t0, $t1, $t2
_⑤_     $t3, $t1, $zero
sltu    $t3, $t3, $t2
bne     $t3, $zero, Overflow
```

Hint: for ①、②、⑤ you can use and、or、xor、nor、not
for ③、④ you can use No_Overflow 、 Overflow

Ans: ① xor ;② xor;③ No_Overflow; ④ Overflow; ⑤ nor

Explanation:

(1) 兩個異號相加不會發生 overflow, 而兩個正數相加變負數或兩個負數相加變正數便會發生 overflow

(2) $t1 + $t2 > (2^32) -1 → overflow
$t2 > (2^32)-1-$t1 → overflow
$t2 > ($t1)' → overflow

[group14]

8. Overflow may occur when

(a) positive number - positive number
(b) positive number - negative number
(c) negative number - positive number
(d) negative number - negative number
(e) positive number + positive number
(f) positive number + negative number
(g) negative number + negative number

Ans : (b), (c), (e), (g)

[group3] (對抗賽)

9. Consider cascaded carry look-ahead method. We would like to compute 32bit numbers. Given that the input bitwise for each ALU is 4, how many stage delays we have to at least experience to get the 27th bit? (Notice: the No. of bit is 1-base. It means that we name bits from 1 to 32)(Definition: “stage delay” is the processing time a block took)

Ans :

因為每個 Block 有 4 個 inputs，算到第 27bit 至少經過七個 stage(分別為[0:3]、[4:7]、[8:11] 、[12:15]、 [16:19]、 [20:23]、 [24:27])

因為每個 stage 會先 and 再 or，共兩個 stage delay，因此 7*2=14，全部共計 14 個 stage delay

[group13] (對抗賽)

10. State the following statement is true or false, giving an explanation if the statement is false.

(A) 0000 is an ALU control signal of output "AND".

(B) 0010 is an ALU control signal of output "OR".

(C) In set-on-less-than, if A is greater than B the output will be 0001 in 32 bits representation, otherwise, the output will be 0000.

(D) The range that overflow might occur is when the 4-bit binary number is greater than or equal to 7 and smaller than or equal to -8.

(E) In MIPS, add, addi, and sub are instructions that see a number as a signed-number. In contrast, addu, addui, and subu instructions see a number as an unsigned-number.

(F) pi = 0 if one of Ai and Bi is 1.

(G) P and G produced by each block (4-bit carry looked ahead unit) can be used to calculate all carryIn directly and rapidly.

Ans :

a. True.

b. False, 0010 is an ALU control signal of output "ADD". ALU control signal of output "OR" is 0001

c. False, if A > B, the output should be 0000, otherwise, should be 0001.

d. False, 8>= 4-bit binary number >=7 is the range that overflow won't occur, 4-bit binary number out of this range occurs overflow.

e. True.

f. False, pi = 1 if one of Ai and Bi is 1. Or, pi = 0 if Ai or Bi is 0.

g. True.