

Q1 :

A + B (signé) 0000φφ

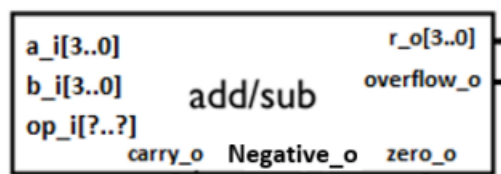
A - B (signé) 0010φφ

A + B (non-signé) 0001φφ

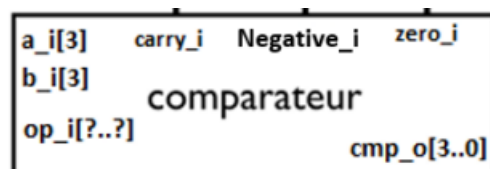
A - B (non-signé) 0011φφ

Tous ces opcodes peuvent commettre une erreur car il peut y avoir un overflow.

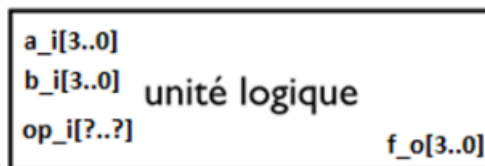
Q2 :



On doit prendre les bits 2 et 3



On doit prendre les bits de 2 à 0



On doit prendre les bits 1 et 0

Q3 :

On a mis un xor entre le 3^{ème} bit du résultat et l'overflow pour vérifier si le négatif est juste. Puis on regarde si c'est signé ou non, en faisant un 'et' sur l'inversion du 1^{er} bit du opcode et du résultat du xor.

Q4 :

A	B	C	D	
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0
8	1	0	0	1
9	1	0	1	0
10	1	1	0	0
11	1	1	1	0
12	1	1	0	0
13	1	1	1	0
14	1	1	1	0
15	1	1	1	0

A			
0	1	12	8
1	5	13	9
3	7	15	11
1	6	14	10
B			

Simplification : $A'B'C'D + A'B'CD' + A'BC'D' + AB'C'D'$

Q5 :

Les cas où l'opcode n'existent pas pour le comparateur.

010xxx (toutes les variantes disponibles)

011000

011111

Q6 :

Il n'y a pas d'overflow et il n'y a pas d'erreur. Cela correspond à nos attentes.