

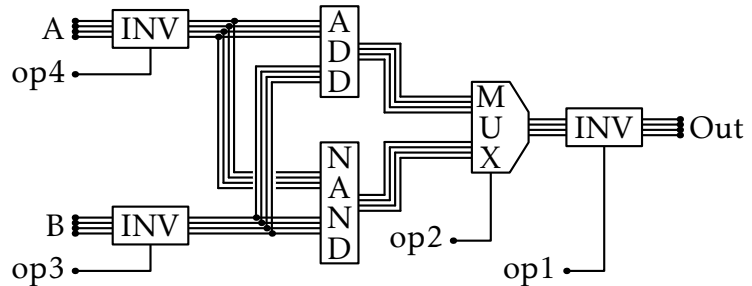
Unité Arithmétique et Logique

UAL

L'Unité Arithmétique et Logique (ou UAL) est la partie du processeur qui réalise les calculs et les opérations logiques élémentaires. Selon le processeur, cette unité peut réaliser des opérations plus ou moins complexes, sur des nombres plus ou moins longs.

Nous allons considérer un UAL simpliste pouvant travailler sur des nombres de 4 bits. Dans la suite, nous noterons \bar{A} l'opposé de A bit à bit. L'architecture globale est la suivante :

- INV: renvoie A si op = 0 et \bar{A} sinon.
- ADD: renvoie la somme des deux nombres.
- NAND: renvoie le résultat de l'opération logique.
- MUX: renvoie A si op = 0 et B sinon.



Les bits op4 à op1 correspondent au **code opération** (*opcode*). Ce code sert à déterminer le type d'opération effectuée par l'unité. Par exemple, avec le code 0000, l'unité réalise $A + B$, alors qu'avec 0010, elle réalise $A \text{ nand } B$.

La plupart des UAL ont des bits supplémentaires en sortie permettant de savoir s'il reste une retenue à la fin de la somme (overflow), si le résultat est négatif (les entiers sont représentés en complément à 2), égal à zéro...

Exemples d'utilisation

Le tableau ci-contre présente quelques exemples d'utilisation. On remarque qu'il est possible de faire une soustraction. En effet, par définition du complément à 2, on a $-A = \bar{A} + 1$, donc $\bar{A} = -A - 1$.
Pour la 4^e ligne, l'opération effectuée correspond à $\bar{A} + B$. On a donc :
 $\bar{A} + B = -A - 1 + B = -(-A + B - 1) - 1 = A - B$

opcode	A	B	Out	commentaire
0000	0011	0101	1000	A + B
0000	0001	1111	0000	overflow
0101	0011	0101	0010	B - A
1001	0011	0101	1110	A - B
0101	0011	0000	1101	-A
0001	1100	0000	0011	non A
0010	0011	0101	1110	A nand B
0011	0011	0101	0001	A et B
1110	0011	0101	0111	A ou B

EXERCICE : Déterminer le résultat des opérations suivantes :

opcode	A	B	Out
0000	1011	0110	
0101	1011	0110	
1001	1011	0110	
0010	1011	0110	
0011	1011	0110	
1110	1011	0110	
0101	1011	0000	
0001	1011	0000	