

Introduction / Guide

Dans le cadre de notre cursus à école nationale des sciences appliquées de Tanger. Nous avons à réaliser une **ALU** (Unité arithmétique et logique).

Pour ce faire, plusieurs séances de cours ont été faites afin de concevoir chaque bloc élémentaire avant de l'insérer dans un bloc plus complexe.

En fait, l'unité arithmétique et logique est l'organe de l'ordinateur chargé d'effectuer les calculs. Le plus souvent, l'ALU est inclus dans l'unité centrale ou le microprocesseur. Elle permet d'effectuer des opérations communes rapidement. Nous allons donc en créer une pouvant réaliser pas moins de 26 opérations différentes.

Celle que nous devons réaliser doit répondre à certaines contraintes :

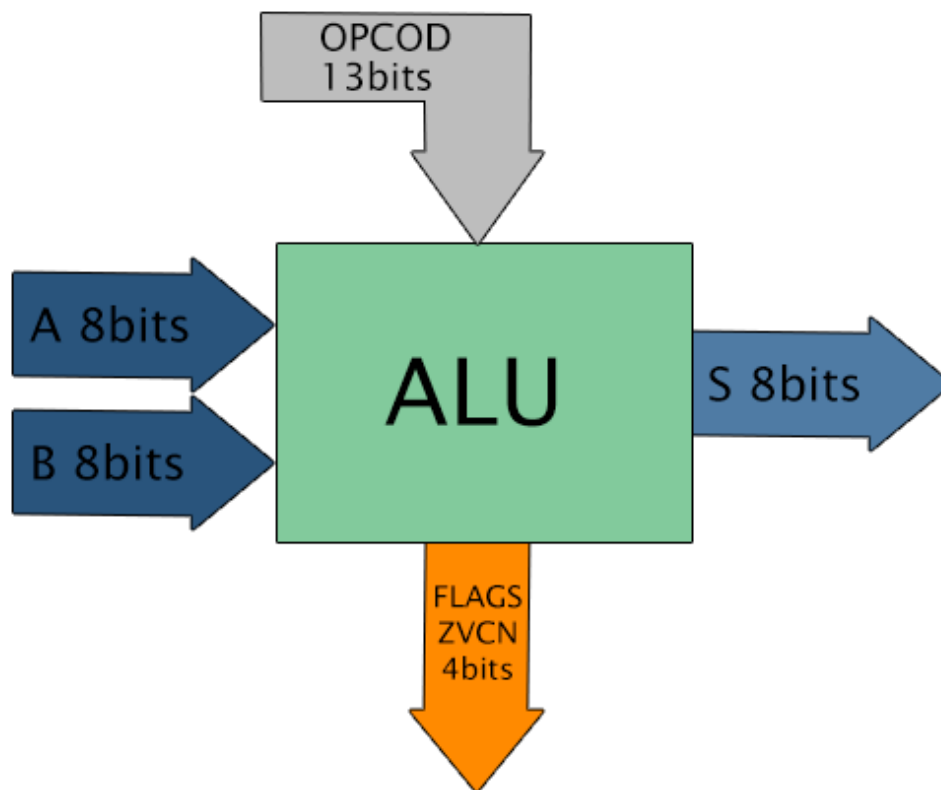


Schéma ALU 8 bits

- **A, B** : signaux d'entrée codés sur 8 bits (type donnée).
- **S** : signal de sortie codé sur 8 bits (type donnée).
S est le résultat de l'opération.
- **ZVCN** : signal de sortie codé sur 4 bits (type sortie de contrôle).
 - bit Z : Zéros (pour savoir si le résultat vaut zéro)
=> Z=1 si le résultat vaut zéro.
 - bit V : détection de débordement (Overflow).
 - bit C : Retenue (Carry donne la retenue de l'opération courante)
 - bit N : Comparaison
=> N=0 si A<B N=1 si non
- **CODOP** : signal d'entrée codé sur 13 bits (type entrée de contrôle).
C'est le signal qui permet de sélectionner l'opération courante parmi les 26 possibles.

OPERATION	Equivalent en MIPS	OPCOD 13bits		A 8bits	B 8 bits
S=A & B	and	00000000 00000		A (8bits)	B (8bits)
S=~ (A B)	nor	00000000 00001		A (8bits)	B (8bits)
S= A B	or	00000000 00010		A (8bits)	B (8bits)
S= A ^ B	xor	00000000 00011		A (8bits)	B (8bits)
S= A + B	add	00000000 00100		A (8bits)	B (8bits)
S= A + B	addu	00000000 00101		A (8bits)	B (8bits)
S= A – B	sub	00000000 00110		A (8bits)	B (8bits)
S= A – B	subu	00000000 00111		A (8bits)	B (8bits)
If(A<B) S=1 ;Else S=0 ;	slt	00000000 01000		A (8bits)	B (8bits)
If(A<B) S=1 ;Else S=0 ;	sltu	00000000 01001		A (8bits)	B (8bits)
S=A << B	sllv	00000000 01010		A (8bits)	B (8bits)
S=A << B	sra	00000000 01011		A (8bits)	B (8bits)
S=A >> B	srlv	00000000 01100		A (8bits)	B (8bits)
Avec la valeur immédiate k		K sur 8bis	OPCOD 5bits		
S=A & k	andi	k	10000	A (8bits)	00000000
S=~ (A k)	nori	k	10001	A (8bits)	00000000
S= A k	ori	k	10010	A (8bits)	00000000
S= A ^ k	xori	k	10011	A (8bits)	00000000
S= A + k	add	k	10100	A (8bits)	00000000
S= A + k	addui	k	10101	A (8bits)	00000000
S= A – k	subi	k	10110	A (8bits)	00000000
S= A – k	subui	k	10111	A (8bits)	00000000
If(A<k) S=1 ;Else S=0 ;	slti	k	11000	A (8bits)	00000000
If(A<k) S=1 ;Else S=0 ;	sltui	k	11001	A (8bits)	00000000
S=A << k	sll	k	11010	A (8bits)	00000000
S=A << k	sra	k	11011	A (8bits)	00000000
S=A >> k	srl	k	11100	A (8bits)	00000000