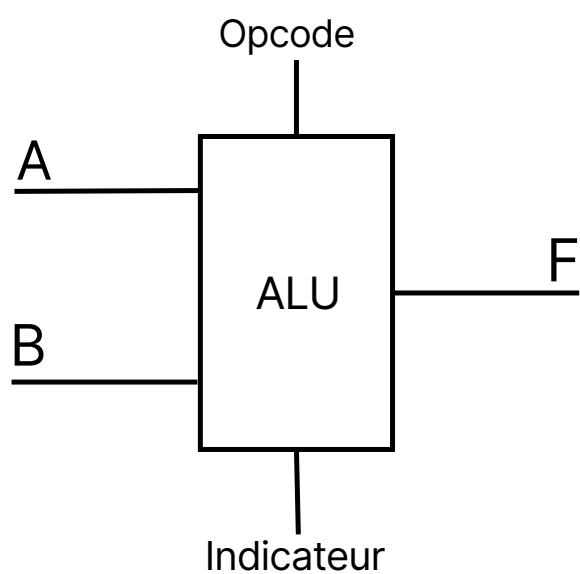


Labo ALU :

Schema :

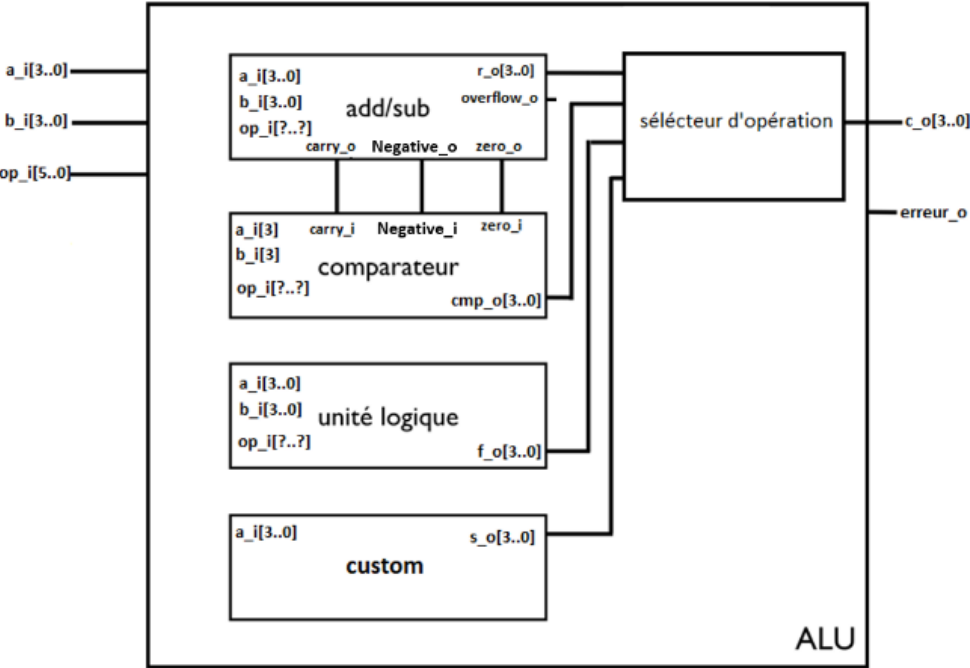


Operations à faire :

Opération	Fonctionnalité	Opcode
A + B (signé)	Add/Sub	0000φφ
A - B (signé)		0010φφ
A + B (non-signé)		0001φφ
A - B (non-signé)		0011φφ
A ≥ B (signé)	Comparateur	011001
A < B (signé)		011010
A ≠ B		011011
A = B		011100
A ≥ B (non-signé)		011101
A < B (non-signé)		011110
A and B	Logique	10φφ00
A or B		10φφ01
A nand B (non-signé)		10φφ10
A xor B (non-signé)		10φφ11
Détecteur de puissances de 2	Custom	11φφφφ

ALU divisé en 4 blocs :

1. Circuit combinatoire additionneur/soustracteur.
2. Circuit combinatoire exploitant le resultat du soustracteur pour savoir si deux nombres sont égaux ou pas.
3. Circuit combinatoire réalisant les fonctions logiques de la liste d'opération.
4. Circuit combinatoire custom réalisant une opération spécifique. Ici savoir le nombre de 1 dans l'entrée A.



Etape 1 : Opcode

Fonctionnalité	Opcode
Add/Sub	0000φφ
	0010φφ
	0001φφ
	0011φφ
Comparateur	011001
	011010
	011011
	011100
	011101
	011110
Logique	10φφ00
	10φφ01
	10φφ10
	10φφ11
Custom	11φφφφ

Add/Sub :

- Utilise les 4 premiers bits
(ou seulement les 2 du milieu)
- Donc Add/Sub si les 2 premiers bits sont "00".

Logique :

- Utilise les 2 premiers et les 2 derniers bits
(ou seulement les 2 derniers)
- Donc Logique si les 2 premiers bits sont "10".

Comparateur :

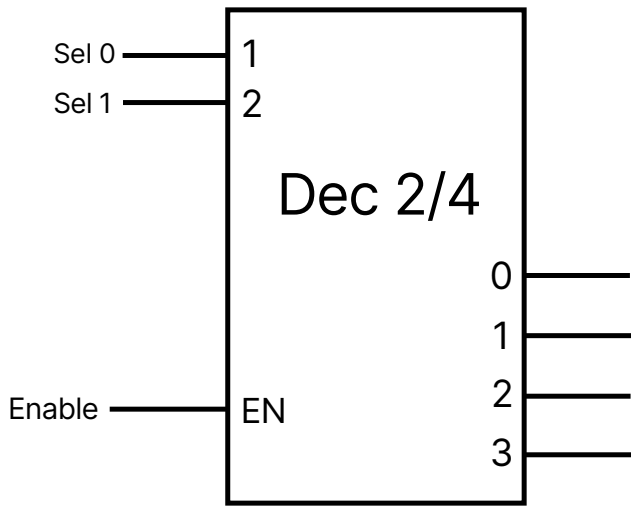
- Utilise les 6 bits
(ou seulement les 3 derniers)
- Donc Comparateur si les 2 premiers bits sont "01".

Custom :

- Utilise les 2 premiers bits
- Donc Custom si les 2 premiers bits sont "11".

Choix de l'opération :

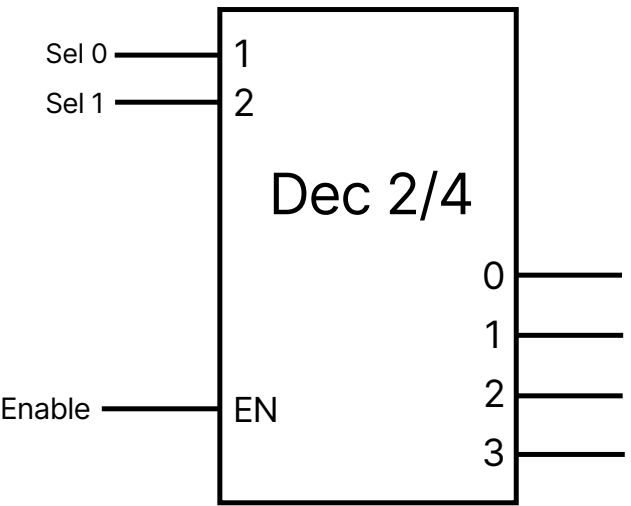
Choisir les bits a trailer avec des portes logiques et des constantes.



Sel 0	Sel 1	opérateur
0	0	Add/Sub
0	1	Comparateur
1	0	Logique
1	1	Custom

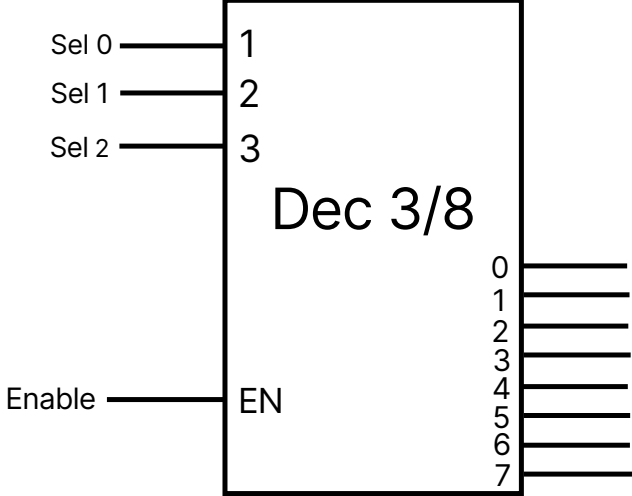
En suite mettre un comparateur après chaque sortie pour choisir l'opération

Add/sub :



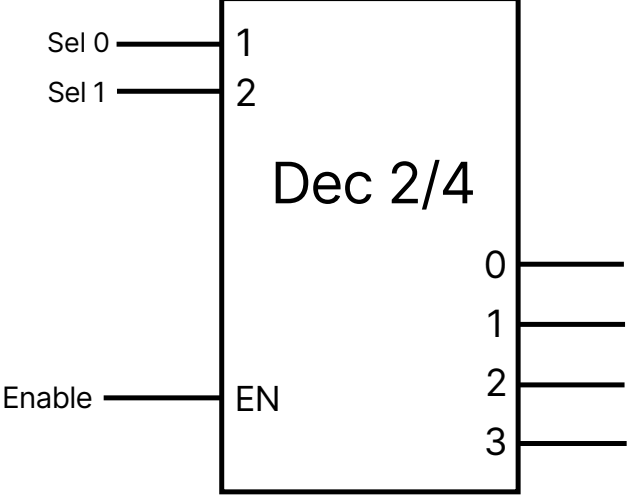
Sel 0	Sel 1	opération
0	0	A + B signé
0	1	A + B non signé
1	0	A - B signé
1	1	A - B non signé

Comparateur :



Sel 0	Sel 1	Sel 2	opération
0	0	0	Error
0	0	1	A ≥ B signé
0	1	0	A < B signé
0	1	1	A ≠ B
1	0	0	A = B
1	0	1	A ≥ B non - signé
1	1	0	A < B non - signé
1	1	1	Error

Logique :



Sel 0	Sel 1	opération
0	0	A and B
0	1	A or B
1	0	A nand B non - signé
1	1	A xor B non - signé

Custom :

une seul possibilité donc pas besoin de comparer.