

Apprentissage – Electronique numérique – TD 2

Conception d'une unité arithmétique et logique 4 bits

On souhaite réaliser une unité arithmétique et logique (ALU) simplifiée pouvant traiter des mots de 4 bits codés en complément à 2. Le schéma bloc est donné ci-dessous. Les parties 1 à 5 vous guideront dans la réalisation des différents blocs fonctionnels nécessaires, la dernière partie étant la synthèse finale de la fonction.

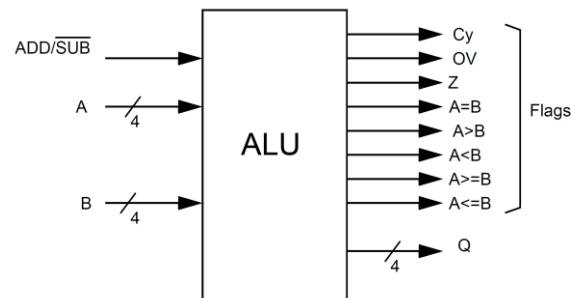


Table de vérité:

$ADD/\overline{SUB} = 1$ réalise $Q = A + B$

$ADD/\overline{SUB} = 0$ réalise $Q = A - B$

OV = overflow (dépassement)

Cy = Carry (retenue)

Z=1 quand Q=0

1. Binaire signé en complément à 2

Soit un nombre sur 4 bits en complément à 2:

- Quelle est la dynamique de codage?
- Etablissez la table de correspondance binaire vers décimal

2. Test d'égalité

Soient 2 nombres P et Q codés en complément à 2 sur 4 bits. Etablissez la table de vérité, les équations logiques et le schéma de deux blocs fonctionnels suivants:

- Le bloc PEQQ qui fournit une sortie $P_EQ_Q = 1$ quand $P=Q$.
- Le bloc PEQ0 qui fournit une sortie $P_EQ_0 = 1$ quand $P=0$.

3. Test P>Q

Soient 2 nombres P et Q codés en complément à 2 sur 4 bits:

Etablissez la table de vérité, les équations logiques et le schéma du bloc fonctionnel suivant PGTQ qui fournit une sortie $P_GT_Q = 1$ quand $P>Q$.

Conseils: lors de l'établissement de la table de vérité, n'inscrivez que les conditions qui réalisent la fonction (sinon, il y aura 256 lignes à votre table!!). Regardez dans quelle mesure la fonction réalisée à l'exercice 2 peut générer des informations utiles pour cette nouvelle fonction. Vous pourrez modifier la solution de l'exercice 2 en conséquence.

4. Autres tests logiques

Soient 2 nombres P et Q codés en complément à 2 sur 4 bits. A l'aide des informations issues des blocs fonctionnels des parties 2 et 3, réalisez:

Un bloc fonctionnel qui fournit une sortie $P_LT_Q = 1$ quand $P < Q$, une sortie $P_GE_Q = 1$ quand $P \geq Q$ et une sortie $P_LE_Q = 1$ quand $P \leq Q$.

5. Additionneur 1 bit avec retenue

Cette fonction ne sera pas réalisée de la façon habituelle mais à l'aide de multiplexeurs.

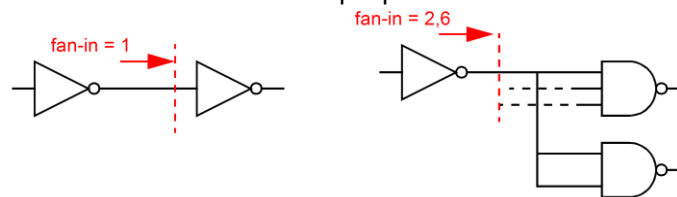
- Combien d'entrées et de sorties doit comporter ce bloc?
- Combien de combinaisons d'entrée faut-il traiter? En déduire le type et le nombre de multiplexeurs à utiliser.
- Donnez le schéma de votre solution (on ne détaillera pas la structure interne des multiplexeurs)

6. Additionneur 1 bit optimisé

- Retrouver le schéma interne du multiplexeur utilisé dans la question précédente. En tenant compte du fait que dans cette application particulière, les entrées du multiplexeur sont reliées à des niveaux logiques fixés, proposer une version minimale du circuit multiplexeur (vous utiliserez des portes INV ainsi que des NAND comportant entre 2 et 4 entrées).
- L'additionneur comporte deux multiplexeurs tels que définis précédemment. Quelles sont les parties communes que l'on peut partager afin de minimiser le nombre de portes? Donnez le schéma final de l'additionneur.

7. Temps de réponse de l'additionneur 1 bit optimisé

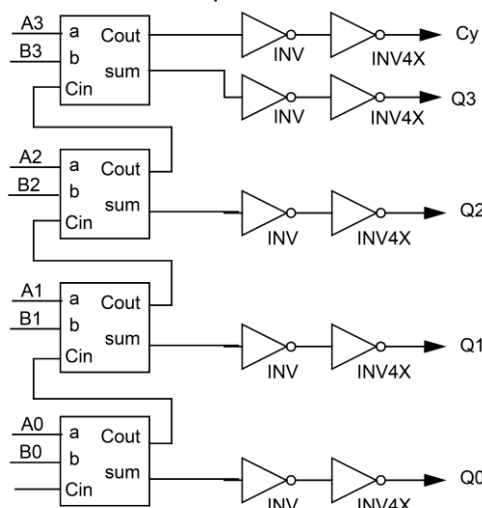
Le temps de propagation t_p d'une porte dépend de la charge qu'elle doit piloter. Cette charge est définie par le paramètre appelé fan-in et dont la valeur est propre à chaque porte. Lorsqu'une sortie attaque plusieurs portes, la charge vue est la somme des fan-in de chaque porte connectée.



Le temps de propagation d'une porte se calcule de la façon suivante: $t_p = t_{p0} + \alpha \sum (fan - in)$. Par exemple, une porte INV qui pilote une porte INV4X aura pour temps de propagation: $t_p = 0,09 + 0,05 \times 3,7 = 0,275$ ns

Fonction	Fan-in	t_{p0} (ns)	α (ns/fan-in)	P_d (μ W/MHz)
INV	1	0,09	0,05	0,5
XOR2	1,6	0,3	0,1	0,6
INV4X	3,7	0,08	0,01	2,8
NAND2	0,85	0,15	0,1	0,6
NAND3	0,9	0,2	0,1	0,6
NAND4	1	0,25	0,1	0,6
NOR2	0,75	0,15	0,1	0,6
NOR3	0,8	0,2	0,15	0,6
AND2	0,9	0,3	0,06	0,9

Soit le circuit additionneur 4 bits ci-dessous dans lequel on utilise les blocs additionneur 1 bit optimisés.



- Déterminer le fan-in total des entrées a, b et Cin pour un additionneur 1 bit optimisé de la question 6b.
- pour les bits 0, 1 et 2 déterminez la charge des sorties Sum et Cout. Déterminer le temps de réponse des blocs additionneur correspondants. Même question pour le bit 3.
- Déterminer le temps de réponse du circuit complet sans charge sur les sorties Q0 à Q3 et Cy.

8. Schéma final

Assemblez les différents blocs étudiés pour réaliser l'ALU. Précisez clairement les interconnexions mais ne détaillez pas la structure interne des blocs.