

Architecture des ordinateurs - TD 10

1 Échauffement : Quine McCluskey

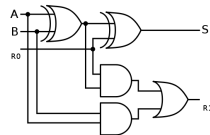
$$f(a, b, c, d) = \sum m(2, 3, 7, 9, 11, 13) + \sum d(1, 10, 15)$$

— Réduire la fonction f par la méthode de Quine McCluskey.

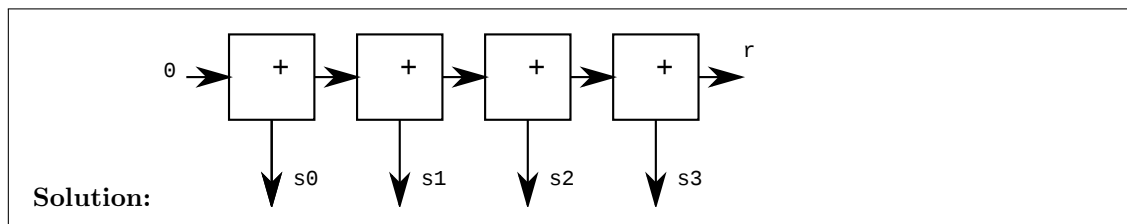
2 Additionneur à Retenue anticipée

2.1 Rappels : Additionneur Série

Le circuit d'un additionneur complet est donné ci-dessous :



1. Rappeler le circuit d'un additionneur 4 bits série :



Le chemin critique dans un circuit électronique représente le chemin le plus long entre les entrées et les sorties. On l'exprimera ici en nombre de portes. Le chemin critique permet de calculer le temps nécessaire pour obtenir un résultat dans un circuit combinatoire.

2. Quelle est la longueur du chemin critique dans ce circuit (en nombre de portes traversées) ?
3. Quelle est la longueur du chemin critique pour un additionneur n bits série ?

2.2 Additionneur à retenue anticipée

Les additionneurs à retenue anticipée permettent d'accélérer le temps de propagation dans un additionneur. Pour cela ils réduisent le nombre de portes nécessaires pour propager la retenue à travers le circuit. On va implémenter un additionneur à retenue anticipée (Carry Lookahead Adder).

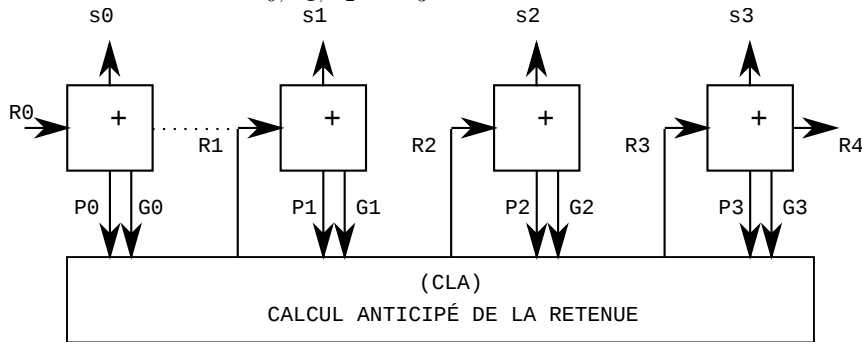
1. Lorsqu'un additionneur complet génère une retenue, deux scénarios sont possibles : l'additionneur génère une retenue lors de la somme ou l'additionneur propage la retenue de l'additionneur précédent. Modifiez le circuit de l'additionneur complet de manière à rajouter deux signaux sortants :
 - P , vrai lorsque l'additionneur propage la retenue précédente.
 - G , vrai lorsque l'additionneur génère une retenue.

Les signaux P et G doivent être calculés uniquement à partir de A et de B .

Solution:

- (a) $P = A + B$
- (b) $G = A.B$

Nous donnons ci-dessous le schéma général d'un additionneur à retenue anticipée. Au lieu de propager la retenue à travers l'ensemble des additionneurs complets, l'unité CLA va directement calculer les retenues R_0, R_1, R_2 et R_3 de manière à réduire la taille du chemin critique.



2. Exprimez R_1 en fonction de P_0, G_0 et R_0 ?

Solution:

$$R_1 = G_0 + P_0.R_0$$

3. Exprimez R_2 en fonction de P_1, G_1, P_0, G_0 et R_0 .

Solution:

$$R_2 = G_1 + P_1.R_1 = G_1 + P_1.(G_0 + P_0.R_0) = G_1 + P_1.G_0 + P_0.P_1.R_0$$

4. Montrez par récurrence que

$$R_i = G_i + \sum_{j=0}^{i-1} (G_j \cdot \prod_{k=j+1}^i P_k) + R_0 \cdot \prod_{j=0}^i P_j$$

Solution: Cas de base $i = 1$: $R_1 = G_0 + R_0.P_0$ vérifie l'équation.

Récurrence, supposons l'hypothèse vraie pour i :

$$R_i = G_i + \sum_{j=0}^{i-1} (G_j \cdot \prod_{k=j+1}^i P_k) + R_0 \cdot \prod_{j=0}^i P_j$$

$$R_{i+1} = G_{i+1} + P_{i+1}.R_i$$

$$R_{i+1} = G_{i+1} + P_{i+1} \cdot (G_i + \sum_{j=0}^{i-1} (G_j \cdot \prod_{k=j+1}^i P_k) + R_0 \cdot \prod_{j=0}^i P_j)$$

$$R_{i+1} = G_{i+1} + \sum_{j=0}^{i-1} (G_j \cdot \prod_{k=j+1}^{i+1} P_k) + R_0 \cdot \prod_{j=0}^{i+1} P_j$$

5. En utilisant les équations précédentes, proposez un circuit pour le CLA.
6. Quelle est la longueur du chemin critique pour l'additionneur à retenue anticipée ?

Solution:

7. Bonus : Peut-on utiliser la même méthode pour implémenter un additionneur 16 bits avec CLA ? 64 bits ? Quels sont les inconvénients ? Comment feriez vous pour implémenter un additionneur 16 bits avec CLA.