

MASTER 1, Cours ANUM
Pr. Habib MEHREZ
Janvier 2007

DURÉE 1H Tous documents autorisés

EXERCICE I:

On se propose de réaliser une architecture permettant le calcul de la fonction $1/X^2$ utilisant la méthode de Newton.

QUESTION 1:

Proposer une fonction permettant de résoudre le problème. En déduire une fonction dont le calcul de la racine permettrait de répondre aux objectifs. Nous supposons que l'architecture disposera comme seules ressources arithmétiques, un multiplieur et un additionneur utilisant la virgule flottante.

Déterminer la suite résultante par l'application de la méthode de Newton et déterminer les conditions de convergence.

QUESTION 2:

Proposer une architecture matérielle mettant en oeuvre la méthode de Newton et permettant une convergence rapide vers la solution.

EXERCICE II:

Une application spécifique a généré une forme particulière de sommande à additionner en utilisant la technique "carry save" et une deuxième utilisant les arbres de Wallace.

La forme particulière à sommer sera la suivante:

```
*
**
****
*****
****
**
*
```

Chaque étoile représente un élément binaire, les lignes correspondent aux sommandes à additionner et chaque colonne représente un poids binaire.

QUESTION 1:

Proposer un réseau de Wallace permettant de calculer la somme totale en utilisant des additionneurs complets ou des demi-additionneurs. *La dernière étape ne doit pas mettre*

en oeuvre un additionneur rapide, et par conséquent on utilisera un additionneur à propagation de retenues.

REMARQUE:

On apportera la réponse sous la forme suivante appliquée à l'exemple donné:

Exemple: soit à additionner la forme suivante:

```

  ***
*****
  ***
*****
  ****

```

On dispose de 5 colonnes.

1ère étape: On calcule le nombre d'éléments binaires correspondants à chaque colonne et on dispose les résultats en ligne, soit:

2 3 5 5 5 (càd 2 bits première colonne, 3 bits 2ème colonne etc.)

2ème étape: la première couche logique d'additionneurs complets (F) et de demi-additionneurs (H) sera structurée de la façon suivante:

```

H F F F F
  H H H

```

Elle va générer les deux lignes suivantes:

```

  1 1 2 2 2   (nombre de bits de sommes pour chaque colonne)
1 1 2 2 2   (nombre de bits de retenues)

```

3ème étape:

Ressources:

```

H F H H H
  H H

```

Éléments générés ou non connectés:

```

1 1 1 2 2 1
1 1 2 2 1

```

4ème étape:

Ressources:

H H F H F
H

.....et ainsi de suite;

(Fin remarque)

En fin d'analyse, donner le nombre total de ressources utilisées ainsi que le nombre de couches traversées en termes de demi-additionneurs.

QUESTION 2:

Proposer une architecture de type carry save (utilisant une propagation diagonale des retenues) pour calculer la somme totale. On remarquera que les deux premières lignes ne nécessitent pas de ressources. Donner le nombre total de ressources utilisées ainsi que le nombre de couches traversées en termes de demi-additionneurs.

QUESTION 3:

Comparer les deux solutions en fonction de la complexité et des performances. Dire si les arbres de Wallace présentent un intérêt ici ou pas. Justifier votre réponse.

EXERCICE : ARCHITECTURES D'UN INCREMENTEUR

On rappelle qu'incrémenter un nombre entier revient à lui rajouter 1.

QUESTION 1: (ARCHITECTURE A PROPAGATION DE RETENUE D'UN INCREMENTEUR)

a) Etablir les équations des bits de somme (S_i) et de retenue (R_i), ainsi que les bits de propagation (P_i) et de génération (G_i) correspondant à chaque poids binaire. En déduire les particularités simplificatrices d'un incrémenteur par rapport à un additionneur.

b) Proposer une architecture parallèle simple de type "ripple" (propagation de la retenue) et donner sa complexité et ses performances en fonction du nombre de bits du nombre à incrémenter.

QUESTION 2: (ARCHITECTURE A SAUT DE RETENUE D'UN INCREMENTEUR)

Toujours en utilisant les particularités simplificatrices d'un incrémenteur par rapport à un additionneur, étudier une architecture d'incrémenteur basée sur le principe du saut de retenue et illustrée pour un nombre de taille $N=16$ bits et de blocs de taille $k=4$ bits.

On s'inspirera utilement de l'architecture d'un additionneur de ce type étudiée en cours. Donner sa complexité et ses performances en fonction du nombre de bits du nombre à incrémenter.

MASTER 1, Cours ANUM

Janvier 2008

DURÉE 1H Tous documents autorisés

ARCHITECTURES DE MULTIPLIEURS AU CARRE

On se propose de concevoir un multiplieur au carré dont l'entrée est donc unique (par exemple X) et la sortie est X^2 (soit $X \cdot X$). Pour simplifier, on considérera que X est un entier non signé.

A) Algorithme de la multiplication.

Il se trouve que cette multiplication est particulière puisque le multiplicande est le même que le multiplicateur. De ce fait, montrer, en reprenant l'illustration de l'algorithme "addition-décalage", que des simplifications importantes vont apparaître. L'illustration se fera sur l'exemple d'un nombre X codé sur 4 bits soient x_0, x_1, x_2 et x_3 .

Indication: Ecrire les sommandes sous forme de produits partiels $x_i x_j$

B) Architecture parallèle.

En tenant compte des simplifications déduites de la question précédente, proposer une architecture parallèle pour ce type de multiplieur avec des additionneurs complets utilisant des "propagations latérales" des retenues. Illustrer ce multiplieur pour le cas où les nombres X seraient codés sur $N=4$ bits. Donner alors la progression des performances ainsi que la complexité de l'architecture en fonction de N

C) Architecture plus rapide

On souhaite améliorer les performances et la complexité de l'architecture précédente. Plusieurs solutions peuvent être envisagées. On vous propose de retenir la solution avec une propagation diagonale de la retenue. Illustrer cette solution pour le cas où $N=4$ bits.

Donner alors la progression des performances ainsi que la complexité de la nouvelle architecture en fonction de N .