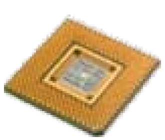
Master ACSI

Architectures pour le Traitement Numérique

Cours Habib MEHREZ

Laboratoire d'Informatique de Paris 6, Université Pierre et Marie Curie

4, place Jussieu, 75252 Paris cedex 5



DIVISION

$D=qd+r$

$D-(2^{n-1}q_{n-1} + \ldots\ldots+2q_1d+q_0d)=r$

Pour tout i, on considère les $q_i=1$.

On commence par $q_{n-1}=1$

Si $D-2^{n-1}q_{n-1}\,d >=0$ alors $q_{n-1}=1$

sinon $q_{n-1}=0$

Et on réitère..

Division restaurante

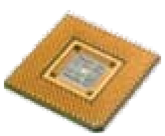
- Exemple: 7/3

Divident	Diviseur
111	11
Reste	Quotient

Retenue sortante

```
                 1 1 1 0
               - 1 1 0 0
1 <=             0 0 1 0
          -        1 1 0
0. <=            1 1 0 0
          +        1 1 0
                 0 0 1 0 0            Restauration
          -          1 1 0
0 <=             1 1 1 1 0
          +          1 1 0
                 0 0 1 0 0 0          Restauration
          -              1 0 0
1 <=             0 0 0 0 1 0 0
          -                1 1 0
0 <=             1 1 1 1 1 1 0
          +                1 1 0
                 0 0 0 0 1 0 0        Reste
```

Temps	$O(n^2)$
Surface	$O(n^2)$



Division non restaurante

- Exemple: 7/3

Au lieu de restaurer, la fois d'après faire + à la place de -

$$(D-2^{n-1}\,d)\ _+\ 2^{n-1}\,d - 2^{n-2}\,d = (D-2^{n-1}\,d) + 2^{n-2}\,d$$

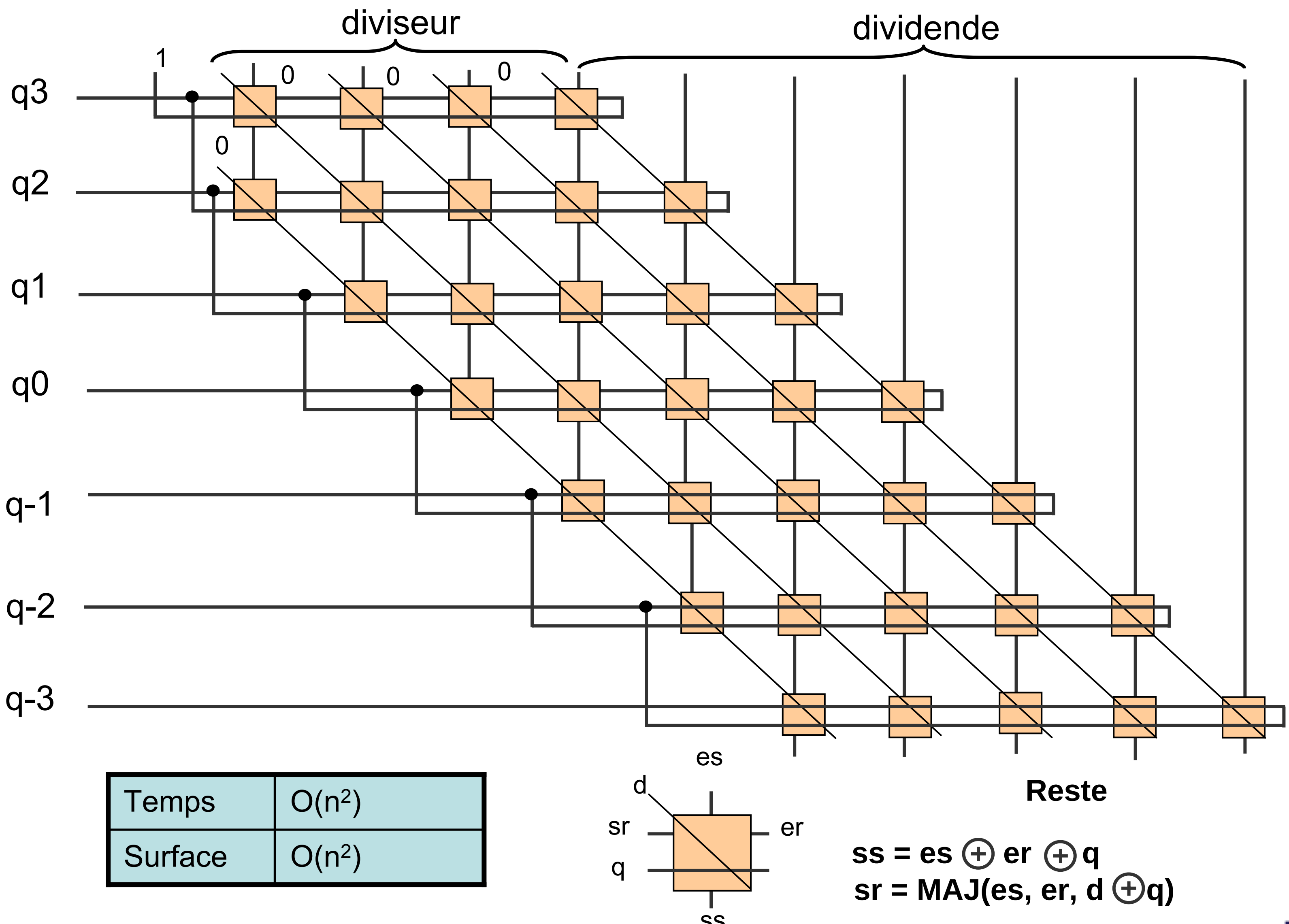
$$A = 2^{n-2},\ 2A = 2^{n-1} \Rightarrow (D-2A.d) + 2A.d - A\,d = (D-2A\,d) + A\,d$$

Retenue sortante

Retenue sortante		
		1 1 1 0
		- 1 1 0 0
1 <=		0 0 1 0
	-	1 1 0
0. <=		1 1 0 0 0
	+	1 1 0
0 <=		1 1 1 1 0 0
	+	1 1 0
1 <=		0 0 0 0 1 0 0
	-	1 1 0
0 <=		1 1 1 1 1 1 0 0
	+	1 1 0
		0 0 0 0. 0 0 1 0

Reste

Division NON restaurante: implémentation



Temps	O(n^2)
Surface	O(n^2)

ss = es ⊕ er ⊕ q

sr = MAJ(es, er, d ⊕q)

Division de NEWTON-RAPHSON

- **Principe**

La division de Newton-Raphson est une division itérative utilisant le principe suivant:

$$C = A/B = A * 1/B$$

- **Méthode de Newton-Raphson**

La méthode de Newton-Raphson permet de résoudre les équations de type:

$$F(x) = 0$$

en effectuant l'itération de l'équation suivante:

$$X_{i+1} = X_i - F(X_i)/F'(X_i)$$

On démontre que si la valeur initiale X_0 est proche de la racine simple de F, alors X_n converge quadratiquement vers cette racine.

Division de NEWTON-RAPHSON

- **Application**

Nous considérons l'équation suivante:

$$F(x) = (1/X) - B$$

L'inverse de B peut donc être trouvé par la méthode de Newton-Raphson en calculant le résultat de l'équation F(x)=0.

$$\begin{aligned} X_{i+1} &= X_i - F(X_i)/ F'(X_i) \\ &= X_i - (1/X_i - B)/ -(X_i)^{-2} \\ &= X_i (2 - B*X_i) \end{aligned}$$

Afin que le résultat de l'équation converge vers l'inverse de B, la valeur initiale $\mathbf{X_0}$ doit être comprise entre:

$$0 < X_i < 2/B \quad \text{si } B>0$$
$$2/B < X_i < 0 \quad \text{si } B<0$$

Division de NEWTON-RAPHSON

Exemple (1/7.25)

Pour le calcul de l'inverse de 7.25 la valeur initiale $\mathbf{X_0}$ doit être comprise entre:

$$0 < X_0 < 2/7.25$$

$$0 < X_0 < 0.275862$$

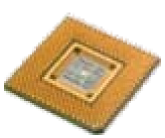
Nous prendrons donc comme valeur initiale $X_0 = 0.1$

Itétation 1: $X_1 = X_0 – (2 – B*X_0)$
$= 0.1(2 – 7.25*0.1)$
$= 0.1275$

Itétation 2: $X_2 = X_1 – (2 – B*X_1)$
$= 0.1275(2 – 7.25*0.1275)$
$= 0.1371421875$

Itétation 3: $X_3 = X_2 – (2 – B*X_2)$
$= 0. 1371421875(2 – 7.25*1371421875)$
$= 0.1379265230$

La valeur exacte de 1/7.25 est égale à: 0.1379310345

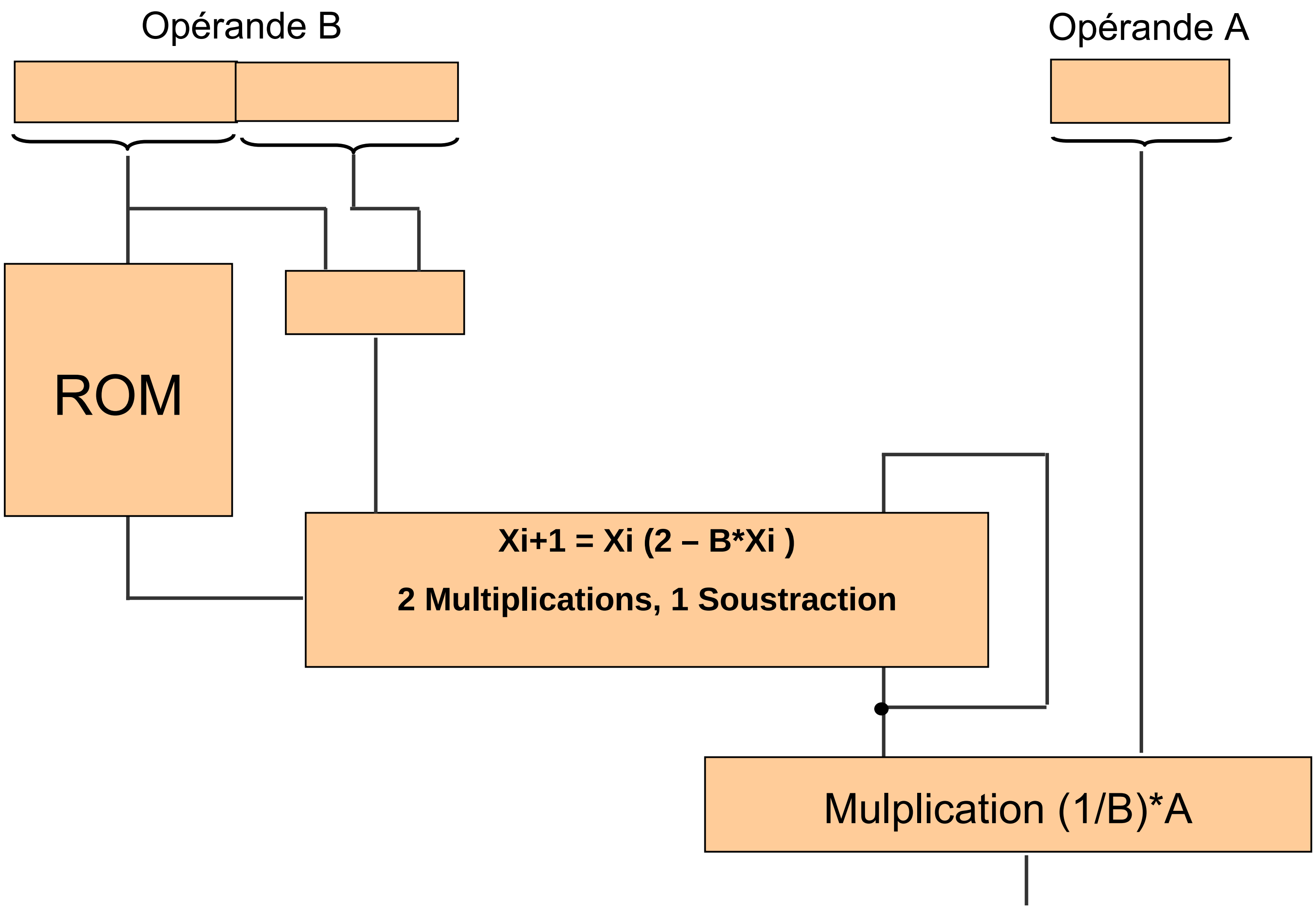


Division de NEWTON-RAPHSON: implémentation

En pratique et pour une convergence rapide, on construit une table des inverses de tous les nombres codés sur un nombre réduit de bits (soient p bits).

Cette table sera inscrite à l'intérieur d'une ROM. La méthode de Newton-Raphson permet donc de calculer l'inverse d'un nombre de façon quadratique avec une précision de plus en plus importante selon le nombre d'itérations effectuées. (2^i p bits avec i nombre d'itération).

Division de NEWTON-RAPHSON: implémentation



Conclusions

Division	Temps	Surface
Division restaurante	$O(n^2)$	$O(n^2)$
Division non restaurante	$O(n^2)$	$O(n^2)$
Division Hamacher et Williams	$O(n)$	$O(n^2)$

La méthode de Newton-Raphson nécessite un certain nombre d’intération dépendant de la précision à atteindre.

Elle peut utiliser les ressources existances plus une ROM spécifique. Elle reste néanmoins très pénalisante en terms de performances par rapport à une solution cablée.

RACINE CARREE

Racine carrée non restaurante

- Exemple: racine carrée de 25 : 011001

Carry sortante

MSB ===➔ **Q0=1 <=**

$$\begin{array}{rcc} & 0 & 1 \\ - & 0 & 1 \\ \hline & 0 & 0 \end{array}$$

Q1=0 <=

$$\begin{array}{rcccc} & 0 & 0 & 1 & 0 \\ - & 0 & 1 & 0 & 1 \\ \hline & 1 & 1 & 0 & 1 \end{array}$$

=====0.Q0.nQ0.1

Q2=1 <=

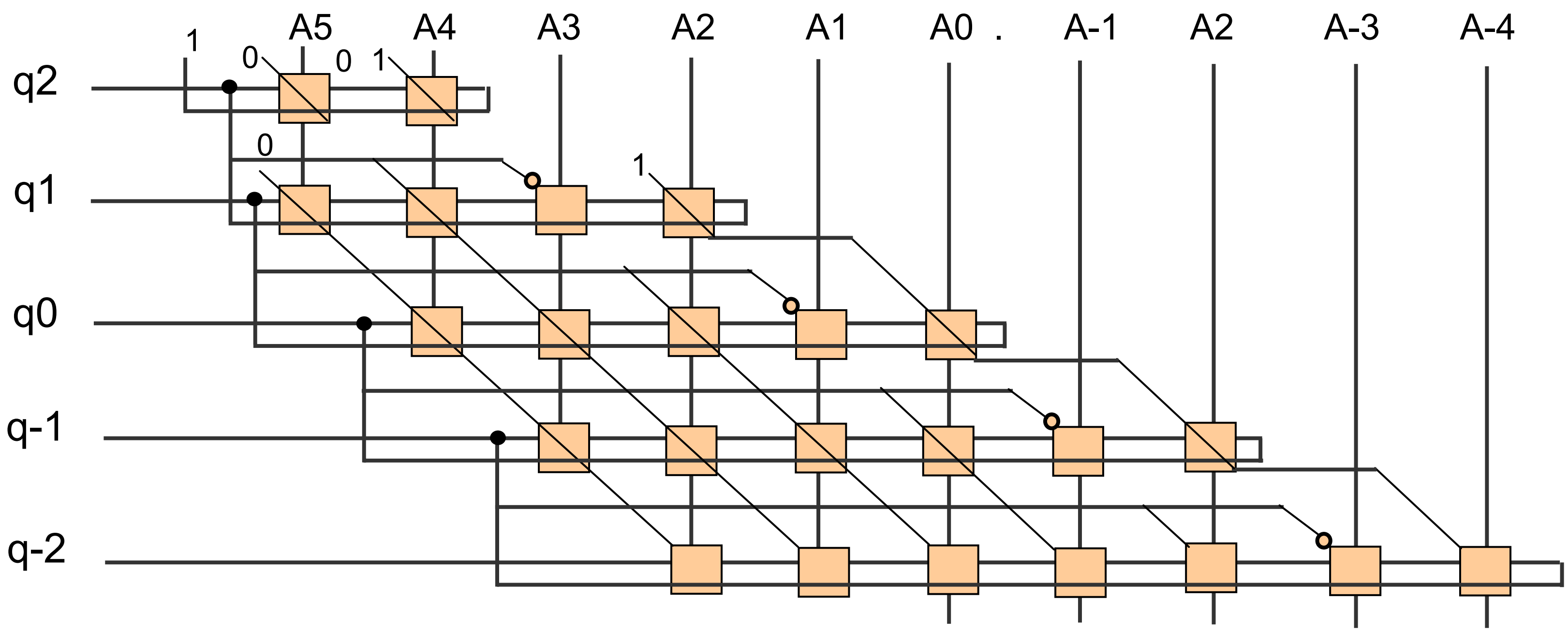
$$\begin{array}{rccccc} & 1 & 0 & 1 & 0 & 1 \\ + & 0 & 1 & 0 & 1 & 1 \\ \hline & 0 & 0 & 0 & 0 & 0 \end{array}$$

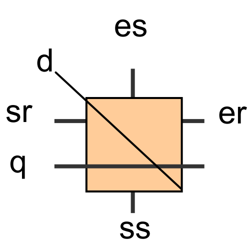
==== 0.Q0.Q1.nQ1.1

sqrt(25)=Q0.Q1.Q2=1.0.1=5

Temps	$O(n^2)$
Surface	$O(n^2)$

Racine carrée non restaurante : implémentation





$$ss = es \oplus er \oplus q$$
$$sr = MAJ(es, er, d \oplus q)$$

Temps	$O(n^2)$
Surface	$O(n^2)$

Racine carrée de NEWTON-RAPHSON

- **Principe**

  La racine carrée de Newton-Raphson est une racine carrée itérative utilisant le principe suivant:

  C=Sqrt(A) => C=A*(1/Sqart(A))

- **Méthode de Newton-Raphson**

  La méthode de Newton-Raphson permet de résoudre les équations de type:

  $$F(x) = 0$$

  en effectuant l'itération de l'équation suivante:

  $$X_{i+1} = X_i - F(X_i)/ F'(X_i)$$

  On démontre que si la valeur initiale X_0 est proche de la racine simple de F, alors X_n converge quadratiquement vers cette racine.

Racine carrée de NEWTON-RAPHSON

- **Application**

Nous considérons l'équation suivante:

$$F(x) = (1/X^2) - A$$

L'inverse de la racine carrée de A peut donc être trouvé par la méthode de Newton-Raphson en calculant le résultat de l'équation F(x)=0.

$$\begin{aligned} X_{i+1} &= X_i - F(X_i)/F'(X_i) \\ &= X_i - (1/X_i^2 - A)/(-2X_i^{-3}) \\ &= X_i (3 - A*X_i^2)/2 \end{aligned}$$

Afin que le résultat de l'équation converge vers l'inverse de A, la valeur initiale $\mathbf{X_0}$ doit être comprise entre:

$$0 < X_i < \mathrm{Sqrt}\,(3/A) \quad \text{si } A > 0$$

Racine carrée de NEWTON-RAPHSON: implémentation

En pratique et pour une convergence rapide, on construit une table des inverse de la racine carrée de tous les nombre codés sur un nombre réduit de bits (soient p bits).

Cette table sera inscrite à l'intérieur d'une ROM. La méthode de Newton-Raphson permet donc de calculer l'inverse de la racine carrée d'un nombre de façon quadratique avec une précision de plus en plus importante selon le nombre d'itérations effectuées. (2^i p bits avec i nombre d'itération).

Racine carrée de NEWTON-RAPHSON: implémentation