

# Opérateurs arithmétiques matériels

## Métriques du matériel

Florent de Dinechin

18 octobre 2005

Rappels de techno VLSI

Quatre cibles matérielles

Trois métriques

Deux philosophies architecturales

# Rappels de techno VLSI

Rappels de techno VLSI

Quatre cibles matérielles

Trois métriques

Deux philosophies architecturales

## Le spécialiste est parti

- Je ne connais que l'électronique numérique
- Je ne connais (et encore) que le CMOS

# Dessignons un transistor

# Chimico-photo-lithographie

# Quatre cibles matérielles

Rappels de techno VLSI

Quatre cibles matérielles

Trois métriques

Deux philosophies architecturales

LCDE :

- puissance (vitesse/compacité)
- flexibilité

sont antagonistes avec

- facilité de mise en œuvre
- coût

Choisissez donc parmi les différentes technos ci-dessous :

- VLSI full-custom
- VLSI précaractérisé
- VLSI en cellules standard
- FPGAs

- Atome de calcul : le transistor
- Fil élémentaire : arbitraire sur n'importe quel niveau
- Vous découpez vos transistors comme vous voulez (plus ou moins large, plus ou moins long)
- Vous faites les fils ou vous voulez
- Vous devez tout placer et les router vous-même.

Bien sûr vous vous aidez d'outils, mais alors (LCDE) vous perdez en flexibilité.

Enfin, vous devez faire fondre le circuit par une usine spécialisée.

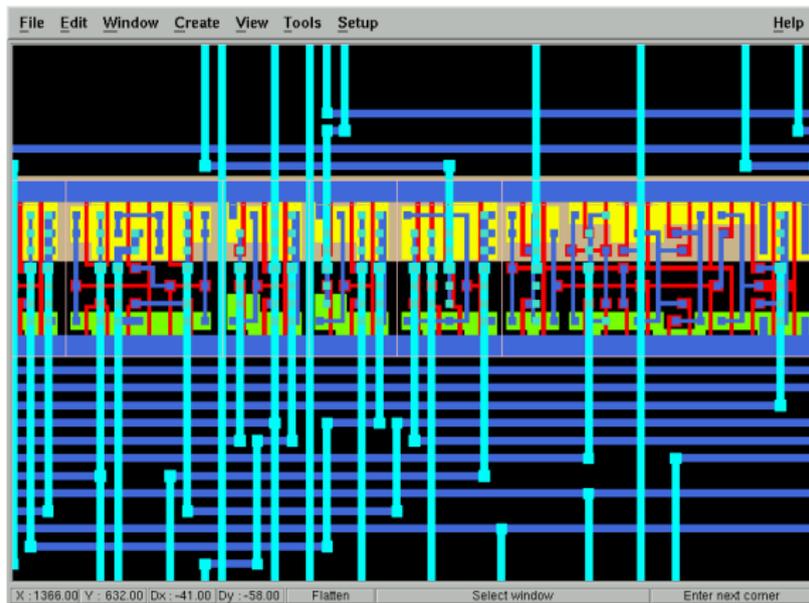
## Circuits précaractérisés

- Vous partez d'un circuit "pas fini" avec plein de portes universelles, mais pas de fils
- Vos atomes de calcul sont ces portes universelles
- Vous devez "mapper" votre circuit sur ces portes, et tirer les fils vous-même (router).
- Le tout en passant encore par une fonderie

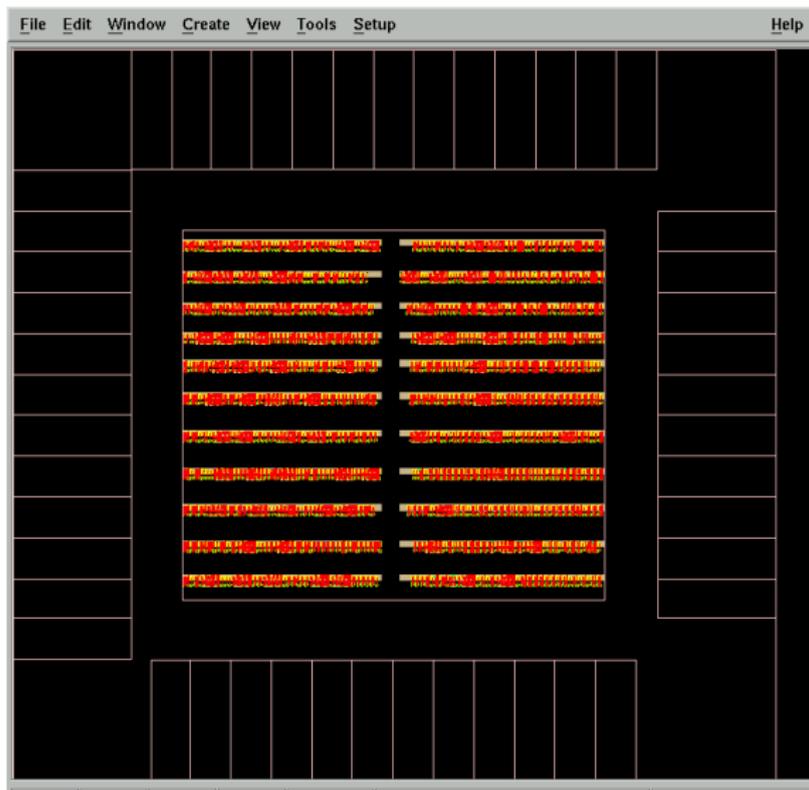
C'est la seule techno dont je ne parlerai pas du tout car je n'y connais rien.

- Vous disposez d'une bibliothèque plus ou moins détaillées de cellules de base, qui peuvent être aussi grosses que des *full adder*.
- Vous devez
  - “mapper” votre circuit sur ces cellules,
  - les placer,
  - les router.mais on utilise des outils sans remords.
- Ici aussi, passage par la fonderie

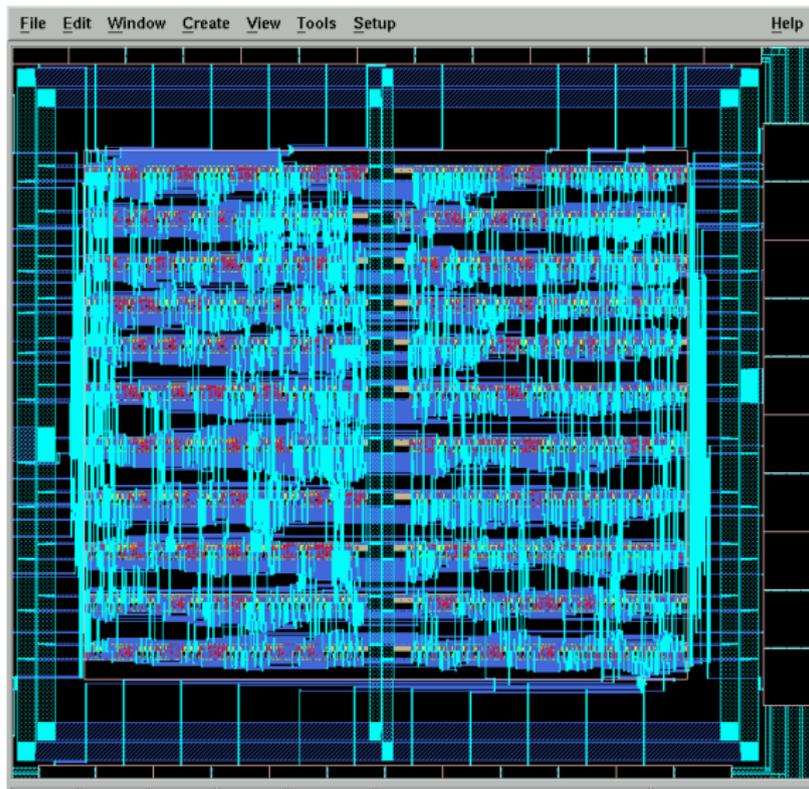
# Jolis dessins



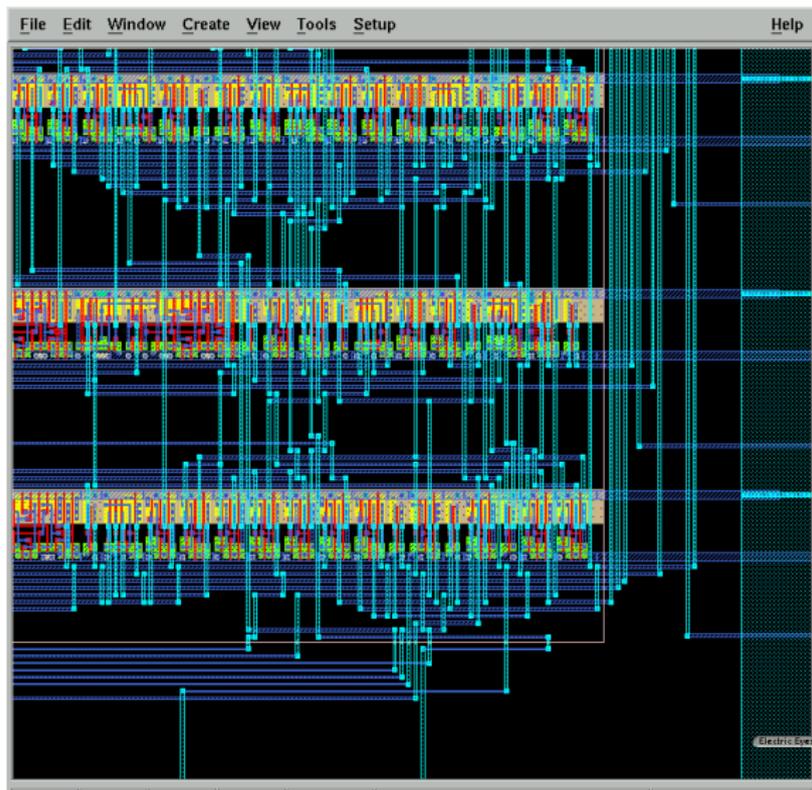
# Jolis dessins



# Jolis dessins



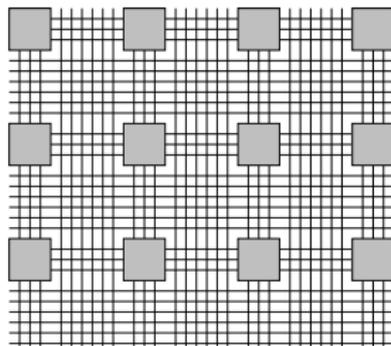
# Jolis dessins



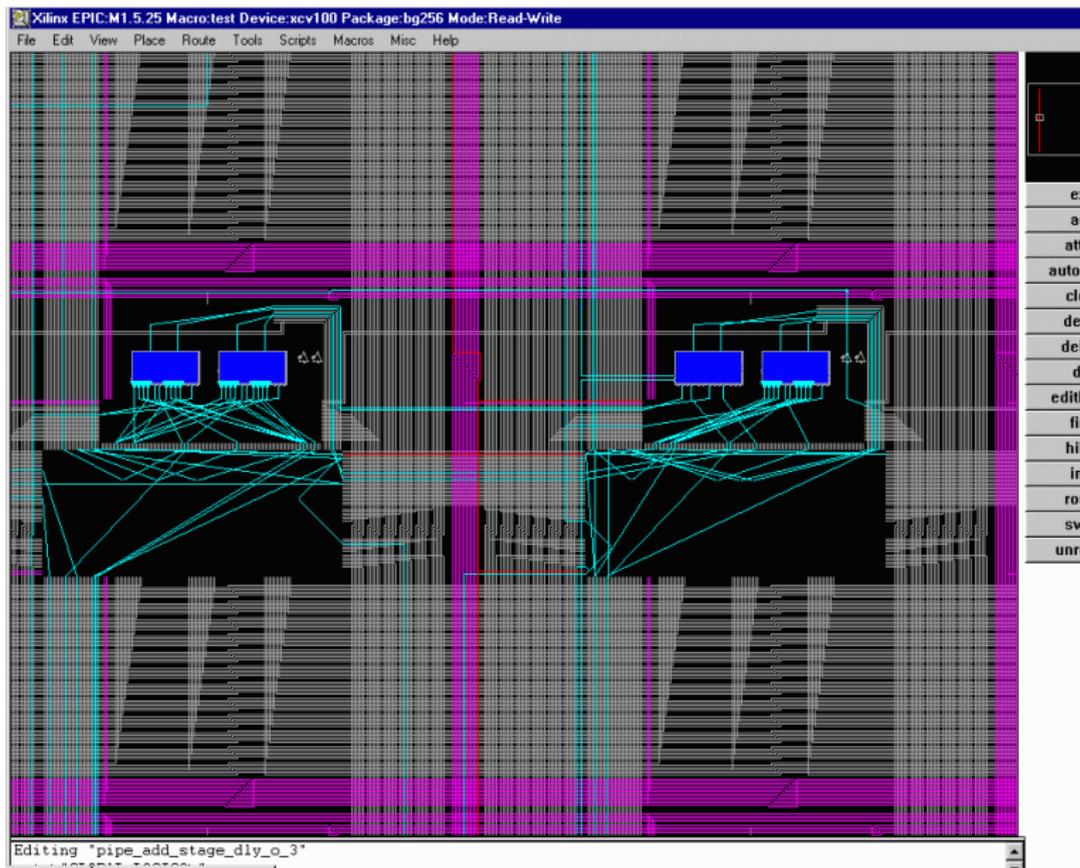
(Field-Programmable Gate Array)

C'est le seul qui ne passe pas par la fonderie

- Vous achetez une puce FPGA : pas besoin de fonderie
- C'est une grille d'atomes de calculs, plutôt rapides (dernier cri de la techno)
- sur lesquels vous devez "mapper" votre application.
- Il y a une infrastructure de routage programmable entre ces atomes
- Il est super lent : comme le routage est programmable, il y a des transistors plein les fils...



# Vue logique d'un vrai FPGA



# Trois métriques

Rappels de techno VLSI

Quatre cibles matérielles

Trois métriques

Deux philosophies architecturales

# Une combinatoire de LCDE

- Vitesse
- Surface
- Consommation

## Vitesse, surface et capacité

- Un transistor est (en gros) une source de courant constante
  - un gros transistor peut fournir plus de courant
- Faire basculer un transistor c'est (en gros) charger sa grille, qui est (en gros) un condensateur.
- Donc la vitesse d'un transistor dépend
  - de sa géométrie (taille), qui fixe sa capacité
  - mais aussi du courant qui alimente la grille.
  - *La vitesse d'un transistor n'a de sens que dans un circuit.*
- Mais ce transistor doit lui-même alimenter d'autres transistors, et sa capacité à le faire dépend de sa surface.
- Donc plus un transistor est gros
  - plus peut charger vite d'autre transistors,
  - mais plus il faut de temps pour le charger lui-même.
- Par ailleurs, un transistor répartit son courant de sortie entre les transistors qu'il alimente...
- À la surprise générale, on sait écrire des équations simples pour gérer tout cela (*logical effort*).

## Et les fils

- Ce sont aussi (en gros) des capacités à charger
- Leur capacité est (en gros) proportionnelle à leur longueur
- On les fait le plus fins possible (on parle des fils qui portent des valeurs logiques, pas des fils d'alimentation)

## En général, quand la techno s'améliore

- La vitesse des transistors s'améliore de manière prédictible
- La vitesse des fils s'améliore, mais moins : elle devient de plus en plus prépondérante.
- On peut faire des circuits avec des fils (relativement) plus longs, et on les fait (loi de Rent).
- On échappe à la loi de Rent en ajoutant des niveaux de métal.

# La surface

Relisez les transparents précédents, on a déjà tout dit...

Le prix (en brozoufs) d'un circuit, dans une technologie donnée, est proportionnel (en gros) au carré de sa surface.  
Pourquoi ? à cause des petits défauts.

C'est  $Ri^2$ .

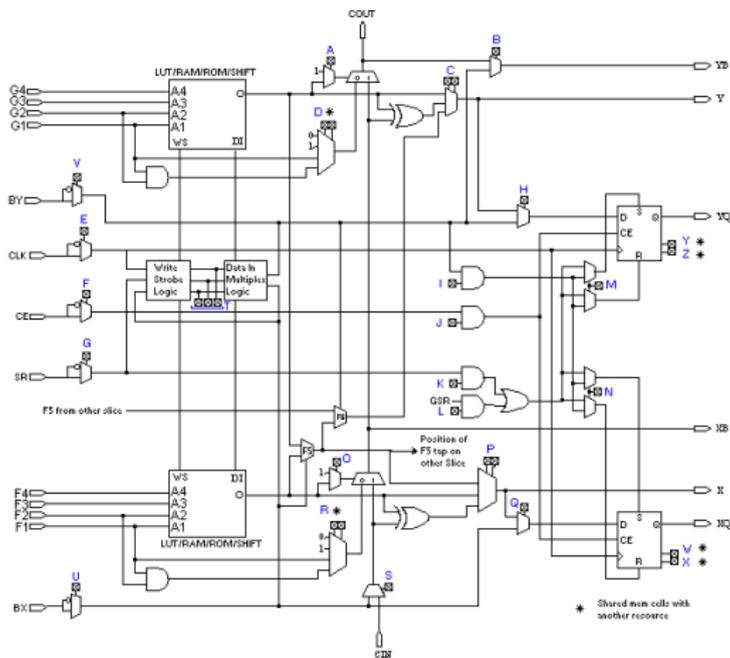
- proportionnelle à la fréquence
- proportionnelle à la surface macroscopique
- proportionnelle au coefficient d'activité (algorithmique)
- microscopiquement, y en a dans les fils et dans les transistors, mais c'est surtout les transistors.

## Et maintenant : métriques FPGA

- vitesse des cellules constante (et pas ridicule)
- vitesse des fils lente
- surface **binaire** (le circuit rentre, ou pas) tant pour les cellules que pour les fils
- cellules universelles mais bricolées pour accélérer certaines opérations courantes...

# Aides à l'addition et à la multiplication dans les FPGA

FPGAs du fabricant Xilinx



# Deux philosophies architecturales

Rappels de techno VLSI

Quatre cibles matérielles

Trois métriques

Deux philosophies architecturales

# Circuits synchrones

- Une horloge contrôle tout
- Philosophie du temps au pire cas
- Avantages
  - Tout le monde bascule en même temps : simplicité conceptuelle
  - Méthodologies et outils (exemple : retiming/pipeline)
- Inconvénients
  - Tout le monde bascule en même temps : inefficacité (because temps de calcul au pire cas)
  - Distribution de l'arbre d'horloge difficile (25% de la surface d'un processeur récent)

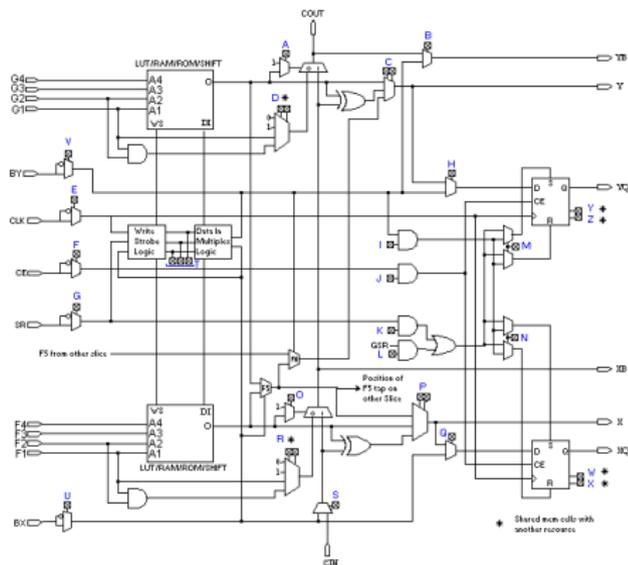
# Circuits asynchrones

- Pas d'horloge. Les échanges de données sont négociés localement.
- Philosophie du temps moyen
- Avantages
  - Efficacité, notamment énergétique
  - Émissions électromagnétiques réduites
  - Pas d'horloge à construire
- Inconvénients
  - Gestion du parallélisme difficile (protocoles)
  - Surcoût matériel (plusieurs fils pour transmettre un bit)

# ASIC vs FPGA

Les FPGA sont faits pour le synchrone :

- De bons arbres d'horloges (ils ne coûtent rien)
- Plein de registres adaptés



## Ce qu'il reste à comparer

Les technos pour les mémoires  
(quand on en aura besoin)