

Estimation et réduction de l'énergie en conception CMOS numérique

Olivier SENTIEYS

ENSSAT - Université de Rennes 1

LASTI - Groupe Signal-Architecture

6 Rue de Kérampont - F-22300 LANNION

sentieys@enssat.fr

<http://archi.enssat.fr>



Plan de l'exposé



Pourquoi s'occuper de la puissance ?

- Evolution de la technologie
- Evolution des applications

Où part la puissance ?

- Puissance dynamique
- Chemin de données, Mémoire, Horloge ...

Comment diminuer la consommation ?

- Réduire Vdd, Réduire la capacité effective
- Transformer l'algorithme
- Optimiser l'horloge, Optimiser la mémoire

Comment estimer la consommation ?

- Estimation aux niveaux transistor ou logique
- Estimation de haut niveau

Conclusions

Quel est le grand challenge ?



- **La puissance des circuits augmente progressivement malgré :**
 - Diminution de la tension d'alimentation Vdd
5V -> 3V -> 2.5V -> ...
 - Amélioration des méthodes de conception des circuits
- **Les effets indésirables sont nombreux**
 - Diminution des performances et de la fiabilité
 - Augmentation du coût
mise en boîtier, énergie, environnement, immeuble, ...
- **Limitations dues à la puissance**
 - Limites physiques
 - Limites dépendantes de l'application
Ordinateurs portables (PDA, Pagers, Desktop), nomades (PAN)
Systèmes embarqués ou ultra basse-consommation
Commutateurs télécom

Watt is the big deal ?

3

25 ans d'évolutions



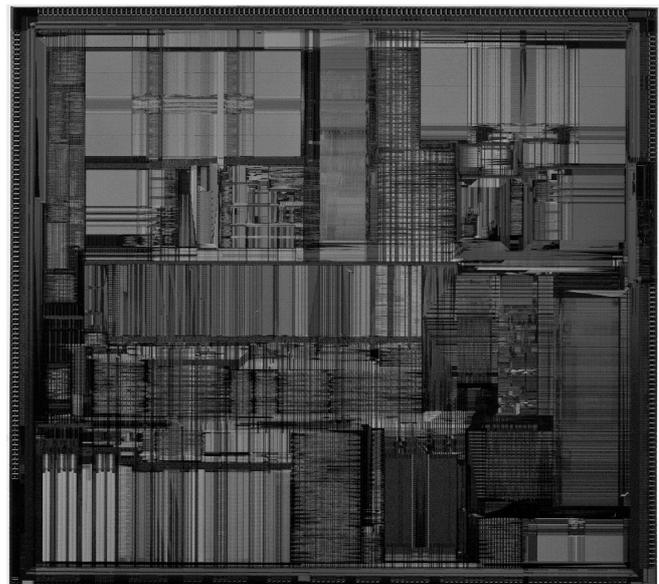
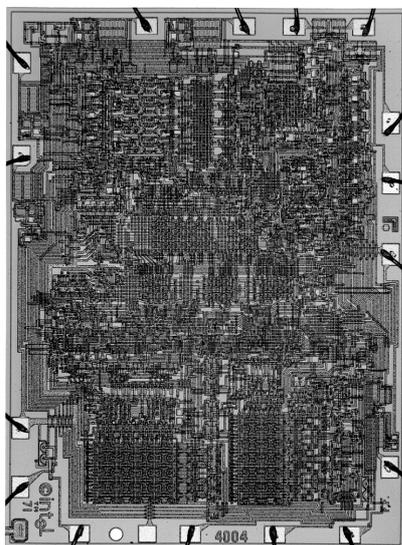
INTEL 4004 (1971)

Données sur 4 bits

2300 transistors, 10 microns

0,06 MOPS, 108 kHz

What is Watt ?



INTEL Pentium II (1996)

Données sur 32 bits

5.5M de transistors, 0.35 μ , 2 cm²

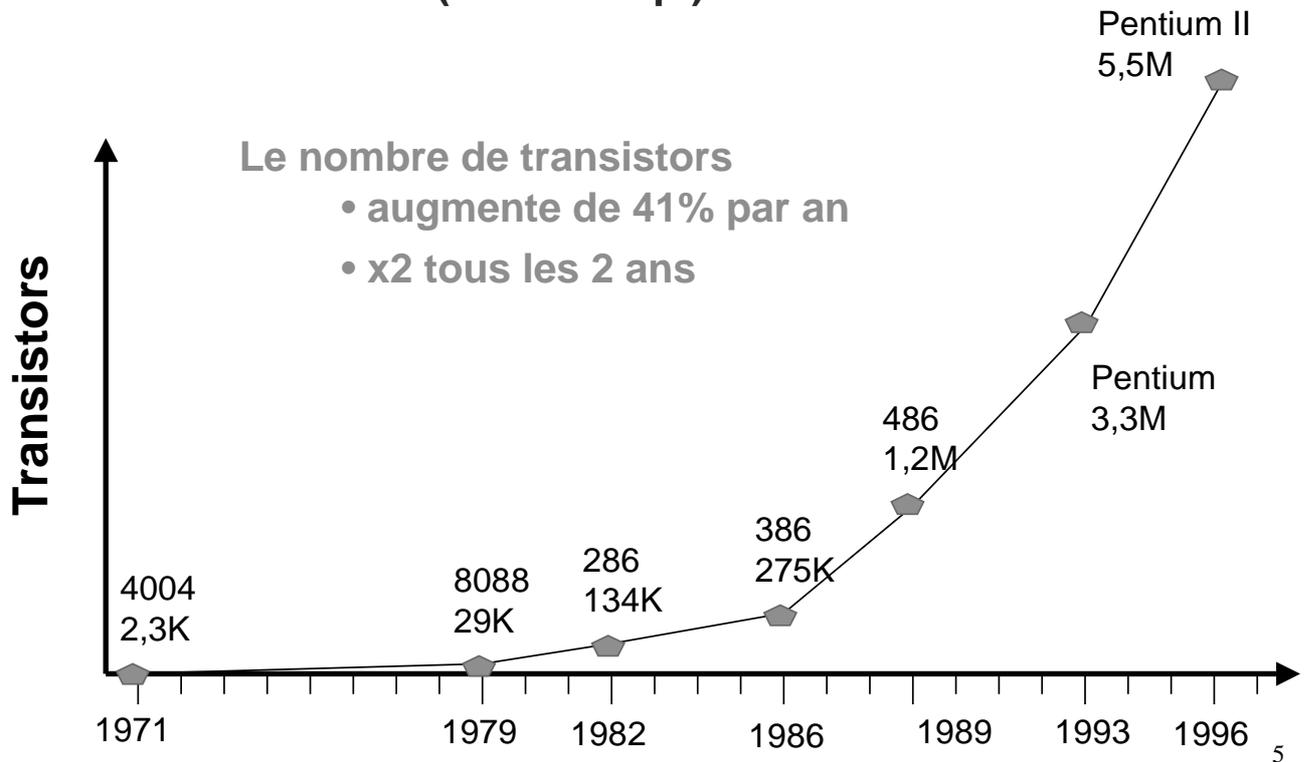
200 MHz, 200 MOPS, 3.3V, 35W

4

Nombre de transistors



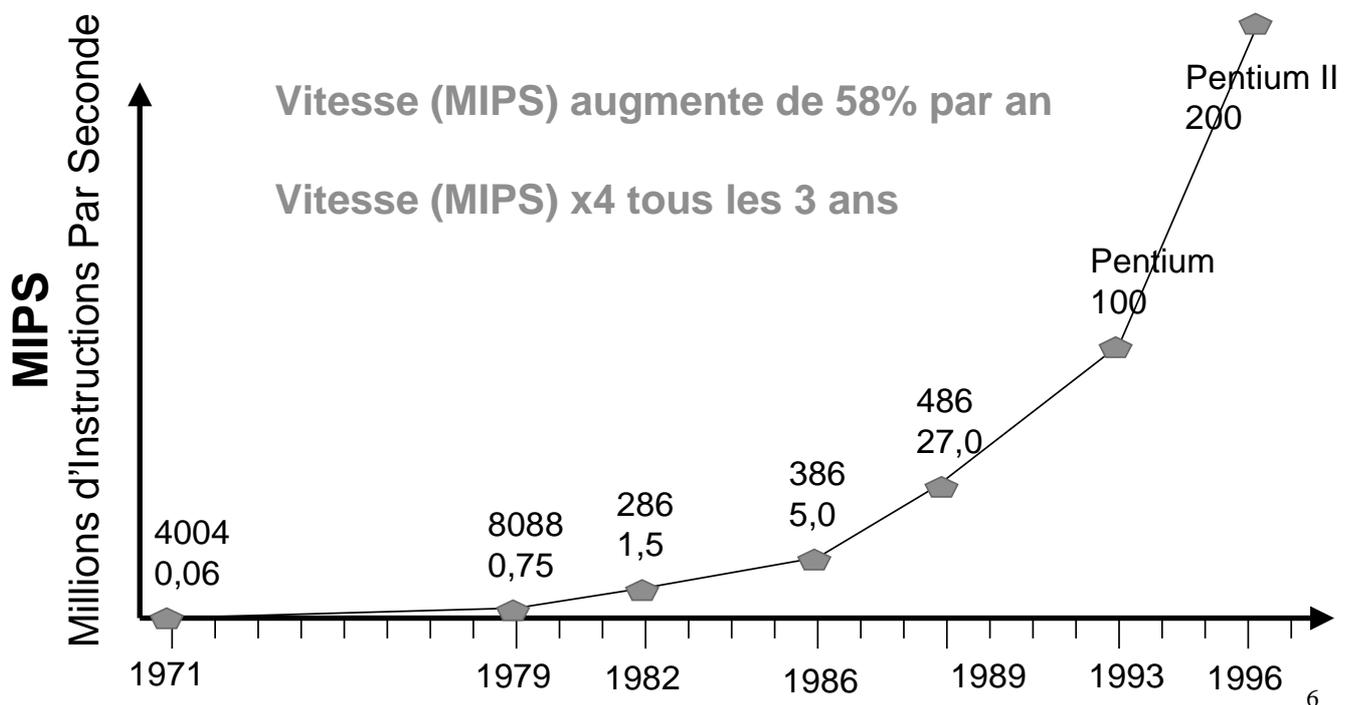
- Loi de G. Moore (INTEL corp.)



Évolution de la vitesse



- Loi de G. Moore (INTEL corp.)



Perspectives : RoadMap SIA



- *0.35 μm en 1995, 0.25 μm en 1997*
- **0.18 μm en 1999**
 - 6 niveaux de métal
 - 1.5-2 Volts, 100 Watts
 - 500 MHz (ASIC) - 1250 MHz (μP)
 - 4 (μP , DRAM) - 8 (ASIC) cm^2
 - Mémoire : 1Gbits, 270 bits/ cm^2 , 10 centimes/Mbits
 - 20 (μP) - 100 (ASIC) MTr

Année d'introduction	1997	1999	2001	2003	2006	2009	2012
Technologie (μm)	0,25	0,18	0,15	0,13	0,1	0,07	0,05
Nb de niveaux de métal	6	6-7	7	7	7-8	8-9	9
Tension d'alimentation (V)	1,8-2,5	1,5-1,8	1,2-1,5	1,2-1,5	0,9-1,2	0,6-0,9	0,5-0,6
Puissance sous radiateur (W)	70	90	110	130	160	170	175
Fréquence de fonctionnement (MHz)							
Horloge locale On-chip μP	750	1250	1500	2100	3500	6000	10000
Horloge locale On-chip ASIC	300	500	600	700	900	1200	1500

[RoadMap Silicon Industry Association]

7

Perspectives : RoadMap SIA



Année d'introduction	1997	1999	2001	2003	2006	2009	2012
Technologie (μm)	0,25	0,18	0,15	0,13	0,1	0,07	0,05
Taille du circuit (mm^2)							
DRAM	280	400	450	560	790	1120	1580
Microprocesseur	300	340	385	430	520	620	750
ASIC	480	800	850	900	1000	1100	1300
Mémoire							
Taille	256M	1G	*	4G	16G	64G	256G
Bits/ cm^2	96M	270M	380M	770M	2,2B	6,1B	17B
Coût/MBit (Centimes)	20	10	5	2,5	0,9	0,32	0,11
Microprocesseur							
Logique Tr/ cm^2	3,7M	6,2M	10M	18M	39M	84M	180M
Coût/MTr (Centimes)	500	290	166	97	42	18	8
ASIC							
Logique Tr/ cm^2	8M	14M	16M	24M	40M	64M	100M
Coût/MTr (Centimes)	8,3	4,1	3,3	2,5	1,6	0,8	0,4

8

Surface puce : 11 cm²
 Alimentation : 0.6 V
 Technologie : 0.07 μm

	Densité Gbits/cm ²	Temps Accès ns
DRAM	8,5	10
DRAM (Logic)	2,5	10
SRAM (Cache)	0,3	1,5

15 fois plus dense

2.5 fois plus dense

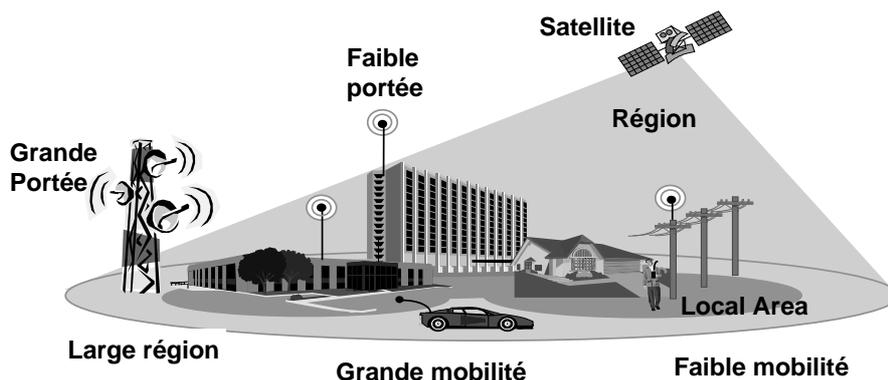
5 fois plus rapide

	Densité Mgates/cm ²	Max Puiss. Moy W/cm ²	Horloge GHz
Custom	25	54	3
Std. Cell	10	27	1,5
Gate Array	5	18	1
Single-Mask GA	2,5	12,5	0,7
FPGA	0,4	4,5	0,25

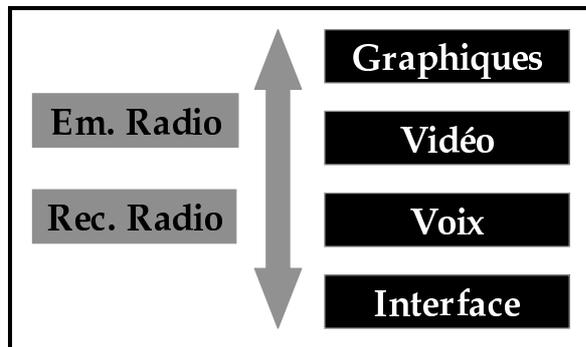
Évolution des applications

- Télécommunications mobiles de 3^{ème} génération

- Fonctions multimédia
- Mobilité
- Grand public



Terminal 3G



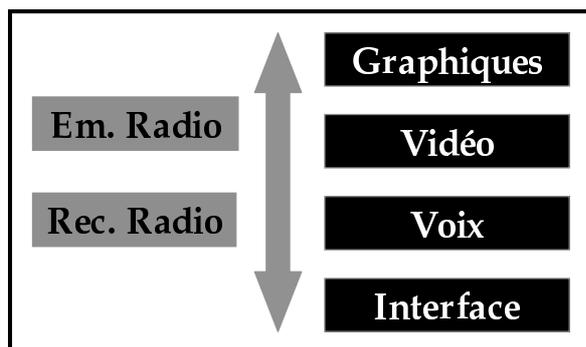
- **Multimédia**
 - Visioconférence
- **Interfaces**
 - Reconnaissance vocale
 - Stylo inertiel
 - Cryptage, authentification
- **Mobilité**
 - UMTS (remplaçant du GSM)
 - Internet Protocol
 - Connexion réseaux locaux

2005 : mobilité et durée de vie des batteries d'un agenda, capacité multimédia d'un PC



11

Terminal 3G Portable



- **Traitement**
 - 6-10 Milliards d'Instructions / sec
- **Autonomie : 10h**
- **Poids : 500g (batteries)**

➔ **500mW @ 6 GIPS**
12 GIPS/W @ 6 GIPS

- **Avec les processeurs actuels**
 - 30 Kg ou 10 minutes !!!



12

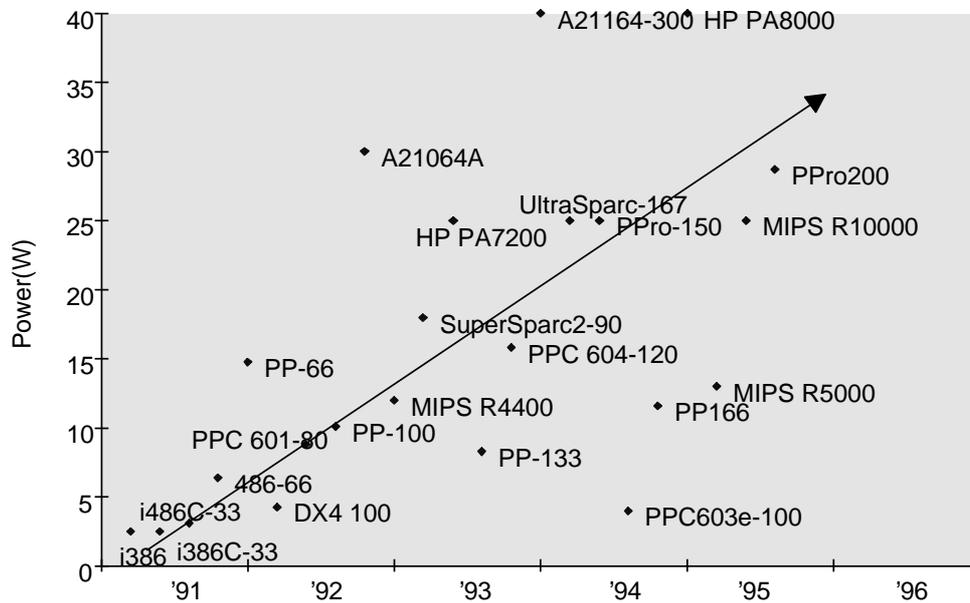
Pourquoi s'occuper de la puissance ?



Dissipation de chaleur

- Evolution de la consommation des microprocesseurs

– x4 tous les 3 ans



[Source: Microprocessor Report]

13

Pourquoi s'occuper de la puissance ?



Dissipation de chaleur

Processeur	Horloge (MHz)	Technologie (µm)	Alimentation (V)	Consommation maximale (W)
Intel Pentium	66	0.8	5.0	16
Intel P6	200	0.35	3.3	35
DEC Alpha 21064	200	0.75	3.3	30
DEC Alpha 21164	300	0.5	3.3	50
PowerPC 620	133	0.5	3.3	30
MIPS R10000	200	0.5	3.3	30
UltraSparc	167	0.45	3.3	30
PowerPC 603	80	0.5	3.3	2.2
IBM 486SLC2	66	0.8	3.3	1.8
MIPS R4200	80	0.64	3.3	1.8
Hitachi SH7032	-	-	-	0.13

- Prévisions de la puissance des µP

$$P = k \cdot \text{surface} \cdot Fclk \text{ avec } k = 0.063 \text{ W/cm}^2 \cdot \text{Mhz}$$

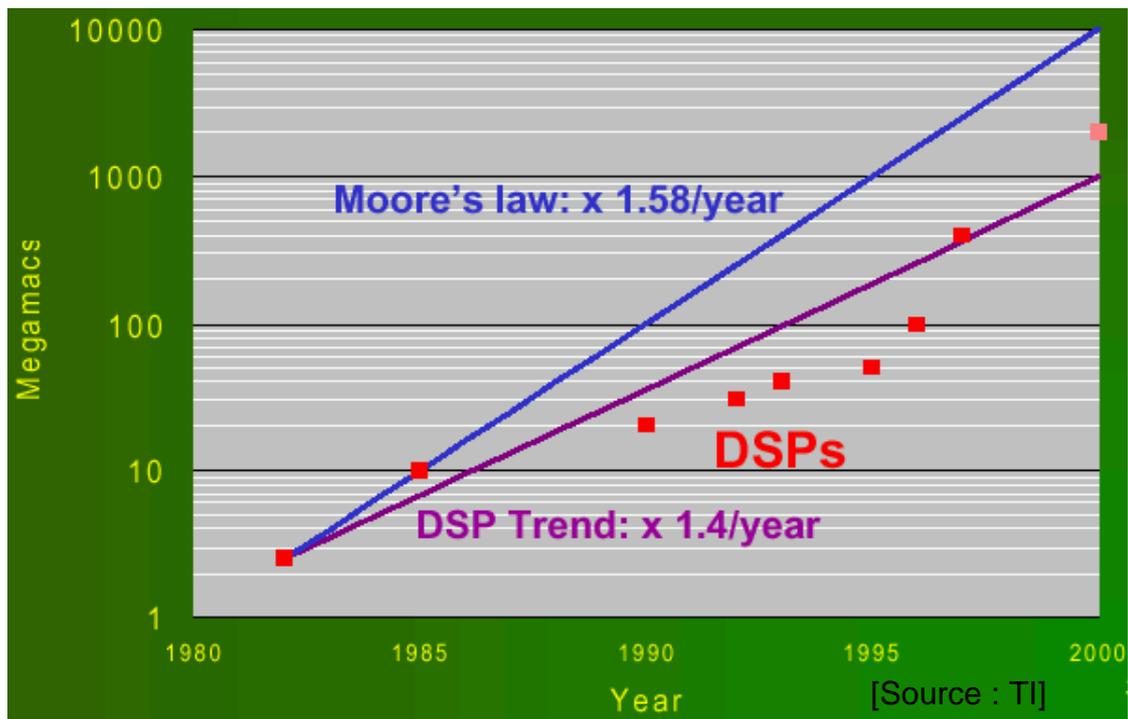
DEC 21164 (3 cm², 300MHz) : 50 Watts

Prochaine génération (10 cm², 500MHz) : 315 Watts !!

14

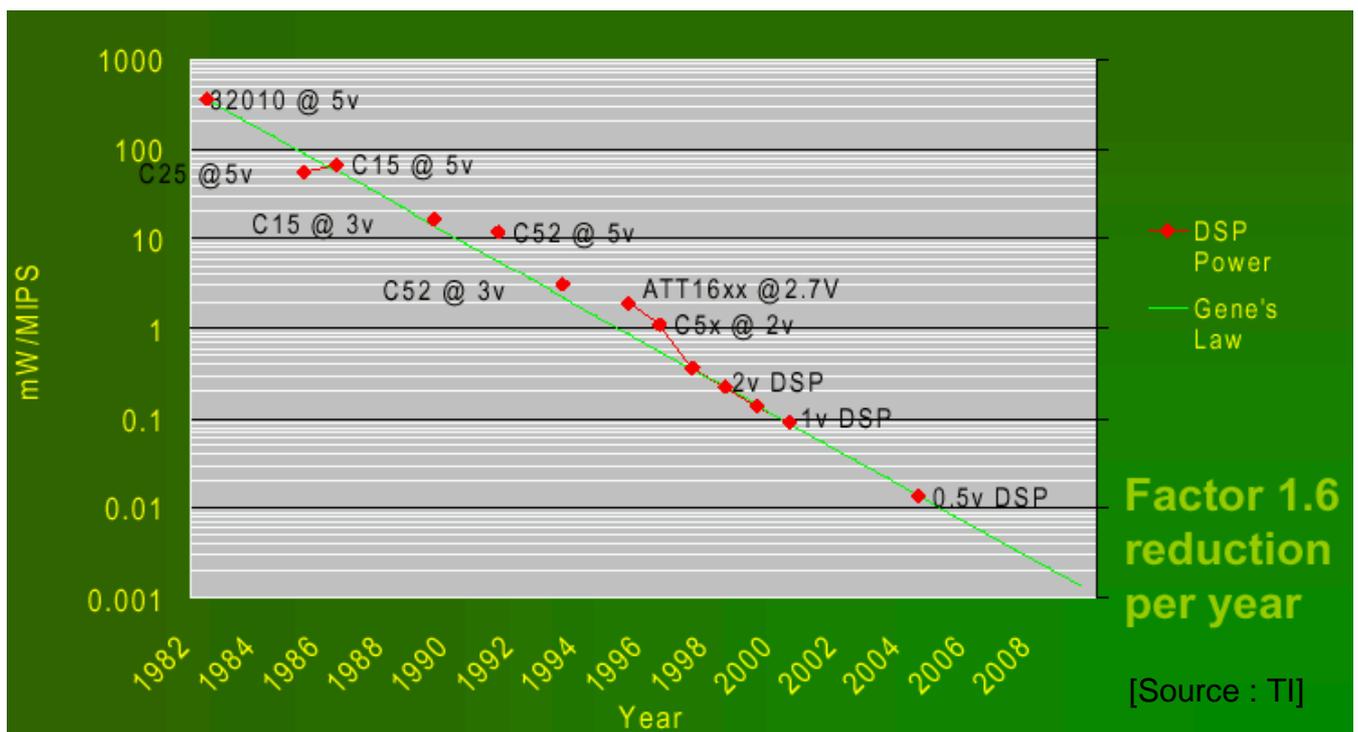
Performances des DSPs

- En retard sur la loi de Moore



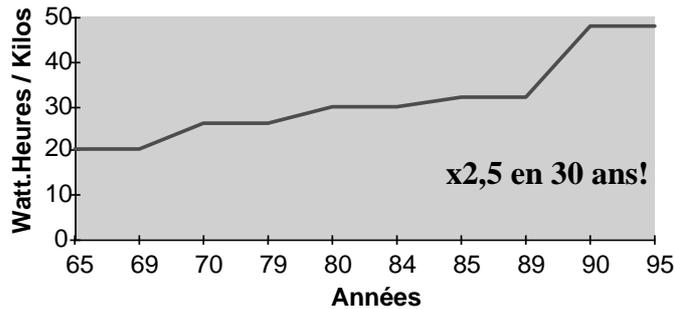
Consommation des DSPs

- Réduction d'un facteur 1,6 par an



Portabilité

- **Nouvelles technologies de batteries (NiMH, Li)**
 - Amélioration limitée dans les prochaines années
 - Pour le Lithium, la tension diminue au cours du temps, ce qui implique de nouvelles contraintes de conception pour les CI
- **Lithium-Ion : 220 Watt-heures/kg**



Métriques pour les processeurs

- **Puissance = utile dans le cas de μP en activité continue**
- **Energie = plus représentative si il existe des modes d'attente**
 - CPO : Clock Per Operation
Nombre de périodes d'horloge par opération (ou instruction)
 - Throughput = $f * 10^{-6} / CPO$ (en MOPS)
 - $MOPS / Watt = 10^{-6} / CPO / C_{eff} / V_{dd}^2$
 - Produit Energie . Temps est une mesure intéressante car dépendant de f : $MOPS^2 / Watt$

$\mu CTRL$	f MHz	CPO	MIPS	Vdd	Puissance	MIPS/Watt	MIPS ² /Watt
8051	20	15	1.35	3V	27 mW	50	67
68HC11	4	4	1	3V	8 mW	125	125
Nordic $\mu RISC$	16	1.5	10.7	3V	10 mW	1100	11500
Atmel AT90	4	1.5	2.67	3V	4.4 mW	600	1620
CoolRisc 81	14	1	14	3V	2.8 mW	5000	70.000
TMS C50	40	1	40	5V	460 mW	87	3500
TMS C50	24	1	24	3V	100 mW	240	5760

Processeurs



- Attention, il faut comparer des μP de fréquence ou de performance équivalente

Processeur	MIPS	Vdd	Pmoy	MIPS/Watt
CoolRisc	14	3V	2.8 mW	5000
TMS C54x	30-200	1.8V	460 mW	1500-3000
TMS C6x	1600	2.5V	2W	800
Dec α	500	1.8-3V	50 W	7-10

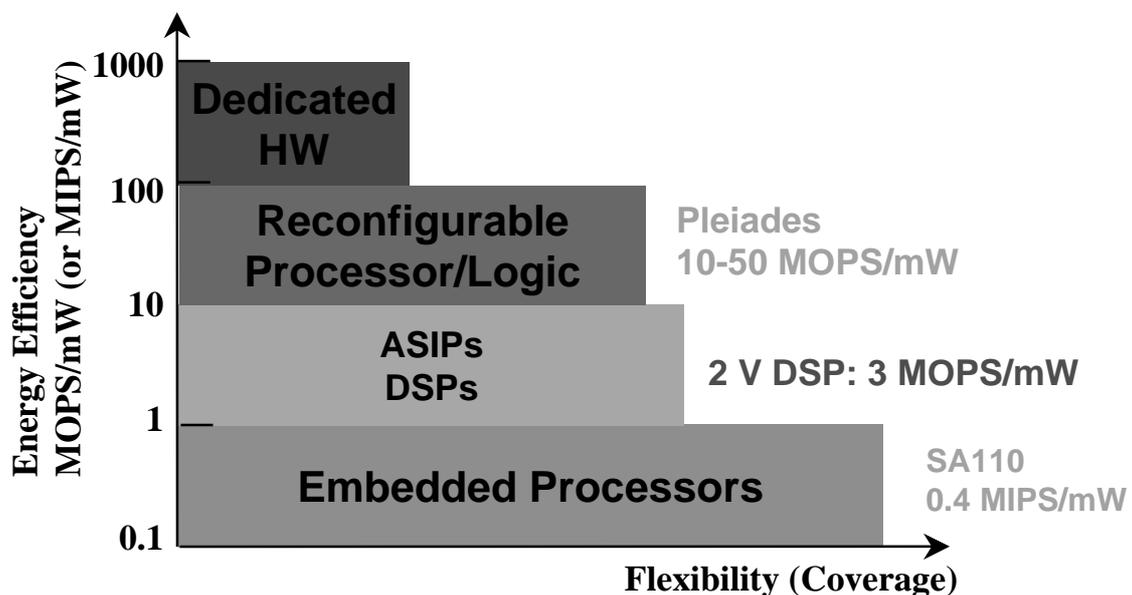
- Si on a besoin de 1200 MIPS, il faut 6 C54x ou 86 CoolRisc !

19

Solutions architecturales

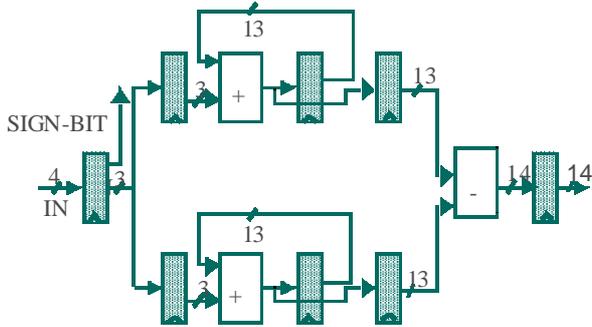


- Flexibilité vs performances



20

Example: Correlator for CDMA Radio:



[After Rabaey]

Energy/Flexibility Tradeoff's

Arm 6 core (5V, 20 MHz)
2765 nJ 167697 fJsec

Xilinx 4003 (5V, 64 MHz)
394 nJ 394 fJsec

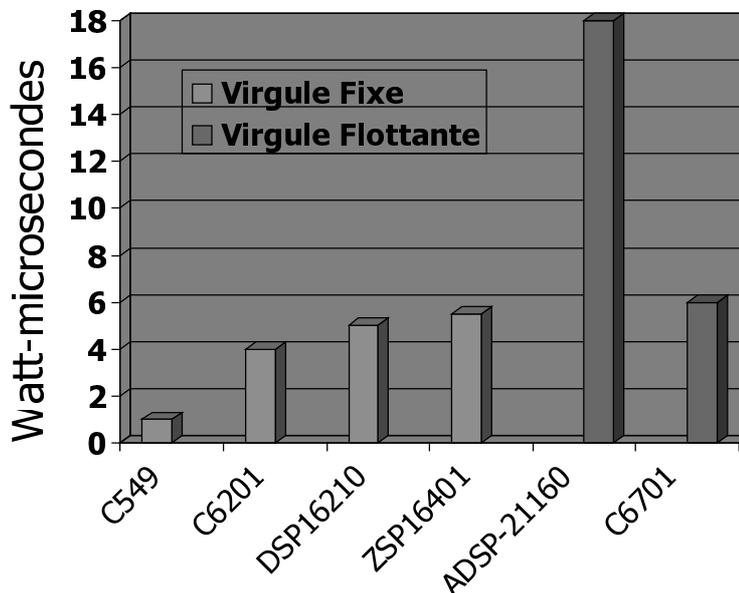
ASIC Datapath (1.5V, 64 MHz)
1.2 nJ 1.04 fJsec

* Energy/symbol

* Normalized Energy-Delay Product (5V)

Consommation

- Filtrage numérique RIF

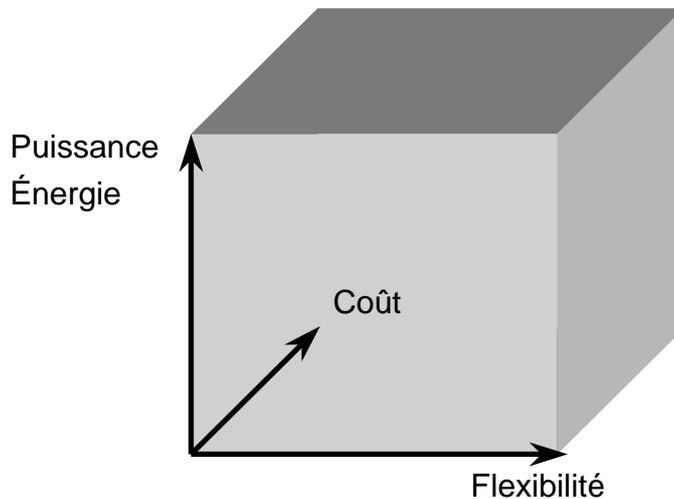


- **C549**
– 100 MHz, 2.5V
- **C6201**
– 200 MHz, 1.8V
- **DSP16210**
– 100 MHz, 3.3V
- **ZSP16401**
– 200 MHz, 2.5V
- **ADSP-21160**
– 100 MHz, 2.5V
- **C6701**
– 167 MHz, 1.8V

Conclusions



- **De nouvelles métriques pour la conception**
 - Puissance et/ou énergie sont les contraintes dominantes



23

Plan de l'exposé

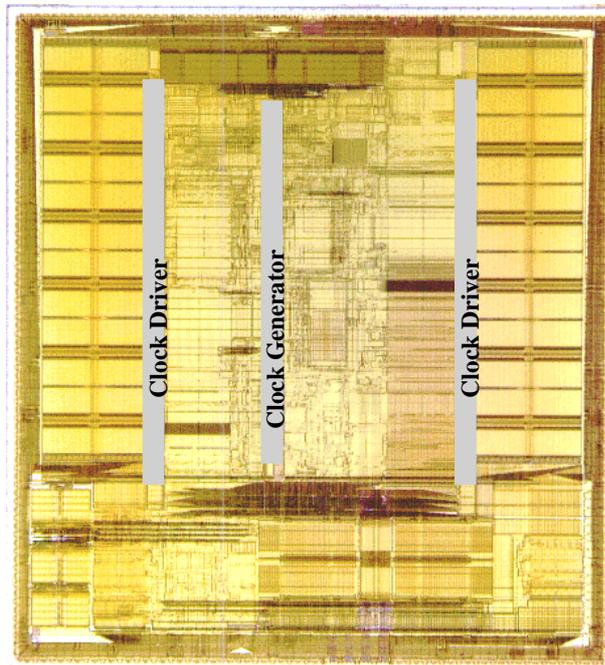


- Pourquoi s'occuper de la puissance ?
- **Où part la puissance ?**
 - Où part l'énergie ?
 - Puissance dynamique
 - Chemin de données, Mémoire, Horloge ...
- **Comment diminuer la consommation ?**
 - Réduire Vdd, Réduire la capacité effective
 - Transformer l'algorithme
 - Optimiser l'horloge, Optimiser la mémoire
- **Comment estimer la consommation ?**
 - Estimation aux niveaux transistor ou logique
 - Estimation de haut niveau
- **Conclusions**

24

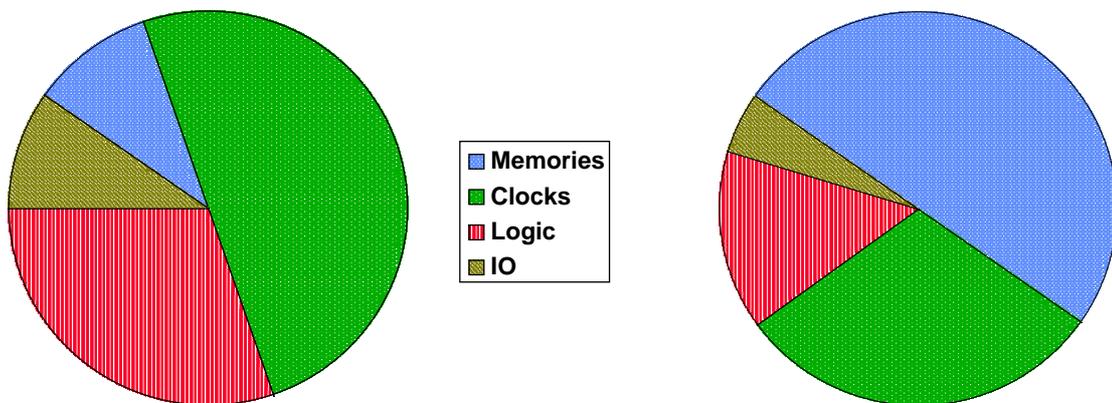
Exemple : DEC α 21164

- 40-50% de la puissance dissipée par l'arbre d'horloge



DIGITAL Corp.
1995
2.5M Portes
9.3M Transistors
300 MHz
64/128 Bits
0.5 μ

Microprocesseurs vs ASICs



DEC Alpha uP

Alcatel telecom chip

Où part l'énergie ?

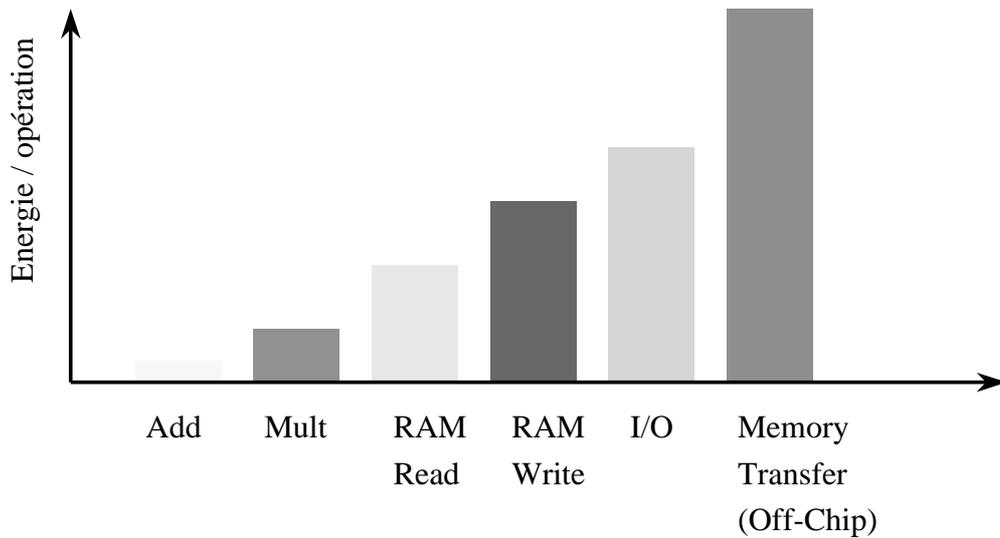
- **Nouvelles Applications Multimédia**

Ex. Codage d'image H263

1500K Opérations et 500K Transferts mémoire

Energie transfert = 33 . Energie Opération

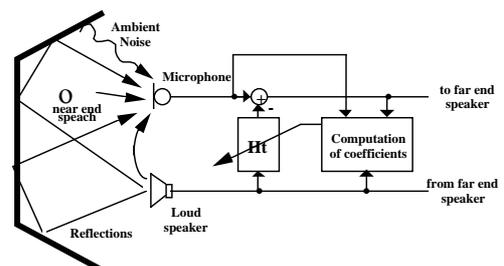
-> Puissance mémoire # 10 . Puissance opérations



Exemple

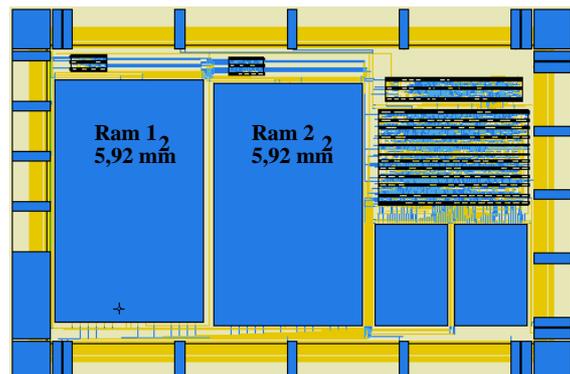
- **Circuit d'annulation d'écho acoustique**

- 16kHz, 1024 points, 0.7µm
- 100 MIPS
- 2700 MIPS/Watt
- Sur DSP 0.5 Watt !



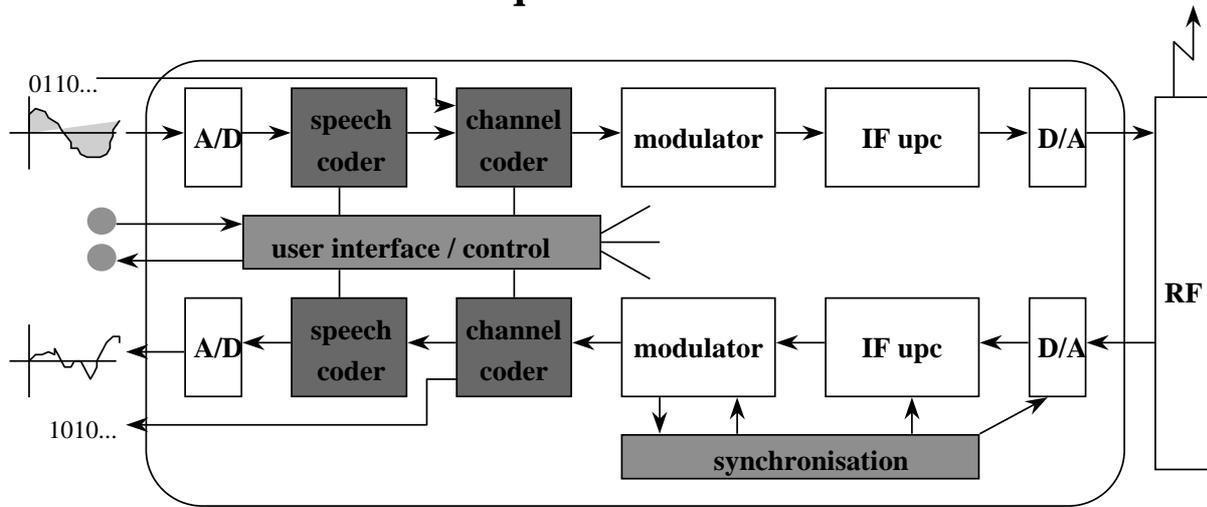
Partie du circuit	Conso. (mW)
interconnexions	12.0
cellules	26.0
RAM	48.0
Puissance Totale	86.0
VDD = 5V	

Vdd = 3.3V → 37mW



Exemple : GSM

Environ équivalant à 60 MIPS



- Coprocessor, Hardware
- DSP, ASIP, ASP
- μ Controller



Analyse des consommation

- **Combiné portable 6V**

	Paging Mode 80% of time	Speaking Mode 20% of time	Total
Radio F	40%	15%	20%
Power Ampli	0%	50%	40%
Base Band Codec Timing	60%	35%	40%

[Source Philips]

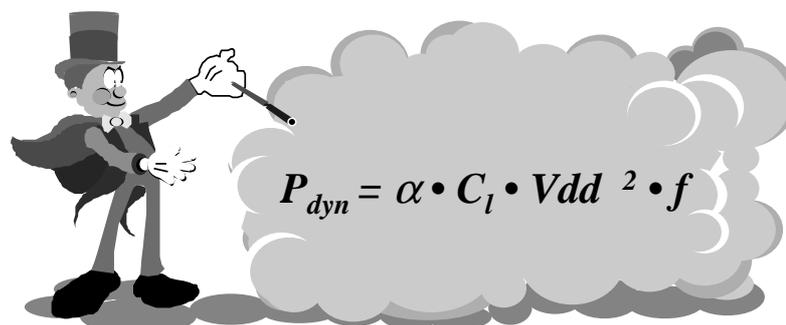
- Pourquoi s'occuper de la puissance ?
- **Où part la puissance ?**
 - Où part l'énergie ?
 - Puissance dynamique
 - Chemin de données, Mémoire, Horloge ...
- Comment diminuer la consommation ?
 - Réduire Vdd, Réduire la capacité effective
 - Transformer l'algorithme
 - Optimiser l'horloge, Optimiser la mémoire
- Comment estimer la consommation ?
 - Estimation aux niveaux transistor ou logique
 - Estimation de haut niveau
- Conclusions

31

Où part la puissance ?

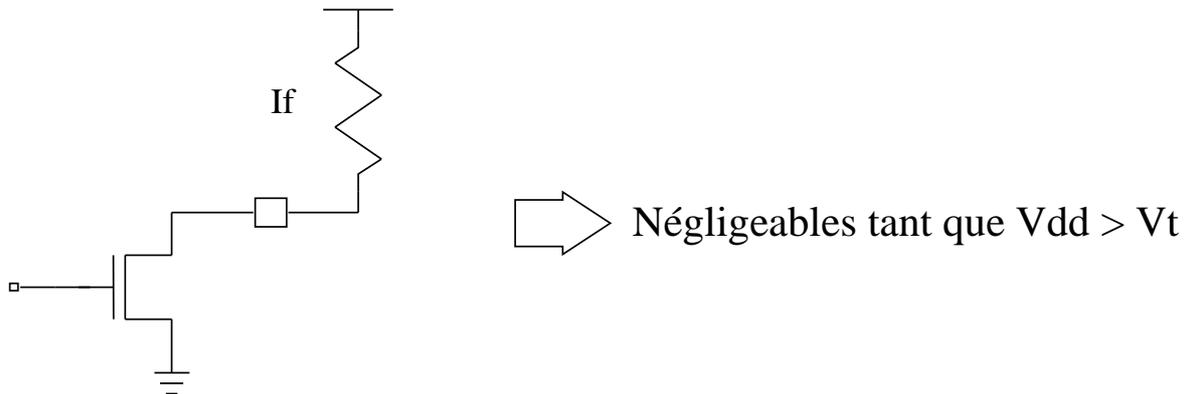
$$P = P_c + P_{cc} + P_f$$

- 
- **Puissance de charge : P_c**
 - Charge et décharge des capacités du circuit
 - **Courants de court-circuit : P_{cc}**
 - Chemin de court circuit pendant la commutation des structures logiques statiques
 - **Courants de fuite : P_f**
 - Jonctions, fonctionnement sous le seuil



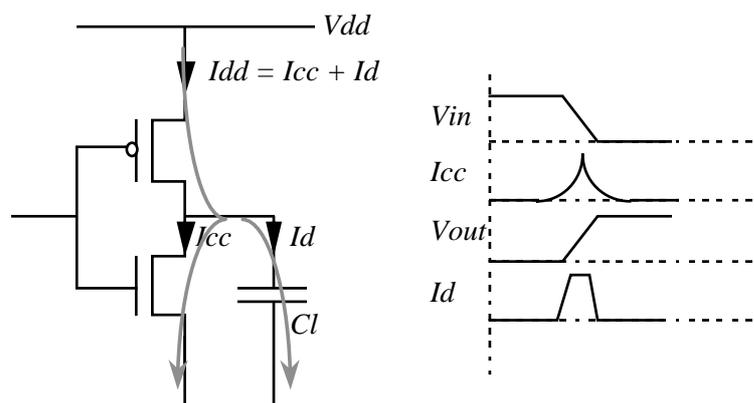
32

- **Courants de fuite (Subthreshold Leakage Current)**
 - Même lorsque $V_{gs} < V_t$ les transistors MOS ne sont pas complètement fermés
 - pA to nA range
 - Si on diminue V_{dd} ($\#V_t$) alors ces courants peuvent devenir significatifs
- **Courants inverses de diode Source/Drain - Substrat**



33

- **Courants de court-circuit : I_{cc}**
 - Chemin de court circuit pendant la commutation des structures logiques statiques
 - Pour les circuits *conçus correctement* $\approx 15\%$
 - En cas de fronts lents (slow input slopes), I_{cc} peut devenir très important
- **Courants de charge/décharge des capacités : I_{dd}**



34

Puissance dynamique

- **Energie / Transition = $C_l \cdot Vdd^2$**
- **Puissance = Energie / Transition • Vitesse de transition**
 $= C_l \cdot Vdd^2 \cdot f$
- **La puissance n'est pas fonction de la taille des transistors**

$$\begin{aligned}
 P_{dyn} &= C_l \cdot Vdd^2 \cdot f_{0 \rightarrow 1} \\
 &= C_l \cdot Vdd^2 \cdot f \cdot P_{0 \rightarrow 1} \\
 &= \alpha \cdot C_l \cdot Vdd^2 \cdot f \\
 &= C_{EFF} \cdot Vdd^2 \cdot f
 \end{aligned}$$

- C_{EFF} est la capacité effective = $C_l \cdot P_{0 \rightarrow 1}$

5-10-20% de la capacité totale

Puissance



Dépend des données
Dépend de l'activité

35

Puissance dépend des données

- **Exemple : porte inverseur** $P(0 \rightarrow 1) = P(\text{OUT} = 0) \cdot P(\text{OUT} = 1) = 1/4$
- **Exemple : porte NOR statique à 2 entrées**

A	B	OUT
0	0	1
0	1	0
1	0	0
1	1	0

Fonction NOR

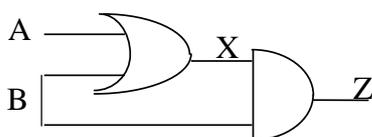
Hypothèse : $P(A=1) = 1/2$ et $P(B=1) = 1/2$

$$P(\text{OUT}=1) = 1/4$$

$$\begin{aligned}
 P(0 \rightarrow 1) &= P(\text{OUT} = 0) \cdot P(\text{OUT} = 1) \\
 &= 3/4 \cdot 1/4 = 3/16
 \end{aligned}$$

$C_{eff} = 3/16 \cdot C_l$

- **Deviens rapidement complexe!**



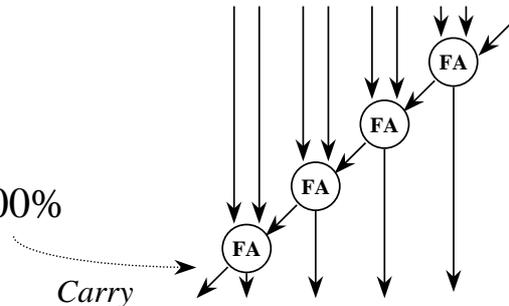
$$P(Z=1) = P(B=1) \cdot P(X=1 | B=1)$$

36

- **Activité α**
 - Dépend de la statistique du signal
 - Activité des noeuds peut être réelle ou parasite : dépend fortement de la fonction
 - Pour un μP - 10%, pour un bloc logique - 6%
- **Estimation de l'activité**
 - Estimation probabiliste
 - Estimation statistique (Monte-Carlo)

- **Exemple : Ripple-Carry**

$$\alpha = \frac{\text{Nb de portes qui commutent}}{\text{Nb total de portes}} - 100\%$$



- **Exemple : Multiplieur**

- Grand nb de glitches
- Difficile à estimer

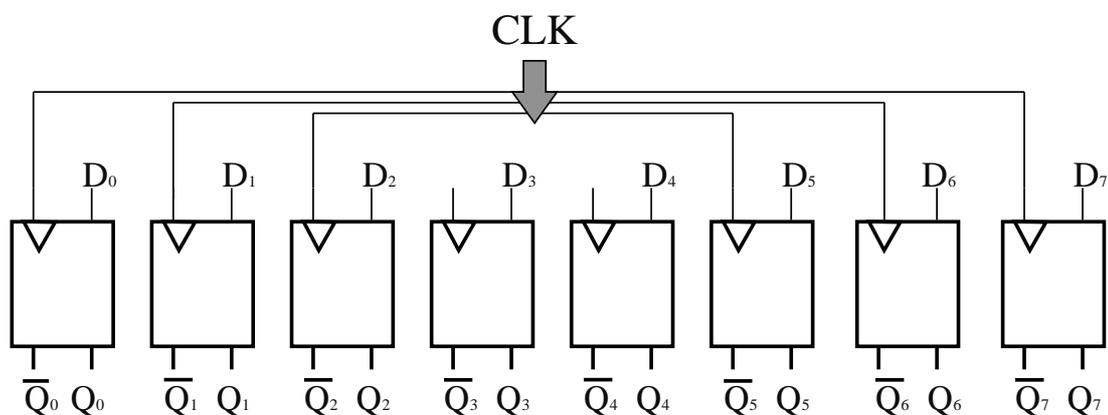
	8 bits	16 bits	32 bits
Tableau	32	1,1	0,04
Wallace	50	5	0,81

inverse du Produit Puissance x Délai

(doit être le plus grand possible)

Registres

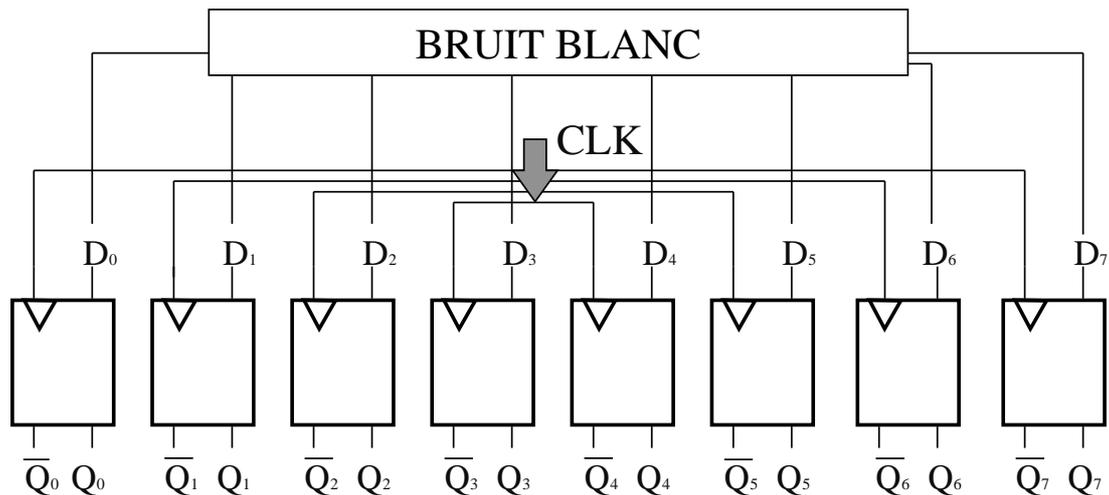
- **consommation d'une bascule : 9.5 μ W/MHz par transition de 0 à 1 à la sortie**



Registres

- **Approche statistique :**

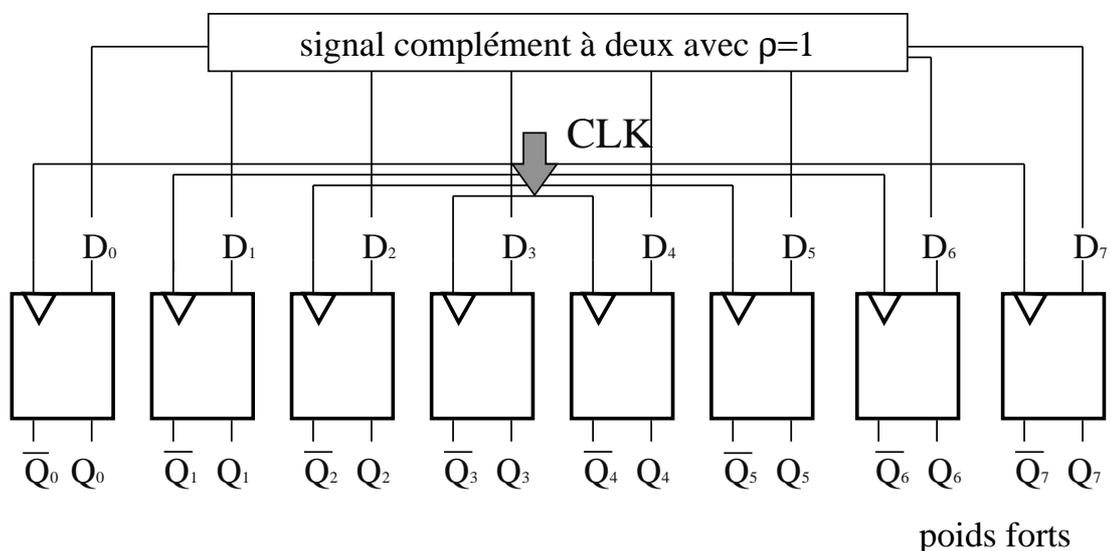
- On envoie un signal aléatoire pour estimer la consommation d'un registre.
- Environ 4 bascules commutent pour un signal aléatoire en entrée, dont deux de 0 à 1.
- $2 \times 9.5 = 19 \mu\text{W}/\text{MHz}$



39

Registres

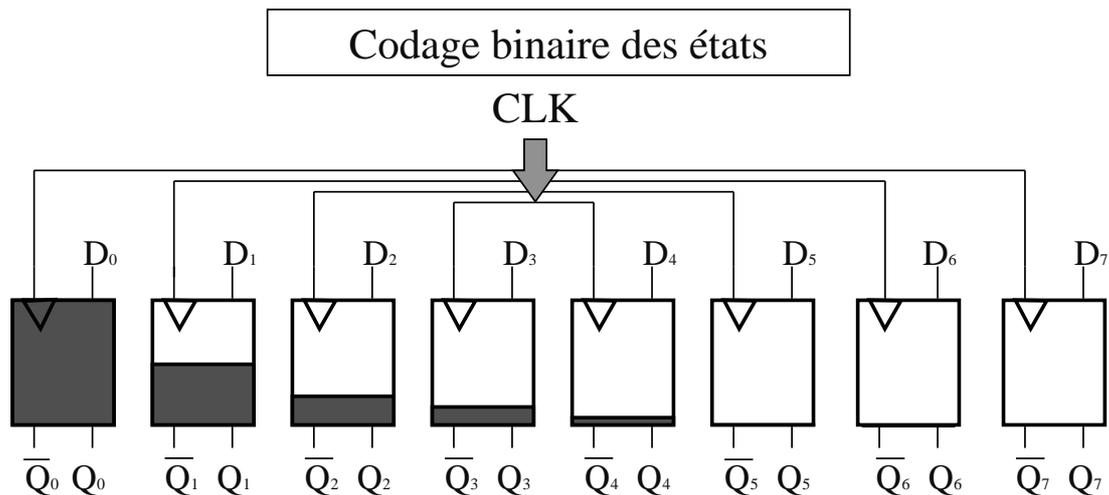
- **Les bits de poids forts ne commutent pas.**
- **La consommation est : $0.25 \times 6 \times 9.5 = 14.25 \mu\text{W}/\text{MHz}$**



40

Registre d'états

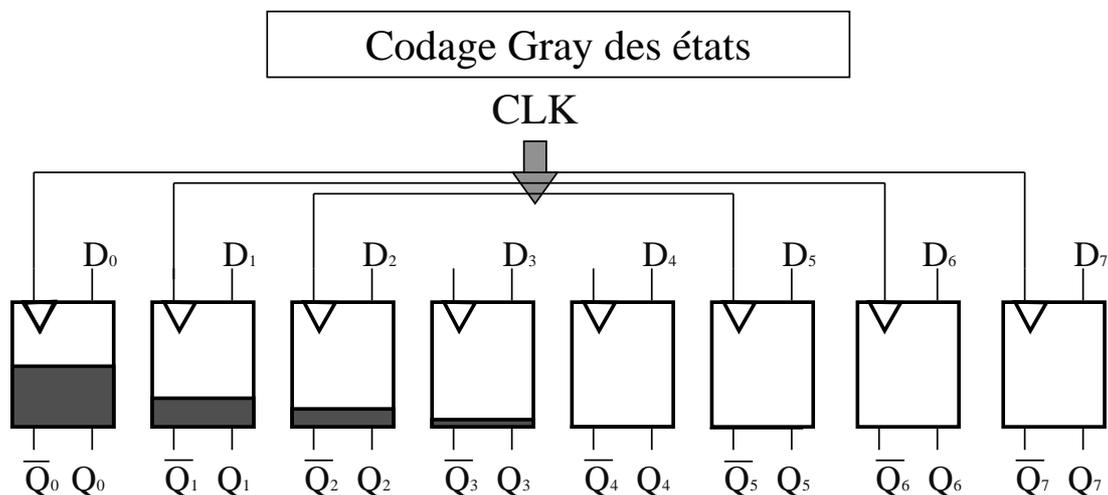
- **Approche probabiliste liée au signal**
 - différentes probabilités suivant le signal et l'utilisation.
- **$1 + 1/2 + 1/4 + 1/8 \dots = 2$**
- **Consommation : $2 \times 9,5 / 2 = 9,5 \mu\text{W}/\text{MHz}$**



41

Registre d'états

- **Approche probabiliste liée au signal**
 - différentes probabilités suivant le signal et l'utilisation.
- **$1/2 + 1/4 + 1/8 \dots = 1$**
- **consommation : $1 \times 9.5 / 2 = 4.75 \mu\text{W}/\text{MHz}$**



42

Pourquoi s'occuper de la puissance ?

Où part la puissance ?

Comment diminuer la consommation ?

- Réduire Vdd
- Réduire la capacité effective
- Transformer l'algorithme
- Optimiser l'horloge
- Optimiser la mémoire

Comment estimer la consommation ?

- Estimation aux niveaux transistor ou logique
- Estimation de haut niveau

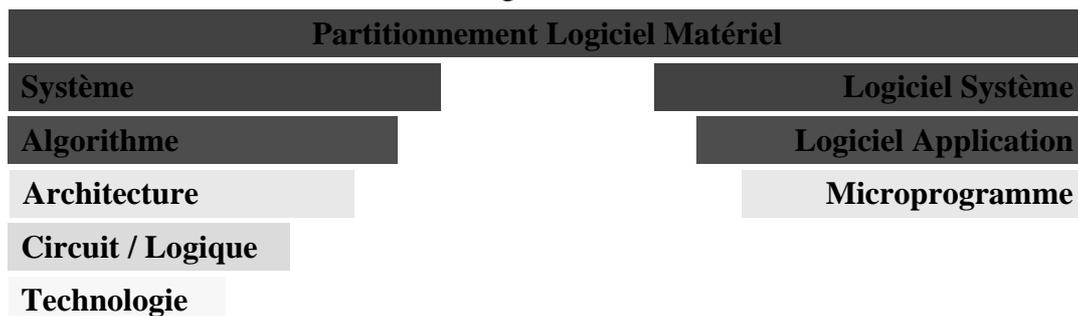
Conclusions

43

Où gagner en consommation ?



Partitionnement logiciel/matériel 10-95%



Système et Algorithme 30-95%

Architecture 10-90%

Logique 15%

Physique 20%

Technologie 30%

Logiciel 10-80% (sur le système)

44

Comment diminuer la consommation ?

- Diminuer au maximum la tension d'alimentation
- Minimiser la capacité effective C_{eff}
- Conserver, si besoin est, la fréquence d'horloge, donc les performances

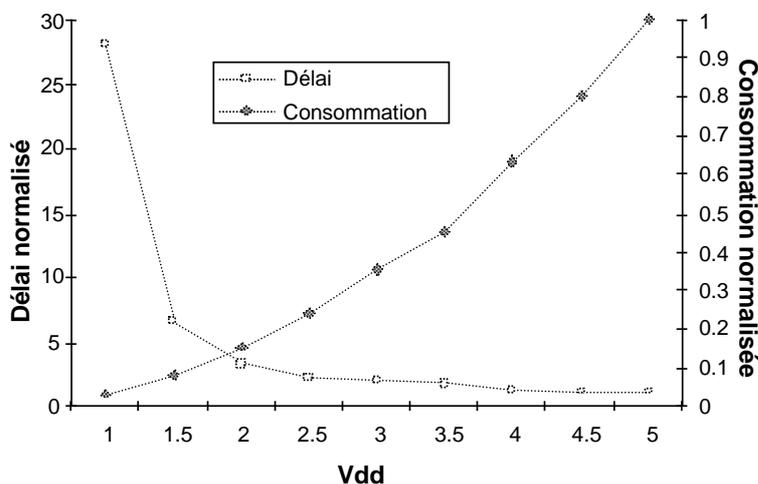


Système	Partitionnement, Mise en veille
Algorithme	Concurrence, Régularité, Localité, Complexité
Architecture	Parallélisme, Pipeline, Codage
Circuit / Logique	Taille, Styles logiques, Routage, Conception logique
Technologie	Réduction V_t

45

Réduire Vdd

- Puissance = fonction quadratique de Vdd
- Le temps de propagation augmente si Vdd est réduit
→ Produit *puissance x delay* ↑ si Vdd ↓
- Influence de Vdd ne dépend pas du style logique



$$T_d = \frac{C_l \cdot V_{dd}}{I} = \frac{C_l \cdot V_{dd}}{k(W/L)(V_{dd} - V_t)^2}$$

$$P_{moy} = \alpha \cdot C_l \cdot V_{dd}^2 \cdot f$$

46

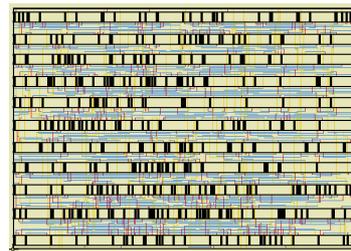
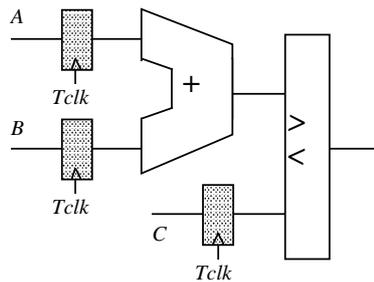
Réduire Vdd : Approche Architecturale

- Compenser la réduction de Vdd apportant une baisse de performances par des optimisations architecturales ou logiques
- Exemple Data Path 16 bits

$$P_{ref} = C_{ref} \cdot V_{ref}^2 \cdot F_{ref} = 14.7 \text{ mW}$$

C_{ref} déterminé par simulation ($Cl = 31\text{pF}$)

$$F_{ref} = 1 / 40\text{ns} \quad V_{ref} = 5\text{V}$$



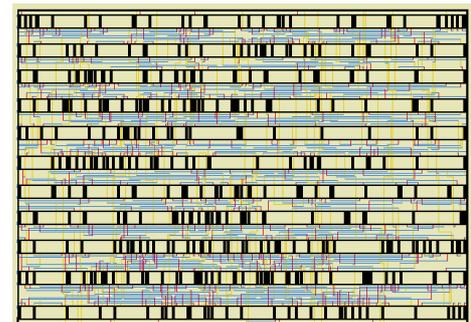
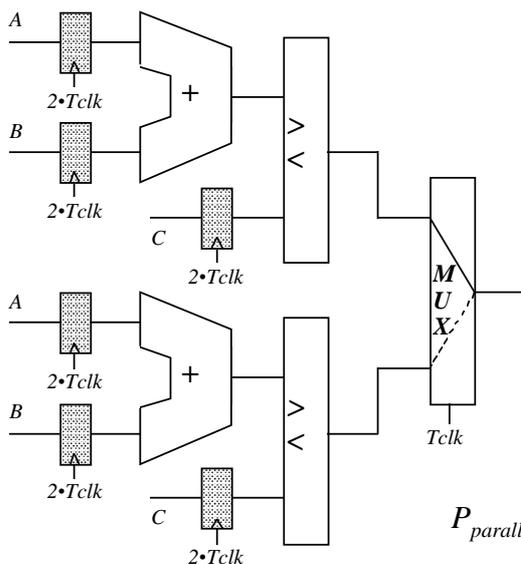
$F_{clk} = 25 \text{ MHz}$
 $V_{dd} = 5\text{V}$
 $P_{moy} = 14.7 \text{ mW}$

Surface = $788 \times 555 \mu\text{m}^2 = 0.44 \text{ mm}^2$

3823 Transistors
 56 mm d'interconnexions

47

Architecture Parallèle



Surface = $1130 \times 768 \mu\text{m}^2 = 0.87 \text{ mm}^2$

7638 Transistors
 111 mm d'interconnexions

$$P_{parallèle} = (2.15 \cdot C_{ref}) \cdot (0.58 \cdot V_{ref})^2 \cdot (0.5 \cdot F_{ref})$$

$$\# 0.36 \cdot P_{ref} = 5.3 \text{ mW}$$

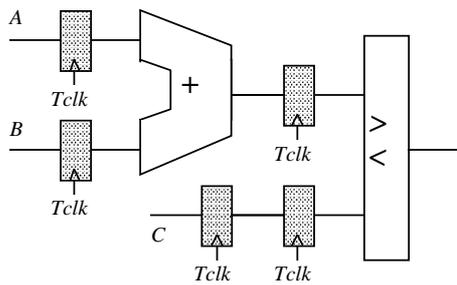
$$C_{parallèle} = 2.15 \cdot C_{ref}$$

$$F_{parallèle} = 1 / 80\text{ns}$$

$$V_{dd \text{ parallèle}} = 2.9\text{V}$$

48

Architecture Pipeline



$$P_{pipeline} = (1.15 \cdot C_{ref}) \cdot (0.58 \cdot V_{ref})^2 \cdot F_{ref}$$

$$\# 0.39 \cdot P_{ref} = 5.7 \text{ mW}$$

$$C_{parallèle} = 1.15 \cdot C_{ref}$$

$$F_{parallèle} = F_{ref}$$

$$V_{dd \text{ parallèle}} = 2.9V$$

• Résumé

Type d'Architecture	Vdd	Surface	Puissance
Référence	5V	1	1
Parallèle	2.9V	1.98	0.36
Pipeline	2.9V	1.1	0.39
Parallèle/Pipeline	2.0V	2.2	0.2

x2,5

x5

49

Réduction de la capacité effective



Ceff = Capacité Physique x Activité de Commutation

• Packaging et Layout

- PCB vs MCM
- Placement / Routage dirigé par l'activité

• Niveau Logique / Transistors

- Logique CMOS Statique ou dynamique, Logique à base de portes de transferts (CPL), Logique différentielle (DCVSL)
- Style Logique (ripple vs carry select)
- Gated Clock

• Niveau Architectural

- Préservation de la corrélation des données
- Hiérarchisation mémoire
- Distribution de l'horloge

• Sélection d'algorithmes

• Architecture Système

Simplicité!

Eviter le gaspillage!

50

Au Niveau Logique

Réduction de la capacité effective



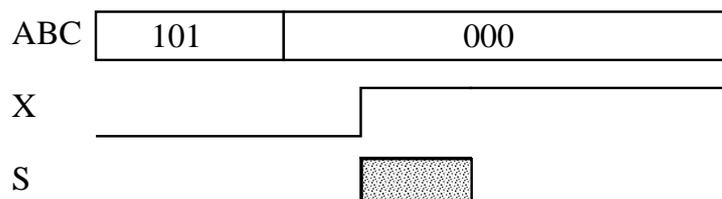
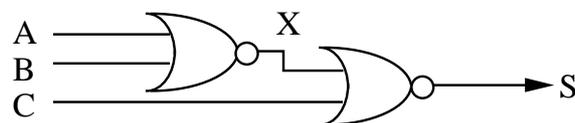
- **Glitch (hasards dynamiques)**

Transitions dynamiques parasites

Consommation majeure (ET INUTILE) de courant

Equilibrage des chemins logiques : Synthèse logique

Facteur 5 sur une version optimisée d'un multiplieur [Callaw92]



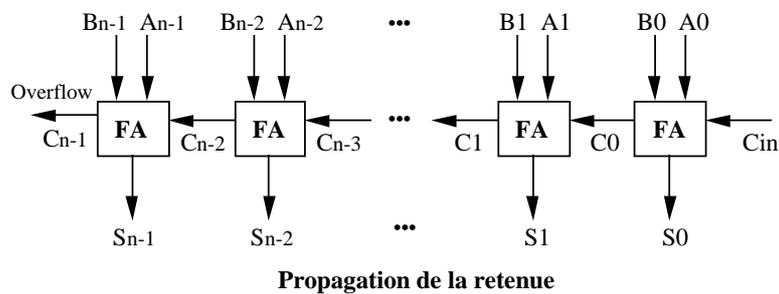
Réduire activité : niveau logique

- Exemple : Etude de l'additionneur

1 / Puissance x Temps de propagation

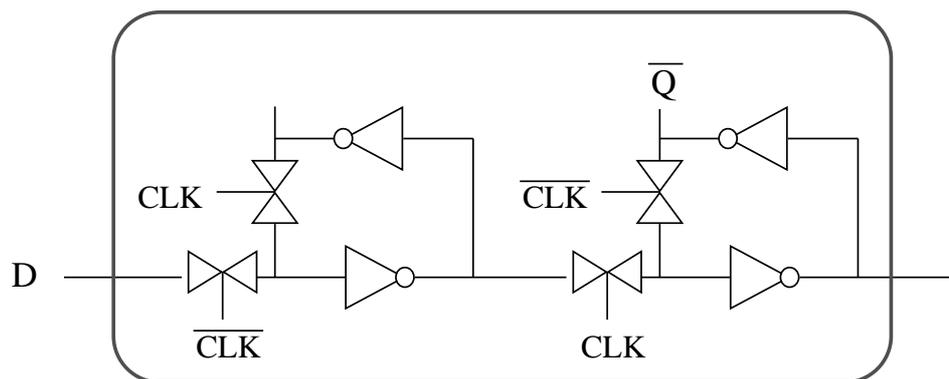
	16 bit	32 bit	64 bit
Ripple Carry	3.09	0.81	0.27
Carry Lookahead	10.0	3.54	1.76
Carry Bypass	5.45	2.39	0.99
Carry Select	4.44	2.08	1.00
Conditional Sum	3.82	1.23	0.42

Après [Callaw92]

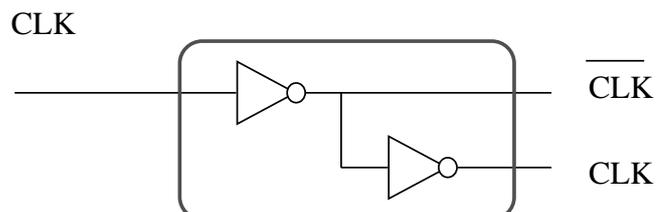


53

Modèle de bascule



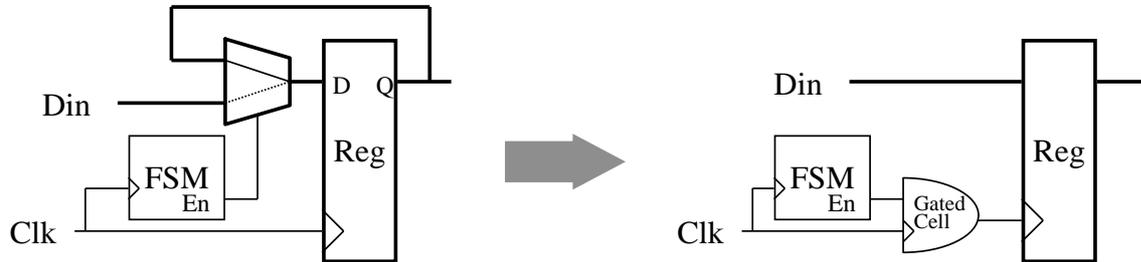
$$P_{\text{basc}} = \alpha P_{\text{dyn}} + P_{\text{horl}}$$



54

Gated Clock

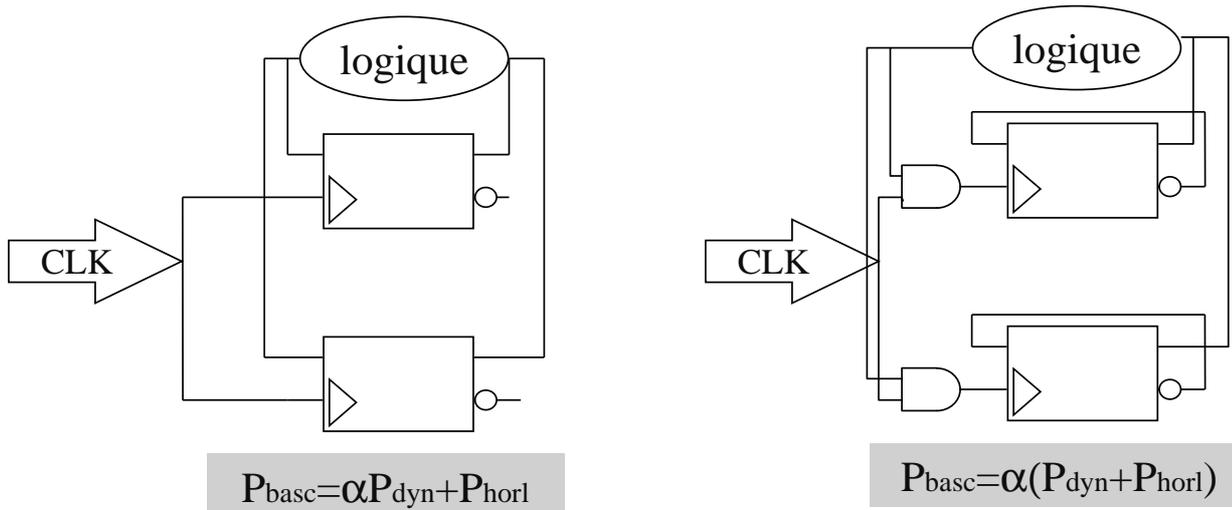
- Supprimer les commutations des blocs qui n'ont pas besoin d'être rechargés



- Modification de la machine d'états
- Gain important
- Attention cependant aux *glitches* et à la couverture de fautes

55

Registres d'états optimisés



$$P_{basc} = \alpha P_{dyn} + P_{horl}$$

$$P_{basc} = \alpha (P_{dyn} + P_{horl})$$

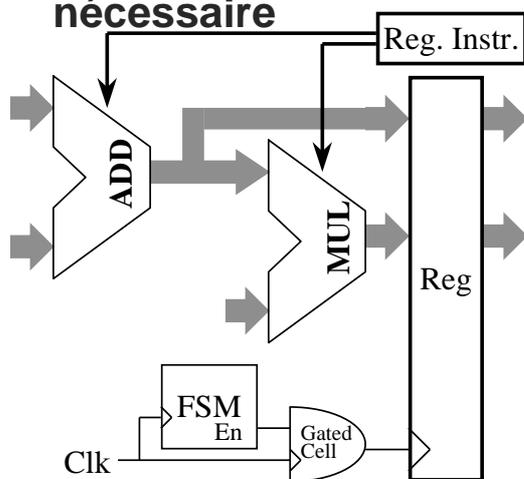
$$\text{Gain} = (1 - \alpha) P_{horl}$$

$$\text{Gain}_{reg} = \sum (1 - \alpha_i) P_{horl} = (N - 1) P_{horl}$$

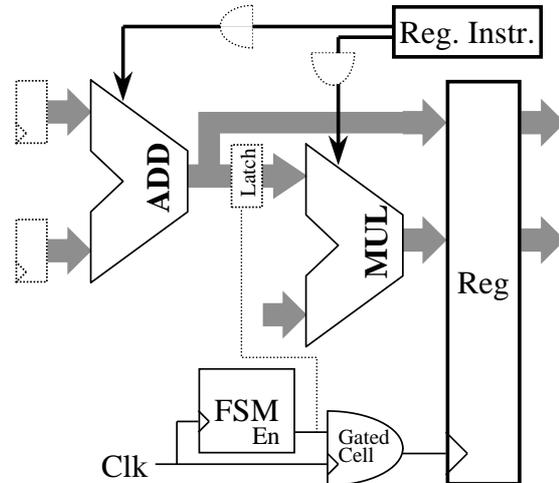
56

Isolation d'opérateurs

- Activer les UF uniquement lorsque cela est nécessaire



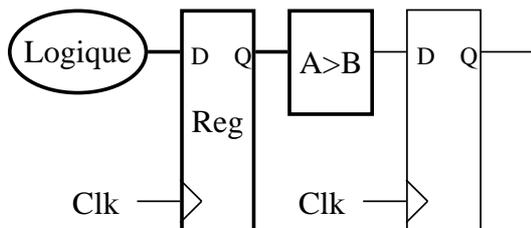
Le multiplieur consomme même si ses sorties sont inutilisées



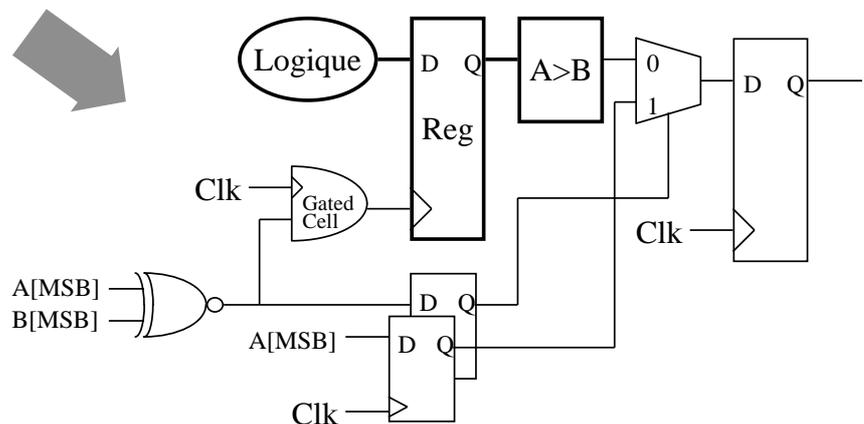
Les entrées du multiplieur sont bloquées si ses sorties sont inutilisées
Les instructions sont bloquées

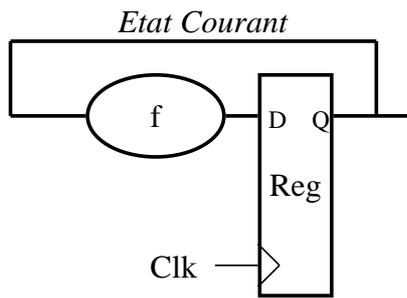
- Bien gérer les Instructions
- Placer des registres en E/S des UF

Précalcul

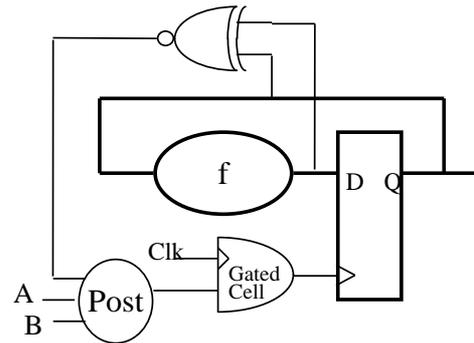
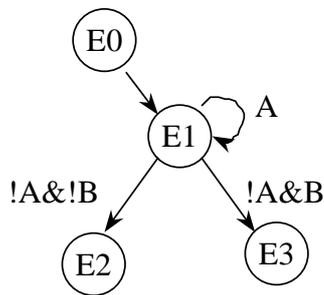


Le comparateur (soustracteur) peut être bloqué quand les deux MS Bits diffèrent ($A[\text{MSB}] \neq B[\text{MSB}]$)





Reconnaître si l'état suivant d'une machine sera le même que l'état courant.
En déduire un signal de transition et un pour le verrouillage dans l'état.



Codage d'état

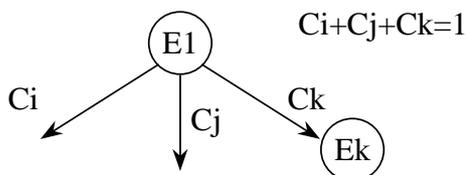
- Code pour les états d'une machines

Gray

- Activité
- + Capacité

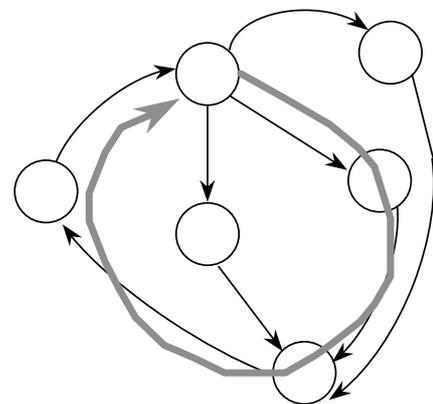
Binaire

- + Activité
- Capacité



Coder le graphe avec le minimum de transitions

Si $Prob(C_k) > 0.8$ Alors
les codes de E1 et Ek doivent différer
avec un minimum de positions
(distance de Hamming)



Chemin le plus probable

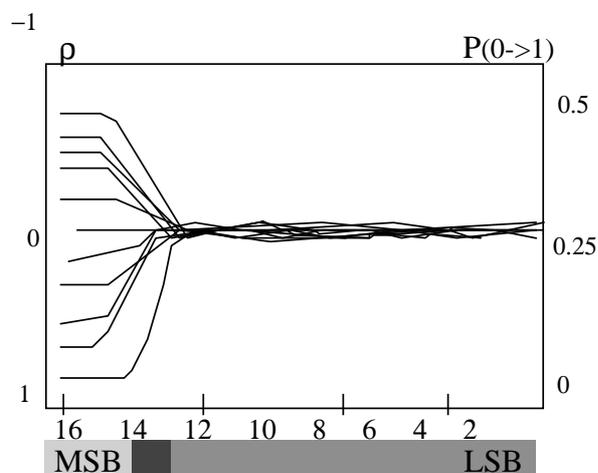
A coder avec un Min de transitions

Au Niveau Architectural

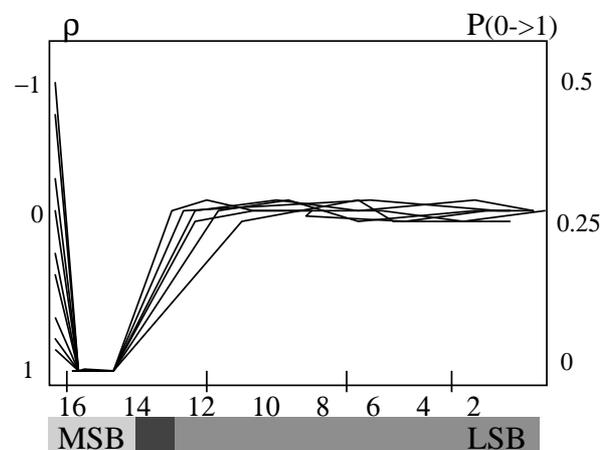
Prise en compte du signal



- **Activités des bits pour une suite de données gaussienne ayant un facteur de corrélation ρ temporel**



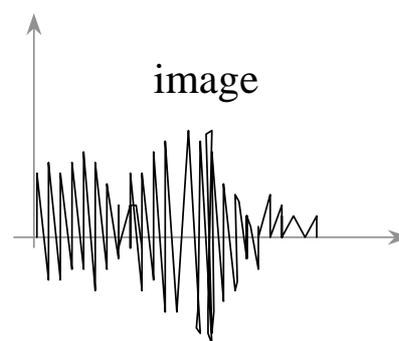
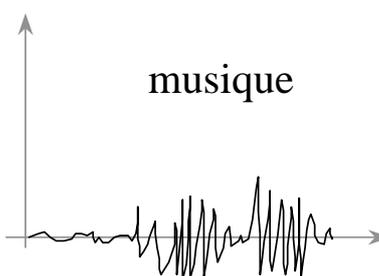
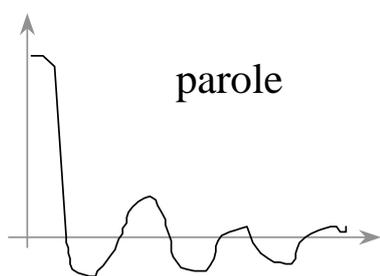
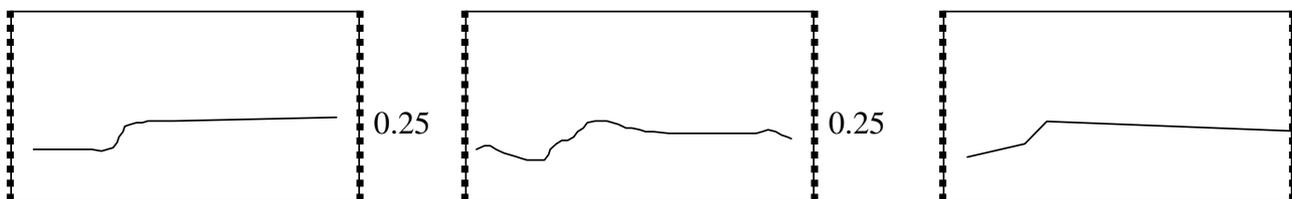
représentation en complément
à deux



représentation signe
et amplitude

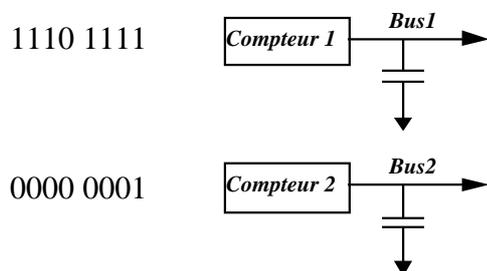
Probabilités du signal

- Probabilité de transition 0 à 1 pour les différents signaux en représentation complément à deux.

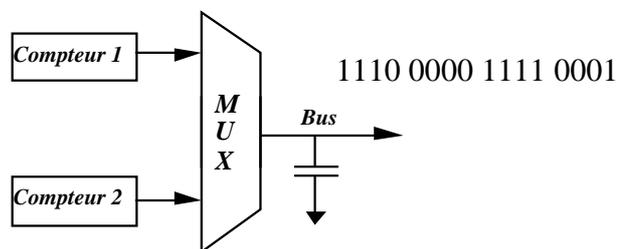


Réduction de la capacité effective

- Partage de ressources augmente l'activité
 - Destruction des corrélations entre signaux



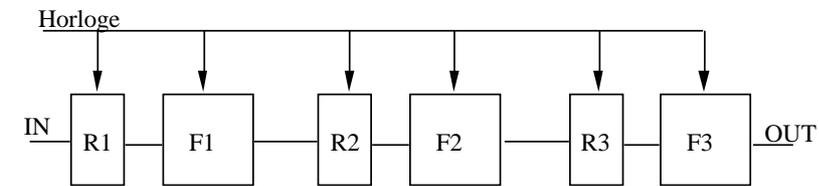
a) Implémentation parallèle



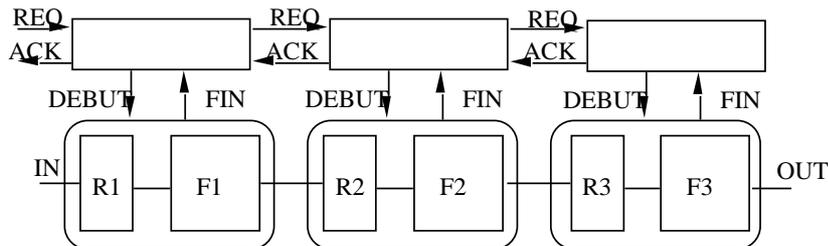
b) Multiplexage temporel

Et l'horloge ?

- Déconnexion de l'horloge
- Synchrones ou localement asynchrones

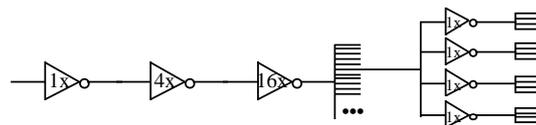


Synchrone



Asynchrone

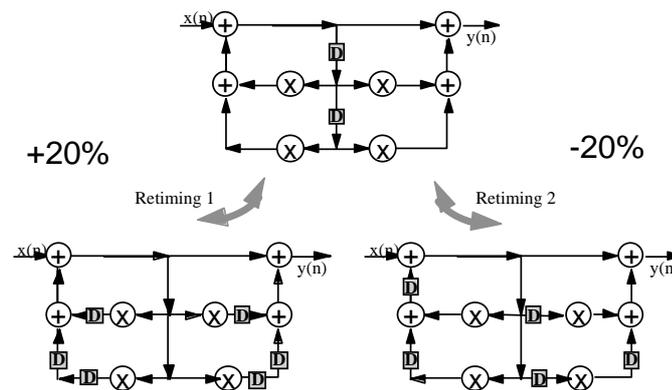
- Distribution optimisée de l'horloge



Au Niveau Algorithmique

Diminuer la consommation

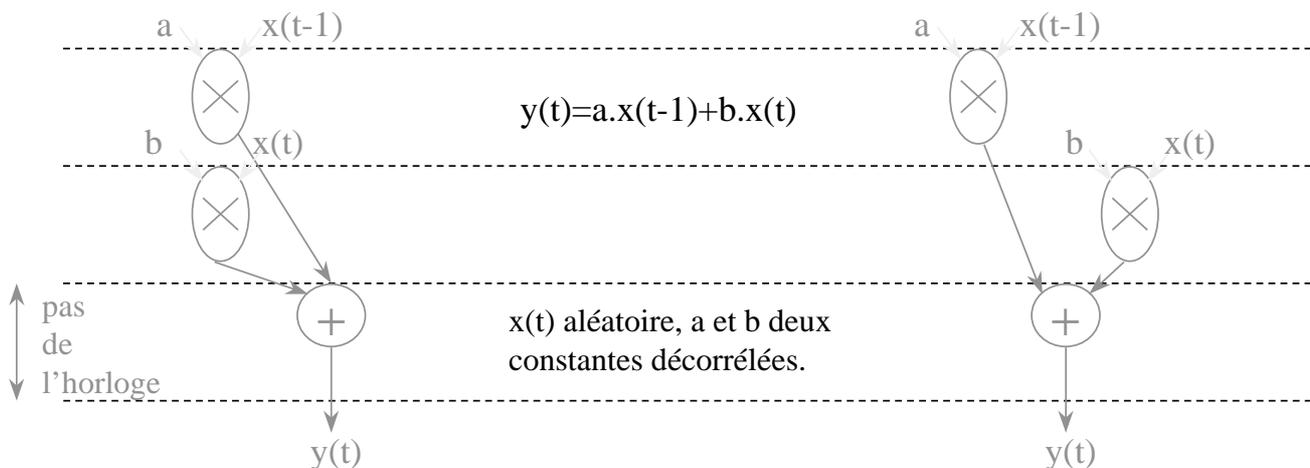
- Réduire Vdd en utilisant des transformations améliorant les performances temporelles
- Réduire Ceff en utilisant des transformations
 - Réduire le nombre d'opérations
 - Remplacement d'opérations
 - Optimisation de l'utilisation des ressources
 - Réduction de la taille des chemins de données



67

Allocation de ressources

- Importance des statistiques du signal



2 multiplications sur le même multiplieur.
 une multiplication dissipera : 7.4mW/MHz
 d'où : environ 15mW/MHz

2 multiplieurs, une multiplication sur chacun.
 multiplication par une constante : 2.9mW/MHz
 soit : environ 6mW/MHz

68

- **Le logiciel détermine la puissance dissipée par le processeur**
 - Pourquoi ne pas modifier le logiciel pour réduire l'énergie ?
- **Cas du logiciel enfoui**
- **Énergie et puissance**
 - Énergie = durée de vie des batteries

MOV DX,[BX]
MOV AX,CX
MOV AX,DX

Puissance : 1.15 W
Energie : $8,6 \cdot 10^{-8}$ J

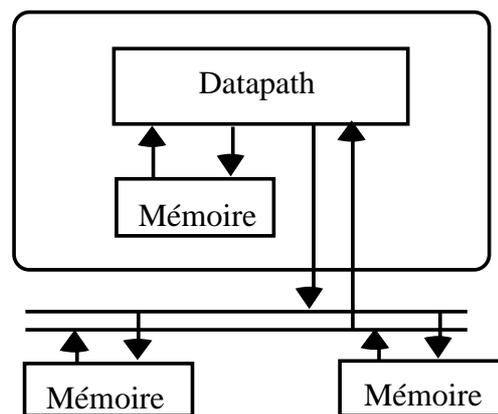
NOP
MOV DX,[BX]
NOP
NOP

Puissance : 0.99 W
Energie : $22,3 \cdot 10^{-8}$ J

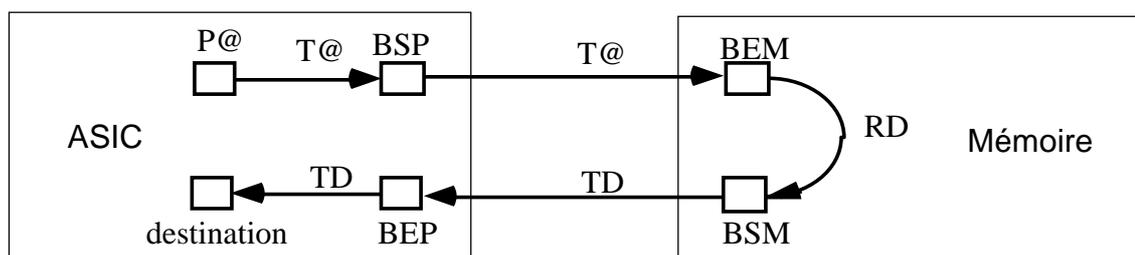
MOV AX,CX
NOP
NOP
MOV AX,DX
NOP

14% de moins
158% de plus

Et la mémoire ?



Architecture mémoire générale



Modélisation de la consommation lors d'un accès mémoire

- Transformations pour réduire le nombre d'opérations de lecture et d'écriture aux mémoires

```
FOR i:= 1 TO N DO  
  B[i] := f(A[i]) ;  
FOR i:= 1 TO N DO  
  C[i] := g(B[i]) ;
```

```
FOR i:= 1 TO N DO BEGIN  
  B[i] := f(A[i]) ;  
  C[i] := g(B[i]) ;  
END ;
```

a) Code avant transformation

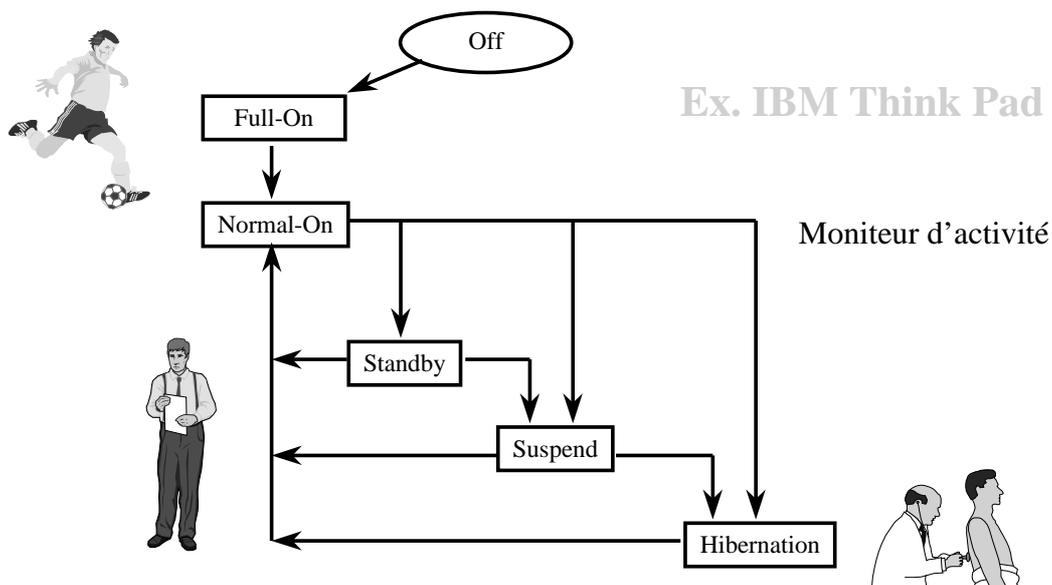
b) Code après transformation

- Mise en place de buffers (cache) pour limiter les accès aux mémoires
- Placement des données les plus souvent accédées dans des mémoires internes
- Minimisation de la taille mémoire totale par réutilisation des points mémoires

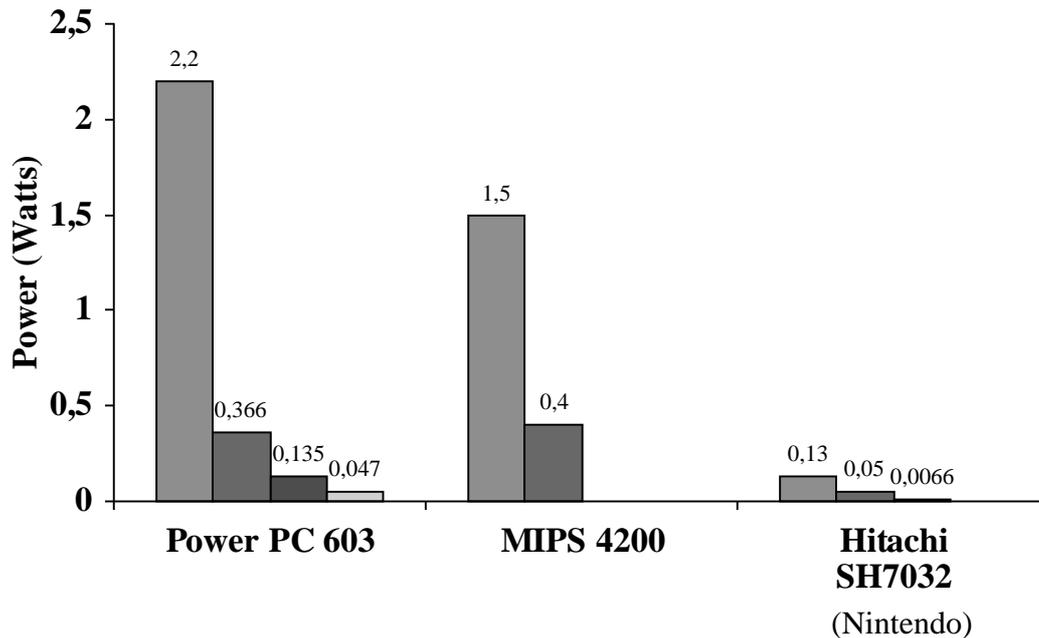
71

Mise en veille

- Différents modes d'opérations



72



Plan de l'exposé



Pourquoi s'occuper de la puissance ?

Où part la puissance ?

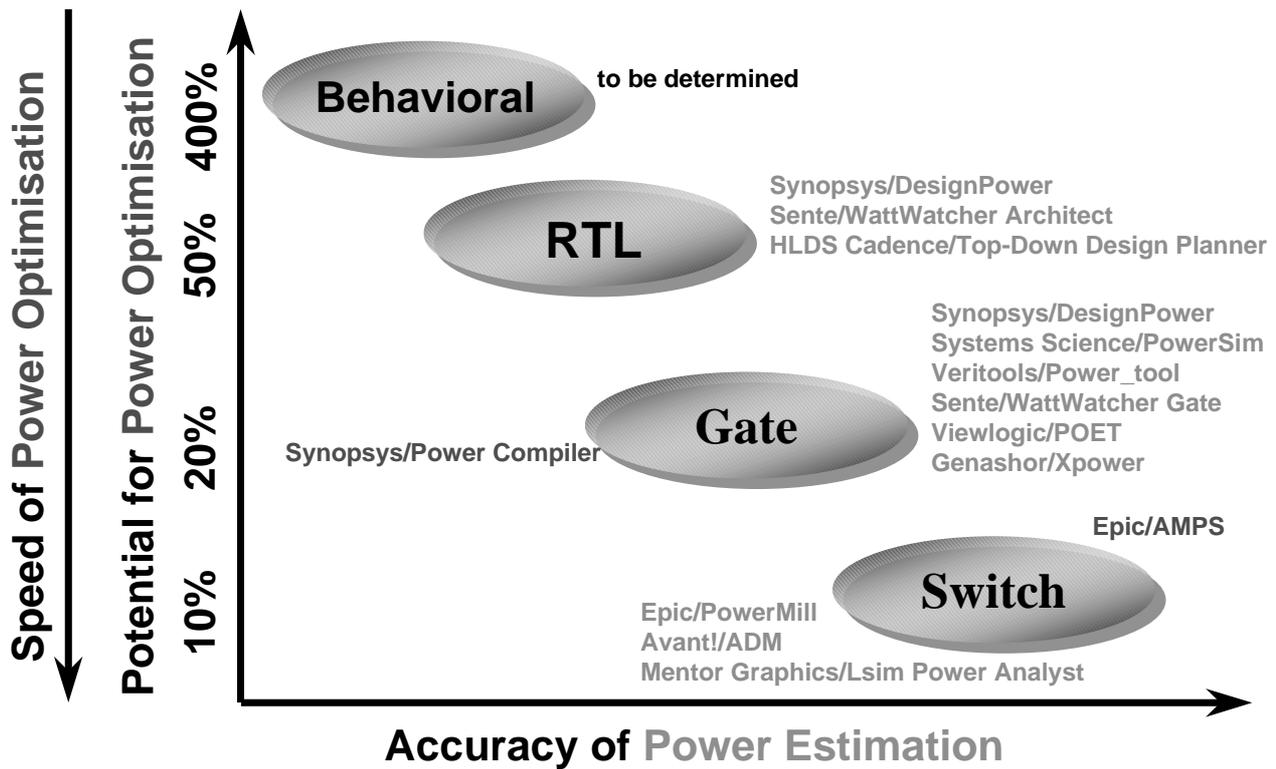
Comment diminuer la consommation ?

Comment estimer la consommation ?

- Estimation au niveau transistor
- Estimation au niveau logique
- Estimation de haut niveau

Conclusions

Outils de conception BC



Panorama des outils commerciaux

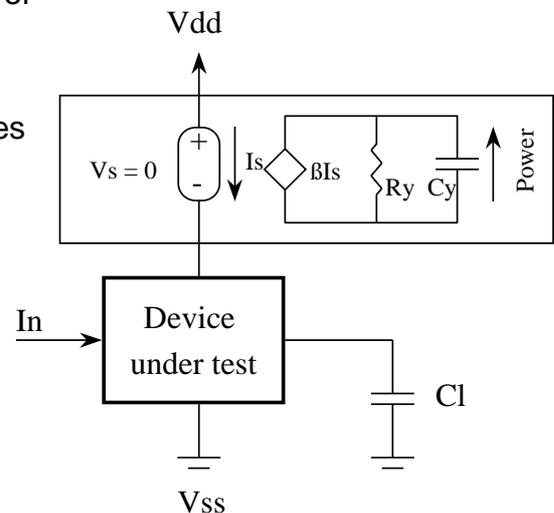


	TRANSISTOR	GATE	RTL	BEHAVIORAL	EMBEDDED SOFTWARE	HW/SW	SYSTEM	ALGORITHMS
ANALYSIS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS
OPTIMIZATION / RESIZING	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS
SYNTHESIS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS
LINKAGE WITH LAYOUT	IMMATURE / NO PRODUCTS	FEW PRODUCTS / NOT WELL SUPPORTED	FEW PRODUCTS / NOT WELL SUPPORTED	IMMATURE / NO PRODUCTS				
POWER BUS ANALYSIS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS
POWER BUS SIZING	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS
THERMAL ANALYSIS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS
SILICON TECHNOLOGY	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS	IMMATURE / NO PRODUCTS

-  IMMATURE / NO PRODUCTS
-  Not Applicable
-  WELL SUPPORTED
-  FEW PRODUCTS / NOT WELL SUPPORTED

Outils au niveau transistor

- **Analyses précises**
 - SPICE (et ses dérivés)
 - PowerMill (Epic), ADM (Avant!), LSIM Power (Mentor)
- **Outils d'optimisation**
 - Optimisation par modification de la taille des transistors : AMPS (Epic)
 - Fiabilité : RailMill, Thunder&Lightning

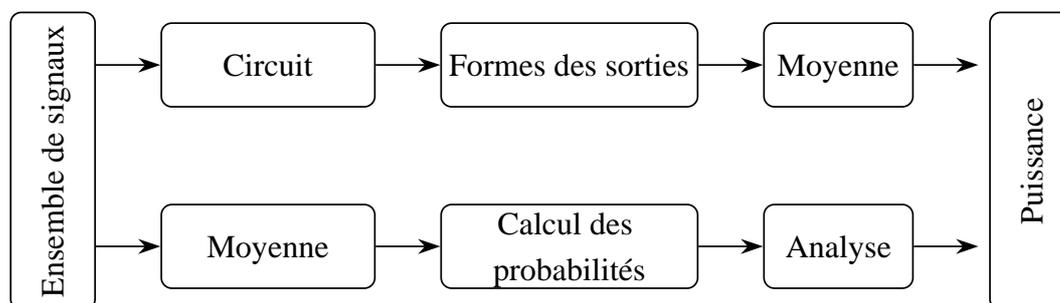


- **Avantages**
 - Haute précision
 - Concept entré dans les moeurs
- **Limitations**
 - Besoin d'extraction 3D pour une meilleure modélisation
 - Capacités limitées
 - Intervient tard dans le processus de conception

77

Outils au niveau logique

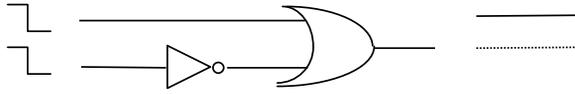
- **Deux techniques**
 - Estimation statistique
 - Estimation probabiliste



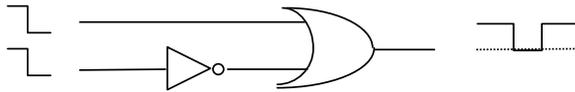
- **Méthodes d'estimation au niveau logique (portes + bascules)**
 - Résolution de la puissance des bascules D par un examen du comportement du circuit comme une FSM, et une mesure de la statistique des sorties des bascules
 - Utilisation des statistiques des sorties des bascules pour calculer la puissance des blocs combinatoires

78

- **Modèle “zero delay”**



- **Modèle “real delay”**



Glitches / Hazards

- Typiquement 20% de la consommation, les *glitches* peuvent atteindre 70% dans les composants arithmétiques (e.g. additionneurs, multiplieurs)

Outils au niveau logique

- **De nombreux exemples**

- PowerGate (Epic), DesignPower (Synopsys), Poet (ViewLogic)
- QuickPower (Mentor Graphics), WattWatcher / Gate (Sente)
- PowerCalc (Zimmerman / Compass)
- PowerCompiler (Synopsys) : optimisation logique

- **Avantages**

- Plus rapides que les outils au niveau transistor
- Utilisent les simulateurs *gate-level* existants
- Utilisent des techniques probabilistes

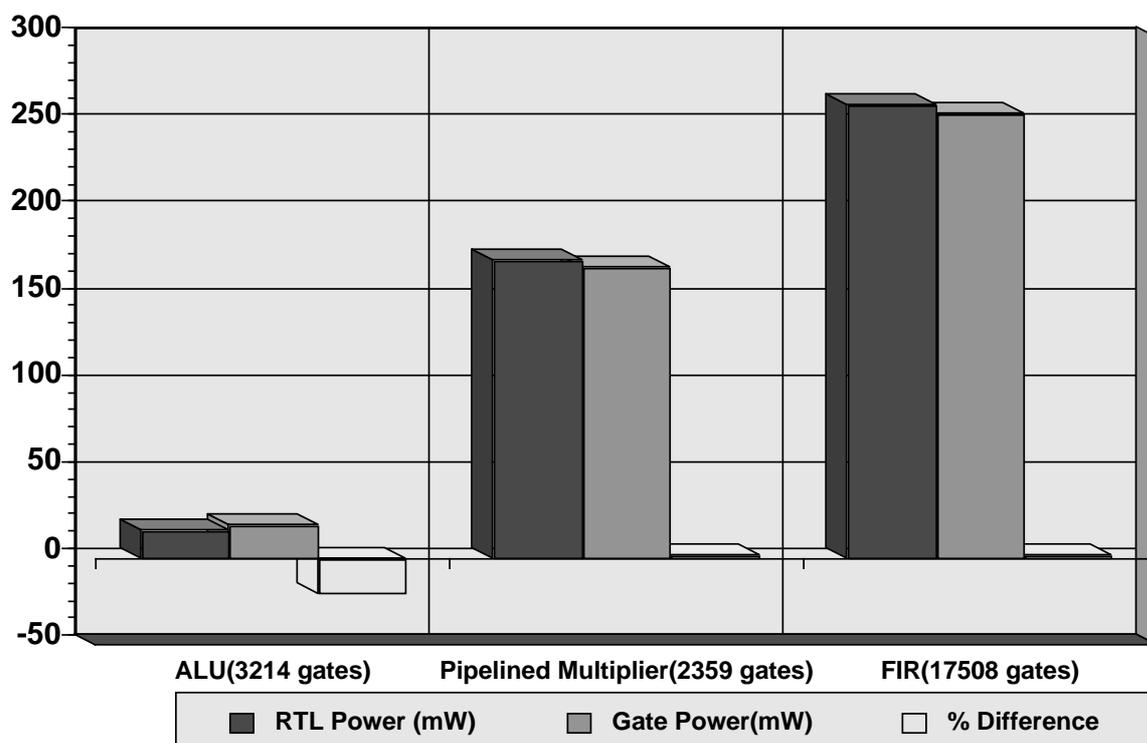
- **Limitations**

- Modèle précis des interconnexions
- Précision de l'estimation des *glitches*
- Capacités et vitesse encore limitées

- **Analyse au niveau RT**
 - WattWatcher / Architect (Sente Inc.)
- **Avantages**
 - Vitesse et capacités
 - Résultats obtenus tôt dans le cycle de conception
 - Analyse complète du circuit possible
- **Limitations**
 - Moins précis que les précédents
- **Concepts clés**
 - Approches “*diviser pour régner*”
 - Macro-modèles
- **Challenges**
 - Estimation des interconnexions
 - Estimation des *glitches*

81

Watt Watcher / Architect

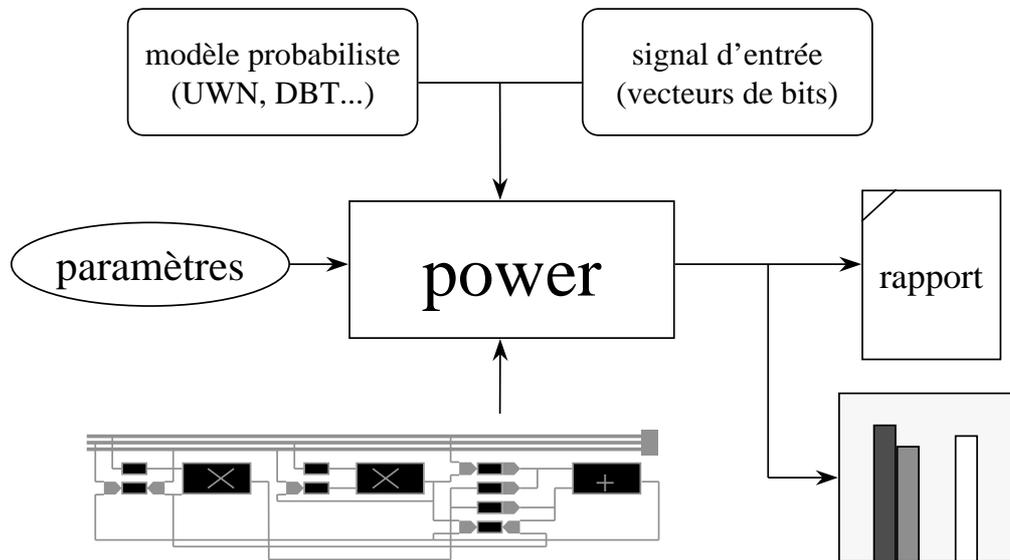


- RTL based analysis is 2% to 20% off the gate level power analysis based on our evaluation.
- Simulation Based RTL analysis In Comparison With Gate Level Analysis

82

• Recherche Académique

- UC Berkeley: Explore, Hyper-LP, PowerPlay
- Illinois: Entropy based analysis
- ENSSAT: Power / GAUT_W



83

Résultats d'estimations

• Applications de type DSP

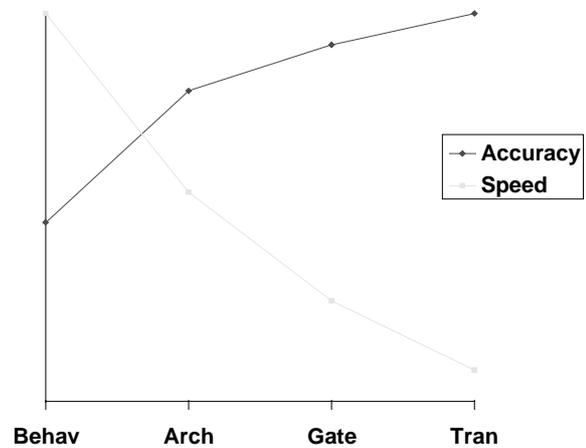
Application	consommation mesurée	consommation estimée
FIR 16	85<-<110 mW	93.4 mW
DCT1d	5<-<7 mW	6.29 mW
LMS 16	600 mW* pour les opérateurs	819.2 mW dont 638 mW pour les opérateurs
FFT 4	135 mW* pour les multiplications	164 mW dont 156 mW pour les opérateurs

84

Etapes critiques dans le cycle



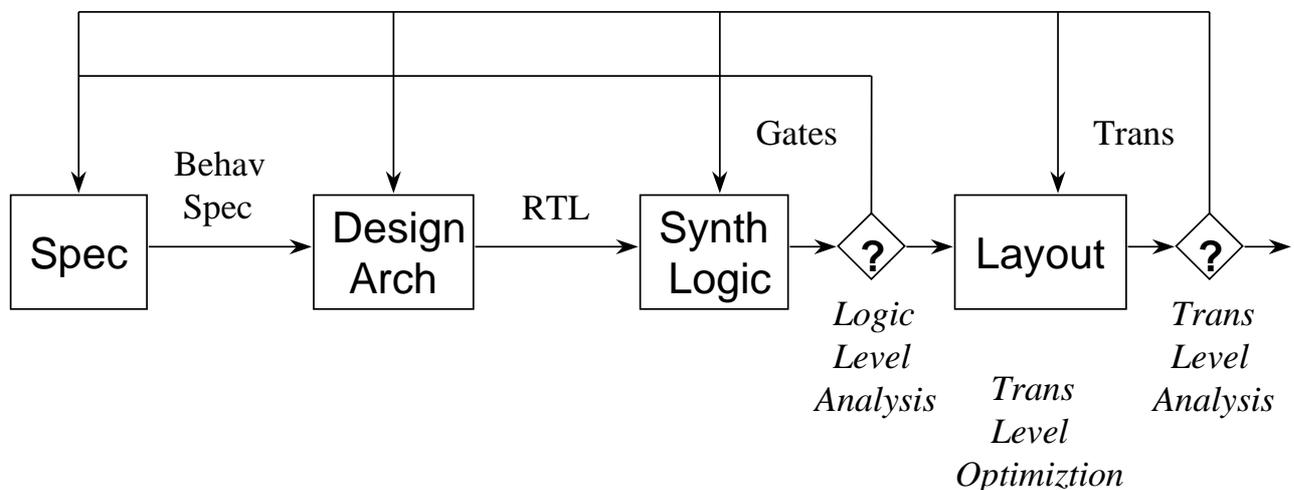
- **More Detailed Information**
 - Produces more accurate results
- **Information**
 - Produces larger improvements
 - Can be analyzed faster
 - Early information has fewer details
- **Accuracy & Run Time**
 - Requirements differ for each level
- **Analyze Early and Often**
 - At each abstraction level
- **Full Chip Analysis**
 - Each section is important



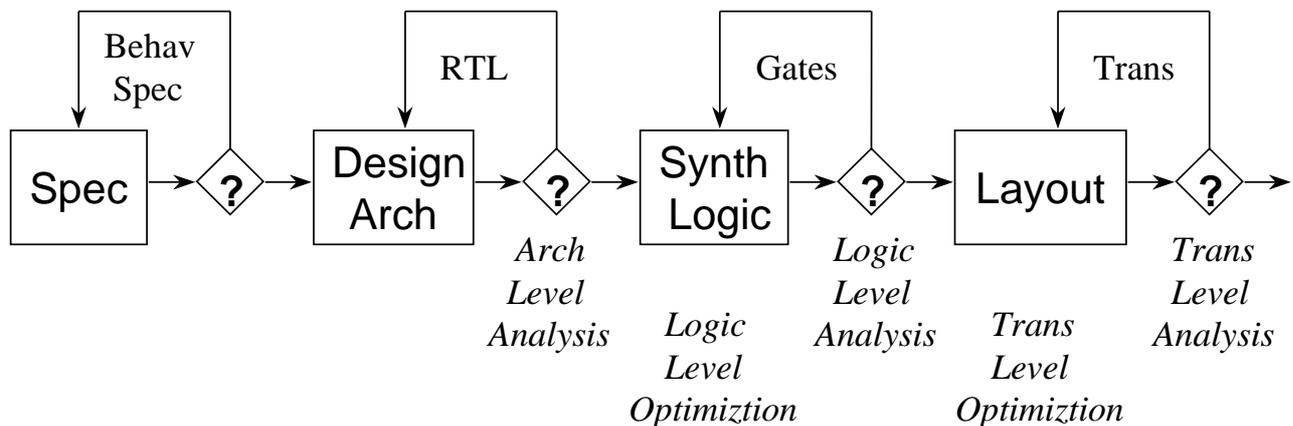
Méthodologie conventionnelle



- **Estimation au niveau logique**



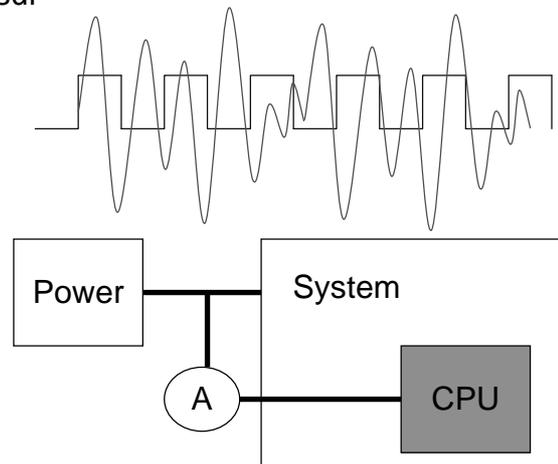
- Intervention à tous les niveaux d'abstraction



87

Estimation au niveau logiciel

- Estimation du courant consommé par le processeur, en particulier par la CPU
- Estimation de puissance pour définir des modèles au niveau des instructions
- Méthodes basées sur la simulation
 - Le programme exécuté est simulé sur un modèle bas-niveau du circuit
 - Difficile en pratique
- Mesure physique
 - Système d'acquisition
 - Mesure le courant pour des séquences d'instructions particulière



88

- **Modèle au niveau des instructions**

$$\sum_i (Base_i \cdot N_i) + \sum_{i,j} (Overhead_{i,j} \cdot N_i) + \sum_k Energy_k$$

- **Mesure sur des séquences d'instructions : Base_i**

- Instructions seules en boucle
- Séquence d'instructions

SPARClite

Instruction	Courant mA	Cycles	Energie nJ
NOP	198	1	3.26
LD	213	1	3.51
ST	346	2	11.40
ADD	199	1	3.28
MULT	198	1	3.26

- **Effet inter-instructions : Overhead_i**

- État précédent du circuit influe sur l'énergie d'une instruction
- e.g. 486DX2
XOR BX,1
ADD RX,DX : overhead 6.8 mA

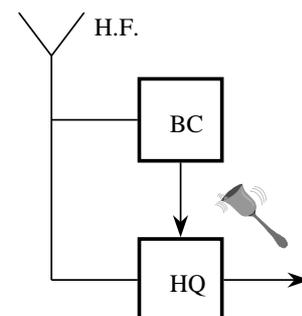
- **Pipeline, *cache misses* : Energy_k**

89

Et tout le reste ?

Ce qui n'a pas pu être traité :

- **Basse consommation analogique**
 - bibliothèques optimisées BT-BC.
 - réduction de la tension d'alimentation
 - compromis entre linéarité et consommation
 - systèmes primaire et secondaire de veille
 - modes de transmission
- **Consommation sur les cartes (PCB ou MCM)**
- **Composants standards BC du commerce**
 - CAN / CNA
 - microcontrôleurs, DSP
 - Emetteur / Récepteur H.F.
 - de plus en plus ...
- **Autres composants à l'étude**
 - Batteries rechargeables Lithium ou Polymères
 - Convertisseurs DC-DC
 - Nouvelles technologies d'écran basse consommation
 - Technologies de disques durs



90

Et tout le reste ?



Ce qui n'a pas pu être traité :

- **Evidement le logiciel**
 - Il n'y a pas que le matériel qui consomme
 - Estimation de la puissance dissipée par le code d'une application
 - Optimisation au niveau système de très grande efficacité
- **Outils d'analyse de puissance**
 - Exemple Synopsys
 - Design power (analyse)
 - Bibliothèque
 - Méthode par simulation ou probabiliste
 - Power Compiler : Logic Optimizer

91

Conclusions



Est il nécessaire de convaincre ?

- **Puissance et/ou énergie deviennent la première contrainte de conception**
- **Consommation doit être optimisée à tous les niveaux, en particulier au niveau système**
- **Diminution de la tension d'alimentation en conservant les performances**
- **Diminution de l'activité de commutation des signaux internes ou externes**

92