

Architectures reconfigurable

Application au récepteur RAKE

Jean-Luc Danger



Plan

- ⇒  **Principes d'architectures reconfigurables**
-  **Exemple : récepteur RAKE et annulateur d'interférences**

Contraintes des récepteurs radio mobiles

📄 **Coût de production minimal**

📄 **Intégration de fonctionnalités multi-standards multi-modes**

☐ => *flexibilité*

📄 **Arrivée rapide sur le marché**

☐ => *flexibilité*

📄 **Consommation minimale**

☐ => *solution 100% sw sous optimale*

Solutions

Approche Radio Logicielle (Software Radio)

- *Traitement du signal le plus possible dans le domaine numérique (En fait SDR Software Defined Radio)*

Adaptativité

- *Possibilité de modifier à la volée les valeurs numériques des paramètres*
 - Changement du nombre de pas dans l'algorithme LMS

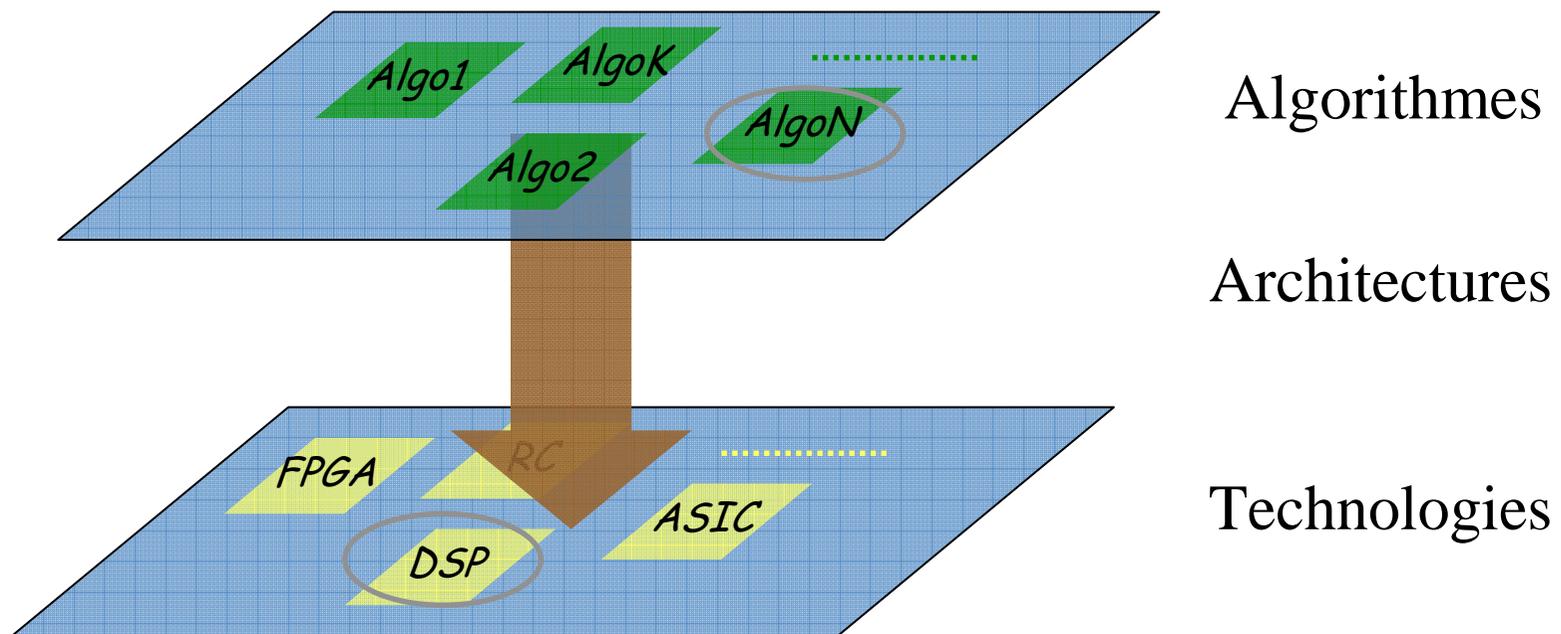
Reconfigurabilité

- *Possibilité de modifier à la volée une fonction numérique par un changement non quantifiable, c'est-à-dire sans représentation suffisante par une valeur numérique*
 - Passage d'un code convolutionnel à un turbo-code

Niveaux de Reconfigurabilité

3 Niveaux :

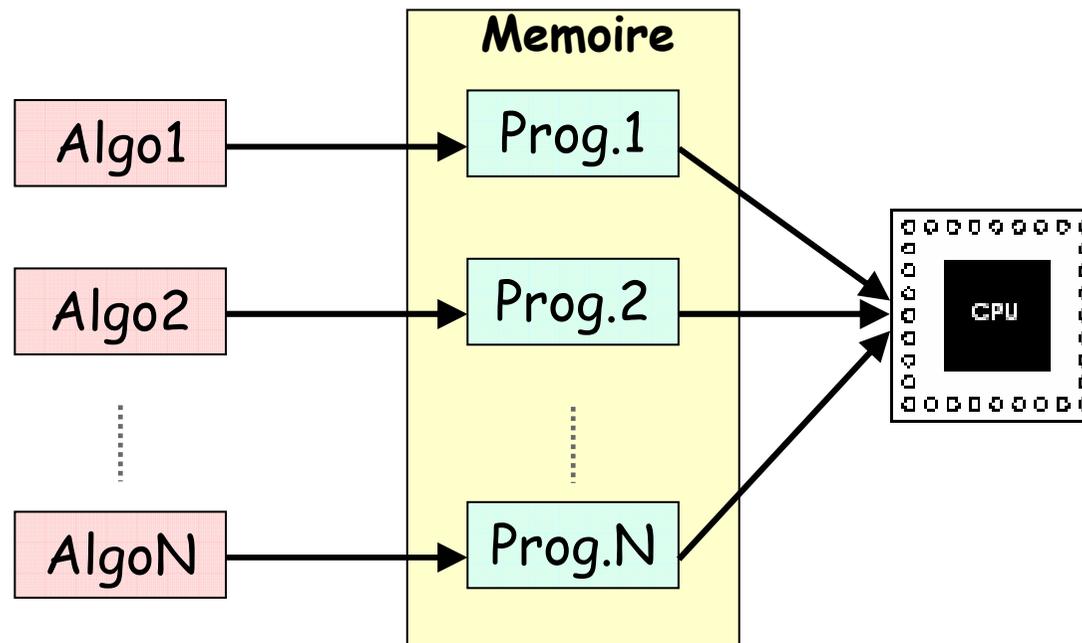
- ❑ *Algorithmique* ($f \rightarrow g$)
- ❑ *Architecture* : lien entre les algorithmes et les technologies
- ❑ *Technologique* (DSP, ASIC, FPGA)



Approche commutation fonctionnelle

📄 Approche classique adaptée à la technologie logicielle

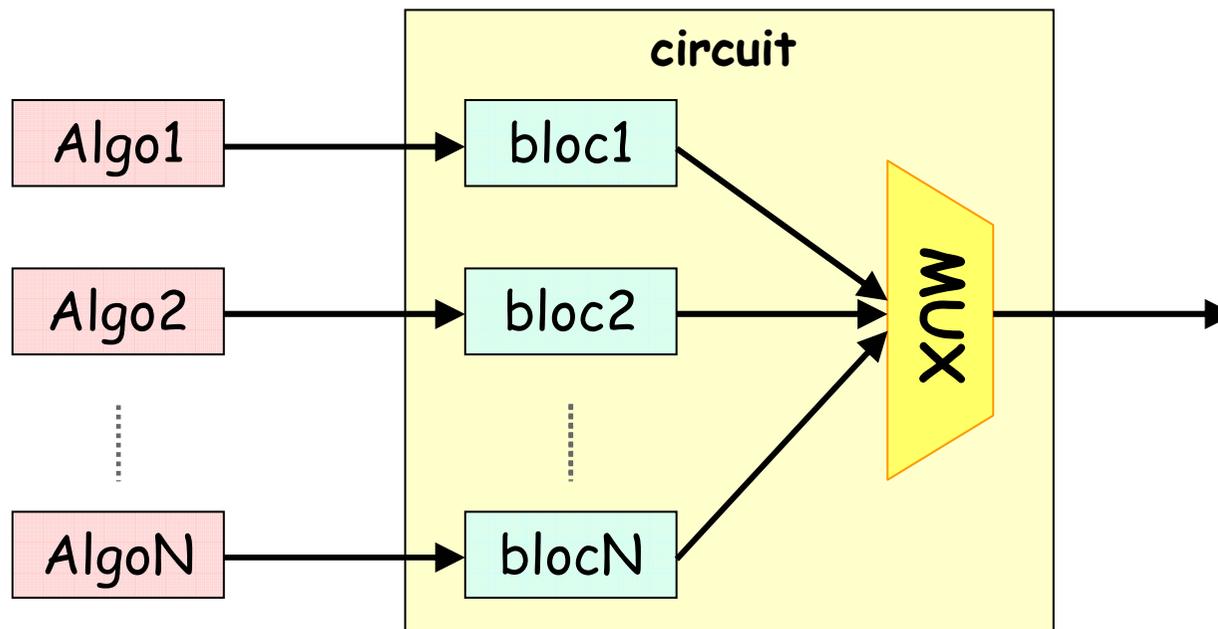
- ☐ + : *grande flexibilité*
- ☐ - : *Vitesse et consommation moindre*



Approche multiplexage

Approche réservée aux technologies ASICs avec l'exploitation de blocs préconçus

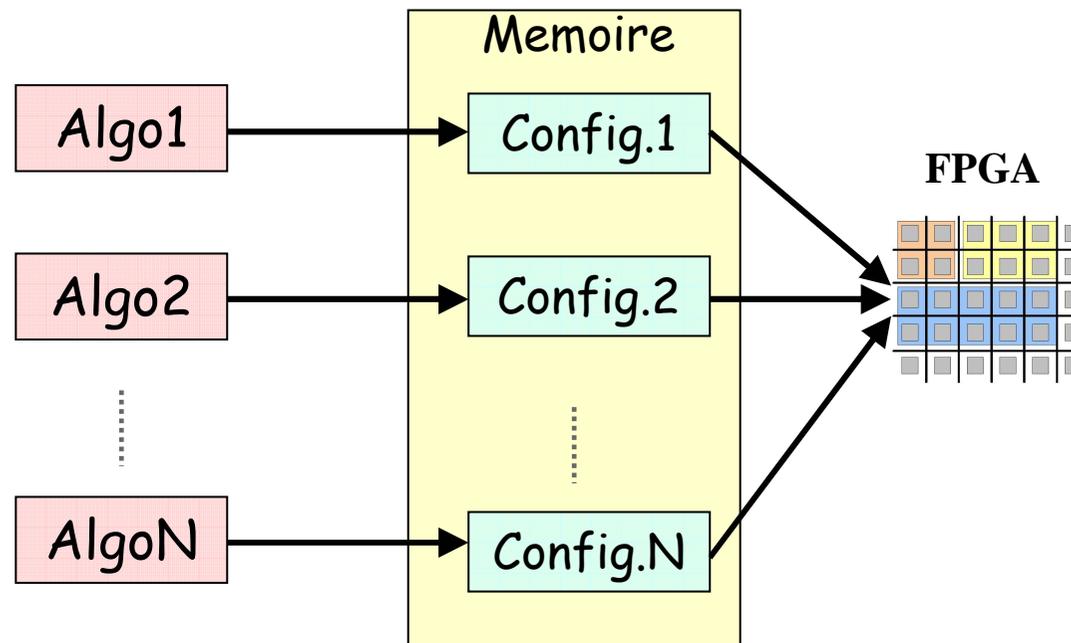
- + : *Conception rapide d'un système multi-standard et respect des performances*
- - : *Une grande partie du matériel peut ne pas être utilisé*



Approche pagination

Approche réservée aux technologies programmables

- + : *matériel optimisé en concevant un FPGA dédié*
- - : *le temps de programmation et la mémoire doivent être petits*
Les FPGAs du commerce ne sont pas adaptés pour respecter les contraintes de performances

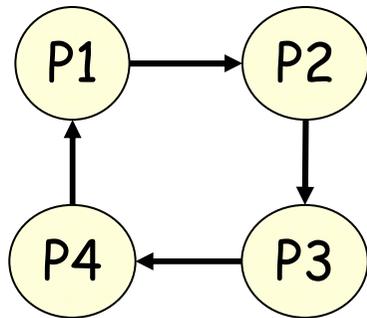


Approche factorisation

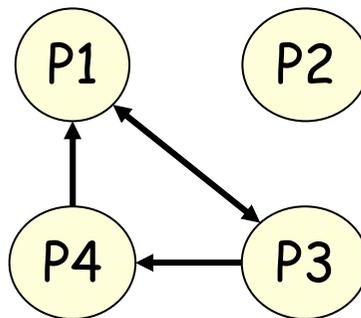
📄 Mise en facteur des opérateurs communs

☐+ : *optimisation des ressources*

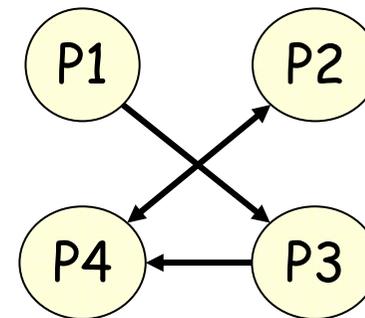
☐- : *les algorithmes doivent être assez proches*



Algo 1



Algo 2



Algo 3

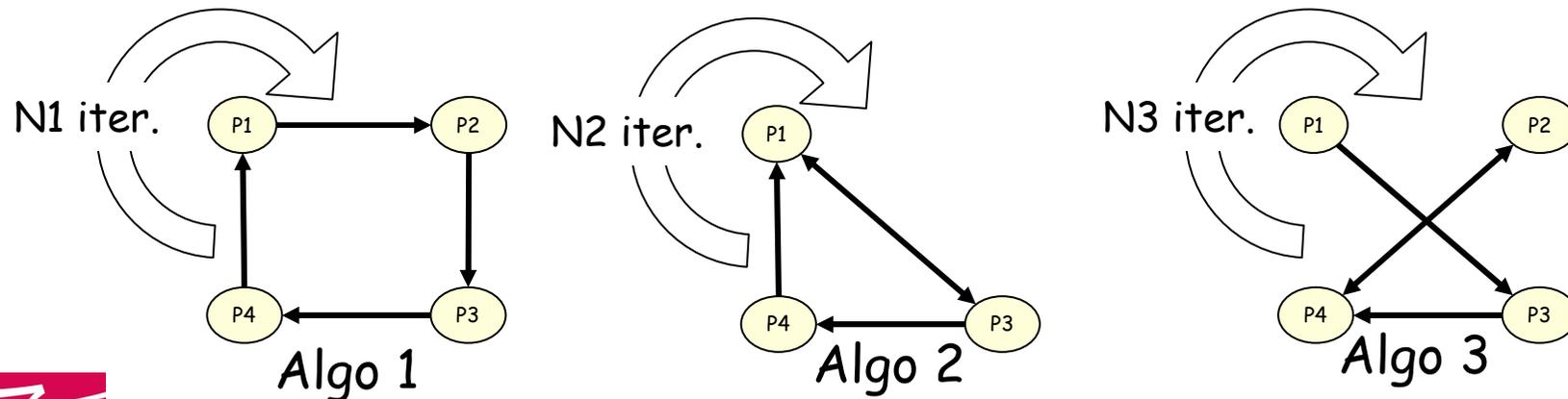
Approche itération

Approche factorisation avec grains de calculs petits et calculs itératifs

□ + :

- Optimisation importante du matériel (grain fin)
- Rajoute un dimension de reconfigurabilité : nombre d'itérations

□ - : *S'applique aux algorithmes assez proches et itératifs*



Approche itération(2)

Adaptée aux applications des systèmes de communications

- ❑ *Nombreuses fonctions facilement sérialisables ou de natures itératives (décodeurs turbo-codes et LDPC, égalisation, etc)*

Consommation :

- ❑ *Consommation dynamique légèrement supérieure*
- ❑ *Consommation statique inférieure (pour les technologies < 100nm)*

Ajustement du nombre d'itérations par le temps imparti pour le traitement

- ❑ *En considérant qu'une itération peut durer un cycle d'horloge (pipeline), le rapport entre la fréquence d'horloge f_{ck} et le débit binaire R offre un nombre important d'itérations*

Plan

 **Principes d'architectures reconfigurables**

→  **Exemple : récepteur RAKE et annulateur d'interférences**

Contexte

☰ Réception de type DS-CDMA (Direct Sequence Code Division Multiple Access)

- ❑ *Utilisée dans les standards de téléphonie mobile de 3ème génération 3G*
- ❑ *Utilise un étalement de spectre variable en fonction du service*

☰ Service très haut débit

- ❑ *Facteur d'étalement faible (2 ou 4)*
- ❑ *Probabilité de n'avoir qu'un seul utilisateur grande (interférences MAI très faibles)*

☰ **Les interférences entre symboles liées à un canal avec de nombreux trajets IPI (Inter Path Interference) sont souvent très importantes et peuvent dégrader significativement les performances du système**

Canal de propagation multi-trajets

- Transmission d'une station de base vers un terminal (*downlink*)

$$r(t) = \sum_{m=1}^M \sum_{l=1}^L \alpha_l(m) b(m) s(t - mT - \tau_l) + n(t)$$

- ▶ $r(t)$ est le signal en bande de base reçu
- ▶ M est la longueur de la fenêtre d'observation en symboles
- ▶ L est le nombre de trajets
- ▶ $\alpha_l(m)$ est un nombre complexe représentant le facteur d'évanouissement
- ▶ $b(m) \in \{\pm 1, \pm 1j\}$ est le symbole QPSK transmis
- ▶ $\tau_l \in [0 \ T]$ est le retard de propagation du trajet l
- ▶ T est la durée du symbole
- ▶ $n(t)$ est le bruit blanc additif gaussien

Récepteur RAKE(1)

- Dans l'équation $r(t)$, le signal $s(t)$ est étalé avec le facteur d'étalement SF (*spreading factor*) sur SF chips

$$s(t) = \sum_{n=0}^{SF-1} c(n)h(t - nT_c)$$

- ▶ T_c est la durée d'un chip
- ▶ $c(n)$ est la valeur du chip n
- ▶ $h(t)$ est une impulsion chip de durée T_c
- Le récepteur RAKE est chargé de
 - ▶ Désétalement le signal $s(t)$ et ce sur les L chemins parcourus
 - ▶ Recombiner les résultats des L chemins

Récepteur RAKE(2)

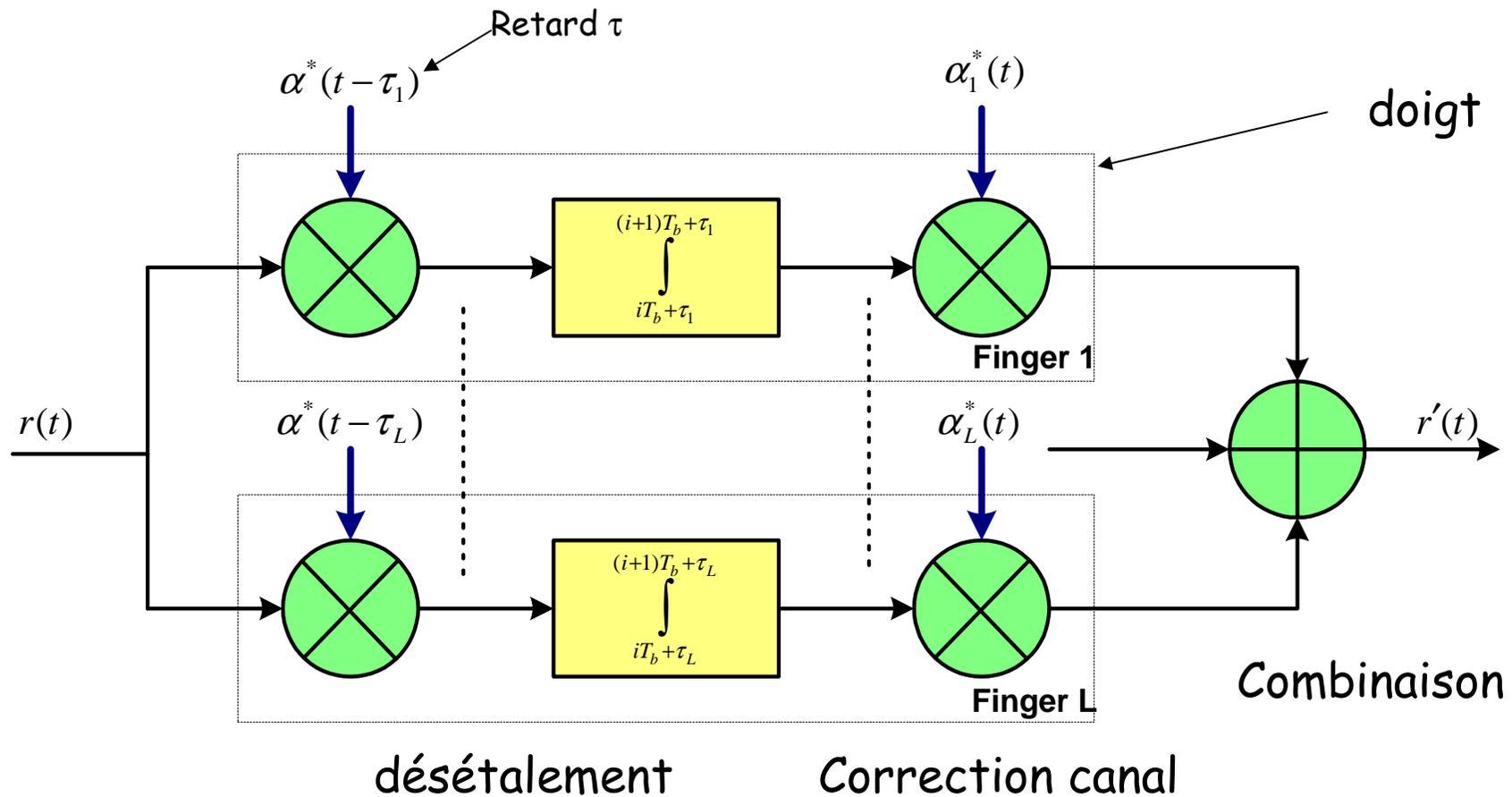
- En supposant une détection cohérente, une connaissance parfaite du canal et une recombinaison des *doigts* du Rake par l'algorithme MRC (*Maximum Ratio Combining*), le signal issu du récepteur RAKE peut s'exprimer par :

$$\tilde{b}(m) = \sum_{l=1}^L \alpha_l^*(m) \int_{(m-1)T+\tau_l}^{mT+\tau_l} r(t)s(t-\tau_l) dt = W + I + N \quad (4)$$

où

- ▶ W est le signal démodulé
- ▶ I est une composante d'interférence IPI
 - Ce terme vient du fait qu'il est impossible de concevoir des codes d'étalement de spectre parfaitement orthogonaux pour tous les décalages de temps
- ▶ N est le bruit

Architecture du RAKE



Equations

Recombinaison : Gain de diversité

$$W = \sum_{l=1}^L |\alpha_l(m)|^2 b(m)T \quad (5)$$

interférences : destructeur

$$I = \sum_{l=1}^L \sum_{q=1, q \neq l}^L \alpha_l(m) \alpha_q^*(m) \int_{(m-1)T + \tau_l}^{mT + \tau_l} b(t - \tau_l) s(t - \tau_l) dt$$

$$= \sum_{l=1}^L \sum_{q=1, q \neq l}^L \alpha_l(m) \alpha_q^*(m) \left[b(m - 1 - \delta_{l,q}) R(\tau_{l,q} - \delta_{l,q}T) + b(m - \delta_{l,q}) \hat{R}(\tau_{l,q} - \delta_{l,q}T) \right] \quad (6)$$

$$N = \sum_{l=1}^L |\alpha_l(m)|^2 \int_{(m-1)T + \tau_l}^{mT + \tau_l} \quad (7)$$

Bruit gaussien : destructeur

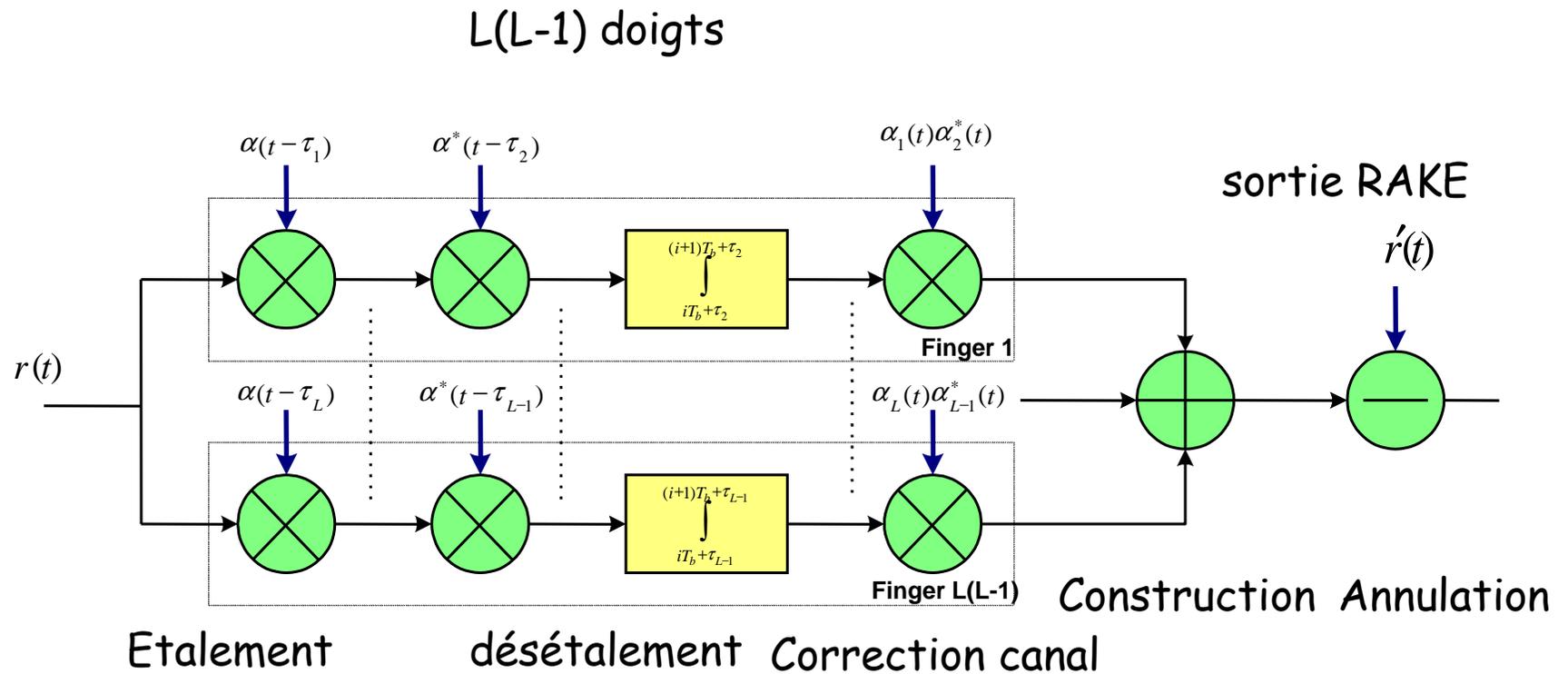
où

$$\delta_{l,q} = \left\lfloor \tau_l - \frac{\tau_q}{T} \right\rfloor \quad \tau_{l,q} = \tau_l - \tau_q \quad R(\tau) = \int_0^T s(t - \tau) dt \quad \hat{R}(\tau) = \int_{\tau}^T s(t - \tau) dt$$

Annulation d'interférences

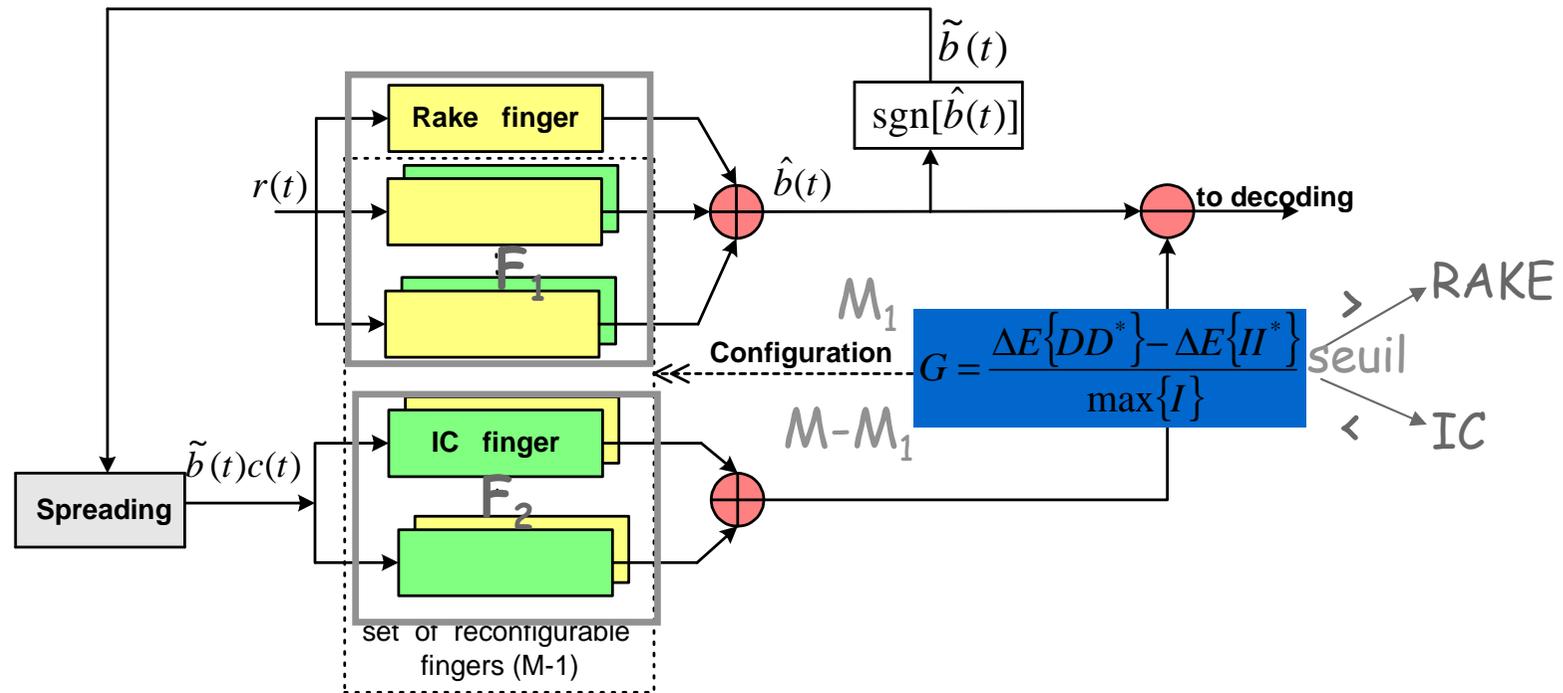
- ☰ **L'annulation d'interférences IC (Interference Cancellation) est basée sur la connaissance des générateurs de bruit que constituent les autres utilisateurs dans le cas du MAI ou les autres symboles dans le cas de l'IPI**
- ☰ **Le principe est de reproduire l'interférence de façon à la soustraire par la suite**
 - *Il est possible de réitérer le processus en utilisant comme signal d'entrée, le signal déjà traité lors d'une première itération*
- ☰ **Dans notre exemple, l'annulation d'interférences est appliquée pour supprimer l'IPI avec un seul utilisateur et pour les hauts débits**

Architecture de l'annulateur d'interférences IPI



=> Architecture très proche de celle du RAKE

Cas 1 : Reconfiguration des doigts entre RAKE et IC



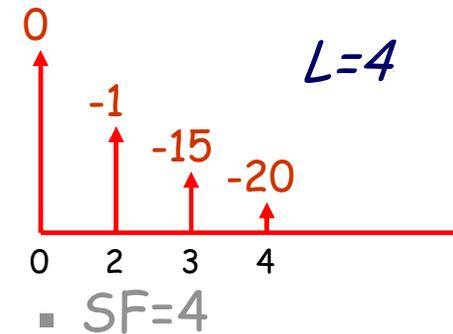
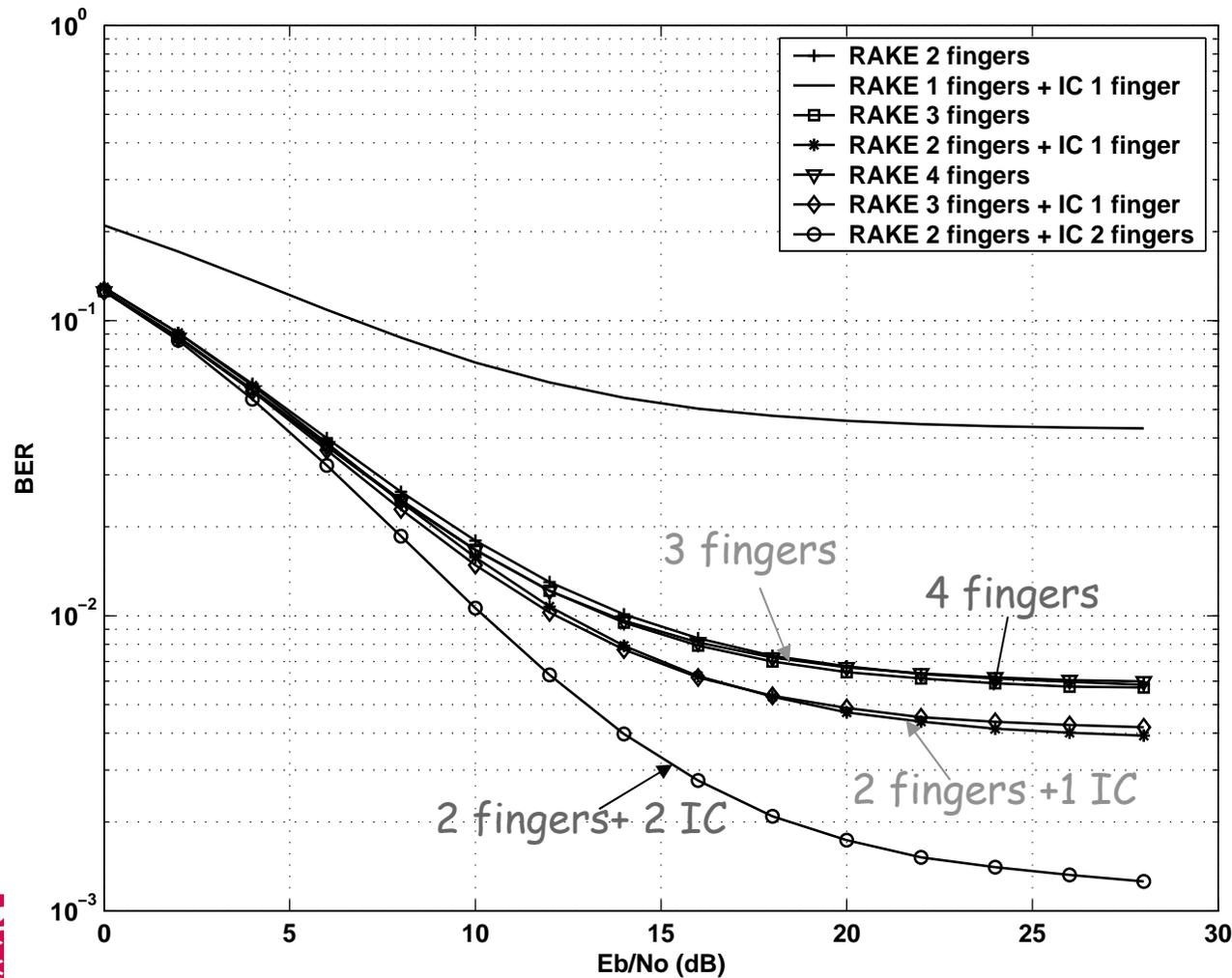
$$\hat{b}(k) = \sum_{m=1}^{M_1} \alpha_m(k) \int_{(k-1)T_b + \tau_m}^{kT_b + \tau_m} r(t)c(t - \tau_m) + n(t) = D + I + n(t)$$

$$I = \sum_{m=1}^{M_1} \sum_{\substack{l=1 \\ l \neq m}}^L \alpha_m^*(k) \alpha_l(k) \int_{(k-1)T_b + \tau_m}^{kT_b + \tau_m} b(t - \tau_l)c(t - \tau_l)c(t - \tau_m) dt$$

$$\tilde{b}(k) = f_{dec}(\hat{b}(k)) = \text{sgn}(\hat{b}(k))$$

$$\hat{b}'(k) = \hat{b}(k) - \hat{I}^{(M-M_1)}(\tilde{b}(k))$$

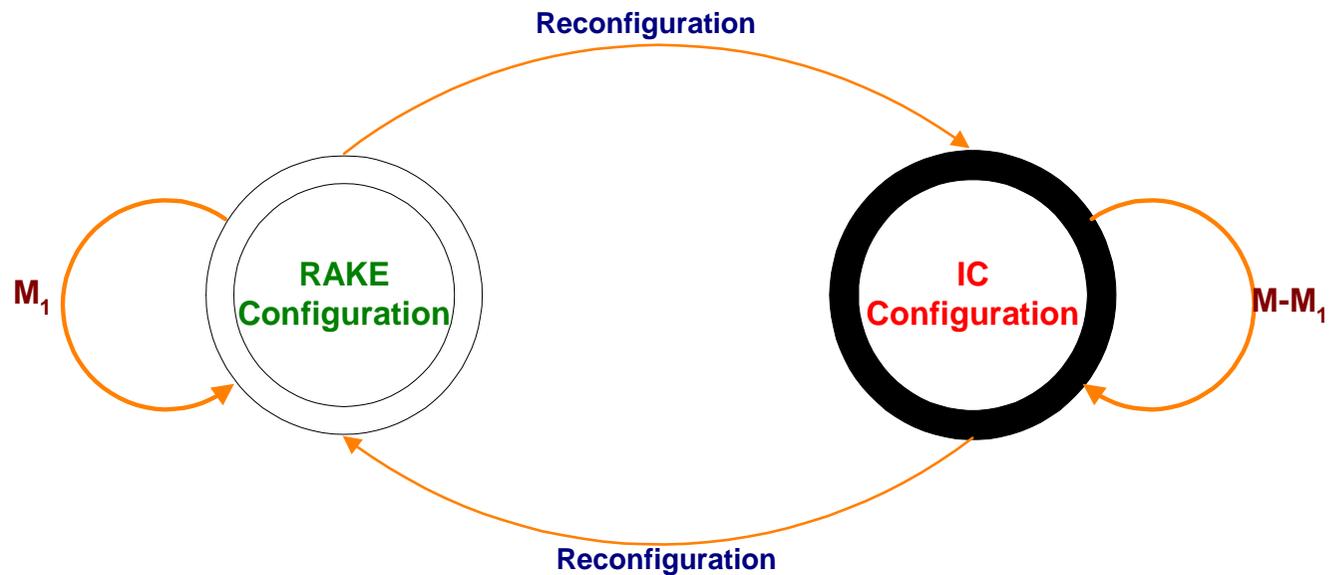
Cas 1 : Gain en performances de la reconfiguration



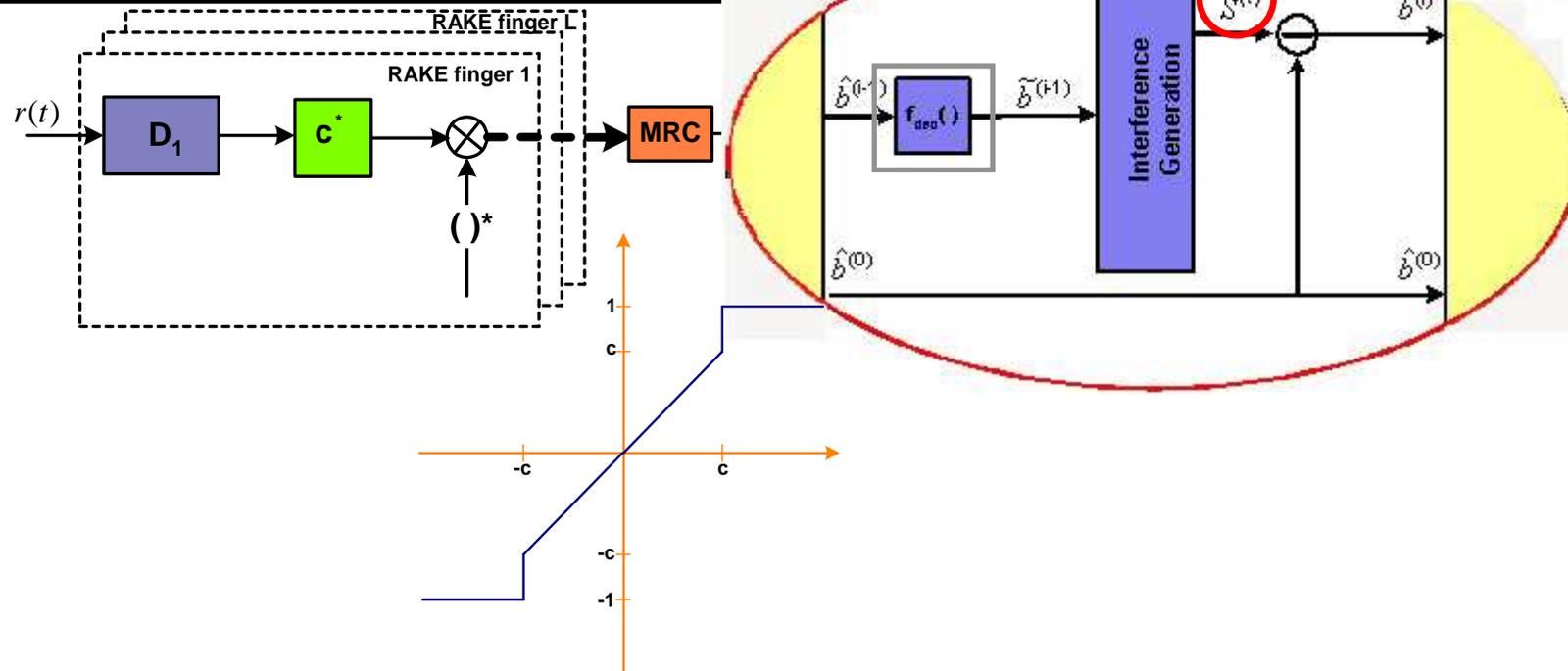
M	RAKE	IC
2	2	0
3	2	1
4	2	2

Cas 1 : graphe de reconfiguration

- RAKE $\rightarrow M_1$ iterations (F_1)
- IC $\rightarrow M_2 (\leq M - M_1)$ iterations (F_2)



Cas 2 : IC multi-étages



$$\hat{b}^{(0)}(k) = \sum_{m=1}^M h_m^*(k) \int_{(k-1)T_b + \tau_m}^{kT_b + \tau_m} r(t) c(t - \tau_m) dt = D + S + W$$

Rake

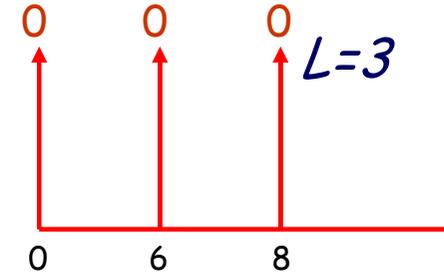
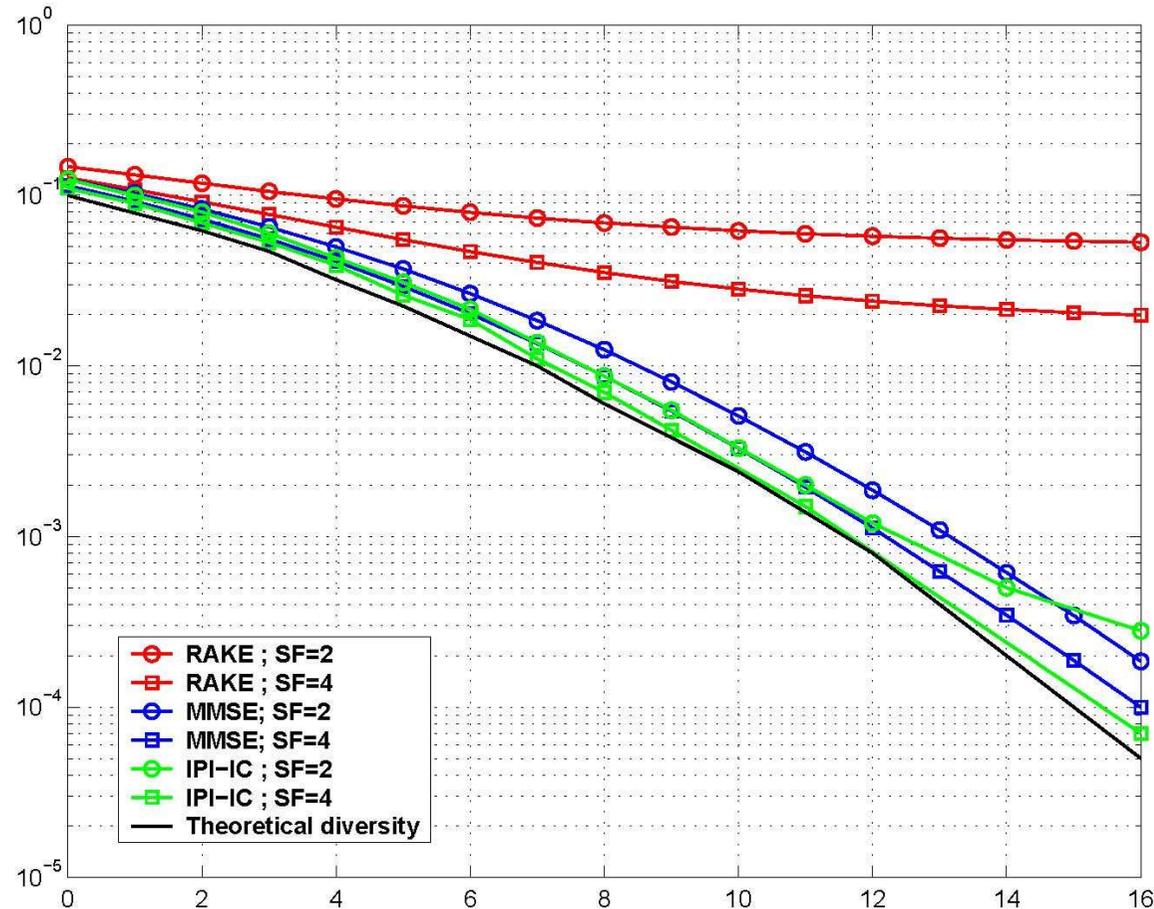
$$\tilde{b}^{(i-1)}(k) = f_{dec}(\hat{b}^{(i-1)}(k))$$

Decision

$$\hat{b}^{(i)}(k) = \hat{b}^{(0)}(k) - \hat{S}^{(i)}(k)$$

Annulation

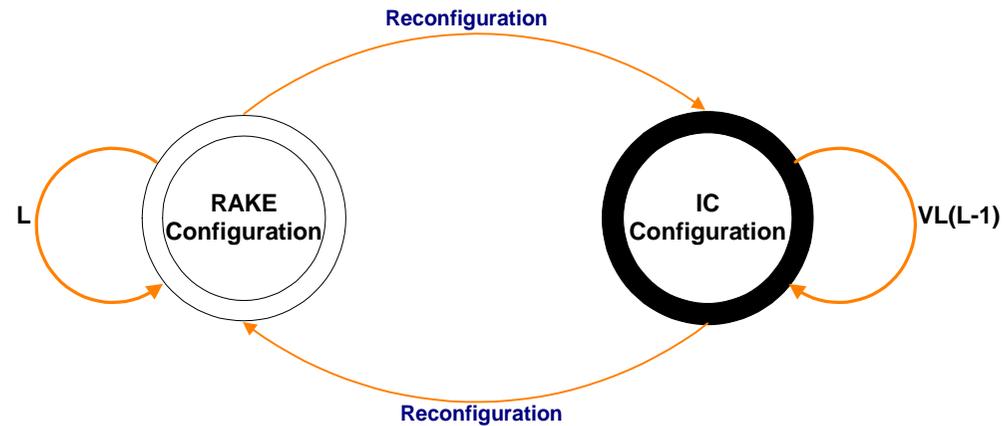
Cas 2 : augmentation des performances



■ $V = \text{nombre d'étages} = 5$

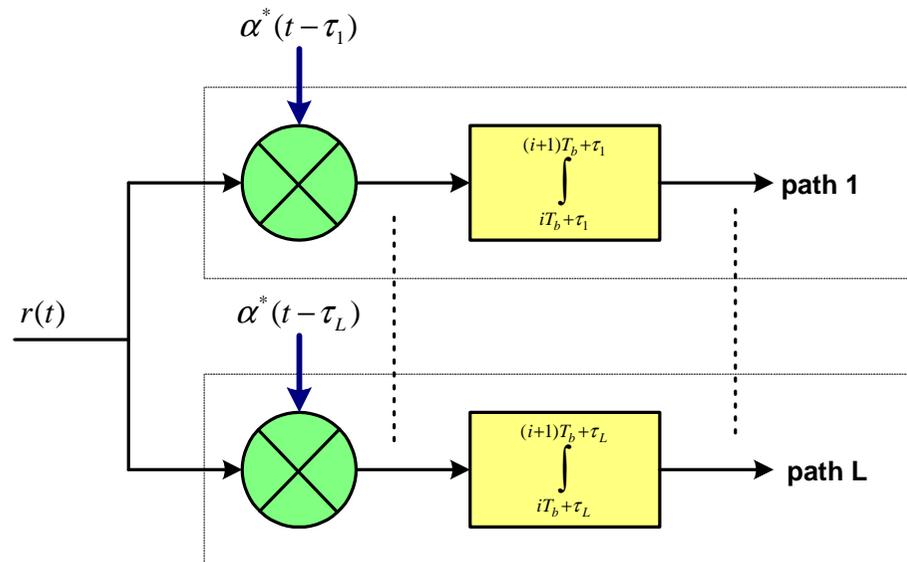
Cas 2 : graphe de reconfiguration

- RAKE demodulation $\rightarrow L$ iterations (F_1)
- Interference Cancellation $\rightarrow VL(L-1)$ iterations (F_3)



Cas 3 : Estimation de canal par canal pilote

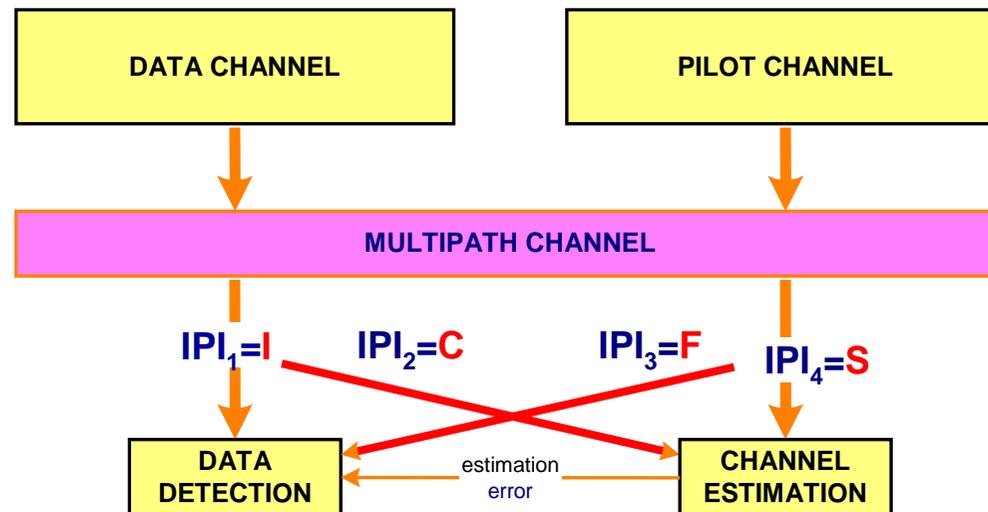
- Dans certains standards 3G, il existe avec le canal **données**, un canal **pilote** permettant d'estimer le canal au niveau de chaque symbole et chaque trajet (préalablement estimé).
- Architecture de l'estimateur :



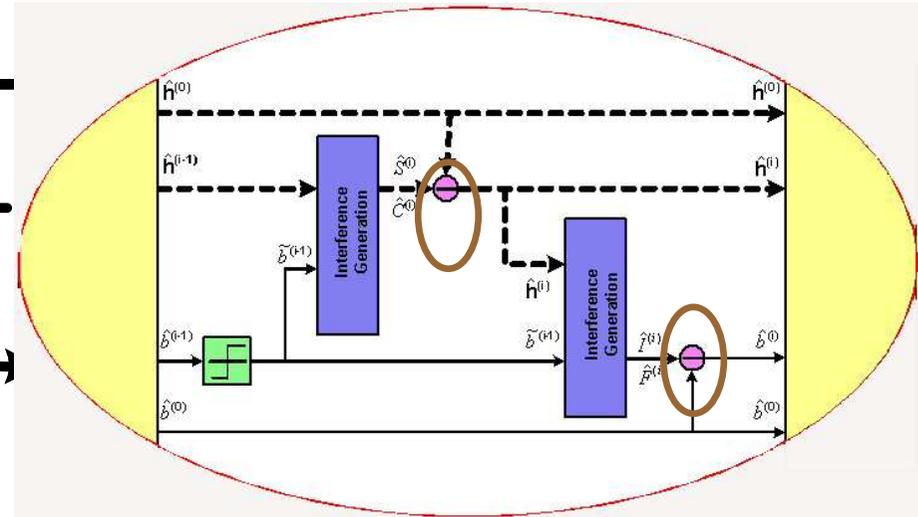
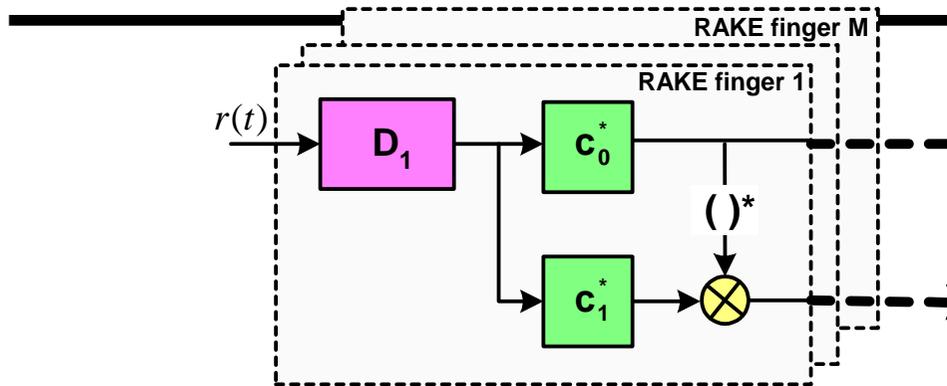
=> Architecture très proche de celle du RAKE

Cas 3 : IC multi-étages avec estimation de canal

Un schéma IC multi-étages avec estimation de canal permet à chaque étage de raffiner la connaissance des données et du canal en diminuant les 4 sources interférences du RAKE ($D \leftrightarrow D, D \leftrightarrow P, P \leftrightarrow D, P \leftrightarrow P$)



Cas 3 : IC multi-étages avec estimation de canal



Rake

$$\tilde{b}^{(0)}(k) = \sum_{m=1}^M \hat{h}_m^*(k) \int_{(k-1)T_b + \tau_m}^{kT_b + \tau_m} r(t) c_1(t - \tau_m) dt = D + I + F + \eta$$

$$\hat{h}_j(k) = \int_{(k-1)T_b + \tau_j}^{kT_b + \tau_j} r(t) c_0(t - \tau_j) dt = h_j + S_j + C_j + \mu_j$$

$$\hat{\mathbf{h}}^{(0)}(\mathbf{k}) = \left[\hat{h}_1^{(0)}(k) \quad \dots \quad \hat{h}_M^{(0)}(k) \right]$$

Estimation de canal initiale

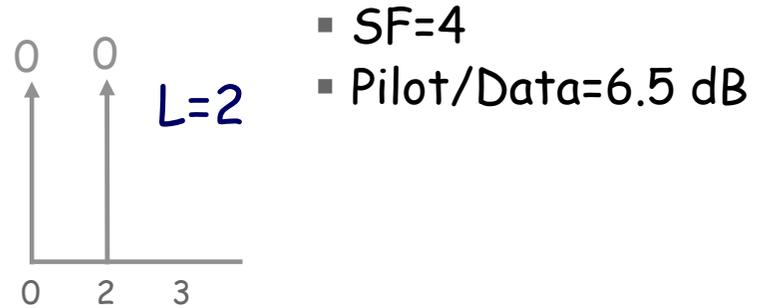
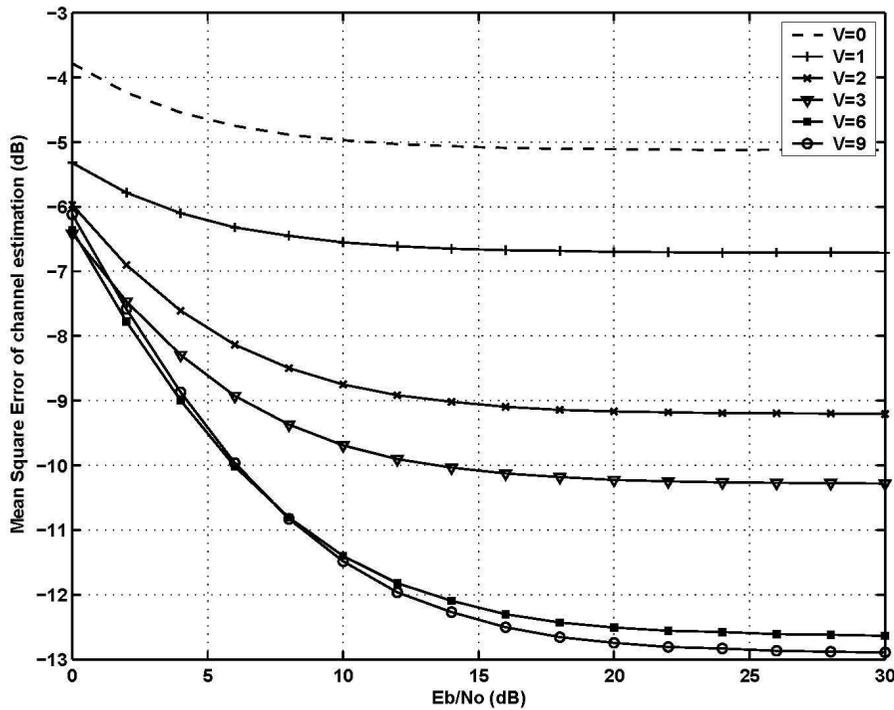
$$\hat{\mathbf{h}}^{(i)}(\mathbf{k}) = \hat{\mathbf{h}}^{(0)}(\mathbf{k}) - \hat{\mathbf{S}}^{(i)}(\mathbf{k}) - \hat{\mathbf{C}}^{(i)}(\mathbf{k})$$

Estimation de canal raffinée

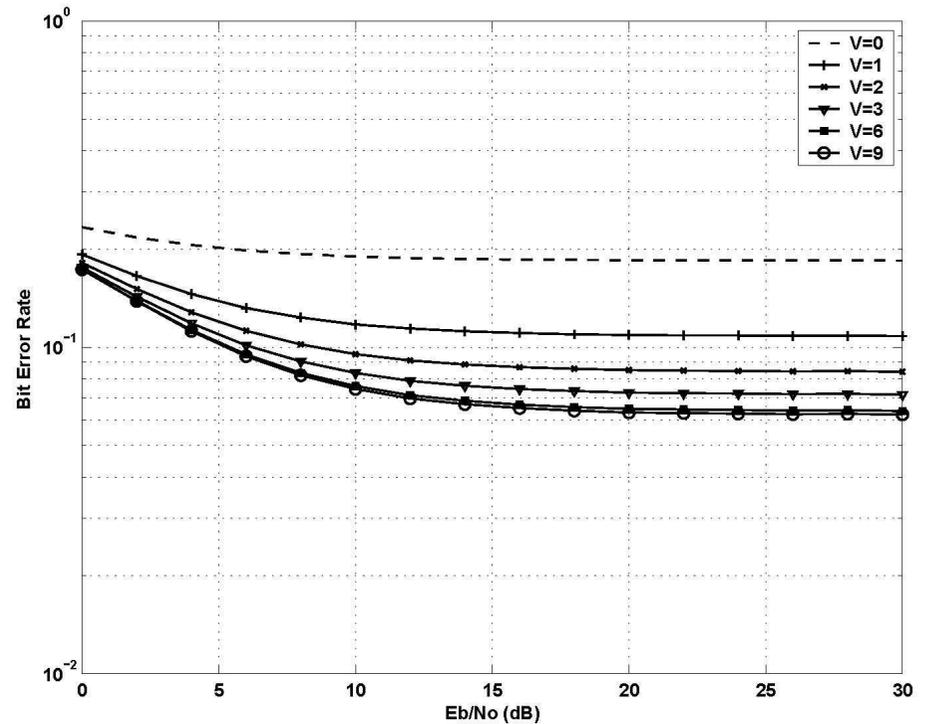
$$\hat{b}^{(i)}(k) = \hat{b}^{(0)}(k) - \hat{I}^{(i)}(k) - \hat{F}^{(i)}(k)$$

Réception améliorée

Cas 3 : augmentation des performances

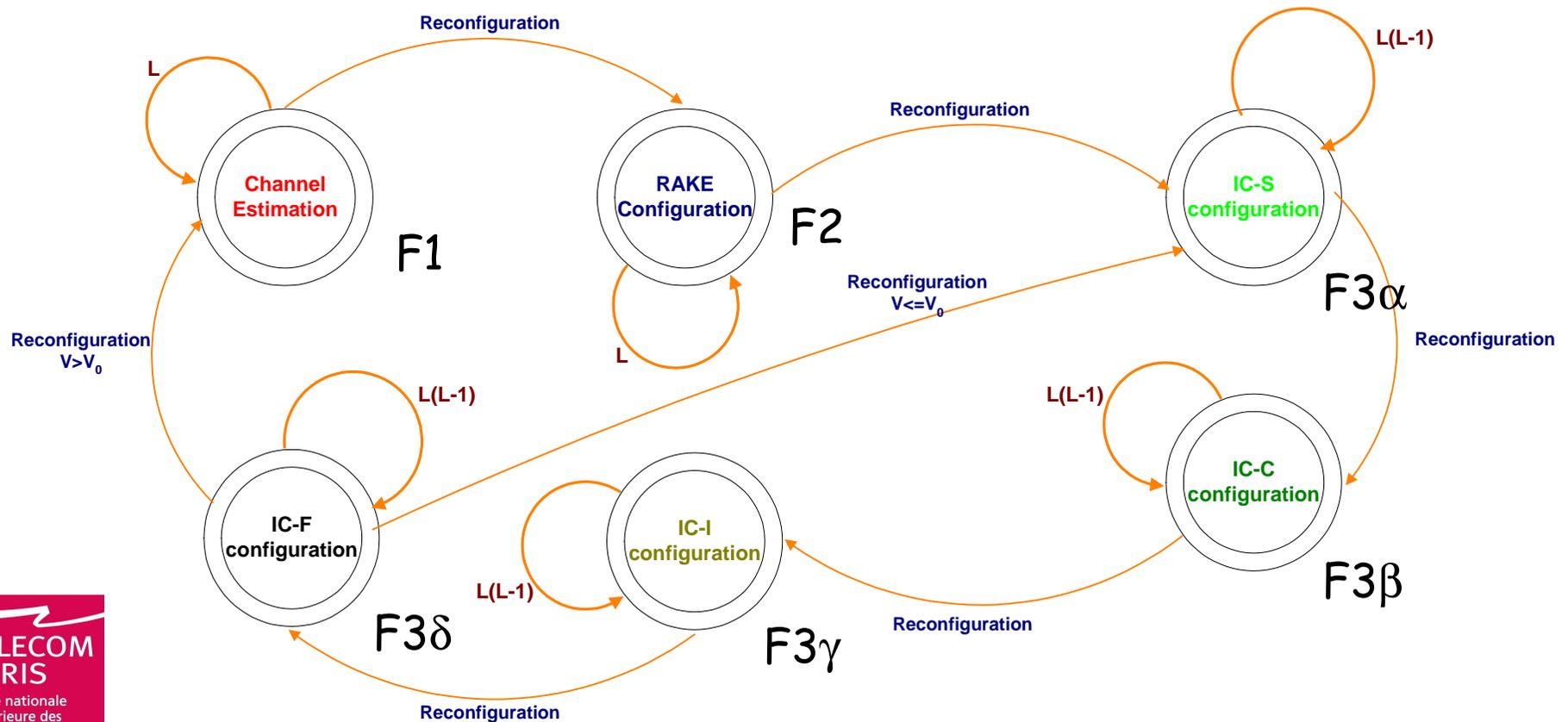


- SF=4
- Pilot/Data=6.5 dB

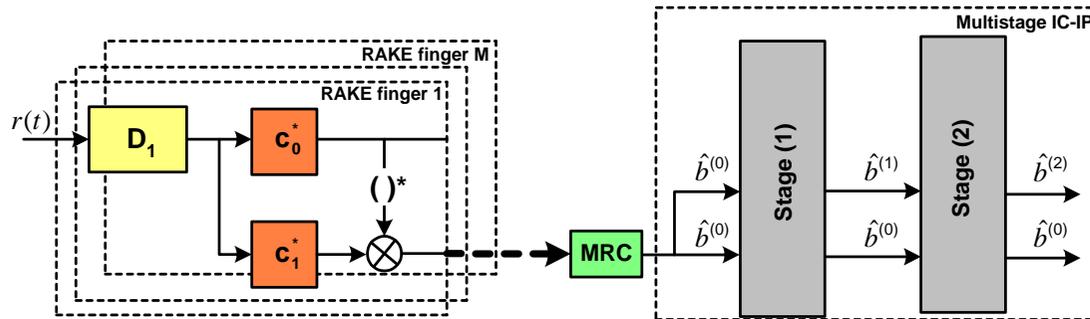


Cas 3 : graphe de reconfiguration

- Channel estimation (F_1)
- RAKE (F_2)
- IC-S (*data*→*pilot*) ($F_{3\alpha}$)
- IC-C (*multipath*→*pilot*) ($F_{3\beta}$)
- IC-I (*pilot*→*data*) ($F_{3\gamma}$)
- IC-F (*multipath*→*data*) ($F_{3\delta}$)



Implémentation du cas 2

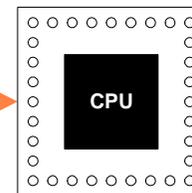
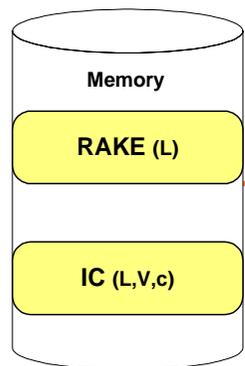


✓ Utilisation d'un DSP avec l'approche "commutation fonctionnelle"

Algorithmic Layer 2



Hardware Layer 1



(Tigersharc 250 MHz)

Nombre de chemins traités

UMTS: $T_{\text{block}} = 666 \mu\text{sec}$

Nb. paths (L)	RAKE (μsec)	IC (μsec)
1	103	-
2	208	458 (2 stages)
3	315	262 (2 terms, 1 stage)
4	362	262 (2 terms; 1 stage)
5	472	-
6	536	-
7	638	-
8	627	-

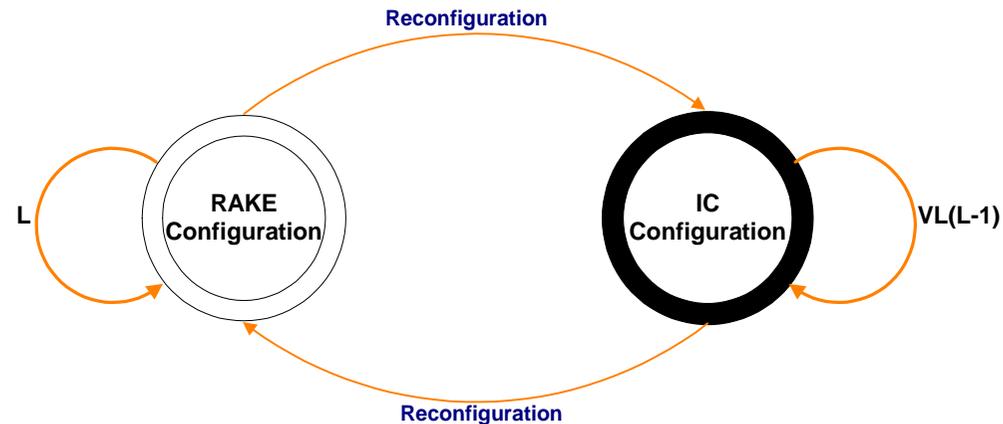


DSP pas assez performant
Si $L > 2$

- Une accélération matérielle est nécessaire
- L'approche itérative peut s'appliquer

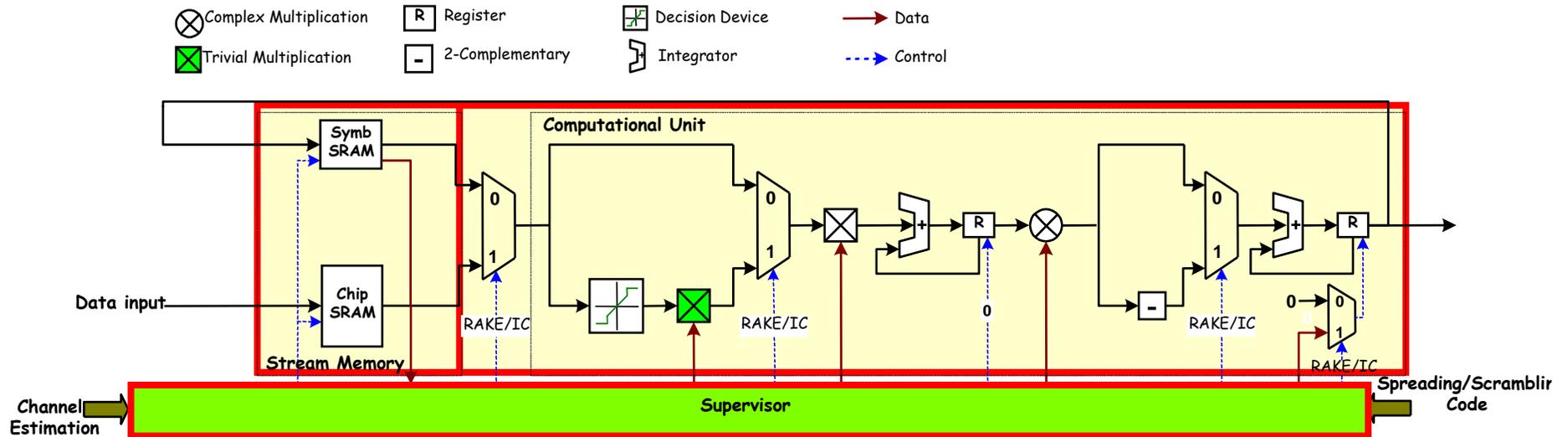
Cas 2 : graphe de reconfiguration

- RAKE demodulation $\rightarrow L$ iterations (F_1)
- Interference Cancellation $\rightarrow VL(L-1)$ iterations (F_3)



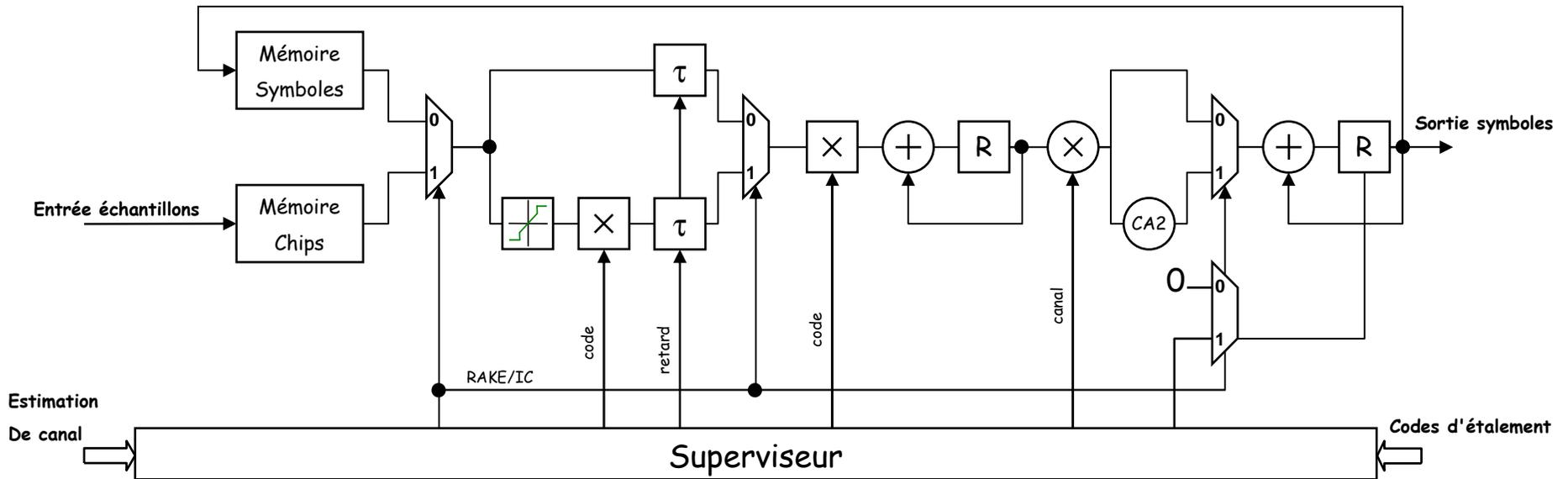
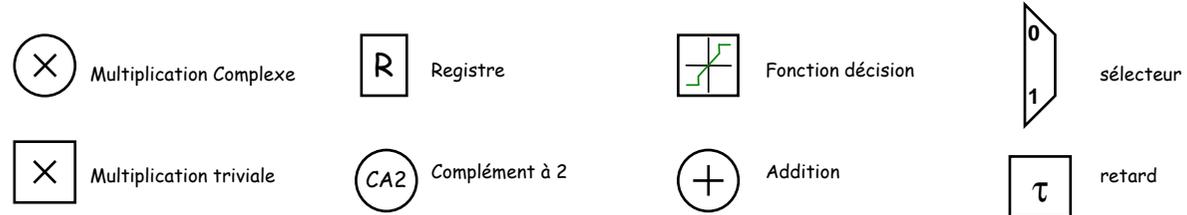
$L + VL(L-1)$ itérations

Cas 2 : Architecture de calcul

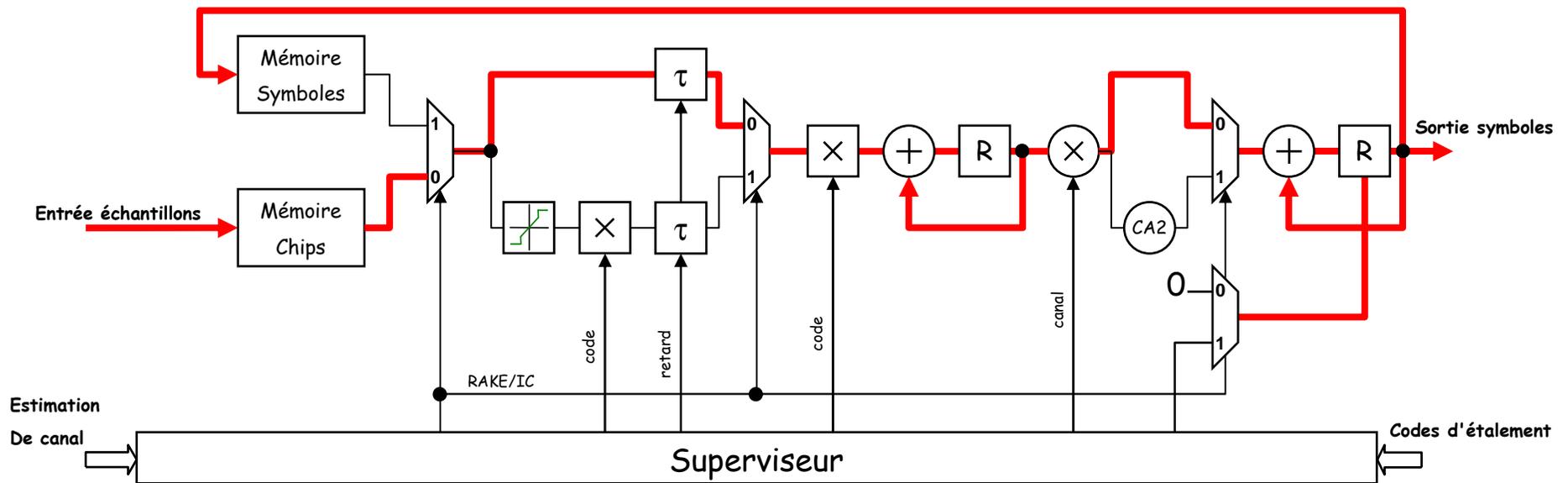
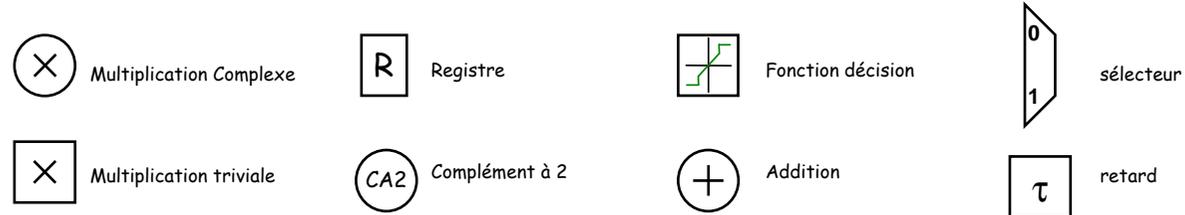


- 3 blocs :
- Mémoire
 - Calcul (RAKE/IC)
 - Superviseur

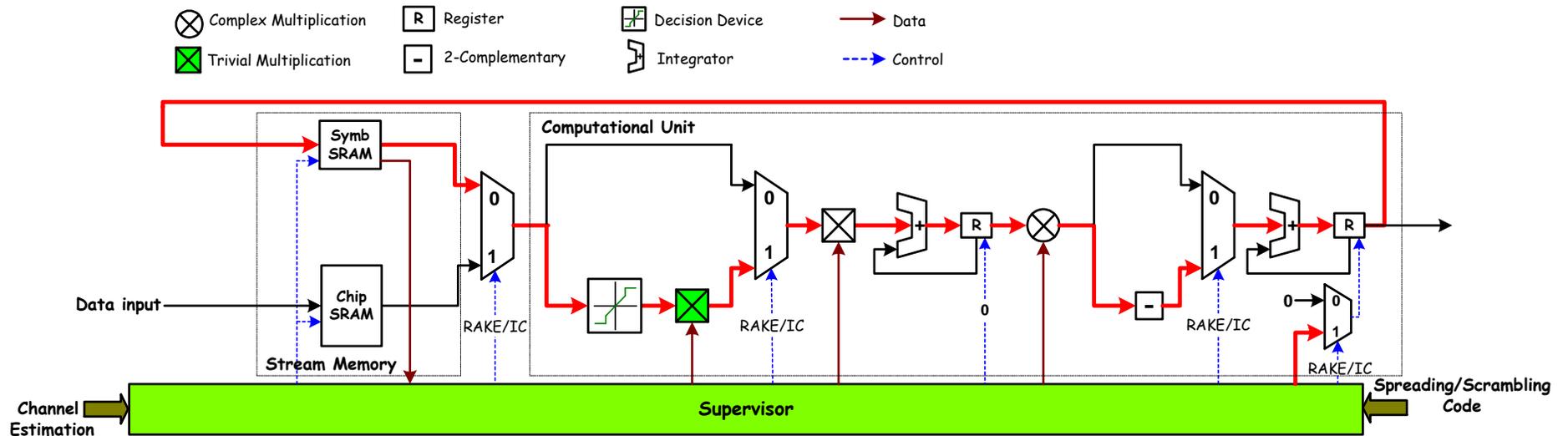
Architecture



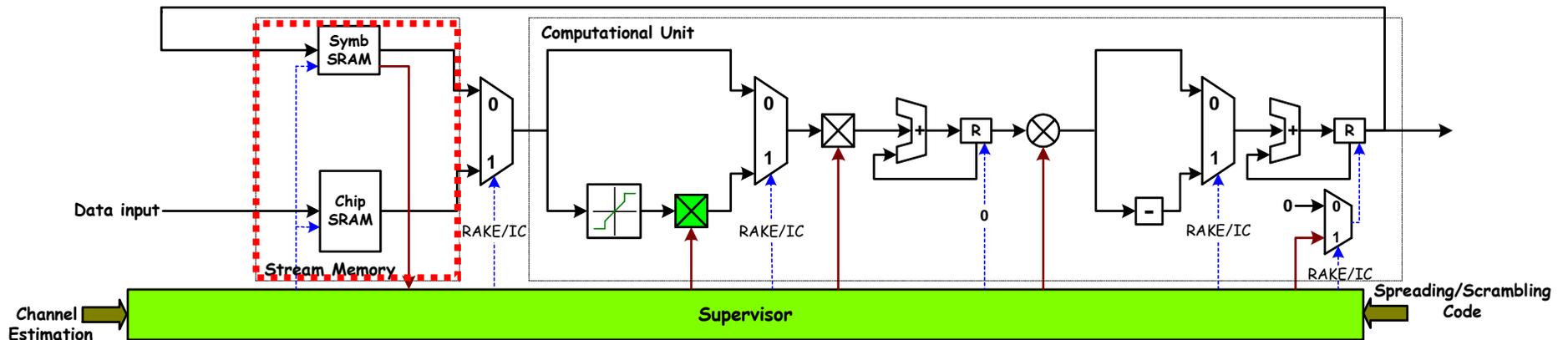
Configuration en mode RAKE



Configuration en mode IC



Bloc mémoire



$$N_{SRAM_chip} = \left\lceil \frac{T_{BLOCK} + T_{delay-spread}}{T_c} \right\rceil 2RN_{bits}$$

$$N_{SRAM_symbol} = 2 \left\lceil \frac{T_{BLOCK}}{SFT_c} \right\rceil 2N'_{bits}$$

T_{BLOCK} (chips)	MEM_{chip} (Kb)	MEM_{symb} (Kb)
256	20.7	2.05
128	10.35	1.025

R	4
T_{block}	666 μ sec
f_c	3.84 Mcps
N_{bits}	10
N'_{bits}	4

R= taux de suréchantillonnage

N= nb bits échantillons

N'= nb bits symboles

Nombre d'itérations maximum

- La reconfiguration s'effectue immédiatement
- La fréquence d'horloge est α plus grande que la fréquence chip

$$\alpha = f_{clk} / f_c$$

- Le nombre de cycles d'horloge par symbole est donc :

$$N_{cycles / symbole} = \frac{f_{clk}}{f_{symbol}} = \alpha \cdot f_c \cdot \left(\frac{SF}{f_c} \right) = \alpha \cdot SF$$

- Comme 1 itération peut être effectuée en SF cycles d'horloge (pipeline)

$$N_{itérations} = \frac{N_{cycles / symbole}}{SF} = \alpha$$

Le nombre d'étages V est donc tel que $\alpha > L + VL(L-1)$

Nombre d'itérations et complexités

✓ FPGA (Virtex4) → $f_{clk} = 168.8 \text{ MHz} \xrightarrow{168.8/3.84} 44 \text{ iterations}$

$V=3$ $L=4$ → ✓ RAKE (4 iterations)
 ✓ IC (36 iterations → 12 iterations/étage, 3 étages)

✓ ASIC (0.13μm CMOS) → $f_{clk} = 500 \text{ MHz} \xrightarrow{500/3.84} 130 \text{ iterations}$

FPGA:

Device Utilization	Percentage
Slices	501/19968 (2%)
DSP48s	2/48 (4%)

ASIC:

0.07 mm² in 130nm

