

# DIXIEME COLLOQUE SUR LE TRAITEMENT DU SIGNAL ET SES APPLICATIONS

1047



NICE du 20 au 24 MAI 1985

---

## METHODOLOGIE DE CONCEPTION DE CIRCUITS INTEGRES ADAPTES AU TRAITEMENT DU SIGNAL

H.BARRAL - N.MOREAU

ENST Dépt SYC - 46 rue Barrault - 75634 PARIS Cedex 13

---

### RESUME

On présente quelques opérateurs de base (multiplieur, diviseur, racine carrée) puis quelques opérateurs spécialisés en traitement du signal (corrélateur, filtre en treillis AR ou MA, calcul des coefficients de corrélation partiel), réalisés suivant une structure bit-série, dans le bût de constituer une bibliothèque de cellules pré-caractérisées.

On montre ensuite que l'on peut développer deux familles de circuits:

- des circuits cascadables rapides (fréquence d'échantillonnage du signal traité supérieure à 100KHz) réalisant une fonction élémentaire,
- des circuits plus lents (adaptés aux signaux de type audio-fréquence par exemple) mais ayant une fonctionnalité plus complexe.

Cette approche ne se justifiant que si on a des temps de conception très courts, il faut pour cela disposer d'outils informatiques adaptés. On indique ceux qui nous semblent nécessaires en traitant l'exemple de l'étude de l'intégrabilité d'un circuit permettant le codage de la parole par prédiction linéaire.

### SUMMARY

Some basic operators are described which include multiplication, division, square root together with operators specialized to signal processing functions such as correlation, AR and MA lattice filtering, extraction of PARCOR coefficients. These functions are implemented according to a bit-serial architecture to form a library of pre-characterized macro-cells.

We then purport to show that two families of circuits can be developed:

- first, fast cascadable circuits (for sampling rates higher than 100KHz) which perform one basic function,
- second, slower circuits (typically for audiofrequency applications) which perform, however, more sophisticated functions.

Such an approach is viable if the design stage remains short. To this end, appropriate software tools must be made available. We describe the tools which, we think, are useful taking the integrability of linear predictive coding of speech as a case study.



## 1.0 INTRODUCTION

Le traitement de signal est arrivé à un stade de maturation tel que son utilisation s'est répandue dans des domaines aussi variés que le traitement de la parole, le radar, les communications, le domaine biomédical, les images... Ces traitements, qui se font sous forme numérique pour des raisons évidentes de fiabilité, de reproductibilité, de facilité de test... concernent des signaux qui ont des fréquences d'échantillonnage de plus en plus élevées.

Le champ d'application d'un algorithme est d'autant plus large qu'il peut être réalisé à un coût réduit. Or ces algorithmes demandent une puissance de calcul très importante, qui va en s'accroissant. On peut penser que les progrès de la technologie peuvent suffire pour répondre à ces besoins de calcul (la réduction de la longueur du canal du transistor MOS entraîne des temps de propagation plus courts); en fait, les algorithmes utilisables dépassent largement ce qu'une simple approche technologique permet d'atteindre.

Lorsqu'on se pose ce problème de vitesse de calcul dans un cadre général de structure de calculateur, on est amené à introduire des concepts de parallélisme, de concurrence, de division des tâches; problèmes très difficiles à résoudre sans accepter des contraintes sur les tâches à réaliser. Dans le contexte du traitement du signal, les algorithmes possèdent intrinsèquement un assez haut degré de parallélisme ou, du moins, une structure très particulière. L'idée est donc d'utiliser pleinement la structure interne de ces algorithmes pour trouver une structure de machine adaptée à leur réalisation.

L'étude des contraintes dues au silicium, à savoir les problèmes de communication interne (bus) et externe (plots), nous a poussé à choisir une réalisation bit-série des opérateurs proposés dans ce papier. Ceci présente en outre l'avantage de permettre de profiter au maximum du parallélisme interne des algorithmes en autorisant la duplication de ces opérateurs lorsque les équations du calcul l'indiquent. La méthode de conception d'un circuit est alors très simple: il suffit d'attribuer à chaque opération écrite dans l'algorithme un opérateur et la structure de la machine calque le plus exactement possible la structure de l'algorithme. Cette démarche s'applique de façon simple aux algorithmes du type "produit scalaire".

Après avoir présenté quelques opérateurs de base puis plusieurs opérateurs plus spécialisés en traitement du signal, on montrera que l'on peut développer deux familles de circuits:

- des circuits cascadables rapides (fréquence d'échantillonnage du signal traité de l'ordre de quelques centaines de KHz) réalisant une fonction élémentaire,
- des circuits plus lents (adaptés aux signaux de type audio-fréquence par exemple) mais ayant une fonctionnalité plus complexe.

Cette approche ne se justifie que si on a des temps de conception très courts. Il faut pour cela disposer d'outils informatiques adaptés. On indiquera ceux qui nous semblent nécessaires.

Toutes les estimations de surface proposées par la suite sont faites relativement à la technologie HMOS2 (longueur de canal 2 microns).

## 2.0 OPERATEURS DE BASE

On a développé et dessiné quelques opérateurs généraux dans le but de constituer progressivement une bibliothèque. Les cellules développées sont :

### 2.1 Un Multiplieur Série-parallèle

Il a été décrit dans [1] et [2]. On rappelle ses caractéristiques:

- structure en "carry-save" avec algorithme de Booth-modifié,
- entrée série pour les deux opérands,
- sortie série pour le résultat à la même fréquence que l'entrée des opérands (un bit à chaque période d'horloge),
- nombre de bits du résultat fixé par un contrôle externe et non par la taille des opérands,
- débit égal à celui d'un additionneur série, ce qui simplifie la connexion de ces deux opérateurs dans le cas de la réalisation d'un multiplieur-accumulateur.

Le dessin de ce multiplieur est très régulier puisqu'il est composé d'un certain nombre d'additionneurs tous identiques. Le nombre d'additionneurs nécessaires est fonction de la précision réclamée sur l'opérande parallèle. Un multiplieur 12\*X bits occupe 0.22\*1.074 mm<sup>2</sup>.

### 2.2 Un Diviseur

On a pris un algorithme classique de division sans restauration. En acceptant une vitesse de travail faible, on peut se satisfaire d'un seul additionneur série et de deux registres à décalage de longueur N qui réalise une division signée de N bits par N bits en N.(N+1) cycles. La surface est du même ordre de grandeur que celle du multiplieur.

### 2.3 Un Opérateur Racine Carrée

Le cas de la racine carrée est similaire à celui de la division, les algorithmes ayant la même structure.

### 2.4 Un Registre à Décalage

Les registres à décalage sont utilisés soit pour servir de mémorisation soit pour faire circuler les données d'un opérateur à l'autre (ou dans un opérateur). Ils sont conçus de manière classique à l'aide de deux pass-transistors et de deux inverseurs. Ils utilisent deux phases d'horloge non recouvrantes. Dans le cas d'une mémorisation, il est nécessaire d'introduire une troisième horloge qui rendra statique ce registre dynamique.

METHODOLOGIE DE CONCEPTION  
DE CIRCUITS INTEGRES ADAPTES AU TRAITEMENT DU SIGNAL

## 2.5 Un Séquenceur

On a deux problèmes: faire fonctionner chaque opérateur et les faire communiquer. Sachant que l'algorithme que l'on implante est constitué de boucles imbriquées, le fait d'utiliser une structure bit-série implique l'utilisation d'un grand nombre d'horloges différentes. Pour simplifier le séquenceur susceptible de générer ces horloges, nous l'avons éclaté en plusieurs morceaux selon la même philosophie que celle qui préside à la génération des opérateurs: chaque fonction possède son propre séquenceur. Celui-ci est constitué d'un simple registre à décalage dans lequel circule un jeton. Le passage d'un jeton génère les signaux d'horloge nécessaires à l'opérateur et au déclenchement des opérateurs suivants. On a ainsi un arbre hiérarchisé de séquenceurs (qui sont en fait de simples compteurs).

## 3.0 OPERATEURS SPECIALISES EN TRAITEMENT DU SIGNAL

### 3.1 Corrélation, Filtrage En Treillis

Ces deux opérateurs, décrits dans [2], réalisent des fonctions assez simples mais d'utilisation large en traitement du signal. Ils sont à la base de toutes les méthodes paramétriques d'analyse du signal.

On rappelle que l'opérateur de corrélation que l'on propose réactualise un coefficient en 34 périodes d'horloge lorsque l'on code les signaux sur 12 bits et que l'on désire un résultat sans débordement sur 512 échantillons. Cela permet une fréquence d'échantillonnage maximum de 300 KHz avec une horloge à 10 MHz. La surface occupée par cet opérateur est, dans ce cas, de  $0.22 \times 2.5 \text{ mm}^2$ .

L'opérateur de filtrage réalise une cellule d'un filtre en treillis AR ou MA. Avec des coefficients codés sur 12 bits, le signal sur 16 bits, on obtient une fréquence d'échantillonnage maximum de l'ordre de 150 KHz et une surface de  $0.22 \times 3.5 \text{ mm}^2$ .

### 3.2 Opérateur En Cours De Développement

Pour utiliser pleinement les opérateurs précédents, il est nécessaire de posséder un opérateur permettant de passer des coefficients de corrélation aux coefficients du filtre en treillis correspondant. On utilise pour cela l'algorithme de Le Roux - Gueguen [3].

Ce traitement nécessite à chaque itération une division et la réactualisation de variables intermédiaires. Dans le cas où celles-ci sont codées sur 16 bits, que l'on calcule 10 coefficients  $K_i$ , on peut montrer que l'on obtient une précision de 15 bits pour le 1er coefficient  $K_i$  et une dizaine de bits pour le dernier. Cet opérateur nécessite une surface inférieure à  $3 \text{ mm}^2$ . Le temps de calcul est alors de 6400 périodes d'horloge c'est à dire 0.640 ms. On peut comparer ce résultat avec celui proposé dans [4].

### 3.3 Opérateurs Envisagés

Pour étendre la classe des problèmes solubles à l'aide de notre bibliothèque, nous envisageons de nous pencher sur l'implantation:

- de l'algorithme de Schur [5],
- de l'algorithme de Viterbi et de la programmation dynamique,
- du calcul de valeurs propres.

## 4.0 CIRCUITS

### 4.1 1ère Démarche

Comme on l'a montré dans [2], la répétition des opérateurs dans un circuit et le fait de rendre cascadables ces circuits permet d'obtenir simplement un débit de calcul important (figure 1). Il faut noter que la réduction du nombre de bits permettrait de rendre ces circuits encore plus rapides.

Le circuit "Correlateur - filtre en treillis" résulte de la "superposition" des deux opérateurs correspondants et de la répétition du motif autant de fois que le permet la surface de silicium disponible. Un séquenceur est placé en amont du flot des données.

Ce circuit a les caractéristiques suivantes :

- il réalise soit le calcul de 11 coefficients de corrélation sur 32 bits à partir d'un signal codé sur 12 bits, soit le filtrage à l'ordre 11, AR ou MA, d'un signal codé sur 16 bits, les coefficients étant représentés sur 12 bits,
- la fréquence d'échantillonnage est comprise entre 0 et 300 KHz pour la corrélation, entre 0 et 150 KHz pour le filtrage,
- il est cascadable sans dégradation des performances,
- il a une surface intra-plots estimée inférieure à  $10 \text{ mm}^2$ .

Il généralise les 2 circuits proposés dans [2].

### 4.2 2ème Démarche

Si l'on vise des débits de calcul adaptés à des signaux de type audio-fréquence par exemple, on peut réduire la surface de silicium utilisée pour réaliser une fonction en multiplexant les opérateurs puisque le débit nécessaire est alors nettement plus faible que celui permis par les opérateurs précédents. Cela permet donc d'introduire, dans un même circuit, plusieurs blocs placés les uns à coté des autres pour réaliser un traitement plus complexe. L'idée est donc de construire un circuit par concaténation de modules remplissant chacun des fonctions bien précises et travaillant en parallèle. Ces modules devront avoir des entrées sorties à des emplacements normalisés de façon à faciliter leur connexions et une forme particulière de façon à optimiser l'utilisation du silicium.

Les problèmes qui se posent dans la conception d'un circuit suivant cette démarche sont, outre le choix de l'algorithme:

- le choix des paramètres de l'algorithme,
- le choix du nombre de bits pour représenter les nombres (dans une réalisation bit-série, ce choix influe assez directement sur la surface; il faut donc utiliser des outils de simulation permettant de guider (rapidement) ce choix afin de répartir au mieux la précision),



METHODOLOGIE DE CONCEPTION  
DE CIRCUITS INTEGRES ADAPTES AU TRAITEMENT DU SIGNAL

- à partir du moment où l'on veut réutiliser un opérateur dans plusieurs calculs (multiplexage), il faut avoir, là aussi, des outils adéquats pour simuler le séquencement du circuit (cette simulation doit être adaptée au bit-série),
- les outils aidant au test sont bien évidemment nécessaires.

#### 5.0 EXEMPLE: EXTRACTEUR DE PARAMETRES LPC

Ce qui suit n'est qu'un exemple de conception. Pour illustrer la démarche on propose l'étude de l'intégrabilité d'un circuit permettant le codage de la parole par des techniques de modélisation par prédiction linéaire [6]

##### 5.1 Choix Des Algorithmes

L'analyse LPC par la méthode d'autocorrélation demande:

- de pré-accentuer le signal par un filtre de la forme  $1-a.z^{-1}$  de façon à centrer le signal et à équilibrer l'énergie entre les basses et les hautes fréquences,
- de passer ce signal pré-accentué au travers de deux fenêtres de pondération de type Hamming en opposition de phase pour atténuer les effets de bord,
- de calculer  $P+1$  coefficients de corrélation  $R_i$  sur chacune des fenêtres d'analyse,
- de calculer  $P$  coefficients  $K_i$  à partir des coefficients  $R_i$ ,
- de filtrer en inverse le signal avec réactualisation des coefficients du filtre.

En terme d'intégration il est nécessaire de prévoir les algorithmes les plus réguliers et les plus parallèles possibles. Toutes les formes ne sont pas équivalentes. Par exemple, pour calculer  $P$  coefficients  $K_i$ , l'algorithme de Le Roux-Gueguen et l'algorithme de Levinson partent des mêmes données et fournissent (en précision infinie) les mêmes résultats. Pourtant le premier est préférable au second non seulement parce que toutes les variables sont bornées (implantation en virgule fixe aisée) mais aussi parce que le calcul est plus régulier (suppression d'un produit scalaire). Le fait que se soit le futur utilisateur du circuit qui soit le concepteur permet une adéquation très fine du circuit au traitement désiré.

##### 5.2 Choix Des Paramètres

Les paramètres de ce problème sont: la fréquence d'échantillonnage  $F_e$ , le paramètre de pré-accentuation  $a$ , le nombre d'échantillons constituant la fenêtre d'analyse  $N$ , l'ordre de la modélisation  $P$ . Excepté le paramètre  $a$ , tous les autres ont une influence directe sur la surface de silicium nécessaire. Là aussi on voit l'avantage que l'utilisateur soit le concepteur.

Ce choix est rendu particulièrement difficile car un critère de comparaison tel que l'écoute est trop subjectif. Il faudrait avoir des critères objectifs de mesure de la qualité de différentes réalisations d'un circuit d'analyse. Très classiquement, on choisit  $F_e = 8\text{KHz}$ ,  $a$  voisin de 1,  $N = 128$ ,  $P = 10$ .

##### 5.3 Choix Du Nombre De Bits

L'avantage de la structure bit-série est que l'on peut choisir une précision adaptée à chaque type de traitement. Dans notre exemple une précision importante est nécessaire pour calculer les  $R_i$  (une trentaine de bits) et les  $K_i$  (16 bits). Par contre le résidu devant être codé sur peu de bits, on peut réaliser le filtrage inverse avec une précision faible (5 ou 6 bits). On peut donc optimiser la surface de chaque partie du circuit, ce qui entraîne une surface totale moindre.

##### 5.4 Plan De Masse

On décompose le circuit en quatre blocs, chacun ayant son propre séquenceur, l'ensemble étant sous le contrôle d'un séquenceur global (figure 2):

- un opérateur de pré-accentuation et de fenêtre,
- un opérateur de corrélation (comme l'on doit actualiser deux jeux de coefficients de corrélation simultanément cet opérateur doit être doublé),
- un opérateur permettant le calcul des coefficients de réflexion  $K_i$  à partir des  $R_i$ ,
- un opérateur de filtrage en treillis.

Il faut mémoriser le signal qui a permis de calculer les  $K_i$  pour pouvoir le filtrer ensuite en inverse. On montre que cela nécessite au maximum 320 mots de 6 à 12 bits (paramètre à fixer). L'approche série, en particulier lorsqu'elle implique l'utilisation systématique de registres à décalage, est mal adaptée à la mémorisation de nombreuses données. A ce niveau de l'étude on rejette cette mémorisation à l'extérieur du bloc.

Les caractéristiques globales estimées approximativement sont :

- surface totale de l'ordre de 10 mm<sup>2</sup> (sans mémorisation et hors plots),
- fréquence d'échantillonnage maximum d'environ 25 KHz.

Cette fréquence d'échantillonnage maximum peut être augmentée en diminuant la précision des calculs.

##### 5.5 Comparaison Avec Des Systèmes à Base De MuPTS

Le gain apporté par cette démarche relativement à une approche micro-processeur de traitement du signal peut être apprécié en comparant ces résultats avec ceux présentés dans [7] et [8]. Ces deux derniers projets ont des caractéristiques à peu près équivalentes. Ils réalisent un vocodeur LPC à 2400 bits/s à l'aide d'une vingtaine de circuits dont un TMS320. Bien que leur fonctionnalité soit supérieure (échantillonnage, détection de pitch, synthèse en plus) on peut faire les remarques suivantes en faveur de notre démarche:

- performances supérieures dues au parallélisme des opérations (environ deux fois plus, augmentation de la fonctionnalité de notre proposition aisée: excitation multi-impulsionnelle par exemple),
- surface de silicium réduite due à une adaptation très fine du circuit à l'algorithme,



METHODOLOGIE DE CONCEPTION  
DE CIRCUITS INTEGRES ADAPTES AU TRAITEMENT DU SIGNAL

- coût de conception et de test considérablement réduit à cause de la modularité du circuit,
- coût de développement du système également réduit.

6.0 CONCLUSION

Il nous parait nécessaire de rechercher de nouvelles structures de machines permettant de répondre aux besoins de calcul d'algorithmes en traitement du signal mais il faut certainement accorder une attention particulière aux structures d'algorithmes; en effet certains algorithmes sont plus facilement transposables sur ces nouvelles machines que nous voulons définir que d'autres.

BIBLIOGRAPHIE

/1/ H.BARRAL, N.MOREAU, D.NGUYEN HUU, P.SAUVEE  
"Circuits intégrés modulaires pour le traitement du signal"  
Neuvième Colloque sur le traitement du signal et ses applications - NICE 16-20 Mai 1983

/2/ H.BARRAL, N.MOREAU  
"Circuits for digital signal processing"  
IEEE Int. Conf. on ASSP - SAN DIEGO March 19-21, 1984

/3/ J.LE ROUX, C.GUEGUEN  
"A fixed point computation of partial correlation coefficients"  
IEEE Trans. on ASSP, June 1977

/4/ S.Y.KUNG, Y.H.HU  
"A highly concurrent algorithm and pipelined architecture for solving Toeplitz systems"  
IEEE Trans. on ASSP, February 1983

/5/ M.XU  
"Implantation de l'algorithme de Schur"  
Rapport de stage 3ème Année, Document interne ENST 1985

/6/ L.R.RABINER, R.W.SCHAFER  
"Digital processing of speech signals"  
Prentice-Hall 1978

/7/ B.BRYDEN, H.HASSANEIN  
"Implementation of a full duplex 2.4 kbps LPC vocoder on a single TMS320 microprocessor chip"  
IEEE Int. Conf. on ASSP - SAN DIEGO March 19-21, 1984

/8/ A.W.HOLCK, W.W.ANDERSON  
"A single processor LPC vocoder"  
IEEE Int. Conf. on ASSP - SAN DIEGO March 19-21, 1984

/9/ Y.H.OH and al.  
"Architecture for a real-time LPC-based feature measurement integrated circuits"  
IEEE Int. Conf. on ASSP - SAN DIEGO March 19-21, 1984

/10/ B.R.MEARS  
"A modular method for designing custom signal processing integrated circuits"  
IEEE Int. Conf. on ASSP - SAN DIEGO March 19-21, 1984

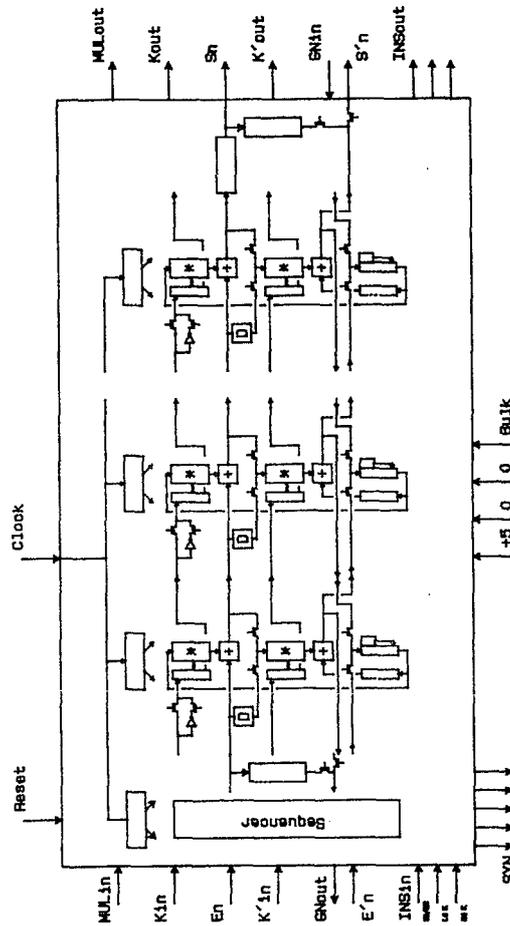


Fig 1 Filtre en Treillis MA ou AR

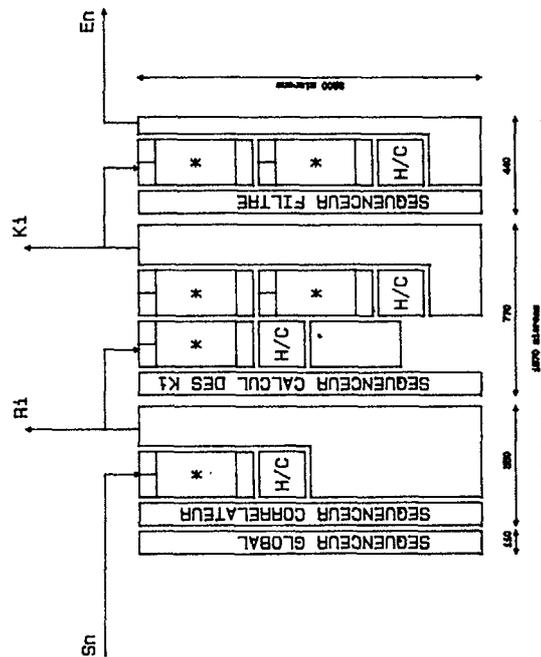


Fig 2 Extracteur LPC

