

# SEPTIEME COLLOQUE SUR LE TRAITEMENT DU SIGNAL ET SES APPLICATIONS

NICE du 28 MAI au 2 JUIN 1979

---

UNE NOUVELLE ARCHITECTURE DE MICROPROCESSEUR POUR  
LE TRAITEMENT DU SIGNAL DE PAROLE

JP.Béraud, D.Esteban, C.Galand

D.Mauduit O.Maurel M.Orsini

C.E.R. I.B.M.  
06610 La Gaude

E.R.A. 835 C.N.R.S.  
Université de Nice

---

## RESUME

Le développement de microprocesseurs rapides permet actuellement de réaliser, en se pliant à des contraintes de temps rigides, des fonctions de plus en plus complexes. Néanmoins, si le choix et les performances du matériel augmentent rapidement, la difficulté consiste à développer le logiciel correspondant tout en restant efficace.

Cet article décrit une architecture de processeur à base de composants rapides conçue pour le traitement du signal et dont la particularité est d'être orientée vers l'utilisation d'un langage de programmation de haut niveau.

Le choix de cette architecture, n'incluant aucun matériel spécialisé du type multiplieur, a nécessité la mise au point de techniques de programmation en vue d'optimiser la réalisation de fonctions de traitement du signal (transformation de Fourier discrète, filtrage, corrélation).

Ce processeur a permis de développer un système de transmission numérique du signal de parole à faible taux d'information mais est également adapté à d'autres applications telles que la transmission de données.

## SUMMARY

The quick development of the microprocessors allows now to perform complex functions even used in real time application. Though the choice and the performances increase, the main difficulty is to develop the corresponding efficient software.

This paper describes a processor architecture based on high speed components, which is high level language oriented to perform signal processing.

The architecture does not use specialized circuits, like a multiplier; to compensate this, some specific software technics are used to optimize signal processing functions (Discrete Fourier Transform, filtering, correlation).

This processor was used to transmit a low bit rate speech signal, but it could be also used on other applications like data transmission.



## UNE NOUVELLE ARCHITECTURE DE MICROPROCESSEUR POUR

### LE TRAITEMENT DU SIGNAL DE PAROLE

#### 1.0 INTRODUCTION

Le développement rapide des microprocesseurs permet actuellement de réaliser des unités de traitement pouvant rivaliser avec du matériel spécialisé. Le choix d'une architecture générale par rapport à un calculateur spécifique se justifie par des avantages de deux ordres :

- La conception, la réalisation, la mise au point ainsi que la maintenance du matériel sont grandement simplifiées. Le marché plus étendu autorise également des prix de revient nettement inférieurs.
- La partie logiciel devient prépondérante, ce qui ajoute de la souplesse d'utilisation. De plus, l'existence d'outils d'aide à la programmation 'classiques' (assembleurs, éditeurs de lien, voire compilateurs) simplifie la tâche de l'utilisateur et minimise le risque d'erreurs au niveau de la conception de systèmes complexes.

Pour des applications dans le domaine du traitement du signal où des fonctions complexes sont mises en oeuvre, telles que filtrages, corrélations, transformées de Fourier, etc., l'emploi de matériel non spécialisé semble incompatible avec la rapidité d'exécution.

Les auteurs proposent une architecture de microprocesseur qui, tout en restant simple, est capable de marier la complexité et la rapidité.

#### 2.0 CHOIX DU MICROPROCESSEUR

A l'heure actuelle il existe deux familles de microprocesseurs qui se différencient par leur technologie, les microprocesseurs MOS et les microprocesseurs bipolaires.

Les premiers sont très intégrés mais fonctionnent à une vitesse moyenne, les seconds sont moins intégrés mais leur vitesse est plus grande.

Dans notre cas, la nécessité de faire du temps réel implique une grande quantité de traitement dans un temps réduit, ce qui nous conduit à choisir la famille bipolaire.

Les composants dans cette famille ne sont pas très nombreux ; notre choix s'est porté sur le microprocesseur AMD2900 /1/ pour les raisons suivantes :

- Le circuit est structuré en tranches de 4 éléments binaires (e.b.), ce qui nous permet de concevoir une arithmétique travaillant sur 16 éléments binaires.

- Le circuit offre le libre choix des instructions.
- Le circuit est compact et comprend 17 registres en plus de l'unité arithmétique.
- Le circuit de séquençement ne nous impose aucune contrainte de programmation.

#### 3.0 ARCHITECTURE

L'architecture utilisée est du type 'pipe line' à 1 niveau, c'est-à-dire que la partie traitement proprement dite, composée essentiellement de l'unité arithmétique et de la mémoire de travail, fonctionne en même temps que la partie de séquençement des instructions. Un registre appelé 'pipe line' est utilisé à cet effet (voir Figure 1). Il est à noter que l'impact de cette structure sur la programmation est nul.

Le cycle d'exécution est unique, une instruction est entièrement exécutée en 250 nanosecondes.

La partie arithmétique travaille sur 16 éléments binaires et la possibilité est donnée pour certaines instructions de travailler en double précision sur deux cycles.

Le format des instructions est fixe, il se compose de deux champs de 8 éléments binaires chacun. Le premier champ est le code opération, il permet donc 256 instructions différentes. Le deuxième champ est l'opérande qui est en fonction du code opération, soit une adresse de registre, soit une valeur immédiate, soit un déplacement par rapport au pointeur d'adresse de la mémoire de travail, soit une adresse de branchement du compteur d'instructions.

La mémoire d'instruction est composée pour la plus grande part de mémoire morte ; seulement 256 mots sont en mémoire vive et servent à générer dynamiquement du programme.

La mémoire de travail est adressée en indirect, elle utilise un registre appelé pointeur d'adresse au contenu duquel est ajouté un déplacement. Un contrôle permet d'adresser la mémoire par le même mécanisme, mais modulo 16, ce qui permet de simplifier le transfert des données dans les opérations de filtrage. Le séquençement des instructions est assuré par le circuit de la même famille que le microprocesseur, le 2911. Il comprend un registre d'adresse incrémentable, un registre de branchement et une entrée directe ; un sélecteur en assure le choix.



UNE NOUVELLE ARCHITECTURE DE MICROPROCESSEUR POUR  
LE TRAITEMENT DU SIGNAL DE PAROLE

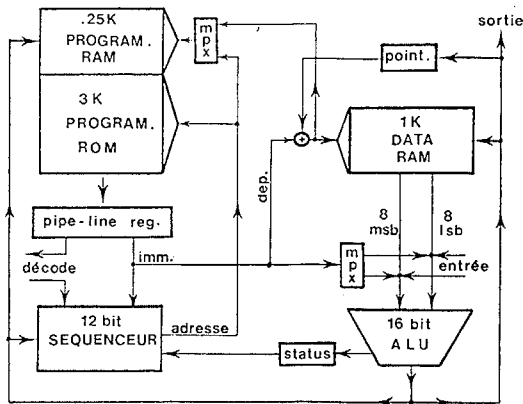


fig.1 : L'architecture du microprocesseur

Le fonctionnement normal d'un programme peut être interrompu pour exécuter un autre programme dit 'programme d'interruption'. Dans ce cas l'adresse courante est conservée dans un registre spécial du séquenceur (2911). Un deuxième pointeur de la mémoire de travail est attribué à ce nouveau niveau de programme.

Le microprocesseur est monté sur une carte de 18 x 24 cm (voir Figure 2), et comprend 80 circuits TTL en plus des 4 unités arithmétiques (2901) et des 3 séquenceurs (2911). La puissance consommée par la carte est de l'ordre de 30 watts.

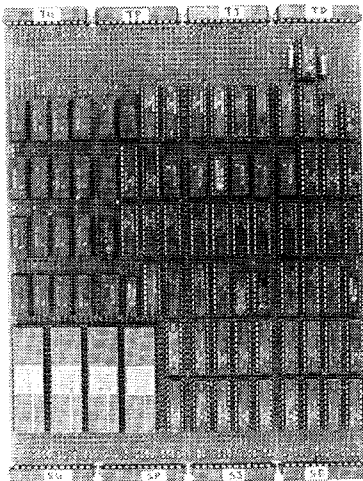


fig.2 : Carte microprocesseur

Les principales caractéristiques se résument comme suit :

- Microprocesseur 16 e.b.
- Temps d'instruction : 250 ns.
- Mémoire d'instruction : 3K ROM + 0.25K RAM.
- Registres : 17.
- Mémoire de travail : 1K mots de 16 e.b.
- Accès direct à la mémoire d'instruction.
- Accès indirect aux mémoires d'instructions et de travail.
- 1 niveau d'interruption.

#### 4.0 SUPPORT DE PROGRAMMATION

Dans un microprocesseur la partie programmation nécessite un effort non négligeable, ce qui implique une aide qui se manifeste sous deux formes :

1. Ecriture symbolique.
2. Langage de haut niveau.

La première est donnée par un assembleur qui permet d'écrire des programmes en utilisant des mnémoniques pour définir les instructions utilisées, et des symboles alphanumériques pour définir des adresses. L'assembleur calculera les adresses et convertira les mnémoniques et ses opérands en mots binaires. Seulement 80 mnémoniques sont à la disposition du programmeur, l'assembleur fait le choix parmi les 256 instructions possibles en se basant sur les opérands des mnémoniques qui sont des registres, des adresses ou des indices.

Le langage de haut niveau est fourni par un langage existant fonctionnant sur les ordinateurs des séries 360/370 dont le code généré est retranscrit pour être assimilable par notre assembleur. Ce langage a l'avantage de travailler en format fixe comme le fait notre machine (16 éléments binaires). Cette traduction a pu être effectuée principalement grâce à la similitude de notre machine et des ordinateurs 360/370, notamment du point de vue des registres. 16 registres sont utilisés dans les deux cas avec des fonctions bien particulières affectées à certains registres ; ces fonctions ont été conservées. Le rapport de traduction est de l'ordre de 1,2 à 1,3 c'est-à-dire que 20 à 30 % d'instructions supplémentaires sont générées par le traducteur. Le langage de haut niveau étant optimisé, une programmation directe en assembleur ne permettrait pas une économie du nombre d'instructions supérieure à 20 %.



UNE NOUVELLE ARCHITECTURE DE MICROPROCESSEUR POUR  
LE TRAITEMENT DU SIGNAL DE PAROLE

Par souci d'efficacité certaines primitives classiques du traitement du signal, comme filtrage, transformée de Fourier et corrélation, font appel à des 'macros' qui ont été optimisées et qui génèrent le code correspondant aux paramètres de ces primitives.

### 5.0 SUPPORT DE MISE AU POINT

Afin de permettre une mise au point facile, autant du point de vue matériel que logiciel, certains outils ont été développés.

#### a. Le panneau de contrôle :

Il a surtout un but d'information, le code de l'instruction, l'état des différents bus, les adresses de l'instruction et de la mémoire de travail sont affichées. La possibilité est donnée, par son intermédiaire, de s'arrêter sur une adresse d'instruction et de fonctionner en pas à pas. On peut également afficher le contenu des registres et des positions de la mémoire de travail.

#### b. La mémoire d'instruction :

Elle est normalement réalisée avec de la mémoire morte (ROM), mais en période de mise au point d'un programme les modifications de celui-ci nécessitent un moyen plus souple de stockage du programme. Nous avons utilisé une mémoire vive (RAM) qui est chargée par un petit ordinateur IBM 7 relié au Centre de Calcul du Laboratoire. Manipulations comprises, le chargement s'effectue en 3 ou 4 minutes.

#### c. Le simulateur :

C'est un programme tournant sur le Centre de Calcul qui simule entièrement le microprocesseur et qui permet d'avoir à chaque instruction les mêmes informations que donnerait le panneau de contrôle. Grâce à cet outil, la mise au point d'un programme peut s'effectuer sans l'aide du microprocesseur. Ceci permet à plusieurs personnes de tester simultanément leurs programmes sans avoir la contrainte de la disponibilité du microprocesseur.

### 6.0 TECHNIQUE DE PROGRAMMATION

L'exécution des primitives du traitement du signal fait appel en grande partie à des multiplications. Notre architecture se voulant simple, aucun organe multiplieur ne s'y trouve, les multiplications sont remplacées par une série d'additions et de décalages en se basant sur la représentation canonique minimale signée des nombres binaires /2/, /3/ au lieu du code en complément à deux.

Deux cas sont à considérer :

#### 1. Le multiplicateur est connu à priori :

Si l'un des termes de la multiplication est connu, la succession des additions et des décalages est écrite à l'assemblage du programme et la séquence sera toujours la même. C'est le cas des filtres qui s'écrivent sous la forme :

$$S(m) = \sum_{n=0}^N f(m-n) \cdot K(n)$$

Pour N supérieur à 4 et des coefficients de 12 e.b., le nombre d'instructions moyen par multiplication est égal à 3, compte tenu de la gestion /4/.

C'est aussi le cas des transformées de Fourier qui s'écrivent :

$$A(k) = (1/N) \cdot \sum_{n=0}^{N-1} f(n) \cdot W^{-kn}$$

$$\text{où } W = \exp(j2\pi/N)$$

En utilisant la méthode de G.D. BERGLAND, /5/, la FFT sur 128 points se fait en 2,9 ms.

#### 2. Le multiplicateur est inconnu à priori :

Si les deux termes sont inconnus, soit qu'ils viennent de l'extérieur, soit qu'ils proviennent de précédents calculs dont le résultat n'est pas prévisible, la décomposition en opérations simples, additions et décalages, doit se faire par programme sur le microprocesseur lui-même. La séquence générée sera ensuite exécutée pour effectuer la multiplication. C'est le cas des corrélations, des égalisations et des adaptations automatiques.



UNE NOUVELLE ARCHITECTURE DE MICROPROCESSEUR POUR  
LE TRAITEMENT DU SIGNAL DE PAROLE

Dans la mesure où les paramètres évoluent lentement et où ils sont considérés comme fixes pendant un certain temps, comme dans le filtrage adaptatif représentant le conduit vocal (LPC) /6/, le temps de préparation devient négligeable devant l'exécution des multiplications proprement dites ; dans notre application, 0,10 ms de préparation pour 0,95 ms d'exécution.

Pour une autocorrélation, les paramètres de la multiplication viennent de l'extérieur, l'exécution est faite une seule fois après avoir fait la préparation, le temps de préparation est du même ordre de grandeur que l'exécution. Dans notre application 10 points d'autocorrélation d'une fenêtre de 256 échantillons, ce qui représente 2560 multiplications et 2560 additions, s'exécutent en 4,2 ms et la préparation prend 3,2 ms.

#### CONCLUSION

Une unité de traitement du signal vocal a été réalisée avec le microprocesseur décrit dans cet article. L'architecture ne fait pas intervenir d'unité spécialisée de traitement mais est capable d'en assurer toutes les fonctions, principalement grâce aux techniques de décomposition canonique mises en oeuvre par la génération dynamique de programmes (programmation dynamique). Les principales fonctions exécutées dans ce microprocesseur pour effectuer la compression du signal de parole à 4800 Bps /7/ sont:

- 10 points d'autocorrélation d'une fenêtre de 256 échantillons.
- Calcul des coefficients du filtre prédictif par l'algorithme de LE ROUX /8/.
- 1 filtre passe-bas transversal de 31 coefficients.
- 6 filtres transversaux décimateurs de 64 coefficients.

Dans la transmission de donnée, la réalisation de MODEM fait intervenir le même type de primitives de traitement du signal ; l'architecture proposée est donc aussi adaptée à ce domaine d'application.

#### REFERENCES

- /1/ ADVANCED MICRO DEVICE INC.  
The AM2900 Family Data Book,  
901 Thompson Place,  
Sunnyvale, Ca. 94086
- /2/ G.W. REITWEISNER  
Binary Arithmetic in Advances  
in Computer, pp 163-168  
Academic Press, New-York 1965.
- /3/ A. PELED  
On the Hardware Implementation  
of Digital Signal Processors,  
IEEE Trans. on ASSP, Vol.24,  
Feb. 1976.
- /4/ M. PEARD, D.J. ESTEBAN  
Performance Evaluation of the Canonical  
Signed Digit Code,  
IEEE Int'l Conf. on ASSP, Hartford,  
May 1977.
- /5/ G.D. BERGLAND  
A Fast Fourier Transform Algorithm  
for Real-Valued Series,  
COMM. ACM 11, No. 10, Oct. 1968,  
pp 703-710.
- /6/ J.D. MARKEL and A.H. GRAY  
Linear Prediction of Speech,  
Springer-Verlag, New-York 1976.
- /7/ C. GALAND, D.J. ESTEBAN, J. MENEZ,  
D. MAUDUIT  
Codage Prédictif du Signal de Parole  
à 4800 Bps.  
GRETSI Nice, 28 Mai-2 Juin 1979.
- /8/ J. LE ROUX  
Optimisation du calcul des coefficients  
PARCOR,  
7èmes Journées d'Etudes sur la Parole,  
Nancy 1976.