

ETUDE ET CONCEPTION DE LA MACHINE DE COMMUNICATION DU COPROCESSEUR FELIN

PH. PAUL*

M. COSNARD, J. DUPRAT, A. GUYOT, B. HOCHET, J.M. MULLER, H. OUAOUICHA, E. ZYSMAN
CNRS, LAB. TIM 3, INPG, 46 AV. FÉLIX-VIALLET, 38031 GRENOBLE CEDEX, FRANCE

RESUME

Nous décrivons l'architecture VLSI de la machine de communication d'un calculateur d'expressions arithmétiques (composées de fonctions élémentaires en virgule flottante) appelé FELIN (Fonctions Élémentaires Intégrées).

Les principaux attraits de l'architecture de la machine de communication sont la régularité, la paramétrisation et la génération automatique du dessin des masques, comme c'était le cas pour la machine à calcul (partie contrôle et partie opérative). Le chip final est dessiné en technologie CMOS deux couches d'aluminium.

Le travail a été soutenu en partie par le GRECO C³ et le GCIS du CNRS français.

* Actuellement en mission au Laboratoire de Microélectronique de l'école Polytechnique de l'Université de São Paulo (Brésil) - Caixa Postal 8174 - Tel.: (011) 815-9322 - Ramal 257 05508 São Paulo.

INTRODUCTION

Le coprocesseur FELIN est composé de deux machines: La machine de communication et la machine de calcul;

La machine de communication stocke les opérateurs et les opérandes correspondants d'une expression arithmétique à calculer. Puis, elle génère un profil d'exécution qui transfère à la machine à calcul la fonction mathématique et ses arguments et stocke les résultats intermédiaires.

La machine à calcul se divise en deux parties:

- une partie contrôle décompose la fonction mathématique en une séquence de plusieurs fonctions au niveau le plus élémentaire, puis génère l'enchaînement des procédures et l'algorithme correspondant pour chacune de celles-ci, basé sur le schéma CORDIC.
- une partie opérative qui exécute le programme généré par la partie contrôle.

FELIN contient un directory de 25 fonctions élémentaires.

- incluant les fonctions trigonométriques et hyperboliques sinus, cosinus et tangente, et leurs inverses, logarithme et exponentielle en base 2, e et 10; la racine carrée, x à la puissance y et, bien sûr l'addition, la soustraction, la multiplication et la division - définies en standard IEEE. Le chip peut travailler avec des nombres à virgule flottante à simple précision (32 bits), double précision (64 bits) et à précision étendue (80 bits). Il n'accepte ni les entiers, ni les nombres décimaux. Toutes les fonctions transcendantales sont calculées sans limitation d'arguments.

Le circuit est dessiné en CMOS à double niveau de métal, la grille étant de 2 μ m, assemblé dans un boîtier de 40 pattes. Il mesure 7,1 x 5,9mm et contient plus de 90.000 transistors.

I. ETUDE PRELIMINAIRE DE LA MACHINE DE COMMUNICATION

Le chip FELIN a été conçu pour être un périphérique de calcul du 68000 de chez MOTOROLA. La connectique de la machine de communication se traduit donc, d'un côté par l'ensemble des commandes et bus qui seront connectés au microprocesseur, et de l'autre par l'ensemble des commandes et bus connectés à la machine à calcul.

Cette dernière ne travaillant qu'avec des mots de 80 bits, il est donc nécessaire de prévoir un interface entre le bus de 16 bits d'entrée et sa partie opérative (fig.1).

Pour cela, nous utilisons cinq registres de 16 bits. Un registre supplémentaire sera nécessaire pour entrer les instructions (code opérateur, format opérande, format résultat). Nous retrouvons le même problème pour la sortie et aurons donc encore 6 registres de 16 bits (5 pour le résultat, 1 compte-rendu) pour assurer l'interface microprocesseur - machine à calcul.

De plus, cette même partie opérative met un certain temps (ou cycle) pour calculer les fonctions mathématiques. Il faut donc prévoir un ensemble de files d'attente, afin de pouvoir stocker les expressions arithmétiques provenant d'un CPU, sans le bloquer; c'est pour cela que nous avons choisi une structure séparant le traitement des données opérandes et des données instructions. Ce qui nous donne, deux files (file d'entrée opérandes et file de sortie résultats) une pile pour les résultats intermédiaires et deux files (file d'entrée instructions et file de sortie compte-rendu) (fig. 2).

Voici un exemple de fonctionnement de ces différentes pile et files avec l'expression arithmétique $y = a \sin (Wt + \phi)$, sachant que la machine de communication utilise la notation polonaise inversée, (fig. 3).

II. FONCTIONNEMENT ALGORITHMIQUE DE LA MACHINE DE COMMUNICATION

La machine de communication est composée de deux parties: Une partie "registres tampons entrées/sorties" et une partie "gestion de données". Celles-ci fonctionnent généralement en parallèle; sauf pendant le transfert de données de l'une dans l'autre.

La description complète est réalisée à l'aide de six algorithmes:

- . L'algorithme "d'entrée/sortie de la machine de communication" gère les entrées et sorties (lecture ou écriture) de mots de 16 bits dans les registres tampons sélectionnés par un bus d'adresses (fig. 4).
- . L'algorithme de "transfert des registres d'entrée tampons" permet de transférer l'ensemble des registres tampons d'entrée en une seule opération dans les files d'attente (INSTRUCTION et OPERANDE).
- . L'algorithme "début de calcul" permet de définir algorithmiquement la gestion de deux types d'opérateurs:
 - opérateurs internes à la machine de communication comme ENTER qui permet d'entrer un opérande en notation polonaise inversée; SPACE opérateur créé pour FELIN, pratiquement identique à ENTER mais réservant en plus une case vide dans la file pour certains opérateurs "bloquants" (tels que DUP, EXC, SIN & COS qui ont besoin d'une place supplémentaire dans la file); DUP qui duplique un opérande ou un résultat; EXC qui échange deux opérandes ou deux résultats ou encore un opérande et un résultat.
 - opérateurs transférés à la machine à calcul, tels les opérateurs unaires (SIN & COS, tangente, exponentielle... etc.) et les opérateurs binaires (+, -, /, * ... etc.).
- . L'algorithme de "fin de calcul" permet l'aiguillage des résultats provenant de la partie opérative, soit dans la pile d'évaluation si l'on a un résultat intermédiaire, soit dans la file de résultat si l'on a un résultat final.

- . L'algorithme "transfert de file résultat dans les registres" permet de transférer le résultat final et le compte-rendu, situés en tête de file, dans les registres tampons de sortie, afin de les communiquer au milieu extérieur
 - . L'algorithme "reset" réactualise la machine de communication à son état initial
- Cet ensemble d'algorithmes nous permet de définir la structure interne de la machine de communication (fig. 5).

III. CHOIX TOPOLOGIQUE POUR LE BLOC PILE-FILES OPÉRANDE

L'idée est de transcrire sur le silicium et dans une seule structure topologique deux files et une pile, de définition comparable au type pointeur en PASCAL (vecteur avec plusieurs champs de données, occupation et pointeur de chaînage). Le tout sera situé dans une RAM de 15 mots de 83 bits, allouée de manière dynamique.

De plus, certaines contraintes devront être respectées:

tous les transferts devront se faire sur un bus unique.

tous les algorithmes de la partie opérande, au nombre de dix, du plus bas niveau devront être exécutés sur trois phases d'horloge (1 cycle = 4 phases).

Une étude manuelle nous a permis de déterminer la structure suivante (fig. 6).

Exemple de fonctionnement de la structure (fig. 6) par l'étude de l'algorithme "empile" qui transfère la tête de file opérande dans le haut de la pile.

En phase 1: précharge de toutes les commandes.

En phase 2: transfert du registre tête de file opérande (TETFO) dans le registre tampon de sélection du décodeur (REGDE) et dans le registre tampon du comparateur (CO). En parallèle, transfert du registre des pointeurs sélectionnés (PTR) dans le registre tampon (REGTP) et activation de la commande SET pour signaler que le vecteur sélectionné est occupé.

Le parallélisme est assuré par le sectionnement du bus, lorsque toutes les portes sont fermées, à l'exception des portes P2 et P5.

En phase 3: transfert du registre haut de pile (REGHP) dans le registre des pointeurs sélectionnés (PTR).

En parallèle, transfert du registre tampon (REGTP) dans la tête de file opérande (TETFO), transfert de la queue de file opérande (QUEFO) dans le comparateur - afin de comparer sa valeur avec la valeur du registre CO (s'il y a égalité, cela signifie que la file opérande est vide) - et activation de la commande SET pour signaler que le vecteur sélectionné est occupé.

Le parallélisme est assuré de la même manière que précédemment mais lorsque la porte P3 seulement est ouverte.

En phase 4: transfert du registre tampon du comparateur (CO) dans le haut de la pile (REGHP).

En parallèle, activation de la commande SET (vecteur sélectionné occupé)

Représentation schématique de l'évaluation de la file et de la pile (fig. 7).

IV. APPLICATION VSLI DE LA PARTIE PILE-FILES OPERANDE

4.1 GÉNÉRALITES

Pour la génération de l'ensemble de la machine de communication, nous allons utiliser l'Editeur Graphique "LUCIE" pour dessiner les cellules élémentaires et le compilateur de silicium LUBRICK pour l'assemblage et le routage de celles-ci. Ceci permettra une implantation régulière, paramétrée et modulaire.

La RAM opérande tient, à elle seule, une grande place à l'intérieur de la machine de communication. C'est pourquoi nous nous alignons sur le chemin de données de la partie opérative. ce qui nous impose un bit slice de hauteur identique à celle du bit slice de cette même partie opérative (qui a, elle aussi, été conçue de manière régulière, paramétrée et modulaire (fig. 8)).

4.2 PARTIE PILE-FILES OPERANDE

Le matériel utilisé pour réaliser la partie pile-files opérande est:

- . une RAM de 15 mots de 83 bits pour le champ de données.
- . quinze portes de sélection en lecture et écriture dans cette même RAM.
- . une RAM de 15 mots de 1 bit pour le champ d'occupation.
- . un bloc logique qui recherche une place libre et génère en permanence une adresse de vecteur libre.
- . un encodeur 16/4.
- . une RAM de 15 mots de 4 bits pour le champ pointeur du vecteur.
- . une tranche d'inverseur.
- . un ensemble de registres de 4 bits queue-file, tête de file, haut de pile et un registre tampon assurant le fonctionnement de la machine.
- . un décodeur 4/16 avec un registre tampon permettant de sélectionner le vecteur adressé en son entrée.
- . un comparateur 4 bits avec un registre.
- . un circuit de synchronisation qui sert à associer une commande à une certaine phase, selon la description algorithmique précédente.

. un petit PLA qui permet de générer le séquençement.

Détermination de la génération du dessin:

D'après l'étude précédente, nous avons un bit slice imposé par la partie opérative. Nous dessinons un joint mémoire de la RAM opérante et utilisons sa largeur pour déterminer un bit slice, pour tous les blocs empilés au-dessus jusqu'à la tranche d'inverseur. Ce bloc est appelé "bloc-mémorisation".

Les registres queue-file, tête de file, haut de pile, REGDE, CD sont dessinés de manière minimale en largeur, afin que le bloc registre - dans lequel on a encore inséré un décodeur et un comparateur-soit de taille peu différente en largeur de la partie inférieure du bloc mémorisation (voir plan de masse, fig. 9).

Le petit PLA est de même dimension (largeur) que tout le reste, ce qui nous permet donc d'avoir un bloc compact, qui sera très facilement "collé" à la partie opérative (voir plan de masse).

4.3 PRÉSENTATION DU LAY-OUT DU BLOC PILE-FILES OPÉRANTE

La figure 10 présente un petit morceau du bloc mémorisation qui est connecté à un décodeur dual CMOS par un routage en aluminium inférieur. Ce dernier est lui même connecté à son registre tampon par un routage à trois niveaux (Poly, métal 1, métal 2). De chaque côté de ce décodeur, nous pouvons distinguer les blocs registres queue files, tête de files, pile... etc. Au dessus, le circuit de synchronisation puis le petit PLA (fig.10).

CONCLUSION

Le coprocesseur FELIN de complexité assez élevée, a été conçu au sein d'une petite équipe, avec un matériel modeste. Il doit être envoyé ou CMP 1987 français.

BIBLIOGRAPHIE

- (1) COSNARD M., GUYOT A., HOCHET B., MULLER J.M., OUAOUICHA H. et ZYSMAN E.
"FELIN, an elementary function cruncher", Future trends on computing, dec. 1985, Edited by J. Wiley & Sons, 1986.
- (2) COSNARD M., GUYOT A., HOCHET B., MULLER J.M., OUAOUICHA H., PAUL P. et ZYSMAN E. - "The FELIN arithmetic coprocessor chip", IEEE Transaction on computers.
- (3) HOCHET B. - "Conception de VLSI et application au calcul numérique", Thèse de Doctorat Institut National Polytechnique de Grenoble, Grenoble, France, Jan. 1987.
- (4) MULLER J.M. - "Méthodologie de calcul des fonctions élémentaires", Thèse de Doctorat, Institut National Polytechnique de Grenoble, Grenoble, France, Sept. 1985.

- (5) PAUL P 'Etude et réalisation de la machine de communication du coprocessor FELIN'
 DEA Microélectronique, Institut National Polytechnique de Grenoble, Grenoble, Juin
 1986.



Figure 1 - Connexions de la machine de communication.

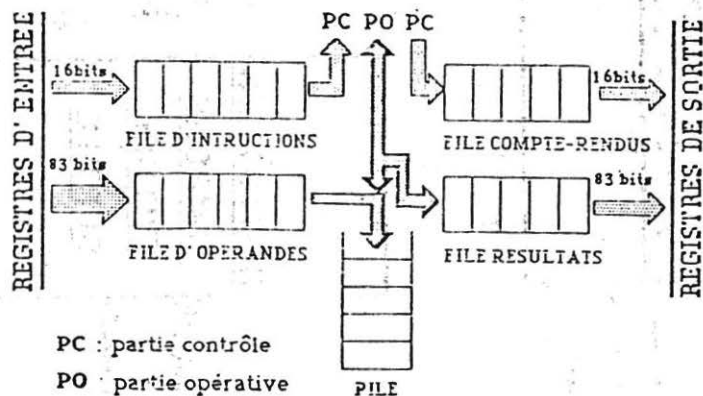


Figure 2 Structure des files d'attente de la machine de communication.

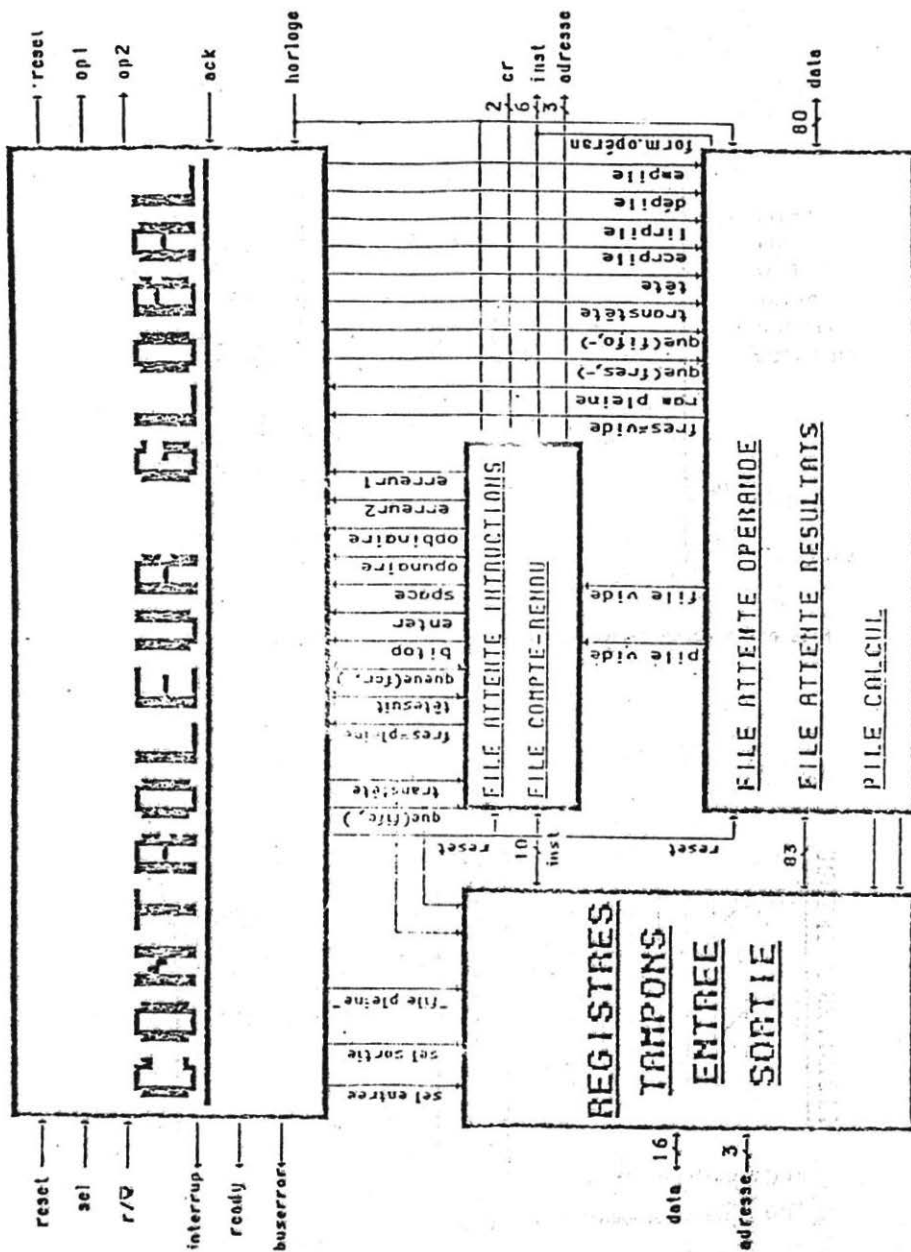


Figure 5 - Structure interne de la machine de communication.

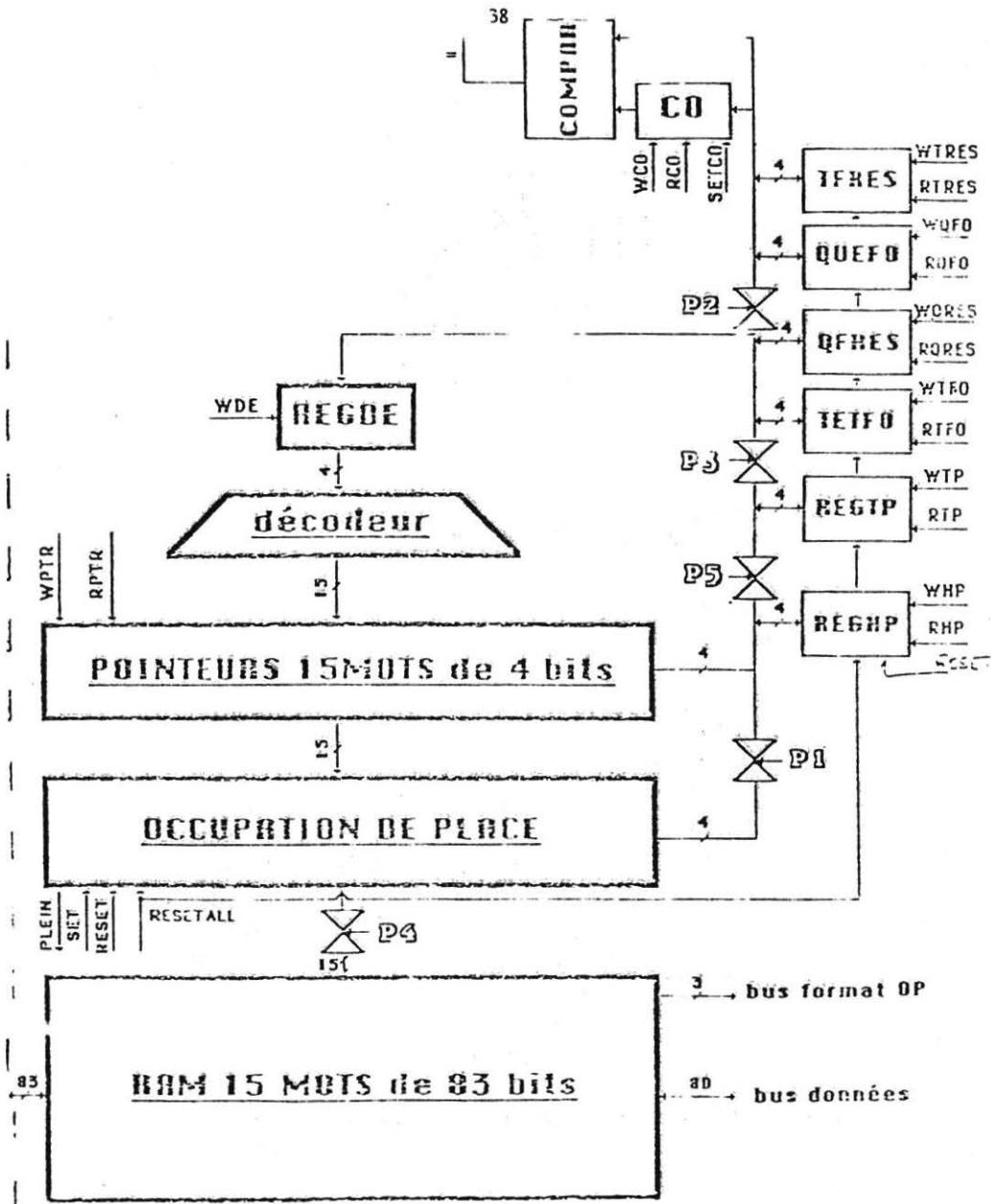


Figure 6 Structure topologique de la partie pile-files opérande

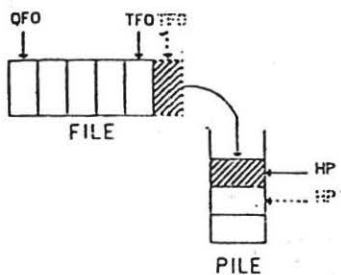


Figure 7 - Représentation schématique de l'évolution de la file et de la pile pour l'algorithme "EMPILE".

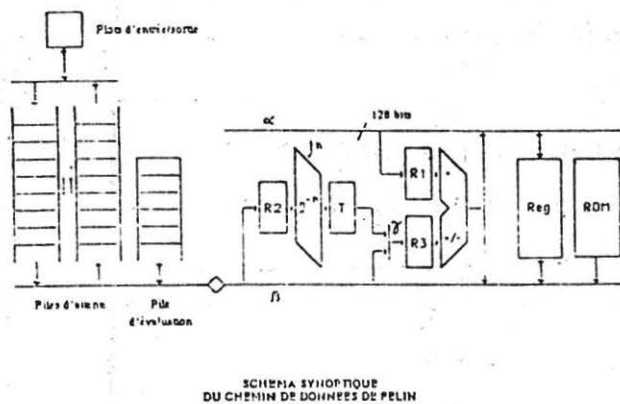


Figure 8 - Schéma synoptique du chemin de données de FELIN.

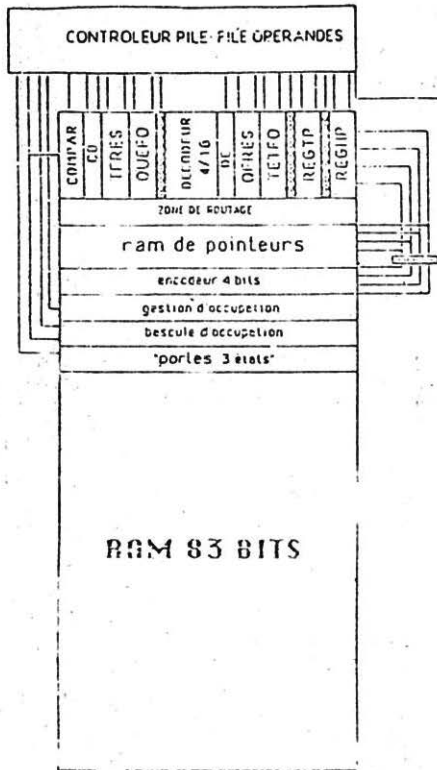


Figure 9 - Plan de masse du layout de la partie pile-files opérande.

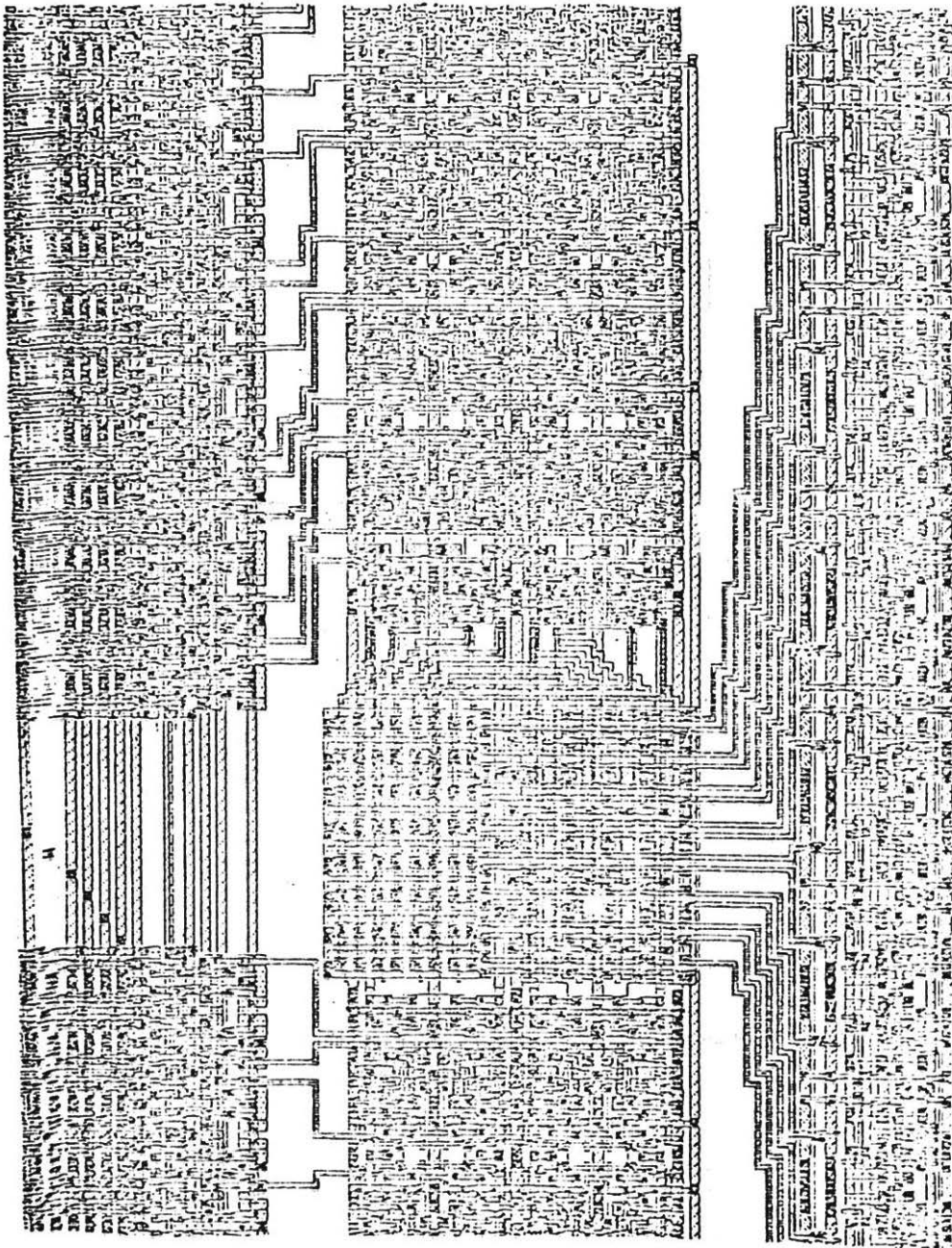


Figure 10 - Layout d'une partie du bloc pile-files opérande .