

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT : D'ELECTRONIQUE

PROJET DE FIN D'ETUDES

SUJET

المدرسة الوطنية للعلوم الهندسية
المكتبة
ECOLE NATIONALE POLYTECHNIQUE
BIBLIOTHEQUE

*Contribution a l'Etude et
Realisation d'une Unité de
Traitement Rapide*

Proposé par :
M^r A. Abdellaoui
Docteur de Spécialité

Etudié par :
M^r M. Madjene
M^r A. Benkrid

Dirigé par :
M^r B. Hamidi
Ingénieur au C.E.N.



PROMOTION : JUIN 1984

الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم والبحث العلمي
MINISTERE DE L'ENSEIGNEMENT ET DE LA RECHERCHE SCIENTIFIQUE

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT : D'ELECTRONIQUE

PROJET DE FIN D'ETUDES

SUJET

*Contribution a l'Etude et
Realisation d'une Unité de
Traitement Rapide*

Proposé par :

M^r A. Abdellaoui

Docteur de Spécialité

Etudié par :

M^r M. Madjene

M^r A. Benkrid

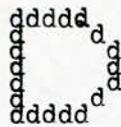
Dirigé par :

M^r B. Hamidi

Ingénieur au C. E. N.



PROMOTION : JUIN 1984



E D I C A C E S .

A la mémoire de mon père .

A ma mère .

A mes frères et mes soeurs .

A ma fiancée .

A toute ma famille .

A tous mes amis(es).

Mahfoud

MADJENE

A mes parents .

A mes frères et mes soeurs.

A toute ma famille .

A tous mes amis(es).

Abdelhak BENKRID

SSSS

SSSS

SSSS O M M A I R E

Pages

CHAPITRE I: INTRODUCTION	01
I Généralités:	01
II Processus de l'étude d'un système	01
CHAPITRE II: ETUDE DE LA TRANSFORMÉE DE FOURIER.	04
I Étude de la DFT	04
II Étude de la FFT.	05
1) Choix d'un algorithme.	05
2) Étude d'un algorithme	05
III Comparaison entre la DFT et la FFT	13
IV Applications de la transformée de Fourier	15
1) Fonction de transfert	15
2) Fonction de cohérence.	
3) Filtrage numérique.	16
4) Applications médicale.....	17
CHAPITRE III : PROGRAMMATION DE LA TRANSFORMÉE DE FOURIER	18
I Organigramme de la DFT et la FFT	19
II Programme en FORTRAN	20
III Résultats obtenus sur quelques exemples.....	22
IV Vérification par la transformée Inverse.	26
1) Définition	
2) Programmation	
3) Résultats	
V Conclusion.	26
CHAPITRE IV : LA CARTE CPU	30
I Le microprocesseur MC 6809.	30
1) Caractéristiques générales.	
2) Structure de base du microprocesseur.	
3) Organisation Software du MC 6809.	36
II L'unité de calcul AM 9511.....	40
1) Généralité	
2) Structure de l'unité AM 9511	
3) Organisation Software	44
III Les circuits annexes.....	
IV Connexion de la carte	49
CHAPITRE V : ENTREE/ SORTIE.	51
I Généralités	
II Étude des interfaces programmables	
III Étude d'un affichage 7 segments	56
1) Adaptation des niveaux	57
2) Fonctionnement de l'affichage	
IV Étude d'un clavier	58
1) Rôles des différents circuits	
2) Fonctionnement du clavier	
V Réalisation pratique: (Clavier et affichage 7 segments) ..	60
VI Programme de gestion du système: (Fonction du J BUG)	65
1) Organigramme générale de fonctionnement.....	65
2) Les différents sous programmes.	66
CHAPITRE VI: CONCLUSION ET RECOMMANDATION	77

BIBLIOGRAPHIE:

The Fast Fourier TransformBRIGHAM.

IEEE Transaction On AUDIO :(AU 17)

IEEE Transaction ON Computer (VOL: C 19)

Revue mesure : Regulation Automatique .

Le microprocesseur MC 6809 CLAUDE DARDANEERYOLLES

Introduction to microprocessor :::BBURNER .

Interfacage du 6800 AU 6809 ..REVERIN .

Revue Haut parleur :Micro informatique .

Brochure de l'unit e AM 9511 .

Systeme   microprocesseurM. AUMIAUXMASSON.

CHAPITRE : I

INTRODUCTION

I GENERALITES.

Le passage de la théorie à la pratique demande souvent beaucoup de temps . Ainsi 150 ans se sont écoulés entre la théorie de J.B.J. Fourier et ses applications. L'analyse de Fourier n'est pas un sujet nouveau. En effet c'est vers les années 1800 que J.B.J. Fourier a développé sa théorie. Depuis les séries et les intégrales de Fourier ont trouvé de plus en plus d'applications dans divers domaines.

Dans le passé l'analyse de Fourier fut utilisée comme une approche mathématique pour obtenir des informations dans le domaine fréquentiel .

Les transformées de Fourier nécessitent trop de temps de calcul. C'est vers 1965 que J.W. COOLEY et J.W. TURKEY ont publié :
 "AN ALGORITHM FOR THE MACHINE CALCULATION OF COMPLEX FOURIER SERIES".
 Cet algorithme plus connu sous le nom de FFT (Fast Fourier Transform est devenu un outil très utile dans l'analyse de Fourier .
 Il permet un traitement rapide de celle-ci.

A l'aide de ces moyens tant mathématiques qu'informatiques le traitement numérique du signal est devenu à la fois commode et précis .

L'étude des phénomènes physiques requiert souvent une analyse spectrale du signal. Ainsi l'étude fréquentielle de ces derniers trouve son application dans plusieurs domaines en l'occurrence:

- Analyse des bruits rayonnés: (Détection Radar Sonar Acoustique)
- Mesure d'isolement aux vibrations.
- Détection des résonances: (Défauts des turbines des moteurs)
- Contrôle des structures en vibrations: (Aéronautique Génie civil)

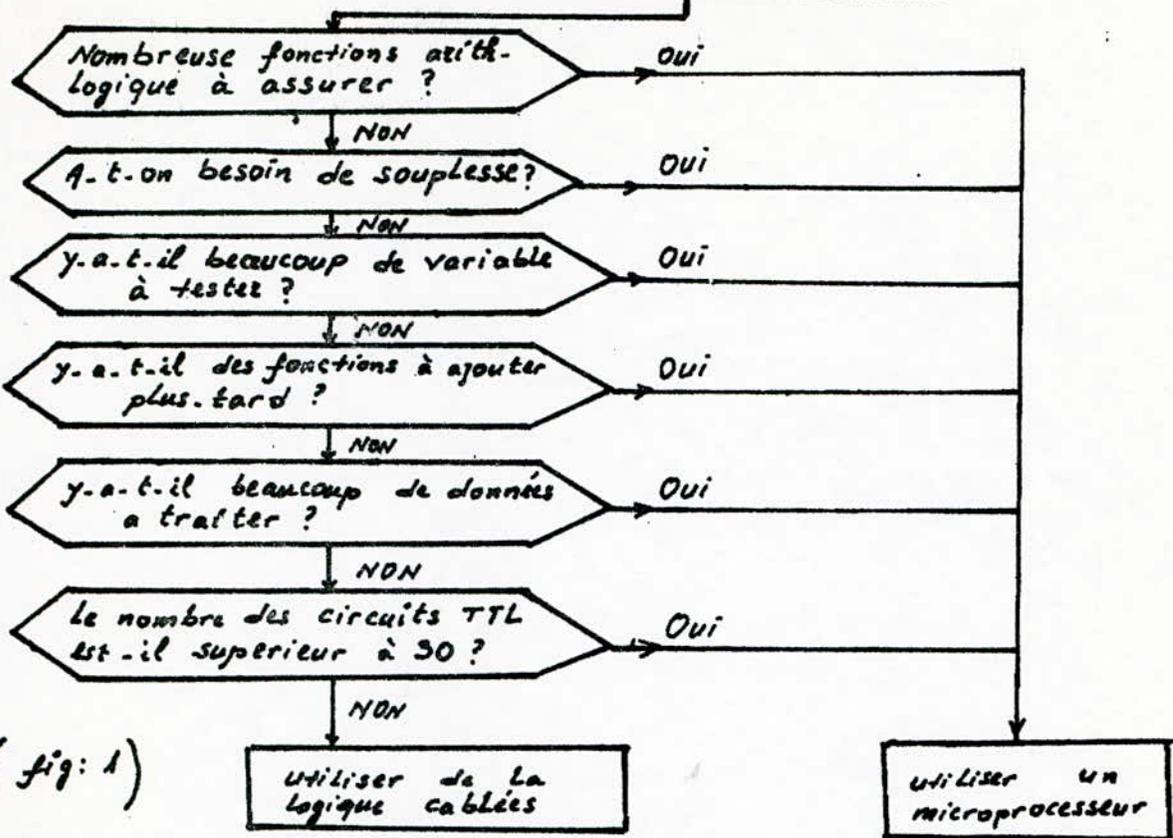
Dans le cadre de notre projet de fin d'étude il nous a été confié l'étude de la FFT en vue de son application dans une unité de traitement rapide à base du microprocesseur MC 6809 . Cette unité devra comprendre :

- La carte CPU.
- La carte RAM
- Le clavier .
- Et les afficheurs.

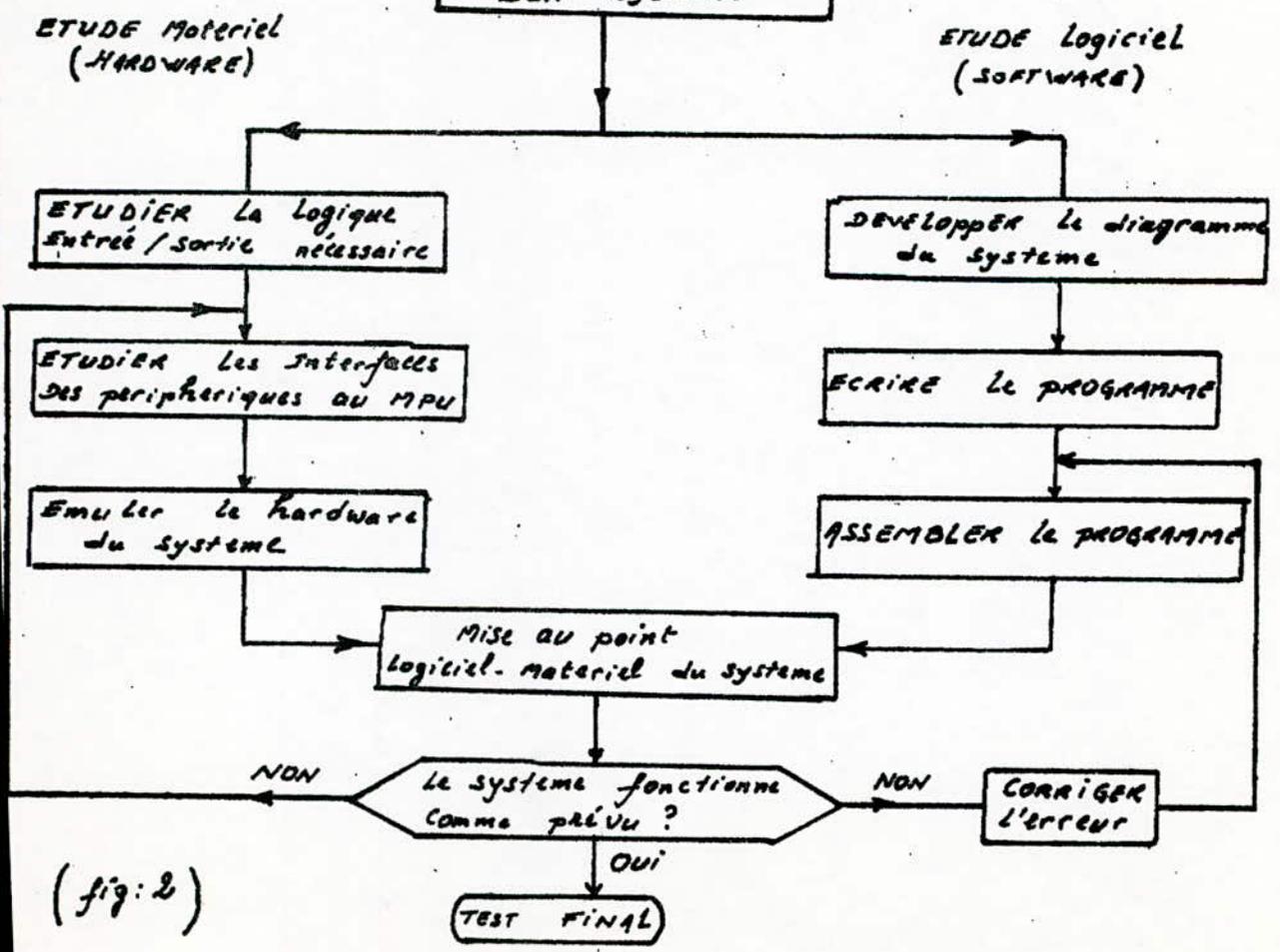
II PROCESSUS DE L'ETUDE D'UN SYSTEME.

Le choix du type du microprocesseur est motivé par le domaine de son application. De ce fait le MC 6809 a été adapté en raison de la taille de ses registres (16 bits) et ses différents modes d'adressages. Voir organigramme fig : 1 .

CONCEPTION DU SYSTEME



EXAMEN DES BESOINS D'UN SYSTEME



La conception d'un système micro informatique résulte d'un compromis entre le hard et le soft.

Le bilan des tâches à allouer au système est alors nécessaire. Il faut faire une répartition judicieuse des tâches entre ces deux parties (HARD et SOFT), afin d'optimiser aussi bien le temps de traitement, que le cout du système. Voir organigramme (Fig. 2)

C H A P I T R E II

ETUDE DE LA TRANSFORME DE FOURRIER

L'analyse de Fourier est un moyen de décomposition d'un signal en une somme de signaux élémentaires particuliers qui ont la propriété d'être facile à mettre en oeuvre et à observer.

I ETUDE DE LA D.F.T.

La transformée de Fourier discrète est définie par l'équation algébrique suivante:

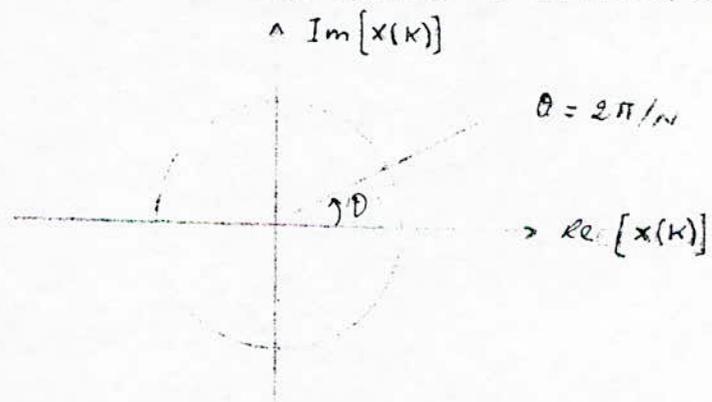
X(K) = 1/N \sum_{n=0}^{N-1} x(n) EXP.(-j2IIInK/N) avec K=0 1,... (N-1)

Cette equation établit une relation entre deux ensembles de N nombres complexes.

La partie réelle : Re (X(K)) = 1/N \sum_{n=0}^{N-1} x(n) COS (2IIInK/N)

La partie imaginaire: Im(X(K)) = 1/N \sum_{n=0}^{N-1} x(n) SIN(2IIInK/N)

Les affixes des nombres complexes W (nK)=EXP (-j2IIInK/N) sont les coefficients de la D.F.T. . Ils se trouvent sur le cercle unité



Cette relation peut s'écrire sous une forme matricielle dont l'opérateur est la matrice carrée de dimension N x N constitué par les termes complexes de Fourier W (nK).

l'équation de la matrice est: [Matrix equation showing X(0) to X(N-1) and W powers]

Le nombre d'opération nécessaires au calcul de toutes les parties réelles et les parties imaginaires est très important. Le temps exigé sera prohibitif à partir d'un certain nombre d'échantillon. Pour remédier à cet inconvénient il faut réduire au maximum le nombre d'opérations.

II ETUDE DE LA F.F.T.

La F.F.T. est un processus de transformation de fourier à laquelle le traitement de la D.F.T. conduit à des algorithmes de calcul. Ces derniers se différencient par le type d'embrouillage entraîné par la multiplication de la matrice de permutation (ou réarrangement) on distingue

Algorithme A: Le désambrouillage se fait au niveau des entrées afin que les séquences de sortie soit dans l'ordre.

Algorithme B : On introduit les données normalement, mais un réarrangement est nécessaire en fin d'exécution.

Algorithme C : On fait le désambrouillage au niveau de la matrice afin que les séquences de données et de sorties soient dans l'ordre.

1 Choix d un algorithme

Pour les deux premiers algorithmes le principe de désambrouillage est simple si le nombre d'échantillons est une puissance de deux car on applique le principe du binaires réfléchis connu sous le nom d'effet miroir pour un nombre d'échantillon qui n'est pas une puissance de deux le réarrangement est délicat. Comme notre objectif est élaboré un algorithme valable pour un nombre quelconque d'échantillons nous optons pour le troisième algorithme.

2 ETUDE D'UN ALGORITHME

Considérons l'expression de la transformé de fourrier

$$X(K) = 1/N \sum_{n=0}^{N-1} x(n) W(nK) \quad K= 0 1 \dots (N - 1)$$

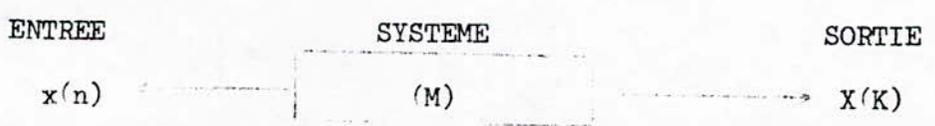
X(K): Séquence de sortie

x (n): Valeurs d'échantillonnage du signal continu x(t)

W(nK) = EXP (-j2II/N) nK : Coefficient de la matrice

Cet algorithme utilise toutes les propriétés de l'exponentielle complexe, qui constitue l'opérateur principal du système

$$X (K) = (M).x (n)$$



L'exponentielle complexe possède les propriétés de:

* Périodicité : W(K + aN) = W(K) a : Entier

* Symétrie par rapport à N/2 : W (K+N/2) = W(K)

La matrice carrée d'ordre N présente des particularités évidentes due à la périodicité et la symétrie du nombre complexe W(nK). Ainsi que l'équivalence entre les lignes et les colonnes. (ième ligne = ième colonne).

- Le nombre d'échantillons N est factorisable en un grand nombre de termes premiers entre eux.

$$N = \prod_{i=1}^m P_i \quad \text{avec } P_i : \text{ nombre premiers.}$$

Ces nombres premiers permettent d'accélérer le processus d'analyse de Fourier, en décomposant la matrice $(N.N)$ en un produit de plusieurs matrices.

Considérons un cas particuliers pour voir le déroulement de la décomposition de la matrice.

$$\text{Soit : } N = 12 \quad \text{tel que } N = \prod_{i=1}^m P_i = 2.2.3.$$

$$X(K) = \sum_{n=0}^{N-1} x(n) W(nK) \quad \text{avec } W(nK) = \text{EXP}(-j2\pi nK/N)$$

On subdivise $X(K)$ en deux séquences :

- Séquence paire : $n = 2r$

- Séquence impair : $n = 2r+1$

$$X(K) = \sum_{r=0}^{N/2-1} x(2r)W(2rK) + \sum_{r=0}^{N/2-1} x(2r+1)W(2r+1)K$$

$$X(K) = \sum_{r=0}^{N/2-1} x(2r)W(2rK) + \sum_{r=0}^{N/2-1} x(2r+1)W(2r+1)K$$

$$X(K) = \sum_{r=0}^{N/2-1} x(2r)W(2rK) + W(K) \sum_{r=0}^{N/2-1} x(2r+1)W(rK)$$

$$\text{Sachant que } W(2rK) = (\text{EXP}(-j2\pi rK/N))^2 = W(-j2\pi rK/(N/2)) = W(rK)$$

$$X(K) = \sum_{r=0}^{N/2-1} x(2r)W(rK) + W(K) \sum_{r=0}^{N/2-1} x(2r+1)W(rK)$$

$$\text{On pose } X(K) = G(K) + W(K) \cdot H(K)$$

$$\text{Avec } G(K) = \sum_{r=0}^{N/2-1} x(2r)W(rK)$$

$$H(K) = \sum_{r=0}^{N/2-1} x(2r+1)W(rK)$$

Si la séquence $N/2$ DFT est encore divisible par 2 on écrit :

$$G(K) = \sum_{r=0}^{N/2-1} g(r)W(rK) = \sum_{\ell=0}^{N/4-1} g(2\ell)W(2\ell K) + \sum_{\ell=0}^{N/4-1} g(2\ell+1)W(2\ell+1)K$$

$$\text{Et que : } W(2\ell K) = W(\ell K)$$

$$G(K) = \sum_{\ell=0}^{N/4-1} g(2\ell)W(\ell K) + W(K) \sum_{\ell=0}^{N/4-1} g(2\ell+1)W(\ell K)$$

$$H(K) = \sum_{\ell=0}^{N/4-1} h(2\ell)W(\ell K) + W(K) \sum_{\ell=0}^{N/4-1} h(2\ell+1)W(\ell K)$$

$$\text{On pose } G(K) = I(K) + W(K)J(K)$$

$$H(K) = L(K) + W(K)M(K)$$

A partir de ces équations, on détermine le diagramme de fluence qui représente graphiquement l'Algorithme, en utilisant les noeuds et les transmittances des différentes branches ainsi que les itérations intermédiaires : voir (fig. 3)

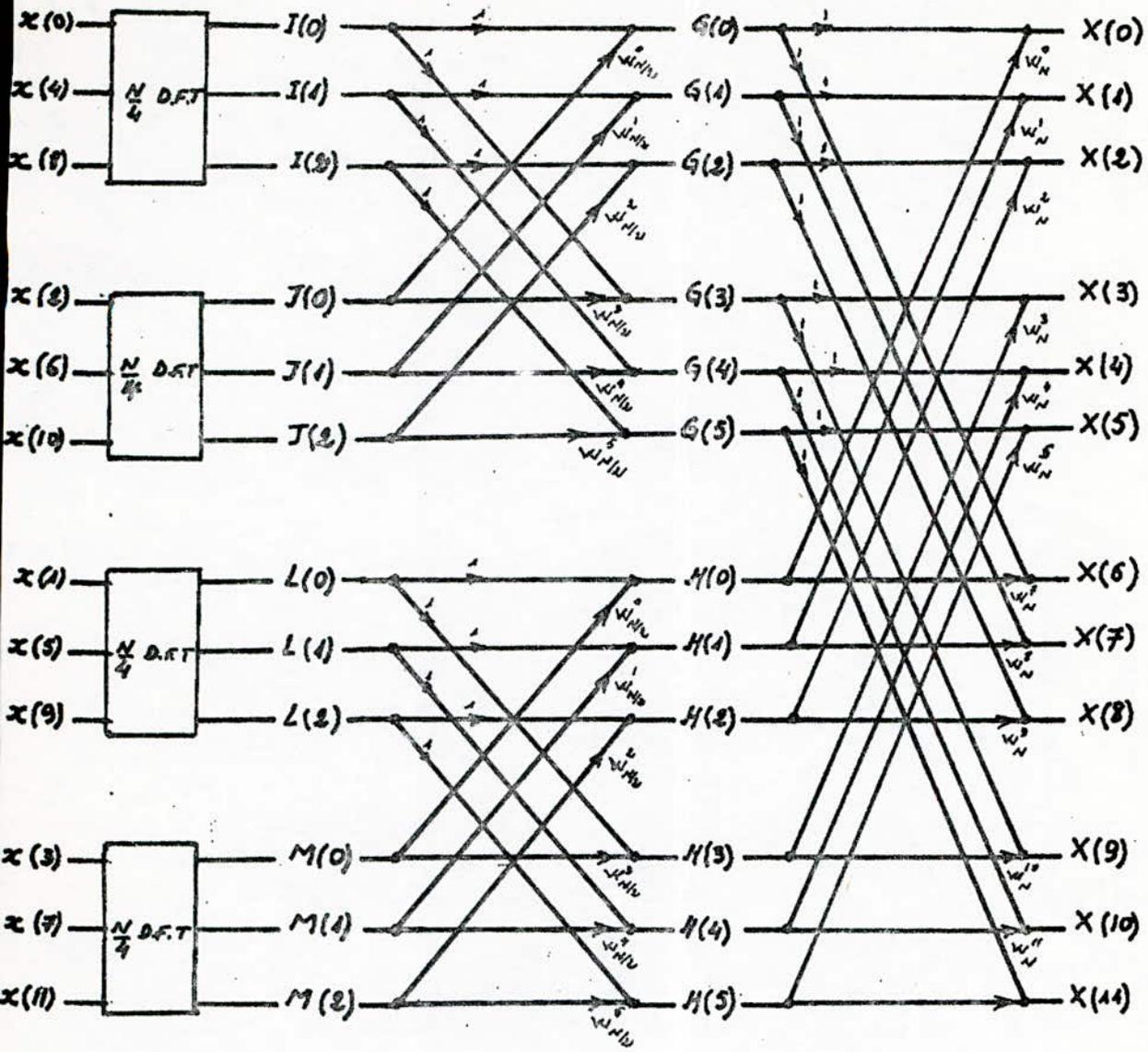
- Le désambrouillage apparaît sur ce diagramme au niveau des entrées.

Donc il faut un réarrangement des résultats enfin d'exécution.

Comme le nombre d'échantillon n'est pas une puissance de 2.

La méthode du binaire réfléchi n'est plus valable.

Il existe une méthode générale de décomposition de la matrice carrée d'ordre N de telle façon que le désambrouillage n'apparaît ni en entrée ni en sortie.



(fig: 3)

Diagramme de Fluence

Méthode de décomposition de la matrice

Si $N = N'r$, alors la matrice (M) est factorisable. Le rang J^{eme} de la matrice (M) s'écrit:

$$m(j)^t = (m(j1) m(j2);;..m(jr) \quad W_{m(j1)}^{r(j-1)} \cdot W_{m(j2)}^{r(j-1)} ;;..W_{m(jr)}^{r(j-1)} ;;.. \\ W_{m(j1)}^{N'r(j-1)} \cdot W_{m(j2)}^{N'r(j-1)} \dots W_{m(jr)}^{N'r(j-1)})$$

t : transposé

] : colonne de la matrice

pour simplifier la notation on pose :

$$m(j1).m(j2);;..m(jr) = a_j$$

$$m(j)^t = (a_j \quad W_{a_j}^{r(j-1)} \dots W_{a_j}^{N'r(j-1)})$$

Sachant que W est périodique : $W^{r(j-1)} = W^{r(j-1)} \pmod{N'}$

Soit $r(j-1) \pmod{N} = r(j-1) \pmod{N'}$

L'équation (1) ayant N' forme distinctes, chacune $(j-1) \pmod{N}$

$$\begin{aligned} & (a_j \quad a_j \quad a_j \quad \dots a_j) \quad (j-1) \pmod{N'} = 0 \\ & (a_j \quad W_{a_j}^r \quad W_{a_j}^{2r} \dots W_{a_j}^{r(N'-1)}) \quad (j-1) \pmod{N'} = 1 \\ m_j^t & (a_j \quad W_{a_j}^{2r} \quad W_{a_j}^{4r} \dots W_{a_j}^{2r(N'-1)}) \quad (j-1) \pmod{N'} = 2 \\ & \dots \dots \dots \\ & (a_j \quad W_{a_j}^{(N'-1)r} \quad W_{a_j}^{2(N'-1)r} ;;.. W_{a_j}^{(N'-1)(N'-1)r}) \quad (j-1) \pmod{N'} = N'-1 \end{aligned}$$

On peut trouver un produit de 2 matrices tel que :

$$\begin{bmatrix} a1 & Qr & Qr; \dots Qr \\ Qr & a2 & Qr; \dots Qr \\ Qr & Qr & a2 \dots Qr \\ \dots \dots \dots \\ Qr & Qr & Qr & aN' \end{bmatrix} \begin{bmatrix} Ur & Ur & Ur \dots Ur \\ Ur & W_{Ur}^r & W_{Ur}^{2r}; \dots W_{Ur}^{(N'-1)r} \\ Ur & W_{Ur}^{2r} & W_{Ur}^{4r}; \dots W_{Ur}^{(N'-2)r} \\ \dots \dots \dots \\ Ur & W_{Ur}^{(N'-1)r} & W_{Ur}^{(N'-2)r}; \dots W_{Ur}^r \end{bmatrix}$$

Avec Qr : submatrice nulle d'ordre $(1Xr)$

Ur : submatrice unitée d'ordre (rXr)

On obtient des submatrices NXN' , de rang $1Xr$, disposées de la façon suivante.

$$\begin{matrix}
 & 1 & 1 & \dots & 1 & 0 & 0 & \dots & 0 & \bullet & \bullet & \dots & \bullet & \dots & 0 & \bullet & \dots & 0 \\
 & 0 & 0 & \dots & 0 & 1 & W & \dots & W^{r-1} & \bullet & 0 & \dots & \bullet & \dots & \bullet & \bullet & \dots & 0 \\
 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 1 & W^2 & \dots & W^{2(r-1)} & \dots & 0 & \dots & \dots & 0 \\
 & \cdot & \cdot & \dots & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot & \dots & \cdot & \dots & \cdot & \cdot & \dots & \cdot \\
 & \cdot & \cdot & \dots & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot & \dots & \cdot & \dots & \cdot & \cdot & \dots & \cdot \\
 F_1 = & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & \dots & 1 & W^{N'-1} & \dots & W^{(N'-1)(r-1)} \\
 & 1 & W^{N'} & \dots & W^{N'(r-1)} & 0 & 0 & \dots & 0 & \bullet & 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 \\
 & 0 & 0 & \dots & 0 & 1 & W^{N'+1} & \dots & W^{(N'+1)(r-1)} & \bullet & 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 \\
 & \cdot & \cdot & \dots & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot & \dots & \cdot & \dots & \cdot & \cdot & \dots & \cdot \\
 & \cdot & \cdot & \dots & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot & \dots & \cdot & \dots & \cdot & \cdot & \dots & \cdot \\
 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & \dots & 1 & W^{N'(r-1)} & \dots & W^{(N'r-1)(r-1)}
 \end{matrix}$$

Le 1^{er} facteur est de rang N' X N', de 1Xr submatrice
 Le 2^{eme} facteur est de la même forme que la matrice (M) sauf que la submatrice interne sont réduites à l'unité, proportionnel à W^r.

Conséquences :

D'une manière récursive , la factorisation continue jusqu'a ce que N ne sera plus décomposable.

On aura :

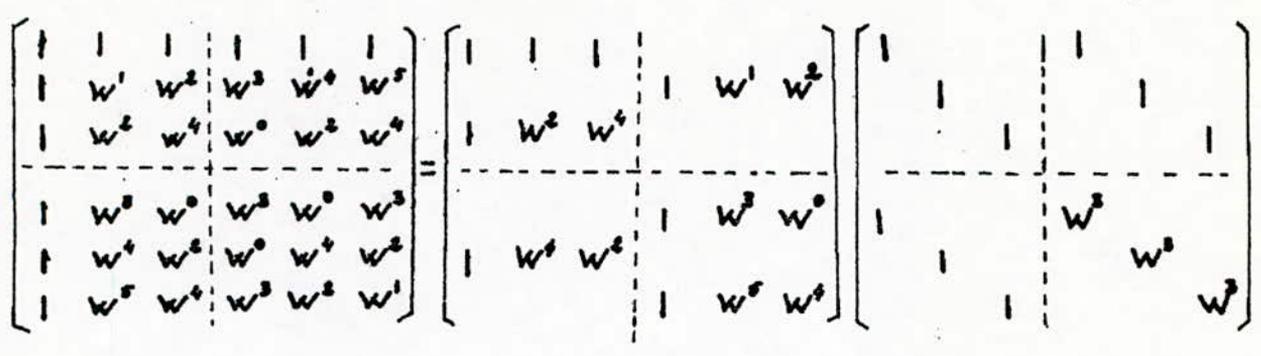
$$\begin{aligned}
 M &= F_1.M' \\
 M &= F_1.F_2.M'' \\
 M &= F_1.F_2.F_3.M''' \\
 &\dots \\
 &\dots \\
 M &= F_1.F_2.F_3 \dots F_k
 \end{aligned}$$

Avec k : nombre de factorisation de N en nombre premier

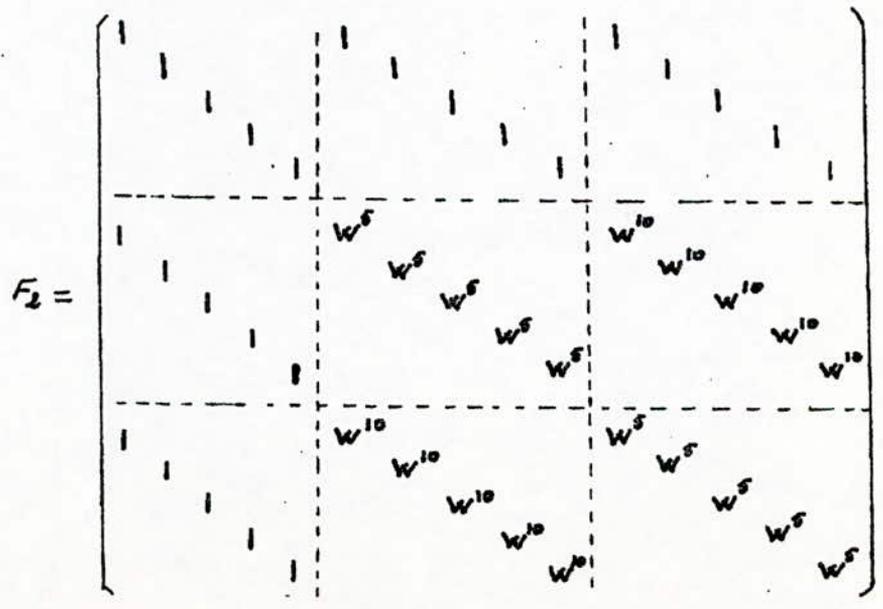
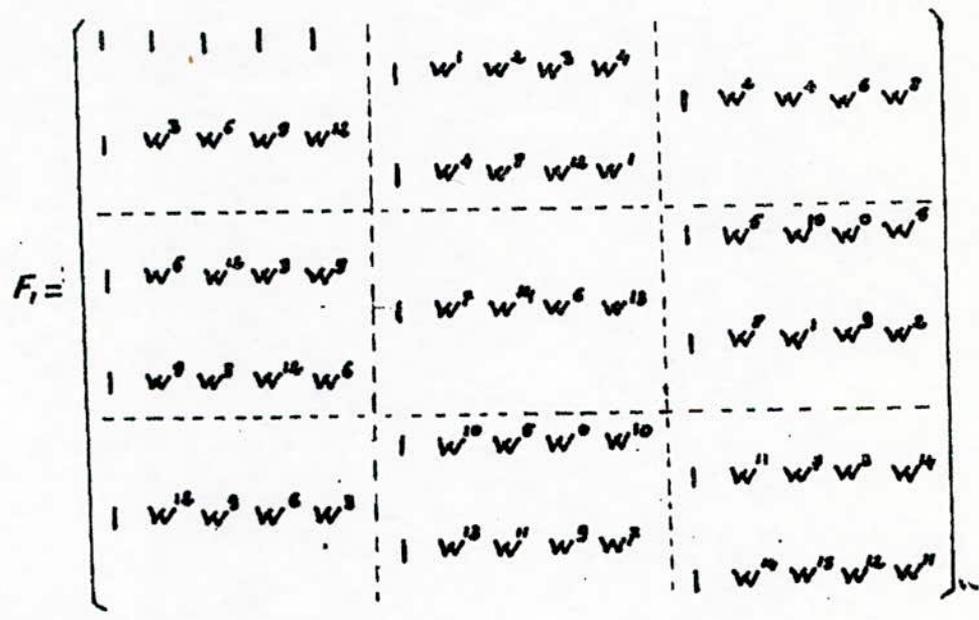
Fi: matrice de passage

Quelques exemples de factorisation de la matrice (M)

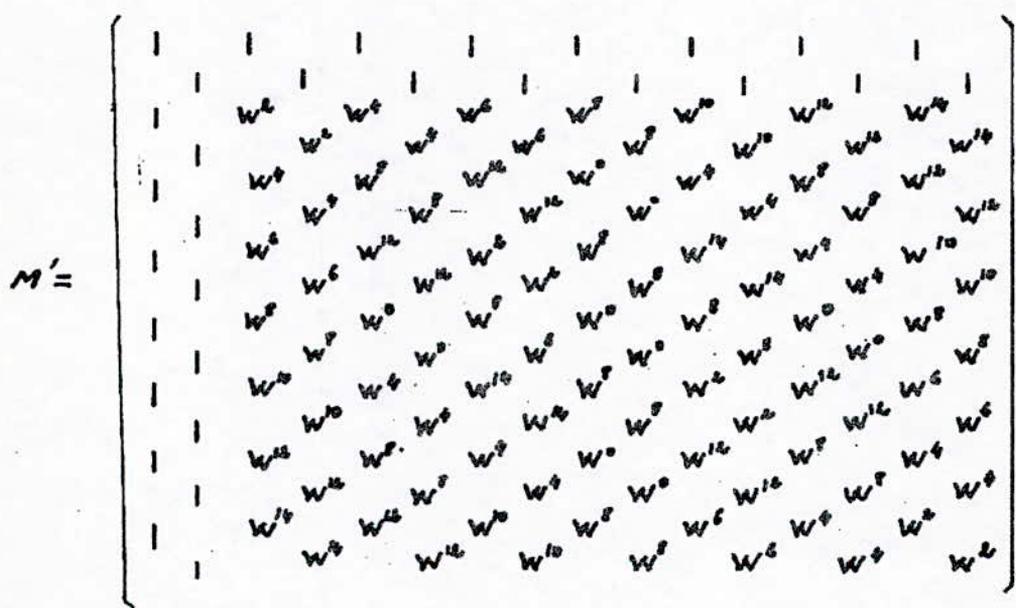
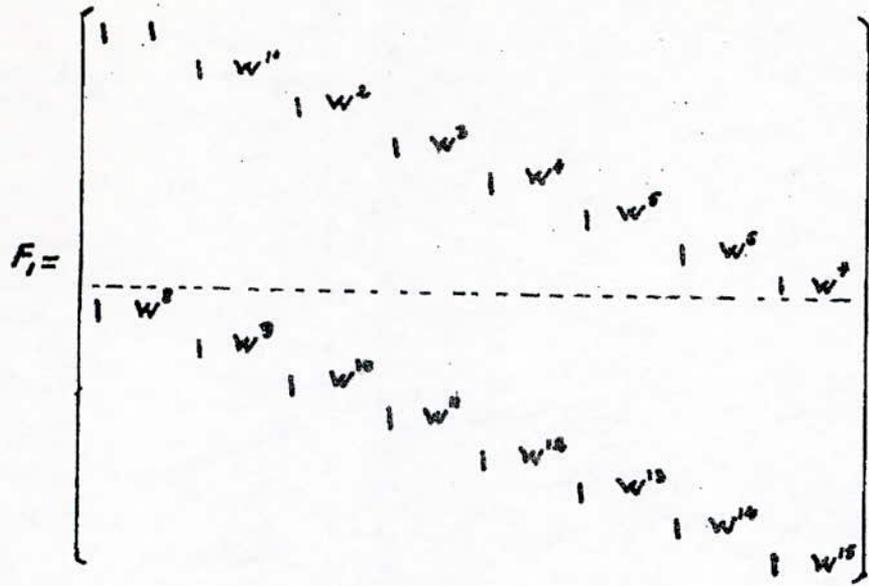
$$N = 6 \implies \begin{matrix} N = 2 \cdot 3 \\ k = 2 \end{matrix} \implies M = F_1 \cdot F_2$$



$$N = 15 \implies \begin{matrix} N = 3 \cdot 5 \\ k = 2 \end{matrix} \implies M = F_1 \cdot F_2$$



$$N = 16 \implies \begin{matrix} N = 2 \cdot 2 \cdot 2 \cdot 2 \\ R = 4 \end{matrix} \implies M = F_1 \cdot F_2 \cdot F_3 \cdot F_4$$



III COMPARAISON ENTRE LA DFT ET LA FFT

- La DFT est un outil valable pour une séquence de donnée limitée.

Quand le nombre d'échantillon augmente le nombre de multiplication et d'addition devient très important soit l'expression de la transformée de fourier directe :

$$X(K) = \sum_{n=0}^{N-1} x(n) W^{nK} \quad K=0, 1, \dots, (N-1)$$

On obtient :

N.N : nombre de multiplication complexe

N.(N-1) : nombre d'addition complexe

La FFT accélère le processus de traitement, en diminuant le nombre d'opération ; voir (fig : 4)

Soient :

- L'équation matricielle

$$X(K) = (M) \cdot x(n) \quad \text{avec : } K=0, 1, \dots, N-1$$

$$n=0, 1, \dots, N-1$$

- La décomposition de la matrice :

$$(M) = F_1 \cdot F_2 \cdot F_3 \dots F_k \quad \text{avec } k : \text{ nombre de factorisation}$$

Le nombre d'opérations à effectuer est donné par :

- nombre d'opération de multiplications

$$P(N) = N \cdot \sum_{i=1}^m (P_i - 1) - (N - 1)$$

- nombre d'opération d'addition

$$S(N) = N \cdot \sum_{i=1}^m (P_i - 1)$$

Avec: P_i :facteur premier de N

$i = 0, 1, \dots, m$, nombre de décomposition

Cas particulier:

$$\left. \begin{array}{l} \text{Si } P_i = P \\ N = \prod_{i=1}^m P_i \end{array} \right\} N = P^m$$

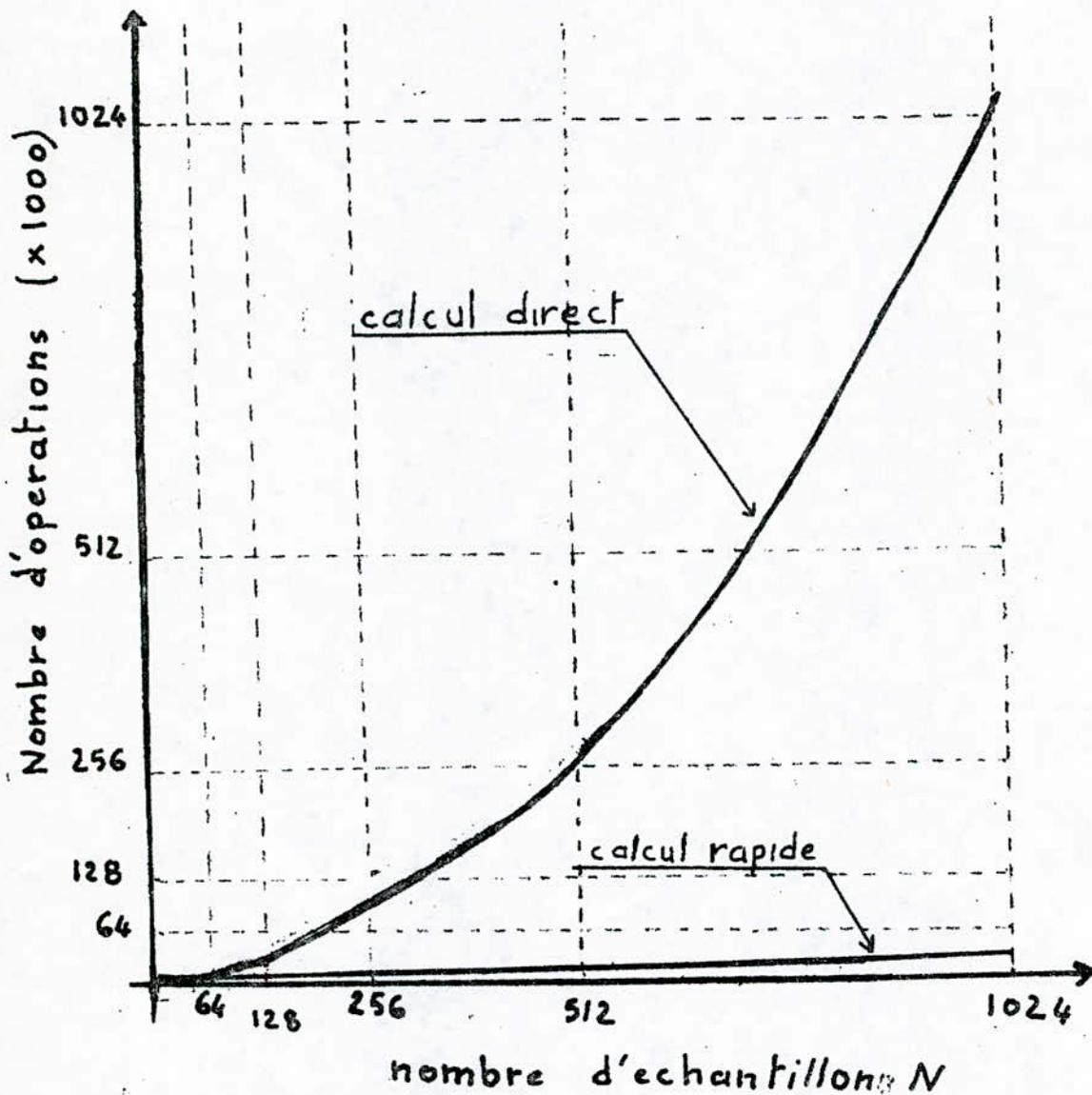
$$S(N) = N \cdot m(P - 1)$$

$$P(N) = N \cdot m(p - 1) - (N - 1)$$

Tableau de comparaison

Algorithme	Formule	Nombre d'opérations	
		Sommes : S(N)	Multiplication P(N)
DFT	$X(K) = \sum_{n=0}^{N-1} x(n) W^{nK}$	N . (N-1)	N . N
FFT	$X(K) = (F_1 F_2 \dots F_k) x(n)$	$N \cdot \sum_{i=1}^m (P_i - 1)$	$N \cdot \sum_{i=1}^m (P_i - 1) - (N - 1)$

Courbe représentative du nombre d'opérations effectuées
par le calcul direct et le calcul rapide en fonction
du nombre d'échantillon N



(fig: 4)

IV - APPLICATIONS DE LA TRANSFORMÉE DE FOURIER

Tout phénomène physique, qu'il soit d'ordre mécanique, électrique, thermique, électromagnétique, acoustique, hydraulique ou biologique, peut être caractérisé par sa réponse en fréquence.

En général, on désire savoir comment ce système répondra à chaque composante fréquentielle d'un signal d'entrée arbitraire.

Pour un système linéaire, la réponse fréquentielle caractérise complètement ce système. Par contre si le système n'est pas linéaire, il ne sera caractérisé que pour des conditions opératoires spécifiques. Les applications de l'analyse de Fourier sont diverses.

A titre d'exemple; nous citons:

a) FONCTION DE TRANSFERT :

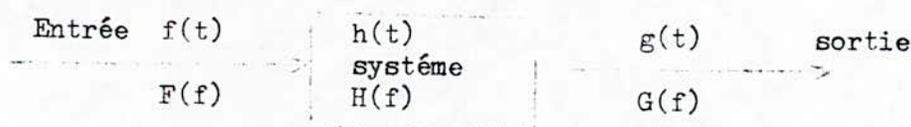
La fonction de transfert est la description mathématique de la relation Entrée / Sortie d'un système, qui peut être un filtre électronique, un moteur d'automobile, une aile d'avion en vibration ou un organe du corps humain.

Les mesures des fonctions de transferts sont désignées en fonction des applications:

- réponse en fréquence
- masse dynamique
- Impédance mécanique
- capacité acoustique
- mobilité inertance

ON utilise la fonction de transfert pour caractériser les systèmes asservis éliminer les vibrations des ailes d'avions ou pour surveiller les mouvements mécaniques dans le cœur d'un réacteur nucléaire.

Soit $h(t)$: la réponse impulsionnelle du système, $H(f)$: sa fonction de transfert :



Le signal de sortie peut être obtenu par la convolution.

$$g(t) = f(t) * h(t) = \int f(\xi) \cdot h(t-\xi) d\xi$$

Dans le domaine des fréquences on a : $G(f) = H(f) \cdot F(f)$

$$H(f) = G(f) / F(f)$$

On écrit : $H(f) = |H(f)| \exp(j\theta_f)$

Sachant : $|H(f)| = |G(f)| / |F(f)|$: l'amplitude de la fonction de transfert.

$\theta = \theta_g - \theta_f$: la phase de la fonction de transfert.

2 - FONCTION DE COHERENCE.

La fonction de cohérence $a^2(f)$ est une mesure de degré de causalité entre l'entrée et la sortie d'un système, lorsque cette fonction de cohérence est égale à 1, cela signifie que la totalité du signal de sortie, pour cette fréquence est dû au signal d'entrée mesuré. Quant elle est inférieure à 1, il y a présence des signaux parasites et des retards dans le système.

On peut également considérer la fonction de cohérence comme une mesure du rapport signal / bruit.

ceci ressort de l'équation .

$$S(f)/N(f) = a^2(f) / (1 - a^2(f))$$

La fonction de cohérence est définie par :

$$a^2(f) = \overline{H(f)}^2 G_{xx}(f) / G_{yy}(f)$$

Avec $\overline{\quad}$: moyenne des valeurs.

3- FILTRAGE NUMERIQUE

Le filtrage numérique est un sujet très vaste, nous contenterons de donner un seul exemple d'application .

Soit un système défini par sa fonction de transfert $H(f)$ et sa réponse impulsionnelle $h(t)$.

On peut définir théoriquement sa courbe en fréquence, sachant que ce système est considéré comme étant un filtre. On peut alors numériquement calculer les paramètres de la réponse impulsionnelle $h(f)$.

Pour filtrer un signal numérique $f(t)$, il suffit alors de faire la convolution de ce signal avec la réponse impulsionnelle du filtre $h(t)$.

$$g(t) = h(t) * f(t)$$

4 - APPLICATIONS MEDICALES.

Les techniques de la transformée de Fourier sont surtout appliquées au dépouillement d'ELECTRO-ENCEPHALO-GRAMMES (E.E.G.).

Plusieurs méthodes peuvent être utilisées.

- Soit l'analyse en temps réel par un analyseur de Fourier
- Soit la numérisation du signal, et son étude par transformée de

Fourier réalisée sur un mini-ordinateur.

Le signal d' E.E.G. est considéré comme une variable aléatoire.

Un des paramètres le plus généralement étudiés est le spectre de puissance défini par : $S(f) = E(X(f) \cdot X^*(f))$

où $X(f)$: est la transformée de Fourier du signal

$X^*(f)$: est la transformée complexe conjuguée

$E(\quad)$: représente la valeur probable de ce qui se trouve entre crochets

REMARQUE:

Pour diminuer l'erreur entre le spectre mesuré et le spectre réel, il est nécessaire d'effectuer un moyennage sur plusieurs spectres

Un autre paramètre important est le CROSS-SPECTRE entre deux enregistrements simultanés d'E.E.G. défini par :

$$S_{xy}(f) = E(X_x(f) \cdot X_y^*(f))$$

Ce cross spectre permet de calculer deux informations importantes qui sont les spectres de cohérence défini par :

$$C_{xy}(f) = | S_{xy}(f) |^2 / S_x(f) \cdot S_y(f)$$

Et le spectre de phase:

$$Q(f) = \arg S_{xy}(f)$$

Tous les paramètres cités ci dessus peuvent être calculés actuellement numériquement; d'une façon plus précise, plus rapide et plus fiable. Ceci grâce à la transformée de Fourier rapide.

C H A P I T R E I I I

PROGRAMMATION DE LA TRANSFORME DE FOURIER

Toute l'étude théorique de la transformé de Fourier peut se résumer par des algorithmes:

- Algorithme de la DFT (Discret Fourier Transform~~X~~)
- Algorithme de la FFT (Fast Fourier Transform)

I- Organigrammes des Algorithmes suivants :

- DFT : Voir fig (5a)
- FFT : Voir fig (5b)

II Programmes en FORRRAN des Algorithmes suivants :

- DFT : Voir listing
- FFT : Voir listing

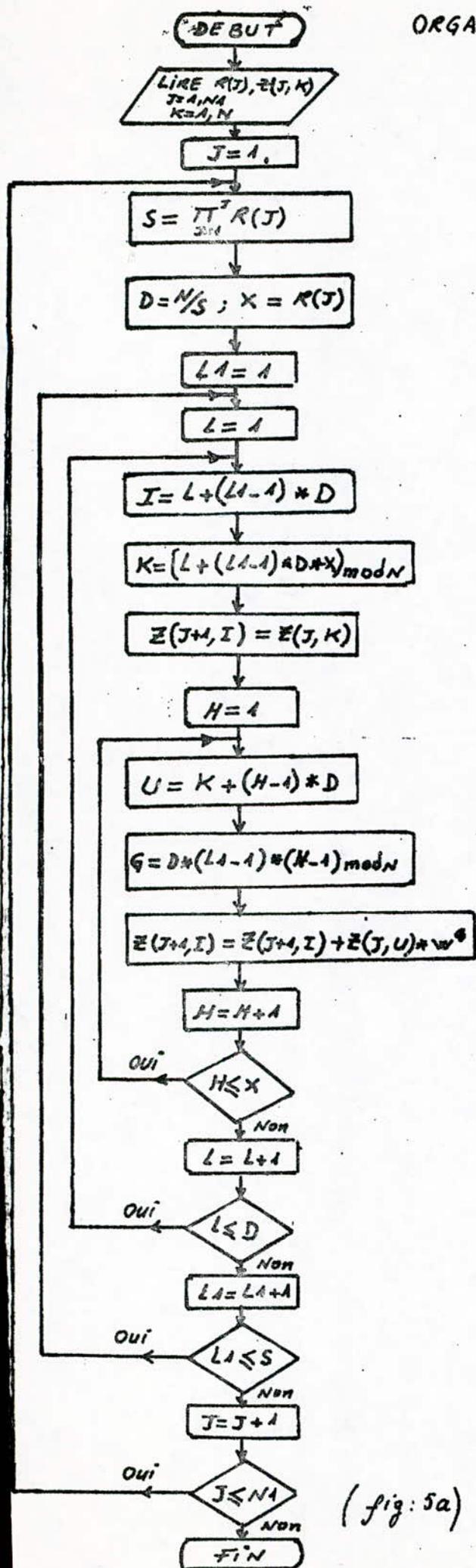
III - Résultats obtenus par les deux méthodes appliqués sur quelques signaux.

Les valeurs des parties réelles et imaginaire sont indiqués respectivement sur chaque spectre .

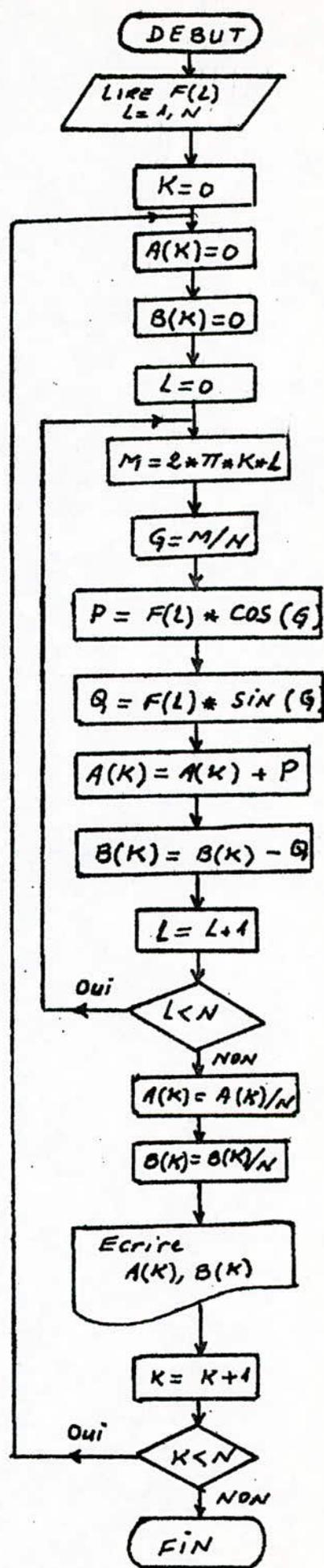
REMARQUE :

L'application de la DFT où la FFT sur n'importe quel signal, leurs résultats sont conformes, équivalants de 10^{-5} près. Par contre l'écart de temps d'exécution augmente de plus en plus que le nombre d'échantillon augmente.

ORGANIGRAMMES



(fig: 5a)



(fig: 5b)

LE PROGRAMME DE LA DFT

```

C TRANS DE FOURIER DIRECT
DIMENSION IVECT(100)
DATA IPIALC, TASTER, IPLUS /1E , 1H*, 1H*/
DIMENSION A(100), B(100)
REAL(105, 100)I
100 FORMAT(1X, I4)
CALL CALCL (A, B, N)
DO 200 II=1, 100
  IVECT(II)=IPLUS
200 CONTINUE
PRINT 205
205 FORMAT(1F1)
PRINT 204
204 FORMAT(20X, 72(1H*), //, 20X, 1F*, 'GRAPHE DE LA TRANSFORME DE FOURIER
  *F(X)=X(T)', //, 20X, 72(1H*), //)
PRINT 201, (IVECT(JJ), JJ=1, 100)
201 FORMAT(///, 10X, 100(A1), /)
DO 206 II=1, 100
  IVECT(II)=IRLANC
206 CONTINUE
DO 202 K=1, I
  IVECT(20)=IPLUS
  TS=A(K)*A(K)+B(K)*B(K)
  TS=100*SQRT(TS)
  II=TS+20
  IVECT(II)=TASTER
  PRINT 203, IVECT
203 FORMAT(10X, 100(A1), /)
  IVECT(II)=IRLANC
202 CONTINUE
STOP
END

```

```

SUBROUTINE CALCL(A, B, N)
DIMENSION F(100), A(50), B(50)
REAL(105, 20)(F(L), L=1, N)
20 FORMAT(10(F8.2))
DO 70 K=1, N
  A(K)=0.
  B(K)=0.
DO 80 L=1, N
  LV=L-1
  KV=K-1
  ARG=2.*3.14159*FLOAT(KV)*FLOAT(LV)/FLOAT(N)
  A(K)=A(K)+F(L)*COS(ARG)
  B(K)=B(K)-F(L)*SIN(ARG)
80 CONTINUE
  A(K)=A(K)/N
  B(K)=B(K)/N
  PRINT 104, (K, A(K), B(K))
104 FORMAT(2X, I2, 2(5X, F10.4))
70 CONTINUE
PRINT 102, (F(L), L=1, N)
102 FORMAT(4X, 6(2X, F8.2), //)
RETURN
END

```

LE PROGRAMME DE LA FFT

```

C TRANS DE FOURIER EN FFT
COMMON N1
DIMENSION IVECT(100)
DATA IRLANC, TASTER, IPLUS /1H, 1H*, 1H*/
DIMENSION RZ(20, 100), AIZ(20, 100)
INTEGER S, D, R, H, G, U, X
REAL(105, 100)N
100 FORMAT(1X, I4)
CALL CALCUL (N7, AI7, N)
DO 200 II=1, 100
  IVECT(II)=IPLUS
200 CONTINUE
  PRINT 205
205 FORMAT(1F1)
  PRINT 204
204 FORMAT(20X, 72(1H*), //, 20X, 1F*, 'GRAPHE DE LA TRANSFORME RAPIDE DE
  *F(X)=X(T)', //, 20X, 72(1H*), //)
  PRINT 201, (IVECT(JJ), JJ=1, 100)
201 FORMAT(///, 10X, 100(A1), /)
  DO 206 II=1, 100
    IVECT(II)=IRLANC
206 CONTINUE
  DO 202 I=1, N
    IVECT(20)=IPLUS
    TS=07(N1+1, I)*RZ(N1+1, I)+AI7(N1+1, I)*AIZ(N1+1, I)
    TS=50*SLFT(TS)
    II=TS+20
    IVECT(II)=TASTER
  PRINT 203, IVECT
203 FORMAT(10X, 100(A1), /)
  IVECT(II)=IRLANC
202 CONTINUE
  STOP
  END

SUBROUTINE CALCUL (RZ, AIZ, N)
COMMON N1
DIMENSION RZ(20, 100), AIZ(20, 100), R(20)
INTEGER S, D, R, H, G, I, X
REAL(105, 15)N1
15 FORMAT(1X, I2)
READ(105, 16) (R(J), J=1, N1)
16 FORMAT(1X, 4(T2))
READ(105, 20) (R7(1, K), K=1, N)
READ(105, 20) (AIZ(1, K), K=1, N)
20 FORMAT(10(F8.2))
S=1
DO 65 J=1, N1
  S=S*R(J)
  Y=R(J)
  D=N/S
  DO 65 L1=1, S
    DO 65 L=1, D
      I=L+(L1-1)*D
      K1=L+(11-1)*D*X
      K=MOD(K1, N)
      RZ(J+1, I)=RZ(J, K)
      AI7(J+1, I)=AIZ(J, K)
      DO 44 H=2, X
        U=K+(H-1)*D
        G1=D*(L1-1)*(H-1)
        G=MOD(G1, N)
        ARG=2.*3.14159*FLOAT(G)/FLOAT(N)
        CO=COS(ARG)
        SI=SIN(ARG)
        RZ(J+1, I)=RZ(J+1, I)+RZ(J, U)*CO+AIZ(J, U)*SI
        AI7(J+1, I)=AIZ(J+1, I)+AIZ(J, U)*CO-FZ(J, U)*SI
44 CONTINUE
      IF(J.NE.N1) GO TO 65
      RZ(J+1, I)=RZ(J+1, I)/N
      AI7(J+1, I)=AI7(J+1, I)/N
      PRINT 10, (I, RZ(J+1, I), AI7(J+1, I))
10 FORMAT(2X, I3, 2(5X, E10.4))
65 CONTINUE
  DO 11 K=1, N
    PRINT 88, (K, RZ(1, K))
88 FORMAT(10X, I3, 2X, E10.4)
11 CONTINUE
  RETURN
  END

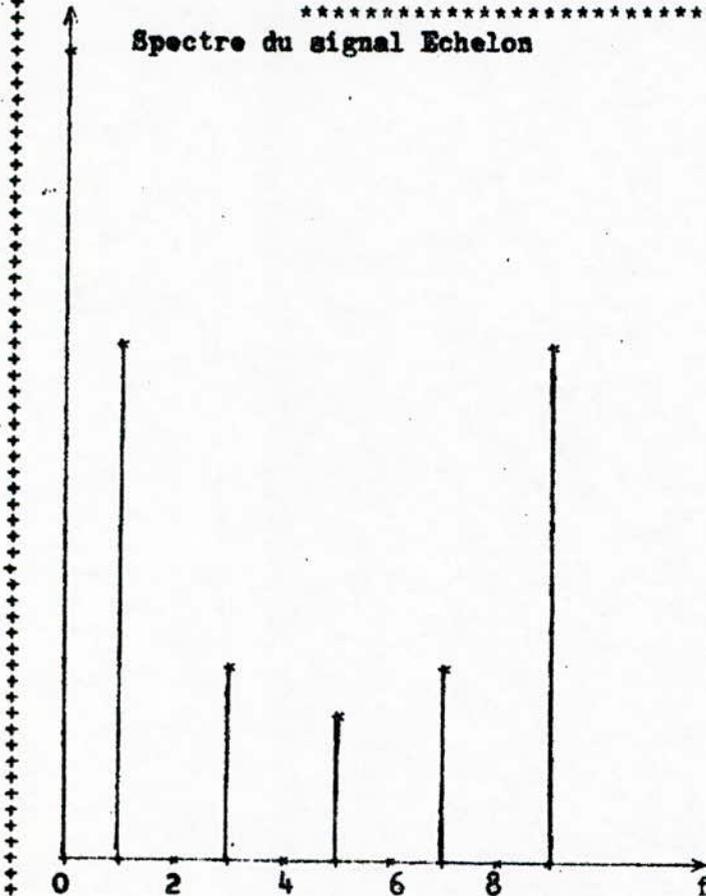
```

N	Coefficients de la DFT		Coefficients de la FFT	
	Re(X(K))	Im(X(K))	Re(X(K))	Im(X(K))
1	.5000E+00	.0000E+00	.5000E+00	.0000E+00
2	-.1000E+00	.3078E+00	-.1000E+00	.3078E+00
3	-.2265E-06	-.2901E-06	-.5245E-06	-.3159E-06
4	-.1000E+00	.7265E-01	-.1000E+00	.7265E-01
5	-.5543E-06	-.1771E-06	-.4470E-06	-.2980E-06
6	-.1000E+00	-.3595E-05	-.1000E+00	.5992E-06
7	.1723E-05	-.1490E-05	-.4470E-06	-.2980E-06
8	-.9999E-01	-.7265E-01	-.1000E+00	-.7265E-01
9	-.2068E-05	-.3207E-05	-.5245E-06	-.3036E-06
10	-.9998E-11	-.3078E+00	-.1000E+00	-.3078E+00

1.00 1.00 1.00 1.00
 Sequences de données

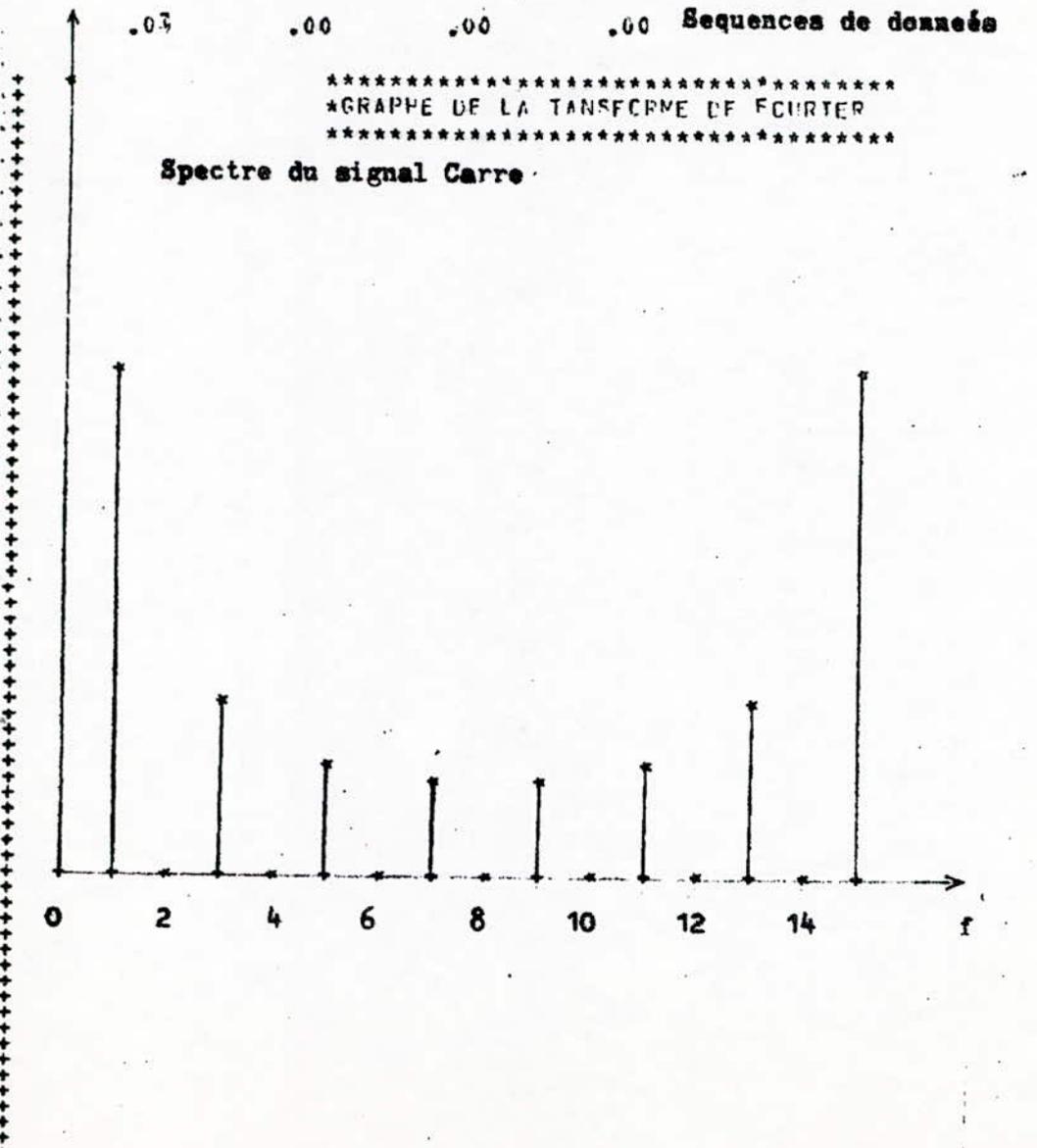
 *GRAPHE DE LA TRANSFORMEE DE FOURIER

Spectre du signal Echelon



N	Coefficients de la DFT		Coefficients de la FFT	
	Re(X(k))	Im(X(K))	Re	Im(X(K))
1	.5000E+00	.0000E+00	.5000E+00	.0000E+00
2	.6250E-01	-.3142E+00	.6250E-01	-.3142E+00
3	-.4321E-06	-.3771E-06	-.4521E-06	-.1873E-06
4	.6250E-01	-.9754E-01	.6250E-01	-.9354E-01
5	-.2207E-06	-.3576E-06	-.3745E-06	-.3745E-06
6	.6250E-01	-.4176E-01	.6250E-01	-.4176E-01
7	-.2421E-06	-.4470E-06	-.7756E-07	-.1873E-06
8	.6250E-01	-.1243E-01	.6250E-01	-.1243E-01
9	.0000E+00	-.7490E-06	.0000E+00	-.7490E-06
10	.6250E-01	.1243E-01	.6250E-01	.1243E-01
11	.4619E-06	-.7153E-06	.7756E-07	-.1873E-06
12	.6250E-01	.4176E-01	.6250E-01	.4176E-01
13	.5444E-06	-.6666E-06	.745E-06	-.3745E-06
14	.6250E-01	-.9754E-01	.6250E-01	.9354E-01
15	.3468E-05	-.9649E-06	.4521E-06	-.1873E-06
16	.6249E-01	.3142E+00	.6250E-01	.3142E+00

1.00 1.00 1.00 1.00 1.00 1.00
 1.00 1.00 .00 .00 .00 .00



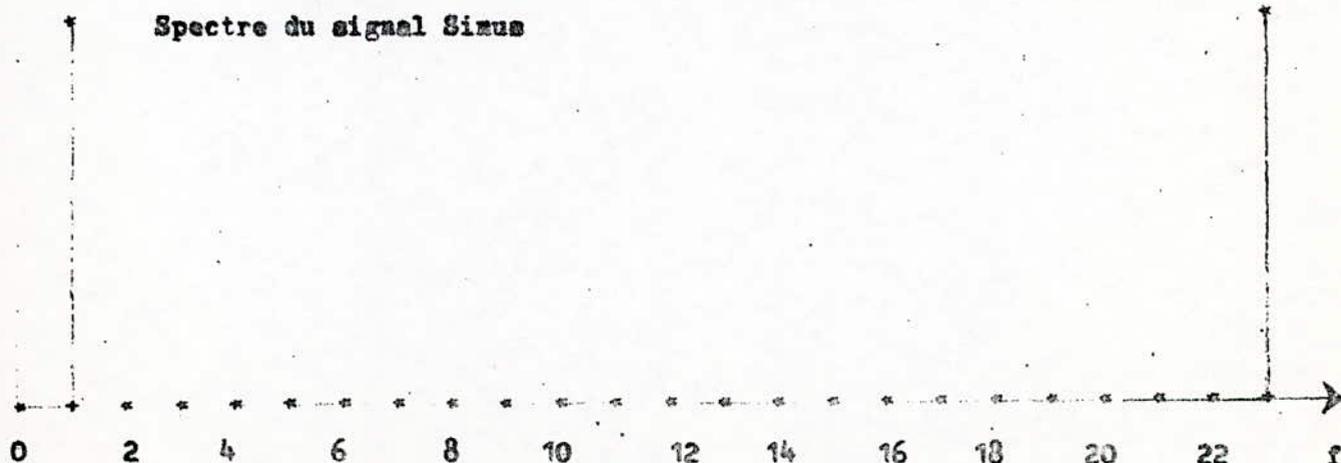
N	Coefficients de la DFT		Coefficients de la FFT	
	Re(X(K))	Im(X(K))	Re(X(K))	Im(X(K))
1	-.2061E-06	.0000E+00	.0000E+00	.0000E+00
2	.1195E-05	-.4970E+00	.1192E-05	-.4970E+00
3	-.6457E-07	.1987E-07	.0000E+00	.0000E+00
4	.2484E-07	.1179E-02	.2060E-06	.1179E-02
5	-.1664E-06	.4520E-06	.0000E+00	.0000E+00
6	-.1021E-05	-.3153E-04	-.4357E-06	-.3125E-04
7	.2088E-06	.1257E-05	.0000E+00	.0000E+00
8	-.7252E-06	.1707E-02	.3576E-06	.1708E-02
9	-.1118E-06	-.1664E-06	.0000E+00	.0000E+00
10	.2459E-06	.1178E-02	-.5166E-06	.1178E-02
11	-.1639E-06	.1267E-05	.0000E+00	.0000E+00
12	.6010E-06	.1303E-02	.1571E-06	.1302E-02
13	.0000E+00	-.6342E-06	.0000E+00	.0000E+00
14	-.2484E-07	-.1302E-02	-.3821E-06	-.1302E-02
15	-.1458E-05	.2310E-06	.0000E+00	.0000E+00
16	.8394E-06	-.1178E-02	.4371E-06	-.1179E-02
17	-.6855E-06	.2235E-06	.0000E+00	.0000E+00
18	.1055E-05	-.1708E-02	-.3974E-06	-.1708E-02
19	-.5007E-06	.9785E-06	.0000E+00	.0000E+00
20	-.5662E-06	.3223E-04	.1293E-06	.3232E-04
21	.7947E-07	.3735E-05	.0000E+00	.0000E+00
22	-.3229E-06	-.1175E-02	-.1945E-06	-.1179E-02
23	-.6929E-06	.4726E-05	.0000E+00	.0000E+00
24	-.3435E-04	.4970E+00	-.4450E-05	.4970E+00

Sequences de	.00	.25	.50	.70	.86	.96
données						
	1.00	.96	.86	.70	.50	.25
	.00	-.25	-.50	-.70	-.86	-.96
	-1.00	-.96	-.86	-.70	-.50	-.25

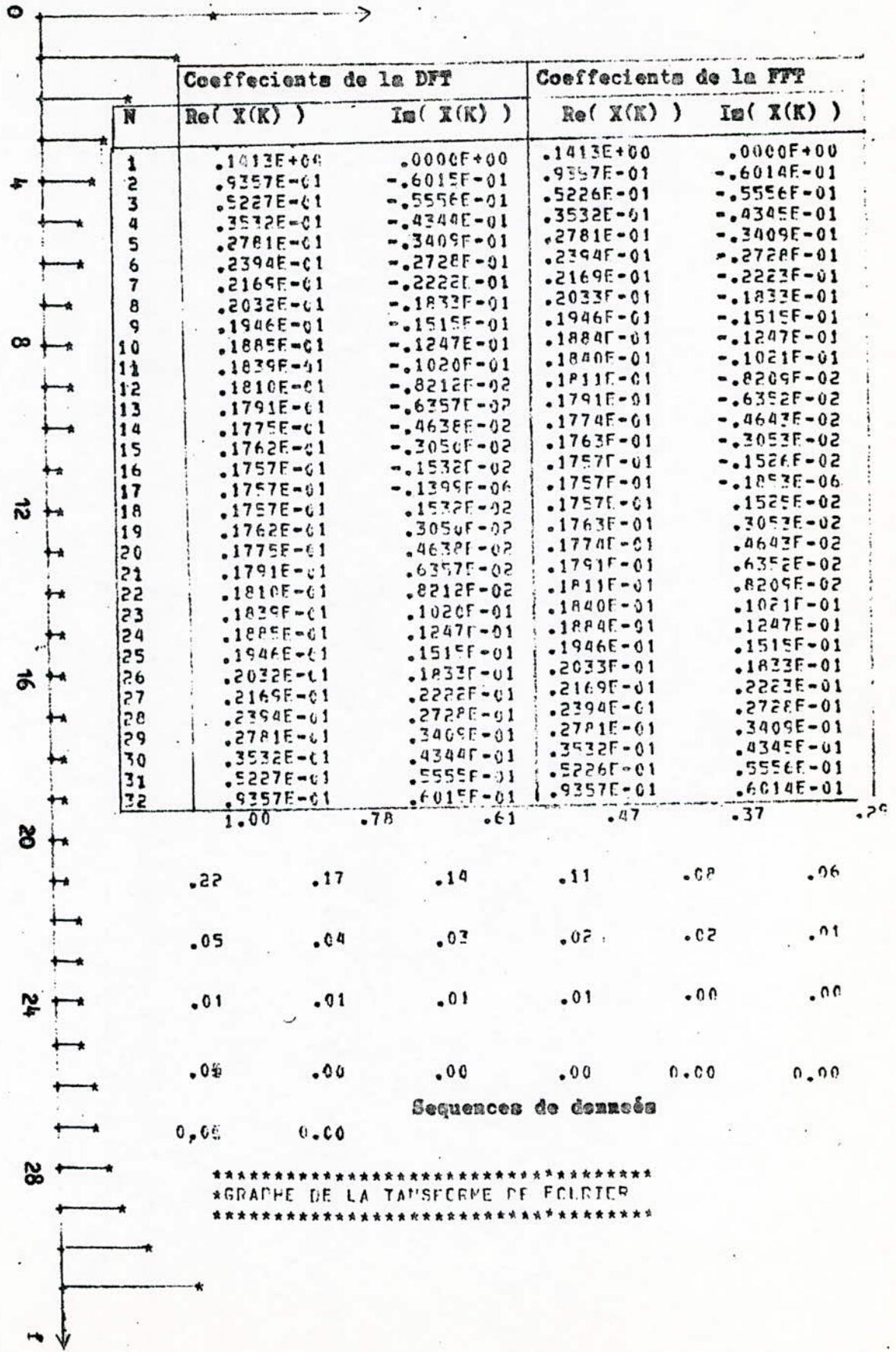
 *GRAPHE DE LA TRANSFORME DE FOURIER

A

Spectre du signal Sinus



Spectre du signal Exponentiel



N	Re(X(K))	Im(X(K))
1	.3146E+03	.0000E+00
2	.1999E+01	-.6040E+01
3	-.1542E+01	-.2057E+01
4	-.4007E+00	-.4764E+00
5	-.0649E+00	-.4554E+00
6	-.2731E+00	.4645E-01
7	.2088E+00	.9135E-02
8	.7997E-01	-.7539E-01
9	-.6565E-01	-.2649E-01
10	-.1779E+00	-.3789E-01
11	-.1820E-02	-.3501E-01
12	.1181E+00	-.8083E-01
13	-.6924E-01	.1714E-01
14	-.2988E-01	-.2847E-01
15	-.2314E-02	-.1868E+00
16	.5296E-01	-.9912E-01
17	.6078E-01	.2324E-01
18	-.5691E-02	.1041E-01
19	.2843E-01	-.1080E+00
20	.6649E-01	-.1072E+00
21	.8872E-01	.3491E-01
22	.1606E-01	.6527E-01
23	-.4329E-01	.1908E-01
24	.7487E-01	-.5431E-01
25	.1363E+00	-.1062E-02
26	.7574E-01	.5752E-01
27	-.4380E-01	-.1979E-01
28	.1669E-01	-.6601E-01
29	.8933E-01	-.3581E-01
30	.6646E-01	.1070E+00
31	.2815E-01	.1073E+00
32	-.4095E-02	-.1015E-01
33	.6022E-01	-.2504E-01
34	.5356E-01	.1004E+00
35	-.1574E-02	.1860E+00
36	-.2942E-01	.2835E-01
37	-.6795E-01	-.1817E-01
38	.1192E+00	.7950E-01
39	.7718E-03	.3421E-01
40	-.1776E+00	.3766E-01
41	-.6395E-01	.2636E-01
42	.8271E-01	.7506E-01
43	.2107E+00	-.9421E-02
44	-.2664E+00	-.4581E-01
45	-.4584E+00	.4547E+00
46	-.3928E+00	.4757E+00
47	-.1570E+01	.2055E+01
48	.2018E+01	.6037E+01

Le signal de la température
du sel.N=48

Les coefficients de la
DFT et la FFT sont identiques.

Sequences de données

	314.20	318.45	322.70	326.75	330.80	333.20
335.67	336.30	337.00	334.75	332.50	330.55	
328.67	328.05	327.50	324.55	321.60	319.15	
316.77	313.60	310.50	309.75	309.00	308.55	
308.17	308.00	307.90	307.65	307.40	307.40	
307.47	306.90	306.40	305.65	305.30	304.90	
304.57	300.61	303.70	303.45	303.20	303.00	
302.87	302.90	303.00	305.25	307.50	308.00	

IV Verification par la transformé de Fourier inverse.

1 Définition:

La transformé de Fourier ~~xxx~~ inverse est défini par l'équation algébrique :

$$x(n) = \sum_{k=0}^{N-1} X(K) \exp (+j2\pi nK/N)$$

Avec $X(K) = \text{Re}(X(K)) + j\text{Im}(X(K))$

2 -Programmation :

L'organigramme, et le programme sont identiques à celui de la transformé directe, seule l'expression algébrique change.

3 -Résultats :

On utilise les parties réelles et imaginaires, obtenus par la transformé directe, comme une séquence de données. On retrouve bien nos signaux.

V CONCLUSION :

L'étude de la transformé de Fourier, l'application des algorithmes de la DFT et FFT aux différents signaux, traités par un calculateur numériques. (le MITRA 125), montrent la rapidité de la FFT par rapport à la DFT. Surtout quand le nombre (N) d'échantillon augmente.

A l'aide du mini ordinateur (MITRA 125), le temps d'exécution d'un traitement du signal d'échantillon N=180 valeurs pour :

La transformé de Fourier rapide (FFT) est de 9 secondes.

La transformé de Fourier discrète (DFT) est de 35 secondes.

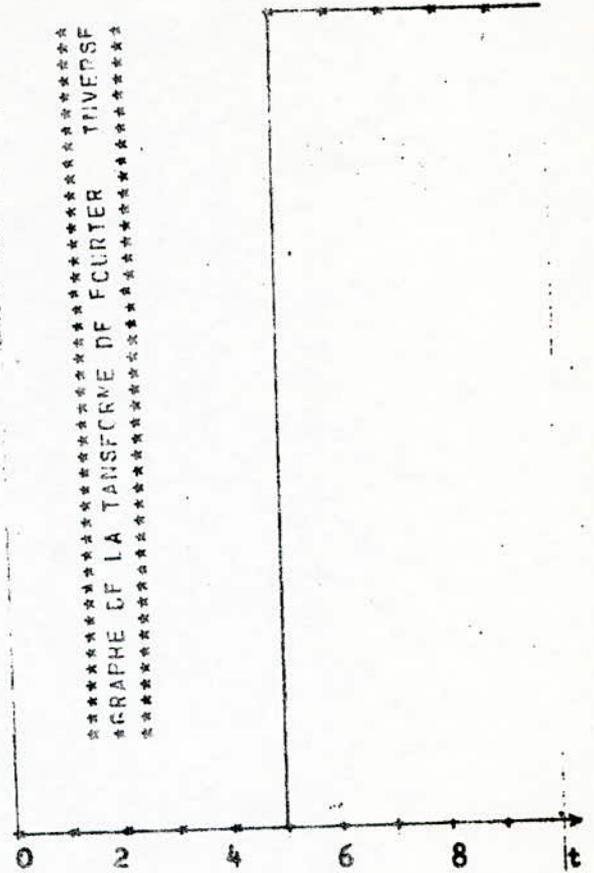
Coeff. de la transformée Inverse

N	Re(X(K))	Im(X(K))
1	-.1192E-06	.0000E+00
2	-.2962E-04	.1172E-05
3	-.6537E-04	.1073E-05
4	-.6537E-04	.1788E-05
5	-.2819E-04	.9577E-06
6	.1000E+01	-.1729E-05
7	.1000E+01	-.2027E-05
8	.1000E+01	-.7636E-05
9	.1000E+01	-.2027E-05
10	.1000E+01	-.2801E-05

Sequences de données

1	.5000E+00	.0000E+00
2	-.1000E+00	.7078E+00
3	.0000E+00	.0000E+00
4	-.1000E+00	.7265E-01
5	.0000E+00	.0000E+00
6	-.1000E+00	.0000E+00
7	.0000E+00	.0000E+00
8	-.1000E+00	-.7265E-01
9	.0000E+00	.0000E+00
10	-.1000E+00	-.7078E+00

Le signal Echelon



GRAPHE DE LA TRANSFORME DE FOURIER INVERSE

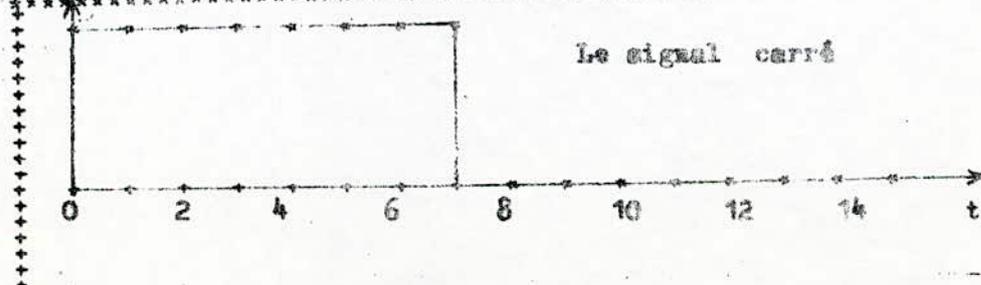
N	Re(X(K))	Im(X(K))
1	.1000E+01	-.1192E-06
2	.1000E+01	-.1729E-05
3	.1000E+01	-.1550E-05
4	.1000E+01	-.2623E-05
5	.1000E+01	-.1264E-05
6	.1000E+01	-.7338E-05
7	.1000E+01	-.2384E-05
8	.1000E+01	-.3934E-05
9	.3572E-12	.1617E-05
10	.5484E-05	.2980E-05
11	.4530E-05	.2623E-05
12	.2027E-04	.4113E-05
13	.2074E-04	.2247E-05
14	.2187E-04	.3934E-05
15	.6674E-05	.7815E-05
16	.1001E-04	.4590E-05

Sequences de données

1	.5000E+00	.0000E+00
2	.6250E-01	-.3142E+00
3	.0000E+00	.0000E+00
4	.6250E-01	-.9354E-01
5	.0000E+00	.0000E+00
6	.6250E-01	-.4176E-01
7	.0000E+00	.0000E+00
8	.6250E-01	-.1243E-01
9	.0000E+00	.0000E+00
10	.6250E-01	.1243E-01
11	.0000E+00	.0000E+00
12	.6250E-01	.4176E-01
13	.0000E+00	.0000E+00
14	.6250E-01	.9354E-01
15	.0000E+00	.0000E+00
16	.6250E-01	.3142E+00

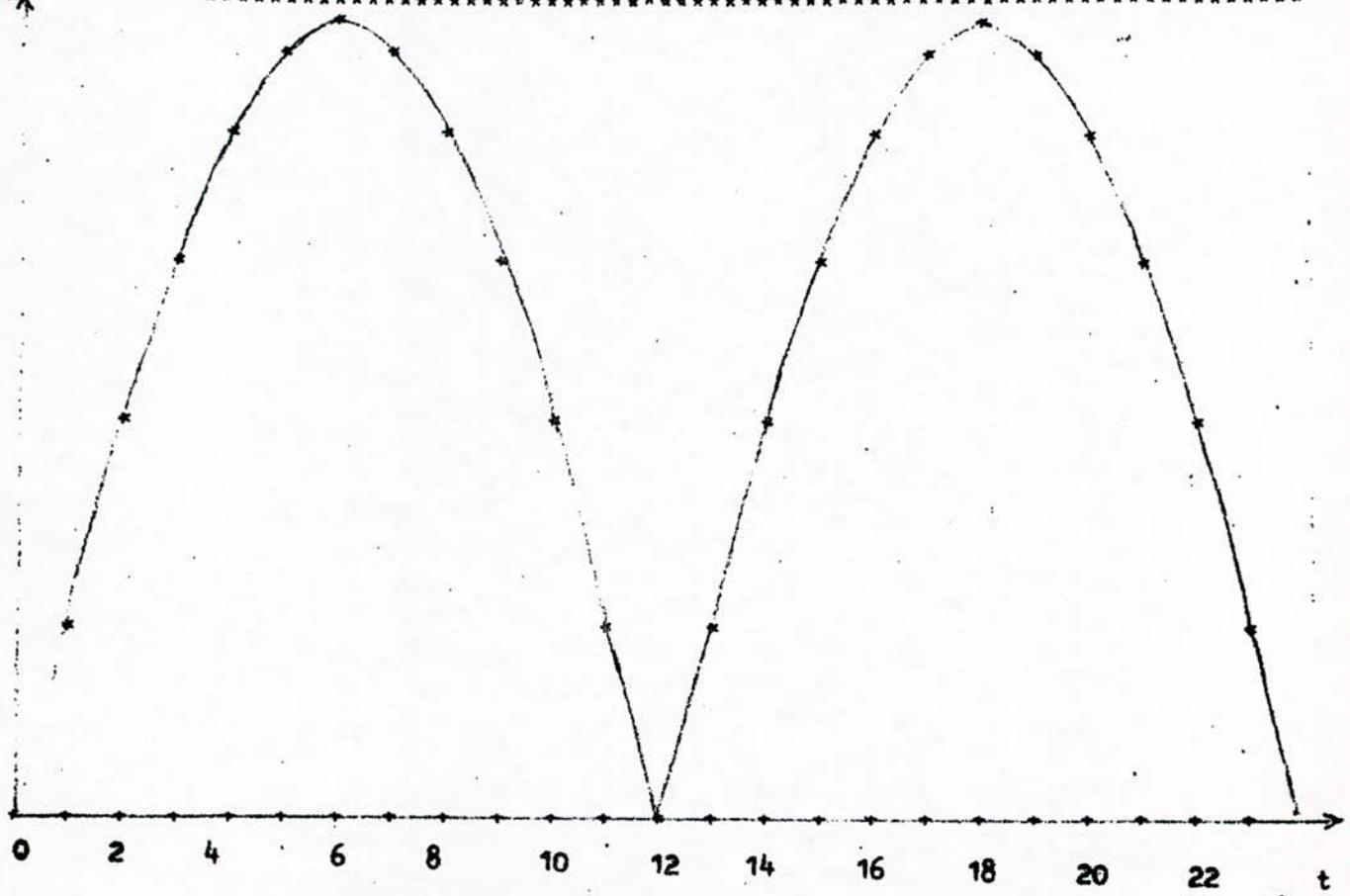
GRAPHE DE LA TRANSFORME DE FOURIER INVERSE

Le signal carré



Sequences de données		Coeff. de la transformé inverse		
N	Re(X(K))	Im(X(K))	Re(X(K))	Im(X(K))
1	.0000E+00	.0000E+00	.0000E+00	-.1192E-06
2	.0000E+00	-.4970E+00	.2500E+00	-.1252E-05
3	.0000E+00	.0000E+00	.5000E+00	-.1421E-05
4	.0000E+00	.1180E-02	.7001E+00	-.3992E-05
5	.0000E+00	.0000E+00	.8601E+00	-.2682E-05
6	.0000E+00	-.3000E-04	.9601E+00	-.6199E-05
7	.0000E+00	.0000E+00	.1000E+01	-.5009E-05
8	.0000E+00	.1710E-02	.9601E+00	-.8225E-05
9	.0000E+00	.0000E+00	.8601E+00	-.2503E-05
10	.0000E+00	.1180E-02	.7001E+00	-.3855E-05
11	.0000E+00	.0000E+00	.5000E+00	-.2911E-05
12	.0000E+00	.1300E-02	.2500E+00	-.1967E-05
13	.0000E+00	.0000E+00	.1819E-11	.1192E-06
14	.0000E+00	-.1300E-02	-.2500E+00	.1609E-05
15	.0000E+00	.0000E+00	-.5000E+00	.2568E-05
16	.0000E+00	-.1180E-02	-.7001E+00	.5786E-05
17	.0000E+00	.0000E+00	-.8601E+00	.2503E-05
18	.0000E+00	-.1710E-02	-.9601E+00	.5126E-05
19	.0000E+00	.0000E+00	-.1000E+01	.5009E-05
20	.0000E+00	.3000E-04	-.9601E+00	.8047E-05
21	.0000E+00	.0000E+00	-.8601E+00	.4688E-05
22	.0000E+00	-.1180E-02	-.7001E+00	.5222E-05
23	.0000E+00	.0000E+00	-.5000E+00	.3641E-05
24	.0000E+00	.4970E+00	-.2500E+00	.1967E-05

 *GRAPHE DE LA TRANSFORME DE FOURIER INVERSE DE X(T)



Le signal Sinus

C H A P I T R E IV

LA CARTE C.P.U.

Cette carte est constituée essentiellement du Micro processeur MC 6809, associé à une unité de calcul rapide AM 9511. Cette unité AM 9511 assure les différentes fonctions trigonométriques, et autres, qui manque dans le micro processeur.

A) Le microprocesseur MC 6809

I Caractéristiques générales :

- Aptitude à réaliser un certain nombre d'opération logique et arithmétique, sous controle de sequence d'instruction.
- Utilisation des mémoires où sont rangés, numérotés par des adresses, les instructions à exécuter, et les données à traiter.
- Aptitude à communiquer-en plus des mémoires - avec l'exterieur

II- Structure de base d'un microprocesseur.

C'est un processeur à 8 bites de haut gamme dont l'organisation interne est orienté vers 16 bits.

Il peut adresse 64 K octets par l'intermédiaire de ces 16 bits du bus d adresses. Comme indiqué sur la figure :1

*1 Description externe :

Le MC 6809 fabriqué en technologie MOS Canal N mono-tension se présente sous forme d'un boîtier DIL 40 broches regroupées en 3 catégories selon leurs fonctions :

Bus d'adresse :

A₀-A₁₅ : Les 16 broches unidirectionnelles permettant le transfert des adresses du micro processeur vers le bus d'adresse du système .

Bus de donnée :

D₀-D₇ : Le microprocesseur 6809 communique avec le Bus des données bidirectionnelles du système.

Bus de controle

R/ \bar{W} : Lecture / écriture:

R/ \bar{W} =1 Lecture : D₀ D₇ sont des entrées

R/ \bar{W} =0 écriture :D₀ D₇ sont des sorties

BA et BS: lignes d'états:

BA: Bus available

BS: Bus state.

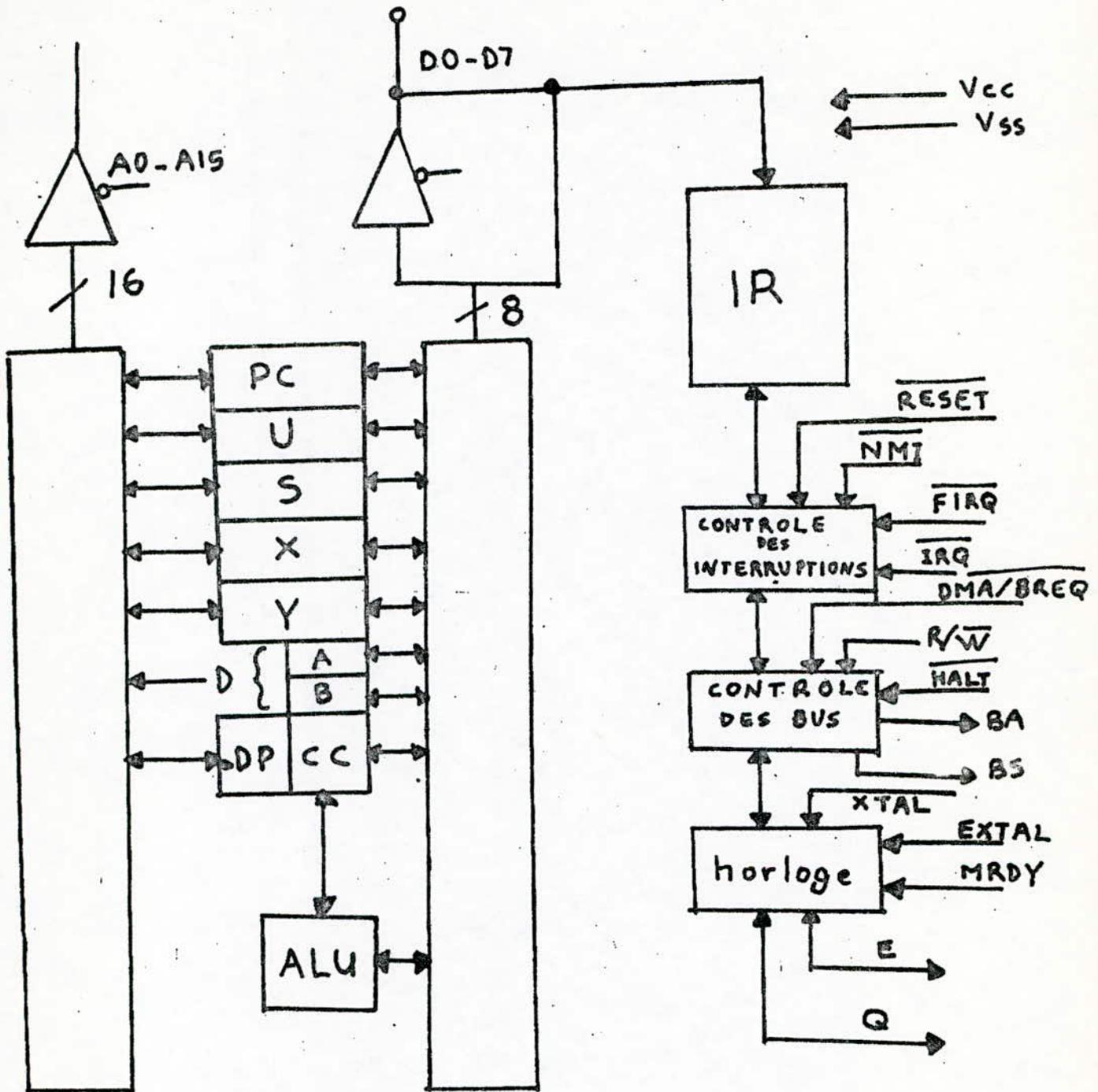


fig:1 Schema fonctionnel du 6809

Le chronogramme du signal R/W du microprocesseur .

FIGURE 28 READ DATA FROM MEMORY OR PERIPHERALS

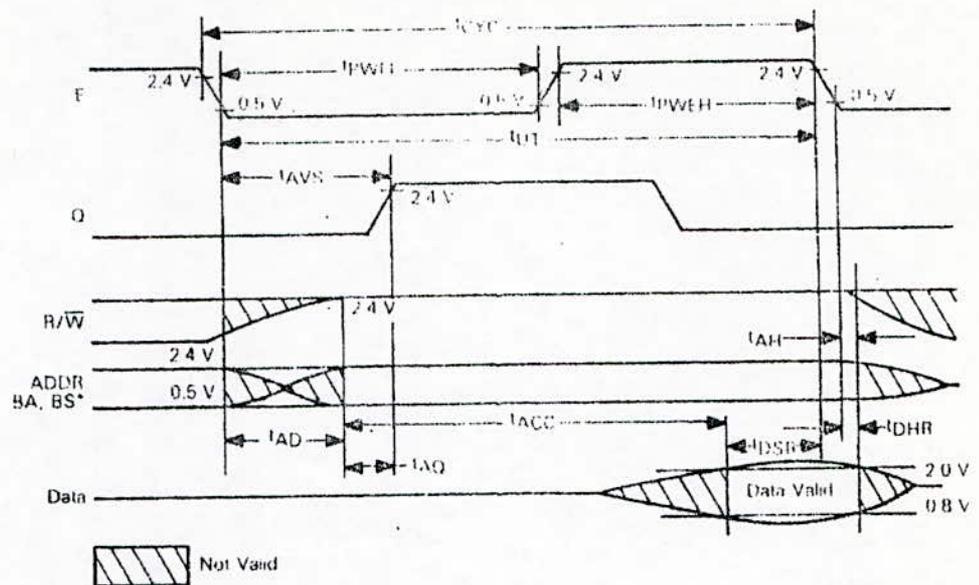
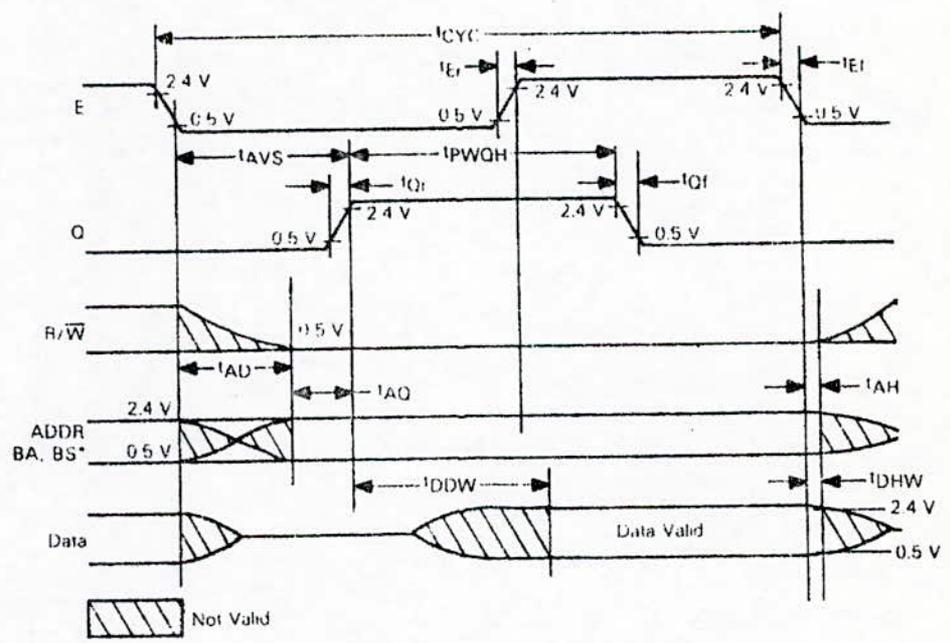


FIGURE 29 WRITE DATA TO MEMORY OR PERIPHERALS



*Hold time for BA, BS not specified

BA	BS	FONCTIONNEMENT DU MICROPROCESSEUR
●	●	Fonctionnement normal, gère le bus d'adresse et de données
●	1	Reconnait des interruptions; reset, NMI, IRQ, SWI1, SWI2, SWI3
1	0	Reconnait une synchronisation externe, les bus sont à l'état haute impédance.
1	1	Indique que le bus est libéré, ou le microprocesseur reconnut l'interruption halt.

- **HALT** : Arrêt du microprocesseur .

Cette entrée permet d'intérrrompre le déroulement du programme de façon hardware. A cet effet :

- Les demandes d'intérruptions **IRQ** et **FIRQ** sont inhibées
- Les demandes **DMA** sont autorisées.
- Les demandes d'intérruptions prioritaire **RESET**, **NMI** sont prises en compte.
- Les horloges fonctionnent normalement.

- **RESET** : Initialisation :

Un niveau bas sur cette entrée entraine une réinitialisation complète du microprocesseur.

- **FIRQ, IRQ** : Interruptions:

Elles sont masquées où validés par logiciel activés sur des niveau bas.

- **NMI** : Interruption :

Elle ne peut pas être masqué par logiciel, activés sur des niveau bas.

- **XTAL** et **EXTAL** : Connection du quartz de 4,096 MHz;

Le quartz assurant ainsi un fonctionnement de bus de 1 MHz lorsque l'on souhaite utiliser une horloge externe, celle ci peut être reliée à **EXTAL** (**XTAL** étant mis à la masse)

- **E.Q** : Sortie d'horloge : (E; Identique à Q2 du 6800)

Q est une horloge en quadrature sur E. Ce signal est un élément fondamental du système puisque les données sont disponible sur le bus lorsque E est à l'état haut.

MRDY : Memory-ready.

L'entrée **MRDY** permet la connection des mémoires où de périphérique lente.

Lorsque **MRDY** passe au niveau bas, l'état haut de E est allongé aussi longtemps que **MRDY** reste à zéro/

-- DMA BREQ : Accès direct mémoire.

Cette ligne demande le bus permettant l'accès direct mémoire.

** Alimentation :

Vcc=+5volts + 5% : tention unique

Vss= 0 masse

2) Description interne.

Le MC 6809 possède :

-- Deux accumulateurs A et B de 8 bits qui peuvent se concatener pour former un seul accumulateur D à 16 bits.

-- Deux registres d'index X et Y à 16 bits.

-- Deux registre pointeurs de pile à 16 bits, utilisable en index U et S.

-- Un pointeur de page DP à 8 bits, permettent de diviser l'espace mémoire du MPU en 256 pages, de 256 octets chacune.

-- Un registre d'état CC de 8 bits.

-- Un pointeur de pile PC à 16 bits.

Il comporte aussi :

-- Une unité arithmétique et logique : UAL

-- Une unité de controle

-- Des Bus internes et buffer entrée/Sorties.

1) Les registres internes :

-- Accumulateurs : A;B,D

Ceux sont des registres où se rangent les données intermédiaire en cours de traitement. Ils doivent communiquer avec UAL et les mémoires de données.

La concatenation de A et B résulte l'accumulateur D à 16 bits.

-- Les registres pointeurs :

* registre d'index X et Y .

Ils sont utilisés dans les modes d'adressages indexés.

* Registre pointeur de piles: U et S

Le registre U (utilisateur) : est utilisé uniquement par le programmeur, pour réaliser des passages d'arguments vers des sous-programmes
Le registre S (système) est utilisé pour la sauve garde des registres interne du microprocesseur pendant des interruptions et aux sauts à des sous programmes.

Les deux registres U et S peuvent être utilisés comme des registres index. c'est la polyvalance du MC 6809.

: - Registre compteur, programme: PC (Compteur Ordinal)

C'est un registre qui se pointe successivement sur les différentes instructions à exécuter. Ces instructions sont rangées en mémoire dont leur ensemble constitue le programme.

Le PC est sur 16 Bits afin de pouvoir atteindre n'importe quelle adresse.

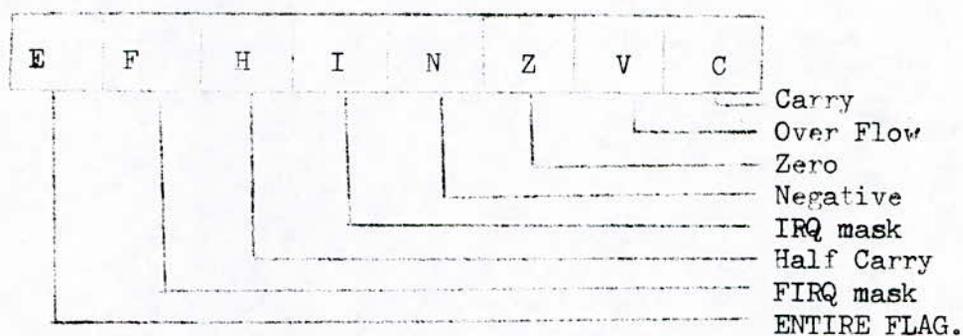
- Registre de page D.P.

Ce registre est prévu pour étendre les possibilités d'adressages direct à tout l'espace mémoire sous contrôle du logiciel. Son contenu représente les poids forts au cours de l'exécution d'une instruction en mode direct.

Il ne peut pas être utilisé pour faire des calculs.

- Registre d'état (C.C.R.) Condition Code Register

Ce registre d'état possède 8 Bits, chacune ayant une signification particulière. Il définit à tout instant l'état des indicateurs de processeur.



E: Indique une interruption, tous les registres sont sauvegardés dans la pile quand $E = 1$

F: Masque d'interruption FIRQ. Pour $F=1$, seules les interruptions prioritaires seront prise en compte.

H: Demi-retenue: c'est la particularité du 6809, qui lui permet d'effectuer directement les opérations en BCB.

I: Lorsqu'il est à 1, masque les interruptions IRQ, sauf celle qui sont prioritaires.

N: Le résultat de l'opération est négative: ($N = 1$)

Z: Le résultat de l'opération est nul ($Z = 1$)

V: Débordement de capacité lors de l'opération à exécuter ($V=1$)

C: La retenue classique

2- L'unité arithmétique est logique:

C'est l'ensemble des circuits combinatoires capable d'effectuer les opérations arithmétiques et logique nécessaire.

3- L'unité de contrôle:

C'est l'ensemble des circuits logiques de contrôle, qui décodent les instructions, les font exécuter par les organes exécutifs (tel que UAL, accumulateur, etc...) au rythme des impulsions d'horloge.

L'unité de contrôle commande les accumulateurs, le compteur de programme, les portes d'aiguillage et gère les échanges avec périphériques.

4- Bus internes et "BUFFER" Entrée/Sortie

- Les Bus internes font communiquer les différents éléments du microprocesseur.

- Les buffers d'entrée/sortie connectés sur les bus externe, servent de tampon entre le micro et l'extérieur.

III: Organisation software du MC 6809

Le microprocesseur 6809 dispose du jeu de modes d'adressage le plus complet. Avec ses 59 instructions de base et ses 10 modes d'adressages, cela correspond à 1464 variantes d'instructions.

1- Différents modes d'adressage.

a) Adressage immédiat:

Ce mode permet de placer une valeur dans un registre EX: LDA #83:
(Charger l'acc.A avec la valeur 83)

b) Adressage étendu

Ce mode permet de retrouver les données en mémoire. Les deux octets qui suivent le code opération spécifient l'adresse de l'opérande.

EX: ~~KA~~ LDA \$ 3104; (Charger l'acc.A/3104 avec le contenu de l'adresse

c) Adressage implicite: (inherent)

Il est utilisé par les registres internes du MPU et non pas sur les mémoires.

* Inherent simple : CLK A

* Inherent porométré: TFR U , S

Le mode inherent porométré un octet supplémentaire afin de préciser les opérandes intervenant dans l'instruction.

d) Adressage direct.

Ce mode est propre au 6809, présente l'avantage de nécessiter que deux octets pour avoir accès à ses données. L'octet de poids forts est contenu par le registre de page

EX: LDA \$ 49, si (DPR) = 10 (Chargement de l'acc. A, avec le contenu de l'adresse 1049.

e) Adressage Indirect étendu

Le champ adresse qui suit l'instruction, contient l'adresse dans laquelle on trouvera l'adresse de l'opérande

EX: ~~KA~~ LDA \$ (\$ 3054), si (\$ 3004) = A845

f) Adressage de registre.

Ce mode un registre ou une paire de registre qui doivent être utilisés par l'instruction.

EX. EXG A, B (échange du contenu des Acc. A \rightarrow B)

g) Adressage relatif.

Ce mode d'adressage ne figure aucune notion d'adresse absolu, puisque tout est référencié par rapport au PC au moyen de déplacement. On dispose de 2 adressages relatifs.

* Relatif court: (sur un octet: - 128 à 127) EX. BEQ

* Relatif Long : (Sur deux octets: - 32768 à 32767) EX. L BEQ

h) Adressage du compteur de programme relatif

Le PC peut-être utilisé comme un registre pointeur avec déplacement signé sur 8 ou 16 bits.

L'adresse effective est obtenue en additionnant le contenu du PC et le déplacement.

i) Adressages indexés.

Ce mode d'adressage permet de calculs, l'adresse effective en ajoutant une valeur appelée déplacement à un registre index, on dispose de plusieurs type d'indexé.

* Indexage avec déplacement nul

* Indexage avec déplacement constant

* Indexage avec déplacement donné par un ACC A,B,D.

* Indexage avec Incrémentation becrementation automatique

* Indexage indirect

2) Jeu d'instruction:

Voir tableau indiqué en fig :3a et 3b

3. Conclusion

La puissance du microprocesseur MC 6809 se résume en 3 points essentiels:

- Nombre et robe des registres internes
- Nombre et possibilité des instructions exentable
- Nombre et possibilité des modes d'adressages

FIGURE 3b - PROGRAMMING AID (CONTINUED)

Instruction	Forms	Addressing Modes												Description	5 H	3 N	2 Z	1 V	0 C				
		Immediate			Direct			Indexed ¹			Extended									Inherent			
		Op	-	#	Op	-	#	Op	-	#	Op	-	#							Op	-	#	
LSL	LSLA													4E	2	1		•	•	•	•	•	
	LSLB				0E	6	2	6E	6	2	7E	7	2	4E	2	1		•	•	•	•	•	
	LSI																	•	•	•	•	•	
LSR	LSRA													44	2	1		•	•	•	•	•	
	LSRB				04	6	2	64	6	2	74	7	2	44	2	1		•	•	•	•	•	
	LSR																	•	•	•	•	•	
MUL														90	1	1	A ← B × C (16-bit)	•	•	•	•	•	
NEG	NECA													4E	2	1		•	•	•	•	•	
	NEGB				0C	6	2	6C	6	2	7C	7	2	4E	2	1		•	•	•	•	•	
	NEG																	•	•	•	•	•	
NOP														12	2	1	No Operation	•	•	•	•	•	
ORA	ORA	5A	2	2	5A	4	2	AA	4	2	6A	5	3				A ← A ∨ B	•	•	•	•	•	
	ORB	CA	2	2	DA	4	2	EA	4	2	FA	5	3					•	•	•	•	•	
	ORCC	1A	3	2														•	•	•	•	•	
PSH	PSHS	34	5	4	2												Push Register 5 Stack	•	•	•	•	•	
	PSHU	36	5	4	2												Push Register 4 Stack	•	•	•	•	•	
PUL	PULS	35	5	4	2												Pop Register 5 Stack	•	•	•	•	•	
	PULU	37	5	4	2												Pop Register 4 Stack	•	•	•	•	•	
ROL	ROLA													40	1	1		•	•	•	•	•	
	ROLB				08	6	2	68	6	2	78	7	2	40	1	1		•	•	•	•	•	
	ROL																	•	•	•	•	•	
ROR	RORA													46	1	1		•	•	•	•	•	
	RORB				0E	6	2	6E	6	2	7E	7	2	46	1	1		•	•	•	•	•	
	ROR																	•	•	•	•	•	
RTI														38	1	1	Return from interrupt	•	•	•	•	•	
RTS														3A	1	1	Return from interrupt	•	•	•	•	•	
SBL	SBLA	82	2	2	82	4	2	A2	4	2	B2	5	3				A ← A ∨ B	•	•	•	•	•	
	SBLB	C2	2	2	D2	4	2	E2	4	2	F2	5	3					•	•	•	•	•	
ST	STA				92	4	2	A2	4	2	B2	5	3				Store to Memory	•	•	•	•	•	
	STR				D2	4	2	E2	4	2	F2	5	3					•	•	•	•	•	
	STB				02	5	3	02	5	3	12	6	4					•	•	•	•	•	
	STS				12	6	4	12	6	4	22	7	5					•	•	•	•	•	
	STU				04	5	3	04	5	3	14	6	4					•	•	•	•	•	
	STA				94	5	3	A4	5	3	B4	6	4					•	•	•	•	•	
	STY				14	6	4	14	6	4	24	7	5					•	•	•	•	•	
	STZ				9E	6	4	AE	6	4	BE	7	5					•	•	•	•	•	
SUB	SUBA	8C	2	2	9C	4	2	AC	4	2	BC	5	3				A ← A - B	•	•	•	•	•	
	SUBB	CC	2	2	DC	4	2	EC	4	2	FC	5	3					•	•	•	•	•	
	SUBD	BC	4	3	93	6	2	A3	6	2	B3	7	3					•	•	•	•	•	
SWI	SWI ^b													8	1	1	Software Interrupt	•	•	•	•	•	
	SWI ^c													9	1	1		•	•	•	•	•	
	SWI ^d													10	1	1		•	•	•	•	•	
	SWI ^e													11	1	1		•	•	•	•	•	
SWNC													12	1	1	Software Interrupt	•	•	•	•	•		
TRP	R1 R2													1F	1	1	TRP - R2	•	•	•	•	•	
TST	TSTA													4D	1	1	Test A	•	•	•	•	•	
	TSTB													5D	1	1		Test B	•	•	•	•	•
	TST				0D	6	2	6D	6	2	7D	7	3					Test M	•	•	•	•	•

- Notes
1. Flag count gives a base cycle and byte count. To obtain total count, add the values obtained from the INDEXED ADDRESSING MODE table.
 2. R1 and R2 may be any pair of 8 bit or any pair of 16 bit registers.
The 8 bit registers are: A, B, CC, DP.
The 16 bit registers are: X, Y, U, S, D, PC.
 3. LA is the effective address.
 4. The PSH and PUL instructions require 6 cycles plus 1 cycle for each byte of data pushed.
 5. 5-6i means 5 cycles if branch not taken, 6 cycles if taken (Branch instruction only).
 6. SWI sets I and F bits. SWI2 and SWI3 do not affect I and F.
 7. Conditions Codes set as a direct result of the instruction.
 8. Value of half carry flag is undefined.
 9. Special Case: Carry set if 17 = 5FF.

B L uniteé de calcul rapide AM 9511

I Généralité:

Le developpement des systemes micro ordinateur de hautes performances a jusqu a present été entravé par deux facteurs:

Une insuffisance très nette dans le domaine du traitement arithmétique.

Un manque d efficacité au niveau du systeme (Aptitude insuffisante à resoudre de façon permanente des probleme donneés.)

ces circuits spécialisés pour les microprocesseurs à huit bits permettent de combler ses lacunes .

II Structure de l'uniteé AM 9511 .

L'uniteé AM 9511 est un mono chip LSI en technologie MOS Canal N à grille sillicium, qui se presente dans un boitier de 24 broches . Il est proposé en gamme civile ou militaire avec une frequence maximale d'horloge de 3 MHZ et demande deux tensions d'alimentation : +5 volts et +12 volts

Voir schema de la figure:4

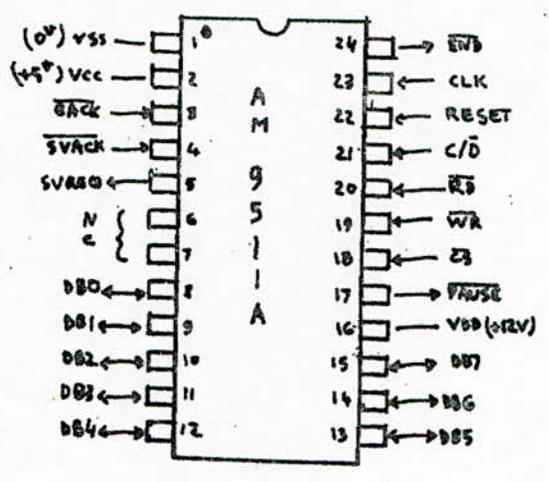
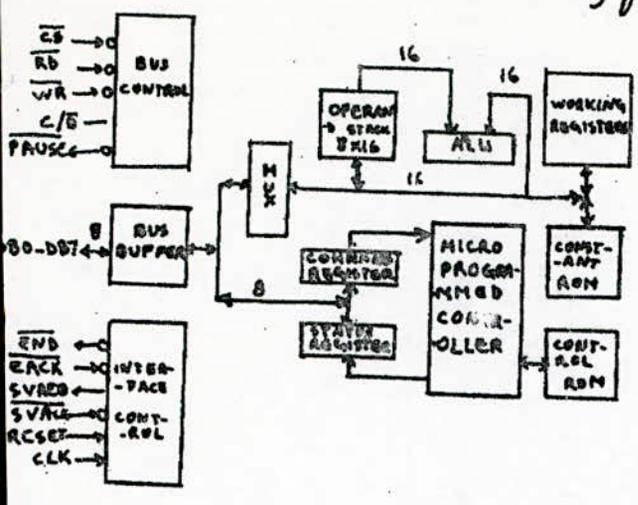
1)Description externe .

En dehors du bus de donne D0 D7 et les broches d'alimentations l uniteé AM 9511 comporte les signaux suivants :

Les signaux d'entrees

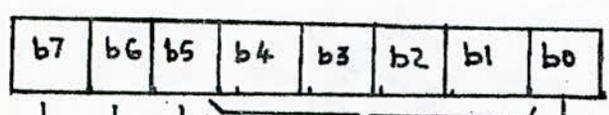
Les signaux de sorties.

fig: 4



BLOCK DIAGRAM

CONNECTION DE L'AM



Retenue

- 0000 - Pas d'erreur
- 1000 - Division par zéro
- 0100 - racine carree ou log d'un nbre < 0
- 1100 - Arg de sin⁻¹ ou cos⁻¹ ou exp d'un nombre N trop grand
- XX10 - overflow
- XX01 - underflow

indicateur de zéro

indicateur de signe

BUSY: indique q'une opération est en cours d'exécution

Registre d'état

a- Signaux d'entrées:

* $\overline{\text{Reset}}$, initialisation du composant, l'activation de cette ligne entraîne l'arrêt de toute les opéraudes en cours. Le RAZ du registre d'état, le positionnement de l'unité dans un état d'attente (IOLE STATE).

* $\overline{\text{CS}}$, Selection du boitier

Valide les echanges d'informations.

* $\overline{\text{C}} / \overline{\text{D}}$ commande / DATA

Indique le type de transfert à effectuer

* $\overline{\text{RD}}$, Lecture

* $\overline{\text{WR}}$, Ecriture

Voir chronogramme : fig: 5a et 5b

$\overline{\text{C/D}}$	$\overline{\text{R/D}}$	$\overline{\text{W/R}}$	F O N C T I O N S
0	1	0	Entrée d'une donnée dans la pile
0	0	1	Sortie de la pile d'un octet
1	1	0	Entrée d'un octet de commande
1	0	1	Sortie d'un mot d'état

* $\overline{\text{BACK}}$, End ocknowledge (acquitement de fin)

remet à zéro la sortie.

* $\overline{\text{SVACK}}$, Service acknowledge (acquitement de service)

remet à zéro le $\overline{\text{SVREQ}}$, indique une demande de service.

* $\overline{\text{CLK}}$, Signal d'horloge.

b- Signaux de sortie

* $\overline{\text{END}}$: Fin d'exécution

* $\overline{\text{SVREQ}}$: Service requet (demande de service)

* $\overline{\text{PAUSE}}$: Permet de synchroniser l'AM 9511 avec MP maitre par l'intermédiaire d'une procédure d'attente

2. Description interne:

L'AM 9511 possède la structure d'un processeur 16 Bits parallèle sur les bus interne de 16 Bits sont connectés les blocs suivants:

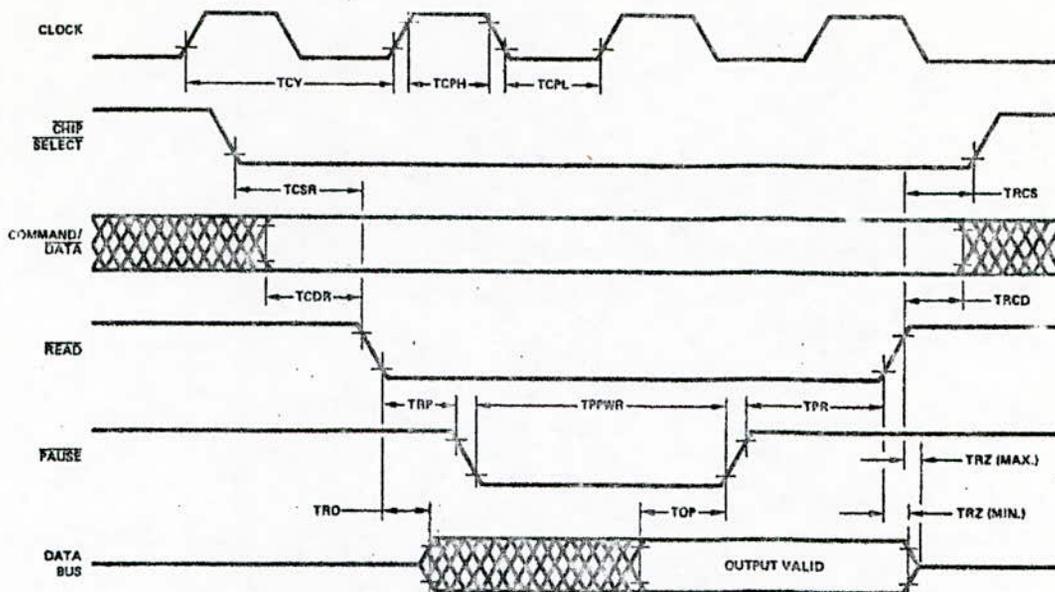
- Une mémoire ROM de 124 constantes
- Les registres de travail au nombre de 10
- Unité arithmétique et logique U.A.L.
- Une pile à 8 niveaux de base où sont stockés les opéraudes et les resultats de calcul.

III- Organisation saftuaire de l'AM 9511

La totalité des transferts s'effectuent par l'intermédiaire d'un bus bidirectionnel de 8 Bits Do- D7.

- Le chronogramme du signal R/W de l'AN 9511
Le signal de lecture .

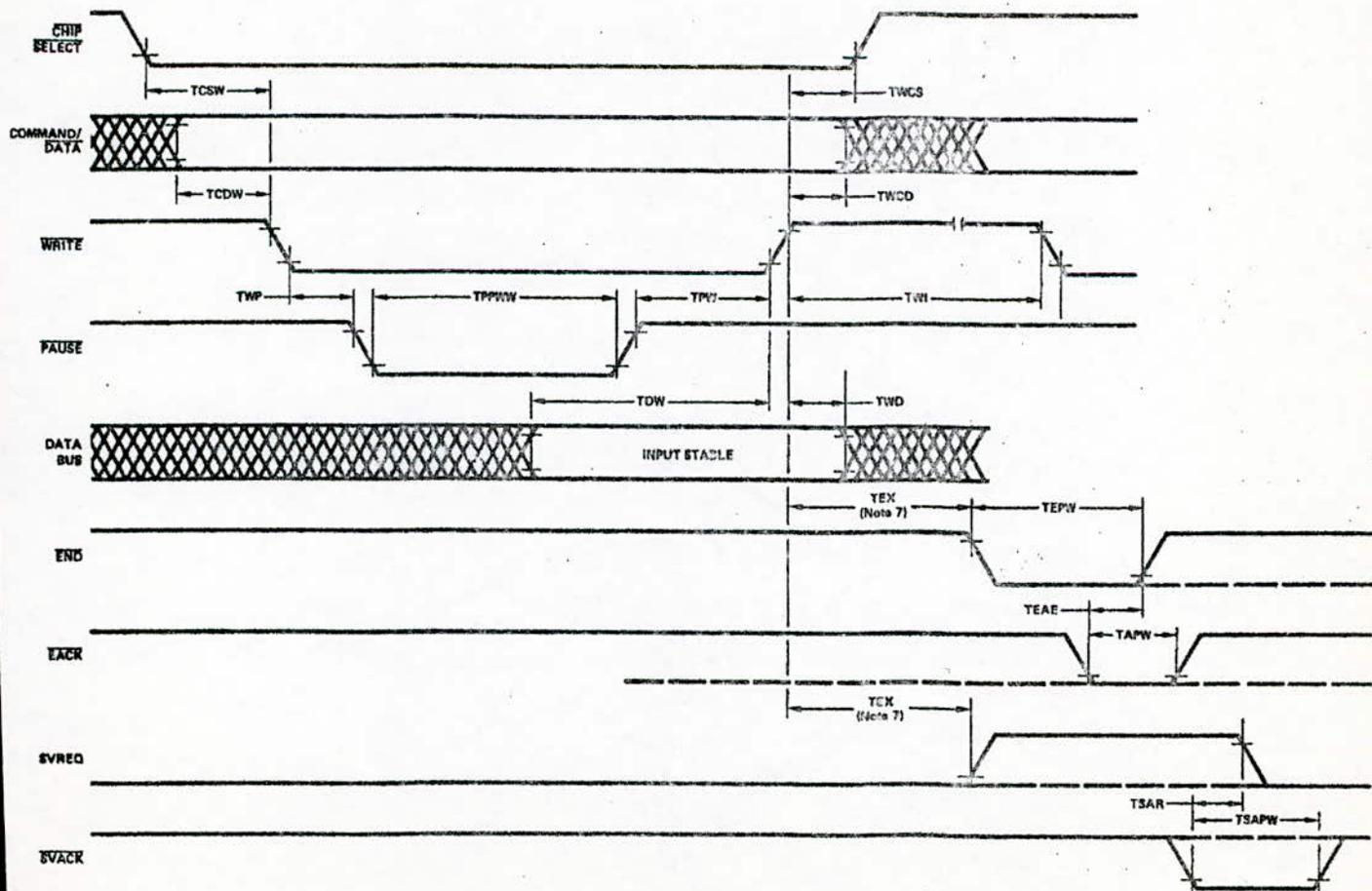
fig:5a READ OPERATIONS



MOS-048

Le signal d'écriture .

fig:5b WRITE OPERATIONS



MOS-049

L'unité AM 9511 comprend un jeu d'instruction propre, destiné à accroître sensiblement la capacité de calcul. Surtout dans le domaine trigonométrique.

Jeu d'instruction de l'AM 9511

Voir Fig: 6

C- Les circuits annexes:

La carte CPU comporte - en plus du M.P. et de l'unité AM 9511- un bus interne permettant la liaison entre les divers circuits:

- Une logique de gestion de bus
- Un ensemble d'amplificateur, permettant de relier le bus interne au bus du fond de panier.

I. Circuits d'interfaces.

Le MP MC 6809 possède des niveaux de sorties, généralement faible. Il faut que tous les signaux des bus (bus de données, d'adresses, ou de contrôle) passent à travers des amplificateurs (Buffers).

Ces buffers sont des circuits logiques inverseurs ou non, inverseurs dont l'entrée est faible et la sortie est relativement élevée.

Ces circuits ne servent pas uniquement à fournir de la puissance ils ont également pour rôle d'isoler le M.P. lorsqu'ils sont en haute impédance. Ceci est nécessaire pour effectuer de l'accès direct mémoire (DMA).

1- Buffers des lignes de contrôles

Les lignes de contrôle, qui passent dans des amplificateurs unidirectionnelles (ST 95), sont validés en permanence non inverseurs

2- Interfaces des lignes d'adresses.

Les 16 lignes d'adresses passent par des amplificateurs unidirectionnel, non inverseurs (ST.95)

Chaque buffers prend six lignes de transmission.

3- Interfaces des lignes de donnée.

Les 8 lignes de données passent par des amplificateurs bidirectionnels, commandés essentiellement par le signal R/\bar{W} qui indique à ces amplificateurs dans quel sens l'information circule:

- Information du M.P. vers les mémoires : écriture
- Information des mémoires vers les M.P.: Lecture

II- Logique de commande et de contrôle.

1- Lecture / Ecriture (R / \bar{W})

Le signal R/\bar{W} amplifiés, détermine le sens de transfert

-a- Opérations d'écritures:

Cette opération a lieu lorsque le signal R/\bar{W} est à l'état bas. Le bus de données du MP doit être validés, dans le sens surtout des buffers MC 8726.

fig:6

Le jeu d'instruction de l'unité AM 9511.

Command Mnemonic	Hex Code (sr = 1)	Hex Code (sr = 0)	Execution Cycles	Summary Description
16-BIT FIXED-POINT OPERATIONS				
SADD	EC	6C	16-18	Add TOS to NOS. Result to NOS. Pop Stack.
SSUB	ED	6D	30-32	Subtract TOS from NOS. Result to NOS. Pop Stack.
SMUL	EE	6E	84-94	Multiply NOS by TOS. Lower result to NOS. Pop Stack.
SMUJ	F6	76	80-98	Multiply NOS by TOS. Upper result to NOS. Pop Stack.
SDIV	EF	6F	84-94	Divide NOS by TOS. Result to NOS. Pop Stack.
32-BIT FIXED-POINT OPERATIONS				
DADD	AC	2C	20-22	Add TOS to NOS. Result to NOS. Pop Stack.
DSUB	AD	2D	38-40	Subtract TOS from NOS. Result to NOS. Pop Stack.
DMUL	AE	2E	194-210	Multiply NOS by TOS. Lower result to NOS. Pop Stack.
DMUJ	B6	36	182-218	Multiply NOS by TOS. Upper result to NOS. Pop Stack.
DDIV	AF	2F	196-210	Divide NOS by TOS. Result to NOS. Pop Stack.
32-BIT FLOATING-POINT PRIMARY OPERATIONS				
FADD	90	10	54-368	Add TOS to NOS. Result to NOS. Pop Stack.
FSUB	91	11	70-370	Subtract TOS from NOS. Result to NOS. Pop Stack.
FMUL	92	12	146-168	Multiply NOS by TOS. Result to NOS. Pop Stack.
FDIV	93	13	154-184	Divide NOS by TOS. Result to NOS. Pop Stack.
32-BIT FLOATING-POINT DERIVED OPERATIONS				
SQRT	81	01	782-870	Square Root of TOS. Result to TOS.
SIN	82	02	3796-4808	Sine of TOS. Result to TOS.
COS	83	03	3840-4878	Cosine of TOS. Result to TOS.
TAN	84	04	4894-5886	Tangent of TOS. Result to TOS.
ASIN	85	05	6230-7938	Inverse Sine of TOS. Result to TOS.
ACOS	86	06	6304-8284	Inverse Cosine of TOS. Result to TOS.
ATAN	87	07	4992-6536	Inverse Tangent of TOS. Result to TOS.
LOG	88	08	4474-7132	Common Logarithm of TOS. Result to TOS.
LN	89	09	4298-6956	Natural Logarithm of TOS. Result to TOS.
EXP	8A	0A	3794-4878	e raised to power in TOS. Result to TOS.
PWR	8B	0B	8290-12032	NOS raised to power in TOS. Result to NOS. Pop Stack.
DATA AND STACK MANIPULATION OPERATIONS				
NOP	80	00	4	No Operation. Clear or set SVREQ.
FIXS	9F	1F	90-214	Convert TOS from floating point format to fixed point format.
FIXD	9E	1E	90-336	
FLTS	9D	1D	62-156	
FLTD	9C	1C	56-342	Convert TOS from fixed point format to floating point format.
CHSS	F4	74	22-24	
CHSD	B4	34	26-28	
CHSF	95	15	16-20	Change sign of floating point operand on TOS.
PTOS	F7	77	16	Push stack. Duplicate NOS in TOS.
PTOD	B7	37	20	
PTOF	97	17	20	
POPS	F8	78	10	Pop stack. Old NOS becomes new TOS. Old TOS rotates to bottom.
POPD	B8	38	12	
POPF	98	18	12	
XCHS	F9	79	18	Exchange TOS and NOS.
XCHD	B9	39	26	
XCHF	99	19	26	
PUPI	9A	1A	16	Push floating point constant π onto TOS. Previous TOS becomes NOS.

La logique de commande est:

$$S_{\text{act}} = R/\bar{W} \cdot \phi 2 \cdot \bar{RG}$$

b- Opération de lecture

Cette opération a lieu lorsque le signal R/\bar{W} est à l'état haut
La validation du bus de données est dans le sens entrant des buffers 8T 26.

La logique de commande lecture est:

$$S_{\text{act}} = R/\bar{W} \cdot \phi \cdot \bar{RG}$$

La table de vérité de la logique de commande de 8T 26.

R/ \bar{W}	$\phi 2$	\bar{RG}	PIN 14 (RECT)	PIN 15 (ECRIT.)
0	0	0	1	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	0	0
1	1	1	1	0

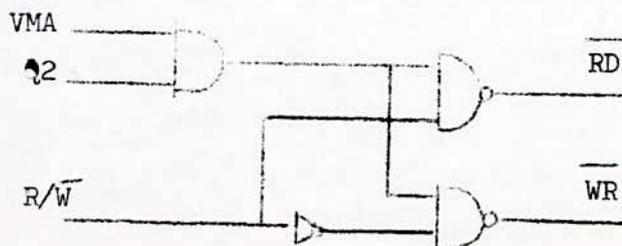
L'activation des buffers d'adresses et du signal R/\bar{W} fait intervenir le signal R.G. (Refresh Grant). Il permet de faire à l'état haute impédances (3 état) les buffers d'adresses et de données. Ce qui protège le NP des courts-circuits accidentels.

C - Les Equations de lecture et écriture de l'AM 9511

$$\bar{WR} = \bar{VMA} \cdot \phi 2 \cdot R/\bar{W}$$

$$\bar{KD} = VMA \cdot \phi 2 \cdot R/\bar{W}$$

VMA	$\phi 2$	R/ \bar{W}	\bar{RD}	\bar{WR}
1	1	0	0	0
1	1	1	0	1
X	0	X	1	1
0	X	X	1	1



2- Le signal V M A

Le signal VM A est généré à partir des deux signaux d'horloge E, Q, et le signal B A.

Sa logique de commande est la suivante:

$$VMA = \overline{E + Q} + BA$$



La table de vérité du signal VMA

EG	Q	BA	VMA
0	0	X	0
0	1	0	1
1	0	0	1
1	1	0	1
X	X	1	0

3- Circuit de réinitialisation

A la mise sous tension, la broche RESET reste à zéro, pendant que la capacité se charge. Dès qu'elle dépasse le seuil de 4 V Le MP commence sa séquence d'initialisation en cas de coupure de courant, la diode permet une décharge rapide de la capacité.

4- Circuit d'horloge.

Le MC 6875 est prévu pour fournir les fréquences 270 pour l'unité rapide AM 9511 et 470 pour le MP 6809. Cette horloge comporte un oscillateur intégré, ainsi qu'un quartz qui déterminent la fréquence de fonctionnement de 4MHZ

5- Circuit de refroidissement

L'élément d'une mémoire est la capacité grille source d'un transistor MOS. Cette capacité perd sa charge. Il faut régulièrement simuler une lecture afin de conserver les informations.

La demande de rafraîchissement est formulé par le signal (Refrec.Requiet). Il est synchronisé avec l'horloge ϕ 2 au moyen d'une bascule D, pour attaquer l'entrée DMA du MC 6809.

III Décodage d'adresses des circuits.

1) Tableau d'adressages.

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	0	0	0	0
0	0	0	0	1
0	0	0	1	0
0	0	0	1	1
0	0	1	0	0
0	0	1	0	1
0	0	1	1	0
0	0	1	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	1	0	1	0	1	1	1	1	1	1	1	0	0	.
1	1	1	0	1	0	1	1	1	1	1	1	1	0	1	.
1	1	1	0	1	0	1	1	1	1	1	1	1	1	.	.

2) Répartition du champ mémoire .

0	0	0	0	
3	F	F	F	RAM de 16 K Octets.
4	0	0	0	
B	F	F	F	ROM de 32 K Octets.
				Non utilisé
E	B	F	8	
E	B	F	9	AM 9511
E	B	F	A	
E	B	F	B	ACIA 6850
E	B	F	C	
E	BB	F	F	PIA 6821
				Non utilisé.
F	0	0	0	
F	F	F	F	RESERVE AU DEBUG.

3) Principe de décodage .

Le décodeur 74138 va nous servir à décoder en premier lieu les boîtiers RAM-ROM; et les éléments de la carte CPU: à l'aide des bits A1 et A2.

Le circuit AM 9511 est vu par le microprocesseur comme deux positions mémoires. UNE fois ce circuit selecté, la ligne A0 relié à C/D (Commande / Data) indique la fonction de transfert :

A0 = 0, Correspond à l'adresse EBF8

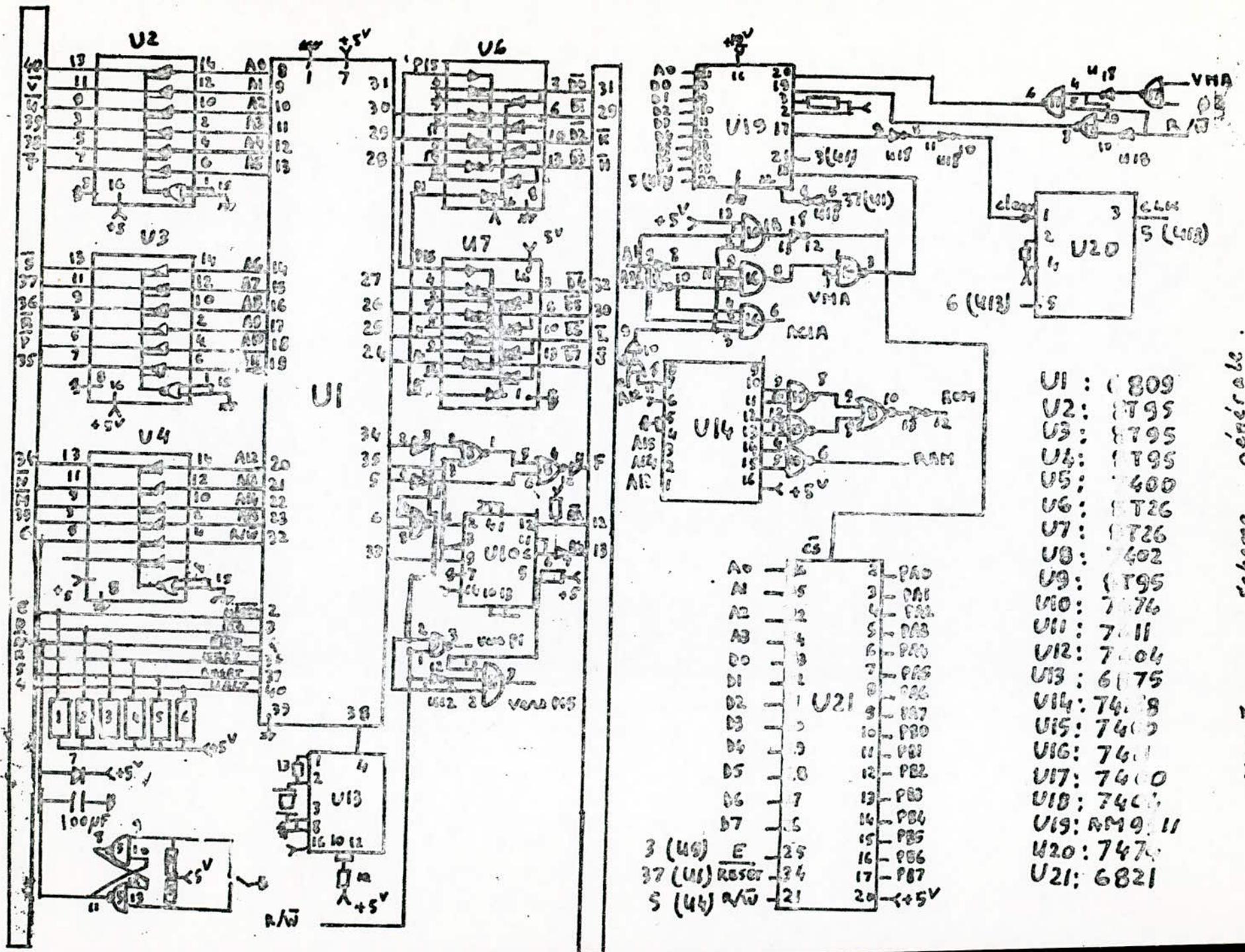
C'est l'introduction des données .

A1 = 1 Correspond à l'adresse EBF9

C'est l'introduction des commandes

IV Connection de la carte .

Voir schema générale.



- U1 : 6809
- U2 : 7495
- U3 : 7495
- U4 : 7495
- U5 : 400
- U6 : 7426
- U7 : 7426
- U8 : 402
- U9 : 7495
- U10 : 7476
- U11 : 7411
- U12 : 7404
- U13 : 6175
- U14 : 7418
- U15 : 7400
- U16 : 7401
- U17 : 7400
- U18 : 7401
- U19 : RM911
- U20 : 7474
- U21 : 6821

A0	1	2	PA0
A1	2	3	PA1
A2	3	4	PA2
A3	4	5	PA3
B0	5	6	PA4
B1	6	7	PA5
B2	7	8	PA6
B3	8	9	PA7
B4	9	10	PA8
B5	10	11	PA9
B6	11	12	PA10
B7	12	13	PA11
	13	14	PA12
	14	15	PA13
	15	16	PA14
	16	17	PA15
	17	18	PA16
	18	19	PA17
3 (U9)	E	25	
37 (U1)	RST	24	
5 (U4)	R/W	21	

fig: 7 Schema générale

CHAPITRE V

ENTREE / SORTIE

I GENERALITE:

L'interfaçage assure l'adaptation entre le microprocesseur et l'exterieur. L'exterieur englobe tous les organes "peripheriques" qui permettent la communication entre le microprocesseur et l'utilisateur, ou entre le microprocesseur et un systeme quelconque suivant l'usage. Voir schema synoptique(fig:1)

Les echanges d'informations necessitent à la fois une partie materielle(HARDWARE); comprenant les circuits d'interface et une partie logicielle (SOFTWARE):comprenant les instructions à fournir au microprocesseur, pour qu'il gere convenablement les echanges.

La partie hardware:

C'est un module à qui le microprocesseur fait appel lors d'une instruction d'entree / sortie. Il est l'intermediaire direct entre ce dernier et l'exterieur.

Ces interfaces comportent une logique de controle qui commande aussi bien le microprocesseur que les peripheriques.

Ils sont caracterisees par leur mode de transmission; on distingue deux types d'interfaces:

ACIA:Asynchro nous Communication Interface Adapter."

Le mode de transmission de l'information est en serie.

PIA:"Peripheral Interface Adapter,""

Le mode de transmission est en parallele.

La partie software:

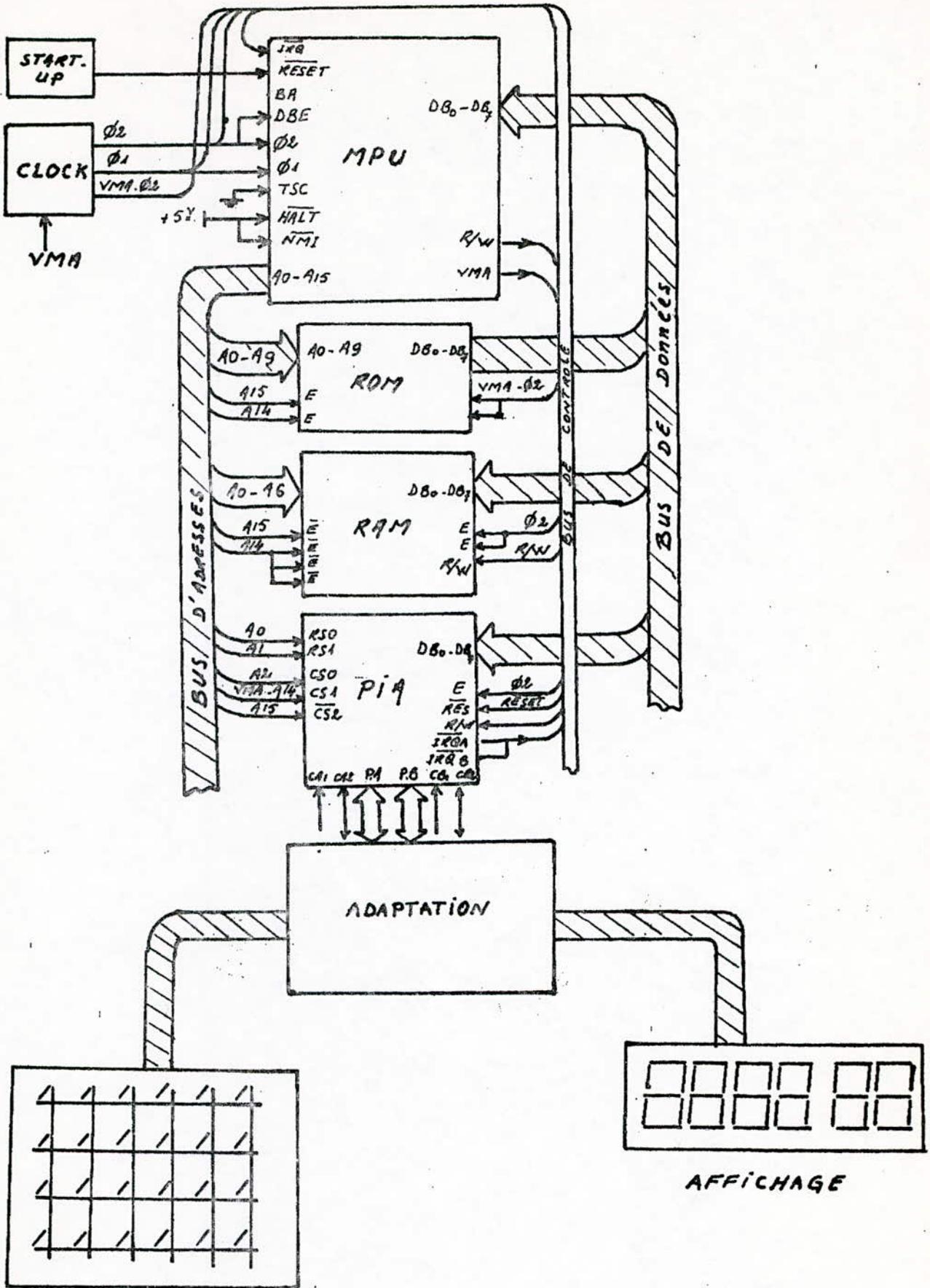
C'est le programme de gestion du peripherique. Pour chaque peripherique, le microprocesseur doit disposer d'un programme specifique. Ce dernier est stocké en memoire, sous forme d'un sous programme auquel se branche le microprocesseur au moment d'une entree / sortie .

II ETUDE DES INTERFACES PROGRAMMABLES

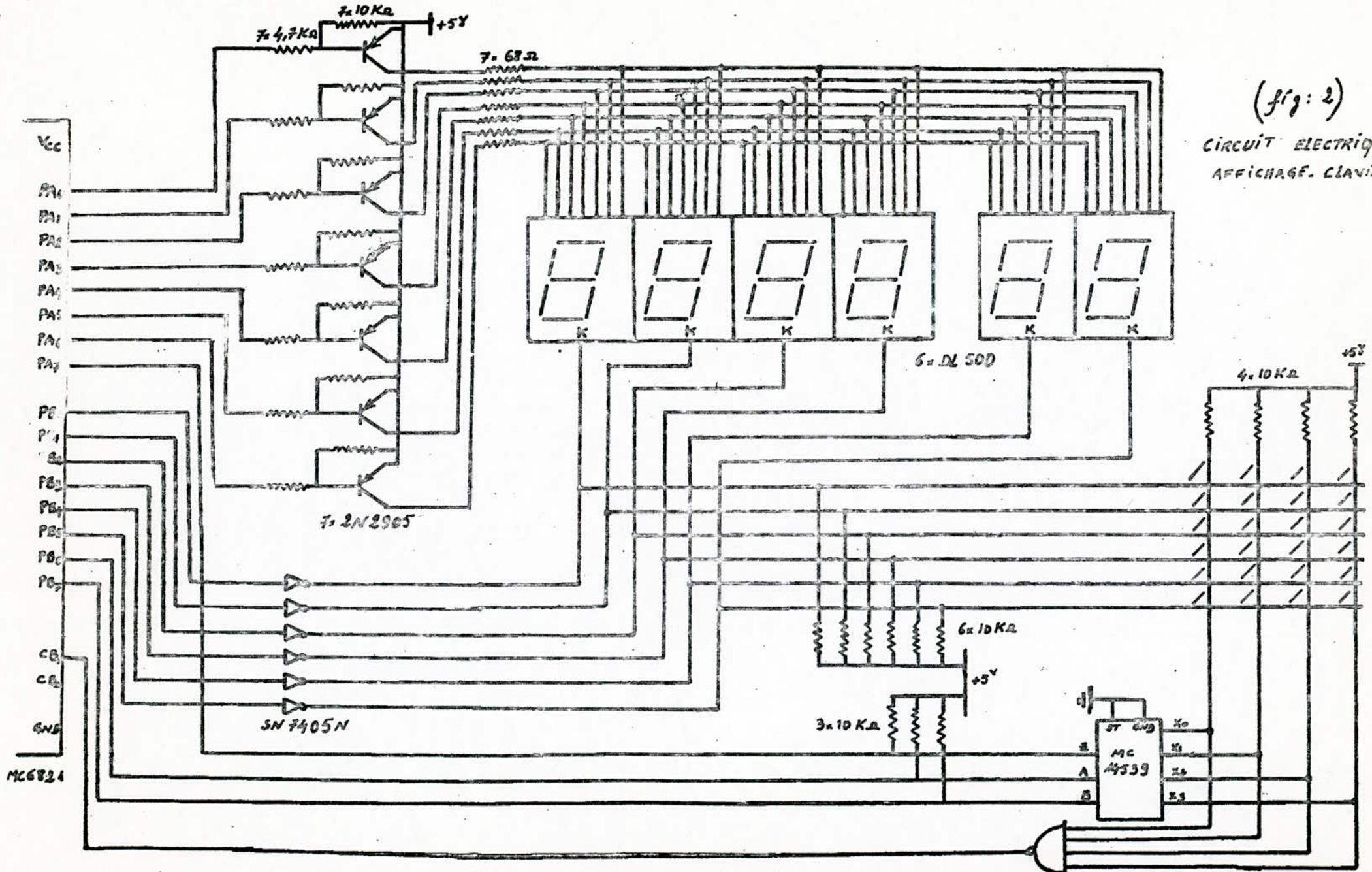
Pour communiquer avec ces peripheriques, le microprocesseur dispose generalement de deux modes de transmissions:(le mode parallele et le mode serie).

La premiere technique correspond à la disposition en parallele d'un certain nombre de conducteur de la carte "mere". La façon de mettre en oeuvre cette technique peut etre materialisee par un PIA; dont la fonction principale sera d'isoler le bus de l'exterieur ; tout en permettant à un certain nombre de signaux d'etre transmis ou reçus.

Ce mode de liaison a pour avantage d'etre rapide .



(fig: 1) schéma synoptique "Entrée/Sortie"



(fig: 2)
CIRCUIT ELECTRIQUE
AFFICHAGE-CLAVIER

1) Description du PIA.

Le PIA "MC6821" est un circuit d'interface parallele, divisé en deux parties, comprenant chacune huit voies d'entree/sortie programmable independamment.

La structure du PIA est donnee par la figure:3a

A) Presentation externe:

Le PIA est caracterise par les differents signaux, qui se classent en deux categories distinctes .

a) Les signaux échangés avec le microprocesseur.

Bus d'adresses:

Le PIA etant selectionne par les chips select "CS0, CS1, CS2" et les quatre combinaisons possible de RS1, RS0, permettant de selectionner les registres internes; il occupe quatre positions memoires vis-à-vis du microprocesseur.

Bus de controle:

Il comprend les lignes suivantes:

*E: Le signal d'activation des echanges.

R/W: Le signal de lecture -écriture

*RESET: Mise à zero. ce signal remet tous les registres interne du PIA à zero.

*IRQA, IRQB: Ils sont relies à IRQ, FIRQ ou NMI du microprocesseur
Ces lignes permettent d'interrompre l'execution d'un programme .

*CS / La selection du bitier PIA; constituee par "CS0, CS1, CS2"

Bus de donnees: "D0 -D7"

Ce sont des bus bidirectionnel, constitués par huit bits permettant les echanges d'informatios.

b) Les signaux échangés avec le peripherique.

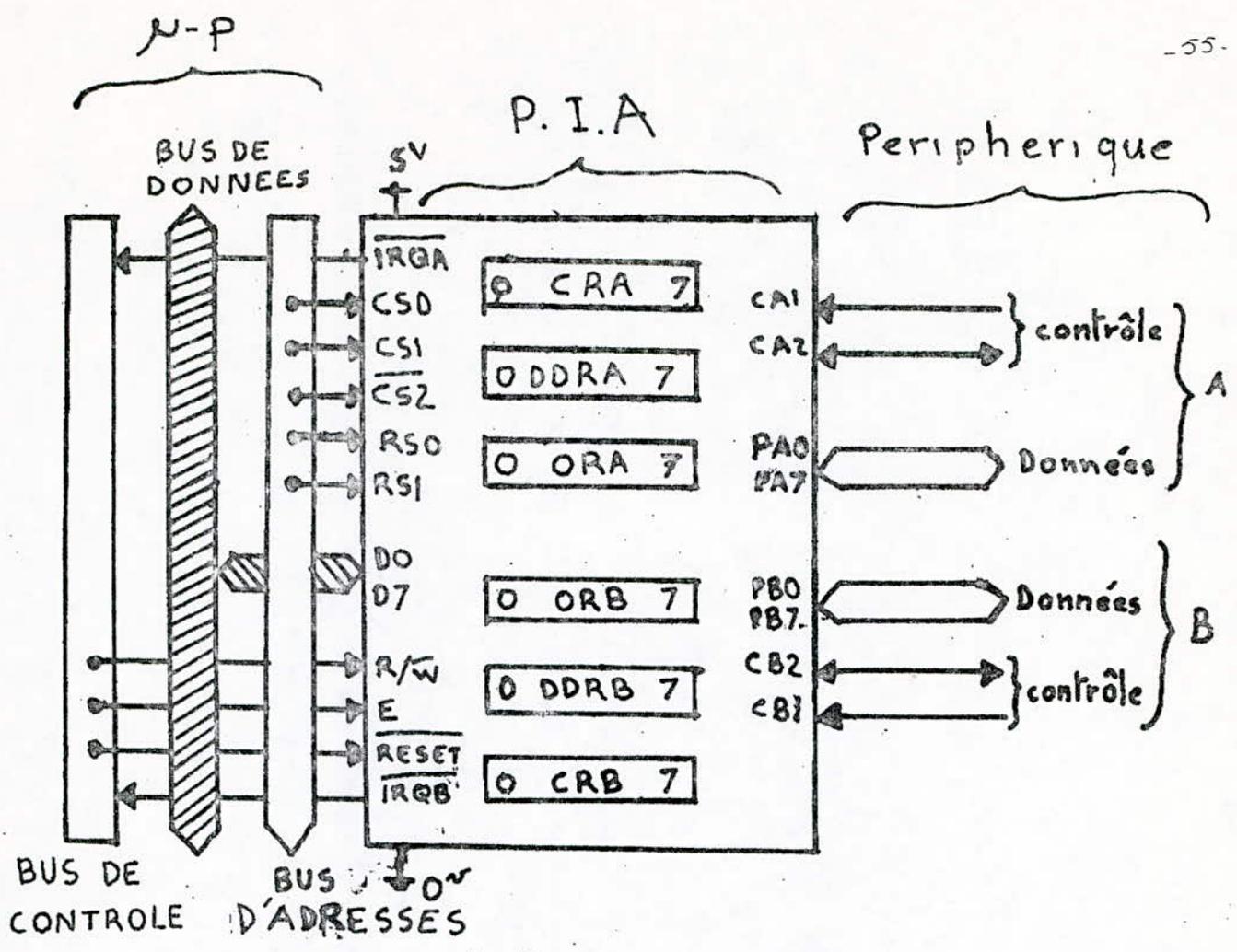
Les lignes de transferts.

"PA0-PA7"; "PB0-PB7" sont programmables individuellement en entree ou en sortie , et sont accessibles par les registres (ORA); (ORB).

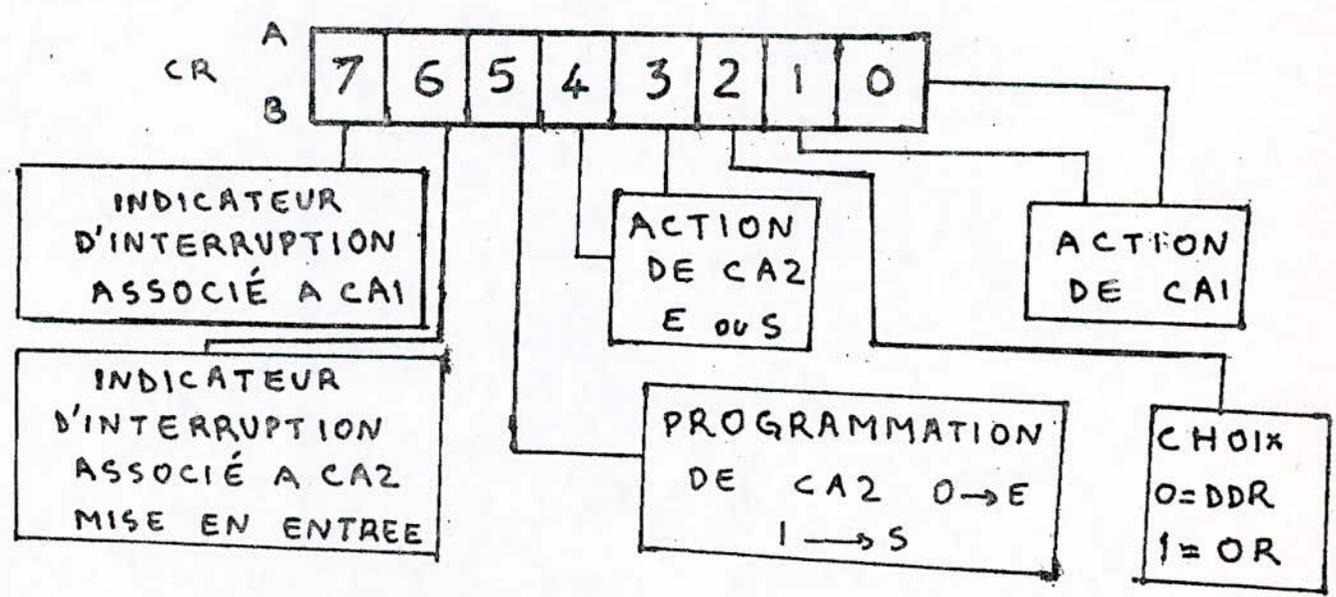
Les lignes de dialogues.

CA1, CBB1: Deux entrees d'interruptions associees respectivement aux ports A et ports B

CA2, CB2: Deux lignes programmables en entree d'interruption ou en sortie de commande



(fig: 3a) organisation interne et externe du P.I.A



(fig: 3b) LES REGISTRES DE COMMANDE

B) Présentation interne.

La structure interne du PIA est constitué de 6 registres:

Registre: "CRA, CRB"

Ces deux registres de 8 bits dont 6 bits sont accessibles en lecture et écriture, 2 bits sont des indicateur d'états en lecture.

Registres: "DRA, DRB"

Ces deux registres définissent le sens de transfert, soit à lecture ou à écriture.

Registres: "ORA, ORB"

Ces deux registres mémorisent une donnée en sortie lors d'une écriture.

Etude d'un affichage 7 segments.

Nous représentons nos résultats sous forme: "Mantisse Exposant"; après une conversion decimal BCD 7 Segments et un multiplexage par logiciel. La commande de ces afficheurs est assurée par un interface programmable le PIA.

Chaque afficheur est constitué de 8 diodes électro lumnéscentes "LED". La structure à 8 bits parallèle de notre microprocesseur MC 6809 permet de générer simultanément une combinaison de 8 éléments sur chaque afficheur. L'organisation de ceux-ci est telle qu'il suffit de connecter un fil du bus de données à chacune des anodes. Les cathodes servent à la validation des digits une à une.

On dresse alors un tableau d'équivalence pour chaque caractère hexadécimal, en tenant compte de l'inverseur de commande:

-Segment allumé: niveau 0

-Segment éteint: niveau 1

Caractere hexadécimal	Equivalence segment							Equivalence sans P.D.
	g	f	e	d	c	b	a	
0	1	0	0	0	0	0	0	40
1	1	1	1	1	0	0	1	79
2	0	1	0	0	1	0	0	24
3	0	1	1	0	0	0	0	30
4	0	0	1	1	0	0	1	19
5	0	0	1	0	0	1	0	12
6	0	0	0	0	0	1	0	02
7	1	1	1	1	0	0	0	78
8	0	0	0	0	0	0	0	00
9	0	0	1	1	0	0	0	18
A	0	0	0	1	0	0	0	08
B	0	0	0	0	0	1	1	03

C	1	●	0	●	●	1	●	46
D	●	1	●	0	0	●	1	21
E	0	0	0	●	1	1	0	●6
F	●	0	0	1	1	1	0	0E
Tiret	●	1	1	1	1	1	1	3F
Effacement	1	1	1	1	1	1	1	7F

1) Adaptation des niveaux.

Les sorties du PIA ne peuvent alimenter qu'une charge TTL classique, or les circuits de la famille MOTOROLA sont réalisés en technologie MOS, nous devons commander les anodes des afficheurs par des transistors montés en commutateurs fonctionnant en bloqués, saturés.

De même, des inverseurs sont utilisés pour attaquer les cathodes.

2) Fonctionnement de l'affichage

Tous les segments homologues des afficheurs sont reliés entre eux. Tandis que les cathodes sont disponibles à la sélection des digits.

Ce qu'on appelle: "Affichage multiplexé".

Dans un premier temps, la cathode de l'afficheur n:1 est sélectionnée par le PIA à travers un inverseur. La donnée à visualiser est présente sur toutes les anodes des différents afficheurs (de a--f).

Seul l'afficheur n:1 s'allume.

Ensuite la cathode n:2 sera sélectionnée, la donnée se trouve sur les différents anodes; Seul l'afficheur n:2 s'allume.

Et ainsi de suite jusqu'au dernier afficheur. Puis un nouveau cycle recommence.

Une temporisation est nécessaire pour que la persistance des impressions rétinienne fait que l'œil a l'impression que tous les afficheurs sont allumés simultanément et non séquentiellement.

IV) Etude d'un clavier.

Pour introduire nos données , ainsi que des commandes. Nous aurons besoin d'un organe d'entrée (Clavier).

Notre clavier comprend 24 touches, dont 16 touches dites de données ou numériques, gradués de 0 à F; et de 8 touches de commande de fonction; que le programme de gestion interprète.

Le codage des touches appuyées sur le clavier nécessite la circuiterie suivante :

- Le clavier en matrice
- Un multiplexeur
- Un inverseur , une porte NAND et des résistances de couplages;
Voir fig:2

1) Rôle des différents circuits.

- Le multiplexeur: MC 14539 Voir annexes

Les quatre colonnes du clavier sont reliées à l'entrée du multiplexeur. La colonne qui se retrouve en sortie -i.e; sur le bit PA7 du PIA , sera, celle sélectionnée; selon l'état du PB6 et PE7. Le multiplexeur joue alors le rôle du décodeur de position des colonnes de la matrice du clavier.

- La porte logique NAND: MC 14012

Les quatre colonnes du clavier sont aussi reliées à la porte NAND, dont la sortie est branchée à CB1 du PIA .
Quand aucune touche n'est enfoncée, l'entrée CB1 est à un niveau bas. Si on appuie une touche quelconque , un front montant déclenche le processus d'interruption, et fait appel au programme de gestion.

- L'inverseur: SN 7405

Les sorties des 6 inverseurs sont reliées aux lignes du clavier par laquelle sont envoyés des niveaux logiques bas .

2) Fonctionnement du clavier.

Le clavier se présente sous forme d'une matrice de 6 lignes et de 4 colonnes. Voir fig:1.

Les traits obliques représentent les touches , lorsqu'une touche est enfoncée, elle met en contact une colonne et une ligne.

Dans un premier temps le microprocesseur envoie à travers le PIA des niveaux logiques bas sur les lignes de (R0 à R5) et examine l'état des colonnes de (C0 à C3).

Si une touche est appuyée la colonne correspondante va passer à l'état bas signalant au CPU deux choses:

Une touche est actionnée: (Demande d'interruption)

Cette touche se trouve dans une telle colonne

ON envoie des niveaux bas successifs sur R0: R1: jusqu'à R5; tout en examinant l'état des colonnes (C0 à C3). Dès qu'on observe un changement sur les colonnes (C0 à C3): il localise ainsi la position de la touche. Cette touche se trouve obligatoirement sur la

ligne et la colonne qui viennent de passer à l'état bas

On obtient le tableau d'équivalence des touches.

Caractere d'une touche	L'état du PORT BB								Equivalence
	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	
0	0	0	0	0	0	0	0	1	01
1	0	0	0	0	0	0	1	0	02
4	0	0	0	0	0	1	0	0	04
7	0	0	0	0	1	0	0	0	08
	0	0	0	1	0	0	0	0	10
M	0	0	1	0	0	0	0	0	20
F	0	1	0	0	0	0	0	0	41
2	0	1	0	0	0	0	1	0	42
5	0	1	0	0	0	1	0	0	44
8	0	1	0	0	1	0	0	0	48
Reset	X	X	X	X	X	X	X	X	XX
Es	0	1	1	0	0	0	0	0	60
E	1	0	0	0	0	0	0	1	81
3	1	0	0	0	0	0	1	0	82
6	1	0	0	0	0	1	0	0	84
9	1	0	0	0	1	0	0	0	88
N	1	0	0	1	0	0	0	0	90
R	1	0	1	0	0	0	0	0	A0
D	1	1	0	0	0	0	0	1	C1
C	1	1	0	0	0	0	1	0	C2
B	1	1	0	0	0	1	0	0	C4
A	1	1	0	0	1	0	0	0	C8
V	1	1	0	1	0	0	0	0	D0
G	1	1	1	0	0	0	0	0	E0

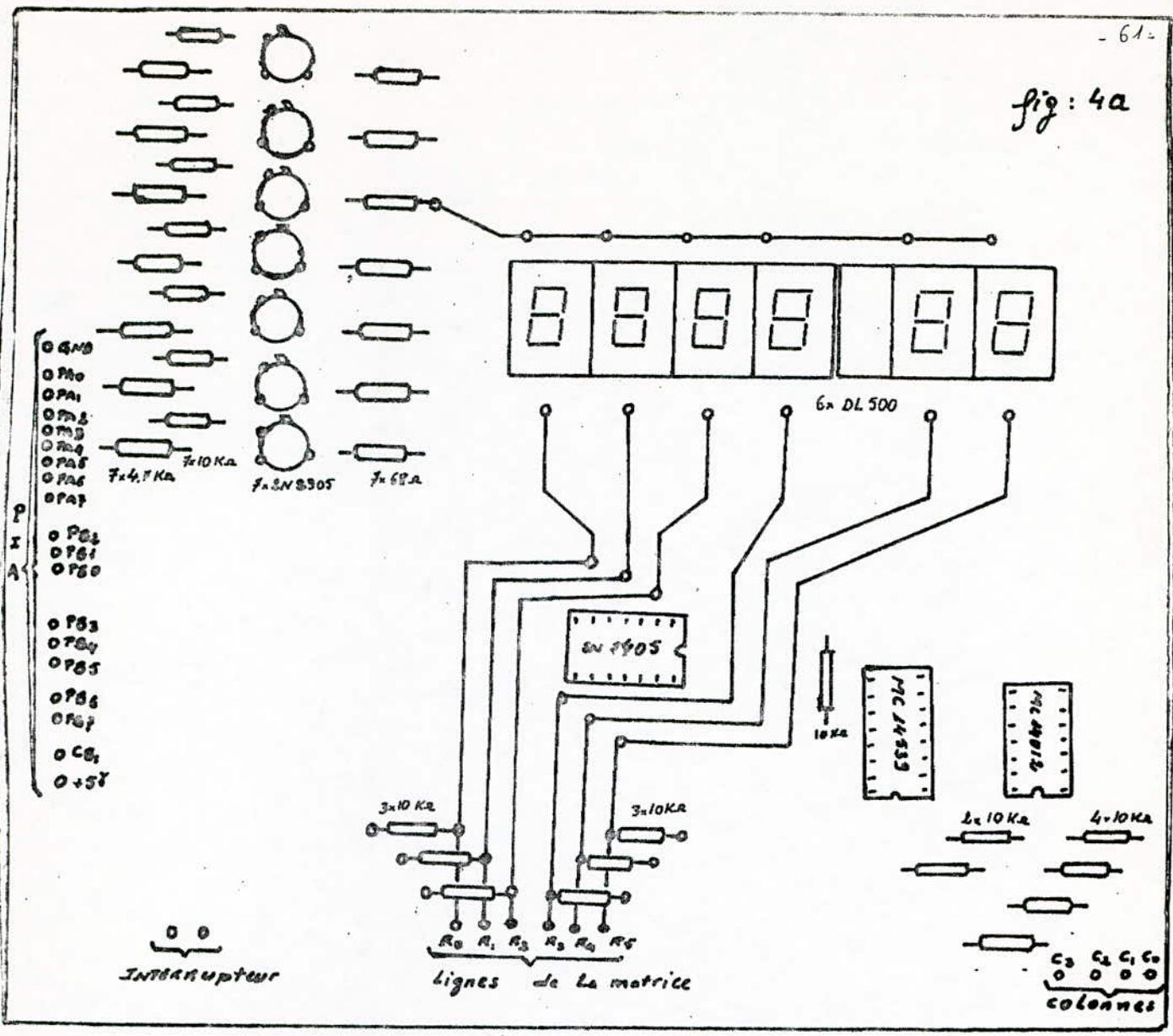
Fonction des touches de commandes.

- *Rese' : Initialisation manuel de tout le systeme
- *M : (Memory display) Permet de visualiser le contenu des memoires.
- *R : (Register display) Permet d'examiner le des registres du CPU
en commençant par le P.C.
- *Es : (Escape) Cette touche permet de revenir sous controle
du moniteur (Debug).
- *G : (Touche GO): C est une commande qui permet de lancer le
programme en executio à l'adresse desireé.
- *V Cette touche autorise la mise en place des points d'arrets.
Le programme s'exécutera jusqu'à ce qu'il arrive à un
tel point et s'arretera.
- *N : Cette touche permet la mise au pont du programme ,en executant
instruction par instruction
- *(-) : C est le signe de la négation.

V REALISATION PRATIQUE / Clavier / Affichage 7 segments)

- 1) Circuit electrique :Voir fig:2
- 2) Schema d implantation des composants :Voir fig:4aet 4b
- 3) Circuit imprimé :Voir fig:5a et 5b

fig: 4a



Reset Interrupteur

lignes de la matrice
 R₀ R₁ R₂ R₃ R₄ R₅

colonne de la matrice
 C₀ C₁ C₂ C₃ C₄ C₅

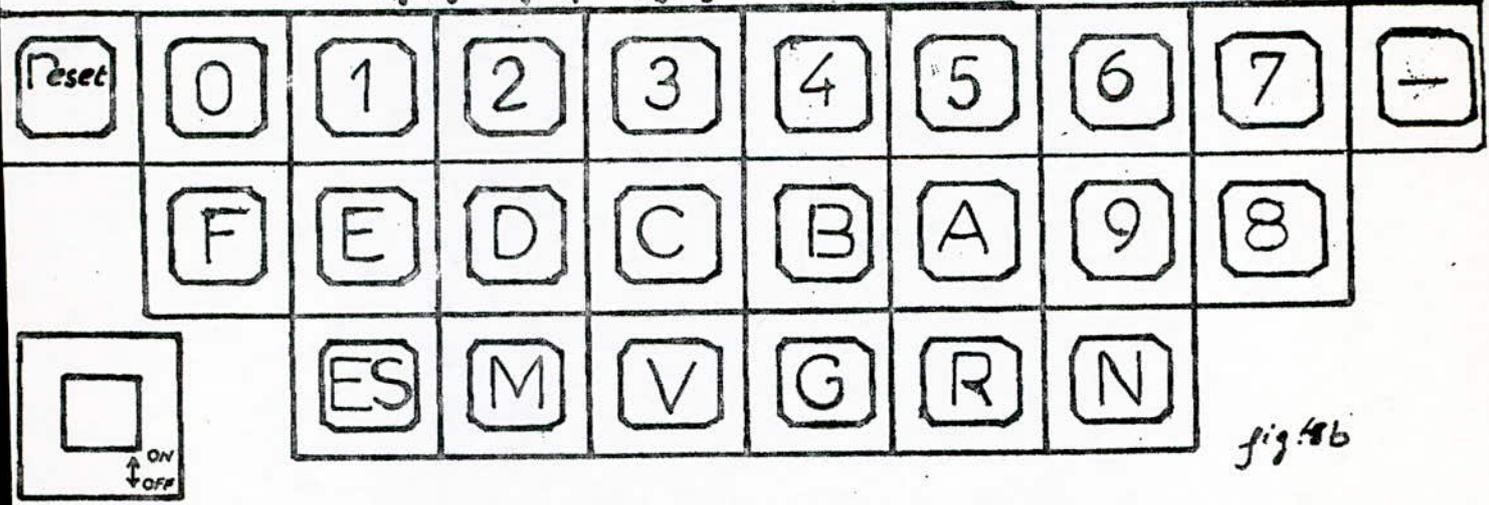


fig: 4b

Schema d'implantation des composants

Circuit Imprimé double face de l'affichage
fig: 5a ; côté cuivre - Echelle : 1

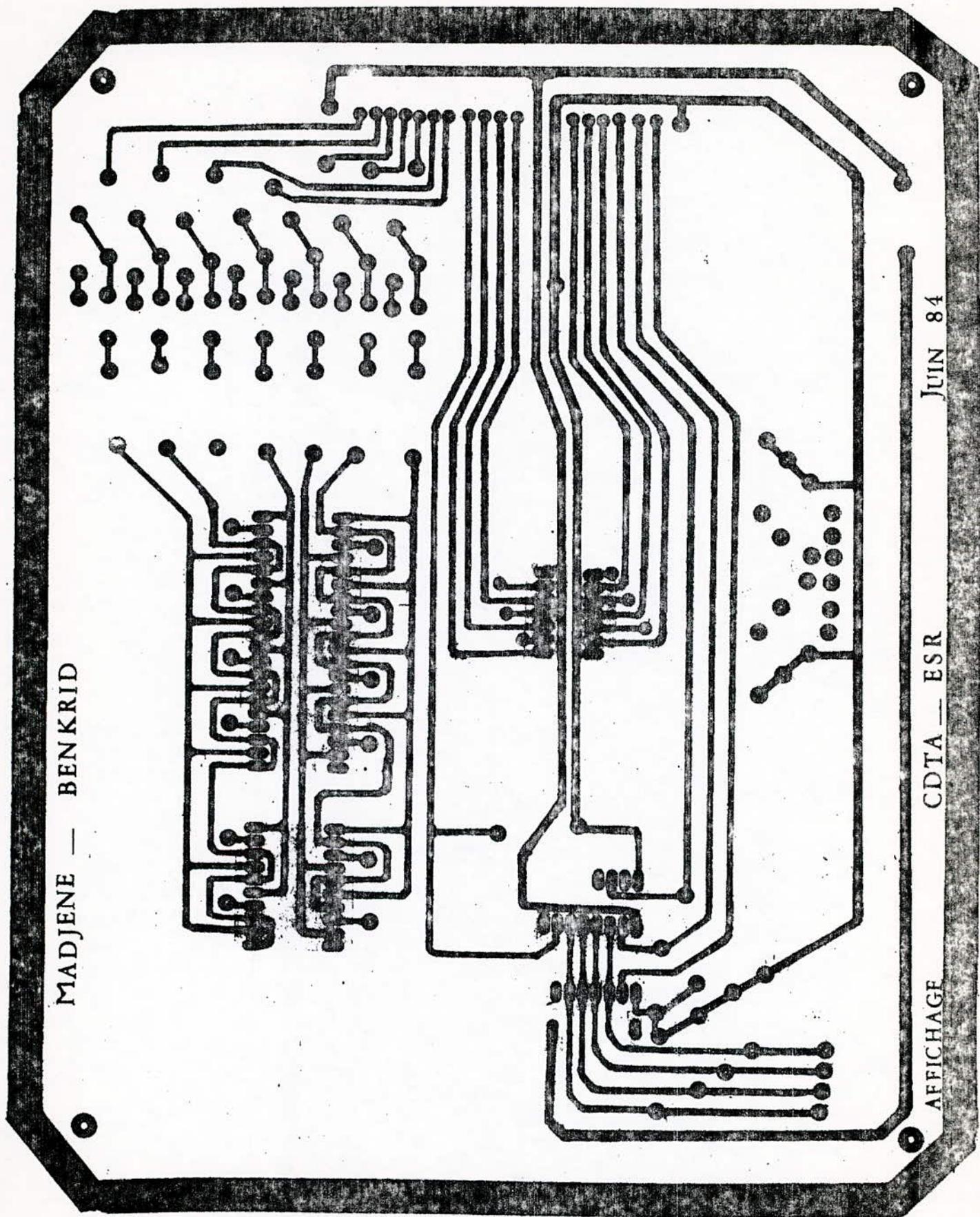


fig: 5a ; côté composants - Echelle : 1

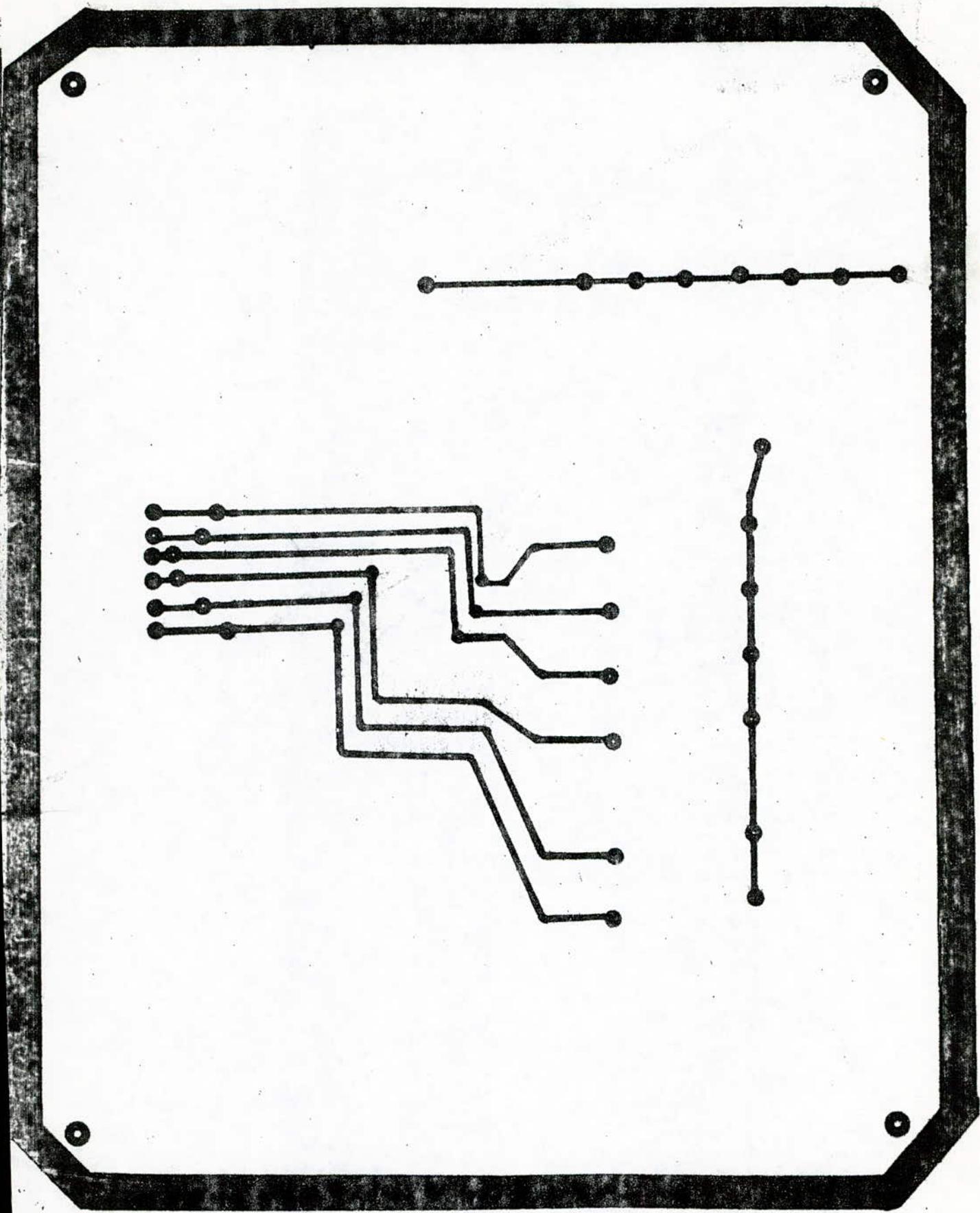
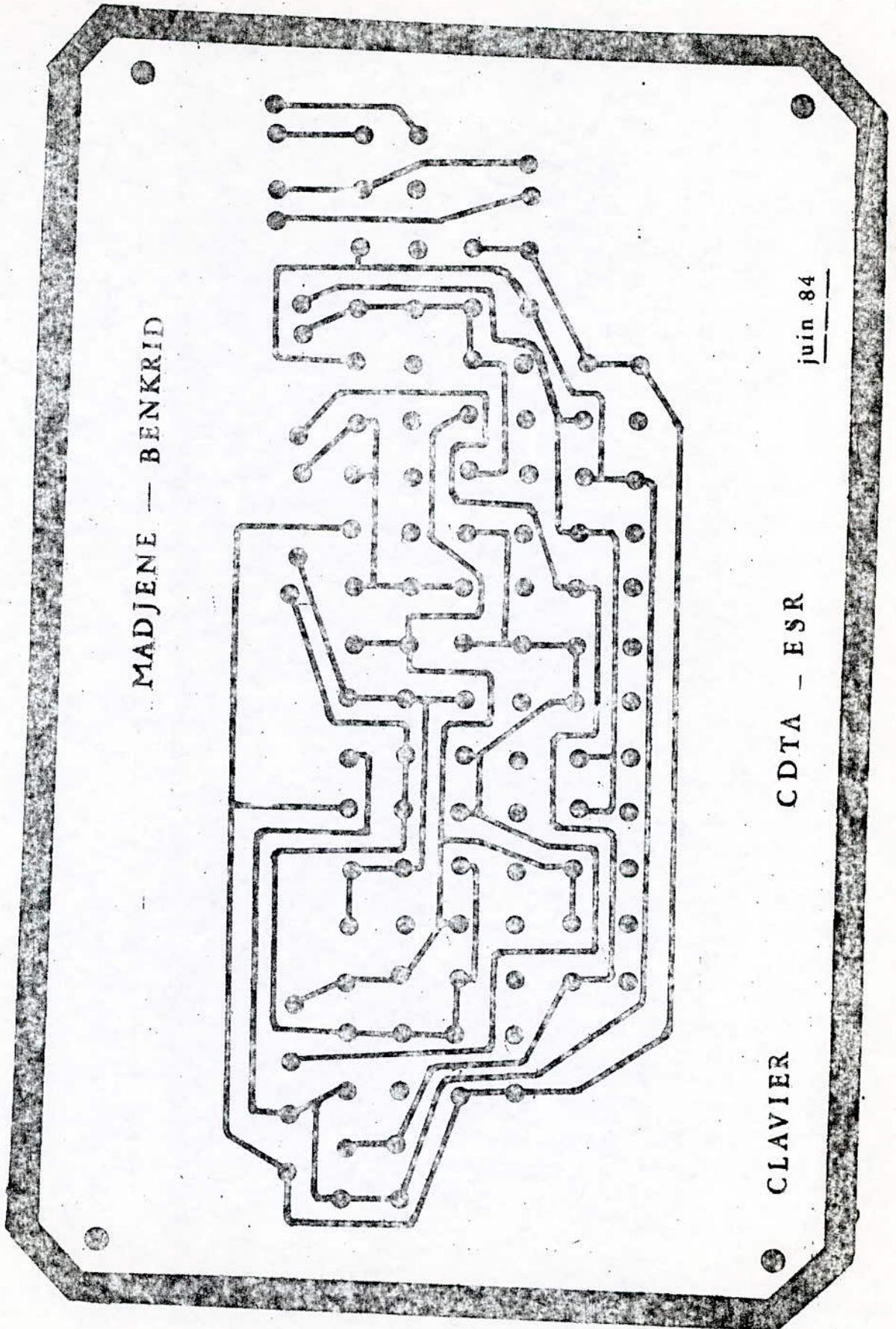


fig: 56

Circuit Imprime du Clavier
côté cuivre - Echelle: 1

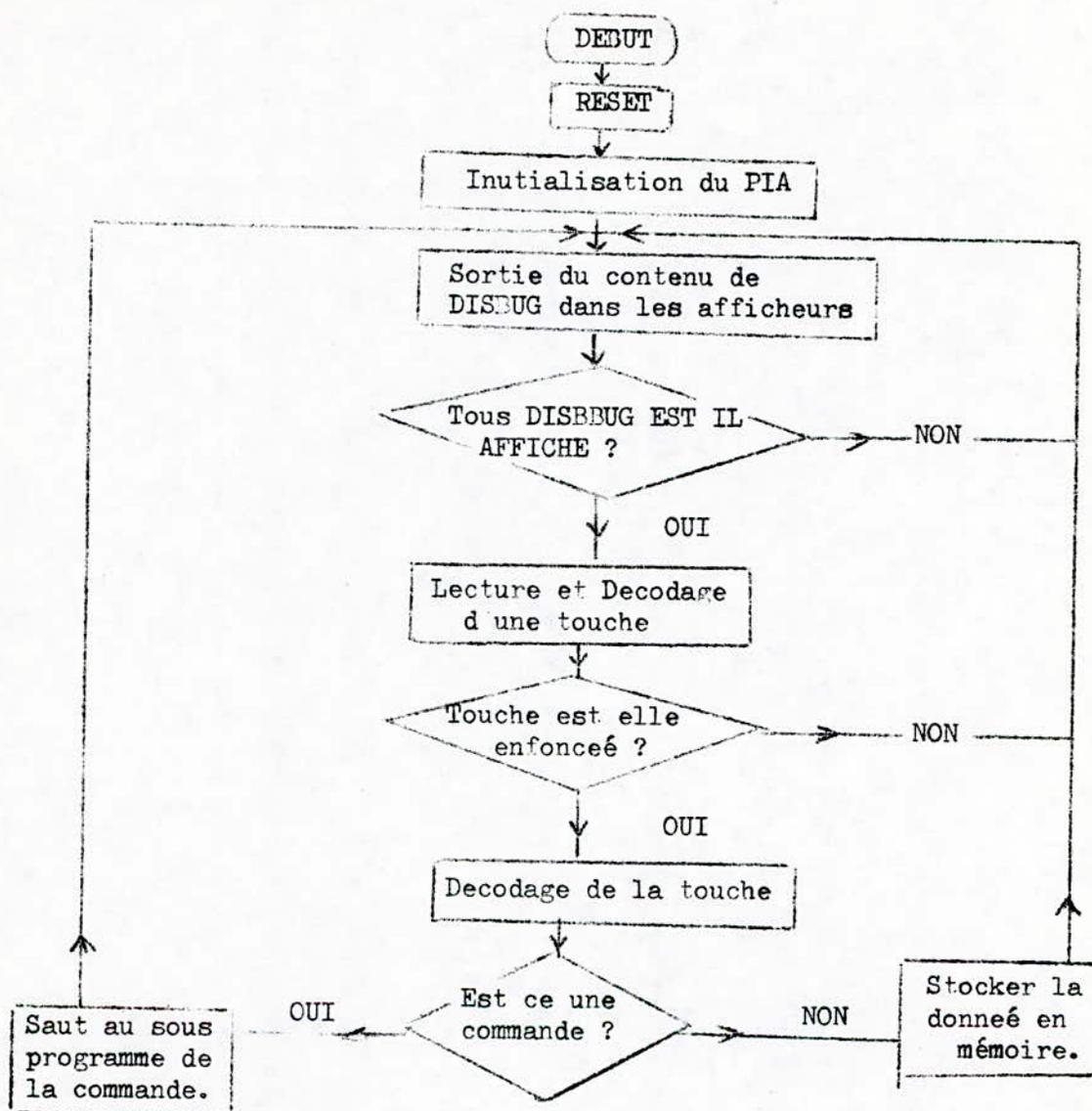


VI PROGRAMME DE GESTION DU SYSTEME ENTREE/ SORTIE.

(FONCTION DE J. BDG.)

1) Organigramme générale de gestion.

Le fonctionnement de la carte est contrôlé à partir du clavier :
gère à l'aide du programme de gestion dont l'organigramme est
le suivant :



Cet organigramme représente le déroulement de plusieurs sous programmes.

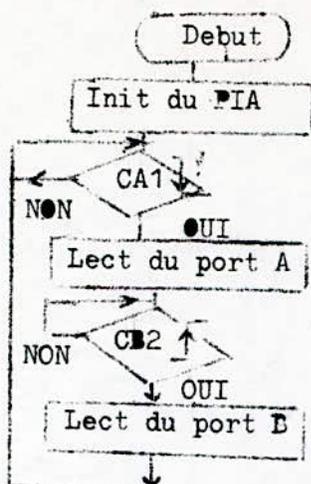
2) Les différents sous programmes:

a) Initialisation du PIA

Du fait que ORx et DDRx ont même adresse il faut programmer CRx puis DDRx

Comme notre système nécessite que les deux ports du PIA soient en sortie sauf PA7 en entrée on aura :

Organigramme



Sous programme

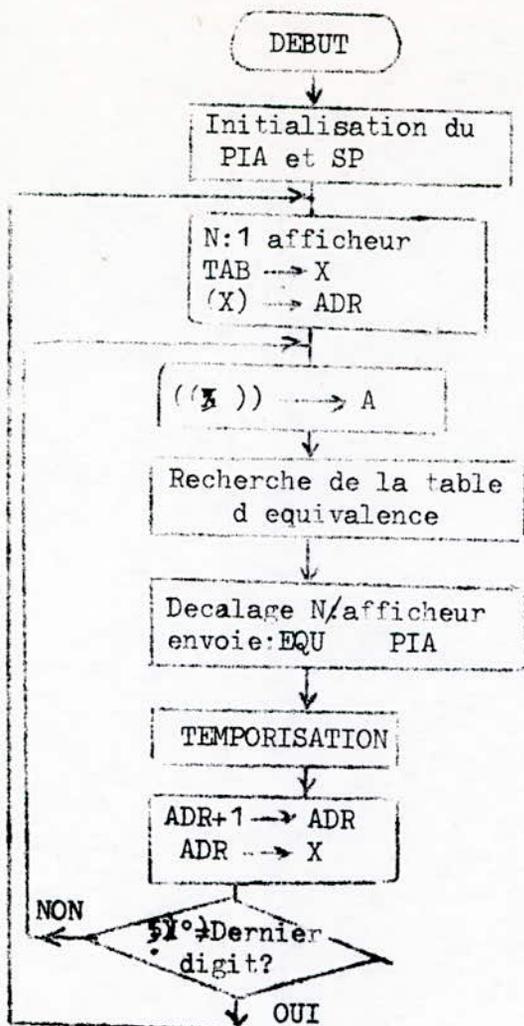
Langage assembleur

commentaire.

CLRA		
STAA	PIA CRA	
STA A	PIA CRB	Initialisation
LDA A	=\$7F	
STA A	PIA DRA	Port A en sortie(Sauf PA7)
LDA A	=\$ ff	
sta a	PIA	Port B en sortie
LP1 LDA A	PIA CRA	
BPL	LP1	Test de CRA7
LDA A	PIA ORA	Lecture Port A
LP2 LDA B	PIA CRB	
ROL B		
BPL	LP2	Test de CRB 6
STA A	PIA ORB	Lecture du Port B
LDA A	PIA ORB	Lecture fictive de DRB
BRA	LP1	

B) Sous programm d'affichage.

L affichage est constitué de six digits (ou caracteres affichables) que l'on suppose rangés en mémoires à partir d'une adresse: DIG. A chaque caractere est associé un code . Voir le tableau d'equivalence des segments .



PROGRAMME

```

PIA DRA EQU $ EBFC
PIA ORA EQU PIA DRA
PIA CRA EQU $EBFD
PIA DRB EQU $EBFE
PIA ORBB EQU PIA DRB
PIA CRB EQU $EBFF
TA B EQU $1000
ADR EQU:
  
```

AFFICH CLR A

```

STA A PIA CRA
STA A PIA CRB
LDA A +$7F
STA A PIA DRA
LDA A +$2F
STA A PIA DRB
LDA A +$04
STA A PIA CRA
STA A PIA CRB
  
```

```
LD S += $6000
```

DEBUT CLR B

```

SEC
LD X +$TAB
ST X ADR
LDA A 0, X
JSR EQUIV
STA A PIA ORA
ROL BB
STA B PIA ORB
  
```

```

JSR TEMPO
INC ADR+1
LD X ADR
  
```

```

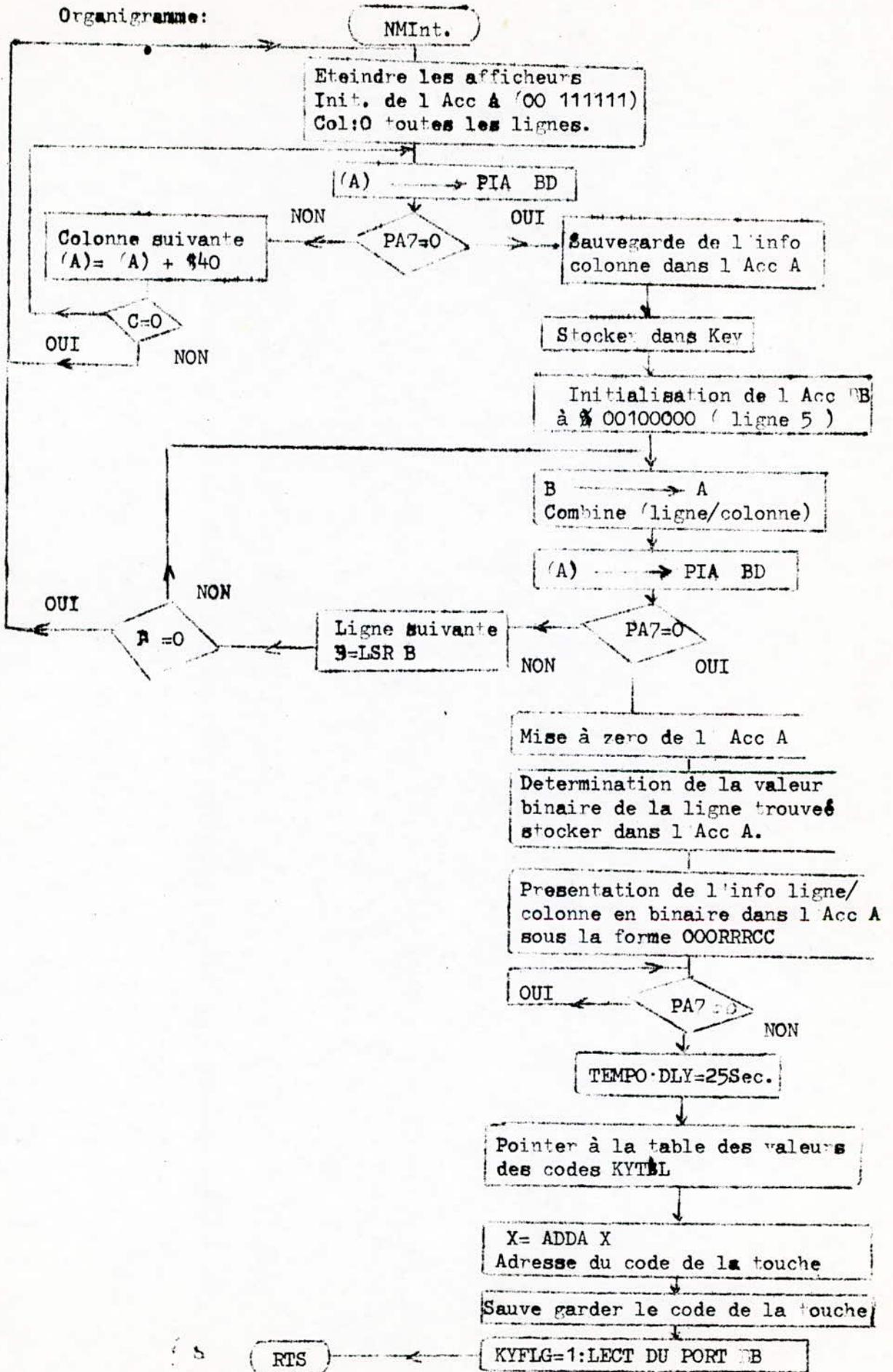
TEMPO LD X +$COORD
RETOUR DE X
BNE RETOUR
RTS
  
```

Suite

```

CPX +$TAB +5
BNE ENCORE
BRA DEBUT
EQUIV LD X += $ E3CA
TEST A
TEST BEQ FIN
INX
DEC A
BRA TEST
FIN LDA A 0 X
RTS
  
```

C) Sous programme du decodage des touches du clavier.



Sous programme

Langage	Assembleur	Commentaire
	GET LDX *PIA	
	LEAA *\$FF	
Recherche de la colonne.	STAA KP COL X	Eteindre les afficheurs
	LDAA *%00111111X	Colonne 0
	LPCOL STAA KP ROW X	Stocker l'info dans key
	TEST KP COL X	
	BPL COL FND	
	ADDA *\$40	Selection du multiplexeur
	BCC LP COL	BRANCHEMENT FIN TOUS COL
	BRA GET	Touche enfonceé Retour
	COL FND ANDA *%11000000	
	STAA Key	
	LDAB *%00100000	Ligne:5
Recherche de lp row la ligne.	TBA	
	ORAA Key	Combine ligne/colonne
	STAA Key ROW	
	TEST KP COL,X	
	BPL ROW FND	
	LSRB	Decalage (Selection des lignes.)
	BNE LP ROW	
	BRA GET	
	ROW FND CLRA	
	LP FND PCS DUN ROW	
	INC A	Determination de la valeur binaire de la ligne
	BRA LP FND	
	DUN ROW ROL Key	
	ROL A	Ligne/Colonne sous forme 00ORRRCC
	ROL Key	
	ROL A	
Le code de la touche enfonceé.	CLOP TEST KP COL?X	Test :touche enfonceé?
	BPL CLOP	
	JSR DLY 25	Tempo. de rebondissement
	LDX *KYTR	
	JSR ADDAX	ADDR. OF Key CODE
	LDAA ?X	
	STAA Key	Sauve garder Valeur du Key
	ldab *	
	STAB KY FLG	
	LDAB PIA ROW	CLEAR NMI.
	RTS	

Sous programme de conversion: "Décimal Hévadécimal."

Pour des pratiques, l'utilisateur introduit les valeurs en décimal seulement le micro processeur ne peut pas traiter ces valeurs il faut qu'elles soient en hévadécimal. D'où la conversion décimal hévadécimal est nécessaire.

L organigramme de conversion : Voir fig: 10

Le programme de conversion:

Langage assembleur	Langage machine du MC 6809.	
LDY ≠ Mi	400 10 8E 12 00	
LDD ≠ X _{decimal}	404 CC 27 60	
LP6 LDU ≠ Ø	407 CE 00 00	
LP1 ADDD FFEA 5 16)	40A C3 FFEA (16)	
BCC FIN	40D 24	
LEAU 1, U	40E 413 ;0 (Calcul du pas).	
BRA LP1	40F 33 61	
FIN ADDD ≠ 16	411 20	
CMPB ≠ 09	412 40A ;0	
BLE LP2	413 C3 00 16	
LDX ≠ 10	416 C1 09	
STX (0000) Mj	418 2F	
LDX ≠ A	419 42C :0	
LP4 CMPB 00 Mj	41A 8E 10	
BEQ LP3	41C 9F 00 00	
LEAX 1, X	41F 8E 0A	
INC Mj	421 D1 00	
BRA LP4	422 27	
LP3 TFR X, D	423 42A ;0	
LP2 STD (Y)	424 30 01	
CMPU ≠ 16	426 0C 00	
LP5 BLO	428 20	
TFR U D	429 421 ;0	
BRA LP6	42A 1F 10	
LP3 LDX ≠ 10	42C ED A2	
STX (0000) Mj	42E 11 83 16	
B ≠ A	431 25	
LP8 CMPR 00 Mj	432 437 ;0	
BEQ LP7	433 1F 30	
LEAX 1 X	435 20	
INC Mj	436 407 ;0	
BRA LP8	437 8E 10	
LP7 STX (Y)	439 9F 00 00	
RTS.	43C 8E 0A	
	43F D1 00	
	440 27	
	441 448 :0	
	442 30 01	
	444 0C 00	
	446 20	Résultat:
	447 43F ;0	Décimal
	448 AF A2	hévadécimal
	44A 39	2760 AC8

Après le traitement: les résultats sont des valeurs hévadécimal: pour faciliter leurs interprétations: on utilise le sous programme de conversion hévadécimal décimal.

L organigramme de conversion : Voir fig: 11

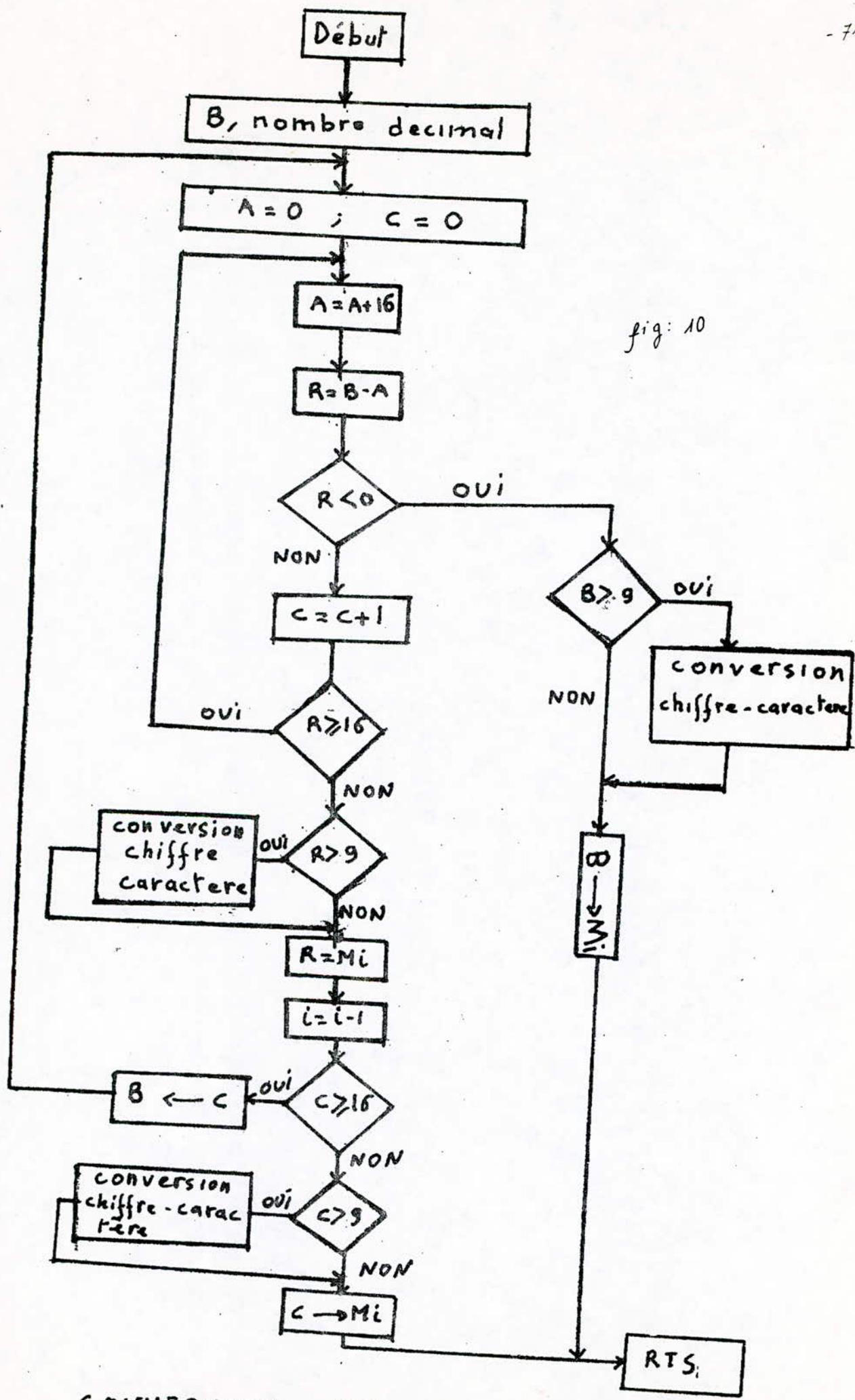


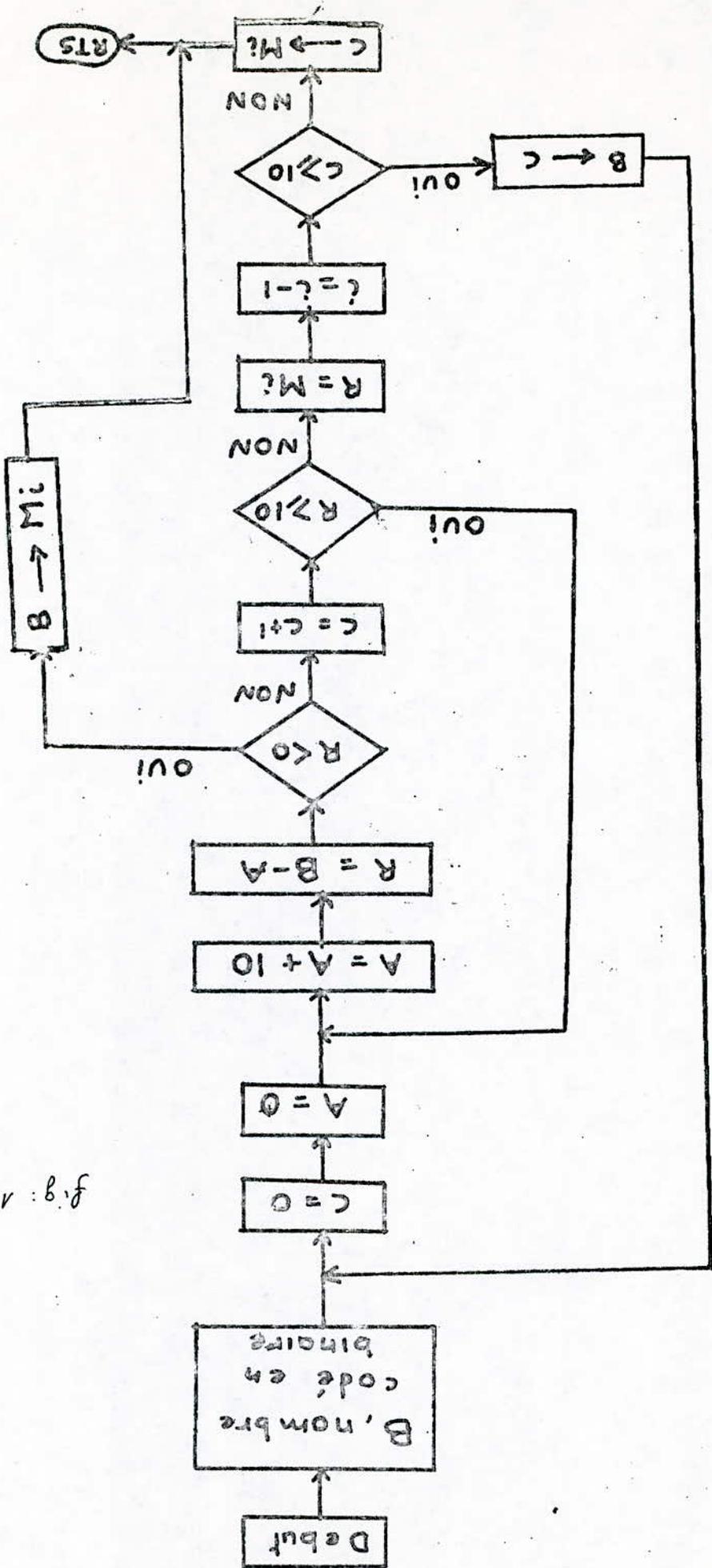
fig: 10

CONVERSION DECIMAL - HEXADECIMAL

Sous-programme de conversion

binnaire - decimal

fig. 11



Sous programme de conversion: "Héxadécimal-Décimal"

Langage assembleur	Langage machine
LDY ## Mi	1000 10 8E 14 00
LDD ## X _{hexa}	1004 CC OFFF (255)
ST LDX ## 00	1007 8E 0000
LP ADDD ## FFF6 (10)	100A C3 FF F6
BCC FIN	100D 24 04
LEAX 1;X	100F 30 01
EBRA LP	1011 20 F7
FIN ADDD ## 000A	1013 C3 00 0A
STD (Y)	1016 ED A2
CMP X ## 000A	1018 8C 00 0A
BLD SA	101B KKK 25 04
TFR X ;D	101D 1E 10
EBRA ST	101F 20 E6
SA STX ; (-Y)	1021 AF A2
RTS	1023 39

Résultats:

Héxadécimal	Décimal
OFFF	255

Quelques sousroutines propres à l'unité AM 9511.

Sous programme d'initialisation.

Ce sousprogramme consiste à vider tous les registres internes et la pile de toute operande ou commande. Cette operation est necessaire avant chaque instruction exécutée par l'AM 9511.

On met un zero dans l'accumulateur qui va servir de compteur .On envoie l'octet 00 dans la pile de l'AM9511 par l'intermediaire de l'accumulateur A .Après chaque envoie le compteur (l'acc B) s'incrémente jusqu'a ce que la pile de l'AM9511 ne renferme que des zeros.

Langage assembleur	langage machine
CLR A	200 4F
CLR B	201 5F
LP STA EEBF8	202 B7 EEB8
INC B	205 5C
CMP ## \$ 10	206 C1 10
BNE LP	208 2F
RTS	209 F8
	20A 39

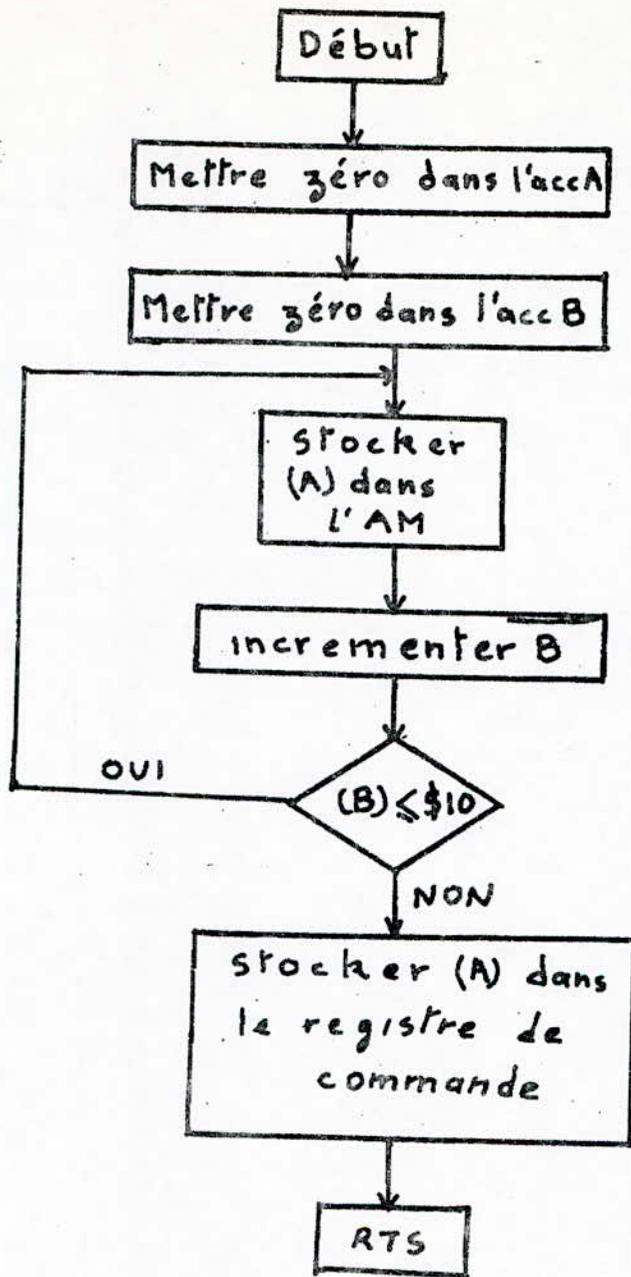


figure a

sous-programme d'initialisation des registres internes et de la pile de l'AM 9511

Sous programme de conversion : "Vergule fixe en vergule flottant.

Cette sousroutine permet la conversion des nombres à vergule fixe en vergule flottante. L'opération inverse peut aussi être exécutée par l'AM 9511. Ces opérations sont valables pour des nombres de *16 bits ou de 32 bits.

A titre d'exemple nous traitons la conversion du fixe au flottant avec 32 bits. Cette opération est nécessaire pour calculer les fonctions trigonométriques; tel que : (Le cosinus ; Le sinus ...)

Exemple:

Valeur en binaire	Valeur hexadécimal
0011 0110	\$ 36
Représentation du binaire flottant	
0.1101 1000 0000 000 2 ⁶	
Format du binaire flottant de l'AM 9511.	
0000 0100.1101 1000 0000 0000 0000 0000	
Qui correspond à la valeur :06 D8 00 00	

Le sous programme de conversion Fixe - flottant.

Langage assembleur	Langage machine
CLR A	400 4F
CLR B	401 5F
LP1 STA EBF8	402 B7 EBF8
INC BB	405 5C
CMP B ≠ \$ 10	406 C1 10
BLE LP1	408 2F
CLR EBF8	409 F8
CLR EBF9	40A 7F EBF8
LDA ≠ \$36	40D 7F EBF9
STA EBE8	410 86 36
LDA 00	412 B7 EBF8
STA EBE8	415 86 00
LDA FLTS	417 B7 EBF8
STA EBE9	41A 86 9D
LDA EBF8	41C B7 EBF8
STA 1000	41F B7 EBF8
LDA EBF8	422 B7 1000
STA 1001	425 B6 EBF8
LDA EBE8	428 B7 1001
STA 1002	42B B6 EBE8
LDA EBF8	42E B7 1002
STA 1003	431 B6 EBF8
SWI	434 B7 1003
	437 3F

ADRESSES:	1000	1001	1002	1003
RESULTATS:	06	D8	00	00

Programme de la fonction : $f(II)=\cos(II)$

Langage assembleur	Langage machine
CLR A	300 4F
CLR B	301 5F
LP1 STA EBE8	302 B7 EBE8
INC B	305 5C
	306 C1 10
	308 2F
CMP B \$10	309 F8
BLE LP1	30A 86 01
LDA 0 01	30C B7 EBE8
STA EBE8	30F 86 00
LDA 0 00	311 B7 EBE8
STA EBF8	314 86 1D
LDA FLTS	316 B7 EBE9
STA EBF9	319 7F EBF9
CLR EBF9	31C 86 1A
LDA PUPI	31E B7 EBF9
STA EBF9	321 7F EBF9
CLR EBF9	324 86 12
LDA FMUL	326 B7 EBF9
STA EBF9	329 7F EBF9
CLR EBF9	32C 86 19
LDA XCHF	32E BB7 EEBF9
STA EBF9	331 7F EBF9
CLR EBF9	334 86 83
LDA COS	336 BB7 EBF9
STA EBF9	339 7F EBE9
CLR EBF9	33C B6 EBE8
LDA EBF8	33F B7 2200
STA 2200	342 B6 EBF8
LDA EBE8	345 B7 2201
STA 2201	348 B7 EBF8
LDA EBF8	34B B7 2202
STA 2202	34E B6 EBF8
LDA EBE8	351 B7 2203
STA 2203	354 3F
LDA EBF8	
STA 2200	ADRESSES: 2200 2201 2202 2203
LDA EBE8	RESULTATS/ 81 80 00 00
STA 2201	
LDA EBF8	
STA 2202	
LDA EBE8	
STA 2203	
SWI	

Programme de la fonction :f(II)=SIN(II/2)

Langage assembleur		Langage machine	
	CLR A	3000	4F
	CLR B	3001	5F
LP1	STA EBF8	3002	B7 EBF8
	INC B	3005	5C
	CMP B ## \$ 10	3006	C1 10
	BLE LP1	3008	2F
	LDA PUPI	3009	3002;0
	STA EBF9	3009	F8 (Pas de branchement)
	LDA NOP	300A	86 1A
	STA EBF9	300C	B7 EBF9
	LDA 02	300F	86 00
	STA EBF8	3011	B7 EBF9
	LDA 00	3014	86 02
	STA EBF8	3016	B7 EBF8
	LDA 00	3019	86 00
	STA EBF8	301B	B7 EBF8
	LDA 00	301E	86 00
	STA EBF8	3020	B7 EBF8
	LDA FLTD	3023	86 00
	STA EBF9	3025	B7 EBF8
	LDA FDIV	3028	86 1C
	STA EBF9	302A	B7 EBF9
	LDA XCHF	302D	86 13
	STA EBF9	302F	B7 EBF9
	LDA SIN	3032	86 19
	STA EBF9	3034	B7 EBF9
	LDA EBF8	3037	86 02
	STA 4500	3039	B7 EBF9
	LDA EBF8	303C	B6 EBF8
	STA 4501	303F	B7 4500
	LDA EBF8	3042	B6 EBF8
	STA 4502	3045	B7 4501
	LDA EBF8	3048	B6 EBF8
	STA 4503	304B	B7 .4502
	SWI	304E	B6 EBF8
		3051	B7 4503
		3054	3F

ADRESSES: 4500 4501 4502 4503
 RESULTATS: 01 80 00 00

C H A P I T R E : VI

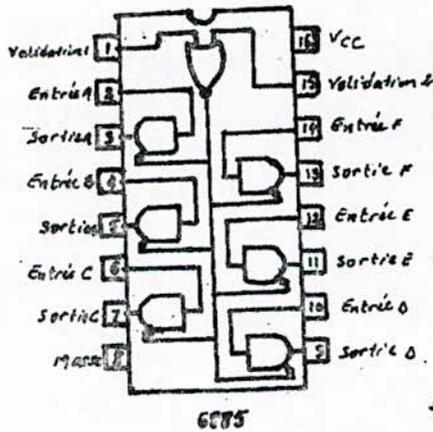
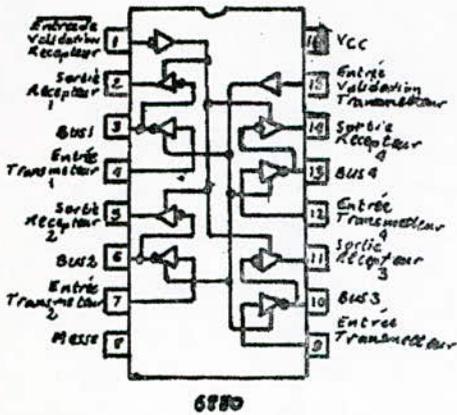
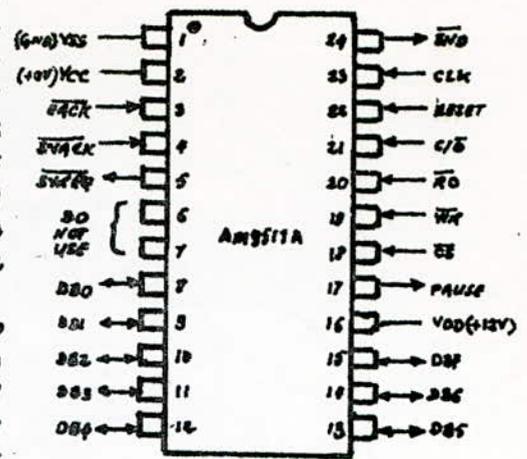
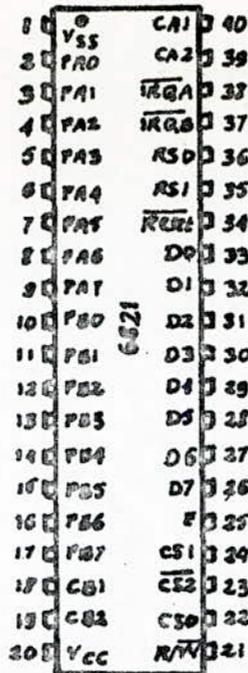
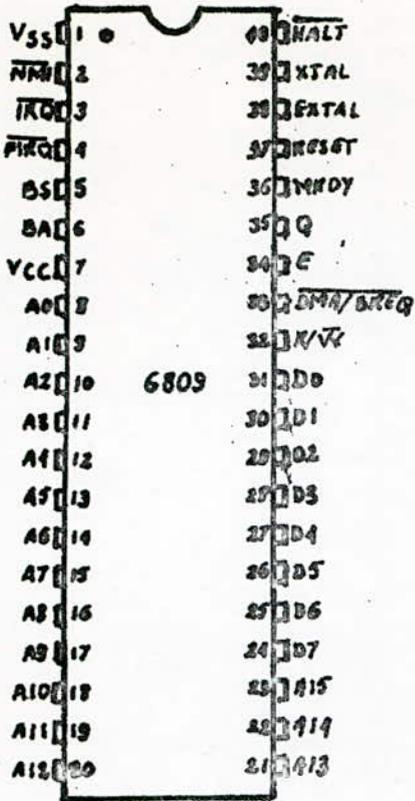
CONCLUSION ET RECOMMANDATIONS.

Notre projet comprend deux parties essentielles
Etude des différents algorithmes de la transformée de Fourier.
Réalisation d'une carte CPU et d'un affichage 7 segments et
d'un clavier .

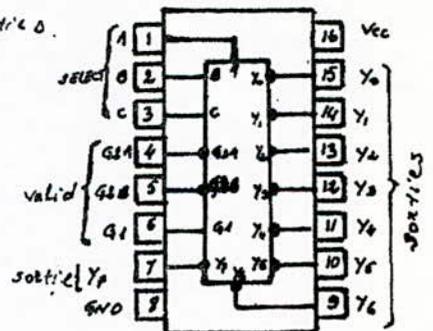
Le langage FORTRAN nous a permis de constater la similitude
entre la méthode directe et la méthode rapide au point de vue
précision . Il nous a aussi permis de mettre en évidence l'importance
de la FFT pour le traitement d'un nombre d'échantillon élevé .
Ce processus accélère grandement le traitement, comparativement
à celui de la méthode directe DFT.

La réalisation tardive de la carte CPU nous a causé beaucoup
de problème à cause de la non compatibilité du système de
développement avec la carte .

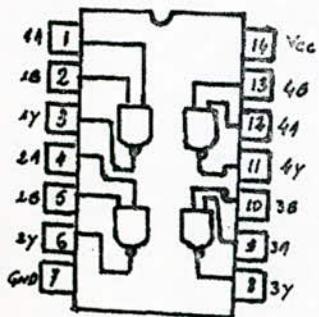
Pour la continuité de notre travail ; nous proposons à ce que
les tests hardware et software se fassent directement sur
la machine elle-même afin d'adapter le travail fait à la
machine autonome .



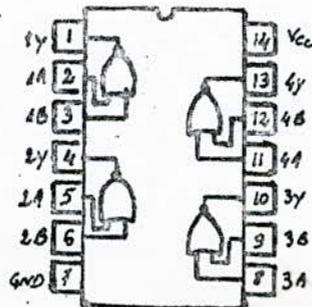
6875



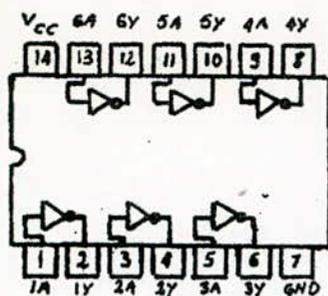
74LS138



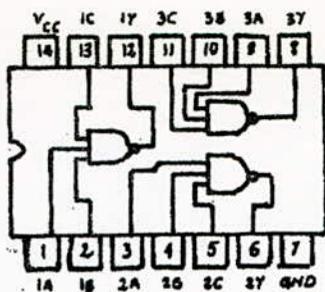
SN7400N



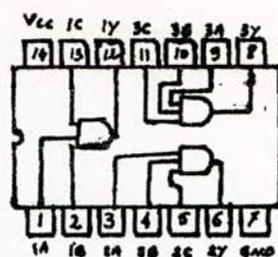
SN7402N



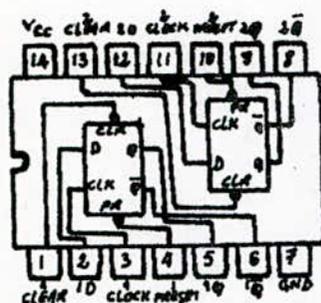
SN 7404 N



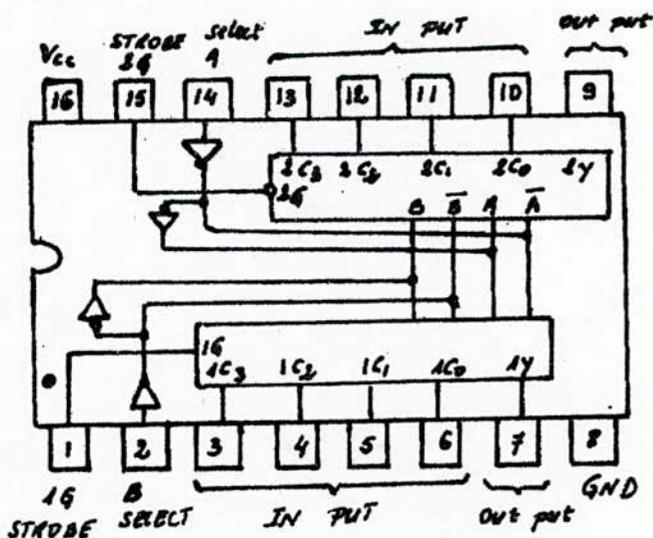
SN 7410 N



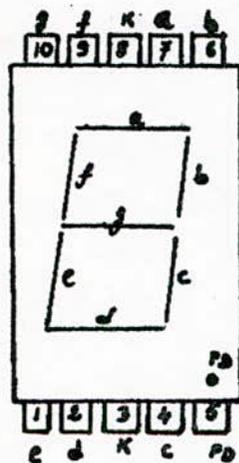
SN 7411 N



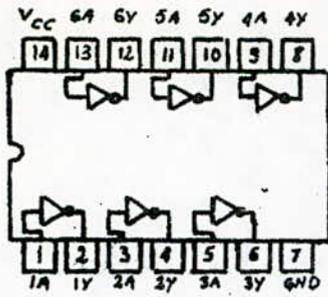
SN 7474 N



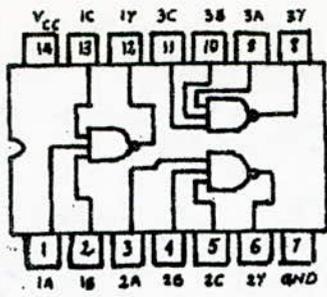
MC 14539



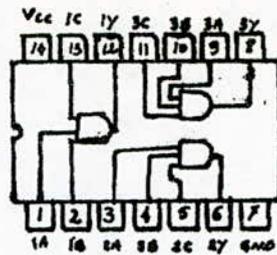
DL 500



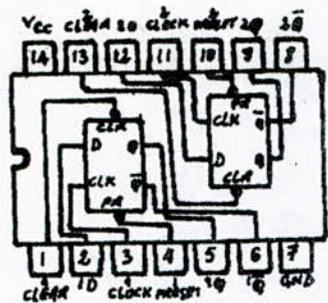
SN7404N



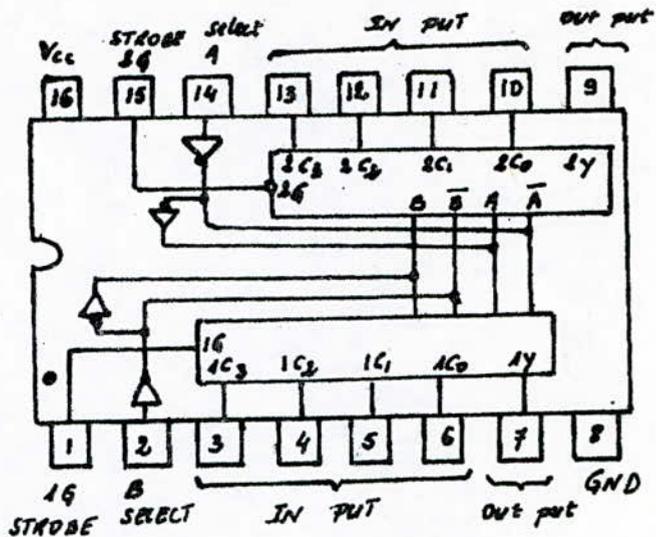
SN7410N



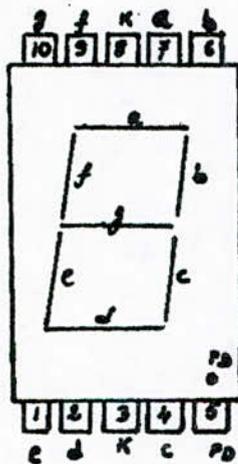
SN7411N



SN7474N



MC 14539



DL 500