



ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT D'ELECTRONIQUE

PROJET DE FIN D'ETUDES

S U J E T

MISE AU POINT D'UN PROGRAMME
DE TRACÉ GRAPHIQUE ET
APPLICATION A L'ETUDE DE LA
TRANSFORMÉE DE FOURIER

Proposé par :
M^r H. DJELOUAH

Etudié par :
FATMA STIHI

Dirigé par :
M^r H. DJELOUAH

PROMOTION : JUIN 1985

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

MISE AU POINT D'UN PROGRAMME
DE TRACÉ GRAPHIQUE ET
APPLICATION A L'ETUDE DE LA
TRANSFORMÉE DE FOURIER

Proposé par :

M^r.H. DJELOUAH

Etudié par :

FATMA STIHI

Dirigé par :

M^r.H. DJELOUAH

PROMOTION : JUIN 1985

-o-o= R E M E R C I E M E N T S =o-o-

=====

JE TIENS A REMERCIER VIVEMENT MON
PROMOTEUR H.DJELOUAH POUR M'AVOIR PROPOSE CE SUJET
ET POUR M'AVOIR GUIDEE ET CONSEILLEE JUDICIEUSEMENT
LE LONG DE L'ELABORATION DE CE TRAVAIL.

MES REMERCIEMENTS VONT EGALEMENT A
MONSIEUR TATAH DIRECTEUR DU C.D.C.E POUR M'AVOIR
AIMABLEMENT ACCEILLIE DANS SON CENTRE.

JE REMERCIE TOUS CEUX QUI ONT
CONTRIBUE DE PRES OU DE LOIN A L'ACCOMPLISSEMENT DE
CE TRAVAIL ET A TOUS LES ENSEIGNANTS QUI ONT
PARTICIPE A MA FORMATION VEUILLENT TROUVER ICI
L'EXPRESSION DE MA PROFONDE RECONNAISSANCE.

/-) E D I C A C E S

- /=) /))on Mari
- /-) /))a Fille IMANE
- /-) /))a Grand-mere
- /-) /))es Chers parents
- /-) /))es Freres et ma soeur
- /-) Toutes mes Amies

/)/ACERA

S O M M A I R E

INTRODUCTION.	PAGE
CHAPITRE I:PROCEDURE DE MISE AU POINT D'UN SOUS-PROGRAMME DE TRACE GRAPHIQUE -----	2
I.1 Recherche du minimum -----	5
I.2 Recherche du maximum -----	5
I.3 Tracé des axes -----	9
I.4 Ecriture de l'axe horizontal -----	10
I.5 Ecriture de l'axe vertical -----	21
I.6 Ecriture du titre -----	23
I.7 Ecriture de l'unité et du facteur multiplicatif sur l'axe horizontal -----	24
I.8 Ecriture de l'unité et du facteur multiplicatif sur l'axe vertical -----	25
I.9 Tracé graphique -----	26
CHAPITRE II:PROCEDURE D'UTILISATION DU SOUS-PROGRAMME DE TRACE GRAPHIQUE ET APPLICATIONS-----	27
II.1 Fonction projectrice -----	32
II.2 Fonction sinus -----	33
II.3 Application du sous-programme de tracé graphique sur T.F -----	34
CHAPITRE III: TRANSFORMATION DE FOURIER -----	35
III.1 Series de Fourier -----	35

III.2	Intégrale de Fourier	-----	36
III.3	Cas des fonctions physiques	-----	37
III.4	Conditions d'existence de la transformation de Fourier	-----	38
III.5	Quelques propriétés de la transformée de Fourier	-----	39
III.5.1	Propriété de linéarité	-----	39
III.5.2	Propriété de parité	-----	39
III.5.3	Propriété de Similitude	-----	39
III.5.4	Propriété de translation	-----	39
III.6	Théorème de Plancherel	-----	40
III.7	Formule de Poisson	-----	40
CHAPITRE IV: FILTRAGE TEMPOREL			----- 41
CHAPITRE V : ECHANTILLONNAGE			----- 43
V.1	Généralités	-----	43
V.2	Théorème d'échantillonnage	-----	43
CHAPITRE VI: TRANSFORMÉE DE FOURIER DISCRÈTE ET			
	F F T	-----	48
VI.1	Sous-programme de calcul de la TF.D	-----	49
VI.2	Calcul de la F F T	-----	50
VI.2.1	Séparation des parties réelles et imaginaires de la F F T	-----	51
VI.2.2	Calcul du module et de la phase de la FFT	-----	52
VI.3	Quelques remarques sur la commande FFT	-----	53

VI.4	Comparaison entre TFD et FFT	-----	55
CHAPITRE VII :UTILISATION DU SOUS-PROGRAMME POUR			
COMMENTER LES PROPRIETES DE LA			
	T.F	-----	60
VII.1	Influence de la frèquence d'échantillonnage	-----	60
VII.2	Influence de la fenêtre temporelle	-----	66
VIII.3	Influence du décalage temporel	-----	73
CHAPITRE VIII: REGLES PRATIQUES POUR AMELIORER LES			
RESULTATS OBTENUS PAR FFT. -----78			
VIII.1	Moyennage du signal	-----	78
VIII.2	Elimination de la composante continue	-----	81
	CONCLUSION	-----	82
	ANNEXE		
	BIBLIOGRAPHIE		

I N T R O D U C T I O N

Dans le passé, l'analyse de Fourier a été utilisée comme une approche mathématique intéressante pour obtenir des informations dans le domaine des fréquences, mais généralement trop difficile à appliquer dans la plupart des cas pratiques, et même avec l'arrivée des ordinateurs, l'analyse de Fourier demandait trop de temps de calcul pour être largement utilisée.

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 algorithm, plus connu sous le nom de Fast Fourier Transform ou F F T, est devenu un outil largement utilisé pour l'analyse de Fourier permettant une application rapide et économique des techniques de Fourier à une très grande variété de problèmes.

L'Objet de ce projet est la mise au point d'un programme de tracé graphique sur Micro-Ordinateur TEKTRONIX 4052 et l'application à l'étude de la transformée de Fourier.

Cette étude comporte deux parties:

Première partie : Description de la procédure de mise au point du sous- programme de tracé graphique."

Deuxième partie: Applications à l'étude des propriétés de la transformée de Fourier.

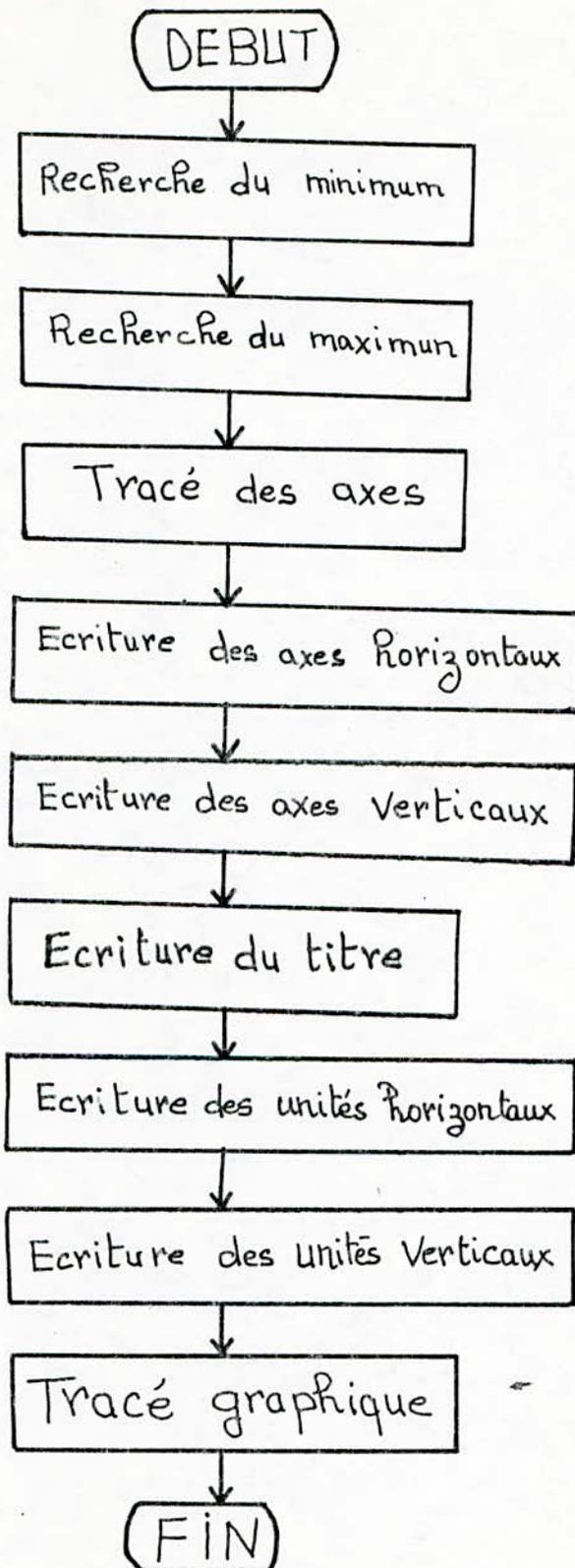
CHAPITRE I

PROCEDURE DE MISE AU POINT D'UN SOUS- PROGRAMME DE TRACE GRAPHIQUE

Ce sous-programme de tracé graphique est destiné à des utilisations diverses, il doit après informations fournies par l'utilisateur (Nombre de points d'échantillonnage, définition de la fonction, unités verticales et horizontales). Permettre la représentation graphique d'une fonction après:

- Tracé des axes.
- Graduations linéaires des axes.
- Ecriture des graduations suivant un format imposé.
- Ecriture des unités et commentaires définis par l'utilisateur.

La procédure suivie est décrite par l'organigramme de tracé graphique et le Schéma de la figure I.1.



ORGANIGRAMME DE TRACÉ GRAPHIQUE

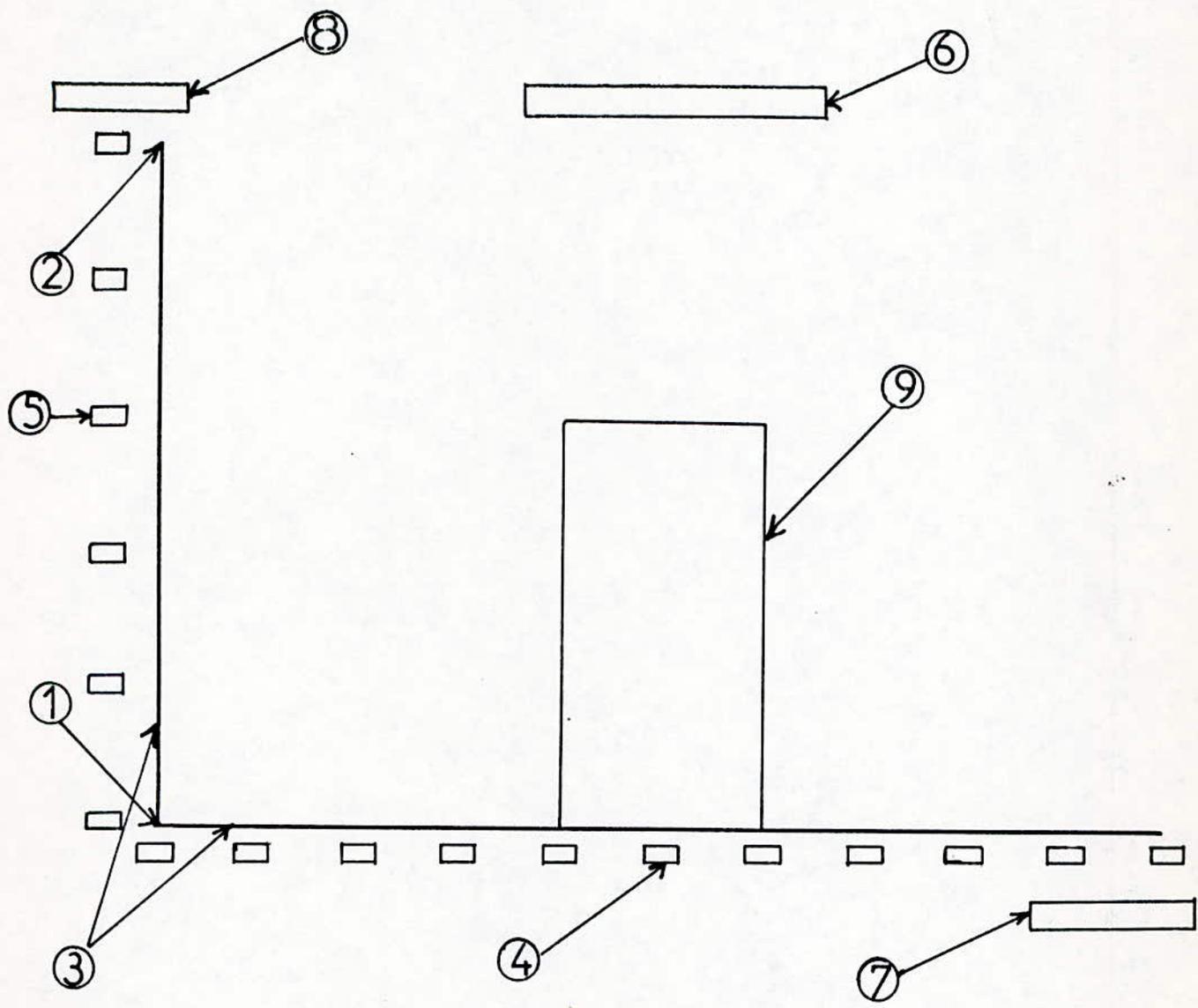


FIG I.1

I.1 RECHERCHE DU MINIMUM:

La valeur minimale est calculée grâce à l'instruction CALL"MIN":

FORME SYNTAXE:

N° DE LIGNE	CALL"MIN",	TABLEAU DE,	VARIABLES,	VARIABLE
		CHIFFRES	NUMERIQUE	NUMERIQUE

La valeur minimale est représentée par la première variable numérique, tandis que la deuxième variable numérique représente l'indice correspondant à la valeur minimale.

EXEMPLE:

N° de ligne CALL"MIN", W,M1,I

Le résultat de cette instruction est:

M1=Valeur minimale qui se trouve dans le tableau de chiffres (W) avec M1=W(I).

I.2 RECHERCHE DU MAXIMUM:

La valeur maximale est calculée grâce à l'instruction CALL"MAX".

FORME SYNTAXE:

N° DE LIGNE	CALL"MAX",	TABLEAU DE,	VARIABLE	,VARIABLE
		CHIFFRES	NUMERIQUE	NUMERIQUE

La première variable numérique représente la valeur maximale, tandis que la deuxième variable numérique représente l'indice correspondant à cette valeur.

EXEMPLE

N° DE Ligne CALL"MAX" , W,M2 ,I

Le résultat de cette instruction est:

M2=Valeur maximale qui se trouve dans le tableau de chiffres (W) avec M2=W(I).

Le sous programme de calcul de la valeur minimale et la valeur maximale se trouvent dans une mémoire ROM on aurait pû les déterminer par programmation.

Le fait de calculer M1 et M2 offre un intérêt important car cela permet de choisir une échelle convenable pour les axes (voir instruction AXIS), ainsi que la détermination des limites de l'espace de données (fenêtre) où le graphe peut être représenté.

Pour définir les limites de l'espace de données on utilise l'instruction WINDOW.

FORME SYNTAXE:

N° DE LIGNE WINDOW	EXPRESSION NUMERIQUE, EXPRESSION NUMERIQUE
	EXPRESSION NUMERIQUE, EXPRESSION NUMERIQUE

La première et la deuxième expression numérique spécifient respectivement la valeur minimale et la valeur maximale horizontales.

La troisième et la quatrième expression numérique spécifient respectivement la valeur minimale et la valeur maximale verticales.

Les données sélectionnées par WINDOW peuvent être représentées sur la totalité de l'écran ou sur une partie de l'écran.

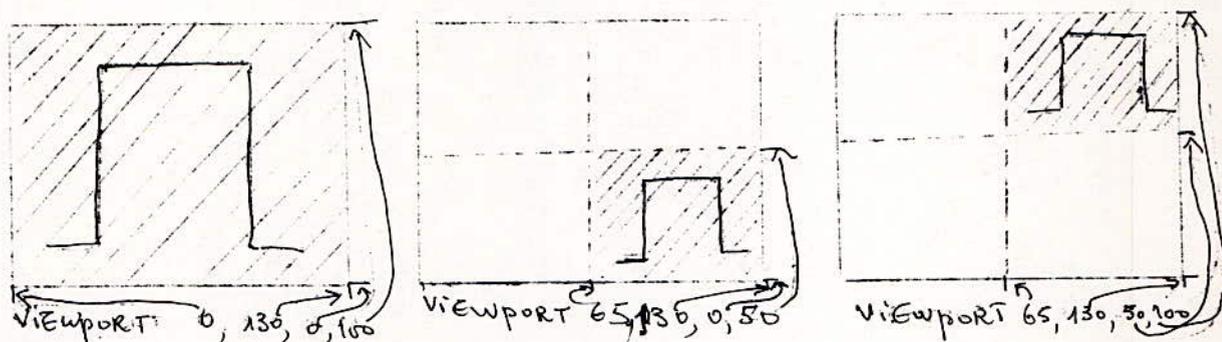
Le choix de la surface de l'écran dans laquelle les données sélectionnées par WINDOW (~~doivent être repré~~
~~xxxxxxx~~) se fait grâce à l'instruction VIEWPORT.

FORME SYNTAXE

N° DE LIGNE VIEWPORT	EXPRESSION NUMERIQUE, EXPRESSION NUMERIQUE
	EXPRESSION NUMERIQUE, EXPRESSION NUMERIQUE

EXEMPLE: REPRESENTATION DE LA FONCTION PROJECTRICE

(WINDOW-130,+130,-100,100)



Les paramètres associés à WINDOW sont exprimés dans l'espace de données. Par contre les paramètres associés à VIEWPORT sont exprimés en GDU (Unités de Représentation graphique).

La différence entre WINDOW et VIEWPORT est schématisée sur la figure (I.2).

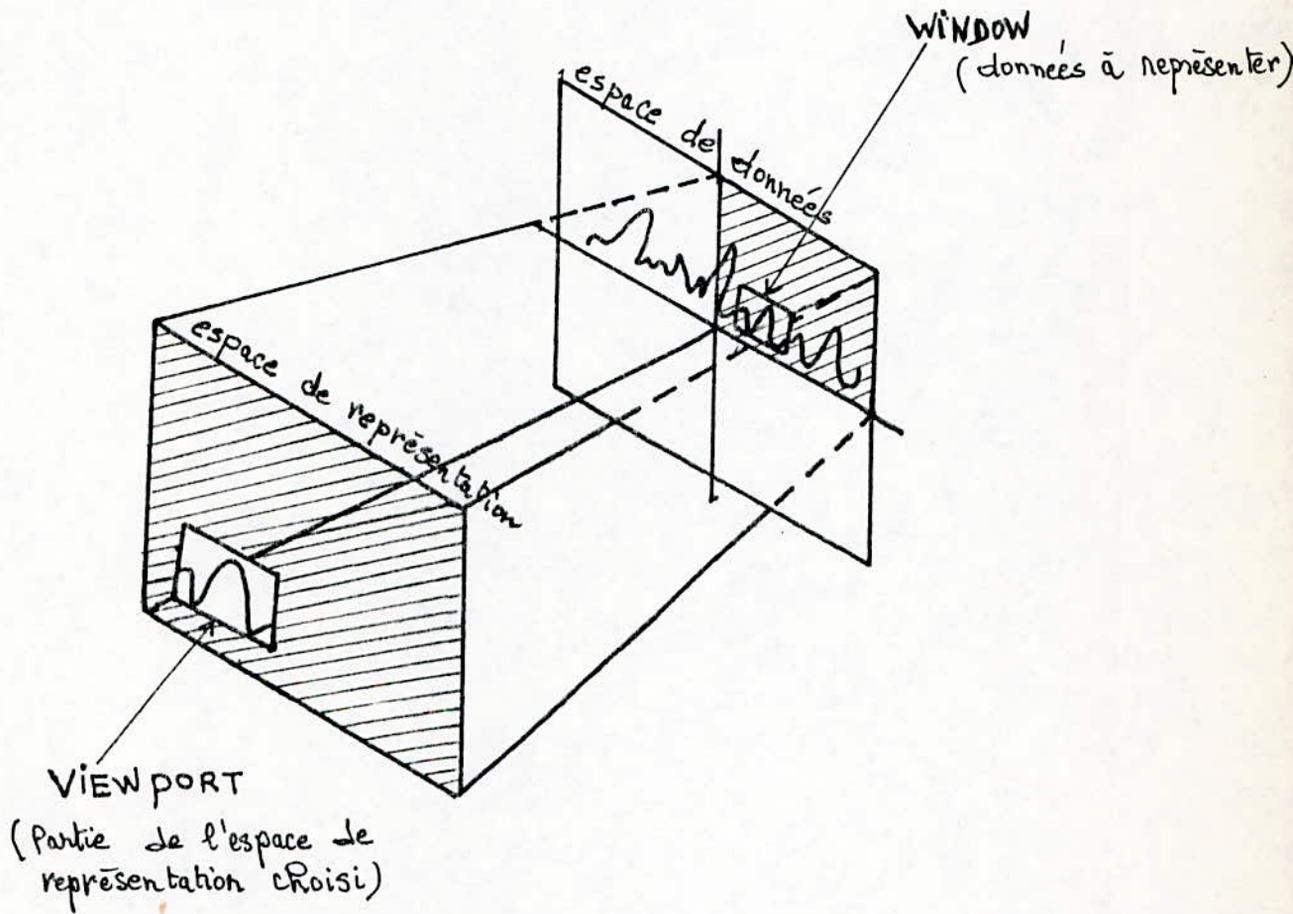


FIG I2

I.3) TRACE DES AXES:

Pour tracer les axes on utilise l'instruction AXIS.

FORME SYNTAXE:

N° DE LIGNE	AXIS	EXPRESSION NUMERIQUE, EXPRESSION NUMERIQUE
		EXPRESSION NUMERIQUE, EXPRESSION NUMERIQUE

La première et la deuxième expression numérique représentent respectivement la graduation sur l'axe horizontal et sur l'axe vertical.

La troisième et la quatrième expression représentent les coordonnées du point d'intersection des deux axes.

EXEMPLE:

N° de ligne AXIS (N-1)/10, (M2-M1)/5, 1, M1

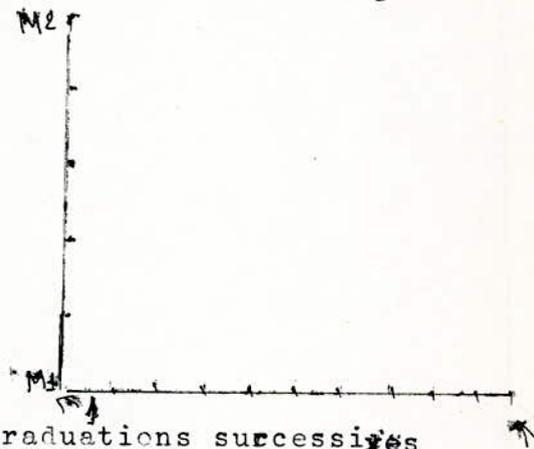
Cette instruction permet de tracer les axes et de les graduer d'après les expressions numériques qui se trouvent dans la forme syntaxe.

FIG I.3

La première expression représente les dix graduations sur l'axe horizontal dont l'écart entre deux graduations successives est de $(N-1)/10$.

La deuxième expression représente les cinq graduations sur l'axe vertical dont l'écart entre deux graduations successives est de $(M2-M1)/5$.

Le point de coordonnée (1, M1) représente le point d'intersection des deux axes. (FIG I.3).



REMARQUE:

L'instruction AXIS peut être utilisée avec deux paramètres qui représentent la graduation horizontale et la graduation verticale.

On peut aussi utiliser AXIS sans paramètre ceci se traduit sur l'écran par le tracé d'une ligne horizontale et une ligne verticale qui se coupent au point $X=0$ et $Y=0$ dans l'espace de données ce point se trouve à l'intérieur du domaine des données sinon les coordonnées de l'intersection des axes seront la valeur minimale x et la valeur minimale y des données.

I.4)

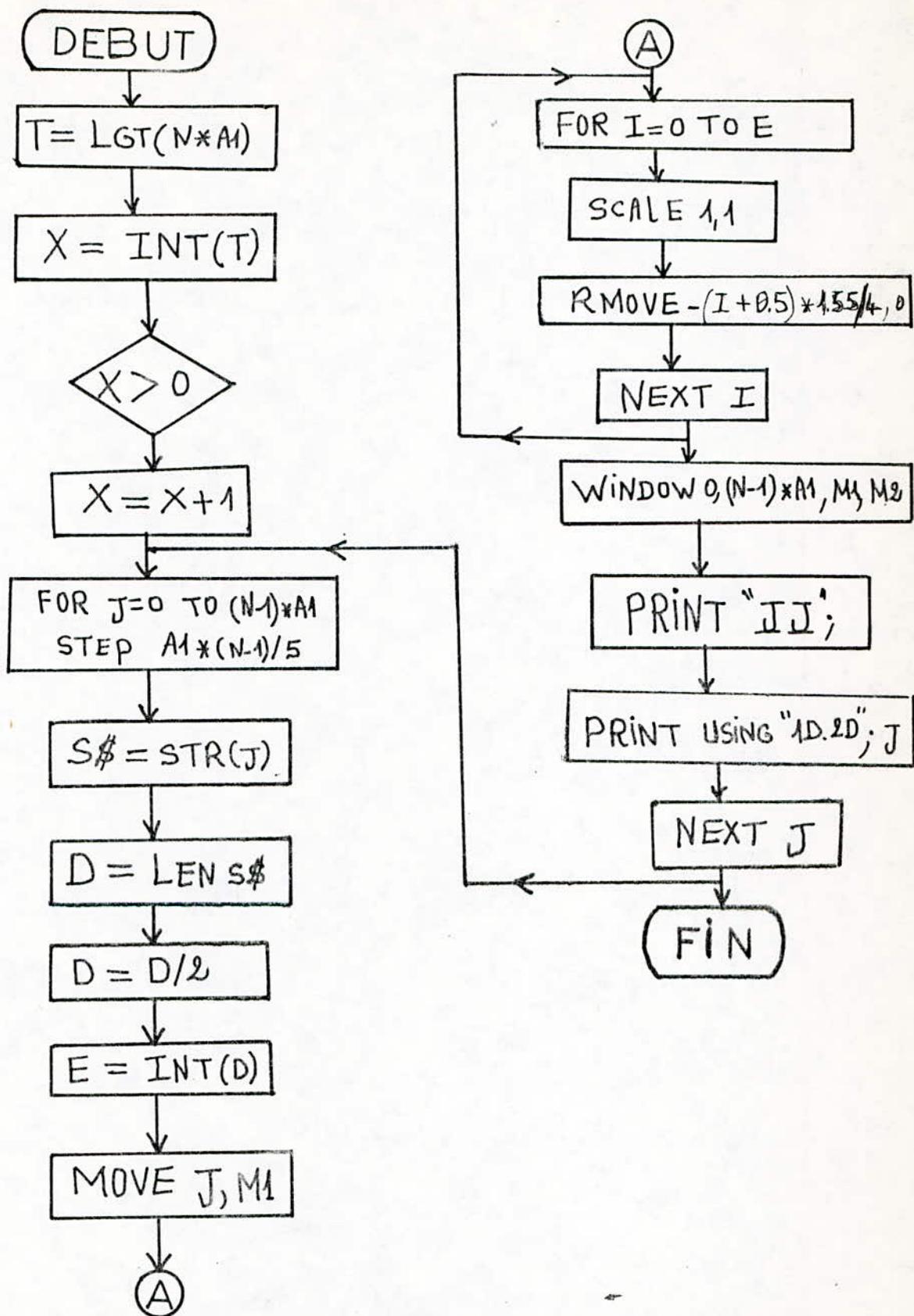
ECRITURE DE L'AXE HORIZONTAL:

PROBLEME POSE: Representation des graduations sur les axes sous le format $\square \square . \square \square \square$ (Un chiffre avant la virgule et deux chiffres ^{signe} après la virgule)

SOLUTION ADOPTEE: (voir organigramme)

En utilisant seulement les instructions MOVE, FORTO, NEXT et l'instruction PRINT, il est impossible de représenter correctement les graduations sur les axes, on remarque que lorsque le nombre de points N augmente ou lorsque la période d'échantillonnage augmente les graduations représentées sur l'axe horizontal se chevauchent et se mélangent entre elles.

Pour résoudre ce problème il a fallu introduire d'autres instructions telles que l'instruction LGT qui permet de calculer le logarithme décimale d'une valeur.



ORGANIGRAMME DE L'ÉCRITURE DES
L'AXE HORIZONTAL

FORME SYNTAXE:

N° DE LIGNE	VARIABLE= LGT (EXPRESSION NUMERIQUE)
-------------	--------------------------------------

L'espace réservé pour l'écriture des graduations est limité ce qui empêche la représentation correcte des graduations comportant un nombre important de chiffres; Pour cela on passe d'une représentation linéaire à une représentation logarithmique en calculant le logarithme décimal des graduations:

EXEMPLE 1:

$$A=0.000032$$

$$T=LGT(A)$$

$$T=LGT(0.000032)=LGT(32 \cdot 10^{-6})=LGT32-6=-4.5.$$

$$T=-4.5.$$

On utilise ensuite l'instruction INT pour calculer la partie entière de l'expression numérique (T)

FORME SYNTAXE:

N° DE LIGNE	VARIABLE = INT (expression Numérique)
-------------	---------------------------------------

EXEMPLE :

$$X=INT (4.5)$$

RESULTAT

$$X=4$$

EXEMPLE 2:

X = INT(-4.5)

RESULTAT : X = -5

On divise chaque graduation par le terme 10^x ce qui entraîne sa réduction d'où la possibilité d'éviter le problème de chevauchement entre les graduations, le terme 10^x sera ensuite imprimé comme facteur multiplicatif des unités représentées.

On remarque que les valeurs portées sur l'axe horizontal sont décalées par rapport aux graduations qui leur correspondent.

Pour les centrer on compte le nombre de caractères constituant chaque chaîne et on positionne la graduation juste au milieu de la longueur de la chaîne de caractères.

Pour compter le nombre de chiffres constituant chaque nombre, on convertit la valeur numérique correspondante en chaîne de caractère à l'aide de l'instruction STR.

FORME SYNTAXE:

N° DE LIGNE	VARIABLE = STR (EXPRESSION NUMERIQUE)
-------------	---------------------------------------

100	INIT
-----	------

110	J=128
-----	-------

120	S S =STR (J)
-----	-------------------------

130	END
-----	-----

Le résultat de ces lignes est que $S\#$ contient maintenant la chaîne de caractères "128" et ceci est équivalent à $S\# = "128"$ on utilise ensuite l'instruction LEN qui permet de compter le nombre de caractères existant dans la chaîne de caractères et l'affecte à la lettre D.

FORME SYNTAXE:

N° DE LIGNE VARIABLE =LEN (CHAINE DE CARACTERES)

EXEMPLE: 100 $S\# = "128"$

110 D = LEN ($S\#$)

Le résultat de cet exemple donne à D la valeur 3.

Dans le cas où D est paire, on considère la partie entière de $D/2$ et on positionne le curseur juste au niveau de la graduation correspondante, et dans le cas où D est impair, on considère la partie entière de $(D-1)/2$ et on positionne le curseur au niveau du chiffre qui se trouve au milieu de la valeur considérée et qui doit se trouver juste au niveau de la graduation. (FIG. I.4)

-MOVE est l'instruction qui permet de déplacer le curseur sans tracer de ligne.

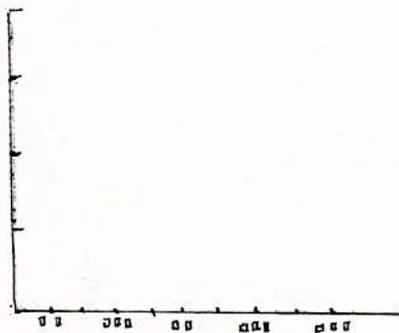


FIG. I.4

FORME SYNTAXE:

N° DE LIGNE MOVE EXPRESSION NUMERIQUE, EXPRESSION NUMERIQUE

La première et la deuxième expression numérique représentent respectivement la valeur horizontale et la valeur verticale dans l'espace de données.

MOVE utilise les emplacements "ABSOLUS", qui sont définis par les coordonnées horizontales et verticales correspondant à un intervalle le long de chaque axe.

L'instruction RMOVE permet de déplacer le curseur à un lieu qui est relatif par rapport aux emplacements actuels.

FORME SYNTAXE:

N° DE LIGNE RMOVE EXPRESSION NUMERIQUE, EXPRESSION NUMERIQUE

La première et la seconde expression numérique indiquent les distances relatives horizontales et verticales.

RMOVE est utilisé avec des emplacements "RELATIFS".

Les emplacements relatifs sont définis par un intervalle horizontal et un intervalle vertical à partir de l'endroit où se trouve le curseur au lieu d'un intervalle le long de chaque axe. (dans l'espace de données).

L'écran de visualisation est divisé en 130 unités GPU suivant l'horizontale et 100 unités suivant la verticale et chaque unité a comme dimension 1.55 suivant l'horizontale et 1.88 suivant la verticale.



Les instructions MOVE et RMOVE utilisent des valeurs dans l'espace de données, par contre dans ce sous-programme de tracé graphique RMOVE doit être utilisée avec des unités GDU où chaque unité G.D.U doit être équivalente à une unité dans l'espace de données, pour cela on utilise le facteur de proportionnalité SCALE:

FORME SYNTAXE:

N° DE LIGNE	SCALE	EXPRESSION NUMERIQUE,	EXPRESSION NUMERIQUE
-------------	-------	-----------------------	----------------------

La première expression numérique détermine le facteur de proportionnalité horizontale, la seconde expression numérique détermine le facteur de proportionnalité verticale. Le terme "SCALE FACTOR" (facteur de proportionnalité) Nous donne le rapport de proportionnalité entre les unités dans l'espace de données et les unités de représentation graphique (G.D.U).

REMARQUE:

SCALE peut être utilisée avec différents arguments:

SCALE 1,1 , SCALE 2,2 , SCALE 100,100.

Avant d'écrire les graduations sur l'axe horizontal on déplace le curseur au dessous de l'axe de deux lignes et ceci grâce à l'instruction PRINT "J".

J: est un paramètre, lorsqu'on l'utilise avec l'instruction PRINT, il permet de déplacer le curseur vers le bas d'une ligne ou de plusieurs lignes suivant le nombre de J existant dans l'instruction PRINT.

Pour l'écriture des graduations selon le format imposé:
Un chiffre avant la virgule et deux chiffres après la
virgule ($\square.\square\square$), on utilise l'instruction PRINT USING
dont la forme syntaxique est:

```
N° DE LIGNE PRINT USING FORMAT:EXPRESSION NUMERIQUE
```

Les éléments qui se trouvent dans le format sont appelés:
Opérateurs de champ de représentation et l'expression
numérique est la valeur à imprimer.

Il existe différents opérateurs de champ de représenta-
tion:

D,E,A,L,...

L'instruction utilisée est la suivante:

```
N° DE LIGNE PRINT USING 1D.2D : J.
```

-L'opérateur D accepte "n" valeurs qui peuvent varier
entre 1 et 255. et chaque valeur représente le nombre de
positions que doit occuper l'expression numérique.

-Les nombres utilisés avec l'opérateur D doivent être
entiers car on a aucune information sur l'endroit où
peut être placé le point décimal.

Pour pouvoir représenter toutes les graduations sur l'axe horizontal on doit répéter une portion du programme un certain nombre de fois ou jusqu'à ce qu'une condition soit satisfaite pour cela on utilise plusieurs boucles. Dans le cas où le nombre d'itérations est connu d'avance, la boucle est plus facilement programmée au moyen de l'instruction FOR.TO et NEXT.

FORME SYNTAXE:

N° DE LIGNE	FOR	VARIABLE NUMERIQUE	TO	VARIABLE NUMERIQUE
-------------	-----	--------------------	----	--------------------

L'instruction FOR.TO comporte une variable numérique appelée variable courante ou variable de contrôle, dont la valeur change à chaque exécution de la boucle, cette variable doit être une variable ordinaire, non indicée. Le nombre d'exécution s'obtient en précisant une valeur initiale et une valeur finale pour la variable de contrôle.

Si la boucle commence par une instruction FOR TO, elle doit impérativement se terminer par une instruction NEXT.

FORME SYNTAXE:

N° DE LIGNE	NEXT	VARIABLE NUMERIQUE
-------------	------	--------------------

La boucle est donc comprise entre l'instruction FOR TO et l'instruction NEXT.

La variable de contrôle qui est suivie du mot clé NEXT doit avoir le même nom que celui qui a été donné dans l'instruction FOR TO correspondante.

EXEMPLE:

```
N° DE LIGNE   FOR J=0 TO (N-1) * A1
-----
-----
-----
N° DE LIGNE   NEXT J.
```

à l'exécution de la boucle, J se verra attribuer la valeur initiale zéro et cette valeur augmentera d'une unité à chaque exécution de la boucle, jusqu'à la dernière exécution pour laquelle J prendra la valeur finale $(N-1) \times A1$. L'exécution de la boucle s'achève dès que J dépasse la valeur finale.

Par défaut, la variable de contrôle augmente d'une unité à chaque fois, mais on peut lui donner un autre rythme de progression au moyen du mot STEP (qui signifie pas), incluse dans l'instruction FOR-TO, comme dans l'exemple suivant:

```
N° DE LIGNE   FOR J=0 TO (N-1) * A1 STEP A1 * (N-1) / 5
-----
-----
-----
N° DE LIGNE   NEXT J.
```

J prend la valeur zéro lors de la première exécution, $A1 \times (N-1) / 5$ lors de la seconde, $2A1 \times (N-1) / 5$ lors de la troisième, jusqu'à prendre la valeur $(N-1) \times A1$.

REMARQUE: Il n'est pas obligatoire que la variable de contrôle soit entière et positive, elle peut prendre des valeurs négatives ou décimales.

On peut la faire décroître au lieu de croître à chaque exécution de la boucle, en donnant une valeur négative au pas spécifié par STEP.

En temps normal, les instructions d'un programme sont exécutées dans l'ordre où elles apparaissent; l'une après l'autre. Cependant il est quelquefois nécessaire de sauter à une autre instruction du programme altérant l'ordre normal d'exécution, on y parvient au moyen de l'instruction GOTO.

FORME SYNTAXE:

N° DE LIGNE	GOTO	N° DE LIGNE
-------------	------	-------------

Cette instruction permet des sauts ou branchements inconditionnels, c'est à dire des sauts ou transfert de contrôle d'un point du programme à un autre point du programme.

Pour effectuer les branchements conditionnels, on utilise l'instruction IF-THEN.

FORME SYNTAXE:

N° DE LIGNE	IF CONDITION	THEN	N° DE LIGNE
-------------	--------------	------	-------------

L'instruction IF-THEN comporte un numéro de ligne d'une autre instruction, lors de l'exécution, le contrôle sera transféré à cette autre instruction si la condition est vérifiée, si elle ne l'est pas c'est l'instruction qui suit l'instruction IF-THEN qui sera exécutée.

EXEMPLE:

```

1500 IF B>0 THEN 1520.
1510 B =B+1
1520 FOR I=M1 TO M2 STEP (M2-M1)/5.

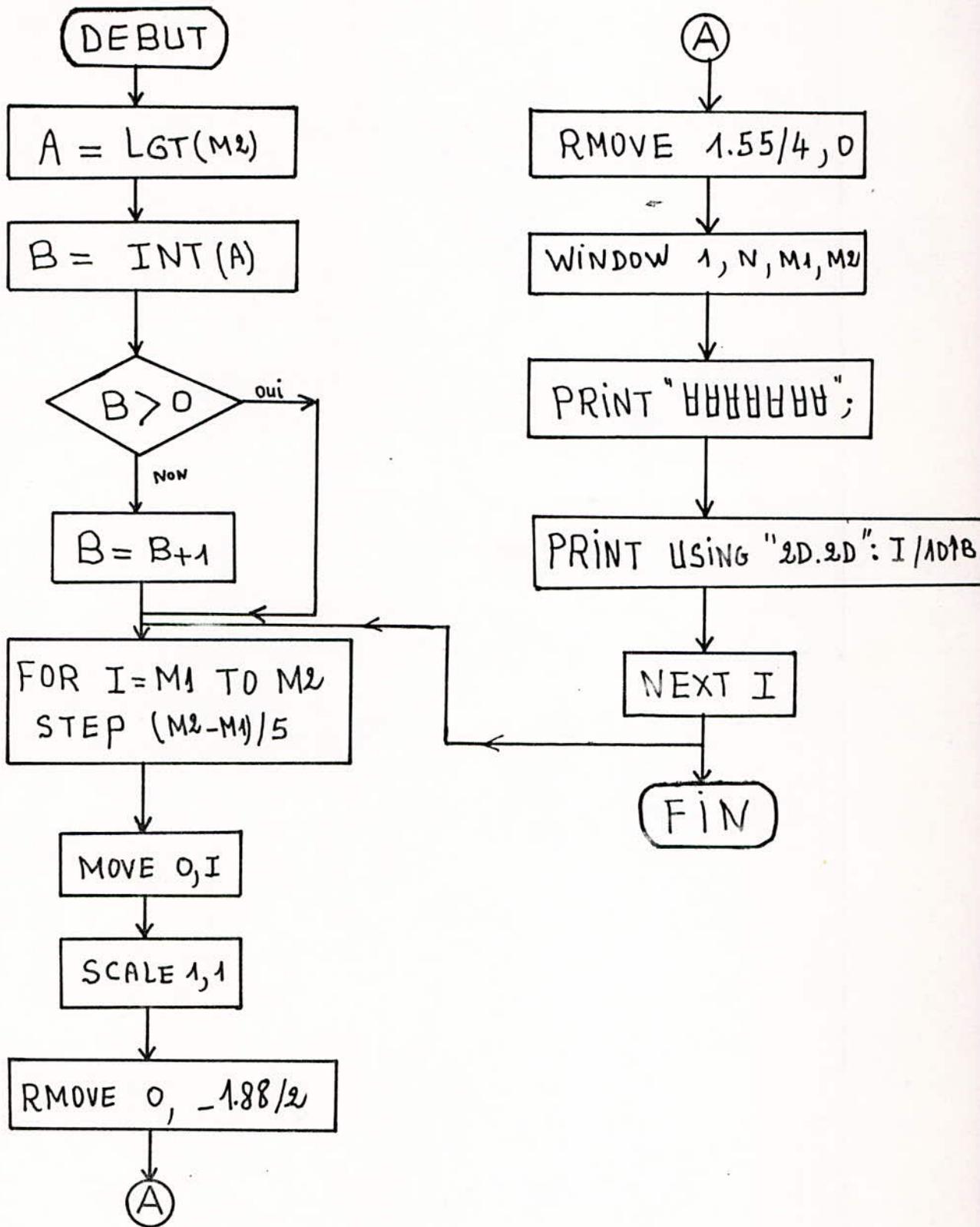
```

à la ligne 1500 du sous-programme on a la condition $B > 0$ si cette condition est vérifiée, le contrôle sera transféré à la ligne 1520, sinon (cas ou $B \leq 0$) le contrôle est transféré à la ligne 1510.

I.5) ECRITURE DE L'AXE VERTICAL: (voir organigramme).

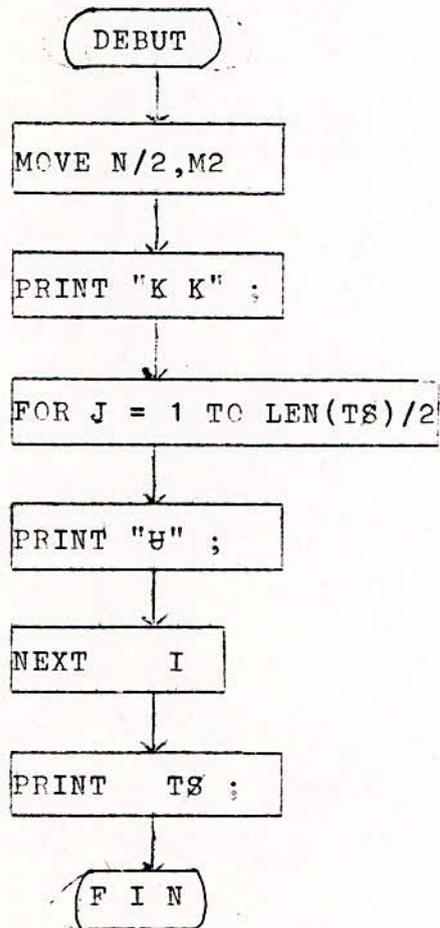
On utilise le même procédé de calcul que dans le cas précédent car les problèmes posés au niveau de l'écriture de l'axe vertical sont les mêmes que dans le cas de l'écriture de l'axe horizontal mais au lieu d'utiliser l'instruction PRINT "J"; (qui permet le déplacement du curseur d'une ligne vers le bas), on utilise l'instruction PRINT "U"; qui permet le déplacement du curseur en arrière (à gauche) d'une position. Dans notre cas on a utilisé un décalage en arrière de 7 positions:

```
PRINT " U U U U U U U ";.
```



ORGANIGRAMME DE L'ECRITURE
DES L'AXES VERTICALIX

I.6 ECRITURE DU TITRE:

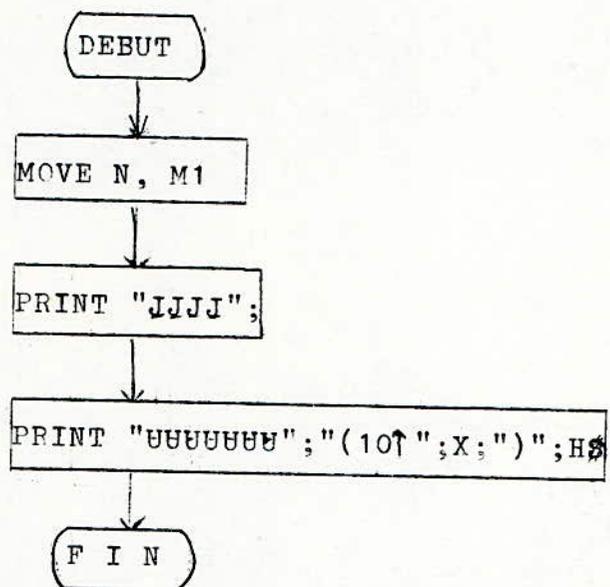


Pour centrer le titre $T\cancel{S}$ on positionne le curseur sur le point de coordonnée $(N/2, M2)$ on le décale ensuite de deux lignes vers le haut grâce à l'instruction `PRINT "K K";`

La boucle (FOR TO-NEXT) est utilisée pour décaler le curseur à chaque fois d'une position vers l'arrière, le nombre d'exécution de cette boucle est égale à la moitié du nombre de lettre existant dans le titre.

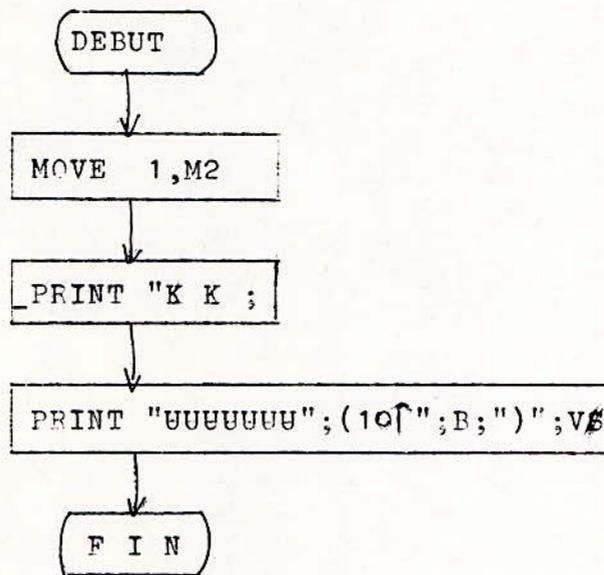
22

I.7 ECRITURE DE L'UNITE ET DU FACTEUR MULTIPLICATIF
SUR L'AXE HORIZONTAL



On positionne le curseur sur le point de coordonnée (N,M1) on le décale ensuite de quatres lignes vers le bas (PRINT "J J J J";), l'exécution de l'instruction PRINT "UUUUUUUU"; "(10↑"; X; ")"; H\$ nous donne un décalage du curseur de 7 positions vers l'arriére et imprime l'expression (10)(H\$).

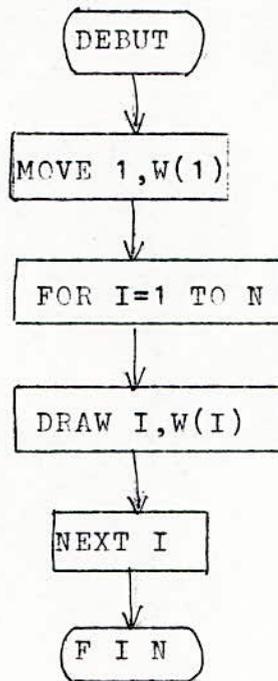
I.8 ECRITURE DE L'UNITE ET DU FACTEUR MULTIPLICATIF
 SUR L'AXE VERTICAL



On positionne le curseur sur le point de coordonnée (1,M2) on le décale ensuite de deux lignes vers le haut (PRINT "K K";).

L'exécution de l'instruction PRINT "UUUUUUU";
 "(10⁷";B;");V nous donne un décalage de curseur de
 7 positions vers l'arrière et imprime (10⁷)(V).

I.9 TRACE GRAPHIQUE



On utilise l'instruction DRAW pour tracer une ligne visible entre la dernière position du curseur et les coordonnées spécifiées par les expressions numériques.

FORME SYNTAXE:

N° DE LIGNE	DRAW	LEXPRESSION NUMERIQUE,	LEXPRESSION NUMERIQUE
-------------	------	------------------------	-----------------------

La première et la deuxième expression numérique représentent respectivement la valeur horizontale et verticale dans l'espace de données.

DRAW utilise les emplacements "absolus" (voir instruction MOVE)

Pour tracer le graphe, on positionne d'abord le curseur au point de coordonnée (1, W(1) ou la fonction échantillonnée que l'on veut représenter.

L'utilisation d'une boucle FOR...TO-NEXT permet de relier tous les points I, W(I) par un segment de droite réalisant ainsi une interpolation linéaire entre deux échantillons successifs.

CHAPITRE U

PROCEDURE D'UTILISATION DU SOUS-PROGRAMME DE

TRACE GRAPHIQUE ET APPLICATION.

Pour utiliser le sous-programme de tracé graphique on considère deux méthodes:

1ere METHODE:

Dans le programme principal on définit le nombre de points d'échantillonnage N , la période d'échantillonnage A_1 , ainsi que la définition de la fonction du titre ($T\#$), des unités horizontales ($H\#$) et des unités verticales ($V\#$). Pour passer du programme principal au sous-programme de tracé graphique on utilise l'instruction GOSUB.

FORME SYNTAXE:

N° DE LIGNE	GOSUB	N° DE LA PREMIERE INSTRUCTION DU S-PROGRAMME
-------------	-------	----------------------------------------------

L'exécution de cette instruction transfère le contrôle au sous-programme.

Le retour se fera à l'instruction qui suit le GOSUB quand une instruction RETURN sera exécutée dans le sous-programme, on peut représenter ceci par l'exemple suivant:

```
100 REM ** Programme Principal.
110 PRINT "Nombre de points:";
120 INPUT N
130 PRINT "Periode d'échantillonnage:";
140 INPUT A1
150 REM **.Définition de la fonction.
```

```
. -----
. -----
. -----
```

```

200 PRINT "TITRE:";
210 INPUT T$
270 PRINT "UNITE HORIZONTAL:";
280 INPUT H$
290 PRINT "UNITE VERTICALE:";
300 INPUT V$
310 REM ** APPEL DU SOUS-PROGRAMME DE TRACE GRAPHIQUE
320 GOSUB 500
330 -----
. -----
. -----
. -----
500 REM ** SOUS-PROGRAMME DE TRACE GRAPHIQUE
.
.
.
1000 RETURN

```

Dans le programme principal on remarque l'utilisation des instructions:PRINT,INPUT,DELETE et DIM.

1°) INSTRUCTION PRINT:

FORME SYNTAXE:

N° DE LIGNE PRINT C:A,B,...

C:Représente l'adresse du périphérique où l'on veut sortir les valeurs numériques A,B,...codées en ASCII. En cas de défaut d'adresse,l'instruction PRINT sert à transmettre les données numériques ou les chaînes au terminal,le mot clé PRINT est suivi d'une liste d'éléments en sortie.

Ces éléments peuvent être des nombres, des formules, ou des chaînes, les éléments successifs doivent être séparés par des virgules ou des points virgules.

2°) INSTRUCTION INPUT:

FORME SYNTAXE:

```
N° DE LIÈNE INPUT @C: A,B,...
```

C: Représente l'adresse du périphérique ou l'on veut entrer les variables codées en ASCII.

Par défaut de C, les données seront entrées par le clavier.

L'Instruction INPUT sert à introduire les données numériques ou caractères au cours de l'exécution d'un programme.

Le mot clé INPUT est suivi d'une liste de variables:

La liste peut comprendre à la fois des variables numériques et des variables de chaînes.

3°) INSTRUCTION DIM

FORME SYNTAXE

N° DE LIGNE DIM TABLEAU, TABLEAU, ...

Le mot clé DIM est suivi d'un ou de plusieurs noms de tableaux séparés par des virgules. Chaque nom de tableau doit être suivi, entre parenthèses, d'un ou deux nombres (un pour une liste, deux pour une table).

Chaque entier donne la valeur maximale que peut prendre l'indice correspondant du tableau.

L'instruction DIM a pour but de réserver en mémoire les emplacements nécessaires aux différentes valeurs des variables indicées,

Avant l'exécution de l'instruction DIM on utilise l'instruction DELETE.

FORME SYNTAXE:

N° DE LIGNE DEL N° DE LIGNE.

Cette instruction sert à effacer une ou plusieurs lignes du programme.

EXEMPLE:1 N° DE LIGNE DEL 800 .

RESULTAT: EFFACEMENT DE LA LIGNE 800.

EXEMPLE.2 N° DE LIGNE DEL 800, 900.

RESULTAT: Effacement de la ligne 800 et la ligne 900 et de toutes les autres lignes qui se trouvent entre 800 et 900.

2° METHODE:

On divise le programme en plusieurs parties et chaque partie représente un sous-programme.

à l'aide des touches "utilisateur" se trouvant sur le clavier de calculateur on fait correspondre à chaque touche un sous-programme.

Exemples:

Déf de la fonction

SINUS	CALCUL DE LA FFT	CALCUL DE LA TFD	DEF \int	TRACE GRAPHIQUE
1	2	3	4	5

EXEMPLE 1) Si on veut le tracé graphique de la fonction Sinus, on utilise d'abord la première touche pour générer la fonction, ensuite la cinquième touche pour avoir le tracé graphique de la fonction Sinus.

EXEMPLE 2) Tracé graphique de la FFT de la fonction projectrice:

On utilise la quatrième-touche pour générer la fonction (\int)

ensuite la deuxième touche pour le calcul de la FFT.

et pour la représentation du graphe de la FFT on utilise la cinquième touche.

REMARQUE:1) La deuxième méthode a été adoptée pour le tracé graphique des fonctions.

REMARQUE 2):(N° de touche utilisateur)X4=N° de ligne d'appel du Sous-programme.

EXEMPLE: 4. GOTO $\boxed{100}$ est équivalent à RUN $\boxed{100}$

8. GOTO $\boxed{310}$ est équivalent à RUN $\boxed{310}$

II.1 FONCTION "PROJECTRICE"

II.1.1 DEFINITION DE LA FONCTION

$$f(t) = \begin{cases} 0 & \text{si } t_D + t_W < t < t_D \\ 1 & \text{si } t_D \leq t \leq t_D + t_W \end{cases}$$

II.1.2 SOUS-PROGRAMME DE DEFINITION DE LA FONCTION "PROJECTRICE"

```

470 REM ** Définition de l'impulsion
480 PRINT "Nombre de points N=";
490 INPUT N
500 REM ** Période d'échantillonnage
510 A1=25 * 10↑3
520 DELETE U
530 DIM U(N)
540 U= 0
550 F= 200
560 FOR I=F TO N
570 U(I)=12.5*10↑-6
580 NEXT I
590 DELETE Y
600 DIM Y(N)
610 Y = 0
620 G = 300
630 FOR I=G TO N
640 Y(I) =-12.5*10↑-6
650 NEXT I
660 DELECTE W
670 DIM W(N)
680 W=U+Y
690 REM ** TITRE DU GRAPHE
!700 TS ="Fonction Projectrice"

La fonction projectrice (W) est représentée comme
étant la somme de deux autres fonctions V et Y.

```

Le choix de F et G permet de définir la largeur de l'impulsion

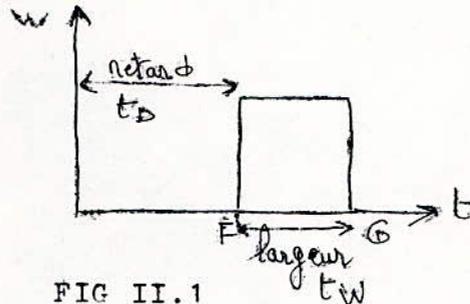


FIG II.1

II.2. FONCTION SINUS:

II.2.1 DEFINITION DE LA FONCTION:

$$f(t) = A_2 \cos \frac{2\pi}{T} NA_1.$$

Soit par exemple: N=32 Points

$$A_2 = 1$$

Nombre de periodes: n=4 périodes.

D'ou $f(t) = \cos \pi t / 4$

II.2.2 SOUS-PROGRAMME DE DEFINITION DE LA FONCTION SINUS.

```

90 REM ** Définition de la fonction
100 REM ** Construction d'une SINUSOIDE
110 Def FNA(I)=A2XCOS(PI/4XI)
120 A2=10^-6
130 FOR I=0 TO N-1
140 J=I+1
150 W(J)=FNA(I)
160 NEXT I
170 TS=" "
180 HS="SECONDE"
190 US="VOLTS"
200 RETURN

```

II.3 APPLICATION DU SOUS-PROGRAMME DE TRACE GRAPHIQUE

SUR T.F

Grâce au sous-programme de tracé graphique on peut étudier et commenter de façon pratique les propriétés de la T.F., ainsi que la représentation graphique de la FFT et pour mieux utiliser les techniques de la FFT, il est nécessaire d'être familiarisé avec la théorie de Fourier non seulement dans le contexte classique mais également dans le contexte numérique.

Pour ceci nous allons commencer par une introduction à la théorie de Fourier classique (Chapitre : III, IV , V). Ensuite nous verrons comment la transformée de Fourier discrète (DFT) et la FFT sont des outils qui permettent une application rapide et facile de la théorie de Fourier. (Chapitre: VI, VII, VIII.)

CHAPITRE III

TRANSFORMATION DE FOURIER

Un Signal quelconque a deux modes de représentations complémentaires:

La représentation dans le domaine temporel et la représentation dans le domaine des fréquences, ces deux représentations sont reliées entre elles par la transformation de Fourier.

La décomposition en fréquence d'un Signal périodique pourra être effectuée par les séries de Fourier, tandis que pour un Signal non périodique, cette décomposition pourra être effectuée par l'intégrale de Fourier.

III.1 Series de Fourier:

Soit $X(t)$ une fonction périodique, de période T .

$X(t)$ peut s'écrire:

$$X(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_n \cos \frac{2\pi n t}{T} + b_n \sin \frac{2\pi n t}{T} \right)$$

$$\text{Avec } f_0 = \frac{1}{T}$$

a_0 : est le terme continu et représente la valeur moyenne du Signal sur une période.

$$a_0 = \int_0^T X(t) dt$$

Les autres coefficients a_n et b_n sont évalués pour tout n par:

$$a_n = \int_{-T/2}^{T/2} X(t) \cos 2\pi f_0 n t dt$$

$$b_n = \int_{-T/2}^{T/2} X(t) \sin 2\pi f_0 n t dt$$

Si l'on pose : $X(nf_0) = \frac{1}{T} \int_{-T/2}^{T/2} (a\tau - jbn) dt$

On a : $X(nf_0) = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{-2\pi jnf_0 t} dt$

$X(nf_0)$ est le Spectre de fréquence, grandeur en general complexe, qui peut se décomposer en:

Spectre d'amplitudes:

$$|X(nf_0)| = \frac{1}{2} \sqrt{a^2 + b^2}$$

Spectre de phases :

$$\varphi(nf_0) = \text{Arc tg} \left(-\frac{bn}{an} \right)$$

Réciproquement , on aura :

$$x(f) = \sum_{n=-\infty}^{\infty} |X(nf_0)| e^{j\varphi(nf_0)}$$

$$x(t) = \sum_{n=-\infty}^{\infty} e^{2\pi jnf_0 t} \frac{1}{T} \int_{-T/2}^{T/2} x(\tau) \cdot e^{-2\pi jnf_0 \tau} d\tau.$$

Ces équations nous montrent que le Spectre d'une fonction périodique de période T est composé de raies dont l'écart minimum est, sur l'axe des fréquences: $f_0 = \frac{1}{T}$

Le Spectre d'une fonction périodique est donc essentiellement discontinu, il n'existe que pour les valeurs de la fréquence multiples de f_0 .

III.2) INTEGRALE DE FOURIER:

On applique l'intégrale de Fourier à une fonction lorsque celle-ci n'est pas périodique.

On peut considérer cette non-périodicité comme résultant d'une extension à l'infini de la période T . L'intervalle

de fréquence $f_0 = \frac{1}{T}$ tend alors vers zéro et le Spectre devient alors une fonction qui peut être continue.

on a :

$$x(t) = \int_{-\infty}^{\infty} e^{2\pi jft} df \int_{-\infty}^{\infty} x(\kappa) e^{-2\pi jf\kappa} d\kappa$$

$$X(f) = \int_{-\infty}^{\infty} x(t) \cdot e^{-2\pi jft} dt.$$

$X(f)$ est appelé la transformée de Fourier de $x(t)$ on écrit habituellement :

$$x(t) \xrightarrow{\quad} X(f)$$

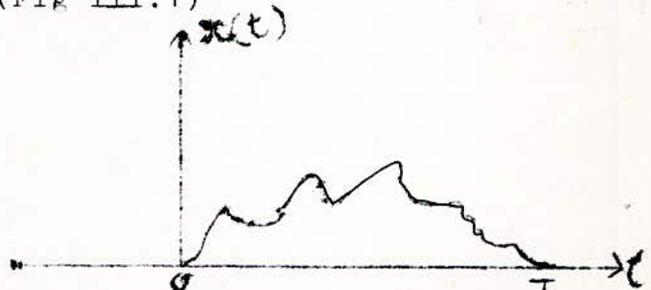
Réciproquement, connaissant $X(f)$, on peut obtenir $x(t)$ par :

$$x(t) = \int_{-\infty}^{\infty} X(f) \cdot e^{j2\pi ft} df.$$

III.3) CAS DES FONCTIONS PHYSIQUES :

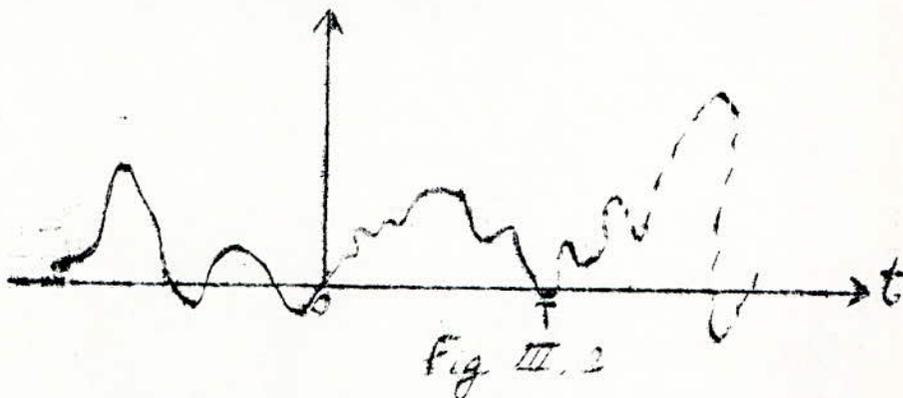
Lorsqu'on considère une fonction physique, donc connue sur un intervalle borné $(0, T)$, deux cas peuvent se présenter :

a) La fonction est nulle en dehors de cet intervalle, donc connaître la fonction dans l'intervalle $(0, T)$ donne sa connaissance de $-\infty$ à $+\infty$ (Fig III.1)



On peut donc espérer connaître le Spectre de cette fonction avec une finesse aussi grande que l'on veut.

b) La fonction est connue dans l'intervalle $(0, T)$ mais on ignore ce qu'elle peut être en dehors de l'intervalle de connaissance. (Fig III.2)



à ce cas se rattache celui où la fonction ne peut pas être décrite par une équation mathématique dans ce cas les techniques de Fourier classiques ne peuvent être utilisées.

Mais si cette fonction peut être échantillonnée et numérisée, alors la transformée de Fourier discrète peut être utilisée à l'aide de l'ordinateur. (c'est ce que nous allons détailler dans les chapitres suivants).

III.4 CONDITIONS D'EXISTENCE DE LA TRANSFORMATION DE FOURIER

Pour qu'une fonction $f(t)$ ait une transformée de Fourier, il faudrait qu'elle vérifie les conditions suivantes:

1°) La fonction $f(t)$ doit être bornée (pas de valeurs infinies).

2°) L'Intégrale de $f(t)$ entre $-\infty$ et $+\infty$ ait une valeur finie.

3°) Les discontinuités de $f(t)$ ainsi que ses maxima et minima soient en nombre fini.

On voit que toutes les fonctions existant physiquement et que l'on peut être amené à traiter, vérifient ces conditions, essentiellement parce qu'on les observe sur un temps fini.

III.5) QUELQUES PROPRIETES DE LA TRANSFORMEE DE FOURIER

III.5.1)-Propriété de linéarité: La transformation de Fourier est une opération linéaire.

$$f(t) \xrightarrow{\quad} F(f)$$

$$g(t) \xrightarrow{\quad} G(f)$$

entraîne $af(t)+bg(t) \xrightarrow{\quad} aF(f)+bG(f).$

III.5.2)-Propriété de parité:

Si $f(t)$ est une fonction réelle quelconque, alors la partie réelle de sa transformée de Fourier est paire et sa partie imaginaire est impaire.

$$f(t) \xrightarrow{\quad} F(f) \begin{cases} \text{Partie réelle paire.} \\ \text{Partie imaginaire impaire.} \end{cases}$$

III.5.3)-Propriété de Similitude:

Le changement d'échelle des temps provoque dans le domaine fréquence un changement d'amplitude et de fréquence.

$$f(t) \xrightarrow{\quad} F(f)$$

alors $f(at) \xrightarrow{\quad} \frac{1}{|a|} F\left(\frac{f}{a}\right)$

(voir page 1.54)

III.5.4)-Propriété de Translation:

$$f(t) \xrightarrow{\quad} F(f)$$

entraîne $f(t-a) \xrightarrow{\quad} e^{-2\pi jaf} F(f)$

Ceci implique que les transformées de Fourier de $f(t)$ et $f(t-a)$ ont même module mais la transformée de Fourier de $f(t-a)$ subit une rotation de phase supplémentaire de $2\pi af$.

III.6) THEOREME DE PLANCHEREL:

Ce théorème établit que la transformée de Fourier d'un produit de convolution est un produit simple et réciproquement:

$$\begin{aligned} f(t) * g(t) &\xrightarrow{\hspace{2cm}} F(f) \cdot G(f) \\ f(t) \cdot g(t) &\xrightarrow{\hspace{2cm}} F(f) * G(f). \end{aligned}$$

III.7) FORMULE DE POISSON:

$$\begin{aligned} f(t) &\xrightarrow{\hspace{2cm}} F(f) \\ \sum_{K=-\infty}^{+\infty} f(K) &= \sum_{n=-\infty}^{+\infty} F(n) \end{aligned}$$

C H A P I T R E IV)

F I L T R A G E T E M P O R E L (OU FENETRE TEMPOREL)

On définit le filtrage temporel comme étant l'opération d'interruption ou d'atténuation d'un signal.

La figure (Figure IV.1) montre un signal temporel $f(t)$ (fig a) et un filtre temporel $h(t)$ (fig b) qui coupe le signal dans l'intervalle $0, T$.

Le résultat de ce filtrage est le produit $f(t).h(t)$. (fig c) donc filtrer $f(t)$ par $h(t)$ c'est donc réaliser le produit $f(t).h(t)$.

En appliquant le théorème de Plancherel:

$$f(t).h(t) \begin{matrix} \longrightarrow \\ \longleftarrow \end{matrix} F(f) \times H(f)$$

On remarque qu'un tel filtrage affecte le spectre $F(f)$ (transformée de Fourier) d'un signal $f(t)$.
Donc, tout filtrage temporel affecte le Spectre du signal filtré.

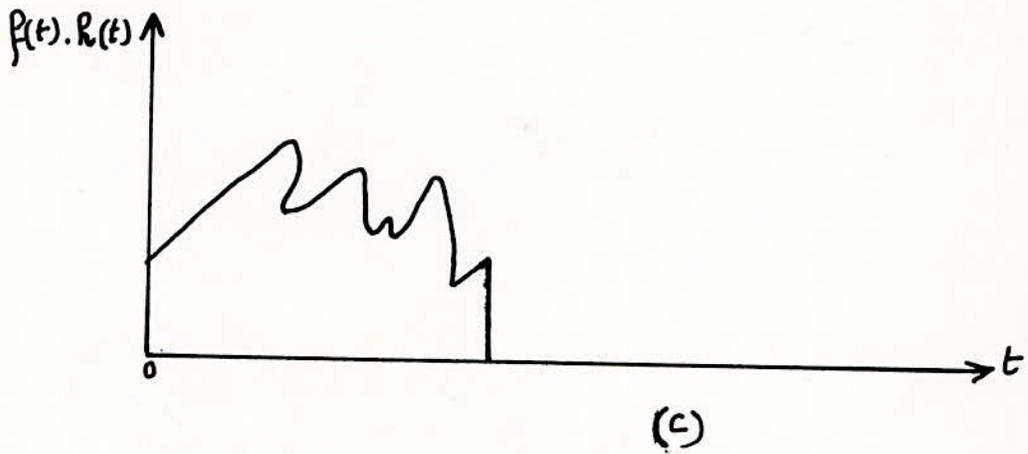
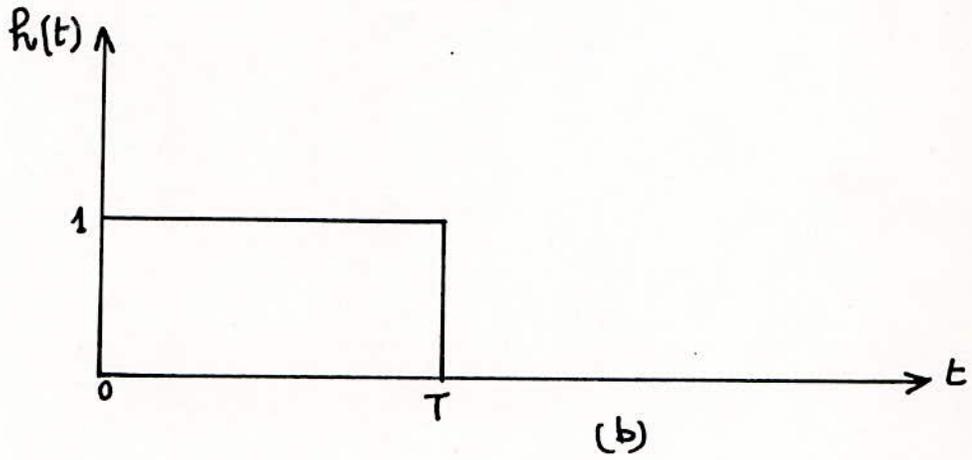
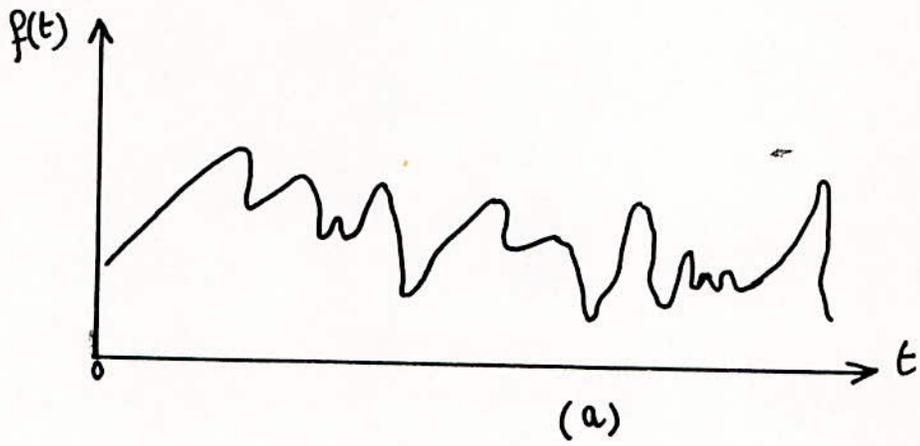


FIG IV.1

C H A P I T R E \mathbf{V}

E C H A N T I L L O N N A G E

V.1) GENERALITES:

Les Signaux que l'on vient d'étudier et qui sont en général analogiques et continus dans le temps doivent être échantillonnés, c'est à dire que ces signaux ne doivent pas être observés d'une manière continue, mais à certains instants seulement ces observations sont périodiques et séparées par Δt , on dit que l'on échantillonne à une fréquence $F_e = \frac{1}{\Delta t}$. (voir fig V.1)

V.2) THEOREME D'ECHANTILLONNAGE:

Echantillonner $x(t)$ à la fréquence F_e revient à multiplier $x(t)$ par une suite d'impulsions de fréquences de récurrence F_e , cette suite d'impulsions s'écrit sous la forme:

$$p_{F_e}(t) = \Delta t \sum_{K=-\infty}^{+\infty} \delta\left(t - \frac{K}{F_e}\right)$$

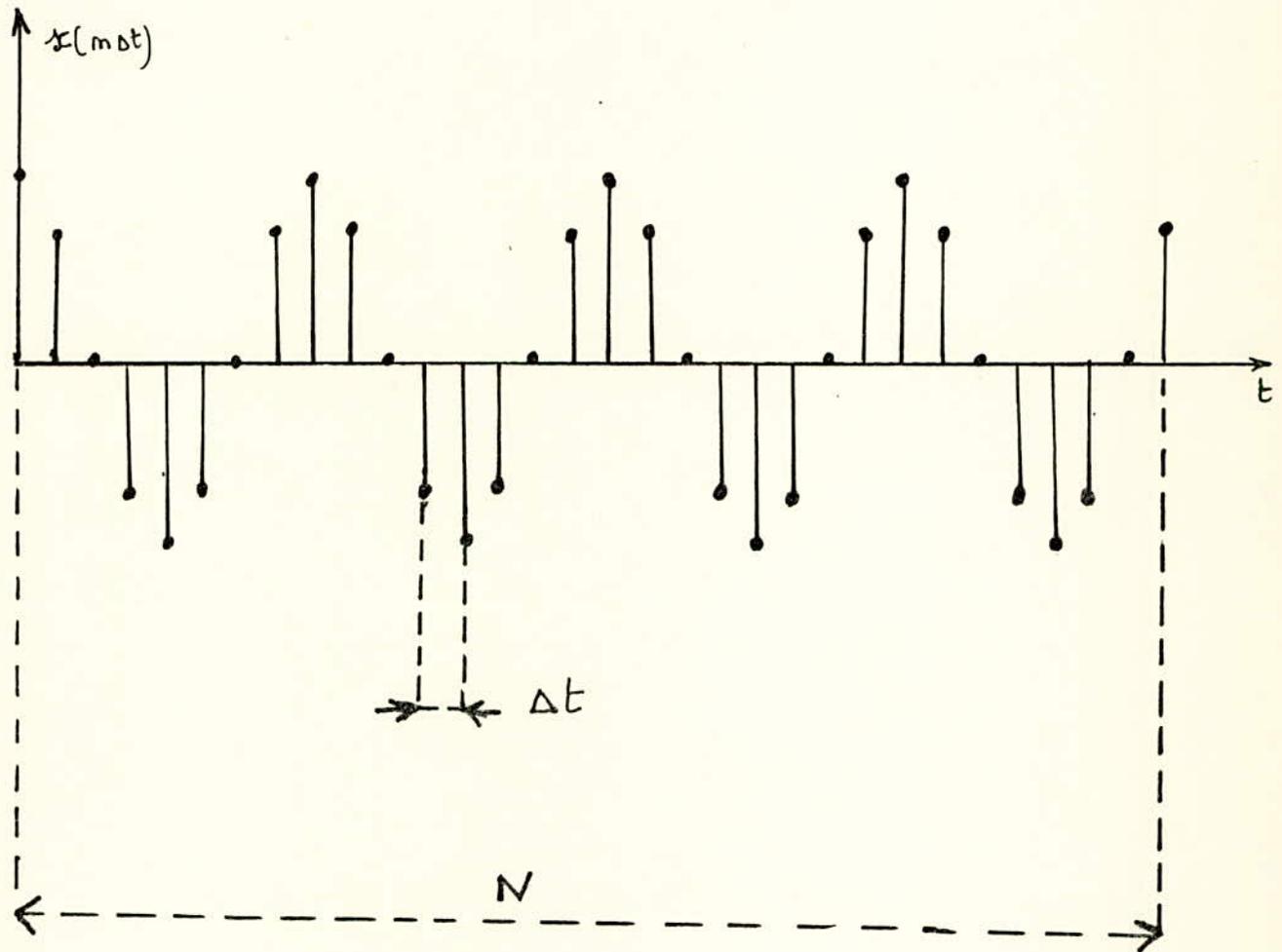
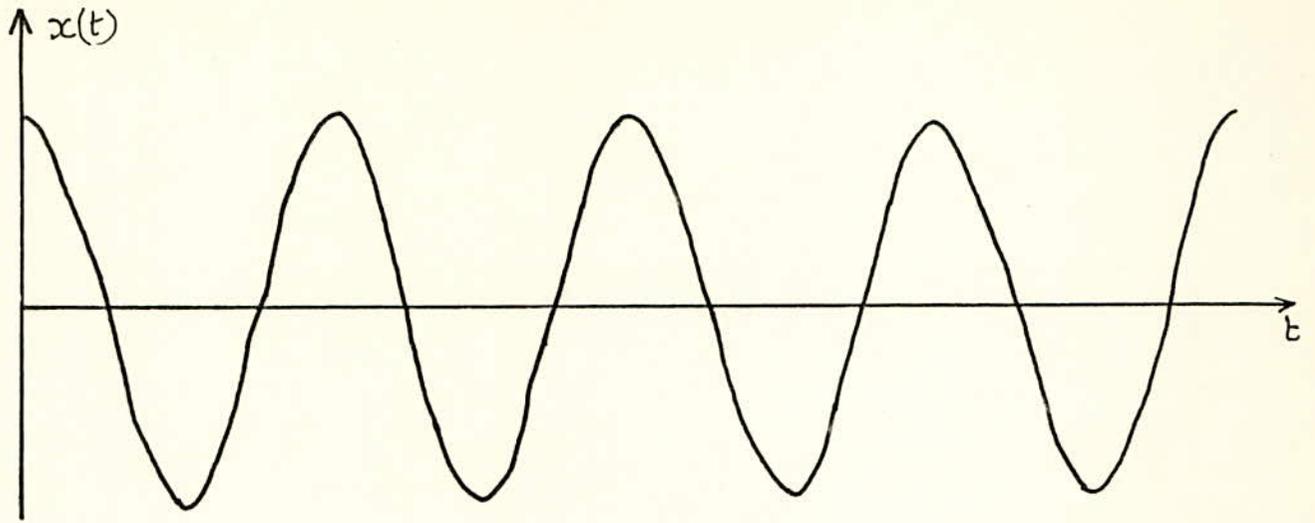
On désigne par $\hat{x}(t)$ la fonction échantillonnée:

$$\hat{x}(t) = \Delta t \sum_{K=-\infty}^{+\infty} x\left(\frac{K}{F_e}\right) \delta\left(t - \frac{K}{F_e}\right)$$

qui peut aussi s'écrire:

$$\hat{x}(t) = x(t) \Delta t \sum_{K=-\infty}^{+\infty} \delta\left(t - \frac{K}{F_e}\right)$$

d'où $\hat{x}(t) \xleftrightarrow{\quad} X(f) \times \sum_{n=-\infty}^{+\infty} \delta(f - nF_e)$



Δt : temps qui sépare deux échantillons, appelé période d'échantillonnage

N : Nombre d'échantillons (dimension du tableau ou du bloc mémoire)

Fig V.1

d'après la formule de poisson.

Ceci signifie que le spectre de $\hat{x}(t)$ est celui de $x(t)$ "périodisé" c'est à dire répété sur l'axe des fréquences avec une période $\Delta t = \frac{1}{F_e}$ (Fig. V .2).

Si $x(t)$ a une transformée de fourier $X(f)$ telle que $X(f)$ soit nulle pour $|f| > F_M$, le spectre de $x(t)$ va donc s'étendre sur une largeur $2F_M$, de $-F_M$ à $+F_M$.

Pour que la répétition périodique de ce spectre ne déforme pas le motif répété, il faut, et il suffit, que la fréquence de répétition F_e , qui est la fréquence d'échantillonnage, soit égale ou supérieure à $2F_M$ (théorème de SHANNON)

DEMONSTRATION:

Considérons le spectre de $\hat{x}(t)$:

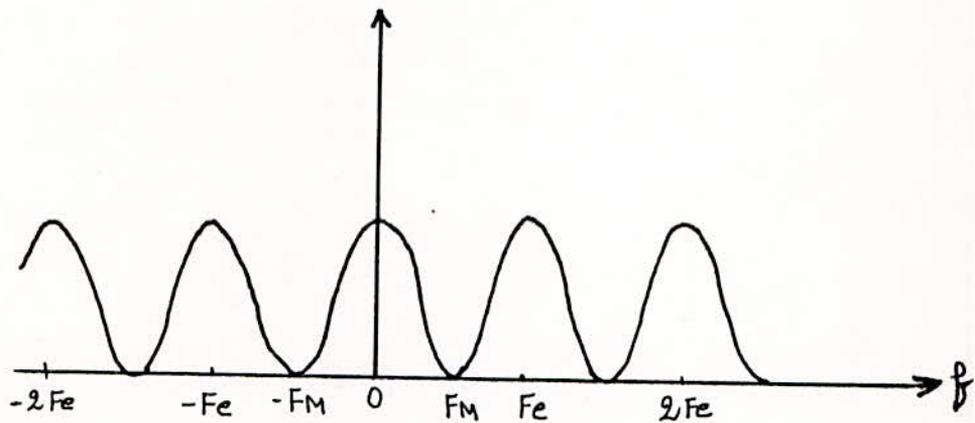
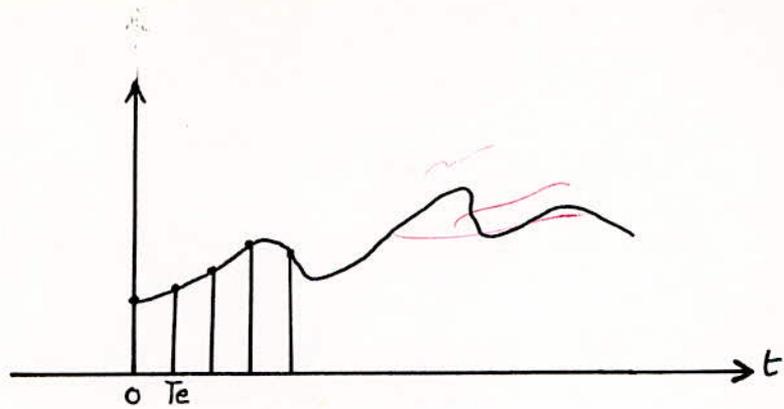
$$X(f) \times \sum_{n=-\infty}^{+\infty} \delta(f - nF_e) = \sum_{n=-\infty}^{+\infty} X(f - nF_e)$$

Si on multiplie cette expression par une fonction porte de largeur F_e , soit $\Pi_{\frac{F_e}{2}}(f)$, cette fonction étant nulle à l'extérieur de l'intervalle $-\frac{F_e}{2}, +\frac{F_e}{2}$,

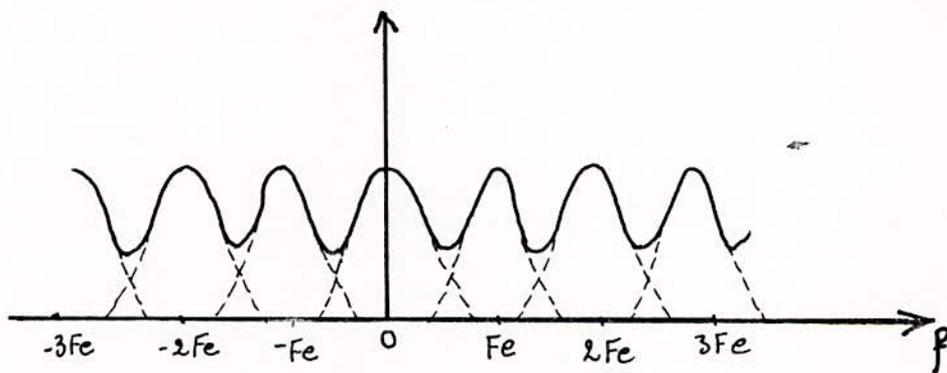
on va retrouver le spectre initial.

On aura donc:

$$\left[X(f) \times \sum_{n=-\infty}^{+\infty} \delta(f - nF_e) \right] \Pi_{\frac{F_e}{2}}(f) = X(f)$$



spectre dans le cas de l'échantillonnage
à $F_e > 2F_M$



spectre dans le cas de l'échantillonnage
à $F_e < 2F_M$

FIG V.2

Si nous prenons la transformée de fourier des deux membres:

$$X(f) * \sum_{n=-\infty}^{+\infty} \delta(f-nF_e) \rightleftharpoons \hat{x}(t)$$

$$\left[\frac{F_e}{2} (f) \rightleftharpoons F_e \frac{\sin \pi F_e t}{\pi F_e t}$$

$$X(f) \rightleftharpoons x(t)$$

d'où: $\hat{x}(t) * F_e \frac{\sin \pi F_e t}{\pi F_e t} = x(t)$

mais: $\hat{x}(t) = T_e \sum_{K=-\infty}^{+\infty} x\left(\frac{K}{F_e}\right) \delta\left(t - \frac{K}{F_e}\right)$

entraîne: $\sum_{K=-\infty}^{+\infty} x\left(\frac{K}{F_e}\right) \delta\left(t - \frac{K}{F_e}\right) * \frac{\sin \pi F_e t}{\pi F_e t} = x(t)$

mais: $\delta\left(t - \frac{K}{F_e}\right) * \frac{\sin \pi F_e t}{\pi F_e t} = \frac{\sin \pi F_e \left(t - \frac{K}{F_e}\right)}{\pi F_e \left(t - \frac{K}{F_e}\right)}$

donc: $x(t) = \sum_{K=-\infty}^{+\infty} x\left(\frac{K}{F_e}\right) \cdot \frac{\sin \pi F_e \left(t - \frac{K}{F_e}\right)}{\pi F_e \left(t - \frac{K}{F_e}\right)}$

Ce résultat très important montre que l'on peut reconstituer une fonction à partir de ses échantillons, pourvu que la fréquence d'échantillonnage soit supérieure ou égale à deux fois la plus haute fréquence contenue dans le Spectre.

C H A P I T R E VI

T R A N S F O R M E E D E F O U R I E R D I S C R E T E

(T F D) E T T R A N S F O R M E E D E F O U R I E R

R A P I D E (F F T)

La T F D et La F F T sont des opérations numériques appliquées à des suites de données qui doivent être numérisées et mémorisées pour un traitement ultérieur.

La T.F.D est donnée par la relation suivante:

$$X(K) = \frac{1}{N} \sum_{n=0}^{n-1} x(n) \cdot e^{-j \frac{2\pi Kn}{N}} \quad K=0,1,\dots,N/2$$

N: dimension du vecteur.

K: indice utilisé pour la génération des coefficients de Fourier.

X(K): K^{ième} coefficient de Fourier.

x(n): (n+1)^{ième} élément des données réelles introduites.

en appliquant la formule d'Euler on peut obtenir directement la partie réelle et la partie imaginaire de la T F D.

$$\text{Re } (K) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) \cdot \cos 2\pi Kn/N$$

$$\text{Im } (K) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) \cdot \sin 2\pi Kn/N$$

Pour $K=0$, on obtient la composante continue qui est la moyenne arithmétique des échantillons.

$$W(1) = X(0) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) \cdot e^0 = x(0) + x(1) + \dots + x(N-1)$$

Le nombre d'opérations nécessaires pour calculer toutes les valeurs des parties réelles et imaginaires est de N^2 .

VI.1 SOUS-PROGRAMME DE CALCUL DE LA T.F.D.

```

470 REM X X  CALCUL DE LA TFD
480 PAGE
490 DELETE R,C
500 DIM R(N)
510 DIM C(N)

520 R=0
530 C=0

540 FOR L=0 TO N-1
550 K=L+1
560 FOR I=1 TO N
570 R(K)=R(K)+W(I) * COS (2 * PI * L * (I-1)/N)
580 C(K)=C(K)+W(I) * SIN (2 * PI * L * (I-1)/N)
590 NEXT I
600 NEXT L
610 R=R/N
620 C=C/N
630 REM X X  CALCUL DU MODULE ET DE LA PHASE
640 DELETE M,P
650 DIM M(N),P(N)
660 FOR I=1 TO N
670 M(I)=(R(I)2+C(I)2)0.5
680 P(I)=ATN(C(I)/R(I)) * 180/PI
690 NEXT I
700 T S ="T.F.D"
710 H S ="HERTZ"
720 U S ="VOLTS"
730 A1 =1
740 RETURN

```

Ce sous-programme permet de calculer les parties réelles et imaginaires ainsi que le module et la phase de la T.F.D.

Les résultats sont donnés sous forme de vecteurs de dimension n chacun.

VI.2 CALCUL DE LA F F T

On appelle transformée de Fourier rapide (F F T) tous les algorithmes de calcul de la F F T qui permettent de réduire le nombre d'itérations.

Les algorithmes les plus connus sont ceux qui font les calculs sur des séries de points de dimension $N=2^p$ où p est un nombre entier, dans ce cas le nombre d'itérations est réduit à $N \log_2 N$.

FORME SYNTAXE:

n° de ligne CALL F.F.T TABLEAU

La commande "FFT" calcule une transformée de Fourier discrète et la place dans le tableau argument.

EXEMPLE: 330 CALL "F F T" , W

L'argument W est nécessairement un vecteur de dimension 2^n compris entre 16 et 1024 (16,32,64,128,256,512,1024). Avant que la commande ne soit exécutée, le tableau doit contenir les valeurs des échantillons du signal sur lequel va être effectuée la FFT.

Une fois la commande exécutée, le tableau initial contient les résultats du calcul de la F.F.T. Les valeurs complexes sont alors placées dans l'ordre suivant, où W est l'argument et N est le nombre de points d'échantillonnage:

W(1): Nombre réel représentant la composante continue du ~~Séparé~~ Signal

W(2): Nombre réel représentant la valeur de la T.F.D à la fréquence de Nyquist ($\frac{1}{2}$ fréquence d'échantillonnage)

W(3): Partie réelle du premier coefficient de Fourier.

W(4): Partie imaginaire du premier coefficient de Fourier.

.
. .

W(N-1): Partie réelle du ($\frac{N}{2} - 1$)^{ième} coefficient de Fourier.

W(N): Partie imaginaire du ($\frac{N}{2} - 1$)^{ième} coefficient de Fourier.

VI 2.1) SEPARATION DES PARTIES REELLES ET IMAGINAIRES DE LA F F T

On peut Séparer les résultats obtenus par "F F T" en deux tableaux, l'un contenant des composantes réelles et l'autre des composantes imaginaires, grâce à la commande "UNLEAV".

FORME SYNTAXE

n) ~~de ligne~~ ~~CALL~~ "UNLEAV" tableau, tableau

n° de ligne CALL "UNLEAV" tableau, tableau, tableau

EXP: 1762 CALL "UNLEAV", W, R, P.

Résultat:

$$\begin{array}{l} R(1) = W(1) \\ R(2) = W(3) \\ \cdot \\ \cdot \\ \cdot \\ R\left(\frac{N}{2}+1\right) = W(2) \end{array}$$

$$\begin{array}{l} P(1) = 0 \\ P(2) = W(4) \\ \cdot \\ \cdot \\ \cdot \\ P\left(\frac{N}{2}\right) = W(N) \\ P\left(\frac{N}{2}+1\right) = 0 \end{array}$$

VI.2.2 CALCUL DU MODULE ET DE LA PHASE DE LA FFT

De même on peut aussi convertir le tableau de données résultant de la commande FFT sous forme de coordonnées rectangulaires à la forme coordonnées polaires. Et ceci en utilisant l'instruction POLAR. La forme polaire consiste en deux tableaux:

L'un contenant les termes d'amplitude, l'autre les termes de phase.

FORME SYNTAXE:

n° de ligne CALL "POLAR", W,M,P

• M,P:tableaux résultant de l'opération POLAR. Ils contiennent l'amplitude et la phase respectivement. Leur dimension est de $\frac{N}{2} + 1$ chacun.

-La phase est calculée modulo 2π ; c'est à dire toutes les valeurs de la phase sont comprises entre $-\pi$ et $+\pi$.

VI.3 QUELQUES REMARQUES SUR LA COMMANDE FFT

Une fois la transformée de Fourier rapide calculée on constate les remarques suivantes:

1) La commande FFT ne tient pas compte du facteur $\frac{1}{N}$ qui se trouve dans l'expression de la T.F.D, il a fallut donc diviser à chaque fois les arguments W par N .

2) Le nombre de points dans le domaine des fréquences est de dimension moitié car l'information dans le domaine des fréquences est répartie en deux suivant le type de représentation partie réelle-partie imaginaire ou amplitude-phase.

3) La gamme de fréquence d'un signal est comprise entre: Zéro hertz et une fréquence maximale appelée fréquence de Nyquist.

4) l'écart de fréquence entre deux points consécutifs définit la résolution fréquentielle: (Δf)
les relations entre ces différents paramètres se trouvent dans le tableau N° 1. Où T représente la durée de la fenêtre utilisée.

T	$\Delta t = \frac{1}{2F_{\text{max}}}$	$\Delta t = \frac{T}{N}$
f	$\Delta f = \frac{1}{T}$	$\Delta f = \frac{F_{\text{max}}}{N/2}$
T	$T = \frac{1}{\Delta f}$	$T = N \times \Delta t$

TABLEAU .1

VI.4 : COMPARAISON ENTRE T.F.D ET F.F.T

Pour faire cette comparaison, on prend l'exemple de la fonction Sinus qu'on échantillonne plusieurs fois, et à chaque fois, on fait varier le nombre de points d'échantillonnage (N) et on calcule le nombre d'opérations dans le cas de la T.F.D et la F.F.T.

Les résultats de calcul se trouvant dans le tableau.2; sont représentés par les graphes de la fig VI.1.

On remarque que le calcul de la FFT est plus rapide car il nécessite un nombre d'opérations égal à $N \log_2 N$ par contre le calcul de la TFD nécessite un nombre d'opérations égal à N^2 .

Les graphes de la Fig VI.2.a et de la Fig VI.2.b représentent respectivement les parties réelles et imaginaires de la T.F.D et de la FFT (tableau 3 et 4).

Dans le cas de la FFT. On remarque que le nombre de points dans le domaine des fréquences est de dimension moitié, ces points représentent les fréquences positives car lorsqu'un système calcul la transformée de Fourier d'une fonction physique, il opère sur une fonction réelle donc sans composante imaginaire; on sait que dans ce cas la transformée de Fourier est douée de Symétrie hermitique c'est à dire que sa partie réelle est paire et sa partie imaginaire est impaire:

$$\begin{array}{l}
 \text{Si } x(t) \text{ est réelle:} \\
 \begin{array}{ccc}
 X(t) & \xleftrightarrow{\hspace{2cm}} & X(f) = R(f) - jI(f) \\
 & & \left. \begin{array}{l} R(f) \text{ paire} \\ I(f) \text{ impaire} \end{array} \right\} \begin{array}{l} \text{Symétrie} \\ \text{Hermetique} \end{array}
 \end{array}
 \end{array}$$

On peut donc, à partir de la transformée de Fourier Calculer pour $f \geq 0$ accéder par Symétries au prolongement de cette transformée de Fourier pour les fréquences négatives.

Nombre de points (N)	Nombre d'opérations: N^2 (Cas T.F.D)	Nombre d'opérations: $N \log_2 N$ (Cas FFT)
16	256	64
32	1024	160
64	4096	384
128	16384	896
256	65536	2048
512	262144	4608
1024	1048576	10240

TABLEAU.2

Nombre d'opérations

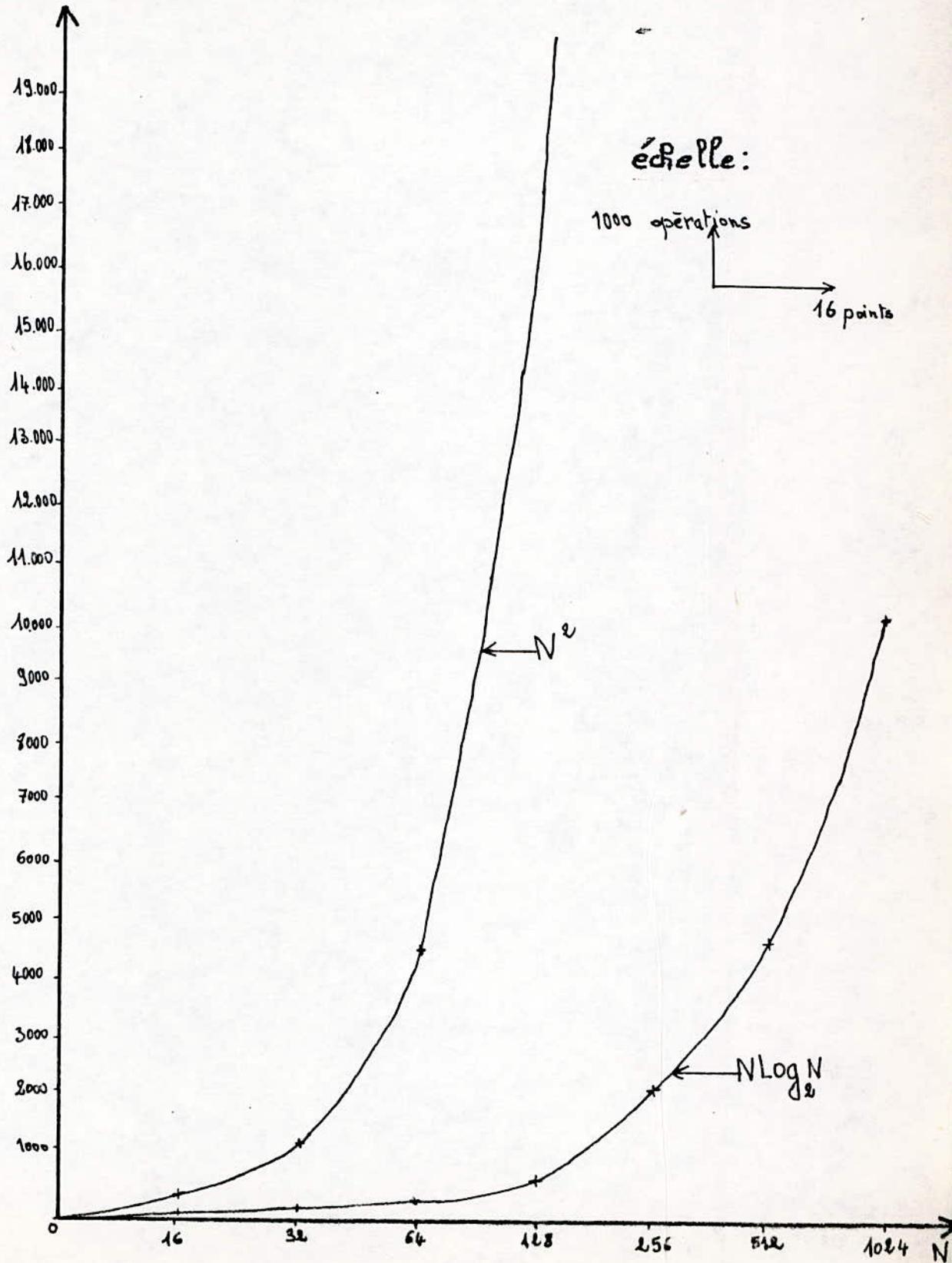


FIG VI.1

tableau. 3

T.F.D
 $N=32$
 $n=4$
 Amplitude
 $A_2 = 10^6$

R	-9.31736242E-21	-1.259961509E-20	-8.364450354E-21	-1.079967008E-20	
	5.0E-7	-1.196434038E-20	-2.001115338E-20	-3.441071348E-21	
	-1.153134154E-20	-3.891057601E-21	-4.870439447E-21	-6.67038446E-21	
	2.964615315E-21	1.450543922E-20	1.270549421E-20	1.535247217E-20	
	4.235164736E-21	2.551686754E-20	-1.312901069E-20	1.778769189E-20	
	2.392868076E-20	3.261076847E-20	5.701590526E-20	5.317778722E-20	
	1.429588669E-20	8.682087709E-20	1.061430162E-19	2.715799387E-19	
	5.0E-7	-1.463249416E-19	-1.327724145E-19	-8.449153649E-20	
	I	0	2.911675756E-22	2.329340605E-21	1.429368098E-21
		6.140988868E-21	-1.376428539E-21	1.03761536E-20	1.175258214E-20
5.929230631E-21		7.940933881E-21	1.122310655E-20	2.466983459E-20	
1.376428539E-21		2.694623563E-20	8.470329473E-21	4.123991662E-20	
1.675628898E-20		-8.232101456E-21	8.470329473E-22	1.995821382E-20	
3.832824086E-20		2.594038401E-20	3.758708703E-20	8.470329473E-21	
3.684593321E-20		5.505714157E-20	9.95263713E-21	4.235164736E-20	
-5.022905377E-19		4.621623518E-20	6.140988868E-20	9.298833574E-20	

58

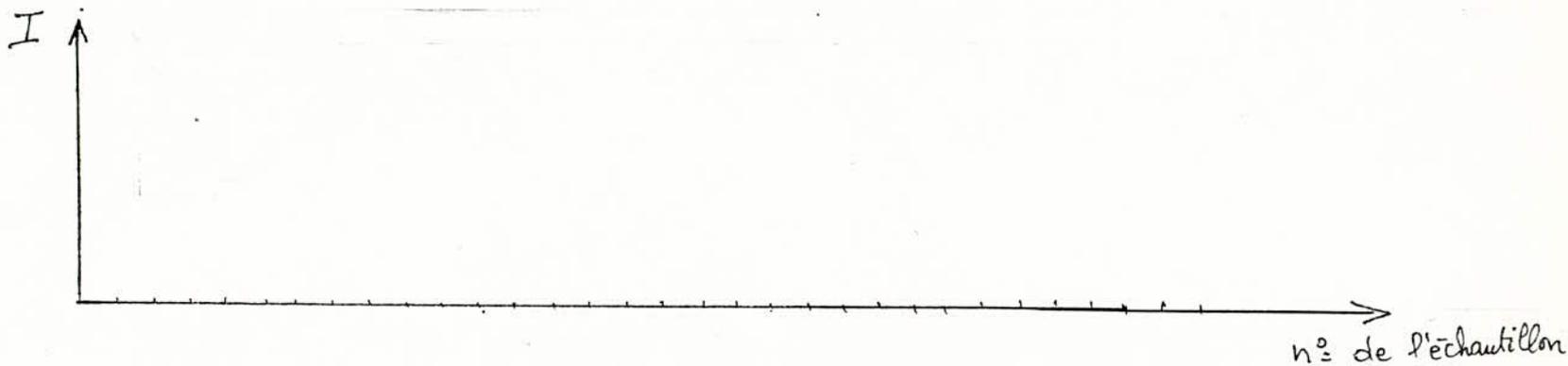
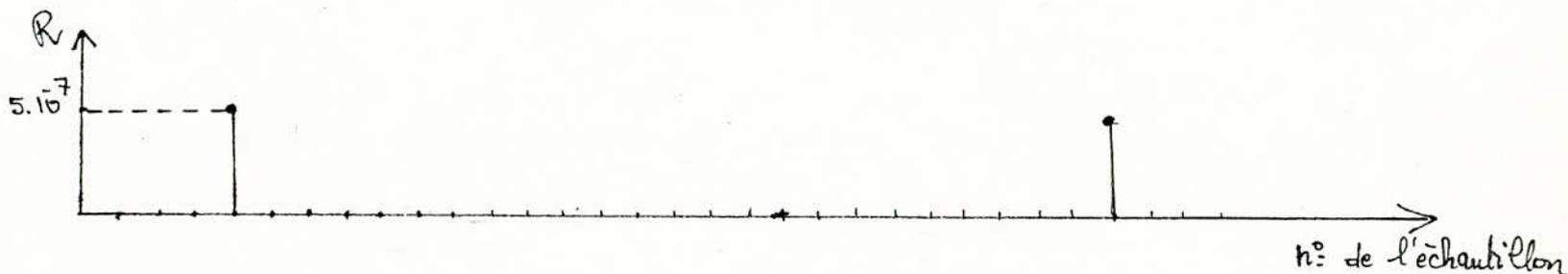


FIG VI 2.a

FFT :

Tableau 4

$N = 32$
 $n = 4$
 Amplitude:
 $A_2 = 10^{-6}$

R {

$3.794255053E-21$	$1.250181108E-20$	$1.070171199E-20$	$2.445964451E-20$
$5.0E-7$	$1.697464889E-20$	$1.599110272E-21$	$7.408566746E-21$
$3.231890491E-21$	$7.372728126E-21$	$2.292438871E-21$	$6.537556433E-21$
$3.638875053E-21$	$3.514097051E-21$	$7.234081689E-21$	$4.961683192E-21$
$3.911239156E-21$			

I {

-3.14159265359	-3.06117360043	-3.08823240845	3.04683095682
$-1.151964908E-13$	-0.00596668924361	-0.699191512219	-0.726222213339
-0.714466875407	-1.16995220895	-0.235963831295	-0.783204863838
-0.19739555985	-0.675364042429	-0.0107155843261	-0.617535076462

59

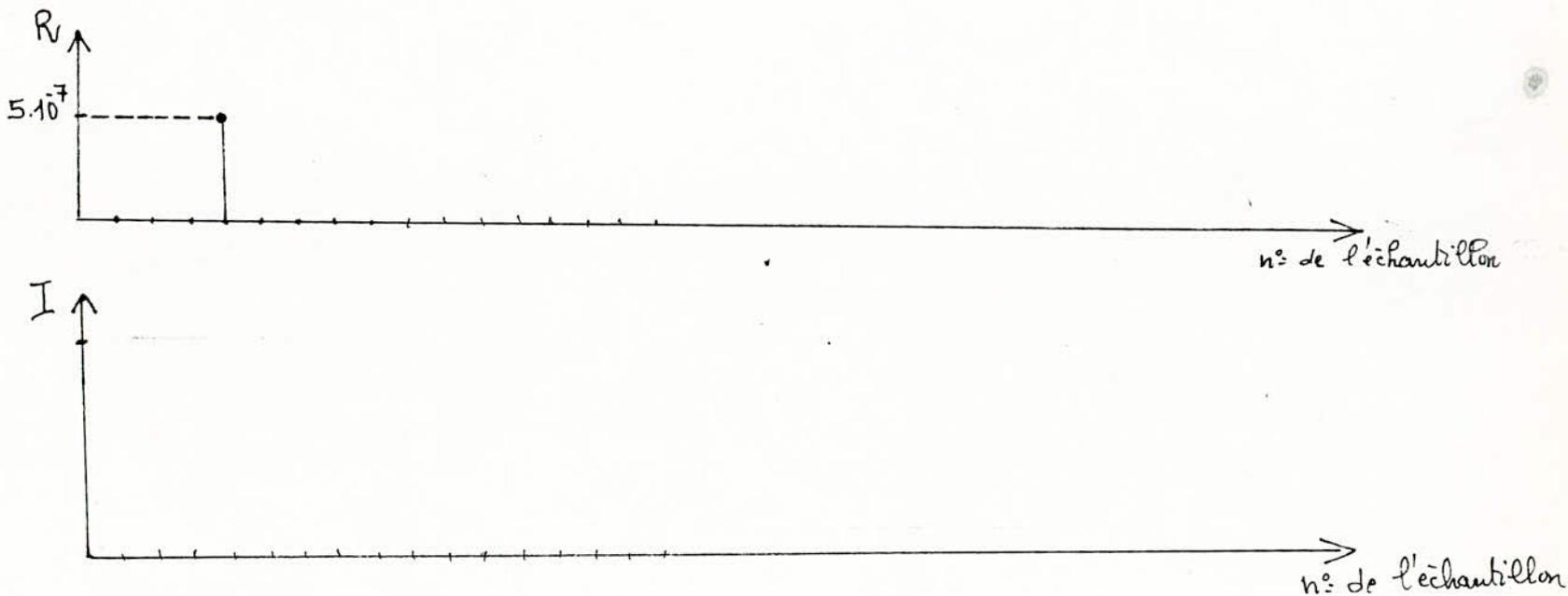


FIG VI.2.b

C H A P I T R E VII

UTILISATION DU SOUS-PROGRAMME DE TRACE GRAPHIQUE

§ POUR COMMENTER LES PROPRIETES DE LA T.F.

VII.1 INFLUENCE DE LA FREQUENCE D'ECHANTILLONNAGE:

On prend à chaque fois un nombre entier de périodes d'un signal sinusoïdal $f(t)$, on calcul pour chaque cas le nombre d'échantillons par période $\frac{1}{T}$ (N éch/période), la période d'échantillonnage (Δt), la fréquence d'échantillonnage (F_e), ainsi que la fréquence de Nyquist. Les résultats obtenus figurent dans le tableau.5.

Pour tous les cas la fréquence du signal $f(t)$ est de 100 KHZ et le nombre de points d'échantillonnage $N = 128$ points.

Grâce au programme de tracer graphique on représente le signal $f(t)$ et la transformée de Fourier correspondante (voir fig VII.1a et fig VII 1.b)

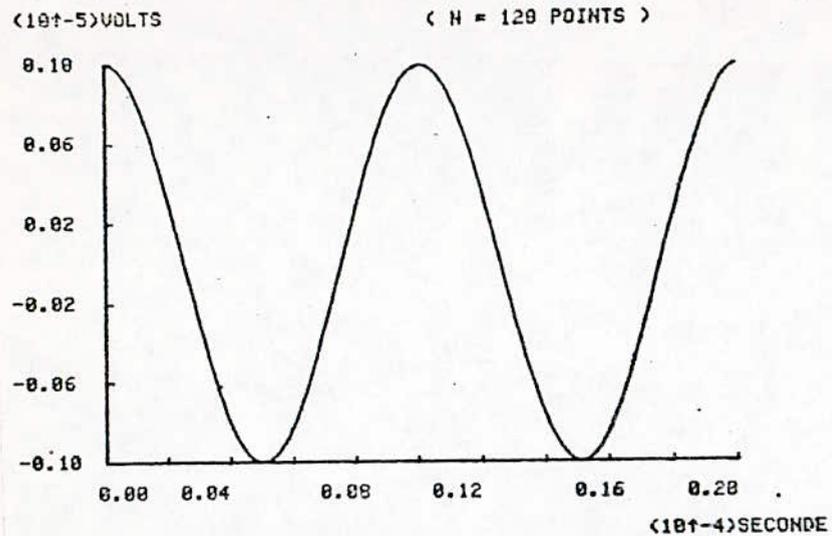
Si on analyse les transformées de Fourier on remarque que pour les cas $n = 2, 8, 64$, la position de la raie sur l'axe des fréquences coïncide avec la fréquence du signal considérée ($f = 100$ KHZ).

Par contre pour le cas $n = 100$, on a une bande de fréquence de 64 KHZ ce qui empêche la représentation de la raie

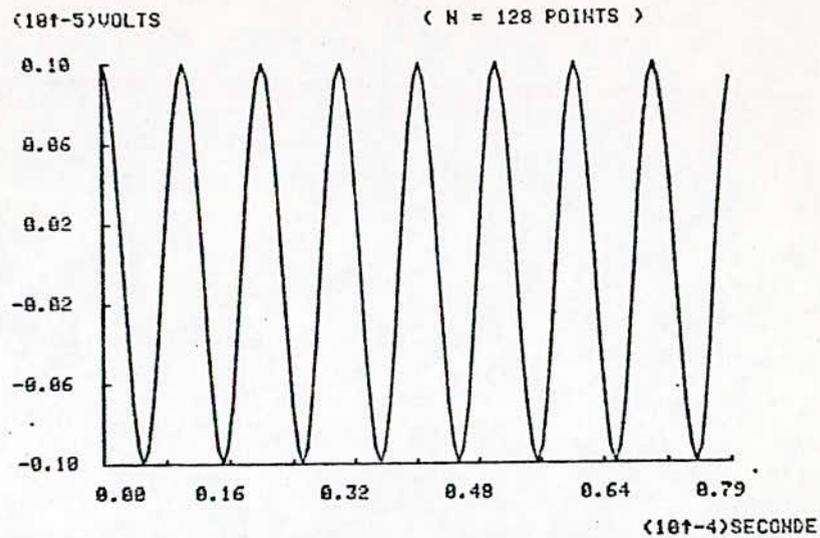
n	$N. \text{éch}/\text{période}$	$\Delta t \cdot 10^{-5} \text{ s}$	$F_e \cdot 10^5 \text{ Hz}$	$F_N \text{ (HZ)}$
2	64	0.015	64	3.20
8	16	0.062	16	8.00
64	2	0.500	2	1.00
100	1.28	0.780	1.28	0.64

TABLEAU .5

$m = 2$ périodes



$m = 8$ périodes



62

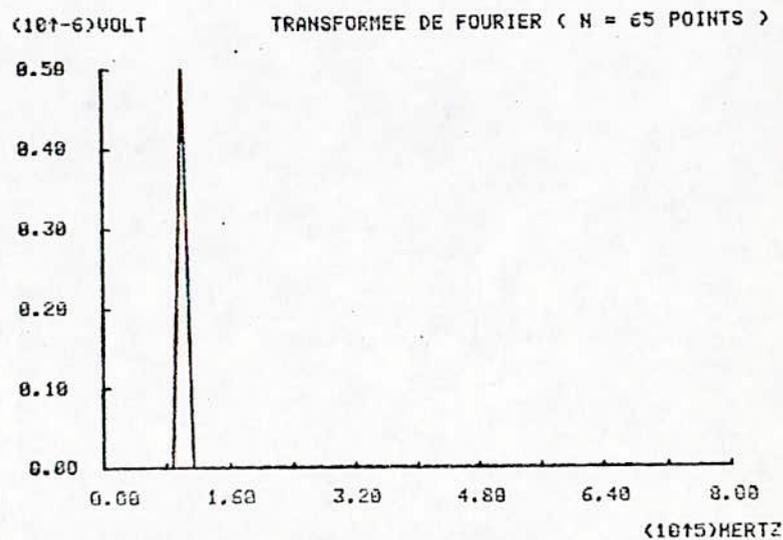
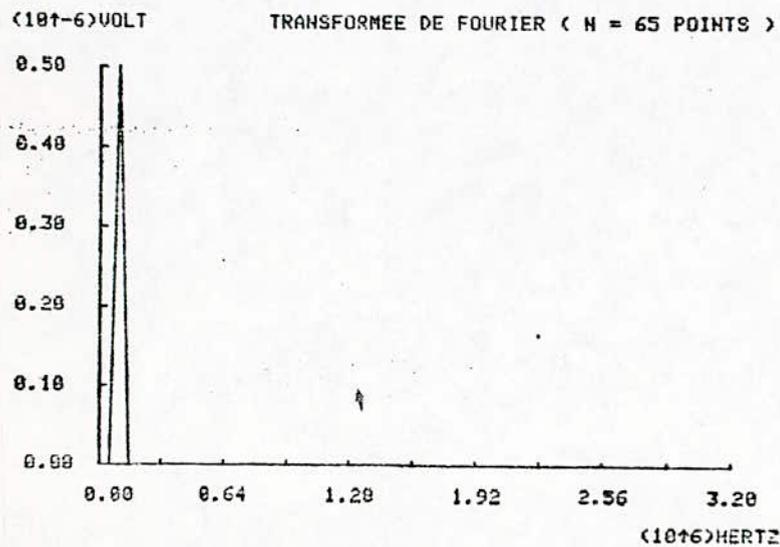
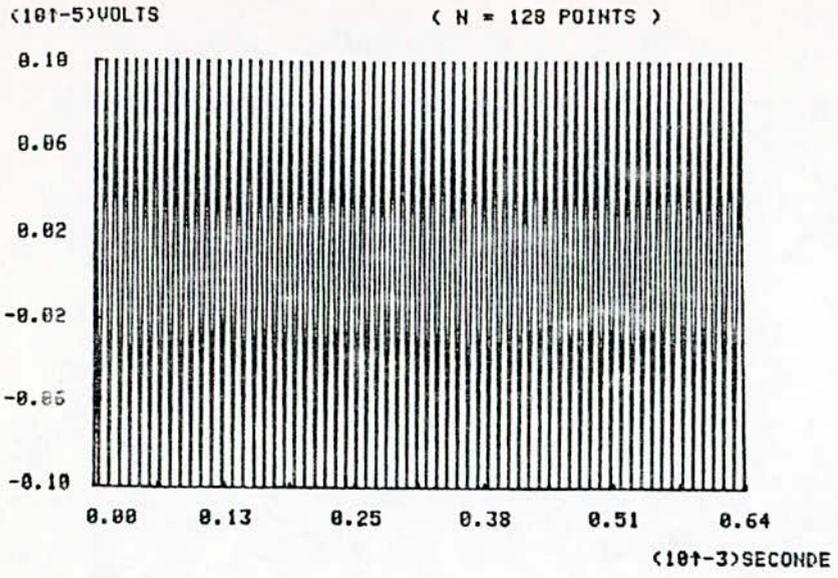


FIG VII.1a

$n = 64$ périodes



$n = 100$ périodes

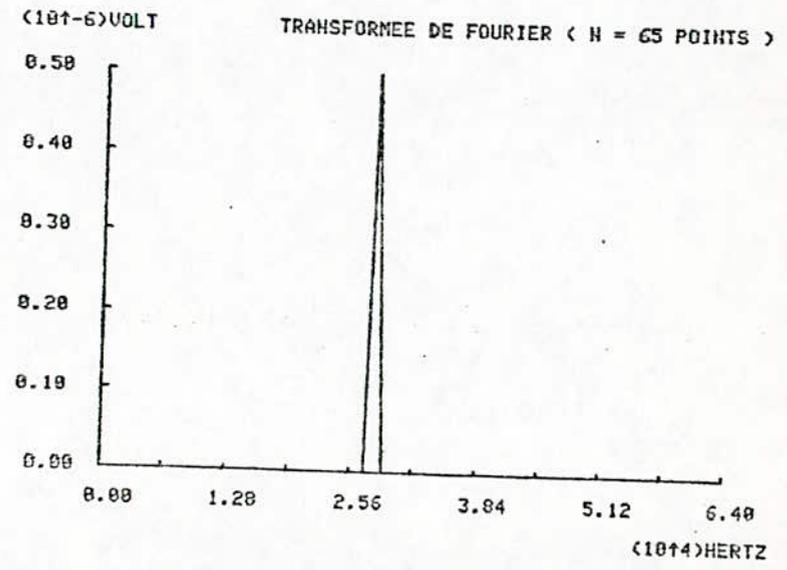
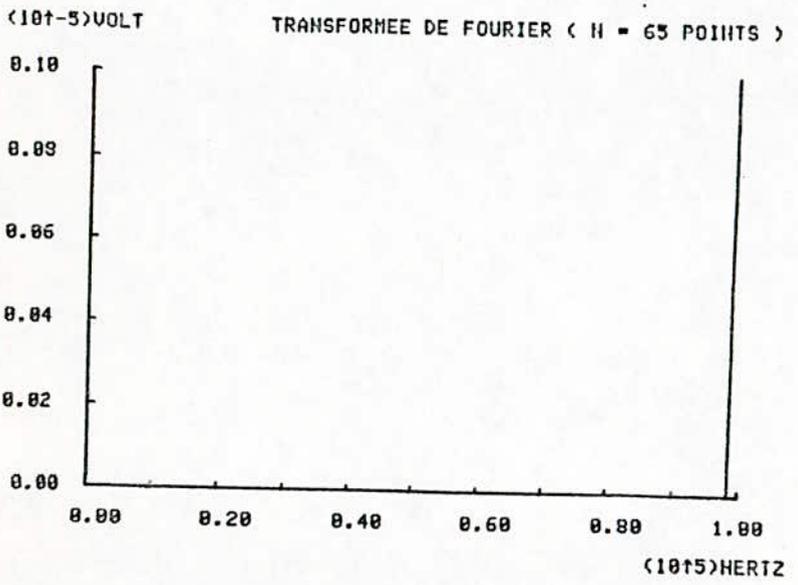
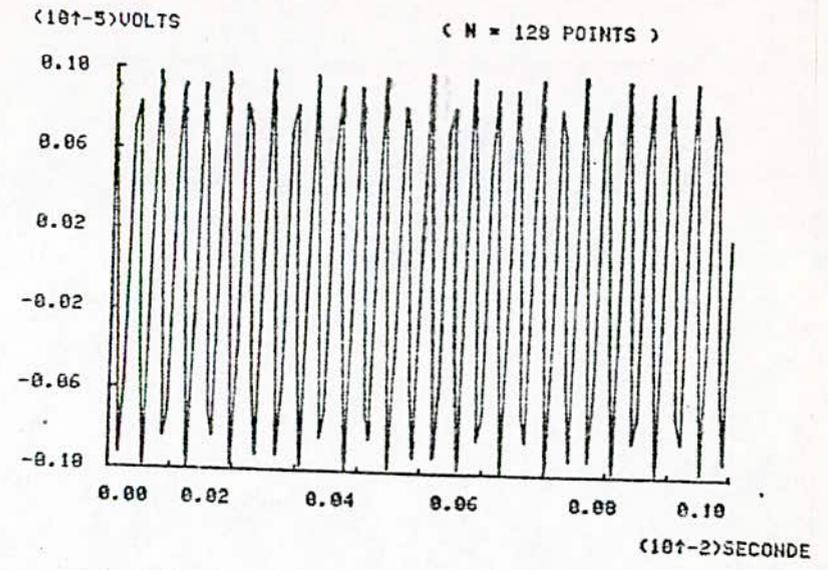


FIG VII 1.b

qui doit se trouver à la fréquence de 100 KHZ mais on remarque sur le graphe de la transformée de Fourier la présence d'une fausse raie ou image se trouvant à la fréquence de 28 KHZ. Pour ce cas on dit qu'on a le phénomène de repliement de Spectre.

Ce phénomène apparaît lorsqu'il existe, dans le signal analogique que l'on veut étudier une fréquence supérieure à la fréquence de Nyquist.

Nous allons illustrer ceci sur un exemple (voir fig VII.2) la fréquence d'échantillonnage choisie est de 120 KHZ, ce qui donne une fréquence de Nyquist de 60 KHZ.

Sur la figure (VII.2a) nous avons représenté une sinusoïde à 100 KHZ en traits pleins. Les points représentent les échantillons qui sont pris sur ce signal, il y a 1,2 échantillon par période. La sinusoïde en traits pointillés représente la fausse raie basse fréquence que l'on va retrouver sur le spectre.

Cette fausse raie se trouve à la fréquence de 20 KHZ.

cette raie est en fait le symétrique de la raie à 100 KHZ par rapport à la fréquence de Nyquist.

Tout se passe comme si la partie du Spectre entre 60 KHZ (F_N) et 120 KHZ (F_e) avait été repliée par rapport à la fréquence de Nyquist. (Fig VII.2 b)

L'Exemple que nous venons d'étudier met bien en évidence l'intérêt du théorème de Shannon.

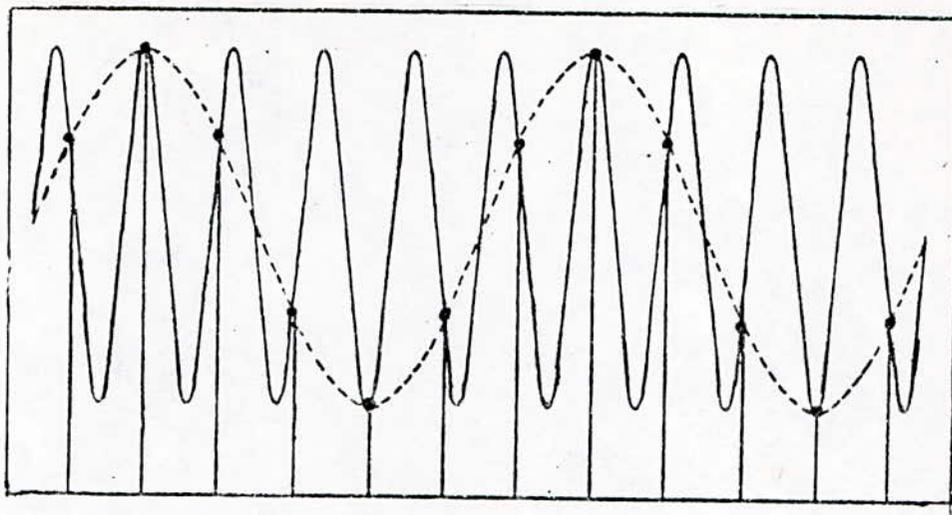


FIG.VII 2-a

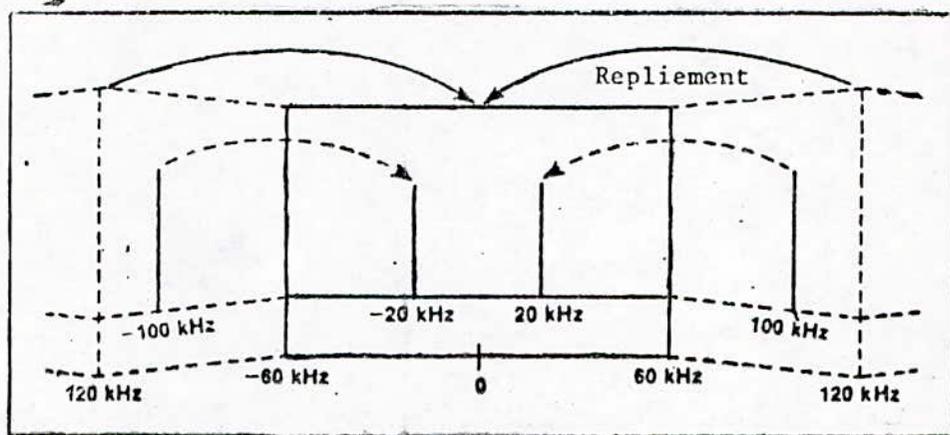


FIG . VII 2.b

FIG.VII.2-REPLIEMENT DU SPECTRE.

De façon pratique, pour étudier un Signal, il faut tout d'abord définir la fréquence d'échantillonnage suivant la bande de fréquence que l'on veut étudier.

En général, on considère qu'une fréquence d'échantillonnage de 4 à 5 fois supérieure à la fréquence de Nyquist donne de bons résultats.

Il faut ensuite définir le nombre de points qui doivent être numérisés suivant la résolution que l'on veut obtenir:

$$N = \frac{2FN}{\Delta f}$$

VII.2) INFLUENCE DE LA FENETRE TEMPORELLE:

Nous avons vu précédemment que le processus de numérisation d'un Signal consistait tout d'abord en la multiplication d'un Signal analogique continu par une fenêtre qui délimite une portion de ce Signal.

La FFT suppose que les données dans la fenêtre se répètent avec une période égale à la fenêtre temporelle. Ainsi, on suppose qu'il ya plusieurs fenêtres s'étendant de chaque côté de la fenêtre physique et chacune est une copie exacte de cette dernière.

On remarque que lorsque la fenêtre contient un nombre entier de périodes, la T.F.D du signal numérisé donnera la bonne position en fréquence et la bonne

amplitude du Signal. Par contre si le nombre de périodes n'est pas entier, à l'intérieur de la fenêtre, la TFD donnera plusieurs raies de part et d'autre de la fréquence du signal et l'amplitude du maximum n'aura pas la valeur attendue.

Nous avons représenté sur la Fig VII.3.b la TFD d'un signal sinusoïdal de 8 périodes dont la fréquence est de 2,50 Hz et sur la fig VII.3d nous avons représenté la TFD d'un signal sinusoïdal de 8,5 périodes et dont la fréquence est de 2,65 Hz, La fenêtre utilisé est de 3,2 secondes.

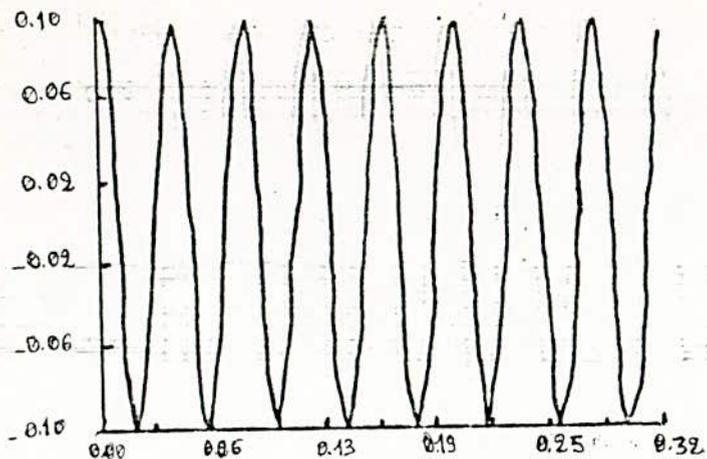
Dans le cas où le nombre de périodes existant dans une fenêtre est impaire, le signal ne sera pas périodique et présente des discontinuités, ceci se traduit dans le domaine des fréquences par la présence des oscillations parasites qui peuvent prêter à confusion. FIG VII.4 l'élargissement du spectre dû à des oscillations parasites peut masquer des raies correspondantes à la réalité et qu'on ne pourra pas observer. Pour atténuer ce cas défavorable on fait appel à une fonction de pondération qui nous permet d'avoir une périodicité du signal.

Il existe plusieurs types de pondérations (voir annexe) parmi ces pondérations on utilise la pondération de Hanning qui est défini par:

$m = 8$ Périodes

(10⁻⁵) VOLTS

(N = 128 points)



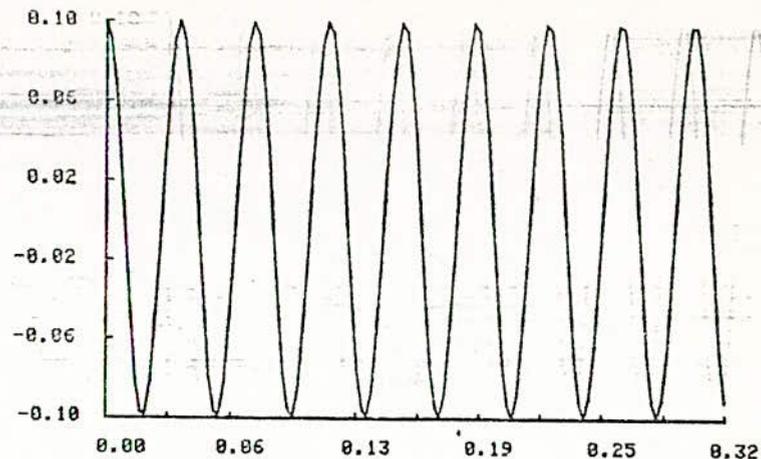
(10⁻¹) SECONDE

FIG VII 3.a

$m = 8.5$ Périodes

(10⁻⁵) VOLTS

(N = 128 POINTS)

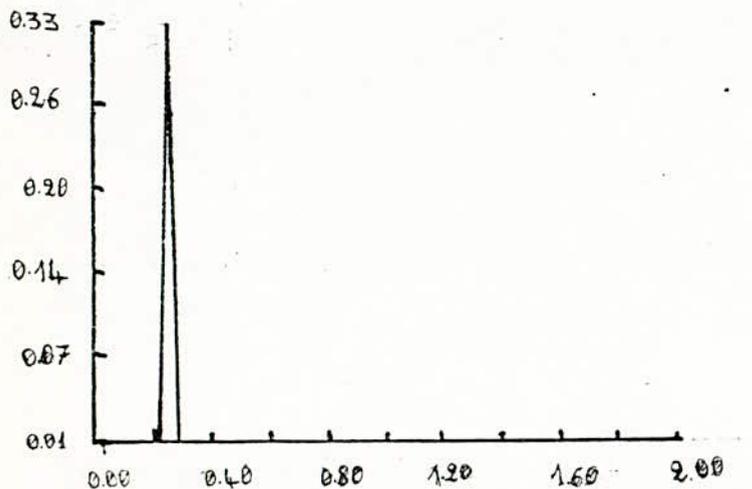


(10⁻¹) SECONDE

FIG VII 3.c

(10⁻⁶) VOLT

TRANSFORMEE DE FOURIER (N = 65 POINTS)

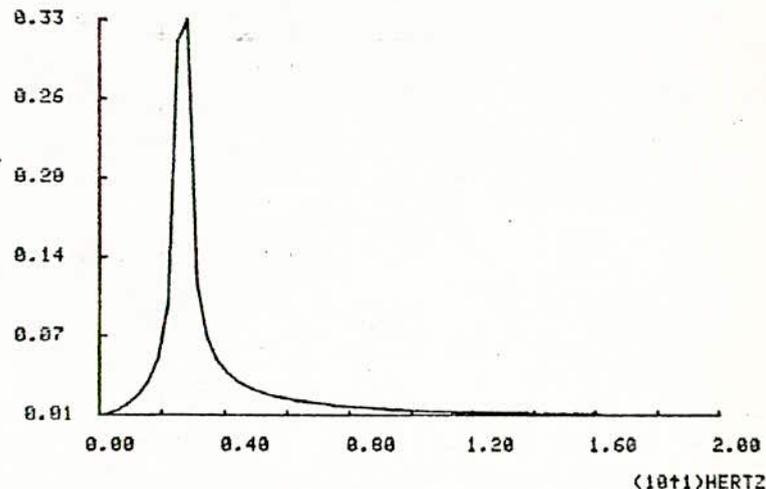


(10⁻¹) HERTZ

FIG VII 3.b

(10⁻⁶) VOLT

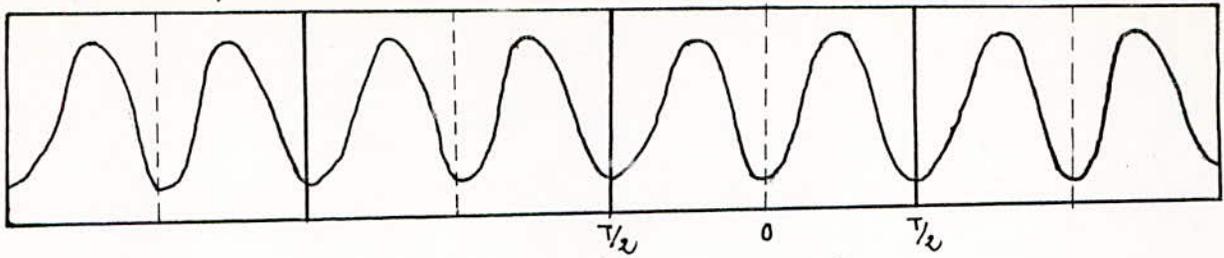
TRANSFORMEE DE FOURIER (N = 65 POINTS)



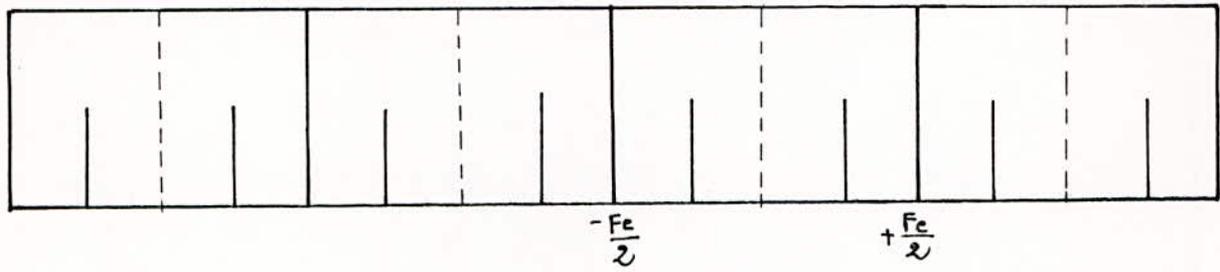
(10⁻¹) HERTZ

FIG VII 3.d

Domaine temporel

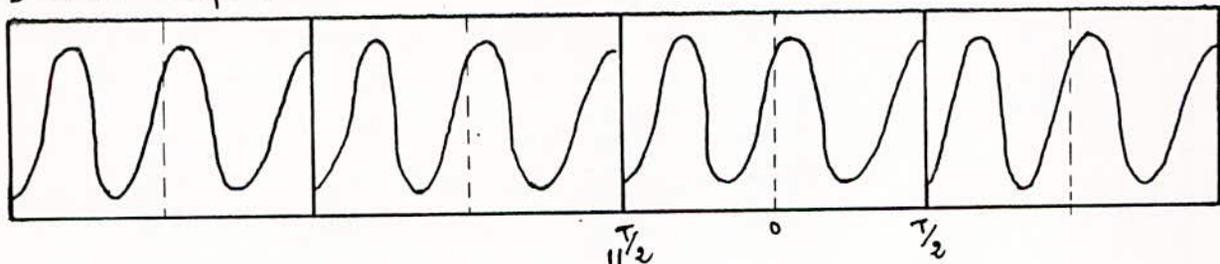


Domaine des fréquences

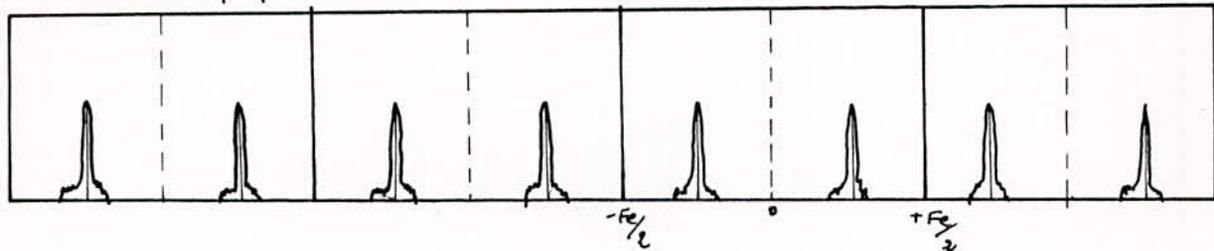


Cas où le nombre de périodes est pair

Domaine temporel



Domaine des fréquences



Cas où le nombre de périodes est impair

FIG VII.4

$$A = \alpha (1 - \cos 2 \pi 5t/T) \text{ Pour } \begin{cases} 0 \leq t \leq T/10 \\ \text{et } 9T/10 \leq t \leq T \end{cases}$$

$$A = 1 \quad \text{Pour } T/10 < t < 9T/10$$

Avec $\alpha = 0,5$.

EXEMPLE:

N° de ligne REM XX utilisation de la fenêtre de pondération.

N° de ligne CALL "TAPER" W, 0,5

Le resultat de ces deux lignes est équivalent à multiplier chaque point de la fonction considérée par la valeur correspondante de la fonction de pondération ($\alpha = 0,5$). Sur la fig VII.5.a et la fig VII.5.b nous avons représenté les fonctions de pondérations ainsi que leurs transformée de Fourier correspondante par les cas où $\alpha = 0,1, 0,3, 0,4, 0,5$. On remarque que lorsque α augmente, les lobes secondaires sont atténués, mais le bout de la raie spectrale a tendance à s'élargir.

Il convient de ne pas perdre de vue que toute pondération si elle lisse le spectre par le fait même diminue sa définition.

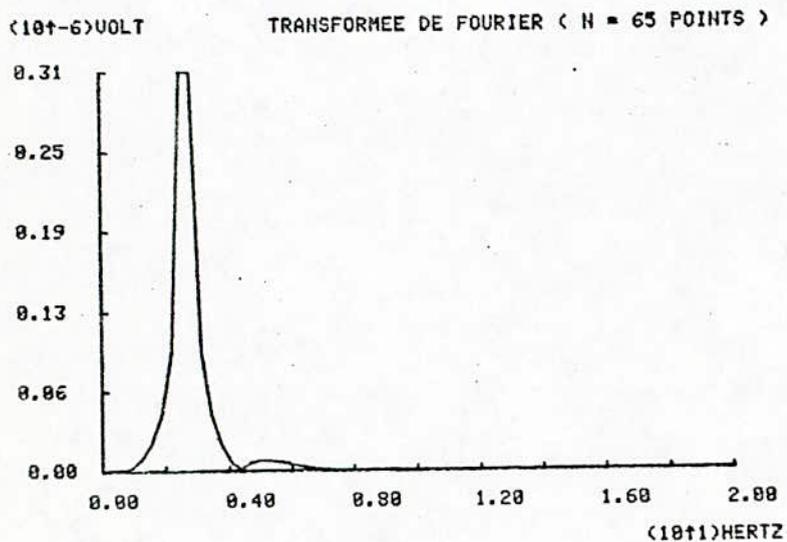
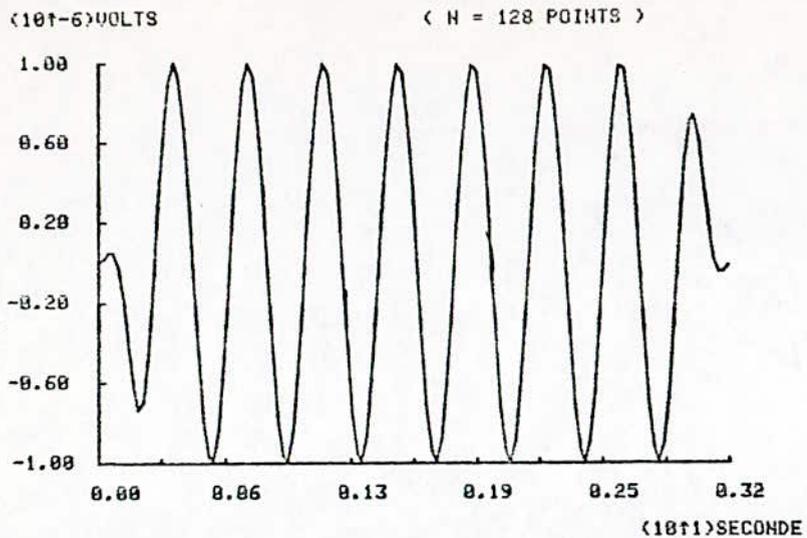
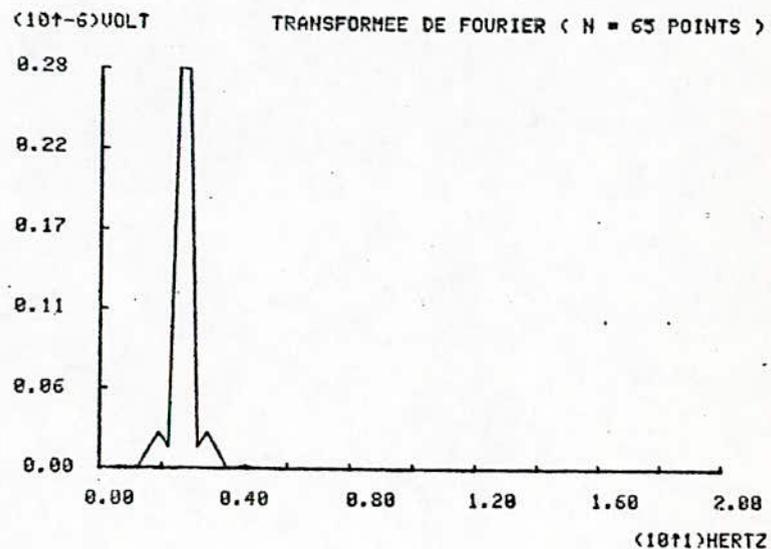
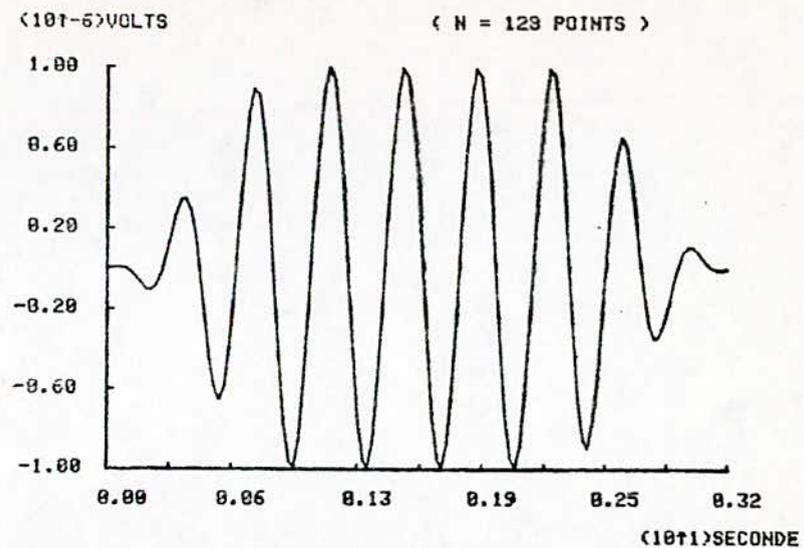
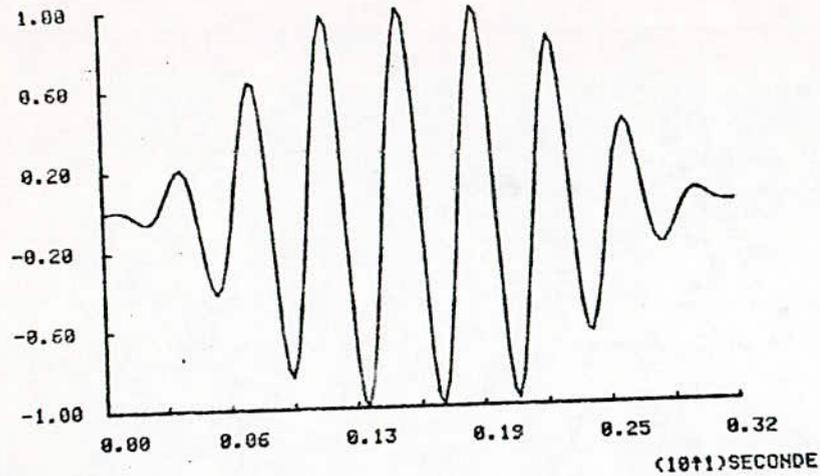
$\alpha = 0.1$  $\alpha = 0.3$ 

FIG VII. 5a

$d = 0.4$

(10⁻⁶)VOLTS

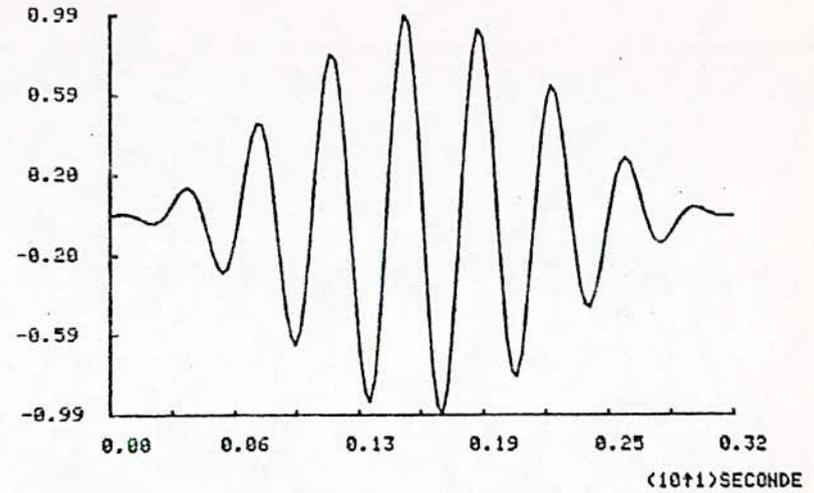
(N = 128 POINTS)



$d = 0.5$

(10⁻⁶)VOLTS

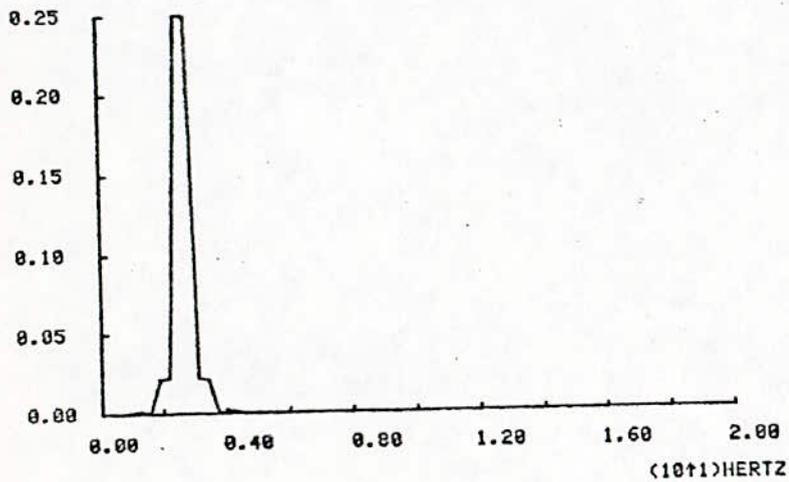
(N = 128 POINTS)



72

(10⁻⁶)VOLT

TRANSFORMEE DE FOURIER (N = 65 POINTS)



(10⁻⁶)VOLT

TRANSFORMEE DE FOURIER (N = 65 POINTS)

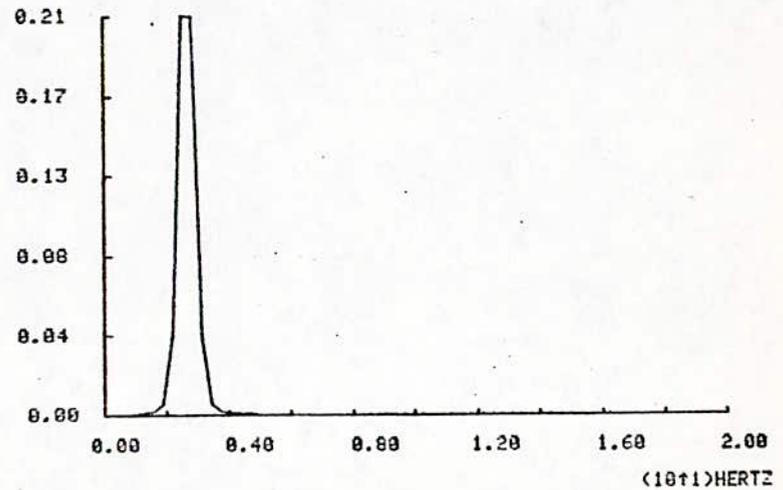


FIG VII 5 b

Il faut donc être très prudent dans l'utilisation de ces pondérations et ne pas les utiliser d'une manière systématique.

Dans le cas douteux on calcule le Spectre avec pondération puis sans pondération.

Nous avons représenté le Spectre avec pondération (Fig VII.6a) et le Spectre sans pondération (Fig VII.6b) des signaux carrés.

Nous avons représenté aussi le Spectre avec pondération (Fig VII.7.a) et le spectre sans pondération (Fig VII 7b) des signaux triangulaires.

VII.3 INFLUENCE DU DECALAGE TEMPOREL:

Un décalage temporel du signal affecte seulement la phase dans le domaine des fréquences de telle sorte que le module reste constant, mais le rapport partie imaginaire sur partie réelle varie et affecte la phase.

Sur les figures VII 8.a et VII 8.b nous avons représenté l'exemple de la fonction "projectrice".

Un décalage dans le temps de la fonction projectrice fait varier les parties réelles et imaginaires de la TFD par contre le module reste inchangé.

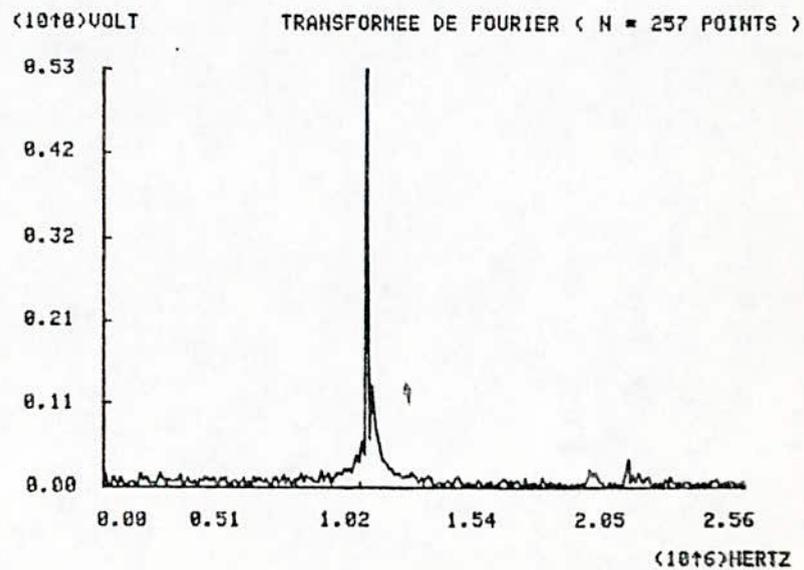
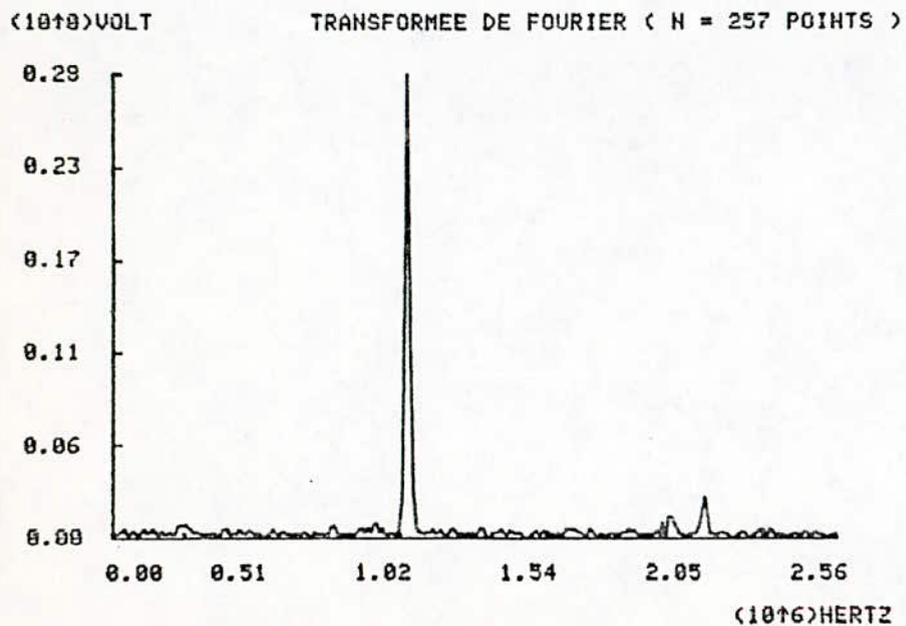
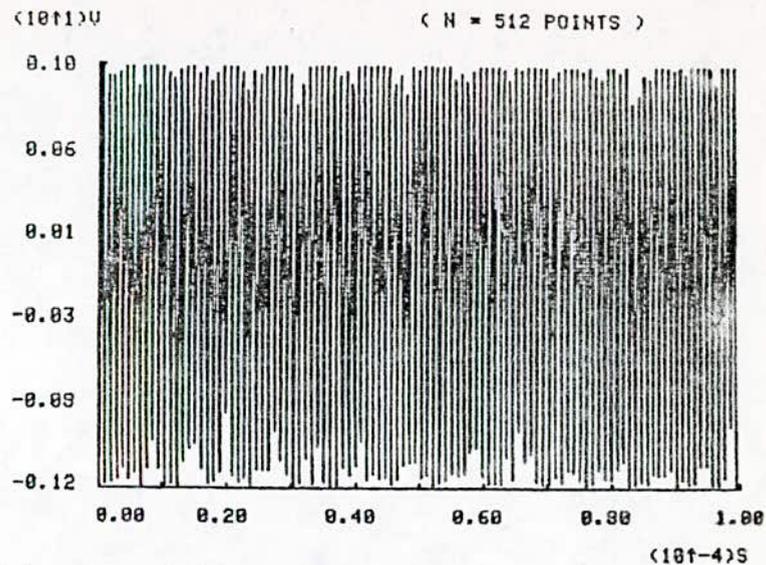
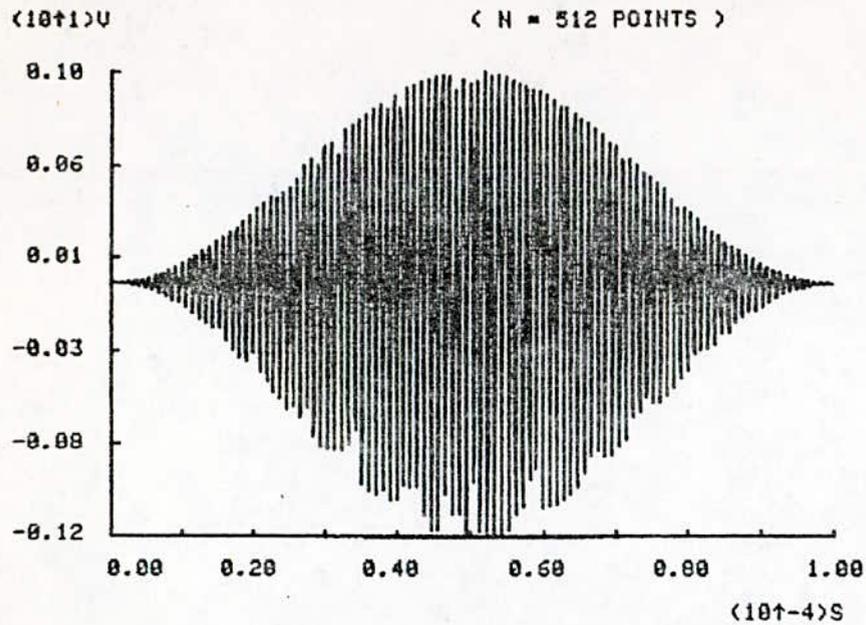
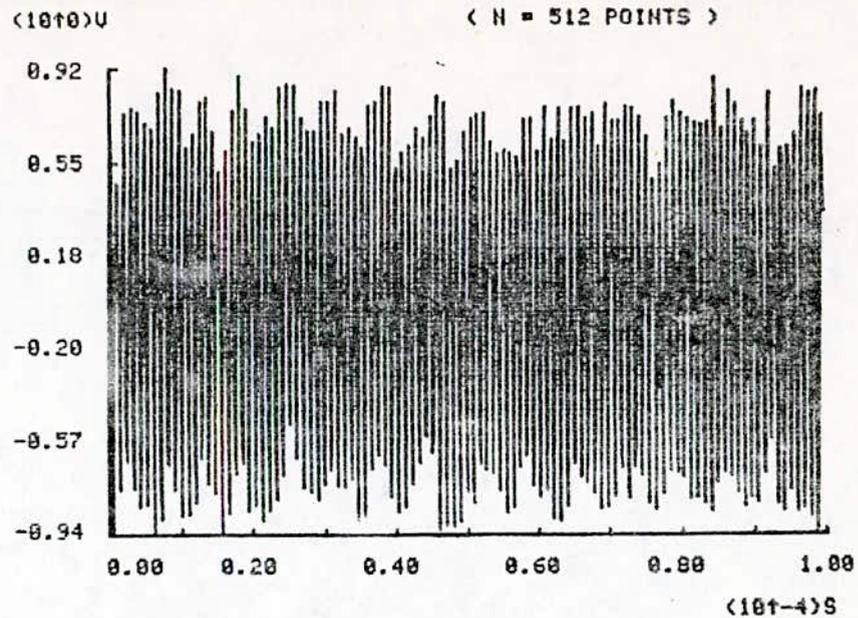
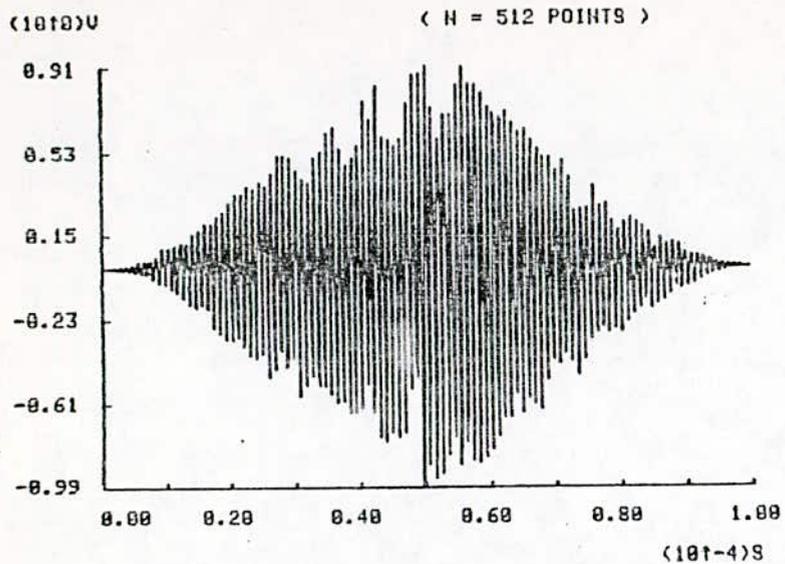


FIG VII 6.a

FIG VII 6.b



75

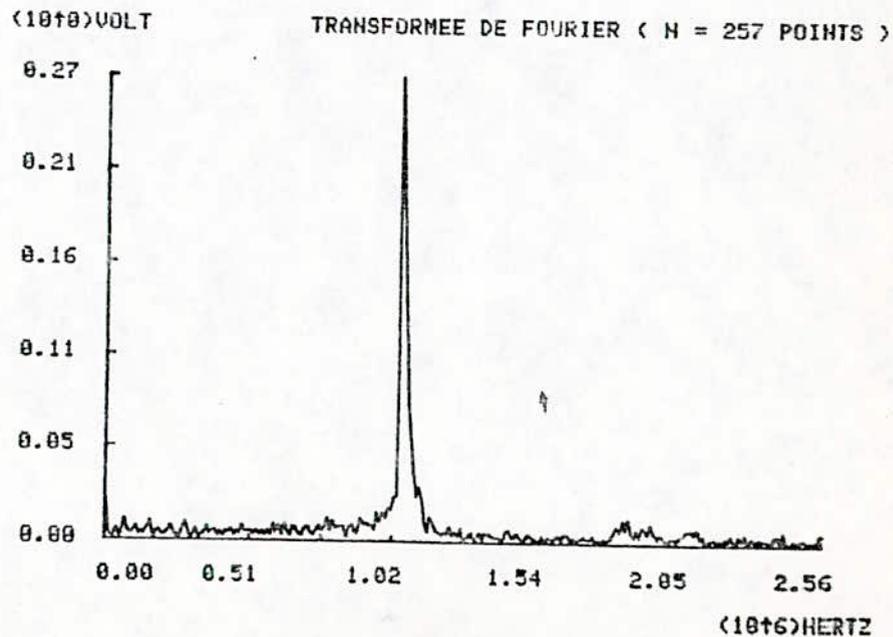
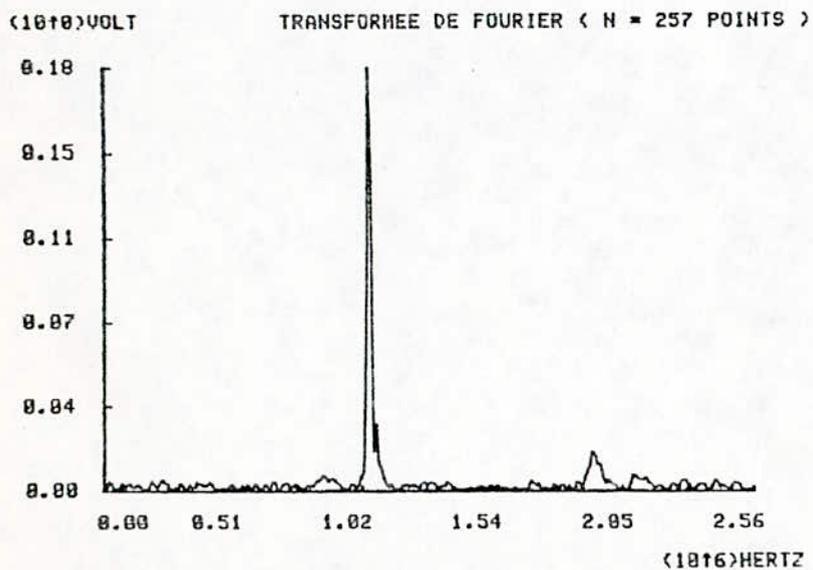
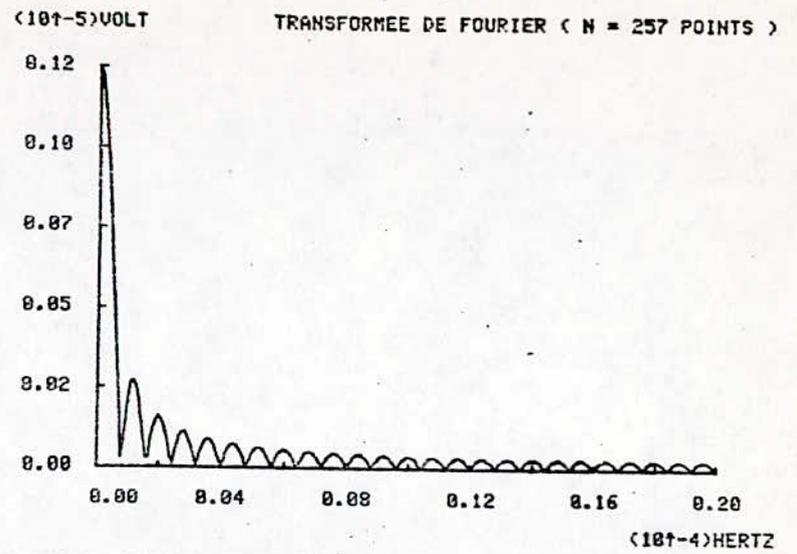
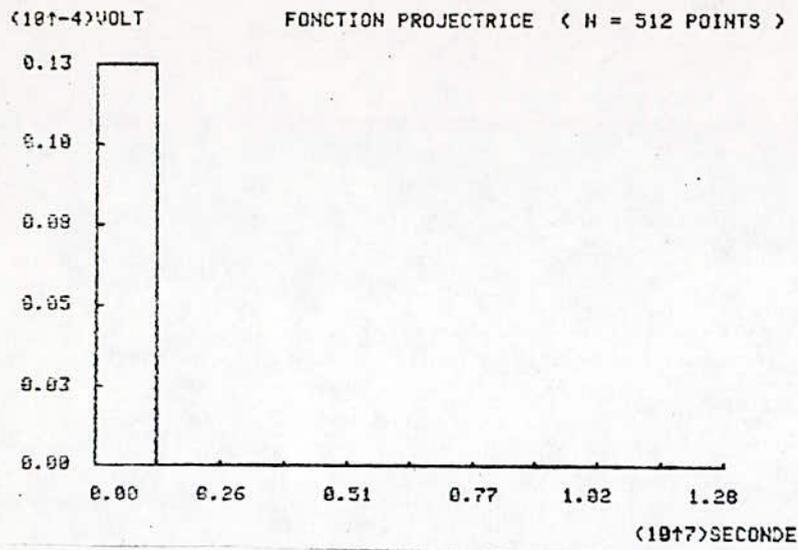


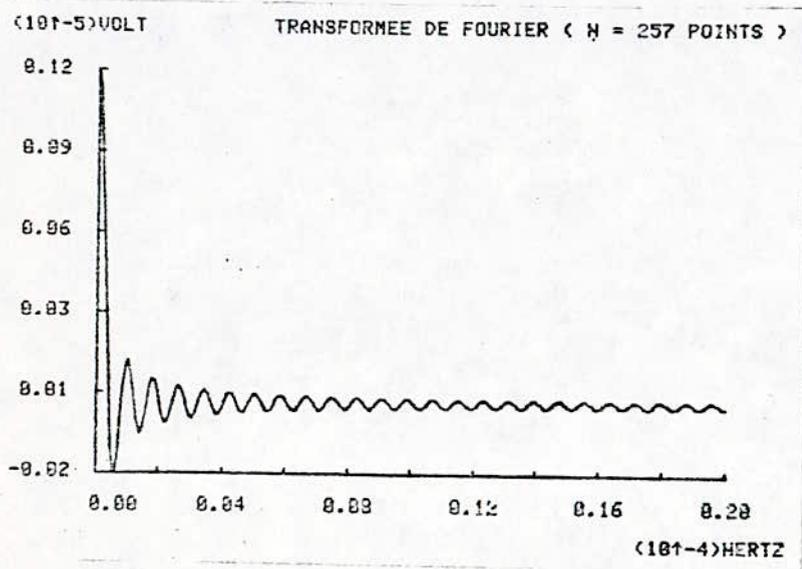
FIG VII 7.a

FIG VII 7.b

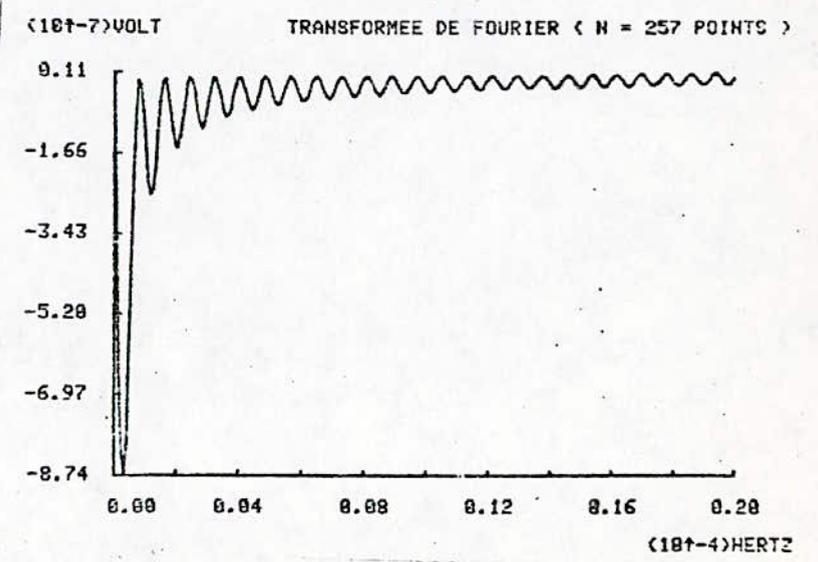


76

FONCTION PROJECTRICE

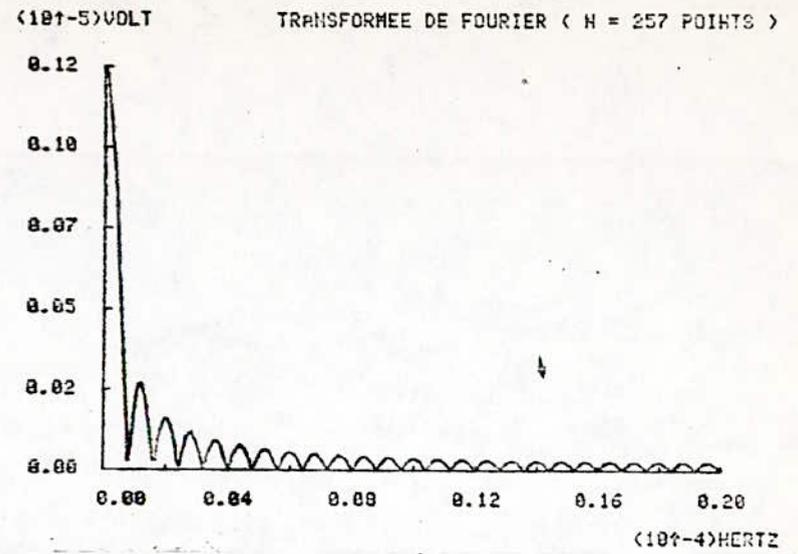
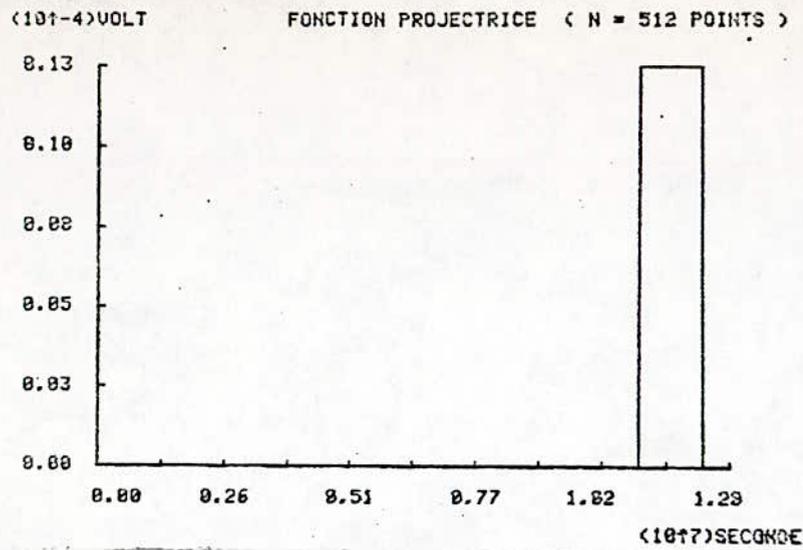


MODULE DE LA T.F



PARTIE-REELLE DE LA T.F.

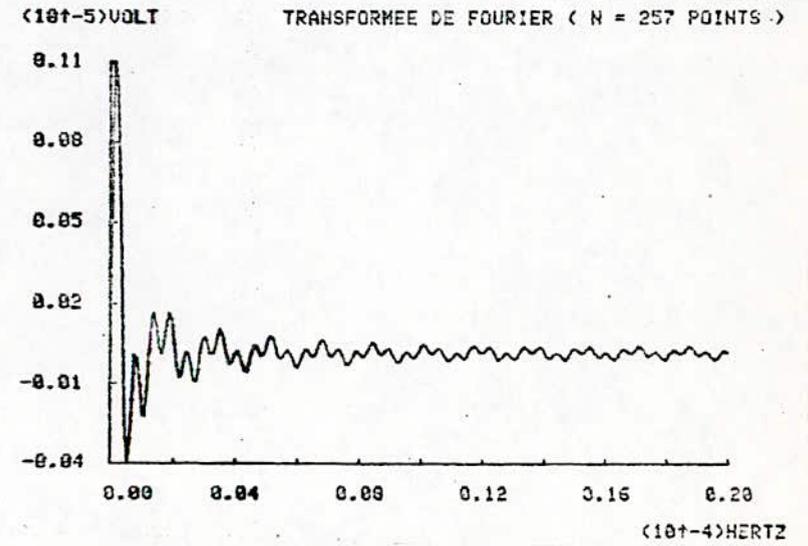
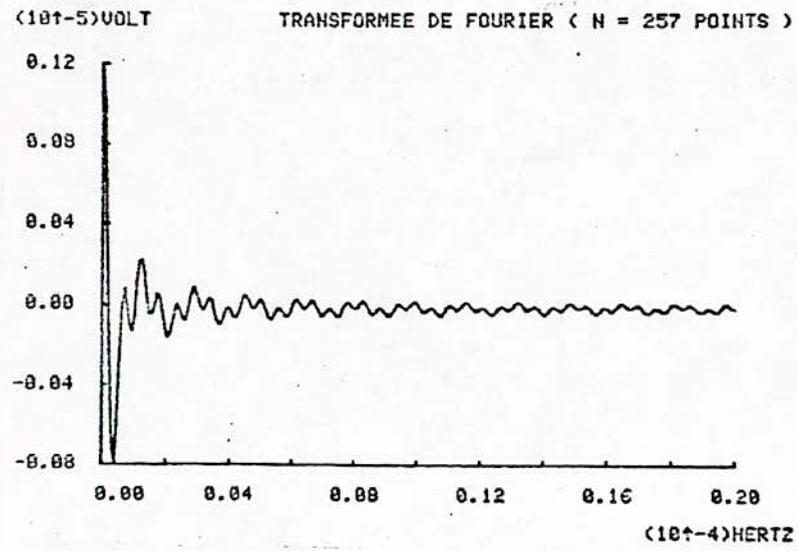
FIG VII.8a PARTIE IMAGINAIRE DE LA T.F



77

FONCTION PROJECTRICE

MODULE DE LA T.F



PARTIE REELLE DE LA T.F

FIG.VII 8-b

PARTIE IMAGINAIRE DE LA T.F

C H A P I T R E VIII

REGLES PRATIQUES POUR AMELIORER LES RESULTATS

OBTENUS PAR FFT

VIII.1 MOYENNAGE DU SIGNAL.

Ce procédé est utilisé pour éliminer le bruit additif:

Le bruit en lui-même est une fonction. Il possède, également une part dans le domaine des fréquences, et en raison de la propriété de linéarité, le bruit qui est ajouté à un signal dans le domaine temporel est aussi ajouté à la représentation fréquentielle du signal.

Le bruit est généralement aléatoire et possède ainsi une valeur moyenne qui tend vers Zéro sur un grand intervalle de temps.

Pour cette raison, des signaux répétitifs peuvent être acquis plusieurs fois et après moyennage le niveau de bruit est réduit.

Cette amélioration du rapport signal sur bruit, obtenue par moyennage du signal est proportionnelle à la racine carrée du nombre de signaux moyennés.

Quand ce nombre est exprimé en termes de puissance de deux, il correspond à une amélioration de 3 dB pour chaque puissance de deux.

EXEMPLE:

128 signaux moyennés peuvent être écrit 2^7 et correspondent à une amélioration de: $7 \times 3\text{dB} = 21\text{dB}$ rapport signal sur bruit.

Sur la figure (VIII.1.a) nous avons représenté la fonction Sinus qui est délivrée par un générateur à la fréquence de 1MHZ. Le Signal est visualisé sur l'écran de l'oscilloscope numérique, la vitesse de balayage est de 10^4M/s , la période d'échantillonnage $A1 = 1.953 \cdot 10^{-7} \text{Sec.}$

Les figures. VIII.1.b, VIII.1.C, VIII.1.d représentent respectivement la transformée de Fourier dans le cas où le moyennage est de 1, 10, 100.

REGLE GENERALE:

-Si le signal est peu bruiteux, un moyennage de 32 Signaux réduit le bruit faible niveau.

-Si le Signal est modérément bruiteux, 128 à 512 Signaux devraient donner une amélioration suffisante.

-Si le signal est purement noyé dans le bruit, beaucoup plus de signaux sont nécessaires pour une amélioration suffisante.

Mais plus le nombre de signaux est élevé plus le temps de calcul est élevé.

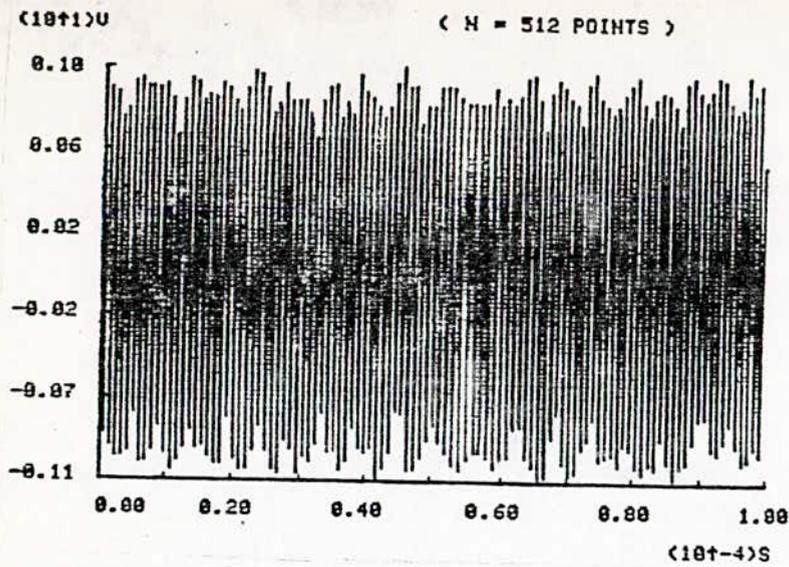


FIG VIII 1.a

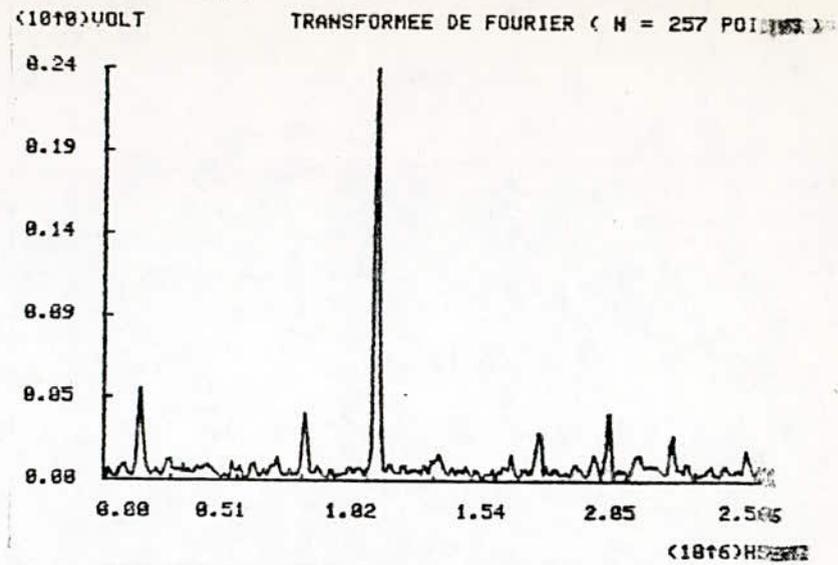


FIG VIII 1.b

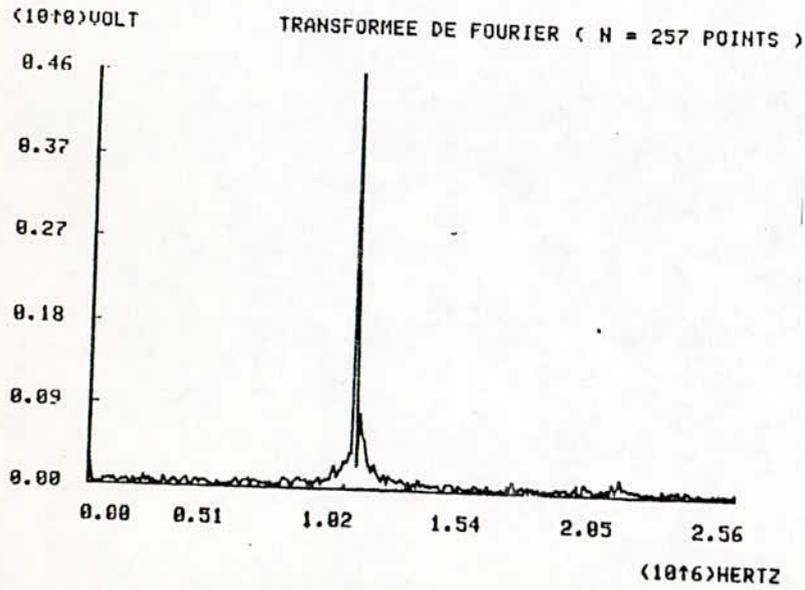


FIG VIII 1.c

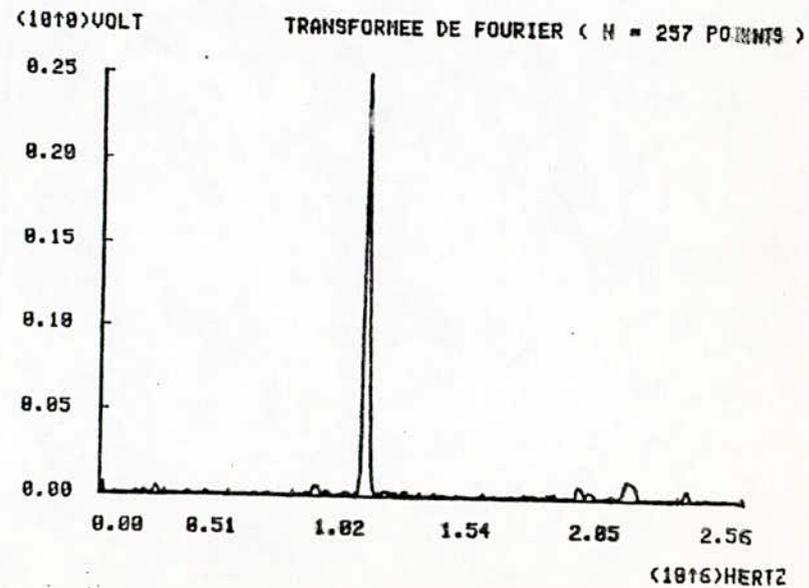


FIG VIII 1.d

VIII.2 ELEMINATION DE LA COMPOSANTE CONTINUE:

La suppression de la valeur moyenne améliore souvent la résolution en amplitude.

Certains signaux ont une valeur moyenne non nulle (ils ont une composante continue). Quelquefois les signaux sont acquis avec une composante continue, d'autres fois, la composante continue est une part indépendante de l'acquisition. Le premier cas apparaît lorsque le niveau de référence n'est pas fixé de manière soignée. Le second cas apparaît lorsque par exemple le signal est une sinusoïde échantillonnée sur une durée qui n'est pas égale à la période de la sinusoïde ni à un nombre entier de périodes.

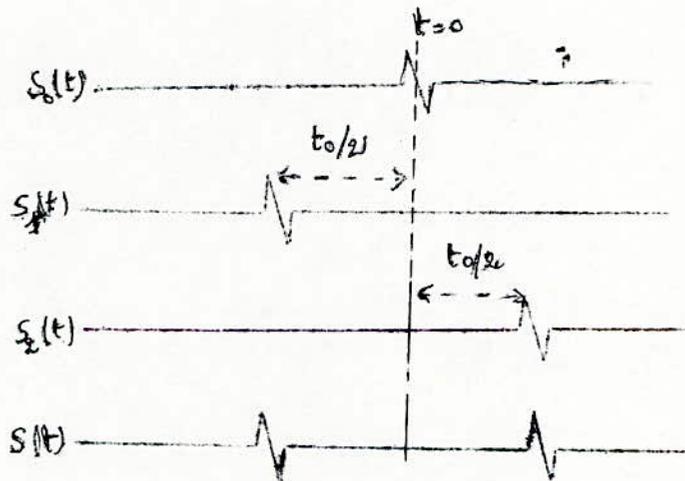
En raison des propriétés de linéarité, la composante continue est ajoutée à la fois dans le domaine temporel et dans le domaine fréquentiel, elle peut avoir une amplitude si élevée qu'elle masque les composantes fréquentielles du signal, pour éviter cela: on calcule la valeur moyenne du signal que l'on soustrait avant de calculer la FFT.

C O N C L U S I O N

Nous terminons ce travail par une application qui nous permet d'utiliser certaines propriétés de la T.F que nous avons étudiées dans ce projet:

-Application à la mesure du temps séparant deux impulsions.

Le signal $S(t)$ représente la somme des deux signaux $S_1(t)$ et $S_2(t)$



En raison de la propriété de linéarité de la T.F

on écrit:

$$F[S(t)] = F[S_1(t) + S_2(t)] = F[S_1(t)] + F[S_2(t)]$$

avec $S_1(t) = S_0(t + t_0)$

$S_2(t) = S_0(t - t_0)$

et en raison de la propriété de translation de la T.F

on a :

$$F [S_1(t)] = F [S_0(t+t_0)] = e^{j\pi ft_0} F [S_0(t)]$$

$$F [S_2(t)] = F [S_0(t-t_0)] = e^{-j\pi ft_0} F [S_0(t)]$$

d'où

$$\left| F [S(t)] \right| = 2 \cos \pi ft_0 \left| F [S_0(t)] \right|$$

Le terme multiplicatif $2 \cos \pi ft_0$ correspond à une modulation d'amplitude du spectre de $s_0(t)$. Cette amplitude est maximale pour : $\cos \pi ft_0 = 1 \implies \pi ft_0 = k \pi$ d'où

$$fk = \frac{k}{t_0} \quad \text{et} \quad f_{k+1} = \frac{k+1}{t_0}$$

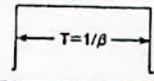
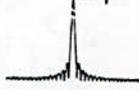
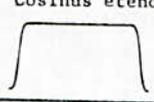
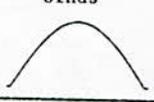
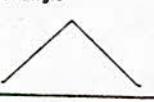
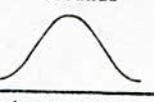
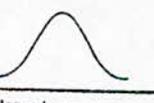
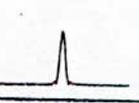
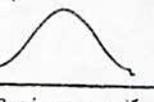
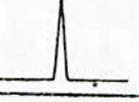
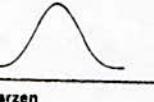
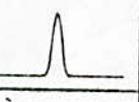
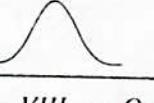
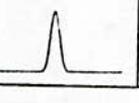
$$\Delta f = f_{k+1} - f_k = \frac{1}{t_0}$$

t_0 = représente la durée entre deux impulsions.

Si t_0 est très petit, sa détermination risque d'être entachée d'une incertitude trop importante, pour éviter cela on calcule Δf qui est grande dans ce cas donc bonne résolution dans le domaine des fréquences ensuite on calcule t_0 avec la relation $\Delta f = \frac{1}{T_0}$.

Ce procédé (dit de Spectroscopie ultrasons) est couramment utilisé en contrôle non destructif pour l'étude de la diffraction d'ondes ultrasonores par des bords de fissures.

ANNEXE

Amplitude	Equation	Représentation fréquentielle	Amplitude du lobe principal	Lobe secondaire (dB)	Bande passante (3 dB)	Atténuation hors bande (dB/Octave)
Rectangle 	$A=1$ for $t=0$ to T		T	-13.2	0.86β	6
Cosinus étendu 	$A=0.5(1-\cos 2\pi 5t/T)$ for $t=0$ to $T/10$ and $t=9T/10$ to T $A=1$ for $t=T/10$ to $9T/10$		$0.9 T$	-13.5	0.95β	18 (beyond 5β)
Sinus 	$A=\sin 2\pi 0.5t/T$ for $t=0$ to T		$0.64 T$	-22.4	1.15β	12
Triangle 	$A=2t/T$ for $t=0$ to $T/2$ $A=-2t/T + 2$ for $t=T/2$ to T		$0.5 T$	-26.7	1.27β	12
Cosinus 	$A=0.5(1-\cos 2\pi t/T)$ for $t=0$ to T		$0.5 T$	-31.8	1.39β	18
Sinus cube 	$A=\sin^3 2\pi 0.5t/T$ for $t=0$ to T		$0.42 T$	-39.8	1.61β	24
Hamming 	$A=0.08 + 0.46(1-\cos 2\pi t/T)$ for $t=0$ to T		$0.54 T$	-41.9	1.26β	6 (Beyond 5β)
Cosinus carré 	$A=(0.5(1-\cos 2\pi t/T))^2$ for $t=0$ to T		$0.36 T$	-48.9	1.79β	30
Parzen 	$A=1-6(2t/T-1)^2+6(2t/T-1)^3$ for $t=T/4$ to $3T/4$ $A=2(1- 2t/T-1)^3$ for $t=0$ to $T/4$ and $t=3T/4$ to T		$0.37 T$	-53.2	1.81β	24

1754 79

Fig. VIII. — Quelques types de fenêtres.

--o=BIBLIOGRAPHIE=o--

- [1] -Programmation basic théorie et applications
serie Schaum - Byrons. GOTTFRIED.
- [2] -Introduction to programing in basic
(TEKTRONIX 1975)
- [3] -Introduction to graphic programing in Basic
(TEKTRONIX 1975)
- [4] -Méthodes et techniques de traitement du signal
et applications aux mesures physiques.
J.MAX (2° édition - Masson)
- [5] -G.GASTAUD, la transformée de Fourier, Mesures-
Régulations - Automatisme. (Mars 1979), PP 87-94