

الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

25/86

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

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT ELECTRONIQUE

المدسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

PROJET DE FIN D'ETUDES

SUJET

*Résonance Magnétique Nucléaire
Réalisation d'un interface avec un CAN
et développement d'un micro-programme
pour un spectromètre RMN*

Proposé par :

Dr. PUVVADA Ramesh

Etudié par

REMIDI Kamel

ZIANI Foudil

Dirigé par :

Dr. PUVVADA Ramesh

Promotion : Janvier 1986

E.N.P 10, Avenue Hacem Badi — EL-HARRACH — ALGER

الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

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

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT ELECTRONIQUE

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

PROJET DE FIN D'ETUDES

SUJET

*Résonance Magnétique Nucléaire
Réalisation d'un interface avec un CAN
et développement d'un micro-programme
pour un spectromètre RMN*

Proposé par :
Dr. PUVVADA Ramesh

Etudié par
REMIDI Kamel
ZIANI Foudil

Dirigé par :
Dr. PUVVADA Ramesh

Promotion : Janvier 1986

E.N.P 10, Avenue Hacén Badi — EL-HARRACH — ALGER

\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
-O. REMERCIEMENT. O-
\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$

Nous tenons à exprimer nos sincères
remerciements à M^r PUVVADA RAMESH notre
promoteur pour ces conseils et pour toute
l'aide durant notre travail.

Nous remercions également MM.
SADOUN et ZERGUI assistants à l'école
nationale polytechnique pour leur aide
précieuse.

Que toutes les personnes qui nous
ont aidé de près ou de loin dans notre
formation trouvent ici notre profonde
reconnaissance.

المدرسة الوطنية المتعددة التقنيات
المكتبة — BIBLIOTHEQUE
Ecole Nationale Polytechnique

\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$

CHAPITRE I / SPECTROMETRE R.M.N

A - Resonance Magnétique Nucléaire .

- 1 - Introduction
- 2 - Magnétisme nucléaire
- 3 - Condition de la resonance

B - Description du spectromètre R.M.N

- 1 - Schéma synoptique du spectromètre R.M.N
- 2 - Principe de fonctionnement
- 3 - Description sommaire des différents étages.
 - 3-1 L'electroaimant
 - 3-2 Le programmeur d'impulsion
 - 3-3 Système émetteur
 - 3-4 Bobine d'excitation
 - 3-5 Système recepneur
 - 3-6 Système de visualisation

C - Mesure du temps de relaxation

- 1 - Mesure du temps T_1
- 2 - Mesure du temps T_2

CHAPITRE II / SYSTEME DE CONVERSION ANALOGIQUE NUMERIQUE

A - Généralités sur les C.A.N

- 1 - Definition d'un C.A.N
- 2 - Classification des C.A.N

B - Etude de l'A.D.C 802

- 1 - Definition
- 2 - Caracteristiques
- 3 - Principe de fonctionnement
- 4 - Brochage
- 5 - Test

A- Etude du Kit D5 6802 M.E.K de Motorola

1-Description du Kit

- 1-1 Le MPU 6802
- 1-2 Système "PIA" (U_2)
- 1-3 Clavier (FEYPAD)
- 1-4 Les mémoires mortes (ROM)
- 1-5 Les mémoires vives (RAM)
- 1-6 Décodeur d'adresse (U_6)
- 1-7 Interface série
- 1-8 Bufferts (U_1 ; U_2 ; U_3 en option)
- 1-9 Interface parallèle programmable (6821 U_9)



2-Répartition des adresses sur la carte

3-Etude de l'interface pallèle programmable du
6802 de Motorola

- 3-1 Présentation du PIA 6821
- 3-2 Adressage du PIA
- 3-3 Programmation du PIA
- 3-4 Programmation des lignes de dialogues
CA2 et CB2 en sortie

B- Calcul de la constante de temps de l'exponentièl

1-Position du probleme

2-PRINCIPE DE linéarisatio de la courbe $M_0(t)$

3-Calcul de l'équation générale donnant t en fonction
de n et T

4-Calcul de l'erreur t introduite par la
linéarisation

C-Realisation de la carte d'interfaçage

- 1-Schéma du montage
- 2-Circuit imprimé
- 3-Essais

D-Developpement de la partie SOFT

- 1-Acquisition des données
- 2-Programme pour la mesure du temps T1

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

INTRODUCTION

CHAPITRE 1

A/ LA RESONNANCE MAGNETIQUE NUCLEAIRE: (R.M.N.)

1-Introduction: La R.M.N. est actuellement un puissant moyen d'identification et de connaissance des structures moléculaires et un bon outil d'étude de structures cristallines. Ses applications en biochimie sont en plein développement et depuis quelques années des tomographes à R.M.N pénètrent progressivement dans des départements d'imagerie médicale des hopitaux.

Sa découverte est cependant relativement récente, elle fut publiée simultanément en 1946 par deux groupes de chercheurs PERCELL, TORREY, POUND d'une part, et BLOCH, HANEN, PACKARD de l'autre part.

La R.M.N a pour avantage de ne pas introduire dans l'organisme des produits de contrastes et de ne pas soumettre le corps humain à des radiations, seul l'application d'un champ magnétique autour du corps humain est une contrainte dont les risques semblent être négligeables

2-Magnétisme Nucléaire: Un atome est constitué d'un noyau, assemblage de protons chargés positivement et de neutrons de charges nulles et d'un nuage d'électrons chargés négativement.

Si on prend comme exemple un atome d'H₂ (d'hydrogène), il est constitué d'un noyau comportant un proton et un électron qui gravite autour du noyau.

L'électron subit deux mouvements de rotations, l'un autour de lui même et l'autre autour du noyau (spin-électron).

Cette hypothèse est annoncée en 1925 par SELFESS .

Tandis que le proton subit un seul mouvement circulaire autour de lui même (spin-spin) et par conséquent il induit un courant et ce courant crée un moment magnétique.

$$\vec{M} = \gamma \vec{\sigma}$$

AVEC

$$\vec{\sigma} = \hbar \vec{I}$$

M : moment magnétique des atomes ou des molécules

σ : moment cinétique des atomes ou des molécules

γ : c'est une constante appelée rapport gyromagnétique

On peut classer les noyaux en trois (3) catégories.

a)- Les noyaux dont le nombre quantique de spin $I=0$ donc ne possédant pas de moment cinétique par conséquent pas de moment magnétique et pas de moment électrique, nul ($eQ=0$)

A cette catégorie appartiennent ^{les noyaux} de nombre de masse A et de nombre de charges paires ex : $^{12}_6\text{C}$, $^{16}_8\text{O}$...

b)- Les noyaux dont le nombre quantique de spin $I=\frac{1}{2}$ donc possédant un moment cinétique par conséquent un moment magnétique mais le moment électrique est égal à zéro ($eQ=0$) .

C'est le cas qui nous intéresse pour avoir la résonance. ex : ^1H , ^{13}C , ^{17}F , ^{31}P ...

c)- Le noyau dont le nombre quantique de spin est supérieur à $\frac{1}{2}$ ($I > \frac{1}{2}$) donc il existe un moment cinétique et un moment magnétique et un moment électrique $eQ \neq 0$.

ex : $^{14}_7\text{N}$, $^{17}_8\text{O}$, $^{11}_5\text{B}$, $^{10}_5\text{B}$...

Ce cas est très compliqué .

3- condition de résonance:

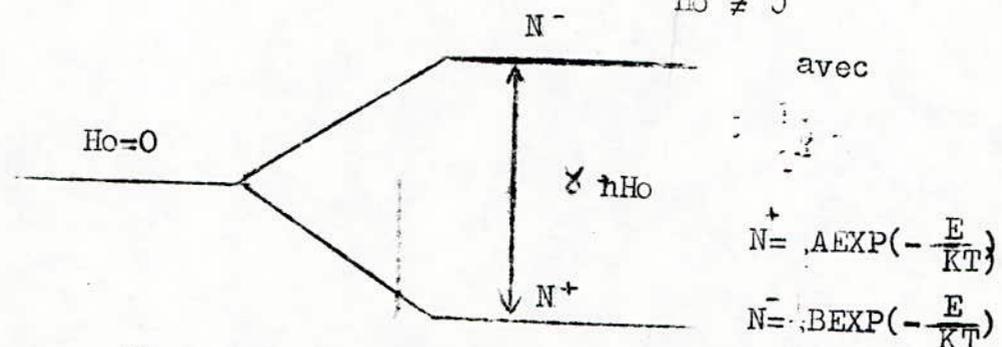
Les noyaux des atomes d' H_2 (d'hydrogène) par exemple possédant des moments magnétiques dont les directions sont aléatoires.

Avant l'application du champ H_0 , le moment résultant est $M_r=0$, Après l'application du champ magnétique H_0 , le moment résultant étant M_0 qui est colinéaire avec H_0 .

Dans ce cas le niveau d'énergie est donnée par la formule suivante:

$$N = 2I + 1, \quad I = \frac{1}{2} \Rightarrow N = 2$$

$H_0 \neq 0$



Si on excite l'échantillon par un signal radiofréquence on a des transitions de protons entre ces deux niveaux.

Ces transitions caractérisées par des rayonnements d'énergie $h\nu_0$ avec ν_0 : fréquence des signaux radiofréquences pour avoir la résonance il faut que l'écart d'énergie entre ces deux sous niveaux vaut $h\nu_0$ d'où:

$$\gamma h H_0 = h\nu_0 \quad \text{avec } h = \frac{h}{2\pi}$$

$$\gamma \frac{h}{2\pi} H_0 = h\nu_0 \Rightarrow \gamma H_0 = 2\pi \nu_0$$

On fait varier H_0 à l'aide du générateur de courant qui alimente l'électroaimant pour satisfaire la condition de résonance $\gamma H_0 = 2\pi \nu_0$

Le coef. γ = Constante
 Le ; $\nu_0 = 40\text{MHz}$ fixe $\Rightarrow H_0 = \text{cst.}$

B/: Description du Spectromètre R.M.N

1- Schéma synoptique (voir le schéma correspondant) de la figure 1)

Le spectromètre R.M.N à impulsions est constitué par :

- L'électroaimant.
- Programmeur d'impulsions.
- Système émetteur par impulsions.
- Une bobine d'excitation radio-fréquence
- Un système récepteur
- Un système de visualisation.

2- Principe de fonctionnement

Le programmeur d'impulsions génère des séquences d'impulsions $\frac{\pi}{2} - \tau - \frac{\pi}{2}$ et $\frac{\pi}{2} - \tau - \pi$.

Le système émetteur a pour rôle de produire des salves : ~~salves~~ d'un signal radiofréquence sous l'action d'impulsions venant du programmeur d'impulsions.

Ces salves vont exciter une bobine pour introduire un champ magnétique tournant H_1 à la fréquence du signal radio-fréquence.

L'échantion ^I étudié est inséré à l'intérieur de la bobine d'excitation qui elle-même placée entre les deux pôles de l'électroaimant. L'électroaimant sert à produire le champ statique H_0 .

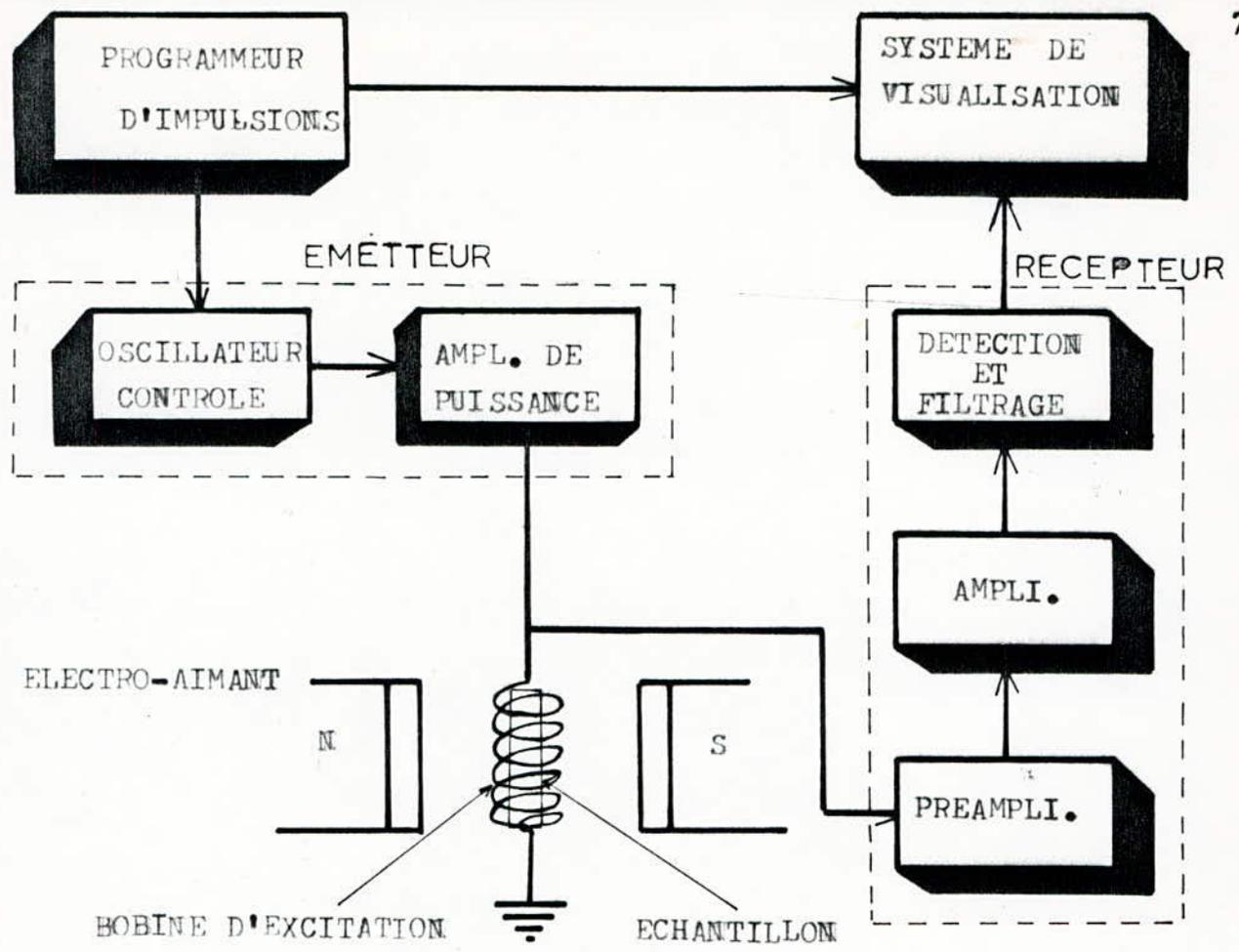


Fig. 1 SCHEMA SYNOPTIQUE DU SPECTROMETRE RMN

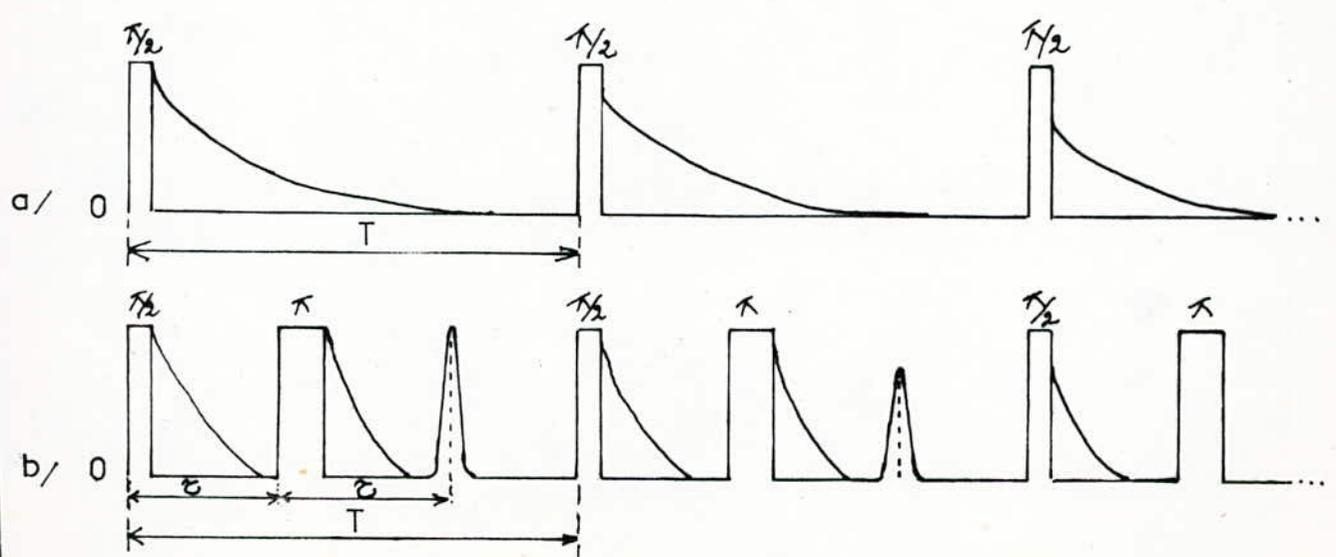


Fig. 2 LES SIGNAUX A LA SORTIE DU RECEPTEUR

Pendant la durée des salves l'échantillon va être excité et son vecteur aimantation subit des fractions de rotations.

A l'instant où H_1 est coupé, c'est-à-dire à la fin de l'impulsion l'aimantation du réseau va revenir à sa position d'origine.

D'après le phénomène de précession libre et d'après la loi de LENZ ce phénomène crée une force électro-motrice (f-e-m) induite dans la bobine d'excitation .

Ce signal recueilli et appliqué dans le récepteur, va être traité et filtré dans un détecteur. Il pourra être visualisé sur l'écran d'un oscilloscope et constitue une information sur le régime transitoire d'excitation. L'étude des courbes obtenues nous permet de mesurer les temps de relaxation T_1 et T_2 . Ces courbes sont données par la figure 2 .

3- Description sommaire des différents étages

1)- L'électroaimant: Il a pour rôle de produire le champ magnétique statique, qui est constitué d'un aimant sur lequel est enroulé un nombre N de spires. On utilise une alimentation stabilisée en courant pour alimenter l'électroaimant afin d'empêcher la dérive dans le temps de l'intensité du champ, et assurer une bonne stabilité dans le temps. Elle devra délivrer des courants allant jusqu'à 15A environ pour avoir un champ assez élevé.

Le champ produit par l'électroaimant doit être:

- Assez élevé afin d'avoir une meilleure sensibilité, car la tension induite est proportionnelle à H_0^2 donc la sensibilité est d'autant plus grande que le champ est très élevé

$$V = W\bar{Q}_0 \quad ; \quad \bar{Q}_0 = KM_0$$

$f =$ fréquence du champ tournant $\omega = 2\pi f$

$M_0 =$ l'aimantation d'équilibre

$K =$ facteur de proportionnalité .

Or la condition de résonance impose que $\omega = \omega_0$, et on sait que $\omega_0 = \gamma H_0$ et de plus on sait que l'aimantation d'équilibre M_0 est proportionnelle à H_0 . $M_0 = \chi_0 H_0$

où χ_0 : susceptibilité nucléaire statique elle dépend du noyau, d'où

$$V = \frac{\gamma H_0 \times K \chi_0 H_0}{\omega_0} = K_1 H_0^2$$

Cependant la méthode de mesure des temps de relaxation est très précise et exige un champ H_0 le plus homogène possible dans l'intervalle de variation.

L'homogénéité doit être assurée dans l'espace et dans le temps. Pour tout cela on utilise des aimants dont le diamètre des pièces polaires est compris entre 150 et 250 mm avec un entrefer de 15 à 25 mm. En plus, ces pièces sont garnies de calottes ajustable, Pour assurer un bon parallélisme qui est très important pour assurer une bonne homogénéité spatiale du champ magnétique.

Les pièces polaires ont un diamètre de 150 mm.

L'électroaimant est maintenu à une température constante grâce à un système de refroidissement.

2)- Programmeur d'impulsions.

Il est capable de générer des séquences d'impulsions $\frac{\pi}{2} - T - \frac{\pi}{2}$ et $\frac{\pi}{2} - \tau - \pi$.

Ces impulsions vont commander un oscillateur contrôlé et servent aussi au déclenchement du balayage de l'oscilloscope pour la synchronisation. Et comme l'oscilloscope est remplacé par un système de conversion analogique-numérique ces impulsions servent à la synchronisation du ⁿlancement de conversion avec le signal à convertir.

Le programmeur d'impulsions doit permettre le réglage de la durée des impulsions, la durée qui sépare les impulsions ainsi que la durée de la séquence. Ces réglages doivent être séparés.

3)- Le Système Emetteur:

Il délivre des salves de signal sinusoïdal radiofréquence qui excitent l'échantillon et provoquent le phénomène de R.M.N (résonance magnétique nucléaire)

Il est constitué:

- d'un oscillateur commandé par les impulsions venant du programmeur.
- D'un amplificateur de puissance qui fournit la puissance nécessaire à ces salves pour obtenir le champ tournant à l'aide de la bobine d'excitation. Lorsque l'émetteur est en marche la bobine joue le rôle d'excitatrice et lorsqu'il est fermé elle joue le rôle de réceptrice et pendant ce temps aucun autre signal n'est provoqué par l'échantillon n'y soit induit dans la bobine. Le système émetteur doit présenter une stabilité en amplitude et en fréquence.

4)- Bobine d'excitation:

Elle est couplée à la sortie de l'émetteur et à l'entrée du récepteur, son rôle est de générer le champ magnétique tournant H_1 et de recueillir le signal correspondant au phénomène de relaxation. Pendant l'état ON des salves radiofréquence (RF) la bobine produit le champ H_1 qui va exciter l'échantillon et elle est donc reliée à l'émetteur pendant l'état OFF des salves RF,

5)- Système Récepteur:

L'importance de la R.M.N réside dans l'analyse du signal induit par l'échantillon car c'est ce signal qui donne des informations sur l'échantillon à étudier.

Aussi la réception de ce signal est d'importance capitale. Le signal induit par l'échantillon est assez faible (de l'ordre de quelques dizaines de microvolts). Il est nécessaire d'amplifier le signal reçu. L'amplification d'un signal d'entrée ne se fait généralement pas une seule fois mais elle nécessite plusieurs étages reliés entre eux par des organes de liaisons.

On a intérêt à ce que la facteur de qualité de la bobine réceptrice demeure le plus élevé possible afin d'avoir un rapport signal sur bruit optimal, pour cela il faut utiliser un préamplificateur à impédance d'entrée élevée. Pendant l'état OFF le récepteur doit passer à son fonctionnement normal le plus vite possible (quelques microsecondes).

Les signaux issus à la sortie du récepteur sont donnés par la Figure. 2

6)- Système de Visualisation:

On utilise le plus souvent un oscilloscope à balayage déclenchable par les impulsions du programmeur. On peut par une mesure de la pente de la courbe visualisée obtenir les valeurs des temps de relaxation. Néanmoins cette méthode est améliorée en utilisant un système de conversion analogique-numérique.

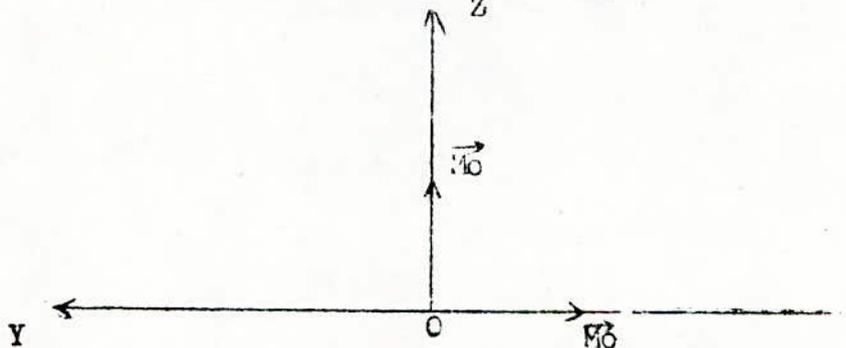
Le but de notre travail est de remplacer ce système de visualisation par un convertisseur analogique numérique (C.A.N) afin de traiter l'information recueillie par un ordinateur et de la stocker on aura une méthode de mesure systématique.

C/ Mesure Du Temps De Relaxation

1)- Mesure Du Temps De Relaxation T_1

Cette mesure est basée sur le phénomène de précession libre de l'aimantation, après une impulsion d'excitation on utilise la séquence périodique, d'impulsion $\frac{\pi}{2}$ de période T. A l'origine le vecteur M est à l'équilibre tel que $M_x = M_y = 0$; et $M_z = M_0$. Une première impulsion $\frac{\pi}{2}$ amène l'aimantation en question sur l'axe OY tel que

$$M_x = M_z = 0 \quad \text{et} \quad M_y = M_0$$

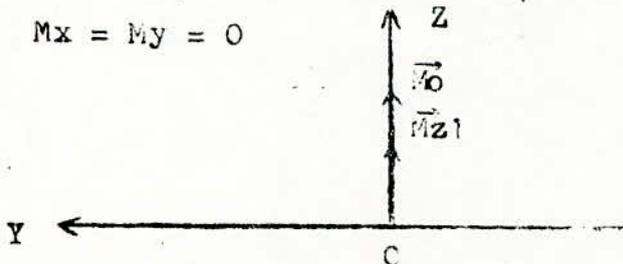


A la fin de cette impulsion le mouvement de précession libre amorcée autour de OZ aura la loi $M_z(t) = M_0 (1 - e^{-\frac{t}{T_1}})$

Après un certain temps T, on aura une aimantation M dirigée suivant Oz tel que

$$M_z = M_{z_1} = M_0 (1 - e^{-\frac{T}{T_1}}) \quad \text{avec} \quad M_{z_1} < M_0$$

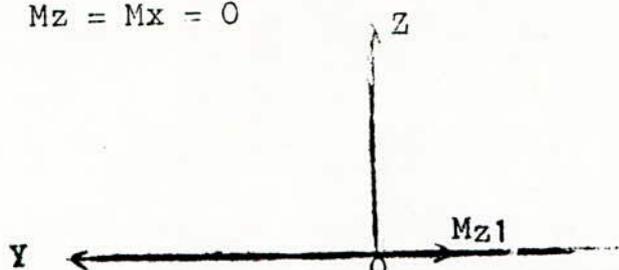
$$M_x = M_y = 0$$



A cet instant, la deuxième impulsion amène le vecteur M_{z_1} sur l'axe OY on aura.

$$M_y = -M_z, = -M_0 \left(1 - e^{-t/T_1} \right)$$

$$M_z = M_x = 0$$



Le retour à l'équilibre de l'aimantation est régi par la loi :

$$M_z(t) = M_{z_1} \left(1 - e^{-\frac{t}{T_1}} \right) \text{ après un certain temps } T \text{ on aura :}$$

$$M_z = M_{z_1} \left(1 - e^{-\frac{T}{T_1}} \right)$$

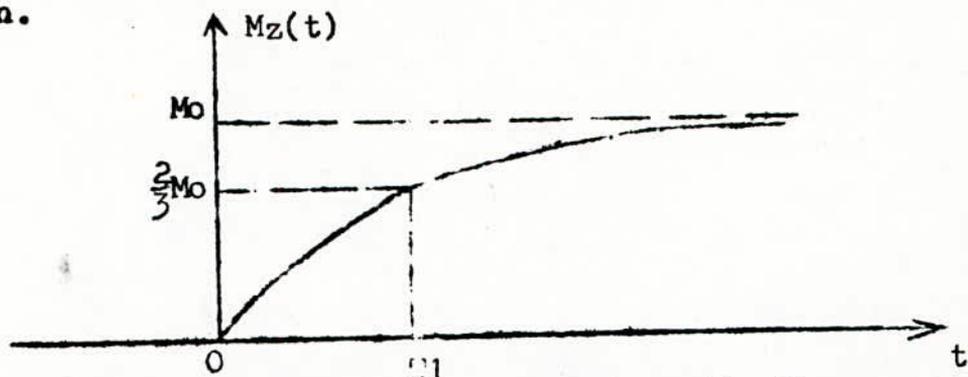
$$M_{z_1}(t) = M_0 \left(1 - e^{-T/T_1} \right)^2$$

$$\text{d'où } M_z(t) = M_{z_2}(t) = M_0 \left(1 - e^{-T/T_1} \right)^2$$

Ainsi pour l'impulsion d'ordre n , la valeur de la composante longitudinale de l'aimantation après ce mouvement de precession libre est $M_n = M_0 \left(1 - e^{-T/T_1} \right)^n$

On voit que chaque quantité diminue à chaque nouvelle

impulsion.



On constate que la répartition de cette figure est obtenue après un certain temps de l'application du champ \vec{H}_0 .

La constante de temps de cette courbe détermine le temps de relaxation T_1 c'est un temps entre spin-réseau.

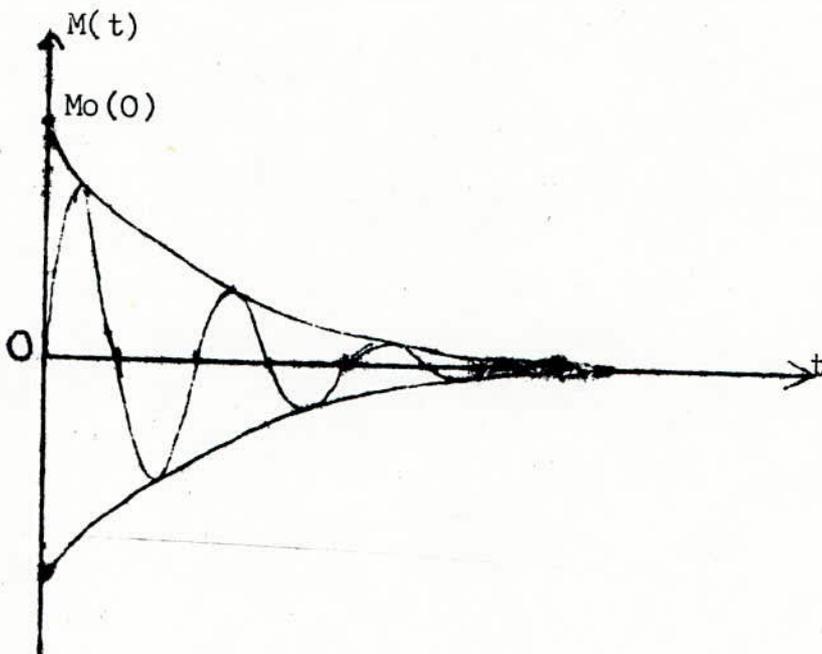
Dans notre cas la bobine réceptrice est placée de façon à recueillir le flux produit par la composante transversale M_T de l'aimantation globale. Comme la variation de cette composante est de la forme: $M_T = M_{T0} e^{-t/T_1} \cos \omega t$

Ou M_{T0} est la valeur initiale de cette composante.
 A chaque impulsion d'ordre $n+1$, cette amplitude initiale de M_T est exactement la valeur à l'équilibre de la composante M_Z après la n ème impulsions.

$$M_{T0}(n+1) = M_{Zn} = M_0 \left(1 - e^{-\frac{T}{T_1}}\right)^n$$

Le détecteur va visualiser sur l'oscilloscope l'enveloppe de la courbe précédente correspondante à la variation de

M_T .



L'amplitude initiale de chaque mouvement de precession diminue à chaque nouvelle impulsion.

Voir figure 2a.

2) Mesure du temps de relaxation T_2 .

Si on applique une impulsion $\frac{\pi}{2}$ à l'aimantation d'équilibre le mouvement de précession libre induit une variation de la composante transversale de l'aimantation tel que :

$$M_T = M_0 \exp\left(-\frac{t}{T_2}\right) \cdot \cos \omega t$$

Ainsi la décroissance de ce signal nous donnerait T_2 . En réalité celui-ci sera très perturbé du fait des interactions entre les spins et la bobine de réception.

Pour remédier à ce genre de problèmes, on utilise la technique d'écho de spin.

Une séquence d'impulsion $\frac{\pi}{2} - \tau - \pi$ produirait un écho aux temps 2τ conformément à ce qui a été dit précédemment.

A l'instant $t=T$ une nouvelle séquence provoque un deuxième écho à $t=T+2\tau$ et ainsi de suite... etc (Voir 2b).

Conclusion: Ces 2 temps de relaxation T_1 et T_2 sont calculés par un programme qui sera donné dans le chapitre III.

CHAPITRE 2

* Systeme De Conversion Analogique numérique:A/- Généralités sur les C.A.N

1- Définition: Effectuer une conversion analogique numérique c'est rechercher une expression numérique dans un code déterminé pour représenter une information analogique avec une précision et une résolution donnée, pour faire la conversion il faut disposer d'une référence, la référence étant l'unité de mesure (l'unité à laquelle on compare la donnée à convertir).

Pour un C.A.N électronique, la référence étant une tension de valeur connue V_{ref} , En présence d'une grandeur analogique, le convertisseur va calculer le rapport $\frac{A}{V_{ref}} = N$ ou V_{ref} représente l'étalon ce qui revient à effectuer une division.

Elle se fait suivant les puissances décroissantes de deux (2) Ceci dans le but d'avoir le code binaire, La grandeur analogique A à convertir s'écrira alors:

$$A = V_{ref} \left(\frac{b_1}{2} + \frac{b_2}{2^2} + \dots + \frac{b_n}{2^n} + \frac{b_{n+1}}{2^{n+1}} \right)$$

La longueur du mot binaire se trouve limitée à n bits compte-tenu de la résolution du système on écrira alors.

$$A = V_{ref} \left(\frac{b_1}{2} + \frac{b_2}{2^2} + \dots + \frac{b_n}{2^n} \right)$$

Les termes $(b_{n+1} ; b_{n+2})$

que l'on néglige représentent l'erreur de conversion appelée aussi erreur de quantification.

De ceci on déduit que la conversion analogique numérique est une opération de quantification qui consiste

à remplacer la grandeur analogique A par une grandeur discrète multiple d'une quantité élémentaire $\frac{V_{ref}}{2^n}$ appelé quantum. On définit trois caractéristiques importantes pour le CAN.

- La résolution: Elle représente le nombre de bits utilisé par le CAN, elle définit la plus petite variation de tension perçue par le convertisseur (Quantum).

- Le temps de conversion: C'est le temps nécessaire au convertisseur pour donner la valeur numérique correspondant au signal analogique appliqué (échantillon). Il comprend un temps correspondant à l'initialisation préalable du convertisseur et un temps correspondant à l'exécution de la conversion.

- La précision: C'est l'écart entre la caractéristique de transfert d'un CAN idéal et celle d'un C.A.N réel. Cet écart est généralement exprimé en nombre de LSB

$$1 \text{ LSB} = \frac{V_{ref}}{2^n}$$

2- Classification Des CAN:

On distingue deux familles de CAN les CAN analogiques et les CAN logiques.

- Les CAN analogiques: Il existe les CAN à rampe, double rampe etc... les principes utilisés sont analogiques, transfert de charge, générateur de tension etc...
- Les CAN logiques: Les CAN logiques les plus utilisés et les plus répandus sont les CAN parallèles et les CAN à approximation successives. On les appelle CAN logique car les composants prépondérants utilisés sont logiques.

Le convertisseur qui nous intéresse pour notre réalisation de l'interface appartient à la famille des CAN logiques.

C'est le CAN à approximations successives c'est le plus populaire de tous les CAN il constitue un excellent choix par suite de ces performances très intéressantes et de son prix relativement élevé.

B/- Etude de L'ADC 802

1- Définition: L'ADC 802 est un circuit CMOS travaillant par approximations successives. Son alimentation est de + 5V. Il est compatible avec la logique du MPU.

2- Caractéristique de L'ADC 802

- Une résolution $n = 8$ bits
- Une tension de référence $U_{ref} = + 5V$
- Une tension Analogique en entrée V_{in} de 0 à 5V
- Un temps d'accès $t_{acc} = 135 \mu s$

La tension de l'echelon correspondant au bit du poids le plus faible (LSB) sera:

$$e = \frac{U_{ref}}{2^n} = \frac{U_{ref}}{2^8} = \frac{5}{256} = 19,53 \text{ mV}$$

$$e = 19,53 \text{ mV}$$

L'erreur commise sur la conversion est donnée par la formule

$$\epsilon_c \leq \frac{1}{2} \frac{U_{ref}}{2^n}$$

donc on aura $\epsilon_c \leq 9,76 \text{ mV}$

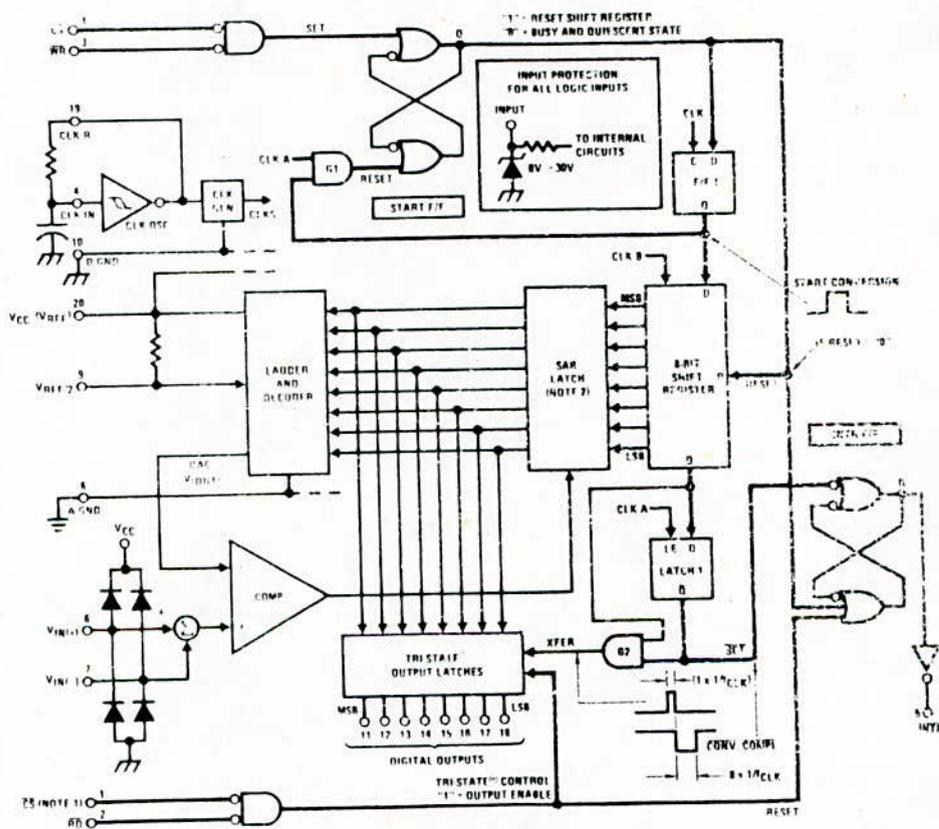
3- Fonctionnement de L'ADC 802 (Voir fig.3)

Pour étudier le fonctionnement de L'ADC 802 il faut tout d'abord présenter les trois commandes suivantes.

\overline{CS} : signal actif à l'état bas, valide les 2 signaux de lecture et d'écriture \overline{RD} et \overline{WR} du CAN et sélectionne le CAN.

\overline{WR} : passage de l'état bas à l'état haut de ce signal, il permet le lancement de la conversion.

\overline{RD} : la présence d'un état bas sur cette entrée permet la lecture



Note 1: CS shown twice for clarity.
 Note 2: SAR - Successive Approximation Register.

FIGURE 3 Block Diagram

de l'information binaire en sortie du CAN.

Une tension Vin appliquée à l'entrée du convertisseur est convertie après exécution de toutes les étapes suivantes.

- L'application simultanée d'un niveau bas sur \overline{CS} et \overline{WR}
- Initialisation du registre de décalage " shift Register"
- Mise à 1 de la sortie \overline{INTR} , ce qui donne une conversion en cours d'exécution.
- Au top d'horloge CLK la sortie Q de la bascule D'F/F1' passe à l'état 1 donc remise à 0 de la bascule.
- Au top d'horloge CLKB le '1' présent à l'entrée du registre à décalage est transmis au MSB du SAR. Le CNA convertit cette valeur numérique en une grandeur analogique (tension) qui sera comparée à Vin suivant le résultat de cette comparaison le MSB du SAR est laissé à 1 ou mis à 0
- Au top d'horloge suivant il ya décalage de 1 present dans le registre à décalage, puis une nouvelle comparaison est lancée.

Après 8 comparaisons, on obtient la valeur numérique N (suite de 1 et de 0)

CORRESPONDANT A LA TENSION VIN. Cette fin de conversion est indiquée par le passage de \overline{INTR} de l'état haut à l'état bas, en effet le LSB transmis sur l'entrée de la bascule DF/F2 permet de mettre à 0 \overline{INTR} par une remise à zero de la bascule RSINTRE/F une mise à l'état bas simultanè de \overline{RD} et \overline{CS} permet/:

- La lecture de la valeur numerique N
- Une remise à zero de la bascule INTRE/F donc passage de \overline{INTR} de l'état bas à l' état haut.
- Une initialisation du registre à decalage.

(Pour les figures voir schémas)

Le chronogramme des signaux correspondants est donné par la figure 4 .

Start conversion

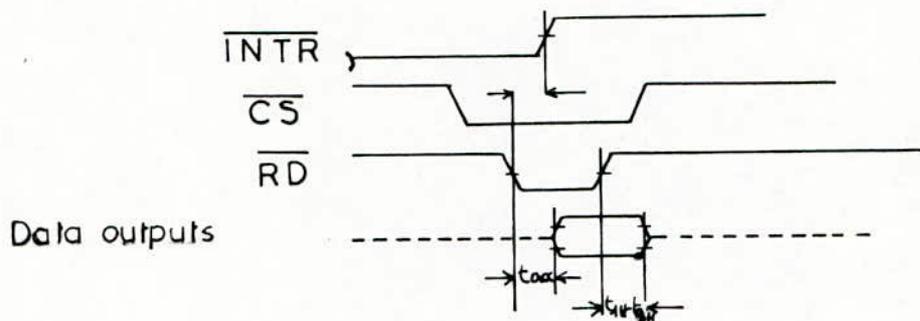
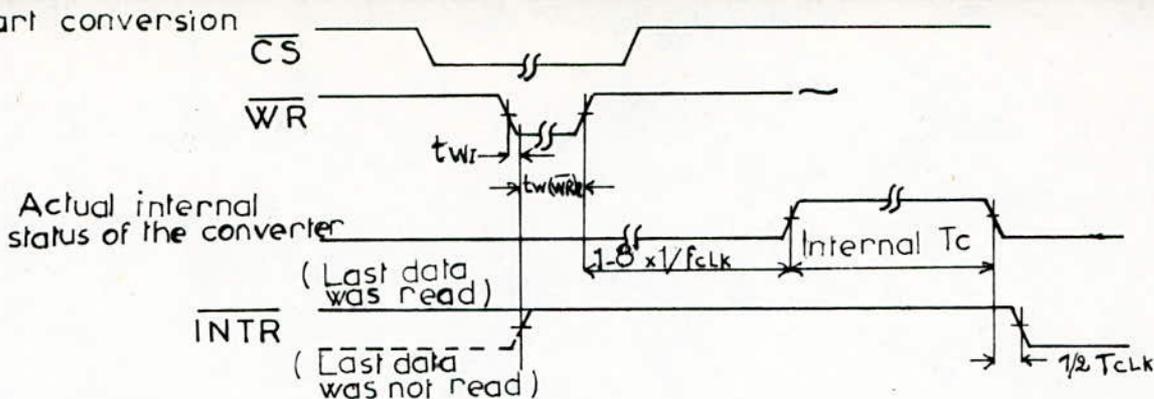


Fig. 4 LES SIGNAUX DONNÉS PAR LE CONSTRUCTEUR

Tension analogique (V)	$6 \cdot 10^3$	$16 \cdot 10^3$	1	2	3	4	5
Tension affichée (binaire)	0000 0000	0000 0001	0011 0010	0110 0110	10011000	110011.00	1111 1111
Tension numérique (V)	0	$19,61 \cdot 10^3$	0,98304	2	2,98304	4	5

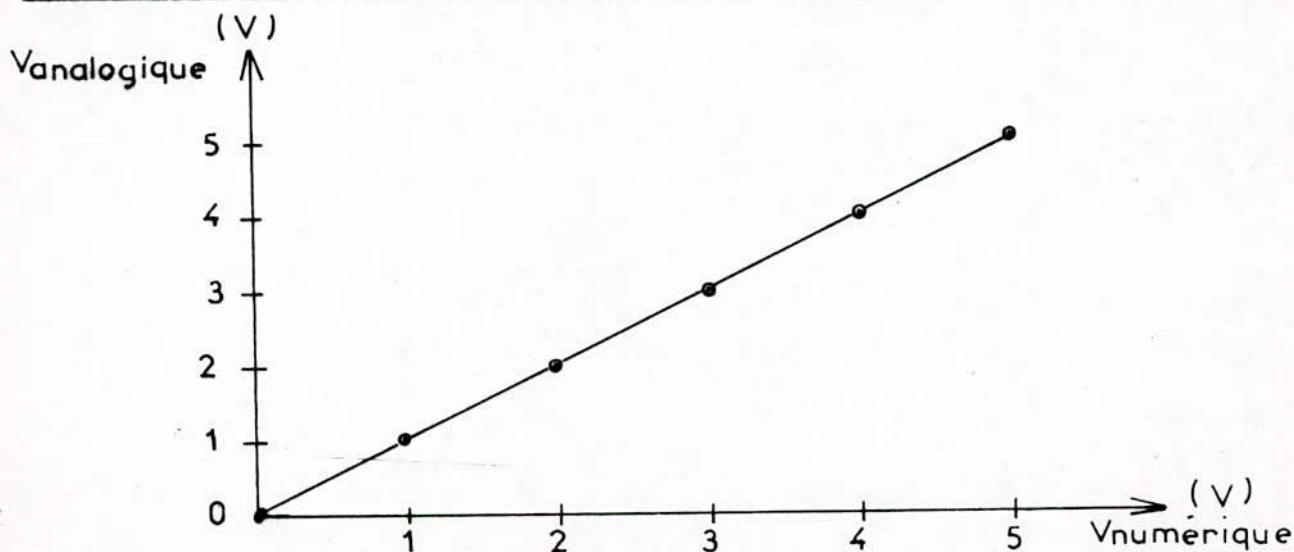


Fig. 5 COURBE DE REPONSE DE L'ADC 8302

4 - Brochage de l'ADC 802 (VOIR FIGURE 7)

5 - TEST DE l'ADC 802)

Plusieurs tests sont possibles, on choisit le suivant.

Al'entrée du convertisseur on applique une tension analogique, à la sortie on utilise des leds pour visualiser la valeur numérique correspondante; (VOIR FIGURE 6).

Remarque : Pour que l'ADC fonctionne normalement il faut que l'interrupteur soit fermé au début et puis on l'ouvre on remarque que l'entrée \overline{WR} passe à l'état haut afin de donner l'ordre de conversion .

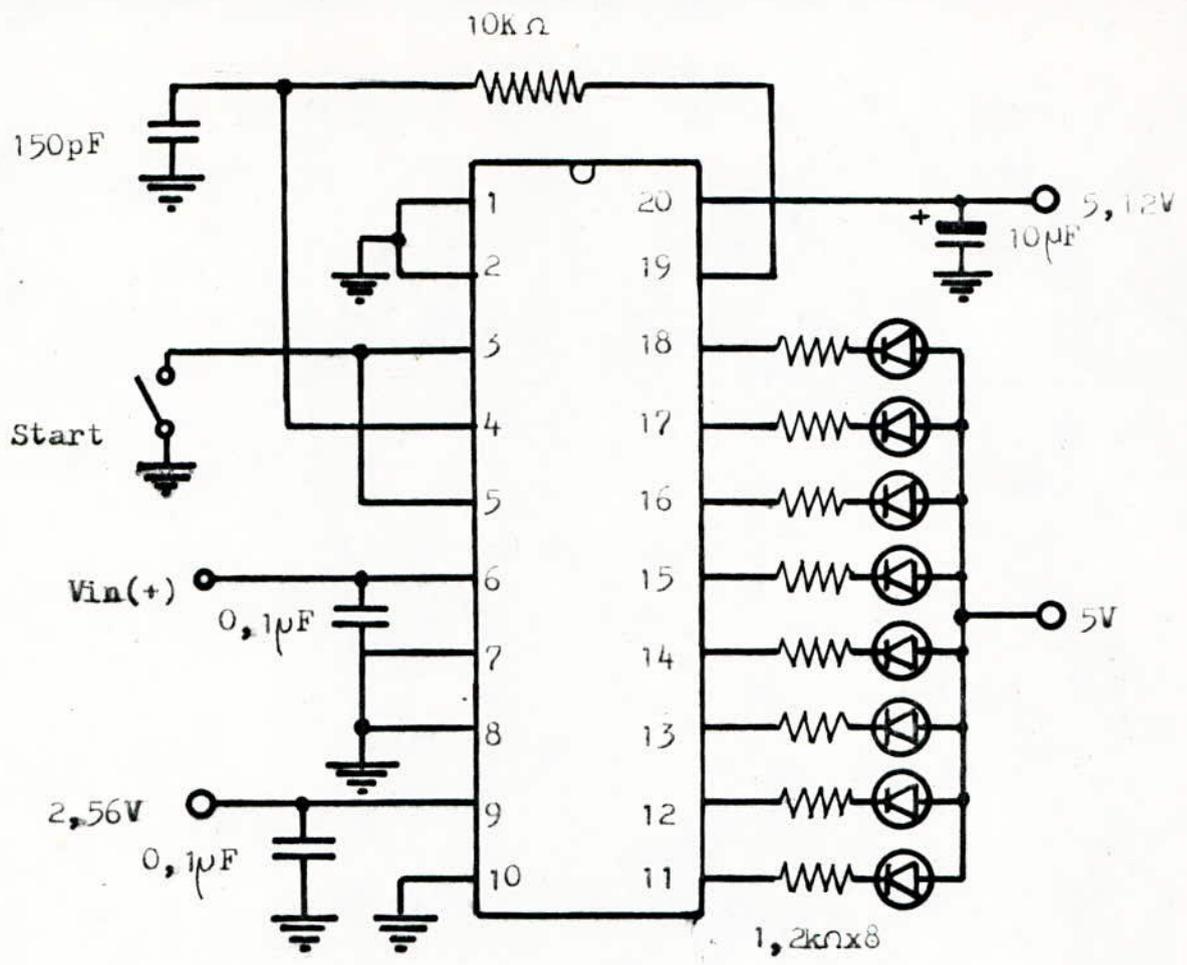
Essais et resultats:

Une tension de 6mV appliquée sur Vin(+) du convertisseur, les leds sont allumés TTTTTTTT (car les leds sont disposés de façon à ce qu'elles donnent la valeur numérique complémentaire) La première transition c'est à dire le passage de TTTTTTTT à TTTTTTT0 est obtenue pour une augmentation de la tension appliquée sur Vin(+) de 16mV ce qui justifie la précision du CAN qui est 1 LSB 19,53mV théoriquement.

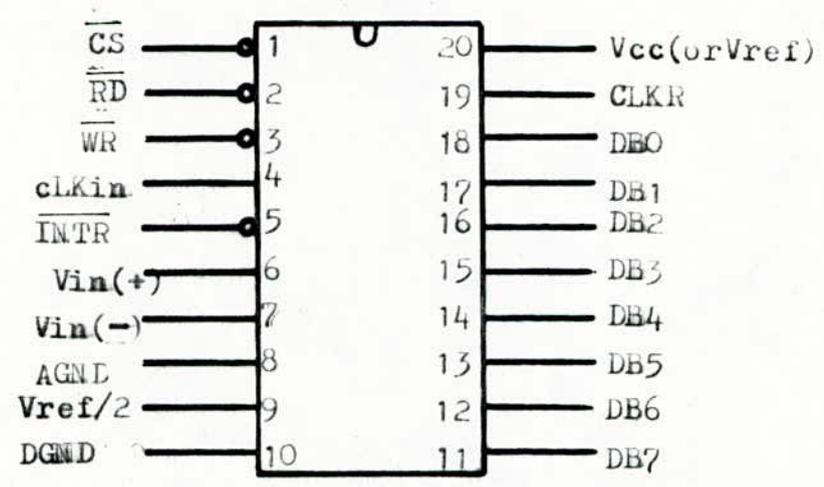
2

Une tension de 5V est appliquée sur Vin(+) donne la pleine échelle c.a.d les huit(8) leds éteintes 00000000 .

La figure 5 donne la caractéristique de notre convertisseur. On voit bien qu'elle est presque identique à celle d'un convertisseur idéal, ce qui vérifie bien la précision de l'ADC 802.



SCHEMA DE TEST DE L'ADC08302
Fig. 6



BROCHAGE DE L'ADC08302
Fig. 7

REMARQUE/:

Le convertisseur qu' on a utilisé convient bien à la conversion de notre signal vis à vis du temps de conversion qui est de l'ordre de 100 μ s sans utiliser un échantillonneur bloqueur; L'utilisation d'un échantillonneur bloqueur provoque toujours des distorsions et des bruits sur tout si le temps de maintien est grand, mais ceci n'interdit pas son utilité dans le cas des signaux rapides et comme dans notre cas le signal est lent ,on n'a pas utilisé l'échantillonneur bloqueur.

Le signal à convertir:

$$Mo(t) = Mo(0) \exp\left(-\frac{t}{T1}\right) \text{ ou } Mo(0) \text{ c'est la valeur maximale du signal}$$

T1 temps de relaxation spin-~~...~~ réseau.

La vitesse maximale de variation de ce signal est:

$$\frac{dMo(t)}{dt_{max}} = \frac{-Mo(t)}{T1} \implies \left| \frac{dMo(t)}{dt_{max}} \right| = \frac{Mo(t)}{T1}$$

On définit le cas le plus défavorable c.a.d la partie la plus grande ΔM pendant le temps $\Delta t = t_c$ pour une valeur de $Mo(0)$ la plus grande ;

Le signal possède les caractéristiques suivantes /

$$0 \leq Mo(0) \leq 5 \quad [Mo(0)] \text{ en V} \quad 100 \leq T1 \leq 2000 [T1] \text{ en ms}$$

On aura $Mo(0) = 5V$

$$Tc = 100 \mu s$$

$$\Delta Mo(t) = \frac{5 \times 100 \times 10^{-3}}{T1} \quad (T1 \text{ en ms})$$

$$\Delta M_{\frac{0}{5}} = \frac{0,5}{T1} \text{ mv} \quad (T1 \text{ en seconde}) \quad \Delta M_{\frac{0}{max}} = 5 \text{ mv}$$

On voit bien que $\Delta M_{\frac{0}{max}}$ est très inférieur à la valeur LSB (19,53mv). Ce qui justifie l'inutilité de l'échantillonneur bloqueur . $\Delta Mo_{\frac{0}{max}} = 5 \text{ mv} < 19,53 \text{ mv}$

CHAPITRE 3

CHAPITRE IIIA/- ETUDE DE Kit D5 6802 DE MOTOROLA

1- Description du Kit: Le Kit MEK D5 6802 de MOTOROLA est un microordinateur basé autour du microprocesseur M6 6802. Il est constitué par différents blocs fonctionnels schématisés par la figure 8 ci-joint:

II-A)- Le MPU 6802 /: Le 6802 est un microprocesseur 8bits dans un boîtier de 40 broches voir figure 9, réalisé en technologie MOS canal N. Il possède un bus de données de 8 bits et un bus d'adresses de 16 bits, ce qui lui permet d'adresser 2^{16} positions mémoires soit 64 Koctets. Il ne nécessite qu'une seule tension d'alimentation + 5 volts.

16161- Organisation interne.

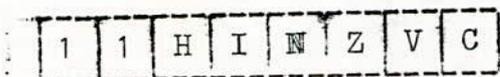
a) L'unité arithmétique et logique (UAL): qui effectue les opérations arithmétiques et logiques nécessaires au traitement de l'information.

b) Les registres internes: Il possède 5 types de registres internes

- 2 accumulateurs A et B de 8 bits
- Un registre index "IX" de 16 bits
- Un compteur ordinal "PC" ou compteur de programme de 16 bits
- Un pointeur de pile SP de 16 bits
- Un registre des codes conditions "CCR" de 8 bits qui contient 6 indicateurs

C : indicateur de retenu ; N : indicateur de signe
 V : indicateur de déplacement H : indicateur demi-retenu
 Z : indicateur de Zéro I : indicateur de masque

d'interruption.



c) L'unité de commande qui assure à partir du registre d'instruction le séquençement de toutes les opérations logiques et la gestion du système au rythme de l'horloge.

d) L'horloge: Le 6802 possède un oscillateur interne piloté par un quartz externe. Les entrées Extal et Xtal sont prévus pour fonctionner avec un quartz de fréquence fondamentale 1 MHz. Cependant le diviseur par quatre intégré dans le 6802 permet l'utilisation d'un quartz de 4 MHz.

1-1-27 Fonctionnement:

a)- Les instructions du 6802: Le 6802 possède un jeu d'instruction de 72 instructions différentes. Ce jeu comprend les instructions suivantes:

Arithmétique binaire et décimale, logique décalage, décalage circulaire, chargements, stockages, branchements conditionnels et inconditionnels. Instruction de manipulation de pile et instruction associées aux interruptions.

b)- Les modes d'adressage: Le 6802 utilise 7 modes d'adressage qui sont:

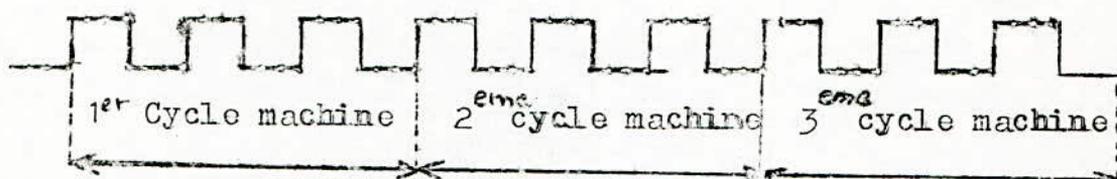
- Adressage accumulateur
- // // immédiat
- // // étendu
- // // indéxé
- // // implicite
- // // relatif

c)- Exécution d'une instruction

L'exécution d'une instruction se fait en synchronisme avec l'horloge \emptyset_2 du système, Aussi pour exécuter une instruction. Le MPU doit réaliser une suite d'opérations élémentaires:

Les microinstructions L'horloge synchronise l'ordre d'apparition de ces microinstructions spécifiées par l'unité de commande.

La fig; montre l'organisation en 3 cycles machines d'une instruction.



La carte utilise un quartz de 3,58 MHz ce qui donne une fréquence de fonctionnement (issue de l'oscillateur interne):

$3,58 \div 4 = 0,895$ MHz celle ci correspond à une période d'horloge $E(\phi_2) = 1,117 \mu s$

Le Vcc standby est relié directement à Vcc = + 5V ce qui ne permet pas la sauvegarde des 32 premiers octets de la RAM interne

1-2/- Système "PIA" (U₂₃) permet le dialogue clavier, microprocesseur en entrée et microprocesseur afficheur en sortie.

1-3/- Système Clavier (KEYPAD) La communication avec l'utilisateur se fait à l'aide d'un clavier de 25 touches dont 16 permettent l'introduction d'information en code hexadécimal et 9 qui élaborent les fonctions de commandes suivantes

a)- RESET (RS): cette touche remet à zéro le système entier après son activation. Le tiret (-) est affiché sur l'afficheur le plus à gauche.

b) ESCAPE (EX): elle permet de revenir sous le contrôle du moniteur

c) Affichage (M) : examen et modification du contenu d'une mémoire.

d) Registre display (RD) permet la visualisation des registres internes et modification de leur contenu.

- e)- Pas à Pas (T/B); permet l'exécution d'un programme pas à pas
- f)- Goto User programme (Go); permet de lancer le programme à l'adresse désirée.
- g) OFFSET CALCULATION (FS) : cette fonction permet de calculer le déplacement pour un branchement.
- h) Punch (P/L); cette fonction donne à l'utilisateur de sauvegarder des zones mémoires sur des bandes magnétiques en utilisant un magnétophone à cassette ordinaire.
- i) touche (FC): Elle est utilisée pour supprimer les points d'arrêts

1-4/ Les mémoires mortes (ROM):

(U₁₂) ROM de 2K octets est indispensable pour la gestion du Kit.

"L'USER ROM" qui peut aller jusqu'à 2K octets est par contre optionnelle.

1-5/ Les mémoires vives (RAM) :

en plus de la RAM interne du 6802, le kit possède 2 autres RAM. La SCRATCH RAM (MC 6810 "U₁₁") composée de 128 octets et "L'USERRAM" divisée en 2 boîtiers (U₇ + U₈) du type MC 2114 de capacité 1K x 4bits.

Les 2 RAMS U₇ et U₈ sont constituées par des cases mémoires de 4 bits. En étant placées en parallèle elle forme le mot complet de 8 bits. L'utilisateur ne peut travailler que sur les mémoires d'adresse 0000-,0007F et E000 à E3FF

1-6/ Décodeur d'adresse (U₆): son rôle est de décoder les lignes d'adresse de poids forts pour former 8 sorties dont chacune sélectionnera un boîtier ou des zones mémoires

1-7/ Interface série: il est constitué d'un interface série L'USER ACIA (U₃₃) et d'un générateur qu'il pourra être utilisé pour des applications Entrées/Sorties en série/:

La télétype, la console à écran à tube cathodique. La synchronisation s'effectue par l'horloge (U30).

1-8) Buffers (U1, U2 et U3 en option)

Se sont des amplificateurs des lignes d'adresses et de données. Ils sont utilisés pour l'extension du kit D5 à travers un connecteur de 86 pins.

1-9) Interface parallèle programmable PIA (6821 U9): fig 10.

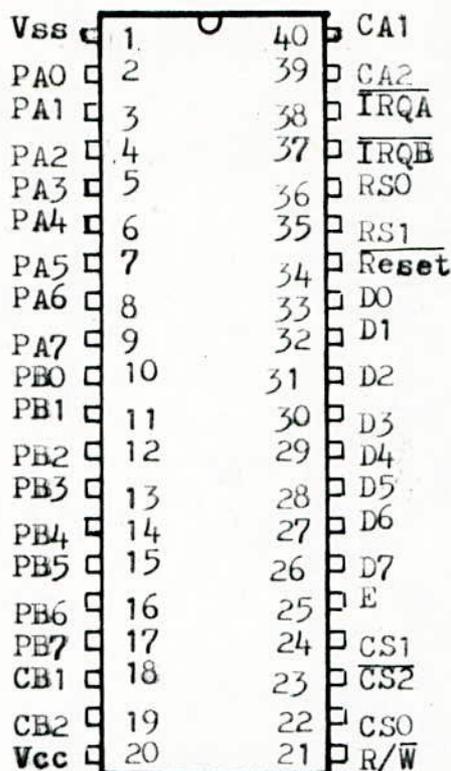
C'est une interface prévue spécialement pour l'utilisation et son usage se rapporte à faire la liaison entre le MPU et les périphériques (convertisseur A/M, imprimante, afficheurs etc ...)

Son connecteur à 24 pins (voir fig 11) nous sera très utile pour des connections.

L'étude détaillée de cette interface se fera dans la paragraphe trois de ce même chapitre (§III-3).

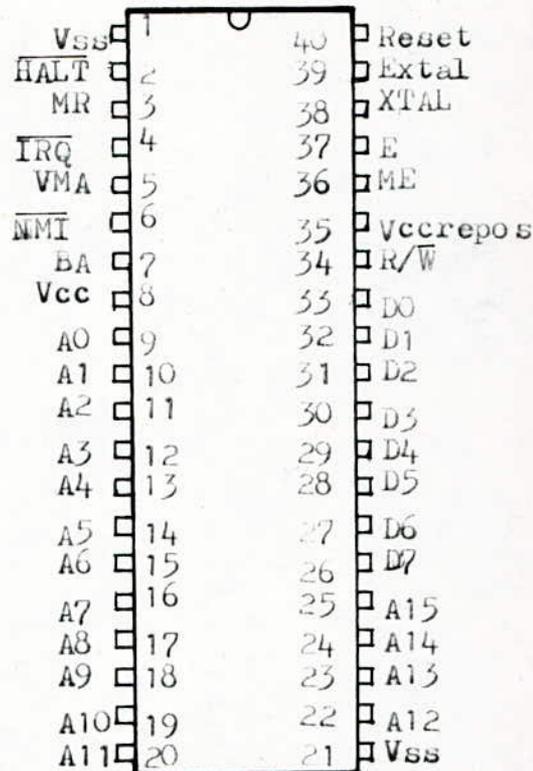
2) Répartition des adresses sur la carte

La répartition des adresses pour chaque boîtier (RAM; ROM) interface parallèle PIA, interface série ACIA est donnée à la fig 12. Les premières adresses de \$ 0000 à \$ 007F sont réservées pour la RAM interne du 6802. Les adresses de \$ 0080 à \$ DFFF (56K.octets) sont utilisées pour l'adressage des circuits externes au Kit. La Scratch RAM, le User PIA et le système PIA ont un adressage successif allant de \$ E400 à \$ E487 ceci pour l'interface série ACIA la carte D5 est conçue pour fonctionner avec 2 ROM de 2 K.octets chacune: La D5 BUG qui occupe la zone de \$ F000 à \$ FFFF et la User ROM optionnelle pour les adresses \$ E800 à \$ EFFF. Les zones mémoires \$ E488 à E6FF, \$ E702 à \$ E7FF et \$ F800 à \$ FFFF sont appelées zones miroirs et ne doivent pas être utilisées. La figure 12 donne cette répartition.



BROCHAGE DE 68302

Fig. 9



BROCHAGE DE PIA 68321

Fig.10

N° de Pin	Description	N° de Pin	Description
1	PA6	13	GND
2	PA7	14	Mc
3	PB0	15	PA5
4	PB1	16	PA4
5	PB2	17	PA3
6	PB3	18	PA2
7	PB4	19	PA1
8	PB5	20	PA0
9	PB6	21	CA2
10	PB7	22	CA1
11	CB1	23	Mc
12	CB2	24	+5v

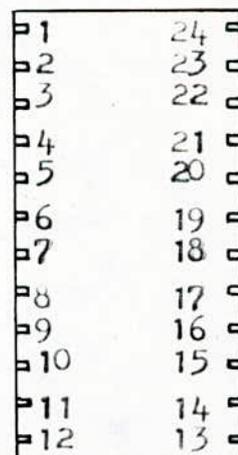


Fig.11

LE "USER CONNECTOR" ET SA NOMENCLATURE

4.2 Addressing (cont'd)

F000 - FFFF	Operating System Mirror (or optional user ROM)
F800 - F7FF	Operating System (D5BUG)
F000 - EFFF	Optional User ROM
E800 - E7FF	Reserved
E700 - E701	System ACIA*
E487 - E700	Reserved
E487	System PIA
E484 - E483	User PIA
E480 - E47F	System RAM
E400 - E3FF	User RAM (1K)
E000 - DFFF	External to MEK6802D5
0080 - 007F	User RAM inside MC6802 (must be disabled if optional Bus buffers are installed)
0000 - 007F	

*ACIA is not supported by D5BUG software.

FIGURE 12 MEMORY MAP

3/ Etude de l'interface parallèle programmable du 6802 de MOTOROLA

3.1 Présentation du PIA 6821

Le 6821 est un circuit d'interface parallèle programmable. Le PIA est vu par le MPU comme 4 positions mémoires (on dit dans ce cas que les E/S sont projetées en mémoire). Le PIA est constitué de 02 parties symétriques le port A et le port B.

Chaque port dispose de 08 lignes d'E/S PD.....à P7 programmables en entrées ou en sorties et de 02 lignes C1 et C2. C1 est une interruption et C2 peut être programmé soit en entrée soit en sortie chaque port du PIA contient 3 registres.

-Un registre de données (OR) image des lignes d'E/S PO....P7 c'est dans ce registre que le MPU viendra:

* Soit lire les données venant d'un périphérique si PO...P7 sont programmés en entrées.

* Soit écrire les données à envoyer vers un périphérique si PO...P7 sont programmés en sorties.

-Un registre de direction de données (DDR):

chaque bit de ce registre permet de définir le sens du travail (entrée ou sortie) de la ligne qui lui correspond. Si un bit de ce registre est à 1 la ligne qui lui correspond est programmée en sortie. Si un bit de ce registre est à 0 la ligne qui lui correspond est programmée en entrée. Ainsi, si le DDRA (DDR du port A) contient la valeur 11100000 les lignes P7, P6, P5 sont en sortie et les lignes P4, P3, P2, P1 et P0 sont en entrée.

-Un registre de contrôle (CR):

Il permet de définir le mode de fonctionnement des lignes de dialogue C1 et C2 dont la signification de chaque bit sera vue.

3.2) Adressage du PIA/

Le PIA possède 06 registres (2OR,2DDR,2CR) et est vu par le MPU comme 4 positions mémoires d'adresses :

E480 pour (ORA ou DDRA).

E481 pour (CRA).

E482 pour (ORB ou DDRB).

E483 pour (CRB).

On voit que ^{ORA} (ORA et DDRA) ont la même adresse ainsi que (ORB et DDRB). En fait cette ambiguïté est levée grâce au bit 2 de chaque registre de contrôle CR; si le bit 2 de CRA est à 1; l'adresse E480 donnera accès à DDRA. De même selon que le bit 2 de CRB est à 1 ou 0 on accèdera à partir de l'adresse E482 respectivement à ORB ou DDRB .

3-3) PROGRAMMATION DU PIA/

Les registres de contrôle du PIA remplissent deux rôles (figure 13); le mode de fonctionnement est déterminé par le contenu des bits 0 à 5 (à la lecture et l'écriture) tandis que les bits 6 et 7 remplissent le rôle des indicateurs d'état associés aux lignes de dialogue : CA1, CA2 , CB1 , CB2.

CA1 et CB1 toujours en entrée sont initialisés par le contenu des bits 0 et 1 de CRA et CRB . Ils positionnent l'indicateur CRA7 ou CRB7 quand ~~elles~~ sont actives .

CA2 et CB2 travaillent en entrée quand $CRX5 = 0$ (X=A ou B) ~~elles~~ sont initialisés de la même manière que CA1 et CB1 par le contenu des bits 3 et 4 de CRA et CRB . ~~elles~~ positionnent l'indicateur ~~elles~~ CRA6 ou CRB6 quand elles sont actives.

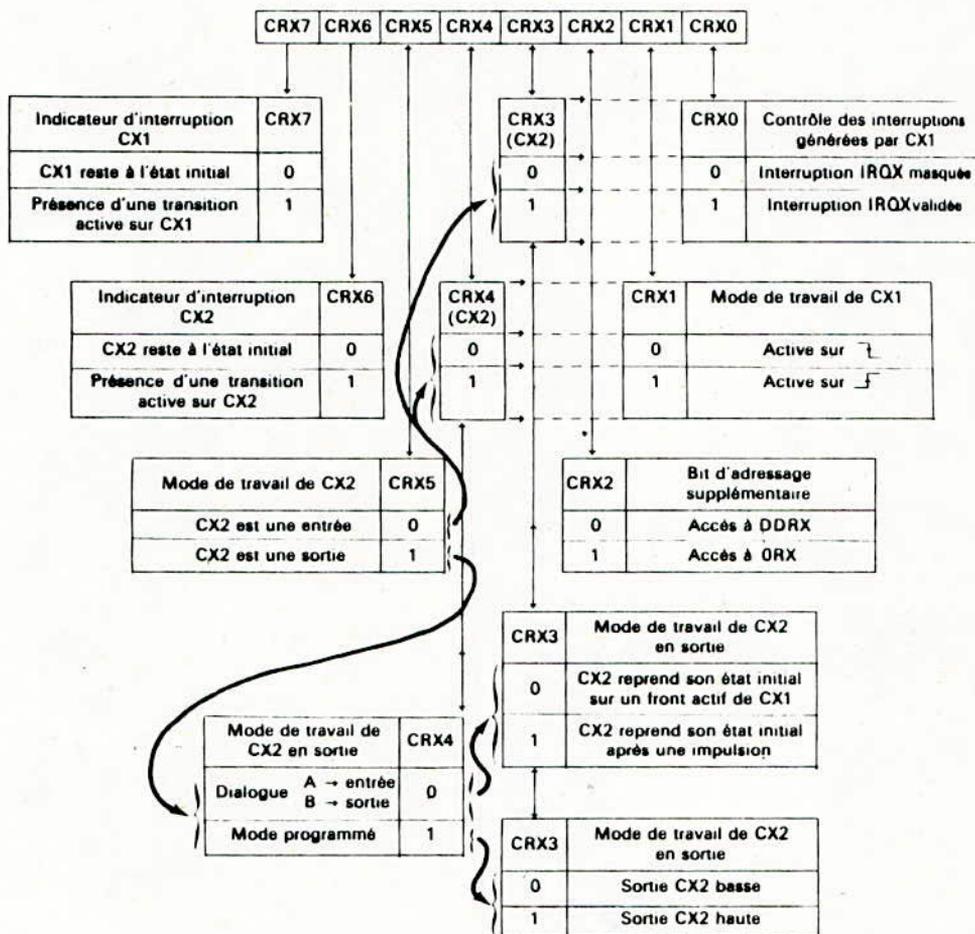


Fig 13 — Programmation des registres de contrôle du PIA — X = A ou B.

3-4 / Programmation des lignes de dialogues CA2 et CB2/ en sortie/

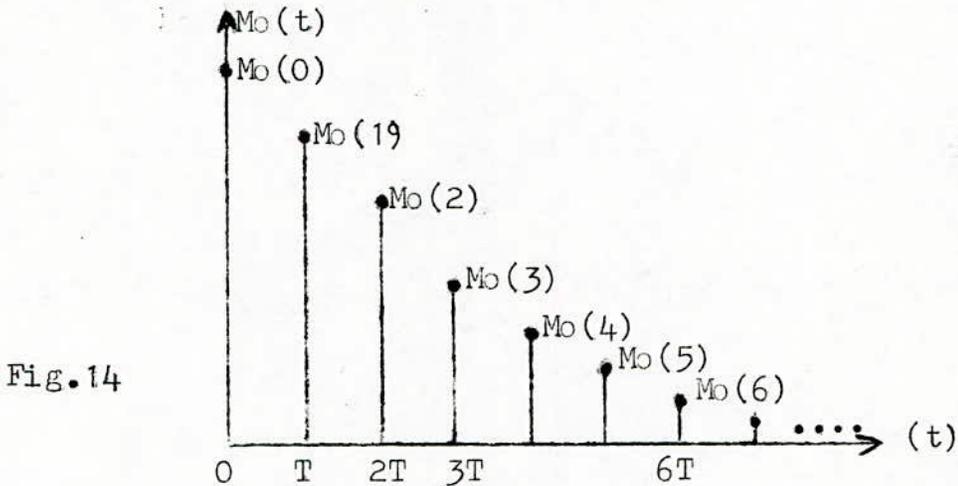
CA2 et CB2 travaillent en sortie quand $CRX5=1$ ($X = A$ ou B). Le contenu du bit 4 permet alors de choisir 2 modes de fonctionnement:

Le mode dialogue ($CRX4 = 0$) ou le mode programmé ($CRX4 = 1$)

- Pour le fonctionnement en mode programmé, les lignes CA2 et CB2 prennent l'état du bit 3. Pour $CRA3 = 0$, par exemple, la ligne CA2 est à l'état bas, l'écriture de $CRA3 = 1$ entraîne le passage de CA2 à l'état haut. Le fonctionnement est identique pour CB2 - CRB2- contrairement aux autres modes de fonctionnement, dans le mode dialogue, les parties A et B du PIA ont un fonctionnement différent. le port A travaille en entrée, la lecture par microprocesseur du registre ORA entraîne le passage à l'état bas de CA2 ($CRA4 = 0$) le port B travaille en sortie, l'écriture par le microprocesseur du registre ORB entraîne le passage à l'état bas de CB2. Dans ces deux cas, c'est le contenu de CRX3 qui détermine le mode de retour à l'état initial de CA2 ou CB2. Les lignes CA2 ou CB2 repassent à l'état haut ($CRX3 = 0$) quand un front actif est détecté sur CA1 ou CB1. Si $CRX3 = 1$, CA2 ou CB2 reprennent leur état initial, quand le circuit est désselecté, sur un front montant (CB2) ou descendant (CA2).

1-Position du problème:

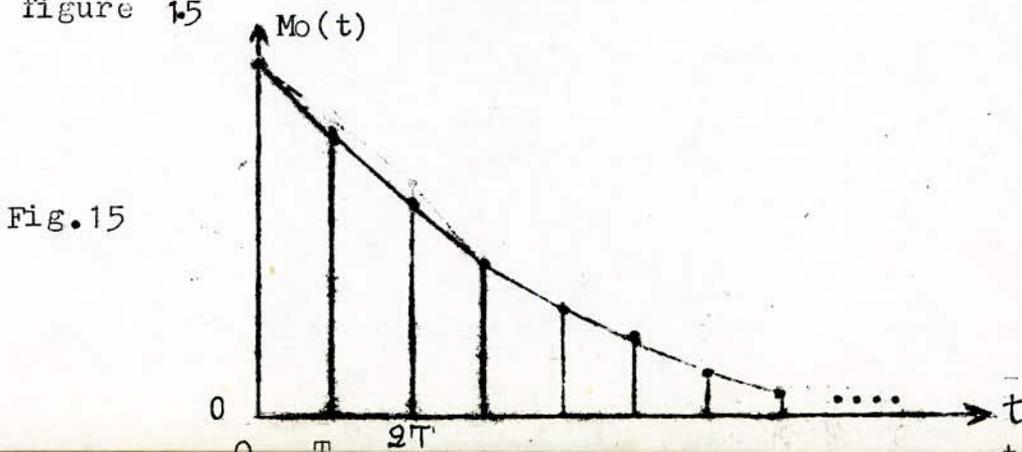
Les données prélevées par le convertisseur analogique numérique seront stockées dans les cases mémoire à partir de l'adresse 000; Ces données constituent une fonction discrète d'un exponentiel comme l'indique la figure 14.



La constante de temps de cette exponentielle est obtenue lorsque $Mo(T)$ décroît d'un taux de 63% DE $Mo(0)$ à cet instant le temps correspondant sera égal à τ . Pour mesurer cette constante de temps on procède au principe suivant qui fait l'objet du paragraphe suivante.

2-Principe : linéarisation de la courbe $Mo(t)$

Pour mesurer la constante de temps avec une grande précision en utilisant un programme. Il suffit de linéariser la courbe entre les instants $0-T, T-2T, 2T-3T, \dots$ etc comme l'indique la figure 15



et de calculer l'équation de chaque droite pour faciliter le programme.

3- Calcul de l'équation générale donnant t en fonction de n et T

$n \in \mathbb{N}$

L'équation d'une telle droite est donnée par l'expression suivante :

$$y = ax + b \quad \text{tg} : a \text{ c'est la pente de la droite } a = \frac{\Delta y}{\Delta x}$$

et b c'est une constante déterminée par les conditions initiales .

Première droite :

$$0 \leq t \leq T \quad a = \frac{Mo(1) - Mo(0)}{T - 0} = \frac{Mo(1) - Mo(0)}{T} < 0$$

$$\text{Donc } Mo(t_1) = \frac{Mo(1) - Mo(0)}{T} \cdot t + Mo(0) \quad \text{avec } 0 \leq t \leq T$$

Deuxième droite :

$$T \leq t_2 \leq 2T$$

$$a = \frac{Mo(2) - Mo(1)}{T}$$

$$b = Mo(1) - T \frac{Mo(2) - Mo(1)}{T}$$

$$= 2Mo(1) - Mo(2)$$

$$\text{Donc } Mo(t_2) = \frac{Mo(2) - Mo(1)}{T} \cdot t + 2Mo(1) - Mo(2)$$

Troisième droite :

$$2T \leq t \leq 3T$$

$$a = \frac{Mo(3) - Mo(2)}{T}$$

$$b = Mo(2) - 2T \cdot \frac{Mo(3) - Mo(2)}{T}$$

$$b = 3Mo(2) - 2Mo(3)$$

$$\text{Donc } Mo(t_3) = \frac{Mo(3) - Mo(2)}{T} \cdot t + 3Mo(2) - 2Mo(3)$$

Pour la $i^{\text{ème}}$ droite :

$$(i-1) T \leq t \leq iT$$

$$\text{On a } Mo(t_i) = \frac{Mo(i) - Mo(i-1)}{T} \cdot t + i Mo(i-1) - (i-1) Mo(i)$$

Si on tire à partir de cette expression t_i , on aura :

$$t_i = \left[\frac{i Mo(i-1) - (i-1) Mo(i) - Mo(t_i)}{Mo(i-1) - Mo(i)} \right] \cdot T$$

$$\Rightarrow t_i = \left[(i-1) + \frac{Mo(i-1) - Mo(t_i)}{Mo(i-1) - Mo(i)} \right] \cdot T$$

Si on pose $n=i-1$, $n \in \mathbb{N}$

$$\text{On aura } t_n = \left[\frac{Mo(n) - Mo(t_n)}{Mo(n) - Mo(n+1)} \right] \cdot T$$

On voit bien que t est facile à calculer dans n'importe quel moment mais à condition qu'on connait n ($n=0,1,2,\dots$)

Remarque : L'approximation que nous allons adopter est valable quelque soit la période du programmeur d'impulsion T parcequ'on a constaté que pour T faible la décroissance de l'exponentielle est plus importante que quand T est grande. Donc la linearisation n'introduit qu'une faible erreur que l'on calculera par la suite.

4- Calcul de l'erreur Δt introduite par la linéarisation.

Le fait qu'on a linearisé la courbe entre les instants ($0-T$) et ($T-2T$) ...etc..., on a évité l'erreur provoquée par le micro-ordinateur. Cette erreur est plus importante et peut atteindre $\pm \frac{1}{2} T$ mais en réalité cette linearisation introduit une erreur Δt faible que l'on admettra sans problème vu l'excellente méthode adoptée (linéarisation).

La fonction réelle $f(t) = Mo(0) \exp(-t/T_1)$ est approximée par le polynôme du premier degré $P(t)$.

$$P(t) = \frac{Mo(n+1) - Mo(n)}{T} \cdot t + (n+1)Mo(n) - nMo(n+1)$$

L'erreur sera donc $\epsilon(t) = f(t) - P(t)$

Soit : $\epsilon(t) = Mo(0) e^{-\frac{t}{T_1}} - \frac{Mo(n+1) - Mo(n)}{T} \cdot t - (n+1)Mo(n) + nMo(n+1)$

D'autre part l'incertitude sur t introduite par l'erreur $\epsilon(t)$ sera calculée comme suit :

On a : $\text{Log}f(t) = \text{Log}Mo(0) - \frac{t}{T_1}$

Le différentiel logarithmique : $\frac{df(t)}{f(t)} = -\frac{dt}{T}$ car $Mo(0)$ est cte

Si on passe aux incertitudes on aura :

$$\frac{\Delta f(t)}{f(t)} = \left| -\frac{\Delta t}{T} \right| = \frac{\Delta t}{T} \Rightarrow \Delta t = T \cdot \frac{\Delta f(t)}{f(t)}$$

Avec $\Delta f(t) = \epsilon(t)$

$$\Rightarrow \Delta t = T_1 \left\{ 1 - \frac{Mo(n+1) - Mo(n)}{Mo(0) \cdot T} \cdot t e^{\frac{t}{T_1}} - \left[(n+1) \frac{Mo(n)}{Mo(0)} - \frac{nMo(n+1)}{Mo(0)} \right] e^{\frac{t}{T_1}} \right\}$$

On remarque que Δt est en fonction du temps qui a un maximum que l'on calculera .

Lorsque Δt est maximum on a : $\frac{\partial(\Delta t)}{\partial t} = 0$

$$\text{on a } \frac{\partial(\Delta t)}{\partial t} = T_1 \left\{ -\frac{Mo(n+1) - Mo(n)}{Mo(0) \cdot T \cdot T_1} \cdot t_0 e^{\frac{t_0}{T_1}} - \left\{ (n+1) \frac{Mo(n)}{Mo(0) T_1} - \frac{nMo(n+1)}{Mo(0) T_1} \right\} e^{\frac{t_0}{T_1}} \right\} = 0$$

$$\Rightarrow -\frac{Mo(n+1) - Mo(n)}{Mo(0) T T_1} t_0 - \frac{Mo(n+1) - Mo(n)}{Mo(0) T} - \frac{(n+1) Mo(n)}{T_1 Mo(0)} + \frac{n Mo(n+1)}{Mo(0) T_1} = 0$$

$$t_0 \frac{Mo(n+1) - Mo(n)}{Mo(0) T T_1} = \frac{1}{Mo(0) T T_1} \left(T_1 [Mo(n+1) - Mo(n)] + T [n Mo(n+1) - (n+1) Mo(n)] \right)$$

$$t_0 = -T_1 + nT - \frac{Mo(n)}{Mo(n+1) - Mo(n)} \cdot T$$

C'est la valeur du temps qui fait l'erreur Δt max

Donc pour trouver Δt max , il suffit de remplacer l'expression de t_0 dans la formule de Δt .

Le tableau suivant donne les valeurs de t_0 pour différentes valeurs de n .

n	0	1	2	3	4	5
t_0	0,53T	1,52T	2,52T	3,54T	4,52T	5,51T

On remarque que t_0 peut s'écrire de la manière suivante:
 $t_0 = \frac{T}{2} (n+0,5)$; en remplaçant celle ci dans l'expression de t
on aura:

$$t_{\max} = T_1 \left[1 - \frac{M_0(n+1) + M_0(n)}{2M_0(0)} \exp(n+0,5) - \frac{T}{T_1} \right]$$

Sachant que $M_0(n+1) = M_0(0) \exp(-n \frac{T}{T_1})$ et

L'expression de Δt_{\max} devient:

$$\Delta t_{\max} = T_1 \cdot 1 - \frac{\exp(-\frac{T}{2T_1})}{2} \quad \text{avec}$$

$$\frac{\exp(-\frac{T}{2T_1}) + \exp(\frac{T}{2T_1})}{2} = \text{ch}\left(\frac{T}{2T_1}\right);$$

$$\text{D'où } \Delta t_{\max} = T_1 \cdot \left[1 - \text{ch}\left(\frac{T}{2T_1}\right) \right].$$

C- REALISATION DE LA CARTE D'INTERFAÇAGE

1/ Schema du montage :

Le schema de la figure (16) illustre la manière du brochage de l'ADC 802 avec le PIA 6821 du micro-ordinateur à travers les pins du " USER CONNECTOR "

Le cahier de charge nous impose de prendre des informations sur un signal analogique appliqué aux entrées $v_{in}(+)$ et $v_{in}(-)$ dans des instants bien déterminés . Ce signal est délivré par un spectrometre RMN , ainsi ces informations doivent être prelevées après chaque front négatif de l'impulsion \overline{Q} , voir fig.17. Pour cela on a synchronisé le lancement de la conversion avec le signal à convertir , tout en commandant l'entrée \overline{WR} de l'ADC par les impulsions du programmeur .Et comme le \overline{WR} est activé qu'à l'état haut (front positif) , on a inversé le signal à l'aide d'une porte logique NAND du CI SN7400 (les entrées de la porte NAND sont court-circuitées) , le signal ainsi inversé attaque directement l'entrée \overline{WR} ce qui resoud complètement le problème de synchronisation .

Dés que la conversion est terminée le signal \overline{INTR} de l'ADC 802 passe à l'état bas , ce passage se traduit par un front négatif sur CB_1 et par consequent le bit b_7 du registre de contrôle CRB du PIA se positionne à l'état 1 . Et enfin pour lire la donnée convertie, on génère un front négatif sur CA_2 qui active \overline{RD} pour que la donnée présentée sur les BUFFERS-TRI-STATE ($DB_0 - DB_7$) puisse être transférée vers le registre de donnée du port B (ORB) .

L'ADC802 nécessite une seule tension d'alimentation de 5v , elle est obtenue en reliant la broche 20 de ce dernier avec la pin n° 24 du USER CONNECTOR du PIA . ainsi que \overline{CS} est toujours à l'état bas pour valider les signaux de commandes \overline{RD} et \overline{WR} , elle est donc reliée à la masse .

Sortie de Programmeur
d'impulsions

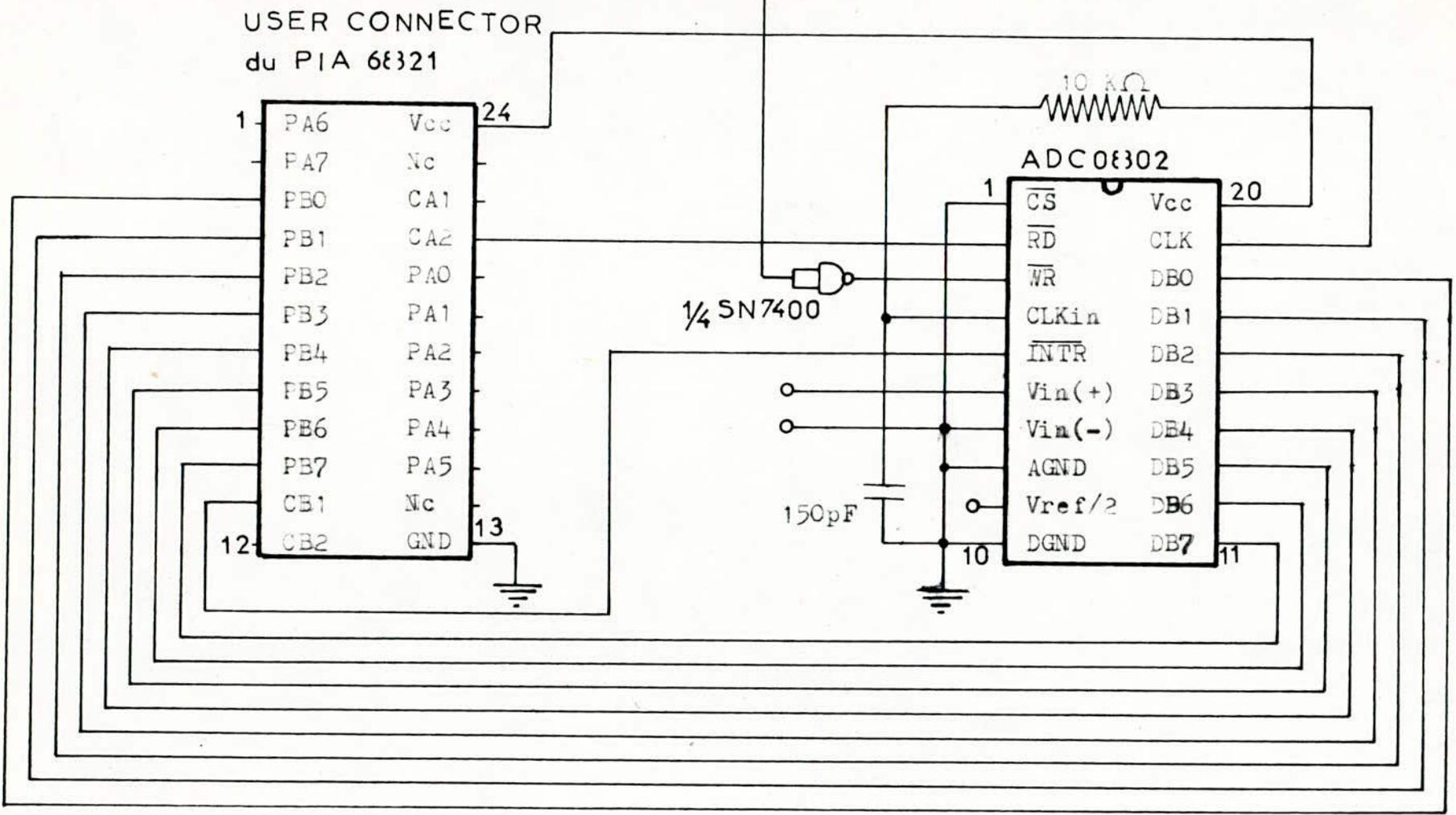


Fig.16 SCHEMA ELECTRONIQUE DU MONTAGE

Remarque / Le montage n'est valable que pour le calcul de T1

2 - Circuit imprimé/

Le schema d'implantation de la carte d'interfaçage est réalisé sur une plaque à simple face en verre epoxy de dimension 10X15cm²(voir figure 18)

L'ADC 802 et le SN7400 sont disposés respectivement sur des supports de 20 broches et 14 broches afin de faciliter le changement de ces derniers dans le cas de deterioration. On signale aussi qu'un strap est necessaire pour assurer la liaison de CS à la masse. La plaque est fixée à l'aide de 4 vis sur un coffret, sur la face de cet coffret figure trois entrées réservées à Vin(+), Vin(-) et la sortie du programmeur .

La face arriére sort d'un fil en nappe de 24 broches qui assure la liaison avec l'USER CONNECTOR du PIA 682I .

3 - Essais/

Le fait que le spectrometre RMN n'est pas prés on fait la simulation suivante le programmeur d'impulsion est remplacé par un timer NE555 qui délivre des impulsions rectangulaires de frequence 50Hz(T=20ms). La tension analogique appliquée à Vin(+) et Vin(-) c'est la tension aux bornes d'une capacité, avec une resistance R en serie alimentés par des impulsions carrées délivrées par un générateur BF . Voir figure 18.

Le programme executé par le micro-ordinateur c'est de stocker 10 données à partir de l'adresse mémoire \$00. Les essais sont faits sur 3 fréquences differentes du générateur BF .

1 - cas: fg=fc: Les 10 données sont identiques($\frac{F}{F}$)=5V

2 - cas: fg \neq fc: 2 valeurs se repetent alternativement mais identiques pour les 10 données .

3 - cas: fg=fc: La donnée se repete après 3 conversions c.a.d la donnée de l'adresse 00 se retrouve dans l'adresse \$04 et de même pour les autres .

Les resultats obtenus sont très satisfaisants .

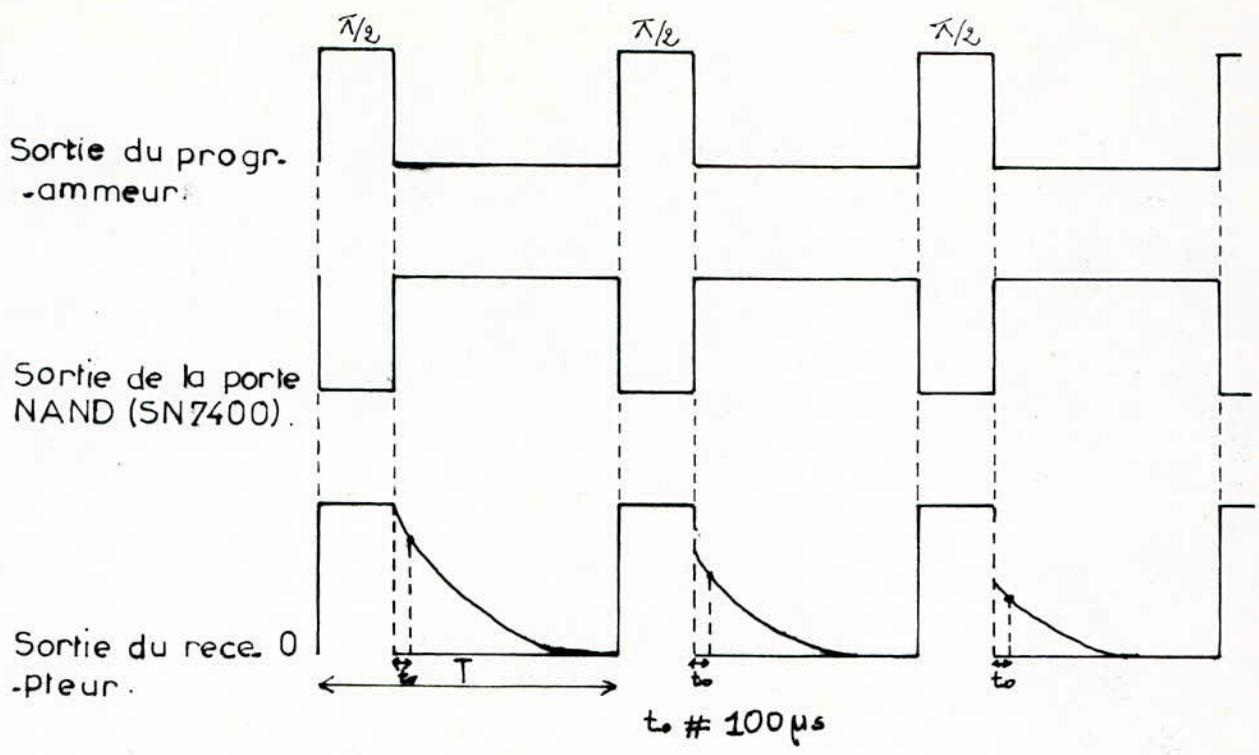


Fig.17 CHRONOGRAMME DES SIGNAUX

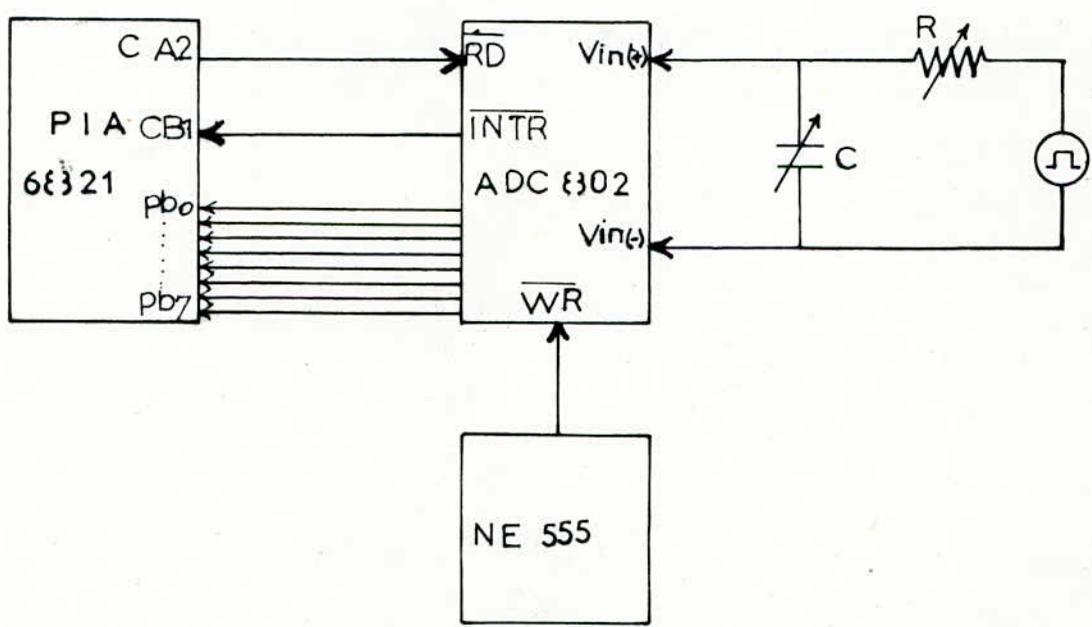


Fig.18 SCHEMA DU MONTAGE D'ESSAI

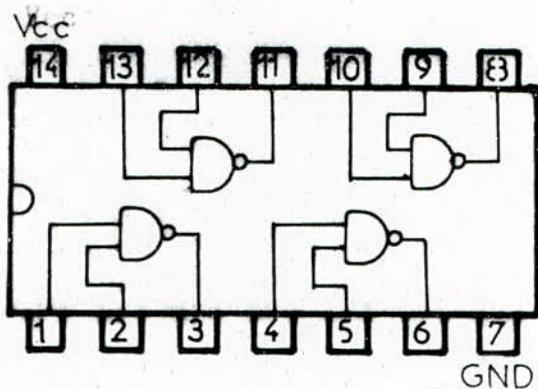


Fig.19

LE CI SN 7400

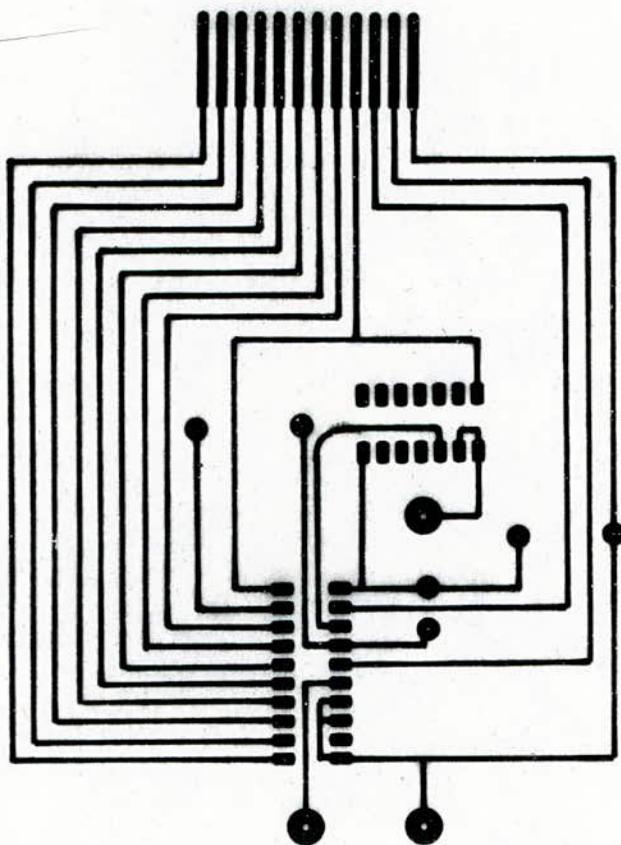


Fig.20

LE CIRCUIT IMPRIMÉ

Dans ce paragraphe on détaille la procedure d'acquisition de données ainsi que le calcul du temps T1 L'organigramme general

1 - Acquisition de données: c correspondant est donné par la planche.

Il s'agit de transférer les données converties par L'ADC 802 vers le PIA et de les stocker à partir de l'adresse mémoire \$00. Le nombre des données est programmable, dans notre cas le nombre est limité à N=10, ce qui correspond à (\$0A) en hexadécimale . Ce transfert nécessite la programmation du PIA pour toutes les données .

- Programmation du PIA /

Le schema du montage de la figure 16, nous aide beaucoup à la programmation du PIA. Le port B est programmé en entrée ce qui revient à initialiser le registre de direction du port B (DDRB) à 0 lors de l'initialisation du PIA. L'entrée CB1 reçoit un signal d'état qui indique la fin de la conversion . Ainsi CA2 du port A est destiné à générer un front négatif à l'entrée \overline{RD} de l'ADC 802 pour lire la donnée .

Le mode de programmation du PIA c'est le mode programmé (b4=1)

- Détermination des mots de commande :

Les mots de commande sont déterminés à partir des bits des registres de contrôle (ou de commande) du port A ou B .

Port A: Dans ce port le seul signal d'état utilisé est CA2

En sortie b5=1, on a choisi le mode programmé b4=1 .

Les bits b0, b1 concernant CA1 sont toujours à zéro (b1=b2=0).

Neanmoins les bits b6 et b7 sont toujours à zéro (b6=b7=0)

Lors de l'initialisation du PIA .

Le bit b2=1 valide l'accès au registre de donnée .

CA2 est à l'état haut.

0	A	0	1	1	1	0	0	0
---	---	---	---	---	---	---	---	---

\$3C

CA2 est à l'état bas .

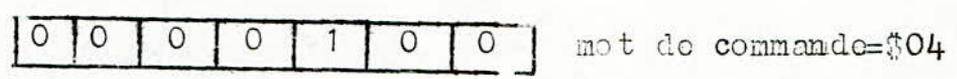
0	0	1	1	0	1	0	0
---	---	---	---	---	---	---	---

\$34

Pour générer un front négatif tout simplement on positionne CA2 de l'état haut à l'état bas, ce qui se traduit par le passage de 1 à 0 du bit b_3 (CRA3)

Port B: Dans ce cas le signal CB2 est en l'air donc $b_5=0$, $b_4=0$, $b_3=0$. Le signal CB1 est en entrée, il est activé au front négatif d'où $b_1=0$.

Dans notre cas aucune interruption ne sera déclenchée lors de la transition active de CB1 donc $b_0=0$ finalement le bit $b_2=1$ pour valider l'accès au registre de donnée.



2/Programme pour la mesure du temps T1:

Le temps T1 est déterminé à partir de la formule déduite au paravant:

$$t = \left[n + \frac{Mo(n) - Mo(t)}{Mo(n) - Mo(n+1)} \right] T$$

Dans ce cas le seul paramètre variable c'est $Mo(t)$.

Après l'initialisation du registre d'index à $n=0000$. On donne à $Mo(t)$ la valeur maximale $Mo(0)$, et puis on la décremente d'une unité. Après chaque décrémentation de $Mo(t)$ on la compare avec la donnée suivante $Mo(n)$ et lorsque $Mo(t)$ devient inférieur ou égal à $Mo(n)$ le registre index sera incrémenté d'une unité. La décrementation de $Mo(t)$ et l'incrementation du registre index se poursuivent jusqu'à l'instant où $Mo(t)$ atteint $3/4$ de sa valeur initiale.

A ce moment là le temps T1 est la somme du contenu du registre d'index(n) et le quotient :

$$\frac{Mo(n) - Mo(T1)}{Mo(n) - Mo(n+1)}$$

Ce dernier peut être calculé en connaissant n , $Mo(n)$ et $Mo(n+1)$ le résultat donc est donné en fonction de la période T. Il est sous forme de deux parties: partie décimale qui se trouve dans le registre d'index(ID) et la partie fractionnelle qui se trouve dans le pointeur de pile (SP).

***** PROGRAMME POUR LA MESURE DU TEMPS DE RELAXATION T1*****

*	ORG \$E000	*	STAA \$69	*
*	LDA \$0A	*	STAB \$6A	*
*	STAA \$63	*	LDA \$69	*
*	LDX #0000	*	SUBA \$66	*
*BCL2	CLR \$E483	*	BNE BCL6	*
*	CLR \$E482	*	LDAB \$6A	*
*	LDAB #04	*	SUBB \$67	*
*	STAB \$E483	*	BNE BCL6	*
*BCL1	TST \$E483	*	LDA \$(00,X)	*
*	BPL BCL1	*	SUBA \$64	*
*	LDAB #3C	*	STAA \$6B	*
*	STAB \$E481	*	LDAB \$(00,X)	*
*	LDAB #34	*	SUBB \$(01,X)	*
*	STAB \$E481	*	STAB \$6D	*
*	LDA \$E482	*	LDA #08	*
*	STAA \$(00,X)	*	STAA \$6E	*
*	INX	*	LDAB #64	*
*	DEC \$0063	*	CLRA	*
*	BNE BCL2	*	LSRB	*
*	LDX #0000	*BCL7	BCC LP2	*
*	LDA \$00	*	ADAA \$6B	*
*	STAA \$64	*LP2	RORA	*
*BCL6	DEC \$0064	*	RORB	*
*	LDAB \$64	*	DEC \$006E	*
*	CMPB \$(01,X)	*	BNE BCL7	*
*	BNE BCL3	*	STAA \$6F	*
*	INX	*	STAB \$70	*
*BCL3	LDA #08	*	LDA #10	*
*	STAA \$65	*	STAA \$71	*
*	LDAB #25	*	CLR \$006C	*
*	CLRA	*	CLRA	*
*	LSRB	*	CLRB	*
*BCL4	BCC LP0	*BCL8	ROL \$0070	*
*	ADDA \$00	*	ROL \$006F	*
*LP0	RORA	*	ROLB	*
*	RORB	*	ROLA	*
*	DEC \$0065	*	SUBB \$6D	*
*	BNE BCL4	*	SBCA \$6C	*
*	STAA \$66	*	BCC LP3	*
*	STAB \$67	*	CLC	*
*	LDA #08	*	ADDB \$6D	*
*	STAA \$68	*	ADCA \$6C	*
*	LDAB #64	*	SEC	*
*	CLRA	*LP3	DEC \$0071	*
*	LSRB	*	BNE BCL8	*
*BCL5	BCC LP1	*	ROL \$0070	*
*	ADDA \$64	*	ROL \$006F	*
*LP1	RORA	*	COM \$0070	*
*	RORB	*	COM \$006F	*
*	DEC \$0068	*	LDS \$6F	*
*	BNE BCL5	*	SWI	*

```

*****
*****PROGRAMME EN LANGAGE MACHINE*****
*****
* E000 86 * E033 7A * E066 F5 * E099 70
* E001 0A * E034 00 * E067 97 * E09A 86
* E002 97 * E035 64 * E068 69 * E09B 10
* E003 63 * E036 D6 * E069 D7 * E09C 97
* E004 CE * E037 64 * E06A 6A * E09D 71
* E005 00 * E038 E1 * E06B 96 * E09E 7F
* E006 00 * E039 01 * E06C 69 * E09F 00
* E007 7F * E03A 26 * E06D 90 * E0A0 6C
* E008 E4 * E03B 01 * E06E 66 * E0A1 4F
* E009 83 * E03C 08 * E06F 26 * E0A2 5F
* E00A 7F * E03D 86 * E070 C2 * E0A3 79
* E00B E4 * E03E 08 * E071 D6 * E0A4 00
* E00C 82 * E03F 97 * E072 6A * E0A5 70
* E00D C6 * E040 65 * E073 D0 * E0A6 79
* E00E 04 * E041 C6 * E074 67 * E0A7 00
* E00F F7 * E042 25 * E075 26 * E0A8 6F
* E010 E4 * E043 4F * E076 BC * E0A9 59
* E011 83 * E044 54 * E077 A6 * E0AA 49
* E012 7D * E045 24 * E078 00 * E0AB D0
* E013 E4 * E046 02 * E079 90 * E0AC 6D
* E014 83 * E047 9B * E07A 64 * E0AD 92
* E015 2A * E048 00 * E07B 97 * E0AE 6C
* E016 FB * E049 46 * E07C 6B * E0AF 24
* E017 C6 * E04A 56 * E07D E6 * E0B0 06
* E018 3C * E04B 7A * E07E 00 * E0B1 0C
* E019 F7 * E04C 00 * E07F E0 * E0B2 DB
* E01A E4 * E04D 65 * E080 01 * E0B3 6D
* E01B 81 * E04E 26 * E081 D7 * E0B4 99
* E01C C6 * E04F F5 * E082 6D * E0B5 6C
* E01D 34 * E050 97 * E083 86 * E0B6 0D
* E01E F7 * E051 66 * E084 08 * E0B7 7A
* E01F E4 * E052 D7 * E085 97 * E0B8 00
* E020 81 * E053 67 * E086 6E * E0B9 71
* E021 B6 * E054 86 * E087 C6 * E0BA 26
* E022 E4 * E055 08 * E088 64 * E0BB E7
* E023 82 * E056 97 * E089 4F * E0BC 79
* E024 A7 * E057 68 * E08A 54 * E0BD 00
* E025 00 * E058 C6 * E08B 24 * E0BE 70
* E026 08 * E058 64 * E08C 02 * E0BF 79
* E027 7A * E05A 4F * E08D 9B * E0C0 00
* E028 00 * E05B 54 * E08E 6B * E0C1 6F
* E029 63 * E05C 24 * E08F 46 * E0C2 73
* E02A 26 * E05D 02 * E090 56 * E0C3 00
* E02B DB * E05E 9B * E091 7A * E0C4 70
* E02C CE * E05F 64 * E092 00 * E0C5 73
* E02D 00 * E060 46 * E093 6E * E0C6 00
* E02E 00 * E061 56 * E094 26 * E0C7 6F
* E02F 96 * E062 7A * E095 F5 * E0C8 9E
* E030 00 * E063 00 * E096 97 * E0C9 6F
* E031 97 * E064 68 * E097 6F * E0CA 3F
* E032 64 * E065 26 * E098 D7
*****

```

CONCLUSION

Fig. 8

4-2

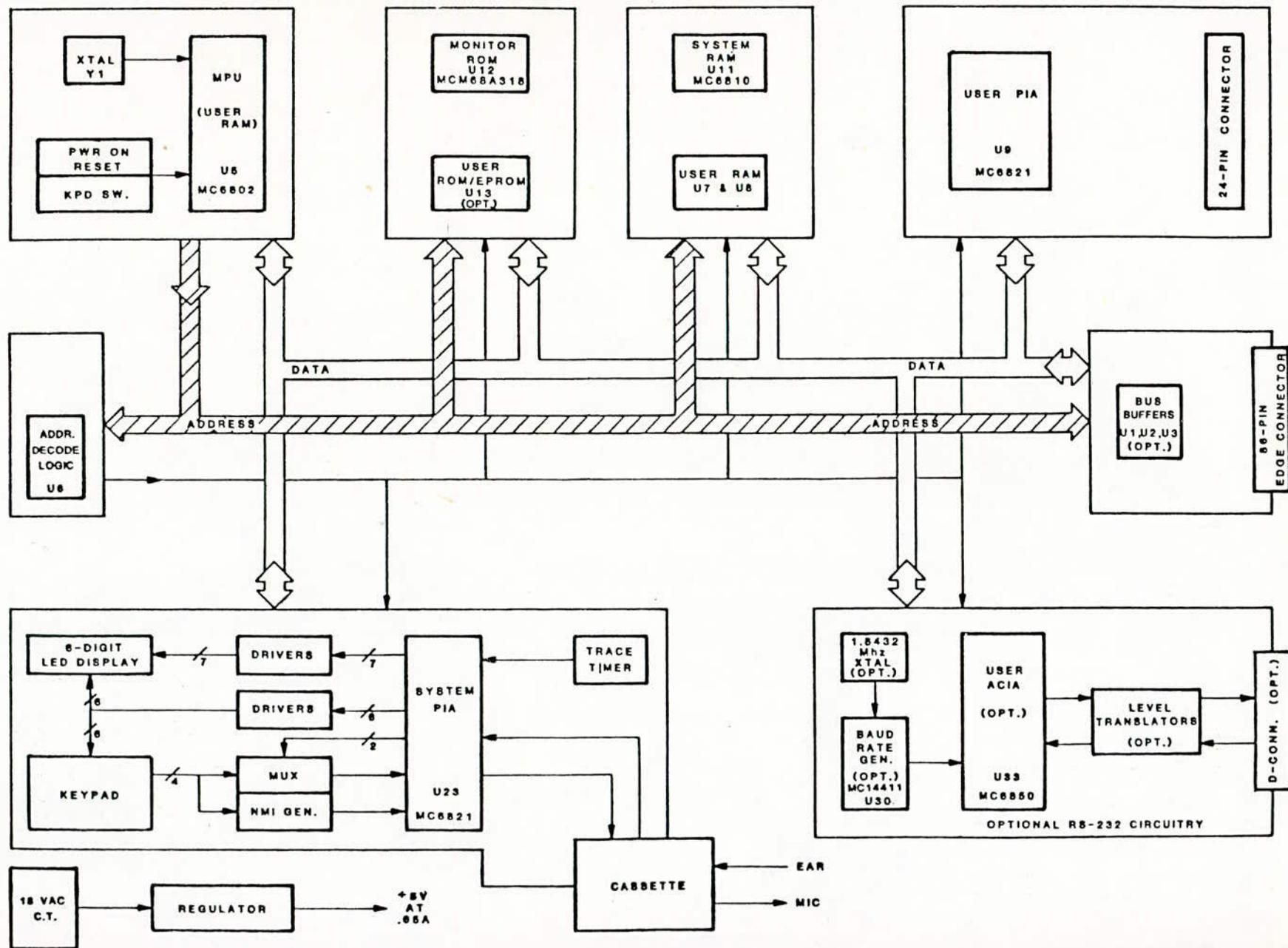


FIGURE 4-1 BLOCK DIAGRAM

		MODES D'ADRESSAGE										OPER. LOGIQUE/ARITHM.	COND. CODE REG.					
OPERATIONS	MNEMONIQUE	IMMED.		DIRECT		INDEX		ETENDU		IMPLICITE		(Toutes les étiquettes de registre se réfèrent au contenu)	5	4	3	2	1	0
		OP	#	OP	#	OP	#	OP	#	OP	#		H	I	N	Z	V	C
Add	ADD	88	2 2	98	3 2	A8	5 2	BB	4 3			A + M · A
	ADDB	CB	2 2	DB	3 2	E8	5 2	FB	4 3			B · M · B	
Add Acmltrs	ABA									18	2 1	A · B · A	
Add with Carry	ADCA	89	2 2	99	3 2	A9	5 2	B9	4 3			A + M · C · A	
	ADCB	C9	2 2	D9	3 2	E9	5 2	F9	4 3			B + M · C · B	
And	ANDA	84	2 2	94	3 2	A4	5 2	B4	4 3			A · M · A	
	ANDB	C4	2 2	D4	3 2	E4	5 2	F4	4 3			B · M · B	
Bit-Test	BITA	85	2 2	95	3 2	A5	5 2	B5	4 3			A · M	
	BITB	C5	2 2	D5	3 2	E5	5 2	F5	4 3			B · M	
Clear	CLR											00 · M	
	CLRA									4F	2 1	00 · A	
	CLRB									5F	2 1	00 · B	
Compare	CMPA	B1	2 2	91	3 2	A1	5 2	B1	4 3			A - M	
	CMPB	C1	2 2	D1	3 2	E1	5 2	F1	4 3			B - M	
Compare Acmltrs	CBA									11	2 1	A - B	
Complement, 1's	COM					63	7 2	73	6 3			M · M	
	COMA									43	2 1	A · A	
	COMB									53	2 1	B · B	
Complement, 2's (Negate)	NEG					60	7 2	70	6 3			00 M · M	
	NEGA									40	2 1	00 - A · A	
	NEGB									50	2 1	00 B · B	
Decimal Adjust, A	DAA									19	2 1	Convertit. add. binaire de caract. DCB en format DCB	
Decrement	DEC					6A	7 2	7A	6 3			M - 1 · M	
	DECA									4A	2 1	A - 1 · A	
	DECB									5A	2 1	B - 1 · B	
Exclusive OR	EORA	88	2 2	98	3 2	A8	5 2	B8	4 3			A ⊕ M · A	
	EORB	C8	2 2	D8	3 2	E8	5 2	F8	4 3			B ⊕ M · B	
Increment	INC					6C	7 2	7C	6 3			M + 1 · M	
	INCA									4C	2 1	A + 1 · A	
	INCB									5C	2 1	B + 1 · B	
Load Acmltr.	LDAA	86	2 2	96	3 2	A6	5 2	B6	4 3			M · A	
	LDAB	C6	2 2	D6	3 2	E6	5 2	F6	4 3			M · B	
Or, Inclusive	ORAA	8A	2 2	9A	3 2	AA	5 2	BA	4 3			A + M · A	
	ORAB	CA	2 2	DA	3 2	EA	5 2	FA	4 3			B + M · B	
Push Data	PSHA									36	4 1	A - Msp, SP - 1 -> SP	
	PSHB									37	4 1	B - Msp, SP - 1 -> SP	
Pull Data	PULA									32	4 1	SP + 1 -> SP, Msp -> A	
	PULB									33	4 1	SP + 1 -> SP, Msp -> B	
Rotate Left	ROL					69	7 2	79	6 3			M	
	ROLA									49	2 1	A	
	ROLB									59	2 1	B	
Rotate Right	ROR					66	7 2	76	6 3			M	
	RORA									46	2 1	A	
	RORB									56	2 1	B	
Shift Left, Arithmetic	ASL					68	7 2	78	6 3			M	
	ASLA									48	2 1	A	
	ASLB									58	2 1	B	
Shift Right, Arithmetic	ASR					67	7 2	77	6 3			M	
	ASRA									47	2 1	A	
	ASRB									57	2 1	B	
Shift Right, Logic	LSR					64	7 2	74	6 3			M	
	LSRA									44	2 1	A	
	LSRB									54	2 1	B	
Store Acmltr.	STAA			97	4 2	A7	6 2	B7	5 3			A - M	
	STAB			D7	4 2	E7	6 2	F7	5 3			B - M	
Subtract	SUBA	80	2 2	90	3 2	A0	5 2	B0	4 3			A - M · A	
	SUBB	C0	2 2	D0	3 2	E0	5 2	F0	4 3			B - M · B	
Subtract Acmltrs	SBA									10	2 1	A - B · A	
Subtr. with Carry	SBCA	82	2 2	92	3 2	A2	5 2	B2	4 3			A - M - C · A	
	SBCB	C2	2 2	D2	3 2	E2	5 2	F2	4 3			B - M - C · B	
Transfer Acmltrs	TAB									16	2 1	A - B	
	TBA									17	2 1	B - A	
Test, Zero or Minus	TST					6D	7 2	7D	6 3			M - 00	
	TSTA									4D	2 1	A - 00	
	TSTB									5D	2 1	B - 00	

LEGENDE:

- OP Code op (hexa);
- ~ Nbre de cycles d'UC;
- # Nbre d'octets de progr.;
- + Plus arithm.;
- Moins arithm.;
- ET logique;

- Msp Contenu de la mém. pointée par le point. de pile;
- + OU logique;
- ⊙ XOUI logique;
- M Complément de M;
- Transféré à;
- 0 Bit = Zéro;

- 00 Octet = Zéro;
- H Demi-retenu du bit 3;
- I Masque d'interruption;
- N Bit de signe;
- Z Zéro (octet);
- V Dépassem., comp. à 2;
- C Retenu du bit 7;

- R Réinit. toujours;
- S Init. toujours;
- ! Test et init. si vrai; sinon RAZ;
- Non affecté;
- CCR Registres des codes condition;
- LS Le moins significatif;
- MS Le plus significatif.

(a) Instructions accumulateur et mémoire

Fig. 11-1 L'ensemble d'instructions du 6800 (Courtesy of Motorola, Inc.)

OPERATIONS DE POINTEUR MNEMONIQUE		IMMED.			DIRECT			INDEX			ETENDU			IMPLICITE			OPER. LOGIQUE/ ARITHM.	5 4 3 2 1 0					
		OP	~	#	OP	~	#	OP	~	#	OP	~	#	OP	~	#		H	I	N	Z	V	C
Compare Index Reg	CPX	8C	3	3	9C	4	2	AC	6	2	8C	5	3				$X_H - M, X_L - (M + 1)$	•	•	①	!	②	•
Decrement Index Reg	DEX													09	4	1	$X - 1 - X$	•	•	•	!	•	•
Decrement Stack Pntr	DES													34	4	1	$SP - 1 - SP$	•	•	•	•	•	•
Increment Index Reg	INX													08	4	1	$X + 1 - X$	•	•	•	!	•	•
Increment Stack Pntr	INS													31	4	1	$SP + 1 - SP$	•	•	•	•	•	•
Load Index Reg	LDX	CE	3	3	0E	4	2	EE	6	2	FE	5	3				$M - X_H, (M + 1) \cdot X_L$	•	•	③	!	R	•
Load Stack Pntr	LDS	8E	3	3	9E	4	2	AE	6	2	BE	5	3				$M \rightarrow SP_H, (M + 1) \rightarrow SP_L$	•	•	③	!	R	•
Store Index Reg	STX				0F	5	2	EF	7	2	FF	6	3				$X_H - M, X_L - (M + 1)$	•	•	③	!	R	•
Store Stack Pntr	STS				9F	5	2	AF	7	2	BF	6	3				$SP_H \cdot M, SP_L \rightarrow (M + 1)$	•	•	③	!	R	•
Idx Reg - Stack Pntr	TXS													35	4	1	$X - 1 - SP$	•	•	•	•	•	•
Stack Pntr - Idx Reg	TSX													30	4	1	$SP + 1 - X$	•	•	•	•	•	•

(b) Instructions de manipulation de pile et de registre d'index

OPERATIONS		MNEMONIQUE	RELATIVE			INDEX			ETENDU			IMPLICITE			TEST BRANCH	5 4 3 2 1 0							
			OP	~	#	OP	~	#	OP	~	#	OP	~	#		H	I	N	Z	V	C		
Branch Always	BRA	20	4	2										Aucun	•	•	•	•	•	•	•	•	
Branch If Carry Clear	BCC	24	4	2										C = 0	•	•	•	•	•	•	•	•	
Branch If Carry Set	BCS	25	4	2										C = 1	•	•	•	•	•	•	•	•	
Branch If = Zero	BEQ	27	4	2										Z = 1	•	•	•	•	•	•	•	•	
Branch If >= Zero	BGE	2C	4	2										$N \oplus V = 0$	•	•	•	•	•	•	•	•	
Branch If > Zero	BGT	2E	4	2										$Z + (N \oplus V) = 0$	•	•	•	•	•	•	•	•	
Branch If Higher	BHI	22	4	2										$C + Z = 0$	•	•	•	•	•	•	•	•	
Branch If <= Zero	BLE	2F	4	2										$Z + (N \oplus V) = 1$	•	•	•	•	•	•	•	•	
Branch If Lower Or Same	BLS	23	4	2										$C + Z = 1$	•	•	•	•	•	•	•	•	
Branch If < Zero	BLT	2D	4	2										$N \oplus V = 1$	•	•	•	•	•	•	•	•	
Branch If Minus	BMI	2B	4	2										N = 1	•	•	•	•	•	•	•	•	
Branch If Not Equal Zero	BNE	26	4	2										Z = 0	•	•	•	•	•	•	•	•	
Branch If Overflow Clear	BVC	28	4	2										V = 0	•	•	•	•	•	•	•	•	
Branch If Overflow Set	BVS	29	4	2										V = 1	•	•	•	•	•	•	•	•	
Branch If Plus	BPL	2A	4	2										N = 0	•	•	•	•	•	•	•	•	
Branch To Subroutine	BSR	8D	8	2											•	•	•	•	•	•	•	•	
Jump	JMP				6E	4	2	7E	3	3				Voir opér. part.	•	•	•	•	•	•	•	•	
Jump To Subroutine	JSR				AD	8	2	BD	9	3					•	•	•	•	•	•	•	•	•
No Operation	NOP												01	2	1	Avanc. comm. prog. seulement	•	•	•	•	•	•	•
Return From Interrupt	RTI												3B	10	1		•	•	•	•	•	•	•
Return From Subroutine	RTS												39	5	1	Voir opér. part.	•	•	•	•	•	•	•
Software Interrupt	SWI												3F	12	1		•	•	•	•	•	•	•
Wait for Interrupt*	WAI												3E	9	1	•	•	•	•	•	•	•	•

(c) Instructions de saut et de branchement

OPERATIONS		MNEMONIQUE	IMPLICITE			OPER. LOGIQUES	5 4 3 2 1 0					
			OP	~	#		H	I	N	Z	V	C
Clear Carry	CLC	0C	2	1	0 → C	•	•	•	•	•	•	R
Clear Interrupt Mask	CLI	0E	2	1	0 → I	•	R	•	•	•	•	•
Clear Overflow	CLV	0A	2	1	0 → V	•	•	•	•	•	R	•
Set Carry	SEC	0D	2	1	1 → C	•	•	•	•	•	•	S
Set Interrupt Mask	SEI	0F	2	1	1 → I	•	S	•	•	•	•	•
Set Overflow	SEV	08	2	1	1 → V	•	•	•	•	•	S	•
Acmltr A → CCR	TAP	06	2	1	A → CCR	①						
CCR → Acmltr A	TPA	07	2	1	CCR → A	②						

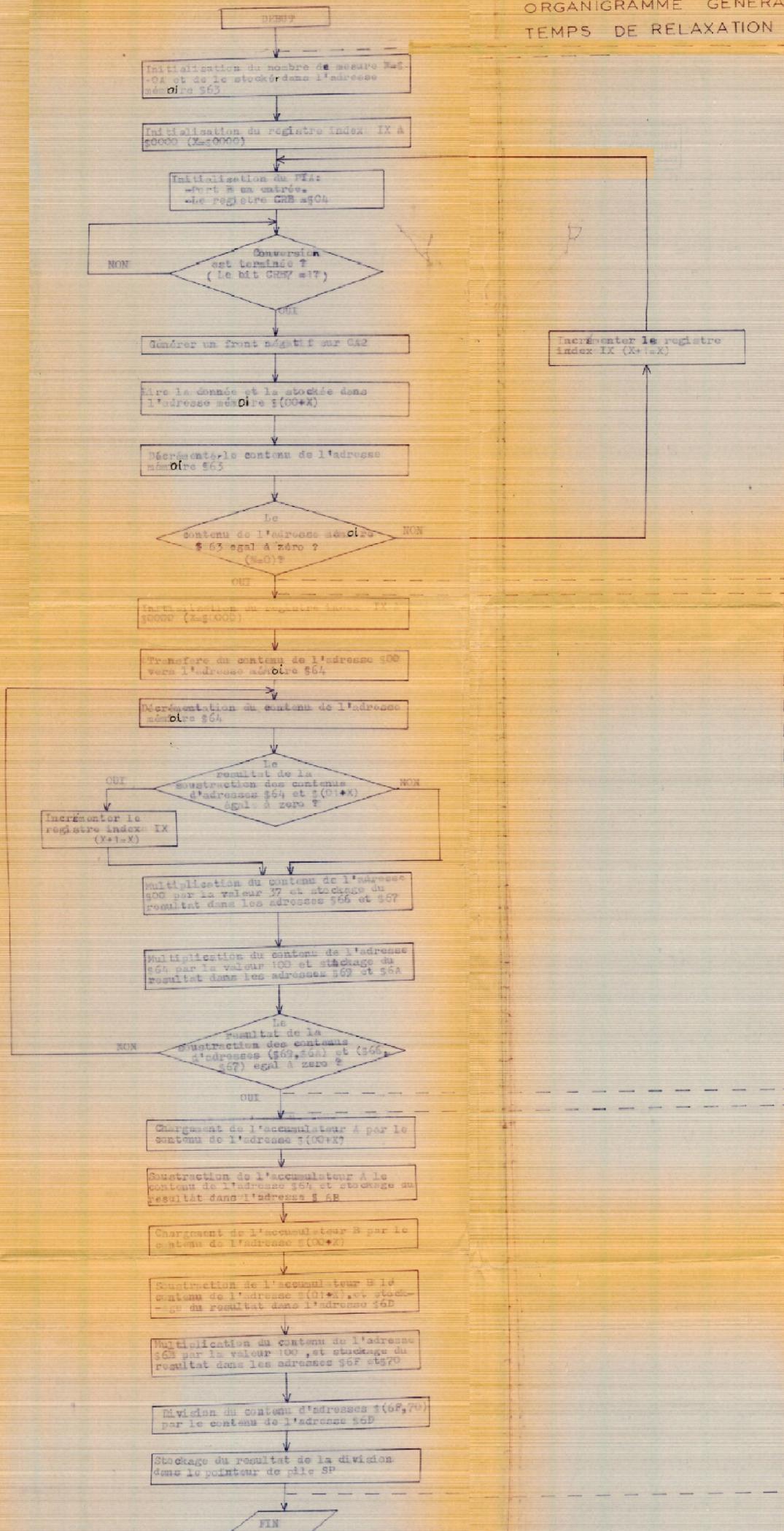
NOTES CONDITION CODE REGISTRE :

- (Bit V) Test : Résultats = 10000000 ?
- (Bit C) Test : Résultats ≠ 00000000 ?
- (Bit C) Test : Valeur décimale du carac. DCB MS > 9 ? (Pas RAZ si init. précédemment)
- (Bit V) Test : Opérande = 10000000 avant exécution ?
- (Bit V) Test : Opérande = 01111111 avant exécution ?
- (Bit V) Test : Init. égal au résultat de N ≠ C après décal.
- (Bit N) Test : Bit de signe de l'OLRS du résultat = 1 ?
- (Bit V) Test : Dépas. compl. à 1 après soust. des OLPS ?
- (Bit N) Test : Résultat < 0 ? (Bit 15 = 1)
- (All) Charger CCR à partir de la pile (voir opér. part.)
- (Bit I) Init. à interrupt. Si init. préalablement, une interrr non masquable est nécessaire pour sortir de l'état WAIT
- (Tous) Init. selon contenu de l'accumulateur A.

(c) Instructions de manipulation du CCR.

Fig. 11-1 (suite)

ORGANIGRAMME GENERAL POUR CALCULER LE TEMPS DE RELAXATION SPIN-RESEAU : T₁



ACQUISITION DE DONNÉES

DECREMENTATION DE LA PREMIERE DONNEE JUSQU'A 37% DE SA VALEUR INITIALE.

CALCUL DE $L_T = \frac{M_0(n) - M_0(t)}{M_0(n) - M_0(n+1)} \cdot n$