

وزارة التعليم و البحث العلمي

MINISTERE DE L'ENSEIGNEMENT ET DE LA RECHERCHE SCIENTIFIQUE

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT D'ELECTRONIQUE



PROJET DE FIN D'ETUDES

SUJET

CONCEPTION ET REALISATION
d'une CARTE D'ACQUISITION RAPIDE

Proposé par: M^{me} Beddek

Dirigé par: M^{me} Beddek

Etudié par:

Réda Baba Aïssa

Rabia Seffouh

DEDICACES

A la mémoire de mon père
A ma mère
A mon frère
A ma belle soeur
A mes soeurs
A mes neveux
A toute ma famille
A tous mes amis

RABIA

المدرسة الوطنية المتعددة التخصصات
المكتبة — BIBLIOTHÈQUE
Ecole Nationale Polytechnique

A ma mère
A mon père
A mes frères et soeurs
A ma famille
A tous mes amis

MOHAMED REDA

REMERCIEMENTS

Nous tenons à remercier particulièrement Madame BEDEK par l'aide précieuse qu'elle nous a fourni ainsi que les conseils qu'elle nous a prodigués tout au long de ce semestre.

Que Monsieur SADOIN trouve en ces lignes l'expression de notre profonde reconnaissance pour l'aide qu'il nous a lui aussi apportée.

Soient également remerciés tous ceux qui nous ont apporté leur aide pour la mise en forme de ce fascicule.



INTRODUCTION	1
CHAPITRE I : Présentation et conception de la carte d'acquisition rapide	
I-1 : Présentation de la carte	2
I-2 : Conception de la carte	3
I-3 : Echantillonneur/ Bloqueur	5
I-4 : Conversion A/N	7
I-5 : Registre à décalage SN74LS164	14
I-6 : Conversion N/A	17
CHAPITRE II : Présentation de la carte TM 990/189	
Modes d'entrées/sorties	
2-1 : Description de la carte TM 990/189	20
2-2 : Le microprocesseur TMS 9980	20
2-3 : Les Bus	22
2-4 : La mémoire	23
2-5 : Les entrées/sorties	26
CHAPITRE III : L'accès direct à la mémoire	
Etude du contrôleur DMA MC 6844	
3-1 : L'accès direct à la mémoire	30
3-2 : Etude du contrôleur DMA MC 6844	31
CHAPITRE IV : Mise en oeuvre de la carte	
4-1 : Utilisation du DMAC dans la carte d'acquisition	40
4-2 : Adaptation du DMAC à la carte TM 990/189	40
4-3 : Connexions du DMAC avec les convertisseurs	42
4-4 : Sélection du DMAC	45
4-5 : Programmation du DMAC	46
4-6 : Utilisation des DAC 830 dans la carte	47
4-7 : Fonctionnement du registre à décalage	50
4-8 : Utilisation d'un CAN en basses fréquences	53
4-9 : Fonctionnement général de la carte d'acquisition	56
CONCLUSION :	59

- INTRODUCTION -

Lorsqu'on veut réaliser un système d'acquisition de données dans le but de faire le traitement d'un signal analogique, il faut tenir compte d'un critère de choix : la vitesse d'acquisition de ces données.

Ainsi en basses fréquences, les variations d'un signal analogique sont lentes, et donc un système d'acquisition de données comprend un seul convertisseur analogique-numérique de vitesse moyenne.

Le transfert des données numériques en mémoire de l'unité de traitement, peut se faire par programme. Mais, pour un signal de fréquence assez élevée, il devient nécessaire d'effectuer une conversion A/N très rapide et d'utiliser un mode d'entrée-sortie non moins rapide, pour le transfert des données en mémoire.

C'est ainsi que dans notre projet, on a été confronté à ce genre de problème.

En effet, dans le but de réaliser à l'aide de la carte TM 990/189, le traitement d'un signal dont la fréquence peut varier jusqu'à 10 khz, il nous a été demandé de concevoir et réaliser une carte d'acquisition rapide.

Ce qui va caractériser cette carte, c'est que :

- la conversion A/N est basée sur l'utilisation de plusieurs convertisseurs de vitesse moyenne ce qui évite de recourir à l'emploi d'un convertisseur rapide, mais très coûteux.
- le transfert des données en mémoire se fera par l'intermédiaire du mode d'entrées-sorties le plus rapide : la DMA (accès direct à la mémoire)
- Elle comportera un bloc de conversion numérique-analogique (N/A) indépendant de la partie acquisition, qui permet au microprocesseur de générer 2 tensions continues utilisées par la carte analogique d'où provient notre signal.

- CHAPITRE I -

PRESENTATION ET CONCEPTION DE
 LA CARTE D'ACQUISITION RAPIDE

I-1. PRESENTATION DE LA CARTE

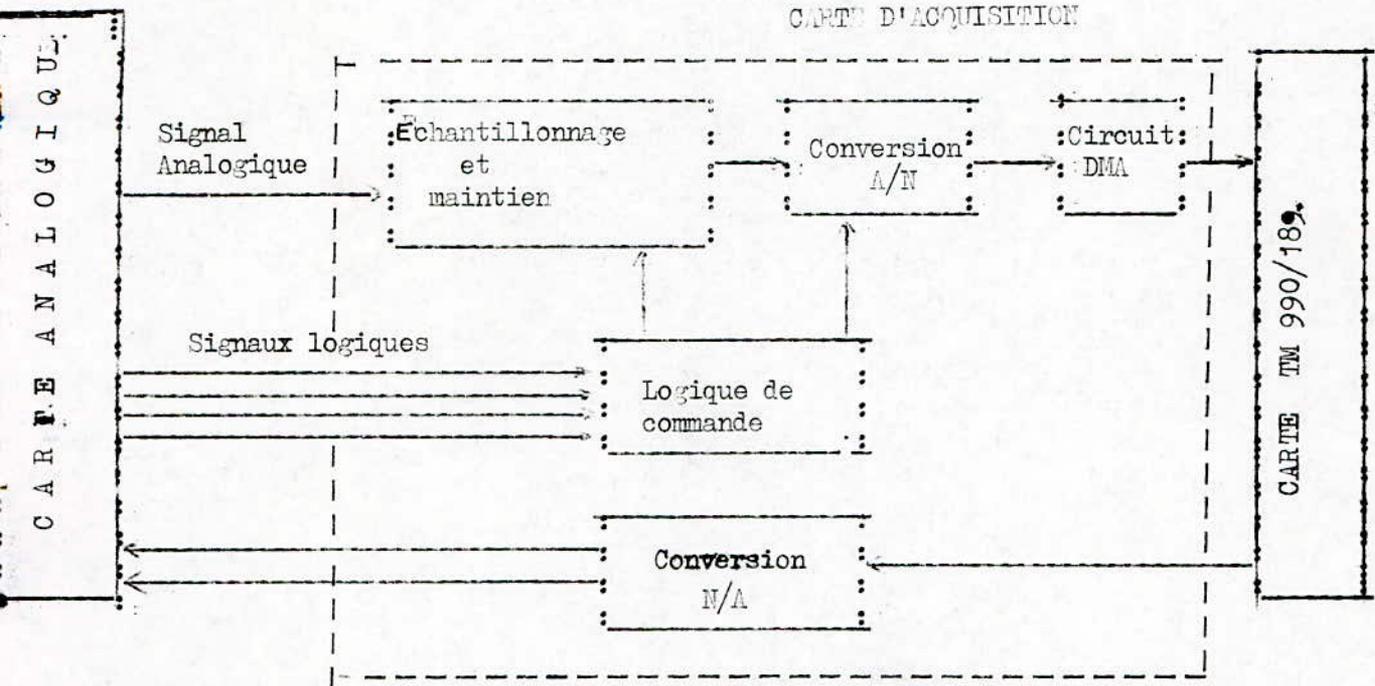


Fig. 1-1. Synoptique général de la carte d'acquisition

Comme le montre la fig. 1-1, la carte d'acquisition constitue une interface entre un microordinateur (la carte TM 990/189) et une carte analogique qui délivre un signal dont la fréquence peut varier de 0,01 hz à 10 khz.

Cette carte aura deux fonctions essentielles :

- a) l'acquisition rapide des données du signal analogique à cause des fréquence élevées qu'il peut atteindre. Cette acquisition va se traduire par :
 - l'échantillonnage du signal analogique puis le maintien de la valeur des échantillons
 - la conversion A/N des échantillons analogiques.

- le transfert des données numériques en mémoire de la carte TM 990/189 par l'intermédiaire d'un circuit de DMA.

b) La conversion numérique-analogique de données provenant du microprocesseur et ce, dans le but de générer 2 tensions continues utilisées par la carte analogique.

Cette carte comprend aussi, un circuit logique de commande associé aux convertisseurs A/N et aux échantillonneurs. Et pour créer les signaux de commande nécessaires, ce circuit disposera de 4 signaux logiques de fréquence : $8w, 4w, 2w$ et w (w étant la fréquence du signal analogique) qui seront issus de la carte analogique. Tout ceci sera d'ailleurs expliqué plus en détail, un peu plus loin.

L'ensemble carte-analogique-carte numérique constitue en quelque sorte une chaîne complète d'acquisition de données.

1-2 : CONCEPTION DE LA CARTE

Cette carte va se charger de convertir un signal analogique dont la fréquence est variable et peut atteindre 10 khz. Nous allons pour cela, utiliser des C A N dont le temps de conversion est de l'ordre de $100 \mu s$. Ce temps est trop long pour permettre à un seul convertisseur de réaliser correctement la conversion du signal aux fréquences élevées. Et sachant qu'il faut tenir compte du théorème de Shannon qui indique la nécessité de prendre au moins 2 échantillons par période, pour réaliser une bonne conversion du signal. Il nous faudra donc associer plusieurs C A N pour convertir notre signal aux fréquences élevées.

D'autre part, pour ces fréquences élevées un signal analogique présente des variations d'amplitude très importantes pour des temps très courts comme on peut le voir sur un exemple.

Pendant la durée T_c de conversion d'un C A N qui est de $100 \mu s$, un signal de forme sinusoïdale d'amplitude $E = 5 \text{ v}$ présente une variation d'amplitude ΔV telle que :

$$\Delta V = 2 \pi f E T_c$$

si l'on considère que $f = 1 \text{ khz}$

$$\Delta V = 2 \times 3,14 \times 10^3 \times 5 \times 10^{-4} = 3,14 \text{ v}$$

Il est donc évident qu'un CAN ne peut convertir un signal qui varie autant pendant la conversion. D'où la nécessité d'utiliser un échantillonneur bloqueur qui permet d'acquérir un échantillon en un temps très court, puis de maintenir la valeur constante pendant le temps nécessaire à la conversion.

Afin de déterminer le signal avec une bonne précision, nous allons acquérir 8 échantillons/période.

Ainsi pour $f \leq 5$ khz, la période minimale est de 200 μ s.

Donc un CAN peut convertir jusqu'à 2 éch./période.

Et l'utilisation de 4 convertisseurs permet la conversion des 8 échantillons/période.

De cette façon :

- le 1er échantillon est converti par le 1er CAN
- le 2ème " " " par le 2ème CAN
- le 4ème " " " par le 4ème CAN
- le 5ème " " " par le 1er CAN qui aura fait la conversion du 1er échantillon. Cela se poursuit ainsi jusqu'au 8ème échantillon qui sera converti par le 4ème CAN.

Pour faire fonctionner les 4 CAN de cette façon, il faut donc que les 4 éch./ Bloqueurs fonctionnent de la même manière en prélevant chacun, un échantillon à tour de rôle. Cela va nécessiter la création de 4 signaux de commande logiques et chacun d'eux servira à la commande d'échantillonnage puis à déclencher la conversion de l'échantillon prélevé.

Ces 4 signaux devront être décalés chacun par rapport au précédent comme le montre la figure 1-2 .

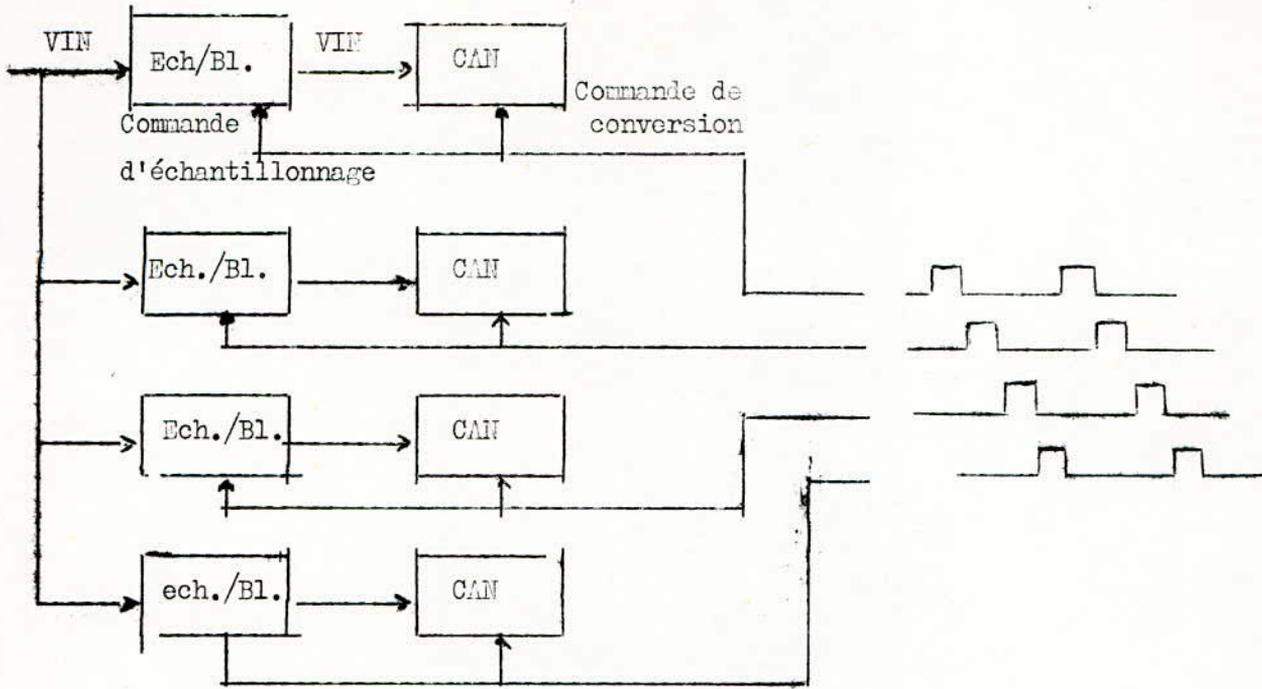


Figure 1-2 .

Les 4 signaux logiques peuvent être générés à partir d'un registre à décalage série/ parallèle, dont l'entrée d'horloge sera le signal carré 8 w de la carte analogique.

Il faut noter enfin que pour $5 \text{ khz} \ll f \ll 10 \text{ khz}$, il n'est plus possible de convertir 8 éch./période car un CAN n'a le temps de convertir qu'un seul échantillon pendant chaque période du signal. On est donc contraint dans ce cas, de se limiter à l'acquisition de 4 échantillons par période.

I - 3 : ECHANTILLONNEUR / BLOQUEUR .

I-3-1 PRINCIPE : L'échantillonnage d'un signal consiste à prélever sa valeur instantanée d'une manière périodique.

La fréquence d'échantillonnage F_e doit obligatoirement obéir au théorème de Shannon qui dit que :

" Un signal $V(t)$ dont la composante spectrale maximale est F_{max} est entièrement déterminé par la suite complète de ses échantillons si $F_e \geq 2 F_{\text{max}}$ " .

Une fois l'échantillon prélevé, il est mémorisé dans un circuit de maintien et cela pendant toute la durée de la conversion T_c car la tension à l'entrée du CAN doit rester constante pendant T_c .

Cette fonction d'échantillonnage et le maintien est réalisée par l'échantillonneur/ Bloqueur.

I-3-2 PRESENTATION ET FONCTIONNEMENT DE L'ECH./BLOQUEUR LF 398

Pour notre carte nous avons choisi comme ech./Bloqueur le LF 398.

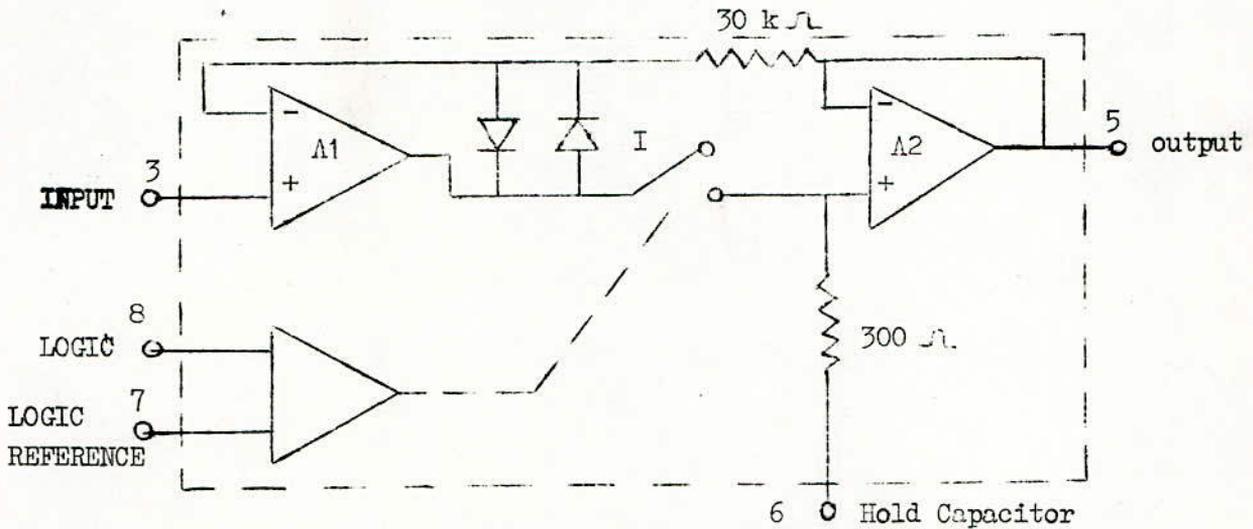
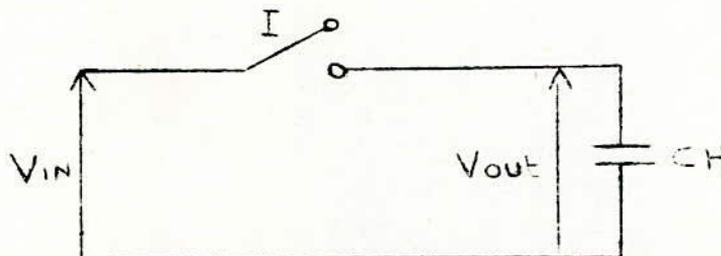


Fig.1-3. Schéma fonctionnel du LF 398

La structure interne du LF 398 (fig. 1-3) comprend :

- 2 amplis opérationnels montés en suiveurs A1 et A2
- 2 diodes montées en tête bêche, afin d'éviter la saturation de A1, en fixant le gain de celui-ci.
- Un interrupteur I
- Une résistance R1 qui sert à limiter le courant entre A1 et A2

Le fonctionnement du LF 398 peut être expliqué à l'aide du schéma suivant;



- Mode échantillonnage:

Sous l'action d'un signal de commande (niveau logique 1) I se ferme et CH se charge à la tension V_{in} . On définit alors un temps d'acquisition (T_{acq}) qui correspond au temps mis par V_{out} pour atteindre la valeur V_{in} .

- Mode maintien :

Il est obtenu en appliquant un niveau logique 0 sur l'entrée logique du LF 398 donc I s'ouvre . CH ne peut alors se décharger car A2 a une impédance d'entrée très grande. Le temps de maintien est égal à T_c .

Remarque : En réalité CH ne reste pas tout à fait constante pendant la période de maintien car il existe des courants de fuite qui la font se décharger lentement. Ce qui peut entraîner des erreurs dans la conversion. Pour éviter ces erreurs, il faut choisir des capacités à faible courant de fuite.

I-4. CONVERSION A/N

I-4-1 : PRINCIPE : La conversion analogique-numérique consiste à représenter une information analogique par une expression numérique avec une précision et une résolution données.

Le circuit chargé d'effectuer cette conversion est appelé convertisseur analogique-numérique (CAN).

Pour faire une conversion ,il faut disposer d'une référence. Pour un convertisseur A/N électronique, cette référence est une tension de valeur connue : V_{ref} .

Quand il veut convertir une tension analogique V , le CAN va calculer le rapport $\frac{V}{V_{ref}}$. Cela revient à effectuer une division par les puissances décroissantes de 2 .

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

La longueur du mot binaire est limitée à n bits compte tenu de la résolution du système. D'où, l'on néglige le terme $\frac{b_{n+1}}{2^{n+1}}$. V_{ref} qui est appelé : erreur de quantification ou erreur de conversion.

La conversion A/N est donc une opération de quantification consistant à remplacer une grandeur analogique V par un nombre entier qui peut prendre des valeurs discrètes et qui est multiple d'une quantité élémentaire V_{ref} appelée : quantun.

Un CAN a trois caractéristiques essentielles :

- La Résolution

Elle représente le nombre de bits utilisés par le CAN. Elle définit la plus petite variation perçue par le convertisseur (quantun)

- Le temps de conversion :

Il représente le temps nécessaire au convertisseur pour donner la valeur numérique correspondant au signal analogique appliqué.

- La précision :

C'est l'écart entre la caractéristique de transfert d'un CAN idéal et celle d'un CAN réel.

Cet écart est exprimé en nombres de L S B (Last significant bit) $1 \text{ LSB} = \frac{V_{ref}}{2^n}$.

Il existe plusieurs types de convertisseurs :

- les CAN analogiques qui comprennent les CAN à simple rampe, double rampe etc...
- les CAN logiques appelés ainsi car ils sont constitués de composants logiques, dont les plus répandues sont des CAN à approximation successives.

Pour notre part, nous utiliserons des CAN à approximations successives car ce sont les plus performants du point de vue qualité-prix. Ils ont notamment des temps de conversion assez courts par rapport aux autres convertisseurs.

I-4-2 ETUDE DU CONVERTISSEUR ADC 804

Dans notre carte nous allons utiliser les CAN ADC 804 qui appartiennent au type de convertisseurs à approximation successives.

A/ PRINCIPE ET FONCTIONNEMENT D'UN CAN A APPROXIMATIONS SUCCESSIVES :

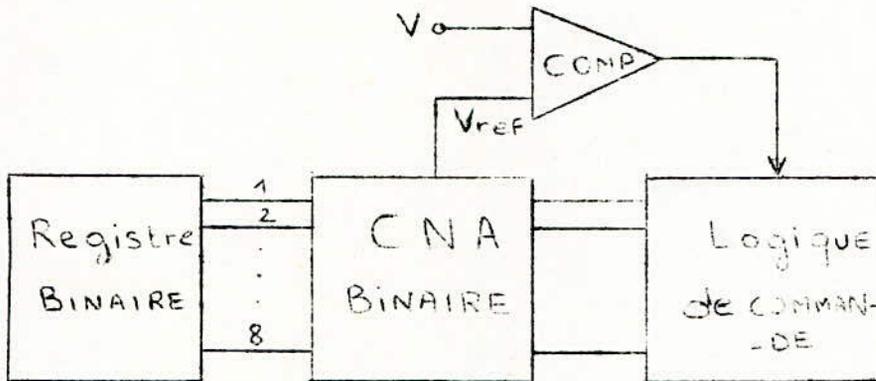


Fig (1-4) schéma de principe d'un CAN à approximation successives.

Le principe de ce type de CAN repose sur l'idée de la pesée par essais successifs.

Ainsi si on veut mesurer un poids x inférieur par exemple à 1 kg, on commence par mettre un poids de 500 g, s'il est insuffisant pour basculer la balance, on rajoute un autre poids inférieur à 500g, sinon on enlève le poids de 500g pour mettre à la place un poids inférieur. On poursuit ainsi les essais et ce, jusqu'à la détermination du poids x avec la précision désirée.

Pour le CAN, c'est le même principe sauf que le poids est remplacé par une grandeur électrique : une tension.

Le CAN à approximation successives (fig. 1-4) est constitué par :

- Un comparateur (qui joue le rôle de la balance) recevant sur l'une de ses entrées la tension V à convertir, et sur l'autre une tension de référence.

- un convertisseur N/A ayant le même nombre de bits que le CAN et qui aura pour rôle de générer une tension de référence suivant le résultat de la comparaison.

- Une logique de commande qui sert à mettre les bits à 1 ou 0 en commençant par le bit de poids fort (H S B)

- Un registre binaire contenant le résultat de la comparaison.

B/ FONCTIONNEMENT DE L'ADC 804

C'est un convertisseur dont la résolution est de 8 bits. Son bus de Données (DBO - DB7) à 3 états peut être connecté directement au bus de données du microprocesseur.

Le convertisseur comporte 3 entrées de commande

- CS (Chip select) qui est active à l'état bas et qui sert à la sélection du boîtier.
- WR : le passage de l'état bas à l'état haut du signal appliqué à cette entrée permet de lancer la conversion.
- RD : La présence d'un état bas à cette entrée permet la lecture de l'information binaire en sortie du CAN.

Une sortie INTR signale la fin de conversion en passant de l'état haut à l'état bas.

Diagrammes de temps

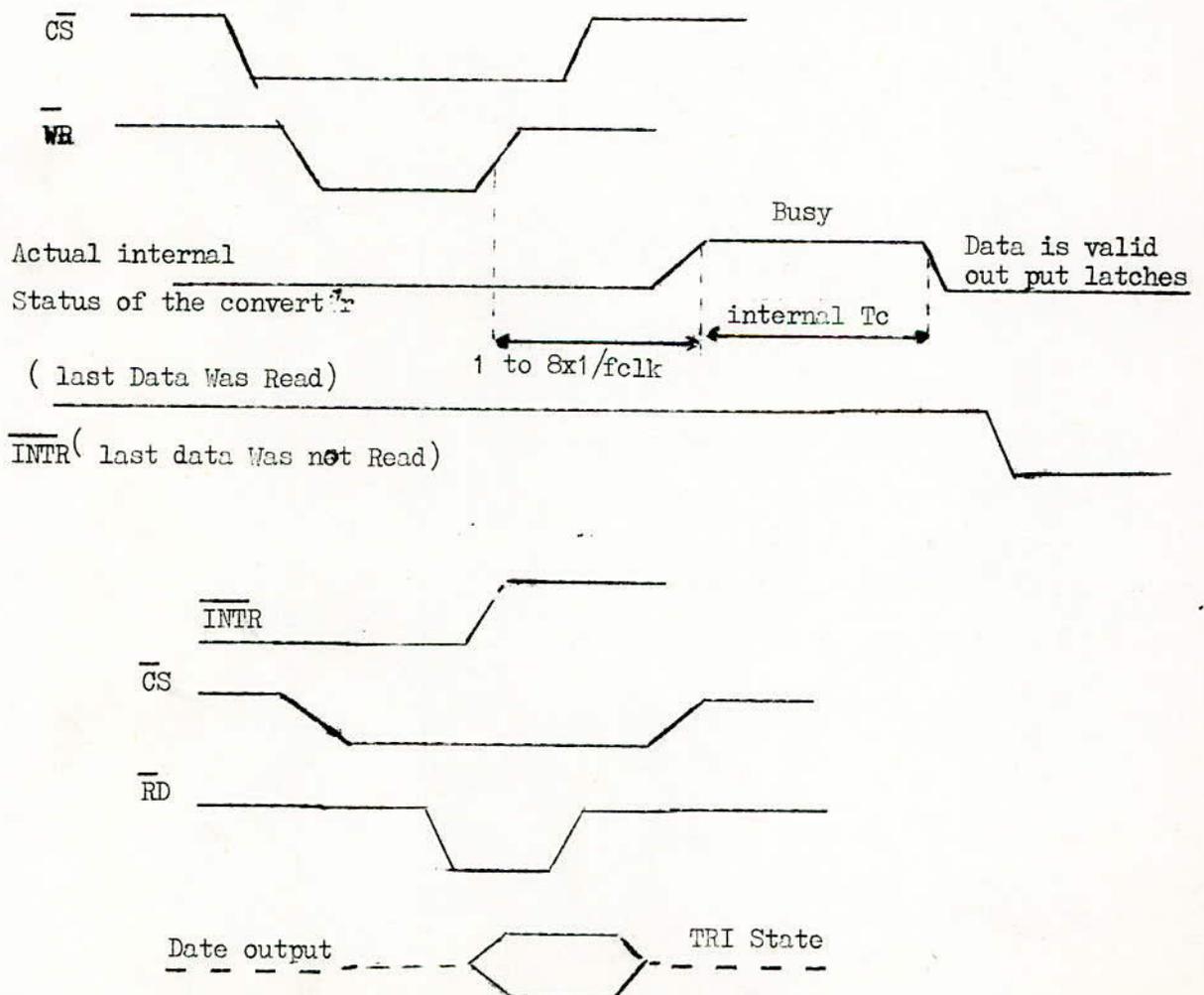
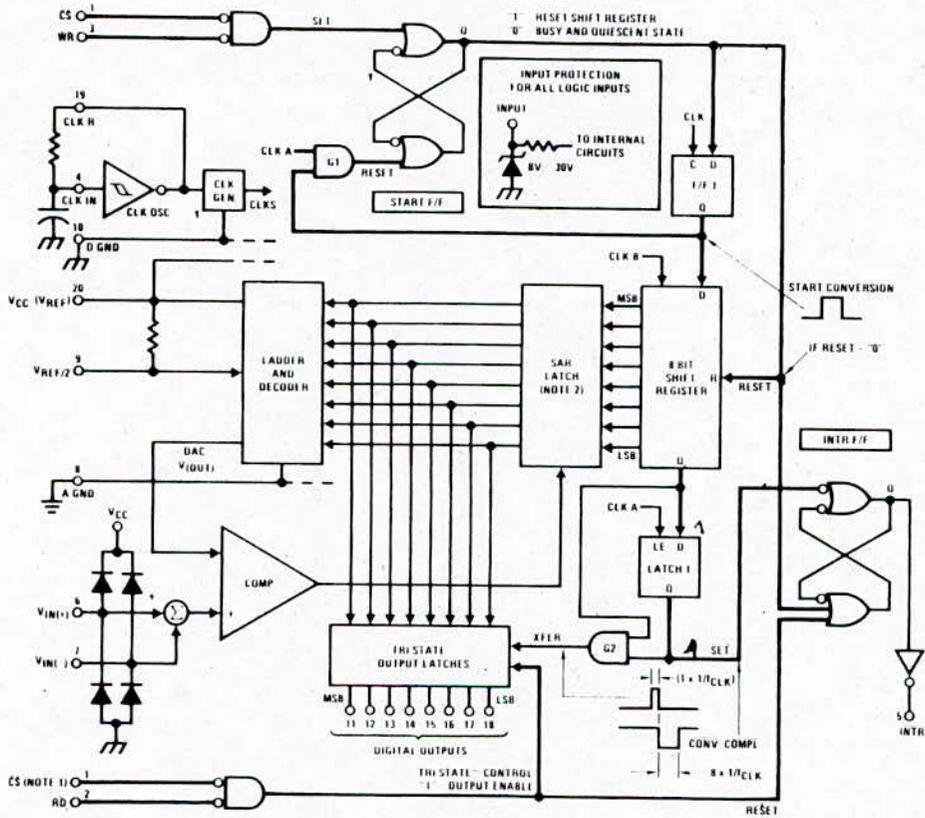


Fig. 1-5. Cycle de lecture et de conversion de l'ADC 804



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

Schema fonctionnel de l'ADC 804

Quand on applique une tension VIN à l'entrée du CNA, celle-ci est convertie après les étapes suivantes.

- Phase d'initialisation

L'application simultanée d'un niveau bas sur \overline{CS} et \overline{WR} provoque le passage à 1 de la sortie de la bascule RS " Start F/F " ou a alors

- initialisation du registre à décalage " Shift register "
- mise à 1 de la sortie INTR
- Au top d'horloge CLK la bascule D " F/F1 " passe à 1 .

Phase de conversion

La passage de \overline{WR} de l'état bas à l'état haut va permettre de lancer la conversion.

En effet, au top d'horloge CLKA, il y a remise à 0 de la sortie de la bascule " Start F/F " éliminant ainsi une seconde initialisation pendant la conversion.

° Au top d'horloge CLKB le 1 présent à l'entrée du registre à décalage est transmis au MSB du SAR (successive approximation register) le CNA convertit cette valeur numérique en une grandeur analogique qui sera comparée à VIN. Et suivant le résultat de la comparaison le MSB du SAR est laissé à 1 ou mis à 0.

° Au top d'horloge suivant, il y a décalage du 1 présent dans le registre à décalage vers le bit suivant du SAR et une nouvelle comparaison est lancée.

Après 8 comparaisons, on obtient la valeur numérique N correspondant à la tension VIN.

A la fin de la conversion le LSB est transmis sur l'entrée de la bascule D " F/F2 " provoquant ainsi le passage de l'INTR de l'état haut à l'état bas.

Le signal X FER en passant à 1 transfère l'information numérique dans le registre " out put latches " à 3 états.

Phase de lecture

Un état bas appliqué simultanément sur \overline{CS} et \overline{RD} envoie l'information numérique stockée dans " out put latches " sur le bus de donnée ce qui rend possible sa lecture.

° Celà provoque aussi la mise à 0 de la sortie Q de la bascule INTR F/F donc le passage de $\overline{\text{INTR}}$ de l'état bas à l'état haut.

I-4-3 CONVERSION D'ECHANTILLONS POSITIFS ET NEGATIFS

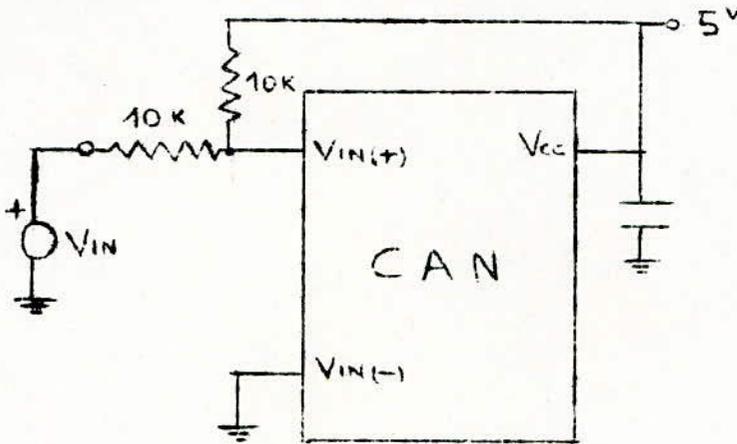


Fig. 1-6

L'ADC 804 est conçue pour convertir des tensions positives comprises entre 0 et 5 v. Comme notre signal analogique présente des alternances positives et négatives, les échantillons négatifs ne pourront être convertis.

Pour résoudre ce problème on réalise le montage de la figure (1-6) qui permet à l'ADC de convertir un signal dont l'amplitude peut varier entre - 5V et + 5V.

Ainsi, le - 5V du V_{IN} correspond au 0V du convertisseur

le + 5V du V_{IN} correspond au 5 V du convertisseur.

1-5 REGISTRE A DECALAGE SN74 LS 164

(Voir brochage en annexe)

1-5-1 Table de vérité

Entrées				Sortie			
Clear	Clock	A	B	Q A	Q B	Qh	
0	X	X	X	0	0	0	
1	0	X	X	Q A0	Q B0	Qh0	
1		1	1	1	Q An	Qh	
1		0	X	0	Q An	Qh	
1		X	0	0	Q An	Qh	

Q_{x0} : état de la sortie X avant que le signal d'horloge soit à 0.

Q_{xn} : état de la sortie X avant le front montant de l'horloge.

1-5-2 Principe de fonctionnement

C'est un registre à entrées séries et sorties parallèles. Il est constitué de 8 bascules en cascade (sortie de l'une, reliée à l'entrée de la suivante). Ainsi par exemple, lorsqu'un " 1 " apparaît à la sortie de la 1ère bascule (QA), il sera décalé en sortie de la bascule suivante, à chaque front montant de l'horloge (CLOCK)

1-5-3 .Utilisation du registre comme circuit de commande

Les 4 sorties QA, QB, QC, QD donneront les signaux de commande des échantillonneurs et des CAN.

Chaque sortie Q sera :

- reliée directement à l'entrée logique d'un échantillonneur
- inversée puis reliée au \overline{WR} d'un convertisseur.

De cette façon, lorsqu'elle est à l'état 1 l'échantillonneur procède à l'acquisition d'un échantillon du signal, puis lorsque Q repasse à 0, l'échantillonneur se retrouve dans le mode de maintien et en même temps, la conversion est lancée car \overline{WR} passe de 0 à 1 .

Dans le cas où l'on doit acquérir 8 éch./ période, on applique à l'entrée d'horloge du registre (CLOCK), le signal carré 8 w fourni

par la carte analogique. Et connaissant les états qu'on désire obtenir pour les 4 sorties, il reste à déterminer à partir de la table de vérité, les signaux x à appliquer aux entrées A et B du registre.

	Q A	Q B	Q C	Q D	A	B
1er échantillon:	1	0	0	0	1	1
2° échantillon :	0	1	0	0	0	1
3° " :	0	0	1	0	0	1
4° " :	0	0	0	1	0	1
5° " :	1	0	0	0	1	1
6° " :	0	1	0	0	0	1
7° " :	0	0	1	0	0	1
8° " :	0	0	0	1	0	1

Les chronogrammes de ces différents signaux sont donnés par la figure (1-7).

Le signal B doit être mis à l'état logique " 1 " tandis que le signal A aura l'allure donnée par le chronogramme correspondant.

Pour générer ce dernier signal, on dispose des signaux carrés fournis par la carte analogique.

Ainsi on peut voir qu'en procédant à un " ET " logique entre les 2 signaux de fréquence $4w$ et $2w$, on obtient exactement le signal A (fig. 1-9)

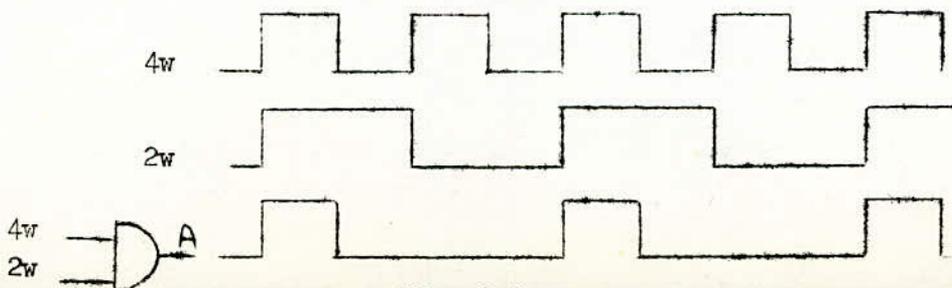


Fig. 1-9

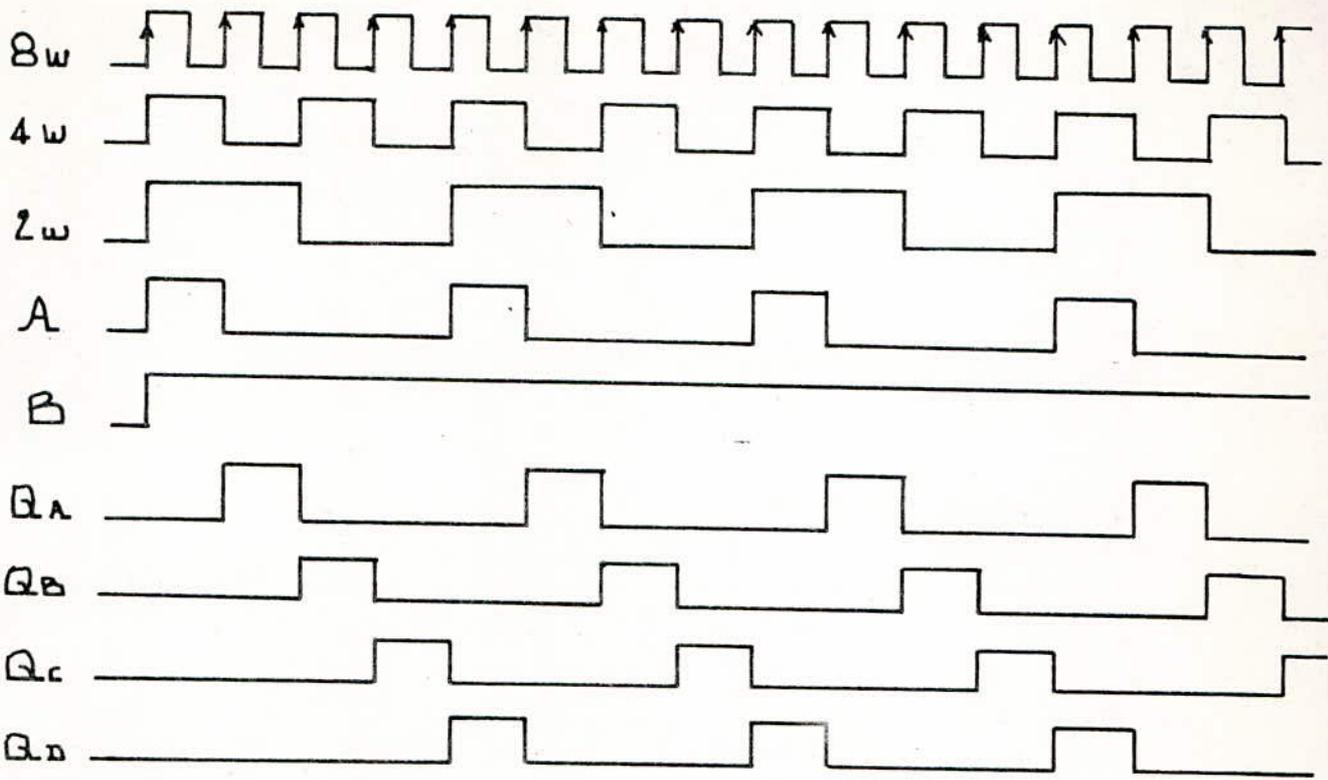


fig. 1-7 Chronogrammes du registre à décalage pour $f \leq 5 \text{ KHz}$

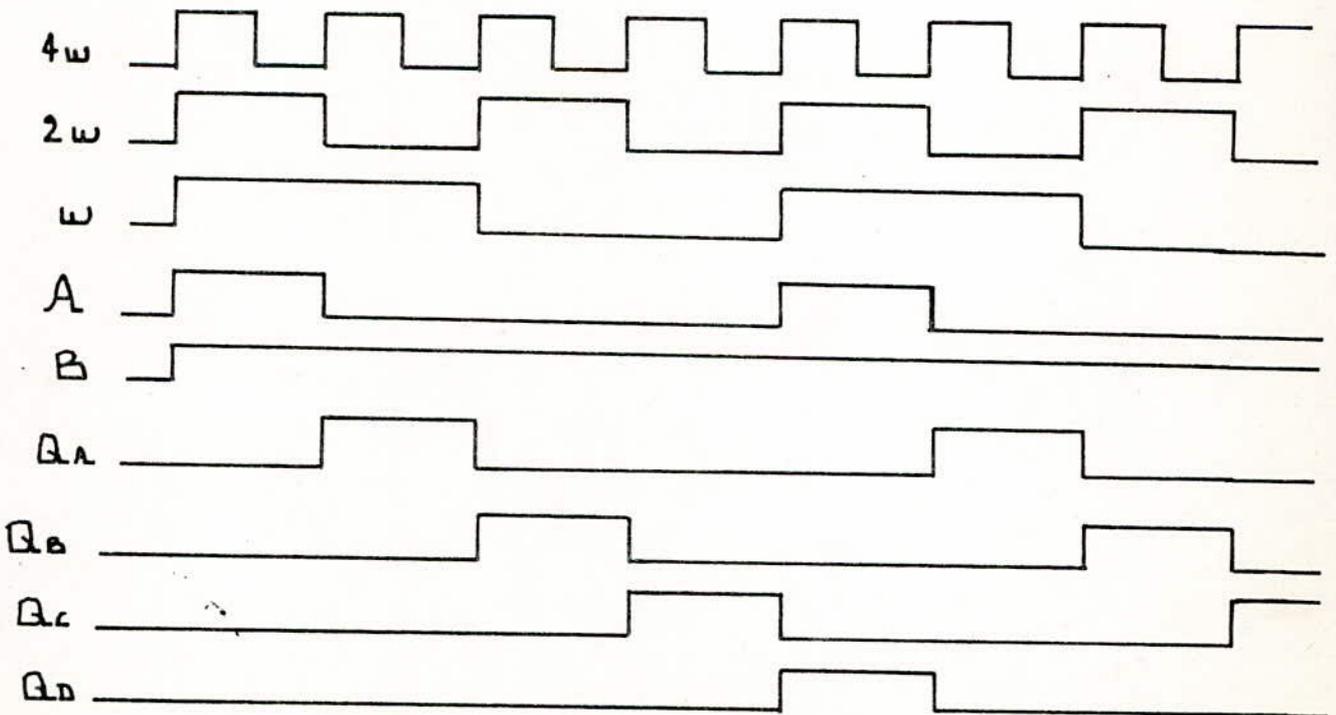


fig. 1-8 Chronogrammes du registre à décalage pour $5 \text{ KHz} < f \leq 10 \text{ KHz}$

Dans le cas où l'on doit acquérir seulement 4 éch./période ($5 \text{ kHz} \leq f \leq 10 \text{ kHz}$) il faudra donc remplacer le signal d'horloge du registre par le signal $4w$. Un nouveau signal A est aussi nécessaire (fig. 1-8) il sera généré cette fois à partir d'un " ET " logique entre les signaux $2w$ et w .

Le passage de la 1ère configuration (8 éch./période à la seconde (4 éch./période) rendue possible par l'utilisation de commutateurs, comme nous le verrons dans le chapitre consacré à la mise en oeuvre de la carte (chap. 4).

I-6 CONVERSION NUMERIQUE-ANALOGIQUE

1-6-1 Principe :

La conversion numérique-analogique consiste à transformer une information numérique de n bits en un signal analogique, grâce à un dispositif qu'on appelle : convertisseur numérique-analogique (CNA)

Si N est le nombre entier qu'on veut convertir, il peut s'écrire :

$$N = d_{n-1} 2^{n-1} + d_{n-2} 2^{n-2} + \dots + d_0 2^0$$

où d_{n-1} est le MSB et d_0 le LSB.

au nombre N on fait correspondre la quantité analogique

$$A = q (d_{n-1} 2^{n-1} + d_{n-2} 2^{n-2} + \dots + d_0 2^0)$$

où q est la quantité élémentaire analogique qu'on appelle aussi : quantum.

En sortie du CNA, on aura une tension ou un courant dont l'amplitude sera $V_s = A$.

Les paramètres caractéristique d'un CNA sont les mêmes que ceux du CAN :

- la résolution
- la précision
- le temps de conversion

1-6-2 Convertisseur N/A D A C 830

Dans notre carte numérique nous allons utiliser 2 CNA pour générer à partir d'informations numériques issues du microprocesseur 2 tensions continues. Les convertisseurs utilisés sont des DAC 830;

Principe de fonctionnement

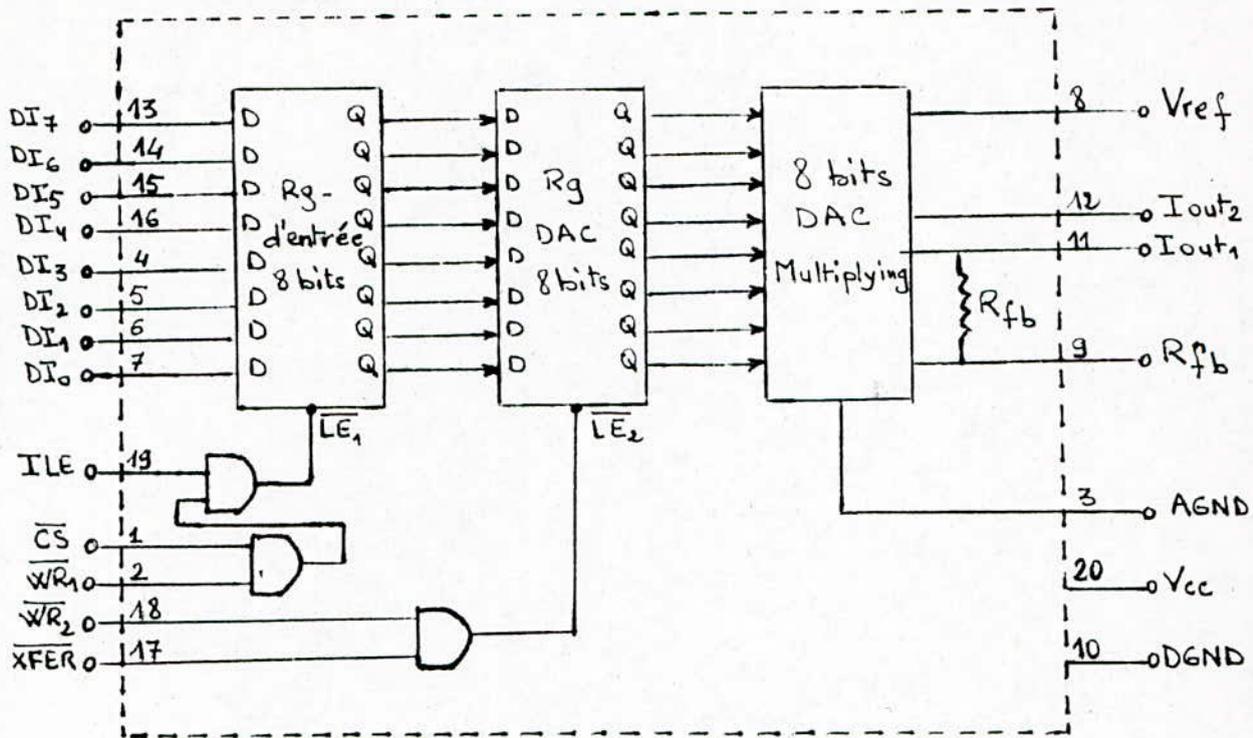


Fig. 1-11 Schéma interne du DAC 830

La DAC 830 a une résolution de 8 bits, il est compatible avec la plupart des microprocesseurs, notamment avec le TMS 9980.

Ce convertisseur comprend 2 registres internes qui sont des bascules D à verrouillage, lui servant à mémoriser les données numériques pendant un certain temps. Ces registres sont commandés par les signaux $\overline{LE1}$ et $\overline{LE2}$

$$\overline{LE1} = \overline{ILE} \cdot (\overline{CS} \cdot \overline{WR1})$$

$$\overline{LE2} = \overline{WR2} \cdot \overline{XFER}$$

* Quand \overline{CS} (chip select) est à 1 la donnée numérique est stockée dans le registre d'entrée.

* Quand \overline{CS} est à 0, le $\overline{WR1}$, $\overline{WR2}$ et \overline{XFER} sont à l'état "0" et que \overline{ILE} est à l'état 1, les données sont transférées vers le bloc de conversion (DAC multiplying) dont le principe de fonctionnement est basé sur la conversion en échelle à réseau de résistance R - 2R.

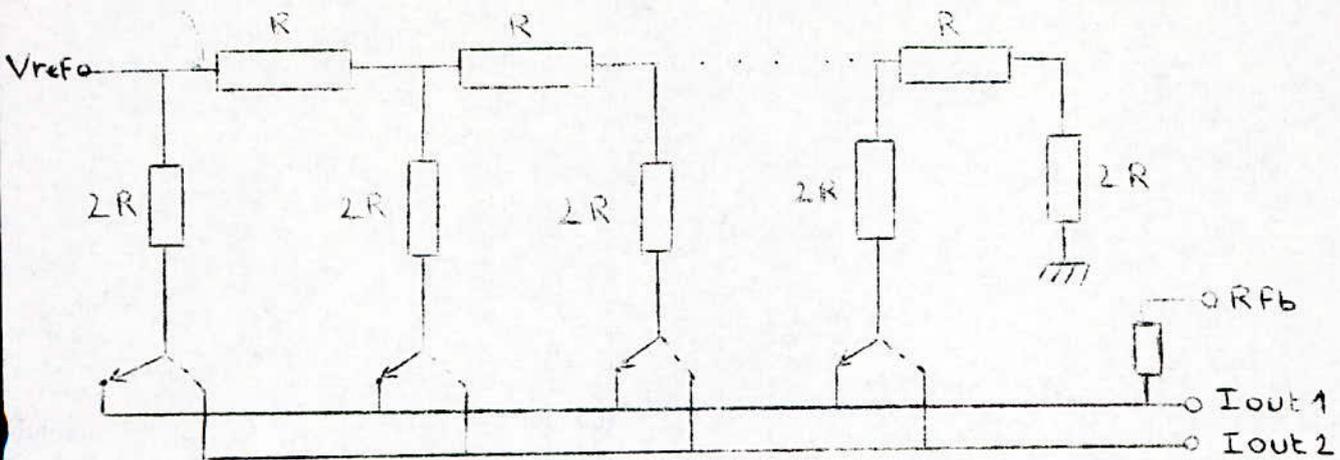


FIG. 1-12

Le réseau de résistance R-2R (fig.1-12) est équivalent à une résistance $R_n = 15 \text{ k}\Omega$

En sortie, on récupère un courant $I_{out 1}$, proportionnel à la tension de référence (V_{ref}) et à la quantité digitale N présente à l'entrée du CNA.

$$I_{out 1} = \frac{V_{ref}}{R_n} \cdot \frac{N}{2^8}$$

Le courant $I_{out 2}$ est proportionnel à V_{ref} et à la quantité digitale complémentaire de celle présente en entrée.

$$I_{out 2} = \frac{V_{ref}}{R_n} \cdot \frac{255 - N}{2^8}$$

Pour convertir le courant de sortie en tension, on utilise un amplificateur opérationnel en appliquant $I_{out 1}$ sur l'entrée inverseuse. Le montage comprendra aussi une résistance de contre réaction égale à R_n ($15 \text{ k}\Omega$)

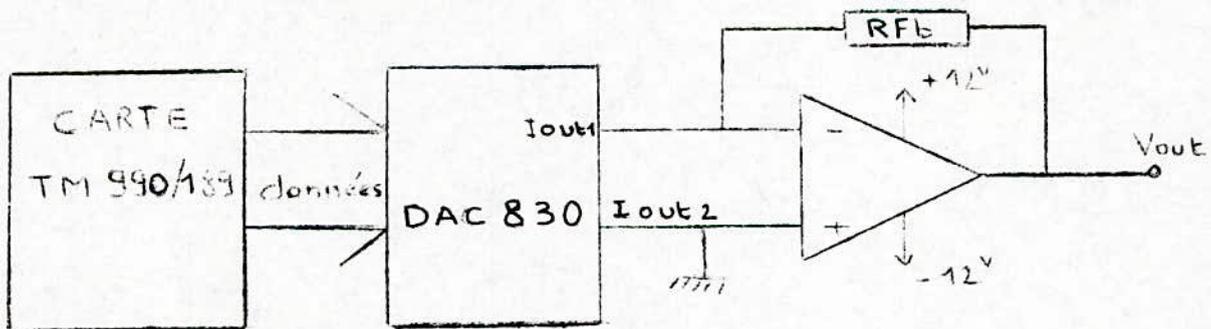


FIG. 1-13

Ainsi, la tension de sortie analogique est donnée par l'expression :

$$V_{out} = V_{ref} \left(\frac{N}{2^8} \right)$$

- CHAPITRE II -

PRESENTATION DE LA CARTE TM 990/189

MODES D'ENTREES/SORTIES

I-INTRODUCTION :

La dernière étape de l'acquisition de données du signal analogique, sera le transfert des données numériques en mémoire d'un micro-ordinateur : la carte TM 990/189. Le rôle de cette carte étant d'effectuer par la suite le traitement de ces données.

Aussi, nous avons pensé qu'il était nécessaire de consacrer un chapitre à la présentation de la carte TM 990/189. Il s'agira d'avoir une idée générale de son fonctionnement, et de s'intéresser plus particulièrement à l'organisation de la mémoire, aux signaux de contrôle et aux modes d'entrées/sorties et ce, pour mieux comprendre de quelle façon se fera le transfert des données.

2 - 1 DESCRIPTION DE LA CARTE TM 990/189 :

Le système TM 990/189 est un microordinateur sur carte dont les principaux composants sont :

- le microprocesseur TMS 9980
- la mémoire qui se divise en mémoire vive (RAM), et en mémoire morte (ROM)
- une voie d'entrée sortie programmable sur 16 bits et contrôle des interruptions (TMS 9901)
- une interface d'extension de bus
- une interface pour cassette audio
- un clavier alphanumérique de 45 touches
- un dispositif de visualisation par LEDS
- des indicateurs optiques et acoustiques.

2 - 2 LE MICROPROCESSEUR TMS 9980

Il appartient à la famille 9900 de TEXAS Instruments.

Il comprend l'unité centrale qui travaille sur 16 bits et une horloge de fréquence 2 Mhz intégrée au boîtier.

Le TMS 9980 possède les caractéristiques suivantes :

- * un bus de données de 8 bits
- * une mémoire extensible jusqu'à 16.384 octets
- * 4 niveaux d'interruptions hiérarchisés.
- * la possibilité d'entrées-sorties par accès direct à la mémoire (DMA)

2-3. LES BUS

2-3-1- Bus de données

Il se compose de 8 lignes bidirectionnelles D₀ à D₇ (D₀ poids fort, D₇ poids faible). Ces lignes canalisent les échanges d'information, entre le processeur et la mémoire, que celle-ci soit située sur la carte ou hors-carte à travers l'interface d'extension de bus.

2-3-2. Bus adresse

C'est un bus unidirectionnel qui se compose de 14 lignes A₀ à A₁₃/CRUOUT (A₀ poids fort, A₁₃ poids faible).

Ces lignes servent à repérer un emplacement mémoire ou CRU donné.

2-3-3. Bus CRU

Ce bus est utilisé lors des transferts d'entrées/sorties par le CRU.

Il se compose de 4 signaux : A₁₃/CRUOUT, CRUCLK, IOCLK et CRUIN.

Pendant les instructions CRU d'émission de données, on trouve sur A₁₃/CRUOUT l'état du bit émis, alors que pour les instructions de réception de données, l'état du bit est présent sur CRUIN.

2-3-4. Le bus de contrôle

IL regroupe les divers signaux de contrôle qui servent au processeur et à ses circuits annexes :

- * MEMEN (Memory enable) : C'est un signal de validation mémoire qui est mis à 0 par le CPU lorsqu'il désire lire ou écrire dans une mémoire ou un périphérique implanté en zone mémoire.
- * WE (Write enable) : signal de validation d'écriture mémoire qui est mis à 0 par le processeur quand il désire écrire dans une mémoire ou un périphérique situé en zone mémoire.
- * DBIN (Data-Bus-In) : Ce signal est mis à 1 quand le CPU veut lire sur son bus de données.
- * READY : Utilisé par la mémoire et les périphériques qui y sont implantés pour notifier au processeur qu'ils sont prêts à échanger des informations.
- * HOLD : Utilisé par le contrôleur DMA, pour indiquer au CPU qu'un transfert est demandé.
- * HOLDA : Signal d'acquiescement fourni par le CPU, pour indiquer au contrôleur DMA, que la demande de transfert est prise en compte.
- * $\overline{\phi}$: Signal d'horloge émis par le processeur pour assurer sa synchronisation avec les dispositifs externes.
- * IA $\overline{\phi}$ (Signal instruction-acquisition) : A l'état haut ce signal indique

que le processeur lit en mémoire, le premier mot de l'instruction.

* CRUCLK : Signal d'horloge utilisé pour échantillonner les données présentes en série sur la ligne CRUOUT (A 13)

* CKIN : Horloge externe.

2 - 4 LA MEMOIRE

2-4-1. Organisation de la mémoire

Elle se répartit en :

- 2K octets de mémoire vive (RAM)
- 4K octets de mémoire morte (ROM) que l'on peut étendre sur la carte à 6 K octets.

Les RAM sont destinés à contenir les programmes et les données. Alors que les ROM contiennent l'assembleur et le moniteur (UNIBUG) qui servent à la gestion de la carte.

L'extension ROM est possible sur ou hors carte de même, l'extension RAM est possible hors carte. On peut d'ailleurs le voir sur la fig. (2-2) qui montre le découpage de la mémoire.

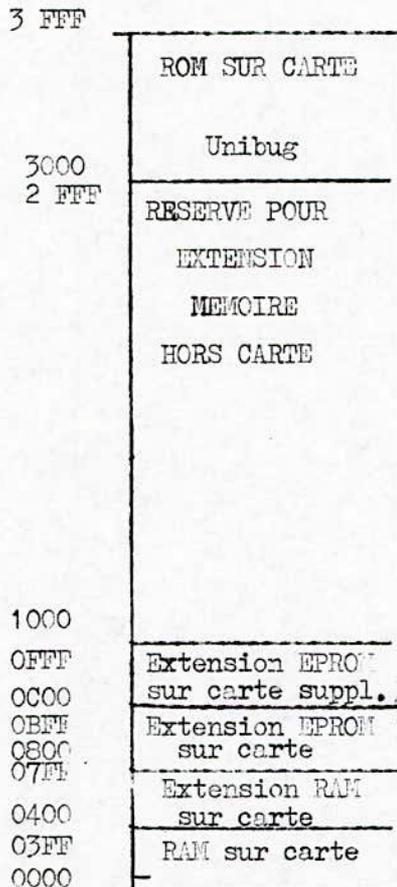


Fig. 2-2. Découpage de la mémoire

Ainsi, avec ses 14 lignes adresses, le TMS 9980 peut accéder à un total de 16.384 cases mémoires contenant un octet chacune. 8192 adresses sont réservées à des dispositifs sur carte, et le microprocesseur travaille sur des mots de 16 bits. Chaque mot est formé de 2 octets consécutifs. Pour l'octet de poids fort $A_{13} = 0$, alors que pour l'octet de poids faible $A_{13} = 1$.

2-4-2. Extension de la mémoire hors carte

Les adresses mémoires comprises entre 1000 et 2 FFF sont réservées à une éventuelle extension de la mémoire hors carte.

Cette extension est rendue possible grâce à la présence sur la carte d'une interface d'extension de bus comprenant 3 buffers chargés d'amplifier les signaux correspondants au bus de données, au bus adresse et au bus de contrôle.

2-4-3. Extension mémoire SIHQ 092

La carte TM 990/189 peut être reliée à l'extension mémoire SIHQ 092 (voir Schéma en annexe) qui comprend 4K octets de mémoire vive (RAM) et 2K octets ou 4K octets de mémoire morte (ROM).

La mémoire vive occupe les adresses comprises entre 1000 et 1 FFF et la mémoire morte occupe celles comprises entre 2000 et 27 FF ou 2000 et 2 FFF (selon qu'elle occupe 2 K ou 4 K octets).

Après installation de cette extension, la mémoire vive du système est portée à 6 K octets et la mémoire morte à 6 K octets (ou 8 K octets).

Pour notre part, nous aurons à utiliser cette extension mémoire pour y transférer les données converties par l'intermédiaire d'un accès direct à la mémoire (DMA). Et nous aurons l'occasion de revenir sur ce point dans le dernier chapitre.

2-4-4. Cycles mémoire

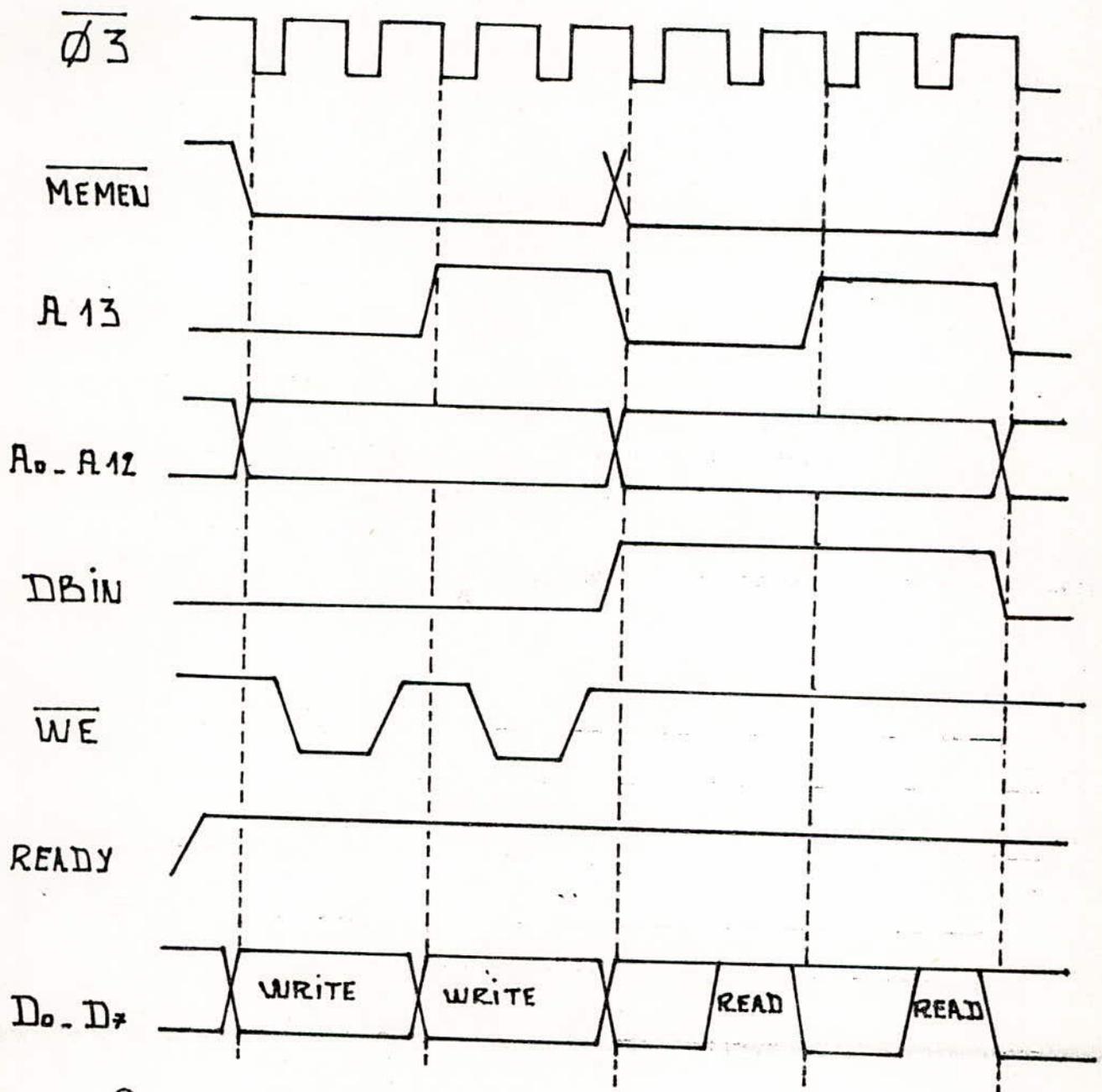
Les chronogrammes de ces cycles sont donnés par la figure 2-3.

Cycle de lecture

- * Les lignes d'adresses en provenance du CPU sont actives.
- * Le signal de validation de bloc est ensuite décodé (sélection du boîtier concerné)
- * Les données situées à l'adresse spécifiée sont présentées sur le bus de données.

Cycle d'écriture

- * Les lignes adresses sont actives
- * Le signal de validation de bloc (chip select) est décodé
- * Le signal \overline{WE} devient actif (état bas)
- * Les données provenant du CPU deviennent valides
- * \overline{WE} repasse à l'état bas ce qui provoque l'écriture des données en mémoire



Cycles d'écriture et de Lecture en mémoire

fig. 2-3

2 - 5: LES ENTREES / SORTIES (E/S)

Pour la carte TH 990/189, 3 modes d'entrées/sorties existent :

2-5-1. Entrelacement avec la memoire (memory mapped)

C'est un mode d'entrée/sortie contrôlé par programme.

Le périphérique est assimilé à une mémoire hors carte et on peut donc lui affecter une adresse.

Un programme, rangé en mémoire, ordonne au processeur d'émettre l'adresse du périphérique sur le bus adresse.

L'interface d'E/S décode l'adresse (sélection du périphérique) et une commande du CPU précise au périphérique la fonction qu'on attend de lui (lecture ou écriture). Ensuite l'échange des données peut se faire (voir fig. 2-4)

2-5-2.- Le CRU (Communication register unit)

C'est une voie d'E/S particulière, qui est spécifique à la famille 9900. Elle est appelée : registre de communication série ou CRU.

Ce mode utilise une interface spécifique (THS 9901) indépendant du bus de données, comprenant une logique interne à base de registre à décalage assurant :

- les E/S pilotées par programme
- les E/S pilotées par interruptions.

A.-E/S pilotées par programme

Elles se font sur 3 broches du THS 9980 :

- la broche CRUIN transmet en série 1 ou plusieurs bits vers le processeur
- la broche CRUOUT permet au processeur d'émettre 1 ou plusieurs bits en série vers un dispositif d'entrées/sorties.
- la broche CRUCLK émet un signal d'échantillonnage et de validation pour piloter les échanges de données.

Pendant l'exécution d'une instruction CRU soit en entrée, soit en sortie, l'adresse du bit du périphérique sélectionné apparaît sur les lignes A2 à A12 du bus adresse. Les adresses de bit CRU sont validés par le signal MEMEN qui est à l'état haut (alors qu'il est à l'état bas quand le bus adresse véhicule une adresse mémoire).

L'adresse d'un bit CRU est combinée à une valeur de base contenue dans le registre de travail R12. Cette valeur de base codée sur 11 bits, va utiliser les bits 4 à 14 de R12.

Le Principal avantage du CRU, c'est qu'il permet d'adresser un seul

bit à la fois pour le mettre soit à 0 soit à 1 ou bien contrôler son état. DE même qu'il permet de lire ou d'écrire par une seule instruction jusqu'à 16 bits.

B.- E/S pilotées par interruptions

Dans ce type d'entrées/sorties le périphérique qui est prêt à émettre n'a pas besoin d'attendre que le programme vienne le questionner. Par une interruption, ce périphérique demande directement au processeur de s'occuper de lui, quelque soit le programme qui était en train de se dérouler.

Le TMS 9901 peut reconnaître 15 interruptions qui peuvent être masquées individuellement. Ces interruptions sont hiérarchisées et un circuit encodeur interne permet de transmettre au CPU la demande d'interruption active la plus prioritaire.

Lorsque le processeur reçoit une demande d'interruption il sauvegarde un volume d'informations suffisant pour lui permettre de revenir plus tard au programme interrompu.

Ensuite, il donne le contrôle à une routine de prise en compte des interruptions qui se chargera du traitement de l'interruption détectée. Lorsque ce traitement est terminé, le contrôle revient au programme interrompu à l'endroit où il a été interrompu.

2.5.3.- L'accès direct à la mémoire : DMA (direct memory access)

Dans ce mode d'E/S, l'échange d'informations entre un périphérique et la mémoire peut se faire sans passer par le processeur. En effet, c'est un circuit spécialisé qu'on appelle " contrôleur DMA " qui va gérer cet échange de données.

Ainsi lorsqu'un périphérique veut accéder à la mémoire, le contrôleur DMA envoie au processeur un signal $\overline{\text{HOLD}}$ (demande de transfert DMA). Celui-ci envoie alors, au contrôleur DMA un signal HOLDA (acquittement du signal $\overline{\text{HOLD}}$ reçu) pour lui signifier qu'il lui abandonne le contrôle des bus du système.

Ainsi grâce à cette technique, on peut transférer un volume important de données de ou vers la mémoire, à très grande vitesse.

2.5.4.- Choix du mode d'E/S approprié

Il s'agit maintenant de savoir lequel des 3 modes d'E/S peut être utilisé pour la réalisation de notre carte d'acquisition rapide.

En comparant les 3 modes d'E/S, on peut arriver aux constatations

suivantes :

- * Le mode entrelacé avec la mémoire (memory mapped) est le plus simple, mais il est limité aux vitesses d'échanges moyenn-es car la durée du transfert des données en mémoire dépend du temps d'exécution du programme de contrôle.
- * Le CRU a l'avantage d'être indépendant du bus de données et de permettre l'adressage individuel de plusieurs bits,mais étant contrôlé lui aussi par programme,il reste soumis aux mêmes contraintes de vitesse que l'entrelaçage avec la mémoire .
- * Quant à la DMA c'est le mode d'E/S le plus compliqué,car il nécessite l'utilisation d'un circuit spécialisé pour contrôler le transfert des données (ce circuit n'est pas présent sur la carte TM 990/189). Mais l'avantage de la DMA,c'est qu'elle ne fait pas intervenir le processeur et donc elle permet d'effectuer des transferts de données très rapides vers la mémoire.

Ainsi la DMA est le mode de transfert le plus approprié à notre carte d'acquisition rapide. D'ailleurs nous pouvons le voir dans l'exemple qui suit.

Si l'on utilisait le CRU et l'entrelaçage avec la mémoire comme modes de transfert, on peut calculer le temps d'exécution d'un programme minimal destiné à transférer en mémoire le résultat d'une conversion A/N.

En supposant que l'adresse de sélection du convertisseur est 1 F 80, le programme est le suivant :

LP2 MOV R0, $\omega > 1 F 80$	Transfert du contenu de R0 à l'adresse 1 F 80 ce qui a pour effet de mettre cette adresse sur le bus adresse - donc sélection du CAN et la conversion est lancée.
LP1 - TB7	Test du bit 7 du CRU auquel est relié l' \overline{INTR} du CAN.
JEQ LP1	Saut à LP1 tant que \overline{INTR} est à 1. (attente de fin de conversion)
MOV $\omega > 1F80$, $\omega > 2F2$ (R1)	La valeur numérique obtenue après conversion,est placée à l'adresse (2F2 + R1)

DEC R1	Decrémenter le registre 1 qui contient le nombre de conversions désiré.
CI R1, 0	Comparer le contenu de R1 à 0
JNE LP2	En cas d'inégalité, sauter à LP2 pour lancer une nouvelle conversion.
END	Sinon, on arrête la conversion du signal.

Ce programme a une durée totale de :

$$22 + 16 + 12 + 22 + 16 + 20 + 12 = 120 \text{ cycles.}$$

Puisqu'un cycle dure 0,5 MS, le temps d'exécution du programme sera :

$$T = 120 \times 0,5 = 60 \mu\text{S}$$

Ainsi un temps minimal de 60 μS séparera le transfert en mémoire de 2 octets correspondants à 2 échantillons consécutifs du signal analogique. Or, si cela ne pose pas de problème en basses fréquences, il n'en est pas de même en hautes fréquences, où il n'est pas possible d'acquérir 8 éch./période.

Par exemple pour une fréquence de 5 khz, le temps séparant 2 échantillons successifs du signal est de 25 μS ce qui montre que dans ce cas la durée du programme est trop longue pour espérer acquérir tous les échantillons.

Par contre, avec la DMA les temps de transfert n'excèdent pas 5 μS par octet (comme on le verra dans le chapitre suivant).

Conclusion : Nous allons donc utiliser la DMA, car c'est le seul mode de transfert qui soit compatible avec la rapidité de notre carte d'acquisition.

- C H A P I T R E III -

L'ACCES DIRECT A LA MEMOIRE
ETUDE DU CONTROLLEUR DMA MC 6844

3-1 : L'ACCES DIRECT A LA MEMOIRE (DMA)

C'est une technique d'entrées/sorties qui permet à un périphérique d'avoir accès directement à la mémoire, sans passer par le processeur. Il est alors possible d'effectuer des échanges de données très rapides.

La DMA nécessite l'utilisation d'un circuit spécialisé qu'on appelle " contrôleur DMA ". Ce circuit va se charger du transfert des données, en prenant le contrôle du bus adresse, du bus de données et des signaux de contrôle de la mémoire.

Avant d'effectuer un transfert de données par DMA, il faut fournir au contrôleur un minimum d'informations, notamment :

- l'adresse mémoire de début du transfert
- le nombre de mots ou d'octets à transférer
- le sens du transfert.

Ainsi les données vont être transmises sur le bus de données, pendant que le bus adresse véhicule l'adresse mémoire vers laquelle (ou à partir de laquelle) s'effectue le transfert par DMA. Si l'on transmet plus d'un mot de données, ceux-ci seront situés à des emplacements mémoire consécutifs.

Pour que le contrôleur puisse utiliser les bus du système, il faut que le processeur ne les utilise pas. Il existe différentes techniques pour parvenir à ce résultat. La plus simple, c'est celle qui consiste à mettre à l'arrêt le MPU pendant toute la durée du transfert.

Ainsi lorsqu'un périphérique sollicite la mémoire pour un transfert de données, il envoie au contrôleur DMA une demande de transfert (fig. 3-1); ce dernier répercute la demande au MPU par son signal HOLD (demande de DMA). En réponse à cette demande le processeur envoie au contrôleur un signal HOLDA (acquittement du signal HOLD) l'information ainsi que le bus adresse, le bus de données et les signaux de contrôle de la mémoire sont déconnectés par leur mise à l'état " haute impédance ", il peut donc les utiliser. Le circuit DMA envoie alors au périphérique un signal " Autorisation de transfert " puis prend le contrôle des bus et génère lui-même les signaux de contrôle de la mémoire.

A la fin du transfert DMA, le MPU reprend son fonctionnement normal.

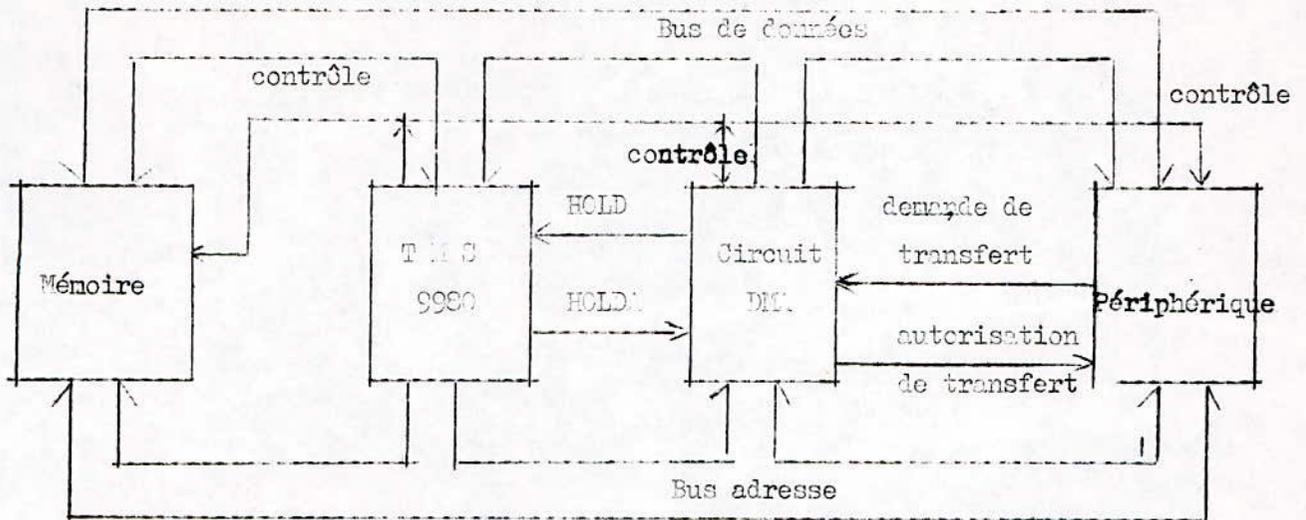


Fig. 3-1 Synoptique d'un échange DMA

3-2. ETUDE DU CONTROLLEUR DMA MC 6844

Il aurait été préférable d'utiliser comme contrôleur DMA, le TMS 9911 qui appartient à la famille 9900 de Texas Instruments; mais n'ayant pas eu à notre disposition ce circuit, nous avons opté pour l'utilisation du MC 6844 qui était disponible.

3-2-1. Fonctionnement du circuit :

(voir brochage en annexe)

Le MC 6844 est un contrôleur DMA que nous appellerons plus simplement DMAC (direct memory access controller).

Il appartient à la famille 6800, d'où la nécessité de l'étudier en détail pour résoudre les problèmes de sa compatibilité avec la carte TM 990/189.

Ce circuit comprend 4 canaux qui peuvent être configurés séparément par un transfert DMA.

A chaque canal est affecté :

- un registre adresse de 16 bits
- un registre nombre d'octets de 16 bits
- un registre de contrôle canal de 8 bits

En plus de ces 12 registres, le DMAC possède 3 autres registres de contrôle, communs aux 4 canaux :

- le registre de contrôle de priorité

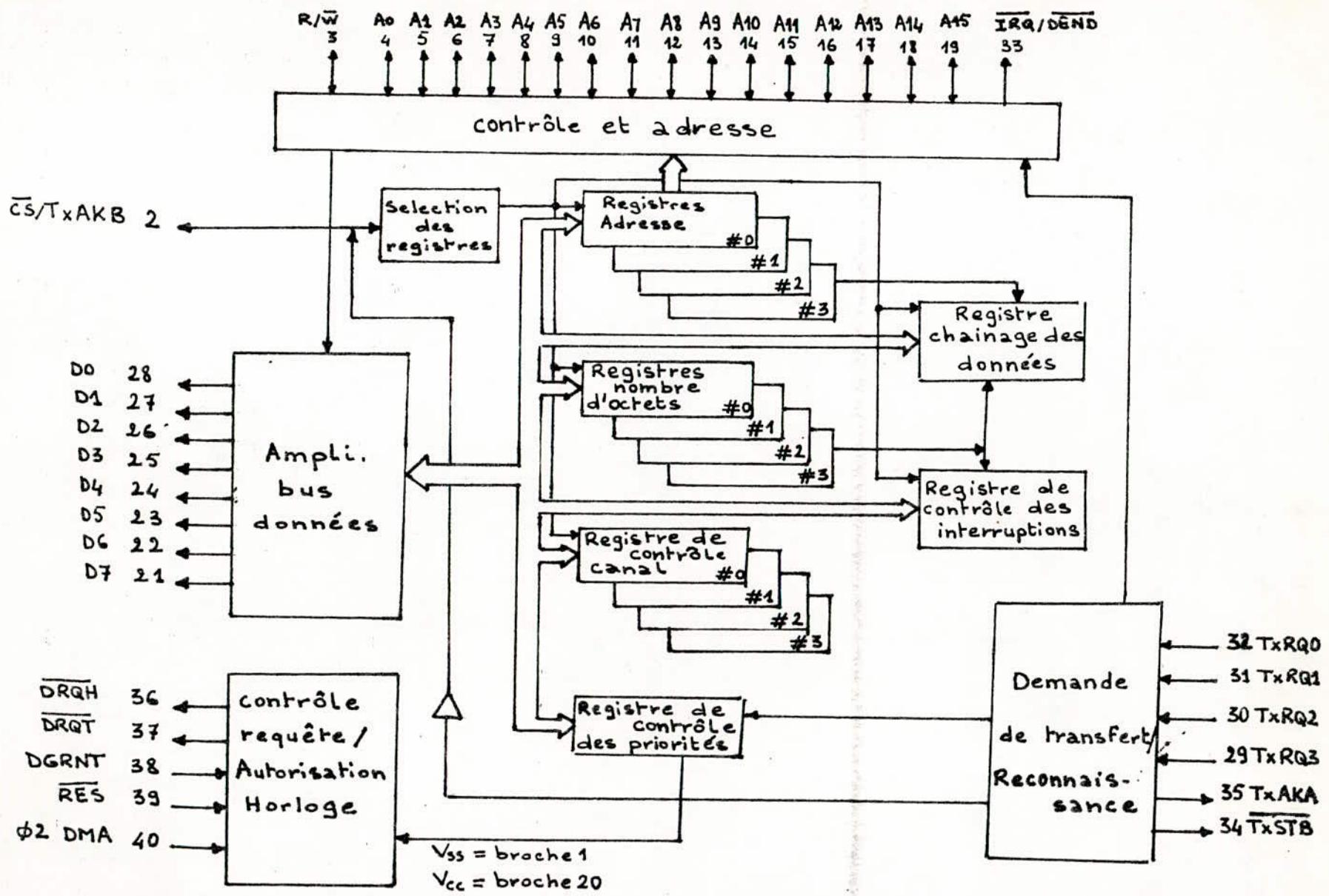


FIGURE 3-2 SCHEMA FONCTIONNEL DU MC6844

- le registre de contrôle d'interruptions
- le registre de chainage des données

Le DMAC possède 3 modes de transfert DMA :

- le mode Vol de cycle par TSC, qui consiste à allonger les horloges du microprocesseur pendant que le DMAC utilise la mémoire. Ce mode n'est pas réalisable avec le TMS 9900.
- le mode Vol de cycle par Halt qui consiste à arrêter le MPU pendant que le DMAC transfère un octet.
- le mode de transfert de bloc de données par Halt, où le DMAC peut transférer un bloc de données. Et pendant toute la durée du transfert le MPU est arrêté.

3-2-2. Les registres du DMAC

Le Registre adresse :

Il contient l'adresse mémoire de départ du transfert. Ce registre sera incrémenté ou décrétementé après chaque transfert d'octet, en fonction de l'état du bit 3 du registre contrôle canal.

Le Registre nombre d'octets :

Il contient le nombre d'octets à transférer. Il est décrétementé au début de chaque cycle DMA.

Le Registre de contrôle canal :

Il peut être considéré comme le registre d'état du canal. Ses bits 4 et 5 ne sont pas utilisés.

- * le bit 0 sert à déterminer la direction du transfert DMA. A l'état haut, il indique une lecture mémoire par le périphérique. A l'état bas, il indique une écriture en mémoire des données contenues dans le périphérique.
- * les bits 1 et 2 servent à sélectionner un mode de transfert parmi les 3 existants.

Bit 2	Bit 1	Mode de transfert DMA
0	0	Vol de cycle par Halt
0	1	Transfert de bloc (Halt)
1	0	vol de cycle par TSC
1	1	Interdit

- * le bit 3 contrôle le changement dans le registre adresse pour chaque cycle DMA.
à l'état Haut, le registre adresse est décrémenté
à l'état bas, le registre adresse est incrémenté.
- * le bit 6 est un indicateur occupé/libre. Ce bit d'état est mis à 1 lors d'un transfert et il passe à 0 après une fin de DMA (quand le signal $\overline{\text{IRQ/DEND}}$ passe à l'état bas)
- * le bit 7 est aussi un bit d'état, c'est un indicateur de fin de DMA.
Il est mis à 1 quand le bit 6 passe à 0 et il est mis à 0 quand le MPU lit le registre de contrôle canal.

Registre contrôle de priorité :

- * Les bits 0-3 servent à la validation des requêtes pour les canaux 0 à 3
A l'état Haut, la requête du canal respectif est validée
A l'état Bas, invalidation de la requête
- * Bits 4,5,6 : non utilisés
- * Bit 7 (contrôle de priorité)
A l'état Haut, la priorité est tournante le canal venant d'être servi prend la priorité la plus basse
A l'état Bas, priorité fixée : 0,1,2,3 .

Registre contrôle d'interruption :

- * Bits 0-3 (DIEO-3) servent à valider une éventuelle requête d'interruption par le signal DEND pour le canal correspondant.
A l'état Haut, la requête est validée
A l'état Bas, la requête est invalidée.
- * Les autres bits sont inutilisés.

Registre chainage de données :

Le bit 0 permet la validation ou non du chainage des données, ce qui se traduit par une lecture ou une écriture répétitive d'un bloc-mémoire. Pour cela les contenus des registres adresses et Nombre d'octets du canal ~~#~~ 3 sont transférés dans les registres respectifs du canal sélectionné par les bits 1 et 2 du registre et cela, après que le registre nombre d'octets ait été décrémenté jusqu'à 0.

- * bit 0 : à l'état haut, validation du chainage des données
à l'état bas invalidation
- * bits 1 et 2 : sélection de canal en chainage de données.

Bit 2	Bit 1	Canal \neq
0	0	0
0	1	1
1	0	2
1	1	interdit

- * Bit 3 Selection de 2/4 canaux
A l'état haut, mode 4 canaux
A l'état Bas, mode 2 canaux
- * Bits 4,5,6 inutilisés

3-2-3 . Les signaux du DMAC

A/ Signaux d'interface avec le MPU

* Le DMAC possède 8 lignes de données bidirectionnelles D0-D7 (D0 poids faible, D7 poids fort) qui seront reliées au bus de données du MPU. Les amplis de sortie du bus de données sont des circuits 3 états qui restent à l'état de haute impédance, sauf quand le MPU exécute des opérations de lecture du DMAC .

* Les lignes d'adresse A0-A15 (A0 poids faible, A15 poids fort) sont à l'état haute impédance en mode MPU, mais ces lignes deviennent des sorties en mode DMA. Elles sont ainsi positionnées par rapport au contenu du registre adresse du canal en cours d'exécution.

* Les lignes d'adresses A0 - A4, servent par contre en entrée et en sortie. En mode MPU, ce sont des entrées haute impédance qui servent à adresser les registres du DMAC.

* La ligne \overline{CS} / TXAKB (chip Select/Transfert Acknowledge B). Cette ligne multiplexée sert aussi bien en entrée qu'en sortie. En entrée, elle sert à adresser le DMAC quand elle est à l'état bas. \overline{CS} / TXAKB n'est une sortie en mode DMA que pour le mode 4 canaux où elle sert à sélectionner en compagnie d'une autre ligne (TXAKA) le canal utilisé.

* Ligne R/\overline{W} : (Read/Write ou lecture/Ecriture). Cette ligne bidirectionnelle sert à contrôler la direction du flot de données via l'interface d'entrée/sortie du DMAC.

En mode MPU, c'est une entrée haute impédance et un mode DMA, c'est une sortie.

En mode MPU quand elle est à l'état haut, et le boîtier sélectionné les amplificateurs de sortie de données du DMAC sont validés et un registre sélectionné est lu.

Quand R/\bar{W} est à l'état bas, les amplificateurs de sortie du DMAC sont désactivés et le MPU écrit dans un registre sélectionné.

En mode DMA, quand R/\bar{W} est à l'état haut, la mémoire est lue et les données contenues dans la mémoire sont écrites dans le périphérique. Quand elle est à l'état bas, le périphérique est lu et ses données sont mises en mémoire.

* $\overline{\text{RES}}$ ($\overline{\text{RES}}$) : L'entrée $\overline{\text{RES}}$ fournit la possibilité de remettre à 0 les registres du DMAC à l'exception des registres adresse et le nombre d'octets. Ceci invalide toutes les requêtes de transfert, masque toutes les interruptions, invalide le chaînage des données et place le registre de contrôle canal dans les conditions de lecture mémoire, en mode Vol de cycle par Halt.

* Demande de DMA en mode Vol de cycle par TSC : $\overline{\text{DRQT}}$ cette sortie active à l'état bas, requiert un transfert DMA pour un canal programmé en mode vol de cycle. Cette ligne connectée au circuit de contrôle d'horloge demande un allongement de l'horloge $\overline{\Phi}$ 1.

* Demande de DMA en mode vol de cycle par Halt (DMA request Halt Steal : DRQH). Cette sortie active à l'état bas requiert un transfert DMA pour un canal en mode vol de cycle par Halt ou en mode de transfert de bloc de données par Halt. Cette ligne est connectée à l'entrée Hold du MPU. Elle demeure à l'état bas jusqu'à ce que le dernier octet commence à être transféré.

* Autorisation de DMA (DMA Grant : DGRNT) : C'est une entrée haute impédance vers le DMAC, lui attribuant le contrôle des bus du système quand elle passe à l'état haut.

* $\overline{\Phi}$ 2 DMA : C'est une entrée haute impédance qui, reliée à l'horloge externe du MPU, permet d'effectuer les transferts en entrée ou en sortie des registres, l'échantillonnage des lignes de requête canal et la validation des autres signaux de contrôle du système

B/ Signaux d'interface avec les périphériques

* Les requêtes de transfert TXRQ 0-3, dont chacune est affectée au canal correspondant, sont des lignes d'entrée haute impédance qui sont reliées aux périphériques. Lorsqu'un périphérique demande un transfert, il place sa ligne TXRQ à l'état haut. Les lignes TXRQ sont échantillonnées en fonction de la priorité fixée dans le registre contrôle de priorité. En modes Vols de

cycle et pour le 1er octet du mode transfert par bloc les signaux TXRQ sont testés sur le front positif de l'horloge @ 2 DMA.

* Reconnaissance de transfert A (TXAKA) : C'est une sortie compatible TTL utilisée avec la ligne $\overline{CS}/TXAKB$ pour sélectionner le canal à échantillonner et cela en mode 4 canaux. En mode 2 canaux, elle sélectionne le canal ~~#~~ 0 ou le canal ~~#~~ 1.

En mode 1 Canal, elle n'est pas utilisée.

CS/TXAKB	TXAKA	Canal #
0	0	0
0	1	1
1	0	2
1	1	3

* Echantillonnage de transfert (\overline{TXSTB}) . C'est une sortie active à l'état bas qui provoque la reconnaissance donnée au périphérique et transfère les données de ou vers la mémoire.

En mode 4 canaux, TXSTB valide le décodage de TXAKA et $\overline{CS}/TXAKB$ pour sélectionner le périphérique reconnu.

* Requête d'interruption/Fin de DMA ($\overline{IRQ}/\overline{DBND}$) En mode DMA cette sortie passe à l'état bas pour le dernier octet de transfert, indiquant ainsi une fin de DMA. Elle peut être utilisée pour interrompre le MPU et pour signaler au périphérique que le transfert par bloc de données est terminé.

3-2-4. Préparation d'un transfert DMA :

Avant qu'un canal procède à un transfert DMA, le MPU doit programmer les registres adresse, nombre d'octets et contrôle canal correspondant au canal concerné par le transfert, ainsi que les registres contrôle de priorité, contrôle d'interruption et chaînage des données.

Les lignes A0 à A4 du DMAC permettent au MPU d'adresser tous les registres. Les différentes adresses sont données par les tableaux des figures 3-3 et 3-4.

	Registre	Adresse (Hex)
CANAL 0	Adresse (poids fort)	0
	Adresse (poids faible)	1
	Nombre d'octets (poids fort)	2
	Nombre d'octets (poids faible)	3
CANAL 1	Adresse (poids fort)	4
	Adresse (poids faible)	5
	Nombre d'octets (poids fort)	6
	Nombre d'octets (poids faible)	7
CANAL 2	Adresse (poids fort)	8
	Adresse (poids faible)	9
	Nombre d'octets (poids fort)	A
	Nombre d'octets (poids faible)	B
CANAL 3	Adresse (poids fort)	C
	Adresse (poids faible)	D
	Nombre d'octets (poids fort)	E
	Nombre d'octets (poids faible)	F

Fig. 3-3 . Registres adresse et Nombre d'octets

Regis-tre	Adresse (Hex)
Contrôle canal 0	10
Contrôle canal 1	11
Contrôle canal 2	12
Contrôle canal 3	13
Contrôle de priorité	14
Contrôle d'interruption	15
Chainage des données	16

Fig. 3-4. Registres de contrôle

3-2- 5 . Mode de transfert DMA approprié

Pour notre acquisition nous pouvons utiliser 2 modes de transfert DMA : le mode vol de cycle par Halt et le mode transfert de bloc de données par Halt. (Le mode vol de cycle par TSC étant impossible à utiliser avec la carte TEXAS).

En utilisant les 4 canaux du DMAC pour les 4 CAN, chaque canal va transférer un octet en mémoire à tour de rôle. Le mode de transfert de bloc qui a la vitesse de transfert la plus rapide (1 S/ octet) serait le plus intéressant si les données étaient disponibles toutes les 1 S. Ce qui n'est pas le cas ici, alors que pour le mode vol de cycle par Halt permet le transfert d'un octet par canal à tour de rôle.

C'est donc le mode de transfert le mieux adapté.

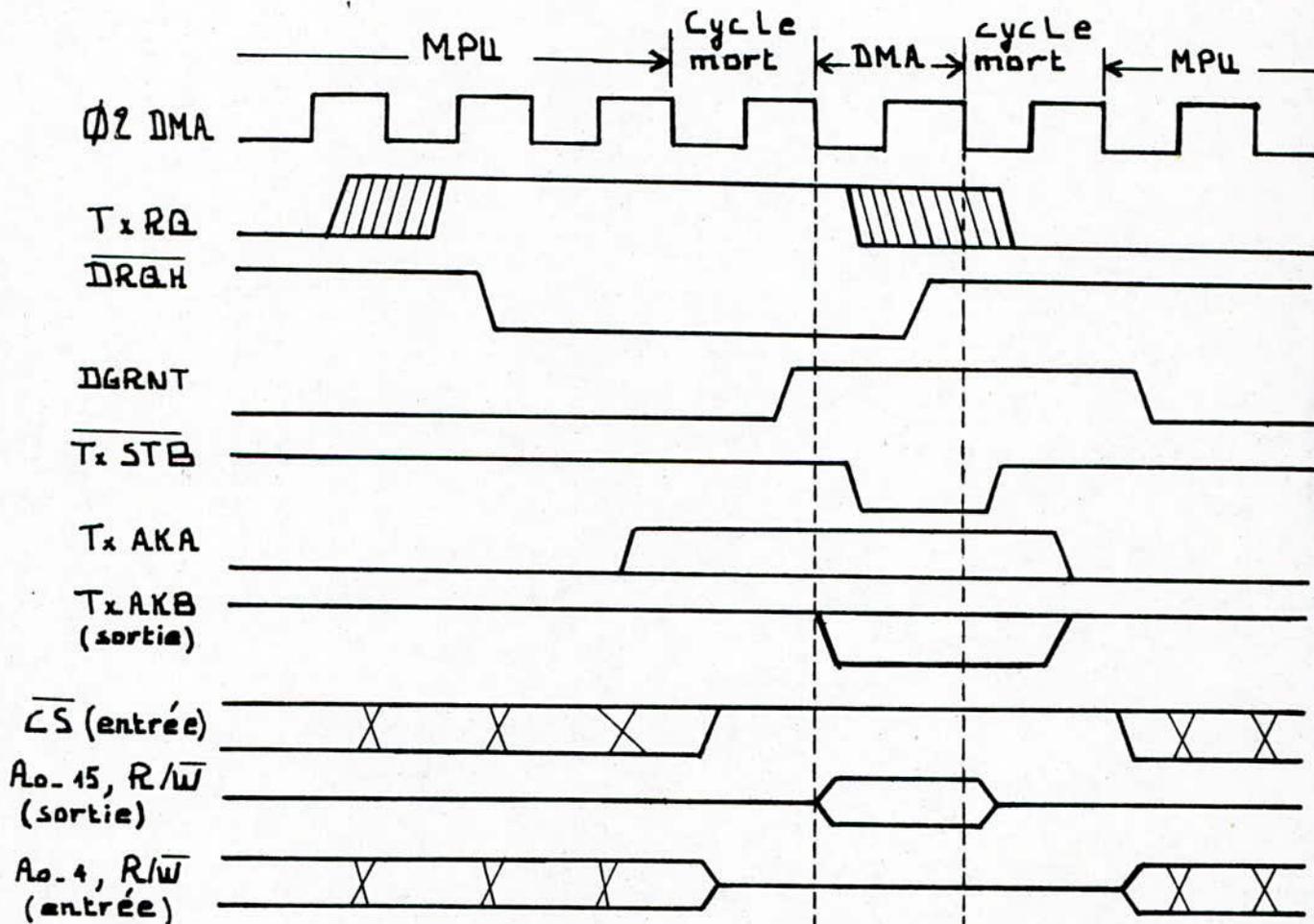


Fig. 3-5 . Mode vol de cycle par Halt

MISE EN OEUVRE DE LA CARTE D'ACQUISITION

INTRODUCTION :

Après avoir expliqué dans les chapitres précédents la conception de la carte d'acquisition et le fonctionnement de ses principaux composants, nous allons aborder dans ce chapitre sa mise en oeuvre.

4-1. Utilisation du DMAC dans la carte d'acquisition

Le DMAC va permettre de transférer en mémoire de la carte TM 990/189 les données numériques provenant des 4 CAN.

Mais lors d'un transfert DMA, les buffers d'extension de bus sont à l'état " haute impédance ", ce qui empêche les données d'être transférées vers la mémoire sur carte. Pour éviter ce problème, on a relié la carte TM 990/189 à l'extension mémoire SIHQ 092 (Voir chapitre 2).

Ainsi, le DMAC transférera les données dans cette extension mémoire qui occupe les adresses comprises entre 1000 et 1 FFT.

A chaque fois qu'un CAN a terminé de convertir un échantillon analogique du signal, il doit solliciter le DMAC pour transférer en mémoire l'octet numérique.

L'utilisation des 4 canaux du DMAC dont chacun sera programmé en mode vol de cycle par Halt, permet de prendre en compte les demandes de transfert DMA émanant à tour de rôle de chaque CAN, puis d'assurer le transfert en mémoire de l'octet converti.

4-2 . Adaptation du DMAC à la carte TM 990/189

L'utilisation du MC 6844 comme contrôleur DMA va poser le problème de la compatibilité de ses signaux avec ceux de la carte TM 990/189. En effet, certains signaux du DMAC vont nécessiter l'utilisation d'une logique destinée à les rendre compatibles avec les signaux de la carte TEXAS.

Ainsi, on aura les connexions suivantes :

- le DRQH sera relié au HOLD
- le DGRNT sera relié au HOLDA
- le RESET sera relié au RESET de la carte Texas.
- le bus de données (DO - D7) sera relié au bus de données (DO - D7)

de la carte Texas, mais dans l'ordre inverse car si pour le DMAC le MSB est D7, pour la carte Texas le MSB est le D0.

- le bus adresse (A0 - A15) comprend 2 lignes de plus que le bus adresse (A0 - A13) de la carte Texas. On reliera donc les lignes (A0 - A13) du DMAC à celles de la carte Texas mais dans l'ordre inverse, car A0 est un LSB pour le DMAC, alors qu'il représente le MSB du bus adresse de la carte TM 990/189.

En ce qui concerne les autres signaux du DMAC, nous allons étudier leur état en mode MPU et en mode DMA afin de déterminer comment les connecter aux signaux de la carte Texas.

Mode MPU

- Le $\overline{CS}/TXAKB$ sera une entrée de sélection du DMAC. Elle sera reliée à la sortie d'un décodeur.
- La ligne R/\overline{W} sera une entrée que le MPU positionne à 1 ou à 0 selon qu'il veut lire ou écrire dans un registre du DMAC.
- L'horloge ϕ_2 devra être reliée à l'horloge du MPU.
- Le RESET peut être relié au RESET de la carte pour initialiser les registres du DMAC.

Mode DMA

- $\overline{CS}/TXAKB$ et TXAKA sont des sorties qui seront utilisées comme entrées d'un décodeur pour donner une reconnaissance de transfert, au périphérique du canal concerné par l'échange DMA.
- TXSTB valide le décodage de TXAKA et $\overline{CS}/TAKB$
- La ligne R/\overline{W} est une sortie qui devra remplacer les signaux de validation de lecture et d'écriture en mémoire.

Ainsi en mode MPU, le DBIN peut donner le R/\overline{W} . En mode DMA R/\overline{W} peut donner le \overline{WE} et le DBIN.

Le TXSTB servira de signal \overline{MEMEN} en mode DMA.

Comme le système doit pouvoir fonctionner en mode MPU et en mode DMA, ces connexions risquent de créer des conflits d'accès aux lignes. Pour résoudre ce problème, nous utiliserons un buffer à 3 états le SN 74 LS 244 qui va permettre de réaliser les connexions désirées aussi bien en mode MPU qu'en mode DMA.

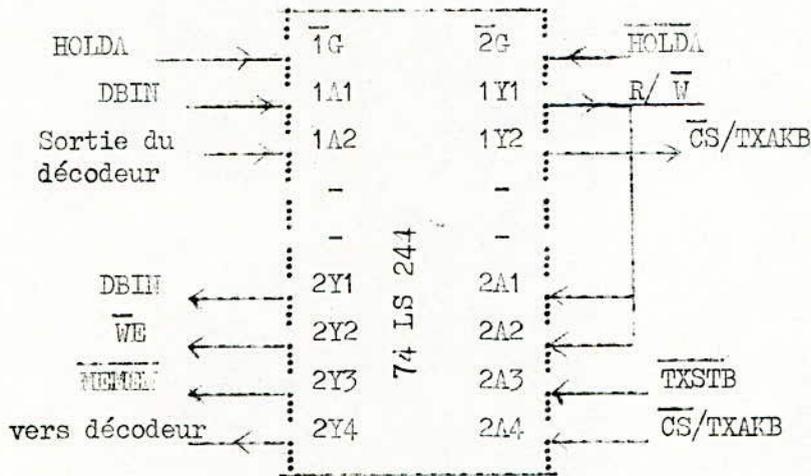


Fig. 4-1

On applique \overline{HOLDA} à l'entrée $2\overline{G}$ et $HOLDA$ à l'entrée $1\overline{G}$ du 74 LS 244.

* Ainsi en mode MPU, \overline{HOLDA} est à 1 et $HOLDA$ est à 0, donc l'entrée $2\overline{G}$ est inhibée. Le signal $DBIN$ appliqué à l'entrée $1A1$ sera transmis à la sortie $1Y1$ et servira de ligne R/\overline{W} . La sortie d'un décodeur appliquée en $1A2$ est transmise en $1Y2$ pour être reliée à l'entrée $\overline{CS}/TXAKB$.

* En mode DMA, $HOLDA$ est à 1 et \overline{HOLDA} est à 0, donc cette fois c'est $1\overline{G}$ qui est inhibée.

Les signaux appliqués aux entrées $2A1, 2A2, 2A3, 2A4$ seront transmis respectivement aux sorties $2Y1, 2Y2, 2Y3, et 2Y4$.

Ainsi R/\overline{W} donnera le \overline{WE} et le $DBIN$.

- \overline{TXSTB} donnera le \overline{MEMEN}

- le $\overline{CS}/TXAKB$ ira vers l'entrée d'un décodeur (74LS139)

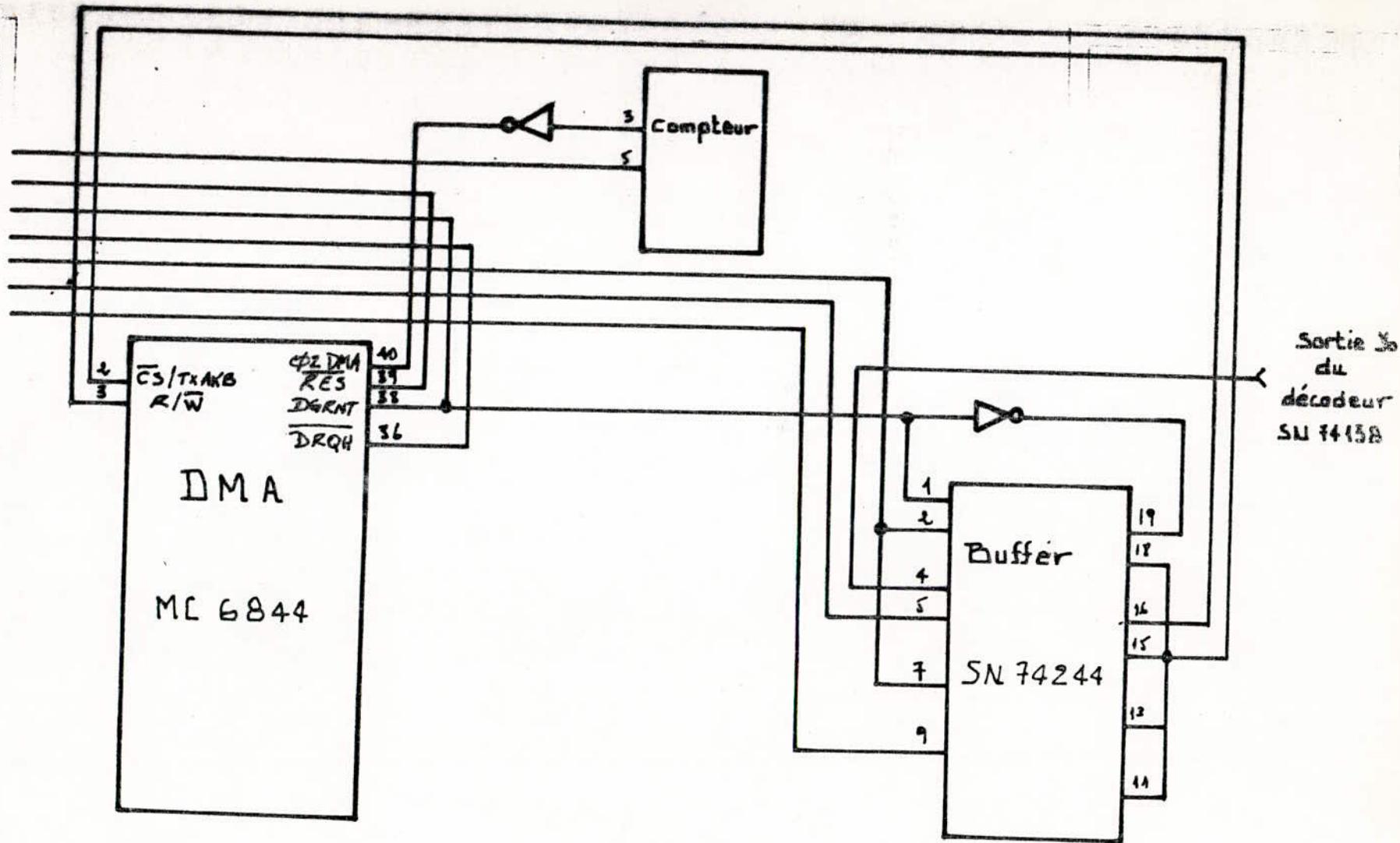
L'horloge ϕ_2 DMA fonctionne à la fréquence de 1 Mhz.

Elle doit être reliée à l'horloge du MPU $\overline{Q3}$ qui elle, est à une fréquence de 2 Mhz. Pour que ϕ_2 fonctionne en synchronisation avec $\overline{Q3}$, on inverse le signal d'horloge $\overline{Q3}$ du MPU, puis on divise par 2 sa fréquence à l'aide d'un compteur. On obtient ainsi un signal d'horloge de 1 Mhz qu'on relie à ϕ_2 DMA. (Voir fig. 4-2)

4-3 . Connexion du DMAC avec les convertisseurs

Les 4 lignes $TXRQ$ du DMAC doivent être reliées aux convertisseurs pour permettre à chacun d'eux de déclencher la demande de

$\overline{\text{CS}}$
RES
HOLDA
HOLD
D $\overline{\text{OIN}}$
MEMEN
WE



Connections entre DMA et La carte TM 990/189

fig 4.2

transfert. On reliera chaque TXRQ à l'INTR inversé d'un convertisseur. Ainsi, lorsqu'un convertisseur a fini de convertir un échantillon analogique, l'INTR passe à 0, donc TXRQ passe à 1 entraînant ainsi une demande de transfert DMA.

Lorsque le transfert est terminé, le DMAC doit placer la ligne \overline{RD} du convertisseur à l'état bas.

$\overline{CS}/TXAKB$ et TXAKA qui sont des sorties en mode DMA, serviront au décodage du canal concerné par le transfert. Ainsi grâce à l'utilisation d'un décodeur, le 74LS139, les sorties seront reliées au \overline{RD} des convertisseurs. (Fig. 4-3)

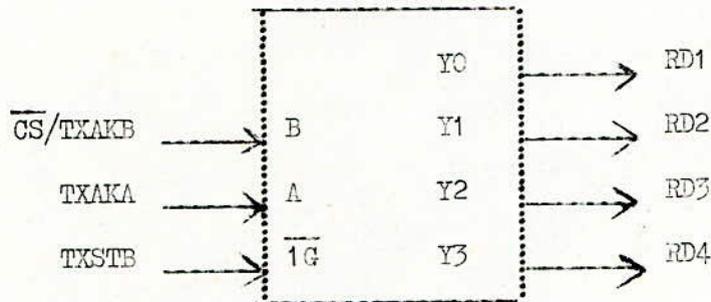


Fig. 4-3

Table de vérité du 74 LS 139

$\overline{1G}$	B	A	1Y0	1Y1	1Y2	1Y3
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0
1	X	X	1	1	1	1

4-4 . Sélection du DMAC

Puisque l'extension mémoire SIHQ 092 utilise les adresses comprises entre 1000 et 1FFF, il nous reste les adresses allant de 2000 à 2FFF pour sélectionner le DMAC. Comme les lignes A0 à A4 du DMAC (A9 à A13 du TMS 9980) permettent d'affecter aux registres du DMAC, les adresses allant de 0 à 16, nous allons réserver au DMAC les adresses comprises entre 2F00 et 2F16.

Les lignes A0 à A8 auront donc la configuration suivante :

A0	A1	A2	A3	A4	A5	A6	A7	A8
1	0	1	1	1	1	0	0	0

Pour obtenir les adresses de sélection, on met aux entrées d'une porte NAND les lignes A0, $\bar{A1}$, A2, A3, A4, A5 accompagnées du MEMEN. Puis la sortie de la porte est inversée puis connectée à l'entrée G1 d'un décodeur le 74 LS 138 qui aura 3 entrées de sélection CBA qui seront A6, A7 et A8

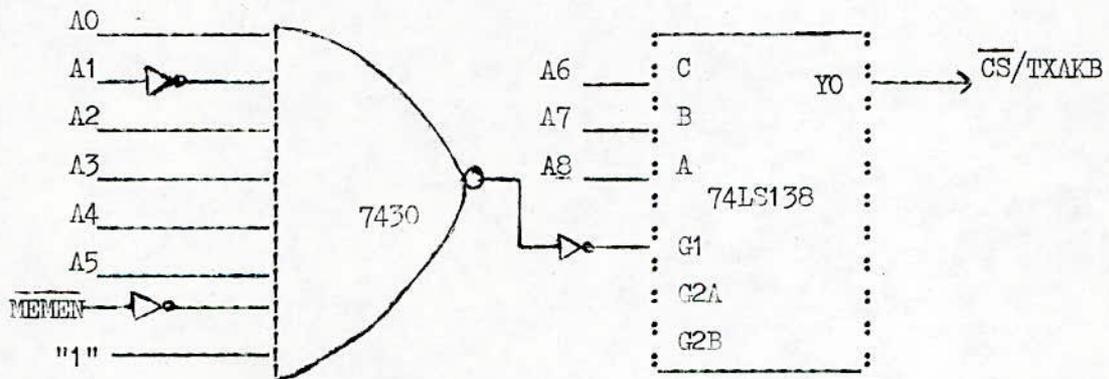


Table de vérité du 74 LS 138

INPUTS					OUTPUTS							
Enable		Select										
G1	G2	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	1	X	X	X	1	1	1	1	1	1	1	1
0	X	X	X	X	1	1	1	1	1	1	1	1
1	0	0	0	0	0	1	1	1	1	1	1	1
1	0	0	0	1	1	0	1	1	1	1	1	1
1	0	0	1	0	1	1	0	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1	1	1
1	0	1	0	0	1	1	1	1	0	1	1	1
1	0	1	0	1	1	1	1	1	1	0	1	1
1	0	1	1	0	1	1	1	1	1	1	0	1
1	0	1	1	1	1	1	1	1	1	1	1	0

Pour les adresses 2F00 à 2F16, A6 A7 A8 = 000 , donc la sortie Y0 est active. Cette sortie sera reliée à $\overline{CS}/TXAKB$ (après être passée par le 74244).

4- . Programmation du DMAC

Nous allons configurer le DMAC pour qu'il fonctionne en mode 4 canaux avec le mode de transfert vol de cycle Halt, pour chaque canal.

Les registres adresses des canaux 0,1,2,3 seront chargés respectivement avec les adresses : 1200,1400,1600,1800. Ces adresses seront ensuite décrémentées. Le nombre d'octets total à transférer va correspondre par exemple à 10 périodes. Donc à raison de 8 éch./période, cela fait un total de $8 \times 10 = 80$ octets. Donc 20 octets par canal.

Enfin chaque canal est programmé pour l'écriture en mémoire des données du convertisseur qui lui est associé.

Ce qui donne par exemple pour le canal 0 .

Registre	Adresse (Hex)	Contenu
Adresse (16 bits)	2F00	1200
Nombre d'octets (16 bits)	2F02	20
Contrôle canal (8 bits)	2F10	08
Contrôle de priorité (8 bits)	2F14	8F
Contrôle d'interruption (8 bits)	2F15	00
Chainage des données (8 bits)	2F16	08

Le programme qui permet de charger tous les registres avec le contenu correspondant est le suivant :

```

A 200
DATA >1200,20, >1400,20, >1600,20, >1800,20, >0808, >0808, >8F00, >0800
L W P I >300
CLR R1
LP MOVE @>200 ( R1 ), @>2F00 ( R1 )
INC R1
CI R1, >17
JNE LP
END
    
```

4-6. Utilisation des DAC 830 dans la carte

4-6-1. Génération des tensions continues variant entre -2,5 v et + 2,5 v

Les 2 convertisseurs N/A DAC 830 vont générer à partir de la carte TM 990/189 deux tensions continues qui vont être utilisées par la carte analogique pour faire varier la fréquence du signal. Ces 2 tensions

continues seront variables entre - 2,5 v et + 2,5 v.

Nous avons vu au paragraphe (1-6) que la tension analogique obtenue après la conversion N/A avait pour expression :

$$V_{out} = V_{ref} \frac{N}{2^8}$$

Pour $V_{ref} = 5$ v, cette tension est négative et peut varier de 0 à - 5v. Pour obtenir une tension variable de - 2,5v à + 2,5v ,on utilisera un montage comprenant un ampli opérationnel utilisé en sommateur fig. (4-4).

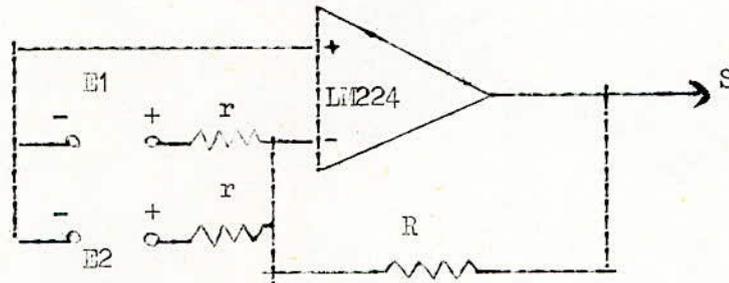


Fig. 4-4

$$S = - \frac{R}{r} (E1 + E2)$$

E1 étant la tension V_{out} on peut déterminer E2 pour que S varie entre - 2,5v et + 2,5v.

Si l'on prend $R = r$,on aura : $S = - (E1 + E2)$

Donc en prenant $E2 = + 2,5v$, on peut voir que :

pour $E1 = - 5v \longrightarrow S = - (-5 + 2,5) = + 2,5$ v

pour $E1 = 0v \longrightarrow S = - (0 + 2,5) = - 2,5$ v

4-6-2. Sélection des CNA

Les 2 DAC 830 seront sélectionnés de la façon suivante :

- a) Pour le 1er DAC 830, la sortie Y2 du décodeur 74 LS 138 est inversée puis mise à l'entrée d'une porte NAND (7430) en compagnie des lignes A9, A10, A11, A12, A13 (fig. 4-5).

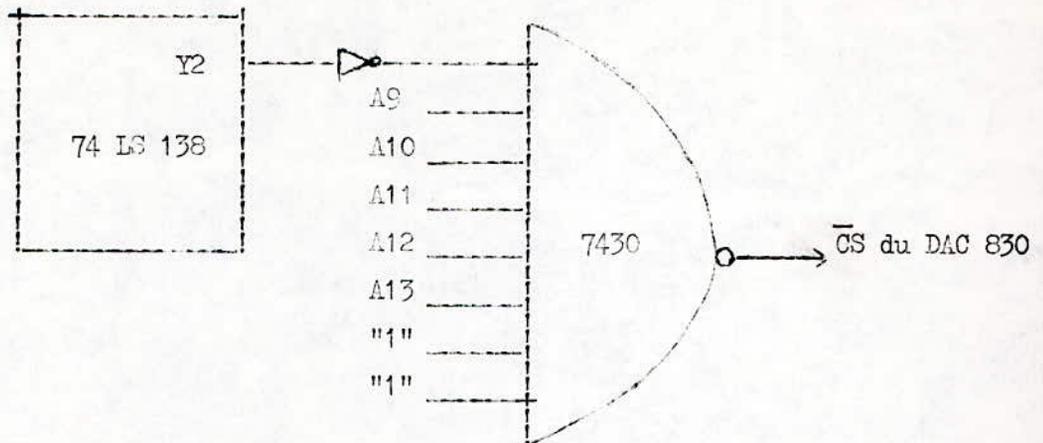


Figure 4-5

Ainsi les lignes A0 à A13 auront la configuration suivante

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

Donc ce DAC aura l'adresse 2F5F.

- b) Pour le 2° DAC 830, on procède de la même façon sauf qu'au lieu de Y2 c'est la sortie Y3 du décodeur qui est inversé puis reliée à une entrée d'une autre porte NAND accompagnée des mêmes lignes adresse que précédemment.

Ainsi ce 2ème DAC 830 aura l'adresse 2F7F.

4-6-3 Modes de fonctionnement des CNA

Sur notre carte d'acquisition nous avons prévu de faire fonctionner les 2 DAC 830 selon deux modes :

- le 1er mode, pendant l'acquisition des données où les DAC 830 généreront les deux tensions continues nécessaires à la carte analogique.
- le 2° mode, après l'acquisition, où les DAC seront utilisés pour restituer le signal analogique à partir des données stockées en mémoire pour reproduire son allure sur table traçante analogique.

Pour passer d'un mode à l'autre, on utilise 4 commutateurs (figure 4-6)

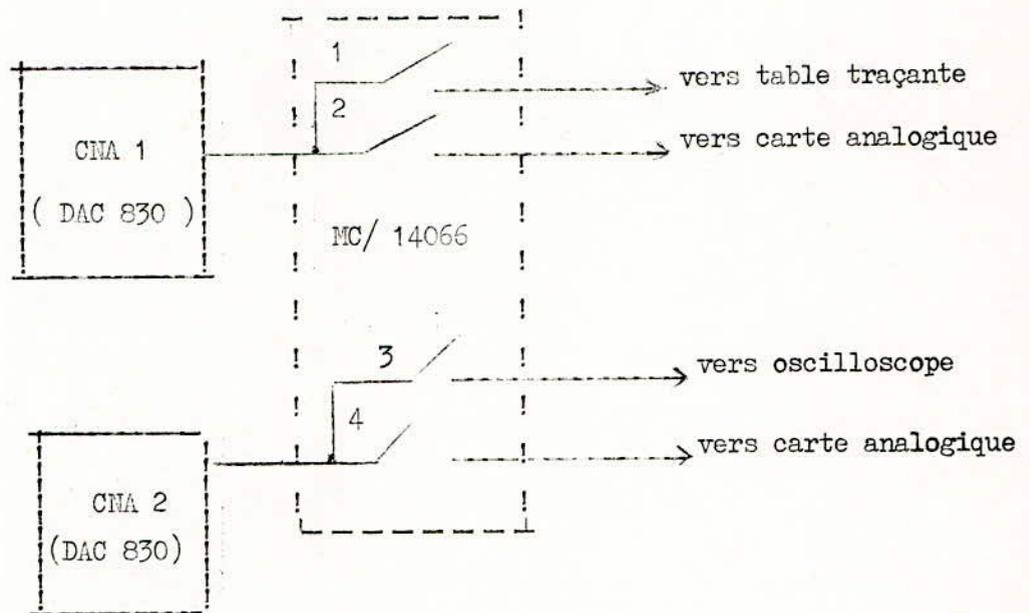


Figure 4-6

Un circuit logique permet de regrouper ces 4 commutateurs : le MC 14066.

Ces commutateurs seront commandés par l'intermédiaire d'1 bit du CRU et de son inverse.

- les commutateurs 1 et 3 seront commandés par le bit 1

- " " 2 et 4 seront commandés par l'inverse du bit 1

* Ainsi lorsque le bit 1 du CRU est à 1 les commutateurs 1 et 3 sont fermés alors que les commutateurs 2 et 4 sont ouverts, donc les DAC envoient la tension analogique sur table traçante et vers l'oscilloscope

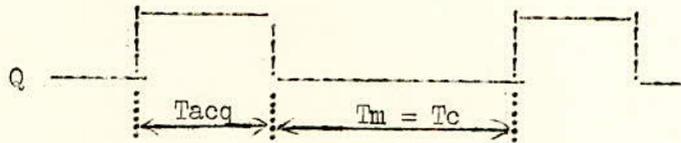
* Lorsque le bit 1 est à 0 les commutateurs 1 et 3 sont ouverts et pendant que les commutateurs 2 et 4 sont fermés. Donc cette fois, les DAC envoient les 2 tensions continues vers la carte analogique.

4-7 . Fonctionnement du registre à décalage

4-7-1. Essais pratiques :

Les essais effectués ont montré qu'en utilisant les sorties QA, QB, QC et QD du registre à décalage comme signaux de commande, les convertisseurs n'arrivent à convertir que jusqu'à une fréquence de 4,5 khz. Au delà de cette fréquence le temps où Q est à l'état bas

(qui correspond au temps de maintien et au temps de conversion) est suffisant. Aussi on a pensé augmenter ce temps en réduisant le temps d'acquisition (Tacq).



Pour celà,chaque sortie Q est appliqué en entrée d'une porte ET accompagnée du signal carré 8 w (voir fig. 4-7)ce qui a pour effet de réduire Tacq de moitié et donc d'augmenter ainsi Tc.

Ainsi pour $f = 5 \text{ khz}$ Tacq passe de $25 \mu\text{S}$ à $12,5 \mu\text{S}$. Ce qui est suffisant car le LF 398 a un temps d'acquisition d'environ $10 \mu\text{S}$.

Avec cette structure on peut convertir (à raison de 8 éch./période) un signal qui peut aller même jusqu'à $6,5 \text{ khz}$.

4-7-2. Fonctionnement du registre en 2 modes

Nous avons vu dans le paragraphe (1-5),que le registre à décalage avait deux modes de fonctionnement.

1er Mode :

L'acquisition de 8 éch./période pour les fréquences du signal comprises entre $0,01 \text{ hz}$ et 5 khz . Le signal d'horloge sera le 8 w et l'entrée A du registre sera générée à partir d'un ET logique entre les signaux 4 w et 2 w.

2ème Mode :

L'acquisition de 4 éch./période pour les fréquences du signal allant de 5 khz à 10 khz . Le signal d'horloge sera cette fois le 4 w et le signal A sera généré à partir d'un ET logique entre les signaux w et 2 w.

Pour passer d'un mode à l'autre on utilisera aussi des circuits MC 14066 qui comportent des commutateurs.

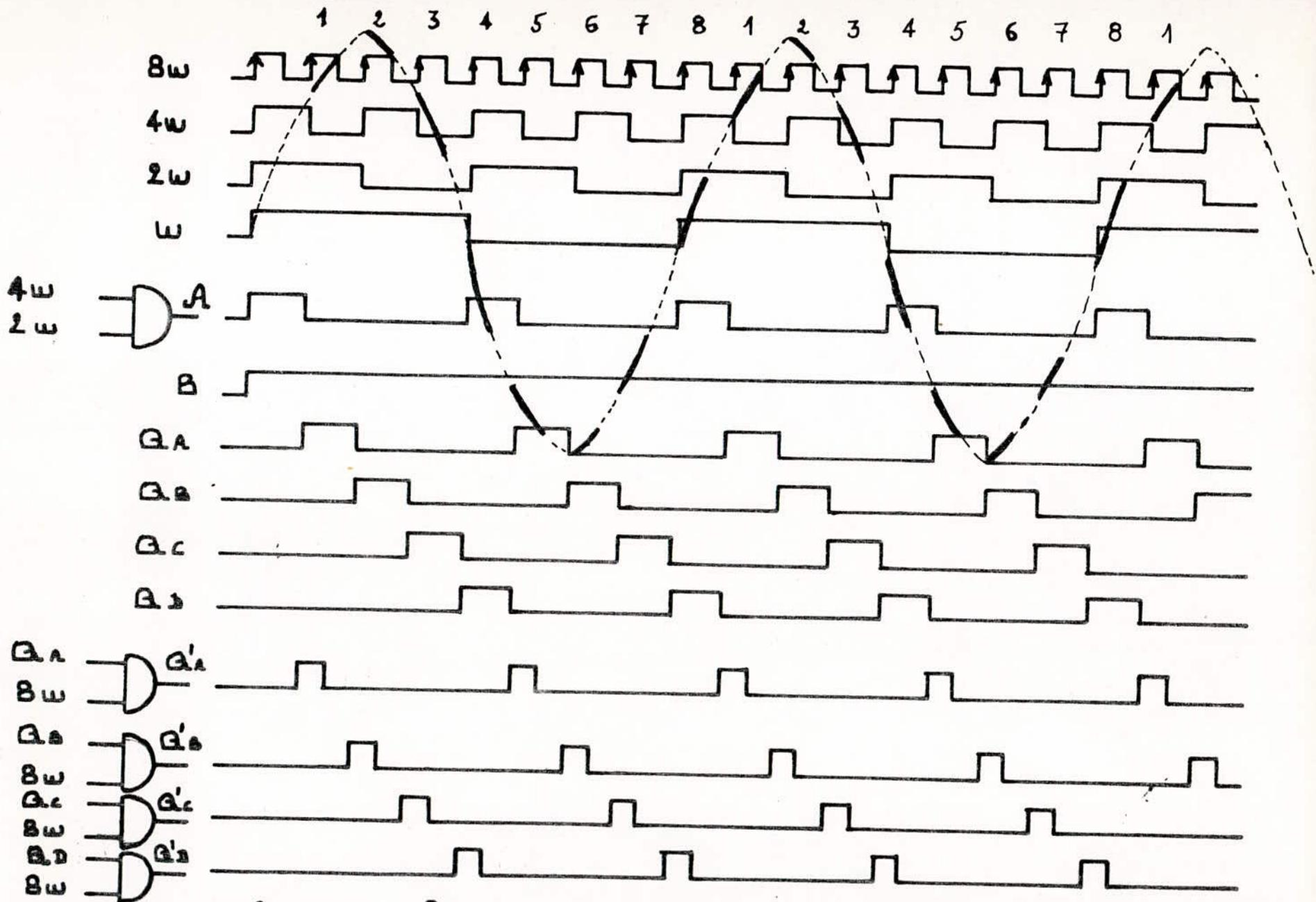
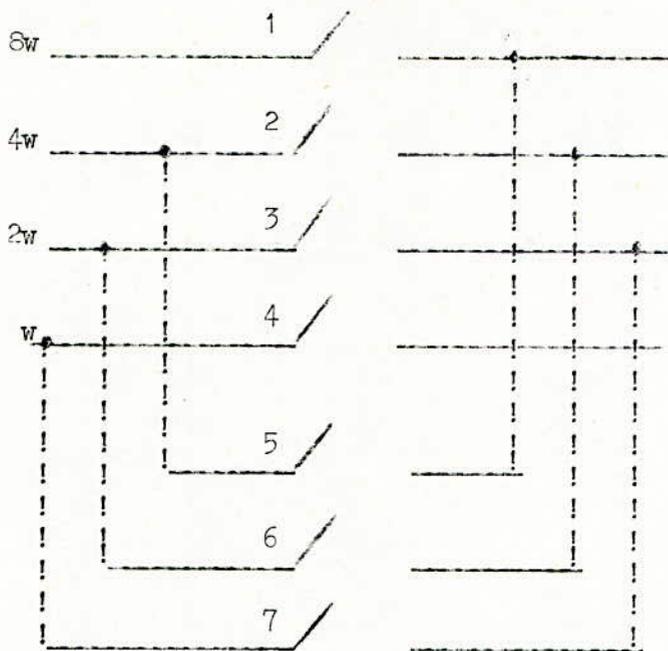


fig. 4-7. Signaux de commande pour l'acquisition de 8 éch./période.



Les commutateurs 1, 2,3 et 4 seront commandés par un même bit du CRU (bit 2) et les commutateurs 5,6,7 par ce bit inversé.

Ainsi pour le 1er mode les commutateurs 1,2,3,4 sont fermés (le bit 2 du CRU est à 1 et son inverse à 0) alors que les commutateurs 5,6,7 sont ouverts.

Pour le 2ème mode c'est l'inverse qu'on obtient en mettant le bit 2 du CRU à 0 (son inverse est alors à 1).

4-8 . Utilisation d'un CAN en basses fréquences

4-8-1 Principe :

En basses fréquences, il n'est pas nécessaire d'utiliser les 4 CAN; un seul suffit pour l'acquisition du signal analogique.

Si le CAN convertit 8 éch./période on peut calculer la fréquence maximale du signal au delà de laquelle le CAN n'a plus le temps de convertir les 8 éch./Période.

$$f \leq \frac{1}{8 \times 10^{-4}} = 1,25 \text{ khz}$$

Donc le CAN peut travailler ainsi jusqu'à la fréquence maximale de 1,25 khz.

Pour utiliser un seul convertisseur, il faut donc bloquer le fonctionnement des 3 autres. Pour cela on va procéder de la manière suivante :

La sortie Y1 du décodeur 74 LS 138 et la sortie QA inversée du registre à décalage sont appliquées aux entrées d'une porte " ET ". La sortie de cette porte est appliquée au \overline{WR} du 1er CAN.

Puis en mettant le CLEAR du registre à décalage à 0, toutes les sorties sont à 0, donc les \overline{WR} des CAN sont à 1. On voit ainsi que les trois autres CAN ne pourront pas convertir. Par contre, pour le premier CAN, la commande de conversion (\overline{WR}) pourra être activée par l'intermédiaire de la sortie Y1 du décodeur; donc lorsqu'on envoie son adresse (fig. 4-8)

Le CLEAR sera commandé à l'aide d'un bit du CRU.

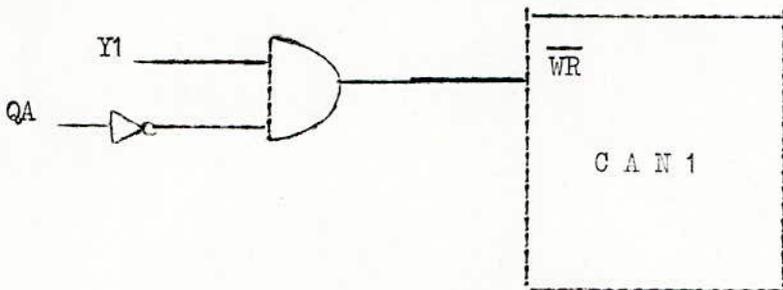


Fig. 4-8

4-8-2 . Adresse du CAN

La sortie Y1 est activée quand les entrées de sélection du CBA décodeur ont les états 001. Ces entrées de sélection étant les lignes A6, A7, A8, le convertisseur occupera les adresses allant de 2 F20 à 2F3F.

4-8-3 . Utilisation du CAN pour plusieurs signaux d'entrée

Sur notre carte on a prévu d'utiliser un des 4 CAN, en basses fréquences aussi bien pour notre signal analogique que pour 2 autres signaux: par exemple la température et le niveau.

Pour réaliser ce multiplexage des 3 signaux, on se servira de trois commutateurs (MC 14066).

En utilisant un décodeur 74 LS 139, on reliera 3 de ses sorties aux entrées de commande des commutateurs.

Les entrées A et B du décodeur seront reliées chacune à 1 bit du CRU.

Ainsi en fonction des états des 2 bits du CRU, une sortie du décodeur sera active et le signal choisi sera envoyé vers le CAN. D'après la table de vérité du 74 LS 139 on peut déduire le tableau suivant :

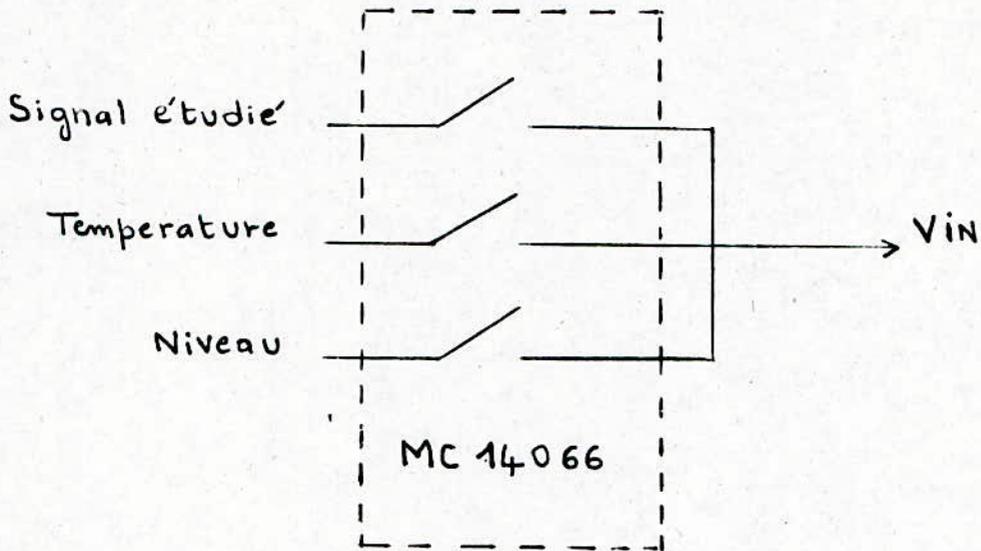


Fig 4-9 Multiplexage de 3 signaux

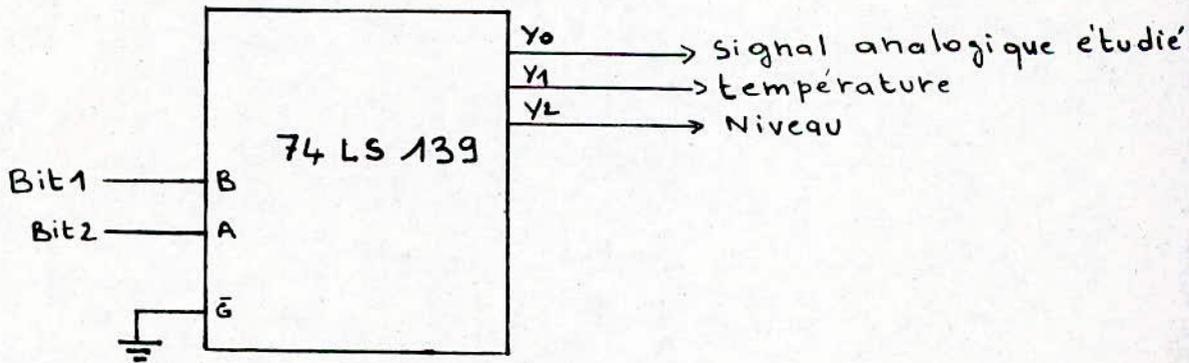


Fig 4-10 Decodage du signal d'entrée

bit 1	bit 2	Sortie active	Signal à convertir
0	0	Y0	Signal analogique étudié
0	1	Y1	Signal de température
1	0	Y2	Signal de niveau

Remarque : Pour la réalisation de notre carte d'acquisition, nous n'avons fait que prévoir la possibilité de multiplexage entre 3 signaux analogiques. L'acquisition correspondant à 2 signaux, autres que celui étudié ne fait pas partie de notre travail.

4-9 . Fonctionnement général de la carte d'acquisition

* Pour les hautes fréquences

Le CLEAR qui est relié à un bit du CRU commande le lancement de l'acquisition.

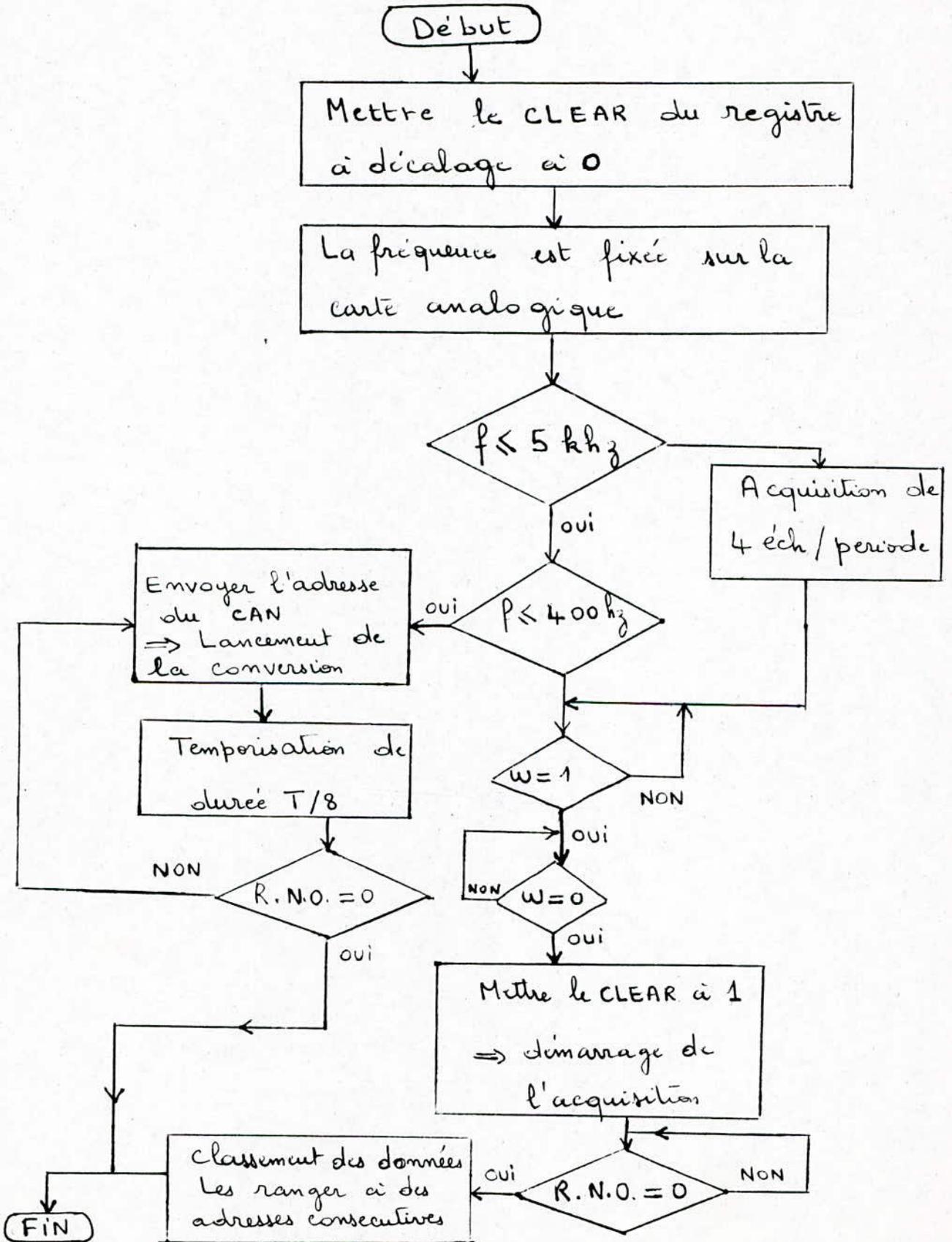
On le positionne d'abord à 0, puis on programme les registres du DMAC, ensuite on teste le w.

On démarre l'acquisition quand $w = 0$ c'est à dire que les premières données transférées correspondront à des échantillons négatifs. On aura ainsi les 4 premiers octets négatifs puis les 4 suivants seront positifs et ainsi de suite. A la fin de l'acquisition, on classe les échantillons dans l'ordre chronologique à des adresses consécutives.

* Pour les basses fréquences

Le CAN utilisé fera l'acquisition des 8 éch./période et sera donc sélectionné à des intervalles de temps $T/8 = 100 \mu S$ ($T =$ période du signal analogique) . Pour générer cette temporisation de $100 \mu S$ on utilise l'horloge temps réel du TMS 9901.

Organigramme Général



Programme de classement

```
LWPI > 300
LI R1, 20
CLR R2
LP MOV B ω > 11EC (R1), ω > 100 (R2)
INC R2
MOV B ω > 13EC (R1), ω > 100 (R2)
INC R2
MOV B ω > 15EC (R1), ω > 100 (R2)
INC R2
MOV B ω > 17EC (R1), ω > 100 (R2)
INC R2
DEC R1
CI R1, 0
JNE LP
END
```

- C O N C L U S I O N S -

Au terme de ce projet, nous avons réalisé une carte permettant l'acquisition rapide des données d'un signal analogique dont la fréquence maximale est de 10 khz.

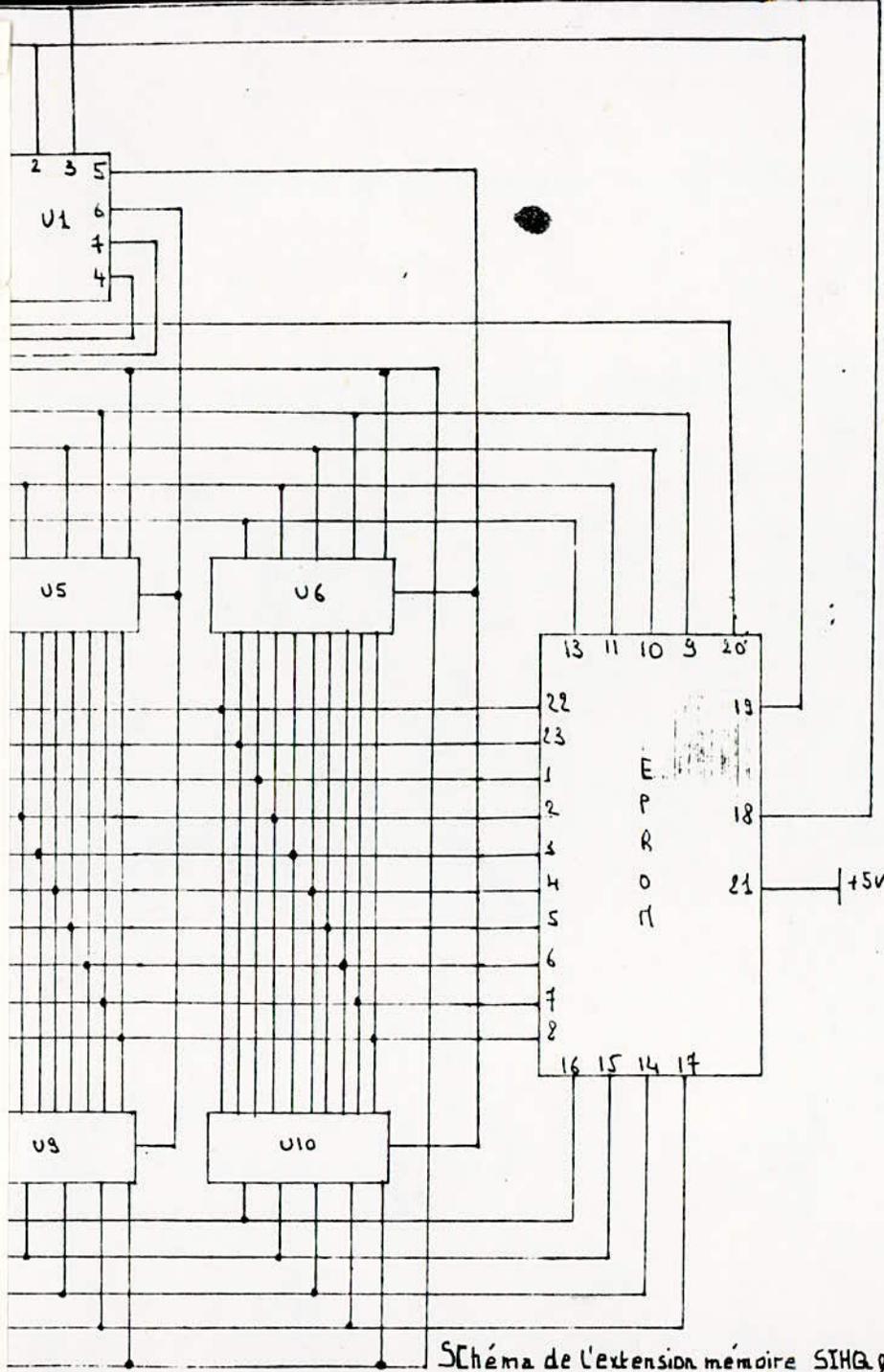
L'utilisation des signaux logiques de commande : 8 w, 4 w, 2w, et w issus de la carte analogique, nous a permis de rendre possible l'acquisition rapide du signal par l'utilisation d'une structure comprenant 4 CAN de vitesse moyenne. Ainsi on a pas eu besoin d'utiliser un CAN rapide mais très couteux.

D'où l'avantage économique de cette carte d'acquisition.

D'autre part, la réalisation de cette carte nous a permis de mettre en oeuvre un mode d'entrée/sortie extrêmement avantageux par sa rapidité : la DMA. Cette technique de transfert nous a permis de voir à quelle point elle s'avérait mieux adaptée à l'acquisition rapide des données si on la compare aux modes de transfert classiques contrôlés par programme.

Enfin, nous ajouterons, qu'en plus de la partie acquisition, cette carte offre d'autres possibilités comme : la conversion N/A et le multiplexage de 3 signaux analogiques d'entrée .

ANNEXE



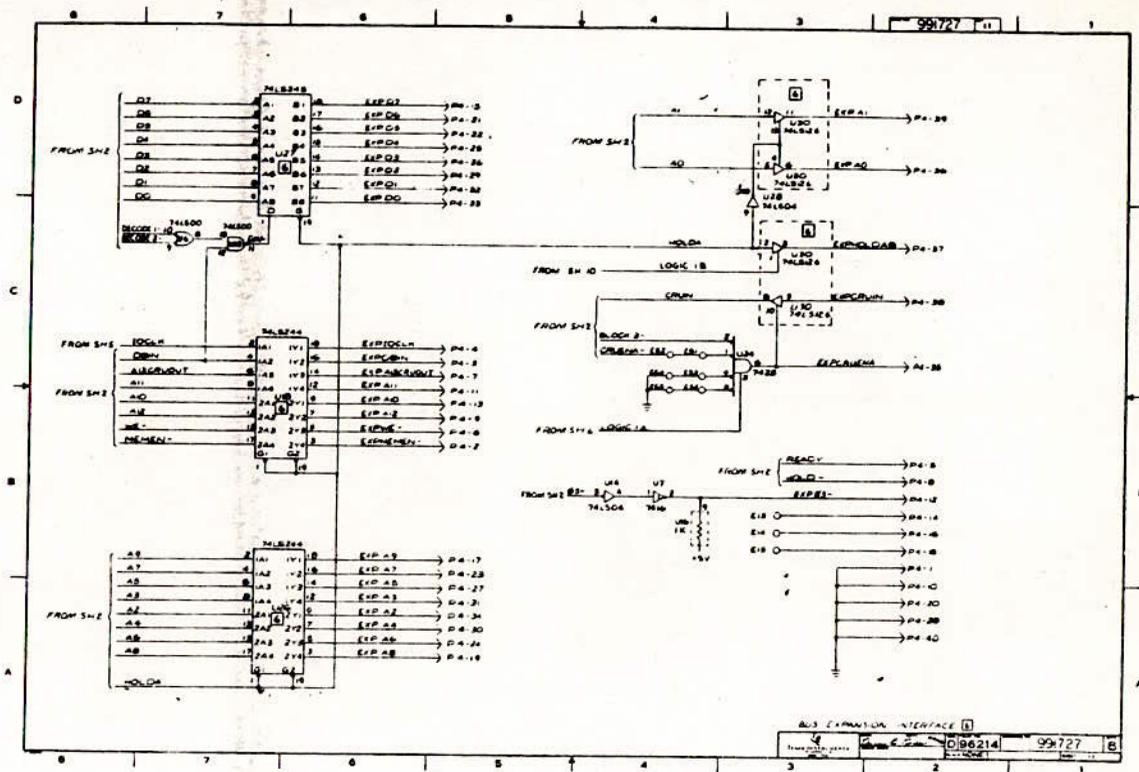
EPROM: 2516-2716-2532

- U1 : 74 LS 139
- U2 : 74 LS 02
- U3 & U4 : 2x 74 LS 139
- C1 : 10nF 10V
- C2 : 22µF 10V

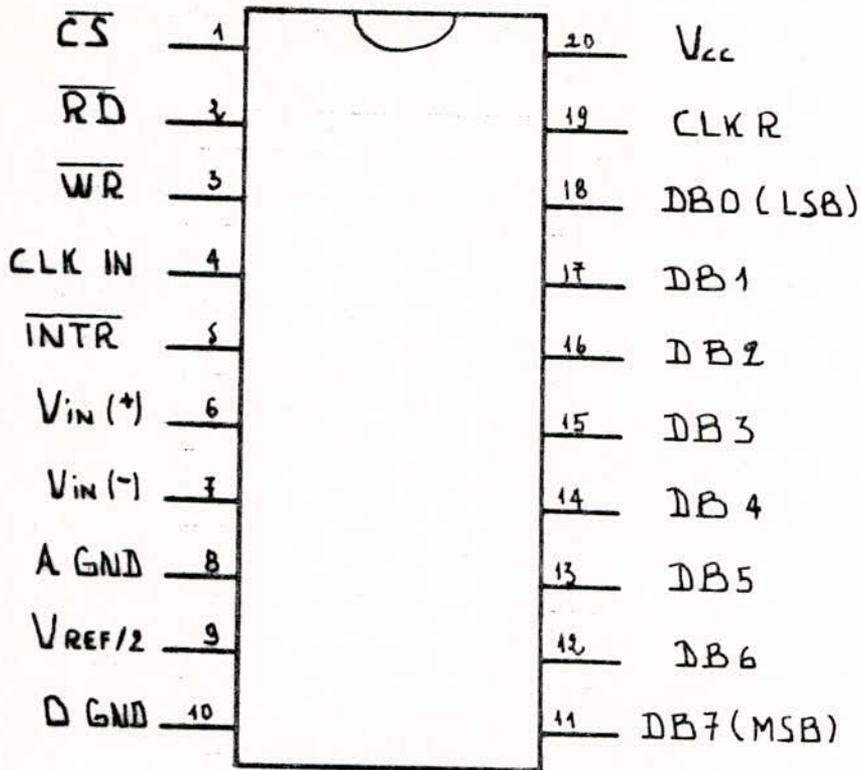
SIHQ 092

SCHEMA de l'extension memoire SIHQ 092





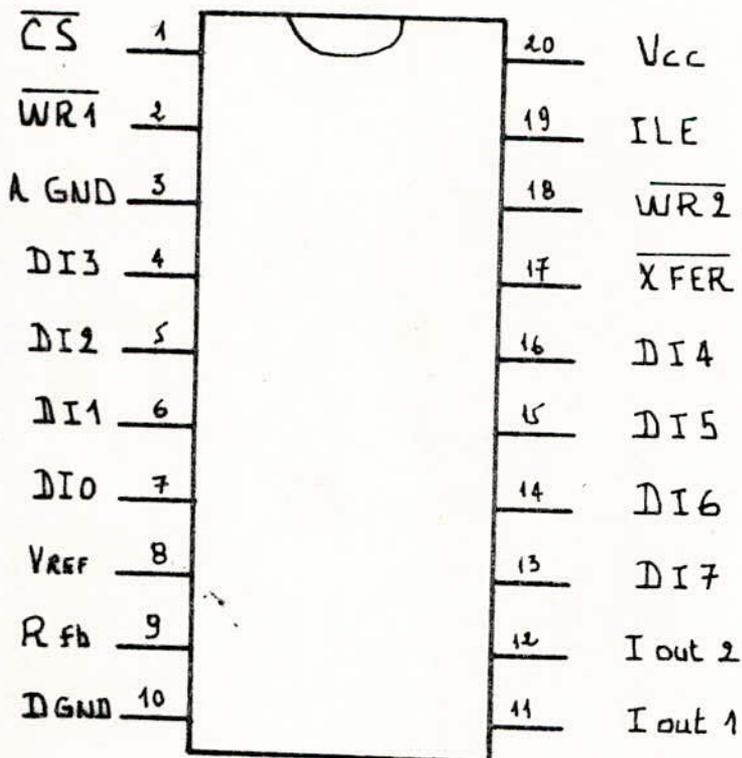
A-11



caractéristiques :

- résolution 8 bits
- temps de conversion 100 μ s
- alimentation 5 V_{cc}
- erreur de conversion ± 1 LSB

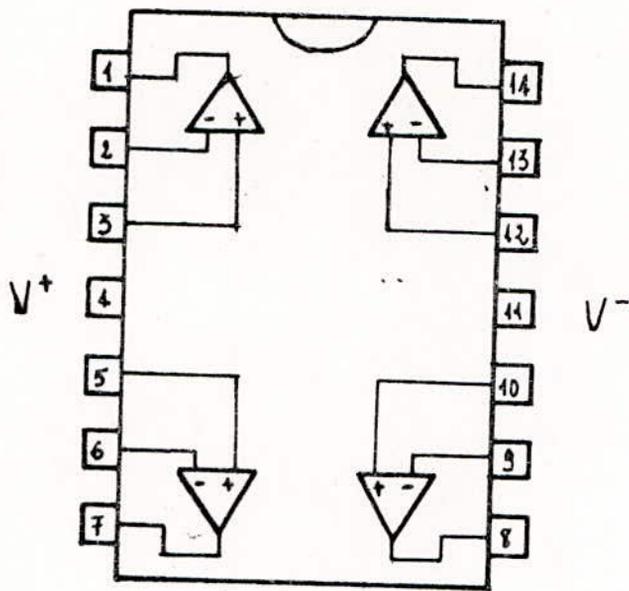
Brochage ADC 804



caractéristiques :

- résolution 8 bits
- puissance dissipée 20 mW
- temps d'établissement 1 μ s
- tension de référence ± 5 V
- tension d'alimentation 5 à 15 V_{cc}

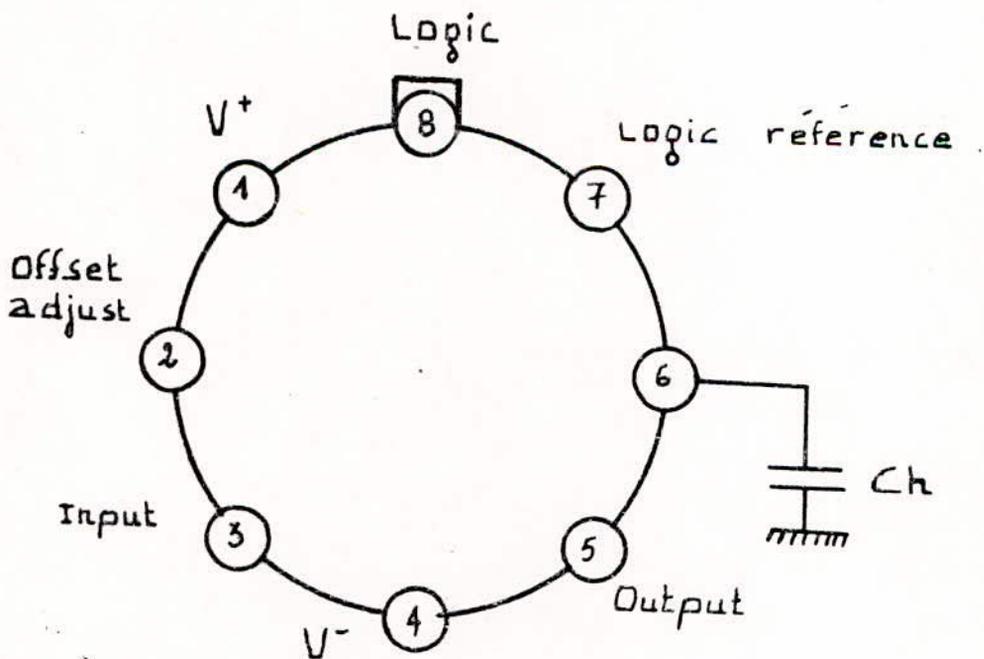
Brochage DAC 831



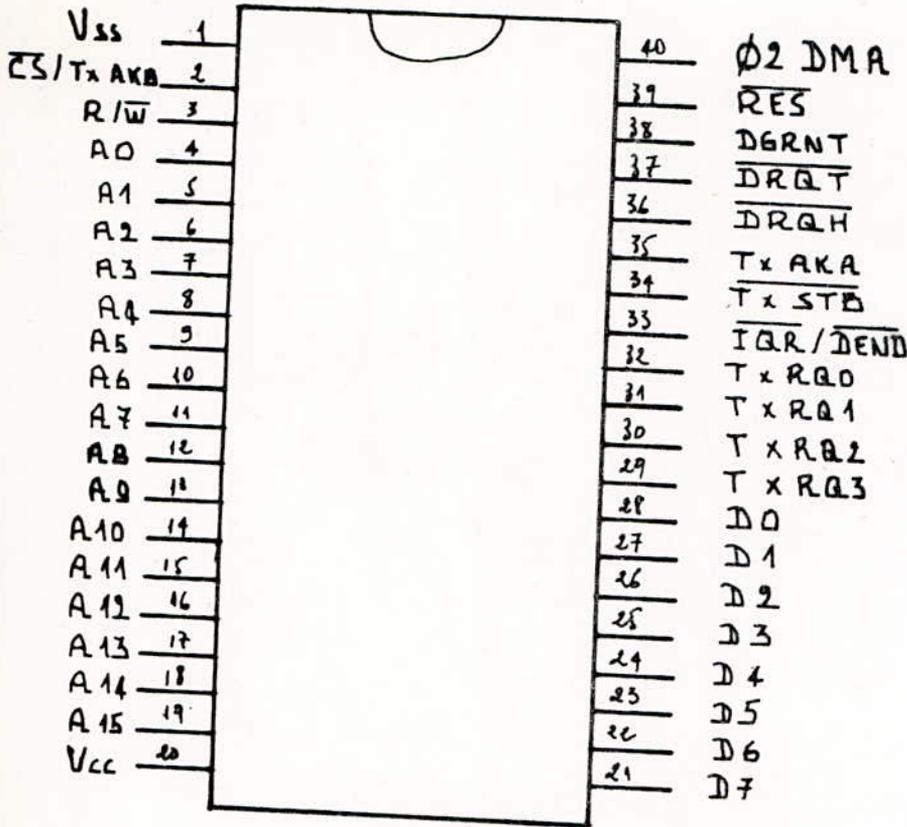
$$V^+ = +12V$$

$$V^- = -12V$$

Brochage LM 224

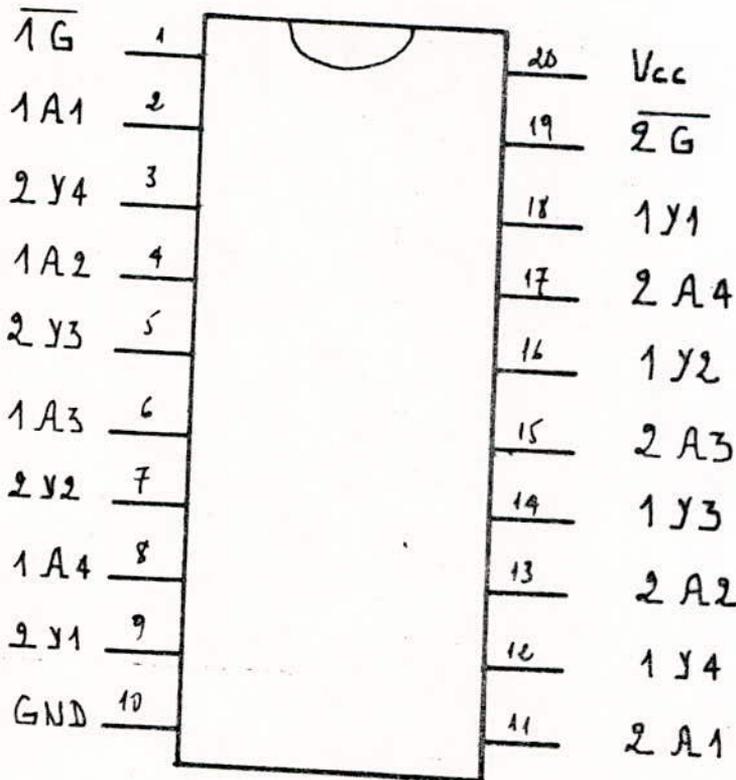


Brochage LF 398 (vue de dessus)

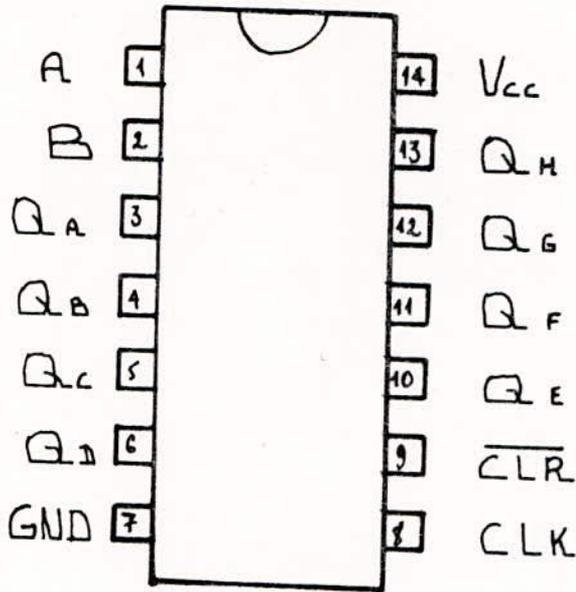


Ø2 DMA = 1 MHz
V_{CC} = 5V
V_{SS} = 0V

Brochage EF 6844

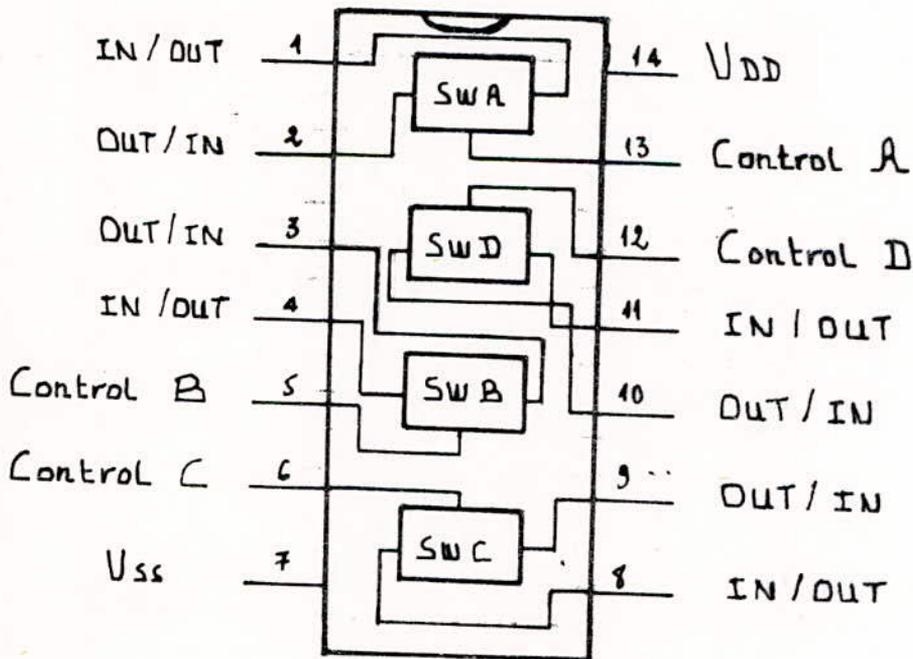


Brochage SN 74244



V_{CC} = 5V

Brochage SN 74164



V_{DD} = 5V

V_{SS} = 0V

Brochage 14066

CONNECTEUR

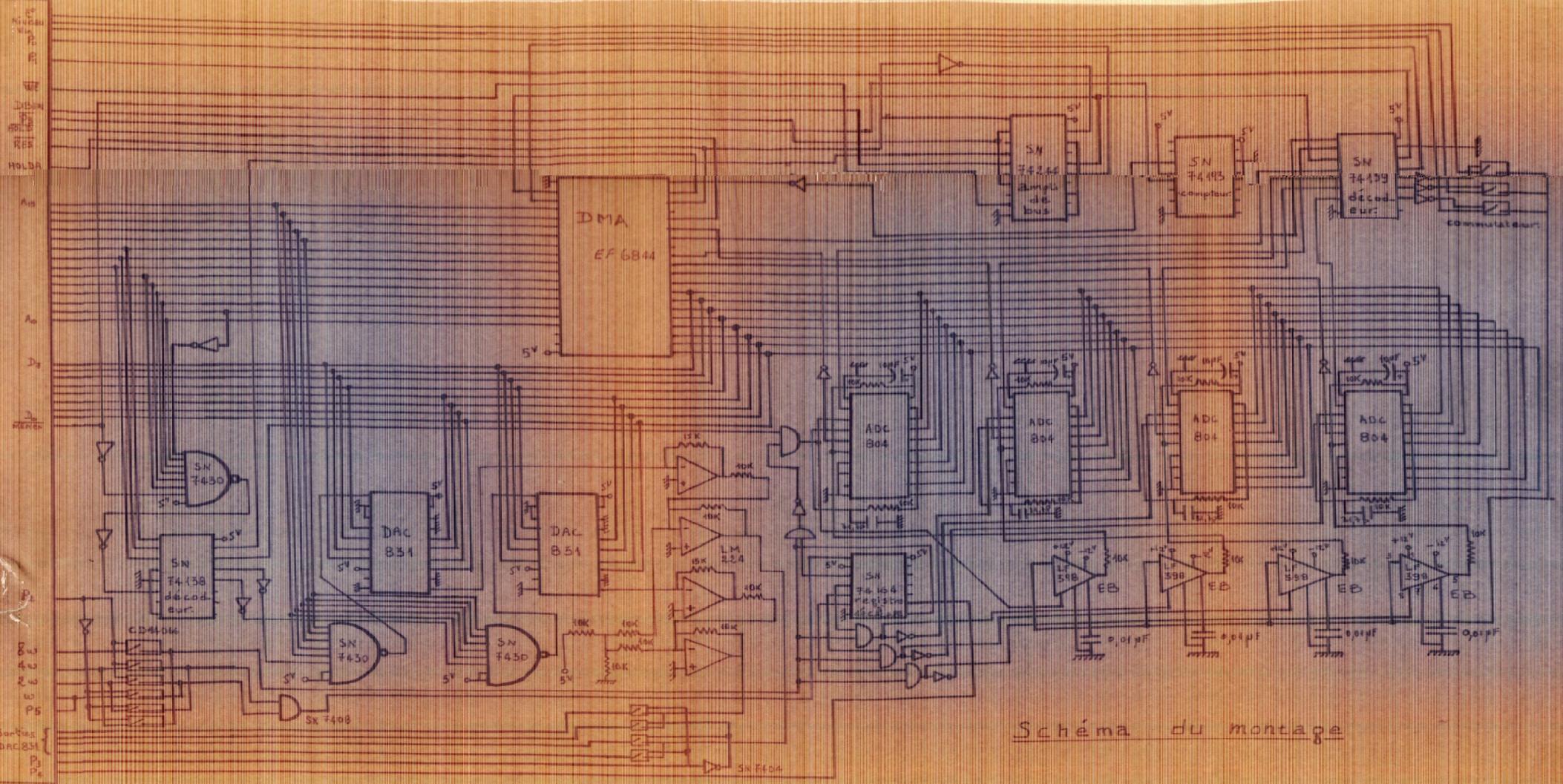


Schéma du montage