

UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE D'ALGER

42/82  
2ca

DEPARTEMENT D'ELECTRONIQUE ET D'ELECTROTECHNIQUE

FILIERE D'INGENIEUR EN ELECTRONIQUE



PROJET DE FIN D'ETUDES

SUJET :

■ UN TERMINAL VIDEO:  
ETUDE et REALISATION

PROPOSE PAR : Mr. TRABELSI

REALISE PAR :

Z.L. NAFA  
A. KHOUAS

SESSION JUIN 82

UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE D'ALGER

DEPARTEMENT D'ELECTRONIQUE ET D'ELECTROTECHNIQUE

FILIERE D'INGENIEUR EN ELECTRONIQUE



SUJET :

■ UN TERMINAL VIDEO:  
ETUDE et REALISATION

PROPOSE PAR : Mr. TRABELSI

REALISE PAR :

Z.L NAFA  
A. KHOUAS

SESSION JUIN 82



A mon père, A ma mère

A mes soeurs et frères

A tous mes camarades

A tous mes parents

A tous mes amis

GHANOU

A mes parents

A tous ceux que j'estime et que  
je chérie

ZINO

## REMERCIEMENTS

---

---

On est parfois embarrassé, surtout en cette fin de semestre passé au sein de la division six du CSTN, pour exprimer sa reconnaissance aux personnes qui nous ont guidé dans notre travail, et qui nous ont accueilli au sein de leur service; Soyez assuré du moins que l'on sera toujours prêt à vous témoigner notre pleine gratitude.

Nous avons vu naître des amitiés sincères avec les personnes qui nous entouraient, au sein de ce même service.

Nous remercions, Monsieur TRABELSI, notre promoteur pour la confiance qu'il a placé en nous et son aide particulier.

Nous remercions Monsieur BENDJABALLAH pour nous avoir accepté dans sa division.

Nos remerciements vont également à Messieurs BOUREKEB et HERRY pour leur aide, leur compréhension et le milieu de travail sympathique qu'ils nous ont offert au cours de cette période.

Nous remercions tout particulièrement Monsieur SAIDJ, pour nous avoir orienté tout le long de notre travail et pour son extrême bienveillance.

Que tout ceux qui ont contribué de près ou de loin à l'élaboration de ce projet, trouvent en ces lignes notre reconnaissance la plus gracieuse.

TABLE DES MATIERES

Introduction

Généralités sur les modes de visualisation

I/Le clavier ASCII

- le générateur de code
- la partie mécanique
- le tableau ASCII

II/L'UART

- Presentation de l'UART
- Circuit emetteur
- Circuit recepteur
- Fonctions des differentes entrées
- Operation d'emission et de reception

III/Le CRTC

- introduction
- Caractéristiques du CRTC
- Organisation interne
- Brochage et fonctions du CRTC

IV/Le décodage

- Introduction
- Principe de décodage
- Programmation
- Technique de programmation

V/Rappels sur la télévision

- Analyse et transmission de l'image
- Principe du balayage entrelacé
- Signaux de synchronisations

VI/Rafraichissement d'ecran

- Mémoire de rafraichissement
- Circuit tampon

VII/Le générateur de caractère

- Principe
- Organisation de la mémoire
- Registre à décalage

VIII/Fonctionnement général de l'ensemble

- Mémorisation du texte
- Visualisation du texte

Conclusion

Annexe

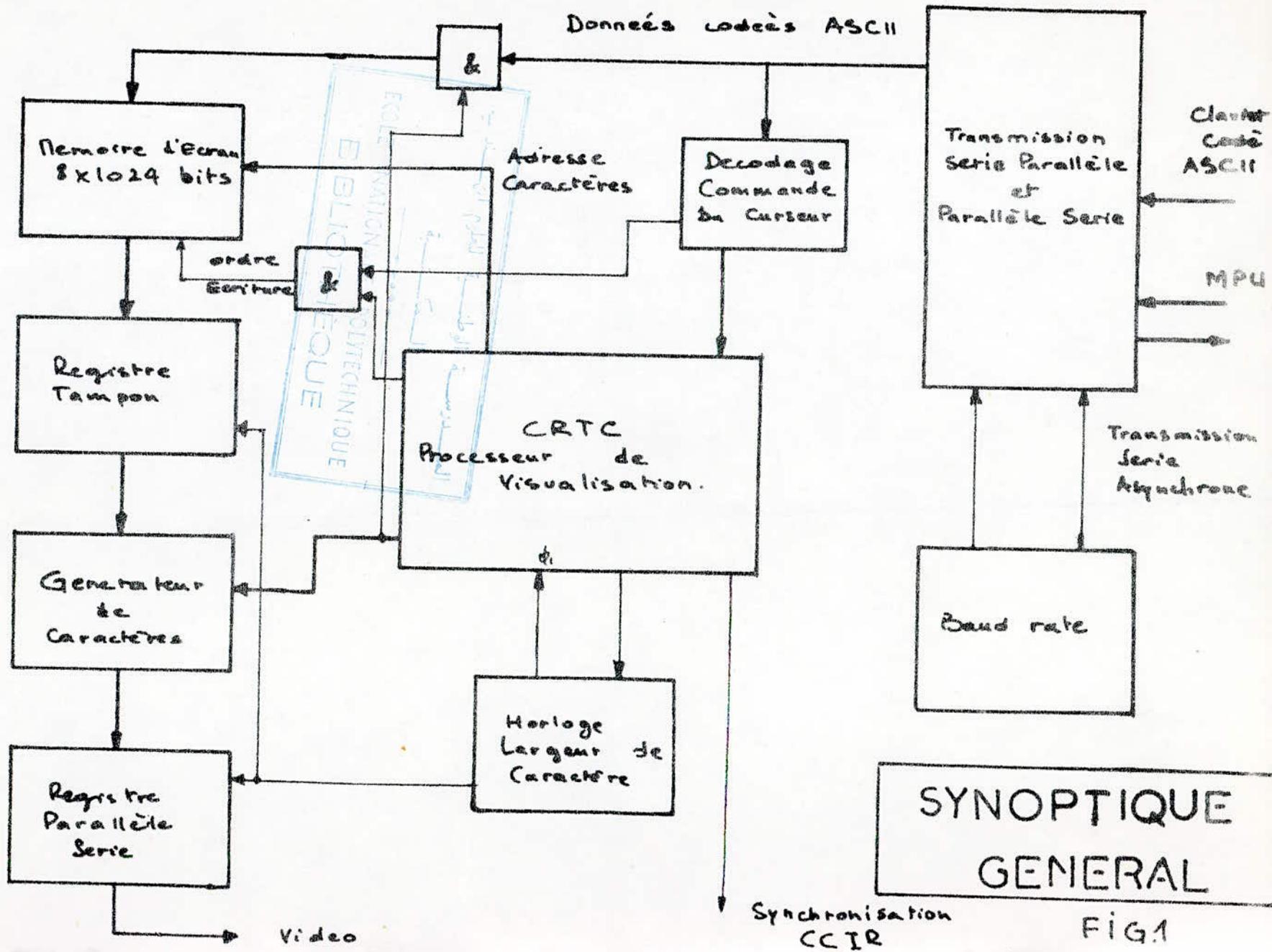
INTRODUCTION

De nos jours, la majeure partie des ordinateurs sont équipés d'un système de visualisation ou d'une télécype sur lequel travaille l'opérateur. L'objet de notre travail consiste en l'étude et la réalisation de ce système de visualisation sur écran cathodique. L'ordinateur étant équipé d'une télécype, le système de visualisation permettra l'utilisation de celui-ci sans usage de papier spécial de la télécype, en cas de panne de cette dernière, et pour tout travail qui ne nécessite pas d'être imprimé.

Ce système de visualisation ou terminal vidéo peut se décomposer en plusieurs ensembles aux fonctions bien définies

- Le clavier codé ASCII et son électronique de codage.
- L'interface asynchrone (UART) entre les circuits de visualisation et le clavier d'une part et l'équipement auquel est raccordé le terminal d'autre part.
- La circuiterie de visualisation avec le contrôleur d'écran (CRIC), la mémoire de rafraîchissement d'écran et le générateur de caractère.
- Le récepteur de télévision avec une entrée vidéo.

La figure I donne le synoptique général d'un tel circuit. Chaque élément de ce circuit fera l'objet d'une étude détaillée.



- 4 -

GENERALITES SUR LES MODES DE  
VISUALISATION

Dans le domaine des échanges ou de dialogues entre l'homme et la machine, on a souvent recours à un appareillage sur lequel l'information peut être reçue ou envoyée. Pour cela plusieurs techniques existent telles:

- la carte perforée
- la bande magnétique
- la télétype
- la visualisation

Cette dernière reste la plus utilisée à cause de sa rapidité d'exploitation, et de la lecture facile.

Lors des traitements d'information par un homme ou une machine, on est amené en fin de chaîne à traduire cette information en information visualisable. Dans ce cas le traducteur sera du type opto-electrique.

Suivant qu'il s'agisse d'une visualisation ordinaire (tel l'affichage d'horaire dans une gare) ou d'une visualisation de communication avec un ordinateur. Les solutions seront différentes. Si dans le premier cas la rapidité d'exécution n'est de rigueur, dans le second cas elle est très importante. De ce fait les circuits peuvent être électroniques électromagnétiques ou électromécaniques pour le premier cas, et doivent être typiquement électroniques pour le second cas.

Il existe plusieurs méthodes d'affichage de caractères alphanumériques. Les principales sont:

## I/Affichage par segments

Le dessin d'un caractère est fait par la juxtaposition de segments lumineux et éteints. Ces segments peuvent être :

- des lampes à filament
- des tubes neon
- des LED (diodes électroluminescentes)

### I.1/Affichage par 7 segments

Le format 7 segments est généralement utilisé pour l'affichage des chiffres décimaux, et éventuellement certains caractères à géométrie simple.

### I.2/Affichage par I4 ou I6 segments

Ce format a été conçu évidemment pour afficher les chiffres décimaux et toutes les lettres de l'alphabet.

## II/Affichage par sélection

Dans ce mode d'affichage, on utilise un tube indicateur qui contient un gaz et des cathodes qui épousent le contour des caractères. Lorsqu'une tension est appliquée entre l'anode commune et l'une des cathodes, le gaz s'ionise et enveloppe la cathode sélectionnée avec émission de lumière: le caractère est ainsi visualisé.

## III/Affichage par points

Cette méthode d'affichage étant l'objet d'une étude détaillée dans les prochains chapitres, nous dirons simplement que celle-ci consiste à dessiner les caractères dans des matrices de points (les uns blancs, les autres noirs).

Le choix entre les différents modes de visualisation dépend des paramètres suivants :

- la rapidité d'exécution
- la densité de texte et le nombre de caractère à visualiser
- la dimension du caractère
- la fiabilité du système
- le cout du système

Pour chaque application, le choix sera porté sur la méthode qui lui conviendrait le mieux. Pour la réalisation d'un terminal d'ordinateur, le choix a déjà fait. La visualisation se fait sur écran de télévision. Il aurait été aberrant d'utiliser une multitude d'afficheurs I6 segments pour visualiser un texte.

## Chapitre I.

### LE CLAVIER ASCII

Avant que l'information arrive au circuit de visualisation pour traitement et visualisation, elle doit être codée. Pour cela on utilisera un clavier Electronique qui donnera à sa sortie, pour chaque caractère, un code.

Pour notre application, nous avons choisi un clavier qui génère le code ASCII.

L'ASCII permet de coder tous les symboles utiles, les majuscules, les minuscules, les chiffres et les mouvements du curseur (voir tableau ASCII).

A chaque enfoncement de touche, le clavier codera sous forme de combinaison 7 bits le caractère élu.

#### Le clavier ASCII

C'est un clavier électronique qui est composé principalement de deux parties :

- Le générateur de code ou encodeur.
- La partie mécanique ou touches.

#### 1°/ Le générateur de code.

L'encodeur utilisé est le 7310. Ce circuit est composé essentiellement d'une mémoire ROM contenant le code des 128 caractères ASCII.

La figure 1 représente le schéma électronique du clavier. Le principe adopté par ce circuit est l'émission d'un caractère lors de chaque sollicitation d'un point de la matrice 8 lignes (X0 - X7) et 11 colonnes (Y0 - Y10).

./.

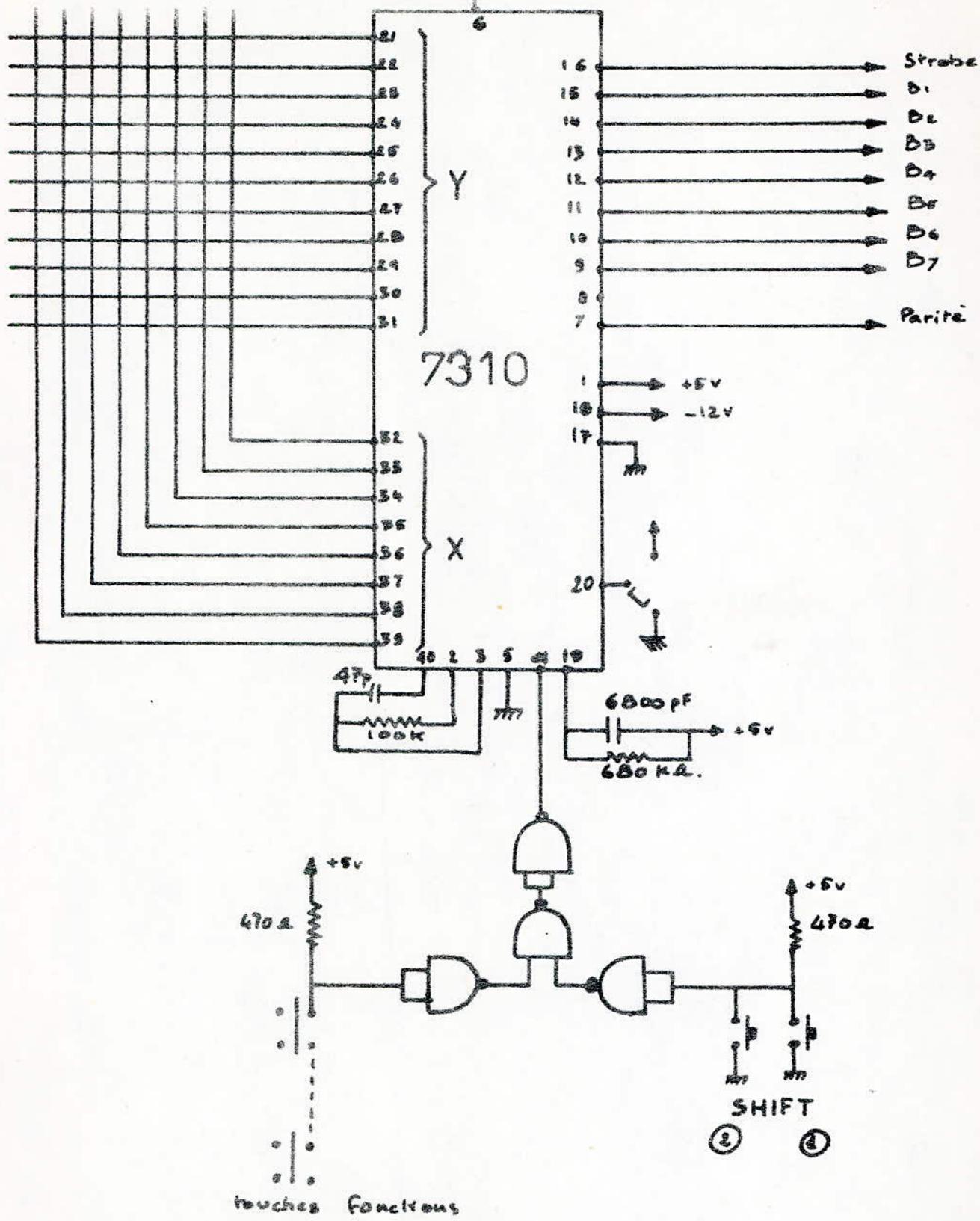


FIG1 LE GENERATEUR DE CODE DU CLAVIER

La sollicitation de l'un des 88 points de la matrice générera un caractère dans un mode dit normal. Les 30 caractères manquants sont générés en sollicitant à la fois l'une des deux touches SHIFT ou CONTROL et un point de la matrice.

Nous avons donc 3 modes possibles.

- Mode normal
- Mode SHIFT
- Mode CONTROL

Soit la possibilité d'avoir  $3 \times 88 = 264$  combinaison. Certains codes ou caractères seront alors utilisés plusieurs fois et cela dans des modes différents.

Pour notre application, le clavier ne comporte que 2 modes seulement :

- Mode Normal
- Mode SHIFT

Le code ASCII émis par le générateur de code est composé de 7 bits en parallèle et d'un huitième bit appelé bit de parité.

La validation de ce code se fait par un signal appelé STROBE. Ce signal ne sera émis que lors de l'arrêt complet des rebondissements de la touche entre son enfoncement et son relachement (Voir Fig. 2). En effet, lors de l'enfoncement de la touche, les contacts mis en présence peuvent rebondir l'un sur l'autre. Un temps de garde ou d'attente est nécessaire.

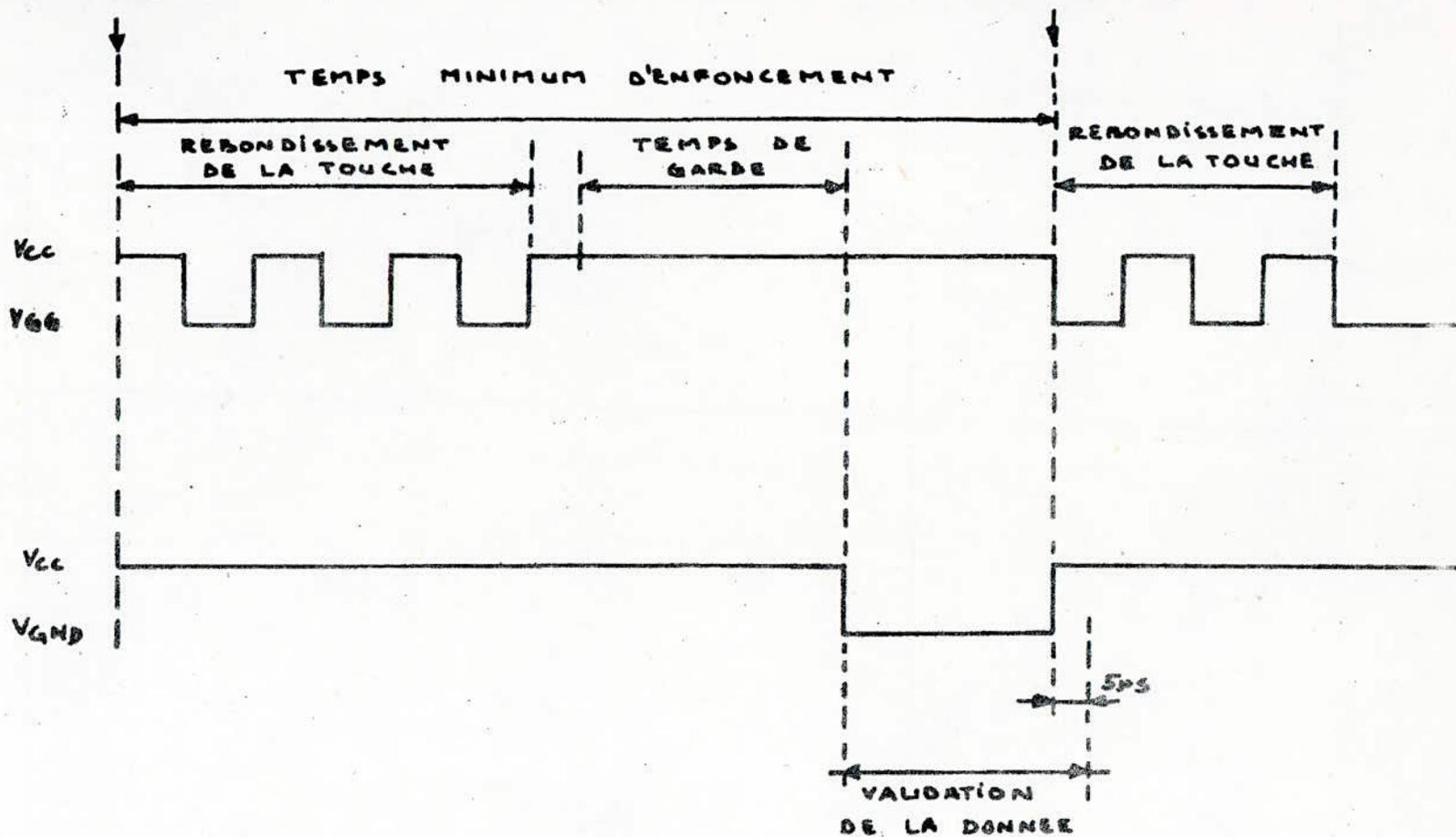
Le circuit interne possède en outre une horloge réglable par les composants R3 et C1. Le circuit donne la possibilité de générer le STROBE et les données en direct (broche 20 à la masse) ou en inverse (broche 20 en l'air).

La broche 19 fixe le délai d'élimination des rebondissements mécaniques des touches du clavier.

La broche 6 sélectionne la parité paire ou impaire, ceci est sans aucun intérêt car la sortie parité n'est pas exploitée.

ENFONCEMENT  
DE LA TOUCHE

RELACHEMENT  
DE LA TOUCHE



-43-

FIG2

DIAGRAMME DES TEMPS ENTRE L'ENFONCEMENT ET  
LE RELACHEMENT D'UNE TOUCHE

2°/ . La partie mécanique : les touches.

Le clavier est composé de 46 touches (ou boutons poussoirs) sur la partie gauche. Chacune représentant un caractère à visualiser. On trouvera 2 touches pour le mode SHIFT et une touche Space multicontact.

Sur la partie droite du clavier on utilisera 9 des 20 touches disponibles. Ces touches gèreront les mouvements du curseur.

Les fonctions et codes correspondants à ces touches sont :

TOUCHE	CODE	FONCTION
BS	08	Retour d'une position à gauche.
HT	09	Retour d'une position à droite.
LF	0A	Descente d'une position.
VT	0B	Montée d'une position.
FF	0C	Effacement de la page et retour en haut à gauche.
CR	0D	Effacement de la fin de ligne et retour en début de ligne.
SUB	1A	Effacement de ligne courante du curseur.
FS	1C	Retour du curseur en haut à gauche.
GS	1D	Retour du curseur en début de ligne.

TABLEAU : CODE ASCII

					b7	0	0	0	0	1	1	1	1
					b6	0	0	1	1	0	0	1	1
					b5	0	1	0	1	0	1	0	1
						0	1	2	3	4	5	6	7
b4	b3	b2	b1		0	NUL	DEL	SP	0	G	P		P
0	0	0	0	0	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	4	EOT	DC4	\$	4	D	T	d	t
0	1	0	1	5	5	END	MARK	%	5	E	U	e	u
0	1	1	0	6	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	7	DEL	ETS	'	7	G	W	g	w
1	0	0	0	8	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	9	UPS	EM	)	9	I	Y	i	y
1	0	1	0	10	10	LF	SUB	#	:	J	Z	j	z
1	0	1	1	11	11	WPL	LF	+	;	K	[	k	{
1	1	0	0	12	12	Clear	H <sup>9</sup>	,	<	L	\	l	:
1	1	0	1	13	13	Clear	RC	-	=	M	]	m	}
1	1	1	0	14	14	SO	RS	.	>	N		n	~
1	1	1	1	15	15	SI	US	/	?	0	-	o	DEL



Code ignoré



Code: mouvement du curseur

L' U. A. R. P.

Lors de la réalisation d'ensembles logiques complexes la nécessité de transférer des informations digitales entre deux ou plusieurs unités se présente souvent.

La transmission peut-être parallèle ou série. La transmission parallèle est utilisée pour des échanges de données très rapide entre deux unités. Chaque variable binaire est transmise sur un fil. La transmission de caractères ASCII nécessite huit fils de liaison et un fil d'horloge (fig.1). La (nécessité) transmission série n'utilise qu'une seule ligne pour transmettre les caractères. Les variables binaires sont transmises l'une après l'autre par le canal de transmission. Ce principe convient pour des transmissions vers des unités lentes (fig.2).

Fig.1

Transmission Parallèle

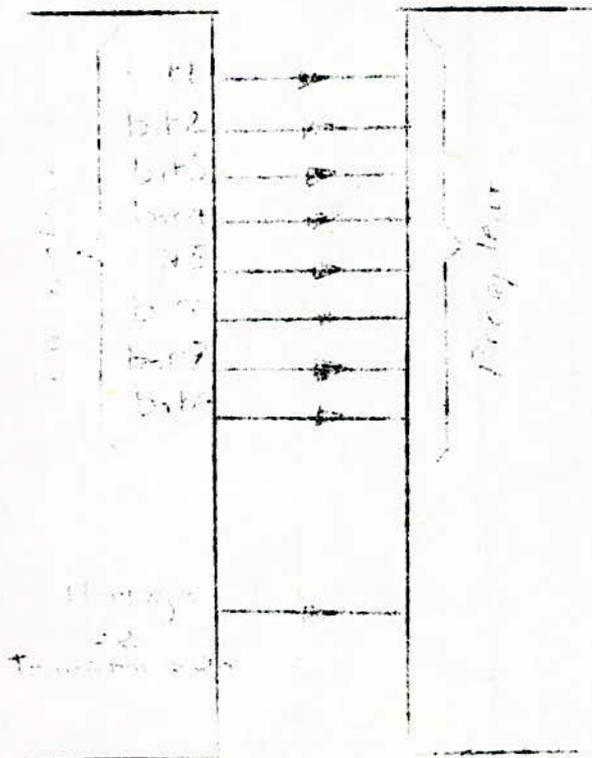


Fig.2

Transmission Série



Le transfert peut-être synchrone  $\Delta$  ou asynchrone. Dans le premier cas, les échanges sont souvent contrôlés par des petits ordinateurs. Dans le second cas, les échanges sont contrôlés par des quittances.

Ce système a le gros avantage d'adapter les vitesses de transmission à celle des autres unités. Le circuit AY-9-10-12 a été conçu pour des applications correspondantes à ce dernier cas.

Comme son nom l'indique l'UART (Universal asynchronous Receiver Transmitter) est un émetteur récepteur universel de données asynchrones. Il peut transmettre et recevoir des caractères de 5 à 8 bits. Un circuit de contrôle interne permet de vérifier automatiquement la validité des caractères de contrôle (parité, initialisation et arrêt) reçus ou générés.

Les entrées et sorties du circuit sont compatibles au niveau DTL et TTL sans adjonction de composants externes. Les sorties trois états peuvent être connectées sur des lignes de bus, ce qui facilite les échanges entre les différents circuits.

Le synoptique de ce circuit LSI (fig. 3) nous montre qu'il est en fait divisé en deux parties quasiment indépendantes et capable de travailler sur des tâches différentes.

Ces deux parties utilisent un registre à décalage et de la logique de contrôle. L'une est le récepteur, les données y entrent sous forme série dans le registre à décalage commandé par une horloge RCP et elles sont converties en données parallèles. Du côté émetteur un registre à décalage effectue le travail inverse; c'est à dire transforme les informations parallèles issues d'un ordinateur ou d'une unité périphérique en informations série, commandé par une autre horloge TCP.

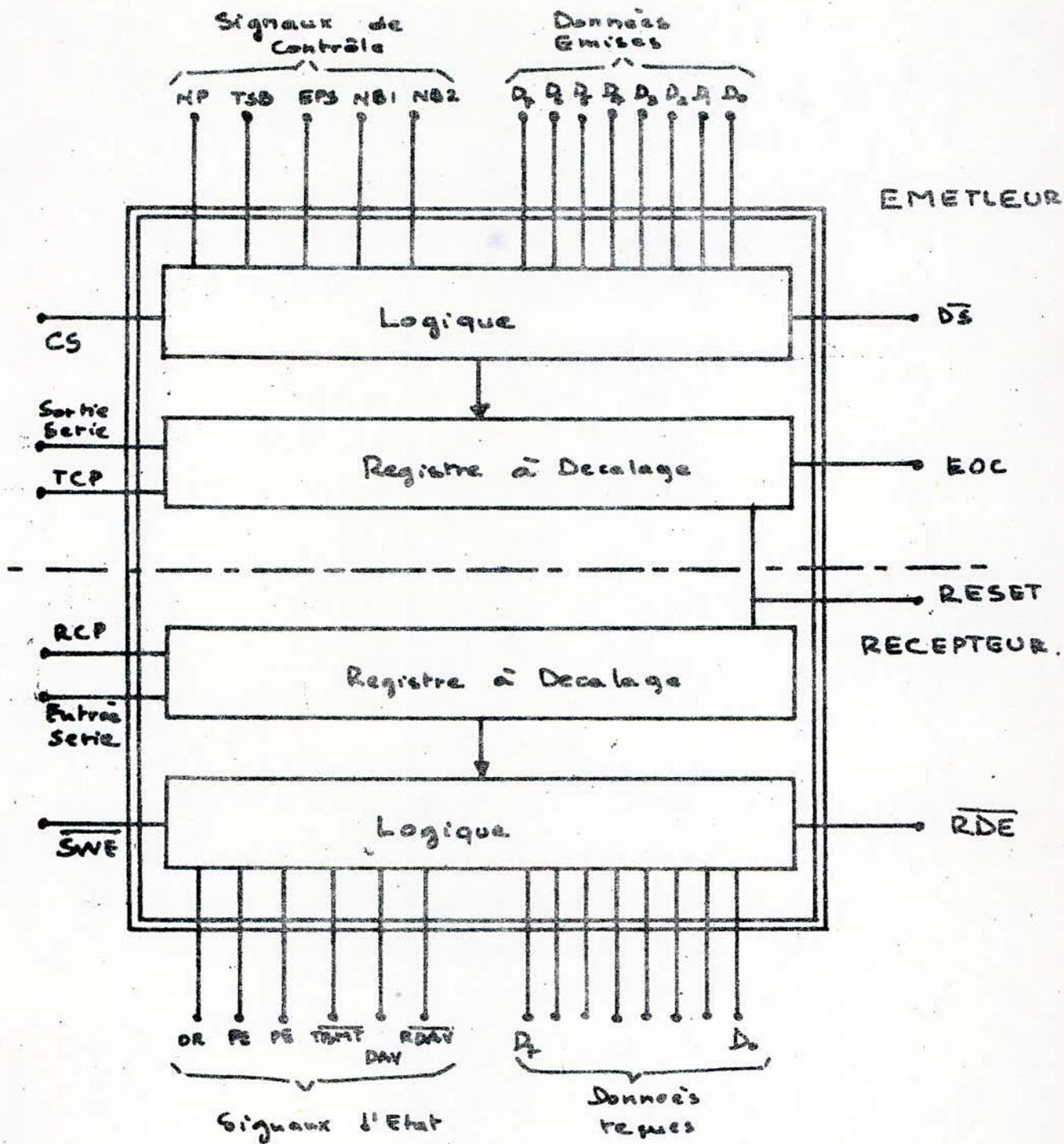


FIG3 SYNOPTIQUE INTERNE DE L'UART

## 2.1 Circuit emetteur

Il est composé essentiellement d'un registre à décalage à sortie serie, des circuits supplémentaires generent le premier bit (bit start), la parité des 5, 6, 7 ou 8 bits des données transmises et le ou les bits de fin de mot (1 ou 2 bits stop). Le format de caractère est alors de la forme :

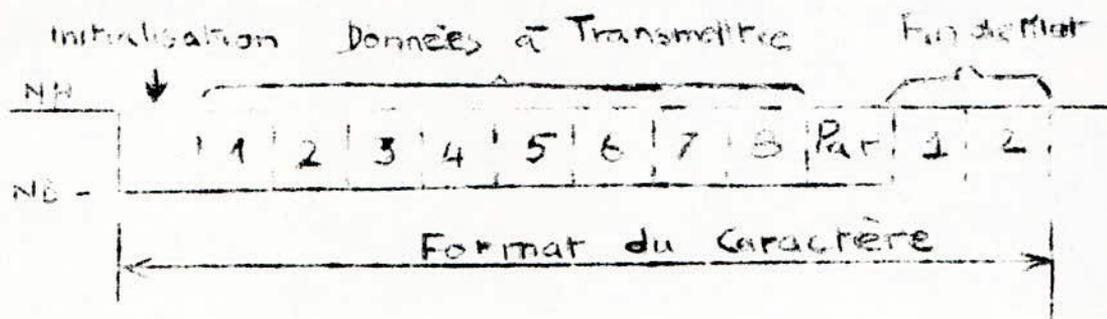


Fig.4

## 2.2 Circuit recepteur

Le circuit recepteur transforme le caractère serie issu de l'emetteur ou de tout autre système utilisant le format du caractère de la figure 4 sous forme parrallèle disponible sur 8 bits.

Il est composé d'un registre à accès serie et d'un registre tampon où sont stockées les données en attente de lecture. Un signal d'horloge permet de transférer les informations du registre de reception au registre tampon. D'autres circuits verifient: la parité, le cadrage, les surcharges et indiquent qu'un caractère/ entier a été reçu.

## 2.3 Fonctions des differentes entrées et sorties

1/tension: +5 V

2/tension: -12V

3/masse

4/Received data enable (RDE): un niveau bas sur cette entrée autorise la deconnexion du registre recepteur du registre tampon

5 à 12/Received data bits (RD8&RDI): ce sont les huit bits de sortie des données.

- 13/Erreur de parité (PE):un niveau haut indique qu'une erreur de parité a été détecté a la reception en fonction de la ligne de contrôle de parité ~~ppp~~ (LPS)
- 14/Erreur de cadrage (FE):un niveau haut indique que le bit d'arrêt n'est pas validé
- 15/Erreur de surcharge(OR): un niveau haut indique que le caractère reçu auparavant n'a pas été lu avant que le present caractère n'ait été transféré dans le registre tampon du recepteur
- 16/Commande des indicateurs(SWE): un niveau bas sur cette entrée deconnecte les sorties des indicateurs d'erreur et de fonctionnement (PE,FE,OR,DA,TBMT)
- 17/Horloge du registre recepteur(RSP):cette horloge est 16 (seize) fois plus rapide que celle de reception des données
- 18/remise à zero de DA (RDA):un niveau bas sur cette sortie remet à zero la sortie DA
- 19/Indicateur de reception des données (DA):un niveau haut sur cette sortie indique qu'un caractère a été reçu et transféré dans le registre tampon du recepteur
- 20/Entrée serie (SI):les informations serie reçues sur cette ligne,entrent dans le registre du recepteur à l'endroit déterminé par la longueur du mot,la parité,et le nombre de bits d'arrêt
- 21/remise a zero externe(XR):remise à zero de tout les registres et mise à 1 de SO,EOC,etTBT.
- 22/registre tampon de l'emetteur vide(TBMT):un niveau haut sur cette sortie indique que le contenu du registre tampon de l'emetteur a été transféré dans le registre de l'emetteur ainsi un nouveau caractère peut être chargé.
- 23/Data strobe (DS):un strobe sur cette ligne fait entrer les bits du caractère dans le registre tampon;la transmission commence au front montant de DS

- 24/Fin de mot (EOC):cette  $\rho$  sortie va au niveau haut à chaque fois qu'un caractère complet a été transmis.Elle reste à l'etat haut jusqu'au prochain caractère.
- 25/Sortie serie (SO):le contenu du registre d'emission est décalé en serie sur cette sortie.Il reste à l'etat haut en l'absence d'emission.
- 26 à 33/Entrées parrallèles des données(DBI-DB8):ce sont les huit entrées parrallèles des données.
- 34/Contrôle strobe (CS):un niveau haut sur cette ligne charge le registre de contrôle avec les bits de contrôle EPS,NBI, NB2,TSB et  $\rho$  NP.Cette entrée peut être commandé par une impulsion ou relié à un niveaue haut.
- 35/Suppression du bit de parité(NP):un niveau haut sur cette ligne supprime la generation et la detection du bit de parité ,par contre un niveau bas les autorise
- 36/Nombre de bits d'arrêt (TSB):cette entrée selectionne le nombre de bits d'arrêt qui suivent le bit de parité.Un niveau haut selectionne deux bits d'arrêt,et un niveau bas,un bit.
- 37-38/Selection de la longueur du mot(NBI-NB2):ces deux lignes selectionnent la longueur du caractère de 5,6,7 ou 8 bits

NBI	NB2	Longueur du mot
0	0	5
I	0	6
0	I	7
I	I	8

- 39/Selection de parité(LIS)/cette ligne selectionne la parité paire ou impaire.Un niveau haut selectionne une parité paire ,un niveau bas une parité impaire.
- 40/Horloge de transmission(TAC):l'horloge est seize fois plus rapide que la vitesse de transmission desirée.

Opération de transmission.

Initialisation : après avoir établi le contact, la remise à zéro extérieur est possible et l'impulsion d'horloge est appliquée (cette impulsion à une fréquence 16 fois la vitesse désirée).

Après initialisation complète, l'utilisateur peut envoyer les bits de contrôles et les bits d'information (la sélection des bits de contrôles doit intervenir avant la sélection des bits d'informations).

On peut émettre la  $\overline{DS}$  et le  $\overline{CS}$  simultanément si les spécifications de largeur minimale de l'impulsion sont respectées. Une fois que le data strabe  $\overline{DS}$  est émis, le signal T.B.M.T. va passer de l'état haut 1 à l'état Bas 0, indiquant que le registre de maintenance des bits d'informations est plein avec un caractère précédent celui-ci. Il est incapable de recevoir de nouveaux bits d'information. Le registre à décalage est entrain de transmettre une information précédemment chargée.

./.

TBMT retournera à l'état haut 1 lorsque le registre de décalage émetteur sera vide. Les bits d'informations du registre de maintenance sont immédiatement chargés dans le registre de décalage émetteur pour transmission. Le décalage des informations d'un registre à l'autre sera contrôlé par SO et EOC (=0). TBMT ira à l'état haut 1 indiquant que l'opération de décalage est finie et que le registre de maintenance est prêt à accepter une nouvelle information.

La transmission d'information est initialisée par un bit de départ (ou start). Suivant ensuite les bits d'informations, le bit de parité (si l'on veut) et les bits d'arrêt. Quant le dernier bit d'arrêt passe sur la ligne de transmission EOC ira à 1 indiquant que le nouveau caractère est prêt pour la transmission. Ce nouveau caractère ne sera transmis que si TBMT est à zéro.

#### Opération de réception.

Initialisation : Le contact est établi, la remise à zéro extérieure est possible et l'impulsion d'horloge est établie. Cette impulsion à la même fréquence que celle employé par le circuit de transmission. Les conditions précédentes mettront DA à 0 une fois l'initialisation terminée, on notera que l'ensemble des bits de contrôle sera utilisé à la fois pour le récepteur et l'émetteur rendant inutile des ensembles de contrôle individuels. La réception des données commencera si la ligne de transition est vide. Ceci annonce le bit de départ. Celui-ci est validé si la ligne SI passe à zéro. Après cette vérification la réception des bits de données, de parité et d'arrêt se fera d'une manière ordonnée. Pendant qu'il reçoit les bits de parité et d'arrêt, le récepteur comparera ces bits avec les bits de contrôle (parité et nombre de bits d'arrêt) précédemment envoyés. Il indiquera une erreur en changeant la bascule erreur de parité et la bascule erreur de cadrage.

- 24 -

Il faut remarquer que si le mode sans parité est choisi, PE (erreur de parité) sera systématiquement placé à l'état bas.

Une fois qu'un caractère entier est reçu, la logique interne contrôlera si la donnée disponible est bien celle qui a été lue. Si le signal DA est à 1, le récepteur comprendra que la donnée n'a pas été lue et mettra le registre de surcharge à 1. Si par contre le signal DA est à 0 le récepteur comprendra que la donnée a été lue.

Une fois que DA passe à 1, le registre à décalage du récepteur est prêt à accepter le prochain caractère.

### Compatibilité des niveaux entrées-sorties de l'UART avec l'ordinateur

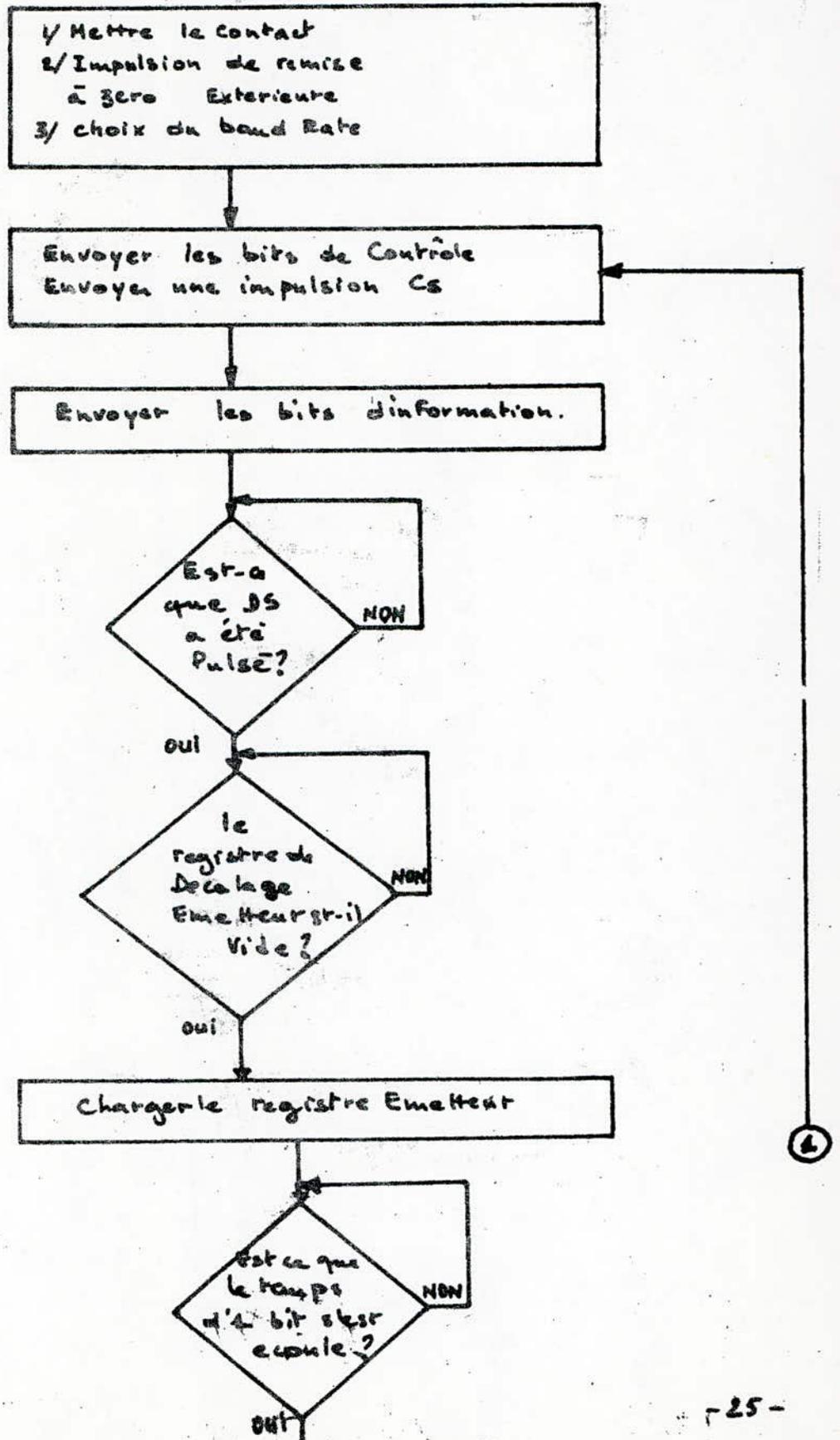
Afin de rendre compatibles les entrées-sorties de l'UART avec une unité centrale, nous avons été amenés à réaliser des montages adaptateurs de niveau pour l'entrée série et la sortie série de l'UART.

Les montages utilisés sont les 4N33 (boitier 6 broches) Ils jouent le rôle de coupleurs-isolateurs et d'interface entre les différents niveaux des différents systèmes.

### Le baudrate

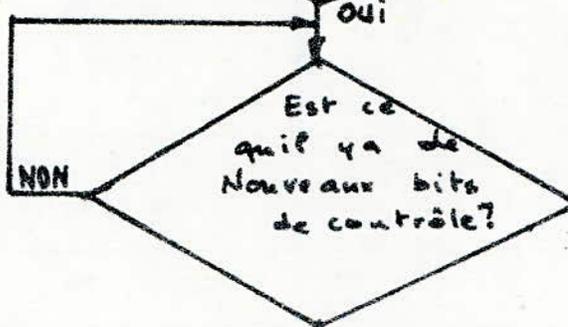
C'est un generateur de signaux carrés qui donnera le cycle d'horloge pour la transmission et la reception des données pour l'UART. Cette vitesse s'exprime en baud: c'est le nombre de bits transmis ou reçus par unité de temps.

I/ OPERATION DE TRANSMISSION



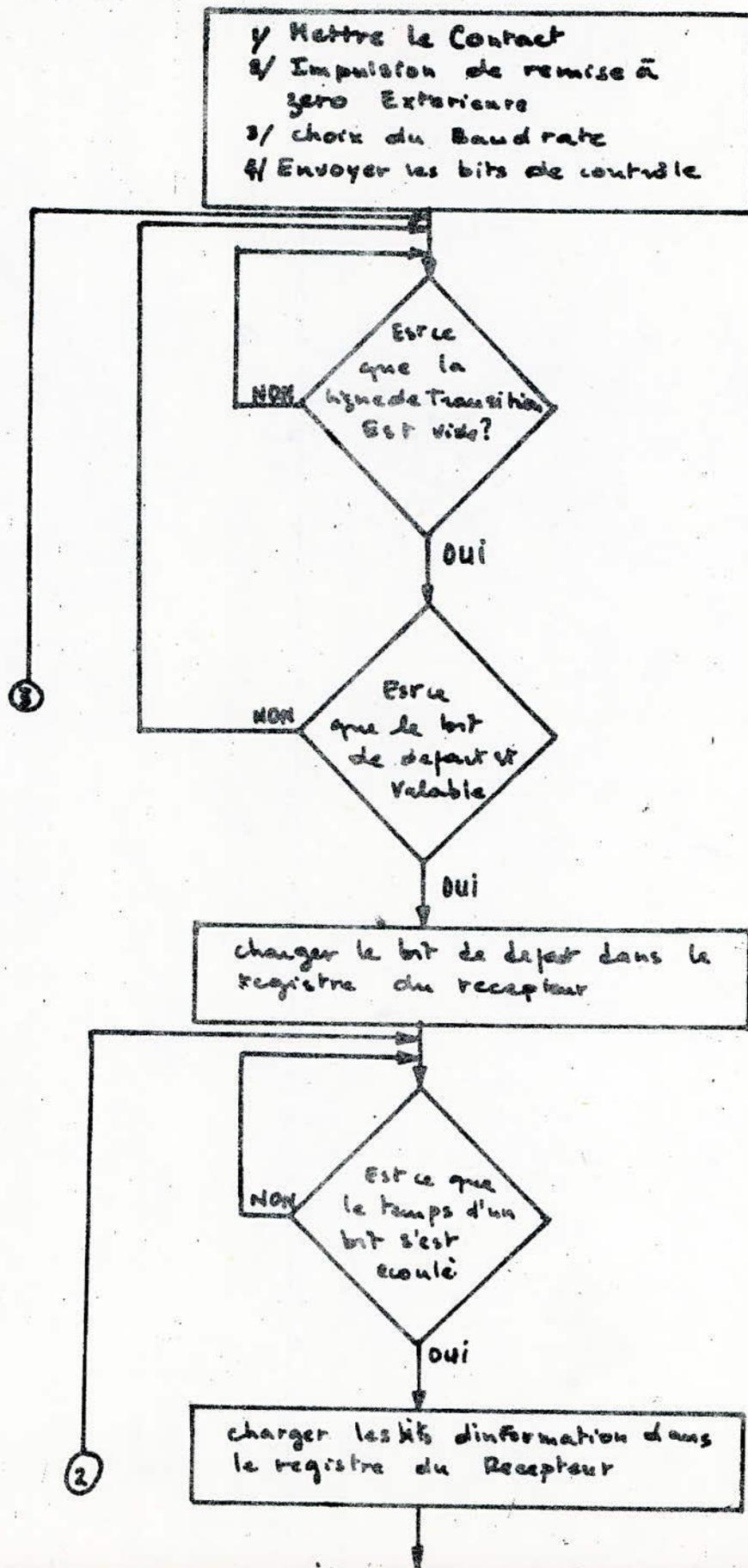
Decaler les bits directement dans le registre à décalage émetteur

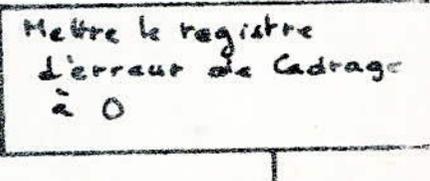
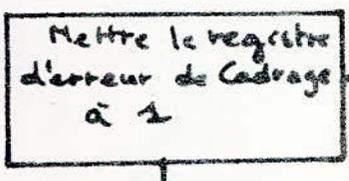
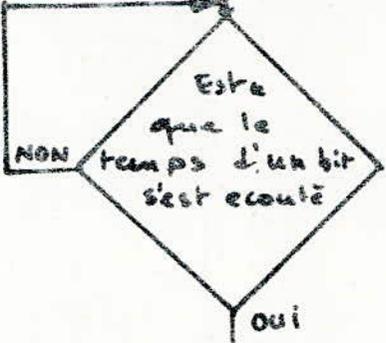
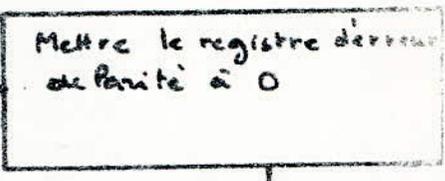
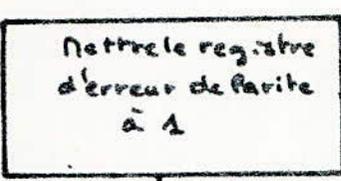
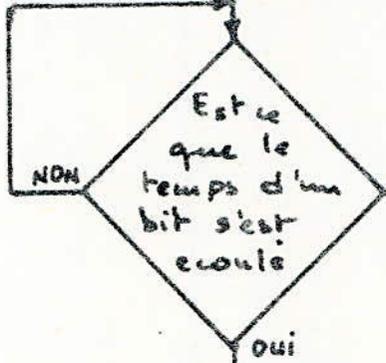
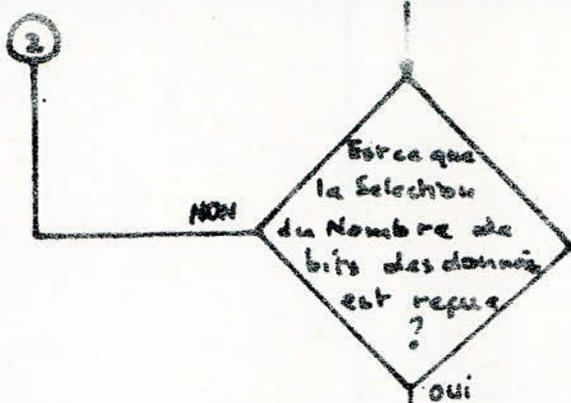
Transmettre : - 1 bit de départ  
- les bits d'information  
- le mode de parité  
- les bits d'arrêt

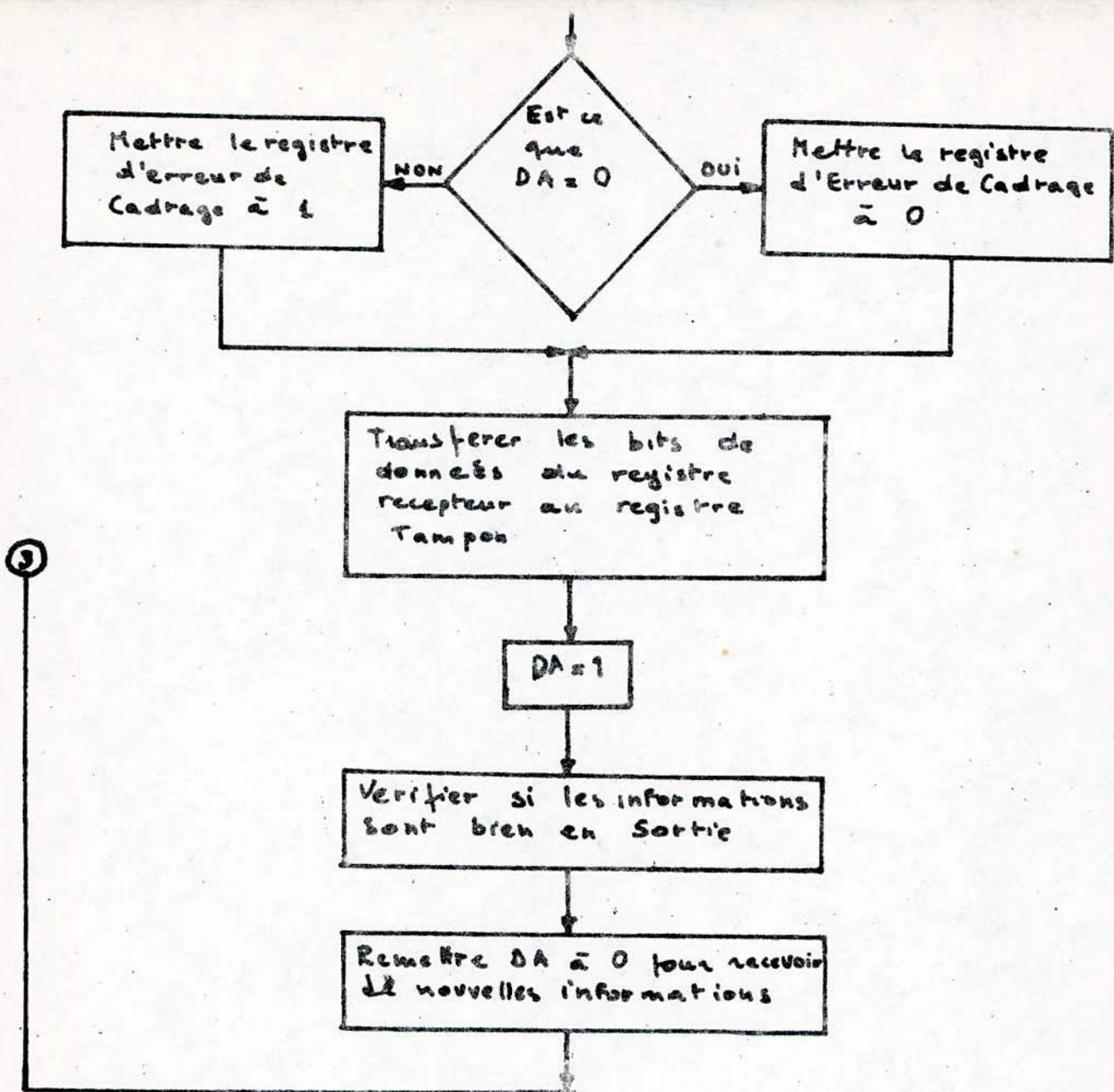


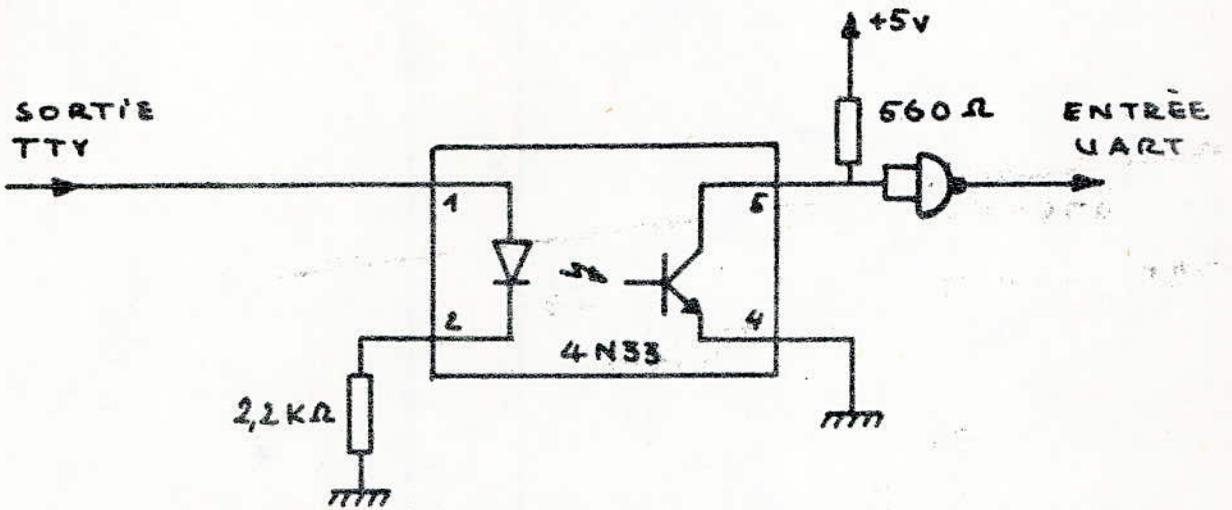
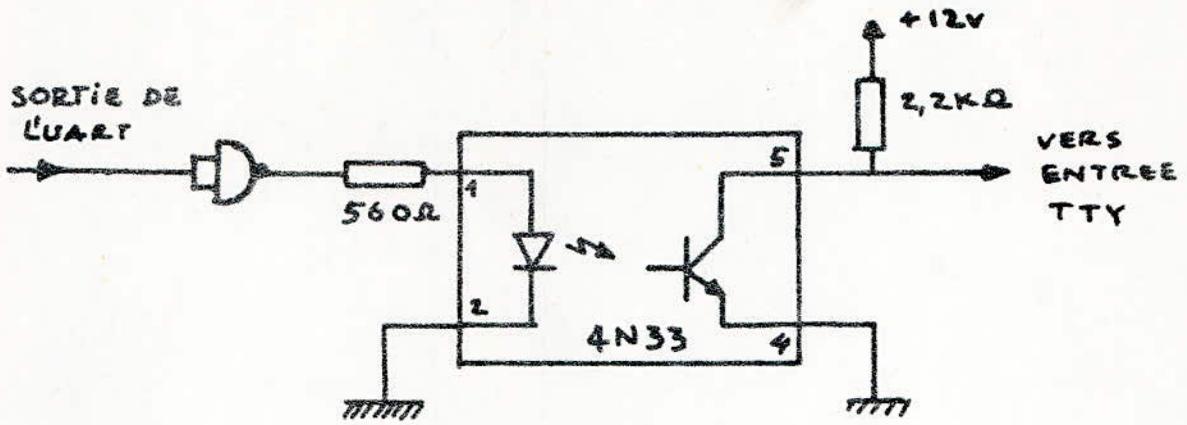
1

## OPERATION DE RECEPTION









MONTAGES D'ADAPTATIONS ENTRE UART  
 ET UNITE CENTRALE

ETUDE DU C.R.T.C.

Introduction.

Plusieurs fonctions caractérisent le circuit de visualisation de caractères alphanumériques sur écran cathodique. Ces fonctions sont les suivantes :

- Commande du mouvement du curseur
- Rafraichissement de l'écran
- Balayage des lignes de caractères
- Effacement de ligne
- Effacement de page
- Génération de signaux de synchronisation TV.

La réalisation d'un tel circuit nécessite un grand nombre de bascules, de portes et de composants électroniques, mais avec l'évolution de la technologie et les possibilités d'intégration, des circuits spécialisés réalisent toutes ces fonctions. Ces circuits sont appelés contrôleur d'écran ou CRTC.

Les principaux CRTC sont :

- MC 6845 de MOTOROLA
- INTEL 8275
- FAIRCHILD 9412
- SFF 9364 de THOMSON

On distingue deux principales méthodes de visualisation d'un texte sur écran cathodique, selon la gestion de celui-ci soit prise en charge par l'unité centrale ou par un processeur spécialisé. Le synoptique de ces deux configurations principales est représenté par la figure I.

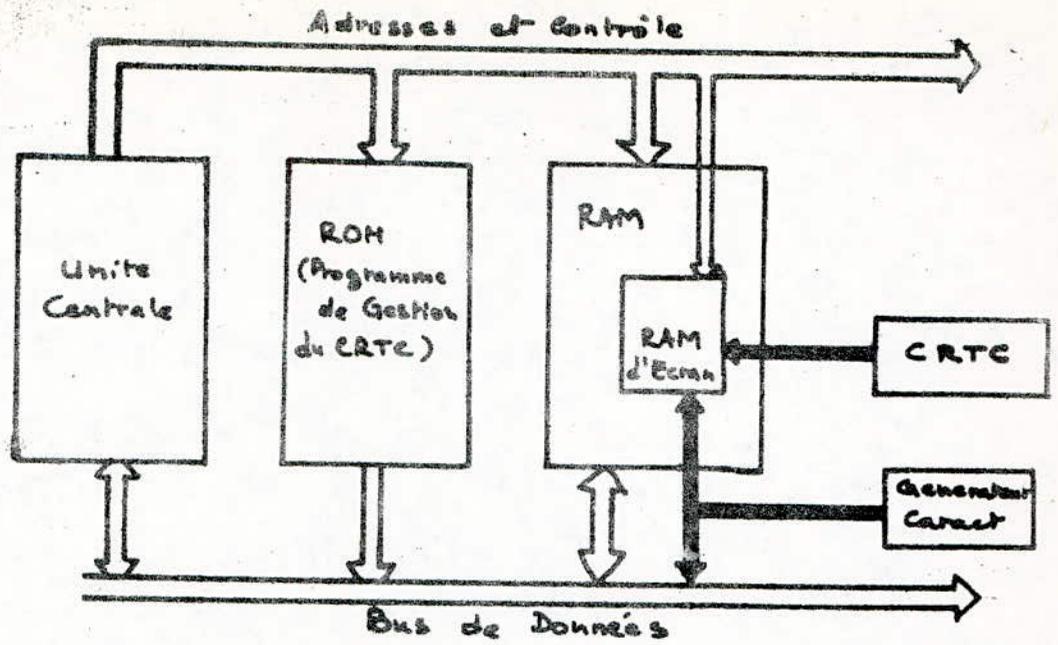


FIG. I.a

Gestion prise en charge par l'unité centrale

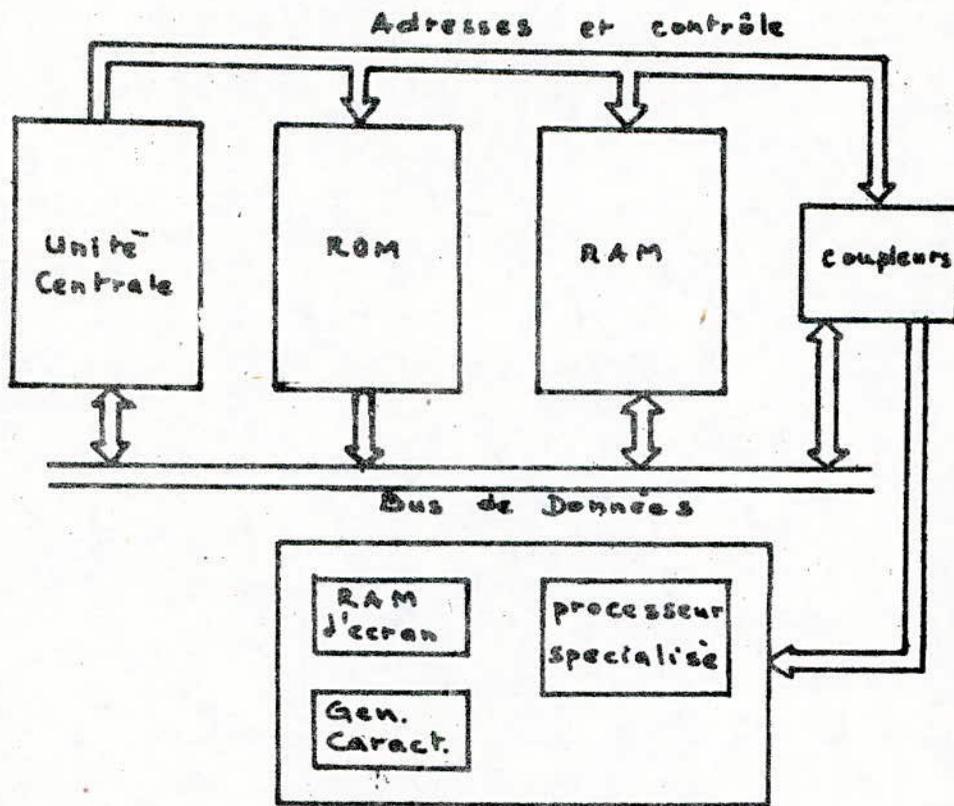


FIG. I.b

Gestion prise en charge par un processeur spécialisé

Dans le premier cas, la mémoire du texte fait partie de l'espace adressable du processeur.

Dans le second cas, on remarque l'existence d'un mémoire à laquelle l'unité n'a pas accès. L'intérêt principal de cette méthode réside dans le fait que l'unité centrale se trouve ainsi dégagée des tâches de gestion de la visualisation. Le tout est alors contrôlé par le contrôleur d'écran. La liaison terminal Video - Ordinateur se fera juste par un double fil pour l'échange d'information.

Nous avons énumérés précédemment les principaux CRTC. Nous avons utilisés le SFF 9364 car il est disponible et il exécute les multiples fonctions dont nous avons besoin.

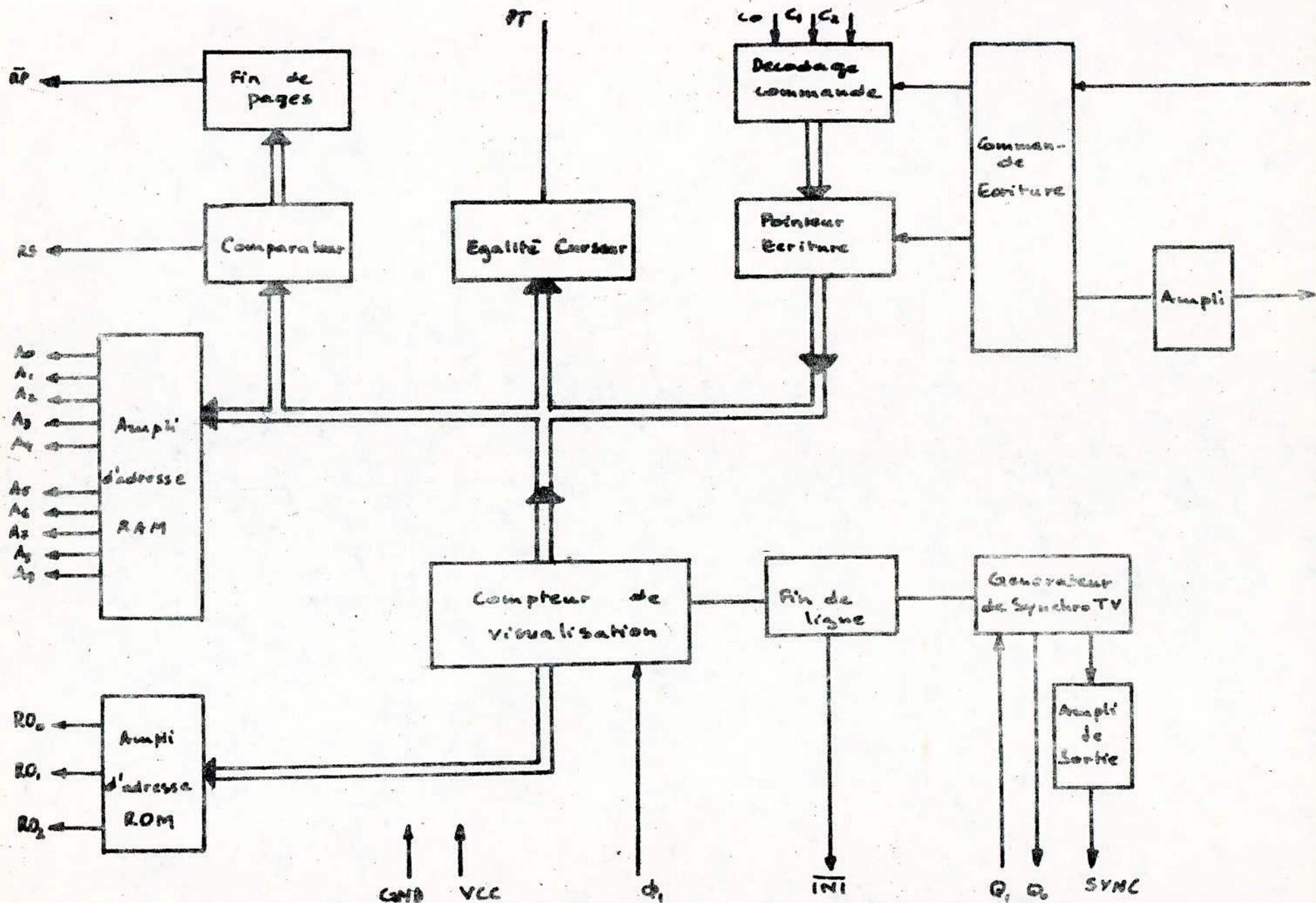
#### Caractéristiques du CRTC.

Le CRTC - SFF 9364 est un circuit réalisé en technologie MOS (canal N - grille au silicium), il est alimenté en 5V uniquement et il est compatible en LS - TTL.

L Les différentes fonctions du SFF 9364 sont :

- Génération de la synchronisation TV (standard CCIR 625 lignes)
- Rafraichissement de l'écran
- Gestion complète du curseur
- Balayage des lignes de caractère
- Commande de lecture et d'écriture
- Blanchissement de ligne ou de page
- Affichage de 1024 caractères
- Enchaînement de plusieurs pages.

./.



SYNOPTIQUE INTERNE DU PROCESEUR DE VISUALISATION

La figure 2 représente le synoptique interne du CRTC. Le coeur de ce circuit est le compteur de visualisation qui assure aussi les circuits de génération des signaux de synchronisation.

Ce circuit commande également un circuit de fin de ligne agissant sur la logique externe par la patte inhibition d'horloge ( $\overline{INI}$ ).

Les signaux de commande du curseur sont présentés aux entrées C0, C1 et C2 où ils sont décodés et agissant sur un pointeur d'écriture qui permet de charger dans la RAM ce qu'il fait effectuer la commande du curseur.

Un comparateur et un indicateur de fin de page sont inclus dans ce circuit pour permettre le contrôle de plusieurs pages d'écran.

Brochage et fonctions du CRTC.

- 1 et 2 / Q0 et Q1 : un quartz de 1,008 Mhz mis en parallèle avec une résistance de quelque mégadums sur ces bornes fournit la fréquence de base du générateur interne de synchronisation TV. Cette fréquence permet d'obtenir 50 trames par seconde.
- 3 / RS : indicateur de frontière de page, il permet de sélectionner la page lors de l'utilisation de plusieurs page.
- 4 à 8 / A5 à A9 : bits des poids les plus forts de la mémoire de rafraichissement.
- 9 / ~~Q1~~ : entrée de l'horloge de commande du circuit.

- 10 /  $\overline{\text{INI}}$  : ce signal permet à la fin de chaque ligne l'arrêt de l'orloge de commande du système.
- 11 à 13/ RO,R1,R2 : adresses des huit lignes des caractères alphanumériques mémorisés dans le générateur de caractères.
- 14 / G.N.D. : masse.
- 15 / P.T. : cette sortie génère le signal Video du circuit.
- 16 / S.T. : le front montant de ce signal autorise l'opération spécifiée par le code CO, C1 et C2.
- 17 / W : signal en phase avec les tops de synchronisation lignes autorisant l'écriture dans la mémoire de rafraichissement.
- 18 à 22/ AO - A4 : bits de poids faibles de la mémoire de rafraichissement
- 23, 24, 25/CO,C1,C2 : ces entrées précisent la nature du mouvement du curseur après apparition du front montant de ST.
- 26 / S.Y.N.C : sortie de la séquence de synchronisation TV compatible CCIR. Cette séquence contient les signaux de synchronisation ligne et les signaux de synchronisation trame.
- 27 /  $\overline{\text{RP}}$  : cette sortie permet d'incrémenter un compteur numéro de pages de texte doivent être enchaîner.

Les différentes combinaisons de C0, C1 et C2 donnant les mouvements du curseur sont :

F O N C T I O N S	C2	C1	C0	DUREE MS
- Effacement de page avec retour du curseur en haut à gauche	0	0	0	132
- Effacement de fin de ligne avec retour du curseur à gauche	0	0	1	8,3
- Déplacement du curseur d'une position vers le bas	0	1	0	8,3
- Inhibition du caractère envoyé	0	1	1	8,3
- Déplacement du curseur d'une position à gauche	1	0	0	8,3
- Effacement de la ligne courante du curseur	1	0	1	8,3
- Déplacement du curseur d'une position vers le haut	1	1	0	8,3
- Caractère normal	1	1	1	8,3

.\.

Les adresses A0-A9 selectionneront les positions caractère dans la mémoire de rafraichissement et sur l'ecran de TV.

A0 - A5 (64 combinaisons) selectionnent la position caractère d'une ligne sur l'ecran.

A6 - A9 (16 combinaisons) selectionnent la position ligne sur l'ecran.

Nous voyons donc que les fonctions utiles pour la gestion d'un circuit de visualisation peuvent être générées par le SFF 9364. Il suffit d'y associer une circuiterie adequate pour transformer tout poste de television en terminal video. Nous detaillerons ceci dans les prochains chapitres.

Deux horloges doivent piloter le contrôleur d'ecran. L'une très précise pour fournir les signaux de synchronisation, et l'autre moins précise pour piloter le circuit. La première sera réalisée par un quartz de 1,008 Mhz, et la seconde par un oscillateur, un trigger et un compteur-diviseur par huit.

Ce diviseur 74 I93 remet à zero ses quatres sorties dès qu'une impulsion positive est envoyée sur son entrée clear. Une impulsion négative sur l'entrée load permet de charger en sortie les données presentes aux entrées du compteur. Le signal d'horloge est injecté en count up. Le comptage commence et fournit une impulsion, tout les huit créneaux, qui chargera le registre à decalage et le registre tampon. Les figures 3.2a et 3.2b representent respectivement le schema interne et le diagramme des temps du compteur 74 I93.

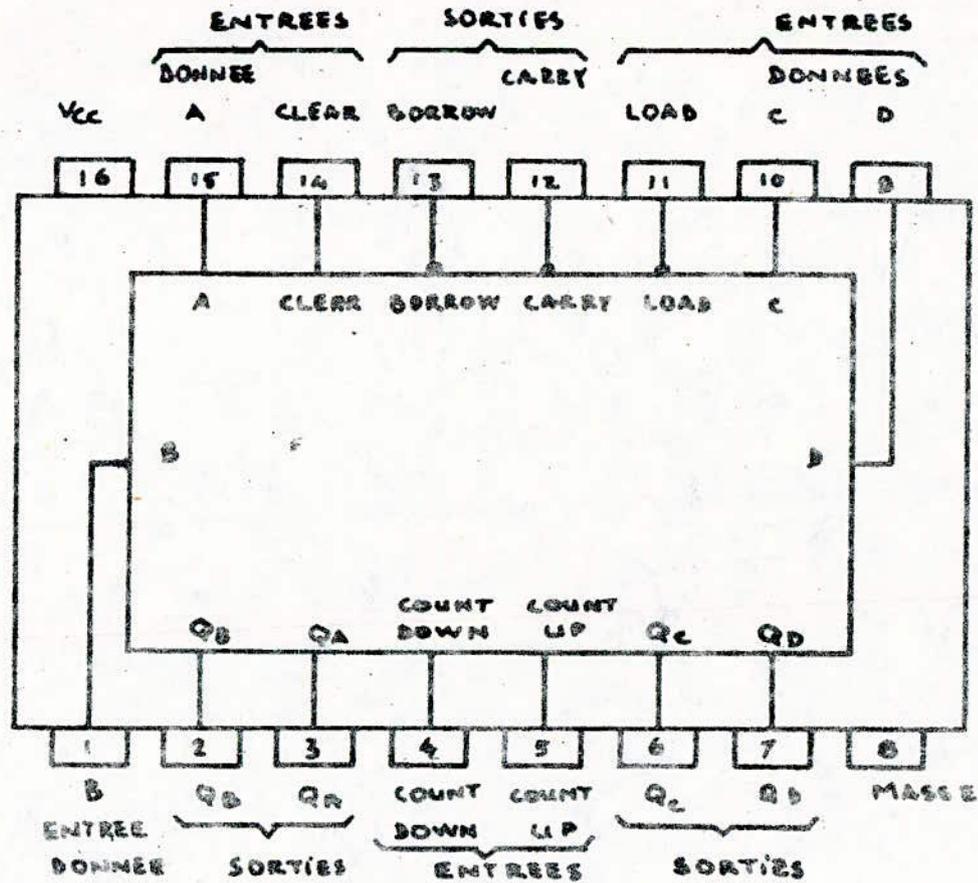


FIG 3.2a

SCHEMA INTERNE DU COMPTEUR 74193

AU NIVEAU BAS DE LOAD ON A :  $Q_A = A$        $Q_C = C$   
 $Q_B = B$        $Q_D = D$

DIAGRAMME DES TEMPS DU COMPTEUR 74 193

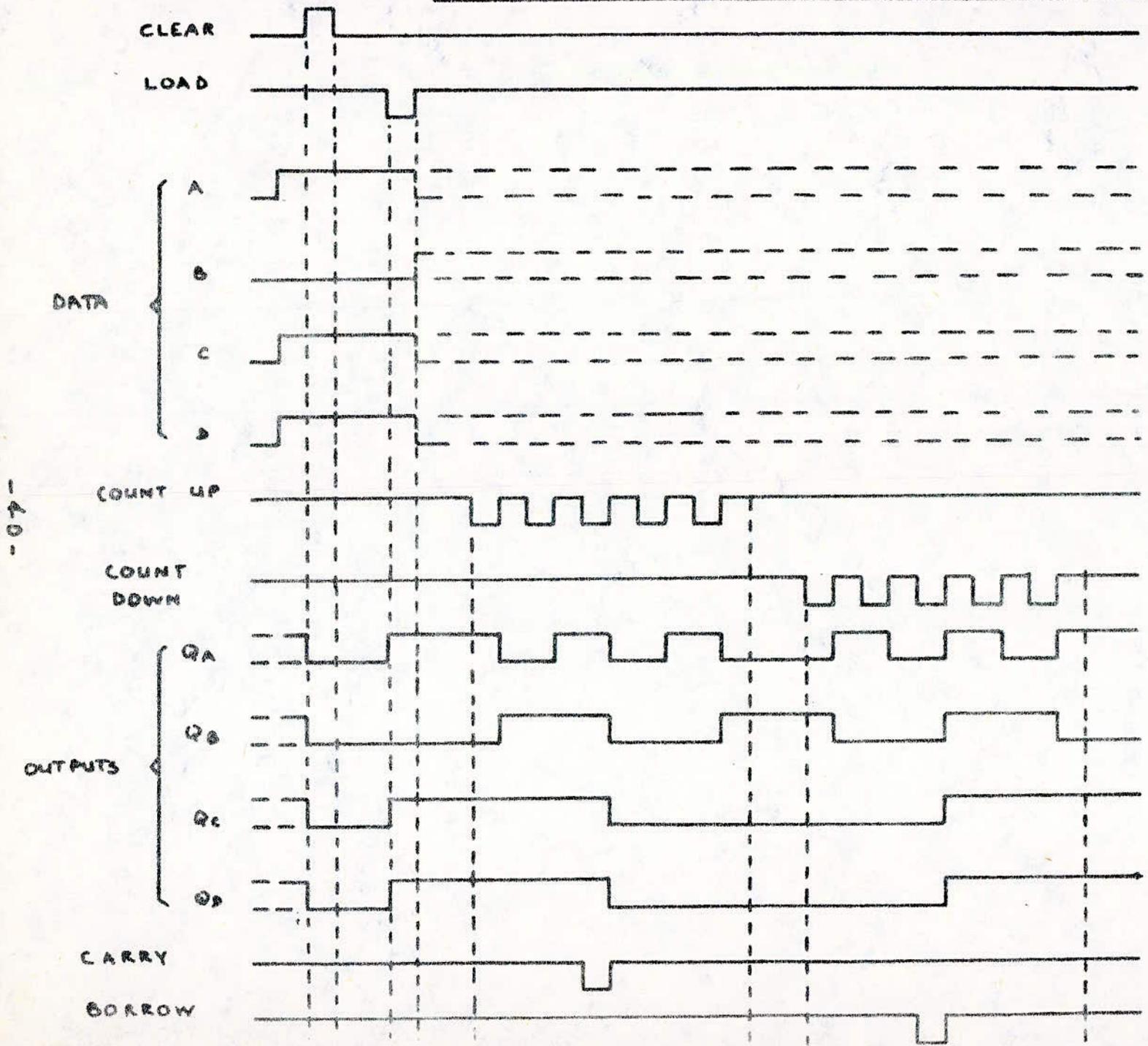
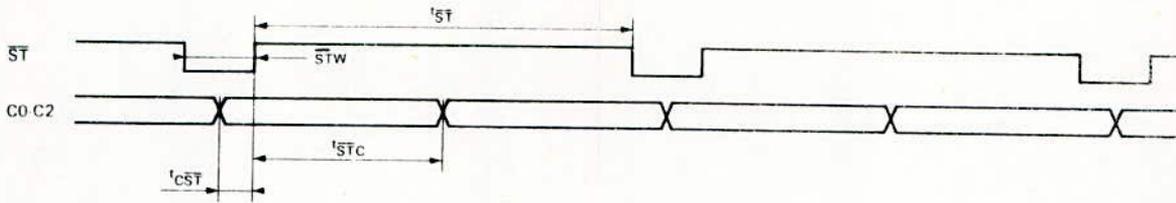


DIAGRAMME DES TEMPS DU COMPTEUR 74 193

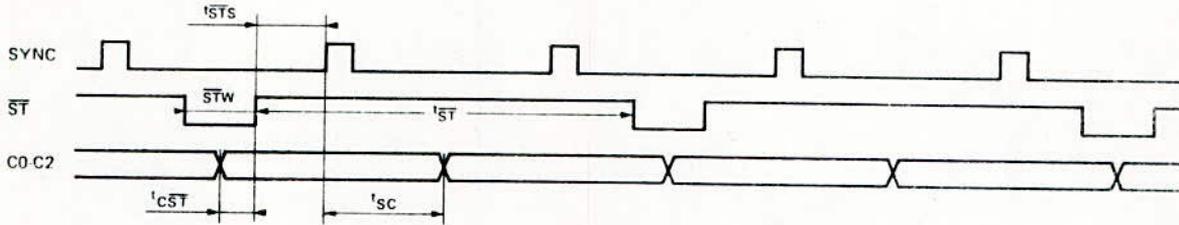
FIG. 3.2.6

## DATA INPUT TIMING

### Asynchronous operation



### Synchronous operation



#### Synchronous Operation

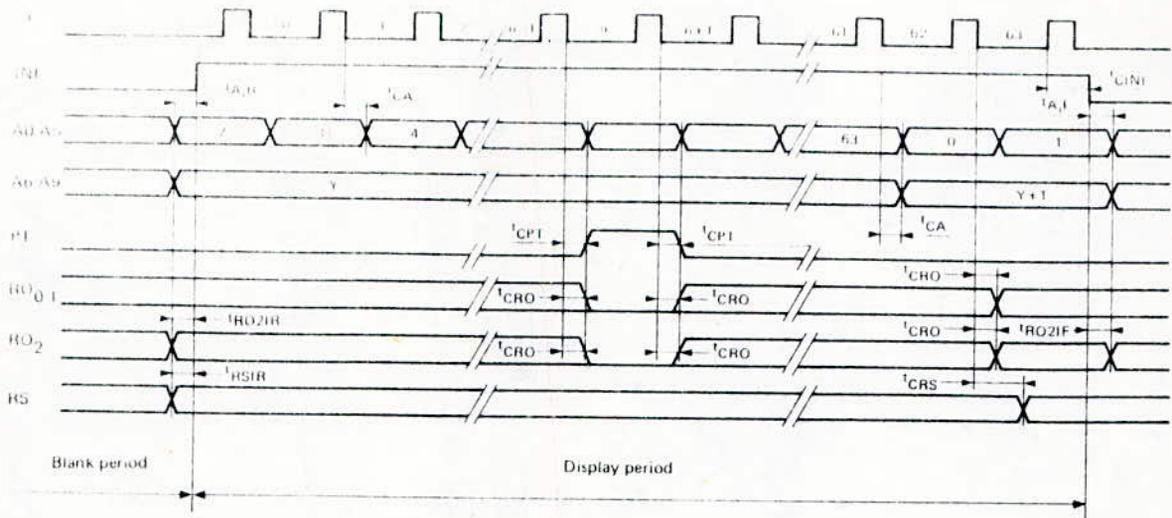
Parameter	Symbol	Min	Typ	Max	Unit
$\overline{ST}$ Pulse Width	$\overline{STW}$	0.5	-	-	$\mu s$
C0-C2 Set Up Time (from $\overline{ST}$ )	$t_{cST}$	1	-	-	$\mu s$
C0-C2 Hold time (from SYNC)	$t_{sc}$	16	-	-	$\mu s$
$\overline{ST}$ Set Up Time (from SYNC)	$t_{STS}$	1	-	-	$\mu s$

#### Minimum Strobe Period $t_{ST}$ (Operation execution time)

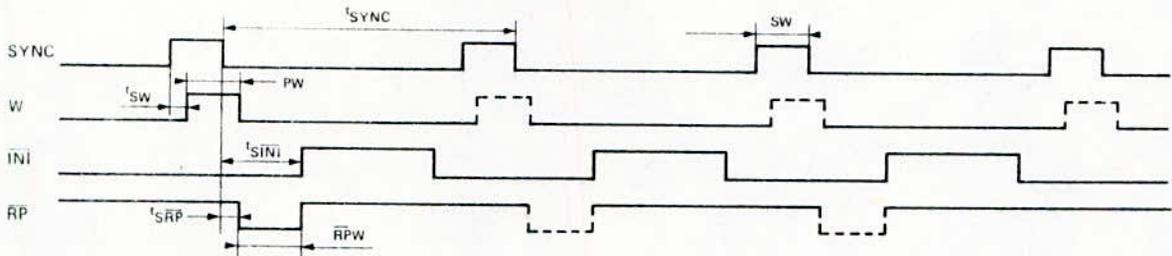
FUNCTION	CONTROL INPUTS			Max Execution time	Unit
	C2	C1	C0		
Page Erase & Cursor Home	0	0	0	132	ms
Erase to End of Line & Return Cursor	0	0	1	4.2	ms
Line Feed (Cursor Down)	0	1	0	80*	$\mu s$
No Operation	0	1	1	80	$\mu s$
Cursor Left	1	0	0	80	$\mu s$
Erasure of Cursor Line	1	0	1	8.3	ms
Cursor Up	1	1	0	80	$\mu s$
Normal Character	1	1	1	80	$\mu s$

\*Will increase to 8.3 ms when text scroll occurs. See «Scrolling» for conditions.

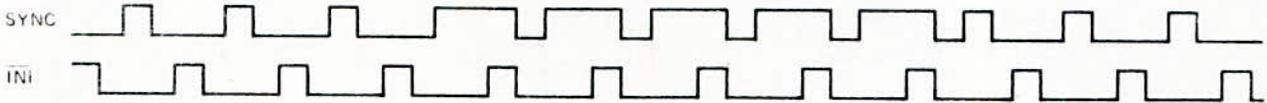
## LINE TIMING



## SYNC TIMING



## FRAME TIMING



## DATA INPUT TIMING

### Asynchronous Operation

Parameter	Symbol	Min	Typ	Max	Unit
ST Pulse Width	$\overline{STW}$	0.5			$\mu s$
C0-C2 Set Up Time	$t_{CST}$	1	-	-	$\mu s$
C0-C2 Hold Time	$t_{STC}$	78	-	-	$\mu s$

### Minimum Strobe period $t_{ST}$ (Operation execution time)

FUNCTION	CONTROL INPUTS			Max Execution Time	Unit
	C2	C1	C0		
Page Erase & Cursor Home	0	0	0	132	ms
Erase to End of Line & Return Cursor	0	0	1	4.2	ms
Line Feed (Cursor Down)	0	1	0	130*	$\mu s$
No Operation	0	1	1	80	$\mu s$
Cursor Left	1	0	0	80	$\mu s$
Erasure of Cursor Line	1	0	1	8.3	ms
Cursor Up	1	1	0	80	$\mu s$
Normal Character	1	1	1	130*	$\mu s$

\* Will increase to 8.3 ms when text scroll occurs. See «Scrolling» for conditions.

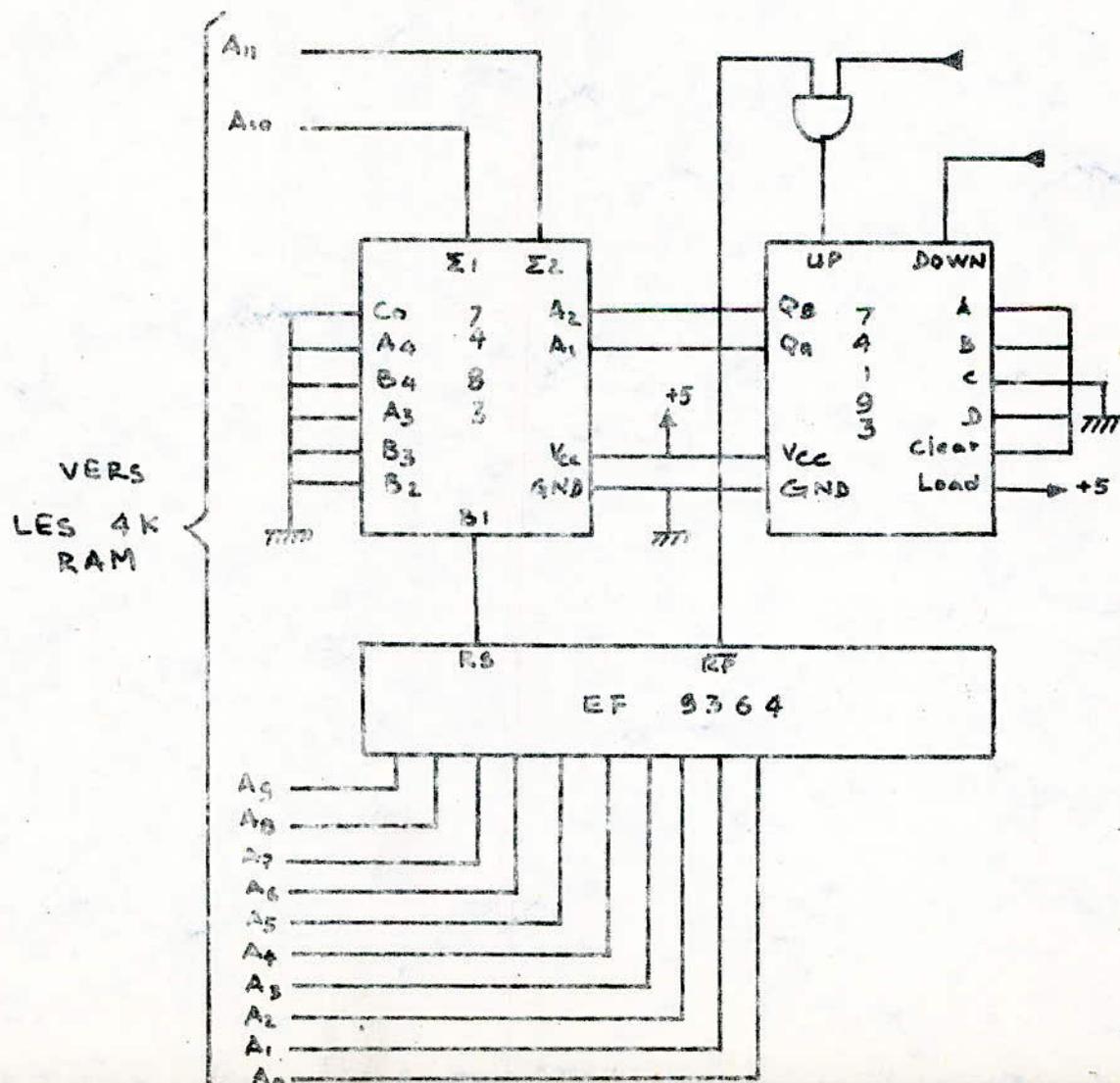
## Possibilité d'enchaînement de pages

Lorsque une page de texte n'est pas suffisante pour l'opérateur, il est possible en utilisant une mémoire de rafraîchissement de capacité plus grande de mémoriser plusieurs pages d'écran et d'en faire l'enchaînement.

Le CRTC peut gérer plusieurs pages grâce aux broches  $\overline{RP}$  et RS

La figure 3.3 nous donne le schéma de montage pour l'enchaînement de 4 pages. Il utilise un compteur 74 193 qui incremente et decremente les pages, et un additionneur qui ajoutera 2 lignes adresses pour la mémoire de rafraîchissement. Les 4 combinaisons de ces 2 lignes adresses augmentent la capacité de la mémoire de rafraîchissement de 1K 8bits à 4K 8bits.

FIG 3.3



LE DECODAGEIntroduction

Comme nous l'avons indiqué précédemment, l'enfoncement de chaque touche sur le clavier émet une information sous forme parallèle 7 bits. L'UART ne fait que transmettre cette information, avec vérification, et à la vitesse désirée. Cette information étant codée en ASCII, il va falloir la decoder pour savoir ce qu'elle représente:

- Un caractère à visualiser
- Un mouvement du curseur
- Une information à ignorer

Ce decodage se fera avec une mémoire EPROM MCM2708.

Principe du décodage

La gestion du curseur et l'inhibition d'un caractère étant contrôlées par le contrôleur d'écran CRTIC sur ses 3 entrées CO, CI, C2, il est donc nécessaire de différencier entre:

- Un caractère à visualiser que l'on enverra directement à la visualisation
- Un mouvement du curseur ou une inhibition que l'on enverra au contrôleur d'écran.

On utilisera un bit en plus pour autoriser l'écriture dans le premier cas, et l'interdire dans le second cas.

L'inhibition d'un caractère se fera lorsque l'information envoyée ne peut être ni visualiser, ni contrôler par le CRTIC comme mouvement du curseur. Le CRTIC se chargera alors de supprimer cette information.

Nous aurons donc besoin d'une mémoire à sept lignes adresses et quatre lignes données pour coder les 128 caractères du code ASCII. Les trois premières iront directement aux entrées CO, CI, C2 du CRTIC et la quatrième autorisera ou

non l'écriture par l'intermédiaire d'une NAND avec le signal ~~signal~~ write du CRIC.

- Oo ----Co
- OI ----CI
- O2 ----C2
- O3 ----(=I écriture autorisée)
- (=0 écriture interdite)

Les différentes combinaisons de CO,CI,C2 sont:

C2	CI	CO	
0	0	0	
0	0	I	Mouvement du curseur
0	I	0	
0	I	I	Inhibition du caractère envoyé
I	0	0	
I	0	I	Mouvement du curseur
I	I	0	
I	I	I	Caractère normal

Pour une écriture ,on doit avoir à la sortie de la mémoire la combinaison IIII.

Dans les deux premières colonnes du tableau ASCII (elles correspondent à la gestion de la visualisation et à la communication avec une unité centrale),seuls quelques codes seront utilisés,ce seront:BS,UPS(HT),LF,LUP(VT),Clear(FF),Clear(RC),SUB,n(Fs),RC(GS).Ceux-ci ayant des fonctions bien définies, à chacun on fera correspondre le code adéquat pour la gestion (pour-le) du curseur.Le codage se fait sur trois bits,le quatrième bit O3 aura pour rôle de différencier entre un mouvement du curseur sans changement du texte visualisé (O3=0) et un mouvement du curseur avec effacement (O3=I).

Pour les codes à ignorer,on mettra dans la case mémoire correspondante du décodeur le code OII qui représente l'inhibition.

Nous utiliserons une mémoire ROM 2708 de 1K 8BITS,c'est-à-dire 1024 cases mémoire de 8 bits.Pour notre application nous avons besoin de 128 cases de 4 bits seulement.Pour cela les

3 entrées de poids le plus fort de la mémoire ~~ni~~ seront mises à la masse. Il restera alors 128 cases disponibles. En sortie nous utiliserons seulement les 4 sorties de poids faible.

### Programmation

La programmation se fait en hexadécimal pour l'adressage et les données, il est nécessaire de faire une conversion binaire-hexadécimal de tous les codes.

La programmation est donnée en page suivante.

### Technique de programmation

Nous jugeons juste de donner un aperçu sur la technique de programmation d'une mémoire EPROM. Ceci n'étant pas le but de notre travail.

La programmation d'EPROM se fait sur système monté sur microprocesseur MOTOROLA 6800 comprenant:

- un logiciel de chargement de programme PROMPHI
- une carte programmeur d'EPROM

Le programme destiné à être chargé sur EPROM est introduit en hexadécimal, byte par byte à l'aide d'une télécrite. PROMPHI gère lui-même l'organisation de la zone mémoire de transition (mémoire RAM). A la fin du chargement, il édite le contenu des RAM afin que l'opérateur puisse le vérifier. Il effectue ensuite le transfert dans l'EPROM avec les contrôles d'usage. Cette dernière est, au préalable, montée sur un support destiné à cet effet sur la carte programmeur d'EPROM. Son adresse dans le système est connue du logiciel PROMPHI. A la fin de l'opération de transfert, le PROMPHI effectue un ultime contrôle, en comparant le contenu des RAM et celui de l'EPROM.

PROGRAMMATION DE MEMOIRE EPROM MCM 2708

Adresses		Touche	Données				Hexa	Fonction
Decimal	Hexa		binaires					
			D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>		
0 à 7	000 à 007	à ignorer	0	0	1	1	0B	
8	008	BS	0	1	0	0	04	← 1 position
9	009	HT (UPS)	0	1	1	1	07	→ 1 position
10	00A	LF	0	0	1	0	02	↓ 1 position
11	00B	VT (LUP)	0	1	1	0	06	↑ 1 position
12	00C	FF (Clear)	1	0	0	0	08	Effacement ↙ de page
13	00D	RC (clear)	1	0	0	1	09	Effacement ligne avec ←
14 à 25	00E à 019	à ignorer	0	0	1	1	0B	
26	01A	SUB	1	1	0	1	0D	Effacement ligne
27	01B	à ignorer	0	0	1	1	0B	
28	01C	FS (H)	0	0	0	0	00	Retour en haut à gauche ↖
29	01D	(RC) GS	0	0	0	1	01	← En début de ligne
30 à 31	01E à 01F	à ignorer	0	0	1	1	0B	
32 à 126	020 à 07E	Caractère	1	1	1	1	0F	Visualisation des caractères
127	07F	à ignorer	0	0	1	1	0B	
128 à 1023	080 à 3FF	inutilisées						

Rappel sur la Télévision

Etude du Standart 625 lignes

Le récepteur de Télévision représente le dernier bloc de notre système de visualisation. Il est donc nécessaire de rappeler brièvement le principe d'analyse et de transmission d'une image de Télévision.

Analyse et transmission de l'image.

L'image que l'on reçoit sur l'écran du tube cathodique n'est en réalité rien d'autre qu'un signal constitué d'une infinité de points élémentaires. Ces points sont plus ou moins lumineux selon l'information qu'ils transportent.

Le mode de reproduction de l'image est une analyse séquentielle de tout les points de l'image afin de les transmettre dans un ordre bien déterminé au récepteur de Télévision. Ce dernier les reproduira (dans leur ordre) avec une brillance proportionnelle à celle du point examiné à l'émission.

Cela revient donc à reproduire exactement à la réception chaque point de l'image dans sa position dans le plan, et ses variations de luminosité dans le temps.

Pour cela on a divisé l'image en un certain nombre de lignes (Standart CCIR 625 lignes).

On analysera ainsi les points d'une ligne successivement, ensuite on passera à la ligne suivante et cela jusqu'à l'exploration de l'image totale.

L'analyse ligne par ligne a certains inconvénients tel que le scintillement. Pour cela l'analyse interligne a été adoptée (avec un nombre impair de ligne).

### Principe du balayage entre lacé.

Ce mode consiste à balayer d'abord les lignes impaires et ensuite les lignes paires de l'image. Ces deux demi images seront appelées trame et au lieu d'avoir 25 images/seconde on aura 50 trames/seconde la figure 2 représente ce genre de balayage sur un exemple de 7 lignes. Le Spot balayera la 1<sup>er</sup>., la 3<sup>o</sup>, la 5<sup>o</sup> et juste la moitié de la 7<sup>o</sup> ligne. Il sera ramené instantanément en haut de l'image. Il balayera cette fois-ci la deuxième moitié de la 7<sup>o</sup> ligne et les 2<sup>o</sup>, 4<sup>o</sup> et 6<sup>o</sup> ligne. Arrivé en 6, il repartira automatiquement vers 1. Ainsi on passera à la prochaine image pour refaire la même chose.

Mais pour que le Spot puisse gérer facilement le signal, il faut synchroniser les lignes et les trames. Pour cela, à l'émission on doit recevoir des signaux qui nous indiqueront la fin de chaque ligne et la fin de chaque trame. Ces signaux de commande s'appellent signaux de synchronisation. Ils seront mélangés au signal Video. On obtiendra alors le signal complet appelé signal Video composite.

### Fréquence ligne et fréquence trame.

Ce sont des fréquences bien déterminées, obtenues à partir d'un oscillateur unique à l'émission. Ces fréquences sont obtenues en utilisant des diviseurs de fréquence.

Dans le Standard 625 lignes, l'oscillateur pilote fonctionne sur 31 250 Hz. Un diviseur par 2 nous donne la fréquence ligne soit  $31\ 250/2 = 15\ 625$  Hz quatre diviseurs par 5 nous donne la fréquence trame soit  $31\ 250/5 \times 5 \times 5 \times 5 = 50$  Hz, qui n'est rien d'autre que la fréquence du secteur.

### Etude du signal Video composite.

Le signal Video composite est comme nous l'avons dit précédemment ; la superposition du signal Video et des signaux de synchronisation.

./.

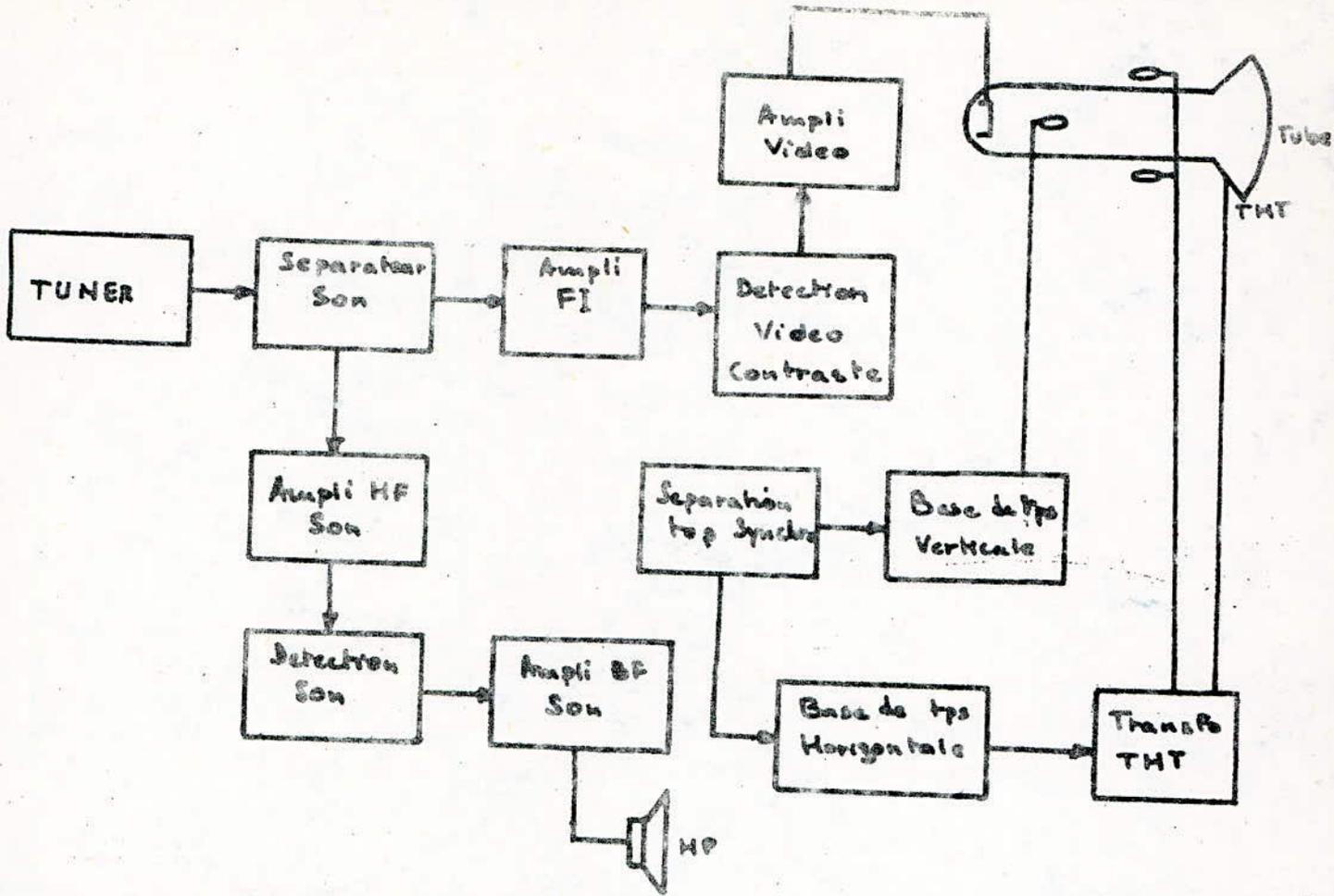


FIG1 SYNOPTIQUE D'UN RECEPTEUR DE TELEVISION NOIR ET BLANC

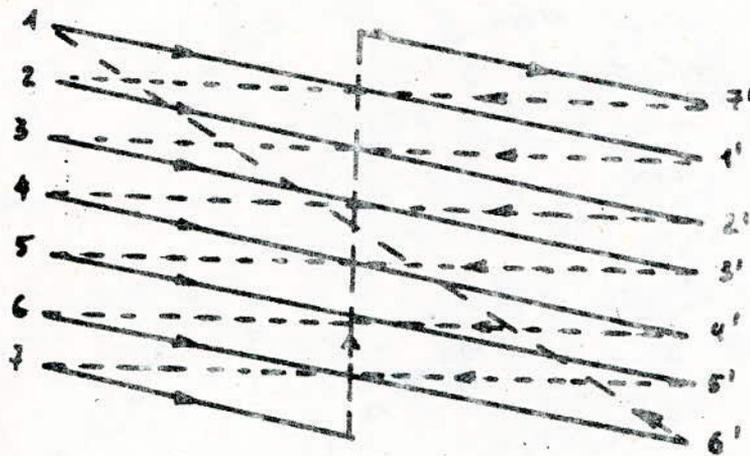


FIG2 PRINCIPE DU BALAYAGE ENTRELACE

Le signal Video est l'ensemble de points à analyser en fonction de leurs positions dans le plan (qui correspondrait à une certaine ligne) et de leurs luminances (qui correspondrait à l'amplitude de la tension de chaque point). Ce signal constitue l'information à visualiser sur l'écran de Télévision. La figure 3 représente l'allure générale du signal Video composite.

A la fin de chaque ligne et de chaque trame, le Spot lumineux doit faire un retour pour balayer la ligne suivante. Il faut donc assurer un blocage du tube cathodique afin que les traces de retours ne soient pas visibles. Le niveau de suppression de l'image pendant les retours lignes et trames est situé légèrement au dessous du niveau du noir (dans l'ultra noir). Les impulsions de synchronisation seront émises pendant la durée de cet effacement.

Les signaux de synchronisation.

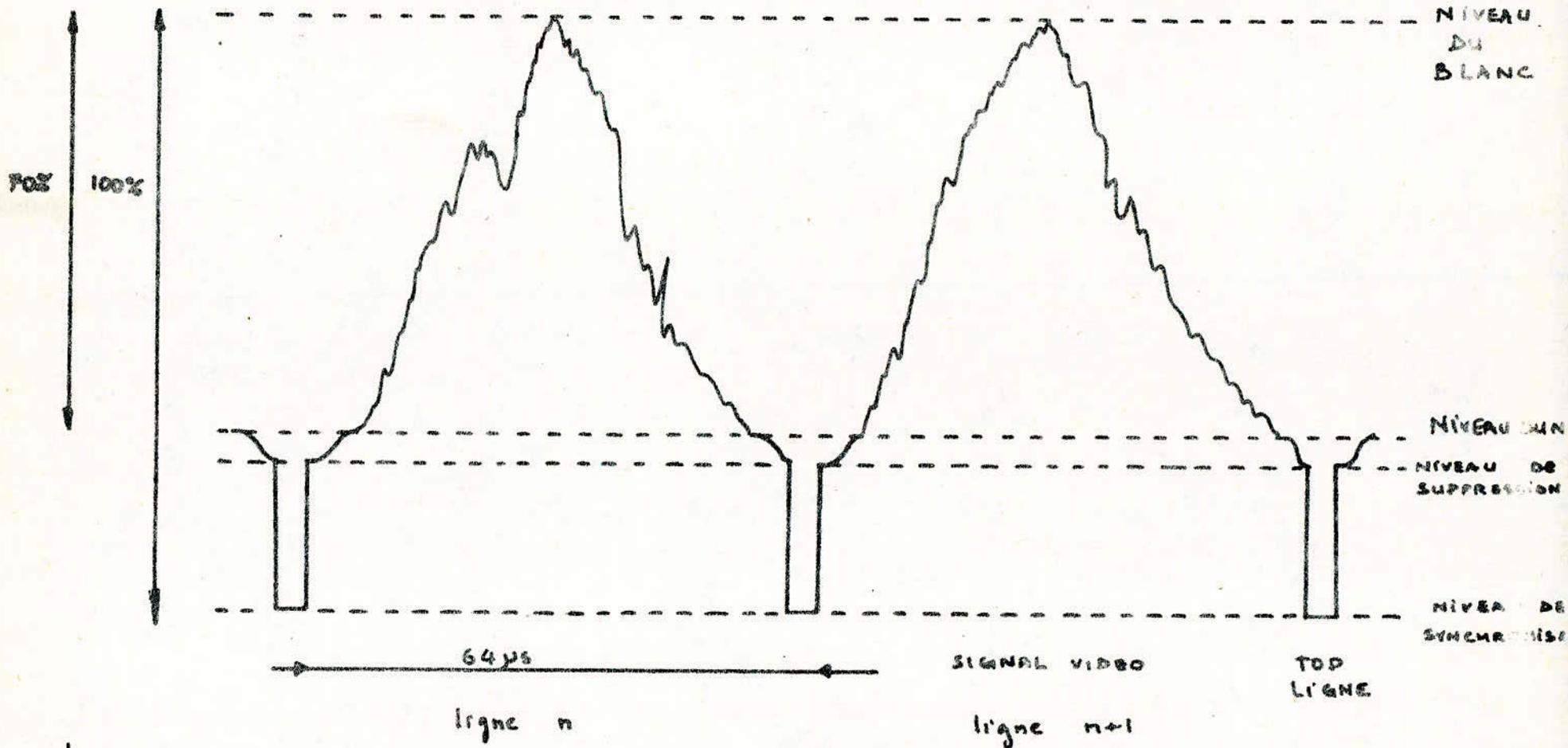
Les figures 4 et 5 représentent simultanément les tops de synchronisation lignes et trames.

La durée des tops lignes est de 4,7 us. Ils sont émis tout les 64 us (durée d'une ligne).

La durée des tops trame est plus grande, c'est ce qui permet de les différencier. Ces derniers se décomposent en 3 ensembles.:

- une série de 5 impulsions de pre-égalisations de période H/2 soit 32 us.
- une série de 5 impulsions de synchronisations trames répartie sur 160 us.
- une série de 5 impulsions de post-égalisations de période H/2 aussi.

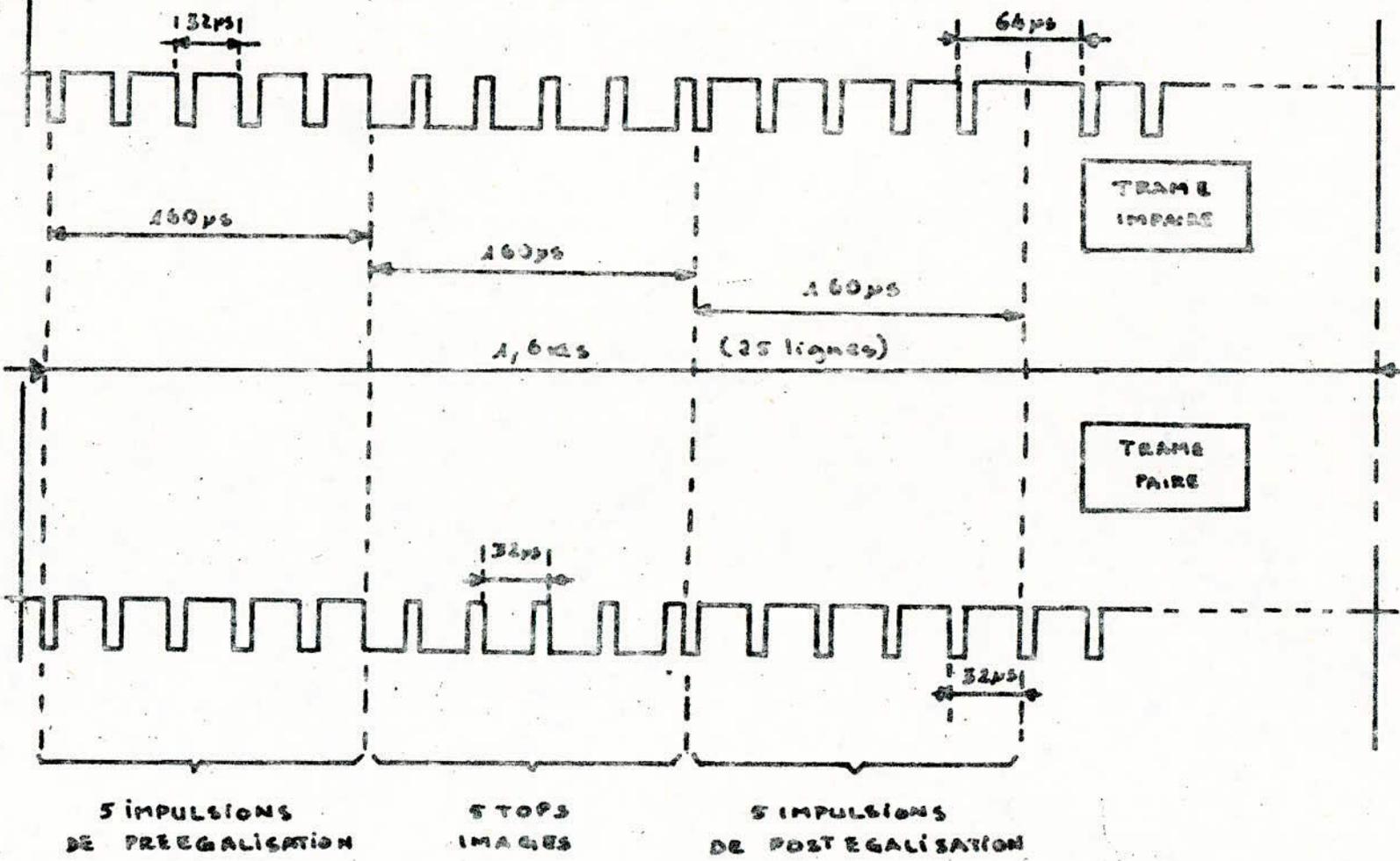
La durée du signal de suppression est de 25 H soit 1,6 ms.



-05-

FIG3 ALLURE GENERALE DU SIGNAL VIDEO COMPOSITE

-51-



FIGS SIGNAUX DE SYNCHRONISATION TRAME (STANDARD CCIR 625 LIGNES)

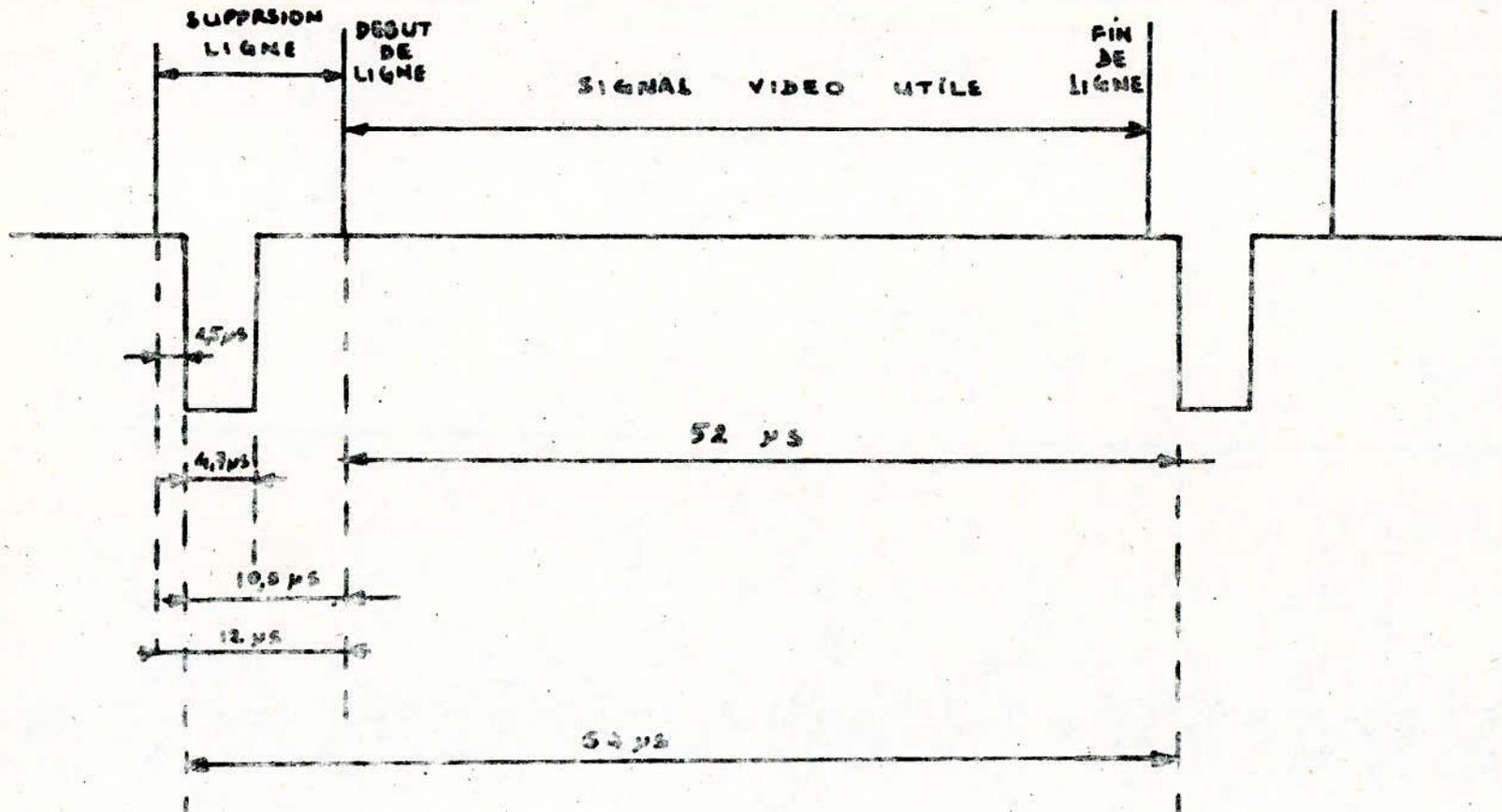


Fig. 1 SIGNAL DE SYNCHRONISATION LIGNE (STANDARD 625 LIGNES) CCIR

Les trames paires et impaires sont différenciées par le premier top de synchronisation ligne qui suit les tops de post-égalisation. Si ce top est à 64 us du dernier top de post-égalisation. C'est une trame impaire et s'il est à 32 us c'est une trame paire.

Bande passante du signal Video.

Pour calculer cette fréquence il faut se placer dans le cas où l'image est constituée par une suite de bandes verticales alternativement noires et blanches. Donc connaître le nombre de points par ligne.

Sachant que la largeur de l'image est égale au 4/3 de la hauteur de l'image est en supposant que la hauteur d'un point est égale à la hauteur de l'image divisée par le nombre de lignes. On aura le nombre de points élémentaires par ligne égale à :

$$N = \frac{625 \times 4}{3} = 830 \text{ points}$$

Pour une suite de bandes verticales noires et blanches la fréquence maximale du signal est :

$$F = \frac{830}{2} \times 625 \times 25 = 6,5 \text{ MHz}$$

Le contrôleur d'écran CRTIC est compatible avec le standard CCIR 625 lignes. Nous aurons donc aucun problème du point de vue synchronisation et bande passante.

CIRCUIT DE RAFRAICHISSEMENT

Dans le chapitre précédent nous avons vu que l'information Video contient constamment l'information de façon à avoir une image sur l'écran. Si l'information envoyée n'est pas entretenue ; elle sera fugitive. L'opérateur ne la verra que pendant quelque dixièmes de seconde grâce à la persistance rétinienne. Pour remédier à cela on constitue un bloc mémoire qui emmagasinera les codes des 1024 caractères affichables sur l'écran (64 caractères x 16 lignes) pour les revisualiser.

Mémoire de rafraichissement.

Chaque caractère est codé sur 7 bits, comme nous avons 1024 caractères à stocker ; il faudrait que le bloc mémoire soit constitué de mémoire 1K 7 bits. Nous utiliserons des mémoires de type RAM car leur contenu peut changer d'une page d'écran à une autre. Selon les mémoires disponibles on peut utiliser :

- 1 RAM de 1k 8 bits
- 2 RAM de 1k 4 bits
- 7 RAM de 1k 1 bits

Rappelons que les RAM sont des mémoires permettant l'écriture, la lecture ou l'effacement d'une information binaire.

Les RAM ou mémoires à accès aléatoire sont ainsi nommées parcequ'il est possible à n'importe quel instant, de noter une information, soit au contraire d'en effectuer la lecture.

Il existe 2 types de mémoires RAM :

- Les RAM Statiques
- Les RAM Dynamiques

Dans les RAM statiques ; une information notée y reste jusqu'à son effacement. Tandis que dans les RAM dynamiques ; l'information ne reste pas constamment stockée. Elle disparaît au bout d'un certain temps. Pour cela le besoin est de rafraîchir constamment ce genre de mémoire c'est à dire de réinscrire l'information périodiquement.

Nous avons opté pour le 1er type de mémoire parce qu'il n'y a pas d'impératifs sur le temps d'accès et qu'elle ne nécessite pas de rafraîchissement.

Pour noter cas nous utiliserons 2 mémoires RAM de 1k 4 bits des 2114. Celles-ci organisées de la façon suivante :

- . des lignes adresses (A0 - A9) pour sélectionner la case mémoire à lire ou à écrire.
- . une entrée de sélection  $\bar{W}$  (à l'état bas elle permet l'écriture et à l'état haut la lecture).
- . des lignes données bidirectionnelles (elles servent d'entrées et de sorties).

$\bar{W} = 0$  ce sont des entrées pour permettre l'écriture

$\bar{W} = 1$  ce sont des sorties pour permettre la lecture

L'assemblage des 2 mémoires se fait en reliant leurs lignes adresses respectives. Elles constituent ainsi un bloc mémoire d'une même capacité du point de vue adresses, mais de capacité double du point de vue données qu'une RAM élémentaire 2114.

Le texte sur l'écran est rafraîchi de mémoire continue (toutes les 20 ms) par la lecture des 1024 positions mémoires quelque soit leurs contenues.

Un curseur généré par le contrôleur d'écran indique la première position mémoire libre où viendrait s'inscrire le caractère tapé sur le clavier. La position de ce curseur est indiquée sur l'écran par 5 points clignotants.

L'envoi d'un ordre FF permet d'effacer l'ecran en mettant à zéro toutes les positions mémoires. Ceci s'effectue par l'inhibition des portes AND (7408) par la sortie R02 pendant le balayage des adresses A0 - A9 et en présence du signal  $\bar{W}$ .

Le circuit tampon ou "Latch".

Entre le bloc de rafraichissement et générateur de caractère est interposé un circuit tampon. Ce circuit tampon est constitué par 8 bascules D à verrouillage. La table de vérité d'une bascule D est donnée ci-dessous.

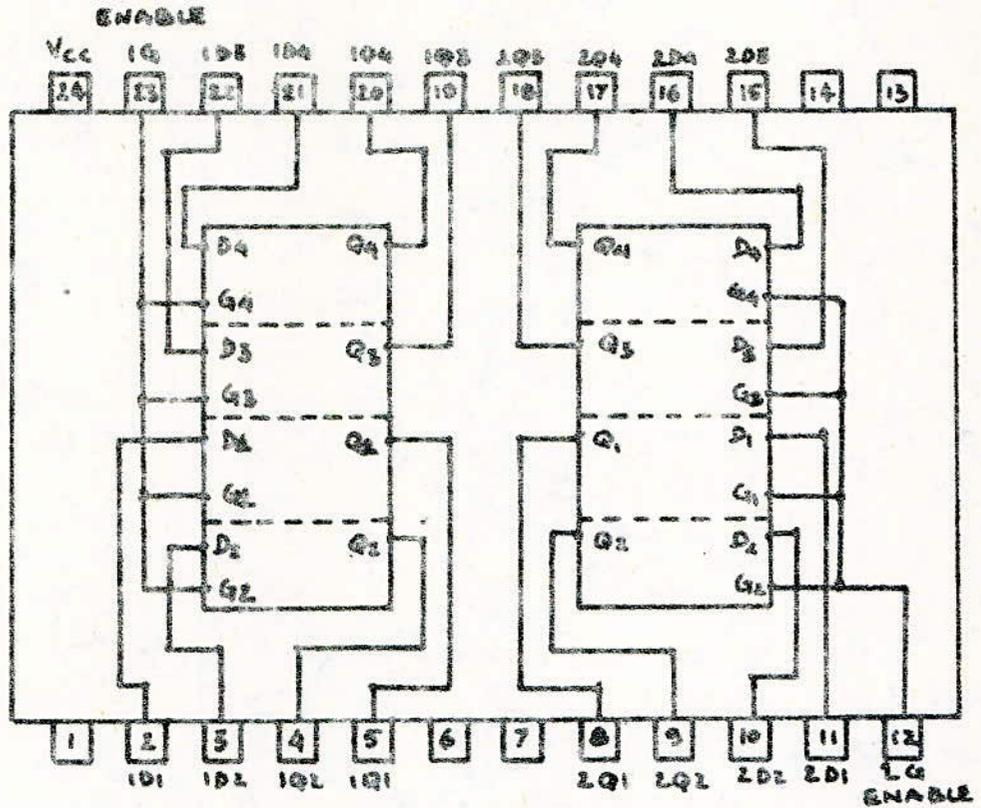
T	T + 1
D	Q
0	0
1	1

$$D = Q$$

La sortie Q est égale à l'entrée D grand vu top d'horloge est envoyé.

A l'état haut, tous les états à l'entrée D sont transmis à la sortie Q. Quand l'horloge retourne à l'état bas, elle verrouille la sortie sur le dernier état enregistré.

Nous avons utilisé le circuit SN 74100 qui contient 8 bascules D et dont la figure 2 représente le schéma interne et la table de vérité. Ces 8 bascules sont groupés par 4. Les entrées horloge 23 et 12 seront alors reliées ensemble.



Entrées		Sorties	
D	G	Q	$\bar{Q}$
0	1	0	1
1	1	1	0
$\beta$	0	$Q_0$	$\bar{Q}_0$

$Q_0$  : niveau de Q  
avant le verrouillage

Fig2

TABLE DE VERITE DU LATCH 74100

Les informations contenues dans les mémoires de rafraichissement sortent de celles-ci dès qu'un signal  $\bar{W} = 1$  est envoyé. Elles se présentent aux entrées des bascules D. Ces bascules présente à leurs sorties Q les informations qui se trouvent aux entrées D dès que l'on envoie un niveau haut sur l'entrée horloge. Lorsque le niveau devient bas, les sorties se verrouillent et gardent l'état pris lors de la commande de verrouillage.

L'emploi de ce circuit tampon est nécessaire pour maintenir l'information permanente en sortie du bloc mémoire pendant que l'adressage (A0 - A9) des mémoires change.

LE GENERATEUR DE CARACTERES

Dans le chapitre I, nous avons vu comment le clavier codait l'information en vue d'un traitement ou d'une visualisation. Dans ce chapitre, ce sera l'inverse que l'on étudiera, c'est à dire comment restituer le caractère à partir de son code ASC II, ou plus exactement comment générer les signaux nécessaires à la représentation du caractère sur l'écran cathodique.

Principe

Sur le récepteur TV, l'analyse de l'image se fait point par point. La génération d'un caractère à visualiser doit être basé sur le même principe. Pour cela il faut dessiner chaque lettre, chaque chiffre et chaque signe dans des matrices. Ces matrices représenteront dorénavant les caractères. Pour visualiser les caractères, il faudrait visualiser leurs matrices.

Pour une image quelconque sur un récepteur TV, la brillance de chaque point est significative. Dans le cas de la visualisation de caractère, la brillance doit être soit Maximum, soit Minimum. Pour un point de l'écran, il est soit noir, soit blanc. C'est le principe de tout ou rien.

Dimension du caractère

2 types de matrices sont généralement utilisés :

- . Matrice 7 X 9
- . Matrice 5 X 7

./.

Il est évident que le premier type donne une meilleure définition du caractère (surtout pour les minuscules) mais demande plus de place, affiche donc moins d'information sur l'écran.

Le choix du second type nous a été imposé par le contrôleur d'écran. Celui-ci a été conçu pour visualiser des caractères sous forme matrice 5 X 7. Le CRTC balaie à ses sorties R0, R1 et R2 les huit lignes des caractères alphanumérique.

En imposant sur la première ligne le code 00000 on peut dessiner le caractère sur les sept lignes restantes.

### Organisation de la mémoire

Nous utiliserons comme mémoire une autre EPROM MCM 2708 de 1K 8 bits.

Les 10 lignes adresses (A9 - A0) de la mémoire sont décomposées en deux blocs ; l'adresse ligne et l'adresse caractère.

L'adresse caractère (bits de poids fort A9 - A3) choisit dans la mémoire le signe ou la lettre correspondant au code ASCII appliqué sur ces lignes adresses.

L'adresse ligne (bits de poids faible, A0 - A2) fait le choix d'une ligne du caractère (déjà choisi par A9 - A3) les bits A0.A1.A2 s'incrementent pour donner les 8 lignes.

Chaque caractère est inscrit dans une matrice 8 X 8. Les trois premières colonnes seront vide pour permettre l'espacement entre caractère. Comme nous l'avons indiqué précédement, la première ligne sera vide aussi, elle augmentera l'espacement inter ligne. Le caractère s'écrira ainsi sous forme 5 X 7.

Les 5 points d'une ligne de la matrice correspondront au 5 sorties de l'EPROM.

./.

Exemple : Pour inscrire le caractère A il faut appliquer aux entrées A9 - A3 le code 100 0001 (qui correspond à 4 1 en Hexa : code ASCII de A) et aux entrées A0 A1 A2 les sorties R0 R1 R2 du contrôleur d'écran pour qu'il balaie les 8 positions lignes de la matrice.

A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Matrice de A
1	0	0	0	0	0	1	0	0	0	0 0 0 0 0
							0	0	1	0 1 1 1 0
							0	1	0	1 0 0 0 1
							0	1	1	1 0 0 0 1
							1	0	0	1 0 0 0 1
							1	0	1	1 1 1 1 1
							1	1	0	1 0 0 0 0
							1	1	1	1 0 0 0 0

Ainsi à chaque adresse 10 bits ou fera correspondre une ligne de la matrice. Tout cela est programme en Hexa.

On aura pour A :

Adresse	Données
208	00
209	0E
20A	11
20B	11
20C	11
20D	1F
20E	11
20F	11

Tout les caractères à visualiser ont été ainsi programmés.

./.

### Le registre à décalage

A la sortie du générateur de caractère, l'information est présentée sous forme 5 bits parallèles. Elle n'est plus codée en ASCII, mais sous forme de signal représentant les différentes lignes du caractère. Pour les injecter sur l'entrée Video d'un récepteur TV, il faudrait d'abord les mettre sous forme série. Pour cela on utilisera le registre à décalage SN 74165 à 8 entrées parallèles et une sortie série. La figure 2 représente le diagramme des temps d'un tel registre.

Trois des entrées du registre sont mises à la masse parceque l'information se présente seulement sur 5 bits, ça permet aussi de faire l'espacement entre caractère.

Le registre à décalage à 2 entrées horloges ; l'une avec inhibition et l'autre sans. Comme nous n'avons point besoin d'inhibition, cette entrée sera mise à la masse.

Le registre à décalage est synchronisé avec Latch. Au front descendant de l'impulsion issue du diviseur par 8, l'information qui est dans le Latch passe dans le générateur de caractère, et à la sortie de celui-ci est recueillie la nouvelle configuration de l'information envoyée. Tout cela se fait pendant l'impulsion LOAD. Pendant le SHIFT, le Latch est bloqué mais le registre à décalage est entrain de Shifter les 8 configurations de l'information. A la fin de ce SHIFT arrive un autre LOAD pour faire parvenir un nouveau caractère.

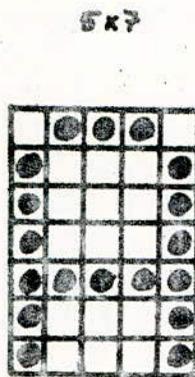
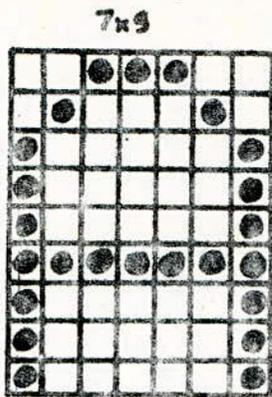
#### Signal Video

L'information peut être produite sur 2 sorties :

- QH (directe) caractères blancs sur fond noir :  
Video directe.
- QH (inversé) caractères noirs sur fond blanc :  
Video inverse.

./.

Exemple de Matrice



Codage de la 5x7  
Pour A

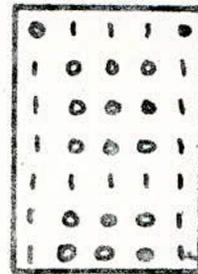


Fig.2 Registre à Decalage 74165

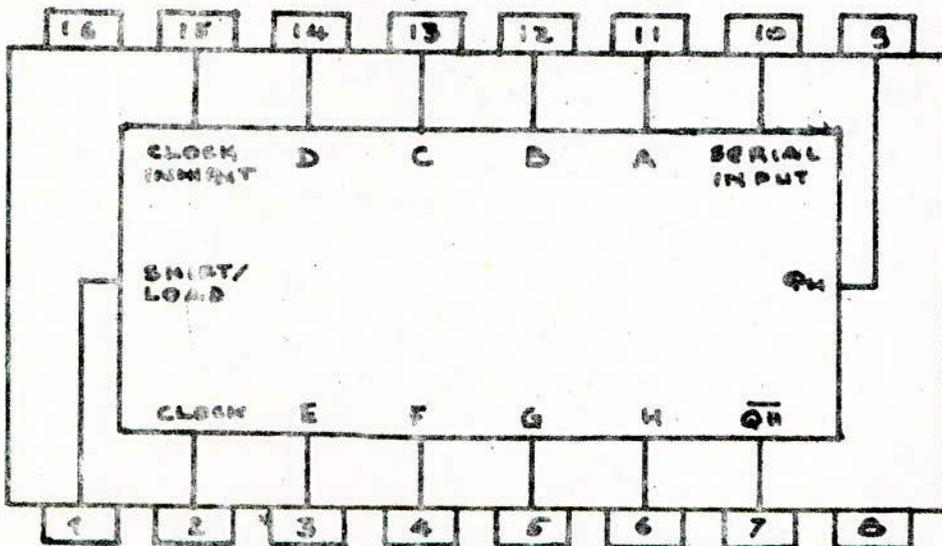


Table de Verité de la 74165

Entrées					Sorties internes		Sortie Q <sub>n</sub>
SHIFT LOAD	CLOCK INHIBIT	CLOCK	Serial	PARALLÈLES A...H	Q <sub>a</sub>	Q <sub>b</sub>	
0	∅	∅	∅	a...h	a	b	h
1	0	0	∅	∅	Q <sub>a0</sub>	Q <sub>b0</sub>	Q <sub>n0</sub>
1	0	↑	1	∅	1	Q <sub>an</sub>	Q <sub>an</sub>
1	0	↑	0	∅	0	Q <sub>an</sub>	Q <sub>an</sub>
1	1	↑	∅	∅	Q <sub>a0</sub>	Q <sub>b0</sub>	Q <sub>n0</sub>

Sur ces sorties on mettra des buffers à collecteurs ouverts pour amplifier le niveau du courant. Pour protéger ces buffers d'un éventuel retour de tension du Moniteur TV, on place une Diode Zener 5V juste à leurs sorties à travers une résistance.

Le signal en sortie est directement composé avec le signal de synchronisation du CRTC.

On place aussi un potentiomètre en sortie pour pouvoir régler la brillance du signal sur l'écran.

## Chapitre VIII

### FONCTIONNEMENT GENERAL DE L'ENSEMBLE

Dans les chapitres precedent, nous avons traité le fonctionnement de chaque partie du circuit. Dans celui-ci nous ferons la recapitulation et expliquerons certaines operations pour une parfaite comprehension du fonctionnement général de l'ensemble.

Nous pouvons, à première vue de ce qui précède, diviser les phases de visualisation en deux étapes:

- la mémorisation du texte dans les RAM
- l'affichage du texte mémorisé

#### I/Mémorisation du texte

Ceci a déjà été traité dans le chapitre rafraichissement d'ecran. Rappelons quand même ce qui se passe avant la mémorisation.

Lorsque l'operateur enfonce une touche du clavier, ce dernier engendre un code caractère formé de 7 bits. Ce code est accompagné d'un bit strobe pour le valider. Ce code entre dans l'UART, où il est directement transféré dans le registre recepteur. L'UART envoie un signal au CRTC pour indiquer que le code caractère est sortie de l'UART. Ce code caractère se presente à la mémoire de décodage d'une part, et passe à travers des portes AND pour se presenter aux entrées des buffers d'autre part. A la sortie de la mémoire de decodage le code spécifié par O2, OI et Oo se presente au CRTC. Selon la combinaison de ce code on aura:

- Soit un caractère à visualiser. Le CRTC autorisera alors l'écriture du code caractère dans les RAM et incrementera le curseur d'une position. Les buffers deviennent passants et le  $\bar{W}$  des RAM passe à 0

- Soit un mouvement du curseur. Le CRTC agit directement sur le curseur sans autorisation d'écriture. Les buffers resteront donc bloqués et le  $\bar{W}$  des RAM sera à 1.
- Soit un ordre d'effacement. La ligne RO2 du CRTC, en envoyant un 0 sur les portes AND forcera le caractère blanc (code: 000 0000) dans les cases mémoires concernées. Les buffers sont passants et le  $\bar{W}$  des RAM est à 0.

Les caractères à visualiser seront stockés dans les mémoires de rafraîchissement, chacun dans l'emplacement qui lui correspond sur l'écran.

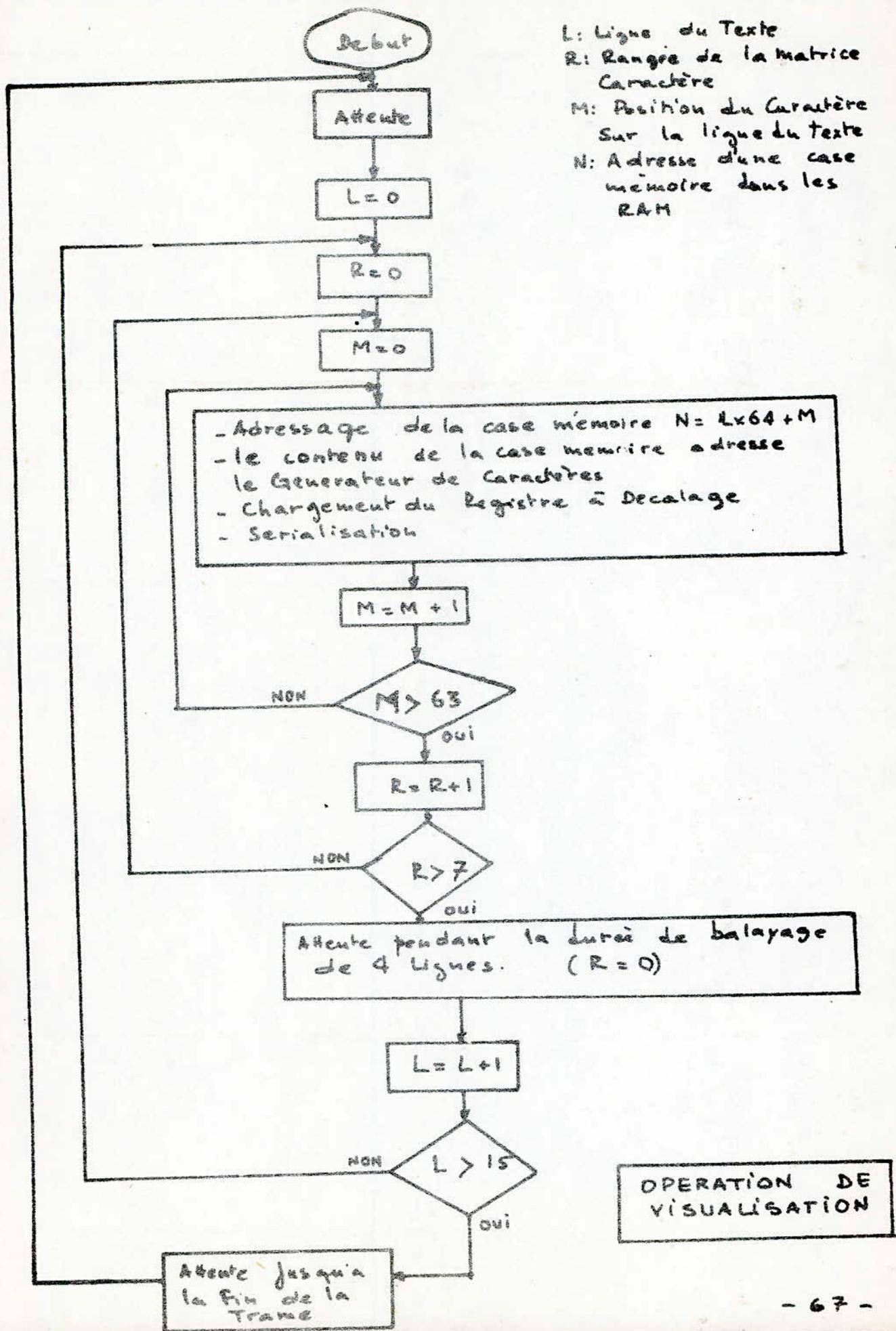
## II/Visualisation du texte

Cette phase, la plus importante d'ailleurs, est légèrement complexe. Les séquences de déroulement de cette phase sont illustrées par la figure 8.1 et l'organigramme.

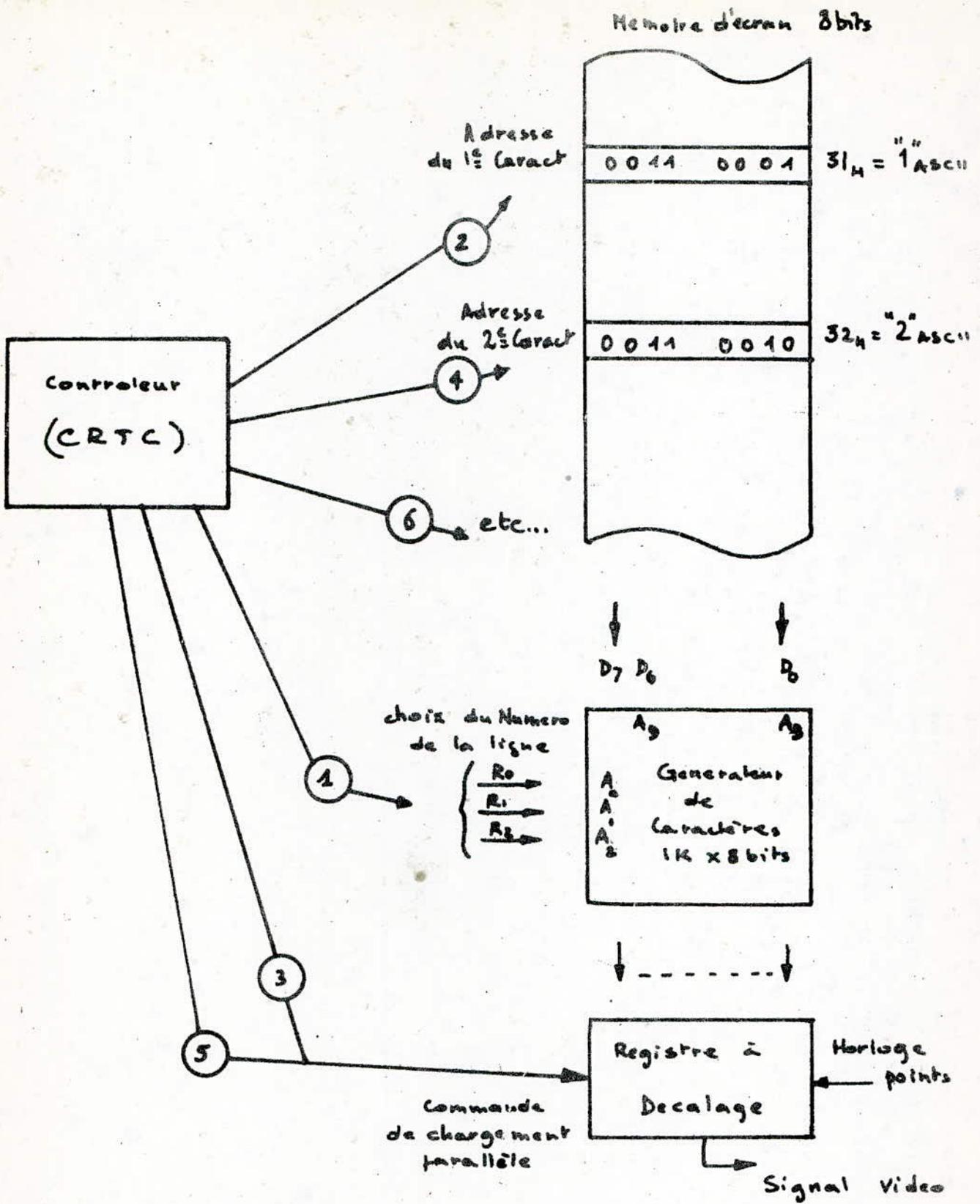
### Interprétation de l'organigramme

- Le CRTC adresse les 64 premières cases mémoires pendant la durée d'une ligne: c'est la première caractère de la 1<sup>o</sup> ligne du texte. On visualise alors la 1<sup>o</sup> ligne de tout les caractères de la 1<sup>o</sup> ligne du texte.
- Le CRTC adresse la ligne suivante, c'est à dire la 2<sup>o</sup> ligne caractère de la 1<sup>o</sup> ligne du texte. Il fait cela jusqu'à balayer les 8 lignes caractères. La 1<sup>o</sup> ligne du texte est ainsi visualisée.
- Le CRTC arrête l'adressage des RAM pendant l'interligne, et le reprend pour la 2<sup>o</sup> ligne de texte. Il adresse: la 1<sup>o</sup> ligne caractère, la 2<sup>o</sup> ligne caractère et cela jusqu'à la 8<sup>o</sup> ligne caractère. La 2<sup>o</sup> ligne de texte est ainsi visualisée.
- Le CRTC fera cela jusqu'à ce que les 16 lignes soient visualisées.

L: Ligne du Texte  
 R: Rangée de la Matrice Caractère  
 M: Position du Caractère Sur la ligne du texte  
 N: Adresse d'une case mémoire dans les RAM



OPERATION DE VISUALISATION



**SEQUENCE DE TRAVAIL DU CONTROLEUR D'ECRAN.**

-Le CRTIC fera cela jusqu'à ce que les 16 lignes soient visualisées.

Tout ce cycle dure  $1/50$  s = 20ms, et correspond à la période d'une trame. Il refera la même chose pour la trame suivante. En effet chaque point d'une matrice caractère correspondrait alors à deux points élémentaires sur l'écran l'un sur la ligne paire et l'autre sur la ligne impaire.

## CONCLUSION

Notre travail entre dans le cadre de l'elaboration d'un "système d'aquisition de données experimantales" au sein de la division du CSTN.

L'affichage d'un texte de 16 lignes de 64 caractères et les possibilités d' enchainement de plusieurs pages font de ce circuit un outil de travail remarquable tant du point de vue rapidité que du point de vue economie.L'utilisation d'une teletype ne sera necessaire que lorsqu'il s'agira de garder les informations sur papier.

La vitesse de transmission entre l'ordinateur et le circuit de visualisation ne peut dépasser 1200 bauds.Ceci est du aux circuits TTL simple que nous avons utilisé,alors que le CRTC peut monter jusqu'à la vitesse de 9600bauds.Nous proposons pour remedier à cela de remplacer les circuits TTL simple par des circuits TTL-LS qui ont des temps de montée de l'ordre d'une dizaine de nanosecondes,ce qui permettrait d'aller au maximum du CRTC

Dernière remarque à faire ;le terminal etant doté d'une sortie teletype,il est connectable à n'importe quelle unité de traitement informatique.Ce terminal est déjà, operationnel dans la division six.

ANNEXE :Caractéristiques et schema de brochage de la 2114

C'est une mémoire à accès aleatoire du type statique. Elle est organisée en  $1K \times 4$  bits soit 4096 bits. Elle est realisée en technologie MOS canal N, grille silicium. Elle necessite une seule tension d'alimentation ce qui facilite grandement son utilisation

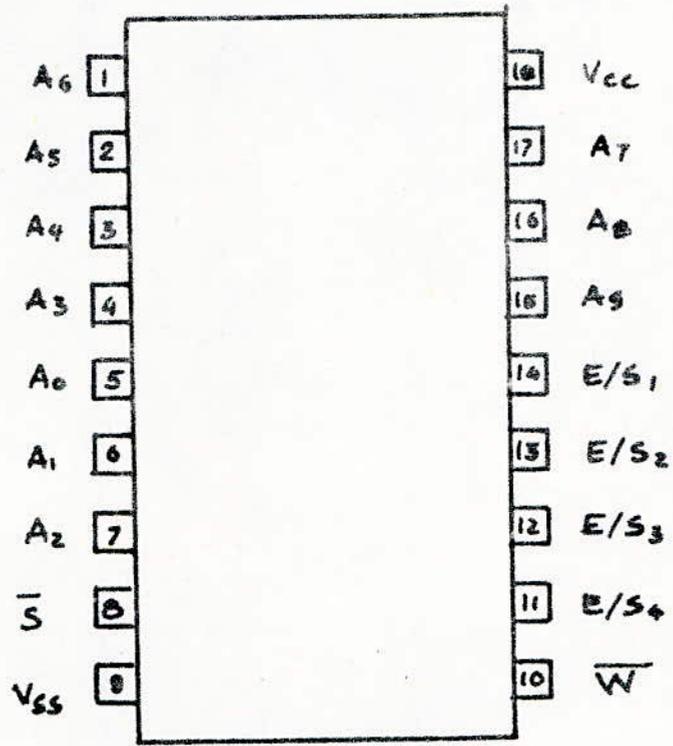
Ses caracteristiques sont:

- fonctionnement statique
- organisation en 1024 demi-octets
- tension d'alimentation: +5V.
- compatible TTL/DTL
- entrées, sorties bidirectionnelles
- sorties à 3 etats
- faible dissipation

BROCHAGE (figure page suivante)

- A0-A9 :lignes adresses
- E/SI - E/S4 :lignes d'entrées-sorties des données.
- $\bar{S}$  :à 0 la RAM est selectionnée  
à 1 la RAM n'est pas selectionnée
- $\bar{W}$  :à 0 écriture autorisée  
à 1 lecture autorisée
- Vcc: +5V
- Vss: Masse

Le temps de lecture est égal au temps de cycle d'écriture qui doit être inférieur à 450 ns



$\bar{S}$	$\bar{W}$	Selection
0	0	Ecriture autorisée
0	1	Lecture autorisée
1	0	haute impédance
1	1	haute impédance

BROCHAGE ET FONCTIONNEMENT DE  
 LA MEMOIRE 2114

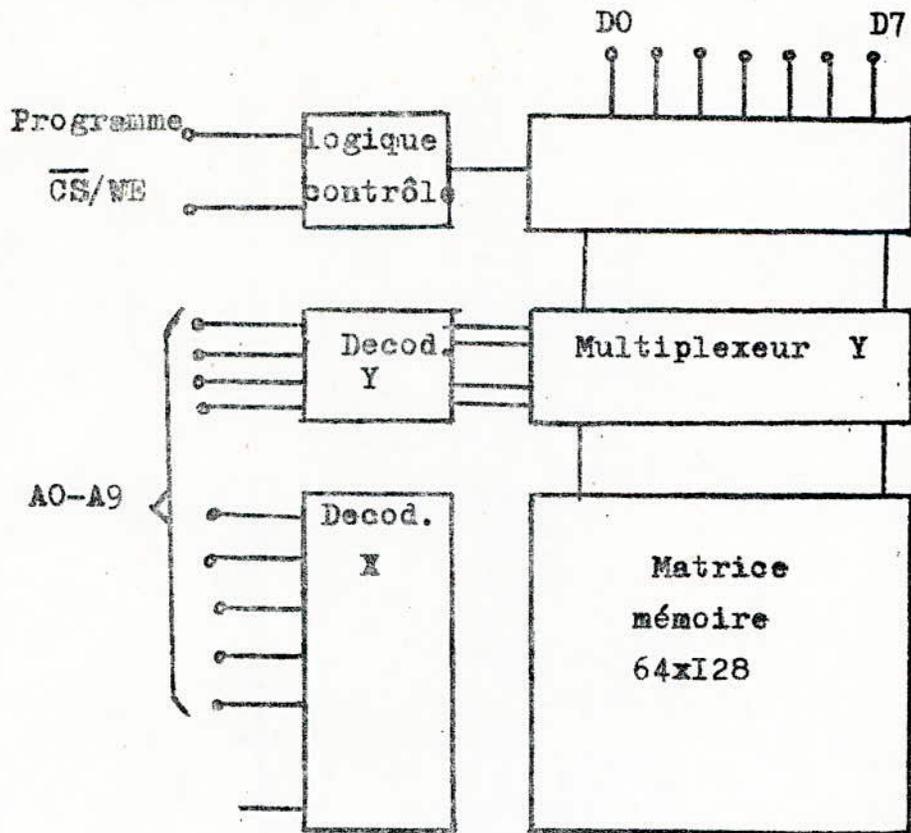
ANNEXE :Caractéristiques et organisation de la MCM 2708

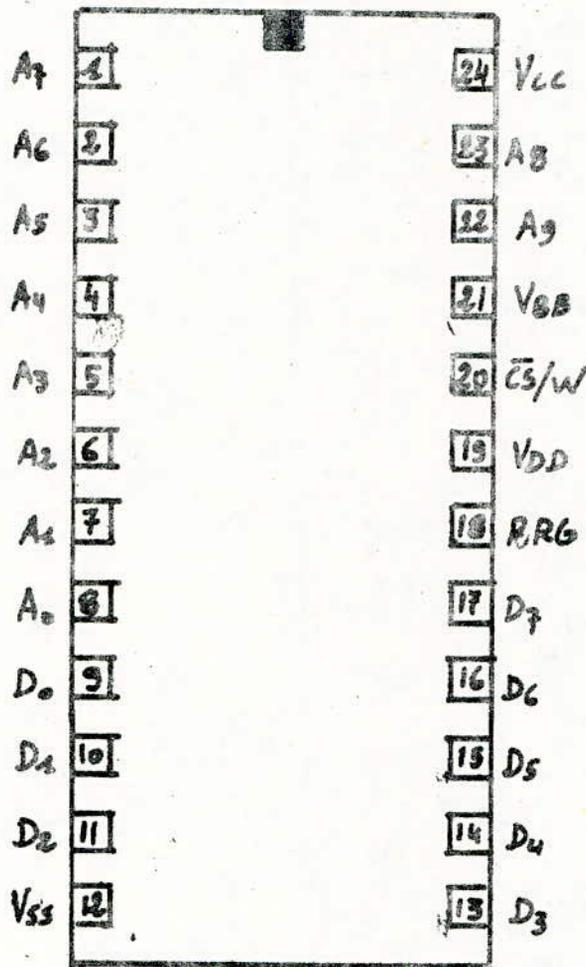
La 2708 est une mémoire effaçable et reprogrammable électriquement de 8192 bits. Elle est utilisable pour la mise au point de système et pour des applications similaires demandant une mémoire non volatile, qui doit être reprogrammée périodiquement. La fenêtre transparente sur le boîtier permet d'effacer aux rayons ultra-violet le contenu de la mémoire.

Ses caractéristiques sont :

- fonctionnement statique
- organisation en 1024 octets
- tension d'alimentation: +5V, -5V, +12V.
- faible dissipation
- compatibilité TTL
- sortie trois états

Son organisation interne est:



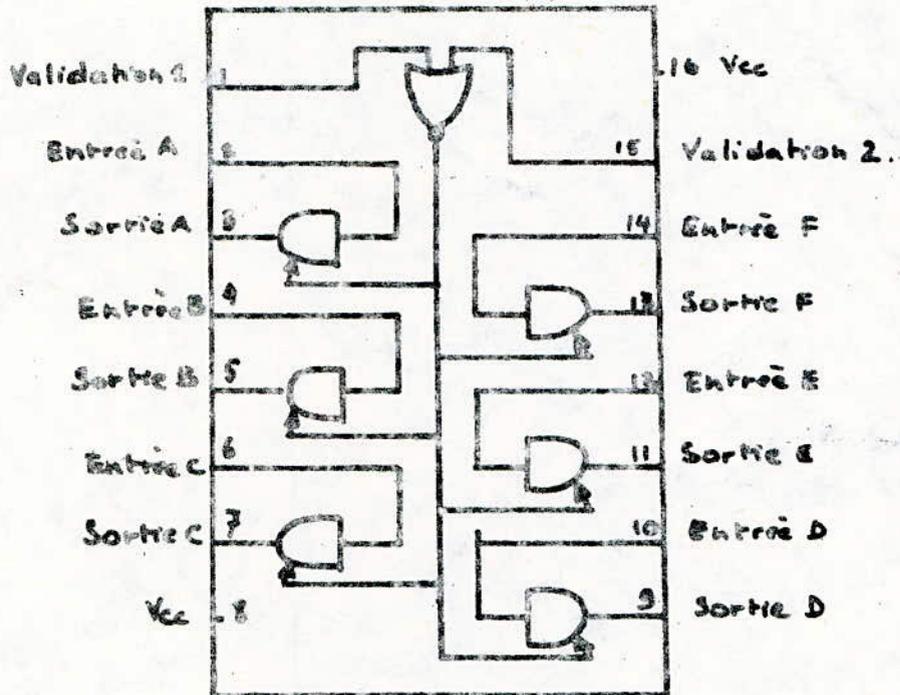


A<sub>0</sub>-A<sub>9</sub> : Adresses.  
 D<sub>0</sub>-D<sub>7</sub> : E/S données  
 VBB = -5V.  
 VDD = +12V  
 Vcc = +5V  
 Vss = 0V.  
 PRG : impulsion  
 de programme  
 CS/W : Sélection  
 Lecture  
 Ecriture.

### BROCHAGE de la MCM 2708.

MODE	AFFECTATION des BROCHES.						
	D <sub>1</sub> -D <sub>7</sub>	12	18	19	20	21	24
LECTURE	Sortie des Données	Vss	Vss	VDD	VDD	VBB	Vcc
ECRITURE	Entrée des Données	Vss	V <sub>ENP</sub>	VDD	V <sub>IHW</sub>	VBB	Vcc

V<sub>ENP</sub>: TENSION A L'ETAT HAUT DE L'IMPULSION PROGRAMME  
 V<sub>IHW</sub>: TENSION A L'ETAT HAUT A L'ENTREE

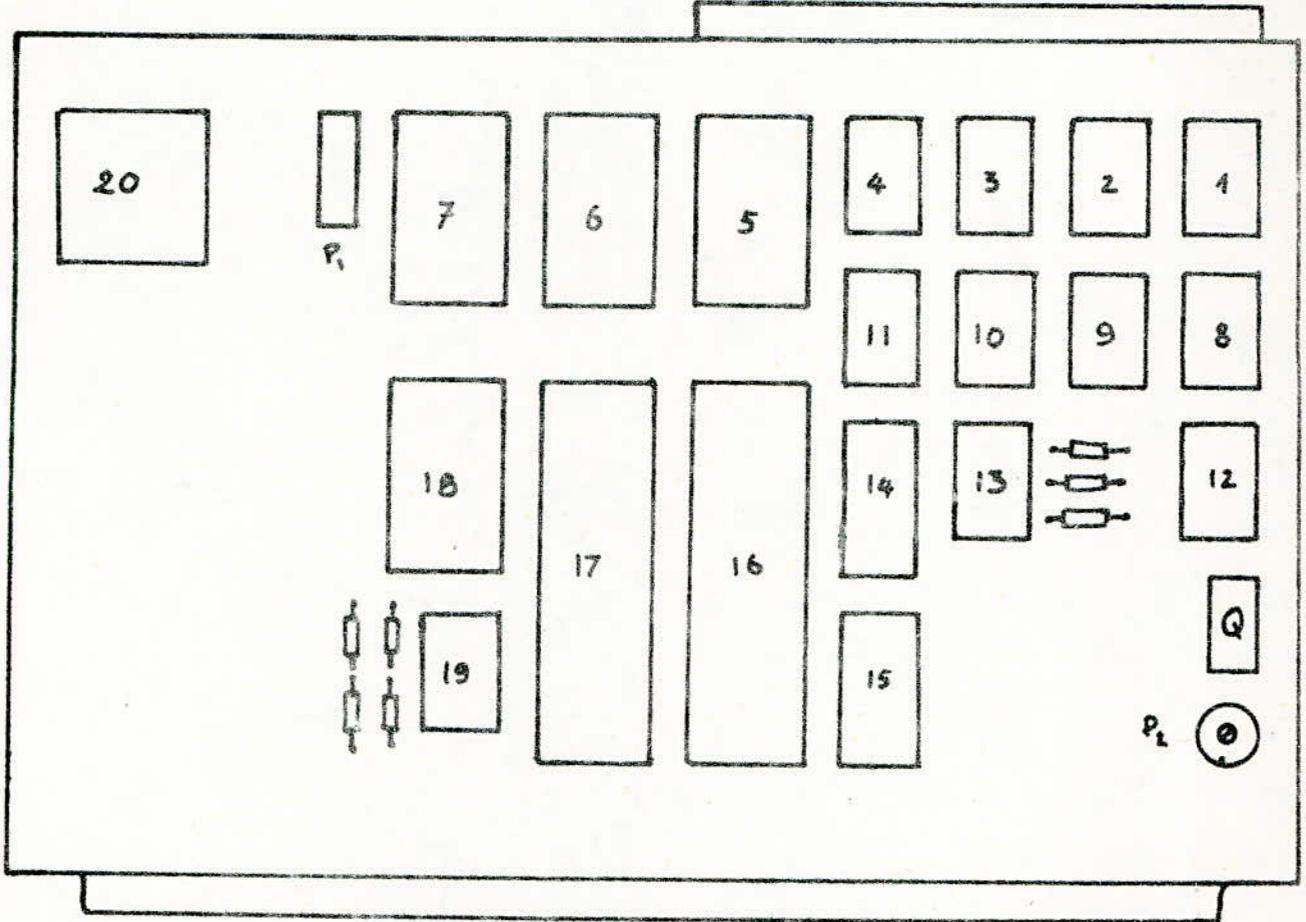


Schema interne des Buffers 3 etats 8T95

Valid 2	Valid 1	Entree	Sortie
0	0	0	0
0	0	1	1
0	1	0	Z
1	0	0	Z
1	1	0	Z

Table de verite des 8T95

CONNECTEUR CLAVIER



- 1/ Compteur
- 2/ Registre à décalage
- 3/ Buffers
- 4/ Buffers
- 5/ Latch
- 6/ Générateur de Caractères
- 7/ Mémoire de décodage
- 8/ Nand Collecteur ouvert
- 9/ Nand
- 10/ And
- 11/ And
- 12/ Trigger
- 13/ Nand
- 14/ Mémoire de Rafraîchissement
- 15/ Mémoire de Rafraîchissement
- 16/ UART
- 17/ CRTC
- 18/ Baudrate
- 19/ coupleurs isolateurs
- 20/ Circuit de Régulation (-5V)

- P<sub>1</sub>/ Réglage du Niveau du Signal Vidéo
- P<sub>2</sub>/ Réglage Oscillateur
- Q/ Quartz 1MHz

SCHEMA DE DISPOSITION  
DES ELEMENTS SUR  
LA PLAQUETTE

Programme charge - Jans 2708      Générateur de Caractères

3000 00 00 00 00 00 00 00 00 00 FF FF FF FF FF FF FF FF .....  
3010 FF .....  
3020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
3030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
3040 FF .....  
3050 FF .....  
3060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
3070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
3080 FF .....  
3090 FF .....  
30A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
30B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
30C0 FF .....  
30D0 FF .....  
30E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....  
30F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

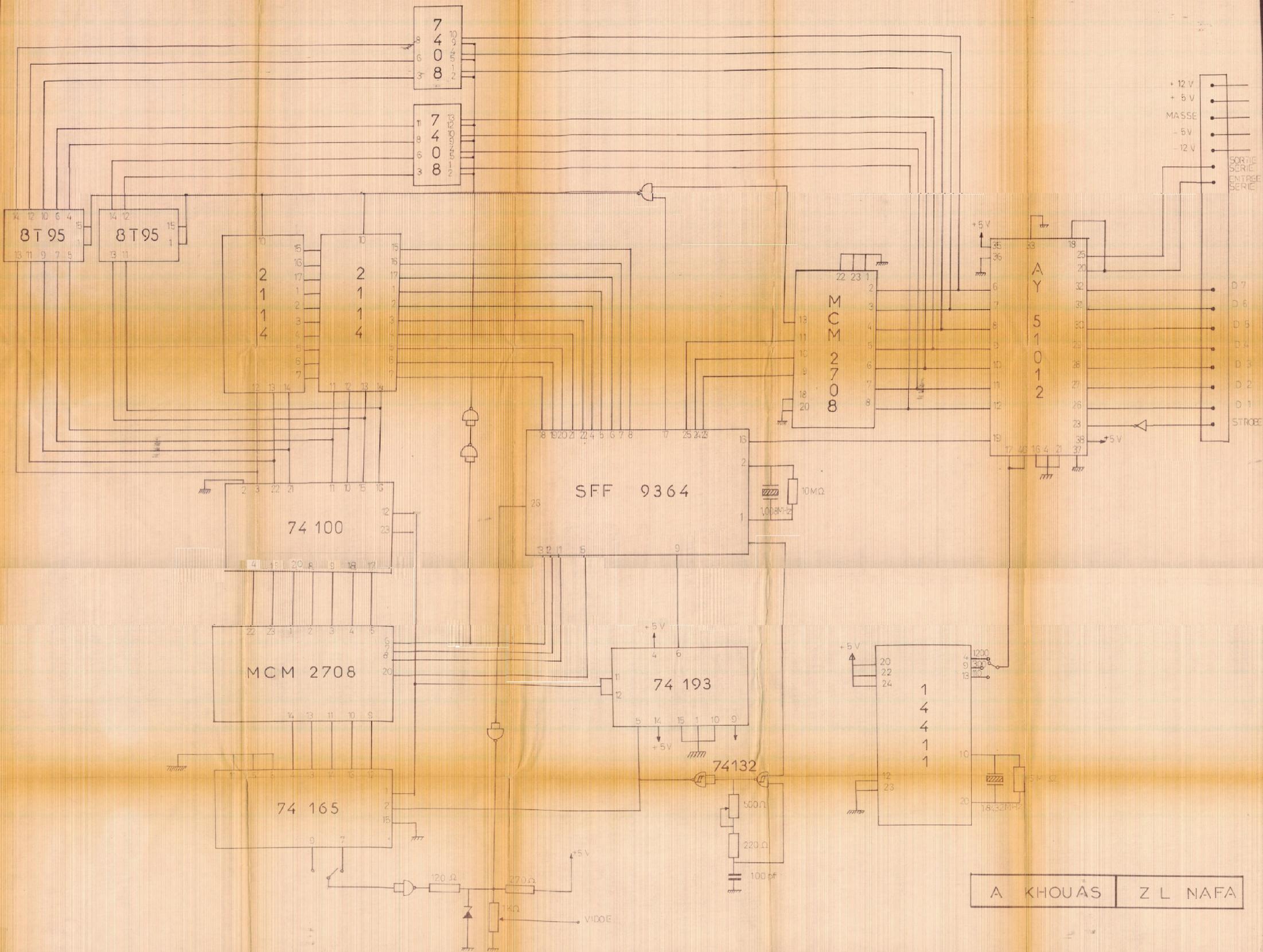
3100 00 00 00 00 00 00 00 00 00 04 04 04 04 04 00 04 .....  
3110 00 0A 0A 0A 00 00 00 00 00 00 0A 1F 0A 1F 0A 00 .....  
3120 00 0E 15 14 0E 05 15 0E 00 18 19 02 04 08 13 03 .....  
3130 00 0C 12 12 0C 13 12 0D 00 06 06 02 04 00 00 00 .....  
3140 00 02 04 08 08 08 04 02 00 08 04 02 02 02 04 08 .....  
3150 00 00 04 15 0E 15 04 00 00 00 04 04 1F 04 04 00 .....  
3160 00 00 00 00 06 06 02 04 00 00 00 00 1F 00 00 00 .....  
3170 00 00 00 00 00 00 00 0C 0C 00 00 01 02 04 08 10 00 .....  
3180 00 0E 11 13 15 19 11 0E 00 04 0C 04 04 04 04 0E .....  
3190 00 0E 11 01 02 0C 10 1F 00 0E 11 01 06 01 11 0E .....  
31A0 00 02 06 0A 12 1F 02 02 00 1F 10 1E 01 01 11 0E .....  
31B0 00 0E 10 10 1E 11 11 0E 00 1F 01 02 04 08 08 08 .....  
31C0 00 0E 11 11 0E 11 11 0E 00 0E 11 11 0F 01 01 0E .....  
31D0 00 00 0C 0C 00 00 0C 0C 00 06 06 00 06 06 04 08 .....  
31E0 00 02 04 08 10 08 04 02 00 00 00 1E 00 1E 00 00 .....  
31F0 00 08 04 02 01 02 04 08 00 06 09 09 02 02 00 02 .....

3200 00 0E 11 01 0D 15 16 0C 00 0E 11 11 11 1F 11 11 .....  
3210 00 1E 09 09 0E 09 09 1E 00 0E 11 10 10 10 11 0E .....  
3220 00 1E 09 09 09 09 09 1E 00 1F 10 10 1E 10 10 1F .....  
3230 00 1F 10 10 1E 10 10 10 00 0F 10 10 17 11 11 0E .....  
3240 00 11 11 11 1F 11 11 11 00 0E 04 04 04 04 04 0F .....  
3250 00 01 01 01 01 01 11 0E 00 11 12 14 18 14 12 11 .....  
3260 00 10 10 10 10 10 10 1F 00 11 1B 15 15 11 11 11 .....  
3270 00 11 19 15 13 11 11 11 00 0E 11 11 11 11 11 0E .....  
3280 00 1E 11 11 1E 10 10 10 00 0E 11 11 11 16 12 0C .....  
3290 00 1E 11 11 1E 14 12 11 00 0E 11 10 0E 01 11 0E .....  
32A0 00 1F 04 04 04 04 04 04 00 11 11 11 11 11 11 0E .....  
32B0 00 11 11 11 0A 0A 04 04 00 11 11 11 16 16 16 0A .....  
32C0 00 11 11 0A 04 0A 11 11 00 11 11 0A 04 04 04 04 .....  
32D0 00 1F 01 02 04 08 10 1F 00 0E 08 08 08 08 08 0E .....  
32E0 00 00 10 08 04 02 01 00 00 0E 02 02 02 02 02 0E .....  
32F0 00 04 04 04 04 04 04 00 00 00 00 00 00 00 1E .....

3300 00 00 00 00 00 00 00 00 00 00 00 1C 02 0E 12 0D .....  
3310 00 10 10 10 1C 12 12 1C 00 00 00 00 0E 10 10 0E .....  
3320 00 02 02 02 0E 12 12 0E 00 00 00 0C 12 1C 10 0C .....  
3330 00 0C 12 10 1C 10 10 10 00 00 00 0C 12 0E 02 1C .....  
3340 00 10 10 10 1C 12 12 12 00 04 00 0C 04 04 04 0E .....  
3350 00 06 02 02 02 12 12 0C 00 10 10 12 14 18 14 12 .....  
3360 00 0C 04 04 04 04 04 0E 00 00 00 00 1A 15 15 15 .....  
3370 00 00 00 00 1C 12 12 12 00 00 00 00 0C 12 12 0C .....  
3380 00 0C 12 12 1C 10 10 10 00 00 10 10 0E 12 12 0C .....



# SCHEMA DE MONTAGE DU CIRCUIT DE VISUALISATION



A KHOUAS      Z L NAFA