

الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

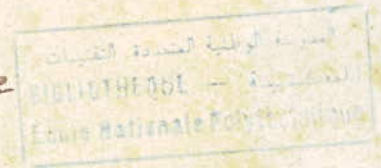
42/87

وزارة التعليم و البحث العلمي
Ministère de l'Enseignement et de la Recherche Scientifique

Texte

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT *Electronique*



PROJET DE FIN D'ETUDES

SUJET

Conception et réalisation d'un
compteur d'erreurs, dans une
transmission numérique

Proposé par : J. Goralski Étudié par : (m) Zbadi a Dirigé par : J. G ORALSKI

PROMOTION: JUIN 87

Dédicaces

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

A mes parents
-et
à mes amis



Remerciements

المدرسة الوطنية المتعددة التقنيات
المكتبة - BIBLIOTHEQUE
Ecole Nationale Polytechnique


Je tiens à remercier Monsieur GORALSKI,
pour avoir proposé ce sujet, accepté de me
diriger, et m'aider durant tout ce travail.

Je profite aussi, de remercier Monsieur SEKHRI
Boualem pour son aide morale, et sans oublier
aussi Monsieur SAHED Abdelmalek pour avoir
rédigé ce manuscrit.

Enfin que tous ceux, qui ont contribué, de près ou
de loin à la réalisation de ce travail, trouvent
ici mes plus vifs remerciements.

Je remercie Messieurs les membres du Jury, qui
par leurs critiques et jugements, ont honoré ce
modeste travail.

Signé
Mohamed ZADIA.



Sommaire

Introduction

A. Conception

I. Transmission en bande de base

II. Modes de transmission - synchronisation

1) Transmission synchrone

2) Transmission asynchrone

III. Extraction des erreurs

1) Méthodes utilisées

2) Comparaison des différentes méthodes.

B. Réalisation

I. Schéma synoptique

II. Emission

1) Horloge

2) Les diviseurs de fréquence

3) l'émetteur

III. Réception

1) Extraction de l'horloge du signal reçu

2) Génération et injection des erreurs

3) Le recepteur

- a) Configuration « synchro »
 - b) Configuration « work »
 - c) les 2 configurations du recepteur
- ### 4) Logique de detection et de commande .
- a) Detection de la synchronisation
 - b) Logique de commande
 - c) Module comptage - décodage - affichage .

Bibliographie .

Conclusion .

Introduction .

Si il s'agit de situer le sujet aussi brièvement que possible, on dira transmission de données, et si il s'agit d'en donner une idée précise, on dira comptage d'erreurs dans une transmission numérique.

Il s'agit en effet de concevoir et de réaliser, un dispositif capable d'indiquer le taux d'erreurs engendrés par une ligne de transmission en bande de base relative à une transmission synchrone.

A : Conception.

I/ Transmission en bande de base.

Dans le cas d'une ligne en bande de base (ligne métallique de courte distance), la bande passante est suffisante pour permettre une transmission numérique. Ce qui procure une grande qualité de transmission, permet l'utilisation de composants électroniques à haut degré d'intégration et évite les transformations analogiques numériques successives.

Cependant la présence fréquente de transformateurs aux extrémités de la ligne limite la bande passante vers les fréquences basses.

II/ Modes de transmission - synchronisation.

Une des difficultés de la transmission de données sous-forme série consiste à synchroniser le récepteur. Il convient en effet à ce dernier de déterminer à quels instants, il doit prendre en compte l'information reçue : C'est la synchronisation bit.

1/ Transmission synchrone

Dans ce mode, les bits sont émis de façon régulière, au rythme du signal d'horloge, sans séparation entre les caractères successifs.

Le récepteur comporte une horloge bit de même fréquence nominale que celle de l'émetteur (autrement dit, il sait à quel débit binaire, il reçoit les données).

Les transitions du signal reçu permettent de reconstituer en permanence le rythme bit et d'asservir en phase l'horloge de réception. Cette opération est facilitée, en début de transmission par l'émission de séquences riches en transitions.

Ce mode de transmission permet des débits binaires importants. Il est systématiquement utilisé pour les débits de 1200 bits/s.

2) Transmission asynchrone (ou START/STOP).

Il est parfois intéressant de transmettre l'information caractère par caractère, au moment où ceux-ci sont émis (c'est le cas d'un clavier par exemple), l'intervalle de temps entre caractères étant quelconque. Il n'y a synchronisation entre émetteur et récepteur que pendant la transmission de chaque caractère. Il est nécessaire d'ajouter des bits au début et à la fin du caractère (par exemple 1 bit START et 1 ou 2 bits STOP) pour permettre la synchronisation du récepteur.

Comme en transmission synchrone, les bits d'un caractère sont transmis à la cadence d'une horloge bit. Le récepteur possède lui aussi une horloge de même fréquence et connaît la structure des caractères (nombre de bits de données, nombre de bits START et STOP). La transition provoquée par le signal START permet au récepteur de caler en phase son horloge.

Le choix est porté sur le premier mode de transmission c'est à dire le mode synchrone.

III Extraction des erreurs.

La mesure des caractéristiques d'un système de transmission digitale fait appel à la génération de séquences digitales de structure connue et qui sont utilisées comme données de test.

La comparaison bit par bit de données transmises et celles reçues permet d'extraire les erreurs introduites par le canal.

1) Méthodes utilisées

Diverses techniques sont utilisées, nous citerons quelques unes.

a) Générer une séquence binaire, la transmettre vers l'utilisateur, retransmettre la séquence reçue vers l'émetteur le long d'un autre canal et effectuer au niveau de l'émetteur une comparaison bit par bit.

b) Générer des séquences pseudo-aléatoires synchronisées identiques au niveau de l'émetteur et du récepteur.

Transmettre la séquence de l'émetteur vers le récepteur et effectuer une comparaison bit par bit de la séquence reçue et celle localement générée.

c) Générer une séquence pseudo-aléatoire en utilisant les techniques de registres à décalage, la transmettre vers le récepteur et traiter la séquence reçue en la faisant passer à travers l'inverse du registre générateur

d) Générer une séquence binaire, la transmettre vers l'utilisateur, enregistrer le flot de bits reçus et effectuer ultérieurement une comparaison bit par bit avec la séquence émise qui est déjà enregistrée.

e) Utilisation d'une combinaison des méthodes b) et c), le récepteur étant en premier lieu synchronisé avec l'émetteur en utilisant la méthode c). Il est alors "libéré" pour générer la même séquence pseudo-aléatoire indépendamment de l'émetteur et une comparaison bit par bit est effectuée comme dans b).

2) Comparaison des différentes méthodes.

La première méthode nécessite l'utilisation d'un canal de rétroaction (feedback channel, fiable qui n'introduit pas à son tour des erreurs) ce qui augmente le coût global du système.

La deuxième méthode est la plus utilisée, mais elle nécessite généralement un équipement de synchronisation complexe.

La troisième est simple. Cependant, elle ne détecte pas l'échantillon d'erreur réel, mais plutôt une fonction de cet échantillon. Pour remédier à ceci, on doit ajouter un autre registre à décalage avec une boucle de réaction: "registre d'erreur".

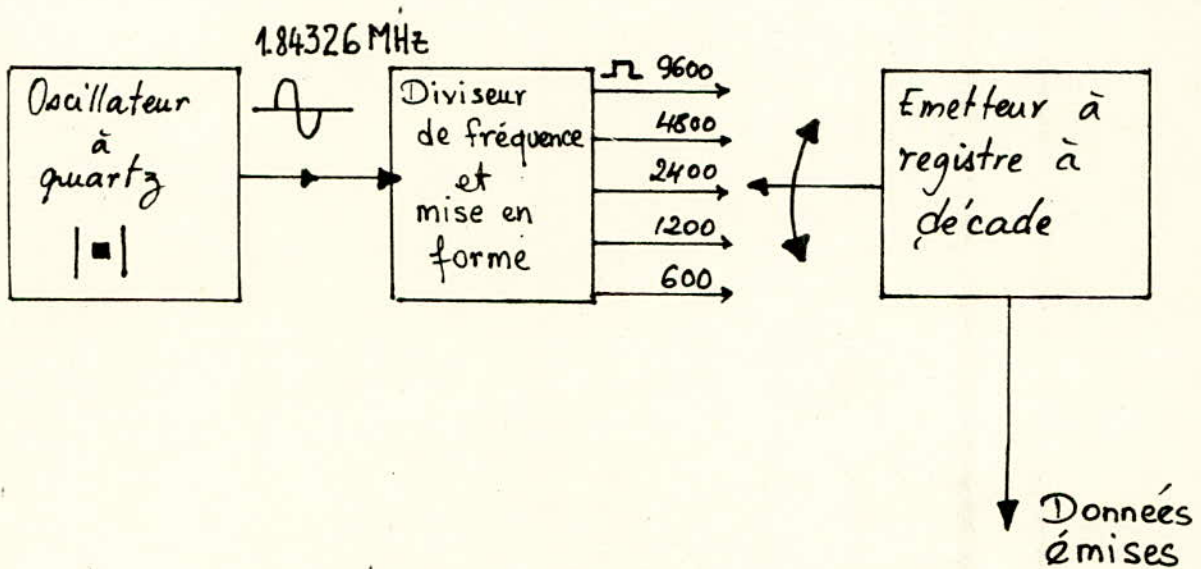
La quatrième méthode devient impraticable pour des tests comportant une très forte densité d'informations.

Le choix est tombé sur la cinquième méthode qui d'un côté utilise l'idée de la 3^{ème} méthode simple et efficace à la fois, qui en faisant passer la séquence reçue à travers l'inverse du registre générateur, synchronise automatiquement le récepteur, et de l'autre côté, à l'aide des commutateurs électroniques, une fois la synchronisation obtenue entre le récepteur et l'émetteur, utilise l'idée de la 2^{ème} méthode.

B : Réalisation

I / Schéma synoptique

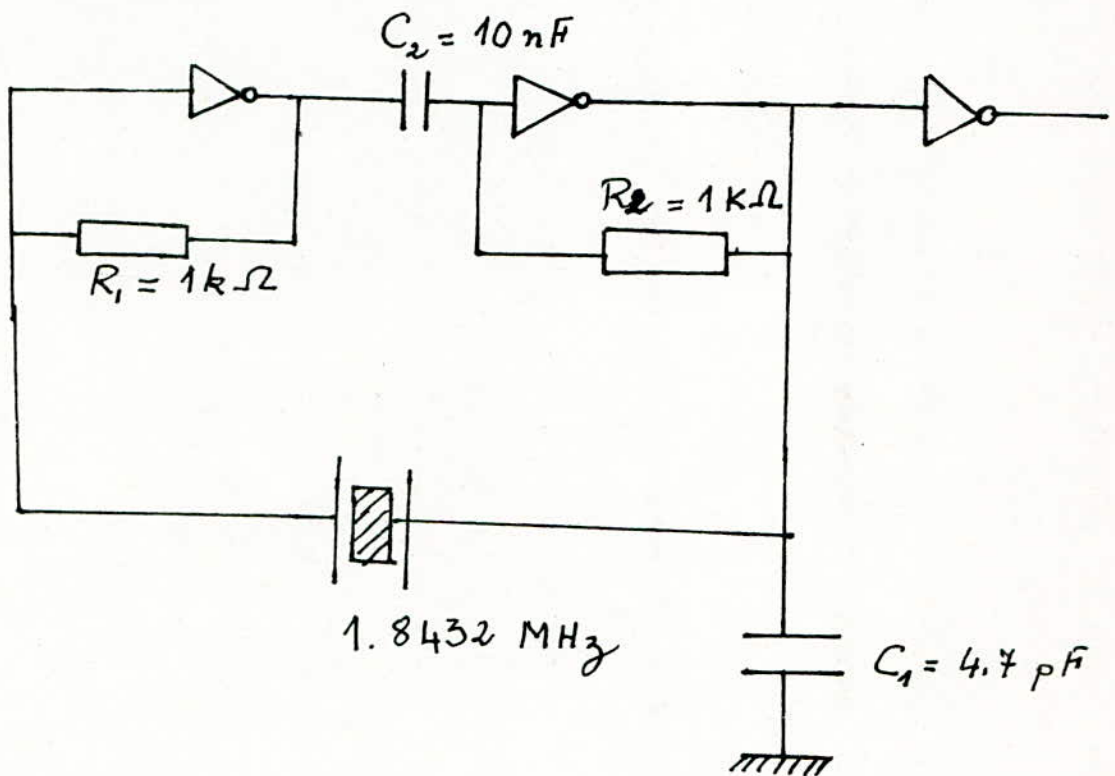
Les deux figures qui suivent donnent les schémas synoptiques de l'émetteur et du récepteur.



Emission

II) Emission : 1°) L'horloge

Un oscillateur à Quartz a été réalisé. Son schéma est le suivant :



Cet Oscillateur est constitué d'un amplificateur de gain supérieur à un dont le déphasage est égal à 360° pour la fréquence désirée et d'une boucle de réaction qui comprend le quartz.

Le déphasage est assuré par 2 inverseurs. Deux résistances R_1 et R_2 sont choisies de manière à amener le point de repos de ses amplificateurs dans la partie linéaire de leur caractéristique de transfert. L'oscillateur à quartz est recommandé pour sa précision et sa stabilité.

2) Les diviseurs de fréquence

Pour obtenir les différentes vitesses de transmission des données, on attaque une série de diviseurs de fréquence par le signal issu de l'oscillateur à quartz. La fréquence délivrée par l'oscillateur est $f_0 = 1.8432 \text{ MHz}$. A partir de cette fréquence, on peut obtenir les fréquences suivantes :

$$f_1 = 1.8432 \text{ MHz} \div 12 \times 16 \longrightarrow 9600 \text{ bits/s}$$

$$f_2 = 1.8432 \text{ MHz} \div 12 \times 16 \times 2 \longrightarrow 4800 \text{ bits/s}$$

$$f_3 = 1.8432 \text{ MHz} \div 12 \times 16 \times 4 \longrightarrow 2400 \text{ bits/s}$$

$$f_4 = 1.8432 \text{ MHz} \div 12 \times 16 \times 8 \longrightarrow 1200 \text{ bits/s}$$

$$f_5 = 1.8432 \text{ MHz} \div 12 \times 16 \times 16 \longrightarrow 600 \text{ bits/s}$$

La division de fréquence est une forme de comptage puisqu'il s'agit d'obtenir une impulsion de sortie pour n impulsions d'entrée, n étant un nombre quelconque.

Pour ces divisions de fréquences, les compteurs choisis sont les SN 74LS93. On a besoin de 3 compteurs pour pouvoir véhiculer les données aux vitesses citées plus haut.

Le compteur SN 74LS93 est un compteur asynchrone c'est à dire que l'horloge est appliquée à la première bascule seulement PIN(14). La sortie Q_n de la $n^{\text{ième}}$ bascule est appliquée à l'entrée d'horloge de la $(n+1)^{\text{ème}}$ bascule et ainsi de suite, ce qui laisse les bascules changer d'état successivement. La liaison entre la première bascule et la $2^{\text{ème}}$ deuxième bascule n'est pas réalisé, il faudra donc

relier la PIN (12) à la PIN (1).

On a besoin d'une division par 12. Le 1^{er} compteur sera câblé de telle manière à réaliser cette division. Le 12 en binaire s'écrit 1100 , donc il s'agira de relier les sorties Q_D Q_C Q_B Q_A Q_C (Pin 8) et Q_D (Pin 11) aux bornes (2) et (3) qui sont les entrées d'une porte NAND intégrée qui elle à son tour est déjà reliée avec la remise à zéro des bascules.

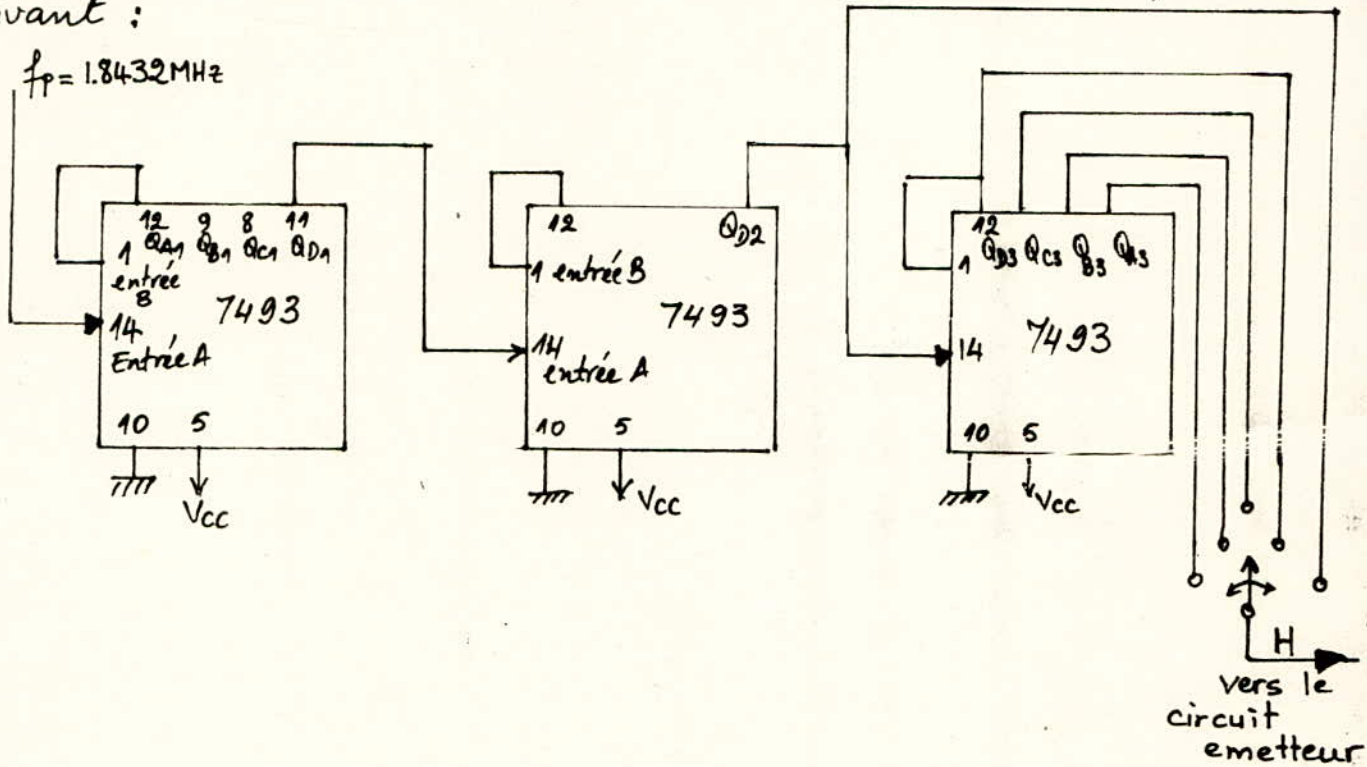
Ces compteurs sont mis en cascade pour avoir les divisions suivantes donc la sortie de la 4^{ème} bascule (Pin 11) de chaque compteur attaque l'entrée (Pin 14) du compteur qui suit. Les 2^{ème} et 3^{ème} compteurs sont utilisés comme diviseurs par 16.

Les vitesses de transmission, avec les différents points où ils sont pris sont :

$$\begin{aligned} f_1 &= 9600 \text{ bits/s} \rightarrow Q_{D2} \text{ (Pin 11 du 2^{ème} compteur SN 7493)} \\ f_2 &= 4800 \text{ bits/s} \rightarrow Q_{A3} \text{ (Pin 12 " 3^{ème} " " ")} \\ f_3 &= 2400 \text{ bits/s} \rightarrow Q_{B3} \text{ (Pin 9 " " " " ")} \\ f_4 &= 1200 \text{ bits/s} \rightarrow Q_{C3} \text{ (Pin 8 " " " " ")} \\ f_5 &= 600 \text{ bits/s} \rightarrow Q_{D3} \text{ (Pin 11 " " " " ")} \end{aligned}$$

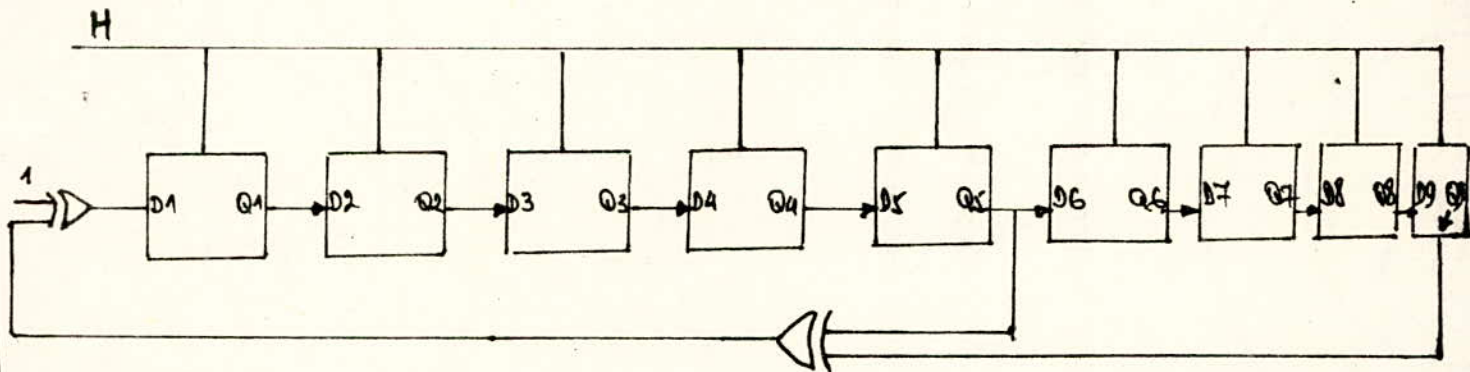
On sélectionnera une des vitesses à la fois à l'aide d'un commutateur rotatif.

Le schéma représentant la division de fréquence issue de l'oscillateur local à quartz et le commutateur rotatif est le suivant :

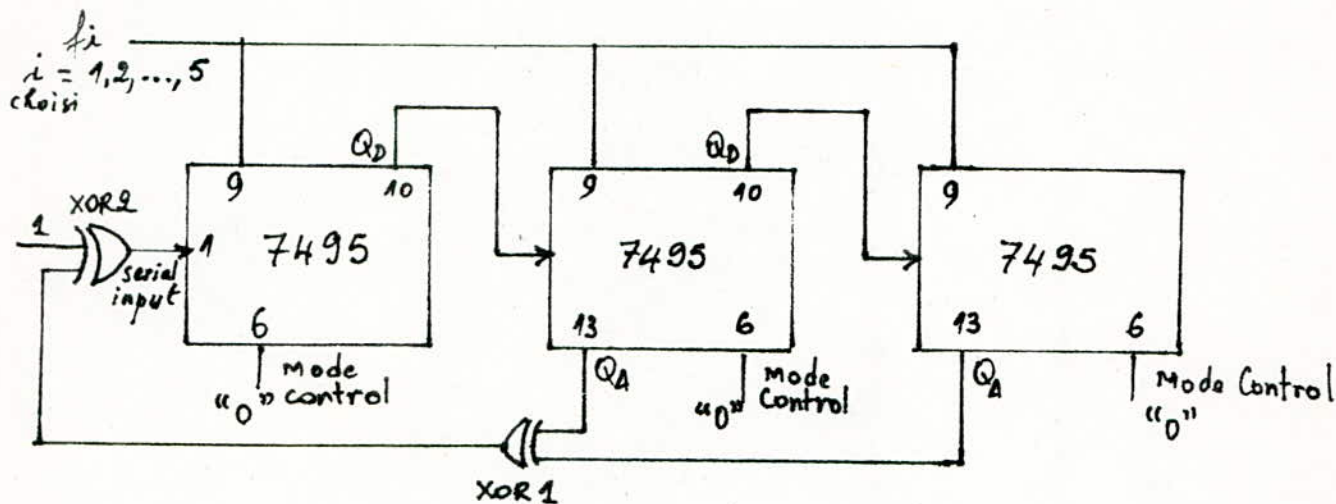


3) L'émetteur

Il s'agit d'un registre à décalage qui bouclé d'une manière particulière, va générer au rythme d'une horloge dont la vitesse est choisie parmi les cinq déjà citées, une séquence pseudo-aléatoire de longueur maximale. On donne son schéma.



Pour ce faire, on utilisera 3 circuits intégrés qu'on branchera comme suit.



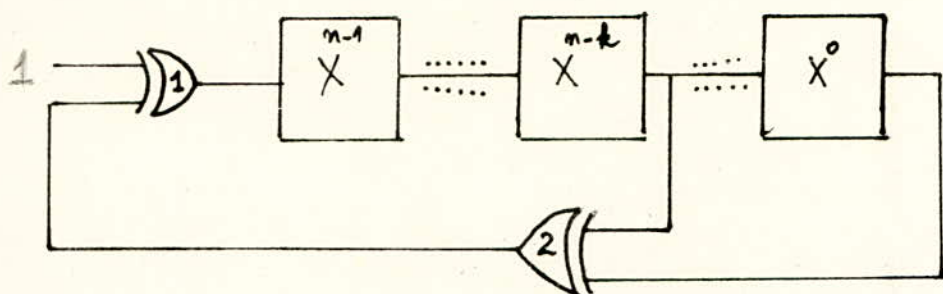
Notre registre à décalage est constitué de 9 bascules. Chaque circuit intégré "7495" contient 4 bascules. Nous en utiliserons donc 3.

La longueur maximale de la séquence pseudo-aléatoire est fonction directe du nombre de bascules D constituant le registre à décalage. Elle s'écrit $2^n - 1$ où n est justement ce nombre. Mais cette longueur maximale ne s'obtient que si les prises de la porte XOR1 sont convenablement choisies. Dans ce cas, les prises sont à la sortie de la 5^{ème} bascule et celle de la 9^{ème} bascule. La longueur de la séquence sera donc de $2^9 - 1 = 511$ bits.

La manière dont la porte XOR2 est bouclé fait que le registre à décalage ne donne jamais "1", ce qui est une valeur stable pour le registre à décalage et donc pas de séquences pseudo-aléatoires.

La notion de séquence pseudo-aléatoire de longueur maximale est liée à la théorie du polynôme.

Tout registre à décalage bouclé sur lui-même peut être représenté par un polynôme qu'on appelle polynôme générateur : exemple $x^n + x^k + x^0$ ou n est le nombre de bascules D constituant le registre à décalage et les sorties des bascules X^0 et X^{n-k} constituent les prises de la porte XOR 2 comme le montre le schéma suivant :

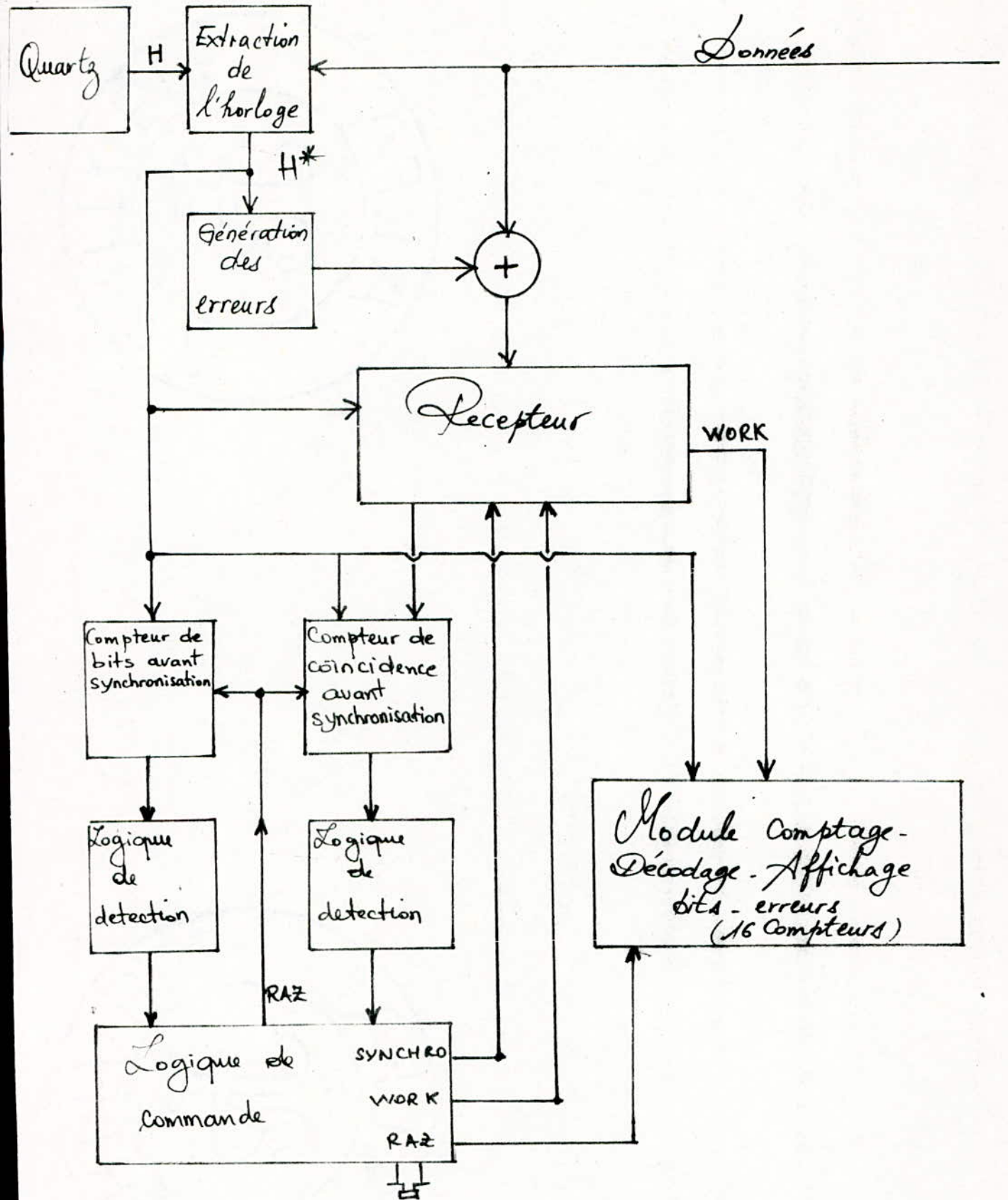


Pour le cas étudié, le polynôme générateur s'écrit $x^9 + x^4 + x^0$ donc 9 indique bien le nombre de bascules et les prises de la porte XOR N° 2 sont comme indiquées à la figure ci-dessus aux points X^0 et X^{n-k} avec $k=5$.

Donc le registre à décalage bouclé sur lui-même (c.a.d ayant un polynôme générateur), va générer une séquence pseudo-aléatoire. Mais cette séquence ne sera maximale de longueur $2^9 - 1 = 511$ bits que si le polynôme générateur est dit « primitif ».

On appelle un polynôme générateur polynôme primitif, un polynôme de degré n qui n'est divisible par aucun polynôme $Z^l - 1$ ou $l < 2^n - 1$.

III Reception



Reception

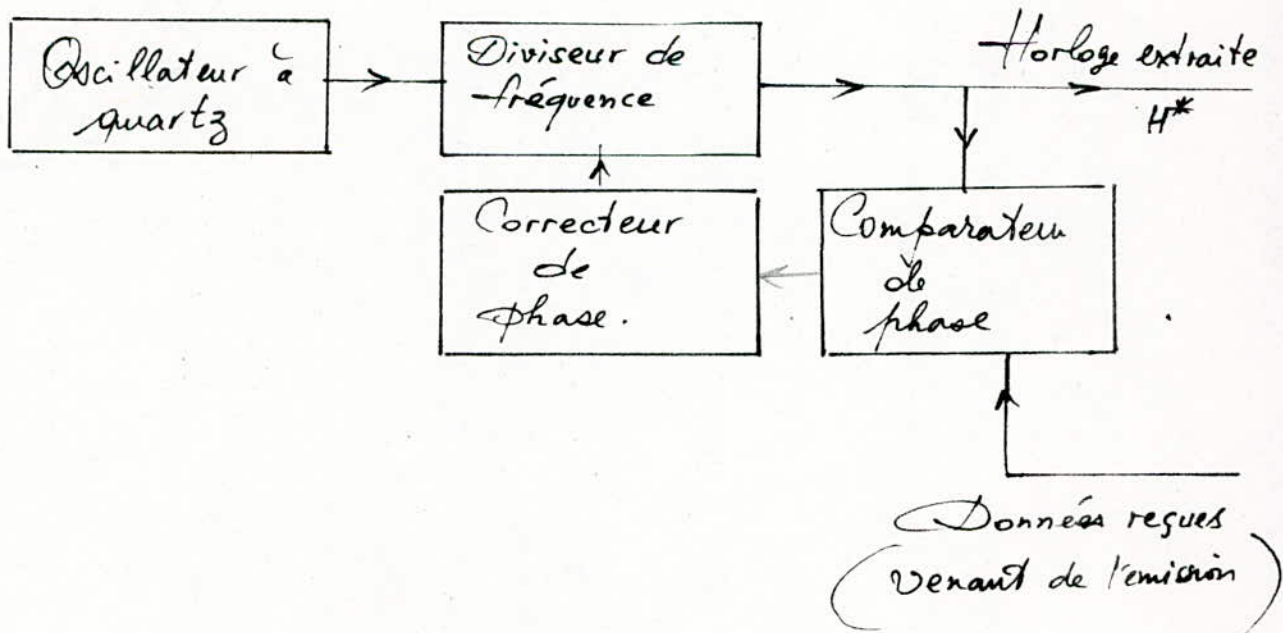
4°) Extraction de l'horloge

Comme le signal d'horloge n'est pas émis, il doit être alors estimé à la réception à partir du signal reçu

Cette méthode consiste à asservir en phase un oscillateur local avec le signal reçu en se basant sur les instants où se produisent les transitions; cela suppose que le nombre de transitions dans le signal reçu est suffisamment grand. Cette condition nécessaire est réalisée à l'émission par l'émetteur qui donne une séquence pseudo-aléatoire de longueur 511 bits conforme aux normes CCITT (Comité consultatif international téléphonique et télégraphique)

L'asservissement en phase est alors d'autant plus meilleur que le nombre de transitions dans le signal reçu est grand.

Le schéma synoptique relatif à l'extraction de l'horloge est le suivant :



a) L'Oscillateur local et les diviseurs de fréquence.

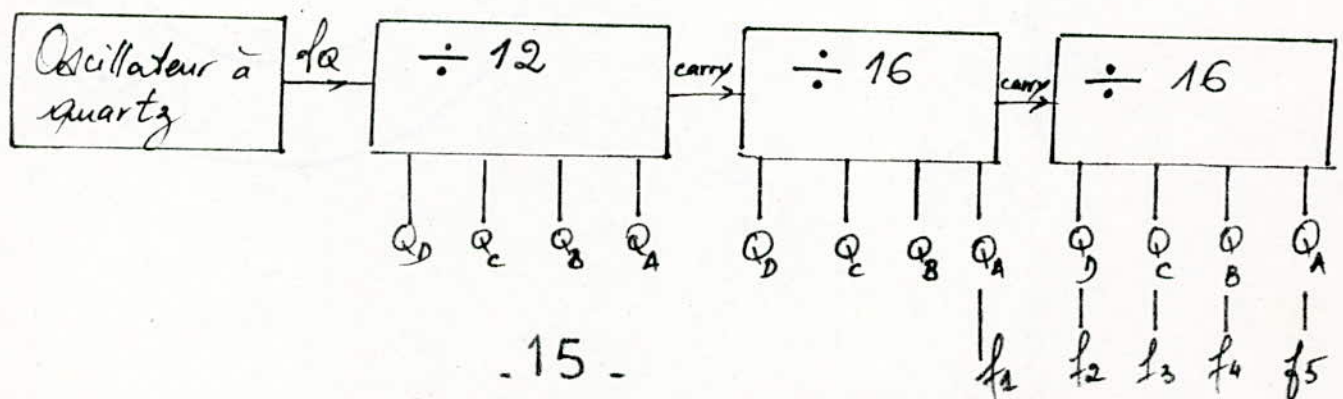
Un oscillateur à quartz identique à celui utilisé à l'émission est placé à la réception. La fréquence du quartz est $f = 1.8432 \text{ MHz}$.

Comme à l'émission, on doit avoir les cinq vitesses de transmission disponibles c'est à dire 9600 Hz ; 4800 ; 2400 ; 1200 ; 600. D'où l'utilisation d'une série de diviseurs de fréquences à partir de celle du quartz comme suit :

$$\begin{aligned}
 1.8432 &\div (12 \times 16) \rightarrow 9600 \text{ Hz} \rightarrow f_1 \\
 1.8432 &\div (12 \times 16 \times 2) \rightarrow 4800 \text{ Hz} \rightarrow f_2 \\
 1.8432 &\div (12 \times 16 \times 4) \rightarrow 2400 \text{ Hz} \rightarrow f_3 \\
 1.8432 &\div (12 \times 16 \times 8) \rightarrow 1200 \text{ Hz} \rightarrow f_4 \\
 1.8432 &\div (12 \times 16 \times 16) \rightarrow 600 \text{ Hz} \rightarrow f_5
 \end{aligned}$$

Donc pour obtenir les cinq vitesses, il nous faut trois (03) diviseurs. Le premier divise par 12, le 2^{ème} et le 3^{ème} divise chacun par 16.

Le schéma synoptique montrant ces diviseurs avec les prises des différentes vitesses de transmission est le suivant :



b) Comparateur de phase.

Il s'agit de comparer le signal reçu des données avec celui de l'horloge locale. Le signal d'horloge étant à 1 des cinq fréquences d'horloge citées précédemment. On choisira la même fréquence d'horloge que celle prise à l'émission.

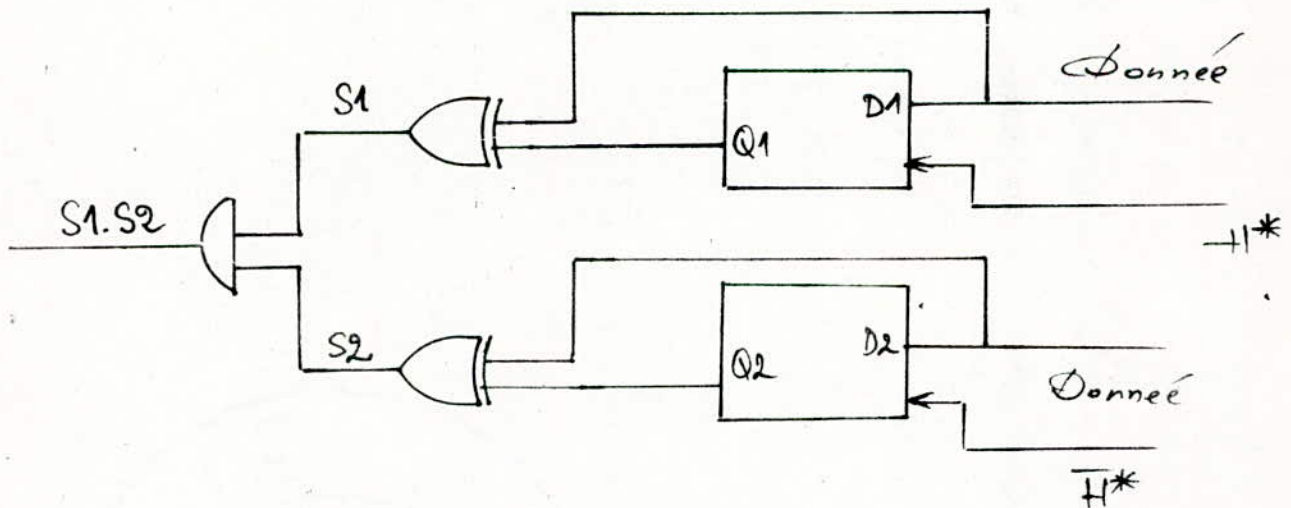
En comparant les deux signaux, trois cas peuvent se présenter.

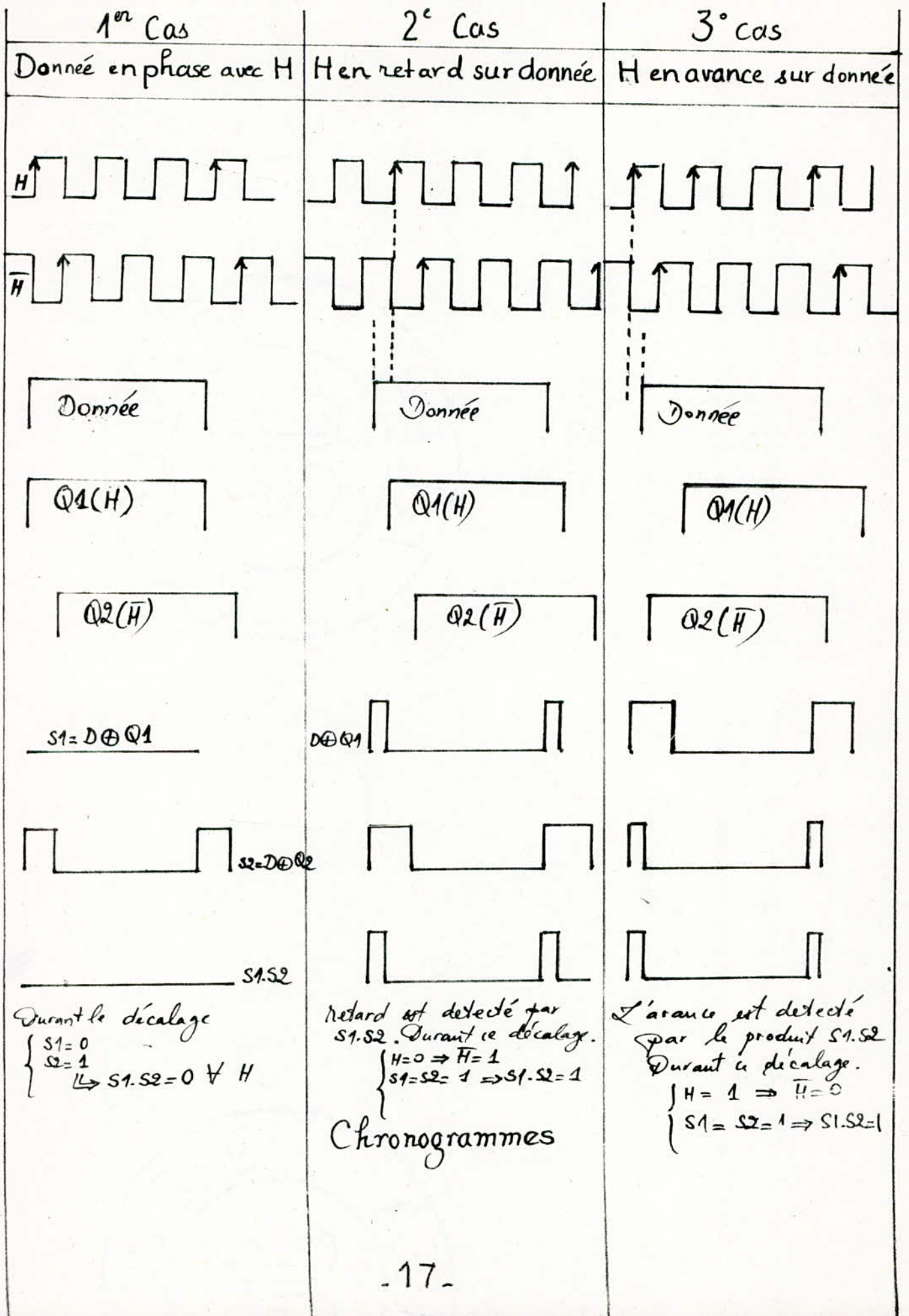
1^{er} cas : Les données sont en phase avec H_{locale}

2^{ème} cas : Les données ne sont pas en phase avec H_{locale} : La donnée est en retard sur H

3^{ème} cas : Les données ne sont pas en phase avec H_{locale} : La donnée est en avance sur H .

Pour sélectionner le retard ou l'avance : On propose le schéma suivant :





c) Correction de phase.

La correction de phase se fait par une logique qui vient après la logique de comparaison et qui attaque le premier compteur diviseur

La correction se fait par ce compteur. Donc ce compteur doit être programmable

1^{er} Cas : Les deux signaux sont en phase, On ne corrige rien,

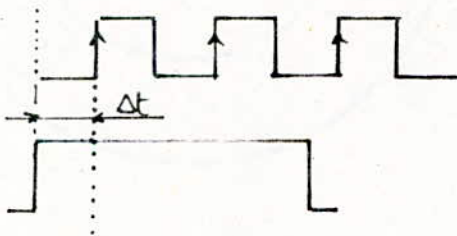
Le premier compteur reste diviseur par 12 comme ça été prévu.

Et pour qu'il compte jusqu'à 12, il faut qu'il indique par ses entrées parallèles la valeur binaire 0100 (4 en binaire).

D C B A

Dans ce cas on a d'après les chronogrammes $S1.S2=0 \forall H$.

2^{ème} Cas : Les deux signaux sont déphasés : H est en retard sur la donnée, comme montré ci-dessous.



On doit diminuer en durée la période de H ou encore augmenter la fréquence de H. Cela se fait à l'aide du compteur programmable diviseur n° 1, Augmenter la fréquence de H revient à diviser la fréquence du quartz ^{par} un nombre plus petit que celui choisi quand il y a synchronisation entre les deux signaux le signal d'horloge et le signal information (Donnée). On divisera par 11 donc on programmera les entrées parallèles comme suit

D C B A
0 1 0 1

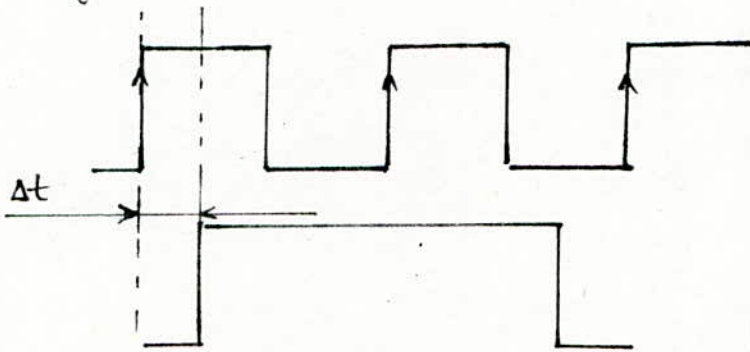
Et la division sera programmée automatiquement quand il y aura retard de H sur la donnée c'est à dire lorsque

$$\left\{ \begin{array}{l} H = 0 \Rightarrow \bar{H} = 1 \\ S1 = 1 \\ S2 = 1 \Rightarrow S1.S2 = 1 \end{array} \right.$$

$S1$ et $S2$ sont égales simultanément à 1 comme ci-indiqué sur les chronogrammes.

3^{ème} cas.

Les deux signaux sont déphasés : H en avance sur la donnée



Par qu'il ait coïncidence entre H et la donnée, il faut que H soit augmenté en durée, donc réduite en fréquence. Cela se fait par le premier compteur diviseur programmable. Réduire la fréquence de H revient à diviser la fréquence du quartz par un nombre plus grand que celui choisi quand il y a synchronisation entre les deux signaux H et la donnée. On divisera ici par 13, donc on programmera les entrées parallèles comme suit :

D C B A
0 0 1 1 (3 binaire).

Cette division sera programmé automatiquement quand il y aura avance de H sur la donnée. C'est à dire lorsque:

$$\left\{ \begin{array}{l} H=1 \Rightarrow \bar{H}=0 \\ S_1=1 \\ S_2=1 \end{array} \Rightarrow S_1 S_2 = 1 \right. \text{ c Simultanément}$$

Comme s'est montré sur les chronogrammes précédents:

Récapitulatif.

Pas de déphasage

H	S ₁ .S ₂	D C B A
X	0	0 1 0 0

Retard de H sur la donnée

H	S ₁ .S ₂	D C B A
0	1	0 1 0 1

Avance de H sur la donnée

H	S ₁ .S ₂	D C B A
1	1	0 0 1 1

Condensons les trois tableaux en un seul.

H	S ₁ .S ₂	D C B A
0	0	0 1 0 0
0	1	0 1 0 1
1	0	0 1 0 0
1	1	0 0 1 1

S ₁ .S ₂	0	1
H	0	1
0	0	0
1	0	0

$$D=0$$

S ₁ .S ₂	0	1
H	0	1
0	1	1
1	1	0

$$C = \bar{H} + \bar{S}_1 S_2$$

$$C = H S_1 S_2$$

S ₁ .S ₂	0	1
H	0	1
0	0	0
1	0	1

$$B = H S_1 S_2$$

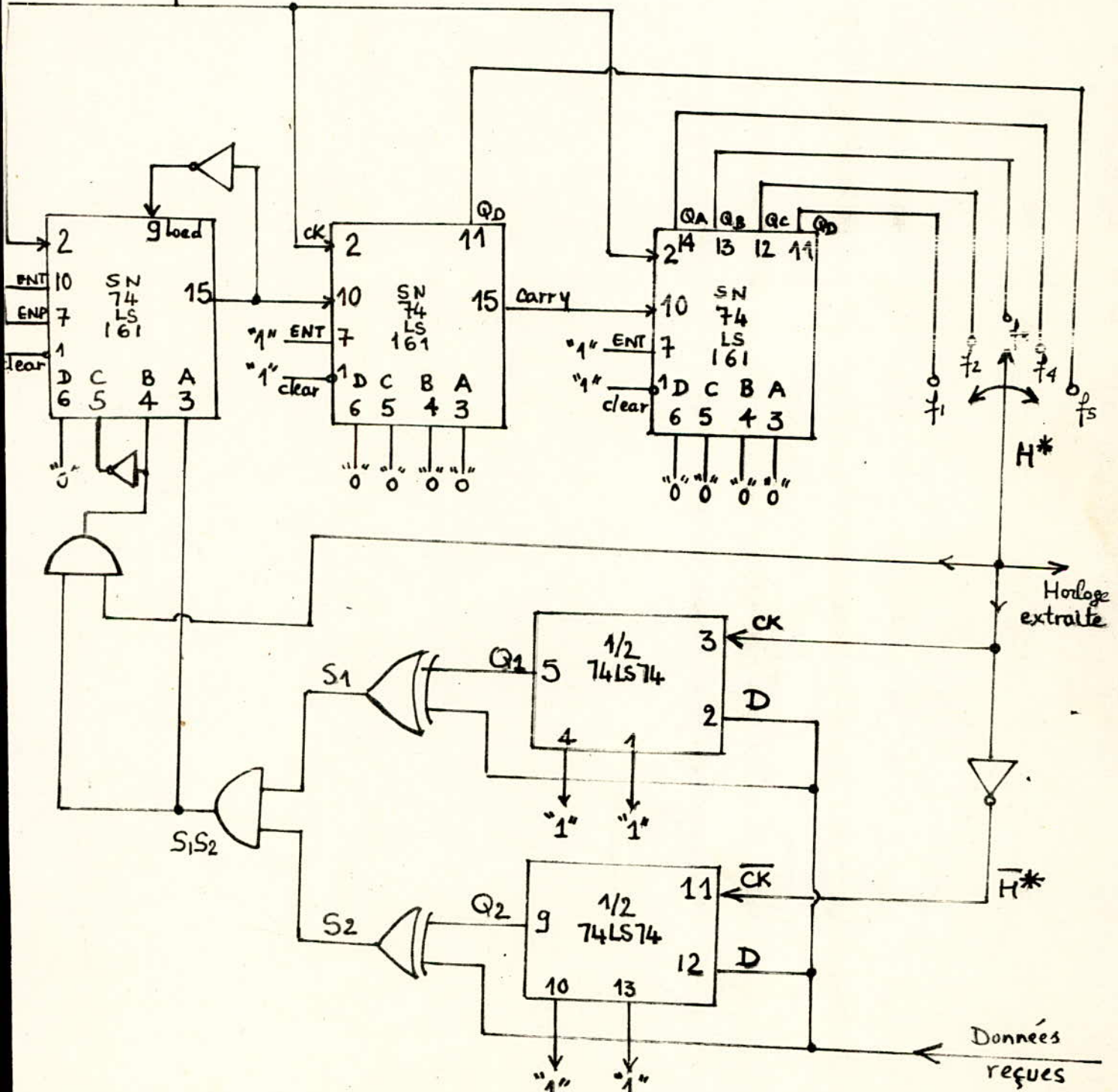
S ₁ .S ₂	0	1
H	0	1
0	0	1
1	0	1

$$A = S_1 S_2$$

Oscillateur
à
Quartz

Extraction de l'horloge.

$f = 1.8432 \text{ MHz}$



2) Génération et injection des erreurs

Les erreurs qui sont généralement introduites par le canal de transmission, viennent s'ajouter au signal constituant les données transmises, d'une manière aléatoire.

Une manière de prouver la fiabilité de ce compteur d'erreurs est de générer des erreurs à un taux fixé connu et d'en faire une addition modulo 2 avec les données reçues. L'addition modulo 2 traduit bien l'effet présence d'erreurs, qui est celui d'inverser le bit information chaque fois qu'il y a erreur.

Le compteur sera jugé fiable, si bien sûr, le rapport du nombre affiché par les compteurs d'erreurs sur celui affiché par les compteurs de bits est le même que le taux d'erreurs déjà fixé.

Pour la réalisation, le taux d'injection d'erreur qui est choisi est de 1 erreur pour 100 bits. Pour cela il suffit tout simplement de détecter le passage de la 100^{ème} impulsion lorsque les compteurs pris pour ce but sont programmés en mode comptage.

En mode décomptage, il faudra détecter le passage du nombre zéro sur les deux compteurs.

Les compteurs choisis sont le MC 94029 des CMOS de Motorola.

Ce ne sont pas des compteurs programmables, mais la présence d'entrées parallèles peut prêter à une confusion. Ces entrées parallèles peuvent être passées en sortie quand la PIN "1" Preset Enable est à 1.

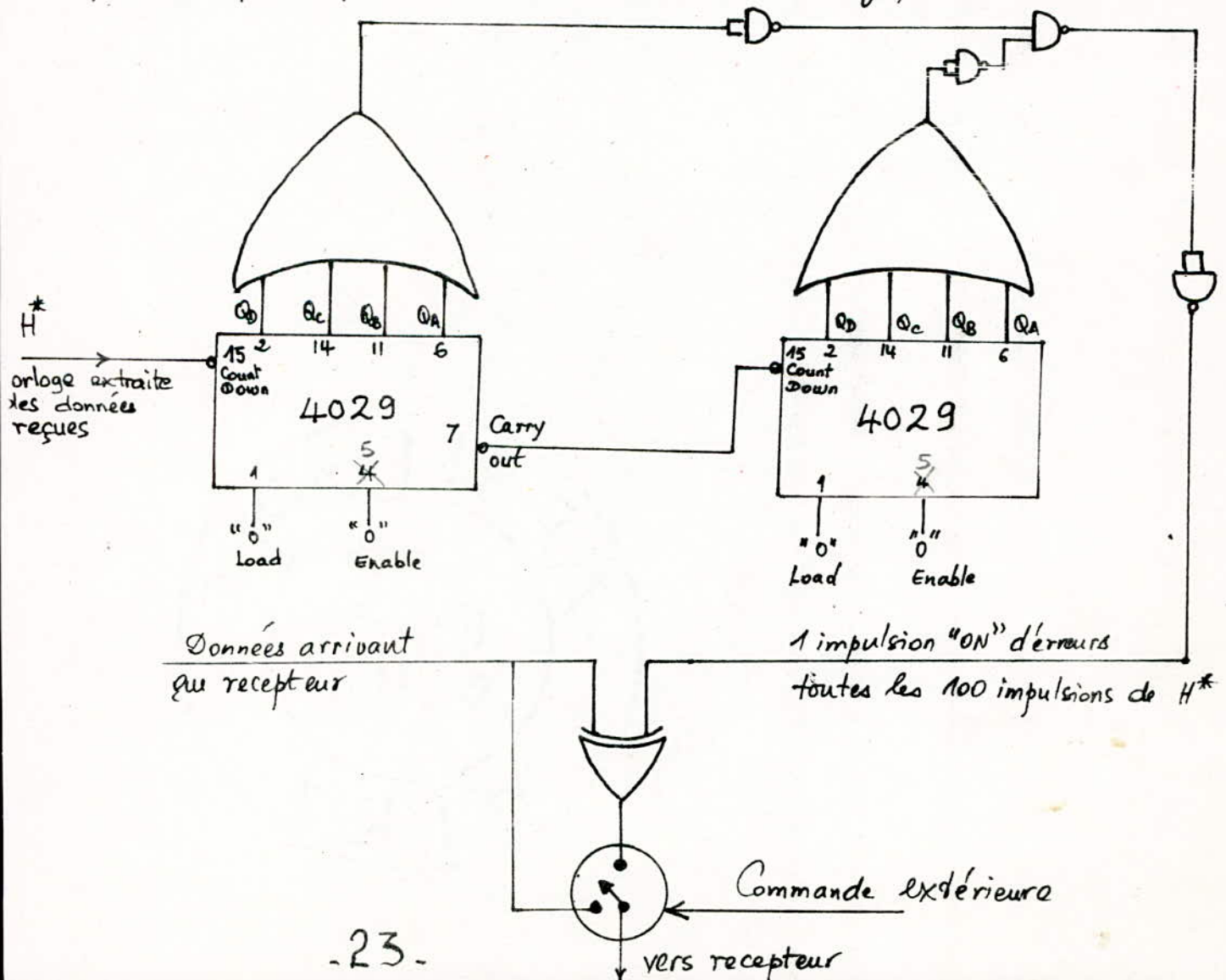
C'est là toute la fonction de ces entrées parallèles.

La PIN "5" (Enable carry in) doit être mise à "0" aussi validant le comptage ou le décomptage. Ces derniers sont justement sélectionnés par la PIN "10" qu'il faut mettre à "1" pour le premier et à zéro pour le second quand à la PIN "9", elle permet de choisir le type de comptage ou décomptage à savoir « Binaire ou décimal » elle est mise à "1" pour le binaire et "0" pour le décimal.

Le choix est fixé sur le mode décomptage donc PIN "10" = 0 (Down) et le type décimal donc PIN "9" = 0.

Donc il s'agit de détecter huit zéros sur les deux compteurs en même temps.

La logique qui donnerait une impulsion toutes les 100 impulsions, cette impulsion qui représenterait l'erreur. La logique est la suivante :



3 / Le receptrer

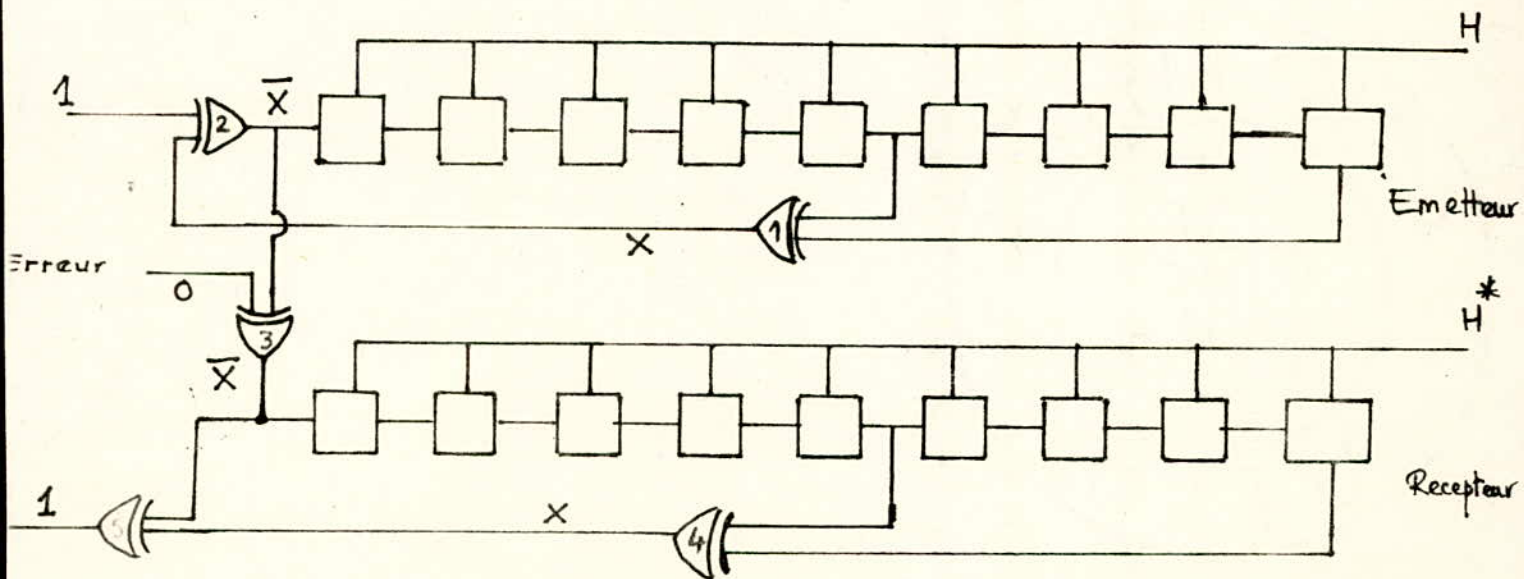
Il est comme l'émetteur, constitué à l'aide de 3 registres à décalage le SN 74 LS 95, chaque circuit intégré contient 4 bascules D, on prendra les 4 bascules D de chacun des deux premiers circuits intégrés et une bascule D du troisième circuit intégré pour ainsi réaliser un registre à décalage à 9 bascules D.

Notre receptrer doit prendre deux configurations différentes comme on l'a déjà expliqué auparavant au chapitre 3 paragraphe 2. C'est à dire qu'il faut dans un premier lieu synchroniser le receptrer et une fois la synchronisation trouvée, le laisser générer des séquences pseudo-aléatoires identiques à l'émetteur en même temps et indépendamment de ce dernier.

a) Configuration "Recherche de synchronisation" qu'on appellera "SYNCHRO"

Le receptrer est câblé comme sur la figure suivante :

On lui a relié directement l'émetteur pour pouvoir facilement expliquer son fonctionnement :



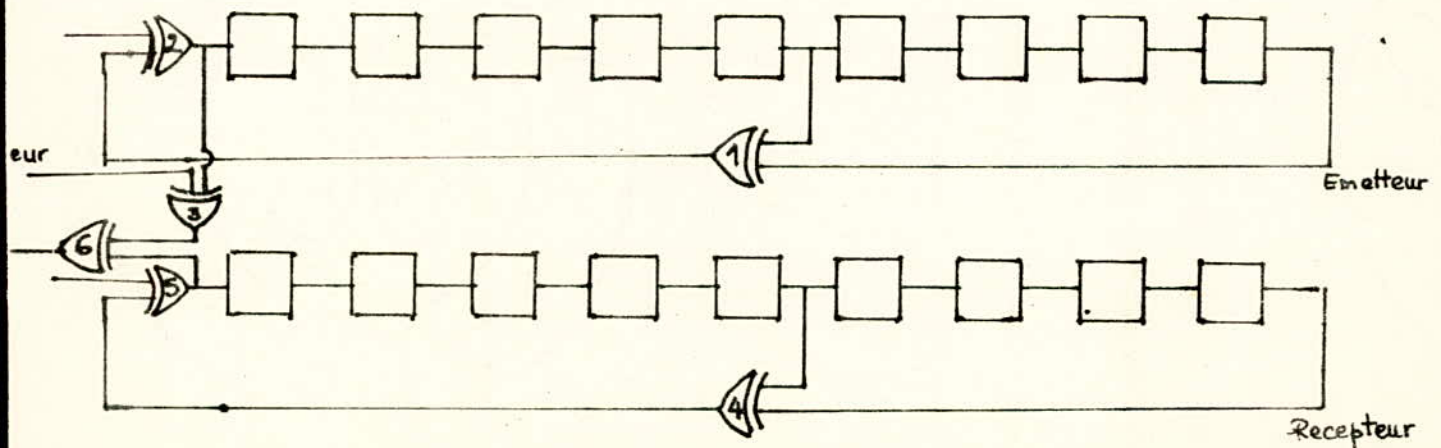
Supposons qu'à la sortie de la porte XOR1, on a une séquence X , à la sortie de la porte XOR2, on aura automatiquement la séquence \bar{X} .

En absence d'erreur, c'est à dire quand l'entrée erreur est égale à zéro "0", on retrouve \bar{X} à la sortie de la porte XOR3. Comme à partir de ce point et jusqu'à la porte XOR4, le récepteur est une copie de l'émetteur d'un point de vue construction, la sortie que l'on obtient à la sortie de la porte XOR4 est la séquence X , à condition qu'il n'y a pas eu d'erreurs durant les neuf impulsions d'horloge passées. Ce qui donnera la valeur "1" à la sortie de la porte XOR5.

En présence d'une erreur, tout s'inverse au niveau du récepteur et on aura un "0" à la sortie de la porte XOR5.

b) Configuration "Comptage d'erreurs" qu'on appellera "WORK"

On ne commence à compter les erreurs qu'une fois la synchronisation entre l'émetteur et le récepteur est réalisée ou en d'autres termes lorsque ce dernier génère indépendamment et en même temps la même séquence que l'émetteur. et quand c'est le cas notre récepteur prendra la configuration suivante :

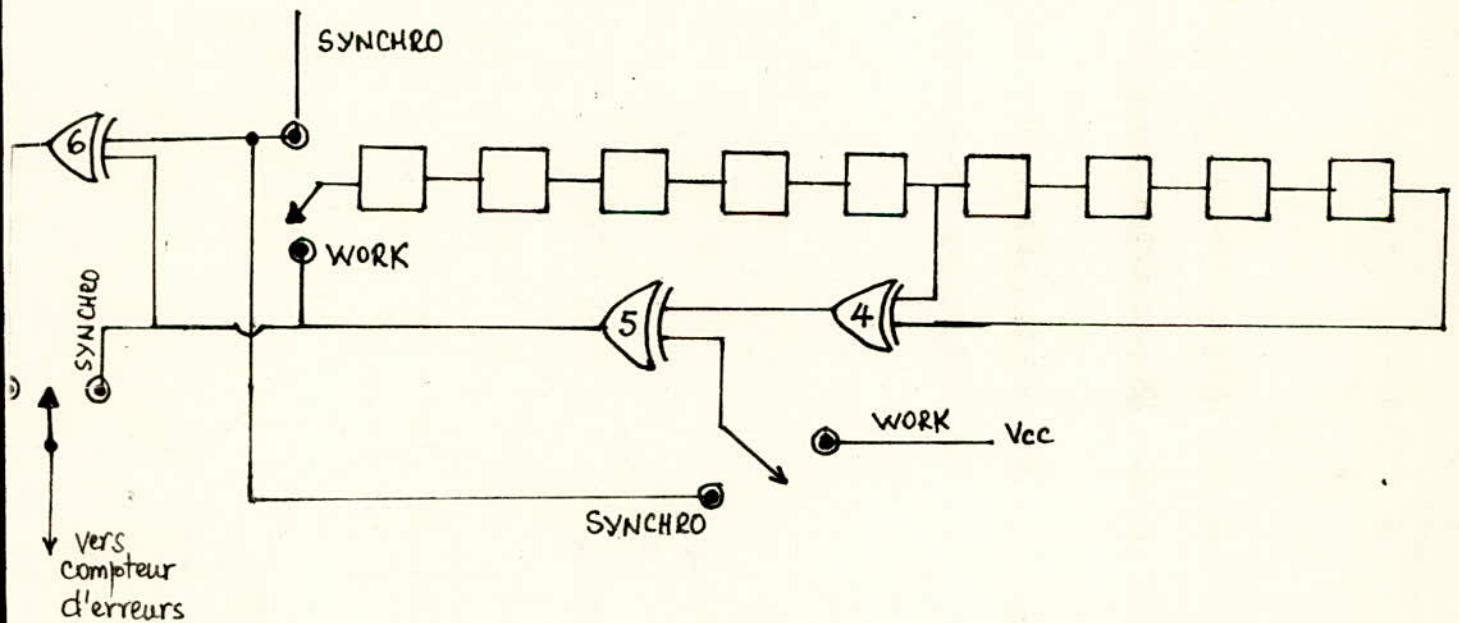


Le comptage des erreurs se fait en comparant la séquence reçue avec la séquence générée localement. Un "1" à la sortie de la porte XOR6 devrait incrémenter le comptage d'erreurs, car cela signifie qu'il y a eu une erreur.

Un "0" à la sortie de cette même porte ne devrait rien changer au contenu du compteur d'erreurs.

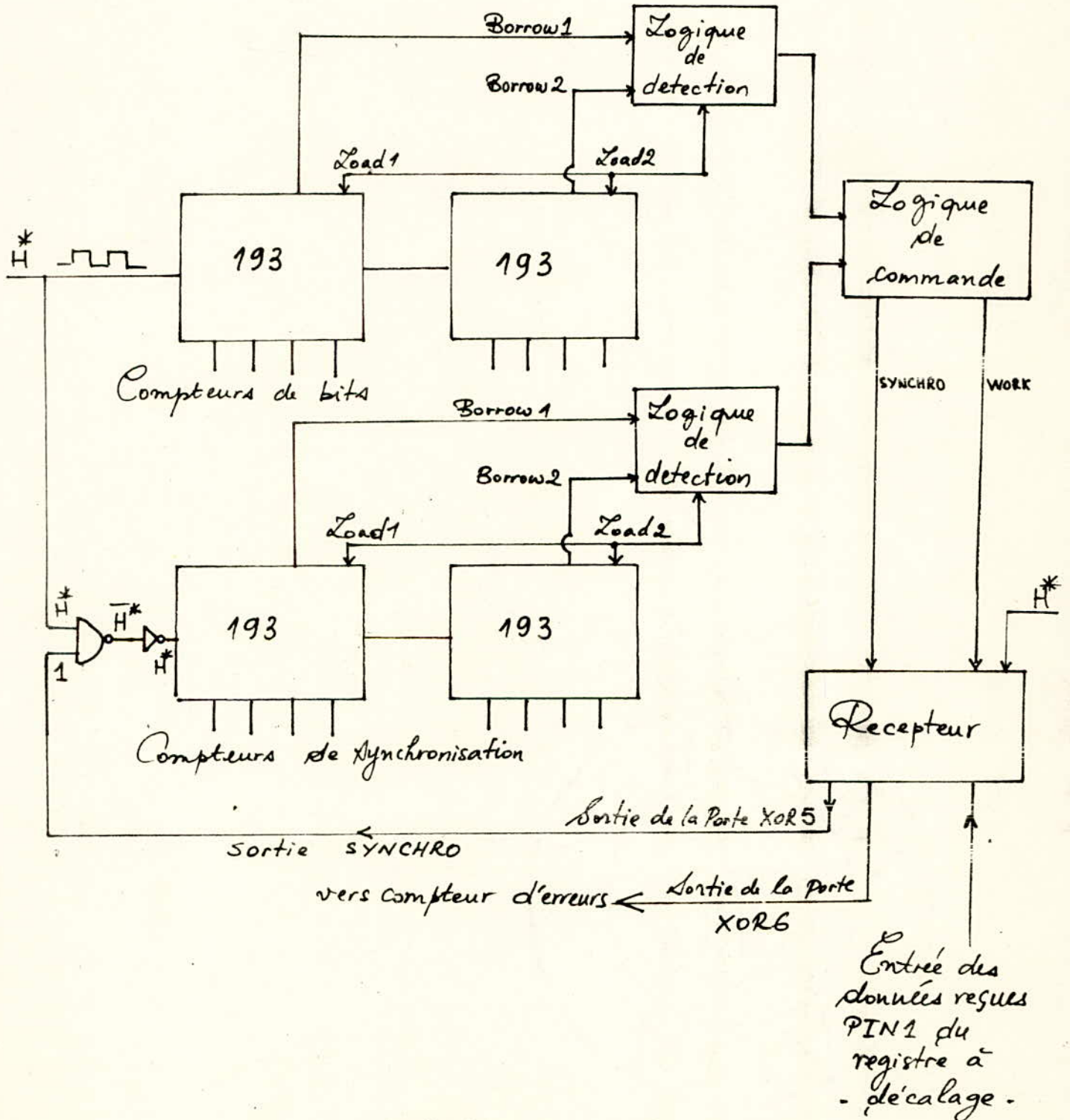
c) Schéma donnant le récepteur dans les deux configurations

Dans ce schéma, on montrera l'emplacement des dispositifs de commutation qui nous permettent le passage d'une configuration à l'autre à savoir la configuration "SYNCHRO" et la configuration "WORK". La figure ou le schéma est le suivant :



4) Logique de detection et de commande

Le schéma synoptique global est donné à la figure suivante.



Il faut avoir une logique qui détecterait l'instant de synchronisation du récepteur avec l'émetteur et qui déclencherait le comptage des erreurs introduites par la voie de transmission dès que cette synchronisation est atteinte, en d'autres termes qui permettrait de faire passer le récepteur d'une configuration vers une autre.

La synchronisation du récepteur dépend directement du nombre de cases c'est à dire du nombre de bascules constituant ce dernier. Quand une erreur est reçue, il faut pour le récepteur ^{lui} autant d'impulsions d'horloge que de cases plus une case pour parcourir le récepteur.

En d'autres termes, comme notre récepteur comporte neuf bascules, donc neuf cases, il faut pour ce dernier 10 impulsions d'horloge après l'arrivée d'une erreur pour se synchroniser.

Donc tant que l'erreur se trouve dans une des bascules constituant notre récepteur, la logique de commande ne doit pas déclencher le compteur d'erreurs.

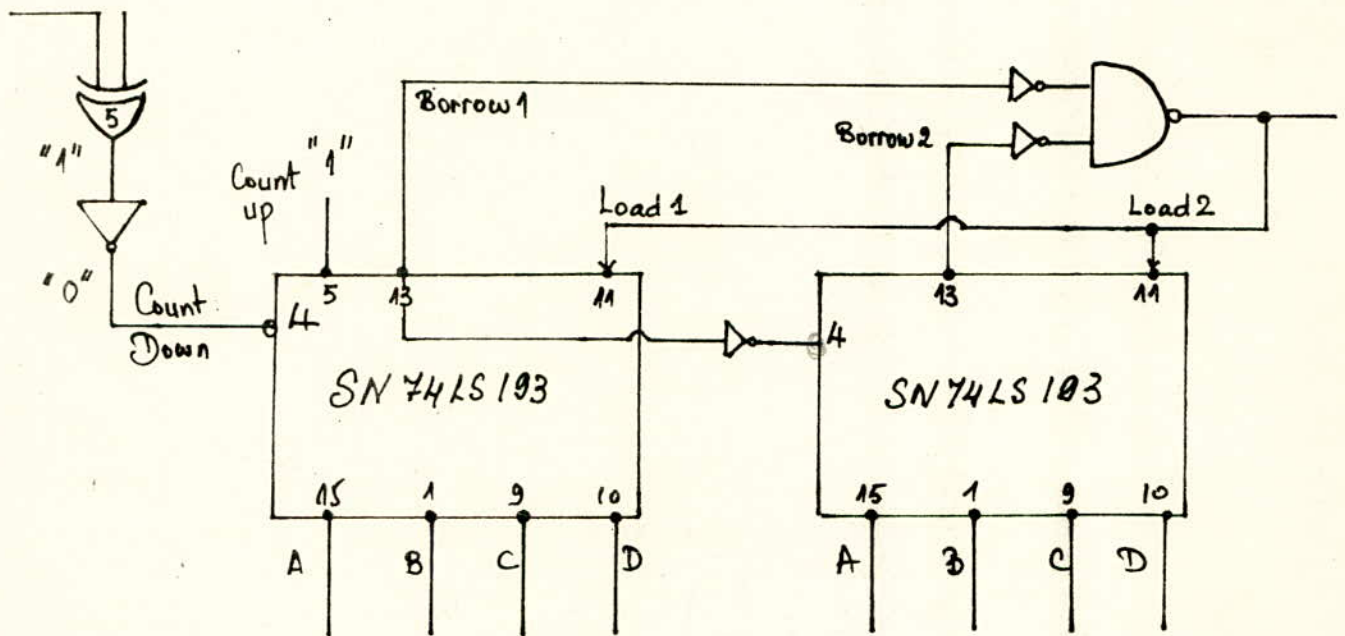
Donc avant la logique de commande proprement dite, il nous faut une logique de détection du nombre 10, c'est à dire la détection de dix (10) "1" Successifs à la sortie de la porte XOR 5.

a) Détection de la Synchronisation.

La détection de dix "1" Successifs signifie que le contenu du récepteur est identique à celui de l'émetteur par conséquent il y a synchronisation. La détection d'un nombre supérieur à 10 de "1" Successifs implique également qu'il y a synchronisation par contre celle d'un nombre inférieur à 10 de "1" Successifs n'implique pas toujours qu'il y a synchronisation.

Il a été prévu pour la détection du "0" ou d'un nombre supérieur à 10, deux compteurs programmables les SN 74LS193 montés en cascade et utilisés en mode décomptage. Pour cela on rentre par la PIN "4" (Count down) et on désactive la PIN "5" (Count up) en la mettant à un niveau "1" logique. Le nombre à détecter est chargé en parallèle et les décompteurs vont se décrémenter à partir de ce nombre jusqu'à la valeur zéro.

La logique de détection du nombre programmé est donné par le schéma de la figure qui suit :



Cette logique de détection a été trouvée comme suit :

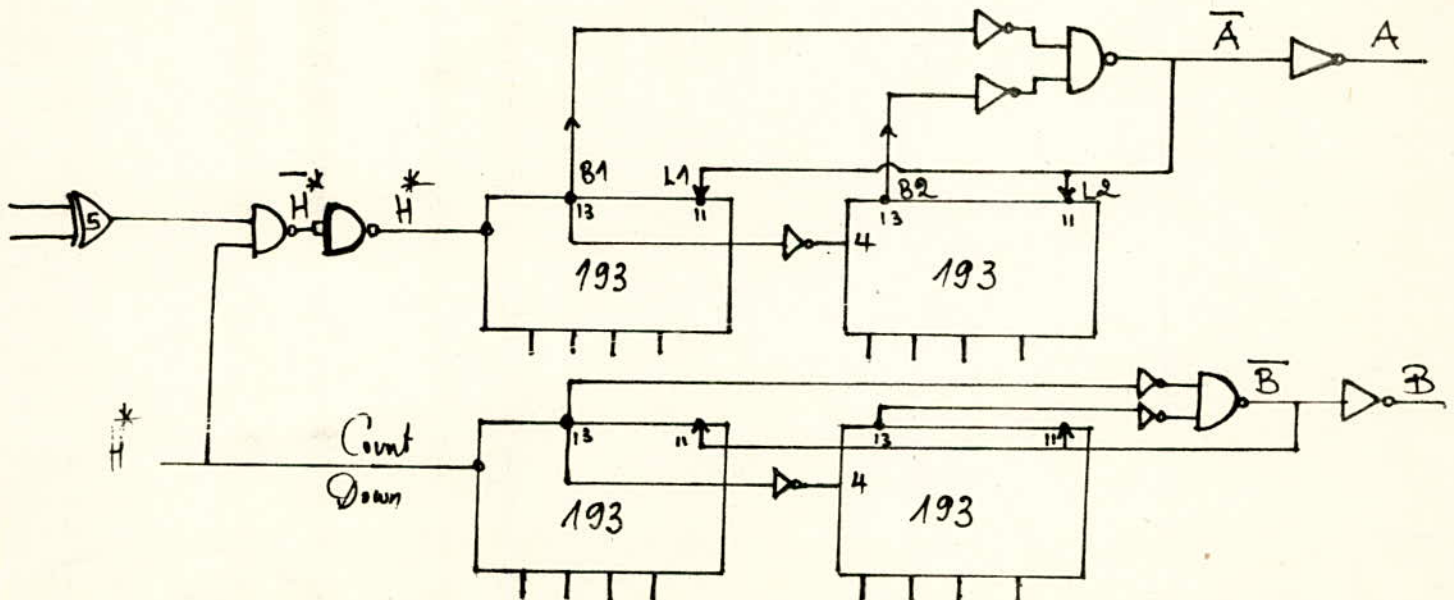
On charge en même temps les deux compteurs par le nombre programmé, donc les deux "load" doivent être reliés ensemble. Et ceci ne peut se faire que quand les deux "barrow" sont actifs en même temps c'est à dire que lorsque nos décompteurs indiquent zéro en même temps. Il reste à savoir quand est ce que ces signaux sont actifs. Ceci est résumé par la table de vérité qui suit :

Borrow 1 B1	Borrow 2 B2	Load 1 = Load 2 = L
0	0	0 Actif
0	1	1 Non actif
1	0	1 Non actif.
1	1	1 Non actif

$$L = \overline{B_1 B_2}$$

Comme L est actif au front montant ; on a $\boxed{L = \overline{B_1 B_2}}$

C'est bien cette logique qui est représentée sur la figure précédente. Mais pour affirmer que le nombre N détecté a été obtenu au soude de N "1" successifs. C'est à dire après N impulsions d'horloge, il faut avoir un compteur de bits c'est à dire d'impulsions d'horloge pour pouvoir comparer les deux nombres programmés : le nombre à à la sortie de la porte XOR5 et les impulsions d'horloge. La figure globale de cette logique de detection sera donnée en dessin



b) Logique de commande :

Cette logique viendrait après celle de détection. Son rôle est de commander le récepteur c'est à dire de lui faire changer de configuration selon que la logique de détection ait détecté la synchronisation du récepteur avec l'émetteur ou non.

Pour concevoir cette logique, il faudra tout d'abord localiser toutes les données possibles c'est à dire toutes les entrées, voir leur états possibles pour en faire la table de vérité et suivant un état passé, en déduire l'état futur.

Les entrées possibles sont les points A et B ou $\bar{A} = \text{Load 1} = \text{Load 2}$ des compteurs de synchronisation et $B = \text{Load 1} = \text{Load 2}$ des compteurs de bits. La remise à zéro des compteurs et l'état passé de la sortie qui'il faut identifier, la sortie est bien sûr, soit le comptage d'erreurs, ou la recherche de synchronisation.

J'appellerai la sortie "Comptage d'erreur", sortie "WORK" et la sortie "Recherche de synchronisation", sortie "SYNCHRO".

Comme on ne peut compter les erreurs et rechercher la synchronisation en même temps, les sorties "WORK" et "SYNCHRO" sont complémentaires.

Il s'en suit qu'une seule sortie suffirait pour l'étude. On a qu'à l'inverser pour obtenir l'autre sortie.

Pour les entrées A et B, il vaudrait mieux considérer leur produit qui simplifierait sûrement l'étude.

On donne la table de vérité de la logique de commande :

P	RAZ manuelle	\bar{Q} (work $\bar{}$)	Q^+ (work $^+$)
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Quand la RAZ manuelle est appliquée (RAZ = 1), on ne compte pas les erreurs ($Q^+ = 0$) et ceci quelque soit l'état du produit P et l'état antérieur de Q^- . En d'autres termes quand on applique la RAZ manuelle, tous les compteurs sont mis à zéro et le récepteur prend la configuration Synchro ou Recherche de Synchronisation. C'est le cas de la 3^{ème}, 4^{ème}, 7^{ème} et 8^{ème} combinaison.

Et quand la RAZ manuelle n'est pas appliquée (RAZ man = 0), il faut voir l'état de P aussi bien que l'état de Q^- .

RAZ manuelle = 0

1^{ère} Combinaison $\left\{ \begin{array}{l} P=0 \rightarrow \text{pas de synchronisation} \\ \bar{Q}=0 \rightarrow \text{On ne comptait pas les erreurs} \end{array} \right\} \rightarrow \begin{array}{l} \text{On ne compte pas les erreurs} \\ \Rightarrow Q^+ = 0 \end{array}$

2^{ème} Combinaison $\left\{ \begin{array}{l} P=0 \rightarrow \text{pas de synchronisation} \\ \bar{Q}=1 \rightarrow \text{on était en train de compter les erreurs} \end{array} \right\} \rightarrow \begin{array}{l} \text{On continue car} \\ \rightarrow \text{y avait eu synchronisation} \\ \Rightarrow Q^+ = 1 \end{array}$

5^{ème} Combinaison $\left\{ \begin{array}{l} P=1 \rightarrow \text{y a synchronisation} \\ \bar{Q}=0 \rightarrow \text{on n'était pas en train de compter les erreurs} \end{array} \right\} \rightarrow \begin{array}{l} \text{On commence} \\ \rightarrow \text{à compter} \\ \Rightarrow Q^+ = 1 \end{array}$

6^{ème} Combinaison $\left\{ \begin{array}{l} P=1 \rightarrow \text{y a synchronisation} \\ \bar{Q}=1 \rightarrow \text{On était en train de compter les erreurs} \end{array} \right\} \rightarrow \begin{array}{l} \text{On continue car} \\ \rightarrow \text{y a synchronisation} \\ \Rightarrow Q^+ = 1 \end{array}$

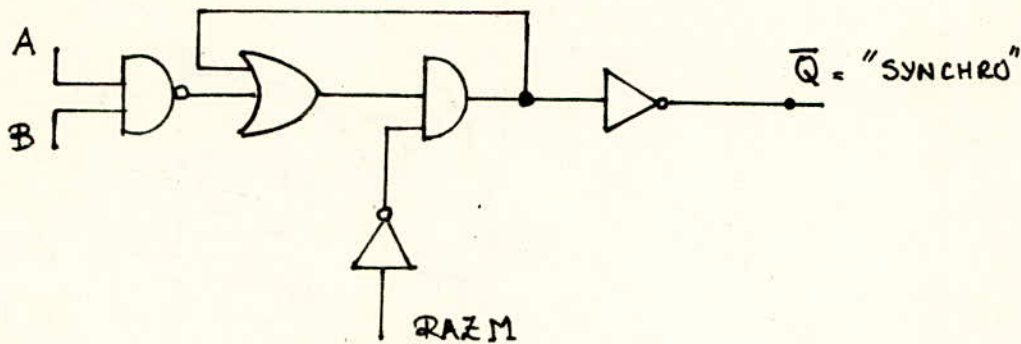
Condensons la table de vérité sous forme de tableau de KARNAUGH pour tirer plus facilement l'expression de Q^+ .

RAZ M P \ \bar{Q}	00	01	10	11
0	0	1	0	0
1	1	1	0	0

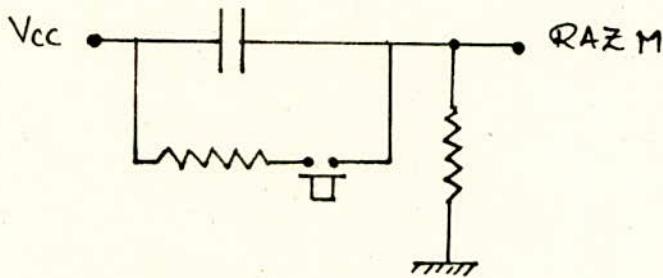
$$Q^+ = P(\overline{\text{RAZ M}}) + (\overline{\text{RAZ M}}) \cdot Q$$

$$Q^+ = \overline{\text{RAZ M}} (P + Q)$$

Ce qui donne le schéma suivant :



La RAZ manuelle peut être représentée par le circuit suivant :

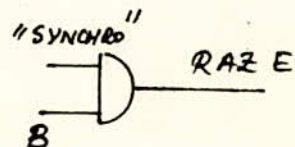


La RAZ est active quand on appuie sur le bouton poussoir, c'est à dire quand elle prend la valeur "1".

D'autre part les compteurs doivent être remis à zéro lorsque le nombre programmé sur le compteur de synchro n'est pas atteint.

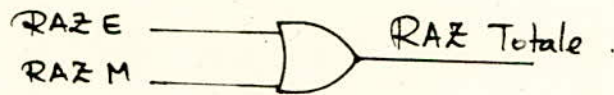
Donc, cette dernière RAZ que l'on appellera RAZ électrique aura pour équation le produit des deux événements qui viennent d'être cités, car il doivent avoir lieu en même temps pour que cette dernière ait un effet finalement

$$RAZE = B \times \text{Synchro}$$

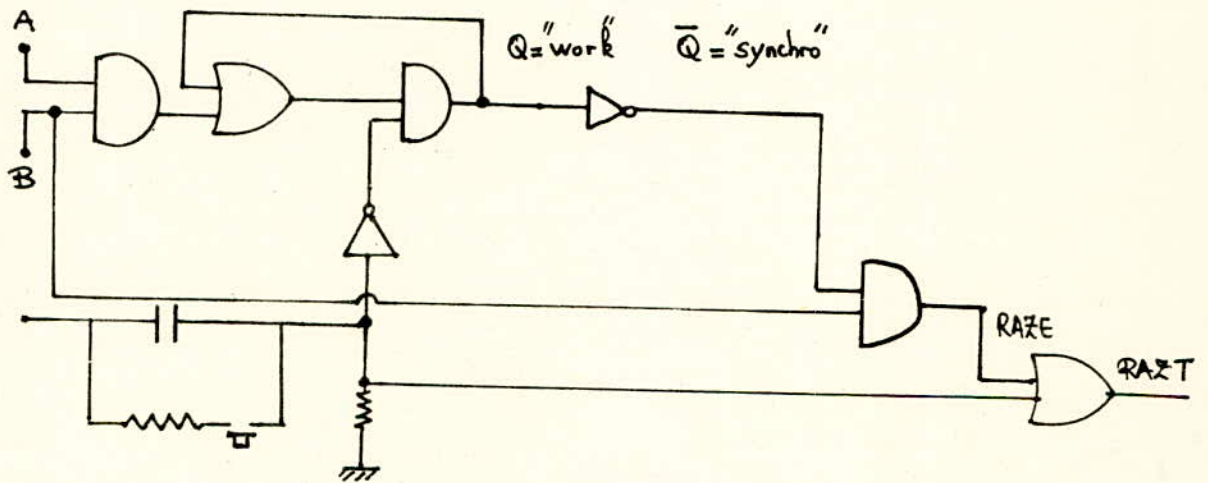


Un dernier point est que chacune des deux RAZ est efficace en agissant toute seule ou avec l'autre. De ceci on tire la relation suivante :

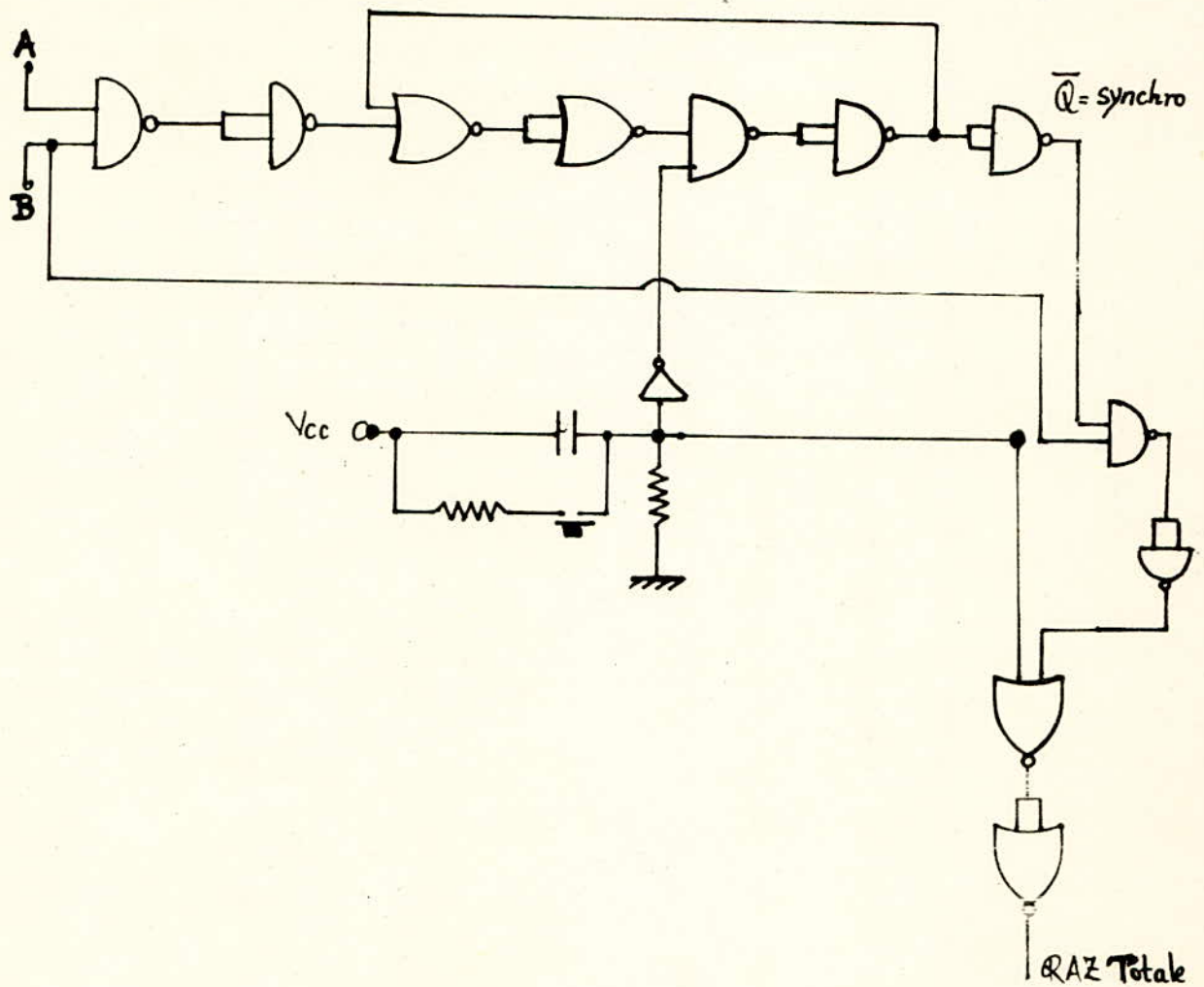
$$RAZ_{\text{totale}} = RAZ_{\text{Electrique}} + RAZ_{\text{manuelle}}.$$



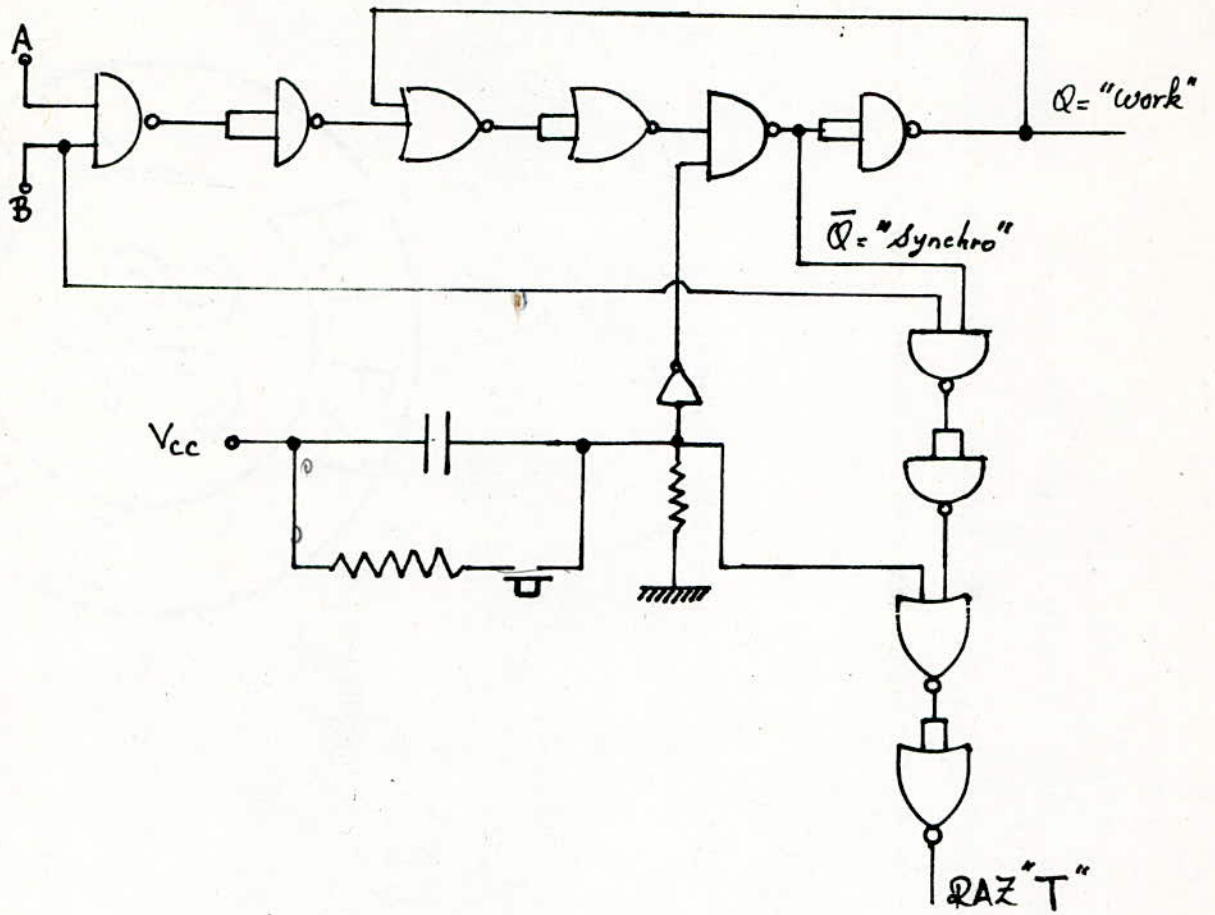
Le schéma global de la logique de commande



Les portes NAND et OR n'étant pas disponibles, le circuit a été réalisé à l'aide de portes NAND et NOR, ce qui nous a conduit au schéma suivant :



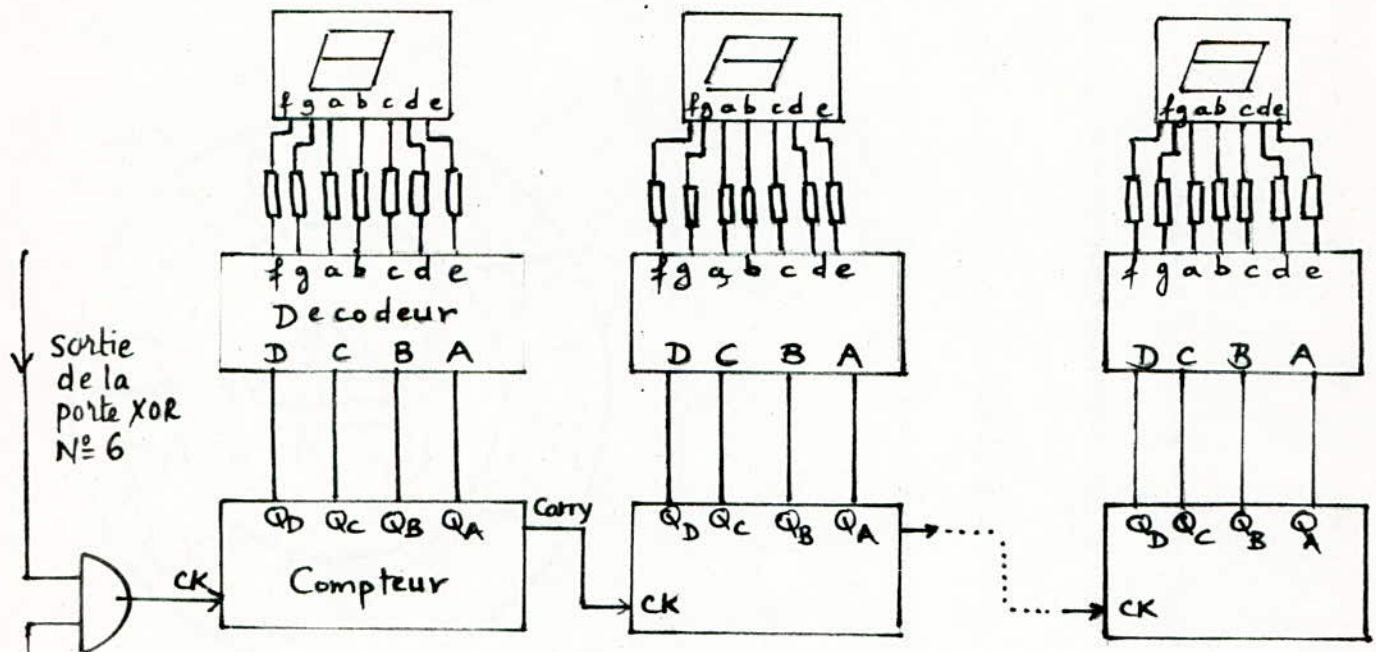
En prenant le point Synchro, un point avant le point "WORK", sur le schéma précédent, on gagnera une porte NAND, ce qui donne le schéma suivant :



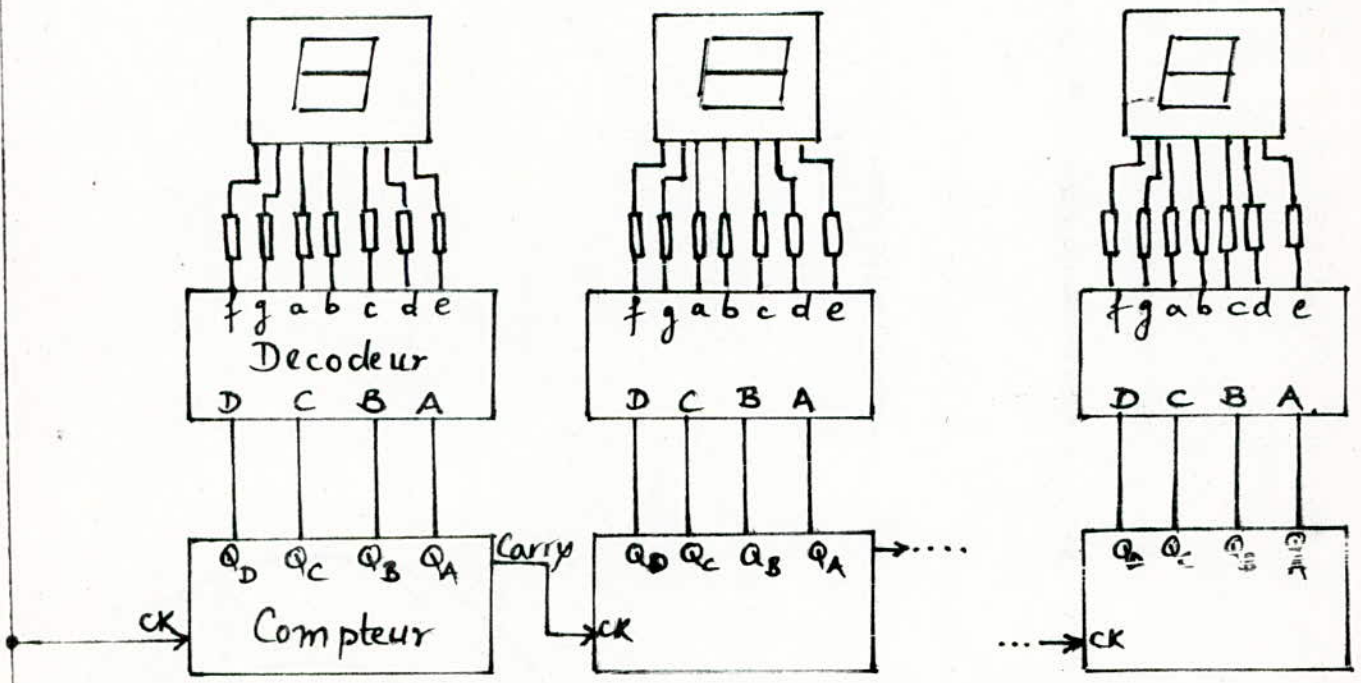
" Schéma définitif
de la logique de commande "

c) Module comptage, sécodage, affichage.

On donne le schéma suivant :



6 Compteurs d'erreurs



H*
Horloge
extraite

8 Compteurs de bits

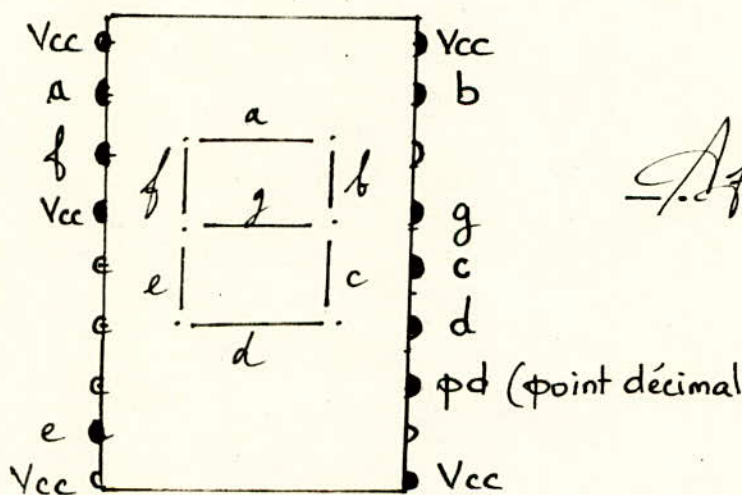
Les circuits intégrés utilisés dans ce module sont :

Compteur : le MC 14029

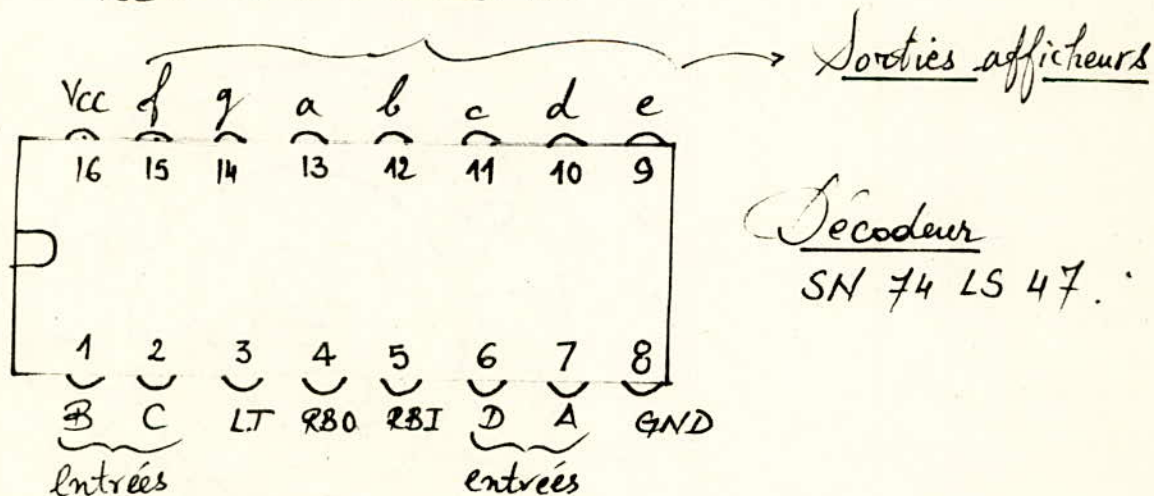
Décodeur : le SN 74LS47

Afficheur : le CQY 84 : 7 segments à anode commune.

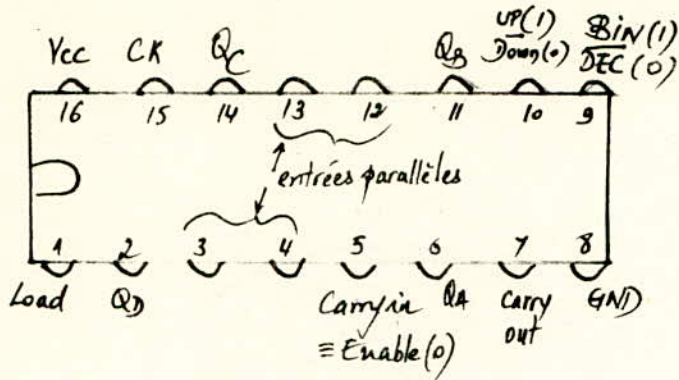
On donne ci-dessous les 3 circuits intégrés avec leur brochage.



Afficheur
CQY 84

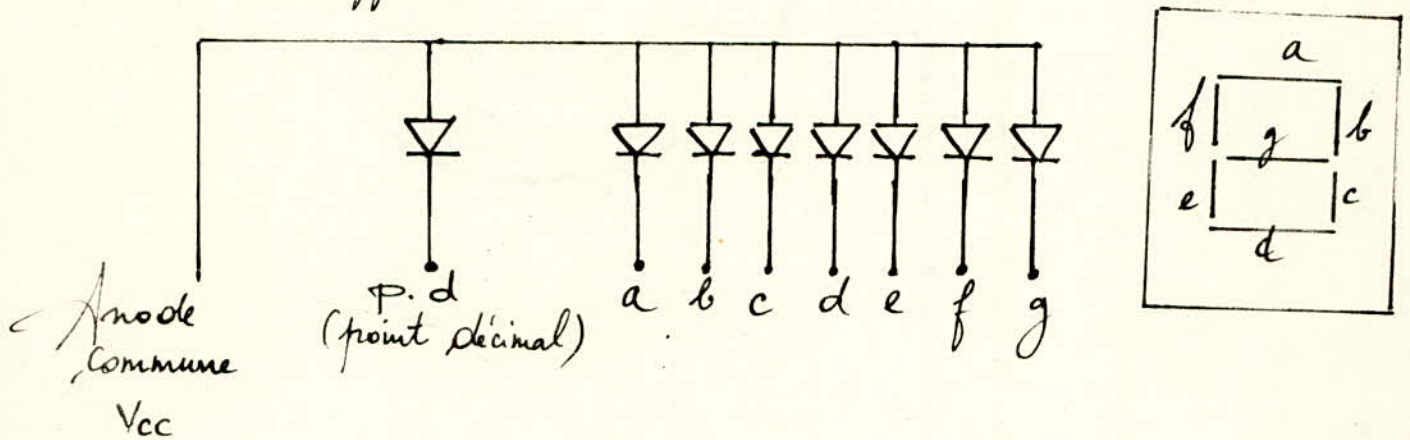


Décodeur
SN 74LS47.



Compteur MC 14029

On donnera dans ce qui suit la constitution intérieure de l'afficheur.



Pour allumer un des bâtonnets a, b, c, d, e, f, ou g de l'afficheur à anode commune, il faut appliquer un zéro logique sur ce bâtonnet.

Donnons la table de vérité d'un afficheur à anode commune. Sur la page suivante :

Table de vérité d'afficheur à Anode Commune

	a	b	c	d	e	f	g
0	0	0	0	0	0	0	1
1	1	0	0	1	1	1	1
2	0	0	1	0	0	1	0
3	0	0	0	0	1	1	0
4	1	0	0	1	1	0	0
5	0	1	0	0	1	0	0
6	1	1	0	0	0	0	0
7	0	0	0	1	1	1	1
8	0	0	0	0	0	0	0
9	0	0	0	1	1	0	0

"0" allumé

"1" éteint

Le décodeur qui convient pour ce genre d'afficheur est le SN 74LS 47, car ce dernier délivre le zéro logique nécessaire aux bâtonnets de s'allumer. En effet les points a, b, c, d, e, f, g du décodeur subissent une inversion à l'intérieur de ce dernier avant d'attaquer l'afficheur à anode commune.

Enfin pour des raisons de protection, des L.E.D des afficheurs, il est indispensable d'insérer en série avec les dernières et les décodeurs, des résistances pour limiter le courant. Les résistances se calculent de la manière suivante :

$$R = \frac{V_A - V_{\text{seuil}}}{I_{\text{direct}}} = \frac{5 - 0,6}{20 \text{ mA}} = 220 \Omega$$

Chaque afficheur - Décodeur, nécessitent 7 résistances, donc il faut de fait qu'il y a 14 couples afficheur - Décodeur
 $14 \times 7 = 126$ résistances.

Conclusion.

Le but fixé pour cette thèse, était de présenter aussi bref que possible une partie théorique et surtout commenter en détails la partie pratique relatives à la conception et à la réalisation d'un compteur d'erreurs dont la capacité peut atteindre le million.

Ce compteur est conçu pour être utilisé dans les transmissions numériques et peut fonctionner à cinq vitesses différentes.

On espère ce but atteint.

Bibliographie

1. Messieurs Benallal Ahmed
Boudjemline Aldja
Conception et réalisation d'un dispositif
d'extraction d'erreurs (Thèse de fin d'études
Juin 1984)
2. M. Louis
Conception et Réalisation d'un modem en
bande de base (Thèse de fin d'études
Année 1983)
3. Larry. C. Shouley, Thomas W. Mc Curnin
Measurement of Bit errors in Data
Transmission Systems I.E.E.E Transactions on
communications (January 1976 page 144-146)
4. J. Clavier, M. Niquil, G. Coffinet, F. Behr
Théorie et technique de la transmission
données (Editions Masson & Cie 1972)
5. J. M. Munier
Introduction à la Télématique.
(Eyrolles 1986)