

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE SAAD DAHLAB BLIDA

INSTITUT D'AERONAUTIQUE

PROJET DE FIN D'ETUDES

Pour l'obtention du diplôme de technicien supérieur en aéronautique

OPTION : AVIONIQUE

THEME



ETUDE ET REALISATION
D'UN CODEUR RADAR
SECONDAIRE

Proposé par : Melle **BENCHEIKH SALIHA**

Présenté par : **GUICHENTI YASMINA**
ABDI HABIBA

SESSION : OCTOBRE 2002

DÉDICACES

Je dédie ce travail :

A ma très chère mère.

A mon cher mari.

A mes chers frères : FETHI , KHALIL .

A mes chères sœurs : WARDA , ISMAHEN, SABIHA .

*A mes amies surtout SOUHILA , RACHIDA , SAMIHA , LILA et
Amina .*

A toute ma famille.

A tous ceux qui ont contribué à ma formation et ma réussite.

HABIBA

DÉDICACES

Je dédie ce modeste travail :

Aux deux êtres sans lesquels je ne serais rien :

Ma très chère mère et mon très cher père.

*A mes chères sœurs : NASSIMA , NAWEL , DALILA , SARAH ,
ABLA .*

A mes chers frères : AHMED , DJAMEL , OKBA .

A mes beaux-frères et mes belles sœurs.

*A mes amies surtout SOUHILA , RACHIDA , SAMIHA , et
RACHIDA .*

A toute ma famille.

A tous ceux qui ont contribué à ma formation et ma réussite.

YASMINA

REMERCIEMENTS

Nous remercions le bon Dieu le tout puissant qui nous a donné la force, le courage et la patience pour achever ce modeste travail.

Nous tenons à remercier également notre promotrice M^{elle} BENCHEIKH SALIHA pour ses conseils et son suivi.

Notre profonde gratitude et reconnaissance vont :

A Mr MEGHELATI ISMAIL.

A Mr BENWARD.

A Mr FAÏD DJALEL.

A Mr RAHIM.

Ainsi qu'aux professeurs de notre institut pour leur aide, leurs conseils, leur gentillesse et surtout leur disponibilité.

MERCI DU FOND DU CŒUR.

GUICHENITI YASMINA.

ABDI HABIBA.

Introduction générale	6
---------------------------------	---

Chapitre I : le radar primaire

I.1 Définition	9
I.2 Principe et fonctionnement du radar primaire	9
I.2.1 Principe général du radar primaire	9
I.2.2 Fonctionnement du radar primaire	9
I.3 Fonctionnement de l'installation du radar primaire	10
I.4 Inconvénients du radar primaire	11

Chapitre II : le radar secondaire

II.1 Introduction	13
II.2 Généralités sur le principe du radar secondaire	13
II.2.1 Définition	13
II.2.2 Principe général du radar secondaire	14
II.3 Fonctionnement d'une station radar secondaire	14
II.3.1 Fonctionnement de l'installation sol	15
II.3.2 Fonctionnement de l'installation bord	16
II.4 Réponses parasites	18
II.4.1 Les échos de sol	18
II.4.2 Réponses asynchrones	18
II.4.3 Réponses par lobes secondaires	20
II.5 Le radar mono impulsion	24
II.6 Le radar mode S	25
II.7 Le codeur	27
II.7.1 La fonction codage	27
II.7.2 Formation des modes d'interrogation	27
II.7.3 Entrelacements des modes	28
II.7.3.1 Entrelacement au rythme de récurrence	28
II.7.3.2 Entrelacement au rythme de tour d'antenne	29
II.7.4 Réalisation du codeur	29

Chapitre III : Présentation du microprocesseur 8085

III.1 Définition d'un système à microprocesseur	31
III.2 Architecture d'un système à microprocesseur	31
III.2.1 L'unité arithmétique et logique	33
III.2.2 L'unité logique de commande	33
III.2.3 Les registres	33
III.2.4 Les bus de données et d'adresses	34
III.3 Les cycles du microprocesseur 8085	34
III.3.1 Cycle d'horloge	34
III.3.2 Cycles machines.	34
III.3.3 Cycles d'instructions	35

TABLE DES MATIÈRES

III.4	Stratégies d'accès aux entrées/sorties	35
III.4.1	La méthode d'E/S programmées	35
III.4.2	La méthode d'interruption	36
III.4.3	La méthode des accès direct mémoire (DMA)	37
III.5	Les interruptions du microprocesseur 8085	37
III.5.1	Niveaux d'interruption	37
III.5.1.1	Interruption non masquable (TRAP)	37
III.5.1.2	Interruption masquable	37
III.5.1.3	Interruption INTR	38
III.5.2	Traitement des interruptions	38
III.5.3	Les instructions :SIM et RIM	38
III.5.3.1	L'instruction SIM	39
III.5.3.2	L'instruction RIM	40
Chapitre IV : Conception de la carte codeur		
IV.1	Circuit d'horloge	42
IV.2	Circuit de réinitialisation	43
IV.3	Circuit du démultiplexage du bus d'adresses	43
IV.4	Bufferisation des bus	45
IV.4.1	Bufferisation du bus de données	46
IV.4.2	Bufferisation du bus d'adresses	46
IV.5	Décodage du bus de contrôle	48
IV.6	Interfaçage des circuits périphériques avec le microprocesseur	49
IV.6.1	Interfaçage de l'EPROM	49
IV.6.2	Interfaçage de la RAM	51
IV.6.3	Interfaçage du clavier	53
IV.6.4	Interfaçage des monostables	55
IV.7	Fonctionnements des monostables	56
Conclusion générale		59
Annexe A		60
Annexe B		65
Bibliographie		70

INTRODUCTION GENERALE

Le contrôle aérien a 3 missions principales : sécuriser, Accélérer et Optimiser le trafic aérien. Les services de contrôle de la circulation aérienne ont donc la charge d'assurer le bon déroulement des vols dans les meilleures conditions de sécurité et de régularité.

La fourniture d'une image représentative de la situation aérienne dans une zone géographique donnée contribue à satisfaire ces deux exigences.

Depuis 1960, deux moyens complémentaires, coimplantés pour les premières générations, permettent d'élaborer cette image : le **radar primaire** de surveillance et le **radar secondaire** de surveillance. Tous deux ont pour fonction de détecter et de positionner les objets évoluant dans l'espace aérien.

Le principe du radar primaire est basé sur la transmission des vagues d'impulsions, de leur réception et du traitement du signal reflété par la surface de l'avion. Ce système permet à la position d'avion d'être déterminée dans la distance et l'azimut. Le radar primaire n'exige aucune collaboration de l'avion.

Le radar secondaire localise l'avion selon le même principe que le radar primaire mais la vague transmet les interrogations périodiques et codées qui activent l'équipement aéroporté (transpondeur), celui-ci répond au radar et transmet un code d'avion assigné par commande de trafic aérien, aussi bien que son altitude. Ce radar demande la collaboration de l'avion.

L'un des principaux éléments du radar secondaire est le codeur, Son rôle essentiel est d'élaborer, au rythme de la fréquence de répétition, les impulsions de mode constituant les interrogations de la cible. Ces impulsions ainsi générées serviront à moduler un émetteur fonctionnant à la fréquence de 1030 MHz.

L'objet du présent travail, consiste à l'étude d'un codeur radar secondaire en réalisant une carte électronique à base de microprocesseur à 8 bits (le 8085 d'INTEL) qui nous permet de générer et d'entrelacer les modes d'interrogation du radar secondaire.

Le mémoire est composé de 4 chapitres :

Le chapitre I présente une introduction concernant la radio-détection, le radar primaire, son principe de fonctionnement et ses inconvénients, d'où la nécessité du radar secondaire.

Le chapitre II est consacré à l'étude du fonctionnement du radar secondaire, les réponses parasites, les différents types de radar secondaire (le radar mono impulsion et le radar mode S) et le fonctionnement détaillé du codeur.

Le chapitre III comporte les principes de base du fonctionnement du microprocesseur 8085 d'INTEL (architecture, l'UAL, l'ULC, les registres, les bus de données et d'adresses, les entrées/sorties et les interruptions...)

Le dernier chapitre est destiné à la conception de la carte codeur. Nous étudierons les circuits d'horloge, de réinitialisation et du démultiplexage du bus d'adresses ensuite l'interfaçage des circuits périphériques (RAM , EPROM, clavier et monostables) .

CHAPITRE I

LE RADAR PRIMAIRE

I.1 Définition

L'expression RADAR est un acronyme pour **R**adio **D**etection **A**nd **R**anging, nom de code adopté en 1940 par la Marine américaine, qui se traduit en détection et mesure par moyen radioélectrique. Un radar primaire est un système qui illumine une portion d'espace avec une onde électromagnétique et qui en retour reçoit les ondes réfléchies par les objets qui s'y trouve, ce qui permet de détecter leurs existences et certaines de leurs caractéristiques (leur position par exemple).

I.2 Principe et fonctionnement du radar primaire

I.2.1 Principe général du radar primaire

Un émetteur génère des impulsions de brèves durées, de forte puissance, de fréquence de répétition "F", une antenne fortement directive rayonne le faisceau, en balayant l'espace à surveiller de façon continue. Ce balayage est le plus souvent obtenu par une antenne tournante autour d'un axe vertical. Quand le faisceau rencontre un obstacle, une partie de son énergie est partiellement captée par l'antenne ayant émis des impulsions.

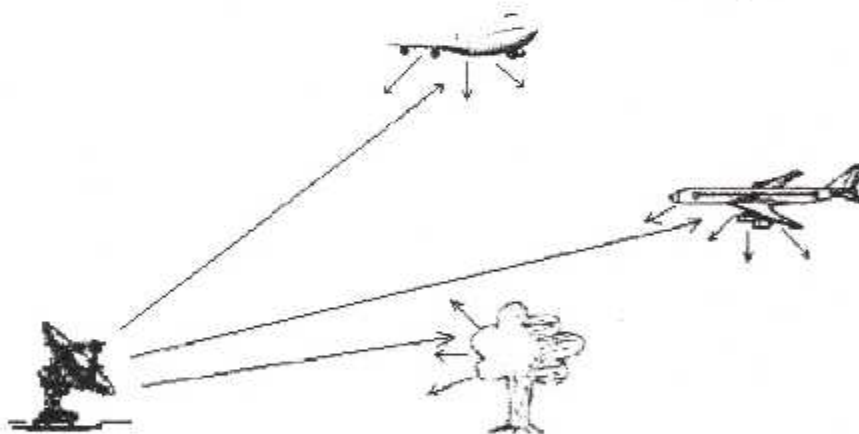


Fig I.1 Schéma de principe du radar primaire

I.2.2 fonctionnement du radar primaire

Les fonctions assurées par le radar primaire sont :

- ✓ **La détection :**
Décision de la présence d'une cible par reconnaissance du signal utile dans le signal reçu.
- ✓ **La mesure de distance :**
Par mesure de temps de propagation du signal sur le trajet radar-cible-radar.
- ✓ **La mesure de vitesse :**
Par les fréquences DOPPLER.
- ✓ **La mesure d'angle :**
Par référence à la position d'une antenne directive en azimut.

I.3 Fonctionnement de l'installation du radar primaire

Un radar primaire comme l'indique la figure I.2 est constitué de :

- Un émetteur qui est chargé de générer, moduler et amplifier les impulsions.
- Une antenne directive utilisée à l'émission comme à la réception.
- Un circulateur, une sorte d'aiguillage permettant d'orienter l'impulsion de puissance vers l'antenne à l'émission, et de diriger les échos vers le récepteur en phase d'écoute.
- Un récepteur assurant l'amplification et la démodulation du signal hyperfréquence en signal basses fréquences dit "vidéo".
- Un extracteur chargé de la détection des échos utiles, de l'élimination des échos parasites et de la formation des plots.
- Un système de traitement de données.

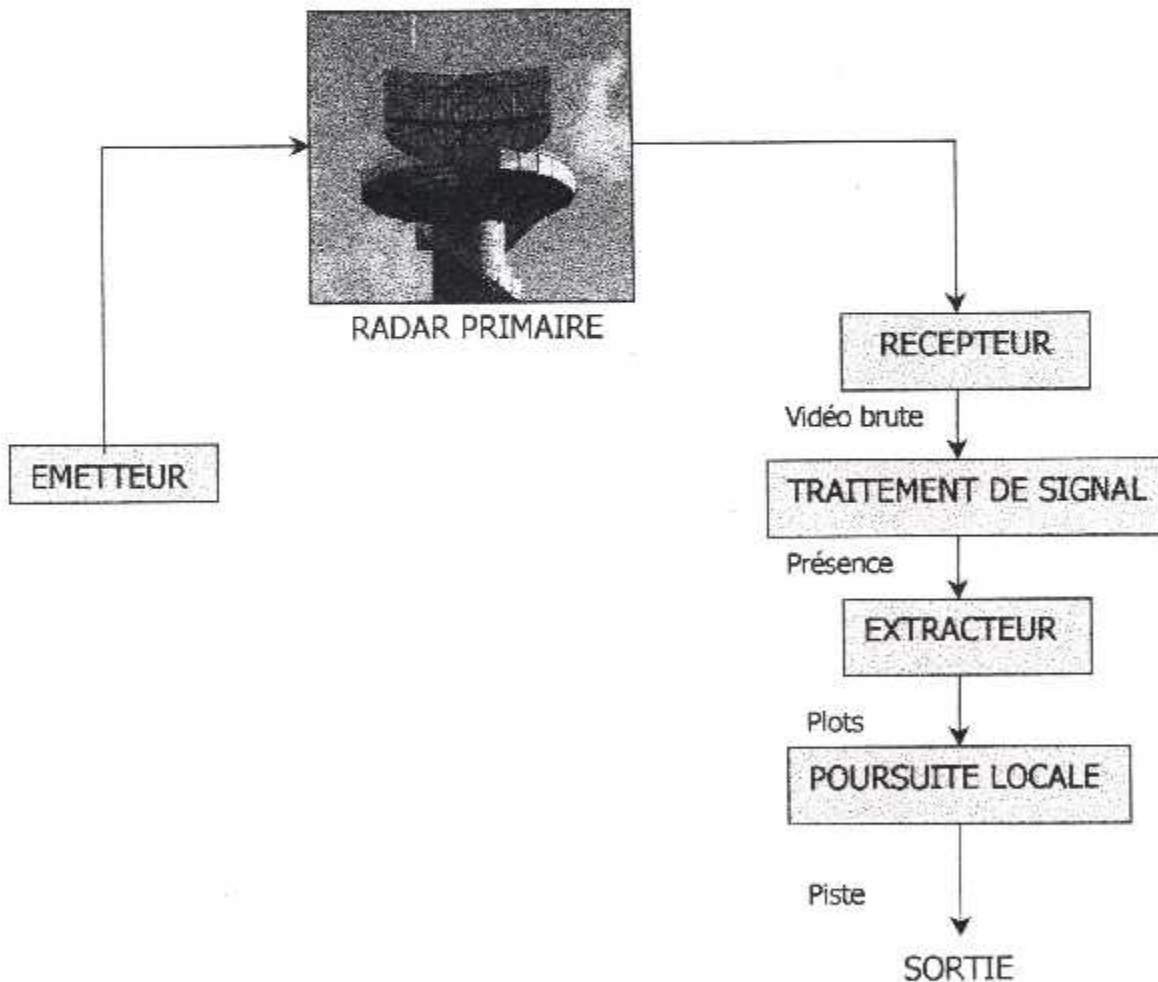


Fig I.2 SCHÉMA SYNOPTIQUE DU RADAR PRIMAIRE

I.4 Inconvénients du radar primaire

Le radar primaire était longtemps considéré comme un outil indispensable dans le domaine de la surveillance aérienne car il peut détecter des cibles très lointaines (donc invisible à l'œil nu) même durant la nuit.

Mais au cours des années, et face à l'augmentation du trafic aérien, le radar primaire a prouvé qu'il est incapable d'assurer avec précision la couverture de l'espace aérien, car il fournit une position de deux dimensions seulement de l'avion, donc ce système ne détecte pas l'altitude. De plus, le radar primaire ne comprend pas ce qu'il voit, il exige ainsi la présence d'un opérateur pour interpréter ce que le radar détecte. Il faut noter que la moitié des objets détectés ne sont pas des avions (des gros oiseaux, des nuages ...).

En terme de fiabilité, les méthodes d'identification employées étaient encombrantes et avec une certaine possibilité d'erreurs graves qui peuvent se produire sans oublier l'effet nocif des réflexions causées par les collines, les obstacles, les tours et les ponts. Dès lors un contact par radio (par fois intensif) avec les pilotes est toujours nécessaire pour vérifier tous les paramètres du vol en question (position, vitesse, coordonnées ...).

Seul, le radar primaire présente une capacité de localisation exhaustive. En contre partie, l'information fournie est relativement pauvre et peu robuste aux fausses alarmes.

Autour de 1950 à 1955, et en raison des problèmes semblables de l'aviation militaire pour distinguer l'avion de l'(ami/ ennemi), un type additionnel de messages a été ajouté à l'information primaire de base existante en employant un autre type d'équipement dit **radar secondaire**.

Ayant été conçu à l'origine pour n'assurer qu'une fonction d'identification à travers un dialogue coopératif entre le capteur sol et un transpondeur installé à bord de l'avion. Le radar secondaire fournit une information plus riche (identification, altitude) et exempte des fausses alarmes.

Ce système exige néanmoins la présence d'un transpondeur actif à bord et peut donc se révéler indisponible pour les aéronefs non équipés ou dont le transpondeur est défaillant.

CHAPITRE II

LE RADAR SECONDAIRE

II.1 Introduction

Un radar sert, comme son nom l'indique, à détecter une cible dans un espace donné et à déterminer sa position, à l'aide d'ondes électromagnétiques. Le radar balaie donc une portion de l'espace avec un faisceau d'ondes électromagnétiques émis par une antenne qui tourne verticalement sur elle-même. Il reçoit en retour les ondes réfléchies par les objets qui se trouvent dans la zone explorée. Signaux très faibles, souvent noyés dans des bruits parasites, qu'il faut amplifier à la réception avant de pouvoir les traiter et les utiliser. Ce qui se traduit par un écho sur un récepteur, généralement un écran cathodique.

Comme on connaît à chaque instant la position du faisceau, on détermine la position de l'objet par rapport à l'émetteur. Pour déterminer la distance, il suffit de mesurer le temps écoulé entre l'émission de l'onde et l'apparition de l'écho, sachant que les ondes électromagnétiques se déplacent à la vitesse de la lumière (300 000 km/s).

Vers 1957, aux Etats-Unis, puis en 1959 en France, le récepteur, associé à un calculateur, devint capable de discerner avec une certitude absolue le signal reçu, même au sein de bruits parasites. Puis - nouvelle amélioration - de fournir au contrôleur uniquement les échos utiles, en effaçant par exemple les échos provoqués par des obstacles situés au sol, autour de l'antenne du radar (lignes à haute tension, bâtiments, etc.).

Sur son écran radar, le contrôleur de la circulation aérienne voit donc et connaît avec précision la position géographique de chacun des appareils qu'il surveille. Les distances sont figurées par des cercles centrés sur le lieu d'implantation du radar. Le contrôleur connaît également le cap suivi par ces avions, matérialisé par un trait lumineux dont la longueur est proportionnelle à la vitesse de déplacement. La trajectoire de chaque appareil est calculée en fonction de ses positions successives.

Ces renseignements sont fort utiles pour identifier chacun des avions avec certitude (donc savoir qui est qui) et lui adresser par radio des consignes - par exemple - d'évitement d'un autre appareil. C'est le **radar** dit **secondaire** qui rend cette identification possible, quand le radar primaire se contente de fournir un écho de la position des avions qu'il détecte.

II.2 Généralités sur le principe du radar secondaire

II.2.1 Définition

Le radar secondaire est un équipement au sol destiné à localiser une cible dans l'espace est aussi d'identifier cette cible. Ceci exige une participation active de la cible, ce qui n'est pas le cas dans le cas du radar primaire.

II.2.2 Principe général du radar secondaire

A l'émission, le radar secondaire interroge la cible grâce à une question codée que l'on appelle **mode d'interrogation**, le radar a la possibilité d'élaborer plusieurs questions. Ces questions sont répétées périodiquement au rythme de la fréquence de répétition du radar.

A bord de l'avion, les informations de mode sont reçues et après décodage de ces informations, l'émetteur de bord (le transpondeur) renvoie une réponse codée que l'on appelle **code** qui contient soit l'information altitude, soit l'identification de la cible, en fonction du mode reçu. Ce code est reçu par le radar et on détecte les impulsions de ce code. L'une d'elles permet de mesurer la distance radar - cible. Le train d'impulsions détecté est ensuite décodé afin de pouvoir l'exploiter.

Les fréquences de la portance utilisées par les radars secondaires sont :

- À l'interrogation : 1030 MHz.
- À la réponse : 1090 MHz.

Les radars secondaires ont une portée supérieure à celle des radars primaires du fait de la participation de la cible.

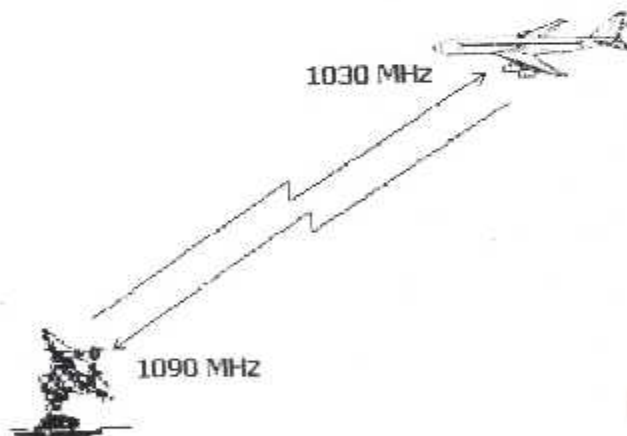


Fig II.1 Schéma de principe du radar secondaire

II.3 Fonctionnement du radar secondaire

Le radar secondaire est un système destiné à faciliter l'identification d'un mobile, il se compose :

- D'une installation sol.
- D'une installation bord.

II.3.1 fonctionnement de l'installation sol

L'installation sol du radar secondaire comprend :

- Une antenne.
- Un duplexeur.
- Un émetteur.
- Un modulateur.
- Un récepteur.
- Un mélangeur.

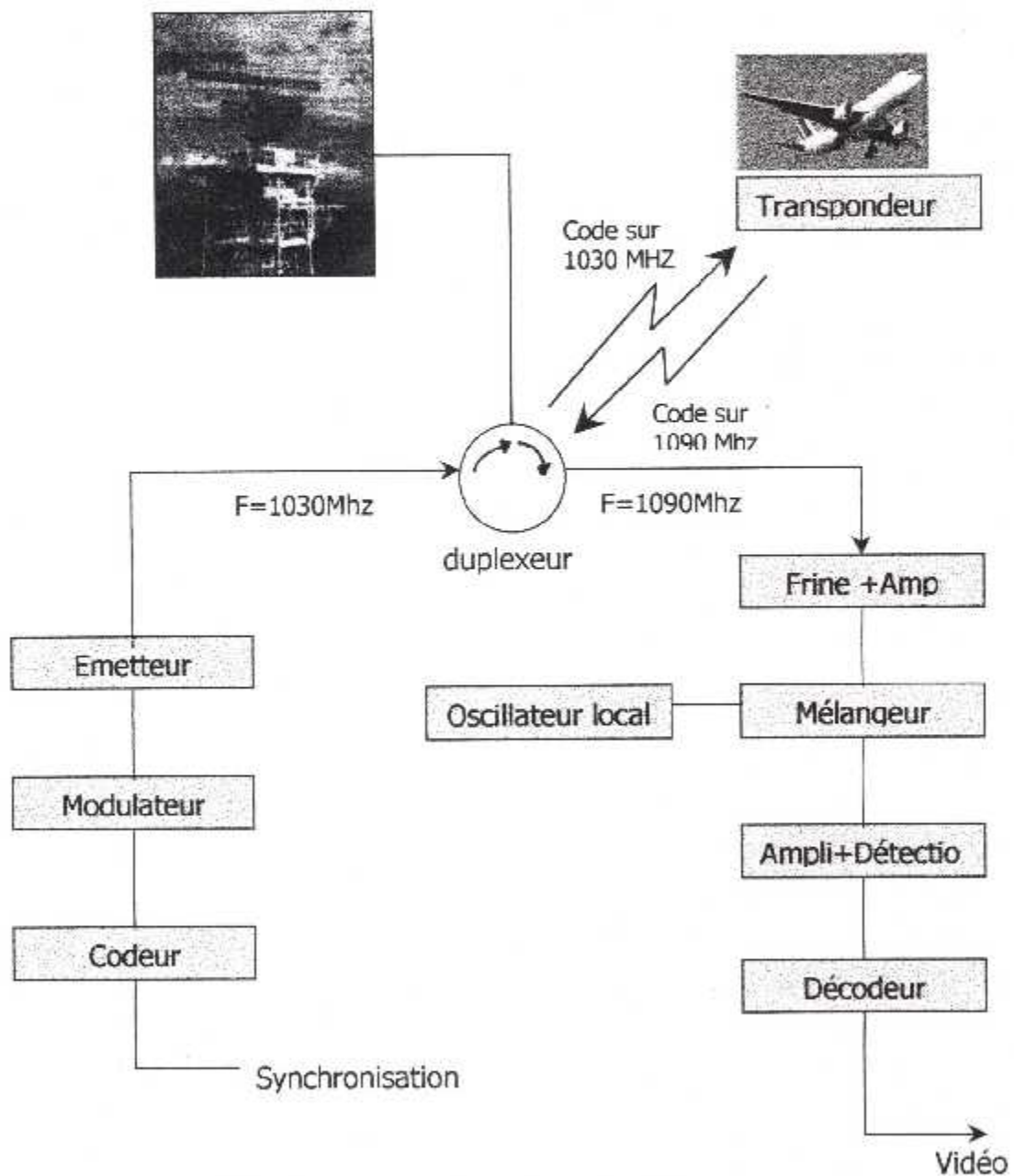


Fig II.2 Schéma synoptique du radar secondaire

- a) **L'ANTENNE :**
C'est l'élément le plus visible du radar, et également le plus connu. Son rôle est de concentrer l'énergie émise par le radar dans un angle déterminé.
- b) **LE DUPLEXEUR :**
C'est un aiguilleur électrique, qui permet, d'une part, au signal émis d'être dirigé vers l'antenne avec une perte minimale tout en isolant convenablement le récepteur, et d'autre part, au signal reçu d'être dirigé en totalité vers le récepteur sans dérivation vers l'émetteur et toujours avec une perte minimale.
- c) **L'ÉMETTEUR :**
Sa partie active, est le tube d'émission dans lequel est engendrée l'impulsion hyperfréquence à la fréquence et la puissance désirées.
- d) **LE MODULATEUR :**
Il constitue la partie active de l'émetteur, il permet de stocker l'énergie pendant les périodes séparant deux émissions successives et de la restituer pendant un temps très bref de l'émission radar.
- e) **LE CODEUR :**
Son rôle essentiel est d'élaborer au rythme de la fréquence de la répétition les impulsions de mode constituant les interrogations de la cible. Ces impulsions ainsi élaborées serviront à moduler un émetteur fonctionnant à la fréquence de 1030 MHz.
- f) **LE RÉCEPTEUR :**
C'est l'élément le plus délicat, et souvent le plus complexe du radar, sa sensibilité doit être grande (jusqu'à $10^{-15}w$), il amplifie les signaux sans déformation et effectue le filtrage et tout les autres traitements adaptés à l'information à obtenir (par exemple : la vitesse, la position...).
- g) **LE MÉLANGEUR :**
Il mélange la fréquence de 1090 MHz provenant du duplexeur avec la fréquence locale f_0 donnée par l'oscillateur local à sa sortie, on obtient la fréquence intermédiaire.

II.3.2 Fonctionnement de l'installation bord (le transpondeur)

Le transpondeur est composé des parties principales suivantes :

- Antenne omnidirectionnelle ;
- Récepteur calé sur 1030 MHz ;
- Décodeur ;
- Codeur ;
- Emetteur calé sur 1090 MHz ;

De plus, on trouve les circuits annexes suivants :

- Filtre passe bande.
- Unité de contrôle.
 - Commande des modes.
 - Commande des codes.
- Le codeur altimétrique.
- Le SLS.

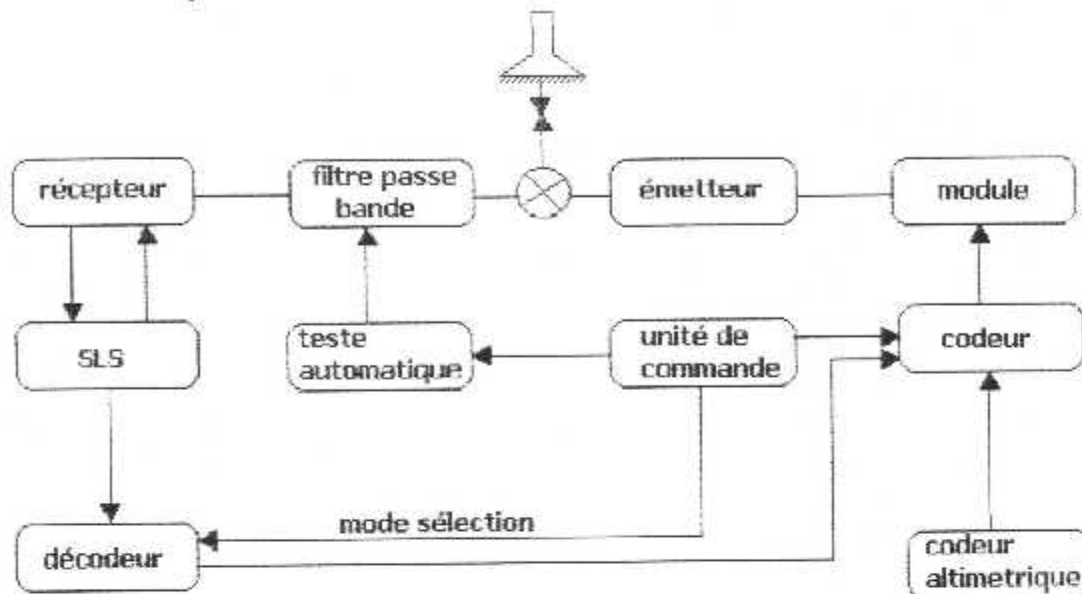


Fig II.3 Schéma synoptique de l'équipement émetteur-récepteur

a) **L'antenne omnidirectionnelle :**

Elle se situe sous fuselage, sert à l'émission et la réception (présence du duplexeur).

b) **Récepteur :**

Le signal d'interrogation de fréquence 1030 MHz est transmis à l'entrée du récepteur par l'intermédiaire d'un duplexeur en passant par un filtre à large bande. L'oscillateur local fournit la fréquence utilisée dans le mélangeur pour obtenir la fréquence intermédiaire, ensuite le train d'impulsions (mode) est acheminé vers les circuits du décodeur.

c) **Décodeur :**

Après que le signal passe par le dispositif ISLS (voir II.4.3 Réponses par lobes secondaires), celui-ci arrive au circuit du décodeur dont le rôle est de connaître le mode d'interrogation. Et ceci en utilisant les temps qui séparent les impulsions d'une même paire ($P1 - P3$), et générer un signal "autorisation de codage" envoyé par l'unité de contrôle pour faire associer à chaque interrogation (mode) la réponse (code) qui lui correspond.

d) **Codeur :**

Son rôle consiste à fournir un train de réponses codées et ceci à partir des informations suivantes :

- Autorisation de codage issue du décodeur.
- Autorisation de codage provenant de la boîte de commande.
- Combinaison de codage des réponses affichées par le pilote ou l'information issue du codeur altimétrique.

e) **Emetteur :**

Pour chaque interrogation (mode) reçue, le codeur forme un code d'impulsions dont l'ordre de distance est fonction du code UHF de fréquence 1090 MHz à la cadence des impulsions. Ce signal sera rayonné par l'antenne par l'intermédiaire du duplexeur.

II.4 Réponses parasites

En radar secondaire, les réponses parasites peuvent être classées en trois grandes familles :

- Les échos de sol.
- Les réponses asynchrones.
- Les réponses sur les lobes secondaires.

II.4.1 Les échos de sol

Théoriquement, il n'y a pas en radar secondaire de réponses dues à la présence du sol, étant donné qu'il y a changement de fréquence au niveau des cibles actives que sont les transpondeurs de bord. Par contre, la réponse codée du transpondeur peut elle-même subir une réflexion des obstacles naturels, et donner, au niveau du récepteur, une réponse dédoublée.

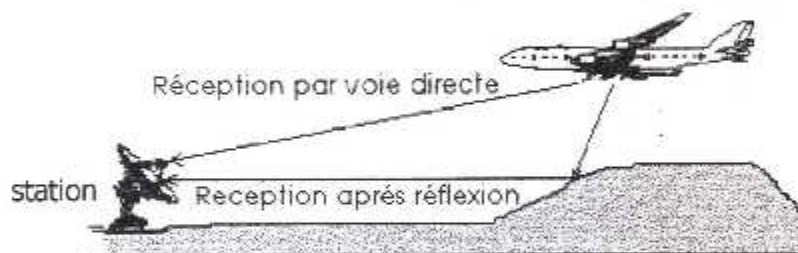


Fig II.4 Les échos de sol

Et, en ce sens, l'influence du sol peut être néfaste, cependant, les niveaux de ces deux réponses sont très différents car celle qui a subi une réflexion est fortement atténuée, aussi sera-t-il facile de faire une discrimination de niveau entre ces deux réponses élémentaires, et l'application sur l'amplificateur Fi d'une loi de désensibilisation (G.V.T) sera généralement suffisante pour neutraliser l'effet de ces réponses indésirables.

II.4.2 Réponses asynchrones

Nous nous sommes intéressés jusqu'ici à la liaison entre un radar et un transpondeur, or tout les radars et tout les transpondeurs fonctionnent sur le même couple de fréquences ; Il se peut donc que la réponse du transpondeur **T** à l'interrogation du radar **A** parviennent également à un autre interrogateur **B** (voir figure II.5).

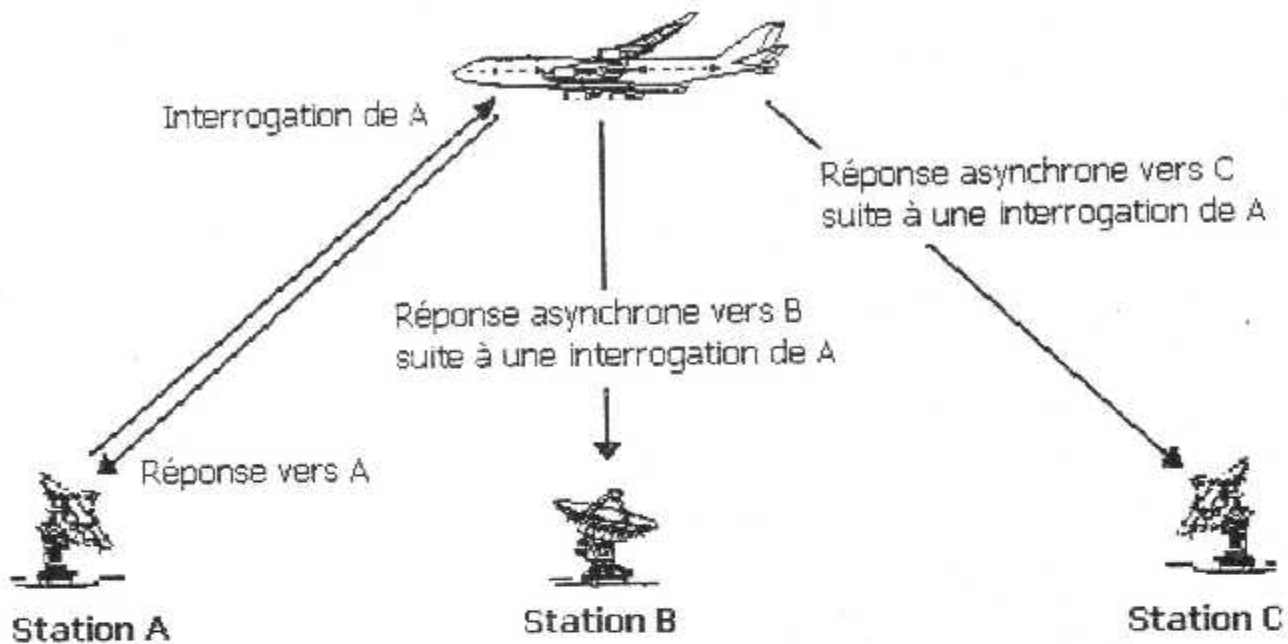


Fig II.5 Réponses asynchrones

Ce type de réponses vers l'interrogateur B peut s'établir sur le lobe principal mais aussi sur le lobe secondaire car il n'y a pas de protection à la réception. Pour remédier à cela, on peut :

- ◆ Régler la période de répétition à des valeurs légèrement différentes pour des radars proches.
- ◆ Faire un traitement de suppression de réponses sur lobes secondaires à la réception (RSLS : Reception Side Lobe Suppression).

II.4.3 Réponses par lobes secondaires

Les lobes secondaires générés par l'antenne perturbent fortement l'exploitation du radar secondaire. Pour un seul avion on arrive à avoir 3 détections.

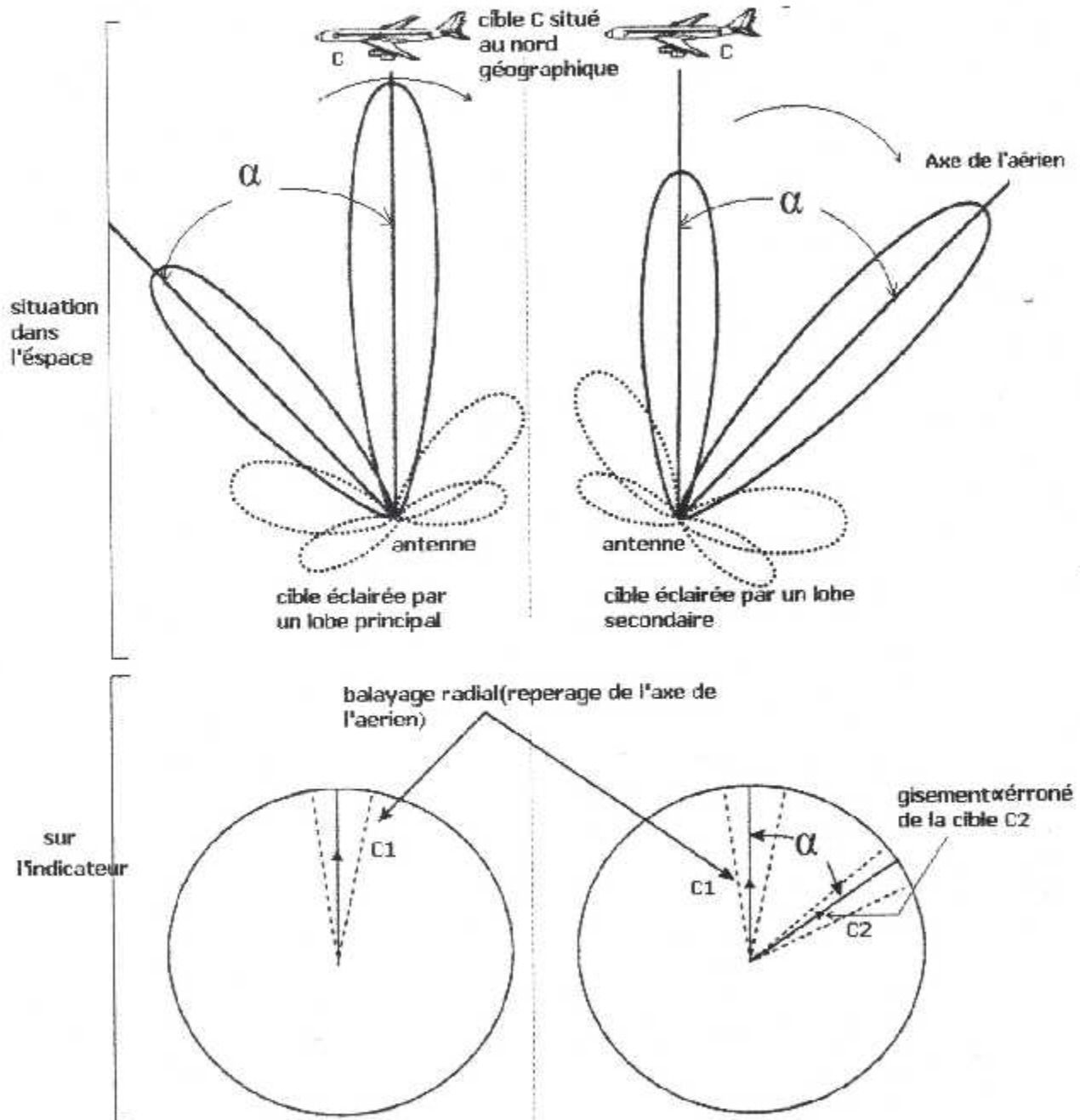


Fig II.6.a Interrogation sur lobe principal

Fig II.6.b Interrogation sur lobe secondaire

Pour remédier à cet inconvénient, il est possible d'utiliser un interrogateur avec suppression des lobes secondaires, c'est le système ISLS (ISLS : Interrogation Side Lobe Suppression) ,ou bien un récepteur avec suppression de lobes secondaires c'est le système RSLs.

Dans la première méthode (le système ISLS) le signal d'interrogation est rayonné au moyen d'une antenne directive. Pour réaliser l'ISLS, ce signal est complété d'une impulsion P2 identique à P1 ou P3, placée 2 μ s après P1, est rayonnée sur une antenne omnidirectionnelle (figure II.7).

Le gain de l'antenne directive est largement supérieur à l'isotrope dans la direction d'un lobe principal, et inférieur dans toutes les autres directions. Dans ces conditions, un répondeur situé dans la direction A de la figure II.7 recevra P1 et P3 à un niveau supérieur à celui de P2. L'inverse se produit dans la direction B.

Dans le répondeur un circuit compare le niveau relatif de P1 et P2. Si P1 est supérieure à P2+X dB, une réponse est émise. Si P1 est inférieure à P2, la réponse est interdite. Entre ces deux seuils existe une zone grise de X dB dans laquelle le comportement du répondeur n'est pas précisé. Selon les normes, X peut aller de 2 à 9. L'inhibition du répondeur consécutive à la reconnaissance de P2 dure $35 \pm 10 \mu$ s, interdisant le décodage de P3 quel que soit le mode d'interrogation.

Toutes les normes imposent l'usage de l'ISLS car il permet d'éviter des réponses désagréables qui peuvent saturer le système inutilement.

Emission de P1 et P3

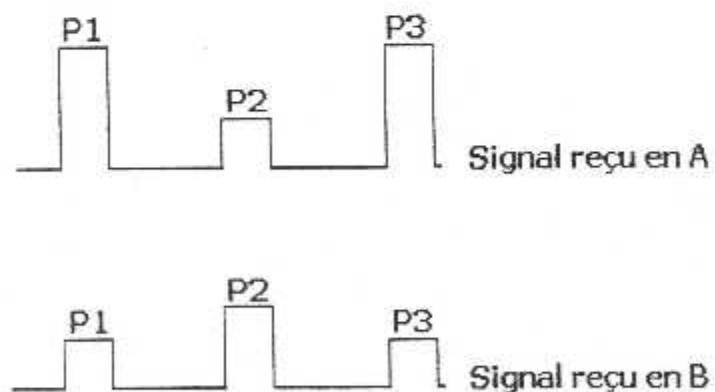
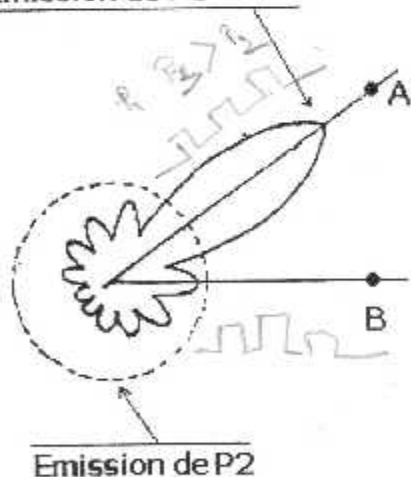


Fig II.7 ISLS : Diagrammes de rayonnement et signaux

La deuxième méthode c'est le système RSLs, sachant que l'ISLS ne peut pas à lui seul résoudre le problème des lobes secondaires. Il n'empêche pas un interrogateur d'être perturbé par des réponses non sollicitées, des multi trajets, un bouilleur et tous signaux susceptibles de l'atteindre par les lobes secondaires de son antenne.

Pour s'en protéger on équipe l'interrogateur de deux voies de réception : deux récepteurs identiques exploitent les diagrammes d'antennes cités pour l'ISLS. Un récepteur est alimenté par l'antenne directive, l'autre par l'antenne omnidirectionnelle. Un circuit compare l'amplitude des signaux sortant des deux récepteurs, et n'en permet la prise en compte que si l'amplitude du signal provenant de la voie directive est supérieure à celle provenant de la voie omnidirectionnelle (figure II.8). Ainsi, sont éliminés tous les signaux ne provenant pas du lobe principal de l'antenne.

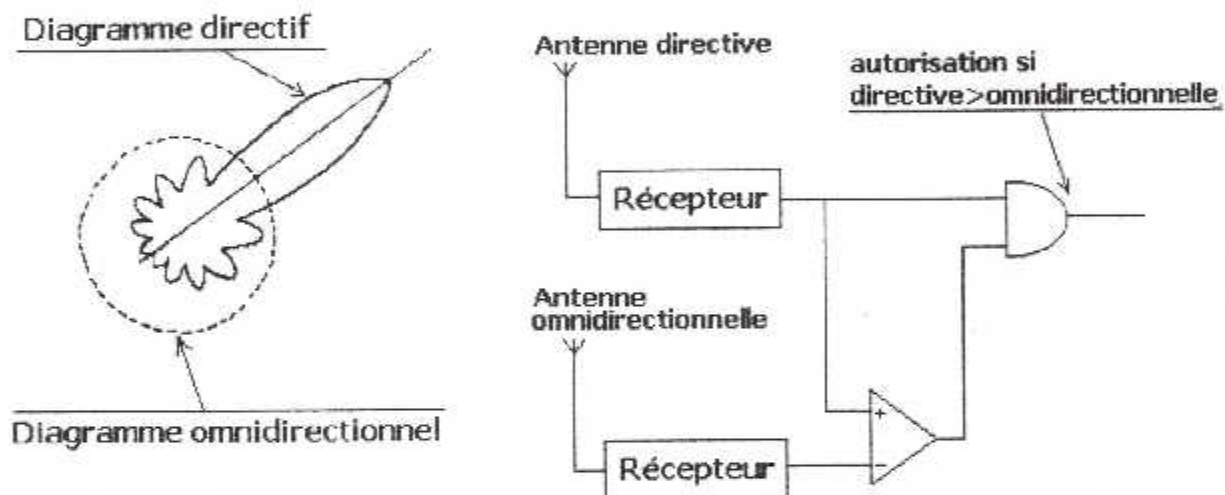


Fig II.8 RSLs : Diagrammes de rayonnement et schéma synoptique

On peut résumer le fonctionnement des systèmes RSLs et ISLS dans la figure II.9.

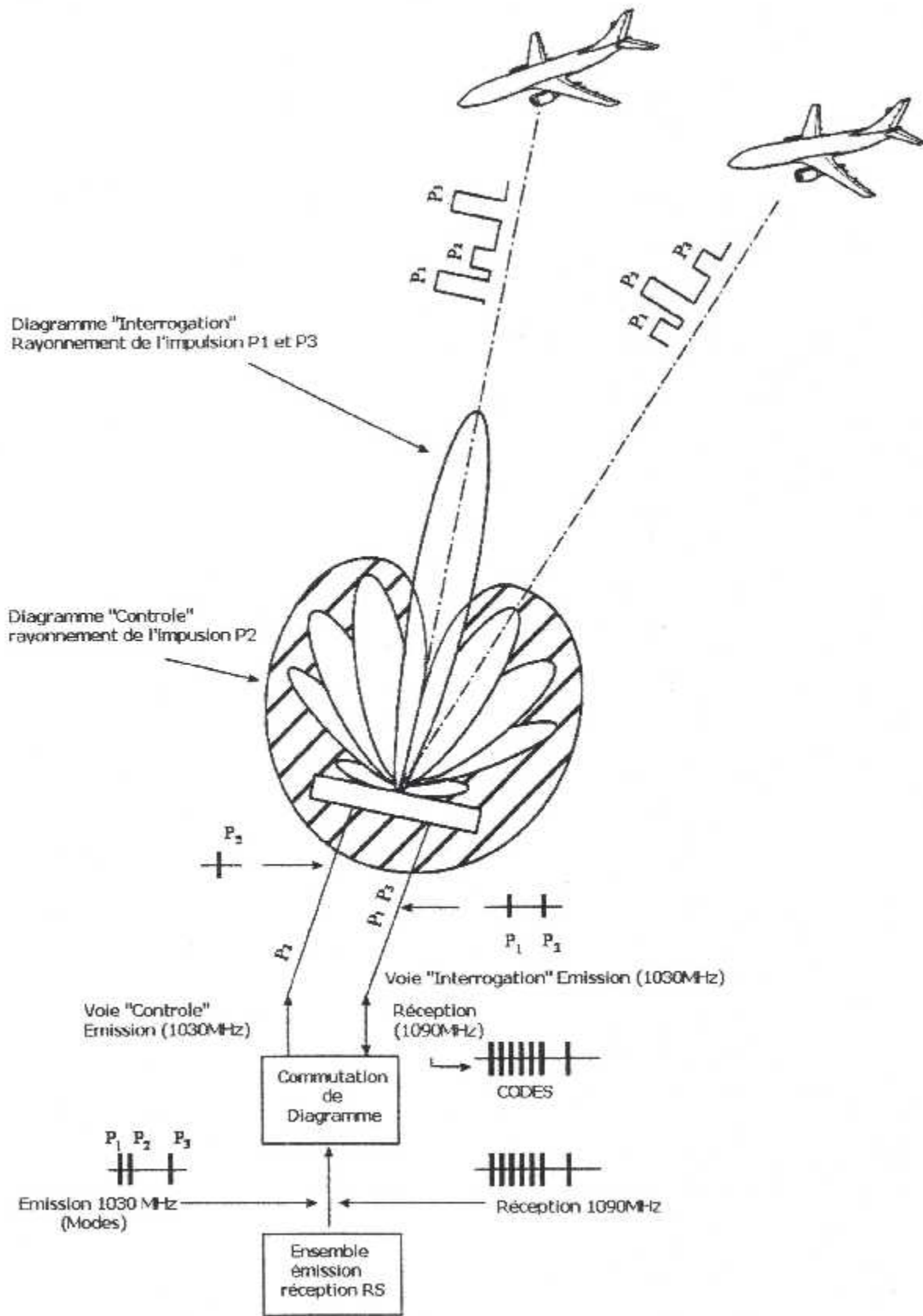


Fig II.9 Réponses sur lobes secondaires

II.5 Le radar mono impulsion

L'idée d'une interrogation sélective des avions pour améliorer les performances des stations radar secondaires classiques remonte à la fin des années 60. De nombreuses études et des expérimentations dans les années 70 ont consolidé les premiers concepts de base : DABS (Discrete Address Beacon System) aux USA, ADSEL (Address Selective) en UK.

Assez rapidement on s'est rendu compte que l'adressage sélectif nécessitait des formats de messages plus longs et des temps d'interrogation par avion plus importants que ceux utilisés pour le radar secondaire classique.

La durée d'éclairement d'un avion par le lobe du radar étant limitée dans le cas des antennes tournantes, une réduction du nombre d'interrogations nécessitait une amélioration de la méthode de détermination de l'azimut de l'avion, basée alors sur le principe de la fenêtre glissante (valeur médiane des azimuts de la dizaine de réponses reçues de l'avion suite aux interrogations périodiques du radar).

Les laboratoires Bell Telephone avaient inventé en 1946 une technique dite "Monopulse" permettant de déterminer l'azimut d'un avion sur une seule récurrence (d'où son nom mono impulsion). Elle consiste à comparer le signal reçu d'un avion sur deux voies différentes (gauche et droite en théorie, somme (SIGMA) et différence (DELTA) dans la réalité). La valeur fournie par la fonction de comparaison permet alors (en l'inversant) de connaître l'angle que fait l'avion avec l'axe de symétrie du lobe principal d'antenne (qui est aussi l'axe de pointage de cette dernière). C'est le principe dit d'"écartométrie".

Dès lors, la position de la cible est connue à l'intérieur du lobe d'antenne sur une seule réponse. Mais il va sans dire que si l'on peut obtenir plusieurs mesures, la position de l'avion n'en sera que plus précise (d'autre part, pour connaître les codes A et C, il faut au moins deux réponses, et pour les valider, il en faut 3 ou 4 par mode).

En fait, la philosophie du Monopulse est de déterminer l'azimut en faisant une moyenne sur des mesures (peut-être imprécises) de l'azimut vrai de l'avion, tandis que la fenêtre glissante faisait une moyenne sur des mesures (également imprécises) de quelque chose qui ne correspondait pas à l'azimut de l'avion mais présentait une série d'erreurs symétriques (un demi-lobe d'antenne) de telle sorte que ces erreurs s'annulaient.

Longtemps limitée aux applications militaires (poursuite de tir, guidage missile,...), cette technique a pu se propager vers les applications radar civil du fait des progrès technologiques et des gains de coûts dans le domaine de la fabrication des antennes.

Compte tenu des gains importants en termes de précision en distance et azimut obtenus par rapport aux radars secondaires classiques, une nouvelle génération de radars secondaires utilisant cette technique mono impulsion fut industrialisée et implantée dans le monde entier depuis le milieu des années 1980. La maîtrise de cette technologie constituait un point incontournable pour la mise en œuvre du Mode S.

II.6 Le radar mode S



Le mode S est un nouveau type de radar secondaire qui est aussi basé sur l'utilisation d'un transpondeur de bord qui répond aux interrogations. Le radar détecte ainsi les aéronefs avec un meilleur budget de liaison et, par-dessus tout, retrouve tous les attributs qui peuvent aider à identifier l'aéronef en même temps.

Le dialogue entre le radar secondaire «traditionnel» et un transpondeur traditionnel utilise deux modes : A et C. Quand le transpondeur est interrogé au mode A, il répond en émettant son code avec le même nom (attribué au vol par le contrôle de la circulation aérienne et entré par le pilote dans le transpondeur au moyen de l'interface). Quand il est interrogé en mode C, le transpondeur répond en donnant son altitude (numérisée et entrée dans le transpondeur automatiquement au moyen d'un altimètre barométrique et d'un altimètre codeur).

Le mode S (sélectif) est un radar secondaire traditionnel amélioré qui fonctionne aux mêmes fréquences (1030/1090 MHz). Sa «sélectivité» est basée sur l'identification non ambiguë des aéronefs au moyen de leurs adresses de 24 bits (une fois que le radar a réussi à acquérir l'aéronef sur sa route, ce qui constitue un autre problème). Ceci sert d'adresses de télécommunications techniques mais ne remplace pas le code de mode A (relié à un vol ou à un plan de vol, alors que le premier est relié à l'aéronef ou, au moins, à son transpondeur). Il y a aussi des plans pour récupérer les codes A et C par l'intermédiaire du mode S.

À part la caractérisation précise de l'aéronef, le mode S protège les données qu'il transmet grâce au calcul des bits de parité (24). Il est ainsi possible de recréer jusqu'à 12 impulsions erronées. Pour la transmission, ces bits de parité sont superposés à ceux de l'adresse du mode S, mais contrairement à ce qu'on pourrait craindre, aucun d'eux ne sont perdus de cette façon.

Enfin, le mode S peut être utilisé pour effectuer des échanges de données plus variées pendant plus longtemps, même si c'est entièrement non planifié. Pour cela, les transmissions en mode S entre la station et le transpondeur utilisent des formats très perfectionnés de 56 et 112 bits qu'on appelle des trames. Elles appartiennent à trois catégories principales : formats de surveillance de 56 bits, formats de communication de 112 bits avec un champ de données de 56 bits qui sont en fait des formats «étendus» de surveillance (liaison montante COMM-A et liaison descendante COMM-B) et des formats de communication de 112 bits avec une liaison de données de 80 bits (liaison montante COMM-C et liaison descendante COMM-D)

Champ de commande 32 Bits	Champ de données 56 Bits	Adresse/parité 24 Bits
------------------------------	-----------------------------	---------------------------

Fig II.10 format 112 bits de communication «courte»

Champ de commande 32 bits	Adresse/parité 24 Bits
------------------------------	---------------------------

Fig II.11 format 56 bits de surveillance

Champ de Commande 8 Bits	Champ de données 80 Bits	Adresse/parité 24 Bits
--------------------------------	--------------------------	---------------------------

Fig II.12 format 112 bits de communication « longue »

On peut relier ensemble les trames pour augmenter le débit : jusqu'à 4 pour COMM-A/B, et avec des «salves» allant jusqu'à 16 pour COMM-C/B. COMM-A/B, qu'ils soient reliés ensemble ou non, portent le nom SLM (message de longueur normale) alors que COMM-C/D, reliés ensemble ou non, portent le nom ELM (message de longueur prolongée).

En utilisant des ondes d'interrogation et de réponse à haute capacité, on peut adresser des messages, échanger des données et détecter et corriger les erreurs (activités qui ne sont pas possibles actuellement avec le SSR) à quatre niveaux différents :

- **Niveau 1** (message de longueur normale) - Europe 1er janvier 00;
- **Niveau 2** (liaison montante pour message de longueur prolongée (ELM)) – plus nécessaire;
(comprend jusqu'à 16 segments consécutifs de mot de 80 bits et demande l'acceptation et le transfert des ELM sol-air)
- **Niveau 3** (liaison descendante pour message de longueur prolongée) – plus nécessaire;
(comprend la transmission des ELM air-sol)
- **Niveau 4** (liaison descendante pour message de longueur prolongée) – plus nécessaire.
(Identique au niveau 4 avec capacité de liaison de données plus élevée.)

II.7 LE CODEUR

II.7.1 la fonction codage

Le codeur du radar secondaire est un dispositif électronique destiné à générer un ensemble de questions codées appelées « **mode d'interrogation** » qui sont émises ensuite par le radar vers les aéronefs survolant un espace aérien donné. Ces aéronefs, s'ils disposent à bord d'un transpondeur qui peut décoder ces questions et répondre en donnant des informations du mode reçu.

Les interrogations du radar, demande d'identification ou d'altitude, sont émises périodiquement au rythme de la fréquence du radar. Lorsque la question posée aux aéronefs est unique, elle est répétée à chaque récurrence. Lorsque plusieurs questions sont posées, elles sont alternées au rythme de récurrence. Ce dernier cas est appelé entrelacement des modes d'interrogation, et la fonction codage donc permet la formation des modes d'interrogation et leur entrelacement.

II.7.2 Formation des modes d'interrogation

Les modes d'interrogation sont constitués d'un train de trois impulsions.

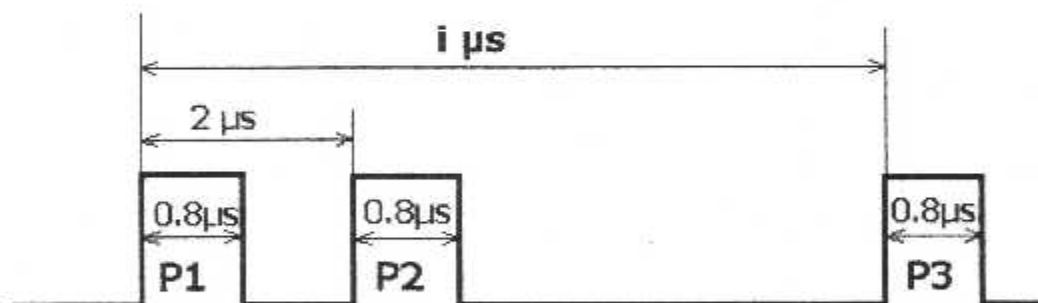


Fig II.13 constitution d'un mode d'interrogation

D'après la figure II.13, on remarque que l'impulsion P2 se trouve toujours à 2 µs de l'impulsion P1. Par contre l'impulsion P3 est séparée de P1 par un intervalle de temps variable " i " qui définit l'interrogation posée aux aéronefs.

Le tableau II.1 donne les différentes valeurs de l'intervalle " i " pour chacune des six modes du radar secondaire.


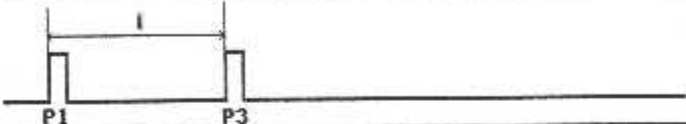
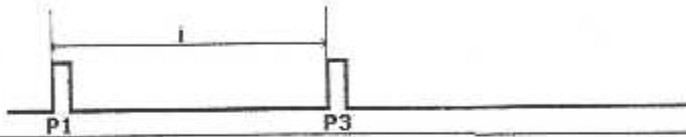
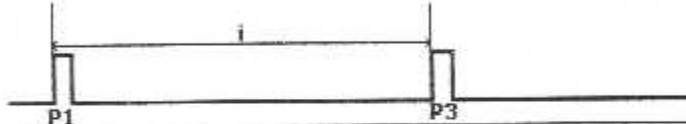
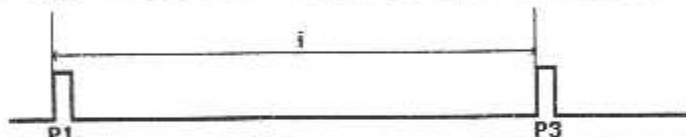
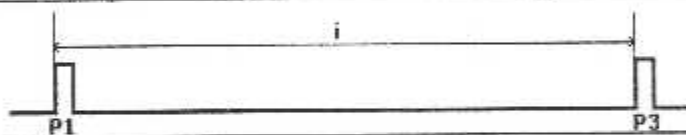
Modes	Representation dans le temps	Valeur de i	Utilisation
1		3 μs	
2		5 μs	
3/A		8 μs	Identification
B		17 μs	Identification
C		21 μs	Info d'alt (niv de vol)
D		25 μs	Non affecté

Tableau II.1 L'intervalle de P1-P3 pour les différents modes d'interrogation

II.7.3 Entrelacement des modes

Pour pouvoir poser plusieurs questions successives et différentes, le radar secondaire doit alterner les modes en réalisant un entrelacement.

On appelle un entrelacement de mode un groupe de mode par exemple **ABC** constitué des modes A , B et C.

Il existe deux types d'entrelacement :

- Entrelacement au rythme de récurrence.
- Entrelacement au rythme de tour d'antenne.

II.7.3.1 Entrelacement au rythme de récurrence

Il s'agit de transmettre un entrelacement donné sans tenir compte des tours d'antenne. On transmet successivement chaque mode du groupe à chaque récurrence, lorsque le groupe est fini, on reprend le même cycle du début. Ceci est illustré par la figure II.14 en utilisant le groupe ABC comme exemple :

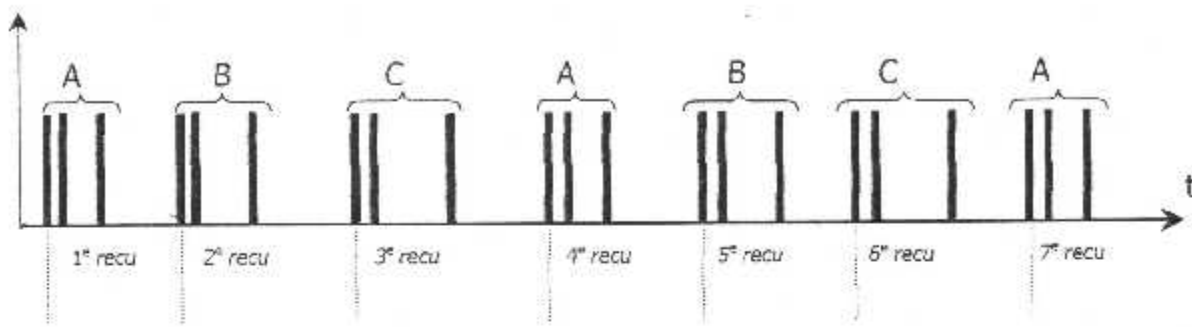


Fig II.14 Entrelacement au rythme des récurrences

II.7.3.2 Entrelacement au rythme de tour d'antenne

Il s'agit ici de transmettre un groupe de mode pendant le tour d'antenne, et de changer ensuite le groupe de modes pendant le tour d'antenne suivant. Ceci est illustré par la figure II.14 si on choisit les groupes ABC et BCC.

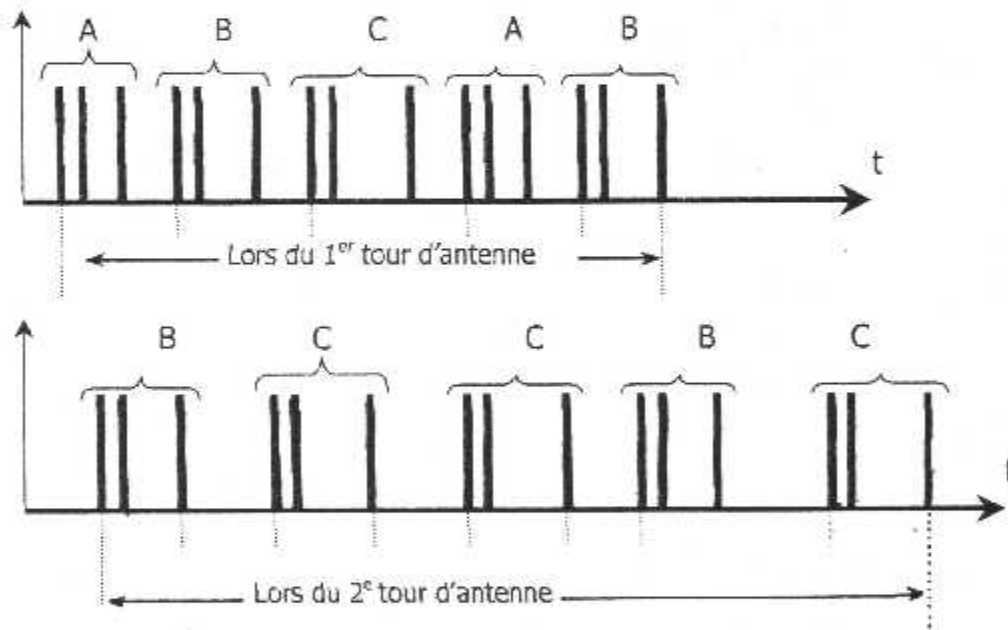


Fig II.15 Entrelacement au rythme des tours d'antenne

II.7.4 Réalisation du codeur

Le codeur radar secondaire est à réaliser sur une carte autour d'un microprocesseur 8085 de INTEL pour des raisons suivantes :

- Le microprocesseur est disponible au niveau de l'institut.
- Il possède une horloge interne.
- Possède un port entrées/sorties série.
- Possède un bus de données de 8 bits.
- Rapidité du traitement des données.

CHAPITRE III
PRESENTATION DU
MICROPROCESSEUR 8085

III.1 Définition

Un système à microprocesseur est un système électronique complexe qui est conçu à partir d'un microprocesseur destiné à traiter de l'information. La conception et la réalisation d'un tel système sont basées sur :

- l'étude du matériel qui comprend :

Quelques circuits intégrés LSI (large scale intégration : intégration à grande échelle soit 500 à 10000 transistors dans un seul circuit LSI) qui sont :

- Un microprocesseur 8085.
 - Une mémoire RAM qui contient les données.
 - Une mémoire EPROM qui contient le programme.
 - Interfaces avec des circuits périphériques.
- L'étude du logiciel, ce dernier est un ensemble de programmes nécessaires pour le fonctionnement du système à microprocesseur.

III.2 Architecture d'un système à microprocesseur

La figure III.1 illustre l'architecture du 8085. Il est muni d'un compteur d'instruction à 16 bits, d'un verrou d'adresses qui charge le bus d'adresses spécialisé ($A_{15}-A_8$) et d'un bus multiplexé (AD_7-AD_0).

Les données parallèles entrent et quittent le microprocesseur par le bus multiplexé (AD_7-AD_0). Ce bus transmet une adresse quand la ligne de commande ALE est à high, et des données quand elle est à low.

Le bus interne de données à 8 bits transporte les données entrantes ou sortantes à l'intérieur de l'unité. Les données peuvent aller du bus de données interne à l'accumulateur à 8 bits au registre temporaire, aux indicateurs, au registre d'instructions, à l'unité de commande, à l'un quelconque des registres généraux (B, C, D, E, H et L), au pointeur de pile à 16 bits, au compteur d'instruction à 16 bits ou au tampon à 8 bits d'adresses/données.

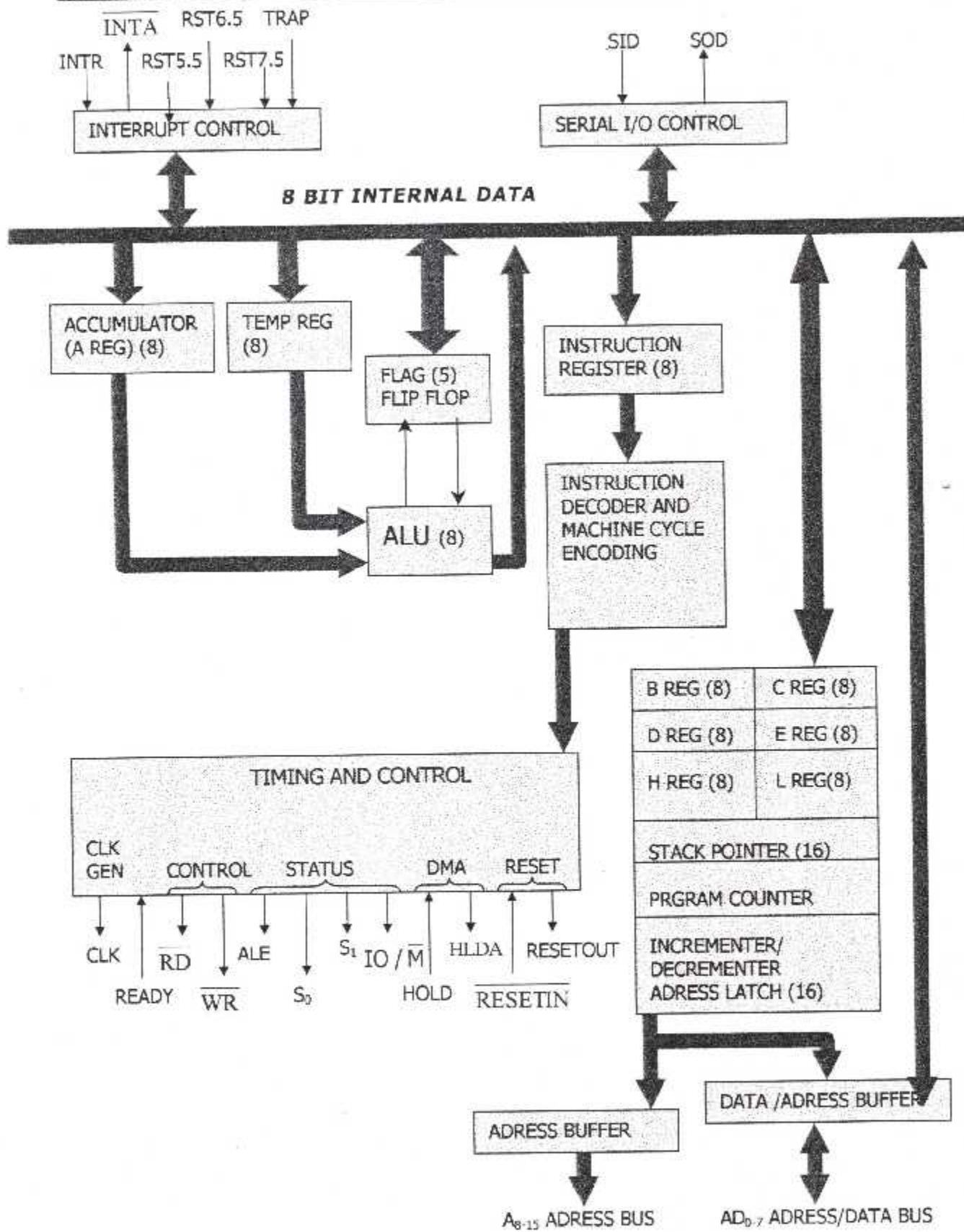


Fig III.1 Architecture d'un système à microprocesseur

III.2.1 L'unité arithmétique et logique

L'UAL est étroitement associée à l'accumulateur, au registre d'état et à quelques registres temporaires non accessibles au programmeur. Les opérations arithmétiques, logiques et de permutation sont exécutées dans L'UAL et les résultats sont déposés dans l'accumulateur.

III.2.2 L'unité logique de commande

Cette unité comprend :

- Un registre d'instructions de 8 bits qui reçoit le 1^{er} octet des instructions à exécuter, cet octet contient le code opération qui indique l'action à exécuter.
- Un décodeur d'instructions qui utilise le code opération pour produire les signaux de commande requis pour sélectionner et activer les circuits concernés par l'exécution de l'instruction.

III.2.3 Les registres

Le 8085 est muni de registres à 8 bits et à 16 bits. Il comporte 8 registres adressables à 8 bits, 6 d'entre eux peuvent être utilisés comme registres à 8 bits ou en paire à 16 bits. De plus le 8085 comporte 2 registres à 16 bits. Les registres sont les suivants :

L'accumulateur (ou registre A) qui est le nœud de toutes les opérations de l'UCM qu'elles soient arithmétiques, logiques, de chargement ou de rangement d'entrées/sorties. C'est un registre à 8 bits.

Les registres généraux (BC, DE et HL) qui peuvent être utilisés comme 6 registres à 8 bits ou trois paires de registres à 16 bits selon l'instruction en cours d'exécution. La paire HL de registre (dite pointeur de données) peut être utilisée pour pointer sur des adresses. Quelques instructions utilisent les paires BC et DE comme pointeur d'adresses mais normalement elles sont utilisées comme registres généraux de données

Le compteur d'instruction qui pointe toujours sur la case mémoire de l'instruction suivante à exécuter, il contient une adresse à 16 bits.

Le registre d'état qui comporte des indicateurs à 1 bit contenant l'information relative à l'état de l'UC. Ces indicateurs sont utilisés par les branchements conditionnels, les appels et retours de sous-programmes, c'est un registre à 8 bits.

III.2.4 Les bus de données et d'adresses

Les lignes d'E/S du 8085 se divisent en 3 ensembles groupés en bus :

- ◆ Le bus d'adresses
C'est un bus unidirectionnel de 16 lignes sortant du microprocesseur, permettant ainsi de pointer une case mémoire ou de définir l'organe d'E/S pour une lecture ou écriture.
- ◆ Le bus de données
C'est un bus bidirectionnel à 8 lignes contenant la donnée qui va être échangée entre le microprocesseur et la mémoire (ou le périphérique d'E/S)
- ◆ Le bus de contrôle
Il permet d'agencer et de synchroniser l'échange de données entre le microprocesseur et ses périphériques, il englobe les signaux suivants :
 - \overline{RD} lecture : actif au 0 logique (read).
 - \overline{WR} écriture : actif au 0 logique (write).
 - IO/\overline{M} sélection mémoire ou E/S (input-output/memory).
 - $IO/\overline{M}=0$: sélection mémoire.
 - $IO/\overline{M}=1$: sélection E/S.
 - ALE ligne sortante, un front descendant ou niveau haut sur cette broche permet de mémoriser les adresses qui sont stabilisées et permet donc le démultiplexage des 8 lignes de poids faible du bus d'adresses.

III.3 Les cycles du microprocesseur 8085

On distingue trois types de cycles superposés, chacun correspond à des événements précis dans le microprocesseur. Par ordre de durée croissante, on distingue :

- Cycle d'horloge
- Cycles machines
- Cycles d'instructions

III.3.1 Cycle d'horloge

Le fonctionnement du 8085 est rythmé par un signal d'horloge ayant la forme d'une onde carrée périodique. La durée du cycle est de 320 ns minimum, mais cette valeur peut aller jusqu'à 2 μ s.

III.3.2 Cycles machines

Un cycle machine correspond à un intervalle de temps comportant généralement un accès à l'extérieur du microprocesseur, mémoire ou périphériques. En fonction de la nature de cet accès, on distingue les cycles machines suivants :

1. Prise en mémoire d'un octet d'instruction.
2. Lecture en mémoire d'un octet de données (memory read : \overline{MR}).
3. Ecriture en mémoire d'un octet de données (memory write : \overline{MW}).

4. Entrée ou lecture depuis un périphérique (input output read :IOR).
5. Sortie ou écriture dans périphérique (input output write :IOW).
6. Acceptation d'interruption (interrupt acknowledge :INA).
7. Bus inactif (bus Idle :BI).

Un cycle machine comporte de 3 à 6 cycles d'horloge, suivant les cas.

III.3.3 Cycles d'instructions

On appelle cycle d'instruction le temps nécessaire pour prendre en mémoire une instruction et l'exécuter. La durée d'un cycle d'instruction dépend donc de la nature de l'instruction, elle comporte au minimum un cycle machine et peut comporter jusqu'à 5 cycles machine. En terme de cycles d'horloge, l'instruction la plus courte en prend 18.

III.4 Stratégies d'accès aux entrées/sorties

On distingue les stratégies suivantes :

- La méthode d'E/S programmée.
- La méthode d'interruption.
- La méthode d'accès directe en mémoire.

III.4.1 La méthode d'E/S programmée :

Mode programmé avec ou sans test d'état (figIII-2).

Les transferts E/S sont prévus à des moments bien précis dans le programme utilisateur. C'est donc le processeur qui a l'initiative de commander le transfert et ceci par le rencontre d'une instruction de transfert. Ainsi lorsque le seul périphérique d'un système est une télétype, l'imprimante de cette dernière est toujours disponible : il suffit alors, au moment opportun dans le programme, d'envoyer un caractère à l'imprimante. C'est un mode programmé.

Une amélioration possible consiste à aller tester l'état on occupé de l'imprimante avant d'effectuer le transfert, et d'attendre dans le cas où l'imprimante n'est pas libre.

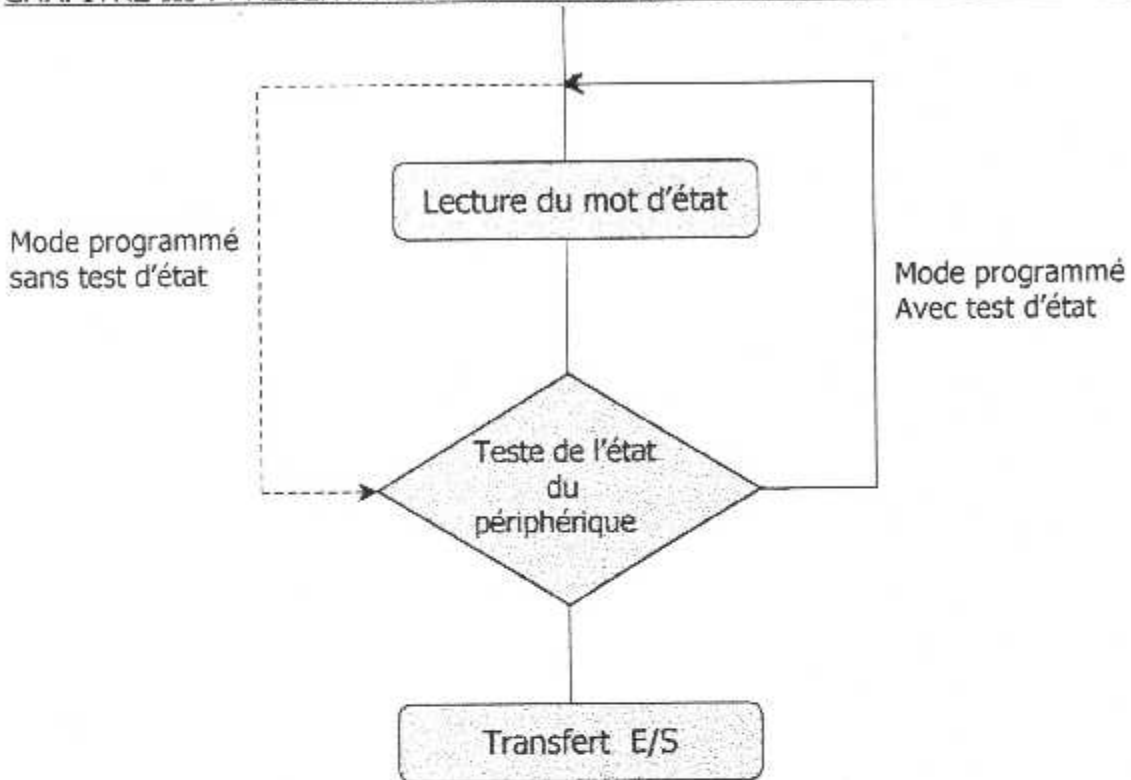


Fig III.2 organigramme d'un transfert E/S en mode programmé

III.4.2 LA METHODE D'INTERRUPTION : (figure III-3)

Les transferts peuvent également s'effectuer à l'initiative du périphérique et non plus du processeur. Ainsi par un clavier, le programmeur ne peut pas savoir à quel moment l'utilisateur viendra appuyer sur une touche du clavier par donner ou demander une information.

Dans ce cas l'appui sur la touche fera une demande d'interruption au processeur. Celui ci prendra acte de cette demande, répondra par l'envoi d'un signal qu'il a bien reçu cette demande, terminera d'exécuter l'instruction en cours de traitement et se mettra à la disposition du clavier.



Fig III.3 transfert d'une demande d'interruption du périphérique au processeur

III.4.3 La méthode d'accès directe mémoire (DMA) :

Lorsqu'un périphérique a un assez grand nombre d'informations à rentrer en mémoire ou lorsqu'il doit en recevoir une quantité importante de la mémoire le transfert par interruption est long.

Un mode de transfert E/S particulier est alors possible : c'est l'accès direct mémoire. Le périphérique peut alors échanger directement avec la mémoire, les bus adresses, données, et commandes étant déconnectés du processeur par leur mise à l'état haute impédance, nous devinons que cette nouvelle possibilité va entraîner deux autres signaux de commande au niveau du microprocesseur :

- L'un qui sera une demande d'accès direct mémoire DMA REQUEST ou DMA REQ ;
- L'autre qui sera une reconnaissance, c'est à dire une autorisation d'accès direct d'un périphérique à la mémoire et qui par le fait même, sera un message signifiant « les bus adresses, données et commandes sont déconnectés du microprocesseur, vous pouvez les utiliser ».

Ce signal DMA ACKNOWLEDGE pourra être désigné par DMACK ou DACK par exemple.

III.5 Les interruptions du microprocesseur 8085

III.5.1 Niveaux d'interruption

Le 8085 possède 5 niveaux hiérarchisés d'interruption qui sont dans l'ordre de priorité décroissante : TRAP, RST 7.5, RST 6.5, RST 5.5, et INTR.

III.5.1.1 Interruption non masquable (TRAP)

Tout front montant ou tout niveau 1 sur cette entrée déclenche le processus de prise en compte d'une interruption par le microprocesseur après qu'il a terminé l'exécution de l'instruction en cours. C'est la demande d'interruption sans condition du 8085.

III.5.1.2 Interruption masquable (RST 7.5, RST 6.5, RST 5.5)

Elles sont prises en compte si deux conditions sont réalisées :

- Le bit d'état IE est positionné à 1 : il est mis à 1 par l'instruction EI (autorisation des interruptions), et à 0 par l'instruction DI (inhibition des interruptions).
- La demande d'interruption n'est pas masquée ce qui se produit dans deux cas :
 - Le bit MSE de l'interruption SIM est à 0.
 - Le bit MSE est à 1 mais le masque spécifique de l'interruption est à 0. ce masque spécifique, mis à 0 ou à 1 par l'instruction SIM est :
 - M7.5 pour l'instruction RST 7.5.
 - M6.5 pour l'instruction RST 6.5.
 - M5.5 pour l'instruction RST 5.5.

L'interruption RST 7.5 n'étant activée que sur un front montant, il est nécessaire que la bascule RST 7.5, qui mémorise la transition, soit d'abord initialisée à 0, cela se fait en mettant à 1 le bit RST 7.5 de l'instruction SIM.

III.5.1.3 Interruption INTR :

Une demande d'interruption sur cette broche entraîne le dépôt sur le bus de données, par l'organe d'E/S qui a provoqué l'interruption, du code RST N affecté à cet organe. Les cinq interruptions du 8085 sont des instructions RST ART mais les quatre premières étudiées ne nécessitent pas, contrairement à INTR, le dépôt d'un code sur le bus de données. Le microprocesseur se branche alors automatiquement à une adresse spécifique de l'interruption.

III.5.2 Traitement des interruptions

La demande d'interruption parvenue au microprocesseur sur les 5 entrées d'interruptions, entraîne l'exécution par le microprocesseur de la séquence spéciale suivante :

- Le microprocesseur termine l'instruction en cours.
- Il vérifie, pour les instructions masquables, si l'interruption est autorisée ou pas.
- Il active le signal \overline{INTA} .
- Il sauvegarde dans la pile l'adresse de retour au programme interrompu.
- Il déclenche ensuite un cycle FETCH ; si la demande d'interruption est INTR pour aller chercher sur le bus de données le code de l'instruction RST N et il se branche à l'adresse spécifique de RST N.

L'opérateur aura placé une instruction de saut au sous-programme relatif à RST N. Si un contrôleur d'interruption est utilisé, ce dernier déposera le code CALL. Pour les autres commandes d'interruption, il n'a aucun code à aller chercher et il se branche aussitôt à l'adresse spécifique de la demande d'interruption qui est donnée sur le tableau suivant :

Interruption	Adresse à partir de laquelle Est rangée l'instruction de saut au SP
TRAP	24
RST 5.5	2C
RST 6.5	34
RST 7.5	3C

Tableau III.1-Traitement des interruptions

III.5.3 Les instructions SIM et RIM

Chacune de ces deux nouvelles instructions du 8085 assure deux fonctions indépendantes l'une de l'autre :

- ✓ Une fonction relative aux interruptions RST 5.5, RST 6.5 et RST 7.5.
- Une fonction relative à l'entrée ou la sortie série soit SID ou SOD.

III.5.3.1 L'instruction SIM (*set interrupt mask*)

a) Fonction relative à RST 5.5, RST 6.5 et RST 7.5.
 L'instruction SIM commande la validation ou non de chacune des interruptions RST 5.5, RST 6.5 et RST 7.5. Cette validation se fait par un mot de commande, préalablement mis dans l'accumulateur A. Pour autoriser ou non la prise en compte des demandes d'interruption arrivant sur chacune des entrées RST 5.5, RST 6.5 et RST 7.5, il faut mettre respectivement à 0 ou à 1 le bit de masque relatif à chacune de ces entrées.

Toutefois pour inhiber les interruptions sur une ou plusieurs de ces trois entrées il faut mettre à 1 le bit de validation des masques (MSE).

RST 7.5 est active sur un front et la bascule correspondante peut être initialisé à 0 en mettant à 1 le bit R7.5 ; aussi pour activer le RST 7.5 il faut initialiser à zéro la bascule RST 7.5 et inhiber RST 6.5 il faut :

R7.5=1 MSE =1 M7.5=0 M6.5=1 M5.5=1

Soit le mot de commande 00011011 ou 1B en hexadécimal.

D'où la programmation MVI A, 1B
 SIM

b) fonction relative à SOD :
 L'instruction SIM sert également à positionner à 0 ou à 1 le verrou (latch) de sortie qui constitue la sortie série SOD (serial output data). Ce positionnement à 0 ou 1 se fait en validant la sortie série par la mise à 1 du bit SOE et en mettant respectivement 0 ou 1 dans le bit SOD c'est à dire le bit 2⁷ du mot de commande. Pour mettre SOD à 0 ou à 1 sans craindre de modifier les masques d'interruption, il suffit de mettre les bits 3 et 4 du mot de commande à 0, pour ce faire, nous mettrons à 0 tous les bits autres que SOD et SOE. Ainsi si nous désirons mettre SOD à 1 le mot de commande sera 11000000 soit C0 d'où la programmation :

MVI A, C0
 SIM

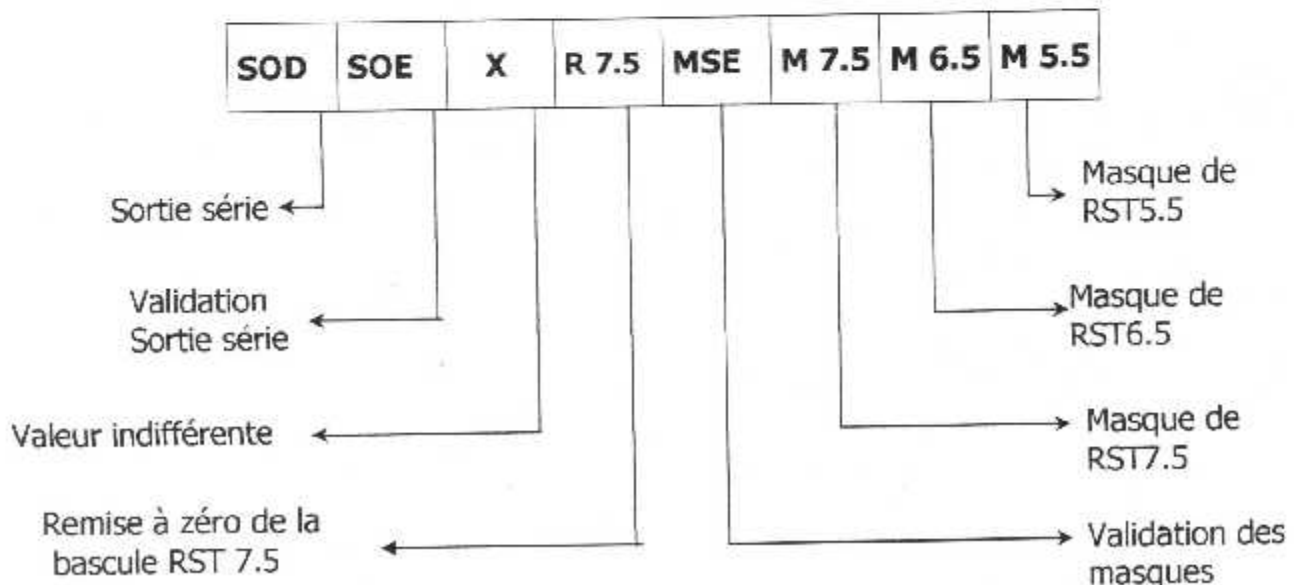


Fig III.4 Synoptique de détermination du mot de commande de l'instruction SIM

III.5.3.2 L'instruction RIM (*Read Interrupt Mask*)

a) Fonction relative à RST 5.5, RST 6.5 et RST 7.5.

L'instruction RIM exécute la lecture d'un mot d'état à 7 bits.

Ce mot d'état indique :

- L'état, c'est à dire la valeur 0 ou 1 de chacun des masques d'interruption M 7.5, M 6.5 et M 5.5 ;
- La validation ($IE=1$) ou l'inhibition ($IE=0$) des demandes d'interruption sur les entrées RST 5.5, RST 6.5 et RST 7.5 ;
- Les interruptions en attente.

b) Fonction relative à SID (*serial input data*).

Le bit 2^7 du mot d'état transmis dans l'accumulateur par l'instruction RIM est l'état 0 ou 1 de l'entrée série SID.

Ces deux fonctions sont exécutées par une instruction RIM. Il est à noter que cette instruction ne positionne aucun indicateur.

Remarque :

L'entrée RESETIN remet à 0 la demande d'interruption TRAP, la bascule associée à RST 7.5 et le bit IE. Elle met aussi à 1 les masques M 7.5, M 6.5 et M 5.5.

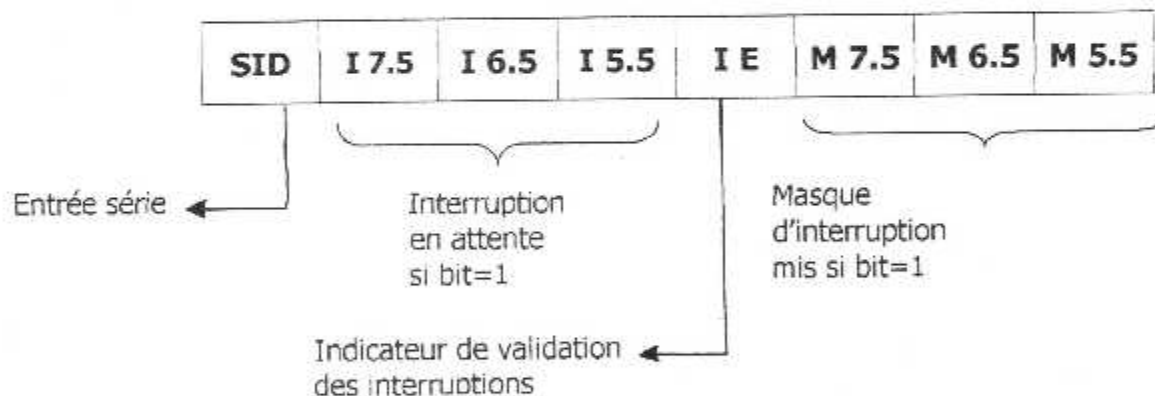


Fig III.5 Mot d'état envoyé dans l'accumulateur par l'instruction RIM

**CHAPITRE IV
CONCEPTION
DE LA CARTE
CODEUR**

VI.1 Circuit d'horloge

Le 8085 comporte un générateur d'horloge interne et ne nécessite que l'adjonction d'un cristal (quartz) afin de régler la fréquence de travail.

Soit :

F_q : fréquence du quartz

F_t : fréquence du travail

On a la relation suivante donnée par le constructeur : $F_t = \frac{F_q}{2}$

Le 8085 admet un cycle d'horloge minimum de 320 ns et un cycle d'horloge maximum de 2 μ s, donc on obtient la relation suivante :

$$500 \text{ KHz} < F_t < 3.125 \text{ MHz}$$

Pour notre carte, nous travaillons avec un quartz de fréquence $F_q = 4 \text{ MHz}$, donc F_t vaut 2 MHz.

La pratique montre que l'adjonction des deux condensateurs de 12 μ F, aux bornes du quartz comme l'indique la figure IV.1, est très conseillée pour des raisons de stabilité du signal d'horloge.

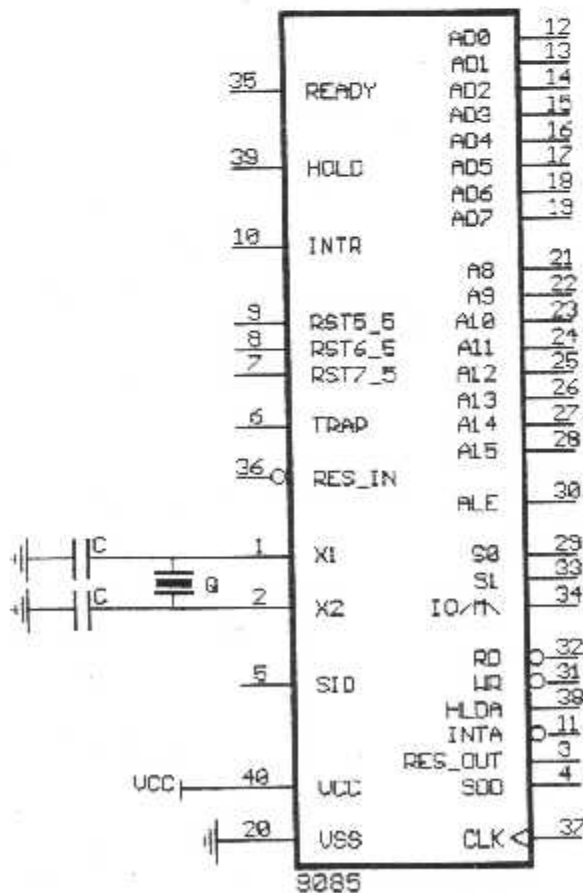


Fig IV.1 Le microprocesseur 8085 et le Circuit d'horloge

VI.2 Circuit de réinitialisation

Le circuit de réinitialisation permet une réinitialisation complète du système en envoyant un signal de niveau logique "0" sur la broche RESET IN du microprocesseur. Le 8085 réémet ce signal inversé sur la ligne RESET OUT permettant ainsi de synchroniser les circuits associés. Après l'application de la tension d'alimentation, il faut attendre un certain temps, voir 3 périodes d'horloge avant que les tensions internes du 8085 ainsi que les informations sur les bus ne se stabilisent, donc le signal RESET IN doit être maintenu au moins pendant trois périodes d'horloge, ce rôle est confié au circuit RC de la figure IV.4.

Quand le RESET IN est désactivé, le 8085 entre dans le cycle «FETCH», et comme le PC a été remis à zéro, le programme démarre nécessairement à l'adresse 0000 H.

Détermination des valeurs de R et de C :

Comme la fréquence du quartz choisie est $F_q=4$ MHz donc la fréquence du travail est $F_t=2$ MHz alors nous déduisons que $T_t=1/F_t=1/2=0.5$ μ s aussi de la condition

$RC > 3T_t \Rightarrow RC > 1.5$ μ s.

Nous pouvons prendre alors **R=330 Ω** et **C=220 μ f**

VI.3 Circuit de démultiplexage du bus d'adresses

Les 8 lignes d'adresses de poids faible sont multiplexées avec les 8 lignes de données, il existe plusieurs méthodes de démultiplexage, la méthode retenue est illustrée dans la figure suivante :

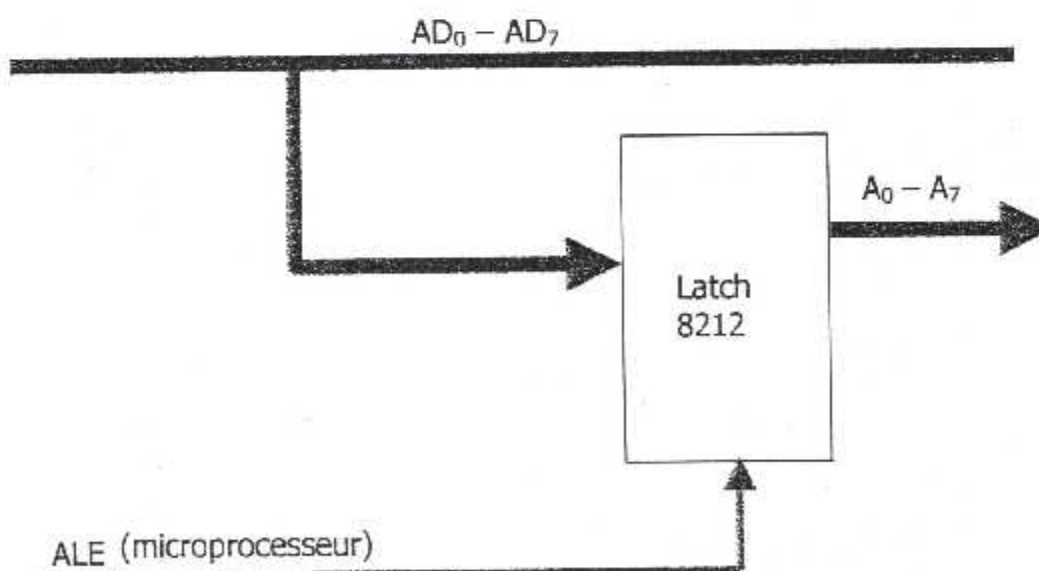
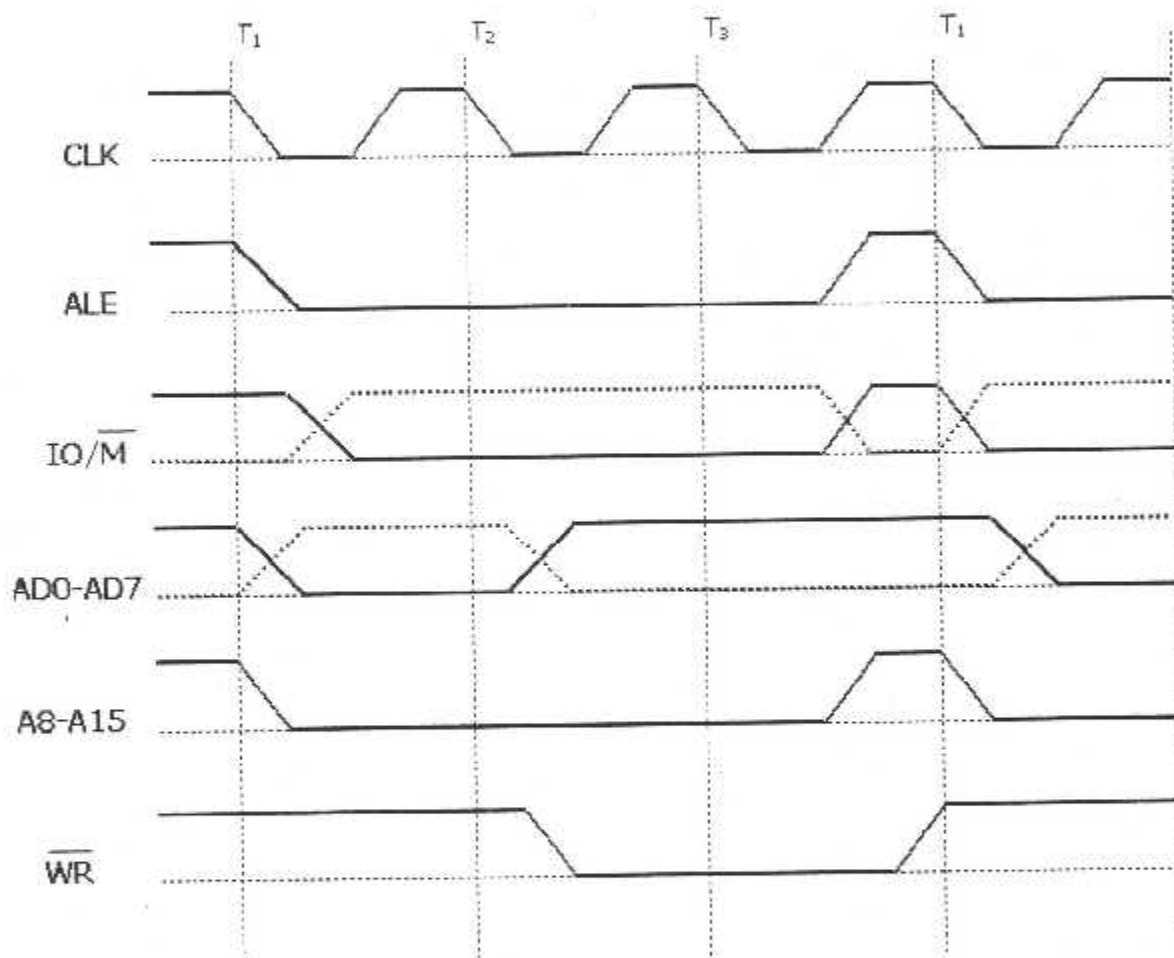


Fig IV.2 Synoptique de démultiplexage du bus d'adresses

Le bus (AD₀- AD₇) transmet une adresse quand la ligne ALE est au niveau haut, et les données quand elle est au niveau bas.
 D'après le chronogramme de la figure IV.3, on voit que les adresses ne se stabilisent en sortie qu'au front descendant de ALE.

Le latch 8212 choisi pour notre réalisation, attend que ALE soit au niveau bas pour mémoriser les informations disponibles à son entrée et ALE attaque l'entrée DS2 (Device select) du latch.



IV.3 Cycles de démultiplexage

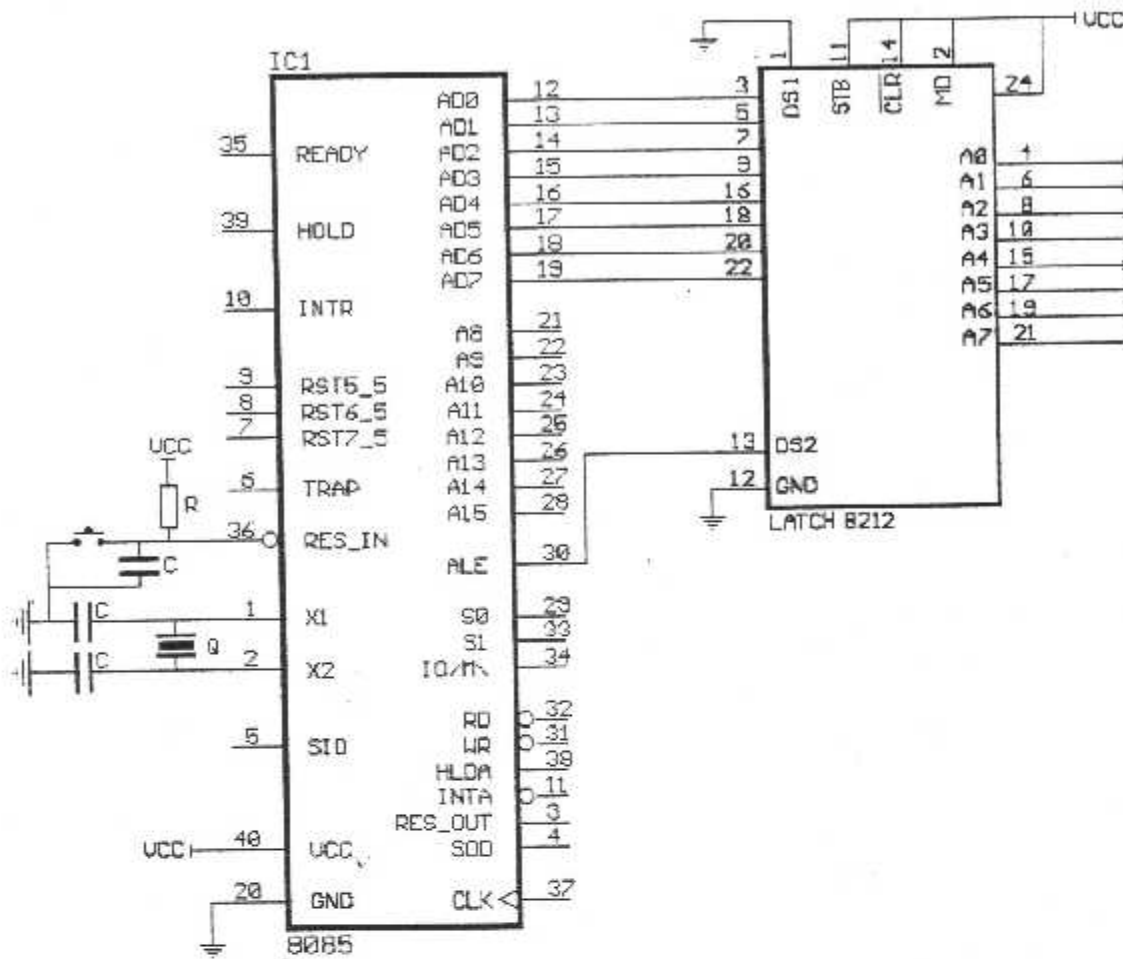


Fig IV.4 Circuit de démultiplexage du bus d'adresse

IV.4 Bufferisation des bus

Notre carte comprend un minimum de circuits LS (Low Shottekey), et notamment des interfaces d'entrées/sorties, une RAM et une EPROM (la mémoire contenant le programme moniteur etc....). Cependent il est nécessaire d'étendre les possibilités de notre carte par l'adjonction de plusieurs organes .Dans ce cas, il est nécessaire d'augmenter la sortance des bus, car ceux-ci ne peuvent pas alimenter un nombre illimité de circuits.

En effet chaque circuit connecté à un bus a un courant de fuite quand il n'est pas validé et un courant d'entrée quand il est validé. De plus il a un condensateur d'entrée, ce qui peut perturber le fonctionnement du système.

IV.4.1 Bufferisation du bus de données

Comme nous l'avons déjà mentionné, le bus de données est un bus bidirectionnel donc nécessitant un buffer bidirectionnel, la direction du transfert des données lues est dictée par la ligne de contrôle \overline{RD} qui est réalisée au pin DIR du buffer (voir la figure IV.5).

$\overline{RD} = 0$: Lecture mémoire ou périphérique.

$\overline{RD} = 1$: Ecriture mémoire ou périphérique.

\overline{RD}	OPERATION	Sens du transfert
0	lecture	B-----> A
1	Ecriture	A-----> B

Tableau IV.1 Fonction et sens du bus de données

IV.4.2 Bufferisation du bus d'adresses

Ce bus qui est unidirectionnel et nécessitant donc un buffer unidirectionnel. Cependant on pourrait toujours utiliser le 74 LS 245 en forçant la pin DIR à **1** de façon à avoir un transfert de données dans un seul sens.

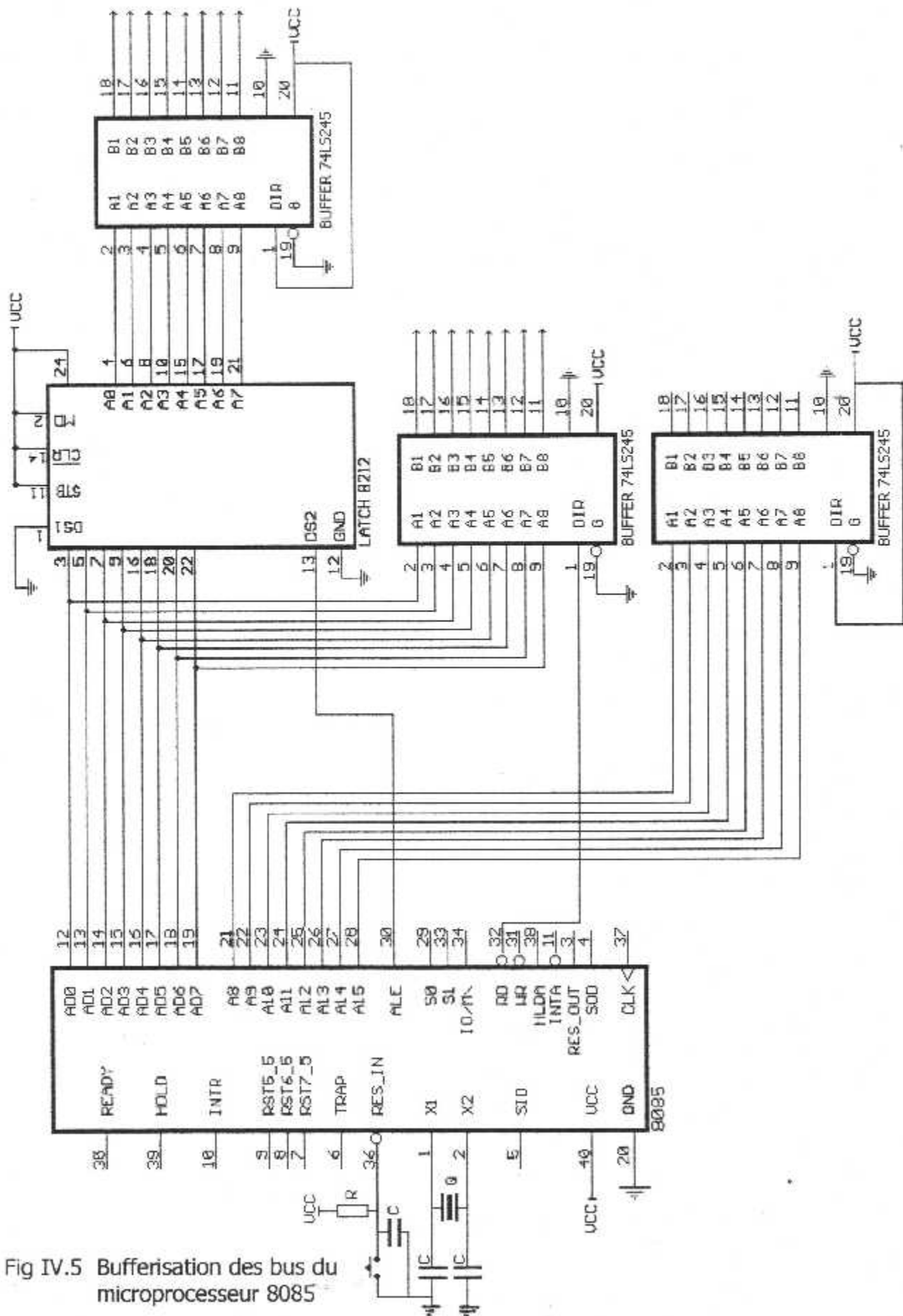


Fig IV.5 Bufferisation des bus du microprocesseur 8085

IV.5 Décodage du bus de contrôle

Le choix fondamental dans le domaine de traitement des E/S consiste à considérer les E/S comme des éléments distincts de la mémoire (E/S spécialisées) ou comme des éléments similaires aux cellules mémoires et donc à les adresser de la même manière (E/S projetées en mémoire). Notre choix s'est porté alors sur E/S spécialisées pour les raisons suivantes :

1. Introduction d'instructions spéciales relatives aux E/S (IN, OUT).
2. A chacune de ces deux instructions on peut associer un espace des adresses des périphériques avec une adresse courte (un octet).
3. L'espace des adresses en mémoire n'est pas affecté par les E/S.
4. L'exécution des instructions (IN et OUT) génère des signaux de commande spécifiques adéquats pour la lecture ou l'écriture.
5. Les listings des programmes sont généralement plus clairs puisque les instructions spéciales d'E/S sont immédiatement reconnaissables.

Sur les pins 34,32 et 31 du 8085, on dispose respectivement des signaux de commande suivants :

(34) $\overline{IO/\overline{M}}$: Sélection de la destination (mémoire ou E/S).

(32) \overline{RD} : lecture.

(31) \overline{WR} : écriture ; écriture mémoire (si $\overline{IO/\overline{M}} = 0$) ; écriture E/S (si $\overline{IO/\overline{M}} = 1$)
donc le décodage a pour but d'obtenir les commandes suivantes :

1. \overline{MEMRD} : lecture mémoire (Memory Read).
2. \overline{MEMWR} : écriture mémoire (Memory Write).
3. \overline{IORD} : lecture périphérique (I/O Read).
4. \overline{IOWR} : écriture périphérique (I/O Write).

L'opération est réalisée suivant la table de vérité suivante :

ENTRÉES			SORTIES				COMMANDE
\overline{RD}	\overline{WR}	$\overline{IO/\overline{M}}$	\overline{MEMRD}	\overline{MEMWR}	\overline{IORD}	\overline{IOWR}	
L	H	L	L	H	H	H	Lecture périphérique <i>mémoire</i>
H	L	L	H	L	H	H	Écriture mémoire
L	H	H	H	H	L	H	Lecture périphérique
H	L	H	H	H	H	L	Écriture périphérique

Tableau IV.2 table de vérité du décodage du bus de contrôle

D'après la table de vérité, on remarque que les signaux \overline{WR} et \overline{RD} sont complémentaires ce qui est normal puisque le microprocesseur ne pourrait pas ordonner une lecture et une écriture en même temps.

Il vient donc d'écrire $\overline{RD} = \overline{WR}$

Les opérations précédentes deviennent donc :

$$\overline{MEMRD} = \overline{IO/\overline{M}} + \overline{RD}$$

$$\overline{MEMWR} = \overline{IO/\overline{M}} + \overline{WR}$$

$$\overline{IORD} = \overline{IO/\overline{M}} + \overline{RD}$$

$$\overline{IOWR} = \overline{IO/\overline{M}} + \overline{WR}$$

IV.6 Interfaçage des circuits périphériques avec le microprocesseur

Comme le montre le schéma synoptique de la figure IV.6, l'ensemble des périphériques entourant notre système à microprocesseur se résume à :

- ✓ Une EPROM pour les routines systèmes.
- ✓ Une RAM pour le stockage momentané des données.
- ✓ Un clavier.
- ✓ Des monostables.

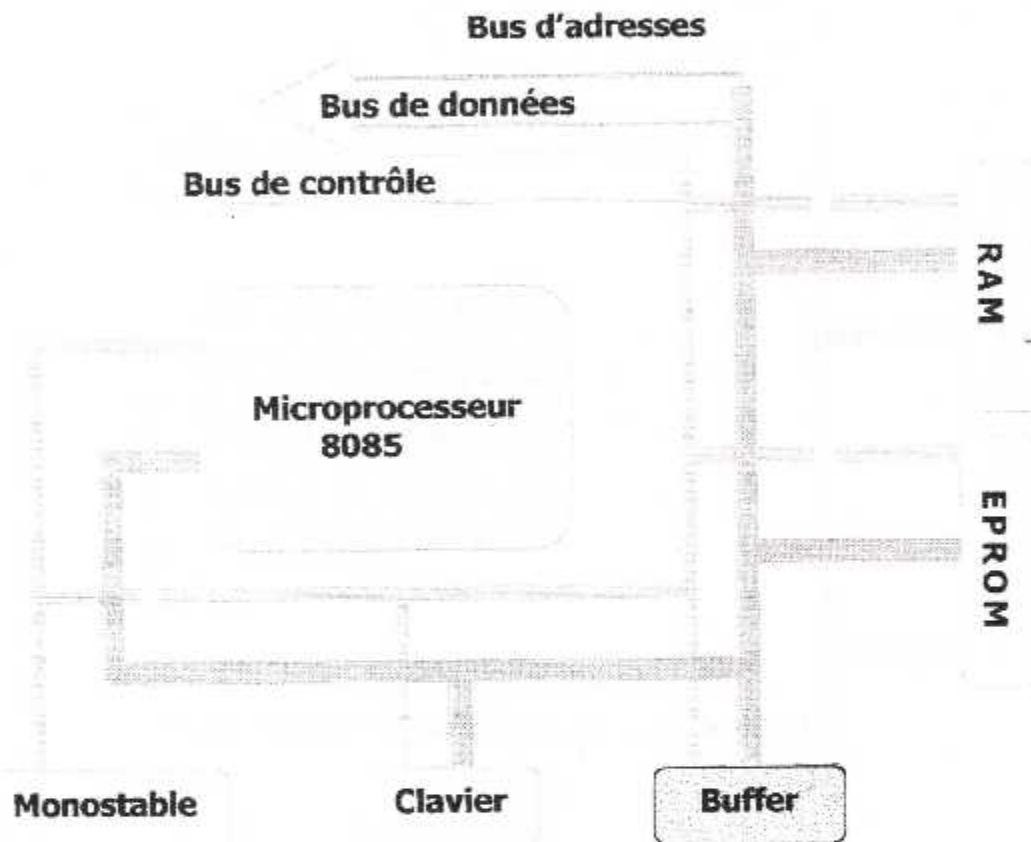


Fig IV.6 Synoptique de la carte mère

IV.6.1 Interfaçage de l'EPROM

L'EPROM retenue pour le présent projet est la 2764 (8 kilo-octet disponible, pour les routines systèmes).

Elle est interfaçée au microprocesseur de la manière suivante : (voir la figure IV.7)

- Les huit lignes du bus de données du microprocesseur (AD_0-AD_7) sont connectées aux bus de données (D_0-D_7) de l'EPROM.
- Les treize lignes du bus d'adresses (A_0-A_{12}) sont connectées aux bus d'adresses de l'EPROM (il est à rappeler que la capacité de l'EPROM est de 8K, donc 8192 cases mémoires).
- La sortie décodée \overline{MEMRD} ou ($\overline{IO/M} + \overline{RD}$) du microprocesseur est connectée à l'entrée \overline{OE} (output enable) de l'EPROM.
- L'entrée \overline{CE} ou \overline{CS} (chip select) de l'EPROM est connectée à la ligne A_{15} du microprocesseur. La sélection de l'EPROM se fait par la mise au niveau bas la ligne A_{15} .

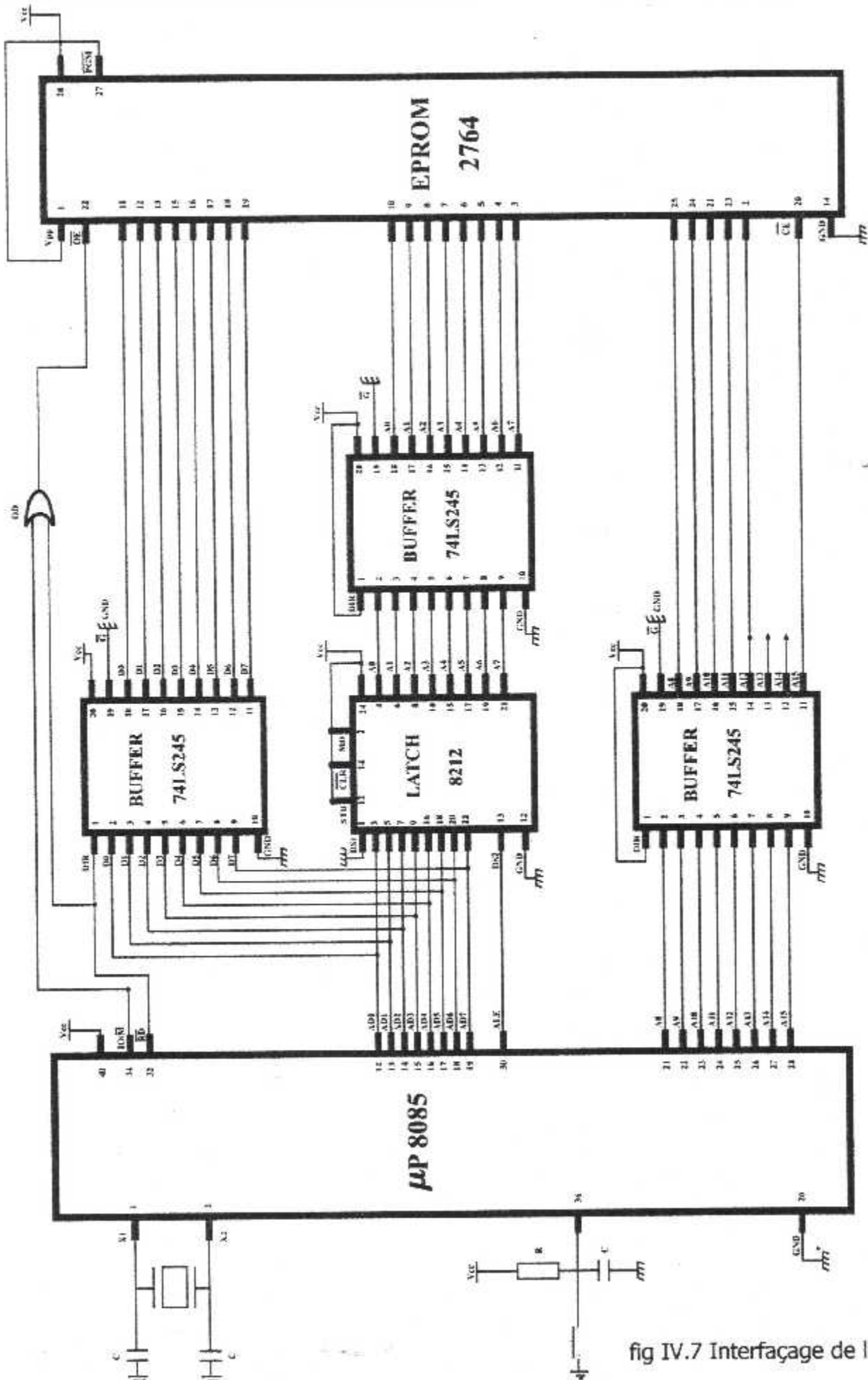


fig IV.7 Interfaçage de l'EPROM

IV.6.2 Interfaçage de la RAM

La RAM retenue pour ce projet est la RAM 6116 ; de type statique (afin de ne pas encombrer le système d'un module de rafraîchissement, indispensable aux mémoires dynamiques), son interfaçage avec le système se fait de la manière suivante (voir la figure IV.8) :

- Le bus de données du microprocesseur est connecté au bus de données de la RAM.
- Les onze lignes du bus d'adresses du microprocesseur ($A_0 - A_{10}$) sont de même connectées aux bus d'adresses de la RAM.
- La sortie $\overline{\text{MEMRD}}$ ou $(\text{IO}/\overline{\text{M}} + \overline{\text{RD}})$ décodée est connectée à l'entrée de validation de sortie du boîtier RAM ($\overline{\text{OE}}$).
- La sortie décodée $\overline{\text{MEMWR}}$ ($\text{IO}/\overline{\text{M}} + \overline{\text{WR}}$) est connectée à son tour à l'entrée R/W de la RAM.

Notons bien à ce niveau que R/W est une entrée de la RAM, cela signifie que quand le microprocesseur n'active pas la sortie $\overline{\text{MEMWR}}$ avec un niveau bas, l'écriture n'est pas possible, et la lecture n'est pas possible que si la sortie $\overline{\text{MEMRD}}$ soit à niveau bas.

- A la fin, l'entrée $\overline{\text{CS}}$ du boîtier de la RAM est connectée directement à la ligne A_{14} du microprocesseur. Alors pour la sélection de la RAM on met au niveau haut la ligne A_{14} .

Les deux opérations possibles entre le microprocesseur et la RAM sont :

➤ La lecture des données de la RAM

Cette opération se déroule de la façon suivante :

- Initialiser les lignes d'adresses $A_0 - A_{14}$.
- Mettre $\overline{\text{MEMRD}}$ à l'état bas délivrant ainsi l'ordre de lecture.

➤ L'écriture des données dans la RAM

Cette opération nécessite les mêmes étapes précédentes sauf que $\overline{\text{MEMWR}}$ doit être mise à l'état bas.

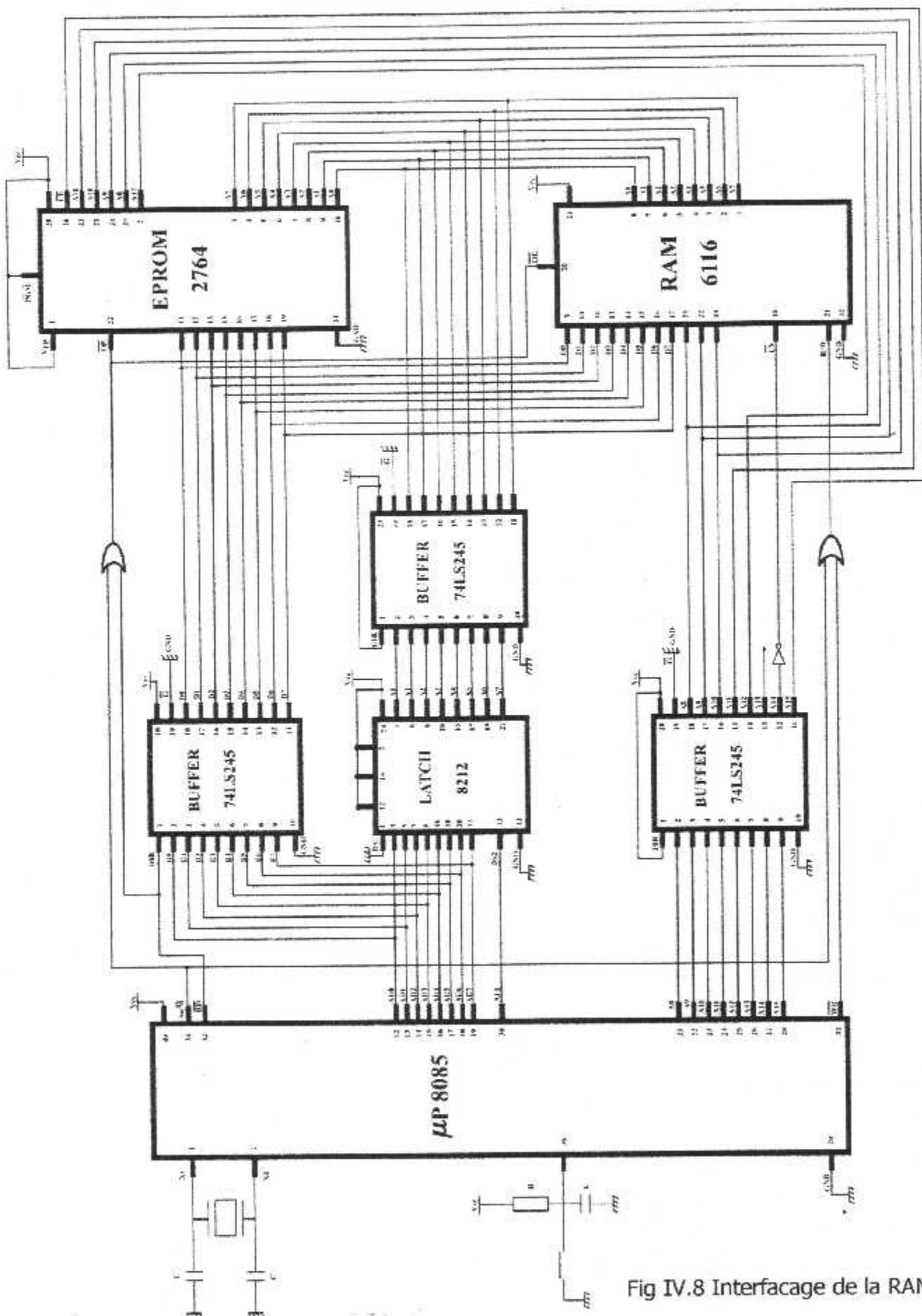


Fig IV.8 Interfacage de la RAM

IV-6-3 Interfaçage du clavier

• Objectif :

Il est important de faciliter l'échange entre l'opérateur et le processeur et ceci par le biais d'un clavier. Ce dernier soit en hexadécimale interrupteur, ou soit en ASCII. Le principe de son interfaçage reste le même. Le travail sera plus simple pour le clavier hexadécimal.

Dans notre travail, le clavier se présente sous l'aspect de 14 touches (Voir figure VI.9), les fonctions des différentes touches sont :

- ◆ 11 touches pour différents groupes de modes.
- ◆ Une touche pour la validation.
- ◆ Une touche pour l'entrelacement récurrence par récurrence X/Y.
- ◆ Une touche pour l'entrelacement par tour d'antenne XY (qui ne fait pas partie de notre travail).

L'action sur une touche met tout simplement en contact deux fils conducteurs qui font partie d'un interrupteur interne du clavier.

Le principe est simple : lorsque l'opérateur appuie sur une touche, le processeur doit l'identifier c'est à dire placer dans une mémoire un mot binaire traduisant la touche sollicitée.

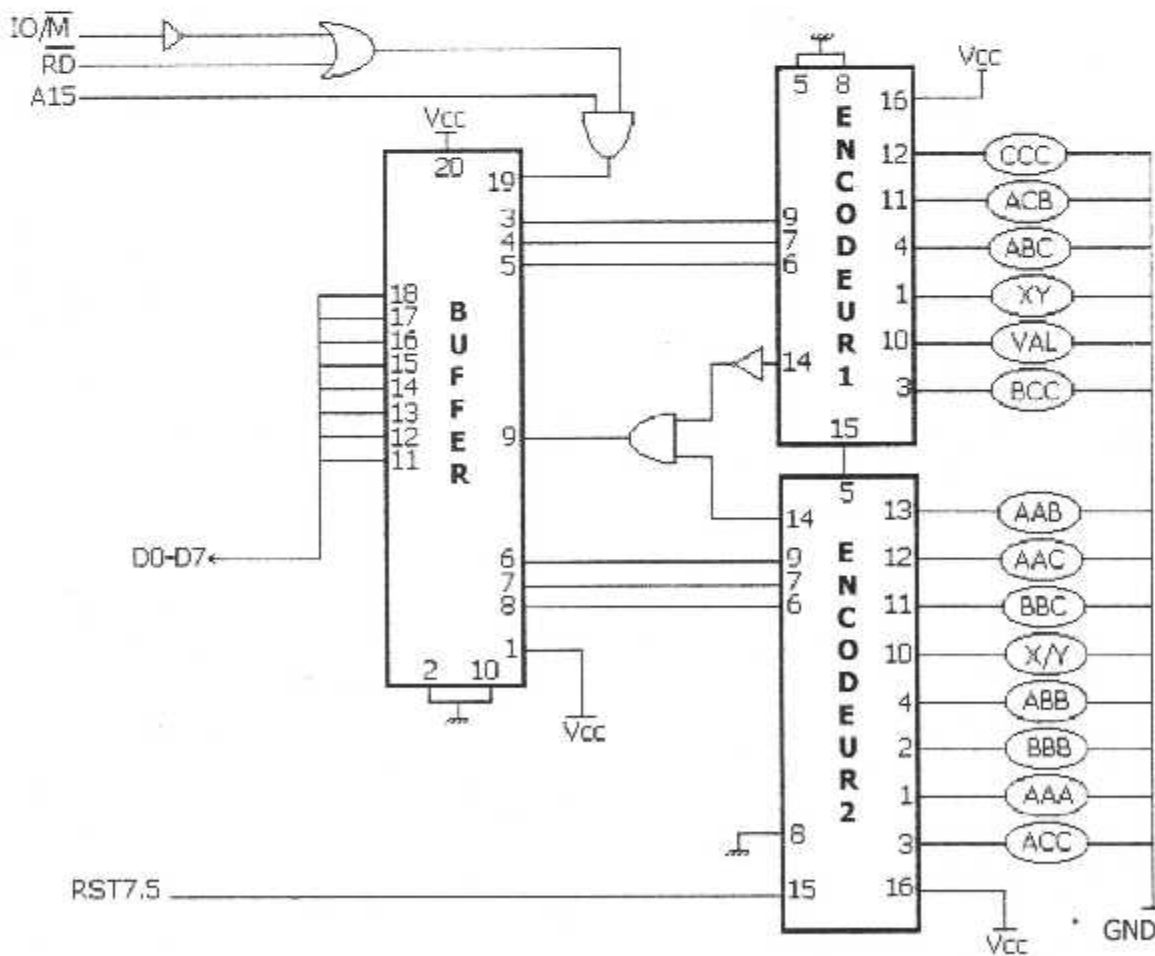


Fig IV.9 Interfaçage du clavier

- **Principe :**

Le premier problème qui se présente est d'avertir le processeur quand le clavier est sollicité afin qu'il puisse procéder à la reconnaissance de la touche enfoncée. Deux méthodes s'imposent :

- Scrutation. (polling)
- Interruption. (voir chapitre III)

Nous avons utilisé une procédure d'interruption ; dès la prise en compte de cette interruption le processeur fera démarrer la procédure d'identification, pour cela un buffer 3 états va jouer le rôle du port d'entrée, l'ordre du transfert lui parvient du processeur qui en activant la ligne (lecture E/S) avec l'adresse adéquate sollicite une lecture du clavier.

Si une touche est enfoncée, le clavier génère le code qui lui correspond, seulement ce code n'est pas donné sur un octet, ce qui pose un problème pour la prise en compte de ce code par le processeur puisque ce dernier travaille avec un bus de données de 8 bits.

Pour résoudre ce problème, deux solutions sont possibles, à savoir :

- * Le multiplexage du clavier.
- * L'encodage.

La première méthode nécessite un organe d'entrées/sorties programmable qui alourdira considérablement le logiciel.

La deuxième méthode utilise des encodeurs dont les sorties sont connectées à un buffer 3 états permettant l'interfaçage avec le bus de données du système.

Pour notre réalisation nous avons adopté la deuxième méthode pour des raisons d'optimisation car la première méthode nécessite un PPI qui est un composant spécialisé ayant un prix nettement plus important que les encodeurs 74 LS148 et le buffer 3 états ; en plus de cela le travail logiciel est nettement plus compliqué que celui adopté par la deuxième méthode.

Cependant pour chaque touche enfoncée, on aura un code correspondant à cette touche activée. Ce code ainsi trouvé en entrée du buffer est un ensemble de bits contenant des "0" et des "1".

Le tableau suivant représente les différentes touches correspondantes aux différentes combinaisons possibles des groupes ainsi que le code en hexadécimale de chaque touche.

Touche	code
AAA	3E
BBB	2E
CCC	FA
ACB	FC
AAB	4E
BBC	6E
AAC	5E
ABB	0E
ACC	1E
BCC	F2
ABC	F0
X/Y	7E
XY	F6
VAL	FE

Tableau IV.3 Touches et codes correspondants

• Génération de l'interruption

La touche appuyée doit automatiquement générer une demande d'interruption au processeur. La méthode la plus évidente est de combiner toutes les touches avec une logique combinatoire simple de manière que si n'importe quelle touche est enfoncée, une demande d'interruption est envoyée automatiquement au processeur (voir figure IV.9).

Une deuxième méthode consiste à utiliser une sortie du deuxième encodeur (OE). Cette sortie des touches hexadécimales est connectée directement à l'entrée demande d'interruption (RST 7.5).

Le choix de l'interruption (RST 7.5) pour les touches de commande est parfaitement justifié par :

- * RST 7.5 est active à un front montant.
- * La demande d'interruption est mémorisée dans un latch interne du processeur, chose qui n'est pas accordée aux entrées d'interruption (RST 6.5) et (RST 5.5)

IV.6.4 Interfaçage des monostables

Les monostables retenus pour notre travail, qui sont au nombre de 4 sont de type 74LS121, 3 d'entre eux sont interfacés au microprocesseur de la manière suivante :

- * La sortie IO/\bar{M} et l'adresse A_{14} sont connectées à l'entrée du monostable N°1 et du monostable N°2.
- * La sortie IO/\bar{M} et l'adresse A_{13} sont connectées à l'entrée du monostable N°4.
- * La sortie du monostable N°2 est une entrée du monostable N°3 à travers un inverseur. Cet interfaçage est bien indiqué dans la figure IV.10.

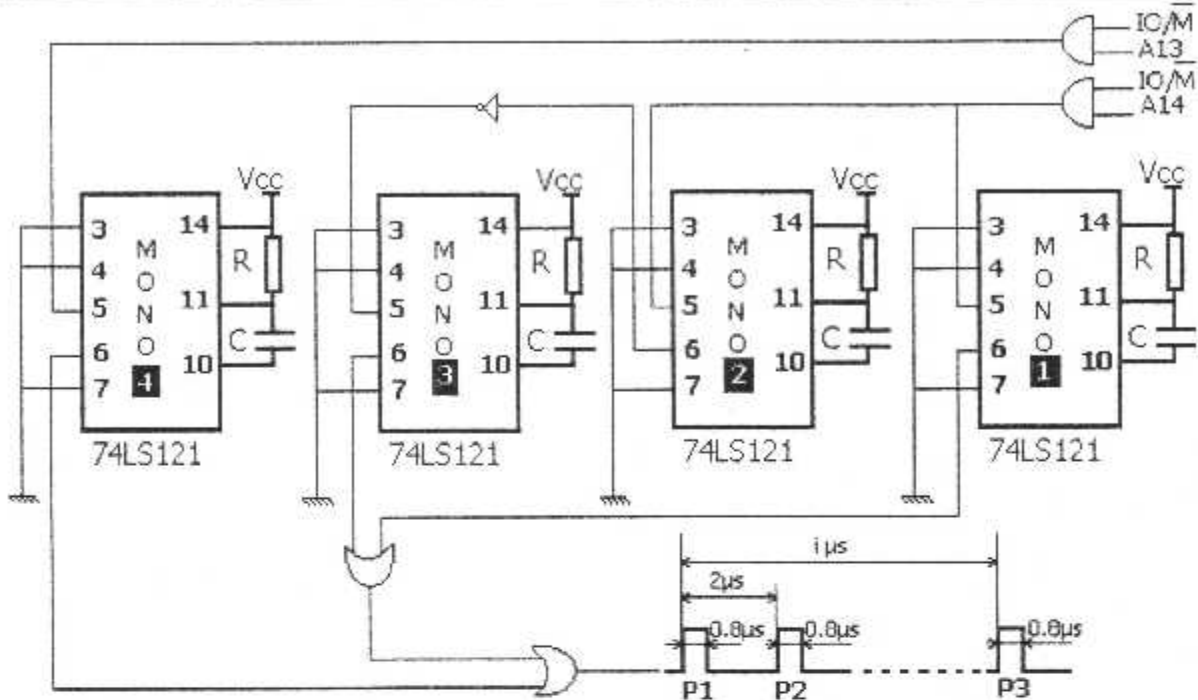


Fig IV.10 Interfaçage des monostables

IV.7 Fonctionnement des monostables

- Le monostable 74LS121 est un circuit déclenché par une impulsion, en sortie il génère une impulsion dont la durée T dépend d'un circuit RC, cette durée est donnée par la relation suivante : $T = RC \ln 2$

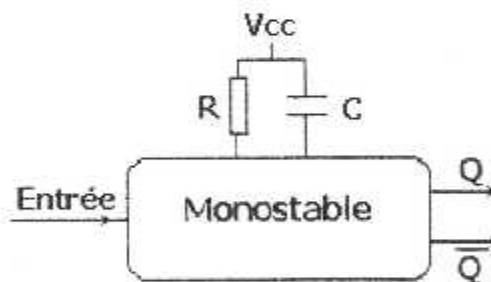


Fig IV.11 schéma synoptique du 74LS121

- Pour générer un signal de 3 impulsions **P1**, **P2** et **P3**, on a besoin de 4 monostables de types 74LS121. Le fonctionnement de ces monostables est le suivant :
 - Le premier monostable génère une impulsion de largeur $0.8 \mu s$ (figure IV.12.a) grâce à une résistance et un condensateur dont les valeurs sont les suivantes : **$R = 11 K\Omega$** , **$C = 100\mu F$** tel que $0.8 = RC \ln 2$.

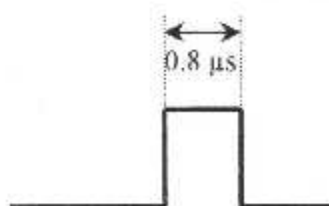


Fig IV.12.a

- Le deuxième monostable génère une impulsion ayant une largeur de $2 \mu\text{s}$ (figure IV.12.b) alors :

$$R=27 \text{ K}\Omega, C=100\mu\text{F} \text{ tel que } 2=RC \ln 2.$$

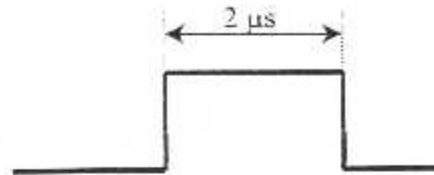


Fig IV.12.b

- L'entrée du monostable N°3 est connectée à la sortie du monostable N°2 à travers un inverseur qui inverse l'impulsion $2 \mu\text{s}$, le monostable N°3 génère une impulsion de $0.8 \mu\text{s}$ (figure IV.12.c) donc $R=11 \text{ k}\Omega$ et $C=100 \mu\text{F}$. Ce dernier délivre un signal de la forme suivante :

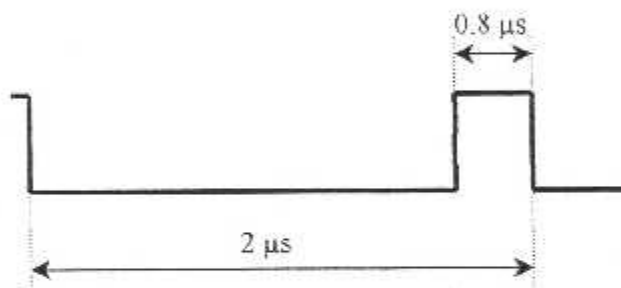


Fig IV.12.c

- En faisant la somme entre la sortie du monostable N°1 et la sortie du monostable N°3, à la sortie de la porte OU, on aura deux impulsions **P1** et **P2** séparées l'une de l'autre de $2 \mu\text{s}$ (Figure IV.12.d).

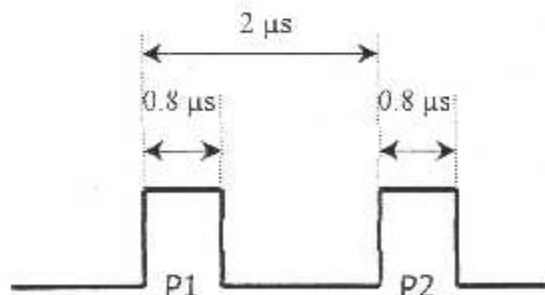


Fig IV.12.d

- Le dernier monostable fournit une impulsion de $0.8 \mu\text{s}$, donc $R=11 \text{ K}\Omega$ et $C=100\mu\text{F}$, qui représente l'impulsion **P3**.
L'activation de ce monostable se fait par programme selon le mode désiré.
- La sortie du monostable N°4 générée une impulsion **P3**. L'activation du monostable N°4 se fait par programme selon le mode désiré.
- On fait la somme (porte OU) entre le signal de deux impulsions **P1** et **P2** avec l'impulsion **P3** pour obtenir un signal de trois impulsions (Figure IV.12.e) c'est ce qu'on appelle « mode interrogation ».

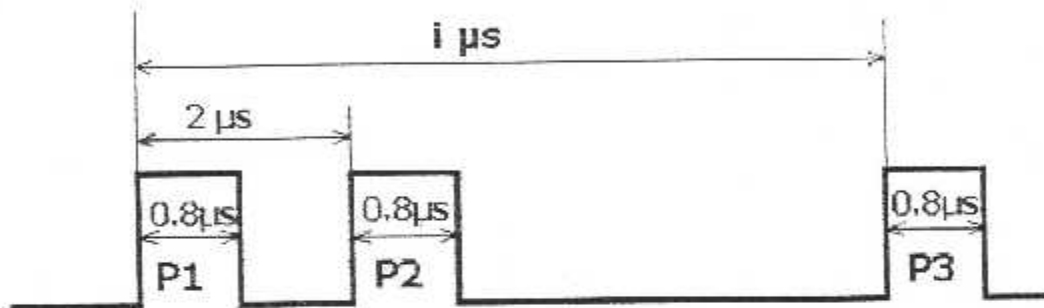
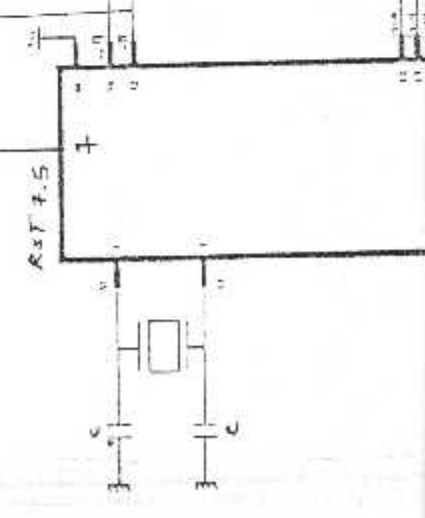
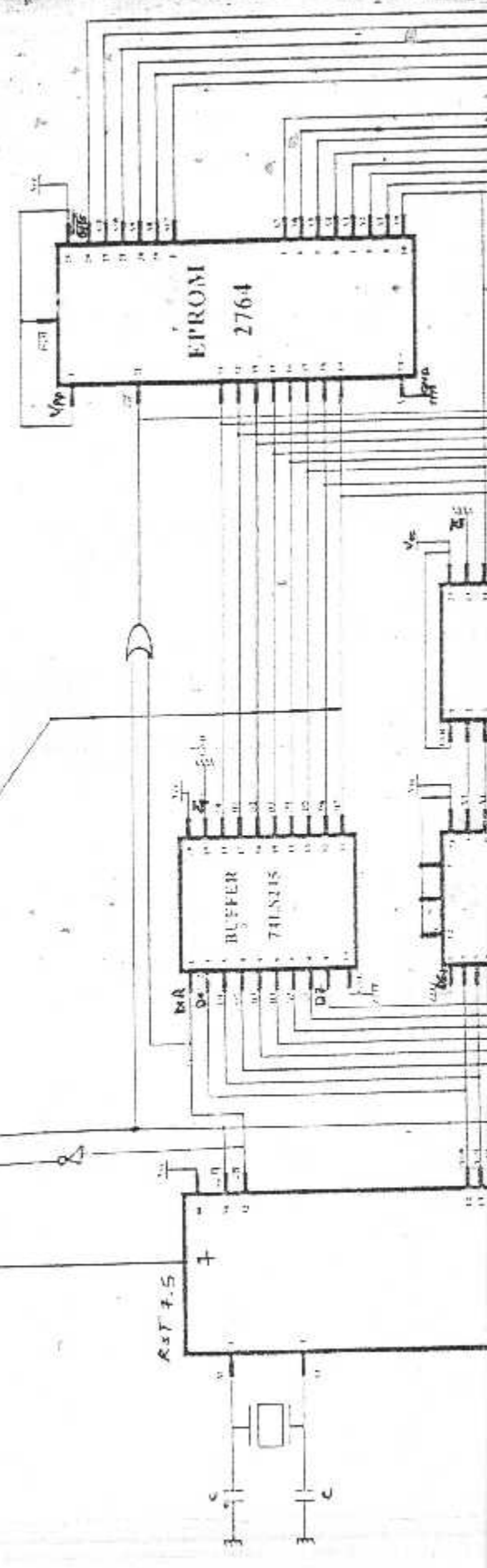
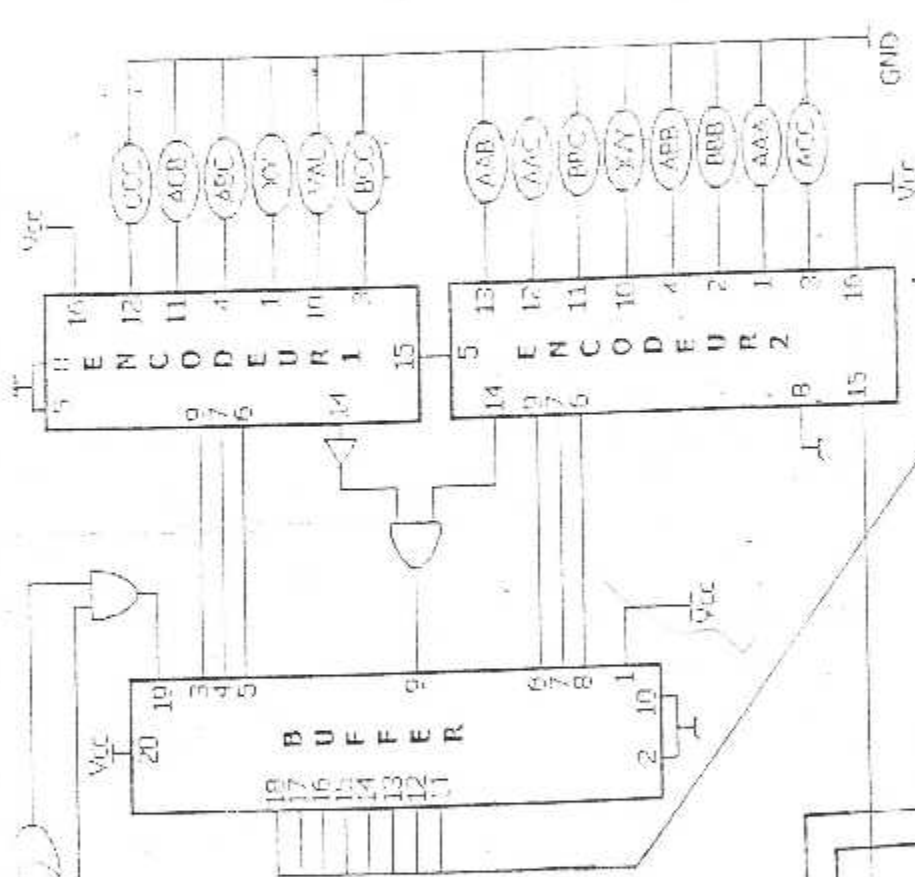
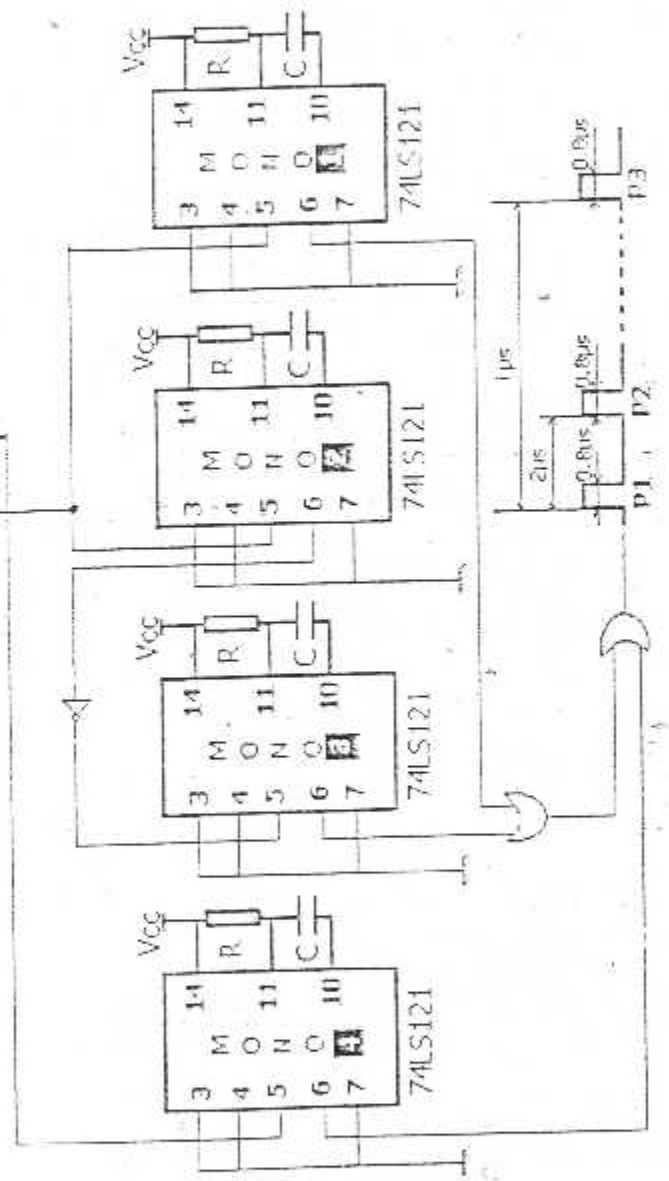
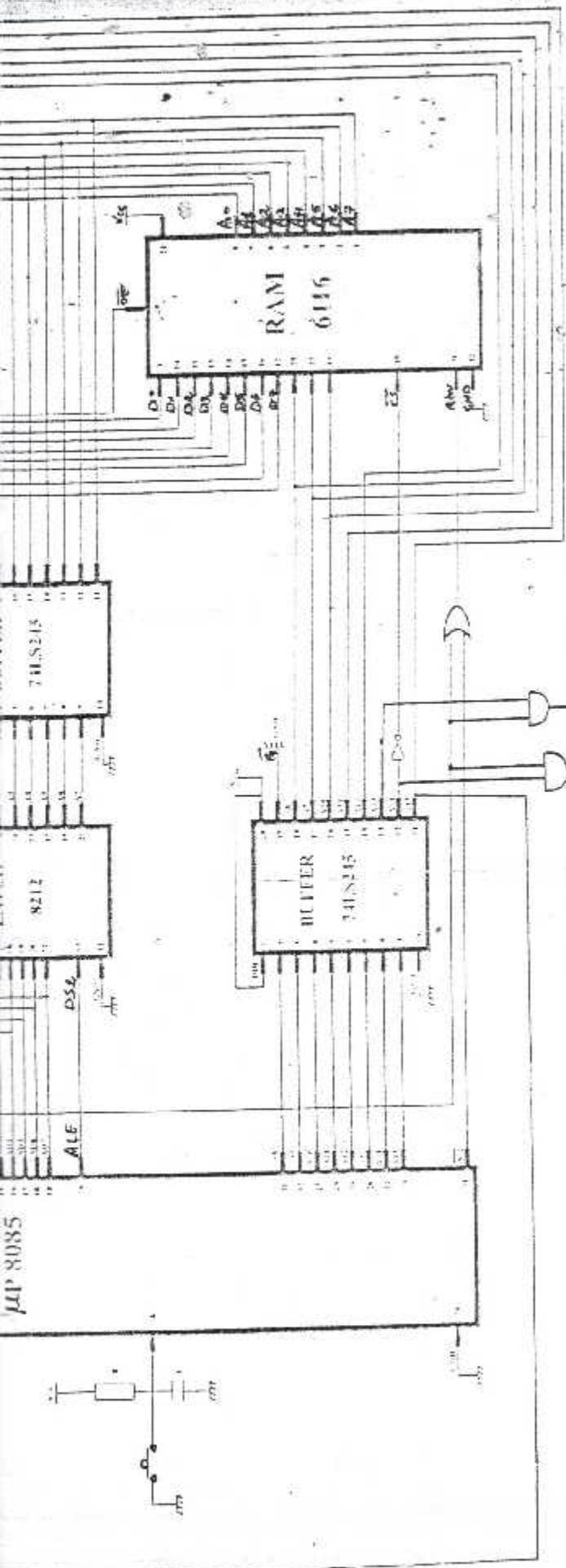


Fig IV.12.e





CONCLUSION GENERALE

CONCLUSION GENERALE

Dans le cadre de notre projet de fin d'études nous avons pour but de réaliser une carte codeur radar secondaire à base de microprocesseur 8085. Cette carte permet de générer les modes d'interrogation civil, à savoir le mode A, B, et C. Les intervalles de temps séparant les impulsions P1 et P3 de chaque mode sont respectivement 8 μ s, 17 μ s et 21 μ s selon les normes de l'OACI.

Le codeur réalisé permet également d'élaborer l'entrelacement par récurrence de ces modes générés.

La carte codeur peut être réalisée à base de circuits logiques TTL, mais nous avons préféré la réaliser à base du microprocesseur 8085 de INTEL puisque cette méthode permet non seulement la génération de n'importe quelle combinaison de modes à transmettre mais aussi une utilisation simple et facile, le tout en un temps réduit. Certes, la mise en œuvre de notre projet n'a pas été facile et la difficulté qui a engendré la réalisation de notre carte codeur est due essentiellement à la complexité du circuit et notre manque d'expérience en ce qui concerne la manipulation de certains composants (charge de l'EPROM et choix du clavier adéquat).

Enfin, la réalisation de ce projet nous a permis d'acquérir des nouvelles connaissances aussi bien dans le domaine de notre filière (spécialement la détection radar) mais aussi dans un domaine très vaste tel que l'électronique, nous espérons que notre travail servira de repère pour les prochaines promos qui souhaiteront développer notre projet.

ANNEXE

A

Brochage du microprocesseur 8085

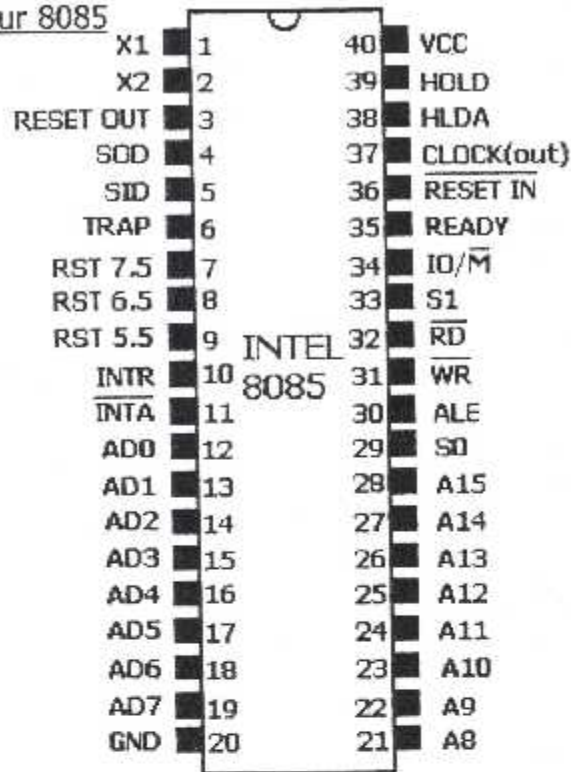


Tableau A.1 Signification des broches du microprocesseur 8085

BROCHE	DESCRIPTION	TYPE
AD0 – AD7	Bus d'adresses/données	Bidirection, 3 états.
A8 – A15	Bus d'adresses	Sortie, 3 états.
ALE	Validation de verrou d'adresse	Sortie.
\overline{RD}	Commande de lecture	Sortie, 3 états.
\overline{WR}	Commande d'écriture	Sortie, 3 états.
IO/M	Indicateur E/S ou mémoire	Sortie, 3 états.
S0, S1	Indicateur d'état de bus	Sortie.
READY	Requête de mode attente	Entrée.
SID	Entrée de données sérielles	Entrée.
SOD	Sortie de données sérielles	Sortie.
HOLD	Requête d'attente	Entrée.
HLDA	Accusé de réception d'attente	Sortie.
INTR	Requête d'interruption	Entrée.
TRAP	Requête d'interruption non masquable	Entrée.
RST 5.5	Requête d'interruption matérielle vectorisée	Entrées.
RST 6.5		
RST 7.5		
\overline{INTA}	Accusé de réception d'interruption	Sortie.
$\overline{RESETIN}$	Réinitialisation système	Entrée.
RESET OUT	Réinitialisation périphérique	Sortie.
X1, X2	Connexion quartz ou RC	Entrée.
CLK	Signal d'horloge	Sortie.
VCC, GND	Alimentation, terre	

Brochage de la RAM 6116

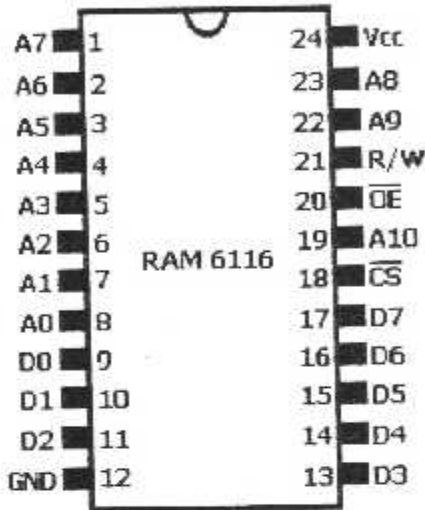
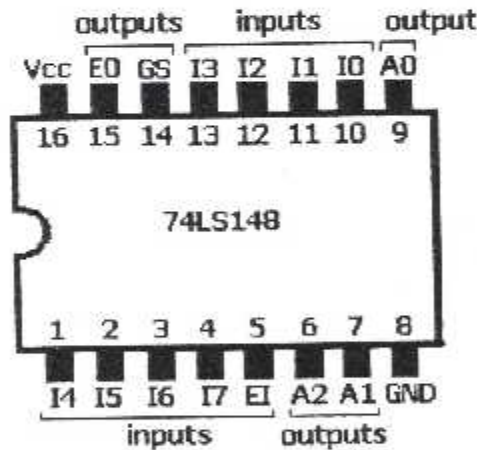


Tableau A.2 Signification des broches de la RAM 6116

BROCHE	SIGNIFICATION
A0 - A10	BUS D'ADRESSES
D0 - D7	BUS DE DONNEES
\overline{CS}	CHIP ENABLE (autorisation du boîtier)
\overline{OE}	OUTPUT ENABLE (autorisation de sortie)
R/W	INPUT ANABLE (autorisation d'entrée)

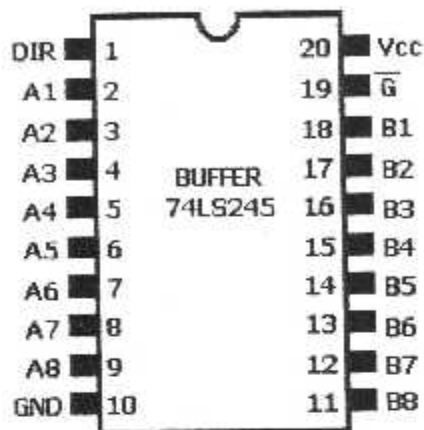
Brochage de l'encodeur 74LS148



INPUTS									OUTPUTS				
EI	I0	I1	I2	I3	I4	I5	I6	I7	A2	A1	A0	GS	E0
H	X	X	X	X	X	X	X	X	H	H	H	H	H
L	H	H	H	H	H	H	H	H	H	H	H	H	L
L	X	X	X	X	X	X	X	L	L	L	L	L	H
L	X	X	X	X	X	X	L	H	L	L	H	L	H
L	X	X	X	X	L	H	H	H	L	H	H	L	H
L	X	X	X	L	H	H	H	H	H	L	L	L	H
L	X	X	L	H	H	H	H	H	H	L	H	L	H
L	X	L	H	H	H	H	H	H	H	H	L	L	H
L	L	H	H	H	H	H	H	H	H	H	H	L	H

Tableau A.3 Table de vérité du fonctionnement de l'encodeur

Brochage du buffer 74LS245



Enable \overline{G}	Controle DIR	operation
L	L	B----->A
L	H	A----->B
H	X	isolation

Tableau A.4 Table de vérité du fonctionnement du buffer 74LS245

Brochage du latch 8212

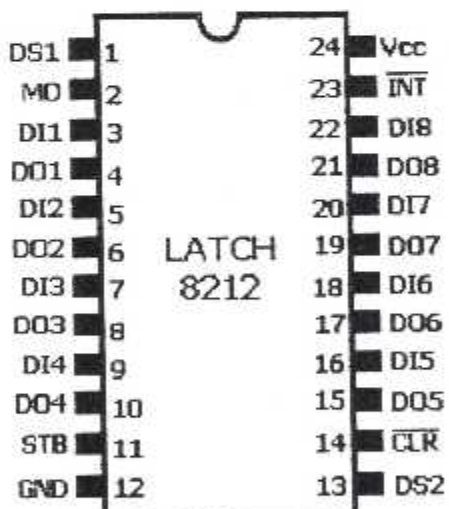
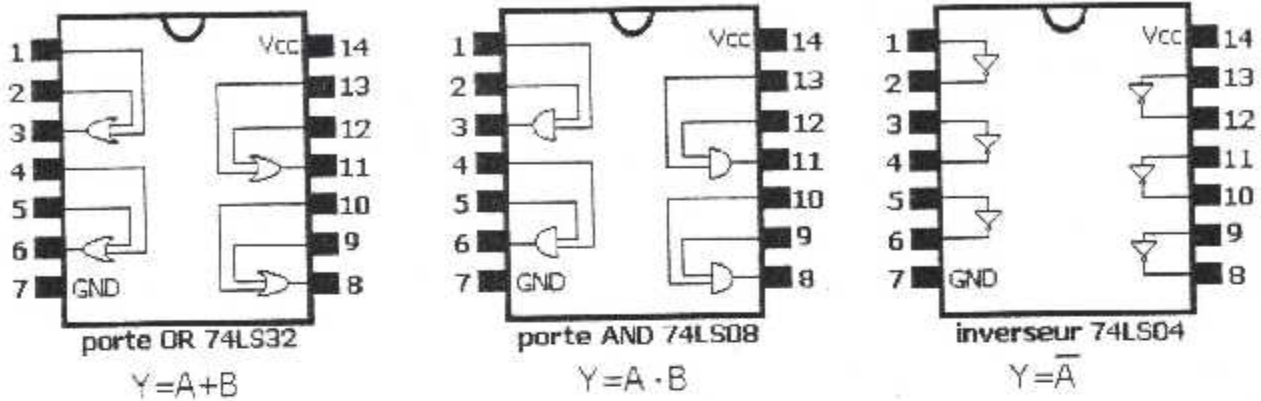


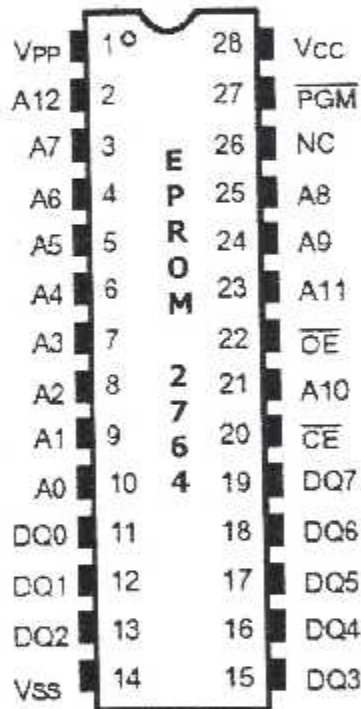
Tableau A.5 Signification des broches du latch 8212

Broche	Signification
DI1 - DI8	DATA IN
DO1 - DO8	DATA OUT
DS1 - DS2	DEVICE SELECT
MD	MODE
STB	STROB
\overline{INT}	INTERRUPT
\overline{CLR}	CLEAR

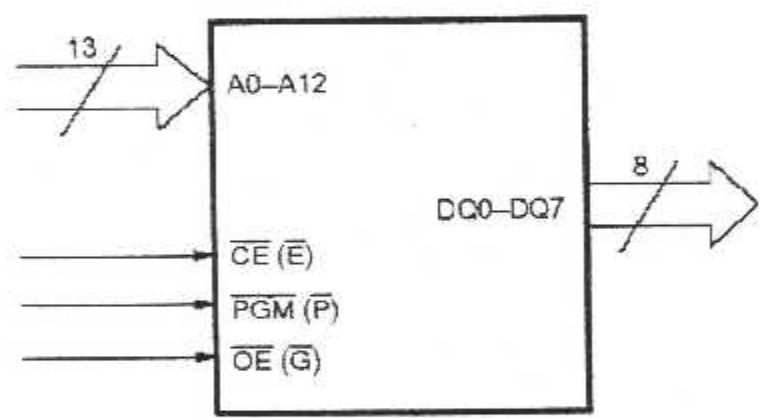
Brochage des portes logiques : A et B sont les entrées, Y est la sortie.



Brochage de l'EPROM 2764



LOGIC SYMBOL



A0-A12	Address Inputs
\overline{CE} (E)	Chip Enable
DQ0-DQ7	Data Inputs/Outputs
\overline{OE} (G)	Output Enable Input
\overline{PGM} (P)	Program Enable Input
Vcc	Vcc Supply Voltage
Vpp	Program Voltage Input
Vss	Ground
NC	No Internal Connection
DU	No External Connection (E

Tableau A.6 Signification des broches de l'EPROM 2764

ANNEXE

B

B.1 Organigramme du programme principal

Dès la remise à zéro du microprocesseur, le programme principal débute à l'adresse 0000 hex . La figure B.1 présente les différentes étapes assurées par le programme principal.

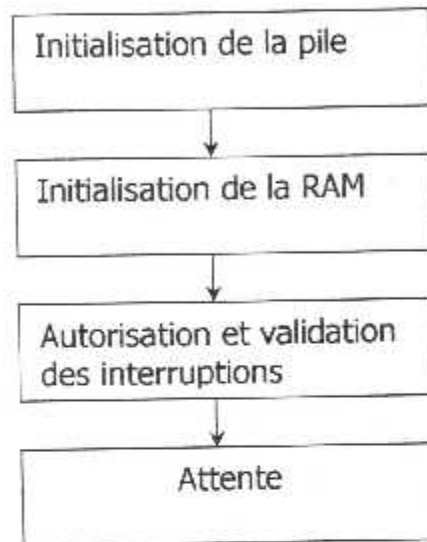


Fig B.1 organisation du programme principal

B.2 Organigramme de saisie et stockage des touches

A chaque fois qu'une touche est appuyée, une interruption (RST 7.5) est déclenchée au niveau du microprocesseur. La routine de saisie permet de brancher le microprocesseur à une adresse correspondante à l'interruption qui vient d'être déclenchée.

Cette adresse contient le sous-programme d'interruption qui ordonne au microprocesseur la lecture de la donnée présentée au buffer.

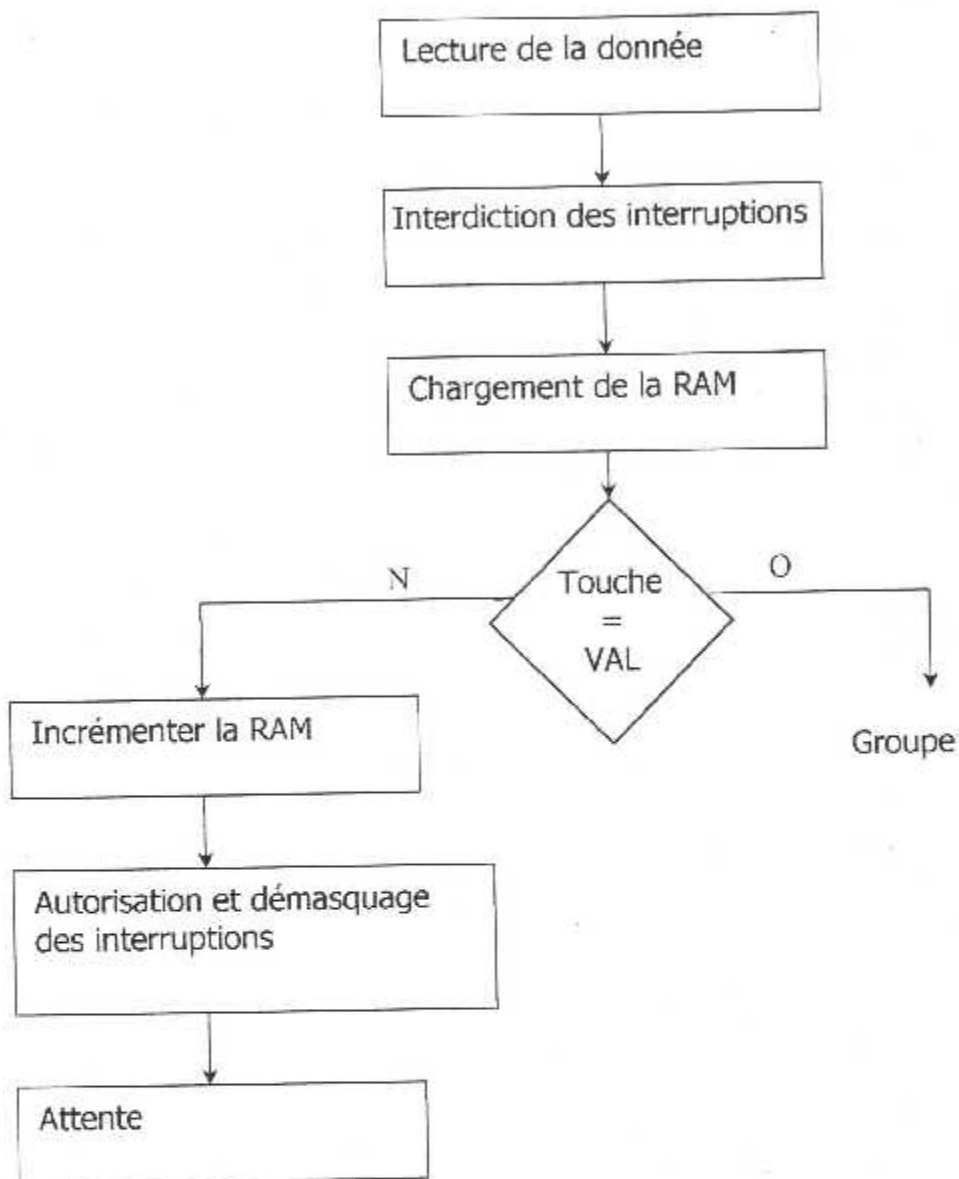


Fig B.2 Organigramme de saisie et stockage des touches

B.3 Organigramme de génération des modes

Cet organigramme permet de générer que les trois modes civils A, B et C. L'intervalle de temps entre P1 et P2 est généré par les monostables 1, 2 et 3, tandis que l'intervalle de temps entre P1 et P3 est généré grâce à une temporisation faite par programme qui déclenche le monostable 4.

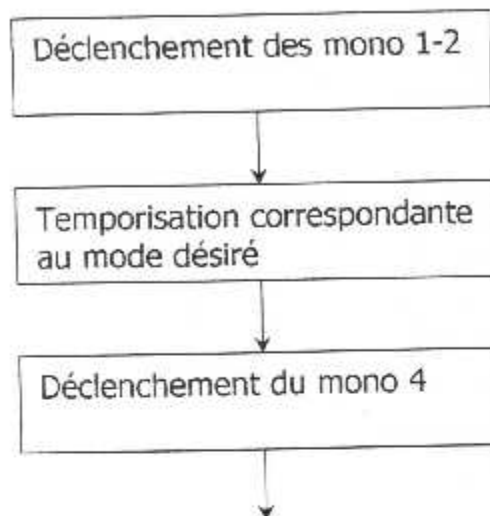


Fig B.3 Organigramme de génération des modes

B.4 Organigramme de génération des groupes

Un groupe est constitué de 3 modes séparés par un intervalle de temps qui correspond à la période de récurrence.

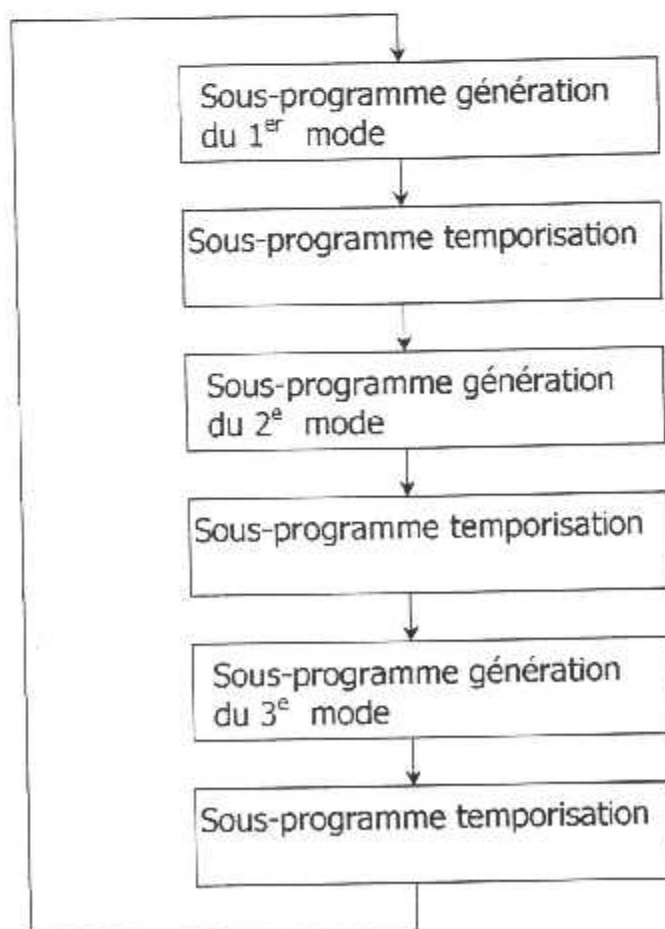
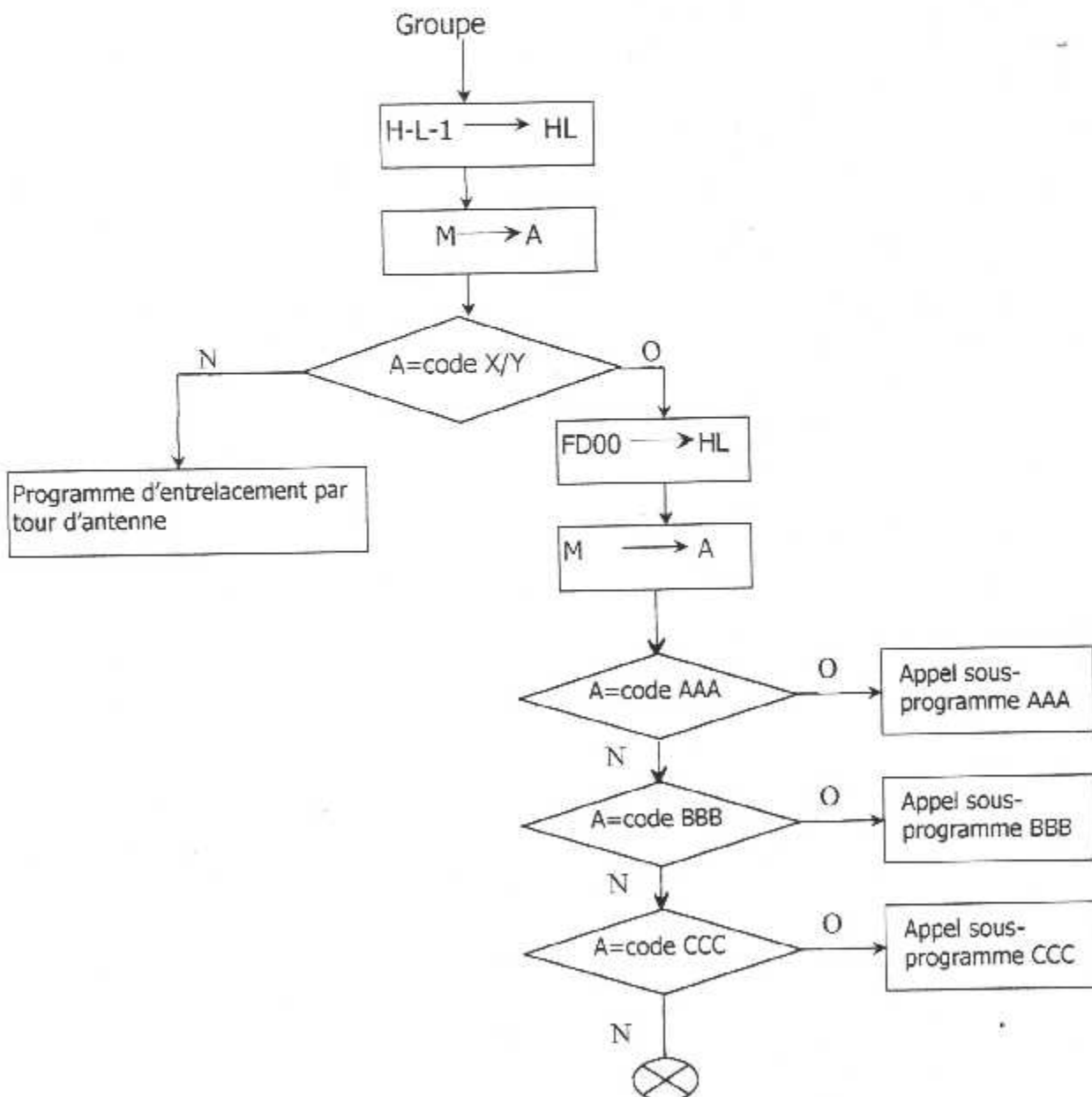


Fig B.4 Organigramme de génération d'un groupe

Le sous-programme sert à définir la fréquence de récurrence ou bien la période de récurrence qui est l'intervalle de temps séparant 2 modes successifs.

B.5 Organigramme d'entrelacement des modes par récurrence

Après que l'opérateur appui sur la touche correspondante au groupe choisi ensuite sur la touche (X/Y) correspondant au sélection de ce type d'entrelacement et après sur la touche (VAL) signifiant la fin du transfert des données, le processeur compare le code de la première touche appuyée avec le code de la touche (X/Y) stocké dans l'EPROM. S'il y a correspondance, le processeur compare le code de la 2^e touche appuyée avec les codes des différents groupes stockés dans l'EPROM. Dès qu'il y a correspondance avec le code d'un groupe, il fait appel au sous-programme génération de ce groupe.



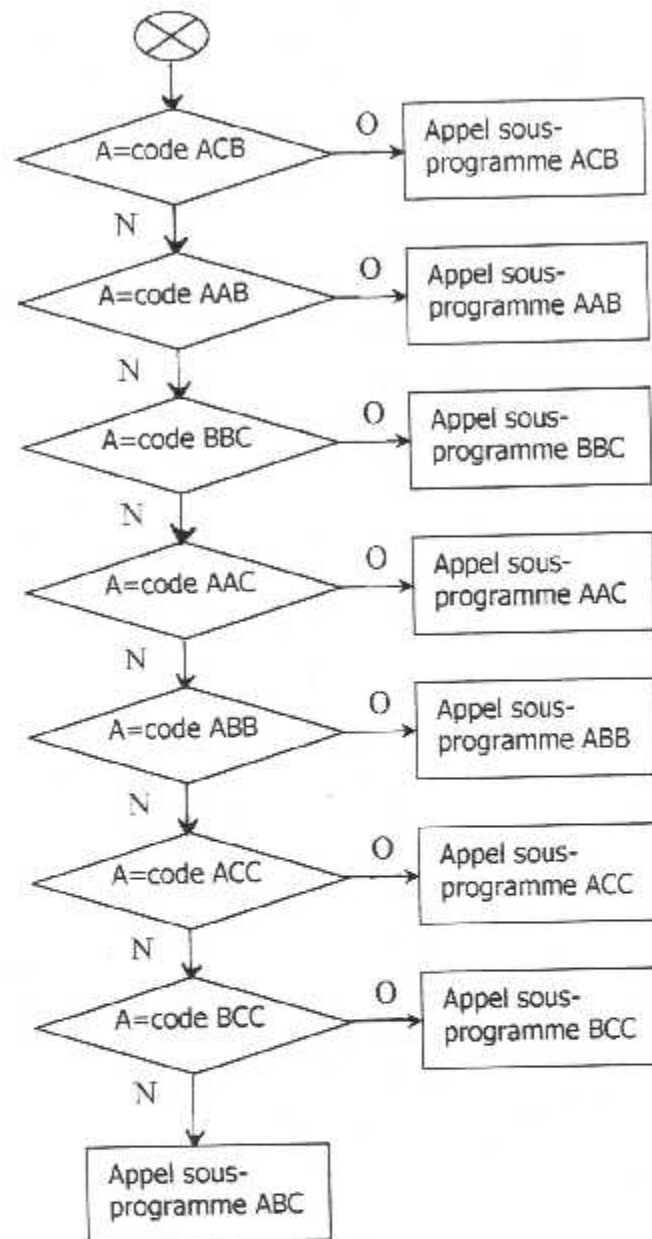


Fig B.5 Organigramme d'entrelacement des modes par récurrence

BIBLIOGRAPHIE

BIBLIOGRAPHIE

- * TECHNIQUE DE L'INGÉNIEUR –AERONAUTIQUE-(1989).
- * F.CHATELET "Le radar secondaire et ses évolutions".
- * PRINCIPES RADAR PRIMAIRE 1993.
- * G.DELOM "LE RADAR SECONDAIRE". polycope ENAC 1979.
- * ROGER L.TOKHEIM "Microprocesseur 2". 1984.
- * MICHEL AUMIAUX "Les microprocesseurs 8 bits",1985.
- * TTL DATA BOOK IN DESIGN ENGINEERING.
- * SCIENCE ET VIE NUMERO 956 MAI 1997.
- * Carte CPU didactique à base de microprocesseur 8085 (session juin 1991). Projet de fin d'études