

الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

46/86

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

2 ex

ECOLE NATIONALE POLYTECHNIQUE

المدرسة الوطنية للتكنولوجيا
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

DEPARTEMENT : ELECTRONIQUE

PROJET DE FIN D'ETUDES

(En vue de l'obtention du Diplôme d'Ingénieur d'Etat)

SUJET

**Test des systèmes L.S.I par
application du concept de la
région de travail propre
(SCHMOO-PLOT)**

Proposé par :

Mme B. KAMINSKA

Etudié par :

A. BENALLEGUE

M. BELKADI

Dirigé par :

Mme B. KAMINSKA

PROMOTION : JUIN 1986

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

الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

«*»

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

«*»

الجمهورية الجزائرية الديمقراطية الشعبية
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique
ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT : ELECTRONIQUE

PROJET DE FIN D'ETUDES

(En vue de l'obtention du Diplôme d'Ingénieur d'Etat)

SUJET

**Test des systèmes L.S.I par
application du concept de la
région de travail propre
(SCHMOO-PLOT)**

Proposé par :

Mme B. KAMINSKA

Etudié par :

A. BENALLEGUE

M. BELKADI

Dirigé par :

Mme B. KAMINSKA

PROMOTION : JUIN 1986

À la mémoire de mon père,
À ma mère et ma grand-mère,
À mes frères et soeurs,
À ma belle soeur et mes beaux frères,
À mes nièces et neveux,
À Noza et sa famille,
À tous mes amis et camarades.

M. Belkady

À mes parents qui ont fait de moi
ce que je suis.
À mes frères et soeurs,
À mes belles soeurs,
À mes neveux et ma nièce,
À tous mes amis et camarades.

A. BENALLEGUE

-o- REMERCIEMENTS -o-

A l'occasion du couronnement de notre travail, nous formulons l'expression de nos remerciements les plus sincères à Mme B. KAMINSKA qui a bien voulu assurer la direction de ce travail durant lequel elle n'a pas cessé de nous apporter le concours de sa compétence qui nous a permis d'élargir l'horizon de nos connaissances et de voir l'aboutissement de notre travail.

Nous voudrions que nos parents, nos amis, nos professeurs trouvent, dans ce modeste ouvrage, l'expression de notre gratitude et un grand hommage à toutes leurs peines, à leur aide moral, à leur patience sans limites, et aux précieux conseils dont notre éducation a été si bien nourrie.

Que tous ceux qui ont contribué, de près ou de loin, à la concrétisation de ce travail trouvent l'expression de nos plus vifs remerciements.

Nous tenons à remercier les membres du Jury, qui, par leur présence et leurs jugements, vont honorer ce modeste travail.

* INTRODUCTION

* PARTIE THEORIQUE

I- Influence des paramètres spécifiques sur le
fonctionnement d'un système.

II- Défauts dans les C.I. numérique.

A- Classification des défauts.

B- Modélisation des défauts.

C- Défauts spécifiques aux mémoires.

D- Défauts spécifiques aux microprocesseurs.

III- Test des circuits intégrés numériques

1- Tests paramétriques.

2- Tests fonctionnels.

3- Tests de caractérisation.

3-1- Concept de région de travail propre.

3-2- Préparation du test de caractérisation.

3-3- Intérêt des tests de caractérisation.

IV- Test des mémoires et microprocesseurs.

1- Test des mémoires.

2- Test des microprocesseurs.

2-1- Méthodes de test.

2-2- Techniques de test.

2-3- Etablissement du test complet d'un microprocesseur.

V- Application du test de caractérisation.

* PARTIE EXPERIMENTALE

I- Présentation des circuits MC 6802 et MC 6821

I-1- Le microprocesseur MC 6802

I-2- L'interface MC 6821 (P.I.A).

II- Test du circuit microprocesseur.

II-1- Principe.

II-2- Description du schéma général du circuit de test.

II-3- Execution du test.

CONCLUSION

ANNEXE

Le but de notre projet est l'étude de tests des systèmes à grande échelle d'intégration (L.S.I) et application du concept de la région de travail propre (schmoo-plot).

L'accroissement de la complexité des systèmes à grande échelle d'intégration a engendré une augmentation des difficultés dans le test de ces systèmes. Ce qui a eu pour effet de restreindre le champ d'action des méthodes de test traditionnelles développées pour les systèmes à faible et moyenne échelle d'intégration (S.S.I et M.S.I), et qui étaient essentiellement des méthodes algorithmiques. Pour pallier cette insuffisance, on a recours à des méthodes heuristiques basées sur l'intuition et l'expérience, en matière de test, de la personne qui veut tester, car outre les modèles de défauts connus aux systèmes S.S.I et M.S.I, il y a d'autres défauts qui ont pris naissance avec l'augmentation du taux d'intégration, mais qui restent encore indéfinis. Ils sont attribués à la sensibilité des systèmes L.S.I.

Dans notre étude, c'est le test de caractérisation qui est l'application du concept de la région de travail propre qui a été développé, vu ses nombreux avantages. Lesquels seront explicités ultérieurement.

L'élément qui a été soumis au test est le microprocesseur. Celui-ci est placé dans son environnement naturel (ou simulé) afin d'obtenir des résultats plus concrets.

Notre étude constitue une suite de celles effectuées dans les projets étudiés dans les semestres précédents.

La première partie consiste en une étude théorique dans laquelle l'origine des défauts a été mise en évidence. A ceci fait suite une classification des tests pouvant être appliqués pour juger du bon fonctionnement du système à tester.

La deuxième partie, quant à celle, est une étude expérimentale, dans laquelle on a utilisé un matériel concret pour nos expériences.

المدرسة الوطنية المتعددة التقنيات
المكتبة — BIBLIOTHEQUE
Ecole Nationale Polytechnique

PARTIE

THEORIQUE

I. - Influence des paramètres spécifiques sur le fonctionnement d'un système.

D'une manière générale, lorsqu'un circuit est mis en vente, il est toujours accompagné d'une notice technique. Laquelle spécifie les valeurs nominales des paramètres de ce circuit ainsi que leurs intervalles de variations admissibles.

Exemple :

La notice technique du microprocesseur MC 6802 de MOTOROLA.

$$(V_{cc} = (5 + 5\%) V, V_{ss} = 0, T_A = 0 \text{ à } 70^\circ\text{C})$$

Caractéristique	Symbole	Min	Typ	Max	Unité
tension max d'entrée Logic Reset	V_{IH}	$V_{ss} + 2.0$ $V_{ss} + 4.0$	-	V_{cc} V_{cc}	V_{dc}
tension min d'entrée Logic Reset	V_{IL}	$V_{ss} - 0.3$ $V_{ss} - 0.3$	-	$V_{ss} + 0.8$ $V_{ss} + 2.3$	V_{dc}
Courant de fuite d'entrée	I_{IN}	-	1.0	2.5	μA_{dc}
tension max de sortie $I_{charge} = -205 \mu A_{dc}, V_{ccmin}$ $I_{ch} = -145 \mu A_{dc}, V_{ccmin}$	V_{OH}	$V_{ss} + 2.4$	-	-	V_{dc}
tension min de sortie	V_{OL}	-	-	$V_{ss} + 0.4$	V_{dc}
Puissance dissipée	P_D	-	0.600	1.2	W
Capacité ($V_{in} = 0, T_A = 25^\circ\text{C}, f = 1.0 \text{ MHz}$) Logic $A_0 - A_{15}, R/\bar{W}, VMA$	C_{IN}		10	12.5	PF
	C_{OUT}			12	PF
fréquence d'opération Horloge freq. cristale	f	0.1	-	1.0	MHz
	f_{xtal}	1.0	-	4.0	
temps d'horloge : temps de cycle Largeur d'impulsion temps de montée.	t_{cyc}	1.0	-	10	μs
	P_{WOH} P_{WOL}	0.450	-	9.500	μs
	t_ϕ	-	-	25	ns

Si le circuit présenté travaille correctement pour les valeurs nominales des paramètres spécifiques, ainsi que pour les valeurs comprises dans l'intervalle de tolérance, Ceci montre que le circuit a subi des tests par le constructeur avant de sortir sur le marché.

En réalité le circuit peut ne pas fonctionner correctement pour certaines valeurs des paramètres spécifiques comprises dans l'intervalle de tolérance. Ceci est dû au fait que le constructeur exécute des tests non exhaustifs à cause du coût très élevé de ces tests.

Le grand nombre des paramètres spécifiques aux circuit nécessite un très grand nombre de tests pour la vérification du circuit. Et leur influence relative, sur le fonctionnement de l'unité soumise au test, nous conduit, dans la pratique, à faire un choix judicieux des paramètres à utiliser dans le test.

Parmi les paramètres spécifiques au circuit à tester, on distingue deux groupes :

I-1- paramètres statiques : tels que la tension d'alimentation, la fréquence, le courant de fuite, les niveaux logiques "0" et "1", etc...

I-2- paramètres dynamiques : tels que le cycle d'horloge, le temps d'accès, le temps de décalage des phases d'horloge (cas du microprocesseur), le temps de montée et de descente, etc...

Exemples de paramètres et leur influence.

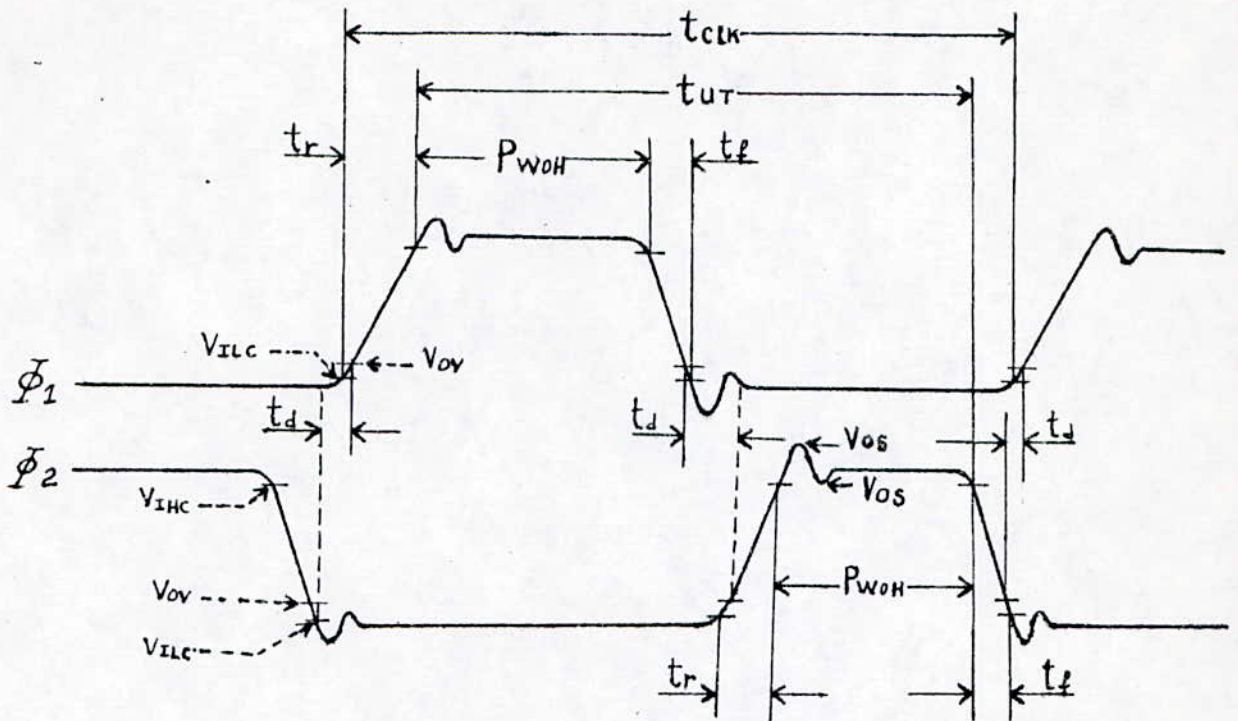
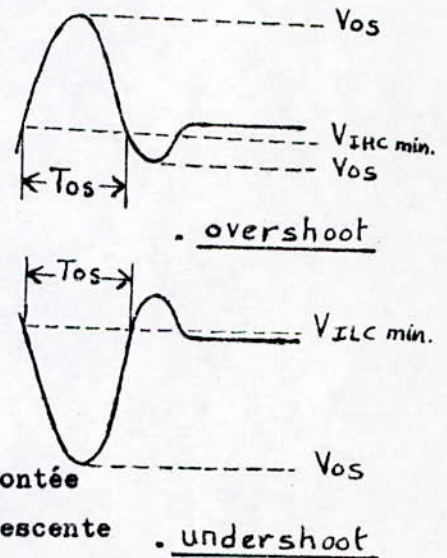


figure: 1-1

- t_r : temps de montée
- t_f : temps de descente
- P_{wOH} : durée d'impression
- t_d : temps de décalage
- t_{ur} : clock high time
- t_{fl} : temps du cycle d'horloge
- T_{os} : durée du Pic
- V_{IHC} : tension correspondant au temps de montée
- V_{IILC} : tension correspondant au temps de descente
- V_{os} : tension des pics.



La figure I-1 représente les signaux d'horloge d'entrée ϕ_1 et ϕ_2 du microprocesseur MC 6800 de MOTOROLA.

On remarque que ces signaux sont caractérisés par plusieurs paramètres susceptibles de varier et produire des perturbations dans le fonctionnement du microprocesseur.

Avant d'étudier l'influence de certains paramètres, il convient de définir les deux signaux d'horloge d'entrée ϕ_1 et ϕ_2 du microprocesseur.

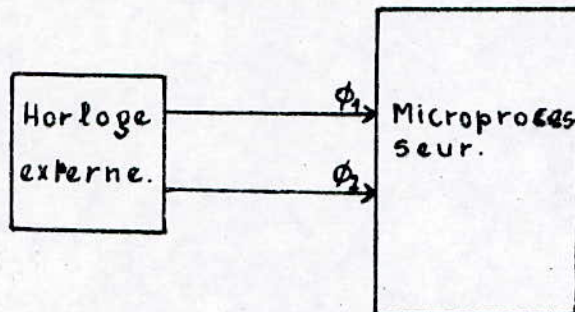


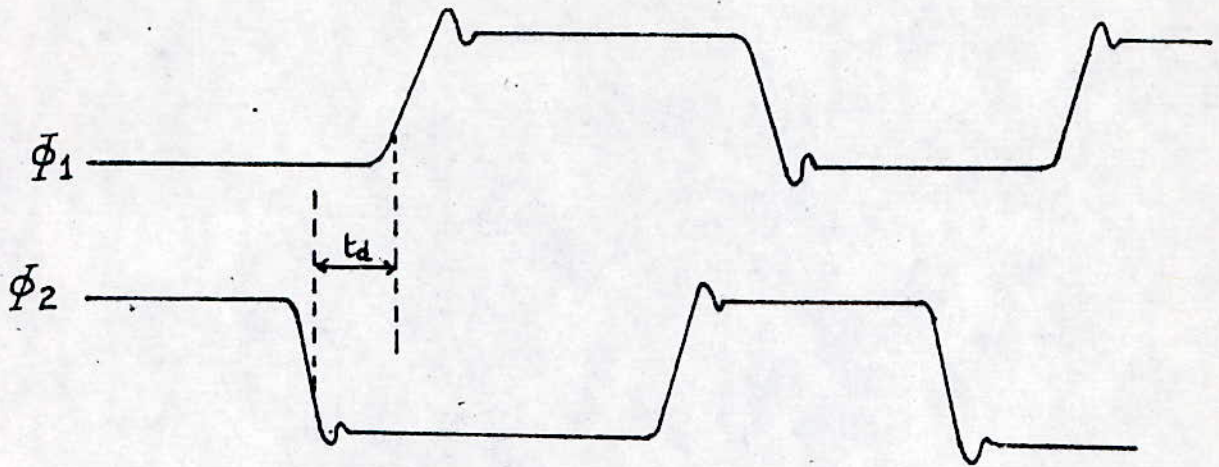
figure. I-2

Comme le montre la figure I-2 une horloge externe fournit deux signaux ϕ_1 et ϕ_2 qui sont en opposition de phase et qui sont les deux signaux d'horloge d'entrée du microprocesseur. On rappelle que, pour le cas du microprocesseur MC 6800 de MOTOROLA, pendant la phase d'horloge ϕ_1 , le microprocesseur effectue des traitements et pendant la phase ϕ_2 fait des échanges avec les circuits d'interface d'entrée / sortie et les mémoires.

a) Influence du temps de décalage t_d

D'après le chronogramme de la figure I-4 on remarque que si l'on décale le signal ϕ_2 vers la gauche (voir fig. I-3), le microprocesseur reste inactif pendant un certain temps. Et à la première impulsion de ϕ_1 , le microprocesseur exécute l'opération qui lui est demandée, mais avant qu'elle ne soit terminée, il commence à communiquer avec l'extérieur. ceci produit un embrouillement dans le fonctionnement du microprocesseur,

car on a une mauvaise synchronisation.



- figure: 1-3

b) Influence du temps de montée t_r .

Si le temps de montée t_r venait à diminuer, la transition positive serait plus rapide. Ceci peut avoir un effet néfaste sur le circuit et son fonctionnement.

En d'autres termes, si t_r diminue, l'amplitude V_{os} du shift augmente, ce qui peut endommager le circuit. A la limite, on aurait un fonctionnement en oscillation tant que l'état stable n'est pas atteint, car le temps T_{os} deviendrait plus grand que la valeur spécifique. Le même raisonnement peut être fait pour le temps de descente t_f .

II- Les défauts dans les circuits intégrés numériques :

Un défaut peut être comme étant un état du système qui conduit à l'apparition d'une panne ou d'une erreur dans la fonction demandée au circuit.

Comme il y a une variété de défauts qui apparaissent lors des tests des circuits intégrés, l'idée d'une classification a pris naissance pour faciliter l'établissement des tests.

II-1- Classification des défauts :

Les défauts pouvant affecter un circuit intégré peuvent être classés comme suit :

1- Défauts technologiques :

Ils sont dus au type de technologie utilisé dans le processus de fabrication. Par ailleurs, chaque technologie utilise des masques géométriques spécifiques qui doivent être bien placés sur la plaquette de silicium et sur lesquels on doit avoir des dessins exactes des différentes couches, sous peine de provoquer, par exemple, un rapprochement excessif qui pourrait donner un court-circuit.

2- Défauts électriques :

Ce sont des défauts qui apparaissent lors du changement des spécifications du circuit dû, soit aux erreurs de conception, soit au vieillissement des composants ou encore aux erreurs de fabrication, par exemple, un rapprochement excessif de deux pattes du circuit pourrait provoquer un court-circuit.

3- Défauts logiques :

Ce type de défauts cause des erreurs dans la fonction logique exigée du circuit.

Ces défauts sont une conséquence des défauts de conception ou des défauts technologiques ou électriques, ou encore des défauts de fabrication.

Exemple : Un court-circuit entre les entrées d'une porte OU-EX provoque une erreur dans la fonction réalisée par cette porte.

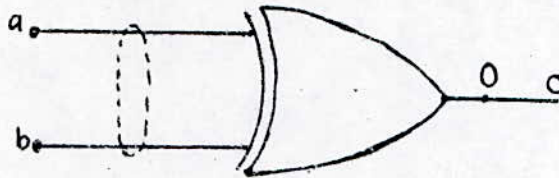


Table de vérité théorique

a	b	c
0	0	0
0	1	1
1	0	1
1	1	0

Table de vérité réelle

a	b	c
0	0	0
0	1	0
1	0	0
1	1	0

Cette classification ne nous permet pas de définir un test spécifique pour chaque type de défaut, car comme on l'a vu, il y a des défauts ayant la même cause ou le même effet, mais appartenant à des classes différentes. Donc l'idée d'une modélisation, de ces défauts, s'impose.

II-2- Modélisation des défauts :

Une modélisation est une représentation d'un ensemble de défauts ayant la même cause ou le même effet par un modèle. Ceci nous permet de dégager un test spécifique pour chaque modèle de défauts. Ainsi, un défaut peut être déterminé à partir du modèle relatif au test appliqué au circuit.

On distingue deux modèles de défauts représentant tous les défauts rencontrés dans le test des systèmes S.S.I. et M.S.I.

1- Défauts de collage à une valeur logique "0" ou "1".

Ce modèle regroupe tous les défauts provoquant un défaut logique, comme les défauts électriques ou technologiques, Ces derniers ne peuvent se matérialiser que par un collage à une valeur logique "0" ou "1".

2- Défauts dynamiques :

Ce sont tous les défauts relatifs au temps ou aux délais des circuits qui sont endehors des performances spécifiques.

exemple : Défaut de synchronisation.

Le classement par modèle reste valable pour les systèmes L.S.I, mais insuffisant, du fait qu'il existe d'autres défauts spécifiques aux systèmes L.S.I, qui constituent un modèle complémentaire. Ces défauts sont généralement dus à la sensibilité de ces systèmes on rencontre des défauts de sensibilité aux données dans le test des mémoires et des défauts de sensibilité aux données et aux commandes pour le cas des microprocesseurs.

II-3- Défauts spécifiques aux mémoires :

II-3-1- Description d'une mémoire dynamique RAM de 16 Kmots de 8 bits.

La figure II-1 représente une mémoire dynamique RAM de 16 Kmots de 8 bits qui comprend :

- deux Buffers omnidirectionnels (un d'entrée et l'autre de sortie) de données.

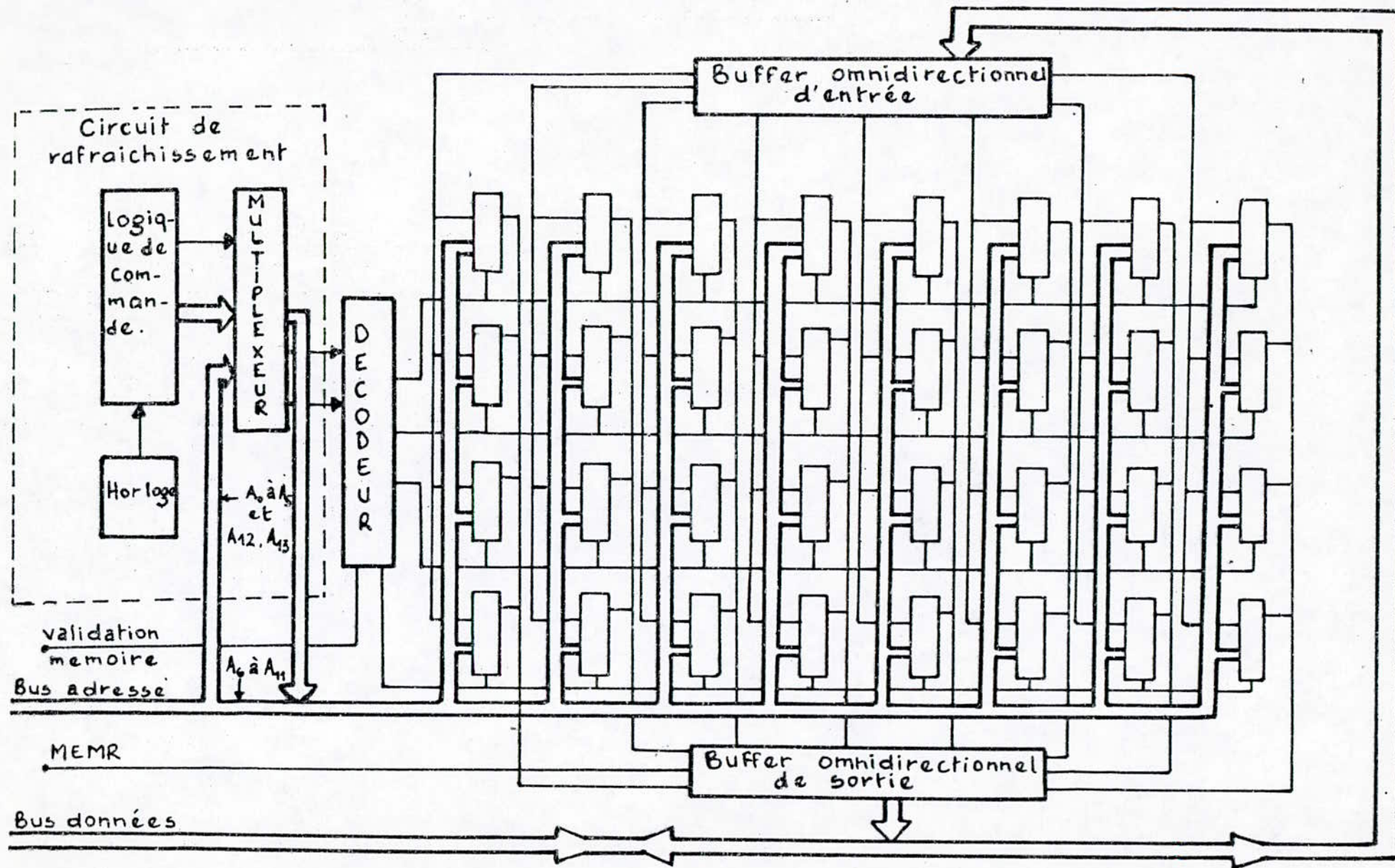


Figure:II-1 - Mémoire dynamique RAM de 16K mots de 8 bits

- Un décodeur d'adresse.
- Une commande lecture / écriture.
- Un chip select qui est entrée permettant la sélection de la mémoire.
- Le bloc mémoire proprement dit.
- Un circuit de rafraichissement comprenant la logique de commande, un multiplexeur permettant la répartition de la tâche sur les différents points mémoire, une horloge permettant la synchronisation.

II-3-2- Défauts spécifiques :

Les défauts qui apparaissent le plus souvent dans le test des mémoires sont présentés comme suit :

- Court-circuit et circuit ouvert : Ce sont des défauts se traduisant par une liaison parasite ou un mauvais contact.
- Claquage de jonction : diode ou tout autre jonction à semi-conducteur claquée.
- Fuite à l'entrée ou à la sortie : Le courant de fuite qui augmente par rapport à la valeur spécifique.
- Décodeur défaillant : Une faille du décodeur entraîne un mauvais adressage.
- Ecriture multiple : Une donnée adressée à une seule position mémoire, se trouve mémorisée, ainsi, dans d'autres positions mémoires.
- Sensibilité aux vecteurs de test : Le dispositif ne répond pas convenablement aux excitations de test d'écriture, de lecture ou d'adressage.
- Rafraichissement défaillant : Durant le temps de rafraichissement spécifique il y a perte d'information dans les RAM dynamiques.
- Largeur du temps d'accès : L'acquisition des données de la mémoire ne s'effectue pas dans les délais prévus.
- Recouvrement de l'écriture : Une lecture qui suit immédiatement une écriture provoque un retard dans l'accès.
- Sensibilité de l'amplificateur : Quand une même donnée accède pour un grand nombre de fois à une position mémoire, puis soudain elle change de valeur, l'amplificateur tend à rester dans le même état, du fait qu'il devient insensible au changement de la donnée.

II-4- Défauts spécifiques aux microprocesseurs :

II-4-1- Description du schéma interne d'un microprocesseur :

La figure II-2 représente le schéma interne d'un microprocesseur qui comprend :

- L'U.A.L : Ce module effectue les opérations logiques et arithmétiques et des tests.

Généralement l'U.A.L. fait appel aux registres de travail, tel que l'accumulateur, pour l'exécution de ces opérations.

-L'unité de commande : dont le rôle est de permettre à tous les éléments constituant le microprocesseur de travailler ensemble et dans le bon. Elle tire les instructions du registre d'instructions, détermine ce qu'il y a lieu de faire avec les données et crée les commandes nécessaires à l'exécution de la tâche considérée.

- Les registres : qui servent au stockage temporaire de données. Certains d'entre eux ont une fonction particulière tel que le compteur programme. D'autres n'ont pas de spécialisation et sont à la disposition du programmeur pour l'exécution de toute opération. Il existe six registres fondamentaux que comportent pratiquement tous les microprocesseurs. Ils peuvent être cités comme suit :

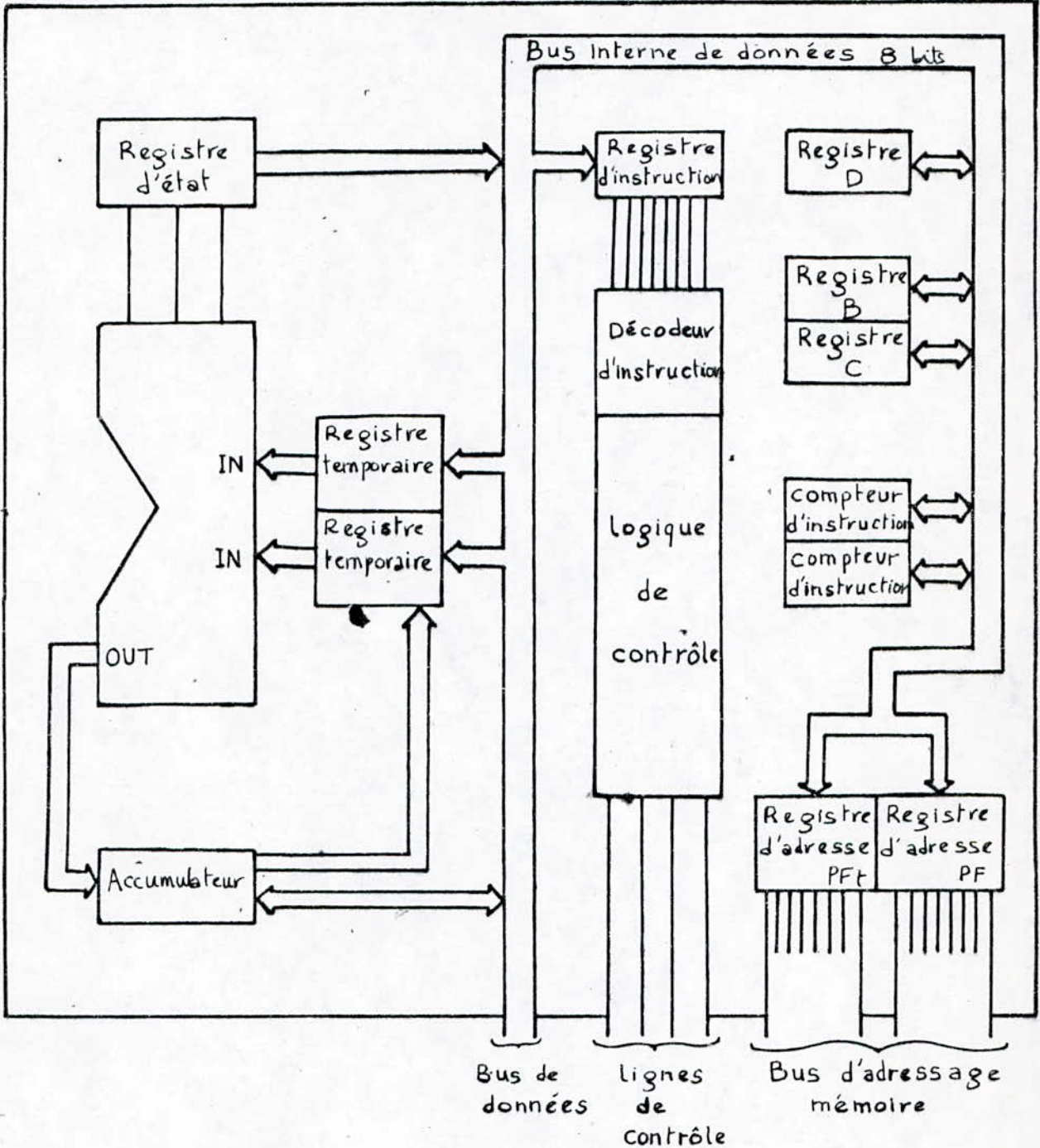
L'accumulateur, le compteur d'instructions, le registre d'adresse mémoire (P.C.) et le registre temporaire de données.

- Le bus interne de données : Il constitue le système de transmission de données entre les différents éléments logiques du microprocesseur.

II-4-2- Défauts spécifiques :

Les microprocesseurs étant des circuits intégrés très complexes. Les défauts des mémoires, tels que les court - circuits, les circuits ouverts et les défaillances des décodeurs existent certainement dans ce type de circuits. Toutefois, ces circuits, vu leur nouveauté, présentent d'autres défauts qui restent encore inconnus.

figure.II.2 -Organisation Interne d'un microprocesseur.



Parmi les défauts qu'on peut rencontrer lors du test d'un microprocesseur, on note :

- Défaillance des registres : le microprocesseur n'exécute pas convenablement le programme introduit dans son espace d'adressage.

Exemple : Si le compteur programme (P.C) est défaillant, des sauts imprévus s'effectuent dans le déroulement du programme.

- Défaillance du séquenceur : si le séquenceur de l'unité de commande est le siège d'une anomalie, des microinstructions imprévues seront générées.

- Défauts de synchronisation : un décalage entre les deux phases d'horloge provoquerait un embrouillement dans l'exécution du programme.

Exemple : L'arrivée du signal de validation d'écriture après la donnée dans une opération de stockage serait la cause de beaucoup de problème.

- Sensibilité du circuit : l'expérience a démontré l'existence de séquences d'instructions et de données pour lesquelles le microprocesseur est insensible.

III- Tests des circuits intégrés numériques :

Les tests ont pris une part importante dans la fabrication des circuits intégrés. En effet, le constructeur, connaissant les résultats des tests, peut améliorer la conception de son produit, le processus de fabrication et trier les unités fabriquées. Ils sont également importants pour l'utilisateur qui peut vérifier le bon fonctionnement du circuit dans les conditions de son utilisation.

On distingue trois catégories de tests des circuits intégrés numériques.

III-1- Tests paramétriques :

Ce sont des tests préliminaires qui permettent :

a- L'évaluation de certains processus technologiques.

Exemple :

Des valeurs incorrectes des tensions de sortie montrent l'irrégularité du processus d'implantation d'ions.

b- La vérification de certains paramètres techniques donnés par le constructeur.

Exemple :

Tension, Courant de fuite, etc...

III-2- Tests fonctionnels :

III-2-1- Tests fonctionnels simples (faible fréquence)

Ce sont des tests préliminaires qui ont pour rôle de détecter les défauts grossiers, en vérifiant l'existence et le fonctionnement des lignes de communication internes du circuit.

Exemple : Vérification du fonctionnement du circuit pour les valeurs nominales, telles que, la tension d'alimentation, la fréquence, etc...

III-2-2- Tests fonctionnels plus complets, avec mesures dynamiques calées sur les spécifications et pour plusieurs paramètres du circuit.

La préparation de la séquence de ces tests repose, comme la précédente sur la connaissance de l'architecture interne du circuit. En général, on test en premier lieu les lignes de communication entre modules, puis on teste les modules un par un en commençant par les modules accessibles directement de l'extérieur, et en poursuivant par les modules les plus internes.

L'accès à ces derniers se faisant par l'intermédiaire d'autres modules que l'on s'efforce de tester au préalable. Le test fonctionnel complet est donc découpé en plusieurs étapes, chacune concernant un module interne du circuit. Chaque étape est répétée pour chaque variation d'un paramètre. Ce dernier étant pris parmi ceux qui ont plus d'influence sur le circuit.

exemple : On fait le test du circuit, de la tension d'alimentation minimale à la tension maximale.

III-3- Tests de caractérisation

Tout test pouvant donner des résultats qui définissent une caractéristique d'un circuit peut être considéré comme étant un test de caractérisation.

Parmi les tests de caractérisation connus, on distingue :

- 1- Le test statistique.
- 2- Le test déterminant la carte d'une plaque à microprocesseurs.
- 3- Le test définissant la région de travail propre du circuit, etc...

III-3-1- Concept de la région de travail propre

Ce concept a été établi pour la première fois vers les années 60 et ceci pour le test des mémoires magnétiques qui permet de donner la région où la mémoire travail correctement pour une combinaison de deux paramètres du circuits. Et avec l'apparition des microprocesseurs, ce concept a été développé pour l'adapter au test des microprocesseurs.

Dans ce concept on considère que les paramètres décrivant un circuit électronique intégré sont divisés en deux groupes.

- groupe des paramètres X : ce sont les paramètres caractérisant les éléments du circuit, telsque, la tension d'alimentation, la fréquence de l'oscillateur, etc... et les paramètres représentant les entrées, telsque les données, les signaux de contrôle IN, etc...

- groupe des paramètres Y : ce sont les paramètres représentant les sorties du circuit, tels que, les signaux de contrôle OUT.

exemple : Signal lecture / Ecriture pour un microprocesseur.

Les relations entre les paramètres X et Y peuvent être décrites par les fonctions ϕ_j telles que :

$$Y_j = \phi_j(x_1, x_2, \dots, x_N) \quad ; \quad j = 1, 2, \dots, M \quad (1)$$

Les valeurs x_i , des paramètres X, sont prises comme étant des composantes d'un vecteur $\bar{x} = [x_1, x_2, \dots, x_N]$, ainsi une valeur spécifique du vecteur \bar{x} peut être prise comme étant un point dans l'espace à N dimensions.

En donnant la gamme de tolérance $2 \times \delta x_i$ pour chaque paramètre, on peut utiliser une représentation géométrique pour introduire le concept d'hypercube de tolérance, pour N paramètres X,

$$D_T = 2 \delta x_1 \times 2 \delta x_2 \times \dots \times 2 \delta x_N$$

L'hypercube de tolérance est un espace à N dimensions constitué par les valeurs nominales des paramètres, ainsi que leur gamme de variations admissibles. Dans le cas d'une combinaison de deux paramètres, on parlera de rectangle de tolérance.

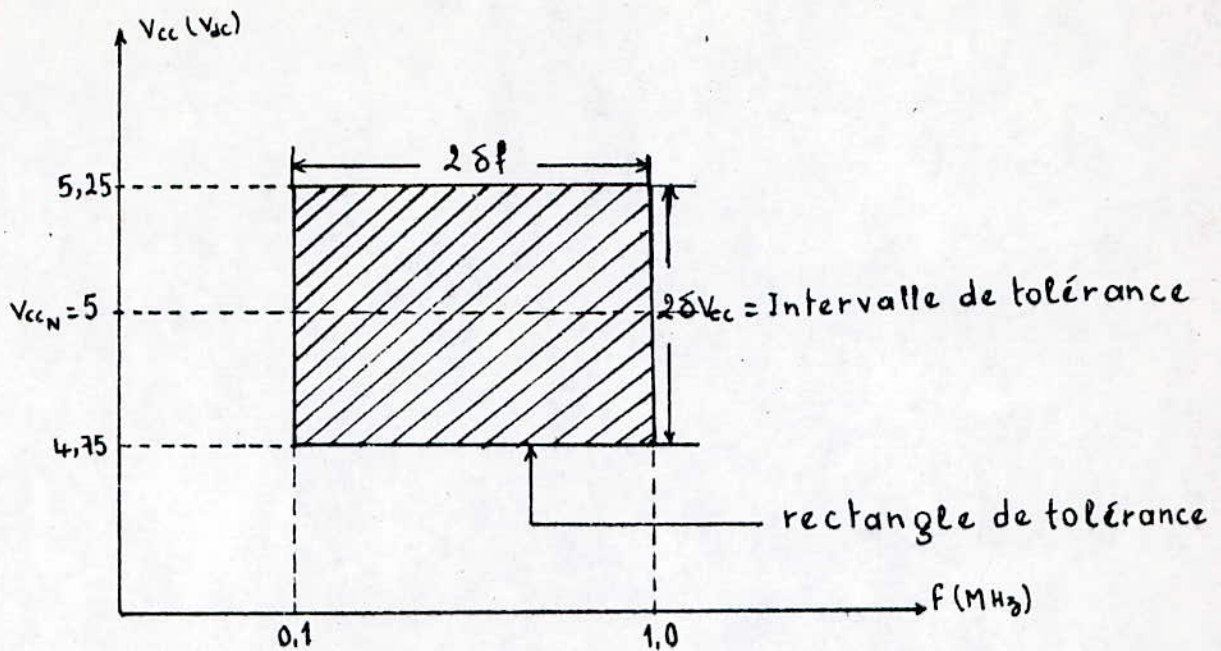
exemple : Dans le cas du microprocesseur MC 6802 de MOTOROLA.

$$V_{CC} = (5 \pm 5\%) \text{ V}$$

$$V_{CCN} = 5 \text{ V}$$

$$f_{\min} = 0,1 \text{ MHz}$$

$$f_{\max} = 1,0 \text{ MHz}$$



Les valeurs des paramètres Y_j doivent vérifier les inégalités :

$$Y_{jmin} < Y_j = \phi_j(x_1, x_2, \dots, x_N) < Y_{jmax} \quad (2)$$

dans lesquelles Y_{jmin} et Y_{jmax} délimitent les intervalles de variations admissibles des paramètres de sortie Y_j .

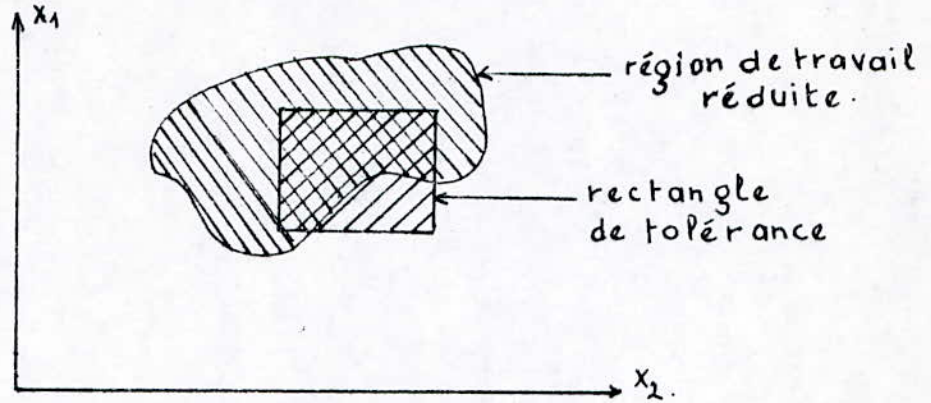
Ces valeurs limites des paramètres de sortie Y_j et les intervalles de tolérance des paramètres d'entrée X_i sont spécifiées dans la notice technique fournie par le constructeur.

La région de travail propre est la région de l'espace à N dimensions où le circuit travaille correctement.

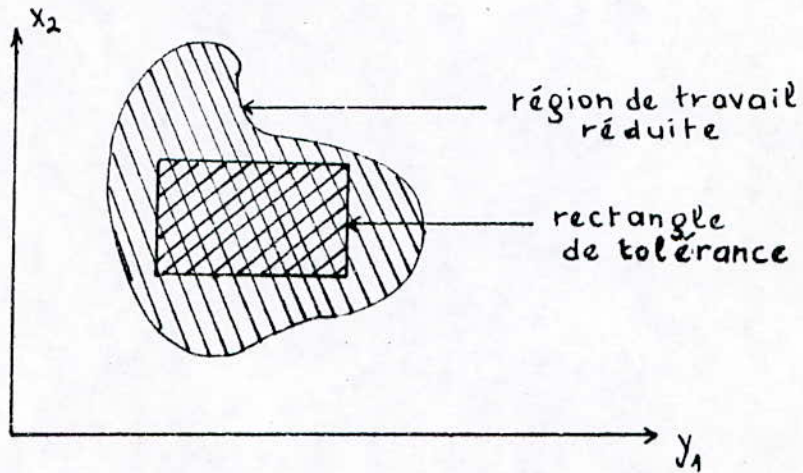
La représentation graphique directe de la région de travail propre à plus de deux dimensions est difficile, sinon impossible. Cependant, la région de travail propre peut être examinée en considérant un ou deux paramètres variables et tous les autres fixés.

Ce qui définit des sections appelées régions de travail réduites.
Ces dernières peuvent être définies de plusieurs façons :

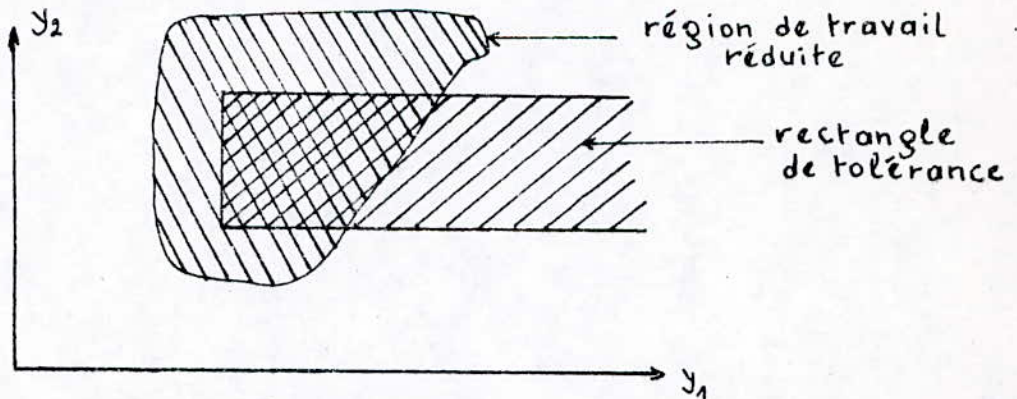
- avec des paramètres X seulement.



- avec des paramètres X et Y :



- avec des paramètres Y seulement.



Pour pouvoir parler de région de travail réelle d'un système, il est évident qu'il faut prendre toutes les combinaisons possibles des paramètres X ou bien X et Y ou encore Y ,

de façon que si l'on choisit n paramètres, on obtient :

$$P = C_n^2 = \frac{n!}{2!(n-2)!} \quad \text{combinaisons possibles, donc}$$

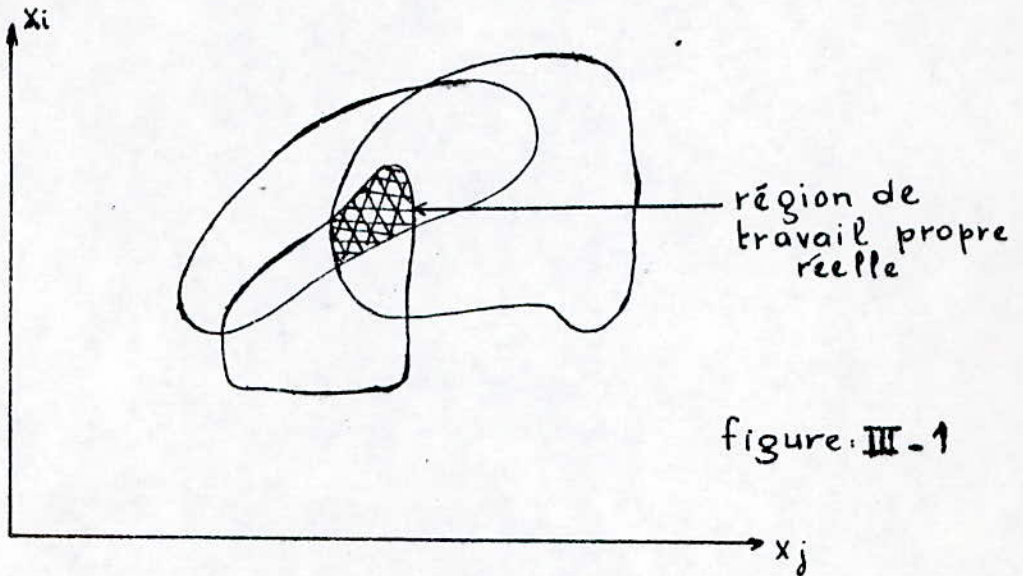
régions de travail réduites. Et la région de travail propre réelle ne peut être que l'intersection de ces régions de travail réduites.

exemple : si on prend 3 paramètres X (X_1, X_2, X_3)

le nombre de régions de travail réduites (schmoo-plot) que l'on peut obtenir est :

$$P = C_3^2 = 3$$

Comme le montre ~~le montre~~ la fig. III-1, la région qui caractérise le bon fonctionnement du circuit est l'intersection des trois surfaces.



3-2- Préparation du test de caractérisation envisagé

Ce test est effectué, à partir de séquences du type de celles utilisées pour les tests fonctionnels complets, mais en faisant varier les paramètres d'utilisation X , par groupe de deux de façon à déterminer les régions de travail réduites et par conséquent, la région de travail réelle qui est leur intersection.

La figure III-2: montre les instruments qui sont :

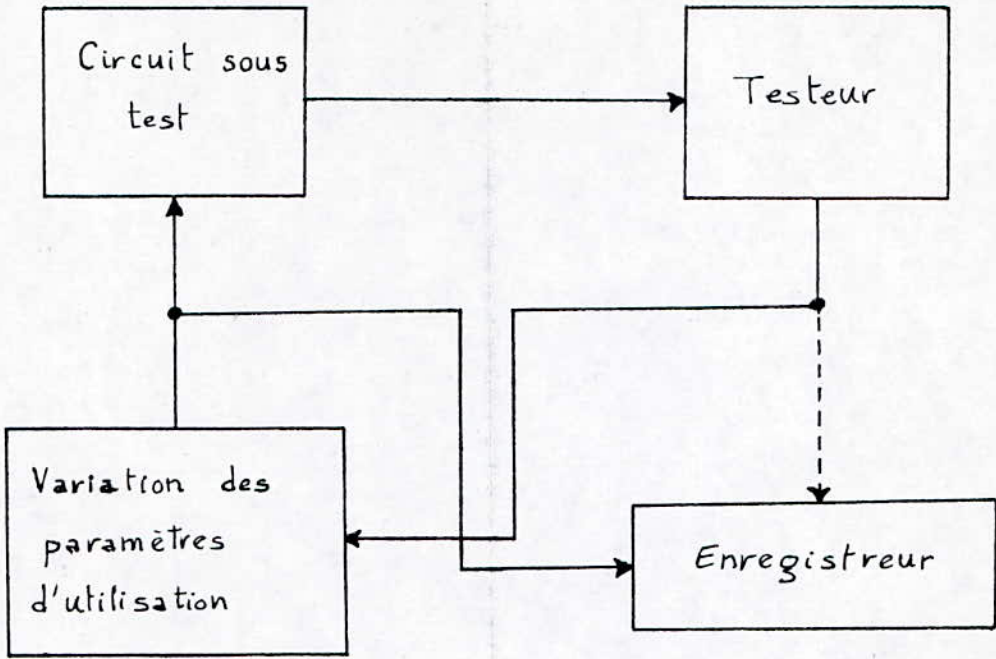


Figure: III-2

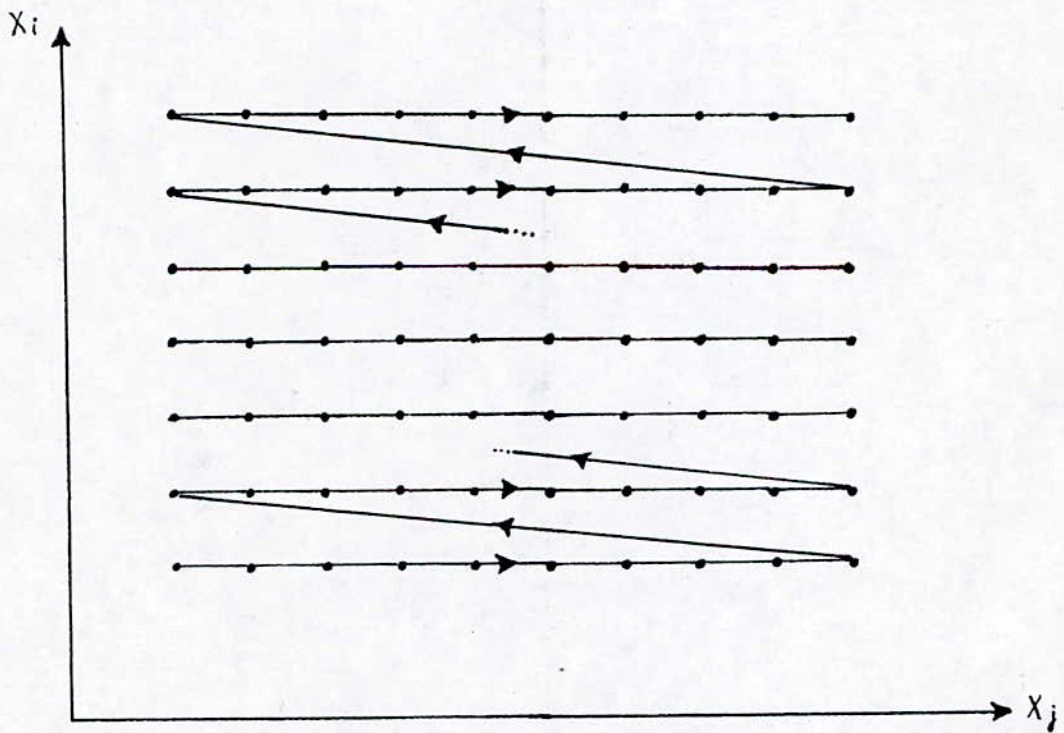


Figure: III-3

utilisés pour effectuer le test et déterminer la région de travail propre du circuit sous test.

L'instrumentation consiste en, un bloc qui fournit les variations des paramètres X , un testeur et un enregistreur.

Le bloc des variations des paramètres choisit les valeurs successives des paramètres X , d'après l'algorithme d'examen de la région de travail propre.

Pour un point choisi, le testeur vérifie l'opération du circuit sous test, après avoir contrôlé que toutes les fonctions soient exécutées correctement et examiner si les paramètres Y_j sont dans leurs intervalles admissibles spécifiés par la notice technique, ce bloc génère un signal si toutes les conditions requises pour considérer le circuit comme bon, sont remplies, sinon il génère un signal d'erreur (informant qu'il y a erreur). Les résultats obtenus sont enregistrés par le bloc enregistreur.

La figure III-3 illustre le procédé de recherche d'une région de travail propre (à deux dimensions) ou région de travail réduite.

3- 3- Intérêt du test de caractérisation :

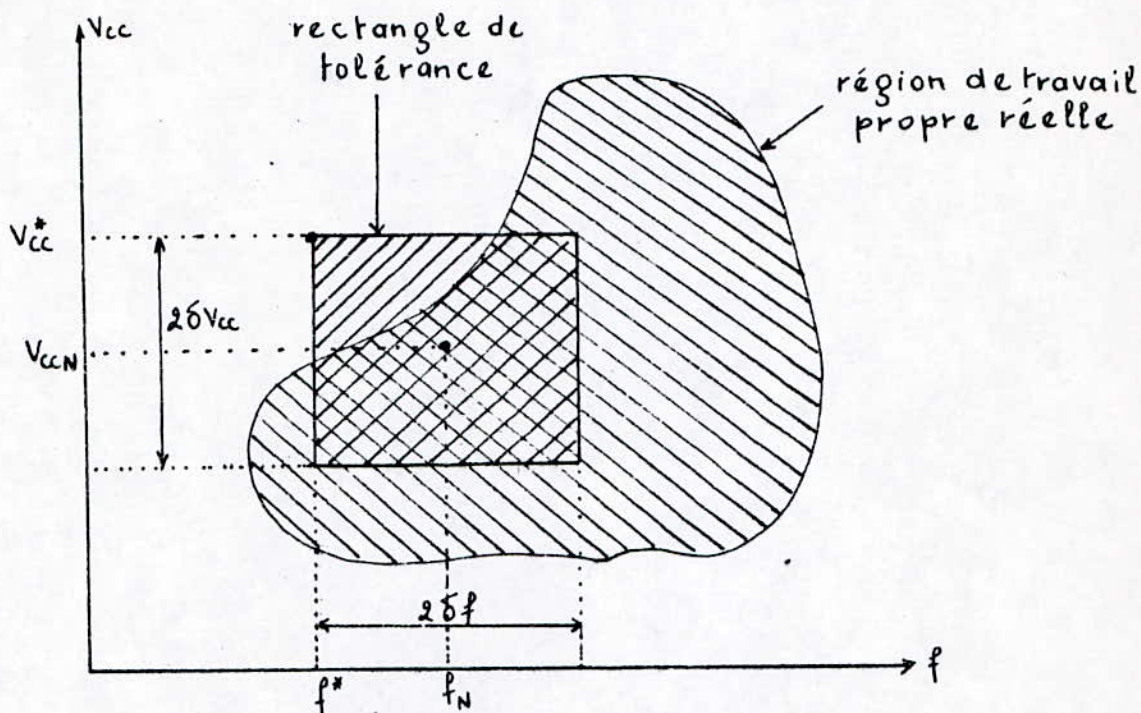
Les tracés effectués dans ce type de test sont d'un intérêt multiple, en effet ils permettent :

- D'analyser l'influence des paramètres extérieurs sur les performances du circuit. Les informations obtenues sont précieuses pour le fabricant, car elle lui permettent d'améliorer la conception de son produit et le processus de fabrication.

- De déceler les composants marginaux. Même si ceux-ci sont conformes aux spécifications, ce sont les premiers candidats aux défaillances futures.

- De déceler les séquences "sensibles" car le domaine de fonctionnement se trouve d'autant plus réduit que la séquence utilisée est plus sévère.

exemple de tracé



V_{ccN} : tension d'alimentation nominale

f_N : fréquence nominale

$2\delta V_{cc}$: intervalle de tolérance de la tension V_{cc}

$2\delta f$: intervalle de tolérance de la fréquence f

Ce tracé nous permet de dire que les intervalles de tolérance fournis par le constructeur ne correspondent pas vraiment à la réalité. Car on voit que dans le rectangle de tolérance, il existe des cas pour lesquels le circuit ne fonctionne pas correctement. Le point A $\begin{pmatrix} V_{cc}^* \\ f^* \end{pmatrix}$ est un point de pire cas optimum.

IV. Test des mémoires et microprocesseurs :

Avant d'entamer la présentation des méthodes de test, il convient de rappeler que les microprocesseurs et les mémoires sont des systèmes L.S.I. ayant des structures différentes.

En effet, les mémoires ont une structure symétrique, alors que celle des microprocesseurs est plus complexe. En outre, comme les mémoires sont des organes à lecture et écriture seulement, outre les défauts connus aux systèmes S.S.I. et M.S.I, elles ont un problème de sensibilité uniquement aux données, alors que dans le cas des microprocesseurs il y a la sensibilité aux commandes qui vient élargir l'éventail des défauts. Ces différences fondamentales nous permettent d'affirmer, à priori, que les tests des mémoires sont faciles relativement à ceux établis pour les microprocesseurs. Et par conséquent, ce sont les méthodes et techniques de test des microprocesseurs qui ont été approfondies dans cette étude.

IV- 1- Test des mémoires :

Comme les mémoires sont souvent appelées par les microprocesseurs, elles doivent être testées, afin de voir la validité de leur fonctionnement. Pour ce faire, on procède par trois méthodes principales.

1- Méthode de parité :

Dans cette méthode, un bit de parité, dépendant de l'information à stocker, est ajouté à cette information dans l'opération d'écriture. A la phase de lecture, on recalcule le bit de parité et on le compare avec le premier. Si la valeur du bit n'a pas changé, on peut conclure que l'information stockée n'a pas été modifiée et que le circuit de mémorisation fonctionne convenablement, sinon il est défaillant.

exemple :

Le mot à écrire est :

b7								bo
1	0	0	0	1	0	1	0	

Le mot lu est :

b7								bo
0	0	0	0	0	0	1	0	

Le bit sept (b7) est le bit de parité

S'il est à 1 : i - e le nombre de 1 dans le mot est pair

S'il est à 0 : i - e le nombre de 1 dans le mot est impair

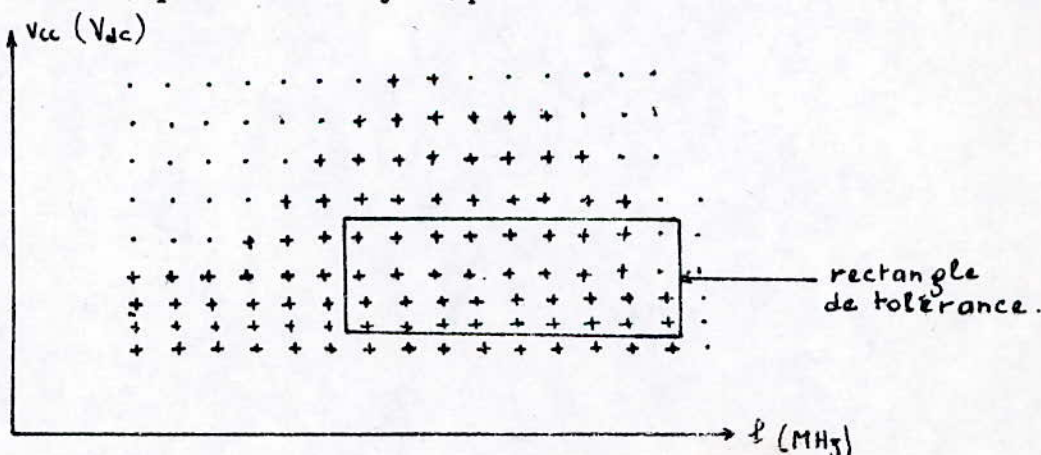
dans cet exemple on voit qu'il y a un défaut, car le mot lu possède un nombre impair de 1, par contre le mot écrit avait un nombre pair de 1.

2- Méthode des graphes :

C'est une application du test de caractérisation. Son principe consiste à tracer les graphes qui lient les paramètres d'utilisation du circuit et permet de définir le domaine de fonctionnement (ou région de travail propre) du circuit pour des variations quasi-continues de ces paramètres.

exemple :

Le tracé du domaine de fonctionnement pour une variation de la tension d'alimentation V_{cc} et la fréquence d'horloge (circuit de rafraichissement) pour une RAM dynamique.



- + bon fonctionnement
- . mauvais fonctionnement

3- Méthode des séquences de mots :

Cette méthode, appelée aussi génération algorithmique de suites de bits, consiste à écrire une suite de bits et de vérifier :

- Qu'elle a été écrite correctement.
- Que rien n'a été écrit ailleurs à cause d'un mal fonctionnement de la mémoire.

Les techniques utilisées dans cette méthode sont :

- Technique GALTCOL
- " DIAPAT
- " MARCHING
- " MASEST
- " WAKPAT
- " GALPAT

Ces techniques ont été étudiées dans des projets des semestres passés.

IV. -2-Tests des microprocesseurs :

Pour établir une méthode de test qui peut satisfaire tous les aspects du test envisagé, la connaissance de la structure exacte du microprocesseur est nécessaire. Or, la réalité de cette structure n'est qu'en partie dévoilée par le constructeur. Aussi, nous sommes conduit à établir des méthodes heuristiques qui approchent au mieux, par des améliorations, le dessin voulu.

IV.-2-1-Méthodes de test :

On distingue trois méthodes de test pour vérifier le bon fonctionnement d'un microprocesseur.

a) Test des modules un par un

Cette méthode consiste à tester les modules un par un en commençant par les modules les plus accessibles tels que les accumulateurs (cas du 6800) et les registres de travail, et en poursuivant par les modules les moins accessibles tels que l'U.A.L. le compteur programme etc...

exemple :

On teste l'accumulateur A (cas du MC 6800) en premier lieu pour pouvoir l'utiliser pour tester l'U.A.L. qui fait appel à l'accumulateur pour exécuter ses opérations.

b) Test de l'unité commande :

Cette méthode consiste à vérifier la fonction essentielle du microprocesseur, à savoir sa capacité de prendre des décisions en fonction des données qui lui sont fournies.

exemple :

Pour le cas du MC6800 on exécute l'opération de chargement de l'accumulateur A par la valeur FF, les données présentes en mémoire sont 86 (chargement de l'accumulateur) et FF (donnée). Le microprocesseur lit les données (86 et puis FF) et exécute une opération. Alors on vérifie si la fonction demandée a été réalisée, ainsi nous aurons testé l'unité de commande (i - e si le microprocesseur a decodé l'instruction convenablement).

c) Test mixte :

Cette méthode consiste à combiner les deux premières méthodes pour effectuer un test complet qui revient à vérifier quelques modules sensibles ainsi que les commandes les plus délicates.

exemple :

On réalise un test mixte en testant l'accumulateur A (cas du MC6800) à l'aide d'un code permettant de vérifier tous ses bits par l'exécution de la majorité des instructions.

IV-2-2- Techniques de test :

Suivant le matériel dont on dispose, on peut définir différentes techniques pour l'application de la méthode proposée.

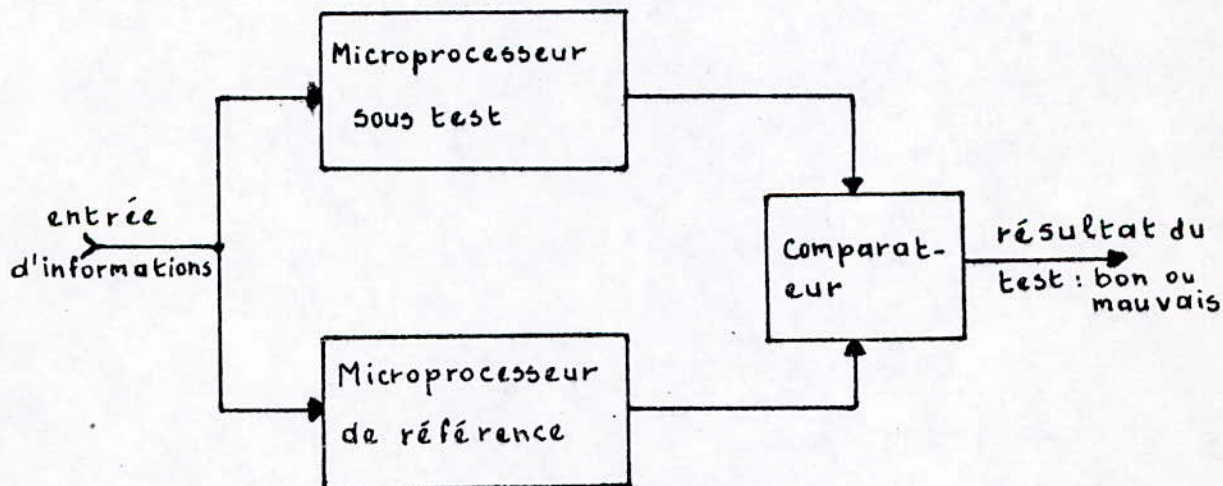
Les techniques existantes à l'heure actuelle sont :

- technique de test par comparaison
- " des patterns enregistrés
- " algorithmique en temps réel
- " LEAD (Learn Execut And Diagnose)
- " d'autodiagnostic (Auto-Test).

Parmi les techniques précitées, c'est la technique d'auto-test qui a été retenue pour la réalisation du test de caractérisation envisagé. Toutefois on a jugé utile de rappeler les autres techniques.

a) Technique de test par comparaison :

Le principe de la technique est illustré par le schéma suivant :



On transmet des informations aux deux microprocesseurs, sous test et de référence, qui sont identiques et proviennent du même constructeur. Mais celui de référence est supposé fonctionner correctement.

On compare leurs réponses, s'il y a une différence, le microprocesseur sous test est déclaré mauvais, sinon il est déclaré bon.

L'inconvénient majeur de cette technique est que les résultats du test sont conditionnés par le microprocesseur de référence. En effet, si les deux circuits présentent le même défaut (défaut de conception) celui-ci ne peut pas être détecté.

b) Technique des patterns enregistrés :

Dans cette technique le microprocesseur est considéré comme un assemblage de circuits logiques, son fonctionnement est simulé par un mini-ordinateur.

Un programme est utilisé pour mesurer en premier lieu les caractéristiques de l'équipement du système à tester dans les conditions dynamiques. Le programme de comparaison est ensuite appliqué à l'équipement. Il engendre les signaux d'entrée. Les sorties sont mesurées et comparées aux réponses enregistrées pour le système. Dans un tel système deux phases sont nécessaires.

- La première phase : est une phase de caractérisation où le mini-ordinateur est utilisé comme référence.
- La deuxième phase : le système fonctionnera en mode de comparaison, en exécutant un programme de test spécifique et en mesurant la réponse.

L'inconvénient majeur de cette technique est la difficulté de simulation de toutes les fonctions d'un circuit aussi complexe que le microprocesseur.

exemple :

La simulation des interruptions est impossible.

c) Technique algorithmique en temps réel :

Dans cette technique, le programme (ou algorithme) de test, écrit en langage machine propre au microprocesseur à tester, est chargé dans une mémoire de masse (disque ou cassette). Chaque instruction est envoyée de la mémoire au microprocesseur sous test qui l'exécute et la réponse fournie est contrôlée par le testeur qui doit être plus rapide que le microprocesseur pour que le processus se déroule en temps réel.

Si la réponse fournie par le microprocesseur est bonne, une autre instruction est envoyée au microprocesseur. Et ce jusqu'à la fin du programme si aucune mauvaise réponse n'a eu lieu.

Cette technique nous permet de générer de longues séquences d'instructions sans utiliser beaucoup de mémoire .

d) Technique LEAD : (Learn Execut And Diagnose).

C'est une technique qui a été développée chez Fairchild. Elle est parmi les meilleures techniques utilisées jusqu'à présent vu ses avantages qui sont: la facilité de mise en oeuvre et de programmation, La connaissance du microprocesseur n'est pas nécessaire, le testeur simule lui-même l'environnement du microprocesseur, et elle permet le test de l'ensemble des instructions sans interruption.

Les phases d'exécution de cette technique sont :

- Création de l'environnement naturel du microprocesseur.
- Génération d'une table de vérité en utilisant un microprocesseur de référence (utilisé une fois) qui exécute un programme diagnostic chargé dans la mémoire du testeur.
- Exécution du programme diagnostic par le microprocesseur sous test et recherche d'erreur par le testeur qui utilise la table de vérité générée préalablement.

c) Technique d'Auto-Test :

Dans cette technique, le microprocesseur détermine lui même s'il est opérationnel ou non. Le principe de base de cette technique repose sur l'exécution, par le microprocesseur à tester, d'une séquence d'instructions préparée à l'avance. Cette séquence est chargée dans un espace d'adressage du microprocesseur et exécutée par ce dernier placé dans un environnement naturel ou simulé (i - e identique ou voisin au système où il est destiné à opérer).

Dans cette technique le microprocesseur est considéré comme une boîte noire chargée d'exécuter des instructions.

Pour l'analyse des résultats de l'exécution, on ne considère que le résultat global de cette dernière, sans considérer les cycles élémentaires mis en oeuvre, ni les conditions dans lesquelles chaque cycle se déroule.

exemple :

On vérifie que le résultat de l'addition de 2 et 3 est bien 5 alors que cette opération nécessite 5 instructions et 51 cycles élémentaires sur un matériel INTEL8080.

Si le microprocesseur à tester est défectueux, il est probable que le programme d'exécution ne se termine pas correctement et que le système tombe en panne. Alors dans cette technique il faut prévoir une mise en garde automatique.

exemple :

Le système imprime un message "bon" après une durée de temps X, le temps X est le temps d'exécution du programme d'auto-test (ce temps est déjà calculé). Si ce message n'est pas imprimé le système sera supposé être défectueux.

De manière optionnelle, des appareils externes peuvent être introduits.

exemple :

Une alarme avec son propre décomptage de temps peut être initialisée au début du test avec un temps $(X + 1)$ (X temps d'exécution du programme d'Auto-test).

Si le décomptage est remis à zéro et le programme n'est pas encore terminé, l'alarme sera activée et signalera qu'il y a une panne du microprocesseur.

IV-2-3- Etablissement du test complet d'un microprocesseur :

Dans ce qui précède, nous avons étudié chaque type de test séparément. Toutefois, un seul type n'est pas suffisant si on veut avoir l'exhaustivité du test. D'où la nécessité de combiner les tests étudiés de façon à établir un test complet., lequel va mieux nous renseigner sur le bon fonctionnement du circuit microprocesseur.

- Un test complet s'exécute en trois phases :

a) Un test paramétrique :

exemple : mesure du courant de consommation

b) Un test fonctionnel minimal : C'est la vérification du bon fonctionnement du circuit pour les valeurs nominales des paramètres d'utilisation.

c) Un test de caractérisation : qui est un test fonctionnel permettant la vérification du bon fonctionnement du circuit pour les gammes des valeurs de paramètres (tension, fréquence, etc...) données par le constructeur.

On rappelle que le test paramétrique nécessite un appareillage de mesure, par contre le test fonctionnel est réalisé en exerçant le circuit avec un programme ou algorithme établi au préalable.

Etablissement du programme de test (fonctionnel)

Ce programme est rédigé dans le langage propre du microprocesseur et doit pouvoir être inclus dans son espace d'adressage. Il doit également pouvoir se dérouler avec une durée raisonnable. Il n'existe malheureusement pas de méthode simple et générale, permettant d'obtenir le programme de test optimum. Cependant on peut énoncer quelques principes directeurs, guidant l'utilisateur pour l'établissement du programme de test :

- Le programme doit contenir toutes les instructions utilisables.
- Il doit contenir les séquences jugées les plus critiques pour les applications envisagées.
- Il doit contenir des séquences de "pire cas", c'est à dire les séquences susceptibles de révéler le maximum de défauts. Or ici on se heurte à un problème délicat, car l'utilisateur d'application n'est pas en mesure, à partir de la seule connaissance du rôle de fonctionnement du circuit, de déterminer ces séquences. Celles-ci sont en effet liées à l'architecture et à la technologie du circuit et à la nature et aux probabilités d'occurrence des divers défauts possibles.

- De par la nature du principe d'auto-test certaines instructions sont utilisées pour tester d'autres.

exemple : pour tester l'instruction d'addition on doit utiliser l'instruction chargement.

Il est donc préférable de les tester elles-mêmes au début du programme de test, ceci permet de minimiser l'effet de masquage de défauts, qui constitue un des défauts inhérent au principe d'auto-test.

- La conception du programme de test doit tenir compte des moyens utilisés pour l'observation des résultats.

V. Application du test de caractérisation :

Lors de la présentation du test de caractérisation, nous avons passé en revue l'intérêt multiple de la région de travail propre. Et pour donner son identité complète, il convient d'ajouter le champ d'application de ce type de test.

Le test de caractérisation défini dans cette étude peut être effectué à trois niveaux.

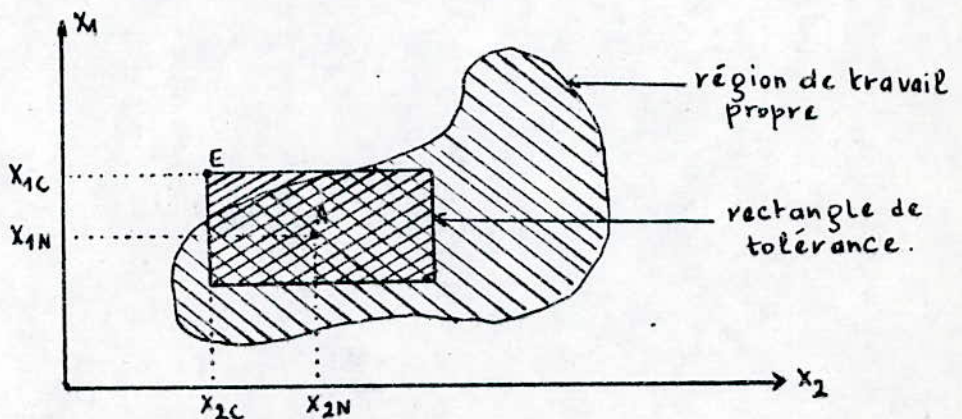
1- Au niveau du projet :

En effectuant ce test sur un circuit prototype on peut vérifier les valeurs déterminées théoriquement comme les valeurs nominales des paramètres spécifiques et leur intervalle de tolérance. Et par conséquent, améliorer la conception si les valeurs réelles, déterminées expérimentalement, ne correspondent pas aux valeurs prévues par la théorie.

Les résultats du test à ce niveau nous permettent de dégager une stratégie de test pour le circuit en fabrication. En effet, ce test nous permet de connaître les paramètres critiques, donc il suffirait de faire le test que pour les valeurs nominales et les valeurs critiques.

exemple :

Le résultat du test d'un circuit microprocesseur prototype est le suivant :



Le point E correspond au cas critique.

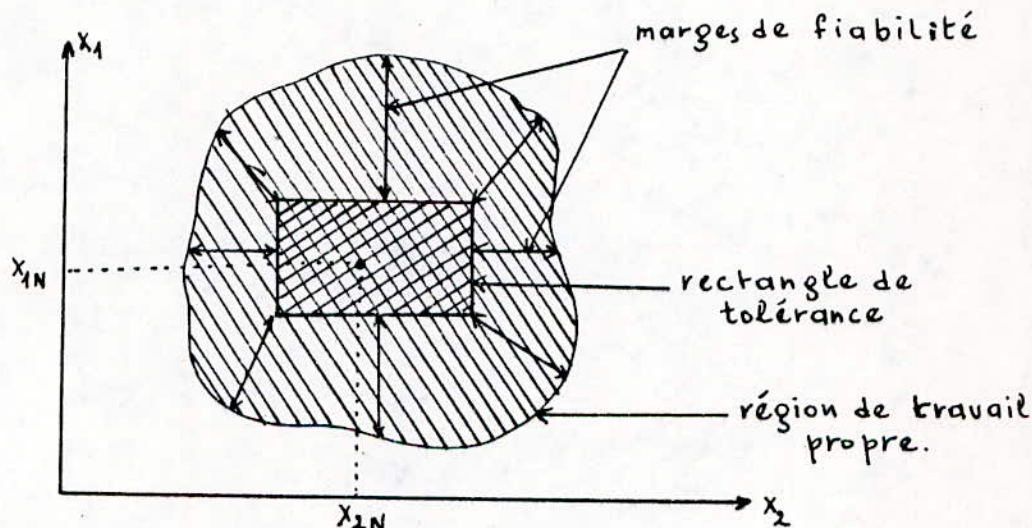
On voit bien qu'il suffirait de faire le test pour les paramètres correspondant aux points A (valeurs nominales) et E pour vérifier que le circuit fabriqué fonctionne correctement ou non dans le rectangle de tolérance.

2- Au niveau de la fabrication :

A ce niveau, ce test est effectué, pour les paramètres dynamiques des microprocesseurs sortis en série, surtout pour voir la qualité des circuits et étudier leur fiabilité.

exemple :

Le tracé suivant est obtenu en effectuant le test de caractérisation sur un circuit .



de ce tracé on peut voir la fiabilité du circuit , ainsi que sa qualité par rapport à d'autres circuits. Ceci peut se voir en comparant leurs régions de travail propre.

3- Au niveau de l'utilisateur :

L'utilisateur aussi peut effectuer ce type de test pour déterminer la cause du mauvais fonctionnement de son circuit placé dans son environnement naturel.

Il peut également le faire quand il se propose de réaliser un système à microprocesseur. Et ce pour connaître les valeurs nominales et les intervalles de tolérance réels des paramètres spécifiques au circuit.

PARTIE

EXPERI.

MENTALE

I- Présentation des circuits MC 6802 et MC 6821

I-1 Le microprocesseur MC 6802.

La figure I-1 représente le schéma interne du microprocesseur MC6802 de MOTOROLA qu'on se propose de tester.

Le MC6802 est un microprocesseur 8 bits dont l'architecture interne est constituée de :

- a) Une horloge interne
- b) Une RAM interne de 128 octets qui commencent de l'adresse $\$ 0000$ à l'adresse $\$ 007F$.
- c) Une unité de commande avec ses lignes de contrôle .
- d) Une unité arithmétique et logique (U.A.L)
- e) Six registres dont trois de 16 bits et les trois autres de 8 bits :
 - Registres 16 bits :

. Le compteur programme (P.C) : Il contient l'adresse suivant celle de l'instruction en cours d'exécution.

. Le pointeur de pile (S.P) : Il contient l'adresse de la position disponible dans une pile externe. Cette pile étant en général en RAM située à une adresse donnée par le constructeur.

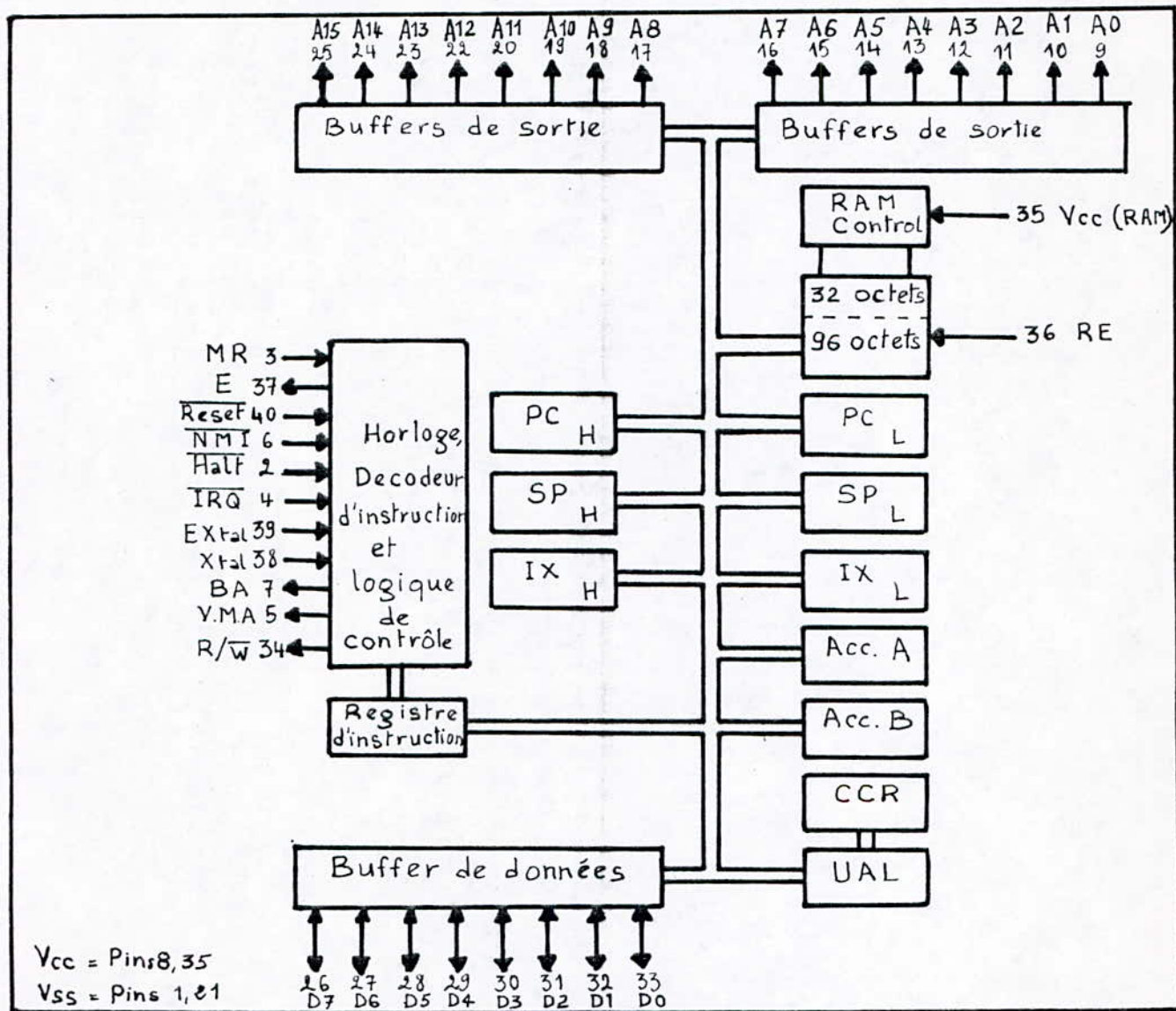
. Le registre d'index : Il peut être utilisé dans le mode des transferts de données ou comme index dans le mode d'adressage indexé.

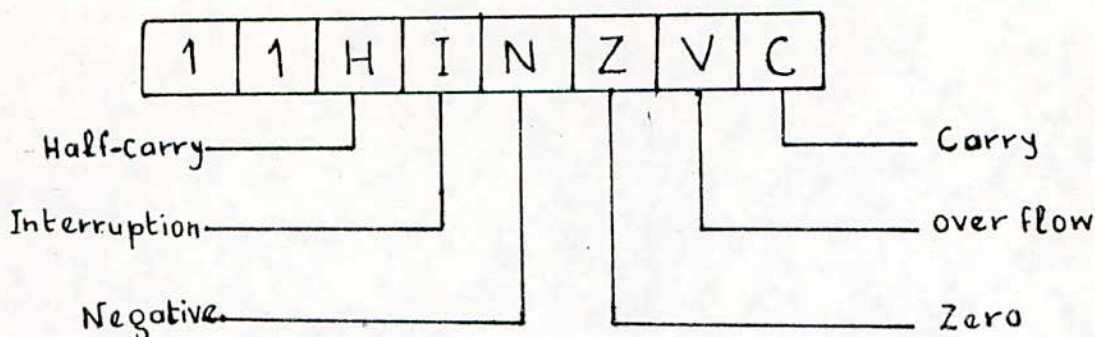
- Registre 8 bits :

. Deux accumulateurs (A et B) : Ils sont utilisés pour contenir des opérandes ou des résultats de l'U.A.L.

. Le registre code conditions (C.C.R) dont 6 bits sont significatifs et deux autres (b_6 et b_7) sont mis à 1. Ce registre indique le résultat d'une opération de l'U.A.L.

figure: I.1 - Organisation interne du MC 6802 de MOTOROLA.





Les bits du C.C.R. du MC6802 Sont testés pour effectuer des branchements conditionnels.

Le logiciel du MC 6802 est compatible avec celui du MC 6800.

I-2- L'interface MC 6821 (P.I.A)

Definition :

Le P.I.A est un interface parallèle programmable dont l'utilisation s'impose chaque fois que l'on a besoin de faire un transfert parallèle (bidirectionnel) entre un microprocesseur est un peripherique.

La figure I-2 représente le schéma interne et externe du P.I.A. qui est constitué par deux port A et B, chacun comprenant :

- Un registre de donnée (ORA, ORB).
- Un registre de contrôle de direction (DDRA, DDRB).
- Un registre de contrôle et d'état (CRA, CRB).
- Des lignes d' E/S (PA0 - PA7 , PB0 - PB7).
- Des lignes pour des signaux de commandes et (respectivement) d'états CA2 , CB2 et CA1 , CB1.

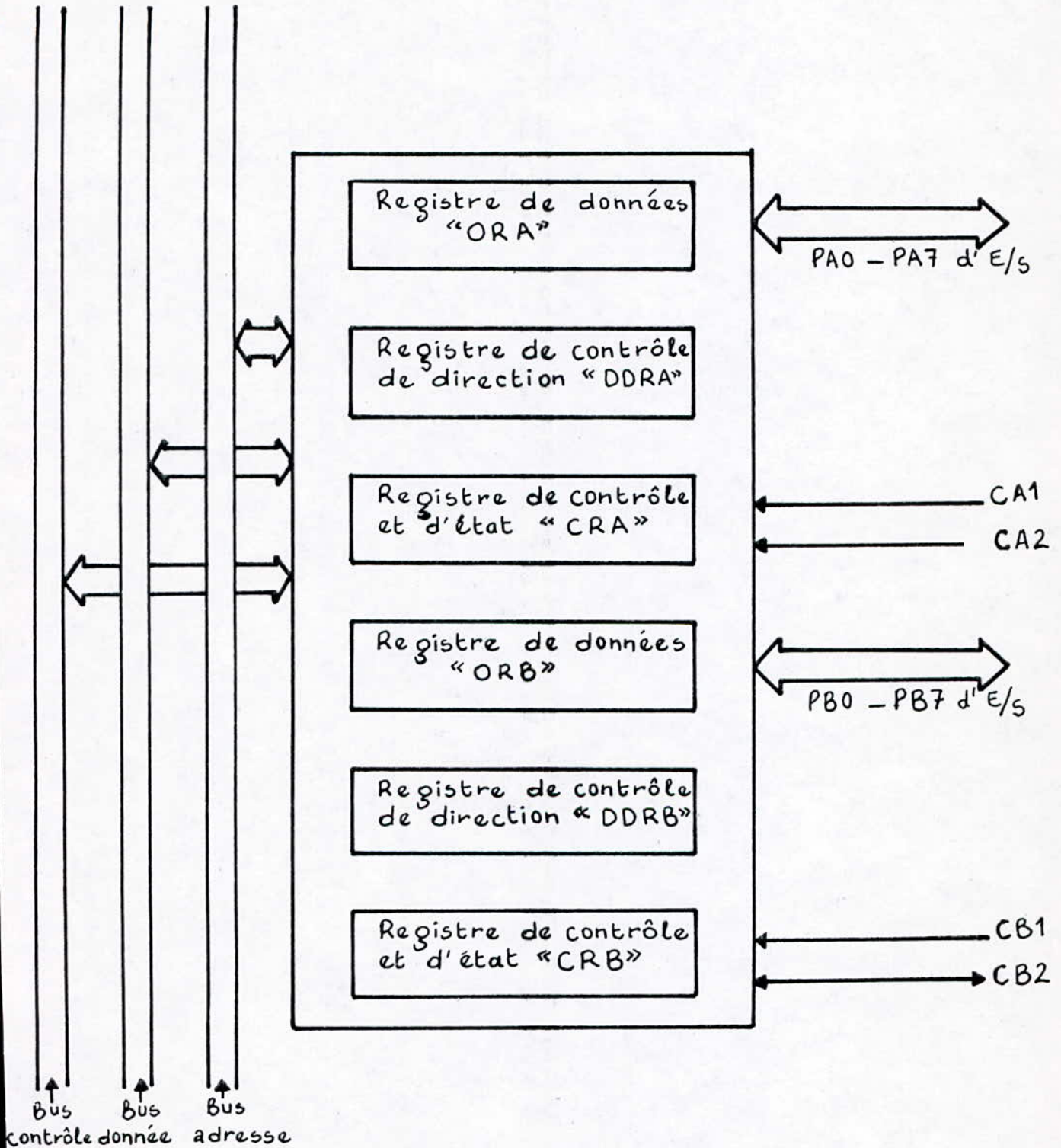
Ces signaux permettent de realiser une dialogue entre l'interface et un (ou des) peripherique.

exemple : dans le cas de l'acquisition.

CA1 : Signal "donnée prête"

CA2 : Signal "donnée reçue"

Figure: I.2 - Schéma interne et externe d'un PIA



II- Test du circuit microprocesseur (MC 6802).

II-1- Principe :

Le test consiste d'abord à varier la valeur du paramètre Vcc (tension d'alimentation) du circuit, dans une plage choisie au préalable, et voir en suite le fonctionnement de ce dernier.

Pour obtenir un lieu de fonctionnement précis, la variation de la valeur du paramètre (Vcc) doit se faire avec un pas petit et régulier. Or, manuellement, il est impossible d'obtenir une bonne précision. Pour y remédier l'idée d'une commande par microprocesseur s'impose. Ceci conduit à une automatisation du test dont l'avantage est de réduire le temps d'exécution du test et d'augmenter la précision du résultat.

Le test effectué est un test fonctionnel se basant sur l'exécution, par le microprocesseur à tester, d'un programme (voir annex) permettant de vérifier l'état de tous les modules internes. Ce programme a été conçu de telle sorte qu'il fasse un test mixte c'est à dire le test des modules un par un avec l'unité de commande.

II-2- Description du schéma général du circuit de test

La figure II-2- représente le schéma général du circuit de test qui comprend :

- Le microprocesseur (I) destiné à subir le test (placé dans son environnement naturel).

- Un microprocesseur (II) utilisé pour la mémorisation du résultat du test, la variation du paramètre Vcc et la commande qui consiste à réinitialiser et lancer le microprocesseur (I) sous test.

II- Test du circuit microprocesseur (MC 6802).

II-1- Principe :

Le test consiste d'abord à varier la valeur du paramètre Vcc (tension d'alimentation) du circuit, dans une plage choisie au préalable, et voir en suite le fonctionnement de ce dernier.

Pour obtenir un lieu de fonctionnement précis, la variation de la valeur du paramètre (Vcc) doit se faire avec un pas petit et régulier. Or, manuellement, il est impossible d'obtenir une bonne précision. Pour y remédier l'idée d'une commande par microprocesseur s'impose. Ceci conduit à une automatisation du test dont l'avantage est de réduire le temps d'exécution du test et d'augmenter la précision du résultat.

Le test effectué est un test fonctionnel se basant sur l'exécution, par le microprocesseur à tester, d'un programme (voir annex) permettant de vérifier l'état de tous les modules internes. Ce programme a été conçu de telle sorte qu'il fasse un test mixte c'est à dire le test des modules un par un avec l'unité de commande.

II-2- Description du schéma général du circuit de test

La figure II-2- représente le schéma général du circuit de test qui comprend :

- Le microprocesseur (I) destiné à subir le test (placé dans son environnement naturel).

- Un microprocesseur (II) utilisé pour la mémorisation du résultat du test, la variation du paramètre Vcc et la commande qui consiste à réinitialiser et lancer le microprocesseur (I) sous test.

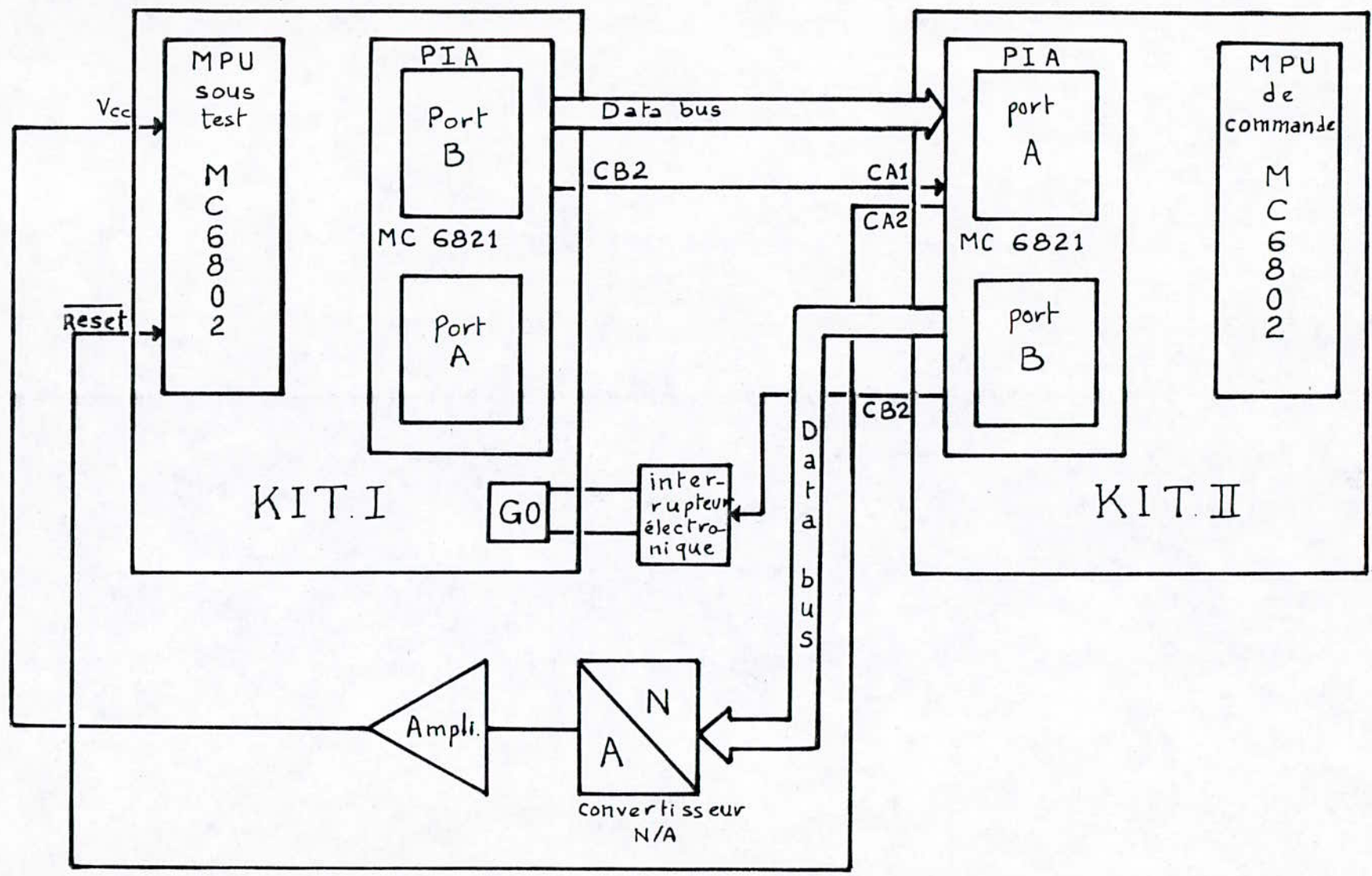


figure: II-2. Schéma synoptique général du circuit de test.

- Un convertisseur numérique - analogique (N/A) et un amplificateur pour appliquer une tension Vcc analogique à la broche Vcc du microprocesseur (I) sous test.

- Un interrupteur électronique permettant de réaliser le lancement du microprocesseur (I) sous test.

Pour faire le test, on doit assembler les composants précités de telle sorte qu'ils définissent une boucle de test. Pour ce faire, on doit établir une communication entre les deux P.I.A des deux microprocesseurs I et II) reliant leurs ports (l'in en sortie, "port B", l'autre en entrée, "port A") par un bus de donnée et en reliant la sortie CB2 du P.I.A (I) à l'entrée CA1 du P.I.A (II).

Le port B du P.I.A (II) est relié aux huit entrées de données du convertisseur qui est suivi d'un amplificateur dont la sortie est connectée à la broche Vcc du microprocesseur (I).

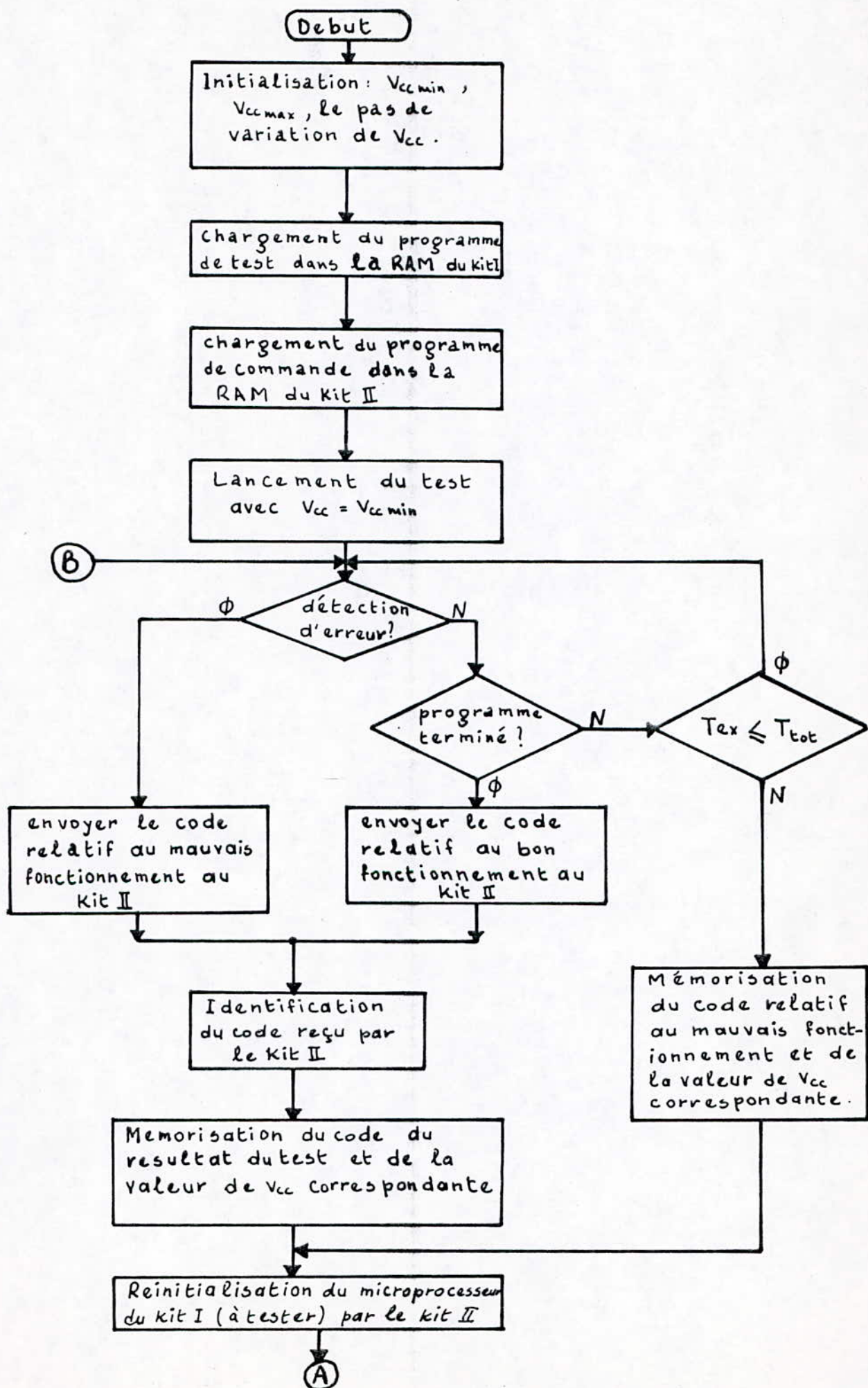
La réinitialisation du microprocesseur (I) se fait grâce à la liaison de la sortie CA2 du P.I.A (II) à l'entrée Reset du microprocesseur (I). Et enfin, le lancement du microprocesseur (I) peut se faire grâce à un interrupteur électronique commandé par la sortie CB2 du port B du P.I.A (II).

II-3- Exécution du test :

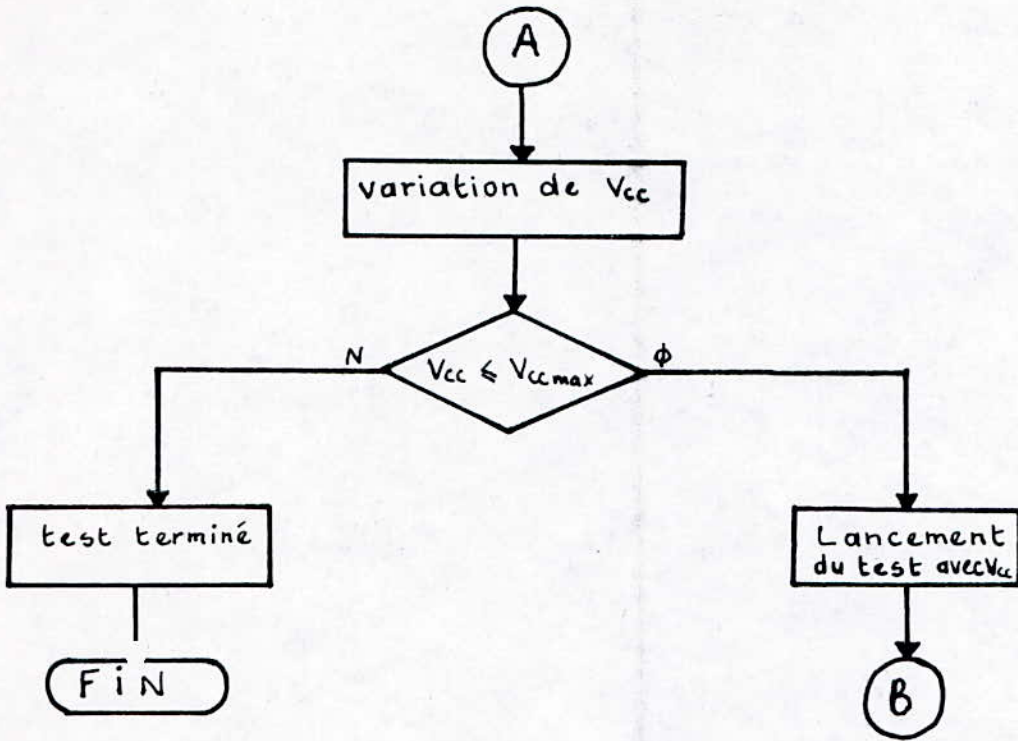
L'organigramme de la figure II-3 représente le déroulement du test.

Après avoir chargé le programme de commande dans la RAM du microprocesseur (II) et le programme de test dans la RAM du microprocesseur (I) à tester, on lance le microprocesseur (II) qui va appliquer, à travers le convertisseur (N/A), la tension Vcc min (définie au préalable) à la broche Vcc du microprocesseur (I), puis le lancer en générant le signal GO par une lecture de son port B de sortie.

Figure: II.3 - Organigramme décrivant le déroulement du test.



(Suite) figure: II.3



* T_{ex} : temps du compteur du temps d'exécution du programme de test.

* T_{tot} : temps d'exécution du programme total quand le microprocesseur fonctionne correctement.

Le microprocesseur (I) exécute, alors, le programme de test et détermine lui même s'il y a une erreur par un test contenu dans le programme de test. Si tel est le cas et à la première erreur détectée il inscrit un code (mot de 8 bits) relatif au mauvais fonctionnement (ex : § 33) dans le registre de données de son port B de sortie et envoie un signal CB2 pour informer le microprocesseur (II) qu'il y a une donnée à prendre. Ce dernier prend en compte la donnée (code), l'identifie par un test, mémorise le code dans une position mémoire en adressage indexé, envoie un signal CA2 pour réinitialiser le microprocesseur (I), augmente la valeur de Vcc et teste si elle n'a pas dépassé Vcc max (défini au préalable). Si cette condition est réalisée, il mémorise la valeur de Vcc en adressage indexé, l'applique à travers le convertisseur à la broche Vcc du microprocesseur (I). Et le cycle continue tant que la plage de variation de Vcc n'est pas entièrement balayée.

Si aucune erreur n'est détectée au cours de l'exécution du programme de test, le microprocesseur (I) inscrit en fin du programme un code relatif au bon fonctionnement (ex : § 44) dans le registre de données de son port B de sortie. Ce code est pris en compte par le microprocesseur (II), lequel va l'identifier par un test, le mémorise en adressage indexé, et la même démarche, décrite plus haut, est répétée.

Mais si le code reçu par le microprocesseur (II) ne correspond pas à celui du bon fonctionnement ni à celui du mauvais fonctionnement, suite à un mauvais fonctionnement du microprocesseur (I), le microprocesseur (II) mémorise de sa propre initiative (i - e par programme) le code relatif au mauvais fonctionnement en adressage indexé avec la valeur de Vcc correspondante.

Cependant, il peut arriver que le microprocesseur (I) se perde (il n'inscrit aucune donnée dans son port B de sortie), suite à un mauvais fonctionnement, ce qui va laisser le microprocesseur (II) toujours en attente et par conséquent interrompre le test. Pour éviter ce problème, le programme de commande a été conçu de telle sorte qu'il compte, au moment de l'attente, le temps d'exécution du programme de test. Si ce dernier dépasse le temps, du programme total, exécuté par un microprocesseur fonctionnant correctement, il mémorise le code relatif au mauvais fonctionnement et réinitialise le microprocesseur (I) et le cycle continue. De cette façon on arrive à réaliser un test automatique.

Pour établir le lieu de fonctionnement du microprocesseur (I) à tester correspondant à la plage de variation de Vcc, il suffit de lire les deux tables, respectivement, de Vcc et des codes (bon ou mauvais).

- CONCLUSION -

Nous déplorons regrettablement le manque de matériel et la presque indisponibilité des Kits de MOTOROLA (MC 6802). Ce qui nous a considérablement retardé dans l'avancement de notre travail.

Le test effectué dans notre travail est un test fonctionnel à l'aide d'un programme regroupant la majorité des instructions du MC 6802. Ce programme est organisé en un seul bloc, de telle sorte qu'il nous permette de voir le fonctionnement global du microprocesseur.

Notre système a été réalisé de façon à ce que le test donne des résultats, qui placés sur un ^{axe} gradué en Vcc, fassent apparaître une ligne de travail propre.

Pour une étude future, on peut envisager, à défaut d'un générateur de fréquences programmable, l'utilisation de deux diodes varicap montées en parallèle sur les deux capacités (C7, C8) associés au quartz, pour pouvoir obtenir une fréquence d'horloge d'entrée variable en fonction de la tension appliquée aux varicaps. Cette dernière peut être commandée par le Kit II. De cette façon, on aura réalisé un système complètement automatisé, avec lequel on peut obtenir une schmo-plot à partir des résultats mémorisés dans la R.A.M. du Kit II. Cependant, on peut obtenir des schmo-plots directement, par une visualisation sur écran ou une impression, en remplaçant le Kit II par un micro-ordinateur avec son écran de visualisation et équipé d'une imprimante.

Nous pensons avoir atteint ce qui a été attendu de nous, et que notre système, avec les améliorations précitées, pourra être utilisé pour tester les microprocesseurs du laboratoire de calculatrice quand ils sont sujets à des problèmes de fonctionnement.

ANNEXE

Programme de test global

Adresse	Langage assembleur	Code	nombre de cycle	Commentaires	
0000	JMP \$E000	7E E0 00	3	Programmation du port B du kit (I) en sortie. (Transmission du code).	
E000 E003	CLR \$E483	7F E4 81	6		
E004 E005	LDAA # \$FF	86 FF	2		
E006 E008	STAA \$E483	87 E4 80	5		
E009 E00A	LDAA # \$2C	86 2C	2		
E00B E00D	STAA \$E483	87 E4 81	5		
E00E	CLRA	4F	2		
E00F	COMA	43	2		
E010 E011	CMPA # \$FF	81 FF	2		
E012 E013	BEQ 03	27 03	4		
E014 E016	JMP \$E17E	7E E1 7E	3	Test des deux acc. A et B, de l'U.A.L et de l'unité de commande.	
E017	CLRB	5F	2		
E018	COMB	53	2		
E019 E01A	CMPB # \$FF	C1 FF	2		
E01B E01D	JMP \$E17E	7E E1 7E	3		
E01E E01F	LDAA # \$0F	86 0F	2		
E020 E021	STAA \$E300	87 E3 00	5		

Adresse	Langage assembleur	Code	nombre de cycle	Commentaires
E022 E024	LDAB \$ E300	F6 E3 00	4	Test des deux acc. A et B, de l'U.A.L et de l'unité de commande.
E025 E027	EORA \$E300	B8 E3 00	4	
E028 E029	BEQ 03	27 03	4	
E02A E02C	JMP\$ E17E	7E E1 7E	3	
E02D E02F	EOR B \$E300	F8 E3 00	4	
E030 E031	BEQ 03	27 03	4	
E032 E034	JMP\$ E17E	7E E1 7E	3	
E035 E036	LDAA# \$33	86 33	2	
E037	CLC	0C	2	
E038	ROLA	49	2	
E039 E03A	LDAB# \$33	C6 33	2	
E03B	CLC	0C	2	
E03C	ROLB	59	2	
E03D E03E	CMPA# \$66	81 66	2	
E03F E040	BEQ 03	27 03	4	
E041 E043	JMP\$ E17E	7E E1 7E	3	
E044	NEGA	40	2	
E045 E046	ADDB # \$10	CB	2	
E047 E048	CMPA# \$AA	81 AA	2	
E049 E04A	BEQ 03	27 03	4	

Adresse	Langage assembleur	Code	nombre de cycle	Commentaires
E04B E04D	JMP\$ E17E	7E E1 7E	3	Test des deux acc. A et B, de l'U.A-L et de l'unité de Commande.
E04E E04F	CMPB # \$AA	C1 AA	2	
E050 E051	BEQ 03	27 03	4	
E052 E054	JMP\$ E17E	7E E1 7E	3	
E055 E056	EORA # \$5A	88 5A	2	
E057 E058	EORB # \$5A	C8 5A	2	
E059 E05B	COM\$E300	73 E3 00	6	
E05C E05E	SUBA\$ E300	B0 E3 00	4	
E05F E060	BEQ 03	27 03	4	
E061 E063	JMP\$ E17E	7E E1 7E	3	
E064 E066	SUBB \$E300	F0 E3 00	4	
E067 E068	BEQ 03	27 03	4	
E069 E06B	JMP\$ E17E	7E E1 7E	3	
E06C E06E	LDA \$E300	B6 E3 00	4	
E06F E071	LDAB \$E300	F6 E3 00	4	
E072	ASR B	57	2	
E073	ASR B	57	2	
E074	ASR A	47	2	
E075	ASR A	47	2	
E076	ABA	1B	2	
E077 E078	CMPA # \$78	81 78	2	

Adresse	Langage assembleur	Code	nombre de cycle	Commentaires
E079 E07A	BEQ 03	27 03	4	Test des deux acc. A et B, de l'u.A.L et de l'unité de Commande.
E07B E07D	JMP\$ E17E	7E E1 7E	3	
E07E	TBA	17	2	
E07F E080	ADDA# \$ 87	8B 87	2	
E081 E082	ADDB# \$ 87	CB 87	2	
E083 E084	CMPA# \$ C3	81 C3	2	
E085 E086	BEQ 03	27 03	4	
E087 E089	JMP\$ E17E	7E E1 7E	3	
E08A E08B	CMPB# \$ C3	C1 C3	2	
E08C E08D	BEQ 03	27 03	4	
E08E E091	JMP\$ E17E	7E E1 7E	3	
E092	CLC	0C	2	
E093	RORA	46	2	
E094	CLC	0C	2	
E095	RORA	46	2	
E096 E097	ORAA# \$ C0	BA C0	2	
E098	CLC	0C	2	
E099	RORB	56	2	
E09A	CLC	0C	2	
E09B	RORB	56	2	
E09C E09D	ORAB# \$ C0	CA C0	2	
E09E	CBA	17	2	
E09F E0A0	BEQ 03	27 03	4	
E0A1 E0A3	JMP\$ E17E	7E E1 7E	3	

Adresse	Langage assembleur	Code	nombre de cycle	Commentaires
EOA4 EOA5	ANDA # \$ 96	84 96	2	<p>Test des deux acc. A et B, de l'U.A.L et de l'unité de Commande.</p>
EOA6 EOA7	ANDB # \$ 96	C4 96	2	
EOA8	ABA	1B	2	
EOA9 EOAA	BEQ 03	27 03	4	
EOAB EOAD	JMP \$ E17E	7E E1 7E	3	
EOAE EOAF	LDAA # \$ A5	86 A5	2	
EOB0	TAB	16	2	
EOB1	COMA	43	2	
EOB2	COMB	53	2	
EOB3 EOB5	STAB \$ E300	F7 E3 00	5	
EOB6 EOB8	ANDA \$ E300	B4 E3 00	4	
EOB9 EOBA	CMPA # \$ 5A	81 5A	2	
EOB8 EOBC	BEQ 03	27 03	4	
EOBD EOBF	JMP \$ E17E	7E E1 7E	3	
EOC0 EOC1	ADDA # \$ 03	8B 03	2	
EOC2 EOC3	ADDB # \$ 03	CB 03	2	
EOC4 EOC5	ANDA # \$ F7	84 F7	2	
EOC6 EOC7	ANDB # \$ F7	C4 F7	2	
EOC8 EOCA	STAB \$ E300	F7 E3 00	5	
EOCB EOCD	ANDA \$ E300	B4 E3 00	4	
EOCE EOCF	CMPA # \$ 55	81 55	4	

Adresse	Langage assembleur	Code	nombre de cycle	Commentaires
E0D0 E0D1	BEQ 03	27 03	4	Test du pointeur de pile, de l'U.A.L et de l'unité de commande.
E0D2 E0D4	JMP\$ E17E	7E E1 7E	3	
E0D5	PSHA	36	4	
E0D6 E0D8	LDA \$E418	86 E4 18	4	
E0D9 E0DA	CMPA# \$55	81 55	2	
E0DB E0DC	BEQ 03	27 03	4	
E0DD E0DF	JMP\$ E17E	7E E1 7E	3	
E0E0	TAB	16	2	
E0E1	PSHB	37	4	
E0E2 E0E4	CMPB \$E417	F1 E4 17	4	
E0E5 E0E6	BEQ 03	27 03	4	
E0E7 E0E9	JMP\$ E17E	7E E1 7E	3	
E0EA	PULA	32	4	
E0EB	PULB	33	4	
E0EC	SBA	10	2	
E0ED E0EE	BEQ 03	27 03	4	
E0EF E0F1	JMP\$ E17E	7E E1 7E	3	
E0F2 E0F4	LDS \$E410	8E E4 10	3	
E0F5 E0F6	LDA# \$33	86 33	2	
E0F7	PSHA	36	4	
E0F8 E0FA	CMPA\$ E410	81 E4 10	4	
E0FB E0FC	BEQ 03	27 03	4	

Adresse	Langage assembleur	Code	nombre de cycle	Commentaires	
E0FD E0FF	JMP\$ E17E	7E E1 7E	3	Test du pointeur de pile, de l'U.A.L et de l'unité de Commande.	
E100 E102	STS \$E310	BF E3 10	6		
E103 E105	LDA \$E310	B6 E3 10	4		
E106 E108	LDAB \$E311	F6 E3 11	4		
E109 E10A	CMPA# \$E4	81 E4	2		
E10B E10C	BEQ 03	27 03	4		
E10D E10F	JMP\$ E17E	7E E1 7E	3		
E110 E111	CMPB# \$09	C1 09	2		
E112 E113	BEQ 03	27 03	4		
E114 E116	JMP\$ E17E	7E E1 7E	3		
E117 E119	LDX # \$9669	8C 96 69	3		
E11A E11C	CPX # \$9669	8C 96 69	3		
E11D E11E	BEQ 03	27 03	4		
E11F E121	JMP\$ E17E	7E E1 7E	3		Test du registre d'index, de l'unité de commande et de l'U.A.L
E122	INX	08	4		
E123 E125	STX \$ E300	FF E3 00	5		
E126 E128	COM \$ E300	73 E3 00	6		
E129 E12B	COM \$ E301	73 E3 01	6		
E12C E12E	LDX \$ E300	FE E3 00	5		

Adresse	Langage assembleur	Code	nombre de cycle	Commentaires	
E12F	INX	08	4	Test du registre d'indexe et de l'U.AL et l'unité de commande	
E130 E132	CPX # \$ 6996	8C 69 96	3		
E133 E134	BEQ 03	27 03	4		
E135 E137	JMP \$ E17E	7E E1 7E	3		
E138	DEX	09	4		
E139	TXS	35	4		
E13A E13C	STS \$ E300	9F E3 00	5		
E13D E13F	LDAA # E300	B6 E3 00	4		
E140 E142	LDAB \$ E301	F6 E3 01	4		
E143 E144	CMPA # \$ 69	81 69	2		
E145 E146	BEQ 03	27 03	4		
E147 E149	JMP \$ E17E	7E E1 7E	3		
E14A E14B	CMPB # \$ 94	C1 94	2		
E14C E14D	BEQ 03	27 03	4		
E14E E150	JMP \$ E17E	7E E1 7E	3		
E151 E152	LDAA # \$ FF	B6 FF	2		Test du c.c.R et de l'unité de commande
E153	TAP	06	2		
E154 E155	BCS 03	25 03	4		
E156 E158	BRA \$ 26	20 26	4		
E159	CLC	0C	2		
E15A	TPA	07	2		
E15B	TSTA	4D	2		

Adresse	Langage assembleur	Code	nombre de cycle	Commentaires	
E15C E15D	BPL 21	2A 21	4	Test du c.c.R et de l'unité de commande.	
E15E E15F	ADDA# \$02	8B 02	2		
E160 E161	BVS 03	29 03	4		
E162 E164	JMP\$ E17E	7E E1 7E	3		
E165	NOP	01	2		
E166 E167	LDAA# \$09	86 09	2		
E168 E169	LDAB# \$0A	C6 0A	2		
E16A	SBA	10	2		
E16B E16C	BMI 03	2B 03	4		
E16D E16F	JMP\$ E17E	7E E1 7E	3		
E170 E172	JSR\$ 000A	BD 00 0A	9		
E173 E174	LDAA# \$44	86 44	2		Ecriture du code relatif au bon fonctionnement (\$44) dans le port B de sortie.
E175 E177	STAA \$E482	B7 E4 82	5		
E178 E17A	LDAA \$E482	B6 E4 82	4		
E17B E17D	JMP\$ E186	7E E1 86	3		
E17E E17F	LDAA# \$33	86 33	2		Ecriture du code relatif au mauvais fonctionnement. dans le port B de sortie.
E180 E182	STAA \$E482	B7 E4 82	5		
E183 E185	LDAA \$E482	B6 E4 82	4		
E186	SWI	3F	12		
000A	NOP	01	2	sous-programme (Subroutine)	
000B	RTS	39	5		

Programme de commande

Adresse	Langage assembleur	code	nombre de cycle	Commentaires
E000	CLR\$E481	7F		Programmation du port A du kit (II) en entrée (acquisition)
E002		E4 81		
E003	CLR\$E480	7F		
E005		E4 80		
E006	LDA A #	86		
E007	\$2C	2C		
E008	STAA	B7		Programmation du port B du kit (II) en sortie (commande)
E00A	\$E481	E4 81		
E00B	CLR\$E483	7F		
E00D		E4 83		
E00E	LDA A #\$FF	86		
E00F		FF		
E010	STAA	B7		
E012	\$E482	E4 82		
E013	LDA A #\$2C	86		
E014		2C		
E015	STAA	B7		chargement du registre d'index pour servir au stockage en indexé
E017	\$E483	E4 83		
E018	LDX # \$	CE		
E01A	E300	E3 00		chargement de l'acc. A par la valeur minimale de Vcc.
E01B	LDA A #	86		
E01C	\$67	67		application de Vcc au microprocesseur (I) sous test
E01D	STAA	B7		
E01F	\$E482	E4 82		
E020	LDA A	B6		Lancement du programme de test, par CB2.
E022	\$E482	E4 82		
E023	STAA \$	A7		stockage de la valeur de Vcc en indexé.
E024	00, X	00		

Adresse	Langage assembleur	Code	nombre de cycle	Commentaires
E025	LDAB	C6	2	Test sur la transition du bit 7 du CRA (presence de donnée?) Et Comptage du temps d'exécution du programme de test. si ce dernier dépasse la valeur limite, le programme se branche à l'adresse \$E047 (Mauvais fonctionnement du microprocesseur (I)).
E026	#\$26	26		
E027	DEC B	5A	2	
E028	BEQ	27	4	
E029	1D	1D		
E02A	TST \$E481	7D	6	
E02C		E4 81		
E02D	BPL	2A	4	
E02E	F8	F8		
E02F	LDAB \$E480	F6 E4	4	acquisition et reinitialisation du microprocesseur (I) à l'aide de CA2. Et identification du code (bon ou mauvais).
E031		80		
E032	CMPB#	C1	2	
E033	\$33	33		
E034	BNE	26	4	
E035	0D	0D		
E036	STAB \$FF,X	E7 FF	6	stockage du code en indexé
E038	INCA	4C	2	Augmentation de la valeur de Vcc.
E039	INCA	4C	2	
E03A	INCA	4C	2	
E03B	INX	08	4	Incrémentation de l'indexe.
E03C	CMPA # \$9A	81 9A	2	test sur la valeur de Vcc. si elle dépasse Vccmax, le test est arrêté.
E03E	BLS	23	4	
E03F	DD			
E040	JMP E04E	7E E0	3	Identification du code, s'il est bon ou non.
E042		4E		
E043	CMPB#	C1	2	
E044	\$44	44		
E045	BEQ	27	4	
E046	EF	EF		
E047	LDAB \$E480	F6 E4	4	génération du signal CA2 pour reinitialiser le kit (I).
E049		80		
E04A	LDAB#	C6	2	chargement du code relatif au mauvais fonctionnement et branchement à l'adresse \$E036
E04B	\$33	33		
E04C	BRA	20	4	
E04D	E8	E8		
E04E	SWI	3F	12	STOP, arrêt du programme

- BIBLIOGRAPHIE -

*1) M. AUMIAUX

L'Emploi des microprocesseurs

Ed. MASSON 1982

*2) D. GIROD et R. DUBOIS

Au coeur des microprocesseurs

Ed. Eyrolles 1982

*3) Charles M; GILMORE

Introduction aux microprocesseurs

Ed. MC GRAW - HILL

*4) A. LESEA et R. ZAKS

Techniques d'interface aux microprocesseurs

Ed. SYBEX

*5) B. KAMINSKA

The concept of proper operation region

In digital integrated circuit testing.

*6) M. SOUFI et M. BOUTRIA

Méthodes de test d'une famille de micro - ordinateur

Projet de fin d'études Janvier 1985

*7) B. SCHUSHEIM

A flexible approach to microprocesseur testing

Computer Design March 1976

*8) W. BARRACLOUGH, A.C.L. CHIANG and W. SOHL

Techniques for testing the microcomputer family

Proceedings of the IEEE Vol. 64 N°6 Jun
1976.

*9) G. DEMARS et J.C. RAULT

Rapport final d'une étude des méthodes de test des microprocesseurs
et des équipements comprenant des microprocesseurs: Août 1977

*10) Notices techniques du MC 6800 et du MC 6802 .