

الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

19/84

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

Des

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT : ELECTRONIQUE

PROJET DE FIN D'ETUDES

SUJET

GESTION D'UN INTERFACE DE COMMUNICATION
ENTRE LE MICROORDINATEUR DEC LSI II
ET LE COORDINATOGRAPHE

Proposé par :

M. MILOUDI

Etudié par :

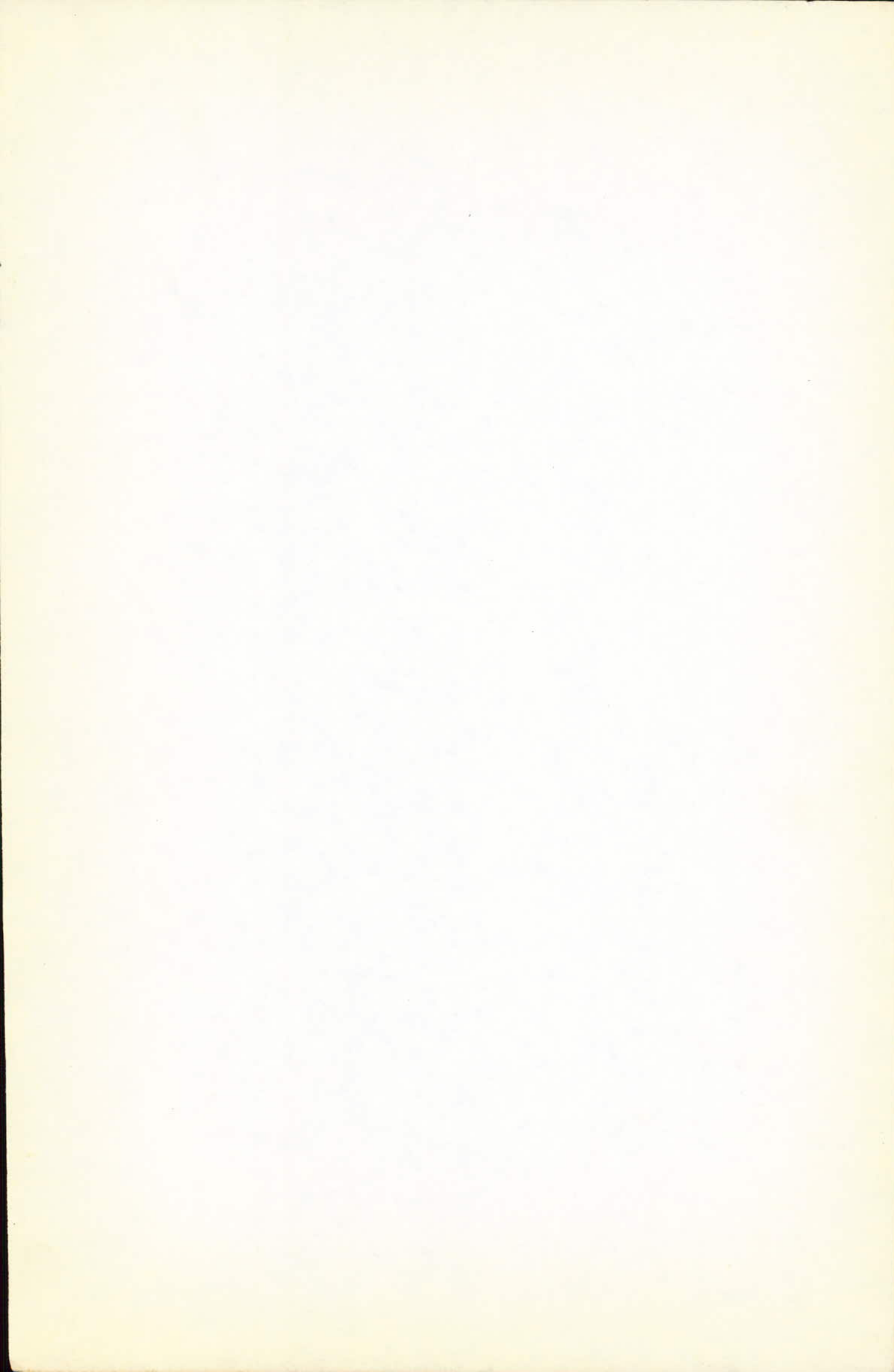
R. BOUTEKEDJIRET
M. STIET

Dirigé par :



PROMOTION :

JANVIER 84



D E D I C A C E S

à mes parents.
à ma famille.
à tous mes ami(e)s.
Rachida.



à mes parents.
à mon frère Arezki.
à mon frère Mohamed et sa femme.
à ma soeur Fatma et ses enfants.
à tous mes frères et soeurs.
à tous mes ami(e)s.
Malika.

REMERCIEMENTS

Nous remercions le directeur de l'INI , pour avoir accepté la réalisation de ce projet au sein de son établissement.

Nousexprimons notre vive reconnaissance à monsieur le chef du département électronique , monsieur H.ADANE , de nous avoir donné une formation en électronique.

Que monsieur M.MILOUDI chef du laboratoire hardware de l'INI soit assuré de nos sentiments de profonde gratitude pour son aide dévouée et efficace,

Madame MILOUDI nacira a supervisé notre travail depuis notre arrivée elle a toujours su trouver le temps de s'interresser à notre travail malgré ses nombreuses occupations , qu'elle trouve l'expression de notre sincère reconnaissance.

Nous n'oublierons pas d'adresser nos remerciements à tous les membres du jury qui ont bien voulu accepter de juger notre travail.

Nous tenons à remercier messieur DJERADI hocine et AIT-DJAFFER brahim pour toute l'aide qu'ils nous ont apportée.

Enfin nous n' oublierons pas d'adresser également nos remerciements aux chercheurs et techniciens du laboratoire hardware ainsi qu'au secrétaire du département électronique de l'ENPA.

P L A N

Introduction

I - Description fonctionnelle du système

- I - 1 . Schéma synoptique du système
- I - 2 . Etude du moteur pas à pas
 - I - 2.1. Définition
 - I - 2.2 Types de moteurs pas à pas
 - I - 2.3. Domaines de définition
 - I - 2.4. Caractéristiques générales
 - I - 2.5. Choix du moteur

I - 3 . Etage de puissance

I - 4 . Etude du MC 6800 et sa famille

- II- I - 4. 1. Etude du MC 6800
- II - 4. 2. Etude du 6820
- II - 4. 3. Etude du 6850

I - 5 . Cartes d'interface

- I - 5. 1. Interface avec les moteurs
- I - 5. 2. Interface avec le DEC

II - Gestion du système

II - 1. Programmes pour l'exploitation du moteur

- II - 1.1. Mode d'excitation simple phase
- II- 1. 2. Mode d'excitation double phase
- II- 1. 3. Mode d'excitation demi-pas
- II- 1. 4. Détermination de la fréquence absolue
- II - 1. 5. Détermination de la fréquence absolue
- II - 1. 6. Accélération et décélération

II - 2. Programme général

II - 3. Sous programme pour la réalisation des pastilles et des segments

Conclusion

INTRODUCTION

Depuis l'avènement des ordinateurs, le domaine de l'automatique et de la robotique a vu un développement important, c'est ainsi que plusieurs réalisations ont été mises à jour.

L'utilisation récente des moteurs pas à pas a encore mieux concrétisé cette évolution, en particulier la commande de ces moteurs par microprocesseur a ouvert de nouvelles perspectives.

Le travail que nous présentons ici, consiste en la conception d'un interface de communication entre le microordinateur DEC LSI 11 et un coordinatographe.

Dans le chapitre I, nous avons donné une étude fonctionnelle du système où nous développons plus particulièrement, le moteur pas à pas et sa commande, l'électronique de puissance et les cartes d'interfaces.

Quant au chapitre II, il est consacré à la gestion du système. Nous y avons détaillé le logiciel permettant la commande de moteur par microprocesseur.

Les programmes nécessaires au tracé des pastilles et des traits de circuits imprimés sont aussi donnés et largement discutés.

I. DESCRIPTION DU SYSTEME

I. 1 : SCHEMA SYNOPTIQUE DU SYSTEME

Le schéma synoptique du système permettant la réalisation d'un circuit imprimé est représenté par la figure 1. Le micro-ordinateur D E C envoie toutes les données et les informations relatives au circuit imprimé, dont le tracé sera réalisé sur un film grattable à l'aide d'un coordinatographe.

Les données arrivent à l'unité centrale par l'intermédiaire de la carte V24, qui réalise l'interfaçage entre le D E C et le microprocesseur.

Ces informations seront stockées dans la mémoire vive (RAM) et seront traitées par l'unité centrale, grâce aux programmes contenus dans la PROM.

L'ensemble dialogue avec deux moteurs par l'intermédiaire d'une carte d'interface entre le microprocesseur et les moteurs. Ces deux moteurs servent à faire mouvoir et positionner l'aiguille du coordinatographe. Le moteur qui s'impose pour cette tâche est le moteur pas à pas, dont les caractéristiques et le fonctionnement feront l'objet du paragraphe suivant.

II. 2 : ETUDE DU MOTEUR PAS A PAS

II.2-1 : DEFINITION

Un moteur pas à pas est un appareil qui convertit des informations numériques en mouvements analogiques.

Le fonctionnement du moteur pas à pas consiste en une suite de déplacements discrets, d'amplitude uniforme, appelée pas. Chaque impulsion provoque un déplacement d'un pas.

II.2-2 : TYPES DE MOTEURS ET PRINCIPE DE FONCTIONNEMENT

II.2-2-1: Les Moteurs à Aimant Permanent

Ils fonctionnent par la réaction entre un champ électromagnétique et un aimant permanent.

Le stator possède un certain nombre de bobinages, alimentés l'un après l'autre, en créant un champ tournant de manière discontinue.

Le rotor consiste en un aimant, et se positionne de telle façon que le moment magnétique de l'aimant s'aligne sur le champ tournant.

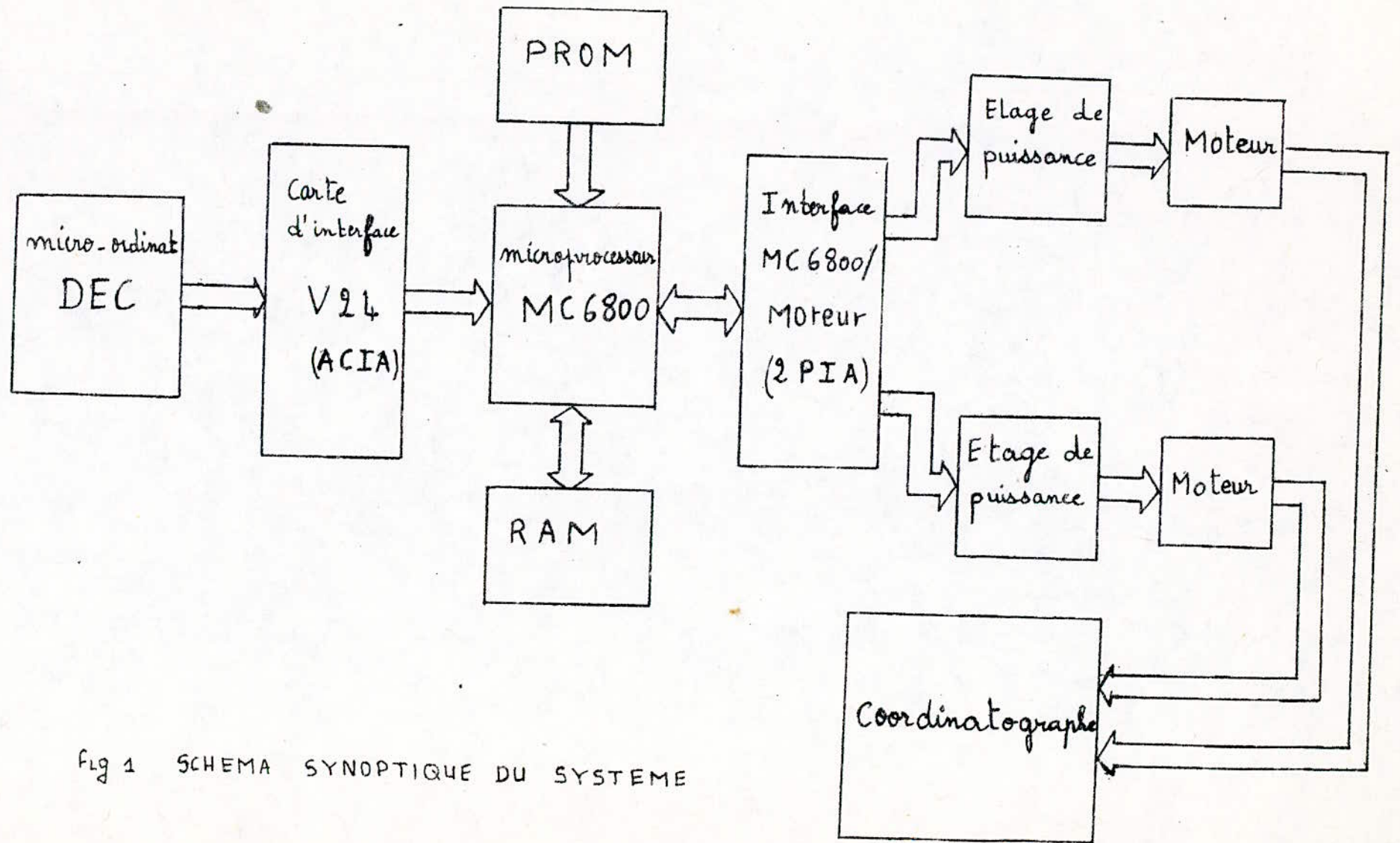


Fig 1 SCHEMA SYNOPTIQUE DU SYSTEME

II.2-2-2: Les Moteurs à Reluctance Variable

Ils fonctionnent par la réaction entre un champ magnétique et un matériau en fer doux.

Le rotor est un noyau en fer doux, dont la reluctance magnétique n'est pas constante suivant tous les angles appelés dents.

Pour obtenir une position stable, le rotor se positionne dans la direction magnétique minimale.

II.2-2-3: Les Moteurs Hybrides

Ils utilisent une combinaison de l'effet d'un aimant permanent, et d'une reluctance variable.

Le stator est constitué d'un certain nombre de bobines disposées chacune, autour d'un pôle saillant, comportant un certain nombre de petites dents.

Le rotor est habituellement double, et est constitué de deux masses polaires, portant également des dents en nombre légèrement différent du nombre de dents du stator. Les deux masses sont séparées par un aimant axial.

La principale caractéristique des moteurs hybrides est de permettre d'obtenir des couples importants, à taille donnée, avec un grand nombre de pas par tour, ce qui s'accompagne d'une haute fréquence de travail, tant en régime réversible, qu'en survitesse.

II.2-3 : DOMAINES DE FONCTIONNEMENT

Les domaines de fonctionnement d'un moteur pas à pas sont limités par les caractéristiques couple-vitesse. La figure 2 nous donne les différents domaines de travail du moteur.

- Les domaines 1 et 3 sont stables; le moteur peut démarrer et s'arrêter en un seul pas.
- Le domaine 2 est instable; à très basse fréquence ou lors d'un arrêt, le mouvement du moteur pas à pas est caractérisé par un phénomène d'oscillations amorties.
- Le domaine 4 est possible après un démarrage dans le domaine 3, et une accélération progressive.
- Le domaine 5 est impossible.

Fd : Fréquence limite de démarrage

Fa : Fréquence limite absolue

Ff : Fréquence limite de freinage.

II.2-4 : CARACTERISTIQUES GENERALES DES MOTEURS PAS A PAS

II.2-4-1: Caractéristiques Statiques

Elles sont conditionnées par la construction mécanique du moteur.

- La résolution : c'est le nombre de pas par tour.
- L'écart ou l'angle de puissance: il représente l'angle de retard du rotor, par rapport à l'axe du champ magnétique qui est provoqué par une charge extérieure.
- Le couple de décrochage: c'est le couple qu'on peut appliquer au moteur arrêté, mais alimenté sans provoquer de rotation.

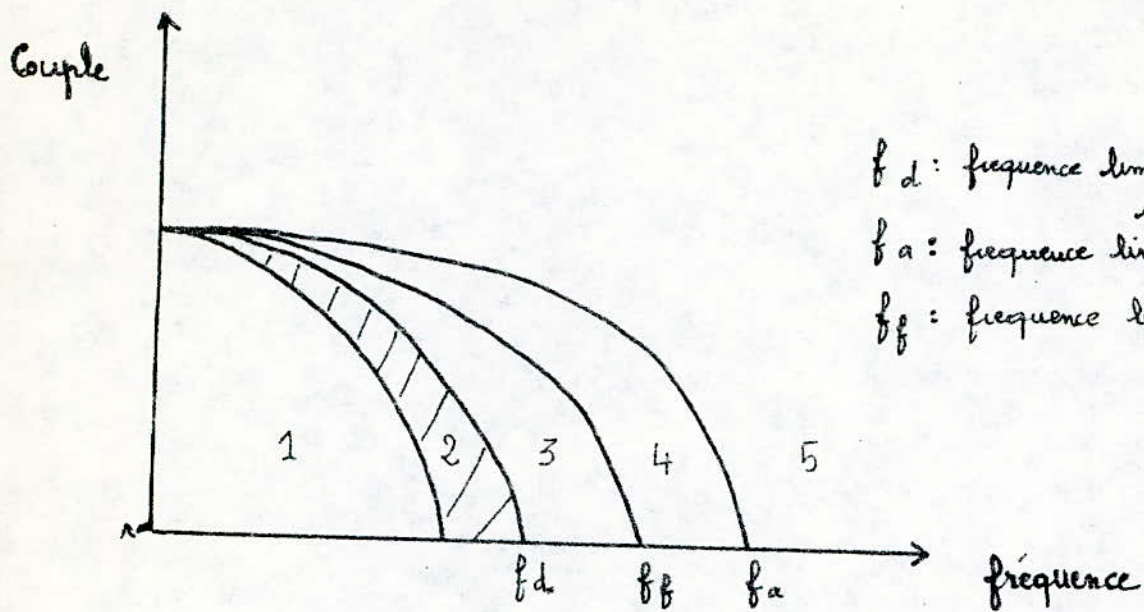


fig 2 : domaines de fonctionnement.

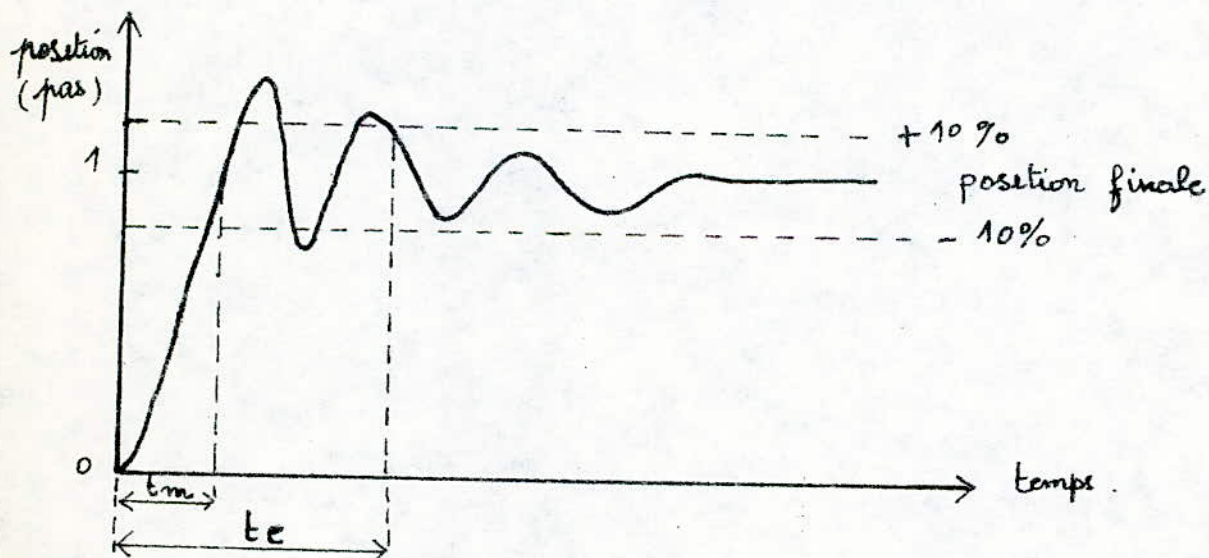


fig 3 : temps de montée et temps d'établissement.

- Le couple d'immobilisation: c'est le couple qu'on peut appliquer au moteur et non alimenté, sans provoquer de rotation.

I.2-4-2: Caractéristiques Dynamiques

La rotation du moteur pas à pas dépend de la commutation de ses phases. L'ensemble des commutations est commandé par des impulsions dont la fréquence f est directement liée au nombre de pas par seconde.

- Fréquence maximale réversible :

C'est le nombre maximum d'impulsions par seconde pouvant être appliqué à un moteur pas à pas dans le sens direct ou inverse, et qui se traduit par des pas synchronisés avec les impulsions.

- Fréquence maximale de démarrage :

C'est la fréquence maximale jusqu'à laquelle le moteur peut suivre la commande au moment de démarrage. Cette fréquence est légèrement plus haute que la fréquence maximale réversible.

Lorsqu'une impulsion est appliquée à un moteur pas à pas au repos, le rotor se met en marche progressivement et avance d'un pas. Il ne s'arrête pas immédiatement à sa nouvelle position d'équilibre, mais oscille autour de cette dernière pendant un bref moment, comme le montre la figure 3 .

- Temps d'établissement :

C'est le temps nécessaire pour établir la position du rotor, autour d'une position d'équilibre, avec une bande d'erreur de position donnée. Sur la figure 3, la bande est prise à $\pm 10\%$ d'un pas.

Pour atteindre la zone de survitesse, la fréquence d'impulsions du moteur doit être augmentée progressivement, mais en un temps extrêmement court, depuis un point situé en dessous de la fréquence maximale de démarrage.

Le temps d'établissement est fonction de l'inertie et du frottement du système moteur ainsi que de l'alimentation.

- Temps de montée :

C'est le temps nécessaire pour passer d'une position d'équilibre à une autre. Il est également limité par l'inertie et le frottement du moteur, ainsi que l'alimentation de ce dernier.

Le temps de montée et le temps d'établissement sont deux facteurs importants lors de l'application du moteur pas à pas. Si le dépassement ou l'ondulation de position sont tolérés on peut réduire le temps de montée; mais ce dernier sera sacrifié si le dépassement de position n'est pas toléré.

Lors de l'augmentation progressive de la fréquence des impulsions, il arrive un moment où la fréquence de pas est égale à la fréquence des oscillations : la zone instable est atteinte. Au delà, chaque commutation a lieu avant que les oscillations du pas précédent n'aient cessé. Ceci est illustré dans la figure 4.

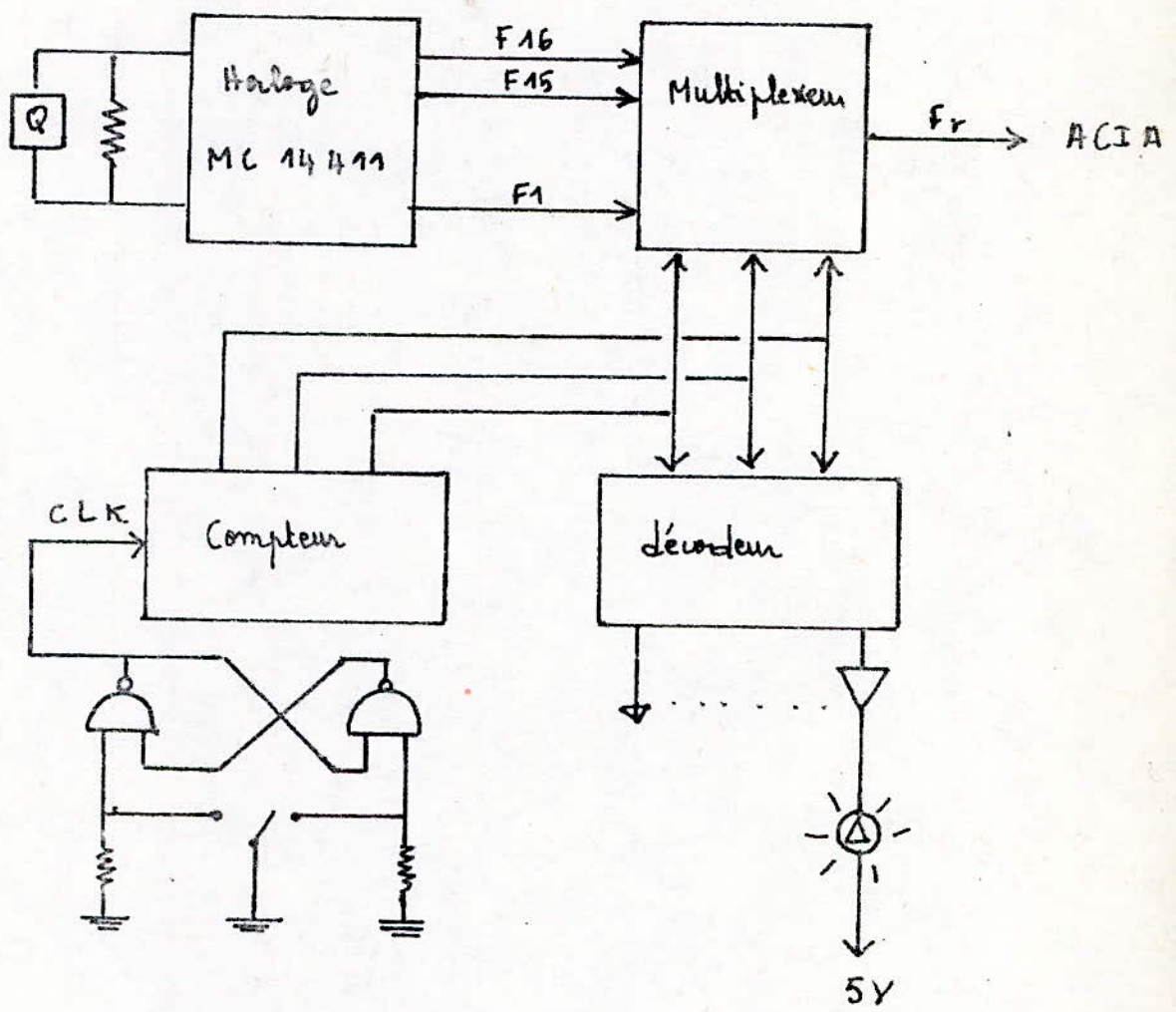


fig 12 Sélection de la vitesse de transmission.

Les circuits MC 1488 et MC 1489 fournissent un système d'interface complet entre les niveaux logiques TTL et les niveaux définis par la RS 232, ils sont donc utilisés pour l'adaptation des deux niveaux logiques. Deux autres lignes bénéficient du même traitement, ce sont la CTS et la RTS
CTS : prêt à émettre, entrée à haute impédance permettant le contrôle automatique de la transmission.
RTS : demande pour émettre, sortie permettant le contrôle automatique d'un modem ou d'un périphérique par le M.P.U. l'état de cette sortie (actif ou inactif) dépend des bits 5 et 6 du registre de contrôle de l'ACIA.

-B. Sélection de la vitesse de transmission

IL faut fournir à l'aide d'un circuit extérieur, un signal d'horloge carré sur chacune des broches TXC et RXC.
L'horloge est réalisée à l'aide d'un générateur de bauds le MC 14411 qui est piloté par un quartz de 1,943 MHz. L'ensemble des circuits utilisés avec l'horloge permettent de sélectionner une fréquence et indique la valeur de cette dernière par un voyant (LED).
Ainsi un compteur linéaire (74161) est incrémenté par un bouton poussoir à travers un circuit anti-rebond et les sorties de ce compteur attaquent à la fois les entrées sélects d'un décodeur 3-----8 (74138) pour allumer la LED correspondante à la fréquence sélectionnée, et les entrées sélects d'un multiplexeur 8-----1 (74151) pour sélectionner une fréquence.
Les sorties du décodeur attaquent les LED à travers des amplificateurs à collecteur ouvert pour satisfaire ces dernières en tension et en courant.

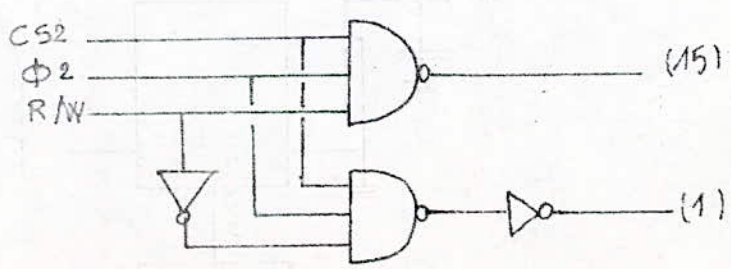
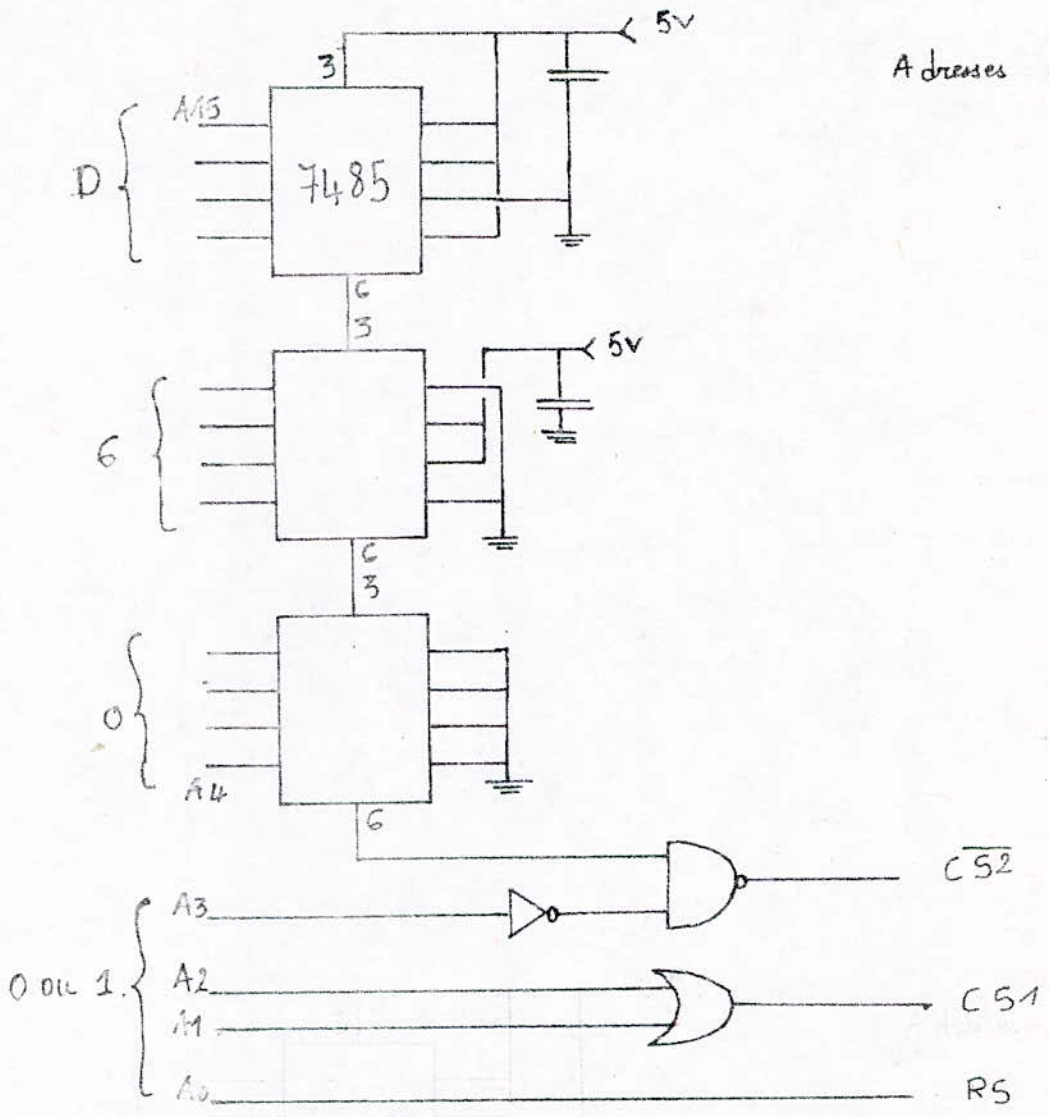
-C. Interface du bus de données

Le bus de données D0 ---D7 de l'ACIA est relié au bus de données du microprocesseur à travers des buffers amplificateurs dont la logique de contrôle est représentée par le même tableau que pour la carte précédente.

-D. Décodage d'adresse

Le circuit de décodage d'adresse est à peu près similaire à celui utilisé dans la carte d'interface précédente.
la figure illustre notre circuit de décodage

Adresses D600 - D60



Logique de commande des buffers.

II-1. Programmes d'exploitation du moteur pas à pas

Nous allons écrire dans ce sous chapitre , tous les programmes qui nous permettent d'exploiter le moteur pas à pas que nous utilisons pour le positionnement de l'aiguille d'un coordinatographe.

Le moteur peut fonctionner selon plusieurs régimes:

- démarrage
- régime permanent
- freinage
- arrêt

A c haque régime de fonctionnement correspondent différents modes d'excitation des phases du moteur, l'étude de chacun d'eux va nous permettre un choix judicieux du mode à utiliser.

Comme nous l'avons déjà vu, le moteur possède quatre bobinages 1, 2, 3, 4.

En leur appliquant des impulsions de courant dans le bon ordre le moteur fait un pas.

En effet, la rotation de l'axe du moteur dans un sens ou dans l'autre est obtenue par la commande adéquate des quatre bobinages.

Les quatre lignes d'un port d'E/S d'un PIA sont reliées aux transistors qui constituent l'étage de puissance alimentant les phases du moteur -(fig.6).

Le niveau "1" sur l'axe des sorties du PIA fait conduire le transistor de puissance et le bobinage correspondant est excité.

Plusieurs séquences sont possibles; cependant nous nous intéressons aux séquences suivantes:

- séquences basse puissance (excitation simple phase)
- séquence normale (excitation double phase)
- séquence demi-pas

II.f-1: MODE D'EXCITATION SIMPLE PHASE

Dans ce mode d'excitation, un seul bobinage est excité à la fois; il suit l'ordre 1,2,3,4....

Diagramme:

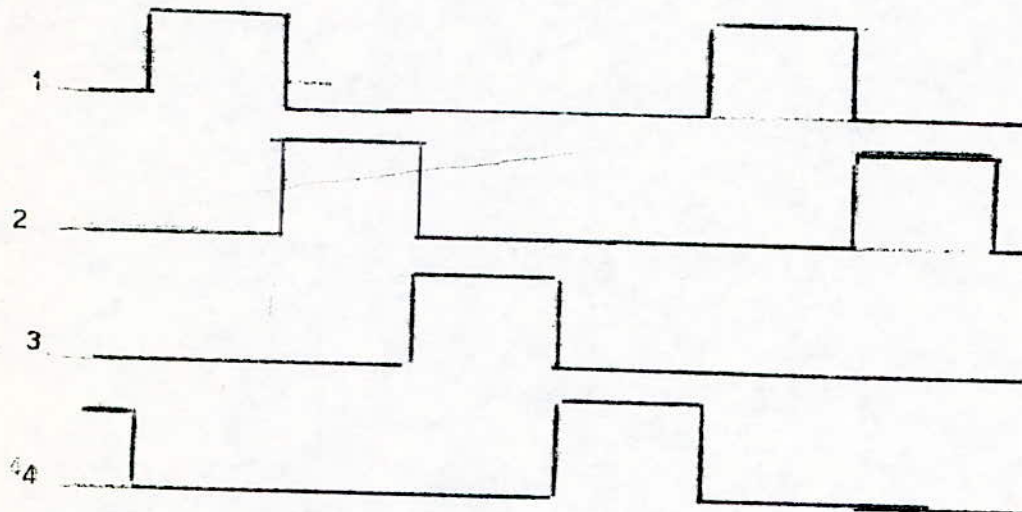


Table de vérité:

tps	1	2	3	4	Etat	\$
a	1	0	0	0	Et1	8
b	0	1	0	0	Et2	4
c	0	0	1	0	Et3	2
d	0	0	0	1	Et4	1

Les quatre mots de quatre bits constituant cette table de vérité sont les états Et1, Et2, Et3, Et4,. L'ordre de succession dans le temps de ces quatre états détermine le sens de déplacement du moteur; le séquençement Et1, Et2, Et3, Et4, Et1....provoque une rotation de l'axe du moteur dans un sens que nous appellerons, par convention, sens positif.

Le séquençement ET4 , ET3 , ET2 , ET1, ET4.....etc
provoque une rotation de l'arbre du moteur dans l'autre sens qui sera donc le sens " négatif " .

Chaque ^{état} est réalisée de la manière suivante: le mot correspondant à l'état est chargé dans le PIA.

Ce mot de quatre bits est maintenu dans l'interface pendant une durée - (définie par temporisation.

Pendant approximation un temps τ , le bobinage correspondant au bit contenant la valeur "1" est excité.

A la fin de la temporisation le prochain mot de quatre bits est chargé dans le PIA de la même manière que précédemment.

La vitesse de déplacement du moteur est déterminée par la valeur T et par la valeur de la tension d'alimentation du bobinage.

Nous obtenons le délai T voulu grâce à un sous programme, ce délai est obtenu par comptage.

On charge un registre de comptage avec une valeur puis on le décrémente.

Le programme boucle sur lui-même et continue à décrémenter jusqu'à ce que le compteur arrive à la valeur "0".

Le temps total requis pour ce processus va réaliser le délai souhaité.

Le programme d'une temporisation sera le suivant:

```
LDAA  # $X
ENC   DECA
      BNE   ENC
      RTS
```

LDAA en mode adressage immédiat prend deux cycles

DECA prend deux cycles

BNE prend quatre cycles

RTS prend cinq cycles

Le délai réalisé par la temporisation est :

$$T = (7+6X)t_{\text{cycle}}$$

Comme la fréquence d'horloge de notre système est de 1 MHz, un cycle correspondra à une microseconde.

Le temps maximum réalisé par un programme de temporisation est:

$$T_{\text{max}} = (7 + 255 \times 6) = 1539 \text{ microsecondes}$$

Or notre moteur, réalise pour l'exécution d'un pas, un temps compris entre 2000 et 3000 microsecondes.

Nous devons donc utiliser un registre de 16 BITS. nous pouvons utiliser pour ce faire deux accumulateurs (A et B) tout en sauvegardant leurs contenus dans des cases mémoire.

L'accumulateur A comprend l'octet inférieur, l'accumulateur B comprend l'octet supérieur.

L'octet inférieur sera chargé avec la valeur Y et l'octet supérieur sera chargé avec la valeur X.

L'accumulateur A entrera dans une boucle de décrémentation, à chaque fois qu'il sera décrémenté à "0", l'accumulateur B sera décrémenté.

Le programme réalisant la temporisation est:

```

TEMPO      LDAB  + $X
ENC1       LDAA  + $Y
ENC2       DECA
           BNE   ENC2
           DECB
           BNE   ENC1
           RTS
    
```

Le temps requis par ce sous programme est:

$$\bar{T} = (7 + (8 + 6 Y)) \text{microsecondes}$$

Nous pouvons varier le délai réalisé par ce programme à volonté en changeant les valeurs X et Y. Le temps maximum que peut réaliser ce programme est égal à 0,4ms.

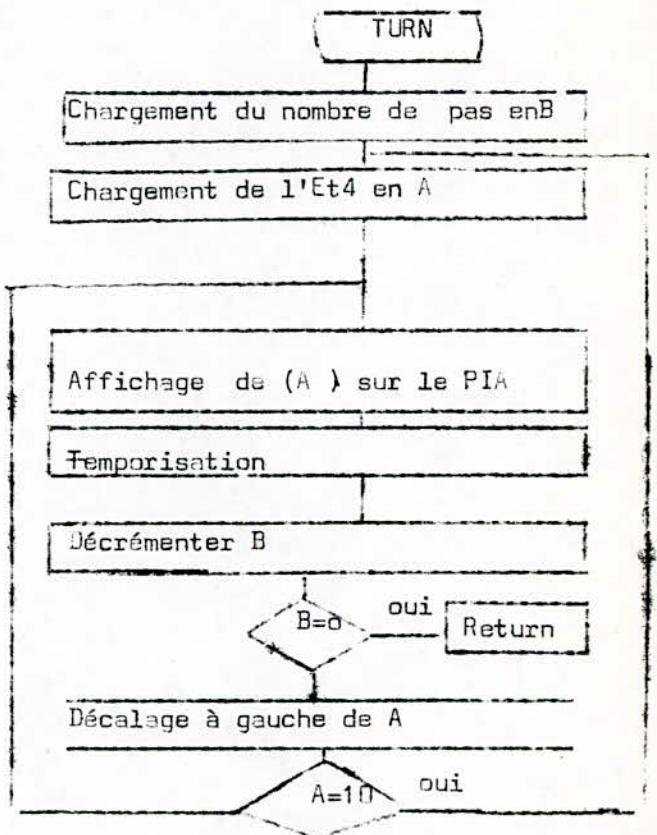
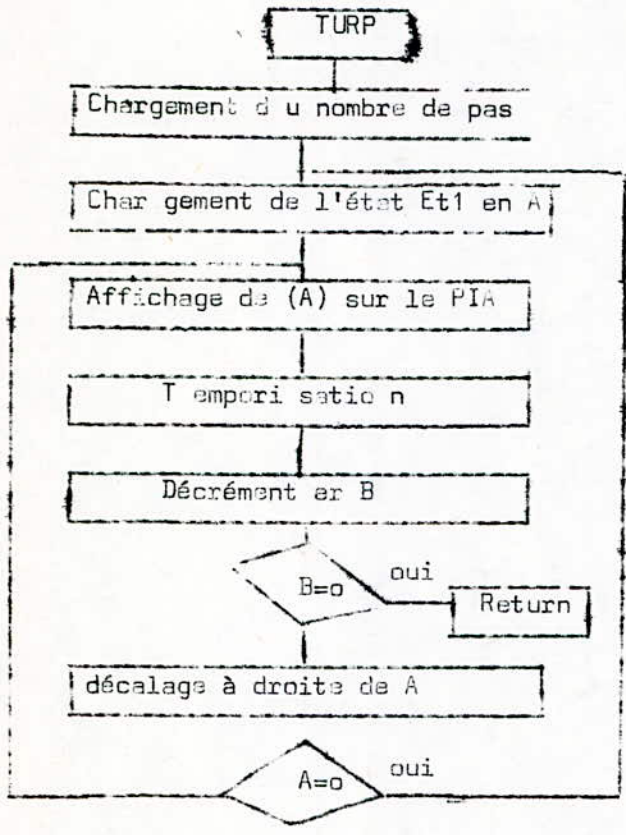
PROGRAMMES POUR LE FONCTIONNEMENT DU MOTEUR

Les états affichés successivement sur le PIA sont représentés dans le tableau suivant; 0,1,2,3,4...,7 sont les bits du registre de sortie ORB nous utiliserons uniquement un demi octet pour exciter les phases du moteur.

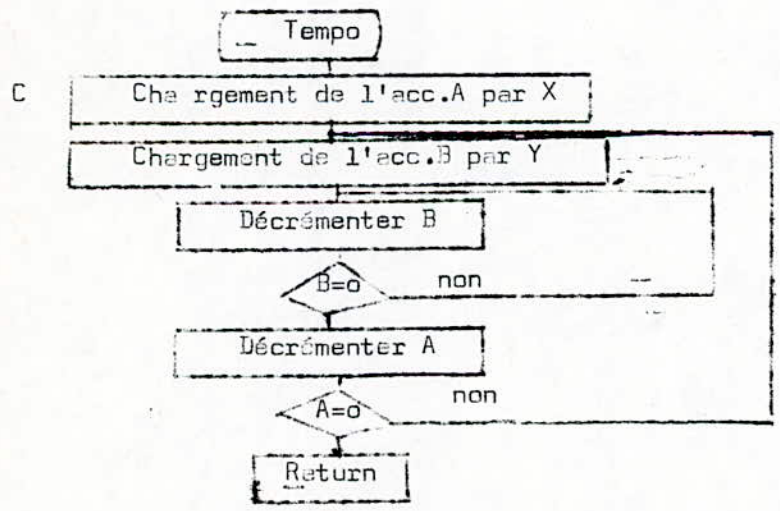
7	6	5	4	3	2	1	0	Etat	\$
1	0	0	0	1	0	0	0	Et1	88
0	1	0	0	0	1	0	0	Et 2	44
0	0	1	0	0	0	1	0	Et3	22
0	0	0	1	0	0	0	1	Et4	11

Nous constatons, d'après la table de vérité, que pour passer d'un état affiché sur le PIA à un autre, il suffit de faire un décalage circulaire, sur 8 bits, à droite ou à gauche selon le sens de rotation qu'on veut donner au moteur.

Nous remarquons que la rotation à gauche sur 8 bits s'obtient grâce à l'addition avec retenue; en effet, nous sommes obligés d'utiliser cette astuce de programmation car la rotation, dans le 6800, se fait sur 9 bits le bit de retenue C étant pris en compte; cette méthode est bien développée dans le mode d'excitation double phase. Ici, dans ce mode, nous employons seulement le décalage et nous chargerons à chaque fois, dans l'accumulateur l'état initial.



Organigrammes pour le mode d'excitation simple phase.



Organigramme de temporisation.

Les organigrammes correspondants aux programmes qui font tourner le moteur dans les deux sens sont donnés dans la figure suivante

-A. Sous programme TURP

C'est le programme qui fait tourner le moteur dans le sens positif pour un nombre de pas donné . Ce nombre de pas est contenu dans la case mémoire NPAS.

-Programme

TURP	L D A B	NPAS	Chargement du nombre de pas en B
DEB1	L D A A	+ \$ 88	Chargement de Et1 en A
	NOB		
DEB2	S T A A	O R B	Affichage de l'état sur le PIA
	S T A A	E T A T	
	S T A A	N P A S	
	J S R	TEMPO	temporisation
	L D A A	E T A T	
	L D A B	N P A S	
	D E C B		
	B E Q	F I N	si le nombre de pas est nul, fin.
	L S R A		décalage à droite de A
	B E Q	D E B 1	si A=0 branchement à DEB1
	J M P	D E B 2	sinon aller à DEB2
FIN	R T S		

-Calcul de la fréquence du moteur

Le temps nécessaire pour la réalisation d'un pas est égal au temps mis pour l'exécution de l'ensemble des instructions du sous programme.

Nous ajoutons des instructions NOP pour égaliser le temps entre les pas : $t = (55 + X(6Y + 8))$ microsecondes

$$f = 1 / t \text{ MHz}$$

Diagrammes:

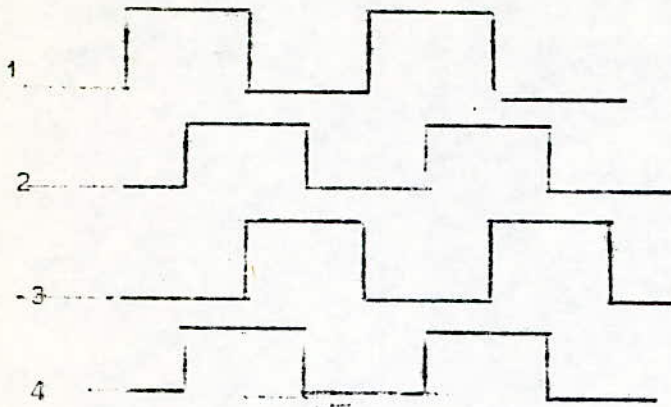


Table de vérité:

tps	1	2	3	4	Etat	\$
a	1	1	0	0	Et1	C
b	0	1	1	0	Et2	6
c	0	0	1	1	Et 3	3
d	1	0	0	1	Ft4	9

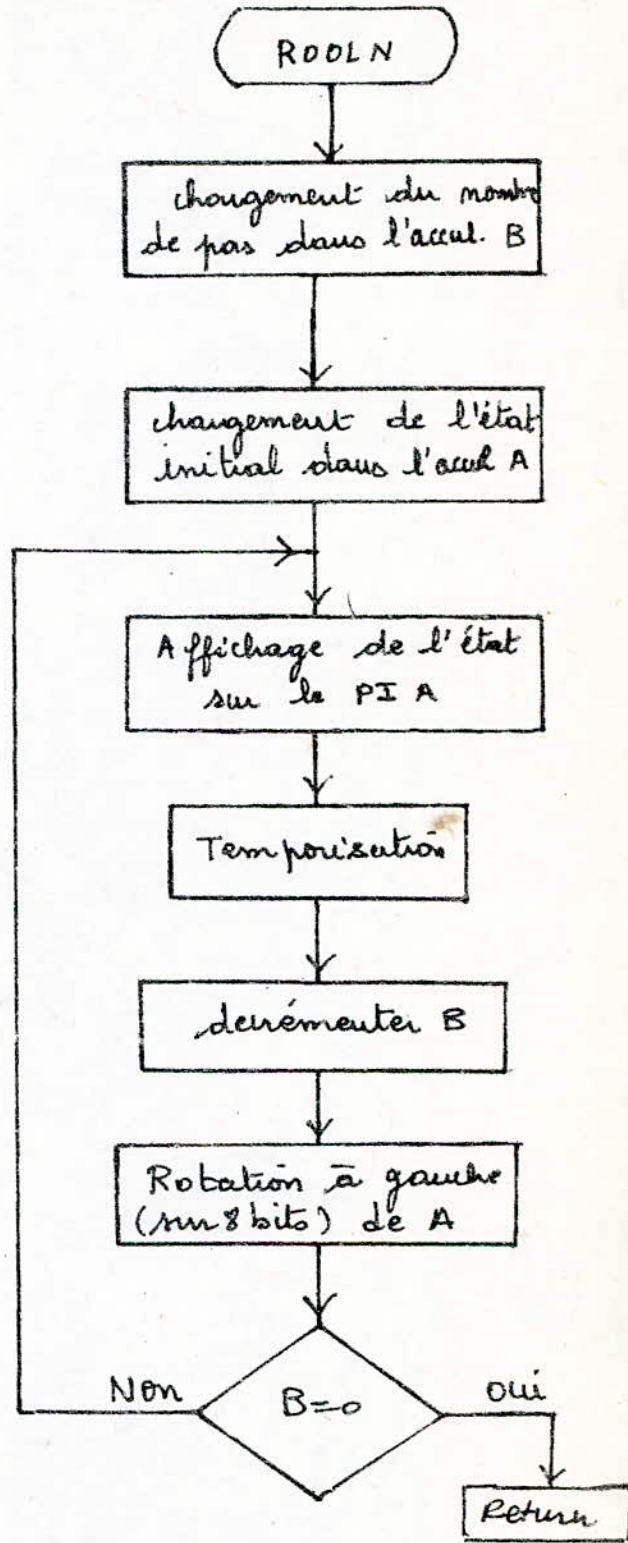
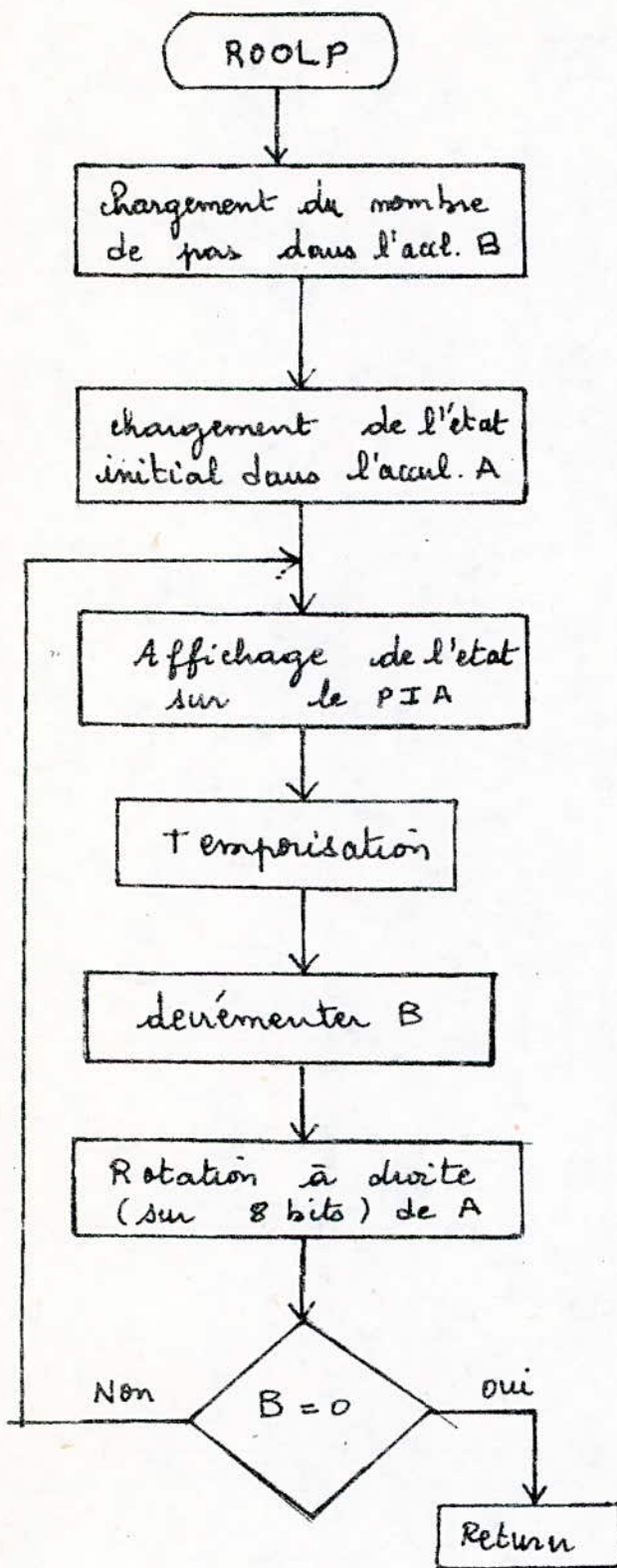
C'est le même principe que pour le mode d'excitation simple phase, aussi nous adoptons les mêmes conventions pour le sens de déplacement du moteur.

PROGRAMMES POUR LE FONCTIONNEMENT DU MOTEUR:

Les états que nous afficherons dans le registre de sortie du PIA sont donnés dans la table suivante:

7	6	5	4	3	2	1	0	Etat	\$
1	1	0	0	1	1	0	0	Et1	CC
0	1	1	1	1	1	1	1	Et2	66
0	0	1	1	0	0	1	1	Et3	33
1	0	0	1	1	0	0	1	Et4	99

Là aussi, nous remarquons, que passer d'un état à un autre nous devons exécuter un décalage circulaire sur 8 bits. La rotation se fera à gauche ou à droite selon que l'on veut faire tourner le moteur dans un sens ou dans l'autre. Pour réaliser ceci, nous utilisons des astuces de programmation qui seront développées dans ce qui suit.



ORGANIGRAMMES. DOUBLE PHASE

-B. Sous programme TURN

C'est un programme qui fait tourner le moteur, dans le sens négatif, avec un nombre de pas donné.

-Programme

TURN	LD AB	NPAS	chargement du nombre de pas en B
DEB1	LDAA	NPAS	chargement de ET4 en A
	NOB		
DEB2	STAA	ORB	affichage de l'état sur le PIA
	STAA	ETAT	sauvegarde de l'état et du nombre de pas
DEB2	STAB	NPAS	
JSR	JSR	TEMPO	temporisation
	LDAA	ETAT	
	LDAB	NPAS	
	DECB		décrémenter B
	BEQ	FIN	si B=0 fin de sous programme
	ASLA		décalage à gauche de A
	CMPA	NPAS	
	BEQ	DEB1	si A=10 branchement à DEB1
	JMP	DEB2	sinon aller à DEB2
FIN	RTS		

-Calcul de la fréquence

Le temps mis pour l'exécution d'un pas est:

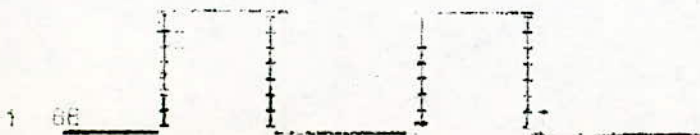
$$t = (57 + X(6Y + 8)) \text{ microsecondes}$$

$$f = 1 / t \text{ MHz}$$

II-1.2. Mode d'excitation double phase

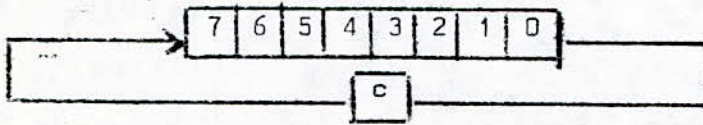
Dans ce mode deux bobinages sont alimentés à la fois; il suit l'ordre 1 et 2; 2 et 3; 3 et 4; 4 et 1.....etc

Diagramme



Sous programme ROOLP

Ce programme permet au moteur de tourner dans le sens positif, avec un nombre de pas fixé. L'instruction de rotation à droite ROR s'exécute, dans l'accumulateur de la manière suivante:



Cette rotation se fait sur 9 bits ; or nous avons besoin de faire un décalage circulaire sur 8 bits ; nous ne pouvons donc pas employer cette instruction .

Nous remarquons que lorsque l'on applique deux fois ADC (l'addition avec retenue) à un état se trouvant dans notre table de vérité nous obtenons l'état suivant . Nous réalisons ainsi un décalage circulaire à droite sur 8 bits avec deux instructions ADC .

Programme:

ROOLP	LDAB	NPAS	charge du nombre de pas en B
	LDAA	CC	chargement d'un état (CC) en A
DEBDEB	STAA	DRBT	affichage de l'état sur le PIA
	STAA	ETAT	
	STAB	NPAS	
	JSR	TEMPO	temporisation
	LDAA	ETAT	
	ADCA	ETAT	
	ADCA	ETAT	
	LDAB	NPAS	
	DECB		
	BNE	DEB	
	RTS		

Calcul de fréquence

Pour exécuter un pas le programme ROOLP met un temps

$$t = (49 + X (6Y + 8)) \text{ microsecondes}$$

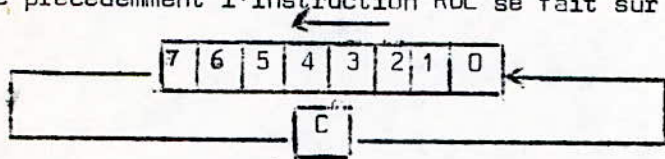
$$f = 1 / t \text{ MHz}$$

-B. Sous programme ROOLN

Ce programme permet au moteur de tourner dans le sens négatif avec un nombre de pas donné dans la case mémoire NPAS.

Nous devons faire cette fois ci une rotation à gauche sur 8 bits.

Comme précédemment l'instruction ROL se fait sur 9 Bits :



L'addition d'un état avec lui même , avec retenue , satisfait au décalage circulaire sur 8 bits .

Programme :

ROOLN	LDAB	NPAS	chargement du nombre de pas
	LDA	CC	
DEB	STAA	ORB	affichage de l'état
	STAB	NPAS	
	STAA	ETAT	
	JSR	TEMPO	temporisation
	LDA	ETAT	
	ADCA	ETAT	décalage circulaire sur 8bits de A
	LDAB	NPAS	
	DECB		
	BNE	DEB	si B ≠ 0 ;branchement à DEB
	RTS		

Calcul de la fréquence :

LE temps mis pour réaliser un pas est :

$$t = 47 + X(6Y+8) \text{ microsecondes}$$

$$f = 1 / t \text{ MHz}$$

II.1-3: . SEQUENCE DEMI PAS

Elle permet d'avancer d'un angle moitié du pas normal; elle suit l'ordre
1 et 2 , 2 , 2 et 3, 3, 3 et 4 , 4 , 4 et 1, ...

diagramme:

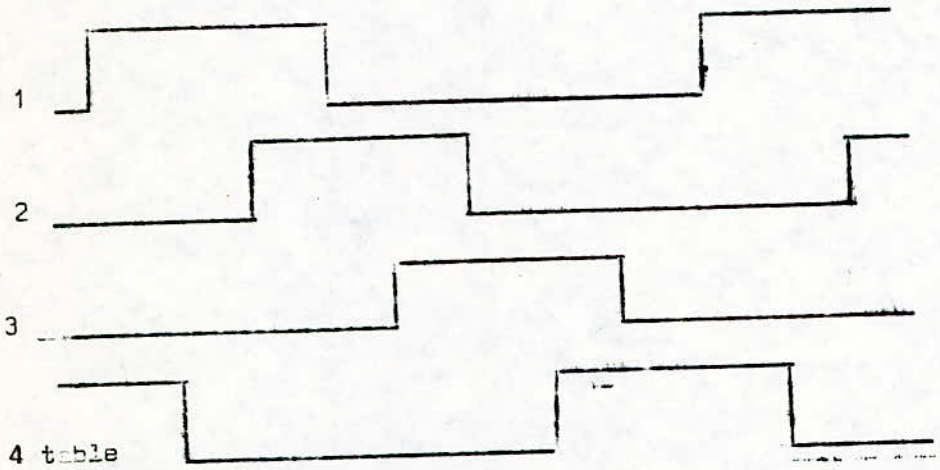


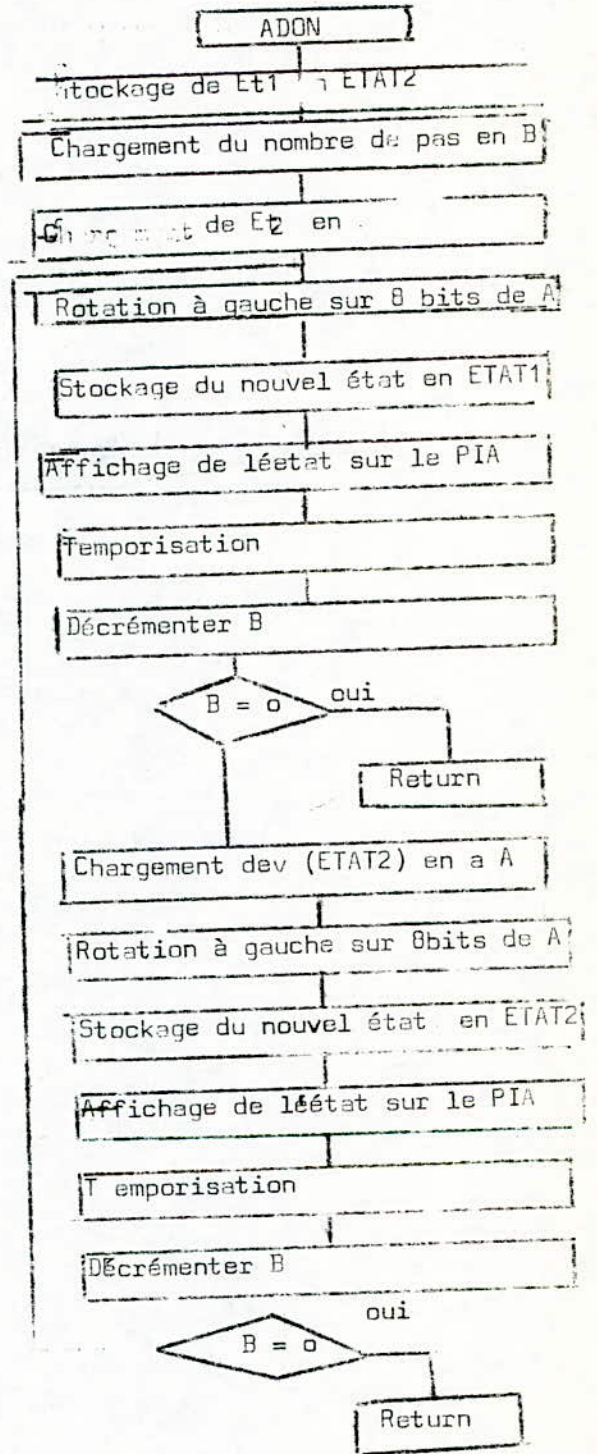
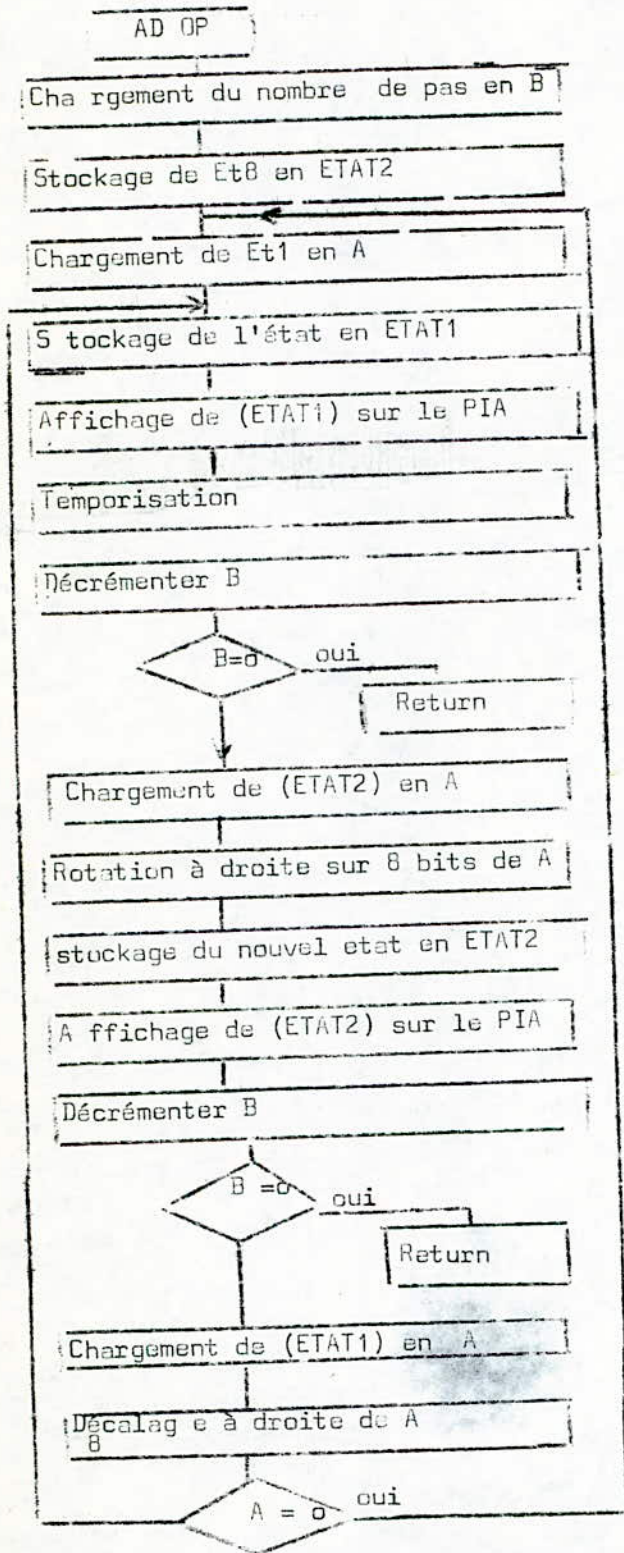
Table de vérité :

tps	1	2	3	4	ETAT	\$
a	1	0	0	0	ET1	8
b	1	1	0	0	ET2	C
c	0	1	0	0	ET3	4
d	0	1	1	0	ET4	6
e	0	0	1	0	ET5	2
f	0	0	1	1	ET6	3
g	0	0	0	1	ET7	1
h	1	0	0	1	ET8	9

Programme pour le fonctionnement du moteur :

Nous adoptons les mêmes conventions que pour les autres modes d'excitations
en ce qui concerne le sens de rotation du moteur, nous affichons sur le PIA
les 8 mots de 8bits suivants:

7	6	5	4	3	2	1	0	ETAT	\$
1	0	0	0	1	0	0	0	ET1	88
1	1	0	0	1	1	0	0	ET2	CC
0	1	0	0	0	1	0	0	ET3	44
0	1	1	0	0	1	1	0	ET4	66
0	0	1	0	0	0	0	0	ET5	22
0	0	1	1	0	0	1	1	ET6	33
0	0	0	1	0	0	0	1	ET7	11
1	0	0	1	1	0	0	1	ET8	99



Organigrammes pour la séquence demi-pas.

Nous constatons d'après ce tableau, que le mode d'excitation demi-pas est la combinaison des autres modes simple phase et double phase.

-A. Sous programme ADOP

Programme qui fait tourner le moteur dans le sens positif avec un nombre de pas donné, l'organigramme de ce programme est donné par la figure suivante:

ADOP	LDAB	NPAS	chargement du nombre de pas en B
	LDAA	NPAS	chargement de l'ETB dans A
	STAA	ETAT2	stockage de l'état dans ET2
DEB	LDAA	NPAS	chargement de l'état ET1 en A
	NOP		
SUIV	STAA	ETAT1	stockage de l'état dans la case ETAT1
	STAA	ORB	affichage de l'état
	STAA	NPAS	
	JSR	TEMPO	temporisation
	LDAB	NPAS	charger B du nombre de pas
	DECB		décrémenter B
	BEQ	FIN	si le nombre de pas est nul fin du s/prog
	LDAA	ETAT2	mettre le contenu de ETAT2 en A
	ADCA	ETAT2	décalage circulaire à droite sur 8 bits DE A
	ADCA	ETAT2	
	STAA	ETAT2	stockage du nouvel état en ETAT2
	STAB	NPAS	
	STAA	ORB	affichage de l'état
	JSR	TEMPO	temporisation
	NOP		
ADOP	LDAB	NPAS	chargement du nombre de pas en B
	DECB		décrémenter B
	BEQ	FIN	Si B=0 fin de programme
DEB	LDAA	ETAT1	chargement de l'état ET1 en A
	LSRA		décalage à droite de A
SUIV	BEQ	DEB	si A=0 branchement à DEB
	JMP	SUIV	sinon aller à suiv
FIN	RTS		

Calcul de la fréquence

le temps mis pour l'exécution d'un pas est:

$$t = 54 + X (6Y + 8) \text{ microsecondes}$$

$$f = 1 / t \text{ MHz}$$

-B. Sous programme ADON

C'est le programme qui fait tourner le moteur dans le sens négatif avec un nombre de pas donné.

l'organigramme est donné dans la figure suivante:

ADON	LDA	A 88	
	STAA	ETAT2	
	LDA	A CC	
	STAA	ETAT1	
	LDAB	NPAS	
DEB	LDA	ETAT1	chargement de l'état dans l'accumulateur A
	ADCA	ETAT1	
	STAA	ORB	affichage de l'état dans ORB
	STAA	ETAT1	sauvegarde de l'état dans la case ETAT1
	STAB	NPAS	sauvegarde du nombre de pas dans NPAS
	JSR	TEMPO	temporisation
	LDAB	NPAS	
	DECB		
	BEQ	FIN	si nombre de pas est nul, fin du S/prog
	LDA	ETAT2	
	ADCA	ETAT2	décalage circulaire à gauche
	STAA	ORB	
	STAA	ETAT2	
	STAB	NPAS	
	JSR	TEMPO	
	LDAB	NPAS	
	DEC		
	BNE	DEB	
FIN	RTS		

Calcul de fréquence

le temps réalisé par ce programme pour exécuter un pas est:

$$t = 47 + X(6Y + 8) \text{ microsecondes}$$

$$f = 1 / t \text{ MHz}$$

CONCLUSION

Le mode d'excitation utilisé , pour le fonctionnement des moteurs pas à pas est celui du double phase. Ce mode présente des avantages:

- couples important
- faible dépassement (la phase que le rotor dépassera va introduire un freinage)
- continuité du flux

La programmation de mode d'excitation, vue au chapitre précédent, nécessite un registre de 8 bits d'où l'utilisation des 8 bits du PORT B.

Les séquences de commande , notamment le nombre de phases alimentées à la fois et les intervalles de commutation peuvent être facilement changés sans modifier l'étage de puissance .

Nous quitterons ce chapitre en remarquant qu'on peut théoriquement augmenter le nombre de pas à l'infini.

Dans le cas du moteur à aimant permanent à quatre bobines, nous avons considéré deux cas:

- 1- séquence un pas (simple ou double phase)
- 2- séquence demi-pas

Nous pouvons imaginer une 3ième séquence où nous appliquerons les tensions suivantes:

	bobinage	1	2	3	4
temps					
a		1	0	0	0
b		$3\frac{1}{2}$	$\frac{1}{2}$	0	0
c		$\frac{1}{2}$	$3\frac{1}{2}$	0	0
d		0	1	0	0

Une telle commutation donne trois pas pour un quart de tour , nous pouvons aussi réaliser des systèmes dix pas par quart de tour , ce qui supprimerait le reproche que l'on fait au moteur pas à pas, c'est à dire d'avoir une résolution limitée.

Détermination des fréquences limitant les domaines de fonctionnement du moteur:

Pour commander le moteur pour l'un de ses régimes de fonctionnement, il faut écrire des programmes permettant de régler la fréquence des impulsions de commande afin de modifier la vitesse du moteur;.

Avant de réaliser ces programmes, il est nécessaire de connaître les limites exactes des domaines de fonctionnement du moteur et particulièrement les fréquences correspondantes.

Détermination de la fréquence maximale de démarrage:

Nous écrivons un programme autour du sous programme TURP, pour déterminer la fréquence maximale de démarrage. Ce programme, appelé FMD, comprend trois parties:initialisation du PIA, initialisation de l'état d'excitation initial et l'exécution du sous programme TURP.

Le temps mis par ce programme pour la réalisation d'un pas est égal à $43+X(6Y+8)$ microsecondes.

Le moteur utilisé exécute un pas en un temps compris entre 2ms et 3ms . Cela nous donne une gamme de fréquences allant de 333Hz à 500Hz . La fréquence du moteur lorsqu'il tourne à l'aide de FMD, est :

$$f = 1 / (43 + X(6Y+8)) \quad 333\text{Hz} < f < 500\text{Hz}$$

Si nous nous fixons la valeur de Y à 10, X prend une valeur comprise entre 28 et 47.

Pour déterminer la fréquence max. de démarrage, nous ferons tourner le moteur à l'aide "de "FMD" à une fréquence voisine de 333Hz, si le moteur démarre bien on réexécute le même programme avec une fréquence plus élevée jusqu'à ce que le moteur ne suive plus la commande.

Quand on a atteint une fréquence à laquelle le moteur ne démarre plus, la fréquence précédente sera notée: fréquence maximale de démarrage.

Détermination de la fréquence maximale réversible

Nous avons vu au chapitre précédent l'existence d'une zone de survitesse que le moteur ne peut atteindre qu'avec une accélération progressive à partir d'un point situé au dessous de la fréquence maximale de démarrage La fréquence max. absolue qui donne la frontière de cette zone sera déterminée expérimentalement à l'aide du programme "FMA" conçu autour du sous programme TURP.

Ce programme comporte 4 parties : Initialisation, Initialisation de l'état d'excitation des phases, l'accélération et la vitesse maximale constante

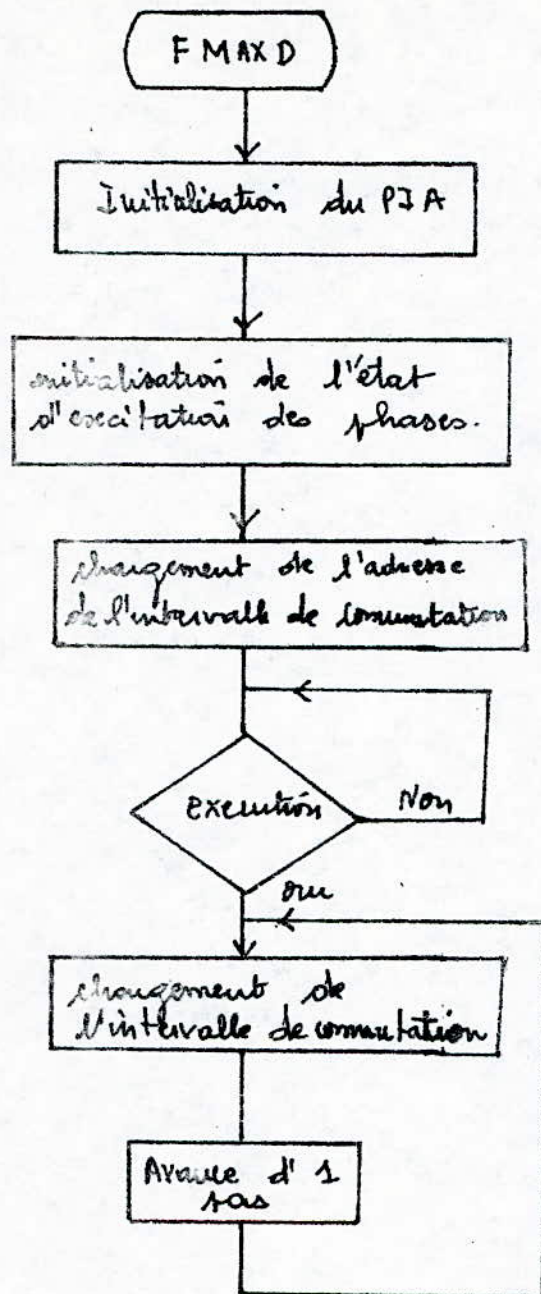
L'accélération constante est obtenue à l'aide d'un tableau dans lequel sont emmagasinés des intervalles de temps de commutations successifs.

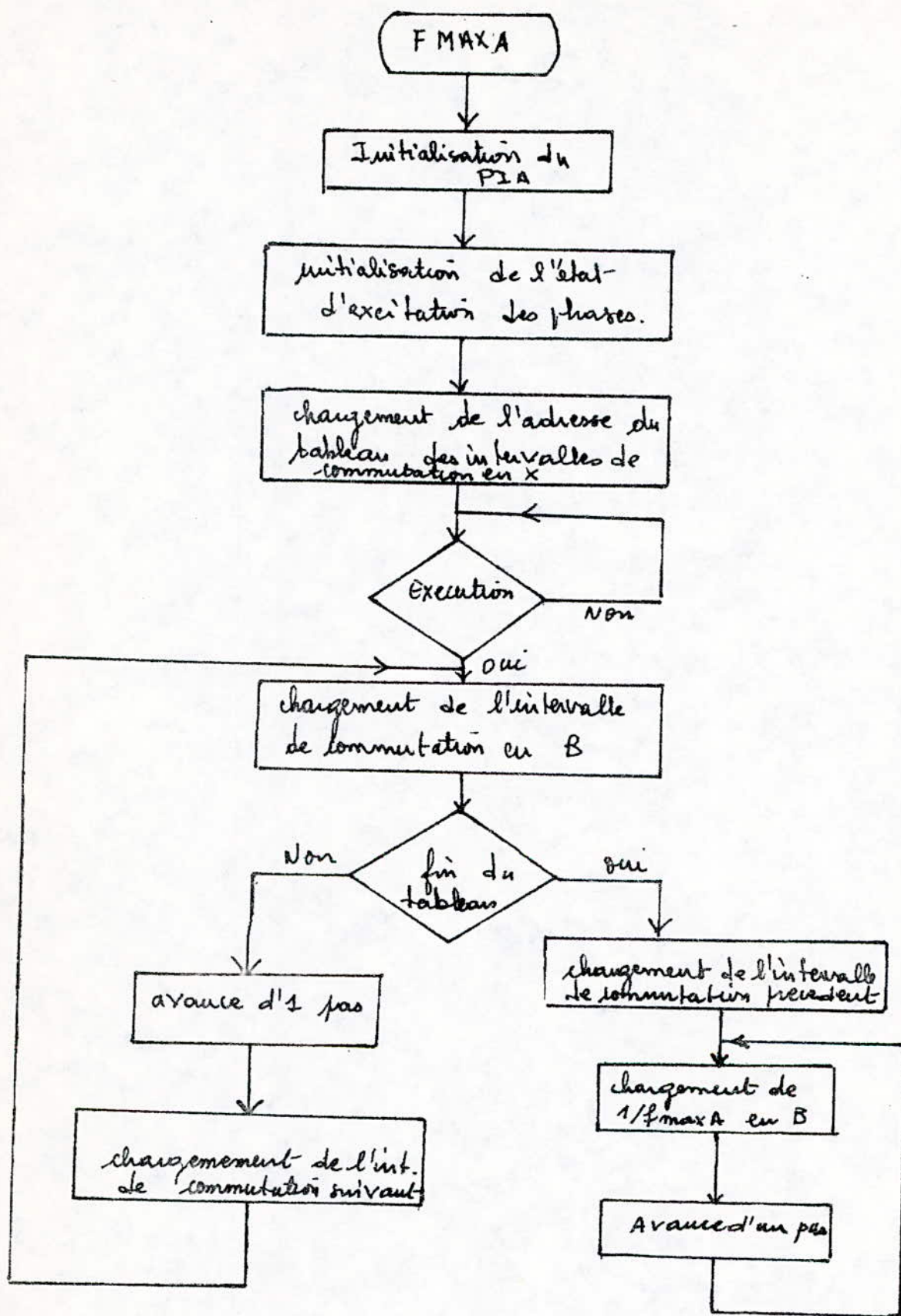
* la fin

Ce programme comporte 4 parties: Initialisation du PIA, initialisation de l'état d'excitation des phases, l'accélération et la vitesse maximale constante.

L'accélération constante est obtenue à l'aide d'un tableau dans lequel sont emmagasinés des intervalles de temps de commutations successives.

A la fin de ce tableau, on trouve un indicatif qui permet le passage au programme qui maintient la vitesse constante. Celle-ci correspond à la fréquence maximale absolue proposée. Si le moteur n'a pas perdu de pas, on recommence "T MaxA" avec une vitesse maximale absolue proposée supérieure.





ACCELERATION ET DECELERATION

ACCELERATION:

Pour assurer un transfert de position ou un temps minimum ou pour imposer une fréquence élevée déterminée, une phase transitoire d'accélération est indispensable.

L'accélération est assurée en imposant une rampe de fréquence en fonction du temps ou de la position.

DECELERATION:

Pour assurer un arrêt sur un seul pas une phase de décélération est nécessaire. Pour effectuer cette opération il est indispensable de viser un point précis. En effet, la décélération doit intervenir avec une marge de sécurité suffisante pour que le dernier pas soit parcouru à une fréquence interne au domaine 3 .fig. Cela peut se faire au moyen d'un compteur décompteur assurant la comparaison entre le nombre de pas parcourus et le nombre de pas à parcourir. Les meilleures séquences pour l'accélération et la décélération seront déterminées expérimentalement.

Il sera une table des valeurs de $B(X)$ dans la temporisation) qui seront emmagasinées en mémoire.

COMMANDE BANG-BANG

La montée et la descente en vitesse sont utilisées quand le nombre de pas à exécuter est important.

Dépendant si le nombre de pas demandé est petit et que le moteur ne peut atteindre sa vitesse maximale, il faut utiliser la commande bang-bang.

Pour une impulsion de commande, le rotor se met en marche progressive en oscillant autour de sa position d'équilibre. Ces oscillations sont défavorables au fonctionnement et à l'utilisation du moteur, il est nécessaire de les éliminer si l'on veut obtenir une avance rapide et des arrêts nets.

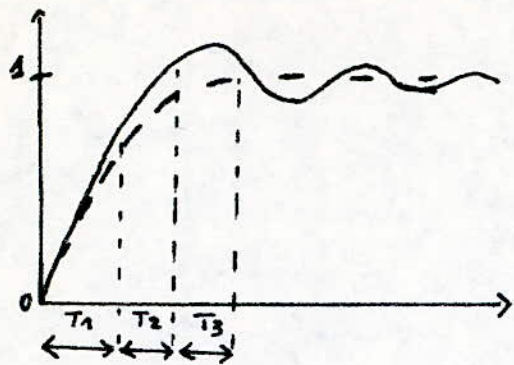
La commande bang-bang consiste à partager l'intervalle d'alimentation en trois temps:

premier temps: (accélération) la phase est alimentée pendant un temps T_1 nécessaire à la mise en rotation du moteur.

deuxième temps: (freinage) avant que le rotor n'atteigne sa position d'équilibre il sera freiné pendant le temps T_2 .

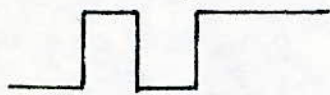
troisième temps: (établissement) le courant de la phase alimentée en premier temps est remis, ce qui maintient le nouvel état d'équilibre.

Cette commande peut être réalisée par un microprocesseur, en jouant sur le sens de rotation et sur les intervalles de commutation.



— : commande normale
 - - - : commande bang-bang

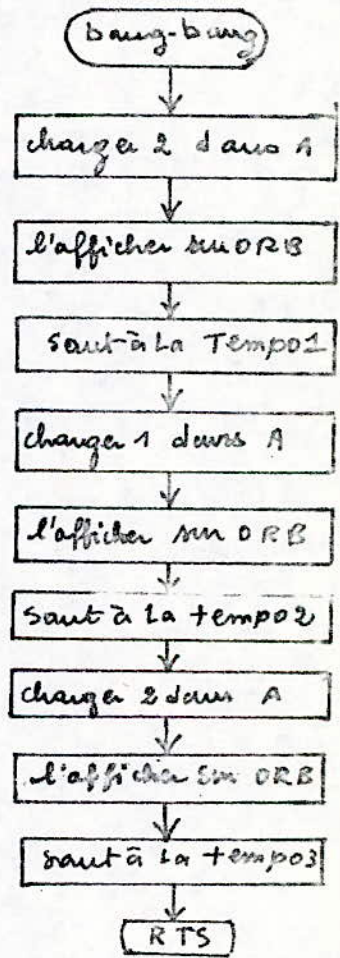
phase A



phase B



commande bang. bang.
et organigramme.



II-2. Programme général

II-2.1. Les paramètres d'étude

L'analyse de circuit imprimé définit les paramètres essentiels permettant la réalisation de celui-ci.

Un circuit imprimé est un ensemble de pastilles, qui représentent les points de connections, et de traits de liaisons qui peuvent être des segments horizontaux ou verticaux.

Notre programme général est basé sur la connaissance du nombre et des emplacements des :

- pastilles
- segments horizontaux
- segments verticaux

Chaque pastille est déterminée par ses coordonnées (X,Y). Chaque segment est déterminé par les coordonnées de son origine (X,Y) et par sa longueur L. Du fait que nous utilisons un moteur pas à pas pour le positionnement et le déplacement de l'aiguille du coordinatographe servant à réaliser le trace de circuit imprimé ; il est très important de convertir nos données en nombres de pas. Dans notre programme général, nous supposons que toutes les données sont exprimées en nombre de pas.

Ces données sont stockées en mémoire de façon à minimiser le temps nécessaire à la réalisation de circuit imprimé.

Le microprocesseur utilisé pour la commande du moteur est de huit bits. Par conséquent, le nombre de pas maximum que nous pouvons stocker en mémoire est 256 pas. Le moteur dont nous disposons fait 200 pas par tour, ce qui correspond à 2,54mm sur le film grattable ; Ceci demande, pour chaque piste à tracer, un nombre considérable de pas que nous devons ranger dans plusieurs cases mémoire.

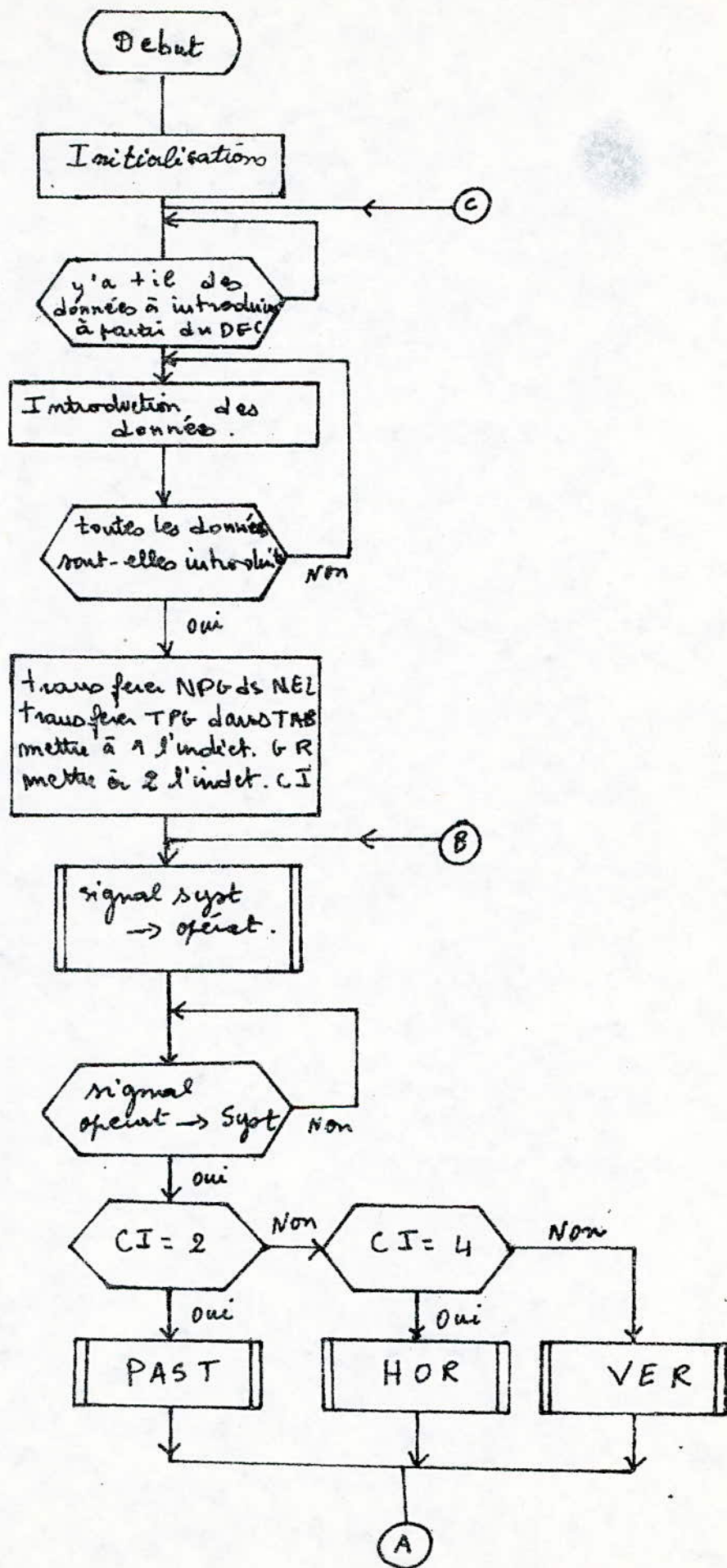
Pour éviter ce gaspillage d'espace mémoire, nous écrivons le nombre de pas considéré sous la forme suivante : $N_p = 256 \times A + B$; avec :

- N_p : nombre de pas à faire exécuter au moteur.

- A : quotient.

- B : reste.

A et B seront stockés dans deux cases mémoire successives. Cependant, par raison d'échelle, nous limitons la valeur maximale d'un octet à 128 au lieu de 256 ; $N_p = 128 \times A + B$; tq. A et B inférieurs à 128.



II.2.2. Description du programme

III Introduction

Le programme réalisant le tracé du circuit imprimé fait appel à trois sous programmes principaux:

- le sous programme réalisant les pastilles: PAST
- le sous programme réalisant les segments horizontaux: HOR
- le sous programme réalisant les segments verticaux: VER

Il fera aussi appel au sous programme signal système - opérateur

Indicateurs

En effet le coordinaatographe n'est pas complètement automatisé, l'utilisateur fera des interventions manuelles .

On utilise deux indicateurs , -GR: indique la grosseur du crayon et peut prendre deux valeurs, 1 pour le gros et 0 pour le fin

-CI: indique l'élément du circuit à réaliser et peut prendre trois valeurs 2 pour une pastille, 4 pour segments horizontaux, 8 pour segments verticaux, les signaux systèmes - opérateur se traduisent par l'allumage de voyants alimentés par le port A d'un PIA.

Chaque voyant correspond à un couple (CI, GR)

Interruptions

Quand le système a émis un signal, le programme entre en attente d'interruption , cette interruption sera envoyée à l'aide d'un bouton poussoir par l'opérateur.

Les lignes d'interruption utilisées sont:

-CA1 du premier PIA

-CA1 du second PIA

Ces lignes sont contrôlées par 2 bits : CRAD et CRA1 du registre de contrôle

CRAD : est utilisé pour autoriser le signal de demande d'interruption

IRQA au microprocesseur.

CRA1 : permet de choisir la transition active du signal d'entrée d'interruption.

Nous utilisons ces deux bits de façon à ce que le microprocesseur prenne en compte les interruptions.

CRAD = 1 autorisation du signal de demande d'interruption par le MPU

CRA1 = 0 transition active négative du signal d'interruption; IRQA

passse à l'état bas

passer à l'état bas quand l'indicateur CRA7 passe à l'état haut.

pour le premier PIA

CRA1 = 0

CRA0 = 1

pour le second PIA

CRA1 = 0

CRA0 = 1

Cette interruption sert à brancher le programme au sous programme de traitement de l'interruption dont l'adresse est contenue dans le vecteur FFF8 et FFF9.

II-3. Sous programmes pour la réalisation des pastilles et des segments

Introduction

Pour positionner une pastille, on utilise les coordonnées cartésiennes (X, Y). Quand au segment, on repère d'abord son origine puis on parcourt sa longueur.

La levée et la descente de l'aiguille du coordinatographe se fait par l'opérateur qui commande aussi la poursuite du programme.

A Réalisation des pastilles

Après la lecture des coordonnées, le programme positionne l'aiguille à l'endroit où la pastille va être grattée sur le film.

Le programme fonctionne de la façon suivante, à chaque réalisation on fait un teste pour voir si on a fait toutes les pastilles, le programme se branche alors à l'étiquette GROS.

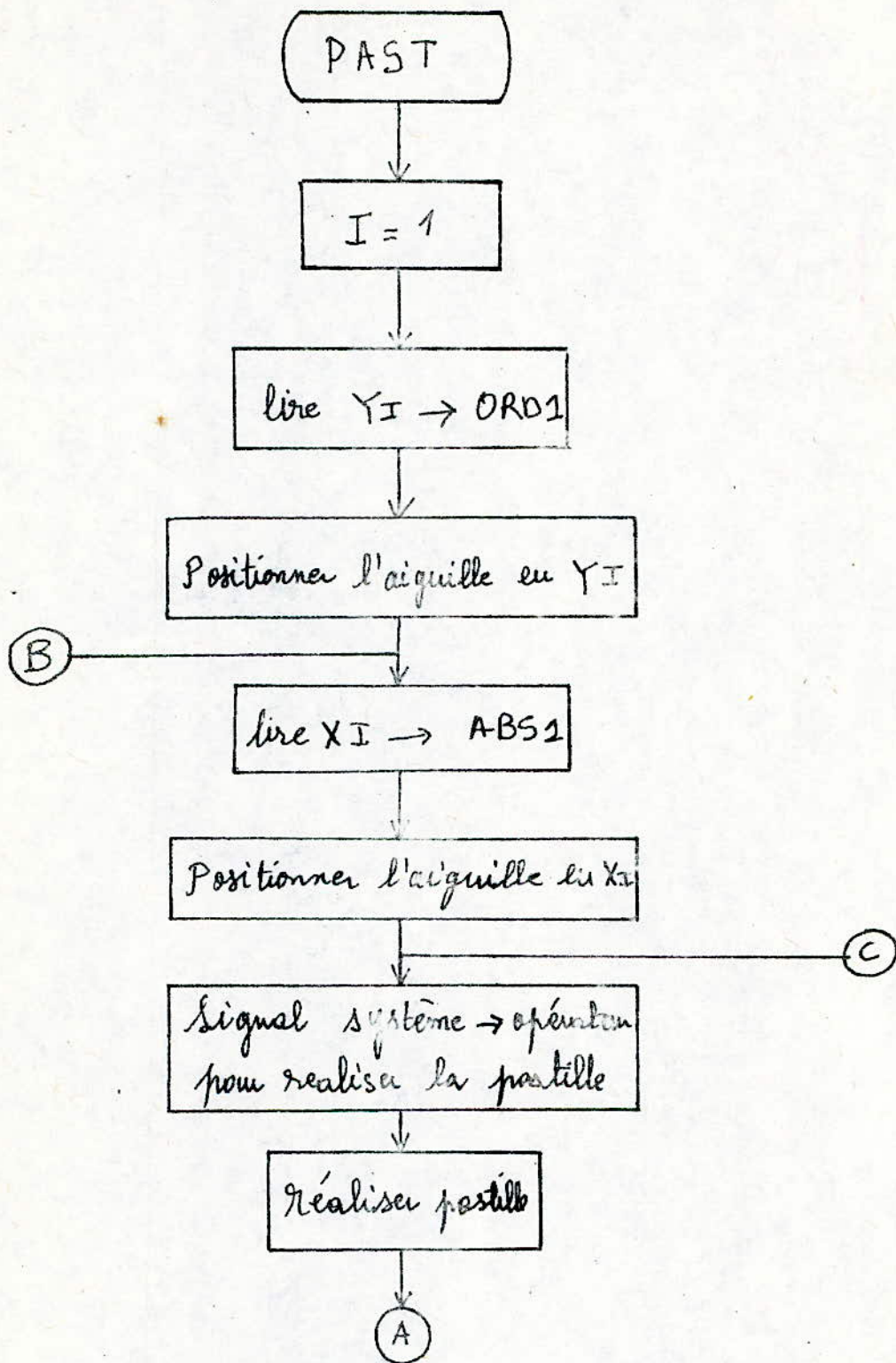
Les pastilles sont obtenues séquentiellement, chaque fois on compare la nouvelle ordonnée à l'ancienne:

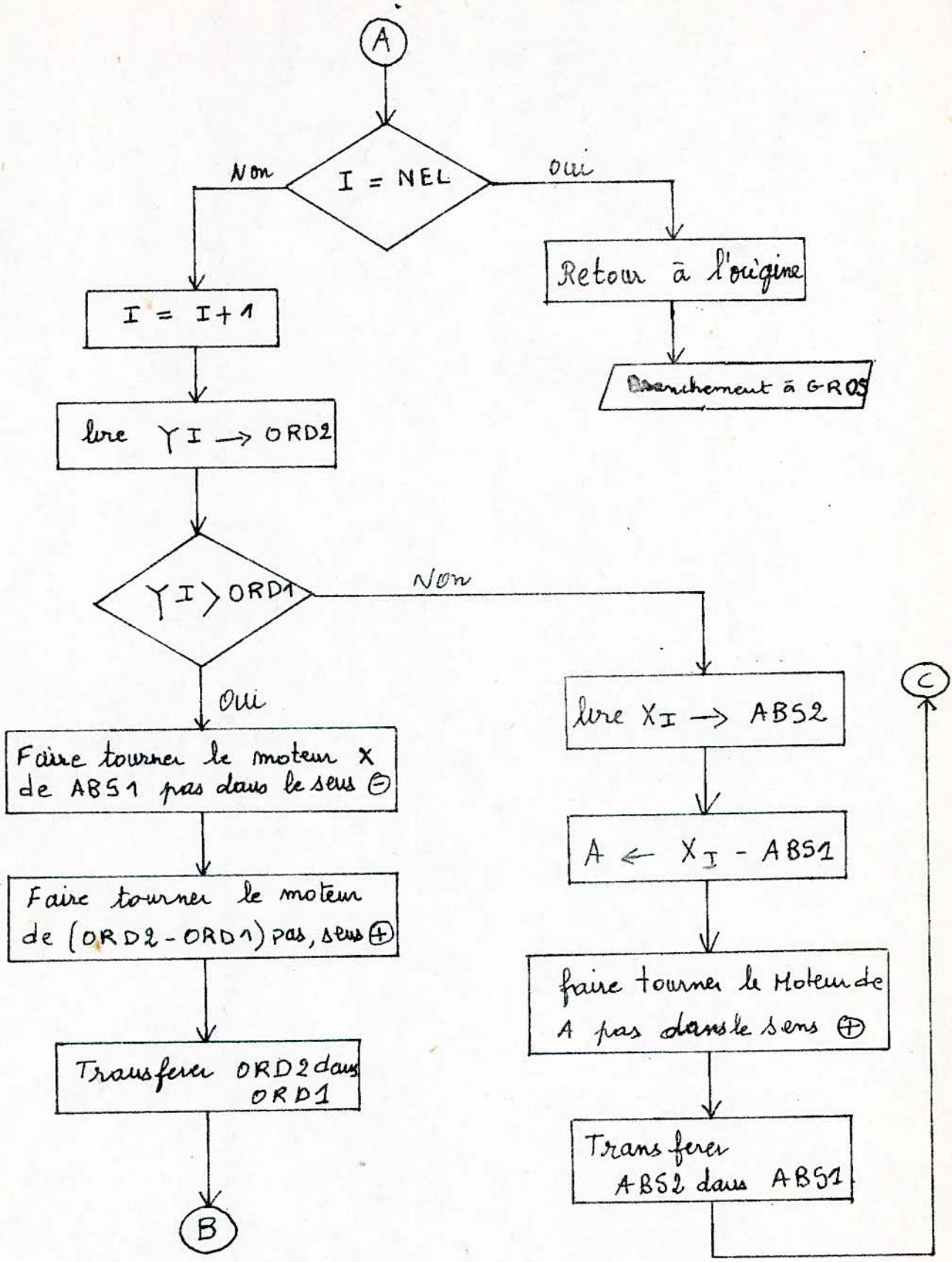
-Si les deux ordonnées sont égales on actionne le moteur 2 seul

-Si les deux ordonnées sont différentes on actionne les 2 moteurs

Dans le premier cas on se branchera à SUITE E, dans le deuxième cas on effectue un retour à la ligne et on se branchera à RETOUR X,Y

Réalisation des segments





Réalisation des segments

Après avoir lu les coordonnées de la première extrémité et la longueur du segment, le programme positionne l'origine de celui-ci à l'aide de l'aiguille du coordinatographe et commence le grattage après que l'opérateur ait abaissé l'aiguille sur le film.

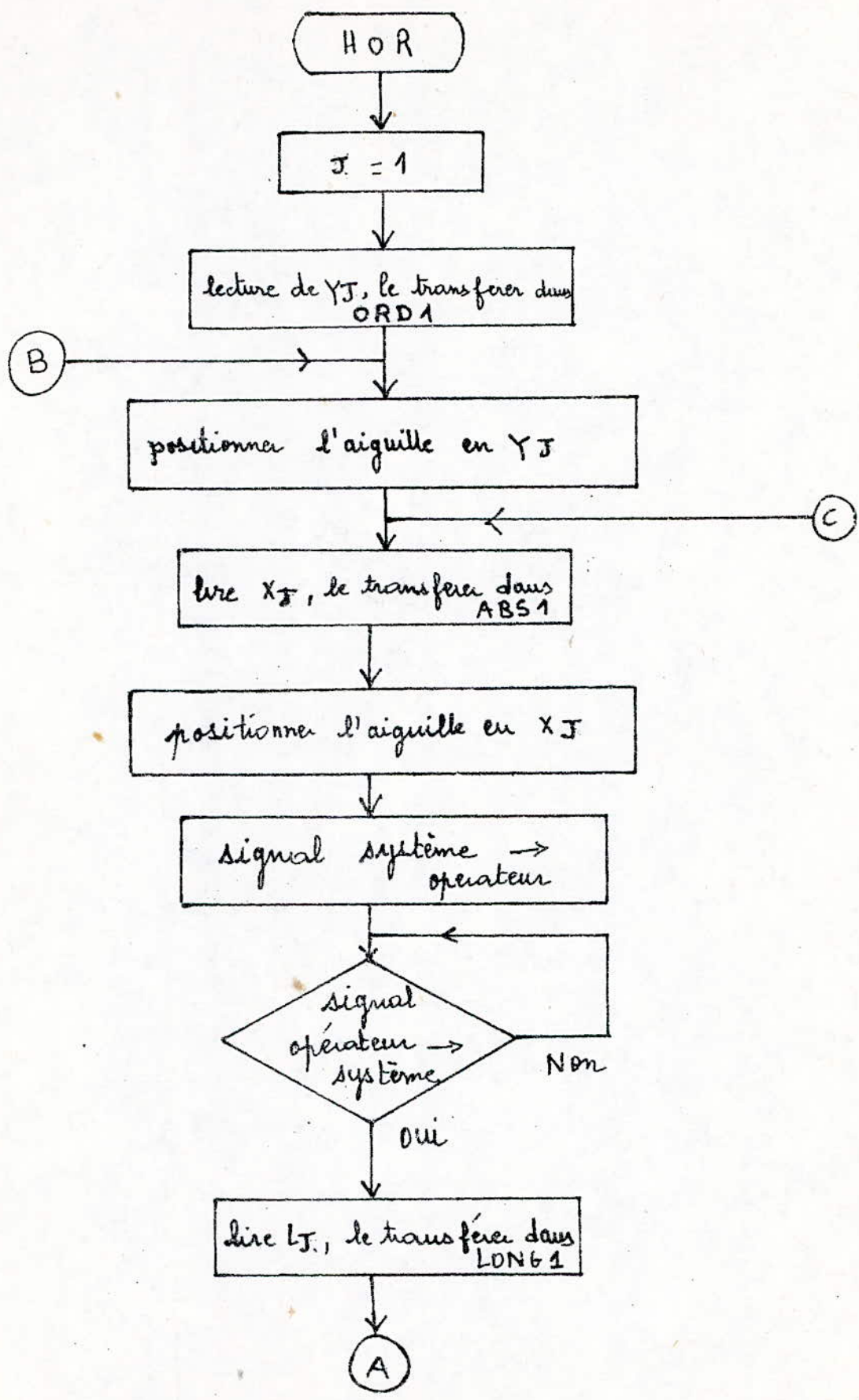
Une fois le tracé fait, un signal indique à l'opérateur qu'il doit lever l'aiguille et on passe au second segment.

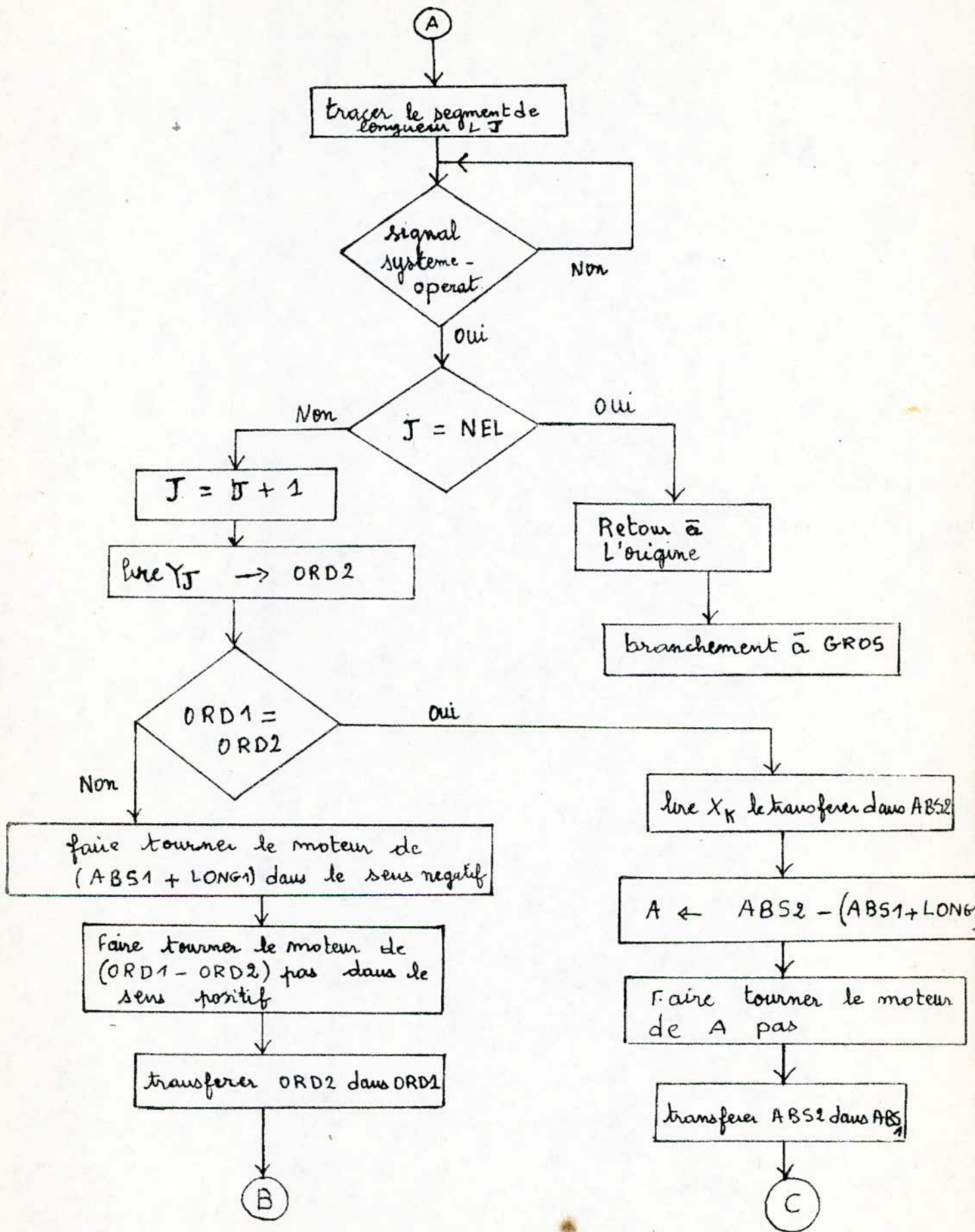
Dans le programme on teste si le nombre de segments voulus a été fait -si oui l'aiguille effectue un retour à l'origine, si non on passe au tracé du segment suivant, dans ce cas on compare les coordonnées du nouveau segment à celles de l'ancien:

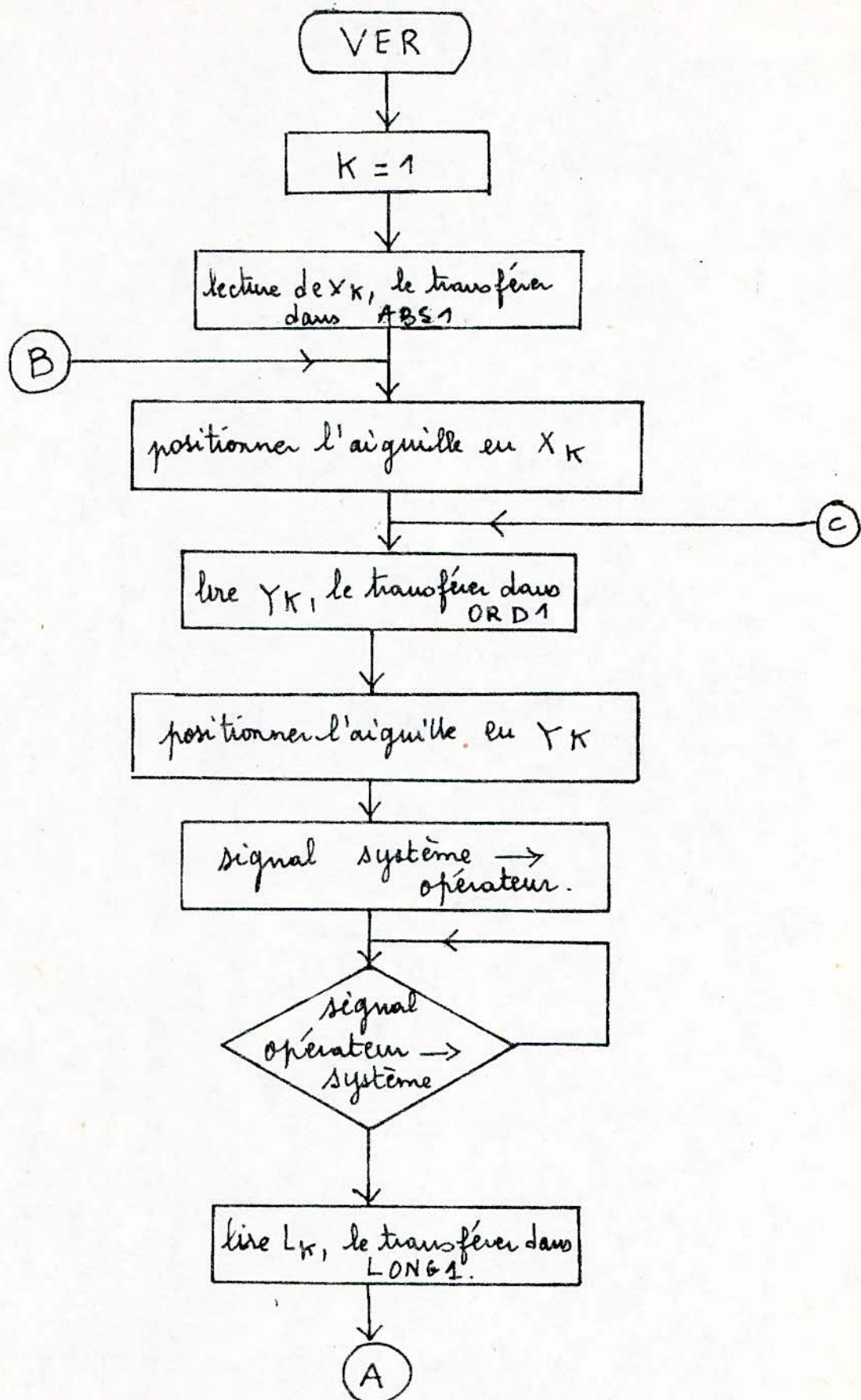
s'il y a égalité des abscisses on trace le segment suivant sur la même ligne

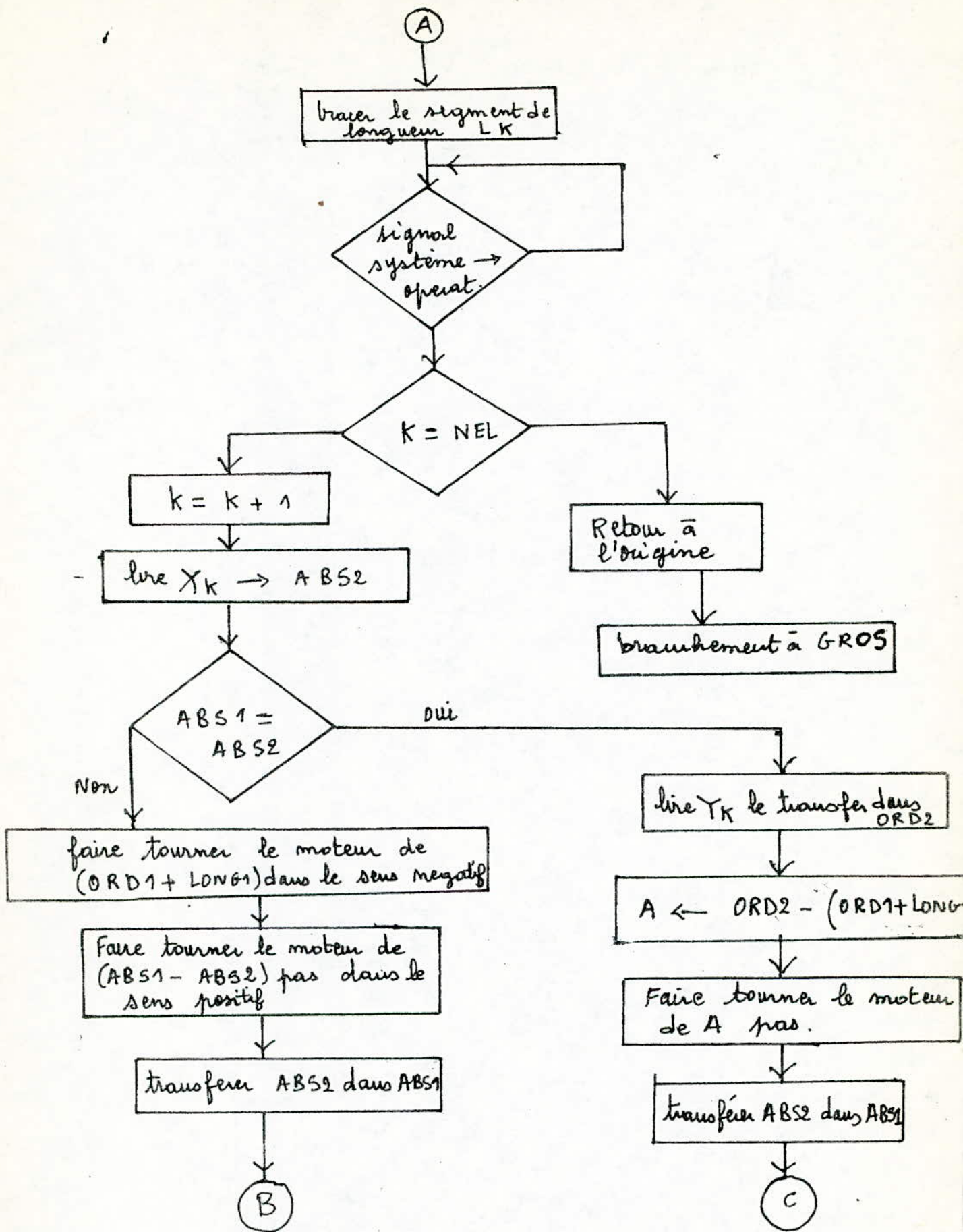
s'il y a égalité des ordonnées on trace le segment suivant sur la même colonne

s'il y a inégalité l'aiguille effectue un retour à la ligne









SIGNIFICATION DES SYMBOLES UTILISES DANS LE PROGRAMME

NPP nombre de petites pastilles

NPG nombre de grosses pastilles

NSHG "" " "" segments horizontaux gros

NSHF " " " segments " fins

NSVG " " " verticaux gros

NSVF " " " " fins

ADRESSES :

TPG grosses pastilles

TPP petites "

TSHG segments horizontaux gros

TSHF " " fins

TSVG " verticaux gros

TSVF " " fins

ETAT :contient le nombre représentant l'état

NPAS " " de pas

NEL " " d'éléments à réaliser

TAB " toujours l'adresse de la table des données

CI indique l'élément à réaliser

" "" la grosseur de l'élément à réaliser

CONCLUSION

Nous pensons que ce travail peut être vu comme un aboutissement d'un projet visant à lier un microordinateur à la réalisation d'un graphisme en général et ... particulier au tracé automatique d'un circuit imprimé à l'aide du coordinatographe .

Nous avons établi les programmes nécessaires à la commande de moteurs pas à pas ceux-ci peuvent être utilisés dans tout processus faisant intervenir ce genre de moteurs.

Par ailleurs, grâce à la commande automatique que nous avons mise au point nous montrons qu'il est possible d'utiliser les moteurs pas à pas pour le positionnement des éléments d'un circuit imprimé.

L'automatisation de cette technique permet une meilleure précision et un gain de temps appréciable par rapport aux méthodes manuelles.

Nous n' avons pu automatiser entièrement la méthode par manque de temps mais nous pensons que c'est tout à fait possible, il suffit pour ce faire d'ajouter un troisième moteur.

Nous utilisons deux moteurs pour positionner l'aiguille mais sa levée et sa descente sur le film reste manuelle

B I B L I O G R A P H I E

Les moteurs pas à pas. Jean Jacquin

Catalogue microprocesseurset mémoires. EFCIS

Programmation du 6800. Rodney Zaks

Les systemes à microprocesseurs. M. Aumiaux

Commande d'un moteur pas à pas par micropricesseur. Miamoti
These 3ième cycle ; Nancy .

Thèse d'ingénieur . Benmohamed et Fallia