



المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

Commande d'un heliostat par microcontrôleur

PROJET DE FIN D'ETUDES

Proposé par DR M HADDADI

étudié par

Nabil DJIDI

&

Sofiane HADJOU DJ

Soutenu en Juillet 1995

Dédicaces

je dédie ce modeste travail à mon père qui m'a beaucoup aidé et soutenu et à ma mère qui a toujours cru en moi, ainsi qu'à mes frères Karim et Rafik et à ma soeur Lynda à qui je souhaite beaucoup de bonheur.

Nabil

Je dédie ce modeste travail à mes chères parents qui ont toujours crus en moi et n'ont cessés de m'encourager, ainsi qu'à ma soeur Farida et mes frères Reda et Lotfi, sans oublier Nawel.

Sofiane

TABLE DES MATIERES

المدرسة الوطنية المتعددة التقنيات
المكتبة — BIBLIOTHEQUE
Ecole Nationale Polytechnique

1. DESCRIPTION DU SYSTEME	6
1.1 BLOC DE CALCUL	6
1.2 BLOC DE TRAITEMENT ET COMMANDE	6
1.2.1 CONVERSION DE LIGNE	8
1.2.2 TRAITEMENT ET COMMANDE	8
1.3 BLOC DE PUISSANCE	8
2. BLOC DE CALCUL	9
2.1 RAPPELS ASTROPHYSIQUES	9
2.2 FORMATAGE DE LA DONNEE	9
2.3 TRANSMISSION DE LA DONNEE	10
2.3.1 INTERFACE RS 232-C	11
3. BLOC DE TRAITEMENT ET DE COMMANDE	13
3.1 CONVERSION DE LIGNE	13
3.1.1 LE MC 1488	13
3.1.2 LE MC 1489	13
3.2 TRAITEMENT ET COMMANDE	14
3.2.1 PRESENTATION DU MICROCONTROLEUR 68705 P3S	14
3.2.2 PROGRAMMATION DU 68705 P3S	17
3.2.3 DESCRIPTION TECHNIQUE	17
4. BLOC DE PUISSANCE	20
4.1 DEFINITION DES MOTEURS PAS A PAS	20
4.2 CLASSIFICATION DES MOTEURS PAS A PAS	20
4.2.1 LE MOTEUR A AIMANT PERMANENT	20
4.3 CARACTERISTIQUE DES MOTEURS PAS A PAS	21
4.3.1 CARACTERISTIQUE STATIQUES	21
4.3.2 CARACTERISTIQUES DYNAMIQUES	21
4.3.3 ZONES DE FONCTIONNEMENT	21
4.4 CONCEPTION DE L'ETAGE DE PUISSANCE	22
4.5 FONCTIONNEMENT DE L'ETAGE DE PUISSANCE	22
4.6 CALCUL PRATIQUE	23
5. PROGRAMMATION DU PORT D'E/S DU MICRO-ORDINATEUR	25
5.1 LES FONCTIONS DOS	25
5.2 LES FONCTIONS BIOS	25
5.3 PROGRAMMATION DIRECTE DES PORTS D'E/S	27

5.3.1 PRESENTATION DE L'UART 8250	المدرسة الوطنية المتعددة التقنيات BIBLIOTHEQUE — المكتبة Ecole Nationale Polytechnique	27
6. PROGRAMME DE LA RECEPTION SERIE		33
6.1 GESTION DE LA COMMUNICATION		33
6.2 PROTOCOLE DE COMMUNICATION		36
6.3 EMISSION DE LA DONNEE		36
6.3.1 MISE EN FORME DE LA DONNEE		37
6.3.2 EMISSION DE LA DONNEE		37
6.4 RECEPTION DE LA DONNEE		38
7. PROGRAMME DE LA COMMANDE		40
7.1 EXTRACTION DE LA DONNEE		40
7.2 COMMANDE DU MOTEUR		42
7.3 ASSERVISSEMENT DE LA VITESSE DU MOTEUR		45
7.4 FIN DE POSITIONNEMENT		50
8. EVOLUTION DU SYSTEME		51
8.1 SELECTION DU PROTOCOLE		51
8.1.1 DETECTION DE LA VITESSE		53
8.1.2 LECTURE DU PROTOCOLE		57
8.2 EXTENSION DE LA COMMANDE A PLUSIEURS MOTEURS		58
8.3 FONCTIONNEMENT		59

LISTE DES FIGURES

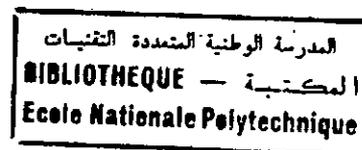


Figure 1-1 : Organisation globale du système	6
Figure 1-2 : Synoptique du bloc de traitement et de commande .	7
Figure 3-1 : Constitution interne des circuits de conversion de ligne .	14
Figure 3-2 : Synoptique de l'organisation interne du 68705 PAS .	15
Figure 3-3 : Organisation de la mémoire .	16
Figure 6-1 : Forme d'onde d'une donnée transmise au format asynchrone	33
Figure 6-2 : Organigramme de la gestion du flux par le microcontrôleur .	35
Figure 6-3 : Organigramme de la gestion du flux par le P.C.	36
Figure 6-4 : Organigramme de la réception série des données .	39
Figure 7-1 : Organigramme global du programme de commande .	40
Figure 7-2 : Organigramme de l'extraction de la donnée .	41
Figure 7-3 : Chronogramme du mode d'excitation monophasé .	42
Figure 7-4 : Organigramme de la commande du moteur	44
Figure 7-5 : Allure de la vitesse du moteur en fonction des pas effectués .	45
Figure 7-6 : Organigramme de calcul de la tempo durant la phase d'accélération .	47
Figure 7-7 : Organigramme de la temporisation .	48
Figure 7-8 : Organigramme de sélection d'ADR	49
Figure 7-9 : Organigramme de fin de communication	50
Figure 8-1 : Format de l'octet de reconnaissance .	51
Figure 8-2 : Organigramme du fonctionnement global de la détection du protocole .	52
Figure 8-3 : Organigramme de détection de la vitesse de transfert .	54
Figure 8-4 : Chronogrammes des temps réels et mesuré .	55
Figure 8-5 : Organigramme de lecture du protocole .	57
Figure 8-6 : Organigramme de détection d'erreur .	58
Figure 8-7 : Organisation d'un système à 8 moteurs .	59
Figure 8-8 : Organisation du programme du micro-ordinateur .	59

INTRODUCTION

المدرسة الوطنية المتعددة التقنيات
المكتبة — BIBLIOTHEQUE
Ecole Nationale Polytechnique

De nos jours, on se préoccupe avec un intérêt croissant, voire avec une certaine anxiété, de trouver de nouvelles sources d'énergie. En effet les besoins énergétiques de la planète devraient atteindre 20 à 25 milliards de kilowatts en l'an 2030[1].

Cependant, à cette date toutes les énergies actuellement employées en abondance par l'homme ne couvriront plus que la moitié de la consommation. Les réserves en ressources d'origine fossiles (gaz et pétrole) hydroélectrique ou nucléaire sont limitées et leur emploi constitue des menaces réelles pour l'environnement (le réchauffement de la planète, l'élimination des déchets radioactifs, accidents...)

La fusion thermonucléaire au formidable potentiel énergétique, quant à elle, n'est qu'à ses premiers pas, et son industrialisation ne saurait être envisagée avant plusieurs décennies et au prix d'un investissement colossal.

De ce fait, la recherche d'une énergie renouvelable propre et capable de satisfaire à des prix raisonnables les besoins de l'ensemble de la planète est aujourd'hui devenue une nécessité absolue.

Le rayonnement solaire semble à première vue constituer une réponse adéquate à ces préoccupations.

En effet, à l'heure actuelle, la consommation mondiale d'énergie, pendant une année équivaut à l'énergie solaire captée par la terre pendant quarante minutes, et les experts estiment que nos besoins seraient satisfaits par seulement 0,1% de la surface terrestre si elle était couverte de capteurs [1].

Pour profiter pleinement de cette manne, les scientifiques doivent encore résoudre nombre de problèmes, essentiellement liés au rendement relativement faible des cellules photovoltaïques (de l'ordre de 15%) et aux pertes de positionnement des panneaux.

C'est ainsi qu'adapter la position des panneaux à celle du soleil, devrait permettre un gain significatif de rendement. L'idée serait de faire tourner ces derniers de manière à ce qu'ils poursuivent le soleil dans sa trajectoire diurne.

Cependant, cette idée se trouve confrontée à un problème de rentabilité du système car en pratique cela pourrait mettre en jeu des puissances pouvant dépasser celles collectées par les panneaux.

Une alternative viserait à obtenir un compromis entre un système statique et un système de poursuite. Celle-ci serait de positionner un panneau suivant un angle optimal présentant un taux d'ensoleillement journalier maximal. Le calcul de cette position est effectué à partir d'équations cosmologiques qui prennent en compte la date et le lieu d'installation du panneau.

Le travail qui nous a été confié consiste en l'étude et la réalisation d'un système basé sur le procédé ainsi décrit, appliqué à une centrale formée d'un champ de panneaux commandés à partir d'un ordinateur central.

1. DESCRIPTION DU SYSTEME

Nous nous proposons donc d'étudier un système de commande d'une centrale génératrice d'électricité formée d'un champ de panneaux photovoltaïques devant être orientés quotidiennement suivant un angle permettant une captation optimale.

Pour ce faire, le calcul de cet angle est basé sur des lois cosmologiques qui seront implémentées sur un ordinateur. Les résultats seront ensuite transmis sous forme de données par liaison série à chaque panneau.

Le traitement et l'exploitation des données transmises se feront grâce à un système conçu autour d'un microcontrôleur dont le rôle sera de positionner le panneau suivant l'angle calculé.

Le synoptique de la figure (1-1) donne l'organisation globale du système.

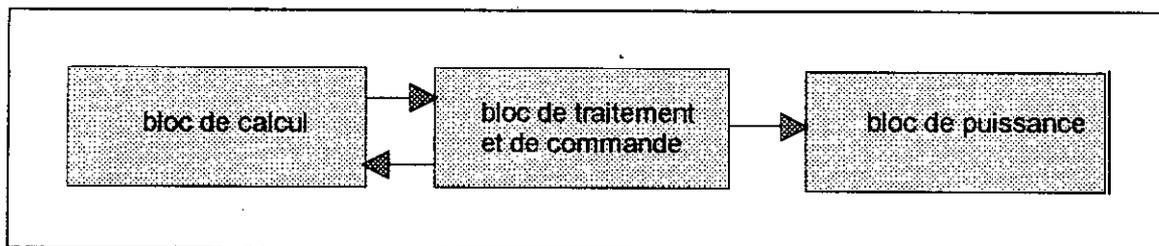


Figure 1-1 Organisation globale du système

1.1 BLOC DE CALCUL

Ce bloc est un ordinateur chargé d'effectuer les calculs de la position optimale à partir d'un programme écrit en langage évolué en tenant compte du lieu d'installation et de la date. Il a pour charge ensuite de mettre la donnée sous un format exploitable par le bloc de traitement et de commande, et enfin de gérer la communication à travers son interface série.

1.2 BLOC DE TRAITEMENT ET COMMANDE

Ce bloc est organisé suivant le synoptique de la figure (1-2)

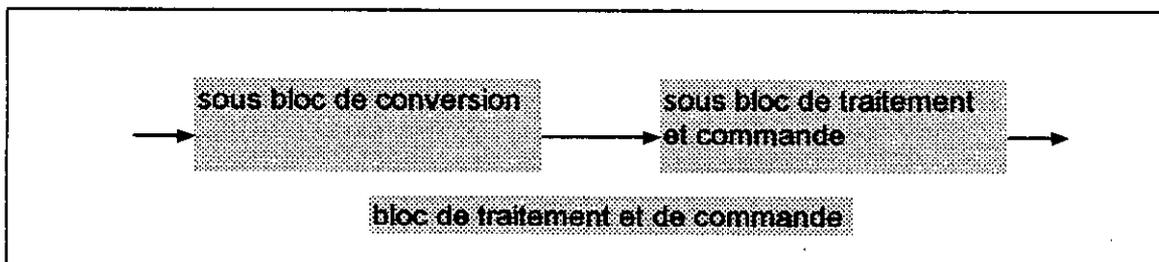


Figure 1-2 Synoptique du bloc de traitement et de commande.

1.2.1 CONVERSION DE LIGNE

La donnée émanant de l'ordinateur étant sous un format série respectant la norme RS 232-C, il est impératif de procéder à un changement de celle-ci en norme TTL exploitable par la circuiterie constituant le système.

1.2.2 TRAITEMENT ET COMMANDE

Constitué du microcontrôleur 68705 P3S de MOTOROLA, ce sous bloc est en fait le coeur du système. En effet, il assure

- ♦ La gestion de la communication avec le bloc de calcul,
- ♦ La conversion de la donnée série au format parallèle pour permettre son exploitation,
- ♦ Le traitement de la donnée,
- ♦ La commande du bloc de puissance.

1.3 BLOC DE PUISSANCE

Généralement, les courants issus du microcontrôleur et disponibles sur ses ports de sortie ne sont pas assez intenses pour commander un moteur. Pour y remédier, on insère un étage de puissance capable de piloter ce dernier.

Le moteur utilisé dans notre réalisation est un moteur pas à pas à aimant permanent de conception relativement simple et robuste.

2. BLOC DE CALCUL

En premier lieu, nous nous attacherons à rappeler des notions de base qui contribueront pour une grande partie à la compréhension des calculs astronomiques permettant la détermination de l'angle optimal de positionnement du panneau solaire.

2.1 RAPPELS ASTROPHYSIQUES

MOUVEMENT APPARENT DU SOLEIL

Le centre de gravité de la terre décrit une ellipse dont le soleil occupe l'un des foyers. Le plan de cette trajectoire s'appelle l'écliptique.

L'angle formé par la direction du soleil et le plan équatorial constitue la déclinaison δ . Celle-ci varie au cours de l'année pour atteindre ces deux valeurs extrêmes $-23^{\circ},27'$ au solstice d'hiver $+23^{\circ} 27'$ au solstice d'été et s'annule aux équinoxes de printemps et d'automne.

On calcule sa valeur, à chaque jour de l'année, par la formule approximative suivante [1] :

$$\delta = 23,5 \sin(2\pi d/365)$$

δ est exprimé en degrés, d est le temps en jours depuis l'équinoxe de printemps, $2\pi d/365$ représente des radians. δ est positif au printemps et en été, négatif en automne et en hiver.

Autre formule utilisée [1] :

$$\sin \delta = 0,4 \sin 0,986 (j - 82)$$

où j est le quantième de l'année depuis le premier janvier, δ s'exprime en degrés ainsi que $0,986 (j - 82)$.

Ces formules sont approchées elles ne suffisent que si l'on ne désire pas une précision meilleure que le degré.

Le développement en série de Fourier limité à l'harmonique 3 fournit une précision de $12''$ d'arc[1]

$$\delta = 0,302 - 22,93 \cos \omega'd - 0,229 \cos 2\omega'd - 0,243 \cos 3\omega'd + 3,851 \sin \omega'd + 0,002 \sin 2\omega'd - 0,055 \sin 3\omega'd$$

où d est le numéro du jour de l'année, avec $\omega' = 2\pi\omega / 366$ exprimé en radians.

2.2 FORMATAGE DE LA DONNEE

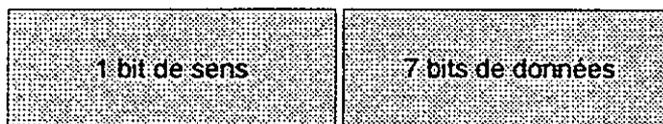
L'angle étant calculé [10] par le micro-ordinateur, il est exclu de l'envoyer directement à travers le port série. Il faudrait avant tout le mettre sous un format binaire exploitable par le microcontrôleur.

Une question se pose alors, comment choisir ce format ?

Avant d'y répondre, il est nécessaire de fixer certaines conditions qui devront être observées à savoir

a) Transmission de la donnée compressée sur un seul octet pour réduire les risques d'erreurs

Cet octet est configuré comme suit



0 \Rightarrow sens positif

1 \Rightarrow sens négatif

b) Précision de positionnement

Il faut savoir que la précision de calcul de l'angle optimal est directement liée à celle de la déclinaison. De ce fait, pour les calculs qui vont suivre, nous allons nous baser sur cette dernière.

La déclinaison varie au cours de l'année de $-23^{\circ},27'$ au solstice d'hiver à $+23^{\circ},27'$ au solstice d'été, donc si on voulait coder celle-ci sur un octet, nous obtiendrions une précision de :

$$P = (23 + 27/60)/64 * 2 = 0,366 \text{ degrés.}$$

Or la plus faible variation journalière de la déclinaison est observée aux solstices (21 juin et 21 décembre) et est de 0,001 degrés[2], ce qui rend un tel codage inopportun.

Pour résoudre ce problème, au lieu d'envoyer au microcontrôleur chaque jour la nouvelle position par rapport au repère fixe basé en 0,0 degré, nous pourrions lui envoyer la position actuelle en prenant pour repère la position antérieure. Ainsi, sachant que la plus grande variation journalière de la déclinaison est observée aux équinoxes (21 mars 21 septembre) et est de 0,4 degrés[2], nous obtiendrions une précision de :

$$P = 0,4/2 * 64 = 0,003125 \text{ degrés}$$

qui est largement suffisante, et un pas de :

$$P * 2 = 0,00625 .$$

2.3 TRANSMISSION DE LA DONNEE

Le bloc de calcul étant distant du bloc de traitement et de commande, la donnée matérialisant la position optimale du panneau solaire ne pourra être fournie sous forme parallèle, car un tel format imposerait au moins huit fils en plus des fils supplémentaire

pour les signaux de dialogue et de contrôle éventuels, ce qui générerait des effets parasites (effets capacitifs et inductifs) dus aux longueurs des fils et à la norme correspondante qui ne permettent pas de communiquer à plus de trois mètres[3].

Pour y remédier, on sérialise la donnée avant de l'envoyer à travers le port RS 232-C. Elle sera transportée théoriquement par un seul fil au lieu de huit. Certes, le temps de transfert de l'information sera au moins huit fois moins rapide mais en contre partie cette disposition permettra une meilleure immunité contre le bruit.

2.3.1 INTERFACE RS 232-C

RS 232-C désigne une norme créée par l'association Américaine de constructeurs de matériels électriques (EIA) pour normaliser le matériel et les procédures de transmission série entre ordinateurs et périphériques à travers un réseau de transmission.

Cette norme définit l'interface entre :

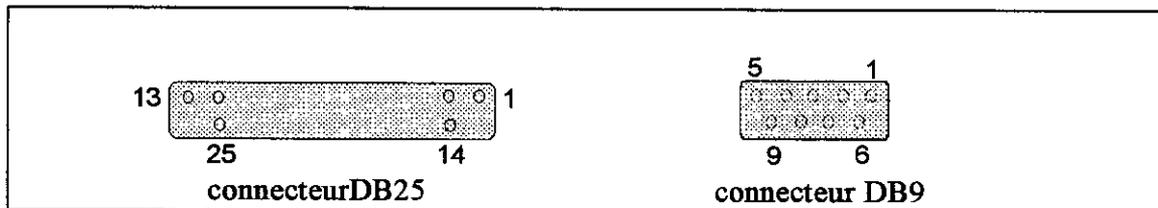
- Un DTE (Data Terminal Equipment) : un équipement périphérique c'est à dire un ordinateur ou un terminal.
- Un DCE (Data Communication Equipment) : un modem.

Dans la plupart des cas, la connexion de périphériques (imprimantes séries, tables traçante, terminal,...) via l'interface série ne passe pas par un modem. On effectue néanmoins une connexion respectant la norme RS 232-C en réalisant un modem fictif.

La norme RS 232-C définit le brochage des connecteurs DB-25 et DB-9 ainsi que les caractéristiques électriques des signaux qui y passent et qui sont décrits dans le tableau N°1[4].

Tableau N° 1 : Brochage des ports série

DB9	DB25	dénomination	signification logique	logique
1	8	data carrier detect	connexion établie	positive
2	3	recieve data	canal de réception	négative
3	2	transmit data	canal de transmission	négative
4	20	data terminal ready	terminal prêt	positive
5	7	ground	masse	
6	6	data set ready	équipement prêt	positive
7	4	request to send	demande à émettre	positive
8	5	clear to send	prêt à émettre	positive
9	22	ring indicator	indicateur d'appel	positive



3. BLOC DE TRAITEMENT ET DE COMMANDE

3.1 CONVERSION DE LIGNE

Il est à remarquer que les sorties du connecteur RS-232-C ne peuvent être appliquées directement aux circuits constituant le système. En effet, la norme RS-232-C utilise des niveaux de tension variants de -15V à +15V, ce qui nous incite à insérer un étage adaptateur de niveau permettant la conversion RS-232-C/TTL et vice versa.

Parmi le grand nombre de circuits spécialisées effectuant cette adaptation, notre choix s'est porté sur les deux circuits classiques suivants :

L'ampli de ligne quad 1488

Le récepteur de ligne quad 1489

Leurs constitutions internes sont données en figure (3-1).

3.1.1 LE MC 1488

L'ampli de ligne quad MC 1488 offre trois portes NAND et un inverseur. Leurs entrées sont compatibles TTL et leurs sorties sont compatibles RS-232-C.

Le MC 1488 exige une alimentation positive de tension comprise entre +9 et +15V et une alimentation négative comprise entre -9V et -15V pour être conforme aux spécifications de la norme RS-232-C.

Pour satisfaire l'exigence de la vitesse de variation de signal de $30V/\mu s$ de RS-232-C, un condensateur de 330 pF doit être connecté à la sortie de chaque porte du MC 1488[3].

3.1.2 LE MC 1489

Le MC 1489 offre quatre portes, toutes inverseuses. Les entrées sont compatibles RS-232-C, tandis que les sorties délivrent des niveaux TTL. Ce circuit n'exige qu'une tension d'alimentation de 5V bien qu'il reçoive des signaux à + et - 15 volts. De plus chaque porte inverseuse possède un signal de commande permettant la validation de celle-ci. La broche correspondante au signal de validation peut être laissée ouverte pour le fonctionnement en RS-232-C, ou reliée à une source de tension externe pour modifier les caractéristiques de seuil d'entrée[3]. Un condensateur à la masse peut y être connecté pour filtrer les points de bruit du signal d'entrée mais ceci se pratique rarement pour les circuits RS-232-C.

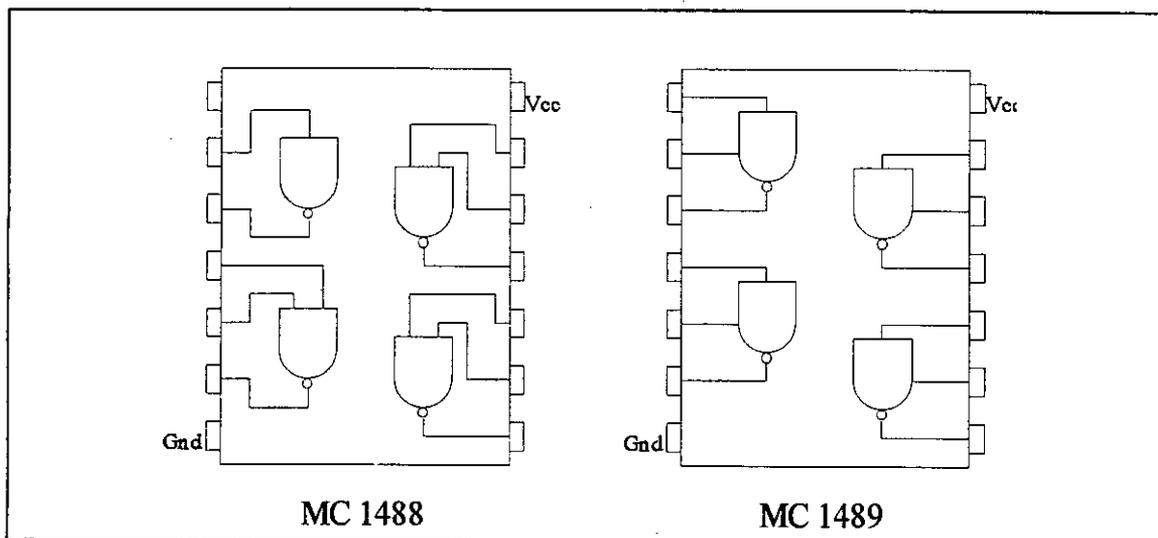


Figure 3-1 : Constitution interne des circuits de conversion de ligne .

3.2 TRAITEMENT ET COMMANDE

Directement issu de la technologie des microprocesseurs, le microcontrôleur est en fait le regroupement dans un seul boîtier de tout un système informatique : unité centrale, mémoire et circuits d'interface.

Contrairement à l'informatique classique, il n'est pas orienté "calcul", ce qui ne veut pas dire qu'il ne sait pas calculer, mais il est plutôt orienté "automatisme" c'est à dire qu'il est parfaitement bien adapté à la commande de relais, d'afficheurs, de moteurs et à la prise en compte de capteurs, de poussoirs ou de claviers.

De l'intégration poussée, dont bénéficient les microcontrôleurs, résulte une spectaculaire baisse de la consommation en plus d'une diminution évidente de l'encombrement du matériel. Il en découle, une augmentation de la fiabilité du système puisque le nombre de composants diminuant, le nombre de sources potentielles de défauts diminue aussi.

Du fait du cumul de ses avantages, le microcontrôleur a permis à certaines applications de voir le jour car elle n'étaient pas réalisables autrement pour diverses raisons : coût et encombrement en général.

3.2.1 PRESENTATION DU MICROCONTROLEUR 68705 P3S

Réaliser en technologie HMOS, ce microcontrôleur qui présente l'avantage d'une faible consommation dispose de très nombreuses ressources internes. En effet, il intègre dans un même boîtier un microprocesseur, une EPROM de 1796 octets utilisateur, une RAM de 112 octets, 20 entrées/sorties parallèles programmables et enfin un temporisateur programmable.

Le 68705 P3S ne dispose ni de bus de données ni de bus d'adresse ce qui rend son interfaçage difficile .

L'organisation interne est mise en évidence par le synoptique de la figure(3-2) .

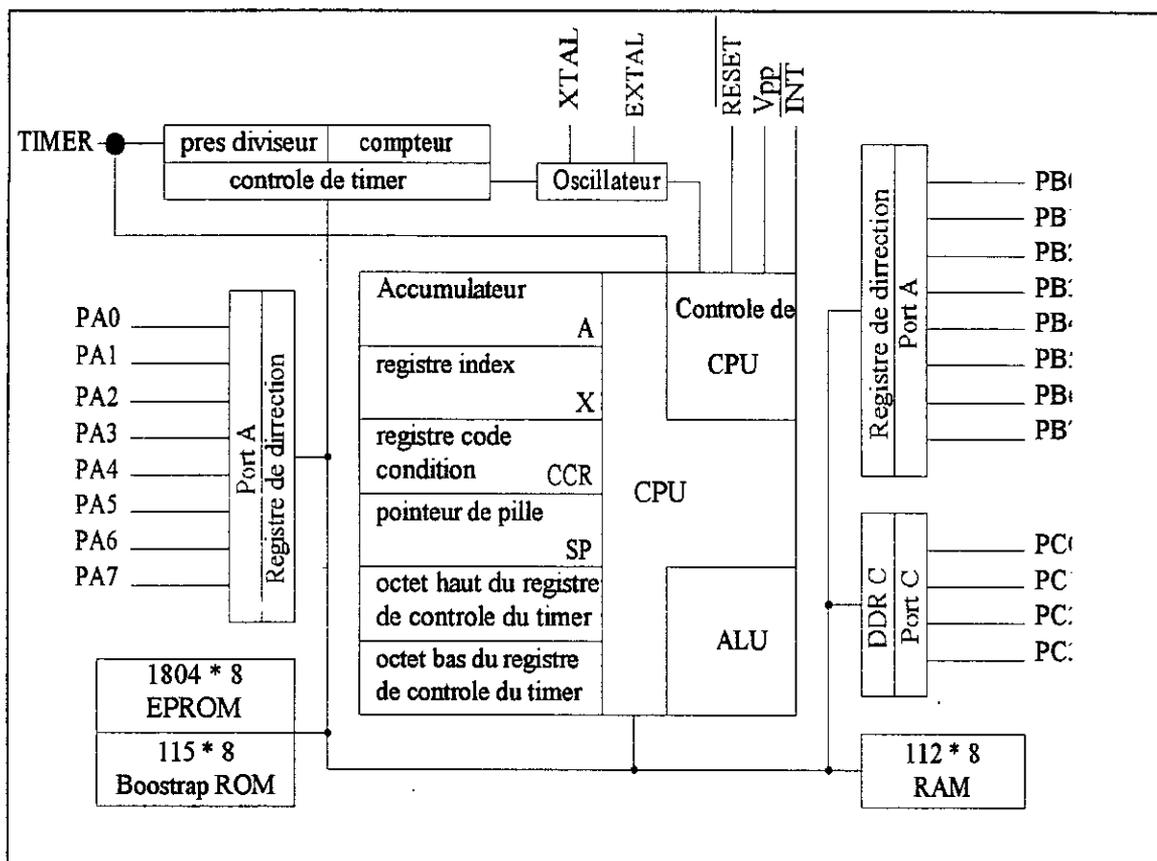


Figure 3-2 : Synoptique de l'organisation interne du 68705 P3S .

Dans ce qui suit, nous nous intéresserons uniquement aux parties utilisées dans notre réalisation.

3.2.1.1 PORTS D'ENTREE / SORTIE

Le 68705 P3S dispose de 20 lignes d'entrée/sortie, pouvant être programmées en entrée ou en sortie à travers les registres de direction¹.

Ces lignes, réparties selon trois ports, sont toutes compatibles TTL en entrée et sortie.

port A formée de 8 lignes compatibles CMOS en sortie.

port B formée de 8 lignes compatibles CMOS en entrée

port C formée de 4 lignes CMOS en entrée et devant être rappelées au niveau haut par des résistances de pull up [5].

3.2.1.2 REGISTRES INTERNES

Ce microcontrôleur possède un CPU de 6800 réduit. Effectivement, il a

_ Un seul accumulateur de 8 bits.

_ Un registre de codes conditions, qui fait 5 bits, qui sont les indicateurs H, I, N, Z et C dans l'ordre. Le bit I est positionné comme avec le 6802.

¹ Il est à remarquer que contrairement à ce que l'on pourrait croire les lignes d'un même port ne peuvent être programmées individuellement. elle doit être programmées de la même manière (tous en entrée ou tous en sortie).

3.2.2 PROGRAMMATION DU 68705 P3S

Le 68705 P3S contient 115 octets de ROM dans lesquelles réside un programme dit BOOSTRAP chargé d'effectuer la programmation et la vérification automatique de son EPROM.

Un programmeur recommandé par le constructeur permet de transférer le programme déjà installé sur une mémoire externe (EPROM 2716) dans celle du 68705 P3S [5].

3.2.3 DESCRIPTION TECHNIQUE

Le schéma de cet étage est donné en figure (3-4).

3.2.3.1 CHOIX DES PORTS

La communication entre le micro-ordinateur et le microcontrôleur sera assurée par :

- Quatre lignes de contrôle de flux dont deux sont reliées au port C³, et les autres au port B et à l'entrée d'interruption, comme indiqué ci dessous.

DTR	→	PC0	programmée en sortie
DSR	→	PB6	programmée en entrée
RTS	→	PC1	programmée en sortie
CTS	→	INT	programmée en entrée

- Une ligne de réception de données connectée au port B.

RxD	→	PB7
-----	---	-----

- Le moteur pas à pas sera commandé par quatre lignes du port B (PB0...PB3). Le choix de ce port parmi les deux autres disponibles (port A et port B) est basé sur les caractéristiques de ses lignes. En effet, elles peuvent absorber un courant de 10 mA en sortie. Ce dernier pourra être appliqué directement à des transistors

Tout les ports, de même que les lignes TIMER, et Interrupt sont disponibles sur un connecteur en sortie permettant ainsi la commande de circuits annexes tel que afficheurs, moteurs etc.... ou la prise de commande du système par un autre système maître.

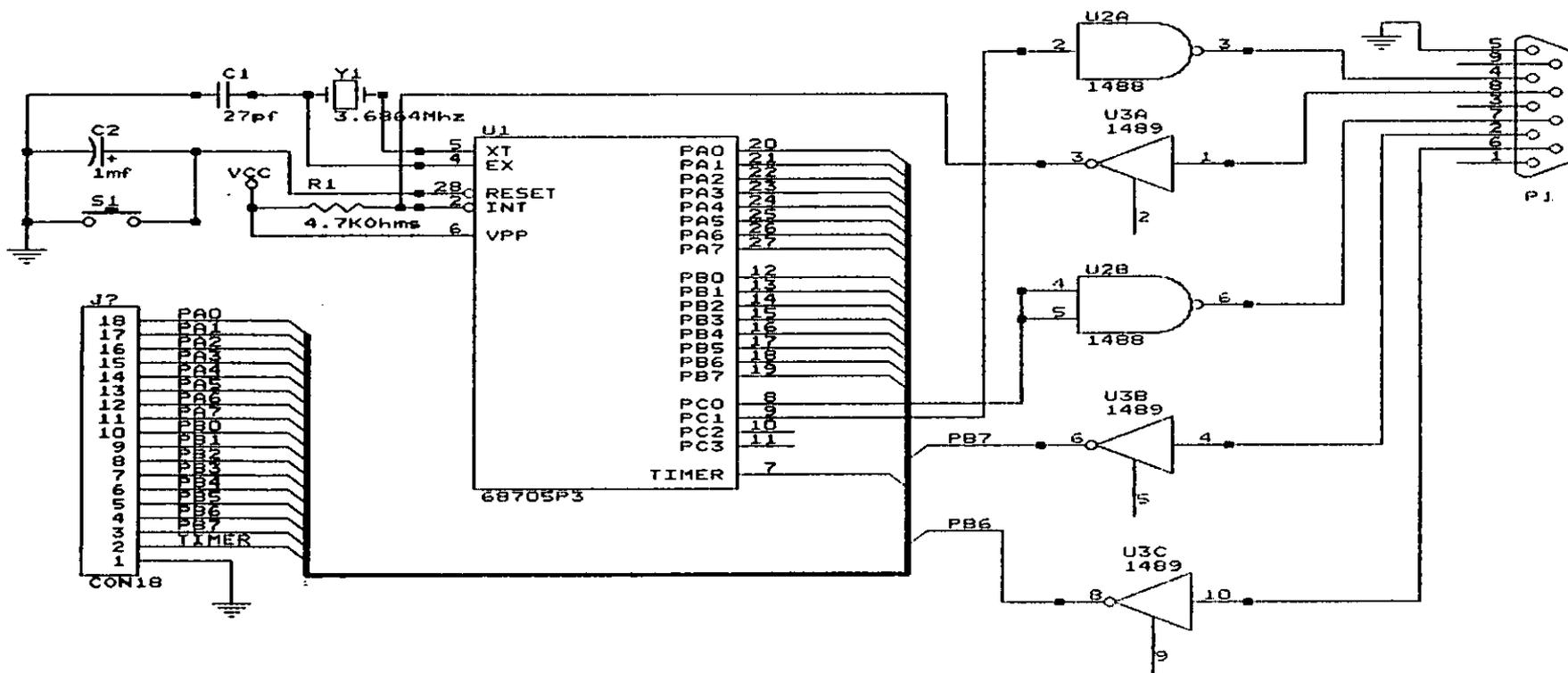
3.2.3.2 CHOIX DE L'HORLOGE

Le microcontrôleur dispose de deux broches XTAL et EXTAL qui permettent de piloter l'oscillateur de l'horloge interne. Ces broches peuvent être utilisées de différentes manières selon la précision désirée.

Pour notre part une bonne précision est requise afin de minimiser les erreurs dues au passage du format série au format parallèle ce qui nous oriente vers l'option utilisant un quartz dont la valeur peut être comprise entre 400 Khz et 4.2 MHz [4].

³ Les lignes du port C doivent être rappelées au niveau haut par des résistances.

Le choix de la valeur du quartz se fait de manière à ce que celle ci soit un multiple des vitesses de transmission normalisées .Sachant que le générateur de cadence de bit qui justement génère ces vitesse la, les obtient par division d'une fréquence étalon de 1.8432 Mhz[6] nous pourrions choisir comme fréquence un multiple de celle ci à savoir 3.6864 Mhz.



4. BLOC DE PUISSANCE

Bien que l'apparition des moteurs pas à pas remonte à très longtemps, leur application n'a connu un véritable essor qu'après l'avènement des microprocesseurs et le développement énorme de l'électronique de puissance.

A l'inverse des moteurs classique dont l'arbre tourne de façon continue, les moteurs pas à pas sont des moteurs dont la rotation s'effectue par mouvements angulaire discrets. Ce qui facilité le rôle d'interface entre le système de traitement numérique et leur environnement mécanique pour peu qu'un étage de puissance soit inséré.

4.1 DEFINITION DES MOTEURS PAS A PAS

Un moteur pas à pas est un moteur électrique synchrone qui convertit directement un signal électrique digital en série de mouvements angulaires discrets.

Sa caractéristique fondamentale est donc d'effectuer à chaque commutation du courant dans les différents enroulements du stator, un déplacement angulaire bien précis dit pas .

Le pas est l'unité de déplacement angulaire du moteur lorsque au moins l'un des enroulements du stator est excité.

Le stator qui constitue la partie fixe du moteur pas à pas possède un certain nombre de dents, sur lesquelles sont disposées les bobines d'alimentation formant ainsi les phases du moteur.

La partie tournante du moteur appelée rotor, dépend du type de moteur.

4.2 CLASSIFICATION DES MOTEURS PAS A PAS

Les moteurs pas à pas sont classés en fonction de leur structure mécanique interne et du phénomène physique qui est à l'origine du couple. Nous distinguons principalement trois catégories [7] :

- Les moteurs à réluctance variable
- Les moteurs à aimant permanent
- Les moteurs hybrides

Pour ce qui est de notre application ,notre choix s'est porté sur le moteur à aimant permanent .En effet ,ce type de moteur est de conception simple et robuste ,de plus il présente un couple élevé .

4.2.1 LE MOTEUR A AIMANT PERMANENT

C'est un moteur dont le rotor est un aimant permanent .Ce type de moteur fonctionne par la réaction entre un champ électromagnétique et un aimant permanent, selon

l'excitation électrique des phases de stator, le circuit magnétique va se fermer en attirant devant un pôle Nord de stator un pôle Sud du rotor.

4.3 CARACTERISTIQUE DES MOTEURS PAS A PAS

Le choix d'un moteur pas à pas se fait selon ses performances statiques et dynamiques.

4.3.1 CARACTERISTIQUE STATIQUES

Ces caractéristiques concernent le moteur à l'état de repos. Elles sont essentiellement liées à la constitution mécanique et électrique du moteur

- a) L'angle de pas: c'est la valeur de l'angle de rotation effectué par le moteur
- b) Le couple de détente.
- c) Le couple de maintien
- d) La précision de positionnement.

4.3.2 CARACTERISTIQUES DYNAMIQUES

Ces caractéristiques sont obtenues en phase de fonctionnement sans perte de pas du moteur, c'est à dire qu'à chaque impulsion de courant correspond un et un seul pas. Elles sont déterminées par [7][9] :

- Le couple dynamique :c'est le couple du moteur en rotation, il dépend essentiellement de l'alimentation et n'est pas une caractéristique du moteur mais de l'ensemble moteur alimentation compte tenu de la charge.
- La fréquence maximale de démarrage-arrêt :c'est la fréquence maximale de commutation permettant un démarrage, montée en vitesse puis arrêt sans perte de pas d'un moteur à vide.
- La relation entre la fréquence des impulsions d'excitation, la vitesse de rotation et le nombre de pas est

$$N = 60 F / S$$

N vitesse de rotation [tours/minute]

F fréquence [hertz]

S nombre de pas

4.3.3 ZONES DE FONCTIONNEMENT

La figure (4-1) présente les zones de fonctionnement suivantes:

- La zone de démarrage-arrêt (1,3): correspond à un fonctionnement synchrone avec contrôle de la position, mais aussi de l'énergie d'accélération. Ainsi, si on envoie sur les différentes phases d'un moteur, un train d'impulsions de courant de fréquence donnée, il faut que le moteur démarre et s'arrête à la fin du train d'impulsions dans la position voulue.
- La zone de fonctionnement en survitesse (4): correspond à un régime qui intervient lorsque la fréquence des impulsions est trop grande ;la phase suivante est excitée alors que le rotor est encore en phase d'accélération.

- La zone de fonctionnement erratique (2): où le fonctionnement est irrégulier, car les fréquences de travail sont très voisines des fréquences de résonance de l'ensemble.

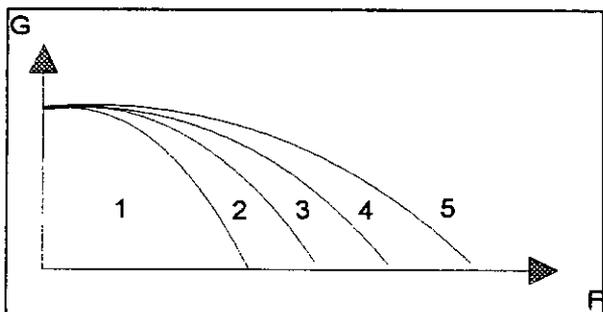


Figure 5-1 :couple en fonction de la fréquence d'excitation des phases

-La zone limite correspond à une fréquence trop importante

4.4 CONCEPTION DE L'ETAGE DE PUISSANCE

Les signaux fournis par le microcontrôleur n'excédant pas quelques milliampères, il est impératif d'insérer un étage de puissance capable de rehausser ces dernières afin de permettre la commande de moteur pas à pas dont la consommation en courant peut atteindre plusieurs ampères.

La structure et les caractéristiques de cet étage sont des éléments importants entrant pour une large part dans la définition des performances réalisés par le système.

La figure (5-2) illustre le schéma adopté pour cet étage

Dans la conception de cet étage de puissance, nous avons opté pour l'utilisation de transistor MOSFET. Effectivement, ces derniers offrent de grandes vitesses de commutation en plus d'une stabilité thermique accrue et sont aptes à faire passer des courants intenses ce qui les dispose tout naturellement pour des commutations de puissance telle la commande de moteurs pas à pas.

4.5 FONCTIONNEMENT DE L'ETAGE DE PUISSANCE

Le blocage du transistor provoque la saturation du MOSFET. Ainsi l'enroulement sera alimenté. Une diode de roue libre est insérée en parallèle avec l'enroulement pour éviter un éventuel claquage du transistor bien qu'il intègre une diode de protection.

La saturation du transistor bipolaire bloquera le MOSFET déconnectant l'enroulement du moteur pas à pas.

Il est à remarquer que le port B attaque directement le transistor bipolaire à travers une résistance qui a pour rôle de limiter le courant. De plus la tension de 24 volts est abaissée à l'aide d'un pont diviseur pour éviter une tension sur le gate du MOSFET supérieure à 20 volts qui est la limite admise par ce dernier[8].

4.6 CALCUL PRATIQUE

Le moteur ne nécessitant pas un courant d'intensité supérieure à 2 ampère, il retourne des caractéristiques de transfert qu'une tension $V_{GS} = 6$ volts est largement suffisante pour satisfaire les besoins de la commande.

De ce fait, on fixe V_{cc} à 6 volts.

$$V_{dd} / V_{cc} = 4 \Omega \Rightarrow R1/R2=3$$

Nous avons décidé de fixer le courant collecteur du transistor bipolaire à 2 mA pour des raisons d'économie car le MOSFET est sensible à la tension et non au courant appliqué à son entrée.

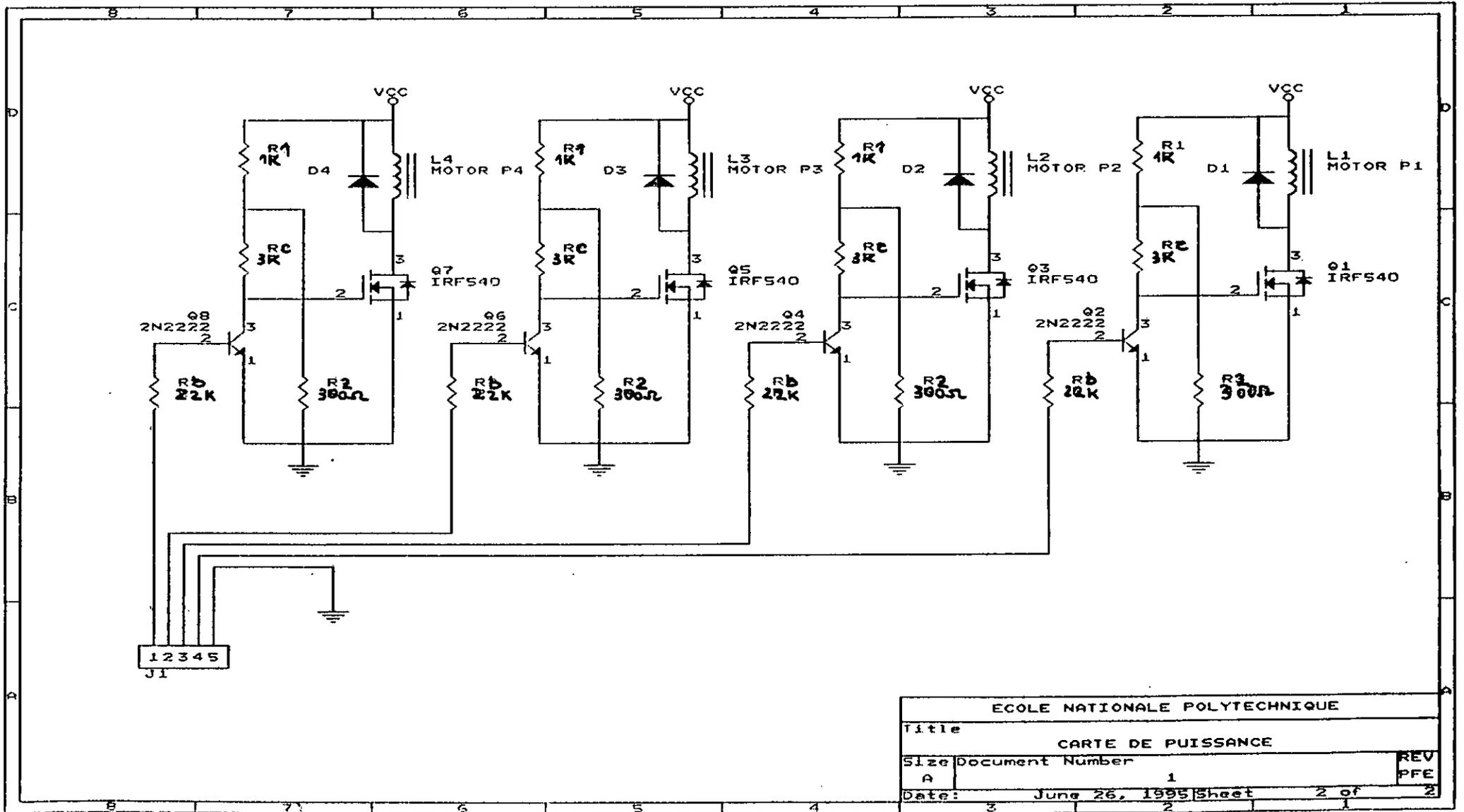
On choisit $I = 10 I_c = 20$ mA

$$R1 = (V_{dd}-V_{cc})/I = 24-6)/0,002 = 900 \Omega \approx 1K\Omega$$

$$R2 = R1/3 = 300 \Omega$$

$$R_c = V_{cc}/I_c = 6/0,002 = 3K\Omega$$

$$R_b = \beta (V_e - V_{be})/I_c = (5-0,6)/200 \cdot 10^{-6} = 22 K\Omega$$



ECOLE NATIONALE POLYTECHNIQUE

Title		CARTE DE PUISSANCE	
Size	Document Number	REV	
A	1	PFE	
Date:	June 26, 1995	Sheet	2 of 2

5. PROGRAMMATION DU PORT D'E/S DU MICRO-ORDINATEUR

Ce chapitre traite des différentes possibilités de gestion du port série d'un micro ordinateur .Il contribuera pour une grande part à la compréhension de la façon avec laquelle les données sont émises et reçues ainsi que le contrôle de la communication et les possibilités de dialogues .

La programmation du port d'E/S du micro-ordinateur peut se faire de différentes manières :

- Par des fonctions DOS et BIOS.
- Par programmation directe des ports.

Les systèmes d'exploitation DOS et BIOS fournissent un certain nombre de fonctions internes qui peuvent être appelées par programme, plusieurs d'entrés elles concernent les communications série.

Les communications série peuvent avoir recours à deux fonctions DOS et quatre fonctions BIOS qui sont appelées par l'intermédiaire d'interruptions programmées.

5.1 LES FONCTIONS DOS

Les deux fonctions DOS qui concernent les communications série sont appelées en exécutant une interruption 21H.

La première fonction sert aux entrées série, celles-ci sont réalisées en mettant le registre AH à 3 et en exécutant une instruction INT21H. Le DOS attend de recevoir un caractère de COM1 et le retourne dans le registre AL.

L'autre fonction DOS concerne les sorties série, elles sont effectuées en mettant le registre AH à 4, en plaçant le caractère à émettre dans le registre DL et en exécutant une instruction INT21H. Le caractère sera envoyé à COM1.

Ces deux fonctions DOS présentent de sérieux handicapes qui restreignent leur domaine d'application . En effet ,elles ne permettent pas d'accéder au contrôle de flux et n'offrent aucun moyen pour initialiser les paramètres de communication (vitesse de transfert, format de mot, etc..) .De plus aucune information d'erreur n'est retournée .

5.2 LES FONCTIONS BIOS

L'interruption 14H fait appel aux quatre fonctions BIOS concernant les communications série. Il faut placer un nombre, de 0 à 3, dans AH pour indiquer laquelle des quatre fonctions est appelée, puis indiquer ensuite un numéro de port dans le registre DX, 0 pour COM1 et 1 pour COM2.

■ La fonction initialisait des paramètres de communication

La première fonction BIOS, la fonction 0, sert à initialiser les paramètres de communication. On y accède en mettant le registre AH à 0 et le registre DX au numéro de port, en plaçant un octet⁴ représentant les paramètres vitesse, parité, longueur du mot et nombre de bits STOP dans le registre AL, puis en exécutant une instruction INT14H.

Lorsque les paramètres ont été initialisés, la fonction retourne l'état courant du port dans le registre AX⁴.

■ La fonction émission de caractères

La deuxième fonction BIOS, fonction 1, sert à l'émission des caractères. On l'appelle en mettant AH à 1 et DX au numéro de port, ensuite on place le caractère à envoyer dans le registre AL, puis on exécute l'instruction INT14H.

Le caractère n'est pas envoyé tant que les lignes de contrôle entrantes (DSR et CTS) ne sont pas à l'état haut.

La mise en oeuvre conventionnelle consiste tout d'abord, à exécuter la fonction saisie de l'état du port, puis la fonction Emission de caractères une fois que l'on sait que les conditions sont favorables, c'est à dire que les lignes de contrôle DSR et CTS sont actives.

En retour, le registre AH rend compte de toute condition d'erreur⁴. Si le bit 7 de AH est à zéro, alors l'émission a réussi, s'il est à un, les bits restant indiquent le type d'erreur qui s'est produite.

■ La fonction réception de caractères

La fonction réception de caractères, fonction 2, est appelée en mettant AH à 2, DX à la valeur du numéro de port, puis en exécutant l'instruction INT14H. Le BIOS attend qu'un caractère en provenance du port série, soit reçu ou bien qu'un dépassement de temps soit atteint. Lorsqu'un caractère est reçu, il est placé dans AL et toute condition d'erreur sera reportée dans AH.

Si AH vaut zéro, aucune erreur ne s'est produite, si AH est différent de zéro, les bits 0 à 7 indiquent alors la condition d'erreur. Toutefois, si le bit 7 est mis à 1, indiquant une erreur de dépassement de temps, les bits restant n'ont aucune signification particulière.

■ La fonction saisie de l'état du port

La fonction saisie de l'état du port, fonction 3, est appelée en mettant AH à 3, DX à la valeur du numéro de port, puis en exécutant l'instruction INT14H. Elle fournit diverses

⁴Pour plus de détails voir la référence [6].

informations à propos de l'état courant du port série ,puis retourne cet état dans le registre AX. La figure.... montre la signification des différents bits.

■ Contrôle de flux avec le BIOS

Le BIOS se comporte très bizarrement vis à vis du contrôle de flux. La fonction Réception de caractère élève le signal DTR et abaisse le signal RTS elle attend ensuite l'arrivée d'un caractère et retourne une erreur de dépassement s'il n'en parvient aucun après un certain laps de temps. La fonction Emission de caractère commute DTR et RTS, puis attend que DSR et CTS soient tous les deux mis à l'état haut par l'appareil connecté au port d'E/S. Si les lignes ne sont pas mises à l'état haut, la fonction retourne une erreur de dépassement de temps.

On voit que le BIOS requiert pour émettre deux signaux de contrôle alors qu'il n'en délivre qu'un seul en réception, de plus, il met RTS à zéro lorsqu'il est en attente de réception.

Par ailleurs, les signaux de contrôle sortant sont à l'état bas, tant que la fonction Emission ou la fonction Réception n'a pas été appelée.

Si on venait à suivre la procédure normale, qui consiste tout d'abord, à appeler le port pour voir si un caractère a été reçu, puis Réception de caractère dans le cas où il y'en a un, cela peut poser problème. En fait, on peut ne rien recevoir du tout, parce que ,l'on n'appelle pas la fonction Réception de caractère tant que quelque chose n'a pas été reçu, mais en fait, rien ne sera reçu car la ligne de contrôle se trouve dans l'état bas, puisque la fonction Réception de caractère n'a pas encore été exécutée

Cette interruption BIOS ,bien que nettement plus intéressante que celle du DOS , présente plusieurs désavantages rendant son usage épineux ,car elle ne propose pas non plus de possibilité de gestion libre du flux ,de plus ,elle limite les possibilités qu'offre l'UART du PC.

5.3 PROGRAMMATION DIRECTE DES PORTS D'E/S

Cette technique de programmation consiste à accéder directement au port d'E/S en utilisant des instructions appropriées.

Au prix d'une nécessité de programmation en langage assembleur elle permet une gestion libre des lignes de contrôle de flux ainsi qu'une exploitation complète des possibilités qu'offre l'UART 8250 .

Toute fois ,la connaissance des différents registres de ce circuit est indispensable.

5.3.1 PRESENTATION DE L'UART 8250

L'UART (Universal Asynchrone Receiver Transmitter) est un circuit d'interface travaillant en mode asynchrone .Il permet de réaliser la liaison série entre le micro-ordinateur et ses interfaces .Ce circuit possède des registres internes qui sont repartis comme suit

5.3.1.1 REGISTRES DE CONTROLE

Quatre registres de contrôle servent à recevoir les commandes provenant de l'unité centrale de traitement (UCT).

■ Le registre de contrôle de ligne

Il est utilisé pour initialiser les paramètres de communication. La signification de chaque bit est présentée ci-dessous

Tableau N°2 : Description du registre de contrôle de ligne .

Bit	Signification ⁵
0	longueur du mot
1	Longueur du mot
2	Bit d'arrêt
3	Valide la parité
4	Sélectionne la parité
5	Bit de parité a 1
6	Signal break
7	Bit d'accès a la bascule de division (DLAB)

■ Registre de contrôle de modem

Le registre de contrôle de modem commande les signaux de contrôle émis par l'UART.

Les bits 0, 1 permettent de mettre les sorties DTR et RTS aux niveaux désirés.

Les bits 2 et 3 contrôlent les sorties auxiliaires.

Le bits 4 valide le mode de test par diagnostic⁶.

Les bits 5 à 7 sont en permanence maintenus à zéro.

■ Registre de validation des interruptions

Ce registre sert à indiquer au 8250, la cause d'une éventuelle interruption .Le tableau 3 décrit la signification de chacun de ces bits .

⁵Pour Plus de détails voir référence [6].

⁶Le mode teste par diagnostique permet de vérifier le bon fonctionnement de l'UART .

Tableau N°3 : Description du registre de validation des interruptions .

Bit	Signification
0	donnée disponible
1	registre d'émission vide
2	Etat de la ligne de réception
3	Etat d modem
4,5,6,7	Forcés à zéro

■ Les bascules de division de la vitesse de transfert

La valeur de la vitesse de transfert est établie en introduisant dans deux registres le nombre par lequel la fréquence de l'horloge d'entrée (1.8432 MHZ) doit être divisée.

La fréquence résultante vaut 16 fois la vitesse de transfert. Ces deux registre sont d'une part l'octet le moins significatif de la bascule de division (DLL) et d'autre part l'octet le plus significatif de la bascule de division (DLM).

5.3.1.2 REGISTRES D'ETATS

■ Registre d'état de la ligne

Ce registre sert à fournir des informations relatives à la réception et à l'émission des données. La signification des bits individuels est illustrée par le tableau suivant .

Tableau N°4 : Description du registre d'état de la ligne .

Bit	Abréviation	Nom	signification (s'il est mis à 1)
0	DR	Donnée prête	Un caractère a été reçu et placé dans le registre tampon de réception
1	OE	Erreur de cadence	un caractère a été reçu avant que le précédent n'ait été retiré
2	IPE	Erreur de parité	Le bit de parité est incorrecte
3	FE	Erreur de trame	un caractère reçu ne contient pas de bit d'arrêt valide
4	BJ	signal BREAK	un BREAK a été reçu*
5	THRE	registre de saisie de l'émetteur vide	l'UART est prêt à recevoir un nouveau caractère pour l'émettre
6	TSRE	registre à décalage de l'émetteur vide	le caractère a été émis
7		réservé	ce bit est forcé à zéro

■ Registre d'état du modem

Ce registre donne des informations sur l'état des lignes de contrôle de flux. La signification de ses bits individuels est indiquée dans le tableau suivant .

Tableau N°5 : Description du registre d'état du modem .

Bit	Nom	Signification (s'il est mis à 1)
0	Delta ⁷ CTS	La ligne CTS a changée
1	Delta RTS	La ligne RTS a changée
2	TERI	L'indicateur d'appel a changé de haut vers le bas
3	Delta RLSL	La détection signal de la ligne de la réception a changé
4	CTS	L'entrée prêt à émettre est au niveau haut
5	DSR	L'entrée poste de donnée prêt est au niveau haut
6	RI	L'indication d'appel est haut
7	RSLD	La détection signal de la ligne de réception est haut

Les quatre bits de poids faible indiquent qu'un changement s'est produit depuis la dernière lecture du registre, tandis que les quatre autres bits restant donnent l'état réel des lignes considérées.

Il est à noter, qu'il n'est pas nécessaire que les lignes de contrôle de flux entrantes soient à 1 pour que l'UART émette.

■ Le registre identificateur d'interruption

⁷ DELTA veut dire changement .

Il fournit l'état courant des interruptions en attentes. Le bit 0 est mis à un s'il y'en a aucune, s'il est mis à zéro, les bits 1 et 2 indiquent, laquelle est en attente, selon la configuration présentée dans le tableau (6). Les bits 3 à 7 sont forcés à zéro.

Tableau N°6: Description des bits 1 et 2 du registre identificateur d'interruptions .

Bit 2	Bit 1	Interruption en attente
1	1	Etat de la ligne
1	0	Donnée reçue disponible
0	1	Registre de saisie de l'émetteur vide
0	0	Registre du modem

5.3.1.3 REGISTRES TAMPONS

Les registres tampons constituent la troisième catégorie de registre de l'UART. Il y'en a deux sortes les tampons de réception et les tampons d'émission.

■ Le registre tampons de réception

Le registre tampon de réception extrait le dernier caractère reçu. Une fois lu, le registre d'état de la ligne indique que le tampon de réception est vide tant qu'un autre caractère n'est pas reçu.

■ Le registre de saisie de l'émetteur

Le registre de saisie de l'émetteur recueille le nouveau caractère à émettre, il est placé là par le programme.

Tout les registres internes de l'UART sont localisés à des emplacements mémoire. Ci-dessous, les adresses utiles des ports et des différents registres.

Tableau N°7 : Adressage des différents registres de l'UART.

Port 1	Port 2	Registre sélectionné	DLAB
3F8	2F8	Tampon de réception	0
3F8	2F8	Tampon d'émission	0
3F8	2F8	Octet fort du diviseur	1
3F9	2F9	Octet faible du diviseur	1
3F9	2F9	Registre de validation des interruptions	0
3FA	2FA	Registre d'identification des interruptions	X
3FB	2FB	Registre de contrôle de Ligne	X
3FC	2FC	Registre de contrôle de modem	X
3FD	2FD	Registre d'état du modem	X
3FE	2FE	Registre d'état de ligne	X

6. PROGRAMME DE LA RECEPTION SERIE

Le microcontrôleur se charge en première étape de réceptionner la donnée série bit par bit à travers une ligne de son port B puis de la reconvertir au format parallèle afin de permettre, après traitement, la commande du mouvement d'inclinaison du panneau .

La transmission étant en mode asynchrone, le format standard du mot série est détaillé ci dessous . Voir figure (6-1)

Le mot transmis comprend :

- Un bit de départ, qui marque le début.
- Les bits constituant la donnée qui peuvent être au nombre de 5,6,7 ou 8.
- Un bit optionnel de parité pour la détection d'erreurs.
- Un ou deux bits de stop pour indiquer la fin de transmission.

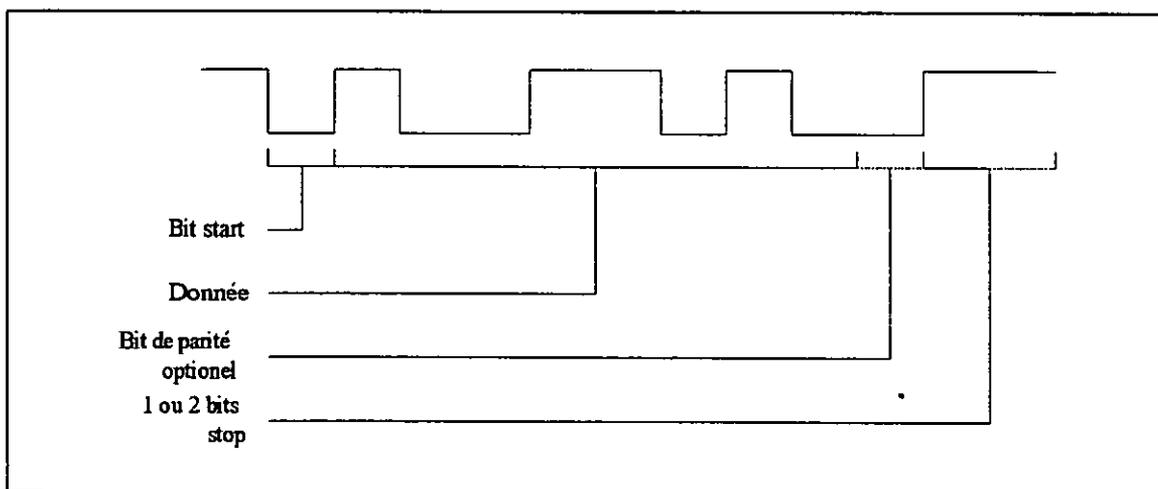
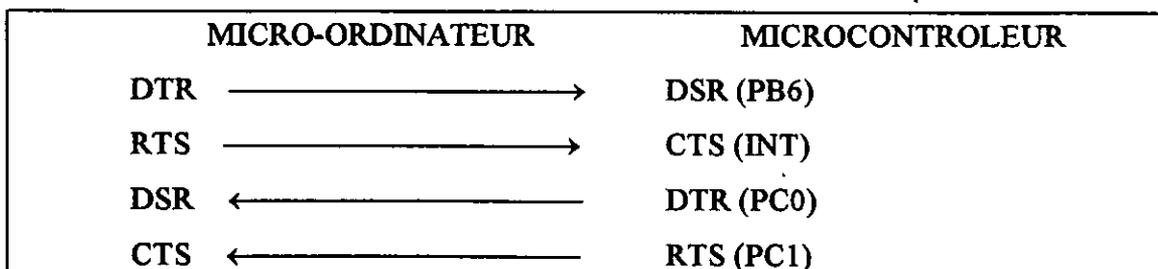


Figure 6-1 :Forme d'onde d'une donnée transmise au format asynchrone

6.1 GESTION DE LA COMMUNICATION

La gestion de la communication est assurée par quatre lignes de contrôle de flux connectés comme indiqué



Le dialogue entre les deux communicants s'effectue après les étapes suivantes :

A l'état initial lors de la connexion du micro-ordinateur ,toutes les lignes de contrôles sont à l'état bas.

Lorsque celui-ci a besoin d'émettre ,il active ses sorties DTR et RTS provoquant ainsi une demande d'interruption ,à la suite de laquelle il attend que ses entrées CTS et DSR passent au niveau haut .

Le microcontrôleur exécute alors le sous programme d'interruption qui commence par tester si son entrée DSR (PB6) est à l'état actif pour vérifier que la demande d'interruption est effective et qu'elle n'est pas due à des fluctuations de la ligne au quel cas il reprendra son travail initial.

Dans le cas contraire le microcontrôleur active ses lignes DTR (PC0) et RTS(PC1) indiquant de ce fait au micro-ordinateur qu'il est prêt a recevoir .

Une fois les lignes d'entrée actives la séquence de transmission commence .

A la fin de cette séquence le microcontrôleur devra indiquer au micro-ordinateur si la donnée a bien été reçue en mettant DTR (PC0) et RTS (PC1) au niveau bas .De ce fait ,la communication sera rompue et le programme de traitement et commande entamé .Si non ,RTS sera maintenue à un alors que DTR sera remise à zéro pendant un certain temps au bout duquel le 68705 réactivera ses lignes de sortie pour demander le renvoi du caractère .

Pour sa part le micro-ordinateur commence par tester si la réception a réussie au quel cas il désactivera la liaison en mettant RTS et DTR au niveau bas .Autrement il se chargera de recommencer la séquence d'émission.

Les organigrammes de la figure (6-2) et la figure (6-3) illustrent le fonctionnement individuel des deux communicants.

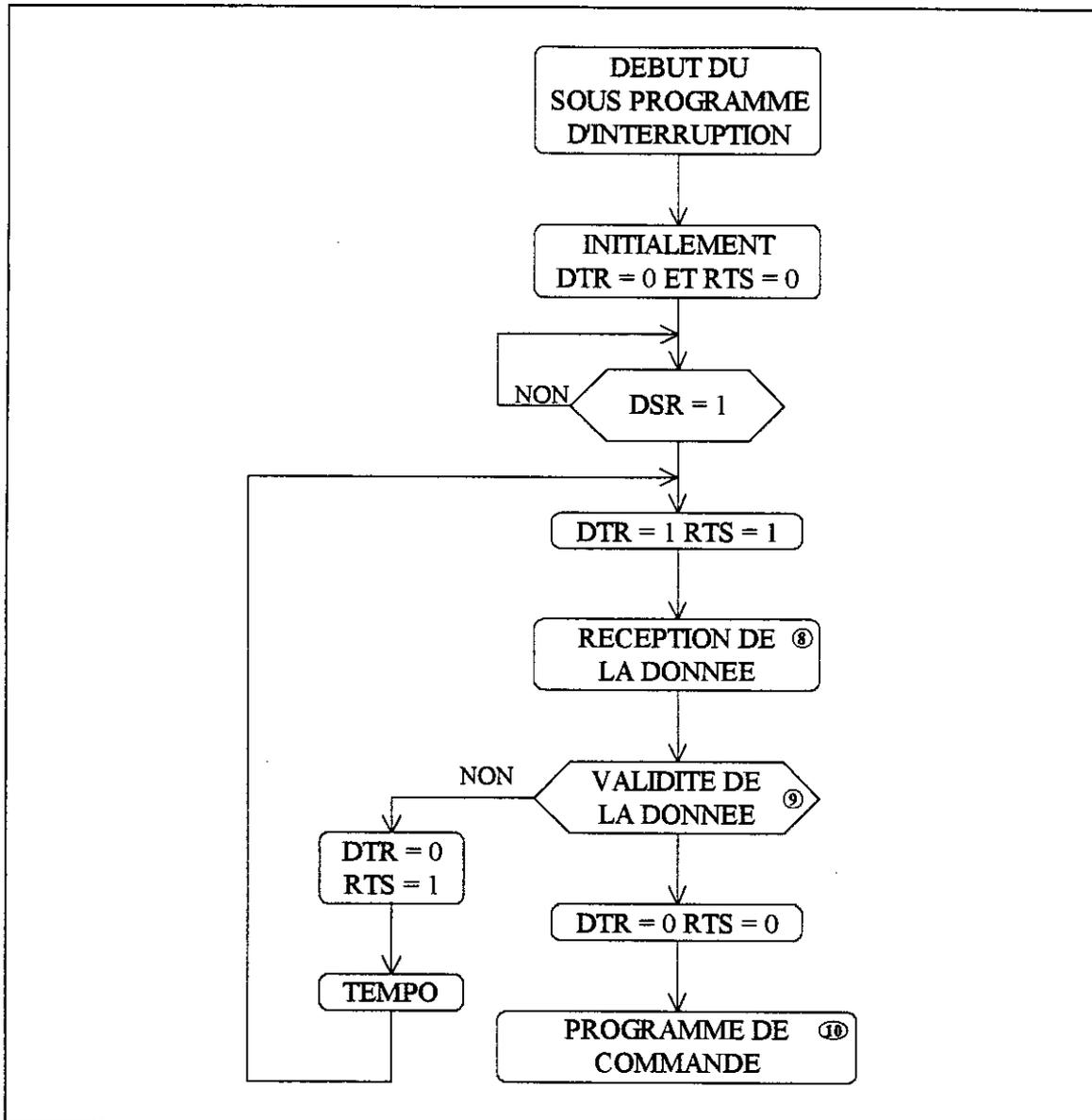


Figure 6-2 : Organigramme de la gestion du flux par le microcontrôleur .

89 10

⁸ L'organigramme de la réception de la donnée est donné en figure (x) .

⁹ L'organigramme de test de la validité de la donnée est donné en figure (x) .

¹⁰ L'études de cette partie fait l'objet du chapitre 8 .

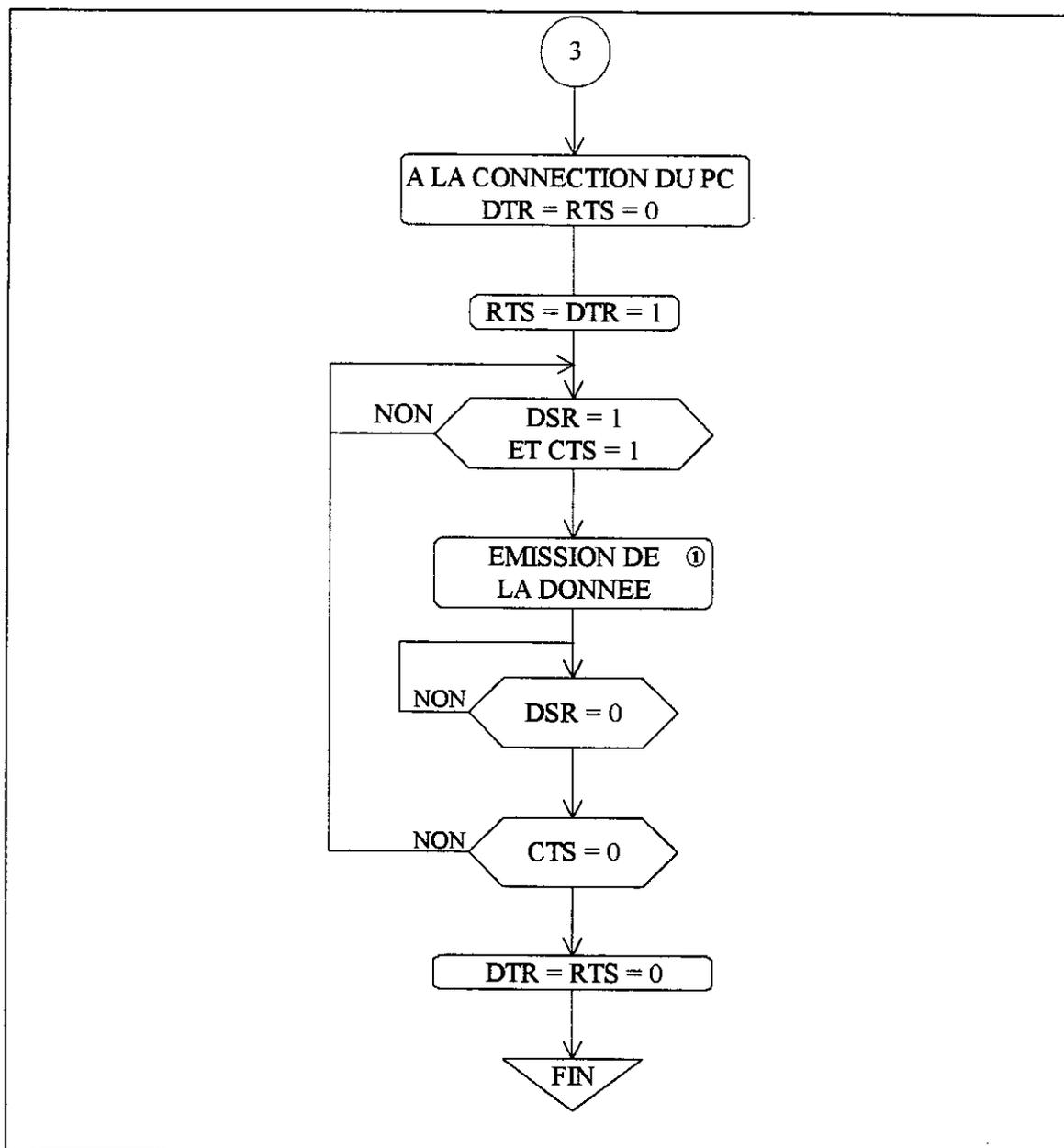


Figure 6-3 : Organigramme de la gestion du flux par le P.C.

6.2 PROTOCOLE DE COMMUNICATION

Le protocole de communication choisi est:

- ◆ 8 bits de données
- ◆ Parité paire
- ◆ Deux bit de stop
- ◆ Vitesse de transmission de 1200 BAUDS

6.3 EMISSION DE LA DONNEE

La séquence d'émission de la donnée est précédée par une mise en forme de la valeur calculée précédemment .

6.3.1 MISE EN FORME DE LA DONNEE

Nous avons vu au chapitre du bloc de calcul que la position optimale était obtenue par une équation cosmologique .Le résultat du calcul étant une valeur réelle ,une mise en forme est indispensable afin de permettre son envoi et son traitement par le microcontrôleur .Ceci se fait comme suit .

- Le micro-ordinateur commence par calculer la différence entre la position précédente du panneau qui fut préalablement sauvegardée dans un fichier et la nouvelle position calculée .
- conformément à ce qui a déjà été expliqué au chapitre (3) ,le codage de la donnée se fait par la formule :

$$N_i = \beta_i * 128 / 0.4$$

Où β_i représente l'angle d'inclinaison du panneau au jour i .

N_i représente un nombre .

$$N = N_i - N_{i-1}$$

- La partie entière de N sera codée sur 7 bits d'un octet dont le huitième bit sera mis à un ou à zéro suivant que la différence N soit négative ou positive .
- La partie décimale de N est soustraite à N_i pour prendre en compte le fait que le positionnement de héliostat ne s'est pas fait de manière exacte ,mais à cette partie décimale pres qui sera réintroduite lors du prochain positionnement .Ainsi on n'aura pas un cumul d'erreurs au fil des jours .
- Ceci fait N_i est sauvegardé dans un fichier pour permettre son utilisation dans le prochain positionnement .

6.3.2 EMISSION DE LA DONNEE

A ce stade le micro-ordinateur se préparera a pour émettre cette donnée en effectuant une initialisation des différents paramètre de la communication (format du mot , vitesse de transmission ,parité ...).

La fonction 0 de l'interruption 14h du BIOS apparaît la mieux indiquée pour cela car elle permet de réaliser une initialisait avec un minimum d'instructions .

Après cela les lignes de contrôle de flux sont activées et testées comme indiqué précédemment par accès directe aux registres internes de l'UART car ce mode est le seul à permettre une gestion libre et poussée de ces lignes .

Une fois que le microcontrôleur est prêt à recevoir ,la donnée sera mise dans le registre tampon d'émission .

Après s'être assuré que la donnée a été correctement reçue ,le micro-ordinateur désactivera la communication et achivera l'exécution du programme .Autrement ,il recommencera la séquence d'émission jusqu'à ce que la donnée soit bien reçue .

6.4 RECEPTION DE LA DONNEE

Nous avons vu au cours de l'étude de la partie matérielle que la donnée série est reçue à une certaine cadence à travers la ligne PB7 du microcontrôleur qui est chargé de reconstituer convenablement la donnée parallèle initiale.

La procédure de réception de la donnée est la suivante :

Avant toute chose ,le micro contrôleur doit s'assurer de l'état de repos de la ligne en effectuant à plusieurs reprises des lectures de niveau haut sur la ligne de réception .

Une fois assuré de la présence du niveau haut, il attend un front descendant indiquant un bit start, et s'assure de sa validité en le retestant après une durée d'un demi bit afin d'exclure d'éventuels fluctuations intempestives.

Les 8 bits de données seront alors lus un à un en commençant par le bit de poids moindre en effectuant des échantillonnages à des intervalles de temps réguliers de largeur égale à la durée d'un bit .La donnée ainsi reconstituée sera sauvegardée sur une position mémoire.

Le bit optionnel de parité est lu de la même manière et informe le microcontrôleur de la véracité de la donnée reçue . Dans le cas d'une erreur ,le 68705 demandera au micro-ordinateur à travers les lignes de contrôle de réémettre la donnée en question.

Le test de la validité de la donnée se fait en calculant le nombre de un contenu dans la donnée reçu et en le comparant au bit de parité .

L'organigramme de la figure (6-4) montre la façon par laquelle le programme de réception de la donnée série se déroule .

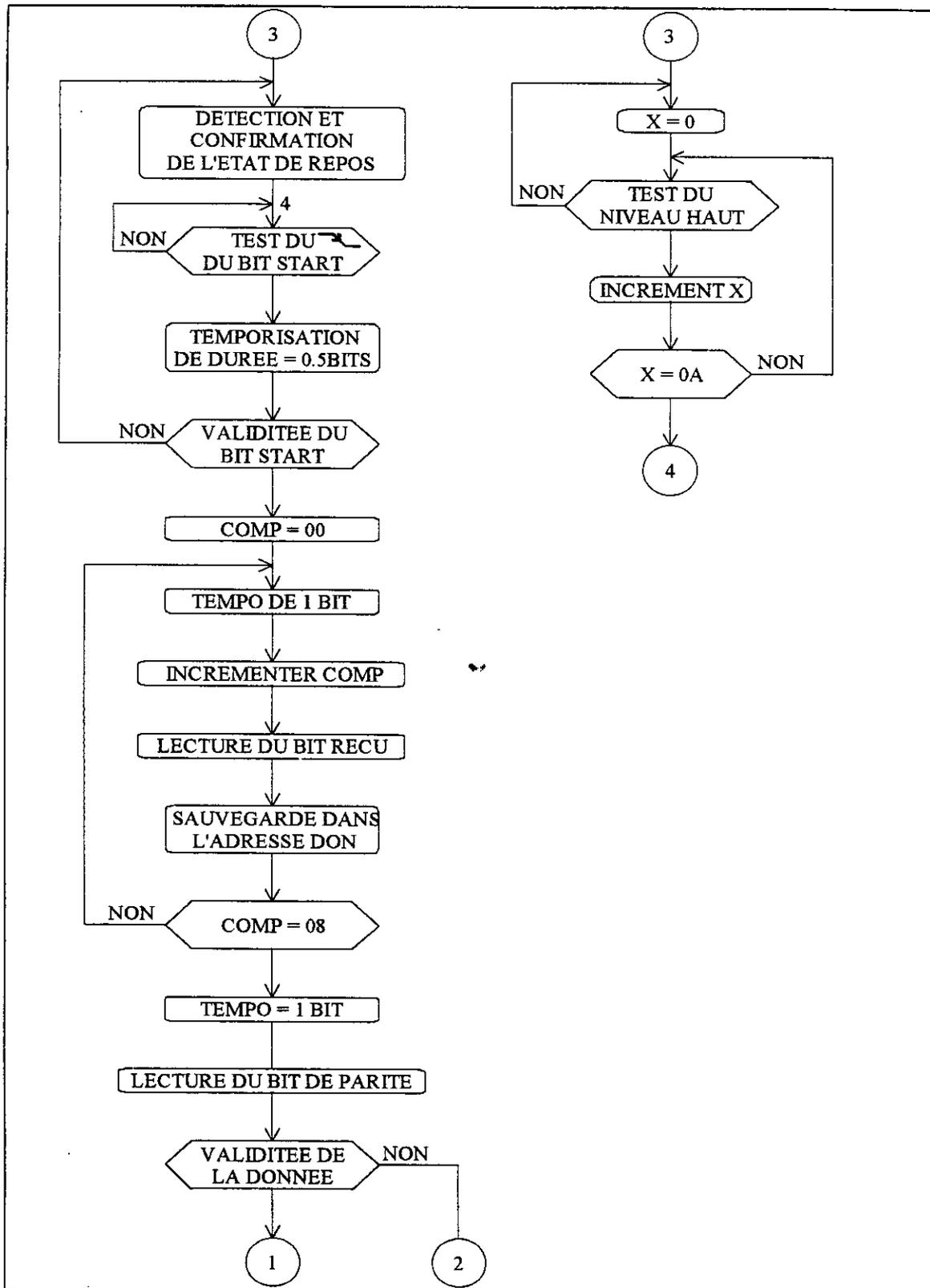


Figure 6-4 : Organigramme de la réception série des données .

7. PROGRAMME DE LA COMMANDE

Cette partie a pour objectif de matérialiser l'angle optimal calculé par le micro-ordinateur en effectuant le positionnement du panneau .

Avant de se mobiliser pour la commande du moteur ,le microcontrôleur commence par 'décompresser' la donnée qu'il aura préalablement reçu pour restituer les données utiles .Après quoi il effectuera des calculs lui permettant de délimiter la durée des régimes transitoires que nous expliquerons bien par la suite .Ceci fait ,il entamera la commande du moteur jusqu'à ce que le positionnement soit terminé . C'est alors que le microcontrôleur exécutera le programme de fin de positionnement qui se solde par un retour de l'interruption vers le programme principal .

Tout ceci se fait conformément aux étapes décrites par l'organigramme¹¹ de la figure (7-1).

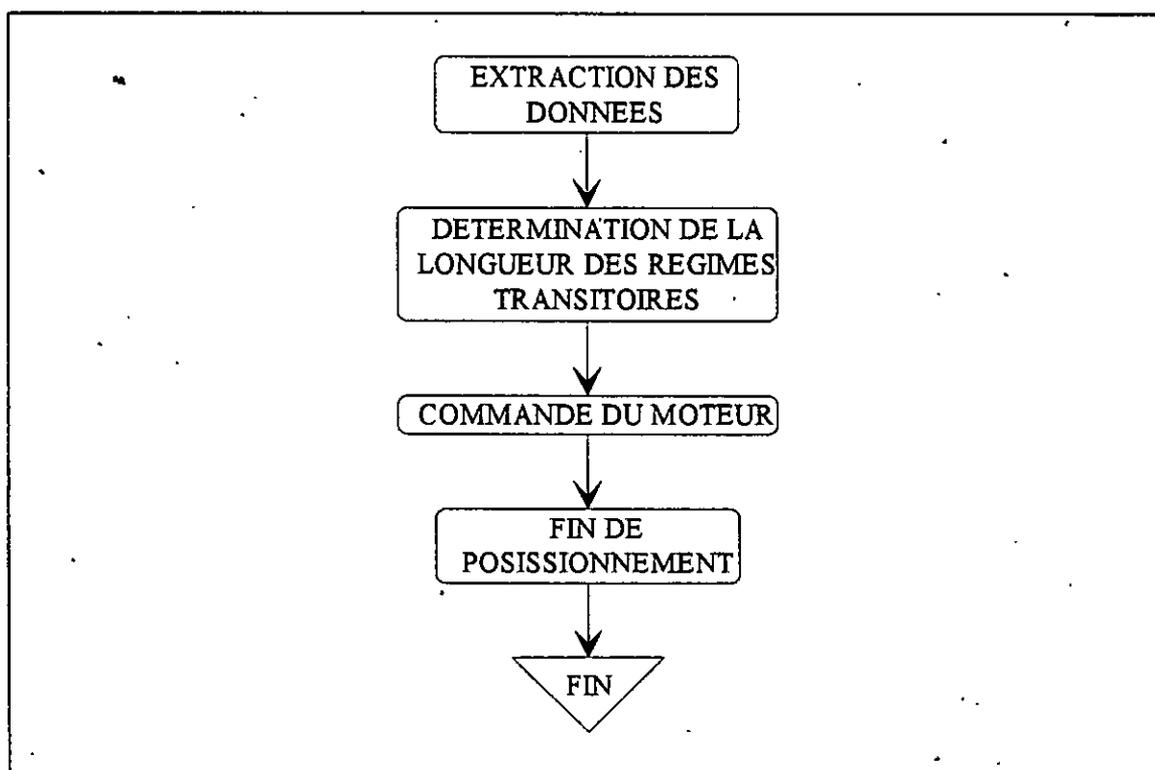


Figure 7-1 : Organigramme global du programme de commande .

7.1. EXTRACTION DE LA DONNEE

L'octet reçu par le microcontrôleur contient deux informations ,l'une concernant le sens de rotation et l'autre le nombre de pas à effectuer pour terminer le positionnement à un rapport près.

¹¹ les organigrammes et les explications de toutes les fonctions qui y figurent seront données dans les sections correspondantes .

La première chose à faire serait d'extraire les deux données et de reconstituer le nombre de pas exacte à exécuter en multipliant la donnée initiale par un facteur traduisant le rapport de démultiplication introduit par les engrenages du mécanisme de la monture. Le résultat du test du bit de signe sera déposé dans un position mémoire. Il servira par la suite pour le programme de commande.

L'organigramme de cette tâche est donné en figure (7-2).

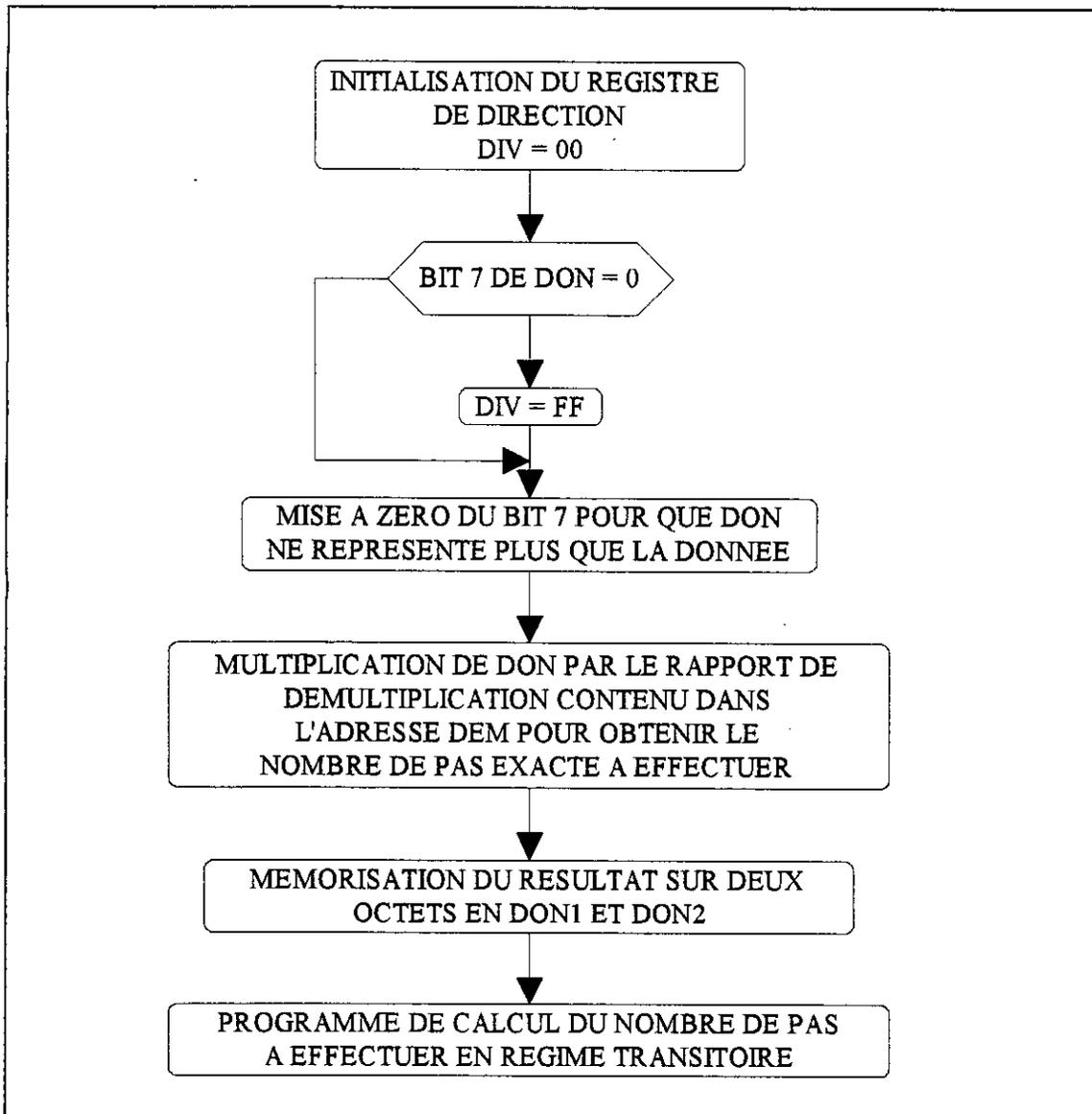


Figure 7-2 : Organigramme de l'extraction de la donnée.

7.2 COMMANDE DU MOTEUR

La rotation du moteur pas à pas s'effectue par une séquence de permutation circulaire des configurations d'excitation dans un sens ou dans l'autre. Les excitations actuelles peuvent être classées selon les modes suivants

- ♦ Mode d'excitation monophasé
- ♦ Mode d'excitation double phases
- ♦ Mode d'excitation à demi-pas

Pour assurer la commande du moteur, notre choix s'est posé sur le mode monophasé. Ce mode offre la plus faible consommation de courant pour des performances à peu près égales aux autres modes. La figure(7-3) montre le chronogramme de ce mode.

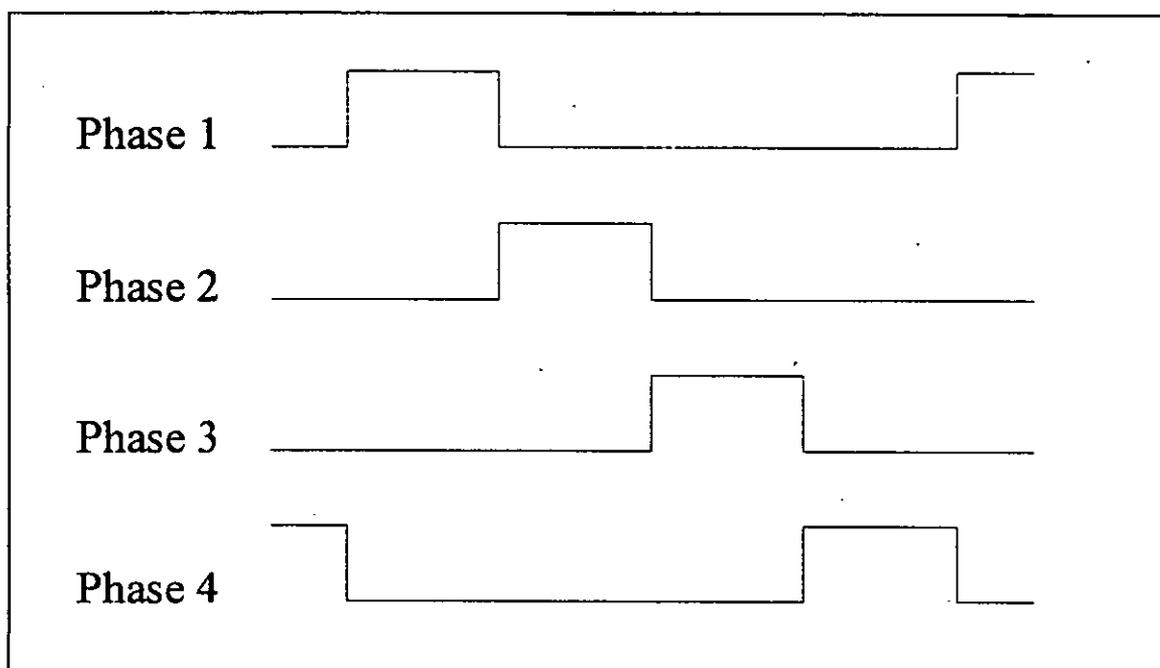


Figure 7-3 : Chronogramme du mode d'excitation monophasé.

Les phases du moteur sont activées par un niveau bas appliqué par les quatre premières lignes du port B.

Il faut tenir compte des lignes PB6 et PB7 qui doivent être mises respectivement à zéro et à un, pour éviter des consommations de courant et des risques de détérioration des composants ; le micro-ordinateur émettant un niveau haut à travers la ligne de transmission et un niveau bas via sa ligne DTE.

Ainsi nous retiendrons quatre configurations¹² d'octets d'attaque de l'étage de puissance comme indiqué dans le tableau suivant :

Tableau : 8

Ligne	Phase	configuration	Rang (X)	emplacement mémoire
PB0	Phase1	PH1 = 10011110	0	40h
PB1	Phase2	PH2 = 10111101	1	41h
PB2	Phase3	PH3 = 10111011	2	42h
PB3	Phase4	PH4 = 10110111	3	43h

La commande du moteur commence par la récupération dans le registre index du rang de la dernière configuration d'excitation des phases .

La boucle d'excitation du moteur débute par une temporisation permettant d'atteindre la largeur d'impulsion de base correspondante à la vitesse maximale ,la détection du sens de rotation permet d'incrémenter ou de décrémenter le registre index tout en testant la fin du cycle d'excitation ,pour pouvoir exciter le moteur avec la configuration approprié .Ceci fait ,une temporisation complémentaire permettant de faire varier la vitesse est exécutée et la donnée est décrémentée à chaque pas exécuté jusqu'à ce qu'elle s'annule indiquant la fin de positionnement .

¹²Le mode d'excitation peut être changé simplement en changeant la configuration de ces octets la .

L'organigramme de commande est donné en figure (7-4).

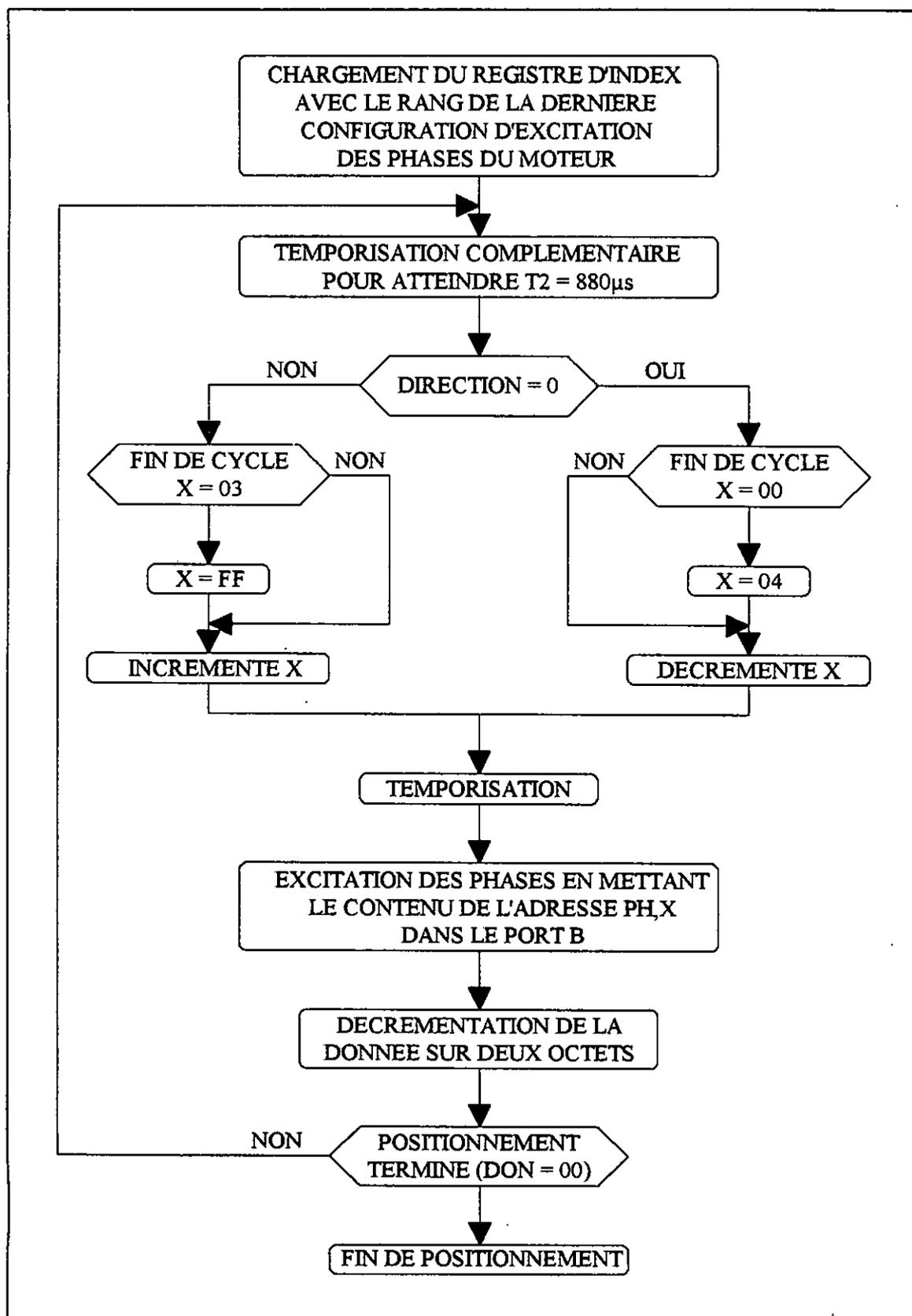


Figure 7-4 : Organigramme de la commande du moteur

7.3 ASSERVISSEMENT DE LA VITESSE DU MOTEUR

Afin de tenir compte des contraintes mécaniques le positionnement devra être fait en trois phases distinctes.

En effet, l'inertie élevée du panneau impose un démarrage à une faible vitesse, suivi d'une accélération jusqu'à atteindre un régime permanent caractérisé par une vitesse de rotation constante du moteur. A l'approche du point assigné la vitesse décroît jusqu'à atteindre celle du démarrage, alors le moteur s'arrêtera au point indiqué. La figure (7-5) illustre bien cela.

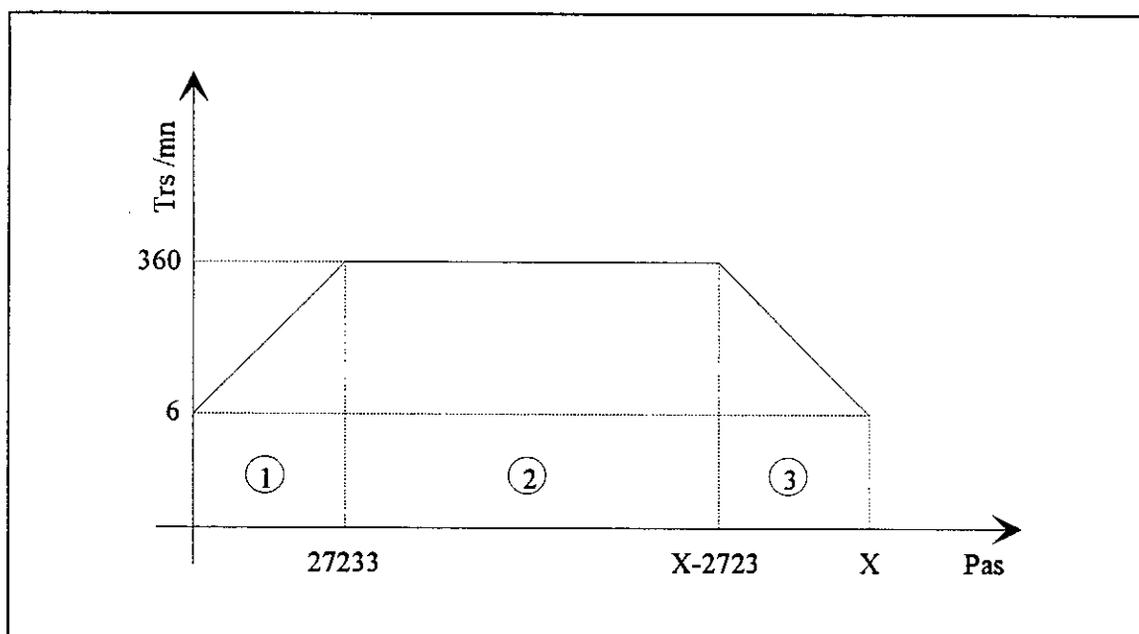


Figure 7-5 Allure de la vitesse du moteur en fonction des pas effectués .

La vitesse de rotation du moteur est liée à la fréquence d'excitation des phases par la relation

$$V = 60 / (S * T) \quad [\text{trs/mn}]$$

Où T est la durée d'excitation d'une phase du moteur exprimée en secondes

S est le nombre de pas par tour (pour le moteur utilisé S=100)

Il apparaît clairement de cette relation que la variation de la vitesse peut s'obtenir aisément en changeant la largeur des impulsions d'attaque des différentes phases .

♦ Phase (1)

C'est la phase accélération durant laquelle le moteur démarre avec une vitesse initiale faible pour atteindre une vitesse de régime établi . Ce passage se fait en diminuant la largeur de l'impulsion d'excitation de la phase .

Les vitesses de démarrage et de régime établi V_1 et V_2 respectivement correspondantes aux largeurs d'impulsions T_1, T_2 sont fixées à :

$$V_1 = 6 \text{ trs / mn} \Rightarrow T_1 = 25.5 \text{ ms}$$

$$V_2 = 3 \text{ trs / s} \Rightarrow T_2 = 880 \mu\text{s}$$

La durée de cette phase est aussi fixée à $D = 3 \text{ s}$

Une idée est de partager la différence ($T_1 - T_2$) en N parties égales à une durée unitaire K . Ainsi le passage de V_1 à V_2 s'effectue en diminuant à T_1 à chaque cycle une durée unitaire jusqu'à atteindre T_2 . d'où

$$T_2 = T_1 + N \cdot K \quad (1)$$

D'autre part on a

$$D = N \cdot T_2 + K \cdot N \cdot (N + 1) / 2 \quad (2)$$

A partir de ces deux équations, on obtient $N = 224$ $K = 110 \mu\text{s}$

$$T_1 = 232 \cdot K \quad T_2 = 8 \cdot K$$

Le calcul de la largeur de l'impulsion complémentaire ($T_i = I * K$) relative à cette phase est bien expliqué dans l'organigramme de la figure (7-6).

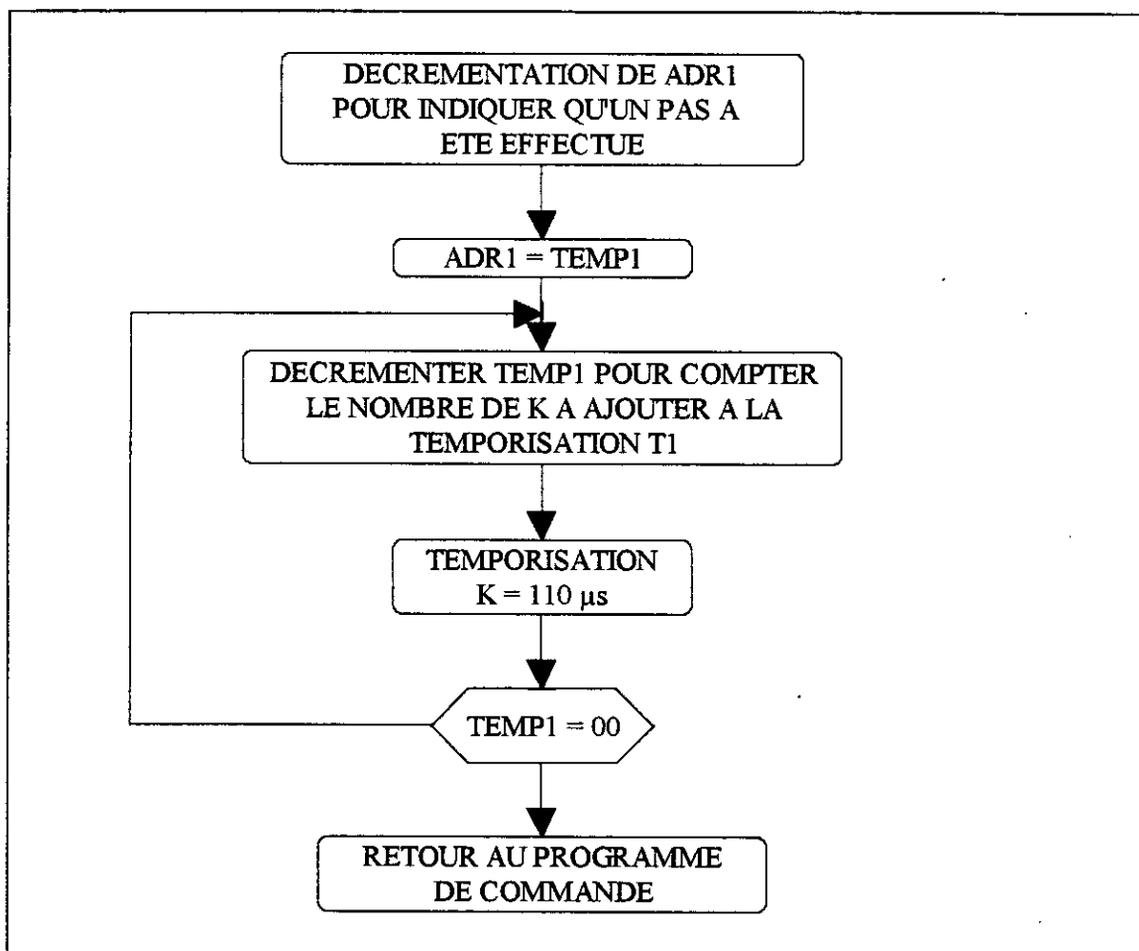


Figure 7-6 : Organigramme de calcul de la tempo durant la phase d'accélération.

♦ phase (2)

Durant cette phase le panneau est entraîné à une vitesse constante égale à celle atteinte en fin de phase accélération. Ce régime dure plus ou moins longtemps suivant la valeur de la donnée.

♦ Phase (3)

La phase de décélération demeure de durée égale à celle de la phase (1). Elle débute à l'approche de la position consignée pour ce terminer exactement au point d'arrêt.

Le calcul des temporisations T_i est décrit par l'organigramme de la figure (7-7).

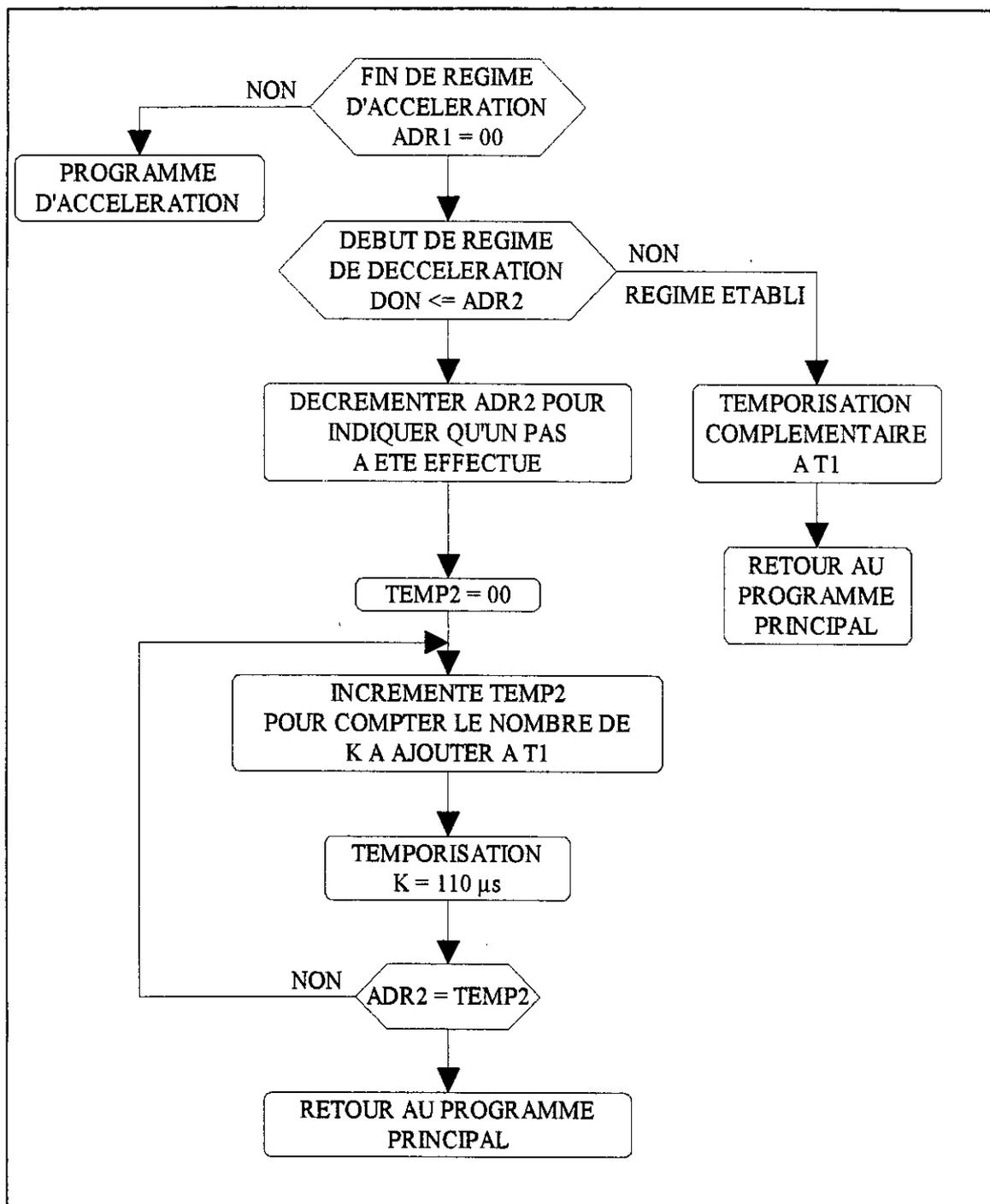


Figure 7-7 : Organigramme de la temporisation .

Il est à remarquer que pour des valeurs faibles de la donnée ,le moteur n'aurait pas le temps d'achever le régime accélération qu'il doit déjà s'arrêter .Pour résoudre ce problème une solution serait d'effectuer une accélération suivie immédiatement d'une décélération de durées égales à la moitié de la donnée sans pour autant passer par la phase (2) à vitesse constante ,voir figure (7-8) .

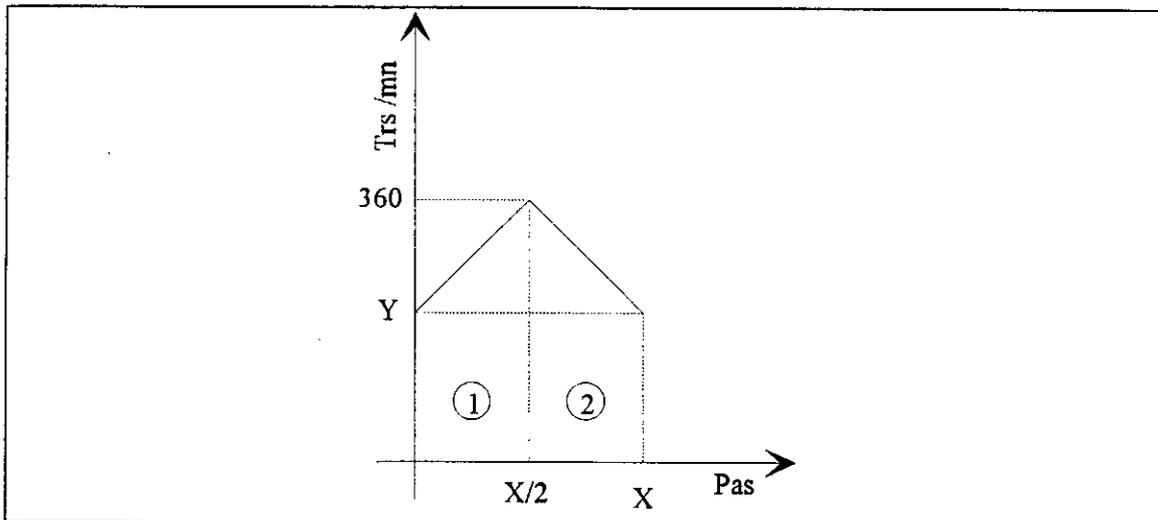


Figure 7-8: Allure de la vitesse pour un déplacement de faible valeur .

L'organigramme de la figure (7-9) décrit la façon avec laquelle est effectué le choix de la durée de la phase d'accélération et de décélération .

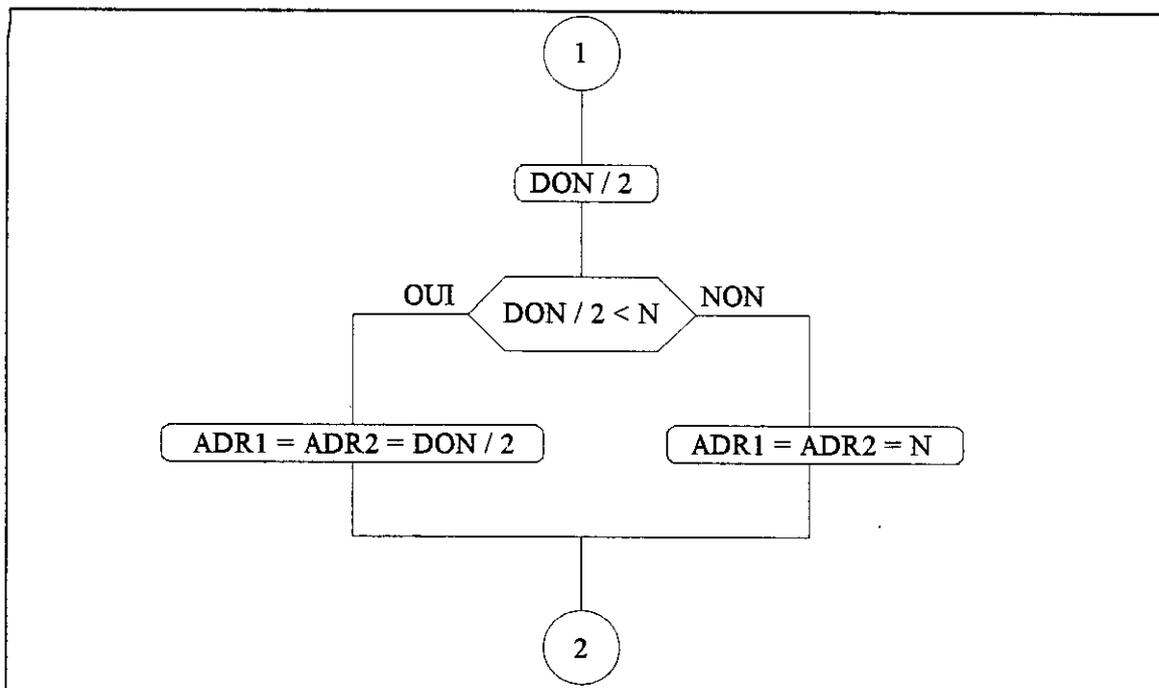


Figure 7-9 : Organigramme de sélection d'ADR

7.4 FIN DE POSITIONNEMENT

Une fois le panneau positionné, la dernière configuration d'excitation des phases devra être maintenue pendant un laps de temps pour assurer l'arrêt total du moteur, puis son rang sera sauvegardé en mémoire, pour éviter de perdre des pas au prochain positionnement.

Les phases du moteur seront alors mises au repos par l'écriture d'un BF sur le port B.

L'organigramme de la figure (7-10) traite de cette partie.

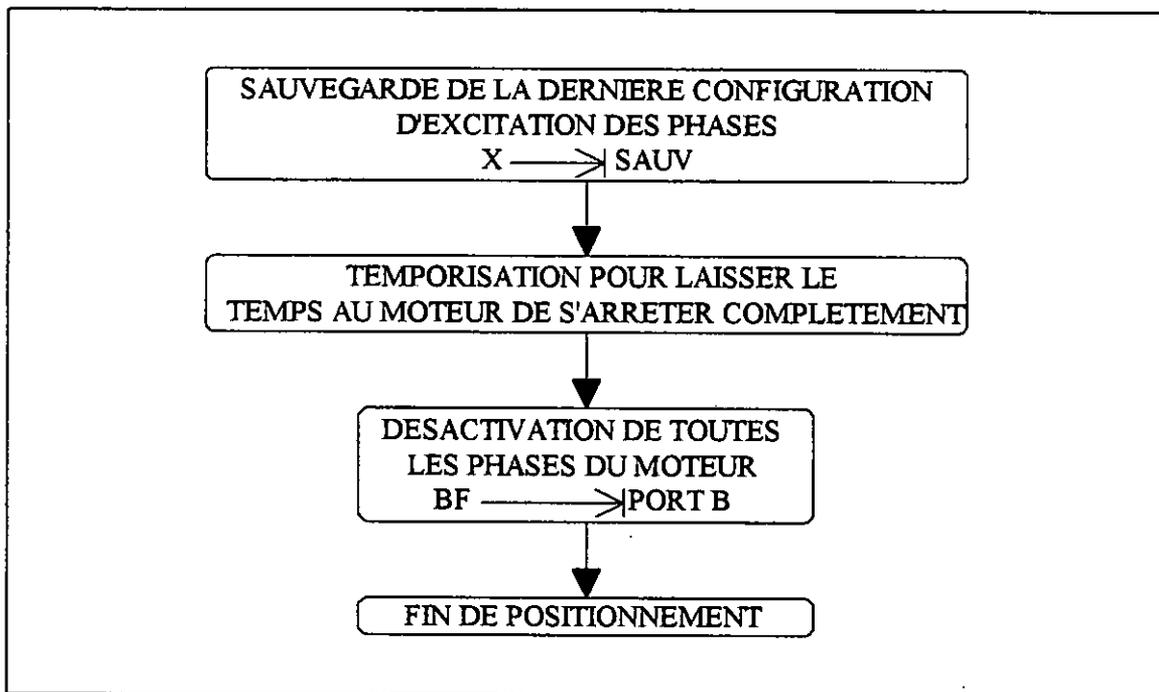


Figure 7-10 : Organigramme de fin de communication

8. EVOLUTION DU SYSTEME

Dans le système que nous avons étudié et réalisé ,le positionnement s'effectuait grâce à un seul moteur ,de plus la communication entre le micro-ordinateur et le microcontrôleur se faisait à une vitesse fixée à 1200 BAUDS .

Une évolution intéressante de notre système serait de rendre son utilisation possible dans un processus industriel basé sur la commande de mouvements d'un outil quelconque dont la trajectoire est préalablement calculée par le micro-ordinateur .

Un exemple d'un tel processus serait la commande du découpage de tôles suivant des formes precalculées ,ou celle d'un bras manipulateur effectuant des points de soudures .

Pour un maximum d'adaptabilité ,notre système devra être capable de fonctionner avec plusieurs cadences de transfert ,et piloter plusieurs moteurs .

8.1 SELECTION DU PROTOCOLE

Le protocole n'étant pas prédéfini sa sélection devra se faire de façon logicielle par l'opérateur .Ainsi le micro-ordinateur imposera la cadence de communication et le microcontrôleur devra s'y adapter .

Pour ce faire ,le P.C. commencera la séquence de transmission par l'envoi d'un octet de reconnaissance qui permettra au microcontrôleur de déceler la vitesse à laquelle elle s'effectue .

Le format de cet octet est donnée en figure (8-1) .

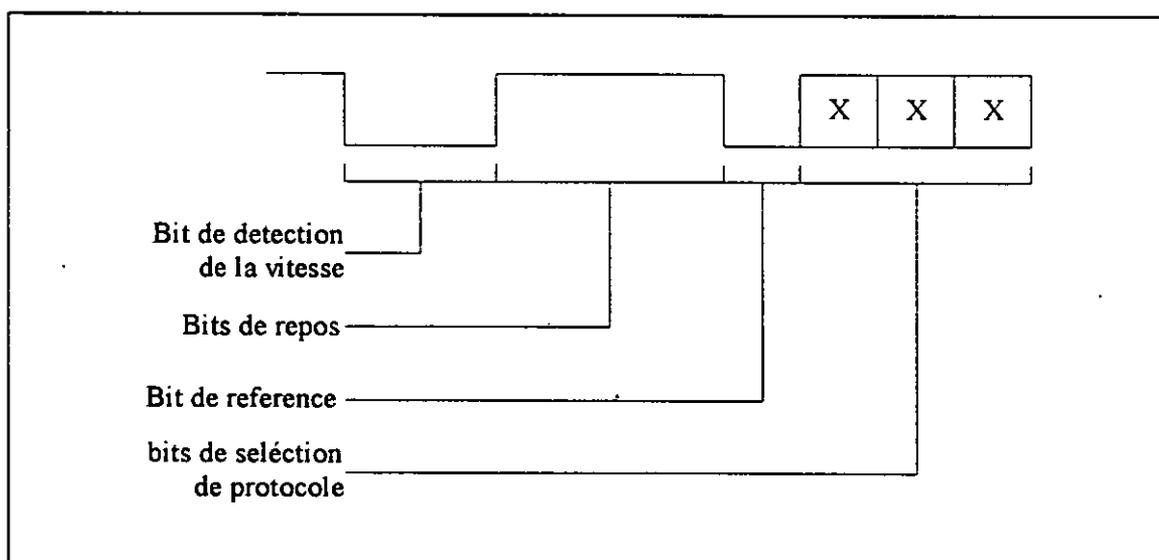


Figure 8-1 : Format de l'octet de reconnaissance .

Le bit 0 est forcé à zéro pour permettre ,conjugué au bit start de mesurer la vitesse de transmission .

Le bit 1 mis à l'état haut permet de délimiter la zone sur laquelle le calcul de vitesse est fait .

Les bits 2 et 3 mis à un servent à laisser le temps au microcontrôleur de terminer toutes les mises aux points .

Le bit 4 mis à zéro ,sert de référence pour l'échantillonnage des trois bits suivants .

Les bits 5,6 et 7 comportent des renseignements sur le protocole de communication adopté .

Tableau 9 : Description des 3 bit de sélection de protocole .

Bit 7	Bit 6	Bit 5	Protocole
0	X	0	Un bit stop + pas de parité
0	X	1	Deux bits stop + pas de parité
1	0	0	Un bit stop + parité paire
1	0	1	Deux bits stop + parité paire
1	1	0	Un bit stop + parité impaire
1	1	1	Deux bits stop + parité impaire

Le choix de cette configuration n'est pas fortuit .Il est fruit d'un compromis entre différentes contraintes que nous allons détailler par la suite .

L'organigramme du fonctionnement global est donné en figure (8-2)

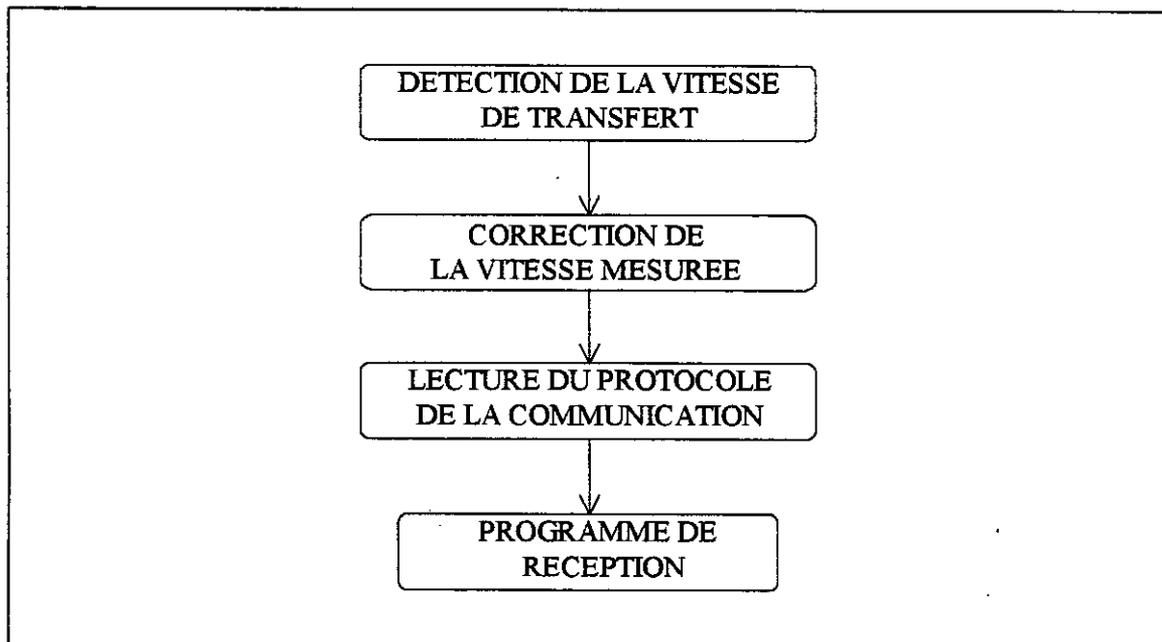


Figure 8-2 : Organigramme du fonctionnement global de la détection du protocole .

8.1.1 DETECTION DE LA VITESSE

La détection de la vitesse se fait en mesurant la largeur de l'impulsion comprise entre le front descendant du bit start et le front montant du bit 1.

L'idée est donc d'incrémenter un compteur d'une unité dès détection du front descendant du bit start et ceci tant que la ligne reste au niveau bas pour indiquer qu'un laps de temps connu a écoulé.

Il est à remarquer que toutes les vitesses de transferts normalisées ont pour plus grand diviseur commun le nombre 16.

L'optimum serait d'utiliser une boucle de test et comptage de durée égale au plus faible diviseur commun ce qui réduirait les incertitudes sur les valeurs mesurées. Malheureusement le nombre de cycle minimum que requière l'exécution de telles instructions est de 14 cycles, ce qui nous contraint à utiliser une boucle de 16 cycles.

La valeur du compteur après le front montant du bit 1 de l'octet de reconnaissance indiquera un rapport de la vitesse de transfert avec une certaine incertitude que l'on devra éliminer.

L'organigramme de la figure (8-3) décrit le fonctionnement.

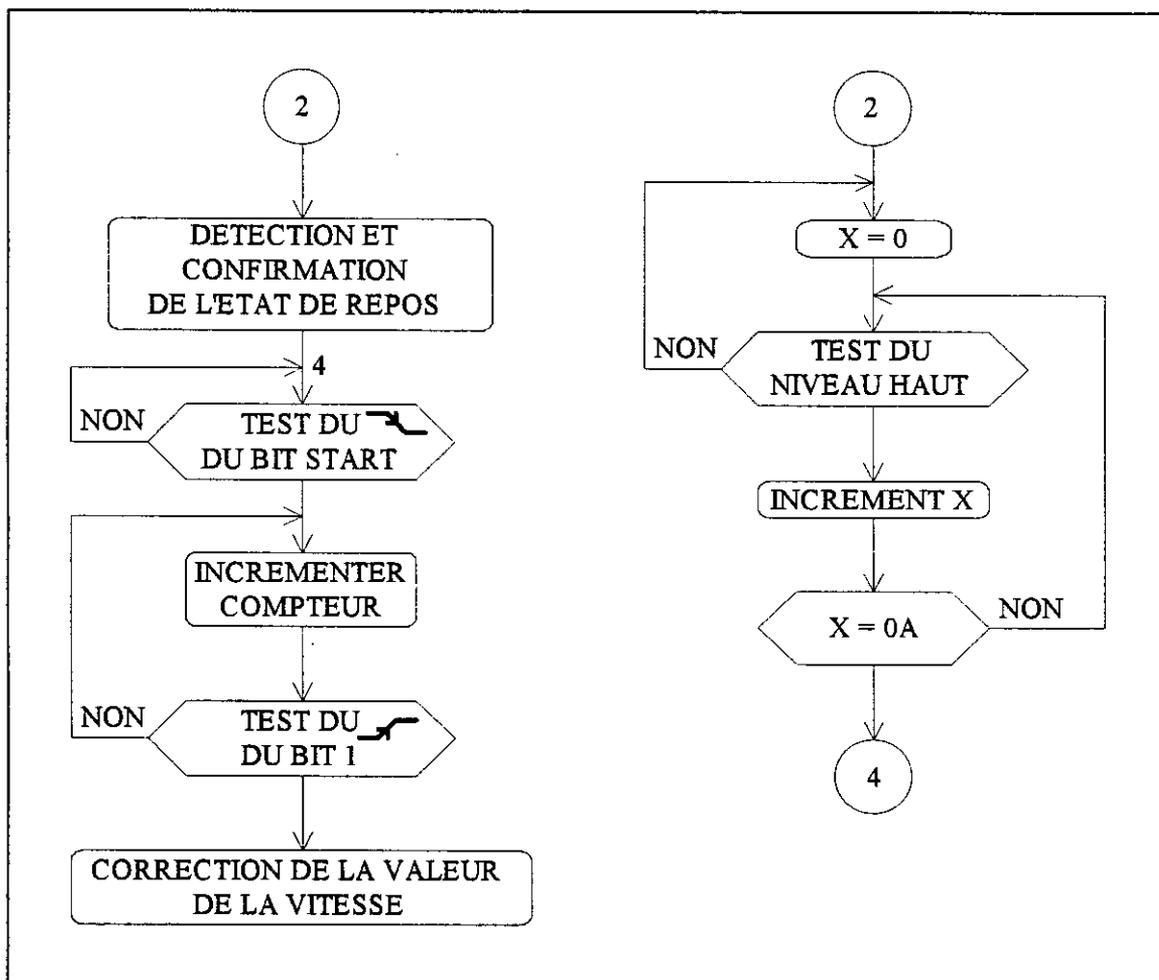


Figure 8-3 : Organigramme de détection de la vitesse de transfert.

8.1.1.1 TRAITEMENT DE L'INCERTITUDE

Le moment auquel le microcontrôleur effectue le test peut ne pas coïncider avec celui de l'arrivée du front descendant du bit start car celle-ci peut se produire à n'importe quel moment de la boucle de test de l'état de la ligne dont la durée est égale à 10 cycles. Ceci introduit une incertitude Δ_1 sur la mesure qui peut varier de 0 à 10 cycles.

Il en est de même pour le front montant où l'incertitude Δ_2 peut varier de 0 à 16 cycles.

La figure (8-4) décrit cela.

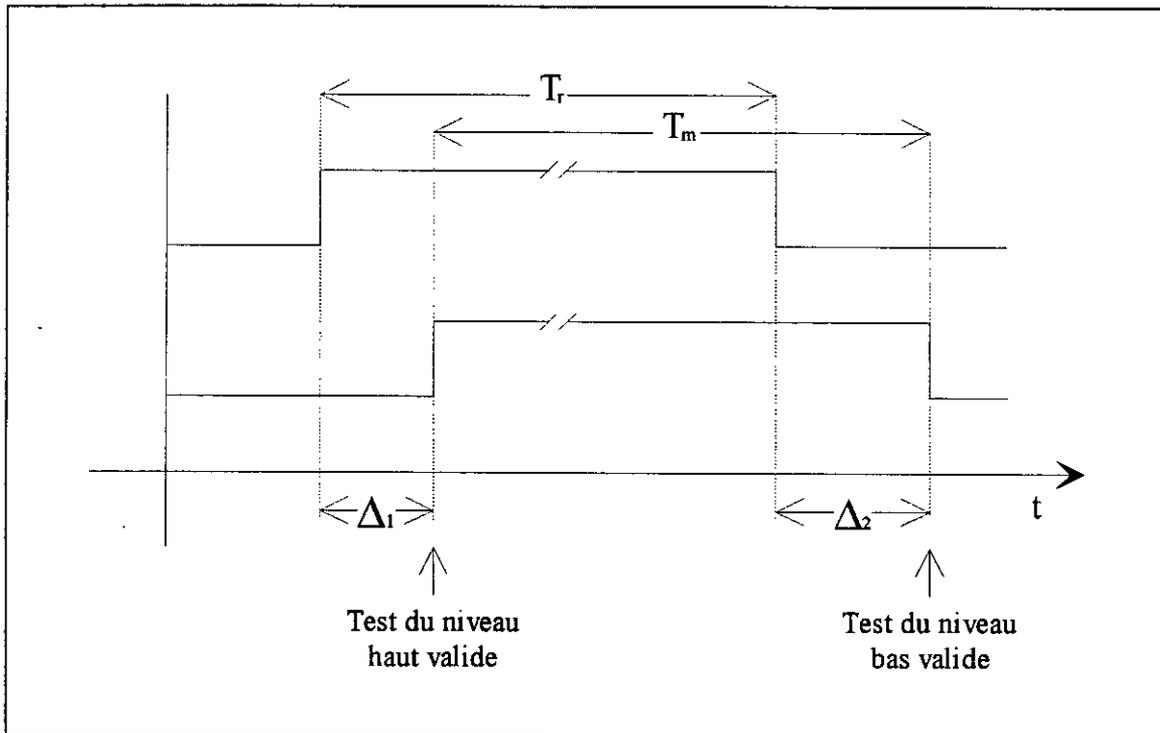


Figure 8-4 : Chronogrammes des temps réels et mesuré.

- $T_r = \Delta_1 + T_m - \Delta_2$
- $\Delta_1 \in [0, 10]$
- $\Delta_2 \in [0, 16]$
- $T_m = X_m * 16$
- $T_r = X_r * 16$

Où : X_r est le rapport réel de la vitesse par 8

X_m est la rapport mesuré

T_r est la durée réelle de deux bits

T_m est la durée mesurée

Pour traiter l'incertitude il faudrait prendre en considération les cas extrêmes .

$$Tr = Tm + 10 \quad (1)$$

$$Tr = Tm - 16 \quad (2)$$

compte tenu de (1) et (2) :

$$Tr \leq Tm - 16 \Rightarrow Xr \leq Xm - 1$$

On obtient la correction sur l'incertitude en posant :

$$Xm = Xm - 1$$

puis on arrondissant la valeur obtenue à celle du rapport réel immédiatement supérieur conformément au tableau (10).

Tableau (10)¹⁴

Vitesse (BAUDS)	Nombre de cycles	Rapport
19200	48	8*6
9600	96	8*12
7200	128	8*16
4800	192	8*24
3600	256	8*32
2400	384	8*48
1800	512	8*64
1200	768	8*96
600	1536	8*192

8.1.1.2 CONTRAINTES LIEES AU CHOIX DES VITESSES

Le choix de ces vitesses est directement lié au choix du nombre de bits de l'octet de reconnaissance sur lesquels s'effectue la mesure de l'état bas .

Pour les vitesses élevées le problème qui se pose est celui du temps accordé au comptage .Si on utilisait un seul bit pour la mesure ,l'incertitude serait du même ordre de grandeur que la mesure elle même ,ce qui rendrait son traitement impossible.

On pourrait penser à première vue que si on étendait la mesure sur plusieurs bits cela résoudrait le problème .En fait il n'en est rien car pour les vitesses faibles la durée d'un

¹⁴ Ces calculs ont été fait sur la base d'un quartz de fréquence égale à 3.6864 Mhz .

bit en nombre de boucles comptée devient vite très importante (supérieure à 255) ce qui imposerait l'utilisation de deux octets pour le comptage. Cela nous conduit à omettre ces vitesses là, si non la boucle élémentaire de comptage faite sur deux octets, deviendrait de durée supérieure à 16 chose qui n'étendrait pas l'usage à des vitesses plus élevées.

Tout ceci nous a orienté vers l'utilisation de la configuration de l'octet de reconnaissance dépeint en figure (8-1).

8.1.2 LECTURE DU PROTOCOLE

Une fois que la vitesse de transmission exacte est connue, le microcontrôleur devra prendre connaissance du reste du protocole à savoir l'existence de la parité et sa nature de même que le nombre de bit d'arrêt.

Ceci est réalisé en effectuant la lecture des bits 5,6 et 7 de l'octet de reconnaissance en prenant comme référence temporelle le front descendant du bit 4.

L'organigramme la lecture du protocole est donné en figure(8-5)

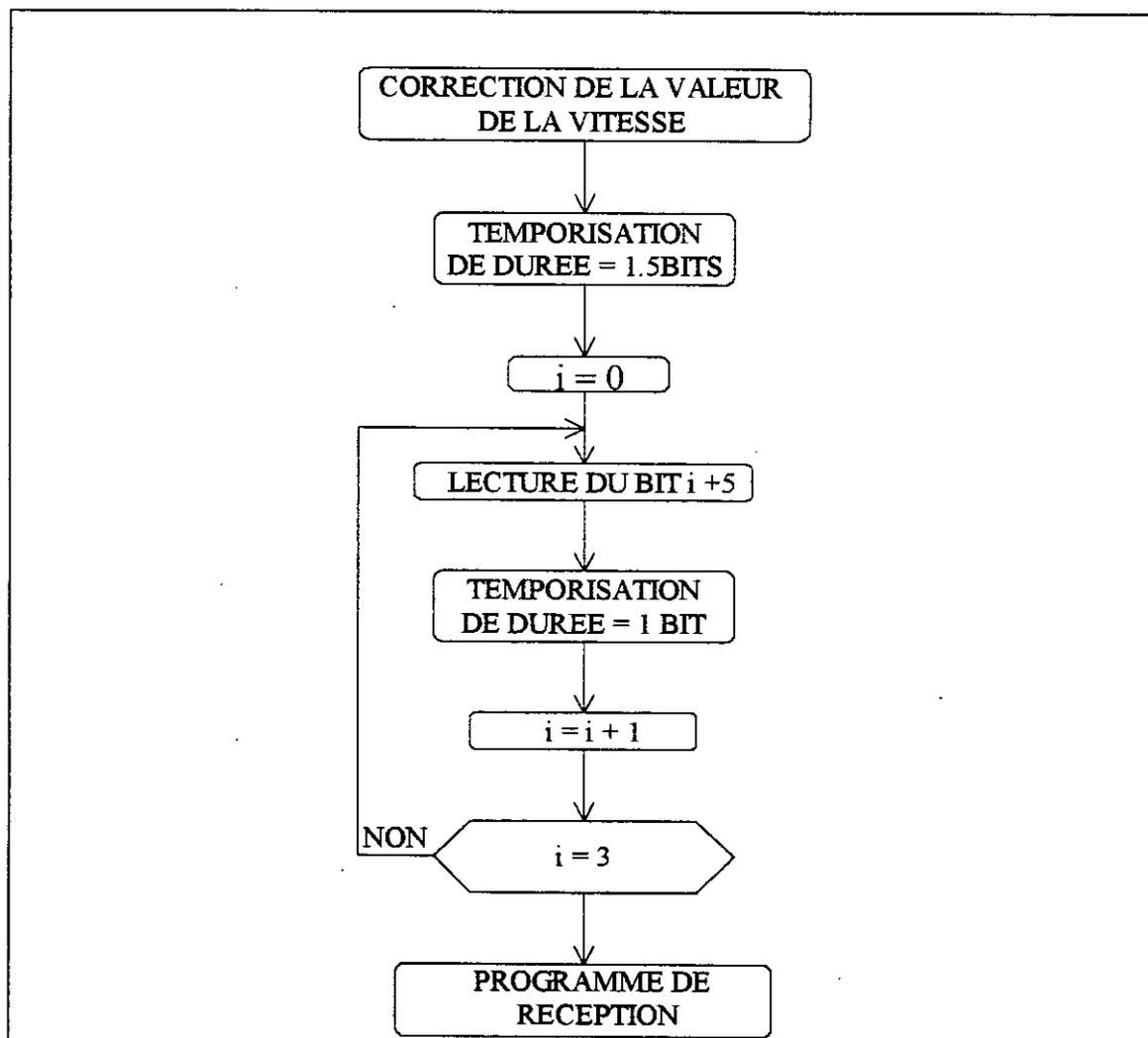


Figure 8-5 : Organigramme de lecture du protocole.

Il est clair qu'avec un protocole non prédefini, le programme de détection d'erreur va changer. Effectivement il faudra tenir compte de l'existence ou non de la parité ainsi que de sa nature. Un organigramme détaillant cette fonction est rapporté en figure (8-6).

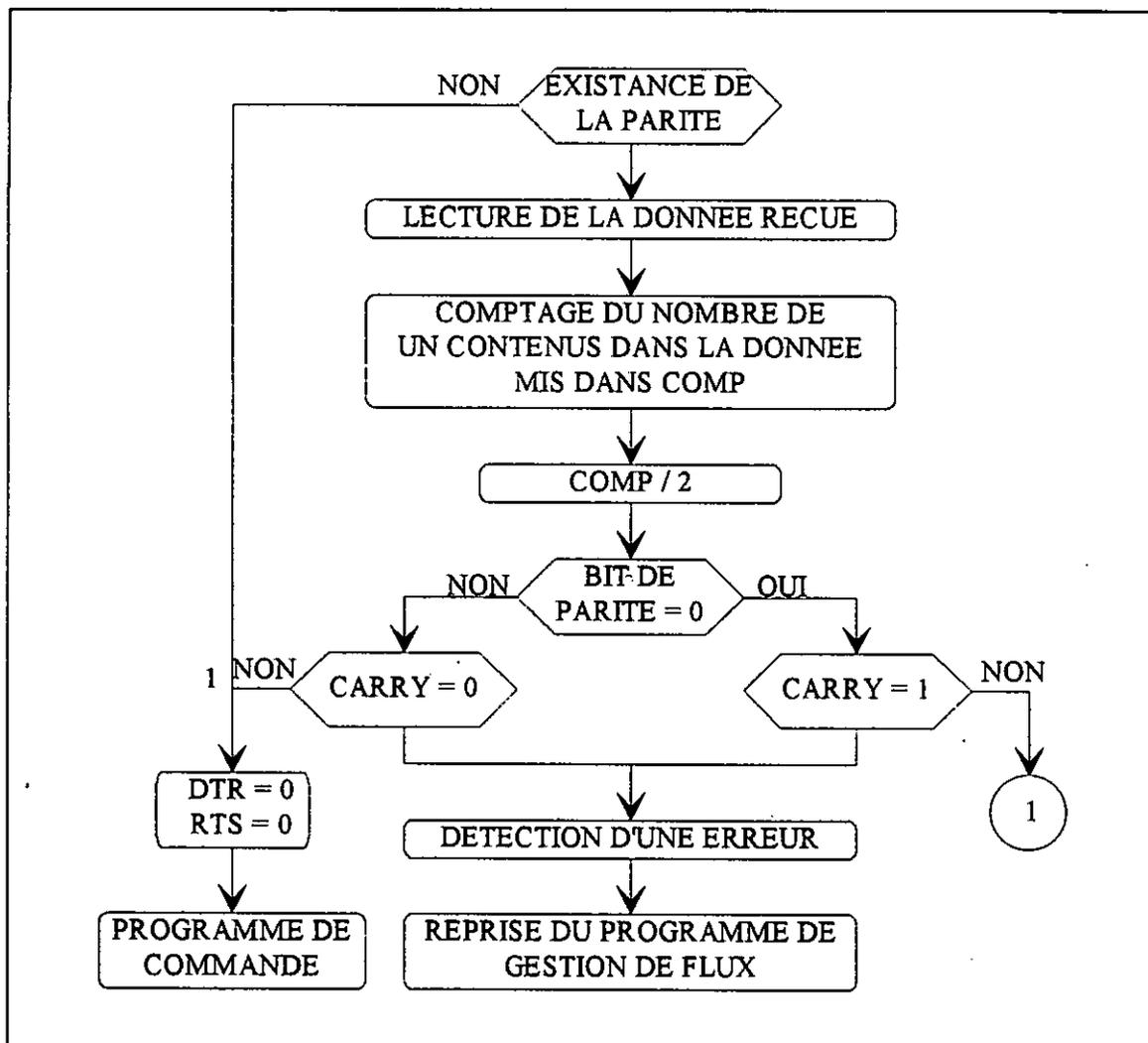


Figure 8-6 : Organigramme de détection d'erreur.

8.2 EXTENSION DE LA COMMANDE A PLUSIEURS MOTEURS

Dans la majorité des applications la commande de 8 moteurs est rarement atteinte. Pour cela et pour des raisons pratiques nous avons préféré restreindre l'extension à ce nombre.

Les quatre lignes du port B attaquent simultanément 8 cartes de puissance différentes. Sur chacune de ces cartes est adjoint un buffer qui sera sélectionné par le microcontrôleur à travers la ligne du port A correspondante. Ainsi l'ordre de déplacement issu du micro-ordinateur sera destiné à un ou plusieurs moteurs ce qui autorise une commande simultanée.

La figure (8-7) montre la constitution du système.

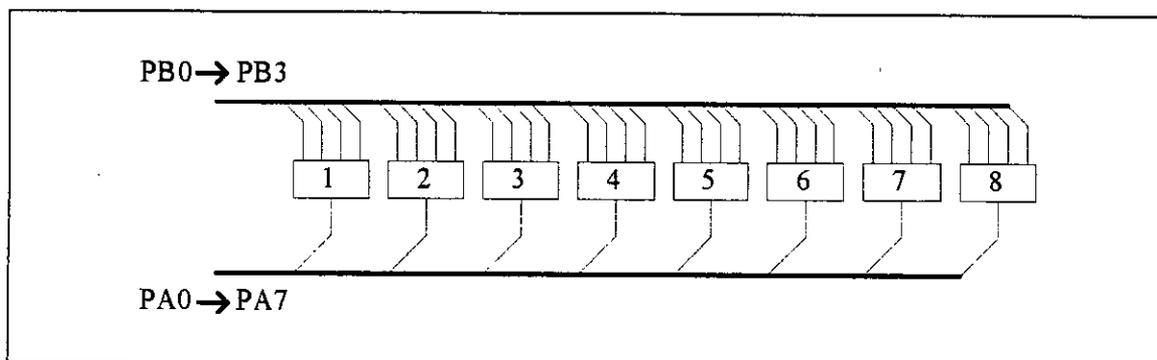


Figure 8-7 : Organisation d'un système à 8 moteurs .

A cette fin tout envoi de donnée indiquant un déplacement sera précédé par l'envoi d'un octet permettant la sélection des cartes de puissances aux quelles l'information est destinée .

La lecture de cet octet se fait de la même manière que celle d'une donnée .

Pour sélectionner les moteurs 3 et 6 ,cet octet aura pour valeur 24h .

8.3 FONCTIONNEMENT

Ce nouveau système est capable de fonctionner avec n'importe quels programmes pour peu que ceux là respectes les démarches décrites dans l'organigramme de la figure (8-8).

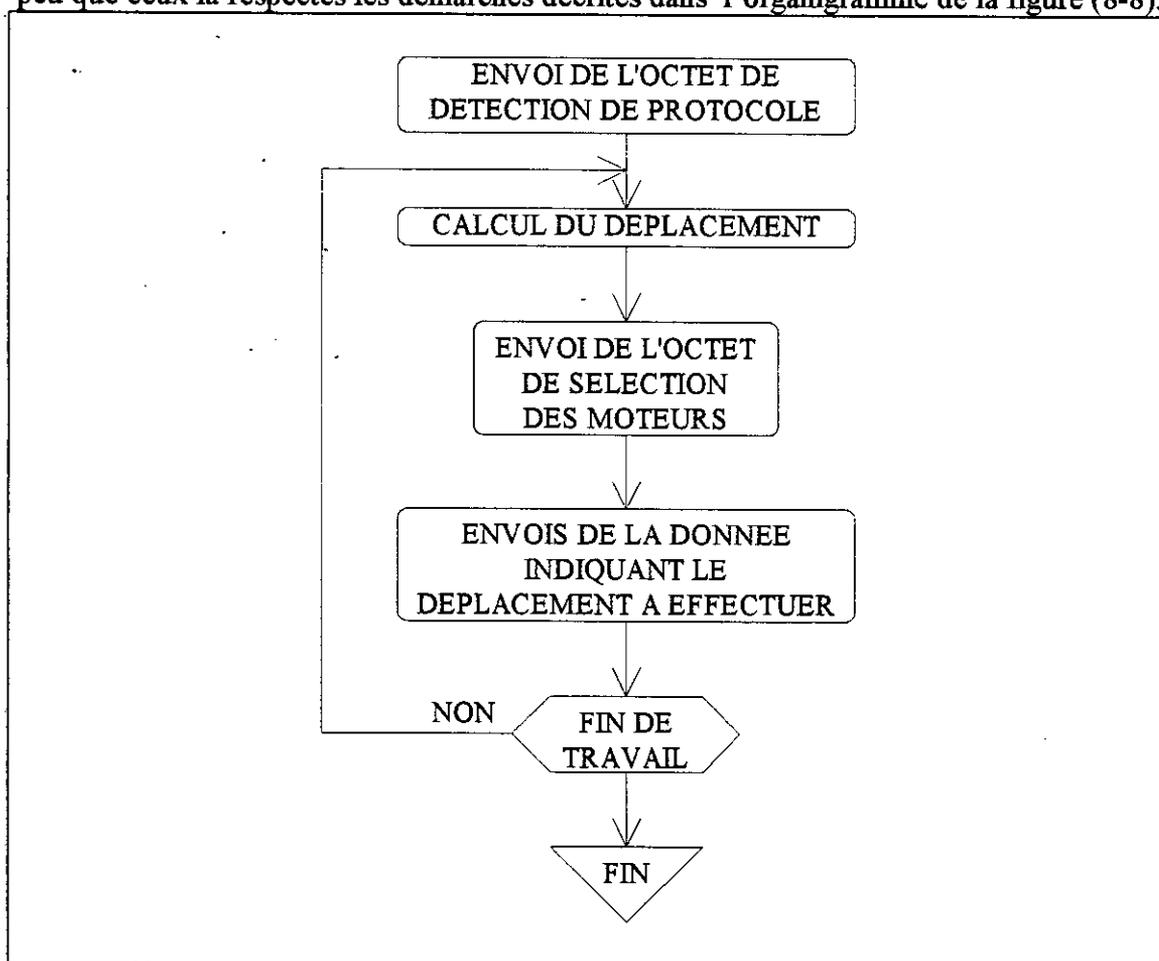


Figure 8-8 : Organisation du programme du micro-ordinateur .

L'Organigramme de fonctionnement du microcontrôleur est donné en figure(8-9)

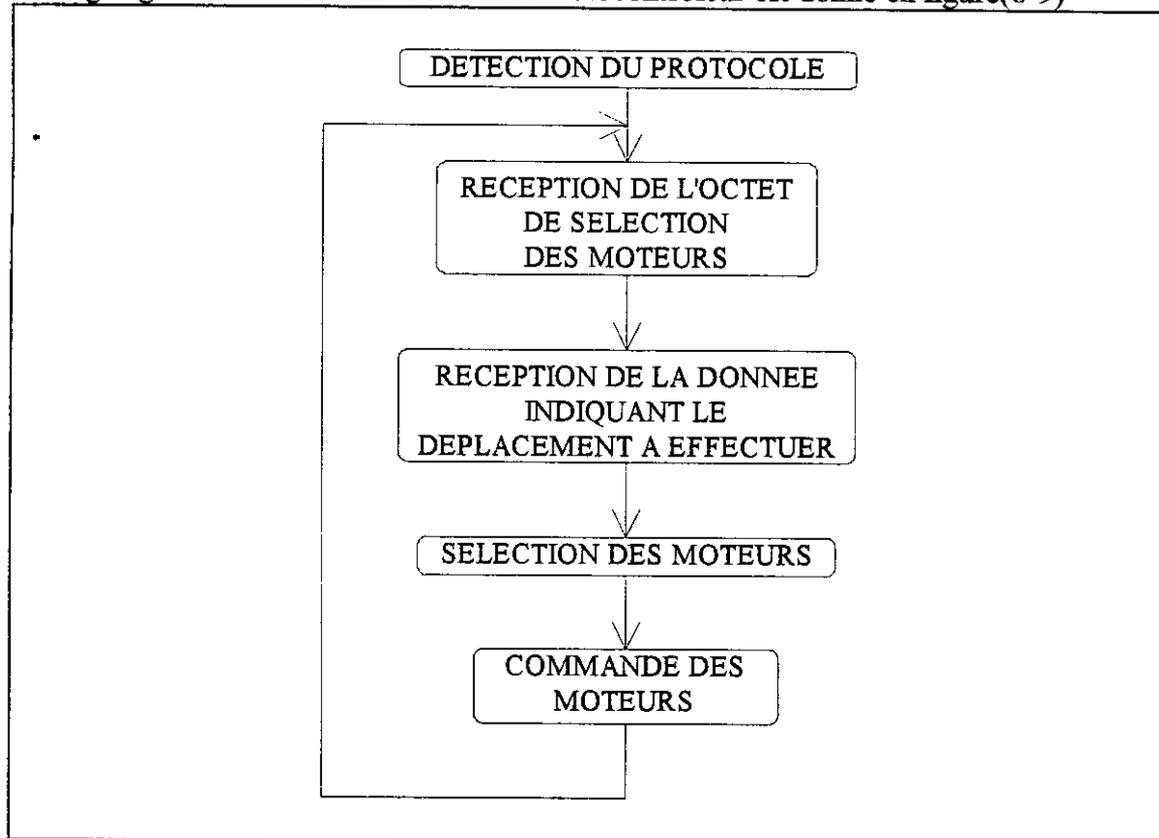


Figure 8-9 : Organigramme de fonctionnement général du microcontrôleur .

Il est important de spécifier que la gestion du flux ainsi que la détection d'erreurs sont omniprésentes dans le fonctionnement des deux communicants de la même manière que dans le système d'orientation du panneau ,ceci même s'ils n'apparaissent pas dans les organigrammes que l'on vient de présenter .

CONCLUSION

La recherche d'une énergie renouvelable propre et capable de concurrencer les autres énergies classiques est devenue une nécessité.

Cette recherche se heurte cependant à une double difficulté, d'ordre technique et d'ordre économique.

Au plan technique, il s'agira d'optimiser la génération photovoltaïque, dont l'optimisation de la captation du rayonnement solaire, est un élément assez déterminant.

Nous avons étudié un système semi statique qui permet d'obtenir un flux maximal, en utilisant un minimum d'énergie pour son positionnement. Ceci est un compromis entre la consommation et la production d'énergie et vraisemblablement une étape dans la recherche de la rentabilité de ce procédé. Le système étudié pourrait être associé sans difficulté à une centrale à tour qui est déjà à des rendements appréciables et qui probablement atteindra le mieux des niveaux de rentabilités comparables aux autres procédés de production.

Au plan économique, si aujourd'hui les coûts de revient n'autorisent pas la comparaison avec les autres formes classiques, notamment à cause des dimensions utiles des installations ainsi que des problèmes de stockage, les recherches allant dans le sens de la maximisation de leur rendements, d'une part et l'augmentation des coûts de cession des énergies classiques liée à leur rareté progressive, d'autre part, rendra l'utilisation de l'énergie solaire de plus en plus concurrentielle.

Au delà de ces considérations, il nous est apparu l'intérêt de donner à notre système un caractère polyvalent lui permettant de s'adapter à de multiples applications notamment dans le domaine industriel.

ANNEXES

Programme de gestion du flux .

```

          CLR    $ 02    DTR ET RTS MIS A ZERO
BCL1     LDA    $ 01
          ANDA   #$ 40    TEST DE DSR=1
          BEQ    BCL1
          LDA    #$ F3
          STA    $ 02    MISE DE DTR ET RTS A 1

```

Programme de détection de la vitesse .

```

          CLR    $ 45    INITIALISATION DES POSITIONS MEMOIRES
          CLR    $ 42    QUI SERONT UTILISEES POUR LE PROGRAMME .
          CLR    $ 05    PORT B EN ENTREE
BCL1     CLRX
BCL2     LDA    $ 01    LECTURE ET TEST DE L'ETAT DE REPOS DE LA
          ANDA   #$ 80    LIGNE
          BEQ    BCL1
          INX
          CMPX   #$ 0A    CONFIRMATION DE L'ETAT DE REPOS DE LA
          BNE    BCL2    BCL2LIGNE
          CLRX
BCL3     LDA    $ 01
          ANDA   #$ 80    TEST DU FRONT DESCENDANT DU BIT START
          BNE    BCL3
BCL4     INX          MESURE DE LA LARGEUR D'UN BIT
          NOP
          LDA    $ 01    TEST DU FRONT MONTANT DU BIT 1 DE L'OCTET
          ANDA   #$ 80    DE RECONAISSANCE .
          BEQ    BCL4
          TXA
          SUB    #$ 01    VALEUR MESUREE - 1
          CMPA   #$ 06    CORRECTION DE LA VALEUR MESUREE

```

	LBLS	BCL5	PAR COMPARAISON AVEC LES DIFFERENTS
	CMPA	#\$ 0C	RAPPORTS DE DIVISION DES CADENCES PAR 8
	LBLS	BCL6	
	CMPA	#\$ 10	
	LBLS	BCL7	
	CMPA	#\$ 18	
	LBLS	BCL8	
	CMPA	#\$ 20	
	LBLS	BCL9	
	CMPA	#\$ 30	
	LBLS	BCL10	
	CMPA	#\$ 40	
	LBLS	BCL11	
	CMPA	#\$ 60	
	LBLS	BCL12	
	CMPA	#\$ C0	
	LBLS	BCL13	
BCL5	LDA	#\$ 06	
	BRA	SUI	
BCL6	LDA	#\$ 0C	
	BRA	SUI	
BCL7	LDA	#\$ 10	
	BRA	SUI	
BCL8	LDA	#\$ 18	
	BRA	SUI	
BCL9	LDA	#\$ 20	
	BRA	SUI	
BCL10	LDA	#\$ 30	
	BRA	SUI	
BCL11	LDA	#\$ 40	
	BRA	SUI	
BCL12	LDA	#\$ 60	
	BRA	SUI	
BCL13	LDA	#\$ C0	
SUI	STA	\$ 30	SAUVEGARDE LA VITESSE.

Programme de lecture du protocole

```

CLC
SUBA  #\$ 04   CORRECTION APPOREE POUR OBTENIR DES
STA   \$  40   TEMPOS DE DUREE EGALE A 1 BIT
SUBA  #\$ 01
STA   \$  46
ADDA  #\$ 05   CORRECTION POUR OBTENIR TEMPO = 1/2 BIT
RORA
SUBA  #\$ 01
STA   \$  43
ADDA  \$  30   CORRECTION POUR OBTENIR TEMPO = 1.5 BIT
STA   \$  41
LDX   \$  41
BCL20 LDA  \$  01
      ANDA #\$ 80   TEST DU FRONT DESCENDANT DU BIT 4 DE
      BNE   BCL20  L'OCTET DE RECONAISSANCE
BCL21 DEX
      BNE   BCL21  TEMPO DE DUREE = 1.5 BITS
BCL22 LDA  \$  01   LECTURE DU BIT RECU
      ANDA #\$ 80   A TRAVERS LA LIGNE PB7.
CLC
ROR   \$  42
ORA   \$  42
STA   \$  42   SAUVEGARDE DE L'OCTET DANS L'ADRESSE 42
LDA   \$  40
BCL23 DECA          TEMPO = 1 BIT
      BNE   BCL23
      INX
      CMPX #\$ 03   TEST DU NOMBRE DE BITS LUS
      BNE   BCL22  SI NOMBRE =3 FIN DE LA LECTURE DE L'OCTET.

```

Programme de réception de l'octet de sélection des moteurs.

```

BCL24 CLRX
BCL25 LDA $ 01 LECTURE ET TEST DE L'ETAT DE REPOS DE LA
      ANDA #$ 80 LIGNE
      BEQ BCL24
      INX
      CMPX #$ 0A CONFIRMATION DE L'ETAT DE REPOS DE LA
      BNE BCL25 LIGNE
      LDX $ 43
BCL26 LDA $ 01
      ANDA #$ 80 TEST DU FRONT DESCENDANT DU BIT START
      BNE BCL26
BCL27 DEX
      BNE BCL27 TEMPO = 1/2 BIT
      LDA $ 01
      ANDA #$ 80 CONFIRMATION DU BIT START
      BNE BCL24
      ROR $ 50 INSTRUCTIONS INUTILES PERMETTANT
      ROR $ 50 D'ATTEINDRE UNE TEMPO DE 1 BIT
      ROR $ 50
BCL29 LDA $ 46
      NOP
BCL28 DECA
      BNE BCL28 TEMPO = 1 BIT
      LDA $ 01 LECTURE DES 8 BITS DE LA DONNEE SERIE
      ANDA #$ 80 BIT PAR BIT
      CLC PUIS LEUR SAUVEGARDE DANS
      ROR $ 45 L'ADRESSE 45H
      ORA $ 45
      STA $ 45
      INX
      CMPX #$ 08
      BNE BCL29
      JSR DEC BRANCHEMENT AU PROGRAMME DE DETECTION
      D'ERREURS

```

Programme de reception de la donnée serie .

```

      CLR    $    05    PORT B EN ENTREE
BCL1  CLRX
BCL2  LDA    $    01    LECTURE ET TEST DE L'ETAT DE REPOS DE LA
      ANDA   #$    80    LIGNE
      BEQ    BCL1
      INX
      CMPX   #$    0A    CONFIRMATION DE L'ETAT DE REPOS DE LA
      BNE    BCL2    LIGNE
BCL3  LDA    $    01
      ANDA   #$    80    TEST DU FRONT DESCENDANT DU BIT START
      BNE    BCL3
      LDA    #$    2F
BCL4  DECA
      BNE    BCL1    TEMPO = 1/2 BIT
      LDX    #$    00
      LDA    $    01
      ANDA   #$    80    TEST DE LA VALIDITE DU BIT START
      BNE    BCL1
BCL7  LDA    #$    4A
BCL5  NOP
      DECA
      BNE    BCL5    TEMPO= 1 BIT
      LDA    $    01    LECTURE DES 8 BITS DE LA DONNEE SERIE
      ANDA   #$    80    BIT PAR BIT
      BEQ    BCL6
      LDA    #$    80
BCL6  CLC
      ROR    $    41    PUIS LEUR SAUVEGARDE DANS
      ORA    $    41    L'ADRESSE 41H
      STA    $    41
      INX
      CMPX   #$    08
      BNE    BCL7
      JSR    DEC    BRANCHEMENT AU PROGRAMME DE DETECTION
                        D'ERREURS

```

JMP PRO2 BRANCHEMENT AU PROGRAMME DE TRAITEMENT

Programme de mise en forme de la donnée .

	LDA	#\$	0A	CHARGEMENT DU RAPPORT DE
	STA	\$	42	DEMUTLIPLICATION
	LDA	\$	41	LECTURE DE LA DONNEE RECUE
	CLR	\$	46	
	BRCLR	7.\$46	BCL1	
	COM	\$	46	
BCL1	BCLR	7	\$46	
	CLC			
	LDA	#\$	08	
	STA	\$	40	
	LDX	\$	42	
	CLRA			MULTIPLICATION DE LA DONNEE RECUE PAR
	LSRX			LE RAPPORT DE DEMUTLIPLICATION INTRODUIT
BCL	BCC	BL		PAR LES ENGRENAGES .
	ADDA	41		
BL	RORA			
	RORX			
	DEC	\$	40	
	BNE		BCL	
	STA	\$	45	STOCKAGE DE LA DONNEE RESULTANTE SUR
	STX	\$	44	DEUX OCTETS

Programme de sélection de la durée du régime transitoire .

```

CLC
RORA          DIVISION PAR 2 DE L'OCTET DE POIDS FORT
BNE          BCL1  SI #0 ALORS ADR1=ADR2=E8
RORX          DIVISION PAR 2 DE L'OCTET DE POIDS FAIBLE
CMPX  # $ E8
BLO          BCL2
BCL1  LDX  # $ E8
BCL2  STX  $  47
      STX  $  48
      INX
      STX  $  4F
      LDX  # $ 00
      LDA  # $ FF
      STA  $  05  MISE DU PORT B EN SORTIE
      STA  $  01
      LDA  # $ FE
      STA  $  40
      LDA  # $ FD  CHARGEMENT DES DIFFERENTES
      STA  $  41  CONFIGURATIONS D'EXCITATION
      LDA  # $ FB  DES PHASES DU MOTEUR .
      STA  $  42
      LDA  # $ F7
      STA  $  43

```

Programme d'excitation du moteur.

```

BC2   LDA   $ 46   TEST DU SENS DE ROTATION
      CMPA  #$ 00
      BEQ   AV
      CMPX  #$ 03   MARCHE ARRIERE
      BNE   BCL4  TEST DE FIN DE CYCLE
      LDX   #$ FF   SI OUI REINITIALISATION DE X
BCL4  INX
      BRA   EXE
AV     CMPX  #$ 00   MARCHE AVANT
      BNE   BC1   TEST DE FIN DE CYCLE
      LDX   #$ 04   SI OUI REINITIALISATION DE X
BC1    DEX           DECREMENTATION DU COMPTEUR DE CYCLE
EXE    LDA   $ 40,X  APPLICATION DE LA CONFIGURATION
      STA   $ 01   D'EXCITATION AUX PHASES DU MOTEUR
      JSR   TMP   TEMPORISATION DE CONTROLE DE VITESSE
      DEC   $ 44
      BNE   BCL3
      LDA   $ 45   DECREMENTATION DE LA DONNEE
      BEQ   SUI   SI DONNEE #0 CONTINUER LE POSITIONNEMENT
      BRA   BC2
BCL3  LDA   $ 44
      CMPA  #$ FF
      BNE   BC2
      DEC   $ 45
      BRA   BC2
SUI    JMP   FIN   SAUT AU PROGRAMME DE FIN DE
                       POSITIONNEMENT

```

Programme de calcul des largeurs d'impulsions commandants la vitesse.

TMP	LDA	#\$	F5	
B1	INC	\$	50	
	INC	\$	50	TEMPO COMPLEMENTAIRE A T2
	INC	\$	50	
	DECA			
	BNE	\$	B1	
B3	LDA	\$	47	
	BNE		ACC	TEST DE FIN DE REGIME D'ACCELERATION
	LDA	\$	45	
	BNE		REG1	BRANCHEMENT AU S/P DE REGIME ETABLI
	LDA	\$	44	
	CMPA	\$	4F	
	BHS		REG2	
	DEC	\$	48	DECREMENTE ADR2 POUR INDIQUER PAS FAIT
	LDA	\$	48	
	STA	\$	4A	
L2	NOP			
	LDA	#\$	0B	
	DECA			
	BNE		L1	TEMPO DE 110 µS
	INC	\$	4A	INCREMENTATION DE TEMP2
	LDA	\$	4A	
	CMPA	\$	4F	
	BNE		L2	
	BRA		GO	RETOUR AU PROGRAMME D'EXCITATION
ACC	DEC	\$	47	
	LDA	#\$	01	
X1	DECA			
	BNE		X1	
	NOP			
	STA	\$	50	
	LDA	\$	47	
	STA	\$	49	
X2	DEC	\$	49	
	NOP			

```
      NOP
      LDA  # $ 0B
X3    DECA
      BNE      X3
      LDA  $ 49
      BNE      X2
      BRA      GO
REG1  CLR  $ 50
REG2  LDA  # $ 01
      NOP
X4    DECA
      BEQ      X4
GO    RTS
```

RETOUR AU PROGRAMME PRINCIPAL

Programme de détection d'erreur .

	LDA	#\$	08	
	STA	\$	42	
	LDA	\$	41	
	CLC			
BCL2	RORA			
	BCC		BCL1	
	INC	\$	50	
BCL1	DEC	\$	41	
	LDA	\$	41	
	CMPA	#\$	00	
	BNE		BCL2	
	CLC			
	ROR	\$	50	DIVISION DE COMP PAR 2
	BCC		JS	
	LDA	#\$	F2	
	STA	\$	02	DTR=1 ET RTS=0 POUR INDIQUER UNE ERREUR
	LDA	#\$	90	
BC	DECA			
	BNE		BC	
	LDA	#\$	F3	ACTIVATION DE DTR ET RTS
	STA	\$	02	
	JMP		RECP	SAUT AU S/P DE RECEPTION
JS	CLR	\$	02	DESACTIVATION DES LIGNES DTR ET RTS
	BRA		PRG	BRANCHEMENT AU PROGRAMME DE COMMANDE


```
program naw(input,output);
uses crt;
label 1,2,3,4;

type tab=array[1..26] of integer;

var
  c:char;
  y:tab;
  i,jour,mois:integer;
  g,g1,g2,beta,l,sl,cl,tl: real;
  x,cd,sd,td,delta,alfa,pr: real;
  br,r,rr,rl,gamma,dt,b1: real;
  y1,z,x1,hp: byte;
  t:shortint;
  b,latitude,a,c1,d:real;
  f:text;           {variable fichier servant ... la sauvegarde
                    de l'angle optimal}

begin
  y[1]:=0;
  y[2]:=31;
  y[3]:=59;           {introduction des numeros
                    des jours de l'ann,e non bissextile}
  y[4]:=90;
  y[5]:=120;
  y[6]:=151;
  y[7]:=181;
  y[8]:=212;
  y[9]:=243;
  y[10]:=273;
  y[11]:=304;
  y[12]:=334;
  y[13]:=365;
  y[14]:=0;           {introduction des numeros
                    des jours de l'ann,e bissextile}
  y[15]:=31;
  y[16]:=60;
  y[17]:=91;
  y[18]:=121;
  y[19]:=152;
  y[20]:=182;
  y[21]:=213;
  y[22]:=244;
  y[23]:=274;
  y[24]:=305;
  y[25]:=335;
  y[26]:=366;

  assign(f,'exbeta.dat');reset(f);read(f,b); close(f); {lecture de
  l'angle optimale de la derni,re inclinaison effectuer}
```

```

writeln('Introduisez la latitude en degres:');
readln(latitude);
1: writeln('Introduisez la date du jour sous la forme: jour mois');
2: readln(jour,mois);
    if (mois=0) or (mois>12) then goto 1;
    if (jour=0) or (jour>31) then goto 1; {test d'erreurs
eventuelles sur les valeurs introduites:jour,mois}
clrscr;
writeln('Ann,e est-elle bissextile (o/n)?');
readln(c);
    if (c='o') or (c='O') then mois:=mois+13;
jour:=jour + y[mois];          {calcul du numero du jour}
l:=3.1415927 * latitude/180;
sl:=sin(l);cl:=cos(l);tl:=sl/cl;
x:=0.0172142 * (jour-1);
a:=0.395205524-22.9015851 *cos(x)+4.07201165*sin(x);
b1:=-0.387263458*cos(2*x)+0.0514366692*sin(2*x);
c1:=-0.153682511*cos(3*x)+0.0858863415*sin(3*x);
d:=-0.009589005*cos(4*x)+0.0053627416*sin(4*x);
delta:= a+b1+c1+d;          {calcul de la d,clinaison}
dt:=3.1415927*delta/180;
sd:=sin(dt);cd:=cos(dt);td:=sd/cd;
alfa:=-tl*td;
pr:=(sqr(1-alfa*alfa))/alfa;
br:=arctan(pr);
if (alfa<0) then br:=br+3.1415927;
r:=td*br/(sin(br));
rr:=arctan(r);
rl:=rr*180/3.1415;
beta:=latitude-rl;          {angle d'inclinaison optimal}

{
*****
MISE EN FORME
*****
}

gamma:=beta-b;          {calcul de la difference entre
angles present et anterieur}
b:=beta;          {sauvegarde de la position}

assign(f,'exbeta.dat');rewrite(f); write(f,b); close(f);
g:=gamma*100;
g1:=round(g);
g2:=128*g1/40;
t:=round(g2);
    if t<0 then          {test du sens de rotation}
    begin

```

```

        t:=t-1;
        t:=not t;
        ASM
            mov ah,t
            or ah,80h      {mise ... 1 du bit7 de l'octet
            mov t,ah      ... emettre}
        END;
    end;
    port[$3fc]:=$03;      {mise ... 1 de DTR et RTS}
3:  x1:=port[$3fe];      {lecture du registre d'etat du modem}
    hp:=0;
    asm
        mov ah,x1
        and ah,20h      {test si DSR=1}
        jz @fin
        or hp,1
        @fin:
    end;
    if hp=0 then goto 3;  {si DSR#1 reboucle et attente }
    hp:=0;
    asm
        mov ah,x1
        and ah,10h      {test si CTS=1}
        jz @fin
        or hp,1
        @fin:
    end;
    if hp=0 then goto 3;  {si CTS#1 reboucle et attente}

        {EMISSION}

4:  y1:=port[$3fe];      {Lecture du registre d'etat du modem}
    hp:=0;
    asm
        mov ah,y1
        and ah,20h      {test si DSR=1}
        jz @fin
        or hp,1
        @fin:
    end;

    if hp=0 then goto 4;  {si DSR#1 attente de DSR=1}
    hp:=0;
    asm
        mov ah,y1
        and ah,10h      {test si CTS=1}
        jz @fin
        or hp,1
        @fin:

```

```
end;
  if hp=0 then goto 3; {SI CTS#1 re,mettre une donn,e}
port[$3fc]=$00;      {Desactivation de DTR et CTS}
writeln('Tapez ECHAP pour sortir');
c:=readkey;
  if c=#27 then exit;
writeln(b:10,t:10);
goto 3;
end.
```

REFERENCES

- [1] R. BERNARD Le rayonnement solaire, conversion thermique et applications ed technique & documentation
- [2] M. KOUSSA , Recueil des paramètres astronomiques relatifs aux sites d'Algerie, 1989
- [3] S. LEIBSON , Manuel des interaces, Mc GRAW-HILL 1984
- [4] G. LEBLANC Turbo c
- [5] THE INTERFACE CIRCUITS DATA BOOK, Texas instruments incorporated PP 271-278
- [6] W. GOFTON Techniques de communication série sur PC et compatibles
- [7] thèse moteur
- [8] data book mosfet
- [9] M. MESSUD , La pratique du microprocesseur, Edition CEPADUES 1991
- [10] M. HADDADI , Conception et optimisation de générateurs à conversion photovoltaïque, These de Doctorat
- [11] J. LILEN, Interfaces pour microprocesseur et micro-ordinateur, Edition RADIO, 1987
- [12] Microprocesseurs et memoires, THOMSON-CSF , 1980
- [13] S. MENACEUR, Electronique digitale, Edition info-Z , 1980
- [14] R. BERNARD , Le rayonnement solaire conversion thermique et application, Technique et documentation , 1980
- [15] C. ETIEVANT , Central receiver plant evaluation, 1986
- [16] J. BERRY , Automatisation d'un héliostat photovoltaïque, CNRS laboratoire d'automatique et d'analyse des systemes, 1978
- [17] C. TAVERNIER, Microcontrôleur 6805 et 68HC05 et les versions uvprom 68705, Edition DUNOD/tech 1993
- [18] L'ordinateur , Les editions scientifiques , Octobre/novembre 1992 .
- [19] H. LILEN , technique et normes pour l'interfaçage d'un PC .
- [20]