

COMMANDE DU ROBOT MULTISOFT

1. DESCRIPTION GENERALE DU ROBOT

Le robot Multisoft possède 6 moteurs pas-à-pas, ce qui lui confère 5 degrés de liberté. Il peut donc se déplacer aisément dans une sphère de 960 mm de diamètre.

Son organe de préhension (doigts) lui permet de porter des charges jusqu'à 300 g, soit 6 % de son propre poids.

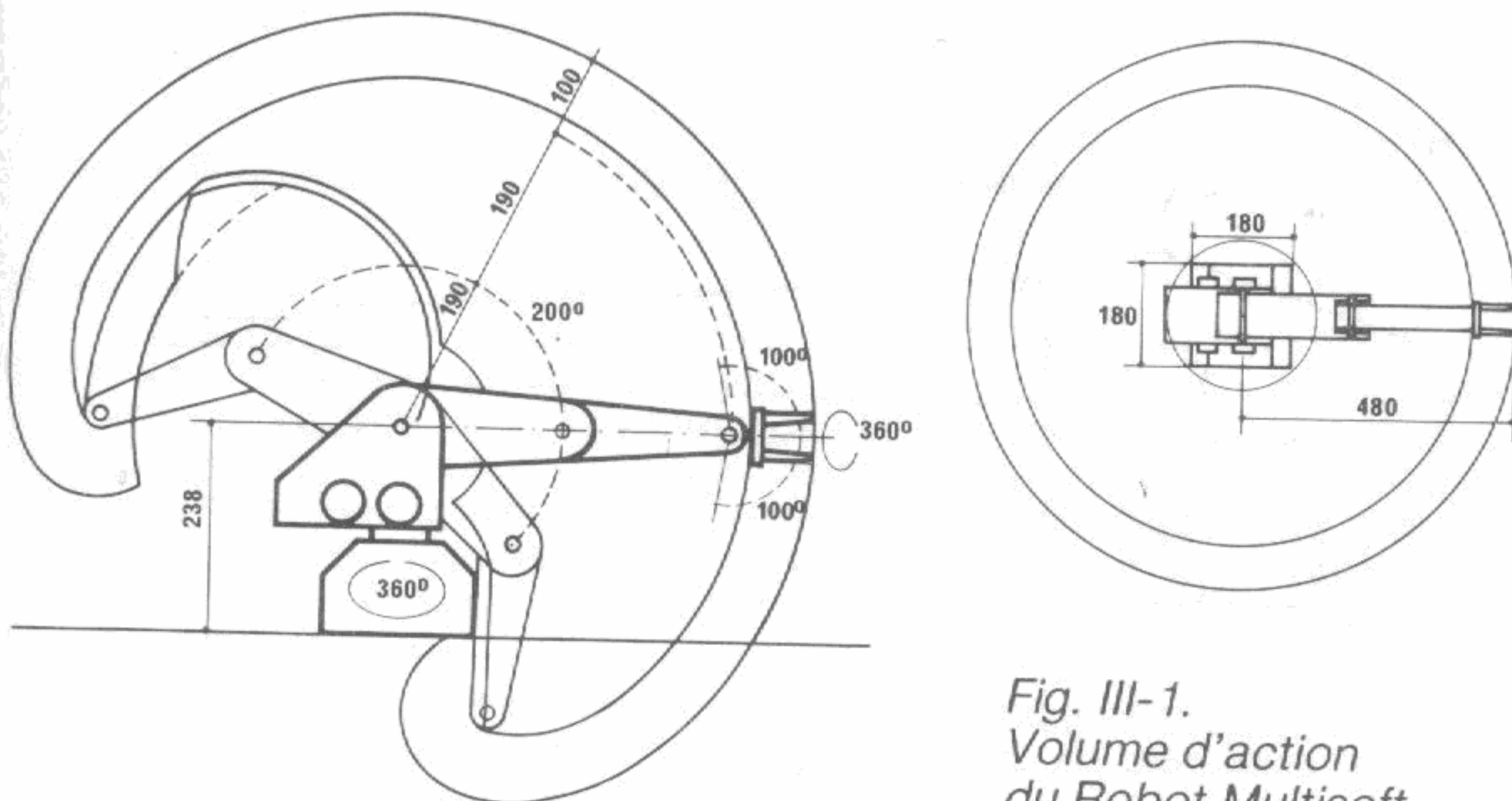
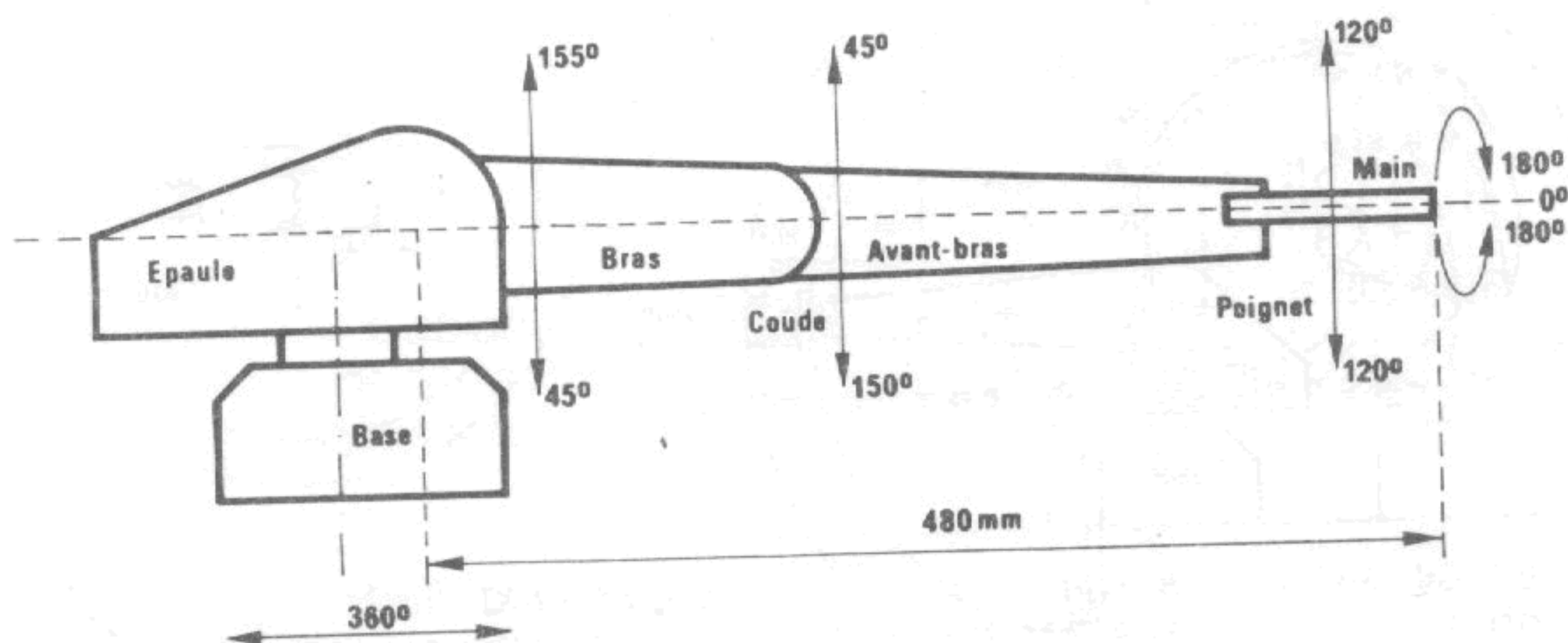


Fig. III-1.
Volume d'action
du Robot Multisoft.

CARACTERISTIQUES MECANQUES		CARACTERISTIQUES DE COMMANDE	
Volume d'action	Ø 960	Raccordement de puissance	12-15 V, 5 A permanent protégé
Précision de répétabilité	± 2 MM	Ordinateur de commande	tout micro-ordinateur avec interface parallèle, 8 bits
Capacité de charge à basse vitesse	3 N ou 300 g	Informations de capteurs	via le port 8 bits bidirectionnel (7 canaux pour position force ou proximité) ou via entrées analogiques
Sécurité contre surcharge	par ressorts et glissement du moteur	Sorties de commandes auxiliaires	2 sorties prévues sur l'interface parallèle (par ex. pour système de préhension magnétique)
Surface au sol	180 x 180	Capacité mémoire	minimum 16 K octets pour un programme d'apprentissage
Poids robot	450 N (4,5 kg)	Commande des moteurs	point par point, ou trajectoire multiaxe
Main	3 doigts ou autre option	Dialogue	interactif via l'écran du micro-ordinateur
Motorisation	6 moteurs pas-à-pas (pas d'entretien)	Mémorisation	tout support (cassette, disquette, etc.)
Cinématique de transmission	réduction par courroies crantées		
Niveau de bruit	< 60 dBA		
Température de travail	10° à 30°		



Axe/Moteur	Plage d'action	Vitesse maxi	Angle par 1/2 pas
Rotation base /I	360° non limité	46°/seconde	0,115°
Epaule /II	200°	32°/s	0,081°
Coude /III	195°	32°/s	0,081°
Poignet /IV	200°	46°/s	0,115°
Rotation main /V	360°	46°/s	0,115°
Doigts /VI	0-80 mm	12 mm/s	0,026 mm

- **Attention** Les numéros des moteurs peuvent être modifiés au montage. Si vous désirez rétablir les numéros correspondants, il suffit d'intervertir les prises moteur sur la carte d'interface.

Tableau III-1. — Micro Robot Multisoft : fiche technique.

Quelques précisions

Degré de liberté

Un corps solide a une position complètement définie dans l'espace lorsque l'on connaît les 6 degrés de liberté x, y, z et α, β, θ , où :

- x, y, z sont les coordonnées d'un point du solide par rapport à un repère fixe,
- α, β, θ étant les angles nautiques qui définissent l'orientation du solide par rapport au repère fixe.

Avec 5 degrés de liberté, le corps manipulé par le robot pourra donc prendre n'importe quelle position dans l'espace d'action.

Moteur pas-à-pas

C'est un moteur qui possède plusieurs positions stables. On peut passer d'une position stable à la suivante (ou à la précédente) en envoyant une (ou plusieurs) impulsion(s) dans une (ou plusieurs) des bobines du stator. L'écart entre deux positions stables successives est appelé **un pas**. En alimentant les bobines de façon simultanée, on peut faire travailler le moteur en demi-pas, ce qui améliore les performances de couple et de positionnement.

Le moteur pas-à-pas est donc un actionneur incrémental qui se prête particulièrement bien à une commande digitale, en particulier à partir d'un système à microprocesseur.

Le Robot Multisoft est doté d'une interface électronique qui permet la commande de l'un quelconque des 6 moteurs à partir d'un octet de commande.

Si les octets sont envoyés au robot à une vitesse suffisante, on aura (à l'échelle humaine) l'impression de voir les moteurs tourner simultanément.

Si la vitesse de la commande est trop grande, les moteurs ne tourneront plus car leur inertie les empêchera de suivre une cadence trop élevée. Ce phénomène est particulièrement sensible au démarrage où les frottements sont maximaux. C'est pourquoi une rampe d'accélération est rendue nécessaire au démarrage de chaque moteur pour atteindre ses performances maximales.

A l'arrêt au contraire, si l'on coupe brusquement le débit des octets de commande, le moteur en rotation sera entraîné par son inertie et continuera donc de tourner en dépassant de plusieurs pas la position qu'on souhaitait lui voir prendre. Il sera nécessaire cette fois de prévoir une décélération progressive quelques pas avant l'arrêt, afin de garantir la position finale.

Un bon programme de commande de robot à moteurs pas-à-pas sera donc écrit en Assembleur et possédera soit une rampe linéaire d'accélération puis de décélération, soit une montée et une descente en vitesse exponentielle (ce dernier cas étant plus performant).

2. FONCTIONNEMENT EN MODE MANUEL

On se propose dans ce mode de ne faire tourner qu'un seul des moteurs.

2.1 Mode de commande d'un moteur

Afin de rendre le robot compatible avec la plupart des micro-ordinateurs, il a été doté d'un port 8 bits bidirectionnel. Ce port est organisé de la façon suivante :

D₁ : Un front descendant sur ce bit provoque un transfert des informations en parallèle du micro vers le robot. Ce bit sera relié à la sortie S0 de la carte automate ou directement à PA0 de la carte PIA6821.

D₄, D₃, D₂ : Ces 3 bits permettent la sélection d'un moteur parmi les 6 suivant le code du tableau ci-contre :

Octet robot

D8	D7	D6	D5	D4	D3	D2	D1	
				0	0	1		Moteur de commande DOIGTS
				0	1	0		Moteur de commande MAIN
				0	1	1		Moteur de commande POIGNET
				1	0	0		Moteur de commande COUDE
				1	0	1		Moteur de commande EPAULE
				1	1	0		Moteur de commande BASE

Tableau III-2. – Codage de sélection des moteurs.

D₈, D₇, D₆, D₅ : Ces 4 bits servent à la commande des bobines du moteur sélectionné selon le *tableau III-3*.

Si tous les bits sont à 1, les bobines sont toutes désactivées et l'on peut faire tourner les axes des moteurs à la main.

Octet robot

D8	D7	D6	D5	D4	D3	D2	D1	N° du pas
0	0	1	1					1
1	0	1	1					1 1/2
1	0	0	1					2
1	1	0	1					2 1/2
1	1	0	0					3
1	1	1	0					3 1/2
0	1	1	0					4
0	1	1	1					4 1/2

Tableau III-3. – Code de commande des bobines d'un moteur.

2.2 Interconnexion micro → robot

On réalisera les liaisons suivantes :

$$\begin{array}{l} \overline{PA0} = S0 \rightarrow D1 \\ \overline{PA1} = S1 \rightarrow D2 \\ \overline{PA2} = S2 \rightarrow D3 \\ \vdots \\ \overline{PA7} = S7 \rightarrow D8 \end{array}$$

micro robot

Le branchement au connecteur du Robot Multisoft se fera conformément à la *figure III-2*. D'autre part, il faut relier la broche + 5 V à la broche COM (point commun des 8 résistances de 10 kΩ).

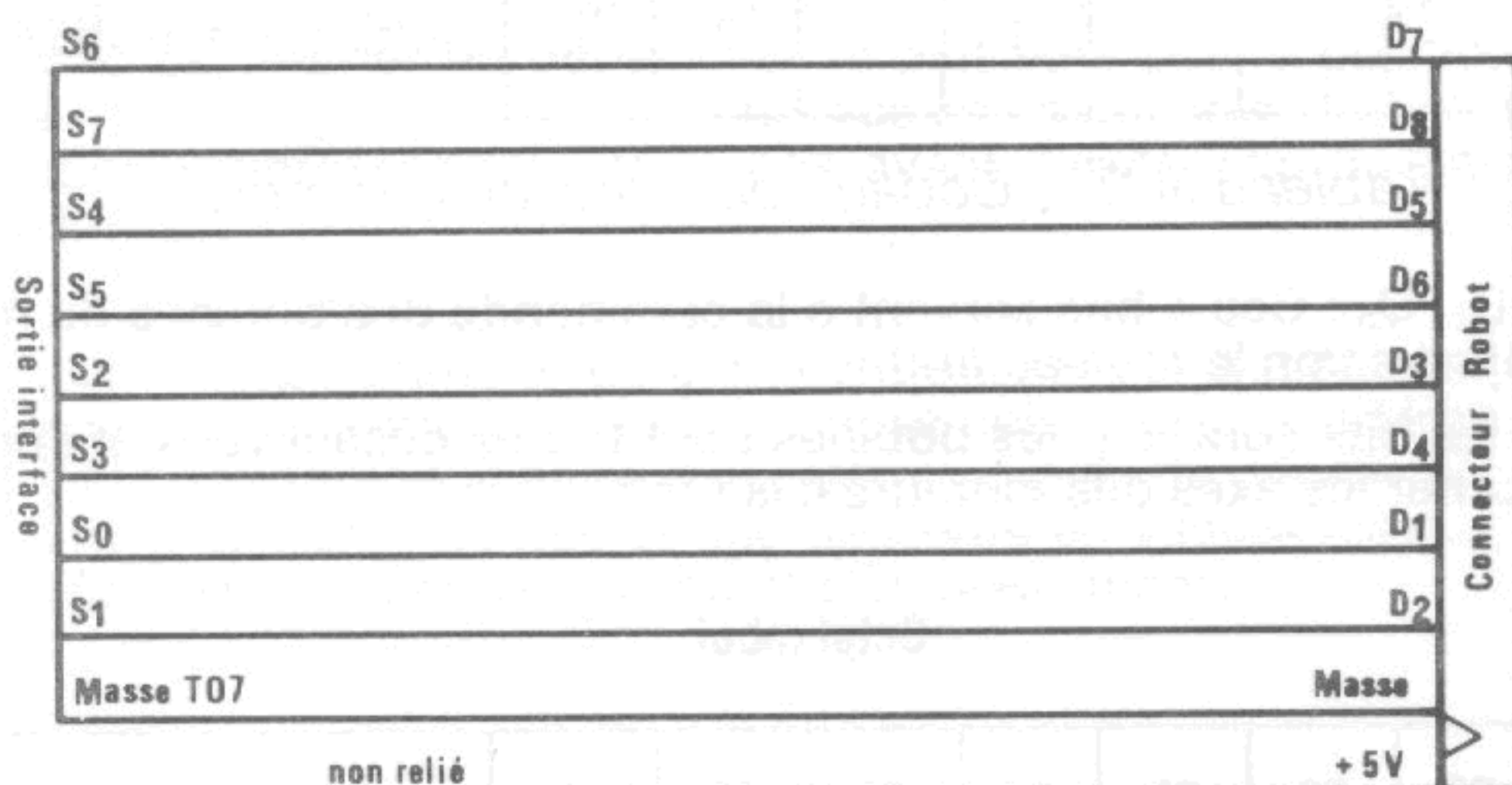


Fig. III-2. – Repérage du connecteur Robot.

2.3 Programme de commande

On se propose de réaliser un programme en Assembleur permettant la rotation d'un des 6 moteurs au choix dans l'un quelconque des sens de rotation en mode demi-pas. Une première partie du programme écrite en Basic permettra de choisir :

- le numéro du moteur, codé de 1 à 6 conformément au *tableau III-2*,
- le sens de rotation + ou –,
- le nombre de pas de rotation.

En mémoire, quatre cases seront réservées au rangement de ces informations. Le sens de rotation sera codé de façon numérique : 0 pour + et 1 pour –.

Ces quatre cases contiendront donc les informations suivantes :

Adresse	Contenu
\$7C00	N° du moteur selon le code Robot
\$7C01	Bit de sens de rotation (0 ou 1)
\$7C02, 7C03	Nombre de pas

● *Etude de la partie Basic*

- Initialisation de l'écran, et de la mémoire.
- Entrée du numéro du moteur.
- Codage du numéro selon le code du Robot et rangement en mémoire à l'adresse Moteur = &H7C00.
- Entrée du sens de rotation « + » ou « - » et codage par 0 ou 1 puis rangement à l'adresse SENS = &H7C01.
- Entrée du nombre de pas et calcul de l'octet de poids fort rangé en &H7C02, puis de l'octet de poids faible rangé en &H7C03.
- Exécution du mouvement par appel du programme écrit en Assembleur et rangé à partir de &H7C04.
- Retour à « Entrée du numéro du moteur »

```

10 ' *****
20 ' *
30 ' * PROGRAMME DE COMMANDE D'UN *
40 ' * MOTEUR DU ROBOT MULTISOFT *
50 ' *
60 ' *****
70 '
80 '
90 ' --- INITIALISATION ---
100 '
110 '
120 CONSOLE 0,24 : SCREEN2,0,0 : CLS : CLEAR,&H7C00
130 FOR N = &H7C04 TO &H7CA6
140 READ X$ : X = VAL("&H"+X$)
150 POKE N, X
160 NEXTN
170 '
180 DATA 34,7F,CC,00,00,FD,E7,F2,CC,FF,00,FD,E7,F0,CC,04
,04,FD,E7,F2,4F,B7,E7,F0,BE,7C,02,27,7B,7D,7C,01,26,27,C
E,7C,9B,A6,C0,B4,7C,00,B7,E7,F0,8A,01,B7,E7,F0,30,1F,27,
5F,BC,00,0B,24,04,BD,4B,20,02,8D,2F,11,B3,7C,A3,27,DB,26
,DC,CE,7C,A3,A6,C2
190 DATA 84,7C,00,B7,E7,F0,8A,01,B7,E7,F0,30,1F,27,3B,BC
,00,0B,24,04,BD,21,20,02,BD,0B,11,B3,7C,9B,27,DB,26,DC,1
0,BE,7C,A3,10,BC,7C,A5,27,06,31,30,10,BF,7C,A3,31,3F,26,
FC,39,10,BE,7C,A3,31,AB,10,10,BF,7C,A3,31,3F,26,FC,39,35
,FF,CE,4E,6E,2E,3E,1E
200 DATA 9E,BE,03,00,02,B0
210 '

```

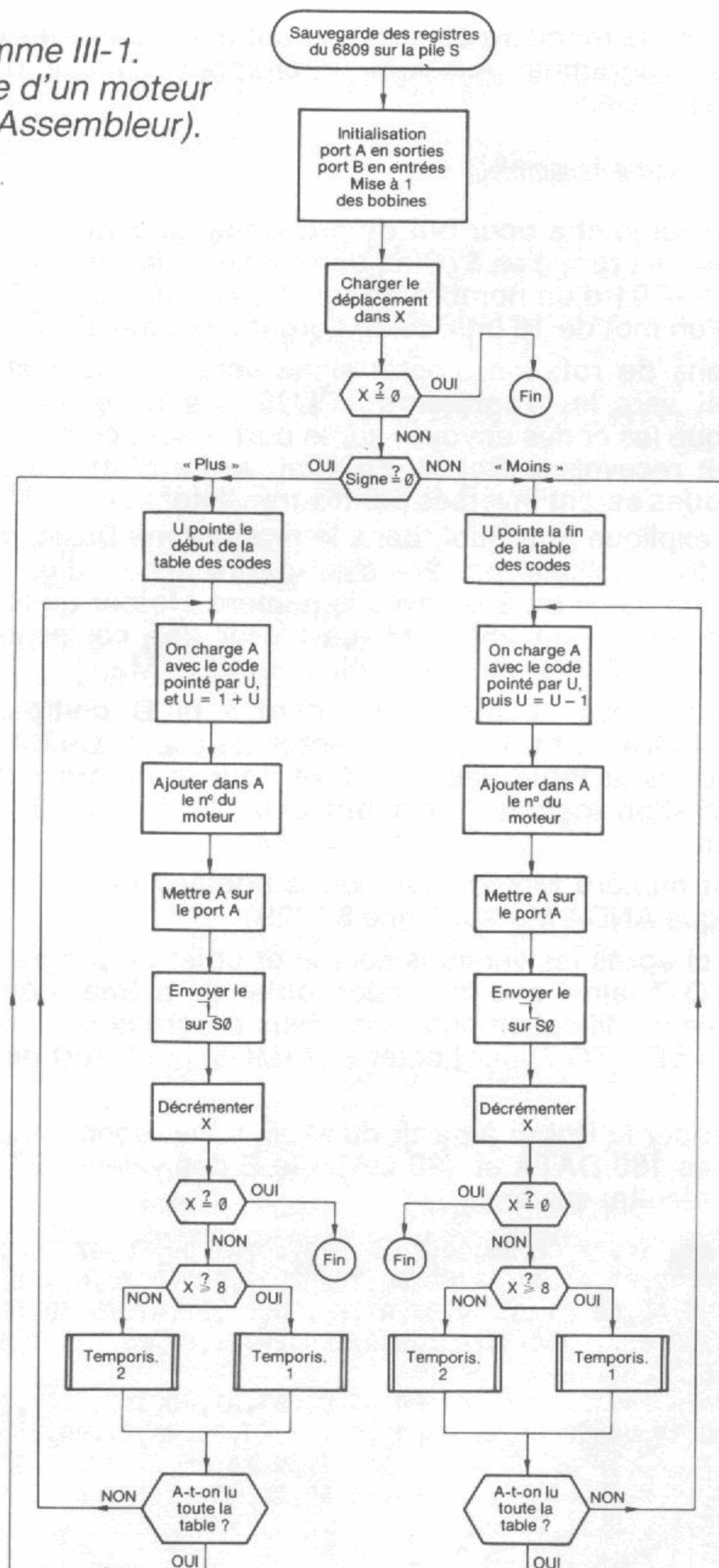
```

220 '
230 '--- CHOIX DU MOTEUR ---
240 '
250 '
260 CLS : LOCATE 5,5 : PRINT"MOTEUR NO 1 : DOIGTS"
270 LOCATE 5,7 : PRINT"MOTEUR NO 2 : POIGNET"
280 LOCATE 5,9 : PRINT"MOTEUR NO 3 : MAIN"
290 LOCATE 5,11 : PRINT"MOTEUR NO 4 : COUDE"
300 LOCATE 5,13 : PRINT"MOTEUR NO 5 : EPAULE"
310 LOCATE 5,15 : PRINT"MOTEUR NO 6 : BASE"
320 LOCATE 5,20 : INPUT"TAPER VOTRE CHOIX (DE 1 A 6) : "
,M
330 '
340 '
350 '--- CODAGE DU MOTEUR ET RANGEMENT ---
360 '
370 '
380 POKE &H7C00,255-2*M
390 '
400 '
410 '--- CHOIX DU SENS, CODAGE ET RANGEMENT ---
420 '
430 '
440 CLS : LOCATE 5,10 : INPUT"CHOISISSEZ LE SENS DE ROTA
TION (+ OU -) : ", S$
450 IF S$="+" THEN POKE &H7C01,0
460 IF S$="-" THEN POKE &H7C01,1
470 '
480 '
490 '--- CHOIX DU DEPLACEMENT ---
500 '
510 '
520 LOCATE 5,20 : INPUT "COMBIEN DE PAS : ", P
530 '
540 '
550 '--- CODAGE EN DEMI-PAS ---
560 '
570 '
580 PH = 2*P @ 256
590 PL = 2*P MOD 256
600 POKE &H7C02 , PH : POKE &H7C03 , PL
610 '
620 '
630 '--- EXECUTION DU DEPLACEMENT ---
640 '
650 '
660 EXEC &H7C04
670 GOTO 260

```

Programme III-1. – Commande d'un moteur du Robot (version TO 7 et TO 7-70).

Organigramme III-1.
Commande d'un moteur
du Robot (Assembleur).



L'initialisation de la mémoire consiste en fait à implanter les codes hexadécimaux du programme Assembleur ci-après à l'aide d'instructions POKE, Adresse, Valeur.

● *Etude du programme Assembleur*

Le programme ci-joint a pour but de provoquer la commande du moteur dont le numéro est rangé en \$7C00, dans le sens de rotation dont le code est rangé en \$7C01 d'un nombre de demi-pas rangé en \$7C02 et \$7C03 sous forme d'un mot de 16 bits, selon l'organigramme III-1.

Suivant le sens de rotation, l'octet signe sera nul ou égal à 1, ce qui conduira, soit vers le programme « PLUS », soit vers « MOINS ». On remarquera que les codes envoyés sur le port A sont complémentaires de ceux que doit recevoir le Robot. En effet, avant d'atteindre l'entrée du Robot, ces codes seront inversés par les transistors de l'ULN 2803A.

C'est ce qui explique pourquoi, dans le programme Basic, le numéro du moteur, une fois multiplié par 2 – c'est-à-dire décalé d'un bit à gauche (pour correspondance bit à bit avec le numéro Moteur du Robot) – a été complété en faisant $255 - 2 \cdot M$. La valeur 255 correspond à l'octet 11111111, donc $255 - 2 \cdot M$ est le complément de $2 \cdot M$.

De même, pour activer la commande Robot, le bit D₁ doit passer de 1 à 0 donc, au contraire, le bit PA0 doit passer de 0 à 1. Ce bit PA0 est à 0 initialement dans la table des codes, et pour le mettre à 1, il suffit de réaliser l'opération logique (bit à bit) ORA # 1 (ligne \$7C31 du programme objet).

L'addition du numéro Moteur aux codes bobines est obtenu grâce à la fonction logique ANDA Moteur (ligne \$7C2B).

On trouvera ci-après les versions source et objet du programme Assembleur pour TO 7, ainsi que la version objet du même programme pour MO5. La seule modification entre ces deux programmes est la modification de l'octet \$E7 (TO 7) par l'octet \$A7 (MO5) poids fort de l'adresse du PLA.

Pour commander le Robot à partir du MO5, il faut donc remplacer (8 fois) dans les lignes 180 DATA et 190 DATA le E des valeurs E7 par un A, ce qui donne le résultat suivant :

```
180 DATA 34,7F,CC,00,00,FD,A7,F2,CC,FF,00,FD,A7,F0,CC,04
,04,FD,A7,F2,4F,B7,A7,F0,BE,7C,02,27,78,7D,7C,01,26,27,C
E,7C,9B,A6,C0,B4,7C,00,B7,A7,F0,8A,01,B7,A7,F0,30,1F,27,
5F,8C,00,0B,24,04,BD,4B,20,02,BD,2F,11,83,7C,A3,27,DB,26
,DC,CE,7C,A3,A6,C2
190 DATA B4,7C,00,B7,A7,F0,8A,01,B7,A7,F0,30,1F,27,3B,8C
,00,0B,24,04,BD,21,20,02,BD,0B,11,83,7C,9B,27,DB,26,DC,1
0,BE,7C,A3,10,BC,7C,A5,27,06,31,30,10,BF,7C,A3,31,3F,26,
FC,39,10,BE,7C,A3,31,AB,10,10,BF,7C,A3,31,3F,26,FC,39,35
,FF,CE,4E,6E,2E,3E,1E
200 DATA 9E,BE,03,00,02,80
```

```

*****
*
*   COMMANDE DU ROBOT MULTISOFT
*   -----
*
*   Cette commande est implantable
*   dans la memoire du T07.
*   Pour le MOS il faut changer les
*   adresses du PORTA et du CRA.
*
*****

```

```

* POUR LE MOS : PORTA EQU $A7F0

```

```

PORTA EQU    $E7F0
PORTB EQU    PORTA+1
CRA EQU     PORTA+2
CRB EQU     PORTA+3
MOTEUR EQU   $7C00
SIGNE EQU    MOTEUR+1
DEF EQU     SIGNE+1

```

```

        ORG    $7C04

```

```

*
*----- INITIALISATION DU PIA -----
*

```

```

INIT    PSHS    U,X,Y,DP,D,CC On sauve
        LDD     #0
        STD     CRA      Acces a DDRA et B
        LDD     #$FF00
        STD     PORTA    PORTA=S  PORTB=E
        LDD     #$0404
        STD     CRA      Acces a PORTA et B
        CLRA
        STA     PORTA    Sorties ULN a 1

        LDX     DEP      X contient le DEP
        BEQ     FIN      Si X=0 alors FIN
        TST     SIGNE    Si 1 alors negatif
        BNE     MOINS

```


PLUS	LDU	#TAB1	U charge la table
BOUC1	LDA	,U+	A charge la bobine
	ANDA	MOTEUR	On ajoute le moteur
	STA	PORTA	En Sortie
	ORA	#1	Mise a 0 de D1
	STA	PORTA	En sortie
	LEAX	-1,X	Un de moins
	BEG	FIN	Si 0 alors FIN
	CMPX	#8	Si x>8 alors on
	BHS	ACC1	accelere, sinon
	BSR	TEMPO2	on decelere.
	BRA	DEC1	
ACC1	BSR	TEMPO1	
DEC1	CMPU	#TAB1+8	Est-on en fin de
	BEG	PLUS	table? Si oui
	BNE	BOUC1	PLUS sinon BOUC1
MOINS	LDU	#TAB1+8	Meme chose que
BOUC2	LDA	, -U	PLUS mais cette
	ANDA	MOTEUR	fois la table
	STA	PORTA	sens contraire.
	ORA	#1	
	STA	PORTA	
	LEAX	-1,X	
	BEG	FIN	
	CMPX	#8	
	BHS	ACC2	
	BSR	TEMPO2	
	BRA	DEC2	
ACC2	BSR	TEMPO1	
DEC2	CMPU	#TAB1	
	BEG	MOINS	
	BNE	BOUC2	
TEMPO1	LDY	VITESS	ACCELERATION
	CMPY	VITMAX	
	BEG	BOUC3	
	LEAY	-\$0010,Y	
	STY	VITESS	
BOUC3	LEAY	-1,Y	
	BNE	BOUC3	
	RTS		
TEMPO2	LDY	VITESS	DECCELERATION
	LEAY	\$0010,Y	
	STY	VITESS	


```

BOUCO  LEAY  -1,Y
      BNE   BOUCO
      RTS

FIN     PULS  'U,X,Y,DP,D,CC,PC

TAB1    FCB   $CE
      FCB   $4E
      FCB   $6E
      FCB   $2E
      FCB   $3E
      FCB   $1E
      FCB   $9E
      FCB   $8E

VITESS  FDB   $0300

VITMAX  FDB   $0280

      END

```

*Programme III-2. – Commande d'un moteur du Robot
(version source).*

```

*****
*
*   COMMANDE DU ROBOT MULTISOFT   *
*   -----                       *
*
*   Cette commande est implantable *
*   dans la memoire du MD5.        *
*   Pour le T07 il faut changer les *
*   adresses du PORTA et du CRA.    *
*
*****

* POUR LE T07 : PORTA EQU $E7F0

A7F0 PORTA EQU $A7F0
A7F1 PORTB EQU PORTA+1
A7F2 CRA EQU PORTA+2
A7F3 CRB EQU PORTA+3
7C00 MOTEUR EQU $7C00
7C01 SIGNE EQU MOTEUR+1
7C02 DEP EQU SIGNE+1

```

7C04

ORG \$7C04

```

*
*----- INITIALISATION DU PIA -----
*

```

7C04	34	7F	INIT	PSHS	U,X,Y,DP,D,CC	On sauve
7C06	CC	0000		LDD	#0	
7C09	FD	A7F2		STD	CRA	Acces a DDRA et B
7C0C	CC	FF00		LDD	##FF00	
7C0F	FD	A7F0		STD	PORTA	PORTA=S PORTB=E
7C12	CC	0404		LDD	##0404	
7C15	FD	A7F2		STD	CRA	Acces a PORTA et B
7C18	4F			CLRA		
7C19	B7	A7F0		STA	PORTA	Sorties ULN a 1
7C1C	BE	7C02		LDX	DEP	X contient le DEP
7C1F	27	78		BEG	FIN	Si X=0 alors FIN
7C21	7D	7C01		TST	SIGNE	Si 1 alors negatif
7C24	26	27		BNE	MOINS	
7C26	CE	7C9B	PLUS	LDU	#TAB1	U charge la table
7C29	A6	C0	BOUC1	LDA	,U+	A charge la bobine
7C2B	B4	7C00		ANDA	MOTEUR	On ajoute le moteu
7C2E	B7	A7F0		STA	PORTA	En Sortie
7C31	BA	01		DRA	#1	Mise a 0 de D1
7C33	B7	A7F0		STA	PORTA	En sortie
7C36	30	1F		LEAX	-1,X	Un de moins
7C3B	27	5F		BEG	FIN	Si 0 alors FIN
7C3A	BC	000B		CMPL	#8	Si x>8 alors on
7C3D	24	04		BHS	ACC1	accelere, sinon
7C3F	8D	48		BSR	TEMPO2	on decelere.
7C41	20	02		BRA	DEC1	
7C43	8D	2F	ACC1	BSR	TEMPO1	
7C45	1183	7CA3	DEC1	CMPL	#TAB1+8	Est-on en fin de
7C49	27	DB		BEG	PLUS	table? Si oui
7C4B	26	DC		BNE	BOUC1	PLUS sinon BOUC1
7C4D	CE	7CA3	MOINS	LDU	#TAB1+8	Meme chose que
7C50	A6	C2	BOUC2	LDA	, -U	PLUS mais cette
7C52	B4	7C00		ANDA	MOTEUR	fois la table
7C55	B7	A7F0		STA	PORTA	sens contraire.
7C5B	BA	01		DRA	#1	
7C5A	B7	A7F0		STA	PORTA	

7C5D	30	1F		LEAX	-1,X	
7C5F	27	38		BEQ	FIN	
7C61	8C	0008		CMPX	#8	
7C64	24	04		BHS	ACC2	
7C66	BD	21		BSR	TEMPO2	
7C68	20	02		BRA	DEC2	
7C6A	8D	0B	ACC2	BSR	TEMPO1	
7C6C	1183	7C9B	DEC2	CMPU	#TAB1	
7C70	27	DB		BEQ	MOINS	
7C72	26	DC		BNE	BOUC2	
7C74	10BE	7CA3	TEMPO1	LDY	VITESS	ACCELERATION
7C78	10BC	7CA5		CMPY	VITMAX	
7C7C	27	06		BEQ	BOUC3	
7C7E	31	30		LEAY	-\$0010,Y	
7C80	10BF	7CA3		STY	VITESS	
7C84	31	3F	BOUC3	LEAY	-1,Y	
7C86	26	FC		BNE	BOUC3	
7C88	39			RTS		
7C89	10BE	7CA3	TEMPO2	LDY	VITESS	DECCELERATION
7C8D	31	AB	10	LEAY	\$0010,Y	
7C90	10BF	7CA3		STY	VITESS	
7C94	31	3F	BOUC0	LEAY	-1,Y	
7C96	26	FC		BNE	BOUC0	
7C98	39			RTS		
7C99	35	FF	FIN	PULS	U,X,Y,DP,D,CC,PC	
7C9B		CE	TAB1	FCB	\$CE	
7C9C		4E		FCB	\$4E	
7C9D		6E		FCB	\$6E	
7C9E		2E		FCB	\$2E	
7C9F		3E		FCB	\$3E	
7CA0		1E		FCB	\$1E	
7CA1		9E		FCB	\$9E	
7CA2		8E		FCB	\$8E	
7CA3		0300	VITESS	FDB	\$0300	
7CA5		02B0	VITMAX	FDB	\$02B0	
		0000	END			

00000 Total Errors

ACC1	7C43
ACC2	7C6A
BOUC0	7C94
BOUC1	7C29
BOUC2	7C50
BOUC3	7CB4
CRA	A7F2
CRB	A7F3
DEC1	7C45
DEC2	7C6C
DEP	7C02
FIN	7C99
INIT	7C04
MOINS	7C4D
MOTEUR	7C00
PLUS	7C26
PORTA	A7F0
PORTB	A7F1
SIGNE	7C01
TAB1	7C9B
TEMPO1	7C74
TEMPO2	7CB9
VITESS	7CA3
VITMAX	7CA5

*Programme III-3. – Commande d'un moteur du Robot
(version objet).*

```

*****
*
*   COMMANDE DU ROBOT MULTISOFT   *
*   -----                       *
*
*   Cette commande est implantable *
*   dans la memoire du T07.        *
*   Pour le MOS il faut changer les *
*   adresses du PORTA et du CRA.    *
*
*****

* POUR LE MOS : PORTA EQU $A7F0

E7F0 PORTA EQU $E7F0
E7F1 PORTB EQU PORTA+1

```


E7F2	CRA	EQU	PORTA+2
E7F3	CRB	EQU	PORTA+3
7C00	MOTEUR	EQU	\$7C00
7C01	SIGNE	EQU	MOTEUR+1
7C02	DEP	EQU	SIGNE+1

7C04		ORG	\$7C04
------	--	-----	--------

```

*
*----- INITIALISATION DU PIA -----
*
```

7C04	34	7F	INIT	PSHS	U,X,Y,DP,D,CC	On sauve
7C06	CC	0000		LDD	#0	
7C09	FD	E7F2		STD	CRA	Acces a DDRA et B
7C0C	CC	FF00		LDD	#\$FF00	
7C0F	FD	E7F0		STD	PORTA	PORTA=S PORTB=E
7C12	CC	0404		LDD	#\$0404	
7C15	FD	E7F2		STD	CRA	Acces a PORTA et B
7C18	4F			CLRA		
7C19	B7	E7F0		STA	PORTA	Sorties ULN a 1
7C1C	BE	7C02		LDX	DEP	X contient le DEP
7C1F	27	7B		BEQ	FIN	Si X=0 alors FIN
7C21	7D	7C01		TST	SIGNE	Si 1 alors negatif
7C24	26	27		BNE	MOINS	
7C26	CE	7C9B	PLUS	LDU	#TAB1	U charge la table
7C29	A6	C0	BOUC1	LDA	,U+	A charge la bobine
7C2B	B4	7C00		ANDA	MOTEUR	On ajoute le moteu
7C2E	B7	E7F0		STA	PORTA	En Sortie
7C31	8A	01		ORA	#1	Mise a 0 de D1
7C33	B7	E7F0		STA	PORTA	En sortie
7C36	30	1F		LEAX	-1,X	Un de moins
7C38	27	5F		BEQ	FIN	Si 0 alors FIN
7C3A	BC	0008		CMPX	#8	Si x>8 alors on
7C3D	24	04		BHS	ACC1	accelere, sinon
7C3F	8D	4B		BSR	TEMPO2	on decelere.
7C41	20	02		BRA	DEC1	
7C43	8D	2F	ACC1	BSR	TEMPO1	
7C45	1183	7CA3	DEC1	CMPU	#TAB1+B	Est-on en fin de
7C49	27	DB		BEQ	PLUS	table? Si oui
7C4B	26	DC		BNE	BOUC1	PLUS sinon BOUC1

7C4D	CE	7CA3	MOINS	LDU	#TAB1+B	Meme chose que PLUS mais cette fois la table sens contraire.
7C50	A6	C2	BOUC2	LDA	, -U	
7C52	B4	7C00		ANDA	MOTEUR	
7C55	B7	E7F0		STA	PORTA	
7C58	8A	01		ORA	#1	
7C5A	B7	E7F0		STA	PORTA	
7C5D	30	1F		LEAX	-1, X	
7C5F	27	38		BEQ	FIN	
7C61	8C	0008		CMPX	#8	
7C64	24	04		BHS	ACC2	
7C66	8D	21		BSR	TEMPO2	
7C68	20	02		BRA	DEC2	
7C6A	8D	08	ACC2	BSR	TEMPO1	
7C6C	11B3	7C9B	DEC2	CMPU	#TAB1	
7C70	27	DB		BEQ	MOINS	
7C72	26	DC		BNE	BOUC2	
7C74	10BE	7CA3	TEMPO1	LDY	VITESS	ACCELERATION
7C7B	10BC	7CA5		CMPY	VITMAX	
7C7C	27	06		BEQ	BOUC3	
7C7E	31	30		LEAY	-\$0010, Y	
7C80	10BF	7CA3		STY	VITESS	
7C84	31	3F	BOUC3	LEAY	-1, Y	
7C86	26	FC		BNE	BOUC3	
7C88	39			RTS		
7C89	10BE	7CA3	TEMPO2	LDY	VITESS	DECELERATION
7C8D	31	AB 10		LEAY	\$0010, Y	
7C90	10BF	7CA3		STY	VITESS	
7C94	31	3F	BOUC0	LEAY	-1, Y	
7C96	26	FC		BNE	BOUC0	
7C98	39			RTS		
7C99	35	FF	FIN	PULS	U, X, Y, DP, D, CC, PC	
7C9B		CE	TAB1	FCB	\$CE	
7C9C		4E		FCB	\$4E	
7C9D		6E		FCB	\$6E	
7C9E		2E		FCB	\$2E	
7C9F		3E		FCB	\$3E	
7CA0		1E		FCB	\$1E	
7CA1		9E		FCB	\$9E	
7CA2		8E		FCB	\$8E	
7CA3		0300	VITESS	FDB	\$0300	

```

7CA5      0280 VITMAX FDB      $0280

          0000      END

```

00000 Total Errors

```

ACC1      7C43
ACC2      7C6A
BOUC0     7C94
BOUC1     7C29
BOUC2     7C50
BOUC3     7C84
CRA       E7F2
CRB       E7F3
DEC1      7C45
DEC2      7C6C
DEP       7C02
FIN       7C99
INIT      7C04
MOINS     7C4D
MOTEUR    7C00
PLUS      7C26
PORTA     E7F0
PORTB     E7F1
SIGNE     7C01
TAB1      7C9B
TEMPO1    7C74
TEMPO2    7CB9
VITESS    7CA3
VITMAX    7CA5

```

Programme III-4. -- Programme Multisoft (version MO5 objet).

3. FONCTIONNEMENT GENERALISE

On trouvera ci-après le programme de commande du Robot à partir d'un TO 7 (+ extension 16 K) ou d'un TO 7-70, tel qu'il est commercialisé par la société Multisoft.

3.1 La partie Basic de ce programme gère l'écran, le clavier et le crayon optique.

Dans ce programme deux modes de commande sont autorisés :

– Le mode **manuel** qui permet l'apprentissage, la mise au point des déplacements moteur après moteur.

Pour passer en mode manuel, il faut pointer MA.

– Le mode **automatique** qui permet l'exécution du programme Robot en mémoire en mode cycle par cycle avec un maximum de 255 cycles programmables.

	DOIG	POIG	MAIN	COUD	EPAU	BASE
1						
2						
3						
4						
5						
6						
7						
MA AU IN SU						

Vous commencez un nouveau travail.

Vous avez le choix entre :

- MAnuel
- AUtomatique
- INsertion
- SUPpression.

Pointez avec le crayon

	DOIG	POIG	MAIN	COUD	EPAU	BASE
1						
2						
3						
4						
5						
6						
7						

Si vous avez choisi le mode MAnuel,



votre ligne de commande s'efface

Fig. III-3. – Ecran de commande de la gestion du Robot.

Pour passer en mode automatique, il faut pointer AU.

L'écran présente un tableau à deux entrées, soit le nom du moteur, soit le numéro du déplacement.

Dans un déplacement, plusieurs moteurs (tous éventuellement) peuvent être commandés simultanément.

En **mode manuel**, on choisit une case (intersection moteur, déplacement) en la pointant avec son crayon optique, un encadrement noir se déplace alors dans la case pointée.

	DOIG	POIG	MAIN	COUD	EPAU	BASE
1						
2						
3						
4						
5						
6						
7						
↓ ↑ D F E S						

Vous choisissez le moteur et le déplacement en pointant la case. Le rectangle noir se déplace. Vous pointez alors la commande D et vous entrez votre déplacement. Vous attendez que le rectangle réapparaisse et vous recommencez.

en bleu (-200) en rouge (+200)

	DOIG	POIG	MAIN	COUD	EPAU	BASE
1		200				
2		200				
3	1/7					
4					100	
5					100	
6	B9					
7						
↓ ↑ D F E S						

en noir sur fond vert

Vous avez déjà entré 6 déplacements dont :

- un branchement sur test ligne 3
- un branchement absolu de la ligne 6 vers la ligne 9.

Pointez alors la ↓ pour passer aux 6 déplacements suivants.

Fig. III-4. – Mode manuel.

Il faut toujours attendre que cet encadrement soit présent avant de changer de commande.

On peut alors pointer la commande D pour « déplacement ». La fenêtre de commande est alors effacée et le système demande le nombre de **pas** souhaité. On tape ce nombre en le faisant toujours précéder du signe + ou -. Le nombre de pas doit être compris entre - 9999 et + 9999.

Si la rotation est positive, la valeur du déplacement s'inscrit en rouge sur l'écran, sinon elle s'inscrit en bleu. Simultanément à la commande « ENTREE » le moteur tourne du nombre de pas indiqué. Pour mettre au point un déplacement, on peut réactiver la commande **D** et taper le **nombre de pas supplémentaire** que l'on veut effectuer. Le résultat s'inscrit dans sa case et le moteur tourne du nombre de pas supplémentaires tapé.

On peut utiliser cette commande autant de fois qu'il est nécessaire.

Cette commande permet également de taper les instructions suivantes :

T20 par exemple, qui permet de lancer une temporisation de 20 secondes. Le maximum est de 120 secondes et le minimum de une seconde. Cette commande doit toujours être tapée dans la première colonne (DOIG), et être la seule d'un déplacement.

B6 par exemple, qui force le robot à se brancher sur le déplacement n° 6. On peut se brancher en avant ou en arrière. Cette commande doit également être tapée dans la première colonne (DOIG), et être la seule d'un déplacement.

1/6 par exemple, qui teste l'entrée numéro 1 (E1) et qui branche le robot au déplacement n° 6 si cette entrée est à « 1 » (logique, c'est-à-dire au + Val) ou qui continue en séquence dans le cas contraire. On peut évidemment faire des tests sur l'une quelconque des 8 entrées.

Cette commande doit obligatoirement être tapée dans la première colonne (DOIG), et être la seule du déplacement.

A tout instant, on peut :

Effacer (commande **E**) toutes les cases de tous les déplacements.

Finir (commande **F**) et revenir au Basic.

Passer à la Suite (commande **S**), ce qui permet de :

- sauvegarder un programme de déplacement sur le LEP,
- charger un programme existant sur le LEP,
- corriger un programme,
- lancer un mode automatique en cycle par cycle.

Passer aux mouvements suivant ou précédant ceux présents sur l'écran par action sur les commandes ↓ ↑. En effet, on peut programmer jusqu'à 150 déplacements (et même un peu plus...), mais on ne les visualise que 6 par 6.

En **mode automatique**, on peut :

- charger un programme,

- corriger un programme,
- exécuter un programme.

Deux fonctions supplémentaires sont prévues :

IN qui permet d'insérer un nouveau déplacement entre deux autres en indiquant son numéro.

SU qui permet de supprimer le déplacement dont on donne le numéro.

On accède à ces commandes en les pointant par le crayon optique.

Le programme Basic ci-joint a été volontairement compacté pour occuper le minimum de place en mémoire, ce qui le rend difficilement lisible.

D'autre part, la liaison au Robot peut être faite directement depuis les ports A et B, afin de ne rendre nécessaire que la carte n° 1. Les liaisons devront donc être conformes à la *figure III-5*.

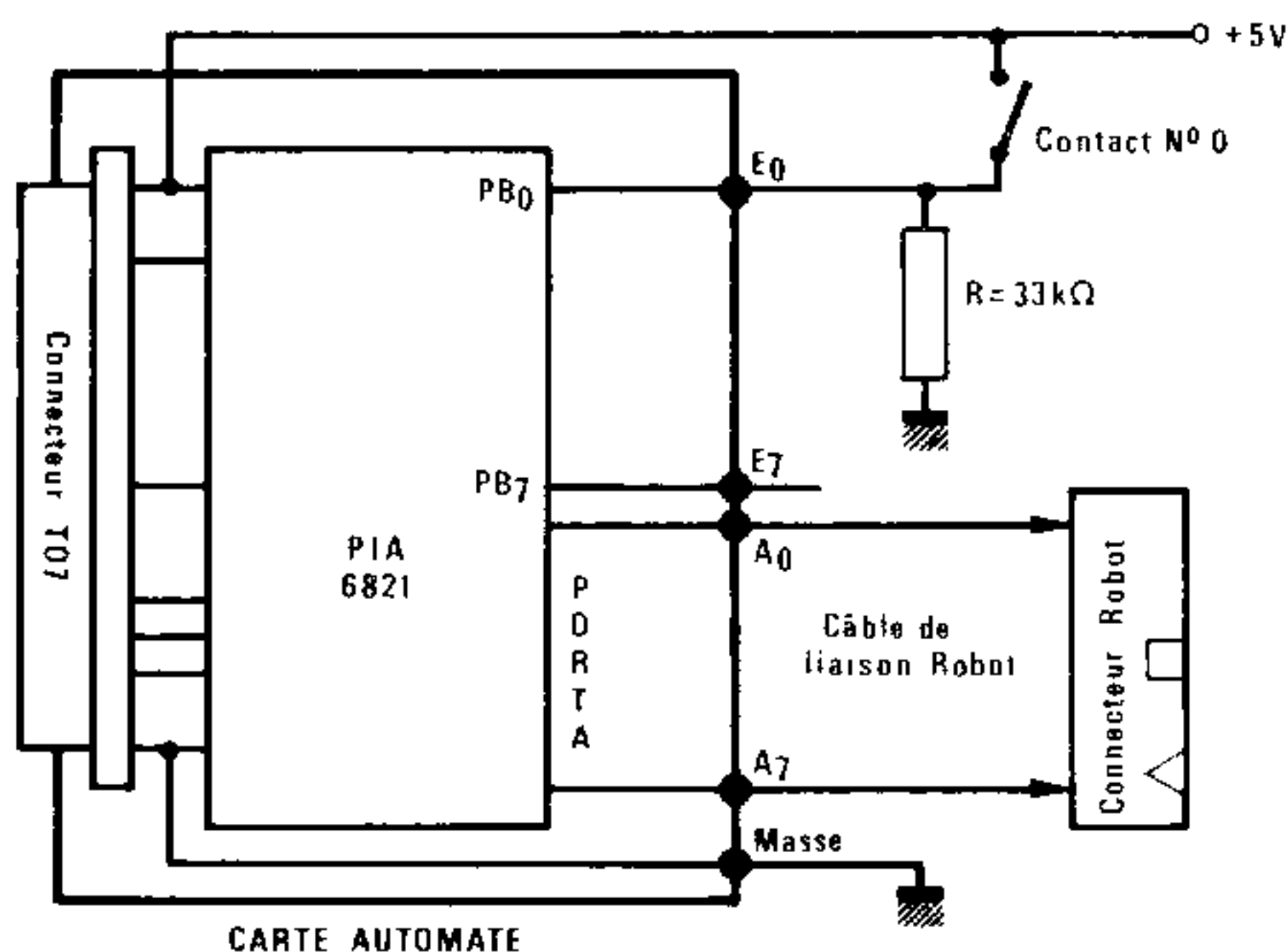


Fig. III-5. – Connexions Robot et entrées (mode 1 carte).

```

70  CONSOLE0,24:CLS:SCREEN1,0,0:ATTRB1,1:CLEAR,&HA000,2:M
EM=&HA200:BOX(0,0)-(319,199),2:LOCATE5,5,0:PRINT"GESTION
  DU ROBOT"
100 LOCATE10,7:PRINT"MULTISOFT":ATTRB0,0:LOCATE15,13:COL
OR6:PRINT"M.Oury":LOCATE5,18:COLOR1:PRINT"Copyright Mult
isoft & E.T.S.F.":LOCATE18,20:PRINT"1984":FORN=1TO2000:N
EXTN
  
```

```

140 CLS:BOX(0,0)-(319,199),2:LOCATE5,5:PRINT"Indiquez vo
tre type d'extension":ATTRB1,1:LOCATE5,10:COLOR4,6:PRINT
"1";:COLOR6,0:PRINT" - ";:COLOR0,3:PRINT"Imprimante"
180 LOCATE5,20:COLOR4,6:PRINT"2";:COLOR6,0:PRINT" - ";:C
OLOR0,3:PRINT"Automate":X$=INPUT$(1):IFVAL(X$)<1ORVAL(X$
)>2THENPLAY"MIDMIPP MIDOMI":GOTO370 ELSEATTRB0,0:IFX$="1
"THENLOADM"CASS:RDB7I.BIN" ELSEIFX$="2"THENLOADM"CASS:RD
B7A.BIN"
370 PRG1=&HA000:PRG2=&HA025:PRG3=&HA0B2:CYCLE=&HAFFF
:DEFMAX=&HA1FE:DEFGR$(0)=0,8,8,73,42,28,8,0:DEFGR$(1)=0,
8,28,42,73,8,8,0
390 EXECPRG1:SCREEN0,2,6:CLS:LOCATE0,0,0:P=1:FMAX=1:POK
EDEFMAX,6:GOSUB540
410 GOSUB450:DNERROR GOTO1720:ONC GOTO840,1300,6000,7000
:END
450 GOSUB1490:LOCATE6,23:ATTRB1,1:COLOR1,2:PRINT"MA";:LO
CATE12,23:PRINT"AU";:LOCATE18,23:PRINT"IN";:LOCATE24,23:
PRINT"SU";:ATTRB0,0:INPUTPENX,Y:IFY<22GOTO450 ELSEC=INT(
X/48):IFC=0THENPLAY"MIDOMI":GOTO450 ELSERETURN
540 CONSOLE0,20:SCREEN0,3:CLS:RESTORE620:BOXF(0,0)-(31,1
66),-3:BOXF(0,0)-(319,23),-3:BOX(0,0)-(319,166),0:FORI=1
TO6:READ X,Y:LINE(X,0)-(X,166),0:LINE(0,Y)-(319,Y),0:NEX
TI
620 DATA 32,24,80,48,128,72,176,96,224,120,272,144
630 COLOR0,3:RESTORE680:GOSUB700:FORI=1TO6:READC$,X,Y:LO
CATEX,Y,0:PRINTC$:NEXTI:RETURN
680 DATA DOIG,5,1,POIG,11,1,MAIN,17,1,COUD,23,1,EPAU,29,
1,BASE,35,1
700 COLOR0,3:FORI=1TO6:LOCATE1,4+3*(I-1):PRINT" ";:LOCA
TE1,4+3*(I-1):PRINTRIGHT$(STR$(6*(P-1)+I),LEN(STR$(6*(P-
1)+I))-1);:NEXTI:GOSUB1570:RETURN
750 LOCATE6,23:ATTRB1,1:COLOR1,2:PRINTGR$(0);:LOCATE12,2
3:PRINTGR$(1);:LOCATE18,23:PRINT"D";:LOCATE24,23:PRINT"F
";:LOCATE30,23:PRINT"E";:LOCATE36,23:PRINT"S";:ATTRB0,0:
RETURN
840 GOSUB1490
870 GOSUB1170:IFX>170ANDX<272THENGOTO390 ELSEIFX>272THEN
GOSUB1530:GOTO410
880 GOSUB1490
900 LOCATE6,22,0:INPUT"TAPEZ VOTRE COMMANDE : ",D$:IFD$=
""GOTO870
920 IFLEN(D$)>5THENPLAY"DOREMIFASOPPLASI":GOTO880
950 AD=MEM+4+18*(D+6*(P-1))+3*M-21:MVT=PEEK(AD+1)*256+PE
EK(AD+2):SIG=PEEK(AD):R$=RIGHT$(D$,LEN(D$)-1)
985 IFLEFT$(D$,1)="T"THENGOSUB5000:GOTO1150
987 IFLEFT$(D$,1)="B"THENGOSUB5200:GOTO1150
988 IFMID$(D$,2,1)="/"THENGOSUB5300:GOTO1150

```



```

990 IFSIG=0THENIFLEFT$(D$,1)="+"THENMVT=MVT+VAL(R$)*2
1000 IFSIG=0THENIFLEFT$(D$,1)="-"THENMVT=MVT-VAL(R$)*2
1010 IFSIG=1THENIFLEFT$(D$,1)="+"THENMVT=-MVT+VAL(R$)*2
1020 IFSIG=1THENIFLEFT$(D$,1)="-"THENMVT=-MVT-VAL(R$)*2
1040 IFMVT<0THENPOKEAD,1ELSEPOKEAD,0
1050 POKEAD+1,INT(ABS(MVT)/256):POKEAD+2,ABS(MVT)-INT(ABS(MVT)/256)*256:POKEMEM,M*2
1070 IFLEFT$(D$,1)="+"THENPOKEMEM+1,0ELSEPOKEMEM+1,1
1080 POKE MEM+2,INT(VAL(R$)/128) : POKE MEM+3,2*VAL(R$)-INT(VAL(R$)/128)*256
1100 LOCATE6*M-1,D*3+1,0:COLOR1,3:PRINT"      ";;LOCATE6*M-1,D*3+1:IFMVT>0THENCOLOR1,3:PRINTRIGHT$(STR$(MVT/2),LEN(STR$(MVT/2))-1); ELSECOLOR4,3:PRINTRIGHT$(STR$(ABS(MVT/2)),LEN(STR$(MVT/2))-1);:COLOR0,5
1110 EXECPRG2
1150 GOSUB750:GOTO870
1170 '
1180 GOSUB1490:GOSUB750:CONSOLE21,24
1190 BOX(48*M-12,24*D+4)-(48*M+28,24*D+20),0:INPUTPENX,Y:IFY<22GOTO1200
1210 IFY>160GOTO1250
1220 BOX(48*M-12,24*D+4)-(48*M+28,24*D+20),-4:M=INT((X-32)/48)+1:D=INT((Y-24)/24)+1:GOTO1190
1250 BOX(48*M-12,24*D+4)-(48*M+28,24*D+20),-4
1260 IFX<80THENIFF<33THENP=P+1:GOSUB700:IFPMAX<P THENPMAX=P:POKEDEFMAX,6*P ELSE ELSE ELSEIFX<128THENIFF>1THENP=P-1:GOSUB700 ELSE ELSEIFX<176THENRETURN ELSEIFX<224THENGOTO1280 ELSEIFX<272THENRETURN ELSERETURN
1270 GOTO1190
1280 GOSUB1490:LOCATE5,22:PRINT"C'est vraiment fini (O/N) ?":C$=INPUT$(1):IFC$="O"THENCONSOLE0,24:SCREEN2,0,0:CLS:END ELSEGOTO1180
1300 GOSUB1490:LOCATE5,22:PRINT "1 - EXECUTION";:LOCATE5,23:PRINT "2 - CHARGEMENT";:C$=INPUT$(1):C=VAL(C$):DNC 6OTO1360,1420
1360 GOSUB1490:LOCATE5,22,0:PRINT"CORRECTIONS (O/N) :";:C$=INPUT$(1):IFC$="O"GOTO870
1370 LOCATE5,22 : PRINT "NOMBRE DE CYCLES : ";; INPUT",C$:C=VAL(C$)
1375 IFC<1THENLOCATE5,23:PRINT"C est trop petit ( min. 1 ). ";;GOTO1370
1380 IFC>255THENLOCATE5,23:PRINT"C est trop grand ( max. 255 ). ";;GOTO1370
1390 POKECYCLE,C:EXECPRG3:GOTO410
1420 '
1430 EXECPRG1:GOSUB1490:LOCATE5,22,0:INPUT"NOM DU PROGRAMME : ",N$:LOADM"CASS:"+N$:P=1:PMAX=PEEK(DEPMAX)/6:GOSUB700:GOTO1360

```

```

1490 CONSOLE21,24:SCREEN0,5:CLS:BOX(0,168)-(319,199),0:R
ETURN
1530 GOSUB1490:LOCATE5,22:PRINT"SAUVEGARDE (O/N) ?":C$=
INPUT$(1):IFC$="O"THENLOCATE5,23:INPUT"NOM DU PROGRAMME
: ",N$:SAVEM"CASS: "+N$,DEPMAX,MEM+4+108*PMAx,MEM:RETURN
ELSERETURN
1570 FOR D=1 TO 6
1580 FOR M=1 TO 6
1590 LOCATE6*M-1,3*D+1,0:COLOR1,3:PRINT"      ":AD=MEM+18
*(D+6*(F-1))+3*M-17:MVT=PEEK(AD+1)*256+PEEK(AD+2):SIG=PE
EK(AD)
1650 IFSIG=1THENCOLOR4,3:MVT=-MVT
1660 LOCATE6*M-1,3*D+1,0
1665 IFSIG=84THENCOLOR0,2:PRINT"T";RIGHT$(STR$(MVT/2),LE
N(STR$(MVT/2))-1):GOTO1680
1667 IFSIG=66THENCOLOR0,2:PRINT"B";RIGHT$(STR$(MVT),LEN(
STR$(MVT))-1):GOTO1680
1668 IFSIG=83THENCOLOR0,2:XH=PEEK(AD+1):PRINTRIGHT$(STR$
(XH),LEN(STR$(XH))-1);"/":XL=PEEK(AD+2):PRINTRIGHT$(STR
$(XL),LEN(STR$(XL))-1):GOTO1680
1670 IF MVT<>0THENPRINTRIGHT$(STR$(MVT/2),LEN(STR$(MVT/2
))-1)
1680 NEXTM,D:M=1:D=1:RETURN
1720 RESUME NEXT
5000 T=VAL(R$):POKEAD,84:POKEAD+1,0:POKEAD+2,2*T:LOCATE6
*M-1,D*3+1,0:COLOR1,3:PRINT"      ":LOCATE6*M-1,D*3+1:COLO
R0,2:PRINTD$:RETURN
5200 A=VAL(R$):POKEAD,66:POKEAD+1,0:POKEAD+2,A:LOCATE6*M
-1,D*3+1,0:COLOR1,3:PRINT"      ":LOCATE6*M-1,D*3+1:COLOR0
,2:PRINTD$:RETURN
5300 A=VAL(LEFT$(D$,1)):B=VAL(RIGHT$(D$,LEN(D$)-2)):POKE
AD,83:POKEAD+1,A:POKEAD+2,B:LOCATE6*M-1,D*3+1,0:COLOR1,3
:PRINT"      ":LOCATE6*M-1,D*3+1:COLOR0,2:PRINTD$:RETURN
6000 GOSUB1490:LOCATE6,22,0:INPUT"No du deplacement : ";
N:FORI=MEM+4+PMAx*108TOMEM+4+18*(N-1)STEP-1:POKEI+18,PEE
K(I):NEXTI:FORI=MEM+4+18*(N-1)TOMEM+3+18*N:POKEI,0:NEXTI
:GOSUB1570:GOTO410
7000 GOSUB1490:LOCATE6,22,0:INPUT"No du deplacement : ";
N:FORI=MEM+4+18*N TO MEM+4+108*(PMAx+1):POKEI-18,PEEK(I)
:NEXTI:GOSUB1570:GOTO410

```

Programme III-5. – Commande du Robot Multisoft (version TO 7).

3.2 La partie Assembleur contient trois modules :

- PROG 1, qui initialise la mémoire et le PIA ;
- PROG 2, qui correspond au mode manuel étudié précédemment ;
- PROG 3, qui est le mode automatique permettant la commande simultanée de plusieurs moteurs.

```
*****
*
*   COMMANDE DU ROBOT MULTISOFT
*   -----
*
*   Cette commande est implantable
*   dans la memoire du T07.
*   Pour le M05 il faut changer les
*   adresses du PORTA et du CRA.
*
*****
```

```
E7F0 PORTA EQU $E7F0
E7F1 PORTB EQU PORTA+1
E7F2 CRA EQU PORTA+2
E7F3 CRB EQU PORTA+3
A200 MOTEUR EQU $A200
A201 SIGNE EQU MOTEUR+1
A202 DEP EQU SIGNE+1
A204 DEB EQU MOTEUR+4
B000 NEWDEB EQU $B000
AFFF CYCLE EQU $AFFF
```

```
A000 ORG $A000
```

```
*
*----- INITIALISATION DU PIA -----
*
```

```
A000 34 7F INIT PSHS U,X,Y,DP,D,CC On sauve
A002 CC 0000 LDD #0
A005 FD E7F2 STD CRA Acces a DDRA et B
A008 CC FF00 LDD #$FF00
A00B FD E7F0 STD PORTA PORTA=S PORTB=E
A00E CC 0404 LDD #$0404
```


A011	FD	E7F2		STD	CRA	Acces a PORTA et B
A014	4F			CLRA		
A015	B7	E7F0		STA	PORTA	Sorties ULN a 1

A018	CE	A200		LDU	#MOTEUR	
A01B	6F	C0	ZERO	CLR	,U+	Remise a zero de
A01D	11B3	B000		CMPU	#CYCLE+1	toute la zone de
A021	26	FB		BNE	ZERO	memoire de depla-

A023	35	FF		PULS	U, X, Y, DP, D, CC, PC	
------	----	----	--	------	------------------------	--

*----- MODE MANUEL -----
*

A025	34	7F	DEBUT	PSHS	U, X, Y, DP, D, CC	
A027	BE	A202		LDX	DEP	
A02A	27	7B		BEQ	FIN	
A02C	7D	A201		TST	SIGNE	
A02F	26	27		BNE	MOINS	

A031	CE	A0A6	PLUS	LDU	#TAB1	
A034	A6	C0	BOUC1	LDA	,U+	
A036	BA	A200		ORA	MOTEUR	
A039	B7	E7F0		STA	PORTA	
A03C	B4	FE		ANDA	##FE	
A03E	B7	E7F0		STA	PORTA	
A041	30	1F		LEAX	-1, X	
A043	27	5F		BEQ	FIN	
A045	8C	000B		CMPX	#B	
A048	24	04		BHS	ACC1	
A04A	8D	4B		BSR	TEMP02	
A04C	20	02		BRA	DEC1	
A04E	8D	2F	ACC1	BSR	TEMP01	
A050	11B3	A0AE	DEC1	CMPU	#TAB1+B	
A054	27	DB		BEQ	PLUS	
A056	26	DC		BNE	BOUC1	

A058	CE	A0AE	MOINS	LDU	#TAB1+B	
A05B	A6	C2	BOUC2	LDA	, -U	
A05D	BA	A200		ORA	MOTEUR	
A060	B7	E7F0		STA	PORTA	
A063	B4	FE		ANDA	##FE	
A065	B7	E7F0		STA	PORTA	
A06B	30	1F		LEAX	-1, X	

A06A	27	38		BEQ	FIN
A06C	BC	0008		CMPX	#8
A06F	24	04		BHS	ACC2
A071	BD	21		BSR	TEMPO2
A073	20	02		BRA	DEC2
A075	BD	08	ACC2	BSR	TEMPO1
A077	11B3	A0A6	DEC2	CMPU	#TAB1
A07B	27	DB		BEQ	MOINS
A07D	26	DC		BNE	BOUC2
A07F	10BE	A0AE	TEMPO1	LDY	VITESS
A083	10BC	A0B0		CMPY	VITMAX
A087	27	06		BEQ	BOUC3
A089	31	30		LEAY	-\$0010,Y
A08B	10BF	A0AE		STY	VITESS
A08F	31	3F	BOUC3	LEAY	-1,Y
A091	26	FC		BNE	BOUC3
A093	39			RTS	
A094	10BE	A0AE	TEMPO2	LDY	VITESS
A09B	31	AB	10	LEAY	\$0010,Y
A09B	10BF	A0AE		STY	VITESS
A09F	31	3F	BOUC0	LEAY	-1,Y
A0A1	26	FC		BNE	BOUC0
A0A3	39			RTS	
A0A4	35	FF	FIN	PULS	U,X,Y,DP,D,CC,PC
A0A6		31	TAB1	FCB	\$31
A0A7		B1		FCB	\$B1
A0AB		91		FCB	\$91
A0A9		D1		FCB	\$D1
A0AA		C1		FCB	\$C1
A0AB		E1		FCB	\$E1
A0AC		61		FCB	\$61
A0AD		71		FCB	\$71
A0AE		0300	VITESS	FDB	\$0300
A0B0		0280	VITMAX	FDB	\$0280
* *----- MODE AUTOMATIQUE ----- *					
A0B2	34	7F	PROG3	PSHS	U,X,Y,DP,D,CC

A0B4	8E	A204	INIT2	LDX	#DEB
A0B7	108E	B000		LDY	#NEWDEB
A0BB	10BF	A1F3		STY	TAB2
A0BF	B6	A1FE	BOUC5	LDA	DEPMAX
A0C2	B7	A1FC		STA	NBDEF
A0C5	A6	80	BOUC4	LDA	,X+
A0C7	A7	A0		STA	,Y+
A0C9	BC	AFDF		CMFX	#NEWDEB-\$21
A0CC	26	F7		BNE	BOUC4
A0CE	86	06	BOUC6	LDA	#6
A0D0	B7	A1FB		STA	NBMDT
A0D3	86	01	BOUC7	LDA	#1
A0D5	B7	A1FD		STA	NBMDT
A0D8	FE	A1F3		LDU	TAB2
A0DB	A6	C0	BOUC8	LDA	,U+
A0DD	AE	C1		LDX	,U++
A0DF	81	42		CMFA	#66
A0E1	1027	00D9		LBEQ	BRANCH
A0E5	81	53		CMFA	#83
A0E7	1027	00C8		LBEQ	CONDIT
A0EB	81	54		CMFA	#B4
A0ED	1027	00A5		LBEQ	MINUT
A0F1	8C	0000		CMFX	#0
A0F4	27	67		BEQ	SUIT1
A0F6	BC	FFFF		CMFX	#\$FFFF
A0F9	27	6A		BEQ	SUIT2
A0FB	30	1F		LEAX	-1,X
A0FD	AF	5E		STX	-2,U
A0FF	81	00		CMFA	#0
A101	26	2D		BNE	MDINS1
A103	8E	A0A5	PLUS1	LDX	#TAB1-1
A106	10BE	A1F4		LDY	#TAB3-1
A10A	B6	A1FD		LDA	NBMDT
A10D	31	A6		LEAY	A,Y
A10F	A6	A4		LDA	,Y
A111	30	86		LEAX	A,X
A113	F6	A1FD		LDB	NBMDT
A116	58			ASLB	
A117	EA	B4		ORB	,X
A119	F7	E7F0		STB	PORTA
A11C	C4	FE		ANDB	#\$FE

A11E	F7	E7F0		STB	PORTA
A121	81	08		CMPA	#8
A123	27	05		BEG	SUIT3
A125	4C			INCA	
A126	A7	A4		STA	,Y
A128	20	3B		BRA	SUIT2
A12A	86	01	SUIT3	LDA	#1
A12C	A7	A4		STA	,Y
A12E	20	35		BRA	SUIT2
A130	8E	A0A5	MOINS1	LDX	#TAB1-1
A133	108E	A1F4		LDY	#TAB3-1
A137	86	A1FD		LDA	NOMOT
A13A	31	A6		LEAY	A,Y
A13C	A6	A4		LDA	,Y
A13E	30	86		LEAX	A,X
A140	F6	A1FD		LDB	NOMOT
A143	58			ASLB	
A144	EA	84		ORB	,X
A146	F7	E7F0		STB	PORTA
A149	C4	FE		ANDB	#\$FE
A14B	F7	E7F0		STB	PORTA
A14E	81	01		CMPA	#1
A150	27	05		BEG	SUIT4
A152	4A			DECA	
A153	A7	A4		STA	,Y
A155	20	0E		BRA	SUIT2
A157	86	08	SUIT4	LDA	#8
A159	A7	A4		STA	,Y
A15B	20	08		BRA	SUIT2
A15D	7A	A1FB	SUIT1	DEC	NBMOT
A160	8E	FFFF		LDX	#\$FFFF
A163	AF	5E		STX	-2,U
A165	7C	A1FD	SUIT2	INC	NOMOT
A168	F6	A1FD		LDB	NOMOT
A16B	8D	20		BSR	TEMPO3
A16D	C1	07		CMPB	#7
A16F	1026	FF68		LBNE	BOUC8
A173	F6	A1FB		LDB	NBMOT
A176	1026	FF59		LBNE	BOUC7
A17A	FF	A1F3		STU	TAB2
A17D	7A	A1FC	SUIT5	DEC	NBDEP
A180	1026	FF4A		LBNE	BOUC6

A184	7A	AFFF		DEC	CYCLE
A1B7	1026	FF29		LBNE	INIT2
A1B8	35	FF	FIN3	PULS	U, X, Y, DP, D, CC, PC
A1BD	10BE	0070	TEMPD3	LDY	##0070
A191	31	3F	BOUC9	LEAY	-1, Y
A193	26	FC		BNE	BOUC9
A195	39			RTS	
A196	34	40	MINUT	PSHS	U
A198	CE	93B5	BOUC10	LDU	##93B5
A198	33	5F	BOUC11	LEAU	-1, U
A19D	11B3	0000		CMPU	#0
A1A1	26	F8		BNE	BOUC11
A1A3	30	1F		LEAX	-1, X
A1A5	8C	0000		CMPX	#0
A1AB	26	EE		BNE	BOUC10
A1AA	35	40		PULS	U
A1AC	33	4F		LEAU	15, U
A1AE	FF	A1F3		STU	TAB2
A1B1	20	CA		BRA	SUIT5
A1B3	1F	10	CONDIT	TFR	X, D
A1B5	1E	B9		EXG	A, B
A1B7	BD	2F		BSR	CONVER
A1B9	B4	E7F1		ANDA	PORTB
A1BC	27	16		BEG	BOUC14
A1BE	1F	10	BRANCH	TFR	X, D
A1C0	CE	B000		LDU	#NEWDEB
A1C3	B6	A1FE		LDA	DEPMAX
A1C6	B7	A1FC		STA	NBDEP
A1C9	5A		BOUC13	DECB	
A1CA	27	0F		BEG	RETOUR
A1CC	33	CB 12		LEAU	18, U
A1CF	7A	A1FC		DEC	NBDEP
A1D2	20	F5		BRA	BOUC13
A1D4	33	4F	BOUC14	LEAU	15, U
A1D6	FF	A1F3		STU	TAB2
A1D9	20	A2		BRA	SUIT5
A1DB	FF	A1F3	RETOUR	STU	TAB2
A1DE	BE	A204		LDX	#DEB
A1E1	10BE	B000		LDY	#NEWDEB
A1E5	16	FEDD		LBRA	BOUC4

A1E8	86	01	CONVER	LDA	#1
A1EA	C1	00	BOUC15	CMPB	#0
A1EC	27	04		BEG	BOUC16
A1EE	48			LSLA	
A1EF	5A			DECB	
A1F0	20	FB		BRA	BOUC15
A1F2	39		BOUC16	RTS	
A1F3			TAB2	RMB	2
A1F5		01	TAB3	FCB	1
A1F6		01		FCB	1
A1F7		01		FCB	1
A1F8		01		FCB	1
A1F9		01		FCB	1
A1FA		01		FCB	1
A1FB			NBMOT	RMB	1
A1FC			NBDEP	RMB	1
A1FD			NOMOT	RMB	1
A1FE		32	DEPMAX	FCB	50
		0000		END	

00000 Total Errors

ACC1	A04E
ACC2	A075
BOUC0	A09F
BOUC1	A034
BOUC10	A198
BOUC11	A198
BOUC13	A1C9
BOUC14	A1D4
BOUC15	A1EA
BOUC16	A1F2
BOUC2	A05B
BOUC3	A0BF
BOUC4	A0C5
BOUC5	A0BF
BOUC6	A0CE
BOUC7	A0D3
BOUC8	A0DB

BOUC9	A191	NEWDEB	B000
BRANCH	A1BE	NOMOT	A1FD
CONDIT	A1B3	PLUS	A031
CONVER	A1EB	PLUS1	A103
CRA	E7F2	PORTA	E7F0
CRB	E7F3	PORTB	E7F1
CYCLE	AFFF	PROG3	A0B2
DEB	A204	RETOUR	A1DB
DEBUT	A025	SIGNE	A201
DEC1	A050	SUIT1	A150
DEC2	A077	SUIT2	A165
DEP	A202	SUIT3	A12A
DEPMAX	A1FE	SUIT4	A157
FIN	A0A4	SUIT5	A17D
FIN3	A18B	TAB1	A0A6
INIT	A000	TAB2	A1F3
INIT2	A0B4	TAB3	A1F5
MINUT	A196	TEMPO1	A07F
MOINS	A058	TEMPO2	A094
MOINS1	A130	TEMPO3	A1BD
MOTEUR	A200	VITESSE	A0AE
NBDEP	A1FC	VITMAX	A0B0
NBMOT	A1FB	ZERO	A01B

*Programme III-6. – Programme Multisoft Assembleur
(version TO 7, 1 carte).*

Comme dans l'exemple précédent, on obtient la version MO5 en changeant l'adresse du PIA, c'est-à-dire en remplaçant \$E7 par \$A7 dans cette adresse.

Mais il faut également changer l'adresse d'implantation en mémoire du programme Assembleur et commencer à l'implanter par exemple en \$6000.

Par conséquent, la version Basic devra être modifiée de la façon suivante :

```

70  CONSOLE0,24:CLS:SCREEN1,0,0:ATTRB1,1:CLEAR,&H6000,2:M
EM=&H6200:BOX(0,0)-(319,199),2:LOCATE5,5,0:PRINT"GESTION
  DU ROBOT"
370  PROG1=&H6000:PROG2=&H6025:PROG3=&H60B2:CYCLE=&H6FFF
:DEPMAX=&H61FE:DEFGR$(0)=0,8,8,73,42,28,8,0:DEFGR$(1)=0,
8,28,42,73,8,8,0

```

3.3 Dans le cas où les deux cartes sont utilisées, ce sont les lignes S0 à S7 (sorties de l'ULN 2803A) qui commandent le Robot. Le programme Assembleur est donc modifié, car il faut inverser les octets de commande. On obtient alors le programme ci-dessous :

```

*****
*
*   COMMANDE DU ROBOT MULTISOFT
*   -----
*
*   Cette commande est implantable
*   dans la memoire du T07.
*   Pour le M05 il faut changer les
*   adresses de PORTA, MOTEUR, NEWDEB
*   et CYCLE ainsi que DRG ($6000)
*
*****

```

```

E7F0 PORTA EQU $E7F0
E7F1 PORTB EQU PORTA+1
E7F2 CRA EQU PORTA+2
E7F3 CRB EQU PORTA+3
A200 MOTEUR EQU $A200
A201 SIGNE EQU MOTEUR+1
A202 DEP EQU SIGNE+1
A204 DEB EQU MOTEUR+4
B000 NEWDEB EQU $B000
AFFF CYCLE EQU $AFFF

```

TITLE ROBOT MULTISOFT

A000 DRG \$A000

```

*
*----- INITIALISATION DU PIA -----
*

```

```

A000 34 7F INIT PSHS U,X,Y,DP,D,CC On sauve
A002 CC 0000 LDD #0
A005 FD E7F2 STD CRA Acces a DDRA et B
A008 CC FF00 LDD #$FF00

```

A00B	FD	E7F0		STD	PORTA	PORTA=S PDRTB=E
A00E	CC	0404		LDD	#0404	
A011	FD	E7F2		STD	CRA	Acces a PORTA et B
A014	4F			CLRA		
A015	B7	E7F0		STA	PORTA	Sorties ULN a 1

A018	CE	A200		LDU	#MOTEUR	
A01B	6F	C0	ZERO	CLR	,U+	Remise a zero de
A01D	1183	B000		CMPU	#CYCLE+1	toute la zone de
A021	26	F8		BNE	ZERO	memoire de depla-

A023	35	FF		PULS	U,X,Y,DP,D,CC,PC	
------	----	----	--	------	------------------	--

*
----- MODE MANUEL -----
*

A025	34	7F	DEBUT	PSHS	U,X,Y,DP,D,CC	
A027	BE	A202		LDX	DEP	
A02A	27	78		BEQ	FIN	
A02C	7D	A201		TST	SIGNE	
A02F	26	27		BNE	MOINS	

A031	CE	A0A6	PLUS	LDU	#TAB1	
A034	A6	C0	BOUC1	LDA	,U+	
A036	B4	A200		ANDA	MOTEUR	
A039	B7	E7F0		STA	PORTA	
A03C	BA	01		DRA	#1	
A03E	B7	E7F0		STA	PORTA	
A041	30	1F		LEAX	-1,X	
A043	27	5F		BEQ	FIN	
A045	BC	000B		CMPX	#8	
A048	24	04		BHS	ACC1	
A04A	BD	48		BSR	TEMPD2	
A04C	20	02		BRA	DEC1	
A04E	BD	2F	ACC1	BSR	TEMPD1	
A050	1183	A0AE	DEC1	CMPU	#TAB1+8	
A054	27	DB		BEQ	PLUS	
A056	26	DC		BNE	BOUC1	

A058	CE	A0AE	MOINS	LDU	#TAB1+8	
A05B	A6	C2	BOUC2	LDA	, -U	
A05D	B4	A200		ANDA	MOTEUR	
A060	B7	E7F0		STA	PORTA	

A063	8A	01		DRA	#1
A065	B7	E7F0		STA	PORTA
A068	30	1F		LEAX	-1,X
A06A	27	3B		BEQ	FIN
A06C	8C	0008		CMPX	#B
A06F	24	04		BHS	ACC2
A071	8D	21		BSR	TEMP02
A073	20	02		BRA	DEC2
A075	8D	08	ACC2	BSR	TEMP01
A077	11B3	A0A6	DEC2	CMPU	#TAB1
A07B	27	DB		BEQ	MDINS
A07D	26	DC		BNE	BOUC2
A07F	10BE	A0AE	TEMP01	LDY	VITESS
A0B3	10BC	A0B0		CMPY	VITMAX
A0B7	27	06		BEQ	BOUC3
A0B9	31	30		LEAY	-\$0010,Y
A0BB	10BF	A0AE		STY	VITESS
A0BF	31	3F	BOUC3	LEAY	-1,Y
A091	26	FC		BNE	BOUC3
A093	39			RTS	
A094	10BE	A0AE	TEMP02	LDY	VITESS
A09B	31	A8 10		LEAY	\$0010,Y
A09B	10BF	A0AE		STY	VITESS
A09F	31	3F	BOUC0	LEAY	-1,Y
A0A1	26	FC		BNE	BOUC0
A0A3	39			RTS	
A0A4	35	FF	FIN	PULS	U,X,Y,DP,D,CC,PC
A0A6		CE	TAB1	FCB	\$CE
A0A7		4E		FCB	\$4E
A0A8		6E		FCB	\$6E
A0A9		2E		FCB	\$2E
A0AA		3E		FCB	\$3E
A0AB		1E		FCB	\$1E
A0AC		9E		FCB	\$9E
A0AD		8E		FCB	\$8E
A0AE		0300	VITESS	FDB	\$0300
A0B0		02B0	VITMAX	FDB	\$02B0

*
 *----- MODE AUTOMATIQUE -----
 *

A0B2	34	7F	PRD63	PSHS	U, X, Y, DP, D, CC
A0B4	8E	A204	INIT2	LDX	#DEB
A0B7	10BE	B000		LDY	#NEWDEB
A0BB	10BF	A1F3		STY	TAB2
A0BF	B6	A1FE	BOUC5	LDA	DEFMAX
A0C2	B7	A1FC		STA	NBDEP
A0C5	A6	80	BOUC4	LDA	, X+
A0C7	A7	A0		STA	, Y+
A0C9	9C	AFDF		CMPX	#NEWDEB-\$21
A0CC	26	F7		BNE	BOUC4
A0CE	B6	06	BOUC6	LDA	#6
A0D0	B7	A1FB		STA	NBMDT
A0D3	B6	01	BOUC7	LDA	#1
A0D5	B7	A1FD		STA	NOMDT
A0DB	FE	A1F3		LDU	TAB2
A0DB	A6	C0	BOUC8	LDA	, U+
A0DD	AE	C1		LDX	, U++
A0DF	81	42		CMPA	#66
A0E1	1027	00D9		LBEG	BRANCH
A0E5	81	53		CMPA	#83
A0E7	1027	00C8		LBEG	CONDIT
A0EB	81	54		CMPA	#84
A0ED	1027	00A5		LBEG	MINUT
A0F1	8C	0000		CMPX	#0
A0F4	27	67		BEG	SUIT1
A0F6	8C	FFFF		CMPX	#\$FFFF
A0F9	27	6A		BEG	SUIT2
A0FB	30	1F		LEAX	-1, X
A0FD	AF	5E		STX	-2, U
A0FF	B1	00		CMPA	#0
A101	26	2D		BNE	MOINS1
A103	BE	A0A5	PLUS1	LDX	#TAB1-1
A106	10BE	A1F4		LDY	#TAB3-1
A10A	B6	A1FD		LDA	NOMDT
A10D	31	A6		LEAY	A, Y

A10F	A6	A4		LDA	,Y
A111	30	86		LEAX	A,X
A113	F6	A1FD		LDB	NOMOT
A116	58			ASLB	
A117	E4	84		ANDB	,X
A119	F7	E7F0		STB	PORTA
A11C	CA	01		ORB	#1
A11E	F7	E7F0		STB	PORTA
A121	B1	08		CMPA	#8
A123	27	05		BEQ	SUIT3
A125	4C			INCA	
A126	A7	A4		STA	,Y
A128	20	3B		BRA	SUIT2
A12A	86	01	SUIT3	LDA	#1
A12C	A7	A4		STA	,Y
A12E	20	35		BRA	SUIT2
A130	8E	A0A5	MOINS1	LDX	#TAB1-1
A133	10BE	A1F4		LDY	#TAB3-1
A137	B6	A1FD		LDA	NOMOT
A13A	31	A6		LEAY	A,Y
A13C	A6	A4		LDA	,Y
A13E	30	86		LEAX	A,X
A140	F6	A1FD		LDB	NOMOT
A143	58			ASLB	
A144	E4	84		ANDB	,X
A146	F7	E7F0		STB	PORTA
A149	CA	01		ORB	#1
A14B	F7	E7F0		STB	PORTA
A14E	B1	01		CMPA	#1
A150	27	05		BEQ	SUIT4
A152	4A			DECA	
A153	A7	A4		STA	,Y
A155	20	0E		BRA	SUIT2
A157	86	08	SUIT4	LDA	#8
A159	A7	A4		STA	,Y
A15B	20	0B		BRA	SUIT2
A15D	7A	A1FB	SUIT1	DEC	NBMOT
A160	BE	FFFF		LDX	#FFFF
A163	AF	5E		STX	-2,U
A165	7C	A1FD	SUIT2	INC	NOMOT
A168	F6	A1FD		LDB	NOMOT
A16B	BD	20		BSR	TEMPO3

A16D	C1	07		CMPB	#7
A16F	1026	FF68		LBNE	BOUC8
A173	F6	A1FB		LDB	NBMOT
A176	1026	FF59		LBNE	BOUC7
A17A	FF	A1F3		STU	TAB2
A17D	7A	A1FC	SUITS	DEC	NBDEP
A180	1026	FF4A		LBNE	BOUC6
A184	7A	AFFF		DEC	CYCLE
A187	1026	FF29		LBNE	INIT2
A18B	35	FF	FIN3	PULS	U, X, Y, DP, D, CC, PC
A1BD	10BE	0070	TEMP03	LDY	##0070
A191	31	3F	BOUC9	LEAY	-1, Y
A193	26	FC		BNE	BOUC9
A195	39			RTS	
A196	34	40	MINUT	PSHS	U
A198	CE	9385	BOUC10	LDU	##9385
A19B	33	5F	BOUC11	LEAU	-1, U
A19D	11B3	0000		CMPU	#0
A1A1	26	F8		BNE	BOUC11
A1A3	30	1F		LEAX	-1, X
A1A5	8C	0000		CMPX	#0
A1A8	26	EE		BNE	BOUC10
A1AA	35	40		PULS	U
A1AC	33	4F		LEAU	15, U
A1AE	FF	A1F3		STU	TAB2
A1B1	20	CA		BRA	SUITS
A1B3	1F	10	CONDIT	TFR	X, D
A1B5	1E	89		EXG	A, B
A1B7	8D	2F		BSR	CONVER
A1B9	B4	E7F1		ANDA	PORTB
A1BC	27	16		BEG	BOUC14
A1BE	1F	10	BRANCH	TFR	X, D
A1C0	CE	B000		LDU	##NEWDEB
A1C3	B6	A1FE		LDA	DEPMAX
A1C6	B7	A1FC		STA	NBDEP
A1C9	5A		BOUC13	DECB	
A1CA	27	0F		BEG	RETOUR
A1CC	33	CB 12		LEAU	18, U
A1CF	7A	A1FC		DEC	NBDEP
A1D2	20	F5		BRA	BOUC13
A1D4	33	4F	BOUC14	LEAU	15, U

A1D6	FF	A1F3	STU	TAB2
A1D9	20	A2	BRA	SUIT5
A1DB	FF	A1F3	RETOUR STU	TAB2
A1DE	8E	A204	LDX	#DEB
A1E1	10BE	B000	LDY	#NEWDEB
A1E5	16	FEDD	LBRA	BOUC4
A1E8	86	01	CONVER LDA	#1
A1EA	C1	00	BOUC15 CMPB	#0
A1EC	27	04	BEQ	BOUC16
A1EE	48		LSLA	
A1EF	5A		DECB	
A1F0	20	F8	BRA	BOUC15
A1F2	39		BOUC16 RTS	
A1F3			TAB2 RMB	2
A1F5	01	TAB3	FCB	1
A1F6	01		FCB	1
A1F7	01		FCB	1
A1F8	01		FCB	1
A1F9	01		FCB	1
A1FA	01		FCB	1
A1FB		NBMOT	RMB	1
A1FC		NBDEP	RMB	1
A1FD		NOMOT	RMB	1
A1FE	32	DEPMAX	FCB	50
	0000		END	

00000 Total Errors

*Programme III-7. – Programme Multisoft Assembleur
(version TO 7, 2 cartes).*

L'avantage de l'utilisation des deux cartes est une protection par photo-coupleurs au niveau des entrées.

Ne pas oublier la liaison + 5 V ↔ COM afin d'alimenter les 8 résistances de 10 kΩ des sorties.

```

*****
*
*   COMMANDE DU ROBOT MULTISOFT
*   -----
*
*   Cette commande est implantable
*   dans la memoire du MO5.
*   Pour le T07 il faut changer les
*   adresses de PORTA, MOTEUR, NEWDEB
*   et CYCLE ainsi que ORG ($A000)
*
*****

```

```

A7F0 PORTA EQU $A7F0
A7F1 PORTB EQU PORTA+1
A7F2 CRA EQU PORTA+2
A7F3 CRB EQU PORTA+3
6200 MOTEUR EQU $6200
6201 SIGNE EQU MOTEUR+1
6202 DEP EQU SIGNE+1
6204 DEB EQU MOTEUR+4
7000 NEWDEB EQU $7000
6FFF CYCLE EQU $6FFF

```

```

TITLE ROBOT MULTISOFT

```

```

6000 ORG $6000

```

```

*
*----- INITIALISATION DU PIA -----
*

```

```

6000 34 7F INIT PSHS U,X,Y,DP,D,CC Dn sauve
6002 CC 0000 LDD #0
6005 FD A7F2 STD CRA Acces a DDRA et B
6008 CC FF00 LDD #$FF00
600B FD A7F0 STD PORTA PORTA=S PORTB=E
600E CC 0404 LDD #$0404
6011 FD A7F2 STD CRA Acces a PORTA et B
6014 4F CLRA
6015 B7 A7F0 STA PORTA Sorties ULN a 1

```


6018	CE	6200		LDU	#MOTEUR
601B	6F	C0	ZERO	CLR	,U+ Remise a zero de
601D	11B3	7000		CMPU	#CYCLE+1 toute la zone de
6021	26	F8		BNE	ZERO memoire de depla-
6023	35	FF		PULS	U,X,Y,DP,D,CC,PC

*
*-----MODE MANUEL-----
*

6025	34	7F	DEBUT	PSHS	U,X,Y,DP,D,CC
6027	BE	6202		LDX	DEP
602A	27	78		BEG	FIN
602C	7D	6201		TST	SIGNE
602F	26	27		BNE	MOINS

6031	CE	60A6	PLUS	LDU	#TAB1
6034	A6	C0	BOUC1	LDA	,U+
6036	B4	6200		ANDA	MOTEUR
6039	B7	A7F0		STA	PORTA
603C	BA	01		DRA	#1
603E	B7	A7F0		STA	PORTA
6041	30	1F		LEAX	-1,X
6043	27	5F		BEG	FIN
6045	8C	0008		CMPX	#8
6048	24	04		BHS	ACC1
604A	BD	48		BSR	TEMPO2
604C	20	02		BRA	DEC1
604E	BD	2F	ACC1	BSR	TEMPO1
6050	11B3	60AE	DEC1	CMPU	#TAB1+8
6054	27	DB		BEG	PLUS
6056	26	DC		BNE	BOUC1
6058	CE	60AE	MOINS	LDU	#TAB1+8
605B	A6	C2	BOUC2	LDA	, -U
605D	B4	6200		ANDA	MOTEUR
6060	B7	A7F0		STA	PORTA
6063	BA	01		DRA	#1
6065	B7	A7F0		STA	PORTA
6068	30	1F		LEAX	-1,X
606A	27	38		BEG	FIN
606C	8C	0008		CMPX	#8
606F	24	04		BHS	ACC2
6071	BD	21		BSR	TEMPO2
6073	20	02		BRA	DEC2

6075	8D	0B	ACC2	BSR	TEMP01
6077	11B3	60A6	DEC2	CMPU	#TAB1
607B	27	DB		BEQ	MOINS
607D	26	DC		BNE	BOUC2
607F	10BE	60AE	TEMP01	LDY	VITESS
6083	10BC	60B0		CMPY	VITMAX
6087	27	06		BEQ	BOUC3
6089	31	30		LEAY	-\$0010,Y
608B	10BF	60AE		STY	VITESS
60BF	31	3F	BOUC3	LEAY	-1,Y
6091	26	FC		BNE	BOUC3
6093	39			RTS	
6094	10BE	60AE	TEMP02	LDY	VITESS
609B	31	AB 10		LEAY	\$0010,Y
609B	10BF	60AE		STY	VITESS
609F	31	3F	BOUC0	LEAY	-1,Y
60A1	26	FC		BNE	BOUC0
60A3	39			RTS	
60A4	35	FF	FIN	PULS	U,X,Y,DP,D,CC,PC
60A6		CE	TAB1	FCB	\$CE
60A7		4E		FCB	\$4E
60A8		6E		FCB	\$6E
60A9		2E		FCB	\$2E
60AA		3E		FCB	\$3E
60AB		1E		FCB	\$1E
60AC		9E		FCB	\$9E
60AD		BE		FCB	\$8E
60AE		0300	VITESS	FDB	\$0300
60B0		0280	VITMAX	FDB	\$0280

*
 *----- MODE AUTOMATIQUE -----
 *

60B2	34	7F	PROG3	PSHS	U,X,Y,DP,D,CC
60B4	BE	6204	INIT2	LDX	#DEB
60B7	10BE	7000		LDY	#NEWDEB

60BB	10BF	61F3		STY	TAB2
60BF	B6	61FE	BOUC5	LDA	DEPMAX
60C2	B7	61FC		STA	NBDEP
60C5	A6	80	BOUC4	LDA	,X+
60C7	A7	A0		STA	,Y+
60C9	8C	6FDF		CMPL	#NEWDEB-\$21
60CC	26	F7		BNE	BOUC4
60CE	86	06	BOUC6	LDA	#6
60D0	B7	61FB		STA	NBMOT
60D3	86	01	BOUC7	LDA	#1
60D5	B7	61FD		STA	NOMOT
60D8	FE	61F3		LDU	TAB2
60DB	A6	C0	BOUC8	LDA	,U+
60DD	AE	C1		LDX	,U++
60DF	81	42		CMPL	#66
60E1	1027	00D9		LBEG	BRANCH
60E5	81	53		CMPL	#83
60E7	1027	00C8		LBEG	CONDIT
60EB	81	54		CMPL	#84
60ED	1027	00A5		LBEG	MINUT
60F1	8C	0000		CMPL	#0
60F4	27	67		BEG	SUIT1
60F6	8C	FFFF		CMPL	#\$FFFF
60F9	27	6A		BEG	SUIT2
60FB	30	1F		LEAX	-1,X
60FD	AF	5E		STX	-2,U
60FF	81	00		CMPL	#0
6101	26	2D		BNE	MOINS1
6103	8E	60A5	PLUS1	LDX	#TAB1-1
6106	108E	61F4		LDY	#TAB3-1
610A	B6	61FD		LDA	NOMOT
610D	31	A6		LEAY	A,Y
610F	A6	A4		LDA	,Y
6111	30	86		LEAX	A,X
6113	F6	61FD		LDB	NOMOT
6116	5B			ASLB	
6117	E4	84		ANDB	,X
6119	F7	A7F0		STB	PORTA
611C	CA	01		ORB	#1
611E	F7	A7F0		STB	PORTA
6121	81	08		CMPL	#8

6123	27	05		BEG	SUIT3
6125	4C			INCA	
6126	A7	A4		STA	,Y
6128	20	3B		BRA	SUIT2
612A	B6	01	SUIT3	LDA	#1
612C	A7	A4		STA	,Y
612E	20	35		BRA	SUIT2
6130	8E	60A5	MOINS1	LDX	#TAB1-1
6133	10BE	61F4		LDY	#TAB3-1
6137	B6	61FD		LDA	NOMOT
613A	31	A6		LEAY	A,Y
613C	A6	A4		LDA	,Y
613E	30	B6		LEAX	A,X
6140	F6	61FD		LDB	NOMOT
6143	58			ASLB	
6144	E4	B4		ANDB	,X
6146	F7	A7F0		STB	PORTA
6149	CA	01		ORB	#1
614B	F7	A7F0		STB	PORTA
614E	B1	01		CMPA	#1
6150	27	05		BEG	SUIT4
6152	4A			DECA	
6153	A7	A4		STA	,Y
6155	20	0E		BRA	SUIT2
6157	B6	0B	SUIT4	LDA	#B
6159	A7	A4		STA	,Y
615B	20	0B		BRA	SUIT2
615D	7A	61FB	SUIT1	DEC	NBMOT
6160	8E	FFFF		LDX	#\$FFFF
6163	AF	5E		STX	-2,U
6165	7C	61FD	SUIT2	INC	NOMOT
616B	F6	61FD		LDB	NOMOT
616B	BD	20		BSR	TEMPO3
616D	C1	07		CMPB	#7
616F	1026	FF6B		LBNE	BOUCB
6173	F6	61FB		LDB	NBMOT
6176	1026	FF59		LBNE	BOUC7
617A	FF	61F3		STU	TAB2
617D	7A	61FC	SUIT5	DEC	NBDEP
6180	1026	FF4A		LBNE	BOUC6
61B4	7A	6FFF		DEC	CYCLE
61B7	1026	FF29		LBNE	INIT2

618B	35	FF	FIN3	PULS	U, X, Y, DP, D, CC, PC
618D	10BE	0070	TEMPD3	LDY	##0070
6191	31	3F	BOUC9	LEAY	-1, Y
6193	26	FC		BNE	BOUC9
6195	39			RTS	
6196	34	40	MINUT	PSHS	U
6198	CE	9385	BOUC10	LDU	##9385
619B	33	5F	BOUC11	LEAU	-1, U
619D	1183	0000		CMPU	#0
61A1	26	F8		BNE	BOUC11
61A3	30	1F		LEAX	-1, X
61A5	8C	0000		CMPX	#0
61A8	26	EE		BNE	BOUC10
61AA	35	40		PULS	U
61AC	33	4F		LEAU	15, U
61AE	FF	61F3		STU	TAB2
61B1	20	CA		BRA	SUIT5
61B3	1F	10	CONDIT	TFR	X, D
61B5	1E	B9		EXG	A, B
61B7	8D	2F		BSR	CONVER
61B9	B4	A7F1		ANDA	PORTB
61BC	27	16		BEG	BOUC14
61BE	1F	10	BRANCH	TFR	X, D
61C0	CE	7000		LDU	#NEWDEB
61C3	B6	61FE		LDA	DEFMAX
61C6	B7	61FC		STA	NBDEP
61C9	5A		BOUC13	DECB	
61CA	27	0F		BEG	RETOUR
61CC	33	CB 12		LEAU	18, U
61CF	7A	61FC		DEC	NBDEP
61D2	20	F5		BRA	BOUC13
61D4	33	4F	BOUC14	LEAU	15, U
61D6	FF	61F3		STU	TAB2
61D9	20	A2		BRA	SUIT5
61DB	FF	61F3	RETOUR	STU	TAB2
61DE	BE	6204		LDX	#DEB
61E1	10BE	7000		LDY	#NEWDEB
61E5	16	FEDD		LBRA	BOUC4
61E8	86	01	CONVER	LDA	#1
61EA	D1	00	BOUC15	CMPB	#0
61EC	27	04		BEG	BOUC16
61EE	48			LSLA	

61EF	5A			DECB	
61F0	20	FB		BRA	BOUC15
61F2	39		BOUC16	RTS	
61F3			TAB2	RMB	2
61F5		01	TAB3	FCB	1
61F6		01		FCB	1
61F7		01		FCB	1
61F8		01		FCB	1
61F9		01		FCB	1
61FA		01		FCB	1
61FB			NBMDT	RMB	1
61FC			NBDEP	RMB	1
61FD			NOMDT	RMB	1
61FE		32	DEPMAX	FCB	50
		0000		END	

00000 Total Errors

*Programme III-8. – Programme Multisoft Assembleur
(version MO5, 2 cartes).*