

## Editorial

Dialectic 16.

Bouh.! Que voilà une vilaine écriture...  
Imprimante en panne, et plus de vous pour la réparer.  
La revue se dégrade à vue d'œil!

Mais j'ai de moins en moins d'articles à vous proposer.  
Je racle aujourd'hui les fonds de tiroir, grâce aux  
envois de deux adhérents qui ont eu le courage  
d'écrire un petit quelque chose.

Je suis désolé de ne rien vous proposer de mieux  
pour l'été.

Mais que sera Dialectic 17 si vous ne m'envoyez rien?

En espérant vous retrouver à la rentrée.

Delannay Bruno

Deux nouveaux Adhérents:

P. Verhoort  
Rue de L'école, 89  
5158 DAVE  
Belgique

Demeyer Eugène.  
Rue Jacques Lemaire, 20  
4500 Jupille sur Meuse  
Belgique.

J'ai remarqué en lisant le dernier numéro de la revue que je suis assez avare en commentaire. Voici des explications:

J'utilise la zone mémoire #F400 à #FFFF comme mémoire morte pour stocker des routines couramment utilisées. Pour cela j'utilise le SPL macro-assembleur pour taper mes programmes et je les assemble à la bonne adresse de cette zone mais je loge le programme assemblé temporairement en mémoire, par exemple à partir de #4000, ensuite je prend mon programmeur d'Eprouis que je vous ai fait parvenir en documentation et je transfère mon programme dans l'Eprouis. Il me suffit ensuite d'attendre le PAI, enfiler l'Eprouis, rallumer, sélectionner la banque Eprouis où se trouve le programme et exécuter celui-ci. Pour vous donner un exemple, j'ai implémenté le New Format Listing, les routines Screen Copy, Split Screen, quelques jeux simples et les routines qui suivent cette lettre. Ouf! passons à la zone #FB00 à #FAFF. Je l'utilise comme mémoire temporaire Ram. Vous pourrez noter cela dans la dernière revue, le programme MLP de Split pour l'écran. Si vous désirez utiliser une autre zone mémoire vous pouvez sans crainte.

Par ailleurs, description des deux routines ci-après. La première routine peut être appelée par:

>CALL #FB00,VAR\$ il fait mettre une valeur de "4" à "5" dans VAR\$ et la valeur retour correspond au déplacement du Joystick selon l'article suivant les routines.

Suite en page 3 →

SPL V1.1 PAGE 1

0000		TITL	'JOYSTICK EN BASIC / MLP'
0000		; ENTREE BASIC	
0000		; VAR\$ = "0" NUMERO DU PADDLE CONCERNE	
0000		; CALLM #xxxx,VAR\$ APPEL DE LA VALEUR	
0000		; VAR\$ = "0" a "8" VALEUR DU DEPLACEMENT	
0000		ORG	0F630H
F630	C5	BASIC	PUSH B
F631	FE22		CPI 22H
F633	C2AFF6		JNZ EXIT
F636	5E		MOV E,M
F637	23		INX H
F638	56		MOV D,M
F639	1A		LDAX D
F63A	FE01		CPI 1H
F63C	C2AFF6		JNZ EXIT
F63F	13		INX D
F640	1A		LDAX D
F641	FE36		CPI 36H
F643	F2AFF6		JP EXIT
F646	D630		SUI 30H
F648	FAAFF6		JM EXIT
F648	CDC0F6		CALL MLP
F64E	E6F0		ANI 0F0H
F650	2E37		MVI L 37H
F652	FEF0		CPI 0F0H
F654	CAADF6		JZ VAL
F657	FEE0		CPI 0E0H
F659	CAADF6		JZ VAL
F65C	FED0		CPI 0D0H
F65E	CAADF6		JZ VAL
F661	2E38		MVI L 38H
F663	FEC0		CPI 0C0H
F665	CAADF6		JZ VAL
F668	FEB0		CPI 0B0H
F66A	CAADF6		JZ VAL
F66D	2E34		MVI L 34H
F66F	FEA0		CPI 0A0H
F671	CAADF6		JZ VAL
F674	FE90		CPI 90H
F676	CAADF6		JZ VAL
F679	2E36		MVI L 36H
F67B	FE80		CPI 80H
F67D	CAADF6		JZ VAL
F680	FE70		CPI 70H
F682	F2ADF6		JP VAL
F685	2E33		MVI L 33H
F687	FE60		CPI 60H
F689	CAADF6		JZ VAL
F68C	FE50		CPI 50H
F68E	CAADF6		JZ VAL
F691	2E35		MVI L 35H
F693	FE40		CPI 40H
F695	CAADF6		JZ VAL
F698	2E32		MVI L 32H
F69A	FE30		CPI 30H
F69C	CAADF6		JZ VAL
F69F	2E31		MVI L 31H
F6A1	FE20		CPI 20H

SPL V1.1 PAGE 2

JOYSTICK EN BASIC / MLP

```

F6A3 CAADF6      JZ      VAL
F6A6 FE10        CPI      10H
F6A8 CAADF6      JZ      VAL
F6AB 2E30        MVI L   30H
F6AD 7D          VAL     MOV A,L
F6AE 12          STAX D
F6AF C1          EXIT    POP B
F6B0 C9          RET
F6B1 FFFF        DW      0FFFFH
F6B3 FFFF        DW      0FFFFH
F6B5 FFFF        DW      0FFFFH
F6B7 FFFF        DW      0FFFFH
F6B9 FFFF        DW      0FFFFH
F6BB FFFF        DW      0FFFFH
F6BD FFFF        DW      0FFFFH
F6BF FF          DB      0FFH
F6C0             ; ENTREE MLP
F6C0             ; ENTREE, A : VALEUR DU PADDLE 0 a 5
F6C0             ; SORTIE, A : VALEUR DU DEPLACEMENT
F6C0             ; LES AUTRES REGISTRES SONT SAUVEGARDES
F6C0 E5          MLP     PUSH H
F6C1 D5          PUSH D
F6C2 C5          PUSH B
F6C3 57          MOV D,A
F6C4 3A4000     LDA      40H
F6C7 E6F8      ANI      0F8H
F6C9 B2          ORA D
F6CA F608      ORI      8H
F6CC CD08DB     CALL    0D808H
F6CF 3E30      MVI A   30H
F6D1 0106FC     LXI B   0FC06H
F6D4 02          STAX B
F6D5 21FFFF     LXI H   0FFFFH
F6D8 2200FC     SHLD   0FC00H
F6DB 3A01FD     LDA      0FD01H
F6DE EB          S1      XCHG
F6DF 3E00      MVI A   0H
F6E1 02          STAX B
F6E2 2A00FC     LHLD   0FC00H
F6E5 CD14DE     CALL    0DE14H
F6E8 DADEF6     JC      S1
F6EB CD26DE     CALL    0DE26H
F6EE 11CEFF     LXI D   0FFCEH
F6F1 19          DAD D
F6F2 DAF8F6     JC      S2
F6F5 210000     LXI H   0H
F6F8 7C          S2      MOV A,H
F6F9 B7          ORA A
F6FA CAFFF6     JZ      S3
F6FD 2EFF      MVI L   0FFH
F6FF 3E36      S3      MVI A   36H
F701 02          STAX B
F702 3A4000     LDA      40H
F705 E6F0      ANI      0F0H
F707 CD06DB     CALL    0D806H
F70A 7D          MOV A,L
F70B C1          POP B
F70C D1          POP D

```

SPL V1.1 PAGE 3

JOYSTICK EN BASIC / MLP

```

F70D E1          POP H
F70E C9          RET
F70F FF          DB      0FFH
F710             END
BASIC :F630 VAL  :F6AD EXIT :F6AF MLP  :F6C0 S1  :F6DE S2  :F6FE
S3      :F6FF

```

→ suite de la dernière page!

On peut aussi appeler cette routine en LIT voici la marche à suivre :

```

MVI R, 4FH
CALL 0F6C0H

```

4FH à 45H valeur du Paddle concerné  
appel de la routine  
en sortie, les autres registres sont conservés  
et la valeur du déplacement se trouve en A.

Cette routine travaille comme la routine interne du DAI pour l'extraction de la valeur du Paddle. Un article paru dans la revue décrivait le moyen d'accélérer cette routine. S'utilise ce moyen dans le programme qui suit. L'appel de ce programme est identique à l'autre mais si vous analysez celui-ci, vous pourrez remarquer que le temps d'attente précédemment perdu est maintenant utilisé et ce, par le programme appelant.

Je n'ai pas essayé de chronométrer mais je pense que le gain de temps n'est pas négligeable dans certains cas.

SPL V1.1 PAGE 1 JOYSTICK EN BASIC / MLP

```

0000          TITL      'JOYSTICK EN BASIC / MLP'
0000          ; ENTREE BASIC
0000          ; VAR$ = "0" NUMERO DU PADDLE CONCERNE
0000          ; CALLM #XXXX,VAR$ APPEL DE LA VALEUR
0000          ; VAR$ = "0" a "8" VALEUR DU DEPLACEMENT
0000          ORG      0F710H
F710  C5      BASIC  PUSH B
F711  FE22          CPI      22H
F713  C28FF7       JNZ      EXIT
F716  5E          MOV E,M
F717  23          INX H
F718  56          MOV D,M
F719  1A          LDAX D
F71A  FE01          CPI      1H
F71C  C28FF7       JNZ      EXIT
F71F  13          INX D
F720  1A          LDAX D
F721  FE36          CPI      36H
F723  F28FF7       JP      EXIT
F726  D630          SUI      30H
F728  FA8FF7       JM      EXIT
F72B  CDA0F7       CALL     MLP
F72E  E6F0          ANI      0F0H
F730  2E37          MVI L   37H
F732  FEF0          CPI      0F0H
F734  CA8DF7       JZ      VAL
F737  FEE0          CPI      0E0H
F739  CA8DF7       JZ      VAL
F73C  FED0          CPI      0D0H
F73E  CA8DF7       JZ      VAL
F741  2E38          MVI L   38H
F743  FEC0          CPI      0C0H
F745  CA8DF7       JZ      VAL
F748  FEB0          CPI      0B0H
F74A  CA8DF7       JZ      VAL
F74D  2E34          MVI L   34H
F74F  FEA0          CPI      0A0H
F751  CA8DF7       JZ      VAL
F754  FE90          CPI      90H
F756  CA8DF7       JZ      VAL
F759  2E36          MVI L   36H
F75B  FE80          CPI      80H
F75D  CA8DF7       JZ      VAL
F760  FE70          CPI      70H
F762  F28DF7       JP      VAL
F765  2E33          MVI L   33H
F767  FE60          CPI      60H
F769  CA8DF7       JZ      VAL
F76C  FE50          CPI      50H
F76E  CA8DF7       JZ      VAL
F771  2E35          MVI L   35H
F773  FE40          CPI      40H
F775  CA8DF7       JZ      VAL
F778  2E32          MVI L   32H
F77A  FE30          CPI      30H
F77C  CA8DF7       JZ      VAL
F77F  2E31          MVI L   31H
F781  FE20          CPI      20H

```

SPL V1.1 PAGE 2 JOYSTICK EN BASIC / MLP

```

F783  CA8DF7       JZ      VAL
F786  FE10          CPI      10H
F788  CA8DF7       JZ      VAL
F78B  2E30          MVI L   30H
F78D  7D          VAL     MOV A,L
F78E  12          STAX D
F78F  C1          EXIT    POP B
F790  C9          RET
F791  FFFF          DW      0FFFFH
F793  FFFF          DW      0FFFFH
F795  FFFF          DW      0FFFFH
F797  FFFF          DW      0FFFFH
F799  FFFF          DW      0FFFFH
F79B  FFFF          DW      0FFFFH
F79D  FFFF          DW      0FFFFH
F79F  FF          DB      0FFH
F7A0          ; ENTREE MLP
F7A0          ; ENTREE, A : VALEUR DU PADDLE 0 a 5
F7A0          ; SORTIE, A : VALEUR DU DEPLACEMENT
F7A0          ; LES AUTRES REGISTRES SONT SAUVEGARDES
F7A0  E5          MLP     PUSH H
F7A1  D5          PUSH D
F7A2  C5          PUSH B
F7A3  67          MOV H,A
F7A4  E5          PUSH H
F7A5  0106FC       LXI B   0FC06H
F7A8  21FFFF       LXI H   0FFFFH
F7AB  EB          S1      XCHG
F7AC  3E00          MVI A   0H
F7AE  02          STAX B
F7AF  2A00FC       LHLD   0FC00H
F7B2  CD14DE       CALL   0DE14H
F7B5  DAABF7       JC     S1
F7B8  CD26DE       CALL   0DE26H
F7BB  11CEFF       LXI D   0FFCEH
F7BE  19          DAD D
F7BF  DAC5F7       JC     S2
F7C2  210000       LXI H   0H
F7C5  7C          S2      MOV A,H
F7C6  B7          ORA A
F7C7  CACCF7       JZ     S3
F7CA  2EFF          MVI L   0FFH
F7CC  3E36          S3      MVI A   36H
F7CE  02          STAX B
F7CF  3A4000       LDA     40H
F7D2  E6F0          ANI    0F0H
F7D4  CD06DB       CALL   0D806H
F7D7  D1          POP D
F7D8  7D          MOV A,L
F7D9  F5          PUSH PSW
F7DA  3A4000       LDA     40H
F7DD  E6F8          ANI    0FBH
F7DF  B2          ORA D
F7E0  F608          ORI    8H
F7E2  CD08DB       CALL   0D808H
F7E5  3E30          MVI A   30H
F7E7  02          STAX B
F7E8  21FFFF       LXI H   0FFFFH

```

SPL V1.1 PAGE 3

JOYSTICK EN BASIC / MLP

```

F7EB 2200FC      SHLD      0FC00H
F7EE 3A01FD      LDA       0FD01H
F7F1 F1         POP PSW
F7F2 C1         POP B
F7F3 D1         POP D
F7F4 E1         POP H
F7F5 C9         RET
F7F6 FFFF       DW        0FFFFH
F7F8 FFFF       DW        0FFFFH
F7FA FFFF       DW        0FFFFH
F7FC           END
BASIC :F710 VAL :F78D EXIT :F78F MLP :F7A0 S1 :F7AB S2 :F7C
S3 :F7CC

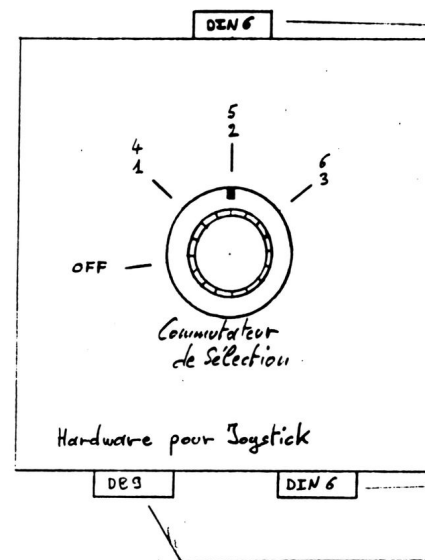
```

Pour terminer je voulais juste vous dire que je n'ai pas encore eu le temps d'examiner tous les programmes que vous m'avez envoyés; c'est vous dire! Mais j'ai eu des problèmes à lire certaines cassettes. Je compte vous les renvoyer si je trouve le temps. J'ai personnellement eu des problèmes avec mes cassettes une fois. J'ai remarqué que c'était des cassettes de mauvaise qualité et cela faisait longtemps que je n'avait pas lu dessus. Les cassettes s'étaient partiellement effacées.

En attendant de vos nouvelles, je vous salue déjà de très bonnes vacances et plein de plaisir avec mes programmes.

Bonnes salutations  
Alain F.

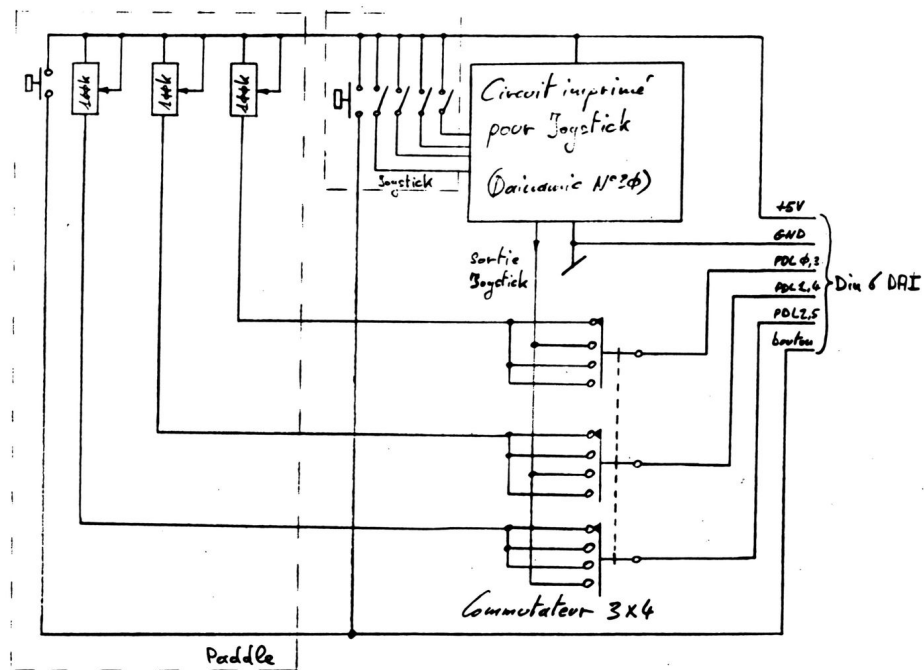
### Circuit de conversion pour Joystick.



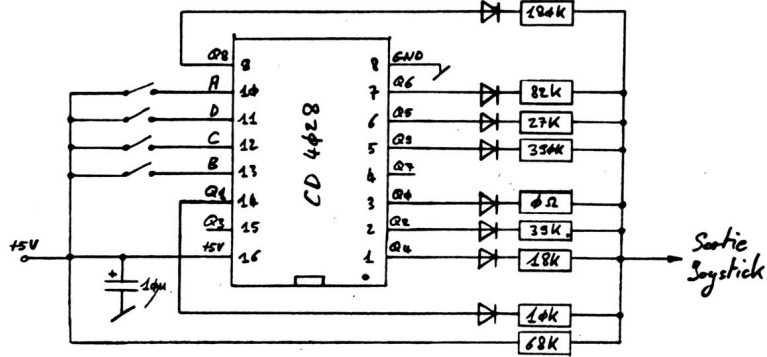
Connecteur allant à une des prises Paddle du DAI.

Boîtier contenant le circuit décrit ci-dessous.  
Un commutateur 3x4 permet de sélectionner ou non le Joystick.

Connecteur pour le Paddle traditionnel.  
Connecteur pour le Joystick.



Montage de conversion pour le Joystick. (Datiadmic N° 34)



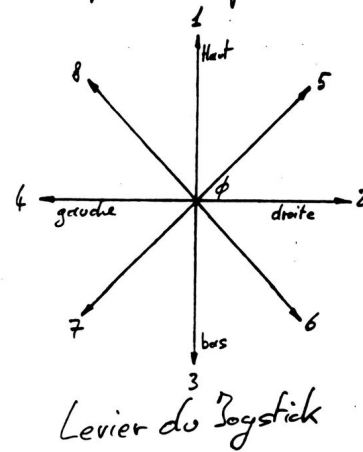
- Liste des éléments:
- 1x CD 4428
  - 1x 10µF/12V
  - 1x 100kΩ
  - 1x 82kΩ
  - 1x 27kΩ
  - 1x 33kΩ
  - 1x 1kΩ
  - 1x 33kΩ
  - 1x 18kΩ
  - 1x 18kΩ
  - 1x 18kΩ
  - 1x 1kΩ
  - 1x 33kΩ
  - 1x 1kΩ
  - 1x 18kΩ
  - 8x 1N4148

Les valeurs ci-dessus sont indicatives. Vous pouvez utiliser des résistances autorisées à 10% de tolérance mais vous risquez d'avoir des problèmes. Le mieux est de mettre une résistance en série avec un trimmer à chaque valeur et d'ajuster le tout.

Exemple:  $R = 68k\Omega$  ou choisira  $R_1 = 56k\Omega + T_1 = 22k\Omega$  etc...

Veillez à avoir de bons contacts, de bonnes soudures et testez le montage à froid et à chaud pour voir s'il vous donne les mêmes résultats. C'est également valable pour le DAI. Vous trouvez un petit programme Basic de ma conception que vous pouvez utiliser sommairement pour les tests à la page suivante.

Voici ce que cela peut vous donner: (Selon le branchement effectué!)



Correspondance entre la direction et la valeur que l'on obtient par la commande PDL.

φ	#φx
1	#1x, #2x
2	#3x
3	#5x, #6x
4	#9x, #Ax
5	#4x
6	#7x, #8x
7	#Dx, #Ex, #Fx
8	#Bx, #Cx

Pour obtenir la valeur on exécute:

$$\text{Valeur} = \text{PDL}(x) \text{ IAND } \#F\phi$$

Vous devez faire des essais sous le masque #Fφ pour voir si vous obtenez une valeur dans une plage ou à la limite. Si vous obtenez par exemple #A8, c'est bon car une variation de température ne va pas modifier la valeur au-delà de #AF ou #AF; mais si vous obtenez #AF vous risquez de voir apparaître des valeurs supérieures à #AF, par exemple la valeur #Bφ ou plus! C'est pour cela que les trimmers sont préférables! Personnellement je n'en ai pas mis. Bonne chance à ceux qui se lanceront dans ce montage.

Dernière remarque: les paddles sont utilisables en même temps que le joystick sauf sur le canal occupé par celui-ci. Tous les boutons sont en parallèle.

```

10  REM ~~~~~
15  REM *** PGM EXEMPLE POUR JOYSTICK ***
20  REM ~~~~~
25  REM VOIR LES EXPLICATIONS EN 65300
30  REM ~~~~~
40  CLEAR 256
50  COLORT 8 0 0 0
60  MODE 0
70  PRINT CHR$(12);
80  COLORG 8 0 0 0
90  MODE 1
91  X=XMAX/2.0:Y=YMAX/2.0
92  NCOL=SCRN(X,Y)
93  DOT X,Y 0
94  WAIT TIME 3
98  DOT X,Y NCOL
99  WAIT TIME 3
100 NBP=3.0
110 GOSUB 65000:REM *** SUBROUTINE POUR JOYSTICK ***
120 IF JOY=1.0 THEN GOSUB 1000
130 IF JOY=2.0 THEN GOSUB 2000
140 IF JOY=3.0 THEN GOSUB 3000
150 IF JOY=4.0 THEN GOSUB 4000
160 IF JOY=5.0 THEN GOSUB 1000:GOSUB 2000
170 IF JOY=6.0 THEN GOSUB 2000:GOSUB 3000
180 IF JOY=7.0 THEN GOSUB 3000:GOSUB 4000
190 IF JOY=8.0 THEN GOSUB 4000:GOSUB 1000
200 F=PEEK(#FD00) IAND 16
210 IF F=16.0 THEN DOT X,Y 0
300 GOTO 92
1000 REM ~~~~~
1010 REM *** EN HAUT ***
1020 REM ~~~~~
1040 Y=Y+1.0
1050 IF Y>YMAX THEN Y=YMAX
1070 RETURN
2000 REM ~~~~~
2010 REM *** A DROITE ***
2020 REM ~~~~~
2040 X=X+1.0
2050 IF X>XMAX THEN X=XMAX
2070 RETURN
3000 REM ~~~~~
3010 REM *** EN BAS ***
3020 REM ~~~~~
3040 Y=Y-1.0
3050 IF Y<0.0 THEN Y=0.0
3070 RETURN
4000 REM ~~~~~
4010 REM *** A GAUCHE ***
4020 REM ~~~~~
4040 X=X-1.0
4050 IF X<0.0 THEN X=0.0
4070 RETURN
65000 REM ~~~~~
65001 REM *** SUBROUTINE POUR JOYSTICK ***
65002 REM ~~~~~
65010 JOY=INT((PDL(NBP) IAND #F0)/16)
65020 IF JOY=0 THEN GOTO 65110
65030 IF JOY=1.0 OR JOY=2.0 THEN JOY=1.0:GOTO 65110
65040 IF JOY=3.0 THEN JOY=2.0:GOTO 65110
65050 IF JOY=4.0 THEN JOY=5.0:GOTO 65110
65060 IF JOY=5.0 OR JOY=6.0 THEN JOY=3.0:GOTO 65110
65070 IF JOY=7.0 OR JOY=8.0 THEN JOY=6.0:GOTO 65110

```

```

65080 IF JOY=9.0 OR JOY=10.0 THEN JOY=4.0:GOTO 65110
65090 IF JOY=11.0 OR JOY=12.0 THEN JOY=8.0:GOTO 65110
65100 JOY=7.0
65110 RETURN
65300 REM ~~~~~
65310 REM *** EXPLICATIONS ***
65320 REM ~~~~~
65330 REM ON POSE NBP=X, OU X VAUT LE NUMERO DE PADDLE.
65340 REM ON APPELLE LA ROUTINE EN 65000, GOSUB 65000.
65350 REM AU RETOUR DE LA ROUTINE ON ABTIENT UN CHIFFRE
65360 REM ENTRE 0 ET 8 DANS JOY.
65370 REM ~~~~~
65380 REM JOY=1 => EN HAUT
65390 REM JOY=2 => A DROITE
65400 REM JOY=3 => EN BAS
65410 REM JOY=4 => A GAUCHE
65420 REM JOY=5 => EN HAUT ET A DROITE
65430 REM JOY=6 => EN BAS ET A DROITE
65440 REM JOY=7 => EN BAS ET A GAUCHE
65450 REM JOY=8 => EN HAUT ET A GAUCHE
65460 REM ~~~~~

```