

UNE UTILISATION PÉDAGOGIQUE DE MICRO-TOUR (suite et fin)

Michel DEVAUD

3) COMMANDER LE TOUR

Nous savons maintenant lire les informations en provenance du tour et nous allons voir comment lui envoyer des ordres.

Tout d'abord, il est indispensable de savoir qu'une fonction est active lorsque le bit correspondant de l'interface parallèle est à zéro. Cela pour des raisons de sécurité: en effet, dans la technologie employée dans les micro-ordinateurs, un fil qui n'est relié à rien est au niveau logique 1. Ainsi, en cas de coupure du câble de liaison entre le tour et l'ordinateur, tout s'arrêtera.

La première commande que nous allons expérimenter est celle du moteur de broche (c'est la plus simple). Ce moteur est piloté à travers un relais commandé par le bit 7 du port A du PIA. La broche tourne si ce bit est à zéro, elle s'arrête lorsqu'il passe à un.

Nous allons écrire un programme qui mettra en marche le moteur de broche si l'on tape un « M » ou l'arrêtera si l'on tape un « A »

- En BASIC :

```
10 GOSUB 20000:REM Initialise interface
20 A$=INPUT$(1)
30 IF A$="M" THEN POKE PORTA,0
40 IF A$="A" THEN POKEPORTA,255
50 GOTO 20
RUN
```

- En LOGO :

```
POUR M
DEP 59340 0
FIN
```

```
POUR A
DEP 59340 255
FIN
INIT
```

- En LSE :

```
10 &INIT()
20 CHAINE ORDRE
30 AFFICHER 'Tapez A ou M suivi de la touche ENTREE'
40 FAIRE 50 TANT QUE .VRAI.
50 LIRE ORDRE;SI TMA(ORDRE)='M' ALORS POKE(59340,0);SI
   TMA(ORDRE)='A' ALORS POKE(59340,255)
EXECUTER A PARTIR DE 10
```

Il est à noter que la commande du moteur de broche se fait alors "grossièrement" en positionnant tous les bits du port A soit à 0, soit à 1. Seul le bit 7 étant dans ce cas utile, on pourrait ne faire varier que ce bit en utilisant des ET et OU logiques. Cette méthode, nécessaire généralement, est ici inutile: en effet, les autres bits du port A sont programmés en entrées et ne sont pas affectés par une écriture dans le port.

Nous pouvons également déplacer l'outil en faisant tourner les moteurs pas à pas. Par exemple, pour faire tourner le moteur de traînard (celui de chariot n'étant pas alimenté), il nous faut successivement écrire dans le port B:

En binaire	soit en hexadécimal	soit en décimal
11111100	FC	252
11111001	F9	249
11110011	F3	243
11110110	F6	246
11111100	FC	252
11111001	F9	249
etc.		

Cela peut conduire au programme BASIC suivant:

```

10 GOSUB 20000:REM Initialisation interface
20 DIM T(3)
30 T(0)=252:T(1)=249:T(2)=243:T(3)=246
40 BOB=0:MARCHE=0:SENS=0
50 '
100 C$=INKEY$
110 IF C$="M" THEN MARCHE=1
120 IF C$="A" THEN MARCHE=0
130 IF C$="D" THEN SENS=1
140 IF C$="G" THEN SENS=-1
150 IF MARCHE GOSUB 500
160 GOTO 100
170 '
500 REM Sous-programme de déplacement
510 BOB=BOB+SENS
520 IF BOB 3 THEN BOB=0
530 IF BOB 0 THEN BOB=3
540 POKE PORTB,T(BOB)
550 RETURN

```

Le fonctionnement de ce petit programme n'est pas difficile à comprendre: la touche "M" démarre le mouvement, la touche "A" l'arrête; les touches "D" et "G" permettent d'inverser le sens de ce mouvement.

Des précautions sont à prendre lors de l'utilisation de ce programme: il convient d'éviter de faire forcer le tour sur ses butées (interrompre le mouvement avant) et il ne faut pas oublier que la touche "A" stoppe bien le mouvement mais laisse le moteur alimenté (il chauffe et consomme plus de courant à l'arrêt qu'en rotation !). Ce dernier point peut-être facilement corrigé en modifiant la ligne 120 comme suit:

```

120 IF C$="A" THEN MARCHE=0:POKE PORTB,255

```

Il est intéressant de faire remarquer aux élèves la lenteur des déplacements: chaque appel au sous-programme de déplacement fait tourner le moteur d'un pas; il en faut 200 pour déplacer le trainard d'un millimètre ! Un langage interprété n'est donc pas utilisable pour ce type de commande.

4) ENSUITE...

Je n'ai pas, pour l'instant, dépassé ce stade avec des élèves. Le dernier programme commence à leur poser quelques difficultés.

Cependant, il est possible, en fonction du public, de proposer un certain nombre d'exercices complémentaires. Voici quelques exemples que j'ai proposés lors de formations d'adultes :

- Faire se déplacer le traînard (ou le chariot) alternativement d'une butée à l'autre; le mouvement pouvant être interrompu par un appui sur le poussoir d'arrêt programmé.
- Réaliser un programme permettant d'usiner un barreau cylindrique par passes successives.
- Piloter toutes les fonctions du tour depuis le clavier de l'ordinateur.

Il me paraît difficile de demander plus. Le moindre programme visant à pouvoir réellement utiliser le tour est un travail relativement complexe.

EN GUISE DE CONCLUSION

Comme vous avez pu le constater (et comme je l'avais annoncé en début d'article), je n'aborde absolument pas la technique du tournage et me consacre seulement au pilotage informatique. MICRO-TOUR permet de voir un certain nombre de principes de base et offre à l'élève la possibilité de commander une "vraie" machine.

A travers MICRO-TOUR, les élèves ont pu faire connaissance avec la commande de relais, de moteurs pas à pas; ils ont dû reconnaître l'état de butées et poussoirs; toutes choses qu'ils retrouveront sur d'autres parties opératives. Mais, surtout, il ont vu et ont compris comment un ordinateur pouvait agir sur le monde extérieur.

Michel DEVAUD