

9/8.21

Programmation du contrôleur de disquettes FDC-765, Check Disk pour lecteur de disquettes

Les amoureux du langage machine et de la partie matérielle des CPC trouveront dans ce chapitre divers renseignements pour programmer le composant gestionnaire de l'unité de disquette FDC-765, et ainsi créer, pourquoi pas leurs disquettes personnalisées.

Nous verrons aussi, qu'il est possible de programmer ce composant en langage Basic, relativement facilement, et vous proposerons par ailleurs un utilitaire de vérification de l'état de votre, ou vos, lecteur(s) de disquettes.

Commençons par rappeler que vous pourrez trouver déjà, dans vos divers tomes, diverses informations sur le brochage du FDC-765 en Partie 2 Chapitre 3.5, le formatage des disquettes en Partie 4 Chapitre 1.6.5 (formater une disquette sous Basic), et sur la lecture ou écriture des secteurs sur une disquette en Partie 9 Chapitre 5.8.10.

Signalons aussi que toutes les explications données ci-après sont utilisables sur les CPC-664, CPC-6128, mais aussi sur les CPC-464 munis du contrôleur et du lecteur de disquettes de marque Amstrad.

LES REGISTRES DE TRAVAIL DU FDC-765

Pour dialoguer avec le microprocesseur, le contrôleur disquettes de l'Amstrad CPC ne possède que deux registres de travail accessibles sur

deux adresses du plan des ports d'entrées/sorties : le registre d'état principal et le registre de données.

Une adresse supplémentaire, ne concernant pas le FDC-765, mais directement en relation avec le lecteur de disquettes, est réservée à la commande de mise en marche et d'arrêt des moteurs des lecteurs de disquettes. Aussi, convient-il d'en décrire le fonctionnement ici.

La commande des moteurs de disquettes

La commande des moteurs de disquettes s'effectue par l'intermédiaire de l'adresse **&FA7E**.

Le fait de transmettre la valeur 1 sur ce port met en route les moteurs de tous les lecteurs connectés à l'Amstrad, la valeur 0 (zéro) provoquant leur arrêt.

Vous pouvez même essayer en Basic par les instructions :

```
OUT &FA7E, 1      : REM MISE EN MARCHÉ MOTEURS
OUT &FA7E, 0      : REM ARRÉT DES MOTEURS
```

La programmation en assembleur s'effectuera par les instructions suivantes (à modifier selon la syntaxe de votre programme d'assemblage) :

```
*** MISE EN MARCHÉ DES MOTEURS ***
      LD A,01H          ; VALEUR DE MISE EN MARCHÉ
      LD BC,0FA7EH      ; PORT MOTEUR
      OUT (C),A         ; MISE EN MARCHÉ
;
*** ARRÉT DES MOTEURS ***
      LD A,00H          ; VALEUR D'ARRÉT
      LD BC,0FA7EH      ; PORT MOTEUR
      OUT (C),A         ; ARRÉT
```

Le registre d'état principal

Accessible à tout moment par le microprocesseur, le registre d'état principal se trouve à l'adresse **&FB7E**, et n'est utilisable qu'en lecture.

En Basic :

```
A% = INP(&FB7F)
PRINT BIN$(A%,8)
```

vous affichera la valeur binaire de ce registre.

Ce registre permet de connaître l'état du contrôleur de disquette, par test de chacun de ses bits (voir figure 1).

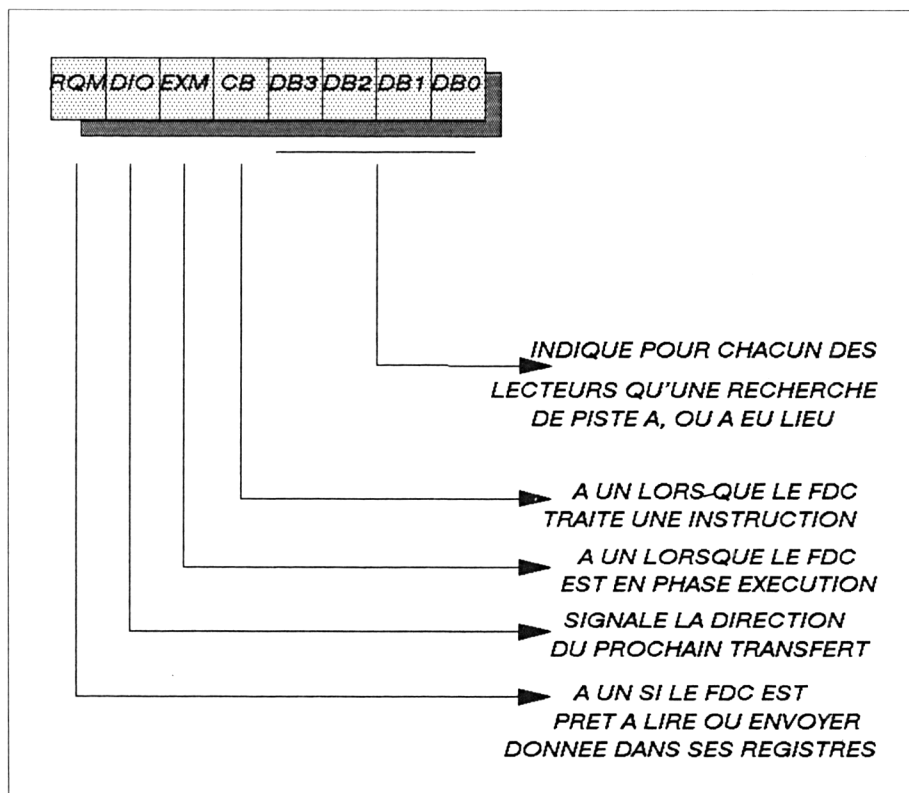


Fig. 1 : Le registre d'état principal.

Le bit **RQM** (*ReQuest for Master*) permet de déterminer si le FDC est prêt à communiquer avec le microprocesseur, s'il est positionné à 1. Une valeur 0 signale que le FDC est occupé.

Le bit **DIO** (*Data Input Output*) signale, lorsque le FDC est prêt à communiquer, le sens de transfert attendu. Positionné à 0, il signale que le FDC attend une donnée, sinon, à 1, qu'il a une donnée à fournir au microprocesseur.

Le bit **EXM** (*EXecution Mode*) est positionné à 1, dans le cas du CPC, lorsque le FDC exécute un ordre qui lui a été transmis (transfert de données avec le support magnétique). Lorsqu'il est à zéro, il signale que, si le FDC fournit encore des données, celles-ci sont les données du compte rendu de l'exécution.

Le bit **CB** (*Controler Busy*) signale en se positionnant à 1 que le FDC ne peut recevoir aucun nouvel ordre de la part du microprocesseur.

DB3, DB2, DB1, DB0 (*Disk Busy 3, 2, 1, 0*) permettent de connaître l'état des lecteurs de disquettes connectés au FDC. Normalement, quatre lecteurs peuvent être connectés au composant, mais Amstrad a choisi, sur ses CPC de limiter ces connexions à 2.

C'est donc en lisant le registre d'état principal qu'il sera possible de déterminer les différentes étapes de communications avec le FDC-765.

Le registre de données

Le registre de données du FDC-765 est accessible en lecture ou en écriture, et se trouve placé à l'adresse **&FB7F**.

Ce registre permet de donner des instructions ou ordres au FDC, de lire ou écrire des données sur la disquette, et de connaître le résultat des ordres donnés.

Trois phases seront donc réalisées, dans l'ordre suivant, à l'aide de ce registre :

- la phase instruction : le microprocesseur fournit les octets au FDC ;
- la phase exécution : des données sont échangées entre le microprocesseur et le lecteur de disquettes, par l'intermédiaire du FDC (il arrive, selon les instructions qu'aucune donnée ne soit échangée, mais qu'une action ne soit réalisée par le lecteur de disquettes ;
- la phase résultat : le FDC fournit le bilan suite à l'exécution de l'instruction demandée.

Signalons que pour certaines instructions, la phase exécution et/ou la phase résultat n'existe pas.

LES INSTRUCTIONS DU FDC

Le microprocesseur doit en premier lieu envoyer au FDC un ordre qui se traduit par un numéro d'instruction, qui sera suivi, selon l'instruction de plusieurs octets d'information (jusque 8).

Le code d'instruction

Le contenu binaire de l'octet de codage de l'instruction envoyée au FDC est représenté en figure 2.

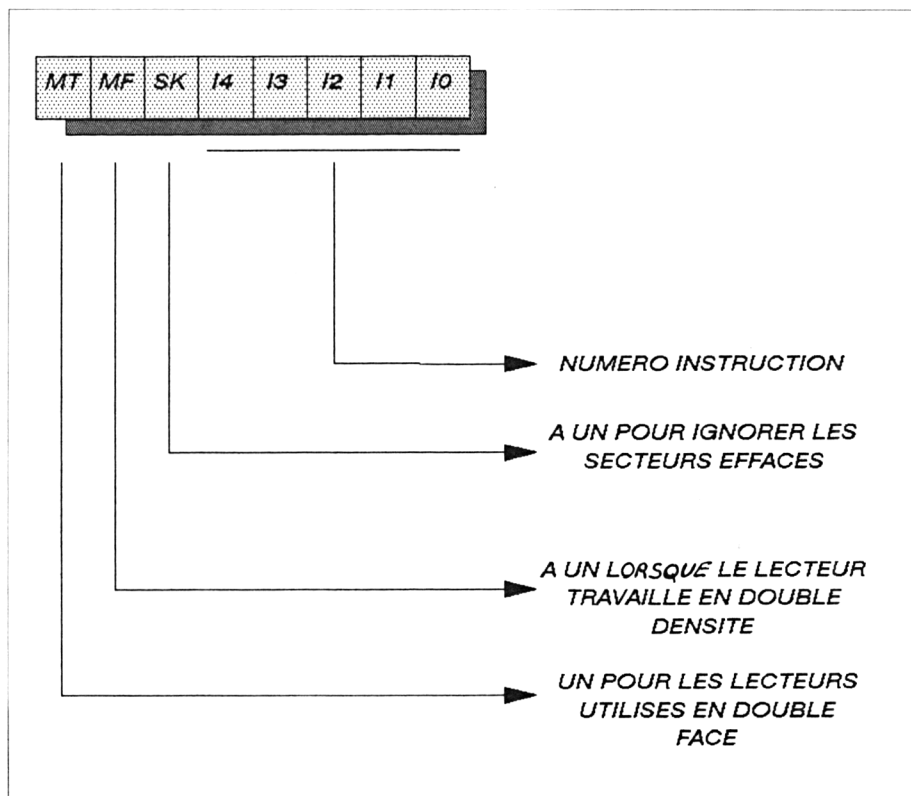


Fig. 2 : Le code instruction.

Le bit **MT** (*Multi Track*) lorsqu'il est positionné à 1 permet de prolonger l'exécution d'une instruction sur la deuxième face du lecteur de disquette, uniquement dans le cas des lecteurs possédant deux têtes (cette possibilité est utilisée par les DOS des lecteurs 700 k-octets, pas dans l'AMSDOS).

Le bit **MF** (*MFM*) signale lorsqu'il est à un que le FDC-765 doit travailler en double densité.

Le bit **SK** (*SKip*) permet, lorsqu'il est positionné à 1, d'ignorer les secteurs effacés sur la disquette.

Les bits **14**, **13**, **12**, **11** et **10** permettent de coder le numéro des instructions disponibles sur le FDC-765.

Le code d'affectation

Le code d'affectation suit, dans pratiquement toutes les instructions, sauf deux, l'octet de codage des instructions. Son contenu est représenté en figure 3.

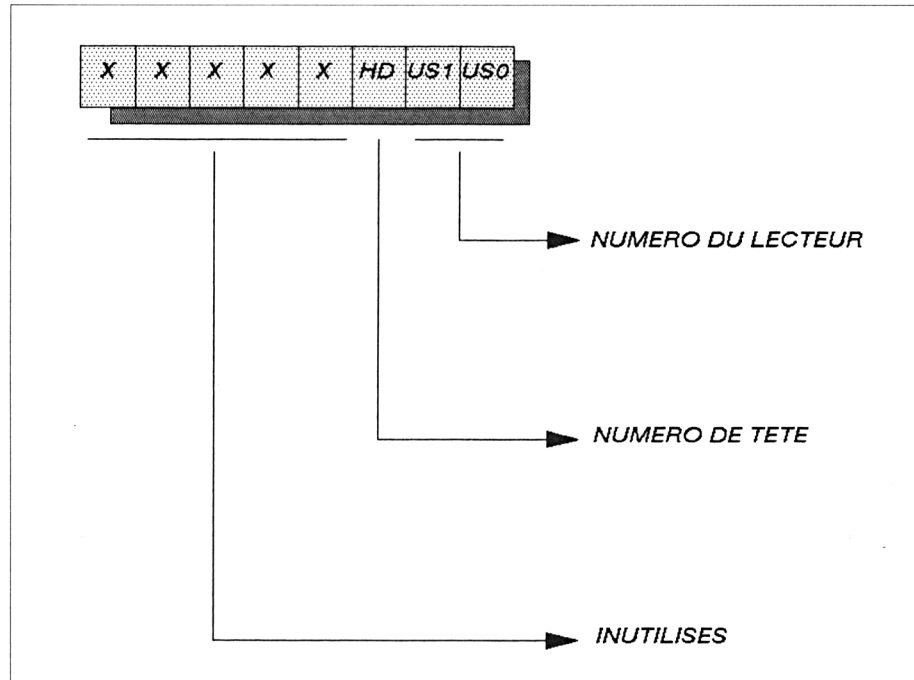


Fig. 3 : L'octet d'affectation des unités sélectionnées.

Le bit **HD** (*Head select*) permet de signaler le numéro de tête concernée par l'instruction. Il ne pourra, éventuellement, être à 1 que dans le cas des lecteurs double tête.

Les deux bits **US1** et **US2** (*Unit Select*) permettent de sélectionner les lecteurs de disquettes :

- 0 1 pour le lecteur B ;
- 1 0 pour le lecteur A.

Les instructions et leurs phases

Comme nous l'avons expliqué précédemment, la plupart des traitements effectués par le FDC-765 suivent trois phases : instruction, exécution et résultat. Nous vous proposons ci-dessous d'étudier brièvement ces trois phases. Vous y découvrirez, notamment dans la phase résultat, que quatre registres d'état peuvent être fournis par le FDC : les registres **0**, **1**, **2** et **3**, que nous détaillerons après l'étude des instructions.

• *Le paramétrage des lecteurs*

Cette instruction ne comporte que la phase instruction (voir figure 4), qui consiste à envoyer successivement le code de l'instruction (**00000011**), les temps d'impulsions de position de tête (**STEP RATE**) et d'élévation de tête (uniquement sur les lecteurs 8 pouces), puis le

décal de chargement de la tête. Il est aussi possible de commander le fonctionnement du FDC en mode **DMA** (*Direct Memory Access* = accès direct à la mémoire), mais celui-ci n'est pas prévu dans le fonctionnement sur CPC.

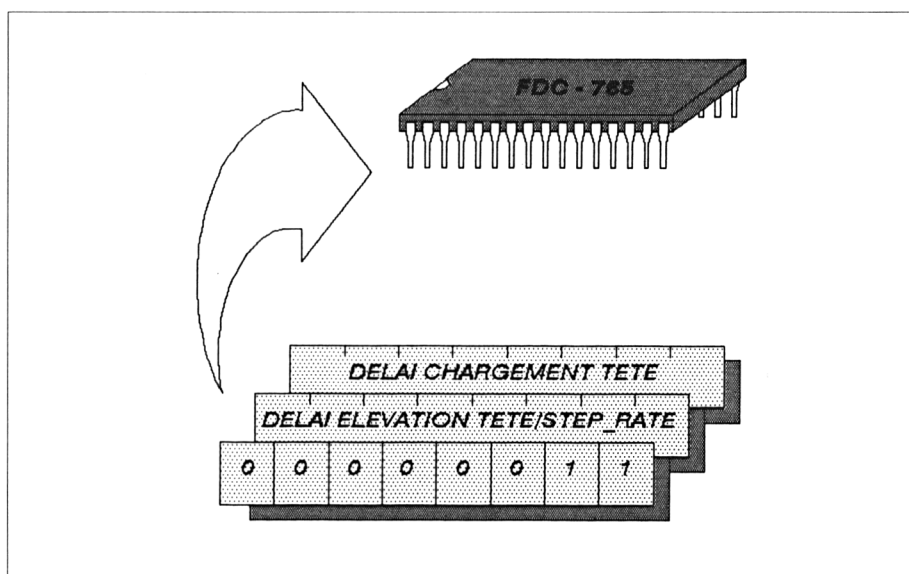


Fig. 4 : Paramètres à fournir pour spécifier les données du lecteur de disquettes.

Cette instruction est la toute première instruction nécessaire au FDC pour paramétrer le(s) lecteur(s) avec lesquels il va travailler. Cette opération est réalisée dès l'initialisation de l'AMSDOS.

- *Le formatage d'une piste*

Représentée en figure 5, la phase instruction du FDC-765 pour formater une piste nécessite six octets.

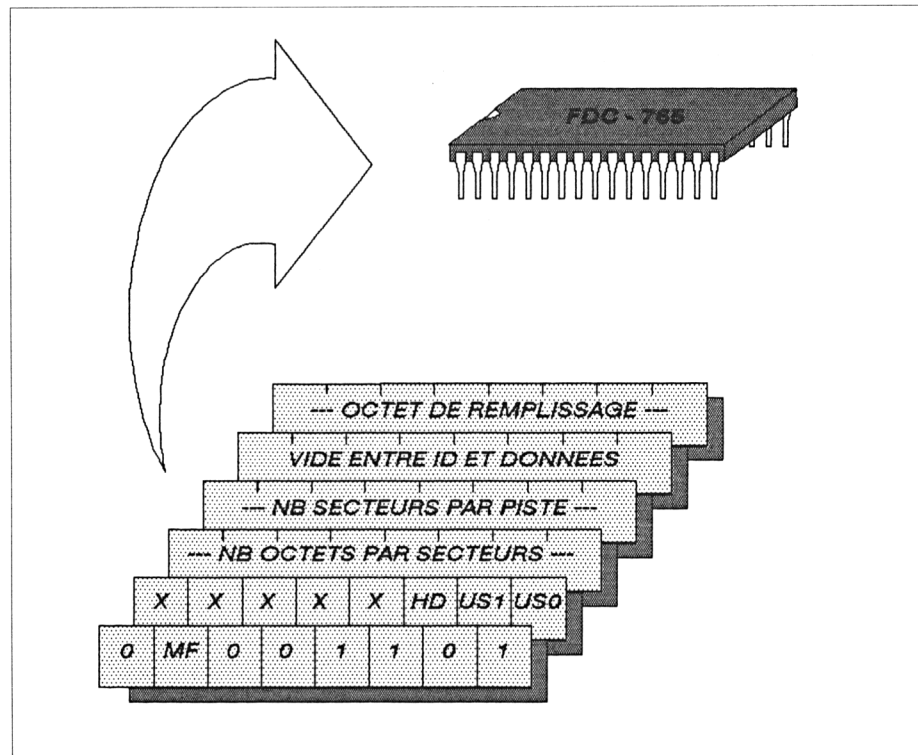


Fig. 5 : Paramètres à fournir pour l'exécution d'une instruction de formatage d'une piste.

La phase exécution consiste à formater la piste positionnée sous la tête, en tenant compte des informations fournies.

Une fois la piste formattée, le FDC-765 va entrer dans sa phase résultat, qui consiste à envoyer dans l'ordre (voir figure 6), les octets suivants :

- l'octet du registre d'état numéro 0 ;
- l'octet du registre d'état numéro 1 ;
- l'octet du registre d'état numéro 2 ;
- le numéro de la piste ;
- le numéro de tête ;
- un numéro de secteur ;
- une taille de secteur.

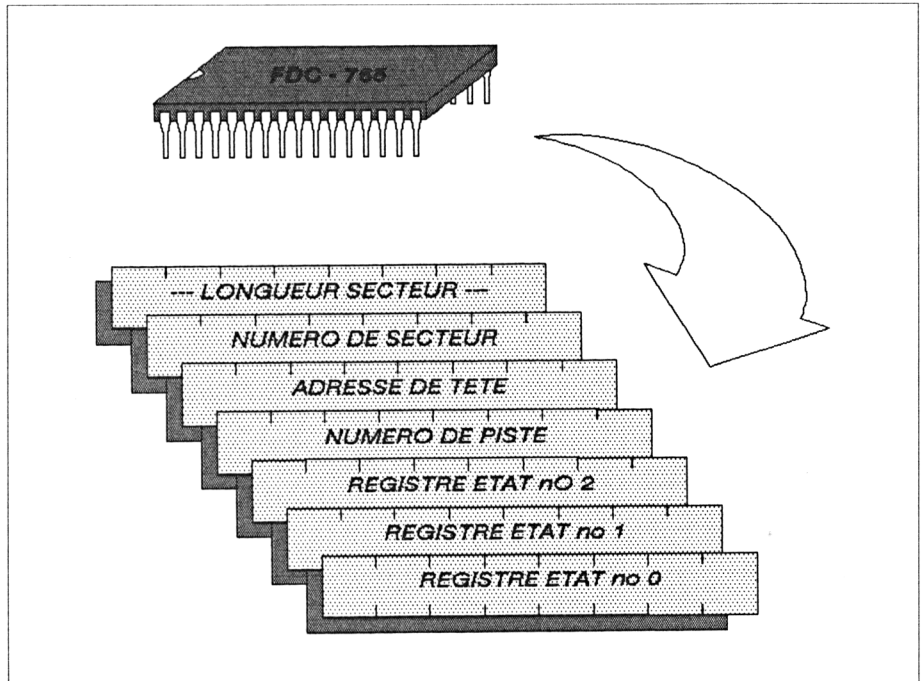


Fig. 6 : Les sept octets fournis par le FDC après exécution de l'instruction.

- *L'identification de formatage*

La figure 7 représente les deux octets de la phase instruction d'identification.

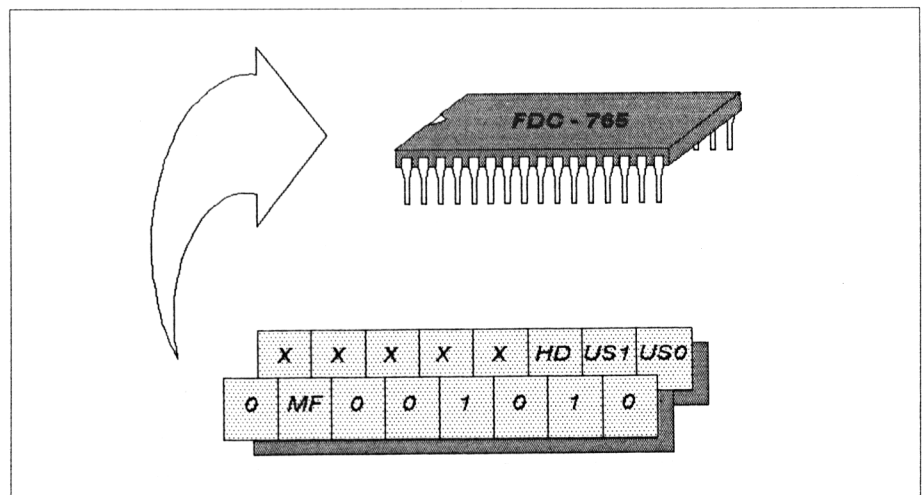


Fig. 7 : Paramètres à fournir pour l'exécution d'une instruction de lecture d'identification de formatage.

La phase exécution consistera à lire et fournir aux registres de données la prochaine identification.

La phase résultat est identique à celle représentée en figure 6.

- *Le positionnement sur la piste zéro*

Deux octets seulement sont à fournir lors de la phase instruction (voir figure 8).

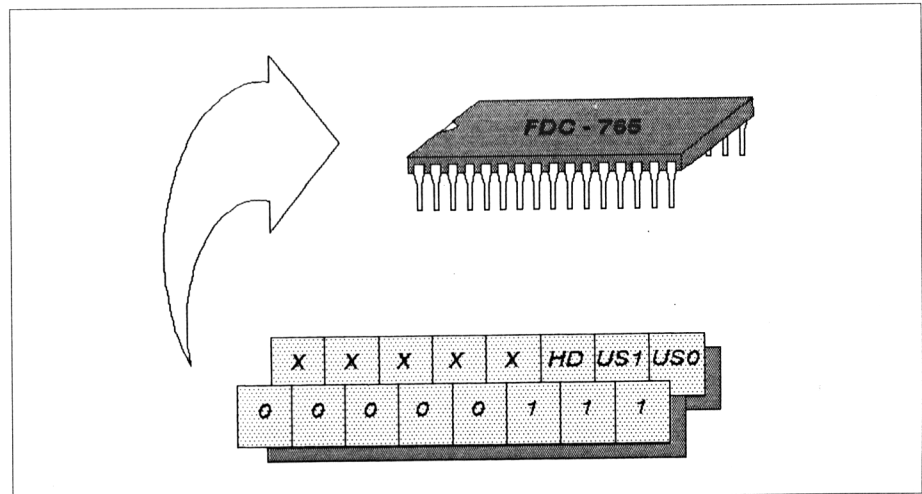


Fig. 8 : Paramètres à fournir pour l'exécution d'une instruction de positionnement sur la piste 0.

La phase exécution consistera à déplacer la tête sur la périphérie de la disquette pour la positionner sur la piste zéro.

Pas de phase résultat.

- *L'interrogation de l'état d'un lecteur*

La phase instruction nécessite deux octets (voir figure 9).

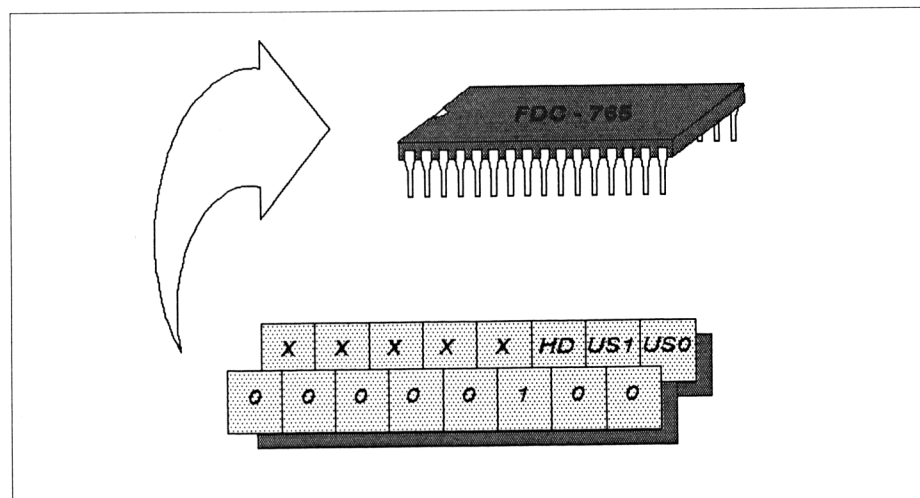


Fig. 9 : Paramètres à fournir pour l'exécution d'une instruction de demande d'état d'un lecteur.

Le FDC-765 renvoi ensuite le contenu du registre numéro 3 (voir figure 10), spécifiant les caractéristiques du lecteur, que nous étudierons dans le détail plus loin.

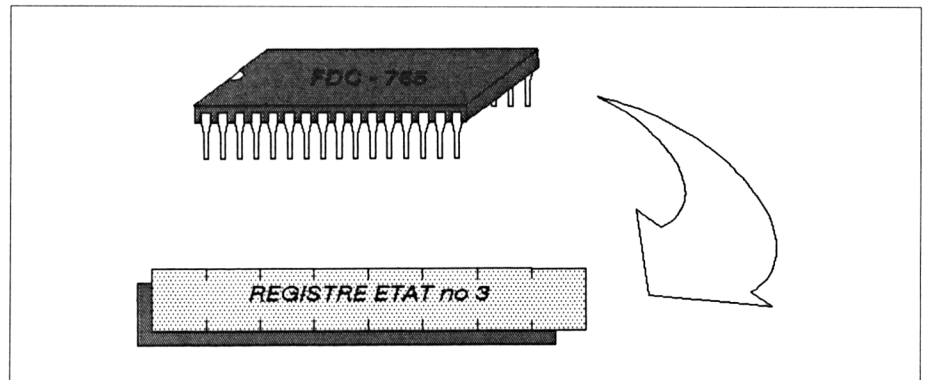


Fig. 10 : Octet résultat fourni par le FCD après exécution de l'instruction de demande d'état du lecteur de disquettes.

- *Le positionnement sur une piste particulière*

La phase instruction est représentée en figure 11.

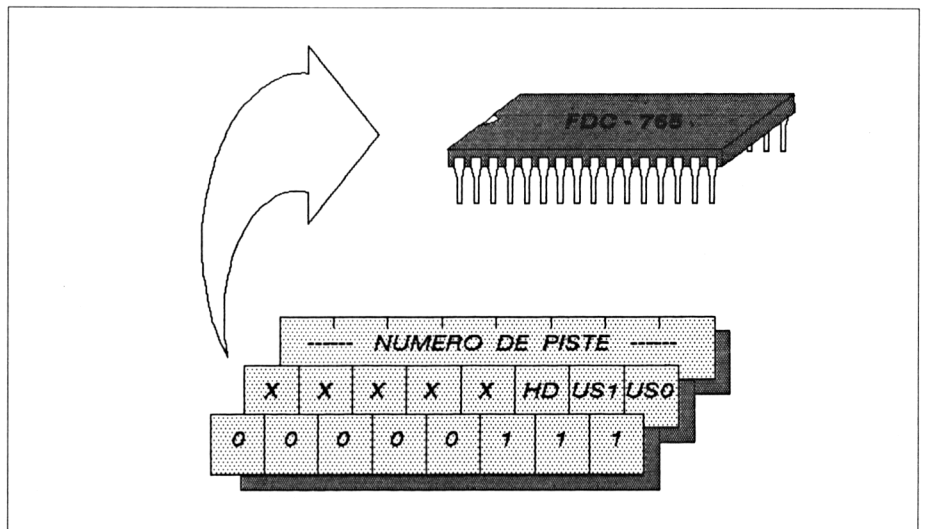


Fig. 11 : Paramètres à fournir pour l'exécution d'une instruction de positionnement sur une piste.

La phase exécution consiste à déplacer la tête sur la piste requise.

Il n'y a pas de phase résultat.

- La lecture d'une piste complète

La phase instruction est représentée en figure 12.

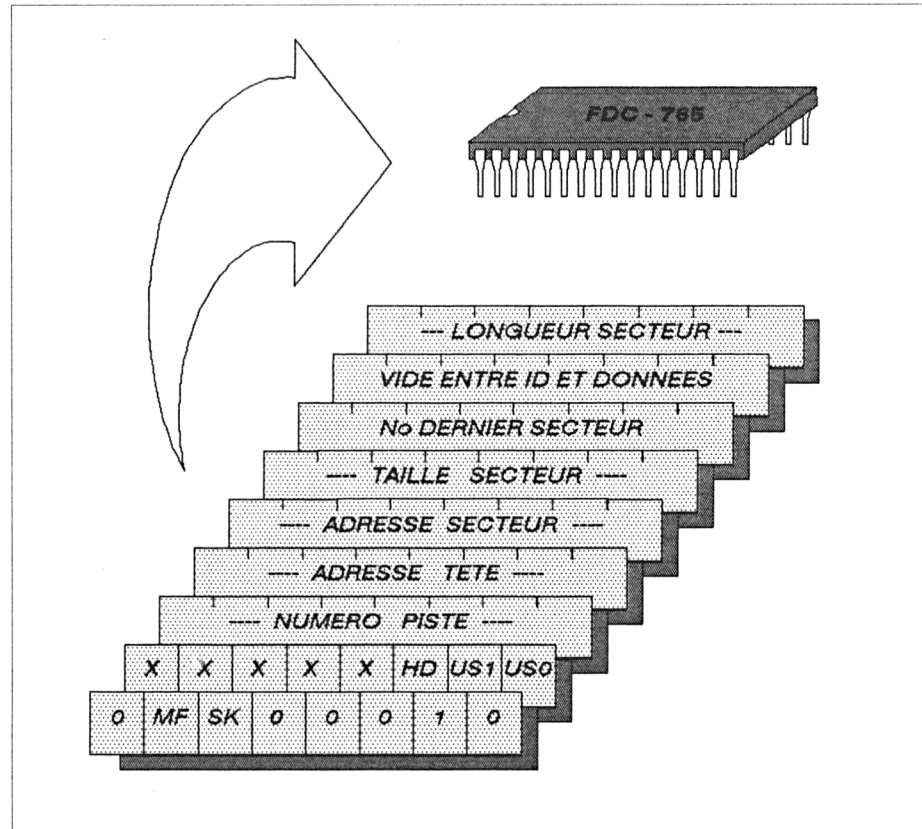


Fig. 12 : Paramètres à fournir pour l'exécution d'une instruction de lecture d'une piste par le FDC-765.

La phase exécution consiste à transférer le contenu complet de la piste sous la tête, depuis le trou d'index, jusqu'à la rencontre du dernier secteur, ou de la détection pour la deuxième fois du trou d'index.

La phase résultat est identique à celle de la figure 6.

- La lecture d'un secteur

La phase instruction est représentée en figure 13.

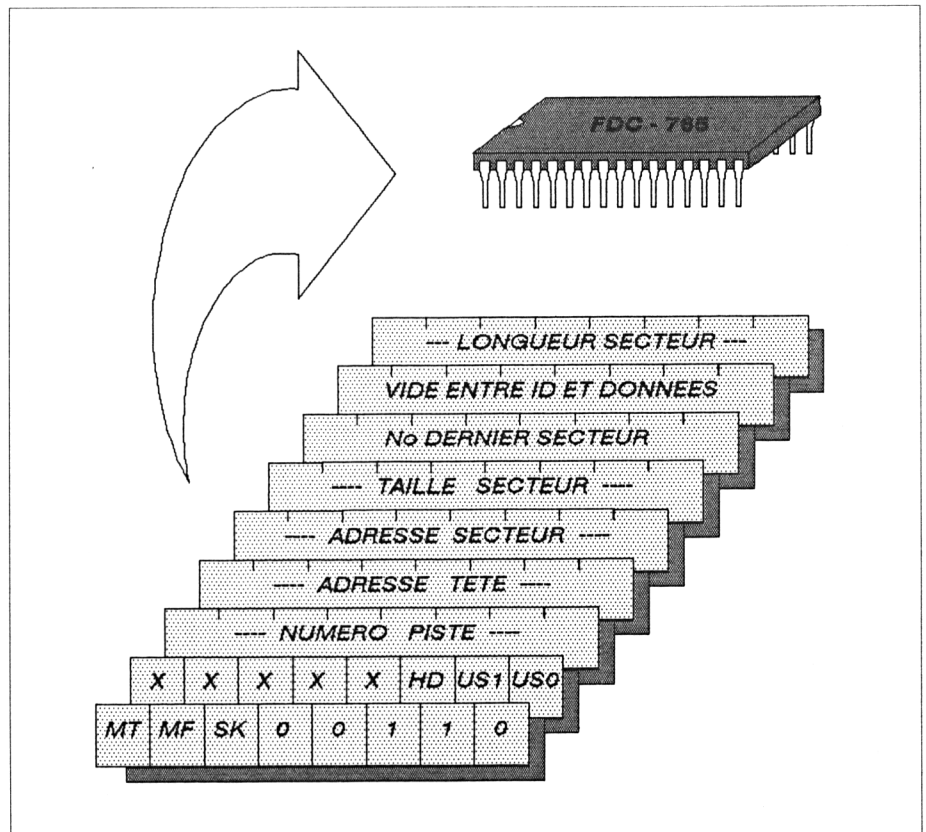


Fig. 13 : Paramètres à fournir pour l'exécution d'une instruction de lecture par le FDC-765.

La phase exécution consiste à transférer vers le microprocesseur les données d'un secteur donné.

La phase résultat est identique à celle de la figure 6.

- *L'écriture d'un secteur*

La phase instruction est représentée en figure 14.

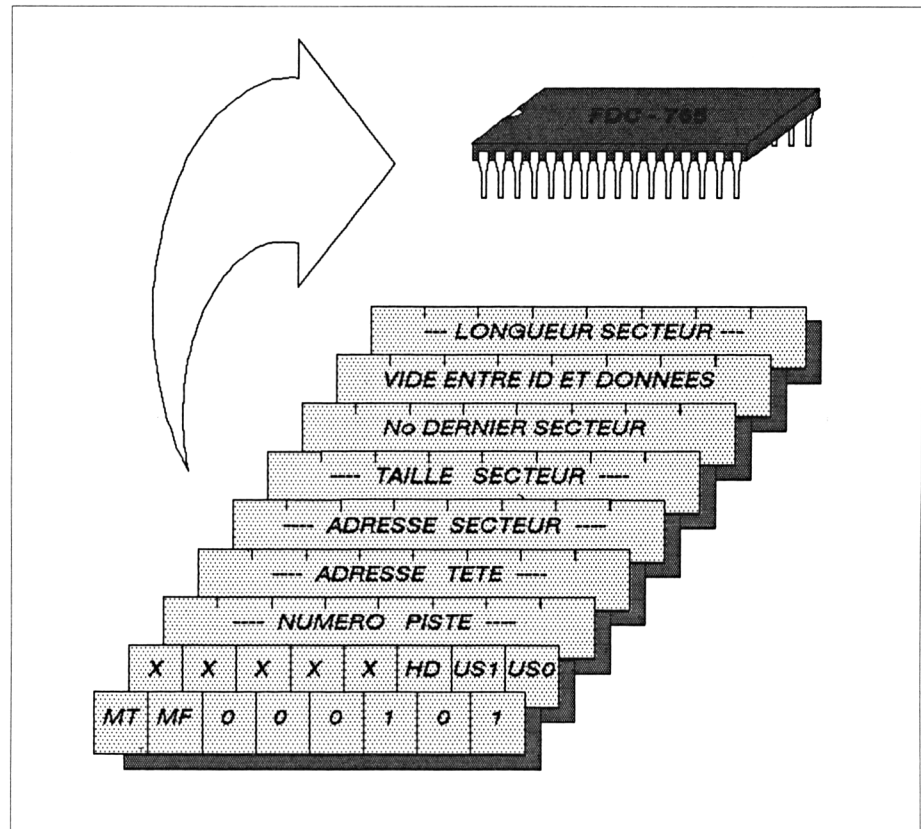


Fig. 14 : Paramètres à fournir pour l'exécution d'une instruction d'écriture par le FDC-765.

La phase exécution consiste à transférer sur le support magnétique, vers le secteur concerné, le contenu d'un secteur fourni par le microprocesseur au FDC.

La phase résultat est identique à celle de la figure 6.

- La lecture d'un secteur effacé

La phase instruction est représentée en figure 15.

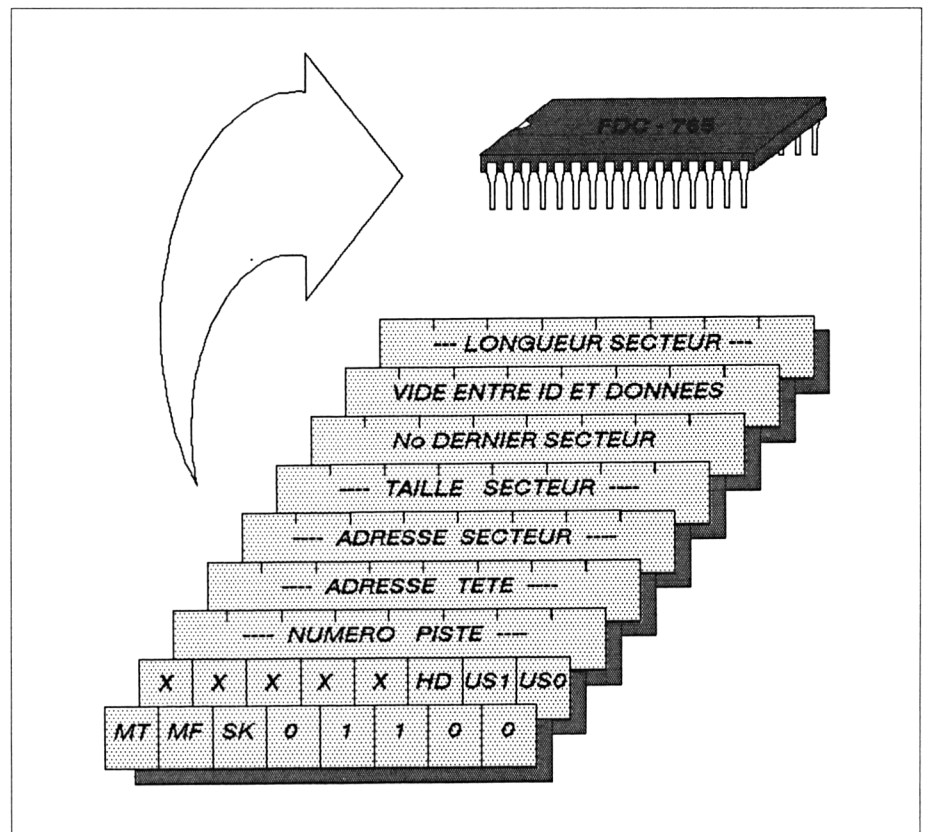


Fig. 15 : Paramètres à fournir pour l'exécution d'une instruction de lecture de données effacées par le FDC-765.

La phase exécution consiste à transférer vers le microprocesseur les données déclarées d'un secteur effacé.

La phase résultat est identique à celle de la figure 6.

- L'écriture d'un secteur effacé

La phase instruction est représentée en figure 16.

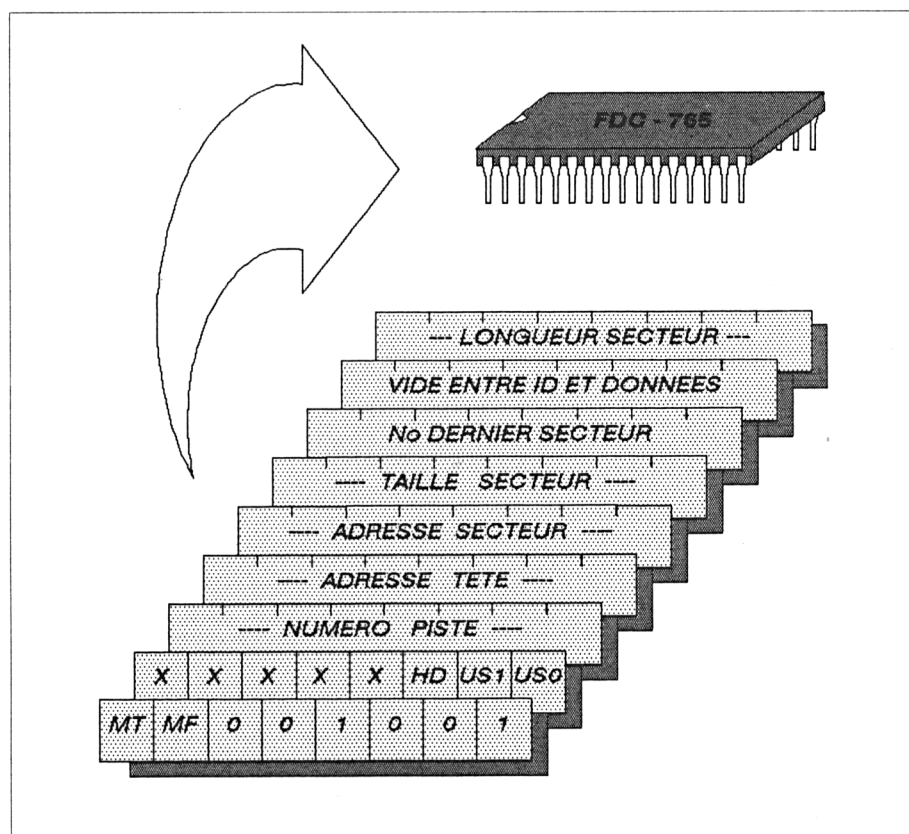


Fig. 16 : Paramètres à fournir pour l'exécution d'une instruction d'écriture de données effacées par le FDC-765.

La phase exécution consiste à transférer sur le support magnétique le contenu d'un secteur en le marquant effacé.

- Les tests de secteurs

Les trois phases instructions de chacun des tests possibles (égalité de secteur, secteur inférieur ou égal, ou secteur supérieur ou égal) sont représentées, respectivement en figures 17, 18 et 19.

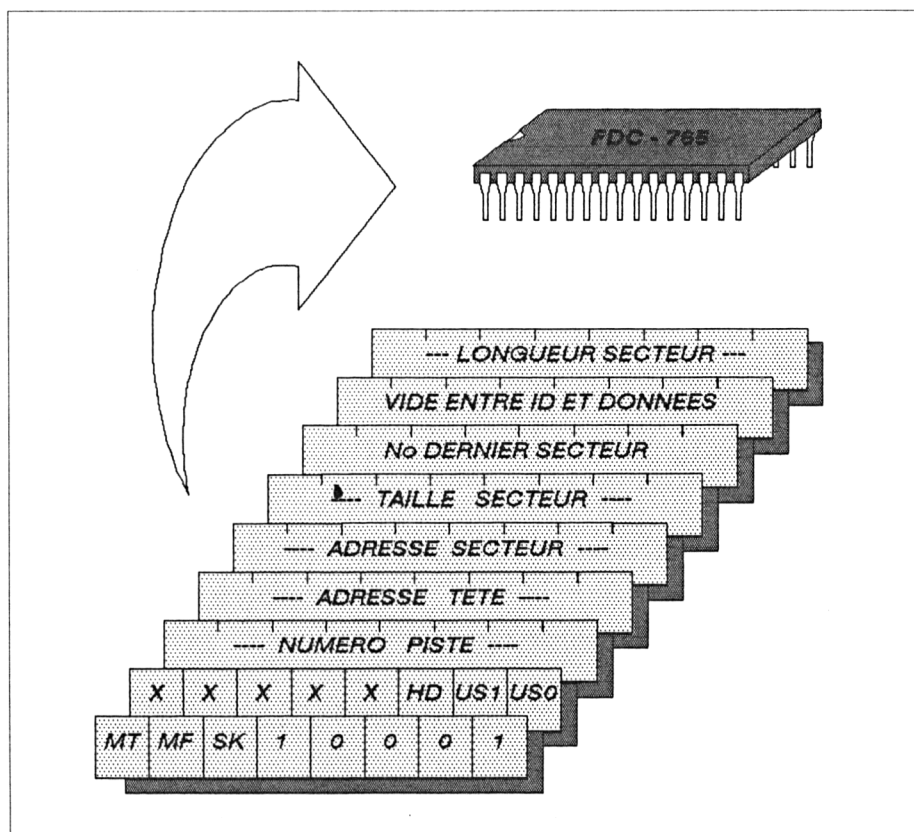


Fig. 17 : Paramètres à fournir pour l'exécution d'une instruction de test d'égalité sur un secteur.

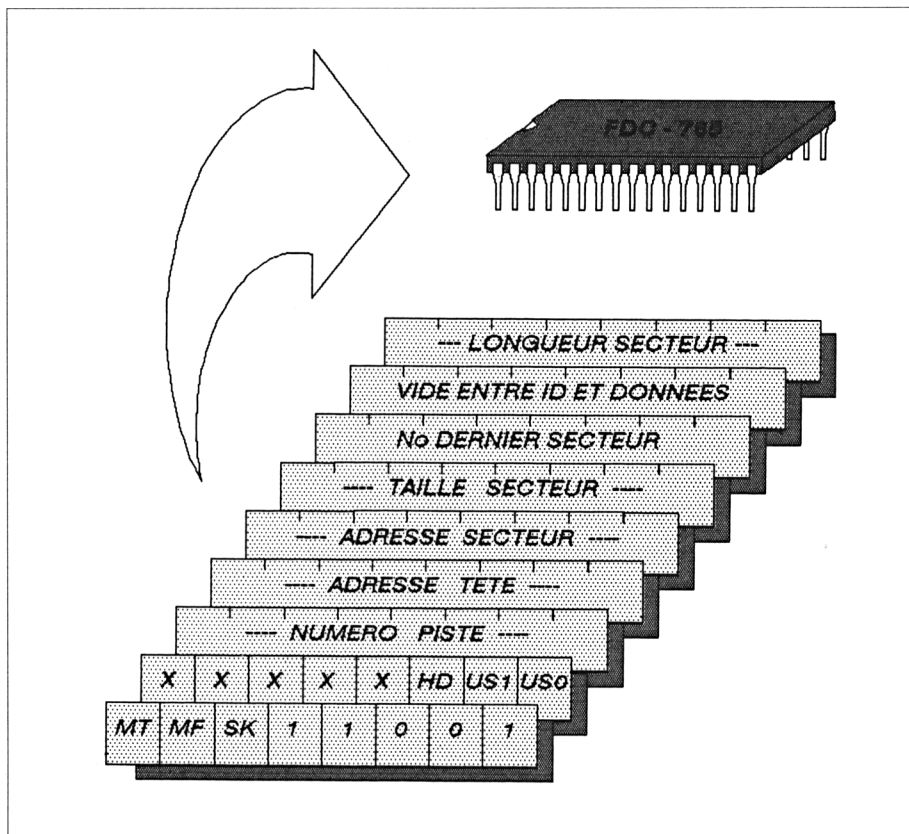


Fig. 18 : Paramètres à fournir pour l'exécution d'une instruction de test d'infériorité (ou égal) sur un secteur.

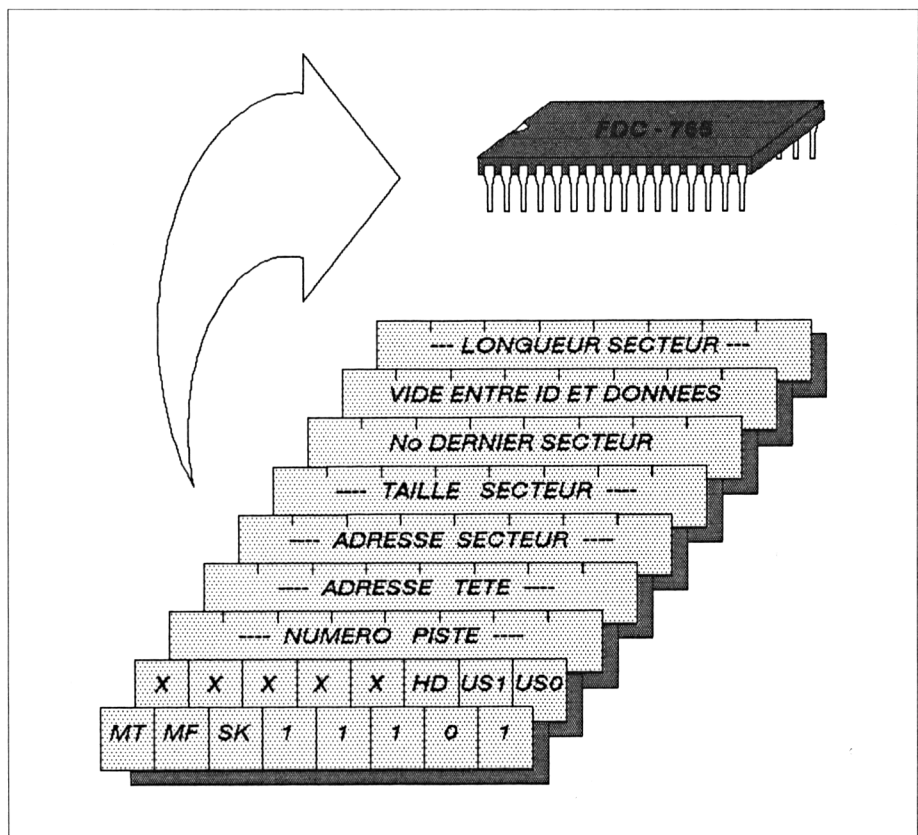


Fig. 19 : Paramètres à fournir pour l'exécution d'une instruction de test de supériorité (ou égal) sur un secteur.

La phase exécution consiste à prendre un à un les octets du secteur spécifié, et de les comparer avec chacune des données, réclamées une à une aussi, au microprocesseur.

La phase résultat est identique à celle de la figure 6.

- *La demande d'état des interruptions*

La phase instruction est représentée en figure 20.

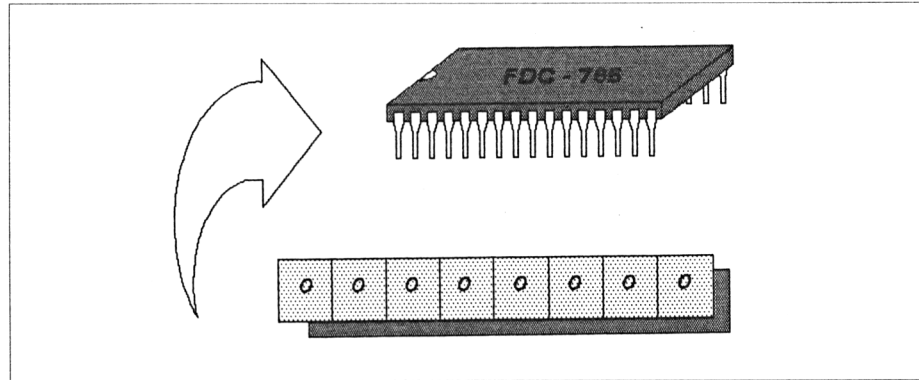


Fig. 20 : Paramètres à fournir pour l'exécution d'une instruction de demande d'état d'interruption.

Il n'y a pas de phase exécution.

La phase résultat est représentée en figure 21.

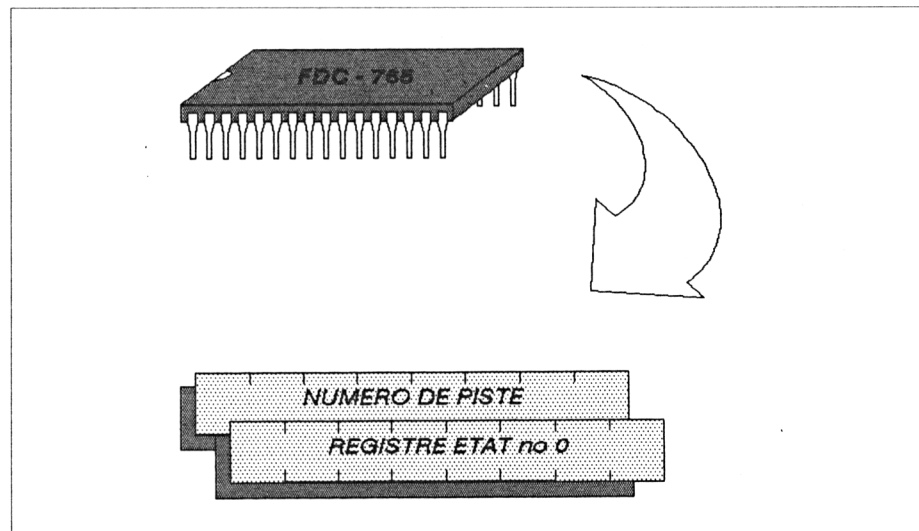


Fig. 21 : Octets résultat fourni par le FDC après exécution de l'instruction de demande d'état d'interruption.

Les registres d'état de la phase résultat

- *Le registre 0*

Ce registre est fourni suite à une demande d'état d'interruption, et est affecté à chaque fois qu'une modification concernant le lecteur de disquette apparaît.

Son contenu est spécifié en figure 22.

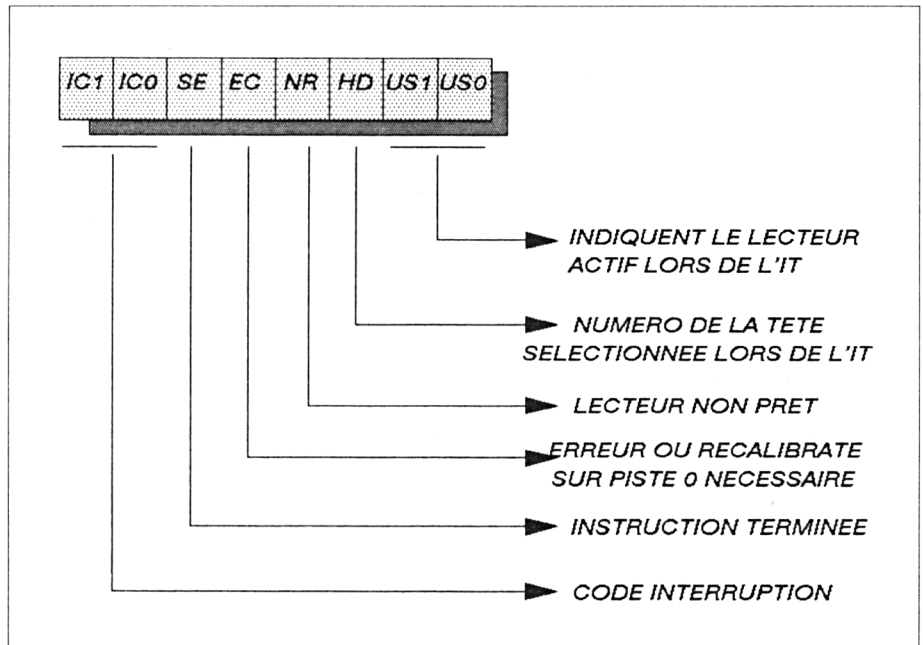


Fig. 22 : Le registre d'état 0.

Les bits **IC1** et **IC0** (*Interrupt Code*) indiquent l'état de déroulement d'une instruction :

IC1 IC0 = 11 : apparaît lors d'un défaut sur un lecteur de disquette, ou du retrait d'une disquette d'un lecteur concerné par une instruction.

IC1 IC0 = 10 : signale que l'instruction envoyée au FDC-765 n'est pas légale, ou encore que la demande d'état d'interruption n'a pas lieu d'être car aucune interruption n'est apparue.

IC1 IC0 = 01 : apparaît lorsqu'une instruction n'a pas pu s'effectuer correctement.

IC1 IC0 = 00 : indique qu'une instruction a été correctement exécutée.

Le bit **SE** (*Seek End*) est placé à 1 dès qu'une instruction est terminée.

Le bit **EC** (*Equipment Check*) est placé à 1 si le lecteur de disquette est positionné en erreur.

Le bit **NR** (*Non Ready*) est placé à 1 lors de l'exécution d'une instruction de lecture ou d'écriture alors que le lecteur de disquette n'est pas prêt.

Le bit **HD** est l'image de la tête en cours de sélection (0 pour la tête numéro 0 et 1 pour la tête numéro 1).

Les bits **US1** et **US0** indiquent le numéro de lecteur actif lors de l'apparition de l'interruption (voir le paragraphe concernant le code d'affectation des instructions).

- *Le registre 1*

Ce registre informe sur le déroulement de la phase exécution (voir son contenu en figure 23).

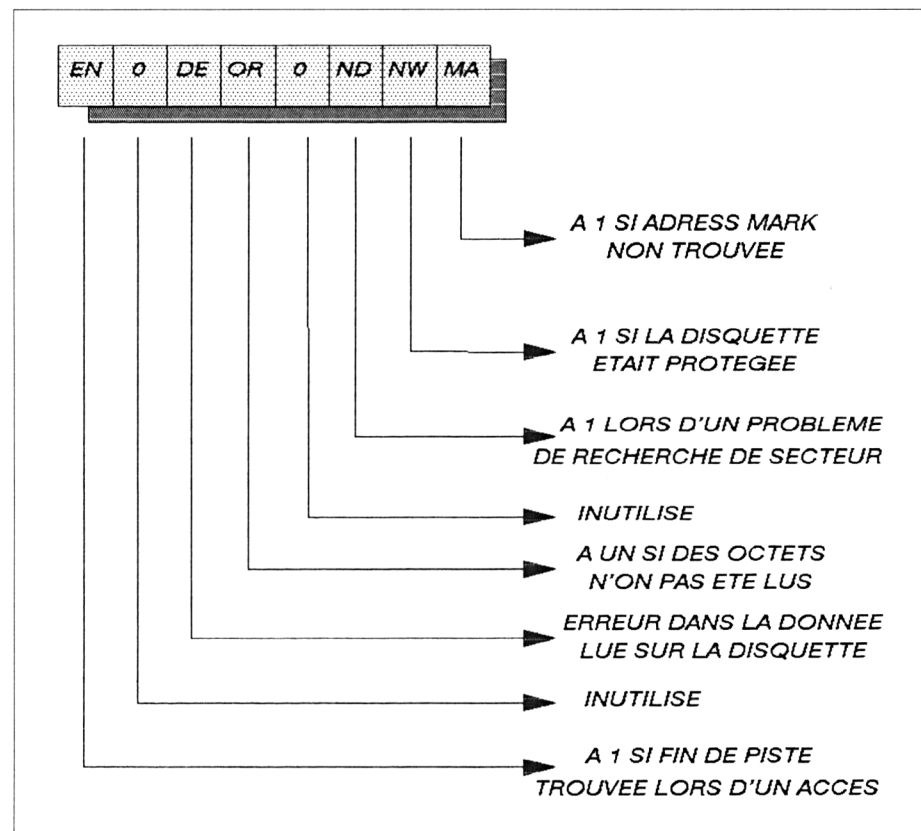


Fig. 23 : Le registre d'état 1.

Le bit **EN** (*ENd*) est placé à 1 lorsque le FDC-765 essaie de lire un secteur alors que la fin de la piste est dépassée.

Le bit **DE** (*Data Error*) indique que la donnée lue sur le support magnétique n'est pas valable.

Le bit **OR** (*Over Run*) est placé à 1 lorsque le microprocesseur n'a pas lu assez rapidement les données fournies par le FDC (lorsqu'une donnée est fournie par le FDC alors que la précédente n'a pas été lue).

Le bit **ND** (*No Data*) est positionné à 1 lorsque le FDC ne trouve pas le secteur recherché, ou y trouve une erreur.

Le bit **NW** (*Non Writable*) est mis à 1 lorsque la disquette est protégée et qu'une instruction d'écriture ou de formatage est demandée au FDC-765.

Le bit **MA** (*Missing Address mark*) est mis à 1 lorsque le FDC n'a pas trouvé l'identification d'un secteur demandé en écriture ou en lecture.

- *Le registre 2*

Ce registre complète le registre 1, et a donc la même fonction. Son contenu est représenté en figure 24.

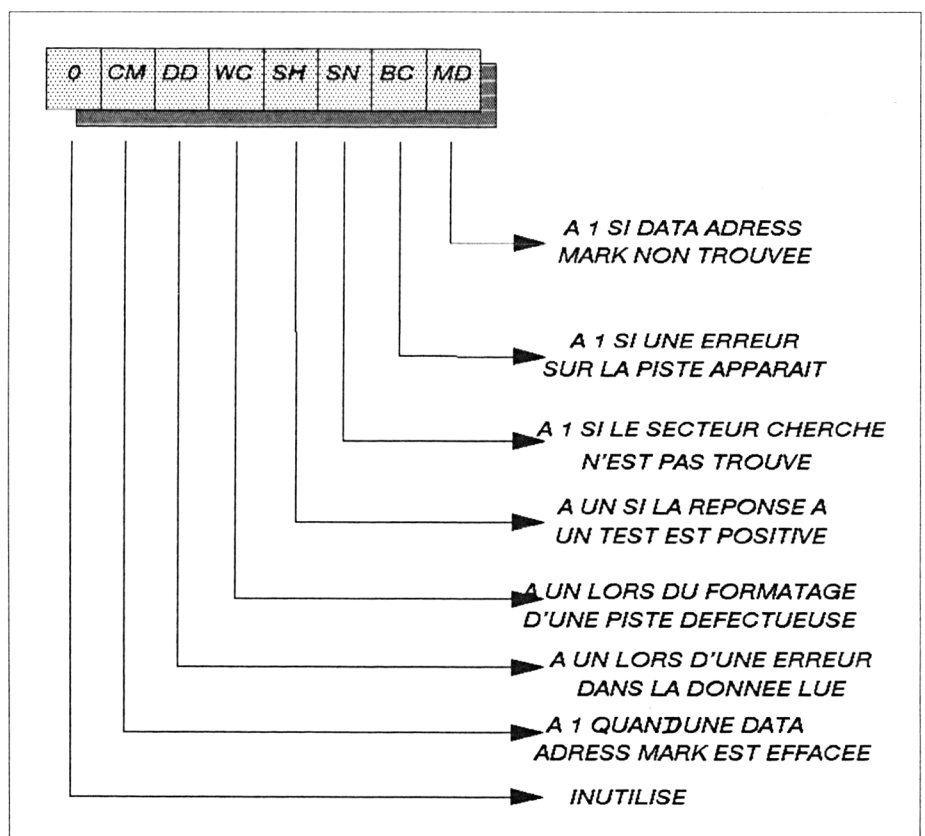


Fig. 24 : Le registre d'état 2.

Le bit **CM** (*Control Mark*) est placé à 1 lorsque le FDC rencontre une *Data Adresse Mark* effacée.

Le bit **DD** (*Data error in Data field*) indique une erreur sur une donnée lue dans un champ de données.

Le bit **WC** (*Wong Cylinder*) est mis à 1 lorsque le numéro lu sur la piste est la piste indiquée dans la phase instruction.

Le bit **SH** (*Scan equal Hit*) est à 1 lors de l'exécution d'une instruction de test d'un secteur, si les données comparées sont identiques.

Le bit **SN** (*Scan Not satisfied*) est placé à 1 dans le cas contraire au bit SH.

Le bit **BC** (*Bad Cylinder*) est mis à 1 lors d'une erreur d'identification de piste.

Le bit **MD** (*Missing adress mark in Data field*) est mis à 1 lorsque le FDC ne trouve pas de *Data Adress Mark* dans un champ de données.

- *Le registre 3*

Ce registre permet, par l'intermédiaire de l'instruction de demande d'état d'un lecteur de disquettes d'en connaître ses caractéristiques (voir figure 25).

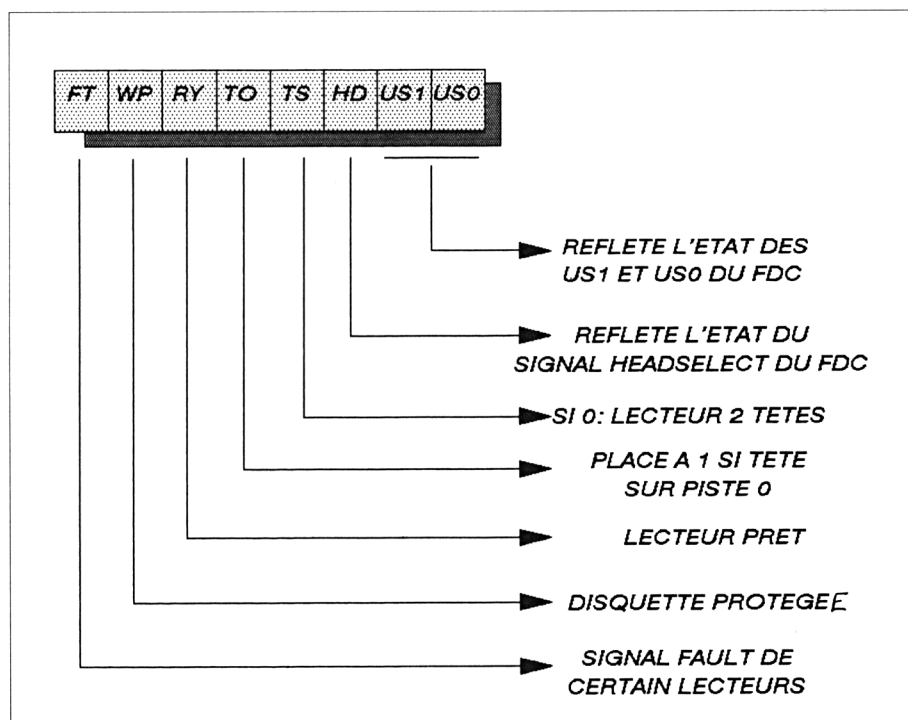


Fig. 25 : Le registre d'état 3.

Le bit **FT** (*Fault*) signale que le lecteur de disquettes indique une erreur.

Le bit **WP** (*Write Protected*) est mis à 1 lorsque la disquette dans le lecteur est protégée contre l'écriture.

Le bit **RY** (*Ready*) est mis à 1 lorsque le lecteur de disquettes est prêt à être utilisé (lorsqu'une disquette s'y trouve surtout).

Le bit **T0** (*Track 0*) est mis à 1 si la tête du lecteur de disquettes se trouve sur la piste 0.

Le bit **TS** (*Two Sides*) est mis à 0 (zéro) si le lecteur de disquettes testé possède deux têtes.

Le bit **HD** (*Head select*) signale l'état de la broche *Head Select*.

Les bits **US1** et **US0** indiquent le numéro du lecteur testé.

UTILISATION SOUS BASIC : UN CHECK DISK

Vous connaissez peut-être l'instruction **CHKDSK** des *Disk Operating System* (DOS) des ordinateurs compatibles PC, qui étudie les caractéristiques d'un lecteur de disquette ou d'un disque dur. Nous vous proposons ici un programme s'en rapprochant, qui détermine l'état des lecteurs de disquettes de votre Amstrad CPC.

Ce programme utilise le registre **3** du FDC en envoyant une demande d'état sur les deux lecteurs de disquettes possibles du CPC.

L'écran fera apparaître pour chaque lecteur :

- son numéro (A ou B) ;
- un défaut s'il y a ;
- la présence d'une disquette ;
- l'état de la disquette insérée (protégée ou non).

Le listing commenté est le suivant :

```

10 REM *****
20 REM **
30 REM ** PROGRAMME DE CHECH-UP **
40 REM ** DES LECTEURS DE **
50 REM ** DISQUETTES **
60 REM **
70 REM *****
80 REM
90 MODE 2
100 REM
110 PRINT:PRINT:PRINT
120 PRINT STRING$(10,CHR$(32));
130 PRINT STRING$(60,CHR$(45))
140 PRINT
150 PRINT STRING$(25,CHR$(32));
160 PRINT "CHEK-UP LECTEUR DE DISQUETTE"
170 PRINT
180 PRINT STRING$(10,CHR$(32));
190 PRINT STRING$(60,CHR$(45))
200 PRINT:PRINT:PRINT

```

```
210 REM
220 GOTO 510
230 REM
240 REM *** MISE EN MARCHE LECTEUR ***
250 REM
260 OUT &FA7E,1
270 FOR I% = 1 TO 2000
280 NEXT I%
290 RETURN
300 REM
310 REM *** ARRET DU LECTEUR ***
320 REM
330 OUT &FA7E,0
340 RETURN
350 REM
360 REM *** ETUDE REGISTRE D'ETAT ***
370 REM *** PRINCIPAL FDC PRET ? ***
380 REM
390 A% = INP(&FB7E) : ' LECTURE
400 IF (A% AND &80) <> &80 THEN GOTO 390
410 RETURN
420 REM
430 REM *** DONNEE DISPONIBLE ***
440 REM
450 A% = INP(&FB7E) : ' LECTURE
460 IF (A% AND &C0) <> &C0 THEN GOTO 450
470 RETURN
480 REM
490 REM *** ETUDE LECTEUR A ***
500 REM
510 LOCATE 15,24
520 PRINT "Check-up Lecteur A ..."
530 REM
540 REM *** ALLUMER LECTEUR ***
550 REM
560 GOSUB 260
570 REM
580 REM *** ETAT FDC ***
590 REM
600 GOSUB 390
610 REM
620 REM *** INSTRUCTION ***
630 REM
640 OUT &FB7F,&4
650 GOSUB 390
660 REM
670 OUT &FB7F,&2
680 GOSUB 450
690 REM
700 REM *** LECTURE RESULTAT ***
```

```
710 REM
720 RESULTAT% = INP(&FB7F)
730 REM
740 REM *** ARRET LECTEUR ***
750 REM
760 GOSUB 330
770 REM
780 LOCATE 15,24
790 PRINT STRING$(60,CHR$(32))
800 LOCATE 20,14
810 PRINT "Lecteur A"
820 REM
830 REM *** TRAITEMENT RESULTAT ***
840 REM ***          LECTEUR      A      ***
850 REM
860 LOCATE 12,16
870 PRINT "Defaut lecteur      : ";
880 IF (resultat% AND &80) = &80 THEN PR
INT "OUI" ELSE PRINT "NON"
890 LOCATE 12,18
900 PRINT "Disquette inseree : ";
910 IF (resultat% AND &20) = &20 THEN PR
INT "OUI" ELSE PRINT "NON"
920 LOCATE 12,20
930 PRINT "Disquette protegee: ";
940 IF (resultat% AND &40) = &40 THEN PR
INT "OUI" ELSE PRINT "NON"
950 IF (resultat% AND &20) <> &20 THEN L
OCATE 32,20 : PRINT "----"
960 LOCATE 12,22
970 PRINT "Lecteur 2 tetes      : ";
980 IF (resultat% AND &8) <> &8 THEN PRI
NT "OUI" ELSE PRINT "NON"
990 IF (resultat% AND &20) <> &20 THEN L
OCATE 32,22 : PRINT "----"
1000 REM
1010 REM *** ETUDE LECTEUR B ***
1020 REM
1030 LOCATE 50,24
1040 PRINT "Check-up Lecteur B ..."
1050 REM
1060 REM *** ALLUMER LECTEUR ***
1070 REM
1080 GOSUB 260
1090 REM
1100 REM *** ETAT FDC ***
1110 REM
1120 GOSUB 390
1130 REM
1140 REM *** INSTRUCTION ***
```

```
1150 REM
1160 OUT &FB7F,&4
1170 GOSUB 390
1180 REM
1190 OUT &FB7F,&1
1200 GOSUB 450
1210 REM
1220 REM *** LECTURE RESULTAT ***
1230 REM
1240 RESULTAT% = INP(&FB7F)
1250 REM
1260 REM *** ARRET LECTEUR ***
1270 REM
1280 GOSUB 330
1290 REM
1300 LOCATE 50,24
1310 PRINT STRING$(25,CHR$(32))
1320 LOCATE 60,14
1330 PRINT "Lecteur B"
1340 REM
1350 REM *** TRAITEMENT RESULTAT ***
1360 REM ***      LECTEUR      B      ***
1370 REM
1380 LOCATE 52,16
1390 PRINT "Defaut lecteur      : ";
1400 IF (resultat% AND &80) = &80 THEN P
RINT "OUI" ELSE PRINT "NON"
1410 LOCATE 52,18
1420 PRINT "Disquette inseree : ";
1430 IF (resultat% AND &20) = &20 THEN P
RINT "OUI" ELSE PRINT "NON"
1440 LOCATE 52,20
1450 PRINT "Disquette protegee: ";
1460 IF ((resultat% AND &40) = &40) THEN
PRINT "OUI" ELSE PRINT "NON"
1470 IF (resultat% AND &20) <> &20 THEN
LOCATE 72,20 : PRINT "----"
1480 LOCATE 52,22
1490 PRINT "Lecteur 2 tetes      : ";
1500 IF (resultat% AND &8) <> &8 THEN PR
INT "OUI" ELSE PRINT "NON"
1510 IF (resultat% AND &20) <> &20 THEN
LOCATE 72,22 : PRINT "----"
1520 REM
1530 LOCATE 30,24
1540 PRINT "Appuyez sur une touche"
1550 a$ = ""
1560 WHILE a$ = ""
1570     a$ = INKEY$
1580 WEND
1590 REM
1600 END
```

On remarquera qu'avant d'exécuter l'instruction, le moteur du lecteur concerné est mis en marche, et qu'une temporisation est lancée. On ne compte le lecteur que quand la phase résultat est lue complètement.

Le registre d'état principal est aussi utilisé avant chaque écriture ou lecture pour connaître l'état du FDC.

