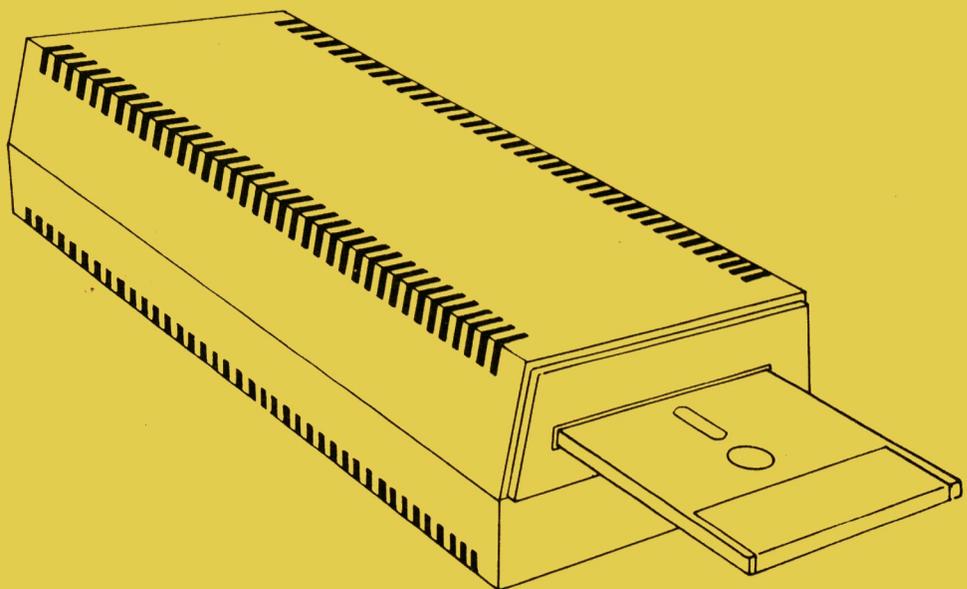


AMSTRAD

DDI 1

MANUEL  
DE  
L'UTILISATEUR







**ATTENTION**

**IMPORTANT**

NE JAMAIS ALLUMER NI  
ÉTEINDRE VOTRE LECTEUR  
DE DISQUETTE  
(DRIVE EN ANGLAIS) AVEC  
UNE DISQUETTE A L'INTÉRIEUR

TOUJOURS ALLUMER  
LE(S) DRIVE(S)  
AVANT L'ORDINATEUR



# LE DDI-1 D'AMSTRAD DISQUE DRIVE ET INTERFACE

*Recevez nos félicitations. Vous pouvez être fier, vous venez en effet de vous porter acquéreur du DDI-1 d'AMSTRAD et vous découvrirez vite la sagacité de votre choix et le bénéfice que vous tirerez de cet investissement.*

Sa rapidité va vous permettre de charger et de sauvegarder en quelques secondes des dossiers (programmes, fichiers) qui vous auraient demandé plusieurs minutes avec le lecteur de cassettes. En outre, chaque disquette est capable d'emmagasiner jusqu'à 180 K octets de données sur chaque face (360 K sur les deux faces de la disquette). Autrement dit, tous vos utilitaires sur une disquette, une dizaine de jeux sur une autre, tout votre courrier courant sur une troisième, les données d'un système indexé sur une quatrième, etc...

Vous disposez en outre, d'un accès direct à vos dossiers, et donc pas besoin de passer en revue tous les dossiers inutiles avant de trouver le bon.

C'est surtout lors de la création de programmes que l'utilisation de l'unité de disquettes se révèle avantageuse. Il est en effet rassurant de pouvoir sauvegarder ce que l'on vient d'écrire car, pour vérifier telle ou telle partie, on peut avoir envie de faire tourner le programme et là, si un 'ON BREAK GOSUB' traîne malencontreusement, l'ordinateur ne vous rendra pas la main et vous perdrez tout. Donc si vous pouvez faire ce travail en 5 secondes plutôt qu'en 5 minutes, vous avancerez, c'est évident beaucoup plus vite.

Lorsque vous programmez en Assembleur, l'utilisation de la disquette, qui vous permet d'assembler, de tester en quelques secondes plutôt que quelques minutes, s'avère un avantage formidable, et pour le programmeur chevronné et pour le débutant qui essaye de se frayer un chemin pour la première fois dans la jungle du code machine.

La présence du CP/M, c'est la porte ouverte à une vaste bibliothèque de logiciels. Le mode 80 colonnes du CPC464, c'est pour vous l'assurance de très peu de modifications pour adapter ces logiciels à votre système. Quant aux livres qui traitent du CP/M, c'est avec des nombres de trois chiffres qu'ils faut les compter !

En résumé, opérer avec l'unité de disquettes, c'est un pas en avant judicieux. Rien de comparable, vous allez voir avec le travail sur cassettes; votre approche de l'informatique a maintenant sa juste dimension.

En outre, le format choisi, 3 pouces, l'enveloppe en matière plastique dure et la protection à glissière sont pour la disquette des atouts majeurs qui éliminent les risques de fausses manœuvres possibles et lui assurent une rigidité parfaite à l'intérieur du drive.

Deux systèmes d'exploitation sont fournis :

L'AMSDOS, c'est une extension du système de gestion de fichiers sur cassettes du BASIC du CPC464. Il permet d'accéder à l'interpréteur BASIC.

Le CP/M 2.2, c'est le système d'exploitation standard, à accès direct, du microprocesseur Z80 et est utilisé par plus de 5000 programmes du commerce qui vont de logiciels de gestion à des logiciels de langages informatiques, des logiciels scientifiques, des logiciels d'analyses techniques. Le CP/M sur le CPC464 a été méticuleusement pensé de façon à ce que l'on puisse ranger sur une même disquette, avec un étiquetage automatique approprié et des dossiers écrits sous AMSDOS, et des dossiers écrits sous CP/M.

Le fameux DR (Digital Research) LOGO a fait la preuve au fil des ans de ses grandes possibilités éducatives, avec d'un côté une tortue simple et sympa à utiliser et de l'autre un traitement de données puissant. DR LOGO est le meilleur LOGO qui existe actuellement et est fourni gratuitement avec votre unité de disquettes.

## © Copyright 1984 AMSOFT, AMSTRAD SA, AMSTRAD plc

"Ni l'information contenue aux présentes, ni le produit décrit dans ce manuel, ne peuvent être modifiées ou reproduites totalement ou partiellement, en tout ou partie, sous quelque forme que ce soit, sans l'accord écrit préalable d'Amstrad".

AMSOFT et AMSTRAD acceptent volontiers vos suggestions à propos de l'ordinateur ou de ce guide.

Toute correspondance doit être adressée à

AMSTRAD FRANCE  
143, Grande Rue  
92310 Sèvres

"Toute maintenance et service après vente concernant le produit doivent être effectués obligatoirement par des revendeurs Amsoft agréés. Ni Amsoft ni Amstrad ne seront responsables, de quelque façon que ce soit, de toute perte ou dommage causé par une maintenance ou service effectué par des personnes non agréées.

Ce guide est seulement destiné à faciliter l'utilisation par le lecteur et, par conséquent, ni Amsoft ni Amstrad ne seront responsables de toute perte ou dommage quelconque qui pourrait résulter de l'utilisation de toutes informations, renseignements, erreurs ou omissions contenus dans ce guide ainsi que de toute utilisation impropre du produit.

Dr LOGO and CP/M are trade marks of Digital Research Inc.

Z80 is the trademark of Zilog Inc.

AMSDOS and CPC 464 are trademarks of AMSTRAD.

First Published 1984

Compiled by Roland Perry and Ivor Spital with acknowledgements to Locomotive Software Ltd.

Published by AMSTRAD

Typeset by AMSOFT Computer Graphics

AMSTRAD is a registered trademark of AMSTRAD Consumer Electronics plc. Unauthorised use of the trademark or word AMSTRAD is strictly forbidden.

# Table des matières

## Cours élémentaire

---

Mise en route  
A propos des disquettes  
Le chargement de logiciels, de jeux  
Introduction à l'AMSDOS et au CP/M

## Chapitre 1 Travail de préparation des disquettes

---

Duplication de la disquette système  
Une disquette système/utilitaires de travail  
Une disquette seulement pour le BASIC  
Disquette à démarrage automatique en BASIC AMSTRAD  
Disquettes et packages sous CP/M  
Configuration des logiciels  
Mise en route des progiciels sous CP/M

## Chapitre 2 Quelques éléments sur l'AMSDOS

---

La partie catalogue de la disquette  
Noms et genres de dossiers sous AMSDOS  
construction des noms de dossiers et jockers  
Exemples d'utilisation de commandes AMSDOS dans un programme  
Sauvegarde de variables et copie d'écran  
Guide de référence des commandes de l'AMSDOS  
Copie de dossiers  
Guide de référence des messages d'erreur

## Chapitre 3 Quelques éléments sur le CP/M

---

L'utilisation du CP/M  
L'organisation des pistes système  
La zone des variables système  
Les commandes d'édition  
L'amorçage de la disquette  
Les commandes du mode direct  
Les commandes transitoires  
Copie de dossiers et de disquettes  
La gestion du système  
Production de disquettes

## **Chapitre 4 Introduction au LOGO**

---

Qu'est-ce que le LOGO ?  
Les "méthodes" du LOGO  
Création de programmes et de méthodes  
Quelques mots sur le fonctionnement  
Sommaire des "primitives" du DR LOGO  
Traitement de textes  
Les opérations arithmétiques  
Les opérations logiques  
Les variables  
Les méthodes  
La correction  
L'écran en mode texte  
L'écran en mode graphique  
La tortue dessine  
Le clavier, la manette de jeu  
Le son  
Les primitives système  
Les variables système  
Les propriétés

## **Chapitre 5 Aspects techniques du firmware**

---

Les en-têtes  
Les exigences de l'implantation mémoire  
Les messages d'erreur  
Les messages de l'AMSDOS  
Les messages du BIOS  
L'organisation de la disquette

## **Appendices**

---

Appendice 1 Petit dictionnaire  
Appendice 2 Contrat d'utilisation des logiciels  
Appendice 3 Index

# LE DDI-1 d'AMSTRAD COURS ÉLÉMENTAIRE

## Eléments 1 : La mise en route

### Le branchement sur le secteur

L'unité de disquettes est alimenté par le secteur 220-240 V 50 Hz

Débranchez l'appareil quand vous ne l'utilisez pas.

N'essayez jamais d'enlever des vis ou d'ouvrir l'unité de disquette. Et respectez la notice qui se trouve sous le drive:

**PIÈCES SOUS TENSION A L'INTÉRIEUR. N'ENLEVEZ AUCUNES VIS**

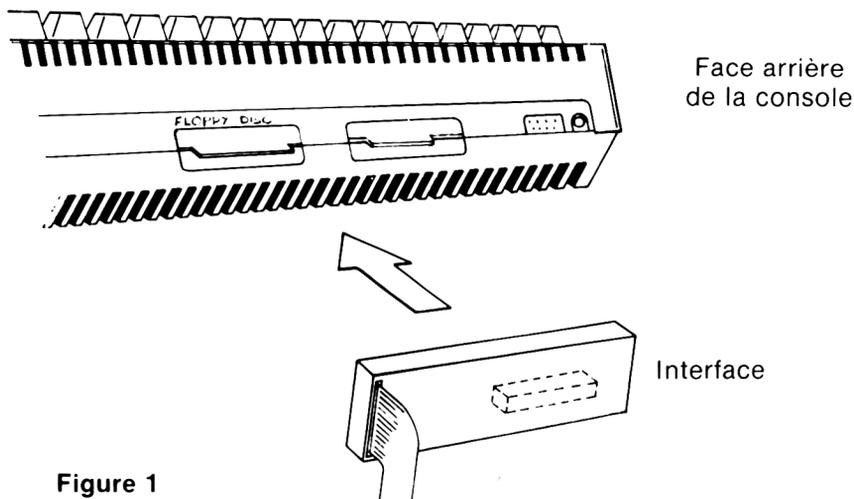
L'interrupteur ON/OFF (Allumé/éteint) du drive est situé à l'arrière.

Assurez-vous toujours qu'il n'y a pas de disquette à l'intérieur du drive quand vous éteignez ou allumez l'appareil.

Allumez toujours le drive avant d'allumer l'ordinateur.

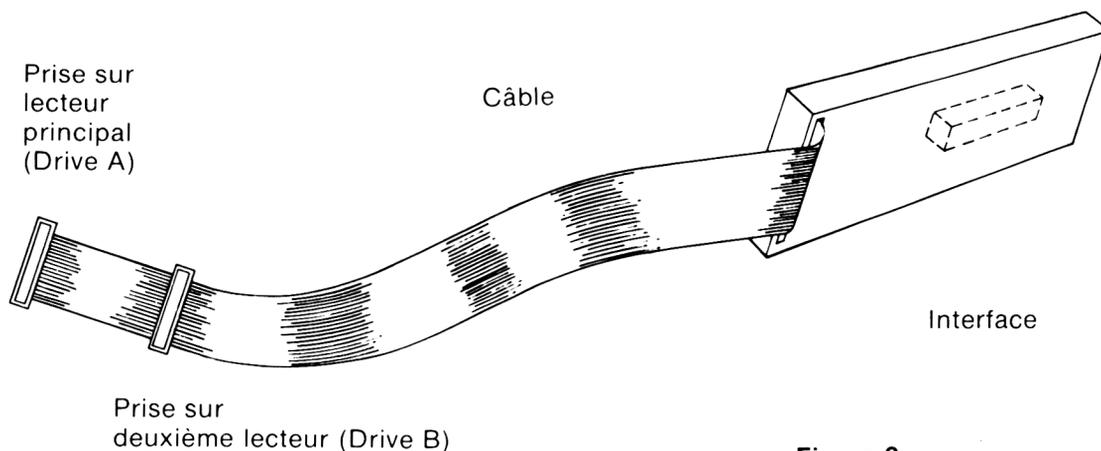
## Branchement du DDI-1 sur l'ordinateur

1. Vérifiez que l'ordinateur et que le(s) drive(s) sont éteints.
2. Enfoncez l'interface solidement dans la prise marquée FLOPPY DISC à l'arrière de l'ordinateur (voir figure 1)



3. A l'extrémité du câble plat qui sort de l'interface, vous trouverez une prise qui sert à la connection au lecteur de disquettes principal (drive A). Voir figure 2.

La connection que vous voyez un peu avant l'extrémité de ce même câble sert uniquement à rattacher le second drive (Drive B); ceci dans le cas où vous avez fait l'acquisition d'une deuxième unité de disquettes.



4. Si vous utilisez un seul drive, branchez l'unité de disquettes au micro-ordinateur comme indiqué sur la figure 3.

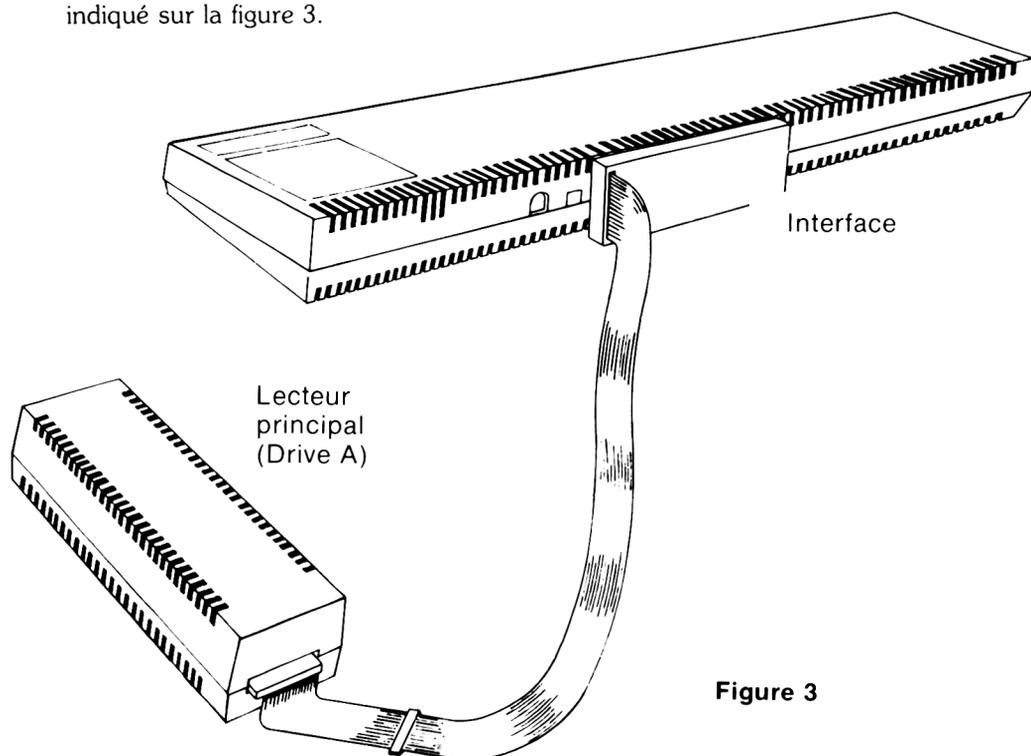


Figure 3

5. Si vous utilisez 2 drives, branchez les comme indiqué sur la figure 4.

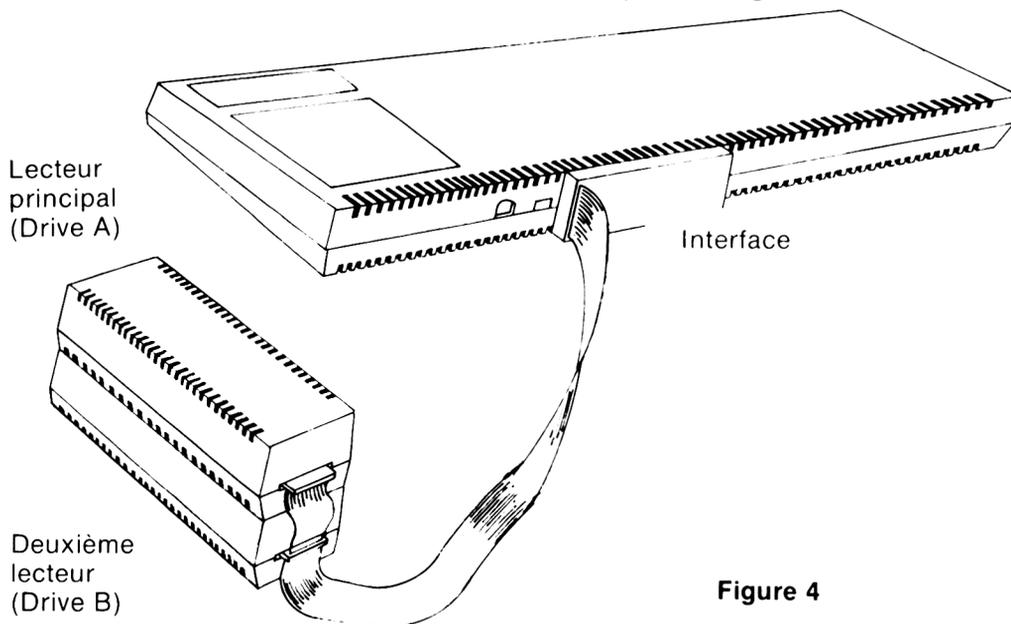
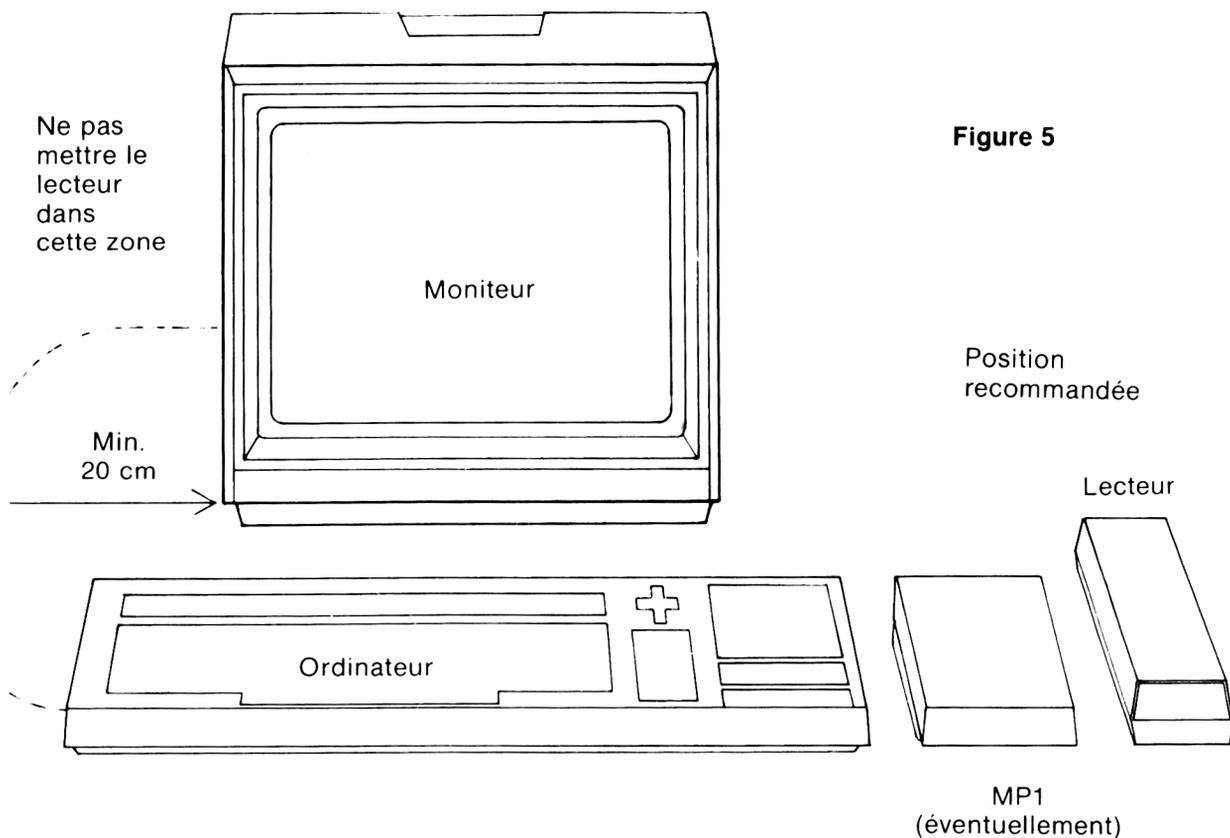


Figure 4

Allumez le ou les drives puis l'ordinateur. Vous êtes prêt maintenant à utiliser votre unité de disquettes.

## Important

1. Par mesure de sécurité, ne placez pas le(s) drive(s) à moins de 20 cm du côté gauche du moniteur comme sur la figure 5. Il est recommandé de le placer à droite de l'ordinateur et même à droite du modulateur MP-1 si vous utilisez ce dernier.



Attention à ne jamais placer votre unité de disquettes à côté d'une source d'émissions électriques, sonores, ce qui pourrait détériorer vos données.

2. Ne jamais placer non plus le drive ou les disquettes à côté d'un champ magnétique important comme par exemple celui produit par une télévision, une enceinte hi-fi, etc...
3. Le câble plat qui va de l'interface au drive ne doit pas être placé parallèle au câble secteur de l'unité de disquettes ou du moniteur, ni à gauche du moniteur.
4. Ne jamais obstruer ni recouvrir les fentes de ventilation du dessus, du dessous et des côtés du(des) drive(s).
5. Ne pas placer ni ranger les drives en lumière directe ou dans un endroit particulièrement chaud, froid, humide, poussiéreux ou sujet à d'importantes vibrations.

## Eléments 2 : Les Disquettes

Le drive AMSTRAD utilise des disquettes compactes de format 3 pouces. Nous vous conseillons fortement, pour une question de fiabilité, d'en utiliser que des CF-2 (Compact Floppy) d'AMSOFT. Cependant les disquettes 3 pouces des principaux fabricants peuvent également convenir.

Chaque face de la disquette peut être utilisée séparément. La disquette doit être insérée avec son étiquette vers l'extérieur et avec la face que vous voulez utiliser vers le haut (Voir figure 6).

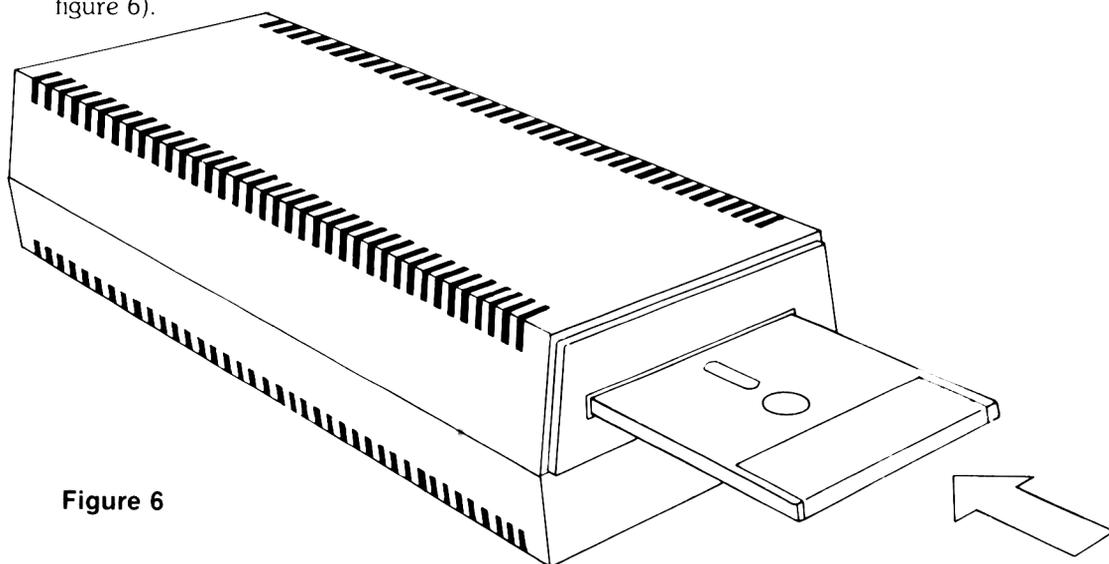


Figure 6

### Protection en écriture

Dans le coin gauche de chaque face d'une disquette, vous verrez une flèche montrant un trou fermé d'un obturateur. C'est le trou de protection en écriture qui vous empêche d'effacer ou d'écrire sur des données que vous ne voulez surtout pas perdre. Voir figure 7.

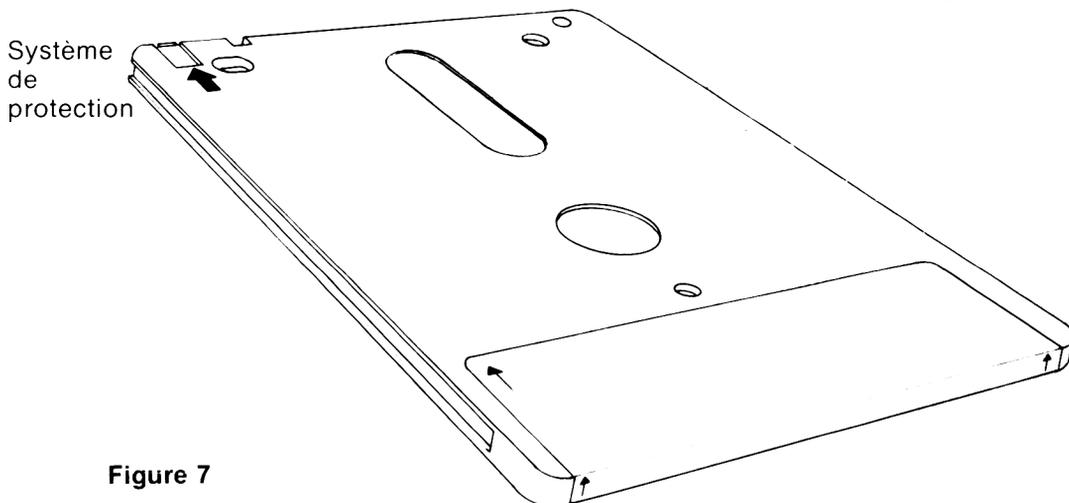


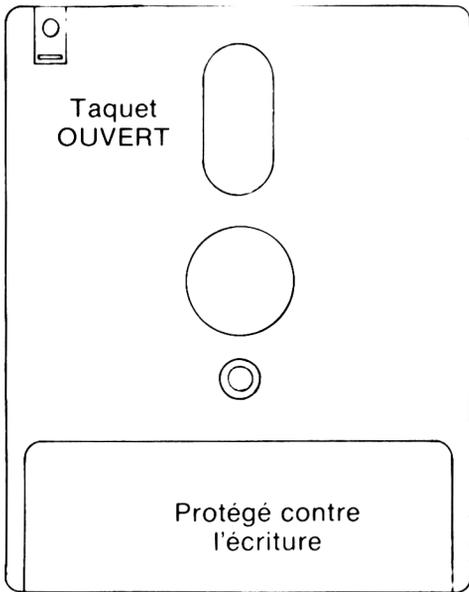
Figure 7

Quand le trou est fermé, l'ordinateur peut écrire sur la disquette. Quand le trou est ouvert, rien ne peut être écrit; vous permettant ainsi d'éviter l'effacement accidentel de programmes de grande importance.

Les fabricants de disquettes emploient des procédés différents pour l'ouverture et la fermeture des trous de protection. Pour la CF-2 d'AMSOFT, suivez les instructions ci-dessous :

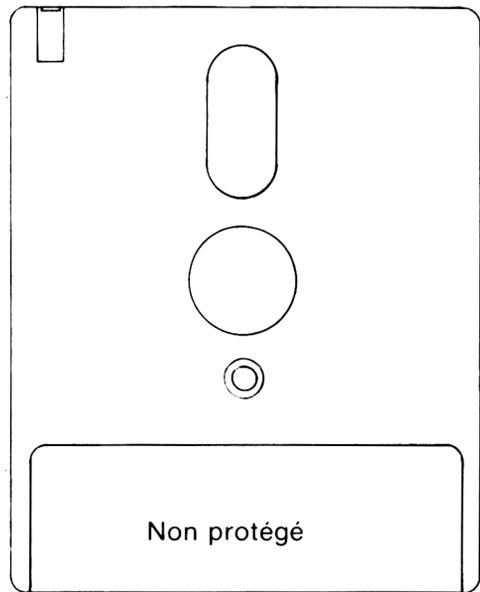
Pour ouvrir le trou de protection, retirez l'obturateur situé sur le coin gauche de la disquette et le trou s'ouvrira, figure 8(a).

Trou de protection (OUVERT)



8(a)

Taquet fermé



8(b)

Figure 8

Pour fermer le trou de protection, refermez l'obturateur, figure 8(b).

Certains autres modèles utilisent un petit levier en plastique situé dans une fente dans le coin gauche, figure 9.

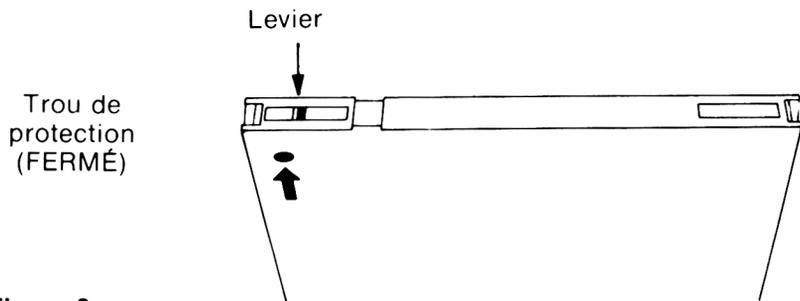


Figure 9

Pour ouvrir le trou de protection sur ce genre de disquette, il faut faire glisser le levier vers le milieu de la disquette en utilisant la pointe d'un stylo par exemple, Voir figure 10.

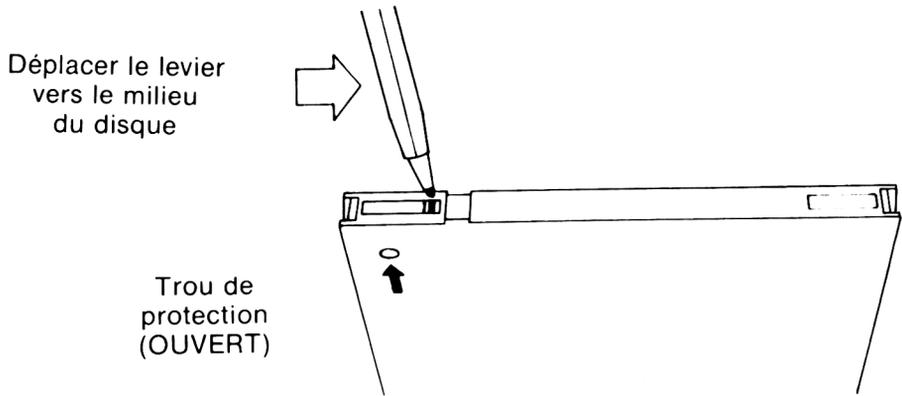


Figure 10

Quelque soit la méthode employée, il faut donc ouvrir le trou de protection si vous voulez éviter de perdre vos programmes, données en écrivant par dessus.

## IMPORTANT

Assurez-vous toujours que les trous de protection de votre disquette système CP/M sont ouverts.

### Quand la disquette est à l'intérieur du drive

Sur le devant du drive, vous pouvez voir un voyant rouge et un bouton poussoir pour l'éjection des disquettes. Voir figure 11.

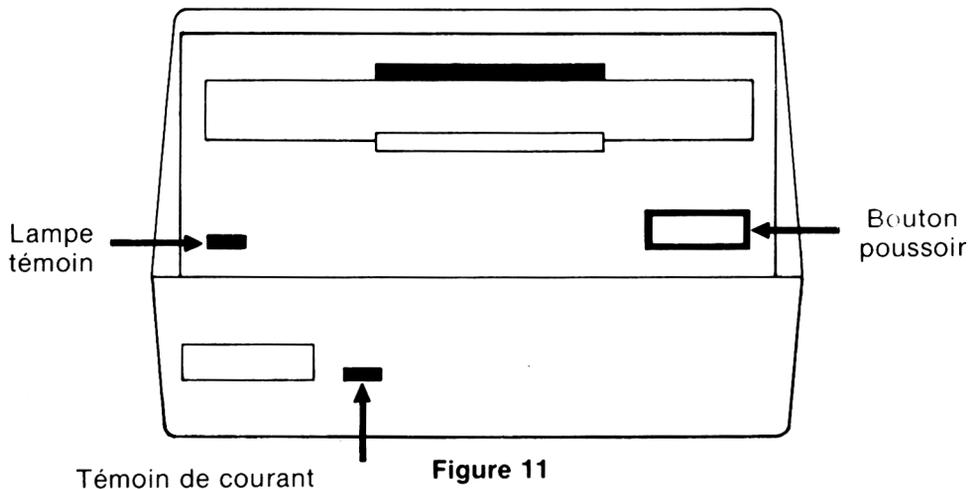


Figure 11

## **Le voyant indicateur**

Ce petit voyant rouge a 2 fonctions :

S'il y a un seul lecteur de disquette connecté, le voyant indique que des données sont lues ou écrites sur la disquette.

S'il y a deux drives de connectés, le voyant du drive B sera toujours allumé, vous permettant ainsi d'identifier les drives rapidement.

Cependant en opération (lecture ou écriture), le voyant retourne à sa fonction première, c'est-à-dire d'indiquer quel drive est utilisé.

## **Le bouton poussoir**

Il vous permet d'éjecter la disquette de son drive.

## **IMPORTANT**

1. TOUJOURS ENLEVER VOTRE DISQUETTE DU DRIVE AVANT D'ALLUMER OU D'ÉTEINDRE VOTRE SYSTÈME.
2. NE JAMAIS ESSAYER D'ÉJECTER UNE DISQUETTE PENDANT UNE OPÉRATION DE LECTURE OU D'ÉCRITURE SUR CETTE MÊME DISQUETTE, CE QUI POURRAIT ENTRAÎNER LA DESTRUCTION DES DONNÉES DE LA DISQUETTE.
3. LE DISQUE MAGNÉTIQUE EST PROTÉGÉ PAR UNE ENVELOPPE DE PLASTIQUE DUR. ATTENTION A NE PAS TOUCHER A LA DISQUETTE ELLE-MÊME.

Nous assumerons à partir de maintenant dans ce manuel que vous vous êtes familiarisé avec votre micro-ordinateur CPC-464 et avec son BASIC et que vous avez lu avec attention et compris le guide de l'utilisateur du CPC-464.

# Eléments 3 :

## Chargement de logiciels/jeux

Les logiciels, jeux disponibles sur disquette pour l'AMSTRAD CPC-464 sont chargés beaucoup plus rapidement que leurs équivalents sur cassettes.

Il est important cependant de se rappeler les principales règles du chargement et de l'initialisation des programmes.

Une commande telle que **RUN**", utilisée normalement pour le chargement d'un logiciel sur cassette, n'a aucun effet sur disquette car ici le programme doit avoir obligatoirement un nom.

Les noms de dossiers sur disquettes ont deux parties ou deux zones séparées par un point.

La première zone peut contenir jusqu'à 8 caractères et c'est habituellement le titre du jeu ou du programme. La seconde zone, optionnelle, décrit le genre du programme. Exemple : BAS ou BIN (BASic ou BINaire).

Notez que les noms de dossiers ne peuvent contenir ni espace ni point dans aucune des deux zones.

Les noms de dossiers, ainsi que les commandes AMSDOS et CP/M peuvent s'écrire soit en majuscules, soit en minuscules.

Vous trouverez plus de détails à propos des noms de dossiers sur disquette aux chapitres AMSDOS et CP/M.

Maintenant pour rentrer un programme tel que par exemple "Roland dans les grottes", tapez :

**run "rolgrot"**

Si vous avez correctement inséré votre disquette contenant ce logiciel, en quelques secondes vous serez en mesure d'utiliser le jeu.

Sinon, un message d'erreur va apparaître sur l'écran indiquant quel a été le problème :

**Drive A : disc missing** (Drive A : Pas de disquette)

**Retry, Ignore or Cancel** (Recommence, Ignore ou Annule)

Ce message signifie que vous n'avez pas placé votre disquette correctement ou que vous l'avez placée dans le drive B.

**ROLGROT. not found** (pas trouvé)

signifie que vous avez introduit une mauvaise disquette ou que vous n'avez pas rentré le nom du programme correctement.

(Suivez toujours scrupuleusement les instructions de chargement qui accompagnent chaque logiciel).

### **Bad command** (Mauvaise commande)

Signifie que votre nom de programme est incorrect, soit parce que vous avez dépassé le nombre de caractères permis, soit parce que vous avez placé malencontreusement un espace ou un point.

### **Type mismatch**

Signifie que vous avez oublié un guillemet ”

### **Syntax error** (erreur de syntaxe)

Vous avez fait une faute de frappe en écrivant le mot run

### **Drive A: read fail** (Drive A : lecture avortée)

### **Retry, Ignore or Cancel?** (Recommence, Ignore ou Annule)

L'ordinateur n'a pas compris les données inscrites sur la disquette. Vérifiez que vous avez utilisé la bonne disquette et tapez **R** (Retry = Recommence). Si ce même message apparaît une deuxième fois, c'est sûrement que la disquette a souffert ou qu'elle a mal été formatée par le CP/M Amstrad.

Enfin si le message :

### **Press play then any key** (Appuyez sur Play puis sur n'importe quelle touche)

apparaît sur l'écran, cela veut dire que le drive ou son interface ont été mal connectés à l'ordinateur ou bien que le drive n'a pas été branché au secteur ou même pas allumé.

Notez que si vous voulez utiliser un programme sur disquette après avoir utilisé le magnéto cassette, vous devez taper la commande **I disc**. Cette commande vous sera expliquée au chapitre suivant.

## **AVERTISSEMENT SUR LA MÉMOIRE UTILISÉE**

L'interface disquette DDI-1 réduit le total de la mémoire disponible de votre ordinateur de 1280 octets. Ce qui peut avoir un effet gênant sur un très long logiciel sur cassette que l'on voudrait rentrer en mémoire avec le DDI-1 branché.

Le programme démonstration "Home Runner" de la cassette Welcome qui est livrée avec le CPC-464 est un exemple de ce type de dépassement mémoire.

Si vous rencontrez ce genre de problème avec un logiciel sur cassette, enlevez-la (les disquette(s) du ou des drives, éteignez l'ordinateur puis le(s) drive(s).

Débranchez l'interface disquette de derrière l'ordinateur et allumez à nouveau votre CPC-464. Vous pouvez maintenant charger votre logiciel sur cassette normalement.

# Éléments 4 :

## Introduction à l'AMSDOS et au CP/M

Quand votre système se met en route, l'ordinateur vérifie automatiquement quels sont les périphériques qui sont connectés à l'arrière. S'il 'sent' que le ou les drives sont branchés, il dirigera toutes commandes vers le lecteur de disquettes. Ainsi les commandes :

```
load"nomprog"  
run"nomprog"  
save"nomprog"  
chain"nomprog"  
merge"nomprog"  
chain merge"nomprog"  
openin"nomprog"  
openout"nomprog"  
closein  
closeout  
cat  
eof  
input #9  
line input #9  
write #9  
list #9
```

agiront sur la disquette et non sur la cassette.

La commande **speed write** seule restera attribuée au maniement du lecteur de cassettes. On ne peut pas en effet changer la vitesse d'écriture sur une disquette.

Donc, ayant senti que le drive était connecté, l'ordinateur donne le contrôle à son système d'exploitation de disquettes, l'AMSDOS. AMSDOS est une abbréviation du mot AMStrad Disc Operating System. Il vous permet de travailler en Amstrad BASIC normal avec des commandes supplémentaires pour la gestion et l'exploitation des disquettes.

Ces commandes sont appelées commandes externes et ne sont disponibles que lorsque l'unité de disquettes est branchée. Le logiciel, qui gère ces commandes est contenu en ROM (Read Only Memory = Mémoire en lecture seule) dans l'interface et dans le drive.

Ces commandes externes sont précédées du symbole **I**. (Vous ferez cette barre **I** en appuyant sur les touches **SHIFT** et **@**).

Les commandes externes les plus utilisées sont :

```
I a  
I b  
I tape (que l'on peut subdiviser en I tape.in et I tape.out)  
I disc (que l'on peut subdiviser en I disc.in et I disc.out)
```

Les commandes **l a** et **l b** indiquent à l'ordinateur vers quel drive il doit diriger les commandes suivantes.

Si vous tapez par exemple :

**l a**

**load "nomprog"**

vous indiquez à l'ordinateur qu'il doit charger en mémoire un programme dont le nom est spécifié et qui se trouve sur une disquette dans le drive A.

Si vous n'utilisez qu'un seul drive, il peut être considéré comme le drive A et donc vous n'avez besoin de spécifier ni **l a**, ni **l b**. Avec un système à un seul drive un **l b** ferait apparaître le message d'erreur suivant :

**Drive B:disc missing** (Pas de disquette dans le drive B)

**Retry,Ignore or Cancel** (Recommence, Ignore ou Annule)

auquel vous devez répondre **C** (to cancel = Annuler)

La commande **l tape** indique à l'ordinateur qu'il doit diriger tout chargement, sauvegarde etc... vers la cassette et non vers la disquette. Si **l tape** n'est pas tapé, l'ordinateur, à chaque mise en route ou reset, dirige toutes opérations automatiquement vers la disquette.

Pour revenir sur disquette après un **l tape**, vous devez rentrer la commande

**l disc**

Par contre, vous pouvez avoir besoin de charger un programme sur cassette et sauvegarder des données sur disquette. Utilisez alors la commande

**l tape.in**

Cette commande dit à l'ordinateur : tu dois lire sur la cassette et écrire sur la disquette.

Inversement, pour lire le disque et écrire sur la cassette, vous devez rentrer la commande **l disc.in** pour annuler le **l tape.in** rentré précédemment puis **l tape.out** pour diriger toutes sorties de données vers la cassette.

On peut ainsi avoir des **l tape.in** et des **l tape.out** qui vont annuler des **l disc.in** et des **l disc.out** et vice-versa.

Vous trouverez des détails plus précis concernant les entrées et sorties plus loin dans ce manuel aux chapitres AMSDOS et CP/M.

## Le stockage de données sur disquette

Avant d'écrire des données sur une disquette vierge, il faut d'abord la formater. Le formatage peut être vu comme la construction sur la disquette d'une série d'étagères avec des compartiments dans lesquels on va stocker des informations. En d'autres mots un châssis organisé de façon à pouvoir prendre ou ranger des données.

Le formatage partage chaque face de la disquette en 360 zones distinctes, voir figure 12.

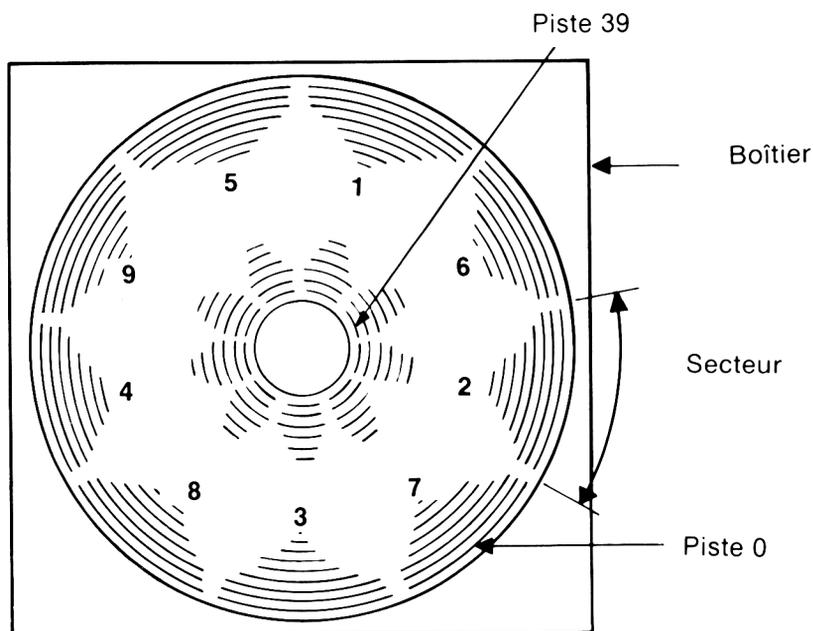


Figure 12

Il y a 40 pistes, qui vont de la piste 0, à l'extérieur, à la piste 39, la plus à l'intérieur. Chaque piste est divisée en 9 secteurs.

Chaque secteur, quelque soit la piste, peut contenir 512 octets de données donnant ainsi un total disponible sur chaque face de la disquette de 180 k octets.

### Premières étapes pour l'utilisation de la disquette CP/M

Pour préparer une disquette vierge à l'écriture et à la lecture de vos programmes, il faudra la formater en utilisant la disquette CP/M.

Allumez votre système, insérez votre disquette CP/M dans le drive.

(Si c'est un système à 2 drives, utilisez le drive A)

Puis tapez :

**l cpm**

Cette commande passe la main de l'AMSDOS au CP/M (CP/M voulant dire Control Program for Microcomputers... Programme de contrôle pour micro ordinateurs).

Après quelques secondes, vous verrez le message suivant apparaître sur l'écran :

**CP/M 2.2 - Amstrad Consumer Electronics plc.**

**A >**

C'est le message d'introduction indiquant que le système d'exploitation passe sous contrôle du CP/M.

Le petit message **A >** sur l'écran est l'équivalent du Ready quand on est en BASIC. Il indique que l'ordinateur est prêt à recevoir vos ordres.

CP/M utilise plusieurs petits messages de ce type et tous vous seront expliqués plus tard sur ce manuel au chapitre CP/M.

Une fois que l'on est sous CP/M, les ordres BASIC ne sont plus compris par l'ordinateur.

Si vous tapez par exemple la commande BASIC :

**cls**

l'ordinateur répondra en répétant le mot incompris suivi d'un point d'interrogation :

**CLS?**

montrant ainsi qu'il ne comprend pas votre commande.

Voyons maintenant quelques commandes du CP/M. Tapez :

**dir**

Vous allez voir apparaître le catalogue des commandes et utilitaires du CP/M. L'un d'eux est **format**. Tapez :

**format**

Et vous verrez apparaître sur l'écran :

**Please insert disc to be formatted into drive A  
then press any key**

Insérez la disquette à formater dans le drive A puis appuyez sur n'importe quelle touche.

Enlevez votre disquette CP/M et mettez à la place une disquette vierge puis appuyez sur n'importe quelle touche blanche ou bleue. Le formatage commence alors, de la piste 0 à la piste 39. Une fois terminé vous verrez apparaître cette question à l'écran :

**Do you want to format another disc (Y/N):**

Voulez-vous formater une autre disquette (Yes/No)

Si vous voulez formater l'autre face de votre disquette vierge ou formater une nouvelle disquette, tapez alors **Y** (pour Yes) et vous verrez réapparaître le message initial.

Cette opération de formatage se répétera jusqu'à ce que vous répondiez **N**(pour No). Dans ce cas le CP/M vous demandera :

**Please insert a CP/M system disc into drive A then press any key:**

Insérez s'il vous plaît votre disquette système CP/M dans le drive A puis appuyez sur n'importe quelle touche :

Si vous procédez ainsi, l'ordinateur vous redonne la main (ce qu'on appelle le mode direct du CP/M) et attend votre prochaine commande. Nous parlerons des autres commandes du CP/M plus loin dans ce manuel. Pour le moment vous n'avez appris qu'à formater une disquette vierge. Pour revenir à l'AMSDOS, tapez :

**amsdos**

Gardez toujours une copie de votre disquette CP/M en lieu sûr car c'est la clef de voûte de votre système. Nous vous montrerons par la suite comment faire une copie de cette précieuse disquette de façon à pouvoir la ranger ensuite soigneusement.

## ATTENTION

### LE FORMATAGE D'UNE DISQUETTE DÉJÀ ENREGISTRÉE EFFACERA SON CONTENU

Vous ne pouvez pas formater une disquette qui a son trou de protection en écriture ouvert. Si ce trou est ouvert vous verrez apparaître ce message :

**Drive A: disc is write protected  
Retry, Ignore or Cancel ?**

Drive A : la disquette est en protection écriture  
Recommence, ignore ou Annule ?

Appuyez sur **C** puis suivez les instructions de l'écran et vous quitterez ainsi le mode format.

## ...Retour à l'AMSDOS

Maintenant que nous avons formaté une disquette vierge (ou 2) nous pouvons commencer à manipuler quelques programmes BASIC, avec des allées et venues sur disquette et cassette en utilisant l'AMSDOS.

## La sauvegarde d'un programme sur disquette

Vous avez tapé un programme dans la mémoire de l'ordinateur. Vous voulez le sauvegarder sur disquette. Tapez :

**save"nomprog"**

Encore une fois, remarquez que vous devez utiliser un nom pour votre programme, un nom ne dépassant pas 8 lettres dans la première zone et 3 lettres dans la deuxième. Si la deuxième zone ou extension n'est pas spécifié, l'AMSDOS y pourvoira lui-même avec **.BAS** (Pour BASic).

Notez que les noms de dossiers ne doivent contenir ni espaces, ni ponctuation dans les deux zones et qu'un point est utilisé pour les séparer.

Vous trouverez plus loin dans ce manuel plus amples informations concernant la dénomination des dossiers aux chapitres AMSDOS et CP/M.

Comme exemple de sauvegarde sur disquette, écrivez un petit programme dans la mémoire de l'ordinateur, insérez une disquette formatée dans le drive et tapez :

### **save "exemple"**

Après quelques secondes, le message **Ready** apparaît sur l'écran, le programme est maintenant sauvegardé sur la disquette. (Dans le cas contraire vérifiez ce que dit le message d'erreur sur l'écran. La disquette peut ne pas avoir été mise dans le bon drive, ou le trou de protection en écriture peut être ouvert ou encore la commande peut avoir été mal écrite ou encore le drive pas branché!, pas allumé!!)

Les dossiers protégés, binaires ou ASCII peuvent être sauvegardés sur disquette de la même façon que sur cassette c'est-à-dire en utilisant les suffixes **.b. .p. .a.**

Les noms de dossiers sur disquette n'ont pas besoin d'être précédés d'un point d'exclamation pour la suppression des messages d'introduction. Tout point d'exclamation sera ignoré et même enlevé du nom, par l'AMSDOS.

## **Catalogue**

Comme sur les cassettes, on peut obtenir un répertoire des dossiers contenus sur les disquettes et beaucoup plus rapidement. Après la sauvegarde du programme précédent tapez :

**cat**

Vous verrez apparaître :

**Drive A: user 0**  
**Exemple. bas 1K**

Le nom du programme sera affiché avec ses 2 zones, la deuxième ayant été spécifiée ou non, ainsi que la longueur du programme au plus gros octet près.

## **Chargement d'un programme d'une disquette**

Les programmes sur disquettes peuvent être chargés dans la mémoire de l'ordinateur en utilisant la commande :

**Load "nomprog"**  
**Run**

ou si vous voulez un démarrage immédiat

**Run "nomprog"**

Cette commande est utilisée seulement avec des programmes protégés (compilés).

Si vous utilisez un système à 2 unités de disquettes, vous devez spécifier de quel drive A ou B vous voulez charger le programme : Utilisez pour cela les commandes **l a** ou **l b** avant vos instructions **save, cat, load,** ou **run.**

## Copie de programmes de disquette à disquette

En utilisant les commandes déjà vues dans ce chapitre, vous constaterez que l'on peut copier un programme d'une disquette sur une autre en le chargeant d'abord dans la mémoire de l'ordinateur puis en le sauvegardant sur l'autre disquette.

Si vous utilisez un système à 2 drives, vous pouvez mettre la disquette avec le programme à charger dans un drive, B par exemple et la disquette qui reçoit ce programme dans l'autre drive, A donc; ce qui donne :

```
l b  
load "nomprog"  
l a  
save "nomprog"
```

Pour copier tout le contenu d'une disquette sur une autre, vous trouverez sûrement plus facile et plus rapide de vous servir des utilitaires **DISCCOPY** et **COPYDISC** du CP/M plutôt que de charger puis de sauvegarder les programmes un par un.

Vous trouverez un peu plus loin dans ce chapitre un résumé rapide sur la façon d'utiliser **DISCCOPY** et **COPYDISC** mais tout sera mieux expliqué dans le chapitre CP/M.

Les utilitaires **DISCCOPY** et **COPYDISC** pourront être utilisés pour votre copie de la disquette CP/M.

## Copie de programme de cassette à disquette

Insérez votre cassette avec son programme dans le lecteur de cassette puis tapez :

```
l tape. in  
load "nomprog" ou load""
```

L'ordinateur répond alors :

```
Press play then any key:
```

(Appuyer sur la touche play puis n'importe laquelle)

Une fois le programme chargé et la cassette arrêtée, tapez

```
save" "nomprog"
```

Notez que les programmes protégés et les logiciels du commerce ne pourront pas être copiés ni sur disquette ni sur cassette.

## Copie des programmes de disquette à cassette

Insérez la disquette avec son programme dans le drive. Si vous avez auparavant chargé un programme à partir d'une cassette, tapez alors :

```
l disc in
```

puis

### **Load "nomprog"**

Une fois le programme chargé et le message **Ready** apparu, sauvegardez le programme sur cassette en tapant :

### **I tape out save "nomprog"**

l'ordinateur vous répondra :

**Press REC and PLAY then any key:** (appuyez sur les touches REC et PLAY puis sur n'importe laquelle).

Quand la cassette s'arrêtera, votre programme sur disquette aura été copié sur la cassette.

Remarquez que la commande **I tape** dirige et les entrées et les sorties de données vers la cassette (c'est-à-dire ce que font **I tape.in** et **I tape.out** réunies) et parallèlement **I disc** dirige entrées et sorties vers la disquette.

## **Copie en utilisant la disquette système CP/M**

Tout le contenu d'une disquette peut être copiée sur une autre en utilisant le CP/M. Si vous avez un système à 1 drive utilisez l'utilitaire **DISCCOPY**.

Avec un système à 2 drives vous trouverez plus aisé et plus rapide d'utiliser l'utilitaire **COPYDISC**.

## **Copie avec l'utilitaire DISCCOPY**

La "source" c'est le programme ou la disquette contenant le programme que vous voulez copier.

La "destination" c'est la disquette sur laquelle vous voulez mettre la copie.

Insérez votre disquette CP/M et tapez :

### **I cpm**

Une fois le message **A >** apparu sur l'écran tapez :

### **discopy**

L'ordinateur va vous répondre :

### **Please insert source disc into drive A then press any key**

Insérez la disquette source dans le drive A  
puis appuyez sur n'importe quelle touche

Enlevez votre disquette système du drive et insérez la disquette que vous voulez copier. Si c'est le CP/M lui-même que vous voulez copier laissez-le dans son drive.

Une fois la disquette source insérée, tapez sur n'importe quelle touche et l'ordinateur affichera sur l'écran :

**Copying started**  
**Reading track 0 to 7**

La copie a commencé  
Lecture de la piste 0 à la piste 7

Puis il vous demandera :

**Please insert destination disc into drive A**  
**then press any key**

Insérez la disquette destination s'il vous plaît puis appuyez sur n'importe quelle touche

Après quoi vous aurez donc à enlever la disquette source pour mettre à la place celle qui va recevoir la copie.

Réalisez bien que la disquette qui va recevoir la copie va être totalement effacée.

Si cette même disquette n'a pas été formatée, elle le sera automatiquement pendant la copie.

La disquette une fois insérée, tapez alors sur n'importe quelle touche et l'ordinateur vous répondra :

**Writing track 0 to 7**

Écriture à partir de la piste 0 jusqu'à la piste 7

Puis il vous invitera à mettre encore la disquette source pour la lecture cette fois des pistes 8 à 15. Vous aurez ainsi des lectures/écritures successives, 8 pistes par 8 pistes, jusqu'à la dernière, c'est-à-dire la piste 39.

Puis il vous demandera :

**Do you want to copy another disc (Y/N)**

Voulez-vous copier une autre disquette (Y/N)

Si vous avez terminé répondez **N** puis suivez les instructions sur l'écran pour sortir du mode **DISCCOPY**.

## Copie avec l'utilitaire COPYDISC

**COPYDISC** ne peut être utilisé qu'avec un système à 2 unités de disquettes. Cet utilitaire vous permet de copier tout le contenu d'une disquette sur une autre disquette, de la même façon que l'utilitaire **DISCCOPY** précédemment décrit mais cette fois, gros avantage, sans avoir à enlever et remettre les disquettes.

Si vous avez lu et compris la façon d'utiliser **DISCCOPY**, vous réussirez avec **COPYDISC** en procédant comme suit :

Insérez la disquette CP/M et tapez :

**I cpm**

Une fois le message **A>** sur l'écran tapez :

**copydisc**

Suivez les instructions qui apparaissent à l'écran, le contenu de votre disquette sera recopié intégralement sur une nouvelle disquette 8 pistes par 8 pistes jusqu'à la dernière, la piste 39. Comme **DISCCOPY**, **COPYDISC** fait lui-même le formatage. **COPYDISC** peut être utilisé pour faire la copie de la disquette CP/M, ce qui vous permettra de ranger soigneusement l'original.

## Vérification des disquettes

La disquette CP/M contient un utilitaire qui va vous permettre de comparer des disquettes entre elles pour vérifier s'il n'y a pas eu d'erreurs de copie.

Si vous avez un seul drive, rentrez le CP/M puis tapez :

**discchk**

(check: vérification)

Et suivez les instructions sur l'écran. Si l'ordinateur détecte une différence entre les 2 disquettes, le message suivant apparaîtra :

**Failed to verify destination disc correctly:  
track x sector y**

J'ai trouvé une erreur

cette erreur se trouve piste x secteur x

Cependant l'ordinateur va continuer sa vérification par groupe de 8 pistes jusqu'à la fin. S'il a trouvé au moins une erreur il y aura un avertissement **WARNING** qui apparaîtra à la fin sur l'écran avant de vous demander si vous avez une autre disquette à vérifier.

Avec 2 drives, la vérification peut être opérée beaucoup plus rapidement en utilisant l'utilitaire **CHKDISC**. Tout se passe de la même façon qu'avec **DISCCHK** mais cette fois sans la laborieuse manipulation des disquettes toutes les 8 pistes.

Entrez le CP/M puis tapez

**chkdisc**

Et suivez les instructions à l'écran.

## Pouce ! J'arrête

On peut quitter une routine CP/M quand on le désire en appuyant sur les 2 touches [CTRL] et **C**. Vous revenez ainsi au mode direct.

Faites un essai, entrez le CP/M puis tapez :

**DISCCOPY**

Quand l'ordinateur vous invite à insérer votre disquette, appuyez sur les touches [CTRL] et **C**. Le travail de copie est alors abandonné.

De plus amples informations à propos des utilitaires **DISCCOPY**, **COPYDISC**, **DISCCHK** et **CHKDISC** ainsi que **FORMAT** et les autres commandes CP/M seront données au chapitre CP/M.

*Pour conclure ce chapitre cours élémentaire, voici un bref résumé des points importants que nous avons vu :*

## **INSTALLATION**

1. Toujours brancher le cordon secteur sur une prise avec terre, en suivant les instructions contenues dans la première partie appelée "mise en marche".
2. Ne jamais brancher le lecteur de disquettes sur un autre secteur que le 220-240 volts 50 Hz.
3. Il n'y a aucun réglage possible à l'intérieur.  
N'essayez pas d'ouvrir l'appareil, adressez-vous à un service compétent.
4. Ne placez pas vos drives à moins de 20 cm sur la gauche de votre moniteur ni trop près d'une source électrique (télé par exemple).
5. Eloignez vos drives, disquettes de tout champ magnétique.
6. Eloignez le câble plat de liaison du cordon secteur.
7. N'obtenez pas les trous de ventilations de l'appareil.
8. Ne stockez pas votre matériel dans un endroit trop chaud, trop froid, trop humide ou trop poussiéreux.

## **UTILISATION**

1. Il ne doit pas y avoir de disquette à l'intérieur du drive quand on l'allume ou quand on l'éteint.
2. Allumez toujours le drive avant l'ordinateur.
3. N'utilisez pas le bouton d'éjection des disquettes en cours de lecture ou d'écriture.
4. Attention à ne pas perdre ou détériorer votre disquette système CP/M. Il vous est fortement recommandé de faire une copie de cette disquette et de penser à la protéger en écriture, trous de protection ouverts.
5. Ne jamais toucher à la disquette elle-même, la laisser à l'intérieur de son boîtier protecteur.
6. Si vous utilisez un logiciel sur cassette pendant que votre DDI-1 est allumé, attention à respecter les recommandations de la page... de ce chapitre.



# CHAPITRE 1

## Préparation de Disquettes de Travail

*Ce chapitre traite de la façon de préparer ses disquettes pour un usage quotidien et introduit quelques uns des utilitaires CP/M.*

Sujets abordés :

- \* La duplication de la disquette système.
- \* La préparation d'une disquette de travail.
- \* L'utilisation d'une disquette pour un travail en BASIC uniquement
- \* Progiciel en AMSTRAD BASIC.
- \* Progiciel sous CP/M.

Le cours élémentaire au chapitre précédent montrait comment formater une disquette vierge que l'on pourra utiliser pour des programmes en **BASIC**, des jeux, pour le CP/M et comment faire des duplications fidèles de disquettes en utilisant les utilitaires **DISCCOPY** (1 drive) ou **COPYDISC** (2 drives). Ce chapitre quant à lui parle de la façon de préparer une disquette et des programmes et utilitaires que l'on va y mettre.

### 1.1 La duplication de la Disquette Système

Il est très important de faire une copie de la disquette système fournie avec votre DDI-1 et de garder l'original en lieu sûr. Il vous en coûtera beaucoup pour la remplacer !

Rappelez-vous que cette disquette a deux faces, la face système et utilitaires et la face Dr. LOGO. Chaque disquette en fait a 2 faces que vous êtes libres d'utiliser à votre guise.

Rappelez-vous que si votre disquette est vierge, les utilitaires **DISCCOPY** et **COPYDISC** vous la formateront pendant la duplication.

## 1.2 Une disquette SYSTEME/UTILITAIRE de travail

S'il est très important de se faire une copie des 2 faces de la disquette système, vous verrez qu'il est également très intéressant de se faire des disquettes de travail contenant seulement les utilitaires que vous allez utiliser le plus. Vous laisserez ainsi plus de place pour vos programmes et si jamais vous avez besoin d'un autre utilitaire vous pourrez toujours le prendre sur votre copie de la disquette système. Mais de cette façon tous les utilitaires les plus employés seront à portée de votre clavier.

D'abord utiliser le CP/M et son utilitaire **FORMAT** pour préparer une nouvelle disquette puis l'utilitaire **FILECOPY** pour copier les utilitaires dont vous avez besoin sur cette nouvelle disquette. Seule l'expérience vous dira lesquels. Dans cet exemple nous avons choisi **AMSDOS.COM**, **FILECOPY.COM** et **DISCOPY.COM**.

Utilisez votre copie de la disquette système, entrez le CP/M et lorsque le message **A >** apparaît tapez : **FILECOPY FILECOPY.COM**. Et suivez les instructions (Le "SOURCE disc" c'est la disquette qui est dans le drive et qui contient le programme que vous allez copier. Le "DESTINATION disc" c'est la nouvelle disquette sur laquelle le programme doit être recopié). Quand c'est terminé recopiez les 2 autres utilitaires ainsi : **FILECOPY AMSDOS.COM** puis **FILECOPY DISCOPY.COM**.

Vous vous êtes fait ainsi une disquette système de travail contenant les utilisations de votre choix et vous pouvez préparer de la même façon la deuxième face de cette disquette ainsi que les deux faces d'autres disquettes simplement en utilisant les utilisations **DISCOPY** ou **COPYDISC**. Attention cependant à lire la réglementation concernant les copies à l'appendice 2 de ce livre.

## 1.3 Une disquette seulement pour le BASIC

Les piste 0 et 1 de la disquette système sont réservées au CP/M et ne peuvent être utilisées. Si vous voulez le maximum de place disponible pour des jeux, des programmes, des données en BASIC et ne JAMAIS utiliser ni le CP/M ni aucun des utilitaires avec cette disquette. Elle peut alors être formatée avec les deux pistes 0 et 1 utilisables. C'est alors une disquette de données.

Elle doit être formatée en utilisant une option de l'utilitaire **FORMAT**. Vous devez taper alors : **FORMAT D** au lieu de **FORMAT**. Pour copier des programmes sur cette disquette, vous devez utiliser **FILECOPY** (que vous aurez auparavant chargé à partir de la disquette système) ou **LOAD** suivi de **SAVE**. Avec un système à 2 drives vous pourrez cependant utiliser l'utilitaire **PIP** du CP/M. **COPYDISC** et **DISCOPY** vous formateront votre nouvelle disquette avec exactement le même formatage que l'original.

## 1.4 Progiciel en BASIC AMSTRAD

Si vous achetez un programme d'application écrit en BASIC AMSTRAD pour le CPC-464. Il se mettra en marche aussitôt le branchement. Tout ce que vous avez à faire ici est de le recopier sur la disquette qui vous convient.

### 1.4.1 Le progiciel sur disquette

Faite une copie de l'original en utilisant **COPYDISC** ou **DISCCOPY**, rangez l'original et utilisez la copie. Suivez les instructions fournies pour l'exécution du programme. Si vous avez besoin d'utilitaires, transférez-les de votre disquette système en utilisant **FILECOPY**.

### 1.4.2 Progiciel en BASIC sur votre disquette travail

Dans ce cas, recopiez les programmes du progiciel sur une autre disquette. Tapez **FILECOPY \*.\*** et suivez les instructions. En particulier répondez **N** à la question :

**Ambiguous file name: Confirm/individual files (Y/N) ?**

Nom de dossier ambigu : Confirmation à chaque dossier (Y/N) ?

L'utilitaire **FILECOPY** vous indiquera le nom des dossiers qui sont copiés. Vous pouvez maintenant lancer votre nouveau programme d'application à partir de votre disquette de travail améliorée.

## 1.5 Progiciel sous CP/M

Le système d'exploitation CP/M vous donne la possibilité de charger puis d'exécuter une immense librairie de logiciels qui ont été écrits pour des micro ordinateurs qui travaillent sous CP/M. La "logique" fondamentale de ces programmes a déjà été trouvée. La seule chose qui reste à faire pour pouvoir les utiliser sur votre CPC-464 est de les transférer sur une disquette correcte et de les renseigner sur la manière originale du CPC-464 de gérer l'écran.

Un ensemble de programmes sur une disquette ayant pour objet une application spécifique est appelé un progiciel. Ces progiciels sont habituellement construits pour être utilisés sur bon nombre d'ordinateurs différents, ayant chacun une taille d'écran particulière et leur propre système de déplacement du curseur. Quelquefois le progiciel aura été adapté à une utilisation sur le CPC-464 ou bien il y pourvoira sous la forme d'un menu pour une adaptation à des machines différentes. Dans les deux cas, suivez les instructions fournies avec le progiciel. S'il n'a pas été prévu de variante pour le CPC-464, la section 1.5.2 un peu plus loin vous indiquera quelles sont les commandes à effectuer pour que le progiciel se conforme aux normes du CPC-464. Normalement cette procédure d'adaptation se fait en rentrant au clavier des codes spécifiques. Encore une fois, suivez les instructions fournies avec le progiciel.

La disquette sur laquelle se trouve le logiciel que vous avez acheté doit être utilisable par le CPC-464. Presque chaque ordinateur utilise sa propre forme de disquette. Et bien que beaucoup aient la même taille cela ne veut pas dire qu'elles soient compatibles. Demandez à votre fournisseur la version 3" du CPC-464 d'AMSTRAD.

### 1.5.1 Copie de travail de votre progiciel

Il est prudent de conserver l'original de votre progiciel et d'en faire une copie sur une autre disquette.

Bien que les instructions ci-dessous soient prévues pour un système à un seul drive, elles sont plus simples à exécuter avec un système à deux drives. (En ignorant le second drive).

Formatez d'abord une nouvelle disquette, puis copiez tous les programmes de votre logiciel en utilisant l'utilitaire **FILECOPY** de votre disquette CP/M.

Tapez **FILECOPY \*.\*** et suivez les instructions de l'écran. En particulier répondez **N** à la question :

**Ambiguous filename: Confirm individual files (Y/N)?**

Nom de dossier ambigu : Confirmation à chaque dossier (Y/N) ?

L'utilitaire **FILECOPY** vous donnera les noms des dossiers copiés.

Quand **FILECOPY** a terminé, vous avez une copie de travail de votre progiciel. Si vous avez besoin d'utilitaires sur cette disquette, copiez-les de votre disquette système en utilisant **FILECOPY**.

## 1.5.2 La configuration d'un programme CP/M

Le CPC-464 possède une gamme étendue de codes de référence pour une adaptation de progiciel sous CP/M. La plupart de ces progiciels ont besoin d'afficher et de lire des messages n'importe où sur l'écran et souvent de comprendre les commandes de gestion du curseur.

Si votre logiciel a déjà été adapté pour le CPC-464, la partie qui suit ne vous concerne pas.

### 1.5.2.1 Adaptation de votre progiciel : les sorties (output)

La procédure de mise en service d'un progiciel consiste en un programme spécial (souvent appelé **INSTALL**) qui va vous poser un certain nombre de questions concernant les paramètres de l'écran du CPC-464. Vous prendrez vos réponses dans le tableau ci-dessous qui est extrait du manuel de référence du BASIC AMSTRAD (SOFT157) :

Valeur Hexa	Valeur Décimale	Opération effectuée
07	7	Cloche
08	8	Espace arrière
09	9	Tabulation horizontale
0A	10	Saut à la ligne
0B	11	Tabulation verticale
0C	12	Saut de page
0D	13	Retour chariot
10	16	Effacement du caractère sous le curseur
11	17	Nettoyage de la fenêtre du début à gauche jusqu'à et y compris la position du curseur.
12	18	Nettoyage de la fenêtre du curseur jusqu'au bord droit
13	19	Nettoyage du début de la fenêtre jusqu'à la position du curseur.

Valeur Hexa	Valeur Décimale	Opération effectuée
14	20	Nettoyage du curseur jusqu'à la fin de la fenêtre
18	24	Inversion vidéo
1E	30	Retour du curseur en haut à gauche
1F 'c' 'r'	31 'c' 'r'	Déplacement du curseur à la position donnée par c (colonne) 1..80 et r (rangée) 1..25.

### 1.5.2.2 Adaptation du progiciel : les entrées (Input)

Les programmes du progiciel doivent être capables d'interroger le clavier. La plupart des touches du clavier du CPC-464 ont des valeurs standard exception faite des touches de déplacement du curseur. Il est possible d'utiliser l'utilitaire **SETUP** (Voir chapitre CP/M) pour redéfinir les codes donnés par le clavier bien qu'il soit préférable autant que possible, de faire en sorte que le progiciel accepte les valeurs standard par défaut.

La colonne "Valeur WP" dans le tableau ci-dessous indique des valeurs type que l'on pourra donner aux touches correspondantes par l'intermédiaire de l'utilitaire **SETUP** pour un progiciel de traitement de texte si par exemple les codes de gestion du curseur sont déjà employés, et pour les touches de gestion du curseur, et pour une autre partie du clavier et que ce progiciel n'est capable de reconnaître qu'un seul code par opération.

La procédure de mise en service d'un progiciel consiste normalement en un certain nombre de questions sur les paramètres du clavier du CPC-464. Les réponses seront prises dans le tableau ci-dessous qui est un extrait du manuel de référence du BASIC de l'AMSTRAD (SOFT157).

Nom de la touche	Valeur Hexa	Valeur Décimale	N° de la touche pour l'utilitaire <b>SETUP</b>	Valeur WP Décimale
Curseur vers le haut	F0	240	0	5
Curseur à droite	F3	243	1	4
Curseur vers le bas	F1	241	2	24
Curseur à gauche	F2	242	8	19
Nettoyage	10	16	16	7
Return (entrée)	OD	13	18	13
Espace	EO	32	47	32
Echappement	FC	252	66	27
Tabulateur	09	9	68	9
Effacement	7F	127	79	127

WP = Word Processing (Traitement de texte)

### 1.5.3 Mise en route d'un progiciel CP/M

Normalement, vous n'avez qu'à rentrer le nom du progiciel après l'apparition du **A >**÷. Par exemple pour lancer un programme de traitement de bulletins de paye appelé **SALAIRES.COM**, tapez simplement : **SALAIRES**

## 1.5.4 Démarrage automatique des progiciels

Le système d'exploitation CP/M permet un démarrage automatique de n'importe quel programme chaque fois que l'on charge le CP/M. Ceci est réalisé par une des options de l'utilitaire **SETUP**. (Voir au chapitre CP/M pour plus de détails).

# CHAPITRE 2

## Les éléments de l'AMSDOS

*Ce chapitre décrit l'AMSDOS, toutes ses commandes disponibles, leurs fonctions et leurs utilisations. Il donne des détails sur la dénomination des dossiers et fournit enfin un répertoire des commandes. Chaque pas est illustré par un exemple.*

Les sujets abordés dans ce chapitre :

- \* Introduction à l'AMSDOS
- \* Le catalogue de la disquette
- \* La dénomination des dossiers
- \* Les entêtes des dossiers AMSDOS
- \* Les jockers
- \* Un exemple de programme avec des commandes AMSDOS
- \* Guide de référence des commandes de l'AMSDOS
- \* Manipulations de dossiers
- \* Guide de référence des messages d'erreur

### 2.1 Introduction

L'AMSDOS est une extension du BASIC AMSTRAD de votre CPC-464 avec des commandes externes supplémentaires et une redéfinition de certaines commandes BASIC. Les commandes externes additionnelles sont identifiées par le symbole | (barre) et sont automatiquement disponibles de la ROM quand l'interface DDI-1 est branchée.

Notez que chaque fois qu'une expression chaîne est requise par une commande externe, celle-ci doit être mise au préalable sous forme de variable alpha-numérique. Voir par exemple la commande **ERA**. Notez également qu'il n'est pas permis d'utiliser le symbole | (barre) après un **REM**.

L'AMSDOS permet à l'utilisateur de changer librement de disquette à condition toutefois qu'il n'y ait pas de fichiers ouverts en écriture, auquel cas un message d'erreur apparaîtrait.

## 2.2 Le catalogue de la disquette

Chaque disquette a 2 parties, la partie qui contient le catalogue et la partie qui contient les données. Le catalogue est la liste des noms de dossiers qui sont sur la disquette, avec une carte des emplacements de chacun de ces dossiers sur la disquette. L'AMSDOS et le CP/M peuvent calculer la taille de chaque dossier par inspection de cette partie catalogue. Le calcul de l'espace disponible se fait par l'addition des longueurs des dossiers existants, la différence avec le total disponible donnant la place qui reste à utiliser.

Chaque fois qu'un dossier est lu, il y a d'abord examen de la partie catalogue qui donne son emplacement sur la disquette. Quand un nouveau dossier est créé, un espace vacant lui est attribué. Une fois effacé, cet espace est libéré. Le catalogue marche par unité de 1K (un kilo octets) et peut contenir jusqu'à 64 entrées différentes. Les dossiers longs auront une entrée par chaque groupe de 16K utilisés, bien que ceci ne soit pas visible par l'utilisateur.

## 2.3 Noms et genres de dossiers sous AMSDOS

Quand vous utilisez le lecteur enregistreur de cassettes du CPC-464 les noms de dossiers peuvent comporter jusqu'à 16 lettres et ne comportent généralement aucune information sur le genre du dossier (Ex : BASIC, langage machine, etc.). Cette information est enregistrée tout au début du dossier sur ce que l'on appelle l'entête et que l'on peut visualiser par la commande **CAT**. Par contre, avec un système à disquette il est d'usage d'ajouter une extension aux noms de dossiers. Cette extension qui indique le type du dossier est conventionnelle et ne joue aucun rôle précis. Par contre, certains programmes n'accepteront un fichier que s'il a une extension d'un type particulier. Le BASIC accepte n'importe quel type de nom mais cherchera de préférence des dossiers d'un certain type s'il n'est rien spécifié de particulier. (Voir paragraphe 2.3.2).

### 2.3.1 Construction des noms de dossiers

Le nom a deux parties séparées par un point. La première partie peut être longue de 8 lettres et la seconde de 3. Ainsi, "**ECRAN.BIN**", "**BIENVENU.BAS**" et "**FORMAT.COM**" sont des noms de dossiers corrects. La deuxième partie est l'indicateur de type. Les noms de dossiers et leurs indicateurs peuvent être composés d'un mélange de lettres et de nombres mais ne peuvent contenir aucun blanc à l'intérieur. Les indicateurs de type les plus communs sont :

- .(espace) Type non spécifié. Peut être un fichier de données créé par un **OPENOUT "RESULTAT"** ou un programme BASIC sauvegardé en utilisant une commande AMSDOS du style **SAVE "PROGRAM" .A**
- .BAS Programme BASIC sauvegardé à l'aide de commandes AMSDOS du style **SAVE"PROGRAM" .P** ou **SAVE"PROGRAM.BAS" .A**
- .BIN Programme ou zone de mémoire sauvegardé par une commande AMSDOS du style **SAVE"MEMOIRE .B**, (paramètres binaires)
- .BAK L'ancienne version d'un dossier quand une nouvelle version a été faite par l'AMSDOS ou un utilitaire. Ceci pour permettre à l'utilisateur de réutiliser l'ancienne version si besoin est.
- .COM C'est une commande. Tous les utilitaires du CP/M ont cette extension.

## 2.3.2 Les entêtes de l'AMSDOS

L'AMSDOS sauvegarde automatiquement chaque dossier avec un indicateur de type et il n'est donc normalement pas nécessaire d'en spécifier un. Sauf si vous en voulez un différent de celui qui est pris par défaut. Programmes BASIC, programmes compilés, fichiers binaires sont sauvegardés sur disquette avec une entête (le même que sur une cassette) de façon à ce que la commande **LOAD** de l'AMSDOS puisse les reconnaître et agir de façon appropriée (comme sur le lecteur de cassette, la commande **LOAD** n'a besoin d'aucune indication sur le type du dossier-ceci est spécifié par l'entête). Si **LOAD** ne trouve pas d'entête, il considère que c'est un dossier écrit en codes ASCII.

Quand on lui demande de charger un dossier dont le nom ne contient pas d'indicateur de type, la commande **LOAD** de l'AMSDOS, nonobstant le contenu de l'entête, cherche d'abord pour un dossier de ce nom du type (.espace), puis s'il ne trouve rien, du type **(.BAS)** et enfin **(.BIN)**.

Ceci pour permettre à l'utilisateur d'abrégier les noms de dossiers, c'est-à-dire de ne pas spécifier le type dans beaucoup de cas.

Un fichier de données sur disquette ouvert avec la commande **OPENOUT** et donc sur lequel on écrit, n'aura pas d'entête et son contenu, venu des commandes BASIC **WRITE**, **PRINT**, **LIST**, sera des codes ASCII. La commande **OPENIN** cherchera pour un fichier de ce type si aucun indicateur n'est spécifié.

## 2.3.3 Noms de dossiers avec deux drives

Avec un système à deux drives, il peut exister des dossiers sur chacun des deux drives. L'ordinateur n'ira pas chercher sur les deux et donc il faut spécifier le drive. Vous avez à votre disposition, soit **l a**, soit **l b**, soit encore **l drive** (description complète au paragraphe 2.5.2) pour sélectionner le bon drive et utiliser ainsi un nom de dossier normal ou vous pouvez forcer le choix du drive en apposant un préfixe au nom. Le préfixe **A:** ou le préfixe **B:**  
Exemple :

```
l b  
save"prog.bas"  
l a
```

```
et  
l a  
save"b: prog.bas"
```

Ces deux séquences sauvegardent le programme sur le drive B.

### 2.3.3 Les jockers

On a souvent besoin d'opérer sur plusieurs dossiers à la fois (le catalogue, la copie, l'effacement, etc...). Quand un nom de dossier est spécifié pour telle ou telle opération, il se fait une vérification dans le catalogue pour une correspondance parfaite avec le nom cité. Il est souvent possible de demander à ce que l'opération soit effectuée sur des dossiers ayant des lettres communes, les autres lettres pouvant être remplacées par des ? (ça m'est égal). On peut abrégé une série de ? en la remplaçant par le symbole \*. Ainsi, par exemple, **FRED.\*** est un raccourci de **FRED.???** et **F\*.BAS** est le raccourci de **F???????\*.BAS**. Enfin, \*.\* veut dire n'importe quel dossier.

Exemple :

Catalogue	Convient à *.BAS	Convient à FRED?.BAS	Convient à F*.BA?
FRED1.BAS FRED2.BAS FRED3.BAK FRED3.BAS FINISH.BAS	FRED1.BAS FRED2.BAS  FRED3.BAS FINISH.BAS	FRED1.BAS FRED2.BAS  FRED3.BAS	FRED1.BAS FRED2.BAS FRED3.BAK FRED3.BAS FINISH.BAS

## 2.4 Exemples d'utilisation de commandes AMSDOS dans un programme

Pour vous aider à bien comprendre les commandes de l'AMSDOS, nous vous conseillons de bien regarder ces exemples et même de revenir sur certaines parties au moment où nous y réfererons dans la suite du chapitre. Ne faites pas fonctionner ces programmes avec l'original de votre disquette système. Utilisez une disquette travail ou une copie.

### 2.4.1 Sauvegarde de variables et recopie d'écran

NOTE : l'utilisation des indicateurs de type de dossier **.DAT** et **.SRN**. Ces indicateurs n'ont d'autre rôle que de nous rappeler ce que contient le dossier. Le dossier **PARAM.DAT** est un fichier contenant des paramètres codés en ASCII, donc sans entête, alors que **FLAGDUMP.SRN** est un dossier AMSDOS en code binaire avec une entête. Ces deux programmes sont sur la disquette système.

Le premier exemple (**EX1.BAS**) dessine le drapeau anglais et sauvegarde le contenu de l'écran sur disquette. Il s'exécute en tapant : **RUN"EX1"**. Comme nous l'avons dit plus haut, l'AMSDOS va se mettre à la recherche d'un dossier de ce nom type **.BAS**. Les particularités de l'écran que l'on va copier, c'est-à-dire le mode, la palette de couleurs et le nom du dossier contenant les données sont copiés sur un fichier. Ceci illustre l'utilisation d'un fichier de données pour la sauvegarde **WRITE** de variables de programmes (dumpfile S) et de constantes (1), que l'on pourra ensuite réutiliser dans un autre programme.

```

10 DIM colour(2)
15 REM initialisation de l'ecran
20 MODE 1: ORIGIN 0,0,0,640,0,400
30 dumpfile$="flagdump.srn"
40 FOR i=0 TO 2
50 READ colour(i)
55 REM lecture des couleurs du DATA
60 INK i,colour(i)
70 NEXT
80 OPENOUT "param.dat"
90 WRITE #9,dumpfile$,1
95 REM sauvegarde du nom de fichier et du mode
100 FOR i=0 TO 2
110 WRITE #9,colour(i)
115 REM sauvegarde des couleurs
120 NEXT i
130 CLOSEOUT
140 BORDER 0
150 FOR x=-165 TO 60 STEP 2
160 MOVE x,400:DRAWR 240,-150,1
170 MOVE x,0:DRAWR 240,150
180 NEXT x
190 FOR x=575 TO 700 STEP 2
200 MOVE x,400:DRAWR -240,-150,1
210 MOVE x,0:DRAWR -240,150
220 NEXT x
230 FOR x=-40 TO 0 STEP 2
240 MOVE x,400:DRAWR 240,-150,2
250 NEXT x
260 FOR x=0 TO 40 STEP 2
270 MOVE x,0:DRAWR 240,150
280 NEXT x
290 FOR x=640 TO 680 STEP 2
300 MOVE x,0:DRAWR -240,150
310 NEXT x
320 FOR X=600 TO 640 STEP 2
330 MOVE X,400:DRAWR -240,-150
340 NEXT x
350 ORIGIN 0,0,260,372,0,400:CLG 1
360 ORIGIN 0,0,0,640,150,250:CLG 1
370 ORIGIN 0,0,284,348,0,400:CLG 2
380 ORIGIN 0,0,0,640,169,231:CLG 2
390 SAVE dumpfile$,b,&C000,&4000
400 DATA 2,26,6

```

Le deuxième exemple **EX2.BAS** est le programme type de restitution d'une copie d'écran en utilisant le fichier des paramètres pour guider cette action. Notez la façon de sortir (**INPUT**) les variables de leur fichier avec la fonction **EOF** (End Of File = Fin de fichier) qui permet une variation automatique des longueurs de fichiers. Il est important d'avoir sauvegardé les paramètres de l'écran, le résultat aurait pu être complètement brouillé. Il fallait veiller également à ce qu'il n'y ait pas de scrolling (Défilement continu de l'image sur l'écran).

```
10 DIM colour(15)
15 REM Place reservée pour 16 couleurs
20 OPENIN "Param.dat"
30 INPUT #9,filename$,screenmode
40 i=0
50 WHILE NOT EOF
60 INPUT #9,colour(i)
70 INK i,colour(i)
80 i=i+1
90 WEND
100 CLOSEIN
110 MODE screenmode:BORDER 0
120 LOAD filename$
```

## 2.4.2 Récupération d'espaces mémoire

Si un programme ayant des entrées/sorties à effectuer sur un fichier de données possède un certain nombre de variables chaîne et particulièrement des arrangements de chaîne, l'ouverture des fichiers sera accélérée si l'ordinateur n'a pas à effectuer à ce moment là de nettoyage de mémoire. Le nettoyage de mémoire est un procédé du BASIC pour vider les emplacements mémoire réservés aux chaînes de façon à garder une zone tampon dynamique pour les données des fichiers.

Si un programme utilise des variables chaîne et des arrangements de chaîne incorporez à votre programme les trois lignes ci-dessous; une zone tampon sera ainsi allouée en permanence, ce qui annulera le nettoyage de mémoire au moment de l'ouverture des fichiers. Notez que les lignes ci-dessous doivent être intégrées après chaque commande **SYMBOL AFTER**.

```
OPENOUT "DUMMY"
MEMORY HIMEM-1
CLOSEOUT
```

## 2.5 Guide de référence des commandes de l'AMSDOS

Référez-vous au chapitre 8 de votre guide de l'utilisateur du CPC-464 si vous avez un doute sur la notation utilisée ou sur l'effet de telle commande sur le lecteur de cassettes.

### 2.5.1 Sommaire des commandes BASIC

Les commandes suivantes sont dirigées automatiquement vers l'unité de disquette plutôt que vers le lecteur de cassettes. Exception faite de la commande **CAT**, dont l'action est singulièrement différente, les effets de ces commandes sont identiques à leur version sur cassette décrite dans le manuel de l'utilisateur du CPC-464.

Commandes de lecture de dossiers

**LOAD, RUN, CHAIN, MERGE, CHAIN MERGE,  
OPENIN, EOF, INPUT #9, LINE INPUT #9, CLOSEIN**

Commandes d'écriture sur dossiers

**SAVE  
OPENOUT, PRINT #9, WRITE #9, LIST #9, CLOSEOUT**

Commande pour un catalogue

**CAT**

Le catalogue de la disquette est trié par ordre alphanumérique et disposé en colonnes avec la longueur de chaque dossier. Il est affiché également la quantité d'espace disponible.

### 2.5.2 Sommaire des commandes externes de l'AMSDOS

Les commandes suivantes sont contenues en ROM à l'intérieur de l'interface de l'unité de disquettes et sont disponibles dès que le DDI-1 est branché et allumé.

**I A**

**I A**

COMMANDE : Dirige les entrées/sorties vers le drive A. Equivalent à **I DRIVE** avec **A** pour paramètre.

**I B**

**I B**

COMMANDE : Dirige les entrées/sorties vers le drive B. Equivalent à **I DRIVE** avec **B** pour paramètre.

## I CPM

### I CPM

COMMANDE : Charge et initialise le système d'exploitation contenu dans la disquette système. Le système d'exploitation fourni avec le DDI-1 est le CP/M 2.2. Cette commande échouera si le drive A ne contient pas la disquette système.

## I DIR

I DIR, chaîne alphanumérique)

f\$ = "\*.BAS"

I DIR.@f\$

COMMANDE : Affiche le catalogue de la disquette (façon CP/M) et l'espace disponible. Si la chaîne est omise, le CP/M supposera que c'est le jockey \*.\*

## I DISC

### I DISC

COMMANDE : L'équivalent des deux commandes I DISC.IN et I DISC.OUT Réunies.

## I DISC.IN

### I DISC.IN

COMMANDE : Force la lecture des données de la disquette.

## I DISC.OUT

### I DISC.OUT

COMMANDE : Force l'écriture sur la disquette.

## I DRIVE

I DRIVE, (chaîne alphanumérique)

a\$ = "A"

I DRIVE@a\$

COMMANDE : Pour désigner un des drives. La commande avortera si l'AMSDOS est incapable de lire la disquette dans le drive désigné.

## **I ERA**

**I ERA**, (chaîne alphanumérique)

**a\$** = "FRED.BAK"

**I ERA,@a\$**

COMMANDE : Tous les fichiers identiques à l'expression chaîne et qui n'ont pas de protection seront effacés. Les jockers sont permis.

## **I REN**

**I REN**, (chaîne alphan.), (chaîne alphan.)

**O \$** = "ANCIENOM.BAS"

**N\$** = "NOUVEAU.BAS"

**I REN.@N\$.@O\$**

COMMANDE : Pour changer le nom d'un dossier. Le nouveau nom doit être différent de ceux qui existent déjà sur la disquette.

## **I TAPE**

**I TAPE**

COMMANDE : L'équivalent des deux commandes **I TAPE.IN** et **I TAPE.OUT** réunies.

## **I TAPE.IN**

**I TAPE.IN**

COMMANDE : Force la lecture des données de la cassette.

## **I TAPE.OUT**

**I TAPE.OUT**

COMMANDE : Force l'écriture sur la cassette.

## **I USER**

**I USER**, (nombre entier)

**I USER,3**

COMMANDE : Seulement pour spécialistes. Consulter le manuel de référence du CP/M.

## 2.6 Copies de dossiers

Nous avons déjà vu le programme CP/M **FILECOPY** et son utilisation pour la copie de la disquette système ou d'un progiciel. Ce paragraphe dit comment copier n'importe quel type de dossier de disquette à disquette et de disquette à cassette (et vice-versa)

### 2.6.1 Dossiers AMSDOS a entête

Il est possible de copier ces dossiers sous CP/M en utilisant **PIP** ou **FILECOPY** (voir chapitre CP/M). N'importe quel dossier créé sous AMSDOS comporte une zone entête (voir paragraphe 3.2.2) et peut être copié intégralement, de disquette à disquette, de disquette à cassette et de cassette à disquette mais en général le contenu de ces dossiers ne pourra pas être compris par le CP/M.

### 2.6.2 Dossiers ASCII

Les dossiers créés sous AMSDOS sans zone entête sont généralement écrits en codes ASCII, et sont compris et par l'AMSDOS et par le CP/M. Ils pourront donc s'échanger librement entre l'AMSDOS et le CP/M.

### 2.6.3 Dossiers en lecture seulement

Il est possible, sous CP/M, de faire en sorte que le dossier ne puisse être que lu, qu'il soit ou non sur le catalogue. De tels attributs sont établis et enlevés par le CP/M mais ils sont respectés par l'AMSDOS. Pour plus de détails, voir au chapitre CP/M le fonctionnement de l'utilitaire STAT.

### 2.6.4 Les procédures de copie de dossiers

Les tableaux qui suivent montrent la façon de copier n'importe quel dossier entre disquettes et cassettes. Ils sont fait pour un système à un seul drive. Il est absolument impossible de copier un programme BASIC protégé ainsi qu'un dossier binaire (comme par exemple un jeu écrit en langage machine) à moins que les adresses de chargement ne soient connues. Vous trouverez plus de détails sur **FILECOPY**, **CLOAD** et **CSAVE** au chapitre CP/M.

La copie d'un dossier d'une disquette sur une autre avec un système à deux drives est simplifiée par l'utilisation de l'utilitaire **PIP** du CP/M (Voir chapitre CP/M).

COPIE SUR :	COPIE DE :		
	<i>AMSTRAD BASIC on tape *</i>	<i>ASCII data on tape *</i>	<i>Binary on tape *</i>
<i>AMSTRAD BASIC on tape *</i>	ITAPE LOAD "FILE" ·change tapes· SAVE "FILE" IDISC		
<i>Binary on tape *</i>			H=HIMEM ITAPE MEMORY ·s·-1 LOAD "FILE" ·change tapes· SAVE "FILE" ,B,·s·,·l·[,·r·] IDISC MEMORY H ·note 2·
<i>ASCII on tape *</i>	ITAPE LOAD "FILE" ·change tapes· SAVE "FILE",A IDISC	ICPM CLOAD "FILE", TEMP ·change tapes· CSAVE TEMP ,"FILE" ERA TEMP AMSDOS ·note 1·	
<i>AMSTRAD BASIC on disc *</i>	ITAPE LOAD "FILE" IDISC SAVE "FILE"		
<i>ASCII on disc</i>	ITAPE LOAD "FILE" IDISC SAVE "FILE",A	ICPM CLOAD "FILE" AMSDOS	
<i>Binary on disc *</i>			H=HIMEM ITAPE MEMORY ·s·-1 LOAD "FILE" IDISC SAVE "FILE" ,B,·s·,·l·[,·r·] MEMORY H ·note 2·

\* Dossier avec entête

note 1 : nécessite un emplacement disponible sur disquette pour le programme de temporisation "**TEMP**"

note 2 : (s) est l'adresse du début du dossier, (l) est la longueur et (r) l'adresse de lancement optionnelle.

COPIE SUR :	COPIE DE :			
	<i>AMSTRAD BASIC on disc *</i>	<i>ASCII data on disc</i>	<i>AMSDOS Binary on disc *</i>	<i>All other on disc</i>
<i>AMSTRAD BASIC on tape *</i>	LOAD "FILE" ITAPE SAVE "FILE" IDISC			
<i>Binary on tape *</i>			H=HIMEM MEMORY (s,-1 LOAD "FILE" ITAPE SAVE "FILE", B,(s),(l),(r) IDISC MEMORY H (note 2.	
<i>ASCII on tape *</i>	LOAD "FILE" ITAPE SAVE "FILE",A IDISC ICPM CSAVE FILE AMSDOS	ICPM CSAVE FILE AMSDOS		ICPM CSAVE FILE AMSDOS (note 3.
<i>AMSTRAD BASIC on disc *</i>	LOAD "FILE" (change discs) SAVE "FILE" - or - ICPM FILECOPY FILE (follow instructions) AMSDOS			
<i>ASCII on disc</i>	LOAD "FILE" SAVE "FILE",A	ICPM FILECOPY FILE (follow instructions) AMSDOS		
<i>AMSDOS Binary on disc *</i>			ICPM FILECOPY FILE (follow instructions) AMSDOS	
<i>All other on disc</i>			ICPM FILECOPY FILE (follow instructions) AMSDOS	

\* Dossier avec entête

note 2 : (s) adresse de début de dossier, (l) longueur, (r) option, adresse de lancement.  
 note 3 : La copie n'est pas utilisable directement par le BASIC. Cependant cette option est utile car rentable (peu de déplacements et peu d'intermédiaires). Le dossier peut être ensuite récupéré par la disquette au moyen de **CLOAD "FICHIER"**

## 2.7 Guide de référence des messages d'erreur de l'AMSDOS

Quand l'AMSDOS pour une raison quelconque ne peut exécuter une commande, il affiche sur l'écran un message d'erreur. Si c'est un problème mécanique, le message est suivi de la question :

### **Retry, Ignore or Cancel ?**

On recommence, on ignore, on annule ?

**R** va faire réexécuter la commande, sûrement parce que l'utilisateur a entre temps palié au problème.

**I** L'ordinateur continue son travail comme s'il ne s'était rien passé. Ce qui peut mener parfois à des résultats inattendus et à d'autres problèmes.

**C** L'opération est annulée. Peut entraîner un autre message d'erreur. Voir le chapitre 5 pour plus de détails.

### **Unknown command** (commande inconnue)

La commande n'a pas été écrite correctement ou l'interface n'est pas branchée, pas allumée.

### **Bad command** (mauvaise commande)

La commande ne peut être exécutée. Erreur de syntaxe ou problème mécanique.

### **"NOMDEDOS" already exists** (existe déjà)

L'utilisateur essaye de renommer un dossier avec un nom déjà utilisé sur la disquette.

### **"NOMDEDOS" not found** (je n'ai pas trouvé de dossier portant ce nom)

Le dossier n'existe pas.

### **Drive (DRIVE): directory full** (plus de place sur le catalogue pour un nouveau venu)

### **Drive (DRIVE): disc full** (plus de place sur la disquette)

### **Drive (DRIVE) : disc changed, closing NOMDEDOS** (la disquette a été enlevée, fermeture du dossier NOMDEDOS)

### **NOMDEDOS is read only** (en lecture seulement)

Le dossier ne peut pas être traité car c'est un dossier en lecture seulement. Cet attribut peut être mis ou enlevé uniquement sous CP/M.

**Drive (DRIVE): disc missing** (pas de disquette)

Pas de disquette dans le lecteur, ou bien encore, elle n'a pas été placée ou refermée correctement. Il est alors recommandé de l'éjecter hors du drive et de la remettre à nouveau puis de taper **R**

**Drive (DRIVE): disc is write protected** (disquette protégée en écriture)

On a essayé d'écrire sur une disquette qui avait son trou de protection en écriture ouvert. Pour l'utiliser, il faut la sortir, fermer le trou de protection, la réinsérer et taper **R**.

**Drive (DRIVE): read fail** (lecture avortée)

Problème mécanique. Il est recommandé d'éjecter la disquette puis de la remettre en place. Tapez **R**.

**Drive (DRIVE): write fail** (écriture avortée)

Comme précédemment.

**Failed to load CP/M** (chargement du CP/M avorté)

Erreur rencontrée pendant la lecture de la disquette système lors de l'exécution de la commande **I cpm**. Ou bien votre disquette ne contient pas de CP/M.

# Chapitre 3 : Les éléments du CP/M

## Utilisation du CP/M

Sujets abordés dans ce chapitre

- \* chargement du CP/M
- \* la zone des variables système
- \* le mode direct
- \* les programmes transitoires
- \* gestion des périphériques

Le CP/M est un système d'exploitation de disquettes. C'est un programme spécial qui vous permet d'utiliser votre CPC-464 au maximum de sa puissance. Le CP/M est disponible sur tant d'ordinateur différents qu'il existe des milliers de logiciels d'application, ainsi qu'une richesse énorme d'expérience et de connaissances sur lesquelles vous pourrez vous appuyer.

Vous trouverez une information complète sur le CP/M ainsi que sur la façon d'écrire ses propres programmes, avec également des renseignements sur les particularités du CP/M du CPC-464 dans le guide du CP/M (Soft159A)

### 3.1 Introduction

Le système d'exploitation CP/M, c'est l'interface avec le mécanisme du drive, une façon pour vous de communiquer avec l'ordinateur et de manipuler des fichiers et des périphériques.

L'intermédiaire principal à votre disposition, c'est le mode direct. On l'identifie par la présence des indicateurs d'attente **A>** ou **B>** à l'écran. Certaines commandes, dites résidentes sont alors disponibles immédiatement mais la plupart des fonctions sont obtenues en chargeant en mémoire des programmes dits transitoires. Transitoires car ils ne transitent par l'ordinateur qu'au moment où on les utilise, contrairement aux commandes résidentes.

À côté des messages d'erreur standard du CP/M, le système de gestion de disquettes AMSTRAD génère ses propres messages. Référez-vous au paragraphe 2.7 de l'AMSDOS.

### 3.2 L'organisation des pistes système

La plus grosse partie du CP/M réside sur les deux premières pistes de la disquette. L'interface disquette DDI-1 charge le CP/M de ces deux pistes dans la mémoire de l'ordinateur en deux fois.

D'abord, la commande **I CPM** de l'AMSDOS charge le premier secteur de la piste 0 dans le CPC-464.

Sur la disquette système, ce secteur s'avère contenir un programme qui gère le chargement du reste des pistes système en mémoire.

Une partie du CP/M reste en permanence en ROM à l'intérieur de l'interface disquette, le reste se chargeant dans la mémoire RAM de l'ordinateur. Dans la configuration standard, 39,75 K octets de RAM restent disponibles pour les programmes transitoires dont 2 K sont utilisés par le module de traitement des commandes qui se charge automatiquement chaque fois que l'on repasse en mode direct.

### 3.3 La zone des variables système

Lors du premier chargement du CP/M se chargent également divers paramètres système. Ces paramètres comprennent entre autre, le message d'introduction, les codes claviers spécifiques etc. The **SETUP** permet d'adapter ce secteur à vos besoins.

### 3.4 Les commandes d'édition

L'univers CP/M, c'est aussi un ensemble de codes clavier spéciaux pour contrôler le déroulement des programmes. Ces codes remplacent le **[ESC]** du BASIC, bien que certains progiciels peuvent redonner à **[ESC]** sa fonction initiale.

**[CTRL] S**Suspend momentanément la présentation des données sur l'écran. On frappe n'importe quelle touche pour continuer.

**[CTRL] C**Au début d'une ligne de commande, redonne la main au mode direct du CP/M. Plusieurs utilitaires et programmes d'application y verront une demande de sortie du programme.

**[CTRL] P**Interrupteur marche/arrêt de votre périphérique d'affichage (généralement l'imprimante)

**[CTRL] Z**Fin de texte

### 3.5 Amorçage de la disquette

A moins du caractère particulier d'un utilitaire du CP/M (comme **FILECOPY** par exemple) le CP/M ne vous permet pas d'écrire sur une disquette sans qu'elle ait été d'abord amorcée. De plus le type de formatage (Système, Data, IBM) n'est pris en compte que lors de l'amorçage. Pour le drive A, c'est le cas chaque fois que, sous CP/M, on repasse en mode direct ou quand on fait un **[CTRL] C** à l'apparition des messages **A >** ou **B >** . Pour le drive B, c'est le cas la première fois qu'on y a accès, une fois que le drive A est amorcé.

Essayez d'écrire sur une disquette qui n'a pas été amorcée, le message d'erreur **Bdos Err on (DRIVE): R/O.** (Erreur BDOS sur le drive spécifié). Appuyez sur n'importe quelle touche pour continuer. Si la disquette changée avait aussi un format différent, un message d'erreur en lecture ou écriture apparaîtra. Tapez **C** pour poursuivre.

## 3.6 Les commandes du mode direct

Au nombre de 5, elles peuvent être tapées à l'apparition du message **A >** ou **B >**. La première d'entre elles, **SAVE** est réservée aux spécialistes.

Les messages d'erreur du CP/M sont affichés de manière économique et répètent généralement la commande ou le nom du dossier erroné suivi d'un point d'interrogation. Retapez votre commande en corrigeant votre erreur.

### 3.6.1 Les noms de dossiers

Beaucoup de commandes ont pour paramètres des noms de dossiers qui quand c'est spécifié, peuvent utiliser des jockers (voir paragraphe 2.3.1 et 2.3.3 de l'AMSDOS). Tous les noms de dossiers sont forcés à leur nombre de caractères maximum.

Les commandes du mode direct n'exigent pas de guillemets de chaque côté d'un nom de dossier. Seuls, **CLOAD** et **CSAVE** (voir paragraphe 3.7.2.3) exigent ces guillemets.

Rappelez vous que les noms dossiers peuvent avoir un préfixe **A:** ou **B:** pour obliger le CP/M à utiliser le drive mentionné, dans le cas d'un système à deux drives.

### 3.6.2 L'aiguillage des drives

Avec un système à deux lecteurs de disquettes, il est possible d'aiguiller le système de sélection des drives vers **A** ou **B** en tapant **A:** ou **B:** après les messages **B >** ou **A >**

### 3.6.3 La commande DIR

**DIR** affiche le catalogue de la disquette. Il n'y a aucun ordre particulier sinon leur ordre d'entrée sur le répertoire de la disquette. Les jockers sont permis.

<b>DIR</b>	fera la liste de tous les dossiers
<b>DIR B:</b>	fera la liste de tous les dossiers présents sur B
<b>DIR *.BAS</b>	tous les dossiers de type <b>.BAS</b>
<b>DIR B:*.BAS</b>	tous les dossiers de type <b>.BAS</b> du drive <b>B</b>
<b>DIR PIP.COM</b>	Il y aura seulement <b>PIP.COM</b> sur la liste, s'il existe.

### 3.6.4 La commande ERA

**ERA** vient de l'anglais **ER**ase qui veut dire effacer. Cette commande sert donc à effacer un ou plusieurs dossiers du catalogue. Seule l'entrée sur le répertoire est effacée. Les données resteront à leur place jusqu'à ce qu'un autre dossier ne les recouvre. Elles ne pourront jamais lus être récupérées. Les jockers sont permis. Si le nom de dossier spécifié est **\*.\***, la commande **ERA** demandera confirmation pour savoir si vous voulez bien effectivement tout effacer. **ERA** ne mentionne pas le nom des dossiers qu'elle efface. Si un des dossiers que vous voulez effacer est de type 'lecture seulement', (voir **STAT**) la commande avortera.

<b>ERA PIP.COM</b>	effacera le programme <b>PIP.COM</b>
<b>ERA B:PIP.COM</b>	effacera le dossier <b>PIP.COM</b> du drive <b>B</b>
<b>ERA *.BAS</b>	effacera tous les dossiers de type <b>.BAS</b>

### 3.6.5 La commande REN

La commande **REN** vous permet de changer le nom (**REN**ommer) d'un dossier existant pour lui en donner un autre. Le nouveau nom est spécifié en premier puis le signe égal = et en fin le nom du dossier que vous voulez renommer. Si le nouveau nom existe déjà sur la disquette, un message d'erreur s'affichera. Les jockers ne sont pas permis.

**REN HELLO.BAS = HALLO.BAS**  
**REN B: HELLO.BAS = HALLO.BAS**

Le dossier qui s'appelait **HALLO.BAS** s'appelle maintenant **HELLO.BAS**.

### 3.6.6 La commande TYPE

(To type = imprimer) La commande **TYPE** demande que le dossier spécifié soit imprimé à l'écran. Si ce n'est pas un dossier de type ASCII, des choses imprévisibles vont apparaître à l'écran.

**TYPE EX1.BAS**

## 3.7 Les commandes transitoires

Pour réaliser sur vos dossiers des opérations un peu plus sophistiquées qu'avec le mode direct, vous pouvez employer un des nombreux programmes utilitaires fournis avec le CP/M. On fait appel à eux simplement en tapant leur nom suivi parfois d'un ou plusieurs paramètres. Vous avez déjà probablement utilisé **FORMAT** et **FILECOPY**.

Nous avons regroupé ces commandes par catégories. Vous en trouverez une documentation complète dans le guide du CP/M (Soft159).

Les commandes **SYSGEN**, **BOOTGEN**, **FILECOPY**, **COPYDISC**, **DISCCOPY**, **CHKDISK**, **DISCCHK**, **FORMAT**, **SETUP**, **CSAVE**, **CLOAD** et **AMSDOS** sont des commandes spéciales à AMSTRAD et ne marchent que sur le CPC-464. Elles ne peuvent pas être employées dans les autres systèmes à CP/M, bien que l'on puisse trouver leurs équivalents (souvent sous le même nom) sur les autres machines.

### 3.7.1 La gestion des périphériques

L'utilitaire **PIP** (Programme d'Interconnexion aux Périphériques) vous permet d'échanger des informations entre l'ordinateur et ses périphériques. La forme de ces commandes est généralement : **PIP** (destination) = (source).

La source et la destination peuvent être, soit un nom de dossier (jockers acceptés), soit le code du périphérique. On pourra utiliser les codes suivant :

En tant que source	En tant que destination
<b>CON</b> : clavier	<b>CON</b> : écran
<b>RDR</b> : interface série	<b>PUN</b> : interface série
	<b>LST</b> : imprimante

exemples :

<b>PIP B:=A:*.COM</b>	Copie tous les dossiers avec extension <b>.COM</b> du drive A sur le drive B.
<b>PIP SAV.BAS=EX1.BAS</b>	Fait une copie de <b>EX1.BAS</b> et l'appelle <b>SAV.BAS</b>
<b>PIP CON:=EX1.BAS</b>	Envoie le dossier <b>EX1.BAS</b> à l'écran (même effet que la commande <b>TYPE EX1.BAS</b> )
<b>PIP LST:=EX1.BAS</b>	Envoie le dossier <b>EX1.BAS</b> sur l'imprimante
<b>PIP TYPEIN.TXT=CON:</b>	Accepte des données au clavier et les range dans un fichier appelé <b>TYPEIN.TXT</b> . L'opération s'achève avec <b>[CTRL]J</b> après <b>ENTER.[CTRL]J</b> rentre le code ASCII du retour à la ligne

Notez que **PIP** ne peut être utilisé pour des copies de dossiers avec un système à un seul drive. Utilisez alors, **FILECOPY**.

## 3.7.2 Copies de dossiers et de disquettes.

### 3.7.2.1 Copie d'un dossier

L'utilitaire **FILECOPY** vous permet de copier des dossiers d'une disquette sur autre avec un système à un seul drive. Il vous donne les instructions nécessaires pour les changements de disquettes. Si vous utilisez des noms de dossiers avec jockers, il vous demandera confirmation pour chaque dossier individuellement en même temps qu'il donne le nom de chaque dossier copié.

<b>FILECOPY *.BAS</b>	Copie tous les dossiers de type <b>.BAS</b>
<b>FILECOPY EX1.BAS</b>	Copie le dossier <b>EX1.BAS</b>

### 3.7.2.2 Copie de disquettes entières et vérificateurs

**DISCCOPY** pour un système à un drive et **COPYDISC** pour un système à deux drives vous permettent de faire des duplication de disquettes entières. Toutes les instructions nécessaires vous seront données à l'écran. Les disquettes non formatées ou d'u formatage différent seront formatées automatiquement. Les utilitaires **DISCCHK** et **CHKDISC** vous permettent de comparer les deux disquettes et de vérifier qu'elles ont bien exactement le même contenu.

### 3.7.2.3 Dossiers sur cassette

Deux utilitaires sont disponibles pour des transferts de cassette à disquette. Mais à part pour des spécialistes, il est peu propable que des transferts de dossiers autres que des textes en codes ASCII puissent susciter un intérêt.

**CLOAD** accepte deux paramètres; le premier, le nom du dossier source (sur cassette) entre guillemets, et le second, le nom du dossier destination (sur disquette). Si ce dernier nom est omis, le dossier sur disquette portera le même nom que celui sur cassette. Si le nom du dossier source est omis, **CLOAD** lira le premier nom qu'il rencontrera sur la bande. Si le premier caractère du nom du dossier sur cassette est un **!**, les messages ordinaires liés à l'utilisation des cassettes seront alors supprimés. Exemple :

**CLOAD "MALETTRE" MALETTRE.TXT**

**CSAVE** accepte trois paramètres. Le premier, le nom du dossier source (sur disquette), le deuxième le nom du dossier destination (sur cassette), encadré de guillemets. Si le nom du dossier destination est omis, il portera le même que l'autre. Si le premier caractère du nom du dossier sur cassette est un **!**, les messages 'cassette' seront supprimés. On pourra alors utiliser un troisième paramètre pour spécifier la vitesse d'enregistrement; 0 pour 1000 bauds, 1 pour 2000 bauds. Exemples :

**CSAVE OUTPUT.TXT "OUTPUT TEXTE" 1**  
**CSAVE DONNEES**

## 3.7.3 Gestion du système

### 3.7.3.1 STAT

La commande **STAT** apporte des sophistications à la gestion du catalogue. Toutes les règles que l'on appliquait aux noms de dossiers restent valables, jockers compris. Elle permet :

L'affichage de l'état de disquette et de l'espace vacant :

**STAT**

**STAT A:**

**STAT B:**

L'affichage de compléments d'information sur un dossier particulier

**STAT \*.COM**

**STAT EX1.BAS**

L'attribution à un dossier du statut 'lecture seulement', de façon à ce qu'il ne puisse pas être détruit accidentellement.

**STAT \*.COM SR/O**

**STAT EX1.BAS \*R/O**

La remise à l'état normal après une mise en 'lecture seulement'

**STAT \*.COM SR/W**

**STAT EX1.BAS SR/W**

L'attribution à un dossier du statut particulier de système qui le rend invisible sur le catalogue de la disquette et inaccessible aux programmes de copie de dossiers. Il restera cependant disponible à tout autre usage.

**STAT \*.COM SSYS**

**STAT SECRET.BAS SSYS**

L'inversion du statut précédent. Redonne au dossier son statut visible par le catalogue :

```
STAT *.COM DIR SDIR
STAT SECRET.BAS SDIR
```

### 3.7.3.2 SETUP

Cet utilitaire vous permet de redéfinir les caractéristiques du clavier de votre CPC-464, du lecteur de disquette, de l'interface série et de lancer diverses actions au moment du chargement du CP/M. Quand il a terminé, il remet à jour la zone des variables système. Ce programme est dirigé par un menu et quand un point est correct ou ne requiert aucune modification, passez au point suivant en répondant **Y** à la question

**is this correct (Y/N):** - (ceci est-il correct ?)

On peut arrêter le programme à tout moment par **[CTRL]C**. Une fois tous vos changements effectués, il vous demandera

**Do you want to update your system disc (Y/N):**

Confirmez vous cette remise à jour de la configuration de votre système

Ce qui vous permet de revenir à l'ancienne configuration (en tapant **N**) et;

**Do you want to restart CP/M (Y/N) :**

Voulez-vous reinitialiser le CP/M

Vous permettant d'initialiser et donc de tester votre nouvelle configuration en répondant **Y**.

Pour copier la zone où sont sauvegardées les variables du système d'une disquette sur une autre, utilisez soit **BOOTGEN** (voir plus loin), soit chargez **SETUP** du disque source, répondez **Y** à toutes les questions, insérez la disquette destination juste avant de répondre à la dernière question.

Les caractères qui ont une valeur ASCII inférieure à 32 peuvent être mis à l'intérieur d'une chaîne alphanumérique en tapant **!** suivi du caractère convenable tiré de la série **,A...Z,[,\, ],>' —**

Les options suivantes sont celles qui reviennent le plus souvent :

©©**Initial command buffer** (zone de commandes à l'initialisation)

Tous les caractères entrés ici, apparaîtront comme s'ils avaient été tapés en mode direct au moment du chargement du CP/M. On peut ainsi démarrer le lancement automatique d'un programme à ce moment-là. Souvenez-vous d'y inclure l'équivalent de la touche **[ENTER]** représenté par les deux caractères **!M**. Par exemple, pour le lancement automatique de la commande **STAT**, vous devez taper dans cette zone : **STAT!M**

**Sign-on string** (le message d'introduction)

C'est le message affiché en haut de l'écran quand on charge le **CP/M**. Notez l'utilisation de **!J** **!M** pour l'entrée de la ligne et le retour chariot.

La première partie de ce message sert à donner des instructions pour la couleur de l'écran et de la bordure et pour le mode (voir le chapitre 9 du guide de l'utilisateur du CPC-464). Elle doit être recopiée telle quelle si l'on garde les mêmes caractéristiques.

**Keyboard** translations (changement des codes clavier). Pour donner de nouvelles valeurs ASCII aux touches du clavier, comme le fait la commande '**KEY DEF**' du BASIC d'AMSTRAD. Référez-vous au guide de l'utilisateur pour connaître les codes ASCII courants.

**Keyboard** expansions (expansion du clavier). Fait la même chose que la commande **BASIC' KEY'**

### 3.7.3.3 AMSDOS

Ce programme permet l'abandon du contrôle par le CP/M pour revenir à l'AMSTRAD BASIC d'où les commandes de gestion des disquettes de l'AMSDOS sont disponibles.

## 3.7.4 Production de disquettes

### 3.7.4.1 FORMAT

L'interface disquette DDI-1 d'AMSTRAD accepte trois sortes de format de disquette dont l'un a deux variantes.

Le format courant est le format système, obtenu en utilisant la commande **FORMAT** toute seule. Les pistes système sont lues de la disquette contenant le programme **FORMAT.COM** et sont automatiquement écrites sur la disquette à formater.

Les autres formats sont obtenus par l'adjonction d'un paramètre, une lettre séparée de la commande par un espace.

Pour un format de type données : **FORMAT D**

Pour un format de type IBM : **FORMAT I**

Pour un format de type Vente : **FORMAT V**

**\*\*AVERTISSEMENT\*\*** Le brevet de votre disquette CP/M (qui possède un numéro de série codé électroniquement) vous permet son utilisation sur un seul micro ordinateur. Ceci signifie en particulier qu'il vous est interdit de donner une copie de votre CP/M (Copie qui posséderait alors votre n° de série). Etant donné que chacune de vos disquette de format système contient le CP/M (le votre), il vous est interdit de vendre, d'échanger tout ce qui pourrait être sur une disquette de ce format. Vous devez donc utiliser le format Vente qui est identique au format système, exception faite des pistes système qui y sont vides. Vous recopiez ensuite votre logiciel sur cette disquette en utilisant FILECOPYY ou PIP. Attention à ce que le logiciel que vous copiez ainsi ne soit pas lui-même sujet à des droits d'auteur ou à des droits réservés.

Si vous faites l'acquisition d'un logiciel sur une disquette de format Vente, vous devez pour l'utiliser, copier ce logiciel sur une disquette système en utilisant FILECOPYY ou PIP ou au contraire en convertissant cette disquette en disquette système en lui adjoignant le CP/M. Utilisez pour ceci les commandes **BOOTGEN** et **SYSGEN**.

**LISEZ ATTENTIVEMENT LE CONTRAT D'UTILISATION  
A L'APPENDICE 2 DE CE MANUEL.**

### 3.7.4.2 MOVCPM

On peut avoir besoin parfois d'utiliser une version du CP/M qui ne se charge pas en mémoire à sa place habituelle. Peut être pour réserver un emplacement mémoire à un autre usage, et donc le CP/M lui-même doit réduire sa taille mémoire. Il est possible de réduire le CP/M par tranches de 256 octets. La position est indiquée par un paramètre qui varie de 64 à 179. Ce paramètre indique le nombre de zones disponibles pour le CP/M et les commandes transitoires.

La redéfinition du CP/M peut alors être écrite sur la disquette système en utilisant **SYSGEN** ou alors est enregistrée en utilisant la commande **MOVCPM**.

La syntaxe de cette commande est : **MOVCPM (taille) \***. Exemple : **MOVCPM 178 \*** va réduire la zone mémoire de 256 octets par rapport à la version standard (Qui est la taille maximum, c'est-à-dire 179)

### 3.7.4.3 SYSGEN

**SYSGEN** écrit le résultat de la commande **MOVCPM** sur les pistes système de la disquette. Il y a trois options.

**SYSGEN \*** enregistrera le CP/M obtenu par la commande **MOVCPM** l'ayant immédiatement précédé.

**SYSGEN** (nom de dossier) lira dans le dossier dont le nom est spécifié (probablement à la suite d'une commande **MOVCPM**) et l'écrira sur les pistes système. Exemple : **SYSGEN CPM44.COM**

**SYSGEN** sans paramètre demandera une disquette source et une disquette destination. C'est l'option à utiliser pour transformer la disquette Vente en disquette Système.

### 3.7.4.4 BOOTGEN

Comme nous l'avons dit aux chapitres 3.2 et 3.3, il n'y a pas que le CP/M sur les pistes système. **BOOTGEN**, qui demandera les disquettes source et destination, copie le secteur 1 de la piste 0 (le chargeur) et le secteur des variables système d'une disquette sur une autre. Utilisez **BOOTGEN** pour la transformation de votre disquette Vente en disquette système ou pour distribuer une zone de variables système nouvellement créée sur toutes vos disquettes.

## 3.7.5 Programmation avancée

Les programmes suivants serviront surtout aux spécialistes et il est recommandé de consulter le guide du CP/M (Soft 159-A) ainsi que d'autres ouvrages spécialisés avant de les utiliser.

<b>ASM</b>	L'assembleur du 8080
<b>DDT</b>	Dispositif de mise au point des codes d'assemblage du 8080
<b>DUMP</b>	Utilitaire de travail sur dossiers en codes hexadécimaux
<b>ED</b>	Editeur de texte simple.
<b>SUBMIT</b>	Mode de commandes clavier d'enchaînement de programmes
<b>XSUB</b>	Utilitaire d'enchaînement de programmes.



# CHAPITRE 4

## Introduction au LOGO

*Objectif de ce chapitre : une prise de contact avec ce nouveau langage. Assorti d'exemples, il se complète d'un guide des commandes disponibles; ce n'est ni un cours exhaustif, ni un guide de référence qui sont l'objet du guide pour le logo (soft 160).*

Sujets abordés :

- \* Le concept de LOGO
- \* Chargement et mise en service
- \* Le graphisme de la tortue
- \* Ecrire vos propres méthodes
- \* Editer vos propres méthodes

### 4.1 Qu'est-ce que le LOGO

Le LOGO peut vous aider à devenir un programmeur, que vous ayez ou non programmé auparavant.

Le logo est un langage puissant qui devient de plus en plus populaire car il est simple à apprendre et à utiliser.

Vous vous servez de "méthodes" pour construire les différentes parties de votre programme. Le Dr. LOGO a sa propre collection de "méthodes" appelées "primitives" et, que vous utiliserez également pour programmer.

Pendant les années 1970, une équipe de chercheurs en informatique et d'éducateurs sous la direction de Seymour Papert mit au point le Logo et sa petite tortue qui dessine; ce qui permit à de très jeunes enfants de programmer et d'utiliser un ordinateur.

La tortue était pour ces jeunes apprentis, comme le dit Papert, un objet permettant de matérialiser un raisonnement, un outil pour aider à apprendre d'une façon différente.

De la forme d'une tête de flèche, la tortue peut être dirigée sur l'écran en utilisant des commandes simples.

## 4.2 Dr. LOGO

Le Dr. LOGO est une version très élaborée du Logo. Elle a été spécialement adapté au CPC-464 pour le rendre encore plus simple à programmer. Des extensions ont permis d'y inclure la possibilité d'utiliser les puissantes facultés sonores du CPC-464 et la correction du programme est facilitée par le pavé de gestion du curseur.

## 4.3 Préparons-nous

Pour mettre en service le Dr. Logo, insérez une copie de la face 2 de votre disquette système dans le drive.

(Vous verrez au début de ce livre les instructions qui permettent la copie de la disquette système).

Appuyez sur les trois touches **CTRL**, **SHIFT** et **ESC** en même temps pour remettre à zéro l'ordinateur.

Tapez **ICPM [ENTER]** et le LOGO se chargera automatiquement. Le message suivant apparaîtra sur l'écran de votre moniteur :

**Welcome to**  
**Amstrad LOGO V1.1**  
**Copyright (C) 1983/4. Digital research**  
**Pacific Grove, California**  
**Dr Logo is a trademark of digital research**  
**Please wait**

Ce message d'introduction disparaîtra bientôt pour faire place à un ?

Ce point d'interrogation vous dit que Dr. Logo est prêt à recevoir vos instructions au clavier.

## 4.4 Les premiers pas

Essayez de taper en minuscules : **fd 60 [ENTER] forward** = EN AVANT

et vous verrez apparaître une tortue qui avancera de 60 unités en laissant un trait derrière elle, du point de départ à son point d'arrivée. L'écran va s'éclaircir en donnant une zone pour le graphique très longue et un petit emplacement pour le texte avec le ? au bas de l'écran.

Dr. Logo décidera souvent de changer l'organisation de l'écran avec soit une grande zone pour le texte, soit une grande zone pour le graphisme selon votre besoin.

Tapez **rt90** (right: droite) **[ENTER]** et la tortue bougera de 90 degrés vers la droite. (A partir de maintenant, nous assumons que vous appuyez sur la touche **[ENTER]** à la fin de chaque commande).

Maintenant tapez **fd 60** et une autre ligne se dessinera de la même longueur et du même angle droit que la première.

Continuez votre apprentissage avec les instructions **fd, bk** (back: arrière) **lt** (left: gauche), pour voir ce qui se passe sur l'écran.

## 4.5 Les “méthodes de LOGO

Une méthode est une liste d'instructions qui va accomplir un certain travail.

Vous allez sûrement écrire vos premières méthodes en utilisant les instructions de Logo appelées “primitives”.

**fd**, **bk**, **rt** et **lt** sont des primitives de Logo que l'on peut utiliser à tout moment pour construire ses propres “méthodes”.

Une autre primitive très utilisée est **cs** elle nettoie l'écran et renvoie la tortue à la case de départ.

### 4.5.1 L'ÉLABORATION D'UNE MÉTHODE SIMPLE

Il est aisé de voir que **fd 60 rt 90**, répétés quatre fois vont tracer un carré de 60 unités de côté.

On peut l'écrire comme suit :

```
repeat 4 [fd 60 rt 90]
```

répétez quatre fois [avant 60 unités, droite 90 degrés]

Nettoyez l'écran et tapez ceci pour voir ce qui se passe.

Pour faire de cette formule une méthode que l'on, appellera carré il suffit de taper :

```
to carre pour carré  
repeat 4 fd 60 rt 90 répétez... end fin
```

Dr. Logo va maintenant comprendre “carré” et à chaque fois qu'il rencontrera le mot carré, il tracera un carré sur l'écran. On aurait pu donner n'importe quel nom à notre méthode, mais “carré” est un moyen on ne peut plus logique pour se souvenir de ce que fait cette méthode.

Dr. Logo nous permet de taper toutes sortes de combinaisons de style.

```
carré rt 45 carré (droite 45 degrés)
```

qui va dessiner 2 carrés, le second ayant un angle de 45 degrés par rapport au premier.

## 4.6 Méthodes avec paramètres

Il est possible de bâtir des méthodes auxquelles on pourra demander “combien” de même que les primitives acceptent les paramètres.

Pour bâtir une méthode qui va construire des carrés de n'importe quelle taille, il va falloir redéfinir “carré” comme suit :

```
to carré quelconque : côté  
repeat 4 [fd: côté rt 90 ] end
```

Cette nouvelle méthode introduit l'idée de variable qui dans ce cas est appelée : côté.

Vous remarquerez que la variable côté est précédée de 2 points qui indiquent à Dr. Logo que : côté est une variable et non pas une commande.

Lorsque nous utiliserons la méthode carré-quelconque : côté doit recevoir une valeur. Ainsi, carré-quelconque 150 donnera un carré en 150 unités de côté.

Essayez d'additionner 2 méthodes pour voir ce qui advient. Par exemple à partir d'une instruction.

```
cs carré-quelconque 100 rt 45 carré-quelconque 150
```

La tortue va dessiner 2 carrés de côtés différents, l'un faisant un angle de 45° par rapport à l'autre.

Notez comment Logo vous rappelle qu'une ligne de commandes s'est divisée en plusieurs lignes sur l'écran.

## 4.7 Utilisation de variables pour stocker des valeurs

Dr. Logo vous permet ainsi de stocker des valeurs dans une variable, variable que l'on peut utiliser également pour distribuer des valeurs à une méthode.

D'abord, faites-vous une méthode appelée triangle :

```
to triangle bord  
repeat 3 [fd: bord rt 120] end
```

On peut tester ceci en tapant

```
make "bord 100  
triangle
```

Si nous voulons connaître la valeur de bord, on a juste à taper : **bord** après le ? et Dr. Logo nous affichera sa valeur.

Enfin, nous pouvons utiliser notre variable : **bord** dans une nouvelle méthode qui va construire un dessin.

Remarquez la façon d'accroître la valeur de : **bord** en ajoutant quelque chose à sa valeur première de façon à ce que notre dessin s'étende.

```
to dessin  
triangle Lt 60 triangle rt 60  
make "bord : bord + 4  
dessin  
end  
make "bord 10  
cs dessin
```

Quand vous en avez assez, appuyez sur **ESC** pour arrêter le programme.

## 4.8 La correction des programmes et des méthodes

Dr. Logo vous permet de corriger les fautes de frappe, de modifier les méthodes. Les touches de correction utilisées sont :

Les touches curseur ↑ ↓ → ← pour le déplacement du curseur d'un caractère à la fois.

Les touches curseur ↑ ↓ → ← pendant que l'on appuie sur la touche **[CTRL]** permettent de se déplacer d'une extrémité à une autre en une seule fois.

[CLR] efface le caractère sous le curseur. [DEL] efface le caractère à gauche du curseur.

[ENTER] dit à LOGO que la correction d'une ligne de commande est terminée ou bien donne la ligne suivante dans le cas de la correction d'une méthode.

[ESC] signifie fin de correction et [COPY] signale à LOGO que la correction de la méthode est terminée.

Quand vous rentrez des commandes ou de nouvelles méthodes, corrigez le texte en face de vous sur l'écran. Tout autre caractère que ceux mentionnés plus haut seront insérés dans le texte à la position du curseur.

Pour la correction d'une méthode existante, utilisez la commande **ed**. Dr. Logo affichera alors l'ancienne version de la méthode sur l'écran et vous pourrez ainsi modifier ce que vous voulez en utilisant les commandes de gestion du curseur.

Essayez la correction de la méthode dessin en tapant **ed "dessin**

Testez les touches de correction. Quand vous avez terminé, appuyez sur la touche [ESC]. LOGO abandonne alors le mode correction et vous redonne la version originale inchangée de votre méthode.

Tapez **ed "dessin** à nouveau puis changez le 4 en 8, appuyez sur la touche [COPY] pour sortir et rappelez la méthode pour visualiser ce qui s'est passé. Songez à donner une valeur à **:bord**.

## 4.9 Quelques mots sur le fonctionnement

L'espace travail utilisé par LOGO est divisé en nœuds. Vous pouvez savoir combien il vous en reste en tapant **nodes**. Occasionnellement, quand presque tous les nœuds sont utilisés, LOGO nettoie tout l'espace travail et vous verrez alors la tortue faire une pause. On peut aussi demander au LOGO de faire ce nettoyage par la commande **recycle**, ceci pour vous permettre de continuer quand LOGO se plaint de n'avoir plus de nœuds disponibles.

Assurez-vous qu'il y a suffisamment de place sur votre disquette au cas où vous décidez de sauvegarder vos méthodes. Vous pouvez utiliser la commande **CAT** de l'AMSDOS. (voir chapitre AMSDOS).

Jetez un coup d'œil sur le dernier paragraphe ci-dessous et essayez quelques exemples. Vous ne comprendrez pas tout la première fois, mais plus vous avancerez dans votre apprentissage et plus vous utiliserez de commandes différentes.

Quand vous en avez terminé avec Dr. Logo, tapez **Bye**

## 4.10 Liste des primitives du Dr. Logo

Ce paragraphe dresse la liste des primitives du Dr. Logo par groupes et ordre alphabétique en montrant à chaque fois la forme syntaxique de la primitive et souvent accompagné d'exemples.

On peut écrire les noms des primitives en utilisant, soit des majuscules, soit des minuscules.

## 4.10.1 TRAITEMENT DE TEXTES ET DE LISTES

Notez que les messages ? et > sont inscrits dans les exemples qui suivent.

### **ascii**

Donne la valeur ASCII du premier caractère du mot rentré.

```
?ascii "G
71
?ascii "g
103
```

### **bf**

(but first = sauf le premier) Redonne tout sauf le premier élément du mot rentré.

```
?bf "sport
sport
?bf [1 2 3]
[2 3]
```

### **bl**

(but last = sauf le dernier) Redonne tout sauf le dernier élément du mot rentré.

```
?bl "sport
spor
?bl [1 2 3 4]
[1 2 3]
```

### **char**

donne le caractère qui correspond à la valeur ASCII rentrée

```
?char 83
S
```

### **count**

Donne le nombre d'éléments du mot rentré.

```
?count "six
3
?count [0 1 2 3]
4
```

## **empty**

répond TRUE (vrai) si le mot rentré est blanc, vide. Dans le cas contraire répond faux (FALSE)

```
?empty"  
TRUE  
?empty [ ]  
TRUE  
?empty [x]  
FALSE  
?make "x [ ]  
?empty:x  
TRUE
```

## **first**

(premier) donne le premier élément du mot rentré

```
?first "zèbre  
z  
?first [1 2 3]  
1
```

## **fput**

first put = met devant. Donne un nouveau mot en rajoutant le premier élément rentré juste devant le deuxième.

```
?fput "s "sportsport  
sportsport  
?fput 1 [2 3]  
[1 2 3]
```

## **item**

article. Donne l'élément du rang spécifié

```
?item 4"sport  
r
```

## **list**

Fait la liste des éléments rentrés et les encadre de crochets. A comparer avec la primitive **se**.

```
?{list 1 2 3 4}  
[1 2 3 4]  
?list "grand pied  
grand pied  
?(list)  
[ ]
```

## **se**

sentence = phrase. Donne la liste des éléments rentrés (à comparer avec list)

```
?make "instr-list rl  
repeat 4 [fd 50 rt 90 ]  
? run [se "cs : instr-list "ht
```

Notez que le tiret - entre **instr** et **list** est obtenu en appuyant sur **[SHIFT]0**

## **word**

mot. Donne un seul mot à partir des mots ou éléments rentrés.

```
?word "para "pluie  
parapluie
```

## **wordp**

Donne la réponse TRUE (vrai) si le ou les élément(s) rentré(s) sont soit un mot, soit un nombre.

```
?wordp "hello  
true  
?wordp [ ]  
FALSE
```

## **4.10.2 OPÉRATIONS ARITHMÉTIQUES**

### **cos**

donne le cosinus du nombre rentré en degrés

```
?cos 60  
0 .5
```

### **int**

Donne la partie entière du nombre rentré

```
?int 4/3  
1
```

### **random**

Donne un nombre entier positif pris au hasard et inférieur au nombre rentré.

```
?random 20
```

### **sin**

Donne le sinus du nombre rentré en degrés

```
?sin 30  
0 .5
```

+

Donne la somme des nombres rentrés

?+ 2 2

4

?2+2

4

—

Donne la différence des deux nombres rentrés

?— 10 5

5

?10 -5

5

\*

Donne le produit des deux nombres rentrés

?\* 4 6

24

?4\*6

24

/

Donne le quotient décimal des deux nombres rentrés

?/ 25 5

5

?25/5

5

### 4.10.3 OPÉRATIONS LOGIQUES

#### and

et. Répond TRUE (vrai) si le résultat de toutes les expressions rentrées est juste

? and (3<4) (7>4)

TRUE

## **not**

Répond TRUE (vrai) si l'expression rentrée est fausse

?not (3=4)

TRUE

?not (3=3)

FALSE

## **or**

ou. Répond FALSE (faux) si toutes les expressions rentrées sont fausses.

?or "TRUE "FALSE

TRUE

?or (3=4) (1=1)

TRUE

=

Répond TRUE (vrai) si les deux expressions rentrées sont équivalentes

?= "LOGO "LOGO

TRUE

?1=2

FALSE

>

Répond TRUE si le premier élément rentré est plus grand que le second

?> 19 20

FALSE

?20> 19

TRUE

< 27 13

FALSE

?13<27

TRUE

## 4.10.4 VARIABLES

### local

Rend les variables spécifiées accessibles à la méthode en cours  
(local "x "y "z)

### make

Pour donner une valeur à une variable

```
?make "côté 50  
?:côté  
50
```

## 4.10.5 MÉTHODES

### end

Indique la fin de la méthode. Ce mot doit être isolé, à la dernière ligne de la méthode.

```
?to carre  
repeat 4 [fd 50 rt 90 ]  
end  
carre defined (défini)  
?carre
```

### po

(print out = affichage. Affiche sur l'écran la méthode désignée ou la valeur des variables spécifiées.

```
?po "carre  
to carre  
repeat 4 [fd 50 rt 90 ]  
end  
?po "x  
x is (est) 3
```

### pots

print out titles = affichage des titres. Affiche les titres de toutes les méthodes de l'espace travail.

```
?pots
```

## **to**

Indique le début de la construction d'une méthode

? to **carre**

> **repeat 4** [fd 50 rt 90 ]

> **end**

**carre defined**

## **4.10.6 CORRECTION**

### **ed**

charge sur l'écran la méthode désignée ou les variables désignées dans la zone tampon qui sert pour la correction.

?ed "carre

## **4.10.7 ÉCRAN TEXTE**

### **ct**

(clear text = déblayage du texte)

Efface tout le texte de la fenêtre ou se promène le curseur et ramène le curseur en haut à gauche de la fenêtre.

?ct

### **pr**

(print = affiche). Affiche les mots rentrés au clavier sur l'écran texte, enlève les crochets et va à la ligne. (à comparer avec **show** et **show** et **type**)

?pr [a b c]

a b c

### **setsplit**

Pour définir le nombre de lignes de l'écran texte

?ss 10

### **show**

Affiche les mots rentrés au clavier sur l'écran texte, garde les crochets extérieurs et va à la ligne. (à comparer avec **pr** et **type**)

?show [a b c]

[a b c]

## **ts**

(text screen = écran texte). Pour réserver tout l'écran pour le texte.

?ts

## **type**

Affiche les mots rentrés au clavier sur l'écran texte, enlève les crochets et ne va pas à la ligne (à comparer avec **pr** et **show**)

?type [a b c]

a b c

## **4.10.8 L'ÉCRAN GRAPHIQUE**

Notez que l'écran est en mode 1, avec 4 couleurs et que c'est le même système de coordination qu'avec le BASIC AMSTRAD. En d'autres termes, chaque position sur l'écran sera arrondie au point écran ayant le numéro pair le plus près. Les quantités de rouge, de vert et de bleu pourront varier entre 0, 1 ou 2.

## **clean**

(to clean = laver). Efface tout l'écran graphique sans toucher à la tortue.

?fd 50

?clean

## **CS**

(clear = vide l'écran). Efface l'écran graphique et ramène la tortue en position (0,0) se dirigeant vers 0 (le nord) et avec le stylo baissé.

?rt 90 fd 50

?cs

## **dot**

Place un point sur l'écran à la position spécifiée par les coordonnées indiquées et dans la couleur de stylo en cours.

?dot [50 10 ]

## **fence**

Barrière. Pose une barrière qui empêche la tortue de sortir des limites de l'écran. **window** enlève cette barrière.

?fence

?fd 30 0

turtle out of bounds (la tortue est hors des limites)

## fs

(full screen = plein écran)  
Pour avoir tout l'écran graphique.

## ?fs

## pal

palette. Donne à l'écran les trois nombres représentant les quantités de rouge, vert, bleu du stylo.

?pal 2  
[0 2 2]

## setpal

Fabrication des couleurs. Les trois chiffres désignent les quantités de rouge, de vert, de bleu pour la couleur du stylo.

?setpal 3 [1 1 2]  
? pal 3  
[1-1 2]

## sf

Les données de l'écran. Affiche des informations sur l'état de l'écran graphique. La structure de cette information est : (couleur du fond), (état de l'écran), (longueur de l'écran texte), (fenêtre), (scrunch), (couleur de fond) = Toujours **0**, (état de l'écran) = **ss**, écran partagé en deux, **fs**, écran totalement graphique, **ts**, écran totalement réservé au texte, (longueur de l'écran texte) = nombre de lignes réservées au texte. (fenêtre) = **window** --pas de barrière, la tortue peut sortir, **wrap** -- réapparition de la tortue de l'autre côté de l'écran, **fence** --barrière; la tortue ne peut pas sortir des limites de l'écran (scrunch) = Toujours 1

?sf  
[0 SS 5 FENCE 1]

## SS

Réserve une fenêtre pour le texte sur l'écran graphique

?ss

## window

Permet à la tortue de sortir des limites de l'écran après un **wrap** ou un **fence**.

?fence fd 30 0  
Turtle out of bounds (tortue hors limites)  
?window  
?fd 30 0

**wrap**

Fait apparaître la tortue du côté opposé ou elle est sortie.

```
?cs wrap  
?rt 5 fd 1000  
?cs window  
?rt 5 fd 1000
```

## 4.10.9 LES GRAPHISMES DE LA TORTUE

**bk**

(back = arrière). Avance la tortue du nombre de pas indiqué dans la direction opposée au sens de la flèche.

```
?cs fd 150  
?bk 50
```

**fd**

(forward = avant). Avance la tortue du nombre de pas indiqué dans le sens de la flèche.

```
?fd 80
```

**ht**

(hide turtle = cache tortue). Pour rendre la tortue invisible. Accélère et clarifie le dessin.

```
?ht  
?cs fd 50  
?st
```

**lt**

(left = gauche). La tortue tourne du nombre de degrés indiqué vers la gauche.

```
?lt 90
```

**pd**

(pen down = stylo baissé). Le stylo est posé sur le papier et donc laisse une trace sur l'écran.

```
?fd 20 pu fd 20  
?pd  
?fd 20
```

## **pe**

(pen erase = stylo qui efface). Cette instruction transforme la couleur du stylo en 0, c'est-à-dire la couleur du fond et donc permet ainsi d'effacer les traits sur l'écran.

```
?fd 50
```

```
?pe
```

```
?bk 25
```

```
?fd 50
```

```
?pd fd 25
```

## **pu**

(pen up = stylo levé). Le stylo est levé et donc ne laisse plus de trace sur l'écran.

```
?fd 30
```

```
?pu
```

```
?fd 30
```

```
?pd fd 30
```

## **px**

(pen reverse = stylo inversé). Change la couleur de tout ce qui a été tracé auparavant par ce stylo en une autre couleur qui est son opposé ou son complément logique.

```
?fd 20 pu fd 20
```

```
?pd setpc 3 fd 20
```

```
?px
```

```
?bk 80
```

```
?fd 80
```

```
?pd bk 10 0
```

## **rt**

(right = droite). La tortue tourne du nombre de degré indiqué vers la droite.

```
?rt 90
```

## **seth**

Changement de direction. Tourne la flèche de la tortue vers la direction spécifiée en degrés. Les nombres positifs font tourner la tortue dans le sens des aiguilles d'une montre. Les nombres négatifs en sens inverse.

```
?seth 90
```

## **setpc**

Changement de la couleur du stylo. Le stylo prend la couleur indiquée par le nombre donné. 0 est la couleur du fond.

## setpos

Changement de position. Envoie la tortue à la position indiquée par les coordonnées spécifiées.

```
?setpos [30 20]
```

## setpos

Changement de position. Envoie la tortue à la position indiquée par les coordonnées spécifiées.

```
?setpos [30 20 ]
```

## st

(show turtle = montre tortue). Pour rendre visible à nouveau une tortue invisible.

```
?ht
```

```
?fd 50
```

```
?st
```

## tf

Les données de la tortue

Affiche une liste d'informations à propos de la tortue.

Le format en est :

(coordonnée x), (coordonnée y), (direction), (état du stylo), (couleur), (visibilité)

coordonnée x la coordonnées x du stylo

coordonnée y la coordonnée y du stylo

direction direction vers laquelle se tourne la tortue. Degrés

état du stylo **PD** stylo baissé **PE** stylo qui efface **PX** stylo inversé

**PU** stylo levé

couleur numéro de la couleur du stylo.

visibilité **TRUE** tortue visible, **FALSE** tortue invisible

```
setpos [15 30 ]
```

```
?rt 60
```

```
?setpc 3
```

```
?pe
```

```
?ht
```

```
?tf
```

```
[15 30 60 PE 3 FALSE]
```

## 4.10.10 GESTION DE L'ESPACE DE TRAVAIL

### ER

Pour enlever une méthode de l'espace travail

```
?er "carre
```

## **ern**

Pour effacer la ou les variables spécifiées de l'espace travail

## **nodes**

Affiche le nombre de nœuds qui restent disponibles sur l'espace travail

?nodes

## **recycle**

Libère toutes les places libres et réorganise l'espace de travail.

?recycle

?nodes

## **4.10.11 LES LISTES DE PROPRIÉTÉ**

### **glist**

(get list: donne la liste) permet d'obtenir une liste de tous les objets de l'espace travail auxquels on a attaché une ou des propriétés.

? glist ". DEF

### **gprop**

(get property: donne la propriété) permet d'obtenir la valeur de la propriété du nom désigné.

? make "hauteur "72"

? gprop "hauteur" APV

72"

### **plist**

(propertylist: liste des propriétés) permet d'obtenir la liste des propriétés rattachées à un mot.

? plist "hauteur

[. APV 72"]

### **pprop**

(put property: donne une propriété) permet de créer un lien entre un mot et une propriété.

? pprop "maître ".APV "corbeau

? :maître

Corbeau

### **remprop**

(remove property: élimine la propriété) élimine la propriété spécifiée de la liste de propriété du mot.

? remprop "maître ".APV

## 4.10.12 DOSSIERS SUR DISQUETTE

### **dir**

Catalogue. Permet d'obtenir une liste de noms de dossiers écrits en LOGO sur une disquette, soit spécifiée, soit prise par défaut.

```
?dir  
[DÉPART ÉTOILE CANARDS]  
?dir "b:  
[MOYENNE OUTILS ADRESSE]  
?dir "???AR???  
[DÉPART CANARDS]
```

### **load**

Charge le dossier spécifié, de la disquette sur l'espace travail de l'ordinateur.

```
?load "dossier  
?load "b:dessins
```

### **save**

écrit le contenu de l'espace travail dans un dossier sur la disquette spécifiée

```
?save "dessins
```

## 4.10.13 CLAVIER MANETTE DE JEU

### **buttonp**

Bouton de tir appuyé. Répond TRUE si le bouton de tir est mis. Les numéros 0 et 1 servent à identifier les deux manettes de jeu possibles.

```
?to tirer  
> label "loop  
> if (buttonp 0 [pr [tirer 0 !]])  
> if (buttonp 1) [pr[tirer 1!]]  
>go "loop  
>end
```

La position de la manette de jeu est testée par paddle

### **keyp**

Répond TRUE si un caractère vient d'être tapé au clavier ou attend d'être enregistré.

```
?to inkey  
> if keyp [op rc] [op " ]  
> en d
```

## **paddle**

Manette de jeu. Pour connaître les états des manettes 0 ou 1.  
Les différentes positions sont codées comme suit :

Valeur retournée	Signification
255	Rien
0	Vers le haut
1	En haut à droite
2	A droite
3	En bas et à droite
4	En bas
5	En bas et à gauche
6	Vers la gauche
7	En haut et à gauche

### **?paddle 0**

**255**

Les boutons de tir sont testés par la commande buttonp

## **rc**

Sort le premier caractère tapé au clavier

**?make "touche rc**

**(puis appuyez sur la touche X)**

**? : touche**

**X**

## **rl**

Retourne une ligne d'instructions. Doit être suivi d'un retour chariot.

**?make "inst liste rl**

**repeat 4 [fd 50 rt 90 ]**

**? : inst liste**

**[repeat 4 [fd 50 rt 90 ]**

## **rq**

Retourne une ligne d'instructions. Doit être suivi d'un retour chariot.

**?make "commande rq**

**repeat 3 [fd 60 rt 120 ]**

**? : commande**

**repeat 3 [fd 60 rt 120 ]**

## 4.10.14 LE SON

Seule la version AMSTRAD de Dr. LOGO possède des commandes pour le son. Elles sont d'ailleurs identiques à leur équivalent BASIC.

Reportez vous au chapitre 6 du guide de l'utilisateur pour plus de détails.

**sound** met un son dans une suite de sons (queue). Le format de cette instruction est (canal, période de ton, durée, volume, enveloppe de volume, enveloppe de ton, période de bruit). Tous les paramètres après durée sont facultatifs.

? sound [1 20 50 ]

### Env

Construit l'enveloppe du volume (la forme de la note). Son format est :

< numéro de l'enveloppe > + < section (s) >

### Ent

Construit l'enveloppe de ton. Son format est :

< numéro de l'enveloppe > < section (s) >

### Release

Libère les canaux qui ont été branchés par la commande sound. Les canaux à libérer sont indiqués par le code suivant :

Valeur rentrée    Canaux libérés

0	Aucun
1	A
2	B
3	B et A
4	C
5	C et A
6	C et B
7	C et B et A

? release 1

## 4.10.15 LES COMMANDES DE DÉROULEMENT DE PROGRAMME

### bye

Lorsque on en a terminé avec Dr. Logo.

? bye

### CO

Pour reprendre, continuer après une pause provoquée par un [CTRL]Z ou ERRACT.

?co

## go

Exécute la ligne de commande à l'intérieur d'une méthode. Cette ligne étant repérée par un mot étiquette avec lequel on l'appelle

```
>go "boucle
```

## if

Exécute une ou deux listes d'instructions suivant la valeur de l'expression rentrée qui doit être entourée de crochets

```
>if (a>b) [pr [a est plus grand]
```

```
>[pr [b est plus grand]
```

## label

Pose une étiquette sur une ligne d'instruction qui pourra ainsi être appelée à l'aide d'un go suivi de l'étiquette donnée.

```
>label "boucle
```

## op

(output: retourne) : prendre un objet quelconque en donnée et faire arrêter la méthode

## repeat

Répète la liste d'instructions, donner le nombre de fois spécifié après repeat

```
? repeat 4[fd 50 rt 90 ]
```

## run

Exécute la ligne d'instructions

```
? make "instr-list [fd 40 rt 90]
```

```
? run: instr-list
```

## stop

Arrête, suspend l'exécution de la méthode en cours et reviens au niveau supérieur TOPLEVEL (Le message ?) ou à la procédure d'appel

```
? stop
```

## wait

arrête l'exécution de la méthode pendant le temps spécifié par le nombre rentré.

La longueur du temps de pause est : nombre rentré (fois X) 0.22 secondes.

```
? wait 20
```

## 4.10.16 TRAITEMENTS DES CAS PARTICULIERS

### catch (attrape)

Pour traiter une erreur ou un cas particulier qui survient pendant l'exécution d'une série d'instructions.

```
> catch "error [+ [ ] [ ]]  
> pr [je suis ici]  
je suis ici
```

### error

Pour obtenir la liste d'instruction qui a causé la dernière erreur.

```
> catch "error [do.until.error]  
> show error
```

### pause

Suspend l'exécution de la méthode en cours permettant ainsi un dialogue entre Logo et l'utilisateur.

```
> if : Taille > 5 [pause]
```

### throw

Renvoie l'exécution de la série d'instructions identifiée par le nom rentré au moment du catch

```
? throw "TOPLEVEL
```

## 4.10.17 LES PRIMITIVES SYSTEMES

- **contents** affiche le contenu de l'emplacement mémoire des symboles
- **deposit** place le deuxième nombre rentré à l'emplacement mémoire donné par le premier nombre
- **examine** affiche le contenu de l'emplacement mémoire spécifié

## 4.10.18 LES VARIABLES SYSTEMES

### ERRACT

Quand TRUE provoque un arrêt au moment d'une erreur; renvoie alors au niveau supérieur TOPLEVEL.

### FALSE

Valeur système

### REDEFP

Quand TRUE permet la redéfinition des primitives.

## **TOPLEVEL**

pour revenir au message ?

**throw "TOPLEVEL"**

## **TRUE**

Valeur système

### **4.10.19 LES PROPRIÉTÉS DU SYSTÈME**

#### **.APV**

Valeur de la variable associée à la propriété.

#### **.DEF**

Définition d'une méthode.

#### **.PRM**

Identifie une primitive.

# CHAPITRE 5

## Aspects techniques du firmware (Logiciel intégré à la machine)

*Il est conseillé de connaître le firmware du CPC-464 avant d'aborder ce chapitre.*

Il est également important d'avoir quelques notions des principes fondamentaux du CP/M. Ce chapitre n'essayera pourtant pas de faire le tour des avantages du CP/M. Une information complète concernant le CP/M, l'environnement du CP/M sur le CPC-464 et les aspects techniques du BIOS sont contenus dans le guide du CP/M (SOFT159). Une information complète sur le firmware du DDI-1 d'AMSTRAD est disponible (SOFT158A) en appendice d'un guide technique très complet (SOFT158)

### 5.0 Introduction

L'AMSDOS est un système d'exploitation de disquettes utilisé par le CPC-464 pour communiquer avec l'interface disquettes DDI-1. L'AMSDOS permet aux programmes BASIC d'avoir accès à des fichiers aussi bien sur disquette que sur cassette et en fait, des programmes existants, créés pour une utilisation sur cassette, pourront servir sur disquette avec très peu ou pas du tout de modification. La source principale d'incompatibilité sera la dénomination des dossiers. Pour l'AMSDOS, ils doivent respecter les normes CP/M, alors que les noms de dossiers sur cassette nécessitent moins de rigueur.

L'AMSDOS a un rôle complémentaire par rapport au CP/M et ne rivalise pas avec lui. Ils ont la même façon de structurer les dossiers et peuvent s'interchanger que ce soit au niveau de la lecture ou de l'écriture. L'AMSDOS réside dans la même ROM que le BIOS du CP/M.

L'AMSDOS canalise automatiquement les entrées/sorties ( 9) de la cassette sur la disquette. Ainsi les commandes de gestion de dossiers sur cassettes deviennent disponibles sur disquettes. De plus, l'affichage du catalogue de la disquette, l'effacement de dossiers, la renomination d'un dossier et la sélection du drive sont grandement facilités.

Tout ceci est accompli, soit par l'interception des routines de gestion de cassette, soit par des commandes externes.

### 5.1 Entêtes

Les dossiers sur cassette sont subdivisés en blocs de 2K octets, chacun d'eux étant précédé d'une entête. Les dossiers CP/M eux, n'ont pas d'entête. Les dossiers AMSDOS peuvent avoir ou ne pas avoir d'entête en fonction du contenu du dossier. Ceci ne posera pas de problème si le programme est écrit en BASIC mais cela reste la grosse différence entre un dossier sur cassette et un dossier sur disquette. Ceci peut être exploité pour la protection des programmes.

Les dossiers ASCII non protégés n'ont pas d'entête. Tous les autres dossiers AMSDOS ont une entête unique contenu dans les premiers 128 octets du dossier, qu'on appelle aussi zone entête. Les entêtes sont détectées par addition des premiers 67 octets de cette zone. Si la somme donne le résultat prévu, alors il y a une entête sinon il n'y en a pas. Ainsi il est peu probable mais néanmoins possible qu'un dossier sans entête puisse être confondu avec un dossier avec entête.

## 5.2 Le changement de disquette

Sous AMSDOS, une disquette peut être sortie chaque fois que le drive n'est en train ni de lire, ni d'écrire et qu'aucun fichier n'ait été ouvert sur ce drive. Contrairement avec le CP/M, il n'y a pas besoin d'amorcer la disquette.

Enlever une disquette pendant une opération d'écriture peut détériorer les données de la disquette. Si elle est enlevée alors qu'un fichier est encore ouvert, dès que l'AMSDOS le détecte, le fichier sera abandonné et un message d'erreur apparaîtra. Les informations restant à écrire seront perdues et la dernière entrée sur le catalogue de la disquette ne sera pas prise en compte. Cependant l'AMSDOS ne peut détecter ce changement que quand il lit le catalogue, ce qu'il fait à chaque groupe de 16 K octets ou quand un fichier s'ouvre ou se ferme. Ainsi potentiellement, 16 K de données seront perdues si une disquette est sortie de son drive avec un fichier encore ouvert.

## 5.3 Les exigences de l'implantation mémoire

Au moment de l'initialisation, l'AMSDOS se réserve 500 octets de mémoire de la mémoire centrale. Cette valeur ne peut pas changer.

Lorsque l'on charge en mémoire un programme en langage machine à partir d'une disquette en utilisant la routine CAS IN DIRECT de l'AMSDOS, il est important que les variables de l'AMSDOS ne soient pas détruites par surimpression. Et c'est un problème car en général on ne sait pas où sont ces variables. Ceci parce que les variables externes sont réparties de façon dynamique en mémoire. Notez que ce problème ne se présente pas pour un chargement à partir d'une cassette car les variables de gestion des cassettes sont dans la zone protégée des variables système.

L'AMSDOS se réserve un emplacement mémoire à partir du haut de la mémoire centrale et dont la solution la plus simple est de stocker le programme en langage machine dans le bas de la mémoire centrale. Le programme pourra alors se déplacer vers une plus haute adresse si besoin est.

Les programmes en langage machine seront chargés en deux temps. D'abord, chargement et exécution d'un petit programme chargeur. MC BOOT PROGRAM aura fermé tous les RSX et les extensions ROM. Le chargeur va maintenant initialiser l'AMSDOS en utilisant KL INIT BACK, forçant ainsi les variables de l'AMSDOS à se mettre à l'endroit voulu. Le chargeur peut maintenant s'occuper du programme en langage machine en utilisant les routines CAS OPEN IN, CAS IN DIRECT et CAS IN CLOSE de l'AMSDOS en même temps que le programme MC START PROGRAMM.

Lorsque l'on utilise KL INIT BACK pour l'initialisation de l'AMSDOS on a besoin du n° de ROM de l'AMSDOS. Vous trouverez ce numéro en regardant le contenu de la zone tampon "interception cassette" avec les routines DISC sélectionnées. Chaque enregistrement est une adresse sur trois octets, le troisième octet étant le numéro de ROM.

## 5.4 Messages d'erreur

L'AMSDOS utilise le BIOS du CP/M pour accéder à la disquette. Et donc ce seront les messages du BIOS qui seront affichés en cas de problème matériel.

DRIVE veut dire A ou B. FILENAME veut dire nom de dossier sous AMSDOS

### 5.4.1 Messages AMSDOS

#### Bad command

La commande a échoué. Il y a une erreur de syntaxe dans la commande, dans le nom du dossier ou alors un C (option pour l'annulation) a été tapé après un message d'erreur du BIOS.

**(nom de dossier) already exists** Ce nom existe déjà.

Tentative d'utilisation d'un nom déjà en service

**(nom de dossier) not found** Pas trouvé

Tentative de travail sur un dossier qui n'existe pas

**Drive (drive): directory full** Le catalogue est plein

Plus d'espace disponible sur le catalogue (64 entrées par disquette)

**Drive (drive): disc full** Disquette pleine

Plus de secteurs disponibles sur la disquette.

**Drive (drive: disc changed, closing** (nom de dossier)

L'utilisateur a sorti la disquette avec un fichier ouvert.

**(nom de dossier) is read only** Dossier en lecture seulement

Tentative pour effacer ou renommer un dossier qui est marqué R/O (read only). Lecture seulement. Peut être causé également par une fermeture de fichier R/O.

### 5.4.2 Les messages du BIOS

Les messages du BIOS sont toujours suivis de la question: **Retry, Ignore or cancel ?**. (Refaire, continuer ou annuler ?). Vous devez alors taper soit **R**, soit **I**, soit **C**. Tout autre entrée se soldera par un petit bip sonore.

Tapez **R** (Retry = Recommence) et le BIOS recommence la manœuvre

Tapez **I** (Ignore). Il ignore la cause du problème et continue ou il en était resté.

Tapez **C** (Cancel = Annuler) et le BIOS abandonne l'opération en cours. Ce qui entraîne souvent un message d'erreur du BDOS.

Après avoir tapé **R**, **I** ou **C**, le curseur disparaît.

Les messages du BIOS AMSTRAD sont les suivants :

**Drive (drive): disc missing** Pas de disquette dans le drive spécifié.

Quand le BIOS fait une tentative sur un drive vide ou qui lui semble vide.

**Failed to load boot sector**

Produit pendant un démarrage à froid (après **1 CPM**) quand le secteur de lancement n'est pas lu correctement ou si tous les octets de ce secteur ont la même valeur.

**Failed to load CP/M**

Ce message se produit pendant un démarrage à chaud quand un secteur du module de traitement des commandes ou du BDOS n'est pas lu correctement ou si tous les octets du premier secteur du module ont la même valeur.

**Drive (drive): disc is write protected**

Quand le BIOS essaye d'écrire sur une disquette protégée en écriture. Si l'utilisateur souhaite écrire sur cette disquette, il doit la sortir, enlever la protection, la remettre dans le drive et taper **R** (Retry = Recommence)

**Drive (drive): read fail**

Message produit par un problème matériel rencontré pendant la lecture de la disquette. Il peut avoir été causé également par une tentative de lecture d'une disquette d'un mauvais format.

**Drive (drive): write fail**

Problème matériel survenu pendant l'écriture.

Il est recommandé après des erreurs de ce type, soit en écriture, soit en lecture de sortir la disquette, de la re rentrer puis de taper **R**. Elle peut avoir été mal positionnée ou gênée par quelque chose. L'importance des duplication ne doit pas être sous-estimée.

## 5.5 Organisation de la disquette

Le BIOS accepte trois formats de disquette différents; le format Système, le format Données et le format IBM. Sous AMSDOS, le format de la disquette est automatiquement détecté, à condition toutefois qu'il n'y ait pas de fichiers ouverts. La détection est possible étant donné la numérotation des secteurs différente pour chaque format.

Les disquettes 3" (3 pouces) sont dites double face. Les deux faces sont accessibles mais une seule à la fois, suivant de quel côté on a inséré la disquette. Il peut y avoir un format différent sur chacune des deux faces.

## **POINTS COMMUNS A TOUS LES FORMATS**

Simple face (les deux sont traitées séparément)

512 octets par secteur physique

40 pistes numérotées de 0 à 39

Un bloc de 1024 octets pour le CP/M

Un catalogue de 64 entrées

## **LE FORMAT SYSTÈME**

9 secteurs par piste numérotés de 41 à 49

2 pistes réservées

Le format système est le format le plus courant car le CP/M ne peut être chargé (démarrage à froid et démarrage à chaud) que d'une disquette de format Système. Les pistes réservées sont utilisées comme suit :

Piste 0 Secteur 41: Secteur d'initialisation

Piste 0 Secteur 42: Secteur des variables système

Piste 0 Secteur 43 à 47: inutilisé

Piste 0 Secteur 48, 49 et Piste 1 Secteurs 41 à 49: Module de traitement des commandes et BDOS.

Note : Le format Vente est une version du format système qui ne contient rien sur les deux pistes réservées. Il est utilisé pour le commerce des logiciels.

## **LE FORMAT DONNÉES**

9 secteurs par piste numérotés de C1 à C9

0 piste réservée

Format promu à un riche avenir mais pas recommandé pour l'instant avec le CP/M car on ne peut pas faire de démarrage à chaud sur ces disquettes. Cependant, si l'AMSDOS seul est utilisé, il fait gagner de la place sur la disquette.

## **FORMAT IBM**

8 secteurs par piste numérotés de 1 à 8

1 piste réservée.

Ce format est de la même logique que celui utilisé par le CP/M de l'IBM PC. Il est réservé à l'usage des seuls spécialistes.

# **5.6 Les routines d'interception par l'AMSDOS**

Quand l'AMSDOS est initialisé, il copie le contenu de la zone tampon 'pointeurs cassette' dans sa propre zone de données. Quand DISC est sélectionné, ce sont les 'pointeurs AMSDOS' qui vont dans la zone tampon. Quand TAPE est sélectionné, les 'pointeurs cassette' retournent dans leur zone initiale.

A l'initialisation ce sont les routines disquette qui sont sélectionnées.

## 5.7 Ré-interception par l'utilisateur

Pour obtenir une copie de la zone tampon, il faut procéder comme suit : Copiez les trois octets de cette zone tampon dans votre propre zone de données. Ne faite aucune supposition sur le contenu de ces trois octets. Puis remettez à sa place le contenu de zone tampon à l'aide d'un JMP, RST ou autre. Quand vous avez la main appelez l'adresse contenu dans la zone tampon. Quand vous reprenez le contrôle une nouvelle fois, remettez l'adresse en place.

## 5.8 Paramètres de retour

Les routines AMSDOS ont la même interface que les routines cassette, bien que dans certains cas, l'interprétation des paramètres de retour est différente. Les erreurs détectées et par les routines disquette et par les routines cassette sont retournées Carry faux, zéro vrai. Les erreurs détectées seulement par les routines disquette sont retournées Carry faux, Zéro vrai. Ce dernier cas correspond à la condition de la routine cassette BREAK. Dans les deux cas, le registre A contient un numéro d'erreur.

Quand une routine échoue (Carry faux), un numéro d'erreur sur 6 bits s'inscrit dans le registre A. Le bit 7 indique si ce numéro d'erreur a déjà été mentionné auparavant. Les numéros d'erreur sont :

- OE Le fichier n'est pas ouvert comme prévu
- OF Fin de fichier matériel
- 1A Fin de fichier logicielle
- 20 Mauvaise commande, sûrement un nom de fichier incorrect
- 21 Le fichier existe déjà
- 22 Le fichier n'existe pas
- 23 Le catalogue est plein
- 24 La disquette est pleine
- 25 La disquette a été changée avec un fichier ouvert
- 26 Fichier en lecture seulement.

Les erreurs détectées par le contrôleur de disquettes sont retournées avec des valeurs comprises entre 40.. 7F. En effet, dans ce cas le bit 6 est toujours 1. Les autres bits sont retournés comme suit :

- |                         |                                  |
|-------------------------|----------------------------------|
| bit 5 erreur de données | - CRC                            |
| bit 4 dépassement       |                                  |
| bit 3 disque pas prêt   | - pas de disque dans le lecteur  |
| bit 2 pas de données    | - ne peut pas trouver le secteur |
| bit 1 on ne peut écrire | - le disque est protégé          |
| bit 0 k.adresse manque  |                                  |

## 5.9 Les routines d'interception du Firmware

Ces routines sont :

En lecture

125	# BC77	CASIN OPEN	Ouvrir un dossier comme entrée
126	# BC7A	CAS IN CLOSE	Ferme le dossier d'entrée
127	# BC7D	CAS IN ABANDON	Ferme le dossier immédiatement
128	# BC80	Lit un caractère	
129	# BC83	CAS IN DIRECT	Lit le dossier
130	# BC86	CAS RETURN	Met le dernier caractère lu
131	# BC89	CAS TEST EOF	La fin du dossier est-elle atteinte

En écriture

132	# BC8C	CAS OUT OPEN	Ouvre un dossier de sortie
133	# BC8F	CAS OUT CLOSE	Ferme le dossier correctement
134	# BC92	CAS OUT ABANDON	Ferme le dossier immédiatement
135	# BC95	CAS OUT DIRECT	Ecrit un caractère
136	# BC98	CAS OUT DIRECT	Ecrit le dossier directement à partir du disque

En catalogue

137	# BC9B	CAS CATALOG	Donne un catalogue à partir de la cassette
-----	--------	-------------	--



# APPENDICE 1

## Lexique

### Accès direct

La possibilité de lire ou d'écrire une information en mémoire ou sur disquette dans n'importe quel ordre.

### AMSDOS

AMStrad Disc Operating System. Système d'exploitation de disquette de l'AMSTRAD. C'est un programme qui permet de travailler en BASIC avec les disquettes.

### ASCII

Dossier ASCII: La représentation en codes ASCII d'une commande ou d'un dossier en code machine.

### BDOS

Module d'exploitation des disquettes. C'est une partie du CP/M qui sert d'interface entre le programme utilisateur et les fonctions du CP/M.

### BIOS

Module de gestion des entrées : sorties. C'est la partie matérielle du CP/M, spécialement écrite pour un type d'ordinateur. Toutes les entrées sorties (écran, clavier, disquette, etc..) sont gérées par le BIOS.

### Catalogue

Une zone sur la disquette où sont notées les entrées de chaque dossier. Egalement, la liste des dossiers contenus sur la disquette.

### CCP

Module de traitement des commandes. Il interprète ce qui est tapé au clavier. Ordinairement, les commandes sont rentrées au clavier, chargées dans le module CCP puis exécutées.

## Commande intégrée

Commande qui fait partie du système d'exploitation. Elles sont toujours plus rapides que les commandes transitoires car on n'a pas besoin d'aller les chercher sur la disquette.

## CP/M

Control Programm for Micro-computers, un système d'exploitation de disquettes mis au point par Digital Research et qui offre la possibilité d'adapter une large gamme de logiciels à des micro-ordinateurs différents.

## Démarrage à chaud

Lorsque l'on fait un **CTRL C** sous CP/M. Le démarrage à chaud réinitialise le système d'exploitation sans détruire les données en mémoire centrale.

## Démarrage à froid

L'amorçage et l'initialisation d'un système d'exploitation. Le démarrage à froid du CP/M se fait par la commande **ICPM**.

## Disquette

Disque, disquette. Une surface ronde en plastique, recouverte de chaque côté d'une couche d'oxyde magnétique que l'on utilise pour sauvegarder des informations. La disquette est logée à l'intérieur d'une enveloppe protectrice, avec une fenêtre pour accéder aux données.

Sur une disquette 3" (trois pouces) la fenêtre possède un obturateur qui se referme automatiquement quand la disquette est hors du drive.

## Dossier

De l'anglais **FILE**. Fichier programme groupe de données, d'information généralement stocké sur une cassette ou sur une disquette.

## Double face

Une disquette qui peut emmagasiner des informations sur chaque face. Un drive qui peut avoir accès aux 2 faces sans que l'on ait besoin de tourner la disquette.

## Drive

Le lecteur de disquettes. Le système mécanique utilisé pour entraîner les disquettes et accéder aux données à la surface.

## Dr. LOGO

C'est le Logo de Digital Research, un langage de programmation avec une tortue pour matérialiser le curseur graphique.

## Duplication

La reproduction fidèle d'une information de façon à se prémunir contre la perte, la destruction de l'original. C'est la plupart du temps la reproduction intégrale d'une disquette.

## **Expression**

En Logo, c'est le nom d'une méthode suivi des variables de la méthode.

## **Graphismes de la tortue**

L'image graphique laissée sur l'écran par le passage de la tortue. Quand la tortue se déplace, elle laisse la trace de son passage sur l'écran graphique de Dr. Logo.

## **Initialisation**

Le chargement du système d'exploitation en mémoire centrale. Quand on commande le CP/M à partir du BASIC, un petit programme d'amorçage se charge automatiquement de la disquette, petit programme qui gère le chargement du reste du système d'exploitation.

## **Interface**

Quelque chose qui permet à deux systèmes indépendants de communiquer. Par exemple le CPC-464 et son lecteur de disquettes.

## **Interface disquette**

Le système électronique qui permet au drive de communiquer avec, et le logiciel, et les opérations internes du micro-ordinateur qui lui est rattaché.

## **Jocker**

Soit \*, soit ?. Le Dr. Logo n'accepte que le ?. Le jocker \* remplace n'importe quel nombre de ?. Ils sont utilisés pour bâtir des noms de dossiers ambigus. Le ? peut être n'importe quel caractère alphanumérique.

## **Lecture seulement R/O**

Une option attribuée à une disquette, à un dossier sur disquette ou à un drive de façon à protéger les données qu'il contient d'un effacement involontaire.

## **Logo**

Le nom d'un langage de programmation dérivé du mot grec logos qui veut dire mot. Le LOGO se propose d'enseigner les principes fondamentaux de programmation.

## **Méthode**

Une série d'expressions, de primitives permettant d'accomplir une tâche précise.

## **Mode direct**

Ou encore mode clavier. Le message **A>** apparaît sur l'écran et le CP/M attend l'entrée au clavier d'une commande ou d'un utilitaire CP/M.

## Nœuds

Une unité de stockage sur l'espace de travail LOGO. Un nœud type utilise 4 octets d'espace mémoire.

## Nom de dossier

En LOGO, un nom de dossier peut contenir huit caractères alphanumériques. En CP/M, une extension de trois caractères précédée d'un point est permise.

## Nom de dossier ambigu

Un nom de dossier contenant un ou plusieurs jockers. Cette ambiguïté est utilisée pour travailler sur plusieurs dossiers à la fois.

## Page zéro

Zone de la mémoire centrale qui va de l'adresse 0000 à l'adresse 0100. Cette zone sous CP/M est utilisée pour stocker les très importants paramètres système.

## Pistes

Ordonnées de façon concentrique sur la disquette; chaque piste contient le même nombre de secteurs. Les pistes et les secteurs se façonnent à un endroit spécifique au moment du formatage.

## Pistes système

Pistes réservées pour le CP/M sur une disquette.

## Primitive

Procédure, opération, commande résidente du Dr. LOGO.

## Progiciel

Programme qui démarre automatiquement à l'initialisation du système .

## Programme transitoire

Programmes utilitaires du CP/M comme par exemple **FILECOPY** qui sont chargés en mémoire (TPA) et exécutés en tapant leurs noms au clavier.

## Protection en écriture

Une sécurité contre la destruction des données sur disquette. Une disquette protégée en écriture ne peut être que lue.

## **Secteur**

Un bloc de données. Le système d'exploitation de l'AMSTRAD utilise une sectorisation de 512 octets.

## **Simple face**

En référence à une disquette qui n'a qu'une seule face disponible pour le stockage des données.

## **Tampon**

Symbole graphique en forme de tête de flèche qui fonctionne comme un curseur graphique sur l'écran graphique du Dr. LOGO.

## **T.P.A. (Transient Program Area)**

Zone des programmes transitoires. Une zone mémoire qui commence à l'adresse 0100 et qui est utilisée par les programmes transitoires du CP/M.

## **Utilitaire**

Un programme sur la disquette qui permet à l'utilisateur d'accomplir certaines tâches. Voir programmes transitoires.



# APPENDICE 2

## CONTRAT D'UTILISATION DES LOGICIELS DIGITAL RESEARCH & AMSTRAD

### Remarques pour les utilisateurs

*Lisez soigneusement cette notice et n'ouvrez pas votre progiciel avant d'avoir lu ce contrat.*

*L'ouvrir indique votre accord avec les conditions et les limitations d'utilisation ci-dessous.*

### 1. DÉFINITIONS

Dans ce contrat, les termes :

1. DRI signifie DIGITAL RESEARCH (CALIFORNIE) INC, propriétaire des droits d'auteur et des brevets des programmes.
2. Machine signifie un micro-ordinateur unique. Les systèmes à plusieurs micro-ordinateurs ont besoin de permis supplémentaires.
3. Programme signifie l'ensemble des programmes qui forme ce logiciel, la documentation et le matériel en relation avec ce progiciel mais les annexes et les mises à jour diverses de DRI, quelque soit la façon de l'utiliser et les modifications qui y sont apportées.
4. AMSTRAD signifie AMSTRAD CONSUMER ELECTRONICS PLC, Brentwood, ESSEX.

Vous assumez la pleine responsabilité du choix de votre programme, ainsi que pour la mise en service, l'utilisation et les résultats obtenus par ce programme.

### 2. PERMIS

Vous pouvez :

1. Utiliser le programme sur une seule machine.
2. Copier le programme sur support magnétique ou sur papier pour des duplications ou pour des modifications mais pour une seule machine. Vous pouvez faire jusqu'à trois copies pour les raisons invoquées ci-dessus. (Certains programmes cependant pourront être prévu pour limiter ou même empêcher toute copie. Ils sont dits alors protégés). La copie de la documentation est interdite. Le désassemblage est interdit.
3. Modifier le programme et/ou l'incorporer à un autre programme pour votre propre usage sur une machine. (Toute partie de logiciel incorporée à un autre programme continuera à être sujette aux mêmes conditions du présent contrat); et,

4. Céder le logiciel et sa licence à un tiers si vous notifiez à DRI le nom et l'adresse de ce tiers et que ce tiers :
  - a) accepte les conditions de la licence
  - b) signe et envoie à DRI une copie de la carte d'enregistrement
  - c) paye les taxes de transfert.

Si vous cédez le programme, vous devez en même temps, soit céder toutes les copies y compris l'original, que ce soit sous forme imprimée ou magnétique, soit alors détruire tout ce qui n'a pas été cédé.

Ceci inclu toutes les modifications et portions de programme contenus ou ajoutés dans d'autres programmes. Vous devez reproduire et inclure à toutes copies, modifications etc., la notice concernant les droits de reproduction réservés.

**CHAQUE DISQUETTE POSSÈDE UN NUMÉRO DE SÉRIE ET VOUS NE DEVEZ NI UTILISER, NI COPIER, NI MODIFIER, NI DONNER A UN TIERS LE PROGRAMME OU UNE COPIE DU PROGRAMME OU UNE MODIFICATION OU UNE PARTIE SAUF DANS LES CONDITIONS PRÉVUES DANS CE CONTRAT. SI VOUS CÉDEZ VOTRE PROGRAMME OU UNE COPIE, MODIFICATION, PARTIE, VOTRE CONTRAT PREND FIN AUTOMATIQUEMENT.**

### 3. ÉCHÉANCES

Le contrat est effectif jusqu'à ce qu'il soit terminé. Vous pouvez le clôre à tout moment en détruisant copies, modifications, parties, etc., avec le programme. Il prend fin également sous certaines conditions citées plus loin dans ce texte ou si vous ne respectez pas les closes du présent contrat. Vous devez dans ce cas détruire programme, copies, modifications, etc...

### 4. LIMITES DE GARANTIE

**LE PROGRAMME EST LIVRÉ TEL QUEL. NI DRI, NI AMSTRAD NE GARANTISSE SOUS QUELQUE FORME QUE CE SOIT, Y COMPRIS LES GARANTIES IMPLICITES A TOUT ACHAT DE MARCHANDISE, NI LE PROGRAMME, NI SON APTITUDE A ACCOMPLIR UNE TACHE PARTICULIÈRE. LE RISQUE, AU NIVEAU DE LA QUALITÉ OU DE LA PERFORMANCE DU PROGRAMME EST ENTIÈREMENT VOTRE. SI LE PROGRAMME SE RÉVÈLE DÉFECTUEUX, TOUTES LES CHARGES, RÉPARATIONS, CORRECTIONS, CHANGEMENTS SONT A VOS FRAIS.**

Ni DRI, ni AMSTRAD ne garantisse que les fonctions contenues dans ce programme vont correspondre à vos besoins ou que son utilisation soit ininterrompue ou sans erreur.

Cependant AMSTRAD garantit que la disquette sur laquelle se trouve le programme est sans défauts et de bonne façon dans des conditions normales d'utilisation pour une période de 90 jours à partir de la date d'achat (prouvé par une photocopie de votre reçu).

## 5. LIMITES DES RECOURS

Devant l'entière responsabilité d'AMSTRAD, votre seul recours sera le remplacement de la disquette défectueuse (voir limites de garantie) que vous nous retournerez en même temps que la photocopie de votre reçu.

**EN AUCUN CAS, NI DRI, NI AMSTRAD NE SERONT RESPONSABLES POUR LES PRÉJUDICES CAUSÉS, Y COMPRIS LES PERTES EN GAIN, LES PERTES EN PROFIT OU LES PERTES INDIRECTES, MÊME SI DRI OU AMSTRAD ONT ÉTÉ AVISÉS DE LA POSSIBILITÉ DE TELS PRÉJUDICES.**

## 6. CARTE D'ENREGISTREMENT

DRI se réserve le droit de mettre à jour, améliorer ses programmes de temps en temps. Les mises à jour vous seront fournies seulement dans le cas où une carte d'enregistrement dûment signée a été retournée au siège de DRI. DRI n'est en aucun cas tenu ni de faire ses mises à jour, ni de vous les envoyer.

**CE CONTRAT NE POURRA ÊTRE MODIFIÉ QUE PAR UN AMENDEMENT ÉCRIT FAIT PAR UNE PERSONNE AUTORISÉ D'AMSTRAD OU DE DRI.  
VOUS RECONNAISSEZ QUE VOUS VENEZ DE LIRE, DE COMPRENDRE ET D'AGRÉER AUX TERMES ET CONDITIONS DE CE CONTRAT.**



## I A

AMSDOS .....	F4.1 Ch2.1
AMSDOS commandes .....	Ch 2.6
AMSDOS entêtes .....	Ch2.3 Ch5.1
AMSDOS jokers .....	Ch2.4
AMSDOS messages d'erreur ..	Ch2.13 Ch 5
AMSDOS noms de dossier .....	Ch2.2
AND .....	Ch4.9
• APV .....	Ch4.24
ASCII .....	Ch2.10 Ch4.6
ASM .....	Ch3.9
Amorçage disquette .....	Ch3.2
Arrêt d'une fonction CP/FM .....	E4.10
I B .....	E4.1 Ch2.7
BF .....	Ch4.6
BK .....	Ch4.15
BL .....	Ch4.6
BOOTGEN .....	Ch3.9
BUTTONP .....	Ch4.19
BYE .....	Ch4.21
Bouton poussoir .....	E2.4
CAT .....	E4.6
CATCH .....	Ch4.23
CHAR .....	Ch4.6
CLEAN .....	Ch4.13
CLOAD .....	Ch3.6
CO .....	Ch4.21
COPYDISC .....	E4.9 Ch1.2
COS .....	Ch4.8
COUNT .....	Ch4.6
I CP/M .....	E 4.3 Ch2.8
CP/M .....	Ch3.1
CP/M noms de dossiers .....	Ch3.3
CP/M pistes systèmes .....	Ch3.1
CS .....	Ch4.13
CSAVE .....	Ch3.6
CT .....	Ch4.12
Cas d'exception .....	Ch4.23
Changement disques .....	Ch5.2
Chargement de logiciel .....	E3.1
Codes de contrôle .....	Ch3.2
Commande manette (LOGO) .....	Ch3.2
Commandes clavier (LOGO) .....	Ch4.19
Commandes écran .....	Ch4.13
Commandes texte .....	Ch4.12
Commandes de la tortue .....	Ch4.15
Commandes disques (LOGO) .....	Ch4.19
Commandes mode direct .....	Ch3.3
Commandes sonores (LOGO) .....	Ch4.21
Commandes textes .....	Ch4.6
Commandes transitoires .....	Ch3.4
Configurer un prog. CP/M .....	Ch1.4
Configurer un secteur .....	Ch3.2
Connections .....	E1.2
Copie d'écran .....	Ch2.4
Copie de disquettes .....	Ch3.5
Copie de dossiers .....	F4.7 Ch2.10 Ch3.5
DDT .....	Ch3.9
• DEF .....	Ch4.24
• DEPOSIT .....	Ch4.23
I DIR .....	Ch2.8
DIR (CP/M) .....	Ch3.3
DIR (LOGO) .....	Ch4.19
I DISC .....	E4.1 Ch2.8
I DISC. IN .....	E4.2 Ch2.8
I DISC. OUT .....	E4.2 Ch2.8
DISCCHK .....	E4.10
DISCCOPY .....	E4.8 Ch3.5
DOT .....	Ch4.13
I DRIVE .....	Ch2.8
DUMP .....	Ch3.9
Directory (catalogue) .....	Ch2.2
Disque BASIC .....	Ch1.2
Disque Système .....	Ch1.2

ED .....	Ch3.9	LOAD (LOGO) .....	Ch4.19
EMPTYP .....	Ch4.7	LOCAL .....	Ch4.11
END .....	Ch4.11	LOGO .....	Ch4
ENT .....	Ch4.21	LT .....	Ch4.15
ENV .....	Ch4.21	Lexique .....	App. 1
ER .....	Ch4.17	Licence .....	App. 2
I ERA .....	Ch2.9	Logiciels .....	E3.1
ERA .....	Ch3.3	MAKE .....	Ch4.11
ERN .....	Ch4.18	MOVCPM .....	Ch3.9
ERRACT .....	Ch4.23	Messages BIOS .....	Ch5.3
ERROR .....	Ch.4.23	Messages d'erreur (CP/M BIOS) ....	Ch.3
• EXAMINE .....	Ch4.23	Messages d'erreur (AMSDOS) ....	Ch2.13
Editer (modifier) .....	Ch4.4 Ch4.12	Mise en route .....	E1
Entêtes .....	Ch2.3 Ch5.1	NODES .....	Ch4.17
FALSE .....	Ch4.23	NOT .....	Ch4.10
FD .....	Ch4.15	Noms de dossier .....	Ch2.2 Ch2.3
FENCE .....	Ch4.13	OP .....	Ch4.22
FILECOPY .....	Ch1.2 Ch3.5	OR .....	Ch4.10
FIRST .....	Ch4.7	Opérateurs arithm. (+—★/) .....	Ch4
FORMAT .....	F4.4 Ch3.8	Opérateurs logiques (= < >) .....	Ch4.10
FPUT .....	Ch4.7	Opérations arithmétiques .....	Ch4
FS .....	Ch4.14	Opérations logiques .....	Ch4.9
Firmware (logiciel cable) .....	Ch5.1	Organisation du disque .....	Ch5.5
Format IBM .....	Ch5.5	PADDLE .....	Ch4.20
Format Système .....	Ch5.5	PAL .....	Ch4.14
Format données seulement .....	Ch5.5	PAUSE .....	Ch4.23
GLIST .....	Ch4.18	PD .....	Ch4.15
GO .....	Ch4.22	PE .....	Ch4.16
GPROP .....	Ch4.18	PIP .....	Ch3.4
HT .....	Ch4.15	PLIST .....	Ch4.18
IF .....	Ch4.22	PO .....	Ch4.11
INT .....	Ch4.8	POTS .....	Ch4.11
ITEM .....	Ch4.7	PPROP .....	Ch4.18
Indicateur .....	E2.4	PR .....	Ch4.12
Interceptions .....	Ch5.7	• PRM .....	Ch4.24
Jeux .....	E3.1	PU .....	Ch4.16
Jokers .....	Ch2.4	PX .....	Ch4.16
Jumpblock .....	Ch5.6	Périphériques .....	Ch3.4
KEYP .....	Ch4.19	Primitives .....	Ch4.5 Ch4.23
LABEL .....	Ch4.22	Primitives Systèmes (LOGO) .....	Ch4.23
LIST .....	Ch4.7	Procédures (méthodes) .....	Ch4.3
LOAD (AMSDOS) .....	E4.6	Progiciel BASIC .....	Ch1.2

Progiciel CP/M .....	Ch1.3	THROW .....	Ch4.23
Propriétés système (LOGO) .....	Ch4.24	TO .....	Ch4.12
Protection (écriture) .....	E2.1	TOPLEVEL .....	Ch4.24
RANDOM .....	Ch4.8	TRUE .....	Ch4.24
RC .....	Ch4.20	TS .....	Ch4.13
RECYCLE .....	Ch4.18	TYPE (CP/M) .....	Ch3.4
REDEFP .....	Ch4.23	TYPE (LOGO) .....	Ch4.13
RELEASE .....	Ch4.21	Types de dossiers .....	Ch2.2
REMPROP .....	Ch4.18	I USER .....	Ch2.9
REN .....	Ch3.4	Utilitaires .....	Ch1.2
I REM .....	Ch2.9	Variables .....	Ch4.4 Ch4.11 Ch4.22
REPEAT .....	Ch4.22	Variables (sauvegarde) .....	Ch2.4
RL .....	Ch4.20	Variables Système (LOGO) .....	Ch4.23
RO .....	Ch4.20	WAIT .....	Ch4.22
RT .....	Ch4.16	WINDOW .....	Ch4.14
RUN (AMSDOS) .....	E4.6	WORD .....	Ch4.8
RUN (LOGO) .....	Ch4.22	WORDP .....	Ch4.8
Read only (lecture seulement) .....	Ch2.1	WRAP .....	Ch4.15
SAVE (AMSDOS) .....	E4.5	XSUB .....	Ch3.9
SAVE (LOGO) .....	Ch4.19		
SE .....	Ch4.8		
SETH .....	Ch4.16		
SETPAL .....	Ch4.14		
SETPC .....	Ch4.16		
SETPOS .....	Ch4.17		
SETSPLIT .....	Ch4.12		
SETUP .....	Ch3.7		
SF .....	Ch4.14		
SHOW .....	Ch4.12		
SIN .....	Ch4.8		
SOUND .....	Ch4.21		
SS .....	Ch4.14		
ST .....	Ch4.17		
STAT .....	Ch3.6		
STOP .....	Ch4.22		
SUBMIT .....	Ch3.9		
SYSGEN .....	Ch3.9		
I TAPE .....	E4.1 Ch2.9		
I TAPE. IN .....	Ch2.9		
I TAPE. OUT .....	Ch2.9		
TF .....	Ch4.17		













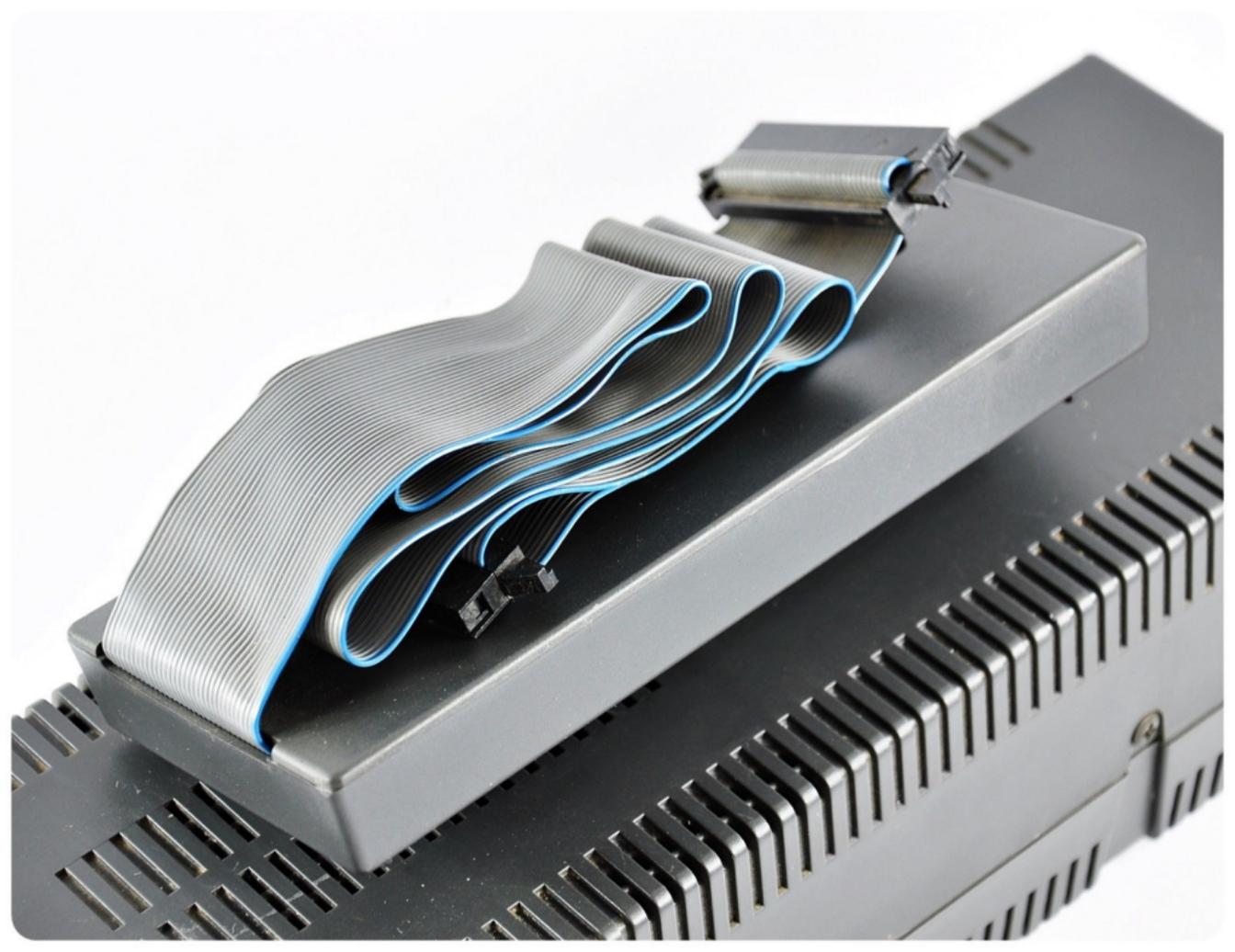




AMSTRAD  
Typ: FLOPPY DISC DRIVE DD1-1  
BETRIEBSSPANNUNG: 220-240V~50Hz  
LEISTUNGS-AUFNAHME: 15W  
VOR FEUCHTIGKEIT UND HITZE  
SCHÜTZEN!  
VOR GERÄTEÖFFNEN  
ZIEHEN!  
MADE IN KOREA

SERIEN NR. 543-8415319

VERSION  
A









**AMSTRAD**  
MODEL DDI-1 FROM MAINS  
DISCONNECT FROM MAINS  
SUPPLY WHEN NOT IN USE  
MAINS SUPPLY 220-240V -  
THIS UNIT COMPLIES WITH  
EMC DIRECTIVE 82/499/EEC  
WARNING: LIVE PARTS INSIDE  
DO NOT REMOVE ANY SCREWS -  
AMSTRAD PLC  
MADE IN KOREA  
SERIAL NO 5438323276

4

**AMSTRAD**

MODEL DDI-1 FROM MAINS  
DO NOT CONNECT WHEN NOT IN USE.  
DISCONNECT FROM MAINS SUPPLY: 220-240V ~  
50/60 HZ. UNIT COMPLIES WITH  
COMMERCIAL UNIT DIRECTIVE 82/499/EEC  
WARNING-LIVE PARTS INSIDE.  
DO NOT REMOVE ANY SCREWS.  
AMSTRAD PLC.

MADE IN KOREA

SERIAL NO. 543-8323276

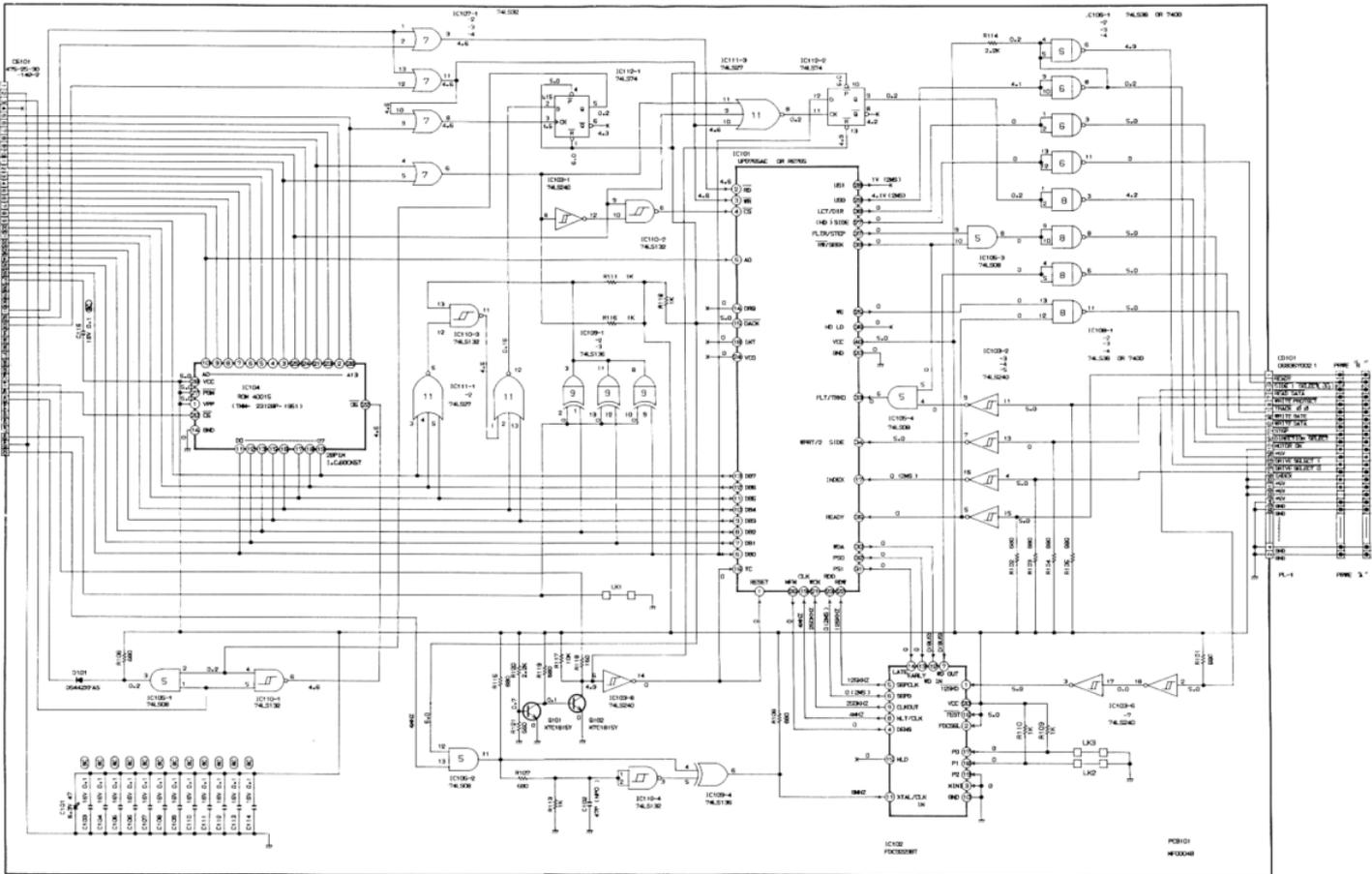
REF'S VERSION  
A







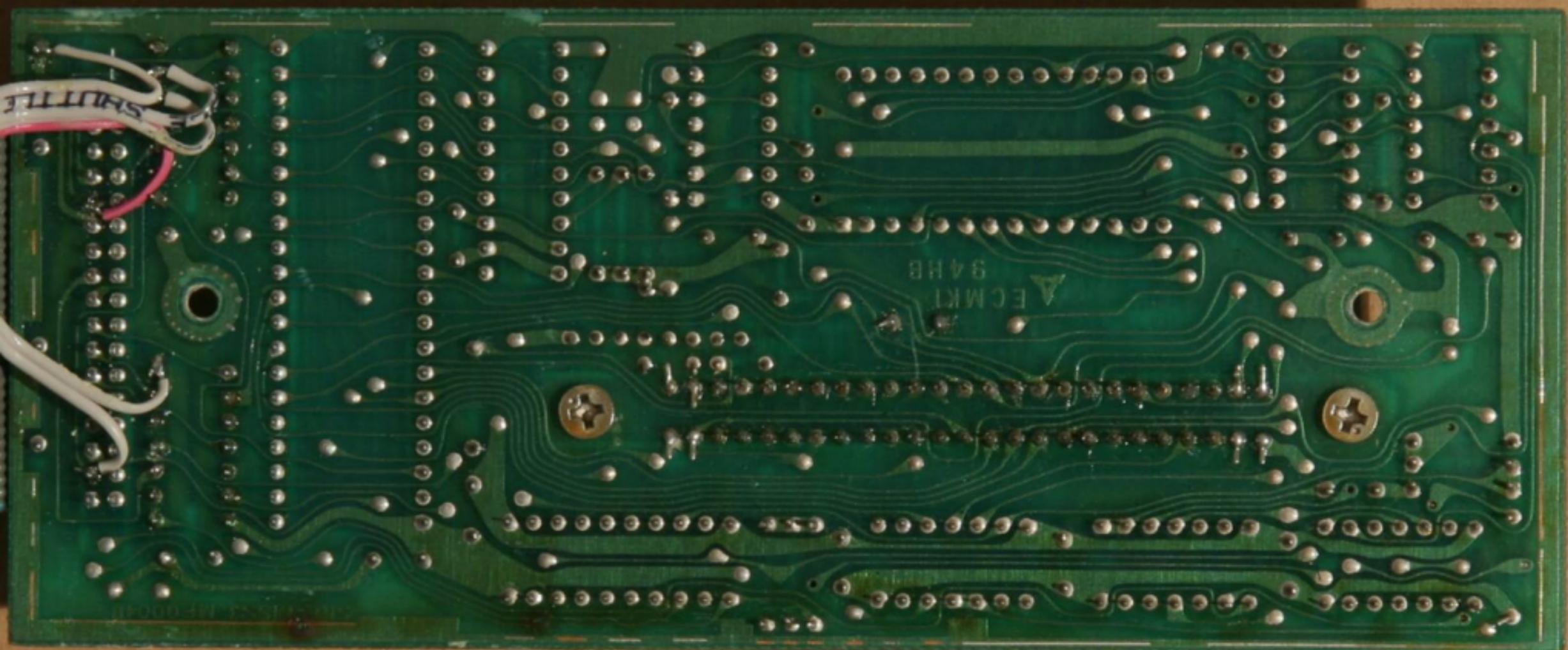
# INTERFACE CIRCUIT DIAGRAM



CONDITION: HOLD SWITCH OF FLIPPY DISK IS (BROKEN OR, NOT READY TO WORK, AND THE DISKETT IS NOT IN DRIVE).

NOTE: THIS SCHEMATIC DIAGRAM IS THE LATEST AT THE TIME OF PRINTING AND SUBJECT TO CHANGE WITHOUT NOTICE.





33-29  
Power line 5V

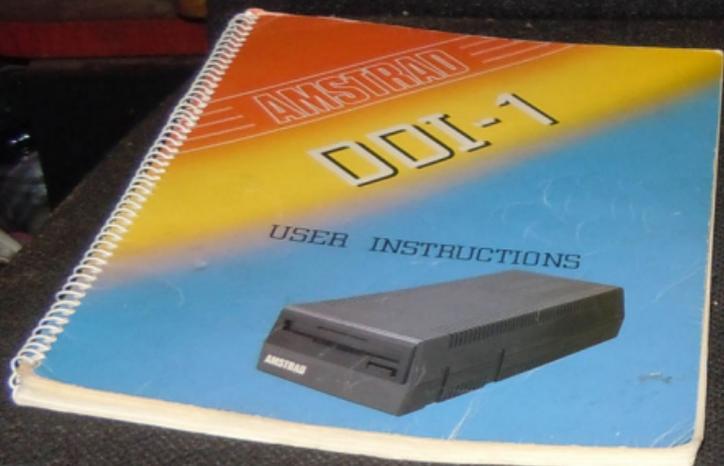
25 DS0

21 Power line 5V

33-29  
No Power Line

23 DS1

21 No Power line



18.10.2014 16:12

