

COPFORM V2.2

par André Chéramy

Bizarre, quand même que ce programme fabuleux de J.-P. Marie ne soit toujours pas utilisé par plus de gens, bien que le Ceo-Mag en ait déjà parlé (n°56 pages 5 et 6 de décembre 1994). En fait je proposais alors une interface utilisateur facilitant l'emploi de Copform et corrigeant quelques bogues et même améliorant ses possibilités (la version d'origine ne permettait pas de travailler sur les disquettes double face).

Utilisant régulièrement ce programme, j'ai été amené à revoir la présentation de cette interface afin quelle soit plus conviviale. Au passage j'ai également corrigé quelques bogues supplémentaires dont une de mon cru. Fondamentalement, il s'agit donc toujours de la même version 2.2 proposée en 1994, simplement un peu améliorée. Le kit sera disponible avec la prochaine disquette trimestrielle (et même avant si besoin : il suffit de demander).

QU'EST-CE QUE COPFORM ?

Rappelons les propriétés magiques de cet utilitaire hors normes. Tout d'abord, **il peut traiter des disquettes provenant de n'importe quel Dos**. Ensuite, il offre les 3 services suivant :

- 1) Copie de disquette : un BACKUP tout DOS. Si toutes les pistes n'ont pas le même nombre de secteurs (pour cause de protection par exemple), il faudra opérer par groupe de pistes. La disquette cible doit avoir été formatée de manière compatible (nombres de face, pistes et secteurs). Sauf format exotique, la commande INIT de Sedoric suffit dans la plupart des cas.
- 2) Vérification de disquette : une sorte de Scandisk façon Oric. Détecte les secteurs défectueux.
- 3) Recherche du nombre de secteurs par piste : Pour connaître le format réel de la disquette à copier et notamment pour déceler un type de protection assez courant : La piste bricolée.

STRUCTURE DU PROGRAMME

La version originale (1.0) de Copform est un pseudo Basic (début #501, fin #1000, checksum #3C3F) qui se résume à une ligne : 10 CALL#600. Vous avez compris : Copform est un programme en langage machine. Mais sa mise au point a dû être très difficile, car sa structure est très compliquée, avec des «greffes» multiples, qui rendent toute modification périlleuse. Une interface très rustique (messages incompréhensibles) et surtout des bogues. Cela ne retire rien au génie de Jean-Pierre Marie.

En 1994, ayant désassemblé le code de Copform, je me suis rendu compte que le plus simple pour améliorer la situation était de POKer mes corrections. Mais la place restante dans la zone Basic étant quasiment nulle, j'ai dû trouver une astuce, simple si possible.

J'ai alors écrit un petit programme (une sorte de «pré-couche»). Après avoir posé quelques questions relatives aux caractéristiques de la disquette source et à la configuration du système, ce programme affichait à l'écran un certain nombre de POKEs, puis chargeait Copform proprement dit. Il suffisait de repasser les POKEs de l'écran avec CTRL/A pour modifier la copie de Copform en mémoire. Et le tour était joué ! Enfin presque, car il fallait encore faire attention à un certain nombre de questions. Par exemple, répondre «O» (oui) à la demande «UTILISEZ VOUS UN 2EME LECTEUR?», même si ce n'était pas le cas...

COPFORM 2

CF2.COM est un programme BASIC utilisant COPFORM2.COM, une version améliorée de COPFORM.COM, qui marche sous Sédoric double face.

CF2 permet de copier et de vérifier des disquettes tout DOS et même d'en déterminer le format piste par piste

- 1) Charger le programme: CF2 <RETURN>
- 2) Répondre aux questions
- 3) Repasser l'écran affiché avec CTRL/A

-Si double face procédez en 2 passes
-Il y a un bug si le nombre de pistes à copier est un multiple de 8
Si besoin, procédez en deux passes

<touche>

```

CAPS
*** INITIALISATION DE COPFORM ***

Ce programme permet la copie ou la
verification des disquettes, quelque
soit le DOS. Il ne peut travailler que
sur une face à la fois. Si besoin
faire deux passes. La détermination
automatique du nombre de secteurs par
piste est basée sur la 1ère piste. En
cas d'erreur sur cette piste, indiquez
manuellement la valeur à utiliser.

Lecteur SOURCE (A à D)? A
Lecteur CIBLE (A à D)? B

Nombre de SECTEURS par piste
(1 à 19, 0 si détermination auto)? 17

No de PISTE initiale (0 si 1ère face,
128 si 2ème, sinon autre valeur)? 0
    
```

```

CAPS

Repasser avec CTRL/A:

POKE#78F, #0:POKE#99D, #0
POKE#D0B, #0:POKE#639, #0
POKE#7BE, #0:POKE#77B, #1

DOKE#7C5, #E54C:DOKE#7C7, #EA07
DOKE#7ED, #084C:DOKE#7EF, #EA08

DOKE#BBA, #A9EA:POKE#BBC, #11
DOKE#B7A, #A9EA:POKE#B7C, #11:CALL#600
Ready
    
```

AMÉLIORATION ACTUELLE

En fait, l'écran à repasser au CTRL/A était assez chargé et j'ai un peu révisé la chose. Il y avait deux sortes de POKEs. Les premiers étaient liés à des bogues permanentes. Les seconds aux choix dépendant des opérations à effectuer dans un cas particulier (par exemple, lecteurs source et cible, 1e ou 2e face etc.).

J'ai donc appliqué les POKEs «permanents» au fichier Copform et je l'ai sauvé sous le nom 'COPFORM2.COM' (début #501, fin #1000, checksum #5E4D). Puis j'ai simplifié ma «pré-couche» en ne gardant que les POKEs liés à des choix de travail.

La nouvelle «pré-couche» s'appelle maintenant 'CF2.COM' (début #501, fin #B2C, checksum #DD66) et appelle 'COPFORM2'. Le tout est accompagné d'un écran d'aide

'CF2.HLP' (voir figure). Une bogue présente à la ligne 230 du listing présenté dans le Ceo-Mag n°56 a aussi été corrigé (variable «S» remplacée par variable «Q»). Cette ligne est dédoublée et devient :

```
@ J.P.MARIE & A.CHERAMY COPFORM 02.2
COPIE
VERIFICATION PISTES
SECTEURS PAR PISTE

VOTRE CHOIX (C/V/S) █
```

```
@ J.P.MARIE & A.CHERAMY COPFORM 02.2
NBRE DE PISTES A COPIER (HEXA) 2A █
@ J.P.MARIE & A.CHERAMY COPFORM 02.2
EST-CE LA DISQUETTE EMETTRICE (O/N) █
```

```
230 IF R$<"A" OR R$>"D" THEN 220
240 Q=ASC(R$) - 65
```

Attention, cette dernière correction est récente et j'ai hélas diffusé une disquette avec une ligne 240 boguée. En cas de doute, vérifier la ligne 240.

MODE D'EMPLOI DE CF2

Il est indiqué dans «CF2.HLP». Après avoir lancé CF2, il faut répondre à quelques questions (lecteurs source et cible, nombre de secteurs par piste (0 pour détermination auto, basée sur la 1e piste traitée) et n° de la 1e piste à traiter. Ne pas confondre «nombre de piste», par exemple 42 et «numéro de piste», qui vont par exemple de 0 à 41.

Dans CF2, on peut indiquer des valeurs décimales ou hexadécimales (avec un «#» devant). Attention Copform lui-même est plus restrictif et réclame des valeurs hexadécimales

(sans «#»). Une dernière bogue subsiste lorsque le nombre de piste à copier est un multiple de 8 (par exemple 80). Il faut alors opérer en deux passes. Pour opérer sur un groupe de pistes particulier (par exemple des pistes n°4 à 41), il faut indiquer le n° de la première piste (ici 4) et plus tard (dans COPFORM2) le nombre de piste (ici 38 soit 26 en hexadécimal). Cette manière de faire est un concept général. Pour copier les pistes de la 1e face, il faut indiquer zéro et pour copier celles de la 2e face, il faut indiquer 128 comme n° de première piste. CF2 affiche alors un écran dont il faut repasser une à une les lignes avec CTRL/A. On se retrouve alors dans COPFORM2.

UTILISATION DE COPFORM2

Un menu d'accueil propose 3 options «C» pour Copie de disquette, «V» pour Vérification de disquette et «S» pour Secteurs par pistes (détermination du format réel, c'est à dire du nombre de secteurs pour chacune des pistes indiquées).

COPIE DE DISQUETTE

Dans «CF2», vous avez indiqué le n° de la première piste à traiter. Maintenant, il faut indiquer en hexadécimal sans «#» le nombre de pistes à copier. Mettre en place la disquette source (et éventuellement la disquette cible, si vous avez indiqué deux drives différents dans «CF2») et répondre «O» à la question suivante.

Ne quittez pas l'écran des yeux pendant la copie, afin de noter les éventuelles erreurs de lecture ou d'écriture (voir le remède plus loin).

```
@ J.P.MARIE & A.CHERAMY COPFORM 02.2
17 (<#11) SECTEURS PAR PISTE █
@ J.P.MARIE & A.CHERAMY COPFORM 02.2
INTRODUIRE LA DISQUETTE A TESTER
NBRE DE PISTES A VERIFIER (HEXA) 2A █
```

VÉRIFICATION DE DISQUETTE

Le programme parcourt la disquette ou plutôt le bloc de pistes à traiter (défini par le n° de la 1^e piste dans «CF2» et par le nombre de pistes à vérifier qui vous a été demandé). Il affiche ensuite la liste des secteurs ayant un problème.

NOMBRE DE SECTEURS POUR CHAQUE PISTE

Ici le programme inspecte chaque piste pour en déterminer le format. Il affiche au fur et à mesure le nombre de secteurs trouvé. Lorsqu'un nombre de secteurs pour une piste donnée est inférieur à celui des pistes voisines, cela peut indiquer soit un secteur illisible, soit une piste spéciale (par exemple protection). En effet, pour compter les secteurs d'une piste, le programme arrête sa lecture dès qu'il arrive sur un secteur illisible.

```
© J.P.MARIE & A.CHERAMY COPFORM 02.2
NBRE DE PISTES A VERIFIER <HEXA> 2A■

17 <#11> SECTEURS PISTE 0 <#00>
17 <#11> SECTEURS PISTE 1 <#01>
17 <#11> SECTEURS PISTE 2 <#02>
17 <#11> SECTEURS PISTE 3 <#03>
17 <#11> SECTEURS PISTE 4 <#04>
17 <#11> SECTEURS PISTE 5 <#05>
17 <#11> SECTEURS PISTE 6 <#06>■

© J.P.MARIE & A.CHERAMY COPFORM 02.2
17 <#11> SECTEURS PISTE 28 <#10>
17 <#11> SECTEURS PISTE 29 <#10>
17 <#11> SECTEURS PISTE 30 <#1E>
17 <#11> SECTEURS PISTE 31 <#1F>
17 <#11> SECTEURS PISTE 32 <#20>
17 <#11> SECTEURS PISTE 33 <#21>
17 <#11> SECTEURS PISTE 34 <#22>
17 <#11> SECTEURS PISTE 35 <#23>
17 <#11> SECTEURS PISTE 36 <#24>
17 <#11> SECTEURS PISTE 37 <#25>
17 <#11> SECTEURS PISTE 38 <#26>
17 <#11> SECTEURS PISTE 39 <#27>
17 <#11> SECTEURS PISTE 40 <#28>
17 <#11> SECTEURS PISTE 41 <#29>■
```

RÉSOLUTION DES PROBLÈMES

Si le format de la disquette à copier est tellement «alambiqué» (tordu) que COPFORM ne s'en sort pas, ou si la qualité physique de la disquette laisse un peu à désirer (erreurs de lecture), vous ne pourrez vous en sortir qu'en faisant appel à NIBBLE de F. Guillemé. Utilisez de préférence la version 2.4 déboguée par Fabrice Francès (voir le Ceo-Mag de février 2003, n°154, pages 54 et 55). Ce second utilitaire permet de copier une piste particulière ou un secteur donné, en toute connaissance de cause et avec des possibilités de récupération. Mais il est peu utilisable pour copier toute une disquette. Avec ces deux utilitaires en tandem, il est possible de venir à bout de n'importe quelle disquette (sauf bien sûr si elle a un problème physique trop grave).

EN PRATIQUE

Si vous avez entre les mains une disquette dont vous ne possédez pas le Dos ou si les commandes de ce Dos ne permettent pas de faire une copie des disquettes «spéciales» alors COPFORM peut vous aider. A l'aide de la fonction «S», vous pourrez lire piste par piste cette disquette et savoir combien il y a de secteurs par piste et de pistes par face. A l'aide de ces paramètres, vous pourrez formater une disquette cible avec la commande INIT de Sedoric, cette commande étant heureusement très souple. La fonction «V» vous renseignera éventuellement sur la fiabilité en lecture de cette disquette. Selon la structure de la disquette source (mono ou double face, piste ou bloc de pistes) vous pourrez alors copier la disquette source sur la disquette cible en une ou plusieurs passes.

BOGUES RÉSIDUELLES

Elles sont acceptables, mais il faut les connaître. Lors de la saisie des valeurs hexadécimales, les caractères «: ; <=> ? et @» dont le code Ascii est situé entre «9» et «A» sont acceptés ! Si le nombre de pistes à traiter est un multiple de 8 (par exemple 80), COPFORM copie aussi le 1^e secteur de la piste suivante (d'où un message d'erreur s'il ne peut pas). En cas de situation critique (rare) il faut contourner le problème en procédant en 2 passes (par exemple les 79 premières pistes, puis la dernière). Enfin, si le nombre d'erreurs rencontrées dépasse 255 (rare), le compteur repasse à zéro ! Il suffit de tendre l'oreille aux renâclements du lecteur de disquette pour s'en rendre compte.

LES LISTINGS DE CF2 ET DE COPFORM2

Tout ce qui suit est destiné aux curieux (ou à toute personne qui voudrait re-écrire une version fraîche de COPFORM). Rien ne vous empêche de sauter la fin de cet article, qui n'est pas indispensable à l'utilisation du programme. Pour les curieux donc, voici d'abord le listing de la «pré-couche» CF2 :

```

100 REM *** CF2.COM *** V141194 pour COPFORM Version 2.2
110 REM SAVEU"A-CF2",AUTO
120 PAPER0:INK7:CLS:PRINT@2,2;" *** INITIALISATION DE COPFORM *** "
130 ACCENT SET:PRINT@2,5;"Ce programme permet la copie ou la"
140 PRINT@2,6;"vérification des disquettes quelque"
150 PRINT@2,7;"soit le DOS. Il ne peut travailler que"
160 PRINT@2,8;"sur une face à la fois. Si besoin"
170 PRINT@2,9;"faire deux passes. La détermination"
180 PRINT@2,10;"automatique du nombre de secteurs par"
190 PRINT@2,11;"piste est basée sur la 1ère piste. En"
200 PRINT@2,12;"cas d'erreur sur cette piste, indiquez"
210 PRINT@2,13;"manuellement la valeur à utiliser."
220 PRINT@2,15;"Lecteur SOURCE (A à D)? «;:GET R$:PRINT R$
230 IF R$<"A" OR R$>"D" THEN 220
240 Q=ASC(R$)-65
250 PRINT@2,16;"Lecteur CIBLE (A à D)? «;:GET R$:PRINT R$
260 IF R$<"A" OR R$>"D" THEN 250
270 C=ASC(R$)-65
280 PRINT@2,18;"Nombre de SECTEURS par piste"
290 PRINT@2,19;"(1 à 19, 0 si détermination auto)";:INPUT S
300 IF S<0 OR S>19 THEN 290
310 PRINT@2,21;"No de PISTE initiale (0 si 1ère face,"
320 PRINT@2,22;"128 si 2ème, sinon autre valeur)";:INPUT P
330 IF P<0 OR P>227 THEN 320
340 CLS:PRINT:PRINT:PRINT"Repasser avec CTRL/A:" :PRINT:PRINT
350 REM mise à jour piste initiale
360 P$=HEX$(P):PRINT"POKE#78F, "P$":POKE#99D, "P$":PRINT:PRINT
370 PRINT"POKE#D0B, "P$;
380 REM lecteurs source et cible
390 Q$=HEX$(Q):PRINT":POKE#639, "Q$":PRINT:PRINT
400 PRINT"POKE#7BE, "Q$":POKE#77B, "HEX$(C):PRINT:PRINT
410 IF Q=C THEN PRINT"DOKE#7C7, #EAEA:DOKE#7EF, #EAEA":PRINT:PRINT:GOTO 440
420 PRINT"DOKE#7C5, #E54C:DOKE#7C7, #EA07":PRINT:PRINT
430 PRINT"DOKE#7ED, #084C:DOKE#7EF, #EA08":PRINT:PRINT
440 REM secteurs par piste
450 IF S=0 THEN 480
460 PRINT"DOKE#BBA, #A9EA:POKE#BBC, "HEX$(S):PRINT:PRINT
470 PRINT"DOKE#B7A, #A9EA:POKE#B7C, "HEX$(S) " : ";
480 PRINT"CALL#600":LOAD"COPFORM2",N

```

Le programme affiche d'abord une page d'information, Après avoir collecté les paramètres «lecteur source» dans «Q», «cible» dans «C», «secteurs par piste» dans «S» et «première piste» dans «P», le programme affiche les POKES à faire pour mettre ces valeurs aux bonnes adresses dans le programme, ainsi qu'un CALL#600 qui lancera le programme après la dernière série de CTRL/A. Enfin il charge COPFORM2 en mémoire, où il sera prêt à être édité par vos CTRL/A.

Je vous ferais grâce du désassemblage complet de Copform. A toute fin utile, voici toutefois la liste des modifications apportées pour passer de COPFORM à COPFORM2.

De 623 à 625, le JMP 0600 (si ni «C», ni «V», ni «S», relance du programme) a été remplacé par JMP 0DE5 (retour au «Ready», le programme n'avait pas de sortie).

De 629 à 62B, le JMP 075F a été remplacé par JMP 077A pour inhiber le message concernant le 2e lecteur (correction de bogue). La nouvelle entrée de la fonction «C» se fait en 75F.

De 6CF à 6EE, le message ambigu «DERNIERE PISTE A COPIER (HEXA) » a été remplacé par «NBRE DE PISTES A COPIER (HEXA) ».

De 789 à 78F, l'ancien code JSR 084E (routine inutile affichant un retour et 5 lignes vides), LDA 00 (bogue, il fallait #00), STA 54 (compteur d'erreurs) a été remplacé par NOP, LDA#00, STA 54, LDA#00 (ce dernier #00, situé en 78F recevra le n° de la 1e piste à copier).

De 792 à 793, LDA#CF a été remplacé par LDA#C8, modifie l'adresse du message «NBRE DE PISTES A COPIER (HEXA) » en supprimant les lignes vides qui le précède.

De 799 à 79b, le JSR 0AC3 (routine qui saisit le nombre de pistes à traiter) a été remplacé par JSR 0DBF (greffon qui corrige cette même routine).

De 80E à 811, l'ancien code LDA 54, BEQ 0815 (teste le nombre d'erreurs) a été remplacé par JSR 0DCA, NOP (greffon qui corrige une bogue concernant l'affichage du nombre d'erreurs).

De 815 à 817, le JMP 077E (reboucle au début de "C") a été remplacé par JMP 060A (reboucle au menu).

De 820 à 821, ADC#07 a été remplacé par ADC#08 (bogue dans la gestion du nombre de piste par cycle de copie).

De 8F4 à 8F5, CMP#09 a été remplacé par CMP#08 (bogue dans la gestion du nombre de piste par cycle de copie). En outre, la copie d'un bloc de 9 pistes formatée en 19 secteurs par piste se traduisait par un dépassement de la capacité de mémoire avec écrasement du jeu de caractères standard !

De 938 à 959, le message ambigu «DERNIERE PISTE A VERIFIER (HEXA) » a été remplacé par «NBRE DE PISTES A VERIFIER (HEXA) ».

De 960 à 96C, le message «PAS D'ERREUR» a été remplacé par «0 ERREUR(S) ». Dans la version d'origine, en absence d'erreur, le programme affichait «PAS D'ERREUR » puis «TEST TERMINE». Mais en cas d'erreurs il affichait seulement «TEST TERMINE» ! Maintenant il affiche soit «0 ERREUR(S) », soit le nombre d'erreurs suivi de la partie «ERREUR(S) » du message.

De 98B à 98D, le JSR 0AC3 (routine qui saisit le nombre de pistes à traiter) a été remplacé par JSR 0DBF (greffon qui corrige cette même routine).

De 9E0 à 9EA, l'ancien code LDA 54, BNE A4EA, LDA 5A, LDY 09, JSR CCB0 (qui teste le nombre d'erreur et s'il n'y en a pas affiche le message «PAS D'ERREUR») a été remplacé par JSR 0DCA et 8 NOPs (greffon qui corrige une bogue).

De 9F5 à 9F7, le JMP 097D (reboucle au début de "VB") a été remplacé par JMP 060A (reboucle au menu).

De AE5 à AE9, l'ancien code LDA#03, STA BB98 a été remplacé par LDA#03, STA BB94 pour repositionner correctement l'attribut INK jaune sur la ligne service, suite au changement de copyright.

De AF0 à B16, le message « © Jean-Pierre MARIE COPFORM V1.0 » a été remplacé par « © J.P.MARIE & A.CHERAMY COPFORM V2.2 ».

De B17 à B3D, le même message, codé en retirant #10 à chaque octet, a été mis à jour de même.

De C99 à CA4, l'ancien code JSR 0CB9, LDA 09FF, JSR E076, JSR E0CF a été remplacé par 3 NOPs, LDX 09FF, LDA #00, NOP, JSR E0C5 pour corriger une bogue dans l'affichage des nombres décimaux.

De CCD à CCF le JSR CCCE (début de la routine d'affichage d'une page d'information périmée, remplacée par l'équivalent dans CF2) a été remplacé par RTS et suite du code inchangé. La routine est donc neutralisée.

De CF7 à CF9, le JSR 0AC3 (routine qui saisit le nombre de pistes à traiter) a été remplacé par JSR 0DDF (greffon qui corrige cette même routine).

De D6B à D6D, le JMP 0CEA (reboucle au début de la routine «S») a été remplacé par JMP 060A (reboucle au menu).

De D71 à D7C, l'ancien code JSR 0CB9, LDA 09FF, JSR E076, JSR E0CF a été remplacé par 3 NOPs, LDX 09FF, LDA #00, NOP, JSR E0C5 pour corriger une bogue dans l'affichage des nombres décimaux.

De D90 à D9B, l'ancien code JSR 0CB9, LDA 09FE, JSR E076, JSR E0CF a été remplacé par 3 NOPs, LDX 09FE, LDA #00, NOP, JSR E0C5 pour corriger une bogue dans l'affichage des nombres décimaux.

De DBF à DC9, nouveau greffon pour déboguer la routine qui saisit le nombre de pistes à traiter : JSR 0AC3 (qui avait été écrasé par le JSR au greffon) LDA 07BF (n° de piste initiale, qui n'est pas toujours nul, afin de pouvoir traiter la 2e face ou un bloc de piste particulier) NOP NOP JMP0ACF (reprise du cours normal du programme). Ce greffon est appelé de 799, 98B et DDF.

De DCA à DDE, nouveau greffon pour déboguer la routine qui affiche le nombre d'erreurs : JSR 0844 (affiche un retour et 3 lignes vides) 3 NOPs LDX 54 (nombre d'erreurs, maxi 255) LDA#00 (octet de poids fort nul) NOP JSR E0C5 (affiche le nombre entier XA en décimal) LDA#62 LDY#09 (pointe le message d'adresse AY) JMP CCB0 (affiche «ERREUR(S)» et retourne. Ce greffon est appelé de 80E et 9E0.

De DDF à DE4, nouveau greffon pour déboguer la routine qui saisit le nombre de pistes à vérifier : JSR 0DBF (voir ci-dessus) DEC 57 (ajustement du nombre de pistes à vérifier) RTS. Ce greffon est appelé de CF7.

De DE5 à DF8, nouveau greffon pour retourner au «Ready» : JSR 04F2 (passe sous Ram overlay) LDA#78 STA D73C LDA#D1 STA D73D (restaure l'adresse D178 en Ram Overlay) JSR 04F2 (passe sous Rom) JSR 04C4 (retourne au «Ready») RTS (un JMP 04C4 aurait permis d'éviter ce RTS). Ce greffon est appelé de 0624.