

E) STRUCTURE DE LA "BITMAP" (SECTEUR N°#02 DE LA PISTE N°#14)

Afin de simplifier les explications, nous allons prendre comme exemple une disquette "Master", simple face, formatée en 42 pistes de 16 secteurs et sur laquelle aucun fichier n'a encore été écrit. Reporter vous à la figure 1 ci-jointe. Elle est formée de 8 blocs de 16 cases. Chaque case donne des renseignements sur un octet. Cette figure représente la première moitié du secteur N°#02 de la piste N°#14 (secteur N°2, piste N°20), qui comprend donc en tout 16 fois 16 octets soit 256 octets (comme tous les secteurs Sédoric).

La première ligne (octets N°#00 à N°#0F) donne des renseignements généraux sur l'organisation de la disquette. En voici le détail:

-- L'octet N°#00 vaut toujours #FF.

-- L'octet N°#01 vaut toujours #00.

-- Les octets N°#02 et N°#03 indiquent le nombre de secteurs libres (exemple 3A02 représente selon la convention courante #023A (inversion des octets de poids faible et de poids fort) soit 570 secteurs libres). On vérifie bien que 570 égale $42 \times 16 = 672$ secteurs moins 102 secteurs occupés ou réservés par Sédoric sur une disquette master. Une disquette analogue, mais formatée en double face, aurait 672 secteurs supplémentaires soit 1242 secteurs libres au total. Les octets N°#02 et N°#03 indiqueraient DA04 c'est à dire #04DA soit 1242.

-- Les octets N°#04 et N°#05 indiquent de la même façon le nombre de fichiers présents sur la disquette (ici 0, puisqu'aucun fichier n'a encore été écrit).

-- L'octet N°#06 indique le nombre total de pistes (ici #2A soit 42, rappel: ces pistes sont numérotées de N°#00 à N°#29 soit de N°0 à N°41).

-- L'octet N°#07 indique le nombre de secteurs par piste (ici #10 soit 16, rappel: ces secteurs sont numérotés de N°#01 à N°#10, soit de N°1 à N°16).

-- L'octet N°#08 qui est ici à #01 indique le nombre de secteurs utilisés pour le catalogue (directory). Chaque secteur de catalogue permet d'inscrire 15 fichiers. Par exemple, pour 40 fichiers présents sur une disquette, cet octet aurait pour valeur #03. Cela signifie que le nom du prochain fichier sera inscrit sur le 3ème secteur de catalogue (piste #14, secteur #0A pour les curieux).

-- L'octet N°#09 indique s'il s'agit d'une disquette SF ou DF. S'il s'agit d'une SF, sa valeur est identique à celle de l'octet N°#06 (ici #2A). S'il s'agit d'une disquette DF sa valeur est égale à celle de l'octet N°#06 augmentée de #80. Cet octet vaudrait #AA (#2A+#80) pour une disquette DF formatée en 42 pistes. Attention, l'opinion est généralement répandue que pour valider la DF, il faut écrire #AA dans cet octet. C'est faux si le nombre de piste par face est différent de 42!

-- L'octet N°#0A indique le type de disquette (#00 pour une disquette "master", #01 pour une disquette "slave").

-- Les octets N°#0B à N°#0F sont toujours à zéro (inutilisés).

Tous les octets suivants (N°#10 à N°#FF, soit du N°16 à N°255) constituent la bitmap proprement dit. Dans la figure 1, chaque case comporte 5 lignes:

-- La première ligne donne le numéro d'ordre de l'octet dans le secteur de bitmap (en hexadécimal).

-- La deuxième ligne donne le N° d'ordre (en hexadécimal) de la piste mappée. Puisque cette disquette est formatée à raison de 16 secteurs par piste, deux octets (soit 16 bits) suffisent pour indiquer si les 16 secteurs d'une piste sont libres (bit à 1) ou occupés (bit à 0). Ainsi les octets N°#10 et N°#11 représentent l'état de la piste N°#00. La dernière piste, portant le N°#29 est mappée par les octets N°#62 et N°#63. Lorsque la disquette est DF, le secteur suivant porte le N°#80 et est mappé par les octets N°#64 et N°#65 etc...

-- La troisième ligne fournit (en hexadécimal) les N° des secteurs représentés par cet octet (exemple: 8-1 indique que l'état des secteurs N°#01 à N°#08 peut être déduit de la valeur des bits de cet octet. Le bit N°0 qui se trouve le plus à droite représente le secteur N° #01, le bit N°7, qui se trouve le plus à gauche, représente le secteur N°#08. De manière analogue, 10-9 indique que le bit N°0 représente le secteur N°#09 et le bit N°7 représente le secteur N°#10. -- A la quatrième ligne, vous trouverez le contenu (en hexadécimal) des octets. C'est cette valeur qu'il faut lire, modifier et récrire.

-- La cinquième ligne indique la même valeur que la 4ème, mais en binaire.

Quelques exemples: Dans une disquette Sédoric "master", les 94 premiers secteurs sont occupés par Sédoric. Pour une disquette formatée en 16 secteurs par piste, cela fait 5 pistes pleines plus une piste incomplète, dont les deux derniers secteurs sont libres. Ces pistes sont représentées par les octets N°#10 à N°#1A qui sont à zéro, plus l'octet N°#1B qui est à #C0 (1100 0000 en binaire):

Octet N°#1B: le bit N°		7	6	5	4	3	2	1	0
(piste N°#05) représente le secteur	N°#10	0F	0E	0D	0C	0B	0A	09	
valeur binaire		1	1	0	0	0	0	0	0
valeur hexadécimale					C				0

Aucun fichier n'étant encore inscrit sur cette disquette, tous les autres secteurs (jusqu'au N°#29) sont libres, les bits correspondants sont à 1 et donc

les octets jusqu'au N°#63 sont à #FF. En fait tous les octets restants sont à #FF, même ceux qui sont situés après l'octet N°#63, d'où l'importance du nombre de secteurs libres qui indique au Sédoric si c'est réellement disponible. Tous les secteurs restants... sauf **huit secteurs situés à la piste N°#14 qui sont réservés** pour la BITMAP, le directory etc.. et qui perturbent le bel alignement des #FF en exhibant un #B0 et un #6D au milieu du tableau! Comme indiqué en annexe du manuel Sédoric, les secteurs N°#01, N°#02, N°#03, N°#04, N°#07, N°#0A, N°#0D et N°#10 sont réservés. Sédoric pourra y inscrire des informations, mais pas des fichiers. Les bits correspondants sont donc toujours à zéro:

Octet N°#38:	le bit N°	7	6	5	4	3	2	1	0
(Piste N°#14)	représente le secteur N°#	08	07	06	05	04	03	02	01
(début)	valeur binaire	1	0	1	1	0	0	0	0
	valeur hexadécimale			B				0	
Octet N°#39:	le bit N°	7	6	5	4	3	2	1	0
(Piste N°#14)	représente le secteur N°#	10	0F	0E	0D	0C	0B	0A	09
(fin)	valeur binaire	0	1	1	0	1	1	0	1
	valeur hexadécimale			6				D	

F) MAINTENANT ON NEUTRALISE!

De ce qui précède, on peut déduire qu'il est possible de neutraliser un secteur défectueux en mettant à zéro le bit qui le représente et en décrémentant le nombre de secteurs libres. **Ceci à conditions que ce secteur défectueux ne soit pas situé dans une zone stratégique incontournable pour Sédoric!** C'est à dire ni dans les 8 secteurs réservés de la piste N°#14 (en fait un peu moins, si le directory n'est pas trop important), ni dans les premiers secteurs de la disquette, situés à partir du secteur N°#01 de la piste N°#00 et qui sont au nombre de 94 pour une "master", de 17 pour une "gameinit" ou de 8 pour une "slave" (c'est le minimum).

La première chose à faire avec une disquette à problèmes est de sauvegarder tous ce qui peut l'être sur une bonne disquette. Nous n'aborderons pas aujourd'hui le sauvetage de programmes et nous supposons que vous êtes bien organisés et que vous avez une copie de sauvegarde de toutes vos "préciosités". Ensuite, il faut choisir entre deux méthodes: sans ou avec reformatage de la disquette.

Sans reformatage (si vous pensez qu'il n'y a qu'un ou deux secteurs défectueux ou si vous ne disposez pas d'un utilitaire de vérification de disquette tel que BDDISK, COPFORM etc...):

1) Notez **les coordonnées des secteur défectueux**, c'est à dire les N° de pistes et de secteurs affichés dans les messages d'erreurs de Sédoric.

2) Il faut aussi **savoir si ces secteurs défectueux se trouvent dans un fichier** (erreur à la lecture) ou **dans une zone libre de la disquette** (erreur à l'écriture). En cas de doute, vérifiez avec un utilitaire tel que SEDUTIL. ou en chargeant un à un les divers fichiers présents sur la disquette (notez celui ou ceux qui donnent un message d'erreur, ainsi que les coordonnées du ou des secteurs défectueux). Ensuite, effacez avec DEL le ou les fichier ayant donné une read fault.

3) **Mettez à zéro le bit correspondant à chaque secteur défectueux** pour le "réserver" (aidez-vous de la figure 1 ou 2 et des exemples ci-après).

4) **Réduisez en conséquence le nombre de secteurs libres** (octets N°#02 et N°#03).

Avec reformatage (solution plus simple et plus efficace, mais nécessitant un utilitaire de vérification de disquette):

1) Reformatez votre disquette en **16 secteurs par piste** pour augmenter la fiabilité (par exemple: INIT A,16,42,S ou INIT A,16,42,D), éventuellement en "slave" ou en "gameinit" si les problèmes se situent dans les toutes premières pistes (N°#00 à N°#05).

2) Recherchez les **secteurs défectueux**. Notez leurs coordonnées. Il est indispensable de disposer d'un utilitaire de vérification de disquette. Il en existe plusieurs tels que BDDISK, COPFORM etc...

3) **Mettez à zéro le bit correspondant** à chaque secteur défectueux pour le "réserver" (aidez-vous de la figure 1 ou 2 et des exemples ci-après).

4) Réduisez en conséquence le **nombre de secteurs libres** (octets N°#02 et N°#03).

a) Exemple d'une disquette formatée en 16 secteurs par piste.

Supposons que votre disquette (toujours celle de la figure 1) ait un secteur défectueux: le N°#06 de la piste N°#08: vous ne parvenez pas à sauvegarder votre dernière petite merveille "TOTO.COM" (write fault). Sauver d'abord "TOTO.COM" sur une autre disquette, puis neutralisez le secteur en cause et recopiez "TOTO.COM" sur votre disquette qui a retrouvé une nouvelle vie. Pour neutraliser un secteur (dans notre exemple, le 6ème de la piste N°8), il faut lire la valeur hexadécimale de l'octet indiquant l'état du secteur en question (dans notre cas, c'est l'octet N°#20), à l'aide d'un éditeur de secteurs. Puis il faut décomposer cette valeur hexadécimale en binaire, mettre le bit N°5 (c'est à dire le 6ème en partant de la droite) à zéro, sans modifier les autres, retransformer en hexadécimal et récrire cette nouvelle valeur là où vous l'aviez lue.

Par exemple, avec BDDISK:

```
-- Choisir l'option 0A du menu principal.
-- Répondre 14 pour le N° de piste et 02 pour le N° de secteur.
-- Taper V pour avoir l'affichage des valeurs en hexadécimal.
-- Regarder celle de l'octet N°#20 (1er octet de la 3ème ligne). Il vaut #FF
soit en binaire: 1111 1111.
-- Mettre à zéro le bit N°5, ce qui donne 1101 1111 soit #DF.
-- Taper M pour modifier,
-- Avec les flèches allez sur la case de l'octet N°#20.
-- Tapez DF puis <ENTER>
-- Ce n'est pas tout, il faut décrémente le nombre de secteurs libres. Pour
ce faire, lire la valeur hexadécimale sur deux octets au début de la BITMAP
(octets N°#02 et N°#03). N'oubliez pas de les inverser: par exemple #3A en
premier et #02 en second signifie #023A.
-- On retire 1, ce qui dans notre cas, revient à écrire #39 dans le premier
des deux octets. Pour ce faire:
-- Taper M pour modifier,
-- Avec les flèches allez sur la case de l'octet N°#02.
-- Tapez 39 puis <ENTER>
-- Vérifiez le tout en retapant V pour avoir un réaffichage (à l'usage vous
verrez qu'il y a un bug mineur dans BDDISK).
-- Puis tapez E pour récrire la BITMAP modifiée sur la disquette.
```

Tout cela peut paraître barbare, mais essayez à l'aide de la figure 1 et vous verrez que **la réalité est plus simple que les explications**. Si vous disposez d'un éditeur de secteurs autre que BDDISK, vous n'aurez aucun mal à vous y retrouver, même si, comme c'est le cas par exemple avec SEDUTIL, le nombre d'octets affichés par ligne d'écran est différent de 16.

b) Et si la disquette est formatée en 17 secteurs par piste?

La méthode est la même. Mais il faut s'aider de la figure 2 car c'est un peu plus compliqué de faire la correspondance entre secteur défectueux et bit à modifier. En effet, pour représenter une piste, il faut deux octets plus un bit ($16+1 = 17$). Ce qui introduit un décalage dans le tableau de 1 bit, puis de 2 bits pour la 2ème piste etc... Au bout de 8 pistes, le décalage est de un octet complet. Au bout de 20 pistes (piste N°19), le décalage est de $20/8 = 2$ octets complets plus 4 bits. Il devient difficile de savoir à quel endroit commence la représentation de la piste N°20 (N°#14 en hexadécimal) et où se trouve le secteur N°2 de cette piste N°#14, c'est à dire la bitmap. La figure 2 est là pour vous y aider: Vous pouvez voir que la bitmap n'est plus représentée par le bit N°1 (le deuxième en partant de la droite) de l'octet N°#38, comme précédemment, mais par le bit N°5 (le 6ème en partant de la droite) de l'octet N°#3A. Pour les mordus: cela représente bien un décalage de 2 octets complets et 4 bits. Ouf!

Voyons un **exemple de neutralisation**. Cette fois-ci c'est le secteur N°#0D de la piste N°#26 qui pose problème! Chargeons la bitmap de notre disquette: c'est toujours le secteur N°#2 de la piste N°#14. La figure 2 indique que le secteur N°#0D de la piste N°#26 est représenté par le bit N°2 de l'octet N°#62. Lisons la valeur de cet octet, soit #FE par exemple ce qui en binaire fait 1111 1110. Si l'on met à 0 le bit correspondant au secteur incriminé, on obtient 1111 1010.

Ce qui en hexadécimal fait #FA et que nous récrivons là où nous avons lu.

Avec BD-DISK:

```
-- Choisir l'option 0A du menu principal.
-- Répondre 14 pour le N° de piste et 02 pour le N° de secteur.
-- Taper V pour avoir l'affichage des valeurs en hexadécimal.
-- Regarder celle de l'octet N°#62 (3ème octet de la 7ème ligne). Il vaut #FE
soit en binaire: 1111 1110.
-- Mettre à zéro le bit N°2, ce qui donne 1111 1010 soit #FA.
-- Taper M pour modifier,
-- Avec les flèches allez sur la case de l'octet N°#62.
-- Tapez FA puis <ENTER>
-- Ce n'est pas tout, il faut décrémenter le nombre de secteurs libres. Pour
ce faire, lire la valeur hexadécimale sur deux octets au début de la BITMAP
(octets N°#02 et N°#03). N'oubliez pas de les inverser: par exemple #1F en
premier et #02 en second signifie #021F.
-- On retire 1, ce qui dans notre cas, revient à écrire #1E dans le premier
des deux octets. Pour ce faire:
-- Taper M pour modifier,
-- Avec les flèches allez sur la case de l'octet N°#02.
-- Tapez 1E puis <ENTER>
-- Vérifiez le tout en retapant V pour avoir un réaffichage.
-- Puis tapez E pour récrire la BITMAP modifiée sur la disquette.
```

G) EN GUISE DE CONCLUSION.

Ceux qui auront eu la patience de lire jusqu'au bout sont des "bidouilleurs" nés. Je crois qu'un des grands attrait de l'Oric est d'être une machine ouverte. Beaucoup de ceux qui continuent à se passionner pour cette machine, le font par plaisir de comprendre comment ça marche. A dire vrai, s'il est utile de neutraliser un secteur défectueux qui vous gâche une disquette, il est aussi très important de s'amuser. Je ne doute pas que vous serez très fiers de "récupérer" une ou plusieurs disquettes. Si vous avez besoin d'aide, n'hésitez pas à m'écrire:

André Chéramy, 54 rue de Sours, 28000 CHARTRES.

