

Metro Cross (3)

par Simon Guyart & André Chéramy

Eh oui, il y a encore un peu à dire sur cette simple animation. Pour conclure cette série initiée par André, nous vous proposons aujourd'hui un retour sur les motivations qui m'ont amené à faire ce style d'animation, avec les avantages, les inconvénients et les perspectives.

LE POURQUOI

Pourquoi choisir une animation en mode TEXT ? La raison en est simple... En réfléchissant à la façon de convertir un jeu Atari sur Oric, j'ai dressé la liste des problèmes que je pourrais rencontrer avec mon niveau de connaissance en programmation sur Oric. Il se trouve que j'avais besoin de dessiner des fenêtres en mode HIRES de manière très rapide, donc si possible à partir d'un fichier sur une disquette. Le but est d'afficher un dessin rectangulaire à l'écran, sans effacer le graphisme en fond, et particulièrement sur la gauche et la droite de la fenêtre. Le problème est que si l'on sauve le dessin de la fenêtre en mode HIRES, la mémoire vidéo de l'Oric étant sauvée séquentiellement, on doit sauver aussi les zones à gauche et à droite. Ces zones effaceront donc le dessin de fond lorsqu'on rechargera la "fenêtre".

Plusieurs solutions s'offraient alors à moi, parmi lesquelles:

- sauver la fenêtre ligne HIRES par ligne HIRES, et la recharger de même, cela m'évitait d'effacer les bords, mais c'est sans doute lent et cela crée une flopée de fichiers !
- sauver ligne par ligne la fenêtre mais dans un seul fichier, et recharger la fenêtre dans une zone mémoire autre que HIRES, et réafficher (avec l'instruction MOVE) ligne par ligne en HIRES par des swaps. En fait, c'est la même méthode que précédemment mais on travaille en mémoire au lieu de travailler sur le disque directement. Mais là encore, c'est lent.

Certains vont rigoler et me dire que je pourrais me mettre à l'assembleur et faire une petite routine rapide pour afficher ma fenêtre. J'y ai pensé mais je n'ai jamais eu le temps de m'y mettre :)

C'est là que m'est venue l'idée d'utiliser les caractères. En effet, un caractère est en soi une fenêtre de 6 pixels sur 8, redéfinissable (donc "redessinable"). Il suffirait d'afficher côte à côte des caractères pour former un rectangle, de les redéfinir tous qu'un coup (car la table de redéfinition des caractères peut être sauvée/rechargée dans un seul fichier sans retraitement), et ma fenêtre apparaît ! Reste à tester la vitesse d'affichage après en HIRES (avec l'instruction CHAR).

Et là encore j'ai laissé le projet en plan faute de temps. Petit à petit a germé l'idée de faire une animation en mode TEXT basée sur ce principe, avec des swaps mémoire. J'ai cherché un moment le sujet... Je pensais à convertir des GIF animés, puis je suis revenu à un jeu Atari, car pour moi l'Atari ST a un potentiel de jeux adaptables sur Oric avec des programmeurs de talent (c'est un appel du pied ;)

LE COMMENT

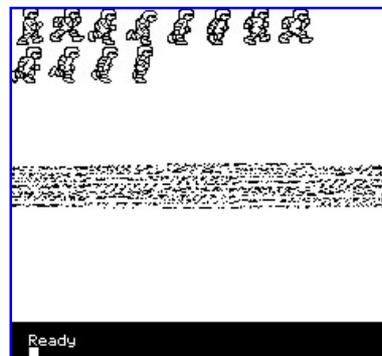
Voilà donc le pourquoi. Pour le comment, je vous renvoie au premier article de cette série écrit il y a quelques mois par André. J'ajouterai simplement un détail concernant le passage d'une image simple à un fichier de redéfinition de caractères.

En effet, passer d'une image HIRES à la même image en caractères TEXT redéfinis impose un retraitement: en HIRES, 8 octets consécutifs correspondent à une ligne de 48 pixels (6 pixels affichables par octets), alors que dans la table de redéfinition des caractères, 8 octets consécutifs correspondent à un caractère, à savoir un carré de 6 pixels sur 8 et non plus une ligne !

J'ai donc pris mes images à animer sur PC, je les ai soigneusement collées les unes à la suite des autres à intervalles réguliers sur une seule image que j'ai ensuite convertie en écran HIRES avec PCHIRES. Puis j'ai écrit un petit programme de conversion, qui prend les octets de l'écran HIRES et les réorganise comme il faut. Voici le listing commenté, je ne vais pas rentrer dans le détail car cette réorganisa-

tion dépend de l'image de départ et des caractères que l'on veut utiliser. Petit rappel : chaque image ici fait 24*24 pixels, soit 4 caractères sur 3.

```
10 HIRES
15 A=#AF80 'adresse de stockage, en HIRES pour voir le résultat
20 CLOAD"IMAGE" 'on charge l'image
25 FOR L = 0 TO 7 '7 images du bonhomme sur la 1ere ligne de l'image HIRES
30 FOR J = 0 TO 2 '3 caractères verticalement
40 FOR I = 0 TO 3 '4 caractères horizontalement
50 FOR K = 0 TO 7 '8 octets par caractère à redéfinir
60 MOVE 40960+I+J*40*8+K*40+L*4,40960+I+J*40*8+K*40+L*4,A 'déplacem. d'un octet
70 A=A+1 'on passe à l'octet suivant
80 NEXT K
90 NEXT I
100 NEXT J
110 NEXT L
130 FOR L = 0 TO 3 'même chose que précédemment pour les 3 bonshommes
140 FOR J = 3 TO 5 'sur la 2e ligne de l'image HIRES.
150 FOR I = 0 TO 3
160 FOR K = 0 TO 7
170 MOVE 40960+I+J*40*8+K*40+L*4,40960+I+J*40*8+K*40+L*4,A
180 A=A+1
190 NEXT K
200 NEXT I
210 NEXT J
220 NEXT L
```



La figure vous montre l'image d'origine, avec en dessous le résultat de la conversion. Ca n'a à priori plus rien à voir, mais déplacez cette "conversion" dans la zone de redéfinition de caractères et vous verrez que le résultat est l'image HIRES en caractères redéfinis !

LES POSSIBILITES

Les utilisations intéressantes d'un tel procédé me semblent multiples. Tout d'abord, comme l'a dit André, en mode TEXT, on peut animer énormément de choses rapidement. On peut même imaginer un scrolling (assez lent peut-être) très long à partir d'une lecture de fichiers de redéfinitions de caractères sur disquette, un peu comme certains jeux sur Atari ou Amiga il y a quelques années, qui chargeaient les décors depuis la disquette au fur et à mesure que le scrolling avançait.

La commande CHAR permet donc aussi d'utiliser la redéfinition de caractères en mode HIRES. Attention toutefois, si les caractères affichés sur l'écran en mode TEXT sont redéfinis dynamiquement (l'écran TEXT change dès qu'on redéfinit le caractère), ceux affichés en mode HIRES par la commande CHAR restent tels qu'ils ont été affichés à l'origine ! Cela permet deux exploitations différentes et ne manque pas d'intérêt. L'exemple précédent du scrolling sera par exemple plus simple à effectuer en mode HIRES car on n'aura pas à se soucier de ce qui avait été redéfini et affiché précédemment.

J'ai même imaginé une méthode de compression basée sur ce principe. Prenons un dessin en HIRES. Ce dessin occupe 8Ko. Or il y a sans doute dedans des zones qui se répètent (par exemple une grande surface noire). On pourrait prendre des zones de 6*8 pixels (taille d'un caractère), redéfinir un caractère avec, et voir si un caractère précédemment redéfini n'est pas identique. S'il l'est, inutile de stocker cette information 2 fois ! On découpe donc ainsi l'image HIRES (240*200 pixels) en un tableau de 40*25 caractères (soit 1Ko) plus les redéfinitions associées. On est évidemment limité par le nombre de caractères redéfinissables, et cette méthode ne pourrait pas s'appliquer à toutes les images, mais elle devrait permettre un réaffichage très rapide en mode TEXT (vous vous souvenez du chat de Loriciels ?); sans doute moins en mode HIRES du fait de la lenteur de la commande CHAR.

En bref, il y a encore des tas de choses à creuser !