

MANUEL PROVISOIRE

Char-EDITOR V.0.0.7
X86-WIN32 TOOL FOR

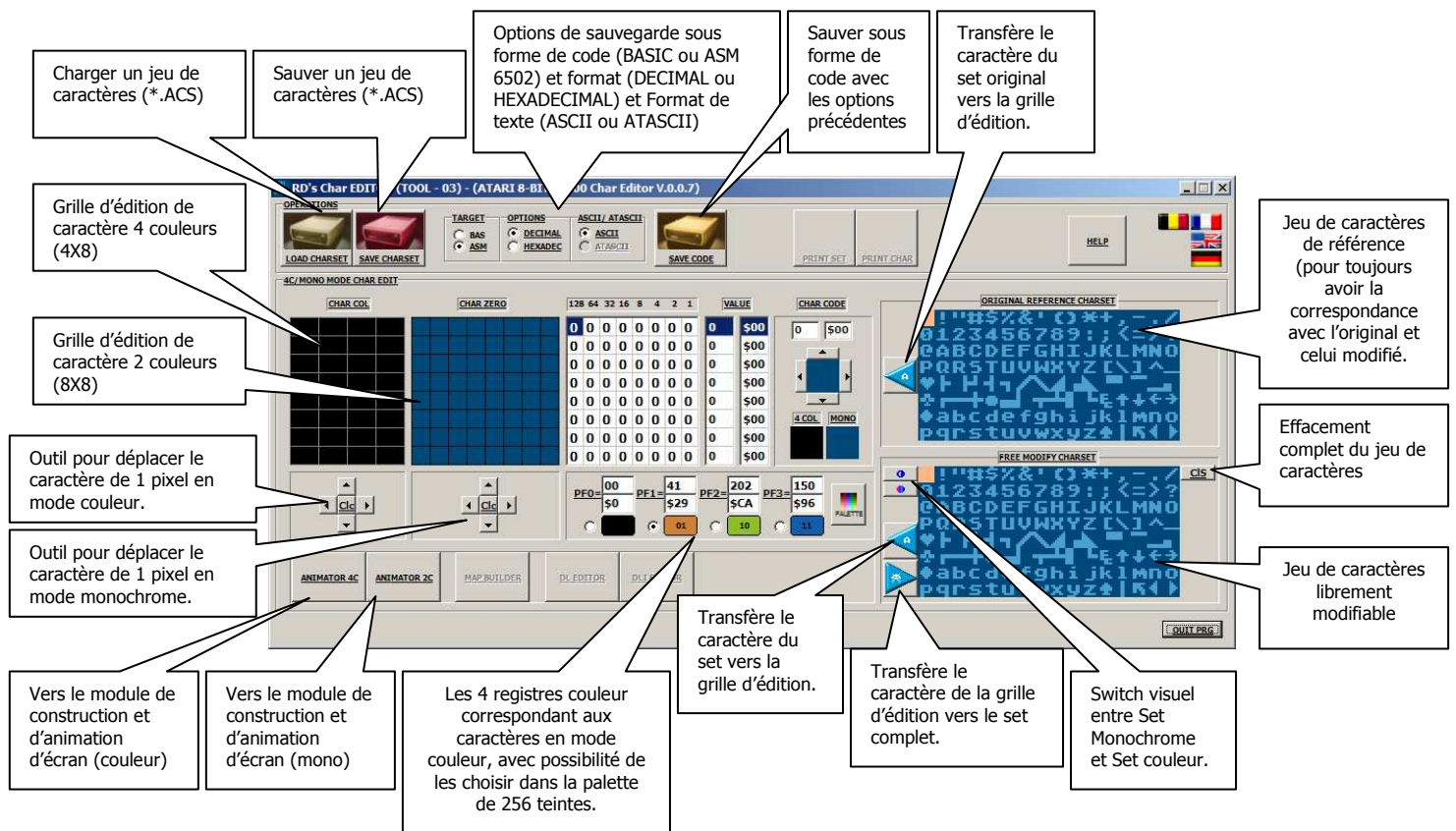
THE
ATARI 800

1. Créer un jeu de caractères

1.1. Les affichages

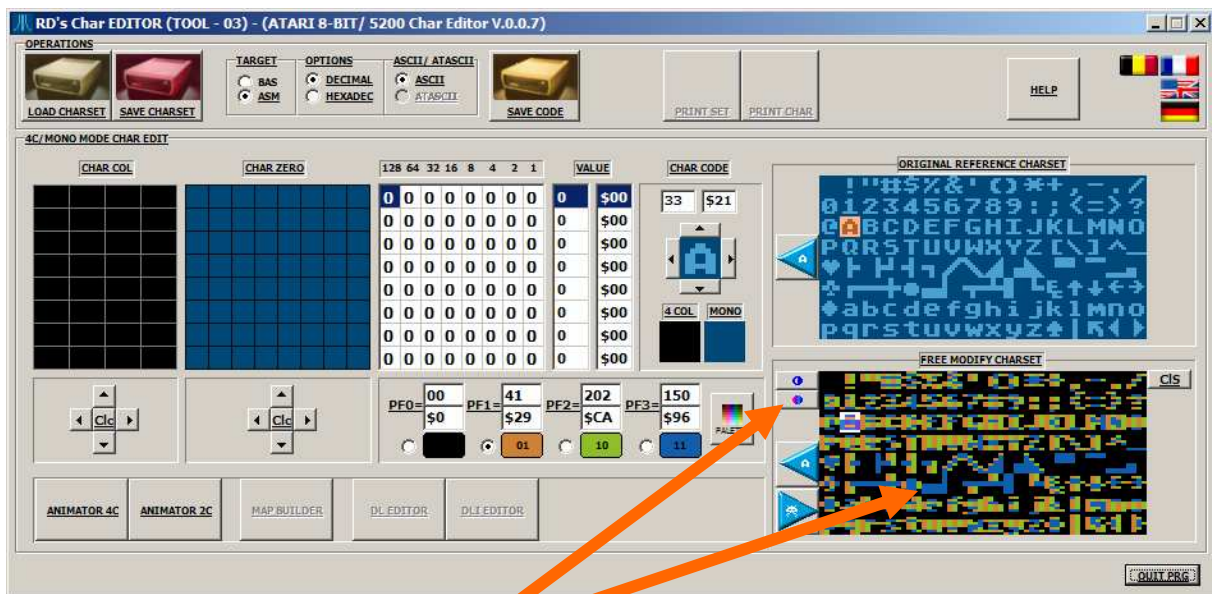
L'écran principal de l'application est morcelé et la zone centrale est celle qui va permettre d'éditer un jeu de caractère de 128 éléments différents.

Sur l'ATARI 8-BIT, le dernier bit (7) ne fait qu'indiquer de mettre en video inverse les 128 premiers caractères et ne sert donc pas beaucoup. Donc nous ne comptons que sur 128 caractères.

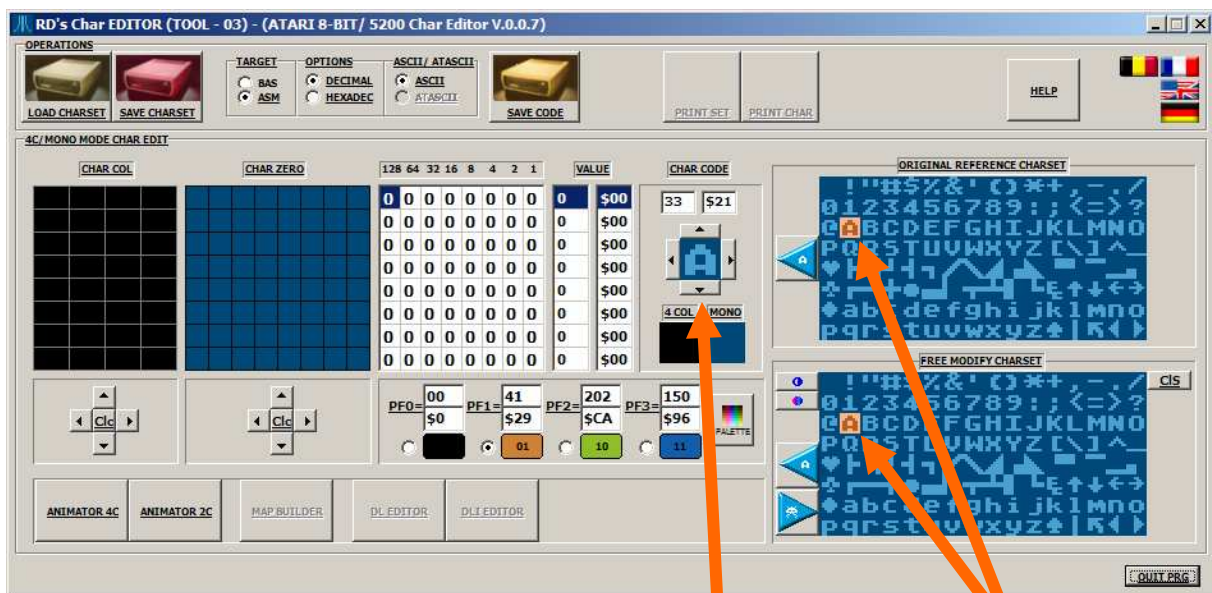


1.2. Modifier le jeu de caractères original.

1.2.1. Première étape : Choisir le mode couleur.

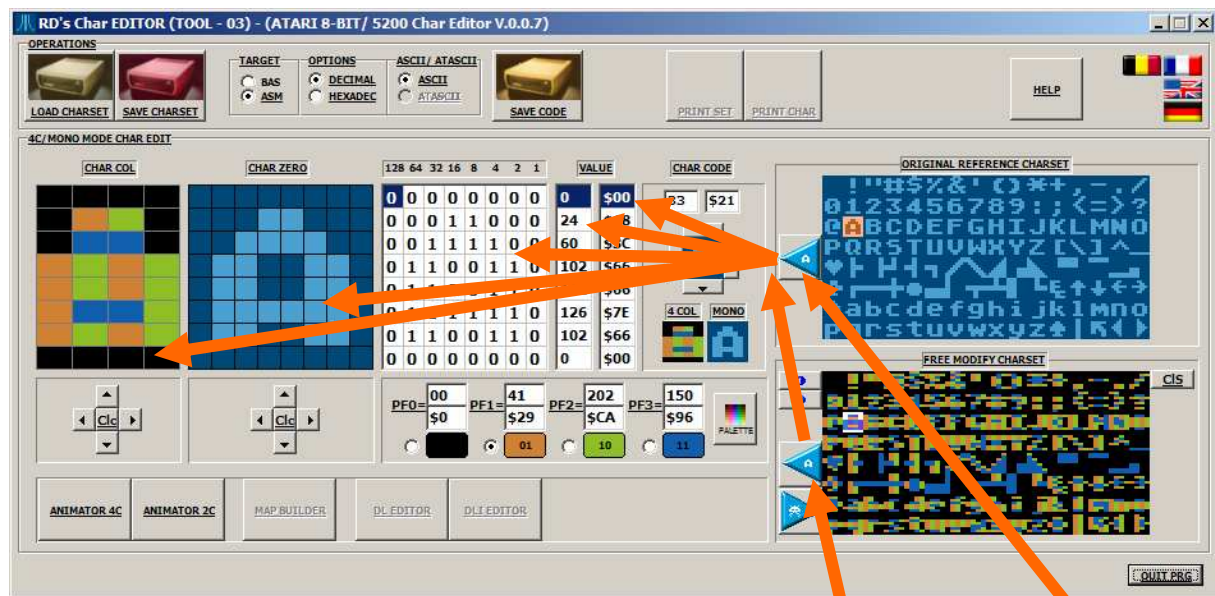


1.2.2. Deuxième étape : choisir un caractère à transformer parmi les 128 disponibles.



2 > On remarque que « A » se retrouve tout de suite dans une zone où l'on peut « voir » son code ATASCII en décimal et hexadécimal, le caractère lui-même, un peu « zoomé »

1.2.3. Troisième étape : Transférer le caractère pour l'éditer.

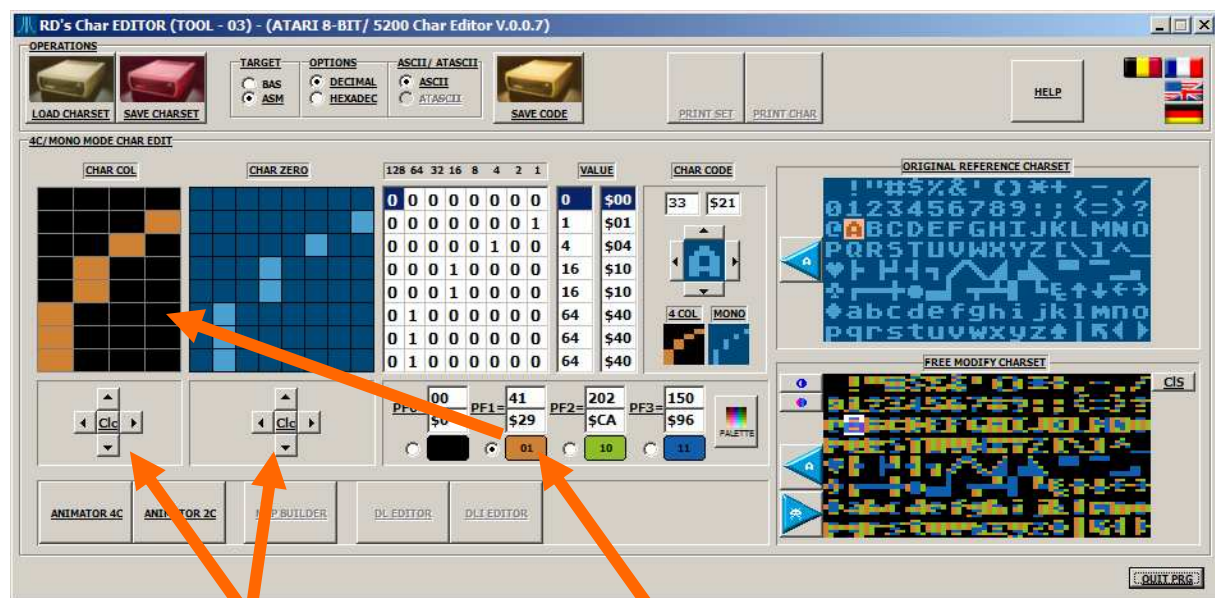


1 > Soit on transfère un caractère original vers la grille d'édition (Caractère contenu dans la ROM de l'ATARI 8-BIT.

2 > Soit on transfère un caractère qui peut être soit un original ou bien un caractère déjà transformé.

3/ A ce stade, le caractère sélectionné « A » se retrouve dans la grille d'édition, aussi bien couleur que monochrome. On visualise également la grille binaire-décimale-hexadécimale correspondante pour chaque ligne.

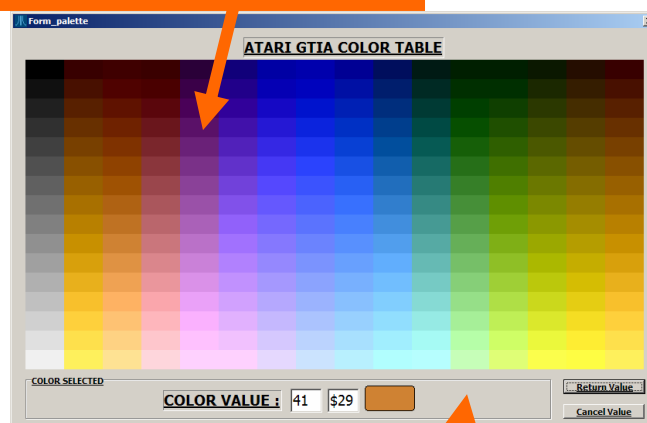
1.2.4. Quatrième étape : Editer le caractère choisi



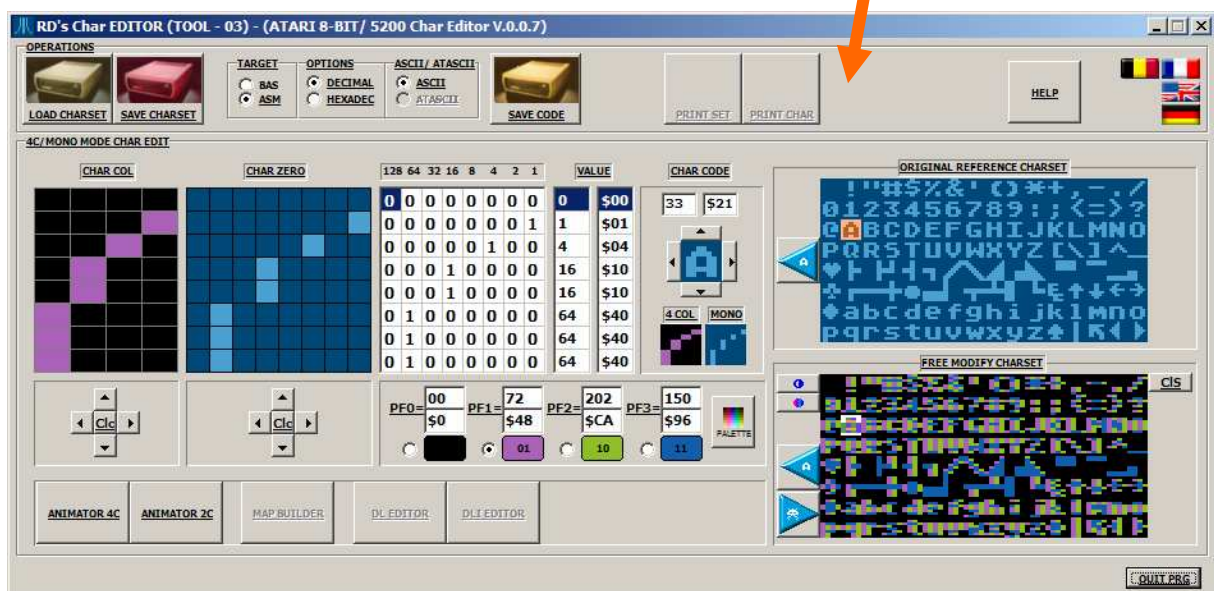
Quelques petits outils sont disponibles, comme le scrolling multidirectionnel de 1 pixel sur le caractère et l'effacement du caractère complet dans la grille d'édition.

En sélectionnant une des 4 couleurs disponibles, on peut dessiner point par point sur la grille couleur/ monochrome. Attention, comme un pixel couleur est codé sur 2 bit, le motif monochrome montre un affichage inégal, ce qui est normal si on utilise la couleur !!!

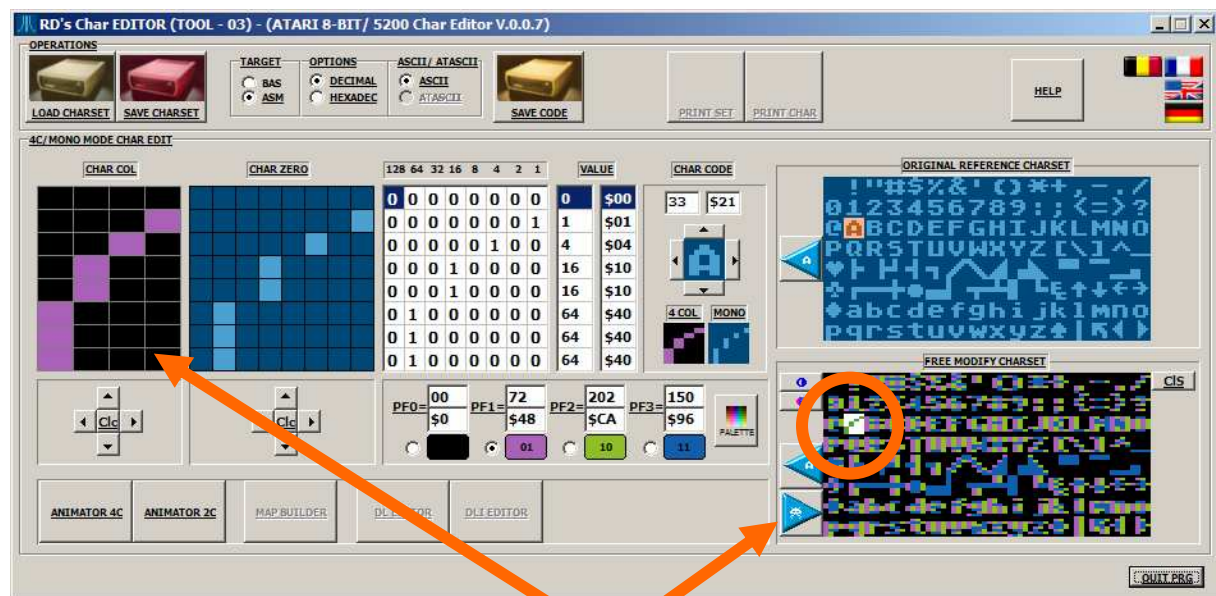
1.2.5. Comment changer une des 4 couleurs ?



Changement de la couleur PF1 en violet



1.2.6. Replacer le caractère modifié dans le Jeu



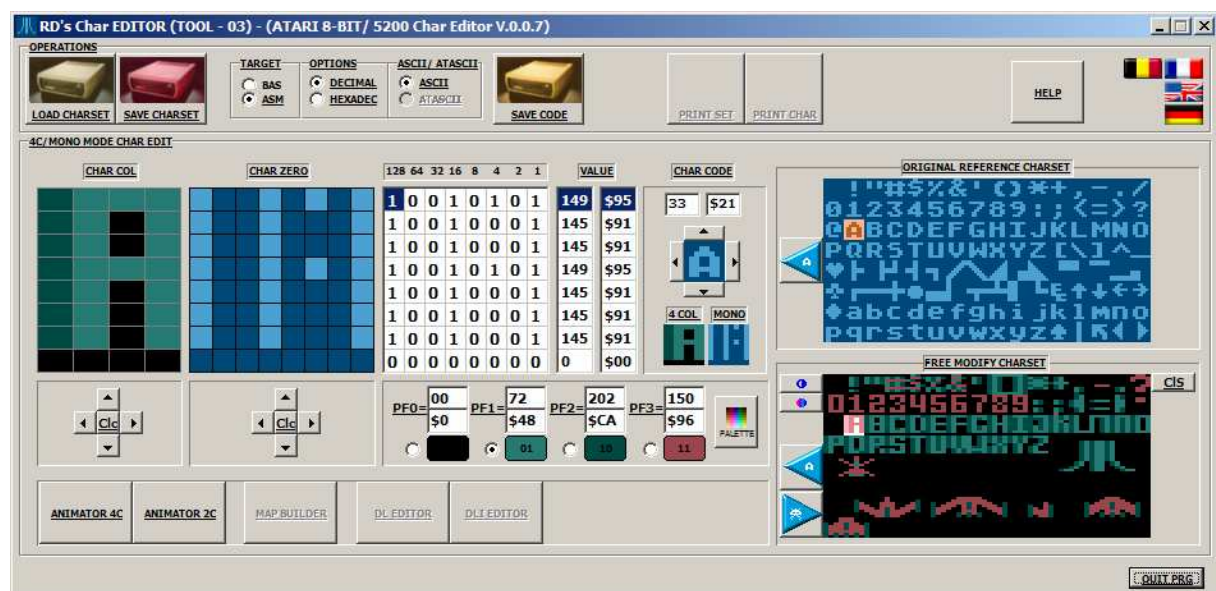
Une fois que le caractère voulu est redéfini (édition terminée) dans la grille d'édition, il faut le replacer dans le jeu de caractère à l'aide de l'autre petite icône flèche bleue. On observe alors que le jeu change. Répéter l'opération pour tous les caractères nécessaires.

1.2.7. Exemple d'un set redéfini :

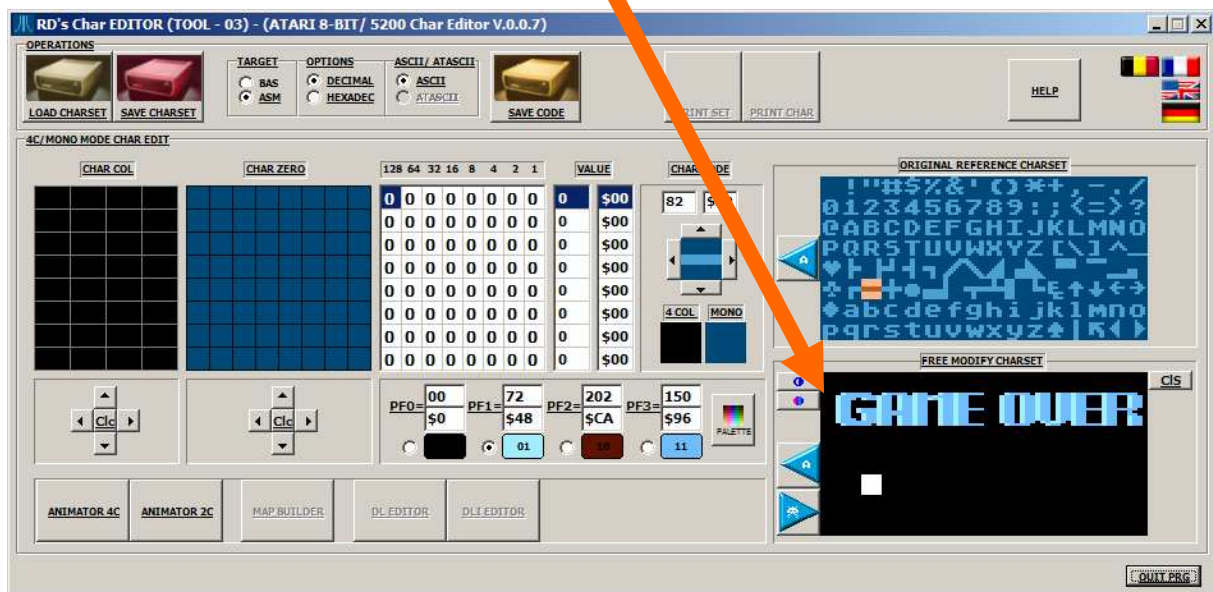
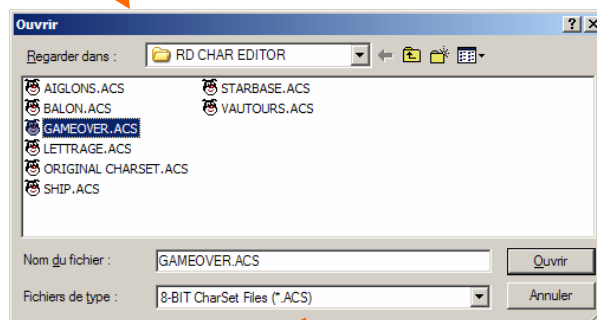
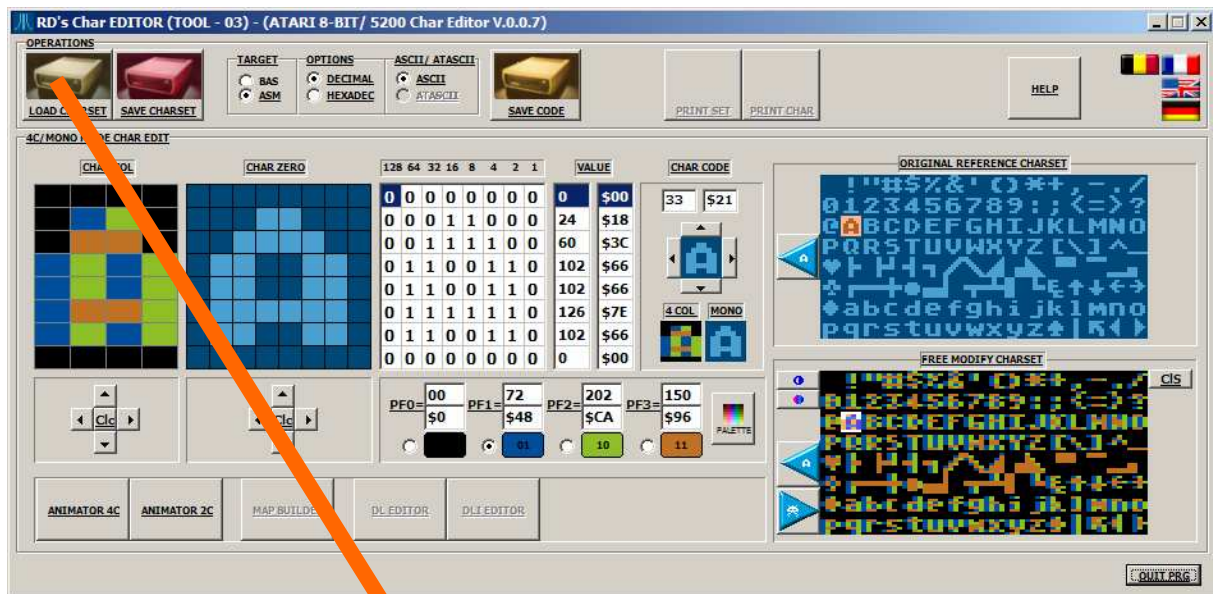
Ici nous avons l'exemple d'un set complet redéfini avec pas mal de « blancs » inclus.

N'oublions pas que l'ATARI 8-Bit peut générer une interruption qui permet au final d'avoir quasiment un jeu de caractères par ligne d'écran ainsi que des couleurs différentes à chaque fois.

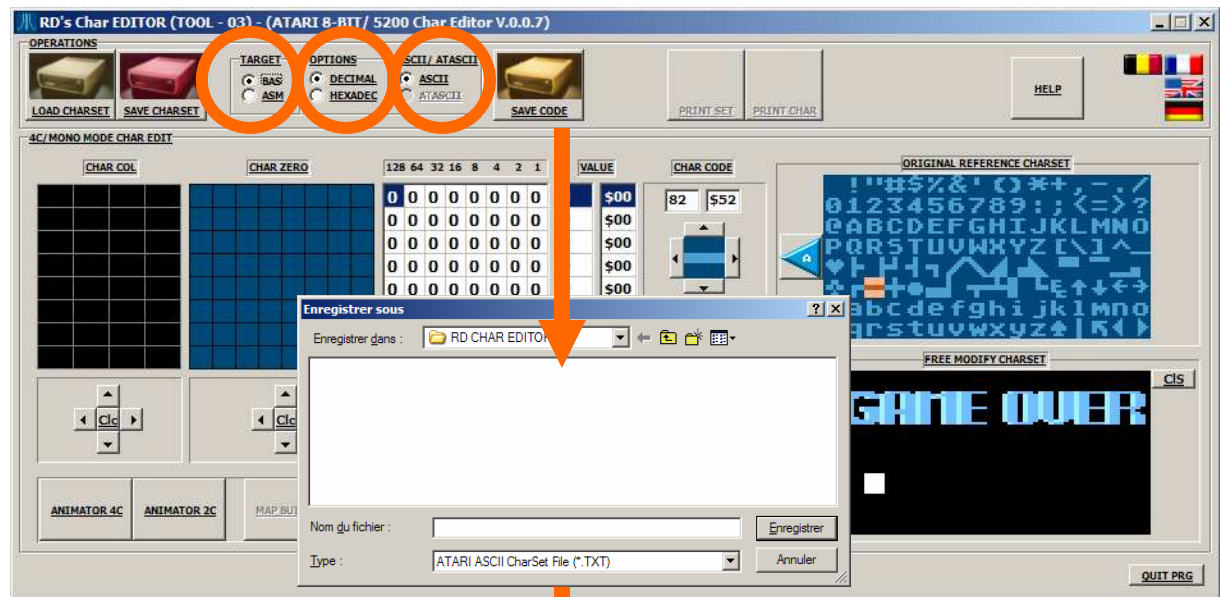
Un simple pointeur à changer et une Display List à adapter.



1.2.8. Exemple d'un Chargement/ Sauvegarde :



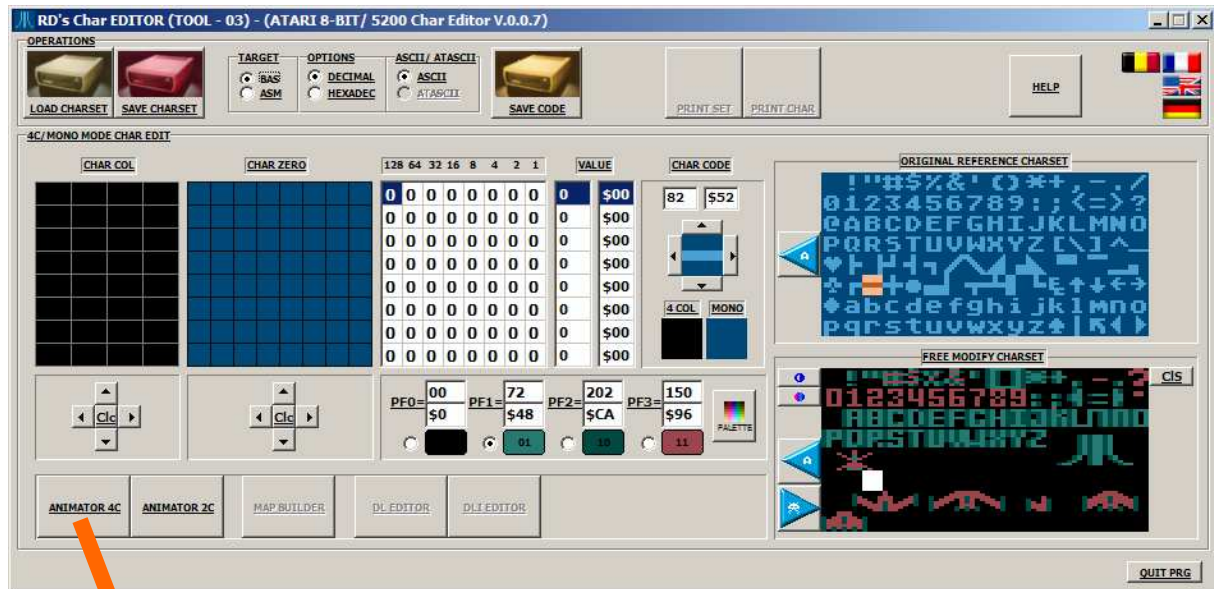
1.2.9. Exemple d'une Sauvegarde-CODE :



1010 REM	10350 DATA	11120 DATA
1020 REM X:\XL-XE\RD PHOENIX PROJECT\RD CHAR	10360 DATA	11130 DATA
EDITOR\GAMEOVER.TXT.LST	10370 DATA	11140 DATA
1030 REM	10380 DATA	11150 DATA
1040 REM *****	10390 DATA	11160 DATA
1050 REM * REDEFINITION DE 128 CARACTERES *	10400 DATA	11170 DATA
1060 REM *****	10410 DATA	11180 DATA
1070 REM	10420 DATA	11190 DATA
1080 REM *****	10430 DATA	11200 DATA
1090 REM RESERVATION DE 1Kb (4 PAGES DE 256 BYTES)	10440 DATA	11210 DATA
1100 REM *****	10450 DATA	11220 DATA
1110 REM	10460 DATA	11230 DATA
1120 TOP=PEEK(106)-4	10470 DATA	11240 DATA
1130 POKE 106,TOP	10480 DATA	11250 DATA
1140 REM	10490 DATA	11260 DATA
1150 REM *****	10500 DATA	11270 DATA
1160 REM RECOPIE DU SET EN ROM VERS LA RAM	10510 DATA	
1170 REM *****	10520 DATA	
1180 REM	10530 DATA	
1190 ROM=PEEK(756)*256	10540 DATA	
1200 RAM=TOP*256	10550 DATA	
1210 REM	10560 DATA	
1220 FOR I=0 TO 1023	10570 DATA	
1230 POKE RAM+I,PEEK(ROM+I)	10580 DATA	
1240 NEXT I	10590 DATA	
1250 REM	10600 DATA	
1260 REM *****	10610 DATA	
1270 REM COPIE DU NOUVEAU SET VERS LA RAM	10620 DATA	
1280 REM *****	10630 DATA	
1290 REM	10640 DATA	
1300 ST=(TOP*256)	10650 DATA	
1310 FOR I=0 TO 1023	10660 DATA	
1320 READ A	10670 DATA	
1330 POKE ST+I,A	10680 DATA	
1340 NEXT I	10690 DATA	
1350 REM	10700 DATA	
1360 REM *****	10710 DATA	
1370 REM ACTIVATION DU NOUVEAU SET	10720 DATA	
1380 REM *****	10730 DATA	
1390 REM	10740 DATA	
1400 POKE 756,TOP	10750 DATA	
1410 REM	10760 DATA	
10000 DATA 000,000,000,000,000,000,000,000,000	10770 DATA	
10010 DATA 000,000,000,000,000,000,000,000,000	10780 DATA	
10020 DATA 000,000,000,000,000,000,000,000,000	10790 DATA	
10030 DATA 000,000,000,000,000,000,000,000,000	10800 DATA	
10040 DATA 000,000,000,000,000,000,000,000,000	10810 DATA	
10050 DATA 000,000,000,000,000,000,000,000,000	10820 DATA	
10060 DATA 000,000,000,000,000,000,000,000,000	10830 DATA	
10070 DATA 000,000,000,000,000,000,000,000,000	10840 DATA	
10080 DATA 000,000,000,000,000,000,000,000,000	10850 DATA	
10090 DATA 000,000,000,000,000,000,000,000,000	10860 DATA	
10100 DATA 000,000,000,000,000,000,000,000,000	10870 DATA	
10110 DATA 000,000,000,000,000,000,000,000,000	10880 DATA	
10120 DATA 000,000,000,000,000,000,000,000,000	10890 DATA	
10130 DATA 000,000,000,000,000,000,000,000,000	10900 DATA	
10140 DATA 000,000,000,000,000,000,000,000,000	10910 DATA	
10150 DATA 000,000,000,000,000,000,000,000,000	10920 DATA	
10160 DATA 000,001,001,001,001,001,001,001,001	10930 DATA	
10170 DATA 127,255,252,240,240,240,241,241	10940 DATA	
10180 DATA 240,253,125,001,001,001,253,253	10950 DATA	
10190 DATA 127,241,241,241,241,241,241,255	10960 DATA	
10200 DATA 199,247,247,247,247,247,247,247	10970 DATA	
10210 DATA 193,247,221,193,193,193,193,193	10980 DATA	
10220 DATA 247,247,247,247,247,247,247,247	10990 DATA	
10230 DATA 255,192,192,192,192,192,255,192	11000 DATA	
10240 DATA 000,001,001,001,001,001,001,001	11010 DATA	
10250 DATA 127,241,241,241,241,241,241,241	11020 DATA	
10260 DATA 199,247,247,247,247,247,247,247	11030 DATA	
10270 DATA 199,199,199,199,199,199,199,199	11040 DATA	
10280 DATA 223,223,223,223,223,223,223,223	11050 DATA	
10290 DATA 253,001,001,001,001,001,253,001	11060 DATA	
10300 DATA 255,240,240,240,240,240,255,240	11070 DATA	
10310 DATA 240,124,124,124,124,124,240,192	11080 DATA	
10320 DATA 001,001,001,001,000,000,000,000	11090 DATA	
10330 DATA 240,240,252,255,127,000,000,000	11100 DATA	
10340 DATA 125,125,125,253,249,000,000,000	11110 DATA	

2. Construire un écran avec un jeu de caractères

2.1. Le constructeur d'écrans



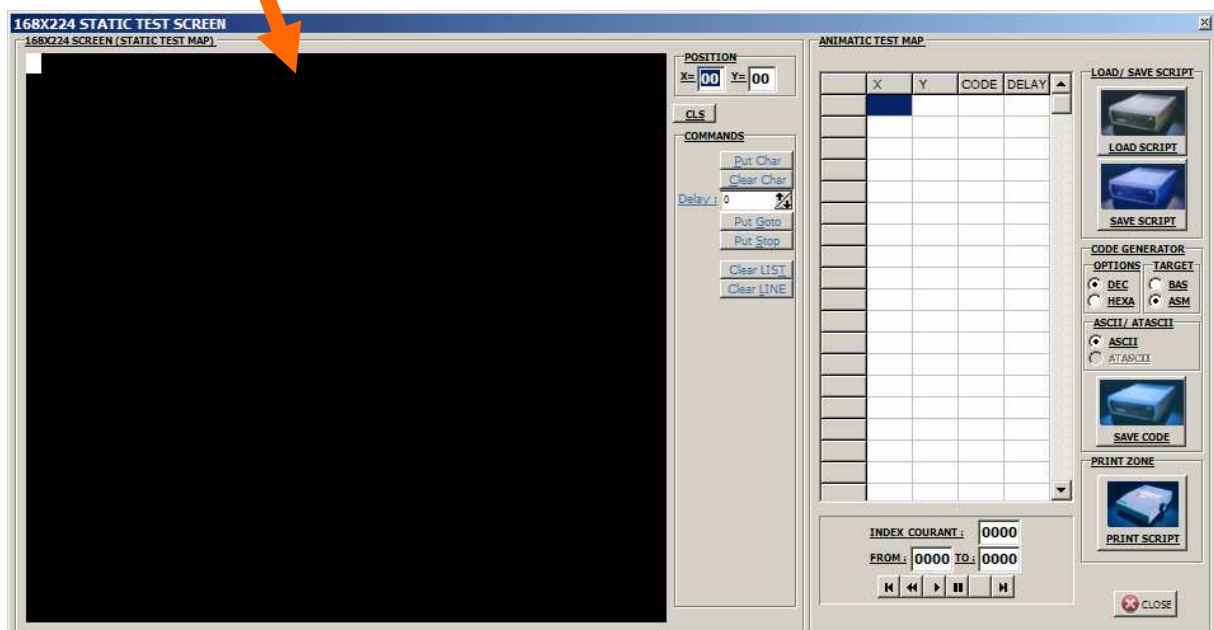
Le constructeur/ Animateur d'écrans est constitué d'une zone d'affichage représentant un écran en mode texte 4 couleurs de l'ATARI 800.

Que peut-on y faire ?

Avec le jeu de caractères de l'écran principal, on peut placer les caractères que l'on sélectionne à la souris.

Divers renseignements sont accessibles comme la position en X,Y. Une grille d'édition permet de voir à quelle position on a placé tel caractère et la possibilité de placer un délai d'attente avant l'affichage suivant.

Au delà de la construction simple, un micro langage est là pour effectuer des animations automatiquement avec 3 instructions : la 1ère est le placement d'un caractère en X,Y, de code C et délai D (en millisecondes), la 2^e est une instruction GOTO XXXX où XXXX est un n° de ligne ou il faut boucler et enfin STOP qui indique la fin de la séquence.



2.2. Exemple de construction

