

Clavier virtuel pour joystick

par André C.



Etat de la question

La plupart des utilisateurs de Super-Oric sont dépourvus de clavier et utilisent uniquement un joystick. C'est aussi bien le cas de ceux qui ont un Super-Oric réel, que de ceux qui se servent d'un émulateur de console Snes. Dans ce dernier cas le clavier du PC est uniquement utilisé pour émuler un joystick. Cependant, de temps en temps, il serait intéressant de pouvoir entrer quelques caractères, par exemple des mots pour un jeu d'aventure (PREND GOURDE), ou même des chiffres pour un jeu

Le listing

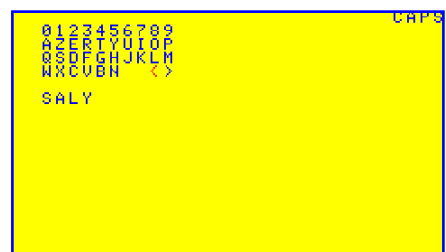
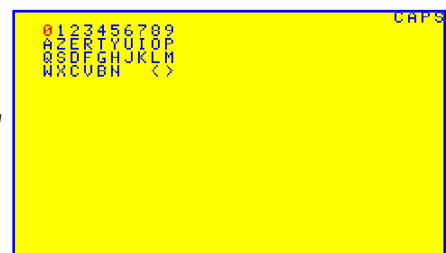
```
10 CLS:POKE#26A,2:GOSUB 910
20 DIM CV(11,4):M$=""
30 Z$=""
40 FOR Y=1 TO 4
50 FOR X=2 TO 11
60 READ CV(X,Y):PLOT X,Y, CV(X,Y)
70 NEXT: NEXT
80 X=2:Y=1:GOSUB 510
90 J=0:WAIT 33:REPEAT:J=USR(0):UNTIL J<>0
100 IF J=32 THEN GOSUB 610:' Select
110 IF J>8 THEN 90:' Reponse non valide
120 IF SCRNX(X,Y)>#400 THEN GOSUB 510
130 IF J=1 THEN X=X+1
140 IF X>11 THEN X=11
150 IF J=2 THEN X=X-1
160 IF X<2 THEN X=2
170 IF J=4 THEN Y=Y+1
180 IF Y>4 THEN Y=4
190 IF J=8 THEN Y=Y-1
200 IF Y<1 THEN Y=1
```

mathématique etc. La routine proposée aujourd'hui sera facile à insérer dans votre application. Comme vous le constaterez, le clavier proposé est simpliste (réduit au strict minimum). Mais il est facile d'ajouter des touches (#, \$, etc.). Dans son état actuel, par exemple, il y a deux touches "espace", il suffit d'éditer l'un des deux codes Ascii 32 pour le remplacer par le code de votre choix. Si cela ne suffit pas, il sera aisé de rajouter des lignes de caractères à votre clavier virtuel et/ou d'allonger ces lignes. Les ajustements à faire sont alors minimes.

A priori, l'idée d'avoir à utiliser un clavier virtuel peut sembler pénible. C'était en tous cas mon sentiment. Après essai, ce clavier est vraiment facile à utiliser et pas casse-pieds du tout. Bon, d'accord, il ne faudrait pas avoir à rédiger une thèse... Mais pour quelques mots ou même phrases, c'est tout à fait viable.

Mon cahier des charges

Dans la version proposée, je me suis limité aux 26 caractères de l'alphabet, plus les 10 chiffres, plus "espace", DEL (pour corriger les erreurs de frappe) et ENTER pour valider le texte tapé. Soit 26+10+3=39. Clin d'œil à la barre d'espace, j'ai ajouté une deuxième touche d'espace pour arriver à 40 touches. J'ai évidemment disposé ces 40 touches en 4 lignes de 10 touches, dans l'ordre d'un clavier AZERTY, celui que nous connaissons le mieux.



```

210 GOSUB 510
220 GOTO 90
500 ` Bascule couleur du caractere en X,Y
510 C=SCRN(X,Y)
520 C=(#400 AND NOT C) OR (NOT #400 AND C):'XOR
530 PLOT X,Y,C:RETURN
600 ` Saisie caractere
610 C=SCRN(X,Y)
620 C=(#400 AND NOT C) OR (NOT #400 AND C):'XOR
630 IF C=62 THEN END:' > = RETURN
640 IF C=60 THEN 710:' < = DEL
650 C$=CHR$(C):M$=M$+C$
660 PLOT 2,6,M$:RETURN
700 ` DEL
710 M$=LEFT$(M$,LEN(M$)-1)
720 PLOT 2,6,Z$:GOTO 660
900 ` Définition des couleurs
910 DEF INK 0,#03FF,#7C00:' Pal #0000, jne, ble
920 DEF INK 4,#03FF,#001F:' Pal #0400, jne, rge
930 RETURN
1000 ` Data clavier virtuel
1010 DATA 48,49,50,51,52,53,54,55,56,57
1020 DATA 65,90,69,82,84,89,85,73,79,80
1030 DATA 81,83,68,70,71,72,74,75,76,77
1040 DATA 87,88,67,86,66,78,32,32,60,62

```

Commentaires sur ce listing

Ligne 10: Ecran vide, pas de curseur, définition des palettes de couleurs.

Lignes 20-30: Tableau CV (Clavier Virtuel) pour recevoir les codes Ascii des 40 touches. Ce tableau est déjà structuré selon la matrice à afficher. Initialisation des variables M\$ (qui cumulera le texte tapé) et Z\$ (qui effacera le texte précédent, lors d'une correction). Attention, dans la version actuelle, si le texte dépasse 30 caractères, il débordera de la ligne et seule la première ligne sera effacée en cas de correction. Notez que j'ai laissé en l'état, car il est peu vraisemblable d'avoir à taper de longues phrases. En cas de besoin, il sera aisé d'améliorer ce point.

Lignes 40-70: Lecture des codes Ascii dans les DATA et affichage des caractères correspondants selon une matrice de clavier: 4 lignes de 10 touches

Ligne 80: Coordonnées de la position initiale du curseur. Le curseur n'apparaît pas en vidéo inverse mais par basculement de la couleur du caractère du bleu au rouge (sous-programme 510).

Ligne 90: Attente qu'une touche du joypad soit enfoncée.

Ligne 100: La touche Select a été pressée (validation d'un caractère). Appel au sous-programme 610, qui lit et traite ce caractère.

Ligne 110: Rebouclage car la réponse est non valide:

Outre la touche Select déjà traitée, ne sont possibles que les flèches, dont le code est toujours compris entre 1 et 8.

Lignes 120: Ce qui va suivre est forcément un déplacement du curseur et donc la touche actuelle (rouge) va repasser au bleu, tandis que la touche ciblée va passer au rouge. On commence par basculer la touche actuelle du rouge au bleu en appelant le sous-programme 510.

Lignes 130-140: Flèche droite (code joypad 1): X est incrémenté sans pouvoir dépasser 11.

Lignes 150-160: Flèche gauche (code joypad 2): X est décrémenté sans pouvoir être inférieur à 2.

Lignes 170-180: Flèche basse (code joypad 4): Y est incrémenté sans pouvoir dépasser 4.

Lignes 190-200: Flèche haute (code joypad 8): Y est décrémenté sans pouvoir être inférieur à 1.

Ligne 210: Appel au sous-programme qui bascule la couleur du caractère au curseur.

Ligne 220: Reboucle pour une nouvelle saisie.

Ligne 500-530: Sous-programme qui bascule la couleur du caractère au curseur. Un XOR #400, (formé à l'aide d'une combinaison de AND, OR et NOT) permet d'inverser le bit n°10 du code lut avec SCRN. Rappel: Le n° de palette (de 0 à 7) est codé sur les bits n° 10 à 12. Ici on bascule entre la palette n°0 (jaune, bleu) et la palette n°1 (jaune, rouge) et réciproquement. Le code modifié est alors

re-PLOTé à la même place.

Lignes 600-660: Sous-programme de lecture et traitement du caractère sélectionné. Ce caractère est lu avec SCRIN. Comme il est en rouge, son bit n°10 est à un. On effectue une inversion de ce bit avec un XOR #400 et on obtient alors son code Ascii.

- S'il s'agit du code Ascii 62 (caractère ">"), utilisé ici pour simuler un ENTER (validation de la chaîne de caractères saisie). Ici donc, un END est exécuté (fin de la routine "Clavier Virtuel"). Pour intégrer cette routine à une application de votre cru, ce END sera remplacé par un RETURN.

- S'il s'agit du code Ascii 60 (caractère "<"), utilisé ici pour simuler un DEL, un sous-programme est appelé en 710.

- Dans tous les autres cas, le caractère est ajouté à la chaîne M\$, qui est affichée un peu plus bas dans l'écran. Pour intégrer la routine "Clavier Virtuel" à une application de votre cru, ce simple affichage sera adapté à vos besoins.

Lignes 700-720: Ce sous-programme "Touche DEL" permet de corriger les fautes de frappe. La chaîne M\$ précédemment enregistrée est raccourcie d'un caractère puis ré-affichée, après avoir nettoyé la ligne de l'écran. Comme je l'ai déjà indiqué, si votre chaîne de caractère dépasse 30 caractères, elle sera affichée sur plusieurs lignes et il faudra donc prévoir d'effacer ces lignes avant de ré-afficher la chaîne corrigée.

Lignes 900-930: Redéfinition de couleurs réduite au minimum. Seules les couleurs n°0 et n°1 des palettes n°0 et n°1 sont redéfinies (respectivement jaune, bleu, jaune et rouge). Le code PLOTé dans l'écran correspond au code Ascii + #0000 pour afficher le caractère avec les couleurs de la palette n°0 et au code Ascii + #0400 pour afficher le caractère avec les couleurs de la palette n°1. Il va de soit que vous pouvez adapter cet exemple à votre application, par exemple utiliser d'autres couleurs ou d'autres palettes. Si vous basculez entre deux autres palettes, le ou les bits à inverser seront différents. Rappel: le n° de palette (de 0 à 7) est codé sur les bits n°10, 11 et 12.

Lignes 1000-1040: Les 40 codes Ascii des touches du clavier virtuel, disposés en 4 lignes de 10 codes, afin qu'il vous soit plus aisé de les éditer: Comme je l'ai déjà indiqué, vous pouvez aussi augmenter la longueur ou le nombre des lignes de touches de votre clavier virtuel. Mais il faudra corriger les variables X et Y en fonction de ces modifications.

Conclusion

Voilà donc votre Super-Oric réel ou virtuel (émulateur Snes) pourvu d'un clavier virtuel, piloté par le joystick. Cela ne remplace pas un vrai clavier pour les longues saisies, mais peut ajouter un sérieux plus aux applications que vous souhaitez développer. Bon amusement donc avec votre Super-Oric!

