

Les sons préprogrammés de l'Oric

1e partie : Récupération et adaptation du code source

par André C.

Les 3 articles précédents "Musique, Note, Octave, Fréquence et Période", "Les enveloppes du PSG 8912" et "Temporisation en langage machine" vous ont fourni quelques éléments de compréhension des sons de l'Oric. Nous allons voir comment exploiter au mieux les sons préprogrammés grâce à une nouvelle série de 3 articles. Afin d'éviter les redites, j'ai zappé les bases déjà décrites et vous invite à vous reporter aux articles précédents en cas de besoin.

Liste des sons préprogrammés

Aux commandes usuelles **PING**, **SHOOT**, **EXPLODE** et **ZAP**, il faut ajouter **TOUCON** (bruit clavier des touches contrôles), **TOUNOR** (idem touches normales) et **OUPS**, le petit dernier du Telestrat.

Tous les sons préprogrammés ont le même inconvénient : il n'est pas possible d'en moduler le volume sonore (d'où divers petits gadgets hardware apparus pour atténuer le son pendant la nuit !). La raison en est simple : Tous ces sons sont figés en Rom et on peut difficilement intervenir.

Il est quand même possible de s'inspirer de leur code pour le mettre en Ram et le modifier. Pour OUPS par exemple, le volume sonore est figé plein pot, mais il est facile de rendre ce paramètre ajustable dans une version en Ram.

Malheureusement, pour les sons préprogrammés utilisant une enveloppe, on est bien obligé de garder celle-ci, sous peine de faire autre chose. Il est toutefois possible de modifier la hauteur de la note, sa durée etc.

Structure des sons préprogrammés

Tous ces sons sont basés sur 2 routines en Rom :

- La routine #F590 envoie un seul des 14 paramètres au PSG (les registres A et Y du 6502 contenant respectivement le paramètre à envoyer et le numéro du registre cible).
- La routine #FA86 envoie 14 paramètres au PSG (les registres X et Y du 6502 contenant l'adresse où se trouvent les 14 paramètres).

En outre ZAP et OUPS font appel à une tempori-

sation à boucle pour gérer la durée du son produit.

Récupération des sources en langage machine

Le plus simple est donc de recopier le code présent en Rom dans un fichier source en langage machine. Il faut aussi modifier les adresses des blocs de paramètres pour les diriger vers la Ram. Enfin, dans le cas d'OUPS, il faut remplacer les adresses Telestrat par les adresses Atmos pour les 2 routines citées ci-dessus.

C'est ainsi que j'ai obtenus les 8 fichiers source suivants : PINGLM.ASM, SHOOTLM.ASM, EXPLODLM.ASM, TOUCONLM.ASM, TOUNORLM.ASM, ZAPLM.ASM, OUPSLM.ASM et même PREDEFM.ASM qui regroupe le code des 7 sons préprogrammés (voir le fichier SonsPreprogr1.zip qui accompagne cet article).

Mise en œuvre des sons préprogrammés

Après compilation, j'ai obtenu 8 fichiers exécutable .TAP qui sans surprise reproduisent fidèlement les 7 sons d'origine. Il ne reste plus qu'à étudier le mécanisme de chaque son et éventuellement à modifier certains paramètres.

Les commandes PING, SHOOT, EXPLODE, TOUCON et TOUNOR utilisent une enveloppe pour contrôler le volume sonore alors que ZAP et OUPS fixent celui-ci au maximum. Le code de tous ces sons envoie un bloc de 14 paramètres au PSG 8912. Mais en outre, le code de ZAP et OUPS met en place des boucles de temporisation pour contrôler le son produit.

L'analyse du bloc de 14 paramètres est facile car tous les paramètres sont à zéro par défaut (sauf R7 qui contrôle les canaux en service et qui est mis à #3F pour tout inhiber). Voir éventuellement "Retour sur la mise en œuvre du PSG AY3-8912" dans le CEO-mag de Janvier 2020. Les paramètres spécifiques d'un son donné sont donc immédiatement apparents ce qui permet de comprendre ce que fait ce son. Attention toutefois au paramètre R13 dont la valeur #00 peut désigner l'enveloppe de n°0 lorsque le contrôle du volume sonore fait appel à une enveloppe.

A suivre...

