Bogue **CSAVE**"NF",A#adr_deb,E#adr_fin

Lorsqu'on dispose d'un lecteur de disquette, ce qui est infiniment plus confortable qu'un lecteur de K7, on a tendance à laisser tomber complètement ce dernier. C'est la raison pour laquelle la bogue CSAVE qui existe depuis plus de 10 ans est passée largement inaperçue. A l'époque où l'utilitaire CAP10.EXE (devenu entre temps CAP11.EXE) n'existait pas, j'ai été amené a convertir des fichiers Oric -> PC pour Euphoric et la seule possibilité a été de sauver chaque fichier au format "cassette" à l'aide de CSAVE, puis d'en ôter l'entête à l'aide d'un éditeur hexadécimal sur PC.

Quelle ne fut pas ma stupéfaction de voir que l'adresse de début n'était jamais l'adresse spécifiée, mais celle de début de programme BASIC (#0501) ! J'ai tourné la question dans tous les sens avant de me résoudre à conclure (à tort, vous allez le voir) que le bel émulateur Euphoric était bogué !

ENQUÊTE...

A l'occasion d'un E-mail, j'ai demandé à Laurent si cette bogue d'Euphoric était déjà répertoriée, voire corrigée. Laurent a transmis à Fabrice Francès qui a répondu laconiquement :
<Ce n'est pas une bogue d'Euphoric, mais de Sédoric>.
Re-stupéfaction : comment Sédoric peut-il boguer une routine de la ROM ?

Quelques jours plus tard, nouvel E-mail de Fabrice (transmis par Laurent) : <<...J'avais trouvé le bug et Jon le connaissait mais ne savait pas la cause : ça vient d'une variable page 0 qui est utilisée par Sédoric et vient interférer avec son utilisation dans la routine CSAVE (elle permet de conserver si le paramètre adresse lu est associé à la lettre A ou E)>>.

Bizarre, bizarre. Je me mets aussitôt en chasse et découvre que Fabrice à bel et bien raison. De #E7B2 à #E852, une routine de la ROM analyse la syntaxe de CLOAD et de CSAVE. En #E82A, cette routine sauve le code de début/fin A ou E dans la variable #0E. En #E83B, elle récupère ce code en relisant la variable #0E. Parfait, mais entre ces deux événements, elle a fait appel à la routine XCRGET (incrémenter TXTPTR et lire un caractère). Cette routine XCRGET, située en page 0 en #E2, était très simple à l'origine (Atmos sans Sédoric). Mais elle a été prolongée pour Sédoric, par une routine en #0400. Et c'est là que se situe le problème, car ce greffon sauve les registres A et X du processeur 6502 dans les variables #0E et #0F. Le code de début/fin A ou E se trouve alors écrasé.

EN ROUTE POUR UNE NOUVELLE AVENTURE...

Comment corriger cette bogue. Dans le principe c'est simple : puisqu'il est quasi impossible de modifier la ROM, il faut sauver le registre A dans une autre variable que la variable #0E. La mise en pratique est plus délicate. En effet, pratiquement toutes les adresses de la page 0 sont utilisées à un moment ou à un autre par la ROM ou par Sédoric. Ayant désassemblé Sédoric, je connais les adresses utilisées par Sédoric (voir "Sédoric à nu"). En ce qui concerne la ROM1.1 j'ai consulté "L'Oric à nu" de Fabrice Broche. Je suis donc parti à la recherche d'adresses utilisées ni par Sédoric ni par la ROM1.I . Je n'en ai trouvé que 2 : #BB et #C1. #BB étant plus amusant que #C1, j'ai opté pour #BB (mais comme vous allez le voir c'était un vilain bébé).

MISE EN PRATIQUE...

J'ai suivi la procédure habituelle : Je boote avec les disquettes DO_IT_1 et DO_IT_2, je tape HIMEM#13FF5 NOYAU5 B-MONAC15 (le moniteur d'André Chénière) et je remplace par des #BB les #0E situés en #160E, #163A, #170E et #173A (ces adresses dans NOYAU correspondent à #C60E, #C63A, #C70E et #C73A dans la RAM overlay). Ces 4 occurrences de #0E correspondent à une paire STA #0E / LDA #0E dans la page 4 destinée à la ROM1.0 (en page #C6) et à la ROM1.1 (en page #C7). Je sauve le NOYAU modifié : SAVE"B-NOYAU",A#1400,E#4FFF5 et le remets en place : B-MISEAJOUR5 Je reboote et je teste. Hourra, le CSAVE fonctionne correctement maintenant ! Les autres commandes courantes BASIC et Sédoric semblent également fonctionner correctement.

DECONVENUE...

Pas si simple... Dès le lendemain, j'ai un problème avec GET$. Après quelques recherches, je trouve que GET$ utilise une routine de READ, laquelle sauve TXTPTR en #BA et #BB et utilise aussi XCRGOT... Un coup pour rien, je n'ai guère été plus chanceux que Fabrice Broche !

C'EST EN FORGEANT...

Cette fois j'ai cherché dans toute la ROM1.1 si la variable #C1 était utilisée (recherche des suites hexadécimales 84C1, 85C1, 86C1, 94C1, 95C1, 96C1, A4C1, A5C1, A6C1, B4C1, B5C1 et B6C1 correspondant à tous les LDA LDX etc...). Bonne nouvelle, #C1 semble réellement inutilisé par la ROM1.1 Je reprends la procédure décrite ci-dessus et change les 4 #BB par des #C1.

CONCLUSION...

Cette fois semble la bonne. Je vous ai décrit l'histoire de ce débogage pour illustrer comment se produisent les interférences lorsque l'on effectue des modifications. Il est très difficile de tenir à jour une liste de toutes les variables utilisées et de tous les entrelacs de sous-programmes.

André Chéramy 54 rue de Sours 28000 CHARTRES