

DUPK7

par André C.

```

dupk7.com
Permet de charger tout fichier en
RAM quelque soit son format exotique!

1) Allumez l'Oric avec une disquette
Master, sans commande dans INIST.
2) Faire HIMEM#7FF QUIT !DUPK7 puis
POKE#080F, #HH et POKE#0813, #LL
#HHLL étant le nombre d'octets à lire
4) CALL#900 (K7 position lecture)
5) Sauvegarde globale (provisoire)
(RESET) et !SAVE"TOUT",A#1000,E#BFFF
6) Examen de ce qui a été chargé
avec moniteur: !MONAC1 puis CALL#AC00
puis D1000. Déterminez l'adresse où
commence actuellement le fichier et
celle où il finit. Déterminez les
adresses de début et fin réelles en
position finale normale de travail.
Faire ESC RETURN pour sortir.
7) Sauvegarde finale (plus propre):
!MOUE#deb,act,#fin act,#deb réel puis
!SAVE"nom",A#deb réel,E#fin réelle.
  
```

Présentation :

DUPK7 permet de charger tout fichier à partir d'un lecteur de K7, même s'il est exotique, c'est à dire s'il ne respecte pas le format TANGERINE. En effet, il est possible de supprimer l'appel au sous programme SYNCHRO et de lire tout ce qu'il y a sur la bande magnétique. Dans tous les cas, les octets lus sont rangés en mémoire vive (RAM) à partir de l'adresse #1000. Le nombre d'octets à saisir peut être fixé à l'avance et n'est donc pas influencé par l'entête du fichier qui est parfois truquée ! En fait tous les paramètres de lecture sont modifiables.

Après chargement, il faut d'abord sauver tout ce qui est en mémoire par mesure de sécurité (par exemple en faisant un !SAVE«TOUT»,A#1000,E#BFFF). Puis il faut examiner ce qui a été chargé à l'aide de votre moniteur favori. J'utilise le merveilleux moniteur autorelogeable d'André Chénière, modifié par Denis Henninot pour le rendre compatible Sedoric. Il faut regarder où le fichier chargé commence réellement, où il fini et quelle est la structure de l'entête (voir plus loin la structure normale du format TANGERINE).

Après cet examen et d'éventuelles corrections (telles que changement ou ajout de caractères de synchronisation, adaptation des adresses de début et de fin de fichier, annulation du démarrage automatique etc...), il est possible de faire une sauvegarde plus propre soit sur K7 au même format exotique ou au format TANGERINE, soit sur disquette au format Sedoric.

Tel qu'il est, DUPK7 est très rudimentaire. Il a volontairement été réduit au minimum pour prendre le moins de place possible. Pour l'utiliser correctement, il faut connaître la structure des fichiers K7 Oric (voir plus loin, ainsi que «L'Oric à nu» de Fabrice Broche). Il n'y a pas d'interface utilisateur, il faut poker soit même la valeur des paramètres. Ceci dit, il est très efficace, dans les cas désespérés. Notamment pour passer sur disquette les programmes que vous avez achetés sur K7 ou pour essayer de récupérer ce qui peut l'être d'une K7 défectueuse.

Package :

Trois fichiers seulement: DUPK7.COM, DUPK7.DEM et DUPK7.HLP, qui sont indépendants. DUPK7.COM est le programme proprement dit, écrit en langage machine (voir plus loin). DUPK7.DEM est un petit fichier allant de #2000 à #2020 soit #21 = 33 octets constitués des 26 lettres de l'alphabet suivies de 7 espaces. Il vous faudra passer ce petit fichier sur K7 avant de passer aux essais (QUIT: !DUPK7.DEM: CSAVE«DUPK7DEM»,A#2000,E#2020 à sauver en plusieurs exemplaires sur la K7). DUPK7.HLP est un bref écran d'aide pouvant être utilisé à part. Enfin, MONAC1.COM le moniteur d'André Chénière revu Denis Henninot a été ajouté. Très brièvement, ce moniteur peut être lancé par CALL#AC00, faire D1000 pour examiner la mémoire à partir de #1000, faire L0800 pour désassembler DUPK7 et enfin Hxxxx pour écrire en hexadécimal à partir de l'adresse xxxx (pour plus de détails, voir le livre «Atmos Oric 1, Manuel de référence, Tome II: Travaux Pratiques» de André Chénière, paru chez Isosoft (1985). Si vous êtes pressés, les listings présentés à la fin de l'article devraient vous permettre d'essayer le programme DUPK7. Sinon, patientez jusqu'à la prochaine disquette trimestrielle...

Structure des fichiers sur K7 au format TANGERINE :

- Sur la bande magnétique, un fichier est constitué d'une succession d'octets dans l'ordre suivant :
- la bande amorce de synchronisation: 259 fois l'octet #16,
 - un octet #24 qui marque le début de l'en tête, laquelle comprend 9 octets :
 - un drapeau «Entiers», (pour STORE/RECALL)

- un drapeau «Chaînes», (pour STORE/RECALL)
- type de fichier (#00 basic, #80 bloc mémoire et #40 données),
- un drapeau de démarrage (AUTO si $\lt; 0$),
- deux octets d'adresse de début de fichier en RAM (HHLL),
- deux octets d'adresse de fin de fichier en RAM (HHLL),
- et enfin un octet #00 marquant la fin de l'en tête,
- Le nom du fichier (de longueur variable, se terminant par #00)
- Le fichier proprement dit.

Notez qu'il n'y a **pas de marque de fin de fichier**, ce sont les adresses de début et de fin qui déterminent le nombre d'octets à charger. Notez aussi que normalement les octets de l'en tête sont recopiés en RAM (de #005E à #0066 pour Oric-1 et de #02A8 à #02B0 pour Atmos), mais l'ordre est inversé, ce qui reconstitue LLHH pour les adresses (octet de poids faible en premier).

Analyse de DUPK7 :

DUPK7 occupe 121 octets de #800 à #878 et utilise 3 adresses en page zéro (#00, #01 et #02) (voir le listing source plus loin). C'est un programme **facilement relogeable** (par exemple avec un !LOAD«DUPK7",A#BB00 suivi d'un DOKE#BB04,#BB54). En outre, il faudra adapter les adresses des DOKEs et des POKEs dans ce qui suit. DUPK7 comporte 3 parties:

1) SAISIE K7 que l'on lance avec un **CALL# 800**, qui permet de charger le fichier en RAM. Attention, pendant le chargement, DUPK7 **n'affiche ni «Searching» ni «Loading»**. Soyez patients! Si le reste de la bande est sans bruit de fond ou que vous avez poké un nombre d'octets à lire beaucoup trop grand, ou si après chargement, DUPK7 ne retourne pas au Ready faire un **reset** Microdisc.

Options : a) L'adresse de début de stockage en mémoire est #1000 par défaut. Si cela ne vous convient pas, il est possible de la modifier par **POKE#807,#HH** et **POKE#809,#LL** (HHLL étant la nouvelle adresse en hexadécimal, pour #1000 HH=#10 et LL=#00).

b) Le nombre d'octets à charger est actuellement #200 (soit 512) par défaut. Ce nombre risque d'être insuffisant. Il permet de faire une première lecture du début du fichier et de se faire une première idée de sa longueur. Vous **devez** le modifier par un **POKE#80F,#HH** et un **POKE#813,#LL** (HHLL étant le nombre d'octets à charger en hexadécimal, pour #200 HH=#02 et LL=#00). N'oubliez pas qu'en plus du fichier proprement dit, DUPK7 chargera environ 1 page d'octets de synchronisation (#100 soit 256 octets) sans compter le bruit de fond de la bande avant le début, si vous avez inhibé l'appel au sous programme SYNCHRO (voir plus loin). Si le nombre que vous pokez est trop grand, DUPK7 continuera à lire après la fin du fichier. En général il trouve des octets dans le bruit de fond, sinon il faut faire un reset Microdisc (ou sous votre Oric si vous travaillez sans lecteur de disquette).

c) Si le caractère de synchronisation est différent de #16 (par exemple #68 pour le programme LOGO V1 de Loriciels), DUPK7 ne chargera rien. Il faut alors **se passer de détection de début de fichier** avec un **POKE#802,#EA** et un **DOKE#803,#EAEA** (au lieu de #20 et #0854 pour JSR 0854). Dans ce cas, DUPK7 stockera en RAM tout ce qu'il lit, même le bruit de fond avant le début du fichier ! L'examen de ce qui a été chargé permet alors de connaître le caractère de synchronisation réel ainsi que la structure du format utilisé.

2) COPIE K7 que l'on lance avec un **CALL#82B**, qui permet de sauvegarder sur K7 le fichier que vous aviez préalablement chargé en RAM et éventuellement modifié. Notez que ce sous programme permet aussi de créer des fichiers exotiques selon votre propre fantaisie: **ce qui se trouve en RAM sera copié octet par octet sur la K7 !**

Options : a) S'il y a lieu, **modifiez** les octets de synchronisation d'origine en les remplaçant en RAM (à l'aide du moniteur déjà utilisé plus haut) par des #16 (qui est la valeur normale). Vous pouvez bien sûr garder les octets «exotiques». Si le nombre d'octets de synchronisation est insuffisant, il est possible d'en **ajouter**, notamment entre #0900 et #1000. Modifiez éventuellement certains octets de l'en tête: **adresses de début** (DEBr) et **de fin** (FINr) (du fichier proprement dit lorsqu'il est à sa place finale de travail en RAM et non adresses après chargement par DUPK7), drapeau d'exécution etc...

b) Il **faut** indiquer à DUPK7 l'**adresse de début de sauvegarde** (DEBs) qui est #1000 par défaut en

faisant un **POKE#82F,#HH** et **POKE#831,#LL** (HLL étant la nouvelle DEBs que vous avez déterminée en examinant en RAM l'adresse du début de bande amorcée de synchronisation et non celle du fichier proprement dit (DEBa). En général l'adresse #1000 reste la bonne.

c) De même il **faudrait** indiquer le **nombre d'octets à sauvegarder** (NOCs) qui est de #200 (soit 512) par défaut en faisant un **POKE#837,#HH** et un **POKE#83B,#LL** (HLL étant le nouveau NOCs, nombre que vous avez déterminé en examinant ce qui avait été chargé. Ce nombre doit inclure les octets de synchronisation, l'en-tête, le nom du fichier et le fichier proprement dit (soit **NOCs = adresse du dernier octet à sauver + 1 adresse du 1er**).

3) SYNCHRO (en lecture) Ce sous-programme de **détection de début de fichier** est une réplique de celui de la ROM (situé en #E735 pour l'Atmos et en #E696 pour l'Oric 1). **Il n'est pas nécessaire**, mais facilite le travail d'examen de la RAM après chargement. Si DUPK7 ne charge rien, c'est qu'il ne trouve pas la série minimum de 3 octets #16. Parmi les protections «classiques», on trouve soit le changement de caractère de synchronisation (exemple LOGO V1.0 de Loricels), soit la réduction à 3 (au lieu de 259 normalement) du nombre de caractères de synchronisation, d'où des problèmes de chargement dès que la K7 est un peu fatiguée ! Dans ces deux cas, il faut carrément inhiber l'appel au sous programme SYNCHRO (**POKE#802,#EA** et **DOKE#803,#EAEA**). Il est par la suite facile de rajouter autant de #16 que nécessaire devant l'en-tête (à l'aide du moniteur), avant de passer à la copie sur K7.

4) Sauvegarde Sedoric: Connaissant l'adresse actuelle de début du fichier proprement dit (DEBa) et son adresse actuelle de fin (FINa) (attention il s'agit des valeurs après chargement avec DUPK7, valeurs trouvées à l'aide du moniteur) ainsi que l'adresse réelle de début (DEBr) et de fin (FINr) (attention il s'agit des adresses finales que le fichier devra occuper en RAM pour être opérationnel) faire un **!SAVE«FICHER»,A#DEBa,E#FINa** puis un **!LOAD«FICHER»,A#DEBr** et enfin un **!SAVEO«FICHER»,A#DEBr,E#FINr**. On peut aussi faire un **!MOVE#DEBa,#FINa,#DEBr** suivi d'un **!SAVE«FICHER»,A#DEBr,E#FINr**.

Mode d'emploi simplifié :

NOCs nombre d'**O**ctets à sauvegarder incluant synchro, en-tête, nom et fichier proprement dit
DEBs adr de **DE**but de zone à sauvegarder sur K7, c'est aussi l'adr du début de bande de synchro
DEBa adr de **DE**but actuel du fichier proprement dit
FINa adr de **FIN** actuelle du fichier proprement dit
Il s'agit des adresses après chargement par DUPK7, hors synchro, en-tête etc...
DEBr adr de **DE**but réelle = adresse finale de travail
FINr adr de **FIN** réelle = adresse finale de travail du fichier proprement dit.
Tous ces nombres sont en hexadécimal et sont bornes incluses.

- 1) Allumez l'Oric avec une disquette master sans INIST
- 2) Faire HIMEM#7FF puis QUIT et !DUPK7
- 3) Chargement : POKE#80F,#HH et POKE#813,#LL (HLL = nombre d'octets à lire)
CALL#800 (bande en position de lecture)
- 4) Sauvegarde provisoire : !SAVE«TOUT»,A#1000,E#(2000+NOCs)
Si l'on n'a aucune idée de la valeurs de NOCs -> E#BFFF)
- 5) Examen de ce qui a été chargé en RAM (avec MONAC1 par exemple)
- 6) Sauvegarde plus propre :
 - 6a) Sur K7 : POKE#82F,#HH et POKE#831,#LL (HLL = DEBs)
POKE#837,#HH et POKE#83B,#LL (HLL = NOCs)
CALL#82B (position d'enregistrement K7)
 - 6b) Sur disquette : !MOVE#DEBa,#FINa,#DEBr et !SAVE«FICHER»,A#DEBr,E#FINr

Essai réel avec DUPK7.DEM :

1) Faire un fichier K7 avec DUPK7.DEM :

- 1a) Allumer l'Atmos avec une disquette master sans INIST
- 1b) Faire QUIT et !DUPK7.DEM

1c) CSAVE«DUPK7DEM»,A#2000,E#2020 (position enregistrement K7)

1d) Répétez le §1c) pour avoir 3 ou 4 copies à la suite sur la K7

2) Essai de DUPK7 en chargement:

2a) Allumer l'Atmos avec une disquette master sans INIST

2b) Faire HIMEM#7FF puis QUIT et !DUPK7.DEM

2c) CALL#800 (position de lecture K7)

3) Première sauvegarde globale: !SAVE«TOUT»,A#1000,E#2200

4) Examen de ce qui a été stocké en RAM (peut être légèrement différent) :

4a) De #1000 à 10FE se trouvent les octets de synchro #16 (259, mais SYNCHRO en a mangé 4)

4b) En #10FF se trouve l'octet #24 de début d'en-tête

4c) De #1100 à 1108 : 9 octets d'en tête soit : 00 00 80 00 20 20 00 00 (fin #2020, début #2000)

4d) De #1109 à 1111 le nom DUPK7DEM suit de #00

4e) Enfin de #1112 à 1132 se trouve le fichier proprement dit :

«ABCDEFGHIJKLMNPOQRSTUVWXYZ »

4f) A la suite, DUPK7 a lu n'importe quoi dans le bruit de fond.

On en déduit que : DEBs = #1000 et NOCs = #0133 (#1132-#1000+1)

DEBa = #1112 et FINa = #1132

DEBr = #2000 et FINr = #2020

5) Sauvegarde propre :

5a) Sur K7 : POKE#837,#01 et POKE#83B,#33 (NOCs = #0133)
CALL#82B (K7 en position d'enregistrement)

5b) Sur disquette : !SAVE«DUPDEM»,A#1112,E#1132 (DEBa, FINa)
!LOAD«DUPDEM»,A#2000 (DEBr)

!SAVEO«DUPDEM»,A#2000,E#2020 (DEBr FINr)

ou encore : !MOVE#1112,#1132,#2000 (DEBa FINa DEBr)
!SAVE«DUPDEM»,A#2000,E#2020 (DEBr FINr)

Programme source commenté :

1) Partie lecture/chargement (SAISIE K7) :

800 20 6A E7 JSRE76A configuration travail K7
803 20 54 08 JSR0854 détection début fichier POKE#803,#EA et
806 A9 10 LDA#10 DOKE#804,#EAEA pour inhiber ce JSR
808 A0 00 LDY#00 en 01 (HH=#10) et 00 (LL=#00) :
80A 85 01 STA01 adresse de stockage soit #1000
80C 84 00 STY00 en 02 (HH=#02) et X (LL=#00)
80E A2 02 LDX#02 nombre d'octets à charger : #0200
810 86 02 STX02 modification : POKE#80F,#HH et
812 A2 00 LDX#00 POKE#813,#LL
814 F0 OD BEQ0823 si X (LL) nul, décrémente 02 (HH)
816 20 C9 E6 JSRE6C9 charge un octet (lecture K7)
819 91 00 STA(00),Y et le stocke en RAM
81B C8 INY indexe place de stockage suivante
81C D0 02 BNE0820 si fin de page de stockage...
81E E6 01 INC01 incrémente HH (page suivante)
820 CA DEX décrémente nombre octets à charger
821 D0 F3 BNE0816 si LL pas nul, charge octet suivant
823 C6 02 DEC02 sinon décrémente HH de nombre octets
825 10 EF BPL0816 si HH pas nul, charge octet suivant
827 20 3D E9 JSRE93D sinon c'est fini : VIA OFF
82A 60 RTS retour au Ready

2) Partie écriture sur K7 (COPIE K7):

82B 20 6A E7 JSRE76A configuration travail K7

| | | | |
|------------|----------|-----------|--|
| 82E | A9 10 | LDA#10 | en 01 (HH=10) et en 00 (LL=00) |
| 830 | A0 00 | LDY#00 | soit #1000 adresse de début |
| 832 | 85 01 | STA01 | de sauvegarde, modifiable par |
| 834 | 84 00 | STY00 | POKE#82F,#HH et POKE#831,#LL |
| 836 | A2 02 | LDX#02 | en 02 (HH=02) et X (LL=00) soit |
| 838 | 86 02 | STX02 | #0200 nombre d'octets à charger |
| 83A | A2 00 | LDX#00 | modification POKE#837,#HH POKE#83B,#LL |
| 83C | F0 0D | BEQ084B | si X (LL) nul, décrémente 02 (HH) |
| 83E | B1 00 | LDA(00),Y | lit octet en RAM |
| 840 | 20 5E E6 | JSRE65E | et l'écrit sur K7 |
| 843 | C8 | INY | indexe place de lecture suivante |
| 844 | D0 02 | BNE08448 | si fin de page de lecture... |
| 846 | E6 01 | INC01 | incréméte HH (page suivante) |
| 848 | CA | DEX | décrémente nombre octets à lire |
| 849 | D0 F3 | BNE083E | si LL pas nul, lit octet suivant |
| 84B | C6 02 | DEC02 | sinon décrémente HH de nombre octets |
| 84D | 10 EF | BPL083E | si HH pas nul, lit octet suivant |
| 84F | 20 3D E9 | JSRE93D | sinon c'est fini : VIA OFF |
| 852 | 60 | RTS | retour au Ready |
| 853 | EA | NOP | No Operation |

3) Sous programme SYNCHRO (recherche bande amorce) :

| | | | |
|------------|----------|---------|---|
| 854 | 20 FC E6 | JSRE6FC | prend un bit sur la K7 |
| 857 | 66 2F | ROR 2F | et assemble l'octet |
| 859 | A9 16 | LDA#16 | charge accumulateur avec octet de référence |
| 85B | C5 2F | CMP 2F | est ce #16 (octet de synchro) |
| 85D | D0 F5 | BNE0854 | si non, reprend recherche |
| 85F | AD 4D 02 | LDA024D | si oui, teste si mode FAST |
| 862 | F0 08 | BEQ086C | si oui, on passe |
| 864 | 20 1C E7 | JSRE71C | si non, saute bit de parité |
| 867 | 20 1C E7 | JSRE71C | et saute les bits de stop |
| 86A | B0 FB | BCS0867 | tant que carry est set |
| 86C | A2 03 | LDX#03 | vérifie que au moins 3 octets #16 |
| 86E | 20 C9 E6 | JSRE6C9 | prend un octet sur K7 |
| 871 | C9 16 | CMP#16 | est ce #16? |
| 973 | D0 DF | BNE0854 | non : fausse alerte recommence tout |
| 975 | CA | DEX | oui, décrémente le nombre à trouver |
| 976 | D0 F6 | BNE086E | jusqu'à compteur nul = synchro OK |
| 878 | 60 | RTS | retour au programme appelant |

4) Adaptation Oric-1, remplacer :

| | | | |
|---------|-----------|---------------|-----------------------------------|
| En #800 | JSR\$E76A | par JSR\$E6CA | configurer le VIA pour travail K7 |
| En #816 | JSR\$E6C9 | par JSR\$E630 | prendre un octet sur la K7 |
| En #827 | JSR\$E93D | par JSR\$E804 | reconfigurer le VIA |
| En #82B | JSR\$E76A | par JSR\$E6CA | configurer le VIA pour travail K7 |
| En #840 | JSR\$E65E | par JSR\$E5C6 | écrire un octet sur la K7 |
| En #84F | JSR\$E93D | par JSR\$E804 | reconfigurer le VIA |
| En #854 | JSR\$E6FC | par JSR\$E65E | prendre un bit sur la K7 |
| En #85F | LDA\$024D | par LDA\$0067 | drapeau mode FAST si nul |
| En #864 | JSR\$E71C | par JSR\$E67D | prendre long. du créneau courant |
| En #867 | JSR\$E71C | par JSR\$E67D | prendre long. du créneau courant |
| En #86E | JSR\$E6C9 | par JSR\$E630 | prendre un octet sur la K7 |