

WINDOWS



McSTRAD est un utilitaire qui permet le transfert d'une partie de l'écran vers une zone mémoire, et l'opération inverse: la reconstitution d'une partie de l'écran à partir des données sauvegardées au préalable dans la mémoire de l'ordinateur.

Ces transferts, au sens propre du terme, s'effectuent grâce à deux commandes RSX.

La première, |SWIN (pour Store WINDOW), sauvegarde une zone délimitée de l'écran dans la mémoire. Sa syntaxe est |SWIN, x, y, lx, ly
Les deux paramètres x (avec $1 <= x <= 80$) et y (tel que $1 <= y <= 25$) constituent ici l'origine de la partie à transférer. Les deux arguments suivants lx et ly définissent la taille de la fenêtre à sauvegarder (lx : nombre de colonnes : ly : nombre de lignes). Toutefois, il faut prendre garde au mode dans lequel ce transfert s'effectue: en effet, si en mode 2, le nombre de caractères d'une ligne à transférer s'écrit lx, en mode 1 pour sauver x caractères sur une ligne, il faut que $lx = x * 2$ et en mode 0 pour le même nombre x de caractères $lx = x * 4$

La deuxième commande, RWIN (pour Recall WINDOW), effectue le

transfert des données stockées en mémoire par |SWIN vers l'écran pour reconstituer la partie sauvegardée. Cette commande ne nécessite aucun argument particulier (elle s'écrit donc |RWIN) car elle emploie ceux utilisés auparavant avec |SWIN (il est donc déconseillé d'essayer de faire |RWIN avant |SWIN)

Le fonctionnement de la routine est extrêmement simple: elle se contente de décomposer chaque caractère à sauvegarder en huit octets (en mode 2 du moins): un octet pour chaque ligne qui compose un caractère puis d'effectuer l'opération inverse pour le réafficher à l'écran. Ce qui permet d'ouvrir et de fermer très facilement une fenêtre à l'écran (comme sur le McIntosh

d'APPLE par exemple).

Le programme de démonstration montre l'une des applications possibles de cette routine:

- lignes 230 à 590: implantation de la routine en langage machine à l'adresse &A000 (attention, sans modification cette routine n'est pas relogeable). Le POKE &A02F, &80 en ligne 240 indique à la routine l'adresse de la zone mémoire où commence le stockage des données (dans ce cas: &8000: d'où le MEMORY &7FFF). Cette adresse peut être modifiée à volonté, il faut l'implanter en &A02E (partie basse) et &A02F (partie haute). Plus l'espace réservé sera grand (donc l'adresse de début plus petite) et plus le nombre de données sauvegardées sera important.

- ligne 630: redéfinition des directions du joystick afin de les faire agir de la même façon que les flèches de curseur (la touche FIRE faisant office de COPY pour la validation du choix).

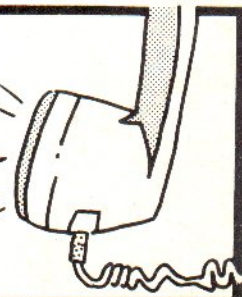
- ligne 710: ti\$ est la variable contenant le titre de la fenêtre à afficher. CH contient le nombre de choix dans la fenêtre. H est la position horizontale de cette fenêtre (colonne) et V la position verticale (ligne). Le RESTORE 1100 sert à réinitialiser le pointeur de DATA de façon à ce que les données sui-

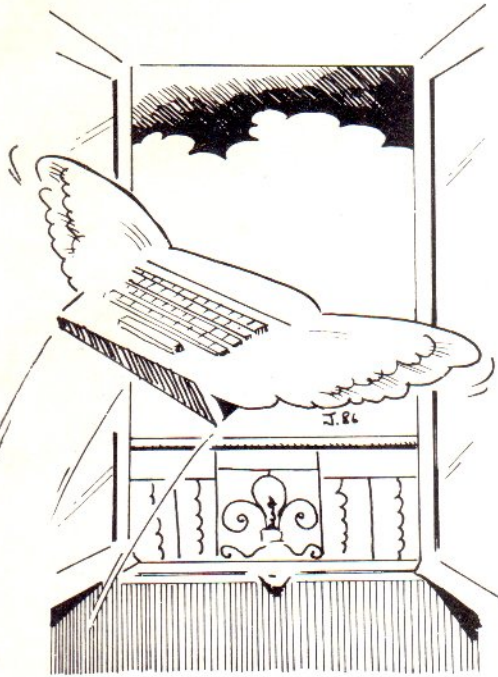
Mark HOEBEKE

Une ligne téléphonique est à votre disposition, vous mettant en contact direct avec la rédaction. Ceci est un service sans égal! Respectez simplement les horaires et les jours que nous vous indiquons:
MERCREDI de 9 h à 12 h et de 14 h à 17 h.
VENDREDI de 9 h à 12 h seulement.
Tout appel en dehors de ces créneaux sera refoulé: ne dépensez pas inutilement votre argent!

Le numéro: **99.52.98.11.**

ALLO!
CPC!
99.52.98.11.





vantes qui seront lues soient les choix contenus dans la fenêtre. Au retour du GOSUB, la variable CH contient le numéro du choix sélectionné.

— ligne 720 : branchement suivant le choix effectué.

— lignes 730 à 920 : exécution de sous-programmes suite aux branchements (rien n'interdit d'y placer à nouveau des fenêtres).

— lignes 940 à 1060 : affichage de la fenêtre et sélection du choix (la taille de la fenêtre est calculée automatiquement à partir du nombre de données et de leur taille).

— ligne 1100 : données concernant les fenêtres à afficher (cf ligne 710).

Maintenant que tout est clair (enfin, je l'espère), il ne vous reste plus qu'à intégrer ce sous-programme dans vos propres réalisations afin de les rendre plus esthétiques.

Alors, bonne chance...



```

100 *****
110 ***          Mc STRAD          ***
120 *** pour AMSTRAD CPC 464 ***
130 ***          (c) HOEBEKE MARK ***
140 ***          et                ***
  
```

```

150 ***          CPC              ***
160 ***          JUIN 1986        ***
170 *****
180 '
190 '
200 '
210 *** IMPLANTATION DE LA ROUTINE ASSEMBLEUR ***
220 '
230 AD=&A000:MEMORY &7FFF
240 READ A$:IF A$="FIN" THEN POKE &A02F,
&B0:CALL &A000:GOTO 630 ELSE POKE AD,VA
L("&*A$):AD=AD+1:GOTO 240
250 DATA 1,9,A0,21,1A,A0,C3,D1,BC,11
260 DATA A0,C3,32,A0,C3,E9,A0,53,57,49
270 DATA CE,52,57,49,CE,0,0,0,9,A0
280 DATA 45,33,88,41,D,80,2B,4C,49,47
290 DATA 29,2C,41,D,9F,D,0,23,22,27
300 DATA FE,4,C0,DD,66,7,DD,6E,6,E5
310 DATA DD,66,5,DD,6E,4,E5,DD,66,3
320 DATA DD,6E,2,E5,DD,66,1,DD,6E,0
330 DATA 22,24,A0,E1,22,22,A0,E1,22,20
340 DATA A0,E1,22,1E,A0,21,0,C0,AF,32
350 DATA 30,A0,11,50,0,3A,30,A0,3C,32
360 DATA 30,A0,4F,3A,20,A0,B9,2B,3,19
370 DATA 18,EF,EB,2A,1E,A0,19,2B,22,26
380 DATA A0,2A,2E,A0,22,2A,A0,2A,26,A0
390 DATA 22,2B,A0,22,2C,A0,AF,32,30,A0
400 DATA 32,31,A0,CD,B7,A0,3A,31,A0,3C
410 DATA 32,31,A0,4F,3E,8,B9,C8,2A,2B
420 DATA A0,11,0,8,19,22,2C,A0,22,2B
430 DATA A0,18,E2,2A,2A,A0,E5,2A,22,A0
440 DATA E5,C1,D1,2A,2C,A0,ED,80,EB,22
450 DATA 2A,A0,3A,30,A0,3C,32,30,A0,4F
460 DATA 3A,24,A0,B9,2B,C,2A,2C,A0,11
470 DATA 50,0,19,22,2C,A0,18,D3,AF,32
480 DATA 30,A0,C9,2A,2E,A0,22,2A,A0,2A
490 DATA 26,A0,22,2B,A0,22,2C,A0,AF,32
500 DATA 31,A0,32,30,A0,CD,1D,A1,3A,31
510 DATA A0,3C,32,31,A0,4F,3E,8,B9,C8
520 DATA 2A,2B,A0,11,0,8,19,22,2B,A0
530 DATA 22,2C,A0,18,E2,2A,2C,A0,E5,2A
540 DATA 22,2A,A0,3A,30,A0,3C,32,30,A0
550 DATA 4F,3A,24,A0,B9,2B,D,11,50,0
570 DATA 2A,2C,A0,19,22,2C,A0,C3,1D,A1
580 DATA AF,32,30,A0,C9,0,0,0,0,0
590 DATA FIN
600 '
610 *** TRANSFORMATION DES DIRECTIONS J
OYSTICK EN TOUCHES CURSEUR ***
620 '
630 KEY DEF 72,1,240,244:KEY DEF 73,1,24
1,245:KEY DEF 74,1,242,246:KEY DEF 75,1,
243,247:KEY DEF 76,1,13,13
640 '
650 *** INITIALISATION ECRAN ***
660 '
670 MODE 2:INK 0,0:INK 1,26:BORDER 1:PAP
ER 1:PEN 0:CLS:CLG 1
680 '
690 *** PREPARATION FENETRE ***
700 '
710 TI$="EXEMPLE DE FENETRES":CH=4:H=28:
V=10:RESTORE 1100:GOSUB 960
720 DN CH GOSUB 740,790,840,770:LOCATE 1
,1:END
730 '
740 *** Traitement de la premiere optio
n ***
  
```

```

750 '.....
760 '.....SOUS PROGRAMME 1.....
770 '.....
780 RETURN
790 *** Traitement de la deuxieme optio
n ***
800 '.....
810 '.....SOUS PROGRAMME 2.....
820 '.....
830 RETURN
840 *** Traitement de la troisieme opti
on ***
850 '.....
860 '.....SOUS PROGRAMME 3.....
870 '.....
880 *** Traitement de la quatrieme opti
on ***
890 '.....
900 '.....SOUS PROGRAMME 4.....
910 '.....
920 RETURN
930 '
940 *** SOUS PROGRAMME DE TRAITEMENT DE
FENETRES ***
950 '
960 LOCATE 1,1:PRINT:z=ch:FOR I=1 TO CH:
READ CH$(I):NEXT
970 TAGOFF:LO=LEN(TI$):MA=CH:FOR XX=1 TO
CH:IF LEN(CH$(XX))>LO THEN LO=LEN(CH$(X
X))
980 NEXT XX:LO=LO+4:BA=V+CH+2:WINDOW H,H
+LO+1,V,BA+1:ISWIN,H,V,LO+2,CH+4:CLS:WIN
DOM 1,80,1,25
990 I=LO+1:X=H+8-4-1:Y=398-(V*16-12)-9:
LX=LO*8-8+1:LY=-((CH+2)*16+6)-5:DX=(H-1
+LO-LEN(TI$))/2*8:DY=398-(V*16)-6:XI=H*
8-1:YI=398-(V*16-8)-9:L1=LO*8-16+1:L2=-2
0
1000 PLOT X,Y,0:DRAW X+LX,Y:DRAW X+LX,Y+
LY:DRAW X,Y+LY:DRAW X,Y:MOVE DX,DY:TAG:P
RINT TI$:PLOT X1,Y1,0:DRAW X1+L1,Y1:DRA
W X1+L1,Y1+L2:DRAW X1,Y1+L2:DRAW X1,Y1
1010 X=H+8-1:Y=398-((V+2)*16-16)-9:LX=LO
*8-16+1:LY=-16*ch-6-5:XI=H*8+4:YI=394-((
V+1)*16)-9:DX=a-1:DY=c+2:L1=K1*8-14+1:L2
=-2*16-4:D1=H*8+4:D2=394-((v+ch)*16)
1020 TAGOFF:PLOT x,y:DRAW x+lx,y:DRAW x+
lx,y+ly:DRAW x,y+ly:DRAW x,y:FOR i=1 TO
ch:LOCATE h+3,v+i+2:PRINT ch$(i):NEXT:P
LOT dx,dy:DRAW dx+11,dy:DRAW dx+11,dy+12
:DRAW dx,dy+12:DRAW dx,dy:ch=1:a$="":WHI
LE a$(<)CHR$(13) AND a$(<)CHR$(58)
1030 PAPER 0:PEN 1:LOCATE h+1,v+ch+2:PRI
NT " ";ch$(ch);SPACE$(10-LEN(ch$(ch))-4)
1040 B$="":WHILE B$(<)CHR$(240) AND B$(<)C
HR$(241) AND B$(<)CHR$(13):B$=INKEY$:WEND
I$=B$:PAPER 1:PEN 0:LOCATE h+1,v+ch+2:P
RINT " ";ch$(ch);SPACE$(10-LEN(ch$(ch))-
4)
1050 ch=ch-(a$=CHR$(240))*ch+1+(a$=CHR
$(241))*ch+a$:WEND
1060 PAPER 1:PEN 0:LOCATE h+2,v+ch+2:PRI
NT CHR$(246);ch$(ch);CHR$(247):IRWIN:RET
URN
1070 '
1080 *** DONNEES CONTENUES DANS LES FEN
ETRES ***
1090 '
1100 DATA PREMIERE OPTION,DEUXIEME OPTIO
N, TROISIEME OPTION, QUATRIEME OPTION
  
```