

Sommaire

• Series 3

- Quelle heure avez-vous ?
- Multitâche
- Intérêts composés
- Oh graphique...
- La touche Tab
- Biorythme
- Manipuler vos images
- A vous de jouer
- Quelle sortie pour ...
- Transferts de données
- Permutation
- Liste des disques présents

• Organiseur II

- La version de votre ...
- ASIN et ACOS
- Mastermind
- Appeler option Comms
- Caractères accentués

• Versions

Edité par

Aware

21 Rue O. METRA - 75020 PARIS
Tél: (1) 46 36 46 47 Fax: (1) 46 36 82 54

Bulletin Technique

La revue du support technique d'Aware

Bien sûr ce n'est pas une obligation....

Vous pourriez vous contenter du puissant gestionnaire de fiches.

Vous pourriez n'avoir besoin que du magnifique traitement de texte.

Vous pourriez n'utiliser que le superbe agenda graphique.

Vous pourriez voyager tout autour du globe dans votre appareil.

Vous pourriez envisager de recalculer tout Einstein sur la calculatrice.

Mais ne pas programmer sur votre Series 3 ???

Le Series 3 nous permet une programmation de rêve. Ses commandes touchent des domaines aussi variés que le graphisme, les menus déroulants, les boîtes de dialogue, la gestion de fiches, la gestion de temps etc..

Bien entendu nous n'avons pas l'intention d'abandonner les fidèles utilisateurs d'organiseurs. Ils trouveront ici de quoi satisfaire leur sympathique machine grise, mais l'explosion du Series 3 nous conduit à réserver un espace important à cette nouvelle et attrayante machine de poche.

Mais assez de discours : Tournez la page !

Quelle heure avez-vous ?

Series 3

Cette petite procédure permet de régler l'horloge Système en dehors de l'application "Heure" et vous montre la manipulation d'une boîte de dialogue ainsi que l'utilisation des fonctions DDATE et DTIME. Le réglage de la date est arbitrairement limité entre le 1er Janvier 1992 et le 31 décembre 2000. Libre à vous de changer ces valeurs !

Les limites de DTIME sont plus conventionnelles : de 00:00:00 à 23:59:59. Il est possible de les changer également, mais l'intérêt de cette manipulation est moins évident.

La procédure fait appel à une routine en code machine, le soin le plus extrême est à apporter à sa frappe.

```
PROC REGLHEUR:
LOCAL AX%,BX%,CX%,DX%,SI%,DI%
LOCAL SEC%,DAT%
DAT%=33602
DINIT
DDATE DAT%,"DATE ",33602,36524
REM *****
REM 33602 = 01/01/1992
REM 36524 = 31/12/2000
REM *****
DTIME SEC%,"HEURE ",1,0,86399
REM *****
REM 86399 = 23:59:59
REM *****
IF DIALOG
DAT%=(DAT%-25567)*86400
SEC%=SEC%+DAT%
CX%=PEEKW(ADDR(SEC%)+2)
DX%=PEEKW(ADDR(SEC%))
AX%=$0300
OS ($89,ADDR(AX%))
ENDIF
ENDP
```

Multitâche

Series 3

Vous savez sans doute, et si vous ne le savez pas vous l'apprenez, qu'il est possible d'effectuer une sauvegarde de l'écran en cours en appuyant en même temps sur les touches Ctrl,Maj,Psion et S. Le fichier ainsi généré est écrit à la racine du disque interne (M) sous le nom SCREEN.PIC. Si jamais vous répétez cette opération, le nouveau fichier viendrait remplacer le précédent fichier puisqu'il porte le même nom (screen.pic) et ainsi le détruire.

La solution consiste, grâce au système d'exploitation multitâche de renommer le fichier SCREEN.PIC lorsqu'il est repéré par le programme ci-dessous. Le nom du fichier est alors recalculé pour tenir compte d'un numéro d'ordre des images.

Voici une solution bien agréable pour illustrer les documentations de vos applications.

Il sera d'ailleurs toujours possible de visualiser l'état actuel des travaux soit en consultant l'écran de l'application ci-dessous, soit en consultant la liste des fichiers située à la racine du disque interne.

```
PROC start:
local s%:REM Compteur
local f$(15):REM Nom du fichier avant
local s$(15):REM Nom du fichier après
f$="m:\screen.pic"
do
if exist(f$)
s%=s%+1
s$="m:\scr"+right$(gen$(10000+s%,5),4)
s$=s$+".pic"
rename f$,s$
print f$,chr$(16),s$
endif
until 0
ENDP
```

Intérêts composés

Series 3

Quelle sera la somme disponible au bout de 5 ans si vous placez 10 000 Francs à 11% ?

Facile : 16850,58 Francs !

Je l'avoue, j'ai triché, j'ai utilisé la procédure INTCOMP. Elle m'a d'abord demandé la somme que je voulais placer et le taux annuel. Ensuite, il m'a suffi d'appuyer cinq fois sur la touche 'Entrer'.

A noter que pour des raisons d'affichage, les sommes supérieures à 9 999 999,00 sont représentées par des "*". A ce niveau, il est recommandé d'acheter un ordinateur plus important ou de supprimer la flèche qui pointe sur "INT."

```
PROC INTCOMP:
LOCAL IN,A%,CAP,CAP$(10),E%
CLS
DINIT
DTEXT "","CALCUL D'INTERETS ",$302
DFLOAT CAP,"CAPITAL ",0,1000000
DFLOAT IN,"TAUX ANNUEL",0,100
REM *****
REM Somme maxi 1 000 000
REM Taux maxi 100% !
REM *****
IF DIALOG
A%=1
DO
PRINT "ANNEE ";1991+A%;"("A%");";
IF A%<10 : E%=1
ELSE E%=0
ENDIF
PRINT REPT$ (" ",2+E%);"> ";
CAP=CAP+(CAP*IN)/100
CAP$=FIX$(CAP,2,-10)
PRINT "CAP.+INT. ";CAP$
A%=A%+1
UNTIL GET=27
ENDIF
ENDP
```

Oh graphique...

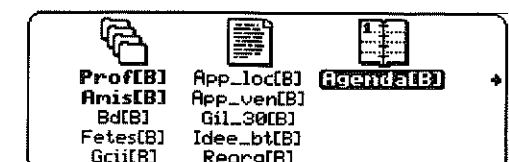
Series 3

Une procédure simple qui nous explique comment "sortir" une image à l'écran. Elle débute par une boîte de dialogue qui est là pour saisir le nom de l'image. Si vous ne possédez pas encore d'image, faites une copie d'écran en appuyant simultanément sur les touches Ctrl/Maj/Psion/S. L'image créée se trouve dans la racine de l'unité interne et a pour nom : "Screen.pic". Un petit "+" à la fin de la procédure : la possibilité d'inverser l'image en pressant la touche "i" !

```
PROC image:
local id%,nom$(128),g%,l%,h%
cls
REM *****
REM boîte de dialogue
REM *****
dinit
dfile nom$,"Fichier : ",0
if dialog
id%=gloadbit (nom$)
l%=gwidth
h%=gheight
REM ***** Largeur
REM ***** Hauteur
guse 1
REM GUSE 1 = Ecran de base
```

```
gcopy id%,0,0,l%,h%,3
REM *****
REM copie de l'image sur
REM cet écran de base de
REM l'image id%.
REM *****
g%=get
if g%=27
REM ***** Si Esc
return
elseif g%=%i or g%=%I
REM ***** Inversion
gat 0,0
ginvert l%,h%
get
endif
endif
endp
```

Exemple de copie d'écran



La touche Tab Series 3

L'utilisation de la touche TAB dans les copies de fichiers.

Sur les Series 3 et sur les SSD, les fichiers sont stockés selon une structure MS-DOS. Cette structure, qui utilise des répertoires et des sous-répertoires, est transparente pour l'utilisateur. Elle est pourtant extrêmement importante pour toutes les opérations de copie et de sauvegarde par exemple, et passe presque toujours par l'utilisation de la touche TAB.

Il est primordial de bien la connaître : un petit entraînement s'impose donc !

Dans l'écran Système, placez la zone inversée sur "CALC" et appuyez sur la touche TAB. Vous vous trouvez au niveau de la racine du disque en cours (dont le nom est indiqué en haut de l'écran). La racine est l'endroit le plus haut dans la hiérarchie DOS. Elle est représentée par une barre de fraction inversée que vous voyez en appuyant sur la flèche haute. Sous cette racine, entourés par ces deux mêmes barres de fraction inversées, se trouvent les premiers sous-répertoires \AGN\, \APP\, \DAT\, \OPD\ etc... et éventuellement des noms de fichiers. Les répertoires d'origine contiennent les fichiers créés par les applications résidentes du Series 3. \WRD\ contiendra par exemple le fichier "Texte.wrd" alors que \WLD\ contient lui "Monde.wld" ; les fichiers sources OPL seront rangés dans le répertoire \OPL\ et auront l'extension ".opl". Vous l'avez compris, chaque répertoire contient les fichiers qui lui sont propres. C'est le Series 3 qui se charge de la gestion de ce "dispatching". Il est néanmoins possible de créer ses propres répertoires, que ce soit au niveau de la racine ou dans un répertoire existant ; ce faisant, nous développons la structure arborescente. Il est également possible de copier un fichier d'un répertoire

vers un autre, ce que nous allons faire en copiant par exemple un fichier OPL dans le traitement de texte.

Rien de plus simple :

Dans l'écran Système, portez-vous sur l'icône OPL et tapez simultanément sur la touche PSION et la lettre "C" (Copie).

Dans la boîte de dialogue qui s'ouvre à vous, placez-vous sur la deuxième ligne et appuyez sur TAB, ce qui vous permet de choisir l'unité de départ ; ensuite, revenez sur la première ligne et appuyez de nouveau sur TAB pour choisir le nom du fichier à copier. Sur la troisième ligne, tapez "\WDR", le nom du répertoire de destination. Sur la quatrième ligne, toujours par la pression de la touche TAB, choisissez l'unité de destination. Ici, il est conseillé de travailler sur "Interne" de manière à pouvoir ensuite libérer facilement la place mémoire de ces essais.

Les deux dernières lignes ne nous intéressent pas, appuyez sur la touche 'Entrer' : votre fichier est copié !

La meilleure façon de s'en assurer est de se placer sur l'icône texte et d'appuyer sur... TAB ! Nous sommes dans le répertoire \WRD\ et le fichier. OPL apparaît.

Il nous est maintenant possible de l'ouvrir ou de le fusionner dans Texte, voire de le renommer en fichier .WRD, pour que son nom puisse apparaître sous l'icône Texte.

Ces manipulations simples devront être bien comprises et maîtrisées avant de passer aux transferts vers une unité distante.

Biorythme Series 3

Voici un classique de l'informatique. Le calcul du biorythme utilise la date de naissance rapportée à la date du jour pour déterminer la forme physique, émotive et cérébrale d'un sujet.

Il est possible d'enregistrer sous forme de fichier, et par conséquent de pouvoir le sélectionner par la suite sous l'icône de cette application (il faudra la créer), les informations relatives à un sujet (date de naissance & nom).

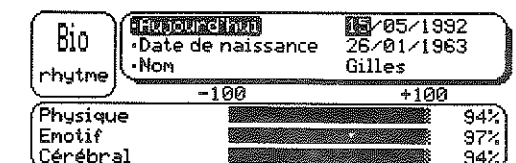
```
APP Bio
type 3
path "\bio"
ext "bio"
icon "biorythm.pic"
ENDP

PROC start:
global file$(128),t$(1)
global nom$(12),dn$,dj$,d$,p$,e$,i$
giprint "Aware 1992 - v1.2F"
dj$=days(day,month,year)
init:
gfont 1
gstyle 0
do
lock on
dinit
dposition 1,-1
ddate
dj$,"Aujourd'hui",days(1,1,1900),days(31,12,2049)
ddate dn$,"Date de naissance",days(1,1,1900),days(31,12,2049)
dedit nom$,"Nom",12
if dialog
if dn>dj$
giprint "Date de naissance > date du jour"
else
d&=(dj$-dn$)
affi:
endif
else
if t$="C"
a.nom$=nom$
a.dn$=dn$
append
elseif t$="O"
if a.nom$<>nom$ or a.dn$<>dn$
a.nom$=nom$
a.dn$=dn$
update
endif
endif
break
endif
until 0
lock off
ENDP
```

```
PROC affi:
gat 100,47 :gpatt -1,100,8,3
gat 100,57 :gpatt -1,100,8,3
gat 100,67 :gpatt -1,100,8,3
p$=int(sin(2*pi*(d$-(d$/23*23))/23)*100)
e$=int(sin(2*pi*(d$-(d$/28*28))/28)*100)
i$=int(sin(2*pi*(d$-(d$/33*33))/33)*100)
gat 100,47 :gfill int((100+p$)/2),8,0
gat 100,57 :gfill int((100+e$)/2),8,0
gat 100,67 :gfill int((100+i$)/2),8,0
gat 205,55 :gprintb num$(p$,4)+"%",30,1
gat 205,65 :gprintb num$(e$,4)+"%",30,1
gat 205,75 :gprintb num$(i$,4)+"%",30,1
ENDP

PROC INIT:
file$=cmd$(2)
t$=cmd$(3)
gat 0,0
gborder $204,45,45
gat 0,45
gborder $204,240,35
gat 5,20
gstyle 8
gprintb "Bio",35,3
gstyle 0
gat 5,35
gprintb "rythme",35,3
gat 85,44 :gprint "-100"
gat 185,44 :gprint "+100"
gat 5,55 :gprintb "Physique",60
gat 5,65 :gprintb "Emotif",60
gat 5,75 :gprintb "Cérébral",60
if t$="C"
trap mkdir left$(file$,12)
create file$,a,nom$,dn$
nom$="??????"
elseif t$="O"
open file$,a,nom$,dn$
nom$=a.nom$
dn$=a.dn$
d$=(dj$-dn$)
affi:
endif
setname file$
ENDP
```

Copie d'un écran Series 3 de l'application
Biorythme



Manipuler vos images Series 3

Voici une procédure de manipulation d'images. Le plus simple pour se procurer une image digne de ce nom, est d'aller dans "Monde" et de faire une copie d'écran.

Tapez la séquence de touches Ctrl/Maj/PSION/S. L'appui simultané sur ces quatre touches va créer un fichier "Screen.pic" dans la racine de l'unité interne. Il est bien sûr possible de copier cette image dans un autre répertoire et surtout de la renommer car le Series 3 ne change jamais le nom ni la destination de la copie d'écran. Mais là n'est pas le but de cet article et la procédure "SCROLL" va chercher de quoi travailler dans la racine.

Cette procédure fait défiler notre image de la gauche vers la droite et utilise deux techniques :

Entre le moment où l'image est complètement invisible et où elle occupe la totalité de l'écran, des "tranches" de plus en plus larges sont amenées à l'écran, ensuite l'image est scrollée de manière classique.

Gupdate Off et Gupdate ne sont là que pour accélérer un peu le processus.

Le Scroll est une possibilité extrêmement riche et sophistiquée sur un appareil de cette taille et fait partie de la bibliothèque très étendue des commandes graphiques de notre machine.

A vous de jouer Series 3

Ce programme, même s'il ne vous assure pas de gagner une fortune au Loto, vous permet tout de même de vous familiariser avec les boîtes de dialogue et avec les manipulations de nombres aléatoires. Il se

```
PROC scroll:
  local id%,i%,x%,y%
  gupdate off
  id%=gloadbit ("m:\screen.pic")
  guse 1
  x%=239
  REM *****
  REM largeur de l'écran
  REM *****
  y%=1
  REM *****
  REM Première "tranche"
  REM *****
  do
    gat 0,0
    gcopy id%,x%,0,y%,80,3
    x%=x%-1 : y%=y%+1
    until x%=-1
  REM *****
  REM L'image est entièrement
  REM visible et va disparaître.
  REM *****
  do
    gscroll 1,0
    i%=i%+1
    until i%=239
    gupdate
  ENDP
```



seconde de l'horloge Système. Que dire de plus, sinon Bonne Chance !

A noter en plus des deux formules classiques du Loto, un tirage aléatoire de nombres.

```
PROC LOTO:
  GLOBAL T%,D%,N%,B%,T%
  DINIT
  DBUTTONS "LOTO"
  SIMPLE,%S,"ALEATOIRE",%A,"LOTO MULTIPLE",%M
  D%=DIALOG
  IF D%=97
  REM ***** ALEATOIRE
    DINIT
    DLONG B%,"MAXIMUM",0,50
    DLONG N%,"QUANTITE",1,30
    IF DIALOG
      T%=1 : T%=1
      IF B%=N%
        N%=N%-1
      ENDIF
      SORTIE:
      ENDIF
    ELSEIF D%=115
  REM ***** SIMPLE
    DINIT
    DLONG T%,"NOMBRE DE GRILLES",1,8
    IF DIALOG
      T%=2 : N%=6 : B%=49
      SORTIE:
      ENDIF
    ELSEIF D%=109
  REM ***** MULTIPLE
    DINIT
    DLONG N%,"NOMBRE DE CHIFFRES",7,10
    DLONG T%,"NOMBRE DE GRILLES",1,8
    IF DIALOG
      B%=49 : T%=3
      SORTIE:
      ENDIF
    ENDIF
  ENDP
```

```
PROC SORTIE:
  GLOBAL AL%(50),I%,NT%
  LOCAL V%
  RANDOMIZE (SECOND)
  REM *****
  REM LA BASE ALEATOIRE EST
  REM LA SECONDE DE L'HORLOGE
  REM *****
  DO
    NT%=NT%+1
  DO
    I%=I%+1
    AL%(I%)=INT(RND*B%)+1
    VERIF:
    UNTIL I%=N%+1
  ECRAN:
  I%=0
  UNTIL NT%=T%
  IF T%=3 AND T%>1
    AT 4,9
    PRINT "GRILLES MUTIPLES ";N%;" CHIFFRES"
  ENDIF
  GET
  ENDP
```

```
PROC VERIF:
  LOCAL V%
  IF I%>1
    DO
      V%=V%+1
      IF AL%(I%)=AL%(V%)
        I%=I%-1
        RETURN
      ENDIF
    UNTIL V%=I%-1
  ENDIF
  ENDP
```

```
PROC ECRAN:
  GSETWIN 0,0,190,80
  STATUSWIN ON
  I%=0
  IF T%=2
    PRINT"GRILLE ";NT%;" --> ";
    ENDIF
  IF T%=1 AND N%<11
    AT (34-(N%*3))/2,4
    ENDIF
  DO
    I%=I%+1
    IF (T%=1 AND I%=11) OR (T%=1 AND I%=21)
      PRINT :ENDIF
      PRINT RIGHT$("0"+FIX$(AL%(I%),0,2),2);"
    ";
    UNTIL I%=N%
  PRINT
  IF T%=1
    AT 13,6 :PRINT "TIRAGE"
    AT 12,7 :PRINT "ALEATOIRE"
    AT 14,8 :PRINT N%;" / ";B%
  ENDIF
  ENDP
```

Dernière heure

Voici les nouvelles dates des formations
OPL Organiseur II, HC ou Series 3 :

Introduction à l'OPL
8 Octobre, 16 Novembre, 14 Décembre

OPL intermédiaire
9 Octobre, 17 Novembre, 15 Décembre

OPL Avancé
19-20-21 Octobre, 18-19-20 Novembre,
16-17-18 Décembre

Contactez Maria Mariannie au
(1) 46 36 46 47 pour de plus amples
informations sur ces cours.

Quelle sortie pour vos impressions ?

Series 3

Voici deux petites routines OPL à intégrer dans vos programmes Series 3 pour ouvrir automatiquement le bon port d'impression configuré à partir de la Config imprimante de l'écran Système. En fait, le Series 3 utilise trois variables d'environnement pour stocker ces informations.

Les variables d'environnement sont des variables stockées dans une zone particulière de la mémoire et sont utilisables par tous les programmes.

La première des variables qui nous intéresse a pour nom P\$D. Elle contient le port à utiliser :

- 0 port parallèle
- 1 port série
- 2 fichier d'impression

La seconde des variables a pour nom P\$S. Elle contient les paramètres du port série.

La troisième et dernière, P\$F contient le nom de l'éventuel fichier d'impression. Par défaut il s'agit de P.LIS sur le disque interne à la racine (LOC::M:\P.LIS).

La procédure GETENV: recherche la variable dont le nom est stocké à l'adresse nom% et retourne son contenu dans le buffer à l'adresse buf%.

La procédure IMPROUT: quant à elle, ouvre le port de sortie automatiquement ou retourne un numéro d'erreur si l'ouverture n'a pu être réalisée.

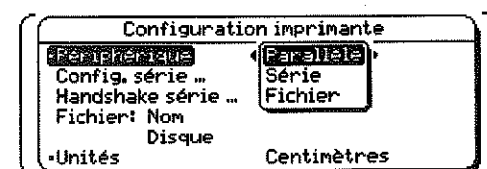
L'utilisation du port se fait ensuite à l'aide de l'instruction LPRINT et se ferme avec LCLOSE.

Attention, cette routine n'utilise pas le contrôleur d'imprimante pour réaliser les sorties, aussi certaines imprimantes peuvent

ne pas correctement réagir, comme par exemple les imprimantes POSTSCRIPT. Le contrôle de l'imprimante reste complètement à votre charge.

```
PROC improutp:
  REM Ouvre la sortie à utiliser
  LOCAL buf$(255),nom$(10),e%,a%,conf$(6)
  TRAP LCLOSE
  nom$="P$D"+CHR$(0)
  e%=getenv:(ADDR(nom$),ADDR(buf$))
  IF e%<0
    buf$="0"
    e%=0
  ENDIF
  IF buf$="0"
    TRAP LOPEN "PAR:A"
    IF ERR<>0:RETURN ERR:ENDIF
  ELSEIF buf$="1"
    TRAP LOPEN "TTY:A"
    IF ERR<>0:RETURN ERR:ENDIF
    nom$="P$S"+CHR$(0)
    e%=getenv:(ADDR(nom$),ADDR(buf$))
    IF e%<0
      POKEB ADDR(buf$),12
      a%=ADDR(buf$)+1:e%=IOW(-1,FSET,#a%,#0)
    ELSE
      e%=0
    ENDIF
  ELSEIF buf$="2"
    nom$="P$F"+CHR$(0)
    e%=getenv:(ADDR(nom$),ADDR(buf$))
    IF e%<0 OR e%>128
      buf$="LOC::M:\P.LIS"
    ENDIF
    TRAP LOPEN buf$
    IF ERR<>0:RETURN ERR:ENDIF
  ENDIF
ENDP
```

```
PROC getenv:(nom%,buf%)
  LOCAL a%(6),e%
  a%(1)=2100
  a%(4)=PEEKb(nom%)-1
  a%(5)=buf%+1
  a%(6)=nom%+1
  e%=OS($8b,ADDR(a%(1)))
  IF (e% AND 1)=0
    POKEB buf%,a%(1):RETURN
  ELSE
    RETURN PEEKb(ADDR(a%(1)))
  ENDIF
ENDP
```



Transfert de données

Series 3

Préalablement à toute manipulation, il convient de connecter le Series 3 au PC ou au Macintosh.

Sur le Series 3 :

- 1) Connectez le câble 3Link
- 2) Tapez PSION-L et fixez la liaison distante à "Actif"
- 3) Laissez la vitesse à 9600 bauds

Sur le PC

- 1) Connectez le câble 3Link
- 2) Lancez le programme McLink
- 3) Le double écran est affiché et, si "Link established" (Lien établi) n'apparaît pas en haut à gauche de la première fenêtre, tapez -P1 -B9600 sur le PC puis validez.

Les deux machines sont liées dès qu'apparaît "Link established" (Lien établi).

Sur le Macintosh

- 1) Branchez le câble 3Link sur la prise Modem
- 2) Lancez McLink
- 3) Un écran est affiché. Si "Lien établi" n'apparaît pas en haut à gauche de cet écran, vérifiez les paramètres de communication dans le menu "Options" du Macintosh : Vitesse 9600 bauds, Port Modem.

Les deux machines sont liées dès qu'apparaît "Lien établi". A partir de là, la communication entre les deux machines étant effective, les transferts de fichiers peuvent commencer. Il sera plus facile de travailler sur le Series 3, c'est en tout cas l'hypothèse de travail choisie ici.

Deux cas de figure concernant les transferts :

a) Nous désirons seulement sauvegarder les fichiers sur l'unité distante.

b) Nous désirons transférer les fichiers pour les utiliser sur l'unité distante.

Dans un premier temps, nous nous contenterons de les "Sauvegarder", (dans un prochain article nous reviendrons point par point sur le transfert pour utilisation). Il est à noter d'ailleurs que la sauvegarde ne nous empêche pas d'utiliser les fichiers mais, dans ce cas, nous nous privons des facilités offertes par le Series 3 (Formats, filtres RTF, fusion de fichiers, etc.)

La sauvegarde :

Placez-vous dans l'écran Système sur l'icône qui contient les fichiers à sauvegarder et appuyez sur TAB. Si ces fichiers ne sont pas rangés dans un sous-répertoire offrant une icône, placez-vous sur "Calc" et appuyez également sur TAB. Vous vous trouverez alors dans la racine, les flèches verticales vous aidant à trouver le répertoire adéquat.

Une fois dans ce répertoire, marquez le, ou les fichiers que vous désirez copier, en pressant simultanément MAJ et la flèche verticale basse. Ensuite, appuyez sur PSION/C qui vous affiche une boîte de dialogue. Sur la première ligne se trouvent les noms des fichiers à copier et sur la deuxième ligne, la localisation de ces fichiers. Passez directement sur la quatrième ligne et appuyez sur TAB. Vous avez alors accès aux différentes unités susceptibles de recevoir vos fichiers. REM:: est le nom de l'unité distante et sera suivi du nom du disque.

Lorsque ce choix est fait, remontez sur la troisième ligne et là encore, pressez la touche TAB pour choisir le répertoire de destination.

"Entrer" validera alors l'opération et les fichiers seront copiés sur l'ordinateur distant, ce qui sera vérifiable en quittant McLink par "EX" et en jetant un oeil dans les répertoires ou les dossiers.

Cette manipulation est très importante. Un entraînement n'est pas inutile, alors : à vos 3Link !

Permutation

Series 3

Toujours basé sur les images graphiques, un petit jeu de reconstitution. Le principe est simple : remettre l'image en ordre !

Au début de la partie, le curseur se trouve en haut à gauche de l'image. Le déplacer avec les flèches et "saisir" le carré à déplacer avec "Entrer". Ce carré se déplace lui aussi à l'aide des flèches et se "dépose" en appuyant sur la touche "P".

Si vous ne disposez pas encore d'une image, allez dans "Monde", et faites une copie d'écran en appuyant simultanément sur les touches Ctrl/Maj/Psion/S. L'image créée a pour nom "SCREEN.PIC" et se trouve dans la racine de l'unité interne.

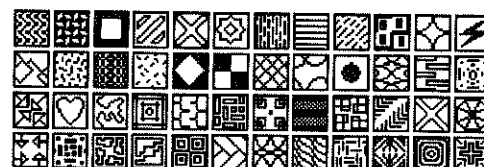
Bon courage !

```
PROC PERMUT:
LOCAL
FNOM$(128), P%, XN%, YN%, D%, XX%, YY%, G%, I%, X%, Y%
GLOBAL IS%, ID%
IS%=GCREATEBIT (50,20)
DINIT
DFILE FNOM$, "IMAGE : ", 0
IF DIALOG
ID%=GLOADBIT (FNOM$)
X%=GWIDTH : Y%=GHEIGHT
GUSE 1
GCOPY ID%, 0, 0, X%, Y%, 3
BROUILLE:
X%=0 : Y%=0
DO
GAT X%, Y%
GINVERT 20, 20
XX%=X% : YY%=Y%
G%=GET
IF G%=256 AND Y%>0 : Y%=Y%-20
ELSEIF G%=257 AND Y%<60 : Y%=Y%+20
ELSEIF G%=258 AND X%<220 : X%=X%+20
ELSEIF G%=259 AND X%>0 : X%=X%-20
ELSEIF G%=13 AND P%=0
D%=1 : P%=1
GUSE IS%
GAT 0, 0
GCOPY 1, X%, Y%, 20, 20, 3
GUSE 1
XN%=X% : YN%=Y%
GAT X%, Y%
```

```
GFILL 20, 20, 1
ELSEIF (G%=%P OR G%=%p) AND D%
GUSE IS%
GAT 30, 0
GCOPY 1, X%, Y%, 20, 20, 3
GINVERT 20, 20
GUSE 1
GAT XN%, YN%
GCOPY IS%, 30, 0, 20, 20, 3
GAT X%, Y%
GCOPY IS%, 0, 0, 20, 20, 3
D%=0 : P%=0
ENDIF
GAT XX%, YY%
GINVERT 20, 20
UNTIL G%=27
ENDIF
ENDP
```

```
PROC BROUILLE:
LOCAL I%, X%, XX%, Y%, YY%
DO
DO
X%=INT(RND*12)*20
Y%=INT(RND*4)*20
XX%=INT(RND*12)*20
YY%=INT(RND*4)*20
UNTIL ((X%<>XX%) AND (Y%<>YY%))
GUSE IS%
GCLS
GAT 0, 0
GCOPY 1, X%, Y%, 20, 20, 3
GAT 30, 0
GCOPY 1, XX%, YY%, 20, 20, 3
GUSE 1
GAT X%, Y%
GCOPY IS%, 30, 0, 20, 20, 3
GAT XX%, YY%
GCOPY IS%, 0, 0, 20, 20, 3
I%=I%+1
UNTIL I%=48
ENDP
```

Ci-dessous un exemple de ce que pourrait donner ce genre de casse-tête...



Liste des disques présents

Series 3

Le programme listé ci-contre permet de produire la liste de la totalité des noeuds et des disques présents et reconnus par le système d'exploitation de la machine.

Par exemple, vous pourrez vous rendre compte lorsque vous avez positionné l'option "Liaison distante" à "Actif", de la possibilité de percevoir la totalité des disques de votre ordinateur de bureau comme étant accessibles par votre Series 3.

La liste des noeuds est obtenue avec l'instruction IOOPEN sur le contrôleur 'FIL:' en donnant au drapeau la valeur \$60 (en hexadécimal).

La requête lit après cela sur le canal ouvert (Requête entrée/sortie) jusqu'à ce qu'un numéro d'erreur signale qu'il n'existe plus d'unité.

Vous pouvez effectuer un IOOPEN pour chacun des noeuds plutôt que d'ouvrir un fichier ou un répertoire, en utilisant la valeur \$50.

Les lectures suivantes sur **CE CANAL** donnent alors la liste des unités disponibles sur ce noeud.

La routine PATCHLEN permet d'écrire la longueur en octet du nom du disque repéré sur le premier octet de la variable alphanumérique DEV\$.

L'utilisation de ce logiciel est très simple. Lorsqu'il s'exécute, le logiciel affiche en tout premier lieu le nom du noeud sur lequel il se trouve. Le premier trouvé est obligatoirement le noeud LOC:: puisque celui-ci correspond aux disques locaux (loc::=local - ABC & M).

Le deuxième noeud est appelé ROM::. Il n'est accessible qu'en lecture. C'est en effet sur ce disque qu'est stocké le système d'exploitation et les différents logiciels que vous utilisez habituellement (Fiche, Texte, Agenda, etc...).

Le troisième et dernier noeud possible est nommé REM:: (REM::=REMOTE=distant). Ce noeud ne peut être présent qu'à la condition de disposer de la liaison RS232 (3Link), du logiciel McLink et que l'option "Liaison distante" soit à "Actif". Les disques reconnus alors par votre Series 3, seront ceux reconnus par votre ordinateur de bureau.

Nous vous laissons imaginer maintenant les applications qui pourraient tirer partie de cette information.

```
PROC devlist:
local nh%
local dh%
local dev$(128)
local ret%
ioopen(nh%, "FIL:", $60)
while 1
ret%=iow(nh%, 1, #addr(dev$)+1, #0)
if ret%
break
endif
patchlen: {addr(dev$)}
print "Unité:", dev$
ret%=ioopen(dh%, dev$, $50)
if ret%
print " Noeuds non supportés"
continue
endif
while 1
ret%=iow(dh%, 1, #addr(dev$)+1, #0)
if ret%
break
endif
patchlen: {addr(dev$)}
print " Disque:", dev$
endwh
ioclose(dh%)
endwh
ioclose(nh%)
get
ENDP

PROC patchlen: (a%)
pokeb a%, call($b9, 0, 0, 0, 0, a%+1)
ENDP
```

La version de votre machine

Organiseur II

L'utilitaire listé ci-dessous permet d'identifier automatiquement le type de machine dont vous disposez ainsi que le numéro de version de votre système d'exploitation. Cela dit, bien que très utile, ce programme ne représente pas à proprement parlé un utilitaire incontournable.

Mais la partie la plus intéressante consiste à montrer comment réaliser un programme multi-machines. C'est-à-dire un programme qui fonctionne sans aucune modification, à la fois sur des machines de deux lignes d'affichage et sur des machines de quatre lignes d'affichage.

Réaliser une application multi-machine nécessite une analyse particulièrement fine de l'application.

En effet, il est important de savoir que seul l'interface utilisateur - donc les routines affichant des informations sur l'écran de la machine (menu, saisie, informations diverses) - devra être compilée en fonction de la machine. Les routines n'utilisant pas de commande liée à l'affichage, devront être compilées en mode deux lignes (CM, XP, POS 200/250/296/350 et AlphaPos).

C'est votre application qui, en fonction du type de machine employé, sélectionnera les routines correspondantes.

Pour conclure, votre programme contiendra en fait deux applications distinctes. Une, pour la version deux lignes et une, pour la version quatre lignes.

```
xplz:
rem *****
rem * Gilles JEAN - AWARE 15/06/1990
rem *****
rem Cette routine permet de tester
rem la machine en cours et
rem de forcer le mode
rem 4 lignes si c'est un LZ.
rem Elle doit être compilée
rem en mode 2 lignes.
rem *****
local fl1%
local fl2%
local mc%(2)
if peekb($ffe8) and 8
  fl1%=1
  if peekb($ffcb) and 128
    fl2%=1
  endif
endif
if fl1% and fl2%
  mc%(1)=$3f82
  mc%(2)=$3900
  usr(addr(mc%()),$100)
  print"Machine: LZ",
  if peekb($ffe8)=13 and
  peekb($ffcb)=192
    print"64"
  else
    print
  endif
  print"Version:",
  print
  left$(hex$(peekb($ffe9)),1);". ";right$(h-
  ex$(peekb($ffe9)),1)
  else
    print"Machine:",
    if peekb($ffe8)=0
      print"CM"
    else
      print"XP"
    endif
    print"Version:",
    print
    left$(hex$(peekb($ffe9)),1);". ";right$(h-
    ex$(peekb($ffe9)),1)
  endif
endif
get
```

Fonctions ASIN & ACOS

Organiseur II

OPL ne dispose pas, aussi bien sur les modèles CM/XP que sur la gamme LZ/LZ64, des fonctions trigonométriques ASIN et ACOS.

Les deux programmes fournis ci-dessous permettent de remédier à ce tout petit problème.

Ces fonctions, car nous pouvons les nommer de la sorte, sont aussi utilisables à l'intérieur de l'option CALC de votre machine en utilisant la syntaxe suivante :

Calc:5.12345*(acos:(2))

```
ASIN:(a1)
if abs(a1)=1
  return a1*pi/2
elseif abs(a1)>1
  cls
  print"Pas possible"
  pause 20
  return
else
  return deg(atan(a1/sqr(1-a1*a1)))
endif

ACOS:(a1)
if a1=1
  return 0
elseif a1=-1
  return pi
elseif abs(a1)>1
  cls
  print"Pas possible"
  pause 20
  return
else
  return deg(1.570796-atan(a1/sqr(1-a1*a1)))
endif
```

Mastermind

Organiseur II

Un petit jeu vous permettant de vous détendre sur le chemin qui vous ramène à la maison (à proscrire dans le cas où vous rentrez chez vous au volant de votre automobile).

Basé sur le principe bien connu du célèbre Mastermind, ce jeu est simple à jouer et très performant. Le principe en est simple. L'Organiseur II choisit 4 chiffres parmi dix possibles. A vous de les retrouver. Pour vous aider, la machine vous indique le nombre de chiffres bien placés et le nombre de chiffres mal placés.

```
mm:
GLOBAL t$(16,16)
GLOBAL a%(4),c%(4),d%(4),g%,i%,j%
global k%,m%,s%,t%,num%,res1%,res2%
rem
rem Auteur : Jonathan Hurwitt V2.12
rem Traduction - Aware 1990
rem
ONERR prterr::
CLS
view(1,"MASTERMIND - 1990")
g%=0
s%=0
t%=0
newgame::
```

```
KSTAT 3
i%=INT((RND+RND+RND)*3333)
i%=i%-(i%/10000)*10000
num%=i%
d%(1)=i%/1000
i%=i%-d%(1)*1000
d%(2)=i%/100
i%=i%-d%(2)*100
d%(3)=i%/10
d%(4)=i%-d%(3)*10
m%=1
CLS
t$(1)=" NB ESSAI B M"
AT 1,2
PRINT t$(1);
DO
  m%=m%+1
  i%=1
  j%=m%
  t$(m%)=""
  "+CHR$(m%-1)/10+48)+CHR$(m%-1-(m%-
  1)/10)*10+48)+" "
DO
  CLS
  PRINT t$(j%-1);
  PRINT t$(j%);
  IF j%<m%
    CURSOR off
  ELSE
    CURSOR on
  ENDIF
  k%=GET
  IF k%=3 AND j%>2
    j%=j%-1
  ELSEIF k%=4 AND j%<m%
    j%=j%+1
  ELSEIF k%>47 AND k%<58
    j%=m%
    a%(i%)=k%-48
```

```

t$(m%)=t$(m%)+CHR$(k%)
i%=i%+1
ELSEIF k%=8 AND i%>1
j%=m%
i%=i%-1
k%=LEN(t$(m%))
t$(m%)=LEFT$(t$(m%),k%-1)
ELSEIF k%=1
CLS
PRINT" Fin du jeu"
BEEP 750,1500
GOTO label::
ELSE
BEEP 15,300
ENDIF
UNTIL i%>4
c%(1)=d%(1)
c%(2)=d%(2)
c%(3)=d%(3)
c%(4)=d%(4)
i%=1
res1%=0
res2%=0
DO
IF a%(i%)=c%(i%)
res1%=res1%+1
a%(i%)=-1
c%(i%)=-2
ENDIF
i%=i%+1
UNTIL i%>4
IF res1%<4
i%=1
DO
j%=1
DO
IF a%(i%)=c%(j%)
res2%=res2%+1
a%(i%)=-1
c%(j%)=-2
ENDIF
j%=j%+1
UNTIL j%>4
i%=i%+1
UNTIL i%>4
ENDIF
DO
AT 1,2
t$(m%)=t$(m%)+ " "+CHR$(res1%+48)+ "
"+CHR$(res2%+48)
print t$(m%);
PAUSE 10
UNTIL res1%=4 OR m%=16
k%=key
PAUSE 15
CLS
t%=t%+1
s%=s%+m%-1
IF res1%=4
g%=g%+1
PRINT m%-1,"Essai(s)"
i%=2000
DO
BEEP 25,i%
i%=i%-100
UNTIL i%<100
ELSE
cls
PRINT"Je gagne !!!"
print num%;
BEEP 1000,2000
ENDIF
label::
KSTAT 1
PRINT"Encore? (O/N)";
IF get=%0:
GOTO newgame::
ENDIF
CURSOR off
IF t%>0
CLS
view(1,"Vous avez joue "+GEN$(t%,3)+" fois
et gagne
"+GEN$(g%,3)+" fois. Score final
"+FIX$(FLT(s%/t%),2,5)+"." )
CLS
ENDIF
RETURN
prterr::
CLS
PRINT ERR$(ERR)
BEEP 1000,2000
GOTO label::

```

Appeler l'option Comms avec un POS Organiseur II

C'est à nos développeurs que s'adresse cet article. En effet, vous êtes nombreux à nous questionner régulièrement sur la possibilité d'utiliser des fonctions standards d'un Organiseur II avec un POS. Pour rappel ou pour information, un POS est en fait un runtime de l'Organiseur II. C'est physiquement la même machine à ceci près qu'elle n'est faite que pour exécuter une application OPL se trouvant sur un Datapak. Et pourtant...

En fait, la Rom du POS (elle contient le système d'exploitation) dispose des applications classiques - Agenda, Calc,

Rech, Sauv, etc. Seule manque à l'appel l'interface utilisateur nécessaire.

Le programme listé ci-contre vous montre comment lancer l'option Comms de la liaison RS232 de l'Organiseur II.

Attention toutefois, car la moindre erreur de saisie de ce programme entraînerait la perte irrémédiable de la totalité des informations de la mémoire vive de votre machine (unité A).

```

BOOT:
LOCAL A%(2)
A%(1)=$3F17
A%(2)=$3900
USR(ADDR(A%()),0)
CALLCOMM:

```

```

BEEP 300,300
BEEP 200,800
BEEP 300,300

```

```

CALLCOMM:
local a%(55),calcadd%
a%(1)=$0118
a%(2)=$dca5
a%(3)=$b320
a%(4)=$4018
a%(5)=$8c03
a%(6)=$e82b
a%(7)=$1cfe
a%(8)=$200c
a%(9)=$bc20
a%(10)=$0e26
a%(11)=$14bc
a%(12)=$2010
a%(13)=$260f
a%(14)=$bc20
a%(15)=$1226
a%(16)=$0abc
a%(17)=$2014
a%(18)=$2605
a%(19)=$bc20
a%(20)=$1627
a%(21)=$0139
a%(22)=$dea9
a%(23)=$3cfe
a%(24)=$2065
a%(25)=$3cde
a%(26)=$a53c
a%(27)=$fff20
a%(28)=$65de
a%(29)=$a73c
a%(30)=$deb6
a%(31)=$3cfe
a%(32)=$23e2
a%(33)=$3cde
a%(34)=$b03c
a%(35)=$deb2
a%(36)=$3cde
a%(37)=$b43c
a%(38)=$18ad
a%(39)=$0038
a%(40)=$deb4
a%(41)=$38df
a%(42)=$b238
a%(43)=$dfb0
a%(44)=$38ff

```

```

a%(45)=$23e2
a%(46)=$38df
a%(47)=$b638
a%(48)=$dfa7
a%(49)=$38df
a%(50)=$a538
a%(51)=$fff20
a%(52)=$6538
a%(53)=$dfa9
a%(54)=$3f46
a%(55)=$3900
calcadd%=findtop%:
BEEP 99,99
usr(addr(a%()),calcadd%)
BEEP 99,99

```

```

FINDTOP%:
local calcadd%,i%,a%,n%,s$(6),o$(5)
i%=peekw($2002)
do
a%=peekb(i%)
if a%=0
return 0
endif
if a%<>5
i%=i%+a%+3
continue
endif
s$=""
n%=1
do
s%=s%+chr$(peekb(i%+n%))
n%=n%+1
until n%=6
if peekb($2186)<2
o$="COMMS"
elseif peekb($2186)=2
o$="COMMS"
else stop
endif
if upper$(s%)<>o$
i%=i%+a%+3
continue
endif
calcadd%=i%+a%+1
calcadd%=peekw(calcadd%)
return(calcadd%)
until 0

```

Caractères accentués - XP, CM & POS Organiseur II

Les caractères accentués sur l'Organiseur II n'ont fait leur apparition qu'à partir de la deuxième génération de machine - à savoir lorsque sont sortis les modèles LZ et LZ64. Toutefois, lorsqu'ils sont nécessaires au bon fonctionnement d'une application, il est possible de les générer.

```

testcar:
rem *****
rem Date : 24 Juin 1991
rem Objet : caracteres àéèêô
rem *****
local a%
testgdu:(0,8,4,14,1,15,17,15,0):rem à
testgdu:(1,2,4,14,17,31,16,14,0):rem é
testgdu:(2,8,4,14,17,31,16,14,0):rem è
testgdu:(3,4,10,14,17,31,16,14,0):rem ê
testgdu:(4,4,10,0,14,17,17,14,0):rem ô

```

```

do
print chr$(a%),
a%=a%+1
until a%>4
get

testgdu:(x%,a%,b%,c%,d%,e%,f%,g%,h%)
rem *****
rem Auteur : Gilles JEAN
rem Date : 24 Juin 1991
rem Objet : caracteres àéèêô
rem *****
pokeb $180,64+x%*8
pokeb $181,a%
pokeb $181,b%
pokeb $181,c%
pokeb $181,d%
pokeb $181,e%
pokeb $181,f%
pokeb $181,g%
pokeb $181,h%

```

Les dernières versions

Series 3

Series 3 - 128 Ko	Ver 1.80
Series 3 - 256 Ko	Ver 1.80
3 LINK	Ver 1.00
PARALLEL LINK	Ver 1.00
MCLINK PC	Ver 3.00
SLINK PC	Ver 2.00
MCPRINT	Ver 1.10
TABLEUR	Ver 1.03
THEOLE	Ver 1.00
3 BANQUE	Ver 1.00

Organiseur II

XP	Ver 3.6
POS200	Ver 3.6
AlphaPOS	Ver 3.6
P250	Ver 3.6
P350	Ver 3.8
P296	Ver 3.8
CM MULTI-LANGUES	Ver 3.7
XP MULTI-LANGUES	Ver 3.7
LZ 32K	Ver 4.6
LZ 64K	Ver 4.6
P432	Ver 4.6
P464	Ver 4.6
POS432	Ver 4.6
POS 464	Ver 4.6
COMMS LINK	Ver 4.2
CL DISK	Ver 2.1
BARCODE READER	Ver 3.2
SWIPE READER	Ver 2.0
PRINTER II PSION	Ver 2.2
FILEPAK (XP)	Ver 1.4
FILEPAK (LZ/LZ64)	Ver 2.0
TOP FINANCE (XP)	Ver 2.2
TOP FINANCE (LZ/LZ64)	Ver 2.3
BARPAK	Ver 0.9
SPELLING CHECKER	Ver 2.0
SPREADSHEET	Ver 3.4
2/4 FORMULATOR	Ver 2.1
TRAVEL PAK	Ver 2.1
FINANCE PACK II	Ver 2.1
GAMES PACK	Ver 1.0
THESAURUS AND SPELLING CHECKER	Ver 1.0
FLASH DRIVER	Ver 1.7

Le contenu du présent document, composé sur PC et WORD Windows, donne l'information la plus complète et la plus exacte possible au moment de la publication, mais n'est garantie ni quant à la complétude, ni quant à l'exactitude. Aware, Psion, Word, Windows, Dos sont des marques déposées.