

# OBJET.BIN

.COPYRIGHT 1985 MICRO-APPLICATION.  
DAMS.

```
ORG #a000
DI ;ROUTINE DE LECTURE
LD de,#a000
LD hl,#Efff
LD bc,#fb7f
CALL escrit
CALL read
EI
RET
DI ;ROUTINE D'ECRITURE
LD hl,#Efff
LD bc,#fb7f
CALL escrit
CALL write
EI
RET
read
loop1
DEC c
IN a,(c)
JP p,loop1
AND #20
RET z
INC c
IN a,(c)
LD (hl),a
INC hl
LD a,h
CP d
RET z
JR read
write
loop2
DEC c
IN a,(c)
JP p,loop2
AND #20
RET z
INC c
LD a,(hl)
OUT (c),a
INC hl
JR write
ecrit
LD a,(hl)
OUT (c),a
```

```
INC hl
LD a,#05
att DEC a
NOP
JR nz,att
RET
```

Text:#68F5  
Hmem:#8D39

End:#6B14

# 21F Bytes

# MONIDISC.BAS

```
10 ***** >LA
20 '* * >LB
30 '* P.ARRIBAT MONIDISC NOVEMBRE 1987 * >LC
40 '* * >LD
50 '***** >LE
60 MEMORY &6FFE:LOAD"objet.bin",&A000 >JT
70 MODE 2:PAPER 0:PEN 1:INK 0,0:INK 1,26:BORDER 0 >QQ
80 DIM m$(18),p$(48),p(48),i$(16),i(16),s$(16),s(16),h( >ZP
27)
90 FOR x=1 TO 18:READ x$:m$(x)=x$:NEXT:FOR x=1 TO 48:RE >ZW
AD x$:p$(x)=x$:NEXT
100 FOR x=1 TO 48:READ y:p(x)=y:NEXT:FOR x=1 TO 16:READ >XD
x$:i$(x)=x$:NEXT
110 FOR x=1 TO 16:READ y:i(x)=y:NEXT:FOR x=1 TO 16:READ >XC
x$:s$(x)=x$:NEXT
120 FOR x=1 TO 16:READ y:s(x)=y:NEXT >EQ
130 FOR x=1 TO 13:h(x*2-1)=x+64:h(x*2)=x+69:NEXT >RJ
```

```

140 WINDOW#1.9.21.1.1:PEN#1.0:PAPER#1.1:WINDOW#2.22.40. >GH
25.25
150 d=&FB7F:c=0:l=4:t=0:b=2:e=9:f=82:g=&E5:j=0:k=12:n=3 >PL
2:o=16:p=0
160 r=0:s=65:u=s:v=42:w=b:zz=0 >AC
170 CLS:LOCATE 30,1:PRINT"MONITEUR de DISQUETTE" >XF
180 FOR x=1 TO 18 STEP 2 >QF
190 LOCATE 3,x+4:PRINT USING"##":x::PRINT" - ":m$(x) >XF
200 LOCATE 47,x+4:PRINT USING"##":x+1::PRINT" - ":m$(x+ >KM
1):NEXT
210 LOCATE 1,25:PRINT"PRET POUR COMMANDE : " >PN
220 CLS#2:INPUT#2,"",a$:a=VAL(a$) >BK
230 IF LEN(a$)<1 OR LEN(a$)>2 OR a<1 OR a>18 THEN 260 >RR
240 IF a<11 THEN ON a GOTO 270,300,330,350,370,390,410, >HC
430,620,650
250 ON a-10 GOTO 670,690,710,760,840,900,1040,1110 >PF
260 LOCATE 60,25:PRINT"COMMANDE INVALIDE":GOTO 220 >WE
270 REM Arret du moteur >TC
280 INK 1.0:FOR x=0 TO 50:NEXT:OUT(&FA7E),0:SOUND 1,300 >BY
.5
290 INK 1,26:GOTO 220 >NG
300 REM Moteur en marche >TT
310 INK 1.0:FOR x=0 TO 50:NEXT:OUT(&FA7E),1:SOUND 1,300 >BT
.5
320 INK 1,26:GOTO 220 >NA
330 REM Lire des donnees >TM
340 CLS:i=70:GOSUB 1330:GOSUB 1840:CALL &A000:GOSUB 169 >LQ
0:GOTO 170
350 REM Ecrire des donnees >VC
360 CLS:i=69:GOSUB 1330:GOSUB 1840:CALL &A012:GOSUB 169 >LE
0:GOTO 170
370 REM Lire donnees effacees >YM
380 CLS:i=78:GOSUB 1330:GOSUB 1840:CALL &A000:GOSUB 169 >LD
0:GOTO 170
390 REM Ecrire donnees effacees >AN
400 CLS:i=73:GOSUB 1330:GOSUB 1840:CALL &A012:GOSUB 169 >LU
0:GOTO 170
410 REM Lecture d'une piste >MC
420 CLS:i=66:GOSUB 1330:GOSUB 1840:CALL &A000:GOSUB 169 >LV
0:GOTO 170
430 REM Formater une piste >WK
440 CLS:i=77:GOSUB 1330:GOSUB 1490:t=1 >GK
450 LOCATE 17,18:PRINT"Taille secteur (0-5) ";USING >TC
"[#]: ";b
460 ll=18:cc=47:xx=5:x=0:y=b:GOSUB 2010:b=y:OUT d,b:GOS >KQ
UB 1490
470 LOCATE 17,19:PRINT"Secteur / piste (1-27) ";USING >RB
"[#]: ";e
480 ll=19:xx=27:x=1:y=e:GOSUB 2010:e=y:OUT d,e:GOSUB 14 >DM
90
490 LOCATE 17,20:PRINT"Nb octets GAP 3 (0-255)";USING"[ >QR
###": ";f
500 ll=20:xx=255:x=0:y=f:GOSUB 2010:f=y:OUT d,f:GOSUB 1 >EE
490
510 LOCATE 17,21:PRINT"Octet formatage (0-255)";USING"[ >UG
###": ";g
520 ll=21:x=0:y=g:GOSUB 2010:g=y:POKE &6FFF,g:GOSUB 149 >CU
530 m=&7000:zz=1:x=0:LOCATE 1,21:PRINT"No secteurs":LOC >QN
ATE 1,22
540 PRINT"(0-255)":FOR z=0 TO e-1:cc=8+INT(z/3)*8:ll=22 >RB
+z-INT(z/3)*3
550 LOCATE cc,11:PRINT"[ ";HEX$(h(z+1),2):"]":cc=cc+5:x >FQ
=0:y=h(z+1):GOSUB 2010
560 h(z+1)=y:POKE m+1,0:POKE m+2,h(z+1):POKE m+3,b:m=m+ >QY
4:NEXT:zz=0
570 LOCATE 54,25:PRINT"No de piste (0-41)";USING"[##]: >JX
";j;
580 ll=25:cc=77:xx=41:x=0:v=j:GOSUB 2010:j=y >FR
590 FOR x=0 TO e-1:POKE &7000+4*x,j >BX
600 IF b=0 THEN POKE &7003+4*x,127 >ZE
610 NEXT:CALL &A012:t=0:GOSUB 1690:GOTO 170 >LY
620 REM Lire ID >ZG
630 CLS:i=74:GOSUB 1330 >RD
640 CLS#1:PRINT#1," EXECUTION":FOR x=0 TO 900:NEXT:GOSU >VK
B 1690:GOTO 170
650 REM Scan equal >MG
660 CLS:i=81:GOSUB 1330:GOSUB 1840:CALL &A012:GOSUB 169 >LB
0:GOTO 170
670 REM Scan low or equal >UX
680 CLS:i=89:GOSUB 1330:GOSUB 1840:CALL &A012:GOSUB 169 >LM
0:GOTO 170
690 REM Scan high or equal >UA
700 CLS:i=93:GOSUB 1330:GOSUB 1840:CALL &A012:GOSUB 169 >LZ
0:GOTO 170
710 REM Chercher piste 0 >TZ
720 CLS:i=7:GOSUB 1430:CLS#1:PRINT#1," EXECUTION" >WQ
730 FOR x=0 TO 500:NEXT:CLS#1:PRINT#1," RESULTAT":GOSU >JW
B 1610
740 j=INP(d):LOCATE 60,18:PRINT"No de piste : ";USING"# >MC
##":j;
750 PRINT" #";HEX$(j,2):GOSUB 1810:GOTO 170 >LA
760 REM Chercher piste No >UR
770 CLS:i=15:GOSUB 1430:GOSUB 1490 >CL
780 LOCATE 17,18:PRINT"No de piste (0-41) ";USING"[##]: >JX
";j;
790 ll=18:cc=42:xx=41:x=0:y=j:GOSUB 2010:j=y:OUT d,j >YX
800 CLS#1:PRINT#1," EXECUTION":FOR x=0 TO 700:NEXT >VA
810 CLS#1:PRINT#1," RESULTAT":GOSUB 1610 >KZ
820 j=INP(d):LOCATE 60,18:PRINT"No de piste : ";USING"# >MB
##":j;
830 PRINT" #";HEX$(j,2):GOSUB 1810:GOTO 170 >KH
840 REM Lire etat lecteur >VA
850 CLS:LOCATE 41,1:PRINT UPPER$(m$(a)):GOSUB 1140 >VD
860 GOSUB 1490:CLS#1:PRINT#1," INSTRUCTION" >NF
870 LOCATE 13,18:PRINT"#04":OUT d,4:GOSUB 1490 >PT
880 LOCATE 13,19:PRINT"#00":OUT d,0:CLS#1:PRINT#1," RE >KV
SULTAT"
890 GOSUB 1490:GOSUB 1550:GOSUB 1810:GOTO 170 >MX
900 REM Indiquer donnees lecteur >CX
910 CLS:LOCATE 41,1:PRINT UPPER$(m$(a)):GOSUB 1150 >VB
920 GOSUB 1540:GOSUB 1490:CLS#1:PRINT#1," INSTRUCTION" >BL
930 LOCATE 13,18:PRINT"#03":OUT d,3:GOSUB 1490 >PM
940 LOCATE 17,18:PRINT"Step rate (2-32) ";USING"[##]: >HK

```

```

";k
950 ll=18:cc=41:xx=32:x=2:y=k:GOSUB 2010:k=y >PP
960 LOCATE 17,19:PRINT"Off head (0-480)";USING"[###]: >BT
";n
970 ll=19:xx=480:x=0:y=n:GOSUB 2010:n=y >JT
980 OUT d,VAL("&x"+BIN$(INT((32-k)/2),4)+"0000")+INT(n/ >DE
32):t=1:GOSUB 1490:t=0
990 LOCATE 17,20:PRINT"On head (4-512)";USING"[###]: >FE
";o
1000 ll=20:xx=512:x=0:y=o:GOSUB 2010:o=y >KA
1010 LOCATE 17,21:PRINT"Mode DMA (0-1) ";USING"[#]: >FZ
";p
1020 ll=21:xx=1:x=0:y=p:GOSUB 2010:p=y >HC
1030 OUT d,INT(o/4)-1+p*128:GOSUB 1810:GOTO 170 >PY
1040 REM ? >GK
1050 CLS:LOCATE 31,1:PRINT"BUFFER DE &7000 a &9FFF (12 >RT
Ko)":y=&7000
1060 INPUT"Nb de secteur (1-24) : ",q:INPUT"Taille sec >GG
teur ( 0-5 ) : ",b
1070 FOR x=1 TO q:PRINT"Secteur No : ";x;" ";HEX$( >JM
y);"-";HEX$(y+128*2^b-1)
1080 FOR y=&7000+(x-1)*128*2^b TO &7000+x*128*2^b-1 >RD
1090 PRINT HEX$(PEEK(y),2);" ";:NEXT:PRINT:NEXT >VY
1100 CALL &BB18:GOTO 170 >RU
1110 REM reinitialise >RR
1120 FOR x=&7000 TO &7000+&2FFF:POKE x,0:NEXT:GOTO 150 >XT
1130 REM s/p cadre ETAT PRINCIPAL et DRIVE STATUS >UD
1140 LOCATE 1,19:PRINT"Inst No 2:" >EP
1150 LOCATE 1,18:PRINT"Inst No 1:":LOCATE 1,1:PRINT"Pha >LK
se : "
1160 CLS#1:PRINT#1," PREPARATION" >CU
1170 PLOT 0,374:DRAW 639,374:DRAW 639,330:DRAW 0,330:DR >GU
AW 0,374
1180 PLOT 60,330:DRAW 60,374:LOCATE 3,3:PRINT"Etat":LOC >FT
ATE 2,4:PRINT"Princ"
1190 PLOT 156,330:DRAW 156,374:PLOT 244,330:DRAW 244,37 >PZ
4:PLOT 348,330
1200 DRAW 348,374:PLOT 444,330:DRAW 444,374:PLOT 0,182: >MX
DRAW 639,182
1210 DRAW 639,136:DRAW 0,136:DRAW 0,182:PLOT 60,136:DRA >GF
W 60,182
1220 PLOT 0,182:DRAW 639,182:DRAW 639,136:DRAW 0,136:DR >GN
AW 0,182
1230 LOCATE 2,15:PRINT"Drive":LOCATE 2,16:PRINT"Status" >GM
";k
1240 PLOT 124,136:DRAW 124,182:PLOT 220,136:DRAW 220,18 >PY
2:PLOT 308,136
1250 DRAW 308,182:PLOT 388,136:DRAW 388,182:PLOT 460,13 >PW
6:DRAW 460,182
1260 PLOT 532,136:DRAW 532,182:RETURN >EW
1270 REM s/p cadre ETAT INTERRUPTION >FP
1280 PLOT 0,326:DRAW 639,326:DRAW 639,282:DRAW 0,282:DR >GZ
AW 0,326
1290 PLOT 60,282:DRAW 60,326:LOCATE 3,6:PRINT"Etat":LOC >FL
ATE 2,7:PRINT"Inter"
1300 PLOT 180,282:DRAW 180,326:PLOT 292,282:DRAW 292,32 >PC
6:PLOT 364,282

```

```

1310 DRAW 364,326:PLOT 460,282:DRAW 460,326:PLOT 532,28 >BZ
2:DRAW 532,326:RETURN
1320 REM s/p cadre ETAT ST1 et ST2 >AN
1330 PLOT 0,278:DRAW 639,278:DRAW 639,234:DRAW 0,234:DR >GH
AW 0,278
1340 PLOT 60,234:DRAW 60,278:LOCATE 3,9:PRINT"Reg":LOCA >AX
TE 3,10:PRINT"ST1"
1350 PLOT 156,234:DRAW 156,278:PLOT 188,234:DRAW 188,27 >QT
8:PLOT 292,234
1360 DRAW 292,278:PLOT 380,234:DRAW 380,278:PLOT 412,23 >PU
4:DRAW 412,278
1370 PLOT 484,234:DRAW 484,278:PLOT 580,234:DRAW 580,27 >ZH
8
1380 PLOT 0,230:DRAW 639,230:DRAW 639,186:DRAW 0,186:DR >GM
AW 0,230
1390 PLOT 60,186:DRAW 60,230:LOCATE 3,12:PRINT"Reg":LOC >BR
ATE 3,13:PRINT"ST2"
1400 PLOT 92,186:DRAW 92,230:PLOT 156,186:DRAW 156,230: >MX
PLOT 236,186
1410 DRAW 236,230:PLOT 324,186:DRAW 324,230:PLOT 404,18 >PJ
6:DRAW 404,230
1420 PLOT 500,186:DRAW 500,230:PLOT 580,186:DRAW 580,23 >YA
0
1430 LOCATE 41,1:PRINT UPPER$(m$(a)):GOSUB 1140:GOSUB 1 >DX
280
1440 GOSUB 1540:GOSUB 1490 >UC
1450 CLS#1:PRINT#1," INSTRUCTION" >DU
1460 LOCATE 11,18:PRINT"#";HEX$(i,2):OUT d,i:GOSUB 1490 >ZY
";k
1470 LOCATE 11,19:PRINT"#00":OUT d,0:RETURN >MA
1480 REM LIT et AFFICHE ETAT PRINCIPAL et ST1 et ST2 >VD
1490 PEN 0:PAPER 1:e$=BIN$(INP(d-1),8) >FG
1500 FOR x=0 TO 7:f$=MID$(e$,x+1,1) >BF
1510 LOCATE p(2*x+1+VAL(f$)+c),1:PRINT p$(2*x+1+VAL(f$) >DF
+c)
1520 NEXT:PAPER 0:PEN 1:IF t=0 THEN CALL &BB18:RETURN E >MG
LSE RETURN
1530 REM LIT et AFFICHE DRIVE STATUS >EJ
1540 OUT d,4:OUT d,0 >NP
1550 PEN 0:PAPER 1:e$=BIN$(INP(d),8) >DM
1560 FOR x=0 TO 5:f$=MID$(e$,x+1,1) >BU
1570 LOCATE s(2*x+1+VAL(f$)),16:PRINT s$(2*x+1+VAL(f$)) >GY
:NEXT
1580 LOCATE 70,16:PRINT s$(13+VAL(MID$(e$,7,2))) >QB
1590 PAPER 0:PEN 1:RETURN >UD
1600 REM LIT et AFFICHE ETAT INTERRUPTION >LZ
1610 OUT d,8:GOSUB 1490 >RH
1620 PEN 0:PAPER 1:e$=BIN$(INP(d),8) >DK
1630 LOCATE 9,7:PRINT i$(1+VAL("&x"+MID$(e$,1,2))) >TG
1640 FOR x=2 TO 5:f$=MID$(e$,x+1,1) >BV
1650 LOCATE i(2*x+1+VAL(f$)),7:PRINT i$(2*x+1+VAL(f$)): >EB
NEXT
1660 LOCATE 70,7:PRINT i$(13+VAL("&x"+MID$(e$,7,2))) >VQ
1670 PEN 1:PAPER 0:GOSUB 1490:RETURN >EQ
1680 REM LIT et AFFICHE 7 RESULTATS >DL
1690 CLS#1:PRINT#1," RESULTAT":GOSUB 1490:GOSUB 1620 >YZ
1700 d=d+1:l=10:c=16:t=1:GOSUB 1490:d=d-1:l=4:c=0:t=0:G >LQ

```

```

POP DE      |RECUPERER POINT RECEPTION - ICI SOURCE = SOURCE + 80
CALL CLOSEBANK
PUSH HL
CALL DESTOCK
POP HL
JR LOAD1

FINLOAD PUSH DE      |SORTIE DE ROUTINE - LA LONGUEUR A TRANSFERER EST<80
CALL OPENBANK        |COMME CI-DESSUS MAIS SAUTE AUX LABELS PERMETTTANT UN
CALL VSTOCK1         |TRANSFERT INFERIEUR A 80 OCTETS DE LONG
CALL CLOSEBANK
POP DE
JP DSTOCK1          |DERNIER TRANSFERT ET RETOUR

|
|*** ROUTINE SAVE ***
|
SAVE          EX DE,HL      |INVERSER SOURCE ET RECEPTION
SAVE1

CALL TESTLEN
JR C,FINSAVE
PUSH DE
CALL VERSTOCK
POP DE
PUSH HL
CALL OPENBANK
CALL DESTOCK
CALL CLOSEBANK
POP HL
JR SAVE1

FINSAVE PUSH DE
CALL VSTOCK1
CALL OPENBANK
POP DE
CALL DSTOCK1
JP CLOSEBANK

|
|*****
|*** SUB ROUTINES ***
|*****
|
|*** OUVRIR FERMER BANQUE ***
|
CLOSEBANK PUSH AF      |PRESERVER SELECTION BANQUE
LD A,192              |CODE DE FERMETURE
JR BANKOP             |SAUTER SUR OPERATION OUT

OPENBANK PUSH AF      |A CAUSE DE POP EN SORTIE

BANKOP PUSH BC        |PRESERVER LE COMPTEUR DE TRANSFERT
LD BC,#7F00
OJT (C),A            |FERMER LA BANQUE
POP BC
POP AF
RET                  |RECUPERER SELECTION BANQUE

|
|*** TESTER SI LONGUEUR A TRANSFERER > 80 (&50 ***
|
TESTLEN PUSH HL      |PRESERVER POINTEUR ADRESSE
LD HL,(LEN)         |LONGUEUR TOTALE DANS HL
AND A
LD BC,#51           |PLUS 1 POUR PAS AVOIR UN TEST JR Z(Economise 1 octet)
SBC HL,BC
JR C,DERNIER        |CARRY MIS DONC SOUSTRACTION NEGATIVE SUR < 80

INC HL
LD (LEN),HL        |COMPENSER LE 1 DE TROP QUI EVITE UN TEST SUR Z
                    |RANGER LE RESTE DE LA LONGUEUR A TRANSFERER

```

```

POP HL
RET

|LE CARRY N'EST PAS MIS SUITE A LA SOUSTRACTION
|LE TEST SUIVANT CALL DEBUT SERA NEGATIF

|
|*** LONGUEUR < 80 ***
|
DERNIER
ADD HL,BC
LD B,H
LD C,L
POP HL
SCF
RET

|
|*****
|*** PASSAGE DES PARAMETRES POUR SAVED ET LOADD ***
|*****
|
|*** PRENDRE GRAND SOIN DE RESPECTER LES ADRESSAGES ABSOLUS ***
|
DEBUT POP IY          |DECREMENTER LE POINTEUR DE PILE POUR REVENIR AU
CP 4
JP NZ,#9D9D         |BASIC EN CAS DE RETOUR DU A UN MESSAGE D'ERREUR
                    |* BAD PARAMETER * (<> 4)

|
LD L,(IX+0)         |RECUPERER ADRESSE DE LA BANQUE DE 0 A #3FFF
LD H,(IX+1)         |VERIFIER QU'ELLE EST BIEN INFERIEURE A #4000
LD A,D
CP #40
JP NC,#9D89         |EN TESTANT L'OCTET FORT
                    |> #4000 * BAD BANK ADRESS *

|
LD C,(IX+2)         |LONGUEUR A TRANSFERER DANS OU DEPUIS LA BANQUE
LD B,(IX+3)         |VERIFIER QUE LA LONGUEUR EST INFERIEURE A #4000
LD (LEN),BC
LD A,B
CP #40
JP NC,#9D84         |>#4000 * VALUE INVALID *
LD A,C
OR B
JP Z,#9D84         |VERIFIER QUE LONGUEUR <> 0
                    |IDEM SI 0

|
PUSH HL
ADD HL,BC
LD A,H
CP #40
POP HL
JP NC,#9D84         |VOIR SI ADRESSE BANQUE + LONGUEUR > #4000
                    |DEBORDEMENT * VALUE INVALID *

|
LD A,H
ADD A,#40
LD H,A
                    |AJOUTER #4000 A L'ADRESSE DE LA BANQUE POUR TOMBER
                    |DANS LA ZONE MEMOIRE RECONNUE PAR LE CPC
                    |ET RANGER CETTE ADRESSE DANS DE

|
LD E,(IX+4)         |ADPROG A TRANSFERER
LD D,(IX+5)

|
LD A,(IX+6)         |RECUPERER LE NUMERO DE BANQUE ET TESTER VALIDITE

LIST CP 17          |64K - 5 / 256K 17 - 2*256K 33

NOLIST JP NC,#9D93  |NUMERO TROP ELEVE * BANK UNAIAVAILABLE *

|
PUSH IY            |PAS D'ERREUR , RECUPERER ADRESSE RETOUR NORMALE

|
|*** CONVERTIT LE NUMERO DE BANQUE EN VALEUR E/S ***
|
PARABANK DEC A
LD C,A
AND 3
LD B,A
LD A,C
                    |CALCULER NUMERO DE BANQUE
                    |EFFECTUE LE CALCUL DU PARAMETRE A TRANSMETTRE DANS
                    |L'INSTRUCTION OUT COMME EXPLIQUE DANS LE MODE
                    |D'EMPLOI DE LA DK'TRONIC

```

AND 28  
ADD A,A  
OR B  
OR 196  
RET

;  
;\*\*\* TRANSFERT DES OCTETS VERS LE TAMPON OU DEPUIS LE TAMPON \*\*\*  
;

VERSTOCK LD BC,#50 ;VERS LE TAMPON SUR 80 OCTETS  
VSTOCK1 LD DE,#A415 ;OU SUR LE RESTE CALCULE EN DERNIER  
JR TRANS

;  
DESTOCK LD BC,#50 ;DEPUIS LE TAMPON  
DSTOCK1 LD HL,#A415  
TRANS PUSH BC  
LDIR  
POP BC  
RET

;  
LEN DW 0  
DB 6 ;CALER LA SUITE DE LA ROUTINE SUR #9F29  
;

LIST  
;\*\*\* MODIFICATION DE PEEK \*\*\*  
;

NOLIST CP 3 ;PASSAGE ET TEST DES PARAMETRES COMME POUR LOADD-SAVE  
JP NZ,#9D9D  
LD E,(IX+0) ;POINTEUR VARIABLE ENTIERE  
LD D,(IX+1)  
DEC DE  
LD A,(DE) ;TESTER TYPE  
CP 1  
JP NZ,#9D9D  
INC DE ;VARIABLE DANS DE  
LD A,(IX+3)  
AND #C0  
JP NZ,#9D89 ;ADRESSE TROP GRANDE  
PUSH DE  
LD L,(IX+2)  
LD H,(IX+3)  
LD A,H  
ADD A,#40 ;AJOUTER #4000  
LD H,A  
POP DE ;ADRESSE BANQUE DANS HL - RECUPERER POINTEUR VARIABLE  
LD A,(IX+05) ;TESTER NUMERO DE BANQUE  
AND A  
JP NZ,#9D98  
LD A,(IX+04)

LIST

CP 17

NOLIST

JP NC,#9D98

;  
CALL PARABANK ;CALCULER NUMERO BANQUE  
CALL OPENBANK ;OUVRIR BANQUE  
LD A,(HL) ;CONTENU DE L'ADRESSE BANQUE DANS A  
CALL CLOSEBANK ;FERMER BANQUE (REGISTRES RENDUS INTACTS)  
LD (DE),A ;OCTET FAIBLE VARIABLE  
XOR A  
INC DE  
LD (DE),A ;OCTET FORT = 0 PUIS QUE VARIABLE ENTIERE  
RET

LIST

DB 0.