

```

1 REM DISC.BAS
10 REM *** KEY DEF 88,0,0,0,0 ' BREAK interdit.
20 MEMORY &A4FD:POKE &SAFE,1:m=0
25 GOSUB B00
30 INK 0,1:INK 1,24:BORDER 1:MODE 2:GOSUB 400
40 LOCATE 36,2:PRINT "S.A.O":LOCATE 36,4:PRINT CHR$(164);"1987"
50 LOCATE 13,9:PRINT "Si vous n'avez pas fait de RESET, il est encore temps !"
60 LOCATE 13,15:PRINT CHR$(24); " Possédez-vous une disquette formatee en DATA :
  (O/N) "
70 LOCATE 30,18:PRINT CHR$(24);"(17 Ko par schema)"
80 rs="":WHILE rs<>"O" AND rs<>"N":rs=UPPER$(INKEY$):WEND
90 IF rs="N" THEN 340
100 CLS
110 LOCATE 8,4:PRINT "OPTIONS PAR DEFALT":LOCATE 8,5:PRINT STRING$(16,"-")
120 LOCATE 6,7:PRINT "(:CHR$(240);") - MONITEUR MONOCHROME"
130 LOCATE 8,9:PRINT "(:CHR$(241);") - 2 LECTEURS DE DISQUETTE (disc systeme e
  n 'A')"
140 LOCATE 8,14:PRINT "Utilisez les fleches pour changer les options ":LOCATE 35
  ,15:PRINT "puis ":CHR$(24);" COPY ":CHR$(24);" pour lancer le programme "
145 LOCATE 5,24:PRINT CHR$(24);"LE DEUXIEME LECTEUR EST-IL ALLUME ?? Sinon RESE
  T !":SPACE$(11):CHR$(24)
150 GOSUB 260
160 MODE 2:LOAD ecr$,&C000 ' ecran presentation
170 BORDER 2:LOAD "ROUTINES.SCH",&A300:BORDER 14
180 REM *** LOAD "COPY-7B.SCH",&A4F4 ' *** hardcopy 7 bit ***
190 REM *** LOAD hard8$,&A4F4 ' *** hardcopy 8 bits ***
200 BORDER 11
210 REM *** LOAD "BITS-INT.SCH",&A600:CALL &A600 ' *** interface 8 bits interne
  ***
220 IF pr=83 OR pr=73 THEN CALL &A4BE:POKE &BC80,&C3:POKE &BC81,&CB:POKE &BC82,&
  A4
225 IF pr=67 THEN POKE &A395,&D0:POKE &A3AA,&30
230 IF m=0 THEN BORDER 9:INK 0,1:INK 1,24
240 IF m=1 THEN BORDER 1:INK 0,13:INK 1,0
245 IF n=0 THEN POKE &SAFE,9 ELSE POKE &SAFE,1
250 RUN prg$
260 CALL &BB18
270 IF (INKEY(0)<>-1 AND m=0) THEN m=1:LOCATE 21,7:PRINT "COULEUR ":GOTO 320
280 IF (INKEY(0)<>-1 AND m=1) THEN m=0:LOCATE 21,7:PRINT "MONOCHROME":GOTO 320
290 IF (INKEY(2)<>-1 AND PEEK(&SAFE)=1) THEN POKE &SAFE,0:LOCATE 12,9:PRINT "1 L
  ECTEUR DE DISQUETTE":GOTO 320
300 IF (INKEY(2)<>-1 AND PEEK(&SAFE)=0) THEN POKE &SAFE,1:LOCATE 12,9:PRINT "2 L
  ECTEURS DE DISQUETTE":GOTO 320
310 IF INKEY(9)<>-1 THEN 330
320 IF PEEK (&SAFE)=1 THEN LOCATE 5,24:PRINT CHR$(24);"LE DEUXIEME LECTEUR EST-I
  L ALLUME ?? Sinon RESET !":SPACE$(11):CHR$(24):ELSE LOCATE 5,24:PRINT SPACE$(71
  )
325 GOTO 260
330 MODE 2:RETURN
340 MODE 2:LOCATE 19,4:PRINT "POUR PREPARER UNE DISQUETTE AU FORMAT 'DATA'."
350 LOCATE 5,10:PRINT CHR$(24);SPC(26);"INTRODUISEZ LE CP/M":SPC(26);CHR$(24)
360 LOCATE 5,15:PRINT "CPC 464/664 ( CP/M 2.2 ) : Il faudra taper 'FORMAT D'."
370 LOCATE 5,18:PRINT "G128 ( CP/M 3.0 ) : Utiliser 'DISKIT3.COM' et suivre le
  s instructions."
380 LOCATE 5,22:PRINT CHR$(24);" APPUYEZ SUR UNE TOUCHE POUR LANCER LE CP/M ":CH
  R$(24);SPC(11);"A tout de suite !"
390 CALL &BB18:CPM
400 SYMBOL 253,96,48,120,12,124,204,118,0:' @ a accentue
410 SYMBOL 254,12,24,80,102,128,96,60,0:' 5 e accent aigu
420 SYMBOL 255,48,24,60,102,128,96,60,0:' 2 e accent grave
430 KEY 141,CHR$(191):KEY DEF 15,0,128,128,141:' @ OHMS
440 KEY 142,CHR$(190):KEY DEF 11,0,136,136,142:' @ SYGNA
450 KEY 143,CHR$(184):KEY DEF 5,0,131,131,143:' @ PI
460 KEY 144,CHR$(183):KEY DEF 10,0,135,135,144:' @ MICRO
470 KEY 145,CHR$(170):KEY DEF 4,0,134,134,145:' @ 3/4
480 KEY 146,CHR$(189):KEY DEF 13,0,129,129,146:' @ 1 1/2
490 KEY 147,CHR$(169):KEY DEF 20,0,132,132,147:' @ 1/4
500 KEY 148,CHR$(164):KEY DEF 3,0,137,137,148:' @ COPYRIGHT
510 KEY 149,"FIGURE No ":KEY DEF 7,0,138,138,149:' @ POINT message
520 KEY 150,CHR$(253):KEY DEF 26,0,64,124,150:' @ a accentue
530 KEY 151,CHR$(254):KEY DEF 12,0,133,133,151:' @ 5 e accent aigu
540 KEY 152,CHR$(255):KEY DEF 14,0,130,130,152:' @ 2 e accent grave
550 RETURN
600 MODE 0:LOCATE 1,10:PRINT "-S-ao -I-ac -O-lao":LOCATE 8,16:PRINT "S . I . C
  ":LOCATE 10,20:PRINT "?? "
610 as="":WHILE as<>"S" AND as<>"I" AND as<>"O":as=UPPER$(INKEY$):WEND
620 pr=ASC(as)
630 IF pr=67 THEN ecr$="CIAO.SAO":hard8$="IMPCIAO.SCH":prg$="CIAO+E2.SCH"
640 IF pr=73 THEN ecr$="COMPO.SAO":hard8$="COPY-8B.SCH":prg$="COMPO.SCH"
650 IF pr=83 THEN ecr$="VU.SAO":hard8$="COPY-8B.SCH":prg$="SCHEMA.SCH"
660 RETURN

```

```

1 REM BITS-INT.DAT
10 ad=&A600:MEMORY ad-1
20 t=0:FOR i=1 TO 40
30 READ v$:v=VAL("&" + v$)
40 POKE ad,v:ad=ad+1:t=t+v:NEXT
50 READ a:IF a<>t THEN PRINT "ERREUR
  DANS LES DATAS":STOP
60 PRINT "SAVE ";CHR$(34);"BITS-INT.
  SCH";CHR$(34);",B,&A600,&0027"
70 END
100 DATA 2A,2C,BD,22,25,A6,3E,C3
110 DATA 32,2B,BD,21,12,A6,22,2C
120 DATA BD,C9,F5,C5,CB,7F,20,05
130 DATA 01,00,F6,18,03,01,20,F6
140 DATA ED,49,C1,F1,CF,00,00,00
150 DATA 4049

```

Figure 2 : 8 bits UNIVERSEL.

```

1 REM IMPCIAO.DAT
10 MEMORY &A4F3:ad=&A4F4
20 WHILE ad<=&A5ED
30 t=0:FOR i=1 TO 64:READ v$:v=VAL("&"+v$)
40 POKE ad,v:t=t+v:ad=ad+1:NEXT
50 READ b,a:IF a<>t THEN PRINT "ERRE
UR DANS LE BLOCK":b:STOP
60 WEND
70 PRINT "SAVE ":CHR$(34);"IMPCIAO.S
CH":CHR$(34);",B,&A4F4,&00FA"
80 END

100 REM ---BLOCK 1---
110 DATA FE,01,20,07,7B,FE,00,28
120 DATA 02,18,02,3E,01,32,EC,A5
130 DATA 21,00,C0,22,E6,A5,3E,02
140 DATA 32,ED,A5,CD,BD,A5,CD,DC
150 DATA A5,32,ED,A5,CD,D1,A5,3A
160 DATA ED,A5,FE,00,20,04,3E,15
170 DATA 18,02,3E,16,CD,C8,A5,22
180 DATA E6,A5,CD,1B,BB,FE,51,CA
190 DATA 1,7505
200 REM ---BLOCK 2---
210 DATA BD,A5,3E,0A,CD,C8,A5,3A
220 DATA EC,A5,47,C5,2A,E8,A5,3E
230 DATA 0D,CD,C8,A5,3E,1B,CD,C8
240 DATA A5,3E,2A,CD,C8,A5,3E,04
250 DATA CD,C8,A5,3E,80,CD,C8,A5
260 DATA 3E,02,CD,C8,A5,3E,50,32
270 DATA EA,A5,16,80,3E,08,32,EB
280 DATA A5,E5,06,04,0E,00,7E,A2
290 DATA 2,7972
300 REM ---BLOCK 3---
310 DATA 28,08,37,CB,11,37,CB,11
320 DATA 18,06,AF,CB,11,AF,CB,11
330 DATA CD,26,BC,10,E9,79,CD,C8
340 DATA A5,AF,CB,1A,E1,3A,EB,A5
350 DATA 3D,FE,00,20,D1,23,3A,EA
360 DATA A5,3D,FE,00,20,C1,C1,10
370 DATA 9A,2A,E8,A5,06,04,CD,26
380 DATA BC,10,FB,3E,C7,BC,C2,12
390 DATA 3,7635
400 REM ---BLOCK 4---
410 DATA A5,3E,D0,BD,28,03,C3,12
420 DATA A5,3E,1B,CD,C8,A5,3E,40
430 DATA CD,C8,A5,C9,CD,2E,BD,38
440 DATA FB,CD,2B,ED,C9,3E,1B,CD
450 DATA C8,A5,3E,33,CD,C8,A5,C9
460 DATA 3A,ED,A5,FE,00,28,02,3D
470 DATA C9,3E,02,C9,00,00,00,00
480 DATA 00,00,00,00,00,00,00,00
490 DATA 4,6812
500 REM ---FIN DU LISTING---

```

Figure 3 : Hardcopy pour CIAO (correction d'échelle).

```

1 KEM CIAO:Z2
5 cob:=PEEK(&5AFF)
10 nlec:=PEEK(&5AFE):IF nlec=1 THEN ds="B:" ELSE ds="A:"
20 DEFINT a-z:e=3:m=3:n$="":g=0:dc=0:sm=1:bro=48:esps="1":pt=496:als="0":sp=1:pa
st$="3":ecart:=&4:lar=3:lar=0:ecar=&4:qt=3:lop=4:longr=16:psa=8:psr=8:ora=0:orb=
1:dec=1
30 trans:=&A439:comut:=&A447:gpen:=&A440:flag:=&A451:disc:=&A452:numlec:=&A458:
sysdat:=&A46A:errdisc:=&A481:inout:=&A482:hardcopy:=&A4F4
40 MODE 2:POKE comut!,0:CALL trans!
50 GOSUB 230:menu
60 DEFINT a-z
70 CALL &A300,Gv,Gw,Gx,Gy,Gz,Et
75 LOCATE 74,1:PRINT FRE(0):LOCATE 74,1:PRINT " "
80 IF (v<0 OR v>639) OR (w<0 OR w>999) THEN SOUND 1,600,15
90 IF INKEY(62)<>-1 THEN IF dc=0 THEN dc=1 ELSE dc=0
100 GOSUB 1040
120 ON t GOSUB 180,200,230,230
130 GOSUB 150:GOTO 70
140 BRANCHMENTS
150 ORIGIN v,w:ON a GOSUB 3010,2210,3210,3310,3410,3505,3610,3710,3805,2210,2210
,2210,2210,2210,2210,5010,5110,5210,5310,5410,5510,5610,5710,3730,3510,3150
,4005,4105,3910,4405,4205,4310,4504
160 RETURN
170 COPY
180 GOSUB 150:POKE comut!,0:CALL trans!:t=0:RETURN
190 DEL
200 POKE flag!,0:CALL gpen!:GOSUB 180:POKE flag!,1:CALL gpen!:t=0:RETURN
210
220 MENU
230 n$="":v=320:w=200:x=1:y=1:z=0:POKE comut!,0:CALL trans!:IF t=3 THEN 450
240 r$="":n$="":t=0:MODE 2:RESTORE 290
250 LOCATE 1,1:PRINT CHR$(24):" (R)epertoire (C)harger (S)auver (E)ramer (I)
mprimer (N)ouveau (Q)uitter ":CHR$(24)
260 FOR i=7 TO 57 STEP 25:FOR j=4 TO 18 STEP 2:READ comp$
270 LOCATE i,j
280 PRINT comp$:NEXT:NEXT
290 DATA 1 POINTS,2 RECTANGLES,3 DECOCHEMENTS,4 POINTILLES, 5 COINS,6 CONNEXION
5,7 CHOIX DU PAS ,8 ALPHA,9 PASTILLAGE,10 LIAISONS,11 DIP's,12 41612 ac, 13 DECA
LES, 14 RACCORDS,15 SHADOW,16 CONNEXIONS AUTO
300 DATA 17 program.past, 18 program.liaisons,19 program.DIP,20 program. 41612,2
1 program.DECAL,22 program.RACCORDS,23 program.SHADOW,24 program.connex.
310 LOCATE 4,22:PRINT "TAPEZ une LETTRE du MENU ou le NUMERO du COMPOSANT : "
320 LINE INPUT ":",r$:r$=UPPER$(r$):IF r$="" THEN a=0:RETURN
330 ok$="ECSEINO":IF INSTR(ok$,r$)=0 THEN 350
340 ON INSTR(ok$,r$) GOTO 490,490,490,490,680,730,770
350 a=VAL(LEFT$(r$,2)):IF a<1 OR a>24 THEN 310
430 RETURN
440 Fenetre choix composant
450 t=0:WINDOW#1,35,45,12,15:CLS#1:PRINT#1,"COMPOSANT":PRINT#1," No:":
460 LINE INPUT#1,"",r$:IF r$="" THEN 240 ELSE a=VAL(LEFT$(r$,2))
470 IF a<1 OR a>24 THEN 450 ELSE 430
480 repertoire/sauvegarde/chargement/eraser
490 CLS:LOCATE 4,4:PRINT CHR$(7):"INSERER LA DISQUETTE DATA DANS LE LECTEUR "
LEFT$(ds,1):" ET APPUYEZ SUR UNE TOUCHE"
500 get$="":get$=UPPER$(INKEY$):IF get$="" THEN 500 ELSE IF get$="M" THEN 240
510 POKE numlec!,nlec:POKE sysdat!,&C1:CALL tdisc!
520 IF PEEK(errdisc!)=255 THEN GOTO 490
530 POKE &A700,nlec!CAT:POKE &A700,0
540 ON INSTR(ok$,r$) GOTO 640,590,560,620
550 sauvegarde
560 LINE INPUT "NOM DU SCHEMA A SAUVEGARDER (SANS extension) : ",n$:IF n$="" THE
N 640
570 n$=d$+n$+".SAO":PRINT:PRINT "SAUVEGARDE EN COURS...":CALL inout!,1,n$:GOTO
640

```

```

580 chargement
590 LINE INPUT "NOM DU SCHEMA A CHARGER (extension par default = .SAO) : ",n$:IF
n$="" THEN 640
600 n$=d$+n$:IF INSTR(n$,".")=0 THEN n$=n$+".SAO"
605 PRINT:PRINT "CHARGEMENT EN COURS...":CALL inout!,0,n$:GOTO 640
610 eraser
620 LINE INPUT "NOM DU SCHEMA A ERASER (AVEC extension) : ",n$:IF n$="" THEN 640
630 n$=d$+n$:PRINT:PRINT "EFFACEMENT...":ERA,n$:POKE &A700,nlec!CAT:POKE &A700
,0
640 PRINT:PRINT "TAPEZ UNE TOUCHE":CALL &BB18:GOTO 240
650 impression
660 WINDOW#0,33,47,8,15:PAPER 1:PEN 0:CLS
670 PRINT:PRINT "APPUYEZ SUR '0':PRINT "POUR CONFIRMER":PRINT
680 PRINT "AUTRE TOUCHE":PRINT:PRINT "ANNULATION"
690 CALL &BB18:IF INKEY(34)=-1 THEN 710
695 MODE 2: INPUT "nombre de passes : ",paz:paz = VAL(paz)
700 MODE 2:POKE comut!,1:CALL trans!:CALL hardcopy!,paz
710 PAPER 0:PEN 1:GOTO 240
720 effacer
730 mo$=" NOUVEAU":lo$="N":o=46:GOSUB 810
740 IF co<>1 THEN 240
750 MODE 2:POKE comut!,0:CALL trans!:GOTO 240
760 quitter
770 mo$=" QUITTER":lo$="Q":o=67:GOSUB 810
780 IF co<>1 THEN 240
790 CALL 0
800 fenetre impression/nouveau/quitter
810 WINDOW#0,33,47,8,15:PAPER 1:PEN 0:CLS
820 PRINT:PRINT "CTRL/SHIFT ":lo$:PRINT:PRINT mo$:PRINT
830 PRINT "AUTRE TOUCHE":PRINT:PRINT "ANNULATION"
840 co=0:CALL &BB18:IF INKEY(o)=160 THEN co=1
850 PAPER 0:PEN 1:RETURN
1030 coordonnees on/off
1040 IF dc=0 THEN RETURN
1050 LOCATE 1,1:PRINT CHR$(24):v,w:CHR$(24):
1060 FOR i=0 TO 100:NEXT " temporisation
1070 LOCATE 1,1:PRINT SPACES(12):RETURN
1100 S/P Dessins Composants
1110 past$="1"
1120 PLOT -2,4:DRAW 2,4:PLOT -4,2:DRAW 4,2:PLOT -4,0:DRAW -2,0:PLOT 2,0:DRAW 4,0
:PLOT -4,-2:DRAW 4,-2:PLOT -2,-4:DRAW 2,-4:RETURN
1130 past$="2v"
1140 PLOT -2,6:DRAW 2,6:PLOT -4,4:DRAW 4,4:PLOT -4,2:DRAW -2,2:PLOT 2,2:DRAW 4,2
:PLOT -4,0:DRAW -2,0:PLOT 2,0:DRAW 4,0:PLOT -4,-2:DRAW -2,-2:PLOT 2,-2:DRAW 4,-2
:PLOT -4,-4:DRAW 4,-4:PLOT -2,-6:DRAW 2,-6:RETURN
1150 past$="2h"
1160 PLOT -6,4:DRAW 6,4:PLOT -8,2:DRAW -8,2:PLOT 8,2:DRAW 8,2:PLOT -8,0:DRAW -8,0
:PLOT 8,0:DRAW 8,0:PLOT -8,-2:DRAW 8,-2:RETURN
1170 past$="3"
1180 PLOT -4,6:DRAW 4,6:PLOT -5,4:DRAW 5,4:PLOT -6,2:DRAW -2,2:PLOT 2,2:DRAW 6,2
:PLOT -6,0:DRAW -8,0:PLOT 8,0:DRAW 6,0:PLOT -6,-2:DRAW -2,-2:PLOT 2,-2:DRAW 6,-2
:PLOT -5,-4:DRAW 5,-4:PLOT -4,-6:DRAW 4,-6:RETURN
1200 past$="4v"
1210 PLOT -3,8:DRAW 3,8:PLOT -5,6:DRAW 5,6:PLOT -5,4:DRAW 5,4:PLOT -5,2:DRAW -2,2
:PLOT 2,2:DRAW 5,2:PLOT -5,0:DRAW -2,0:PLOT 2,0:DRAW 5,0:PLOT -5,-2:DRAW -2,-2
:PLOT 2,-2:DRAW 5,-2:PLOT -5,-4:DRAW 5,-4:PLOT -5,-6:DRAW 5,-6:PLOT -3,-8:DRAW 3,-8
:RETURN
1220 past$="4h"
1230 PLOT -6,6:DRAW 6,6:PLOT -8,4:DRAW 8,4:PLOT -8,2:DRAW -8,2:PLOT 8,2:DRAW 8,2
:PLOT -8,0:DRAW -8,0:PLOT 8,0:DRAW 8,0:PLOT -8,-2:DRAW 8,-2:PLOT -6,-4:DRAW 6,-4
:RETURN
1250 past$="5"
1260 PLOT -6,10:DRAW 6,10:PLOT -8,8:DRAW 8,8:PLOT -10,6:DRAW 10,6:PLOT -10,4:DRAW
10,4:PLOT -10,2:DRAW -2,2:PLOT 2,2:DRAW 10,2:PLOT -10,0:DRAW -8,0:PLOT 8,0:DRAW
10,0:PLOT -10,-2:DRAW -2,-2:PLOT 2,-2:DRAW 10,-2:PLOT -10,-4:DRAW 10,-4
1270 PLOT -10,-6:DRAW 10,-6:PLOT -8,-8:DRAW 8,-8:PLOT -6,-10:DRAW 6,-10:RETURN

```

```

2000 'SP passages DIN
2010 PLOT 0,0:G*Y:DRAW 0,-8*Y:DRAW -8*X:,-24*Y:DRAW -8*X:,-38*Y:PLOT -1*X,-6
    *Y:DRAW -1*X,-8*Y:DRAW -9*X,-24*Y:DRAW -9*X,-38*Y:PLOT 1*X,-6*Y:DRAW 1*
    X,-8*Y:DRAW -7*X,-24*Y:DRAW -7*X,-38*Y:RETURN
2020 PLOT 0,-6*Y:DRAW 0,-8*Y:DRAW -8*X,-24*Y:DRAW -8*X,-28*Y:PLOT -1*X,-6
    *Y:DRAW -1*X,-8*Y:DRAW -9*X,-24*Y:DRAW -9*X,-28*Y:PLOT 1*X,-6*Y:DRAW 1*
    X,-8*Y:DRAW -7*X,-24*Y:DRAW -7*X,-28*Y:RETURN
2200 ' S/P pointage
2208 ' croix
2210 IF t=2 THEN PLOT 0,0:RETURN
2215 MOVE 0,8:DRAW 0,12:MOVE 6,0:DRAW 12,0:MOVE 0,-6:DRAW 0,-12:MOVE -6,0:DRAW
    -12,0:POKE flag!,0:CALL gpen!:PLOT 0,5:DRAW 0,-5:PLOT 5,0:DRAW -5,0:POKE flag!,1:
    CALL gpen!:PLOT 0,0
2216 IF t=1 THEN POKE comut!,1:CALL trans!:v1=v:w1=w:IF a=2 THEN a=27 ELSE IF a=
    6 THEN a=26 ELSE IF a=11 THEN a=28 ELSE IF a=12 THEN a=29 ELSE IF a=10 THEN a=30
    ELSE IF a=15 THEN a=31 ELSE IF a=13 THEN a=32 ELSE IF a=14 THEN a=33 ELSE IF a=
    16 THEN a=34
2220 RETURN
3000 ' points
3010 IF z=1 THEN 3030
3020 DRAW 4*X,0:PLOT 0,-2*Y:DRAW 4*X,-2*Y:PLOT 0,2*Y:DRAW 4*X,2*Y:RETURN
3030 PLOT -1,0:DRAW 1,0:RETURN
3110 '
3150 ORIGIN 0,0:PLOT v1,w1:IF z=1 THEN BORDER 15 ELSE BORDER cob!
3160 DRAW v1,u:DRAW v,u:DRAW v,w:DRAW v,w:DRAW v1,w1:
3170 IF t<1 AND t<2 THEN POKE comut!,1:CALL trans!:RETURN
3175 IF w1>w THEN ss=2 ELSE ss=-2
3180 IF z=1 THEN FOR i=w TO w1 STEP ss:PLOT v1,i:DRAW w,i:NEXT
3185 a=2:v=v1:w=w1:BORDER cob!:RETURN
3200 ' decrochements
3210 ix=x:iy=y:IF z=1 THEN 3230
3220 GOTO 2020
3230 PLOT -6*X,0:DRAW -8*X,0:DRAW -24*X,-8*Y:DRAW -26*X,-8*Y:PLOT -8*X,-2*Y:DRAW
    -6*X,-2*Y:DRAW -24*X,-10*Y:DRAW -26*X,-10*Y:RETURN
3300 ' pointilles
3310 IF z=1 THEN 3330
3320 DRAW 4,0:PLOT 8,0:DRAW 12,0:PLOT 16,0:DRAW 20,0:PLOT 0,-2:DRAW 4,-2:PLOT 8,
    -2:DRAW 12,-2:PLOT 16,-2:DRAW 20,-2:RETURN
3330 FOR i=0 TO 8:PLOT i,0:DRAW i,-4:PLOT i,-8:DRAW i,-12:PLOT i,-16:DRAW i,-20:
    NEXT i:RETURN
3400 ' croix
3410 IF z=1 THEN 3430
3420 PLOT 4*X,0:DRAW 16*X,0:PLOT 2*X,-2*Y:DRAW 16*X,-2*Y:PLOT 0,-4*Y:DRAW 4*X,-4
    *Y:FOR i=6 TO 14 STEP 2:PLOT 0,-i*Y:DRAW 3*X,-i*Y:NEXT i:RETURN
3430 PLOT 6*X,2*Y:DRAW 16*X,2*Y:PLOT 4*X,0:DRAW 18*X,0:PLOT 2*X,-2*Y:DRAW 16*X,-
    2*Y:PLOT 0,-4*Y:DRAW 7*X,-4*Y:FOR i=6 TO 14 STEP 2:PLOT 0,-i*Y:DRAW 5*X,-i*Y:NEX
    T:RETURN
3500 ' connexions
3505 GOTO 2210
3510 ORIGIN 0,0:PLOT v1,w1:v2=v:w2=w:IF lar=0 THEN lar=4
3515 IF z=1 THEN y=-1 ELSE y=1
3520 IF z=1 THEN x=-1 ELSE x=1
3525 IF v1=v2 AND w1=w2 THEN 3560
3530 IF v1<>v2 AND w1<>w2 THEN 3550
3535 IF v1<>v2 THEN 3540 ELSE 3550
3540 FOR i=1 TO lar-1:PLOT v1,w1:DRAW v2,w2:PLOT v1,w1+((i-1)*y):DRAW v2,w2+((i-
    1)*x):NEXT
3545 IF T=1 OR T=2 THEN a=6:RETURN ELSE POKE comut!,1:CALL trans!:RETURN
3550 FOR i=1 TO lar-1:PLOT v1,w1:DRAW v2,w2:PLOT v1+(i*x),w1:DRAW v2+(i*x),w2:NE
    XT
3555 IF T=1 OR T=2 THEN a=6:RETURN ELSE POKE comut!,1:CALL trans!:RETURN
3580 PLOT 0,0:IF T=1 OR T=2 THEN a=6:RETURN ELSE POKE comut!,1:CALL trans!:RETUR
    N
3580 ' pas variable
3610 MODE 2:t=0:WINDOW#1,30,70,8,9:CLS#1:INPUT #1,"Pas LIBRE(0) - 1.27(1) - 2.54
    (2)":var
3620 IF var=1 THEN hn=8:hs=16:hc=64:vn=8:vs=16:vc=64 ELSE IF var=2 THEN hn=16:hs
    =32:hcb=64:vn=16:vs=32:vc=64 ELSE hn=1:hs=8:hcb=48:vn=1:vs=8:vc=48
3630 POKE &A387,(256-hn):POKE &A38C,hn:POKE &A38E,(256-hs):POKE &A3A3,hs:POKE &A
    385,(256-hc):POKE &A3AA,hc:POKE &A372,vn:POKE &A35D,(256-vn):POKE &A379,vs:POKE
    &A394,(256-vs):POKE &A380,vc:POKE &A36B,(256-vc):GOTO 240
3700 ' alphanumerique
3710 ORIGIN 0,0:TAC:MOVE v,u:PRINT CHR$(143):;IF t=1 THEN POKE comut!,1:CALL tra
    ns!:a=25
3720 TAGOFF:RETURN
3730 ORIGIN 0,0:MOVE v,w
3740 TAG:b$=INKEY$:IF b$="" THEN 3740
3750 IF b$=CHR$(13) THEN 3790
3760 IF b$=CHR$(16) THEN POKE comut!,1:CALL trans!:w=u+16:GOTO 3790
3770 IF b$=CHR$(127) THEN MOVE (XPOS-8),u:PRINT " ":MOVE (XPOS-8),u:GOTO 3740
3780 PRINT b$:GOTO 3740
3790 POKE comut!,0:CALL trans!:w=w-16:a=a-6:TAGOFF:RETURN
3800 ' PASTILLAGE
3805 '
3810 ORIGIN v,w:IF z=1 THEN 3820
3815 GOSUB 3830:ORIGIN v+ecart,w:GOSUB 3830:RETURN
3820 GOSUB 3840:ORIGIN v,w+ecart:GOSUB 3840:RETURN
3830 IF past$="1" THEN GOSUB 1120 ELSE IF past$="2" THEN GOSUB 1130 ELSE IF past
    $="3" THEN GOSUB 1210 ELSE IF past$="4" THEN GOSUB 1180 ELSE IF past$="5" THEN G
    OSUB 1260
3835 RETURN
3840 IF past$="1" THEN GOSUB 1120 ELSE IF past$="2" THEN GOSUB 1130 ELSE IF past
    $="3" THEN GOSUB 1230 ELSE IF past$="4" THEN GOSUB 1180 ELSE IF past$="5" THEN G
    OSUB 1260
3845 RETURN
3850 IF lop=1 THEN GOSUB 1120 ELSE IF lop=2 THEN GOSUB 1130 ELSE IF lop=3 THEN G
    OSUB 1210 ELSE IF lop=4 THEN GOSUB 1180 ELSE IF lop=5 THEN GOSUB 1260
3855 RETURN
3860 IF lop=1 THEN GOSUB 1120 ELSE IF lop=2 THEN GOSUB 1130 ELSE IF lop=3 THEN G
    OSUB 1230 ELSE IF lop=4 THEN GOSUB 1180 ELSE IF lop=5 THEN GOSUB 1260
3865 RETURN
3900 ' Traits programmes
3910 IF t=1 THEN POKE comut!,1:CALL trans!:IF a=30 THEN a=10
3920 IF z=1 THEN 3950
3930 FOR i=0 TO lar STEP psr:ORIGIN v+I,w:GOSUB 3940:NEXT i:RETURN
3940 FOR el=0 TO lar-1 STEP 1:PLOT el,0:DRAW el,-longr:NEXT el:RETURN
3950 FOR i=0 TO lar STEP psr:ORIGIN v,w+I:GOSUB 3960:NEXT i:RETURN
3960 FOR el=0 TO lar-1 STEP 1:PLOT 0,-el:DRAW longr,-el:NEXT el:RETURN
4000 ' .....DIP
4005 IF t=1 THEN POKE comut!,1:CALL trans!:IF a=28 THEN a=11
4010 IF esp$="1" THEN esp=48
4015 IF esp$="2" THEN esp=96
4020 IF esp$="0" THEN esp=0
4025 IF esp$="0" THEN bro=((VAL(bro$))-1)*16
4030 IF z=1 THEN 4050
4035 FOR i=0 TO bro STEP 16:ORIGIN v+I,w:IF sm=1 THEN GOSUB 1200 ELSE GOSUB 1130
4040 NEXT:FOR i=0 TO bro STEP 16:ORIGIN v+I,w-esp:IF sm=1 THEN GOSUB 1200 ELSE G
    OSUB 1130
4045 NEXT:RETURN
4050 FOR i=0 TO bro STEP 16:ORIGIN v,w-I:IF sm=1 THEN GOSUB 1230 ELSE GOSUB 1150
4055 NEXT:FOR i=0 TO bro STEP 16:ORIGIN v+esp,w-I:IF sm=1 THEN GOSUB 1230 ELSE
    GOSUB 1150
4060 NEXT:RETURN

```

Figure 4b

```

4100 *****DIN 4182 32*2
4105 IF t=1 THEN POKE comut!,1:CALL trans!:IF a=29 THEN a=12
4110 IF x=-1 THEN cal=8 ELSE cal=0
4115 GOSUB 4145:IF al$="0" OR al$="" THEN RETURN
4118 IF z=1 THEN RETURN
4120 IF al$="1" THEN 4150
4125 IF al$="2" THEN 4155
4130 IF al$="3" THEN GOSUB 4150:GOTO 4155
4135 IF al$="4" THEN GOSUB 4150:GOSUB 4155:GOTO 4160
4140 IF al$="5" THEN GOSUB 4150:GOSUB 4155:GOTO 4165
4145 IF z=1 THEN 4147 ELSE IF al$="0" AND sp=2 THEN 4146 ELSE FOR I=0 TO pt STEP
(16*sp):ORIGIN v+1,w:GOSUB 1140:NEXT:FOR I=0 TO pt STEP (16*sp):ORIGIN v+1,w-32
:GOSUB 1140:NEXT:RETURN
4146 FOR I=0 TO pt STEP (16*sp):ORIGIN v+1,w:GOSUB 1210:NEXT:FOR I=0 TO pt STEP
(16*sp):ORIGIN v+1,w-32:GOSUB 1210:NEXT:RETURN
4147 IF al$="0" AND sp=2 THEN 4148 ELSE FOR I=0 TO pt STEP (16*sp):ORIGIN v,w+1:
GOSUB 1180:NEXT:FOR I=0 TO pt STEP (16*sp):ORIGIN v-32,w+1:GOSUB 1180:NEXT:RETUR
N
4148 FOR I=0 TO pt STEP (16*sp):ORIGIN v,w+1:GOSUB 1230:NEXT:FOR I=0 TO pt STEP
(16*sp):ORIGIN v-32,w+1:GOSUB 1230:NEXT:RETURN
4150 ix=(-1*x):iy=-1:FOR I=0 TO pt STEP (16*sp):ORIGIN v+1,w-32:GOSUB 2010: NEXT
:RETURN
4155 ix=(1*x):iy=1:FOR I=0 TO pt STEP (16*sp):ORIGIN v+1,w:GOSUB 2010:NEXT:RETUR
N
4160 FOR I=0 TO pt+(8*sp) STEP (8*sp):ORIGIN ((v-(8*sp))+cal)+I,w-32:ix=(-1*x):i
y=-1:GOSUB 2020:NEXT:RETURN
4165 FOR I=0 TO pt+(8*sp) STEP (8*sp):ORIGIN (v-cal)+I,w:ix=(1*x):iy=-1:GOSUB 20
20:NEXT:RETURN
4200 *****DECALES
4205 IF z=1 THEN 4245
4210 v4=v
4215 FOR i=1 TO ((q1-1)/2):ORIGIN v4,w:v4=v4+pas:GOSUB 3850:NEXT i:
4220 v4=v-(pas/2):w4=w-(ecar*y):
4225 FOR i=1 TO ((q1+1)/2):ORIGIN v4,w4:v4=v4+pas:GOSUB 3850:NEXT i:
4230 ORIGIN v,w:RETURN
4245 w4=w
4250 FOR i=1 TO ((q1-1)/2):ORIGIN v,w4:w4=w4-pas:GOSUB 3860:NEXT i:
4255 v4=v+(ecar*x):w4=w+(pas/2)
4260 FOR i=1 TO ((q1+1)/2):ORIGIN v4,w4:w4=w4-pas:GOSUB 3860:NEXT i:
4265 ORIGIN v,w:RETURN
4300 *****RACCORDES programmes
4310 IF t=1 THEN POKE comut!,1:CALL trans!:IF a=33 THEN a=14
4320 IF z=1 THEN 4350
4330 FOR i=0 TO lir STEP psr:ORIGIN v+(i*x),w-((i*ora)*(y*dec)):GOSUB 4340:NEXT
i:RETURN
4340 FOR el=0 TO lar-1:PLOT (el*x),0:DRAW (el*x),-(y*dec)*((dec*longr)+ex*(i/ps
r))+((i*orb)):NEXT el:RETURN
4350 FOR i=0 TO lir STEP psr:ORIGIN v-((i*ora)*(x*dec)),w+(i*y):GOSUB 4360:NEXT
i:RETURN
4360 FOR el=0 TO lar-1:PLOT 0,(-el*y):DRAW ((dec*longr)+ex*(i/psr)+(i*orb))*(i-x
dec),(-el*y):NEXT el:RETURN
4400 *****SHADOW
4405 IF t=1 THEN POKE comut!,1:CALL trans!:IF a=31 THEN a=15
4410 PLOT 0,0
4415 IF z=1 THEN 4445
4420 IF bout=1 THEN v3=v+(88*x):w3=w+(18*y) ELSE v3=v+(138*x):w3=w+(18*y)
4425 IF typ=1 THEN GOTO 4440
4430 IF typ=2 THEN GOSUB 4440 :v3=v3+(76*x):GOTO 4440
4435 IF typ=3 THEN GOSUB 4440 :v3=v3+(76*x):GOSUB 4440 :v3=v3+(76*x):GOTO 4440
4440 ORIGIN v3,w3:GOSUB 1180:ORIGIN v3+(25*x),w3:GOSUB 1180:ORIGIN v3+(50*x),w3:
GOSUB 1180:ORIGIN v3,w3-(36*y):GOSUB 1180:ORIGIN v3+(25*x),w3-(36*y):GOSUB 1180:
ORIGIN v3-(50*x),w3-(36*y):GOTO 1180
4445 IF bout=1 THEN v3=v+(18*x):w3=w-(88*y) ELSE v3=v+(18*x):w3=w-(138*y)
4450 IF typ=1 THEN GOTO 4465
4455 IF typ=2 THEN GOSUB 4465:w3=w3-(74*y):GOTO 4465
4460 IF typ=3 THEN GOSUB 4465:w3=w3-(74*y):GOSUB 4465:w3=w3-(74*y):GOTO 4465

```

```

4455 ORIGIN v3,w3:GOSUB 1180:ORIGIN v3,w3-(26*y):GOSUB 1180:ORIGIN v3,w3-(50*y):
GOSUB 1180:ORIGIN v3-(36*x),w3:GOSUB 1180:ORIGIN v3-(36*x),w3-(26*y):GOSUB 1180:
ORIGIN v3-(36*x),w3-(50*y):GOTO 1180
4500 ' connexions
4502 GOTO 2210
4504 ORIGIN 0,0:PLOT v1,w1:v2=v:w2=w:IF lar=0 THEN lar=4
4506 IF v1>v2 THEN mn=0:se=0
4508 IF v1<v2 THEN mn=0:se=1
4510 IF w1>w2 THEN mn=1:se=0
4512 IF w1<w2 THEN mn=1:se=1
4514 'IF Z=1 THEN x=-1 ELSE x=1
4516 'IF Z=1 THEN y=-1 ELSE y=1
4518 IF v1=v2 AND w1=w2 THEN 4530
4520 IF v1<v2 AND w1<w2 THEN GOTO 4528
4522 IF v1<v2 THEN 4524 ELSE 4528
4524 FOR i=1 TO lar-1:PLOT v1,w1:DRAW v2,w2:PLOT v1,w1+((i-1)*y):DRAW v2,w2+((i-
1)*y):NEXT
4526 IF T=1 OR T=2 THEN a=16:mo=mn:sa=se:RETURN ELSE POKE comut!,1:CALL trans!:G
OTO 4532
4528 FOR i=1 TO lar-1:PLOT v1,w1:DRAW v2,w2:PLOT v1+(i*x),w1:DRAW v2+(i*x),w2:NE
XT
4530 IF T=1 OR T=2 THEN a=16:mo=mn:sa=se:RETURN ELSE POKE comut!,1:CALL trans!:G
OTO 4532
4532 IF sa=1 AND mn=1 AND (mn<mo) THEN y=-1:x=-1:
4534 IF sa=0 AND mn=1 AND (mn<mo) THEN y=-1:x=1:
4536 IF sa=1 AND mn=0 AND (mn<mo) THEN y=-1:x=-1
4538 IF sa=0 AND mn=0 AND (mn<mo) THEN y=-1:x=1
4540 IF mn=1 AND mn=mo THEN se=sa:
4542 IF sa=0 AND (mn=mo) THEN se=sa:
4544 IF v1=v2 AND w1=w2 THEN se=sa:mn=mo
4546 IF se=0 AND (mn=mo) THEN se=sa:
4548 RETURN
5000 *****programmation pastillage
5010 MODE 2:t=0:WINDOW#1,30,80,8,9:CLS#1:INPUT #1,"TYPE DE PASTILLE (1-5):":past
$:
5020 INPUT #1,"ECART (fois 1.27):":ecart$
5030 ecart=VAL(ecart$)*8:CLS#1:a=9:RETURN
5100 *****Programmation traits
5110 MODE 2:t=0:WINDOW#1,30,80,8,9:CLS#1:INPUT #1,"Nbre de liaisons:":lir$
5120 INPUT #1,"longueur (fois 1.27):":longr$:longr=(VAL(longr$))*8
5130 INPUT #1,"PAS:1.27 =0 /2.54 =2 /5.08 =5 >":PASR$:IF PASR$="2" THEN psr=16 EL
SE IF PASR$="5" THEN psr=32 ELSE psr=8
5135 INPUT #1,"largeur des traits:":lar
5140 lir=((VAL(lir$)-1)*psr
5150 a=10:RETURN
5200 *****programmation dip
5210 MODE 2:t=0:WINDOW#1,30,80,8,9:CLS#1:INPUT #1,"Nbre de broches:":bro$
5220 INPUT #1,"ESPACE=":esp$
5230 INPUT #1,"SMALL=0 LARGE=1:":sm
5240 bro=((VAL(bro$)/2)-1)*16
5250 a=11:RETURN
5300 *****Programmation 41812
5310 MODE 2:t=0:WINDOW#1,30,80,8,9:CLS#1:INPUT #1,"Nbre de points:":pt$
5320 INPUT #1,"espacement 2.54=2 / 5.08=5:":var
5330 IF var=0 THEN sp=0 ELSE IF var=2 THEN sp=1 ELSE IF var=5 THEN sp=2:
5335 IF sp=2 THEN al$="0":GOTO 5350
5340 INPUT #1,"code sorties 0-5=":al$
5350 pt=((VAL(pt$)/2)-1)*16:IF sp=2 THEN pt=pt*2
5360 a=12:RETURN
5400 *****programmation DECALES
5410 MODE 2:t=0:WINDOW#1,30,80,8,9:CLS#1:INPUT #1,"PAS (fois 1.27):":pas:pas=pas
*8
5420 INPUT #1,"ECART (fois 1.27):":ecar:ecar=ecar*8
5430 INPUT #1,"Nombre de points:":iq
5440 INPUT #1,"PASTILLES 1-5:":lop
5450 a=13:RETURN

```

```

5500 ' ' ' ' ' ' Programmation RACCORDES
5510 MODE 2:t=0:WINDOW#1,25,80,6,15:CLS#1:INPUT #1,"Nbre de raccords: ";lir$
5520 INPUT #1,"base du decalage (fois 1.27)=";longr$:longr=(VAL(longr$))*8
5530 INPUT #1,"PAS de depart:1.27=0/2.54=2 /5.08=5>";PASR$:IF PASR$="2"THEN psr=
16 ELSE IF PASR$="5"THEN psr=32 ELSE psr=8
5540 INPUT #1,"PAS d'arrivee:1.27=0 /2.54=2 /5.08=5>";PASA$:IF PASA$="2"THEN psa
=16 ELSE IF PASA$="5"THEN psa=32 ELSE psa=8
5550 INPUT #1,"ORIGINE FLAT=0 / decallee =1";ora:IF ora=0 THEN orb=1:dec=1 ELSE o
rb=0:IF psr>psa THEN dec=-1 ELSE dec=1
5560 INPUT #1,"largeur des traits :";lar
5570 lir=((VAL(lir$))-1)*psr
5580 ex=(psa-psr)
5590 a=14:RETURN
5600 ' ' ' ' ' ' programmation SHADOW
5610 MODE 2:t=0:WINDOW#1,30,70,6,9:CLS#1:INPUT #1,"bouton FG=1 / FA=2 ";bout:
5620 INPUT #1,"cellule F2=1 / F4=2 / F6=3 :";typ
5630 a=15:RETURN
5700 ' ' ' ' ' ' program.connexions
5710 MODE 2:t=0:WINDOW#1,30,70,6,9:CLS#1:INPUT #1,"LARGEUR des traits: ";lar$
5720 lar=(VAL(lar$))
5730 a=8:RETURN

```

FIN FIGURE No: 4

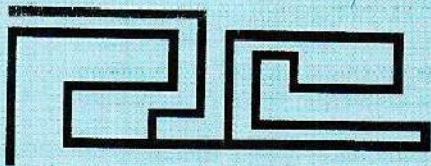
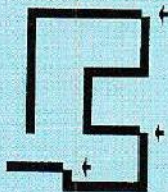
(R)epertoire (C)harger (S)auver (E)raser (I)mprimer (N)ouveau (Q)uitter

1 POINTS	9 PASTILLAGE	17 program.past.
2 RECTANGLES	10 LIAISONS	18 program.liaisons
3 DECROCHEMENTS	11 DIP's	19 program.DIP
4 POINTILLES	12 41612 ac	20 program. 41612
5 COINS	13 DECALES	21 program.DECAL.
6 CONNEXIONS	14 RACCORDS	22 program.RACCORDS
7 CHOIX DU PAS	15 SHADOW	23 program.SHADOW
8 ALPHA	16 CONNEXIONS AUTO	24 program.connex.

TAPEZ une LETTRE du MENU ou le NUMERO du COMPOSANT :

Figure 5 : Menu de CIAO.

Défauts des raccords
de segments (sans "R")



CONNEXIONS AUTO : le défaut disparaît
Mais le procédé comporte encore quelques erreurs
(avec les retours en arrière) Correction délicate

Figure 6 : Problèmes de traits (6 et 16).

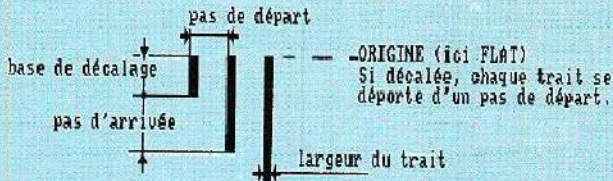
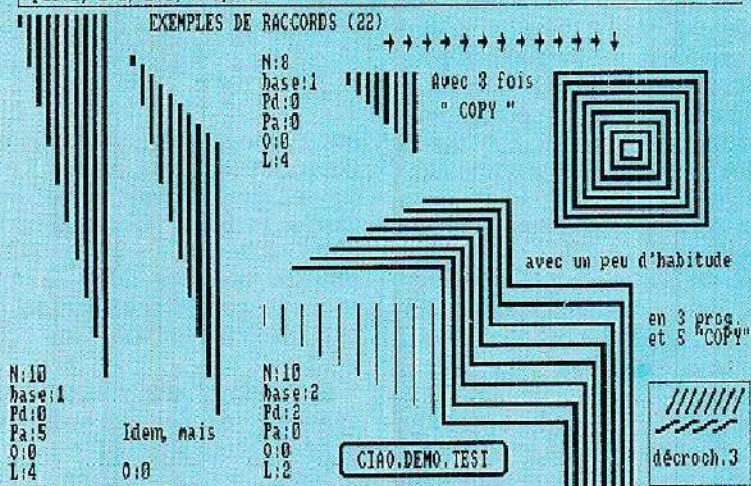
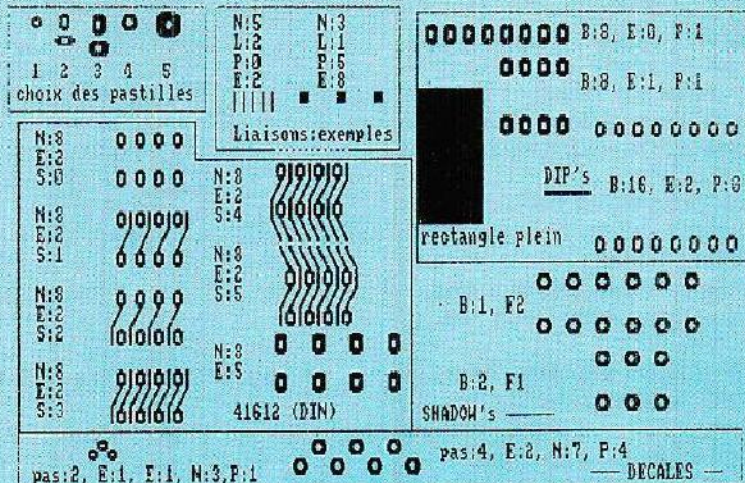


Figure 7 : codes des raccords (22).

Figure 8 : DEMO (à consulter !).



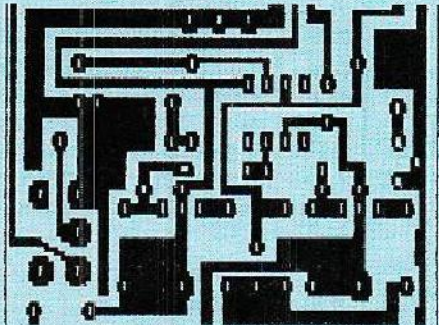
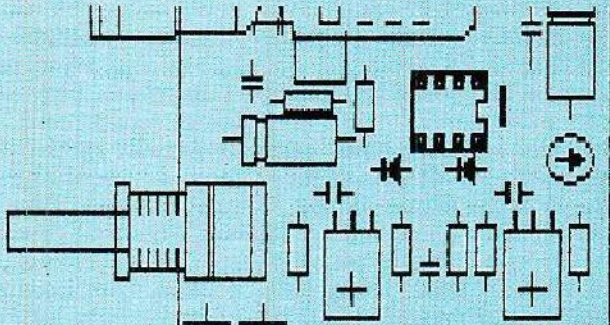


Figure 9 : Après réduction, IAO rentre en action (éch. 1) Le travail est plus facile en modifiant IAO à l'aide de : POKE &A395, (96) : POKE &A3AA, (160), (cf.texte)



```

1 REM TTO.BAS
10 MEMORY &44BF
20 LOAD "BITS-INT.SCH",&A600:CALL &A600
30 LOAD "TTO.BIN",&A400
40 POKE &A558,21:POKE &A588,1
50 ' menu
60 ON ERREUR GOTO 60:MODE 2
70 LOCATE 2,5:PRINT "1 CATALOGUE"
80 LOCATE 2,7:PRINT "2 CHARGEMENT"
90 LOCATE 2,9:PRINT "3 IMPRESSION DOUBLE DENSITE (7 ou 8 bits)"
100 LOCATE 2,11:PRINT "4 IMPRESSION ech:2 -> ech:1 (8 bits)"
110 LOCATE 2,13:PRINT "5 REDUCTION ET SAUVEGARDE (ech:2 -> ech:1)"
120 LOCATE 2,15:PRINT "6 QUITTER"
130 LOCATE 10,20:PRINT CHR$(24);" ENTREZ VOTRE CHOIX : ";CHR$(24)
140 r$="":WHILE r$<"1" OR r$>"6":r$=INKEY$:WEND
150 MODE 2:ON VAL(r$) GOSUB 180,210,250,290,330,370
160 GOTO 60
170 ' catalogue
180 INPUT "CATALOGUE: A ou B ";d$:d$=d$+": "
190 DIR,@d$:CALL &BB18:RETURN
200 ' chargement
210 nc$="":INPUT "CHARGEMENT de ";nc$:IF nc$="" THEN RETURN
220 MODE 2:LOAD nc$,&C000:CALL &A5EC,0
230 CALL &BB18:RETURN
240 ' impression dd
250 id$="":INPUT "IMPRESSION DOUBLE DENSITE : Appuyez sur 1 ",id$
260 IF UPPER$(id$)<>"1" THEN RETURN
270 MODE 2:CALL &A5EC,1:CALL &A54A:RETURN
280 ' impression ech:1
290 ic$="":INPUT "IMPRESSION ech:2 -> ech:1 : Appuyez sur 1 ",ic$
300 IF UPPER$(ic$)<>"1" THEN RETURN
310 MODE 2:CALL &A5EC,1:CALL &A499:RETURN
320 ' sauvegarde ech:1
330 ns$="":INPUT "NOM DU DESSIN REDUIT ";ns$:IF ns$="" THEN RETURN
340 MODE 2:CALL &A5EC,1:CALL &A400
350 SAVE ns$.B.&C000,&4000:RETURN
360 ' quitter
370 PRINT "GOTO 60 POUR RELANCER":END

```

Figure 10: Le cadeau du mois : TTO
Lanceur basic.

```

1 REM TTO.DAT
10 MEMORY &A3FF:ad=&A400
20 WHILE ad<=&A5FE
30 t=0:FOR i=1 TO 64:READ v$:v=VAL("
  &t+v$)
40 POKE ad,v:t=t+v:ad=ad+1:NEXT
50 READ b,a:IF a<>t THEN PRINT "ERRE
UR DANS LE BLOCK";b:STOP
60 WEND
70 PRINT "SAVE ";CHR$(34);"TTO.DAT":
  CHR$(34);",B,&A400,&01FE
80 END

100 REM ---BLOCK 1---
110 DATA 21,00,C0,11,C0,84,08,84
120 DATA C5,E5,06,50,7E,12,23,13
130 DATA 10,FA,E1,CD,26,BC,CD,26
140 DATA BC,C1,10,EC,3E,50,32,98
150 DATA A4,CD,77,A4,21,C0,84,11
160 DATA 00,C0,06,64,C5,D5,06,28
170 DATA AF,77,1A,CB,7F,28,02,CB
180 DATA FE,CB,6F,28,02,CB,F8,CB
190 DATA 1,7461
200 REM ---BLOCK 2---
210 DATA 5F,28,02,CB,EE,CB,4F,28
220 DATA 02,CB,EB,13,1A,CB,7F,28
230 DATA 02,CB,DE,CB,6F,28,02,CB
240 DATA D6,CB,5F,28,02,CB,CE,CB
250 DATA 4F,28,02,CB,C6,23,13,10
260 DATA C7,D1,EB,CD,26,CB,EB,C1
270 DATA 10,BA,3E,28,32,98,A4,3E
280 DATA 02,CD,0E,BC,11,C0,84,21
290 DATA 2,7511
300 REM ---BLOCK 3---
310 DATA 00,C0,06,64,C5,E5,3A,98
320 DATA A4,47,1A,77,23,13,10,FA
330 DATA E1,CD,26,BC,C1,10,ED,C9
340 DATA 00,CD,32,A5,3E,1B,CD,3D
350 DATA A5,3E,38,CD,3D,A5,3E,16
360 DATA CD,3D,A5,21,00,C0,22,46
370 DATA A5,CD,1B,BB,FE,51,CA,32
380 DATA A5,3E,0A,CD,3D,A5,3E,0D
390 DATA 3,7179
400 REM ---BLOCK 4---
410 DATA CD,3D,A5,3E,1B,CD,3D,A5
420 DATA 3E,2A,CD,3D,A5,3E,04,CD
430 DATA 3D,A5,3E,40,CD,3D,A5,3E
440 DATA 01,CD,3D,A5,3E,50,32,48
450 DATA A5,16,80,3E,08,32,49,A5
460 DATA E5,08,08,0E,00,7E,A2,28
470 DATA 05,37,CB,11,18,03,AF,CB
480 DATA 11,CD,26,CB,10,EF,79,CD
490 DATA 4,6270

500 REM ---BLOCK 5---
510 DATA 3D,A5,AF,CB,1A,CB,1A,E1
520 DATA 3A,49,A5,3D,3D,FE,00,20
530 DATA D4,23,3A,48,A5,3D,FE,00
540 DATA 20,C4,2A,46,A5,06,08,CD
550 DATA 26,BC,10,FB,3E,C7,BC,C2
560 DATA AE,A4,3E,D0,BD,28,03,C3
570 DATA AE,A4,3E,1B,CD,3D,A5,3E
580 DATA 40,CD,3D,A5,C9,CD,2E,BD
590 DATA 5,7596
600 REM ---BLOCK 6---
610 DATA 38,FB,CD,2B,BD,C9,00,00
620 DATA 00,00,CD,32,A5,3E,1B,CD
630 DATA 3D,A5,3E,33,CD,3D,A5,3E
640 DATA 13,CD,3D,A5,CD,BA,BB,CD
650 DATA E7,BB,32,EB,A5,11,00,00
660 DATA 21,8F,01,22,E9,A5,3E,07
670 DATA 32,E8,A5,3E,0A,CD,3D,A5
680 DATA 3E,0D,CD,3D,A5,3E,1B,CD
690 DATA 6,7037
700 REM ---BLOCK 7---
710 DATA 3D,A5,3E,2A,CD,3D,A5,3E
720 DATA 02,CD,3D,A5,3E,7F,CD,3D
730 DATA A5,3E,02,CD,3D,A5,0E,00
740 DATA 3A,EB,A5,47,E5,C5,D5,CD
750 DATA F0,BB,D1,C1,21,EB,A5,BE
760 DATA E1,37,20,01,A7,CB,11,2B
770 DATA 2B,10,E9,79,CD,3D,A5,13
780 DATA E5,21,7F,02,37,ED,52,E1
790 DATA 7,7803
800 REM ---BLOCK 8---
810 DATA 38,05,2A,E9,A5,18,CF,CD
820 DATA 1B,BE,FE,51,20,03,C3,32
830 DATA A5,7C,B4,20,05,7D,B5,CA
840 DATA 32,A5,7C,FE,FF,CA,32,A5
850 DATA 11,00,00,22,E9,A5,18,8B
860 DATA 00,00,00,00,7B,21,C0,44
870 DATA 11,00,C0,01,00,40,FE,01
880 DATA 28,01,EB,ED,E0,C9,00,00
890 DATA 8,6492
900 REM ---FIN---

```

Figure 11 : TTO.DAT.

```

1 REM MOUSE.DAT
10 MEMORY &A5B6:ad=&A5B7
20 WHILE ad<=&A626
30 t=0:FOR i=1 TO 56:READ v$:v=VAL("&" + v$)
40 POKE ad,v:t=t+v:ad=ad+1:NEXT
50 READ b,a:IF a<>t THEN PRINT "Erreur dans le bloc";b:STOP
60 WEND
70 POKE ad,0
80 PRINT "SAVE ";CHR$(34);"MOUSE.SCH";CHR$(34);",B,&A5B7,113
90 END
100 REM -----BLOCK1-----
110 DATA 3E,4E,CD,1E,BB,20,04,3E
120 DATA 00,18,02,3E,01,32,27,A6
130 DATA CD,24,BB,1F,DC,E3,A5,1F
140 DATA DC,F5,A5,1F,DC,17,A6,1F
150 DATA DC,05,A6,1F,DA,12,A4,1F
160 DATA DA,0B,A4,C9,F5,11,04,00
170 DATA 3A,27,A6,FE,00,28,03,11
180 DATA 1,5497
190 REM -----BLOCK 2-----
200 DATA 30,00,F1,C3,CD,A3,F5,11
210 DATA FC,FF,3A,27,A6,FE,00,28
220 DATA F1,11,D0,FF,18,EC,F5,11
230 DATA 02,00,3A,27,A6,FE,00,28
240 DATA 03,11,18,00,F1,C3,C7,A3
250 DATA F5,11,FE,FF,3A,27,A6,FE
260 DATA 00,28,F1,11,B8,FF,18,EC
270 DATA 2,7155
280 REM -----TERMINE-----

```

Figure 12

