

Maxifont.bas

[fonts.zip](#)

This module is part of the original MMBasic library. It is reproduced here with kind permission of Hugh Buckle and Geoff Graham. Be aware it may reference functionality which has changed or is deprecated in the latest versions of MMBasic.

Maxifont is a small program written by Dennis Wyatt for the Maximite Computer by Geoff Graham 2011/2012.

It will create a font, based on your input and save the font to a filename of your choice, appended with the .fnt descriptor.

It can add, flip horizontally or vertically, any of the characters it creates.

You only have to run the Maxifont.bas program and it will check the current Version of MMBasic and run the appropriate program for the version Installed - Currently only versions 2.7B, 3, 3.0A and 3.1 are supported.

The limitations of the software is the limited memory size for variables and arrays and therefore, only small font sizes can have lots of characters. The larger the font size, the less characters can be edited. This lies in the way Arrays are stored and in part, the way I have written Maxifont.

I will endeavour to Update the program so that it could enable larger fonts, with support for more characters

Regards

Dennis Wyatt dpwyatt(at)iinet.net.au

Maxifont.BAS:

```
10 '*****
20 '*      Maxifont startup program      *
30 '*      Detects which version of firmware  *
40 '*      V2.7B , V3.0 , V3.0A or V3.1      *
50 '*****
60 version = MM.VER
70 IF version = 3.01 THEN RUN "Maxf3_1.bas"
80 IF version = 3 or version = 3.0001 THEN RUN "Maxf3_0.bas"
90 IF version = 2.0702 THEN RUN "MaxF2_7.bas"
100 CLS
120 PRINT " Sorry your firmware version is not supported yet"
130 END
```

MaxF2_7.BAS:

```
10 '*****
15 '*           Maxi-Font           *
20 '*   Font Editor by Dennis Wyatt *
30 '*           Using mmBasic for the *
40 '*           Maximite computer    *
50 '*****
55 ' get maximum screen dimension-split into 20 parts
60 Max_x=MM.HRES : Min_x=CINT(Max_x/20)
70 Max_y=MM.VRES : Min_y=CINT(max_y/20)
80 CLS
84 addchar=1
85 '*****
86 ' Set to Capture SD card errors
87 '*****
90 OPTION Error Continue
95 '*****
96 '*           Show front page
97 '*****
100 FONT LOAD "gothic.fnt" AS #4 : FONT #4
110 LOCATE 0,1 : ?"2000000000000004"
115 LOCATE 0,33 : ? "1" : LOCATE max_x-32,33 : ? "1"
116 LOCATE 0,65 : ? "1" : LOCATE max_x-32,65 : ? "1"
120 LOCATE 0,92
130 ?"3000000000000005"
135 '*****
136 '*           return to font 1 and unload gothic font
137 '*           need the memory
138 '*****
140 FONT #1 : FONT unload #4
150 FONT LOAD "invade.fnt" AS #5 :FONT #5
160 LOCATE 40,57 : ? "0  2  4": LOCATE 15*min_x,57: ? "1  3  5"
170 FONT #1
180 LOCATE 9*min_x,37: ? "Welcome to"
190 FONT #1,3 : LOCATE 7.3*min_x,52 : ? "MaxiFont"
200 LINE (0,124)-(max_x,124),1
210 FONT #1,1
220 LOCATE 2*min_x,130
230 ? "This program will help you to design some interesting fonts, with"
240 LOCATE 2*min_x,145
250 ? "the added scope of allowing the design of program sprites, for the"
260 LOCATE 2*min_x,160
270 ? "inclusion into some Maximite games, or just make some gothic
borders."
280 LOCATE 2*min_x,190
290 ? "The design of the font should take into account the available"
300 LOCATE 2*min_x,205
310 ? "memory in the system. This is about 37 Kb, for arrays, used in the"
320 LOCATE 2*min_x,220
330 ?"design process. This limits the number of characters, in the editor"
340 LOCATE 2*min_x,235
350 ?"at one time. A 255 pixel wide font by 64 pixels high can only have"
```

```
360 LOCATE 2*min_x,250
370 ?"2 Characters, if I rewrite the code. A 10 pixel by 10 pixel font can"
380 LOCATE 2*min_x,265
390 ?"have only 70 characters in the editor at one time."
400 LOCATE 2*min_x,280
410 ?"You can always append the files together later. Once saved the font"
420 LOCATE 2*min_x,295
430 ?"takes up one quarter the size."
440 LOCATE 2*min_x,310
450 ?"It would be easier to design a smaller font and just piece them"
460 LOCATE 2*min_x,325
470 ?"together to form one large picture."
480 LOCATE 2*min_x,340
490 ?"If you encounter an out of memory error, then reduce the number of "
500 LOCATE 2*min_x,355
510 ?"Characters"
520 LOCATE 9*min_x,385 : ?"press any key"
530 DO WHILE (INKEY$="")
540 FONT #5
550 PAUSE 100
560 frontpage=NOT frontpage
580 IF (frontpage=1) THEN
590 LOCATE 40,57 : ? "1 3 5": LOCATE 15*min_x,57: ? "0 2 4"
600 ELSE
610 LOCATE 40,57 : ? "0 2 4": LOCATE 15*min_x,57: ? "1 3 5"
620 ENDIF
630 LOOP
640 FOR a=1 TO 5
650 PAUSE 150
660 ON a GOTO 670,680,690,700,710
670 LOCATE 40,57 : ? "6 6 6": LOCATE 15*min_x,57: ? "6 6 6":NEXT a
680 LOCATE 40,57 : ? "7 7 7": LOCATE 15*min_x,57: ? "7 7 7":NEXT a
690 LOCATE 40,57 : ? "8 8 8": LOCATE 15*min_x,57: ? "8 8 8":NEXT a
700 LOCATE 40,57 : ? "9 9 9": LOCATE 15*min_x,57: ? "9 9 9":NEXT a
710 t$=CHR$(59)+" "+CHR$(59)+" "+CHR$(59)
720 LOCATE 40,57 : ? t$: LOCATE 15*min_x,57: ? t$ :NEXT a
730 FONT #1 : FONT unload #5
735 t$=" "
740 LINE(2*min_x,130)-(max_x,max_y),0,BF
745 LOCATE 0,130
750 ? t$; : INPUT "Width of Font? ",fwidth
760 IF (fwidth>255) THEN
761 ? t$+"Cannot be bigger than 255 pixels": GOTO 750
762 ENDIF
770 ? t$; : INPUT "Height of Font ? ",fheight
780 IF (fheight > 64) THEN
781 ? t$+"Cannot be higher than 64 pixels":GOTO 770
782 ENDIF
790 IF (fheight =0) THEN
791 ? t$+"Cannot be lower than 1 pixels":GOTO 770
792 ENDIF
```

```
800 ? t$; : INPUT "Number of Characters ? ",fnumber
810 IF (fnumber =0) THEN
811 ? t$+"Cannot be less than 1 Character":GOTO 800
812 ENDIF
820 IF (fnumber >128) THEN
821 ? t$+"Cannot be more than 128 Characters":GOTO 800
822 ENDIF
830 ? t$; : INPUT "Start Character set at ?";fstart
840 IF (fstart<32) THEN
841 ? t$+"Cannot be less than 32 ": GOTO 830
842 ENDIF
850 CLS
860 OPTION base 0
870 DIM values(4)
880 DIM plot(fnumber,fwidth,fheight)
890 LINE (0,0)-(max_x-1,100),1,BF
900 LINE (10,10)-(max_x-11,90),0,BF
910 GOSUB 3050
920 LINE (0,101)-(max_x-1,max_y-1),1,B
930 LINE (2,103)-(6*min_x,max_y-3),1,B
940 screenx=Max_x-1-(7*min_x)
950 screenstepx=CINT(screenx/fwidth-1)
960 countxmax=screenstepx*fwidth+7*min_x
970 screeny=max_y-104 : screenstepy=CINT(screeny/fheight-1)
980 countymax=screenstepy*fheight+104
990 LINE (7*min_x,103)-(19*min_x,19*min_y),0,BF
1000 FOR countx = 7*min_x TO countxmax STEP screenstepx
1010 LINE (countx,103)-(countx,countymax),1
1020 NEXT countx
1030 FOR county = 103 TO countymax STEP screenstepy
1040 LINE (7*min_x,county)-(countxmax,county),1
1050 NEXT county
1060 box_x=screenstepx-2 : box_y=screenstepy-2
1070 cursorx=7*min_x+1 : cursory=104
1080 GOSUB 3100
1090 GOSUB 2880 ' read font info into display
1100 TIMER=0
1110 plotx=1 : ploty=1 : charxy=1
1120 LOCATE 160,15 : ?"X-Position":GOSUB 1830
1130 LOCATE 160,30 : ?"Y-Position":GOSUB 1850
1140 LOCATE 160,45 : ?"Character #":GOSUB 1870
1145 LOCATE 160,60 : ? "Character code ":GOSUB 1880
1150 DO
1160 text$=INKEY$
1170 IF (text$<>"") THEN GOSUB 1270
1180 timerloop=TIMER
1190 IF (timerloop>oldloop+100) THEN
1200 oldloop=oldloop+100
1210 cmode=NOT cmode
1220 GOSUB 1250
1230 ENDIF
```

```
1240 LOOP
1250 LINE (cursorx,cursory)-(cursorx+box_x,cursory+box_y),cmode,BF
1260 RETURN
1270 IF (ASC(text$)=131) THEN
1280 IF (plotx=fwidth) THEN
1290 ' do nothing at right edge already
1300 ELSE
1310 cursorx=cursorx+screenstepx
1320 plotx=plotx+1
1330 GOSUB 1830
1340 ENDIF
1350 IF (plot(charxy,plotx-1,ploty)=0) THEN direction=1:GOSUB 1890
1360 IF(plot(charxy,plotx-1,ploty)=1) THEN direction=1 :GOSUB 1920
1370 ENDIF
1380 ENDIF
1390 IF (ASC(text$)=130) THEN
1400 IF (plotx=1) THEN
1410 ' do nothing at left edge already
1420 ELSE
1430 cursorx=cursorx-screenstepx
1440 plotx=plotx-1
1450 GOSUB 1830
1460 ENDIF
1470 IF (plot(charxy,plotx+1,ploty)=0) THEN direction=-1 : GOSUB 1890
1480 IF(plot(charxy,plotx+1,ploty)=1) THEN direction=-1 :GOSUB 1920
1490 ENDIF
1500 ENDIF
1510 IF (ASC(text$)=128) THEN
1520 IF (ploty=1) THEN
1530 ' do nothing at top edge already
1540 ELSE
1550 cursory=cursory-screenstepy
1560 ploty=ploty-1
1570 GOSUB 1850
1580 ENDIF
1590 IF (plot(charxy,plotx,ploty+1)=0) THEN direction=-1 : GOSUB 1950
1600 IF(plot(charxy,plotx,ploty+1)=1) THEN direction=-1 :GOSUB 1980
1610 ENDIF
1620 ENDIF
1630 IF (ASC(text$)=129) THEN
1640 IF (ploty=fheight) THEN
1650 ' do nothing at bottom edge already
1660 ELSE
1670 cursory=cursory+screenstepy
1680 ploty=ploty+1
1690 GOSUB 1850
1700 IF (plot(charxy,plotx,ploty-1)=0) THEN direction=1 : GOSUB 1950
1710 IF(plot(charxy,plotx,ploty-1)=1) THEN direction=1 :GOSUB 1980
1720 ENDIF
1730 ENDIF
1740 IF(text$="- ") THEN GOSUB 2720
```

```
1745 IF (ASC(text$)=132) THEN GOSUB 5000
1750 IF(text$="e"OR text$="E") THEN
1760 CLS : ? "Hope you found this tool useful!" : END
1770 ENDIF
1775 IF (text$="c" OR text$="C") THEN GOSUB 5500
1776 IF (text$="v" OR text$="V") THEN GOSUB 6500
1777 IF (text$="h" OR text$="H") THEN GOSUB 7500
1780 IF (ASC(text$)=139) THEN GOSUB 2010
1790 IF(text$="+") THEN GOSUB 2800
1800 IF (ASC(text$)=32) THEN GOSUB 2040
1810 IF (text$="s" OR text$="S") THEN GOSUB 2070
1820 IF (text$="l" OR text$="L") THEN GOTO 2260
1830 LOCATE 220,15 : ? "      ":LOCATE 220,15 : ? plotx
1840 RETURN
1850 LOCATE 220,30 : ? "      ":LOCATE 220,30: ? ploty
1860 RETURN
1870 LOCATE 230,45 : ? "      " : LOCATE 230,45 : ? charxy
1880 LOCATE 245,60 : ? "      " : LOCATE 245,60 : ? fstart+charxy-1
1885 RETURN
1890 tempx=cursorx-(direction*screenstepx)
1900 LINE (tempx,cursory)-(tempx+box_x,cursory+box_y),0,BF
1910 RETURN
1920 tempx=cursorx-(direction*screenstepx)
1930 LINE (tempx,cursory)-(tempx+box_x,cursory+box_y),1,BF
1940 RETURN
1950 tempy=cursory-(direction*screenstepy)
1960 LINE (cursorx,tempy)-(cursorx+box_x,tempy+box_y),0,BF
1970 RETURN
1980 tempy=cursory-(direction*screenstepy)
1990 LINE (cursorx,tempy)-(cursorx+box_x,tempy+box_y),1,BF
2000 RETURN
2010 PIXEL(52+plotx,292+ploty)=1
2020 plot(charxy,plotx,ploty)=1
2030 RETURN
2040 plot(charxy,plotx,ploty)=0
2050 PIXEL(52+plotx,292+ploty)=0
2060 RETURN
2070 GOSUB 5300: LOCATE 270,20 : ? "                                "
2075 LOCATE 270,20 : INPUT "Filename to Save : ",savename$
2080 IF (RIGHT$(savename$,3)<>"fnt") THEN savename$=savename$+".fnt"
2090 OPEN savename$ FOR output AS #1
2100 typefile$="save"
2110 ON MM.ERRNO GOTO 3350,3410,3470,3530,3590,3650
2120 PRINT #1,fheight","fwidth","fstart","fstart+fnumber-addchar
2130 FOR numchar=1 TO fnumber
2140 FOR row = 1 TO fheight
2150 temp$=""
2160 FOR column=1 TO fwidth
2170 IF (plot(numchar,column,row)=1) THEN temp$=temp$+"X"
2180 IF (plot(numchar,column,row)=0) THEN temp$=temp$+" "
2190 NEXT column
```

```
2200 PRINT #1,temp$
2210 NEXT row,numchar
2215 IF addchar=0 THEN GOSUB 5100
2220 CLOSE #1
2230 GOSUB 5300
2240 LOCATE 270,20 : ?"save ok"
2250 RETURN
2260 GOSUB 5300: LOCATE 270,20 : ? "
2270 LOCATE 270,20 : INPUT "Filename to load : ",fname$
2280 IF (RIGHT$(fname$,3)<>"fnt") THEN fname$=fname$+".fnt"
2290 OPEN fname$ FOR input AS #2
2300 typefile$="load"
2310 ON MM.ERRNO GOTO 3350,3410,3470,3530,3590,3650
2320 LINE INPUT #2,temp$
2330 ERASE values
2340 DIM values(4)
2350 pointer=1: fwidth =0: fheight=0 :fnumber=0 :fstart=0
2360 FOR a= 1 TO LEN(temp$)
2370 a$=MID$(temp$,a,1)
2380 IF (a$=",") THEN
2390     values(pointer)=VAL(rwidth$)
2400     pointer=pointer+1
2410     rwidth$=""
2420 ELSE
2430     rwidth$=rwidth$+a$
2440 ENDIF
2450 NEXT a
2460 LOCATE 300,45
2480 values(4)= VAL(rwidth$)
2490 LOCATE 300,60
2510 fwidth=values(2)
2520 fheight=VAL(temp$)
2530 fstart=values(3)
2540 fnumber=values(4)-values(3)+1
2550 ERASE plot : GOSUB 3290
2560 DIM plot(fnumber,fwidth,fheight)
2565 GOSUB 3290
2570 FOR a= 1 TO fnumber
2580 FOR c=1 TO fheight
2590 LINE INPUT #2,temp$
2600 FOR b=1 TO fwidth
2610 IF (MID$(temp$,b,1)=CHR$(32)) THEN
2620 plot(a,b,c)=0
2630 ELSE
2640 plot(a,b,c)=1
2650 ENDIF
2660 NEXT b,c,a
2670 CLOSE #2
2680 GOSUB 5300
2690 LOCATE 270,20
2695 ?"load ok"
```

```
2700 GOSUB 3050
2710 GOTO 950
2720 IF (charxy =1) THEN
2730 'rem do nothing
2740 ELSE
2750 charxy=charxy-1
2760 GOSUB 1870
2770 GOSUB 2880
2780 ENDIF
2790 RETURN
2800 IF (charxy =fnumber) THEN
2810 'rem do nothing
2820 ELSE
2830 charxy=charxy+1
2840 GOSUB 1870
2850 GOSUB 2880
2860 ENDIF
2870 RETURN
2880 LINE (52,292)-(52+fwidth,292+fheight),0,BF
2890 a=charxy : IF (a=0) THEN a=1
2900 cursorx=7*min_x+2 : cursory=105
2910 plotx=1 :ploty=1
2920 FOR c=1 TO fheight
2930 FOR b=1 TO fwidth
2940 tempx=cursorx+(b-1)*screenstepx : tempy=cursory+(c-1)*screenstepy
2950 temp2x=tempx+box_x : temp2y=tempy+box_y
2960 IF (plot(a,b,c)=0) THEN
2970 LINE (tempx,tempy)-(temp2x,temp2y),0,BF
2980 ELSE
2990 LINE (tempx,tempy)-(temp2x,temp2y),1,BF
3000 PIXEL(52+b,292+c)=1
3010 ENDIF
3020 NEXT b,c
3030 GOSUB 1870
3040 RETURN
3050 LOCATE 100,15 : ? "      ":LOCATE 20,15: ? "Width ";fwidth
3060 LOCATE 50,30 : ? "      ":LOCATE 20,30: ? "Height ";fheight
3070 LOCATE 130,45 : ? "      ":LOCATE 20,45: ? "Characters in Set ";fnumber
3080 LOCATE 80,60 : ? "      ":LOCATE 20,60: ? "Start Character ";fstart
3090 RETURN
3100 LOCATE 10,110 : ? "Move --- Cursor keys"
3110 LOCATE 10,125 : ? "Set    ___ Left Alt."
3120 LOCATE 10,140 : ? "Erase --- Space"
3130 LOCATE 10,155 : ? "Load    --- l or L"
3140 LOCATE 10,170 : ? "Save     --- s or S"
3150 LOCATE 10,185 : ? "Char Up  ___ +"
3160 LOCATE 10,200 : ? "Char Down --- -"
3165 LOCATE 10,215 : ? "Add Char ___ Insert"
3166 LOCATE 10,230 : ? "Copy Char ___ c or C"
3167 LOCATE 10,245 : ? "Flip Vert. ___ v or V"
3168 LOCATE 10,260 : ? "Flip Horiz. ___ h or H"
```



```
3170 LOCATE 10,275 : ? "Exit      ___ e or E"
3180 LINE (4,290)-(6*min_x-2,max_y-6),1,B
3190 LOCATE 10,300: ?"Pixel":LOCATE 10,320: ? "Size"
3200 LINE (50,291)-(6*min_x-4,max_y-70),0,BF
3210 LINE (50,290)-(50+fwidth+4,290+fheight+4),1,B
3220 LOCATE 30,Max_y-65
3230 ? "Code Written by"
3240 LOCATE 15,Max_y-45
3250 ? "Dennis Wyatt ";:FONT LOAD "copyr.fnt" AS #6 :FONT #6
3252 ? " "; : FONT #1 : FONT unload #6 : ? " 2011"
3260 LOCATE 15,Max_y-25
3270 ? "dpwyatt@iinet.net.au"
3280 RETURN
3290 'pause routine for array erasure completion
3300 TIMER=0
3310 DO WHILE timertemp<3000
3320 timertemp=TIMER
3330 LOOP
3340 RETURN
3350 LOCATE 270,40: ? "ERROR -No SD Card present"
3352 LOCATE 270,55: ? "Please insert card and re-enter"
3354 LOCATE 270,70: ? "Filename at the prompt above"
3356 PAUSE 2000
3360 IF (typefile$="save") THEN
3370 GOTO 2070
3380 ELSE
3390 GOTO 2260
3400 ENDIF
3410 LOCATE 270,40: ? "Card is Write Protected"
3412 LOCATE 270,55: ? "Please write enable and re-enter"
3414 LOCATE 270,70: ? "Filename at the prompt above"
3420 IF (typefile$="save") THEN
3430 GOTO 2070
3440 ELSE
3450 GOTO 2260
3460 ENDIF
3470 LOCATE 270,40: ? "not enough space"
3472 LOCATE 270,55: ? "Please rectify and re-enter"
3474 LOCATE 270,70: ? "Filename at the prompt above"
3480 IF (typefile$="save") THEN
3490 GOTO 2070
3500 ELSE
3510 GOTO 2260
3520 ENDIF
3530 LOCATE 270,40 : ? "All root dir. taken"
3540 IF (typefile$="save") THEN
3550 GOTO 2070
3560 ELSE
3570 GOTO 2260
3580 ENDIF
3590 LOCATE 270,40 : ? "Invalid Filename"
```

```
3592 LOCATE 270,55: ? "Please rectify and re-enter"
3594 LOCATE 270,70: ? "Filename at the prompt above"
3600 IF (typefile$="save") THEN
3610 GOTO 2070
3620 ELSE
3630 GOTO 2260
3640 ENDIF
3650 LOCATE 270,40: ? "Cannot find File"
3652 LOCATE 270,55: ? "Please re-enter"
3654 LOCATE 270,70: ? "Filename at the prompt above"
3660 IF (typefile$="save") THEN
3670 GOTO 2070
3680 ELSE
3690 GOTO 2260
3700 ENDIF
5000 ' insert character function
5001 addchar=0
5010 savename$="temp.fnt"
5015 GOSUB 2090
5020 fname$="temp.fnt"
5030 GOSUB 2290
5088 addchar=1
5089 KILL "temp.fnt"
5090 RETURN
5100 temp$=""
5105 FOR column=1 TO fwidth
5110 temp$=temp$+" "
5120 NEXT column
5130 FOR row = 1 TO fheight
5140 PRINT #1,temp$
5150 NEXT row
5160 RETURN
5300 LOCATE 270,20 : LINE (270,20)-(19*min_x,80),0,bf
5310 RETURN
5500 ' Copy function
5510 GOSUB 5300
5520 LOCATE 270,20
5530 INPUT "Copy which Character # ";copychar
5535 IF copychar>fnumber THEN
5536 LOCATE 270,35 : ? "Not that many Char." : GOTO 5520
5537 ENDIF
5540 LOCATE 270,35
5550 INPUT "To which Character # ";tochar
5555 IF tochar>fnumber THEN
5556 LOCATE 270,50 : ? "Not that many Char." : GOTO 5540
5557 ENDIF
5560 FOR copy1= 1 TO fwidth
5570 FOR copy2= 1 TO fheight
5580 plot(tochar,copy1,copy2)=plot(copychar,copy1,copy2)
5590 NEXT copy2,copy1
5600 LOCATE 270,50 : ? " "
```

```
5610 LOCATE 270,50 : ? "done"
6000 RETURN
6500 ' flip vertical function
6510 FOR flipV = 1 TO INT(fheight/2)
6520 FOR flipV1 = 1 TO fwidth
6530 flipvtemp=plot(charxy,flipV1,flipV)
6540 plot(charxy,flipV1,flipV)=plot(charxy,flipV1,fheight-flipV+1)
6550 plot(charxy,flipV1,fheight-flipV+1)=flipvtemp
6560 NEXT flipV1 : NEXT flipV
6570 GOSUB 5300
6580 LOCATE 270,20
6590 ? "Vertical Flip Done"
6600 GOSUB 2880
7000 RETURN
7500 ' flip horizontal function
7510 FOR flipH = 1 TO INT(fwidth/2)
7520 FOR flipH1 = 1 TO fheight
7530 fliphtemp=plot(charxy,flipH,flipH1)
7540 plot(charxy,flipH,flipH1)=plot(charxy,fwidth-flipH+1,flipH1)
7550 plot(charxy,fwidth-flipH+1,flipH1)=fliphtemp
7560 NEXT flipH1
7570 NEXT flipH
7580 GOSUB 5300
7590 LOCATE 270,20
7600 ? "Horizontal Flip Done"
7610 GOSUB 2880
8000 RETURN
```

MaxF3_0.BAS:

```
'*****
'*           Maxi-Font           *
'*   Font Editor by Dennis Wyatt *
'*   Using mmBasic for the       *
'*   Maximite computer           *
'*****
' get maximum screen dimension-split into 20 parts
Max_x=MM.HRes : Min_x=Cint(Max_x/20)
Max_y=MM.VRes : Min_y=Cint(max_y/20)
Cls
addchar=1
'*****
' Set to Capture SD card errors
'*****
Option Error Continue
'*****
'*           Show front page
'*****
Font Load "gothic.fnt" As #4 : Font #4
```

```
Locate 0,1 : ?"2000000000000004"
Locate 0,33 : ? "1" : Locate max_x-32,33 : ? "1"
Locate 0,65 : ? "1" : Locate max_x-32,65 : ? "1"
Locate 0,92
?"3000000000000005"
'*****
'*    return to font 1 and unload gothic font
'*    need the memory
'*****
Font #1 : Font unload #4
Font Load "invade.fnt" As #5 :Font #5
Locate 40,57 : ? "0 2 4": Locate 15*min_x,57: ? "1 3 5"
Font #1
Locate 9*min_x,37: ? "Welcome to"
Font #1,3
Locate 7.3*min_x,52 : ? "MaxiFont"
Line (0,124)-(max_x,124),1
Font #1,1
Locate 2*min_x,130
? "This program will help you to design some interesting fonts, with"
Locate 2*min_x,145
? "the added scope of allowing the design of program sprites, for the"
Locate 2*min_x,160
? "inclusion into some Maximite games, or just make some gothic borders."
Locate 2*min_x,190
? "The design of the font should take into account the available"
Locate 2*min_x,205
? "memory in the system. This is about 37 Kb, for arrays, used in the"
Locate 2*min_x,220
?"design process. This limits the number of characters, in the editor"
Locate 2*min_x,235
?"at one time. A 255 pixel wide font by 64 pixels high can only have"
Locate 2*min_x,250
?"2 Characters, if I rewrite the code. A 10 pixel by 10 pixel font can"
Locate 2*min_x,265
?"have only 70 characters in the editor at one time."
Locate 2*min_x,280
?"You can always append the files together later. Once saved the font"
Locate 2*min_x,295
?"takes up one quarter the size."
Locate 2*min_x,310
?"It would be easier to design a smaller font and just piece them"
Locate 2*min_x,325
?"together to form one large picture."
Locate 2*min_x,340
?"If you encounter an out of memory error, then reduce the number of "
Locate 2*min_x,355
?"Characters"
Locate 9*min_x,385 : ?"press any key"
Do While (Inkey$="")
Font #5
```

```
Pause 100
frontpage=Not frontpage
If (frontpage=1) Then
Locate 40,57 : ? "1 3 5": Locate 15*min_x,57: ? "0 2 4"
Else
Locate 40,57 : ? "0 2 4": Locate 15*min_x,57: ? "1 3 5"
EndIf
Loop
For a=1 To 5
Pause 150
On a GoTo explode1,explode2,explode3,explode4,explode5
explode1:
Locate 40,57 : ? "6 6 6": Locate 15*min_x,57: ? "6 6 6":Next a
explode2:
Locate 40,57 : ? "7 7 7": Locate 15*min_x,57: ? "7 7 7":Next a
explode3:
Locate 40,57 : ? "8 8 8": Locate 15*min_x,57: ? "8 8 8":Next a
explode4:
Locate 40,57 : ? "9 9 9": Locate 15*min_x,57: ? "9 9 9":Next a
explode5:
t$=Chr$(59)+" "+Chr$(59)+" "+Chr$(59)
Locate 40,57 : ? t$: Locate 15*min_x,57: ? t$ :Next a
Font #1 : Font unload #5
t$=""
Line(2*min_x,130)-(max_x,max_y),0,BF
Locate 0,130
Fontwidth:
? t$; : Input "Width of Font? ",fwidth
If (fwidth>255) Then
? t$+"Cannot be bigger than 255 pixels": GoTo Fontwidth
EndIf
Fontheight:
? t$; : Input "Height of Font ? ",fheight
If (fheight > 64) Then
? t$+"Cannot be higher than 64 pixels":GoTo Fontheight
EndIf
If (fheight =0) Then
? t$+"Cannot be lower than 1 pixels":GoTo Fontheight
EndIf
Fontnumber:
? t$; : Input "Number of Characters ? ",fnumber
If (fnumber =0) Then
? t$+"Cannot be less than 1 Character":GoTo Fontnumber
EndIf
If (fnumber >128) Then
? t$+"Cannot be more than 128 Characters":GoTo Fontnumber
EndIf
Fontstart:
? t$; : Input "Start Character set at ?";fstart
If (fstart<32) Then
? t$+"Cannot be less than 32 ": GoTo Fontstart
```

```
EndIf
Cls
Option base 0
Dim values(4)
Dim plot(fnumber,fwidth,fheight)
Line (0,0)-(max_x-1,100),1,BF
Line (10,10)-(max_x-11,90),0,BF
GoSub UpdateFontDetails
Line (0,101)-(max_x-1,max_y-1),1,B
Line (2,103)-(6*min_x,max_y-3),1,B
screenx=Max_x-1-(7*min_x)
Setupgrid:
screenstepx=Cint(screenx/fwidth-1)
countxmax=screenstepx*fwidth+7*min_x
screeny=max_y-104 : screenstepy=Cint(screeny/fheight-1)
countymax=screenstepy*fheight+104
Line (7*min_x,103)-(19*min_x,19*min_y),0,BF
For countx = 7*min_x To countxmax Step screenstepx
Line (countx,103)-(countx,countymax),1
Next countx
For county = 103 To countymax Step screenstepy
Line (7*min_x,county)-(countxmax,county),1
Next county
box_x=screenstepx-2 : box_y=screenstepy-2
cursorx=7*min_x+1 : cursory=104
GoSub Instructions
GoSub Displayfontchar          ' read font info into display
Timer=0
plotx=1 : ploty=1 : charxy=1
Locate 160,15 : ?"X-Position":GoSub update_x_position
Locate 160,30 : ?"Y-Position":GoSub update_y_position
Locate 160,45 : ?"Character #":GoSub update_char_number
Locate 160,60 : ? "Character code ":GoSub update_char_code
Do
text$=Inkey$
If (text$<>"") Then GoSub Continueloop
timerloop=Timer
If (timerloop>oldloop+100) Then
oldloop=oldloop+100
cmode=Not cmode
GoSub flash_cursor_pos
EndIf
Loop
flash_cursor_pos:
Line (cursorx,cursory)-(cursorx+box_x,cursory+box_y),cmode,BF
Return
Continueloop:
If (Asc(text$)=131) Then
If (plotx=fwidth) Then
' do nothing at right edge already
Else
```

```
cursorx=cursorx+screenstepx
plotx=plotx+1
GoSub update_x_position
EndIf
If (plot(charxy,plotx-1,ploty)=0) Then direction=1:GoSub clear_right
If(plot(charxy,plotx-1,ploty)=1) Then direction=1 :GoSub draw_right
EndIf
EndIf
If (Asc(text$)=130) Then
If (plotx=1) Then
' do nothing at left edge already
Else
cursorx=cursorx-screenstepx
plotx=plotx-1
GoSub update_x_position
EndIf
If (plot(charxy,plotx+1,ploty)=0) Then direction=-1 : GoSub clear_right
If(plot(charxy,plotx+1,ploty)=1) Then direction=-1 :GoSub draw_right
EndIf
EndIf
If (Asc(text$)=128) Then
If (ploty=1) Then
' do nothing at top edge already
Else
cursory=cursory-screenstepy
ploty=ploty-1
GoSub update_y_position
EndIf
If (plot(charxy,plotx,ploty+1)=0) Then direction=-1 : GoSub clear_down
If(plot(charxy,plotx,ploty+1)=1) Then direction=-1 :GoSub clear_up
EndIf
EndIf
If (Asc(text$)=129) Then
If (ploty=fheight) Then
' do nothing at bottom edge already
Else
cursory=cursory+screenstepy
ploty=ploty+1
GoSub update_y_position
If (plot(charxy,plotx,ploty-1)=0) Then direction=1 : GoSub clear_down
If(plot(charxy,plotx,ploty-1)=1) Then direction=1 :GoSub clear_up
EndIf
EndIf
If(text$="-") Then GoSub Char_up
If (Asc(text$)=132) Then GoSub Insert_char
If(text$="e"Or text$="E") Then
Cls : ? "Hope you found this tool useful!" : End
EndIf
If (text$="c" Or text$="C") Then GoSub Copy_char
If (text$="v" Or text$="V") Then GoSub Vertical_flip
If (text$="h" Or text$="H") Then GoSub horizontal_flip
```

```
If (Asc(text$)=139) Then GoSub Set_plot
If(text$="+") Then GoSub Char_down
If (Asc(text$)=32) Then GoSub Clear_plot
If (text$="s" Or text$="S") Then GoSub Save_font
If (text$="l" Or text$="L") Then GoTo Load_font
update_x_position:
  Locate 220,15 : ? plotx;"      "
  Return

update_y_position:
  Locate 220,30: ? ploty;"      "
  Return

update_char_number:
  Locate 230,45 : ? charxy;"      "

update_char_code:
  Locate 245,60 : ? fstart+charxy-1;"      "
  Return

clear_right:
  tempx=cursorx-(direction*screenstepx)
  Line (tempx,cursory)-(tempx+box_x,cursory+box_y),0,BF
  Return

draw_right:
  tempx=cursorx-(direction*screenstepx)
  Line (tempx,cursory)-(tempx+box_x,cursory+box_y),1,BF
  Return

clear_down:
  tempy=cursory-(direction*screenstepy)
  Line (cursorx,tempy)-(cursorx+box_x,tempy+box_y),0,BF
  Return

clear_up:
  tempy=cursory-(direction*screenstepy)
  Line (cursorx,tempy)-(cursorx+box_x,tempy+box_y),1,BF
  Return

Set_plot:
  Pixel(52+plotx,292+ploty)=1
  plot(charxy,plotx,ploty)=1
  Return

Clear_plot:
  plot(charxy,plotx,ploty)=0
  Pixel(52+plotx,292+ploty)=0
  Return

Save_font:
```



```

    GoSub Clear_box : Locate 270,20 : ? "
    Locate 270,20 : Input "Filename to Save : ",savename$
    If (Right$(savename$,3)<>"fnt") Then savename$=savename$+".fnt"
Insert_save_font:
    Open savename$ For output As #1
    typefile$="save"
    On MM.Errno GoTo No_sd_card, Card_protected, no_space_left,
All_root_gone, Invalid_filename, Cannot_find_file
    Print #1,fheight", "fwidth", "fstart", "fstart+fnumber-addchar
    For numchar=1 To fnumber
        For row = 1 To fheight
            temp$=""
            For column=1 To fwidth
                If (plot(numchar,column,row)=1) Then temp$=temp$+"X"
                If (plot(numchar,column,row)=0) Then temp$=temp$+" "
            Next column
            Print #1,temp$
        Next row
    next numchar
    If addchar=0 Then GoSub Insert_Char_now
    Close #1
    GoSub Clear_box
    Locate 270,20 : ?"save ok"
    Return

Load_font:
    GoSub Clear_box : Locate 270,20 : ? "
    Locate 270,20 : Input "Filename to load : ",fname$
    If (Right$(fname$,3)<>"fnt") Then fname$=fname$+".fnt"
Insert_Load_Font:
    Open fname$ For input As #2
    typefile$="load"
    On MM.Errno GoTo No_sd_card, Card_protected, no_space_left,
All_root_gone, Invalid_filename, Cannot_find_file
    Line Input #2,temp$
    Erase values
        Dim values(4)
    pointer=1: fwidth =0: fheight=0 :fnumber=0 :fstart=0
    For a= 1 To Len(temp$)
        a$=Mid$(temp$,a,1)
        If (a$=",") Then
            values(pointer)=Val(rwidth$)
            pointer=pointer+1
            rwidth$=""
        Else
            rwidth$=rwidth$+a$
        EndIf
    Next a
    values(4)= Val(rwidth$)
    fwidth=values(2)
    fheight=Val(temp$)

```

```
fstart=values(3)
fnumber=values(4)-values(3)+1
Erase plot : GoSub My_pause
  Dim plot(fnumber,fwidth,fheight)
GoSub My_pause
For a= 1 To fnumber
  For c=1 To fheight
    Line Input #2,temp$
    For b=1 To fwidth
      If (Mid$(temp$,b,1)=Chr$(32)) Then
        plot(a,b,c)=0
      Else
        plot(a,b,c)=1
      EndIf
    Next b
  nextc
next a
Close #2
GoSub Clear_box
Locate 270,20
  ?"load ok"
GoSub UpdateFontDetails
GoTo Setupgrid
```

Char_up:

```
If (charxy =1) Then
  'rem do nothing
Else
  charxy=charxy-1
  GoSub update_char_number
  GoSub Displayfontchar
EndIf
Return
```

Char_down:

```
If (charxy =fnumber) Then
  'rem do nothing
Else
  charxy=charxy+1
  GoSub update_char_number
  GoSub Displayfontchar
EndIf
Return
```

Displayfontchar:

```
Line (52,292)-(52+fwidth,292+fheight),0,BF
a=charxy : If (a=0) Then a=1
cursorx=7*min_x+2 : cursory=105
plotx=1 :ploty=1
For c=1 To fheight
  For b=1 To fwidth
```

```

        tempx=cursorx+(b-1)*screenstepx :
tempy=cursory+(c-1)*screenstepy
        temp2x=tempx+box_x : temp2y=tempy+box_y
        If (plot(a,b,c)=0) Then
            Line (tempx,tempy)-(temp2x,temp2y),0,BF
        Else
            Line (tempx,tempy)-(temp2x,temp2y),1,BF
            Pixel(52+b,292+c)=1
        EndIf
    Next b
next c
GoSub update_char_number
Return

```

UpdateFontDetails:

```

    Locate 100,15 : ? "      ":Locate 20,15: ? "Width ";fwidth
    Locate 50,30 : ? "      ":Locate 20,30: ? "Height ";fheight
    Locate 130,45 : ? "      ":Locate 20,45: ? "Characters in Set ";fnumber
    Locate 80,60 : ? "      ":Locate 20,60: ? "Start Character ";fstart
    Return

```

Instructions:

```

    Locate 10,110 : ? "Move   --- Cursor keys"
    Locate 10,125 : ? "Set      ___ Left Alt."
    Locate 10,140 : ? "Erase    --- Space"
    Locate 10,155 : ? "Load      --- l or L"
    Locate 10,170 : ? "Save      --- s or S"
    Locate 10,185 : ? "Char Up   ___ +"
    Locate 10,200 : ? "Char Down --- -"
    Locate 10,215 : ? "Add Char  ___ Insert"
    Locate 10,230 : ? "Copy Char  ___ c or C"
    Locate 10,245 : ? "Flip Vert.  ___ v or V"
    Locate 10,260 : ? "Flip Horiz. ___ h or H"
    Locate 10,275 : ? "Exit      ___ e or E"
    Line (4,290)-(6*min_x-2,max_y-6),1,B
    Locate 10,300: ? "Pixel":Locate 10,320: ? "Size"
    Line (50,291)-(6*min_x-4,max_y-70),0,BF
    Line (50,290)-(50+fwidth+4,290+fheight+4),1,B
    Locate 30,Max_y-65
    ? "Code Written by"
    Locate 15,Max_y-45
    ? "Dennis Wyatt ";:Font Load "copyr.fnt" As #6 :Font #6
    ? " "; : Font #1 : Font unload #6 : ? " 2011"
    Locate 15,Max_y-25
    ? "dpwyatt@iinet.net.au"
    Return

```

My_pause: 'pause routine for array erasure completion

```

    Timer=0
    Do While timertemp<3000
        timertemp=Timer
    
```

```
oop
Return
```

```
No_sd_card:
  Locate 270,40: ? "ERROR -No SD Card present"
  Locate 270,55: ? "Please insert card and re-enter"
  Locate 270,70: ? "Filename at the prompt above"
  Pause 2000
  If (typefile$="save") Then
    GoTo Save_font
  Else
    GoTo Load_font
  EndIf
```

```
Card_protected:
  Locate 270,40: ? "Card is Write Protected"
  Locate 270,55: ? "Please write enable and re-enter"
  Locate 270,70: ? "Filename at the prompt above"
  If (typefile$="save") Then
    GoTo Save_font
  Else
    GoTo Load_font
  EndIf
```

```
no_space_left:
  Locate 270,40: ? "not enough space"
  Locate 270,55: ? "Please rectify and re-enter"
  Locate 270,70: ? "Filename at the prompt above"
  If (typefile$="save") Then
    GoTo Save_font
  Else
    GoTo Load_font
  EndIf
```

```
All_root_gone:
  Locate 270,40 : ? "All root dir. taken"
  If (typefile$="save") Then
    GoTo Save_font
  Else
    GoTo Load_font
  EndIf
```

```
Invalid_filename:
  Locate 270,40 : ? "Invalid Filename"
  Locate 270,55: ? "Please rectify and re-enter"
  Locate 270,70: ? "Filename at the prompt above"
  If (typefile$="save") Then
    GoTo Save_font
  Else
    GoTo Load_font
  EndIf
```

```
Cannot_find_file:
```

```
Locate 270,40: ? "Cannot find File"
Locate 270,55: ? "Please re-enter"
Locate 270,70: ? "Filename at the prompt above"
If (typefile$="save") Then
    GoTo Save_font
Else
    GoTo Load_font
EndIf
```

```
Insert_char:      ' insert character function
addchar=0
savename$="temp.fnt"
    GoSub Insert_save_font
fname$="temp.fnt"
    GoSub Insert_Load_Font
addchar=1
Kill "temp.fnt"
Return
```

```
Insert_Char_now:
temp$=""
For column=1 To fwidth
    temp$=temp$+" "
Next column
For row = 1 To fheight
    Print #1,temp$
Next row
Return
```

```
Clear_box:
Locate 270,20 : Line (270,20)-(19*min_x,80),0,bf
Return
```

```
Copy_char:      ' Copy function
GoSub Clear_box
```

```
Copy_input1_error:
Locate 270,20
Input "Copy which Character # ";copychar
If copychar>fnumber Then
    Locate 270,35 : ? "Not that many Char."
    GoTo Copy_input1_error
EndIf
```

```
Copy_input2_error:
Locate 270,35
Input "To which Character # ";tochar
If tochar>fnumber Then
    Locate 270,50 : ? "Not that many Char." : GoTo Copy_input2_error
EndIf
For copy1= 1 To fwidth
    For copy2= 1 To fheight
```

```

        plot(tochar,copy1,copy2)=plot(copychar,copy1,copy2)
    Next copy2
next copy1
Locate 270,50 : ? "
Locate 270,50 : ? "done"
Return

Vertical_flip:      ' flip vertical function
    For flipV = 1 To Int(fheight/2)
        For flipV1 = 1 To fwidth
            flipvtemp=plot(charxy,flipV1,flipV)
            plot(charxy,flipV1,flipV)=plot(charxy,flipV1,fheight-flipV+1)
            plot(charxy,flipV1,fheight-flipV+1)=flipvtemp
        Next flipV1
    Next flipV
    GoSub Clear_box
    Locate 270,20
    ? "Vertical Flip Done"
    GoSub Displayfontchar
    Return

horizontal_flip:    ' flip horizontal function
    For flipH = 1 To Int(fwidth/2)
        For flipH1 = 1 To fheight
            fliphtemp=plot(charxy,flipH,flipH1)
            plot(charxy,flipH,flipH1)=plot(charxy,fwidth-flipH+1,flipH1)
            plot(charxy,fwidth-flipH+1,flipH1)=fliphtemp
        Next flipH1
    Next flipH
    GoSub Clear_box
    Locate 270,20
    ? "Horizontal Flip Done"
    GoSub Displayfontchar
    Return

```

MaxF3_1.BAS:

```

'*****
'*           Maxi-Font           *
'*   Font Editor by Dennis Wyatt *
'*       Using mmBasic for the   *
'*           Maximite computer   *
'*           V 3.1               *
'*****
' get maximum screen dimension-split into 20 parts
Max_x=MM.HRes : Min_x=Cint(Max_x/20)
Max_y=MM.VRes : Min_y=Cint(max_y/20)
Cls
addchar=1

```

```
'*****
' Set to Capture SD card errors
'*****
Option Error Continue
'*****
'*      Show front page
'*****
Font Load "gothic.fnt" As #4 : Font #4
Print @(0,1) "2000000000000004"
Print @(0,33) "1"
Print @(max_x-32,33) "1"
Print @(0,65) "1"
Print @(max_x-32,65) "1"
Print @(0,92) "3000000000000005"
'*****
'*      return to font 1 and unload gothic font
'*      need the memory
'*****
Font #1
Font unload #4
Font Load "invade.fnt" As #5
Font #5
Print @(40,57) "0  2  4"
Print @(15*min_x,57) "1  3  5"
Font #1
Print @(9*min_x,37) "Welcome to"
Font #1,3
Print @(7.3*min_x,52)"MaxiFont"
Line (0,124)-(max_x,124),1
Font #1,1
Print @(2*min_x,130)"This program will help you to design some
interesting fonts, with"
Print @(2*min_x,145)"the added scope of allowing the design of program
sprites, for the"
Print @(2*min_x,160)"inclusion into some Maximite games, or just make
some gothic borders."
Print @(2*min_x,190)"The design of the font should take into account the
available"
Print @(2*min_x,205)"memory in the system. This is about 37 Kb, for
arrays, used in the"
Print @(2*min_x,220)"design process. This limits the number of
characters, in the editor"
Print @(2*min_x,235)"at one time. A 255 pixel wide font by 64 pixels high
can only have"
Print @(2*min_x,250)"2 Characters, if I rewrite the code. A 10 pixel by
10 pixel font can"
Print @(2*min_x,265)"have only 70 characters in the editor at one time."
Print @(2*min_x,280)"You can always append the files together later. Once
saved the font"
Print @(2*min_x,295)"takes up one quarter the size."
Print @(2*min_x,310)"It would be easier to design a smaller font and just
```

```
piece them"
  Print @(2*min_x,325)"together to form one large picture."
  Print @(2*min_x,340)"If you encounter an out of memory error, then reduce
the number of "
  Print @(2*min_x,355)"Characters"
  Print @(9*min_x,385)"press any key"
  Do While (Inkey$="")
  Font #5
  Pause 100
  frontpage=Not frontpage
  If (frontpage=1) Then
  Print @(40,57)"1 3 5" @(15*min_x,57)"0 2 4"
  Else
  Print @(40,57)"0 2 4" @(15*min_x,57)"1 3 5"
  EndIf
  Loop
  For a=1 To 5
  Pause 150
  On a GoTo explode1,explode2,explode3,explode4,explode5
explode1:
  Print @(40,57)"6 6 6" @(15*min_x,57)"6 6 6"
  Next a
explode2:
  Print @(40,57)"7 7 7" @(15*min_x,57)"7 7 7"
  Next a
explode3:
  Print @(40,57)"8 8 8" @(15*min_x,57)"8 8 8"
  Next a
explode4:
  Print @(40,57)"9 9 9" @(15*min_x,57)"9 9 9"
  Next a
explode5:
  t$=Chr$(59)+" "+Chr$(59)+" "+Chr$(59)
  Print @(40,57)t$ @(15*min_x,57) t$
  Next a
  Font #1
  Font unload #5
  t$=" "
  Line(2*min_x,130)-(max_x,max_y),0,BF
  Print @(0,130)"";
Fontwidth:
  ? t$;
  Input "Width of Font? ",fwidth
  If (fwidth>255) Then
    ? t$+"Cannot be bigger than 255 pixels"
    GoTo Fontwidth
  EndIf
Fontheight:
  ? t$;
  Input "Height of Font ? ",fheight
  If (fheight > 64) Then
```



```
        ? t$+"Cannot be higher than 64 pixels"
        GoTo Fontheight
    EndIf
    If (fheight =0) Then
        ? t$+"Cannot be lower than 1 pixels"
        GoTo Fontheight
    EndIf
Fontnumber:
    ? t$;
    Input "Number of Characters ? ",fnumber
    If (fnumber =0) Then
        ? t$+"Cannot be less than 1 Character"
        GoTo Fontnumber
    EndIf
    If (fnumber >128) Then
        ? t$+"Cannot be more than 128 Characters"
        GoTo Fontnumber
    EndIf
Fontstart:
    ? t$; : Input "Start Character set at ?";fstart
    If (fstart<32) Then
        ? t$+"Cannot be less than 32 "
        GoTo Fontstart
    EndIf

Cls

Option base 0

Dim values(4)

Dim plot(fnumber,fwidth,fheight)

Line (0,0)-(max_x-1,100),1,BF
Line (10,10)-(max_x-11,90),0,BF

GoSub UpdateFontDetails

Line (0,101)-(max_x-1,max_y-1),1,B
Line (2,103)-(6*min_x,max_y-3),1,B

screenx=Max_x-1-(7*min_x)

Setupgrid:
    screenstepx=Cint(screenx/fwidth-1)
    countxmax=screenstepx*fwidth+7*min_x
    screeny=max_y-104
    screenstepy=Cint(screeny/fheight-1)
    countymax=screenstepy*fheight+104
    Line (7*min_x,103)-(19*min_x,19*min_y),0,BF
    For countx = 7*min_x To countxmax Step screenstepx
```

```
Line (countx,103)-(countx,countymax),1
Next countx
For county = 103 To countymax Step screenstepy
  Line (7*min_x, county) - (countxmax, county), 1
Next county
box_x=screenstepx-2
box_y=screenstepy-2
cursorx=7*min_x+1
cursory=104
GoSub Instructions
GoSub Displayfontchar          ' read font info into display
Timer=0
plotx=1 : ploty=1 : charxy=1
Print @(160,15)"X-Position"
  GoSub update_x_position
Print @(160,30)"Y-Position"
  GoSub update_y_position
Print @(160,45)"Character #"
  GoSub update_char_number
Print @(160,60)"Character code "
  GoSub update_char_code
Do
  text$=Inkey$
  If (text$<>"") Then GoSub Continueloop
    timerloop=Timer
    If (timerloop>oldloop+100) Then
      oldloop=oldloop+100
      cmode=Not cmode
      GoSub flash_cursor_pos
    EndIf
Loop

flash_cursor_pos:
  Line (cursorx,cursory)-(cursorx+box_x,cursory+box_y),cmode,BF
Return

Continueloop:
  If (Asc(text$)=131) Then
    If (plotx=fwidth) Then
      ' do nothing at right edge already
    Else
      cursorx=cursorx+screenstepx
      plotx=plotx+1
      GoSub update_x_position
    EndIf
    If (plot(charxy,plotx-1,ploty)=0) Then direction=1 : GoSub
clear_right
      If(plot(charxy,plotx-1,ploty)=1) Then direction=1 : GoSub
draw_right
    EndIf
  EndIf
```

```
If (Asc(text$)=130) Then
  If (plotx=1) Then
    ' do nothing at left edge already
  Else
    cursorx=cursorx-screenstepx
    plotx=plotx-1
    GoSub update_x_position
  EndIf
  If (plot(charxy,plotx+1,ploty)=0) Then direction=-1 : GoSub
clear_right
  If(plot(charxy,plotx+1,ploty)=1) Then direction=-1 :GoSub draw_right
  EndIf
EndIf

If (Asc(text$)=128) Then
  If (ploty=1) Then
    ' do nothing at top edge already
  Else
    cursory=cursory-screenstepy
    ploty=ploty-1
    GoSub update_y_position
  EndIf
  If (plot(charxy,plotx,ploty+1)=0) Then direction=-1 : GoSub
clear_down
  If(plot(charxy,plotx,ploty+1)=1) Then direction=-1 :GoSub clear_up
  EndIf
EndIf

If (Asc(text$)=129) Then
  If (ploty=fheight) Then
    ' do nothing at bottom edge already
  Else
    cursory=cursory+screenstepy
    ploty=ploty+1
    GoSub update_y_position
    If (plot(charxy,plotx,ploty-1)=0) Then direction=1 : GoSub
clear_down
    If(plot(charxy,plotx,ploty-1)=1) Then direction=1 :GoSub
clear_up
  EndIf
EndIf

If(text$="-") Then GoSub Char_up

If (Asc(text$)=132) Then GoSub Insert_char

If(text$="e"Or text$="E") Then
  Cls
  ? "Hope you found this tool useful!"
  End
EndIf
```

```
If (text$="c" Or text$="C") Then GoSub Copy_char

If (text$="v" Or text$="V") Then GoSub Vertical_flip

If (text$="h" Or text$="H") Then GoSub horizontal_flip

If (Asc(text$)=139) Then GoSub Set_plot

If(text$="+") Then GoSub Char_down

If (Asc(text$)=32) Then GoSub Clear_plot

If (text$="s" Or text$="S") Then GoSub Save_font

If (text$="l" Or text$="L") Then GoTo Load_font

update_x_position:
  Print @(220,15)plotx;"      "
  Return

update_y_position:
  Print @(220,30)ploty;"      "
  Return

update_char_number:
  Print @(230,45)charxy;"      "
  Return

update_char_code:
  Print @(245,60)fstart+charxy-1;"      "
  Return

clear_right:
  tempx=cursorx-(direction*screenstepx)
  Line (tempx,cursory)-(tempx+box_x,cursory+box_y),0,BF
  Return

draw_right:
  tempx=cursorx-(direction*screenstepx)
  Line (tempx,cursory)-(tempx+box_x,cursory+box_y),1,BF
  Return

clear_down:
  tempy=cursory-(direction*screenstepy)
  Line (cursorx,tempy)-(cursorx+box_x,tempy+box_y),0,BF
  Return

clear_up:
  tempy=cursory-(direction*screenstepy)
  Line (cursorx,tempy)-(cursorx+box_x,tempy+box_y),1,BF
  Return
```

```
Set_plot:
  Pixel(52+plotx,292+ploty)=1
  plot(charxy,plotx,ploty)=1
  Return

Clear_plot:
  plot(charxy,plotx,ploty)=0
  Pixel(52+plotx,292+ploty)=0
  Return

Save_font:
  GoSub Clear_box
  Print @(270,20) "
  Print @(270,20);
  Input "Filename to Save : ",savename$
  If (Right$(savename$,3)<>"fnt") Then savename$=savename$+".fnt"

Insert_save_font:
  Open savename$ For output As #1
  typefile$="save"
  On MM.Errno GoTo No_sd_card, Card_protected, no_space_left,
All_root_gone, Invalid_filename, Cannot_find_file
  Print #1,fheight",fwidth",fstart",fstart+fnumber-addchar
  For numchar=1 To fnumber
    For row = 1 To fheight
      temp$=""
      For column=1 To fwidth
        If (plot(numchar,column,row)=1) Then temp$=temp$+"X"
        If (plot(numchar,column,row)=0) Then temp$=temp$+" "
      Next column
      Print #1,temp$
    Next row
  next numchar

  If addchar=0 Then GoSub Insert_Char_now
  Close #1
  GoSub Clear_box
  Print @(270,20)"save ok"
  Return

Load_font:
  GoSub Clear_box
  Print @(270,20) "
  Print @(270,20);
  Input "Filename to load : ",fname$
  If (Right$(fname$,3)<>"fnt") Then fname$=fname$+".fnt"

Insert_Load_Font:
  Open fname$ For input As #2
  typefile$="load"
  On MM.Errno GoTo No_sd_card, Card_protected, no_space_left,
```

```
All_root_gone, Invalid_filename, Cannot_find_file
Line Input #2,temp$
Erase values
  Dim values(4)
pointer=1: fwidth =0: fheight=0 :fnumber=0 :fstart=0
For a= 1 To Len(temp$)
  a$=Mid$(temp$,a,1)
  If (a$=",") Then
    values(pointer)=Val(rwidth$)
    pointer=pointer+1
    rwidth$=""
  Else
    rwidth$=rwidth$+a$
  EndIf
Next a
values(4)= Val(rwidth$)
fwidth=values(2)
fheight=Val(temp$)
fstart=values(3)
fnumber=values(4)-values(3)+1
Erase plot
GoSub My_pause
  Dim plot(fnumber,fwidth,fheight)
GoSub My_pause
For a= 1 To fnumber
  For c=1 To fheight
    Line Input #2,temp$
    For b=1 To fwidth
      If (Mid$(temp$,b,1)=Chr$(32)) Then
        plot(a,b,c)=0
      Else
        plot(a,b,c)=1
      EndIf
    Next b
  next c
next a
Close #2
GoSub Clear_box
Print @(270,20)"load ok"
  GoSub UpdateFontDetails
  GoTo Setupgrid
Char_up:
  If (charxy =1) Then
    'rem do nothing
  Else
    charxy=charxy-1
    GoSub update_char_number
    GoSub Displayfontchar
  EndIf
Return
```

Char_down:

```
If (charxy =fnumber) Then
    'rem do nothing
Else
    charxy=charxy+1
    GoSub update_char_number
    GoSub Displayfontchar
EndIf
Return
```

Displayfontchar:

```
Line (52,292)-(52+fwidth,292+fheight),0,BF
a=charxy
If (a=0) Then a=1
cursorx=7*min_x+2 : cursory=105
plotx=1 :ploty=1
For c=1 To fheight
    For b=1 To fwidth
        tempx=cursorx+(b-1)*screenstepx
        tempy=cursory+(c-1)*screenstepy
        temp2x=tempx+box_x
        temp2y=tempy+box_y
        If (plot(a,b,c)=0) Then
            Line (tempx,tempy)-(temp2x,temp2y),0,BF
        Else
            Line (tempx,tempy)-(temp2x,temp2y),1,BF
            Pixel(52+b,292+c)=1
        EndIf
    Next b
next c
GoSub update_char_number
Return
```

UpdateFontDetails:

```
Print @(100,15) "      " @(20,15)"Width ";fwidth
Print @(50,30) "      " @(20,30)"Height ";fheight
Print @(130,45) "      " @(20,45)"Characters in Set ";fnumber
Print @(80,60) "      "@(20,60)"Start Character ";fstart
Return
```

Instructions:

```
Print @(10,110)"Move   --- Cursor keys"
Print @(10,125)"Set     ____ Left Alt."
Print @(10,140)"Erase   --- Space"
Print @(10,155)"Load     --- l or L"
Print @(10,170)"Save     --- s or S"
Print @(10,185)"Char Up   ____ +"
Print @(10,200)"Char Down --- -"
Print @(10,215)"Add Char ____ Insert"
Print @(10,230)"Copy Char ____ c or C"
Print @(10,245)"Flip Vert. ____ v or V"
```

```
Print @(10,260)"Flip Horiz.____ h or H"
Print @(10,275)"Exit      ____ e or E"
Line (4,290)- (6*min_x-2,max_y-6),1,B
Print @(10,300)"Pixel" @(10,320)"Size"
Line (50,291)- (6*min_x-4,max_y-70),0,BF
Line (50,290)- (50+fwidth+4,290+fheight+4),1,B
Print @(30,Max_y-65)"Code Written by"
Print @(15,Max_y-45)"Dennis Wyatt ";
Font Load "copyr.fnt" As #6
Font #6
Print " ";
Font #1
Font unload #6
Print " 2011"
Print @(15,Max_y-25)"dpwyatt@iinet.net.au"
Return
```

My_pause: 'pause routine for array erasure completion

```
Timer=0
Do While timertemp<3000
timertemp=Timer
Loop
Return
```

No_sd_card:

```
Print @(270,40)"ERROR -No SD Card present"
Print @(270,55)"Please insert card and re-enter"
Print @(270,70)"Filename at the prompt above"
Pause 2000
If (typefile$="save") Then
GoTo Save_font
Else
GoTo Load_font
EndIf
```

Card_protected:

```
Print @(270,40)"Card is Write Protected"
Print @(270,55)"Please write enable and re-enter"
Print @(270,70)"Filename at the prompt above"
If (typefile$="save") Then
GoTo Save_font
Else
GoTo Load_font
EndIf
```

no_space_left:

```
Print @(270,40)"not enough space"
Print @(270,55)"Please rectify and re-enter"
Print @(270,70)"Filename at the prompt above"
If (typefile$="save") Then
GoTo Save_font
Else
GoTo Load_font
```



```
EndIf

All_root_gone:
  Print @(270,40)"All root dir. taken"
  If (typefile$="save") Then
    GoTo Save_font
  Else
    GoTo Load_font
  EndIf

Invalid_filename:
  Print @(270,40)"Invalid Filename"
  Print @(270,55)"Please rectify and re-enter"
  Print @(270,70)"Filename at the prompt above"
  If (typefile$="save") Then
    GoTo Save_font
  Else
    GoTo Load_font
  EndIf

Cannot_find_file:
  Print @(270,40)"Cannot find File"
  Print @(270,55)"Please re-enter"
  Print @(270,70)"Filename at the prompt above"
  If (typefile$="save") Then
    GoTo Save_font
  Else
    GoTo Load_font
  EndIf

Insert_char:      ' insert character function
  addchar=0
  savename$="temp.fnt"
  GoSub Insert_save_font
  fname$="temp.fnt"
  GoSub Insert_Load_Font
  addchar=1
  Kill "temp.fnt"
  Return

Insert_Char_now:
  temp$=""
  For column=1 To fwidth
    temp$=temp$+" "
  Next column
  For row = 1 To fheight
    Print #1,temp$
  Next row
  Return

Clear_box:
  Line (270,20)-(19*min_x,80),0,bf
```

Return

```
Copy_char:      ' Copy function
  GoSub Clear_box
```

```
Copy_input1_error:
  Print @(270,20);
  Input "Copy which Character # ";copychar
  If copychar>fnumber Then
    Print @(270,35)"Not that many Char."
    GoTo Copy_input1_error
  EndIf
```

```
Copy_input2_error:
  Print @(270,35);
  Input "To which Character # ";tochar
  If tochar>fnumber Then
    Print @(270,50) "Not that many Char."
    GoTo Copy_input2_error
  EndIf
  For copy1= 1 To fwidth
    For copy2= 1 To fheight
      plot(tochar,copy1,copy2)=plot(copychar,copy1,copy2)
    Next copy2
  Next copy1
  Print @(270,50)"
  Print @(270,50)"done"
  Return
```

```
Vertical_flip:      ' flip vertical function
  For flipV = 1 To Int(fheight/2)
    For flipV1 = 1 To fwidth
      flipvtemp=plot(charxy,flipV1,flipV)
      plot(charxy,flipV1,flipV)=plot(charxy,flipV1,fheight-flipV+1)
      plot(charxy,flipV1,fheight-flipV+1)=flipvtemp
    Next flipV1
  Next flipV
  GoSub Clear_box
  Print @(270,20)"Vertical Flip Done"
  GoSub Displayfontchar
  Return
```

```
horizontal_flip:      ' flip horizontal function
  For flipH = 1 To Int(fwidth/2)
    For flipH1 = 1 To fheight
      fliphtemp=plot(charxy,flipH,flipH1)
      plot(charxy,flipH,flipH1)=plot(charxy,fwidth-flipH+1,flipH1)
      plot(charxy,fwidth-flipH+1,flipH1)=fliphtemp
    Next flipH1
  Next flipH
  GoSub Clear_box
```

```
Print @(270,20)"Horizontal Flip Done"  
GoSub Displayfontchar  
Return
```

From:

<https://fruitoftheshed.com/wiki/> - **FotS**

Permanent link:

https://fruitoftheshed.com/wiki/doku.php?id=mmbasic_original:maxifont_bas

Last update: **2024/01/19 09:39**

