

**BMPSHOW.BAS**[bmpshow\\_bitmaps.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.*

```
10 ' BMPSHOW: displays a 1-bit BMP image
20 ' from Bruce Mitchell <brucepython@gmail.com>
30 '
40 ' BMP is a format with many variations and vagaries.
50 ' Don't be surprised if some images appear inverted, negative or
60 ' otherwise weird.
70 ' See en.wikipedia.org/wiki/BMP_file_format for basic info and some
80 ' useful references.
90 ' Max image size displayed is 480 wide x 432 high.
100 'Larger images might load OK but only a corner will be shown.
110 'It's slow!
120 '
130 'Possible enhancements might include choices to display the image as:
140 ' * inverted
150 ' * negative
160 ' * flipped horizontally
170 ' * located at x,y rather than top left.
180 '
190 INPUT "BMP file to load: "; f$
200 IF UCASE$(RIGHT$(f$,4))<>".BMP" THEN f$=f$+".BMP"
210 OPEN f$ FOR input AS #1
220 '
230 boundary=4 : ' BMP line data comes in 4 byte chunks.
240 FOR ptr=0 TO 9:a$=INPUT$(1,#1):NEXT: ' Skip the signature field.
250 '
260 'Get data offset.
270 GOSUB 770: offset = byte-1
280 FOR ptr=ptr TO ptr+5:a$=INPUT$(1,#1):NEXT
290 '
300 GOSUB 770:width=byte
310 FOR ptr=ptr TO ptr+1:a$=INPUT$(1,#1):NEXT
320 GOSUB 770:height=byte
330 ptr=ptr+8
340 '
350 ' Get colour depth in bits from &1C.
360 FOR ptr=ptr TO ptr+3:a$=INPUT$(1,#1)
370 NEXT
380 GOSUB 770: depth=byte
390 '
400 IF depth<>1 THEN PRINT "Only 1-bit BMP images are supported.":END
410 '
420 REM PRINT "Height=";height:' Remove REM if you need to check the image
```

```
size.
430 REM PRINT "Width =";width:' See above.
440 '
450 CLOSE #1: OPEN f$ FOR input AS #1:' Reset to start of file.
460 FOR ptr=0 TO offset:a$=INPUT$(1,#1):NEXT:' ptr points to start of
bitmap.
470 '
480 'Copy the bitmap to the screen.
490 CLS
500 FOR y=height-1 TO 0 STEP -1
510 'Clear the flag that signals the final byte(s) in the row.
520   endbyte=false
530 'Plot the row.
540   FOR x=0 TO width-8 STEP 8
550     GOSUB 820
560   NEXT x
570 ' Are we at a 4-byte boundary?
580   IF x < width THEN
590     endbyte = true
600     ' Padding to the nearest 4-byte boundary exists.
610     ' Count the pixels to be drawn and draw them, then skip the padding.
620     n=width-x : ' There are n pixels left in the row.
630     IF n>=8 THEN GOSUB 820 : x=x+8 : GOTO 630:'Display eight bits.
640     byte=ASC(INPUT$(1,#1))
650   ENDIF
660   DO WHILE (x MOD boundary)>0
670     a$=INPUT$(1,#1):x=x+1
680   LOOP
690 NEXT y
700 '
710 'Indicate job is done and wait for a keypress.
720 SOUND 440,100
730 IF INKEY$="" THEN GOTO 730
740 END
750 '
760 '===== SUBROUTINES =====
770 ' Get next two bytes from ptr, return as 'byte', then inc file ptr.
780 lsb=ASC(INPUT$(1,#1)):msb=ASC(INPUT$(1,#1))
790 byte=(msb * 256) + lsb
800 RETURN
810 '
820 ' Display next 8 pixels in the row.
830 byte=ASC(INPUT$(1,#1))
840 IF endbyte=true THEN GOTO 870: 'Short cuts can spill past the edge.
850 IF byte=&hFF THEN RETURN:' Nothing to draw.
860 IF byte=0 THEN LINE(x,y)-(x+8,y),1:RETURN:'Drawing a line is faster.
870 FOR bit=7 TO 0 STEP -1
880   PIXEL(x+7-bit,y)=NOT(byte AND 2^bit)
890 NEXT bit
900 RETURN
```

From:

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

Permanent link:

[https://fruitoftheshed.com/wiki/doku.php?id=mmbasic\\_original:bmpshow\\_bas](https://fruitoftheshed.com/wiki/doku.php?id=mmbasic_original:bmpshow_bas)

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

