

Embedded Fonts for the Micromite

MMBasic for the Micromite allows a BASIC program to define a font in the program. This allows the programmer to use unusual fonts to suit a particular need not covered by the standard built in fonts. Embedded fonts work exactly same as the built in fonts (ie, selected using the FONT command or specified in the TEXT command).

Defining a Font

The format of an embedded font is:

```
DefineFont #Nbr
    hex [[ hex[...]]
    hex [[ hex[...]]
END DefineFont
```

An embedded font must start with the keyword "DefineFont" followed by the font number (which may be preceded by an optional # character). Any font number in the range of 1 to 16 can be specified including a font number used by one of the built in fonts. In that case the embedded font will replace the built in font.

The body of the font is a sequence of 8-digit hex words with each word separated by one or more spaces or a new line. The first 8-digit hex word is the characteristics of the font (number of characters, size, etc) while the following words define the bitmap of the font. The format must follow the UTFT standard (see <http://www.rinkydinkelectronics.com> for the details).

The font definition is terminated by an "End DefineFont " keyword.

When a BASIC program is saved to program memory MMBasic will search through it looking for any embedded fonts. The data specified will be extracted and programmed into flash memory and the font table updated to include the new fonts.

During execution MMBasic will skip over the DefineFont command and the data specified. This means that embedded fonts can be placed anywhere in the program however at the end is the normal location.

Fonts in the LIBRARY area

Embedded fonts can also be saved to the library area. When this is done the font is, for all intent and purpose, permanently added to the MMBasic language. When MMBasic scans for embedded fonts it will search the library area first so any fonts defined there will be added to the table before fonts in the main program.

When a font is saved to the library area MMBasic will strip out the hex codes in the font definition (because they are no longer needed). This will save a considerable amount of memory and is recommended if you are using some of the larger fonts.

See the LIBRARY command for details on how to use the library.

Obtaining Fonts

The standard distribution of MMBasic for the Micromite includes over a dozen fonts in a format ready for use in an MMBasic program. These fonts were converted from UTFT fonts kindly provided by Henning Karlsen at www.rinkydinkelectronics.com.

All the fonts are the same as the fonts listed at http://www.rinkydinkelectronics.com/r_fonts.php and it is worth visiting that web page to see examples of the character sets, etc.

Also included in the standard distribution of MMBasic is the program UTFTConv which can be used to convert UTFT fonts on this web page to the format required by MMBasic. The program is provided in two forms, a Windows EXE file (provided by Robert Severson) and an MMBasic (.BAS) program which will run on the Maximite, Micromite Plus or the DOS version of MMBasic.

Editing Fonts

Jim Hiley (TassyJim on The Back Shed forum) has created an editor for font files used by the Micromite. Using this you can edit a character, add or delete characters, scale the font and scan an image to create a new character.

This program FontTweak.exe (for Windows) is included in the standard distribution of MMBasic in the folder "Embedded Fonts". It does not need installation; you just run the .exe file.