Metafont 2.7
for the Commodore Amiga

ported by: Stefan Becker

October 28, 1990

1 What is Metafont?

METAFONT is a program which produces raster fonts for the TeX system. The user creates a program for each character of a given font, using the special METAFONT language. METAFONT runs through every program and creates a raster bitmap for the character in the resolution of the given output device. At the same time it calculates the font metrics which TeX needs.

This program is a port of METAFONT 2.7 for the Commodore Amiga Computer. It is full compatible to implementations on other systems, because it has passed the trap test. Only a graphic output for character screening at runtime was added. This documentation describes **only** the operation of METAFONT and its utilities on the Amiga. Further information, especially about the METAFONT language, is available in the literature, e.g. [1] & [2].

2 The operation of Metafont

METAFONT can only be called from the CLI, as the following example shows:

```
> virmf
This is METAFONT, C Version 2.7
** _
```

METAFONT shows its banner line and awaits the name of a METAFONT source file. Such files have a name ending with *.mf.* If your system is equiped with too few RAM, it can happen that METAFONT stops with the error message `Not enough memory (x)!` (x is a number). To circumvent this situation you can reduce the size of the internal memory arrays of METAFONT, as described below.

METAFONT accepts a number of parameters at call time. This is especially helpful for the writing of script files. The following parameters are allowed:

name Name of a METAFONT source file.

&name Name of a *.base* file. This file contains frequently used procedures in a pre-compiled form (see below).

"\ . . ." METAFONT language commands, separated by ';', e.g.:

batchmode Suppress the METAFONT terminal output.

input name Process the METAFONT source file *name.mf*.

mag:=size Size of the font to be produced.

mode:=name Produce a font for the output device *name*.

screenchars Show completed characters in the graphic output window.

screenstrokes Show all creation stages of a character in the graphic output window.

METAFONT can produce fonts for any output device. The user must create a so called *mode definition* for each device. The name of such a definition must be given behind the *mode* command. The file *mydevices.mf* contains some examples. Detailed information about *mode definitions* and font sizes can be obtained from [1].

Two special output devices exist, which can be used to produce big enlarged fonts. These modes have the names *proof* and *smoke*. If one of this modes is used, the command *screenchars* will be executed. It opens a graphic output window with the size of $500 * 400$ (or $250 * 200$, see below), which will be used to display the characters. If the window couldn't be opened, METAFONT does *not* display an error message, but continues its work.

METAFONT displays the number of the currently processed character at runtime. If an error occurs, METAFONT stops and displays an extensive but cryptic error message and a prompt. This is the METAFONT error mode. In this interactive mode you can try to correct the error. A command list is available with '?'. With 'X' you can leave the program and METAFONT continues to work with *RETURN*. At runtime you can reach this error mode with *CTRL-C*.

If you have typed in the wrong file name, then METAFONT prompts you for the right one. In this mode you can leave the program with *CTRL-*\.

And now an example run of METAFONT. We want to produce the font *cmb10* in the size *magstep 4* for the screen previewer. The command line and the (shorten) example run looks as follows:

```
> virmf "\mode:=previewer; mag:=magstep(4); input cmb10"
This is METAFONT, C Version 2.7
```

```
(mf:mfinputs/cmb10.mf (mf:mfinputs/roman.mf
(mf:mfinputs/romanu.mf [65] [66] [67] [68] [69] [70]
[......]
[34] [45] [92] [123] [124]) ) )
Font metrics written on cmb10.tfm.
Output written on cmb10.207gf (128 characters, 9620 bytes).
Transcript written on cmb10.log.
> _
```

3 Metafont Utilities

This section describes the additional METAFONT utilities.

3.1 gftopk

SYNTAX: `gftopk` *GF file name* *PK file name*

This program converts fonts from METAFONT common *GF* format to the compressed *PK* format. It is used by the most printer drivers.

3.2 gftodvi

SYNTAX: `gftodvi`

This program converts fonts, which were produced with the *proof* or *smoke* mode, into a printable *.dvi* file. Each character of the font fills one page. The program needs two special fonts *black* and *gray*, which contain mainly gray rasters. To produce these fonts, consult the appendix in [1].

3.3 gftype

SYNTAX: `gftype` [–m] [–i] *GF file name*

This program extracts the contents of a *GF* font file and displays them in a readable form. With the two options –m and –i you can choose, how much information will be displayed.

3.4 mft

SYNTAX: `mft` *MF file name* [–s] [*style file*]

If you want to print out your METAFONT source files in a more readable form, then you can use this program to convert *.mf* files into *.tex* files. These files can then be processed with TEX. The style file contains the parameters for the conversion. This packet contains the style file *plain.mft* and the macro packet *mftmac.tex* for the post-processing with TEX.

4 The configuration of Metafont

The paths, which METAFONT and its utilities use for the search of files, can be set with environment variables. These variables can be set with the AmigaDOS command `setenv`. The different paths are seperated with ','. The following variables are used (the default path is enclosed in parenthesis, '.' stands for the current directory):

mfinputs (.,mf:inputs) Path for the *.mf* files.

mfbases (.,mf:bases) Path for the *.base* file.

mfpool (.,mf:bases) Used by `inimf` for the *mf.pool* file (see below).

texfonts (.,tex:fonts) Path for the *.tfm* files.

dviconfig (.,mf:config) Path for the *MF.mem* file (see below).

The size of the internal memory arrays of METAFONT can be controlled with the configuration file *MF.mem*. This text file consists of several lines with one number in each line. **BEWARE:** The order and number of lines is important! The example file supplied with this packet has a comment behind each line, which describes the function of the number. The following explains only the important numbers, because all others have only an internal meaning:

memmax Size of the METAFONT main memory array in memory words. One memory word is 4 Bytes large.

memtop Highest usable memory word. The following **must** be true: *memtop* ≤ *memmax*

scalefactor Size of the METAFONT graphic output window:

- **1** Big window, characters are displayed in normal size.
- **2** Small window, characters are displayed in half of the normal size.

If you have changed the configuration, you must create a new *.base* file, which corresponds to the new values. This file is created by a special METAFONT version, called `inimf`. This version "compiles" frequently needed parts, e.g. *plain.mf*, *cmbase.mf*, error messages and your own *mode definitions*, into the *.base* file, which the normal METAFONT version `virmf` only needs to load into memory. The error messages are contained in the file *mf.pool*. A sample run of `inimf` might look as follows:

```
> inimf
This is METAFONT, C Version 2.7 (INIMF)
**plain
(plain.mf
Preloading the plain base, version 2.0: preliminaries,
 basic constants and mathematical macros,
 macros for converting from device-independent units to pixels,
 macros and tables for various modes of operation,
 macros for drawing and filling,
 macros for proof labels and rules,
 macros for character and font administration,
and a few last-minute items.)
*\input mydevices
(mf:inputs/mydevices.mf)
*\input cmbase
(mf:inputs/cmbase.mf)
*\dump
Beginning to dump on file plain.base
 (base=plain 90.9.24)
1622 strings of total length 24962
15588 memory locations dumped; current usage is 3789&11746
```

```
743 symbolic tokens
> _
```

The file *plain.base*, created by this example run, must then be moved into a directory, which is contained in the environment variable *mfbases*. This examples has created a file, which contains all procedures needed to create the *Computer Modern Roman* fonts, the standard fonts of TeX. Further information may be obtained from [1] & [2].

5 Appendix

BIBLIOGRAPHY:

1 *The Metafont Book* by Donald E. Knuth, the inventor of METAFONT.

2 *Computer Modern Typefaces* by Donald E. Knuth, a detailed description of the CMR fonts.

I hope my english was not too bad…