

Metafont 2.7
für den Commodore Amiga

Portiert von: Stefan Becker

28. Oktober 1990

1 Was ist Metafont?

METAFONT ist ein Programm, das mit Hilfe einer ausgefeilten, PASCAL-ähnlichen Programmiersprache die Raster-Zeichensätze für das T_EX-System erstellt. Für jeden Buchstaben eines Zeichensatz erstellt der Benutzer ein Programm in dieser Sprache. METAFONT arbeitet jedes dieser Programme ab und erstellt dabei ein gerastertes Abbild der Buchstaben für die Auflösung des eingestellten Ausgabegerätes. Gleichzeitig ermittelt es die für T_EX benötigten Zeichensatzmetriken.

Das vorliegende Programm ist eine Portierung von METAFONT 2.7 für den Commodore Amiga Computer. Es ist voll kompatibel zu Implementierungen auf anderen Rechnern, da es den Trap-Test bestanden hat. Zusätzlich wurde noch die Möglichkeit der Grafikausgabe während des Programmablaufs eingebaut. Diese Anleitung beschreibt **nur** die Bedienung von METAFONT und den dazugehörigen Utilities auf dem Amiga. Alle weiteren Informationen, insbesondere über die METAFONT-Sprache, entnehmen Sie bitte der entsprechenden Literatur, z.B. aus [1] & [2].

2 Die Bedienung von Metafont

METAFONT läßt sich nur über das CLI aufrufen. Ein entsprechender Aufruf sieht folgendermaßen aus:

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

METAFONT meldet sich mit seiner Titelzeile und erwartet nun die Eingabe des Namens der METAFONT-Quelltextdatei, die bearbeitet werden soll. Diese Dateien sind an der Endung *.mf* zu erkennen. Falls Ihr System zu wenig Speicher hat, dann kann es passieren, daß METAFONT mit der Fehlermeldung **Not enough memory (x)!** abbricht (x ist eine Zahl). Um dies zu umgehen, können Sie die internen Speicherfelder von METAFONT verkleinern, wie es weiter unten beschrieben wird.

Man kann METAFONT eine Reihe von Parametern beim Aufruf mitgeben, was z.B. für die Erstellung einer Batchdatei wichtig ist. Es sind folgende Parameter zulässig:

name Name einer METAFONT-Quelltextdatei.

&name Name einer *.base* Datei, die oft benötigte Prozeduren in vorbearbeiteter Form enthält (s.u.).

“\...” Kommandos aus der METAFONT-Sprache, getrennt durch ‘;’, z.B.:

batchmode Ausgaben von METAFONT auf dem Bildschirm sollen unterbunden werden.

input name Die METAFONT-Quelltextdatei *name.mf* soll bearbeitet werden.

mag:=größe Gibt die gewünschte Größe des Zeichensatzes an.

mode:=name Stellt das Ausgabegerät *name* ein.

screenchars Fertige Buchstaben sollen in der Grafikausgabe dargestellt werden.

screenstrokes Es sollen alle Zwischenstufen des Buchstabens in der Grafikausgabe dargestellt werden.

METAFONT kann für beliebige Ausgabegeräte Zeichensätze erzeugen. Für jedes Gerät muß vom Benutzer eine sogenannte *Mode Definition* erstellt werden. Den Namen dieser Definition gibt man dann hinter dem *mode* Befehl an. Die mitgelieferte Datei *mydevices.mf* enthält ein paar Beispiele. Näheres über *Mode Definitions* und über die Größenangabe bei Zeichensätzen entnehmen Sie bitte aus [1].

Es gibt zwei fiktive Ausgabegeräte, mit denen man sich die Zeichensätze in starker Vergrößerung erzeugen lassen kann. Diese haben die Namen *proof* und *smoke*. Wird einer dieser beiden Modi angewählt, so wird automatisch der Befehl *screenchars* ausgeführt. Es öffnet sich die Grafikausgabe, ein Fenster der Größe 500*400 (oder 250*200, s.u.), in dem nun alle Buchstaben bildschirmgroß ausgegeben werden. Falls das Fenster nicht geöffnet werden kann, wird *keine* Fehlermeldung ausgegeben, aber METAFONT setzt seine Arbeit fort.

Wenn METAFONT einen Zeichensatz erstellt, dann zeigt es die Nummer des gerade erzeugten Buchstabens an. Sollte ein Fehler auftreten, so meldet sich METAFONT mit einer umfangreichen, teilweise leider auch sehr kryptischen Fehlermeldung und einem Prompt. METAFONT befindet sich nun im Fehlermodus und Sie haben die Möglichkeit den Fehler interaktiv zu beheben. Wenn Sie ein '?' eingeben, erhalten Sie eine Übersicht über alle zugänglichen Befehle. Mit 'X' koennen Sie das Programm beenden und mit *RETURN* setzt METAFONT seine Arbeit fort. Sie können während der Programmlaufs jederzeit durch Drücken von *CTRL-C* in den Fehlermodus gelangen.

Sollten Sie sich mal in einem Dateinamen vertippt haben, so fordert METAFONT einen neuen Namen an. Wollen Sie an dieser Stelle das Programm abbrechen, so müssen Sie *CTRL-* drücken.

Zum Schluß noch ein Beispiellauf. Es soll der Zeichensatz *cmb10* in der Größenstufe *magstep 4* für den Bildschirmtreiber (Modename *previewer*) erzeugt werden. Die entsprechende Kommandozeile und der dazugehörige METAFONT-Lauf sieht dann (gekürzt) folgendermaßen aus:

```
> 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

Zu METAFONT gehören ein paar Zusatzprogramme, die im Folgenden beschrieben werden.

3.1 gftopk

AUFRUF: `gftopk GF Dateiname PK Dateiname`

Dieses Programm dient dazu, die Zeichensätze aus METAFONT's allgemeinem *GF* Format in das komprimierte *PK* Format umzuwandeln. Dieses Format wird von den meisten Druckertreibern benutzt.

3.2 gftodvi

AUFRUF: `gftodvi`

Mit diesem Programm können Zeichensätze die mit dem *proof* oder *smoke* Modus erstellt worden sind, in eine ausdrückbare *.dvi* Datei umgewandelt werden. Jedes einzelne Zeichen ist in dieser Datei auf einer Seite blattgroß vorhanden, was zur Überprüfung der einzelnen Zeichen verwendet werden kann. Das Programm und der Ausdruck benötigt die speziellen Zeichensätze *black* und *gray*, die im wesentlichen verschiedene Grauraster enthalten. Wie man diese erstellt, steht im Anhang von [1].

3.3 gftype

AUFRUF: `gftype [-m] [-i] GF Dateiname`

Dieses Programm gibt den Inhalt einer Zeichensatzdatei, die im *GF* Format vorliegt, als Klartext aus. Mit den beiden Optionen `-m` und `-i` können Sie auswählen, wieviel Information aus der Datei angezeigt wird.

3.4 mft

AUFRUF: `mft MF Dateiname [-s] [Stil-Datei]`

Falls Sie ihre METAFONT-Quelltextdateien in leserlicher Form ausdrucken wollen, dann können Sie mit diesem Programm die *.mf* Dateien in *.tex* Dateien umwandeln. Diese Dateien können dann mit T_EX weiterverarbeitet werden. Die Stil-Datei gibt dabei die Parameter für die Umwandlung an. In diesem Paket ist die Stil-Datei *plain.mft* und für die Bearbeitung mit T_EX die Makrodatei *mftmac.tex* enthalten.

4 Die Konfiguration von Metafont

Die Pfade, in denen METAFONT und seine Utilities nach Dateien suchen, lassen sich durch Umgebungsvariablen einstellen. Diese werden durch den AmigaDOS Befehl **setenv** gesetzt. Dabei werden die einzelnen Pfade durch ‘,’ getrennt. Es werden folgende Variablen benutzt (in Klammern stehen die Voreinstellungen, ‘.’ bezeichnet das aktuelle Verzeichnis):

mfinputs (**.,mf:inputs**) Hier sucht METAFONT die *.mf* Dateien.

mfbases (**.,mf:bases**) Hier sucht METAFONT die *.base* Datei (s.u.).

mfpool (**.,mf:bases**) Hier sucht in *inimf* die Datei *mf.pool* (s.u.).

texfonts (**.,tex:fonts**) Hier werden die *.tfm* Dateien abgelegt.

dviconfig (**.,mf:config**) Hier sucht METAFONT die Datei *MF.mem* (s.u.).

Die Größen der internen Speicherfelder von METAFONT lassen sich mit Hilfe der Konfigurationsdatei *MF.mem* einstellen. Diese Textdatei enthält mehrere Zeilen, in denen jeweils eine Zahl steht. **ACHTUNG:** Die Reihenfolge und Anzahl der Zahlen ist wichtig! In der mitgelieferten Beispieldatei steht am Ende jeder Zeile ein Kommentar, der die jeweilige Zahl beschreibt. Es werden hier nur die wichtigsten Werte erklärt, da die meisten anderen nur eine interne Bedeutung haben:

memmax Größe des METAFONT-Hauptspeichers in Speicherworten. Ein Speicherwort ist 4 Bytes groß.

memtop Höchste nutzbare Speicherzelle. Dabei **muß** gelten: $memtop \leq memmax$

scalefactor Größe der METAFONT-Grafikausgabe:

- 1 Großes Fenster, Buchstaben in Normalgröße
- 2 Kleines Fenster, Buchstaben um den Faktor 2 verkleinert

Falls man die Konfiguration geändert hat, so muß man eine neue *.base* Datei erstellen, die diesen neuen Werten entspricht. Diese Datei wird mit einer speziellen METAFONT-Version, genannt *inimf*, erzeugt. Diese Version „compiliert“ häufig benötigte Teile, wie z.B. *plain.mf*, *cmbase.mf*, die Fehlermeldungen und die eigenen *Mode Definitions*, in die *.base* Datei, die von der normalen METAFONT-Version *virmf* nur noch in den Speicher geladen wird. Die Fehlermeldungen stehen in der Datei *mf.pool*. Es folgt der Mitschnitt einer Beispielsitzung mit *inimf*:

```
> 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
> _
```

Die so entstandene Datei *plain.base* muß dann in dem entsprechenden Verzeichnis abgelegt werden, auf das die Umgebungsvariable *mfbases* zeigt. Die in dem Beispiel erstellte Datei enthält alle Prozeduren, um die *Computer Modern Roman* Zeichensätze zu erstellen, die Standardzeichensätze für T_EX. Weitere Einzelheiten kann man [1] & [2] entnehmen.

5 Diverses

LITERATUR:

- 1 *The Metafont Book* von Donald E. Knuth, dem Erfinder von METAFONT.

2 *Computer Modern Typefaces* von Donald E. Knuth, eine genaue Beschreibung der CMR Zeichensätze.