NAME

groff_mm − groff mm macros

SYNOPSIS

groff −mm [*options*...] [*files*...]

DESCRIPTION

The groff mm macros are intended to be compatible with the DWB mm macros with the following limitations:

- no Bell Labs localisms implemented.
- the macros OK and PM are not implemented.
- groff mm does not support cut marks

mm is intended to be international. Therefore it is possible to write short national macrofiles which change all english text to the preferred language. Use **mmse** as an example.

A file called **locale** or *lang***_locale** is read after the initiation of the global variables. It is therefore possible to localize the macros with companyname and so on.

In this manual square brackets is used to show optional arguments.

Number registers and strings

Many macros can be controlled by number registers and strings. A number register is assigned with the **nr** command:
.nr *XXX [+-]n [i]*
XXX is the name of the register, **n** is the value to be assigned, and **i** is increment value for auto-increment. **n** can have a plus or minus sign as prefix if an increment or decrement of the current value is wanted. (Auto-increment or decrement occurs if the number register is used with a plus or minus sign, **\n+[XXX]** or **\n-[XXX]**.)

Strings is defined with **ds**.
.ds *YYY string*
The string is assigned everything to the end of the line, even blanks. Initial blanks in *string* should be prefixed with a double-quote. (Strings are used in the text as ***[YYY]**.)

Special formatting of number registers

A number register is printed with normal digits if no format has been given. Set the format with **af**:
.af *R c*
R is the name of the register, *c* is the format.

Form	**Sequence**
1	0, 1, 2, 3, ...
001	000, 001, 002, 003, ...
i	0, i, ii, iii, iv, ...
I	0, I, II, III, IV, ...
a	0, a, b, c, ..., z, aa, ab, ...
A	0, A, B, C, ..., Z, AA, AB, ...

Macros:

)E level text

Adds **text** (heading-text) to the table of contents with **level** either 0 or between 1-7. See also **.H**. This macro is used for customized table of contents.

1C [1] Begin one column processing. An **1** as argument disables the page-break. Use wide footnotes, small footnotes may be overprinted.

2C Begin two column processing. Splits the page in two columns. It is a special case of **MC**. See also **1C**.

AE Abstract end, see **AS**.

AF [name of firm]
Authors firm, should be called before **AU**, see also **COVER**.

AL [type [text-indent [1]]]
Start autoincrement list. Items are numbered beginning on one. The *type* argument controls the type of numbers.

Arg	Description
1	Arabic (the default)
A	Upper-case letters (A-Z)
a	Lower-case letters (a-z)
I	Upper-case roman
i	Lower-case roman

Text-indent sets the indent and overrides **Li**. A third argument will prohibit printing of a blank line before each item.

APP name text
Begin an appendix with name *name*. Automatic naming occurs if *name* is "". The appendixes starts with **A** if auto is used. An new page is ejected, and a header is also produced if the number variable **Aph** is non-zero. This is the default. The appendix always appear in the 'List of contents' with correct pagenumber. The name *APPENDIX* can be changed by setting the string **App** to the desired text. The string **Apptxt** contains the current appendix text.

APPSK name pages text
Same as **.APP**, but the pagenr is incremented with *pages*. This is used when diagrams or other non-formatted documents are included as appendixes.

AS [arg [indent]]
Abstract start. Indent is specified in 'ens', but scaling is allowed. Argument *arg* controls where the abstract is printed.

Arg	Placement
0	Abstract will be printed on page 1 and on the cover sheet if used in the released-paper style (**MT 4**), otherwise it will be printed on page 1 without a cover sheet.
1	Abstract will only be printed on the cover sheet (**MT 4** only).
2	Abstract will be printed only on the cover sheet (other than **MT 4** only). The cover sheet is printed without need for **CS**.

Abstract is not printed at all in external letters (**MT 5**). The *indent* controls the indentation of both margins, otherwise will normal text indent be used.

AST [title]
Abstract title. Default is **ABSTRACT**. Sets the text above the abstract text.

AT title1 [title2 ...]
Authors title. **AT** must appear just after each **AU**. The title will show up after the name in the signature block.

AU [name [initials [loc [dept [ext [room [arg [arg [arg]]]]]]]]]
Author information, specifies the author of the memo or paper, and will be printed on the cover sheet and on other similar places. **AU** must not appear before **TL**. The author information can contain initials, location, department, telephone extension, room number or name and up to three extra arguments.

AV [name [1]]
Approval signature, generates an approval line with place for signature and date. The string **APPROVED:** can be changed with variable **Letapp**, and the string **Date** in **Letdate**.

AVL [name]
Letter signature, generates a line with place for signature.

B [bold-text [prev-font-text [bold...]]]
Begin boldface. No limit on the number of arguments. All arguments will be concatenated to one word, the first, third and so on will be printed in boldface.

B1 Begin box (as the ms macro). Draws a box around the text. The text will be indented one character, and the right margin will be one character shorter.

B2 End box. Finish the box started by **B1**.

BE End bottom block, see **BS**.

BI [bold-text [italic-text [bold-text [...]]]]
Bold-italic. No limit on the number of arguments, see **B**.

BL [text-indent [1]]
Start bullet list, initialize a list with a bullet and a space in the beginning of each list item (see **LI**). *Text-indent* overrides the default indentation of the list items set by number register **Pi**. A third argument will prohibit printing of a blank line before each item.

BR [bold-text [roman-text [bold-text [...]]]]
Bold-roman. No limit on the number of arguments.

BS Bottom block start. Begins the definition of a text block which is printed at the bottom of each page. Block ends with **BE**.

BVL text-indent [mark-indent [1]]
Start of broken variable-item list. Broken variable-item list has no fixed mark, it assumes that every **LI** has a mark instead. The text will always begin at the next line after the mark. *Text-indent* sets the indent to the text, and *mark-indent* the distance from the current indent to the mark. A third argument will prohibit printing of a blank line before each item.

COVER [arg]
COVER begins a coversheet definition. It is important that **.COVER** appears before any normal text. **.COVER** uses *arg* to build the filename /usr/share/groff/1.18.1.4/tmac/mm/*arg*.cov. Therefore it is possible to create unlimited types of coversheets. *ms.cov* is supposed to look like the **ms** coversheet. **.COVER** requires a **.COVEND** at the end of the coverdefinition. Always use this order of the covermacros:
.COVER
.TL
.AF
.AU
.AT
.AS
.AE
.COVEND
However, only **.TL** and **.AU** are required.

COVEND
This finish the cover description and prints the cover-page. It is defined in the cover file.

DE Display end. Ends a block of text, display, that begins with **DS** or **DF**.

DF [format [fill [rindent]]]
Begin floating display (no nesting allowed). A floating display is saved in a queue and is printed in the order entered. *Format, fill* and *rindent* is the same as in **DS**. Floating displays are controlled by the two number registers **De** and **Df**.

De register

0 Nothing special, this is the default.

1 A page eject will occur after each printed display, giving only one display per page and no text following it.

Df register

0 Displays are printed at the end of each section (when section-page numbering is active) or at the end of the document.

1 A new display will be printed on the current page if there is enough space, otherwise it will be printed at the end of the document.

2 One display will be printed at the top of each page or column (in multi-column mode).

3 Print one display if there is enough space for it, otherwise it will be printed at the top of the next page or column.

4 Print as many displays that will fit in a new page or column. A page break will occur between each display if **De** is not zero.

5 Fill the current page with displays and the rest beginning at a new page or column. (This is the default.) A page break will occur between each display if **De** is not zero.

DL [text-indent [1 [1]]]

Dash list start. Begins a list where each item is printed after a dash. *Text-indent* changes the default indentation of the list items set by number register **Pi**. A second argument prevents the empty line between each list item to be printed. See **LI**. A third argument will prohibit printing of a blank line before each item.

DS [format [fill [rindent]]]

Static display start. Begins collection of text until **DE**. The text is printed together on the same page, unless it is longer than the height of the page. **DS** can be nested to a unlimited depth (reasonably :-).

format

"" No indentation.

none No indentation.

L No indentation.

I Indent text with the value of number register **Si**.

C Center each line

CB Center the whole display as a block.

R Right adjust the lines.

RB Right adjust the whole display as a block

L, I, C and CB can also be specified as 0, 1, 2 or 3 for compatibility reasons. (Don't use it. :-)

fill

"" Line-filling turned off.

none Line-filling turned off.

N Line-filling turned off.

F Line-filling turned on.

N and F can also be specified as 0 or 1. An empty line will normally be printed before and after the display. Setting number register **Ds** to 0 will prevent this. *Rindent* shortens the line length by that amount.

EC [title [override [flag [refname]]]]

Equation title. Sets a title for an equation. The *override* argument change the numbering.

flag

none *override* is a prefix to the number.
0 *override* is a prefix to the number.
1 *override* is a suffix to the number.
2 *override* replaces the number.

EC uses the number register **Ec** as counter. It is possible to use **.af** to change the format of the number. If number register **Of** is 1, then the format of title will use a dash instead of a dot after the number.

The string **Le** controls the title of the List of Equations, default is *LIST OF EQUATIONS*. The List of Equations will only be printed if number register **Le** is 1, default 0. The string **Liec** contains the word *Equation*, wich is printed before the number. If *refname* is used, then the equation number is saved with **.SETR**, and can be retrieved with **.GETST** *refname*.

Special handling of the title will occur if **EC** is used inside **DS**/**DE**, it will not be affected by the format of **DS**.

EF [arg]

Even-page footer, printed just above the normal page footer on even pages, see **PF**.

EH [arg]

Even-page header, printed just below the normal page header on even pages, see **PH**.

EN Equation end, see **EQ**.

EOP End of page user-defined macro. This macro will be called instead of the normal printing of the footer. The macro will be executed in a separate environment, without any trap active. See **TP**.

Strings available to EOP

EOPf Argument from **PF**.
EOPef Argument from **EF**.
EOPof Argument from **OF**.

EPIC [-L] width height [name]

EPIC draws a box with the given *width* and *height*, it will also print the text *name* or a default string if *name* is not specified.. This is used to include external pictures, just give the size of the picture. **-L** will leftadjust the picture, the default is to center adjust. See **PIC**

EQ [label]

Equation start. **EQ**/**EN** are the delimiters for equations written for **eqn**. **EQ**/**EN** must be inside a **DS**/**DE**-pair, except when **EQ** is only used to set options in **eqn**. The *label* will appear at the right margin of the equation, unless number register **Eq** is 1. Then the label will appear at the left margin.

EX [title [override [flag [refname]]]]

Exhibit title, arguments are the same as for **EC**. **EX** uses the number register **Ex** as counter. The string **Lx** controls the title of the List of Exhibits, default is *LIST OF EXHIBITS*. The List of Exhibits will only be printed if number register **Lx** is 1, default 1. The string **Liex** contains the word *Exhibit*, which is printed before the number. If *refname* is used, then the exhibit number is saved with **.SETR**, and can be retrieved with **.GETST** *refname*.

Special handling of the title will occur if **EX** is used inside **DS**/**DE**, it will not be affected by the format of **DS**.

FC [closing]

Prints *Yours very truly*, as a formal closing of a letter or memorandum. The argument replaces the defualt string. The default is stored in string variable **Letfc**.

FD [arg [1]]

Footnote default format. Controls the hyphenation (hyphen), right margin justification (adjust), indentation of footnote text (indent). It can also change the label justification (ljust).

arg	hyphen	adjust	indent	ljust
0	no	yes	yes	left
1	yes	yes	yes	left
2	no	no	yes	left
3	yes	no	yes	left
4	no	yes	no	left
5	yes	yes	no	left
6	no	no	no	left
7	yes	no	no	left
8	no	yes	yes	right
9	yes	yes	yes	right
10	no	no	yes	right
11	yes	no	yes	right

Argument greater than or equal to 11 is considered as arg 0. Default for mm is 10.

FE Footnote end.

FG [title [override [flag [refname]]]]

Figure title, arguments are the same as for **EC**. **FG** uses the number register **Fg** as counter. The string **Lf** controls the title of the List of Figures, default is *LIST OF FIGURES*. The List of Figures will only be printed if number register **Lf** is 1, default 1. The string **Lifg** contains the word *Figure*, wich is printed before the number. If *refname* is used, then the figure number is saved with **.SETR**, and can be retrieved with **.GETST** *refname*.
Special handling of the title will occur if **FG** is used inside **DS**/**DE**, it will not be affected by the format of **DS**.

FS [label]

Footnote start. The footnote is ended by **FE**. Footnotes is normally automatically numbered, the number is available in string **F**. Just add ***F** in the text. By adding *label*, it is possible to have other number or names on the footnotes. Footnotes in displays is now possible. An empty line separates footnotes, the height of the line is controlled by number register **Fs**, default value is 1.

GETHN refname [varname]

Includes the headernumber where the corresponding **SETR** *refname* was placed. Will be X.X.X. in pass 1. See **INITR**. If *varname* is used, **GETHN** sets the stringvariable *varname* to the headernumber.

GETPN refname [varname]

Includes the pagenumber where the corresponding **SETR** *refname* was placed. Will be 9999 in pass 1. See **INITR**. If *varname* is used, **GETPN** sets the stringvariable *varname* to the pagenumber.

GETR refname

Combines **GETHN** and **GETPN** with the text 'chapter' and ', page'. The string *Qrf* contains the text for reference:

```
.ds Qrf See chapter \\*[Qrfh], page \\*[Qrfp].
```

Qrf may be changed to support other languages. Strings *Qrfh* and *Qrfp* are set by **GETR** and contains the page and headernumber.

GETST refname [varname]

Includes the string saved with the second argument to **.SETR**. Will be dummystring in pass 1. If varname is used, **GETST** sets the stringvariable *varname* to the saved string. See **INITR**.

H level [heading-text [heading-suffix]]

Numbered section heading. Section headers can have a level between 1 and 7, level 1 is the top level. The text is given in *heading-text*, and must be surrounded by double quotes if it contains spaces. **Heading-suffix** is added to the header in the text but not in the table of contents. This is normally used for footnote marks and similar things. Don't use ***F** in *heading-suffix*, it won't

work. A manual label must be used, see **FS**.

An eventual paragraph, **P**, directly after **H** will be ignored, **H** is taking care of spacing and indentation.

Page ejection before heading
Number register **Ej** controls page ejection before the heading. Normally, a level one heading gets two blank lines before it, higher levels gets only one. A new page is ejected before each first-level heading if number register **Ej** is 1. All levels below or equal the value of **Ej** gets a new page. Default value for **Ej** is 0.

Heading break level
A line break occurs after the heading if the heading level is less or equal to number register **Hb**. Default value 2.

Heading space level
A blank line is inserted after the heading if the heading level is less or equal to number register **Hs**. Default value 2.

Text will follow the heading on the same line if the level is greater than both **Hb** and **Hs**.

Post-heading indent
Indentation of the text after the heading is controlled by number register **Hi**, default value 0.

Hi

0	The text will be left-justified.
1	Indentation of the text will follow the value of number register **Pt**, see **P**.
2	The text will be lined up with the first word of the heading.

Centered section headings
All headings whose level is equal or below number register **Hc** and also less than or equal to **Hb** or **Hs** is centerered.

Font control of the heading
The font of each heading level is controlled by string **HF**. It contains a fontnumber or fontname for each level. Default is **2 2 2 2 2 2 2** (all headings in italic). Could also be written as **I I I I I I I**. Note that some other implementations use **3 3 2 2 2 2 2** as the default value. All omitted values are presumed to be a 1.

Point size control.
String **HP** controls the pointsize of each heading, in the same way as **HF** controls the font. A value of 0 selects the default point size. Default value is **0 0 0 0 0 0 0**. Beware that only the point size changes, not the vertical size. That can be controlled by the user specified macro **HX** and/or **HZ**.

Heading counters
Seven number registers, named **H1** thru **H7** contains the counter for each heading level. The values are printed using arabic numerals, this can be changed with the macro **HM** (see below). All marks are concatenated before printing. To avoid this, set number register **Ht** to 1. That will only print the current heading counter at each heading.

Automatic table of contents
All headings whose level is equal or below number register **Cl** is saved to be printed in the table of contents. Default value is 2.

Special control of the heading, user-defined macros.
These macros can be defined by the user to get a finer control of vertical spacing, fonts or other features. Argument *level* is the level-argument to **H**, but 0 for unnumbered headings (see **HU**). Argument *rlevel* is the real level, it is set to number register **Hu** for unnumbered headings. Argument *heading-text* is the text argument to **H** and **HU**.

HX *level rlevel heading-text*
HX is called just before the printing of the heading. The following register is available for **HX**. **HX** may alter **}0**, **}2** and **;3**.

string }0
Contains the heading mark plus two spaces if *rlevel* is non-zero, otherwise empty.

register ;0
Contains the position of the text after the heading. 0 means that the text should follow the heading on the same line, 1 means that a line break should occur before the text and 2 means that a blank line should separate the heading and the text.

string }2
Contains two spaces if register **;0** is 0. It is used to separate the heading from the text. The string is empty if **;0** is non-zero.

register ;3
Contains the needed space in units after the heading. Default is 2v.

Can be used to change things like numbering (**}0**), vertical spacing (**}2**) and the needed space after the heading.

HY *dlevel rlevel heading-text*
HY is called after size and font calculations and might be used to change indentation.

HZ *dlevel rlevel heading-text*
HZ is called after the printing of the heading, just before **H** or **HU** exits. Could be used to change the page header according to the section heading.

HC [hyphenation-character]
Set hyphenation character. Default value is \%. Resets to the default if called without argument. Hyphenation can be turned off by setting number register **Hy** to 0 in the beginning of the file.

HM [arg1 [arg2 [... [arg7]]]]
Heading mark style. Controls the type of marking for printing of the heading counters. Default is 1 for all levels.

Argument

1	Arabic numerals.
0001	Arabic numerals with leading zeroes, one or more.
A	Upper-case alphabetic
a	Lower-case alphabetic
I	Upper-case roman numerals
i	lower-case roman numerals
empty	Arabic numerals.

HU heading-text
Unnumbered section header. **HU** behavies like **H** at the level in number register **Hu**. See **H**.

HX dlevel rlevel heading-text
Userdefined heading exit. Called just before printing the header. See **H**.

HY dlevel rlevel heading-text
Userdefined heading exit. Called just before printing the header. See **H**.

HZ dlevel rlevel heading-text
Userdefined heading exit. Called just after printing the header. See **H**.

I [italic-text [prev-font-text [italic-text [...]]]]
Italic. Changes the font to italic if called without arguments. With one argument it will set the word in italic. With two argument it will concatenate them and set the first word in italic and the second in the previous font. There is no limit on the number of argument, all will be concatenated.

IA [addressee-name [title]]
Begins specification of the addressee and addressee's address in letter style. Several names can be specified with empty **IA**/**IE**-pairs, but only one address. See **LT**.

IB [italic-text [bold-text [italic-text [...]]]]
Italic-bold. Even arguments is printed in italic, odd in boldface. See **I**.

IE Ends the address-specification after **IA**.

INITI type filename [macro]
Initialize the new index system, sets the filename to collect index lines in with **IND**. Argument *type* selects the type of index, page number, header marks or both. The default is *N*.

It is also possible to create a macro that is responsible for formatting each row. Add the name of the macro as argument 3. The macro will be called with the index as argument(s).

type

N	Page numbers
H	Header marks
B	Both page numbers and header marks, tab separated

INITR filename
Initialize the refencemacros. References will be written to stderr and is supposed to be written to *filename.qrf*. Requires two passes with groff, this is handled by a separate program called **mmroff**, the reason is that groff is often installed without the unsafe operations that **INITR** requiered. The first pass looks for references and the second one includes them. **INITR** can be used several times, but it is only the first occurrence of **INITR** that is active.

See also **SETR**, **GETPN** and **GETHN**.

IND arg1 [arg2 [...]]
IND writes a line in the index file selected by **INITI** with all arguments and the page number or header mark separated by tabs.

Examples
arg1\tpage number
arg1\targ2\tpage number
arg1\theader mark
arg1\tpage number\theader mark

INDP **INDP** prints the index by running the command specified by string variable **Indcmd**, normally *sort -t\t*. **INDP** reads the output from the command to form the index, normally in two columns (can be changed by defining **TYIND**). The index is printed with string variable **Index** as header, default is **INDEX**. One-column processing is returned after the list. **INDP** will call the user-defined macros **TXIND**, **TYIND** and **TZIND** if defined. **TXIND** is called before printing **INDEX**, **TYIND** is called instead of printing **INDEX**. **TZIND** is called after the printing and should take care of restoring to normal operation again.

ISODATE [0]
ISODATE changes the predefined date string in **DT** to ISO-format, ie YYYY-MM-DD. This can also be done by adding **-rIso=1** on the command line. Reverts to old date format if argument is **0**.

IR [italic-text [roman-text [italic-text [...]]]]
Italic-roman. Even arguments is printed in italic, odd in roman. See **I**.

LB text-indent mark-indent pad type [mark [LI-space [LB-space]]]
List begin macro. This is the common macro used for all lists. *Text-indent* is the number of spaces to indent the text from the current indent.

Pad and *mark-indent* controls where to put the mark. The mark is placed within the mark area, and *mark-indent* sets the number of spaces before this area. It is normally 0. The mark area ends where the text begins. The start of the text is still controlled by *text-indent*.

The mark is left justified whitin the mark area if *pad* is 0. If *pad* is greater than 0, then *mark-indent* is ignored, and the mark is placed *pad* spaces before the text. This will right justify the mark.

If *type* is 0 the list will have either a hanging indent or, if argument *mark* is given, the string *mark* as mark.

If *type* is greater than 0 automatic numbering will occur, arabic if *mark* is empty. *Mark* can then be any of **1**, **A**, **a**, **I** or **i**.

Type selects one of six possible ways to display the mark.

type	
1	x.
2	x)
3	(x)
4	[x]
5	<x>
6	{x}

Every item in the list will get *LI-space* number of blank lines before them. Default is 1.

LB itself will print *LB-space* blank lines. Default is 0.

LC [list-level]
List-status clear. Terminates all current active lists down to *list-level*, or 0 if no argmuent is given. This is used by **H** to clear any active list.

LE [1] List end. Terminate the current list. **LE** outputs a blank line if an argument is given.

LI [mark [1]]
List item precedes every item in a list. Without argument **LI** will print the mark determined by the current list type. By giving **LI** one argument, it will use that as the mark instead. Two arguments to **LI** will make *mark* a prefix to the current mark. There will be no separating space between the prefix and the mark if the second argument is **2** instead of **1**. This behaviour can also be achieved by setting number register **Limsp** to zero. A zero length *mark* will make a hanging indent instead.

A blank line is normally printed before the list item. This behaviour can be controlled by number register **Ls**. Pre-spacing will occur for each list-level less than or equal to **Ls**. Default value is 99. (Nesting of lists is unlimited. :-)

The indentation can be changed thru number register **Li**. Default is 6.

All lists begins with a list initialization macro, **LB**. There are, however, seven predefined listtypes to make lists easier to use. They all call **LB** with different default values.

AL Automatically Incremented List
ML Marked List
VL Variable-Item List
BL Bullet List
DL Dash List
RL Reference List
BVL Broken Varable List.

These lists are described at other places in this manual. See also **LB**.

LT [arg]

Formats a letter in one of four different styles depending on the argument. See also **INTERNALS**.

Arg	**Style**
BL	Blocked. Date line, return address, writer's address and closing begins at the center of the line. All other lines begin at the left margin.
SB	Semi-blocked. Same as blocked, except that the first line in every paragraph is indented five spaces.
FB	Full-blocked. All lines begin at the left margin.
SP	Simplified. Almost the same as the full-blocked style. Subject and the writer's identification are printed in all-capital.

LO type [arg]

Specify options in letter (see **.LT**). This is a list of the standard options:

CN Confidential notation. Prints **CONFIDENTIAL** on the second line below the date line. Any argument replaces **CONFIDENTIAL**. See also string variable **LetCN**.

RN Reference notation. Prints **In reference to:** and the argument two lines below the date line. See also string variable **LetRN**.

AT Attention. Prints **ATTENTION:** and the argument below the inside address. See also string variable **LetAT**.

SA Salutation. Prints **To Whom It May Concern:** or the argument if it was present. The salutation is printed two lines below the inside address. See also string variable **LetSA**.

SJ Subject line. Prints the argument as subject prefixed with **SUBJECT:** two lines below the inside address, except in letter type **SP**. Then the subject is printed in all-captial without any prefix. See also string variable **LetSJ**.

MC column-size [column-separation]

Begin multiple columns. Return to normal with **1C**. **MC** will create as many columns as the current line length permits. *Column-size* is the width of each column, and *column-separation* is the space between two columns. Default separation is the column-size/15. See also **1C**.

ML mark [text-indent [1]]

Marked list start. The *mark* argument will be printed before each list item. *Text-indent* sets the indent and overrides **Li**. A third argument will prohibit printing of a blank line before each item.

MT [arg [addressee]]

Memorandum type. The *arg* is part of a filename in */usr/share/groff/1.18.1.4/tmac/mm/*.MT*. Memorandum type 0 thru 5 are supported, including *"string"*. *Addressee* just sets a variable, used in the AT&T macros.

arg	
0	Normal memorandum, no type printed
1	Memorandum with *MEMORANDUM FOR FILE* printed
2	Memorandum with *PROGRAMMER'S NOTES* printed
3	Memorandum with *ENGINEER'S NOTES* printed
4	Released paper style
5	External letter style

See also **COVER**/**COVEND**, a more flexible type of front page.

MOVE y-pos [x-pos [line-length]]
Move to a position, pageoffset set to *x-pos*. If *line-length* is not given, the difference between current and new pageoffset is used. Use **PGFORM** without arguments to return to normal.

MULB cw1 space1 [cw2 space2 [cw3 ...]]
Begin a special multi-column mode. Every columns width must be specified. Also the space between the columns must be specified. The last column does not need any space-definition. **MULB** starts a diversion and **MULE** ends the diversion and prints the columns. The unit for width and space is 'n', but **MULB** accepts all normal unitspecifications like 'c' and 'i'. **MULB** operates in a separate environment.

MULN Begin the next column. This is the only way to switch column.

MULE End the multi-column mode and print the columns.

nP [type]
Print numbered paragraph with header level two. See **.P**.

NCOL Force printing to the next column, don't use this together with the **MUL*** macros, see **2C**.

NS [arg [1]]
Prints different types of notations. The argument selects between the predefined type of notations. If the second argument is available, then the argument becomes the entire notation. If the argument doesn't exist in the predefined, it will be printed as **Copy (***arg***) to**. It is possible to add more standard notations, see the string variable **Letns** and **Letnsdef**.

Arg	**Notation**
none	Copy To
""	Copy To
1	Copy To (with att.) to
2	Copy To (without att.) to
3	Att.
4	Atts.
5	Enc.
6	Encs.
7	Under separate cover
8	Letter to
9	Memorandum to
10	Copy (with atts.) to
11	Copy (without atts.) to
12	Abstract Only to
13	Complete Memorandum to
14	CC

ND new-date
New date. Override the current date. Date is not printed if *new-date* is an empty string.

OF [arg]
Odd-page footer, a line printed just above the normal footer. See **EF** and **PF**.

OH [arg]
Odd-page header, a line printed just below the normal header. See **EH** and **PH**.

OP Make sure that the following text is printed at the top of an odd-numbered page. Will not output an empty page if currently at the top of an odd page.

P [type]
Begin new paragraph. **P** without argument will produce left justified text, even the first line of the paragraph. This is the same as setting *type* to 0. If the argument is 1, then the first line of text following **P** will be indented by the number of spaces in number register **Pi**, normally 5.

Instead of giving 1 as argument to **P** it is possible to set the paragraph type in number register **Pt**.

Using 0 and 1 will be the same as adding that value to **P**. A value of 2 will indent all paragraphs, except after headings, lists and displays.

The space between two paragraphs is controlled by number register **Ps**, and is 1 by default (one blank line).

PGFORM [linelength [pagelength [pageoffset [1]]]]
Sets linelength, pagelength and/or pageoffset. This macro can be used for special formatting, like letterheads and other. It is normally the first command in a file, though it's not necessary. **PGFORM** can be used without arguments to reset everything after a **MOVE**. A line-break is done unless the fourth argument is given. This can be used to avoid the pagenumber on the first page while setting new width and length. (It seems as if this macro sometimes doesn't work too well. Use the command line arguments to change linelength, pagelength and pageoffset instead. Sorry.)

PGNH No header is printed on the next page. Used to get rid of the header in letters or other special texts. This macro must be used before any text to inhibit the pageheader on the first page.

PIC [-L] [-C] [-R] [-I n] filename [width [height]]
PIC includes a Postscript file in the document. The macro depends on **mmroff** and **INITR**. **-L**, **-C**, **-R** and **-I n** adjusts the picture or indents it. The optionally *width* and *height* can also be given to resize the picture.

PE Picture end. Ends a picture for **pic**, see the manual for **pic**.

PF [arg]
Page footer. **PF** sets the line to be printed at the bottom of each page. Normally empty. See **PH** for the argument specification.

PH [arg]
Page header, a line printed at the top of each page. The argument should be specified as "'left-part'center-part'right-part'", where left-, center- and right-part is printed left-justified, centered and right justified. The character **%** is changed to the current page number. The default page-header is "''- % -''", the page number between two dashes.

PS Picture start (from pic). Begins a picture for **pic**, see the manual.

PX Page-header user-defined exit. **PX** is called just after the printing of the page header in *no-space* mode.

R Roman. Return to roman font, see also **I**.

RB [roman-text [bold-text [roman-text [...]]]]
Roman-bold. Even arguments is printed in roman, odd in boldface. See **I**.

RD [prompt [diversion [string]]]
Read from standard input to diversion and/or string. The text will be saved in a diversion named *diversion*. Recall the text by writing the name of the diversion after a dot on an empty line. A string will also be defined if *string* is given. *Diversion* and/or *prompt* can be empty ("").

RF Reference end. Ends a reference definition and returns to normal processing. See **RS**.

RI [roman-text [italic-text [roman-text [...]]]]
Even arguments are printed in roman, odd in italic. See **I**.

RL [text-indent [1]]
Reference list start. Begins a list where each item is preceded with a automatically incremented number between square brackets. *Text-indent* changes the default indentation.

RP [arg1 [arg2]]
Produce reference page. **RP** can be used if a reference page is wanted somewhere in the document. It is not needed if **TC** is used to produce a table of content. The reference page will then be printed automatically.

The reference counter will not be reset if *arg1* is 1.

Arg2 tells **RP** whether to eject a page or not.

Arg2

0 The reference page will be printed on a separate page. This is the default.

1 Do not eject page after the list.

2 Do not eject page before the list.

3 Do not eject page before and after the list.

The reference items will be separated by a blank line. Setting number register **Ls** to 0 will suppress the line.

The string **Rp** contains the reference page title and is normally set to *REFERENCES*.

RS [string-name]

RS begins an automatically numbered reference definition. Put the string ***(Rf** where the reference mark should be and write the reference between **RS**/**RF** at next new line after the reference mark. The reference number is stored in number register **:R**. If *string-name* is given, a string with that name will be defined and contain the current reference mark. The string can be referenced as ***[***string-name***]** *later in the text.*

S [size [spacing]]

Set point size and vertical spacing. If any argument is equal 'P', then the previous value is used. A 'C' means current value, and 'D' default value. If '+' or '-' is used before the value, then increment or decrement of the current value will be done.

SA [arg]

Set right-margin justification. Justification is normally turned on. No argumenent or **0** turns off justification, a **1** turns on justification.

SETR refname [string]

Remember the current header and page-number as *refname*. Saves *string* if *string* is defined. *string* is retrieved with **.GETST**. See **INITR**.

SG [arg [1]]

Signature line. Prints the authors name(s) after the formal closing. The argument will be appended to the reference data, printed at either the first or last author. The reference data is the location, department and initials specified with **.AU**. It will be printed at the first author if the second argument is given, otherwise at the last. No reference data will be printed if the author(s) is specifed thru **.WA**/**.WE**. See **INTERNALS**.

SK [pages]

Skip pages. If *pages* is **0** or omitted, a skip to the next page will occur unless it is already at the top of a page. Otherwise it will skip *pages* pages.

SM string1 [string2 [string3]]

Make a string smaller. If *string2* is given, *string1* will be smaller and *string2* normal, concatenated with *string1*. With three argument, all is concatenated, but only *string2* is made smaller.

SP [lines]

Space vertically. *lines* can have any scalingfactor, like *3i* or *8v*. Several **SP** in a line will only produce the maximum number of lines, not the sum. **SP** will also be ignored until the first textline in a page. Add a **\&** before **SP** to avoid this.

TAB reset tabs to every 5n. Normally used to reset any previous tabpositions.

TB [title [override [flag [refname]]]]

Table title, arguments are the same as for **EC**. **TB** uses the number register **Tb** as counter. The string **Lt** controls the title of the List of Tables, default is *LIST OF TABLES*. The List of Tables will only be printed if number register **Lt** is 1, default 1. The string **Litb** contains the word *TABLE*, wich is printed before the number.

Special handling of the title will occur if **TB** is used inside **DS**/**DE**, it will not be affected by the format of **DS**.

TC [slevel [spacing [tlevel [tab [h1 [h2 [h3 [h4 [h5]]]]]]]]]

Table of contents. This macro is normally used at the last line of the document. It generates a table of contents with headings up to the level controlled by number register **Cl**. Note that **Cl** controls the saving of headings, it has nothing to do with **TC**. Headings with level less than or equal to *slevel* will get *spacing* number of lines before them. Headings with level less than or equal to *tlevel* will have their page numbers right justified with dots or spaces separating the text and the page number. Spaces is used if *tab* is greater than zero, otherwise dots. Other headings will have the page number directly at the end of the heading text (*ragged right*).

The rest of the arguments will be printed, centered, before the table of contents.

The user-defined macros **TX** and **TY** are used if **TC** is called with at most four arguments. **TX** is called before the printing of *CONTENTS*, and **TY** is called instead of printing *CONTENTS*.

Equivalent macros can be defined for list of figures, tables, equations and excibits by defining **TXxx** or **TYxx**, where **xx** is **Fg**, **TB**, **EC** or **EX**.

String **Ci** can be set to control the indentations for each heading-level. It must be scaled, like **.ds Ci .25i .5i .75i 1i 1i**. The indentation is normally controlled by the maxlength of headings in each level.

All texts can be redefined, new stringvariables *Lifg*, *Litb*, *Liex*, *Liec* and *Licon* contain "Figure", "TABLE", "Exhibit", "Equation" and "CONTENTS". These can be redefined to other languages.

TE Table end. See **TS**.

TH [N] Table header. See **TS**. **TH** ends the header of the table. This header will be printed again if a page-break occurs. Argument *N* isn't implemented yet.

TL [charging-case number(s) [filing-case number(s)]]

Begin title of memorandum. All text up to the next **AU** is included in the title. *Charging-case number* and *filing-case* are saved for use in the front page processing.

TM [num1 [num2 [...]]]

Technical memorandumnumbers used in **.MT**. Unlimited number of arguments may be given.

TP Top of page user-defined macro. This macro is called instead of the normal page header. It is possible to get complete control over the header. Note that header and footer is printed in a separate environment. Linelength is preserved though.

TS [H] Table start. This is the start of a table specification to **tbl**. See separate manual for **tbl**. **TS** ends with **TE**. Argument *H* tells **mm** that the table has a header. See **TH**.

TX Userdefined table of contents exit. This macro is called just before **TC** prints the word *CONTENTS*. See **TC**.

TY Userdefined table of contents exit (no "CONTENTS"). This macro is called instead of printing *CONTENTS*. See **TC**.

VERBON [flag [pointsize [font]]]

Begin verbatim output using courier font. Usually for printing programs. All character has equal width. The pointsize can be changed with the second argument. By specifying the font-argument it is possible to use another font instead of courier. *flag* controls several special features. It contains the sum of all wanted features.

Value	Description
1	Disable the escape-character (\). This is normally turned on during verbose output.
2	

Add an empty line before the verbose text.

4 Add an empty line after the verbose text.

8 Print the verbose text with numbered lines. This adds four digitsized spaces in the beginning of each line. Finer control is available with the string-variable **Verbnm**. It contains all arguments to the **troff**-command **.nm**, normally '1'.

16 Indent the verbose text with five 'n':s. This is controlled by the number-variable **Verbin** (in units).

VERBOFF

End verbatim output.

VL text-indent [mark-indent [1]]

Variable-item list has no fixed mark, it assumes that every **LI** have a mark instead. *Text-indent* sets the indent to the text, and *mark-indent* the distance from the current indent to the mark. A third argument will prohibit printing of a blank line before each item.

VM [-T] [top [bottom]]

Vertical margin. Adds extra vertical top and margin space. Option **-T** set the total space instead. No argument resets the margin to zero or the default *(7v 5v)* if **-T** was used. It is higly recommended that macro **TP** and/or **EOP** are defined if using **-T** and setting top and/or bottom margin to less than the default.

WA [writer-name [title]]

Begins specification of the writer and writer's address. Several names can be specified with empty **WA**/**WE**-pairs, but only one address.

WE Ends the address-specification after **.WA**.

WC [format]

Footnote and display width control.

N Set default mode, **-WF**, **-FF**, **-WD** and **FB**.

WF Wide footnotes, wide also in two-column mode.

-WF Normal footnote width, follow column mode.

FF All footnotes gets the same width as the first footnote encountered.

-FF Normal footnotes, width follows **WF** and **-WF**.

WD Wide displays, wide also in two-column mode.

-WD Normal display width, follow column mode.

FB Floating displays generates a line break when printed on the current page.

-FB Floating displays does not generate line break.

Strings used in mm:

App A string containing the word "APPENDIX".

Apptxt

The current appendix text.

EM Em dash string

H1txt Will be updated by **.H** and **.HU** to the current heading text. Also updated in table of contents & friends.

HF Fontlist for headings, normally "2 2 2 2 2 2 2". Nonnumeric fontnames may also be used.

HP Pointsize list for headings. Normally "0 0 0 0 0 0 0" which is the same as "10 10 10 10 10 10 10".

Index

Contains *INDEX*.

Indcmd

Contains the index command, *sort -t\t*.

Lifg String containing *Figure*.

Litb String containing *TABLE*.

Liex String containing *Exhibit*.

Liec String containing *Equation*.

Licon String containing *CONTENTS*.

Lf Contains "LIST OF FIGURES".

Lt Contains "LIST OF TABLES".

Lx Contains "LIST OF EXHIBITS".

Le Contains "LIST OF EQUATIONS".

Letfc Contains "Yours very truly,", used in **.FC**.

Letapp Contains "APPROVED:", used in **.AV**.

Letdate

Contains "Date", used in **.AV**.

LetCN Contains "CONFIDENTIAL", used in **.LO CN**.

LetSA Contains "To Whom It May Concern:", used in **.LO SA**.

LetAT Contains "ATTENTION:", used in **.LO AT**.

LetSJ Contains "SUBJECT:", used in **.LO SJ**.

LetRN Contains "In reference to:", used in **.LO RN**.

Letns is an array containing the different strings used in **.NS**. It is really a number of stringvariables prefixed with **Letns!**. If the argument doesn't exist, it will be included between **()** with **Letns!copy** as prefix and **Letns!to** as suffix. Observe the space after **copy** and before **to**.

Name	**Value**
Letns!0	Copy to
Letns!1	Copy (with att.) to
Letns!2	Copy (without att.) to
Letns!3	Att.
Letns!4	Atts.
Letns!5	Enc.
Letns!6	Encs.
Letns!7	Under separate cover
Letns!8	Letter to
Letns!9	Memorandum to
Letns!10	Copy (with atts.) to
Letns!11	Copy (without atts.) to
Letns!12	Abstract Only to
Letns!13	Complete Memorandum to
Letns!14	CC
Letns!copy	Copy "
Letns!to	" to

Letnsdef

Defines the standard-notation used when no argument is given to **.NS**. Default is **0**.

MO1 - MO12
Strings containing *January* thru *December*.

Qrf String containing "See chapter *[Qrfh], page \\n[Qrfp].".

Rp Contains "REFERENCES".

Tcst Contains current status of table of contents and list of XXXX. Empty outside **.TC**. Useful in user-defined macros like **.TP**.

Value	**Meaning**
co	Table of contents
fg	List of figures
tb	List of tables
ec	List of equations
ex	List of exhibits
ap	Appendix

Tm Contains \(tm, trade mark.

Verbnm
Argument to **.nm** in **.VERBON**, default: **1**.

Number variables used in mm:

Aph Print an appendix-page for every new appendix if this numbervariable is non-zero. No output will occur if **Aph** is zero, but there will always be an appendix-entry in the 'List of contents'.

Cl Contents level [0:7], contents saved if heading level <= Cl, default 2.

Cp Eject page between LIST OF XXXX if Cp == 0, default 0.

D Debugflag, values >0 produces varying degree of debug. A value of 1 gives information about the progress of formatting, default 0.

De Eject after floating display is output [0:1], default 0.

Dsp Controls the space output before and after static displays if defined. Otherwise is the value of Lsp used.

Df Floating keep output [0:5], default 5.

Ds **Lsp** space before and after display if == 1 [0:1], default 1.

Ej Eject page, default 0.

Eq Equation lable adjust 0=left, 1=right. Default 0.

Fs Footnote spacing, default 1.

H1-H7 Heading counters

H1dot Append a dot after the level one heading number if > 0. Default is 1.

H1h

Copy of number register **H1**, but it is incremented
just before the page break. Useful in user defined header macros. **Hb** Heading break level [0:7], default 2.

Hc Heading centering level, [0:7]. Default 0.

Hi Heading temporary indent [0:2], default 1.
0 -> 0 indent, left margin
1 -> indent to right , like .P 1
2 -> indent to line up with text part of preceding heading

Hps Numbervariable with the heading pre-space level. If the heading-level is less than or equal to **Hps**, then two lines will precede the section heading instead of one. Default is first level only. The real amount of lines is controlled by the variables **Hps1** and **Hps2**.

Hps1 This is the number of lines preceding **.H** when the heading-level is greater than **Hps**. Value is in units, normally 0.5.

Hps2 This is the number of lines preceding **.H** when the heading-level is less than or equal to **Hps**. Value is in units, normally 1.

Hs Heading space level [0:7], default 2.

Hss This is the number of lines that follows **.H** when the heading-level is less than or equal to **Hs**. Value is in units, normally 1.

Ht Heading numbering type, default 0. 0 -> multiple (1.1.1 ...)
1 -> single

Hu Unnumbered heading level, default 2.

Hy Hyphenation in body, default 1.
0 -> no hyphenation
1 -> hyphenation 14 on

Iso Set this variable to 1 on the command line to get ISO-formatted date string. (**-rIso=1**) Useless inside a document.

L Page length, only for command line settings.

Letwam
Max lines in return-address, used in **.WA**/**.WE**. Default 14.

Lf, Lt, Lx, Le
Enables (1) or disables (0) the printing of List of figures, List of tables, List of exhibits and List of equations. Default: Lf=1, Lt=1, Lx=1, Le=0.

Li List indent, used by .AL, default 6.

Limsp Flag for space between prefix and mark in automatic lists (.AL).
0 == no space
1 == space

Ls List space, if current listlevel > Ls then no spacing will occur around lists. Default 99.

Lsp The size of an empty line. Normally 0.5v, but it is 1v if **n** is set (**.nroff**).

N Numbering style [0:5], default 0.
0 == (default) normal header for all pages.
1 == header replaces footer on first page, header is empty.
2 == page header is removed on the first page.
3 == "section-page" numbering enabled.
4 == page header is removed on the first page.
5 == "section-page" and "section-figure" numbering enabled. See also the number-register Sectf and Sectp.

Np Numbered paragraphs, default 0.
0 == not numbered
1 == numbered in first level headings.

O Page offset, only for command line settings.

Of Format of figure,table,exhibit,equation titles, default 0.
0 = ". "
1 = " - "

P Current page-number, normally the same as % unless "section-page" numbering is enabled.

Pi paragraph indent, default 5.

Pgps Controls whether header and footer pointsize should follow the current setting or just change when the header and footer is defined.

Value	Description
0	Pointsize will only change to the current setting when **.PH**, **.PF**, **.OH**, **.EH**, **.OF** or **.OE** is executed.
1	Pointsize will change after every **.S**. This is the default.

Ps paragraph spacing, default 1.

Pt Paragraph type, default 0.
0 == left-justified
1 == indented .P
2 == indented .P except after .H, .DE or .LE.

Sectf Flag controlling "section-figures". A non-zero value enables this. See also register N.

Sectp Flag controlling "section-page-numbers". A non-zero value enables this. See also register N.

Si Display indent, default 5.

Verbin Indent for **.VERBON**, default 5n.

W Line length, only for command line settings.

.mgm Always 1.

INTERNALS

The letter macros is using different submacros depending on the letter type. The name of the submacro has the letter type as suffix. It is therefore possible to define other letter types, either in the national macro-file, or as local additions. **.LT** will set the number variables **Pt** and **Pi** to 0 and 5. The following strings and macros must be defined for a new letter type:

let@init_*type*
This macro is called directly by **.LT**. It is supposed to initialize variables and other stuff.

let@head_*type*
This macro prints the letter head, and is called instead of the normal page header. It is supposed to remove the alias **let@header**, otherwise it will be called for all pages.

let@sg_*type name title n flag [arg1 [arg2 [...]]]*
.SG is calling this macro only for letters, memorandums has its own processing. *name* and *title* is specified thru **.WA**/**.WB**. *n* is the counter, 1-max, and *flag* is true for the last name. Any other argument to **.SG** is appended.

let@fc_*type closing*
This macro is called by **.FC**, and has the formal closing as argument.

.LO is implemented as a general option-macro. **.LO** demands that a string named **Let***type* is defined, where *type* is the letter type. **.LO** will then assign the argument to the string variable **let*lo-***type*.

AUTHOR

Jörgen Hägg, Lund, Sweden <jh@axis.se>.

FILES

/usr/share/groff/1.18.1.4/tmac/tmac.m

/usr/share/groff/1.18.1.4/tmac/mm/*.cov

/usr/share/groff/1.18.1.4/tmac/mm/*.MT

/usr/share/groff/1.18.1.4/tmac/mm/locale

SEE ALSO

groff(1), **troff**(1), **tbl**(1), **pic**(1), **eqn**(1)
groff_mmse(7)