

**NAME**

groff\_mm – memorandum macros for GNU roff

**SYNOPSIS**

**groff -mm** [*option ...*] [*file ...*]  
**groff -m mm** [*option ...*] [*file ...*]

**DESCRIPTION**

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

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

**mm** is intended to support easy localization. Use **mmse** as an example how to adapt the output format to a national standard. Localized strings are collected in the file */usr/share/groff/1.22.4/tmac/xx.tmac*, where *xx* denotes the two-letter code for the *language*, as defined in the ISO 639 standard. For Swedish, this is 'sv.tmac' – not 'se', which is the ISO 3166 two-letter code for the *country* (as used for the output format localization).

A file called *locale* or *country\_locale* is read after the initialization of the global variables. It is therefore possible to localize the macros with a different company name and so on.

In this manual, square brackets are 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 [ $\pm$ ]*n* [*i*]

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

Strings are 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.

<b>Form</b>	<b>Sequence</b>
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, ...

**Fonts**

In **mm**, the fonts (or rather, font styles) **R** (normal), **I** (italic), and **B** (bold) are hardwired to font positions **1**, **2**, and **3**, respectively. Internally, font positions are used for backwards compatibility. From a practical point of view it doesn't make a big difference – a different font family can still be selected with a call to the **.fam** request or using **groff**'s **-f** command-line option. On the other hand, if you want to replace just, say,

font **B**, you have to replace the font at position 2 (with a call to ‘.fp 2 . . .’).

## Macros

### **)E** *level text*

Add heading text *text* to the table of contents with *level*, which is either 0 or in the range 1 to 7. See also **.H**. This macro is used for customized tables of contents.

**1C** [**1**] Begin one-column processing. A **1** as an 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*]

Author’s firm, should be called before **AU**, see also **COVER**.

### **AL** [*type* [*text-indent* [**1**]]]

Start auto-increment list. Items are numbered beginning with one. The *type* argument controls the format of numbers.

<b>Arg</b>	<b>Description</b>
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 indentation and overrides **Li**. A third argument prohibits printing of a blank line before each item.

### **APP** *name text*

Begin an appendix with name *name*. Automatic naming occurs if *name* is "". The appendices start with **A** if automatic naming is used. A 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 appears in the ‘List of contents’ with correct page numbers. 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 page number is incremented with *pages*. This is used when diagrams or other non-formatted documents are included as appendices.

### **AS** [*arg* [*indent*]]

Abstract start. Indentation is specified in ‘ens’, but scaling is allowed. Argument *arg* controls where the abstract is printed.

#### **Arg**    **Placement**

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

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

### **AST** [*title*]

Abstract title. Default is ‘ABSTRACT’. Sets the text above the abstract text.

### **AT** *title1* [*title2* [. . .]]

Author’s title. **AT** must appear just after each **AU**. The title shows 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 is 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**; it is replaced with an empty line if there is a second argument. The string ‘Date’ can be changed with variable **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 are concatenated to one word; the first, third and so on is printed in boldface.

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

**B2** End box. Finishes the box started with **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. Initializes 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 prohibits 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. The 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 always begins at the next line after the mark. *text-indent* sets the indentation to the text, and *mark-indent* the distance from the current indentation to the mark. A third argument prohibits printing of a blank line before each item.

**COVER** [*arg*]

Begin a coversheet definition. It is important that **.COVER** appears before any normal text. This macro uses *arg* to build the filename */usr/share/groff/1.22.4/tmac/mm/arg.cov*. Therefore it is possible to create unlimited types of cover sheets. *ms.cov* is supposed to look like the *ms* cover sheet. **.COVER** requires a **.COVEND** at the end of the cover definition. Always use this order of the cover macros:

```
.COVER
.TL
.AF
.AU
.AT
.AS
.AE
.COVEND
```

However, only **.TL** and **.AU** are required.

**COVEND**

Finish the cover description and print the cover page. It is defined in the cover file.

**DE** Display end. Ends a block of text or 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* are 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 occurs 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 is printed on the current page if there is enough space, otherwise it is printed at the end of the document.
- 2 One display is 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 is printed at the top of the next page or column.
- 4 Print as many displays as possible in a new page or column. A page break occurs 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 occurs 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 an empty line between each list item. See **LI**. A third argument prohibits 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 arbitrarily.

**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.

The values 'L', 'I', 'C', and 'CB' can also be specified as '0', '1', '2', and '3', respectively, for compatibility reasons.

**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' and '1', respectively.

By default, an empty line is printed before and after the display. Setting number register **Ds** to 0 prevents 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 changes the numbering.

**flag**

none	<i>override</i> is a prefix to the number.
0	<i>override</i> is a prefix to the number.
1	<i>override</i> is a suffix to the number.
2	<i>override</i> replaces the number.

**EC** uses the number register **Ec** as a counter. It is possible to use **.af** to change the format of the number. If number register **Of** is 1, the format of title uses 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 is only printed if number register **Le** is 1. The default is 0. The string **Liec** contains the word 'Equation', which 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 occurs if **EC** is used inside **DS/DE**; it is not affected by the format of **DS**.

**EF** [*arg*]

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

This macro defines string **EOPef**.

**EH** [*arg*]

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

This macro defines string **TPeh**.

**EN** Equation end, see **EQ**.

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

**Strings available to EOP**

EOPf	argument of <b>PF</b>
EOPef	argument of <b>EF</b>
EOPof	argument of <b>OF</b>

**EPIC** [-**L**] *width height* [*name*]

Draw a box with the given *width* and *height*. It also prints 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** left-adjusts the picture; the default is to center. See **PIC**.

**EQ** [*label*]

Equation start. **EQ/EN** are the delimiters for equations written for **eqn(1)**. **EQ/EN** must be inside of a **DS/DE** pair, except if **EQ** is used to set options for **eqn** only. The *label* argument appears at the right margin of the equation, centered vertically within the **DS/DE** block, unless number register **Eq** is 1. Then the label appears at the left margin.

If there are multiple **EQ/EN** blocks within a single **DS/DE** pair, only the last equation label (if any) is printed.

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

Exhibit title. The arguments are the same as for **EC**. **EX** uses the number register **Ex** as a counter. The string **Lx** controls the title of the List of Exhibits; default is 'LIST OF EXHIBITS'. The List of Exhibits is only printed if number register **Lx** is 1, which is the default. The string **Lieux** contains the word 'Exhibit', which is printed before the number. If *refname* is used, the exhibit

number is saved with **.SETR**, and can be retrieved with **'GETST *refname*'**.

Special handling of the title occurs if **EX** is used inside **DS/DE**; it is not affected by the format of **DS**.

#### **FC** [*closing*]

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

#### **FD** [*arg* [1]]

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

<b>arg</b>	<b>hyphen</b>	<b>adjust</b>	<b>indent</b>	<b>ljust</b>
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

An argument greater than or equal to 11 is considered as value 0. Default for **mm** is 10.

**FE** Footnote end.

#### **FG** [*title* [*override* [*flag* [*refname*]]]]

Figure title. The arguments are the same as for **EC**. **FG** uses the number register **Fg** as a counter. The string **Lf** controls the title of the List of Figures; default is 'LIST OF FIGURES'. The List of Figures is only printed if number register **Lf** is 1, which is the default. The string **Lifg** contains the word 'Figure', which 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 occurs if **FG** is used inside **DS/DE**, it is not affected by the format of **DS**.

#### **FS** [*label*]

Footnote start. The footnote is ended by **FE**. By default, footnotes are 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 are 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*]

Include the header number where the corresponding **'SETR *refname*'** was placed. This is displayed as 'X.X.X.' in pass 1. See **INITR**. If *varname* is used, **GETHN** sets the string variable *varname* to the header number.

#### **GETPN** *refname* [*varname*]

Include the page number where the corresponding **'SETR *refname*'** was placed. This is displayed as '9999' in pass 1. See **INITR**. If *varname* is used, **GETPN** sets the stringvariable *varname* to the page number.

#### **GETR** *refname*

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



**Point size control**

String **HP** controls the point size 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 0 0 0 0 0 0 0**

Beware that only the point size changes, not the vertical size. The latter can be controlled by the user-specified macros **HX** and/or **HZ**.

**Heading counters**

Fourteen number registers named **H1** up to **H14** contain 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. This only prints the current heading counter at each heading.

**Automatic table of contents**

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

**Special control of the heading, user-defined macros**

The following 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*

This macro is called just before the printing of the heading. The following registers are available for **HX**. Note that **HX** may alter **}0**, **}2**, and **;3**.

**}0** (string)

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

**;0** (register)

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.

**}2** (string)

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.

**;3** (register)

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*

This macro is called after size and font calculations and might be used to change indentation.

**HZ** *dlevel rlevel heading-text*

This macro is called after the printing of the heading, just before **H** or **HU** exits. Can 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 at the beginning of the file.

**HM** [*arg1* [*arg2* [... [*arg14*]]]]

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
""	Arabic numerals.

**HU** *heading-text*

Unnumbered section header. **HU** behaves like **H** at the level in number register **Hu**. See **H**.

**HX** *dlevel rlevel heading-text*

User-defined heading exit. Called just before printing the header. See **H**.

**HY** *dlevel rlevel heading-text*

User-defined heading exit. Called just before printing the header. See **H**.

**HZ** *dlevel rlevel heading-text*

User-defined 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 sets the word in italic. With two arguments it concatenates them and sets the first word in italic and the second in the previous font. There is no limit on the number of argument; all are concatenated.

**IA** [*addressee-name* [*title*]]

Begin 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 are printed in italic, odd in boldface. See **I**.

**IE** End the address specification after **IA**.

**INITI** *type filename* [*macro*]

Initialize the new index system and set 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 page numbers.

It is also possible to create a macro that is responsible for formatting each row; just add the name of the macro as a third argument. The macro is then called with the index as argument(s).

**type**

N	Page numbers
H	Header marks
B	Both page numbers and header marks, separated with a tab character.

**INTR** *filename*

Initialize the cross reference macros. Cross references are written to stderr and are supposed to be redirected into file *filename.qrf*. Requires two passes with groff; this is handled by a separate program called **mmroff**(1). This program exists because **groff**(1) by default deactivates the unsafe operations that are required by **INTR**. The first pass looks for cross references, and the second one includes them. **INTR** can be used several times, but it is only the first occurrence of **INTR** that is active.

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

**IND** *arg1* [*arg2* [. . .]]

Write 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** Print the index by running the command specified by string variable **Indcmd**, which has ‘sort -t\’ as the default value. **INDP** reads the output from the command to form the index, by default in two columns (this can be changed by defining **TYIND**). The index is printed with string variable **Index** as header, default is ‘INDEX’. One-column processing is reactivated after the list. **INDP** calls the user-defined macros **TXIND**, **TYIND**, and **TZIND** if defined. **TXIND** is called before printing the string ‘INDEX’, **TYIND** is called instead of printing ‘INDEX’, and **TZIND** is called after the printing and should take care of restoring to normal operation again.

**ISODATE** [0]

Change the predefined date string in **DT** to ISO-format, this is, ‘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 are 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 indentation.

*pad* and *mark-indent* control where to put the mark. The mark is placed within the mark area, and *mark-indent* sets the number of spaces before this area. By default it is 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 within the mark area if *pad* is 0. If *pad* is greater than 0, *mark-indent* is ignored, and the mark is placed *pad* spaces before the text. This right-justifies the mark.

If *type* is 0 the list either has a hanging indentation or, if argument *mark* is given, the string *mark* as a mark.

If *type* is greater than 0 automatic numbering occurs, using arabic numbers 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 gets *LI-space* number of blank lines before them. Default is 1.

**LB** itself prints *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 argument is given. This is used by **H** to clear any active list.

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

**LI** [*mark* [1|2]]

List item preceding every item in a list. Without argument, **LI** prints the mark determined by the current list type. By giving **LI** one argument, it uses that as the mark instead. Two arguments to **LI** makes *mark* a prefix to the current mark. There is 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* makes a hanging indentation instead.

A blank line is printed before the list item by default. This behaviour can be controlled by number register **Ls**. Pre-spacing occurs for each list level less than or equal to **Ls**. Default value is 99. There is no nesting limit.

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

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

<b>AL</b>	Automatically Incremented List
<b>ML</b>	Marked List
<b>VL</b>	Variable-Item List
<b>BL</b>	Bullet List
<b>DL</b>	Dash List
<b>RL</b>	Reference List
<b>BVL</b>	Broken Variable List.

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

**LT** [*arg*]

Format a letter in one of four different styles depending on the argument. Also see section “Internals” below.

<b>Arg</b>	<b>Style</b>
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 <b>LetCN</b> .
RN	Reference notation. Prints ‘In reference to:’ and the argument two lines below the date line. See also string variable <b>LetRN</b> .
AT	Attention. Prints ‘ATTENTION:’ and the argument below the inside address. See also string variable <b>LetAT</b> .
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 <b>LetSA</b> .
SJ	Subject line. Prints the argument as subject prefixed with ‘SUBJECT:’ two lines below the inside address, except in letter type ‘SP’, where the subject is printed in all-capital without any prefix. See also string variable <b>LetSJ</b> .

**MC** *column-size* [*column-separation*]

Begin multiple columns. Return to normal with **1C**. **MC** creates 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 *column-size*/15. See also **1C**.

**ML** *mark* [*text-indent* [**1**]]

Marked list start. The *mark* argument is printed before each list item. *text-indent* sets the indent and overrides **Li**. A third argument prohibits printing of a blank line before each item.

**MT** [*arg* [*addressee*]]

Memorandum type. The argument *arg* is part of a filename in */usr/share/groff/1.22.4/tmac/mm/\*.MT*. Memorandum types 0 to 5 are supported, including type 'string' (which gets internally mapped to type 6). *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, setting page offset to *x-pos*. If *line-length* is not given, the difference between current and new page offset is used. Use **PGFORM** without arguments to return to normal.

**MULB** *cw1 space1* [*cw2 space2* [*cw3 . . .*]]

Begin a special multi-column mode. All columns widths must be specified. The space between the columns must be specified also. 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 the width and space arguments is 'n', but **MULB** accepts all normal unit specifications like 'c' and 'i'. **MULB** operates in a separate environment.

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

Print 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 select a predefined type, it is printed as 'Copy (*arg*) to'. It is possible to add more standard notations, see the string variables **Letns** and **Letnsdef**.

<b>Arg</b>	<b>Notation</b>
<i>none</i>	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. Overrides 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**.

This macro defines string **EOPof**.

**OH** [*arg*]

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

This macro defines string **TPoh**.

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

**P** [*type*]

Begin new paragraph. **P** without argument produces left-justified text, even the first line of the paragraph. This is the same as setting *type* to 0. If the argument is 1, the first line of text following **P** is indented by the number of spaces in number register **Pi**, by default 5.

Instead of giving an argument to **P** it is possible to set the paragraph type in number register **Pt**. Using 0 and 1 is the same as adding that value to **P**. A value of 2 indents all paragraphs, except after headings, lists, and displays (this value can't be used as an argument to **P** itself).

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

**PGFORM** [*linelength* [*pagelength* [*pageoffset* [**1**]]]]

Set line length, page length, and/or page offset. This macro can be used for special formatting, like letter heads and other. It is normally the first command in a file, though it is not necessary. **PGFORM** can be used without arguments to reset everything after a **MOVE** call. A line break is done unless the fourth argument is given. This can be used to avoid the page number 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 line length, page length, and page offset instead.)

**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 page header on the first page.

**PIC** [**-B**] [**-L**] [**-C**] [**-R**] [**-I** *n*] *filename* [*width* [*height*]]

Include a PostScript file in the document. The macro depends on **mmroff**(1) and **INITR**. The arguments **-L**, **-C**, **-R**, and **-I** *n* adjust the picture or indent it. With no flag the picture is adjusted to the left. Adding **-B** draws a box around the picture. The optional *width* and *height* can also be given to resize the picture.

**PE** Picture end. Ends a picture for **pic**(1).

**PF** [*arg*]

Page footer. **PF** sets the line to be printed at the bottom of each page. Empty by default. See **PH** for the argument specification.

This macro defines string **EOPf**.

**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-part*, *center-part*, and *right-part* are printed left-justified, centered, and right justified, respectively. Within the argument to **PH**, the character '%' is changed to the current page number.

The default argument is

```
""- % -""
```

which gives the page number between two dashes.

This macro defines string **TPh**.

- PS** Picture start (from pic). Begins a picture for **pic(1)**.
- PX** Page header user-defined exit. This macro 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 are printed in roman, odd in boldface. See **I**.
- RD** [*prompt* [*diversion* [*string*]]]  
Read from standard input to diversion and/or string. The text is 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 is also 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* [...]]]]  
Print even arguments in roman, odd in italic. See **I**.
- RL** [*text-indent*[**1**]]  
Reference list start. Begins a list where each item is preceded with an automatically incremented number between square brackets. *text-indent* changes the default indentation.
- RP** [*arg1* [*arg2*]]  
Produce reference page. This macro 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 contents. The reference page is then printed automatically.  
The reference counter is not reset if *arg1* is 1.  
*arg2* tells **RP** whether to eject a page or not.
- arg2**
- 0 The reference page is printed on a separate page.
  - 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 are separated by a blank line. Setting number register **Ls** to 0 suppresses the line.
- The string **Rp** contains the reference page title and is set to 'REFERENCES' by default. The number register **Rpe** holds the default value for the second argument of **RP**; it is initially set to 0.
- RS** [*string-name*]  
Begin 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 is defined and contains 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 to 'P', the previous value is used. A 'C' means current value, and 'D' the default value. If '+' or '-' is used before the value, the current value is incremented or decremented, respectively.

- SA** [*arg*]  
 Set right-margin justification. Justification is turned on by default. No argument or value ‘0’ turns off justification, and ‘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 **INTR**.
- SG** [*arg* [1]]  
 Signature line. Prints the authors name(s) after the formal closing. The argument is 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 is printed at the first author if the second argument is given, otherwise at the last. No reference data is printed if the author(s) is specified through **.WA/WE**. See section “Internals” below.
- SK** [*pages*]  
 Skip pages. If *pages* is 0 or omitted, a skip to the next page occurs unless it is already at the top of a page. Otherwise it skips *pages* pages.
- SM** *string1* [*string2* [*string3*]]  
 Make a string smaller. If *string2* is given, *string1* is made smaller and *string2* stays at normal size, concatenated with *string1*. With three arguments, everything is concatenated, but only *string2* is made smaller.
- SP** [*lines*]  
 Space vertically. *lines* can have any scaling factor, like ‘3i’ or ‘8v’. Several **SP** calls in a line only produces the maximum number of lines, not the sum. **SP** is ignored also until the first text line in a page. Add **\&** before a call to **SP** to avoid this.
- TAB** Reset tabs to every 5n. Normally used to reset any previous tab positions.
- TB** [*title* [*override* [*flag* [*refname*]]]]  
 Table title. The arguments are the same as for **EC**. **TB** uses the number register **Tb** as a counter. The string **Lt** controls the title of the List of Tables; default value is ‘LIST OF TABLES’. The List of Tables is only printed if number register **Lt** is 1, which is the default. The string **Litb** contains the word ‘TABLE’, which is printed before the number.  
 Special handling of the title occurs if **TB** is used inside **DS/DE**, it is not affected by the format of **DS**.
- TC** [*slevel* [*spacing* [*tlevel* [*tab* [*h1* [*h2* [*h3* [*h4* [*h5*]]]]]]]]]]]]]]  
 Table of contents. This macro is normally used as 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 a level less than or equal to *slevel* get *spacing* number of lines before them. Headings with a level less than or equal to *tlevel* have their page numbers right-justified with dots or spaces separating the text and the page number. Spaces are used if *tab* is greater than zero, dots otherwise. Other headings have the page number directly at the end of the heading text (*ragged-right*).  
 The rest of the arguments is 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 the string ‘CONTENTS’, and **TY** is called instead of printing ‘CONTENTS’.  
 Equivalent macros can be defined for list of figures, tables, equations and exhibits by defining **TX<sub>xx</sub>** or **TY<sub>xx</sub>**, where *xx* is ‘Fg’, ‘TB’, ‘EC’, or ‘EX’, respectively.  
 String **Ci** can be set to control the indentations for each heading-level. It must be scaled, like  
 .ds Ci .25i .5i .75i li li

By default, the indentation is controlled by the maximum length of headings in each level.

The string variables **Lifg**, **Litb**, **Liex**, **Liec**, and **Licon** contain ‘Figure’, ‘TABLE’, ‘Exhibit’, ‘Equation’, and ‘CONTENTS’, respectively. 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 is printed again if a page break occurs. Argument ‘N’ isn’t implemented yet.

**TL** [*charging-case-number* [*filing-case-number*]]

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

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

Technical memorandum numbers used in **.MT**. An 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 the header and the footer are printed in a separate environment. Line length is preserved, though. See **EOP**.

#### strings available to TP

TPh argument of **PH**

TPeh argument of **EH**

TPoh argument of **OH**

**TS** [**H**] Table start. This is the start of a table specification to **tbl**(1). **TS** ends with **TE**. Argument ‘H’ tells **mm** that the table has a header. See **TH**.

**TX** User-defined table of contents exit. This macro is called just before **TC** prints the word ‘CONTENTS’. See **TC**.

**TY** User-defined table of contents exit. This macro is called instead of printing ‘CONTENTS’. See **TC**.

**VERBON** [*flag* [*point-size* [*font*]]]

Begin verbatim output using Courier font. Usually for printing programs. All characters have equal width. The point size can be changed with the second argument. By specifying a third argument it is possible to use another font instead of Courier. *flag* controls several special features. Its value is the sum of all wanted features.

#### Arg 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 digit-sized spaces in the beginning of each line. Finer control is available with the string variable <b>Verbnm</b> . It contains all arguments to the <b>troff</b> (1) command <b>.nm</b> , normally ‘1’. |
| 16 | Indent the verbose text by ‘5n’. This is controlled by the number-variable <b>Verbin</b> (in units).  |

#### VERBOFF

End verbatim output.

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

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

**VM** [**-T**] [*top* [*bottom*]]

Vertical margin. Increase the top and bottom margin by *top* and *bottom*, respectively. If option **-T** is specified, set those margins to *top* and *bottom*. If no argument is given, reset the margin to

zero, or to the default ('7v 5v') if **-T** is used. It is highly recommended that macros **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*]]

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

**WE** End the address specification after **.WA**.

**WC** [*format1*] [*format2*] [...]

Footnote and display width control.

**N** Set default mode which is equal to using the options **-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** Updated by **.H** and **.HU** to the current heading text. Also updated in table of contents & friends.

**HF** Font list for headings, '2 2 2 2 2 2' by default. Non-numeric font names may also be used.

**HP** Point size list for headings. By default, this is '0 0 0 0 0 0' which is the same as '10 10 10 10 10 10'.

**Index** Contains the string 'INDEX'.

**Indcmd**

Contains the index command. Default value is 'sort -t\|'.

**Lifg** String containing 'Figure'.

**Litb** String containing 'TABLE'.

**Liex** String containing 'Exhibit'.

**Liec** String containing 'Equation'.

**Licon** String containing 'CONTENTS'.

**Lf** Contains the string 'LIST OF FIGURES'.

**Lt** Contains the string 'LIST OF TABLES'.

**Lx** Contains the string 'LIST OF EXHIBITS'.

**Le** Contains the string 'LIST OF EQUATIONS'.

**Letfc** Contains the string 'Yours very truly,', used in **.FC**.

**Letapp** Contains the string 'APPROVED:', used in **.AV**.

**Letdate**

Contains the string 'Date', used in **.AV**.

**LetCN** Contains the string 'CONFIDENTIAL', used in **.LO CN**.

**LetSA** Contains the string 'To Whom It May Concern:', used in **.LO SA**.

- LetAT** Contains the string ‘ATTENTION:’, used in **.LO AT**.
- LetSJ** Contains the string ‘SUBJECT:’, used in **.LO SJ**.
- LetRN** Contains the string ‘In reference to:’, used in **.LO RN**.
- Letns** is an array containing the different strings used in **.NS**. It is really a number of string variables prefixed with **Letns!**. If the argument doesn’t exist, it is included between () with **Letns!copy** as a prefix and **Letns!to** as a suffix. Observe the space after ‘Copy’ and before ‘to’.

<b>Name</b>	<b>Value</b>
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 ( <i>with trailing space</i> )
Letns!to	to ( <i>note leading space</i> )

**Letnsdef**

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

**MO1 – MO12**

Strings containing the month names ‘January’ through ‘December’.

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

**Rp** Contains the string ‘REFERENCES’.

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

<b>Value</b>	<b>Meaning</b>
co	Table of contents
fg	List of figures
tb	List of tables
ec	List of equations
ex	List of exhibits
ap	Appendix

**Tm** Contains the string ‘\ (tm’, the trade mark symbol.

**Verbnm**

Argument to **.nm** in the **.VERBON** command. Default is 1.

**Number variables used in nm**

**Aph** Print an appendix page for every new appendix if this number variable is non-zero. No output occurs if **Aph** is zero, but there is always an appendix entry in the ‘List of contents’.

**Cl** Contents level (in the range 0 to 14). The contents is saved if a heading level is lower than or equal to the value of **Cl**. Default is 2.

- Cp** Eject page between list of table, list of figure, etc., if the value of **Cp** is zero. Default is 0.
- D** Debug flag. Values greater than zero produce debug information of increasing verbosity. A value of 1 gives information about the progress of formatting. Default is 0.
- De** If set to 1, eject after floating display is output. Default is 0.
- Dsp** If defined, it controls the space output before and after static displays. Otherwise the value of **Lsp** is used.
- Df** Control floating keep output. This is a number in the range 0 to 5, with a default value of 5. See **.DF**.
- Ds** If set to 1, use the amount of space stored in register **Lsp** before and after display. Default is 1.
- Ej** If set to 1, eject page before each first-level heading. Default is 0.
- Eq** Equation labels are left-adjusted if set to 0 and right-adjusted if set to 1. Default is 0.
- Fs** Footnote spacing. Default is 1.

**H1 – H7**

Heading counters

- H1dot** Append a dot after the level-one heading number if value is greater than zero. Default is 1.
- H1h** A copy of number register **H1**, but it is incremented just before the page break. Useful in user-defined header macros.
- Hb** Heading break level. A number in the range 0 to 14, with a default value of 2. See **.H**.
- Hc** Heading centering level. A number in the range 0 to 14, with a default value value of 0. See **.H**.
- Hi** Heading temporary indent. A number in the range 0 to 2, with a default value of 1.
- 0 no indentation, left margin
  - 1 indent to the right, similar to **'P 1'**
  - 2 indent to line up with text part of preceding heading
- Hps** Heading pre-space level. If the heading level is less than or equal to **Hps**, two lines 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** Number of lines preceding **.H** if the heading level is greater than **Hps**. Value is in units, default is 0.5.
- Hps2** Number of lines preceding **.H** if the heading level is less than or equal to **Hps**. Value is in units, default is 1.
- Hs** Heading space level. A number in the range 0 to 14, with a default value of 2. See **.H**.
- Hss** Number of lines following **.H** if the heading level is less than or equal to **Hs**. Value is in units, default is 1.
- Ht** Heading numbering type.
- 0 multiple levels (1.1.1, 1.1.2, etc.)
  - 1 single level
- Default is 0.
- Hu** Unnumbered heading level. Default is 2.
- Hy** Hyphenation status of text body.
- 0 no hyphenation
  - 1 hyphenation on, set to value 6
- Default is 0.

- Iso** Set this variable to 1 on the command line to get an ISO-formatted date string (`-rIso=1`). Useless inside of a document.
- L** Page length, only for command-line settings.
- Letwam**  
Maximum lines in return-address, used in `.WA/.WE`. Default is 14.
- Lf, Lt, Lx, Le**  
Enable (1) or disable (0) the printing of List of figures, List of tables, List of exhibits and List of equations, respectively. Default values are `Lf=1`, `Lt=1`, `Lx=1`, and `Le=0`.
- Li** List indentation, used by `.AL`. Default is 6.
- Limsp** A flag controlling the insertion of space between prefix and mark in automatic lists (`.AL`).
- 0 no space
  - 1 emit space
- Ls** List space threshold. If current list level is greater than **Ls** no spacing occurs around lists. Default is 99.
- Lsp** The vertical space used by an empty line. The default is 0.5v in troff mode and 1v in nroff mode.
- N** Page numbering style.
- 0 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 style enabled.
  - 4 page header is removed on the first page.
  - 5 'section-page' and 'section-figure' numbering style enabled.
- Default is 0. See also the number registers **Sectf** and **Sectp**.
- Np** A flag to control whether paragraphs are numbered.
- 0 not numbered
  - 1 numbered in first-level headings.
- Default is 0.
- O** Page offset, only for command-line settings.
- Of** Format of figure, table, exhibit, and equation titles.
- 0 ". "
  - 1 " - "
- Default is 0.
- P** Current page-number, normally the same as '%' unless 'section-page' numbering style is enabled.
- Pi** Paragraph indentation. Default is 5.
- Pgps** A flag to control whether header and footer point size should follow the current settings or just change when the header and footer are defined.
- 0 Point size only changes to the current setting when `.PH`, `.PF`, `.OH`, `.EH`, `.OF`, or `.OE` is executed.
  - 1 Point size changes after every `.S`. This is the default.
- Ps** Paragraph spacing. Default is 1.
- Pt** Paragraph type.
- 0 left-justified
  - 1 indented paragraphs

2 indented paragraphs except after **.H**, **.DE**, or **.LE**.

Default is 0.

- Rpe** Set default value for second argument of **.RP**. Default is 0.
- Sectf** A flag controlling ‘section-figures’ numbering style. A non-zero value enables this. See also register **N**.
- Sectp** A flag controlling ‘section-page’ numbering style. A non-zero value enables this. See also register **N**.
- Si** Display indentation. Default is 5.
- Verbin** Indentation for **.VERBON**. Default is 5n.
- W** Line length, only for command-line settings.
- .mgm** Always 1.

## INTERNALS

The letter macros are 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** sets the number variables **Pt** and **Pi** to 0 and 5, respectively. 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 is called for all pages.

### **let@sg***\_type name title n flag [arg1 [arg2 [...]]]*

**.SG** is calling this macro only for letters; memorandums have its own processing. *name* and *title* are specified through **.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 the argument.

**.LO** is implemented as a general option-macro. It demands that a string named **Lettype** is defined, where *type* is the letter type. **.LO** then assigns the argument to the string variable **let\*lo-type**.

## FILES

*/usr/share/groff/1.22.4/tmac/m.tmac*

*/usr/share/groff/1.22.4/tmac/mm/\*.cov*

*/usr/share/groff/1.22.4/tmac/mm/\*.MT*

*/usr/share/groff/1.22.4/tmac/mm/locale*

## AUTHORS

The GNU version of the *mm* macro package was written by Jörgen Hägg (jh@axis.se) of Lund, Sweden.

## SEE ALSO

**groff(1)**, **troff(1)**, **tbl(1)**, **pic(1)**, **eqn(1)**  
**groff\_mmse(7)**