

NAME

groff_opmode – control GNU roff “pen state”

DESCRIPTION

Primarily intended to support the operation of **pdfroff(1)**, but also potentially useful in other, similar contexts, the **groff_opmode** supplementary macro package offers a mechanism for synchronization of the typesetter’s “pen-up” and “pen-down” states, as appropriate to particular phases of document production. In the case of **pdfroff(1)**, it is used to suppress the output of the document body content, when processing the phase for production of a table of contents, (when this is output *following* processing of the body content), and vice-versa.

USAGE

Typically loaded from within **groff(7)** document source, or by another dependent macro package, using the **groff(7)** request:

.mso opmode.tmac

the **groff_opmode** macro package provides the **.OP** macro, which exhibits the invocation syntax:

.OP [*<output-phase>* ...]

A call of the **.OP** macro should be placed at the beginning of any section of the **groff(7)** input file, for which the content is designated for processing in a different phase from that which has gone before; for example, when using the emulated classical table of contents relocation feature of **pdfroff(7)**, the call:

.OP 2

would be placed at the beginning of the document body content, (normally the beginning of the input file), indicating that the following content should be output during **pdfroff(7)**’s document body processing phase, whereas the complementary call:

.OP 1

would be placed after the document body content, to prepare for output of the collected table of contents, in **pdfroff(7)**’s table of contents processing phase.

The behaviour of the **.OP** macro is determined according to the setting of the **PHASE** register, which is expected to have been set, usually as a command line assignment for an individual invocation of **groff(1)**, depending on the particular phase of output file processing to which the invocation relates — **pdfroff(1)** sets it to *one* when initiating its table of contents output phase, and to *two* for its document body output phase. This behaviour may be described as follows:

- If the **PHASE** register is *not* defined, (as is the case during **pdfroff(1)**’s initial document analysis phase — which produces no physical output), **.OP** sets the **OPMODE** register to *one*, but has no effect on **groff(7)**’s “pen state”; this effectively indicates that **groff(7)** is expected to be operating in its default “pen-down” state, *without* actually enforcing this.
- When the **PHASE** register *is* defined, then each specified “*<output-phase>*” argument is compared, in turn, with **PHASE**, until one compares as equal, or no more remain; if an equal match is found, then the **OPMODE** register is set to *one*, and **groff(7)**’s “pen-down” state is activated; otherwise, when no equal match is found, the **OPMODE** register is set to *zero*, and **groff(7)**’s “pen-up” state is activated.
- If the **PHASE** register *is* defined, but no “*<output-phase>*” arguments have been specified, then the **OPMODE** register is *immediately* set to *one*, and **groff(7)**’s “pen-down” state is activated.

This provides a mechanism for tracking the anticipated **groff(7)** “pen state”, when performing multiple phase **groff(1)** document formatting, supporting modification of the formatter’s behaviour on the basis of the indicated “pen state”; for example, the **groff_pdfmark(7)** macros, and associated macro packages such as **groff_mspdf(7)**, may use the information conveyed by **OPMODE** to suppress generation of *pdfmark* code, while processing document sections in which the “pen-up” state has been activated.

CONTROL REGISTERS

The following two numeric registers are associated with the operation of the **.OP** macro:

PHASE

Defined by **pdfroff(1)**, this numeric register is assigned a value of *one*, when the processing objective is to produce a table of contents, and a value of *two*, for production of the document body content.

OPMODE

Assigned on execution of the **.OP** macro, this numeric register assumes a value of *zero*, when the typesetter is switched to its “*pen-up*” state, and a value of *one*, following a switch to the “*pen-down*” state.

When used in conjunction with the **groff_pdfmark(7)** macros, the **OPMODE** register is aliased to that package’s **PDFOPMODE** register.

FILES

/usr/local/share/groff/site-tmac/opmode.tmac

Provides the implementation of the **.OP** macro.

CAVEATS

Assignment of the **PHASE** register lies firmly within the purview of whatever process is responsible for driving multiple phase **groff(1)** document formatting. It should neither be defined, nor reassigned within any document source file; doing so may result in undefined behaviour.

AUTHORS

The **groff_opmode** supplementary macro implementation is provided by the *groff-pdfmark* package, which was written by Keith Marshall <keith.d.marshall@ntlworld.com>; this was formerly distributed with *GNU roff*, but it is now independently maintained at, and distributed from Keith’s *groff-pdfmark* project hosting web-site <<https://savannah.nongnu.org/projects/groff-pdfmark/>>, whence the latest version may *always* be obtained.

SEE ALSO

groff(1), **pdfroff(1)**, **groff(7)**, **groff_mspdf(7)**, **groff_pdfmark(7)**, **pdfroff(7)**

More comprehensive documentation on the use of **pdfroff(1)**, (support of which is the primary function of **groff_opmode**), and of the *groff-pdfmark* macro suite in general, may be found, in PDF format, in the reference guide “*Portable Document Format Publishing with GNU Troff*”, which has also been written by Keith Marshall; the most recently published version of this guide may be read online, by following the appropriate document reference link on the *groff-pdfmark* project hosting web-site <<https://savannah.nongnu.org/projects/groff-pdfmark/>>, whence a copy may also be downloaded.