

The `count1to` package^{*†}

Martin Schröder

Crüsemannallee 3

D-28213 Bremen

Martin.Schroeder@ACM.org

PGP-Key: 2048 bit / KeyID 292814E5
7E86 6EC8 97FA 2995 82C3 FEA5 2719 090E

1999/06/08

Abstract

This package sets `\count1` to `\count8` with the values of `page` to `subparagraph`. `\count9` is used to flag odd pages. The values of these counters are displayed and written in the `.dvi` file by `TEX` and can later be used to select the pages of certain parts of the document for printing if the device driver supports this.

It also gives access to the total number of pages of the document via the label `TotalPages`.

Contents

1	Introduction	1	4	The implementation	3
2	Options	3	5	Acknowledgements	5
3	Required packages	3			

1 Introduction

Most of the time users want to print only certain parts of a document; but these can only be selected by using the page numbers of these parts with most device drivers. This can be difficult or impossible if pages in different parts of the document have the same number—e. g. in the frontmatter and the first text pages (iii vs. 3).

`TEX` provides an easy solution to this problem: whenever a page is completed by the output routine and shipped out via `\shipout`, it displays the values of `\count0` to `\count9` on the display (e. g. [1]) *and writes them to the `.dvi` file*.

^{*}The version umber of this file is v1.22, last revised 1999/06/08.

The name `count1to` is a tribute to the 8 + 3 file-naming convention of certain “operating systems”; strictly speaking it should be `count1to9`.

[†]This document has 7 pages.

“The first ten `\count` registers, `\count0` to `\count9`, are reserved for a special purpose: `TEX` displays these ten counts on your terminal whenever outputting a page, and it transmits them to the output file as an identification of that page. The counts are separated by decimal points on your terminal, with trailing ‘.0’ patterns suppressed. Thus, for example, if `\count0=5` and `\count2=7` when a page is shipped out to the `dvi` file, and if the other counters are zero, `TEX` will type ‘[5.0.7]’. Plain `TEX` uses `\count0` for the page number, and it keeps `\count1` through `\count9` equal to zero; that is why you see ‘[1]’ when page 1 is being output. In more complex applications the page numbers can have further structure; ten counts are shipped out so that there will be plenty of identification.”[2, p. 119]

Surprisingly, until recently there existed no package for `LATEX` that used these `\counters` although some drivers allow the selection of pages based on other `\counts` than `\count0` (e. g. `emTEX`).

This package is the solution: It uses the `everyshi` package[3] to set `\count1` to `\count9` before each `\shipout` with these values:

<code>\count</code>	value
0	relative page number (set by <code>L^AT_EX</code>)
1	absolute page number
2	number of current <code>\part</code>
3	number of current <code>\chapter</code> (0 with article class)
4	number of current <code>\section</code>
5	number of current <code>\subsection</code>
6	number of current <code>\subsubsection</code>
7	number of current <code>\paragraph</code>
8	number of current <code>\subparagraph</code>
9	1 on odd pages, 0 on even pages ¹

`count1to` also works with classes that do not define some or all of the sectioning commands and their counters, like `letter`. Although it is of somewhat little use then :-).

A note for users of this package: When you select the pages of some part of your document with a lower structure than `\chapter`, remember that only `\parts` and `\chapters` start on a new page; if you want to print a complete `\section`, you should also select the first page of the next `\section`. Also note that `TEX` ships out the values of the counters instead of their visual representation (produced with `\thecounter`), so appendix A sets `\count4` to 1 in the article class.

A note for developers of device drivers: Please add support for `\count1` to 9 to your programs. It would also be nice if users could easily select the next page(s) after a certain count (something like “*.*.*.*.2+1.*” should be possible for selecting all pages with `\count4` \mapsto section = 2 plus the first page of section 3).

TotalPages

The setting of `\count1` with the absolute page number makes it possi-

¹If you have a better application for `\count9`, let me know.

ble to provide the *total* number of pages of the document (as opposed to the “number” on the last page provided by the `lastpage` package[1]). By referencing the label `TotalPages` (e.g. by `\ref{TotalPages}`) you get the total number of pages the document had at the last run of `LATEX`. If you want to use this feature, you should load `countlto` as the *last* package in your document because `countlto` executes some code at `\end{document}` and has to be sure that its code is the *last* code executed there.

2 Options

The package has no options.

3 Required packages

The package requires the `everyshi` package[3].

4 The implementation

```
1 (*package)
```

We need the `everyshi` package.

```
2 \RequirePackage{everyshi}[1994/12/09]
```

We need various `\ifs` to check if the used counters are defined.

```
3 \newif\if@have@part
```

```
4 \newif\if@have@chapter
```

```
5 \newif\if@have@section
```

```
6 \newif\if@have@subsection
```

```
7 \newif\if@have@subsubsection
```

```
8 \newif\if@have@paragraph
```

```
9 \newif\if@have@subparagraph
```

`\@countItoIX@ifs` `\@countItoIX@ifs` is used to set all these `\ifs` at `\begin{document}`. It is undefined after use to save some space.

```
10 \newcommand*{\@countItoIX@ifs}{
```

```
11   \ifundefined{c@part}           {}{\@have@parttrue}
```

```
12   \ifundefined{c@chapter}       {}{\@have@chaptertrue}
```

```
13   \ifundefined{c@section}       {}{\@have@sectiontrue}
```

```
14   \ifundefined{c@subsection}    {}{\@have@subsectiontrue}
```

```
15   \ifundefined{c@subsubsection} {}{\@have@subsubsectiontrue}
```

```
16   \ifundefined{c@paragraph}     {}{\@have@paragraphtrue}
```

```
17   \ifundefined{c@subparagraph}  {}{\@have@subparagraphtrue}
```

```
18 }
```

```
19 \AtBeginDocument{\@countItoIX@ifs\let\@countItoIX@ifs\undefined}
```

`\@countItoIX@bugfix` The current version of `LATEX` has the “feature” that only the first level of counters associated with a counter via the optional argument of `\newcounter` is reset when the counter is stepped; so when you start a new chapter, the number for the subsection is not reset. This is normally no problem, but with this package, it is: If this bug is not fixed or worked-around, then whenever you use `\subparagraph` or some other lower sectioning command, and after that a sectioning command that is at least two

levels higher (e.g. `\subsubsection`), the value of `subparagraph` would not be reset and would still be displayed and shipped out. So we have to associate *all* lower-level sectioning commands with the higher levels. This is done via `\@addtoreset` at `\begin{document}`. `\@countItoIX@bugfix` is undefined after use to save some space.

```

20 \newcommand*{\@countItoIX@bugfix}{
21   \if@have@part
22     \@addtoreset{section}      {part}
23     \@addtoreset{subsection}  {part}
24     \@addtoreset{subsubsection}{part}
25     \@addtoreset{paragraph}   {part}
26     \@addtoreset{subparagraph}{part}
27   \fi
28   \if@have@chapter
29     \@addtoreset{subsection}   {chapter}
30     \@addtoreset{subsubsection}{chapter}
31     \@addtoreset{paragraph}    {chapter}
32     \@addtoreset{subparagraph}{chapter}
33   \fi
34   \if@have@section
35     \@addtoreset{subsubsection}{section}
36     \@addtoreset{paragraph}    {section}
37     \@addtoreset{subparagraph}{section}
38   \fi
39   \if@have@subsection
40     \@addtoreset{paragraph}    {subsection}
41     \@addtoreset{subparagraph}{subsection}
42   \fi
43   \if@have@subsubsection
44     \@addtoreset{subparagraph}{subsubsection}
45   \fi
46 }
47 \AtBeginDocument{\@countItoIX@bugfix\let\@countItoIX@bugfix\undefined}

```

`\@countItoIX@shipout` `\@countItoIX@shipout` is used for setting `\count1` to `\count9` at each `\shipout`.

```

48 \newcommand*{\@countItoIX@shipout}{

```

`\count1` is the absolute page number, which we have to maintain by ourselves.

```

49   \global\advance \count1 by 1

```

`\count2` to `\count8` are set with the values of `\part` to `\subparagraph`, if these commands are defined.²

```

50   \if@have@part           \count2\value{part}           \fi
51   \if@have@chapter        \count3\value{chapter}        \fi
52   \if@have@section        \count4\value{section}        \fi
53   \if@have@subsection     \count5\value{subsection}     \fi
54   \if@have@subsubsection  \count6\value{subsubsection}  \fi
55   \if@have@paragraph      \count7\value{paragraph}      \fi
56   \if@have@subparagraph   \count8\value{subparagraph}   \fi

```

²We should probably also check the value of `secnumdepth`, but I don't think this is necessary.

\count9 is set to 1 on odd pages and to 0 on even pages.

```

57 \ifodd\count1\count9=1 \else\count9=0 \fi
58 }
59 \EveryShipout{\@countItoIX@shipout}

```

TotalPages The label **TotalPages** is set on the *absolut* last page of the document and holds the *total* number of pages of the document, i. e. the value of \count1. But before we write the value to the auxfile we call a \clearpage to force all pending floats to be output. To do this we have to be sure to be the *last* macro called by \AtEndDocument—but this is nearly impossible to assure (see the troubles the *lastpage* package[1] goes through to achieve this), so we try to do our very best and set our entry for \AtEndDocument at \begin{document} by using \AtBeginDocument.

```

60 \AtBeginDocument{%
61   \AtEndDocument{%
62     \clearpage
63     \immediate\write\@mainaux{\string\newlabel{TotalPages}{\the\count1}{\the\count1}}
64   }%
65 }
66 \end{package}

```

5 Acknowledgements

As usual Rebecca Stiels improved the quality of this documentation. Rolf Niepraschk (niepraschk@ptb.de) provided the idea and the first implementation of the **TotalPages** label.

References

- [1] Jeff Goldberg. The *lastpage*-package. CTAN:tex-archive/macros/latex/contrib/other/lastpage. L^AT_EX 2_ε package.
- [2] Donald E. Knuth. *The T_EXBook*, volume A of *Computers and Typesetting*. Addison-Wesley, 1986. Eleventh printing, revised, May 1991.
- [3] Martin Schröder. The *everyshi* package. CTAN:tex-archive/macros/latex/contrib/supported/ms/everyshi.dtx. L^AT_EX 2_ε package.

Index

Numbers written in *italic* refer to the page where the corresponding entry is described, the ones underlined to the code line of the definition, the rest to the code lines where the entry is used.

Symbols	
<code>\@addtoreset</code>	22–26, 29–32, 35–37, 40, 41, 44
<code>\@countItoIX@bugfix</code>	<u>20</u>
<code>\@countItoIX@ifs</code>	<u>10</u>
<code>\@countItoIX@shipout</code>	<u>48</u>
<code>\@have@chaptertrue</code>	12
<code>\@have@paragraphtrue</code>	16
<code>\@have@parttrue</code>	11
<code>\@have@sectiontrue</code>	13
<code>\@have@subparagraphtrue</code>	17
<code>\@have@subsectiontrue</code>	14
<code>\@have@subsubsectiontrue</code>	15
<code>\@ifundefined</code>	11–17
<code>\@mainaux</code>	63
A	
<code>\advance</code>	49
<code>\AtBeginDocument</code>	19, 47, 60
<code>\AtEndDocument</code>	61
C	
<code>\clearpage</code>	62
<code>\count</code>	49–57, 63
E	
<code>\else</code>	57
<code>\EveryShipout</code>	59
F	
<code>\fi</code>	27, 33, 38, 42, 45, 50–57
G	
<code>\global</code>	49
I	
<code>\if@have@chapter</code>	4, 28, 51
<code>\if@have@paragraph</code>	8, 55
<code>\if@have@part</code>	3, 21, 50
<code>\if@have@section</code>	5, 34, 52
<code>\if@have@subparagraph</code>	9, 56
<code>\if@have@subsection</code>	6, 39, 53
<code>\if@have@subsubsection</code>	7, 43, 54
<code>\ifodd</code>	57
<code>\immediate</code>	63
L	
<code>\let</code>	19, 47
N	
<code>\newcommand</code>	10, 20, 48
<code>\newif</code>	3–9
<code>\newlabel</code>	63
R	
<code>\RequirePackage</code>	2
S	
<code>\string</code>	63
T	
<code>\the</code>	63
<code>TotalPages</code>	2, <u>60</u>
U	
<code>\undefined</code>	19, 47
V	
<code>\value</code>	50–56
W	
<code>\write</code>	63

Change History

v1.00	after usage	4
General: New	<code>\@countItoIX@ifs</code> : destruct af-	
v1.01	ter usage	3
General: Documentation im-	General: Documentation im-	
proved	proved	1
v1.02	fixed typos	1
<code>\@countItoIX@bugfix</code> : destruct		

v1.03		use ifthen	4
General: Fixed use of <code>\newline</code>		<code>\@countItoIX@ifs</code> : don't use	
in title.	1	ifthen	3
v1.10		<code>\@countItoIX@shipout</code> : don't	
<code>\@countItoIX@bugfix: \ifthenelse</code>		use ifthen	4
.	4	General: <code>\newif</code> again	3
<code>\@countItoIX@ifs: \setboolean</code>		Documentation improved . . .	1
.	3	don't require ifthen	3
<code>\@countItoIX@shipout:</code>			
<code>\ifthenelse</code>	4	v1.21	
General: <code>\newboolean</code>	3	<code>TotalPages</code> : write to <code>\@mainaux</code>	
require ifthen	3	instead of <code>\@auxout</code>	5
<code>TotalPages</code> : new	5	v1.22	
v1.20		General: Moved to LPPL	1
<code>\@countItoIX@bugfix</code> : don't			