The **engord** package

Heiko Oberdiek

<oberdiek@uni-freiburg.de>

2006/02/20 v1.2

Abstract

The package generates the suffix of English ordinal numbers. It can be used with plain and LaTeX formats.

Contents

1 Usage

$\boxed{\texttt{\textbackslash engord}\,\{\langle \textit{LaTeX counter name}\rangle\}}$

It prints the value of the LaTeX counter as English ordinal number. It can be used in the same way as `\arabic`, `\roman`, or `\alph`. The command is not available in plain-TeX.

$\boxed{\texttt{\textbackslash engordnumber}\,\{\langle \textit{any TeX number}\rangle\}}$

It prints the number as English ordinal number.

$\boxed{\texttt{\textbackslash engordletters\{\#1\}}}$

This command formats the English ordinal letters after the number. It defaults to `\textsuperscript`.

> `\engorderror{#1}`

It can be redefined, if an other error handling is wanted. The argument is a negative number or zero.

> `\engordraisetrue`
> `\engordraisefalse`

These commands set the switch `\ifengordraise` that is asked by the default `\engordletters` before raising the ordinal letters.

1.1 Package options

normal: `\engordraisefalse`

raise: `\engordraisetrue`

Default is raise.

1.2 Examples

- `\usepackage[normal]{engord}`
  `\engordnumber{1}` → 1st
  `\engordnumber{12}` → 12th
  `\engordnumber{123}` → 123rd
  `\engord{page}` → 1st (if page has the value of one)
  `\engordraisetrue`
  `\engordnumber{12}` → 12th

- The default output of a counter can be redefined:

    `\newcounter{mycounter}`
    `\renewcommand{\theengcounter}{\engord{mycounter}}`

- Because the implementation of `\engord` and `\engordnumber` is kept expandable, these commands can be used to make command names with an appropriate definition of `\engordletters`:

    `\renewcommand*{\engordletters}[#1]{#1}`
    `\@namedef{My\engordnumber{3}Command}{...}`

  This generates the command name '`\My4rdCommand`'. Since version 1.2 the redefinition can be dropped if the letters are not raised.

- If the letters should not be raised, use LATEX package option **normal** or use

    `\engordraisefalse`

  Also `\engordletters` could be redefined for this purpose:

    `\renewcommand*{\engordletters}[1]{#1}`

2 Implementation

2.1 Reload check and identification

1 ⟨∗package⟩

Reload check, especially if the package is not used with LATEX.

2 `\begingroup`
3 `  \expandafter\let\expandafter\x\csname ver@engord.sty\endcsname`
4 `  \ifcase 0%`

```
 5    \ifx\x\relax % plain
 6    \else
 7      \ifx\x\empty % LaTeX
 8      \else
 9        1%
10      \fi
11    \fi
12  \else
13    \expandafter\ifx\csname PackageInfo\endcsname\relax
14      \def\x#1#2{%
15        \immediate\write-1{Package #1 Info: #2.}%
16      }%
17    \else
18      \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
19    \fi
20    \x{engord}{The package is already loaded}%
21    \endgroup
22    \expandafter\endinput
23  \fi
24 \endgroup
```

Package identification:

```
25 \begingroup
26   \expandafter\ifx\csname ProvidesPackage\endcsname\relax
27     \def\x#1#2#3[#4]{\endgroup
28       \immediate\write-1{Package: #3 #4}%
29       \xdef#1{#4}%
30     }%
31   \else
32     \def\x#1#2[#3]{\endgroup
33       #2[{#3}]%
34       \ifx#1\relax
35         \xdef#1{#3}%
36       \fi
37     }%
38   \fi
39 \expandafter\x\csname ver@engord.sty\endcsname
40 \ProvidesPackage{engord}%
41   [2006/02/20 v1.2 Provides the ifpdf switch (HO)]
```

2.2   Help commands for plain compatibility

\EO@atcode   Save catcode of @.

```
42 \expandafter\chardef\csname EO@atcode\endcsname\catcode`\@
43 \catcode`\@=11 %
```

\EO@def   Definitions, \newcommand does not exist in plain-TeX.

```
44 \begingroup\expandafter\expandafter\expandafter\endgroup
45 \expandafter\ifx\csname newcommand\endcsname\relax
46   \def\EO@def{\def}%
47 \else
48   \def\EO@def#1{%
49     \newcommand*{#1}{}%
50     \def#1%
51   }%
52 \fi
```

\EO@PackageWarning

```
53 \begingroup\expandafter\expandafter\expandafter\endgroup
54 \expandafter\ifx\csname PackageWarning\endcsname\relax
55   \def\EO@PackageWarning#1#2{%
56     \immediate\write16{%
57       Package #1 Warning: #2 on input line \the\inputlineno.%
```

```
58       }%
59     }%
60 \else
61   \def\EO@PackageWarning{\PackageWarning}%
62 \fi
```

2.3   User macros

\ifengordraise   The switch \ifengordraise, whether the ordinal letters are raised or not. Default is raised because of compatibility.

```
63 \newif\ifengordraise
64 \engordraisetrue
```

In LaTeX this also can be controlled by option normal or raise.

```
65 \begingroup\expandafter\expandafter\expandafter\endgroup
66 \expandafter\ifx\csname DeclareOption\endcsname\relax
67 \else
68   \DeclareOption{normal}{\engordraisefalse}%
69   \DeclareOption{raise}{\engordraisetrue}%
70   \ProcessOptions*\relax
71 \fi
```

\engordletters   \engordletters is called with one argument, the english ordinal letters, and contains the code to format them. It defaults to \textsuperscript depending on \ifengordraise.

```
72 \expandafter\ifx\csname engordletters\endcsname\relax
73   \EO@def\engordletters{%
74     \ifengordraise
75       \expandafter\engordtextsuperscript
76     \fi
77   }%
78 \fi
```

\engordtextsuperscript   For plain-TeX the definition is quite ugly, redefine \engordtextsuperscript if you have a better one.

```
79 \expandafter\ifx\csname engordtextsuperscript\endcsname\relax
80   \begingroup\expandafter\expandafter\expandafter\endgroup
81   \expandafter\ifx\csname textsuperscript\endcsname\relax
82     \def\engordtextsuperscript#1{%
83       \relax
84       \ifmmode
85         ^{\rm#1}%
86       \else
87         $^{\rm#1}$%
88       \fi
89     }%
90   \else
91     \def\engordtextsuperscript{\textsuperscript}%
92   \fi
93 \fi
```

\engorderror   \engorderror is called, if the number is zero or negative.

```
94 \expandafter\ifx\csname engorderror\endcsname\relax
95   \EO@def\engorderror#1{%
96     #1\engordletters{!ERROR!}%
97     \EO@PackageWarning{engord}{%
98       '#1' is not an ordinal number%
99     }%
100   }%
101 \fi
```

<dl>
<dt>\engord</dt>
<dd>

\engord expects a LaTeX counter name as argument and calls \engordnumber. It is defined only, if LaTeX is used.

```
102 \begingroup\expandafter\expandafter\expandafter\endgroup
103 \expandafter\ifx\csname newcounter\endcsname\relax
104 \else
105   \EO@def\engord#1{%
106     \engordnumber{\value{#1}}%
107   }%
108 \fi
```
</dd>

<dt>\engordnumber</dt>
<dd>

\engordnumber is the user command to print a number as english ordinal number. The argument can be any TeX number like explicit numbers, register values, . . .

In a safe way it converts the TeX number argument into a form that only consists of decimal digits.

```
109 \EO@def\engordnumber#1{%
110   \expandafter\EO@number\expandafter{\number#1}%
111 }
```
</dd>
</dl>

2.4 Suffix generation

<dl>
<dt>\EO@number</dt>
<dd>

\EO@number expects a number with decimal digits as argument and looks at the size of the number and the count of the digits:

```
112 \def\EO@number#1{%
113   \ifnum#1<1 % handle the error case
114     \engorderror{#1}%
115   \else
116     \ifnum#1<21
117       \EO@ord{#1}%
118     \else
119       \ifnum#1<100
120         \EO@twodigits#1%
121       \else
122         \@ReturnAfterFi{%
123           \EO@reverse#1\@nil{}\EO@afterreverse
124         }%
125       \fi
126     \fi
127   \fi
128 }
```
</dd>

<dt>\@ReturnAfterFi</dt>
<dd>

An internal help macro to prevent a too deep \if nesting.

```
129 \long\def\@ReturnAfterFi#1\fi{\fi#1}
```
</dd>

<dt>\EO@ord</dt>
<dd>

\EO@ord prints the number with ord letters.
#1: decimal digits, #1 < 21

```
130 \def\EO@ord#1{%
131   #1%
132   \expandafter\engordletters
133   \ifcase#1{th}\or
134     {st}\or
135     {nd}\or
136     {rd}\else
137     {th}%
138   \fi
139 }
```
</dd>

<dt>\EO@twodigits</dt>
<dd>

\EO@twodigits expects a number with two digits,
20 < number < 100

```
140 \def\EO@twodigits#1#2{%
141   #1\EO@ord{#2}%
142 }
```
</dd>
</dl>

\EO@reverse	\EO@reverse reverses the digits of the number.

#1: next digit
#2: rest of the digits
#3: already reversed digits
#4: next command to call with the reversed number as argument

```
143 \def\EO@reverse#1#2\@nil#3#4{%
144   \ifx\\#2\\%
145     #4{#1#3}%
146   \else
147     \@ReturnAfterFi{%
148       \EO@reverse#2\@nil{#1#3}{#4}%
149     }%
150   \fi
151 }
```

\EO@afterreverse	\EO@afterreverse calls \EO@reverseback so that \EO@reverseback can inspect the digits of the number.

```
152 \def\EO@afterreverse#1{%
153   \EO@reverseback#1\@nil
154 }
```

\EO@reverseback	\EO@reverseback reverses the reversion.

#1: the last digit of the number
#2: the second last digit of the number
#3: first digits of the number in reversed order, it is not empty, because \EO@reverseback is only called with numbers > 100.

```
155 \def\EO@reverseback#1#2#3\@nil{%
156   \EO@reverse#3\@nil{}\@firstofone
157   \ifnum#2#1<21
158     \EO@ord{#2#1}%
159   \else
160     #2\EO@ord{#1}%
161   \fi
162 }
```

Restore catcode of @.

```
163 \catcode`\@=\EO@atcode
```

```
164 ⟨/package⟩
```

3 Installation

CTAN. This package is available on CTAN[1]:

CTAN:macros/latex/contrib/oberdiek/engord.dtx The source file.

CTAN:macros/latex/contrib/oberdiek/engord.pdf Documentation.

Unpacking. The .dtx file is a self-extracting docstrip archive. The files are extracted by running the .dtx through plain-TeX:

```
tex engord.dtx
```

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as texmf tree):

```
engord.sty   →   tex/generic/oberdiek/engord.sty
engord.pdf   →   doc/latex/oberdiek/engord.pdf
engord.dtx   →   source/latex/oberdiek/engord.dtx
```

---

[1] ftp://ftp.ctan.org/tex-archive/

If you have a `docstrip.cfg` that configures and enables `docstrip`'s TDS installing feature, then some files can already be in the right place, see the documentation of `docstrip`.

Refresh file databases. If your TeX distribution (teTeX, mikTeX, . . .) rely on file databases, you must refresh these. For example, teTeX users run `texhash` or `mktexlsr`.

3.1 Some details for the interested

Attached source. The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is `pdftk`, e.g. unpack the file into the current directory:

```
pdftk engord.pdf unpack_files output .
```

Unpacking with LaTeX. The `.dtx` chooses its action depending on the format:

plain-TeX: Run `docstrip` and extract the files.

LaTeX: Generate the documentation.

If you insist on using LaTeX for `docstrip` (really, `docstrip` does not need LaTeX), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{engord.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfLaTeX:

```
pdflatex engord.dtx
makeindex -s gind.ist engord.idx
pdflatex engord.dtx
makeindex -s gind.ist engord.idx
pdflatex engord.dtx
```

4 History

[2000/05/23 v1.0]

- First public release.

[2003/04/28 v1.1]

- Bug fix for 30, 40, 50, . . . , 100, 130, . . .
- `\ordletters` renamed to documented `\engordletters`.

[2006/02/20 v1.2]

- Support for plain-TeX.
- Switch `\ifengordraise` added.
- Package options `raise` and `normal` added.
- DTX framework.

5 Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.