

# The **engord** package

Heiko Oberdiek  
<oberdiek@uni-freiburg.de>

2007/04/26 v1.4

## Abstract

The package generates the suffix of English ordinal numbers. It can be used with plain and L<sup>A</sup>T<sub>E</sub>X formats.

## Contents

<b>1</b>	<b>Usage</b>	<b>1</b>
1.1	Package options . . . . .	2
1.2	Examples . . . . .	2
<b>2</b>	<b>Implementation</b>	<b>3</b>
2.1	Reload check and identification . . . . .	3
2.2	Help commands for plain compatibility . . . . .	4
2.3	User macros . . . . .	4
2.4	Suffix generation . . . . .	5
<b>3</b>	<b>Installation</b>	<b>7</b>
3.1	Download . . . . .	7
3.2	Bundle installation . . . . .	7
3.3	Package installation . . . . .	7
3.4	Refresh file name databases . . . . .	7
3.5	Some details for the interested . . . . .	8
<b>4</b>	<b>History</b>	<b>8</b>
	[2000/05/23 v1.0] . . . . .	8
	[2003/04/28 v1.1] . . . . .	8
	[2006/02/20 v1.2] . . . . .	8
	[2007/04/11 v1.3] . . . . .	8
	[2007/04/26 v1.4] . . . . .	8
<b>5</b>	<b>Index</b>	<b>9</b>

## 1 Usage

`\engord{<LATEX counter name>}`

It prints the value of the L<sup>A</sup>T<sub>E</sub>X 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-T<sub>E</sub>X.

`\engordnumber{<any TEX number>}`

It prints the number as English ordinal number.

`\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}` → 12<sup>th</sup>

- 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 L<sup>A</sup>T<sub>E</sub>X 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 L<sup>A</sup>T<sub>E</sub>X.

```
2 \begingroup
3   \catcode44 12 % ,
4   \catcode45 12 % -
5   \catcode46 12 % .
6   \catcode58 12 % :
7   \catcode64 11 % @
8   \expandafter\let\expandafter\x\csname ver@engord.sty\endcsname
9   \ifcase 0%
10    \ifx\x\relax % plain
11    \else
12      \ifx\x\empty % LaTeX
13      \else
14        1%
15      \fi
16    \fi
17  \else
18    \expandafter\ifx\csname PackageInfo\endcsname\relax
19      \def\x#1#2{%
20        \immediate\write-1{Package #1 Info: #2.}%
21      }%
22    \else
23      \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
24    \fi
25    \x{engord}{The package is already loaded}%
26  \endgroup
27  \expandafter\endinput
28  \fi
29 \endgroup
```

Package identification:

```
30 \begingroup
31   \catcode40 12 % (
32   \catcode41 12 % )
33   \catcode44 12 % ,
34   \catcode45 12 % -
35   \catcode46 12 % .
36   \catcode47 12 % /
37   \catcode58 12 % :
38   \catcode64 11 % @
39   \expandafter\ifx\csname ProvidesPackage\endcsname\relax
40     \def\x#1#2#3[#4]{\endgroup
41       \immediate\write-1{Package: #3 #4}%
42       \xdef#1{#4}%
43     }%
44   \else
45     \def\x#1#2[#3]{\endgroup
46       #2[#3]%
47       \ifx#1\relax
48         \xdef#1{#3}%
49       \fi
50     }%
51   \fi
52   \expandafter\x\csname ver@engord.sty\endcsname
53   \ProvidesPackage{engord}%
54   [2007/04/26 v1.4 Provides the ifpdf switch (H0)]
```

## 2.2 Help commands for plain compatibility

`\EO@atcode` Save catcode of @.

```

55 \expandafter\chardef\csname EO@atcode\endcsname\catcode'\@
56 \catcode'\@=11 %

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

57 \begingroup\expandafter\expandafter\expandafter\endgroup
58 \expandafter\ifx\csname newcommand\endcsname\relax
59   \def\EO@def{\def}%
60 \else
61   \def\EO@def#1{%
62     \newcommand*{#1}{}%
63     \def#1%
64   }%
65 \fi

66 \begingroup\expandafter\expandafter\expandafter\endgroup
67 \expandafter\ifx\csname RequirePackage\endcsname\relax
68   \input infwarerr.sty\relax
69 \else
70   \RequirePackage{infwarerr}%
71 \fi
```

## 2.3 User macros

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

```

72 \newif\ifengordraise
73 \engordraisetrue
```

In L<sup>A</sup>T<sub>E</sub>X this also can be controlled by option `normal` or `raise`.

```

74 \begingroup\expandafter\expandafter\expandafter\endgroup
75 \expandafter\ifx\csname DeclareOption\endcsname\relax
76 \else
77   \DeclareOption{normal}{\engordraisefalse}%
78   \DeclareOption{raise}{\engordraisetrue}%
79   \ProcessOptions*\relax
80 \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`.

```

81 \expandafter\ifx\csname engordletters\endcsname\relax
82   \EO@def\engordletters{%
83     \ifengordraise
84       \expandafter\engordtextsuperscript
85     \fi
86   }%
87 \fi
```

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

```

88 \expandafter\ifx\csname engordtextsuperscript\endcsname\relax
89   \begingroup\expandafter\expandafter\expandafter\endgroup
90   \expandafter\ifx\csname textsuperscript\endcsname\relax
91     \def\engordtextsuperscript#1{%
92       \relax
93       \ifmode
94         ~{\rm#1}%
95       \else
```

```

96      $\rm#1}$%
97      \fi
98    }%
99    \else
100      \def\engordtextsuperscript{\textsuperscript}%
101      \fi
102    \fi

\engorderror \engorderror is called, if the number is zero or negative.
103 \expandafter\ifx\csname engorderror\endcsname\relax
104   \EO@def\engorderror#1{%
105     #1\engordletters{!ERROR!}%
106     \@PackageWarning{engord}{%
107       ‘#1’ is not an ordinal number%
108     }%
109   }%
110   \fi

\engord \engord expects a LATEX counter name as argument and calls \engordnumber. It
is defined only, if LATEX is used.
111 \begingroup\expandafter\expandafter\expandafter\endgroup
112 \expandafter\ifx\csname newcounter\endcsname\relax
113 \else
114   \EO@def\engord#1{%
115     \engordnumber{\value{#1}}%
116   }%
117   \fi

\engordnumber \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.
118 \EO@def\engordnumber#1{%
119   \expandafter\EO@number\expandafter{\number#1}%
120 }
```

## 2.4 Suffix generation

```

\EO@number \EO@number expects a number with decimal digits as argument and looks at the
size of the number and the count of the digits:
121 \def\EO@number#1{%
122   \ifnum#1<1 % handle the error case
123     \engorderror{#1}%
124   \else
125     \ifnum#1<21 %
126       \EO@ord{#1}%
127     \else
128       \ifnum#1<100 %
129         \EO@twodigits#1%
130       \else
131         \@ReturnAfterFi{%
132           \EO@reverse#1\@nil}\EO@afterreverse
133         }%
134       \fi
135     \fi
136   \fi
137 }

\@ReturnAfterFi An internal help macro to prevent a too deep \if nesting.
138 \long\def\@ReturnAfterFi#1\fi{\fi#1}
```

```

\EO@ord \EO@ord prints the number with ord letters.
#1: decimal digits, #1 < 21
139 \def\EO@ord#1{%
140   #1%
141   \expandafter\engordletters
142   \ifcase#1{th}\or
143     {st}\or
144     {nd}\or
145     {rd}\else
146     {th}%
147   \fi
148 }

\EO@twodigits \EO@twodigits expects a number with two digits,
20 < number < 100
149 \def\EO@twodigits#1#2{%
150   #1\EO@ord{#2}%
151 }

\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
152 \def\EO@reverse#1#2\@nil#3#4{%
153   \ifx\#2\%
154     #4{#1#3}%
155   \else
156     \@ReturnAfterFi{%
157       \EO@reverse#2\@nil{#1#3}{#4}%
158     }%
159   \fi
160 }

\EO@afterreverse \EO@afterreverse calls \EO@reverseback so that \EO@reverseback can inspect
the digits of the number.
161 \def\EO@afterreverse#1{%
162   \EO@reverseback#1\@nil
163 }

\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.
164 \def\EO@reverseback#1#2#3\@nil{%
165   \EO@reverse#3\@nil{}\@firstofone
166   \ifnum#2#1<21 %
167     \EO@ord{#2#1}%
168   \else
169     #2\EO@ord{#1}%
170   \fi
171 }

Restore catcode of @.
172 \catcode'\@=\EO@atcode
173 </package>

```

## 3 Installation

### 3.1 Download

**Package.** This package is available on CTAN<sup>1</sup>:

[CTAN:macros/latex/contrib/oberdiek/engord.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/engord.pdf](#) Documentation.

**Bundle.** All the packages of the bundle ‘oberdiek’ are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

[CTAN:macros/latex/contrib/oberdiek/oberdiek-tds.zip](#)

*TDS* refers to the standard “A Directory Structure for  $\text{\TeX}$  Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

### 3.2 Bundle installation

**Unpacking.** Unpack the `oberdiek-tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek-tds.zip -d ~/texmf
```

**Script installation.** Check the directory `TDS:scripts/oberdiek/` for scripts that need further installation steps. Package `attachfile2` comes with the Perl script `pdfatfi.pl` that should be installed in such a way that it can be called as `pdfatfi`. Example (linux):

```
chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/
```

### 3.3 Package installation

**Unpacking.** The `.dtx` file is a self-extracting `docstrip` archive. The files are extracted by running the `.dtx` through plain- $\text{\TeX}$ :

```
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
```

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`.

### 3.4 Refresh file name databases

If your  $\text{\TeX}$  distribution (te $\text{\TeX}$ , mi $\text{\TeX}$ , ...) relies on file name databases, you must refresh these. For example, te $\text{\TeX}$  users run `texhash` or `mktextlsr`.

---

<sup>1</sup><http://ftp.ctan.org/tex-archive/>

### 3.5 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 L<sup>A</sup>T<sub>E</sub>X.** The `.dtx` chooses its action depending on the format:

**plain-T<sub>E</sub>X:** Run `docstrip` and extract the files.

**L<sup>A</sup>T<sub>E</sub>X:** Generate the documentation.

If you insist on using L<sup>A</sup>T<sub>E</sub>X for `docstrip` (really, `docstrip` does not need L<sup>A</sup>T<sub>E</sub>X), 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 pdfL<sup>A</sup>T<sub>E</sub>X:

```
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-T<sub>E</sub>X.
- Switch `\ifengordraise` added.
- Package options `raise` and `normal` added.
- DTX framework.

[2007/04/11 v1.3]

- Line ends sanitized.

[2007/04/26 v1.4]

- Use of package `infwarerr`.



## 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.

Symbols	
\@ . . . . .	55, 56, 172
\@PackageWarning . . . . .	106
\@ReturnAfterFi . . . . .	131, <u>138</u> , 156
\@firstofone . . . . .	165
\@nil . . . . .	132, 152, 157, 162, 164, 165
\\ . . . . .	153
C	
\catcode . . . . .	3, 4, 5, 6, 7, 31, 32, 33, 34, 35, 36, 37, 38, 55, 56, 172
\chardef . . . . .	55
\csname . . . . .	8, 18, 39, 52, 55, 58, 67, 75, 81, 88, 90, 103, 112
D	
\DeclareOption . . . . .	77, 78
E	
\empty . . . . .	12
\endcsname . . . . .	8, 18, 39, 52, 55, 58, 67, 75, 81, 88, 90, 103, 112
\endinput . . . . .	27
\engord . . . . .	1, <u>111</u>
\engorderror . . . . .	2, <u>103</u> , 123
\engordletters . . . . .	2, <u>81</u> , 105, 141
\engordnumber . . . . .	1, 115, <u>118</u>
\engordraisefalse . . . . .	2, 77
\engordraisetrue . . . . .	2, 73, 78
\engordtextsuperscript . . . . .	84, <u>88</u>
\EO@afterreverse . . . . .	132, <u>161</u>
\EO@atcode . . . . .	<u>55</u> , 172
\EO@def . . . . .	<u>57</u> , 82, 104, 114, 118
\EO@number . . . . .	119, <u>121</u>
\EO@ord . . . . .	126, <u>139</u> , 150, 167, 169
\EO@reverse . . . . .	132, <u>152</u> , 165
\EO@reverseback . . . . .	162, <u>164</u>
\EO@twodigits . . . . .	129, <u>149</u>
I	
\ifcase . . . . .	9, 142
\ifengordraise . . . . .	<u>72</u> , 83
\ifmmode . . . . .	93
\ifnum . . . . .	122, 125, 128, 166
\ifx . . . . .	10, 12, 18, 39, 47, 58, 67, 75, 81, 88, 90, 103, 112, 153
\immediate . . . . .	20, 41
\input . . . . .	68
N	
\newcommand . . . . .	62
\newif . . . . .	72
\number . . . . .	119
P	
\PackageInfo . . . . .	23
\ProcessOptions . . . . .	79
\ProvidesPackage . . . . .	53
R	
\RequirePackage . . . . .	70
\rm . . . . .	94, 96
T	
\textsuperscript . . . . .	100
V	
\value . . . . .	115
W	
\write . . . . .	20, 41
X	
\x . . . . .	8, 10, 12, 19, 23, 25, 40, 45, 52