

# 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 . . . . .	3
2.3	User macros . . . . .	4
2.4	Suffix generation . . . . .	5
<b>3</b>	<b>Installation</b>	<b>6</b>
3.1	Download . . . . .	6
3.2	Bundle installation . . . . .	7
3.3	Package installation . . . . .	7
3.4	Refresh file name databases . . . . .	7
3.5	Some details for the interested . . . . .	7
<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>8</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   \catcode44 12 % ,
32   \catcode45 12 % -
33   \catcode46 12 % .
34   \catcode58 12 % :
35   \catcode64 11 % @
36   \expandafter\ifx\csname ProvidesPackage\endcsname\relax
37     \def\x#1#2#3[#4]{\endgroup
38       \immediate\write-1{Package: #3 #4}%
39       \xdef#1{#4}%
40     }%
41   \else
42     \def\x#1#2[#3]{\endgroup
43       #2[#{#3}]%
44       \ifx#1\relax
45         \xdef#1{#3}%
46       \fi
47     }%
48   \fi
49   \expandafter\x\csname ver@engord.sty\endcsname
50   \ProvidesPackage{engord}%
51   [2007/04/26 v1.4 Provides the ifpdf switch (H0)]
```

### 2.2 Help commands for plain compatibility

\EO@atcode Save catcode of @.

```
52 \expandafter\chardef\csname EO@atcode\endcsname\catcode'\@
53 \catcode'\@=11 %
```

```

\EO@def Definitions, \newcommand does not exist in plain-TeX.
54 \begingroup\expandafter\expandafter\expandafter\endgroup
55 \expandafter\ifx\csname newcommand\endcsname\relax
56   \def\EO@def{\def}%
57 \else
58   \def\EO@def#1{%
59     \newcommand*{#1}{}%
60     \def#1%
61   }%
62 \fi

63 \begingroup\expandafter\expandafter\expandafter\endgroup
64 \expandafter\ifx\csname RequirePackage\endcsname\relax
65   \input infwarerr.sty\relax
66 \else
67   \RequirePackage{infwarerr}%
68 \fi

```

## 2.3 User macros

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

```

69 \newif\ifengordraise
70 \engordraisetrue

```

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

```

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

```

78 \expandafter\ifx\csname engordletters\endcsname\relax
79   \EO@def\engordletters{%
80     \ifengordraise
81       \expandafter\engordtextsuperscript
82     \fi
83   }%
84 \fi

```

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

```

85 \expandafter\ifx\csname engordtextsuperscript\endcsname\relax
86   \begingroup\expandafter\expandafter\expandafter\endgroup
87   \expandafter\ifx\csname textsuperscript\endcsname\relax
88     \def\engordtextsuperscript#1{%
89       \relax
90       \ifmmode
91         ~{\rm#1}%
92       \else
93         ${\rm#1}$%
94       \fi
95     }%
96   \else
97     \def\engordtextsuperscript{\textsuperscript}%
98   \fi
99 \fi

```

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

```

100 \expandafter\ifx\csname engorderror\endcsname\relax
101   \EO@def\engorderror#1{%
102     #1\engordletters{!ERROR!}%
103     \@PackageWarning{engord}{%
104       ‘#1’ is not an ordinal number%
105     }%
106   }%
107 \fi

```

`\engord` `\engord` expects a L<sup>A</sup>T<sub>E</sub>X counter name as argument and calls `\engordnumber`. It is defined only, if L<sup>A</sup>T<sub>E</sub>X is used.

```

108 \begingroup\expandafter\expandafter\expandafter\endgroup
109 \expandafter\ifx\csname newcounter\endcsname\relax
110 \else
111   \EO@def\engord#1{%
112     \engordnumber{\value{#1}}%
113   }%
114 \fi

```

`\engordnumber` `\engordnumber` is the user command to print a number as english ordinal number. The argument can be any T<sub>E</sub>X number like explicit numbers, register values, ...

In a safe way it converts the T<sub>E</sub>X number argument into a form that only consists of decimal digits.

```

115 \EO@def\engordnumber#1{%
116   \expandafter\EO@number\expandafter{\number#1}%
117 }

```

## 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:

```

118 \def\EO@number#1{%
119   \ifnum#1<1 % handle the error case
120     \engorderror{#1}%
121   \else
122     \ifnum#1<21 %
123       \EO@ord{#1}%
124     \else
125       \ifnum#1<100 %
126         \EO@twodigits#1%
127       \else
128         \@ReturnAfterFi{%
129           \EO@reverse#1\@nil}\EO@afterreverse
130       }%
131     \fi
132   \fi
133 \fi
134 }

```

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

```

135 \long\def\@ReturnAfterFi#1\fi{\fi#1}

```

`\EO@ord` `\EO@ord` prints the number with ord letters.

#1: decimal digits, #1 < 21

```

136 \def\EO@ord#1{%
137   #1%
138   \expandafter\engordletters
139   \ifcase#1{th}\or
140     {st}\or
141     {nd}\or

```

```

142     {rd}\else
143     {th}%
144 \fi
145 }

\EO@twodigits \EO@twodigits expects a number with two digits,
20 < number < 100
146 \def\EO@twodigits#1#2{%
147   #1\EO@ord{#2}%
148 }

\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
149 \def\EO@reverse#1#2\@nil#3#4{%
150   \ifx\#2\%
151     #4{#1#3}%
152   \else
153     \@ReturnAfterFi{%
154       \EO@reverse#2\@nil{#1#3}{#4}%
155     }%
156   \fi
157 }

\EO@afterreverse \EO@afterreverse calls \EO@reverseback so that \EO@reverseback can inspect
the digits of the number.
158 \def\EO@afterreverse#1{%
159   \EO@reverseback#1\@nil
160 }

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

Restore catcode of @.
169 \catcode'\@=\EO@atcode
170 </package>

```

## 3 Installation

### 3.1 Download

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

[CTAN:macros/latex/contrib/oberdiek/engord.dtx](http://ftp.ctan.org/tex-archive/macros/latex/contrib/oberdiek/engord.dtx) The source file.

[CTAN:macros/latex/contrib/oberdiek/engord.pdf](http://ftp.ctan.org/tex-archive/macros/latex/contrib/oberdiek/engord.pdf) Documentation.

---

<sup>1</sup>[ftp://ftp.ctan.org/tex-archive/](http://ftp.ctan.org/tex-archive/)

**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

## 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-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 TeX distribution (TeX, MikTeX, ...) relies on file name databases, you must refresh these. For example, TeX users run texhash or mktexlsr.

## 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-TeX:** 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- $\text{\TeX}$ .
- 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		6, 7, 31, 32, 33, 34, 35, 52, 53, 169	
\@	52, 53, 169	\chardef	52
\@PackageWarning	103	\csname	8, 18, 36, 49,
\@ReturnAfterFi	128, <u>135</u> , 153		52, 55, 64, 72, 78, 85, 87, 100, 109
\@firstofone	162		
\@nil	129, 149, 154, 159, 161, 162		
\\	150		
		D	
		\DeclareOption	74, 75
C		E	
\catcode	3, 4, 5,	\empty	12



\endcsname	8, 18, 36, 49, 52, 55, 64, 72, 78, 85, 87, 100, 109	\input	65
\endinput	27	<b>N</b>	
\engord	1, 108	\newcommand	59
\engorderror	2, 100, 120	\newif	69
\engordletters	2, 78, 102, 138	\number	116
\engordnumber	1, 112, 115	<b>P</b>	
\engordraisefalse	2, 74	\PackageInfo	23
\engordraisetrue	2, 70, 75	\ProcessOptions	76
\engordtextsuperscript	81, 85	\ProvidesPackage	50
\EO@afterreverse	129, 158	<b>R</b>	
\EO@atcode	52, 169	\RequirePackage	67
\EO@def	54, 79, 101, 111, 115	\rm	91, 93
\EO@number	116, 118	<b>T</b>	
\EO@ord	123, 136, 147, 164, 166	\textsuperscript	97
\EO@reverse	129, 149, 162	<b>V</b>	
\EO@reverseback	159, 161	\value	112
\EO@twodigits	126, 146	<b>W</b>	
<b>I</b>		\write	20, 38
\ifcase	9, 139	<b>X</b>	
\ifengordraise	69, 80	\x	8, 10, 12, 19, 23, 25, 37, 42, 49
\ifmmode	90		
\ifnum	119, 122, 125, 163		
\ifx	10, 12, 18, 36, 44, 55, 64, 72, 78, 85, 87, 100, 109, 150		
\immediate	20, 38		