

unitsdef – Typesetting units with L^AT_EX 2 _{ε} ^{*}

PATRICK HAPPEL[†]

2005/01/04

Abstract

There are a lot of packages for typesetting units in L^AT_EX 2 _{ε} . Some define macros to typeset a lot of units but do not suit to the actual font settings, some make the characters needed available but do not predefined any unit.

This package tries to comply with both requirements. It predefines common units, defines an easy to use interface to define new units and changes the output concerning to the surrounding font settings.

Contents

1 Packages loaded by <code>unitsdef</code>	3	5.1 Defining new units	7
		5.2 Typesetting μ , Ω , $^{\circ}$ and $^{\circ}\text{C}$	8
2 General usage	3	6 Local configuration file	10
3 Options	3	7 Predefined units and prefices	10
3.1 Options related to gen- symb	3	8 To Do	15
3.2 More options	4	9 Code	15
4 Typesetting units	5	9.1 switches	15
4.1 Inline units	5	9.2 Options	17
4.2 Typesetting angles and arcs	6	9.3 Loading packages	18
4.3 Logical markup for units	6	9.4 The interface	18
5 The interface	7	9.5 Definition of prefices	24
		9.6 Definitions of units	25
		9.7 Loading abbreviations	31

^{*}Version v0.2

[†]patrick.happel@rub.de

Changes

v0.1		
	General: First public release	1
v0.11		
	General: Added electronvolt	1
v0.12		
	General: Fixed some bugs relating to <code>gensymb</code>	1
v0.13		
	General: Some bugfixes and some units added.	1
v0.13a		
	General: More bugfixes. <code>\utimes</code> is now called <code>\unittimes</code>	1
v0.2		
	<code>\newnosepunit</code> : Added <code>\newnosepunit</code> and <code>\renewnosepunit</code> for units without any space be- tween value and unitsign . . .	19
	<code>\newunit</code> : <code>\newunit</code> now uses <code>\DeclareRobustCommand</code> in-	
	stead of <code>\newcommand</code>	18
	Added code for the use of <code>xspace</code>	18
	Now <code>unitsdef</code> uses <code>\unitvaluesep</code> instead of <code>\,</code> , thereby the in- terface is more flexible and allows the users to commit changes.	19
	<code>\renewunit</code> : Added code for the use of <code>xspace</code> . <code>\DeclareRobustCommand</code> is used instead of <code>\renewcommand</code>	19
	<code>\SI</code> : Added <code>\SI</code> to typeset units and values obeying the SI- rules.	24
	<code>\unitSIdef</code> : Added <code>\unitSIdef</code> to define units typeset by <code>\SI</code> . 24	
	<code>\unitsignonly</code> : Now defined by <code>\DeclareRobustCommand</code> . . .	18

Thanks

I have to thank some people who helped me by delivering suggestions, bug-reports, feature request, providing general ideas or anything else: FRANK

HEIMANN, STEPHAN MANN, FRANK
KÜSTER, STEFAN SALEWSKI,
STEPHAN HENNIG, KLAUS WACKER,
CHRISTOPH LÖSER

1 Packages loaded by `unitsdef`

`unitsdef` loads (and so needs) the following packages with the given options:

- [T1]fontenc
- textcomp
- amsmath
- units
- xspace (as long as `noxspace`, see section 3.2, isn't specified)

The first two packages are needed for different characters in the encodings set by these packages, `amsmath` provides the very useful `\text` macro and `units` is used for typesetting the units. `xpsace` is included to obtain the facility to write `5\mm` and `some text` instead of `5\mm\ and some text`.

2 General usage

To use `unitsdef` simply specify

```
\usepackage[<options>]{unitsdef}
```

in the preamble of your document.

3 Options

3.1 Options related to `gensymb`

`unitsdef` knows three options to avoid conflicts with the `gensymb`-package that also predefines macros for typesetting » μ «, » Ω « and » $^{\circ}\text{C}$ «. Some macros are defined both by `gensymb` and `unitsdef`. To solve this problem three different options exist:

OHM makes `unitsdef` use uppercase macros (`\Ohm`, `\Celsius`, `\Micro` and `\Degree`). The predefined units with prefixes still appear lowercase (`\kiloohm`).

ohm is the opposite to **OHM** and is the default setting.

redef-gensymb This option copies `\ohm`, `\celsius`, `\micro` and `\degree` from `gensymb` to `\gensymbohm`, `\gensymbcelsius`, `\gensymbmicro` and `\gensymbdegree`. `unitsdef` then redefines this four commands so that they have the same meaning as without `gensymb`. **Important:** To use this option `gensymb` has to be loaded *before* `unitsdef`!

3.2 More options

`unitsdef` knows some other options to change its behaviour:

LITER Makes `unitsdef` use the *uppercase* L as the sign produced by `\liter`.
This is default behaviour.

liter Makes `unitsdef` use the *lowercase* l as the sign produced by `\liter`.

noxspace Maybe some problems occur due to the `xspace` functionality of `unitsdef`. This option forces `unitsdef` to do without `xspace`.

noconfig Tells `unitsdef` not to look for a local `unitsdef` configuration file `unitsdef.cfg` (see section 6 for details).

noabbrv Is an abbreviation for *no abbreviations*. `unitsdef` does not define any short commands for a unit. This might be helpfull if another package defines a command with the same name and L^AT_EX exits with an error.

ugly Applies the option `ugly` to `units`, thus all units containing fraction will be typeset like `m/s` in text mode and `$\frac{m}{s}$` in mathmode.

skipping only particular abbreviations Usually only one macro causes conflict between a definition made by `unitsdef` and another package. There are some options to avoid loading only particular abbreviations to avoid the definition that causes the conflict:

noamperageabbr Avoids defining the abbreviations for units of amperage (`\pA`, `\nA`, `\micA`, `\mA`, `\kA`).

nofrequencyabbr Avoids defining `\kHz`, `\MHz` and `\GHz`.

nomolabbr Avoids defining `\fmol`, `\pmol`, `\nmol`, `\micmol` and `\mmol`.

novoltageabbr Avoids defining `\kV` and `\mv`.

novolumeabbr Avoids defining `\fL`, `\pL`, `\nL`, `\micL`, `\mL`, `\cL` and `\dL`.

noweighthabbr Avoids defining `\kg`, `\fg`, `\pg`, `\nanog`, `\micg` and `\mg`.

noenergyabbr Avoids defining `\kJ`, `\eV`, `\meV`, `\keV`, `\MeV`, `\GeV` and `\TeV`.

nolengthabbr Avoids defining `\picom`, `\nm`, `\micm`, `\mm`, `\cm`, `\dm` and `\km`.

notimeabbr Avoids defining `\sek`, `\fs`, `\ps`, `\ns`, `\mics` and `\ms`.

4 Typesetting units

`unitsdef` includes a large set of predefined units. They are listed in section 7. These units, as well as units defined by you with `\newunit` (see below), can be used as follows:

```
\langle unitmacro \rangle [\langle value \rangle]
```

As you see, the value is an optional argument. Thus you can write `5\kg` (instead of `\kg[5]`), this may simplify the writing process. The distance in both cases is `\unitvaluesep`.

If the optional argument is not used you still can write `5\kg and some text` due to the `xspace` functionality of `unitsdef`.

`\unitvaluesep` If you want another distance between value and unit you simply have to redefine `\unitvaluesep`:

```
\renewcommand{\unitvaluesep}{\hspace*{\langle distance \rangle}}
```

`\unitsignonly` If you want to typeset the sign of a unit without any value (for introducing a new unit-sign f.e.) `\unitvaluesep` must not be set. This is provided by the macro `\unitsignonly{\<unitmacro>}`.

Important! If you want to specify a unit with a prefix, never use something like `\milli\hour`. This will lead to problems. Always define the units first using `\newunit` or use the command `\ilu` to typeset an unit without defining it first!

4.1 Inline units

`\ilu` Some packages for typesetting units provide the possibility to write things like `\micro\meter` to obtain μm . With `unitsdef` this will cause some problems due to the toggling of some internal switches. If you want to use prefixes without defining a new unit you have to use the command `\ilu`. It has one optional and one mandatory argument. The first contains the value to be typeset, the

latter contains the unit:

```
A force of  
\ilu[17]{\micro\newton}  
is applied.  
$F=17\ilu{\micro\newton}$  
\textsf{17\ilu{\micro\newton}}
```

A force of $17 \mu\text{N}$ is applied. $F =$
 $17 \mu\text{N}$ $17 \mu\text{N}$

4.2 Typesetting angles and arcs

If you type $5\degree 10'\ 12''$ you will get the following result:
» $5^\circ 10' 12''$ «. There are full spaces between the single values and the preceding unit-symbol. This is due to the `xspace` functionality of `unitsdef`. The correct output you will obtain by using the macro `\arc`¹. Using it in this way `\arc{5;10;12}` will lead to » $5^\circ 10' 12''$ «. There are some more features of this macro, best understood if you look at this example:

```
\begin{enumerate}  
  \item \arc{1}      % 1  
  \item \arc{1;2}    % 2  
  \item \arc{1;2;3}  % 3  
  \item \arc{;2;3}   % 4  
  \item \arc{;;3}    % 5  
  \item \arc{1;;2}   % 6  
  \item \arc{1;;}    % 7  
  \item \arc{;;;}    % 8  
\end{enumerate}
```

1. 1°
2. $1^\circ 2'$
3. $1^\circ 2' 3''$
4. $0^\circ 2' 3''$
5. $0^\circ 0' 3''$
6. $1^\circ 0' 2''$
7. 1°
- 8.

4.3 Logical markup for units

One advantage of `unitsdef` is that it typesets the units suiting to the surrounding font settings. But some people (magazines, universities, ...) want units always to be typeset in a special way. There is a checklist available at <http://physics.nist.gov/cuu/Units/checklist.html>. `unitsdef` does not fit to all these requirements by default. But there are some macros for figuring out the output. First, there is the command `\SI` which typesets (by default) all units in upright shape, but doesn't change anything else according to the font settings. The command `\SI` has two mandatory arguments, the

¹It is expected to have (nearly?) the same usage and results as `\ang` from the `S!style` package.

first has to contain the value, the latter contains the unit:

```
\textbf{\textit{A force of  
SI{17}{\newton}  
was applied.}}
```

A force of 17 N was applied.

If you want to customize the output of `\SI`, you can redefine the macro `\unitSIdf`. By default it contains `\upright`, but if you want units to be typeset upright in serif family, you can achieve this by redefining it to `\upshape\rmfamily`:

```
\renewcommand{\unitSIdf}{  
  \upshape\rmfamily  
  \textsf{\textbf{\textit{  
    A force of  
    SI{17}{\newton}  
    was applied.}}}}
```

A force of 17 N was applied.

5 The interface

5.1 Defining new units

`\newunit` Besides the predefined unit-macros (see section 7) `unitsdef` offers the possibility to define new units. This is done with the macro `\newunit`:

```
\newunit{\<unitmacro>}{\<unitsign>}
```

To typeset a unit *Newton* (represented by `N`) you have to define² it as follows:

```
\newunit{\newton}{N}
```

After this you can use the macro `\newton`:

```
5\newton plus \newton[3] is 8\newton.
```

This will give the following output:

5 N plus 3 N is 8 N.

To define a unit *millinewton* (`mN`) simply define:

```
\newunit{\millinewton}{\milli\newton}
```

<code>\renewunit</code>	For redefining units, the macro <code>\renewunit</code> can be used with the same syntax as <code>\newunit</code> (on the lines of <code>\newcommand</code> and <code>\renewcommand</code>).
<code>\newnosepunit</code>	As an additional facility to define new units the macro <code>\newnosepunit</code> exists with the same syntax as <code>\newunit</code> . It allows you to define a unit without any space between the value and the unit-symbol. The only unit I know typeset without leading space is <i>degree</i> . You typeset 5° instead of 5° . But maybe there are other units that must not have leading space, so I included this command in the user interface.
<code>\renewnosepunit</code>	If you want to redefine a unit to be typeset without leading space, use <code>\renewnosepunit</code> .
<code>\per</code>	To typeset units containing a fraction the macro <code>\per</code> is defined. The usage is:
	<code>\per{\langle numerator \rangle}{\langle denominator \rangle}</code>
	To define <i>newton per squaremeter</i> you have to say:
	<code>\newunit{\newtonpersmeter}{\per{\newton}{\squaremeter}}</code>
<code>\unittimes</code>	To typeset a multiplication sign between two units the macro <code>\unittimes</code> is used:
	<code>\newunit{\newtonmeter}{\newton\unittimes\meter}</code> <code>\newunit{\newtonmeterpersec}{\per{\newton\unittimes\meter}{\second}}</code>
<code>\unitsep</code>	To typeset additional space (instead of <code>\unittimes</code>) between two units you can use <code>\unitsep</code> (do NOT use <code>\,</code>):
	<code>\newunit{\newtonmeter}{\newton\unitsep\meter}</code>
	The default is <code>\,</code> , but if you want different amount of space you can simply <code>\renewcommand{\unitsep}{\langle distance \rangle}</code> .
<code>\unitsuperscript</code>	To typeset superscripts use <code>\unitsuperscript</code> :
	<code>\newunit{\squaremeter}{\meter\unitsuperscript{2}}</code>

5.2 Typesetting μ , Ω , $^\circ$ and $^\circ\text{C}$

`\setTextOmega` Typesetting units would be quite easy without *Ohm* and *micro* (*degree* and *de-*

`\setMathOmega` ²You do not really have to define *newton*, it is already defined by `unitsdef`. This is just an example.

`\setTextmu`

`\setMathmu`

`\setTextCelsius` 8

`\setMathCelsius`

`\setMathDegree`

`\setTextDegree`

gree Celsius as well). The representing characters μ and Ω have to exist in three variants: One for use in math-mode (suiting to `\mathrm`) the latter two for use in `\rmfamily` and `\sffamily`. Most fonts have a suiting μ but very few fonts own a Ω . The fonts from BITSTREAM (delivered with some versions of COREL DRAW) have an italic non-suiting μ , so you have to take care of a μ when using such fonts, too. Also the μ in Latin Modern fonts isn't very good looking yet (but LM is still in beta-state). Because of nearly infinite combinations of different fonts it is impossible to choose the correct characters for μ , Ω and $^{\circ}\text{C}$ automatically. To define these characters use the macros `\setTextOmega`, `\setMathOmega`, `\setTextmu`, `\setMathmu`, `\setTextCelsius`, `\setMathCelsius`, `\setMathDegree` and `\setTextDegree`:

```
\setTextOmega{\langle serif-definition \rangle}{\langle sans-serif-definition \rangle}
\setMathOmega{\langle definition \rangle}

\setTextmu{\langle serif-definition \rangle}{\langle sans-serif-definition \rangle}
\setMathmu{\langle definition \rangle}

\setTextCelsius{\langle serif-definition \rangle}{\langle sans-serif-definition \rangle}
\setMathCelsius{\langle definition \rangle}

\setTextDegree{\langle serif-definition \rangle}{\langle sans-serif-definition \rangle}
\setMathDegree{\langle definition \rangle}
```

Important! The macros concerning typesetting in math-mode use `amsamth`'s `\text`-macro. So you have to switch to math-mode to use a character out of the math alphabet. The reason for this is `units`. `units` uses `\mathrm` to typeset in math-mode. But only few mathfonts have a suitable Ω , for example:

```
\documentclass{article}

\usepackage[T1]{fontenc}
\usepackage{textcomp}
\usepackage{mathpazo}

\begin{document}
$ \Omega \quad \mathrm{\Omega} $
\end{document}
```

`unitsdef` sets the defaults in a way suitable to the CM-family:

```
\setMathOmega{$\mathrm{\Omega}$}
\setMathmu{\textmu}
\setTextOmega{\textohm}{\textohm}
```

```
\setTextmu{\textmu}{\textmu}
\setMathCelsius{\textcelsius}
\setTextCelsius{\textcelsius}{\textcelsius}
\setMathDegree{\textdegree}
\setTextDegree{\textdegree}{\textdegree}
```

To find a suitable Ω for different fonts use WALTER SCHMIDT's `gensymb`-package³ and read the documentation of this package.

6 Local configuration file

Since version 0.2 `unitsdef` by default looks for a file `unitsdef.cfg` and inputs its contents. This is useful if you frequently use a similar set of units in your documents. You put your definitions in `unitsdef.cfg` and they are included automatically in your document⁴

A configuration file may look like this:

```
\ProvidesFile{unitsdef.cfg}%
[2004/12/09 v1.0 some definitions for unitsdef]

\newunit{\molar}{\per{\mole}{\liter}}
\newunit{\millimolar}{\per{\millimole}{\liter}}

\endinput
```

If, by some reason, you don't want your local config file to be loaded specify the option `noconfig` in the `\usepackage` command.

7 Predefined units and prefixes

Table 1: predefined prefixes

<i>name</i>	<i>prefix</i>	<i>macro</i>	<i>value</i>	<i>name</i>	<i>prefix</i>	<i>macro</i>	<i>value</i>
yocto	y	\yocto	10^{-24}	zepto	z	\zepto	10^{-21}

³CTAN:macros/latex/contrib/was/

⁴Keep in mind that TeX has to find the file, so follow the instructions given by your TeX-distribution. On *nix-like systems and a TDS compliant distribution it might be possible to put `unitsdef.cfg` in `~/texmf/tex/latex/unitsdef/config/` and update the filename database.

Table 1: predefined prefixes

<i>name</i>	<i>prefix</i>	<i>macro</i>	<i>value</i>	<i>name</i>	<i>prefix</i>	<i>macro</i>	<i>value</i>
atto	a	\attono	10^{-18}	femto	f	\femtono	10^{-15}
pico	p	\picono	10^{-12}	nano	n	\nanono	10^{-9}
micro	μ	\microno	10^{-6}	milli	m	\millino	10^{-3}
centi	c	\centino	10^{-2}	deci	d	\decino	10^{-1}
deca	da	\decano	10^{+1}	hecto	h	\hectono	10^{+2}
kilo	k	\kilono	10^{+3}	mega	M	\megano	10^{+6}
giga	G	\giganono	10^{+9}	tera	T	\terano	10^{+12}
peta	P	\petano	10^{+15}	exa	E	\exano	10^{+18}
zetta	Z	\zettano	10^{+21}	yotta	Y	\yottano	10^{+24}

Table 2: predefined units

<i>name</i>	<i>sign</i>	<i>macro</i>	<i>alias</i>
<i>base units</i>			
meter	m	\meter	
kilogram	kg	\kilogram	\kg
mole	mol	\mole	
second	s	\second	\sek
ampere	A	\ampere	
K	\kelvin		
candela	cd	\candela	
<i>units of length</i>			
picometer	pm	\picometer	\picom
nanometer	nm	\nanometer	\nm
micrometer	μm	\micrometer	\micm
millimeter	mm	\millimeter	\mm
centimeter	cm	\centimeter	\cm
decimeter	dm	\decimeter	\dm
kilometer	km	\kilometer	\km
<i>units of weight</i>			
gram	g	\gram	
femtogram	fg	\femtogram	\fg
picogram	pg	\picogram	\pg

Table 2: predefined units

<i>name</i>	<i>sign</i>	<i>macro</i>	<i>alias</i>
nanogram	ng	\nanogram	\nanog
microgram	μg	\microgram	\micg
milligram	mg	\milligram	\mg
<i>units of amount of substance</i>			
femtomole	fmol	\femtomole	\fmol
picomole	pmol	\picomole	\pmol
nanomole	nmol	\nanomole	\nmol
micromole	μmol	\micromole	\micmol
millimole	mmol	\millimole	\mmol
<i>units of time</i>			
attosecond	as	\attosecond	
femtosecond	fs	\femtosecond	\fs
pikosecond	ps	\picosecond	\ps
nanosecond	ns	\nanosecond	\ns
microsecond	μs	\microsecond	\mics
millisecond	ms	\millisecond	\ms
<i>units of amperage</i>			
picoampere	pA	\picoampere	\pA
nanoampere	nA	\nanoampere	\nA
microampere	μA	\microampere	\micA
milliampere	mA	\milliampere	\mA
kiloampere	kA	\kiloampere	\kA
<i>units of volume</i>			
liter	L	\liter	
femtoliter	fL	\femtoliter	\fl
picoliter	pL	\picoliter	\pl
nanoliter	nL	\nanoliter	\nl
microliter	μL	\microliter	\micl
milliliter	mL	\milliliter	\ml
centiliter	cL	\centiliter	\cl
deciliter	dL	\deciliter	\dl
hectoliter	hL	\hectoliter	\hl
cubicmeter	m^3	\cubicmeter	
cubicmicrometer	μm^3	\cubicmicrometer	
cubicmillimeter	mm^3	\cubicmillimeter	

Table 2: predefined units

<i>name</i>	<i>sign</i>	<i>macro</i>	<i>alias</i>
<i>units of area</i>			
squaremeter	m^2	\squaremeter	
ar	a	\ar	
Hektar	ha	\hektar	
squarecentimeter	cm^2	\squarecentimeter	
squaremillimeter	mm^2	\squaremillimeter	
squarekilometer	km^2	\squarekilometer	
<i>more units of weight</i>			
ton	t	\ton	
<i>derived units of electricity</i>			
volt	V	\volt	
millivolt	mV	\millivolt	\mV
kilovolt	kV	\kilovolt	\kv
Watt	W	\watt	
milliwatt	mW	\milliwatt	
kilowatt	kW	\kilowatt	
megawatt	MW	\megawatt	
coulomb	C	\coulomb	
ohm	Ω	\ohm or \Ohm	
kiloohm	$k\Omega$	\kiloohm	
megaohm	$M\Omega$	\megaohm	
gigaohm	$G\Omega$	\gigaohm	
siemens	S	\siemens	
millisiemens	mS	\millisiemens	
farad	F	\farad	
femtofarad	fF	\femtofarad	
picofarad	pF	\picofarad	
nanofarad	nF	\nanofarad	
microfarad	μF	\microfarad	
millifarad	mF	\millifarad	
<i>units of energy</i>			
joule	J	\joule	
millijoule	mJ	\millijoule	
kilojoule	kJ	\kilojoule	\kJ
megajoule	MJ	\megajoule	

Table 2: predefined units

<i>name</i>	<i>sign</i>	<i>macro</i>	<i>alias</i>
calorie	cal	\calory	
kilocalorie	kcal	\kilocalory	
electronvolt	eV	\electronvolt	\eV
millielectronvolt	meV	\millielectronvolt	\meV
kiloelectronvolt	keV	\kiloelectronvolt	\keV
megaelectronvolt	MeV	\megaelectronvolt	\MeV
gigaelectronvolt	GeV	\gigaelectronvolt	\GeV
teraelectronvolt	TeV	\teraelectronvolt	\TeV
<i>more units of time</i>			
minute	min	\minute	
hour	h	\hour	
days	d	\days	
<i>units of temperature</i>			
degree Celsius	°C	\celsius or \Celsius	
<i>units of angle</i>			
radian	rad	\radian	
steradian	sr	\steradian	
degree	°	\degree or \Degree	
arc minute	'	\arcmin	
arc second	"	\arcsec	
<i>units of frequencies</i>			
hertz	Hz	\hertz	
kilohertz	kHz	\kilohertz	\kHz
megahertz	MHz	\megahertz	\MHz
gigahertz	GHz	\gigahertz	\GHz
<i>units of force</i>			
newton	N	\newton	
millinewton	mN	\millinewton	
kilonewton	kN	\kilonewton	
<i>units of pressure</i>			
pascal	Pa	\pascal	
hectopascal	hPa	\hectopascal	
bar	bar	\uBar	
millibar	mbar	\millibar	

Table 2: predefined units

<i>name</i>	<i>sign</i>	<i>macro</i>	<i>alias</i>
<i>units of magnetism</i>			
weber	Wb	\weber	
tesla	T	\tesla	
henry	H	\henry	
<i>units of light</i>			
lumen	lm	\lumen	
lux	lx	\lux	
<i>units of radioactivity</i>			
becquerel	Bq	\becquerel	
megabecquerel	MBq	\megabecquerel	
curie	Cu	\curie	
sievert	Sv	\sievert	
millisievert	mSv	\millisievert	
<i>percent</i>			
percent	%	\percent	

8 To Do

There are a lot of things to be done. Some are mentioned here:

- Add some more macros to figure out the ouptut (something similar to \SI). Provide the possibility to typeset all units in math mode.
- Improve the documentation.
- Adding interfaces units for Å, % and °F

9 Code

1 <*package>

9.1 switches

\if@setunitsep This switch is used internally to decide, whether a distance is to be typesetted or not.
 2 \newif\if@setunitsep
 3 \@@setunitseptrue

\ifunit@@ohm This switch decides whether \ohm or \Ohm and \celsius or \Celsius is defined (according to gensymb).

4 \newif\ifunit@@ohm%
5 \unit@@ohmfalse

\ifunit@@redefgensymb This switch makes gensymb's macros \ohm and \celsius available as \gensymbohm and \gensymbcelsius. unitsdef uses \ohm and \celsius.

6 \newif\ifunit@@redefgensymb
7 \unit@@redefgensymbfalse

\ifunit@@liter This switch decides whether the output of the unit \iter is typeset as l or L

8 \newif\ifunit@@liter
9 \unit@@literfalse

\ifunit@@xspace This switch decides whether xspace is used.

10 \newif\ifunit@@xspace
11 \unit@@xspacetru

\ifunit@@xspace This switch decides whether abbreviations are loaded.

12 \newif\ifunit@@useabbrv
13 \unit@@useabbrvtrue

The following switches decide whether to load a .cfg file containing some abbreviations or not.

14 \newif\ifunit@@useampabbrv
15 \unit@@useampabbrvtrue
16 \newif\ifunit@@usefreqabbrv
17 \unit@@usefreqabbrvtrue
18 \newif\ifunit@@usemolabbrv
19 \unit@@usemolabbrvtrue
20 \newif\ifunit@@usevoltabbrv
21 \unit@@usevoltabbrvtrue
22 \newif\ifunit@@usevolabbrv
23 \unit@@usevolabbrvtrue
24 \newif\ifunit@@useweightabbrv
25 \unit@@useweightabbrvtrue
26 \newif\ifunit@@useenergyabbrv
27 \unit@@useenergyabbrvtrue
28 \newif\ifunit@@uselengthabbrv
29 \unit@@uselengthabbrvtrue
30 \newif\ifunit@@usetimeabbrv
31 \unit@@usetimeabbrvtrue

```
\ifunit@@useconfigfile This switch stores whether the option noconfig is given.
```

```
32 \newif\ifunit@@useconfigfile  
33 \unit@@useconfigfiletrue
```

9.2 Options

```
34 \DeclareOption{OHM}{\unit@@Ohmtrue}  
35 \DeclareOption{ohm}{\unit@@Ohmfalsel}  
36 \DeclareOption{redef-gensymb}{%  
37   \@ifpackageloaded{gensymb}{%  
38     \unit@@redefgensymbtrue%  
39     \AtBeginDocument{  
40       \let\gensymbohm\ohm%  
41       \let\gensymbcelsius\celsius%  
42       \let\gensymbmicro\micro%  
43       \renewunit{\ohm}{\unitOmega}%  
44       \renewunit{\celsius}{\unitCelsius}%  
45       \renewcommand{\micro}{\unitmu\@setunitsepfalse}%  
46     }  
47   }%  
48   \PackageError{unitsdef}{  
49     You requestet me to save some macros from the\MessageBreak  
50     gensymb-package. This package is not loaded.\MessageBreak  
51     If you load it later, tell me to use uppercase\MessageBreak  
52     macronames where conflicts appear by giving me\MessageBreak  
53     the option OHM.  
54   }  
55 }  
56 }  
57 \DeclareOption{liter}{\unit@@litertrue}  
58 \DeclareOption{LITER}{\unit@@literfalse}  
59 \DeclareOption{noxspace}{\unit@@xspacefalse}  
60 \DeclareOption{noabbrv}{\unit@@useabbrvfalse}  
61 \DeclareOption{ugly}{\PassOptionsToPackage{ugly}{units}}  
62 \DeclareOption{noampageabbr}{\unit@@useampabbrvfalse}  
63 \DeclareOption{nofreqencyabbr}{\unit@@usefreqabbrvfalse}  
64 \DeclareOption{nomolabbr}{\unit@@usemolabbrvfalse}  
65 \DeclareOption{novoltageabbr}{\unit@@usevoltabbrvfalse}  
66 \DeclareOption{novolumeabbr}{\unit@@usevolabbrvfalse}  
67 \DeclareOption{noweighthabbr}{\unit@@useweightabbrvfalse}  
68 \DeclareOption{noenergyabbr}{\unit@@useenergyabbrvfalse}  
69 \DeclareOption{nolengthabbr}{\unit@@uselengthabbrvfalse}  
70 \DeclareOption{notimeabbr}{\unit@@usetimeabbrvfalse}  
71 \DeclareOption{noconfig}{\unit@@useconfigfilefalse}
```

```

72 \ProcessOptions

73 \RequirePackage[T1]{fontenc}
74 \RequirePackage{amsmath}
75 \RequirePackage{textcomp}
76 \RequirePackage{units}
77 \ifunit@xspace
78   \RequirePackage{xspace}%
79   \let\unit@xspace\xspace%
80 \else
81   \let\unit@xspace\relax
82 \fi

```

9.4 The interface

\unitsignonly This macro is used to typeset a unit without leading spacing. To achieve this \@@setunitsep is set to **false**. \unitsignonly has an mandatory argument containig the unit to typeset: \unitsignonly{\(unit\)}.

```

83 \DeclareRobustCommand{\unitsignonly}[1]{%
84   \@@setunitsepfalse%
85   \begingroup%
86   \let\unit@xspace\relax%
87   #1%
88   \endgroup}

```

\unitvaluesep This Macro specifies the distance between value and unit, default value is \,.
 89 \newcommand{\unitvaluesep}{}
 90 \let\unitvaluesep\,%

\newunit This macro is the interface to define new units. Usage is:

```
\newunit{\(unitmacro\)}{\(unitsign\)}.
```

```
91 \newcommand{\newunit}[2]{%
```

First there is a check whether the macro already exists:

```
92   \newcommand{#1}{}%
```

then its definition is performed.

```
93   \DeclareRobustCommand{#1}[1][]{##1}%
```

The redefinition of \unitvaluesep has to stay local:

```
94   \begingroup%
```

\xpsace must not do anything when invoked inside a unit. So it is set to \relax inside this group.

```

95      \let\unit@@xspace\relax%
96      \if@setunitsep%
97          \unitvaluesep%

```

Once a spacing is typeset there must no further spacing be typeset. To avoid typesetting more spacings I redefine \unitvaluesep to \relax.

```

98      \let\unitvaluesep\relax%
99      \fi%
100     \unit{\#2}\global\@setunitseptrue%
101     \endgroup%

```

Now \xspace has to be invoked, as long as `noxspace` is not set.

```

102     \unit@@xspace%
103     }%
104 }

```

`\renewunit` This macro is to redefine existing units.

```

105 \newcommand{\renewunit}[2]{%
106     \renewcommand{\#1}{\%}
107     \DeclareRobustCommand{\#1}[1][]{\#\#1%
108         \begingroup%
109         \let\unit@@xspace\relax%
110         \if@setunitsep%
111             \unitvaluesep%
112             \let,\relax%
113         \fi%
114         \unit{\#2}\global\@setunitseptrue%
115         \endgroup%
116         \unit@@xspace%
117     }%
118 }
119

```

`\newnosepunit`

```

120 \newcommand{\newnosepunit}[2]{%
121     \newcommand{\#1}{\%}
122     \DeclareRobustCommand{\#1}[1][]{\#\#1%
123         \begingroup%
124         \let\xspace\relax%
125         \if@setunitsep%
126             \let,\relax%
127         \fi%

```

```

128      \unit{\#2}\global\@@setunitseptrue%
129      \endgroup%
130      \unit@xspace%
131  }%
132 }
133 \newcommand{\renewnosepunit}[2]{%
134   \renewcommand{\#1}{\%}
135   \DeclareRobustCommand{\#1}[1][]{\#\#1%
136   \begingroup%
137   \let\unit@xspace\relax%
138   \if@@setunitsep%
139   \let,\relax%
140   \fi%
141   \unit{\#2}\global\@@setunitseptrue%
142   \endgroup%
143   \unit@xspace%
144 }%
145 }

\per
146 \newcommand{\per}[2]{%
147   \@@setunitsepfalse%
148   \unitfrac{\#1}{\#2}%
149 }

\ilu The command \ilu provides the possibility to typeset inline-units that are not
defined by a previous \newunit command.
150 \newcommand{\ilu}[2][]{%
151   \begingroup%
152   \@@setunitsepfalse%
153   \let\unit@xspace\relax%
154   #1,\unit{\#2}%
155   \endgroup%
156 }

\unittimes
157 \newcommand{\unittimes}{\@@setunitsepfalse\ensuremath{\cdot} }

\unitsep
158 \let\unitsep\relax

\unitsuperscript
159 \newcommand{\unitsuperscript}[1]{%
160   \ifmmode^{\#1}\else\textsuperscript{\#1}\fi%
161 }

```

```

\unitMathOmega
162 \newcommand{\unitMathOmega} {}

\unitTextOmega
163 \newcommand{\unitTextOmega} {}

\unittextmu
164 \newcommand{\unitTextmu} {}

\unitmathmu
165 \newcommand{\unitMathmu} {}

\unitMathCelsius
166 \newcommand{\unitMathCelsius} {}

\unitTextCelsius
167 \newcommand{\unitTextCelsius} {}

\unitTextDegree
168 \newcommand{\unitTextDegree} {}

\unitMathDegree
169 \newcommand{\unitMathDegree} {}

\unitCelsius
170 \newcommand{\unitCelsius}{%
171   \ifmmode\unitMathCelsius\else\unitTextCelsius\fi%
172 }

\unitmu
173 \newcommand{\unitmu}{%
174   \ifmmode\unitMathmu\else\unitTextmu\fi%
175 }

\unitOmega
176 \newcommand{\unitOmega}{%
177   \ifmmode\unitMathOmega\else\unitTextOmega\fi%
178 }

\unitDegree
179 \newcommand{\unitDegree}{%
180   \ifmmode\unitMathDegree\else\unitTextDegree\fi%
181 }

```

```

\setMathOmega
182 \newcommand{\setMathOmega}[1]{%
183   \renewcommand{\unitMathOmega}{\text{#1}}%
184 }
185 \setMathOmega{$\mathrm{\Omega}$}

\setMathmu
186 \newcommand{\setMathmu}[1]{%
187   \renewcommand{\unitMathmu}{\text{#1}}%
188 }
189 \setMathmu{\textmu}

\setMathCelsius
190 \newcommand{\setMathCelsius}[1]{%
191   \renewcommand{\unitMathCelsius}{\text{#1}}%
192 }
193 \setMathCelsius{\textcelsius}

\setMathDegree
194 \newcommand{\setMathDegree}[1]{%
195   \renewcommand{\unitMathDegree}{\text{#1}}%
196 }
197 \setMathDegree{\textdegree}

\setTextOmega
198 \newcommand{\setTextOmega}[2]{%
199   \renewcommand{\unitTextOmega}{%
200     \begingroup%
201       \edef\@tempa{\sfdefault}%
202       \ifx\f@family\@tempa%
203         #2%
204       \else%
205         #1%
206       \fi%
207     \endgroup%
208   }%
209 }
210 \setTextOmega{\textohm}{\textohm}

\setTextmu
211 \newcommand{\setTextmu}[2]{%

```

```

212 \renewcommand{\unitTextmu}{%
213   \begingroup%
214     \edef\@tempa{\sfdefault}%
215     \ifx\f@family\@tempa%
216       #2%
217     \else%
218       #1%
219     \fi%
220   \endgroup%
221 }%
222 }%
223 \setTextmu{\textmu}{\textmu}

```

\setTextCelsius This macro is to define the »°C« that is used in text mode. The first argument is used when `\rmfamily` is active, the latter when `\sffamily` is active.

```

224 \newcommand{\setTextCelsius}[2]{%
225   \renewcommand{\unitTextCelsius}{%
226     \begingroup%
227       \edef\@tempa{\sfdefault}%
228       \ifx\f@family\@tempa%
229         #2%
230       \else%
231         #1%
232       \fi%
233     \endgroup%
234 }%
235 }%
236 \setTextCelsius{\textcelsius}{\textcelsius}

```

\setTextDegree This macro is to define the »°« that is used in text mode. The first argument is used when `\rmfamily` is active, the latter when `\sffamily` is active.

```

237 \newcommand{\setTextDegree}[2]{%
238   \renewcommand{\unitTextDegree}{%
239     \begingroup%
240       \edef\@tempa{\sfdefault}%
241       \ifx\f@family\@tempa%
242         #2%
243       \else%
244         #1%
245       \fi%
246     \endgroup%
247 }%
248 }%
249 \setTextDegree{\textdegree}{\textdegree}

```

```

\unitSIdef
250 \newcommand{\unitSIdef}{\upshape}

\SI
251 \newcommand{\SI}={}
252 \DeclareRobustCommand{\SI}[2]{%
253   \begingroup%
254     \let\unit@xspace\relax%
255     \unitSIdef\selectfont%
256     #1#2%
257   \endgroup%
258 }

```

9.5 Definition of prefixes

259 \newcommand{\yocto}{y\@@setunitsepfalse}	% -24
260 \newcommand{\zepto}{z\@@setunitsepfalse}	% -21
261 \newcommand{\atto}{a\@@setunitsepfalse}	% -18
262 \newcommand{\femto}{f\@@setunitsepfalse}	% -15
263 \newcommand{\pico}{p\@@setunitsepfalse}	% -12
264 \newcommand{\nano}{n\@@setunitsepfalse}	% -9
265 \ifunit@@Ohm	
266 \newcommand{\Micro}{\unitmu\@@setunitsepfalse}	
267 \let\@unit@micro\Micro	
268 \else	
269 \ifunit@@redefgensymb\else	
270 \newcommand{\micro}{\unitmu\@@setunitsepfalse}	
271 \let\@unit@micro\micro	
272 \fi	
273 \fi	
274 \newcommand{\milli}{m\@@setunitsepfalse}	% -3
275 \newcommand{\centi}{c\@@setunitsepfalse}	% -2
276 \newcommand{\deci}{d\@@setunitsepfalse}	% -1
277	
278 \newcommand{\deca}{da\@@setunitsepfalse}	% +1
279 \newcommand{\hecto}{h\@@setunitsepfalse}	% +2
280	
281 \newcommand{\kilo}{k\@@setunitsepfalse}	% +3
282 \newcommand{\mega}{M\@@setunitsepfalse}	% +6
283 \newcommand{\giga}{G\@@setunitsepfalse}	% +9
284 \newcommand{\tera}{T\@@setunitsepfalse}	% +12
285 \newcommand{\peta}{P\@@setunitsepfalse}	% +15
286 \newcommand{\exa}{E\@@setunitsepfalse}	% +18
287 \newcommand{\zetta}{Z\@@setunitsepfalse}	% +21

```
288 \newcommand{\yotta}{Y\@@setunitsepfalse} % +24
289
```

9.6 Definitions of units

9.6.1 base units

```
290 \newunit{\meter}{m}
291     \newunit{\gram}{g}
292 \newunit{\kilogram}{\kilo\gram}
293 \newunit{\mole}{mol}
294 \newunit{\second}{s}
295 \newunit{\ampere}{A}
296 \newunit{\kelvin}{K}
297 \newunit{\candela}{cd}
298
```

9.6.2 Units of length

```
299
300 \newunit{\picometer}{\pico\meter}
301 \newunit{\nanometer}{\nano\meter}
302 \newunit{\micrometer}{\@unit@micro\meter}
303 \newunit{\millimeter}{\milli\meter}
304 \newunit{\centimeter}{\centi\meter}
305 \newunit{\decimeter}{\deci\meter}
306 \newunit{\kilometer}{\kilo\meter}
307
```

9.6.3 Units of weight

```
308
309 \newunit{\femtogram}{\femto\gram}
310 \newunit{\picogram}{\pico\gram}
311 \newunit{\nanogram}{\nano\gram}
312 \newunit{\microgram}{\@unit@micro\gram}
313 \newunit{\milligram}{\milli\gram}
314
```

9.6.4 Units of quantity

```
315
316 \newunit{\femtomole}{\femto\mole}
317 \newunit{\picomole}{\pico\mole}
318 \newunit{\nanomole}{\nano\mole}
319 \newunit{\micromole}{\@unit@micro\mole}
320 \newunit{\millimole}{\milli\mole}
321
```

322

9.6.5 Units of time

```
323  
324 \newunit{\attosecond}{\atto\sek}  
325 \newunit{\femtosecond}{\femto\sek}  
326 \newunit{\picosecond}{\pico\sek}  
327 \newunit{\nanosecond}{\nano\sek}  
328 \newunit{\microsecond}{\@unit@micro\sek}  
329 \newunit{\millisecond}{\milli\sek}  
330  
331
```

9.6.6 amperage

```
332  
333 \newunit{\picoampere}{\pico\ampere}  
334 \newunit{\nanoampere}{\nano\ampere}  
335 \newunit{\microampere}{\@unit@micro\ampere}  
336 \newunit{\milliampere}{\milli\ampere}  
337 \newunit{\kiloampere}{\kilo\ampere}  
338
```

9.6.7 Percent

```
339  
340 \newunit{\percent}{\%}  
341
```

9.6.8 Volumes

```
342  
343 \ifunit@@liter  
344   \newunit{\liter}{l}  
345 \else  
346   \newunit{\liter}{L}  
347 \fi  
348  
349 \newunit{\femtoliter}{\femto\liter}  
350 \newunit{\picoliter}{\pico\liter}  
351 \newunit{\nanoliter}{\nano\liter}  
352 \newunit{\microliter}{\@unit@micro\liter}  
353 \newunit{\milliliter}{\milli\liter}  
354 \newunit{\centiliter}{\centi\liter}  
355 \newunit{\deciliter}{\deci\liter}  
356 \newunit{\hectoliter}{\hecto\liter}  
357
```

```
358 \newunit{\cubicmeter}{\meter\unitsuperscript{3}}
359 \newunit{\cubicmicrometer}{\micrometer\unitsuperscript{3}}
360 \newunit{\cubicmillimeter}{\millimeter\unitsuperscript{3}}
361
362
```

9.6.9 Areas

```
363
364 \newunit{\squaremeter}{\meter\unitsuperscript{2}}
365
366 \newunit{\ar}{a}
367 \newunit{\hektar}{\hecto\ar}
368
369 \newunit{\squarecentimeter}{\centimeter\unitsuperscript{2}}
370 \newunit{\squaremillimeter}{\millimeter\unitsuperscript{2}}
371 \newunit{\squarekilometer}{\kilometer\unitsuperscript{2}}
```

9.6.10 more units of weight

```
372
373 \newunit{\ton}{t}
374
```

9.6.11 Derived electrical units

```
375
376 \newunit{\volt}{V}
377 \newunit{\millivolt}{\milli\volt}
378
379 \newunit{\kilovolt}{\kilo\volt}
380
381 \newunit{\watt}{W}
382 \newunit{\milliwatt}{\milli\watt}
383 \newunit{\kilowatt}{\kilo\watt}
384 \newunit{\megawatt}{\mega\watt}
385
386 \newunit{\coulomb}{C}
```

Don't forget the options when typesetting Ω !

```
387 \ifunit@@Ohm
388   \newunit{\Ohm}{\unitOmega}
389   \newunit{\kiloohm}{\kilo\Ohm}
390   \newunit{\megaohm}{\mega\Ohm}
391   \newunit{\gigaohm}{\giga\Ohm}
392 \else
393   \ifunit@@redefgensymb\else
394     \newunit{\ohm}{\unitOmega}
```

```

395  \fi
396  \newunit{\kiloohm}{\kilo\ohm}
397  \newunit{\megaohm}{\mega\ohm}
398  \newunit{\gigaohm}{\giga\ohm}
399 \fi
400 \newunit{\siemens}{S}
401 \newunit{\millisiemens}{\milli\siemens}
402 \newunit{\farad}{F}
403 \newunit{\femtofarad}{\femto\farad}
404 \newunit{\picofarad}{\pico\farad}
405 \newunit{\nanofarad}{\nano\farad}
406 \newunit{\microfarad}{\@unit@micro\farad}
407 \newunit{\millifarad}{\milli\farad}

```

9.6.12 Units of energy

```

408 \newunit{\joule}{J}
409 \newunit{\millijoule}{\milli\joule}
410 \newunit{\kilojoule}{\kilo\joule}
411 \newunit{\megajoule}{\mega\joule}
412
413 \newunit{\calory}{cal}
414 \newunit{\kilocalory}{\kilo\calory}
415 \newunit{\electronvolt}{eV}
416 \newunit{\millielectronvolt}{\milli\electronvolt}
417 \newunit{\kilolectronvolt}{\kilo\electronvolt}
418 \newunit{\megaelectronvolt}{\mega\electronvolt}
419 \newunit{\gigaelectronvolt}{\giga\electronvolt}
420 \newunit{\teraelectronvolt}{\tera\electronvolt}
421

```

9.6.13 more units of time

```

422 \newunit{\minute}{min}
423 \newunit{\hour}{h}
424 \newunit{\days}{d}

```

9.6.14 more units of temperature

Don't forget `gensymb` when typesetting degree Celsius.

```

425 \ifunit@@Ohm
426   \newunit{\Celsius}{\unitCelsius}
427 \else
428   \ifunit@@redefgensymb\else
429     \newunit{\celsius}{\unitCelsius}
430   \fi
431 \fi

```

9.6.15 Angles and arcs

```

432 \newunit{\radian}{rad}
433 \newunit{\steradian}{sr}
434 \ifunit@@@Ohm
435   \newnosepunit{\Degree}{\unitDegree}
436 \else
437   \ifunit@@@redefgensymb\else
438     \newnosepunit{\degree}{\unitDegree}
439   \fi
440 \fi
441
442 \newunit{\arcmin}{\ensuremath{{}^{\prime}}}
443 \newunit{\arcsec}{\ensuremath{{}^{\prime\prime}}}

```

Some TeX to realize the syntax using ; as separator.

```

444 \DeclareRobustCommand{\arc}[1]{\expandafter\unit@arc#1; ;!}
445
446 \def\unit@arc#1;#2;#3!{%
447   \ifx\\#1\\\def\unit@arcdegreevalue{0}%
448   \else\def\unit@arcdegreevalue{#1}\fi%
449   \ifx\\#2\\\def\unit@arcminvalue{0}%
450   \else\def\unit@arcminvalue{#2}\fi%
451   \ifx\\#3\\\def\unit@arcsecvalue{0}%
452   \else\edef\unit@arcsecvalue{\expandafter\unit@strip#3;!}\fi%
453   \begingroup%
454     \let\unit@@xspace\relax%
455     \ifnum\unit@arcsecvalue=0\relax%
456       \ifnum\unit@arcminvalue=0\relax%
457         \ifnum\unit@arcdegreevalue=0\relax\else%
458           \unit@arcdegreevalue\degree%
459         \fi%
460       \else%
461         \unit@arcdegreevalue\degree%
462         \unitvaluesep%
463         \unit@arcminvalue\arcmin%
464       \fi%
465     \else%
466       \unit@arcdegreevalue\degree%
467       \unitvaluesep%
468       \unit@arcminvalue\arcmin%
469       \unitvaluesep%
470       \unit@arcsecvalue\arcsec%
471     \fi%
472   \endgroup%

```

```

473 }
474
475 \def\unit@strip#1;#2!{%
476   \ifx\#1\O\else#1\fi%
477 }
478

9.6.16 Frequencies

479 \newunit{\hertz}{Hz}
480   \newunit{\kilo\hertz}{\kilo\hertz}
481   \newunit{\megahertz}{\mega\hertz}
482   \newunit{\gigahertz}{\giga\hertz}
483

9.6.17 Force

484 \newunit{\newton}{N}
485   \newunit{\millinewton}{\milli\newton}
486   \newunit{\kilonewton}{\kilo\newton}

9.6.18 Pressure

487 \newunit{\pascal}{Pa}
488   \newunit{\hectopascal}{\hecto\pascal}
489 \newunit{\uBar}{bar}
490   \newunit{\millibar}{\milli\uBar}

9.6.19 magnetic field strength

491 \newunit{\weber}{Wb}

9.6.20 magnetic flux density

492 \newunit{\tesla}{T}

9.6.21 Induction

493 \newunit{\henry}{H}

9.6.22 Lumen

494 \newunit{\lumen}{lm}

9.6.23 Illuminance

495 \newunit{\lux}{lx}

9.6.24 Radioactivity

496 \newunit{\becquerel}{Bq}
497   \newunit{\megabecquerel}{\mega\becquerel}
498 \newunit{\curie}{Cu}

```

9.6.25 Sievert

```
499 \newunit{\sievert}{Sv}
500   \newunit{\millisievert}{\milli\sievert}
```

9.7 Loading abbreviations

Now the abbreviations are loaded if no option is specified to withhold some abbreviations.

```
501 \ifunit@@useabbrv
502   \ifunit@@useampabbrv
503     \InputIfFileExists{ampabbrv.cfg}%
504       {\PackageInfo{unitsdef}{Abbreviations for units of amperage loaded.}}%
505       {\PackageWarning{unitsdef}{ampabbrv.cfg not found!}}%
506   \fi
507   \ifunit@@usefreqabbrv
508     \InputIfFileExists{freqabbr.cfg}%
509       {\PackageInfo{unitsdef}{Abbreviations for units of frequency loaded.}}%
510       {\PackageWarning{unitsdef}{freqabbr.cfg not found!}}%
511   \fi
512   \ifunit@@usemolabbrv
513     \InputIfFileExists{molabbrv.cfg}%
514       {\PackageInfo{unitsdef}{Abbreviations for units of amount of substances loaded.}}%
515       {\PackageWarning{unitsdef}{molabbrv.cfg not found!}}%
516   \fi
517   \ifunit@@usevoltabbrv
518     \InputIfFileExists{voltabbr.cfg}%
519       {\PackageInfo{unitsdef}{Abbreviations for units of voltage loaded.}}%
520       {\PackageWarning{unitsdef}{voltabbr.cfg not found!}}%
521   \fi
522   \ifunit@@usevolabbrv
523     \InputIfFileExists{volabbrv.cfg}%
524       {\PackageInfo{unitsdef}{Abbreviations for units of volume loaded.}}%
525       {\PackageWarning{unitsdef}{volabbrv.cfg not found!}}%
526   \fi
527   \ifunit@@useweightabbrv
528     \InputIfFileExists{weigabbr.cfg}%
529       {\PackageInfo{unitsdef}{Abbreviations for units of weight loaded.}}%
530       {\PackageWarning{unitsdef}{weigabbr.cfg not found!}}%
531   \fi
532   \ifunit@@useenergyabbrv
533     \InputIfFileExists{enerabbr.cfg}%
534       {\PackageInfo{unitsdef}{Abbreviations for units of energy loaded.}}%
535       {\PackageWarning{unitsdef}{enerabbr.cfg not found!}}%
536   \fi
```

```

537 \ifunit@@uselengthabbrv
538   \InputIfFileExists{lengabbr.cfg}%
539     {\PackageInfo{unitsdef}{Abbreviations for units of length loaded.}}%
540     {\PackageWarning{unitsdef}{lengabbr.cfg not found!}}}%
541 \fi
542 \ifunit@@usetimeabbrv
543   \InputIfFileExists{timeabbr.cfg}%
544     {\PackageInfo{unitsdef}{Abbreviations for units of time loaded.}}%
545     {\PackageWarning{unitsdef}{timeabbr.cfg not found!}}}%
546 \fi
547 \fi
548 \ifunit@@useconfigfile
549   \InputIfFileExists{unitsdef.cfg}%
550     {\PackageInfo{unitsdef}{Local config file loaded.}}%
551     {\PackageInfo{unitsdef}{No local config file found.}}}%
552 \else
553   \PackageError{unitsdef}{Local config file not loaded.}
554 \fi
555 %
556 
```

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	A	B	C	D
\%	340	\ampere .. 295, 333–337		\celsius .. 41, 44, 429
\,	90, 112,	\ar .. 366, 367		\centi .. 275, 304, 354
	126, 139, 154, 158	\arc .. 444		\centiliter .. 354
\@setunitsepfalse		\arcmin .. 442, 463, 468	\centimeter .. 304, 369	
 45, 84,	\arcsec .. 443, 470	\coulomb .. 386	
	147, 152, 157,	\atto .. 261, 324	\cubicmeter .. 358	
	259–264, 266,	\attosecond .. 324	\cubicmicrometer .. 359	
	270, 274–276,		\cubicmillimeter .. 360	
	278, 279, 281–288			\curie .. 498
\@setunitseptrue 3,		\becquerel .. 496, 497		
	100, 114, 128, 141			\days .. 424
\@unit@micro			\deca .. 278	
 267, 271,	\calory .. 413, 414	\deci .. 276, 305, 355	
	302, 312, 319,	\candela .. 297	\deciliter .. 355	
	328, 335, 352, 406	\Celsius .. 426	\decimeter .. 305	

\Degree	435	\ifunit@@redefgensymb	\kilometer ..	306, 371
\degree			\kilonewton	486
438, 458, 461, 466		269, 393, 428, 437	\kiloohm	389, 396
		\ifunit@@useabbrv	\kilovolt	379
E	 12, 501	\kilowatt	383
\electronvolt	415	\ifunit@@useampabbrv		
\eV	416–420		L	
\exa	286	\ifunit@@useconfigfile	\liter	344, 346, 349–356
			\lumen	494
F		\ifunit@@useenergyabbrv	\lux	495
\farad	402–407			
\femto ...	262, 309,	\ifunit@@usefreqabbrv		
316, 325, 349, 403			M	
\femtofarad	403	\ifunit@@uselengthabbrv	\mega	282,
\femtogram	309		384, 390, 397,	
\femtoliter	349		411, 418, 481, 497	
\femtomole	316	\ifunit@@usemolabbrv	\megabecquerel ...	497
\femtosecond	325		\megaelectronvolt	418
G		\ifunit@@usetimeabbrv	\megahertz	481
\gensymbcelsius ...	41		\megajoule	411
\gensymbmicro	42	\ifunit@@usevolabbrv	\megaohm	390, 397
\gensymbohm	40		\megawatt	384
\giga	283,	\ifunit@@usevoltabbrv	\meter	290,
391, 398, 419, 482			300–306, 358, 364	
\gigaelectronvolt	419	\ifunit@@useweightabbrv	\Micro	266, 267
\gigahertz	482		\micro ..	42, 45, 270, 271
\gigaohm	391, 398	\ifunit@@xspace ..	\microampere	335
\gram ...	291, 292, 309–313 10, 12, 77	\microfarad	406
		\ilu	\microgram	312
H	 5, 150	\microliter	352
\hecto ...	279, 356, 367, 488	J	\micrometer ..	302, 359
\hectoliter	356	\joule	\micromole	319
\hektopascal	488		\microsecond	328
\hektar	367	K	\milli ..	274, 303, 313,
\henry	493	\kelvin	320, 329, 336,	
\hertz	479–482	\kilo ..	353, 377, 382,	
\hour	423	281, 292, 306,	401, 407, 409,	
		337, 379, 383,	416, 485, 490, 500	
		389, 396, 410,		
		414, 417, 480, 486	\milliampere	336
I		\kiloampere	\millibar	490
\if@setunitsep _2,		\kilocalory	\millielectronvolt	416
96, 110, 125, 138		\kiloelectronvolt	\millifarad	407
\ifunit@@liter . 8, 343		\kilogram	\milligram	313
\ifunit@@Ohm ... 4,		\kilohertz	\millijoule	409
265, 387, 425, 434		\kilojoule	\milliliter	353

\backslash millimeter	O	\backslash steradian 433
... 303, 360, 370	\backslash Ohm	388–391
\backslash millimole 320	\backslash ohm 40, 43, 394, 396–398	
\backslash millinewton 485	\backslash Omega 185	
\backslash millisecond 329	P	
\backslash millisiemens 401	\backslash pascal 487, 488	
\backslash millisievert 500	\backslash per 8, <u>146</u>	
\backslash millivolt 377	\backslash percent 340	
\backslash milliwatt 382	\backslash peta 285	
\backslash minute 422	\backslash pico 263,	
\backslash mole 293, 316–320	300, 310, 317,	
	326, 333, 350, 404	
N	\backslash picoampere 333	
\backslash nano 264,	\backslash picofarad 404	
301, 311, 318,	\backslash picogram 310	
327, 334, 351, 405	\backslash picoliter 350	
\backslash nanoampere 334	\backslash picometer 300	
\backslash nanofarad 405	\backslash picomole 317	
\backslash nanogram 311	\backslash picosecond 326	
\backslash nanoliter 351	R	
\backslash nanometer 301	\backslash radian 432	
\backslash nanomole 318	\backslash renewnosepunit 8, 133	
\backslash nanosecond 327	\backslash renewunit 8, 43, 44, <u>105</u>	
\backslash newnosepunit 8, <u>120</u> , 435, 438	S	
\backslash newton 484–486	\backslash second 294	
\backslash newunit 7, <u>91</u> ,	\backslash sek 324–329	
290–297, 300–	\backslash selectfont 255	
306, 309–313,	\backslash setMathCelsius 8, <u>190</u>	
316–320, 324–	\backslash setMathDegree . 8, <u>194</u>	
329, 333–337,	\backslash setMathmu 8, <u>186</u>	
340, 344, 346,	\backslash setMathOmega . 8, <u>182</u>	
349–356, 358–	\backslash setTextCelsius 8, <u>224</u>	
360, 364, 366,	\backslash setTextDegree . 8, <u>237</u>	
367, 369–371,	\backslash setTextmu 8, <u>211</u>	
373, 376, 377,	\backslash setTextOmega . 8, <u>198</u>	
379, 381–384,	\backslash SI 6, <u>251</u>	
386, 388–391,	\backslash siemens 400, 401	
394, 396–398,	\backslash sievert 499, 500	
400–411, 413–	\backslash squarecentimeter 369	
420, 422–424,	\backslash squarekilometer . 371	
426, 429, 432,	\backslash squaremeter 364	
433, 442, 443,	\backslash squaremillimeter 370	
479–482, 484–500	\backslash steradian 433	
T	\backslash tera 284, 420	
	\backslash teraelectronvolt 420	
	\backslash tesla 492	
	\backslash textcelsius . 193, 236	
	\backslash textdegree . 197, 249	
	\backslash textmu 189, 223	
	\backslash textohm 210	
	\backslash textsupserscript . 160	
	\backslash ton 373	
U	\backslash uBar 489, 490	
	\backslash unit@xspace	
	... 79, 81, 86,	
	95, 102, 109,	
	116, 130, 137,	
	143, 153, 254, 454	
	\backslash unit@arc 444, 446	
	\backslash unit@arcdegreevalue	
	... 447, 448,	
	457, 458, 461, 466	
	\backslash unit@arcminvalue	
	... 449,	
	450, 456, 463, 468	
	\backslash unit@arcsecvalue	
	451, 452, 455, 470	
	\backslash unit@strip 452, 475	
	\backslash unitCelsius	
	. 44, <u>170</u> , 426, 429	
	\backslash unitDegree	
	... <u>179</u> , 435, 438	
	\backslash unitfrac 148	
	\backslash unitMathCelsius	
	... <u>166</u> , 171, 191	
	\backslash unitMathDegree	
	... <u>169</u> , 180, 195	
	\backslash unitMathmu	
	... 165, 174, 187	
	\backslash unitmathmu <u>165</u>	
	\backslash unitMathOmega	
	... <u>162</u> , 177, 183	

\unitmu	45, <u>173</u> , 266, 270	... <u>168</u> , 180, 238	W
\unitOmega 43, <u>176</u> , 388, 394	\unitTextmu 164, 174, 212	\watt 381–384
\unitsep 8, <u>158</u>	\unittextmu <u>164</u>	\weber 491
\unitSIdef	... <u>250</u> , 255	\unitTextOmega <u>163</u> , 177, 199	Y
\unitsignonly	... 5, <u>83</u>	\unittimes 8, <u>157</u>	\yocto 259
\unitsuperscript	. . 8, <u>159</u> , 358–360, 364, 369–371	\unitvaluesep 5, <u>89</u> , 97, 98, 111, 462, 467, 469	\yotta 288
\unitTextCelsius	. . . <u>167</u> , 171, 225	V	Z
\unitTextDegree	\volt 376, 377, 379	\zepto 260
			\zetta 287