siunitx – A comprehensive (SI) units package*

Joseph Wright†
Released 2021-12-03

Contents

I siunitx-angle – Formatting angles 1
1 Formatting angles 1
  1.1 Key–value options 1

II siunitx-compound – Compound numbers and quantities 3

III siunitx.locale – Localisation 5

IV siunitx-number – Parsing and formatting numbers 6
1 Formatting numbers 6
  1.1 Key–value options 8

V siunitx-print – Printing material with font control 12
1 Printing quantities 12
  1.1 Key–value options 13

VI siunitx-quantity – Quantities 15

VII siunitx-symbol – Symbol-related settings 16

VIII siunitx-table – Formatting numbers in tables 17

∗This file describes v3.0.37, last revised 2021-12-03.
†E-mail: joseph.wright@morningstar2.co.uk
1 Numbers in tables 17
1.1 Key–value options ......................................... 17

IX siunitx-unit – Parsing and formatting units 19
1 Formatting units 19
2 Defining symbolic units 21
3 Per-unit options 22
4 Units in (PDF) strings 22
5 Pre-defined symbolic unit components 22
5.1 Key–value options ......................................... 25

X siunitx-abbreviations – Abbreviations 28

XI siunitx-binary – Binary units 31

XII siunitx-command – Units as document command 32
1 Creating units as document commands 32
1.1 Key–value options ......................................... 32

XIII siunitx-emulation – Emulation 33

Index 34
Part I

**siunitx-angle** – Formatting angles

1 Formatting angles

\[
\text{\textbackslash siunitx\_angle:nn} \quad \text{\textbackslash siunitx\_angle:n} \{\langle\text{angle}\rangle\}
\]

\[
\text{\textbackslash siunitx\_angle:nnn} \{\langle\text{degrees}\rangle\} \{\langle\text{minutes}\rangle\} \{\langle\text{seconds}\rangle\}
\]

Typeset the \langle angle \rangle (which may be given as separate \langle degree \rangle, \langle minute \rangle and \langle second \rangle components). The \langle angle \rangle (or components) may be given as expressions. The \langle angle \rangle should be a number as understood by \texttt{\textbackslash siunitx\_format\_number:nN}, with no uncertainty, exponent or imaginary part. The unit symbols for degrees, minutes and seconds are \texttt{\degree}, \texttt{\arcminute} and \texttt{\arcsecond}, respectively

1.1 Key–value options

The options defined by this submodule are available within the \texttt{l3keys siunitx} tree.

- **angle-mode**
  
  \texttt{angle-mode = (choice)}
  
  Selects how angles are formatted: a choice from the options \texttt{arc}, \texttt{decimal} and \texttt{input}. The option \texttt{arc} means that angles will always be typeset in arc (degree, minute, second) format, whilst \texttt{decimal} means that angles are typeset as a single decimal value. The \texttt{input} setting means that the input format (i.e. difference between \texttt{\textbackslash siunitx\_angle:n} and \texttt{\textbackslash siunitx\_angle:nnn}) is maintained. The standard setting is \texttt{input}.

- **angle-symbol-degree**
  
  \texttt{angle-symbol-degree = (symbol)}
  
  Sets the symbol used for arc degrees, minutes or seconds, respectively.

- **angle-symbol-minute**

- **angle-symbol-second**

- **angle-symbol-over-decimal**
  
  \texttt{angle-symbol-over-decimal = true|false}
  
  Determines if the arc separator is printed over the decimal marker, a format used in astronomy. The standard setting is \texttt{false}.

- **arc-separator**
  
  \texttt{arc-separator = (separator)}
  
  Inserted between arc parts (degree, minute and second components). The standard setting is \texttt{\,}.

- **fill-angle-degrees**
  
  \texttt{fill-arc-degrees = true|false}
  
  Determines whether a missing degrees part is zero-filled when printing an arc. The standard setting is \texttt{false}.

- **fill-angle-minutes**
  
  \texttt{fill-arc-minutes = true|false}
  
  Determines whether a missing minutes part is zero-filled when printing an arc. The standard setting is \texttt{false}.
| fill-angle-seconds | fill-arc-seconds = true|false |
|-------------------|----------------------|
|                   | Determines whether a missing seconds part is zero-filled when printing an arc. The standard setting is false. |

<table>
<thead>
<tr>
<th>number-angle-product</th>
<th>number-angle-product = ⟨separator⟩</th>
</tr>
</thead>
</table>
|                      | Inserted between the value of an angle and the unit (degree, minute or second component). The standard setting is \.,
Part II

\textbf{siunitx-compound} – Compound numbers and quantities

% siunitx-compound_number:n
% \siunitx_compound_number:n \{\textit{entries}\}
% Prints a set of numbers in the \textit{entries}, each of which should be given as a \textit{balanced text}. Unlike \texttt{\siunitx_number_list:nn}, this function may semantically take any form.

% siunitx_compound_quantity:nn
% \siunitx_compound_quantity:nn \{\textit{entries}\} \{\textit{unit}\}
% Prints a set of quantities in the \textit{entries}, each of which should be given as a \textit{balanced text}. Unlike \texttt{\siunitx_quantity_list:nn}, this function may semantically take any form.

% siunitx_number_list:nn
% \siunitx_number_list:nn \{\textit{entries}\}
% Prints the list of numbers in the \textit{entries}, each of which should be given as a \textit{balanced text}.

% siunitx_quantity_list:nn
% \siunitx_quantity_list:nn \{\textit{entries}\} \{\textit{unit}\}
% Prints the list of quantities in the \textit{entries}, each of which should be given as a \textit{balanced text}.

% siunitx_number_product:n
% \siunitx_number_product:n \{\textit{entries}\}
% Prints the series of numbers in the \textit{entries}, each of which should be given as a \textit{balanced text}.

% siunitx_quantity_product:nn
% \siunitx_quantity_product:nn \{\textit{entries}\} \{\textit{unit}\}
% Prints the series of quantities in the \textit{entries}, each of which should be given as a \textit{balanced text}.

% siunitx_number_range:nn
% \siunitx_number_range:nn \{\textit{start}\} \{\textit{end}\}
% Prints the range of numbers from the \textit{start} to the \textit{end}.

% siunitx_quantity_range:nn
% \siunitx_quantity_range:nn \{\textit{start}\} \{\textit{end}\} \{\textit{unit}\}
% Prints the range of quantities from the \textit{start} to the \textit{end}.

\sloppy
\begin{tabular}{l}
  \texttt{\l\_siunitx_list_separator_pair_tl} \texttt{\l\_siunitx_list_separator_tl} \texttt{\l\_siunitx_list_separator_final_tl} \\
\end{tabular}
Separators for lists of numbers and quantities.

3
Phrase (or similar) used between limits of a range.

- `compound-exponents` = combine|combine-bracket|individual
- `compound-final-separator` = (text)
- `compound-pair-separator` = (text)
- `compound-separator` = (text)
- `compound-separator-mode` = number|text
- `compound-units` = bracket|repeat|single
- `list-exponents` = combine|combine-bracket|individual
- `list-final-separator` = (text)
- `list-pair-separator` = (text)
- `list-separator` = (text)
- `list-units` = bracket|repeat|single
- `product-exponents` = combine|combine-bracket|individual
- `product-mode` = phrase|choice
- `product-phrase` = (text)
- `product-symbol` = (symbol)
- `range-exponents` = combine|combine-bracket|individual
- `range-final-separator` = (text)
- `range-pair-separator` = (text)
- `range-separator` = (text)
- `range-units` = bracket|repeat|single
Part III
siunitx-locale – Localisation

This submodule is concerned with localisation of siunitx output based on the locale. If the translations package is available, this is loaded here and used to provide various fixed strings for output.

```latex
\textbf{locale}  \hspace{1em} \texttt{locale = \langle locale \rangle}

Selects the (locale) used to apply standard settings for other keys, principally exponent-product, inter-unit-product and output-decimal-marker.
```
Part IV

**siunitx-number – Parsing and formatting numbers**

This submodule is dedicated to parsing and formatting numbers. A small number of \(\text{\LaTeX}\) \(\text{\texttt{\textbackslash m}}\), \(\text{\texttt{\textbackslash pm}}\), \(\text{\texttt{\textbackslash ll}}\), \(\text{\texttt{\textbackslash le}}\), \(\text{\texttt{\textbackslash gg}}\) and \(\text{\texttt{\textbackslash ge}}\) are used to replace two-character input; \(\text{\texttt{\textbackslash pm}}\) is also required for the output of uncertainties. The standard settings require \(\text{\texttt{\textbackslash times}}\). For the display of colored negative numbers, the command \(\text{\texttt{\textbackslash color}}\) is assumed to be available. Where the latter may apply, numbers should be printed inside a group: note that \(\text{\LaTeX}\) grouping is not added within formatted numbers as they may need to be decomposed into parts (see `\texttt{\textbackslash siunitx\textunderscore number\textunderscore output:NN}`). Such a color will be the first part of the result, meaning that a test for an initial \(\text{\texttt{\textbackslash color}}\) and following brace group may be used to detect/remove/adjust this part.

1 Formatting numbers

\begin{verbatim}
\texttt{\textbackslash siunitx\textunderscore number\textunderscore parse:nN \{\langle number\rangle\} \langle tl var\rangle}
\texttt{\texttt{\textbackslash siunitx\textunderscore number\textunderscore parse:VN}}
\end{verbatim}

Parses the number and stores the resulting internal representation in the \(\langle tl\ var\rangle\). The parsing is influenced by the various key–value settings for numerical input. The \(\langle number\rangle\) should comprise a single real value, possibly with comparator, uncertainty and exponent parts. If the number is invalid, or if number parsing is disabled, the result will be an entirely empty \(\langle tl\ var\rangle\).

The structure of a valid number is:

\[
\langle\text{comparator}\rangle\langle\text{sign}\rangle\langle\text{integer}\rangle\langle\text{decimal}\rangle\langle\text{uncertainty}\rangle\langle\text{exponent sign}\rangle\langle\text{exponent}\rangle
\]

where the two sign parts must be single tokens if present, and all other components must be given in braces. The number will have at least one digit for both the \(\langle\text{integer}\rangle\) and \(\langle\text{exponent}\rangle\) parts; these are required. The \(\langle\text{uncertainty}\rangle\) part should either be blank or contain an \(\langle\text{identifier}\rangle\) (as a brace group), followed by one or more data entries. Valid \(\langle\text{identifiers}\rangle\) currently are

8 A single symmetrical uncertainty (e.g. a statistical standard uncertainty)
\texttt{\textbackslash siunitx\_number\_process:NN} \texttt{\textbackslash siunitx\_number\_process:Н (tl var1) (tl var2)}

Applies a set of number processing operations to the \textit{(internal number)} stored in the \texttt{⟨tl var1⟩}, \textit{viz. in order}

1. Dropping uncertainty
2. Converting to scientific mode (or similar)
3. Rounding
4. Dropping zero decimal part
5. Forcing a minimum number of digits

with the result stored in \texttt{⟨tl var2⟩}.

\texttt{\textbackslash siunitx\_number\_output:N} \texttt{\textbackslash siunitx\_number\_output:N} \texttt{⟨number⟩}
\texttt{\textbackslash siunitx\_number\_output:n} \texttt{\textbackslash siunitx\_number\_output:n} \texttt{⟨number⟩ ⟨marker⟩}
\texttt{\textbackslash siunitx\_number\_output:nN} \texttt{\textbackslash siunitx\_number\_output:nN} \texttt{⟨number⟩ ⟨marker⟩}
\texttt{\textbackslash siunitx\_number\_output:nN} \texttt{\textbackslash siunitx\_number\_output:nN} \texttt{⟨number⟩ ⟨marker⟩}

Formats the \textit{(number)} (in the \texttt{siunitx} internal format), producing the result in a form suitable for typesetting in math mode. The details for the formatting are controlled by a number of key--value options. Note that formatting does not apply any manipulation (processing) to the number. This function is usable in an \texttt{e-} or \texttt{x-type expansion}, and further uncontrolled expansion is prevented by appropriate use of \texttt{\exp\_not:n} internally.

In the \texttt{NN} version, the \textit{(marker)} token is inserted at each possible alignment position in the output, \textit{viz.}

- Between the comparator and the integer \textit{(before} any sign for the integer)
- Between the sign and the first digit of the integer
- Both sides of the decimal marker
- Both sides of the separated uncertainty sign \textit{(i.e. after} the decimal part and before any integer uncertainty part)
- Both sides of the decimal marker for a separated uncertainty
- Both sides of the multiplication symbol for the exponent part.

The \texttt{n} and \texttt{nN} version take a token list, which should be in the internal \texttt{siunitx} format.

\texttt{\textbackslash siunitx\_number\_format:nN} \texttt{\textbackslash siunitx\_number\_format:nN \{(number)}\texttt{⟨tl var⟩)}

Carries out a combination of \texttt{\textbackslash siunitx\_number\_parse:nN}, \texttt{\textbackslash siunitx\_number\_process:NN} and \texttt{\textbackslash siunitx\_number\_output:N} using \texttt{x-type expansion} to place the result in the \texttt{⟨tl var⟩}. If \texttt{\_siunitx\_number\_parse\_bool} if \texttt{false}, the input is simply stored inside the \texttt{⟨tl var⟩} inside \texttt{\ensuremath{}}.

\texttt{\textbackslash siunitx\_number\_adjust\_exponent:Nn} \texttt{\textbackslash siunitx\_number\_adjust\_exponent:Nn} \texttt{⟨number⟩ \{(fp expr)}\texttt{\}}

\texttt{\textbackslash siunitx\_number\_adjust\_exponent:nn} \texttt{\textbackslash siunitx\_number\_adjust\_exponent:nn} \texttt{⟨number⟩ \{(fp expr)}\texttt{\}}

Adjusts the exponent of the \textit{(number)} (in internal format) by the \textit{(fp expr)} and leaves the result in the input stream.
Replaces all multi-token signs and comparators in the \(tl\ \text{var}\) with their single-token equivalents. Replaces any active hyphen tokens with non-active versions.

Determines if the \(\text{tokens}\) form a valid number which can be fully parsed by \textsf{siunitx}.

Determines if the \(\text{token}\) is valid in a number based on those tokens currently set up for detection in a number.

A switch to control whether ambiguous numbers are bracketed: this can also be covered in quantity formatting by a setting there.

A switch to control whether any parsing is attempted for numbers.

The list of possible input comparators, exponent markers and signs.

The list of possible input decimal marker(s), and the output marker.

1.1 Key–value options

The options defined by this submodule are available within the \texttt{l3keys siunitx} tree.

\begin{verbatim}
\texttt{bracket-ambiguous-numbers = true|false}
\texttt{bracket-negative-numbers = true|false}
\texttt{drop-exponent = true|false}
\texttt{drop-uncertainty = true|false}
\texttt{drop-zero-decimal = true|false}
\end{verbatim}
evaluate-expression = true|false

exponent-base = (base)

exponent-mode = engineering|fixed|input|scientific

exponent-product = (symbol)

expression = (expression)

fixed-exponent = (exponent)

group-digits = all|decimal|integer|none

group-minimum-digits = (value)

group-separator = (symbol)

input-close-uncertainty = (tokens)

input-comparators = (tokens)

input-open-uncertainty = (tokens)

input-decimal-markers = (tokens)

input-digits = (tokens)

input-exponent-markers = (tokens)

input-signs = (tokens)

input-uncertainty-signs = (tokens)
minimum-decimal-digits minimum-decimal-digits = \langle \text{min} \rangle

minimum-integer-digits minimum-integer-digits = \langle \text{min} \rangle

negative-color negative-color = \langle \text{color} \rangle

output-close-uncertainty output-close-uncertainty = \langle \text{symbol} \rangle

output-decimal-marker output-decimal-marker = \langle \text{symbol} \rangle

output-open-uncertainty output-open-uncertainty = \langle \text{symbol} \rangle

parse-numbers parse-numbers = true|false

print-implicit-plus print-implicit-plus = true|false

print-unity-mantissa print-unity-mantissa = true|false

print-zero-exponent print-zero-exponent = true|false

retain-explicit-plus retain-explicit-plus = true|false

retain-zero-uncertainty retain-zero-uncertainty = true|false

round-half round-half = even|up

round-minimum round-minimum = \langle \text{min} \rangle

round-mode round-mode = \text{figures}|\text{none}|\text{places}|\text{uncertainty}

round-pad round-pad = true|false

round-precision round-precision = \langle \text{precision} \rangle

tight-spacing tight-spacing = true|false
uncertainty-mode  uncertainty-mode = compact|compact-marker|full|separate

uncertainty-separator  uncertainty-separator = ⟨separator⟩
Part V

siunitx-print – Printing material with font control

1 Printing quantities

This submodule is focussed on providing controlled printing for numbers and units. Key to this is control of font: conventions for printing quantities mean that the exact nature of the output is important. At the same time, this module provides flexibility for the user in terms of which aspects of the font are responsive to the surrounding general text. Printing material may also take place in text or math mode.

The printing routines assume that normal \LaTeX \texttt{2e} font selection commands are available, in particular \texttt{\textbackslash bfseries, \textbackslash mathrm, \textbackslash mathversion, \textbackslash fontfamily, \textbackslash fontseries and \textbackslash fontshape, \textbackslash familydefault, \textbackslash seriesdefault, \textbackslash shapedefault and \textbackslash selectfont.} It also requires the standard \LaTeX \texttt{2e} kernel commands \texttt{\textbackslash ensuremath, \textbackslash mbox, \textbackslash textsubscript and \textbackslash textsuperscript} for printing in text mode. The following packages are also required to provide the functionality detailed.

- \texttt{\textbackslash color}: support for color using \texttt{\textbackslash textcolor}
- \texttt{\textbackslash textcomp}: \texttt{\textbackslash textminus, \textbackslash textpm \textbackslash texttimes} and \texttt{\textbackslash textcenteredperiod} for printing in text mode
- \texttt{\textbackslash amstext}: the \texttt{\textbackslash text} command for printing in text mode

For detection of math mode fonts, as well as \texttt{\textbackslash mathrm}, the existence of \texttt{\textbackslash symoperators} is assumed; other math font commands are not \textit{required} to exist.

\begin{verbatim}
\siunitx_print_number:n {⟨material⟩}
\siunitx_print_unit:n {⟨material⟩}
\end{verbatim}

Prints the \texttt{⟨material⟩} according the the prevailing settings for the submodule as applicable to the \texttt{⟨type⟩} of content \texttt{(number or unit)}. The \texttt{⟨material⟩} should comprise normal \LaTeX \texttt{2e} mark-up for numbers or units. In particular, units will typically use \texttt{\textbackslash mathrm} to indicate material to be printed in the current upright roman font, and ^ and _ will typically be used to indicate super- and subscripts, respectively. These elements will be correctly handled when printing for example using \texttt{\textbackslash mathsfb} in math mode, or using only text fonts.

\begin{verbatim}
\siunitx_print_match:n {⟨material⟩}
\siunitx_print_math:n {⟨material⟩}
\siunitx_print_text:n {⟨material⟩}
\end{verbatim}

Prints the \texttt{⟨material⟩} as described for \texttt{\textbackslash siunitx_print...:n} but with a fixed text or math mode output. The printing does \textit{not set} color (which is managed on a \texttt{unit/number} basis), but otherwise sets the font as described above. The \texttt{match} function uses either the prevailing math or text mode.
1.1 Key–value options

The options defined by this submodule are available within the \texttt{l3keys siunitx} tree.

\begin{itemize}
\item \texttt{color} \quad \texttt{color} = \langle \texttt{color} \rangle
  \begin{itemize}
  \item Color to apply to printed output: the latter should be a named color defined for use with \texttt{textcolor}. The standard setting is empty (no color).
  \end{itemize}
\item \texttt{mode} \quad \texttt{mode} = \texttt{match}|\texttt{math}|\texttt{text}
  \begin{itemize}
  \item Selects which mode (math or text) the output is printed in: a choice from the options \texttt{match}, \texttt{math} or \texttt{text}. The option \texttt{match} matches the mode prevailing at the point \texttt{\siunitx_print...:n} is called. The \texttt{math} and \texttt{text} options choose the relevant \TeX{} mode for printing. The standard setting is \texttt{math}.
  \end{itemize}
\item \texttt{number-color} \quad \texttt{number-color} = \langle \texttt{color} \rangle
  \begin{itemize}
  \item Color to apply to numbers in output: the latter should be a named color defined for use with \texttt{textcolor}. The standard setting is empty (no color).
  \end{itemize}
\item \texttt{number-mode} \quad \texttt{number-mode} = \texttt{match}|\texttt{math}|\texttt{text}
  \begin{itemize}
  \item Selects which mode (math or text) the numbers are printed in: a choice from the options \texttt{match}, \texttt{math} or \texttt{text}. The option \texttt{match} matches the mode prevailing at the point \texttt{\siunitx_prin_number:n} is called. The \texttt{math} and \texttt{text} options choose the relevant \TeX{} mode for printing. The standard setting is \texttt{math}.
  \end{itemize}
\item \texttt{propagate-math-font} \quad \texttt{propagate-math-font} = \texttt{true}|\texttt{false}
  \begin{itemize}
  \item Switch to determine if the currently-active math font is applied within printed output. This is relevant only when \texttt{\siunitx_print...:n} is called from within math mode: in text mode there is not active math font. When not active, math mode material will be typeset using standard math mode fonts without any changes being made to the supplied argument. The standard setting is \texttt{false}.
  \end{itemize}
\item \texttt{reset-math-version} \quad \texttt{reset-math-version} = \texttt{true}|\texttt{false}
  \begin{itemize}
  \item Switch to determine whether the active \texttt{\mathversion} is reset to \texttt{normal} when printing in math mode. Note that math version is typically used to select \texttt{\boldmath}, though it is also be used by \texttt{e.g. sansmath}. The standard setting is \texttt{true}.
  \end{itemize}
\item \texttt{reset-text-family} \quad \texttt{reset-text-family} = \texttt{true}|\texttt{false}
  \begin{itemize}
  \item Switch to determine whether the active text family is reset to \texttt{\rmfamily} when printing in text mode. The standard setting is \texttt{true}.
  \end{itemize}
\item \texttt{reset-text-series} \quad \texttt{reset-text-series} = \texttt{true}|\texttt{false}
  \begin{itemize}
  \item Switch to determine whether the active text series is reset to \texttt{\mdseries} when printing in text mode. The standard setting is \texttt{true}.
  \end{itemize}
\item \texttt{reset-text-shape} \quad \texttt{reset-text-shape} = \texttt{true}|\texttt{false}
  \begin{itemize}
  \item Switch to determine whether the active text shape is reset to \texttt{\upshape} when printing in text mode. The standard setting is \texttt{true}.
  \end{itemize}
\end{itemize}
text-family-to-math text-family-to-math = true|false
Switch to determine if the family of the current text font should be applied (where possible) to printing in math mode. The standard setting is false.

text-font-command text-font-command = (cmd)
Command applied to text during output, inserted after any reset of font set-up. This can therefore be used to apply non-standard font set up when printing in text mode. The standard setting is empty.

text-series-to-math text-series-to-math = true|false
Switch to determine if the weight of the current text font should be applied (where possible) to printing in math mode. This is achieved by setting the \mathversion, and so will override reset-math-version. The mappings between text and math weight are set. The standard setting is false.

unit-color unit-color = ⟨color⟩
Color to apply to units in output: the latter should be a named color defined for use with \textcolor. The standard setting is empty (no color).

unit-mode unit-mode = match|math|text
Selects which mode (math or text) units are printed in: a choice from the options match, math or text. The option match matches the mode prevailing at the point \siunitx_print...n is called. The math and text options choose the relevant \TeX mode for printing. The standard setting is math.

series-version-mapping series-version-mapping / ⟨weight⟩ = ⟨version⟩
Defines how \siunitx maps from text font weight to math font version. The pre-defined weights are those used as-standard by autoinst:

- ul
- el
- l
- sl
- m
- sb
- b
- eb
- ub

As standard, the m weight maps to normal math version whilst all of the b weights map to bold and all of the l weights map to light.
Part VI

\textbf{siunitx-quantity} – Quantities

This submodule is focussed on providing controlled printing for quantities: the combination of a number and a unit. It largely builds on the submodules \texttt{siunitx-number} and \texttt{siunitx-unit}. A small number of adjustments are made to standard set up in the latter to reflect additional functionality added here.

\begin{itemize}
  \item \texttt{\textbackslash siunitx\_quantity:nn} \texttt{\textbackslash siunitx\_quantity:nn \{\langle number\rangle\} \{\langle unit\rangle\}}
  \begin{itemize}
    \item Parses the \langle number\rangle and the \langle unit\rangle as detailed for \texttt{\textbackslash siunitx\_number\_parse:nN} and \texttt{\textbackslash siunitx\_unit\_format:nN}, the prints the results using \texttt{\textbackslash siunitx\_print\_unit:n}.
  \end{itemize}
  \item \texttt{\textbackslash siunitx\_quantity\_print:nn} \texttt{\textbackslash siunitx\_quantity\_print:nn \{\langle number\rangle\} \{\langle unit\rangle\}}
  \begin{itemize}
    \item A low-level function which prints the quantity directly: there is no processing applied to either the \langle number\rangle or \langle unit\rangle. The two parts are printed using \texttt{\textbackslash siunitx\_print\_unit:n} and appropriate spacing and break-prevention is applied.
  \end{itemize}
  \item allow-quantity-breaks \texttt{allow-quantity-breaks = true|false}
  \begin{itemize}
    \item Specifies whether breaks are permitted between units. The standard setting is \texttt{false}.
  \end{itemize}
  \item prefix-mode \texttt{prefix-mode = combine-exponent|extract-exponent|input}
  \begin{itemize}
    \item Selects the method used for producing prefixes: a choice from the options \texttt{combine-exponent}, \texttt{extract-exponent} and \texttt{input}. The option \texttt{combine-exponent} combines any exponent from the number with the prefix of the first unit, and prints the updated prefix. The option \texttt{extract-exponent} removes all prefixes from the unit, and combines them with the exponent of number. The option \texttt{input} prints prefixes and exponent as given in the source. The standard setting is \texttt{input}.
  \end{itemize}
  \item quantity-product \texttt{quantity-product = \{tokens\}}
  \begin{itemize}
    \item The product marker used between a number and the unit. The standard setting is \texttt{,}.
  \end{itemize}
  \item separate-uncertainty-units \texttt{separate-uncertainty-units = bracket|repeat|single}
  \begin{itemize}
    \item Specifies how units are applied when a separated uncertainty is present: a choice from \texttt{bracket}, \texttt{repeat} and \texttt{single}. The option \texttt{bracket} places brackets around the number, with the unit given after these. The option \texttt{repeat} means that the unit it printed with the main value and with the uncertainty. When \texttt{single} is set, the unit is printed only once and no brackets are applied. The standard setting is \texttt{bracket}.
  \end{itemize}
\end{itemize}
Part VII

\texttt{siunitx-symbol} – Symbol-related settings
Part VIII

\textbf{siunitx-table – Formatting numbers in tables}

\section{Numbers in tables}

This submodule is concerned with formatting numbers in table cells or similar fixed-width contexts. The main function, \texttt{\siunitx\_cell\_begin:w}, is designed to work with the normal LaTeX tabular cell construct featuring \texttt{\ignorespaces}. Therefore, if used outside of a LaTeX tabular, it is necessary to provide this token.

\begin{verbatim}
\siunitx\_cell\_begin:w \siunitx\_cell\_begin:w (preamble) \ignorespaces \siunitx\_cell\_end:
\end{verbatim}

Collects the \texttt{(preamble)} and \texttt{(content)} tokens, and determines if it is text or a number (as parsed by \texttt{\siunitx\_number\_parse:nN}). It produces output of a fixed width suitable for alignment in a table, although it is not required that the code is used within a cell. Note that \texttt{\ignorespaces} must occur in the “cell”: it marks the end of the \LaTeX \texttt{\halign} template.

\subsection{Key–value options}

The options defined by this submodule are available within the \texttt{l3keys siunitx} tree.

\begin{verbatim}
\texttt{table-align-comparator} = true|false
\texttt{table-align-exponent} = true|false
\texttt{table-align-text-after} = true|false
\texttt{table-align-text-before} = true|false
\texttt{table-align-uncertainty} = true|false
\end{verbatim}

Switch which determines whether alignment of comparators is attempted within table cells. The standard setting is \texttt{true}.

Switch which determines whether alignment of exponents is attempted within table cells. The standard setting is \texttt{true}.

Switch which determines whether alignment of text falling after a number is attempted within table cells. The standard setting is \texttt{true}.

Switch which determines whether alignment of text falling before a number is attempted within table cells. The standard setting is \texttt{true}.

Switch which determines whether alignment of separated uncertainty values is attempted within table cells. The standard setting is \texttt{true}.
| **table-alignment**   | table-alignment = center|left|right |
|-----------------------|-------------------------|
| Selects the alignment of all tabular content with the margins of the table cell (or other boundary). See also **table-number-alignment** and **table-text-alignment**. The standard setting is **center**. |

| **table-alignment-mode** | table-alignment-mode = format|marker|none |
|-------------------------|-----------------------------|
| Selects the method used to align numbers with the desired position in the cell (set by **table-alignment**). When set to **format**, a dedicated amount of space is calculated from the **table-format**. When **marker** is selected, alignment is carried out symmetrically around the decimal marker. Finally, **none** switches off all alignment: numbers are parsed and formatted but with no attempt at placement within the cell. The standard setting is **marker**. |

| **table-auto-round** | table-auto-round = true|false |
|----------------------|-----------------------|
| Switch which determines whether numbers are rounded to fit within the **table-format** specification (if possible). The standard setting is **false**. |

<table>
<thead>
<tr>
<th><strong>table-column-width</strong></th>
<th>table-column-width = ⟨width⟩</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sets the width of the table column used for numbers. This is only used when <strong>table-fixed-width</strong> is <strong>true</strong>.</td>
<td></td>
</tr>
</tbody>
</table>

| **table-fixed-width** | table-fixed-width = true|false |
|-----------------------|-----------------------|
| Switch which determines whether a fixed-width column is used for numbers in tables. When **true**, the width is taken from **table-column-width**. The standard setting is **false**. |

<table>
<thead>
<tr>
<th><strong>table-format</strong></th>
<th>table-format = ⟨format⟩</th>
</tr>
</thead>
<tbody>
<tr>
<td>Describes the amount of space that should be reserved when <strong>table-alignment-mode</strong> is set to <strong>format</strong>. The ⟨format⟩ takes the same general form as input for a table cell, with the numerical parts describing how many digits to reserve space for. For example, 1.2e3 would allow space for one digit in the integer part, two in the decimal part and three in the exponent part. Signs can be allowed for using any valid input sign, so for example +1.2 \pm 1.2 would allow for a sign, a number with one integer and two decimal digits and an uncertainty of the same size.</td>
<td></td>
</tr>
</tbody>
</table>

| **table-number-alignment** | table-number-alignment = center|left|right |
|---------------------------|-----------------------------|
| Selects the alignment of numerical content with the margins of the table cell (or other boundary). See also **table-alignment** and **table-text-alignment**. The standard setting is **center**. |

| **table-text-alignment** | table-text-alignment = center|left|none|right |
|--------------------------|-----------------------------|
| Selects the alignment of non-numerical content with the margins of the table cell (or other boundary). See also **table-alignment** and **table-number-alignment**. Notice the additional support for **none** here. The standard setting is **center**. |
Part IX
siunitx-unit – Parsing and formatting units

This submodule is dedicated to formatting physical units. The main function, \siunitx_unit_format:nN, takes user input specify physical units and converts it into a formatted token list suitable for typesetting in math mode. While the formatter will deal correctly with “literal” user input, the key strength of the module is providing a method to describe physical units in a “symbolic” manner. The output format of these symbolic units can then be controlled by a number of key–value options made available by the module.

A small number of L\TeX\ \epsilon math mode commands are assumed to be available as part of the formatted output. The \texttt{mathchoice} command (normally the \LaTeX primitive) is needed when using \texttt{per-mode = symbol-or-fraction}. The commands \texttt{frac}, \texttt{mathrm}, \texttt{mbox}, \texttt{\_} and \texttt{\_\_}, are used by the standard module settings. For the display of colored (highlighted) and cancelled units, the commands \texttt{textcolor} and \texttt{cancel} are assumed to be available.

1 Formatting units

\begin{verbatim}
\siunitx_unit_format:nN \siunitx_unit_format:xN
\end{verbatim}

This function converts the input \texttt{(units)} into a processed \texttt{(tl var)} which can then be inserted in math mode to typeset the material. Where the \texttt{(units)} are given in symbolic form, described elsewhere, this formatting process takes place in two stages: the \texttt{(units)} are parsed into a structured form before the generation of the appropriate output form based on the active settings. When the \texttt{(units)} are given as literals, processing is minimal: the characters . and ~ are converted to unit products (boundaries). In both cases, the result is a series of tokens intended to be typeset in math mode with appropriate choice of font for typesetting of the textual parts.

For example,

\begin{verbatim}
\siunitx_unit_format:nN \{ \kilo \metre \per \second \} \l_tmpa_tl
\end{verbatim}

will, with standard settings, result in \texttt{\l_tmpa_tl} being set to

\begin{verbatim}
\mathrm{km} \mathrm{\_} \mathrm{s}^{-1}
\end{verbatim}

19
This function formats the \( \langle \text{units} \rangle \) in the same way as described for \( \text{\texttt{\textbackslash unit\_format:nN}} \). When the input is given in symbolic form, any decimal unit prefixes will be extracted and the overall power of ten that these represent will be stored in the \( \langle \text{fp var} \rangle \).

For example,

\begin{verbatim}
\texttt{\textbackslash unit\_format\_extract\_prefixes:nNN \{ \texttt{kilo} \ \texttt{metre} \ \texttt{per} \ \texttt{second} \} \ l\_tmpa\_tl \ l\_tmpa\_fp}
\end{verbatim}

will, with standard settings, result in \( \l\_tmpa\_tl \) being set to \( \text{\texttt{m\,s^{-1}}} \) with \( \l\_tmpa\_fp \) taking value 3. Note that the latter is a floating point variable: it is possible for non-integer values to be obtained here.

This function formats the \( \langle \text{units} \rangle \) in the same way as described for \( \text{\texttt{\textbackslash unit\_format:nN}} \). The \( \langle \text{exponent} \rangle \) is combined with any prefix for the first unit of the \( \langle \text{units} \rangle \), and an updated prefix is introduced.

For example,

\begin{verbatim}
\texttt{\textbackslash unit\_format\_combine\_exponent:nnN \{ \texttt{metre} \ \texttt{per} \ \texttt{second} \} \ \{ 3 \} \ l\_tmpa\_tl}
\end{verbatim}

will, with standard settings, result in \( \l\_tmpa\_tl \) being set to \( \text{\texttt{km\,s^{-1}}} \)

These function formats the \( \langle \text{units} \rangle \) in the same way as described for \( \text{\texttt{\textbackslash unit\_format:nN}} \). The units are multiplied by the \( \langle \text{factor} \rangle \), and further processing takes place as previously described.

For example,

\begin{verbatim}
\texttt{\textbackslash unit\_format\_multiply:nnN \{ \texttt{metre} \ \texttt{per} \ \texttt{second} \} \ \{ 3 \} \ l\_tmpa\_tl}
\end{verbatim}

will, with standard settings, result in \( \l\_tmpa\_tl \) being set to \( \text{\texttt{km^{3}\,s^{-3}}} \)
2 Defining symbolic units

\sinunitx_declare_prefix:Nnn \sinunitx_declare_prefix:Nnn \langle \text{prefix} \rangle \{ \langle \text{power} \rangle \} \{ \langle \text{symbol} \rangle \}
\sinunitx_declare_prefix:Nnx

Defines a symbolic \langle \text{prefix} \rangle (which should be a control sequence such as \texttt{\textbackslash kilo}) to be converted by the parser to the \langle \text{symbol} \rangle. The latter should consist of literal content \textit{(e.g. k)}. In literal mode the \langle \text{symbol} \rangle will be typeset directly. The prefix should represent an integer \langle \text{power} \rangle of 10, and this information may be used to convert from one or more \langle \text{prefix} \rangle symbols to an overall power applying to a unit. See also \sinunitx_declare_prefix:Nnn.

\sinunitx_declare_prefix:Nn \sinunitx_declare_prefix:Nn \langle \text{prefix} \rangle \{ \langle \text{symbol} \rangle \}

Defines a symbolic \langle \text{prefix} \rangle (which should be a control sequence such as \texttt{\textbackslash kilo}) to be converted by the parser to the \langle \text{symbol} \rangle. The latter should consist of literal content \textit{(e.g. k)}. In literal mode the \langle \text{symbol} \rangle will be typeset directly. In contrast to \sinunitx_declare_prefix:Nnn, there is no assumption about the mathematical nature of the \langle \text{prefix} \rangle, i.e. the prefix may represent a power of any base. As a result, no conversion of the \langle \text{prefix} \rangle to a numerical power will be possible.

\sinunitx_declare_power:NNn \sinunitx_declare_power:NNn \langle \text{pre-power} \rangle \langle \text{post-power} \rangle \{ \langle \text{value} \rangle \}

Defines two symbolic \langle \text{powers} \rangle (which should be control sequences such as \texttt{\textbackslash squared}) to be converted by the parser to the \langle \text{value} \rangle. The latter should be an integer or floating point number in the format defined for \texttt{l3fp}. Powers may precede a unit or be give after it; both forms are declared at once, as indicated by the argument naming. In literal mode, the \langle \text{value} \rangle will be applied as a superscript to either the next token in the input (for the \langle \text{pre-power} \rangle) or appended to the previously-typeset material (for the \langle \text{post-power} \rangle).

\sinunitx_declare_qualifier:Nn \sinunitx_declare_qualifier:Nn \langle \text{qualifier} \rangle \{ \langle \text{meaning} \rangle \}

Defines a symbolic \langle \text{qualifier} \rangle (which should be a control sequence such as \texttt{\textbackslash catalyst}) to be converted by the parser to the \langle \text{meaning} \rangle. The latter should consist of literal content \textit{(e.g. cat)}. In literal mode the \langle \text{meaning} \rangle will be typeset following a space after the unit to which it applies.

\sinunitx_declare_unit:Nn \sinunitx_declare_unit:Nn \langle \text{unit} \rangle \{ \langle \text{meaning} \rangle \}
\sinunitx_declare_unit:Nnn \sinunitx_declare_unit:Nnn \langle \text{unit} \rangle \{ \langle \text{meaning} \rangle \} \{ \langle \text{options} \rangle \}
\sinunitx_declare_unit:Nxn \sinunitx_declare_unit:Nxn \langle \text{unit} \rangle \{ \langle \text{meaning} \rangle \} \{ \langle \text{options} \rangle \}

Defines a symbolic \langle \text{unit} \rangle (which should be a control sequence such as \texttt{\textbackslash kilogram}) to be converted by the parser to the \langle \text{meaning} \rangle. The latter may consist of literal content \textit{(e.g. kg)}, other symbolic unit commands \textit{(e.g. \texttt{\textbackslash kilo\textbackslash gram}) or a mixture of the two. In literal mode the \langle \text{meaning} \rangle will be typeset directly. The version taking an \langle \text{options} \rangle argument may be used to support per-unit options: these are applied at the top level or using \sinunitx_unit_options_apply:n.

\l_{\text{\textbackslash siunitx_unit\_font\_tl}}
The font function which is applied to the text of units when constructing formatted units: set by \texttt{font-command}.
The fraction function which is applied when constructing fractional units: set by `\frac{\ldots}{\ldots}`.

This sequence contains all of the symbolic names defined: these will be in the form of control sequences such as `\kilogram`. The order of the sequence is unimportant. This includes prefixes and powers as well as units themselves.

This sequence contains all of the symbolic unit names defined: these will be in the form of control sequences such as `\kilogram`. In contrast to `\l_sunitx_unit_symbolic_seq`, it only holds units themselves.

### 3 Per-unit options

`\sunitx_unit_options_apply:n \sunitx_unit_options_apply:n \{unit(s)\}`

Applies any unit-specific options set up using `\sunitxDeclareUnit:Nnn`. This allows there use outside of unit formatting, for example to influence spacing in quantities. The options are applied only once at a given group level, which allows for user over-ride via `\keys_set:nn { sunitx } { ... }`.

### 4 Units in (PDF) strings

`\sunitx_unit_pdfstring_context: `\group_begin:`\sunitx_unit_pdfstring_context:`\{Expansion context\}\{units\}`\group_end:`

Sets symbol unit macros to generate text directly. This is needed in expansion contexts where units must be converted to simple text. This function is itself not expandable, so must be using within a surrounding group as show in the example.

### 5 Pre-defined symbolic unit components

The unit parser is defined to recognise a number of pre-defined units, prefixes and powers, and also interpret a small selection of “generic” symbolic parts.

Broadly, the pre-defined units are those defined by the BIPM in the documentation for the *International System of Units* (SI) [1]. As far as possible, the names given to the command names for units are those used by the BIPM, omitting spaces and using only ASCII characters. The standard symbols are also taken from the same documentation. In the following documentation, the order of the description of units broadly follows the SI Brochure.
The base units as defined in the SI Brochure [2]. Notice that \texttt{meter} is defined as an alias for \texttt{metre} as the former spelling is common in the US (although the latter is the official spelling).

\texttt{kilogram} is defined using an SI prefix: as such the (derived) unit \texttt{gram} is required by the module to correctly produce output for the \texttt{kilogram}.

Prefixes, all of which are integer powers of 10: the powers are stored internally by the module and can be used for conversion from prefixes to their numerical equivalent. These prefixes are documented in Section 3.1 of the SI Brochure.

Note that the \texttt{kilo} prefix is required to define the base \texttt{kilogram} unit. Also note the two spellings available for \texttt{deca/deka}.  

23
The defined SI units with defined names and symbols, as given in Table 4 of the SI Brochure. Notice that the names of the units are lower case with the exception of \degreeCelsius, and that this unit name includes “degree”.

Units accepted for use with the SI: here \minute is a unit of time not of plane angle. These units are taken from Table 8 of the SI Brochure.

For the unit \litre, both l and L are listed as acceptable symbols: the latter is the standard setting of the module. The alternative spelling \liter is also given for this unit for US users (as with \metre, the official spelling is “re”).

Units for plane angles accepted for use with the SI: to avoid a clash with units for time, here \arcminute and \arcsecond are used in place of \minute and \second. These units are taken from Table 8 of the SI Brochure.

The mathematical concept of percent, usable with the SI as detailed in Section 5.4.7 of the SI Brochure.

Pre-defined unit powers which apply to the next \langle prefix ⟩ \langle unit ⟩ combination.
Pre-defined unit powers which apply to the preceding \( \text{prefix}/\text{unit} \) combination.

\( \backslash \text{per} \ ( \text{prefix} \  \text{unit} \  \text{power} ) \)

Indicates that the next \( \text{prefix}/\text{unit}/\text{power} \) combination is reciprocal, i.e. raises it to the power \(-1\). This symbolic representation may be applied in addition to a \backslash power, and will work correctly if the \backslash power itself is negative. In literal mode \backslash per will print a slash (“/”).

\( \backslash \text{cancel} \ ( \text{prefix} \  \text{unit} \  \text{power} ) \)

Indicates that the next \( \text{prefix}/\text{unit}/\text{power} \) combination should be “cancelled out”. In the parsed output, the entire unit combination will be given as the argument to a function \backslash cancel, which is assumed to be available at a higher level. In literal mode, the same higher-level \backslash cancel will be applied to the next token. It is the responsibility of the calling code to provide an appropriate definition for \backslash cancel outside of the scope of the unit parser.

\( \backslash \text{highlight} \  \text{color} \ ( \text{prefix} \  \text{unit} \  \text{power} ) \)

Indicates that the \text{qualifier} applies to the current \text{prefix}/\text{unit}/\text{power} combination. In parsed mode, the display of the result will depend upon module options. In literal mode, the \text{qualifier} will be printed in parentheses following the preceding \text{unit} and a full-width space.

\( \backslash \text{raiseto} \  \text{power} \ ( \text{prefix} \  \text{unit} ) \)
\( \backslash \text{tothe} \ \text{power} \ ( \text{prefix} \  \text{unit} ) \)

Indicates that the \text{power} applies to the current \text{prefix}/\text{unit} combination. As shown, \backslash raiseto applies to the next \text{unit} whereas \backslash tothe applies to the preceding unit. In literal mode the \backslash power will be printed as a superscript attached to the next token (\backslash raiseto or \backslash tothe) as appropriate.

\section{5.1 Key–value options}

The options defined by this submodule are available within the \text{l3keys siunitx} tree.

\text{bracket-unit-denominator} \quad \text{bracket-unit-denominator = true|false}

Switch to determine whether brackets are added to the denominator part of a unit when printed using inline fractional form (with \text{per-mode} as \text{repeated-symbol}, \text{symbol} or \text{symbol-or-fraction}). The standard setting is \text{true}.
**extract-mass-in-kilograms**

`extract-mass-in-kilograms = true|false`

Determines whether prefix extraction treats kilograms as a base unit; when set `false`, grams are used. The standard setting is `true`.

**forbid-literal-units**

`forbid-literal-units = true|false`

Switch which determines if literal units are allowed when parsing is active; does not apply when `parse-units` is `false`.

**fraction-command**

`fraction-command = ⟨command⟩`

Command used to create fractional output when `per-mode` is set to `fraction`. The standard setting is \(\frac\).

**inter-unit-product**

`inter-unit-product = ⟨separator⟩`

Inserted between unit combinations in parsed mode, and used to replace . and ~ in literal mode. The standard setting is \(\,\).

**parse-units**

`parse-units = true|false`

Determines whether parsing of unit symbols is attempted or literal mode is used directly. The standard setting is `true`.

**per-mode**

`per-mode = fraction|power|power-positive-first|repeated-symbol|symbol|symbol-or-fraction`

Selects how the negative powers (\(\per\)) are formatted: a choice from the options `fraction`, `power`, `power-positive-first`, `repeated-symbol`, `symbol` and `symbol-or-fraction`. The option `fraction` generates fractional output when appropriate using the command specified by the `fraction-command` option. The setting `power` uses reciprocal powers leaving the units in the order of input, while `power-positive-first` uses the same display format but sorts units such that the positive powers come before negative ones. The `symbol` setting uses a symbol (specified by `per-symbol`) between positive and negative powers, while `repeated-symbol` uses the same symbol but places it before every unit with a negative power (this is mathematically “wrong” but often seen in real work). Finally, `symbol-or-fraction` acts like `symbol` for inline output and like `fraction` when the output is used in a display math environment. The standard setting is `power`.

**per-symbol**

`per-symbol = ⟨symbol⟩`

Specifies the symbol to be used to denote negative powers when the option `per-mode` is set to `repeated-symbol`, `symbol` or `symbol-or-fraction`. The standard setting is `/`.

**qualifier-mode**

`qualifier-mode = bracket|combine|phrase|subscript`

Selects how qualifiers are formatted: a choice from the options `bracket`, `combine`, `phrase` and `subscript`. The option `bracket` wraps the qualifier in parenthesis, `combine` joins the qualifier with the unit directly, `phrase` joins the material using `qualifier-phrase` as a link, and `subscript` formats the qualifier as a subscript. The standard setting is `subscript`.

**qualifier-phrase**

`qualifier-phrase = ⟨phrase⟩`

Defines the `⟨phrase⟩` used when `qualifier-mode` is set to `phrase`. 

26
**sticky-per**  \[\text{sticky-per} = \text{true|false}\]

Used to determine whether \texttt{\per} should be applied one a unit-by-unit basis (when \texttt{false}) or should apply to all following units (when \texttt{true}). The latter mode is somewhat akin conceptually to the \LaTeX \texttt{\over} primitive. The standard setting is \texttt{false}.

**unit-font-command**  \[\text{unit-font-command} = (\text{command})\]

Command applied to text during output of units: should be command usable in math mode for font selection. Notice that in a typical unit this does not (necessarily) apply to all output, for example powers or brackets. The standard setting is \texttt{\mathrm}.

## References


Abbreviations for currents.
\[A\]  \[pA\]  \[nA\]  \[uA\]  \[mA\]  \[kA\]

Abbreviations for masses.
\[fg\]  \[pg\]  \[ng\]  \[ug\]  \[mg\]  \[g\]  \[kg\]

Abbreviations for temperature.
\[K\]

Abbreviations for lengths.
\[m\]  \[pm\]  \[nm\]  \[um\]  \[mm\]  \[cm\]  \[dm\]  \[km\]

Abbreviations for times.
\[s\]  \[as\]  \[fs\]  \[ps\]  \[ns\]  \[us\]  \[ms\]

Abbreviations for frequencies.
\[Hz\]  \[mHz\]  \[kHz\]  \[MHz\]  \[GHz\]  \[THz\]
Abbreviations for moles.
\mol
\fmol
\pmol
\nmol
\umol
\mmol
\kmol

Abbreviations for potentials.
\V
\pV
\nV
\uV
\mV
\kV

Abbreviations for volumes.
\hl
\l
\ml
\ul
\hL
\L
\mL
\uL

Abbreviations for powers.
\W
\uW
\mW
\kW
\MW
\GW

Abbreviations for energies.
\kJ
\J
\mJ
\uJ
\eV
\meV
\keV
\MeV
\GeV
\TeV

Abbreviations for forces.
\N
\mN
\kN
\MN
Abbreviations for pressures.
\(\text{Pa}\)
\(\text{kPa}\)
\(\text{MPa}\)
\(\text{GPa}\)

Abbreviations for resistance.
\(\text{mohm}\)
\(\text{kohm}\)
\(\text{Mohm}\)

Abbreviations for capacitance.
\(\text{F}\)
\(\text{fF}\)
\(\text{pF}\)
\(\text{nF}\)
\(\text{uF}\)

Abbreviation for decibel.
\(\text{dB}\)

Abbreviation for kilowatt–hours.
\(\text{kWh}\)
Part XI

siunitx-binary – Binary units

This submodule provides binary units and prefixes. These are not formally part of the SI but are recommended by BIPM as units of information.

\kibi \mebi \gibi \tebi \pebi \exbi \zebi \yobi

Prefixes, all of which are integer powers of 2: the powers are not stored or available for conversion.

\bit \byte

Units for bits and bytes.
Part XII

\textbf{siunitx-command} – Units as document command

This submodule provides support for creating free-standing document commands for unit macros.

1 Creating units as document commands

\texttt{\textbackslash siunitx\_command\_create:}

Maps over the list of known unit commands and creates the appropriate document command to support them, as controlled by the options below.

1.1 Key–value options

The options defined by this submodule are available within the \texttt{l3keys siunitx} tree. These options are all preamble-only.

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>free-standing-units</td>
<td>false</td>
<td>Switch to determine whether free standing document commands are created for symbolic units. This will include not only units themselves but also prefixes, etc. The standard setting is false.</td>
</tr>
<tr>
<td>overwrite-commands</td>
<td>false</td>
<td>Switch to determine whether when creating free standing document commands, any existing document commands are overwritten. The standard setting is false.</td>
</tr>
<tr>
<td>space-before-unit</td>
<td>false</td>
<td>Switch to determine whether a space is inserted before free standing document commands. The standard setting is false.</td>
</tr>
<tr>
<td>unit-optional-argument</td>
<td>false</td>
<td>Switch to determine whether free standing document commands take an optional argument (a number). The standard setting is false.</td>
</tr>
<tr>
<td>use-xspace</td>
<td>false</td>
<td>Switch to determine whether free standing document commands use the \texttt{xparse} package to insert space after the command names. The standard setting is false. When set true, the \texttt{xparse} package will be loaded at the start of the document if not already available.</td>
</tr>
</tbody>
</table>
Part XIII

\texttt{siunitx-emulation} – Emulation
## Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

<table>
<thead>
<tr>
<th>Symbols</th>
<th>( A )</th>
<th>( B )</th>
<th>( C )</th>
<th>( D )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( , )</td>
<td>15, 19</td>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>( \backslash A )</td>
<td></td>
<td></td>
<td></td>
<td>28</td>
</tr>
<tr>
<td>allow-quantity-breaks</td>
<td>28</td>
<td></td>
<td></td>
<td>12</td>
</tr>
<tr>
<td>( \backslash ampere )</td>
<td>23</td>
<td></td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>angle-mode</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>angle-symbol-degree</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>angle-symbol-minute</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>angle-symbol-over-decimal</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>angle-symbol-second</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>arc-separator</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>( \backslash arcminute )</td>
<td>24</td>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>( \backslash arcsecond )</td>
<td>24</td>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>( \backslash as )</td>
<td>28</td>
<td></td>
<td></td>
<td>28</td>
</tr>
<tr>
<td>( \backslash astronomicalunit )</td>
<td>24</td>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>( \backslash atto )</td>
<td>23</td>
<td></td>
<td></td>
<td>23</td>
</tr>
<tr>
<td>( \backslash becquerel )</td>
<td>24</td>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>( \backslash bel )</td>
<td>24</td>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>( \backslash bfseries )</td>
<td>12</td>
<td></td>
<td></td>
<td>12</td>
</tr>
<tr>
<td>( \backslash bit )</td>
<td>31</td>
<td></td>
<td></td>
<td>31</td>
</tr>
<tr>
<td>( \backslash boldmath )</td>
<td>13</td>
<td></td>
<td></td>
<td>13</td>
</tr>
<tr>
<td>bracket-ambiguous-numbers</td>
<td>8</td>
<td></td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>bracket-negative-numbers</td>
<td>8</td>
<td></td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>bracket-unit-denominator</td>
<td>25</td>
<td></td>
<td></td>
<td>25</td>
</tr>
<tr>
<td>( \backslash byte )</td>
<td>31</td>
<td></td>
<td></td>
<td>31</td>
</tr>
<tr>
<td>( \backslash cancel )</td>
<td>19, 25</td>
<td></td>
<td></td>
<td>19, 25</td>
</tr>
<tr>
<td>( \backslash candela )</td>
<td>23</td>
<td></td>
<td></td>
<td>23</td>
</tr>
<tr>
<td>( \backslash cm )</td>
<td>28</td>
<td></td>
<td></td>
<td>28</td>
</tr>
<tr>
<td>( \backslash color )</td>
<td>6</td>
<td></td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>( \text{color} )</td>
<td>13</td>
<td></td>
<td></td>
<td>13</td>
</tr>
<tr>
<td>compound-exponents</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>compound-final-separator</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>compound-pair-separator</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>compound-separator</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>compound-separator-mode</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>compound-units</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>( \backslash coulomb )</td>
<td>24</td>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>( \backslash cubed )</td>
<td>25</td>
<td></td>
<td></td>
<td>25</td>
</tr>
<tr>
<td>( \backslash cubic )</td>
<td>24</td>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>( \backslash dalton )</td>
<td>24</td>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>( \backslash day )</td>
<td>24</td>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>( \backslash dB )</td>
<td></td>
<td></td>
<td></td>
<td>12</td>
</tr>
<tr>
<td>( \backslash deci )</td>
<td>23</td>
<td></td>
<td></td>
<td>23</td>
</tr>
<tr>
<td>( \text{decibel} )</td>
<td>21</td>
<td></td>
<td></td>
<td>21</td>
</tr>
<tr>
<td>( \text{degree} )</td>
<td>21</td>
<td></td>
<td></td>
<td>21</td>
</tr>
<tr>
<td>( \text{degreeCelsius} )</td>
<td>21</td>
<td></td>
<td></td>
<td>21</td>
</tr>
<tr>
<td>( \backslash degree )</td>
<td>23</td>
<td></td>
<td></td>
<td>23</td>
</tr>
<tr>
<td>( \backslash dm )</td>
<td>28</td>
<td></td>
<td></td>
<td>28</td>
</tr>
<tr>
<td>drop-exponent</td>
<td>8</td>
<td></td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>drop-uncertainty</td>
<td>8</td>
<td></td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>drop-zero-decimal</td>
<td>8</td>
<td></td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>( \backslash electronvolt )</td>
<td>24</td>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>( \backslash ensuremath )</td>
<td>7, 12</td>
<td></td>
<td></td>
<td>29</td>
</tr>
<tr>
<td>( \backslash eV )</td>
<td>29</td>
<td></td>
<td></td>
<td>29</td>
</tr>
<tr>
<td>evaluate-expression</td>
<td>9</td>
<td></td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>( \backslash exa )</td>
<td>23</td>
<td></td>
<td></td>
<td>23</td>
</tr>
<tr>
<td>( \backslash exbi )</td>
<td>31</td>
<td></td>
<td></td>
<td>31</td>
</tr>
<tr>
<td>( \text{exponent-base} )</td>
<td>9</td>
<td></td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>( \text{exponent-mode} )</td>
<td>9</td>
<td></td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>( \text{exponent-product} )</td>
<td>9</td>
<td></td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>expression</td>
<td>9</td>
<td></td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>extract-mass-in-kilograms</td>
<td>26</td>
<td></td>
<td></td>
<td>26</td>
</tr>
<tr>
<td>( \backslash F )</td>
<td>30</td>
<td></td>
<td></td>
<td>30</td>
</tr>
<tr>
<td>( \text{familydefault} )</td>
<td>12</td>
<td></td>
<td></td>
<td>12</td>
</tr>
<tr>
<td>( \backslash farad )</td>
<td>24</td>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>( \text{femto} )</td>
<td>23</td>
<td></td>
<td></td>
<td>23</td>
</tr>
<tr>
<td>( \backslash fg )</td>
<td>30</td>
<td></td>
<td></td>
<td>30</td>
</tr>
<tr>
<td>( \text{fill-angle-degrees} )</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>( \text{fill-angle-minutes} )</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>( \text{fill-angle-seconds} )</td>
<td>2</td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>( \text{fixed-exponent} )</td>
<td>9</td>
<td></td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>( \text{f mol} )</td>
<td>29</td>
<td></td>
<td></td>
<td>29</td>
</tr>
<tr>
<td>( \text{fontfamily} )</td>
<td>12</td>
<td></td>
<td></td>
<td>12</td>
</tr>
<tr>
<td>( \text{fontseries} )</td>
<td>12</td>
<td></td>
<td></td>
<td>12</td>
</tr>
<tr>
<td>( \text{fontshape} )</td>
<td>12</td>
<td></td>
<td></td>
<td>12</td>
</tr>
<tr>
<td>( \text{forbid-literal-units} )</td>
<td>26</td>
<td></td>
<td></td>
<td>26</td>
</tr>
<tr>
<td>( \text{fp commands:} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \text{\textbackslash t m p a } _ \text{fp} )</td>
<td>20</td>
<td></td>
<td></td>
<td>20</td>
</tr>
<tr>
<td>( \text{\textbackslash frac} )</td>
<td>19</td>
<td></td>
<td></td>
<td>19</td>
</tr>
<tr>
<td>( \text{fraction-command} )</td>
<td>26</td>
<td></td>
<td></td>
<td>26</td>
</tr>
<tr>
<td>( \text{free-standing-units} )</td>
<td>32</td>
<td></td>
<td></td>
<td>32</td>
</tr>
<tr>
<td>( \text{\textbackslash fs} )</td>
<td>28</td>
<td></td>
<td></td>
<td>28</td>
</tr>
</tbody>
</table>

34
| \mmol                  | 29 |
| \MN                   | 29 |
| \mN                   | 29 |
| \MPa                  | 30 |
| \Mole                 | 23 |
| \mp                   | 6  |
| \mohm                 | 30 |
| \mohm                 | 30 |
| \mol                  | 29 |
| \mole                 | 23 |
| \N                    | 29 |
| \nA                   | 28 |
| \nano                 | 23 |
| \neper                | 24 |
| \newton              | 24 |
| \nF                   | 30 |
| \ng                   | 28 |
| \nm                   | 28 |
| \nmol                 | 28 |
| \number-angle-product | 29 |
| \number-color         | 13 |
| \number-mode          | 29 |
| \nV                   | 29 |
| \OA                   | 25 |
| \ohm                  | 24 |
| \over                 | 27 |
| overwrite-commands   | 32 |
| \Pa                   | 30 |
| \Pa                    | 28 |
| parse-numbers         | 10 |
| parse-units           | 26 |
| \pascal               | 24 |
| \pebi                 | 31 |
| \per                  | 25-27 |
| \per-mode             | 26 |
| \per-symbol           | 26 |
| \per                    | 24 |
| \Peta                 | 23 |
| \PF                   | 30 |
| \Q                    | 28 |
| \radian               | 24 |
| \raiseeto             | 25 |
| \range-exponents      | 4  |
| \range-mode           | 4  |
| \range-units          | 4  |
| \rmfamily             | 13 |
| \round-half           | 13 |
| \round-minimum        | 13 |
| \round-mode           | 13 |
| \round-pad            | 13 |
| \round-precision      | 10 |
| \s                    | 28 |
| \second               | 23, 24 |
| \selectfont           | 12 |
| \separate-uncertainty-units | 15 |
| \series-version-mapping | 14 |
| \seriesdefault        | 12 |
| \shapedefault         | 12 |
| \siemens              | 24 |
| \siemendefinition     | 24 |
| \siunitx              | 1 |
| \siunitx_angle:n      | 1 |

<p>| \s                    | 28 |
| \second               | 23, 24 |
| \selectfont           | 12 |
| \separate-uncertainty-units | 15 |
| \series-version-mapping | 14 |
| \seriesdefault        | 12 |
| \shapedefault         | 12 |
| \siemens              | 24 |
| \siemendefinition     | 24 |
| \siunitx              | 1 |
| \siunitx_angle:n      | 1 |</p>
<table>
<thead>
<tr>
<th>Command</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>\	extminus</td>
<td>12</td>
</tr>
<tr>
<td>\textpm</td>
<td>12</td>
</tr>
<tr>
<td>\textsubscript</td>
<td>12</td>
</tr>
<tr>
<td>\textsuperscript</td>
<td>12</td>
</tr>
<tr>
<td>\texttimes</td>
<td>12</td>
</tr>
<tr>
<td>\texttimes</td>
<td>28</td>
</tr>
<tr>
<td>\texttimes</td>
<td>28</td>
</tr>
<tr>
<td>\texttimes</td>
<td>28</td>
</tr>
<tr>
<td>\texttimes</td>
<td>28</td>
</tr>
<tr>
<td>tight-spacing</td>
<td>10</td>
</tr>
<tr>
<td>\times</td>
<td>6</td>
</tr>
<tr>
<td>\l_tmpa_tl</td>
<td>19, 20</td>
</tr>
<tr>
<td>\tonne</td>
<td>24</td>
</tr>
<tr>
<td>\tothe</td>
<td>25</td>
</tr>
<tr>
<td>U</td>
<td></td>
</tr>
<tr>
<td>\uA</td>
<td>28</td>
</tr>
<tr>
<td>\uF</td>
<td>30</td>
</tr>
<tr>
<td>\ug</td>
<td>28</td>
</tr>
<tr>
<td>\um</td>
<td>29</td>
</tr>
<tr>
<td>\ul</td>
<td>29</td>
</tr>
<tr>
<td>\us</td>
<td>29</td>
</tr>
<tr>
<td>\umol</td>
<td>29</td>
</tr>
<tr>
<td>uncertainty-mode</td>
<td>11</td>
</tr>
<tr>
<td>uncertainty-separator</td>
<td>11</td>
</tr>
<tr>
<td>unit-color</td>
<td>14</td>
</tr>
<tr>
<td>unit-font-command</td>
<td>27</td>
</tr>
<tr>
<td>unit-mode</td>
<td>14</td>
</tr>
<tr>
<td>unit-optional-argument</td>
<td>32</td>
</tr>
<tr>
<td>\upshape</td>
<td>13</td>
</tr>
<tr>
<td>use-xspace</td>
<td>32</td>
</tr>
<tr>
<td>\uV</td>
<td>29</td>
</tr>
<tr>
<td>\volt</td>
<td>24</td>
</tr>
<tr>
<td>\w</td>
<td></td>
</tr>
<tr>
<td>\V</td>
<td>29</td>
</tr>
<tr>
<td>\W</td>
<td>29</td>
</tr>
<tr>
<td>\yoobi</td>
<td>31</td>
</tr>
<tr>
<td>\yocto</td>
<td>23</td>
</tr>
<tr>
<td>\yotta</td>
<td>23</td>
</tr>
<tr>
<td>\zebi</td>
<td>31</td>
</tr>
<tr>
<td>\zepto</td>
<td>23</td>
</tr>
<tr>
<td>\zetta</td>
<td>23</td>
</tr>
</tbody>
</table>