The engsymbols package*

Fábio Fortkamp
fabio@fabiofortkamp.com

December 4, 2014

1 Introduction

This document describes the engsymbols package, a collection of macros to facilitate the writing of common engineering symbols.

The following packages are prerequisites:

- siunitx

This package follows the conventions specified by ISO standards of typesetting mathematics [1].

engsymbols is actually just a collection of commands I, as a Ph.D. student in Mechanical Engineering, find useful, and I hope other can find it to. There isn't any special design principles.

2 Implementation

2.1 Basic operations

\ped \ap These macros by [1] typesets the argument in math roman font, to indicate a object. Italic subscripts should be used only to refer to another variables, for example, c_P is the specific heat obtained by mantaining the pressure, a physical parameter, fixes. By contrast, h_{L} (produced by `h\ped{L}`) is the liquid enthalpy; liquid is not a variable. The command `\ap{`⟨*index*⟩`}` does the same to superscripts, like T^{I} for the interface temperature.

```
\newcommand{\ped}[1]{\ensuremath{_{\mathrm{#1}}}}
\newcommand{\ap}[1]{\ensuremath{^{\mathrm{#1}}}}
```

\nvector \nmatrix We define vector and matrix commands according to ISO standards: bold italic for vectors ($\boldsymbol{x}$) and matrices ($\boldsymbol{A}$). The "n" in names stands for "notation". This requires the bm package.

```
\newcommand{\nvector}[1]{\bm{#1}}
\newcommand{\nmatrix}[1]{\bm{#1}}
```

*This document corresponds to engsymbols v0.1, dated 2014/12/02.

2.2 Special individual symbols

`\volume` This macro produces a calligraphic V to indicate volume, as $\mathcal{V}$. This is usually done to avoid confusion with velocity.

```
\newcommand{\volume}{\mathcal{V}}
```

`\diffd` This macro produces the differential d operator, as in $\mathrm{d}x$. The definition is fairly complex beacuse it tries to do an optimal spacing, and is described by [1].

```
\newcommand{\diffd}{\@ifnextchar^{\DIfF}{\DIfF^{}}}
\def\DIfF^#1{%
  \mathop{\mathrm{\mathstrut d}}%
      \nolimits^{#1}\gobblespace}
\def\gobblespace{%
  \futurelet\diffarg\opspace}
\def\opspace{%
  \let\DiffSpace\!%
  \ifx\diffarg(%
      \let\DiffSpace\relax
  \else
      \ifx\diffarg[%
          \let\DiffSpace\relax
      \else
          \ifx\diffarg\{%
              \let\DiffSpace\relax
          \fi\fi\fi\DiffSpace}
```

`\hheat`
`\hmass` These macros produces a “crossed” h as in $\hbar$. This is done in some texts to denote the convection heat transfer coefficient and differentiate it from enthalpy h. This is actually just an alias to the existing command `\hbar`, to give a more meaningful name. There is also `\hmass` to produce $\hbar_{\mathrm{m}}$, used to indicate a mass transfer coefficient.

```
\newcommand{\hheat}{\hbar}
\newcommand{\hmass}{\hbar\ped{m}}
```

`\universalgasconstant` A simple command to produce R_{u}

```
\newcommand{\universalgasconstant}{R\ped{u}}
```

`\diffusivitybinary` This is a shorthand for the diffusivity of a binary mixture, $\mathcal{D}_{12}$.

```
\newcommand{\diffusivitybinary}{\mathcal{D}_{12}}
```

2.3 Common operations

`\average` This command puts a line above the argument (like $\overline{x}$), a notation widely used to indicate some type of average.

```
\newcommand{\average}[1]{\overline{#1}}
```

`\rate` This macro denotes the rate of something, like $\dot{m}$ for a mass flow rate.

```
\newcommand{\rate}[1]{\dot{#1}}
```

\flux

Produces q''.

```
\newcommand{\flux}[1]{{#1}''}
```

\divergent
\divergentpar
divergentn

These two macros produce the diverget of a vector $\nabla \cdot \boldsymbol{q}$. The `par` variant automatically adds parentheses, useful for multiple arguments like $\nabla \cdot (\rho \boldsymbol{V})$ (produced with `\divergentpar{\rho \nvector}`). The `\divergentv` command automatically converts the argument to a vector

```
\newcommand{\divergent}[1]{\nabla \cdot #1}
\newcommand{\divergentv}[1]{\divergent{\nvector{#1}}}
\newcommand{\divergentpar}[1]{\divergent{\left( #1 \right)}}
```

\gradient
\gradientpar

Gradient of a scalar ∇T. The `par` variant introduces parentheses (e.g. $\nabla\left(\frac{\rho_1}{\rho}\right)$.

```
\newcommand{\gradient}[1]{\nabla {#1}}
\newcommand{\gradientpar}[1]{\gradient{\left( {#1} \right)}}
```

\laplacian
\laplacianpar

The laplacian of a scalar x is defined as $\nabla^2 x = \nabla \cdot \nabla x$. One could also use $\nabla^2 \left(\rho c_p T\right)$.

```
\newcommand{\laplacian}[1]{\nabla^2 #1}
\newcommand{\laplacianpar}[1]{\laplacian{\left( #1 \right)}}
```

References

[1] Claudio Beccari. Typesetting mathematics for science and technology according to iso 31/xi. *TUGboat*, 18(1):39–48, 1997.