

The engsymbols package*

Fábio Fortkamp
fabio@fabiofortkamp.com

December 5, 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:

- `bm`
- `amsmath`
- `esdiff`

This package follows the conventions specified by ISO standards of typesetting mathematics [1]. The main points of this convention are:

- variables are always written in italic, *and also* in bold for vector and matrices
- in other words, vectors don't have arrows above it
- subscripts and superscripts are written in roman upright font when they do not indicate a variable per se; e.g. when they refer to a physical object, a fluid phase, a state etc

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

- Command names are not short, but are descriptive. The user can always define an alias, and many editors can parse the packages and offer tab completion.
- These macros are not intended to be the absolute true way of scientific writing. As I said, I try to follow some standards and observe what is commonly used.

Please notice that the user should refer to other references such as papers and textbooks to get the meaning of the symbols I describe here.

*This document corresponds to `engsymbols` v0.1, dated 2014/12/05.

2 Available commands

2.1 Special symbols

First we have some basic symbols, commands that take no arguments.

`\volume` This `\volume` macro typesets a special symbol \mathcal{V} to indicate the total volume of something. The more logical V is avoided, mainly in fluid mechanics texts, because it's used for velocity. I chose the calligraphic \mathcal{V} because the often used \forall is really ugly and is intended to use with logical propositions ("for all $x \dots$ ").

`\hheat` The symbols \hbar (`\hheat`) and \hbar_m (`\hmass`) are used in some textbooks to indicate the convection heat (or mass) transfer coefficient and to avoid confusion with enthalpy h . I'm aware that \hbar is also used for the reduced Planck constant, but I don't think it's used in engineering literature. I also know that the distinction between an enthalpy value and a heat transfer coefficient value can be inferred by the context, but in a equation with many terms, including terms with enthalpy and heat transfer rates, this can make things clearer. Also notice the m subscript in roman fonts; it's a coefficient for mass transfer, and not a coefficient for a variable m .

`\diffusivitybinary` I deal with mixtures a lot, so there is `\diffusivitybinary` \mathcal{D}_{12} , which is used to indicate the mass diffusivity of a binary mixture.

`\universalgasconstant` Finally, there is a command `\universalgasconstant` to write R_u , the universal gas constant. The u index (also in roman font) marks this as a constant; each fluid has its own "gas constant", defined as $R = \frac{R_u}{M}$, where M is the molar mass; hence for a mixture, this gas constant varies with the composition, and of course different fluids have different R . R_u is universal for all fluids.

2.2 Auxiliary commands

`\ped` These macros are defined and explained in [1], and produce a subscript and superscript in roman font; this is probably the most confusing command. `h\ped{L}` produces h_L , the enthalpy of the liquid phase, while `e_L` produces e_L , for example, the error in a *variable* L . These commands can also be written in text mode, like `2\ap{nd}` to write 2^{nd} .

2.3 Vector commands

`\nvector` The command `\nvector{<arg>}` typesets $\{\langle arg \rangle\}$ in vector form, like \mathbf{x} and ϕ .
`\nmatrix` There is also a matrix command. The "n" indicates notation, to avoid defining something as general as `\vector`.

`\vectornorm` The other commands are used in vector operations. The vector norm $\|a\|$ is written as `\vectornorm{\nvector{a}}` and the divergent $\nabla \cdot \mathbf{V}$ as `\divergent{\nvector{V}}`. The gradient of a scalar ∇T (`\gradient{T}`) transforms a scalar into a vector (which is why I included it into this group of "vector operations"), and the laplacian $\nabla^2 x$, with many applications in differential equations, is defined as `\laplacian{x} = \divergent{\gradient{x}}` (try to understand this definition and to imagine how it would be typeset).
`\laplacian`

`\divergentpar` The `\laplacian`, `\divergent` and `\gradient` also have `par` variants to automatically add parentheses; for example, `\divergentpar{\rho \nvector{V}}` is the term $\nabla \cdot (\rho \mathbf{V})$ widely used in fluid mechanics.

`\gradientpar`

`\laplacianpar`

`\divergentv` Finally, for the divergent, there is a command `\divergentv{q}` that automatically wraps the argument into vector form, as in $\nabla \cdot \mathbf{q}$.

2.4 Other operations

`\rate` The command `\rate{m}` produces \dot{m} , which can be understood as a “mass flow rate”. This is just an alias for `\dot`, and is actually harder to type, but it’s more meaningful. Similarly, there is `\flux{q}` to write the heat flux q'' .

`\flux` The macro `\average{f}` results in \overline{f} . I’ve seen some definitions that shortens the bar above the argument, but I think this length is just fine.

`\average`

`\diffpar` Sometimes I want to write the derivative of a product; I really like the `\diff` command from the `esdiff` package, but $\frac{d\rho V}{dx}$ is a little ugly for my taste. I prefer something like $\frac{d}{dx}(\rho V)$: the numerator in the derivative is empty, and the argument, the function to be derived, is written right after the fraction with parentheses around it (this expression was written as `\diffpar{\rho V}{x}`). There is also

`\diffppar` the `\diffppar{\rho V}{x}` variant, resulting in $\frac{\partial}{\partial x}(\rho V)$.

References

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