The `ushort` package*

Martin Väth†

2013/11/26

Abstract

With this package you may write shorter (or longer) underlines/underbars. This is particularly useful for 'underlined' variables which are needed by some engineers.

Contents

1 Changes

v2.2 (*2013/11/26*) Update email, add license. Date/version remains the same since only commments and documentation are modified.

v2.2 (*2001/06/13*) As Alessandro Russo `<russo@ian.pv.cnr.it>` suggested, the case of arguments consisting of more than a single character is now documented. Corresponding macros `\ushortw` and `\ushortdw` have been added.

*This package has version number 2.2, last revised 2001/06/13. The package may be distributed and/or modified under the conditions of the LaTeX Project Public License, either version 1.3c of this license or (at your option) any later version. The latest version of this license is in `http://www.latex-project.org/lppl.txt`, and version 1.3c or later is part of all distributions of LaTeX version 2005/12/01 or later.

†`martin@mvath.de`

v2.1 (*2001/06/07*) Improved speed dramatically (up to factor 4) by not using `\phantom` anymore. Unfortunately, if bad hacks had been used for the argument "Additional calculations" in `\ushortCreate`, they could fail to work now. For this reason, the major release number has been changed.

Further internal change: In `\ushrt@Main` the arguments are now expanded by a loop.

v1.4 (*2001/04/06*) Set defaults for the parameters as documented.

v1.3 (*2001/04/05*) Added the argument for additional calculations. Changed the default values of the arguments (and thus of `\ushort`).

v1.2 (*2001/04/04*) Removed a typing error in the scan macro and added the commands `\ushortd` and `\ushortdline`.

v1.1 (*2001/04/04*) Introduced clean version of `\ushortEnsuremath` (and eliminated the corresponding previous hack).

v1.0 (*2001/04/03*) First beta release.

2 Introduction

In the newsgroup `de.comp.text.tex` the question arose whether it is possible to define 'underlined' variables: Such variables are required for complex vectors by electrical engineers. The usual `\underline` command is not appropriate for this purpose, because the underlines of concatenated variables merge which is not desired. One may use the macro `\ushort` of this package instead.
However, this package also provides a generic macro which allows to create new commands which work similarly but use different parameters for the underlines. In particular, it is possible to create e.g. a macro which draws longer underlines than usual or which does a similar task with `\underbar`.

3 Installation

This package was tested with plain TEX, LATEX2.09, and LATEX 2_ε, and it should actually run with most other TEX formats.
To use "ushort", you have to put the file `ushort.sty` in a path where TEX looks for its input files. The TEX documents using ushort need the following modifications in their header:

- If you use LATEX 2_ε, put in the preamble the command

  ```
  \usepackage{ushort}
  ```

- If you use LATEX2.09, use `ushort` as a style option, e.g.

  ```
  \documentstyle[ushort]{article}
  ```

or

```
\documentstyle[ushort,12pt]{article}
```

- If you use some other (non-LaTeX) format, you will probably have to insert a line like

```
\catcode`\@=11\relax \input ushort.sty \catcode`\@=12\relax
```

For TeX insiders: The only LaTeX-specific commands used in `ushort.sty` are:

- `\newcommand` (used only in the form `\newcommand{`⟨*command*⟩`}{}` to ensure that ⟨*command*⟩ was not defined before)
- `\ProvidesPackage`
- `\typeout`

The above commands are used only if they are defined (for `\typeout` a reasonable substitute is used, otherwise).

4 Description of the provided macros

This package provides the following macros.

`\ushort` `\ushort`: This command is used analogously to `\underline` with the difference that the underlines are slightly shorter than usual (and so, in particular, in concatenations of `\ushort`'ed variable names the underlines do not merge). The length of the underline is chosen under the assumption that the argument is a single character. If you want to underline a variable name which consists of more than one character, use instead the following command `\ushortw`.

`\ushortw` `\ushortw`: (Mnemonic: "ushort word"). This command is analogous to `\ushort` but such that the length of the underline is chosen appropriate for a 'long' argument. (This does not look as good as `\ushort` if the argument is a single character).

`\ushortd` `\ushortd`: (Mnemonic: "ushort double"). This is analogous to `\ushort` except that the underline is doubled.

`\ushortdw` `\ushortdw`: (Mnemonic: "ushortd word"). This is analogous to `\ushortd` except that the underline is doubled.

`\ushortdline` `\ushortdline`: This is analogous to `\underline` except that the underline is doubled. This command is internally used by `\ushortd` and `\ushortdw`, and thus redefinitions change also the behavior of `\ushortd` and `\ushortdw` correspondingly. In Plain TeX, this macro can be used only in math mode.

\ushortCreate

\ushortCreate: This can be used to generate a macro which behaves similarly to \ushort or \ushortd but for which certain parameters can be changed. For a detailed description, see below.

\ushortEnsuremath

\ushortEnsuremath: This is rather equivalent to the LaTeX 2_ε macro \ensuremath (but is available also e.g. in compatibility mode or if this package is used with plain TeX or LaTeX2.09).

Only \ushortCreate needs a detailed description. Its call syntax is:

\ushortCreate*:\⟨*command*⟩[⟨*summand*⟩]<⟨*factor before*⟩>
(⟨*factor after*⟩)+{⟨*additional calculations*⟩}{⟨*name*⟩}

All arguments except for the last one {⟨*name*⟩} are optional. Moreover, also the order of these arguments can be changed arbitrarily, and they can be repeated (all except for the last occurrence of each each type of argument is ignored). Only the last argument {⟨*name*⟩} can not be repeated and must be at the end.
This command creates a new macro \⟨*name*⟩. It is not tested whether \⟨*name*⟩ already exists; if it exists already, it is overridden. In particular, it is admissible to use \ushortCreate to redefine \ushort, \ushortw, \ushortd, or \ushortdw. This is the recommended method, if you want to change the default values.
The macro \⟨*name*⟩ can be used analogously to \underline, but the precise behavior depends on the arguments of \ushortCreate.
The meaning of the arguments is as follows:

* If the * is present, then the command \underbar is used (instead of the default \underline) to draw the underline. Moreover, the command \underbar is not used in math mode (while \underline is used in math mode).

:\⟨*command*⟩ Use \⟨*command*⟩ instead of the default \underline to draw the underline. Whether \⟨*command*⟩ is executed in math mode or not depends on whether * has been used.

[⟨*summand*⟩] Add ⟨*summand*⟩ to the length of the underline before drawing. Values like -.3\ht0 are admissible. ⟨*summand*⟩ is expanded only when \⟨*name*⟩ is expanded, and \wd0 or \ht0 give the width respectively height of the argument given to \name. In particular, also the value of the unit em depends on the mode in which \⟨*name*⟩ is called. The default is empty (which means 0pt but is faster).

<⟨*factor before*⟩> Multiply the length of the underline before adding ⟨*summand*⟩. The empty string means 1 (and costs no time). The default is empty.

(⟨*factor after*⟩) Multiply the length of the underline after adding ⟨*summand*⟩. The empty string means 1 (and costs no time). The default is 0.8.

+{⟨*Additional calculations*⟩} With this argument, you can introduce an arbitrary command sequence into the generated command: This sequence is executed (within a group) after ⟨*summand*⟩ was added, but before ⟨*factor after*⟩ is

applied. This sequence is meant to do additional calculation with `\dimen0` which is the length of the generated underline. When the calculations are carried out, `\box0` contains the content which is to be underlined. However, in contrast to previous releases of this package, modifications of this box are ignored. Instead, the registers `\dimen1` and `\dimen2` also contain the height and depth of this box – you have to modify these values to attain the desired effect.

Examples: The most important examples are the macros `\ushort`, `\ushortw`, `\ushortd`, and `\ushortdw` which have been created by

```
\ushortCreate{ushort}
\ushortCreate()[-.1em]{ushortw}
\ushortCreate:\ushortdline{ushortd}
\ushortCreate:\ushortdline()[-.1em]{ushortdw}
```

A further example demonstrating other parameters is

```
\ushortCreate*(3)(.5){uline}
```

This creates the macro `\uline` which underbars its argument but such that the underbar has only half of the usual length (the argument `(3)` in the above call is overridden by the later argument `(.5)`).
A more useful example is

```
\ushortCreate()[-.3\ht0]{ushort}
```

This modifies `\ushort` such that the underline is shrunken by .3 times the height of the argument of `\ushort`. If the argument is one letter, this means: The higher the letter, the shorter the underline. This looks good for italic fonts with some exceptions (like the letter `b`).