

# The *alphabet* package

Günter Milde

2020/10/30

## Abstract

The *alphabet* package makes the standard macros for Greek letters in mathematical mode also available in text mode. This way, you can input Greek letters “by name” everywhere in the document. The mode determines whether the characters are taken from the text or math font.

With 8-bit TeX and *greek-inputenc*, literal Unicode characters can also be used in mathematical mode.<sup>1</sup>

## 1 Requirements and Conflicts

The *alphabet* package depends on *textalpha* (both are part of *greek-fontenc*). It can be used under 8-bit TeX as well as XeTeX/LuaTeX (this document is typeset with 8-bit TeX and fontenc with TeX fonts). Depending on the converter and fonts, different [limitations](#) apply.

The package conflicts with *unicode-math*.

It also fails, if the `utf8x` input encoding is selected (interface to the *ucs* package using a non-compatible definition of `\DeclareUnicodeCharacter`).

## 2 Usage

Load this package in the preamble of your document (after font and math setup) with

```
\usepackage{alphabet}
```

Now you can write a single Greek symbol (like  $\Psi$  or  $\mu$ ) or a  $\lambda\gamma\omicron\varsigma$  in non-Greek text as well as ISO-conforming formulas with upright symbols for constants like  $A = \pi r^2$  (instead of  $A = \pi r^2$ ).<sup>2</sup>

Just like Latin letters, the Greek counterparts are by default italic in math mode<sup>3</sup> and upright in text:

---

<sup>1</sup>This document was compiled using the [font encoding](#) T1 (8-bit fonts). For a version using Unicode fonts, see [alphabet-tu.pdf](#).

<sup>2</sup>The [isomath documentation](#) describes more alternatives for upright Greek symbols in math mode.

<sup>3</sup>Capital Greek letters are upright in TeX unless a package selects the “ISO” math-style. See the [isomath documentation](#) for a detailed discussion of math-styles.

Text:  $L \Gamma l \gamma$ , mathematics:  $L \Gamma l \gamma$

See the source of this document [alphabetadoc.tex](#) for a setup and usage example.

## 2.1 options

Package options are passed to the *textalpha* package. Example call with options:

```
\usepackage[normalize-symbols,keep-semicolon]{alphabetadoc}
```

`normalize-symbols` merges “letter” and “symbol” variants of some Greek letters to the “letter” character:<sup>4</sup> Without this option, the symbol variant characters cannot be used in text, because they are not supported by 8-bit Greek fonts (LGR encoding). The `normalize-symbols` option is ignored, if you compile the document with XeTeX or LuaTeX using Unicode fonts. **Attention:** Be careful in cases where the distinction between the symbol variants may be important (e.g. in a mathematical or scientific context). Use XeTeX/LuaTeX with Unicode fonts or the respective characters in mathematical mode (e.g.  $\pi$  vs.  $\varpi$ ).

The option `keep-semicolon` prevents conversion of the semicolon to an *anoteleia* in 8-bit TeX (see *textalpha-doc*).

## 2.2 symbol variants

Mathematical notation uses variant shapes of some Greek letters as additional symbols. The variations have no syntactic meaning in Greek text and text fonts may use the variant shapes in place of the “regular” ones as a stylistic choice.

Unicode defines separate code points for the symbol variants. TeX supports some of the variant shape symbols in mathematical mode, but its concept of “standard” vs. “variant” symbols differs from the distinction between “GREEK LETTER ...” vs. “GREEK ... SYMBOL” in the Unicode standard.

The *alphabetadoc* package defines generic macros for these variants that are short forms of the set defined in `tuenc-greek.def` (cf. [test-tuenc-greek](#)):

```
\<name> selects the Unicode GREEK LETTER ... variant,  
\<name>symbol selects the Unicode GREEK ... SYMBOL variant,  
\var<name> selects the variant shape according to TeX' mathematical mode
```

See [Table 1](#) for the full list.

---

<sup>4</sup>The `normalize-symbols` option was added in version 0.13 (2015-08-03). Unicode input of the symbol variants requires at least version 1.6 (2015-08-05) of *greek-inputenc*.

### 3 Limitations

With 8-bit TeX, the limitations described in the [textalpha documentation](#) apply. See also the tests in section [8 bit limitations](#). These limitations are lifted, if the text language is switched to “greek” with Babel, the text part is wrapped in `\ensuregreek`, or the document is compiled with XeTeX/LuaTeX.

With XeTeX/LuaTeX and Unicode fonts, literal Unicode characters cannot be used in formulas (the log file reports missing characters) This is a generic TeX limitation which *alphabet* overcomes if used under 8-bit TeX. Under XeTeX/LuaTeX it may be lifted using the [unicode-math](#) package. However, *unicode-math* conflicts with *alphabet*.

### 4 Tests and examples

#### 4.1 Greek alphabet

Greek letters via generic “name” macros without language/font-encoding switch (active font encoding T1):

```
A B Γ Δ E Z H Θ I K Λ M N Ξ O Π P Σ T Υ Φ X Ψ Ω
α β γ δ ε ζ η θ ι κ λ μ ν ξ ο π ρ σ ς τ υ φ χ ψ ω
f F τ ς5 ι ι6 ϑ ϱ ϒ λ λ
```

Greek letters via Unicode (active font encoding T1):

```
A B Γ Δ E Z H Θ I K Λ M N Ξ O Π P Σ T Υ Φ X Ψ Ω
α β γ δ ε ζ η θ ι κ λ μ ν ξ ο π ρ σ ς τ υ φ χ ψ ω
f F τ ι ι ϑ ϱ ϒ λ λ
```

#### 4.2 Diacritics

Accent macros are set up for use with the generic macros by definition of “TextComposite” commands.

Diacritics (except the dialytika) should be placed before capital letters and dropped with `MakeUppercase`:

```
á ê í ñ ò ó ð
`A `E `I `H `O `Υ `Ω
A E Ĩ H O Υ Ω
```

---

<sup>5</sup>There is no separate Unicode code point for a stigma variant symbol, `\varstigma` is not defined with Xe/LuaTeX and similar to `\stigma` in some fonts.

<sup>6</sup>In LGR, there is no separate code point for Koppa as typographical practice normally does not observe a contrast between uppercase and lowercase forms for numeric koppa.

### 4.3 normalize-symbols

The `normalize-symbols` option merges “letters” and “symbol” variants of some Greek letters to the “letter” character. It is ignored, if the document uses Unicode fonts and is compiled with XeTeX or LuaTeX. (This document is compiled using 8-bit TeX.)

The source of this quote uses both variants for beta ( $\beta|\beta$ ), epsilon ( $\varepsilon|\varepsilon$ ), phi ( $\varphi|\varphi$ ), kappa ( $\chi|\chi$ ), pi ( $\pi|\pi$ ), rho ( $\varrho|\varrho$ ), theta ( $\theta|\theta$ ), and Theta ( $\Theta|\Theta$ ) in the LaTeX source.

### 4.4 Έλληνικά (Έλληνικά) in PDF strings

With the `alphabet` package, you get Greek letters in both, the document body and PDF metadata generated by `hyperref` if the input uses Unicode literals or macros. Wrapping in `\ensuregreek` ensures the right placement of the accents and breathings (before, not above capital letters). With LICR input (accent macros as well as symbol macros), non-standard diacritics are missing in the PDF data, as `hyperref`'s PU encoding currently does not support polytonic Greek. (Here, the dasia is dropped at the start of the word in parentheses in the PDF toc. The warning “Glyph not defined in PU encoding, removing ‘\<’ on input line 145.” is written to the log.)

### 4.5 Greek in maths $\Gamma = \sin \alpha / \cos \beta$

In the main document, Greek in math mode should work as usual:

$$\Gamma = \frac{\sin \alpha}{\cos \beta}.$$

Greek letters and symbols in math input as macro:<sup>7</sup>

$\Gamma \Delta \Theta \Lambda \Xi \Pi \Sigma \Upsilon \Phi \Psi \Omega$   
 $\alpha \beta \gamma \delta \epsilon \zeta \eta \theta \iota \kappa \lambda \mu \nu \xi \pi \rho \sigma \tau \upsilon \phi \chi \psi \omega$   
 $\vartheta \varpi \varrho \varphi \varepsilon$

PDF strings do not know math mode. The content of a formula or equation is evaluated in text mode with non-valid commands discarded (and warnings written to the log). This works reasonably well for simple formulas (but not, e.g., for super-/subscripts). With the `alphabet` package, it works also for Greek letters.

---

<sup>7</sup>There are no math macros for Greek letters which exist with similar shape in the Latin alphabet

## 4.6 Greek Unicode characters in math (only under 8-bit TeX)

With the `utf8` option of `inputenc` and `greek-inputenc`, literal Greek Unicode characters are supported also in mathematical mode.

$$\Gamma = \frac{\sin \alpha}{\cos \beta}.$$

Greek letters and symbols in math input as Unicode literals:

$\Gamma \Delta \Theta \Lambda \Xi \Pi \Sigma \Upsilon \Phi \Psi \Omega$   
 $\alpha \beta \gamma \delta \varepsilon \zeta \eta \theta \iota \kappa \lambda \mu \nu \xi \pi \rho \sigma \tau \upsilon \varphi \chi \psi \omega$   
 $\vartheta \phi \varpi F \varrho \epsilon$

This does not work with XeTeX/LuaTeX (unless in 8-bit emulation mode).

The “normal” vs. “variant” shape of letters is used so that the output matches the Unicode reference glyph (cf. Table 1). This corresponds to the behaviour of `unicode-math`.

## 4.7 8-bit limitations

Certain limitations apply if Greek symbols are used in non-Greek context.

- Composition of diacritics (like `\>\'`) fails:  $\acute{\alpha} \acute{\epsilon} \acute{\iota} \acute{\eta} \acute{o} \acute{\upsilon} \acute{\omega}$   
Simple diacritics and long names (like `\accdasiaoxia`) work in any font encoding, however they do not select precomposed characters (the difference becomes obvious if you drag-and-drop text from the PDF version of this document):  $\check{\alpha} \check{\alpha} \check{\alpha}$  (LFP) vs.  $\check{\alpha}$  (T1)
- `MakeUppercase` fails with composite diacritics in other font encodings.
- There is no kerning between Greek letters, if the font encoding does not support Greek: compare  $\text{ΑΥΑ}$  (LFP) to  $\text{ΑΥΑ}$  (T1).

The `\ensuregreek` macro ensures that the argument is typeset with a font encoding supporting Greek. This keeps kerning (if the kerning pair is inside the argument,  $\text{ΑΥΑ}$ ), and allows combining of accent macros where pre-composed characters are selected ( $\check{\alpha}$ ). Setting the correct language for Greek quotes with the `Babel` package additionally ensures correct hyphenation.

macro	text	math
<code>\beta</code>	$\beta$	$\beta$
<code>\varbeta</code>	$\beta$	$\text{\textcircled{\beta}}$
<code>\betasymbol</code>	$\beta$	$\text{\textcircled{\beta}}$
<code>\epsilon</code>	$\epsilon$	$\epsilon$
<code>\varepsilon</code>	$\epsilon$	$\epsilon$
<code>\epsilonsymbol</code>	$\epsilon$	$\epsilon$
<code>\phi</code>	$\varphi$	$\phi$
<code>\varphi</code>	$\varphi$	$\varphi$
<code>\phisymbol</code>	$\varphi$	$\phi$
<code>\kappa</code>	$\kappa$	$\kappa$
<code>\varkappa</code>	$\kappa$	$\varkappa$
<code>\kappasymbol</code>	$\kappa$	$\varkappa$
<code>\pi</code>	$\pi$	$\pi$
<code>\varpi</code>	$\pi$	$\varpi$
<code>\pisymbol</code>	$\pi$	$\varpi$
<code>\rho</code>	$\rho$	$\rho$
<code>\varrho</code>	$\rho$	$\varrho$
<code>\rhosymbol</code>	$\rho$	$\varrho$
<code>\sigma</code>	$\sigma$	$\sigma$
<code>\varsigma</code>	$\varsigma$	$\varsigma$
<code>\finalsigma</code>	$\varsigma$	$\varsigma$
<code>\theta</code>	$\vartheta$	$\theta$
<code>\vartheta</code>	$\vartheta$	$\vartheta$
<code>\thetasymbol</code>	$\vartheta$	$\vartheta$
<code>\Theta</code>	$\Theta$	$\Theta$
<code>\varTheta</code>	$\text{\textcircled{\Theta}}$	$\Theta$
<code>\Thetasymbol</code>	$\Theta$	$\text{\textcircled{\Theta}}$

Table 1: Macros for Greek symbol variants ( $\text{\textcircled{\beta}}$ = symbol only available with additional packages). With 8-bit TeX and the `normalize-symbols` option, the output for both variants in text mode is the same (8-bit Greek text fonts contain only one symbol variant).