The lgrmath package

Jean-François Burnol
jfbu (at) free (dot) fr

This is 1.0 (2022/11/16) from source file having Time-stamp: <16-11-2022 at 21:29:30 CET>
Report issues at https://github.com/jfbu/lgrmath/issues

Contents

List of Tables

1 Description

The lgrmath package sets the Greek letters in math mode (**only**) to use glyphs from the LGR-encoded font of one's choice.

Thus lgrmath is for people who want *only* to adjust Greek letters in math mode (and easily configure usage of upright or italic/slanted shapes), perhaps in the context of having changed Latin letters as well, e.g. from using the frenchmath[1] package which makes uppercase Latin letters in math mode render upright, among quite a few other adjustments tailored for French mathematical typesetting, or the mathastext[2] package. Actually lgrmath is in part inspired from this latter package `LGRgreek` option and `\MTgreekfont` command. But lgrmath currently does not incorporate a mechanism for defining and using multiple math versions, each one with its own font for Greek letters, as is already provided by mathastext.

The package is also related to libgreek[3], also by the author, and shares most of its codebase, after dropping matters related to `libgreek-legacy`, and the `scale` option which can not be implemented generically.

[1] Antoine Missier, *Typesetting mathematics according to French rules*, https://ctan.org/pkg/frenchmath.
[2] Verfasser, *Use the text font in math mode*, https://ctan.org/pkg/mathastext.
[3] Verfasser, *Greek letters in math mode from Libertinus or Linux Libertine/Biolinum*, https://ctan.org/pkg/libgreek.

The Greek letters all come with `\...up` and `\...it` named variants, and whether "bare" control sequences map to the '`up`' or '`it`' ones can be configured via package options, even midway in the document via `\lgrmathsetup`. Further, the package optionally defines two math alphabets `\lgrmathup` and `\lgrmathit`. What '`up`' and '`it`' actually mean can be configured using the `upshape` and `itshape` keys at package loading time.

2 Options of the lgrmath package

`\Alpha`	Α	`\Nu`	Ν	`\alpha`	α	`\nu`	ν
`\Beta`	Β	`\Xi`	Ξ	`\beta`	β	`\xi`	ξ
`\Gamma`	Γ	`\Omicron`	Ο	`\gamma`	γ	`\omicron`	ο
`\Delta`	Δ	`\Pi`	Π	`\delta`	δ	`\pi`	π
`\Epsilon`	Ε	`\Rho`	Ρ	`\epsilon`	ε	`\rho`	ρ
`\Zeta`	Ζ	`\Sigma`	Σ	`\zeta`	ζ	`\sigma`	σ
`\Eta`	Η	`\Tau`	Τ	`\eta`	η	`\tau`	τ
`\Theta`	Θ	`\Upsilon`	Υ	`\theta`	ϑ	`\upsilon`	υ
`\Iota`	Ι	`\Phi`	Φ	`\iota`	ι	`\phi`	φ
`\Kappa`	Κ	`\Chi`	Χ	`\kappa`	ϰ	`\chi`	χ
`\Lambda`	Λ	`\Psi`	Ψ	`\lambda`	λ	`\psi`	ψ
`\Mu`	Μ	`\Omega`	Ω	`\mu`	μ	`\omega`	ω

`\varsigma`	ς	`\digamma`	ϝ	`\varSigma`	Ϲ
`\varvarsigma`	ϛ	`\koppa`	ϟ	`\Sampi`	Ϡ
`\sampi`	ϡ			`\Digamma`	Ϝ

Table 1: Greek letters, upright shapes, default family

Here are the options recognized by the package:

font=⟨*font_name*⟩ This specifies which font (font family in the sens of the LaTeX font selection scheme) to use. It defaults to `lmr`.

In Table 1 and Table 2 we display the glyphs from this default font `lmr` in LGR encoding, available to LaTeX thanks to the support files from the package (in the sense of CTAN or TeXLive, not of a LaTeX document) cbfonts-fd.[4] It is recommended to user to have a look at its documentation

```
texdoc cbfonts
```

in particular the section on Customizations which mentions alternate shapes (such as `rs`, `ro`, `li`, `iv`, `uv` — those last two are actually for sans-serif `lmss` —, and there are also comments relative to the series) and use appropriately the `upshape`, `itshape`, `series` and `boldseries` lgrmath keys which are documented next.

[4] Claudio Beccari, *LaTeX font description files for the CB Greek fonts*, https://ctan.org/pkg/cbfonts-fd.

\Alpha	A	\Nu	N	\alpha	α	\nu	ν
\Beta	B	\Xi	Ξ	\beta	β	\xi	ξ
\Gamma	Γ	\Omicron	O	\gamma	γ	\omicron	o
\Delta	Δ	\Pi	Π	\delta	δ	\pi	π
\Epsilon	E	\Rho	P	\epsilon	ϵ	\rho	ρ
\Zeta	Z	\Sigma	Σ	\zeta	ζ	\sigma	σ
\Eta	H	\Tau	T	\eta	η	\tau	τ
\Theta	Θ	\Upsilon	Υ	\theta	ϑ	\upsilon	υ
\Iota	I	\Phi	Φ	\iota	ι	\phi	φ
\Kappa	K	\Chi	X	\kappa	κ	\chi	χ
\Lambda	Λ	\Psi	Ψ	\lambda	λ	\psi	ψ
\Mu	M	\Omega	Ω	\mu	μ	\omega	ω

\varsigma	ς	\digamma	$\digamma$	\varSigma	$\mathit{\Sigma}$
\varvarsigma	ς	\koppa	$\koppa$	\Sampi	$\Sampi$
\sampi	$\sampi$			\Digamma	F

Table 2: Greek letters, italic shapes, default family

The allowable names ⟨*font_name*⟩'s are those `foo` for which a file `LGRfoo.fd` or `lgrfoo.fd` exists on the system.

The above remarks about customization apply generally to all fonts, try to see if there is some documentation associated with the font you choose. Ultimate experts will look into the `.fd` files to see (for example) if there is some interface to rescale the fonts by some factor.

Here is now a list of suitable such font definition files from which you can extract usable font family names. This has been obtained via exercising the Unix `find` utility in a TEXLive 2022 installation (possibly only partial). To test a font the package provides `\lgrmathgreektable` and `\lgrmathgreektableextra` which are documented in the next section.

```
in directory  /usr/local/texlive/2022/texmf-dist/tex/latex we execute
    find . -name 'LGR*fd'
and then rearrange somewhat the output to put it in alphabetical order,
and gain some space horizontally so as to obtain a two-column display
Naturally in many instances the various -TLF, -OsF, and so on, refer
to options of digit characters and have no impact on the Greek letters,
nevertheless I kept all filenames, just pick one, drop LGR and .fd parts.

./alegreya/                                     ./librefranklin/
    LGRAlegreya-Inf.fd                              LGRLibreFranklin-Sup.fd
    LGRAlegreya-LF.fd                               LGRLibreFranklin-TLF.fd
    LGRAlegreya-OsF.fd                          ./linguisticspro/
    LGRAlegreya-Sup.fd                              LGRLinguisticsPro-LF.fd
    LGRAlegreya-TLF.fd                              LGRLinguisticsPro-OsF.fd
    LGRAlegreya-TOsF.fd                         ./nimbus15/
```

```
    LGRAlegreyaSans-Inf.fd
    LGRAlegreyaSans-LF.fd
    LGRAlegreyaSans-OsF.fd
    LGRAlegreyaSans-Sup.fd
    LGRAlegreyaSans-TLF.fd
    LGRAlegreyaSans-TOsF.fd
./clara/
    LGRClara-Sup.fd
    LGRClara-TLF.fd
    LGRClara-TOsF.fd
./cochineal/
    LGRCochineal-LF.fd
    LGRCochineal-OsF.fd
    LGRCochineal-TLF.fd
    LGRCochineal-TOsF.fd
./comfortaa/
    LGRcomfortaa.fd
./dejavu/
    LGRDejaVuSans-TLF.fd
    LGRDejaVuSansCondensed-TLF.fd
    LGRDejaVuSansMono-TLF.fd
    LGRDejaVuSerif-TLF.fd
    LGRDejaVuSerifCondensed-TLF.fd
./domitian/
    LGRDomitian-Inf.fd
    LGRDomitian-Sup.fd
    LGRDomitian-TLF.fd
    LGRDomitian-TOsF.fd
./droid/
    LGRdroidsans.fd
    LGRdroidsansmono.fd
    LGRdroidserif.fd
./ebgaramond/
    LGREBGaramond-Inf.fd
    LGREBGaramond-LF.fd
    LGREBGaramond-OsF.fd
    LGREBGaramond-Sup.fd
    LGREBGaramond-TLF.fd
    LGREBGaramond-TOsF.fd
    LGREBGaramondInitials-TLF.fd
./fira/
    LGRFiraMono-Sup.fd
    LGRFiraMono-TLF.fd
    LGRFiraMono-TOsF.fd
    LGRFiraSans-LF.fd
    LGRFiraSans-OsF.fd
    LGRFiraSans-Sup.fd
    LGRFiraSans-TLF.fd
    LGRFiraSans-TOsF.fd
./garamond-libre/
    LGRGaramondLibre-Inf.fd
    LGRGaramondLibre-LF.fd
    LGRGaramondLibre-OsF.fd
    LGRNimbuSans.fd
    LGRNimbusMono.fd
    LGRNimbusMonoN.fd
    LGRNimbusSerif.fd
./noto/
    LGRNotoSans-LF.fd
    LGRNotoSans-OsF.fd
    LGRNotoSans-Sup.fd
    LGRNotoSans-TLF.fd
    LGRNotoSans-TOsF.fd
    LGRNotoSansMono-Sup.fd
    LGRNotoSansMono-TLF.fd
    LGRNotoSansMono-TOsF.fd
    LGRNotoSerif-LF.fd
    LGRNotoSerif-OsF.fd
    LGRNotoSerif-Sup.fd
    LGRNotoSerif-TLF.fd
    LGRNotoSerif-TOsF.fd
./oldstandard/
    LGROldStandard-Sup.fd
    LGROldStandard-TLF.fd
./opensans/
    LGRopensans-LF.fd
    LGRopensans-OsF.fd
    LGRopensans-TLF.fd
    LGRopensans-TOsF.fd
./plex/
    LGRIBMPlexSans-Sup.fd
    LGRIBMPlexSans-TLF.fd
./roboto/
    LGRRoboto-LF.fd
    LGRRoboto-OsF.fd
    LGRRoboto-TLF.fd
    LGRRoboto-TOsF.fd
    LGRRobotoMono-TLF.fd
    LGRRobotoSerif-LF.fd
    LGRRobotoSerif-OsF.fd
    LGRRobotoSerif-Sup.fd
    LGRRobotoSerif-TLF.fd
    LGRRobotoSerif-TOsF.fd
    LGRRobotoSlab-TLF.fd
./sourcesanspro/
    LGRSourceSansPro-Dnom.fd
    LGRSourceSansPro-Inf.fd
    LGRSourceSansPro-LF.fd
    LGRSourceSansPro-Numr.fd
    LGRSourceSansPro-OsF.fd
    LGRSourceSansPro-Sup.fd
    LGRSourceSansPro-TLF.fd
    LGRSourceSansPro-TOsF.fd
./step/
    LGRSTEP-Inf.fd
    LGRSTEP-Sup.fd
```

```
    LGRGaramondLibre-Sup.fd
./gofonts/
    LGRGo-TLF.fd
    LGRGoMono-TLF.fd
./lato/
    LGRlato-LF.fd
    LGRlato-OsF.fd
    LGRlato-TLF.fd
    LGRlato-TOsF.fd
./libertinegc/
    LGRLinuxLibertineT-LF.fd
    LGRLinuxLibertineT-OsF.fd
    LGRLinuxLibertineT-TLF.fd
    LGRLinuxLibertineT-TOsF.fd
./libertinus-type1/
    LGRLibertinusSans-LF.fd
    LGRLibertinusSans-OsF.fd
    LGRLibertinusSans-Sup.fd
    LGRLibertinusSans-TLF.fd
    LGRLibertinusSans-TOsF.fd
    LGRLibertinusSerif-LF.fd
    LGRLibertinusSerif-OsF.fd
    LGRLibertinusSerif-Sup.fd
    LGRSTEP-TLF.fd
    LGRSTEP-TOsF.fd
./stepgreek/
    LGRSTEPGreekTest-Sup.fd
    LGRSTEPGreekTest-TLF.fd
    LGRSTEPGreekTest-TOsF.fd
./theanodidot/
    LGRTheanoDidot-TLF.fd
    LGRTheanoDidot-TOsF.fd
./theanomodern/
    LGRTheanoModern-TLF.fd
    LGRTheanoModern-TOsF.fd
./theanooldstyle/
    LGRTheanoOldStyle-TLF.fd
    LGRTheanoOldStyle-TOsF.fd
LGRLibertinusSerif-TLF.fd
LGRLibertinusSerif-TOsF.fd
LGRLibertinusSerifDisplay-LF.fd
LGRLibertinusSerifDisplay-OsF.fd
LGRLibertinusSerifDisplay-Sup.fd
LGRLibertinusSerifDisplay-TLF.fd
LGRLibertinusSerifDisplay-TOsF.fd
LGRLibertinusSerifInitials-TLF.fd

And now for more, with lowercase `lgr' filenames: find . -name 'lgr*fd'

./txfontsb/lgrtxr.fd
./txfontsb/lgrtxrc.fd
./txfontsb/lgrtxry.fd
./txfontsb/lgrtxryc.fd
./gfsbodoni/lgrbodoni.fd
./lxfonts/lgrllcmtt.fd
./lxfonts/lgrllcmss.fd
./kerkis/lgrkfn.fd
./kerkis/lgrmaksf.fd
./kerkis/lgrmak.fd
./cbfonts-fd/lgrcmro.fd
./cbfonts-fd/lgrcmss.fd
./cbfonts-fd/lgrlmr.fd
./cbfonts-fd/lgrlcmtt.fd
./cbfonts-fd/lgrlmtt.fd
./cbfonts-fd/lgrlmss.fd
./cbfonts-fd/lgrlmro.fd
./cbfonts-fd/lgrlcmss.fd
./cbfonts-fd/lgrcmtt.fd
./cbfonts-fd/lgrcmr.fd
./cm-lgc/lgrfcm.fd
./cm-lgc/lgrfct.fd
./cm-lgc/lgrfcs.fd
./epigrafica/lgrepigrafica.fd
./gfssolomos/lgrsolomos.fd
./tempora/lgrtempora-tlf.fd
./tempora/lgrtempora-tosf.fd
./gfscomplutum/lgrcomplutum.fd
./gfsartemisia/lgrartemisiaeuler.fd
./gfsartemisia/lgrartemisia.fd
./gentium-tug/lgrgentiumbook.fd
./gentium-tug/lgrgentium.fd
./gfsbaskerville/lgrgfsbaskerville.fd
./miama/lgrfmm.fd
./gfsneohellenic/lgrneohellenic.fd
./gfsdidot/lgrudidot.fd
./gfsporson/lgrporson.fd
```

upshape=⟨*shape*⟩ Declares the shape to be used by the `\...up` Greek letters and the `\lgrmathup` math alphabet. Defaults to '`n`' (without the quotes).

itshape=⟨*shape*⟩ Declares the shape to be used by the `\...it` Greek letters and the `\lgrmathit` math alphabet. Defaults to '`it`'.

style=⟨ISO|UP|TeX⟩ specifies the shape style of the Greek letters.

`ISO` means 'italic' for lowercase and uppercase, `UP` means 'upright' for lowercase and uppercase, `TeX` means 'italic' for lowercase and 'upright' for uppercase. The lowercase forms `iso`, `up` and `tex` are also accepted (or any mixed case).

One can also use `French` or `french` as an alias to `UP` or `up`.

This option will override any `greek` or `Greek` option. The package defaults to `style=TeX`.

What 'upright' and 'italic' mean is configured by the `upshape` and `itshape` respective settings.

greek=⟨up|it|...⟩ Says whether Greek letters will be 'upright' or 'italic' i.e. whether they obey the `upshape` or `itshape` setting, i.e. whether `\alpha` et al. are `\let` to `\alphaup` (et al.) or to `\alphait` (et al.).

So `greek=it` is like `style=ISO`, and `greek=up` is like `style=French`.

Other shape values, such as '`n`' and '`sl`' or even '`sc`', are accepted. For more details, see the explanations for `Greek`. For example `greek=n` is like `style=UP`.

This option is ignored if `style` is used (order does not matter).

Greek=⟨up|it|...⟩ Says whether uppercase Greek letters (and only them) will be 'upright' or 'italic' i.e. whether they use `upshape` or `itshape`, i.e. whether `\Alpha` et al. are `\let` to `\Alphaup` (et al.) or to `\Alphait` (et al.).

So to obtain lowercase to be 'upright' and uppercase to be 'italic', use `greek=up` and then `Greek=it` (`Greek` must appear after `greek` else it will be shadowed by it).

This option, like the `greek` option, is ignored if the `style` option is used.

Other shape values, such as '`n`' and '`sl`', are accepted. They will then override the `upshape` setting for it to match it. For example `Greek=sc` will force `upshape` to be `sc`, because the assumed style is the TEX one of italic lowercase and upright uppercase, so setting the shape of uppercase must update the `upshape` value.

series=⟨*series*⟩ This tells which series to use. The default is the value of `\seriesdefault` at the time of loading the package. There is no interface to configure distinct series for the 'upright' and 'italic' shapes.

boldseries=⟨*series*⟩ This tells which series to use in bold math. Default is `\bfdefault` at the time of loading the package. There is no interface to configure distinct series for the 'upright' and 'italic' shapes.

alphabets Says whether to define `\lgrmathup` and `\lgrmathit`.

3 Commands of the lgrmath package

Here are the commands defined by the package:

\lgrmathsetup{⟨*key=value,...*⟩} The only allowed keys are `style`, `greek` and `Greek`. And for the latter two only the values `up` or `it` should be used (or values matching the `upshape` or `itshape` settings), as it is only possible after package loading time to toggle between 'upright' and 'italic' depending on whether the letter is uppercase or lowercase, but one can not switch to an altogether different shape as this would require re-declaring the symbol fonts.

If the `style` key is used, then `greek/Greek` are ignored. However, one can always naturally reuse later `\lgrmathsetup` using only the `greek` and/or `Greek` keys.

\lgrmathup This is a math alphabet. It is defined only if the package received the `alphabets` option.

\lgrmathit This is a math alphabet. It is defined only if the package received the `alphabets` option.

\lgrmathgreektable{⟨*family*⟩}{⟨*series*⟩}{⟨*shape*⟩} Produces a tabular display of the Greek letters available with this font. Here is for example using

```
\lgrmathgreektable{Alegreya-TLF}{regular}{n}
```

\Alpha	Α	\Nu	Ν	\alpha	α	\nu	ν
\Beta	Β	\Xi	Ξ	\beta	β	\xi	ξ
\Gamma	Γ	\Omicron	Ο	\gamma	γ	\omicron	ο
\Delta	Δ	\Pi	Π	\delta	δ	\pi	π
\Epsilon	Ε	\Rho	Ρ	\epsilon	ε	\rho	ρ
\Zeta	Ζ	\Sigma	Σ	\zeta	ζ	\sigma	σ
\Eta	Η	\Tau	Τ	\eta	η	\tau	τ
\Theta	Θ	\Upsilon	Υ	\theta	θ	\upsilon	υ
\Iota	Ι	\Phi	Φ	\iota	ι	\phi	φ
\Kappa	Κ	\Chi	Χ	\kappa	κ	\chi	χ
\Lambda	Λ	\Psi	Ψ	\lambda	λ	\psi	ψ
\Mu	Μ	\Omega	Ω	\mu	μ	\omega	ω

We used `regular` for the ⟨*series*⟩ mandatory argument after seeing Font Info messages in the `.log` file about the `m` series not being available and being substituted for by `regular`, so we used `regular` to avoid those messages.

\lgrmathgreektableextra{⟨*family*⟩}{⟨*series*⟩}{⟨*shape*⟩} Produces a tabular with eight additional glyphs. Here is an example, using

```
\lgrmathgreektableextra{LibertinusSans-TLF}{m}{n}
```

\varsigma	ς	\digamma	ϝ	\varSigma	ϛ
\varvarsigma	ς	\koppa	ϟ	\Sampi	Ϡ
\sampi	ϡ			\Digamma	Ϝ

Beware that if we had forgotten the `-TLF` suffix, the font would have been substituted in favour of fall-back `lmr` by LaTeX. Always check log for font substitutions messages...

And see also the last remark below.

Miscellaneous remarks:

1. Even if not receiving the option `alphabets`, the package will declare all Greek letters to be of type `\mathalpha`.

2. The **lgrmath** package ignores global class options. It handles only options originating from the `\usepackage` preamble declaration (or some options handed over via `\PassOptionsToPackage` or options passed to `\lgrmathsetup` in the preamble or body).

3. The libgreek package defines `\mathchar`'s mapping to lowercase Greek letters with diacritics, but for time being it has been decided that **lgrmath** would restrict its definitions to the 24+24 base glyphs and the 8 "extra" ones for which there are slots in the LGR encoding table.

4. These 8 "extra glyphs" will not always be available, depending on the font. Here is for example with `Alegreya-TLF`:

`\varsigma`	ς	`\digamma`	`\varSigma`
`\varvarsigma`		`\koppa`	`\Sampi`
`\sampi`			`\Digamma`

   Adding `\tracinglostchars=3` will cause TeX to raise an error in case such missing characters are encountered.

This is the end of the user documentation. The next section is a code listing with some comments for the advanced users.

4 Implementation of the lgrmath package

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{lgrmath}
          [2022/11/16 1.0 Greek in math mode via LGR font of one's choice (JFB)]
```

We will use kvoptions to handle options with `key=value` syntax.

```
\RequirePackage{kvoptions}
```

To minimize the author's task, we keep close to `libgreek.sty` code with minimal adaptations. In particular I decided to keep the fact that `style` option makes the `Greek` and `greek` options ignored. But there are some complications originating in the addition of the `\lgrmathsetup`, which requires to keep a trace of various things, for example if `style` option is used at package level and then later on using `\lgrmathsetup` the user employs the `Greek/greek` options.

This package assigns two symbol fonts, one for upright, the other one for italic-like.

The `upshape` and `itshape` keys allow to configure what the `\...up` and `\...it` macros will actually use as shapes.

```
\def\lgrmath@fontfamily{lmr}
\def\lgrmath@scale{1}
\def\lgrmath@upshape{n}
\def\lgrmath@itshape{it}
\newif\iflgrmath@upper@up\lgrmath@upper@uptrue
\newif\iflgrmath@lower@up
\edef\lgrmath@series{\seriesdefault}
\edef\lgrmath@boldseries{\bfdefault}
\def\lgrmath@upper@shape{\lgrmath@upshape}
\def\lgrmath@lower@shape{\lgrmath@itshape}
\def\lgrmath@style{TeX}
\newif\iflgrmath@sty
```

We use the keyval interface mostly to not have to rework everything, if at all possible, into the kvoptions declarative interface. It is a very good thing that the latter package can be used without forcing on the user its own declarative interface...

```
\define@key{lgrmath}{font}[lmr]{\def\lgrmath@fontfamily{#1}}
\define@key{lgrmath}{upshape}{\edef\lgrmath@upshape{#1}}
\define@key{lgrmath}{itshape}{\edef\lgrmath@itshape{#1}}
```

Compared to `libgreek 1.1` I decide to use `\lowercase` and allow `UP` as alias of `French`.

```
\define@key{lgrmath}{style}{%
   \edef\lgrmath@style{#1}%
   \lowercase\expandafter{\expandafter\def\expandafter\lgrmath@style
               \expandafter{\lgrmath@style}}%
   \lgrmath@stytrue
}
\define@key{lgrmath}{Greek}{\edef\lgrmath@upper@shape{#1}}
```

Attention to not introduce a space token, as this may be used via `\lgrmathsetup` in document body.

```
\define@key{lgrmath}{greek}{\edef\lgrmath@lower@shape{#1}%
                            \edef\lgrmath@upper@shape{#1}}
\define@key{lgrmath}{series}{\edef\lgrmath@series{#1}}
\define@key{lgrmath}{boldseries}{\edef\lgrmath@boldseries{#1}}
```

The single Boolean option, a true one as it uses kvoptions interface.

```
\DeclareBoolOption[false]{alphabets}
```

We need some auxiliaries to handle the `style` values. As mentioned already, some extra stuff is executed for reasons of various scenarii with `\lgrmathsetup`.

```
\def\lgrmath@style@iso{%
    \lgrmath@upper@upfalse
    \lgrmath@lower@upfalse
    \let\lgrmath@upper@shape\lgrmath@itshape
    \let\lgrmath@lower@shape\lgrmath@itshape
}
\def\lgrmath@style@french{%
    \lgrmath@upper@uptrue
    \lgrmath@lower@uptrue
    \let\lgrmath@upper@shape\lgrmath@upshape
    \let\lgrmath@lower@shape\lgrmath@upshape
}
\let\lgrmath@style@up\lgrmath@style@french
\def\lgrmathk@style@tex{%
    \lgrmath@upper@uptrue
    \lgrmath@lower@upfalse
    \let\lgrmath@upper@shape\lgrmath@upshape
    \let\lgrmath@lower@shape\lgrmath@itshape
}
```

This always resets the `\iflgrmath@sty` to false for `\lgrmathsetup` being usable with `greek` and `Greek` keys.

```
\def\lgrmath@process@style{%
   \lgrmath@styfalse
   \ifcsname lgrmath@style@\lgrmath@style\endcsname
     \csname lgrmath@style@\lgrmath@style\endcsname
   \else
     \PackageWarning{lgrmath}{Unknown (here, lowercased) style `\lgrmath@style'}%
   \fi
}
```

This stuff is a bit involved.

```
\def\lgrmath@process@shapes{%
   \edef\lgrmath@upper@shape{\lgrmath@upper@shape}%
   \edef\lgrmath@lower@shape{\lgrmath@lower@shape}%
   \ifx\lgrmath@upper@shape\lgrmath@upshape
      \lgrmath@upper@uptrue
   \else
      \ifx\lgrmath@upper@shape\lgrmath@itshape
         \lgrmath@upper@upfalse
      \else
   \expandafter\in@\expandafter{\expandafter.\lgrmath@upper@shape,}{.up,}%
   \ifin@\lgrmath@upper@uptrue
   \else
     \expandafter\in@\expandafter{\expandafter.\lgrmath@upper@shape,}{.it,}%
```

```
    \ifin@\lgrmath@upper@upfalse
    \else
       \lgrmath@process@upper@lastresort
  \fi\fi\fi\fi
  \ifx\lgrmath@lower@shape\lgrmath@itshape
     \lgrmath@lower@upfalse
  \else
     \ifx\lgrmath@lower@shape\lgrmath@upshape
        \lgrmath@lower@uptrue
     \else
  \expandafter\in@\expandafter{\expandafter.\lgrmath@lower@shape,}{.it,}%
  \ifin@\lgrmath@lower@upfalse
  \else
    \expandafter\in@\expandafter{\expandafter.\lgrmath@lower@shape,}{.up,}%
    \ifin@\lgrmath@lower@uptrue
    \else
       \lgrmath@process@lower@lastresort
  \fi\fi\fi\fi
}%
\def\lgrmath@process@upper@lastresort{%
    \lgrmath@upper@uptrue
    \let\lgrmath@upshape\lgrmath@upper@shape
}
\def\lgrmath@process@lower@lastresort{%
    \lgrmath@lower@upfalse
    \let\lgrmath@itshape\lgrmath@lower@shape
}
```

The fact that packages may be handed global options is rather dangerous. Fortunately kvoptions has an interface to handle only local options.

```
\ProcessLocalKeyvalOptions*
```

We now do the post-processing regarding the shape configuration after option parsing. Once this is done we will reconfigure slightly `\lgrmath@process@shapes` for usability in the document preamble or body, after the symbol fonts have been declared. As is well-known the LaTeX interface to math fonts is full of "only-preamble" restrictions.

```
\iflgrmath@sty
  \lgrmath@process@style
\else
  \lgrmath@process@shapes
\fi
\def\lgrmath@process@upper@lastresort{%
    \PackageWarning{lgrmath}{%
                      Too late for the shape `\lgrmath@upper@shape'\MessageBreak
                      originating in Greek or greek option. Ignored.\MessageBreak
                      Use `up' or `it'}%
}
\def\lgrmath@process@lower@lastresort{%
    \PackageWarning{lgrmath}{%
                      Too late for the shape `\lgrmath@lower@shape'\MessageBreak
```

```
                        originating in greek option. Ignored.\MessageBreak
                        Use `up' or `it'}%
}
\def\lgrmathsetup#1{%
    \setkeys{lgrmath}{#1}%
    \iflgrmath@sty\lgrmath@process@style\else\lgrmath@process@shapes\fi
    \lgrmath@setgreekcs
}
```

Almost all options must be restricted to the package loading time only.

```
\DisableKeyvalOption{lgrmath}{font}
\DisableKeyvalOption{lgrmath}{upshape}
\DisableKeyvalOption{lgrmath}{itshape}
\DisableKeyvalOption{lgrmath}{series}
\DisableKeyvalOption{lgrmath}{boldseries}
\DisableKeyvalOption{lgrmath}{alphabets}
```

Declarations of the two symbol fonts, one for 'upright' (or whatever is configured by the `upshape` key), one for 'italic' (or whatever is configured by the `itshape` key). One can not specify distinct series, both 'upright' and 'italic' use the same font series. This could be added but I doubt anyone will use the package to start with...

The libgreek of `2022/11/11` extracted the `-TLF` postfix from the font family name, to reinsert it here explicitly, the options `serif/sans` deciding whether to use `LibertinusSerif-TLF` or `LibertinusSans-TLF` for reasons now escaping me. I vaguely remember it was useful at some point during development. Ah yes, now I remember this separation was for the handling of the `scale` option. And we haven't one here.

```
\DeclareFontEncoding{LGR}{}{}
\DeclareSymbolFont{lgrmathup}{LGR}{\lgrmath@fontfamily}
                                  {\lgrmath@series}
                                  {\lgrmath@upshape}
\SetSymbolFont{lgrmathup}{bold}{LGR}{\lgrmath@fontfamily}
                                    {\lgrmath@boldseries}
                                    {\lgrmath@upshape}
\DeclareSymbolFont{lgrmathit}{LGR}{\lgrmath@fontfamily}
                                  {\lgrmath@series}
                                  {\lgrmath@itshape}
\SetSymbolFont{lgrmathit}{bold}{LGR}{\lgrmath@fontfamily}
                                    {\lgrmath@boldseries}
                                    {\lgrmath@itshape}
```

As all Greek letters are already available in `\...up` and `\...it` variants, it is indeed not immediately pressing to have math alphabets, so let's not do it by default.

```
\iflgrmath@alphabets
   \DeclareSymbolFontAlphabet{\lgrmathup}{lgrmathup}
   \DeclareSymbolFontAlphabet{\lgrmathit}{lgrmathit}
\fi
```

Definition of the '`up`' `\mathchar`'s. There are 48 'standard' ones plus 8 extras.

Hesitation whether I should declare with `\mathalpha` *only* if `alphabets` is passed to the package.

```
\DeclareMathSymbol{\Alphaup}{\mathalpha}{lgrmathup}{65}
```

```
\DeclareMathSymbol{\Betaup}{\mathalpha}{lgrmathup}{66}
\DeclareMathSymbol{\Gammaup}{\mathalpha}{lgrmathup}{71}
\DeclareMathSymbol{\Deltaup}{\mathalpha}{lgrmathup}{68}
\DeclareMathSymbol{\Epsilonup}{\mathalpha}{lgrmathup}{69}
\DeclareMathSymbol{\Zetaup}{\mathalpha}{lgrmathup}{90}
\DeclareMathSymbol{\Etaup}{\mathalpha}{lgrmathup}{72}
\DeclareMathSymbol{\Thetaup}{\mathalpha}{lgrmathup}{74}
\DeclareMathSymbol{\Iotaup}{\mathalpha}{lgrmathup}{73}
\DeclareMathSymbol{\Kappaup}{\mathalpha}{lgrmathup}{75}
\DeclareMathSymbol{\Lambdaup}{\mathalpha}{lgrmathup}{76}
\DeclareMathSymbol{\Muup}{\mathalpha}{lgrmathup}{77}
\DeclareMathSymbol{\Nuup}{\mathalpha}{lgrmathup}{78}
\DeclareMathSymbol{\Xiup}{\mathalpha}{lgrmathup}{88}
\DeclareMathSymbol{\Omicronup}{\mathalpha}{lgrmathup}{79}
\DeclareMathSymbol{\Piup}{\mathalpha}{lgrmathup}{80}
\DeclareMathSymbol{\Rhoup}{\mathalpha}{lgrmathup}{82}
\DeclareMathSymbol{\Sigmaup}{\mathalpha}{lgrmathup}{83}
\DeclareMathSymbol{\Tauup}{\mathalpha}{lgrmathup}{84}
\DeclareMathSymbol{\Upsilonup}{\mathalpha}{lgrmathup}{85}
\DeclareMathSymbol{\Phiup}{\mathalpha}{lgrmathup}{70}
\DeclareMathSymbol{\Chiup}{\mathalpha}{lgrmathup}{81}
\DeclareMathSymbol{\Psiup}{\mathalpha}{lgrmathup}{89}
\DeclareMathSymbol{\Omegaup}{\mathalpha}{lgrmathup}{87}
\DeclareMathSymbol{\alphaup}{\mathalpha}{lgrmathup}{97}
\DeclareMathSymbol{\betaup}{\mathalpha}{lgrmathup}{98}
\DeclareMathSymbol{\gammaup}{\mathalpha}{lgrmathup}{103}
\DeclareMathSymbol{\deltaup}{\mathalpha}{lgrmathup}{100}
\DeclareMathSymbol{\epsilonup}{\mathalpha}{lgrmathup}{101}
\DeclareMathSymbol{\zetaup}{\mathalpha}{lgrmathup}{122}
\DeclareMathSymbol{\etaup}{\mathalpha}{lgrmathup}{104}
\DeclareMathSymbol{\thetaup}{\mathalpha}{lgrmathup}{106}
\DeclareMathSymbol{\iotaup}{\mathalpha}{lgrmathup}{105}
\DeclareMathSymbol{\kappaup}{\mathalpha}{lgrmathup}{107}
\DeclareMathSymbol{\lambdaup}{\mathalpha}{lgrmathup}{108}
\DeclareMathSymbol{\muup}{\mathalpha}{lgrmathup}{109}
\DeclareMathSymbol{\nuup}{\mathalpha}{lgrmathup}{110}
\DeclareMathSymbol{\xiup}{\mathalpha}{lgrmathup}{120}
\DeclareMathSymbol{\omicronup}{\mathalpha}{lgrmathup}{111}
\DeclareMathSymbol{\piup}{\mathalpha}{lgrmathup}{112}
\DeclareMathSymbol{\rhoup}{\mathalpha}{lgrmathup}{114}
\DeclareMathSymbol{\sigmaup}{\mathalpha}{lgrmathup}{115}
\DeclareMathSymbol{\tauup}{\mathalpha}{lgrmathup}{116}
\DeclareMathSymbol{\upsilonup}{\mathalpha}{lgrmathup}{117}
\DeclareMathSymbol{\phiup}{\mathalpha}{lgrmathup}{102}
\DeclareMathSymbol{\chiup}{\mathalpha}{lgrmathup}{113}
\DeclareMathSymbol{\psiup}{\mathalpha}{lgrmathup}{121}
\DeclareMathSymbol{\omegaup}{\mathalpha}{lgrmathup}{119}
```

Defintion of the ‘`it`’ `\mathchar`’s.

```
\DeclareMathSymbol{\Alphait}{\mathalpha}{lgrmathit}{65}
```

```
\DeclareMathSymbol{\Betait}{\mathalpha}{lgrmathit}{66}
\DeclareMathSymbol{\Gammait}{\mathalpha}{lgrmathit}{71}
\DeclareMathSymbol{\Deltait}{\mathalpha}{lgrmathit}{68}
\DeclareMathSymbol{\Epsilonit}{\mathalpha}{lgrmathit}{69}
\DeclareMathSymbol{\Zetait}{\mathalpha}{lgrmathit}{90}
\DeclareMathSymbol{\Etait}{\mathalpha}{lgrmathit}{72}
\DeclareMathSymbol{\Thetait}{\mathalpha}{lgrmathit}{74}
\DeclareMathSymbol{\Iotait}{\mathalpha}{lgrmathit}{73}
\DeclareMathSymbol{\Kappait}{\mathalpha}{lgrmathit}{75}
\DeclareMathSymbol{\Lambdait}{\mathalpha}{lgrmathit}{76}
\DeclareMathSymbol{\Muit}{\mathalpha}{lgrmathit}{77}
\DeclareMathSymbol{\Nuit}{\mathalpha}{lgrmathit}{78}
\DeclareMathSymbol{\Xiit}{\mathalpha}{lgrmathit}{88}
\DeclareMathSymbol{\Omicronit}{\mathalpha}{lgrmathit}{79}
\DeclareMathSymbol{\Piit}{\mathalpha}{lgrmathit}{80}
\DeclareMathSymbol{\Rhoit}{\mathalpha}{lgrmathit}{82}
\DeclareMathSymbol{\Sigmait}{\mathalpha}{lgrmathit}{83}
\DeclareMathSymbol{\Tauit}{\mathalpha}{lgrmathit}{84}
\DeclareMathSymbol{\Upsilonit}{\mathalpha}{lgrmathit}{85}
\DeclareMathSymbol{\Phiit}{\mathalpha}{lgrmathit}{70}
\DeclareMathSymbol{\Chiit}{\mathalpha}{lgrmathit}{81}
\DeclareMathSymbol{\Psiit}{\mathalpha}{lgrmathit}{89}
\DeclareMathSymbol{\Omegait}{\mathalpha}{lgrmathit}{87}
\DeclareMathSymbol{\alphait}{\mathalpha}{lgrmathit}{97}
\DeclareMathSymbol{\betait}{\mathalpha}{lgrmathit}{98}
\DeclareMathSymbol{\gammait}{\mathalpha}{lgrmathit}{103}
\DeclareMathSymbol{\deltait}{\mathalpha}{lgrmathit}{100}
\DeclareMathSymbol{\epsilonit}{\mathalpha}{lgrmathit}{101}
\DeclareMathSymbol{\zetait}{\mathalpha}{lgrmathit}{122}
\DeclareMathSymbol{\etait}{\mathalpha}{lgrmathit}{104}
\DeclareMathSymbol{\thetait}{\mathalpha}{lgrmathit}{106}
\DeclareMathSymbol{\iotait}{\mathalpha}{lgrmathit}{105}
\DeclareMathSymbol{\kappait}{\mathalpha}{lgrmathit}{107}
\DeclareMathSymbol{\lambdait}{\mathalpha}{lgrmathit}{108}
\DeclareMathSymbol{\muit}{\mathalpha}{lgrmathit}{109}
\DeclareMathSymbol{\nuit}{\mathalpha}{lgrmathit}{110}
\DeclareMathSymbol{\xiit}{\mathalpha}{lgrmathit}{120}
\DeclareMathSymbol{\omicronit}{\mathalpha}{lgrmathit}{111}
\DeclareMathSymbol{\piit}{\mathalpha}{lgrmathit}{112}
\DeclareMathSymbol{\rhoit}{\mathalpha}{lgrmathit}{114}
\DeclareMathSymbol{\sigmait}{\mathalpha}{lgrmathit}{115}
\DeclareMathSymbol{\tauit}{\mathalpha}{lgrmathit}{116}
\DeclareMathSymbol{\upsilonit}{\mathalpha}{lgrmathit}{117}
\DeclareMathSymbol{\phiit}{\mathalpha}{lgrmathit}{102}
\DeclareMathSymbol{\chiit}{\mathalpha}{lgrmathit}{113}
\DeclareMathSymbol{\psiit}{\mathalpha}{lgrmathit}{121}
\DeclareMathSymbol{\omegait}{\mathalpha}{lgrmathit}{119}
```

Extras: alternate shapes and other glyphs, ‘upright’.

```
\DeclareMathSymbol{\varsigmaup}{\mathalpha}{lgrmathup}{99}
```

```
\DeclareMathSymbol{\varvarsigmaup}{\mathalpha}{lgrmathup}{6}
\DeclareMathSymbol{\varSigmaup}{\mathalpha}{lgrmathup}{22}
\DeclareMathSymbol{\Sampiup}{\mathalpha}{lgrmathup}{23}
\DeclareMathSymbol{\sampiup}{\mathalpha}{lgrmathup}{27}
\DeclareMathSymbol{\digammaup}{\mathalpha}{lgrmathup}{147}
\DeclareMathSymbol{\Digammaup}{\mathalpha}{lgrmathup}{195}
\DeclareMathSymbol{\koppaup}{\mathalpha}{lgrmathup}{18}
```

Extras: alternate shapes and other glyphs, 'italic'.

```
\DeclareMathSymbol{\varsigmait}{\mathalpha}{lgrmathit}{99}
\DeclareMathSymbol{\varvarsigmait}{\mathalpha}{lgrmathit}{6}
\DeclareMathSymbol{\varSigmait}{\mathalpha}{lgrmathit}{22}
\DeclareMathSymbol{\Sampiit}{\mathalpha}{lgrmathit}{23}
\DeclareMathSymbol{\sampiit}{\mathalpha}{lgrmathit}{27}
\DeclareMathSymbol{\digammait}{\mathalpha}{lgrmathit}{147}
\DeclareMathSymbol{\Digammait}{\mathalpha}{lgrmathit}{195}
\DeclareMathSymbol{\koppait}{\mathalpha}{lgrmathit}{18}
```

Some glyphs with diacritics. I decided not to keep this in **lgrmath**. Let's wait for extremely improbable feature request, as I won't do the feature request and will probably remain the sole user. Actually I don't think I will ever use this package as contexts where it could be useful are those where I would use mathastext and its `LGRgreek` option and `\MTgreekfont` command...

```
% \DeclareMathSymbol{\alphatonosup}{\mathalpha}{lgrmathup}{136}
% \DeclareMathSymbol{\epsilontonosup}{\mathalpha}{lgrmathup}{232}
% \DeclareMathSymbol{\etatonosup}{\mathalpha}{lgrmathup}{160}
% \DeclareMathSymbol{\iotatonosup}{\mathalpha}{lgrmathup}{208}
% \DeclareMathSymbol{\omicrontonosup}{\mathalpha}{lgrmathup}{236}
% \DeclareMathSymbol{\upsilontonosup}{\mathalpha}{lgrmathup}{212}
% \DeclareMathSymbol{\omegatonosup}{\mathalpha}{lgrmathup}{184}
% \DeclareMathSymbol{\upsilondieresistonosup}{\mathalpha}{lgrmathup}{246}
% \DeclareMathSymbol{\iotadieresisup}{\mathalpha}{lgrmathup}{240}
% \DeclareMathSymbol{\iotadieresistonosup}{\mathalpha}{lgrmathup}{242}
% \DeclareMathSymbol{\upsilondieresisup}{\mathalpha}{lgrmathup}{244}
% \DeclareMathSymbol{\alphatonosit}{\mathalpha}{lgrmathit}{136}
% \DeclareMathSymbol{\epsilontonosit}{\mathalpha}{lgrmathit}{232}
% \DeclareMathSymbol{\etatonosit}{\mathalpha}{lgrmathit}{160}
% \DeclareMathSymbol{\iotatonosit}{\mathalpha}{lgrmathit}{208}
% \DeclareMathSymbol{\omicrontonosit}{\mathalpha}{lgrmathit}{236}
% \DeclareMathSymbol{\upsilontonosit}{\mathalpha}{lgrmathit}{212}
% \DeclareMathSymbol{\omegatonosit}{\mathalpha}{lgrmathit}{184}
% \DeclareMathSymbol{\upsilondieresistonosit}{\mathalpha}{lgrmathit}{246}
% \DeclareMathSymbol{\iotadieresisit}{\mathalpha}{lgrmathit}{240}
% \DeclareMathSymbol{\iotadieresistonosit}{\mathalpha}{lgrmathit}{242}
% \DeclareMathSymbol{\upsilondieresisit}{\mathalpha}{lgrmathit}{244}
```

Definition of the `\mathchar`'s without '`up/it`' postfix. There are 27=24+3 uppercase and 29=24+5 lowercase letters, for a total of 56=48+8 glyphs. Actually, I had done some work with LGR in September 2011. I kept the file around. But at no point did I go back to check if I had done exhaustive work in 2011 and whether some other glyphs could be accounted for by LGR (not using ligatures) (I did re-check an old file about the LGR encoding I had from that 2011 work, but did not try to check for updates). Anyway, it is very doubtful whether it made

any sense for **lgrmath** to define control sequences for Greek letters with diacritics...

```
\def\lgrmath@setgreekcs{%
    \iflgrmath@upper@up
       \let\Alpha\Alphaup
       \let\Beta\Betaup
       \let\Gamma\Gammaup
       \let\Delta\Deltaup
       \let\Epsilon\Epsilonup
       \let\Zeta\Zetaup
       \let\Eta\Etaup
       \let\Theta\Thetaup
       \let\Iota\Iotaup
       \let\Kappa\Kappaup
       \let\Lambda\Lambdaup
       \let\Mu\Muup
       \let\Nu\Nuup
       \let\Xi\Xiup
       \let\Omicron\Omicronup
       \let\Pi\Piup
       \let\Rho\Rhoup
       \let\Sigma\Sigmaup
       \let\Tau\Tauup
       \let\Upsilon\Upsilonup
       \let\Phi\Phiup
       \let\Chi\Chiup
       \let\Psi\Psiup
       \let\Omega\Omegaup
       \let\Sampi\Sampiup
       \let\Digamma\Digammaup
       \let\varSigma\varSigmaup
    \else
       \let\Alpha\Alphait
       \let\Beta\Betait
       \let\Gamma\Gammait
       \let\Delta\Deltait
       \let\Epsilon\Epsilonit
       \let\Zeta\Zetait
       \let\Eta\Etait
       \let\Theta\Thetait
       \let\Iota\Iotait
       \let\Kappa\Kappait
       \let\Lambda\Lambdait
       \let\Mu\Muit
       \let\Nu\Nuit
       \let\Xi\Xiit
       \let\Omicron\Omicronit
       \let\Pi\Piit
       \let\Rho\Rhoit
       \let\Sigma\Sigmait
```

```
      \let\Tau\Tauit
      \let\Upsilon\Upsilonit
      \let\Phi\Phiit
      \let\Chi\Chiit
      \let\Psi\Psiit
      \let\Omega\Omegait
      \let\Sampi\Sampiit
      \let\Digamma\Digammait
      \let\varSigma\varSigmait
   \fi
   \iflgrmath@lower@up
      \let\alpha\alphaup
      \let\beta\betaup
      \let\gamma\gammaup
      \let\delta\deltaup
      \let\epsilon\epsilonup
      \let\zeta\zetaup
      \let\eta\etaup
      \let\theta\thetaup
      \let\iota\iotaup
      \let\kappa\kappaup
      \let\lambda\lambdaup
      \let\mu\muup
      \let\nu\nuup
      \let\xi\xiup
      \let\omicron\omicronup
      \let\pi\piup
      \let\rho\rhoup
      \let\sigma\sigmaup
      \let\tau\tauup
      \let\upsilon\upsilonup
      \let\phi\phiup
      \let\chi\chiup
      \let\psi\psiup
      \let\omega\omegaup
      \let\varsigma\varsigmaup
      \let\varvarsigma\varvarsigmaup
      \let\sampi\sampiup
      \let\digamma\digammaup
      \let\koppa\koppaup
```

The doc `macrocode` makes no provision for being interrupted invisibly, it is very complicated (but possible) to do this (see

https://github.com/latex3/latex2e/issues/847

), but simplest is to babble something here like this paragraph.

```
% \let\alphatonos\alphatonosup
% \let\epsilontonos\epsilontonosup
% \let\etatonos\etatonosup
% \let\iotatonos\iotatonosup
% \let\omicrontonos\omicrontonosup
```

```
  % \let\upsilontonos\upsilontonosup
  % \let\omegatonos\omegatonosup
  % \let\upsilondieresistonos\upsilondieresistonosup
  % \let\iotadieresis\iotadieresisup
  % \let\iotadieresistonos\iotadieresistonosup
  % \let\upsilondieresis\upsilondieresisup
```

babble

```
345     \else
346        \let\alpha\alphait
347        \let\beta\betait
348        \let\gamma\gammait
349        \let\delta\deltait
350        \let\epsilon\epsilonit
351        \let\zeta\zetait
352        \let\eta\etait
353        \let\theta\thetait
354        \let\iota\iotait
355        \let\kappa\kappait
356        \let\lambda\lambdait
357        \let\mu\muit
358        \let\nu\nuit
359        \let\xi\xiit
360        \let\omicron\omicronit
361        \let\pi\piit
362        \let\rho\rhoit
363        \let\sigma\sigmait
364        \let\tau\tauit
365        \let\upsilon\upsilonit
366        \let\phi\phiit
367        \let\chi\chiit
368        \let\psi\psiit
369        \let\omega\omegait
370        \let\varsigma\varsigmait
371        \let\varvarsigma\varvarsigmait
372        \let\sampi\sampiit
373        \let\digamma\digammait
374        \let\koppa\koppait
```

babble

```
  % \let\alphatonos\alphatonosit
  % \let\epsilontonos\epsilontonosit
  % \let\etatonos\etatonosit
  % \let\iotatonos\iotatonosit
  % \let\omicrontonos\omicrontonosit
  % \let\upsilontonos\upsilontonosit
  % \let\omegatonos\omegatonosit
  % \let\upsilondieresistonos\upsilondieresistonosit
  % \let\iotadieresis\iotadieresisit
  % \let\iotadieresistonos\iotadieresistonosit
```

```
  % \let\upsilondieresis\upsilondieresisit
```
babble
```
    \fi
}%
\lgrmath@setgreekcs
```
Finally we define `\lgrmathgreektable` and `\lgrmathgreektableextra`.
```
\def\lgrmathgreektable#1#2#3{%
\begingroup
\def\s##1{{\usefont{T1}{mlmtt}{m}{n}\string##1}}%
\usefont{LGR}{#1}{#2}{#3}%
\begin{tabular}{|lc|lc|lc|lc|}
\hline
\s\Alpha  &A &\s\Nu     &N &\s\alpha  &a &\s\nu     &n \\
\s\Beta   &B &\s\Xi     &X &\s\beta   &b &\s\xi     &x \\
\s\Gamma  &G &\s\Omicron&O &\s\gamma  &g &\s\omicron&o \\
\s\Delta  &D &\s\Pi     &P &\s\delta  &d &\s\pi     &p \\
\s\Epsilon&E &\s\Rho    &R &\s\epsilon&e &\s\rho    &r \\
\s\Zeta   &Z &\s\Sigma  &S &\s\zeta   &z &\s\sigma  &s \\
\s\Eta    &H &\s\Tau    &T &\s\eta    &h &\s\tau    &t \\
\s\Theta  &J &\s\Upsilon&U &\s\theta  &j &\s\upsilon&u \\
\s\Iota   &I &\s\Phi    &F &\s\iota   &i &\s\phi    &f \\
\s\Kappa  &K &\s\Chi    &Q &\s\kappa  &k &\s\chi    &q \\
\s\Lambda &L &\s\Psi    &Y &\s\lambda &l &\s\psi    &y \\
\s\Mu     &M &\s\Omega  &W &\s\mu     &m &\s\omega  &w \\
\hline
\end{tabular}
\endgroup
}%
\def\lgrmathgreektableextra#1#2#3{%
\begingroup
\def\s##1{{\usefont{T1}{mlmtt}{m}{n}\string##1}}%
\usefont{LGR}{#1}{#2}{#3}%
\begin{tabular}{lclclc}
\s\varsigma   &\char99   &\s\digamma&\char147 &\s\varSigma&\char22 \\
\s\varvarsigma&\char6    &\s\koppa  &\char18  &\s\Sampi   &\char23 \\
\s\sampi      &\char27   &          &         &\s\Digamma &\char195\relax
\end{tabular}
\endgroup
}%
```
And we have now reached the end of the **lgrmath** package code. The actual `.sty` file will contain an `\endinput` added by the DocStrip extraction.