

The **libgreek** package

Jean-François BURNOL

jfbu (at) free (dot) fr

This is 1.1 (2022/11/11) from source file having Time-stamp: <13-11-2022 at 17:14:21 CET>

Contents

1	Description	1
2	Documentation of the libgreek package	2
3	Documentation of the libgreek-legacy package	8
4	Release history	10
5	Implementation of the libgreek package	12
6	Implementation of the libgreek-legacy package	23

List of Tables

1	Greek letters via Libertinus Serif (upright)	4
2	Greek letters via Libertinus Serif (italic)	5
3	Greek letters via Libertinus Sans (upright)	6
4	Greek letters via Libertinus Sans (italic)	7
5	Greek letters via libgreek-legacy and Linux Libertine	11

1 Description

The **libgreek** package sets the Greek letters in math mode (**only**) to use glyphs ultimately from the Libertinus¹ Unicode font, as made available to the “8bit” traditional T_EX engines thanks to the **libertinus-type1** package². The package **libertinus-type1** is not loaded, but must be present.

Thus **libgreek** 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 **mathastext** or the **frenchmath** packages.

¹Khaled HOSNY originated the <https://github.com/alerque/libertinus> project, now maintained by Caleb MACLENNAN, as a fork of the **Linux Libertine** fonts designed by Philipp H. POLL.

²Bob TENNENT, *Support for using Libertinus fonts with L^AT_EX/pdfL^AT_EX*, <https://ctan.org/pkg/libertinus-type1>

Nota bene: complete math configuration is provided by the `libertinust1math` package³, and you very probably do not want using `libgreek` if already loading `libertinust1math`!

The Greek letters all come with `\dotsup` and `\dotsit` 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 `\libgreeksetup`. Further, the package optionally defines two math alphabets `\libgreekup` and `\libgreekit`. What ‘up’ and ‘it’ actually mean can be configured using the `upshape` and `itshape` keys at package loading time.

On old systems where the (now obsolete) `libertine-legacy`⁴ is installed, `libgreek` will switch automatically for backwards compatibility to a “legacy” mode which uses rather the font support files as provided by `libertine-legacy`. This “legacy” package `libgreek-legacy` uses almost unchanged the code from the original 2011 release.

On \LaTeX installations with `libertine-legacy` present you can author new documents using directly

```
\usepackage[options]{libgreek-legacy}
```

but they will not be portable to installations lacking `libertine-legacy`.

The “legacy” package does not provide the `\dotsup` and `\dotsit` control sequences nor does it provide the `\libgreekup` and `\libgreekit` math alphabets, or the `\libgreeksetup` command.

To author new documents which will always use `libertinus-type1` even if moved to systems having kept in some local `texmf` tree the obsolete (and not included in \TeX Live for some years) `libertine-legacy` package files, use

```
\usepackage[libertinus,options]{libgreek}
```

This is not needed if anyone of the options recognized only by `libgreek` and not by `libgreek-legacy` is used: i.e. `serif`, `sans`, `upshape`, `itshape` and alphabets.

2 Documentation of the `libgreek` package

Here are the options recognized by the package:

libertinus This option forces usage of the font support files from package `libertinus-type1` even if `libertine-legacy` is available on the \LaTeX installation.

serif Configures the package to use the (type1) Libertinus Serif font. This is the default behaviour (if the “legacy” mode is not activated).

libertine This is a deprecated alias for `serif`, which is provided for backwards compatibility with existing documents and does not inhibit the “legacy” mode: on a system having the `libertine-legacy` files it will end up being handed over to `libgreek-legacy`, except if some other option inhibits this.

³Michael SHARPE, *A Type 1 font and \LaTeX support for Libertinus Math*, <https://ctan.org/pkg/libertinust1math>.

⁴Bob TENNENT, *Linux Libertine fonts for \LaTeX and pdf \LaTeX users*, <https://ctan.org/pkg/libertine-legacy>

sans Configures the package to use the (type1) Libertinus Sans font.

biolinum Configures the package to use either the Libertinus Sans font or the (type1) Biolinum font if the “legacy” mode kicks in.

upshape=*<shape>* Declares the shape to be used by the `\dots up` Greek letters and the `\libgreekup` math alphabet. Defaults to ‘n’ (without the quotes).

itshape=*<shape>* Declares the shape to be used by the `\dots it` Greek letters and the `\libgreekit` math alphabet. Defaults to ‘it’.

scale=*<factor>* will scale the font by the given factor, relative to nominal size.

Example: `scale=1.2` will scale by 20%.

style=*<ISO|French|TeX>* specifies the shape style of the Greek letters.

ISO means ‘italic’ for lowercase and uppercase, **French** means ‘upright’ for lowercase and uppercase, **TeX** means ‘italic’ for lowercase and ‘upright’ for uppercase.

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`.

For backwards compatibility with the behaviour of the 2011 release, 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=French`.

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). Alternatively, do not use the `greek/Greek` keys but set `upshape=it`, `itshape=n`, as the package by default assigns ‘italic’ to lowercase and ‘upright’ to uppercase.

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

For backwards compatibility with the behaviour of the 2011 release, 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.

It is however recommended to use rather the `upshape|itshape|style` keys.

Using `greek=up` or `Greek=up` does not inhibit “legacy” mode although the legacy package did not a priori accept that value ‘up’. But since \LaTeX 2020-02-02, the ‘up’ shape is recognized and triggers `up`→`n` substitutions, so one can use that syntax even in contexts which will let the package fall back to “legacy” mode. Not recommended, though.

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 `\libgreekup` and `\libgreekit`.

<code>\alphatonos</code>	ά	<code>\upsilontonos</code>	ύ	<code>\iotadieresis</code>	ι̇
<code>\epsilontonos</code>	έ	<code>\omegatonos</code>	ώ	<code>\iotadieresis</code>	ι̇
<code>\etatonos</code>	ή	<code>\upsilondieresis</code>	υ̇	<code>\upsilondieresis</code>	υ̇
<code>\iotatonos</code>	ί	<code>\iotaadieresis</code>	ϊ̇		
<code>\omicrontonos</code>	ό				

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

<code>\varsigma</code>	ς	<code>\digamma</code>	Ϝ	<code>\Sampi</code>	Ϡ
<code>\varvarsigma</code>	ς	<code>\koppa</code>	Ϝ	<code>\Digamma</code>	F
<code>\sampi</code>	Ϡ	<code>\varSigma</code>	Σ		

Table 1: Greek letters from LibertinusSerif in LGR encoding, upright shapes

<code>\alphatonos</code>	$\acute{\alpha}$	<code>\upsilontonos</code>	$\acute{\upsilon}$	<code>\iotadieresis</code>	$\text{\textcircled{\scriptsize i}}$
<code>\epsilontonos</code>	$\acute{\epsilon}$	<code>\omegatonos</code>	$\acute{\omega}$	<code>\iota</code>	$\text{\textcircled{\scriptsize i}}$
<code>\etatonos</code>	$\acute{\eta}$	<code>\upsilondieresis</code>	$\text{\textcircled{\scriptsize i}}$	<code>\upsilondieresis</code>	$\text{\textcircled{\scriptsize i}}$
<code>\iotatonos</code>	$\acute{\iota}$	<code>\iota</code>	$\text{\textcircled{\scriptsize i}}$		
<code>\omicrontonos</code>	$\acute{\omicron}$	<code>\iotadieresis</code>	$\text{\textcircled{\scriptsize i}}$		

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

<code>\varsigma</code>	ς	<code>\digamma</code>	$\text{\textcircled{\scriptsize f}}$	<code>\Sampi</code>	$\text{\textcircled{\scriptsize s}}$
<code>\varvarsigma</code>	ς	<code>\koppa</code>	$\text{\textcircled{\scriptsize s}}$	<code>\Digamma</code>	$\text{\textcircled{\scriptsize f}}$
<code>\sampi</code>	$\text{\textcircled{\scriptsize s}}$	<code>\varSigma</code>	ς		

Table 2: Greek letters from LibertinusSerif in LGR encoding, italic shapes

<code>\alphatonos</code>	ά	<code>\upsilontonos</code>	ύ	<code>\iotadieresis</code>	ι̇
<code>\epsilontonos</code>	έ	<code>\omegatonos</code>	ώ	<code>\iotaadieresis</code>	ι̇
<code>\etatonos</code>	ή	<code>\upsilondieresis</code>	υ̇	<code>\upsilondieresis</code>	υ̇
<code>\iotatonos</code>	ί	<code>\iotaadieresis</code>	ϊ		
<code>\omicrontonos</code>	ό				

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

<code>\varsigma</code>	ς	<code>\digamma</code>	Ϝ	<code>\Sampi</code>	Ϻ
<code>\varvarsigma</code>	ς	<code>\koppa</code>	Ϝ	<code>\Digamma</code>	Ϝ
<code>\sampi</code>	Ϻ	<code>\varSigma</code>	Σ		

Table 3: Greek letters from LibertinusSans in LGR encoding, upright shapes

<code>\alphatonos</code>	$\acute{\alpha}$	<code>\upsilontonos</code>	$\acute{\upsilon}$	<code>\iotadieresis</code>	$\text{\textcircled{iota}}$
<code>\epsilontonos</code>	$\acute{\epsilon}$	<code>\omegatonos</code>	$\acute{\omega}$	<code>\iota</code>	$\text{\textcircled{iota}}$
<code>\etatonos</code>	$\acute{\eta}$	<code>\upsilondieresis</code>	$\text{\textcircled{upsilon}}$	<code>\upsilondieresis</code>	$\text{\textcircled{upsilon}}$
<code>\iotatonos</code>	$\acute{\iota}$	<code>\iota</code>	$\text{\textcircled{iota}}$		
<code>\omicrontonos</code>	$\acute{\omicron}$	<code>\iotadieresis</code>	$\text{\textcircled{iota}}$		

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

<code>\varsigma</code>	ς	<code>\digamma</code>	$\text{\textcircled{gamma}}$	<code>\Sampi</code>	$\text{\textcircled{gamma}}$
<code>\varvarsigma</code>	ς	<code>\koppa</code>	$\text{\textcircled{gamma}}$	<code>\Digamma</code>	F
<code>\sampi</code>	$\text{\textcircled{gamma}}$	<code>\varSigma</code>	ς		

Table 4: Greek letters from LibertinusSans in LGR encoding, italic shapes

Here are the commands defined by the package:

`\libgreeksetup{⟨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 `\libgreeksetup` using only the `greek` and/or `Greek` keys.

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

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

Miscellaneous remarks:

1. Even if not receiving the option `alphabets`, the package will declare all Greek letters to be of type `\mathalpha`. The legacy package on the other hand declares them all to be of type `\mathord`.
2. The `libgreek` 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 `\libgreeksetup` in the preamble or body). The `libgreek`-legacy package on the other hand handles global class options if it recognizes them.

3 Documentation of the `libgreek`-legacy package

Except for a fix to a problem with `\varvarsigma` and `\varvarkappa`, and the handling of option `libertine` (which formerly would have remained effectless if used after `biolinum`) `libgreek`-legacy is functionally equivalent with the original 1.0 release of `libgreek` from 2011/03/14.

Compared to the 2022 `libgreek`, and apart from the fact that the legacy mode relies on presence of the files of `libertine-legacy`, the main difference is that this legacy version does not define `\dotsup` and `\dotsit` shape variants, does not provide `\libgreekup` and `\libgreekit` math alphabets, and does not define a command `\libgreeksetup` allowing to change the shapes midway in a document. All Greek letters are defined as math symbols of type `\mathord`, whereas the 2022 `libgreek` uses type `\mathalpha`.

Also, the legacy package defines even more control sequences for letters with diacritics or special variant glyphs, than does the 2022 `libgreek` relying on `libertinus-type1` LGR-encoded fonts.

Now to the description of the recognized options.

With the exception of `libertine` and `biolinum`, all options are of the `key=value` type.

libertine Configures the package to use the (type1) Libertine font (i.e. the serif font). This is the default behaviour.

biolinum Configures the package to use the (type1) Biolinum font (i.e. the sans serif font).

scale=*<factor>* will scale the font by the given factor, relative to nominal size (when the Libertine/Biolinum fonts are used elsewhere in the document, they will also be scaled by this factor). Note that the similar option of the `libertine-legacy` package is called `scaled` and has precisely the same effect. Example: `scale=1.2` will scale by 20%.

style=*<ISO|French|TeX>* specifies the shape of the Greek letters. `ISO` means italic for lowercase and uppercase, `French` means upright for lowercase and uppercase, `TeX` means italic for lowercase and upright for uppercase. This option will override any `greek` or `Greek` option. The package defaults to `style=TeX`.

greek=*<shape>* specifies the shape (`n`, `it`, or `sl`) for both the lowercase and uppercase Greek letters. So `greek=it` is like `style=ISO`, and `greek=n` is like `style=French`.

Greek=*<shape>* specifies the shape (`n`, `it`, or `sl`) for only the uppercase Greek letters. To have lowercase upright and uppercase italic, use `greek=n,Greek=it`.

series=*<series>* tells which series to use. The default is the value of `\seriesdefault` at the time of loading the package. See the `libertine-legacy` package documentation for the admissible values (they include `m`, `b`, `bx` (`=b`) for Libertine and `m`, `b`, `bx` (`=b`), `o`, `s` for Biolinum).

boldseries=*<series>* tells which series to use in bold math. Default is `\bfdefault` at the time of loading the package.

(1) the bold italic Greek Libertine glyphs are missing from `libertine v5` and later versions up to `libertine-legacy` (or some bug in the font definition files triggers usage of `TEX` font files not having the glyphs): the log file will contain “Missing character” entries, and the PDF will simply display nothing. With `\tracinglostchars=3` and `TEXLive 2021` or later, these “Missing character”’s will be reported as build errors by `TEX`, as they should. Use bold slanted instead.

(2) the bold lowercase Greek Biolinum letters are in fact not bold.

Advanced example of use (we use slanted rather than italic to circumvent the problem mentioned above):

```
\documentclass{article}
\usepackage[scale=2,series=b,greek=sl,Greek=n]{libgreek}
```

```

\begin{document}
$\alpha\beta\gamma\phi\psi \ \Alpha\Beta\Gamma\Phi\Psi$
\end{document}

```

As shown in the [Table 5](#), the package also defines control sequences to access quite a few letters with diacritics. The displayed glyphs correspond to using `libgreek-legacy` with its default, i.e. the serif font (Libertine) is used, the lowercase italic letters are italic, and the uppercase are upright.

4 Release history

`libgreek` was initially developed on a computer equipped with T_EXLive 2010 and was released on March 14, 2011. It got a documentation update on September 23, 2012, but its source code and version number 1.0 did not change then.

The initial release used files as included in the v4 version of the `libertine` package then maintained by Michael NIEDERMAIR and which allowed traditional non-Unicode T_EX engines to access glyphs of the Unicode Linux Libertine fonts as designed by Philipp H. POLL.

By the time of T_EXLive 2012 (and even earlier), the files needed by `libgreek` had been moved to the `libertine-legacy` package (maintained by Bob TENNENT). This documentation was updated in September 2012 to mention this.

But around circa 2018 (I think) `libertine-legacy` package was moved to the “Obsolete” section of CTAN, and stopped being included in new T_EXLive releases. This means that `libgreek`, although still included in T_EXLive, was actually broken on new installations. Theoretically the user could still install `libertine-legacy` manually, but this is not easy as it is not distributed as a `.tds.zip` archive on CTAN mirrors, and furthermore the `libertine.map` may need to be renamed before being listed in some `updmap.cfg` file, else it would clash with the `libertine.map` from the current `libertine` package (maintained by Bob TENNENT, and which now can be used on all engines as it auto-detects the context).

This situation, and the fact that the `libertinus-type1` package (also maintained by Bob TENNENT, thanks Bob!) provides support files for the LGR encoding, thus allowing to access Greek glyphs, motivated the 2022 release of `libgreek`.

Thanks to Antoine MISSIER (author of `frenchmath`) for making the author aware that `libgreek` had ceased functioning and was in need of an update, and indeed for quite some years already.

<code>\Alphatonos</code>	Α	<code>\varUpsilondieresis</code>	ÿ	<code>\upsilondieresis</code>	ÿ
<code>\Epsilontonos</code>	Ε	<code>\Omegatonos</code>	Ω	<code>\upsilondieresisistonos</code>	
<code>\Etatonos</code>	Η	<code>\alphatonos</code>	ά	<code>\omicrontonos</code>	ό
<code>\Iotadieresis</code>	ϊ	<code>\epsilontonos</code>	έ	<code>\omegatonos</code>	ώ
<code>\Iotatonos</code>	Ι	<code>\etatonos</code>	ή	<code>\anoteleia</code>	·
<code>\Omicrontonos</code>	Ο	<code>\iotatonos</code>	ί	<code>\tonos</code>	´
<code>\Upsilontonos</code>	Υ	<code>\iotadieresis</code>	ϊ	<code>\dieresisistonos</code>	¨
<code>\varUpsilontonos</code>	Υ	<code>\iotadieresisistonos</code>	ι̇		
<code>\Upsilondieresis</code>	ÿ	<code>\upsilontonos</code>	ύ		

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

<code>\varsigma</code>	ς	<code>\varepsilon</code>	ε	<code>\varTheta</code>	Θ
<code>\varvarsigma</code>	ς	<code>\reversedvarepsilon</code>		<code>\varUpsilon</code>	Υ
<code>\vartheta</code>	ϑ	ε		<code>\Digamma</code>	Ϝ
<code>\varphi</code>	φ	<code>\varrho</code>	ρ	<code>\Koppa</code>	Ϟ
<code>\varpi</code>	ω	<code>\digamma</code>	Ϝ	<code>\Sampi</code>	Ϸ
<code>\varkappa</code>	κ	<code>\koppa</code>	Ϟ		
<code>\varvarkappa</code>	κ	<code>\sampi</code>	Ϸ		

Table 5: Greek letters via libgreek-legacy and default configuration (i.e. no options, or equivalently options `[libertine,style=TeX]` giving upright uppercase, italic lowercase, and usage of the Libertine glyphs). In the 1.0 **libgreek** release of 2011 the `\varvarkappa` and `\varvarsigma` would have been upright like the uppercase letters, due to a bug which was fixed for the 2022 transformation into **libgreek**-legacy.

5 Implementation of the **libgreek** package

```
1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{libgreek}
3     [2022/11/11 1.1 Greek in math mode via libertine-type1 (JFB)]
4 \@ifpackageloaded{libgreek-legacy}{%
5     \PackageWarning{libgreek}%
6     {libgreek-legacy loaded previously, now aborting\@gobble}%
7     \endinput
8 }{}%
```

We will use [kvoptions](#) to handle options with key=value syntax. First let's check for some hint that [libertine-legacy](#) is present and then hand over to the `libgreek-legacy`, but only if only legacy options have been used.

```
9 \RequirePackage{kvoptions}
10 \IfFileExists{ufxl03.fd}{%
11     \define@key{libgreek}{libertinus}[0]{\global\let\libgreek@nolegacy\@empty}
12     \define@key{libgreek}{serif}[0]{\global\let\libgreek@nolegacy\@empty}
13     \define@key{libgreek}{libertine}[0]{}
14     \define@key{libgreek}{sans}[0]{\global\let\libgreek@nolegacy\@empty}
15     \define@key{libgreek}{biolinum}[0]{}
16     \define@key{libgreek}{scale}[0]{}
17     \define@key{libgreek}{upshape}[0]{\global\let\libgreek@nolegacy\@empty}
18     \define@key{libgreek}{itshape}[0]{\global\let\libgreek@nolegacy\@empty}
19     \define@key{libgreek}{style}[0]{}
20     \define@key{libgreek}{Greek}[0]{}
21     \define@key{libgreek}{greek}[0]{}
22     \define@key{libgreek}{series}[0]{}
23     \define@key{libgreek}{boldseries}[0]{}
24     \define@key{libgreek}{alphabets}[0]{\global\let\libgreek@nolegacy\@empty}
```

Good that this will do only scope-obeying things.

```
25     {\ProcessLocalKeyvalOptions*}%
26     \ifdefined\libgreek@nolegacy\expandafter\@gobble\else\expandafter\@firstofone\fi
27     {%
28     \PackageInfo{libgreek}{%
29     *****\MessageBreak
30     File ufxl03.fd exists!\MessageBreak
31     Thus, now handing over to libgreek-legacy.\MessageBreak
32     To avoid this, use libertine option.\MessageBreak
33     *****\@gobbletwo
34     }%
```

Let's transfer all options to `libgreek-legacy` without trying some shortcut.

```
35     \DeclareOption*{\PassOptionsToPackage{\CurrentOption}{libgreek-legacy}}%
36     \ProcessOptions\relax
37     \endinput
38     \RequirePackage{libgreek-legacy}%
39     }%
40 }{}%
```

To minimize the author’s task, we keep close to original 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 `\libgreeksetup`, which requires to keep a trace of various things, for example if `style` option is used at package level and then later on using `\libgreeksetup` the user employs the `Greek/greek` options.

Although the user interface is kept in a maximally backwards compatible way (i.e. even if not falling back to “legacy” the option behaviour should be stable), the underlying architecture of the symbol fonts is completely different because this package assigns two symbol fonts, one for upright, the other one for italic-like, whereas the legacy package assigns one or two symbol fonts depending on whether lowercase and uppercase letters have been assigned or not the same shape. In the legacy package the symbol fonts are mapped to either lowercase or uppercase, but in this more modern package, the symbol fonts are for the ‘upright’ and ‘italic’ shapes.

The new `upshape` and `itshape` keys allow to configure what the `\dotsup` and `\dotsit` macros will actually use as shapes.

```
41 \def\libgreek@fontfamily{LibertinusSerif}
42 \def\libgreek@scale{1}
43 \def\libgreek@upshape{n}
44 \def\libgreek@itshape{it}
45 \newif\iflibgreek@upper@up\libgreek@upper@uptrue
46 \newif\iflibgreek@lower@up
47 \edef\libgreek@series{\seriesdefault}
48 \edef\libgreek@boldseries{\bfdefault}
49 \def\libgreek@upper@shape{\libgreek@upshape}
50 \def\libgreek@lower@shape{\libgreek@itshape}
51 \def\libgreek@style{TeX}
52 \newif\iflibgreek@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... however this means that inputs such as `serif=false` do not generate errors but are handled exactly the same as `serif=true`...

```
53 \define@key{libgreek}{libertinus}[true]{}
54 \define@key{libgreek}{serif}[true]{\def\libgreek@fontfamily{LibertinusSerif}}
55 \define@key{libgreek}{libertine}[true]{\def\libgreek@fontfamily{LibertinusSerif}}
56 \define@key{libgreek}{sans}[true]{\def\libgreek@fontfamily{LibertinusSans}}
57 \define@key{libgreek}{biolinum}[true]{\def\libgreek@fontfamily{LibertinusSans}}
58 \define@key{libgreek}{scale}[1]{\def\libgreek@scale{#1}}
59 \define@key{libgreek}{upshape}{\edef\libgreek@upshape{#1}}
60 \define@key{libgreek}{itshape}{\edef\libgreek@itshape{#1}}
61 \define@key{libgreek}{style}{\edef\libgreek@style{#1}\libgreek@stytrue}
62 \define@key{libgreek}{Greek}{\edef\libgreek@upper@shape{#1}}
63 \define@key{libgreek}{greek}{\edef\libgreek@lower@shape{#1}%
64 \edef\libgreek@upper@shape{#1}}
65 \define@key{libgreek}{series}{\edef\libgreek@series{#1}}
66 \define@key{libgreek}{boldseries}{\edef\libgreek@boldseries{#1}}
```

The single truly Boolean option...

```
67 \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 `\libgreeksetup`.

```

68 \def\libgreek@style@ISO{%
69   \libgreek@upper@upfalse
70   \libgreek@lower@upfalse
71   \let\libgreek@upper@shape\libgreek@itshape
72   \let\libgreek@lower@shape\libgreek@itshape
73 }
74 \def\libgreek@style@French{%
75   \libgreek@upper@uptrue
76   \libgreek@lower@uptrue
77   \let\libgreek@upper@shape\libgreek@upshape
78   \let\libgreek@lower@shape\libgreek@upshape
79 }
80 \def\libgreek@style@TeX{%
81   \libgreek@upper@uptrue
82   \libgreek@lower@upfalse
83   \let\libgreek@upper@shape\libgreek@upshape
84   \let\libgreek@lower@shape\libgreek@itshape
85 }

```

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

```

86 \def\libgreek@process@style{%
87   \libgreek@styfalse
88   \ifcsname libgreek@style@\libgreek@style\endcsname
89     \csname libgreek@style@\libgreek@style\endcsname
90   \else
91     \PackageWarning{libgreek}{Unknown style `'\libgreek@style'}`%
92   \fi
93 }

```

This stuff is a bit involved, but this is done for maximal backwards compatibility in terms of stability of behaviour (of course the real maximal stability is when the “legacy” mode gets activated).

```

94 \def\libgreek@process@shapes{%
95   \edef\libgreek@upper@shape{\libgreek@upper@shape}%
96   \edef\libgreek@lower@shape{\libgreek@lower@shape}%
97   \ifx\libgreek@upper@shape\libgreek@upshape
98     \libgreek@upper@uptrue
99   \else
100     \ifx\libgreek@upper@shape\libgreek@itshape
101       \libgreek@upper@upfalse
102     \else
103       \expandafter\in@\expandafter{\expandafter.\libgreek@upper@shape,}{.up,}%
104       \ifin@\libgreek@upper@uptrue
105     \else
106       \expandafter\in@\expandafter{\expandafter.\libgreek@upper@shape,}{.it,}%
107       \ifin@\libgreek@upper@upfalse
108     \else
109       \libgreek@process@upper@lastresort
110   \fi\fi\fi\fi

```

```

111 \ifx\libgreek@lower@shape\libgreek@itshape
112 \libgreek@lower@upfalse
113 \else
114 \ifx\libgreek@lower@shape\libgreek@upshape
115 \libgreek@lower@uptrue
116 \else
117 \expandafter\in@\expandafter{\expandafter.\libgreek@lower@shape,}{.it,}%
118 \ifin@\libgreek@lower@upfalse
119 \else
120 \expandafter\in@\expandafter{\expandafter.\libgreek@lower@shape,}{.up,}%
121 \ifin@\libgreek@lower@uptrue
122 \else
123 \libgreek@process@lower@lastresort
124 \fi\fi\fi\fi
125 }%
126 \def\libgreek@process@upper@lastresort{%
127 \libgreek@upper@uptrue
128 \let\libgreek@upshape\libgreek@upper@shape
129 }
130 \def\libgreek@process@lower@lastresort{%
131 \libgreek@lower@upfalse
132 \let\libgreek@itshape\libgreek@lower@shape
133 }

```

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

```
134 \ProcessLocalKeyvalOptions*
```

This is as per the `libertinus-type1.sty` code.

```
135 \expandafter\edef\csname\libgreek@fontfamily @scale\endcsname{\libgreek@scale}
```

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

```

136 \iflibgreek@sty
137 \libgreek@process@style
138 \else
139 \libgreek@process@shapes
140 \fi
141 \def\libgreek@process@upper@lastresort{%
142 \PackageWarning{libgreek}{%
143 \hspace{10em}Too late for the shape `\'libgreek@upper@shape'\MessageBreak
144 \hspace{10em}originating in Greek or greek option. Ignored.\MessageBreak
145 \hspace{10em}Use `up' or `it'}%
146 }
147 \def\libgreek@process@lower@lastresort{%
148 \PackageWarning{libgreek}{%
149 \hspace{10em}Too late for the shape `\'libgreek@lower@shape'\MessageBreak
150 \hspace{10em}originating in greek option. Ignored.\MessageBreak
151 \hspace{10em}Use `up' or `it'}%

```

```

152 }
153 \def\libgreeksetup#1{%
154   \setkeys{libgreek}{#1}%
155   \iflibgreek@sty\libgreek@process@style\else\libgreek@process@shapes\fi
156   \libgreek@setgreekcs
157 }

```

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

```

158 \DisableKeyvalOption{libgreek}{libertinus}
159 \DisableKeyvalOption{libgreek}{serif}
160 \DisableKeyvalOption{libgreek}{libertine}
161 \DisableKeyvalOption{libgreek}{sans}
162 \DisableKeyvalOption{libgreek}{biolinum}
163 \DisableKeyvalOption{libgreek}{scale}
164 \DisableKeyvalOption{libgreek}{upshape}
165 \DisableKeyvalOption{libgreek}{itshape}
166 \DisableKeyvalOption{libgreek}{series}
167 \DisableKeyvalOption{libgreek}{boldseries}
168 \DisableKeyvalOption{libgreek}{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...

```

169 \DeclareFontEncoding{LGR}{-}{-}
170 \DeclareSymbolFont{libgreekup}{LGR}{\libgreek@fontfamily-TLF}
171   {\libgreek@series}
172   {\libgreek@upshape}
173 \SetSymbolFont{libgreekup}{bold}{LGR}{\libgreek@fontfamily-TLF}
174   {\libgreek@boldseries}
175   {\libgreek@upshape}
176 \DeclareSymbolFont{libgreekit}{LGR}{\libgreek@fontfamily-TLF}
177   {\libgreek@series}
178   {\libgreek@itshape}
179 \SetSymbolFont{libgreekit}{bold}{LGR}{\libgreek@fontfamily-TLF}
180   {\libgreek@boldseries}
181   {\libgreek@itshape}

```

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

```

182 \iflibgreek@alphabets
183   \DeclareSymbolFontAlphabet{\libgreekup}{libgreekup}
184   \DeclareSymbolFontAlphabet{\libgreekit}{libgreekit}
185 \fi

```

Definition of the ‘up’ `\mathchar`’s. There are 48 ‘standard’ ones plus 8 extras and 11 with diacritics for a total of 67 ones.

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

```

186 \DeclareMathSymbol{\Alphaup}{\mathalpha}{libgreekup}{65}
187 \DeclareMathSymbol{\Betaup}{\mathalpha}{libgreekup}{66}

```



```

188 \DeclareMathSymbol{\Gammaup}{\mathalpha}{libgreekup}{71}
189 \DeclareMathSymbol{\Deltaup}{\mathalpha}{libgreekup}{68}
190 \DeclareMathSymbol{\Epsilonup}{\mathalpha}{libgreekup}{69}
191 \DeclareMathSymbol{\Zetaup}{\mathalpha}{libgreekup}{90}
192 \DeclareMathSymbol{\Etaup}{\mathalpha}{libgreekup}{72}
193 \DeclareMathSymbol{\Thetaup}{\mathalpha}{libgreekup}{74}
194 \DeclareMathSymbol{\Iotaup}{\mathalpha}{libgreekup}{73}
195 \DeclareMathSymbol{\Kappaup}{\mathalpha}{libgreekup}{75}
196 \DeclareMathSymbol{\Lambdaup}{\mathalpha}{libgreekup}{76}
197 \DeclareMathSymbol{\Muup}{\mathalpha}{libgreekup}{77}
198 \DeclareMathSymbol{\Nuup}{\mathalpha}{libgreekup}{78}
199 \DeclareMathSymbol{\Xiup}{\mathalpha}{libgreekup}{88}
200 \DeclareMathSymbol{\Omicronup}{\mathalpha}{libgreekup}{79}
201 \DeclareMathSymbol{\Piup}{\mathalpha}{libgreekup}{80}
202 \DeclareMathSymbol{\Rhoupp}{\mathalpha}{libgreekup}{82}
203 \DeclareMathSymbol{\Sigmaup}{\mathalpha}{libgreekup}{83}
204 \DeclareMathSymbol{\Tauup}{\mathalpha}{libgreekup}{84}
205 \DeclareMathSymbol{\Upsilonup}{\mathalpha}{libgreekup}{85}
206 \DeclareMathSymbol{\Phiup}{\mathalpha}{libgreekup}{70}
207 \DeclareMathSymbol{\Chiup}{\mathalpha}{libgreekup}{81}
208 \DeclareMathSymbol{\Psiup}{\mathalpha}{libgreekup}{89}
209 \DeclareMathSymbol{\Omegaup}{\mathalpha}{libgreekup}{87}
210 \DeclareMathSymbol{\alphaup}{\mathalpha}{libgreekup}{97}
211 \DeclareMathSymbol{\betaup}{\mathalpha}{libgreekup}{98}
212 \DeclareMathSymbol{\gammaup}{\mathalpha}{libgreekup}{103}
213 \DeclareMathSymbol{\deltaup}{\mathalpha}{libgreekup}{100}
214 \DeclareMathSymbol{\epsilonup}{\mathalpha}{libgreekup}{101}
215 \DeclareMathSymbol{\zetaup}{\mathalpha}{libgreekup}{122}
216 \DeclareMathSymbol{\etaup}{\mathalpha}{libgreekup}{104}
217 \DeclareMathSymbol{\thetaup}{\mathalpha}{libgreekup}{106}
218 \DeclareMathSymbol{\iotaup}{\mathalpha}{libgreekup}{105}
219 \DeclareMathSymbol{\kappaup}{\mathalpha}{libgreekup}{107}
220 \DeclareMathSymbol{\lambdaup}{\mathalpha}{libgreekup}{108}
221 \DeclareMathSymbol{\muup}{\mathalpha}{libgreekup}{109}
222 \DeclareMathSymbol{\nuup}{\mathalpha}{libgreekup}{110}
223 \DeclareMathSymbol{\xiup}{\mathalpha}{libgreekup}{120}
224 \DeclareMathSymbol{\omicronup}{\mathalpha}{libgreekup}{111}
225 \DeclareMathSymbol{\piup}{\mathalpha}{libgreekup}{112}
226 \DeclareMathSymbol{\rhoup}{\mathalpha}{libgreekup}{114}
227 \DeclareMathSymbol{\sigmaup}{\mathalpha}{libgreekup}{115}
228 \DeclareMathSymbol{\tauup}{\mathalpha}{libgreekup}{116}
229 \DeclareMathSymbol{\upsilonup}{\mathalpha}{libgreekup}{117}
230 \DeclareMathSymbol{\phiup}{\mathalpha}{libgreekup}{102}
231 \DeclareMathSymbol{\chiup}{\mathalpha}{libgreekup}{113}
232 \DeclareMathSymbol{\psiup}{\mathalpha}{libgreekup}{121}
233 \DeclareMathSymbol{\omegaup}{\mathalpha}{libgreekup}{119}
Defintion of the 'it' \mathchar's.
234 \DeclareMathSymbol{\Alphait}{\mathalpha}{libgreekit}{65}
235 \DeclareMathSymbol{\Betait}{\mathalpha}{libgreekit}{66}

```

```

236 \DeclareMathSymbol{\Gammait}{\mathalpha}{libgreekit}{71}
237 \DeclareMathSymbol{\Deltait}{\mathalpha}{libgreekit}{68}
238 \DeclareMathSymbol{\Epsilont}{\mathalpha}{libgreekit}{69}
239 \DeclareMathSymbol{\Zetai}{\mathalpha}{libgreekit}{90}
240 \DeclareMathSymbol{\Etai}{\mathalpha}{libgreekit}{72}
241 \DeclareMathSymbol{\Thetait}{\mathalpha}{libgreekit}{74}
242 \DeclareMathSymbol{\Iotait}{\mathalpha}{libgreekit}{73}
243 \DeclareMathSymbol{\Kappait}{\mathalpha}{libgreekit}{75}
244 \DeclareMathSymbol{\Lambdait}{\mathalpha}{libgreekit}{76}
245 \DeclareMathSymbol{\Mu}{\mathalpha}{libgreekit}{77}
246 \DeclareMathSymbol{\Nu}{\mathalpha}{libgreekit}{78}
247 \DeclareMathSymbol{\Xi}{\mathalpha}{libgreekit}{88}
248 \DeclareMathSymbol{\Omicron}{\mathalpha}{libgreekit}{79}
249 \DeclareMathSymbol{\Pi}{\mathalpha}{libgreekit}{80}
250 \DeclareMathSymbol{\Rho}{\mathalpha}{libgreekit}{82}
251 \DeclareMathSymbol{\Sigma}{\mathalpha}{libgreekit}{83}
252 \DeclareMathSymbol{\Tau}{\mathalpha}{libgreekit}{84}
253 \DeclareMathSymbol{\Upsilon}{\mathalpha}{libgreekit}{85}
254 \DeclareMathSymbol{\Phi}{\mathalpha}{libgreekit}{70}
255 \DeclareMathSymbol{\Chi}{\mathalpha}{libgreekit}{81}
256 \DeclareMathSymbol{\Psi}{\mathalpha}{libgreekit}{89}
257 \DeclareMathSymbol{\Omega}{\mathalpha}{libgreekit}{87}
258 \DeclareMathSymbol{\alpha}{\mathalpha}{libgreekit}{97}
259 \DeclareMathSymbol{\beta}{\mathalpha}{libgreekit}{98}
260 \DeclareMathSymbol{\gamma}{\mathalpha}{libgreekit}{103}
261 \DeclareMathSymbol{\delta}{\mathalpha}{libgreekit}{100}
262 \DeclareMathSymbol{\epsilon}{\mathalpha}{libgreekit}{101}
263 \DeclareMathSymbol{\zeta}{\mathalpha}{libgreekit}{122}
264 \DeclareMathSymbol{\eta}{\mathalpha}{libgreekit}{104}
265 \DeclareMathSymbol{\theta}{\mathalpha}{libgreekit}{106}
266 \DeclareMathSymbol{\iota}{\mathalpha}{libgreekit}{105}
267 \DeclareMathSymbol{\kappa}{\mathalpha}{libgreekit}{107}
268 \DeclareMathSymbol{\lambda}{\mathalpha}{libgreekit}{108}
269 \DeclareMathSymbol{\mu}{\mathalpha}{libgreekit}{109}
270 \DeclareMathSymbol{\nu}{\mathalpha}{libgreekit}{110}
271 \DeclareMathSymbol{\xi}{\mathalpha}{libgreekit}{120}
272 \DeclareMathSymbol{\omicron}{\mathalpha}{libgreekit}{111}
273 \DeclareMathSymbol{\pi}{\mathalpha}{libgreekit}{112}
274 \DeclareMathSymbol{\rho}{\mathalpha}{libgreekit}{114}
275 \DeclareMathSymbol{\sigma}{\mathalpha}{libgreekit}{115}
276 \DeclareMathSymbol{\tau}{\mathalpha}{libgreekit}{116}
277 \DeclareMathSymbol{\upsilon}{\mathalpha}{libgreekit}{117}
278 \DeclareMathSymbol{\phi}{\mathalpha}{libgreekit}{102}
279 \DeclareMathSymbol{\chi}{\mathalpha}{libgreekit}{113}
280 \DeclareMathSymbol{\psi}{\mathalpha}{libgreekit}{121}
281 \DeclareMathSymbol{\omega}{\mathalpha}{libgreekit}{119}
  Alternate shapes and other glyphs.
282 \DeclareMathSymbol{\varsigmaup}{\mathalpha}{libgreekup}{99}
283 \DeclareMathSymbol{\varvarsigmaup}{\mathalpha}{libgreekup}{6}

```

```

284 \DeclareMathSymbol{\varSigmaup}{\mathalpha}{libgreekup}{22}
285 \DeclareMathSymbol{\Sampiup}{\mathalpha}{libgreekup}{23}
286 \DeclareMathSymbol{\sampiup}{\mathalpha}{libgreekup}{27}
287 \DeclareMathSymbol{\digammaup}{\mathalpha}{libgreekup}{147}
288 \DeclareMathSymbol{\Digammaup}{\mathalpha}{libgreekup}{195}
289 \DeclareMathSymbol{\koppaup}{\mathalpha}{libgreekup}{18}
290 \DeclareMathSymbol{\varsigmaait}{\mathalpha}{libgreekit}{99}
291 \DeclareMathSymbol{\varvarsigmaait}{\mathalpha}{libgreekit}{6}
292 \DeclareMathSymbol{\varSigmaait}{\mathalpha}{libgreekit}{22}
293 \DeclareMathSymbol{\Sampiit}{\mathalpha}{libgreekit}{23}
294 \DeclareMathSymbol{\sampiit}{\mathalpha}{libgreekit}{27}
295 \DeclareMathSymbol{\digammait}{\mathalpha}{libgreekit}{147}
296 \DeclareMathSymbol{\Digammait}{\mathalpha}{libgreekit}{195}
297 \DeclareMathSymbol{\koppait}{\mathalpha}{libgreekit}{18}

```

Some glyphs with diacritics. I hesitated keeping this, which comes from some file I had constituted in September 2011 to use LGR encoded fonts, and which I am reusing here. I hesitated also defining for them both ‘up’ and ‘it’ shapes.

```

298 \DeclareMathSymbol{\alphatonosup}{\mathalpha}{libgreekup}{136}
299 \DeclareMathSymbol{\epsilontonosup}{\mathalpha}{libgreekup}{232}
300 \DeclareMathSymbol{\etatonosup}{\mathalpha}{libgreekup}{160}
301 \DeclareMathSymbol{\iotatonosup}{\mathalpha}{libgreekup}{208}
302 \DeclareMathSymbol{\omicrontonosup}{\mathalpha}{libgreekup}{236}
303 \DeclareMathSymbol{\upsilontonosup}{\mathalpha}{libgreekup}{212}
304 \DeclareMathSymbol{\omegatonosup}{\mathalpha}{libgreekup}{184}
305 \DeclareMathSymbol{\upsilondieresistonosup}{\mathalpha}{libgreekup}{246}
306 \DeclareMathSymbol{\iotadieresisup}{\mathalpha}{libgreekup}{240}
307 \DeclareMathSymbol{\iotadieresistonosup}{\mathalpha}{libgreekup}{242}
308 \DeclareMathSymbol{\upsilondieresisup}{\mathalpha}{libgreekup}{244}
309 \DeclareMathSymbol{\alphatonosit}{\mathalpha}{libgreekit}{136}
310 \DeclareMathSymbol{\epsilontonosit}{\mathalpha}{libgreekit}{232}
311 \DeclareMathSymbol{\etatonosit}{\mathalpha}{libgreekit}{160}
312 \DeclareMathSymbol{\iotatonosit}{\mathalpha}{libgreekit}{208}
313 \DeclareMathSymbol{\omicrontonosit}{\mathalpha}{libgreekit}{236}
314 \DeclareMathSymbol{\upsilontonosit}{\mathalpha}{libgreekit}{212}
315 \DeclareMathSymbol{\omegatonosit}{\mathalpha}{libgreekit}{184}
316 \DeclareMathSymbol{\upsilondieresistonosit}{\mathalpha}{libgreekit}{246}
317 \DeclareMathSymbol{\iotadieresisit}{\mathalpha}{libgreekit}{240}
318 \DeclareMathSymbol{\iotadieresistonosit}{\mathalpha}{libgreekit}{242}
319 \DeclareMathSymbol{\upsilondieresisit}{\mathalpha}{libgreekit}{244}

```

Definition of the `\mathchar`’s without ‘up/it’ postfix. There are 27 uppercase and 40 lowercase letters, for a total of 67 glyphs. Actually, as stated above, I had done some work with LGR in September 2011 when I realized that bold italic had gone missing from the “8bit” font support of *libertine v5* compared to *libertine v4*, and was vaguely hoping that going via the LGR encoded font definition files would help, but it did not. I kept the file around, and basically only had to replace the font family name, or so I thought about ten days ago when I started re-examining the matter. 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 (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 **libgreek** to define control sequences for Greek letters with diacritics...

```
320 \def\libgreek@setgreekcs{%
321   \iflibgreek@upper@up
322     \let\Alpha\Alphaup
323     \let\Beta\Betaup
324     \let\Gamma\Gammaup
325     \let\Delta\Deltaup
326     \let\Epsilon\Epsilonup
327     \let\Zeta\Zetaup
328     \let\Eta\Etaup
329     \let\Theta\Thetaup
330     \let\Iota\Iotaup
331     \let\Kappa\Kappaup
332     \let\Lambda\Lambdaup
333     \let\Mu\Muup
334     \let\Nu\Nuup
335     \let\Xi\Xiup
336     \let\Omicron\Omicronup
337     \let\Pi\Piup
338     \let\Rho\Rhouup
339     \let\Sigma\Sigmaup
340     \let\Tau\Tauup
341     \let\Upsilon\Upsilonup
342     \let\Phi\Phiup
343     \let\Chi\Chiup
344     \let\Psi\Psiup
345     \let\Omega\Omegaup
346     \let\Sampi\Sampiup
347     \let\Digamma\Digammaup
348     \let\varSigma\varSigmaup
349   \else
350     \let\Alpha\Alphait
351     \let\Beta\Betait
352     \let\Gamma\Gammait
353     \let\Delta\Deltait
354     \let\Epsilon\Epsilonit
355     \let\Zeta\Zetait
356     \let\Eta\Etait
357     \let\Theta\Thetait
358     \let\Iota\Iotait
359     \let\Kappa\Kappait
360     \let\Lambda\Lambdait
361     \let\Mu\Muit
362     \let\Nu\Nuit
363     \let\Xi\Xiit
364     \let\Omicron\Omicronit
365     \let\Pi\Piit
366     \let\Rho\Rhoit
```

```

367     \let\Sigma\Sigmait
368     \let\Tau\Tauit
369     \let\Upsilon\Upsilonit
370     \let\Phi\Phiit
371     \let\Chi\Chiit
372     \let\Psi\Psiit
373     \let\Omega\Omegait
374     \let\Sampi\Sampiit
375     \let\Digamma\Digammit
376     \let\varSigma\varSigmit
377     \fi
378     \iflibgreek@lower@up
379         \let\alpha\alphaup
380         \let\beta\betaup
381         \let\gamma\gammaup
382         \let\delta\deltaup
383         \let\epsilon\epsilonup
384         \let\zeta\zetaup
385         \let\eta\etaup
386         \let\theta\thetaup
387         \let\iota\iotaup
388         \let\kappa\kappaup
389         \let\lambda\lambdaup
390         \let\mu\muup
391         \let\nu\nuup
392         \let\xi\xiup
393         \let\omicron\omicronup
394         \let\pi\piup
395         \let\rho\rhoup
396         \let\sigma\sigmaup
397         \let\tau\tauup
398         \let\upsilon\upsilonup
399         \let\phi\phiup
400         \let\chi\chiup
401         \let\psi\psiup
402         \let\omega\omegaup
403         \let\varsigma\varsigmaup
404         \let\varvarsigma\varvarsigmaup
405         \let\sampi\sampiup
406         \let\digamma\digammaup
407         \let\koppa\koppaup
408         \let\alphanos\alphanosup
409         \let\epsilontonos\epsilontonosup
410         \let\etatonos\etatonosup
411         \let\iotatonos\iotatonosup
412         \let\omicrontonos\omicrontonosup
413         \let\upsilontonos\upsilontonosup
414         \let\omegatonos\omegatonosup
415         \let\upsilonndieresistonos\upsilonndieresistonosup

```

```

416     \let\iotadieresis\iotadieresisup
417     \let\iotadieresisistonos\iotadieresisistonosup
418     \let\upsilondieresis\upsilondieresisup
419   \else
420     \let\alpha\alphait
421     \let\beta\betait
422     \let\gamma\gammait
423     \let\delta\deltait
424     \let\epsilon\epsilonit
425     \let\zeta\zetait
426     \let\eta\etait
427     \let\theta\thetait
428     \let\iota\iotait
429     \let\kappa\kappait
430     \let\lambda\lambdait
431     \let\mu\muait
432     \let\nu\nuait
433     \let\xi\xiit
434     \let\omicron\omicronit
435     \let\pi\piit
436     \let\rho\rhoit
437     \let\sigma\sigmait
438     \let\tau\tauait
439     \let\upsilon\upsilonit
440     \let\phi\phiit
441     \let\chi\chiit
442     \let\psi\psiit
443     \let\omega\omegait
444     \let\varsigma\varsigmait
445     \let\varvarsigma\varvarsigmait
446     \let\sampi\sampiit
447     \let\digamma\digammait
448     \let\koppa\koppait
449     \let\alphatonos\alphatonosit
450     \let\epsilontonos\epsilontonosit
451     \let\etatonos\etatonosit
452     \let\iotatonos\iotatonosit
453     \let\omicrontonos\omicrontonosit
454     \let\upsilontonos\upsilontonosit
455     \let\omegatonos\omegatonosit
456     \let\upsilondieresisistonos\upsilondieresisistonosit
457     \let\iotadieresis\iotadieresisit
458     \let\iotadieresisistonos\iotadieresisistonosit
459     \let\upsilondieresis\upsilondieresisit
460   \fi
461 }%
462 \libgreek@setgreekcs

```

And we have now reached the end of the **libgreek** package code. The actual .sty file will contain an `\endinput` added by the DocStrip extraction.

6 Implementation of the **libgreek-legacy** package

We use a rather short summary line because the package name is long and we are maniac and want the thing to occupy only one line in the log file...

```
1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{libgreek-legacy}
3 [2022/11/11 1.1 libertine-legacy math Greek (JFB)]
```

The font support from obsolete package [libertine-legacy](#) must be present. The legacy code used one of two possible font definition files for the LaTeX font system with U encoding, either associated with Libertine or Biolinum. We check if the Libertine-associated file exists, else abort with an Error.

```
4 \IfFileExists{ufxl03.fd}{%
5   }{%
6 \PackageError{libgreek-legacy}{%
7   Required font support file not present%
8   }{%
9   Is (now obsolete) libertine-legacy or earlier libertine v4 or v5\MessageBreak
10  package installed on your system?\MessageBreak
11  Maybe you meant to use libgreek not libgreek-legacy?\MessageBreak
12  Try again with libgreek, not libgreek-legacy. Will abort now, sorry}%
13  \endinput
14 }
```

Now back to the old 2011 code.

```
15 \def\libgreek@font{fxl03}
16 \def\libgreek@shape{it}
17 \def\libgreek@uppershape{n}
18 \newif\iflibgreek@twoshapes\libgreek@twoshapetrue
19 \edef\libgreek@series{\seriesdefault}
20 \edef\libgreek@boldseries{\bfdefault}
21 \newif\iflibgreek@sty
```

Original package used only [keyval](#) package and handled options via a `\ProcessOptionsWithKV` which had been copied-pasted from some forgotten source. Let's rather load [kvoptions](#). But we will not use its declarative interface, only `\define@key` directly.

No need for a `\fxb@scale` for Biolinum as file `ufxb03.fd` also uses `\fxl@scale`.

```
22 \RequirePackage{kvoptions}
23 \define@key{libgreek}{scale}[1]{\def\xl@scale{#1}}
```

The 1.0 code did not let the `libertine` key do the `\def` as has been added at 1.1, thus in situation with `biolinum,libertine` it did not revert action of `biolinum`.

By the way this [keyval](#) strange syntax means that `libertine=false` will have exact same effect `libertine=true`... it is only there to allow its usage without a `=value` part...

```
24 \define@key{libgreek}{libertine}[true]{\def\libgreek@font{fxl03}}
25 \define@key{libgreek}{biolinum}[true]{\def\libgreek@font{fxb03}}
26 \define@key{libgreek}{style}{\libgreek@stytrue\edef\libgreek@style{#1}}
27 \define@key{libgreek}{Greek}{\def\libgreek@Greeksshape{#1}}
28 \define@key{libgreek}{greek}{\edef\libgreek@shape{#1}
29   \edef\libgreek@uppershape{#1}}
30 \define@key{libgreek}{series}{\edef\libgreek@series{#1}}
```

```
31 \define@key{libgreek}{boldseries}{\edef\libgreek@boldseries{#1}}
```

Now we call the `kvoptions` interface. We did not change the family key from legacy code so we have to inform `kvoptions` about that. Then `\ProcessKeyvalOptions` replaces the 1.0 definition of some `\ProcessOptionsWithKV`. No need to set the prefix key as we don't use `kvoptions` declarative interface.

```
32 \SetupKeyvalOptions{family=libgreek}
33 \ProcessKeyvalOptions*
```

The 1.0 code uses one or two Symbol fonts, depending on whether the shape for uppercase is same or differs from the shape for lowercase. Indeed it assigns one symbol font to lowercase and one symbol font for uppercase.

```
34 \def\lbg@ISO{ISO}
35 \def\lbg@French{French}
36 \iflibgreek@sty
37   \ifx\libgreek@style\lbg@ISO
38     \def\libgreek@shape{it}
39     \libgreek@twoshapesfalse
40   \else
41     \ifx\libgreek@style\lbg@French
42       \def\libgreek@shape{n}
43       \libgreek@twoshapesfalse
44     \else
45       \def\libgreek@shape{it}
46       \def\libgreek@uppershape{n}
47   \fi\fi
48 \else
49 \ifx\libgreek@Greeksshape\undefined
50   \else\edef\libgreek@uppershape{\libgreek@Greeksshape}\fi
51 \ifx\libgreek@shape\libgreek@uppershape\libgreek@twoshapesfalse\fi
52 \fi
```

The `libgreekfont` is the one for lowercase (and perhaps uppercase).

```
53 \DeclareSymbolFont{libgreekfont}{U}{\libgreek@font}
54                                     {\libgreek@series}
55                                     {\libgreek@shape}
56 \SetSymbolFont{libgreekfont}{bold}{U}{\libgreek@font}
57                                     {\libgreek@boldseries}
58                                     {\libgreek@shape}
59 \def\libgreek@Greek{libgreekfont}
```

If a separate one is needed for uppercase it is `libGreefont`.

```
60 \iflibgreek@twoshapes
61   \DeclareSymbolFont{libGreefont}{U}{\libgreek@font}
62                                     {\libgreek@series}
63                                     {\libgreek@uppershape}
64   \SetSymbolFont{libGreefont}{bold}{U}{\libgreek@font}
65                                     {\libgreek@boldseries}
66                                     {\libgreek@uppershape}
67   \def\libgreek@Greek{libGreefont}
68 \fi
```


The macro `\libgreek@Greek` holds the symbol font name for uppercase.

Here the legacy code has all definitions done in the order of the codepoint; this was a bit perilous as lowercase may be intermixed with uppercase, and indeed on two occasions (`\varvarkappa` and `\varvarsigma`) the possibly wrong font was assigned.

Apart from that, it is not exactly clear why I decided to define control sequences for letters with diacritics; it was not even really documented, as the user was only invited to check a `libgreekcheck.tex` file (now not included anymore among extracted files from the `.dtx`).

Another thing to notice is that all those Greek letters are here defined to be of type `\mathord`.

```
69 \DeclareMathSymbol{\Alphatonos}{\mathord}{\libgreek@Greek}{134}
70 \DeclareMathSymbol{\anoteleia}{\mathord}{libgreekfont}{135}
71 \DeclareMathSymbol{\Epsilontonos}{\mathord}{\libgreek@Greek}{136}
72 \DeclareMathSymbol{\Etatonos}{\mathord}{\libgreek@Greek}{137}
73 \DeclareMathSymbol{\Iotatonos}{\mathord}{\libgreek@Greek}{138}
74 \DeclareMathSymbol{\Omicrontonos}{\mathord}{\libgreek@Greek}{140}
75 \DeclareMathSymbol{\Upsilontonos}{\mathord}{\libgreek@Greek}{142}
76 \DeclareMathSymbol{\Omegatonos}{\mathord}{\libgreek@Greek}{143}
77 \DeclareMathSymbol{\iotadieresis-tonos}{\mathord}{libgreekfont}{144}
78 \DeclareMathSymbol{\Alpha}{\mathord}{\libgreek@Greek}{145}
79 \DeclareMathSymbol{\Beta}{\mathord}{\libgreek@Greek}{146}
80 \DeclareMathSymbol{\Gamma}{\mathord}{\libgreek@Greek}{147}
81 \DeclareMathSymbol{\Delta}{\mathord}{\libgreek@Greek}{148}
82 \DeclareMathSymbol{\Epsilon}{\mathord}{\libgreek@Greek}{149}
83 \DeclareMathSymbol{\Zeta}{\mathord}{\libgreek@Greek}{150}
84 \DeclareMathSymbol{\Eta}{\mathord}{\libgreek@Greek}{151}
85 \DeclareMathSymbol{\Theta}{\mathord}{\libgreek@Greek}{152}
86 \DeclareMathSymbol{\Iota}{\mathord}{\libgreek@Greek}{153}
87 \DeclareMathSymbol{\Kappa}{\mathord}{\libgreek@Greek}{154}
88 \DeclareMathSymbol{\Lambda}{\mathord}{\libgreek@Greek}{155}
89 \DeclareMathSymbol{\Mu}{\mathord}{\libgreek@Greek}{156}
90 \DeclareMathSymbol{\Nu}{\mathord}{\libgreek@Greek}{157}
91 \DeclareMathSymbol{\Xi}{\mathord}{\libgreek@Greek}{158}
92 \DeclareMathSymbol{\Omicron}{\mathord}{\libgreek@Greek}{159}
93 \DeclareMathSymbol{\Pi}{\mathord}{\libgreek@Greek}{160}
94 \DeclareMathSymbol{\Rho}{\mathord}{\libgreek@Greek}{161}
95 \DeclareMathSymbol{\Sigma}{\mathord}{\libgreek@Greek}{163}
96 \DeclareMathSymbol{\Tau}{\mathord}{\libgreek@Greek}{164}
97 \DeclareMathSymbol{\Upsilon}{\mathord}{\libgreek@Greek}{165}
98 \DeclareMathSymbol{\Phi}{\mathord}{\libgreek@Greek}{166}
99 \DeclareMathSymbol{\Chi}{\mathord}{\libgreek@Greek}{167}
100 \DeclareMathSymbol{\Psi}{\mathord}{\libgreek@Greek}{168}
101 \DeclareMathSymbol{\Omega}{\mathord}{\libgreek@Greek}{169}
102 \DeclareMathSymbol{\Iotadieresis}{\mathord}{\libgreek@Greek}{170}
103 \DeclareMathSymbol{\Upsilondieresis}{\mathord}{\libgreek@Greek}{171}
104 \DeclareMathSymbol{\alphatonos}{\mathord}{libgreekfont}{172}
105 \DeclareMathSymbol{\epsilontonos}{\mathord}{libgreekfont}{173}
106 \DeclareMathSymbol{\etatonos}{\mathord}{libgreekfont}{174}
107 \DeclareMathSymbol{\iotatonos}{\mathord}{libgreekfont}{175}
108 \DeclareMathSymbol{\upsilondieresis-tonos}{\mathord}{libgreekfont}{176}
109 \DeclareMathSymbol{\alpha}{\mathord}{libgreekfont}{177}
```

```

110 \DeclareMathSymbol{\beta}{\mathord}{libgreekfont}{178}
111 \DeclareMathSymbol{\gamma}{\mathord}{libgreekfont}{179}
112 \DeclareMathSymbol{\delta}{\mathord}{libgreekfont}{180}
113 \DeclareMathSymbol{\epsilon}{\mathord}{libgreekfont}{181}
114 \DeclareMathSymbol{\zeta}{\mathord}{libgreekfont}{182}
115 \DeclareMathSymbol{\eta}{\mathord}{libgreekfont}{183}
116 \DeclareMathSymbol{\theta}{\mathord}{libgreekfont}{184}
117 \DeclareMathSymbol{\iota}{\mathord}{libgreekfont}{185}
118 \DeclareMathSymbol{\kappa}{\mathord}{libgreekfont}{186}
119 \DeclareMathSymbol{\lambda}{\mathord}{libgreekfont}{187}
120 \DeclareMathSymbol{\mu}{\mathord}{libgreekfont}{188}
121 \DeclareMathSymbol{\nu}{\mathord}{libgreekfont}{189}
122 \DeclareMathSymbol{\xi}{\mathord}{libgreekfont}{190}
123 \DeclareMathSymbol{\omicron}{\mathord}{libgreekfont}{191}
124 \DeclareMathSymbol{\pi}{\mathord}{libgreekfont}{192}
125 \DeclareMathSymbol{\rho}{\mathord}{libgreekfont}{193}
126 \DeclareMathSymbol{\varsigma}{\mathord}{libgreekfont}{194}
127 \DeclareMathSymbol{\sigma}{\mathord}{libgreekfont}{195}
128 \DeclareMathSymbol{\tau}{\mathord}{libgreekfont}{196}
129 \DeclareMathSymbol{\upsilon}{\mathord}{libgreekfont}{197}
130 \DeclareMathSymbol{\phi}{\mathord}{libgreekfont}{198}
131 \DeclareMathSymbol{\chi}{\mathord}{libgreekfont}{199}
132 \DeclareMathSymbol{\psi}{\mathord}{libgreekfont}{200}
133 \DeclareMathSymbol{\omega}{\mathord}{libgreekfont}{201}
134 \DeclareMathSymbol{\iotadieresis}{\mathord}{libgreekfont}{202}
135 \DeclareMathSymbol{\upsilondieresis}{\mathord}{libgreekfont}{203}
136 \DeclareMathSymbol{\omicrontonos}{\mathord}{libgreekfont}{204}
137 \DeclareMathSymbol{\upsilontonos}{\mathord}{libgreekfont}{205}
138 \DeclareMathSymbol{\omegatonos}{\mathord}{libgreekfont}{206}
139 \DeclareMathSymbol{\vartheta}{\mathord}{libgreekfont}{209}
140 \DeclareMathSymbol{\varUpsilon}{\mathord}{libgreek@Greek}{210}
141 \DeclareMathSymbol{\varUpsilontonos}{\mathord}{libgreek@Greek}{211}
142 \DeclareMathSymbol{\varUpsilondieresis}{\mathord}{libgreek@Greek}{212}
143 \DeclareMathSymbol{\varphi}{\mathord}{libgreekfont}{213}
144 \DeclareMathSymbol{\varpi}{\mathord}{libgreekfont}{214}

```

The 1.0 code used `\libgreek@Greek` here for the next two but this was a bug, so after some hesitation I fix it for the 1.1 release, eleven years later...

```

145 \DeclareMathSymbol{\varvarkappa}{\mathord}{libgreekfont}{215}
146 \DeclareMathSymbol{\varvarsigma}{\mathord}{libgreekfont}{219}
147 \DeclareMathSymbol{\Digamma}{\mathord}{libgreek@Greek}{220}
148 \DeclareMathSymbol{\digamma}{\mathord}{libgreekfont}{221}
149 \DeclareMathSymbol{\Koppa}{\mathord}{libgreek@Greek}{222}
150 \DeclareMathSymbol{\koppa}{\mathord}{libgreekfont}{223}
151 \DeclareMathSymbol{\Sampi}{\mathord}{libgreek@Greek}{224}
152 \DeclareMathSymbol{\sampi}{\mathord}{libgreekfont}{225}
153 \DeclareMathSymbol{\varkappa}{\mathord}{libgreekfont}{240}
154 \DeclareMathSymbol{\varrho}{\mathord}{libgreekfont}{241}
155 \DeclareMathSymbol{\varTheta}{\mathord}{libgreek@Greek}{244}
156 \DeclareMathSymbol{\varepsilon}{\mathord}{libgreekfont}{245}

```

```
157 \DeclareMathSymbol{\reversedvarepsilon}{\mathord}{libgreekfont}{246}
```

Those for some reason got listed here last, despite having lower codepoints.

```
158 \DeclareMathSymbol{\tonos}{\mathord}{libgreekfont}{132}
```

```
159 \DeclareMathSymbol{\dieresis-tonos}{\mathord}{libgreekfont}{133}
```

And we have now reached the end of the **libgreek**-legacy package code. The actual `.sty` file will contain an `\endinput` added by the DocStrip extraction.