

Church Slavonic Fonts

Aleksandr Andreev*

Nikita Simmons

June 22, 2023 version v2.3

(PDF file generated on July 30, 2024)

Contents

1	Introduction	2
1.1	License	3
1.2	Description	3
1.3	Installation and Usage	3
1.3.1	Font Formats	3
1.4	Source Packages	4
1.5	System Requirements	4
1.5.1	Microsoft Windows	4
1.5.2	GNU/Linux	4
1.5.3	OS X	5
1.6	Private Use Area	5
2	OpenType Technology	6
2.1	On Microsoft Windows	6
2.2	On GNU/Linux	6
2.3	OpenType Features	6
2.3.1	Combining Mark Positioning	6
2.3.2	Glyph Composition and Decomposition	7
2.3.3	Language-based Features	7
2.3.4	Stylistic Alternatives and Stylistic Sets	8
2.4	SIL Graphite Technology	9
2.5	Support of Advanced Features in LibreOffice	9
3	Fonts for Synodal Slavonic	10
3.1	Ponomar Unicode	10
3.1.1	Advanced Features of the Font	12

*Comments may be directed to aleksandr.andreev@gmail.com.

4	Fonts for Pre-Nikonian Printed Slavonic	14
4.1	Fedorovsk Unicode	14
4.1.1	Advanced Features of the Font	15
5	Fonts for Working with Ancient Manuscripts	15
5.1	Menaion Unicode	15
5.1.1	Advanced Features of the Font	17
6	Fonts for Academic Work	17
6.1	Monomakh Unicode	17
6.1.1	Advanced Features of the Font	21
6.2	Shafarik	22
7	Decorative Fonts	22
7.1	Indiction Unicode	22
7.2	Vertograd Unicode	23
7.3	Cathisma Unicode	23
7.4	Oglavie Unicode	24
7.5	Pomorsky Unicode	24
7.5.1	Advanced Features of the Font	24
8	Technical Fonts: FiraSlav	24
9	Known Issues	26
10	Credits	26

1 Introduction

Church Slavonic (also called Church Slavic, Old Church Slavonic or Old Slavonic; ISO 639-2 code cu) is a literary language used by the Slavic peoples; presently it is used as a liturgical language by the Russian Orthodox Church, other local Orthodox Churches, as well as various Byzantine-Rite Catholic and Old Ritualist communities. The package fonts-churchslavonic provides fonts for representing Church Slavonic text.

The fonts are designed to work with Unicode text encoded in UTF-8. Texts encoded in legacy codepages (such as HIP and UCS) may be converted to Unicode using a separate bundle of utilities. See the [Slavonic Computing Initiative website](#) for more information.

1.1 License

The fonts distributed in this package are licensed under the SIL Open Font License (version 1.1 or later).

As free software, these fonts are distributed in the hope that they will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the SIL Open Font License for more details.

This document is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit the [Creative Commons website](#).

1.2 Description

The package provides several fonts that are intended for working with Church Slavonic text of various recensions and other texts related to Church Slavonic: modern Church Slavonic text (“Synodal Slavonic”), historical printed Church Slavonic text and manuscript uncial (ustav) Church Slavonic text (in either Cyrillic or Glagolitic), as well as text in Sakha (Yakut), Aleut (Fox Island dialect), and Romanian (Moldovan) Cyrillic, all written in the ecclesiastical script. The coverage of the various fonts agrees with the guidelines for font coverage specified in [Unicode Technical Note #41: Church Slavonic Typography in Unicode](#). Generally speaking, it includes most (but not all) characters in the Cyrillic, Cyrillic Supplement, Cyrillic Extended-A, Cyrillic Extended-B, Cyrillic Extended-C (as of Unicode 9.0), Glagolitic, and Glagolitic Supplement blocks of Unicode. Characters not used in Church Slavonic, however, are not included (except for some characters used in modern Russian, Ukrainian, Belorussian, Serbian and Macedonian for purposes of compatibility with some applications).

1.3 Installation and Usage

If you are reading this document, then you probably have already downloaded the font package. You may check if you have the most recent version by visiting the [Slavonic Computing Initiative website](#).

1.3.1 Font Formats

All fonts are currently available in a single format:



[OpenType](#) fonts with PostScript outlines (also called OpenType-CFF fonts).



TrueType fonts have been deprecated, and are no longer provided. If you need TrueType fonts, see the [Legacy Fonts package](#).

You may need the legacy TrueType fonts in the following scenarios:

- OpenOffice.org and older versions of LibreOffice require use of SIL Graphite, which is only available in the TTF version. This limitation has been fixed as of LibreOffice 5.3, which now has [full OpenType support](#).
- OpenType-CFF fonts were poorly supported in Java prior to Oracle Java SE 7.
- In Microsoft products, OpenType glyph positioning is not supported for glyphs in the Private Use Area or characters outside of the Unicode 7.0 range. You should use LibreOffice if you need positioning of combining Glagolitic characters.

1.4 Source Packages

You can also download the FontForge sources for all of the fonts from the [GitHub repository](#). This is only useful if you are planning on editing the fonts in the [FontForge](#) font editor. In general, you will not gain any productivity improvements from rebuilding the font files, so rebuilding from source is not recommended, unless you have a real need to modify the fonts, for example, to add your own additional glyphs to the Private Use Area.

1.5 System Requirements

All of these fonts are large Unicode fonts and require a Unicode-aware operating system and software environment. Outside of a Unicode-aware environment, you will only be able, at most, to access the first 256 glyphs of a font.

1.5.1 Microsoft Windows

Unicode-encoded OpenType-CFF fonts are supported starting with Windows 2000. You will need a word processor that can handle Unicode-based documents, such as Microsoft Word 97 and above, or [LibreOffice](#). If using \TeX , you will need a Unicode-aware \TeX engine, such as \XeTeX or \LuaTeX .

You will also need a way to enter the Unicode characters that are not directly accessible from standard keyboards. We recommend installing a Church Slavonic or Russian-Extended keyboard layout, available from the [Slavonic Computing Initiative website](#). It is also possible to enter characters using the Windows Character Map utility or by codepoint, but this is not recommended.

1.5.2 GNU/Linux

In order to be able to handle OpenType fonts, your system should have the [freetype](#) library installed and enabled; this is normally done by default in all

modern distributions. You will need a Unicode-aware word processor, such as [LibreOffice](#). If using \TeX , you will need a Unicode-aware \TeX engine, such as \XeTeX or \LuaTeX .

You will need a keyboard driver to input Unicode characters. Under GNU/Linux, this is handled by the `m17n` library and database. See the [Slavonic Computing Initiative website](#) for more details.

1.5.3 OS X

Not sure.

1.6 Private Use Area

The Unicode Private Use Area (PUA) is a set of three ranges of codepoints (U+E000 to U+F8FF, Plane 15 and Plane 16) that are guaranteed to never be assigned to characters by the Unicode Consortium and can be used by third parties to allocate their own characters. The Slavonic Computing Initiative has established an industry standard for character allocation in the PUA, which is described in full in the [PUA Allocation Policy](#).

The PUA in these fonts contains various additional glyphs: contextual alternatives, stylistic alternatives, ligatures, hypothetical and nonce glyphs, various glyphs not yet encoded in Unicode, and various technical symbols. Most of these glyphs (the alternative glyphs and ligatures) are normally accessible via [OpenType](#) features. Thus, you generally do not need to access glyphs in the PUA directly. There may be some exceptions:

- If you need to access characters not yet encoded in Unicode and nonce glyphs.
- If you need to access alternative glyphs and ligatures on legacy systems that do not support OpenType or Graphite features.
- If you are a computer programmer and need to work with glyphs on a low level without relying on OpenType: having all alternatives mapped to the PUA allows for a simple way to access glyphs by codepoint instead of working with glyph indexes, which can change between versions of a font.

For the characters mapped in the PUA and other technical considerations, please see the [PUA Allocation Policy](#).

2 OpenType Technology

OpenType is a “smart font” technology for advanced typography developed by Microsoft Corporation and Adobe Systems and based on the TrueType font format. It allows for correct typography in complex scripts as well as providing advanced typographic effects. This is achieved by applying various *features*, or *tags*, described in the OpenType specification. Some of these features are supposed to be enabled by default, while others are considered optional, and may be turned on and off by the user when desired.

2.1 On Microsoft Windows

In order to use these advanced typographic features, in addition to a “smart” font (like the fonts in this package), you need an OpenType-aware application. Not all applications currently support OpenType, and not all applications that claim to support OpenType actually support all features or provide an interface to access features. Older versions of Microsoft’s Uniscribe library did not support OpenType features for Cyrillic and Glagolitic, but beginning with Windows 7, this has been resolved.

Generally speaking, you will get best results in X_YTeX or LuaTeX using the fontspec package or using advanced desktop publishing software such as Adobe InDesign. Most OpenType features are also accessible in Microsoft Office 2010 and later. LibreOffice also supports OpenType features starting with version 4.1, and support for turning off and on optional features was added in version 5.3. Please see the section [Support of Advanced Features in LibreOffice](#), below.

2.2 On GNU/Linux

OpenType support is provided by the HarfBuzz shaping library, which is accessible through FreeType, part of most standard distributions of the X Window System. Thus, OpenType will be available in any application that uses FreeType, though some applications lack an interface to turn on and off optional features. Generally speaking, you will get best results in X_YTeX or LuaTeX using the fontspec package. LibreOffice also supports OpenType features starting with version 4.1, and support for turning off and on optional features was added in version 5.3. Please see the section [Support of Advanced Features in LibreOffice](#), below.

2.3 OpenType Features

2.3.1 Combining Mark Positioning

OpenType allows smart diacritic positioning: if you type a letter followed by a diacritic, the diacritic will be placed exactly above or below the letter; this is

provided by the mark feature. In addition, the *mkmk* feature is used to position two marks with respect to each other, so that an additional diacritic can be stacked properly above the first. This behavior is demonstrated below:

$$\begin{array}{ll} \text{А} + \text{◌} \rightarrow \text{А́} \\ \text{Ѧ} + \text{◌} \rightarrow \text{Ѧ́} & \text{(glyph positioning via } \textit{mark} \text{ feature)} \\ \text{◌} + \text{◌} \rightarrow \text{◌} & \text{(glyph positioning via } \textit{mkmk} \text{ feature)} \end{array}$$

The fonts provide proper mark and *mkmk* anchor points for all Cyrillic and Glagolitic letters and combining marks, allowing you to enter them in almost any combination (even those that are implausible). Most OpenType renderers (except older versions of Adobe’s Cooltype library) support these features, so you should be able to achieve correct positioning in most OpenType-aware applications (for example, in MS Word 2010 or newer, LibreOffice 4.1 or newer, and Xe_{La}TeX).

2.3.2 Glyph Composition and Decomposition

The Glyph Composition / Decomposition (*ccmp*) feature is used to compose two characters into a single glyph for better glyph processing. This feature is also used to create precomposed forms of a base glyph with diacritical marks when use of only mark and *mkmk* cannot achieve the necessary positioning. It is also used to create alternative glyph shapes, such as the alternative version of the Psili used over capital letters and the truncated forms of the letter Uk used with accent marks, as is demonstrated in the examples below:

$$\begin{array}{ll} \text{◌} \rightarrow \text{◌} & \text{(glyph substitution using } \textit{ccmp} \text{ feature)} \\ \text{◌} + \text{◌} \rightarrow \text{◌} & \text{(ligature substitution using } \textit{ccmp} \text{ feature)} \\ \text{Ѧ} + \text{◌} + \text{◌} \rightarrow \text{Ѧ} & \text{(ligature substitution using } \textit{ccmp} \text{ feature)} \\ \text{Ѧ} + \text{◌} \rightarrow \text{Ѧ} & \text{(contextual substitution using } \textit{ccmp} \text{ feature)} \end{array}$$

Generally speaking, the *ccmp* feature is not supposed to (and often just cannot) be turned off, and thus this functionality should work properly in any OpenType-aware application. For more details on ligatures, see [Unicode Technical Note #41: Church Slavonic Typography in Unicode](#).

2.3.3 Language-based Features

Language-based features such as the *locl* (localized forms) feature provide access to language-specific alternate glyph forms, such as the alternate forms of the Cyrillic Letter I used in Ukrainian and Belorussian:

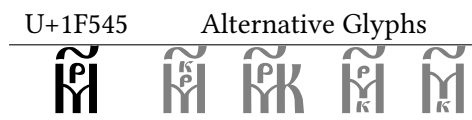
$$\begin{array}{ll} \text{І} & \text{(Church Slavonic text)} \\ \text{і} & \text{(Ukrainian text)} \end{array}$$

To make use of these features, you need an OpenType-aware application that supports specifying the language of text, for example X_YTeX or LuaTeX using the `fontspec` or `polyglossia` packages. Since many software applications do not allow you to specify Church Slavonic as a language of text, it is assumed by default that the font is being used to represent Church Slavonic text, and thus all glyphs have Church Slavonic appearances unless another language is specified.

LibreOffice allows you to specify that text is in Church Slavonic starting with version 5.0. This will allow you to take advantage of other features, such as Church Slavonic hyphenation (see the [Slavonic Computing Initiative website](#) for more information). Microsoft Corporation does not recognize Church Slavonic as a valid language, so you will not be able to set the language of text to Church Slavonic in any Microsoft product¹.

2.3.4 Stylistic Alternatives and Stylistic Sets

Stylistic Alternatives (`salt` feature) provide variant glyph shapes that may be selected by the user at will. Typically, these are glyphs that differ from the base glyph only in graphical appearance where the use of these glyphs does not follow any language-based or typography-based rules, but rather is just an embellishment. For example, the following variant forms of U+1F545 Symbol for Marks Chapter are provided:



Stylistic sets are used to enable a group of stylistic variant glyphs, designed to harmonize visually, and make them automatically substituted instead of the default forms. OpenType allows to specify up to 20 stylistic sets, marking them features `ss01`, `ss02`, ... `ss20`.

Use of Stylistic Alternatives and Stylistic Sets requires an OpenType-aware application that provides an interface to turn off and on advanced features (since by default these features are turned off). This is possible in X_YTeX or LuaTeX using the `fontspec` package and in LibreOffice (starting with version 5.3) by use of a special syntax that appends the needed option to the font name. See the section [Support of Advanced Features in LibreOffice](#), below. In Microsoft Office 2010 and later, Stylistic Sets may be turned off and on under OpenType features on the Advanced tab of the Font dialog. However, Microsoft Office does not allow you to select multiple Stylistic Sets simultaneously or to access the `salt` feature. If necessary, you may access alternate glyphs by codepoint from the

¹Please do not contact the font maintainers about this issue. Instead, complain to Microsoft Customer Service in the USA at 1-800-642-7676 or in Canada at +1 (877) 568-2495.

Private Use Area (PUA). However, relying on the PUA as a data exchange mechanism is discouraged.

2.4 SIL Graphite Technology



As of version 1.3 of this package, support for [SIL Graphite](#) features has been discontinued. If you need Graphite support, see the [Legacy Fonts package](#).

2.5 Support of Advanced Features in LibreOffice

Support for OpenType features is available in LibreOffice and all OpenOffice.org derivatives starting with version 3.2 of OpenOffice.org. While correct positioning, attachment and substitutions will work automatically, earlier versions of LibreOffice had no mechanism to turn off and on optional features. Support for turning OpenType features off and on is available starting with LibreOffice version 4.1. As of LibreOffice 6.2, OpenType features may be turned off and on via the Font Features window, which is accessible from the Character window. From the Format menu choose Character, then in the Character window, press the Features button. The Font Features window will display all of the features available in the given font, and you can use the checkboxes to turn specific features off and on.

This functionality will be useful for LibreOffice users relying on automatic hyphenation. Since LibreOffice has [no mechanism to set the hyphenation character](#), the Ponomar Unicode and Monomakh Unicode fonts provide the underscore as a hyphenation character via Stylistic Set 1 in OpenType.

Prior to version 6.2, OpenType features had to be turned off and on using a special extended font name. This method is still available if you don't wish to use the graphical interface: in order to activate an optional feature, its ID, followed by an equals sign and the ID of the desired setting, are appended directly to the font name string. An ampersand is used to separate different feature/settings pairs.

For example, the following “font” should be used in order to enable the ss01 (Stylistic Set 1) feature:

```
Ponomar Unicode:ss01=1
```

The same syntax is used to turn off and on optional Stylistic Alternatives (salt), where 1 indicates the first alternate glyph, 2 – the second alternate glyph, and so forth.

3 Fonts for Synodal Slavonic

3.1 Ponomar Unicode

Ponomar Unicode is a font that reproduces the typeface of Synodal Church Slavonic editions from the beginning of the 20th Century. It is intended for working with modern Church Slavonic texts (Synodal Slavonic). Ponomar Unicode is based on the Hirmos UCS font designed by Vlad Dorosh, but has been modified by the authors of this package. Examples of text set in Ponomar Unicode are presented below.

Бл҃жєнѣ мѡжѣ, ѣже не ѣде на совѣтѣ нечестѣвыхѣ, ѣ
на пѡтѣ грѣшныхѣ не стѣ, ѣ на сѣдѣлицѣ гѡбѣтелей не
сѣде: но въ законѣ г҃ни вола ѣгѡ, ѣ въ законѣ ѣгѡ
пѡчѣнѣ дѣнь ѣ нѡцѣ. Ѣ вѡдетѣ ѣкѡ дрѣво насаждѣное
при ѣсхѡдѣнѣхѣ вѡдѣ, ѣже плѡдѣ сѡѡн дѣстѣ во вѣмѣ
сѡѡ, ѣ лѣстѣ ѣгѡ не ѡпадѣтѣ: ѣ всѣ, ѣлѣка ѣще тѡ-
рѣтѣ, ѡспѣетѣ. Не тѣкѡ нечестѣвѣн, не тѣкѡ: но ѣкѡ
прѣхѣ, ѣгѡже возметѣетѣ вѣтрѣ ѡ лицѣ землѣ. Гегѡ
рѣдѣ не воскресѣтѣ нечестѣвѣн на сѡдѣ, ниже грѣшницѣ
въ совѣтѣ прѣвѣныхѣ. Ѣкѡ вѣстѣ г҃ѣ пѡтѣ прѣвѣныхѣ, ѣ
пѡтѣ нечестѣвыхѣ погѣбнетѣ.

Kievan Church Slavonic uses a number of variant glyph forms, such as U+1C81 Long-Legged De (ѣ) and U+A641 Variant Ze (ѣ):

Бл҃жєнѣ мѡжѣ, ѣже не ѣде на совѣтѣ нечестѣвыхѣ, ѣ
на пѡтѣ грѣшныхѣ не стѣ, ѣ на сѣдѣлицѣ гѡбѣтелей не
сѣде: но въ законѣ г҃ни вола ѣгѡ, ѣ въ законѣ ѣгѡ
пѡчѣнѣ дѣнь ѣ нѡцѣ. Ѣ вѡдетѣ ѣкѡ дрѣво насаждѣное
при ѣсхѡдѣнѣхѣ вѡдѣ, ѣже плѡдѣ сѡѡн дѣстѣ во вѣмѣ
сѡѡ, ѣ лѣстѣ ѣгѡ не ѡпадѣтѣ: ѣ всѣ, ѣлѣка ѣще тѡ-
рѣтѣ, ѡспѣетѣ. Не тѣкѡ нечестѣвѣн, не тѣкѡ: но ѣкѡ
прѣхѣ, ѣгѡже возметѣетѣ вѣтрѣ ѡ лицѣ землѣ. Гегѡ
рѣдѣ не воскресѣтѣ нечестѣвѣн на сѡдѣ, ниже грѣшницѣ
въ совѣтѣ прѣвѣныхѣ. Ѣкѡ вѣстѣ г҃ѣ пѡтѣ прѣвѣныхѣ, ѣ

пѸтъ нечестѣвыхъ погнѣнетъ.

Other Languages The Ponomar Unicode font may also be used to typeset liturgical texts in other languages that use the ecclesiastic Cyrillic alphabet. Three such examples are fully supported by the font: Romanian (Moldovan) in its Cyrillic alphabet, Aleut (Fox Island or Eastern dialect) in its Cyrillic alphabet, and Yakut (Sakha) as written in the alphabet created by Bishop Dionysius (Khitrov).

Here is an example of the Lord's Prayer in Romanian (Moldovan) Cyrillic:

ТѦТѦЛЬ НѦСТРѸ КѦРЕЛЕ ѼЩИ ꙗ ЧѦрюри: СФНЦѢСКХСЕ НѸ-
МЕЛЕ ТѦѸ: КІЕ ꙗПХХЦІѦ ТѦ: ФІЕ ВѦѦ ТѦ, ПРЕКѸМЪ ꙗ ЧѦрю
ШИ ПРЕ ПХМЖНТЪ. ПЖННѢ НѦСТРХ ЧѢ ДЕПѸРѸРѢ ДЖНЕ НѦѦѦ
ѦСТХЖИ. ШИ НЕ ѦРТХ НѦѦѦ ГРЕШѦЛЕЛЕ НѦСТРЕ, ПРЕКѸМЪ ШИ
НОѢ ѼРТѦМЪ ГРЕШНЦНѦѦРЪ НѦЦРН. ШИ НѸНЕ ДЖЧЕ ПРЕ НѦѢ
ꙗ НСПНТХ. ЧИ НЕ НЖЕХѢЦЕ ДЕ ЧѦЛЬ РѸѸ.

And here is an example of the Lord's Prayer in Aleut Cyrillic:

ТѸманнѢ Ѧдакх! Ѧманх акѸхтхннх Ѣннх кѸгннх,
ѦсѢнх амчѸгѢсѢдѢгта, Ѧгалннх ѢкѢгта, ѦнѸтанѢтхннх
малгѢгтанх Ѣннмх кѸганх кѢюхх тѢнамх кѸганх. КѢл-
гѢдамх анѸтанѢ нѢнх акачѢ ѸѢѢмх: кѢюхх тѸманннх
ѢѸѸнх нѢнх нгнѢда, ѢмѢкѸнх тѸманх кѢюхх малгалнгннх
нѢнх ѢѸгнннх нгнѢдѢкѸнх: кѢюхх тѸманх сѸглѢтѢхнѢ-
нахтхннх, тѢгѢ ѢѢлѢѢдамх нѢѢнх тѸманх ѢгнѢчѢ.

And here is an example of the Lord's Prayer in Yakut (Sakha):

Хаалланнырх юрдуоһуһуһуһу бэрх аҕабытх бисеһа! СѢѢ-
теһдѢннннх Ѣтхннх эһеһа; кѢллннх царствѢннх эһеһа; снрх
юрдуоһуһуһу кѢнгуоһуһу эһеһа, хааллнх юрдуоһуһу кѸрдѸкх бол-
лѸнх; бюгуоһуһу кѸннѢгннх асырх аспытхннх бисеһннх кѸлѸ
бисеһа бюгуоһуһу; бисеһннх да естѢрбѢнтннх хаалларх бисеһа, хѢй-
тахх бисеһннх да хаалларабытх бѢэбѢнтх естѢхтѢрбѢнтннхуһуһу;

КНЛАРМАА ДА БИСИГНИН АЛБАНХАЙГА ; ХАТА БЫСА БИСИГНИН АЛБЫНТАНХ .

3.1.1 Advanced Features of the Font

Ponomar Unicode places some characters in the Private Use Area (PUA). For the general PUA mappings, please see the [PUA Allocation Policy](#).

In addition to the general PUA mappings, some characters have been allocated the open range section of the PUA. These are:

- U+F400 – Alternatives of SMP glyphs: This section contains copies in the BMP of SMP glyphs for support in legacy applications. Currently, the following are available: U+F400 - U+F405 – Typicon symbols (copies of U+1F540 through U+1F545).
- U+F410 – Presentation forms: Contains various presentation forms and ligatures used internally by the font. Generally, these are not intended to be called by users or external applications.
- U+F420 – Linguistic alternatives: Contains alternative shapes of glyphs that are language specific. Right now, these are modern punctuation shapes for use with Latin characters. These are not intended to be called externally.
- U+F441 and on – stylistic alternatives of Latin characters (Blackletter forms). These can be called via the Stylistic Set 2, but, if necessary, they may be called from the PUA directly. They are mapped to in the same order as in the Basic Latin block, beginning with U+F441 (for U+0041 Latin Capital Letter A). In addition to the Basic Latin repertoire, we also have: U+F4DE – Blackletter Thorn; U+F4FE – Lowercase Blackletter Thorn; and U+F575 – Blackletter Long S

The font provides a number of ligatures, which are made by inserting the Zero Width Joiner (U+200D) between two characters. A list of ligatures is provided in Table 1.

Table 1: Ligatures available in Ponomar Unicode

Name	Sequence	Appearance
Ligature A-U	U+0430 U+200D U+0443	АҮ
Ligature El-U	U+043B U+200D U+0443	ЭҮ
Ligature Te-Ve	U+0442 U+200D U+0432	ТѢ

In OpenType, a number of Stylistic Alternatives are defined. They are listed in Table 2. In addition to additional decorative glyphs for the Symbol for Mark’s Chapter, the feature provides the alternate forms of the letter U+0423 Ъ that look exactly like U+A64A Ъk (this usage is found in some publications), and an alternative form for the U+0404 Wide Ye for use in contexts where it needs to be distinguished from U+0415 Ye (mostly for Ukrainian text stylized in a Church Slavonic font).

Table 2: Stylistic Alternatives in Ponomar Unicode

	Base Form	Alternate Forms
U+1F545	Ѣ	Ѣ̂ Ѣ̃ Ѣ̄ Ѣ̅ Ѣ̆
		Ѣ̇ Ѣ̈ Ѣ̉ Ѣ̊
U+0423	Ѣ	Ѣ̂
U+040E	Ѣ̂	Ѣ̃
U+0404	Ѣ̄	Ѣ̅

For the Cyrillic letters, the stylistic alternatives feature also allows access to truncated letter forms; the order of the alternate forms is always: lower truncation, upper truncation, left truncation, right truncation. Table 3 demonstrates which truncated forms are available. Generally speaking, truncation should be handled automatically by desktop publishing software and T_EX, though this is difficult to accomplish.

Table 3: Truncated Forms Accessible via Stylistic Alternatives Feature in Ponomar Unicode

	Base Form	Truncated Forms
U+0440	Ѣ̂	Ѣ̂
U+0443	Ѣ̃	Ѣ̃
U+0444	Ѣ̄	Ѣ̄ Ѣ̅
U+0445	Ѣ̅	Ѣ̅ Ѣ̆ Ѣ̇
U+0446	Ѣ̆	Ѣ̆
U+0449	Ѣ̇	Ѣ̇
U+0471	Ѣ̈	Ѣ̈ Ѣ̉
U+A641	Ѣ̉	Ѣ̉
U+A64B	Ѣ̊	Ѣ̊ Ѣ̋ Ѣ̌

Stylistic Set 1 (ss01) is provided as a temporary workaround to [LibreOffice Bug 85731](#), which does not allow you to specify the hyphenation character in LibreOffice. When turned on, it replaces all instances of U+002D Hyphen-Minus and U+2010 Hyphen with U+005F Low Line (underscore) for use as a hyphenation character. Please note that this feature will be deprecated once the neces-

sary functionality is added to LibreOffice.

There is also defined Stylistic Set 2 (“ss02”), Blackletter forms. When this stylistic set is turned on, Latin letters appear in blackletter as opposed to their modern forms. This is useful for setting Latin text side-by-side with Slavonic in some contexts. See the following example:

Regular	The quick brown fox. 1234567890. Ѧ ѿѧ по словѣнскѧ.
Blackletter	The quick brown fox. 1234567890. Ѧ ѿѧ по словѣнскѧ.

Note that as of version 2.0 of the font, the ASCII digits (commonly called “Arabic numerals”) are provided in roman form. Use Stylistic Set 2 to access the blackletter forms, if necessary.

4 Fonts for Pre-Nikonian Printed Slavonic

4.1 Fedorovsk Unicode

Fedorovsk Unicode is based on the Fedorovsk font designed by Nikita Simmons. It has been re-encoded for Unicode, with added OpenType features. The Fedorovsk typeface is supposed to reproduce the typeface of the printed editions of Ivan Fedorov produced in Moscow, for example, the Apostol of 1564. The font is intended primarily for typesetting pre-Nikonian (Old Rite) liturgical texts or for working with such texts in an academic context.

Here is an example from the 1564 Apostol of Ivan Fedorov:

Первое ѡубо слово сотворѣхъ ѡ всѣхъ , ѿ , аѡфиле ,
ѡ нѣже начѣхъ ѿ , творѣхъ нѣ ѡуби . до негоже дѣе , за
повѣдавъ апломъ дѣхъ стѣи , нѣже нѣзбрѣ вознесѣа . прѣнѣмнѣ
нѣ постѣвнѣе жѣва по страданѣи свѣмъ . во мнозѣхъ нѣстинныхъ
знаменѣихъ . дѣмн четѣридесятѣмн ѡблѣаа нѣмъ нѣглѣ ѡже
ѡцѣвѣи бѣи . снѣмнѣ нѣады , повелѣваа нѣмъ ѡѣросалѣма
не ѡадыаа . нѣ ждѣти ѡбѣтовѣа нѣе ѡѣе , ѣже слышастѣ ѡменѣ .
ѡако ѡаааа ѡубо крѣхъ ѣтъ водою . выже нѣматѣ крестѣхъ
дѣхъ стѣмъ , не помнозѣхъ сѣхъ дѣе .

Here is an example from the Flowery Triodion of 1648:

ѡры пѣаа . глѣа , ѣ . аа во крѣхъ бѣа , * нѣ разѣдѣа
вразѣ ѣа . Пѣаа сѣаааа нѣмъ дѣеа показѣа , пѣаа нѣва
стѣа , пѣаа таѣнѣаааа , пѣаа всѣаааа , пѣаа хрѣа нѣ
бѣаааа , пѣаа непорѣааа , пѣаа вѣаааа , пѣаа вѣаааа ,

πάχα δέβρη ρήνικτλ ήλμζ ѿβερζάνцлл , πάχα βεΐχζ ѿεϊλάцлл
βεΐρηγζ .

4.1.1 Advanced Features of the Font

The font provides a number of ligatures, which are made by inserting the Zero Width Joiner (U+200D) between two characters. A list of ligatures is provided in Table 4.

Table 4: Ligatures available in Fedorovsk Unicode

Name	Sequence	Appearance
Ligature A-U	U+0430 U+200D U+0443	Ѡ
Ligature El-U	U+043B U+200D U+0443	ѡ
Ligature A-Izhitsa	U+0430 U+200D U+0475	Ѣ
Ligature El-Izhitsa	U+043B U+200D U+0475	ѣ
Ligature Te-Ve	U+0442 U+200D U+0432	Ѥ
Ligature Er-Yat	U+0440 U+200D U+0463	Ѧ

In OpenType, a number of Stylistic Alternatives are defined. They are listed in Table 5. In addition to providing alternative glyph shapes for U+1F545 Symbol for Mark’s Chapter, they allow you to control the positioning of diacritical marks over certain letters.

Additionally, three stylistic sets have been defined in the font. Stylistic Set 1 (“Right-side accents”) positions the accents over the Yat and the Uk on the right side and Stylistic Set 2 (“Left-side accents”) positions the accents over the Yat and the Uk on the left side. These stylistic sets are useful when a text uses one of these positionings throughout. Stylistic Set 10 (“Equal Baseline Variants”) sets the capital letters on the same baseline as the lowercase letters (useful for working with the font in an academic context where the traditionally lowered baseline of uppercase letters can cause vertical spacing issues when working with text that is both in Latin and Cyrillic scripts). Here is an example:

Χρίστός βοικρε ήζ' μέργβγχζ (regular text)
Χρίστός βοικρε ήζ' μέργβγχζ (Stylistic Set 10 enabled)

5 Fonts for Working with Ancient Manuscripts

5.1 Menaion Unicode

The Menaion typeface is supposed to be used for working with text of Ustav-era manuscripts. It contains the full repertoire of necessary Cyrillic and Glagolitic

Table 5: Stylistic Alternatives in Fedorovsk Unicode

	Base Form	Alternate Forms					
U+0404	ѐ	Ა					
U+0426	ђ	Გ					
U+0491	ѣ	Უ					
U+A64C	Ѹ	Ი					
U+047C	Ѽ	Ნ					
U+047E	Ѿ	Პ					
U+047F	ѿ	Ჟ					
U+1F545	Პ	Ჟ	Რ	Ს	Ტ	Უ	Ფ Ქ
U+0463 U+0486	Ѣ	Ტ					
U+0463 U+0300	Ѣ̃	Ტ̃	Ტ̃				
U+0463 U+0301	Ѣ̇	Ტ̇	Ტ̇				
U+0463 U+0311	Ѣ̊	Ტ̊	Ტ̊				
U+0463 U+0486 U+0301	Ѣ̇̂	Ტ̇̂					
U+A64B U+0486	Ѣ̂	Ტ̂					
U+A64B U+0300	Ѣ̃̂	Ტ̃̂	Ტ̃̂				
U+A64B U+0301	Ѣ̇̂	Ტ̇̂	Ტ̇̂				
U+A64B U+0311	Ѣ̊̂	Ტ̊̂	Ტ̊̂	Ტ̊̂			
U+A64B U+0486 U+0301	Ѣ̇̂̂	Ტ̇̂̂					

glyphs as well as glyphs of Byzantine Ecphonetic notation of the kind used in Cyrillic or Glagolitic manuscripts.

The Menaion font was originally designed by Victor A. Baranov at [the Manuscript Project](#). It was re-encoded for Unicode by Aleksandr Andreev with permission of the original author.

Samples of text in Menaion Unicode are presented in Figures 1 and 2. Please note that combining Glagolitic letters (Glagolitic Supplement) became available in Unicode 9.0. In older versions of Microsoft software, correct glyph positioning for these characters using OpenType features may not be possible. To achieve the desired output, we recommend you use LibreOffice, X_YTeX, LuaTeX, or advanced desktop publishing software such as Adobe InDesign.

5.1.1 Advanced Features of the Font

The font provides a number of ligatures, which are made by inserting the Zero Width Joiner (U+200D) between two characters. The list of ligatures is provided in Table 6.

6 Fonts for Academic Work

6.1 Monomakh Unicode

Monomakh Unicode is based on the Monomachus font designed by Alexey Kryukov. It has been modified with permission. Monomakh Unicode is a Cyrillic font implemented in a mixed ustav/poluustav style and intended to cover needs of researches dealing with Slavic history and philology. It includes all historical Cyrillic characters currently defined in Unicode, as well as a set of Latin letters designed to be stylistically compatible with the Cyrillic part. This may be useful for typesetting bilingual editions in Church Slavonic and languages written in the Latin script, especially those that use many diacritical marks, as in the Romanian example below.

Бѣ́же́нъ мѡ́жъ, ѣ́же не ѣ́де на
совѣ́тъ нечесті́выхъ, ѡ́ на пѣ́ти
грѣ́шныхъ не ста́, ѡ́ на сѣ́далищи
гѣ́бі́телей не сѣ́де: но въ за́конѣ́
гѣ́ни во́ла ѣ́го, ѡ́ въ за́конѣ́ ѣ́го
по́вѣ́стѣ́ де́нь ѡ́ но́щѣ́. Ѣ́детъ
ѣ́къ дре́во наса́жде́ное при ѡ́схѡ́ди-
щи́хъ во́дѣ́, ѣ́же пло́дъ сво́й да́стъ
во вре́мѣ́ своѣ́.

Fericit bărbatul, care n-a um-
blat în sfatul necredincioșilor
și în calea păcătoșilor nu a stat
și pe scaunul hulitorilor n-a șe-
zut; ci în legea Domnului e voia
lui și la legea Lui va cugeta ziua
și noaptea. și va fi ca un pom ră-
sădit lângă izvoarele apelor, ca-
re rodul său va da la vremea sa.

Figure 1: Cyrillic text from the Ostromir Gospel (11th century)

1 НѢСОНН БѢ СЛОВО
 2 Н СЛОВО БѢ ОТЪ
 3 БѢ Н БѢ БѢ
 4 СЛОВО ꙗко БѢ
 5 НѢСОНН ОУ
 6 БѢ ꙗко Н ТѢМЪ ЪСА БЫ
 7 ША ꙗко Н БѢЗ НЕГО НН
 8 УБТОЖЕ НЕ БЫСТЬ .
 9 КЖЕ БЫСТЬ ꙗко ВЪ ТО
 10 МЪ ЖИВОТЪ БѢ . Н
 1 ЖИВОТЪ БѢ СВѢТЪ
 2 УЛОВѢКОМЪ ꙗко Н СВѢ
 3 ТЪ ВЪ ТѢМѢ СВѢТН
 4 ТЪСА . Н ТѢМА КЖЕ
 5 НЕ ОБАТЪ ꙗко БЫСТЬ
 6 УЛѢКЪ ПОСЛААНЪ
 7 ОТЪ БѢ . НМА КМОУ
 8 НОАНЪ ꙗко ТЪ ПРИДЕ
 9 ВЪ СВѢДѢТЕЛЬ
 10 СТВО . ДА СВѢДѢТЕ
 2.2 1 ЛЪСТВОУЮТЪ О СВѢ
 2 ТѢ ꙗко ДА ВЪСН ВѢРЖ
 3 НМЖТЪ НМЪ ꙗко НЕ БѢ
 4 ТЪ СВѢТЪ ꙗко НЪ ДА
 5 СВѢДѢТЕЛЬСТВОУ
 6 ЮТЪ О СВѢТѢ ꙗко БѢ
 7 СВѢТЪ ИСТИННЫИ
 8 Н . НЖЕ ПРОСВѢЩАН
 9 ТЪ ЪСАКОГО УЛКА ,
 10 ГРАДЖЩА ВЪ МНРЪ ꙗко

Figure 2: Glagolitic text from *Codex Assemanius* (11th century)


1 ✠ Ǫꙋⱁ ⱁⱏ ⱄꙋǫ
 2 ⱄⱏⱏⱏⱏⱏⱏ ⱏⱏ
 3 ⱄⱏⱏⱏⱏⱏ .
 4 ⱄ ⱄⱏⱏⱏⱏⱏ
 5 ⱏⱏ ⱏⱏ ⱏⱏ .
 6 ⱄ ⱏⱏⱏ ⱏⱏ
 7 ⱄⱏⱏⱏⱏⱏ .
 8 ⱄⱏ ⱏⱏ ⱄⱏⱏⱏ
 9 ⱏⱄ . ⱏⱏ ⱏⱏⱏ . ⱏⱏ
 10 ⱄⱏ ⱏⱏⱏⱏⱏ ⱏⱏⱄ
 11 ⱏⱏ . ⱄ ⱏⱏⱏ ⱏⱏⱏⱏ
 12 ⱏⱏⱏⱏⱏⱏⱏⱏ
 13 ⱏⱏ ⱏⱏⱄⱄⱏⱏⱏ . ⱏ
 14 ⱏⱏ ⱏⱏⱄⱏⱏⱏ .
 15 ⱏⱏ ⱏⱏⱏⱏⱏ ⱏⱏⱏⱏ
 16 ⱏⱏⱏ ⱏⱏ . ⱄ ⱏⱏⱏⱏ
 17 ⱏⱏⱏ ⱏⱏ ⱄⱏⱏⱏⱏ
 18 ⱏⱏⱏⱏⱏⱏ . ⱄ ⱄⱏⱏⱏ
 19 ⱏⱏ ⱏⱏⱏⱏⱏ ⱄⱏⱏ
 20 ⱏⱏⱏⱏⱏ ⱄⱏ . ⱄ ⱏⱏⱏ
 21 ⱏⱏⱏ ⱏⱏⱏ ⱏⱏ ⱏⱏⱏⱏ

Table 6: Ligatures available in the Menaion Unicode font


Name	Sequence	Appearance
Small Ligature I-Ye	U+0438 U+200D U+0435	ИѢ
Small Ligature En-I	U+043d U+200D U+0438	НН
Small Ligature En-Small Yus	U+043d U+200D U+0467	НА
Small Ligature Es-Ve	U+0441 U+200D U+0432	Ѧ
Small Ligature Te-Er	U+0442 U+200D U+0440	Ѧ
Capital Litagure A-U	U+0410 U+200D U+0423	АѦ
Small Ligature A-U	U+0430 U+200D U+0443	аѦ
Small Ligature A-Te	U+0430 U+200D U+0442	Ѧ
Capital Ligature I-Ye	U+0418 U+200D U+0415	ИѢ
Capital Ligature El-Ge	U+041b U+200D U+0413	Ѧ
Small Ligature El-Ge	U+043b U+200D U+0433	Ѧ
Capital Ligature En-I	U+041d U+200D U+0418	НН
Capital Ligature En-Small Yus	U+041d U+200D U+0466	НА
Capital Ligature Es-Ve	U+0421 U+200D U+0412	Ѧ
Small Ligature Te-Yat	U+0442 U+200D U+0463	Ѧ
Capital Ligature Te-Ve	U+0422 U+200D U+0412	Ѧ
Small Ligature Te-Ve	U+0442 U+200D U+0432	Ѧ
Capital Ligature Te-I	U+0422 U+200D U+0418	Ѧ
Small Ligature Te-I	U+0442 U+200D U+0438	Ѧ
Capital Ligature Te-Er	U+0422 U+200D U+0420	Ѧ
Ligature Capital A-Small Te	U+0410 U+200D U+0442	Ѧ
Capital Ligature Te-Soft Sign	U+0422 U+200D U+042c	Ѧ
Small Ligature Te-Soft Sign	U+0442 U+200D U+044C	Ѧ
Small Ligature Te-A	U+0442 U+200D U+0430	Ѧ

6.1.1 Advanced Features of the Font



The Monomakh font offers a number of optional OpenType features that may be turned on or off by the user. These are:

- Stylistic Set 1 (*ss01*) is provided as a temporary workaround to [LibreOffice Bug 85731](#), which does not allow you to specify the hyphenation character in LibreOffice. When turned on, it replaces all instances of U+002D Hyphen-Minus and U+2010 Hyphen with U+005F Low Line (underscore) for use as a hyphenation character. Please note that this feature will be deprecated once the necessary functionality is added to LibreOffice.
 - Stylistic Set 6 (*ss06*) displays U+0456 Cyrillic Small Letter Ukrainian / Belorussian I with one dot above and Stylistic Set 7 (*ss07*) displays the same character with two dots above. By default, U+0456 is displayed with no dots.
 - Stylistic Set 8 (*ss08*) displays the characters U+0417 Cyrillic Capital Letter Ze and U+0437 Cyrillic Small Letter Ze as a “sharp zemlya”, i.e., like the characters U+A640 Cyrillic Capital Letter Zemlya and U+A641 Cyrillic Small Letter Zemlya, respectively. Generally, this change should be handled at the codepoint level, so the use of this feature is discouraged.
 - Stylistic Set 9 (*ss09*) displays the characters U+0427 Cyrillic Capital Letter Che and U+0447 Cyrillic Small Letter Che in their archaic form, with the descender in the middle (e.g., **Ѣ** instead of **ѣ**).
 - Stylistic Set 10 (*ss10*) displays the characters U+0429 Cyrillic Capital Letter Shcha and U+0449 Cyrillic Small Letter Shcha in their modern form, with the descender on the right (e.g., **щ** instead of **ѣ**).
 - Stylistic Set 11 (*ss11*) displays the characters U+044B Cyrillic Small Letter Yeru and U+A651 Cyrillic Small Letter Yeru with Back Yer with the two glyphs connected (e.g., **ѣ** instead of **ѣ**).
 - Stylistic Set 13 (*ss13*) displays the character U+0463 Cyrillic Small Letter Yat with the left stem extended to the baseline (e.g., as **ѣ**). Please note that this is not the same as U+A653 Cyrillic Small Letter Iotified Yat.
 - The same functionality of these Stylistic Sets is provided in OpenType also by the Stylistic Alternatives (*salt*) feature.
-  Previous versions of the font provided Stylistic Set 1 (*ss01*), which displayed U+015E Latin Capital Letter S with Cedilla, U+0162 Latin Capital Letter T with Cedilla, and their lowercase analogs, as U+0218 Latin Capital

Letter S with Comma Below, U+021A Latin Capital Letter T with Comma Below, and their lowercase analogs. However, since the use of U+015E, U+0162 and their lowercase analogs for the encoding of Romanian text is considered erroneous, this feature is deprecated. Users are strongly encouraged to convert their text at the codepoint level to use the correct characters for Romanian orthography. However, for the sake of compatibility with text that has been erroneously encoded, this feature is still available.

-  Stylistic Set 15 (*ss15*), which provides combining Cyrillic letters with an automatic *pokrytie* where warranted by Synodal orthography is also deprecated and may be removed. Users should explicitly encode the *pokrytie* as U+0487 Combining Cyrillic Pokrytie. See [UTN 41: Church Slavonic Typography in Unicode](#) for more information.

Two additional features were available in SIL Graphite only; however support for SIL Graphite has been discontinued. If you need these features, see the [Legacy Fonts package](#):

-  The Graphite feature Convert Arabic Digits to Church Slavonic (*cnum*), when turned on, will automatically display Western Digits (“Arabic numerals”) as Cyrillic numerals. This is helpful, for example, for page numbering in software that does not support Cyrillic numerals.
-  The Graphite feature Convert HIP-6B Keystrokes to Church Slavonic Characters (*hipb*), when turned on, will display text encoded in the legacy HIP codepage as Church Slavonic. The use of this feature is discouraged and users are encouraged instead to convert HIP-encoded text to Unicode.

6.2 Shafarik

The Shafarik font is a specialized font intended for an academic presentation of Old Church Slavonic (OCS) texts written in both the Cyrillic or Glagolitic alphabets. The documentation for the font is provided in a [separate file](#).


7 Decorative Fonts

7.1 Indiction Unicode

The Indiction Unicode font reproduces the decorative style of drop caps used in Synodal Slavonic editions since the late 1800’s.


The original Indyction font was developed by Vladislav V. Dorosh and was distributed as Indyction UCS as part of CSLTeX, licensed under the \LaTeX Project

Public License. The font was reencoded for Unicode and edited by Aleksandr Andreev, and is now distributed as Indiction Unicode under the SIL Open Font License. It is intended for use with *bukvitsi* (drop caps) in modern Church Slavonic editions.

 ѿжѣнѣ мѡжѣ, ѿже не ѿде на совѣтъ нечестивыхъ, ѿ на
пѡтъ грѣшныхъ не стѣ, ѿ на сѣдѣлицѣ гѡбнѣтелей не
сѣде: но въ законѣ гдѣни вола ѣгѡ, ѿ въ законѣ ѣгѡ
пѡчнѣтъ дѣнь ѿ нѡщѣ. ѿ вѣдетъ ѿкъ древо насажденое
при ѿсходницѣхъ вола, ѣже плодъ своѣ дѣтъ во время своѣ,
ѿ лиетъ ѣгѡ не ѡпадѣтъ: ѿ всѣ, ѣлика ѿце творитъ,
ѡспѣетъ.

7.2 Vertograd Unicode

Vertograd Unicode (based on Vertograd UCS by Vlad Dorosh) is another font for decorative dropcaps and titling. The typeface was commonly used in pre-revolution Russian liturgical editions. Note that some letters that are typically not encountered in capital form are not available. Please file an issue if you need a letter that is not available.

 ѿжѣнѣ мѡжѣ, ѿже не ѿде на совѣтъ нечестивыхъ, ѿ на
пѡтъ грѣшныхъ не стѣ, ѿ на сѣдѣлицѣ гѡбнѣтелей не
сѣде: но въ законѣ гдѣни вола ѣгѡ, ѿ въ законѣ ѣгѡ
пѡчнѣтъ дѣнь ѿ нѡщѣ. ѿ вѣдетъ ѿкъ древо насажденое
при ѿсходницѣхъ вола, ѣже плодъ своѣ дѣтъ во время своѣ,
ѿ лиетъ ѣгѡ не ѡпадѣтъ: ѿ всѣ, ѣлика ѿце творитъ,
ѡспѣетъ.

7.3 Cathisma Unicode

Cathisma Unicode is based on Kathisma UCS, designed by Vlad Dorosh. The typeface is used for titling in many 18th–20th century liturgical editions.

ЧИНЪ ОУТРЕНИ ВЪСНОЩАЮ ВДѢНІА

7.4 Oglavie Unicode

Oglavie Unicode is based on Oglavie UCS, designed by Vlad Dorosh. The typeface is also used for decorative titling in 18th–20th century liturgical editions.

ТРИ́ЗЪ НО́ТНАГЪ ПѢНІА

7.5 Pomorsky Unicode

The Pomorsky Unicode font is a close reproduction of the decorative calligraphic style of book and chapter titles, which was most likely developed in the 1700’s by the scribes of the Old Ritualist Vyg River Hermitage (Выговская пустынь). It is seen extensively in the chant manuscripts, liturgical manuscripts, hagiographic and polemical works of the Pomortsy and Fedoseyevtsy communities, and is a traditional and “organic” style of lettering lacking any obvious influence from Western European typography. The Pomorsky typeface was originally designed by Nikita Simmons. It is intended for use with *bukvitsi* (drop caps) and decorative titling.

УМНЪ СЪУСЪУМН.

7.5.1 Advanced Features of the Font

Several versions of many glyphs are provided in the font. The ornate forms of the letters are default and provided at the uppercase Cyrillic codepoints; they should be used as much as possible. Simpler forms can be used whenever the letters need a less ornate appearance, or when diacritics might conflict with the ornamentation (or when the ornamentation of one character will conflict with the ornamentation of another); these simple forms are available as Stylistic Set 1. There are a few additional characters that are stylistic variants, which are provided as Stylistic Alternatives (salt). Since the font is intended for drop caps and titling, lowercase characters are not available. The base form, the “simple” form, and any stylistic alternatives of a character are demonstrated in Table 7.

8 Technical Fonts: FiraSlav

FiraSlav is a monospaced font for Church Slavonic, intended for editing Church Slavonic text in a text editor. All diacritical marks and combining letters are presented as spacing symbols and a monospaced appearance is maintained. The

Table 7: Character shapes provided by Fontelsky Unicode

font includes both regular (FiraSlav Regular) and bold (FiraSlav Bold) flavors. It is especially useful for software and website development applications:

```
$number = /^(?:([$])*(?:_*([$]))*($)?)([клмнѝопч]?)([$o]?)$/;
var letter = '(?:oy|Oy|oy\\S)[ёё' + "ЁЁБбВвГгДдЕеЖжЗзИиКкЛлМмНнОоПпРрСсТтУуФфХхЦцЧчШшЩщЪъЫыЬьЭэЮюЯя]*';
```

Бл̑же́нь мѣжѹ, иѡ́же не иѣде на совѣть нече́стиѢвыхъ

9 Known Issues

See the [Issue Tracker](#). Before reporting issues, please check that your software properly supports OpenType. We suggest checking for expected behavior in Xe_{La}TeX or Lua_{La}TeX.

10 Credits

The authors would like to thank the following people:

- Vladislav Dorosh, who allowed his [Hirmos](#) font to be re-encoded in Unicode and modified, leading to the creation of the Ponomar font.
- Viktor Baranov of the [Manuscripts](#) project, who allowed the re-encoding and modification of his Menaion font.
- Michael Ivanovich for help in designing the characters for Sakha (Yakut), partially taken from his Sakha UCS font.
- Alexey Kryukov, who answered various questions about FontForge, allowed his Monomachus font to be modified and repackaged, and whose extensive documentation for the [Old Standard](#) font was consulted and partially reused.
- Mike Kroutikov, who put together the \TeX package of the fonts.