

Ruwudu - Font Features

Ruwudu is a TrueType font with smart font capabilities added using the OpenType font technology. The Ruwudu font includes a number of optional features that provide alternative rendering that might be preferable for use in some contexts. The sections below enumerates the details of these features. Whether these features are available to users will depend on both the application and the rendering technology being used. Features are available in OpenType. Some applications let the user control certain features such as Stylistic Sets to turn on the rendering of variant characters. However, at this point, many applications do not make use of those features so another solution is needed to show the variant characters. For applications that do not make use of the OpenType Character Variants, you can now download fonts customized with the variant glyphs you choose. Read this document, visit [TypeTuner Web](#), then choose the variants and download your font.

See [Using Font Features](#). Although that page is not targeted at Arabic script support, it does provide a comprehensive list of applications that make full use of the OpenType (and Graphite) font technologies.

See also [Arabic Fonts – Application Support](#). It provides a fairly comprehensive list of applications that make full use of the OpenType font technology.

This page uses web fonts (WOFF) to demonstrate font features and should display correctly in all modern browsers. For a more concise example of how to use Ruwudu as a web font see *Ruwudu-webfont-example.html* in the font package web folder.

If this document is not displaying correctly a PDF version is also provided in the documentation/pdf folder of the release package.

Complete feature list

Language system tags

Affects: U+065C, U+0627 (with U+064E, U+064F, U+0650), U+063F, U+069F, U+0751, U+0763, U+08C3, U+08C4

Unfortunately, the UI needed to access the language-specific behavior is not yet present in many applications. Some Harfbuzz-based apps, e.g., XeTeX, can access language-specific behavior.




Language	Imala e	Touching	Wagaf	Feature setting
default	ٲ	آآٲآٲآٲ	شٲٲٲٲٲ ظظظظٲ ٲٲٲٲٲ ككككٲ ععععٲ قققق	
Hausa	ٲ	آآٲآٲآٲ	شٲٲٲٲٲ ظظظظٲ ٲٲٲٲٲ ككككٲ ععععٲ قققق	lang=ha

Firefox allows you to use U+06DD followed by the digits and proper rendering occurs. Some applications require the following:

- precede the entire sequence (subtending mark plus following digits) with 202D LEFT-TO-RIGHT OVERRIDE
- follow the entire sequence with U+202C POP DIRECTIONAL FORMATTING.

Surrounding the sequence with U+202D and U+202C seems to give the most reliable results in different browsers. However, we have not found a solution that works in Internet Explorer/Edge.

In the example below, the following codepoints are used: U+202D U+06DD U+0031 U+0032 U+0033 U+202C U+202D U+06DD U+0611 U+0622 U+0663 U+202C.

Feature	Sample	Feature setting
Standard		ss02=0 ss03=0
End of ayah A		ss02=1
End of ayah B		ss03=1

Other user font features

Proportional Figures

Ruwudu supports the OpenType **Proportional Figures (pnum)** for Latin digits. *This feature is not supported in TypeTuner Web.*

Affects: U+0030..U+0039

Feature	Sample	Feature setting
Tabular Figures	5 4 3 2 1 0 9 8 7 6	pnum=0
Proportional Figures	5 4 3 2 1 0 9 8 7 6	pnum=1

Contextual Alternates

Ruwudu has a few contextual rules applying subtle differences in the shape and position of certain of the characters depending on context (sometimes shortening, sometimes lengthening or lowering a “tail”). InDesign is able to utilize some of these cross word contextual alternates; however, most applications are not advanced enough to use this behavior in the font. Unfortunately, at this point in the development of the font, combining marks above or below may break these contextual alternates.

Substitutions to avoid collisions in strings with *yeh barree* character**

يٰٓ

يٰٓ

Note!

If you used previous versions of the Alkalami and Alkalami Light font, we made some decisions which were unfortunately not Unicode compliant. Since the initial release of that font, many of these non Unicode compliant characters have been added to Unicode. This font only supports the Unicode compliant

characters. You will need to re-encode some characters to be fully Unicode compliant. The characters affected are listed below:

Glyph	Non Unicode Compliant	→	Glyph	Convert to
ك	0643 ARABIC LETTER KAF	→	ك ككك	06A9 ARABIC LETTER KEHEH
ك	06AD ARABIC LETTER NG	→	ك ككك	0763 ARABIC LETTER KEHEH WITH THREE DOTS ABOVE
ب	067B ARABIC LETTER BEEH (ss05)	→	ب	0751 ARABIC LETTER BEH WITH DOT BELOW AND THREE DOTS ABOVE (use Stylistic Set ss09 if you wish a small <i>wagaf</i>)
ظ	069F ARABIC LETTER TAH WITH THREE DOTS ABOVE (ss05)	→		You may continue to use this codepoint (use Stylistic Set ss09 if you wish a small <i>wagaf</i>)
ع	06A0 ARABIC LETTER AIN WITH THREE DOTS ABOVE (ss05)	→	ع	08C3 ARABIC LETTER GHAIN WITH THREE DOTS ABOVE (use Stylistic Set ss09 if you wish a small <i>wagaf</i>)
ق	06A8 ARABIC LETTER QAF WITH THREE DOTS ABOVE (ss05)	→	ق	08C4 ARABIC LETTER AFRICAN QAF WITH THREE DOTS ABOVE (use Stylistic Set ss09 if you wish a small <i>wagaf</i>)
ي	06CC ARABIC LETTER FARSI YEH (ss05)	→	ي	063F ARABIC LETTER FARSI YEH WITH THREE DOTS ABOVE (use Stylistic Set ss09 if you wish a small <i>wagaf</i>)
ك	0763 ARABIC LETTER KEHEH WITH THREE DOTS ABOVE (ss05)	→		You may continue to use this codepoint (use Stylistic Set ss09 if you wish a small <i>wagaf</i>)
ن	0646 ARABIC LETTER NOON (ss01)	→	ن	08BD ARABIC LETTER AFRICAN NOON (no Stylistic Set required)
ي	064A ARABIC LETTER YEH (ss01)	→	ي	06CC ARABIC LETTER FARSI YEH (no Stylistic Set required)
ف	06A2 ARABIC LETTER FEH WITH DOT MOVED BELOW (ss01)	→	ف	08BB ARABIC LETTER AFRICAN FEH (no Stylistic Set required)
ق	06A7 ARABIC LETTER QAF WITH DOT ABOVE (ss01)	→	ق	08BC ARABIC LETTER AFRICAN QAF (no Stylistic Set required)

This guide is from the [Ruwwudu project](#) version 3 and is copyright © 2015-2025 SIL Global.