CronosPro Support for LaTeX

Sebastian Schubert

v0.1 – 2011/12/19

Contents

1 Overview

The CronosPro package provides support for the CronosPro font family from Adobe. You can use these fonts in a LaTeX document by adding the command

```
\usepackage{CronosPro}
```

to the preamble. This will change the sans serif text font only. If you want to use MyriadPro as your main font, add

```
\renewcommand{\familydefault}{\sfdefault}
```

to your preamble.

Acknowledgements

CronosPro is heavily based on the MinionPro package by Achim Blumensath, Andreas Bühmann and Michael Zedler.

2 Interference with other packages

The CronosPro package automatically loads the following packages: textcomp and fontaxes. If you want to pass options to these packages you can either put the corresponding `\usepackage` command before the `\usepackage{CronosPro}` or you can include the options in the `\documentclass` command.

The CronosPro package includes support files for the microtype package (version 1.8 or higher), consult the package's documentation for further details.

There is also a slight incompatibility with the dcolumn package which expects all figures to have the same width. If you want to use this package you either have to specify the mathtabular option (this is the brute force solution, not recommended), or you can use the `\figureversion{tabular}` command to switch to tabular figures in front of every table (much better, but also more work). In addition, dcolumn sets figures in math mode, hence the choice of math figures (see Section 3) determines if text or lining figures are used.

3 Options

Font selection

The following options specify which version of the fonts you want to use. The default settings are marked with an asterisk*.

smallfamily*	use only regular and bold face
medfamily	use semibold face in addition to smallfamily
noopticals*	use only the optical size Text
opticals	use the optical sizes Caption, Text, Subhead, and Display
slides	use only the optical size Caption (useful for slides)

normalsize*	adapt optical sizes to the normal font size (10 pt, 11 pt, 12 pt)
nonormalsize	use static settings for the optical sizes

Since CronosPro comes in only four different optical sizes we use a variable mapping from font size to the optical size. This means that, both for 10 pt and 11 pt documents, text set in `\small` size will use the Caption size. Sometimes it might be desirable to turn off this automatism – for instance, if you want to load the CronosPro package before the `\documentclass` command. In these cases you can use the nonormalsize option to do so.

Miscellaneous options

scale=<*factor*>	scale the font size by <*factor*>
footnotefigures	use special figures for footnote marks, i.e., example[6,9] instead of example[6,9]. This option can only be used if the footnote marks consist *solely* of figures.

4 Figure selection

CronosPro offers four different figure versions. One can choose between *text figures* (lowercase figures) and *lining figures* (uppercase figures) and one can choose between *proportional* figures (figures with different widths) and *tabular* figures (all figures have the same width, useful mainly for tables).

	text figures	lining figures
proportional	0123456789	0123456789
tabular	0123456789	0123456789

The `\figureversion` command can be used to switch between different figure versions. Possible parameters are:

text, osf	text figures
lining, lf	lining figures
tabular, tab	tabular figures
proportional, prop	proportional figures

Usually it is desirable to set most text with proportional figures and to use tabular figures only in tables and lists. Unfortunately most LaTeX document classes do not support fonts with several figure versions. Use the package tabfigures that patches some common document classes and packages (the standard LaTeX classes, KOMA-Script, memoir, and amsmath) to use tabular figures at some places.

5 Additional font shapes and symbols

In addition to the normal small caps shape `sc` there is a letterspaced version called `ssc`. It is accessible via the commands `\sscshape` and `\textssc`. In order to use the `ssc` shape

throughout your document specify `\renewcommand{\scdefault}{ssc}` in the preamble of your document.

Swash capitals like *'Canadian Mountain Holidays'* are accessed via the `sw` fontshape and the commands `\swshape` and `\textsw`.

`sc`	THIS IS A SAMPLE TEXT
`ssc`	THIS IS A SAMPLE TEXT
`sw`	*This is a Sample Text*

Ornaments can be accessed via the pifont package with the command

`\Pisymbol{CronosPro-Extra}{`⟨*number*⟩`}`

The available glyphs with their numbers are listed in the table below.

100	101	102	103	104	105	106	107	108	109	110	111	112
												

6 Language support

The following encodings are supported:

Latin	OT1, T1, TS1, LY1

7 Searching for figures or for words containing ligatures in PDF documents

Searching for figures or for words containing ligatures in PDF documents may not be possible depending on the way the PDF file was created. The following table gives an overview of which glyphs may cause problems.

font version	program	problems
1.000	Ghostscript, pre-1.40 pdfTEX	LF/TOsF, non-standard ligatures, swashes
1.001, 2.000	Ghostscript, pre-1.40 pdfTEX	LF/OsF/TOsF, ligatures, swashes, small caps
1.00x	Distiller, dvipdfmx	LF/TOsF
1.00x	pdfTEX 1.40	ok
2.000	Distiller, dvipdfmx, pdfTEX 1.40	ok

To make figures and ligatures searchable when using pdfTEX 1.40, you need to enable glyph-to-unicode translation and load the default mapping table:

```
\input glyphtounicode
\pdfgentounicode=1
```

See the pdfTEX manual for details.

8 NFSS classification

Parenthesised combinations are provided via substitutions.

encoding	family	series	shape
OT1, T1, TS1, LY1	CronosPro-OsF, CronosPro-LF, CronosPro-TOsF, CronosPro-TLF	m, b (sb, bx), eb	n, it (sl), sw[1], sc, scit (scsl, scsw), ssc, sscit (sscsl, sscsw)
U	CronosPro-Extra	m, b (sb, bx), eb	n, it (sl)

9 Version history

Version 0.1: First version

10 The main style file

10.1 Options

```
⟨*style⟩
\RequirePackage{kvoptions}
\SetupKeyvalOptions{
  family = Cr,
  prefix = Cr@
}
```

Font sets

The package CronosPro-FontDef adapts the font definitions to the requested font set (see section 12). So we simply pass on the relevant options including the font scale factor; only CronosPro integrals are handled here in CronosPro.

```
\DeclareStringOption[1.]{scale}
\newcommand\Cr@minionint@opticals{-NoOpticals}
\newcommand\Cr@minionint@bold{-Bold}
\DeclareVoidOption{slides}{%
  \def\Cr@minionint@opticals{-NoOpticals}%
  \PassOptionsToPackage{slides}{CronosPro-FontDef}}
\DeclareVoidOption{noopticals}{%
  \def\Cr@minionint@opticals{-NoOpticals}%
  \PassOptionsToPackage{noopticals}{CronosPro-FontDef}}
\DeclareVoidOption{opticals}{%
  \def\Cr@minionint@opticals{}%
```

[1] via substitution in TS1 encoding

```
  \PassOptionsToPackage{opticals}{CronosPro-FontDef}}
\DeclareVoidOption{smallfamily}{%
  \def\Cr@minionint@bold{-Bold}%
  \PassOptionsToPackage{smallfamily}{CronosPro-FontDef}}
\DeclareVoidOption{medfamily}{%
  \def\Cr@minionint@bold{-Semibold}%
  \PassOptionsToPackage{medfamily}{CronosPro-FontDef}}
%\DeclareVoidOption{fullfamily}{%
%  \def\Cr@minionint@bold{-Semibold}%
%  \PassOptionsToPackage{fullfamily}{CronosPro-FontDef}}
\DeclareVoidOption{normalsize}{%
  \PassOptionsToPackage{normalsize}{CronosPro-FontDef}}
\DeclareVoidOption{nonormalsize}{%
  \PassOptionsToPackage{nonormalsize}{CronosPro-FontDef}}
```

Figure style

```
\newcommand\Cr@Text@Fig{OsF}
\newcommand\Cr@Math@Fig{OsF}
\newcommand\Cr@Text@Family{CronosPro-\Cr@Text@Fig}
\newcommand\Cr@Math@Family{CronosPro-\Cr@Math@Fig}
\newcommand\Cr@Math@TFamily{CronosPro-T\Cr@Math@Fig}
\newcommand\Cr@Math@LetterShape{it}
```

```
\DeclareVoidOption{textosf}{\def\Cr@Text@Fig{OsF}}
\DeclareVoidOption{textlf}{\def\Cr@Text@Fig{LF}}
\DeclareVoidOption{mathosf}{\def\Cr@Math@Fig{OsF}}
\DeclareVoidOption{mathlf}{\def\Cr@Math@Fig{LF}}
\DeclareVoidOption{osf}{\setkeys{Cr}{textosf,mathosf}}
\DeclareVoidOption{lf}{\setkeys{Cr}{textlf,mathlf}}
\DeclareVoidOption{mathtabular}{\let\Cr@Math@Family\Cr@Math@TFamily}
```

Miscellaneous options

Footnote figures, extra spacing for the apostrophe.

```
\DeclareVoidOption{footnotefigures}{%
  \def\@makefnmark{%
    \begingroup
    \normalfont
    \fontfamily{CronosPro-Extra}\fontencoding{U}\selectfont
    \@thefnmark
    \endgroup}}
%
\newcommand\Cr@Quote@Spacing{}
\DeclareVoidOption{loosequotes}{%
  \def\Cr@Quote@Spacing{\Cr@Quote@Spacing@Loose}}
```

Defaults

```
\ProcessKeyvalOptions{Cr}\relax
```

10.2 Font declarations

```
\RequirePackage{CronosPro-FontDef}
\@ifpackageloaded{textcomp}{}{\RequirePackage{textcomp}}
```

By default, we use `b` for the bold series. If `CronosPro-Semibold` is not available this might internally be mapped to `CronosPro-Bold` (see CronosPro-FontDef).

```
\edef\sfdefault{\Cr@Text@Family}
```

If a recent verion of microtype is loaded then we implement an option to increase the side bearings of all quote glyphs.

```
\def\Cr@Quote@Spacing@Loose{%
  \@ifpackageloaded{microtype}{}{\RequirePackage[kerning=true]{microtype}}
  \@ifundefined{SetExtraKerning}{}{
    \let\Cr@Set@Quote@Spacing\SetExtraKerning}
%      \SetExtraKerning
%        [ unit = 1em ]
%        { encoding = {OT1,T1,U,LY1},
%          family   = {CronosPro-OsF,CronosPro-LF,CronosPro-TOsF,CronosPro-TLF},
%          shape    = n }
%        { \textquotedblleft = {30,30},  \textquotedblright = {30,30},
%          \textquoteleft    = {30,30},  \textquoteright    = {30,30} }}
}
\newcommand*\Cr@Set@Quote@Spacing[3][]{}
\Cr@Quote@Spacing
\Cr@Set@Quote@Spacing
[ unit = 1em ]
{ encoding = {OT1,T1,U,LY1},
  family   = {CronosPro-OsF,CronosPro-LF,CronosPro-TOsF,CronosPro-TLF},
  shape    = {n,it} }
{ \textquotedblleft = {30,30},  \textquotedblright = {30,30},
  \textquoteleft    = {30,30},  \textquoteright    = {30,30} }
```

10.3 Font selection

The font selection commands such as `\figureversion`, `\textsw`, and `\textssc` are provided by the package fontaxes.

```
\RequirePackage{fontaxes}[2005/05/04]
```

We define an additional short hand for compatibility's sake.

```
\let\oldstylenums\textfigures
```

10.4 pdfTEX to-unicode support

Old versions of CronosPro have non-standard glyph names.

```
\@ifundefined{pdfglyphtounicode}{}{
  \pdfglyphtounicode{uniEFD5}{03DD}% uni03DD
  \pdfglyphtounicode{uniEFED}{02D9}% dotaccent.cap
  \pdfglyphtounicode{uniEFEE}{02D8}% breve.cap
  \pdfglyphtounicode{uniEFF1}{02DB}% ogonek.cap
  \pdfglyphtounicode{uniEFF2}{00B8}% cedilla.cap
```

```
\pdfglyphtounicode{uniEFF3}{02DA}% ring.cap
\pdfglyphtounicode{uniEFF5}{02DC}% tilde.cap
\pdfglyphtounicode{uniEFF7}{02C6}% circumflex.cap
\pdfglyphtounicode{uniF628}{2030}% perthousand.oldstyle
\pdfglyphtounicode{uniF62C}{0028}% parenleft.denominator
\pdfglyphtounicode{uniF62D}{0029}% parenright.denominator
\pdfglyphtounicode{uniF631}{0028}% parenleft.numerator
\pdfglyphtounicode{uniF632}{0029}% parenright.numerator
\pdfglyphtounicode{uniF638}{0030}% zero.slash
\pdfglyphtounicode{uniF639}{0030}% zero.fitted
\pdfglyphtounicode{uniF63A}{0032}% two.fitted
\pdfglyphtounicode{uniF63B}{0033}% three.fitted
\pdfglyphtounicode{uniF63C}{0034}% four.fitted
\pdfglyphtounicode{uniF63D}{0035}% five.fitted
\pdfglyphtounicode{uniF63E}{0036}% six.fitted
\pdfglyphtounicode{uniF63F}{0037}% seven.fitted
\pdfglyphtounicode{uniF640}{0038}% eight.fitted
\pdfglyphtounicode{uniF641}{0039}% nine.fitted
\pdfglyphtounicode{uniF642}{0025}% percent.oldstyle
\pdfglyphtounicode{uniF643}{0030}% zero.taboldstyle
\pdfglyphtounicode{uniF644}{0031}% one.taboldstyle
\pdfglyphtounicode{uniF645}{0032}% two.taboldstyle
\pdfglyphtounicode{uniF646}{0033}% three.taboldstyle
\pdfglyphtounicode{uniF647}{0034}% four.taboldstyle
\pdfglyphtounicode{uniF648}{0035}% five.taboldstyle
\pdfglyphtounicode{uniF649}{0036}% six.taboldstyle
\pdfglyphtounicode{uniF64A}{0037}% seven.taboldstyle
\pdfglyphtounicode{uniF64B}{0038}% eight.taboldstyle
\pdfglyphtounicode{uniF64C}{0039}% nine.taboldstyle
\pdfglyphtounicode{uniF64D}{20A1}% colonmonetary.taboldstyle
\pdfglyphtounicode{uniF64E}{20AC}% Euro.taboldstyle
\pdfglyphtounicode{uniF64F}{0192}% florin.taboldstyle
\pdfglyphtounicode{uniF650}{0023}% numbersign.taboldstyle
\pdfglyphtounicode{uniF651}{00A3}% sterling.taboldstyle
\pdfglyphtounicode{uniF652}{00A5}% yen.taboldstyle
\pdfglyphtounicode{uniF653}{0024}% dollar.taboldstyle
\pdfglyphtounicode{uniF654}{00A2}% cent.taboldstyle
\pdfglyphtounicode{uniF655}{0030}% zero.denominator
\pdfglyphtounicode{uniF656}{0031}% one.denominator
\pdfglyphtounicode{uniF657}{0032}% two.denominator
\pdfglyphtounicode{uniF658}{0033}% three.denominator
\pdfglyphtounicode{uniF659}{0034}% four.denominator
\pdfglyphtounicode{uniF65A}{0035}% five.denominator
\pdfglyphtounicode{uniF65B}{0036}% six.denominator
\pdfglyphtounicode{uniF65C}{0037}% seven.denominator
\pdfglyphtounicode{uniF65D}{0038}% eight.denominator
\pdfglyphtounicode{uniF65E}{0039}% nine.denominator
\pdfglyphtounicode{uniF65F}{002C}% comma.denominator
\pdfglyphtounicode{uniF660}{002E}% period.denominator
\pdfglyphtounicode{uniF661}{0030}% zero.numerator
```

```
  \pdfglyphtounicode{uniF662}{0031}% one.numerator
  \pdfglyphtounicode{uniF663}{0032}% two.numerator
  \pdfglyphtounicode{uniF664}{0033}% three.numerator
  \pdfglyphtounicode{uniF665}{0034}% four.numerator
  \pdfglyphtounicode{uniF666}{0035}% five.numerator
  \pdfglyphtounicode{uniF667}{0036}% six.numerator
  \pdfglyphtounicode{uniF668}{0037}% seven.numerator
  \pdfglyphtounicode{uniF669}{0038}% eight.numerator
  \pdfglyphtounicode{uniF66A}{0039}% nine.numerator
  \pdfglyphtounicode{uniF66B}{002C}% comma.numerator
  \pdfglyphtounicode{uniF66C}{002E}% period.numerator
  \pdfglyphtounicode{uniF66D}{0103}% abreve.sc
  \pdfglyphtounicode{uniF66F}{0105}% aogonek.sc
  \pdfglyphtounicode{uniF671}{0107}% cacute.sc
  \pdfglyphtounicode{uniF672}{010D}% ccaron.sc
  \pdfglyphtounicode{uniF675}{010F}% dcaron.sc
  \pdfglyphtounicode{uniF676}{0111}% dcroat.sc
  \pdfglyphtounicode{uniF678}{011B}% ecaron.sc
  \pdfglyphtounicode{uniF67B}{014B}% eng.sc
  \pdfglyphtounicode{uniF67C}{0119}% eogonek.sc
  \pdfglyphtounicode{uniF67D}{011F}% gbreve.sc
  \pdfglyphtounicode{uniF684}{0133}% ij.sc
  \pdfglyphtounicode{uniF687}{0129}% itilde.sc
  \pdfglyphtounicode{uniF68A}{013A}% lacute.sc
  \pdfglyphtounicode{uniF68B}{013E}% lcaron.sc
  \pdfglyphtounicode{uniF68E}{0144}% nacute.sc
  \pdfglyphtounicode{uniF68F}{0148}% ncaron.sc
  \pdfglyphtounicode{uniF692}{0151}% ohungarumlaut.sc
  \pdfglyphtounicode{uniF695}{0155}% racute.sc
  \pdfglyphtounicode{uniF696}{0159}% rcaron.sc
  \pdfglyphtounicode{uniF698}{015B}% sacute.sc
  \pdfglyphtounicode{uniF699}{015F}% scedilla.sc
  \pdfglyphtounicode{uniF69D}{0165}% tcaron.sc
  \pdfglyphtounicode{uniF69E}{0163}% tcommaaccent.sc
  \pdfglyphtounicode{uniF6A0}{0171}% uhungarumlaut.sc
  \pdfglyphtounicode{uniF6A3}{016F}% uring.sc
  \pdfglyphtounicode{uniF6A4}{0169}% utilde.sc
  \pdfglyphtounicode{uniF6AA}{1EF3}% ygrave.sc
  \pdfglyphtounicode{uniF6AB}{017A}% zacute.sc
  \pdfglyphtounicode{uniF6AC}{017C}% zdotaccent.sc
  \pdfglyphtounicode{uniF6DC}{0031}% one.fitted
}
```

10.5 Superior and inferior figures

We define commands to convert numbers to numerator figures and denominator figures.

```
\def\@for@tok#1:=#2\do#3{%
  \expandafter\def\expandafter\@fortmp\expandafter{#2}%
  \ifx\@fortmp\@empty \else
```

```
    \expandafter\@forloop@tok#2\@nil\@nil\@@#1{#3}%
  \fi}
\def\@forloop@tok#1#2#3\@@#4#5{%
  \def#4{#1}%
  \ifx #4\@nnil \else
    #5%
    \def#4{#2}%
    \ifx #4\@nnil \else
      #5\@iforloop@tok #3\@@#4{#5}%
  \fi\fi}
\def\@iforloop@tok#1#2\@@#3#4{%
  \def#3{#1}%
  \ifx #3\@nnil
    \expandafter\@fornoop
  \else
    #4\relax\expandafter\@iforloop@tok
  \fi
  #2\@@#3{#4}}
%
\newcommand*\Cr@extra@font{%
  \fontencoding{U}\fontfamily{CronosPro-Extra}\selectfont}
\newcommand*\Cr@numerator@fig[1]{{\Cr@extra@font\Cr@@numerator@fig{#1}}}
\newcommand*\Cr@denominator@fig[1]{{\Cr@extra@font\Cr@@denominator@fig{#1}}}
\newcommand*\Cr@superior@fig[1]{{\Cr@extra@font\Cr@@superior@fig{#1}}}
\newcommand*\Cr@inferior@fig[1]{{\Cr@extra@font\Cr@@inferior@fig{#1}}}
\newcommand*\Cr@@numerator@fig[1]{%
  \@for@tok\@nf@fig:=#1\do{%
    \ifcase\@nf@fig
       \char'00%
    \or\char'01%
    \or\char'02%
    \or\char'03%
    \or\char'04%
    \or\char'05%
    \or\char'06%
    \or\char'07%
    \or\char'10%
    \or\char'11%
    \else
      \@latex@error{invalid argument to \string\Cr@@numerator@fig}%
    \fi
    }}
\newcommand*\Cr@@denominator@fig[1]{%
  \@for@tok\@nf@fig:=#1\do{%
    \ifcase\@nf@fig
       \char'20%
    \or\char'21%
    \or\char'22%
    \or\char'23%
    \or\char'24%
```

```
    \or\char'25%
    \or\char'26%
    \or\char'27%
    \or\char'30%
    \or\char'31%
    \else
      \@latex@error{invalid argument to \string\Cr@@denominator@fig}%
    \fi
    }}
\newcommand*\Cr@@superior@fig[1]{%
  \@for@tok\@nf@fig:=#1\do{%
    \ifcase\@nf@fig
       \char'60%
    \or\char'61%
    \or\char'62%
    \or\char'63%
    \or\char'64%
    \or\char'65%
    \or\char'66%
    \or\char'67%
    \or\char'70%
    \or\char'71%
    \else
      \@latex@error{invalid argument to \string\Cr@@superior@fig}%
    \fi
    }}
\newcommand*\Cr@@inferior@fig[1]{%
  \@for@tok\@nf@fig:=#1\do{%
    \ifcase\@nf@fig
       \char'100%
    \or\char'101%
    \or\char'102%
    \or\char'103%
    \or\char'104%
    \or\char'105%
    \or\char'106%
    \or\char'107%
    \or\char'110%
    \or\char'111%
    \else
      \@latex@error{invalid argument to \string\Cr@@inferior@fig}%
    \fi
    }}
```

`\Cr@ensure@text` switches to text mode, if necessary.

```
\newcommand*\Cr@ensure@text[1]{%
  \ifmmode
    \Mn@Text@With@MathVersion{#1}%
  \else
    #1%
```

```
  \fi}
```

\smallfrac and \slantfrac assemble numerical fractions.

```
\newcommand*\@Cr@smallfrac[2]{%
  \leavevmode
  \setbox\@tempboxa
    \vbox{%
      \baselineskip\z@skip%
      \lineskip.25ex%
      \lineskiplimit-\maxdimen
      \ialign{\hfil##\hfil\crcr
              \vbox to 2.13ex{\vss\hbox{\Cr@numerator@fig{#1}}\vskip.68ex}\crcr
              \leavevmode\leaders\hrule height 1.1ex depth -1.01ex\hfill\crcr
              \vtop to 1ex{\vbox{}\hbox{\Cr@denominator@fig{#2}}\vss}\crcr
              \noalign{\vskip-1.47ex}}}%
  \dp\@tempboxa=0.49ex%
  \box\@tempboxa}
\newcommand*\@Cr@slantfrac[2]{%
  {\Cr@extra@font\Cr@@numerator@fig{#1}\kern-0.05em/\kern-0.06em\Cr@@denominator@fig{#2}}}
\DeclareRobustCommand*\smallfrac[2]{\Cr@ensure@text{\kern0.06em\@Cr@smallfrac{#1}{#2}\kern0.09
\DeclareRobustCommand*\slantfrac[2]{\Cr@ensure@text{\kern0.06em\@Cr@slantfrac{#1}{#2}\kern0.09
```

10.6 Additional symbols

```
% fix \r A
\DeclareTextCompositeCommand{\r}{OT1}{A}
{\leavevmode\setbox\z@\hbox{!}\dimen@\ht\z@\advance\dimen@-1ex%
  \ooalign{\hss\raise.67\dimen@\hbox{\char23}\hss\crcr A}}

\DeclareEncodingSubset{TS1}{CronosPro-LF}  {1}%
\DeclareEncodingSubset{TS1}{CronosPro-TLF} {1}%
\DeclareEncodingSubset{TS1}{CronosPro-OsF} {1}%
\DeclareEncodingSubset{TS1}{CronosPro-TOsF}{1}%
\AtBeginDocument{
  \UndeclareTextCommand{\textvisiblespace}{T1}%
  \UndeclareTextCommand{\textcompwordmark}{T1}%
  \UndeclareTextCommand{\textsterling}{T1}%
  \UndeclareTextCommand{\j}{T1}%
  \UndeclareTextCommand{\j}{LY1}%
}
```

10.7 Logos

Correct logos.

```
\def\TeX{T\kern-.1667em\lower.4ex\hbox{E}\kern-.125emX\@}
\DeclareRobustCommand{\LaTeX}{L\kern-.32em%
  {\sbox\z@ T%
    \vbox to\ht\z@{\hbox{\check@mathfonts
        \fontsize\sf@size\z@
        \math@fontsfalse\selectfont
```

```
        A}%
      \vss}%
  }%
  \kern-.15em%
  \TeX}
```

Make the changes take effect. This concludes the main style file.

```
%\normalfont
```
⟨/style⟩

11 Support for character protrusion

The microtype configuration. All four CronosPro families use the same file (cf. section 12).

```
⟨*mtcfg⟩
\SetProtrusion
  [ name     = CronosPro-OT1-Roman ]
  { encoding = OT1,
    family   = {CronosPro-OsF,CronosPro-LF,CronosPro-TOsF,CronosPro-TLF},
    shape    = n }
  {
      A = {40,40},
      F = {  ,60},
      J = {90,  },
      K = {  ,50},
      L = {  ,60},
      T = {50,50},
      V = {40,40},
      W = {30,30},
      X = {50,50},
      Y = {50,50},
      k = {  ,60},
      r = {  ,80},
      t = {  ,100},
      v = {70,70},
      w = {40,40},
      x = {60,60},
      y = {70,70},
      ! = {70,180},
      ( = {60,30},    ) = {30,60},
      [ = {100,160},  ] = {160,100},
    {,} = {440,700},
      . = {660,700},
      : = {400,480},
      ; = {350,440},
      - = {700,700},
    \textendash       = {390,480},  \textemdash        = {220,270},
    \textquotedblleft = {380,250},  \textquotedblright = {250,380},
    \textquoteleft    = {670,450},  \textquoteright    = {450,670},
  }
```

```
\SetProtrusion
  [ name     = CronosPro-T1-Roman,
    load     = CronosPro-OT1-Roman ]
  { encoding = T1,
    family   = {CronosPro-OsF,CronosPro-LF,CronosPro-TOsF,CronosPro-TLF},
    shape    = n }
  {
    023 = {  ,40}, % fft ligature
    032 = {  ,50}, % ft ligature
    191 = {30,30}, % Th ligature
    127 = {620,700}, % hyphen
    \AE = {40,  }, % AE
    \quotesinglbase = {670,670},  \quotedblbase   = {370,370},
    \guilsinglleft  = {500,360},  \guilsinglright = {360,500},
    \guillemotleft  = {320,230},  \guillemotright = {230,320},
  }
\SetProtrusion
  [ name     = CronosPro-OT1-Italic]
  { encoding = OT1,
    family   = {CronosPro-OsF,CronosPro-LF,CronosPro-TOsF,CronosPro-TLF},
    shape    = {it,sl,sw} }
  {
      A = {120,50},
      B = {90,-50},
      C = {50,-60},
      D = {70,-30},
      E = {90,-50},
      F = {100,-40},
      G = {50,-60},
      H = {70,-40},
      I = {150,-90},
      J = {250,-130},
      K = {80,-50},
      L = {90,60},
      M = {60,-40},
      N = {70,-40},
      O = {70,-30},
      P = {70,-110},
      Q = {40,-40},
      R = {80,-50},
      S = {70,-70},
      T = {130,  },
      U = {70,-40},
      V = {120,30},
      W = {90,20},
      X = {50,  },
      Y = {160,  },
      Z = {50,-50},
      d = {60,-60},
```

```
    f = {  ,-190},
  027 = {  ,-70}, % ff ligature
    g = {-70,-70},
    i = {  ,-110},
  025 = {  ,-60}, % dotlessi
  028 = {  ,-60}, % fi ligature
  030 = {  ,-30}, % ffi ligature
    j = {-90,-150},
    p = {-40,  },
    r = {  ,80},
    t = {  ,100},
    v = {90,  },
    w = {60,10},
    x = {90,  },
    ! = {190,40},
    ( = {90,  },    ) = {90,  },
    [ = {90,90},    ] = {120,60},
  {,} = {210,680},
    . = {640,680},
    : = {380,430},
    ; = {   ,430},
    - = {750,750},
  \textquoteleft    = {690,140},  \textquoteright    = {470,230},
  \textendash       = {400,500},  \textemdash        = {220,280},
  \textquotedblleft = {520,130},  \textquotedblright = {520,130},
}
\SetProtrusion
 [ name     = CronosPro-T1-Italic,
   load     = CronosPro-OT1-Italic ]
 { encoding = T1,
   family   = {CronosPro-OsF,CronosPro-LF,CronosPro-TOsF,CronosPro-TLF},
   shape    = {it,sl,sw} }
 {
   023 = {  ,40}, % fft ligature
   032 = {  ,50}, % ft ligature
   191 = {80,30}, % Th ligature
   127 = {660,750}, % hyphen
   \AE = {90,-40}, % AE
   131 = {80,-30}, % Dcaron
   132 = {70,-40}, % Ecaron
   156 = {80,-60}, % IJ
   \OE = {50,-30}, % OE
   188 = {  ,-80}, % ij
   184 = {70,70}, % ydieresis
   253 = {70,70}, % yacute
   \quotesinglbase = {220,700},  \quotedblbase   = {130,400},
   \guilsinglleft  = {500,180},  \guilsinglright = {350,350},
   \guillemotleft  = {310,110},  \guillemotright = {230,230},
 }
```

We have no protruding values for small caps yet. The following stubs are unnecessary at the moment, but they are here as a reminder.

```
\SetProtrusion
  [ name     = CronosPro-OT1-Smallcaps ]
  { encoding = OT1,
    family   = {CronosPro-OsF,CronosPro-LF,CronosPro-TOsF,CronosPro-TLF},
    shape    = {sc,ssc} }
  {}
\SetProtrusion
  [ name     = CronosPro-T1-Smallcaps,
    load     = CronosPro-OT1-Smallcaps ]
  { encoding = T1,
    family   = {CronosPro-OsF,CronosPro-LF,CronosPro-TOsF,CronosPro-TLF},
    shape    = {sc,ssc} }
  {}
\SetProtrusion
  [ name     = CronosPro-OT1-SmallcapsItalic ]
  { encoding = OT1,
    family   = {CronosPro-OsF,CronosPro-LF,CronosPro-TOsF,CronosPro-TLF},
    shape    = {scit,sscit} }
  {}
\SetProtrusion
  [ name     = CronosPro-T1-SmallcapsItalic,
    load     = CronosPro-OT1-SmallcapsItalic ]
  { encoding = T1,
    family   = {CronosPro-OsF,CronosPro-LF,CronosPro-TOsF,CronosPro-TLF},
    shape    = {scit,sscit} }
  {}
\SetProtrusion
  [ name     = CronosPro-other-Roman ]
  { encoding = {U},
    family   = {CronosPro-OsF,CronosPro-LF,CronosPro-TOsF,CronosPro-TLF},
    shape    = n }
  {
      ! = {70,180},
      ( = {60,30},    ) = {30,60},
      [ = {100,160},  ] = {160,100},
    {,} = {440,700},
      . = {660,700},
      : = {400,480},
      ; = {350,440},
      - = {700,700},
    \textendash       = {390,480},  \textemdash        = {220,270},
    \textquotedblleft = {380,250},  \textquotedblright = {250,380},
    \textquoteleft    = {670,450},  \textquoteright    = {450,670},
  }
\SetProtrusion
  [ name     = CronosPro-other-Italic ]
```

```
  { encoding = {U},
    family   = {CronosPro-OsF,CronosPro-LF,CronosPro-TOsF,CronosPro-TLF},
    shape    = {it,sl,sw} }
  {
      ! = {190,40},
      ( = {90,  },    ) = {90,  },
      [ = {90,90},    ] = {120,60},
    {,} = {210,680},
      . = {640,680},
      : = {380,430},
      ; = {   ,430},
      - = {750,750},
    \textquoteleft    = {690,140},  \textquoteright    = {470,230},
    \textendash       = {400,500},  \textemdash        = {220,280},
    \textquotedblleft = {520,130},  \textquotedblright = {520,130},
  }
⟨/mtcfg⟩
```

12 Font definition files

As all the font definitions look the same we introduce macros to ease the configuration. These macros are stored in the file CronosPro-FontDef.sty which is included by every FD file. Note that CronosPro-FontDef.sty will be included several times and that we do not know in which context the code is executed. Therefore, we have to define all non-private commands as globals.

Since this package should be loadable in an FD file we have to avoid all `\preambleonly` commands. Therefore, we use `\ProvidesFile` instead of `\ProvidesPackage`.

We add a guard so that this file is executed only once even if it is included multiple times.

```
⟨*fontdef⟩
\ifx\Cr@DeclareFontShape\@undefined\else\endinput\fi
```

We distinguish between being loaded directly or via `\usepackage` in the preamble by checking `\@nodocument`.

```
\ifx\@nodocument\relax
  \input{otfontdef.sty}
\else
  \NeedsTeXFormat{LaTeX2e}
  \RequirePackage{otfontdef}
\fi
```

Reset `\escapechar` (which is set to -1 in FD files) to make `\newcommand` work. The additional group does not harm; we have to make the important commands global anyway.

```
\ifx\@nodocument\relax
  \begingroup\escapechar`\\
\fi
```

These are the default values if it is impossible to process options.

```
\newcommand\Cr@option@opticals{noopticals}
\newcommand\Cr@option@fontset{smallfamily}
```

```
\newdimen\Cr@option@normalsize
\global\Cr@option@normalsize10pt
```

Whether we should adapt the configuration to the `\normalsize` of the document. This switch is only needed locally.

```
\newif\ifCr@option@normalsize
\Cr@option@normalsizetrue
```

```
\ifx\@nodocument\relax\else
  \DeclareOption{slides}     {\let\Cr@option@opticals\CurrentOption}
  \DeclareOption{opticals}   {\let\Cr@option@opticals\CurrentOption}
  \DeclareOption{noopticals} {\let\Cr@option@opticals\CurrentOption}
  \DeclareOption{smallfamily}{\let\Cr@option@fontset\CurrentOption}
  \DeclareOption{medfamily}  {\let\Cr@option@fontset\CurrentOption}
%  \DeclareOption{fullfamily} {\let\Cr@option@fontset\CurrentOption}
  \DeclareOption{normalsize}  {\Cr@option@normalsizetrue}
  \DeclareOption{nonormalsize}{\Cr@option@normalsizefalse}
  \ExecuteOptions{smallfamily,noopticals,normalsize}
  \ProcessOptions\relax
\fi
```

The method to determine the main font size is inspired by microtype's implementation.

```
\ifCr@option@normalsize
  \begingroup
  \def\set@fontsize#1#2#3#4\@nil{%
    \@defaultunits\global\Cr@option@normalsize#2pt\relax\@nnil}%
  \normalsize\@nil
  \endgroup
\fi
```

We use `\otf@makeglobal` from otfontdef to "export" the definitions that are needed globally.

```
\otf@makeglobal{Cr@option@opticals}
\otf@makeglobal{Cr@option@fontset}
\ifx\@nodocument\relax\else
  \PackageInfo{CronosPro-FontDef}{%
    Configuration:\space\Cr@option@fontset,\space\Cr@option@opticals,\space
    normalsize=\the\Cr@option@normalsize}%
\fi
```

Configuration database

```
\newcount\Cr@config@cnt
\Cr@config@cnt=0
\newcommand\Cr@curr@config{Cr@config@\romannumeral\Cr@config@cnt}
```

These commands help in setting up the configuration database. They do not need to be global. But the config database itself has to be.

#3 is added to all instances listed in #2 of configuration class #1. #3 is read with NFSS catcodes.

```
\newcommand\Cr@AddToConfig{%
```

```
  \begingroup
  \nfss@catcodes
  \expandafter\endgroup
  \Cr@AddToConfig@
}
\newcommand\Cr@AddToConfig@[3]{%
  \advance\Cr@config@cnt\@ne
  \@namedef{\Cr@curr@config}{#3}%
  \otf@makeglobal{\Cr@curr@config}
⟨debug & show⟩\expandafter\show\csname\Cr@curr@config\endcsname
  \@for\Cr@tempa:=#2\do{%
    \@ifundefined{Cr@config@#1@\Cr@tempa}{%
      \@temptokena{}%
    }{%
      \@temptokena\expandafter\expandafter\expandafter
        {\csname Cr@config@#1@\Cr@tempa\endcsname}%
    }%
    \@expandtwoargs\@namedef{Cr@config@#1@\Cr@tempa}{%
      \the\@temptokena
      \expandafter\noexpand\csname\Cr@curr@config\endcsname
    }%
    \otf@makeglobal{Cr@config@#1@\Cr@tempa}% perhaps defer to only execute once
⟨debug & show⟩\expandafter\show\csname Cr@config@#1@\Cr@tempa\endcsname
  }%
}
```

Let us look at an example of how the configuration database looks internally for (shape, sw), which is specified below in three steps. The following lines show different depths of expansion of the macro \Cr@config@shape@sw, which finally yields the complete configuration:

```
\Cr@config@shape@sw
\Cr@config@xi \Cr@config@xiv \Cr@config@xv
<-8>otf*[spacing=l1]<->otf*[variant=swash]<->otf*CronosPro-It
```

The following commands are used in the Declare...Family commands to access the previously built configuration database. They must be expandable. #3 is used as a default if no entry is found in the database.

```
\newcommand*\Cr@UseConfig[2]{%
  \Cr@UseConfigOrDefault{#1}{#2}{}%
}
\newcommand*\Cr@UseConfigOrDefault[3]{%
  \@ifundefined{Cr@config@#1@#2}{#3}%
    {\@nameuse{Cr@config@#1@#2}}%
}
\newcommand*\Cr@TheConfig[2]{%
  \@ifundefined{Cr@config@#1@#2}{}{%
    \expandafter\noexpand\csname Cr@config@#1@#2\endcsname
  }%
}
\otf@makeglobal{Cr@UseConfig}
```

```
\otf@makeglobal{Cr@UseConfigOrDefault}
\otf@makeglobal{Cr@TheConfig}
```

The size range in the configuration has to be divided by the scaling factor to take the changed size into account because the scaling takes place after choosing the right combination. Provide calculation routine here.

```
\RequirePackage{fltpoint}
\fpDecimalSign{.}
\newcommand*{\Cr@calc@bsize}[2]{\fpDiv{#1}{#2}{\Cr@scale}}
```

Here comes the configuration.

```
\Cr@calc@bsize{\Cr@s@capt}{8.5}
\Cr@calc@bsize{\Cr@s@text}{13.1}
\Cr@calc@bsize{\Cr@s@subh}{20}
\Cr@AddToConfig{opticals}{opticals}{
             <-\Cr@s@capt>  otf* [optical=Capt]
  <\Cr@s@capt-\Cr@s@text>  otf* [optical=Text]
  <\Cr@s@text-\Cr@s@subh>  otf* [optical=Subh]
  <\Cr@s@subh->            otf* [optical=Disp]
}
\Cr@AddToConfig{opticals}{noopticals}{
      <->     otf* [optical=Text]
}
\Cr@AddToConfig{opticals}{slides}{
      <->     otf* [optical=Capt]
}
```

```
\ifdim\Cr@option@normalsize<10.1pt
  \Cr@calc@bsize{\Cr@s@semif}{6}
  \Cr@calc@bsize{\Cr@s@medif}{8.5}
\else
  \Cr@calc@bsize{\Cr@s@semif}{6}
  \Cr@calc@bsize{\Cr@s@medif}{10.1}
\fi
\Cr@AddToConfig{fontset/weight}{fullfamily/m}{
             < -\Cr@s@semif>  otf* [weight=Semibold]
  <\Cr@s@semif-\Cr@s@medif>  otf* [weight=Medium]
  <\Cr@s@medif->             otf* [weight=Regular]
}
\Cr@calc@bsize{\Cr@s@semim}{6}
\Cr@AddToConfig{fontset/weight}{medfamily/m}{
              <-\Cr@s@semim> otf* [weight=Semibold]
  <\Cr@s@semim->            otf* [weight=Regular]
}
\Cr@AddToConfig{fontset/weight}{smallfamily/m}{
      <->     otf* [weight=Regular]
}
%
\Cr@calc@bsize{\Cr@s@bold}{6}
\Cr@AddToConfig{fontset/weight}{fullfamily/b,medfamily/b}{
             <-\Cr@s@bold>  otf* [weight=Bold]
```

```
  <\Cr@s@bold->            otf* [weight=Semibold]
}
\Cr@AddToConfig{fontset/weight}{smallfamily/b}{
      <->     otf* [weight=Bold]
}
%
\Cr@AddToConfig{weight}{eb}{
      <->     otf* [weight=Bold]
}

\Cr@AddToConfig{shape}{ssc,sscit}{
      <->     otf* [spacing=l2]
}
\Cr@calc@bsize{\Cr@s@spac}{8}
\Cr@AddToConfig{shape}{n,it,sw,sc,scit}{
      <-\Cr@s@spac>    otf* [spacing=l1]
}
\Cr@AddToConfig{encoding/shape}{U/n,U/it}{
      <->     otf* [spacing=]
}
%
\Cr@AddToConfig{shape}{sc,ssc,scit,sscit}{
      <->     otf* [variant=sc]
}
\Cr@AddToConfig{shape}{sw}{
      <->     otf* [variant=swash]
}

\Cr@AddToConfig{shape}{it,scit,sscit,sw}{
      <->     otf* CronosPro-It
}
\Cr@AddToConfig{shape}{n,sc,ssc}{
      <->     otf* CronosPro
}
\Cr@AddToConfig{encoding/shape}{OML/it}{
      <->     otf* [figures=] CronosPro-Mixed
}
\Cr@AddToConfig{encoding/shape}{OML/n}{
      <->     otf* [figures=] CronosPro-French
}
\Cr@AddToConfig{scale}{scale}{
      <->     otf* [scale=\Cr@scale]
}
```

Substitutions

```
\Cr@AddToConfig{sub:series} {sb}    {b}
\Cr@AddToConfig{sub:series} {bx}    {b}
\Cr@AddToConfig{sub:shape}  {sl}    {it}
\Cr@AddToConfig{sub:shape}  {scsl}  {scit}
\Cr@AddToConfig{sub:shape}  {sscsl} {sscit}
\Cr@AddToConfig{sub:shape}  {scsw}  {scit}
```

```
\Cr@AddToConfig{sub:shape}  {sscsw} {sscit}
\Cr@AddToConfig{sub:encoding/shape}{TS1/sw}{it}
```

Code for the last argument of `\DeclareFontShape`

```
\Cr@AddToConfig{code:shape}{sw}{
  \skewchar\font='337
}
```

Declaration of font families and shapes

```
\newcommand*\Cr@DeclareFontShape[6][]{%
```

Check if any substitutions are specified.

```
  \edef\@tempa{%
    \Cr@UseConfig{sub:series}{#4}%
    \Cr@UseConfigOrDefault{sub:encoding/shape}{#2/#5}{%
      \Cr@UseConfig{sub:shape}{#5}}%
  }%
  \ifx\@tempa\@empty
```

Collect the configuration and declare the font shape. `\DeclareFontShape` fully expands its fifth argument (with our macros `\Cr@UseConfig` in it), but we have to retrieve the code for the sixth argument ourselves.

```
    \@temptokena={%
      \DeclareFontShape{#2}{#3-#6}{#4}{#5}{%
        \Cr@UseConfig{opticals}      {\Cr@option@opticals}%
        \Cr@UseConfig{fontset/weight}{\Cr@option@fontset/#4}%
        \Cr@UseConfig{weight}        {#4}%
        \Cr@UseConfig{encoding/shape}{#2/#5}%
        \Cr@UseConfig{shape}         {#5}%
        \Cr@UseConfig{scale}         {scale}%
      }}%
    \edef\@tempa{\the\@temptokena{\Cr@TheConfig{code:shape}{#5}}}%
    \@tempa
  \else
```

Generate the substitution. (All substitutions are silent at the moment.)

```
    \DeclareFontShape{#2}{#3-#6}{#4}{#5}{%
      <->ssub*#3-#6%
      /\Cr@UseConfigOrDefault{sub:series}{#4}{#4}%
      /\Cr@UseConfigOrDefault{sub:encoding/shape}{#2/#5}{%
        \Cr@UseConfigOrDefault{sub:shape}{#5}{#5}}%
    }{}%
  \fi
}
\otf@makeglobal{Cr@DeclareFontShape}
\otf@makeglobal{\string\Cr@DeclareFontShape}
```

#2 contains the encoding, #3 the family, and #1 a list of figure versions (or `Extra`).

```
\newcommand*\Cr@DeclareLargeFontFamily[3][LF,OsF,TLF,TOsF]{%
  \Cr@DeclareFontFamily{#1}{#2}{#3}
    {m,sb,b,bx,eb} {n,it,sc,ssc,scit,sscit,sw,scsl,scsw,sscsl,sscsw,sl}%
```

```
}
\newcommand*\Cr@DeclareSmallFontFamily[3][LF,OsF,TLF,TOsF]{%
  \Cr@DeclareFontFamily{#1}{#2}{#3}
    {m,sb,b,bx,eb} {n,it,sl}%
}
\newcommand*\Cr@DeclareMathFontFamily[3][TOsF]{%
  \Cr@DeclareFontFamily[\skewchar\font=255]{#1}{#2}{#3}
    {m,sb,b,bx,eb} {n,it}%
}
```

An additional macro `\csname\string\foo\endcsname` is generated by `\newcommand` for processing an optional argument of `\foo`.

```
\otf@makeglobal{Cr@DeclareLargeFontFamily}
\otf@makeglobal{\string\Cr@DeclareLargeFontFamily}
\otf@makeglobal{Cr@DeclareSmallFontFamily}
\otf@makeglobal{\string\Cr@DeclareSmallFontFamily}
\otf@makeglobal{Cr@DeclareMathFontFamily}
\otf@makeglobal{\string\Cr@DeclareMathFontFamily}
\newcommand*\Cr@DeclareFontFamily[6][]{%
  \@for\Cr@variant:=#2\do{%
    \DeclareFontFamily {#3}{#4-\Cr@variant}{#1}%
  }%
  \Cr@DeclareFontShapes{#3}{#4}
    {#5} {#6} {#2}%
}
\otf@makeglobal{Cr@DeclareFontFamily}
\otf@makeglobal{\string\Cr@DeclareFontFamily}
\newcommand*\Cr@DeclareFontShapes[5]{%
  \@for\Cr@series:=#3\do{%
    \@for\Cr@shape:=#4\do{%
      \@for\Cr@variant:=#5\do{%
        \Cr@DeclareFontShape{#1}{#2}{\Cr@series}{\Cr@shape}{\Cr@variant}%
      }%
    }%
  }%
}
\otf@makeglobal{Cr@DeclareFontShapes}
```

Adjust font dimension #1 of the current font. The function in #2 should replace the old value in dimen `\Cr@fontdimen` with a new one (which may depend on other parameters like `\f@size`).

```
\newdimen\Cr@fontdimen
\newcommand*\Cr@adjust@fontdimen[2]{%
  \Cr@fontdimen=\fontdimen#1\font
  #2%
  \fontdimen#1\font=\Cr@fontdimen
}
\otf@makeglobal{Cr@adjust@fontdimen}
\ifx\@nodocument\relax
  \endgroup
```

```
\fi
⟨*debug⟩
\newcommand\old@DeclareFontFamily{}
\let\old@DeclareFontFamily\DeclareFontFamily
\renewcommand\DeclareFontFamily[3]{
  \begingroup\escapechar`\\%
  \edef\@tempa{\noexpand\DeclareFontFamily{#1}{#2}}%
  \@temptokena\expandafter{\@tempa{#3}}%
  \message{\the\@temptokena}%
  \endgroup
  \old@DeclareFontFamily{#1}{#2}{#3}%
}
\newcommand\old@DeclareFontShape{}
\let\old@DeclareFontShape\DeclareFontShape
\renewcommand\DeclareFontShape[6]{
  \begingroup\escapechar`\\%
  \edef\@tempa{\noexpand\DeclareFontShape{#1}{#2}{#3}{#4}{#5}}%
  \@temptokena\expandafter{\@tempa{#6}}%
  \message{\the\@temptokena}%
  \endgroup
  \old@DeclareFontShape{#1}{#2}{#3}{#4}{#5}{#6}%
}
⟨/debug⟩
```

We define font family aliases so that we can place all configurations for the CronosPro family variants into one microtype file: mt-CronosPro.cfg. We use microtype's hook if microtype has not been loaded yet (which should be the case); otherwise we can execute the alias definitions directly.

```
\gdef\Cr@MicroType@Aliases{%
  \DeclareMicrotypeAlias{CronosPro-LF}{CronosPro}%
  \DeclareMicrotypeAlias{CronosPro-OsF}{CronosPro}%
  \DeclareMicrotypeAlias{CronosPro-TLF}{CronosPro}%
  \DeclareMicrotypeAlias{CronosPro-TOsF}{CronosPro}%
}
\@ifundefined{Microtype@Hook}{%
  \global\let\Microtype@Hook\Cr@MicroType@Aliases
}{%
  \g@addto@macro\Microtype@Hook{\Cr@MicroType@Aliases}%
}%
\@ifundefined{DeclareMicroTypeAlias}{}{\Cr@MicroType@Aliases}%
⟨/fontdef⟩
```

Using these macros the various FD files become simple one-liners.

```
⟨*fd⟩
\input{CronosPro-FontDef.sty}%
⟨Uextra⟩     \Cr@DeclareSmallFontFamily[Extra]{U}  {CronosPro}
⟨OT1⟩         \Cr@DeclareLargeFontFamily       {OT1}{CronosPro}
⟨T1⟩          \Cr@DeclareLargeFontFamily       {T1} {CronosPro}
⟨LY1⟩         \Cr@DeclareLargeFontFamily       {LY1}{CronosPro}
⟨TS1⟩         \Cr@DeclareLargeFontFamily       {TS1}{CronosPro}
```

826 ⟨/fd⟩