

# adfor

Clea F. Rees\*

v1.3 (SVN Rev: 10986) 2025/03/31

## Abstract

Hirwen Harendal, Arkandis Digital Foundry (ADF) has produced Ornaments ADF. This guide outlines the T<sub>E</sub>X/L<sup>A</sup>T<sub>E</sub>X support provided with version 1.001 of the font in postscript type 1 format.

## §1 Introduction

This document explains how to use the T<sub>E</sub>X/L<sup>A</sup>T<sub>E</sub>X support included with version 1.001 of Ornaments ADF in postscript type 1 format. The font was developed by Hirwen Harendal of the Arkandis Digital Foundry (ADF), and information about the font itself, together with a copy of the font in opentype format, can be found at <http://pagesperso-orange.fr/arkandis/ADF/tugfonts.htm>. The font is released under the GPL. For details, see README, NOTICE and COPYING.

The T<sub>E</sub>X/L<sup>A</sup>T<sub>E</sub>X support package consists of all files listed in manifest.txt and these files are released under the L<sup>A</sup>T<sub>E</sub>X Project Public Licence as explained in the included licensing notices and README. Please let me know of any problems so that I can solve them if I can. If you can correct the problems and send me the fix, that would be even better. Unlike the font itself, the T<sub>E</sub>X/L<sup>A</sup>T<sub>E</sub>X support is somewhat experimental.

adfor includes a copy of the font in type 1 format (OrnamentsADF.pfb, OrnamentsADF.pfm and OrnamentsADF.afm), documentation and support files for T<sub>E</sub>X/L<sup>A</sup>T<sub>E</sub>X including a L<sup>A</sup>T<sub>E</sub>X package file, adfor.sty.

## §2 The support package

```
adfor (pkg.) \usepackage[options]{adfor}
```

---

\*Bug tracker: [codeberg.org/cfr/nfssext/issues](https://codeberg.org/cfr/nfssext/issues) | Code: [codeberg.org/cfr/nfssext](https://codeberg.org/cfr/nfssext) | Mirror: [github.com/cfr42/nfssext](https://github.com/cfr42/nfssext)

`adorn` provides access to the ornaments and symbols in `OrnamentsADF` via two sets of commands. First, it provides a single command which takes a range of arguments. The different arguments determine which ornament is typeset. Second, it provides a separate command for each ornament. The choice of command determines which ornament is typeset. The two mechanisms are equivalent<sup>1</sup>.

The package supports a lonely option to scale the fonts.

`scale (opt.) = <scaling factor>`

Scale the font by `<scaling factor>`, which should be a positive integer or simple decimal such as 2 or 1.2. This option is intended for cases where the fonts should be scaled to match other fonts used in the document e.g. for consistency with the size of regular text or superscript markers.

Initially empty, which is equivalent to 1 but more efficient.

## §2.1 One command; many arguments

`adorn` provides the command `\adorn{}` which takes a single numerical argument. There are 75 ornaments in the font which can be produced by feeding the relevant number between 1 and 75 to `\adorn{}`<sup>2</sup>:

1: ◀	16: ⤿	31: ◉	46: ⤿	61: ⚙
2: ▶	17: ⤿	32: ⚙	47: ⤿	62: ⚙
3: *	18: ⤿	33: ⚙	48: ⤿	63: ⚙
4: *	19: ⤿	34: ⚙	49: ⤿	64: ⚙
5: *	20: ⤿	35: ⚙	50: ⤿	65: ⚙
6: *	21: ⤿	36: ⚙	51: ⤿	66: ⚙
7: *	22: ⤿	37: ⚙	52: ⤿	67: ⚙
8: *	23: ⤿	38: ⚙	53: ⤿	68: ⚙
9: *	24: ⤿	39: ⚙	54: ⤿	69: ⚙
10: *	25: ⤿	40: ⚙	55: ⚙	70: ◀
11: *	26: ⤿	41: ⚙	56: ⚙	71: ◻
12: *	27: ⚙	42: ◀	57: ⚙	72: ▶
13: ⚙	28: ⚙	43: ▶	58: ⚙	73: •
14: ◊	29: ⚙	44: ⤿	59: ⚙	74: §
15: ⚙	30: ⚙	45: ⤿	60: ⚙	75: §

<sup>1</sup>The only difference is that the first allows you to typeset a space by passing it the argument `o` whereas there is no command to typeset the space in the second set. For all practical purposes, this difference is irrelevant since you should not use such a command to typeset a space in `TEX` in any case and it is difficult to see why anybody would want to.

<sup>2</sup>As mentioned above, the argument `o` will simply typeset a space and should be avoided as using it may interfere with `TEX`'s spacing algorithms. The problem is that `TEX` will not recognise it as a space and so will treat it instead as a character.

For example,

```
% \adorn{21}\quad\adorn{11}\quad\adorn{49}
%
```

produces:



## §2.2 Many commands; no arguments

In addition to the numerical interface, a number of additional commands are provided as an alternative means of accessing the various symbols and ornaments. The following list groups them roughly according to kind. In each case, the number of the ornament is given first. This may be used directly with the `\adorn{}` command as explained above. The alternative command is given next. This command may be used to typeset the same ornament. For example both `\adorn{14}` and `\adfdiamond` produce  $\diamond$ . Finally, the ornament produced by the two commands is typeset to their right.

no.	command		no.	command
BASIC SYMBOLS & SHAPES				
74	§	<code>\adfS</code>		§
75	Œ	<code>\adfgee</code>		Œ
14	◇	<code>\adfdiamond</code>		◇
71	□	<code>\adfsquare</code>		□
73	•	<code>\adfbullet</code>		•
FANCY ASTERISKS & BULLETS				
3	✱	<code>\adfast1</code>		✱
4	✱	<code>\adfast2</code>		✱
5	✱	<code>\adfast3</code>		✱
6	✱	<code>\adfast4</code>		✱
7	✱	<code>\adfast5</code>		✱
8	✱	<code>\adfast6</code>		✱
9	✱	<code>\adfast7</code>		✱
10	✱	<code>\adfast8</code>		✱
11	✱	<code>\adfast9</code>		✱
12	✱	<code>\adfast{10}</code>		✱
ARROWS & ARROWHEADS				
70	↤	<code>\adfhalfleftarrow</code>		↤
72	↠	<code>\adfhalfrightarrow</code>		↠

---

BASIC SYMBOLS & SHAPES CONT.













































---

42	◀	<code>\adffleftarrowhead</code>	◀
43	▶	<code>\adffrightarrowhead</code>	▶
1	◀	<code>\adfhalfleftarrowhead</code>	◀
2	▶	<code>\adfhalfrightarrowhead</code>	▶

---

FLOURISHES





---

20		<code>\adfflourishleft</code>	
48		<code>\adfflourishright</code>	
21		<code>\adfflourishlefthdouble</code>	
49		<code>\adfflourishrightdouble</code>	
17		<code>\adfopenflourishleft</code>	
45		<code>\adfopenflourishright</code>	
18		<code>\adfclosedflourishleft</code>	
46		<code>\adfclosedflourishright</code>	
22		<code>\adfsingleflourishleft</code>	
50		<code>\adfsingleflourishright</code>	
19		<code>\adfdoubleflourishleft</code>	
47		<code>\adfdoubleflourishright</code>	
26		<code>\adftripleflourishleft</code>	
54		<code>\adftripleflourishright</code>	
23		<code>\adfsharpflourishleft</code>	
51		<code>\adfsharpflourishright</code>	
24		<code>\adfdoublesharpflourishleft</code>	
52		<code>\adfdoublesharpflourishright</code>	
25		<code>\adsickleflourishleft</code>	
53		<code>\adsickleflourishright</code>	
16		<code>\adfwavesleft</code>	
44		<code>\adfwavesright</code>	

---

FLOWERS

















---

60		<code>\adfflowerleft</code>	
32		<code>\adfflowerright</code>	

---

LEAVES

---













































66		<code>\adffleafleft</code>	
38		<code>\adffleafright</code>	
59		<code>\adfsolidleafleft</code>	
31		<code>\adfsolidleafright</code>	
13		<code>\adfhalfleafleft</code>	
15		<code>\adfhalfleafright</code>	
58		<code>\adfoutlineleafleft</code>	
30		<code>\adfoutlineleafright</code>	

---

---

BASIC SYMBOLS & SHAPES CONT.

---

68		<code>\adfsmallleafleft</code>	
40		<code>\adfsmallleafright</code>	
64		<code>\adfflatleafleft</code>	
36		<code>\adfflatleafright</code>	
57		<code>\adfflatleafoutlineleft</code>	
29		<code>\adfflatleafoutlineright</code>	
65		<code>\adfflatleafsolidleft</code>	
37		<code>\adfflatleafsolidright</code>	
67		<code>\adfdownleafleft</code>	
39		<code>\adfdownleafright</code>	
61		<code>\adfdownhalfleafleft</code>	
33		<code>\adfdownhalfleafright</code>	
55		<code>\adfflatdownhalfleafleft</code>	
27		<code>\adfflatdownhalfleafright</code>	
56		<code>\adfflatdownoutlineleafleft</code>	
28		<code>\adfflatdownoutlineleafright</code>	
35		<code>\adfhangingleafleft</code>	
63		<code>\adfhangingleafright</code>	
69		<code>\adfsmallhangingleafleft</code>	
41		<code>\adfsmallhangingleafright</code>	
62		<code>\adfhangingleafleft</code>	
34		<code>\adfhangingleafright</code>	

---

So,

`\adfflourishleftdouble\quad\adfast9\quad\adfflourishrightdouble`

will produce the same output as the example code given in the previous section:



## §A Implementation

You do not need to read the remainder of this document in order to install or use the fonts.

### §A.1 Package

Simple wrappers.

```

1 \NeedsTeXFormat{LaTeX2e}
2 \RequirePackage{svn-prov}
3 \ProvidesPackageSVN[\filebase.sty]{$Id: adorn.dtx 10986 2025-03-31 05:38:02Z cfrees $}[v1.3 \
4 \DefineFileInfoSVN[adorn]
5 \newif\if@adorn@digonnew

```

Copied verbatim, excepting format and modulo package/module name from Joseph Wright's `siunitx.sty` under LPPL

```

6 \@ifundefined{ExplLoaderFileDate}{%
7   \IfFileExists{expl3.sty}{%
8     \RequirePackage{expl3}%
9   }{%
10    \@adorn@digonnewfalse
11  }%
12 }{\@adorn@digonnewtrue}

```

`scale` takes a factor by which to scale the fonts. This is empty by default, which is equivalent to 1, but more efficient.

```

13 \if@adorn@digonnew
14 \ExplSyntaxOn
15 \keys_define:nn { adorn }
16 {
17   scale .tl_set:N = \adorn@scale,
18   scale .initial:V = \@empty,
19 }
20 \else
21 \let\adorn@scale\@empty
22 \fi

```

Provide `\ProcessKeyOptions`, `\IfFormatAtLeastTF` on older kernels. Joseph Wright: from `siunitx.sty` ; <https://chat.stackexchange.com/transcript/message/64327823#64327823>

```

23 %%%%%%%%%%%
24 \providecommand \IfFormatAtLeastTF { \@ifl@t@r \fmtversion }
25 \IfFormatAtLeastTF { 2022-06-01 }
26 {
27   \ProcessKeyOptions [ adorn ]
28 }{
29   \RequirePackage { 13keys2e }
30   \ProcessKeysOptions { adorn }
31 }
32 %%%%%%%%%%%
33 \ExplSyntaxOff

```

`\adorn@style`

```

34 \DeclareRobustCommand{\adorn@style}{%% do NOT break line below!

```

```
35 \not@math@alphabet\adorn@style\relax
36 \fontencoding{U}\fontfamily{OrnamentsADF}\fontseries{m}\fontshape{n}\selectfont
37 }
```

`\adorn`

```
38 \newcommand*\adorn[1]{\adorn@style\char#1}
```

`\alfleftarrowhead`

```
39 \newcommand*\adfhalfleftarrowhead{\adorn{1}}
```

`\alfrightarrowhead`

```
40 \newcommand*\adfhalfrightarrowhead{\adorn{2}}
```

`\adfast`

```
41 \gdef\adfast#1{%
42 \ifcase #1 \relax
43 \or \adorn{3}%
44 \or \adorn{4}%
45 \or \adorn{5}%
46 \or \adorn{6}%
47 \or \adorn{7}%
48 \or \adorn{8}%
49 \or \adorn{9}%
50 \or \adorn{10}%
51 \or \adorn{11}%
52 \or \adorn{12}%
53 \fi}
```

`\adfhalfleafleft`

```
54 \newcommand*\adfhalfleafleft{\adorn{13}}
```

`\adfdiamond`

```
55 \newcommand*\adfdiamond{\adorn{14}}
```

`\adfhalfleafright`

```
56 \newcommand*\adfhalfleafright{\adorn{15}}
```

`\adfwavesleft`

```
57 \newcommand*\adfwavesleft{\adorn{16}}
```

openflourishleft

58 \newcommand\*{\adfopenflourishleft}{\adorn{17}}

osedflourishleft

59 \newcommand\*{\adfclosedflourishleft}{\adorn{18}}

oubleflourishleft

60 \newcommand\*{\adfdoubleflourishleft}{\adorn{19}}

\adfflourishleft

61 \newcommand\*{\adfflourishleft}{\adorn{20}}

ourishlefthdouble

62 \newcommand\*{\adfflourishlefthdouble}{\adorn{21}}

ngleflourishleft

63 \newcommand\*{\adfsingleflourishleft}{\adorn{22}}

harpflourishleft

64 \newcommand\*{\adfsharpflourishleft}{\adorn{23}}

harpflourishleft

65 \newcommand\*{\adfdoublesharpflourishleft}{\adorn{24}}

ckleflourishleft

66 \newcommand\*{\adfsickleflourishleft}{\adorn{25}}

ipleflourishleft

67 \newcommand\*{\adftripleflourishleft}{\adorn{26}}

ownhalfleafright

68 \newcommand\*{\adfflatdownhalfleafright}{\adorn{27}}

outlineleafright

69 \newcommand\*{\adfflatdownoutlineleafright}{\adorn{28}}



leafoutlineright

70 \newcommand\*{\adfflatleafoutlineright}{\adfor{29}}

outlineleafright

71 \newcommand\*{\adfoutlineleafright}{\adfor{30}}

dfsolidleafright

72 \newcommand\*{\adfsolidleafright}{\adfor{31}}

\adfflowerright

73 \newcommand\*{\adfflowerright}{\adfor{32}}

downhalfleafright

74 \newcommand\*{\adfdownhalfleafright}{\adfor{33}}

hingflatleafright

75 \newcommand\*{\adfhingflatleafright}{\adfor{34}}

fhangingleafleft

76 \newcommand\*{\adfhangingleafleft}{\adfor{35}}

adfflatleafright

77 \newcommand\*{\adfflatleafright}{\adfor{36}}

atleafsolidright

78 \newcommand\*{\adfflatleafsolidright}{\adfor{37}}

\adfleafright

79 \newcommand\*{\adfleafright}{\adfor{38}}

adfdownleafright

80 \newcommand\*{\adfdownleafright}{\adfor{39}}

dfssmallleafright

81 \newcommand\*{\adfsmallleafright}{\adfor{40}}

hangingleafright

82 \newcommand\*{\adfsmallhangingleafright}{\adfor{41}}

adleftarrowhead

83 \newcommand\*{\adleftarrowhead}{\adfor{42}}

adrightarrowhead

84 \newcommand\*{\adfrightharrowhead}{\adfor{43}}

\adwavesright

85 \newcommand\*{\adwavesright}{\adfor{44}}

openflourishright

86 \newcommand\*{\adfopenflourishright}{\adfor{45}}

closedflourishright

87 \newcommand\*{\adfclosedflourishright}{\adfor{46}}

doubleflourishright

88 \newcommand\*{\adfdoubleflourishright}{\adfor{47}}

adflourishright

89 \newcommand\*{\adfflourishright}{\adfor{48}}

flourishrightdouble

90 \newcommand\*{\adfflourishrightdouble}{\adfor{49}}

singleflourishright

91 \newcommand\*{\adfsingleflourishright}{\adfor{50}}

sharpflourishright

92 \newcommand\*{\adfsharpflourishright}{\adfor{51}}

doublesharpflourishright

93 \newcommand\*{\adfdoublessharpflourishright}{\adfor{52}}

kleftflourishright

94 \newcommand\*{\adfsickleftflourishright}{\adorn{53}}

pleftflourishright

95 \newcommand\*{\adftripleleftflourishright}{\adorn{54}}

downhalfleafleft

96 \newcommand\*{\adfflatdownhalfleafleft}{\adorn{55}}

noutlineleafleft

97 \newcommand\*{\adfflatdownoutlineleafleft}{\adorn{56}}

tleafoutlineleft

98 \newcommand\*{\adfflatleafoutlineleft}{\adorn{57}}

foutlineleafleft

99 \newcommand\*{\adfoutlineleafleft}{\adorn{58}}

adfsolidleafleft

100 \newcommand\*{\adfsolidleafleft}{\adorn{59}}

\adfflowerleft

101 \newcommand\*{\adfflowerleft}{\adorn{60}}

downhalfleafleft

102 \newcommand\*{\adfdownhalfleafleft}{\adorn{61}}

gingflatleafleft

103 \newcommand\*{\adfhangingleafleft}{\adorn{62}}

hangingleafright

104 \newcommand\*{\adfhangingleafright}{\adorn{63}}

\adfflatleafleft

105 \newcommand\*{\adfflatleafleft}{\adorn{64}}

latleafsolidleft

106 \newcommand\*{\adflatleafsolidleft}{\adfor{65}}

\adfleafleft

107 \newcommand\*{\adfleafleft}{\adfor{66}}

\adfdownleafleft

108 \newcommand\*{\adfdownleafleft}{\adfor{67}}

adfsmallleafleft

109 \newcommand\*{\adfsmallleafleft}{\adfor{68}}

lhangingleafleft

110 \newcommand\*{\adfsmallhangingleafleft}{\adfor{69}}

adhalfleftarrow

111 \newcommand\*{\adhalfleftarrow}{\adfor{70}}

\adfsquare

112 \newcommand\*{\adfsquare}{\adfor{71}}

dfhalfrightarrow

113 \newcommand\*{\adhalfrightarrow}{\adfor{72}}

\adfbullet

114 \newcommand\*{\adfbullet}{\adfor{73}}

115 \newcommand\*{\adfS}{\adfor{74}}

\adfgee

116 \newcommand\*{\adfgee}{\adfor{75}}

I don't know why somebody would use these fonts with a Unicode engine, but, just in case, map for that as well as pdf $\TeX$ .

Lua $\TeX$  manual page 49.

117 \ExplSyntaxOn

118 \bool\_if:nT { \sys\_if\_engine luatex\_p: }

119 {

```

120 \protected\def\pdfglyphtounicode {\pdfextension glyphtounicode }
121 }
122 \bool_if:nT { \sys_if_engine luatex_p: || \sys_if_engine pdftex_p: }
123 {

```

glyphtounicode\_seq This seems ... insane?

It would be more efficient to just set everything directly, but this is easier to set up and only read once. First, a sequence to hold glyph names.

```

124 \seq_new:N \l__adorn_glyphtounicode_seq
125 \seq_set_from_clist:Nn \l__adorn_glyphtounicode_seq
126 {
127   parenleft, %% parenleft
128   parenright, %% parenright
129   zero, %% zero
130   one, %% one
131   two, %% two
132   three, %% three
133   four, %% four
134   five, %% five
135   six, %% six
136   seven, %% seven
137   eight, %% eight
138   nine, %% nine
139   less, %% less
140   equal, %% equal
141   greater, %% greater
142   A, %% A
143   B, %% B
144   C, %% C
145   D, %% D
146   E, %% E
147   F, %% F
148   G, %% G
149   H, %% H
150   I, %% I
151   J, %% J
152   K, %% K
153   L, %% L
154   M, %% M
155   N, %% N
156   O, %% O
157   P, %% P
158   Q, %% Q
159   R, %% R
160   S, %% S
161   T, %% T
162   U, %% U

```

```

163     V, %% V
164     W, %% W
165     X, %% X
166     Y, %% Y
167     Z, %% Z
168     bracketleft, %% bracketleft
169     bracketright, %% bracketright
170     a, %% a
171     b, %% b
172     c, %% c
173     d, %% d
174     e, %% e
175     f, %% f
176     g, %% g
177     h, %% h
178     i, %% i
179     j, %% j
180     k, %% k
181     l, %% l
182     m, %% m
183     n, %% n
184     o, %% o
185     p, %% p
186     q, %% q
187     r, %% r
188     s, %% s
189     t, %% t
190     u, %% u
191     v, %% v
192     w, %% w
193     x, %% x
194     y, %% y
195     z, %% z
196     braceleft, %% braceleft
197     bar, %% bar
198     braceright, %% braceright
199     bullet, %% bullet
200     section, %% section
201     paragraph, %% paragraph
202 }

```

`\rn_tounicode_seq` A sequence to hold Unicode targets. These are, for the most part, simply arbitrary. I frankly have no idea what these should map to. I don't know whether there are corresponding Unicode points. If there are, I don't know where they are or how to find them. If I can find them, I have no idea whether to count most of these are the same symbol or which of several codepoints to choose.

```

203 \seq_new:N \l__adfor_tounicode_seq
204 \seq_set_from_clist:Nn \l__adfor_tounicode_seq

```

```

205 {
206 2B98 , %% 1 highlight left arrowhead
207 2B9A , %% 2 highlight right arrowhead
208 273F , %% 3 BLACK FLORETTE (Dingbats)
209 2748 , %% 4 heavy sparkle
210 2747 , %% 5 sparkle
211 274A , %% 6 8 teardrop-spoked propeller asterisk
212 274B , %% 7 heavy 8 teardrop-spoked propeller asterisk
213 2747 , %% 8 sparkle
214 2748 , %% 9 heavy sparkle
215 274A , %%10 8 teardrop-spoked propeller asterisk
216 273F , %%11 BLACK FLORETTE (Dingbats)
217 273F , %%12 BLACK FLORETTE (Dingbats)
218 2619 , %%13 REVERSED ROTATED FLORAL HEART BULLET
219 2B26 , %%14 White Medium Diamond? or 25C7? or 1F754?
220 2767 , %%15 rotated floral heart bullet?

```

These are not characters. No idea what to do with them.

```

221 2FFFE, %%16 <Not A Character>
222 2FFFE, %%17 <Not A Character>
223 2FFFE, %%18 <Not A Character>
224 2FFFE, %%19 <Not A Character>
225 2FFFE, %%20 <Not A Character>
226 2FFFE, %%21 <Not A Character>
227 2FFFE, %%22 <Not A Character>
228 2FFFE, %%23 <Not A Character>
229 2FFFE, %%24 <Not A Character>
230 2FFFE, %%25 <Not A Character>
231 2FFFE, %%26 <Not A Character>

```

Arbitrary, but back to characters, at least.

```

232 1F654, %%27 Turned North West Pointing Leaf
233 1F654, %%28 Turned North West Pointing Leaf white
234 1F65B, %%29 South East Pointing Vine Leaf
235 2767 , %%30 rotated floral heart bullet? white
236 2767 , %%31 rotated floral heart bullet? black
237 1F660, %%32 NORTH WEST POINTING BUD
238 2766 , %%33 FLORAL HEART (Dingbats)
239 2767 , %%34 rotated floral heart bullet?
240 2766 , %%35 FLORAL HEART (Dingbats)
241 2767 , %%36 rotated floral heart bullet?
242 2767 , %%37 rotated floral heart bullet?
243 2767 , %%38 rotated floral heart bullet?
244 2766 , %%39 FLORAL HEART (Dingbats)?
245 2767 , %%40 rotated floral heart bullet?
246 2767 , %%41 rotated floral heart bullet?
247 2B9C , %%42 Black Leftwards Equilateral Arrowhead
248 2B9E , %%43 black rightwards equilateral arrowhead

```

Not characters. See above.

```

249    2FFFF, %%44 <Not A Character>
250    2FFFF, %%45 <Not A Character>
251    2FFFF, %%46 <Not A Character>
252    2FFFF, %%47 <Not A Character>
253    2FFFF, %%48 <Not A Character>
254    2FFFF, %%49 <Not A Character>
255    2FFFF, %%50 <Not A Character>
256    2FFFF, %%51 <Not A Character>
257    2FFFF, %%52 <Not A Character>
258    2FFFF, %%53 <Not A Character>
259    2FFFF, %%54 <Not A Character>

```

Arbitrary, but back to characters, at least.

```

260    1F651, %%55 southwest pointing leaf
261    1F651, %%56 southwest pointing leaf
262    1F658, %%57 North West Pointing Vine Leaf white
263    2619 , %%58 REVERSED ROTATED FLORAL HEART BULLET white
264    2619 , %%59 REVERSED ROTATED FLORAL HEART BULLET
265    1F662, %%60 NORTH EAST POINTING BUD
266    2766 , %%61 FLORAL HEART (Dingbats)
267    2619 , %%62 REVERSED ROTATED FLORAL HEART BULLET
268    2766 , %%63 FLORAL HEART (Dingbats)
269    2619 , %%64 REVERSED ROTATED FLORAL HEART BULLET
270    2619 , %%65 REVERSED ROTATED FLORAL HEART BULLET
271    2619 , %%66 REVERSED ROTATED FLORAL HEART BULLET
272    2619 , %%67 REVERSED ROTATED FLORAL HEART BULLET
273    2619 , %%68 REVERSED ROTATED FLORAL HEART BULLET
274    2619 , %%69 REVERSED ROTATED FLORAL HEART BULLET
275    1F66C, %%70 Leftwards Rocket
276    274F , %%71 shadowed square
277    1F66E, %%72 Rightwards Rocket
278    2022 , %%73 bullet
279    00A7 , %%74 section
280    2761 , %%75 curved stem paragraph ornament?
281    }

```

orn\_tounicode:mn TFM-specific mapping.

pdfTEX manual page 33.

```

282    \cs_new_nopar:Npn \__adfor_tounicode:mn #1#2
283    {
284    \pdfglyphtounicode { tfm:OrnamentsADF/#1 } { #2 }
285    }

```

Generate the actual mappings.



```

286 \seq_map_pairwise_function:NNN \l__adorn_glyphtounicode_seq
287 \l__adorn_tounicode_seq \__adorn_tounicode:nn
288 }
289 \ExplSyntaxOff

290 %% end adorn.sty

```

The remaining files are not used directly, but are required to generate the files which allow  $\text{T}_{\text{E}}\text{X}$  and  $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$  to use the fonts. The sources use `fontinst` as explained in the (sparse) comments. While you can install these files into a  $\text{T}_{\text{E}}\text{X}$  tree, they are not required for typesetting.

## §A.2 Driver

The file does all the initial setup of the fonts. It organises the fonts into families, defines shapes and reencodes as required.

```

291 /OrnamentsADFEencoding [
292 /space
293 /parenleft
294 /parenright
295 /zero
296 /one
297 /two
298 /three
299 /four
300 /five
301 /six
302 /seven
303 /eight
304 /nine
305 /less
306 /equal
307 /greater
308 /A
309 /B
310 /C
311 /D
312 /E
313 /F
314 /G
315 /H
316 /I
317 /J
318 /K
319 /L
320 /M
321 /N

```

322 /O  
323 /P  
324 /Q  
325 /R  
326 /S  
327 /T  
328 /U  
329 /V  
330 /W  
331 /X  
332 /Y  
333 /Z  
334 /bracketleft  
335 /bracketright  
336 /a  
337 /b  
338 /c  
339 /d  
340 /e  
341 /f  
342 /g  
343 /h  
344 /i  
345 /j  
346 /k  
347 /l  
348 /m  
349 /n  
350 /o  
351 /p  
352 /q  
353 /r  
354 /s  
355 /t  
356 /u  
357 /v  
358 /w  
359 /x  
360 /y  
361 /z  
362 /braceleft  
363 /bar  
364 /braceright  
365 /bullet  
366 /section  
367 /paragraph  
368 /.notdef  
369 /.notdef  
370 /.notdef

371 /.notdef  
372 /.notdef  
373 /.notdef  
374 /.notdef  
375 /.notdef  
376 /.notdef  
377 /.notdef  
378 /.notdef  
379 /.notdef  
380 /.notdef  
381 /.notdef  
382 /.notdef  
383 /.notdef  
384 /.notdef  
385 /.notdef  
386 /.notdef  
387 /.notdef  
388 /.notdef  
389 /.notdef  
390 /.notdef  
391 /.notdef  
392 /.notdef  
393 /.notdef  
394 /.notdef  
395 /.notdef  
396 /.notdef  
397 /.notdef  
398 /.notdef  
399 /.notdef  
400 /.notdef  
401 /.notdef  
402 /.notdef  
403 /.notdef  
404 /.notdef  
405 /.notdef  
406 /.notdef  
407 /.notdef  
408 /.notdef  
409 /.notdef  
410 /.notdef  
411 /.notdef  
412 /.notdef  
413 /.notdef  
414 /.notdef  
415 /.notdef  
416 /.notdef  
417 /.notdef  
418 /.notdef  
419 /.notdef

420 /.notdef  
421 /.notdef  
422 /.notdef  
423 /.notdef  
424 /.notdef  
425 /.notdef  
426 /.notdef  
427 /.notdef  
428 /.notdef  
429 /.notdef  
430 /.notdef  
431 /.notdef  
432 /.notdef  
433 /.notdef  
434 /.notdef  
435 /.notdef  
436 /.notdef  
437 /.notdef  
438 /.notdef  
439 /.notdef  
440 /.notdef  
441 /.notdef  
442 /.notdef  
443 /.notdef  
444 /.notdef  
445 /.notdef  
446 /.notdef  
447 /.notdef  
448 /.notdef  
449 /.notdef  
450 /.notdef  
451 /.notdef  
452 /.notdef  
453 /.notdef  
454 /.notdef  
455 /.notdef  
456 /.notdef  
457 /.notdef  
458 /.notdef  
459 /.notdef  
460 /.notdef  
461 /.notdef  
462 /.notdef  
463 /.notdef  
464 /.notdef  
465 /.notdef  
466 /.notdef  
467 /.notdef  
468 /.notdef

469 /.notdef  
470 /.notdef  
471 /.notdef  
472 /.notdef  
473 /.notdef  
474 /.notdef  
475 /.notdef  
476 /.notdef  
477 /.notdef  
478 /.notdef  
479 /.notdef  
480 /.notdef  
481 /.notdef  
482 /.notdef  
483 /.notdef  
484 /.notdef  
485 /.notdef  
486 /.notdef  
487 /.notdef  
488 /.notdef  
489 /.notdef  
490 /.notdef  
491 /.notdef  
492 /.notdef  
493 /.notdef  
494 /.notdef  
495 /.notdef  
496 /.notdef  
497 /.notdef  
498 /.notdef  
499 /.notdef  
500 /.notdef  
501 /.notdef  
502 /.notdef  
503 /.notdef  
504 /.notdef  
505 /.notdef  
506 /.notdef  
507 /.notdef  
508 /.notdef  
509 /.notdef  
510 /.notdef  
511 /.notdef  
512 /.notdef  
513 /.notdef  
514 /.notdef  
515 /.notdef  
516 /.notdef  
517 /.notdef

```

518 /.notdef
519 /.notdef
520 /.notdef
521 /.notdef
522 /.notdef
523 /.notdef
524 /.notdef
525 /.notdef
526 /.notdef
527 /.notdef
528 /.notdef
529 /.notdef
530 /.notdef
531 /.notdef
532 /.notdef
533 /.notdef
534 /.notdef
535 /.notdef
536 /.notdef
537 /.notdef
538 /.notdef
539 /.notdef
540 /.notdef
541 /.notdef
542 /.notdef
543 /.notdef
544 /.notdef
545 /.notdef
546 /.notdef
547 /.notdef
548 ] def

```

### §A.3 Font Definitions

`ornementsadf.fd` (*fd.*) Font declarations for OrnamentsADF font

```

549 \ProvidesFile{uornements.fd}[v1.2 2024/09/29 font definitions for U/OrnamentsADF.]
addaswyd o t1phv.fd (dyddiad y ffeil fd: 2020-03-25)
550 \expandafter\ifx\csname adfor@scale\endcsname\relax
551 \let\adfor@@scale@empty
552 \else
553 \edef\adfor@@scale{s*[\csname adfor@scale\endcsname]}%
554 \fi
555 \DeclareFontFamily{U}{OrnamentsADF}{}
556 \DeclareFontShape{U}{OrnamentsADF}{m}{n}{
557 <-> \adfor@@scale OrnamentsADF
558 }{}

```

```

559 \DeclareFontShape{U}{OrnamentsADF}{m}{sc}{<->ssub * OrnamentsADF/m/n}{}
560 \DeclareFontShape{U}{OrnamentsADF}{m}{it}{<->ssub * OrnamentsADF/m/sc}{}
561 \DeclareFontShape{U}{OrnamentsADF}{m}{sl}{<->ssub * OrnamentsADF/m/it}{}
562 \DeclareFontShape{U}{OrnamentsADF}{m}{si}{<->ssub * OrnamentsADF/m/sl}{}
563 \DeclareFontShape{U}{OrnamentsADF}{m}{scit}{<->ssub * OrnamentsADF/m/si}{}
564 \DeclareFontShape{U}{OrnamentsADF}{m}{scsl}{<->ssub * OrnamentsADF/m/scit}{}
565 \DeclareFontShape{U}{OrnamentsADF}{b}{n}{<->ssub * OrnamentsADF/m/scsl}{}
566 \DeclareFontShape{U}{OrnamentsADF}{b}{sc}{<->ssub * OrnamentsADF/b/n}{}
567 \DeclareFontShape{U}{OrnamentsADF}{b}{it}{<->ssub * OrnamentsADF/b/sc}{}
568 \DeclareFontShape{U}{OrnamentsADF}{b}{sl}{<->ssub * OrnamentsADF/b/it}{}
569 \DeclareFontShape{U}{OrnamentsADF}{b}{si}{<->ssub * OrnamentsADF/b/sl}{}
570 \DeclareFontShape{U}{OrnamentsADF}{b}{scit}{<->ssub * OrnamentsADF/b/si}{}
571 \DeclareFontShape{U}{OrnamentsADF}{b}{scsl}{<->ssub * OrnamentsADF/b/scit}{}
572 \DeclareFontShape{U}{OrnamentsADF}{bx}{n}{<->ssub * OrnamentsADF/b/scsl}{}
573 \DeclareFontShape{U}{OrnamentsADF}{bx}{sc}{<->ssub * OrnamentsADF/bx/n}{}
574 \DeclareFontShape{U}{OrnamentsADF}{bx}{it}{<->ssub * OrnamentsADF/bx/sc}{}
575 \DeclareFontShape{U}{OrnamentsADF}{bx}{sl}{<->ssub * OrnamentsADF/bx/it}{}
576 \DeclareFontShape{U}{OrnamentsADF}{bx}{si}{<->ssub * OrnamentsADF/bx/sl}{}
577 \DeclareFontShape{U}{OrnamentsADF}{bx}{scit}{<->ssub * OrnamentsADF/bx/si}{}
578 \DeclareFontShape{U}{OrnamentsADF}{bx}{scsl}{<->ssub * OrnamentsADF/bx/scit}{}

```

## Change History

v1.2	scaling. ....	22
General: Belated update for (New)	v1.3	
NFSS and revised nfssect-cfr. Try	General: Add /ToUnicode values	
switching to DTX/INS. ....	(adfor). ....	12
\adfor: Remove pifont dependency. . .	v??	7
uornementsadf.fd: Support for	General: First public release. ....	1

## Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

Symbols	A	
\@adfor@digonnewfalse <i>10</i>	\adfast ..... <u>41</u>	\adfdoubleflourishright ..... <u>88</u>
\@adfor@digonnewtrue . <i>12</i>	\adfbullet ..... <u>114</u>	\adfdoublesharppflourishleft ..... <u>65</u>
\@empty ..... <i>18, 21, 551</i>	\adfclosedflourishleft <u>59</u>	\adfdoublesharppflourishright ..... <u>93</u>
\@ifl@t@r ..... <i>24</i>	\adfclosedflourishright ..... <u>87</u>	\adfdownhalfleafleft . <u>102</u>
\@ifundefined ..... <i>6</i>	\adfdiamond ..... <u>55</u>	\adfdownhalfleafright <u>74</u>
\__adfor_tounicode:nn ..... <u>282, 287</u>	\adfdoubleflourishleft <u>60</u>	

<code>\adfdownleafleft</code> . . . . .	<a href="#">108</a>	59, 60, 61, 62, 63,			
<code>\adfdownleafright</code> . . . . .	<a href="#">80</a>	64, 65, 66, 67, 68,			
<code>\adfflatdownhalfleafleft</code> .....	<a href="#">96</a>	69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79,			
<code>\adfflatdownhalfleafright</code> .....	<a href="#">68</a>	80, 81, 82, 83, 84, 85, 86, 87, 88, 89,			
<code>\adfflatdownoutlineleafleft</code> .....	<a href="#">97</a>	90, 91, 92, 93, 94, 95, 96, 97, 98, 99,			
<code>\adfflatdownoutlineleafright</code> .....	<a href="#">69</a>	100, 101, 102, 103, 104, 105, 106, 107,			
<code>\adfflatleafleft</code> . . . . .	<a href="#">105</a>	108, 109, 110, 111,			
<code>\adfflatleafoutlineleft</code> .....	<a href="#">98</a>	112, 113, 114, 115, 116			
<code>\adfflatleafoutlineright</code> .....	<a href="#">70</a>	adfor (pkg.) . . . . .	<a href="#">1</a>		
<code>\adfflatleafright</code> . . . . .	<a href="#">77</a>	<code>\adfor@@scale</code> . . . . .	<a href="#">551, 553, 557</a>		
<code>\adfflatleafsolidleft</code> <a href="#">106</a>		<code>\adfor@scale</code> . . . . .	<a href="#">17, 21</a>		
<code>\adfflatleafsolidright</code> <a href="#">78</a>		<code>\adfor@style</code> . . . . .	<a href="#">34, 38</a>		
<code>\adfflourishleft</code> . . . . .	<a href="#">61</a>	<code>\adfoutlineleafleft</code> . . . . .	<a href="#">99</a>		
<code>\adfflourishlefthdouble</code> <a href="#">62</a>		<code>\adfoutlineleafright</code> . . . . .	<a href="#">71</a>		
<code>\adfflourishright</code> . . . . .	<a href="#">89</a>	<code>\adfrightarrowhead</code> . . . . .	<a href="#">84</a>		
<code>\adfflourishrighthdouble</code> .....	<a href="#">90</a>	<code>\adfS</code> . . . . .	<a href="#">115</a>		
<code>\adfflowerleft</code> . . . . .	<a href="#">101</a>	<code>\adfsharpflourishleft</code> <a href="#">64</a>			
<code>\adfflowerright</code> . . . . .	<a href="#">73</a>	<code>\adfsharpflourishright</code> <a href="#">92</a>			
<code>\adfgee</code> . . . . .	<a href="#">116</a>	<code>\adfsickleflourishleft</code> <a href="#">66</a>			
<code>\adfhalfleafleft</code> . . . . .	<a href="#">54</a>	<code>\adfsickleflourishright</code> .....	<a href="#">94</a>		
<code>\adfhalfleafright</code> . . . . .	<a href="#">56</a>	<code>\adfsingleflourishleft</code> <a href="#">63</a>			
<code>\adfhalfleftarrow</code> . . . . .	<a href="#">111</a>	<code>\adfsingleflourishright</code> .....	<a href="#">91</a>		
<code>\adfhalfleftarrowhead</code> <a href="#">39</a>		<code>\adfsmlhangingleafleft</code> .....	<a href="#">110</a>		
<code>\adfhalfrightarrow</code> . . . . .	<a href="#">113</a>	<code>\adfsmlhangingleafright</code> .....	<a href="#">82</a>		
<code>\adfhalfrightarrowhead</code> <a href="#">40</a>		<code>\adfsmlleafleft</code> . . . . .	<a href="#">109</a>		
<code>\adfhangingleafleft</code> .....	<a href="#">103</a>	<code>\adfsmlleafright</code> . . . . .	<a href="#">81</a>		
<code>\adfhangingleafright</code> .....	<a href="#">75</a>	<code>\adfsolidleafleft</code> . . . . .	<a href="#">100</a>		
<code>\adfhangingleafleft</code> . . . . .	<a href="#">76</a>	<code>\adfsolidleafright</code> . . . . .	<a href="#">72</a>		
<code>\adfhangingleafright</code> . . . . .	<a href="#">104</a>	<code>\adfsquare</code> . . . . .	<a href="#">112</a>		
<code>\adfleafleft</code> . . . . .	<a href="#">107</a>	<code>\adftripleflourishleft</code> <a href="#">67</a>			
<code>\adfleafright</code> . . . . .	<a href="#">79</a>	<code>\adftripleflourishright</code> .....	<a href="#">95</a>		
<code>\adfleftarrowhead</code> . . . . .	<a href="#">83</a>	<code>\adfwavesleft</code> . . . . .	<a href="#">57</a>		
<code>\adfopenflourishleft</code> . . . . .	<a href="#">58</a>	<code>\adfwavesright</code> . . . . .	<a href="#">85</a>		
<code>\adfopenflourishright</code> <a href="#">86</a>					
<code>\adfor</code> . . . . .	<a href="#">38, 39, 40,</a> <a href="#">43, 44, 45, 46, 47,</a> <a href="#">48, 49, 50, 51, 52,</a> <a href="#">54, 55, 56, 57, 58,</a>				
		<b>B</b>			
		<code>\bool_if:nT</code> . . . . .	<a href="#">118, 122</a>		
				<b>C</b>	
				<code>\char</code> . . . . .	<a href="#">38</a>
				<code>\cs_new_nopar:Npn</code> . . . . .	<a href="#">282</a>
				<code>\csname</code> . . . . .	<a href="#">550, 553</a>
				<b>D</b>	
				<code>\DeclareFontFamily</code> . . . . .	<a href="#">555</a>
				<code>\DeclareFontShape</code> . . . . .	<a href="#">556, 559, 560, 561,</a> <a href="#">562, 563, 564, 565,</a> <a href="#">566, 567, 568, 569,</a> <a href="#">570, 571, 572, 573,</a> <a href="#">574, 575, 576, 577, 578</a>
				<code>\DeclareRobustCommand</code> <a href="#">34</a>	
				<code>\def</code> . . . . .	<a href="#">120</a>
				<b>E</b>	
				<code>\edef</code> . . . . .	<a href="#">553</a>
				<code>\else</code> . . . . .	<a href="#">20, 552</a>
				<code>\endcsname</code> . . . . .	<a href="#">550, 553</a>
				<code>\expandafter</code> . . . . .	<a href="#">550</a>
				<b>F</b>	
				<code>\fi</code> . . . . .	<a href="#">22, 53, 554</a>
				<code>\fmtversion</code> . . . . .	<a href="#">24</a>
				font definitions:	
				<u>uornementsadf.fd</u> <a href="#">549</a>	
				<code>\fontencoding</code> . . . . .	<a href="#">36</a>
				<code>\fontfamily</code> . . . . .	<a href="#">36</a>
				<code>\fontseries</code> . . . . .	<a href="#">36</a>
				<code>\fontshape</code> . . . . .	<a href="#">36</a>
				<b>G</b>	
				<code>\gdef</code> . . . . .	<a href="#">41</a>
				<b>I</b>	
				<code>\if@adfor@digonnew</code> <a href="#">5, 13</a>	
				<code>\ifcase</code> . . . . .	<a href="#">42</a>
				<code>\IfFileExists</code> . . . . .	<a href="#">7</a>
				<code>\IfFormatAtLeastTF</code> <a href="#">24, 25</a>	
				<code>\ifx</code> . . . . .	<a href="#">550</a>
				<b>K</b>	
				<code>\keys_define:nm</code> . . . . .	<a href="#">15</a>
				<b>L</b>	
				<code>\l__adfor_glyphfontunicode_seq</code> .....	<a href="#">124, 286</a>
				<code>\l__adfor_tfontunicode_seq</code> .....	<a href="#">203, 287</a>



<code>\let</code> .....	21, 551	<code>\not@math@alphabet</code> ..	35	<b>R</b>
				<code>\relax</code> .....
				35, 42, 550
		<b>N</b>	<b>O</b>	<b>S</b>
<code>\newcommand</code>	38, 39, 40,	options:		scale (opt.) .....
	54, 55, 56, 57, 58,	scale .....	2	2
	59, 60, 61, 62, 63,	<code>\or</code> .....	43, 44, 45, 46,	<code>\selectfont</code> .....
	64, 65, 66, 67, 68,		47, 48, 49, 50, 51, 52	36
	69, 70, 71, 72, 73,			<code>\seq_map_pairwise_function:NNN</code>
	74, 75, 76, 77, 78, 79,	<b>P</b>		.....
	80, 81, 82, 83, 84,	packages:		286
	85, 86, 87, 88, 89,	adorn .....	1	<code>\seq_new:N</code> .....
	90, 91, 92, 93, 94,	<code>\pdfextension</code> .....	120	124, 203
	95, 96, 97, 98, 99,	<code>\pdfglyphtounicode</code> ..		<code>\seq_set_from_clist:Nn</code>
	100, 101, 102, 103,	.....	120, 284	.....
	104, 105, 106, 107,	<code>\ProcessKeysOptions</code> .	30	<code>\sys_if_engine_luatex_p:</code>
	108, 109, 110, 111,	<code>\protected</code> .....	120	.....
	112, 113, 114, 115, 116	<code>\providecommand</code> .....	24	118, 122
<code>\newif</code> .....	5	<code>\ProvidesFile</code> .....	549	<code>\sys_if_engine_pdftex_p:</code>
				.....
				122
				<b>U</b>
				uornementsadf.fd (fd.)
				549