

The Pacioli fonts*

Peter Wilson
Catholic University of America
Now at `peter.r.wilson@boeing.com`

2001/01/02

Abstract

The `pacioli` package provides a set of uppercase fonts designed by Fra Luca de Pacioli in 1497.

Contents

1	Introduction	1
2	The <code>pacioli</code> package	2
3	The Metafont code	3
3.1	The parameter file	3
3.2	The driver file	5
3.3	The majuscule code	10
3.4	The punctuation file	31
3.5	The ligatures and dashes file	35
3.6	Extra punctuation	37
4	The font definition files	38
5	The <code>pacioli</code> package code	39

1 Introduction

Around the beginning of the 16th century several authors published designs for uppercase characters. The more well known among these were Albrecht Dürer who wrote *On the Just Shaping of Letters* which was published in Nuremberg in 1525, and Geoffroy Tory who wrote *Champ Fleury* which was published in Paris in 1529. Fra Luca Bartolomeo de Pacioli preceeded both of these by developing

*This file has version number v1.0a, last revised 2001/01/02.

his recipe for uppercase characters in 1497 and which was published in *De Divina Proportione* at Venice in 1509. Stanley Morison [Mor94] provides a more accessible source for Pacioli's characters.

This manual is typeset according to the conventions of the L^AT_EX DOC-STRIP utility which enables the automatic extraction of the L^AT_EX macro source files [GMS94].

Section 2 describes the usage of the package. Commented code for the fonts is in Sections 3 and 4 and source code for the package is in Section 5.

2 The pacioli package

Pacioli gave instructions for most of the uppercase letters in the alphabet, except for J, U, W and Z. All the designs were based on a square within which the letters were fitted. The width of the thickest limbs are 1/9 of the size of the square. Thinner limbs were normally half the width of the thick limbs and occasionally, as in the cross-bar of the A, were 1/3 of the width.

Albrecht Dürer's characters were similar to Pacioli's, and I have used a modified version of Alan Hoenig's rendition of the Dürer font for the missing characters.¹

I have also added modified versions of Knuth's Computer Modern punctuation characters so that, perhaps, the Pacioli font could be used as a titling font.

Pacioli's font as provided here consists of the uppercase letters, the punctuation marks !,;,:'?', the hyphen together with the en- and em-dash, and the alphabetic characters ()[]&. All the characters are accessed by typing them as you usually do in L^AT_EX.

Pacioli's fonts are designed using ruler and compass, but 'real' font design is more subtle than this. Not only do the characters have to be well-formed and consistent, but they must also appear to be consistent. Because of optical illusions, certain characters, like a C or an O, may look smaller than they actually are when used in conjunction with other characters, like an A or an M. Therefore the Cs and Os are often enlarged slightly in a vertical direction. I have made these optical adjustments to the characters obtained when using uppercase input. Alternative non-adjusted characters are also provided which can be accessed by inputting lowercase characters. These are the C, G, O and Q characters; typing c, g, o or q will give the non-adjusted uppercase versions.

The Pacioli font, which I have called `cpc` (for Computer PaCioli), is provided in an upright and a slanted version only.

`\cpcfamily` This `pacioli` package command will switch any following text into the Pacioli font family.

`\textcpc` The `pacioli` package command `\textcpc{<text>}` will typeset its `<text>` argument using the Pacioli font family.

¹ Alan Hoenig's Dürer font is available from CTAN in the `duerer` subdirectory.

3 The Metafont code

3.1 The parameter file

We deal with the two parameter files first, and start by announcing what they are for.

```

1 <*up | sl>
2 <up> %%% CPCR10.MF Computer Pacioli Roman at 10 point design size.
3 <sl> %%% CPCSL10.MF Computer Pacioli Slanted at 10 point design size.
4 %

```

Parameters from CMR10.MF are used as much as possible. We also make sure that cmbase is loaded as well as plain Metafont.

```

5 if unknown cmbase: input cmbase fi
6 %
7 <up>font_identifier="CPCR"; font_size 10pt#;
8 <sl>font_identifier="CPCSL"; font_size 10pt#;
9 %

```

The majority of the parameters and values are the same for both fonts.

```

u
  hstretch 10 u#:=20/36pt#;          % unit width
  vstretch 11 hstretch:=1;          % horizontal stretching factors; in lieu of u
  width-adj 12 vstretch:=1;         % vertical stretching factor
  serif-fit 13 width_adj#:=0pt#;     % width adjustment for certain characters
cap-serif-fit 14 serif_fit#:=0pt#;   % extra sidebar near lowercase serifs
letter-fit 15 cap_serif_fit#:=5/36pt#; % extra sidebar near uppercase serifs
16 letter_fit#:=0pt#;               % extra space added to all sidebars

body-height
  asc-height 17 body_height#:=270/36pt#; % height of tallest characters
  cap-height 18 asc_height#:=250/36pt#;  % height of lowercase ascenders
  fig-height 19 cap_height#:=246/36pt#;  % height of caps
  x-height 20 fig_height#:=232/36pt#;    % height of numerals
  math-axis 21 x_height#:=155/36pt#;     % height of lowercase without ascenders
  bar-height 22 math_axis#:=90/36pt#;    % axis of symmetry for math symbols
  comma-depth 23 bar_height#:=87/36pt#;  % height of crossbar in lowercase e
  desc-depth 24 comma_depth#:=70/36pt#;  % depth of comma below baseline
  half-height 25 desc_depth#:=70/36pt#;  % depth of lowercase descenders
26 half_height#:=1/2cap_height#; % half the height of caps
27 %

side
  thickfudge 28 side#:=cap_height#; % Height and width of the square
mediumfudge 29 thickfudge:=1/9; mediumfudge:=1/2; thinfudge:=1/3;
  thinfudge 30 thick#:=thickfudge*side#; % thickness of all thick limbs in Roman font
  thick 31 medium#:=mediumfudge*thick#; % width of medium limbs
  medium 32 thin#:=thinfudge*thick#; % thinness of all thin limbs
  thin 33 %

```

```

eta
serif-darkness 34 eta:=2-sqrt2;          % darkness for serifs; $\eta \approx 0.58$
crisp          35 serif_darkness:=eta;
tiny          36 crisp#:=0pt#;           % diameter of serif corners
fine          37 tiny#:=0pt#;           % diameter of rounded corners
rule-thickness 38 fine#:=7/36pt#;        % diameter of sharply rounded corners
              39 rule_thickness#:=.4pt#; % thickness of lines in math symbols
              40 %

cap-stem
  stem        41 cap_stem#:=thick#;      % thickness of majuscule stems
  dish        42 stem#:=thick#;          % thickness of minuscule stems
  bracket     43 dish#:=1/36pt#;         % amount erased at top or bottom of serifs
  jut         44 bracket#:=thick#;       % vertical distance from serif base to tangent
  cap-jut     45 jut#:=thick#;           % protrusion of lowercase serifs
  beak-jut    46 cap_jut#:=2/3thick#;     % protrusion of uppercase serifs for I
  slab        47 beak_jut#:=10/36pt#;    % horizontal protrusion of beak serifs
  dslab       48 slab#:=eps*1pt#;        % serif and arm thickness
              49 dslab#=0;               %slab value used in Pacioli fonts
              50 %

Hrad          Pacioli uses several different radii for serifs, with cap_jut being normal.
Lrad          51 Hrad#:=2thick#;         % Huge
lrad          52 Lrad#:=3/2thick#;       % Large
srad          53 lrad#:=thick#;         % large
trad          54 srad#:=1/2thick#;       % small
              55 trad#:=3/8thick#;       % tiny
              56 %

hair          hair is the lowercase hairline breadth, and vair is the vertical diameter of hair-
vair          lines.
              57 hair#:=vair#:=thin#;
              58 %

o
apex-o        59 o#:=8/36pt#;            % amount of overshoot for curves
apex-corr     60 apex_o#:=thin#;         % amount of overshoot for diagonal junctions
              61 apex_corr#:=0pt#;        % fine tuning of overshoot
              62 %

slant         slant is the amount the font slopes to the right. It is different for the two fonts.
              63 <up>slant:=0;             % tilt ratio $(\Delta x/\Delta y)$
              64 <sl>slant:=1/6;          % tilt ratio $(\Delta x/\Delta y)$

fudge
superness     65 %
superpull     66 fudge:=1;              % factor applied to weights of heavy characters
beak-darkness 67 superness:=1/sqrt2;     % parameter for superellipses
              68 superpull:=0;          % extra openness inside bowls

```

```

69 beak_darkness:=11/30;    % fraction of triangle inside beak serifs
70 %

serifs We are generating a seriffed, variable-width font.
monospace 71 serifs:=true;      % should serifs and bulbs be attached?
72 monospace:=false;    % should all characters have the same width?
73 %

dot-size These parameters are required for the punctuation glyphs.
square-dots 74 dot_size#:=38/36pt#; % size of dots
hefty 75 square_dots:=false; % should dots be square?
ligs 76 hefty:=false; % should we try hard not to be overweight?
77 lig_s:=2; % level of ligatures to be included
78

flare These parameters are required for the ampersand and question mark.
stem-corr 79 flare#:=33/36pt#; % diameter of bulbs or breadth of terminals
curve 80 stem_corr#:=1/36pt#; % small refinement of stem breadth
81 curve#:=30/36pt#; % lowercase curve breadth
82

whole-font We are not generating a complete font.
83 boolean whole_font; % try generating a complete font?
84 whole_font:=false;
85

Finally, call the driver file for the Pacioli font.
86 generate cpctitle %% switch to the driver file---Pacioli title
87
88 </up | sl>

```

3.2 The driver file

To a large extent the driver file is based on Alan Hoenig's dttitle.mf.

```

89 <*mfd>
90 % This is CPCTITLE.MF, adapted from Alan Hoenig's dttitle.mf
91 % This makes a short font.
92 % This file also contains special macros universally needed in the
93 % Pacioli font but either not in the cmbase.mf file or
94 % modifications of this.
95 %
96 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
97 %% MACRO SECTION %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
98 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
99 %

makebox This is a new version of the cmbase.mf makebox macro. Essentially, as we are only
generating an uppercase font, we don't need all the horizontals for the lowercase
characters.

```

```

100 % A new version of cmbase.mf makebox macro
101 def makebox(text rule) =
102   for y=0,half_height,cap_height,body_height,x_height,-desc_depth,-body_depth:
103     rule((l,y)t_,(r,y)t_); endfor % horizontals
104   for x=l,r: rule((x,-body_depth)t_,(x,body_height)t_); endfor % verticals
105   for x=u*(1+floor(1/u)) step u until r-1:
106     rule((x,-body_depth)t_,(x,body_height)t_); endfor % more verticals
107   if charic<>0:
108     rule((r+charic*pt,h.o_),(r+charic*pt,.5h.o_)); fi % italic correction
109 enddef;
110

```

`proofpcbb` The macro `proofpcbb(S, xoff, yoff)` draws crosses at the corners of Pacioli's square, and at the bottom and top midpoints, where the point `zS` is at position `(xoff,yoff)` in the square.

```

111
112 % A macro to delineate Pacioli's bounding square in proof mode.
113 % Point z$ is at (xoff,yoff) in the bounding square.
114 def proofpcbb(suffix $(expr xoff, yoff) =
115 if proofing > 0:
116   x$bl = x$ - xoff; y$bl = y$ - yoff; % Bottom Left Hand Corner
117   z$br = (x$bl+cap_height, y$bl); % BRHC
118   z$tr = (x$bl+cap_height, y$bl+cap_height); % TRHC
119   z$t1 = (x$bl, y$bl+cap_height); % TLHC
120   z$bc = 1/2[z$bl,z$br]; z$tc = 1/2[z$t1,z$tr]; % horizontal mid points
121   pair hc, vc;
122   hc = (2u,0); vc = (0,2u);
123   pickup pensquare scaled 2;
124   for s = z$bl, z$br, z$tr, z$t1, z$bc, z$tc:
125     draw s-hc--s+hc;
126     draw s-vc--s+vc;
127   endfor
128 fi
129 enddef;
130

```

`cm-arm` Hoenig's modification to Knuth's arm macro.

```

arm 131
132 let cm_arm=arm;
133
134 vardef arm(suffix $,$$,@)(expr darkness, armjut) = % arm from |z$| to |z$$|
135 numeric vjut_; vjut_ = abs(bot y$1-bot y$$);
136 x@0=good.x(x$$r-armjut); y@0=y$r;
137 if serifs: y@1=y$1; z@1=z$$1+whatever*(z$$r-z@0);
138   if x$$>=x$: x@2=x@1 - vjut_; else: x@2=x@1+vjut_; fi y@2=y$1;
139   filldraw z$1{z@1-z$$1}...darkness[z@1,.5[z@2,z$$1]]{z@2-z$$1}...
140   z@2if x$$>=x$: {left} else: {right} fi
141   ---z$1--z$r--z@0--z$$r..cycle; % arm and beak
142 else: filldraw z$1--z$r--z@0--z$$r--cycle; fi % sans-serif arm

```

```

143 penlabels(@0,@1,@2); enddef;
144

```

foot This is a modified version of Hoenig's foot macro to draw the bottom leg of the E and L characters.

```

145
146 vardef foot(suffix $, $$, @)(expr darkness, armjut) = % curved beak on E and L
147 numeric vjut_; vjut_ = abs(bot y$-bot y$$);
148 x@0=good.x(x$$r-armjut); y@0=y$r;
149 if serifs:
150   y@1=y$1; z@1=z$$1+whatever*(z$$r-z@0);
151   x@2=x@1-vjut_; y@2=y$1;
152   filldraw z$$1{z@1-z$$1}...darkness[z@1,.5[z@2,z$$1] ]...z@2
153   ---z$1--z$r--z@0--z$$r--cycle; % arm and beak
154 else:
155   filldraw z$1--z$r--(x$$r,y$r)--(x$$r,y$1)--cycle;
156 fi
157 penlabels(@0,@1,@2); enddef;
158

```

pulled-super-arc Hoenig's rewrite of Knuth's macros.

```

pulled-arc 159
160 vardef pulled_super_arc.l(suffix $, $$)(expr darkness)=
161 pair corner;
162 if y$=y$r: corner=(x$1, y$$1);
163 else: corner=(x$$1,y$1); fi
164 z$1{corner-z$1}...(darkness)[corner, .5[z$1, z$$1]]{z$$1-z$1}
165 ...{z$$1-corner}z$$1 enddef;
166
167 vardef pulled_super_arc.r(suffix $, $$)(expr darkness)=
168 pair corner;
169 if y$=y$r: corner=(x$r,y$$r);
170 else: corner=(x$$r, y$r); fi
171 z$r{corner-z$r}...(darkness)[corner, .5[z$r, z$$r]]{z$$r-z$r}
172 ...z$$r{z$$r-corner} enddef;
173
174 vardef pulled_arc@#(suffix $, $$)=
175 pulled_super_arc@#($,$$(eta) enddef;
176

```

cm-font-setup Hoenig's font set up.

```

slim-font-setup 177
178 let cm_font_setup=font_setup; % Knuth's original font_setup
179
180 def slim_font_setup = % trimmed font_setup for use with Pacioli caps only
181 if monospace: let adjust_fit=mono_adjust_fit;
182   def mfudged=fudged enddef;
183   mono_charic#:=body_height#*slant;
184   if mono_charic#<0: mono_charic#:=0; fi

```

```

185 mono_charwd#:=9u#; define_whole_pixels(mono_charwd);
186 else: let adjust_fit=normal_adjust_fit;
187   def mfudged= enddef; fi
188 define_pixels(u,width_adj,serif_fit,cap_serif_fit,jut,cap_jut,bar_height,
189   dish,bracket,beak_jut,apex_o,apex_corr);
190 define_whole_pixels(letter_fit,fine,crisp,tiny);
191 define_whole_vertical_pixels(body_height,asc_height,
192   cap_height,x_height,comma_depth,desc_depth);
193 define_whole_blacker_pixels(hair,stem,cap_stem);
194 define_whole_vertical_blacker_pixels(vair,dslab,slab);
195 if slab<eps: slab:=eps fi;
196 define_corrected_pixels(o);
197 forsuffices $=hair,stem,cap_stem:
198   fudged$.#:=fudge*$.#; fudged$:=hround(fudged$.#*hppp+blacker);
199   forever: exitif fudged$>.9fudge*$; fudged$:=fudged$+1; endfor endfor
200 rule_thickness:=ceiling(rule_thickness#*hppp);
201 heavy_rule_thickness:=ceiling(3rule_thickness#*hppp);
202 oo:=vround(.5o#*hppp*o_correction)+eps;
203 apex_oo:=vround(.5apex_o#*hppp*o_correction)+eps;
204 lowres_fix(stem) 1.3;
205 ess:=(ess#/stem#)*stem; cap_ess:=(cap_ess#/cap_stem#)*cap_stem;
206 dw:=(curve#-stem#)*hppp; bold:=curve#*hppp+blacker;
207 dh#:=.6designsize;
208 more_super:=max(superness,sqrt .77superness);
209 hein_super:=max(superness,sqrt .81225258superness); % that's  $2^{-.3}$ 
210 clear_pen_memory;
211 if fine=0: fine:=1; fi
212 forsuffices $=fine,crisp,tiny:
213   %% fine $ %%% temporary formatting convention for MFT
214   if $>fudged.hair: $:=fudged.hair; fi
215   $.breadth:=fudged.hair;
216   pickup if $=0: nullpen else: pencircle scaled $; $:=$-eps fi;
217   $.nib:=savepen; breadth_[$.nib]:=$;
218   forsuffices $$=lft,rt,top,bot: shiftdef($,$,$,$ 0); endfor endfor
219   %% @ $ %%% restore ordinary formatting for $
220   min_Vround:=max(fine.breadth,crisp.breadth,tiny.breadth);
221   if min_Vround<vround min_Vround: min_Vround:=vround min_Vround; fi
222   pickup pencircle scaled rule_thickness; rule.nib:=savepen;
223   currenttransform:=identity slanted slant
224   yscaled aspect_ratio scaled granularity;
225   if currenttransform=identity: let t_=relax
226   else: def t_ = transformed currenttransform enddef fi;
227   numeric paren_depth#; .5[body_height#,-paren_depth#]=math_axis#;
228   numeric asc_depth#; .5[asc_height#,-asc_depth#]=math_axis#;
229   body_depth:=desc_depth+body_height-asc_height;
230   shrink_fit:=1+hround(2letter_fit#*hppp)-2letter_fit;
231   if not string mode: if mode<=smoke: shrink_fit:=0; fi fi
232   enddef;
233

```


`pacioli-font-setup` This macro contains additional set up for the Pacioli fonts.

```
234
235 def pacioli_font_setup= % contains special stuff for Pacioli fonts
236   define_pixels(side,thick,thin,medium,half_height);
237   define_pixels(Hrad,Lrad,lrad,srad,trad);
238   define_pixels(dot_size,math_axis,flare,stem_corr,curve);
239   if not known dslab#: dslab=eps; fi
240 enddef;
241
242
243 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
244 %% END OF MACROS %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
245 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
246
```

Set the mode and font setup.

```
247
248 font_coding_scheme:="ASCII caps and punctuation";
249
250 mode_setup;
251
252 pacioli_font_setup;
253 if whole_font: font_setup; %slab:=eps;
254   define_whole_vertical_blacker_pixels(dslab,slab);
255 else: slim_font_setup; fi
256
```

We generate Pacioli's uppercase letters, together with (matching) Computer Modern Roman punctuation and some analphabetics.

```
257
258 input cpcromanu; % Pacioli upper case (majuscules)
259 input cpcpunct; % punctuation
260 input cpclig; % the several dashes
261 input cpcromanp % ampersand and ?mark
262
```

Finally, do the uppercase ligtable.

```
263
264 font_slant slant; font_x_height x_height#;
265 if monospace: font_normal_space 9u#; % no stretching or shrinking
266 font_quad 18u#;
267 font_extra_space 9u#;
268 else: font_normal_space 6u#+2letter_fit#;
269 font_normal_stretch 3u#; font_normal_shrink 2u#;
270 font_quad 18u#+4letter_fit#;
271 font_extra_space 2u#;
272 k#:=-.5u#; kk#:= -1.5u#; kkk#:= -2u#; % three degrees of kerning
273 ligtable "P": "T": "Y": "A" kern kk#;
274 ligtable "F": "V": "W": "A" kern if serifs: kkk# else: kk#fi,
275 "K": "X": "Q" kern k#, "C" kern k#, "G" kern k#, "Q" kern k#;
```

```

276 ligtable "O": "D": "X" kern k#, "W" kern k#, "A" kern k#,
277 "V" kern k#, "Y" kern k#;
278 ligtable "A": if serifs: "R": fi
279 "C" kern k#, "O" kern k#, "G" kern k#, "U" kern k#, "Q" kern k#,
280 "L": "T" kern kk#, "Y" kern kkk#, "V" kern kk#, "W" kern kkk#;
281 ligtable "I": "I" kern -k#; fi % Richard III
282
283 bye.
284
285 </mfd>

```

3.3 The majuscule code

The following code is based on Alan Hoenig's code for the Dürer characters. The geometry, of course, is different for Pacioli's characters.

```

286 <*maj>
287 % CPCROMANU.MF Program files for formal Computer Pacioli fonts.
288 % Based on Alan Hoenig's dromanu.mf for Durer fonts
289
290 def checkthin=
291 if thin-currentbreadth<eps: thin:=currentbreadth+eps; fi;
292 if thin-currentbreadth<eps: thin:=currentbreadth+eps; fi enddef;
293
294 numeric P_width#; P_width#=cap_jut#+3thick#+1/4side#-thin#;
295 define_pixels(P_width);
296
297 pair xeps, yeps; xeps=(eps,0); yeps=(0,eps);
298 numeric tail_corr; tail_corr=1/8;
299 numeric cap_jut_factor; cap_jut_factor=8/7;
300
301 numeric pcshiftx#; % x value of LHS of Pacioli box wrt origin of char box
302 define_pixels(pcshiftx);
303

```

- A The letter A. The right and left legs are **thick** and **medium** respectively, while the cross-bar is **thin**. With serifs, it just fits inside the square. The apex is slightly extended, then an arc is removed.

```

304
305 cmchar "Pacioli's letter A";
306 beginchar("A", 2u#+side#, cap_height#, 0);
307 adjust_fit(cap_serif_fit#, cap_serif_fit#);
308 pcshiftx := u;
309 top y3= top y2=h + 2cap_jut;
310 % for left leg
311 numeric alpha[]; alpha1=(h++(0.5side - cap_jut - 0.5thin))/(side);
312 penpos1(alpha1*medium,0); penpos2(alpha1*medium,0);
313 % and for the right leg
314 alpha2=(h++(0.5side-0.5thick))/(side);
315 penpos3(alpha2*thick,0); penpos4(alpha2*thick,0);

```

```

316 lft x1l=cap_jut_factor*lrاد+.5u; rt x4r=pcshiftx + side;
317 bot y1=bot y4=0 if monospace: +1/2 crisp fi;
318 x3r=x2l = 0.5side + pcshiftx;
319 z0=whatever[z1r,z2r]=whatever[z3l,z4l];
320 fill z0--diag_end(0,4l,1,1,4r,3r)--z3r..z2l--diag_end(2l,1l,1,1,1r,0)
321 --z0..cycle; % diagonals
322 top y5r=top y6r=hround(h/2); top y5r-bot y5l=top y6r-bot y6l=thin;
323 penpos5(whatever, angle(z2-z1)); penpos6(whatever, angle(z3-z4));
324 z5=whatever[z1,z2]; z6=whatever[z3,z4];
325 penstroke z5e--z6e; % bar line
326 pickup tiny.nib;
327 numeric top_bar; top_bar=top y5r-bot y6l;
328 if monospace: pickup pencircle scaled 0pt;
329 pos11(top_bar,90); pos12(top_bar,90);
330 top y11r=top y12r=h; w/2=.5[x11,x12]; x12-x11=apex_corr+2cap_jut;
331 fill z11l--z12l{right}..tension atleast.95..{left}z12r--z11r{left}..
332 tension atleast.95..{right}cycle;
333 penlabels(11,12); fi
334 if serifs: bracket:=10/7 bracket; cap_jut:=cap_jut_factor*cap_jut;
335 pickup tiny.nib; numeric jut_factor;
336 if monospace: jut_factor:=1; else: jut_factor:=3/4; fi
337 serif(1,2,a,.8serif_darkness,-cap_jut);serif(4,3,d,.8serif_darkness, cap_jut);
338 bracket:=jut_factor*bracket; if monospace: jut_factor:=.6; fi
339 serif(1,2,b,7/8 serif_darkness,jut_factor*cap_jut);
340 serif(4,3,c,7/8 serif_darkness,-jut_factor*cap_jut);
341 fi
342 % remove the apex
343 x20 = x2l - 0.5lrاد; y20 = side+lrاد;
344 erase filldraw fullcircle scaled 2lrاد shifted (x20,y20);
345 proofpcbb(1, lrاد+0.5medium, 0);
346 penlabels(0,1,2,3,4,5,6,20); endchar;
347

```

B The letter B.

```

348
349 cmchar "Pacioli's letter B";
350 beginchar("B",u#+cap_jut#+0.64side#, cap_height#, 0);
351 adjust_fit(cap_serif_fit#,0); pickup tiny.nib;
352 pcshiftx := 0.5u;
353 pos1(thick,0); pos2(thick,0); lft x1l=lft x2l=pcshiftx+cap_jut;
354 bot y2=0; top y1=h; filldraw stroke z1e--z2e; % vertical stem
355 pos3(thin,-90); pos5(thick,0); pos7(medium,90);
356 z3r=z2; y4l=y3l; x4l=x3l+thick; y4r=y3r; x4r=x4l+thick-1/2thin; z4=.5[z4l,z4r];
357 y7=h/2; x7=x1; y6l=y7l; x6l=x4l; top y6r-bot y6l=medium; x6r=x4r;
358 z6=.5[z6r,z6l];
359 x5=hround(x1+0.64side-thick); y5=0.25side;
360 filldraw stroke z3e--z4e..pulled_arc.e(4,5)&pulled_arc.e(5,6)..z6e--z7e;
361 % bottom lobe
362 pos7'(thin,-90); pos8(thick,0); pos10(medium,90);
363 z7'=z7; z6'=z6; x6'r=x4r; x6'l=x4l; y6'r=y7'r; y6'l=y7'l;

```

```

364 z10r=z1; x9r=x6'r; x9l=x6'l; y9r=y10r; top y9r-bot y9l=medium; z9=.5[z9r,z9l];
365 x8=hround(x1+0.53side-thick); y8=.75side;
366 filldraw stroke z7'e--z6'e..pulled_arc.e(6',8)&pulled_arc.e(8,9)
367 ..z9e--z10e; % upper lobe
368 if serifs: serif(1,2,a, serif_darkness,-cap_jut);
369 serif(2,1,b,serif_darkness,-cap_jut);
370 numeric tempa_, tempb_; tempa_=slab; tempb_=bracket;
371 slab:=thin; bracket:=2/3cap_jut; serif(2,1,c,serif_darkness,2/3cap_jut);
372 slab:=tempa_; bracket:=tempb_; fi
373 proofpcbb(2, cap_jut+0.5thick, 0);
374 penlabels(1,2,3,4,5,6,7,8,9,10,11,12);
375 penlabels(6',7'); endchar;
376

```

c The letter C. This is Pacioli's original definition.

```

377
378 cmchar "Pacioli's original letter C";
379 beginchar("c",2u#+.95side#, cap_height#, 0);
380 adjust_fit(0, 0); if monospace: pickup fine.nib; else: pickup tiny.nib;fi
381 pcshiftx := u;
382 numeric alpha; alpha=w-u-thick;
383 pos2(thin, 90); x2=.5[x3r,x0r]; top y2r=h;% topmost pen position
384 pos3(thick,180); lft z3r=(u,h/2);
385 pos4(thin, 270); x4=x2; bot y4r=0; % bottommost pen position
386 pos0(thick,0); lft z0l=(w+thick-u,h/2);
387 path p, p', line.u, line.b; line.u=(alpha,1.5h)--(alpha,h/2);
388 line.b=(.5[alpha,x0],0)--(.5[alpha,x0],h/2);
389 p=pulled_arc.r(0,2)&pulled_arc.r(2,3)&pulled_arc.r(3,4)
390 &pulled_arc.r(4,0)&cycle;
391 p'=pulled_arc.l(0,2)&pulled_arc.l(2,3)&pulled_arc.l(3,4)
392 &pulled_arc.l(4,0)&cycle;
393 z'1l=line.u intersectionpoint p'; z'1r=line.u intersectionpoint p;
394 bot rt z1l=z'1l; top rt z1r=z'1r; z1=.5[z1l,z1r];
395 z'5l=line.b intersectionpoint p'; z'5r=line.b intersectionpoint p;
396 z5r = (side+pcshiftx-3/4thick, 3/4thick); z5l = (x5r+o, y5r+1.5thick); z5=.5[z5r,z5l];
397 if not monospace:
398 x1l:=x1l+.6thin; y1l:=y1r-1.0thick-2tiny;
399 y5l:=2thick+y5r-2tiny; fi
400 if monospace: x1:=x1l:=x1r:=x5; y1:=y1-thin; y1l:=y1l-thin; y1r:=y1r-.6thin; fi
401 % readjust to have a better opening in CDTT
402 numeric slope_factor; if monospace:slope_factor=4/3; else:slope_factor=3/4; fi
403 filldraw
404 z1r{-1,slope_factor}...{left}z2r..pulled_arc.r(2,3)&
405 pulled_arc.r(3,4)..{right}z4r..tension 0.9..z5r--
406 reverse
407 (z1l..tension 0.9..{left}z2l..pulled_arc.l(2,3)&pulled_arc.l(3,4)..{right}z4l
408 ..tension 0.9..z5l)
409 ..(z1l+(eps,0))--(z1r+(eps,0))..cycle;
410 proofpcbb(3, 0.5thick, 0.5side);
411 penlabels(0,1,2,3,4,5); endchar;

```

412

- c The letter C. This includes an optical adjustment at the top and bottom of the character.

```

413
414 cmchar "Pacioli's letter C";
415 beginchar("C",2u#+.95side#, cap_height#, 0);
416 adjust_fit(0, 0); if monospace: pickup fine.nib; else: pickup tiny.nib;fi
417 pcshiftx := u;
418 numeric alpha; alpha=w-u-thick;
419 pos2(thin, 90); x2=.5[x3r,x0r]; top y2r=h+o;% topmost pen position
420 pos3(thick,180); lft z3r=(u,h/2);
421 pos4(thin, 270); x4=x2; bot y4r=-o; % bottommost pen position
422 pos0(thick,0); lft z0l=(w+thick-u,h/2);
423 path p, p', line.u, line.b; line.u=(alpha,1.5h)--(alpha,h/2);
424 line.b=(.5[alpha,x0],0)--(.5[alpha,x0],h/2);
425 p=pulled_arc.r(0,2)&pulled_arc.r(2,3)&pulled_arc.r(3,4)
426   &pulled_arc.r(4,0)&cycle;
427 p'=pulled_arc.l(0,2)&pulled_arc.l(2,3)&pulled_arc.l(3,4)
428   &pulled_arc.l(4,0)&cycle;
429 z'1l=line.u intersectionpoint p'; z'1r=line.u intersectionpoint p;
430 bot rt z1l=z'1l; top rt z1r=z'1r; z1=.5[z1l,z1r];
431 z'5l=line.b intersectionpoint p'; z'5r=line.b intersectionpoint p;
432 z5r = (side+pcshiftx-3/4thick, 3/4thick); z5l = (x5r+o, y5r+1.5thick); z5=.5[z5r,z5l];
433 if not monospace:
434   x1l:=x1l+.6thin; y1l:=y1r-1.0thick-2tiny;
435   y5l:=2thick+y5r-2tiny; fi
436 if monospace: x1:=x1l:=x1r:=x5; y1:=y1-thin; y1l:=y1l-thin; y1r:=y1r-.6thin; fi
437 % readjust to have a better opening in CDTT
438 numeric slope_factor; if monospace:slope_factor=4/3; else:slope_factor=3/4; fi
439 filldraw
440   z1r{-1,slope_factor}...{left}z2r..pulled_arc.r(2,3)&
441   pulled_arc.r(3,4)..{right}z4r..tension 0.9..z5r--
442   reverse
443   (z1l..tension 0.9..{left}z2l..pulled_arc.l(2,3)&pulled_arc.l(3,4)..{right}z4l
444   ..tension 0.9..z5l)
445   ..(z1l+(eps,0))--(z1r+(eps,0))..cycle;
446 proofpcbb(3, 0.5thick, 0.5side);
447 penlabels(0,1,2,3,4,5); endchar;
448

```

- D The letter D. With serifs, it fills the square.

```

449
450 cmchar "Pacioli's letter D";
451 beginchar("D", cap_jut#+.95side#+u#, cap_height#, 0);
452 adjust_fit(cap_serif_fit#,0);
453 pickup tiny.nib; pos1(cap_stem,0); pos2(cap_stem,0);
454 lft x1l=.5u+cap_jut; x2=x1; bot y2=0; top y1=h;
455 filldraw stroke z1e--z2e; % stem
456 pos3(thin, 90); pos5(thick, 0); pos7(5/4thin, -90);

```

```

457 x3=x7=x1; top y3r=h; bot y7r=0;
458 y5=good.y h/2; x5=h-3/2thick+x1;
459 if not monospace: x4r=x6r=hround(w/2); top y4r=h; bot y6r=0;
460 bot y4l=vround(h-thin); top y6l=vround 5/4thin;
461 x4l=x6l; 2(lft x5l-x4l)=y4l-y6l;
462 z4=.5[z4l, z4r]; z6=.5[z6l, z6r];
463 else: pos4(thin,90); pos6(thin,-90); x4=x6=hround w/2; top y4r=h; bot y6r=0;
464 fi
465 filldraw stroke z3e{right}...pulled_super_arc.e(4,5,eta)&
466 pulled_super_arc.e(5,6,eta)...{left}z7e; % lobe
467 if serifs: serif(1, 2, a, serif_darkness, -cap_jut);
468 serif(2, 1, b, serif_darkness, -cap_jut);
469 if not monospace: numeric temp_; temp_=slab; slab:=5/4thin;
470 serif(2, 1, c, serif_darkness, cap_jut); slab:=temp_; fi fi
471 proofpcbb(2, cap_jut+0.5thick, 0);
472 penlabels(1,2,3,4,5,6,7); endchar;
473

```

- E The letter E. It is half the width of the square, but the bottom arm is slightly longer.

```

474
475 cmchar "Pacioli's letter E";
476 numeric topbar_; topbar_=1/2; % half the square for Pacioli
477 numeric middlebar_; middlebar_=10/31; % as measured
478 beginchar("E", u#+cap_jut#+5.25thick#, cap_height#,0);
479 adjust_fit(cap_serif_fit#,0);
480 pcshiftx := 0.5u;
481 pickup tiny.nib; pos1(cap_stem,0); pos2(cap_stem,0);
482 lft x1l=lft x2l=.5u+cap_jut; top y1=h; bot y2=0;
483 filldraw stroke z1e--z2e; % stem
484 pickup crisp.nib; % prepare for arms and serifs
485 pos3(medium,90); pos4(dslab,0);
486 x1=x3; top y3r=h; rt x4l=x2l+topbar_*side;
487 bot y4=vround(h-if monospace: 2 fi vstretch*thick);
488 arm(3,4,d,serif_darkness,0); % top arm and beak
489 pos5(thin,-90); pos0(thin,90); pos6(dslab,0); pos7(dslab,0);
490 x5=x0=x1; y5=y0=hround(h/2); rt x6r=rt x7r= x1r+middlebar_*side;
491 top y6-bot y7= thin+3/4thick; y5=.5[y6,y7];
492 arm(5,6,e,eta,0); arm(0,7,f,eta,0); % middle arms and beaks
493 % now do the bottom arm
494 pos8(medium,-90); pos9(dslab,0); x8=x2; bot y8r=0; rt x9r=hround(x2l+5.25thick);
495 y9=y8r+thick;
496 foot(8,9,g,serif_darkness,0.25thick); % lower arm
497 if serifs: serif(1,2,a,serif_darkness,-cap_jut);
498 serif(2,1,b,serif_darkness,-cap_jut);
499 if not monospace: temp_:=slab; slab:=medium; tmp_:=bracket; bracket:=cap_jut+medium;
500 serif(2,1,c,serif_darkness,cap_jut);
501 slab:=temp_; bracket:=tmp_; fi fi
502 proofpcbb(2, cap_jut+0.5thick, 0);
503 penlabels(0,1,2,3,4,5,6,7,8,9); endchar; % end of "E"

```

504

F The letter F. This is an E without the bottom arm.

```

505
506 cmchar "Pacioli's letter F";
507 numeric topbar_; topbar_=1/2; % half the square for Pacioli
508 numeric middlebar_; middlebar_=10/31; % as measured
509 beginchar("F", u#+1/2side#+cap_jut#, cap_height#,0);
510 adjust_fit(cap_serif_fit#,0);
511 pcshiftx := 0.5u;
512 pickup tiny.nib; pos1(cap_stem,0); pos2(cap_stem,0);
513 lft x1l=lft x2l=.5u+cap_jut; top y1=h; bot y2=0;
514 filldraw stroke z1e--z2e; % stem
515 pickup crisp.nib; % prepare for arms and serifs
516 pos3(medium,90); pos4(dslab,0);
517 x1=x3; top y3r=h; rt x4l= x2l+topbar_*side;
518 bot y4=vround(h-if monospace: 2 fi vstretch*thick);
519 arm(3,4,d,serif_darkness,0); % top arm and beak
520 pos5(thin,-90); pos0(thin,90); pos6(dslab,0); pos7(dslab,0);
521 x5=x0=x1; y5=y0=hround(h/2); rt x6r=rt x7r= x1r+middlebar_*side;
522 top y6-bot y7= thin + 3/4thick; y5=.5[y6,y7];
523 arm(5,6,e,eta,0); arm(0,7,f,eta,0); % middle arms and beaks
524 if serifs: serif(1,2,a,serif_darkness,-cap_jut);
525 serif(2,1,b,serif_darkness,-cap_jut); serif(2,1,c,serif_darkness, cap_jut);
526 fi
527 proofpcbb(2, cap_jut+0.5thick, 0);
528 penlabels(0,1,2,3,4,5,6,7); endchar; % end of "F"
529

```

g The letter G. This is based on the C and is Pacioli's original definition.

```

530
531 cmchar "Pacioli's original letter G";
532 beginchar("g",2u#+.925side#, cap_height#, 0);
533 adjust_fit(0, cap_serif_fit#);
534 pcshiftx := u;
535 if monospace: pickup fine.nib; else: pickup tiny.nib;fi
536 % This code is identical to that for the letter C.
537 numeric alpha; alpha=w-u-thick;
538 pos2(thin, 90); x2=.5[x3r,x0r]; top y2r=h;% topmost pen position
539 pos3(thick,180); lft z3r=(u,h/2);
540 pos4(thin, 270); x4=x2; bot y4r=0;
541 pos0(thick,0); lft z0l=(w+thick-u,h/2);
542 path p, p', line.u, line.b; line.u=(alpha,1.5h)--(alpha,h/2);
543 line.b=(.5[alpha,x0],0)--(.5[alpha,x0],h/2);
544 p=pulled_arc.r(0,2)&pulled_arc.r(2,3)&pulled_arc.r(3,4)
545 &pulled_arc.r(4,0)&cycle;
546 p'=pulled_arc.l(0,2)&pulled_arc.l(2,3)&pulled_arc.l(3,4)
547 &pulled_arc.l(4,0)&cycle;
548 z'1l=line.u intersectionpoint p'; z'1r=line.u intersectionpoint p;
549 bot rt z1l=z'1l; top rt z1r=z'1r; z1=.5[z1l,z1r];

```

```

550 z'5l=line.b intersectionpoint p'; z'5r=line.b intersectionpoint p;
551 z5r = (side+pcshiftx-3/4thick, 3/4thick); z5l = (x5r+o, y5r+1.5thick); z5=.5[z5r, z5l];
552 if not monospace:
553   x1l:=x1l+.6thin; y1l:=y1r-1.0thick-2tiny;
554   y5l:=2thick+y5r-2tiny; fi
555 if monospace: x1:=x1l:=x1r:=x5; y1:=y1-thin; y1l:=y1l-thin; y1r:=y1r-.6thin; fi
556 % readjust to have a better opening in CDTT
557 numeric slope_factor; if monospace:slope_factor=4/3; else:slope_factor=3/4; fi
558 % Additional G work (preceding program is the same as that for letter C)!
559 path p,p',line.o, line.i;
560 p=z1r{-1,slope_factor}..{left}pulled_arc.r(2,3)&
561   pulled_arc.r(3,4){right}..tension 0.9..z5r;
562 p'= reverse
563   (z1l..tension 0.9..{left}pulled_arc.l(2,3)&pulled_arc.l(3,4){right}
564     ..tension 0.9..z5l);
565 pos7(thick,0);
566 top z7=(x1r,7/2thick);
567 numeric counter; counter=0;
568 forever:
569   counter:=counter+1;
570   line.o:=z7r--(x7r,-2o); line.i:=(z7r-(thick,0))--(x7r-thick,-2o);
571   numeric t[],tt[];
572   (t1,tt1)=p intersectiontimes line.o; (t2,tt2)=p' intersectiontimes line.i;
573   exitif t1*t2>0; forsuffices $$=1, r: x7$$:=x7$$-.1u; endfor
574   x7:=x7-.1u; endfor
575 x6r=x7r; x6l=x7l; bot y6r=y part point t1 of p; bot y6l=y part point t2 of p';
576 z6=.5[z6r,z6l];
577 p:= subpath(0,t1) of p..(z6r-xeps+yeps)--(z6l+xeps+yeps)..
578   subpath(t2,infinity) of p'..(z1l+xeps)--cycle;
579 filldraw p;
580 filldraw z6l--z7l--z7r--z6r--cycle;
581 if serifs: serif(7,6,a,serif_darkness,-srاد);
582   serif(7,6,b,serif_darkness, srاد); fi
583 proofpcbb(3, 0.5thick, 0.5side);
584 penlabels(0,1,2,3,4,5,6,7); endchar;
585

```

G The letter G. This is based on the C and has optical adjustments.

```

586
587 cmchar "Pacioli's letter G";
588 beginchar("G",2u#+.925side#, cap_height#, 0);
589 adjust_fit(0, cap_serif_fit#);
590 pcshiftx := u;
591 if monospace: pickup fine.nib; else: pickup tiny.nib;fi
592 % This code is identical to that for the letter C.
593 numeric alpha; alpha=w-u-thick;
594 pos2(thin, 90); x2=.5[x3r,x0r]; top y2r=h+o;% topmost pen position
595 pos3(thick,180); lft z3r=(u,h/2);
596 pos4(thin, 270); x4=x2; bot y4r=-o;
597 pos0(thick,0); lft z0l=(w+thick-u,h/2);

```



```

598 path p, p', line.u, line.b; line.u=(alpha,1.5h)--(alpha,h/2);
599 line.b=(.5[alpha,x0],0)--(.5[alpha,x0],h/2);
600 p=pulled_arc.r(0,2)&pulled_arc.r(2,3)&pulled_arc.r(3,4)
601   &pulled_arc.r(4,0)&cycle;
602 p'=pulled_arc.l(0,2)&pulled_arc.l(2,3)&pulled_arc.l(3,4)
603   &pulled_arc.l(4,0)&cycle;
604 z'1l=line.u intersectionpoint p'; z'1r=line.u intersectionpoint p;
605 bot rt z1l=z'1l; top rt z1r=z'1r; z1=.5[z1l,z1r];
606 z'5l=line.b intersectionpoint p'; z'5r=line.b intersectionpoint p;
607 z5r = (side+pcshiftx-3/4thick, 3/4thick); z5l = (x5r+o, y5r+1.5thick); z5=.5[z5r, z5l];
608 if not monospace:
609   x1l:=x1l+.6thin; y1l:=y1r-1.0thick-2tiny;
610   y5l:=2thick+y5r-2tiny; fi
611 if monospace: x1:=x1l:=x1r:=x5; y1:=y1-thin; y1l:=y1l-thin; y1r:=y1r-.6thin; fi
612 % readjust to have a better opening in CDTT
613 numeric slope_factor; if monospace:slope_factor=4/3; else:slope_factor=3/4; fi
614 % Additional G work (preceding program is the same as that for letter C)!
615 path p,p',line.o, line.i;
616 p=z1r{-1,slope_factor}..{left}pulled_arc.r(2,3)&
617   pulled_arc.r(3,4){right}..tension 0.9..z5r;
618 p'= reverse
619   (z1l..tension 0.9..{left}pulled_arc.l(2,3)&pulled_arc.l(3,4){right}
620     ..tension 0.9..z5l);
621 pos7(thick,0);
622 top z7=(x1r,7/2thick);
623 numeric counter; counter=0;
624 forever:
625   counter:=counter+1;
626   line.o:=z7r--(x7r,-2o); line.i:=(z7r-(thick,0))--(x7r-thick,-2o);
627   numeric t[],tt[];
628   (t1,tt1)=p intersectiontimes line.o; (t2,tt2)=p' intersectiontimes line.i;
629   exitif t1*t2>0; forsuffices $$=l, r: x7$$:=x7$$-.1u; endfor
630   x7:=x7-.1u; endfor
631 x6r=x7r; x6l=x7l; bot y6r=y part point t1 of p; bot y6l=y part point t2 of p';
632 z6=.5[z6r,z6l];
633 p:= subpath(0,t1) of p..(z6r-xeps+yeps)--(z6l+xeps+yeps)..
634   subpath(t2,infinity) of p'..(z1l+xeps)--cycle;
635 filldraw p;
636 filldraw z6l--z7l--z7r--z6r--cycle;
637 if serifs: serif(7,6,a,serif_darkness,-srاد);
638   serif(7,6,b,serif_darkness, srاد); fi
639 proofpcbb(3, 0.5thick, 0.5side);
640 penlabels(0,1,2,3,4,5,6,7); endchar;
641

```

H The letter H. With serifs, this fills the square. The two legs are thick and the cross-bar is thin like the A.

```

642
643 cmchar "Pacioli's letter H";
644 beginchar("H", u#+side#, cap_height#,0);

```

```

645 adjust_fit(cap_serif_fit#, cap_serif_fit#); pickup tiny.nib;
646 forsuffixes t=1,2,3,4: pos.t(thick,0); endfor
647 lft x1l=lft x2l=.5u+cap_jut; bot y2=0; top y1=h;
648 filldraw stroke z1e--z2e; % left vertical stem
649 x3=x4=w-x1; y3=y1; y4=y2;
650 filldraw stroke z3e--z4e; % right vertical stem
651 pos5(thin, 90); pos6(thin,90);
652 x5=.5[x1r,x1l]; x6=w-x5; top y5r=top y6r=h/2;
653 filldraw stroke z5e--z6e; % horizontal crossbar
654 if serifs:
655   serif(1,2,a,serif_darkness,-cap_jut); serif(1,2,b,serif_darkness,cap_jut);
656   serif(2,1,c,serif_darkness,-cap_jut); serif(2,1,d,serif_darkness,cap_jut);
657   serif(3,4,e,serif_darkness,-cap_jut); serif(3,4,f,serif_darkness,cap_jut);
658   serif(4,3,g,serif_darkness,-cap_jut); serif(4,3,h,serif_darkness,cap_jut); fi
659 proofpcbb(2, cap_jut+0.5thick, 0);
660 labels(1,2,3,4,5,6); endchar;
661

```

I The letter I.

```

662
663 cmchar "Pacioli's letter I";
664 beginchar("I", .5u#+2cap_jut#+cap_stem#+.5u#, cap_height#,0);
665 adjust_fit(cap_serif_fit#, cap_serif_fit#);
666 pickup tiny.nib; checkthin; pos1(cap_stem,0); pos2(cap_stem,0);
667 x1=x2=hround(w/2); top y1=h; bot y2=0;
668 filldraw stroke z1e--z2e; % vertical stem
669 if serifs:
670   serif(1, 2, a, serif_darkness, -cap_jut);
671   serif(1, 2, b, serif_darkness, cap_jut); % top lft,rt
672   serif(2, 1, c, serif_darkness, -cap_jut);
673   serif(2, 1, d, serif_darkness, cap_jut); % bot lft,rt
674 fi
675 proofpcbb(2, 0.5side, 0);
676 penlabels(1, 2); endchar;
677

```

J The letter J. Pacioli didn't specify a J, so we'll use Dürer's J instead (with suitable changes to the serifs).

```

678
679 cmchar "Pacioli's letter J";
680 %% Pacioli didn't have a J, so use Durer's.
681 beginchar("J",u#+3thick#+cap_jut#,cap_height#, 1/4 cap_height#);
682 adjust_fit(0, cap_serif_fit#); pickup tiny.nib;
683 pos1(thick,0); pos2(thick,0); pos3(thin, -90);
684 top y1=h; rt x1r=w-.5u-cap_jut; x2=x1; bot y2=0; lft x3=.5u; bot y3r=-d;
685 filldraw stroke z1e--z2e{down}..{left}z3e;
686 if serifs: serif(1,2,a,serif_darkness,-cap_jut);
687   serif(1,2,b,serif_darkness,cap_jut); fi
688 proofpcbb(2, 0.5side, 0);
689 penlabels(1,2,3); endchar;

```

690

K The letter K.

```
691
692 cmchar "Pacioli's letter K";
693 beginchar("K", 3/2u#+side#, cap_height#,0);
694 adjust_fit(cap_serif_fit#,0); pickup tiny.nib;
695 pcshiftx := 3/4u;
696 pair durer[]; % represent Pacioli's corner points
697 durer1=(pcshiftx,side); durer2=(pcshiftx+side,side); durer3=(xpart durer1,0);
698 durer4=(xpart durer2,0);
699 numeric diag_angle[];
700 diag_angle1=angle(durer2-durer3);
701 diag_angle2=angle(durer1-durer4);
702 numeric alpha;
703 alpha=((xpart durer2 - xpart durer1)+h)/h;
704 pos1(thick,0); pos2(thick,0);
705 lft x1l=lft x2l=pcshiftx+cap_jut;
706 top y1=h; bot y2=0; filldraw stroke z1e--z2e; % left stem
707 pos3(alpha*medium,0); pos4(alpha*medium,0); z4l = (x2,h/2);
708 top y3=h; z3r-z4r=whatever*(durer2-durer3);
709 z0=whatever[z1r,z2r]=whatever[z3l,z4l];
710 filldraw z4l--z3l--z3r--z4r--cycle; % upper diagonal stroke
711 pos5(alpha*thick,0); pos6(alpha*thick,0);
712 z5l = z4l;
713 bot y6=0; z5r-z6r=whatever*(durer1-durer4);
714 z10 = whatever[z3r,z4r]=whatever[z6r,z5r];
715 filldraw z6l--z5l--z10--z6r--cycle; % bottom leg
716 if serifs:
717   serif(1,2,a,serif_darkness,-cap_jut); serif(1,2,b,serif_darkness,cap_jut);
718   serif(2,1,c,serif_darkness,-cap_jut); serif(2,1,d,serif_darkness,cap_jut);
719   numeric old_jut, old_brack; old_jut=cap_jut; old_brack=bracket;
720 % serifs on top leg
721 bracket:=.5old_brack;cap_jut:=.5old_jut;
722 if not monospace: serif(3,4,e,serif_darkness,-cap_jut); fi
723 bracket:=2old_brack; cap_jut:=2old_jut;
724 serif(3,4,f,.6serif_darkness,if monospace:.7 fi cap_jut);
725 % serifs on bottom leg
726 bracket:=.5old_brack;cap_jut:=.5old_jut;
727 if not monospace: serif(6,5,g,serif_darkness,-cap_jut); fi
728 bracket:=2old_brack; cap_jut:=2old_jut;
729 serif(6,5,h,.6serif_darkness,if monospace:.7 fi cap_jut);
730 cap_jut:=old_jut; bracket:=old_brack;
731 fi
732 proofpcbb(2, pcshiftx+cap_jut+0.5thick, 0);
733 penlabels(0,1,2,3,4,5,6,7); endchar;
734
```

L The letter L. This is similar, but not identical, to the left leg and bottom arm of Pacioli's E.

```

735
736 cmchar "Pacioli's letter L";
737 numeric topbar_; if monospace: topbar_=11/15;
738   else: topbar_=17/30; fi % Duerer's wierd value
739 beginchar("L", u#+cap_jut#+5.75thick#, cap_height#,0);
740 adjust_fit(cap_serif_fit#,0);
741 pcshiftx := 0.5u;
742 pickup tiny.nib; pos1(cap_stem,0); pos2(cap_stem,0);
743 lft x1l=lft x2l=.5u+cap_jut; top y1=h; bot y2=0;
744 filldraw stroke z1e--z2e; % stem
745 pickup crisp.nib; % prepare for arms and serifs
746 pos3(thin,90);
747 pos8(medium,-90); pos9(dslab,0); x8=x2; bot y8r=0; rt x9r=hround(x2l+5.75thick);
748 y9=y8r+1.5thick; foot(8,9,g,serif_darkness,0.75thick);
749 % lower arm
750 if serifs: serif(1,2,a,serif_darkness,-cap_jut);
751   serif(2,1,b,serif_darkness,-cap_jut); serif(1,2,d,serif_darkness,cap_jut);
752   if not monospace: temp_:=slab; slab:=medium; tmp_:=bracket; bracket:=thick+medium;
753   serif(2,1,c,serif_darkness,thick);
754   slab:=temp_; bracket:=tmp_; fi fi
755 proofpcbb(2, cap_jut+0.5thick, 0);
756 penlabels(0,1,2,3,4,5,6,7,8,9); endchar; % end of "L"
757

```

- M The letter M. This has slightly sloping right and left legs. The serifs extend a little way outside the square.

```

758
759 cmchar"Pacioli's letter M";
760 beginchar("M", u#+cap_jut#+side#, cap_height#, 0);
761 adjust_fit(cap_serif_fit#, cap_serif_fit#); pickup tiny.nib;
762 pcshiftx := 0.5u;
763 pos1(medium,0); pos2(medium,0);
764 lft x2l=pcshiftx+cap_jut; top y1=h; bot y2=0;
765 lft x1l = lft x2l + side*sind(1); % left leg has 1 degree angle
766 pos3(thick,0); pos4(thick,0);
767 x4r=pcshiftx+side; top y3=h; bot y4=0;
768 x3r=x4r-side*sind(3); % right leg has 3 degree angle
769 numeric long, short, hypot, alpha; long=side; short=1/2side;
770 hypot=long++short; alpha=hypot/long;
771 pos5(alpha*thick,0); pos6(alpha*thick,0); % left inner
772 z5l=(2pcshiftx,side); bot lft z6l=(pcshiftx+1/2side,0);
773 pos7(alpha*medium,0); pos8(alpha*medium,0); % right inner
774 z7r=z6l; z8r=z3r; z0=whatever[z5r,z6r]=whatever[z7l,z8l];
775 z10=whatever[z5l,z6l]=whatever[z1r,z2r];
776 z11=whatever[z7r,z8r]=whatever[z3l,z4l];
777 % draw the M all in one go
778 filldraw z5r--z0--z8l--z3r--z4r--z4l--z11--z6l--z10--z2r--z2l--z1l--cycle;
779 if serifs:
780   serif(2,1,a,serif_darkness,-cap_jut); serif(1,2,b,serif_darkness,-lrad);
781   serif(3,4,c,serif_darkness, lrad); serif(4,3,d,serif_darkness, lrad);

```

```

782 serif(2,1,e,serif_darkness,cap_jut); serif(4,3,f,serif_darkness,-cap_jut);
783 % lower, inner serifs
784 fi
785 proofpcbb(2, cap_jut+0.5medium, 0);
786 penlabels(0,1,2,3,4,5,6,7,8,10,11); endchar;
787

```

- N The letter N. With serifs it fills the square, and perhaps the top left serif protrudes a tiny bit.

```

788
789 cmchar "Pacioli's letter N";
790 beginchar("N", u#+2cap_jut#+.8side#, cap_height#,0);
791 adjust_fit(cap_serif_fit#, cap_serif_fit#); pickup tiny.nib;
792 numeric long, short, hypot, alpha; long=h; short=side-2cap_jut;
793 hypot=long++short; alpha=hypot/long;
794 pos1(medium,0); pos2(medium,0); lft x1l=lft x2l=.5u+cap_jut; top y1=h; bot y2=0;
795 pos3(medium,0); pos4(medium,0); x3=x4; x3=x2+side-2cap_jut-medium; y3=y1; y4=y2;
796 pos 5(alpha*thick,0); pos6(alpha*thick,0);
797 z5r=z1r; z6l=z4r; z0=whatever[z1r,z2r]=whatever[z5l,z6l]; % under notch point
798 z0'=whatever[z5r,z6r]=whatever[z3l,z4l];
799 filldraw z1r--z1l--z2l--z2r--z0--z6l--z3r--z3l--z0'--cycle;
800 if serifs:
801 serif(2,1,b,serif_darkness,-cap_jut); serif(2,1,c,serif_darkness,cap_jut);
802 % bottom, left serifs
803 serif(3,4,d,serif_darkness,-cap_jut); serif(3,4,e,serif_darkness,cap_jut);
804 % top right serifs
805 z.a=(x1l-1.5cap_jut,y1); pickup tiny.nib; filldraw
806 z0{z0-z4r}..{left}(z.a-(0,eps))..z.a--z1r--cycle;
807 % top left serif
808 serif(1,2,a,serif_darkness,-lrad);
809 fi
810 proofpcbb(2, cap_jut+0.5medium, 0);
811 penlabels(0,0',1,2,3,4,5,6); endchar;
812

```

- o The letter O. He actually specifies two versions of the letter, but this one is *perfectissimo*. It is oversquare horizontally.

```

813
814 cmchar "Pacioli's original letter O";
815 beginchar("o", 2u#+side#+2o#, cap_height#, 0);
816 adjust_fit(0,0); pair durer[];
817 pcshiftx := u;
818 x2r=pcshiftx-o; x4r=w-pcshiftx+o; y1r=h; y3r=0; z0=(w/2,h/2);
819 z0=.5[z5,z6]; numeric axis_angle;
820 durer1=(x2r,y1r-o); durer2=(x4r, ypart durer1);
821 durer3=(xpart durer1, y3r+o); durer4=(xpart durer2, ypart durer3);
822 axis_angle=angle(durer2-durer3);
823 z5-z6=(thick,0)rotated axis_angle;
824 y2r=y6; y4r=y5; x1r=x5; x3r=x6;

```

```

825 forsuffixes $=1,2,3,4: z$=.5[z$l,z$r]; endfor
826 x1l=x3r; y1l-y6=y6-y3r; x5-x2l=x4r-x5; y2l=y4r;
827 x3l=x1r; y1r-y5=y5-y3l; x4l-x6=x6-x2r; y4l=y2r;
828 path p,p';
829 p=pulled_arc.r(1,2)&reverse pulled_arc.r(3,2)&
830   pulled_arc.r(3,4)&reverse pulled_arc.r(1,4)&cycle;
831 path anti_diag; anti_diag=z0--durer1;
832 z21r=p intersectionpoint anti_diag; penpos21(thin, axis_angle+90);
833 x43r=w-x21r; y43r=h-y21r; penpos43(thin, axis_angle+270);
834 path diag; diag=z0--durer2;
835 z41r=diag intersectionpoint p; penpos41(thick, axis_angle);
836 x32r=w-x41r; y32r=h-y41r; penpos32(thick, axis_angle+180);
837 if not monospace: p'=
838   z1l{left}..z21l..z21{down}..z32l..
839   z3l{right}..z43l..
840   z4l{up}..z41l..{left}cycle;
841 else:p'=z1l..z21l..z32l..z3l..z43l..z41l..cycle; fi
842 fill p; unfill p';
843 proofpcbb(0, 0.5side, 0.5side);
844 penlabels(0,1,2,3,4,5,6,21,32,43,41); endchar;
845

```

0 The letter O. This includes optical adjustments at the top and bottom.

```

846
847 cmchar "Pacioli's letter 0";
848 beginchar("0", 2u#+side#+2o#, cap_height#, 0);
849 adjust_fit(0,0); pair durer[];
850 pcshiftx := u;
851 x2r=pcshiftx-o; x4r=w-pcshiftx+o; y1r=h+o; y3r=-o; z0=(w/2,h/2);
852 z0=.5[z5,z6]; numeric axis_angle;
853 durer1=(x2r,y1r-o); durer2=(x4r, ypart durer1);
854 durer3=(xpart durer1, y3r+o); durer4=(xpart durer2, ypart durer3);
855 axis_angle=angle(durer2-durer3);
856 z5-z6=(thick,0)rotated axis_angle;
857 y2r=y6; y4r=y5; x1r=x5; x3r=x6;
858 forsuffixes $=1,2,3,4: z$=.5[z$l,z$r]; endfor
859 x1l=x3r; y1l-y6=y6-y3r; x5-x2l=x4r-x5; y2l=y4r;
860 x3l=x1r; y1r-y5=y5-y3l; x4l-x6=x6-x2r; y4l=y2r;
861 path p,p';
862 p=pulled_arc.r(1,2)&reverse pulled_arc.r(3,2)&
863   pulled_arc.r(3,4)&reverse pulled_arc.r(1,4)&cycle;
864 path anti_diag; anti_diag=z0--durer1;
865 z21r=p intersectionpoint anti_diag; penpos21(thin, axis_angle+90);
866 x43r=w-x21r; y43r=h-y21r; penpos43(thin, axis_angle+270);
867 path diag; diag=z0--durer2;
868 z41r=diag intersectionpoint p; penpos41(thick, axis_angle);
869 x32r=w-x41r; y32r=h-y41r; penpos32(thick, axis_angle+180);
870 if not monospace: p'=
871   z1l{left}..z21l..z21{down}..z32l..
872   z3l{right}..z43l..

```

```

873 z4l{up}..z41l..{left}cycle;
874 else:p'=z1l..z21l..z32l..z3l..z43l..z41l..cycle; fi
875 fill p; unfill p';
876 proofpcbb(0, 0.5side, 0.5side);
877 penlabels(0,1,2,3,4,5,6,21,32,43,41); endchar;
878

```

P The letter P.

```

879
880 cmchar "Pacioli's letter P";
881 beginchar("P", 1.5u#+P_width#, cap_height#, 0);
882 adjust_fit(cap_serif_fit#, 0);
883 pickup tiny.nib; pos1(thick,0); pos2(thick,0);
884 top lft z1l=(.5u+cap_jut,h); bot lft z2l=(.5u+cap_jut,0);
885 filldraw stroke z1e--z2e; % vertical stem
886 pos3(medium,90); pos4(medium,90); pos5(thick,0);
887 pos6(if not monospace and serifs: .5thin else:thin fi,-90);
888 top y3r=top y4r=h; x3=x1; x4=x1r+thick; filldraw stroke z3e--z4e;
889 x6=x4; y6=hround(h/2 -medium); rt x5r=hround(x2+0.65side-cap_jut-0.5medium); y5=.75side;
890 y7=y6; if serifs: x7=.5[rt x1r, x6]; else: x7=x3; fi
891 pos7(if not monospace and serifs: .5thin else:thin fi,-90);
892 filldraw stroke pulled_super_arc.e(4,5,eta)&pulled_super_arc.e(5,6,eta)
893 ..z7e; % lobe
894 if serifs: serif(1,2,a,serif_darkness,-cap_jut);
895 serif(2,1,b,serif_darkness,-cap_jut);
896 serif(2,1,c,serif_darkness,cap_jut); fi
897 proofpcbb(2, cap_jut+0.5thick, 0);
898 penlabels(1,2,3,4,5,6,7); endchar;
899

```

q The letter Q. It is essentially an O with a long (not very attractive) tail.

```

900
901 cmchar "Pacioli's original letter Q";
902 beginchar("q", 2u#+side#+2o#, cap_height#, 0);
903 adjust_fit(0,0); pair durer[];
904 pcshiftx := u;
905 x2r=pcshiftx-o; x4r=w-pcshiftx+o; y1r=h; y3r=0; z0=(w/2,h/2);
906 z0=.5[z5,z6]; numeric axis_angle;
907 durer1=(x2r,y1r-o); durer2=(x4r, ypart durer1);
908 durer3=(xpart durer1, y3r+o); durer4=(xpart durer2, ypart durer3);
909 axis_angle=angle(durer2-durer3);
910 z5-z6=(thick,0)rotated axis_angle;
911 y2r=y6; y4r=y5; x1r=x5; x3r=x6;
912 forsuffices $=1,2,3,4: z$=.5[z$l,z$r]; endfor
913 x1l=x3r; y1l=y6=y6-y3r; x5-x2l=x4r-x5; y2l=y4r;
914 x3l=x1r; y1r-y5=y5-y3l; x4l-x6=x6-x2r; y4l=y2r;
915 path p,p';
916 p=pulled_arc.r(1,2)&reverse pulled_arc.r(3,2)&
917 pulled_arc.r(3,4)&reverse pulled_arc.r(1,4)&cycle;
918 path anti_diag; anti_diag=z0--durer1;

```

```

919 z21r=p intersectionpoint anti_diag; penpos21(thin, axis_angle+90);
920 x43r=w-x21r; y43r=h-y21r; penpos43(thin, axis_angle+270);
921 path diag; diag=z0--durer2;
922 z41r=diag intersectionpoint p; penpos41(thick, axis_angle);
923 x32r=w-x41r; y32r=h-y41r; penpos32(thick, axis_angle+180);
924 if not monospace: p'=
925   z1l{left}..z21l..z2l{down}..z32l..
926   z3l{right}..z43l..
927   z4l{up}..z41l..{left}cycle;
928 else:p'=z1l..z21l..z32l..z3l..z43l..z41l..cycle; fi
929 fill p; unfill p';
930 % now for the tail!
931 z10r=z3l; z10l=z3r;
932 z11r=(pcshiftx+side,-0.25side); z11l=z11r-(0,thick);
933 z12l=z11l+(0.5side+medium,-thick); z12r=z12l+(0,0.5thin);
934 filldraw z10r..z11r..z12r--z12l..z11l..z10l--cycle;
935 proofpcbb(0, 0.5side, 0.5side);
936 penlabels(0,1,2,3,4,5,6,10,11,12,13,14,15); endchar;
937

```

Q The letter Q. This includes the optical adjustments.

```

938
939 cmchar "Pacioli's letter Q";
940 beginchar("Q", 2u#+side#+2o#, cap_height#, 0);
941 adjust_fit(0,0); pair durer[];
942 pcshiftx := u;
943 x2r=pcshiftx-o; x4r=w-pcshiftx+o; y1r=h+o; y3r=-o; z0=(w/2,h/2);
944 z0=.5[z5,z6]; numeric axis_angle;
945 durer1=(x2r,y1r-o); durer2=(x4r, ypart durer1);
946 durer3=(xpart durer1, y3r+o); durer4=(xpart durer2, ypart durer3);
947 axis_angle=angle(durer2-durer3);
948 z5-z6=(thick,0)rotated axis_angle;
949 y2r=y6; y4r=y5; x1r=x5; x3r=x6;
950 forsuffixes $=1,2,3,4: z$=.5[z$1,z$r]; endfor
951 x1l=x3r; y1l-y6=y6-y3r; x5-x2l=x4r-x5; y2l=y4r;
952 x3l=x1r; y1r-y5=y5-y3l; x4l-x6=x6-x2r; y4l=y2r;
953 path p,p';
954 p=pulled_arc.r(1,2)&reverse pulled_arc.r(3,2)&
955   pulled_arc.r(3,4)&reverse pulled_arc.r(1,4)&cycle;
956 path anti_diag; anti_diag=z0--durer1;
957 z21r=p intersectionpoint anti_diag; penpos21(thin, axis_angle+90);
958 x43r=w-x21r; y43r=h-y21r; penpos43(thin, axis_angle+270);
959 path diag; diag=z0--durer2;
960 z41r=diag intersectionpoint p; penpos41(thick, axis_angle);
961 x32r=w-x41r; y32r=h-y41r; penpos32(thick, axis_angle+180);
962 if not monospace: p'=
963   z1l{left}..z21l..z2l{down}..z32l..
964   z3l{right}..z43l..
965   z4l{up}..z41l..{left}cycle;
966 else:p'=z1l..z21l..z32l..z3l..z43l..z41l..cycle; fi

```



```

967 fill p; unfill p';
968 % now for the tail!
969 z10r=z3l; z10l=z3r;
970 z11r=(pcshiftx+side,-0.25side); z11l=z11r-(0,thick);
971 z12l=z11l+(0.5side+medium,-thick); z12r=z12l+(0,0.5thin);
972 filldraw z10r..z11r..z12r--z12l..z11l..z10l--cycle;
973 proofpcbb(0, 0.5side, 0.5side);
974 penlabels(0,1,2,3,4,5,6,10,11,12,13,14,15); endchar;
975

```

R The letter R. Another character that, with serifs, fills the square.

```

976
977 cmchar "Pacioli's letter R";
978 beginchar("R", u#+side#, cap_height#,0);
979 adjust_fit(cap_serif_fit#,0);
980 pickup tiny.nib; pos1(thick,0); pos2(thick,0);
981 top lft z1l=(.5u+cap_jut,h); bot lft z2l=(.5u+cap_jut,0);
982 filldraw stroke z1e--z2e; % vertical stem
983 pos3(thin,90); pos4(thin,90); pos5(thick,0);
984 pos6(thin, -90);
985 top y3r=top y4r=h; x3=x1; x4=x1r+thick;
986 x6=x4; y6=hround h/2; rt x5r=hround(.5u+P_width if not monospace:
987 +1/2thin fi); y5=.5[y4,y6];
988 y7=y6; x7=x3; pos7(thin, -90);
989 filldraw stroke z3e--pulled_super_arc.e(4,5,eta)&
990 pulled_super_arc.e(5,6,eta)..z7e; % lobe
991 numeric half_letter_width; 2half_letter_width+1/2u=w;
992 if serifs: half_letter_width:=half_letter_width+1/2u;
993 else: half_letter_width:=half_letter_width-1.6thin; fi
994 path p; p=pulled_super_arc.r(5,6,eta)..z7r;
995 z0r=p intersectionpoint ((half_letter_width,0)--(half_letter_width,h));
996 path circ_; circ_ = (((halfcircle scaled 2thick) rotated 180) shifted z0r);
997 z0l= p intersectionpoint circ_;
998 if not monospace: z0'=(.45[half_letter_width,w],0);
999 else: z0'=(.42[half_letter_width,w],0); fi
1000 numeric tail_angle; tail_angle=angle(z0l-z0');
1001 penpos8(thick, tail_angle-90); z8l=whatever[z0', z0l];
1002 x8=1/5[half_letter_width,w];
1003 penpos9(2/3thick,tail_angle-90); z9r=whatever[z0r, z8r];
1004 x9r=.5[half_letter_width, w];
1005 %penpos10(if monospace:1/2 fi dslab,90); z10l=(w,0);
1006 penpos10(if monospace:1/2 dslab else: .5thin fi,90); z10l=(w,0);
1007 fill z0l--z8l...{right}z10l--z10r{-w,1/4h}..z9r--z0r--cycle; % tail
1008 if serifs: serif(1,2,a,serif_darkness,-cap_jut);
1009 serif(2,1,b,serif_darkness,-cap_jut);
1010 serif(2,1,c,serif_darkness,cap_jut); fi
1011 proofpcbb(2, 1.5cap_jut+0.5thick, 0);
1012 penlabels(0,1,2,3,4,5,6,7,8,9,10); labels(0'); endchar;
1013

```

- S The letter S. Pacioli's instructions are unclear. I have used a modified version of Hoenig's modification of the Knuthian S, which appears to match Pacioli's drawing.

```

1014
1015 cmchar "The Knuthian S";
1016 beginchar("S", 2u#+1/2 side# if not monospace:+thick# fi, cap_height#, 0);
1017 adjust_fit(0,0); pickup tiny.nib;
1018 pair durer[]; % points to contain corner coords of Durer square
1019 ypart durer3=ypart durer4=0; xpart durer4-xpart durer1=side;
1020 xpart durer1=u-1.5thick;
1021 durer1=(xpart durer3, h); durer2=(xpart durer 4, ypart durer1);
1022 numeric slope, diag_angle; diag_angle=angle(durer2-durer3);
1023 slope=-if not monospace: 3/4 else: 1/2 fi h/(xpart durer4-xpart durer3);
1024 pos0(thick,diag_angle); x0=.52[xpart durer1, xpart durer2];
1025 y0=h/2;
1026 lft x3l=u if not monospace:+2/3thick fi; rt x3r=lft x3l+5/6thick;
1027 pos2(thin,-diag_angle); top y2l=h;
1028 if x0-x3r>thick: x2l=x0 if monospace:+.4thin fi; else:
1029   x2l=x0r; slope=-1/2; fi
1030 ellipse_set(2l,3l,4l,0l); ellipse_set(2r,3r,4r,0r);
1031 pos7(thin,-diag_angle); x7r=x0; bot y7r=0; rt x6r=w-if monospace:.75 fi u;
1032 lft x6l=rt x6r-thick;
1033 ellipse_set(7l,6l,5l,0l); ellipse_set(7r,6r,5r,0r);
1034 pos1(max(currentbreadth+eps,5/6 thick),-90);
1035 top y1l=h-thin; rt x1=x0+2thick+o;
1036 pos8(1.5thick,-80);lft x8r=u; bot y8r=medium;
1037 if not monospace:
1038   filldraw z1l..z2l{left}..{down}z3l..z4l--z5l..{down}z6l..{left}z7l..z8l
1039   --z8r..{right}z7r..{up}z6r..z5r--z4r..{up}z3r..{right}z2r..z1r--cycle;
1040 else: draw z1l..z2l{left}..{down}z3l..z4l--z5l..{down}z6l..{left}z7l..z8l;
1041   draw z8r..{right}z7r..{up}z6r..z5r--z4r..{up}z3r..{right}z2r..z1r;
1042 fi
1043 proofpcbb(0, 0.5side, 0.5side);
1044 penlabels(0,1,2,3,4,5,6,7,8); endchar;
1045

```

- T The letter T.

```

1046
1047 cmchar "Pacioli's letter T";
1048 beginchar("T",2u#+side#-2medium#, cap_height#,0);
1049 adjust_fit(0,0); pickup tiny.nib;
1050 pos1(thick,0); pos2(thick,0); .5[x1r,x1l]=w/2; .5[x2r,x2l]=w/2;
1051 top y1=h; bot y2=0; filldraw stroke z1e--z2e; % central stem
1052 % prepare for the top horizontal bar
1053 numeric top_bar; top_bar=0.5side-medium; % half length of top bar
1054 pos3(medium,90); pos4(dslab,0);
1055 x1=x3; top y3r=h; rt x4l=x1+top_bar;
1056 bot y4=vround(h-if monospace: 2 fi vstretch*(medium+cap_jut));
1057 arm(3,4,c,serif_darkness,0);

```

```

1058 pos5(dslab,180);
1059 lft x5r=x1-top_bar;
1060 bot y5= bot y4;
1061 arm(3,5,d,serif_darkness,0);
1062 if serifs:
1063 serif(2,1,a,serif_darkness,-cap_jut);serif(2,1,b,serif_darkness,cap_jut);
1064 fi
1065 proofpcbb(2, 0.5side, 0);
1066 penlabels(1,2,3,4); endchar;
1067

```

U The letter U. Pacioli doesn't give a U, so I have used Hoenig's U, suitably modified for leg and serif dimensions.

```

1068
1069 cmchar "Pacioli's letter U";
1070 beginchar("U", u#+2cap_jut#+.8side#, cap_height#,0);
1071 adjust_fit(cap_serif_fit#, cap_serif_fit#);
1072 pickup tiny.nib; pos1(thick,0); pos2(thick,0); pos2'(thick,180);
1073 pos3(medium,-90); pos4(medium,0); pos5(medium,0);
1074 x1=x2; x3=.5[x1,x5]; x4=x5; x1l=w-x5r;
1075 top y1=top y5=h; y2=y4=1/3h; bot y3r=0; z2'=z2;
1076 lft x1l=hround .5u+cap_jut;
1077 filldraw stroke z1e--z2e; % left stem
1078 filldraw stroke pulled_super_arc.e(2',3,eta)
1079 & pulled_super_arc.e(3,4,eta)&z4e--z5e;
1080 if serifs: serif(1,2,a,serif_darkness,-cap_jut);
1081 serif(1,2,b,serif_darkness,cap_jut);
1082 serif(5,4,c,serif_darkness,-cap_jut); serif(5,4,d,serif_darkness,cap_jut); fi
1083 penlabels(1,2,3,4,5); endchar;
1084

```

V The letter V.

```

1085
1086 cmchar "Pacioli's letter V";
1087 beginchar("V",u#+side#, cap_height#, 0);
1088 adjust_fit(cap_serif_fit#, cap_serif_fit#); pickup tiny.nib;
1089 pcshiftx := 0.5u;
1090 z10 = (pcshiftx+0.5side,0);
1091 numeric alpha; alpha=((0.5h-cap_jut)+(h))/(h);
1092 pos1(alpha*thick,0); pos2(alpha*thick,0);
1093 pos3(alpha*medium,0); pos4(alpha*medium,0);
1094 top y1=top y4=h; lft x1l=pcshiftx+cap_jut; x4r=x1l+side-2cap_jut;
1095 lft x2l=hround(pcshiftx+0.5side); x3r=x2l; bot y2=bot y3=0;
1096 z0=whatever[z1r,z2r]=whatever[z3l,z4l]; % notch point
1097 filldraw z0--diag_end(0,4l,1,1,4r,3r)--z2l
1098 --diag_end(2l,1l,1,1,1r,0)--cycle;
1099 if serifs: bracket:=cap_jut_factor*bracket; cap_jut:=cap_jut_factor*cap_jut;
1100 pickup tiny.nib; numeric jut_factor;
1101 if monospace: jut_factor=1; else: jut_factor=3/4; fi
1102 serif(1,2,a,.8serif_darkness,-Lrad);

```

```

1103 serif(4,3,d,.8serif_darkness,lrad);
1104 bracket:=jut_factor*bracket;
1105 serif(1,2,b,7/8 serif_darkness, jut_factor*cap_jut);
1106 serif(4,3,c,7/8 serif_darkness,-jut_factor*cap_jut); fi
1107 proofpcbb(10, 0.5side, 0);
1108 penlabels(0,1,2,3,4); endchar;
1109

```

W The letter W. Again, Pacioli misses this out. I use a modified version of Hoenig's W.

```

1110
1111 cmchar "Pacioli's letter W";
1112 numeric alpha;
1113 alpha=((cap_height#+cap_jut#)+(4side#+cap_jut_factor*cap_jut#))/
1114      (cap_height#+cap_jut#);
1115 beginchar("W", 2cap_jut_factor*cap_jut#+1.6side#-alpha*thick#, cap_height#, 0);
1116 adjust_fit(.5cap_serif_fit#, cap_serif_fit#); pickup tiny.nib;
1117 mid_corr:=if monospace: -apex_corr else: 1/3jut fi;
1118 pos1(alpha*thick,0); pos2(alpha*thick,0);
1119 pos3(alpha*medium,0); pos4(alpha*medium,0);
1120 pos5(alpha*thick,0); pos6(alpha*thick,0);
1121 pos7(alpha*medium,0); pos8(alpha*medium,0);
1122 numeric V_width; V_width=2cap_jut_factor*cap_jut+4/5side;
1123 if not monospace:
1124   top y1=top y4=h; lft x1l=cap_jut_factor*cap_jut; x4r=V_width-x1l;
1125   lft x2l=hround V_width/2; x3r=x2l; bot y2=bot y3=0;
1126   pair shift_amt; ypart shift_amt=0;
1127   xpart shift_amt=V_width-2cap_jut_factor*cap_jut-alpha*(thick);
1128   z0=whatever[z1r,z2r]=whatever[z3l,z4l]; % notch point
1129   path p; p=z0--diag_end(0,4l,1,1,4r,3r)--z2l
1130     --diag_end(2l,1l,1,1,1r,0)--cycle;
1131   filldraw p; filldraw (p shifted shift_amt); % the 2 V's
1132   forsuffixes t=1,2,3,4: z[t+4]=z.t shifted shift_amt; endfor
1133 else: top y1=h; lft x1l=cap_jut_factor*cap_jut; bot y2=bot y3=0;
1134   x2l+apex_corr=x3l;
1135   x8=w-x1; y8=y1; x2-x1=x4-x3=x6-x5=x8-x7;
1136   x6l+apex_corr=x7l; y6=y7=y2;
1137   y4:=vround .6h; y5=y4;
1138   x5=x4;
1139   y4l:=y4r:=y4; % ??
1140   z0 =whatever[z1r,z2r];
1141   z0'=whatever[z7l,z8l]; y0=y0'=h/9;
1142   z45=whatever[z3r,z4r]=whatever[z5l,z6l];
1143   filldraw z0--diag_end(0,4l,1,1,4r,3r)--z2l--diag_end(2l,1l,1,1,1r,0)--cycle;
1144   filldraw z0'--diag_end(0',8l,1,1,8r,7r)--z7l--diag_end(7l,5l,1,1,5r,0')
1145     --cycle; fi
1146 if serifs: bracket:=cap_jut_factor*bracket; cap_jut:=cap_jut_factor*cap_jut;
1147 pickup tiny.nib; numeric jut_factor;
1148 if monospace: jut_factor=1.8; else: jut_factor=3/4; fi
1149 serif(1,2,a,.8serif_darkness,-cap_jut);

```

```

1150 serif(8,7,f,.8serif_darkness,cap_jut);
1151 if not monospace: serif(5,6,d,.8serif_darkness,cap_jut);
1152 serif(5,6,c,.8serif_darkness,-cap_jut); fi
1153 bracket:=jut_factor*bracket;
1154 serif(1,2,b,7/8 serif_darkness, jut_factor*cap_jut);
1155 serif(8,7,e,7/8 serif_darkness,-jut_factor*cap_jut);
1156 fi
1157 penlabels(0,0',1,2,3,4,5,6,7,8,45); endchar;
1158

```

X The letter X.

```

1159
1160 cmchar "Pacioli's letter X";
1161 beginchar("X", u#+side#, cap_height#, 0);
1162 adjust_fit(cap_serif_fit#, cap_serif_fit#);
1163 pcshiftx := 0.5u;
1164 z10=(pcshiftx+0.5side, 0.5side); % center of the square
1165 numeric alpha; alpha=(h++(h-2thick))/h;
1166 penpos1(alpha*thick,0); penpos4(alpha*thick,0);
1167 penpos2(alpha*medium,0); penpos3(alpha*medium,0);
1168 lft x3l = pcshiftx+thick; y3=0; y2=side;
1169 0.5[z3l,z2l]=z10;
1170 y4 = 0; y1 = side;
1171 lft x1l = pcshiftx+1.5thick;
1172 0.5[z4l,z1l]=z10;
1173 filldraw z1l--z1r--z4r--z4l--cycle; % draw the thick limb
1174 filldraw z2l--z2r--z3r--z3l--cycle; % draw the other limb
1175 if serifs: bracket:=cap_jut_factor*bracket; cap_jut:=cap_jut_factor*cap_jut;
1176 pickup tiny.nib; numeric jut_factor;
1177 if monospace: cap_jut:=.5cap_jut; jut_factor=1; else: jut_factor=3/4; fi
1178 serif(1,4,a,.8serif_darkness,-Lrad);
1179 serif(4,1,d,.8serif_darkness,cap_jut);
1180 serif(2,3,f,.8serif_darkness,cap_jut);
1181 serif(3,2,g,.8serif_darkness,-lrad);
1182 bracket:=jut_factor*bracket;
1183 serif(1,4,b,7/8 serif_darkness,.75cap_jut);
1184 serif(4,1,c,7/8 serif_darkness,-.75cap_jut);
1185 serif(2,3,e,7/8 serif_darkness,-.75cap_jut);
1186 serif(3,2,h,7/8 serif_darkness,.75cap_jut); fi
1187 proofpcbb(10, 0.5side, 0.5side);
1188 penlabels(1,2,3,4); endchar;
1189

```

Y The letter Y.

```

1190
1191 cmchar "Pacioli's letter Y";
1192 beginchar("Y", u#+side#, cap_height#, 0);
1193 adjust_fit(cap_serif_fit#, cap_serif_fit#); pickup pencircle scaled Opt;
1194 pcshiftx := 0.5u;
1195 z10 = (pcshiftx+0.5side,0.5side); % center of the square

```

```

1196 numeric alpha, long, short, hypot; long=h/2;
1197 short=0.5side-thick;
1198 hypot=long++short; alpha=hypot/long;
1199 penpos1(alpha*thick,0);
1200 lft x1l=pcshiftx+thick;
1201 top y1=h;
1202 penpos4(alpha*medium,0); y4=y1; x4r=x1l+side-2thick;
1203 penpos2(alpha*thick,0); penpos3(alpha*medium,0);
1204 y2 = y3 = 0.5side;
1205 penpos5(thick,0); penpos6(thick,0);
1206 z2l = z3r = z100;
1207 z6 = (pcshiftx+0.5side,0);
1208 z5 = (x6,side);
1209 z0=whatever[z1r, z2r]=whatever[z3l,z4l]; % notch point
1210 z7=whatever[z6l,z5l] = whatever[z2l,z1l]; % left stem and V
1211 z8=whatever[z6r,z5r] = whatever[z3r,z4r]; % right stem and V
1212 filldraw z1r--z0--z4l--z4r--z8--z6r--z6l--z7--z1l--cycle;
1213 if serifs: numeric jut_factor;
1214 serif(6,5,e,serif_darkness,-cap_jut); serif(6,5,f,serif_darkness,cap_jut);
1215 bracket:=cap_jut_factor*bracket; cap_jut:=cap_jut_factor*cap_jut;
1216 if monospace: cap_jut:=.5cap_jut; jut_factor=1; else: jut_factor=3/4; fi
1217 % top outer serifs
1218 serif(1,2,a,serif_darkness,-lrad);serif(4,3,d,serif_darkness,lrad);
1219 bracket:=jut_factor*bracket;
1220 if monospace: cap_jut:=1.5cap_jut; fi
1221 % top inner serifs
1222 serif(1,2,b,7/8 serif_darkness,.6cap_jut);
1223 serif(4,3,c,7/8 serif_darkness,-.6cap_jut);
1224 fi
1225 proofpcbb(100, 0.5side, 0.5side);
1226 penlabels(0,1,2,3,4,5,6); endchar; % End of "Y"
1227

```

- Z The letter Z. The last of the letters that Pacioli doesn't describe. Again, use a modified Dürer Z.

```

1228
1229 cmchar "Pacioli's letter Z";
1230 beginchar("Z", 2u#+.8side#, cap_height#,0);
1231 adjust_fit(0, 0); pickup tiny.nib;
1232 pcshiftx := u;
1233 numeric alpha, long, short, hypot;
1234 long=h; short=w-2u-thick; hypot=long++short;
1235 alpha=hypot/long;
1236 pos1(medium,90); pos2(medium,90); pos3(alpha*thick, 0);
1237 pos4(alpha*thick, 0); pos5(medium,-90); pos6(medium,-90);
1238 rt x3r=w-u; top y3=h; x4l=w-x3r; bot y4=0;
1239 top y1r=h; lft x1=u; y2=y1; rt x2=x3;
1240 bot y5r=bot y6r=0; lft x5=x4; rt x6=w-u;
1241 filldraw stroke z1e--z2e; % top bar
1242 filldraw stroke z3e--z4e; % diagonal

```

```

1243 filldraw stroke z5e--z6e; % bottom bar
1244 z100 = (0.5(x1+x3r),0.5side); % center of square
1245 if serifs: pickup crisp.nib; % prepare for arms and serifs
1246 pos7(medium, 90); pos8(dslab,180); z7=z2; lft x8r=u;
1247 bot y8=h-if monospace:2 fi vstretch*thick;
1248 arm(7,8,a,serif_darkness,0); % upper dropped serif
1249 pos9(medium,-90); z9=z5; pos10(dslab,0); rt x10r=w-u;
1250 top y10=if monospace: 2.5 fi vstretch*thick;
1251 arm(9,10,b,serif_darkness,0); fi
1252 proofpcbb(100, 0.5side, 0.5side);
1253 penlabels(1,2,3,4,5,6,7,8,9,10); endchar;
1254
1255
1256 </maj>

```

3.4 The punctuation file

This is essentially a copy of Knuth's Computer Modern Roman punctuation file `punct.mf`.

```

1257 <*punct>
1258 % CPCPUNCT.MF This file contains punctuation marks common to
1259 %           Pacioli roman and italic styles.
1260
! The ! character.

1261
1262 cmchar "Exclamation point";
1263 beginchar("!",5u#+width_adj#,asc_height#,0);
1264 italcorr asc_height#*slant-2u#-.5width_adj#+.5dot_size#;
1265 adjust_fit(0,0);
1266 pickup tiny.nib; pos3(dot_size,0); pos4(dot_size,90);
1267 lft x3l=hround(.5w-.5dot_size); bot y4l=0; z3=z4; dot(3,4); % dot
1268 numeric bot_width;
1269 bot_width=if hefty:max(hround .8dot_size,fine.breadth) else: hair fi;
1270 pickup fine.nib; pos1(dot_size,0); pos2(bot_width,0);
1271 x1=x2=x3; bot y2=.25[top y4r,x_height]+1;
1272 if square_dots: top y1=h;
1273 filldraw stroke z1e--z2e; % stem
1274 else: top z0=(x1,h+o); y1+.5dot_size=h+o;
1275 filldraw z1r...z0...z1l---z2l--z2r---cycle; fi % stem and bulb
1276 penlabels(0,1,2,3,4); endchar;
1277

' The ' character.

1278
1279 cmchar "Apostrophe";
1280 beginchar("'",5u#,asc_height#,0);
1281 italcorr asc_height#*slant+.5dot_size#-2u#;
1282 adjust_fit(0,0);

```

```

1283 x1-.5dot_size=hround(.5w-.5dot_size); y1+.5dot_size=h;
1284 if monospace: comma(1,a,dot_size,.28u,vround 1.5comma_depth); % large comma
1285 else: comma(1,a,dot_size,.25u,comma_depth); fi % comma with increased jut
1286 penlabels(1); endchar;
1287

( The ( character.

1288
1289 cmchar "Left parenthesis";
1290 beginchar("(",7u# if monospace: -u# fi,body_height#,paren_depth#);
1291 italcorr body_height#*slant-.5u#;
1292 adjust_fit(0,0); pickup fine.nib;
1293 pos1(vair,0); pos2(.75[hair,stem],0); pos3(vair,0);
1294 rt x1r=rt x3r=hround(w-u); lft x2l=hround(x1-4u if monospace: +4/3u fi);
1295 top y1=h; y2=.5[y1,y3]=math_axis;
1296 filldraw stroke z1e{3(x2e-x1e),y2-y1}...z2e
1297 ...{3(x3e-x2e),y3-y2}z3e; % arc
1298 penlabels(1,2,3); endchar;
1299

) The ) character.

1300
1301 cmchar "Right parenthesis";
1302 beginchar(")",7u# if monospace: -u# fi,body_height#,paren_depth#);
1303 italcorr math_axis#*slant-.5u#;
1304 adjust_fit(0,0); pickup fine.nib;
1305 pos1(vair,0); pos2(.75[hair,stem],0); pos3(vair,0);
1306 lft x1l=lft x3l=hround u; rt x2r=hround(x1+4u if monospace: -4/3u fi);
1307 top y1=h; y2=.5[y1,y3]=math_axis;
1308 filldraw stroke z1e{3(x2e-x1e),y2-y1}...z2e
1309 ...{3(x3e-x2e),y3-y2}z3e; % arc
1310 penlabels(1,2,3); endchar;
1311

, The , character.

1312
1313 cmchar "Comma";
1314 numeric dot_diam#; dot_diam#=if monospace: 5/4 fi\\ dot_size#;
1315 define_whole_blacker_pixels(dot_diam);
1316 beginchar(",",5u#,dot_diam#,comma_depth#);
1317 adjust_fit(0,0);
1318 x1-.5dot_diam=hround(.5w-.5dot_diam); y1-.5dot_diam=0;
1319 comma(1,a,dot_diam,.2u,comma_depth); % dot and tail
1320 penlabels(1); endchar;
1321

. The . character.

1322
1323 cmchar "Period";

```



```

1324 numeric dot_diam#; dot_diam#=if monospace: 5/4 fi\\ dot_size#;
1325 define_whole_blacker_pixels(dot_diam);
1326 beginchar(".",5u#,dot_diam#,0);
1327 adjust_fit(0,0); pickup fine.nib;
1328 pos1(dot_diam,0); pos2(dot_diam,90);
1329 lft x1=hround(.5w-.5dot_diam); bot y2l=0; z1=z2; dot(1,2); % dot
1330 penlabels(1,2); endchar;
1331

/ The / character.
1332
1333 cmchar "Virgule (slash)";
1334 beginchar("/",9u#,body_height#,paren_depth#);
1335 italcorr body_height#*slant-.5u#;
1336 adjust_fit(0,0); pickup rule.nib;
1337 rt x1=hround(w-u)+eps; top y1=h+eps;
1338 lft x2=hround u-eps; bot y2=-d-eps;
1339 draw z1--z2; % diagonal
1340 penlabels(1,2); endchar;
1341

: The : character.
1342
1343 cmchar "Colon";
1344 numeric dot_diam#; dot_diam#=if monospace: 5/4 fi\\ dot_size#;
1345 define_whole_blacker_pixels(dot_diam);
1346 beginchar(":",5u#,x_height#,0);
1347 italcorr x_height#*slant+.5dot_diam#-2u#;
1348 adjust_fit(0,0); pickup fine.nib;
1349 pos1(dot_diam,0); pos2(dot_diam,90);
1350 lft x1=hround(.5w-.5dot_diam); top y2r=h; z1=z2; dot(1,2); % upper dot
1351 pos3(dot_diam,0); pos4(dot_diam,90);
1352 x3=x1; bot y4l=0; z3=z4; dot(3,4); % lower dot
1353 penlabels(1,2,3,4); endchar;
1354

; The ; character.
1355
1356 cmchar "Semicolon";
1357 numeric dot_diam#; dot_diam#=if monospace: 5/4 fi\\ dot_size#;
1358 define_whole_blacker_pixels(dot_diam);
1359 beginchar(";",5u#,x_height#,comma_depth#);
1360 italcorr x_height#*slant+.5dot_diam#-2u#;
1361 adjust_fit(0,0); pickup fine.nib;
1362 pos1(dot_diam,0); pos2(dot_diam,90);
1363 lft x1=hround(.5w-.5dot_diam); top y2r=h; z1=z2; dot(1,2); % upper dot
1364 x3-.5dot_diam=hround(.5w-.5dot_diam); y3-.5dot_diam=0;
1365 comma(3,a,dot_diam,.05u,comma_depth); % lower dot and tail
1366 penlabels(1,2,3); endchar;
1367

```

[The [character.

```

1368
1369 cmchar "Left bracket";
1370 numeric wd#; wd#=max(5u#,4.5u#+.5if hefty:stem# else:rule_thickness# fi);
1371 beginchar("[",wd#,body_height#,paren_depth#);
1372 italcorr body_height#*slant;
1373 adjust_fit(0,0);
1374 numeric top_thickness,side_thickness;
1375 if hefty: top_thickness=vair;
1376 side_thickness=max(crisp.breadth,stem-2stem_corr);
1377 else: top_thickness=side_thickness=rule_thickness; fi;
1378 pickup crisp.nib; pos1(side_thickness,0); pos2(side_thickness,0);
1379 top y1=h; bot y2=-d; lft x1l=lft x2l=hround(2.5u-.5side_thickness)-1-eps;
1380 filldraw stroke z1e--z2e; % stem
1381 pos3(top_thickness,90); pos4(top_thickness,90);
1382 pos5(top_thickness,90); pos6(top_thickness,90);
1383 x3=x5=x1l; rt x4=rt x6=ceiling(w-.4u)+eps; y3r=y4r=y1; y5l=y6l=y2;
1384 filldraw stroke z3e--z4e; % upper bar
1385 filldraw stroke z5e--z6e; % lower bar
1386 penlabels(1,2,3,4,5,6); endchar;
1387

```

] The] character.

```

1388
1389 cmchar "Right bracket";
1390 numeric wd#; wd#=max(5u#,4.5u#+.5if hefty:stem# else:rule_thickness# fi);
1391 beginchar("]",wd#,body_height#,paren_depth#);
1392 italcorr body_height#*slant-2u#+.5if hefty:stem# else:rule_thickness# fi;
1393 adjust_fit(0,0);
1394 numeric top_thickness,side_thickness;
1395 if hefty: top_thickness=vair;
1396 side_thickness=max(crisp.breadth,stem-2stem_corr);
1397 else: top_thickness=side_thickness=rule_thickness; fi;
1398 pickup crisp.nib; pos1(side_thickness,0); pos2(side_thickness,0);
1399 top y1=h; bot y2=-d; rt x1r=rt x2r=hround(w-2.5u+.5side_thickness)+1+eps;
1400 filldraw stroke z1e--z2e; % stem
1401 pos3(top_thickness,90); pos4(top_thickness,90);
1402 pos5(top_thickness,90); pos6(top_thickness,90);
1403 x3=x5=x1r; lft x4=lft x6=floor .4u-eps; y3r=y4r=y1; y5l=y6l=y2;
1404 filldraw stroke z3e--z4e; % upper bar
1405 filldraw stroke z5e--z6e; % lower bar
1406 penlabels(1,2,3,4,5,6); endchar;
1407

```

‘ The ‘ character.

```

1408
1409 cmchar "Reverse apostrophe";
1410 beginchar("'",5u#,asc_height#,0);
1411 italcorr asc_height#*slant+.5dot_size#-2u#;

```

```

1412 adjust_fit(0,0);
1413 x1-.5dot_size=hround(.5w-.5dot_size); y1+.5dot_size=h-comma_depth;
1414 if monospace: ammoc(1,a,dot_size,.28u,vround 1.5comma_depth); % large ammoc
1415 else: ammoc(1,a,dot_size,.25u,comma_depth); fi % normal ammoc
1416 penlabels(1); endchar;
1417
1418
1419 </punct>

```

3.5 The ligatures and dashes file

This is essentially a copy of the Computer Modern comlig.mf file.

```

1420 <*lig>
1421 % CPCLIG.MF This file defines characters common to
1422 %           Pacioli roman and italic text fonts
1423 %           that appear only when |ligs>0|.
1424
1425 First the ligtables.
1426
1426 ligtable "‘": "‘"=:oct"134";
1427 ligtable "’": "’"=:oct"042", "?" kern 2u#, "!" kern 2u#;
1428
1429 ligtable "-": "-"=:oct"173";
1430 ligtable oct"173": "-"=:oct"174";
1431

```

’ The ’ ligature.

```

1432
1433 cmchar "Closing quotes";
1434 beginchar(oct"042",7u#+max(2u#,dot_size#),asc_height#,0);
1435 italcorr asc_height#*slant+dot_size#-4.1u#;
1436 adjust_fit(0,0);
1437 x1-.5dot_size=hround .6u; y2+.5dot_size=h;
1438 x2-x1=hround(1.5u+max(2u#,dot_size#)); y2=y1;
1439 comma(1,a,dot_size,.25u,comma_depth); % left dot and tail
1440 comma(2,b,dot_size,.25u,comma_depth); % right dot and tail
1441 penlabels(1,2); endchar;
1442

```

- The - character.

```

1443
1444 cmchar "Hyphen";
1445 beginchar("-",6u#,x_height#,0);
1446 italcorr .5x_height#*slant-.5u#;
1447 adjust_fit(0,0);
1448 numeric thickness; thickness=if hefty:bar else:.75[hair,stem] fi;
1449 pickup crisp.nib; pos1(thickness,90); pos2(thickness,90);
1450 top y1r=top y2r=vround(.5h+.5thickness); rt x2=hround(w-u)+eps;

```

```

1451 if monospace: x2=w-x1 else: lft x1=hround .2u-eps fi;
1452 filldraw stroke z1e--z2e; % bar
1453 penlabels(1,2); endchar;
1454

“ The “ ligature.

1455
1456 cmchar "Opening quotes";
1457 beginchar(oct"134",7u#+max(2u#,dot_size#),asc_height#,0);
1458 italcorr asc_height#*slant-.1u#;
1459 adjust_fit(0,0);
1460 x2+.5dot_size=hround(w-.6u); y1+.5dot_size=h-comma_depth;
1461 x2-x1=hround(1.5u#+max(2u#,dot_size#)); y2=y1;
1462 ammoc(1,a,dot_size,.25u#,comma_depth); % left dot and tail
1463 ammoc(2,b,dot_size,.25u#,comma_depth); % right dot and tail
1464 penlabels(1,2); endchar;
1465

-- The – ligature. I think that the CM en-dash is too thin for the Pacioli font, so I
   have thickened it a little.

1466
1467 cmchar "En dash";
1468 beginchar(oct"173",9u#,x_height#,0);
1469 italcorr .61803x_height#*slant+.5u#;
1470 adjust_fit(0,0);
1471 %pickup crisp.nib; pos1(vair,90); pos2(vair,90);
1472 pickup crisp.nib; pos1(medium,90); pos2(medium,90);
1473 top y1r=top y2r=vround(.61803h+.5vair); lft x1=-eps; rt x2=w+eps;
1474 filldraw stroke z1e--z2e; % bar
1475 penlabels(1,2); endchar;
1476

--- The — ligature. I think that the CM em-dash is too thin for the Pacioli font, so
    I have thickened it a little.

1477
1478 cmchar "Em dash";
1479 beginchar(oct"174",18u#,x_height#,0);
1480 italcorr .61803x_height#*slant+.5u#;
1481 adjust_fit(letter_fit#,letter_fit#);
1482 %pickup crisp.nib; pos1(vair,90); pos2(vair,90);
1483 pickup crisp.nib; pos1(medium,90); pos2(medium,90);
1484 top y1r=top y2r=vround(.61803h+.5vair); lft x1=-eps; rt x2=w+eps;
1485 filldraw stroke z1e--z2e; % bar
1486 penlabels(1,2); endchar;
1487
1488
1489 </lig>

```

3.6 Extra punctuation

This is an abbreviated copy of romanp.mf.

```
1490 (*xp)
1491 % CPCROMANP.MF This file contains '\&' and '?' in the so-called roman style.
1492
```

& The & character.

```
1493
1494 cmchar "Ampersand";
1495 beginchar("&",14u#,asc_height#,0);
1496 italcorr x_height#*slant-serif_fit# - if serifs: .4u# else: 1.5u# fi;
1497 adjust_fit(0,serif_fit#);
1498 pickup tiny.nib; pos2(slab,-90); x2=4u; bot y2r=-o;
1499 if not hefty:(x,y2l)=whatever[z2r,(w-5u,x_height)]; x2l:=x; fi
1500 if serifs: pos0(fudged.hair,0);
1501 rt x0r+jut=hround(w-.9u); top y0=x_height;
1502 pos1(fudged.hair,0); z1=whatever[z0,(.6[x0,x2],0)];
1503 y1=max(y0-1.5bracket-.2x_height,2/3x_height);
1504 filldraw stroke z0e---z1e...{left}z2e; % short diagonal
1505 else: pickup fine.nib; pos1(.25[slab,flare],-15); rt x1r=hround(w-2u);
1506 y1r=good.y .75[bar_height,x_height]; x1l:=good.x x1l; y1l:=good.y y1l;
1507 top z2'l=(x2l,tiny.top y2l); bot z2'r=(x2r,tiny.bot y2r);
1508 filldraw stroke term.e(2',1,right,1,4); fi % short diagonal and terminal
1509 pickup tiny.nib; numeric slope,theta,reduced_hair;
1510 slope=(h-2vair-slab)/10.5u; theta=angle(-slope,1);
1511 reduced_hair=max(tiny.breadth,hround(fudged.hair if hefty:-2stem_corr fi));
1512 lft x3r=hround .75u; x5=.5[x3r,x6l]; lft x6r=hround .5(w-u);
1513 x3l-x3r=curve-tiny; pos6(reduced_hair,180);
1514 pos5(vair,theta); y5=.5h;
1515 ellipse_set(2l,3l,4l,5l); ellipse_set(2r,3r,4r,5r);
1516 pos7(vair,270); top y7l=h+o; x7=.45[x6r,x8r];
1517 pos8(fudged.stem,30); x8l=good.x(x8l+3.5u-x8); y8r=y6;
1518 ellipse_set(7l,6l,5',5l);
1519 filldraw stroke z2e{left}...z3e{up}...z4e---z5e...{up}z6e
1520 ...z7e{left}...z8e{down}; % bowls
1521 pos10(slab,90); x10=w-3.5u; bot y10l=-o;
1522 pos9(fudged.stem,angle(z8-z10)-90);
1523 z9=.5[z8,z10]+(1.75u,0) rotated(angle(z8-z10)+90);
1524 filldraw stroke z8e{down}...z9e{z10-z8}...{right}z10e; % long diagonal
1525 if serifs: pickup crisp.nib; pos10'(slab,90); z10'=z10;
1526 pos11(fudged.hair,180); rt x11l=hround(w-u); y11=.5bar_height;
1527 filldraw stroke z10'e{right}...{up}z11e; % terminal
1528 numeric inner_jut; if rt x6l+.5u<lft x0l-1.5jut: inner_jut=1.5jut;
1529 else: rt x6l+.5u=lft x0l-inner_jut; fi
1530 dish_serif(0,1,a,.6,inner_jut,b,.5,jut)(dark); % serif
1531 else: pickup fine.nib; pos10'(slab,90); z10'=z10;
1532 pos11(Vround .5[slab,flare],90);
1533 rt x1l=hround(r-letter_fit-u); bot y11l=vround .07bar_height-o;
1534 filldraw stroke term.e(10',11,right,1,4); fi % terminal
```

```

1535 penlabels(0,1,2,3,4,5,6,7,8,9,10,11); endchar;
1536
? The ? character. I have changed this slightly because of the redefinition of the
  pulled_super_arc macro.
1537
1538 cmchar "Question mark";
1539 beginchar("?",8.5u#,asc_height#,0);
1540 italcorr .8asc_height#*slant;
1541 adjust_fit(0,0);
1542 pickup tiny.nib; pos7(dot_size,0); pos8(dot_size,90);
1543 lft x7l=hround(.5w-.25u-.5dot_size); bot y8l=0; z7=z8; dot(7,8); % dot
1544 numeric bot_width;
1545 bot_width=if hefty:max(hround .8dot_size,fine.breadth) else: hair fi;
1546 pickup fine.nib; pos2(vair,90); pos3(curve,0);
1547 pos4(vair,-90); pos5(bot_width,0); pos6(bot_width,0);
1548 x2=x4=x5=x6=x7; rt x3r=hround(w-u); bot y6=1+.25[top y8r,x_height];
1549 top y2r=h+oo; y3=.75[y6,y2]; y4=.5[y6,y2]; y5=.1[y6,y2];
1550 {{interim superness:=more_super;
1551 %%filldraw stroke pulled_super_arc.e(2,3)(superpull) % macro changed in cpctitle.mf
1552 filldraw stroke pulled_super_arc.e(2,3,eta)
1553 & subpath (0,1) of super_arc.e(3,4) .. z5e---z6e\\}}; % arc and stem
1554 if serifs: pos1(hair,180); pos0(flare,180);
1555 lft x1r=hround u; y1=y3; bulb(2,1,0); % bulb
1556 else:
1557 pos1(Vround 5/7[vair,flare],110);
1558 lft x1r=hround u; top y1r=vround .9[y6,top y2r];
1559 filldraw stroke term.e(2,1,left,1,4);
1560 fi % terminal
1561 penlabels(0,1,2,3,4,5,6,7,8); endchar;
1562
1563
1564 </xp>

```

4 The font definition files

The font definition file for the OT1 encoding.

```

1565 <*fdot1>
1566 \DeclareFontFamily{OT1}{cpc}{ }
1567 \DeclareFontShape{OT1}{cpc}{m}{n}{ <-> cpcr10 }{}
1568 \DeclareFontShape{OT1}{cpc}{bx}{n}{ <-> sub * cpc/m/n }{}
1569 \DeclareFontShape{OT1}{cpc}{b}{n}{ <-> sub * cpc/m/n }{}
1570 \DeclareFontShape{OT1}{cpc}{m}{sl}{ <-> cpsl10 }{}
1571 \DeclareFontShape{OT1}{cpc}{m}{it}{ <-> sub * cpc/m/sl }{}
1572 </fdot1>

```

The font definition file for the T1 encoding.

```

1573 <*fdt1>
1574 \DeclareFontFamily{T1}{cpc}{ }

```

```

1575 \DeclareFontShape{T1}{cpc}{m}{n}{ <-> cpcr10 }{}
1576 \DeclareFontShape{T1}{cpc}{bx}{n}{ <-> sub * cpc/m/n }{}
1577 \DeclareFontShape{T1}{cpc}{b}{n}{ <-> sub * cpc/m/n }{}
1578 \DeclareFontShape{T1}{cpc}{m}{sl}{ <-> cpcsl10 }{}
1579 \DeclareFontShape{T1}{cpc}{m}{it}{ <-> sub * cpc/m/sl }{}
1580 </fdtl>

```

5 The pacioli package code

Announce the name and version of the package, which requires L^AT_EX 2_ε.

```

1581 <*usc>
1582 \NeedsTeXFormat{LaTeX2e}
1583 \ProvidesPackage{pacioli}[1999/03/14 v1.0 package for Pacioli fonts]

```

`\cpcfamly` Selects the Pacioli font family using the OT1 encoding.

```

1584 \newcommand{\cpcfamly}{\usefont{OT1}{cpc}{m}{n}}

```

`\textcpc` Text command for the Pacioli font family.

```

1585 \DeclareTextFontCommand{\textcpc}{\cpcfamly}

```

The end of this package.

```

1586 </usc>

```

References

[GMS94] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley Publishing Company, 1994.

[Mor94] Stanley Morison. *Pacioli's Classic Roman Alphabet*. Dover, 1994.

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
<code>\!</code>	<u>1261</u>	<code>\---</code>	<u>1477</u>
<code>\&</code>	1491, <u>1493</u>	<code>\.</code>	<u>1322</u>
<code>\'</code>	<u>1278</u>	<code>\/</code>	<u>1332</u>
<code>\''</code>	<u>1432</u>	<code>\:</code>	<u>1342</u>
<code>\(</code>	<u>1288</u>	<code>\;</code>	<u>1355</u>
<code>\)</code>	<u>1300</u>	<code>\?</code>	<u>1537</u>
<code>\,</code>	<u>1312</u>	<code>\[</code>	<u>1368</u>
<code>\-</code>	<u>1443</u>	<code>\]</code>	<u>1388</u>
<code>\--</code>	<u>1466</u>	<code>\'</code>	<u>1408</u>
		<code>\''</code>	<u>1455</u>
			B
		<code>\B</code>	<u>348</u>

<code>\bar-height</code>	<u>17</u>	<code>\hair</code>	<u>57</u>	S	
<code>\beak-darkness</code>	<u>65</u>	<code>\half-height</code>	<u>17</u>	<code>\S</code>	<u>1014</u>
<code>\beak-jut</code>	<u>41</u>	<code>\hefty</code>	<u>74</u>	<code>\serif-darkness</code> . . .	<u>34</u>
<code>\body-height</code>	<u>17</u>	<code>\Hrad</code>	<u>51</u>	<code>\serif-fit</code>	<u>10</u>
<code>\bracket</code>	<u>41</u>	<code>\hstretch</code>	<u>10</u>	<code>\serifs</code>	<u>71</u>
C		I		<code>\side</code>	<u>28</u>
<code>\C</code>	<u>413</u>	<code>\I</code>	<u>662</u>	<code>\slab</code>	<u>41</u>
<code>\c</code>	<u>377</u>	J		<code>\slant</code>	<u>63</u>
<code>\cap-height</code>	<u>17</u>	<code>\J</code>	<u>678</u>	<code>\slim-font-setup</code> . .	<u>177</u>
<code>\cap-jut</code>	<u>41</u>	<code>\jut</code>	<u>41</u>	<code>\square-dots</code>	<u>74</u>
<code>\cap-serif-fit</code>	<u>10</u>	K		<code>\srad</code>	<u>51</u>
<code>\cap-stem</code>	<u>41</u>	<code>\K</code>	<u>691</u>	<code>\stem</code>	<u>41</u>
<code>\cm-arm</code>	<u>131</u>	L		<code>\stem-corr</code>	<u>79</u>
<code>\cm-font-setup</code>	<u>177</u>	<code>\L</code>	<u>735</u>	<code>\superness</code>	<u>65</u>
<code>\comma-depth</code>	<u>17</u>	<code>\letter-fit</code>	<u>10</u>	<code>\superpull</code>	<u>65</u>
<code>\cpcfamily</code> 2, <u>1584</u> , <u>1585</u>		<code>\ligs</code>	<u>74</u>	T	
<code>\crisp</code>	<u>34</u>	<code>\Lrad</code>	<u>51</u>	<code>\T</code>	<u>1046</u>
<code>\curve</code>	<u>79</u>	<code>\lrad</code>	<u>51</u>	<code>\textcpc</code>	2, <u>1585</u>
D		M		<code>\thick</code>	<u>28</u>
<code>\D</code>	<u>449</u>	<code>\M</code>	<u>758</u>	<code>\thickfudge</code>	<u>28</u>
<code>\DeclareFontFamily</code> .		<code>\makebox</code>	<u>100</u>	<code>\thin</code>	<u>28</u>
.	1566, 1574	<code>\math-axis</code>	<u>17</u>	<code>\thinfudge</code>	<u>28</u>
<code>\DeclareFontShape</code> .		<code>\medium</code>	<u>28</u>	<code>\tiny</code>	<u>34</u>
.	1567–	<code>\mediumfudge</code>	<u>28</u>	<code>\trad</code>	<u>51</u>
1571, 1575–1579		<code>\monospace</code>	<u>71</u>	U	
<code>\DeclareTextFontCommand</code>		N		<code>\U</code>	<u>1068</u>
.	1585	<code>\N</code>	<u>788</u>	<code>\u</code>	<u>10</u>
<code>\Delta</code>	63, 64	O		V	
<code>\desc-depth</code>	<u>17</u>	<code>\O</code>	<u>846</u>	<code>\V</code>	<u>1085</u>
<code>\dish</code>	<u>41</u>	<code>\o</code>	<u>59</u> , <u>813</u>	<code>\vair</code>	<u>57</u>
<code>\dot-size</code>	<u>74</u>	P		<code>\vstretch</code>	<u>10</u>
<code>\dslab</code>	<u>41</u>	<code>\P</code>	<u>879</u>	W	
E		<code>\pacioli-font-setup</code> <u>234</u>		<code>\W</code>	<u>1110</u>
<code>\E</code>	<u>474</u>	<code>\proofpcbb</code>	<u>111</u>	<code>\whole-font</code>	<u>83</u>
<code>\eta</code>	34, <u>34</u>	<code>\ProvidesPackage</code> .	1583	<code>\width-adj</code>	<u>10</u>
F		<code>\pulled-arc</code>	<u>159</u>	X	
<code>\F</code>	<u>505</u>	<code>\pulled-super-arc</code> .	<u>159</u>	<code>\X</code>	<u>1159</u>
<code>\fig-height</code>	<u>17</u>	Q		<code>\x-height</code>	<u>17</u>
<code>\fine</code>	<u>34</u>	<code>\Q</code>	<u>938</u>	Y	
<code>\flare</code>	<u>79</u>	<code>\q</code>	<u>900</u>	<code>\Y</code>	<u>1190</u>
<code>\foot</code>	<u>145</u>	R		Z	
<code>\fudge</code>	<u>65</u>	<code>\R</code>	<u>976</u>	<code>\Z</code>	<u>1228</u>
G		<code>\rule-thickness</code> . . .	<u>34</u>		
<code>\G</code>	<u>586</u>				
<code>\g</code>	<u>530</u>				
H					
<code>\H</code>	<u>642</u>				