

# The bidi package

François Charette

<firmicus@gmx.vet>

2008/07/07      v0.3

## Contents

<b>1 Basic direction switching commands</b>	<b>1</b>	5.1 Basic bidi macros . . .	5
1.1 Direction of plain footnotes . . . . .	2	5.2 Logos . . . . .	6
<b>2 Colored text</b>	<b>2</b>	5.3 Redefinition of various internal L <sup>A</sup> T <sub>E</sub> X macros .	7
<b>3 Scripts written from right-to-left</b>	<b>3</b>	5.4 Bidi switching commands	10
<b>4 Acknowledgements</b>	<b>4</b>	5.5 Footnotes (cont.) . . . .	10
<b>5 Implementation</b>	<b>4</b>	5.6 Captions . . . . .	11
		5.7 \textcolor . . . . .	14
		5.8 Tabular environments .	15
		5.9 Equation labels in RL documents . . . . .	17
		5.10 TODO . . . . .	17

## 1 Basic direction switching commands

The bidi package provides a useful interface for typesetting bidirectional texts with X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X.<sup>1</sup> Most of its macros are directly copied or adapted from the Babel bidirectional module for Hebrew (by Boris Lavva).<sup>2</sup>

<sup>1</sup>Actually, bidi can also be used with any  $\epsilon$ -T<sub>E</sub>X-based engine, notably pdfL<sup>A</sup>T<sub>E</sub>X, but at this time I do not see how it could be useful outside of the X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X world.

<sup>2</sup>See section 60.4 of the [Babel user documentation](#).

`\RL` For short right-to-left insertions in left-to-right environments, `\RL{...}`  
`\LR` does the job, as does `\LR{...}` in the opposite case. To typeset a full paragraph  
`\setRL` right-to-left, use the switch `\setRL`, and to revert to left-to-right, use `\setLR`  
`\setLR` (or equivalently `\unsetRL`).

```
\setRL
This is a right-to-left paragraph.
\setLR
```

<sup>3</sup>paragraph. right-to-left a is This

## 1.1 Direction of plain footnotes

`\setfootnoteRL` With `\setfootnoteRL`, one can insert RL footnotes in LR mode,<sup>4</sup> and con-  
`\setfootnoteLR` versely with `\setfootnoteLR` (or equivalently `\unsetfootnoteRL`).

With the package `manyfoot`, it is possible to have several layers of foot-  
notes, each of which can be set in a different direction. See the documentation  
to `ArabXYTeX` for more details. This should also be possible with `bigfoot`.

Some other  $\text{\LaTeX}$  macros are redefined internally to ensure their correct  
behaviour in bidirectional display. Contrary to the Babel approach, we leave  
the commands defined in standard  $\text{\LaTeX}$  classes untouched, as we think this  
should be done by class authors themselves (who are of course encouraged to  
make them bidi-compatible :-).

## 2 Colored text

Because of the way bidirectionality is implemented in  $\varepsilon\text{-TeX}$ , no information  
on directionality is included in the `dvi` (or `xdv`) output. Consequently, if a  
color `\special` is inserted in the middle of a paragraph, the color change will  
be in effect as though the text would run from left to right. For `\textcolor`,  
the output will be correct only if the textual argument falls on a single line. A

<sup>3</sup>Note that when processing fonts that are not TFM-based, `XYTeX`, contrary to `TeX`,  
uses (roughly speaking) one box per word, instead of one per character. The command `\RL`  
is thus usually superfluous when inserting a single word.

<sup>4</sup>ملاحظات من اليمين إلى البصار

better approach is to set the color of text by means of X<sub>Y</sub>TeX's font features, as in this example:

```
\newcommand{\textblue}[1]{\addfontfeature{Color=0000FF}#1}
\setRL
{\arabicfont
لِسود لِسود لِسود لِسود لِسود لِسود لِسود لِسود لِسود لِسود
لِسود لِسود \textblue{ازرق ازرق ازرق ازرق ازرق ازرق ازرق ازرق ازرق ازرق}
لِسود لِسود لِسود لِسود
}
اسود اسود اسود اسود اسود اسود اسود اسود اسود اسود اسود
ازرق ازرق ازرق ازرق ازرق ازرق اسود اسود اسود اسود اسود اسود
```

### 3 Scripts written from right-to-left

The Arabic and Hebrew scripts are of course not the only ones to be written from right-to-left. The following is a list of historical<sup>5</sup> and contemporary scripts that are written primarily or exclusively from right-to-left.<sup>6</sup> Those still in use today are marked with an asterisk.

Aramaic and Sogdian	Nabatean
Arabic*	N'Ko*
Avestan	Old Hungarian
Berber (ancient)	Orkhon (Turkic runes)
Cypriot Syllabary	Pahlavi and Parthian
Etruscan and Oscan	Phoenician
Hebrew*	Proto-Sinaitic
Iberian (Northern)	Samaritan
Kharosthi	South Arabian and Thamudic
Lydian	Syriac*
Mandaean	Thâna*
Meroitic (cursive)	

<sup>5</sup>The historical right-to-left scripts that are part of Unicode 5.0 are: Cypriot, Old Italic (i.e., Etruscan), Kharosthi and Phoenician.

<sup>6</sup>See <http://www.omniglot.com/writing/direction.htm#rtl>, <http://www.unicode.org/iso15924/>, and <http://www.ancientscripts.com/ws.html>.

• Old Italic (Etruscan): <sup>7</sup>	
• N'ko: <sup>8</sup>	
• Thâna:	
• Phoenician:	
• Kharosthi:	
• Cypriot:	

Thanks to Boris Lavva (author of `rlbabel.def`), the authors of FarsiT<sub>E</sub>X, Youssef Jabri [يوسف جبري] (author of `Arabi`), and Vafa Khalighi [وفا خلیقی] (author of `xepersian`).

```

1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{bidi}
3 [2008/07/07 v0.3 Bidirectional typesetting in XeLaTeX]
4 \newif\if@rlmain
5 \@rlmainfalse
6 \DeclareOption{rldocument}{\@rlmaintrue}
7 \ProcessOptions
8 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
9 \ifx\TeXeTstate\undefined\else%
10   \TeXeTstate=1
11 \fi
12 \ifx\beginL\@undefined%

```

<sup>8</sup>As can be seen in this example, the ICU rendering engine does not yet support contextual substitutions for N’Ko.

```

13 \newlinechar`^^J
14 \typeout{^^JTo avoid this error message,^^J%
15     run TeX--XeT or e-TeX engine instead of regular TeX.^^J}
16 \errmessage{Right-to-Left Support Error: use TeX--XeT or e-TeX
17     engine}%
18 \fi
19 \newif\if@rl
20 \AtBeginDocument{% Here we set the main document direction
21     \if@rlmain\else
22         \if@rl\@rlmaintrue%
23         \else\@rlmainfalse%
24         \fi%
25     \fi%
26 }

```

## 5.1 Basic bidi macros

This defines the macros for RL insertion in LR context and conversely. Most of it is taken from the file `rlbabel.def` in the Babel distribution, with minor changes.

```

27 \let\n@xt=\
28 \def\LR{\protect\pLR}%
29 \def\pLR{\protect\afterassignment\moreLR \let\n@xt= }
30 \def\moreLR{\bracetext \aftergroup\endL \beginL\@rlfalse}
31 \def\RL{\protect\pRL}
32 \def\pRL{\protect\afterassignment\moreRL \let\n@xt= }
33 \def\moreRL{\bracetext \aftergroup\endR \beginR\@rltrue}
34 \def\bracetext{\ifcat\n@xt{\else\ifcat\n@xt}\fi
35     \errmessage{Missing left brace has been substituted}\fi \bgroup}
36 \everydisplay{\if@rl\aftergroup\beginR\fi }

```

For global right-to-left typesetting, the directional primitive must be included within an `\everypar` construct. The traditional approach was to re-define `\everypar` as a one-argument macro, but since some packages such as `ledmac` also play tricks with `\everypar`, this can become problematic. Hence we have adopted Youssef Jabri's approach in his Arabi package (شكراً جزيلاً).

```

37 \def\rl@everypar{\if@rl{\setbox\z@\lastbox\beginR\usebox\z@}\fi}
38 \let\o@everypar=\everypar

```

39 %The following, surely, will not play well with all packages, e.g. led-  
mac

```
40 %\def\everypar#1{\o@everypar{\rl@everypar#1}}
41 % But we now use this instead, taken from Arabicore.sty
42 \newtoks\n@everypar
43 \n@everypar\expandafter{\the\n@everypar}
44 \o@everypar{\rl@everypar\the\n@everypar}
45 \let\everypar=\n@everypar
```

This is similar to `\ensuremath` in  $\text{\LaTeX}$ :

```
46 \def\@ensure@RL#1{\if@rl#1\else\RL{#1}\fi}
47 \def\@ensure@LR#1{\if@rl\LR{#1}\else#1\fi}
```

This does the opposite (which is useful for ensuring correct directionality in boxes or tabular environments, for instance):

```
48 \def\@ensure@dir#1{\if@rl\RL{#1}\else{#1}\fi}
49 \def\@ensure@maindir#1{\if@rlmain\RL{#1}\else{#1}\fi}
```

These are not yet used...

```
50 \def\@@RL{RL}
51 \def\@@LR{LR}
52 \def\save@dir{\if@rl\gdef\save@@dir{RL}\else\gdef\save@@dir{LR}\fi}
53 \def\reset@dir{\ifx\save@@dir\@@RL\setRL\else\ifx\save@@dir\@@LR\setLR\else\relax\fi\fi}
```

## 5.2 Logos

The following redefinitions ensure proper typesetting of the various logos.

```
54 %preserve logos
55 \let\@@TeX\TeX
56 \def\TeX{\@ensure@LR{\@@TeX}}
57 \let\@@LaTeX\LaTeX
58 \def\LaTeX{\@ensure@LR{\@@LaTeX}}
59 \let\@@LaTeXe\LaTeXe
60 \def\LaTeXe{\@ensure@LR{\@@LaTeXe}}
61 \ifpackageloaded{xltextra}{
62 \let\@@XeTeX\XeTeX
63 \def\XeTeX{\@ensure@LR{\@@XeTeX}}
64 \let\@@XeLaTeX\XeLaTeX
65 \def\XeLaTeX{\@ensure@LR{\@@XeLaTeX}}
66 }{}
```

## 5.3 Redefinition of various internal L<sup>A</sup>T<sub>E</sub>X macros

### 5.3.1 `\text`

`\text` is defined in `amstext.sty`.

```
67 \ifundefined{text}{}{%  
68 \let\t@xt=\text  
69 \def\text#1{\t@xt{\@ensure@dir{#1}}}}
```

### 5.3.2 `\list`

```
70 \def\list#1#2{%  
71   \ifnum \@listdepth >5\relax  
72     \@toodeep  
73   \else  
74     \global\advance\@listdepth\@ne  
75   \fi  
76   \rightmargin\z@  
77   \listparindent\z@  
78   \itemindent\z@  
79   \csname @listromannumeral\the\@listdepth\endcsname  
80   \def\@itemlabel{#1}%  
81   \let\makelabel\@mklab  
82   \@nmblistfalse  
83   #2\relax  
84   \@trivlist  
85   \parskip\parsep  
86   \parindent\listparindent  
87   \advance\linewidth -\rightmargin  
88   \advance\linewidth -\leftmargin  
89   \if@rl  
90     \advance\@totalleftmargin \rightmargin  
91   \else  
92     \advance\@totalleftmargin \leftmargin  
93   \fi  
94   \parshape \@ne \@totalleftmargin \linewidth  
95   \ignorespaces}
```

### 5.3.3 Dotted line in table of contents

```

96 \def\@dottedtocline#1#2#3#4#5{%
97   \ifnum #1>\c@tocdepth \else
98     \vskip \z@ \@plus.2\p@
99     {\if@r\rightskip\else\leftskip\fi #2\relax
100      \if@r\leftskip\else\rightskip\fi \@tocrmarg \parfillskip
101      -\if@r\leftskip\else\rightskip\fi
102      \parindent #2\relax\@afterindenttrue
103      \interlinepenalty\@M
104      \leavevmode
105      \@tempdima #3\relax
106      \advance\if@r\rightskip\else\leftskip\fi \@tempdima
107      \null\nobreak\hskip -\if@r\rightskip\else\leftskip\fi
108      {#4}\nobreak
109      \leaders\hbox{$\m@th
110        \mkern \@dotsep mu\hbox{.}\mkern \@dotsep
111        mu$}\hfill
112      \nobreak
113      \hb@xt@\@pnumwidth{\hfil\normalfont \normalcolor \beginL#5\endL}%
114      \par}%
115 \fi}

```

### 5.3.4 \raggedright and \raggedleft

```

116 \def\raggedright{%
117   \let\\\@centercr
118   \leftskip\z@skip\rightskip\@flushglue
119   \parindent\z@\parfillskip\z@skip}
120 \let\@raggedleft=\raggedleft
121 \let\@raggedright=\raggedright
122 \renewcommand\raggedleft{\if@r\@raggedright%
123   \else\@raggedleft\fi}
124 \renewcommand\raggedright{\if@r\@raggedleft%
125   \else\@raggedright\fi}

```

### 5.3.5 \underline

```

126 \let\@@underline=\underline
127 \def\underline#1{\@@underline{\if@r\RL{#1}\else #1\fi}}
128 \if@compatibility
129   \let\undertext=\underline

```

130 \fi

### 5.3.6 \@outputdblcol

```
131 \let\lr@outputdblcol\@outputdblcol
132 \renewcommand{\@outputdblcol}{%
133   \if@rlmain%
134     \rl@outputdblcol%
135   \else%
136     \lr@outputdblcol%
137   \fi%
138 }
139 \newcommand{\rl@outputdblcol}{%
140   \if@firstcolumn
141     \global \@firstcolumnfalse
142     \global \setbox\@leftcolumn \box\@outputbox
143   \else
144     \global \@firstcolumntrue
145     \setbox\@outputbox \vbox {\hb@xt@\textwidth {%
146                                   \hskip\columnwidth%
147                                   \hfil\vrule\@width\columnseprule\hfil
148                                   \hb@xt@\columnwidth {%
149                                     \box\@leftcolumn \hss}%
150                                   \hb@xt@\columnwidth {%
151                                     \hskip-\textwidth%
152                                     \box\@outputbox \hss}%
153                                   \hskip\columnsep%
154                                   \hskip\columnwidth}}}%
155   \@combinedblfloats
156   \@outputpage
157   \begingroup
158     \@dblfloatplacement
159     \@startdblcolumn
160     \@whiles\if@fcolmade \fi
161     {\@outputpage
162      \@startdblcolumn}%
163   \endgroup
164 \fi}
165
```

```

166 % \subsection{Footnotes}
167 % This implements the bidirectionality of plain \LaTeX\ footnotes.
168 % \begin{macrocode}
169 \newif\if@rl@footnote
170 \if@rlmain\@rl@footnotetrue\else\@rl@footnotefalse\fi
171 \let\@@footnoterule=\footnoterule
172 \def\LRfootnoterule{\@@footnoterule}
173 %%bug?-->this causes new par to be set, hence hack with vskip :-{
174 \def\RLfootnoterule{\vskip -\baselineskip\hb@xt@\hsize{\hss\vbox{\@@footnoterule}}}}
175 %
176 \def\setfootnoteRL{\@rl@footnotetrue}
177 \def\unsetfootnoteRL{\@rl@footnotefalse}
178 \def\setfootnoteLR{\unsetfootnoteRL}
179 %
180 \def\footnoterule{\if@rlmain\RLfootnoterule\else\LRfootnoterule\fi}

```

## 5.4 Bidi switching commands

Commands for switching between LR and RL global contexts.

```

181 \def\setRL{\@rltrue\@rl@footnotetrue}
182 \def\unsetRL{\@rlfalse\@rl@footnotefalse}
183 \def\setLR{\unsetRL}

```

## 5.5 Footnotes (cont.)

This redefines L<sup>A</sup>T<sub>E</sub>X's internal footnote macros.

```

184 \long\def\@footnotetext#1{\insert\footins{%
185   \if@rl@footnote\@rltrue\else\@rlfalse\fi
186   \reset@font\footnotesize
187   \interlinepenalty\interfootnotelinepenalty
188   \splittopskip\footnotesep
189   \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
190   \hsize\columnwidth \@parboxrestore
191   \protected@edef\@currentlabel{%
192     \csname p@footnote\endcsname\@thefnmark
193   }%
194   \color@begingroup
195   \@makefnmark{\@footnotetext{#1}}

```

```

196      \rule\z@\footnotesep\ignorespaces#1\@finalstrut\strutbox}%
197      \color@endgroup}}%
198 %... also in minipages
199 \long\def\@mpfootnotetext#1{%
200   \global\setbox\@mpfootins\vbox{%
201     \if@rl@footnote\@rltrue\else\@rlfalse\fi
202     \unvbox\@mpfootins
203     \reset@font\footnotesize
204     \hsize\columnwidth
205     \@parboxrestore
206     \protected@edef\@currentlabel
207       {\csname p@mpfootnote\endcsname\@thefnmark}%
208     \color@begingroup
209     \@makefntext{%
210       \rule\z@\footnotesep\ignorespaces#1\@finalstrut\strutbox}%
211     \color@endgroup
212   }}

```

## 5.6 Captions

This redefines the macros responsible for captions in various popular classes (article, book, report, scrartcl, scrbook, screprt, and memoir).

```

213 \@ifundefined{KOMAClassName}{%
214 \@ifclassloaded{memoir}{%
215 \long\def\@makecaption#1#2{\let\@memtempa\relax
216   \ifdim\prevdepth>-99\p@ \vskip\abovecaptionskip
217   \else \def\@memtempa{\vbox to\topskip{}}\fi
218   \let\@contfnote\footnote \renewcommand{\footnote}[2]{}
219   \let\@contfmark\footnotemark \renewcommand{\footnotemark}[1]{}
220   \sbox\@tempboxa{\@contnfont #1\@contdelim \@conttfont #2}
221   \let\footnote\@contfnote
222   \let\footnotemark\@contfmark
223   \ifdim\wd\@tempboxa<\linewidth \centering \fi
224   \if@contcw
225     \centering
226     \parbox{\@contcwidth}{%
227       \ifdim\wd\@tempboxa<\@contcwidth \centering \fi
228     \fi

```

```

229 \if@conthang
230   \sbox\@tempboxa{\@contnfont #1\@contdelim}
231   \@contpre%
232   \@ensure@dir{\@contcstyle\hangindent=\wd\@tempboxa
233     \noindent\box\@tempboxa\@memtempa \@conttfont #2}\par
234 \else
235   \if@contindent
236     \@contpre%
237     \@ensure@dir{\@contnfont #1\@contdelim}\@memtempa
238     \@ensure@dir{\@contcstyle\hangindent=\@contindw
239       \hangafter=\@ne\@conttfont #2}\par% <- v1.4
240   \else
241     \@contpre%
242     \@ensure@dir{\@contnfont #1\@contdelim}\@memtempa
243     \@ensure@dir{\ifdim\wd\@tempboxa<\linewidth
244       \@contcshortstyle\else \@contcstyle\fi% <- v1.4
245   \@conttfont #2}\par
246   \fi
247   \fi
248   \@contpost
249   \if@contcw
250     \par
251   } % end of the \parbox
252   \fi
253   \vskip\belowcaptionskip}
254}% else we redefine \@makecaption for the standard classes
255{\@ifundefined{@makecaption}{%
256%warning ??
257}{\long\def\@makecaption#1#2{%
258  \vskip\abovecaptionskip
259  \sbox\@tempboxa{\@ensure@dir{#1: #2}}}%
260  \ifdim \wd\@tempboxa >\hsize
261    \@ensure@dir{#1: #2}\par
262  \else
263    \global \@minipagefalse
264    \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
265  \fi
266  \vskip\belowcaptionskip}%

```

```

267 }%
268 }%
269 }% ELSE KOMASCRIP
270 {\@ifclassloaded{scr1tr2}}}% do nothing for letter class
271 {\renewcommand{\@makecaption}[3]{%
272   \cap@margin
273   \ifonelinecaptions
274     \@tempcnta=\value{mpfootnote}\@tempcntb=\value{footnote}%
275     \let\scr@tmp@footnotetext=\@footnotetext
276     \let\@footnotetext=\@gobble
277     \sbox\@tempboxa{%
278       {\capfont\strut\ignorespaces
279         #1{\caplabelfont#2\captionformat}}}%
280       #3\unskip\strut}%
281   }%
282   \let\@footnotetext=\scr@tmp@footnotetext
283   \let\scr@tmp@footnotetext=\undefined
284   \setcounter{footnote}{\@tempcntb}%
285   \setcounter{mpfootnote}{\@tempcnta}%
286 \else
287   \sbox\@tempboxa{\hspace*{2\cap@width}}%
288 \fi
289 \begingroup%
290   \setlength{\@tempdima}{\cap@width}%
291   \ifdim \wd\@tempboxa >\@tempdima
292     \hb@xt@\hsize{%
293       \setbox\@tempboxa\vbox{\hsize=\cap@width
294         \ifdofullc@p
295           \@ensure@dir{\capfont\@hangfrom{\scr@smashdp{\strut\ignorespaces
296             #1{\caplabelfont#2\captionformat}}}%
297             {#3\unskip\strut\par}}}%
298         \else%
299           \ifdim\cap@indent<\z@
300             \@ensure@dir{\capfont\strut\ignorespaces
301               #1{\caplabelfont #2\captionformat\par}}%
302               \noindent\hspace*{-\cap@indent}#3\unskip\strut\par}%
303             \else%
304               \if@capbreak

```

```

305         \@ensure@dir{\capfont\strut\ignorespaces
306         #1{\caplabelfont #2\captionformat\par}}}%
307         \noindent\@hangfrom{\hspace*{\cap@indent}}{\#3\par}}%
308     \else
309     \@ensure@dir{\capfont\@hangfrom{\scr@smashdp\strut\hspace*{\cap@indent}}%
310     {\hspace{-\cap@indent}\scr@smashdp{\ignorespaces#1%
311     {\caplabelfont#2\captionformat}}}\#3\unskip\strut\par}}%
312     \fi
313 \fi
314 \fi
315 }%
316 \setlength{\@tempdima}{\ht\@tempboxa}%
317 \addtolength{\@tempdima}{\dp\@tempboxa}%
318 \addtolength{\@tempdima}{-\ht\strutbox}%
319 \ht\@tempboxa\ht\strutbox
320 \dp\@tempboxa\@tempdima
321 \strut\cap@left
322 \box\@tempboxa
323 \cap@right\strut
324 }%
325 \else
326     \global \@minipagefalse
327     \sbox\@tempboxa{%
328         \@ensure@dir{\capfont\scr@smashdp{\strut\ignorespaces
329         #1{\caplabelfont#2\captionformat}}}%
330         #3}%
331     }%
332     \hb@xt@\hsize{\strut\cap@left\box\@tempboxa\cap@right\strut}%
333 \fi
334 \endgroup}
335 }%

```

## 5.7 \textcolor

\textcolor needs to be redefined so that the color-switching commands in the output will affect the correct portion of text also in RL context, but this only works if the textual argument does not overlap more than one line, which is why we issue a warning. See § 2 above for an alternative approach to colored

text using X<sub>Y</sub>T<sub>E</sub>X font features. The following code (less the warning) is again taken from `rlbabel.def`.

```

336 \ifx\@textcolor\undefined\else%
337 \AtBeginDocument{%
338   \def\@textcolor#1#2#3{%
339     \if@rl\PackageWarning{bidi}{%
340       \textcolor will not work in RL mode if the textual argu-
        ment spans more than one line. ^^J%
341       With XeTeX you should set color as a font feature instead.}%
342     \beginL\protect\leavevmode{\color#1{#2}\beginR#3\endR}\endL%
343     \else%
344       \protect\leavevmode{\color#1{#2}#3}%
345     \fi%
346   }%
347 }%
348 \fi

```

## 5.8 Tabular environments

This is taken almost verbatim from FarsiT<sub>E</sub>X, with the addition of a patch for the array package (which does not work correctly yet).

```

349 \newif\if@tab@rl
350 \if@rlmain\@tab@rltrue\else\@tab@rlfalse\fi
351 \@ifpackageloaded{array}%
352 {\PackageWarningNoLine{bidi}{%
353   You are using the package array (or the memoir class). ^^J
354   Unfortunately right-to-left tabular environments are not ^^J
355   yet supported with the array package.}
356 \let\@author=\author
357 \def\author#1{\@author{\@ensure@dir{#1}}}
358 }%
359 %\def\@array[#1]#2{%
360 %  \@tempdima \ht \strutbox
361 %  \advance \@tempdima by\extrarowheight
362 %  \setbox \@arstrutbox \hbox{\vrule
363 %    \@height \arraystretch \@tempdima
364 %    \@depth \arraystretch \dp \strutbox
365 %    \@width \z@}%

```

```

366 % \begingroup
367 % \@mkpream{#2}%
368 % \xdef\@preamble{\noexpand \ialign \@halignto
369 % \bgroup \@arstrut \@preamble
370 % \tabskip \z@ \cr}%
371 % \endgroup
372 % \@arrayleft
373 % \if #1t\vtop \else \if#1b\vbox \else \vcenter \fi \fi
374 % \bgroup
375 % \let \@sharp ##\let \protect \relax
376 % \lineskip \z@
377 % \baselineskip \z@
378 % \if@tab@rl\hbox\bgroup\beginR\vbox\bgroup\fi
379 % \m@th
380 % \let\\\@arraycr \let\tabularnewline\\\let\par\@empty
381 % \@preamble}
382 %\def\@tabular{%
383 % \if@rl\global\@tab@rltrue\fi
384 % \leavevmode
385 % \hbox \bgroup\if@tab@rl\beginR\fi
386 % $\col@sep\tabcolsep
387 % \let\d@llarbegin\begingroup \let\d@llarend\endgroup
388 % \@tabarray}
389 %\def\endtabular{\endarray\if@tab@rl\egroup\endR\egroup\fi
390 %$\if@tab@rl\endR\fi
391 %\egroup}%
392 {% ELSE DEFAULT LATEX MACROS
393 \def\@tabular{%
394 \if@rl\global\@tab@rltrue\fi
395 \leavevmode \hbox \bgroup \if@tab@rl\beginR\fi
396 $\let\@acol\@tabacol
397 \let\@classz\@tabclassz
398 \let\@classiv\@tabclassiv \let\\\@tabularcr\@tabarray}
399 \def\endtabular{\crrcr\egroup\if@tab@rl\egroup\endR\egroup\fi
400 \egroup $\if@tab@rl\endR\fi \egroup}%
401 \def\@array[#1]#2{\setbox\@arstrutbox=\hbox{\vrule
402 height\arraystretch \ht\strutbox
403 depth\arraystretch \dp\strutbox

```

```

404      width\z@}\@mkpream{#2}\edef\@preamble{%
405 \halign \noexpand\@halignto
406 \bgroup \tabskip\z@ \@arstrut \@preamble \tabskip\z@ \cr}%
407 \let\@startpbox\@startpbox \let\@endpbox\@endpbox
408 \if #1t\vtop \else \if#1b\vbox \else \vcenter \fi\fi
409 \bgroup \let\par\relax
410 \let\@sharp#\let\protect\relax \lineskip\z@\baselineskip\z@
411 \if@tab@rl\hbox\bgroup\beginR\vbox\bgroup\fi
412 \@preamble}
413 }

```

## 5.9 Equation labels in RL documents

Adapted from FarsiT<sub>E</sub>X.

```

414 \def\@open@paren{\if@rlmain}\else(\fi}
415 \def\@close@paren{\if@rlmain(\else)\fi}
416 \def\tagform#1{\maketag@@@{\@open@paren\ignorespaces#1\unskip\@italiccorr\@close@paren}}

```

## 5.10 TODO

### 5.10.1 Unicode CTRL Chars

```

417 %% Make these Unicode chars active and define them so that they work
418 %% in conformance to the Unicode bidi algorithm!
419 %% \catcode"200E=\active
420 %% \catcode"200F=\active
421 %% \def{\setLR}
422 %% \def{\setRL}
423 %% 202A -> {\@ensure@LR\bgroup} ? or \LR
424 %% 202B -> {\@ensure@RL\bgroup} ? or \RL
425 %% 202C -> {\egroup} ? or \if@rl\@rlfalse\else\@rltrue\fi ?
426 %% 202D -> {?} %LR0
427 %% 202E -> {?} %RL0

```