

The letltxmacro package

Heiko Oberdiek*

2016/05/16 v1.5

Abstract

T_EX's `\let` assignment does not work for L^AT_EX macros with optional arguments or for macros that are defined as robust macros by `\DeclareRobustCommand`. This package defines `\LetLtxMacro` that also takes care of the involved internal macros.

Contents

1	Documentation	2
1.1	Supported macro definition commands	2
2	Implementation	2
2.1	Show cases	2
2.1.1	letltxmacro-showcases.tex	2
2.1.2	Result	4
2.2	Package	4
2.2.1	Catcodes and identification	5
2.2.2	Main macros	5
3	Test	8
3.1	Catcode checks for loading	8
3.2	Package tests	9
4	Installation	13
4.1	Download	13
4.2	Bundle installation	13
4.3	Package installation	13
4.4	Refresh file name databases	14
4.5	Some details for the interested	14
5	History	14
	[2008/06/09 v1.0]	14
	[2008/06/12 v1.1]	14
	[2008/06/13 v1.2]	15
	[2008/06/24 v1.3]	15
	[2010/09/02 v1.4]	15
	[2016/05/16 v1.5]	15

*Please report any issues at <https://github.com/ho-tex/oberdiek/issues>

1 Documentation

If someone wants to redefine a macro with using the old meaning, then one method is T_EX's command `\let`:

```
\newcommand{\Macro}{\typeout{Test Macro}}
\let\SavedMacro=\Macro
\renewcommand{\Macro}{%
  \typeout{Begin}%
  \SavedMacro
  \typeout{End}%
}
```

However, this method fails, if `\Macro` is defined by `\DeclareRobustCommand` and/or has an optional argument. In both cases L^AT_EX defines an additional internal macro that is forgotten in the simple `\let` assignment of the example above.

`\LetLtxMacro {<new macro>} {<old macro>}`

Macro `\LetLtxMacro` behaves similar to T_EX's `\let` assignment, but it takes care of macros that are defined by `\DeclareRobustCommand` and/or have optional arguments. Example:

```
\DeclareRobustCommand{\Macro}[1][default]{...}
\LetLtxMacro{\SavedMacro}{\Macro}
```

Then macro `\SavedMacro` only uses internal macro names that are derived from `\SavedMacro`'s macro name. Macro `\Macro` can now be redefined without affecting `\SavedMacro`.

`\GlobalLetLtxMacro {<new macro>} {<old macro>}`

Like `\LetLtxMacro`, but the `<new macro>` is defined globally. Since version 2016/05/16 v1.4.

1.1 Supported macro definition commands

<code>\newcommand</code> , <code>\renewcommand</code>	latex/base
<code>\newenvironment</code> , <code>\renewenvironment</code>	latex/base
<code>\DeclareRobustCommand</code>	latex/base
<code>\newrobustcmd</code> , <code>\renewrobustcmd</code>	etoolbox
<code>\robustify</code>	etoolbox 2008/06/22 v1.6

2 Implementation

2.1 Show cases

2.1.1 letltxmacro-showcases.tex

```
1 <*showcases>
2 \NeedsTeXFormat{LaTeX2e}
3 \makeatletter
```

\Line The result is displayed by macro `\Line`. The percent symbol at line start allows easy grepping and inserting into the DTX file.

```

4 \newcommand*{\Line}[1]{%
5   \typeout{\@percentchar#1}%
6 }

7 \newcommand*{\ShowCmdName}[1]{%
8   \@ifundefined{#1}{}{%
9     \Line{%
10      \space\space(\expandafter\string\csname#1\endcsname) = %
11      (\expandafter\meaning\csname#1\endcsname)%
12    }%
13  }%
14 }
15 \newcommand*{\ShowCmds}[1]{%
16   \ShowCmdName{#1}%
17   \ShowCmdName{#1 }%
18   \ShowCmdName{\#1}%
19   \ShowCmdName{\#1 }%
20 }
21 \let\\ \@backslashchar

\ShowDef

22 \newcommand*{\ShowDef}[2]{%
23   \begingroup
24     \Line{}%
25     \newcommand*{\DefString}{#2}%
26     \@onelevel@sanitize\DefString
27     \Line{\DefString}%
28     #2%
29     \ShowCmds{#1}%
30   \endgroup
31 }

32 \typeout{}
33 \Line{* LaTeX definitions:}
34 \ShowDef{cmd}{%
35   \newcommand{\cmd}[2][default]{}%
36 }
37 \ShowDef{cmd}{%
38   \DeclareRobustCommand{\cmd}{}%
39 }
40 \ShowDef{cmd}{%
41   \DeclareRobustCommand{\cmd}[2][default]{}%
42 }
43 \typeout{}

The minimal version of package etoolbox is 2008/06/12 v1.6a because it fixes
\robustify.

44 \RequirePackage{etoolbox}[2008/06/12]%
45 \Line{}
46 \Line{* etoolbox's robust definitions:}
47 \ShowDef{cmd}{%
48   \newrobustcmd{\cmd}{}%
49 }
50 \ShowDef{cmd}{%
51   \newrobustcmd{\cmd}[2][default]{}%
52 }
53 \Line{}
54 \Line{* etoolbox's \string\robustify:}
55 \ShowDef{cmd}{%

```

```

56 \newcommand{\cmd}[2][default]{ } %
57 \robustify{\cmd}%
58 }
59 \ShowDef{cmd}{%
60 \DeclareRobustCommand{\cmd}{ } %
61 \robustify{\cmd}%
62 }
63 \ShowDef{cmd}{%
64 \DeclareRobustCommand{\cmd}[2][default]{ } %
65 \robustify{\cmd}%
66 }
67 \typeout{ }
68 \@@end
69 </showcases>

```

2.1.2 Result

* LaTeX definitions:

```

\newcommand {\cmd }[2][default]{ }
  (\cmd) = (macro:->\@protected@testopt \cmd \cmd {default})
  (\cmd) = (\long macro:[#1]#2->)

\DeclareRobustCommand {\cmd }{ }
  (\cmd) = (macro:->\protect \cmd_ )
  (\cmd_ ) = (\long macro:->)

\DeclareRobustCommand {\cmd }[2][default]{ }
  (\cmd) = (macro:->\protect \cmd_ )
  (\cmd_ ) = (macro:->\@protected@testopt \cmd_ \cmd_ {default})
  (\cmd_ ) = (\long macro:[#1]#2->)

```

* etoolbox's robust definitions:

```

\newrobustcmd {\cmd }{ }
  (\cmd) = (\protected\long macro:->)

\newrobustcmd {\cmd }[2][default]{ }
  (\cmd) = (\protected macro:->\@testopt \cmd {default})
  (\cmd) = (\long macro:[#1]#2->)

```

* etoolbox's \robustify:

```

\newcommand {\cmd }[2][default]{ } \robustify {\cmd }
  (\cmd) = (\protected macro:->\@protected@testopt \cmd \cmd {default})
  (\cmd) = (\long macro:[#1]#2->)

\DeclareRobustCommand {\cmd }{ } \robustify {\cmd }
  (\cmd) = (\protected macro:->)

\DeclareRobustCommand {\cmd }[2][default]{ } \robustify {\cmd }
  (\cmd) = (\protected macro:->\@protected@testopt \cmd_ \cmd_ {default})
  (\cmd_ ) = (macro:->\@protected@testopt \cmd_ \cmd_ {default})
  (\cmd_ ) = (\long macro:[#1]#2->)

```

2.2 Package

```

70 < *package>

```

2.2.1 Catcodes and identification

```

71 \begingroup\catcode61\catcode48\catcode32=10\relax%
72 \catcode13=5 % ^^M
73 \endlinechar=13 %
74 \catcode123=1 % {
75 \catcode125=2 % }
76 \catcode64=11 % @
77 \def\x{\endgroup
78   \expandafter\edef\csname llm@AtEnd\endcsname{%
79     \endlinechar=\the\endlinechar\relax
80     \catcode13=\the\catcode13\relax
81     \catcode32=\the\catcode32\relax
82     \catcode35=\the\catcode35\relax
83     \catcode61=\the\catcode61\relax
84     \catcode64=\the\catcode64\relax
85     \catcode123=\the\catcode123\relax
86     \catcode125=\the\catcode125\relax
87   }%
88 }%
89 \x\catcode61\catcode48\catcode32=10\relax%
90 \catcode13=5 % ^^M
91 \endlinechar=13 %
92 \catcode35=6 % #
93 \catcode64=11 % @
94 \catcode123=1 % {
95 \catcode125=2 % }
96 \def\TMP@EnsureCode#1#2{%
97   \edef\llm@AtEnd{%
98     \llm@AtEnd
99     \catcode#1=\the\catcode#1\relax
100   }%
101   \catcode#1=#2\relax
102 }
103 \TMP@EnsureCode{40}{12}% (
104 \TMP@EnsureCode{41}{12}% )
105 \TMP@EnsureCode{42}{12}% *
106 \TMP@EnsureCode{45}{12}% -
107 \TMP@EnsureCode{46}{12}% .
108 \TMP@EnsureCode{47}{12}% /
109 \TMP@EnsureCode{58}{12}% :
110 \TMP@EnsureCode{62}{12}% >
111 \TMP@EnsureCode{91}{12}% [
112 \TMP@EnsureCode{93}{12}% ]
113 \edef\llm@AtEnd{%
114   \llm@AtEnd
115   \escapechar\the\escapechar\relax
116   \noexpand\endinput
117 }
118 \escapechar=92 % ``\

Package identification.
119 \NeedsTeXFormat{LaTeX2e}
120 \ProvidesPackage{letltxmacro}%
121 [2016/05/16 v1.5 Let assignment for LaTeX macros (HO)]

```

2.2.2 Main macros

\LetLtxMacro

```

122 \newcommand*{\LetLtxMacro}{%

```

```

123 \llm@ModeLetLtxMacro{%
124 }

\GlobalLetLtxMacro

125 \newcommand*{\GlobalLetLtxMacro}{%
126 \llm@ModeLetLtxMacro\global
127 }

\llm@ModeLetLtxMacro

128 \newcommand*{\llm@ModeLetLtxMacro}[3]{%
129 \edef\llm@escapechar{\the\escapechar}%
130 \escapechar=-1 %
131 \edef\reserved@a{%
132 \noexpand\protect
133 \expandafter\noexpand
134 \csname\string#3 \endcsname
135 }%
136 \ifx\reserved@a#3\relax
137 #1\edef#2{%
138 \noexpand\protect
139 \expandafter\noexpand
140 \csname\string#2 \endcsname
141 }%
142 #1\expandafter\let
143 \csname\string#2 \expandafter\endcsname
144 \csname\string#3 \endcsname
145 \expandafter\llm@LetLtxMacro
146 \csname\string#2 \expandafter\endcsname
147 \csname\string#3 \endcsname{#1}%
148 \else
149 \llm@LetLtxMacro{#2}{#3}{#1}%
150 \fi
151 \escapechar=\llm@escapechar\relax
152 }

\llm@LetLtxMacro

153 \def\llm@LetLtxMacro#1#2#3{%
154 \escapechar=92 %
155 \expandafter\llm@CheckParams\meaning#2:->\@nil{%
156 \begingroup
157 \def\@protected@testopt{%
158 \expandafter\@testopt\@gobble
159 }%
160 \def\@testopt##1##2{%
161 \toks@={##2}%
162 }%
163 \let\llm@testopt\@empty
164 \edef\x{%
165 \noexpand\@protected@testopt
166 \noexpand#2%
167 \expandafter\noexpand\csname\string#2\endcsname
168 }%
169 \expandafter\expandafter\expandafter\def
170 \expandafter\expandafter\expandafter\y
171 \expandafter\expandafter\expandafter{%
172 \expandafter\llm@CarThree#2{}{}\llm@nil
173 }%
174 \ifx\x\y

```

```

175      #2%
176      \def\llm@testopt{%
177        \noexpand\@protected@testopt
178        \noexpand#1%
179      }%
180    \else
181      \edef\x{%
182        \noexpand\@testopt
183        \expandafter\noexpand
184        \csname\string#2\endcsname
185      }%
186      \expandafter\expandafter\expandafter\def
187      \expandafter\expandafter\expandafter\y
188      \expandafter\expandafter\expandafter{%
189        \expandafter\llm@CarTwo#2{}}\llm@nil
190      }%
191      \ifx\x\y
192        #2%
193        \def\llm@testopt{%
194          \noexpand\@testopt
195        }%
196      \fi
197    \fi
198    \ifx\llm@testopt\@empty
199    \else
200      \llm@protected\xdef\llm@GlobalTemp{%
201        \llm@testopt
202        \expandafter\noexpand
203        \csname\string#1\endcsname
204        {\the\toks@}%
205      }%
206    \fi
207    \expandafter\endgroup\ifx\llm@testopt\@empty
208      #3\let#1=#2\relax
209    \else
210      #3\let#1=\llm@GlobalTemp
211      #3\expandafter\let
212        \csname\string#1\expandafter\endcsname
213        \csname\string#2\endcsname
214    \fi
215  }{%
216    #3\let#1=#2\relax
217  }%
218 }

\llm@CheckParams
219 \def\llm@CheckParams#1:->#2\@nil{%
220   \begingroup
221     \def\x{#1}%
222     \ifx\x\llm@macro
223       \endgroup
224     \def\llm@protected{}%
225     \expandafter\@firstoftwo
226   \else
227     \ifx\x\llm@protectedmacro
228       \endgroup
229     \def\llm@protected{\protected}%
230     \expandafter\expandafter\expandafter\@firstoftwo

```

```

231     \else
232     \endgroup
233     \expandafter\expandafter\expandafter\@secondoftwo
234     \fi
235     \fi
236 }

\llm@macro
237 \def\llm@macro{macro}
238 \@onelevel@sanitize\llm@macro

\llm@protectedmacro
239 \def\llm@protectedmacro{\protected macro}
240 \@onelevel@sanitize\llm@protectedmacro

\llm@CarThree
241 \def\llm@CarThree#1#2#3#4\llm@nil{#1#2#3}%

\llm@CarTwo
242 \def\llm@CarTwo#1#2#3\llm@nil{#1#2}%

243 \llm@AtEnd%
244 </package>

```

3 Test

3.1 Catcode checks for loading

```

245 (*test1)

246 \NeedsTeXFormat{LaTeX2e}
247 \documentclass{minimal}
248 \makeatletter
249 \def\RestoreCatcodes{}
250 \count@=0 %
251 \loop
252   \edef\RestoreCatcodes{%
253     \RestoreCatcodes
254     \catcode\the\count@=\the\catcode\count@\relax
255   }%
256 \ifnum\count@<255 %
257   \advance\count@\@ne
258 \repeat
259
260 \def\RangeCatcodeInvalid#1#2{%
261   \count@=#1\relax
262   \loop
263     \catcode\count@=15 %
264     \ifnum\count@<#2\relax
265       \advance\count@\@ne
266     \repeat
267 }
268 \def\Test{%
269   \RangeCatcodeInvalid{0}{47}%
270   \RangeCatcodeInvalid{58}{64}%
271   \RangeCatcodeInvalid{91}{96}%
272   \RangeCatcodeInvalid{123}{127}%
273   \catcode`\@=12 %

```

```

274 \catcode`\=0 %
275 \catcode`\{=1 %
276 \catcode`\}=2 %
277 \catcode`\#=6 %
278 \catcode`\[=12 %
279 \catcode`\]=12 %
280 \catcode`\%=14 %
281 \catcode`\_ =10 %
282 \catcode13=5 %
283 \RequirePackage{letltxmacro}[2016/05/16]\relax
284 \RestoreCatcodes
285 }
286 \Test
287 \csname @@end\endcsname
288 \end
289 </test1>

```

3.2 Package tests

```

290 (*test2)

291 \NeedsTeXFormat{LaTeX2e}
292 \nofiles
293 \documentclass{minimal}
294
295 \usepackage{letltxmacro}[2016/05/16]
296
297 \usepackage{qstest}
298 \IncludeTests{*}
299 \LogTests{log}{*}{*}
300
301 \makeatletter
302
303 \def\TestDef#1{%
304   \begingroup
305     \@makeother\%
306     \@makeother\_ %
307     \expandafter\@TestDef\csname #1\endcsname
308 }
309 \def\@TestDef#1#2{%
310   \edef\a{\expandafter\strip@prefix\meaning#1}%
311   \edef\b{\detokenize{#2}}%
312   \ifx\a\b
313     \else
314       \typeout{1 [\a]}%
315       \typeout{2 [\b]}%
316     \fi
317   \endgroup
318   \Expect*\expandafter\strip@prefix\meaning#1*{\detokenize{#2}}%
319 }
320 \def\TestEquals#1#2{%
321   \Expect*{%
322     \expandafter\ifx\csname#1\expandafter\endcsname
323       \csname#2\endcsname
324     equals%
325     \else
326       wrong%
327     \fi
328   }{equals}%

```

```

329 }
330 \def\SaveA{%
331   \let\SavedA\TestA
332   \expandafter\let\csname\string\SavedA\expandafter\endcsname
333     \csname\string\TestA\endcsname
334   \expandafter\let\csname SavedA \expandafter\endcsname
335     \csname TestA \endcsname
336 }
337 \def\CheckA{%
338   \TestEquals{SavedA}{TestA}%
339   \TestEquals{\string\SavedA}{\string\TestA}%
340   \TestEquals{SavedA }{TestA }%
341 }
342
343 \begin{qstest}{robust}{robust}
344   \begingroup
345     \DeclareRobustCommand*\TestA{Test}%
346     \SaveA
347     \TestDef{TestA}{\protect \TestA }%
348     \CheckA
349     \DeclareRobustCommand{\TestA}{Test}%
350     \SaveA
351     \TestDef{TestA}{\protect \TestA }%
352     \LetLtxMacro\TestB\TestA
353     \TestDef{TestB}{\protect \TestB }%
354     \TestEquals{TestB }{TestA }%
355     \CheckA
356     \GlobalLetLtxMacro\TestC\TestA
357     \CheckA
358   \endgroup
359   \TestDef{TestC}{\protect \TestC }%
360   \DeclareRobustCommand{\TestA}{Test}%
361   \TestEquals{TestC }{TestA }%
362 \end{qstest}
363
364 \begin{qstest}{default}{default}
365   \begingroup
366     \newcommand{\TestA}[1][\relax default \empty]{TestA #1}%
367     \SaveA
368     \TestDef
369     {TestA}{\@protected@testopt \TestA \TestA {\relax default \empty }}%
370     \LetLtxMacro\TestB\TestA
371     \TestDef
372     {TestB}{\@protected@testopt \TestB \TestB {\relax default \empty }}%
373     \CheckA
374     \TestEquals{\string\TestB}{\string\TestA}%
375     \GlobalLetLtxMacro\TestC\TestA
376     \CheckA
377   \endgroup
378   \TestDef
379   {TestC}{\@protected@testopt \TestC \TestC {\relax default \empty }}%
380   \newcommand{\TestA}[1][\relax default \empty]{TestA #1}%
381   \TestEquals{\string\TestC}{\string\TestA}%
382 \end{qstest}
383
384 \begin{qstest}{robustdefault}{robustdefault}
385   \begingroup
386     \DeclareRobustCommand{\TestA}[1][\default]{TestA}%

```

```

387 \SaveA
388 \LetLtxMacro\TestB\TestA
389 \TestDef{TestB}{\protect \TestB }%
390 \TestDef{TestB }{\@protected@testopt \TestB \TestB {\default }}%
391 \CheckA
392 \GlobalLetLtxMacro\TestC\TestA
393 \CheckA
394 \endgroup
395 \TestDef{TestC}{\protect \TestC }%
396 \TestDef{TestC }{\@protected@testopt \TestC \TestC {\default }}%
397 \end{qstest}
398
399 \begin{qstest}{plain}{plain}
400 \LetLtxMacro\NewRelax\relax
401 \Expect*{\meaning\relax}*{\string\relax}%
402 \Expect*{\meaning\NewRelax}*{\string\relax}%
403 \LetLtxMacro\NewHbox\hbox
404 \Expect*{\meaning\hbox}*{\string\hbox}%
405 \Expect*{\meaning\NewHbox}*{\string\hbox}%
406 \LetLtxMacro\NewEmpty\empty
407 \Expect*{\meaning\empty}{macro:->}%
408 \Expect*{\meaning\NewEmpty}{macro:->}%
409 \def\TestA{\iffalse}%
410 \LetLtxMacro\TestB\TestA
411 \TestDef{TestB}{\iffalse }%
412 \TestEquals{TestA}{TestB}%
413 \end{qstest}

414 \usepackage{etoolbox}[2008/06/22]
415
416 \begin{qstest}{oldnewrobustcmd}{oldnewrobustcmd}
417 \begingroup
418 \protected\edef\TestA{%
419 \noexpand\@protected@testopt
420 \noexpand\TestA
421 \expandafter\noexpand\csname\string\TestA\endcsname
422 {\noexpand\default}}%
423 }%
424 \expandafter\protected\expandafter
425 \def\csname\string\TestA\endcsname{TestA}%
426 \SaveA
427 \LetLtxMacro\TestB\TestA
428 \TestDef{TestB}{\@protected@testopt \TestB \TestB {\default }}%
429 \protected\edef\TestC{%
430 \noexpand\@protected@testopt
431 \noexpand\TestB
432 \expandafter\noexpand\csname\string\TestB\endcsname
433 {\noexpand\default}}%
434 }%
435 \TestEquals{TestB}{TestC}%
436 \CheckA
437 \GlobalLetLtxMacro\TestD\TestA
438 \CheckA
439 \endgroup
440 \TestDef{TestD}{\@protected@testopt \TestD \TestD {\default }}%
441 \protected\edef\TestC{%
442 \noexpand\@protected@testopt
443 \noexpand\TestD
444 \expandafter\noexpand\csname\string\TestD\endcsname

```

```

445     {\noexpand\default}%
446   }%
447   \TestEquals{TestD}{TestC}%
448 \end{qstest}
449
450 \begin{qstest}{newrobustcmd}{newrobustcmd}
451   \begingroup
452     \newrobustcmd{\TestA}[1][\default]{TestA}%
453     \SaveA
454     \LetLtxMacro\TestB\TestA
455     \TestDef{TestB}{\@testopt \TestB {\default }}%
456     \protected\edef\TestC{%
457       \noexpand\@testopt
458       \expandafter\noexpand\csname\string\TestB\endcsname
459       {\noexpand\default}%
460     }%
461     \TestEquals{TestB}{TestC}%
462     \CheckA
463     \GlobalLetLtxMacro\TestD\TestA
464     \CheckA
465   \endgroup
466   \TestDef{TestD}{\@testopt \TestD {\default }}%
467   \protected\edef\TestC{%
468     \noexpand\@testopt
469     \expandafter\noexpand\csname\string\TestD\endcsname
470     {\noexpand\default}%
471   }%
472   \TestEquals{TestD}{TestC}%
473 \end{qstest}
474
475 \begin{qstest}{robustifyopt}{robustifyopt}
476   \newcommand{\TestA}[2][\default]{}%
477   \TestDef{TestA}{\@protected@testopt \TestA \TestA {\default }}%
478   \robustify\TestA
479   \TestDef{TestA}{\@protected@testopt \TestA \TestA {\default }}%
480   \protected\edef\TestC{%
481     \noexpand\@protected@testopt
482     \noexpand\TestA
483     \expandafter\noexpand\csname\string\TestA\endcsname
484     {\noexpand\default}%
485   }%
486   \TestEquals{TestA}{TestC}%
487   \SaveA
488   \LetLtxMacro\TestB\TestA
489   \TestDef{TestB}{\@protected@testopt \TestB \TestB {\default }}%
490   \protected\edef\TestC{%
491     \noexpand\@protected@testopt
492     \noexpand\TestB
493     \expandafter\noexpand\csname\string\TestB\endcsname
494     {\noexpand\default}%
495   }%
496   \TestEquals{TestB}{TestC}%
497   \CheckA
498 \end{qstest}
499
500 \begin{qstest}{robustifydeclare}{robustifydeclare}
501   \DeclareRobustCommand\TestA{\iffalse}%
502   \robustify\TestA

```

```

503 \SaveA
504 \LetLtxMacro\TestB\TestA
505 \TestDef{TestB}{\iffalse }%
506 % before etoolbox 2008/06/22: \protected\long\def
507 % since etoolbox 2008/06/22: \protected\def
508 \protected\def\TestC{\iffalse}%
509 \TestEquals{TestB}{TestC}%
510 \CheckA
511 \end{qstest}
512
513 \begin{qstest}{\robustifydeclarelong}{\robustifydeclarelong}
514 \DeclareRobustCommand\TestA[1]{\iffalse}%
515 \robustify\TestA
516 \SaveA
517 \LetLtxMacro\TestB\TestA
518 \TestDef{TestB}{\iffalse }%
519 % before etoolbox 2008/06/22: \protected\long\def
520 % since etoolbox 2008/06/22: \protected\def
521 \protected\long\def\TestC#1{\iffalse}%
522 \TestEquals{TestB}{TestC}%
523 \CheckA
524 \end{qstest}
525 \csname @@end\endcsname
526 \</test2>

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

[CTAN:macros/latex/contrib/oberdiek/letltxmacro.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/letltxmacro.pdf](#) Documentation.

Bundle. All the packages of the bundle ‘oberdiek’ are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

[CTAN:install/macros/latex/contrib/oberdiek.tds.zip](#)

TDS refers to the standard “A Directory Structure for T_EX Files” ([CTAN:pkg/tds](#)). Directories with `texmf` in their name are usually organized this way.

4.2 Bundle installation

Unpacking. Unpack the `oberdiek.tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek.tds.zip -d ~/texmf
```

4.3 Package installation

Unpacking. The `.dtx` file is a self-extracting docstrip archive. The files are extracted by running the `.dtx` through plain T_EX:

```
tex letltxmacro.dtx
```

¹[CTAN:pkg/letltxmacro](#)

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

```
letltxmacro.sty      → tex/latex/oberdiek/letltxmacro.sty
letltxmacro.pdf      → doc/latex/oberdiek/letltxmacro.pdf
letltxmacro-showcases.tex → doc/latex/oberdiek/letltxmacro-showcases.tex
test/letltxmacro-test1.tex → doc/latex/oberdiek/test/letltxmacro-test1.tex
test/letltxmacro-test2.tex → doc/latex/oberdiek/test/letltxmacro-test2.tex
letltxmacro.dtx      → source/latex/oberdiek/letltxmacro.dtx
```

If you have a `docstrip.cfg` that configures and enables `docstrip`'s TDS installing feature, then some files can already be in the right place, see the documentation of `docstrip`.

4.4 Refresh file name databases

If your $\text{T}_{\text{E}}\text{X}$ distribution ($\text{T}_{\text{E}}\text{X}$ Live, `mik $\text{T}_{\text{E}}\text{X}$` , ...) relies on file name databases, you must refresh these. For example, $\text{T}_{\text{E}}\text{X}$ Live users run `texhash` or `mktexlsr`.

4.5 Some details for the interested

Unpacking with $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$. The `.dtx` chooses its action depending on the format:

plain $\text{T}_{\text{E}}\text{X}$: Run `docstrip` and extract the files.

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$: Generate the documentation.

If you insist on using $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ for `docstrip` (really, `docstrip` does not need $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$), then inform the `autodetect` routine about your intention:

```
latex \let\install=y\input{letltxmacro.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with `pdf $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$` :

```
pdflatex letltxmacro.dtx
makeindex -s gind.ist letltxmacro.idx
pdflatex letltxmacro.dtx
makeindex -s gind.ist letltxmacro.idx
pdflatex letltxmacro.dtx
```

5 History

[2008/06/09 v1.0]

- First version.

[2008/06/12 v1.1]

- Support for `etoolbox`'s `\newrobustcmd` added.

[2008/06/13 v1.2]

- Support for etoolbox's `\robustify` added.

[2008/06/24 v1.3]

- Test file adapted for etoolbox 2008/06/22 v1.6.

[2010/09/02 v1.4]

- `\GlobalLetLtxMacro` added.

[2016/05/16 v1.5]

- Documentation updates.

6 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; plain numbers refer to the code lines where the entry is used.

Symbols		A	
<code>\#</code>	277	<code>\a</code>	310, 312, 314
<code>\%</code>	280	<code>\advance</code>	257, 265
<code>\@</code>	273	B	
<code>\@@end</code>	68	<code>\b</code>	311, 312, 315
<code>\@TestDef</code>	307, 309	<code>\begin</code>	343, 364, 384, 399, 416, 450, 475, 500, 513
<code>\@backslashchar</code>	21	C	
<code>\@empty</code>	163, 198, 207	<code>\catcode</code>	71, 72, 74, 75, 76, 80, 81, 82, 83, 84, 85, 86, 89, 90, 92, 93, 94, 95, 99, 101, 254, 263, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282
<code>\@firstoftwo</code>	225, 230	<code>\CheckA</code>	337, 348, 355, 357, 373, 376, 391, 393, 436, 438, 462, 464, 497, 510, 523
<code>\@gobble</code>	158	<code>\cmd</code>	35, 38, 41, 48, 51, 56, 57, 60, 61, 64, 65
<code>\@ifundefined</code>	8	<code>\count@</code>	250, 254, 256, 257, 261, 263, 264, 265
<code>\@makeother</code>	305, 306	<code>\csname</code>	10, 11, 78, 134, 140, 143, 144, 146, 147, 167, 184, 203, 212, 213, 287, 307, 322, 323, 332, 333, 334, 335, 421, 425, 432, 444, 458, 469, 483, 493, 525
<code>\@ne</code>	257, 265	D	
<code>\@nil</code>	155, 219	<code>\DeclareRobustCommand</code>	38, 41, 60, 64, 345, 349, 360, 386, 501, 514
<code>\@onelevel@sanitize</code>	26, 238, 240	<code>\default</code>	386, 390, 396, 422, 428, 433, 440, 445, 452, 455, 459, 466, 470, 476, 477, 479, 484, 489, 494
<code>\@percentchar</code>	5	<code>\DefString</code>	25, 26, 27
<code>\@protected@testopt</code>	157, 165, 177, 369, 372, 379, 390, 396, 419, 428, 430, 440, 442, 477, 479, 481, 489, 491		
<code>\@secondoftwo</code>	233		
<code>\@testopt</code>	158, 160, 182, 194, 455, 457, 466, 468		
<code>\[</code>	278		
<code>\</code>	18, 19, 21, 118, 274, 305, 369, 372, 379, 390, 396, 428, 440, 455, 466, 477, 479, 489		
<code>\{</code>	275		
<code>\}</code>	276		
<code>\]</code>	279		
<code>_</code>	281, 306		

<code>\detokenize</code>	311, 318		
<code>\documentclass</code>	247, 293		
E			
<code>\empty</code>	366, 369, 372, 379, 380, 406, 407		
<code>\end</code>	288, 362, 382, 397, 413, 448, 473, 498, 511, 524		
<code>\endcsname</code>	10, 11, 78, 134, 140, 143, 144, 146, 147, 167, 184, 203, 212, 213, 287, 307, 322, 323, 332, 333, 334, 335, 421, 425, 432, 444, 458, 469, 483, 493, 525		
<code>\endinginput</code>	116		
<code>\endlinechar</code>	73, 79, 91		
<code>\escapechar</code>	115, 118, 129, 130, 151, 154		
<code>\Expect</code>	318, 321, 401, 402, 404, 405, 407, 408		
G			
<code>\GlobalLetLtxMacro</code>	2, 125, 356, 375, 392, 437, 463		
H			
<code>\hbox</code>	403, 404, 405		
I			
<code>\iffalse</code>	409, 411, 501, 505, 508, 514, 518, 521		
<code>\ifnum</code>	256, 264		
<code>\ifx</code>	136, 174, 191, 198, 207, 222, 227, 312, 322		
<code>\IncludeTests</code>	298		
L			
<code>\LetLtxMacro</code>	2, 122, 352, 370, 388, 400, 403, 406, 410, 427, 454, 488, 504, 517		
<code>\Line</code>	4, 9, 24, 27, 33, 45, 46, 53, 54		
<code>\llm@AtEnd</code>	97, 98, 113, 114, 243		
<code>\llm@CarThree</code>	172, 241		
<code>\llm@CarTwo</code>	189, 242		
<code>\llm@CheckParams</code>	155, 219		
<code>\llm@escapechar</code>	129, 151		
<code>\llm@GlobalTemp</code>	200, 210		
<code>\llm@LetLtxMacro</code>	145, 149, 153		
<code>\llm@macro</code>	222, 237		
<code>\llm@ModeLetLtxMacro</code> ..	123, 126, 128		
<code>\llm@nil</code>	172, 189, 241, 242		
<code>\llm@protected</code>	200, 224, 229		
<code>\llm@protectedmacro</code>	227, 239		
<code>\llm@testopt</code>	163, 176, 193, 198, 201, 207		
<code>\LogTests</code>	299		
<code>\loop</code>	251, 262		
M			
<code>\makeatletter</code>	3, 248, 301		
<code>\meaning</code>	11, 155, 310, 318, 401, 402, 404, 405, 407, 408		
N			
<code>\NeedsTeXFormat</code>	2, 119, 246, 291		
<code>\newcommand</code>	4, 7, 15, 22, 25, 35, 56, 122, 125, 128, 366, 380, 476		
<code>\NewEmpty</code>	406, 408		
<code>\NewHbox</code>	403, 405		
<code>\NewRelax</code>	400, 402		
<code>\newrobustcmd</code>	48, 51, 452		
<code>\nofiles</code>	292		
P			
<code>\protect</code>	132, 138, 347, 351, 353, 359, 389, 395		
<code>\protected</code>	229, 239, 418, 424, 429, 441, 456, 467, 480, 490, 506, 507, 508, 519, 520, 521		
<code>\ProvidesPackage</code>	120		
R			
<code>\RangeCatcodeInvalid</code>	260, 269, 270, 271, 272		
<code>\repeat</code>	258, 266		
<code>\RequirePackage</code>	44, 283		
<code>\reserved@a</code>	131, 136		
<code>\RestoreCatcodes</code> ..	249, 252, 253, 284		
<code>\robustify</code>	54, 57, 61, 65, 478, 502, 515		
S			
<code>\SaveA</code>	330, 346, 350, 367, 387, 426, 453, 487, 503, 516		
<code>\SavedA</code>	331, 332, 339		
<code>\ShowCmdName</code>	7, 16, 17, 18, 19		
<code>\ShowCmds</code>	15, 29		
<code>\ShowDef</code>	22, 34, 37, 40, 47, 50, 55, 59, 63		
<code>\space</code>	10		
<code>\strip@prefix</code>	310, 318		
T			
<code>\Test</code>	268, 286		
<code>\TestA</code>	331, 333, 339, 345, 347, 349, 351, 352, 356, 360, 366, 369, 370, 374, 375, 380, 381, 386, 388, 392, 409, 410, 418, 420, 421, 425, 427, 437, 452, 454, 463, 476, 477, 478, 479, 482, 483, 488, 501, 502, 504, 514, 515, 517		
<code>\TestB</code>	352, 353, 370, 372, 374, 388, 389, 390, 410, 427, 428, 431, 432, 454, 458, 488, 489, 492, 493, 504, 517		
<code>\TestC</code>	356, 359, 375, 379, 381, 392, 395, 396, 429, 441, 456, 467, 480, 490, 508, 521		
<code>\TestD</code>	437, 440, 443, 444, 463, 469		
<code>\TestDef</code>	303, 347, 351, 353, 359, 368, 371, 378, 389, 390, 395, 396, 411, 428, 440, 455, 466, 477, 479, 489, 505, 518		

<code>\TestEquals</code>	...	320, 338, 339, 340, 354, 361, 374, 381, 412, 435, 447, 461, 472, 486, 496, 509, 522	
<code>\the</code>	79, 80, 81, 82, 83, 84, 85, 86, 99, 115, 129, 204, 254	
<code>\TMP@EnsureCode</code>	.	96, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112	
<code>\toks@</code>	161, 204	
<code>\typeout</code>	5, 32, 43, 67, 314, 315	
			U
			<code>\usepackage</code> 295, 297, 414
			X
			<code>\x</code> 77, 89, 164, 174, 181, 191, 221, 222, 227
			Y
			<code>\y</code> 170, 174, 187, 191