

数字时钟 `tdclock` 宏包 v2.2

Luis Rández & Juan I. Montijano [著]

黄旭华 [译]

2011 年 4 月 27 日

1 宏包

`tdclock` 宏包是一个 $\text{\LaTeX}2\text{e}$ 宏包，它用于将数字时钟 (ticking digital clock) 插入到由 \LaTeX 创建的 pdf 文档中，该数字时钟可以显示正在读取文档的日期和/或时间，例如：

该宏包由命令 `\usepackage[“options”]{tdclock}` 加载。

选项 (options)：

- `timeinterval= n`

n 可以是任何正整数。每 n 秒时钟会更新其内部状态 (internal status)。因此，如果加载该宏包时带有 `[timeinterval=120]` 选项，时钟每 2 分钟就会更新其显示状态。

默认的内部更新时间为 29 秒。

注意， n 取小值时，在文档打开的期间 Adobe 会占用很大的内存并会增加。不建议将 n 的值设置成小于 10。

- `font=“adobe 字体”`

“adobe 字体”可以选用下列 9 种 adobe 字体之一：例如，`[font=HelvI]`

Helv HelvI HelvBI Times TimesI TimesBI

Cour CourI CourBI

目前，没有其它字体如 $\text{T}_{\text{E}}\text{X}$ 字体可用。这是因为该“动态 (dynamic)”时钟是基于嵌入到 pdf 文档中的 javascript 代码的。

- `timeduration= “time (in minutes)”`

“time” 可以是任何整数 (integer) 或十进制数 (integer)。该选项应用于演示稿 (例如 beamer) 中。它的值就是演示稿持续的时间。该选项和下面的选项联用时很有意义。

- `timewarningfirst= n`

n 的取值是 1 和 100 之间的整数。如果设定了该选项，在 `timeduration` 中指定的 $n\%$ 之后的时间，根据下一个选项，改变 `tdclock` 时间的颜色和背景。这可用于第一次提醒你演示稿还剩多少时间。默认值为 90。

- `timewarningsecond= n`

n 的取值是 1 和 100 之间的整数 (通常大于 `timewarningfirst` 的取值)。如果设定了该选项，在 `timeduration` 中指定的 $n\%$ 之后的时间，根据下一个选项，改变 `tdclock` 时间的颜色和背景。这可用于第二次提醒你演示稿还剩很少的时间。默认值为 95。

- `colorwarningfirst= “color”`

该选项用于指定由 `timeduration` 和 `timewarningfirst` 选项指定的时间之后的 `tdclock` 时间的颜色。默认值为橘黄色 (orange)。

- `fillcolorwarningfirst= “color”`

该选项用于指定由 `timeduration` 和 `timewarningfirst` 选项指定的时间之后的 `tdclock` 时间的颜色背景 (color background)。默认值为“透明 (transparent)”。该选项只有应用于由 `\crono`、`\tdclock` 和 `\tdtime` 指定的时间序列中的时间才有效。

- `colorwarningsecond= “color”`

该选项用于指定由 `timeduration` 和 `timewarningsecond` 选项指定的时间之后的 `tdclock` 时间的颜色。默认值为红色 (red)。

- `fillcolorwarningsecond= “color”`

该选项用于指定由 `timeduration` 和 `timewarningsecond` 选项指定的时间之后的 `tdclock` 时间的颜色背景。默认值为透明 (`transparent`)。该选项只有应用于由 `\crono`, `\tdclock` 和 `\tdtime` 指定的时间序列中的时间才有效。

- `timedeath= 0` 或 `1`

将该选项设为 `1` 时, 由 `timeduration` 指定的时间的 `110%` 时将关闭 pdf 文档。默认值是 `0` (不会关闭文档)。

下面是使用上述选项的一些例子:

- `\usepackage[font=Times, timeinterval=59]{tdclock}`

指定显示时钟的字体为 `Times`, 每 `59` 秒更新一次时钟。没有警示提示演示稿还剩多少时间。

- `\usepackage[font=Times, timeinterval=30, timeduration=20, timewarningfirst=85, timewarningsecond=90, fillcolorwarningsecond=white!60!yellow]{tdclock}`

指定显示时钟的字体为 `Times`, 在 `20` 分钟的演讲中每 `30` 秒更新一次时钟。时钟字体的颜色在演讲进行到 `17` 分钟 (`20` 分钟的 `85%`) 后改变成红色, 演讲进行到 `18` 分钟 (`20` 分钟的 `90%`) 后时钟的背景改变成 `white!60!yellow`。

- `\usepackage[timeduration=60, timewarningfirst=90, timewarningsecond=95, colorwarningfirst=blue, fillcolorwarningfirst=white!60!yellow, fillcolorwarningsecond=white!10!yellow, timedeath=1]{tdclock}`

指定演示稿将耗时 `60` 分钟。在 `54` 分钟 (`60` 分钟的 `90%`) 后时钟字体的颜色变为蓝色, 时钟的背景变为 `white!60!yellow`。在 `57` 分钟后时钟字体的颜色变为红色, 时钟的背景变为 `white!10!yellow`。在 `66` 分钟后自动关闭文档。

2 限制 – 需求

该宏包依赖于 `hyperref`、`xkeyval` 和 `xcolor` 宏包。它能用 `PDFLATEX` 编译，也可以用 `LATEX`→`DviPs`→`Ps2pdf` 序列编译。在由 `Dvipdfm` 创建的文档中，时钟可能无法正常工作。

因为该时钟使用 `javascript` 代码，并非所有的 pdf 阅读器能解释 `javascript`，只有一部分 pdf 阅读器才能正确显示时钟。经测试，在 windows 和 Linux 系统中，Adobe reader 和 Adobe acrobat 均能正确显示时钟。

3 安装

将 `tdclock.sty` 文件复制到 `LATEX` 能找到的目录中即可。

4 获得该宏包

该宏包的下载地址是：<http://pcmap.unizar.es/numerico/software>

5 宏指令

该数字时钟必须用 `\initclock` 命令初始化，通常将该命令放置于文档（document）的开始处，`\begin{document}` 之后。

下面的宏指令能自动显示当前的日期和/或时间：

命令	行为
<code>\tdclock</code>	显示完整的时钟
<code>\tdtime</code>	显示当前的时间
<code>\tddate</code>	显示当前的日期
<code>\tdday</code>	显示当日
<code>\tdmonth</code>	显示当月
<code>\tdyear</code>	显示当年
<code>\tdhours</code>	显示当前的小时
<code>\tdminutes</code>	显示当前的分钟
<code>\tdseconds</code>	显示当前的秒

该宏包也提供了秒表（计时表、停表，`stopwatch`），这是通过下面的宏命令来实现的：

命令	行为
<code>\crono</code>	显示一个秒表 (stopwatch)
<code>\cronohours</code>	显示一个 crono (只有小时)
<code>\cronominutes</code>	显示一个 crono (只有分钟)
<code>\cronoseconds</code>	显示一个 crono (只有秒钟)

另外, 该宏包包括两个用于显示按钮的宏。按下其中之一按钮可以将秒表 (stopwatch) 的值重置为 0。其命令代码是 `\resetcrono{“按钮文本”}`。例如, `\resetcrono{\fbox{reset}} \crono` 产生的按钮 (按下它看看会发生什么) 是: `reset`。

按下另一个按钮可以将当前时间切换 (toggle) 到秒表 (stopwatch)。例如, `\toggleclock{\fbox{toggleclock}} \tdtime` 产生的按钮 (按下它看看会发生什么) 是: `toggle`。

注意: `\toggleclock` 只有用于 `\tdtime` 命令时才有效, 用于 `\crono` 命令时无效。

5.1 格式化输出

5.1.1 尺寸和颜色

时间 (time) 和日期 (date) 字符的尺寸 (size) 和颜色 (color) 就是 L^AT_EX 当前字体的尺寸和颜色。例如:

`\centerline{\textcolor{blue}{\Huge \tdclock}}` 生成的时钟是:

另外一个宏 `\factorclockfont{“因子 (factor)”}`, 它可以按所需的因子 (factor) 增大时钟的尺寸。要将时钟的尺寸重置为其初始的尺寸, 可以使用下面的命令: `\factorclockfont{1}`

5.1.2 格式化外观

有两个命令用于更改 `\time` 和 `\crono` 的外观:

- `\hhmmss` 重定义上述命令, 它可以显示时、分、秒 (这是默认的),
- `\hhmm` 重定义上述命令, 它只显示时、分。

时、分、秒由在宏 `\timeseparator` 中定义的字符分隔。因此, 通过重定义该宏就可以设定时间分隔 (time separator)。例如,

`\renewcommand{\timeseparator}{;} \tdtime`
生成的时钟为：

； ；

依据 `\tdhours`、`\tdminutes` 和 `\seconds` 等宏可以将时间 (time) 设置成不同的格式。例如：

`\minutes.\seconds` 生成的时钟是： .

依据 `\cronohours`、`\cronominutes` 和 `\cronoseconds` 等宏可以将秒表 (stopwatch) 设置成不同的格式。例如：

`\cronominutes.\cronoseconds` 生成的秒表是： .

实际上，`\tdtime` 和 `\crono` 分别被默认定义成了

`\tdhours\timeseparator\tminutes\timeseparator\tseconds`
和

`\cronohours\timeseparator\cronominutes\timeseparator\cronoseconds`。

关于日期 (date)，有两条命令可以改变 `\tddate` 的显示效果：

- `\ddmmyyy` 重定义上述命令，它可以显示日 (day)、月 (month)、年 (year) (这是默认的)，
- `\mddyyy` 重定义上述命令，它可以显示月 (month)、日 (day)、年 (year)。

日、月、年由在 `\dateseparator` 宏中定义的字符分隔。因此，通过重定义该宏就可以设定日期分隔 (date separator)。例如，

`\renewcommand{\dateseparator}{;} \tddate`
生成的时钟是：

； ；

依据 `\tdday`、`\tdmonth` 和 `\tdyear` 等宏可以将日期 (date) 设置成不同的格式。例如：

`\day-\month` 生成的时钟是： -

默认情况下，`\tddate` 被定义成了

`\tdday\dateseparator\tdmonth\dateseparator\tdyear`。

因为用于动态显示日期和时间的字体是 adobe 字体之一，我们用两条命令 `\pdfslash` 和 `\pdfcolon` 提供字符“斜线 (slash)”和“冒号 (colon)”，这样，定义了 `\renewcommand{\timeseparator}{\pdfcolon}` 后，命令 `\tdtime` 将使所有字符使用相同的字体。同理，如果我们使用了 `\pdfslash` 作为分隔符，命令 `\tddate` 将使所有字符使用相同的字体。

6 宏概要

时间 - 日期 (Time-Date) 宏

宏	结果	行为
<code>\initclock</code>		初始化时钟
<code>\tdclock</code>		显示时钟全部
<code>\tdtime</code>		显示当前时间
<code>\tddate</code>		显示当前日期
<code>\tdday</code>		显示当日
<code>\tdmonth</code>		显示当月
<code>\tdyear</code>		显示当年
<code>\tdhours</code>		显示当前小时
<code>\tdminutes</code>		显示当前分钟
<code>\tdseconds</code>		显示当前秒钟
<code>\crono</code>		显示秒表
<code>\cronohours</code>		crono 时
<code>\cronominutes</code>		crono 分
<code>\tdwarningbox{\cronominutes}</code>		提醒框 crono 分
<code>\cronoseconds</code>		crono 秒
<code>\resetcrono{"button"}</code>	<input type="button" value="reset"/>	重置 crono 时为 0
<code>\toggleclock{"button"}</code>	<input type="button" value="toggle"/>	切换 time-crono

格式化宏

<code>\hhmm\tdtime</code>	
<code>\hhmmss\tdtime</code>	
<code>\mddy\tddate</code>	
<code>\ddmy\tddate</code>	
<code>\renewcommand{\dateseparator}{--}\tddate</code>	--
<code>\renewcommand{\timeseparator}{.}\tdtime</code>	.
<code>\pdfslash</code>	
<code>\pdfcolon</code>	
<code>\factorclockfont{2.0}\tdtime</code>	