The `circledsteps` package: circled numbers, circled steps
and more

Romano Giannetti <romano.giannetti@gmail.com>

January 12, 2020

1 Introduction

This package was born thanks to the discussion on tex.stackexchange.com, and with the help of the several authors that contributed to the answers. The base idea is to have a command that can create arbitrary “circled” text (numbers or letters) and that can work in normal text and in a node in a `tikzpicture` or derivative. That forbid to use TikZ itself for the circles — you can’t safely nest `tikzpicture`s.

This package provides two things: the first one, macros to generate the circled text that use the original `pict2e` method, and then a (simple, to be taken as an example) set of macro to generate sequential circled numbers that can be referenced afterward.

The package loads (if not already loaded) the packages `pict2e`, `pictures`, `xcolor` and `pgfkeys` (the latter one is automatically loaded by `tikz`; you do not need `tikz` to use just `circledsteps`). Moreover, `etoolbox` is loaded to modify `pict2e` to have filled ovals.

The manual is for version v1.3. The main changes between version are listed below.

v1.3 Minor fixes to the manual; added a warning in case the patching of `picture.sty` fails.

v1.2 Added optional argument to the commands and make them sensitive to math mode (thanks to @tallmarmot for the suggestion and example code); fixed several bugs.

v1.1 Default to normal colors; keep the “red/blue” theme as optional. If you use the package option `redblue` or `legacy` you will have the same effect than

\begin{verbatim}
\pgfkeys{/csteps/inner color=red, /csteps/outer color=blue}
\end{verbatim}

v1.0 Initial release.

2 Basic commands

The basic commands are:

\begin{verbatim}
\Circled[]{}        circled text using the package colors, text on the baseline
\CircledTop[]{}    circled text using the package colors, circle on the baseline
\CircledText[]{}   circled text using the current color\footnote{It may fail when used in headers or footers, in the sense it can select the wrong color in case of page breaks.}
\end{verbatim}
The parameters for the output are controlled using \pgfkeys; you can change them with \pgfkeys or \tikzset; you can obviously limit the effect of change using normal \LaTeX scoping rules. Also, you can specify parameters (without the initial /csteps/ as optional arguments to the commands.

The pseudo-color none means that the current color is used for both the text and the circle.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Meaning</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>/csteps/inner ysep</td>
<td>vertical spacing</td>
<td>4pt</td>
</tr>
<tr>
<td>/csteps/inner xsep</td>
<td>horizontal spacing</td>
<td>4pt</td>
</tr>
<tr>
<td>/csteps/inner color</td>
<td>color of the text</td>
<td>none</td>
</tr>
<tr>
<td>/csteps/outer color</td>
<td>color of the circle</td>
<td>none</td>
</tr>
<tr>
<td>/csteps/fill color</td>
<td>background color of the circle</td>
<td>none</td>
</tr>
</tbody>
</table>

The usage of the package is better explained with examples.

This is normal text: ① is on the baseline, ② is on top. This is more evident if you have descendents, like p: ③ and ④. You can set options as optional arguments, like ⑤ or ⑥. Math mode is detected too: ⑦ and ⑧.

For big horizontal things the circle becomes oval: ⑨ or ⑩.

Inside \texttt{tikz} they work ok:

\texttt{\begin{tikzpicture}[scale=0.8, rotate=30, text width=3cm, transform shape]
  \node [draw,color=purple]{text \Circled{1} \Circled[inner color=blue]{2}};
\end{tikzpicture}}

If you like more breathing space: ⑪ or ⑫.

If you want to change colors it’s easy: ⑬ ⑭ ⑮ ⑯.

And you can have filled circles and ovals: ⑰ ⑱.
You can have inline numbers like \( \text{1} \) or exponents\(^2 \) and so on. They follow the current color: \( \text{1} \) and \( \text{2} \) automatically, as you can see: \( \text{1} + \text{1} \approx 3 \). In-text circled numbers look better when a bit smaller, though, as you can see in \( \small{\text{1}} \) for example.

You can have also quite nice math mode effects, thanks to suggestion from @tallmarmot:

\[
\int_{-\text{1}}^{\text{1}} \! \mathrm{d}x \, f_{\text{1}}(x)
\]

\[
\int_{-\text{1}}^{\text{1}} \! \mathrm{d}x \, f_{\text{2}}(x)
\]

\[
\int_{-\text{1}}^{\text{1}} \! \mathrm{d}x \, f_{\text{3}}(x)
\]

3 Automatically generated numbers

The command \cstep will generate a circled number, starting from 1, that can be referenced with the normal \label/\ref mechanism. You can reset the numbering with \startcstep. For example:

\begin{figure}[h]
\centering
\begin{circuitikz}
\draw (0,0) to [V, o, l=0.3cm] (0,-1); \node [red] at (0.2,-0.8) {1};
\draw (0,-1) to [R, l=$R_1$, o, v=2] (0,0);
\node [red] at (0.2,-1.2) {2};
\draw (0,-1) to [R, l=$R_2$, o] (0,-2);
\node [red] at (0.2,-1.8) {3};
\draw (0,-1) to [R, l=$R_3$, o] (0,-2);
\node [red] at (0.2,-2.2) {4};
\draw (0,-1) to [C, o] (0,-2);
\node [red] at (0.2,-2.8) {5};
\draw (0,0) to [I, o, v=-0.7, l=$v_s$] (0,-1);
\node [red] at (0.2,-0.4) {6};
\draw (0,-1) to [I, o, v=0.93, l=$V_{CC}$] (0,0);
\node [red] at (0.2,-0.7) {7};
\draw (0,-1) to [I, o, v=-0.93, l=$V_{EE}$] (0,0);
\node [red] at (0.2,-0.6) {8};
\end{circuitikz}
\end{figure}

\begin{enumerate}
\item let’s neglect (to be confirmed!) \( I_B, V_B \approx 0 \text{ V} \);
\item \( I_{E1Q} = (V_{E1Q} - V_{EE})/R_3 \);
\item \( I_{C1Q} \approx 0.93 \text{ mA} \);
\item \( V_{CC} - R_2 I_{C2Q} = 5.35 \text{ V}; V_{CE1Q} = 6.05 \text{ V}; \)
\end{enumerate}

\footnote{Notice that in the example the numbers in the references on the right are blue due to hyperref configuration!}
4 Personalize it!

The definition of \CircledText is simply the following one; you can get idea and define your own easily (beware of spaces at the end of the lines, though!):

\newcommand{\CircledText}[2][{}]{\CircledParamOpts{inner color=none, outer color=none, #1}{1}{#2}}

where \CircledParamOpts is a macro with three arguments: the first one is the list of options (without the /csteps/ part), the second one is 1 is you want the circle content on the baseline and 0 if you want it on top of it, and finally the third one is the text to be “circled”.

Also the implementation of the \cstep command and relatives is quite simple, and you can play a lot with it to change things (formats, colors, the type of numbering, and so on):

\newcounter{cstepcnt}
\newcommand{\startcstep}{\setcounter{cstepcnt}{0}}
\newcommand{\resetcstep}{\setcounter{cstepcnt}{0}}
\newcommand{\cstep}{\refstepcounter{cstepcnt}Circled{\scriptsize\arabic{cstepcnt}}}