The circledsteps package: circled numbers, circled steps and more

Romano Giannetti <romano.giannetti@gmail.com>

August 11, 2023

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 tikzpictures.

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 v1.3.3. The main changes between version are listed below.

v1.3.x Fixes to the manual; added a warning in case the patching of pict2e.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

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

v1.0 Initial release.

2 Basic commands

The basic commands are:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>\Circled[]{}</td>
<td>circled text using the package colors, text on the baseline</td>
</tr>
<tr>
<td>\CircledTop[]{}</td>
<td>circled text using the package colors, circle on the baseline</td>
</tr>
<tr>
<td>\CircledText[]{}</td>
<td>circled text using the current color¹</td>
</tr>
</tbody>
</table>

¹It may fail when used in headers or footers, in the sense it can select the wrong color in case of page breaks.
The parameters for the output are controlled using \texttt{\pgfkeys}; you can change them with \texttt{\pgfkeys} or \texttt{\tikzset}; you can obviously limit the effect of change using normal \LaTeX{} scoping rules. Also, you can specify parameters (without the initial \texttt{/csteps/} as optional arguments to the commands.

The pseudo-color \texttt{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>\texttt{/csteps/inner ysep}</td>
<td>vertical spacing</td>
<td>4pt</td>
</tr>
<tr>
<td>\texttt{/csteps/inner xsep}</td>
<td>horizontal spacing</td>
<td>4pt</td>
</tr>
<tr>
<td>\texttt{/csteps/inner color}</td>
<td>color of the text</td>
<td>none</td>
</tr>
<tr>
<td>\texttt{/csteps/outer color}</td>
<td>color of the circle</td>
<td>none</td>
</tr>
<tr>
<td>\texttt{/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 descenders, like p: p and p.
You can set options as optional arguments, like \texttt{R} or \texttt{g}.
Math mode is detected too: A and A

For big horizontal things the circle becomes oval: 200 or 199.

Inside \texttt{tikz} they work ok:

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

If you like more breathing space:

1
2 p 200 199.

If you want to change colors it’s easy:

1 2 p 200 199.

And you can have filled circles and ovals: 1 199.
You can have inline numbers like \(1\) or exponents\(2\) and so on. They follow the current color: \(1\) and \(2\) automatically, as you can see here: \(1 + 1 \approx 3\). In-text circled numbers look better when they are a bit smaller, though, as you can see in \(1\) for example.

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

\[
\begin{align*}
\int_{-1}^{1} f_1(x) \\
\int_{-1}^{1} f_2(x)
\end{align*}
\]

3 Automatically generated numbers

The command \texttt{\cstep} will generate a circled number, starting from 1, that can be referenced with the normal \texttt{\label}/\texttt{\ref} mechanism.\(^2\) You can reset the numbering with \texttt{\startcstep}. For example:

\begin{itemize}
\item \(V_{CC} = 10\) V
\item \(R_1 = 1\) k\(\Omega\)
\item \(R_2 = 5\) k\(\Omega\)
\item \(R_3 = 10\) k\(\Omega\)
\item \(C_1 = 0.93\) V
\item \(V_{EE} = -10\) V
\item \(V_{CC} - R_2 I_{C2Q} = 5.35\) V; \(V_{CE1Q} = 6.05\) V;
\item \(I_{E1Q} = (V_{E1Q} - V_{EE})/R_3\);
\item \(I_{C1Q} \approx 0.93\) mA;
\item \(V_{CC} \approx 0.93\) mA;
\end{itemize}

\(^2\) Notice that in the example the numbers in the references on the right are blue due to \texttt{hyperref} configuration!
4 Personalize it!

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

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

where \texttt{\CircledParamOpts} is a macro with three arguments: the first one is the list of options (without the /csteps/ part), the second one is 1 if 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 \texttt{\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):

\begin{verbatim}
\newcounter{cstepcnt}
\newcommand{\startcstep}{\setcounter{cstepcnt}{0}}
\newcommand{\resetcstep}{\setcounter{cstepcnt}{0}}
\newcommand{\cstep}{% \refstepcounter{cstepcnt} % \Circled{\scriptsize\arabic{cstepcnt}}% \renewcommand{\thecstepcnt}{\arabic{cstepcnt}}}
\end{verbatim}