

# LaTeX

## Constructing circuit diagrams with `pst-circ`\*

Herbert Voß

### Abstract

In this article the group of packages collectively referred to as `pstricks` is extended with a package which adds the ability to generate electronic circuit diagrams. These diagrams follow standard conventions in engineering and can be particularly handy when it comes to simple construction of circuit diagrams for publication, without having to come to grips with vector drawing programs.

### 1 The basic concept

The package `pst-circ` described here is in principle based on `pst-node` in which a graphical object is placed between reference points or “nodes” (Figure 1), while the underlying details of the object *per se* are secondary.

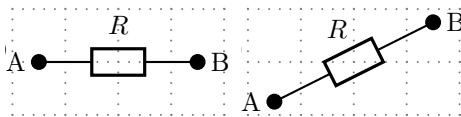
One can define such a coordinate pair two ways:

```
resistor(-1.5,0)(+1.5,0){R}
```

or

```
\pnode(-1.5,0){A}
\pnode(1.5,0){B}
resistor(A)(B){R}
```

where, in the second case, the location of A and B has been defined separately. When there is no need to explicitly to define a coordinate pair as a node, you can simply use the  $(x, y)$ -form.



**Figure 1:** Example displays

The text labels are displayed horizontally by default but can be rotated by invoking a specific parameter. Listing 1 illustrates the coding for Figure 1 (without the code which generated the coordinate axes and labels on the nodes).

Listing 1: Coding for example displays

```
1 \begin{pspicture}(-2,-1)(2,1)
```

\* Translated from *Die TEXnische Komödie 3/2003*, p. 33-49, with permission. Translation by Douglas Waud.

```
2 \pnode(-1.5,0){A}
3 \pnode(1.5,0){B}
4 \resistor(A)(B){R}
5 \end{pspicture}\hspace{0.5cm}
6 \begin{pspicture}(-2,-1)(2,1)
7 \pnode(-1.5,-0.75)
8 \pnode(1.5,0.75){B}
9 \resistor(A)(B){R}
10 \end{pspicture}
```

### 2 Circuit components

Tables 1 through 3 give summaries of the circuit components presently available (version 1.2). For purposes of description these are divided into

**Dipoles:** Two terminal circuit elements.

**Multidipoles:** Combinations of two terminal circuit elements.

**Tripoles:** Three terminal circuit elements.

**Quadrupoles:** Four terminal circuit elements.

So the tables can focus on the key features, possible options are not included. These are covered later in this article. Macros are called generally in the format

```
\<Object name>(<node 1>)(<node 2>)
... (<Label>)
```

The last argument (Label) can be empty. Some of the tripoles do not make this argument available, so a label must be generated separately.

#### 2.1 Two terminal circuit elements

See Table 1 for the dipole elements available. Here we have the greatest number of available objects. All but `\wire` can have a label specified.

The symbol `\circledipole` can be used, in particular, for display of current and voltage sources as variations on the battery symbol. With the `\labeloffset=0` option, one can place the label in the center of the circle as, for example, in Figure 11.

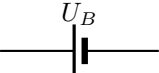
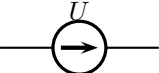
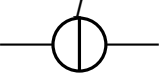
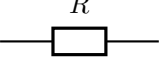
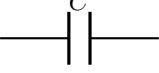

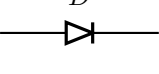
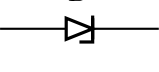


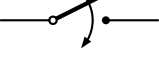

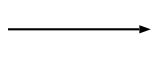

#### 2.2 Multidipole

This is nothing more than a linear chain of dipoles which can lead to a simplification of the coding. Thus one can, for example, represent a real inductance simply by defining a new macro as a multipole as in Figure 2.

Listing 2: Code for the multidipole of figure 2

```
1 \begin{pspicture}(-2.75,-0.5)(2.75,1)
2 \pnode(-2.75,0){A}
3 \pnode(2.75,0){B}
4 \multidipole(A)(B)%
5 \coil{L}%
6 \resistor{R}.
7 \end{pspicture}
```

Table 1: Predefined two terminal circuit elements

Name	Macro	Output
Battery	<code>\battery</code>	
Voltage source	<code>\Ucc</code>	
Current source	<code>\Icc</code>	
Resistance	<code>\resistor</code>	
Capacitance	<code>\capacitor</code>	
Inductance	<code>\coil</code>	
Diode	<code>\diode</code>	
Zener-Diode	<code>\Zener</code>	
LED	<code>\LED</code>	
Lamp	<code>\lamp</code>	
Switch	<code>\switch</code>	
Wire	<code>\wire</code>	
Arrow	<code>\tension</code>	
Circle	<code>\circledipole</code>	

Note that the period at the end of line 6 ends the definition of the multipole. The number of dipoles has no hard limit. The size of the drawing space presents the practical limit.

### 2.3 Three terminal circuit elements

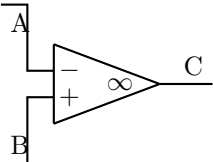
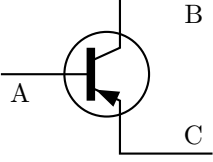
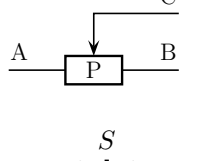
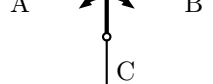
In Table 2 the three terminals have the names of the labels added to the sample displays. This makes it clearer what the order of the (A), (B) and (C) calls in the coding refers to. A label is available only for switches.



**Figure 2:** Code defining a multipole and the result

For the other two, one must use the `\uput` command (see Listing 1).

Table 2: Predefined three terminal circuit elements

Name	Macro	Graphic
Op Amp	<code>\OA</code>	
pnp Transistor	<code>\transistor</code>	
Potentiometer	<code>\potentiometer</code>	
SPDT switch	<code>\Tswitch</code>	

### 2.4 Four terminal circuit elements

In Table 3 the names of the nodes are again superimposed on the example diagrams as (A), (B), (C), and (D) to facilitate matching the coding. The order of these nodes is meaningful since the calls all involve (A)(B)(C)(D) in a specific order.

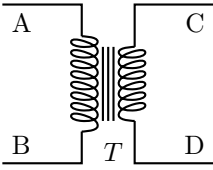
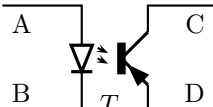
## 3 Options

The package `pst-circ` itself has no options of any kind; in contrast, the macros have a considerable number, including options for color displays which will be particularly useful in PDF output.

### 3.1 Circuit direction arrows

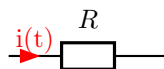
Each object can be provided with an arrow indicating direction of current. The example in Figure 3 illustrates use of all the available options for the characterization

Table 3: Predefined four terminal circuit element

Name	Macro	Graphic
Trans- former	<code>\transformer</code>	
Opto- coupler	<code>\optoCoupler</code>	

of the arrow. The default values for these options can be obtained from the `pst-circ` documentation.

The direction of current is determined by the order of the nodes—in the previous example of coding, from (A) towards (B). This can, however, be changed with the option `\directconvention=false`.



```

1 \begin{pspicture}(-2,-1)(2,1)
2   \nnode(-1.5,0){A}
3   \nnode(1.5,0){B}
4   \resistor[%
5     intensity=true,%
6     intensitycolor=red,%
7     intensitylabel=i(t),%
8     intensitylabelcolor=red,%
9     intensitylabeloffset=0.3,%
10    intensitywidth=2pt](A)(B){R}
11 \end{pspicture}

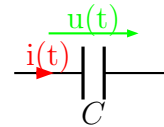
```

Figure 3: Indicating current direction

### 3.2 Arrows indicating potential differences

Analogous to an arrow showing current direction, each object can have an arrow to indicate a potential difference. The example in Figure 4 illustrates use of all the available options for the characterization of the arrow. The default values for these options can be obtained from the `pst-circ` documentation. The listing shows only the options for the potential difference arrow; those for the current arrow would be the same as in the preceding example.

This example also shows that the label, “C”, which normally appears above the object, can also appear below.



```

1 \begin{pspicture}(-2,-1)(2,1.5)
2   \nnode(-1.5,0){A}
3   \nnode(1.5,0){B}
4   \capacitor[%
5     labeloffset=-0.75,%
6     tension=true,%
7     tensioncolor=green,%
8     tensionoffset=0.75,%
9     tensionlabel=u(t),%
10    tensionlabelcolor=green,%
11    tensionlabeloffset=1,%
12    tensionwidth=\pslinewidth](A)(B){C}
13 \end{pspicture}

```

Figure 4: Displaying a potential difference

indicating potential is, like that for current, directed from B to A.

If, for example, an alternative convention is desired, arrows can be easily reversed with the options

`dipoleconvention=generator|receptor`

where `load` is the default. Also both arrows can be jointly reversed with the option

`directconvention=false|true`

### 3.3 Parallel circuits

This case comes up frequently, for example, in the equivalent circuit for a real capacitor, so `pst-circ` provides extra options for it. Figure 5 shows a simple example with all the options including those in the preceding examples (but without repetition of the code already presented).

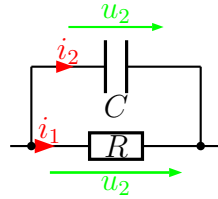
The idea is that both objects themselves have nodes but one of the two, with the option `parallel`, is placed above or below the other. A negative value for `parallelarm` would place the object under the other; for example, with the specific value `-2`, the separation would be 2 units. Similarly, parallel connections with three objects are possible.

### 3.4 Alternative display forms

Different conventions for presentation frequently exist, particularly between European and American symbols. Most can be handled by the options summarized in Table 4.

### 3.5 Variable elements

With the option `variable=true`, diagonal arrows are added to indicate the component (resistor, inductance, or capacitor) is variable, as illustrated in Figure 6:



```

1 \begin{pspicture}(-2,-1)(2,3)
2 \node(-2,0){A}
3 \node(2,0){B}
4 \resistor[%
5   labeloffset=0,%
6   [ ... ]
7   tensionwidth=1pt](A)(B){R$}
8 \capacitor[%
9   labeloffset=0,%
10  parallel=true,%
11  parallelnode=true,%
12  parallesep=0.2,%
13  parallelarm=1.5,%
14  [ ... ]
15  tensionwidth=\pslinewidth](A)(B){C$}
16 \end{pspicture}

```

Figure 5: A parallel circuit

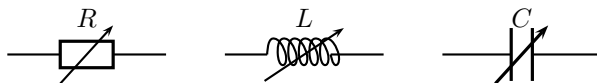


Figure 6: Adjustable objects

### 3.6 Transistors

The transistor in Table 1 was presented in bare-bones fashion. Figure 7 shows further options.

- The boolean option `Transistorinvert` reverses the emitter and collector connections.
- The higher-level option `intensity=true` sets all three current arrows for base/emitter/collector to `true`.
- The colors for the current arrows can be set globally with the option `intensitycolor`, and for labels with `intensitylabelcolor`.

### 3.7 Operational amplifier

Table 1 illustrated a standard operational amplifier with infinite gain. Figure 8 illustrates some of the possible options. Here, as with the transistor, one can simplify coding with the higher level option `intensity` to control all three current arrows.

With the option `tripolestyle=french` one can also invoke the alternative presentation illustrated in Figure 9.

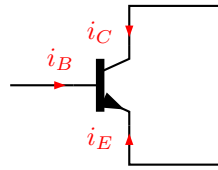
Table 4: Alternative formats

Macro	dipolestyle=...	Graphic
<code>\resistor -</code>		
	<code>zigzag</code>	
<code>\coil -</code>		
	<code>rectangle</code>	
	<code>curved</code>	
	<code>elektor</code>	
	<code>elektorcurved</code>	
<code>\capacitor -</code>		
	<code>chemical</code>	
	<code>elektor</code>	
	<code>elektorchemical</code>	
<code>\diode -</code>		
	<code>thyristor</code>	
	<code>GTO</code>	
	<code>triac</code>	

A final option, `OAinvert`, makes it possible to reverse the connections of the two inputs.

### 3.8 Transformers

Finally Figure 10 illustrates the options available for display of transformers. Again the labels on the current

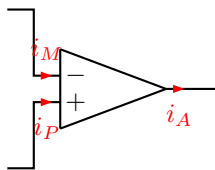


```

1 \begin{pspicture}(4,3)
2   \pnode(0,1.5){A}\uput[-45](A){A}
3   \pnode(4,3){B}\uput[-135](B){B}
4   \pnode(4,0){C}\uput[135](C){C}
5   \transistor[%
6     transistorcircle=false,%
7     transistortype=NPN,%
8     transistoribase=true,%
9     transistoricollector=true,%
10    transistoriemitter=true,%
11    transistoribaselabel={\red $i_B$},%
12    transistoricollectorlabel=$i_C$,%
13    transistoriemitterlabel=$i_E$,%
14    intensitycolor=red,%
15    intensitylabelcolor=red](A)(B)(C)
16 \end{pspicture}

```

Figure 7: Transistor options



```

1 \begin{pspicture}(4,3.5)
2   \pnode(0,3){A}
3   \pnode(0,0){B}
4   \pnode(4,1.5){C}
5   \OA[%
6     OApertect=false,%
7     OAiplus=true,%
8     OAiminus=true,%
9     OAiout=true,%
10    OAipluslabel=$i_P$,%
11    OAiminuslabel=$i_M$,%
12    OAioutlabel=$i_A$,%
13    intensitycolor=red,%
14    intensitylabelcolor=red](A
15    )(B)(C)
16 \end{pspicture}

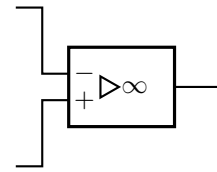
```

Figure 8: Operational amplifier options

arrows can be controlled jointly with the higher level option `intensity`. Furthermore, the German convention of using rectangles for the windings can be chosen with the option `dipole-style=rectangle` to match the parallel usage shown earlier with inductances.

#### 4 Example of use for a complex diagram

With `pst-circ` one cannot readily design extremely complex circuits. Still one can easily display small circuits or equivalent circuits for real devices if one works in the framework of a coordinate system such as that

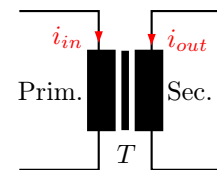


```

1 \begin{pspicture}(4,3.5)
2   \pnode(0,3){A}
3   \pnode(0,0){B}
4   \pnode(4,1.5){C}
5   \OA[tripolestyle=french](A)(B)(C)
6 \end{pspicture}

```

Figure 9: Alternative format for an operational amplifier



```

1 \begin{pspicture}(4,4)
2   \pnode(0,3){A}
3   \pnode(0,0){B}
4   \pnode(4,3){C}
5   \pnode(4,0){D}
6   \transformer[%
7     dipolestyle=rectangle,%
8     primarylabel={Prim.},%
9     secondarylabel={Sec.},%
10    transformerprimary=true,%
11    transformersecondary=true,%
12    transformerprimarylabel=$i_{in}$,%
13    transformersecondarylabel=$i_{out}$,%
14    intensitycolor=red,%
15    intensitylabelcolor=red](A)(B)(C)(D){T$}
16 \end{pspicture}

```

Figure 10: Transformer options

provided by `psgrid`. Figure 11 displays the equivalent circuit for a constant current generator as produced by the code in Listing 3.

Listing 3: Code for Figure 11

```

1 \psset{intensitycolor=red,%
2   intensitylabelcolor=red,%
3   tensioncolor=green,%
4   tensionlabelcolor=green,%
5   intensitywidth=3pt}%
6 \begin{pspicture}(-1.5,-0.2)(13.5,9)

```

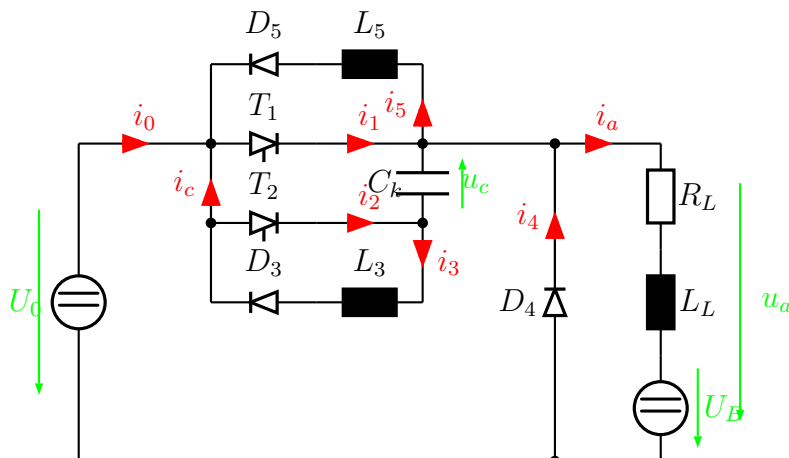


Figure 11: Example of capabilities of pst-circ

```

7 \psgrid[griddots=5,gridlabels=7pt,subgriddiv=0]
8 \circledipole[
9   tension,%
10  tensionlabel=$U_0$,%
11  tensionoffset=0.75,%
12  labeloffset=0](0,0)(0,6){\LARGE\textbf{=}}
13 \wire[intensity,intensitylabel=$i_0$](0,6)
14   (2.5,6)
15 \diode[dipolestyle=thyristor](2.5,6)(4.5,6){$T_1$}
16 \wire[intensity,intensitylabel=$i_1$](4.5,6)
17   (6.5,6)
18 \multidipole(6.5,7.5)(2.5,7.5)%
19 \coil[dipolestyle=rectangle,labeloffset
20   =-0.75]{$L_5$}%
21 \diode[labeloffset=-0.75]{$D_5$}.
22 \wire[intensity,intensitylabel=$i_5$](6.5,6)
23   (6.5,7.5)
24 \wire(2.5,7.5)(2.5,3)
25 \wire[intensity,intensitylabel=$i_c$](2.5,4.5)
26   (2.5,6)
27 \qdisk(2.5,6){2pt}\qdisk(6.5,6){2pt}
28 \diode[dipolestyle=thyristor](2.5,4.5)(4.5,4.5)
29   {$T_2$}
30 \wire[intensity,intensitylabel=$i_2$](4.5,4.5)
31   (6.5,4.5)
32 \capacitor[tension,tensionlabel=$u_c$,%
33  tensionoffset=-0.75,tensionlabeloffset
34   =-1](6.5,4.5)(6.5,6){$C_k$}
35 \qdisk(2.5,4.5){2pt}\qdisk(6.5,4.5){2pt}
36 \wire[intensity,intensitylabel=$i_3$](6.5,4.5)
37   (6.5,3)
38 \multidipole(6.5,3)(2.5,3)%
39 \coil[dipolestyle=rectangle,labeloffset
40   =-0.75]{$L_3$}%
41 \diode[labeloffset=-0.75]{$D_3$}.
42 \wire(6.5,6)(9,6)\qdisk(9,6){2pt}
43 \diode(9,0)(9,6){$D_4$}
44 \wire[intensity,intensitylabel=$i_4$](9,3.25)
45   (9,6)
46 \wire[intensity,intensitylabel=$i_a$](9,6)(11,6)
47 \multidipole(11,6)(11,0)%
48 \resistor{$R_L$}
49 \coil[dipolestyle=rectangle]{$L_L$}%
50 \circledipole[labeloffset=0,%
51  tension,tensionoffset=0.7,%
52  tensionlabel=$U_B$]{\LARGE\textbf{=}}.
53 \wire(0,0)(11,0)\qdisk(9,0){2pt}
54 \tension(12.5,5.5)(12.5,0.5){$u_a$}
55 \end{pspicture}

```

The package `pst-circ` is a great help with switching circuit diagrams of reasonable complexity, in the same way as noted earlier with equivalent circuits.

## 5 PDF output

Since the package `pst-circ` [1] is based on PostScript [3] (like all `pstricks` [8] packages), it cannot be used directly with `pdfTEX`. However, there are a number of alternative ways to get PDF output:

- The `pdftricks` package [7]; unfortunately, this very frequently, because of the use of PostScript, leads to problems with bounding boxes.
- The `ps4pdf` package [5, 6], which requires installation of the  $\LaTeX$  package `preview` [2].
- The program `VTeX`[4] (free for Linux and OS/2).
- The program `ps2pdf` with the sequential conversions `dvi`  $\rightarrow$  `ps`  $\rightarrow$  `pdf`.

`ps4pdf` also offers the possibility of storing figures generated with `pstricks` in a single PDF or EPS file.

## 6 Summary

This article describes most of the common features of the `pst-circ` package. Features not mentioned, for example

crossing of connections and use of switches, can be found in the package documentation.

## References

- [1] Christophe Jorssen and Herbert Voß. *pst-circ: PostScript macros for drawing electronic circuits*. <http://ctan.tug.org/tex-archive/graphics/pstricks/contrib/pst-circ/>, 2003.
- [2] David Kastrup. *preview-latex*. <http://ctan.tug.org/tex-archive/support/preview-latex/>, 2003.
- [3] Nikolai G. Kollock. *PostScript richtig eingesetzt: vom Konzept zum praktischen Einsatz*. IWT, Vaterstetten, 1989.
- [4] Micropress. *VTeX/Lnx*. <http://www.micropress-inc.com/linux/>, 2003.
- [5] Rolf Niepraschk. *ps4pdf*. <http://ctan.tug.org/tex-archive/macros/latex/contrib/ps4pdf/>, 2003.
- [6] Rolf Niepraschk and Herbert Voß. The package *ps4pdf*: from PostScript to PDF. *TUGboat*, 22:290–292, December 2001.
- [7] Herbert Voß. *PSTricks Support for PDF*. <http://www.pstricks.de/pdf/pdftricks.phtml>, 2002.
- [8] Timothy Van Zandt. *pstricks: PostScript macros for generic TEX*. <http://www.tug.org/applications/PSTricks/>, 1993.

◇ Herbert Voß  
Wasenstr. 21  
14129 Berlin  
Germany  
[voss@perce.de](mailto:voss@perce.de)  
<http://www.perce.de>