

CircuiTikZ

version 0.4

Massimo A. Redaelli

May 1, 2015

Contents

1	Introduction	2
1.1	About	3
1.2	Loading the package	3
1.3	License	3
1.4	Feedback	3
1.5	Requirements	3
1.6	Incompatible packages	3
1.7	Introduction to version 0.3.0	4
1.8	Introduction to version 0.2.3	4
1.9	ConT _E Xt compatibility	4
2	Options	5
3	The components	6
3.1	Monopoles	7
3.2	Bipoles	9
3.2.1	Instruments	9
3.2.2	Basic resistive bipoles	9
3.2.3	Resistors and the like	10
3.2.4	Stationary sources	11
3.2.5	Diodes and such	12
3.2.6	Basic dynamical bipoles	14
3.2.7	Sinusoidal sources	16
3.2.8	Square sources	16
3.2.9	Switch	16
3.2.10	HF components	16
3.3	Tripoles	19
3.3.1	Controlled sources	19
3.3.2	Transistors	20
3.3.3	HF	22
3.3.4	Switch	23
3.3.5	Other bipole-like tripoles	23
3.3.6	Misc	23
3.4	Double bipoles	23
3.5	Logic gates	25

3.6	Amplifiers	27
3.7	Support shapes	28
4	Usage	28
4.1	Labels	29
4.2	Currents	30
4.3	Voltages	31
4.3.1	European style	31
4.3.2	American style	31
4.4	Nodes	32
4.5	Special components	32
4.6	Integration with <code>siunitx</code>	34
4.7	Mirroring	35
4.8	Putting them together	35
5	Not only bipoles	36
5.1	Anchors	36
5.1.1	Logical ports	36
5.1.2	Transistors	36
5.1.3	Other tripoles	38
5.1.4	Operational amplifier	39
5.1.5	Double bipoles	40
5.2	Input arrows	41
5.3	Transistor paths	42
6	Customization	42
6.1	Parameters	42
6.2	Components size	44
6.3	Colors	45
7	FAQ	47
8	Examples	47
9	Revision history	52
	Index of the components	55

1 Introduction

After two years of little exposure only on my personal website¹, I did a major rehaul-
ing of the code of `CircuitikZ`, fixing several problems and converting everything to
TikZ version 2.0.

I'm not too sure about the result, because my (La)TeX skills are much to be
improved, but it seems it's time for more user feedback. So, here it is...

I know the documentation is somewhat scant. Hope to have time to improve it a
bit.

¹Now the package is moved to its own git repository: <https://github.com/mredaelli/circuitikz>. Contributions are welcome.

1.1 About

This package provides a set of macros for naturally typesetting electrical and (somewhat less naturally, perhaps) electronical networks.

It was born mainly for writing my own exercise book and exams sheets for the Elettrotecnica courses at Politecnico di Milano, Italy. I wanted a tool that was easy to use, with a lean syntax, native to \LaTeX , and supporting directly PDF output format.

So I based everything with the very impressive (if somewhat verbose at times) TikZ package.

1.2 Loading the package

```
\usepackage{circuitikz}
```

TikZ will be automatically loaded.

1.3 License

Copyright © 2007–2011 Massimo Redaelli. This package is author-maintained. Permission is granted to copy, distribute and/or modify this software under the terms of the \LaTeX Project Public License, version 1.3.1, or the GNU Public License. This software is provided ‘as is’, without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

1.4 Feedback

Much appreciated: ✉ <mailto:m.redaelli@gmail.com>. Although I don’t guarantee quick answers.

1.5 Requirements

- tikz, version ≥ 2 ;
- xstring, not older than 2009/03/13;
- siunitx, if using siunitx option.

1.6 Incompatible packages

TikZ’s own circuit library, which is based on CircuiTikZ, (re?)defines several styles used by this library. In order to have them work together I added the `compatibility` package option, which basically prefixes the names of all CircuiTikZ `to[]` styles with an asterisk.

So, if loaded with said option, one must write `(0,0) to[*R] (2,0)` and, for transistors on a path, `(0,0) to[*Tnmos] (2,0)`, and so on (but `(0,0) node [nmos] {}`). See example at page ??.

1.7 Introduction to version 0.3.0

Probably nobody is hoping or caring for a new version of the package at this point, seeing how long it took me for this next release. But here it is, fixing a big problem (voltage labels in the wrong place, in some cases) and adding several components.

Thanks for bug reporting and suggesting improvements.

1.8 Introduction to version 0.2.3

Having waited a long time before updating the package, many feature requests piled on my desk. They should all be implemented now.

There are a number of backward incompatibilities—I'm sorry, but I had to make a choice in order not to have a schizophrenic interface. They are mostly, in my opinion, minor problems that can be dealt with with appropriate package options:

- `potentiometer` is now the standard resistor-with-arrow-in-the-middle; the old potentiometer is now known as `variable resistor` (or `vR`), similarly to `variable inductor` and `variable capacitor`;
- `american inductor` was not really the standard american inductor. The old american inductor has been renamed `cute inductor`;
- `transformer`, `transformer core` and `variable inductor` are now linked with the chosen type of inductor;
- styles for selecting shape variants (like `[american resistors]`) are now in the plural to avoid conflict with paths (like `to[american resistor]`).

1.9 ConT_EXt compatibility

As requested by some users, I fixed the package for it to be compatible with ConT_EXt. Just use `\usemodule[circuitikz]` in your preamble and include the code between `\startcircuitikz` and `\endcircuitikz`. Please notice that the package `siunitx` is *not* available for ConT_EXt: the option `siunitx` simply defines a few measurement units typical in electric sciences.

In actually using CircuiTikZ with TikZ version 2 in ConT_EXt an error comes up, saying something like

```
! Undefined control sequence.
\tikz@cc@mid@checks -> \pgfutil@ifnextchar!
```

The solution has been suggested to me by Aditya Mahajan, and involves modifying a file in TikZ:

Here is the fix. In `tikzlibrarycalc.code.tex` change

```
\def\tikz@cc@mid@checks{
  \pgfutil@ifnextchar !{%AM: Added space
    \tikz@cc@mid%
  }{%
    \advance\pgf@xa by\tikz@cc@factor\pgf@xb%
    \advance\pgf@ya by\tikz@cc@factor\pgf@yb%
```

```

        \tikz@cc@parse% continue
    }%
}

\def\tikz@cc@mid !{%AM Added space
    \pgfutil@ifnextchar({%
        \tikz@scan@one@point\tikz@cc@project%
    }{%
        \tikz@cc@mid@num%
    }%
}

```

As far as I know, this is a bug in TikZ, and I notified the author, but until he fixes it, you know the workaround.

2 Options

- `europenvoltages`: uses arrows to define voltages, and uses european-style voltage sources;
- `americenvoltages`: uses $-$ and $+$ to define voltages, and uses american-style voltage sources;
- `europencurrents`: uses european-style current sources;
- `americancurrents`: uses american-style current sources;
- `europenresistors`: uses rectangular empty shape for resistors, as per european standards;
- `americanresistors`: uses zig-zag shape for resistors, as per american standards;
- `europeninductors`: uses rectangular filled shape for inductors, as per european standards;
- `americaninductors`: uses "4-bumps" shape for inductors, as per american standards;
- `cuteinductors`: uses my personal favorite, "pig-tailed" shape for inductors;
- `americanports`: uses triangular logic ports, as per american standards;
- `europenports`: uses rectangular logic ports, as per european standards;
- `americangfsurgearrester`: uses round gas filled surge arresters, as per american standards;
- `europengfsurgearrester`: uses rectangular gas filled surge arresters, as per european standards;
- `european`: equivalent to `europencurrents`, `europenvoltages`, `europenresistors`, `europeninductors`, `europenports`, `europengfsurgearrester`;

- `american`: equivalent to `americancurrents`, `americanvoltages`, `americanresistors`, `americaninductors`, `americanports`, `americangfsurgearrester`;
- `siunitx`: integrates with `SIunitx` package. If labels, currents or voltages are of the form `#1<#2>` then what is shown is actually `\SI{#1}{#2}`;
- `nosiunitx`: labels are not interpreted as above;
- `fulldiodes`: the various diodes are drawn *and* filled by default, i.e. when using styles such as `diode`, `D`, `sD`, ... Un-filled diode can always be forced with `Do`, `sDo`, ...
- `emptydiodes`: the various diodes are drawn *but not* filled by default, i.e. when using styles such as `diode`, `D`, `sD`, ... Filled diode can always be forced with `D*`, `sD*`, ...
- `arrowmos`: pmos and nmos have arrows analogous to those of pnp and npn transistors;
- `noarrowmos`: pmos and nmos do not have arrows analogous to those of pnp and npn transistors;
- `straightlabels`: labels on bipoles are always printed straight up, i.e. with horizontal baseline;
- `rotatelabels`: labels on bipoles are always printed aligned along the bipole;
- `smartlabels`: labels on bipoles are rotated along the bipoles, unless the rotation is very close to multiples of 90°;
- `compatibility`: makes it possible to load `CircuiTikZ` and `TikZ` circuit library together.

The old options in the singular (like `american voltage`) are still available for compatibility, but are discouraged.

Loading the package with no options is equivalent to my own personal liking, that is to the following options:

```
[europeancurrents, europeanvoltages, americanresistors, cuteinductors,
americanports, nosiunitx, noarrowmos, smartlabels].
```

In `ConTeXt` the options are similarly specified: `current=european|american`, `voltage=european|american`, `resistor=american|european`, `inductor=cute|american|european`, `logic=american|european`, `siunitx=true|false`, `arrowmos=false|true`.

3 The components

Here follows the list of all the shapes defined by `CircuiTikZ`. These are all `pgf` nodes, so they are usable in both `pgf` and `TikZ`.

Each bipole (plus triac and thyristors) are shown using the following command, where `#1` is the name of the component²:

²If `#1` is the name of the bipole/the style, then the actual name of the shape is `#1shape`.

```
\begin{center}\begin{circuitikz} \draw
  (0,0) to[ #1 ] (2,0)
; \end{circuitikz} \end{center}
```

The other shapes are shown with:

```
\begin{center}\begin{circuitikz} \draw
  (0,0) node[ #1 ] {}
; \end{circuitikz} \end{center}
```

Please notice that for user convenience transistors can also be inputted using the syntax for bipoles. See section 5.3.

If using the `\tikzexternalize` feature, as of Tikz 2.1 all pictures must end with `\end{tikzpicture}`. Thus you *cannot* use the `circuitikz` environment. Which is ok: just use `tikzpicture`: everything will work there just fine.

3.1 Monopoles

- Ground (ground)



- Reference ground (rground)



- Signal ground (sground)



- Noiseless ground (nground)



- Protective ground (pground)



- Chassis ground³ (cground)

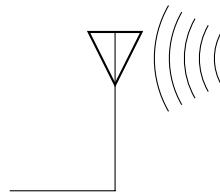


³These last three were contributed by Luigi «Liverpool»

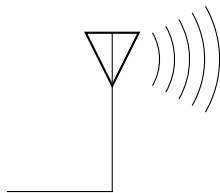
- Antenna (antenna)



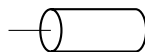
- Receiving antenna (rxantenna)



- Transmitting antenna (txantenna)



- Transmission line stub (tlinestub)



- VCC (vcc)



- match (match)



- LO⁴ (oscillator)



⁴These last three come from Stefan Erhardt's contribution of HF components

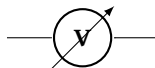
3.2 Bipoles

3.2.1 Instruments

- Ammeter (ammeter)



- Voltmeter (voltmeter)



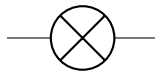
3.2.2 Basic resistive bipoles

- Short circuit (short)

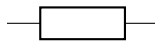


- Open circuit (open)

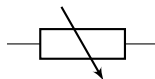
- Lamp (lamp)



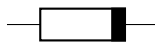
- Generic (symmetric) bipole (generic)



- Tunable generic bipole (tgeneric)



- Generic asymmetric bipole (ageneric)



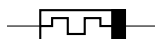
- Generic asymmetric bipole (full) (fullgeneric)



- Tunable generic bipole (full) (tfullgeneric)



- Memristor (memristor, or Mr)



3.2.3 Resistors and the like

If (default behaviour) `americanresistors` option is active (or the style `[american resistors]` is used), the resistor is displayed as follows:

- Resistor (`R`, or `american resistor`)



- Variable resistor (`vR`, or `variable american resistor`)

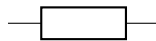


- Potentiometer (`pR`, or `american potentiometer`)

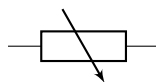


If instead `europenresistors` option is active (or the style `[european resistors]` is used), the resistors, variable resistors and potentiometers are displayed as follows:

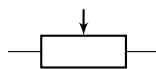
- Resistor (`R`, or `european resistor`)



- Variable resistor (`vR`, or `european variable resistor`)

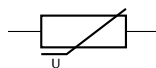


- Potentiometer (`pR`, or `european potentiometer`)

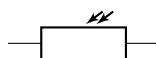


Other miscellaneous resistor-like devices:

- Varistor (`varistor`)



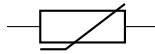
- Photoresistor (`phR`, or `photoresistor`)



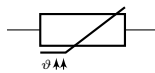
- Thermocouple (thermocouple)



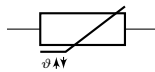
- Thermistor (thR, or thermistor)



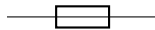
- PTC thermistor (thRp, or thermistor ptc)



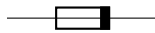
- NTC thermistor (thRn, or thermistor ntc)



- Fuse (fuse)



- Asymmetric fuse (afuse, or asymmetric fuse)



3.2.4 Stationary sources

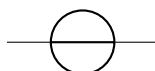
- Battery (battery)



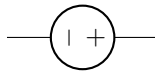
- Single battery cell (battery1)



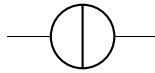
- Voltage source (european style) (european voltage source)



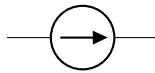
- Voltage source (american style) (american voltage source)



- Current source (european style) (european current source)



- Current source (american style) (american current source)

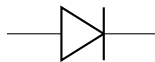


If (default behaviour) `européancurrents` option is active (or the style `[european currents]` is used), the shorthands `current source`, `isource`, and `I` are equivalent to european current source. Otherwise, if `americancurrents` option is active (or the style `[american currents]` is used) they are equivalent to american current source.

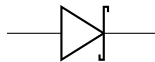
Similarly, if (default behaviour) `européanvoltages` option is active (or the style `[european voltages]` is used), the shorthands `voltage source`, `vsources`, and `V` are equivalent to european voltage source. Otherwise, if `americanvoltages` option is active (or the style `[american voltages]` is used) they are equivalent to american voltage source.

3.2.5 Diodes and such

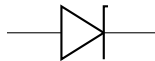
- Empty diode (empty diode, or `Do`)



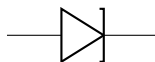
- Empty Schottky diode (empty Schottky diode, or `sDo`)



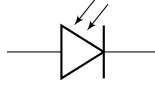
- Empty Zener diode (empty Zener diode, or `zDo`)



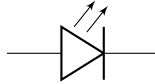
- Empty tunnel diode (empty tunnel diode, or `tDo`)



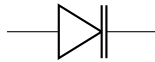
- Empty photodiode (empty photodiode, or pDo)



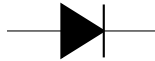
- Empty led (empty led, or leDo)



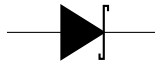
- Empty varcap (empty varcap, or VCo)



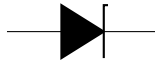
- Full diode (full diode, or D*)



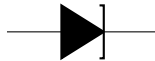
- Full Schottky diode (full Schottky diode, or sD*)



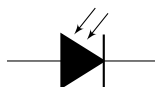
- Full Zener diode (full Zener diode, or zD*)



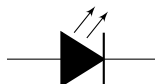
- Full tunnel diode (full tunnel diode, or tD*)



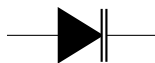
- Full photodiode (full photodiode, or pD*)



- Full led (full led, or leD*)

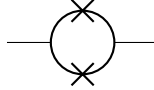


- Full varcap (full varcap, or VC*)



The options `fulldiodes` and `emptydiodes` (and the styles `[full diodes]` and `[empty diodes]`) define which shape will be used by abbreviated commands such that `D`, `sD`, `zD`, `tD`, `pD`, `leD`, and `VC`.

- Squid (`squid`)



- Barrier (`barrier`)



- European gas filled surge arrester (`european gas filled surge arrester`)



- American gas filled surge arrester (`american gas filled surge arrester`)

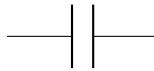


If (default behaviour) `europeangfsurgearrester` option is active (or the style `[european gas filled surge arrester]` is used), the shorthands `gas filled surge arrester` and `gf surge arrester` are equivalent to the european version of the component.

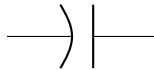
If otherwise `americangfsurgearrester` option is active (or the style `[american gas filled surge arrester]` is used), the shorthands `gas filled surge arrester` and `gf surge arrester` are equivalent to the american version of the component.

3.2.6 Basic dynamical bipoles

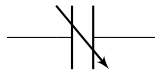
- Capacitor (`capacitor`, or `C`)



- Polar capacitor (`polar capacitor`, or `pC`)

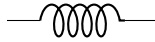


- Variable capacitor (`variable capacitor`, or `vC`)



If (default behaviour) `cuteinductors` option is active (or the style `[cute inductors]` is used), the inductors are displayed as follows:

- Inductor (`L`, or `cute inductor`)



- Variable inductor (`vL`, or `variable cute inductor`)



If `americaninductors` option is active (or the style `[american inductors]` is used), the inductors are displayed as follows:

- Inductor (`L`, or `american inductor`)



- Variable inductor (`vL`, or `variable american inductor`)



Finally, if `europeaninductors` option is active (or the style `[european inductors]` is used), the inductors are displayed as follows:

- Inductor (`L`, or `european inductor`)

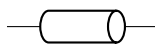


- Variable inductor (`vL`, or `variable european inductor`)



There is also a transmission line:

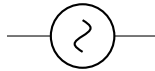
- Transmission line (`TL`, or `transmission line`, `tline`)



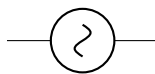
3.2.7 Sinusoidal sources

Here because I was asked for them. But how do you distinguish one from the other?!

- Sinusoidal voltage source (sinusoidal voltage source, or vsourcesin, sV)

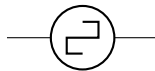


- Sinusoidal current source (sinusoidal current source, or isourcesin, sI)



3.2.8 Square sources

- Square voltage source (square voltage source, or vsourcesquare, sqV)



3.2.9 Switch

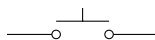
- Closing switch (closing switch, or cspst)



- Opening switch (opening switch, or ospst)



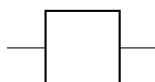
- Push button (push button)




3.2.10 HF components

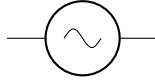
Contributed by Stefan Erhardt.

- generic two port⁵ (twoport)

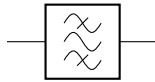


⁵To specify text to be put in the component: twoport[t=text]: 

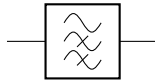
- vco (vco)



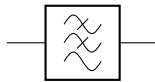
- bandpass (bandpass)



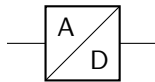
- highpass (highpass)



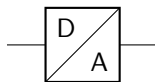
- lowpass (lowpass)



- A/D converter (adc)



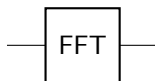
- D/A converter (dac)



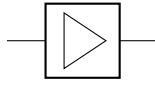
- DSP (dsp)



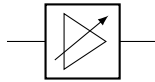
- FFT (fft)



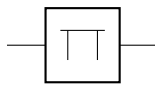
- amplifier (amp)



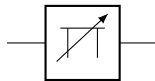
- VGA (vamp)



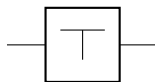
- π attenuator (piattenuator)



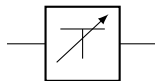
- var. π attenuator (vpiattenuator)



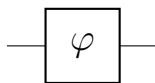
- T attenuator (tattenuator)



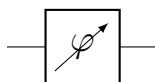
- var. T attenuator (vtattenuator)



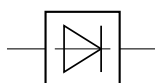
- phase shifter (phaseshifter)



- var. phase shifter (vphaseshifter)



- detector (detector)

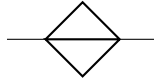


3.3 Tripoles

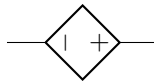
3.3.1 Controlled sources

Admittedly, graphically they are bipoles. But I couldn't...

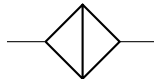
- Controlled voltage source (european style) (european controlled voltage source)



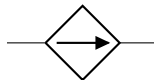
- Controlled voltage source (american style) (american controlled voltage source)



- Controlled current source (european style) (european controlled current source)



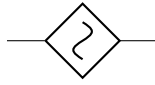
- Controlled current source (american style) (american controlled current source)



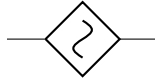
If (default behaviour) `européancurrents` option is active (or the style `[european currents]` is used), the shorthands `controlled current source`, `cisource`, and `cI` are equivalent to european controlled current source. Otherwise, if `americancurrents` option is active (or the style `[american currents]` is used) they are equivalent to american controlled current source.

Similarly, if (default behaviour) `européanvoltages` option is active (or the style `[european voltages]` is used), the shorthands `controlled voltage source`, `cvsource`, and `cV` are equivalent to european controlled voltage source. Otherwise, if `americanvoltages` option is active (or the style `[american voltages]` is used) they are equivalent to american controlled voltage source.

- Controlled sinusoidal voltage source (controlled sinusoidal voltage source, or controlled `vsourcesin`, `cvsourcesin`, `csV`)

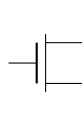


- Controlled sinusoidal current source (controlled sinusoidal current source, or controlled isourcesin, cisourcesin, csI)

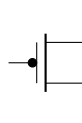


3.3.2 Transistors

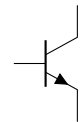
- nmos (nmos)



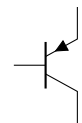
- pmos (pmos)



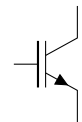
- npn (nnp)



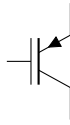
- pnp (pnp)



- npigt (nigt)

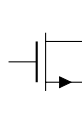


- pigbt (pigbt)

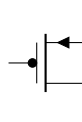


If the option `arrowmos` is used (or after the command `\ctikzset{tripoles/mos style/arrows}` is given), this is the output:

- `nmos (nmos)`

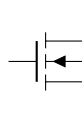


- `pmos (pmos)`

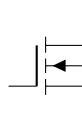


`nfet` and `pfet` have been incorporated based on code provided by Clemens Helfmeier and Theodor Borsche:

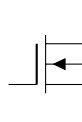
- `nfet (nfet)`



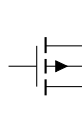
- `nigfete (nigfete)`



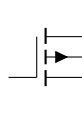
- `nigfetd (nigfetd)`



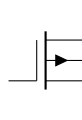
- `pfet (pfet)`



- pigfete (pigfete)

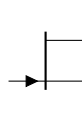


- pigfetd (pigfetd)

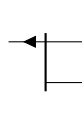


njfet and pjfet have been incorporated based on code provided by Danilo Piazzalunga:

- njfet (njfet)

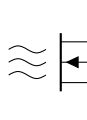


- pjfet (pjfet)



isfet

- isfet (isfet)



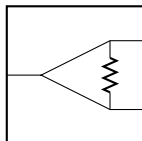
3.3.3 HF

These come from Stefan Erhardt's contribution of HF components.

- circulator (circulator)



- wilkinson divider (wilkinson)

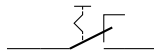


3.3.4 Switch

- spdt (spdt)



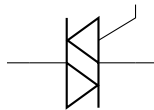
- Toggle switch (toggle switch)



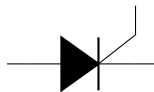
3.3.5 Other bipole-like tripoles

The following tripoles are entered with the usual command of the form

- triac (triac, or Tr)



- thyristor (thyristor, or Ty)



3.3.6 Misc

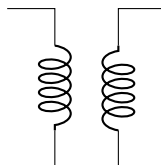
- Mixer (mixer)



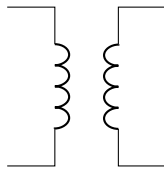
3.4 Double bipoles

Transformers automatically use the inductor shape currently selected. These are the three possibilities:

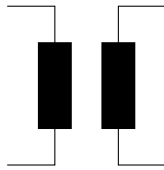
- Transformer (cute inductor) (transformer)



- Transformer (american inductor) (transformer)

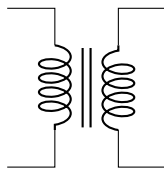


- Transformer (european inductor) (transformer)

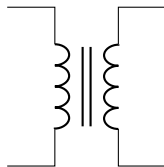


Transformers with core are also available:

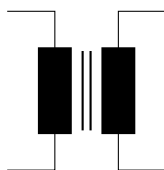
- Transformer core (cute inductor) (transformer core)



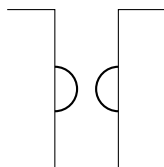
- Transformer core (american inductor) (transformer core)



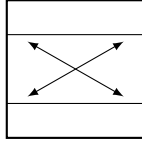
- Transformer core (european inductor) (transformer core)



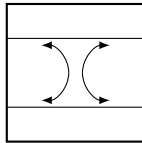
- Gyrator (gyrator)



- Coupler (coupler)

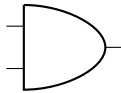


- Coupler, 2 (coupler2)

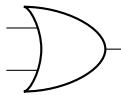


3.5 Logic gates

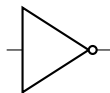
- American and port (american and port)



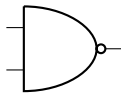
- American or port (american or port)



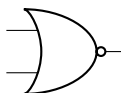
- American not port (american not port)



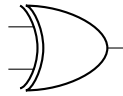
- American nand port (american nand port)



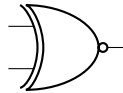
- American nor port (american nor port)



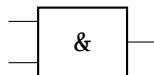
- American xor port (american xor port)



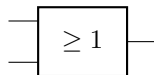
- American xnor port (american xnor port)



- European and port (european and port)



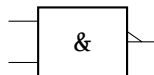
- European or port (european or port)



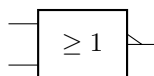
- European not port (european not port)



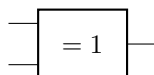
- European nand port (european nand port)



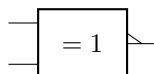
- European nor port (european nor port)



- European xor port (european xor port)



- European xnor port (european xnor port)

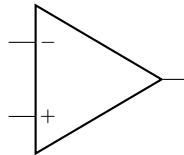


If (default behaviour) `americanports` option is active (or the style `[american ports]` is used), the shorthands `and port`, `or port`, `not port`, `nand port`, `not port`, `xor port`, and `xnor port` are equivalent to the american version of the respective logic port.

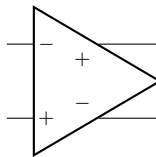
If otherwise `europoport` option is active (or the style `[european ports]` is used), the shorthands `and port`, `or port`, `not port`, `nand port`, `not port`, `xor port`, and `xnor port` are equivalent to the european version of the respective logic port.

3.6 Amplifiers

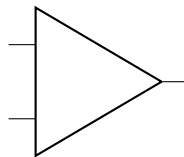
- Operational amplifier (`op amp`)



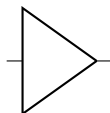
- Fully differential operational amplifier⁶ (`fd op amp`)



- Plain amplifier (`plain amp`)



- Buffer (`buffer`)



⁶Contributed by Kristofer M. Monisit.

3.7 Support shapes

- Arrows (current and voltage) (`currarrow`)



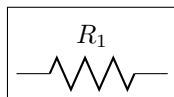
- Connected terminal (`circ`)



- Unconnected terminal (`ocirc`)



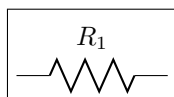
4 Usage



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, l=$R_1$] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R=$R_1$] (2,0);
3 \end{circuitikz}

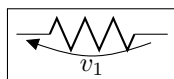
```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i=$i_1$] (2,0);
3 \end{circuitikz}

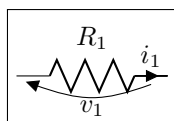
```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, v=$v_1$] (2,0);
3 \end{circuitikz}

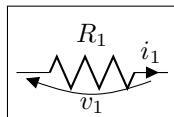
```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R=$R_1$, i=$i_1$, v=$v_1$] (2,0);
3 \end{circuitikz}

```

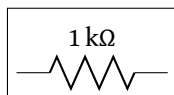


```

1 \begin{circuitikz}
2   \draw (0,0) to[R=$R_1$, i=$i_1$, v=$v_1$] (2,0);
3 \end{circuitikz}

```

Long names/styles for the bipoles can be used:

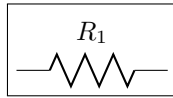


```

1 \begin{circuitikz} \draw
2   (0,0) to[resistor=1<\kilo\ohm>] (2,0)
3 ;\end{circuitikz}

```

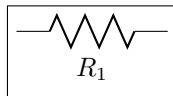
4.1 Labels



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, l^=$R_1$] (2,0);
3 \end{circuitikz}

```

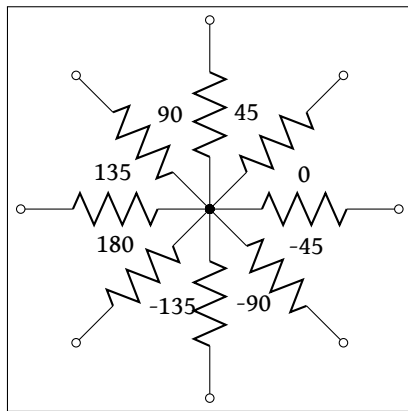


```

1 \begin{circuitikz}
2   \draw (0,0) to[R, l_=$R_1$] (2,0);
3 \end{circuitikz}

```

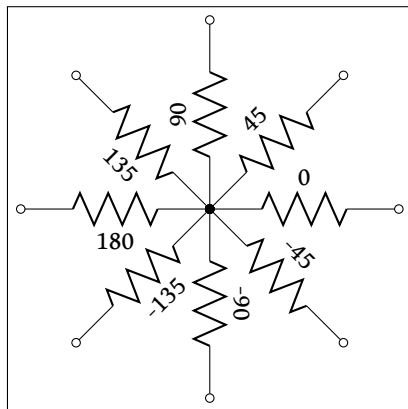
The default orientation of labels is controlled by the options `smartlabels`, `rotatelabels` and `straightlabels` (or the corresponding `label/align` keys). Here are examples to see the differences:



```

1 \begin{circuitikz}
2   \ctikzset{label/align = straight}
3   \def\DIR{0,45,90,135,180,-90,-45,-135}
4   \foreach \i in \DIR {
5     \draw (0,0) to[R=\i, *-o] (\i :2.5);
6   }
7 \end{circuitikz}

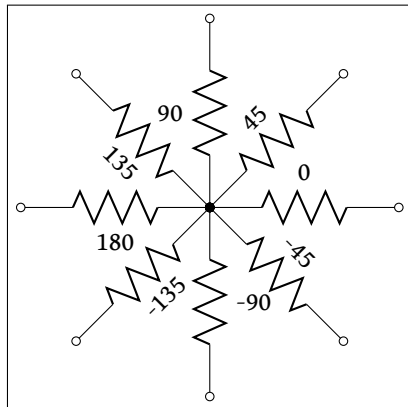
```



```

1 \begin{circuitikz}
2   \ctikzset{label/align = rotate}
3   \def\DIR{0,45,90,135,180,-90,-45,-135}
4   \foreach \i in \DIR {
5     \draw (0,0) to[R=\i, *-o] (\i :2.5);
6   }
7 \end{circuitikz}

```

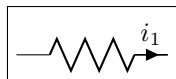


```

1 \begin{circuitikz}
2 \ctikzset{label/align = smart}
3 \def\DIR{0,45,90,135,180,-90,-45,-135}
4 \foreach \i in \DIR {
5   \draw (0,0) to[R=\i, *-o] (\i :2.5);
6 }
7 \end{circuitikz}

```

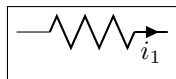
4.2 Currents



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i^>=$i_1$] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i_>=$i_1$] (2,0);
3 \end{circuitikz}

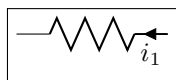
```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i^<=$i_1$] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i_<=$i_1$] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i^>=$i_1$] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i>_=$i_1$] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i<^=$i_1$] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i<_=$i_1$] (2,0);
3 \end{circuitikz}

```

Also



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i<=$i_1$] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i>=$i_1$] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i^=$i_1$] (2,0);
3 \end{circuitikz}

```



```

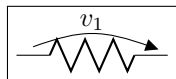
1 \begin{circuitikz}
2   \draw (0,0) to[R, i_=$i_1$] (2,0);
3 \end{circuitikz}

```

4.3 Voltages

4.3.1 European style

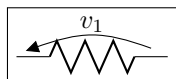
The default, with arrows. Use option `europeanvoltage` or style `[european voltages]`.



```

1 \begin{circuitikz}[european voltages]
2   \draw (0,0) to[R, v^>=$v_1$] (2,0);
3 \end{circuitikz}

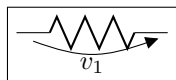
```



```

1 \begin{circuitikz}[european voltages]
2   \draw (0,0) to[R, v^<=$v_1$] (2,0);
3 \end{circuitikz}

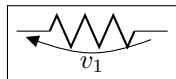
```



```

1 \begin{circuitikz}[european voltages]
2   \draw (0,0) to[R, v_>=$v_1$] (2,0);
3 \end{circuitikz}

```



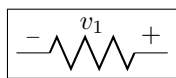
```

1 \begin{circuitikz}[european voltages]
2   \draw (0,0) to[R, v_<=$v_1$] (2,0);
3 \end{circuitikz}

```

4.3.2 American style

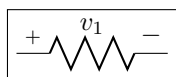
For those who like it (not me). Use option `americanvoltage` or set `[american voltages]`.



```

1 \begin{circuitikz}[american voltages]
2   \draw (0,0) to[R, v^>=$v_1$] (2,0);
3 \end{circuitikz}

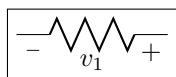
```



```

1 \begin{circuitikz}[american voltages]
2   \draw (0,0) to[R, v^<=$v_1$] (2,0);
3 \end{circuitikz}

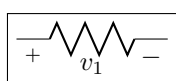
```



```

1 \begin{circuitikz}[american voltages]
2   \draw (0,0) to[R, v_>=$v_1$] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}[american voltages]
2   \draw (0,0) to[R, v_<=$v_1$] (2,0);
3 \end{circuitikz}

```

4.4 Nodes



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, o-o] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, -o] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, o-] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, *-] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, -*] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, *-] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, o-*] (2,0);
3 \end{circuitikz}

```



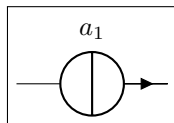
```

1 \begin{circuitikz}
2   \draw (0,0) to[R, *-o] (2,0);
3 \end{circuitikz}

```

4.5 Special components

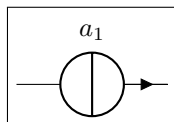
For some components label, current and voltage behave as one would expect:



```

1 \begin{circuitikz}
2   \draw (0,0) to[I=$a_1$] (2,0);
3 \end{circuitikz}

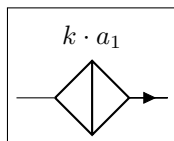
```



```

1 \begin{circuitikz}
2   \draw (0,0) to[I, i=$a_1$] (2,0);
3 \end{circuitikz}

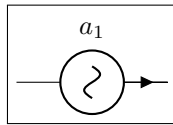
```



```

1 \begin{circuitikz}
2   \draw (0,0) to[cI=$k\cdot a_1$] (2,0);
3 \end{circuitikz}

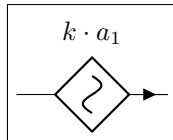
```

```

1 \begin{circuitikz}
2   \draw (0,0) to[sI=$a_1$] (2,0);
3 \end{circuitikz}

```

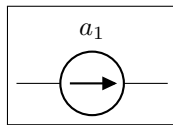


```

1 \begin{circuitikz}
2   \draw (0,0) to[csi=$k\cdot a_1$] (2,0);
3 \end{circuitikz}

```

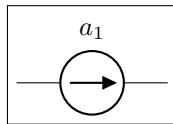
The following results from using the option `american current` or using the style `[american current]`.



```

1 \begin{circuitikz}[american current]
2   \draw (0,0) to[I=$a_1$] (2,0);
3 \end{circuitikz}

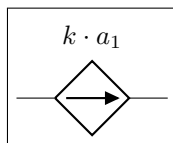
```



```

1 \begin{circuitikz}[american current]
2   \draw (0,0) to[I, i=$a_1$] (2,0);
3 \end{circuitikz}

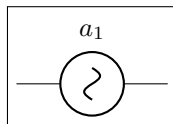
```



```

1 \begin{circuitikz}[american current]
2   \draw (0,0) to[ci=$k\cdot a_1$] (2,0);
3 \end{circuitikz}

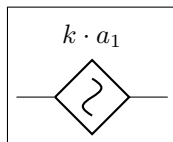
```



```

1 \begin{circuitikz}[american current]
2   \draw (0,0) to[sI=$a_1$] (2,0);
3 \end{circuitikz}

```

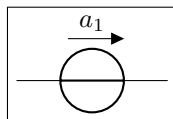


```

1 \begin{circuitikz}[american current]
2   \draw (0,0) to[csi=$k\cdot a_1$] (2,0);
3 \end{circuitikz}

```

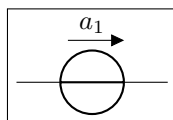
The same holds for voltage sources:



```

1 \begin{circuitikz}
2   \draw (0,0) to[V=$a_1$] (2,0);
3 \end{circuitikz}

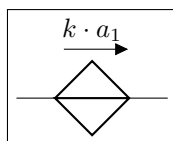
```



```

1 \begin{circuitikz}
2   \draw (0,0) to[V, v=$a_1$] (2,0);
3 \end{circuitikz}

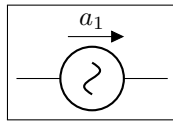
```



```

1 \begin{circuitikz}
2   \draw (0,0) to[cV=$k\cdot a_1$] (2,0);
3 \end{circuitikz}

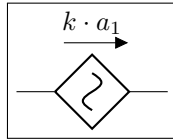
```



```

1 \begin{circuitikz}
2   \draw (0,0) to[sV=$a_1$] (2,0);
3 \end{circuitikz}

```

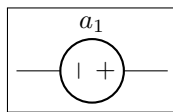


```

1 \begin{circuitikz}
2   \draw (0,0) to[csV=$k\cdot a_1$] (2,0);
3 \end{circuitikz}

```

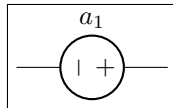
The following results from using the option `american voltages` or the style `[american voltages]`.



```

1 \begin{circuitikz}[american voltages]
2   \draw (0,0) to[V=$a_1$] (2,0);
3 \end{circuitikz}

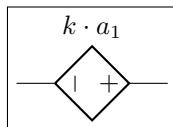
```



```

1 \begin{circuitikz}[american voltages]
2   \draw (0,0) to[V, v=$a_1$] (2,0);
3 \end{circuitikz}

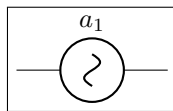
```



```

1 \begin{circuitikz}[american voltages]
2   \draw (0,0) to[cV=$k\cdot a_1$] (2,0);
3 \end{circuitikz}

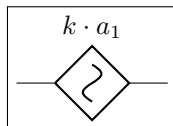
```



```

1 \begin{circuitikz}[american voltages]
2   \draw (0,0) to[sV=$a_1$] (2,0);
3 \end{circuitikz}

```



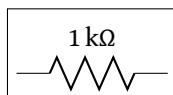
```

1 \begin{circuitikz}[american voltages]
2   \draw (0,0) to[csV=$k\cdot a_1$] (2,0);
3 \end{circuitikz}

```

4.6 Integration with `siunitx`

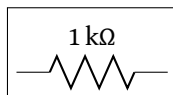
If the option `siunitx` is active (and *not* in `ConTeXt`), then the following are equivalent:



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, l=1<\kilo\ohm>] (2,0);
3 \end{circuitikz}

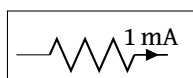
```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, l=$\SI{1}{\kilo\ohm}$] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i=1<\milli\ampere>] (2,0);
3 \end{circuitikz}

```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, i=\SI{1}{\milli\ampere}] (2,0);
3 \end{circuitikz}

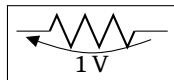
```



```

1 \begin{circuitikz}
2   \draw (0,0) to[R, v=1<\volt>] (2,0);
3 \end{circuitikz}

```

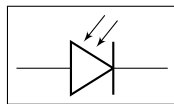


```

1 \begin{circuitikz}
2   \draw (0,0) to[R, v=\SI{1}{\volt}] (2,0);
3 \end{circuitikz}

```

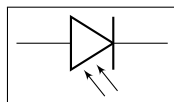
4.7 Mirroring



```

1 \begin{circuitikz}
2   \draw (0,0) to[pD] (2,0);
3 \end{circuitikz}

```

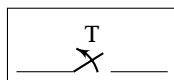


```

1 \begin{circuitikz}
2   \draw (0,0) to[pD, mirror] (2,0);
3 \end{circuitikz}

```

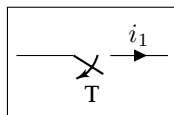
At the moment, placing labels and currents on mirrored bipoles works:



```

1 \begin{circuitikz}
2   \draw (0,0) to[ospst=T] (2,0);
3 \end{circuitikz}

```

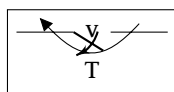


```

1 \begin{circuitikz}
2   \draw (0,0) to[ospst=T, mirror, i=i_1] (2,0);
3 \end{circuitikz}

```

But voltages don't:



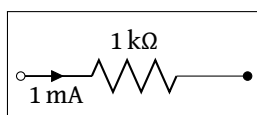
```

1 \begin{circuitikz}
2   \draw (0,0) to[ospst=T, mirror, v=v] (2,0);
3 \end{circuitikz}

```

Sorry about that.

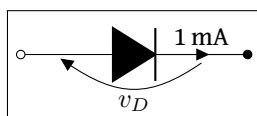
4.8 Putting them together



```

1 \begin{circuitikz}
2   \draw (0,0) to[R=1<\kilo\ohm>,
3     i>=1<\milli\ampere>, o-] (3,0);
4 \end{circuitikz}

```



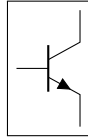
```

1 \begin{circuitikz}
2   \draw (0,0) to[D*, v=v_D,
3     i=1<\milli\ampere>, o-] (3,0);
4 \end{circuitikz}

```

5 Not only bipoles

Since only bipoles (but see section 5.3) can be placed "along a line", components with more than two terminals are placed as nodes:



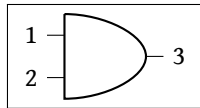
```
1\tikz \node[npn] at (0,0) {};
```

5.1 Anchors

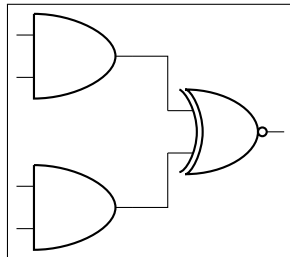
In order to allow connections with other components, all components define anchors.

5.1.1 Logical ports

All logical ports, except not, have two inputs and one output. They are called respectively in 1, in 2, out:

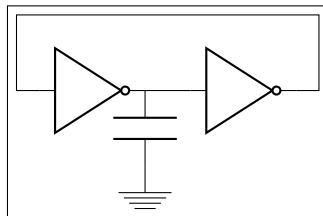


```
1\begin{circuitikz} \draw
2 (0,0) node[and port] (myand) {}
3 (myand.in 1) node[anchor=east] {1}
4 (myand.in 2) node[anchor=east] {2}
5 (myand.out) node[anchor=west] {3}
6;\end{circuitikz}
```



```
1\begin{circuitikz} \draw
2 (0,2) node[and port] (myand1) {}
3 (0,0) node[and port] (myand2) {}
4 (2,1) node[xnor port] (myxnor) {}
5 (myand1.out) -| (myxnor.in 1)
6 (myand2.out) -| (myxnor.in 2)
7;\end{circuitikz}
```

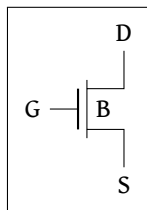
In the case of not, there are only in and out (although for compatibility reasons in 1 is still defined and equal to in):



```
1\begin{circuitikz} \draw
2 (1,0) node[not port] (not1) {}
3 (3,0) node[not port] (not2) {}
4 (0,0) -- (not1.in)
5 (not2.in) -- (not1.out)
6 ++(0,-1) node[ground] {} to[C] (not1.out)
7 (not2.out) -| (4,1) -| (0,0)
8;\end{circuitikz}
```

5.1.2 Transistors

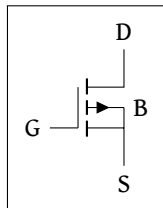
For nmos, pmos, nfet, nignfet, nignfetd, pfet, pigfete, and pigfetd transistors one has base, gate, source and drain anchors (which can be abbreviated with B, G, S and D):



```

1 \begin{circuitikz} \draw
2   (0,0) node[nmos] (mos) {}
3   (mos.base) node[anchor=west] {B}
4   (mos.gate) node[anchor=east] {G}
5   (mos.drain) node[anchor=south] {D}
6   (mos.source) node[anchor=north] {S}
7 ;\end{circuitikz}

```

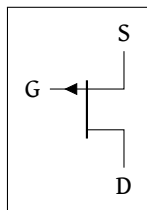


```

1 \begin{circuitikz} \draw
2   (0,0) node[pigfete] (pigfete) {}
3   (pigfete.B) node[anchor=west] {B}
4   (pigfete.G) node[anchor=east] {G}
5   (pigfete.D) node[anchor=south] {D}
6   (pigfete.S) node[anchor=north] {S}
7 ;\end{circuitikz}

```

Similarly njfet and pjfet have gate, source and drain anchors (which can be abbreviated with G, S and D):

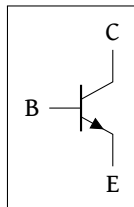


```

1 \begin{circuitikz} \draw
2   (0,0) node[pjfet] (pjfet) {}
3   (pjfet.G) node[anchor=east] {G}
4   (pjfet.D) node[anchor=north] {D}
5   (pjfet.S) node[anchor=south] {S}
6 ;\end{circuitikz}

```

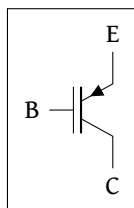
For npn, pnp, night, and pigbt transistors the anchors are base, emitter and collector anchors (which can be abbreviated with B, E and C):



```

1 \begin{circuitikz} \draw
2   (0,0) node[npn] (nnp) {}
3   (nnp.base) node[anchor=east] {B}
4   (nnp.collector) node[anchor=south] {C}
5   (nnp.emitter) node[anchor=north] {E}
6 ;\end{circuitikz}

```

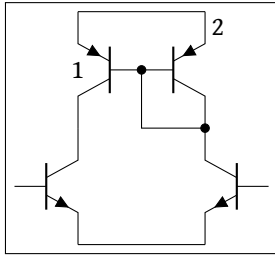


```

1 \begin{circuitikz} \draw
2   (0,0) node[pigbt] (pigbt) {}
3   (pigbt.B) node[anchor=east] {B}
4   (pigbt.C) node[anchor=north] {C}
5   (pigbt.E) node[anchor=south] {E}
6 ;\end{circuitikz}

```

Here is one composite example (please notice that the `xscale=-1` style would also reflect the label of the transistors, so here a new node is added and its text is used, instead of that of `pnp1`):

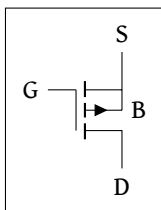


```

1 \begin{circuitikz} \draw
2   (0,0) node[pnp] (pnp2) {}
3   (pnp2.B) node[pnp, xscale=-1, anchor=B] (pnp1) {}
4   (pnp1) node {}
5   (pnp1.C) node[npn, anchor=C] (npn1) {}
6   (pnp2.C) node[npn, xscale=-1, anchor=C] (npn2) {}
7   (pnp1.E) -- (pnp2.E) (npn1.E) -- (npn2.E)
8   (pnp1.B) node[circ] {} |- (pnp2.C) node[circ] {}
9 ;\end{circuitikz}

```

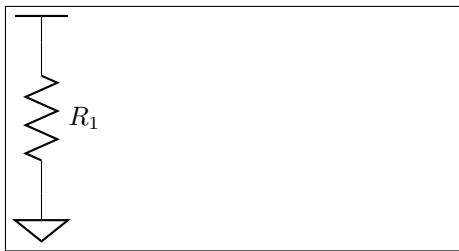
Similarly, transistors and other components can be reflected vertically:



```

1 \begin{circuitikz} \draw
2   (0,0) node[pigfete, yscale=-1] (pigfete) {}
3   (pigfete.B) node[anchor=west] {B}
4   (pigfete.G) node[anchor=east] {G}
5   (pigfete.D) node[anchor=north] {D}
6   (pigfete.S) node[anchor=south] {S}
7 ;\end{circuitikz}

```



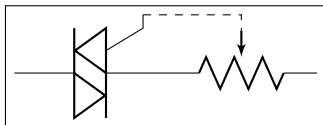
```

1 \begin{circuitikz}
2   \draw (0,2)
3     node[rground, yscale=-1] {}
4     to[R=$R_1$] (0,0)
5     node[sground] {};
6 \end{circuitikz}

```

5.1.3 Other tripoles

When inserting a thyristor, a triac or a potentiometer, one needs to refer to the third node—gate (gate or G) for the former two; wiper (wiper or W) for the latter one. This is done by giving a name to the bipole:

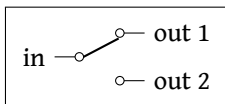


```

1 \begin{circuitikz} \draw
2   (0,0) to[Tr, n=TRI] (2,0)
3     to[pR, n=POT] (4,0);
4   \draw[dashed] (TRI.G) -| (POT.wiper)
5 ;\end{circuitikz}

```

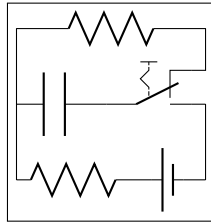
As for the switches:



```

1 \begin{circuitikz} \draw
2   (0,0) node[spdt] (Sw) {}
3   (Sw.in) node[left] {in}
4   (Sw.out 1) node[right] {out 1}
5   (Sw.out 2) node[right] {out 2}
6 ;\end{circuitikz}

```

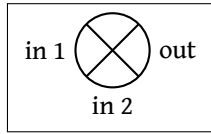


```

1 \begin{circuitikz} \draw
2   (0,0) to[C] (1,0) to[toggle switch , n=Sw] (2.5,0)
3   -- (2.5,-1) to[battery1] (1.5,-1) to[R] (0,-1) -- (0,0)
4   (Sw.out 2) -- (2.5, 1) to[R] (0,1) -- (0,0)
5 ;\end{circuitikz}

```

And the mixer:



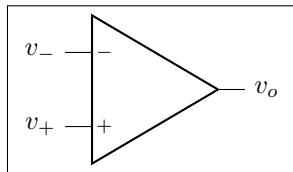
```

1 \begin{circuitikz} \draw
2   (0,0) node[mixer] (mix) {}
3   (mix.in 1) node[left] {in 1}
4   (mix.in 2) node[below] {in 2}
5   (mix.out) node[right] {out}
6 ;\end{circuitikz}

```

5.1.4 Operational amplifier

The op amp defines the inverting input ($-$), the non-inverting input ($+$) and the output (out) anchors:

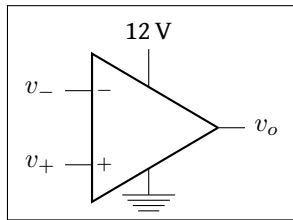


```

1 \begin{circuitikz} \draw
2   (0,0) node[op amp] (opamp) {}
3   (opamp.+) node[left] {$v_+$}
4   (opamp.-) node[left] {$v_-$}
5   (opamp.out) node[right] {$v_o$}
6 ;\end{circuitikz}

```

There are also two more anchors defined, up and down, for the power supplies:

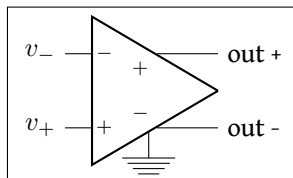


```

1 \begin{circuitikz} \draw
2   (0,0) node[op amp] (opamp) {}
3   (opamp.+) node[left] {$v_+$}
4   (opamp.-) node[left] {$v_-$}
5   (opamp.out) node[right] {$v_o$}
6   (opamp.down) node[ground] {}
7   (opamp.up) ++ (0,.5) node[above] {\SI{12}{\ volt}}
8   -- (opamp.up)
9 ;\end{circuitikz}

```

The fully differential op amp defines two outputs:

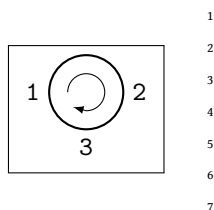


```

1 \begin{circuitikz} \draw
2   (0,0) node[fd op amp] (opamp) {}
3   (opamp.+) node[left] {$v_+$}
4   (opamp.-) node[left] {$v_-$}
5   (opamp.out +) node[right] {out +}
6   (opamp.out -) node[right] {out -}
7   (opamp.down) node[ground] {}
8 ;\end{circuitikz}

```

The circulator defined ports 1, 2 and 3:

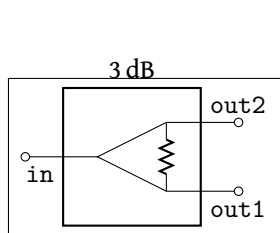


```

1 \begin{circuitikz} \draw
2   (0,0) node[circulator] (c) {}
3   (c.1) node[left] {\texttt{1}}
4   (c.2) node[right] {\texttt{2}}
5   (c.3) node[below] {\texttt{3}}
6   ;
7 \end{circuitikz}

```

The Wilkinson divider has:



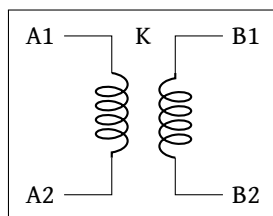
```

1 \begin{circuitikz} \draw
2   (0,0) node[wilkinson] (w) {\SI{3}{dB}}
3   (w.in) to[short,-o] ++(-0.5,0)
4   (w.out1) to[short,-o] ++(0.5,0)
5   (w.out2) to[short,-o] ++(0.5,0)
6   (w.in) node[below left] {\texttt{in}}
7   (w.out1) node[below right] {\texttt{out1}}
8   (w.out2) node[above right] {\texttt{out2}}
9   ;
10 \end{circuitikz}

```

5.1.5 Double bipoles

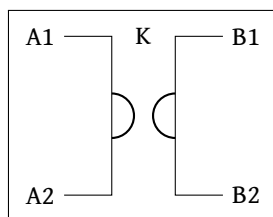
All the (few, actually) double bipoles/quadrupoles have the four anchors, two for each port. The first port, to the left, is port A, having the anchors A1 (up) and A2 (down); same for port B. They also expose the base anchor, for labelling:



```

1 \begin{circuitikz} \draw
2   (0,0) node[transformer] (T) {}
3   (T.A1) node[anchor=east] {A1}
4   (T.A2) node[anchor=east] {A2}
5   (T.B1) node[anchor=west] {B1}
6   (T.B2) node[anchor=west] {B2}
7   (T.base) node{K}
8 ;\end{circuitikz}

```

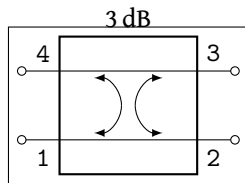
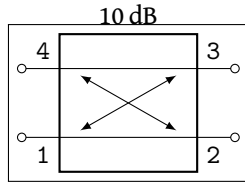


```

1 \begin{circuitikz} \draw
2   (0,0) node[gyrator] (G) {}
3   (G.A1) node[anchor=east] {A1}
4   (G.A2) node[anchor=east] {A2}
5   (G.B1) node[anchor=west] {B1}
6   (G.B2) node[anchor=west] {B2}
7   (G.base) node{K}
8 ;\end{circuitikz}

```

However:



```

1      \begin{circuitikz} \draw
2          (0,0) node[coupler] (c) {\SI{10}{dB}}
3          (c.1) to[short,-o] ++(-0.5,0)
4          (c.2) to[short,-o] ++(0.5,0)
5          (c.3) to[short,-o] ++(0.5,0)
6          (c.4) to[short,-o] ++(-0.5,0)
7          (c.1) node[below left] {\texttt{1}}
8          (c.2) node[below right] {\texttt{2}}
9          (c.3) node[above right] {\texttt{3}}
10         (c.4) node[above left] {\texttt{4}}
11         ;
12     \end{circuitikz}

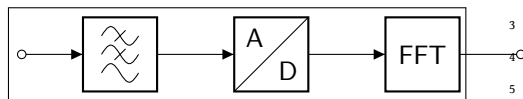
1      \begin{circuitikz} \draw
2          (0,0) node[coupler2] (c) {\SI{3}{dB}}
3          (c.1) to[short,-o] ++(-0.5,0)
4          (c.2) to[short,-o] ++(0.5,0)
5          (c.3) to[short,-o] ++(0.5,0)
6          (c.4) to[short,-o] ++(-0.5,0)
7          (c.1) node[below left] {\texttt{1}}
8          (c.2) node[below right] {\texttt{2}}
9          (c.3) node[above right] {\texttt{3}}
10         (c.4) node[above left] {\texttt{4}}
11         ;
12     \end{circuitikz}

```

5.2 Input arrows

two ports

The new two ports all have a defined direction. If one needs opposite direction, just place them from right to left. With the option > a input arrow is drawn:



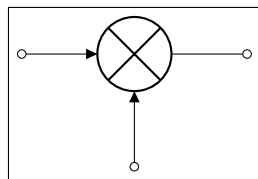
```

1      \begin{circuitikz} \draw
2          (0,0) to[short,o-] ++(0.3,0)
3          to[lowpass,>] ++(2,0)
4          to[adc,>] ++(2,0)
5          to[fft,>] ++(2,0)
6          to[short,-o] ++(0.3,0);
7      \end{circuitikz}

```

multi ports

Since inputs and outputs can vary, input arrows can be placed as nodes:



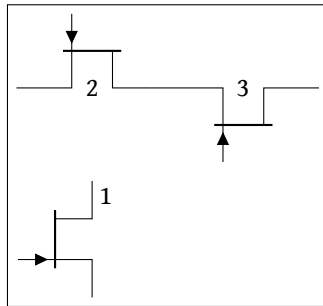
```

1      \begin{circuitikz} \draw
2          (0,0) node[mixer] (m) {}
3          (m.in1) to[short,-o] ++(-1,0)
4          (m.in2) to[short,-o] ++(0,-1)
5          (m.out) to[short,-o] ++(1,0)
6          (m.in1) node[inputarrow] {}
7          (m.in2) node[inputarrow,rotate=90] {};
8      \end{circuitikz}

```

5.3 Transistor paths

For syntactical convenience transistors can be placed using the normal path notation used for bipoles. The transistor type can be specified by simply adding a “T” (for transistor) in front of the node name of the transistor. It will be placed with the base/gate orthogonal to the direction of the path:

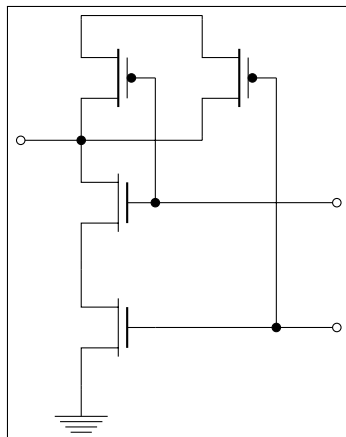


```

1 \begin{circuitikz} \draw
2   (0,0) node[njfet] {1}
3   (-1,2) to[Tnjfet=2] (1,2)
4     to[Tnjfet=3, mirror] (3,2);
5 \end{circuitikz}

```

Access to the gate and/or base nodes can be gained by naming the transistors with the n or name path style:



```

1 \begin{circuitikz} \draw[yscale=1.1, xscale=.8]
2   (2,4.5) -- (0,4.5) to[Tpmos, n=p1] (0,3)
3     to[Tnmos, n=n1] (0,1.5)
4     to[Tnmos, n=n2] (0,0) node[ground] {}
5   (2,4.5) to[Tpmos, n=p2] (2,3) to[short, *-] (0,3)
6   (p1.G) -- (n1.G) to[short, *-o] ($(n1.G)+(3,0)$)
7   (n2.G) ++(2,0) node[circ] {} -- (p2.G)
8   (n2.G) to[short, -o] ($(n2.G)+(3,0)$)
9   (0,3) to[short, -o] (-1,3)
10 \end{circuitikz}

```

The name property is available also for bipoles, although this is useful mostly for triac, potentiometer and thyristor (see 3.3.5).

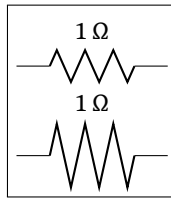
6 Customization

6.1 Parameters

Pretty much all CircuiTikZ relies heavily on pgfkeys for value handling and configuration. Indeed, at the beginning of `circuitikz.sty` a series of key definitions can be found that modify all the graphical characteristics of the package.

All can be varied using the `\ctikzset` command, anywhere in the code.

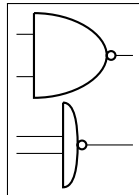
Shape of the components (on a per-component-class basis)



```

1 \tikz \draw (0,0) to[R=1<\ohm>] (2,0); \par
2 \ctikzset {bipoles/resistor/height=.6}
3 \tikz \draw (0,0) to[R=1<\ohm>] (2,0);

```

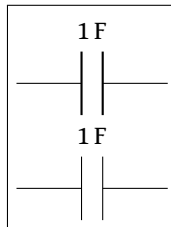


```

1 \tikz \draw (0,0) node[nand port] {}; \par
2 \ctikzset {tripoles/american nand port/input height=.2}
3 \ctikzset {tripoles/american nand port/port width=.2}
4 \tikz \draw (0,0) node[nand port] {};

```

Thickness of the lines (globally)

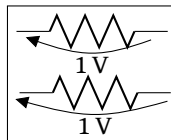


```

1 \tikz \draw (0,0) to[C=1<\farad>] (2,0); \par
2 \ctikzset {bipoles/thickness=1}
3 \tikz \draw (0,0) to[C=1<\farad>] (2,0);

```

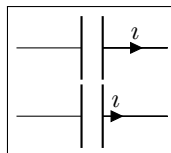
Global properties Of voltage and current



```

1 \tikz \draw (0,0) to[R, v=1<\volt>] (2,0); \par
2 \ctikzset {voltage/distance from node=.1}
3 \tikz \draw (0,0) to[R, v=1<\volt>] (2,0);

```

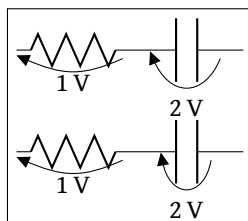


```

1 \tikz \draw (0,0) to[C, i=$\imath$] (2,0); \par
2 \ctikzset {current/distance = .2}
3 \tikz \draw (0,0) to[C, i=$\imath$] (2,0);

```

However, you can override the properties `voltage/distance from node`⁷, `voltage/bump b`⁸ and `voltage/european label distance`⁹ on a per-component basis, in order to fine-tune the voltages:



```

1 \tikz \draw (0,0) to[R, v=1<\volt>] (1.5,0)
2 to[C, v=2<\volt>] (3,0); \par
3 \ctikzset {bipoles/capacitor/voltage/%
4 distance from node/.initial =.7}
5 \tikz \draw (0,0) to[R, v=1<\volt>] (1.5,0)
6 to[C, v=2<\volt>] (3,0); \par

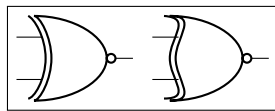
```

⁷That is, how distant from the initial and final points of the path the arrow starts and ends.

⁸Controlling how high the bump of the arrow is — how curved it is.

⁹Controlling how distant from the bipole the voltage label will be.

Admittedly, not all graphical properties have understandable names, but for the time it will have to do:



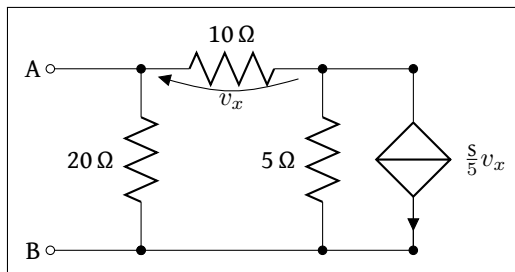
```

1 \tikz \draw (0,0) node[xnor port] {};
2 \ctikzset { tripoles /american xnor port/aaa=.2}
3 \ctikzset { tripoles /american xnor port/bbb=.6}
4 \tikz \draw (0,0) node[xnor port] {};

```

6.2 Components size

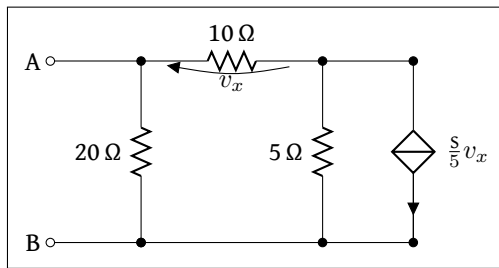
Perhaps the most important parameter is `\circuitikzbasekey/bipoles/length`, which can be interpreted as the length of a resistor (including reasonable connections): all other lengths are relative to this value. For instance:



```

1 \ctikzset {bipoles/length=1.4cm}
2 \begin{circuitikz} [scale=1.2] \draw
3   (0,0) node[anchor=east] {B}
4     to[short, o-*] (1,0)
5     to[R=20<\ohm>, *-] (1,2)
6     to[R=10<\ohm>, v=v_x$] (3,2) -- (4,2)
7     to[ci=$\frac{5}{5} v_x$, *-] (4,0) -- (3,0)
8     to[R=5<\ohm>, *-] (3,2)
9   (3,0) -- (1,0)
10  (1,2) to[short, -o] (0,2) node[anchor=east]{A}
11 \end{circuitikz}

```



```

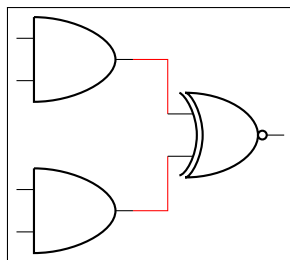
1 \ctikzset{bipoles/length=.8cm}
2 \begin{circuitikz}[scale=1.2]\draw
3   (0,0) node[anchor=east]{B}
4     to[short, o-*] (1,0)
5     to[R=20<\ohm>, *-] (1,2)
6     to[R=10<\ohm>, v=$v_x$] (3,2) -- (4,2)
7     to[ci=$\frac{5}{5} v_x$, *-] (4,0) -- (3,0)
8     to[R=5<\ohm>, *-] (3,2)
9   (3,0) -- (1,0)
10  (1,2) to[short, -o] (0,2) node[anchor=east]{A}
11 \end{circuitikz}

```

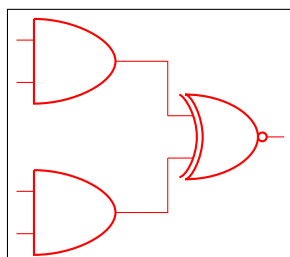
6.3 Colors

The color of the components is stored in the key `\circuitikzbasekey/color`. CircuiTikZ tries to follow the color set in TikZ, although sometimes it fails. If you change color in the picture, please do not use just the color name as a style, like `[red]`, but rather assign the style `[color=red]`.

Compare for instance



and



```

1 \begin{circuitikz} \draw[red]
2   (0,2) node[and port] (myand1) {}
3   (0,0) node[and port] (myand2) {}
4   (2,1) node[xnor port] (myxnor) {}
5   (myand1.out) -- (myxnor.in 1)
6   (myand2.out) -- (myxnor.in 2)
7 \end{circuitikz}

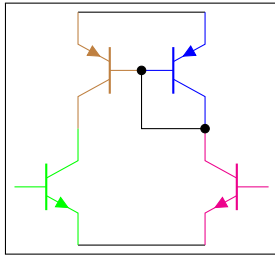
```

```

1 \begin{circuitikz} \draw[color=red]
2   (0,2) node[and port] (myand1) {}
3   (0,0) node[and port] (myand2) {}
4   (2,1) node[xnor port] (myxnor) {}
5   (myand1.out) -- (myxnor.in 1)
6   (myand2.out) -- (myxnor.in 2)
7 \end{circuitikz}

```

One can of course change the color *in medias res*:

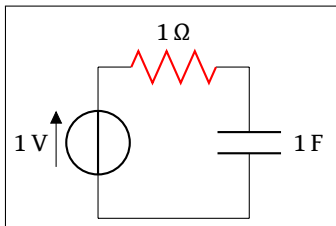


```

1 \begin{circuitikz} \draw
2   (0,0) node[pnp, color=blue] (pnp2) {}
3   (pnp2.B) node[pnp, xscale=-1, anchor=B, color=brown] (pnp1) {}
4   (pnp1.C) node[npn, anchor=C, color=green] (nnp1) {}
5   (pnp2.C) node[npn, xscale=-1, anchor=C, color=magenta] (nnp2) {}
6   (pnp1.E) -- (pnp2.E) (nnp1.E) -- (nnp2.E)
7   (pnp1.B) node[circ] {} |- (pnp2.C) node[circ] {}
8 ;\end{circuitikz}

```

The all-in-one stream of bipoles poses some challenges, as only the actual body of the bipole, and not the connecting lines, will be rendered in the specified color. Also, please notice the curly braces around the to:

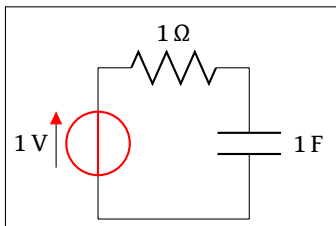


```

1 \begin{circuitikz} \draw
2   (0,0) to[V=1<\volt>] (0,2)
3   { to[R=1<\ohm>, color=red] (2,2) }
4   to[C=1<\farad>] (2,0) -- (0,0)
5 ;\end{circuitikz}

```

Which, for some bipoles, can be frustrating:

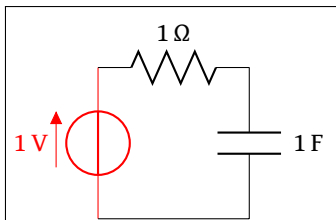


```

1 \begin{circuitikz} \draw
2   (0,0){ to[V=1<\volt>, color=red] (0,2) }
3   to[R=1<\ohm>] (2,2)
4   to[C=1<\farad>] (2,0) -- (0,0)
5 ;\end{circuitikz}

```

The only way out is to specify different paths:



```

1 \begin{circuitikz} \draw[color=red]
2   (0,0) to[V=1<\volt>, color=red] (0,2);
3   \draw (0,2) to[R=1<\ohm>] (2,2)
4   to[C=1<\farad>] (2,0) -- (0,0)
5 ;\end{circuitikz}

```

And yes: this is a bug and *not* a feature...

7 FAQ

Q: When using `\tikzexternalize` I get the following error:

! Emergency stop.

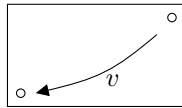
A: The TikZ manual states:

Furthermore, the library assumes that all \TeX pictures are ended with `\end{tikzpicture}`.

Just substitute every occurrence of the environment `circuitikz` with `tikzpicture`. They are actually pretty much the same.

Q: How do I draw the voltage between two nodes?

A: Between any two nodes there is an open circuit!



```

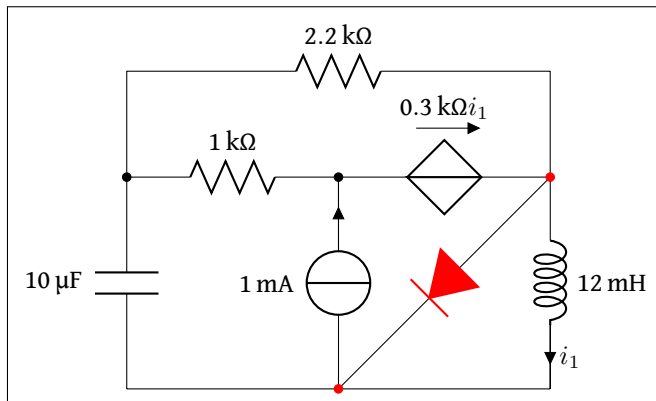
1 \begin{circuitikz} \draw
2   node[ocirc] (A) at (0,0) {}
3   node[ocirc] (B) at (2,1) {}
4   (A) to[open, v=$v$] (B)
5 \end{circuitikz}

```

Q: I cannot write `to[R = $R_1=12V$]` nor `to[ospst = open, 3s]`: I get errors.

A: It is a limitation of the TikZ parser. Use `to[R = $R_1{=}12V$]` and `to[ospst = open{,} 3s]` instead.

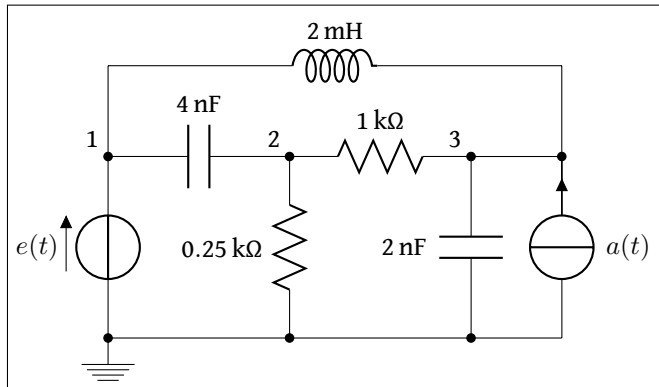
8 Examples



```

1 \begin{circuitikz}[ scale =1.4]\draw
2   (0,0) to[C, l=10<\micro\farad>] (0,2) -- (0,3)
3   to[R, l=2.2<\kilo\ohm>] (4,3) -- (4,2)
4   to[L, l=12<\milli\henry>, i=$i_1$] (4,0) -- (0,0)
5   (4,2) { to[D*, *-*, color=red] (2,0) }
6   (0,2) to[R, l=1<\kilo\ohm>, *-] (2,2)
7   to[cV, v=$\SI{.3}{\kilo\ohm} i_1$] (4,2)
8   (2,0) to[I, i=1<\milli\ampere>, *-] (2,2)
9 \end{circuitikz}

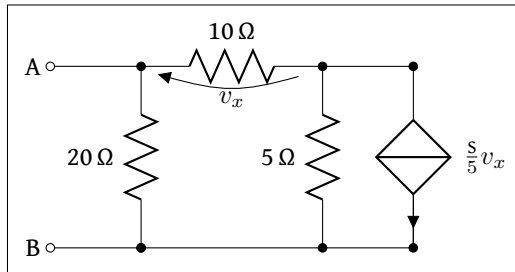
```



```

1 \begin{circuitikz} [ scale =1.2] \draw
2   (0,0) node[ground] {}
3     to[V=$e(t)$, *-] (0,2) to[C=4<\nano\farad>] (2,2)
4     to[R, l_=.25<\kilo\ohm>, *-] (2,0)
5     (2,2) to[R=1<\kilo\ohm>] (4,2)
6     to[C, l_ =2<\nano\farad>, *-] (4,0)
7     (5,0) to[I, i_=$a(t)$, -] (5,2) -- (4,2)
8     (0,0) -- (5,0)
9     (0,2) -- (0,3) to[L, l=2<\milli\henry>] (5,3) -- (5,2)
10
11   {[anchor=south east] (0,2) node {1} (2,2) node {2} (4,2) node {3}}
12 ;\end{circuitikz}

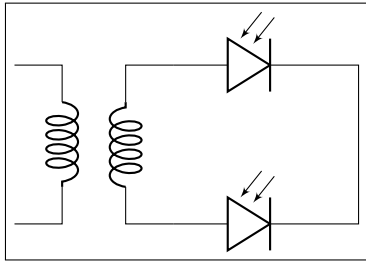
```



```

1 \begin{circuitikz} [ scale =1.2] \draw
2   (0,0) node[anchor=east] {B}
3     to[short, o-] (1,0)
4     to[R=20<\ohm>, *-] (1,2)
5     to[R=10<\ohm>, v=$v_x$] (3,2) -- (4,2)
6     to[ci=$\frac{\siemens}{5} v_x$, *-] (4,0) -- (3,0)
7     to[R=5<\ohm>, *-] (3,2)
8     (3,0) -- (1,0)
9     (1,2) to[short, -o] (0,2) node[anchor=east]{A}
10 ;\end{circuitikz}

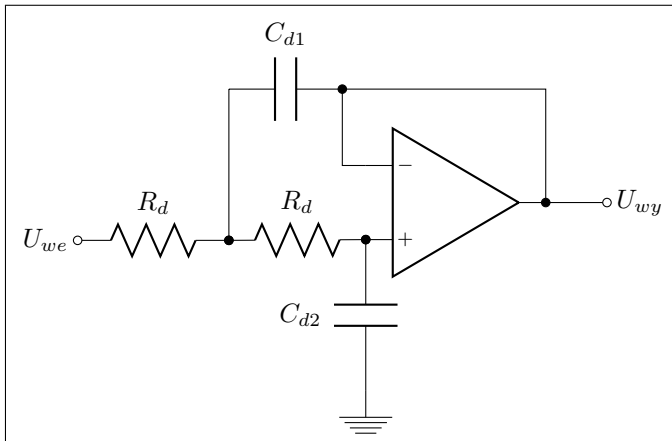
```

```

1 \begin{circuitikz}[scale=1]\draw
2     (0,0) node[transformer] (T) {}
3     (T.B2) to[pD] ($(T.B 2)+(2,0)$) -| (3.5, -1)
4     (T.B1) to[pD] ($(T.B 1)+(2,0)$) -| (3.5, -1)
5 \end{circuitikz}

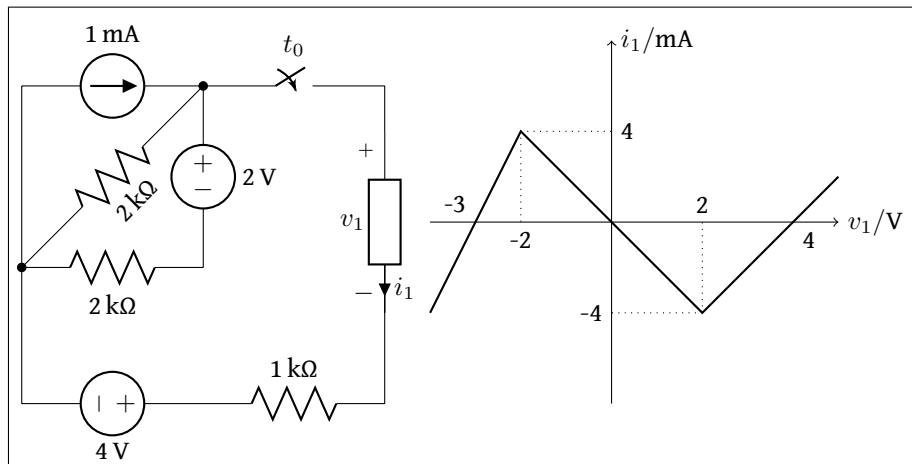
```



```

1 \begin{circuitikz}[scale=1]\draw
2     (5,5) node[op amp] (opamp) {}
3     (0,0) node[left] {$U_{we}$} to [R, l=$R_d$, o-] (2,0)
4     to [R, l=$R_d$, *-] (opamp.-)
5     to [C, l=$C_{d2}$, *-] ($(opamp.+) + (0,-2)$) node[ground] {}
6     (opamp.out) -| (3.5,2) to [C, l=$C_{d1}$, *-] (2,2) to [short] (2,0)
7     (opamp.-) -| (3.5,2)
8     (opamp.out) to [short, *-o] (7,5) node[right] {$U_{wy}$}
9 \end{circuitikz}

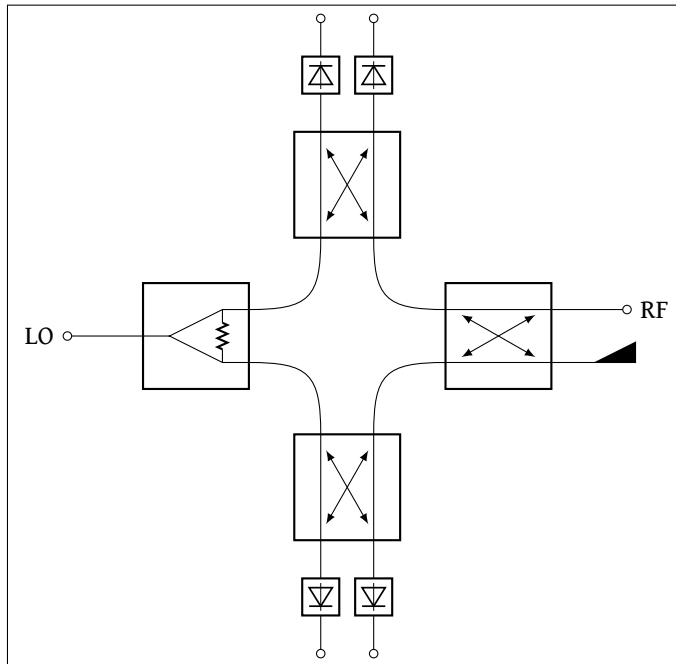
```



```

1 \begin{circuitikz}[ scale =1.2, american]\draw
2   (0,2) to[I=1<\milli\ampere>] (2,2)
3     to[R, l_=2<\kilo\ohm>, *-] (0,0)
4     to[R, l_=2<\kilo\ohm>] (2,0)
5     to[V, v_=2<\volt>] (2,2)
6     to[cspst, l=$t_0$] (4,2) -- (4,1.5)
7     to [generic, i=$i_1$, v=$v_1$] (4,-.5) -- (4,-1.5)
8   (0,2) -- (0,-1.5) to[V, v_=4<\volt>] (2,-1.5)
9     to [R, l=1<\kilo\ohm>] (4,-1.5);
10
11 \begin{scope}[xshift =6.5cm, yshift =.5cm]
12   \draw [->] (-2,0) -- (2.5,0) node[anchor=west] {$v_1/\text{V}$};
13   \draw [->] (0,-2) -- (0,2) node[anchor=west] {$i_1/\text{mA}$};
14   \draw (-1,0) node[anchor=north] {-2} (1,0) node[anchor=south] {2}
15         (0,1) node[anchor=west] {4} (0,-1) node[anchor=east] {-4}
16         (2,0) node[anchor=north west] {4}
17         (-1.5,0) node[anchor=south east] {-3};
18   \draw [thick] (-2,-1) -- (-1,1) -- (1,-1) -- (2,0) -- (2.5,.5);
19   \draw [dotted] (-1,1) -- (-1,0) (1,-1) -- (1,0)
20             (-1,1) -- (0,1) (1,-1) -- (0,-1);
21 \end{scope}
22 \end{circuitikz}

```



```

1      \begin{circuitikz} [ scale=1]
2          \ctikzset {bipoles/detector/width=.35}
3          \ctikzset {quadpoles/coupler/width=1}
4          \ctikzset {quadpoles/coupler/height=1}
5          \ctikzset {tripoles/wilkinson/width=1}
6          \ctikzset {tripoles/wilkinson/height=1}
7          %\draw[help lines , red , thin , dotted] (0,-5) grid (5,5);
8          \draw
9              (-2,0) node[wilkinson](w1){}
10             (2,0) node[coupler] (c1) {}
11             (0,2) node[coupler,rotate=90] (c2) {}
12             (0,-2) node[coupler,rotate=90] (c3) {}
13             (w1.out1) .. controls ++(0.8,0) and ++(0,0.8) .. (c3.3)
14             (w1.out2) .. controls ++(0.8,0) and ++(0,-0.8) .. (c2.4)
15             (c1.1) .. controls ++(-0.8,0) and ++(0,0.8) .. (c3.2)
16             (c1.4) .. controls ++(-0.8,0) and ++(0,-0.8) .. (c2.1)
17             (w1.in) to[short,-o] ++(-1,0)
18             (w1.in) node[left=30] {LO}
19             (c1.2) node[match,yscale=1] {}
20             (c1.3) to[short,-o] ++(1,0)
21             (c1.3) node[right=30] {RF}
22             (c2.3) to[detector,-o] ++(0,1.5)
23             (c2.2) to[detector,-o] ++(0,1.5)
24             (c3.1) to[detector,-o] ++(0,-1.5)
25             (c3.4) to[detector,-o] ++(0,-1.5)
26             ;
27      \end{circuitikz}

```

9 Revision history

version 0.3.1 (?)

1. typo in documentation
2. fixed an error in the angle computation in voltage and current routines
3. fixed problem with label size when scaling a tikz picture
4. added gas filled surge arrester
5. added compatibility option to work with Tikz's own circuit library

version 0.3.0 (20121229)

1. fixed gate node for a few transistors
2. added mixer
3. added fully differential op amp (by Kristofer M. Monisit)
4. now general settings for the drawing of voltage can be overridden for specific components
5. made arrows more homogeneous (either the current/voltage one, or latex' by pgf)
6. added the single battery cell
7. added fuse and asymmetric fuse
8. added toggle switch
9. added varistor, photoresistor, thermocouple, push button
10. added thermistor, thermistor ptc, thermistor ptc
11. fixed misalignment of voltage label in vertical bipoles with names
12. added isfet
13. added noiseless, protective, chassis, signal and reference grounds (Luigi Luigi «Liverpool»)

version 0.2.4 (20110911).

1. added square voltage source (contributed by Alistair Kwan)
2. added buffer and plain amplifier (contributed by Danilo Piazzalunga)
3. added squid and barrier (contributed by Cor Molenaar)
4. added antenna and transmission line symbols contributed by Leonardo Azzinnari
5. added the changeover switch spdt (suggestion of Fabio Maria Antoniali)
6. rename of context.tex and context.pdf (thanks to Karl Berry)
7. updated the email address
8. in documentation, fixed wrong (non-standard) labelling of the axis in an example (thanks to prof. Claudio Beccaria)
9. fixed scaling inconsistencies in quadrupoles
10. fixed division by zero error on certain vertical paths

11. introduced options `straighlabels`, `rotatelabels`, `smartlabels`

version 0.2.3 (20091118).

1. fixed compatibility problem with label option from `tikz`
2. Fixed resizing problem for shape `ground`
3. Variable capacitor
4. polarized capacitor
5. ConTeXt support (read the manual!)
6. `nfet`, `nigfete`, `nigfetd`, `pfet`, `pigfete`, `pigfetd` (contribution of Clemens Helfmeier and Theodor Borsche)
7. `njfet`, `pjfet` (contribution of Danilo Piazzalunga)
8. `pigbt`, `nigbt`
9. *backward incompatibility* potentiometer is now the standard resistor-with-arrow-in-the-middle; the old potentiometer is now known as variable resistor (or `vR`), similarly to variable inductor and variable capacitor
10. triac, thyristor, memristor
11. new property "name" for bipoles
12. fixed voltage problem for batteries in american voltage mode
13. european logic gates
14. *backward incompatibility* new american standard inductor. Old american inductor now called "cute inductor"
15. *backward incompatibility* transformer now linked with the chosen type of inductor, and version with core, too. Similarly for variable inductor
16. *backward incompatibility* styles for selecting shape variants now end are in the plural to avoid conflict with paths
17. new placing option for some tripoles (mostly transistors)
18. mirror path style

version 0.2.2 (20090520).

1. Added the shape for lamps.
2. Added options `europeanresistor`, `europeaninductor`, `americanresistor` and `americaninductor`, with corresponding styles.
3. **Fixed:** error in transistor arrow positioning and direction under negative `xscale` and `yscale`.

version 0.2.1 (20090503).

1. Op-amps added.
2. Added options `arrowmos` and `noarrowmos`.

version 0.2 First public release on CTAN (20090417).

1. **Backward incompatibility:** labels ending with `:angle` are not parsed for positioning anymore.

2. Full use of TikZ keyval features.
3. White background is not filled anymore: now the network can be drawn on a background picture as well.
4. Several new components added (logical ports, transistors, double bipoles, ...).
5. Color support.
6. Integration with `siunitx`.
7. Voltage, american style.
8. Better code, perhaps. General cleanup at the very least.

version 0.1 First public release (2007).

Index of the components

- adc, 17
- afuse, 11
- ageneric, 9
- american and port, 25
- american controlled current source, 19
- american controlled voltage source, 19
- american current source, 12
- american gas filled surge arrester, 14
- american inductor, *see* L
- american nand port, 25
- american nor port, 25
- american not port, 25
- american or port, 25
- american potentiometer, *see* pR
- american resistor, *see* R
- american voltage source, 11
- american xnor port, 26
- american xor port, 25
- ammeter, 9
- amp, 17
- antenna, 8
- asymmetric fuse, *see* afuse

- bandpass, 17
- barrier, 14
- battery, 11
- battery1, 11
- buffer, 27

- C, *see* capacitor
- capacitor, 14
- cground, 7
- circ, 28
- circulator, 22
- cisourcesin, *see* controlled sinusoidal current source
- closing switch, 16
- controlled isourcesin, *see* controlled sinusoidal current source
- controlled sinusoidal current source, 20
- controlled sinusoidal voltage source, 19
- controlled vsourcesin, *see* controlled sinusoidal voltage source
- coupler, 25

- coupler2, 25
- csI, *see* controlled sinusoidal current source
- cspst, *see* closing switch
- csV, *see* controlled sinusoidal voltage source
- currarrow, 28
- cute inductor, *see* L
- cvsourcesin, *see* controlled sinusoidal voltage source

- D*, *see* full diode
- dac, 17
- detector, 18
- Do, *see* empty diode
- dsp, 17

- empty diode, 12
- empty led, 13
- empty photodiode, 13
- empty Schottky diode, 12
- empty tunnel diode, 12
- empty varcap, 13
- empty Zener diode, 12
- european and port, 26
- european controlled current source, 19
- european controlled voltage source, 19
- european current source, 12
- european gas filled surge arrester, 14
- european inductor, *see* L
- european nand port, 26
- european nor port, 26
- european not port, 26
- european or port, 26
- european potentiometer, *see* pR
- european resistor, *see* R
- european variable resistor, *see* vR
- european voltage source, 11
- european xnor port, 26
- european xor port, 26

- fd op amp, 27
- fft, 17
- full diode, 13
- full led, 13
- full photodiode, 13

full Schottky diode, 13
 full tunnel diode, 13
 full varcap, 13
 full Zener diode, 13
 fullgeneric, 9
 fuse, 11

 generic, 9
 ground, 7
 gyrator, 24

 highpass, 17

 isfet, 22
 isourcesin, *see* sinusoidal current
 source

 L, 15
 lamp, 9
 leD*, *see* full led
 leDo, *see* empty led
 lowpass, 17

 match, 8
 memristor, 9
 mixer, 23
 Mr, *see* memristor

 nfet, 21
 nground, 7
 nigbt, 20
 nigfetd, 21
 nigfete, 21
 njfet, 22
 nmos, 20, 21
 npn, 20

 ocirc, 28
 op amp, 27
 open, 9
 opening switch, 16
 oscillator, 8
 ospst, *see* opening switch

 pC, *see* polar capacitor
 pD*, *see* full photodiode
 pDo, *see* empty photodiode
 pfet, 21
 pground, 7
 phaseshifter, 18
 photoresistor, *see* phR

phR, 10
 piattenuator, 18
 pigbt, 20
 pigfetd, 22
 pigfete, 22
 pjfet, 22
 plain amp, 27
 pmos, 20, 21
 pnp, 20
 polar capacitor, 14
 pR, 10
 push button, 16

 R, 10
 rground, 7
 rxantenna, 8

 sD*, *see* full Schottky diode
 sDo, *see* empty Schottky diode
 sground, 7
 short, 9
 sl, *see* sinusoidal current source
 sinusoidal current source, 16
 sinusoidal voltage source, 16
 spdt, 23
 square voltage source, 16
 squid, 14
 sqV, *see* square voltage source
 sV, *see* sinusoidal voltage source

 tattenuator, 18
 tD*, *see* full tunnel diode
 tDo, *see* empty tunnel diode
 tfullgeneric, 9
 tgeneric, 9
 thermistor, *see* thR
 thermistor ntc, *see* thRn
 thermistor ptc, *see* thRp
 thermocouple, 11
 thR, 11
 thRn, 11
 thRp, 11
 thyristor, 23
 TL, 15
 tline, *see* TL
 tlinestub, 8
 toggle switch, 23
 Tr, *see* triac
 transformer, 23, 24
 transformer core, 24

transmission line, *see* TL

triac, 23

twoport, 16

txantenna, 8

Ty, *see* thyristor

vamp, 18

variable american inductor, *see* vL

variable american resistor, *see* vR

variable capacitor, 14

variable cute inductor, *see* vL

variable european inductor, *see* vL

varistor, 10

vC, *see* variable capacitor

VC*, *see* full varcap

vcc, 8

VCo, *see* empty varcap

vco, 17

vL, 15

voltmeter, 9

vphaseshifter, 18

vpiattenuator, 18

vR, 10

vsourcesin, *see* sinusoidal voltage
source

vsourcesquare, *see* square voltage
source

vtattenuator, 18

wilkinson, 22

zD*, *see* full Zener diode

zDo, *see* empty Zener diode