CircuiTi*k*Z

version 0.2.3

Massimo A. Redaelli

November 18, 2009

Contents

1 Introduction

After two years of little exposure only on my personal website[1], I did a major rehauling of the code of CircuiTi*k*Z, fixing several problems and converting everything to Ti*k*Z 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.

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.

[1] `http://home.dei.polimi.it/mredaelli`.

So I based everything with the very impressive (if somewhat verbose at times) Ti*k*Z package.

1.2 Loading the package

```
\usepackage{circuitikz}
```

Ti*k*Z will be automatically loaded.

1.3 License

1.4 Feedback

Much appreciated: `mredaelli@elet.polimi.it`. 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

None, as far as I know.

1.7 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]`).

1.8 ConTEXt compatibility

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

In actually using CircuiTi*k*Z with Ti*k*Z version 2 in ConTEXt 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 Ti*k*Z:

```
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 Ti*k*Z, and I notified the author, but until he fixes it, you know the workaround.

2 Options

- `europeanvoltages`: uses arrows to define voltages, and uses european-style voltage sources;
- `americanvoltages`: uses − and + to define voltages, and uses american-style voltage sources;
- `europeancurrents`: uses european-style current sources;

- `americancurrents`: uses american-style current sources;
- `europeanresistors`: uses rectangular empty shape for resistors, as per european standards;
- `americanresistors`: uses zig-zag shape for resistors, as per american standards;
- `europeaninductors`: 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;
- `europeanports`: uses rectangular logic ports, as per european standards;
- `european`: equivalent to `europeancurrents`, `europeanvoltages`, `europeanresistors`, `europeaninductors`, `europeanports`;
- `american`: equivalent to `americancurrents`, `americanvoltages`, `americanresistors`, `americaninductors`, `americanports`;
- `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.

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]`.

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 CircuiTi*k*Z. These are all `pgf` nodes, so they are usable in both `pgf` and Ti*k*Z.

Each bipole (plus triac and thyristors) are shown using the following command, where `#1` is the name of the component[2]:

```
\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 5.2.

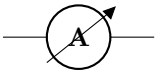
3.1 Monopoles

- Ground (`ground`)

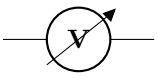
3.2 Bipoles

3.2.1 Instruments

- Ammeter (`ammeter`)



- Voltmeter (`voltmeter`)



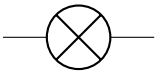
3.2.2 Basic resistive bipoles

- Short circuit (`short`)

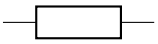
- Open circuit (`open`)

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

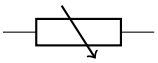
- Lamp (`lamp`)



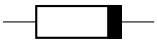
- Generic (symmetric) bipole (`generic`)



- Tunable generic bipole (`tgeneric`)

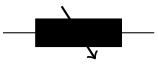


- Generic asymmetric bipole (`ageneric`)

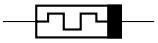


- Generic asymmetric bipole (full) (`fullgeneric`)

- 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:

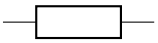
- Resistor (`R`, or `american resistor`)

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

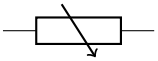
- Potentiometer (`pR`, or `american potentiometer`)

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

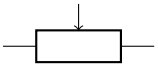
- Resistor (`R`, or `european resistor`)



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



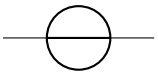
- Potentiometer (`pR`, or `european potentiometer`)



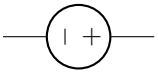
3.2.4 Stationary sources

- Battery (`battery`)

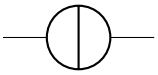
- Voltage source (european style) (`european voltage source`)



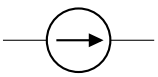
- Voltage source (american style) (`american voltage source`)



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



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

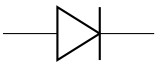


If (default behaviour) `europeancurrents` option is active (or the style `[european currents]` is used), the shorthands `current source`, `isource`, and `I` are equivalent to `european current source`. Otherwise, if `americancurrents` option is active (or the style `[american currents]` is used) they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

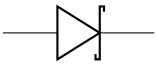
Similarly, if (default behaviour) `europeanvoltages` option is active (or the style `[european voltages]` is used), the shorthands `voltage source`, `vsource`, and `V` are equivalent to `european voltage source`. Otherwise, if `americanvoltages` option is active (or the style `[american voltages]` is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

3.2.5 Diodes and such

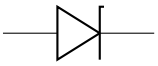
- Empty diode (`empty diode`, or `Do`)



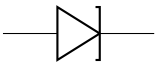
- Empty Schottky diode (`empty Schottky diode`, or `sDo`)



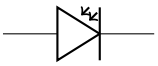
- Empty Zener diode (`empty Zener diode`, or `zDo`)



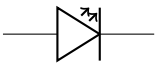
- Empty tunnel diode (`empty tunnel diode`, or `tDo`)



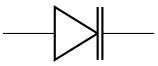
- Empty photodiode (`empty photodiode`, or `pDo`)



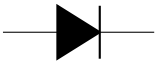
- Empty led (`empty led`, or `leDo`)



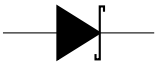
- Empty varcap (`empty varcap`, or `VCo`)



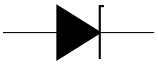
- Full diode (`full diode`, or `D*`)



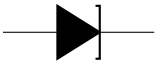
- Full Schottky diode (`full Schottky diode`, or `sD*`)



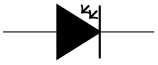
- Full Zener diode (`full Zener diode`, or `zD*`)



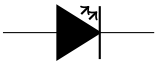
- Full tunnel diode (`full tunnel diode`, or `tD*`)



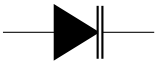
- Full photodiode (`full photodiode`, or `pD*`)



- Full led (`full led`, or `leD*`)



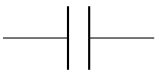
- Full varcap (`full varcap`, or `VC*`)



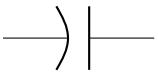
The options `fulldiodes` and `emptydiodes` (and the styles `[full diodes]` and `[empty diodes]`) define which shape will be used by abbreviated commands such that `D`, `sD`, `zD`, `tD`, `pD`, `leD`, and `VC`.

3.2.6 Basic dynamical bipoles

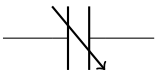
- Capacitor (`capacitor`, or `C`)



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

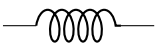


- Variable capacitor (`variable capacitor`, or `vC`)



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`)

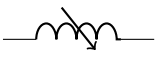


- Variable inductor (`vL`, or `variable cute inductor`)

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

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

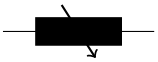
- Variable inductor (`vL`, or `variable american inductor`)



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

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

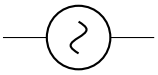
- Variable inductor (`vL`, or `variable european inductor`)



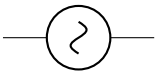
3.2.7 Sinusoidal sources

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`)

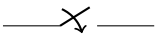


- Sinusoidal current source (`sinusoidal current source`, or `isourcesin`, `sI`)

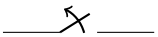


3.2.8 Switch

- Closing switch (`closing switch`, or `cspst`)



- Opening switch (`opening switch`, or `ospst`)

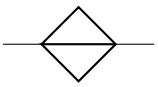


3.3 Tripoles

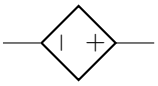
3.3.1 Controlled sources

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

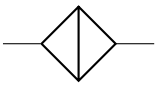
- Controlled voltage source (european style) (`european controlled voltage source`)



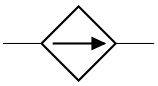
- Controlled voltage source (american style) (`american controlled voltage source`)



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



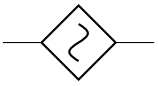
- Controlled current source (american style) (`american controlled current source`)



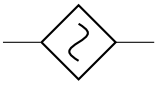
If (default behaviour) `europeancurrents` option is active (or the style `[european currents]` is used), the shorthands `controlled current source`, `cisource`, and `cI` are equivalent to `european controlled current source`. Otherwise, if `americancurrents` option is active (or the style `[american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

Similarly, if (default behaviour) `europeanvoltages` 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`)

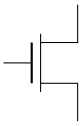


- Controlled sinusoidal current source (`controlled sinusoidal current source`, or `controlled isourcesin, cisourcesin, csI`)

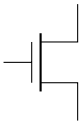


3.3.2 Transistors

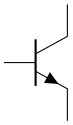
- NMOS (`nmos`)



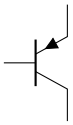
- PMOS (`pmos`)



- NPN (`npn`)

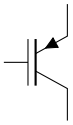


- PNP (`pnp`)



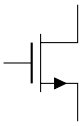
- NPIGBT (`nigbt`)

- PIGBT (`pigbt`)

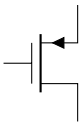


If the option `arrowmos` is used (or after the commant `\ctikzset{tripoles/mos style/arrows}` is given), this is the output:

- NMOS (`nmos`)

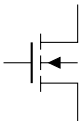


- PMOS (`pmos`)

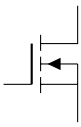


NFETs and PFETs have been incorporated based on code provided by Clemens Helfmeier and Theodor Borsche:

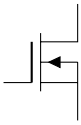
- NFET (`nfet`)



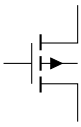
- NIGFETE (`nigfete`)



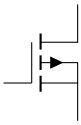
- NIGFETD (`nigfetd`)



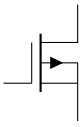
- PFET (`pfet`)



- PIGFETE (`pigfete`)

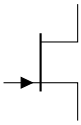


- PIGFETD (`pigfetd`)

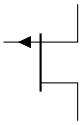


NJFET and PJFET have been incorporated based on code provided by Danilo Piazzalunga:

- NJFET (`njfet`)



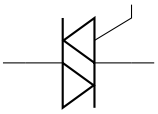
- PJFET (`pjfet`)



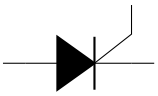
3.3.3 Other bipole-like tripoles

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

- triac (`triac`, or `Tr`)



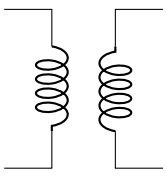
- thyristor (`thyristor`, or `Ty`)



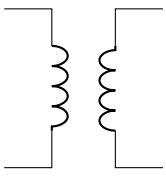
3.4 Double bipoles

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

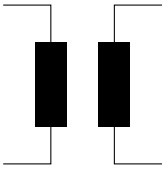
- Transformer (cute inductor) (`transformer`)



- Transformer (american inductor) (`transformer`)

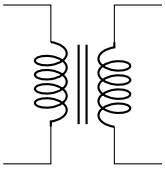


- Transformer (european inductor) (`transformer`)

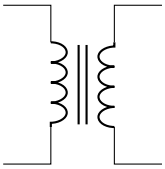


Transformers with core are also available:

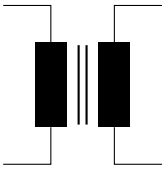
- Transformer core (cute inductor) (`transformer core`)



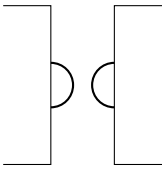
- Transformer core (american inductor) (`transformer core`)



- Transformer core (european inductor) (`transformer core`)

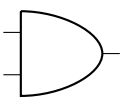


- Gyrator (`gyrator`)

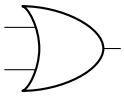


3.5 Logic gates

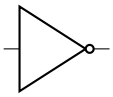
- American AND port (`american and port`)



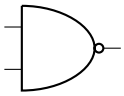
- American OR port (`american or port`)



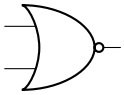
- American NOT port (`american not port`)



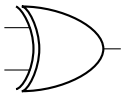
- American NAND port (`american nand port`)



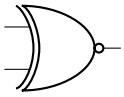
- American NOR port (`american nor port`)



- American XOR port (`american xor port`)

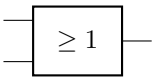


- American XNOR port (`american xnor port`)

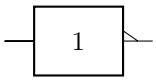


- European AND port (`european and port`)

- European OR port (`european or port`)

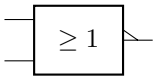


- European NOT port (`european not port`)

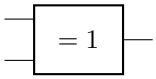


- European NAND port (`european nand port`)

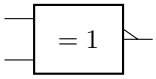
- European NOR port (`european nor port`)



- European XOR port (`european xor port`)



- European XNOR port (`european xnor port`)

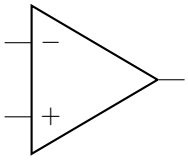


If (default behaviour) `americanports` option is active (or the style `[american ports]` is used), the shorthands `and port`, `or port`, `not port`, `nand port`, `not port`, `xor port`, and `xnor port` are equivalent to the american version of the respective logic port.

If otherwise `europeanports` 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 Operational Amplifier

- Operational amplifier (`op amp`)



3.7 Support shapes

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

▸

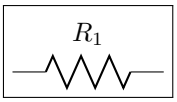
- Connected terminal (`circ`)

•

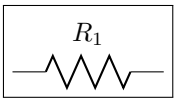
- Unconnected terminal (`ocirc`)

◦

4 Usage

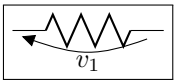



```
\begin{circuitikz}
    \draw (0,0) to[R, l=$R_1$] (2,0);
\end{circuitikz}
```

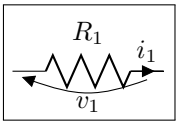



```
\begin{circuitikz}
    \draw (0,0) to[R=$R_1$] (2,0);
\end{circuitikz}
```

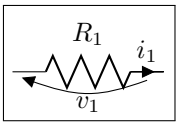

```
\begin{circuitikz}
    \draw (0,0) to[R, i=$i_1$] (2,0);
\end{circuitikz}
```




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

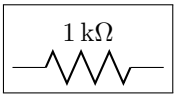



```
\begin{circuitikz}
    \draw (0,0) to[R=$R_1$, i=$i_1$, v=$v_1$] (2,0);
\end{circuitikz}
```



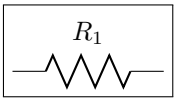

```
\begin{circuitikz}
    \draw (0,0) to[R=$R_1$, i=$i_1$, v=$v_1$] (2,0);
\end{circuitikz}
```

Long names/styles for the bipoles can be used:

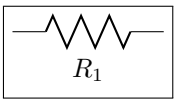



```
\begin{circuitikz}\draw
  (0,0) to[ resistor =1<\kilo\ohm>] (2,0)
;\end{circuitikz}
```

4.1 Labels



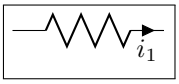

```
\begin{circuitikz}
    \draw (0,0) to[R, l^=$R_1$] (2,0);
\end{circuitikz}
```




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

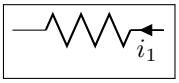
4.2 Currents


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



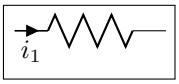

```
\begin{circuitikz}
    \draw (0,0) to[R, i_>=$i_1$] (2,0);
\end{circuitikz}
```


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



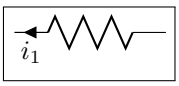

```
\begin{circuitikz}
    \draw (0,0) to[R, i_<=$i_1$] (2,0);
\end{circuitikz}
```


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




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


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



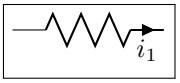

```
\begin{circuitikz}
    \draw (0,0) to[R, i<_=$i_1$] (2,0);
\end{circuitikz}
```

Also


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


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


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

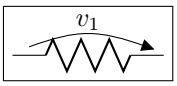



```
\begin{circuitikz}
    \draw (0,0) to[R, i_=$i_1$] (2,0);
\end{circuitikz}
```

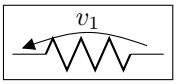
4.3 Voltages

4.3.1 European style

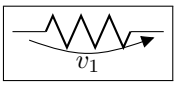
The default, with arrows. Use option `europeanvoltage` or style `[european voltages]`.



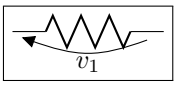

```
\begin{circuitikz}[european voltages]
    \draw (0,0) to[R, v^>=$v_1$] (2,0);
\end{circuitikz}
```




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



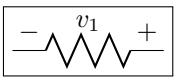

```
\begin{circuitikz}[european voltages]
    \draw (0,0) to[R, v_>=$v_1$] (2,0);
\end{circuitikz}
```



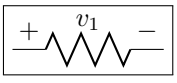

```
\begin{circuitikz}[european voltages]
    \draw (0,0) to[R, v_<=$v_1$] (2,0);
\end{circuitikz}
```

4.3.2 American style

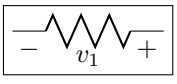
For those who like it (not me). Use option `americanvoltage` or set `[american voltages]`.



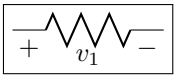

```
\begin{circuitikz}[american voltages]
    \draw (0,0) to[R, v^>=$v_1$] (2,0);
\end{circuitikz}
```




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




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




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

4.4 Nodes

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

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

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

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

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

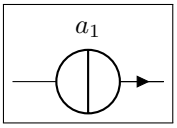
```
\begin{circuitikz}
    \draw (0,0) to[R, *-] (2,0);
\end{circuitikz}
```

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

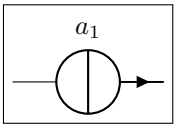
```
\begin{circuitikz}
    \draw (0,0) to[R, *-o] (2,0);
\end{circuitikz}
```

4.5 Special components

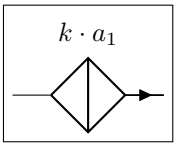
For some components label, current and voltage behave as one would expect:



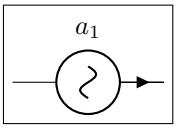

```
\begin{circuitikz}
    \draw (0,0) to[I=$a_1$] (2,0);
\end{circuitikz}
```



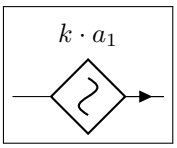

```
\begin{circuitikz}
    \draw (0,0) to[I, i=$a_1$] (2,0);
\end{circuitikz}
```




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

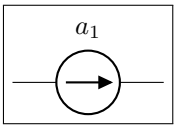



```
\begin{circuitikz}
    \draw (0,0) to[sI=$a_1$] (2,0);
\end{circuitikz}
```

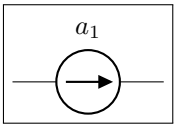



```
\begin{circuitikz}
    \draw (0,0) to[csI=$k\cdot a_1$] (2,0);
\end{circuitikz}
```

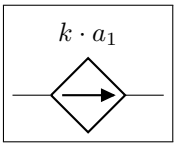
The following results from using the option `americancurrent` or using the style `[american currents]`.



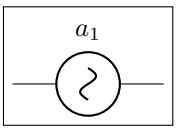

```
\begin{circuitikz}[american currents]
    \draw (0,0) to[I=$a_1$] (2,0);
\end{circuitikz}
```



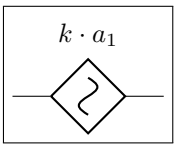

```
\begin{circuitikz}[american currents]
    \draw (0,0) to[I, i=$a_1$] (2,0);
\end{circuitikz}
```




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

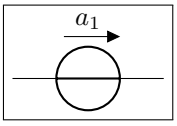



```
\begin{circuitikz}[american currents]
    \draw (0,0) to[sI=$a_1$] (2,0);
\end{circuitikz}
```

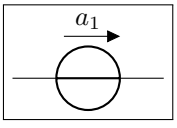



```
\begin{circuitikz}[american currents]
    \draw (0,0) to[csI=$k\cdot a_1$] (2,0);
\end{circuitikz}
```

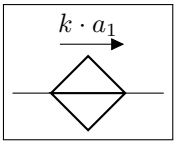
The same holds for voltage sources:



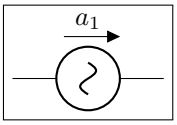

```
\begin{circuitikz}
    \draw (0,0) to[V=$a_1$] (2,0);
\end{circuitikz}
```



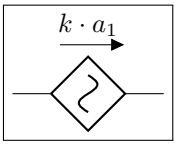

```
\begin{circuitikz}
    \draw (0,0) to[V, v=$a_1$] (2,0);
\end{circuitikz}
```




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

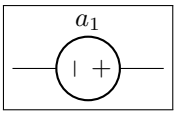



```
\begin{circuitikz}
    \draw (0,0) to[sV=$a_1$] (2,0);
\end{circuitikz}
```

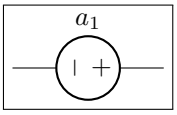



```
\begin{circuitikz}
    \draw (0,0) to[csV=$k\cdot a_1$] (2,0);
\end{circuitikz}
```

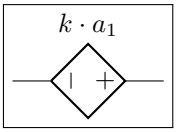
The following results from using the option `americanvoltage` or the style `[american voltages]`.



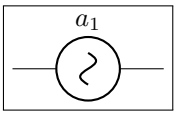

```
\begin{circuitikz}[american voltages]
    \draw (0,0) to[V=$a_1$] (2,0);
\end{circuitikz}
```



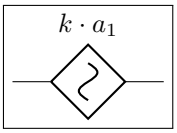

```
\begin{circuitikz}[american voltages]
    \draw (0,0) to[V, v=$a_1$] (2,0);
\end{circuitikz}
```




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



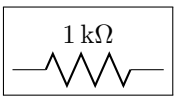

```
\begin{circuitikz}[american voltages]
    \draw (0,0) to[sV=$a_1$] (2,0);
\end{circuitikz}
```



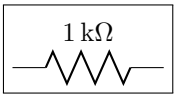

```
\begin{circuitikz}[american voltages]
    \draw (0,0) to[csV=$k\cdot a_1$] (2,0);
\end{circuitikz}
```

4.6 Integration with `siunitx`

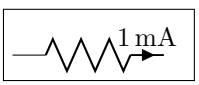
If the option `siunitx` is active (and *not* in ConTEXt), then the following are equivalent:



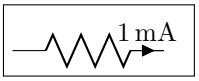

```
\begin{circuitikz}
    \draw (0,0) to[R, l=1<\kilo\ohm>] (2,0);
\end{circuitikz}
```



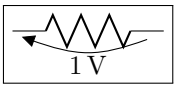

```
\begin{circuitikz}
    \draw (0,0) to[R, l=$\SI{1}{\kilo\ohm}$] (2,0);
\end{circuitikz}
```



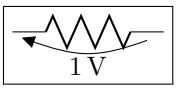

```
\begin{circuitikz}
    \draw (0,0) to[R, i=1<\milli\ampere>] (2,0);
\end{circuitikz}
```




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

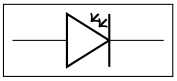



```
\begin{circuitikz}
    \draw (0,0) to[R, v=1<\volt>] (2,0);
\end{circuitikz}
```

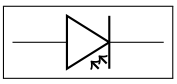



```
\begin{circuitikz}
    \draw (0,0) to[R, v=$\SI{1}{\volt}$] (2,0);
\end{circuitikz}
```

4.7 Mirroring

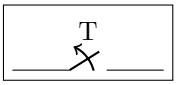


```
\begin{circuitikz}
    \draw (0,0) to[pD] (2,0);
\end{circuitikz}
```

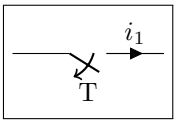


```
\begin{circuitikz}
    \draw (0,0) to[pD, mirror] (2,0);
\end{circuitikz}
```

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

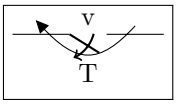



```
\begin{circuitikz}
    \draw (0,0) to[ospst=T] (2,0);
\end{circuitikz}
```




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

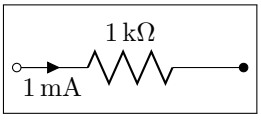
But voltages don't:



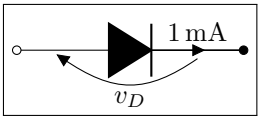

```
\begin{circuitikz}
    \draw (0,0) to[ospst=T, mirror, v=v] (2,0);
\end{circuitikz}
```

Sorry about that.

4.8 Putting them together



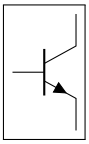

```
\begin{circuitikz}
    \draw (0,0) to[R=1<\kilo\ohm>,
        i>_=1<\milli\ampere>, o-*] (3,0);
\end{circuitikz}
```




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

5 Not only bipoles

Since only bipoles (but see section 5.2) can be placed "along a line", components with more than two terminals are placed as nodes:



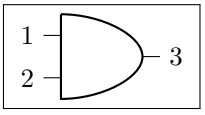
```
\tikz \node[npn] at (0,0) {};
```

5.1 Anchors

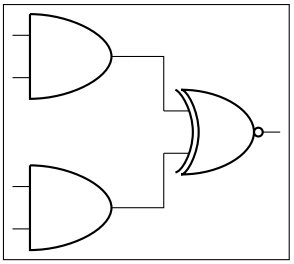
In order to allow connections with other components, all components define anchors.

5.1.1 Logical ports

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

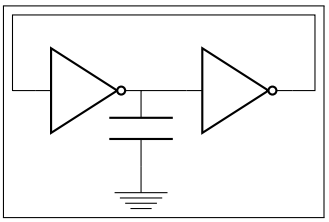



```
\begin{circuitikz} \draw
  (0,0) node[and port] (myand) {}
  (myand.in 1) node[anchor=east] {1}
  (myand.in 2) node[anchor=east] {2}
  (myand.out) node[anchor=west] {3}
;\end{circuitikz}
```



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

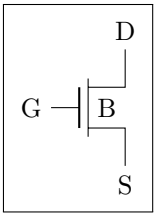
In the case of NOT, there are only `in` and `out` (although for compatibility reasons `in 1` is still defined and equal to `in`):



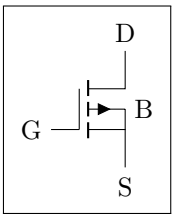
```
\begin{circuitikz} \draw
  (1,0) node[not port] (not1) {}
  (3,0) node[not port] (not2) {}
  (0,0) -- (not1.in)
  (not2.in) -- (not1.out)
  ++(0,-1) node[ground] {} to[C] (not1.out)
  (not2.out) -| (4,1) -| (0,0)
;\end{circuitikz}
```

5.1.2 Transistors

For NMOS, PMOS, NFET, NIGFETE, NIGFETD, PFET, PIGFETE, and PIGFETD transistors one has `base`, `gate`, `source` and `drain` anchors (which can be abbreviated with `B`, `G`, `S` and `D`):

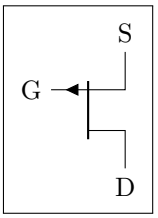



```
\begin{circuitikz} \draw
  (0,0) node[nmos] (mos) {}
  (mos.base) node[anchor=west] {B}
  (mos.gate) node[anchor=east] {G}
  (mos.drain) node[anchor=south] {D}
  (mos.source) node[anchor=north] {S}
;\end{circuitikz}
```



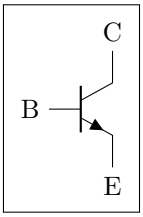

```
\begin{circuitikz} \draw
  (0,0) node[pigfete] (pigfete) {}
  (pigfete.B) node[anchor=west] {B}
  (pigfete.G) node[anchor=east] {G}
  (pigfete.D) node[anchor=south] {D}
  (pigfete.S) node[anchor=north] {S}
;\end{circuitikz}
```

Similarly NJFET and PJFET have `gate`, `source` and `drain` anchors (which can be abbreviated with `G`, `S` and `D`):

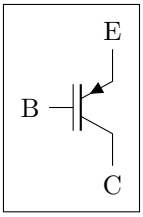



```
\begin{circuitikz} \draw
  (0,0) node[pjfet] (pjfet) {}
  (pjfet.G) node[anchor=east] {G}
  (pjfet.D) node[anchor=north] {D}
  (pjfet.S) node[anchor=south] {S}
;\end{circuitikz}
```

For NPN, PNP, NIGBT, and PIGBT transistors the anchors are `base`, `emitter` and `collector` anchors (which can be abbreviated with `B`, `E` and `C`):

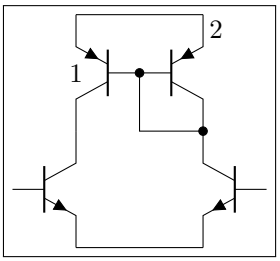



```
\begin{circuitikz} \draw
  (0,0) node[npn] (npn) {}
  (npn.base) node[anchor=east] {B}
  (npn.collector) node[anchor=south] {C}
  (npn.emitter) node[anchor=north] {E}
;\end{circuitikz}
```



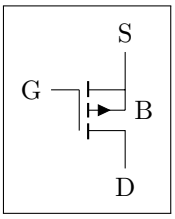

```
\begin{circuitikz} \draw
  (0,0) node[pigbt] (pigbt) {}
  (pigbt.B) node[anchor=east] {B}
  (pigbt.C) node[anchor=north] {C}
  (pigbt.E) node[anchor=south] {E}
;\end{circuitikz}
```

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`):




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

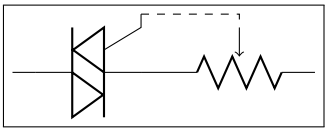
Similarly, transistors can be reflected vertically:




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

5.1.3 Other tripoles

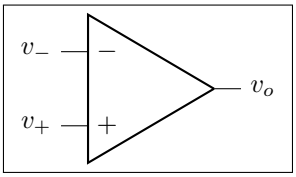
When inserting a thrystor, a triac or a potentiometer, one needs to refer to the third node — gate (`gate` or `G`) for the former two; wiper (`wiper` or `W`) for the latter one. This is done by giving a name to the bipole:



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

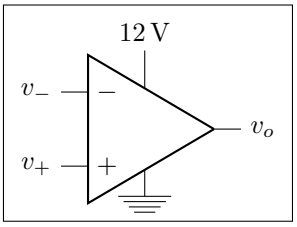
5.1.4 Operational amplifier

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




```
\begin{circuitikz} \draw
  (0,0) node[op amp] (opamp) {}
  (opamp.+) node[left] {$v_+$}
  (opamp.-) node[left] {$v_-$}
  (opamp.out) node[right] {$v_o$}
;\end{circuitikz}
```

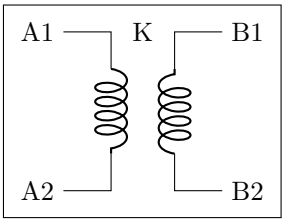
There are also two more anchors defined, `up` and `down`, for the power supplies:



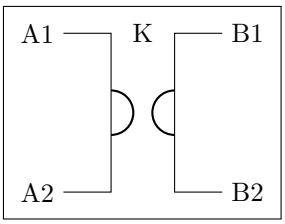

```
\begin{circuitikz} \draw
  (0,0) node[op amp] (opamp) {}
  (opamp.+) node[left] {$v_+$}
  (opamp.−) node[left] {$v_−$}
  (opamp.out) node[right] {$v_o$}
  (opamp.down) node[ground] {}
  (opamp.up) ++ (0,.5) node[above] {\SI{12}{\volt}}
     −− (opamp.up)
;\end{circuitikz}
```

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:



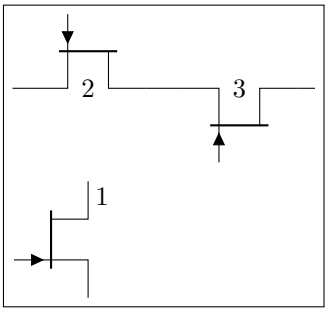

```
\begin{circuitikz} \draw
  (0,0) node[transformer] (T) {}
  (T.A1) node[anchor=east] {A1}
  (T.A2) node[anchor=east] {A2}
  (T.B1) node[anchor=west] {B1}
  (T.B2) node[anchor=west] {B2}
  (T.base) node{K}
;\end{circuitikz}
```




```
\begin{circuitikz} \draw
  (0,0) node[gyrator] (G) {}
  (G.A1) node[anchor=east] {A1}
  (G.A2) node[anchor=east] {A2}
  (G.B1) node[anchor=west] {B1}
  (G.B2) node[anchor=west] {B2}
  (G.base) node{K}
;\end{circuitikz}
```

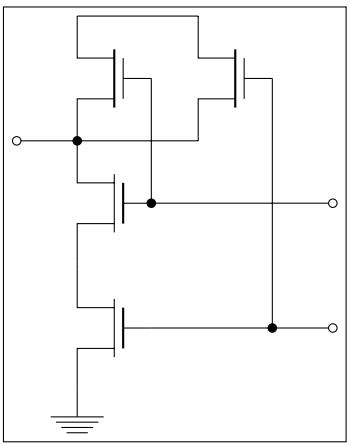
5.2 Transistor paths

For syntactical convenience transistors can be placed using the normal path notation used for bipoles. The transitor 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:




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

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



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

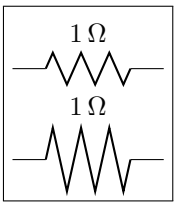
The `name` property is available also for bipoles, although this is useful mostly for triac, potentiometer and thyristor (see 3.3.3).

6 Customization

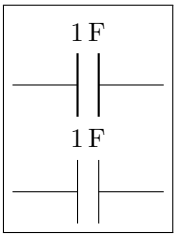
6.1 Parameters

Pretty much all CircuiTi*k*Z 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.

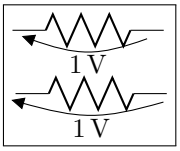
All can be varied using the `\ctikzset` command, anywhere in the code:



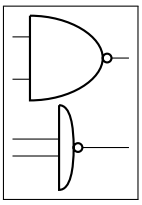

```
\tikz \draw (0,0) to[R=1<\ohm>] (2,0); \par
\ctikzset{bipoles/resistor/height=.6}
\tikz \draw (0,0) to[R=1<\ohm>] (2,0);
```



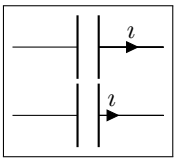

```
\tikz \draw (0,0) to[C=1<\farad>] (2,0); \par
\ctikzset{bipoles/thickness=1}
\tikz \draw (0,0) to[C=1<\farad>] (2,0);
```




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

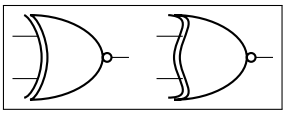


```
\tikz \draw (0,0) node[nand port] {}; \par
\ctikzset{tripoles/american nand port/input height=.2}
\ctikzset{tripoles/american nand port/port width=.2}
\tikz \draw (0,0) node[nand port] {};
```




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

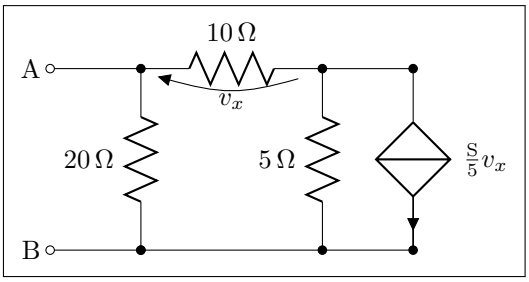
Admittedly, not all graphical properties have understandable names, but for the time it will have to do:



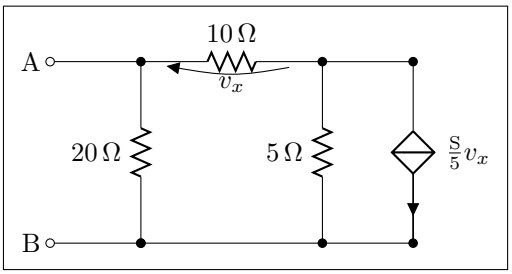
```
\tikz \draw (0,0) node[xnor port] {};
\ctikzset{tripoles/american xnor port/aaa=.2}
\ctikzset{tripoles/american xnor port/bbb=.6}
\tikz \draw (0,0) node[xnor port] {};
```

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 lenghts are relative to this value. For instance:




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

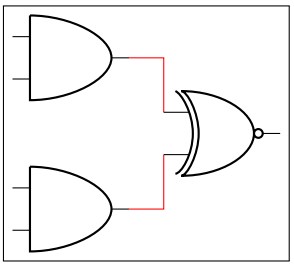



```
\ctikzset{bipoles/length=.8cm}
\begin{circuitikz}[scale=1.2]\draw
  (0,0) node[anchor=east] {B}
        to[short, o-*] (1,0)
        to[R=20<\ohm>, *-*] (1,2)
        to[R=10<\ohm>, v=$v_x$] (3,2) -- (4,2)
        to[cI=$\frac{\siemens}{5} v_x$, *-*] (4,0) -- (3,0)
        to[R=5<\ohm>, *-*] (3,2)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

6.3 Colors

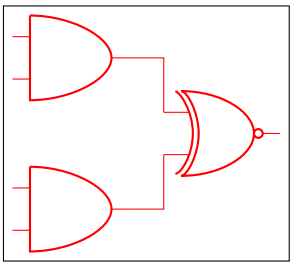
The color of the components is stores in the key `\circuitikzbasekey/color`. CircuiTi*k*Z tries to follow the color set in Ti*k*Z, although sometimes it fails. If you change color in the picture, please do not use just the color name as a style, like `[red]`, but rather assign the style `[color=red]`.

Compare for instance



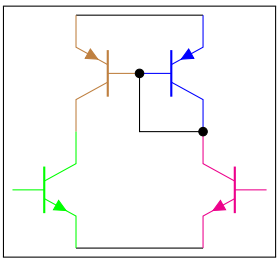
```
\begin{circuitikz} \draw[red]
  (0,2) node[and port] (myand1) {}
  (0,0) node[and port] (myand2) {}
  (2,1) node[xnor port] (myxnor) {}
  (myand1.out) -| (myxnor.in 1)
  (myand2.out) -| (myxnor.in 2)
;\end{circuitikz}
```

and



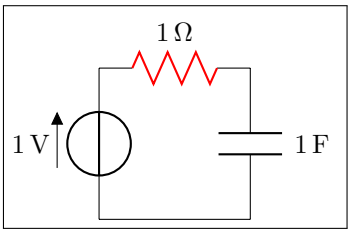
```
\begin{circuitikz} \draw[color=red]
  (0,2) node[and port] (myand1) {}
  (0,0) node[and port] (myand2) {}
  (2,1) node[xnor port] (myxnor) {}
  (myand1.out) -| (myxnor.in 1)
  (myand2.out) -| (myxnor.in 2)
;\end{circuitikz}
```

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



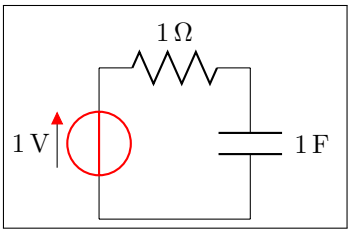
```
\begin{circuitikz} \draw
  (0,0)  node[pnp, color=blue] (pnp2) {}
  (pnp2.B) node[pnp, xscale=-1, anchor=B, color=brown] (pnp1) {}
  (pnp1.C) node[npn, anchor=C, color=green] (npn1) {}
  (pnp2.C) node[npn, xscale=-1, anchor=C, color=magenta] (npn2) {}
  (pnp1.E) -- (pnp2.E) (npn1.E) -- (npn2.E)
  (pnp1.B) node[circ] {} |- (pnp2.C) node[circ] {}
;\end{circuitikz}
```

The all-in-one stream of bipoles poses some challanges, 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`:



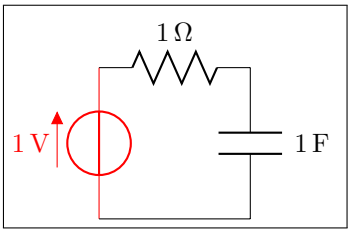

```
\begin{circuitikz} \draw
  (0,0)  to[V=1<\volt>] (0,2)
       { to[R=1<\ohm>, color=red] (2,2) }
         to[C=1<\farad>] (2,0) -- (0,0)
;\end{circuitikz}
```

Which, for some bipoles, can be frustrating:




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

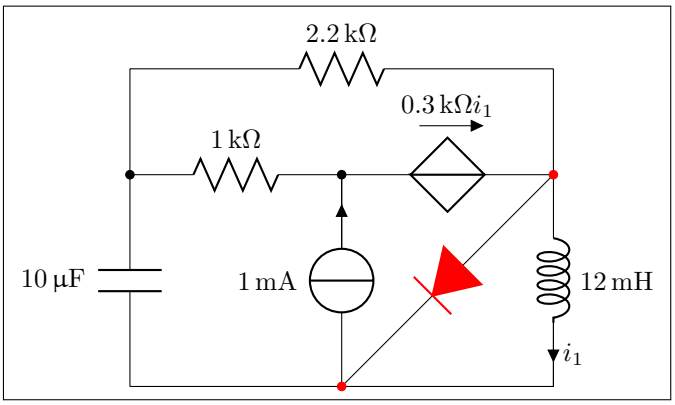
The only way out is to specify different paths:



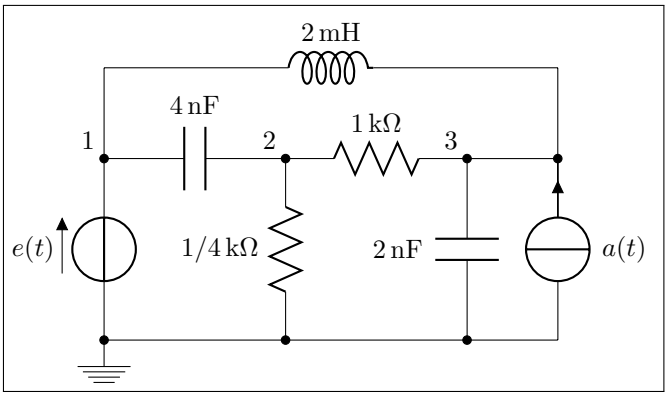

```
\begin{circuitikz} \draw[color=red]
  (0,0)  to[V=1<\volt>, color=red] (0,2);
  \draw (0,2) to[R=1<\ohm>] (2,2)
        to[C=1<\farad>] (2,0) -- (0,0)
;\end{circuitikz}
```

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

7 Examples

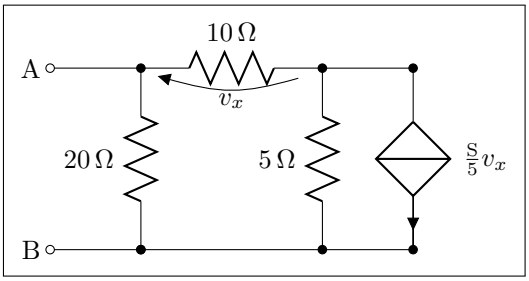



```
\begin{circuitikz}[scale=1.4]\draw
  (0,0) to[C, l=10<\micro\farad>] (0,2) -- (0,3)
        to[R, l=2.2<\kilo\ohm>] (4,3) -- (4,2)
        to[L, l=12<\milli\henry>, i=$i_1$] (4,0) -- (0,0)
  (4,2) { to[D*, *-*, color=red] (2,0) }
  (0,2) to[R, l=1<\kilo\ohm>, *-] (2,2)
        to[cV, v=$\SI{.3}{\kilo\ohm}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

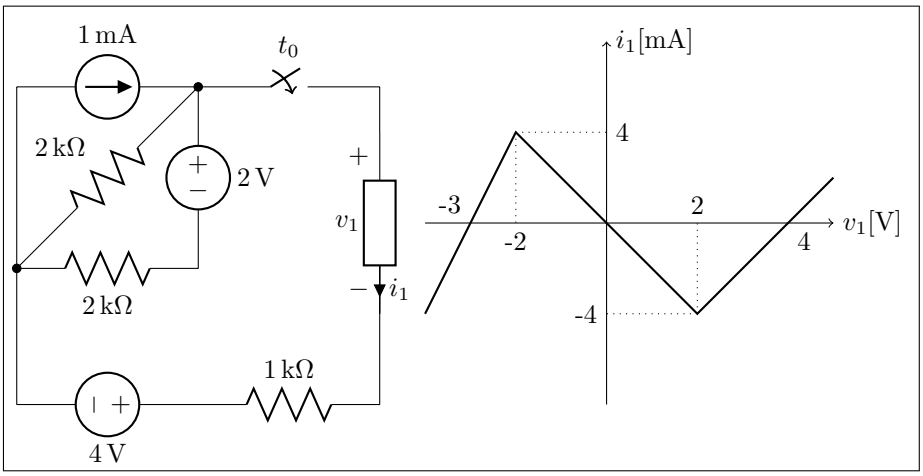



```
\begin{circuitikz}[scale=1.2]\draw
  (0,0) node[ground] {}
        to[V=$e(t)$, *-*] (0,2) to[C=4<\nano\farad>] (2,2)
        to[R, l_=1/4<\kilo\ohm>, *-*] (2,0)
  (2,2) to[R=1<\kilo\ohm>] (4,2)
        to[C, l_=2<\nano\farad>, *-*] (4,0)
  (5,0) to[I, i_=$a(t)$, -*] (5,2) -- (4,2)
  (0,0) -- (5,0)
  (0,2) -- (0,3) to[L, l=2<\milli\henry>] (5,3) -- (5,2)

 {[anchor=south east] (0,2) node {1} (2,2) node {2} (4,2) node {3}}
;\end{circuitikz}
```




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




```
\begin{circuitikz}[scale=1.2, american]\draw
  (0,2) to[I=1<\milli\ampere>] (2,2)
        to[R, l_=2<\kilo\ohm>, *−*] (0,0)
        to[R, l_=2<\kilo\ohm>] (2,0)
        to[V, v_=2<\volt>] (2,2)
        to[cspst, l=$t_0$] (4,2) −− (4,1.5)
        to [generic, i=$i_1$, v=$v_1$] (4,−.5) −− (4,−1.5)
  (0,2) −− (0,−1.5) to[V, v_=4<\volt>] (2,−1.5)
        to [R, l=1<\kilo\ohm>] (4,−1.5);

  \begin{scope}[xshift=6.5cm, yshift=.5cm]
   \draw [−>] (−2,0) −− (2.5,0) node[anchor=west] {$v_1 [\volt]$};
   \draw [−>] (0,−2) −− (0,2) node[anchor=west] {$i_1 [\SI{}{\milli\ampere}]$} ;
   \draw (−1,0) node[anchor=north] {−2} (1,0) node[anchor=south] {2}
         (0,1) node[anchor=west] {4} (0,−1) node[anchor=east] {−4}
         (2,0) node[anchor=north west] {4}
         (−1.5,0) node[anchor=south east] {−3};
   \draw [thick] (−2,−1) −− (−1,1) −− (1,−1) −− (2,0) −− (2.5,.5);
   \draw [dotted] (−1,1) −− (−1,0) (1,−1) −− (1,0)
         (−1,1) −− (0,1) (1,−1) −− (0,−1);
  \end{scope}
\end{circuitikz}
```

8 Revision history

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 Ti*k*Z 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).