

made with the thought of uniformity and usability, which makes possible changing the function of the symbol on the fly, while still preserving the general distinct appearance. Some functions may seem unneeded duplicates, such as  $b-a$ , or  $b/a$ , but the purpose is the ability to keep the functionality without forcing the user to re-wire (this is mostly related to testing, where, too often, it becomes cumbersome).

For example, placing the **Math1** symbol with its default **ABS** value means the symbol will perform the  $abs()$  function; placing the **Math2r** symbol with **x** means multiplication. Both of these are easily recognizable in the schematic, and easily changed by either renaming or by right-clicking the symbol and selecting the desired entry from the drop-down menu. For example switching from  $sin()$  to  $cos()$  means renaming **SIN** to **COS** ([49.2kB.webm](#)); if  $b/a$  is needed instead of  $a/b$ , it can be achieved by simply renaming **/** to **\** ([63.7kB.webm](#)).

The models are divided in three: **mathfunc1.sub**, holding the one-input/variable functions (e.g.  $sin(x)$ ,  $x$  being the input), **mathfunc2.sub**, holding the two-inputs/variables functions (e.g.  $atan2(x,y)$ , with  $x$  and  $y$  being the inputs), and **mathfunc.sub**, holding the oddball 5-input summer.

Sometimes, in addition to the usual input(s) as variable(s), there are some other parameters that can be defined, e.g. the **ATAN2** block has the additional parameters **{a}**, **{b}**, and **{dc}**. All the blocks have floating input and  $1\Omega$  output resistance, unless otherwise noted.

And, finally, in many cases the names of the functions are blatantly obvious and should not need explanations, yet the tables below display them. This has nothing to do with treating the reader as stupid, and all to do with consistency. There are no stand-alone examples here, but some usage can be seen in the [examples](#) section.

Quick menu:


[mathfunc1.sub](#)

[mathfunc2.sub](#)

[mathfunc.sub](#)

[Downloads](#)

Subcircuits in mathfunc1.sub:

- All subcircuits from **mathfunc1.sub** make use of the **Math1** symbol: . The input is marked with a squared arrow.
- Unlike **mathfunc2.sub**, the names of the functions are the same as the name given in the subcircuit definitions.
- Some common parameters, **{y}**, **{phi}**, have different meanings, shown in the **Output** column.
- The hidden parameters, where used, have common meanings:

fmax	[Hz]	Estimated bandwidth for the implicit methods, default <i>1meg</i>
lim	[V]	Internal limits to avoid clipping, default <i>1g</i>
Cout	[F]	Output capacitance for aid in convergence, default <i>0</i>
tripdv, tripdt	[V, s]	LTspice's specific for B-sources, defaults <i>&lt;0,0&gt;</i>

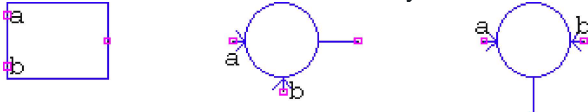
Function	Subcircuit name	Output	Parameters used
$1/x$	" "	$1/V(in)$	lim, fmax
$abs()$	" "	$abs(V(in))$	lim, Cout
$acos()$	" "	$acos(V(in) + \phi)$	phi, tripdv, tripdt
$acosh()$	" "	$acosh(V(in))$	tripdv, tripdt
$asin()$	" "	$asin(V(in) + \phi)$	phi, tripdv, tripdt
$asinh()$	" "	$asinh(V(in))$	tripdv, tripdt
$atan()$	" "	$atan(V(in) + \phi)$	phi, tripdv, tripdt
$atanh()$	" "	$atanh(V(in))$	tripdv, tripdt
$ceil()$	" "	$ceil(V(in))$	tripdv, tripdt

LTspice	Symbol	LTspice	Parameters
<i>ln()</i>	" "	<i>log<sub>e</sub>(V(in))</i>	tripdv, tripdt
<i>log()</i>	" "	<i>log<sub>e</sub>(V(in))</i>	tripdv, tripdt
<i>log10()</i>	" "	<i>log<sub>10</sub>(V(in))</i>	tripdv, tripdt
<i>mod1()</i>	" "	<i>mod(V(in), y)</i>	y, tripdv, tripdt
<i>pow()</i>	" "	<i>pow(V(in), y)</i>	y, tripdv, tripdt
<i>pwr()</i>	" "	<i>pwr(V(in), y)</i>	y, tripdv, tripdt
<i>pwr5()</i>	" "	<i>pwr5(V(in), y)</i>	y, tripdv, tripdt
<i>quant()</i>	" "	<i>int(V(in)*y)/y</i>	y, tripdv, tripdt
<i>round()</i>	" "	<i>round(V(in))</i>	tripdv, tripdt
<i>sgn()</i>	" "	<i>sgn(V(in) + y)</i>	y, tripdv, tripdt
<i>sin()</i>	" "	<i>sin(V(in) + phi)</i>	phi, tripdv, tripdt
<i>sinh()</i>	" "	<i>sinh(V(in))</i>	tripdv, tripdt
<i>sqrt()</i>	" "	<i>√V(in)</i>	lim, fmax
<i>tan()</i>	" "	<i>tan(V(in) + phi)</i>	phi, tripdv, tripdt
<i>tanh()</i>	" "	<i>tanh(V(in))</i>	tripdv, tripdt
<i>u()</i>	" "	<i>if(V(in)+y&gt;0, 1, 0)</i>	y, Cout, tripdv, tripdt
<i>uramp()</i>	" "	<i>if(V(in)+y&gt;0, V(in)+y, 0)</i>	y, Cout, lim

TOP

Subcircuits in mathfunc2.sub:

- mathfunc2.sub makes use of three symbols:


- The subcircuit names for the arithmetic and logic functions are different than the functions themselves, but chosen such as to be (hopefully) easily recognizable.
- The same notes about [parameters](#) are valid here, too.

Function	Subcircuit name	Output	Parameters used
<i>a+b</i>	<b>+</b>	<i>V(a) + V(b)</i>	
<i>a-b</i>	<b>-</b>	<i>V(a) - V(b)</i>	
<i>b-a</i>	<b> </b>	<i>V(b) - V(a)</i>	
<i>a*b</i>	<b>*</b>	<i>V(a)*V(b)</i>	lim
<i>a/b</i>	<b>/</b>	<i>V(a)/V(b)</i>	lim, fmax
<i>b/a</i>	<b>\</b>	<i>V(b)/V(a)</i>	lim, fmax
<i>a==b</i>	<b>~</b> <a href="#">[note]</a>	<i>if(V(a)==V(b), 1, 0)</i>	tripdt
<i>a!=b</i>	<b>!~</b> <a href="#">[note]</a>	<i>if(V(a)!=V(b), 1, 0)</i>	tripdt
<i>atan2(a,b)</i>	<b>ATAN2</b>	<i>atan2(sgn(a)*V(a), sgn(b)*V(b)) + dc</i>	a, b, dc, tripdv, tripdt
<i>hypot(a,b)</i>	<b>HYPOT</b>	<i>hypot(V(a), V(b))</i>	lim, fmax
<i>max(a,b)</i>	<b>MAX</b>	<i>max(V(a), V(b))</i>	lim
<i>min(a,b)</i>	<b>MIN</b>	<i>min(V(a), V(b))</i>	lim
<i>mod(a,b)</i>	<b>MOD</b>	<i>mod(V(a), V(b))</i>	tripdv, tripdt

<sup>[note]</sup> Names with "=" won't be recognized by LTspice, and using "==", or "!=", will make the names visible in the list, but not usable. Thus, "~" and "!~" were chosen, hopefully, as visually and functionally similar alternatives.





