

## Known Issues Running LTspice® Models in SIMetrix

The following are issues that have been found in actual models. It is not an exhaustive list of all issues

### log() function problem

log() in SIMetrix means log to base 10 originating from SPICE3. log() in LTspice® mean log to base e. From version 9.2 this is changing for arbitrary source expressions and .PARAM expressions. The script language and SIMPLIS .VAR statements will remain unchanged.

Some LTspice® models use the log() function and for simulating with version 9.1 or earlier these need to be edited to use ln() which means log to base e.

For SIMetrix scripts and SIMPLIS .VAR statements we recommend never use log(). Use ln() for log to base e and log10 for log to base 10.

### Resistors with Circuit Dependent Variables

Example:

```
R1 CR+ CR- R=220µ*(1+0.0039*(V(dT)-25))
```

Resistor's value is dependent on V(dT) which is the voltage on the dT node. The default SIMetrix resistor model does not accept such expressions but the LEVEL 2 resistor does. To convert to a LEVEL 2 resistor change to:

```
R1 CR+ CR- rmod "220µ*(1+0.0039*(V(dT)-25))"  
.model rmod R LEVEL=2 ACRESMOD=1
```

The LEVEL 2 resistor was originally designed for compatibility with Hspice which does not model the resistor consistently in AC analysis whereas LTspice® behaves correctly. By default SIMetrix replicates the Hspice behaviour but this can be disabled using the parameter ACRESMOD=1 as shown above.

An alternative is to convert the device to an arbitrary source which may be easier but such a device is noiseless which would be incorrect for noise analysis.

### Use of delay() Function in Arbitrary Sources

delay() implements a time delay. SIMetrix supports this in Verilog-A but not currently in arbitrary sources. A delay can be implemented using a transmission line

### Brace Character: '{' and '}'

LTspice® uses braces in places where SIMetrix does not expect them and vice-versa. Most issues are resolved using this option setting:

```
.options ltspicecompatibility=1
```

### .PARAM Used to Define Default Subcircuit Parameters

The following is allowed in LTspice® but will lead to an error in SIMetrix

```
.SUBCKT SUBNAME 1 2 3
.PARAM P1=1
.PARAM P2=2
...
.ENDS
X1 A B C SUBNAME params: P1=2 P2=3
```

The error message will be something like:

```
test.cir (3): .PARAM P1=1
*** ERROR *** Parameter 'X1.P1' has already been defined
```

```
test.cir (4): .PARAM P2=2
*** ERROR *** Parameter 'X1.P2' has already been defined
```

To fix, replace the .PARAM statement with params: as follows:

```
.SUBCKT SUBNAME 1 2 3 params: P1=1 P2=2
...
.ENDS
X1 A B C SUBNAME params: P1=2 P2=3
```

## Out of Order .PARAM Statements

Some models use values for parameters that are defined later in the netlist. SIMetrix will accept these under some circumstances but not all, especially if the .PARAM statements are inside a subcircuit definition. Rearrange .PARAM statements to fix.

## Unknown Function “Id()”

Known to affect models ADP1761,1762,1763,1764,1765. Edit model to replace:

```
Id(device)
```

with

```
I(device,d)
```

## Model Name Containing a ‘

Known to affect AD8029 leads to error:

```
*** ERROR *** Cannot find definition for model '6.5uA'
```

Edit all occurrences of 6.5uA to 65uA to fix

## Use of Unsupported Devices

E.g. SRFLOP, DFLOP and VARISTOR devices using a table definition. These would need to be converted to equivalent SIMetrix devices. Known to affect these models:

```
2862-1 2862-2 2863-1 2863-2 2864-1 2864-2 2865 3011 3026 3063 6210 6560 6561 ADA4522
ADA4625 ADA4807 ADG5248F ADG5249F ADP7102 ADP7104 ADP7105 ADP7118 LTZ1000 LTZ1000A
```

## Known Issues with Application Circuits

### Generic Diode Model Name 'D'

A diode in a schematic with value D automatically leads to the generation of the line:

```
.MODEL D D
```

This needs to be added manually in SIMetrix. Just add that line to the F11 window

### Floating Nodes

Nodes with no DC path to ground are accepted in LTspice®. In SIMetrix these lead to a singular matrix error. LTspice® initialises floating nodes to zero volts. This is harmless in most situations but cannot be guaranteed to be, so SIMetrix does not provide a means of replicating this directly.

Solution is to provide a DC path to ground. This can be done with an initial condition or a high value resistor.

### Extra Parameters on Signal Sources

Some signal sources have additional parameters which SIMetrix does not accept.

Pulse sources can have an additional parameter to define a fixed number of cycles. Usually this is zero in which case it can simply be removed. If it is non-zero, a PWLS source will need to be used to replicate the behaviour.

Sine sources can have up to two additional parameters. Again if they are zero just omit them. If they are non-zero a PWLS source will need to be designed for equivalent behaviour

## Models Supplied with LTspice® that Need Editing or .OPTION Settings

1. ADP1761,1762,1763,1764,1765. See "Unknown Function "ld()" above
2. AD8029. See "Model Name Containing a '.'" above
3. 1067-50 1068-50 1389-1.25 1389-2.5 1389-4.096 1389-5 AD8031 AD8040 AD8041 AD820 AD823 AD8675 ADA4177-1

These models require:

```
.OPTIONS ltspicecompatibility=1
```

## Models Supplied with LTspice® the Use Unsupported Devices

The following use unsupported devices and are not easily modified to work with SIMetrix:

2862-1 2862-2 2863-1 2863-2 2864-1 2864-2 2865 3011 3026 3063 6210 6560 6561 ADA4522  
ADA4625 ADA4807 ADG5248F ADG5249F ADP7102 ADP7104 ADP7105 ADP7118 LTZ1000 LTZ1000A