Parameter Index

In the following index, the number following each parameter name indicates where to find the description of that parameter.

Stability Analysis (stb)

Description

The STB analysis linearizes the circuit about the DC operating point and computes the loop gain, gain and phase margins (if the sweep variable is frequency), for a feedback loop or a gain device.

Spectre can perform the analysis while sweeping a parameter. The parameter can be frequency, temperature, component instance parameter, component model parameter, or netlist parameter. If changing a parameter affects the DC operating point, the operating point is recomputed on each step. You can sweep the circuit temperature by giving the parameter name as `temp` with no `dev` or `mod` parameter. You can sweep a netlist parameter by giving the parameter name with no `dev`, or `mod` parameter. After the analysis has completed, the modified parameter returns to its original value.

Definition

```
Name stb parameter=value ...
```

Parameters

1 `prevoppoint=no` Use operating point computed on the previous analysis. Possible values are `no` or `yes`.

Sweep interval parameters

2 `start=0` Start sweep limit.

3 `stop` Stop sweep limit.

4 `center` Center of sweep.

5 `span=0` Sweep limit span.

6 `step` Step size, linear sweep.

7 `lin=50` Number of steps, linear sweep.

8 `dec` Points per decade.

9 `log=50` Number of steps, log sweep.

10 `values=[...]` Array of sweep values.

Sweep variable parameters

11 `dev` Device instance whose parameter value is to be swept.

12 `mod` Model whose parameter value is to be swept.

13 `param` Name of parameter to sweep.

14 `freq (Hz)` Frequency when parameter other than frequency is being swept.

Probe parameters

15 `probe` Probe instance around which the loop gain is calculated.

State-file parameters

16 `readns`	File that contains estimate of DC solution (nodeset).

Output parameters

17 `save`	Signals to output. Possible values are `all`, `lvl`, `allpub`, `lvlpub`, `selected`, or `none`.
18 `nestlvl`	Levels of subcircuits to output.
19 `oppoint=no`	Should operating point information be computed, and if so, where should it be sent. Operating point information would not be output if operating point is computed in the previous analysis and is unchanged. Possible values are `no`, `screen`, `logfile`, or `rawfile`.

Convergence parameters

20 `restart=yes`	Restart the DC solution from scratch if any condition has changed. If not, use the previous solution as initial guess. Possible values are `no` or `yes`.

Annotation parameters

21 `annotate=sweep`	Degree of annotation. Possible values are `no`, `title`, `sweep`, `status`, or `steps`.
22 `stats=no`	Analysis statistics. Possible values are `no` or `yes`.
23 `title`	Analysis title.

You can specify sweep limits by giving the end points or by providing the center value and the span of the sweep. Steps can be linear or logarithmic, and you can specify the number of steps or the size of each step. You can give a step size parameter (`step`, `lin`, `log`, `dec`) to determine whether the sweep is linear or logarithmic. If you do not give a step size parameter, the sweep is linear when the ratio of stop to start values is less than 10, and logarithmic when this ratio is 10 or greater. All frequencies are in Hertz.

The small-signal analysis begins by linearizing the circuit about an operating-point. By default this analysis computes the operating-point if it is not known, or recomputes it if any significant component or circuit parameter has changed. However, if a previous analysis computed an operating point, you can set `prevoppoint=yes` to avoid recomputing it. For example, if you use this option when the previous analysis was a transient analysis, the operating point is the state of the circuit on the final time point.

Understanding Loop based and Device Based Algorithms

Two algorithms--the loop based and the device based, are available for small-signal stability analysis. Both algorithms are based on the calculation of Bodes return ratio. Loop gain waveform, gain margin, and phase margin are the analysis output.

The `probe` parameter must be specified to perform stability analysis. When it points to a current probe or voltage source instance, the loop based algorithm will be invoked; when it points to a supported active device instance, the device based algorithm will be invoked.

Loop Based Algorithm

The loop based algorithm calculates the true loop gain that consists of normal loop gain and reverse loop gain. The loop based algorithm requires the `probe` being placed on the feedback loop to identify and characterize the particular loop of interest. The introduction of the probe component should not change any of the circuit characteristics.

The loop based algorithm provides accurate stability information for single loop circuits, and multiloop circuits in which a `probe` component can be placed on a critical wire to break all loops. For a general multiloop circuit, such a critical wire may not be available. The loop based algorithm can only be performed on individual feedback loops to ensure they are stable. Although the stability of all feedback loops is only a necessary condition for the whole circuit to be stable, the multiloop circuit tends to be stable if all individual loops are associated with reasonable stability margins.

Device Based Algorithm

The device based algorithm calculates the loop gain around a particular active device. This algorithm is often applied to assess the stability of circuit design in which local feedback loops cannot be neglected; the loop based algorithm cannot be performed for these applications since the local feedback loops are inside the devices, they are not accessible from the schematic level or netlist level to insert the `probe` component.

With the `probe` parameter points to a particular active device, the dominant controlled source in the device will be nulled during the analysis. The dominant controlled source is defined as by nulling this source renders the active device to be passive. The device based algorithm produces accurate stability information for a circuit in which a critical active device can be identified such that nulling the dominant gain source of this device renders the whole network to be passive.

Stability Analysis of Differential Feedback Circuits

A balanced fully differential feedback circuit is illustrated below:

```
                |---------[ ZF ]---------|
                |                        |
                |                        X1
                |  |------------------|  |
                |  |                  |  |
   ... ---[ ZS ]---|--|in+        out- |--|---[ ZL ]--- ...
                   |      OPAMP        |
   ... ---[ ZS ]---|--|in-        out+ |--|---[ ZL ]--- ...
                |  |                  |  |
                |  |------------------|  |
                |                        X2
                |                        |
                |---------[ ZF ]---------|
```

The feedback loops are broken at X1 and X2, with x1in and x2in being the input side nodes, x1out and x2out being the output side nodes. The following subcircuit connects these four nodes togather:

subckt diffprobe x1in x2in x1out x2out

ibranch inout x1out iprobe

vinj inout x1in iprobe

evinj x2in x2out x1in x1out vcvs gain=0

fiinj 0 x2out pcccs probes=[ibranch vinj] coeffs=[0 1 1] gain=0

ends diffprobe

Let `diffprobe_inst` be the instance of subcircuit `diffprobe`, the following analysis measures the differential-mode loop gain:

DMalterv alter dev=diffprobe_inst.evinj param=gain value=-1

DMalteri alter dev=diffprobe_inst.fiinj param=gain value=-1

DMloopgain stb probe=diffprobe_inst.vinj

and the following analysis measures the common-mode loop gain:

CMalterv alter dev=diffprobe_inst.evinj param=gain value=1

CMalteri alter dev=diffprobe_inst.fiinj param=gain value=1

CMloopgain stb probe=diffprobe_inst.vinj

Parameter Index

In the following index, the number following each parameter name indicates where to find the description of that parameter.

annotate 21	log 9	probe 15	stats 22
center 4	mod 12	readns 16	step 6
dec 8	nestlvl 18	restart 20	stop 3
dev 11	oppoint 19	save 17	title 23
freq 14	param 13	span 5	values 10