Agilent Technologies

Advanced Design System 2011.01

Feburary 2011

DC Simulation

Acknowledgments

The following third-party libraries are used by the NlogN Momentum solver:

Intel@ Math Kernel Library, http://www.intel.com/software/products/mkl

Errata The ADS product may contain references to "HP" or "HPEESOF" such as in file names and directory names. The business entity formerly known as "HP EEsof" is now part of Agilent Technologies and is known as "Agilent EEsof". To avoid broken functionality and to maintain backward compatibility for our customers, we did not change all the names and labels that contain "HP" or "HPEESOF" references.

Warranty The material contained in this document is provided "as is", and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied, with regard to this documentation and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

Technology Licenses The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license. Portions of this product include the SystemC software licensed under Open Source terms, which are available for download at http://systemc.org/ . This software is redistributed by Agilent. The Contributors of the SystemC software provide this software "as is" and offer no warranty of any kind, express or implied, including without limitation warranties or conditions or title and non-infringement, and implied warranties or conditions merchantability and fitness for a particular purpose. Contributors shall not be liable for any damages of any kind including without limitation direct, indirect, special, incidental and consequential damages, such as lost profits. Any provisions that differ from this disclaimer are offered by Agilent only.

Restricted Rights Legend U.S. Government Restricted Rights. Software and technical data rights granted to the federal government include only those rights customarily provided to end user customers. Agilent provides this customary commercial license in Software and technical data pursuant to FAR 12.211 (Technical Data) and 12.212 (Computer Software) and, for the Department of Defense, DFARS 252.227-7015 (Technical Data - Commercial Items) and DFARS 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation).

About DC Simulation

This documentation describes DC simulation used in ADS. It outlines when to use a DC simulation, how to set it up, and the data it generates. Examples are provided to show how to use this simulation. Detailed information describes the parameters, theory of operation, and troubleshooting information.

DC simulation calculates the DC operating point characteristics of a design under test (DUT). DC analysis is fundamental to all Analog/RF simulations, so it is used for all Analog/RF designs. Following a topology check, the simulator performs an analysis of the DC operating point, including the circuit's power consumption.

You can also set up the DC simulation to sweep one or more parameters, enabling you to perform tasks such as verifying model parameters by comparing the simulated DC transfer characteristics (I-V curves) of the model with actual measurements.

In ADS, the DC simulation component is available in the Simulation-DC palette.

Refer to the following topics for details on DC simulation:

- *Using DC Simulation* (cktsimdc) explains when to use DC simulation, describes the minimum setup requirements, and gives a brief explanation of the DC simulation process.
- *Examples of DC Simulation* (cktsimdc) contains detailed examples of DC simulation setups for ADS, one for calculating a single DC point and another that sweeps variables to generate a set of I-V curves.
- *DC Simulation Parameters* (cktsimdc) provides details about the parameters available in ADS for the DC simulation controller.
- *Theory of Operation for DC Simulation* (cktsimdc) provides details about DC simulation operation. This is provided if you have an interest in understanding the underlying technology. Typical users should not need to examine this material.
- *Troubleshooting a DC Simulation* (cktsimdc) offers suggestions on what to do in the rare event that a DC simulation does not converge or converges too slowly.
- *References for DC Simulation* (cktsimdc) lists the references relevant to the theory of operation. This is provided if you have an interest in understanding the underlying technology. Typical users should not need to examine this material.

Using DC Simulation

This section describes when to use a DC simulation, how to set it up, and the basic simulation process used to collect data.

License Requirements

The DC simulation uses the Linear Simulator license (sim_linear). You must have this license to run DC simulations. You can work with examples described here and installed with the software without the license, but you will not be able to simulate them.

When to Use DC Simulation

A single-point DC simulation automatically precedes every AC, S-parameter, transient, harmonic balance, and circuit envelope simulation. This serves as the starting point for these simulations. For AC and S-parameter simulations, it determines linearized models for the nonlinear components. For transient, harmonic balance, and circuit envelope simulations, it determines an initial estimate used for nonlinear simulation.

A DC simulation also can be done manually in ADS by using the DC simulation controller. Start by creating your design, add any relevant current probes, then identify and name the nodes from which you want to collect data. The DC simulation setup then provides for both single-point and swept simulations. Swept variables can be related to voltage or current source values, or to other component parameter values. By performing a DC swept bias or swept variable simulation, you can check the operating point of the circuit against a swept parameter such as a bias supply voltage or a temperature.

Use DC simulation to:

- Verify the proper DC operating characteristics of the design under test.
- Determine the power consumption of your circuit.
- Verify model parameters by comparing the DC transfer characteristics (I-V curves) of the model with actual measurements.
- Display voltages and currents after a simulation.
- Provide data for DC back-annotation.

How to Use DC Simulation

The following guidelines will help you set up a DC simulation in ADS.

- Add the DC simulation component to the schematic. If you do not edit the default settings, values will be calculated for the current probes and nodes added to the circuit, based on the settings of the schematic's other components.
- To sweep a parameter over a range, such as varying an input voltage or changing a resistor value, double-click the simulation component and select the Sweep tab. Enter the name of the parameter to sweep. Select the sweep type and enter the range.
- If additional sweeps are needed, an external parameter sweep can be added.
- For details about each parameter, click *Help* from the open dialog box.

What Happens During DC Simulation

The DC simulation technique relies on an iterative process of mathematical convergence toward a solution. By manipulating a system of nonlinear ordinary differential equations, it solves for an equilibrium point in the nonlinear algebraic equations that describe a circuit based on a set of assumptions. For details, see *Theory of Operation for DC Simulation* (cktsimdc).

Examples of DC Simulation

There are two examples that describe how to set up and run a DC simulation in ADS:

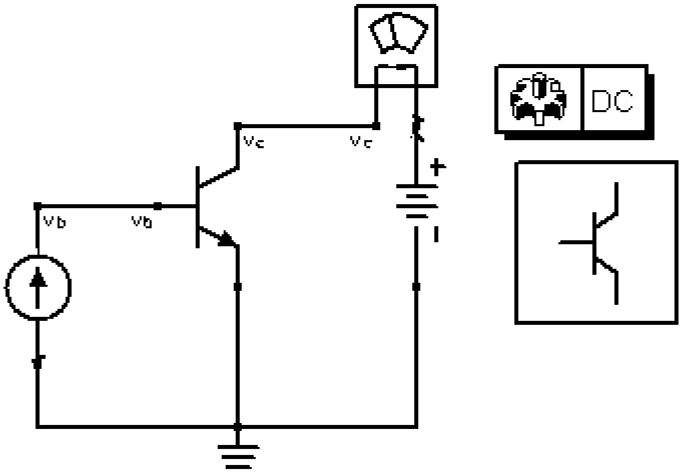
- Simulating a BJT in ADS shows how to calculate a single-point DC bias of a BJT.
- Sweeping Parameters in ADS shows how to sweep one or more circuit parameters over a range of values during a simulation.

Simulating a BJT in ADS

The following figure illustrates the setup for a DC simulation of a BJT. This simulation calculates a single DC operating point.

> **Note**
> This design, *DC1*, is in the *Examples* directory under *Tutorial/SimModels_wrk*. The results are in *DC1.dds*.

Figure: Setup for a single-point DC bias simulation



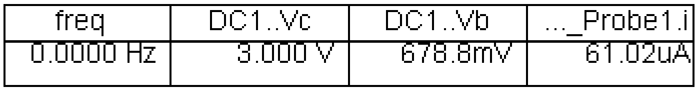
To perform a single-point DC simulation:

1. Place the BJT instance and model on the schematic.
2. From the **Sources-Freq Domain** palette, select **V_DC**. Place the DC voltage source on the schematic to provide the collector voltage and modify the voltage as needed.

   > **Note**
   > Ensure that sources are connected either by wires or by means of a named connection (*Insert > Wire/Pin Label*).

3. From the **Sources-Freq Domain** palette, select **I_DC**. Place the source on the schematic to provide base current and modify the current as needed.
4. From the **Probe Components** palette, select **I_Probe**. Place this current probe in an appropriate location in the circuit.
5. From the **Simulation-DC** palette, select and place the **DC** simulation component on the schematic.

6. **Simulate**. When the simulation is finished, you can immediately exercise the following DC simulation options:
   - For a direct representation of currents and voltages at all nodes on the schematic, choose *Simulate > Annotate DC Solution*.
   - For the details of the device, choose *Simulate > Detailed Device Operating Point* or *Brief Device Operating Point*, then click the device when the crosshairs appear.
   - Choose *Simulate > Clear DC Annotation* to remove the annotation.
7. In the Data Display, the data from the simulation is presented as a list of DC currents and voltages:



freq	DC1..Vc	DC1..Vb	..._Probe1.i
0.0000 Hz	3.000 V	678.8mV	61.02uA

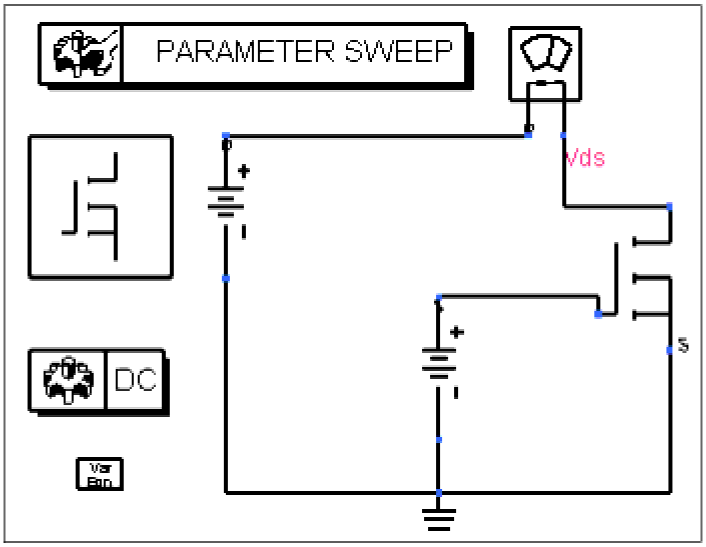
Sweeping Parameters in ADS

By performing a DC swept bias or a swept variable simulation, you can check the operating point of the circuit against a swept parameter such as temperature or bias supply voltage. The resulting data can be plotted to resemble the appearance of a curve tracer.

The following figure illustrates a setup for performing an idealized, swept DC bias simulation of a MOSFET. Vdd and Vgg are swept across a range of voltages. The result is a family of curves representing drain current versus drain voltage, for varying values of gate voltage.

> **Note**
> This design, *DC2*, is in the *Examples* directory under *Tutorial/SimModels_wrk*. The results are in *DC2.dds*.

Figure: Example setup for a swept DC simulation



> **Note**
> This example uses a referenced model. The parameter *cells* refers to the extraction parameter known as *binning* and reflects the electrical characteristics of this particular model. The number of cells has been referenced in a VarEqn component.

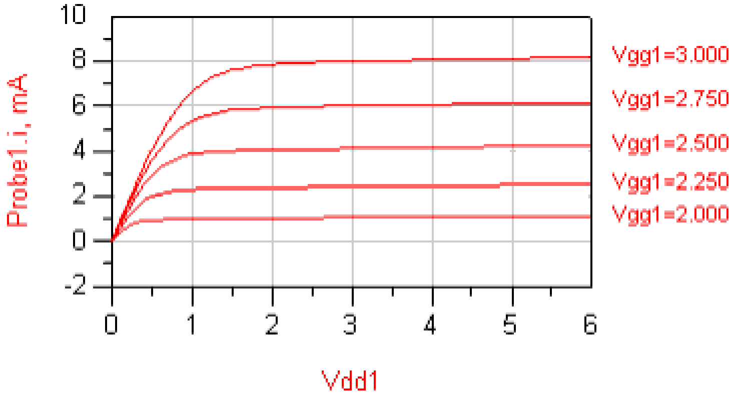
To perform an idealized, swept DC bias simulation:

1. Place the MOSFET instance and model on the schematic.
2. From the **Sources-Freq Domain** palette, select **V_DC**. Place this V_DC component (SRC1) between the base of the device and ground. Edit this component and define Vdc = Vgg1, rather than a numerical value.
3. Place another V_DC component (SRC2) between the drain of the device and ground. Open this component and define Vdc = Vdd1, rather than a numerical value.
   This makes it possible to sweep these values. Any swept parameter must be initialized by means of an equation, as will be demonstrated later.
4. From the **Probe Components** palette, select **I_Probe**. Place the probe between the drain and SRC2, and rename it if you like (in this case, it has been called Probe1).
5. From the **Simulation-DC** palette, select and place the **DC** simulation component on the schematic.
6. Edit the **DC** component by selecting the **Sweep** tab and modifying the sweep parameters as follows:
   - Parameter to sweep = **Vdd1**. This appears as *SweepVar* on the schematic if this parameter is displayed on the schematic.

     > **Note**
     > Variables entered in this field will appear in quotes on the schematic. If you enter a variable directly on the schematic (in this case, as the right-hand side of the SweepVar statement of the DC component), you must surround the variable with double quotes. This applies to the other user-defined variables shown in subsequent steps.

   - Start = **0**
   - Stop = **6**
   - Step = **0.1**
7. Click **OK** to accept changes and close the dialog box.
8. Choose **Simulation-DC** palette, select **Prm Swp** (ParamSweep). Place this component on the schematic and edit it as follows:
   - On the Sweep tab:
     - Parameter to sweep = **Vgg1**
     - Start = **2**
     - Stop = **3**
     - Step = **0.25**
   - On the Simulations tab:
     - Simulation 1 = **DC1**. This appears as *SimInstanceName[1]* on the schematic if this parameter is displayed on the schematic.
9. Click **OK** to accept changes and close the dialog box.
10. Choose the **Data Items** palette, select **Var eqn** (Variables and equations). Place this component on the schematic and enter the following equations:
    - **Vdd1 = 4**
    - **Vgg1 = 2.5**
11. Click **OK** to accept changes and close the dialog box.
12. **Simulate**. When the simulation is finished, plot the probe current (Probe1.i). The resulting family of curves representing drain current versus drain voltage, for varying

values of gate voltage, is shown next:



DC Simulation Parameters

ADS provides access to DC simulation parameters enabling you to define aspects of the simulation listed in the following table:

Tab Name	**Description**	**For details, see...**
Sweep	Defining sweep characteristics.	Setting Up a Sweep
Parameters	Provides options to set the following: • Status levels for summary information • Device operating point information level • Save output solutions • Advanced convergence parameters	Defining Simulation Parameters
Output	Selectively save simulation data to a dataset.	*Selectively Saving and Controlling Simulation Data* (cktsim).
Display	Control the visibility of simulation parameters on the schematic.	*Displaying Simulation Parameters on the Schematic* (cktsim).
	Additional parameters that you may find useful.	Additional Parameters

Setting Up a Sweep

Setting up the sweep portion of the simulation consists of three basic parts:

- Identifying the parameter you want to sweep.
- Selecting the sweep type and setting the associated characteristics.
- Optionally, specifying a sweep plan.

To shorten simulation time in any parameter sweep, select a start point where convergence is easy, and vary the parameter gradually. When selecting a start point, if you find that convergence is easier at one end of a sweep and harder at the other, use the easy end as the start of the sweep. Varying the parameter gradually yields better estimates for the next simulation, and achieves convergence more rapidly than if the parameter is changed abruptly. However, too small a step can lead to a prohibitively large number of sweep points, so be careful. For additional information about parameter sweep techniques, see About Preparing Sweeps.

The following table describes the parameter details. Names listed in the *Parameter Name* column are used in netlists and on schematics.

DC Simulation Sweep Parameters

Setup Dialog Name	Parameter Name	Description
Parameter to sweep	SweepVar	The name of any defined parameter to be swept.
Parameter sweep - The sweep type and parameters. Disable *Use sweep plan* to set these parameters.		
Sweep Type		
Single point	Pt	Enables simulation at a specific value for the parameter. Specify the desired value in the *Parameter* field.
Linear		Enables sweeping a range of values based on a linear increment. Click *Start/Stop* to set start and stop values for the sweep, or *Center/Span* to set the center value and a span of the sweep.
Log		Enables sweeping a range of values based on a logarithmic increment. Click *Start/Stop* to set start and stop values for the sweep, or *Center/Span* to set the center value and the span of the sweep.
Start/Stop Start, Stop, Step-size, Pts/decade, Num. of pts.	Start Stop Step Dec Lin	Select the *Start/Stop* option to sweep based on start, stop, step-size and number of points. *Step-size* is *Pts./decade* for a Log sweep. - Start-the start point of a sweep - Stop-the stop point of a sweep - Step-size-the increments at which the sweep is conducted - Pts./decade-number of points per decade - Num. of pts.-the number of points over which sweep is conducted
Center/Span Center, Span, Step-size, Pts./decade, Num. of pts.	Center Span Step Dec Lin	Select the *Center/Span* option to sweep based on center and span, points per decade and number of points. *Pts./decade* is *Step-size* for a Linear sweep. - Center-the center point of a sweep - Span-the span of a sweep - Step-size-the increments at which the sweep is conducted - Pts./decade-number of points per decade - Num. of pts.-the number of points over which sweep is conducted
Note: Changes to any of the Start, Stop, etc. fields causes the remaining fields to be recalculated automatically.		
Use sweep plan	SweepPlan	Enables use of an existing sweep plan component (*SweepPlan*). Select this option and enter the name of the plan or select it from the drop-down list.

About Preparing Sweeps

In ADS, the DC simulation setup dialog enables you to sweep a parameter. You can sweep a parameter using the internal sweep or use the ParamSweep component for an external sweep. The internal parameter sweep is recommended over an external parameter sweep when only one parameter is swept. To improve the convergence process during such an internally swept DC bias simulation, the simulator uses the results from the previous sweep point as an initial guess for the next sweep point and employs a sophisticated arc-length algorithm. In many cases, this makes for a faster and more robust simulation than one done with an external ParamSweep component. The latter simulation simply repeats the DC simulation at each sweep point, with no arc-length continuation. When multiple parameters are swept, the internal parameter sweep can be employed for one parameter and an external parameter sweep must be used for the remainder of the parameters.

The results of the swept DC bias simulation are saved in a binary form in a temporary file. This file is used to post-process selected measurements. Because this file contains *all* of the results, new measurements can be specified and displayed without the need for a subsequent simulation.

Defining Simulation Parameters

Defining the simulation parameters consists of three basic parts:

- Specifying the desired level of detail in the simulation status summary.
- Specifying the amount of device operating-point information to save in ADS.
- Optionally, choosing to save all solutions to the dataset.

> **Note**
> Advanced simulation parameters are accessible with this group. However, as a result of the improvements made to the DC simulation algorithm, it is extremely unlikely that the default values need to be modified. *You are strongly encouraged to leave the advanced parameters set to their default values*. If you encounter a circuit for which a DC analysis does not converge using the default values, or you find it necessary to change the value of any of these parameters, please contact Agilent EEsof Technical Support. See Defining Advanced Simulation Parameters for details about these parameters.

> **Caution**
> In ADS, simulator parameters saved in ADS design files in previous releases are supported in later releases. The advanced simulation parameters saved prior to and opened in ADS 2005A are recognized and populated in the simulation setup dialog box. However, due to the improvement in robustness and speed of the default DC simulation algorithm the user-defined values are disabled, and factory-defined default values are used. *Changing these default values is not recommended*. However, if you find it necessary to restore the original user-defined values, you must manually enable *Advanced Settings* to restore them.

The following table describes the parameter details. Names listed in the *Parameter Name* column are used in netlists and on schematics.

DC Simulation Parameters

Setup Dialog Name	**Parameter Name**	**Description**
Levels - Enables you to set the level of detail in the simulation status report.		
Status level	StatusLevel	Prints simulation information in the Status/Summary part of the Message Window.- 0 reports little or no information, depending on the simulation engine. - 1 and 2 yield more detail. - Use 3 and 4 sparingly since they increase process size and simulation times considerably. The type of information printed may include the sum of the current errors at each circuit node, whether convergence is achieved, resource usage, and where the dataset is saved. The amount and type of information depends on the status level value and the type of simulation.
Device operating point level	DevOpPtLevel	Options to save device operating-point information for most active devices and some linear devices in the circuit to the dataset. If this simulation performs more than one DC analysis (from multiple DC controllers), the device operating point data for all DC analyses will be saved, not just the last one. Default setting is None.
None	None (0)	No information is saved.
Brief	Brief (2)	Saves device currents, power, and some linearized device parameters.
Detailed	Detailed (4)	Saves the DC operating point values which include the device's currents, power, voltages, and linearized device parameters.
Output solutions		
Output solutions at all steps	OutputAllSolns	Instructs the simulator to save all solutions in the dataset. When the simulator is required to use points between steps in order to converge, the resulting information is stored and can subsequently be used for more detailed analysis.
Advanced		Click Advanced on the Parameters tab, and enable Advanced Settings to set these parameters.For parameter descriptions, see the following section, *Defining Advanced Simulation Parameters*.

Defining Advanced Simulation Parameters

The stand-alone DC simulator's sole role is to do a DC analysis. All other simulators such as AC, S-parameter, transient, harmonic balance, and circuit envelope do an initial DC analysis as their first step. The advanced simulation parameters are used for controlling a stand-alone DC simulation. For information about setting DC convergence parameters for these analyses, see *Setting Convergence Options* (cktsim).

The robustness and speed of the default DC analysis algorithm was significantly improved in ADS 2005A. All DC analyses with factory-default settings are expected to converge to the correct solution with near-optimal speed. This means that it is extremely unlikely that any of the following advanced simulation parameters must be altered:

ConvMode	*ArcMaxStep*
MaxDeltaV	*ArcLevelMaxStep*
MaxIters	*ArcMinValue*
MaxStepRatio	*ArcMaxValue*
MaxShrinkage	*LimitingMode*

The following table describes the parameter details. Names listed in the *Parameter Name* column are used in netlists and on schematics.

DC Simulation Advanced Settings Parameters

Setup Dialog Name	Parameter Name	Description
Advanced Settings		Click Advanced Settings to set these parameters.
Max. Delta V	MaxDeltaV	Maximum change in node voltage per iteration. If no value is specified, the default value is four times the thermal voltage, or approximately 0.1 V.
Max. Iterations	MaxIters	Maximum number of iterations to be performed. The simulation will iterate until it converges, an error occurs, or this limit is reached.
Mode	ConvMode	Offers a choice between different convergence algorithms. Note that the convergence algorithm selected here applies to this DC simulation only. You can choose from these same convergence algorithms using the simulator options to apply them to all analyses performed for this design.
Auto sequence	0	Default convergence mode. Cycles through various algorithms and parameter values and has been optimized for both robustness and speed. Should converge for all circuits, and is therefore strongly recommended over all other convergence modes.
Newton-Raphson	3	Iterative process that terminates when the sum of the currents into each node equals zero at each node, and the node voltages converge. Used by other convergence modes.
Forward source-level sweep	4	Sets all DC sources to zero and then gradually sweeps them to their full values. The source steps are determined via homotopy/continuation methods.
Rshunt sweep	5	Inserts a small resistor from each node to ground and then sweeps this value to infinity.
Reverse source-level sweep	6	Rarely used, but available for those few cases where it is necessary. Similar to Forward source-level sweep, except in the reverse direction. Use Reverse source-level sweep when Forward source-level sweep returns an "out of bounds" error. This error indicates that there is a negative resistance in the circuit when all the DC sources are zero. This is a rare situation but can occur with ideal models of oscillators, such as those described by the van der Pol equation.
Hybrid solver	7	Combination of various algorithms. Starts with Forward source-level sweep with the source steps determined via heuristics. If this fails, Forward source-level sweep with the source steps determined via homotopy/ continuation methods is attempted. If this fails, Reverse source-level sweep with the source steps determined via homotopy/continuation methods is attempted. If this fails, Rshunt sweep

		is attempted. If this fails, Gmin relaxation, where a 1 Mohm resistor is inserted from each node to ground and then swept to infinity, is attempted.
Pseudo transient	8	Variant of the source stepping algorithm. Performs a transient simulation on a pseudo circuit derived from the original circuit. The transition from the zero solution to the final solution is of no interest in this analysis, so the truncation error is ignored and the timestep is taken as large as possible. After this pseudo transient analysis, a Newton-Raphson analysis is performed with the pseudo transient solution as the initial guess. If this fails, a Newton-Raphson analysis with Gmins of 1e-12 Siemens inserted from each node to ground is attempted. If this succeeds, the Gmins are removed and a Newton-Raphson analysis with the Gmin solution as the initial guess is attempted.
Arc Max Step	ArcMaxStep	Limits the maximum size of the arc-length step during arc-length continuation. During arc-length continuation, the arc-length is increased in steps. The step size is calculated automatically for each problem. However, if ArcMaxStep is specified and is nonzero, it will define an upper limit for the size of the arc-length step. The default is 0, meaning there is no upper limit for the arc-length step.
Arc Level Max Step	ArcLevelMaxStep	Limits the maximum arc-length step size for source-level continuation. The default is 0 which means there is no limit for the arc-length step.
Arc Min Value	ArcMinValue	Set relative to ArcMaxValue. ArcMinValue determines the lower limit that is allowed for the continuation parameter *p* during the simulation. During arc-length continuation, *p* can trace a complicated manifold and its value can vary non-monotonically. ArcMinValue specifies a lower bound for *p* such that if during the arc-length continuation *p* becomes smaller than ArcMinValue, the simulation is considered to have failed to converge. The default is p_{min} - *delta*, where *delta* is p_{max} - p_{min}, where p_{min} is the lower end of the parameter sweep and p_{max} is the upper end of the parameter sweep.
Arc Max Value	ArcMaxValue	Set relative to ArcMinValue. ArcMaxValue determines the upper limit that is allowed for the continuation parameter *p* during the simulation. During arc-length continuation, *p* can trace a complicated manifold and its value can vary non-monotonically. ArcMaxValue specifies an upper bound for *p* such that if during the arc-length continuation *p* becomes greater than ArcMaxValue, the simulation is considered to have failed to converge. The default is p_{max} + *delta*, where *delta* is p_{max} - p_{min}, where p_{min} is the lower end of the parameter sweep and p_{max} is the upper end of the parameter sweep.
Max Step Ratio	MaxStepRatio	Controls the maximum number of continuation steps. The default is 100.
Max Shrinkage	MaxShrinkage	Controls the minimum size of the arc-length step. The default is 1e-5.
Limiting Mode	LimitingMode	Sets the type of limiting done on the changes of nodes at each iteration.
Global Element Compression	0	Limits the changes at the nonlinear nodes with a log function at each iteration when the changes exceed the internally determined value.
Global Device-based Limiting	1	Performs limiting on the changes at each iteration for the nonlinear components.
Dynamic Element Compression	2	Limits the changes at the nonlinear nodes with a log function at each iteration when the changes exceed the internally determined value.

Dynamic Vector Compression	3	Limits the changes of all nodes with a log function at each iteration when the changes exceed the internally determined value.
Global Vector Compression	4	Limits the changes of all nodes with a log function at each iteration when the changes exceed the internally determined value.
Global Vector Scaling	5	Scales the changes of the node at each iteration with internally determined scale factor.
No Limiting	6	No limiting will be done on the changes of all the nodes at each iteration.

Additional Parameters

The following table includes additional parameters that you may find useful.

Additional Parameters

Setup Dialog Name	**Parameter Name**	**Description**
Optimization		
Compute Sensitivities Using Finite Difference	UseFiniteDiff	Perform sensitivity analysis optimization using the Finite Difference approximation method. This requires N+1 circuit simulations, where N is the number of optimization variables.To set this parameter in ADS, select the parameter name on the Display tab, then enter the value directly on the schematic. Allowed values: *yes* and *no*.
Other		Use *Other* to enable access to hidden parameters, and assign values to them. The format is `Other=HiddenParameter1=value1 HiddenParameter2=value2...` Hidden parameters typically are used when troubleshooting convergence problems.To set this parameter in ADS, select the parameter name on the Display tab, then enter the value directly on the schematic.
Restart	Restart	Use *Restart* to control how an External DC Sweep picks an initial guess. Allowed values are *1* or *0*. Restart=1 (the default value) instructs the External DC Sweep to treat each sweep point as an independent DC simulation. Restart=0 instructs the External DC Sweep to use the solution at each point as an initial guess for the next point. To perform an External DC Sweep, use a Parameter Sweep component. *Restart* has no effect on internal DC sweep (which is set from the Sweep tab of the DC simulation controller). Internal DC sweep is an arc-length continuation and by definition always uses the solution at the current point as an initial guess for the next simulation point.

Theory of Operation for DC Simulation

This section discusses the theory of operation for DC simulation. It can help you understand the underlying technology. Typical users should not need to examine this material.

Simulation Basics

The simulators compute the response of a given circuit to a particular stimulus by converting, based on certain assumptions, a system of nonlinear ordinary differential circuit equations into a system of nonlinear algebraic equations and then solving them numerically. The various simulators convert ordinary differential equations to algebraic equations differently and use different numerical techniques for solving the resulting algebraic equations, leading to the many different simulator flavors (DC, AC, S-parameter, transient, harmonic balance, circuit envelope). For example, the DC and the harmonic balance simulators treat the d/dt operator differently, leading to different algebraic equations. The numerical simulation techniques rely on various iterative processes to achieve mathematical convergence toward an equilibrium point in the nonlinear algebraic equations that describe the circuit. Once this equilibrium point is reached to within certain tolerances, a solution is said to have been found.

The specific assumptions for the DC simulator are described in Simulation Assumptions.

Simulation Assumptions

DC voltages and currents are signals of zero frequency. The simulator uses this concept when performing a DC simulation, and the following conditions apply:

- Independent sources are constant valued.
- Linear elements are replaced by their (real) conductances at zero frequency.
- Capacitors, microstrip gaps, AC coupled lines, and similar items are treated as open circuits.
- Inductors, conductive discontinuities, and similar items are treated as short circuits.
- Time-derivatives are constant (zero).
- Transmission lines are replaced by DC conductance values calculated from their length, cross-sectional area, and conductivity.
- Scattering parameter (S-parameter) files must include zero frequency data to operate properly at DC (this is also required for harmonic balance analysis). Otherwise, the simulator extrapolates each S-parameter for the zero-frequency case, and uses the real part as the DC response.
- The simulator has built-in safeguards against nodes that are DC-isolated (that have no DC path to ground), as well as against DC source-inductor loops. Nevertheless, try to avoid these conditions.

For More Details

The descriptions of the various convergence modes in the table *DC Simulation Advanced Settings Parameters* (cktsimdc) mention techniques such as the Newton-Raphson algorithm, source-level sweeping, arc-length continuation, Gmin relaxation, and pseudo-transient analysis. For more details about these and related techniques, see the publications listed in *References for DC Simulation* (cktsimdc).

Troubleshooting a DC Simulation

This section presents suggestions for using this simulator and improving the accuracy of results.

The robustness and speed of the default DC analysis algorithm has been significantly improved in ADS 2005A. All DC analyses with factory-default settings are expected to converge to the correct solution with near-optimal speed. This means that it is extremely unlikely that any of the following advanced simulation parameters must be altered:

ConvMode	*ArcMaxStep*
MaxDeltaV	*ArcLevelMaxStep*
MaxIters	*ArcMinValue*
MaxStepRatio	*ArcMaxValue*
MaxShrinkage	*LimitingMode*

Note
As a result of the improvements made to the DC simulation algorithm, it is extremely unlikely that the factory-default values for the advanced simulation parameters need to be modified. *You are strongly encouraged to leave these parameters set to their default values*. If you encounter a circuit for which a DC analysis does not converge using the default values, or you find it necessary to change the value of any of these parameters, please contact Agilent EEsof Technical Support. See *Defining Advanced Simulation Parameters* (cktsimdc) for details about these parameters.

Use the following checklist to locate and fix problems. For details about specific problems, see the sections that follow.

- Use the simulator Options to turn on *Issue warnings* (*GiveAllWarnings=yes*) and increase *Maximum number of warnings* (*MaxWarnings=10* by default). To set these parameters in ADS, place the Options component on your schematic from the Simulation-DC palette. For details about simulator options, see *Using the Simulator Options Component* (cktsim).
- Read all error and warning messages from the Simulation/Synthesis Messages window and make corrections accordingly.
- Read all status information from the Status/Summary window (select *Show Complete Status Message* under Window tab) and make corrections accordingly.
- Check your setup for common errors (see Common Setup Errors).
- Loosen the simulation Options parameters for absolute and/or relative simulation tolerances (see Adjusting Tolerance Parameters).
- Turn on/off the Topology Checker (on by default) using the simulation Options parameters (see Topology Check and Correction).
- Check if you have created a circuit with no solution (see Impossible Circuits).
- Check if Imax should be increased (see Model Parameter Imax is Too Small).

Common Setup Errors

If your circuit does not converge, or if it converges but your DC I-V curves appear different than expected, it can be due to one or more of the following reasons:

- Check component and model parameter values and units, especially small series resistor values in model parameters.
- Check if your independent swept parameters are swapped on the grid display. For example, if SRC1 sweeps a transistor gate voltage and SRC2 sweeps a drain voltage, and if the current is displayed with SRC1 (gate voltage) on the *x*-axis, these traces will not look like typical DC I-V traces. Exchange the order of the swept variables SRC1 and SRC2 on the grid.
- Check if you are sweeping one source with large variations and another with small variations. For example, say that you are sweeping the base current and the collector voltage of a transistor by connecting a DC current source at the base and a DC voltage source at the collector. Furthermore, say that you are using SRC1 to sweep the base current and SRC2 to sweep the collector voltage. A DC bias-current sweep typically involves very small increments – on the order of microamperes, for example. Inside the program, the SRC1 sweep always occurs inside the SRC2 sweep, so the net effect is that the base current is changing more frequently. Because of the small increments in the current sweep, proceeding from one current value to the next may not change the circuit state (operating point) in the simulator. This causes the program to converge to the same solution as the previous current value, resulting in jagged I-V curves.
- Check if you are measuring the DC current in the wrong direction. By default, positive current through a two-terminal element, such as a resistor or ammeter, is measured as flowing from pin 1 to pin 2. To measure current in the opposite direction, you can edit the schematic by changing the element's orientation, or specify the current at a different pin, or change the orientation of the current probe.
- If you are having convergence problems during a swept DC simulation, reduce the step size of the parameter being swept.

Adjusting Tolerance Parameters

Several tolerance parameters can be changed so that simulations are more likely to converge, although at the cost of a somewhat less accurate result. These parameters include:

- Current relative tolerance (*I_RelTol*)
- Voltage relative tolerance (*V_RelTol*)
- Current absolute tolerance (*I_AbsTol*)
- Voltage absolute tolerance (*V_AbsTol*)

The equations that are used to check for convergence apply to all analysis types, not just DC. In ADS, you can change these parameters in the *Options* component on the Convergence tab.

Topology Check and Correction

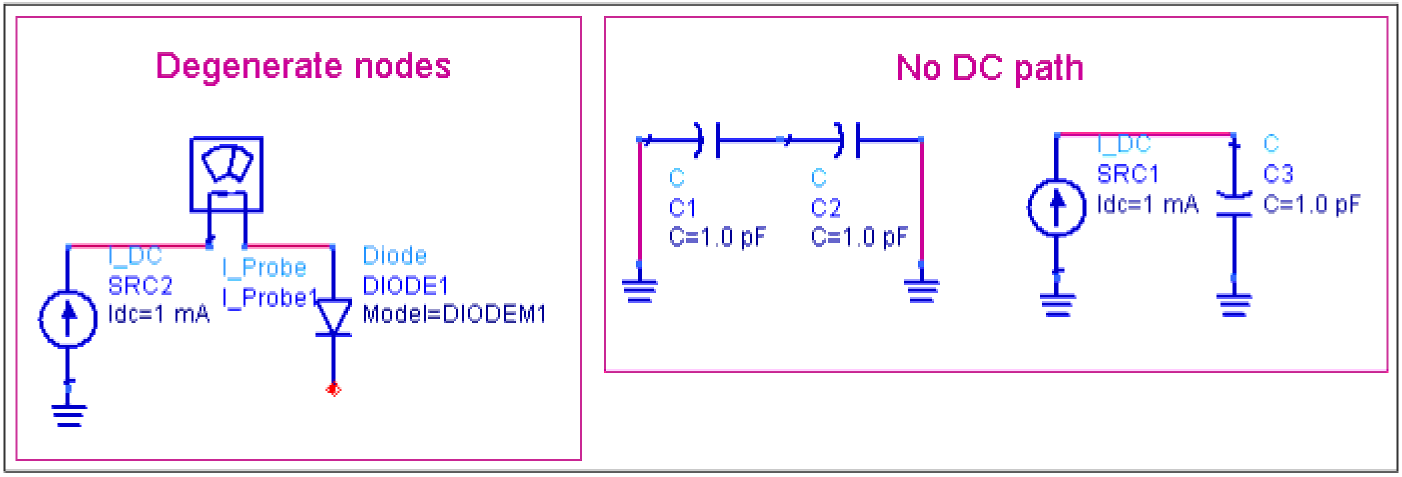
By default, a topology check is performed before any simulation is executed. The topology check finds common circuit errors that would lead to more serious errors later in the DC simulation. These errors are reported in the Simulation/Synthesis Messages window. See Setting the Topology Checker Mode to learn how to turn the topology checker on and off, and to change the error message format. However, it is recommended to keep the topology checker turned on.

The topology check verifies the existence of the following conditions, some of which are corrected automatically by the simulator if the topology checker is turned on:

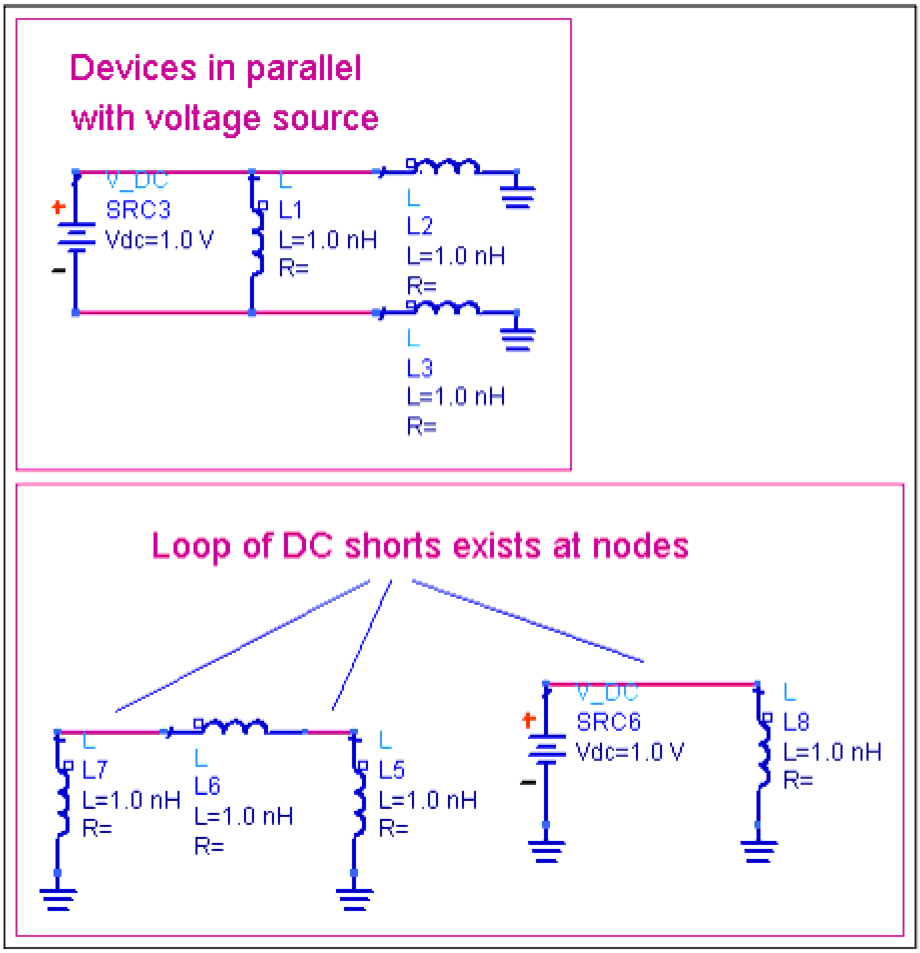
- There are no degenerate nodes.
  At least two components must be connected to each node. Degenerate nodes have only one device attached to them, making it difficult for the simulator to solve the circuit. If the topology checker is on, the simulator automatically corrects this problem by inserting a large resistor with a default value of 1e12 ohms from the offending node to ground. The value of the resistor is controlled by the GMIN parameter in the Options component.
- There is a DC path from every node to ground.
  If the DC circuit simulation is to succeed, all circuit nodes must have a DC path to ground. This eliminates nodes that float at an undefined DC voltage value and allows the DC simulator to find the bias point of all nodes in the circuit. This error is commonly caused by two capacitors in series. If the topology checker is on, the simulator corrects this problem automatically by inserting a large resistor with a default value of 1e12 ohms from the offending node to ground. This adds a DC path to ground without affecting the circuit's performance. The value of the resistor is controlled by the GMIN parameter in the Options component.
- There are no devices in series with current sources.
  Devices in series with a current source create problems during DC simulation. This should be corrected manually.
- There are no loops of DC shorts, or voltage or current sources.
  It is possible to create a circuit containing short-circuited DC loops that cannot be handled by the DC simulation. For instance, if two ideal inductors are connected in parallel, they create a DC short-circuit loop. The DC current in this loop is undefined; it can have any value without affecting the circuit solution. This should be corrected manually.
- There are no devices parallel to voltage sources.
  Devices parallel to a voltage source create problems during DC simulation. This should be corrected manually.

The following figures show examples of the topology conditions checked when the topology checker is turned on. The simulator corrects the errors such as those shown in the first figure below by inserting a large resistor (default = 1e12 ohms) from the offending node to ground. However, the simulator cannot correct the errors such as those shown in the second figure, which you should correct manually. Additionally, the topology checker does not check and correct any of these conditions through transmission lines.

By default a summary of the topological problems found is printed to the Simulation/Synthesis Messages window when the topology checker is turned on. To see a list of all the nodes that have topological problems, set the message format mode to verbose.



Common circuit errors corrected during topology check



Common circuit errors *not* corrected during topology check

If the circuit fails to simulate due to topological problems, turn off the topology checker.

This will eliminate all topology checking, and simulations will proceed directly to the DC simulation stage. The DC simulator will speed up simulations slightly, which may be helpful in rare cases where it is certain that the DC simulation will succeed despite the topology.

> **Note**
> Errors found by the topology checker almost always lead to DC simulation failures, so turning off the topology checker is not recommended.

If you have existing circuits in which there are many topological problems, turning on the topology checker could significantly speed up the simulation. In this case the topology checker corrects the common topological problems before the circuit is simulated.

Setting the Topology Checker Mode

The topology checker parameters are located with the simulator Options. These parameters enable you to turn the topology checker on (default is *TopologyCheck=yes*) and turn it off (*TopologyCheck=no*). You can also set the warning messages format to summary (default is *TopologyCheckMessages=summary*) or verbose (*TopologyCheckMessages=verbose*).

To change these settings in dialog boxes place an Options component on your schematic, open its setup dialog box, then see the Misc tab. To turn on the topology checker, enable *Perform topology check and correction*. Disable the setting to turn it off. To set the message format, choose *Summary* or *Verbose* for *Format topology check warning messages*.

Checking Nodes and Pins

If the topology check fails, the offending nodes are highlighted in the Simulation/Synthesis Messages window. Choose *Simulate > Highlight Node* to identify these nodes.

Unconnected Wire Errors

In the case of a circuit with a wire that is missing, a message such as the following can appear in the Simulation/Synthesis Messages window during a simulation:

```
Warning detected by HPEESOFSIM during DC analysis "DC1".
Circuit as given has no unique solution.
A virtual resistance of 1 TOhms was added to each node.
```

In the case of unconnected wires or pins, choose *Tools > Check Representation*, then select *Open connections* and click *OK*. All unconnected wires and components are highlighted, making them easily visible on the screen. Connect these to the correct circuit node.

Port/Pin Mismatches

To check for mismatches between ports and named pins on an underlying schematic, use the Check Representation feature. Choose *Tools* > *Check Representation*, then select *Port/Pin mismatches* and click *OK*. In a circuit without a topology problem, the simulator reports the following:

```
Unconnected pins: 0
Port/Pin mismatches: 0
```

Where a wire is missing between two pins, for example, the components and the disconnected pins will be highlighted in red on the schematic, and a message such as the following will appear, indicating the coordinates of the connectors:

```
Unconnected pins: 2
I_DC SRC1, pin 2 (2.500,-0.125)
BJT_NPN BJT1, pin 2 (3.250,0.375)
```

To clear the highlighting, choose *View* > *Clear Highlighting*.

> **Note**
> Always ensure that there is no duplication of port numbers.

Impossible Circuits

A circuit can fail to have a solution. Examples of such impossible circuits are circuits with two DC current sources (or, perhaps, current-mirror circuits) in series. The simulator attempts to analyze these circuits by adding a large resistor in parallel with every node in the circuit. If the DC simulation then succeeds, the resistor value is increased as high as possible (up to 1 Tohm). This allows the DC simulation to proceed, and usually to succeed. However, the offending circuit node or nodes often cause problems with a subsequent harmonic balance simulation because the harmonic balance simulator does not add these resistors. It is best to find the problem node and manually add a DC path (such as a large resistor) to ground.

If virtual resistors were added to each node during DC simulation but the simulation still failed, look for circuit problems such as loops of inductors and voltage sources.

Model Parameter Imax is Too Small

If the RHS_NORM oscillates between/among iterations, the reason could be that the diode's p-n junction in this circuit conducts more than 1 A. The default values of *Imax* for nonlinear semiconductor devices are around 1 A. It is possible that all device's p-n junction currents in a circuit do not exceed *Imax* after the circuit converges but have problems converging during the iteration process because of the small *Imax*. The remedy is to increase nonlinear device model parameter *Imax* to 1e3. For BSIM3, the model parameter is *Ijth*.

References for DC Simulation

1. K.S. Kundert, "The Designer's Guide to SPICE and Spectre," Kluwer Academic Publishers, Boston, MA, USA, 1995.
2. K.S. Kundert, J.K. White, and A. Sangiovanni-Vincentelli, "Steady-State Method for Simulating Analog and Microwave Circuits," Kluwer Academic Publishers, Boston, MA, USA, 1990.
3. L.W. Nagel, "SPICE2: A Computer Program to Simulate Semiconductor Circuits," Ph.D. Dissertation, Electronics Research Laboratory, College of Engineering, University of California, Berkeley, CA, USA, 1975.