



**February 2011**  
**Gain Compression Simulation**

© **Agilent Technologies, Inc. 2000-2011**

5301 Stevens Creek Blvd., Santa Clara, CA 95052 USA

No part of this documentation may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Agilent Technologies, Inc. as governed by United States and international copyright laws.

**Acknowledgments**

Mentor Graphics is a trademark of Mentor Graphics Corporation in the U.S. and other countries. Mentor products and processes are registered trademarks of Mentor Graphics Corporation. \* Calibre is a trademark of Mentor Graphics Corporation in the US and other countries. "Microsoft®, Windows®, MS Windows®, Windows NT®, Windows 2000® and Windows Internet Explorer® are U.S. registered trademarks of Microsoft Corporation. Pentium® is a U.S. registered trademark of Intel Corporation. PostScript® and Acrobat® are trademarks of Adobe Systems Incorporated. UNIX® is a registered trademark of the Open Group. Oracle and Java and registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners. SystemC® is a registered trademark of Open SystemC Initiative, Inc. in the United States and other countries and is used with permission. MATLAB® is a U.S. registered trademark of The Math Works, Inc.. HiSIM2 source code, and all copyrights, trade secrets or other intellectual property rights in and to the source code in its entirety, is owned by Hiroshima University and STARC. FLEXIm is a trademark of Globetrotter Software, Incorporated. Layout Boolean Engine by Klaas Holwerda, v1.7 <http://www.xs4all.nl/~kholwerd/bool.html> . FreeType Project, Copyright (c) 1996-1999 by David Turner, Robert Wilhelm, and Werner Lemberg. QuestAgent search engine (c) 2000-2002, JObjects. Motif is a trademark of the Open Software Foundation. Netscape is a trademark of Netscape Communications Corporation. Netscape Portable Runtime (NSPR), Copyright (c) 1998-2003 The Mozilla Organization. A copy of the Mozilla Public License is at <http://www.mozilla.org/MPL/> . FFTW, The Fastest Fourier Transform in the West, Copyright (c) 1997-1999 Massachusetts Institute of Technology. All rights reserved.

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

"This program includes Metis 4.0, Copyright © 1998, Regents of the University of Minnesota", <http://www.cs.umn.edu/~metis> , METIS was written by George Karypis (karypis@cs.umn.edu).

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

SuperLU\_MT version 2.0 - Copyright © 2003, The Regents of the University of California, through Lawrence Berkeley National Laboratory (subject to receipt of any required approvals from U.S. Dept. of Energy). All rights reserved. SuperLU Disclaimer: THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS

INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

7-zip - 7-Zip Copyright: Copyright (C) 1999-2009 Igor Pavlov. Licenses for files are: 7z.dll: GNU LGPL + unRAR restriction, All other files: GNU LGPL. 7-zip License: This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA. unRAR copyright: The decompression engine for RAR archives was developed using source code of unRAR program. All copyrights to original unRAR code are owned by Alexander Roshal. unRAR License: The unRAR sources cannot be used to re-create the RAR compression algorithm, which is proprietary. Distribution of modified unRAR sources in separate form or as a part of other software is permitted, provided that it is clearly stated in the documentation and source comments that the code may not be used to develop a RAR (WinRAR) compatible archiver. 7-zip Availability: <http://www.7-zip.org/>

AMD Version 2.2 - AMD Notice: The AMD code was modified. Used by permission. AMD copyright: AMD Version 2.2, Copyright © 2007 by Timothy A. Davis, Patrick R. Amestoy, and Iain S. Duff. All Rights Reserved. AMD License: Your use or distribution of AMD or any modified version of AMD implies that you agree to this License. This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Permission is hereby granted to use or copy this program under the terms of the GNU LGPL, provided that the Copyright, this License, and the Availability of the original version is retained on all copies. User documentation of any code that uses this code or any modified version of this code must cite the Copyright, this License, the Availability note, and "Used by permission." Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included. AMD Availability: <http://www.cise.ufl.edu/research/sparse/amd>

UMFPACK 5.0.2 - UMFPACK Notice: The UMFPACK code was modified. Used by permission. UMFPACK Copyright: UMFPACK Copyright © 1995-2006 by Timothy A. Davis. All Rights Reserved. UMFPACK License: Your use or distribution of UMFPACK or any modified version of UMFPACK implies that you agree to this License. This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at

your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Permission is hereby granted to use or copy this program under the terms of the GNU LGPL, provided that the Copyright, this License, and the Availability of the original version is retained on all copies. User documentation of any code that uses this code or any modified version of this code must cite the Copyright, this License, the Availability note, and "Used by permission." Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included. UMFPACK Availability: <http://www.cise.ufl.edu/research/sparse/umfpack> UMFPACK (including versions 2.2.1 and earlier, in FORTRAN) is available at <http://www.cise.ufl.edu/research/sparse> . MA38 is available in the Harwell Subroutine Library. This version of UMFPACK includes a modified form of COLAMD Version 2.0, originally released on Jan. 31, 2000, also available at <http://www.cise.ufl.edu/research/sparse> . COLAMD V2.0 is also incorporated as a built-in function in MATLAB version 6.1, by The MathWorks, Inc. <http://www.mathworks.com> . COLAMD V1.0 appears as a column-preordering in SuperLU (SuperLU is available at <http://www.netlib.org> ). UMFPACK v4.0 is a built-in routine in MATLAB 6.5. UMFPACK v4.3 is a built-in routine in MATLAB 7.1.

Qt Version 4.6.3 - Qt Notice: The Qt code was modified. Used by permission. Qt copyright: Qt Version 4.6.3, Copyright (c) 2010 by Nokia Corporation. All Rights Reserved. Qt License: Your use or distribution of Qt or any modified version of Qt implies that you agree to this License. This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Permission is hereby granted to use or copy this program under the terms of the GNU LGPL, provided that the Copyright, this License, and the Availability of the original version is retained on all copies. User documentation of any code that uses this code or any modified version of this code must cite the Copyright, this License, the Availability note, and "Used by permission." Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included. Qt Availability: <http://www.qtsoftware.com/downloads> Patches Applied to Qt can be found in the installation at: `$HPEESOF_DIR/prod/licenses/thirdparty/qt/patches`. You may also contact Brian Buchanan at Agilent Inc. at [brian\\_buchanan@agilent.com](mailto:brian_buchanan@agilent.com) for more information.

The HiSIM\_HV source code, and all copyrights, trade secrets or other intellectual property rights in and to the source code, is owned by Hiroshima University and/or STARC.

**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 Gain Compression Simulation . . . . .	7
Performing a Gain Compression Simulation . . . . .	8
Example of Gain Compression Simulation (ADS only) . . . . .	9
Gain Compression Concept . . . . .	13
XDB Simulation Parameters . . . . .	14
Freq . . . . .	14
XdB . . . . .	15
Defining Simulation Parameters . . . . .	16
Solver . . . . .	18

# About Gain Compression Simulation

To calculate the X dB gain compression point of an amplifier or mixer, use the XDB simulation component, from the Simulation-XDB palette. The simulator sweeps the input power upward from a small value, and when the required amount of gain compression is seen at the output, the analysis is complete. A common calculation is the 1-dB gain-compression point; this is the default.

Refer to the following topics for details of gain compression simulation:

- *Performing a Gain Compression Simulation* (cktsimgain) describes the minimum setup requirements for a gain-compression simulation.
- *Example of Gain Compression Simulation (ADS only)* (cktsimgain) is a detailed setup for calculating gain compression, using a power amplifier as the example.
- *Gain Compression Concept* (cktsimgain) is a brief explanation of the gain compression calculation, plus information on how it compares to a harmonic balance simulation.
- *XDB Simulation Parameters* (cktsimgain) provides details on the tabs and fields in the XDB simulation controller.



**Note**

You must have the Gain Compression license to run the simulator. You can build the examples in this section without the license, but you will not be able to simulate them.

# Performing a Gain Compression Simulation

Start by creating your design, then add current probes and identify the nodes from which you want to collect data

For a successful analysis:

- Use port-type sources at the inputs, such as the P\_1Tone under *Sources-Freq Domain*.
- Terminate outputs with port-impedance terminations. You can find this type of port under *Simulation-S\_Param*. Verify the impedance value.
- The *Num* values in the port-impedance terminations are used to specify input and output ports.
- Add the Gain Compression control element to the schematic. Fill in the fields under the Freq and XdB tabs:
  - For Freq, set the fundamental frequency and order
  - For X db, specify the input and output ports, the frequency at each port, power variation for each port, and maximum input power.
- For a faster simulation of large circuits, use the Krylov option. Refer to the topic "Selecting a Solver" in the section "Harmonic Balance Basics" in the *Harmonic Balance Simulation* documentation for instructions on how to use this option.
- You can use previous simulation solutions to speed up the simulation process. For more information, refer to the topic "Reusing Simulation Solutions" in the section "Harmonic Balance Basics" in the *Harmonic Balance Simulation* documentation.

For details about each field, click *Help* from the dialog box.

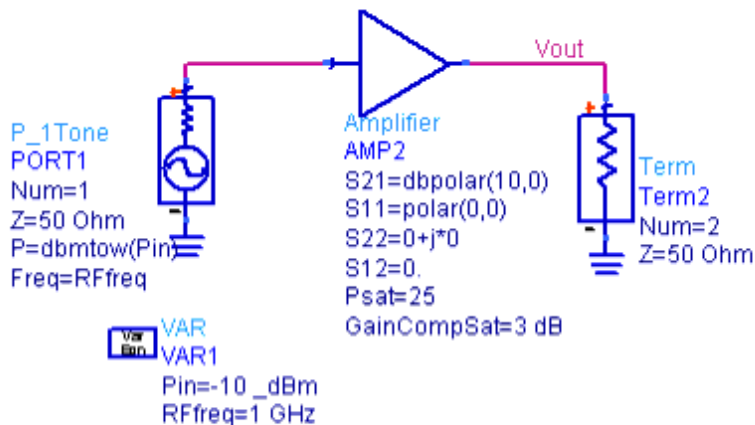


# Example of Gain Compression Simulation (ADS only)

*Gain-compression simulation example* (cktsimgain) illustrates the setup for a basic gain compression simulation of a power amplifier circuit. The design includes a harmonic balance controller as a comparison.

## Note

This design, *XDB1*, is in the directory *examples / Tutorial/SimModels\_wrk*. The simulation results are in *XDB1.dds*.



## GAIN COMPRESSION

```

XDB
HB2
Freq[1]=RFfreq
Order[1]=5
GC_XdB=1
GC_InputPort=1
GC_OutputPort=2
GC_InputFreq=1.0 GHz
GC_OutputFreq=1.0 GHz
GC_InputPowerTol=1 e-3
GC_OutputPowerTol=1 e-3
GC_MaxInputPower=100
  
```

## HARMONIC BALANCE

```

HarmonicBalance
HB1
Freq[1]=RFfreq
Order[1]=5
SweepVar="Pin"
SweepPlan="Plan1"
  
```

## SWEEP PLAN

```

SweepPlan
Plan1
Start=-10.0 Stop=15.0 Step=1.0 Lin=
UseSweepPlan=
SweepPlan=
Reverse=no
  
```

### Gain-compression simulation example

To perform a gain-compression simulation:

1. From the **Sources-Freq Domain** palette, select a port-type source such as a `_P_1Tone_` or `_P_nTone_`. Place the source at the input of the component or circuit under test, then define input power and edit other parameters as required.
2. From the **Simulation-S\_Param** palette, select a **Term** component and place it at the output of the component or circuit under test. Edit the impedance if necessary.
3. Ensure that the input port source is identified as Num=1, and the output port

termination as Num=2. Also ensure that you have labeled the inputs and outputs of nodes at which you want data to be reported.

4. From the **Simulation-XDB** palette list, select an **XDB** simulation component, and place it on the schematic, then edit it.
5. Select the **Freq** tab, and edit frequencies, orders, and fundamentals as required.

**Hint**

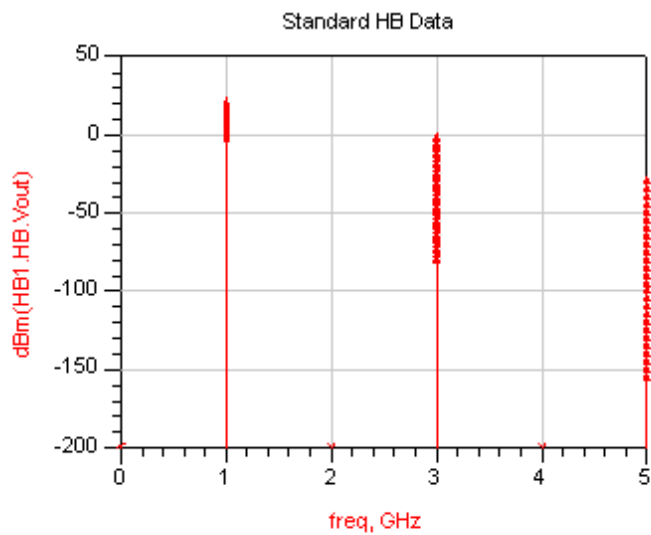
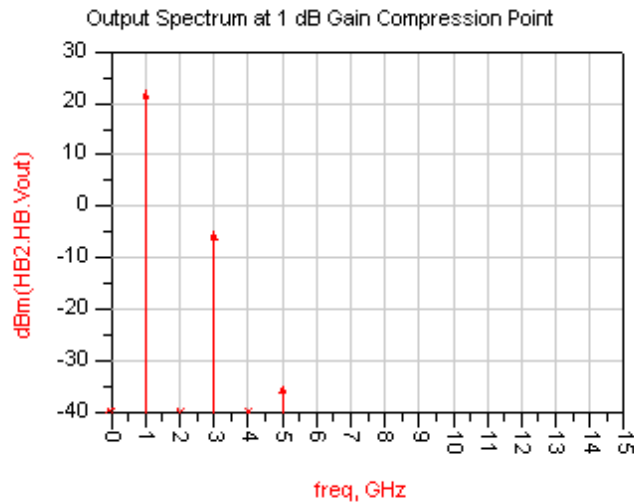
In a one-tone simulation such as this one, *MaxOrder* is irrelevant.

- Ensure that values are established for all of the frequencies of interest (for example, RF, LO, and IF frequencies) in the design under test. (Where mixers are employed, set the LO frequency to Freq[1]; it is easier to achieve convergence if the frequency of the signal with the largest amplitude is assigned to Freq[1].) To add or delete frequencies, select them in the *Select* field, then click *Add* or *Cut* . Display them on the schematic to facilitate editing.
6. Select the **X dB** tab and set the following:
    - In the *Gain compression* field, either accept the default or enter a new gain-compression value. The default is 1 dB.
    - Under *Port numbers* , ensure that appropriate values are entered for the input port source (generally Num=1) and the output termination (generally Num=2).
    - Under *Port frequencies* , set appropriate values for the input and output frequencies. In a circuit employing a mixer, the input frequency would be that of the RF input, the output frequency that of the IF output.
    - Under *Power tolerances* , set appropriate values for acceptable variations in power at the input and output ports. In the *Max. Input Power* field, set the maximum input power over which the simulator is allowed to search.
  7. To reduce memory requirements significantly where many frequencies and orders are involved, select the **Solver** tab, then select **Krylov Solver** .

**Note**

We do not recommend using the Krylov option on small problems. For details, refer to the topic "Selecting a Solver" in "Harmonic Balance Basics" in the *Harmonic Balance Simulation* documentation.

8. From the **Simulation-HB** palette list, select an **HB** simulation component, and place it on the schematic. On the **Freq** tab, set the values for the *Fundamental Frequencies* . On the **Sweep** tab, enter Pin for *Parameter to sweep* . Then, check *Use sweep plan* and enter Plan1.
9. **Simulate** . Gain compression data items may be identified with an HB prefix. The following is a Rectangular plot of the results of the simulation. At the top are the gain compression results, with the standard harmonic balance results at the bottom for comparison.

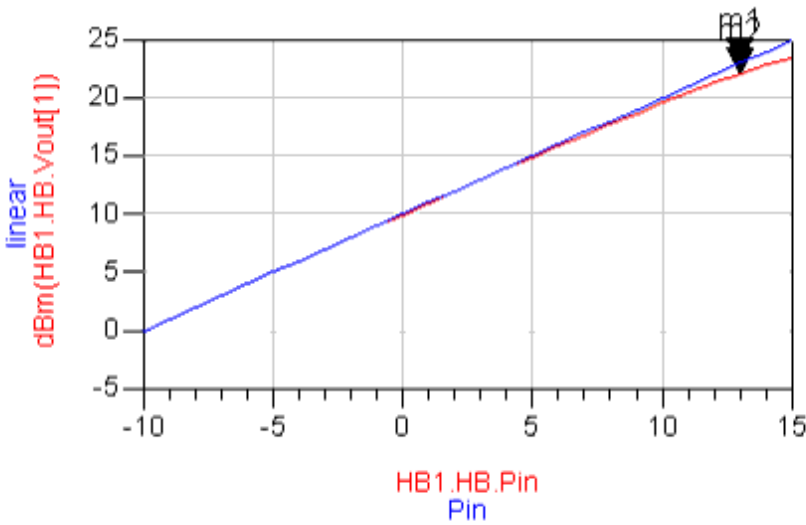


The following graphs and their associated equations illustrate how to extract more meaningful results from the simulation. The equation *Gain* finds gain by subtracting the input power  $P_{in}$  from the output voltage of the fundamental [ $V_{out}[1]$ ]. The equation *linear* adds input power to gain to produce the linear plot in the graph. To find the input power that leads to a compression point of approximately 1 dB, place a marker on each trace and move them so that their difference is as close to 1 dB as the simulation results allow. The simple equation *compression* subtracts the values of the markers and the *compression* List plot shows the equation's results to be a gain compression of approximately 1 dB. The next List plot shows the input power and output power at which the approximately 1 dB compression point occurs.

**Eqn** Gain=dBm(HB1.HB.Vout[1])-HB1.HB.Pin

**Eqn** linear=Gain[0]+HB1.HB.Pin

m1 Pin=13.000 linear=22.996	m2 indep(m2)=13.000 plot_vs(dBm(HB1.HB.Vout[1]), HB1.HB.Pin)=22.098
-----------------------------------	---



**Eqn** compression=m1-m2

compression
0.898

Input and Output Powers at 1 dB Compression Point

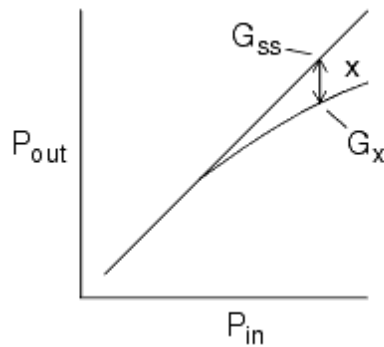
inpwr[1]	outpwr[1]
13.382	22.382

# Gain Compression Concept

Gain compression is the difference, in dB, between a point on an idealized linear (small-signal) power-gain slope and a corresponding point on the actual power curve. That is, for a gain compression of  $x$  dB,

- $x = 10 \log_{10}(G_x/G_{SS})$

where  $G_x$  is the point on the idealized linear gain slope that is  $x$  dB directly below the point  $G_{SS}$  on the small-signal power curve, as illustrated below.



The simulator stops its analysis when it reaches that point. The default setting is 1 dB.

The XDB simulator uses a harmonic balance algorithm, and as such shares many of the parameters and options that the Harmonic Balance simulator provides. However, this simulation requires that the input and output of the component or circuit whose gain compression is being simulated be defined by an appropriate source and termination.

# XDB Simulation Parameters

The Gain-Compression (Xdb) Simulation controller enables you to define these aspects of the simulation:

- **Freq** -Frequencies of fundamentals and status level for summary information.
- **XdB** - Gain compression, port numbers and frequencies, and port tolerances
- **Params** -Parameters related to device operating-point levels, FFT oversampling and convergence, initial guess, writing final solution to a specified file, budget simulation, and harmonic balance assisted harmonic balance.
- **Solver** -Set up convergence mode, matrix solver, and memory management.
- **Output** -Selectively save simulation data to a dataset. For details, see *Selectively Saving and Controlling Simulation Data* (cktsim) in *Using Circuit Simulators* (cktsim).
- **Display** -Control the visibility of simulation parameters on the Schematic. For details, see *Displaying Simulation Parameters on the Schematic* (cktsim) in *Using Circuit Simulators* (cktsim).

## Freq

Following are descriptions of the fields to set up frequency for this component. In the following table, names used in netlists and ADS schematics appear under *Parameter Name*.

### XDB Simulation Freq Options

Setup Dialog Name	Parameter Name	Description
Fundamental Frequencies		
Edit		Edit the Frequency and Order fields, then use the buttons to Add the frequency to the list displayed under Select.
Frequency	Freq[n]	The frequency of the fundamental(s). Change by typing over the entry in the field. Select the units (None, Hz, kHz, MHz, GHz) from the drop-down list.
Order	Order[n]	The maximum order (harmonic number) of the fundamental(s) that will be considered. Change by typing over the entry in the field.
Select		Contains the list of fundamental frequencies. Use the Edit field to add fundamental frequencies to this window. Add -- Allows you to add an item. Cut -- Allows you to delete an item. Paste -- Allows you to take an item that has been cut and place it in a different order.
Maximum mixing order	MaxOrder	The maximum order of the intermodulation terms in the simulation. The combined order is the sum of the individual frequency orders that are added or subtracted to make up the frequency list. For example, assume there are two fundamentals and Order (see below) is 3. If Maximum order is 0 or 1, no mixing products are simulated. The frequency list consists of the fundamental and the first, second, and third harmonics of each source. If Maximum order is 2, the sum and difference frequencies are added to the list. If Maximum order is 3, the second harmonic of one source can mix with the fundamental of the others, and so on.
Levels-Enables you to set the level of detail in the simulation status report.		
Status level	StatusLevel	Prints information about the simulation in the Status/Summary part of the Message Window. A value of 0 causes no or minimal information to be reported, depending on the simulation engine. Higher values print more detail. 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.

## XdB

Following are the fields used to define XDB simulation. In the following table, names used in netlists and ADS schematics appear under *Parameter Name* .

### XDB Simulation X dB Options

Setup Dialog Name	Parameter Name	Description
Gain compression	GC_XdB	Enter the gain compression, the default is 1 dB.
Port numbers		
Input	GC_InputPort	The number of the input port.
Output	GC_OutputPort	The number of the output port.
Port frequencies-Ports must be placed and defined. Set the port number of the input source to 1, and the port number of the output Term (termination) component to 2, then set frequencies.		
Input	GC_InputFreq	The frequency of the input port.
Output	GC_OutputFreq	The frequency of the output port.
Power tolerances		
Input	GC_InputPowerTol	The variation in power, measured as a change in dB (delta dB), allowed at the input port.
Output	GC_OutputPowerTol	The variation in power, measured as a change in dB (delta dB), allowed at the input port.
Max. Input Power	GC_MaxInputPower	The maximum input power (in dBm) allowed before the simulator aborts. This setting is used to prevent the simulator from continuing to search if it is having difficulty finding a solution.

## Defining Simulation Parameters

Defining the simulation parameters consists of four basic parts:

- Enabling the budget simulation.
  - Specifying the desired level of detail in the simulation status summary
  - Specifying the amount of device operating-point information to save
- In the following table, names used in netlists and ADS schematics appear under *Parameter Name* .

### XDB Simulation Parameter Options

Setup Dialog Name	Parameter Name	Description
Device operating point level	DevOpPtLevel	Enables you to save all the device operating-point information to the dataset. In ADS, if this simulation performs more than one XDB analysis (from multiple XDB controllers), the device operating point data for all XDB analyses will be saved, not just the last one. Default setting is None.
None	None	No information is saved.
Brief	Brief	Saves device currents, power, and some linearized device parameters.
Detailed	Detailed	Saves the operating point values which include the device's currents, power, voltages, and linearized device parameters.
FFT		



Fundamental Oversample	FundOversample	Sets the FFT oversampling ratio. Higher levels increase the accuracy of the solution by reducing the FFT aliasing error and improving convergence. Memory and speed are affected less when the direct harmonic balance method is used than when the Krylov option is used.
More...	Oversample[n]	Displays a small dialog box. To increase simulation accuracy, enter in the field an integer representing a ratio by which the simulator will oversample each fundamental.

## Initial Guess

Use Initial Guess	UseInFile	Check this box to enter a file name for a solution to be used as initial guesses. This file is typically generated from a previous simulation by enabling Write Final Solution. If no initial guess file name is supplied, a default name (using DC solution) is generated internally, using the design name and appending the suffix .hbs. A suffix is neither required nor added to any user-supplied file name. For example, if you have saved the Harmonic Balance solution from a previous simulation, you can later do a nonlinear noise simulation and use this saved solution as the initial guess, removing the time required to recompute the nonlinear Harmonic Balance solution. Or you could quickly get to the initial Harmonic Balance solution, then sweep a parameter to see the changes. In this latter case, you will probably either want to disable the Write Final Solution option or use a different file name for the final solution to avoid overwriting the initial guess solution. (See <i>Write Final Solution</i> option below.)The Annotate value specified in the DC Solutions tab in the Options block is also used to control the amount of annotation generated when there are topology changes detected during the reading of the initial guess file. Since HB simulations also utilize the DC solution, to get optimum speed-up, both the DC solution and the HB solution should be saved and re-used as initial guesses. See DC Simulation.The initial guess file does not need to contain all the HB frequencies. For example, one could do a one-tone simulation with just a very nonlinear LO, save that solution away and then use it as an initial guess in a two tone simulation. The exact frequencies do not have to match between the present analysis and the initial guess solution. However, the fundamental indexes should match. For example, a solution saved from a two tone analysis with Freq[1] = 1GHz and Freq[2] = 1kHz would not be a good match for a simulation with Freq[1] = 1kHz and Freq[2] = 1 GHz.If the simulator cannot converge with the supplied initial guess, it then attempts to a global node-setting by connecting every node through a small resistor to an equivalent source. It then attempts to sweep this resistor value to a very large value and eventually tries to remove it.
File	InFile	Specify a filename to save results.
Regenerate Initial Guess for ParamSweep (Restart)		

## Final Solution

Write Final Solution	UseOutFile	Check this box to save your final HB solution to the output file. If a filename is not supplied, a file name is internally generate using the design name, followed by an .hbs suffix. If a file name is supplied, the suffix is neither appended nor required. If this box is checked, then the last HB solution is put out to the specified file. If this is the same file as that used for the Initial Guess, this file is updated with the latest solution. Transient simulations can also be programmed to generate a harmonic balance solution that can then be used as an initial guess for an HB simulation. Refer to Harmonic Balance Simulation.
----------------------	------------	--

File	OutFile	Specify a filename to save results.
Budget		
Perform Budget simulation	OutputBudgetIV	Enables Budget simulation, which reports current and voltage data at the pins of devices following a simulation. Current into the nth terminal of a device is identified as ...device_name.tn.i. Voltage at the nth terminal of a device is identified as ...device_name.tn.v.
Harmonic Balance Assisted Harmonic Balance	HBAHB_Enable	Set the HBAHB mode to <i>Auto</i> , <i>On</i> , or <i>Off</i> . Default is <i>Auto</i> and is recommended.

## Solver

Refer to the topic *Selecting a Harmonic Balance Solver Technique* (cktsimhb) in *Harmonic Balance Simulation* (cktsimhb).