Agilent Technologies

Advanced Design System 2011.01

Feburary 2011
Oscillator DesignGuide

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. FLEXlm 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 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).

Oscillator QuickStart Guide

The Oscillator QuickStart Guide serves as a simple introduction to using the Oscillator DesignGuide. For more detailed reference information, refer to *Oscillator DesignGuide Reference* (dgosc).

The DesignGuide is applicable to any oscillator, but is especially useful for RF Board and Microwave applications. It is designed to help both experts and novices to create designs of various complexity.

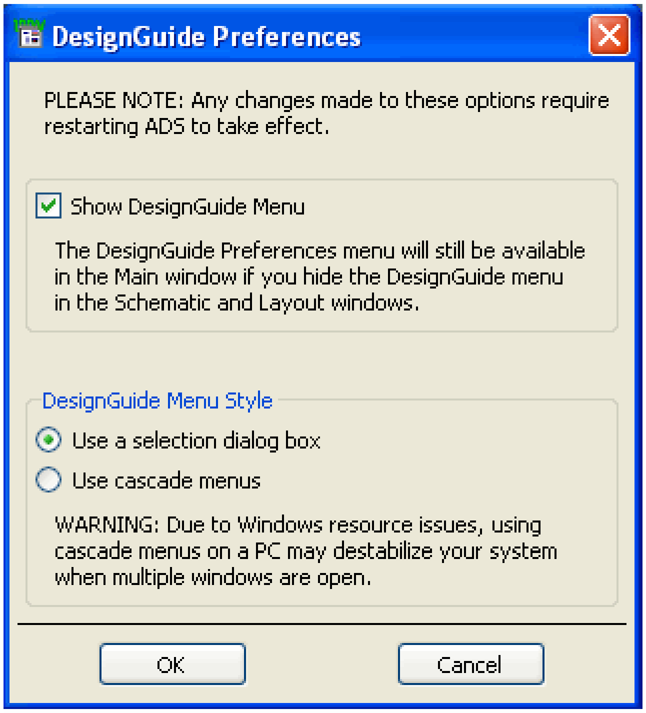
Using DesignGuides

All DesignGuides can be accessed in the Schematic window through either cascading menus or dialog boxes. You can configure your preferred method in the Advanced Design System Main window.

Select the **DesignGuide** menu from Advanced Design System Main window.

The commands in this menu are as follows:

- **DesignGuide Developer Studio > Developer Studio Documentation** is only available on this menu if you have installed the DesignGuide Developer Studio. It brings up the DesignGuide Developer Studio documentation. Another way to access the Developer Studio documentation is by selecting **Help > Topics and Index > DesignGuides > DesignGuide Developer Studio** (from any ADS program window).
- **DesignGuide Developer Studio** > **Start DesignGuide Studio** is only available on this menu if you have installed the DesignGuide Developer Studio. It launches the initial Developer Studio dialog box.
- **Add DesignGuide** brings up a directory browser in which you can add a DesignGuide to your installation. This is primarily intended for use with DesignGuides that are custom-built through the Developer Studio.
- **List/Remove DesignGuide** brings up a list of your installed DesignGuides. Select any that you would like to uninstall and choose the **Remove** button.
- **Preferences** brings up a dialog box that allows you to:
    - Disable the DesignGuide menu commands (all except Preferences) in the Main window by unchecking this box. In the Schematic and Layout windows, the complete DesignGuide menu and all of its commands will be removed if this box is unchecked.
    - Select your preferred interface method (cascading menus vs. dialog boxes).



DesignGuide Preferences

PLEASE NOTE: Any changes made to these options require restarting ADS to take effect.

☑ Show DesignGuide Menu

The DesignGuide Preferences menu will still be available in the Main window if you hide the DesignGuide menu in the Schematic and Layout windows.

DesignGuide Menu Style

● Use a selection dialog box

○ Use cascade menus

WARNING: Due to Windows resource issues, using cascade menus on a PC may destabilize your system when multiple windows are open.

Close and restart the program for your preference changes to take effect.

Note
On PC systems, Windows resource issues might limit the use of cascading menus. When multiple windows are open, your system could become destabilized. Thus the dialog box menu style might be best for these situations.

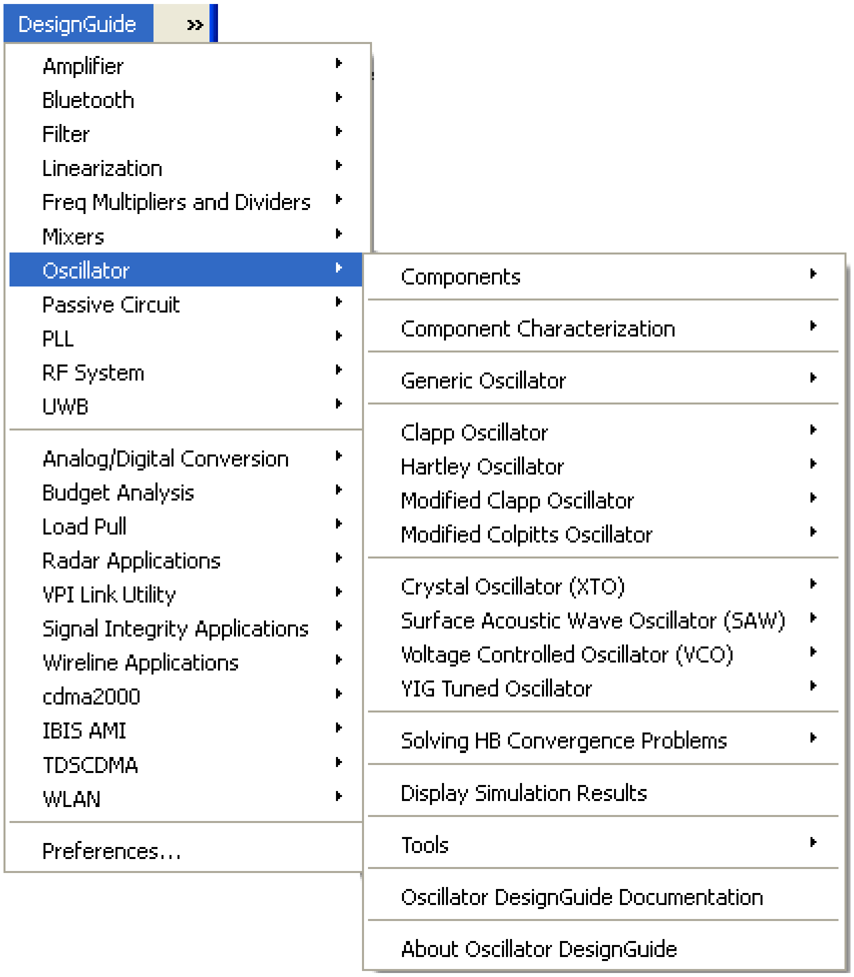
Accessing the Documentation

To access the documentation for the DesignGuide, select either of the following:

- **DesignGuide** > **Oscillator** > **Oscillator DesignGuide Documentation** (from ADS Schematic window)
- **Help** > **Topics and Index** > **DesignGuides** > **Oscillator** (from any ADS program window)

Basic Procedures

Access the Oscillator DesignGuide from the ADS Schematic window. Select **DesignGuide > Oscillator**, as shown here. All features of the Design Guide are available from the **Oscillator** DesignGuide menu.



The Guide contains the following:

- Nine oscillator circuits (Generic Oscillator, Clapp Oscillator, Hartley Oscillator, Modified Clapp Oscillator, Modified Colpitts Oscillator, XTO, SAW, VCO, and YTO), containing ready-to-use typical oscillator structures for fixed frequency (XTO, SAW) and tunable (VCO, YTO) oscillators in various frequency ranges.
- Component library (Components), providing useful building blocks for oscillator design.
- Selection of circuits that simulate their behavior (Component Characterization), providing simulations that characterize 1-ports and 2-ports.

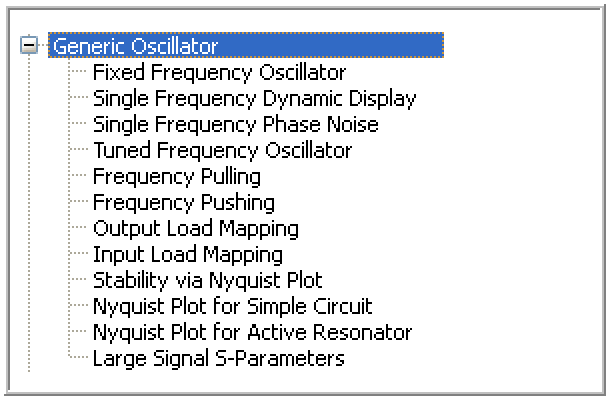
Note
Selection of a component brings a component into a design. All other selections (oscillators and component characterization) bring a circuit and simulation into a design (replacing a previous design).

Component Sub-menu Structure

The **Components** menu contains a small custom library of resonators and devices, which can help in either modifying an existing oscillator or assembling a new one. They include device DC and S-parameter characteristics, resonator and filter S-parameter, and impedance/admittance characteristics.

Oscillator Sub-menu Structure

Select **DesignGuide > Oscillator > Generic Oscillator** and explore the entries, as shown here.



Each Oscillator design is divided into two groups: large-signal measurements and linear/nonlinear design tools.

Easy-to-Use Large-Signal Measurements

Easy-to-use large-signal measurements (for *push-button* nonlinear analysis), contain simulations of the following:

- Single-frequency oscillations
- Phase noise
- Tuned oscillations
- Frequency pulling
- Frequency pushing

This group is recommended as starting point for both an expert and a novice user. For an expert, it provides an overview of tool capabilities. For a novice user, it provides a working oscillator together with simulations of its typical characteristics. You can choose either the Generic Oscillator, Clapp Oscillator, Hartley Oscillator, Modified Clapp Oscillator, Modified Colpitts Oscillator, or one of the examples (XTO, SAW, VCO, YTO) to start a desired application.

Linear and Nonlinear Design Tools

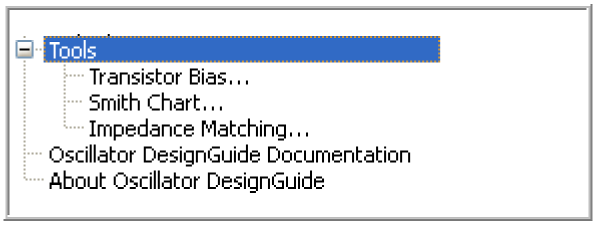
Linear and nonlinear design tools include Output Load Mapping, Input Load Mapping, Stability via Nyquist Plot, Nyquist Plot for Simple Circuit, Nyquist Plot for Active Resonator, and Large Signal S-Parameters.
These tools are intended as an aide in designing an oscillator from scratch and in gaining insight into an existing oscillator. The full choice of tools is contained in the Generic Oscillator. The examples use only those tools that are useful in their particular case. The full set of tools include the following:

- Load mapping for load-to-resonator
- Resonator-to-load
- Nyquist stability criterion for varying Zo
- Two additional examples (only in Generic Oscillator), explaining the role of Zo

For nonlinear designs, large-signal S-parameters are defined and applied to oscillator power and frequency prediction.

Tools Sub-menu Structure

These utilities provide added functionality to this DesignGuide. A brief description is provided for each. For more information select the **Help** button located in each utility.



Transistor Bias Utility

The Transistor Bias Utility provides SmartComponents and automated-assistants for the design and simulation of common resistive and active transistor bias networks. The automated capabilities can determine the transistor DC parameters, design an appropriate network to achieve a given bias point, and simulate and display the achieved performance. All SmartComponents can be modified when selected. You simply select a SmartComponent and with little effort redesign or verify their performance.

Smith Chart Utility

This DesignGuide Utility provides full smith chart capabilities, synthesis of matching networks, allowing impedance matching and plotting of constant Gain/Q/VSWR/Noise circles. This guide assumes you have installed the associated DesignGuide with appropriate licensing codewords.

Impedance Matching Utility

The Impedance Matching Utility performs the synthesis of lumped and distributed impedance matching networks based on provided specifications. The Utility features automatic simulation, sensitivity analysis, and display setup to enable simple and efficient component verification.

Design Flow Example

Following is a simple design flow example for a fixed frequency oscillator.

Preliminary Steps

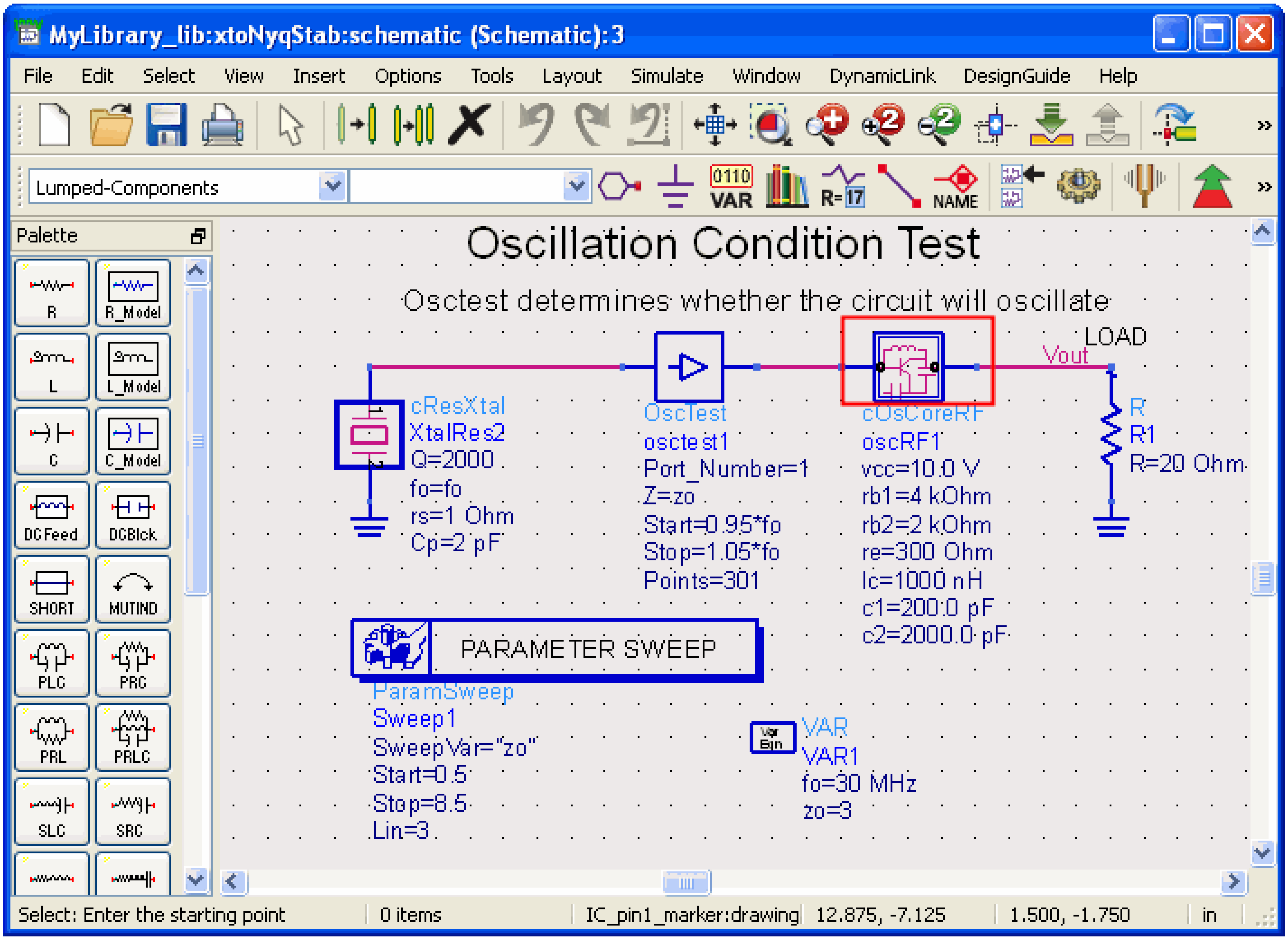
1. Open ADS.
2. Open a new or existing workspace.
3. Open a new Schematic window.
4. Select **DesignGuide** > **Oscillator**.
5. Select **Crystal Oscillator (XTO)** > **Stability via Nyquist Plot**.

Important Preliminary Decisions

The schematic hides the following important choices:

- Device (a BJT)
- Biasing circuit
- Feedback scheme (Colpitts used in OsCore)
- Biasing point, which is shown in the OsCore subcircuit (which you can see from the Schematic window by clicking the **OsCore** component in the design, then pressing the down-arrow from the Main menu).

The schematic is shown in the following illustration.



Oscillation Condition Test

Osctest determines whether the circuit will oscillate

cResXtal
XtalRes2
Q=2000
fo=fo
rs=1 Ohm
Cp=2 pF

OscTest
osctest1
Port_Number=1
Z=zo
Start=0.95*fo
Stop=1.05*fo
Points=301

cOsCoreRF
oscRF1
vcc=10.0 V
rb1=4 kOhm
rb2=2 kOhm
re=300 Ohm
lc=1000 nH
c1=200.0 pF
c2=2000.0 pF

R
R1
R=20 Ohm

ParamSweep
Sweep1
SweepVar="zo"
Start=0.5
Stop=8.5
Lin=3

VAR
VAR1
fo=30 MHz
zo=3

Moreover, the 30 MHz resonance frequency is assumed and the 20-ohm load resistance that models the actual load seen through a buffer amplifier and a matching circuit.
At this point, you might want to modify the resonator and the **OsCore** (or replace them by your own circuit).

The modified circuit can be saved as a new design. We want the S11 trace shown on the polar plot in the data display window to encircle the point *1+j*0* .
If this does not happen, the circuit must be modified. In that case, the menu selections **Output Load Mapping** and **Input Load Mapping** will help in determining the circuit and the load matching. Refer to the *Oscillator DesignGuide Reference* (dgosc) for details.

Oscillator Performance

The following menu selections determine the oscillator performance:

- Fixed Frequency Oscillator
- Single Frequency Phase Noise
- Frequency Pulling

- Frequency Pushing

They determine the oscillation frequency and power, phase and amplitude noise, and circuit elements that contribute most to noise.
You can find frequency variations with load and bias. Modify (or replace) the subcircuits (the resonator, the OsCore, and the load resistance).

Oscillator DesignGuide Reference

This document provides reference information on the use of the Oscillator DesignGuide.

Oscillator DesignGuide Structure

The Oscillator DesignGuide is integrated into Agilent EEsof's Advanced Design System environment, working as a smart library and interactive handbook for the creation of useful designs. It allows you to quickly design oscillators, interactively characterize their components, and receive in-depth insight into their operation. It is easily modifiable to user-defined configurations. The first release of this DesignGuide focuses on RF printed circuit boards and microwave oscillations.

In addition to the requirements of the ADS software, the Oscillator DesignGuide requires approximately 30 MBytes of additional storage space.

> **Note**
> This document assumes that you are familiar with all of the basic ADS program operations. For additional information, refer to *Schematic Capture and Layout* (usrguide).

The Oscillator DesignGuide contains templates that can be used in the ADS software environment. It consists of generic colpitts, clapp, modified colpitts, modified clapp, and hartley oscillator design examples, and a library of components and component characterization tools.
To assist both expert and novice oscillator developers in creating designs of various complexity, each example design is divided into three groups:

- Quick and simple push-button nonlinear oscillator measurements
- Easy-to-use design tools for small- and large-signal designs
- Customized library of components and component characterization tools

To access these groups, select **DesignGuide > Oscillator DesignGuide** from the ADS Schematic window, then select the appropriate examples and tools.

Push-Button Nonlinear Measurements

The push-button nonlinear measurements are recommended as a starting point for both expert and novice users creating large-signal designs. For the expert, these measurements provide an overview of tool capabilities. For the novice user, they provide a working oscillator together with simulations of its typical characteristics of nonlinear designs. The full set of available large-signal measurements in the Generic Oscillator example are described in the following table. Subsets of these measurements appear in other examples. Refer to the section Additional Examples

Descriptions of Push-Button Measurements

Measurement	Schematic Filenames	Data Display Filename
Fixed Frequency Oscillator	FixedFreqOsc	FixedFreqOsc.dds
Single Frequency Dynamics Display	n/a	LargeSignalDynamics.dds
Single Frequency Phase Noise	PhaseNoise	PhaseNoise.dds
Tuned Frequency Oscillator	FreqTune	FreqTune.dds
Frequency Pulling	FreqPull	FreqPull.dds
Frequency Pushing	FreqPush	FreqPush.dds

Linear and Nonlinear Design Tools

The linear and nonlinear design tools are intended to facilitate you in designing an oscillator from scratch and in gaining insight into an existing oscillator. The full selection of tools is contained in the Generic Oscillator example. Other examples use only those tools that are useful in their particular case.

Descriptions of Design Tools

Measurement	Schematic Filename	Data Display Filename
Output Load Mapping	MapLoad	MapLoad.dds
Input Load Mapping	MapInput	MapInput.dds
Stability via Nyquist Plot	NyqStab	NyqStab.dds
Nyquist Plot for Simple Circuit	NyqPlot	NyqPlot.dds
NyqPlot for Active Resonator	NyqPlotA	NyqPlotA.dds
LSSpar	LSSpar	LSSpar.dds

Components and Component Characterization Tools

The items in the Components and Component Characterization libraries contain a small custom library of resonators and devices, which can help in either modifying an existing oscillator or assembling a new one. They include device DC and S-parameter characteristics, as well as resonator and filter S-parameter and impedance/admittance characteristics.

The following tables provide schematic filenames and brief descriptions for each component and brief descriptions for each component characterization tool.

Active Device Components

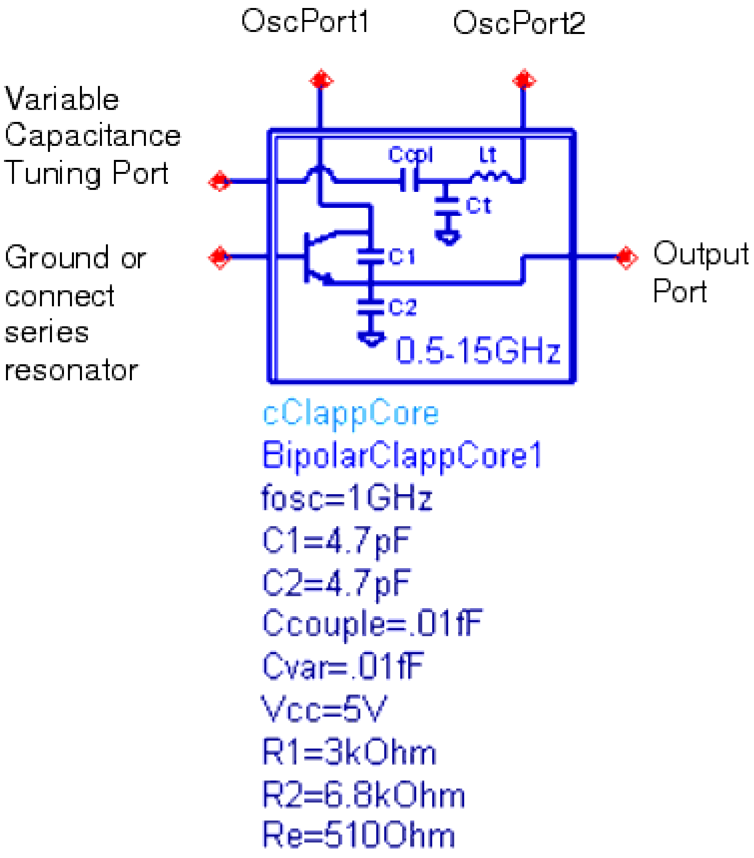
Component Description	Schematic Filename	Description
Biased BJT	cBJTBiased	Common Emitter BJT with a standard (1-voltage source) biasing circuit
Biased RF BJT	cBJTRFBiased	The RF version of BJT used in Crystal Oscillator
Biased MESFET	cFETBiased	n/a
Varactor Diode	cVar	Varactor diode model, included for convenience. Within this DesignGuide, it always appears with the reverse-biasing circuit (see next table entry).
Biased Varactor Diode	cbVar	Reversed-biasing varactor

Subcircuit Components

Component Description	Schematic Filename	Description
Buffer Amplifier (microwave)	cAmpBuff	A simple amplifier with capacitive feedback used in frequency pull and push simulations, used above 2GHz (for lower frequencies, see below). You are encouraged to replace it by your own amplifier and matching circuit.
Buffer Amplifier (1 - 2 MHz)	cAmpBuffS	Buffer amplifier with reactive components adjusted for 1GHz to 2GHz range, used in SAW oscillator
Buffer Amplifier (10 - 100 MHz)	cAmpBuffX	Buffer amplifier with reactive components adjusted for 10MHz to 100MHz range, used in crystal oscillator
Oscillator Core	cOsCore	Colpitts structure with a BJT with standard bias
RF Oscillator Core	cOsCoreRF	Oscillator Core adapted to MHz frequency range
Clapp Oscillator Core	cClappCore	Bipolar Clapp Oscillator covering a frequency range of 0.5 to 15GHz.
Hartley Oscillator Core	cHartleyCore	Bipolar Hartley Oscillator covering a frequency range of 1 to 1000MHz
Modified Clapp Oscillator Core	cModifiedClappCore	Bipolar Modified Clapp Oscillator covering a frequency range of 0.7 to 7.2GHz.
Modified Colpitts Oscillator Core	cModifiedCopittsCore	Bipolar Colpitts Oscillator covering a frequency range of 0.8 to 6.5GHz.
Fixed VSWR Complex Load	cLoadEqs	Load determined through VSWR and phase of the reflection coefficient

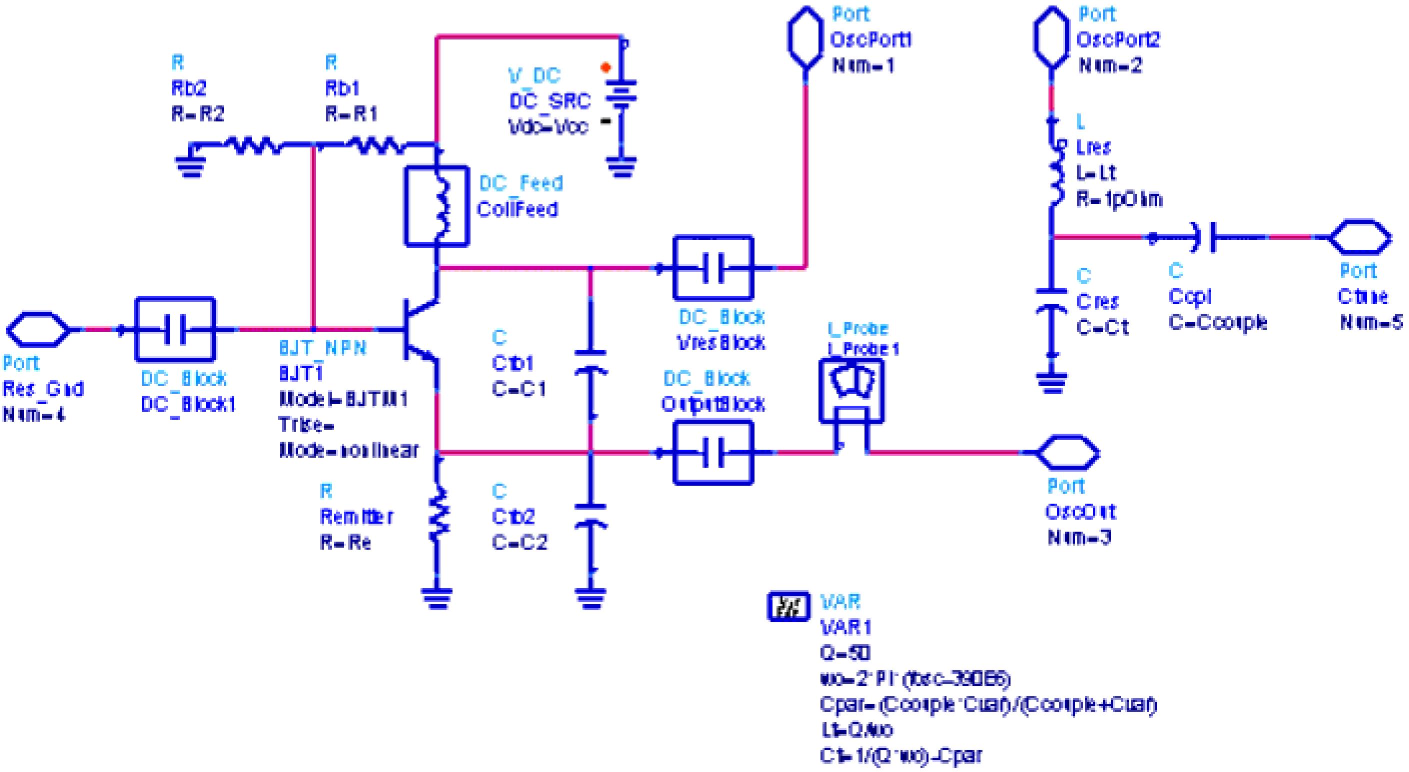
Oscillator Core Circuits

The following image shows the cClappCore oscillator core schematic symbol.



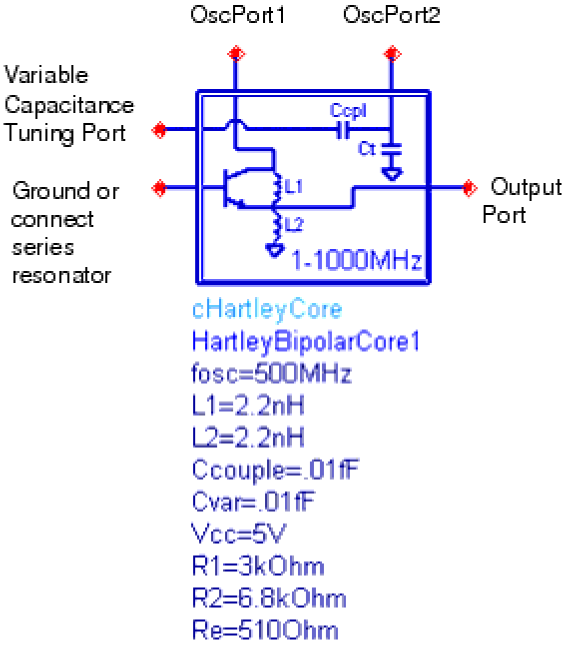
Bipolar Clapp Oscillator Schematic Symbol

The cClappCore oscillator operates from 0.5 to 15GHz using the existing component values in the Clapp oscillator sub-circuit. Resonator tank components *Ct* and *Lt* are automatically calculated with approximations referenced to 1GHz and displayed on some display pages. The *Ground or Series Resonator* port is either connected directly to ground or connected to ground through a series resonator. The *Variable Capacitance Tuning Port* is used for *VCO* design by coupling a varactor diode across the tank capacitor *Ct*. Coupling is accomplished by capacitor *Ccpl*. Larger values of *Ccpl* yield tighter coupling and wider tuning range for a given amount of tuning capacitance variation. The *OscPort1* and *OscPort2* ports are either connected directly together or connected through a series resonator. You can connect the *OscPort* test probe between these two ports for harmonic balance oscillator simulations. The *Output Port* is used for oscillator signal output. The variable *Ctune* sets the oscillator at the desired oscillation frequency when the capacitance across the *Variable Capacitance Tuning Port* is equal to the *Ctune* set value. The following image shows an example of the Clapp Oscillator subcircuit.



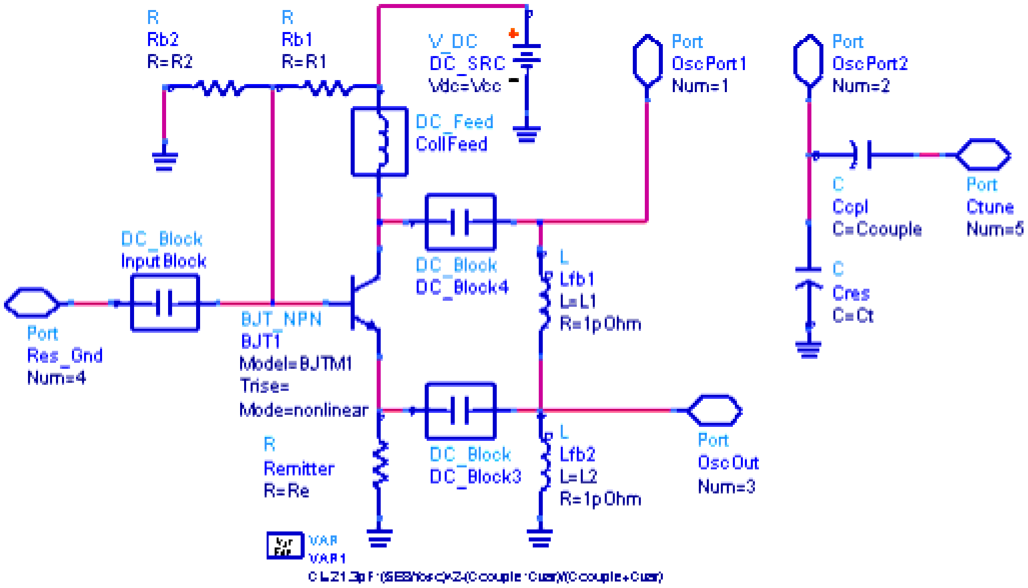
Bipolar Clapp Oscillator Subcircuit

Increase capacitors *C1* and *C2* for lower frequency oscillator circuits. Adjust resonator frequency offset (currently 390E6) to recenter oscillator frequency.



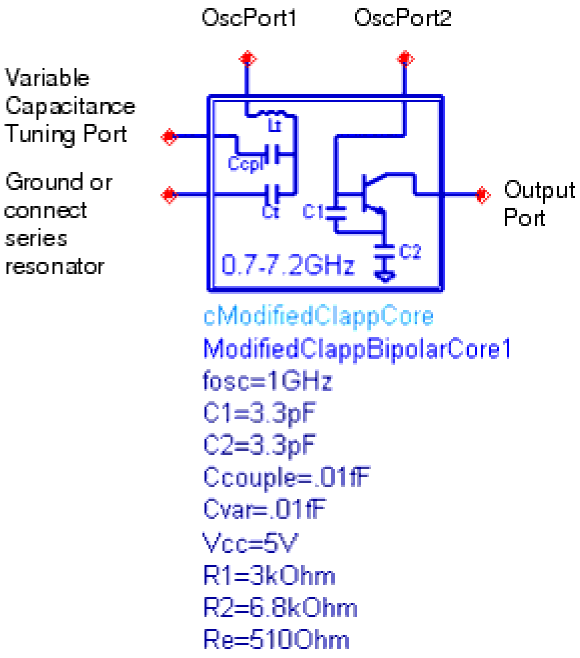
Bipolar Hartley Oscillator Schematic Symbol

The preceding image shows the *cHartleyCore* oscillator core schematic symbol. This oscillator operates from 1 to 1000MHz using the existing component values in the Hartley oscillator sub-circuit. The resonator tank component *Ct* is scaled from 500MHz with an approximation and displayed on some display pages. You can connect the *Ground or Series Resonator* to the port directly to ground, or it can be connected to ground through a series resonator. The *Variable Capacitance Tuning Port* is used for VCO design by coupling a varactor diode across the tank capacitor *Ct*. Coupling is accomplished by capacitor *Ccpl*. Larger values of *Ccpl* yield tighter coupling and wider tuning range for a given amount of tuning capacitance variation. The *OscPort1* and *OscPort2* ports are either connected directly together or connected through a series resonator. For harmonic balance oscillator simulations, connect the *OscPort* test probe between these two ports. The *Output Port* is used for oscillator signal output. The variable *Ctune* sets the oscillator at the desired oscillation frequency when the capacitance across the *Variable Capacitance Tuning Port* is equal to the *Ctune* set value. The following image shows the Hartley Oscillator subcircuit.



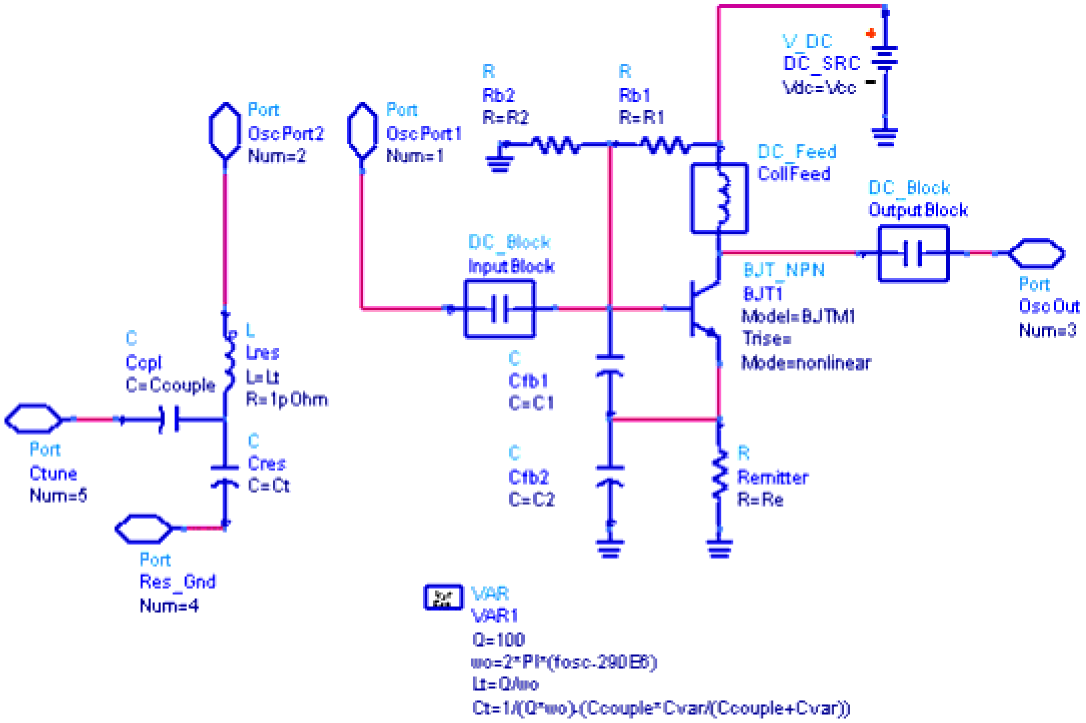
Bipolar Hartley Oscillator Subcircuit

Increase inductors *L1* and *L2* for lower frequency oscillator circuits. Adjust resonator capacitor *Ct* reference value (currently 21.3pF) to recenter oscillator frequency.



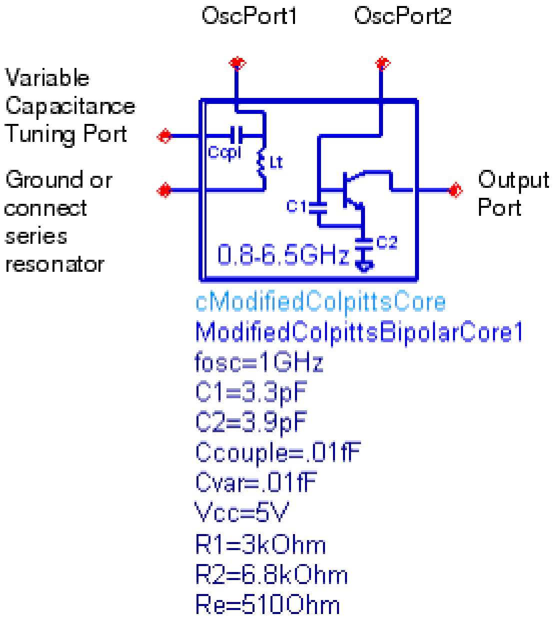
Bipolar Modified Clapp Oscillator Schematic Symbol

The preceding image shows the cModifiedClappCore oscillator core schematic symbol. The cModifiedClappCore oscillator operates from 0.7 to 7.2GHz using the existing component values in the Modified Clapp oscillator sub-circuit. Resonator tank components Ct and Lt are automatically calculated with approximations referenced to 1GHz and displayed on some display pages. The *Ground or Series Resonator* port can be connected directly to ground or connected to ground through a series resonator. The *Variable Capacitance Tuning Port* is used for VCO design by coupling a varactor diode across the tank capacitor *Ct*. Coupling is accomplished by capacitor *Ccpl*. Larger values of *Ccpl* yield tighter coupling and wider tuning range for a given amount of tuning capacitance variation. The *OscPort1* and *OscPort2* ports are either connected directly together or connected through a series resonator. You can connect the *OscPort* test probe between these two ports for harmonic balance oscillator simulations. The *Output Port* is used for oscillator signal output. Variable *Ctune* sets the oscillator at the desired oscillation frequency when the capacitance across the *Variable Capacitance Tuning Port* is equal to the *Ctune* set value. The following image shows the Modified Clapp Oscillator subcircuit.



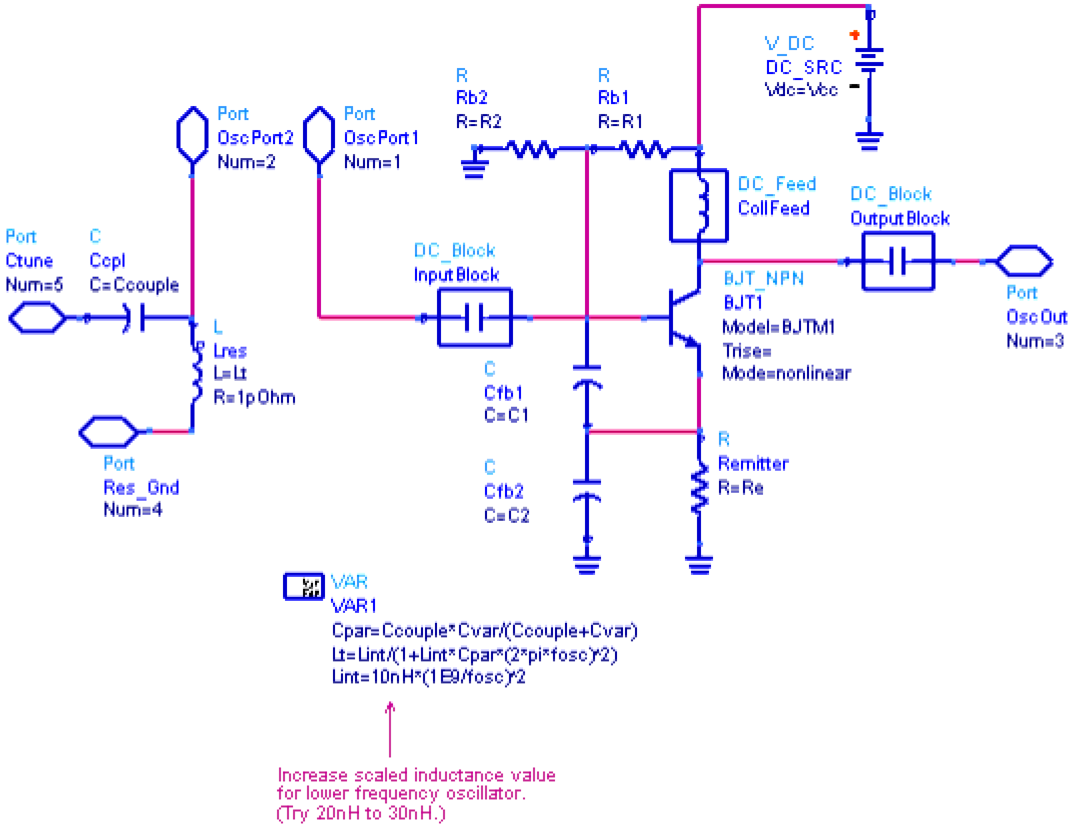
Bipolar Modified Clapp Oscillator Subcircuit

Increase capacitors C1 and C2 for lower frequency oscillator circuits. Adjust resonator frequency offset (currently 290E6) to recenter oscillator frequency.



Bipolar Modified Colpitts Oscillator Schematic Symbol

The preceding image shows the *cModifiedColpittsCore* oscillator core schematic symbol. This oscillator operates from 0.8 to 6.5GHz using the existing component values in the Modified Colpitts oscillator sub-circuit. Resonator tank inductor *Lt* is automatically calculated with an approximation and displayed on some display pages. The *Ground or Series Resonator* port can be connected directly to ground or connected to ground through a series resonator. The *Variable Capacitance Tuning Port* is used for VCO design by coupling a varactor diode across the tank inductor *Lt*. Coupling is accomplished by capacitor *Ccpl*. Larger values of *Ccpl* yield tighter coupling and wider tuning range for a given amount of tuning capacitance variation. The *OscPort1* and *OscPort2* ports are either connected directly together or connected through a series resonator. Connect the *OscPort* test probe between these two ports for harmonic balance oscillator simulations. The *Output Port* is used for oscillator signal output. The variable *Ctune* sets the oscillator at the desired oscillation frequency when the capacitance across the *Variable Capacitance Tuning Port* is equal to the *Ctune* set value. The following image shows the Modified Colpitts Oscillator subcircuit.



Bipolar Modified Colpitts Oscillator Subcircuit

Increase scaled 10nH inductance reference value for lower frequency oscillator circuits. Tank inductance Lt is scaled from 1GHz using an approximation. Increasing C1 and C2 capacitors yields a lower frequency oscillator as well.

Component Description	Schematic Filename	Description
Crystal Resonator	cResXtal	Straightforward resonator model
SAW Resonator	cResSAW	Straightforward resonator model
YIG Resonator	cResYIG	Straightforward resonator model
Parallel Resonator	cResP	Straightforward resonator model
Series Resonator	cResS	Straightforward resonator model

Available Component Characterization Tools

Tool Description	**Schematic Filename**	**Data Display Filename**	**Description**
S-parameters for 1-port	cz1PortSp	cz1PortSp.dds	S-parameter simulation of a 1-port
S-parameters for 2-port	cz2PortSp	cz2PortSp.dds	S-parameter simulation of a 2-port. Uses the Buffer Amplifier
BJT Curve Tracer	czBJTCurveTracer	czBJTCurveTracer.dds	DC Curves for a common emitter BJT, they can be observed independently or combined with periodic waveforms in LargeSignal-Dynamics.dds
RF BJT Curve Tracer	czBJTRFCurveTracer	czBJTRFCurveTracer.dds	The RF version of BJT used in Crystal Oscillator
FET Curve Tracer	cFETBiased	czFETCurveTracer.dds	n/a
S-parameters for Biased BJT	czBJTSp	czBJTSp.dds	S-parameter simulation of biased BJT
Capacitance and Admittance of Biased Varactor	czbVarSp	czbVarSp.dds	S-parameter simulation of the reversed biased varactor. Displays admittance values and capacitance versus the biasing voltage
S-parameters for Parallel Resonator	czResPSp	czResPSp.dds	n/a
S-parameters for Series Resonator	czResSSp	czResSSp.dds	n/a
S-parameters for Generic Resonator	czResScvSp	czResScvSp.dds	The resonator contains a series resonator with parallel capacitance.
S-parameters for Crystal Resonator	czResXtalSp	czResXtalSp.dds	n/a
S-parameters for SAW Resonator	czResSAWSp	czResSAWSp.dds	n/a
S-parameters for YIG Resonator	czResYIGSp	czResYIGSp.dds	n/a

Generic Oscillator Example

The oscillator circuit for the Generic Oscillator example is set up as follows:

- Resonator
- Oscillator active part (OscCore)
- Lload, which can include buffering amplifier and matching circuits

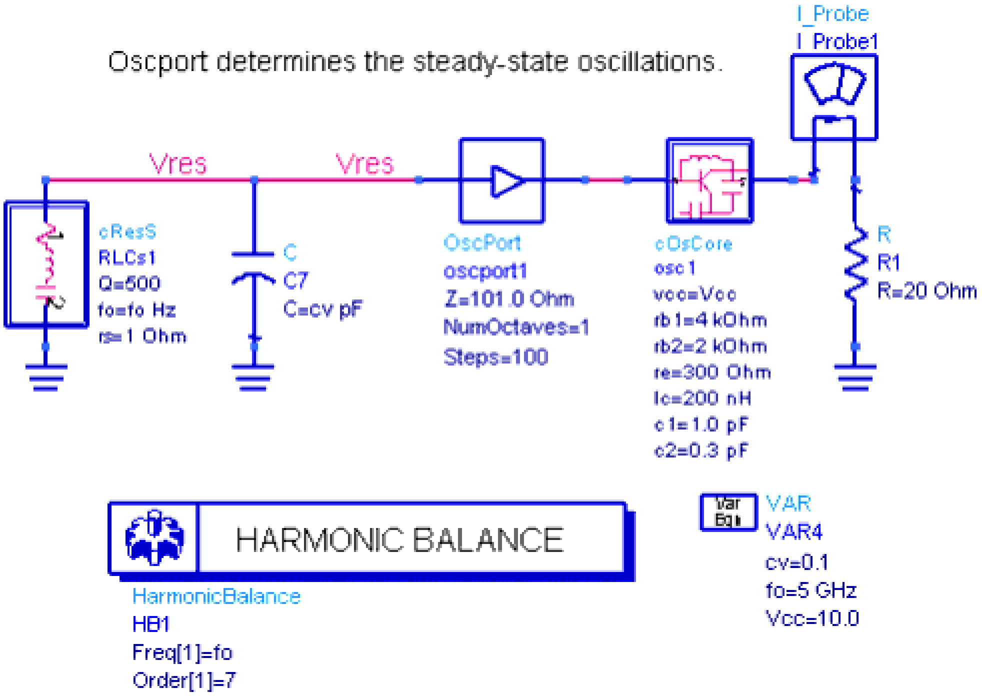
The tools consists of three parts, as explained in the Oscillator Design Guide Structure section.
For simplicity, we show the buffering amplifier in two designs only and don't include the matching circuits. The generic resonator is presented here by a series resonant circuit shunted by a capacitance, which can model either a tuning capacitance, or the effect of packaging.

The OsCore is the Colpitts structure, which was introduced soon after the invention of triode (called audion at the time). It is still widely used. It uses a capacitive RF transformer to provide feedback. The transformer capacitors together with the inductor determine the oscillation frequency $f = 1/sqrt(L \times C1 \times C2/(C1+C2))$.

The Load models impedance, as seen by the OsCore component. Typically, this is the input impedance of the buffering amplifier(s) (including matching circuits) and the actual load.

Oscillator Simulation

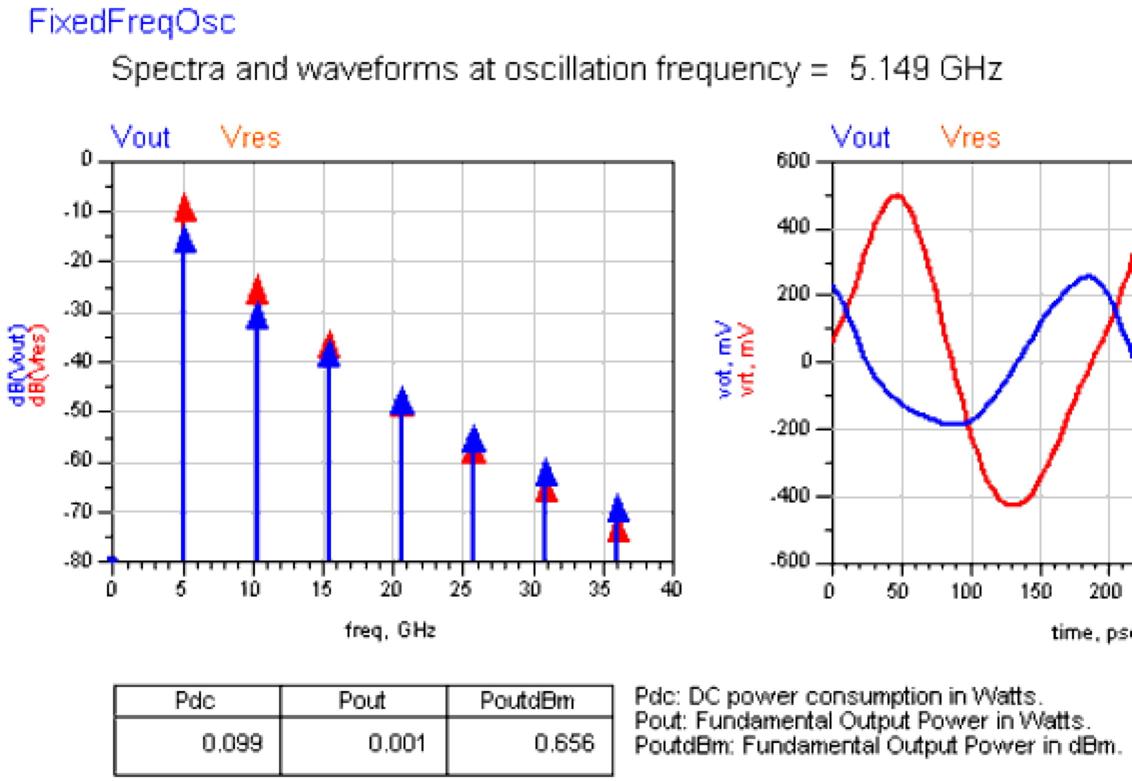
The oscillator designs included in this DesignGuide provide easy access to observing the nonlinear behavior of an oscillator. The circuit design for the generic example is shown in the Structure of Oscillator Circuit section It includes fixed load, a clearly visible active circuit, and the tunable resonator separated by the *OscPort* component.



Structure of Oscillator Circuit

Single-Frequency Oscillations

The results of single-frequency oscillations in the following image show output and resonator voltages. They also provide oscillations frequency, power harmonic content, the corresponding time-domain waveform, and the values of DC power and RF output power.



FixedFreqOsc

Spectra and waveforms at oscillation frequency = 5.149 GHz

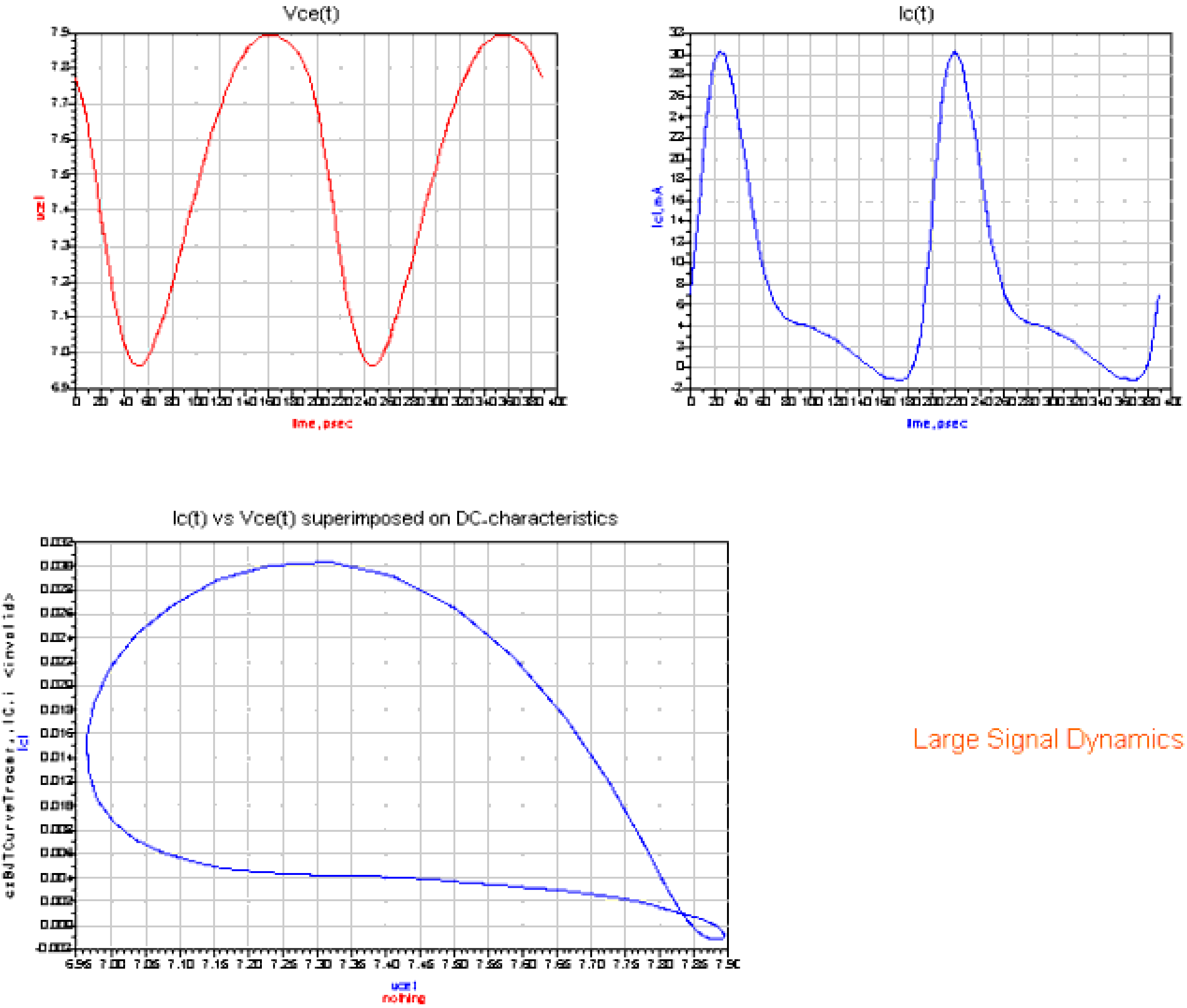
Pdc	Pout	PoutdBm
0.099	0.001	0.656

Pdc: DC power consumption in Watts.
Pout: Fundamental Output Power in Watts.
PoutdBm: Fundamental Output Power in dBm.

Results of Single-Frequency Oscillations

Dynamics of Single-Frequency Oscillations

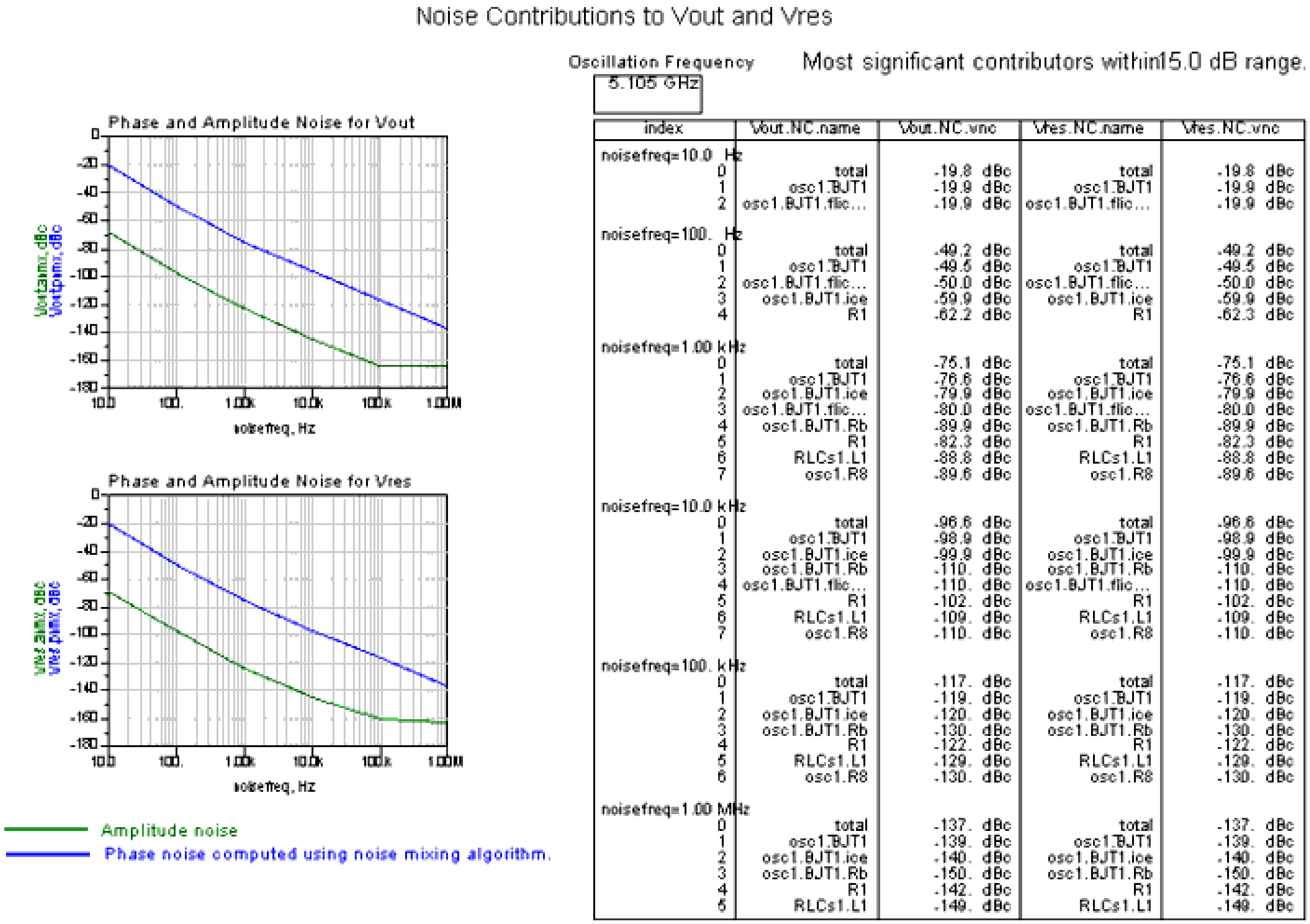
The graph in the following image shows waveforms of Collector-Emitter voltage and the collector current of the OsCore BJT superimposed on BJT's DC characteristics.



Dynamics of Single-Frequency Oscillation

Single-Frequency Oscillations with Noise

The graph in the following image shows single-frequency results with the noise characteristics of Vout and Vres. It also lists the components that affect the noise the most. You can specify the range after which small contributors to noise will be neglected. In this example, the range is set to 15 dB.



Noise Contributions to Vout and Vres

Oscillation Frequency
5.105 GHz

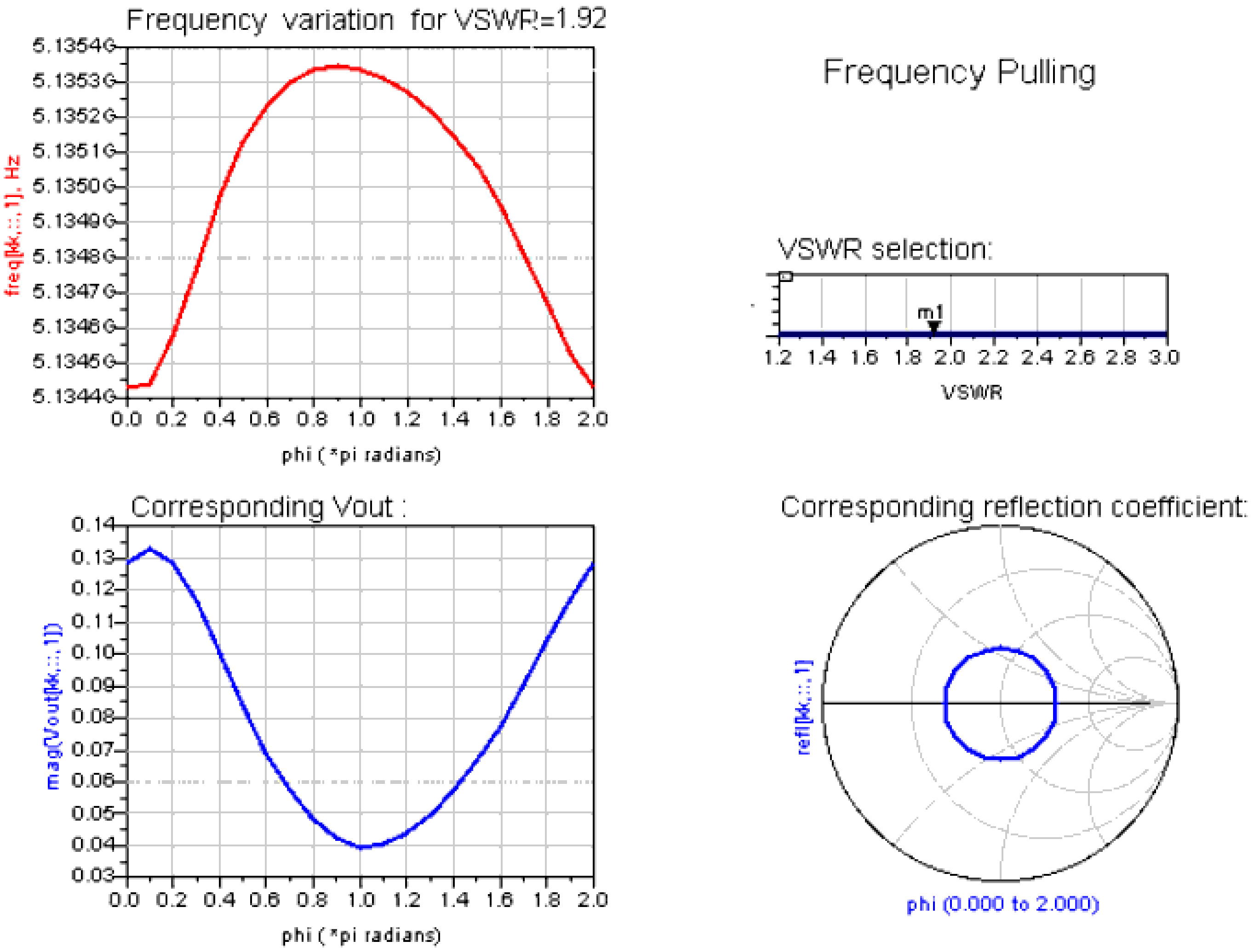
Most significant contributors within15.0 dB range.

index	Vout.NC.name	Vout.NC.vnc	Vres.NC.name	Vres.NC.vnc
noisefreq=10.0 Hz				
0	total	-19.8 dBc	total	-19.8 dBc
1	osc1.BJT1	-19.9 dBc	osc1.BJT1	-19.9 dBc
2	osc1.BJT1.flic...	-19.9 dBc	osc1.BJT1.flic...	-19.9 dBc
noisefreq=100. Hz				
0	total	-49.2 dBc	total	-49.2 dBc
1	osc1.BJT1	-49.5 dBc	osc1.BJT1	-49.5 dBc
2	osc1.BJT1.flic...	-50.0 dBc	osc1.BJT1.flic...	-50.0 dBc
3	osc1.BJT1.ice	-59.9 dBc	osc1.BJT1.ice	-59.9 dBc
4	R1	-62.2 dBc	R1	-62.3 dBc
noisefreq=1.00 kHz				
0	total	-75.1 dBc	total	-75.1 dBc
1	osc1.BJT1	-76.6 dBc	osc1.BJT1	-76.6 dBc
2	osc1.BJT1.ice	-79.9 dBc	osc1.BJT1.ice	-79.9 dBc
3	osc1.BJT1.flic...	-80.0 dBc	osc1.BJT1.flic...	-80.0 dBc
4	osc1.BJT1.Rb	-89.9 dBc	osc1.BJT1.Rb	-89.9 dBc
5	R1	-82.3 dBc	R1	-82.3 dBc
6	RLCs1.L1	-88.8 dBc	RLCs1.L1	-88.8 dBc
7	osc1.R8	-89.6 dBc	osc1.R8	-89.6 dBc
noisefreq=10.0 kHz				
0	total	-96.6 dBc	total	-96.6 dBc
1	osc1.BJT1	-98.9 dBc	osc1.BJT1	-98.9 dBc
2	osc1.BJT1.ice	-99.9 dBc	osc1.BJT1.ice	-99.9 dBc
3	osc1.BJT1.Rb	-110. dBc	osc1.BJT1.Rb	-110. dBc
4	osc1.BJT1.flic...	-110. dBc	osc1.BJT1.flic...	-110. dBc
5	R1	-102. dBc	R1	-102. dBc
6	RLCs1.L1	-109. dBc	RLCs1.L1	-109. dBc
7	osc1.R8	-110. dBc	osc1.R8	-110. dBc
noisefreq=100. kHz				
0	total	-117. dBc	total	-117. dBc
1	osc1.BJT1	-119. dBc	osc1.BJT1	-119. dBc
2	osc1.BJT1.ice	-120. dBc	osc1.BJT1.ice	-120. dBc
3	osc1.BJT1.Rb	-130. dBc	osc1.BJT1.Rb	-130. dBc
4	R1	-122. dBc	R1	-122. dBc
5	RLCs1.L1	-129. dBc	RLCs1.L1	-129. dBc
6	osc1.R8	-130. dBc	osc1.R8	-130. dBc
noisefreq=1.00 MHz				
0	total	-137. dBc	total	-137. dBc
1	osc1.BJT1	-139. dBc	osc1.BJT1	-139. dBc
2	osc1.BJT1.ice	-140. dBc	osc1.BJT1.ice	-140. dBc
3	osc1.BJT1.Rb	-150. dBc	osc1.BJT1.Rb	-150. dBc
4	R1	-142. dBc	R1	-142. dBc
5	RLCs1.L1	-149. dBc	RLCs1.L1	-149. dBc

Single-Frequency Oscillations with Noise

Frequency Pulling

The following image shows results after the original circuit is modified for frequency pull. Changes include the varying load and a simple buffer amplifier, which was added to make pulling values realistic. The load is specified in terms of VSWR. You can determine the best variation. The graph shows that frequency variation for varying phase of the load. VSWR is fixed, with its value shown above the plot. By moving the marker on the VSWR selection plot, you can obtain the results for other VSWR values. The corresponding load characteristic and the corresponding value of Vout fundamental are shown in the lower plots. The equation with pulling value will be added.

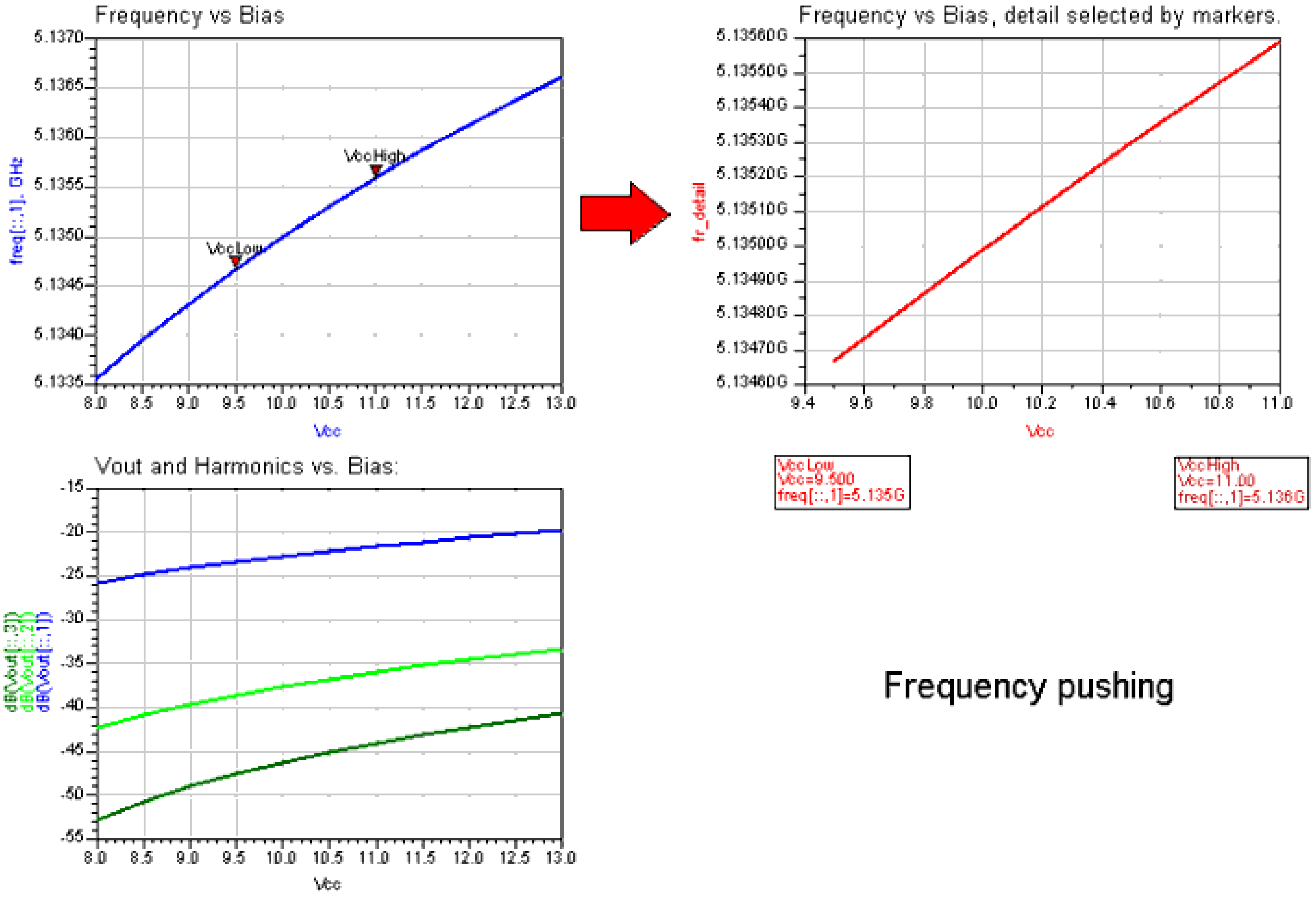


Results with Circuit Modified for Frequency Pulling

Frequency Pushing

The following image shows results after the frequency pull circuit is modified for frequency push, with fixed load (*vswr=1, phi =0*) and varying bias on oscillator's transistor Vcc. The display presents frequency variation with Vcc.

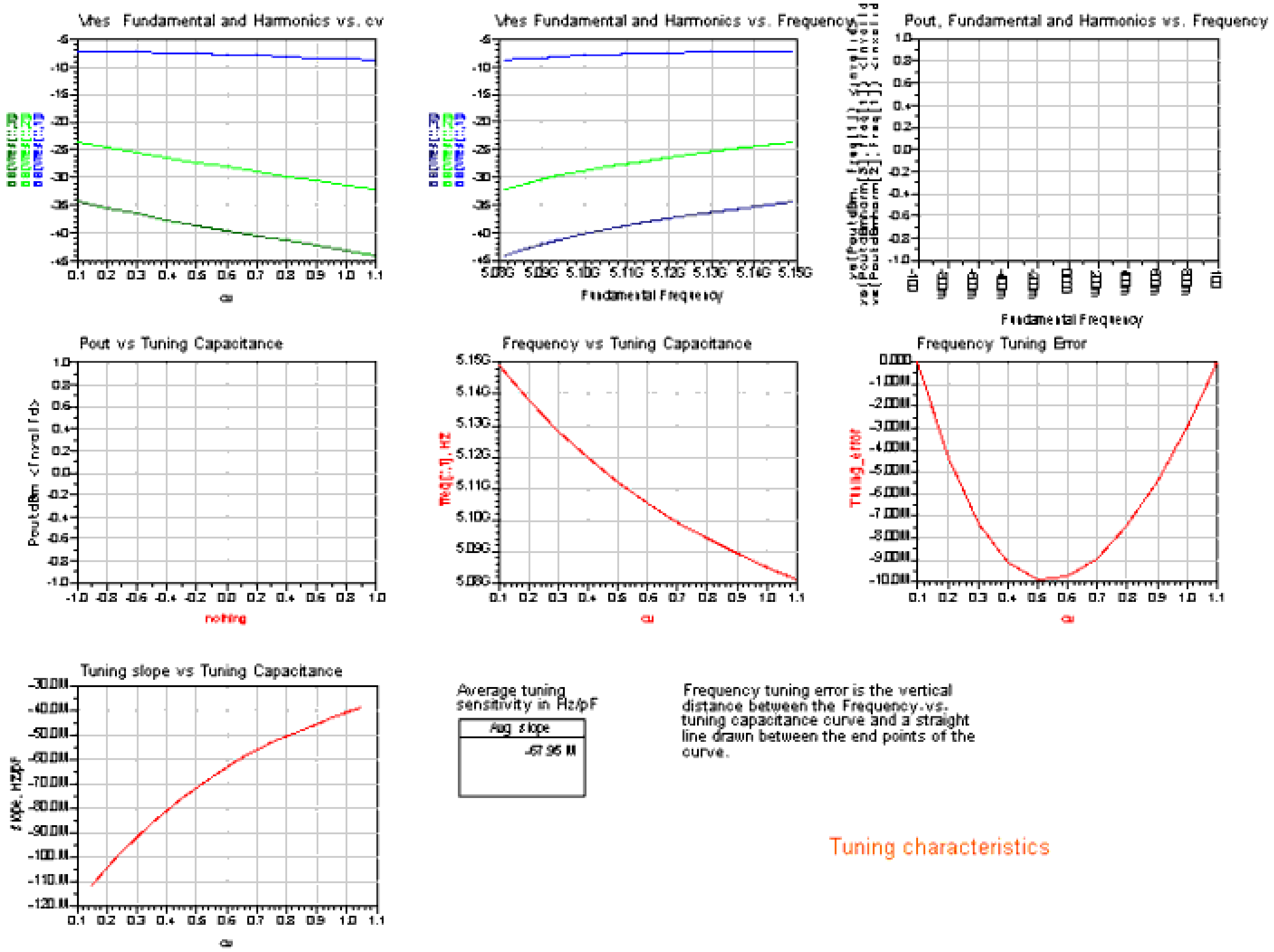
For Vcc=8V, the circuit does not oscillate, which results in the error message. Nevertheless, the sweep is performed, showing oscillations for higher bias. Two markers on the plot allow us to zoom in at the frequency plot. The plot to the right is determined by markers position. The corresponding value of Vout fundamental is shown in the lower plot.



Results with Circuit Modified for Frequency Pushing

Tuned Oscillations

The following image shows the resonator voltage and its harmonics. It provides the tuning characteristics of sweeps vs. capacitance vs. frequency.



Average tuning sensitivity in Hz/pF

Frequency tuning error is the vertical distance between the Frequency-vs-tuning capacitance curve and a straight line drawn between the end points of the curve.

Resonator Voltage and Harmonics

Linear Design Tools

The designs used for linear applications provide tools to investigate oscillations conditions. They belong to two groups, as follows:

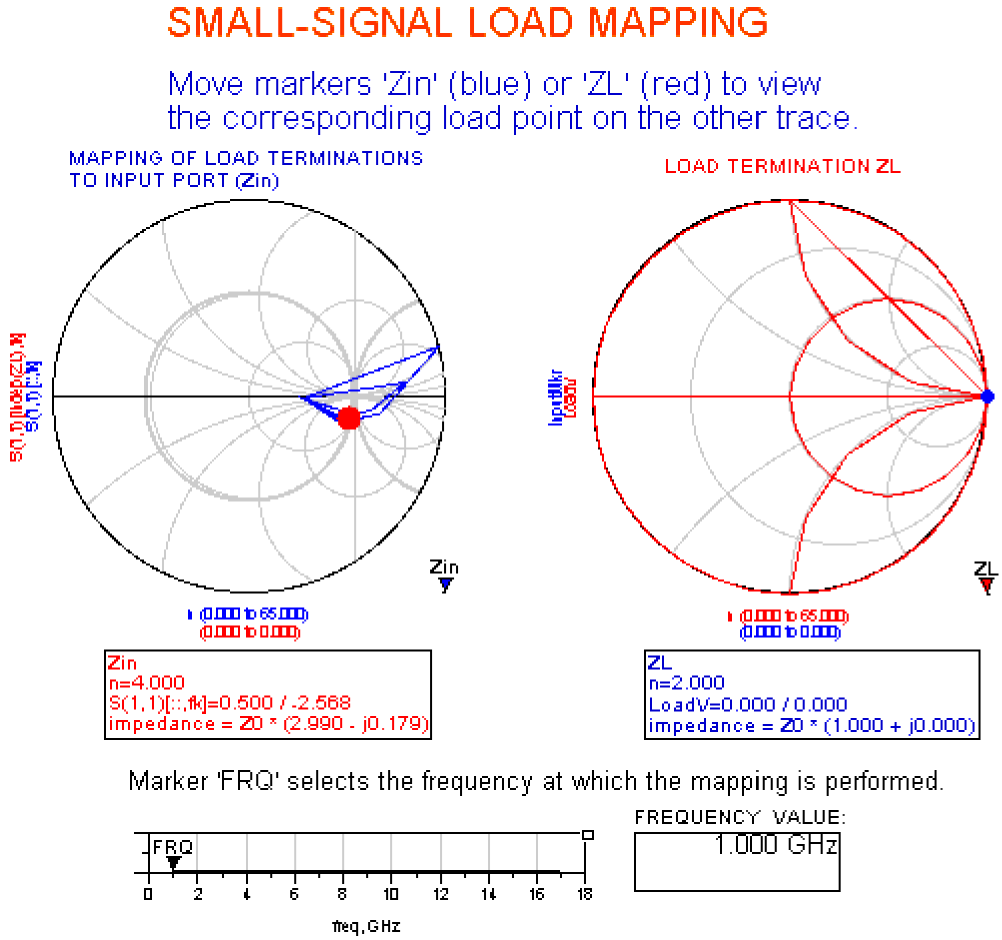
- Necessary oscillation conditions
- Check for Nyquist stability criterion, using the linearized version of the OscPort component, called OscTest.

Load-to-Resonator Mapping

Load-to-resonator mapping is represented by the design *MapLoad* and the graph *MapLoad.dds*. The design consists of the oscillator circuit without the resonator. The buffer and load are replaced by varying load. The load values correspond to main traces on a Smith chart (shown in the *loadmap.dds* display). You can specify the number of samples per trace and the radius of the small circle. On the resonator side, the circuit is terminated by an S-parameter port. S-parameter analysis is performed over frequency band determined that you specify so that the mapping can be analyzed at various frequencies.

The purpose of the analysis is to observe how the different values of the load will be detected by the resonator. The values that map outside the unit circle are of particular interest. These are the values that will provide negative resistance facing the resonator, so that the necessary oscillations conditions will be satisfied.

The graph *MapLoad.dds*, as shown in the following image, represents load values and their image in the resonator plane. Color-coded markers facilitate orientation. The marker on the bottom plot selects frequency at which the mapping is performed.



SMALL-SIGNAL LOAD MAPPING

Move markers 'Zin' (blue) or 'ZL' (red) to view the corresponding load point on the other trace.

Zin
n=4.000
S(1,1)[::,fk]=0.500 / -2.568
impedance = Z0 * (2.990 - j0.179)

ZL
n=2.000
LoadV=0.000 / 0.000
impedance = Z0 * (1.000 + j0.000)

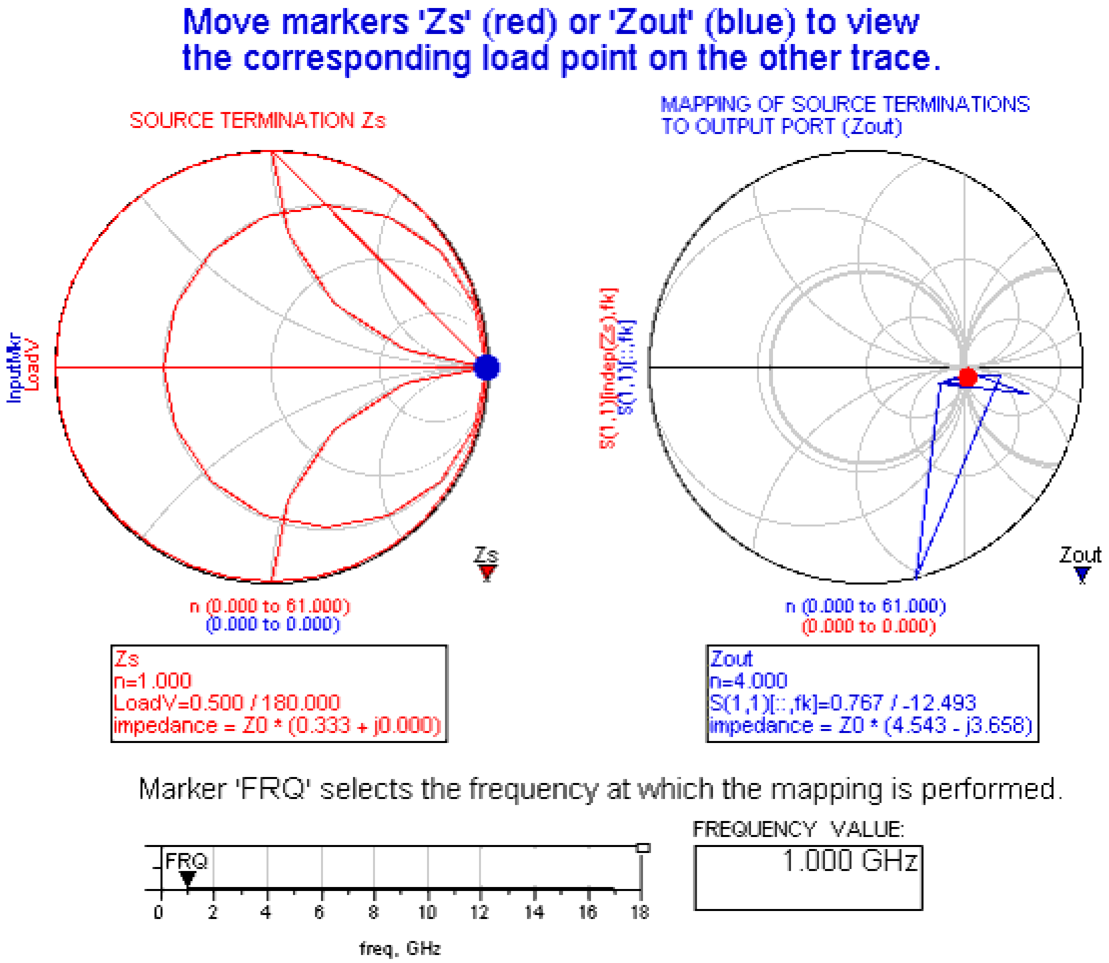
Marker 'FRQ' selects the frequency at which the mapping is performed.

FREQUENCY VALUE:
1.000 GHz

Load Values and Their Image in Resonator Plane

Resonator-to-Load Mapping

Resonator-to-load mapping is represented by the design *MapInput* and the graph *MapInput.dds*, as shown in the following image. The design is dual to load-to-resonator mapping, and it determines the image of the resonator at the buffer amplifier input. Consequently, it is useful in designing of the amplifier matching circuit. A special case restricted to a unit circle input gave rise to the method of stability circles (see Reference 1 in the Bibliography).



Move markers 'Zs' (red) or 'Zout' (blue) to view the corresponding load point on the other trace.

Zs
n=1.000
LoadV=0.500 / 180.000
impedance = Z0 * (0.333 + j0.000)

Zout
n=4.000
S(1,1)[::,fk]=0.767 / -12.493
impedance = Z0 * (4.543 - j3.658)

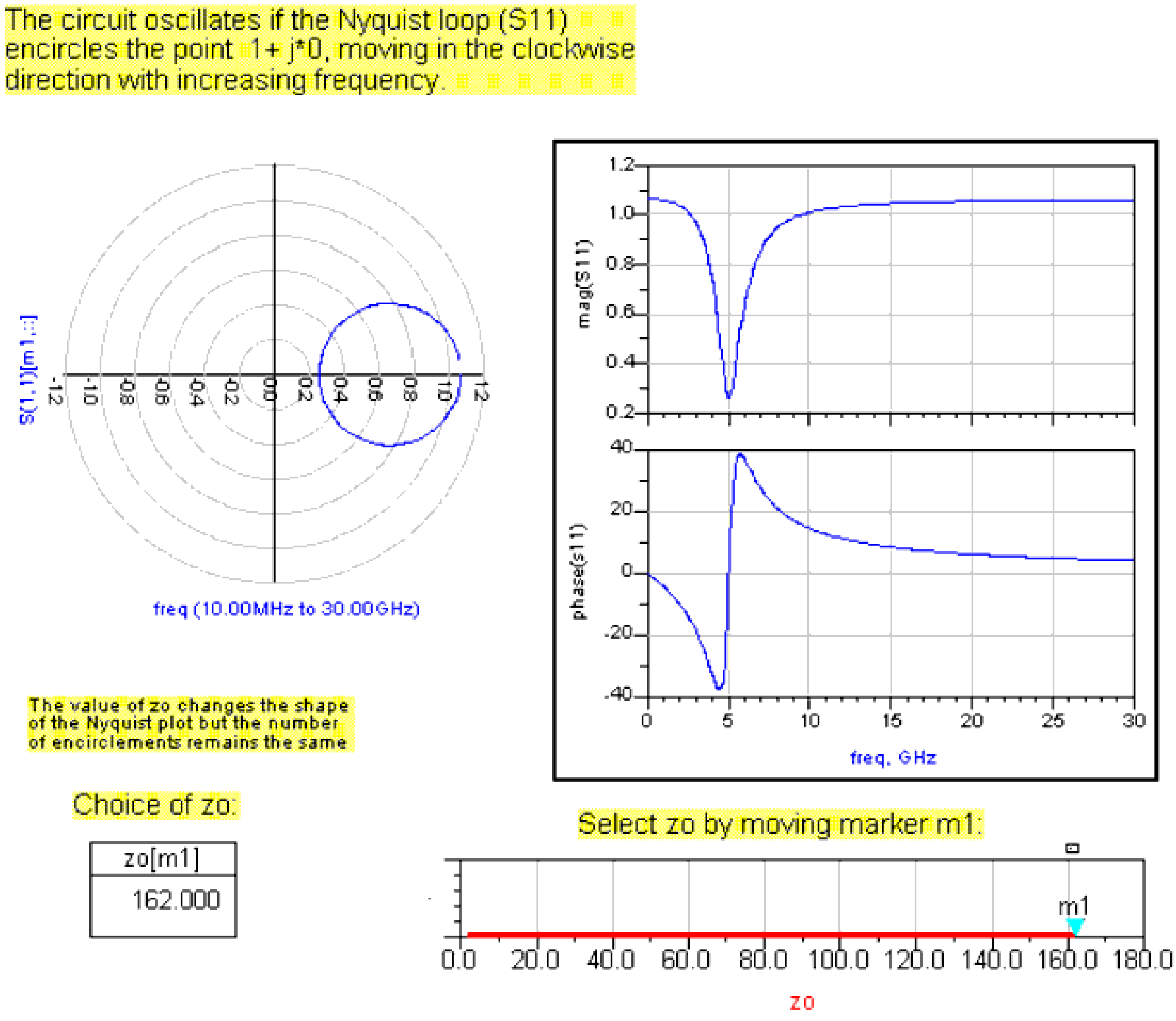
Marker 'FRQ' selects the frequency at which the mapping is performed.

FREQUENCY VALUE:
1.000 GHz

Resonator-to-Load Mapping

Stability Via Nyquist Plots

The design Nyquist Stab shows the use of the OscTest component. The results shown for different choices of the OscTest characteristic impedance Zo illustrate the importance of the Nyquist plot. Zo is swept from 1.5ohm to 21.5 ohm in 10 steps. The plots clearly show that it is the encirclement of *1+j0* that matters (as we know from the Nyquist theorem) and not the value of S11 at the crossing of the real axis. The justification of this statement is illustrated by two simple designs (NyqStab, NyqStabA) described in the next section.



The circuit oscillates if the Nyquist loop (S11) encircles the point 1+ j*0, moving in the clockwise direction with increasing frequency.

The value of zo changes the shape of the Nyquist plot but the number of encirclements remains the same

Choice of zo:

zo[m1]
162.000

Select zo by moving marker m1:

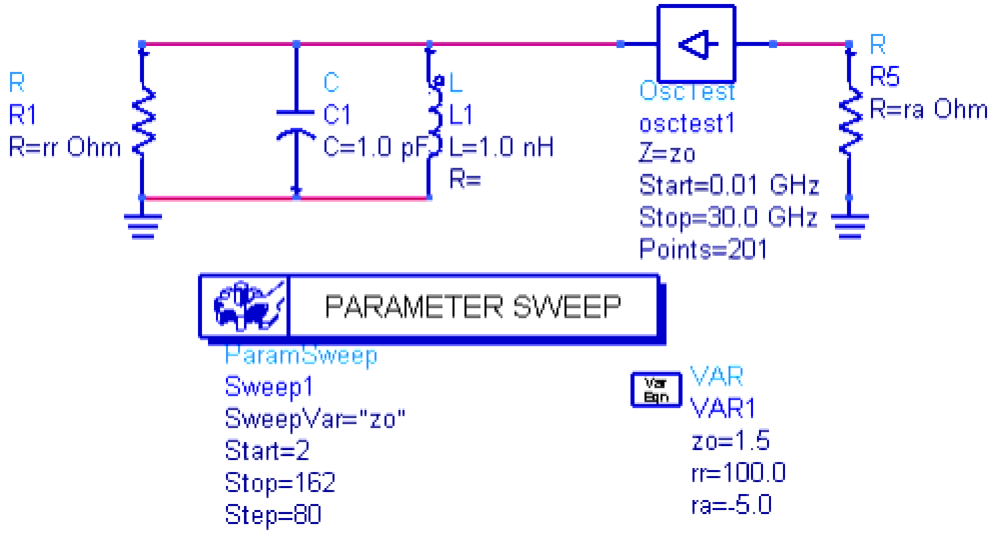
Resonator-to-Load Mapping

Theory of Stability vs. Nyquist Plots

There is a widespread belief [3,7,9,10,11,12,13,14,15] that the stability of oscillators can be determined by a particular criterion. When the phase of the transfer function is zero and the magnitude (at the same frequency) is larger than one, the system is unstable. The circuit shown in the criterion is usually presented as two equations:

`arg(Sn) = -arg(Sr), | Sn Sr | > 1`

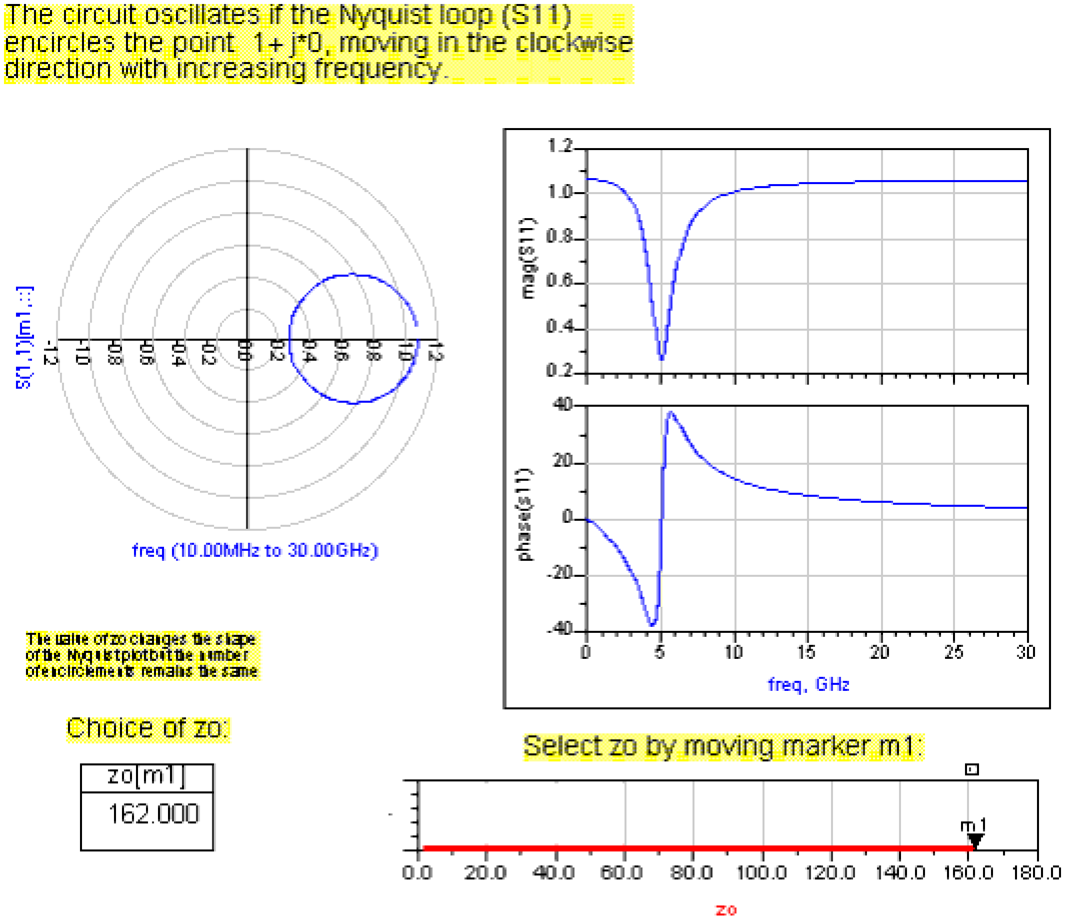
Consider a simplest possible linearized oscillator, as shown in the following image. The circuit has the resonator's resistance *rr=1/G = 100.0* ohm and the active (linearized) resistance *ra = 1/g'(Vo) = -5.0* ohm and is obviously unstable.



Stability Via Nyquist Plots

The Nyquist plots and equation were checked for different values of the characteristic impedance Zo. The system stability depends on the position of poles of the transfer function *Sr(s)Sn(s)*. If the function possesses poles in the right half plane, the system is unstable. It follows from the Nyquist criterion that the presence of poles in the right-half plane and the system instability can be determined by the encirclements of point *1+j0* by the osctest generated contour *S11 = Sr(jw)Sn(jw)*.

The Nyquist plots obtained for *Zo = 2.0, 82.0, 162.0* ohm are shown in the following image.



The circuit oscillates if the Nyquist loop (S11) encircles the point 1+j*0, moving in the clockwise direction with increasing frequency.

The value of zo changes the shape of the Nyquist plot but the number of encirclements remains the same

Choice of zo:

zo[m1]
162.000

Select zo by moving marker m1:

Nyquist Plot for Simple Linearized Oscillator

The three values of Zo are chosen so that we get respectively:

`Zo < |ra| < rr, |ra| < Zo < rr`

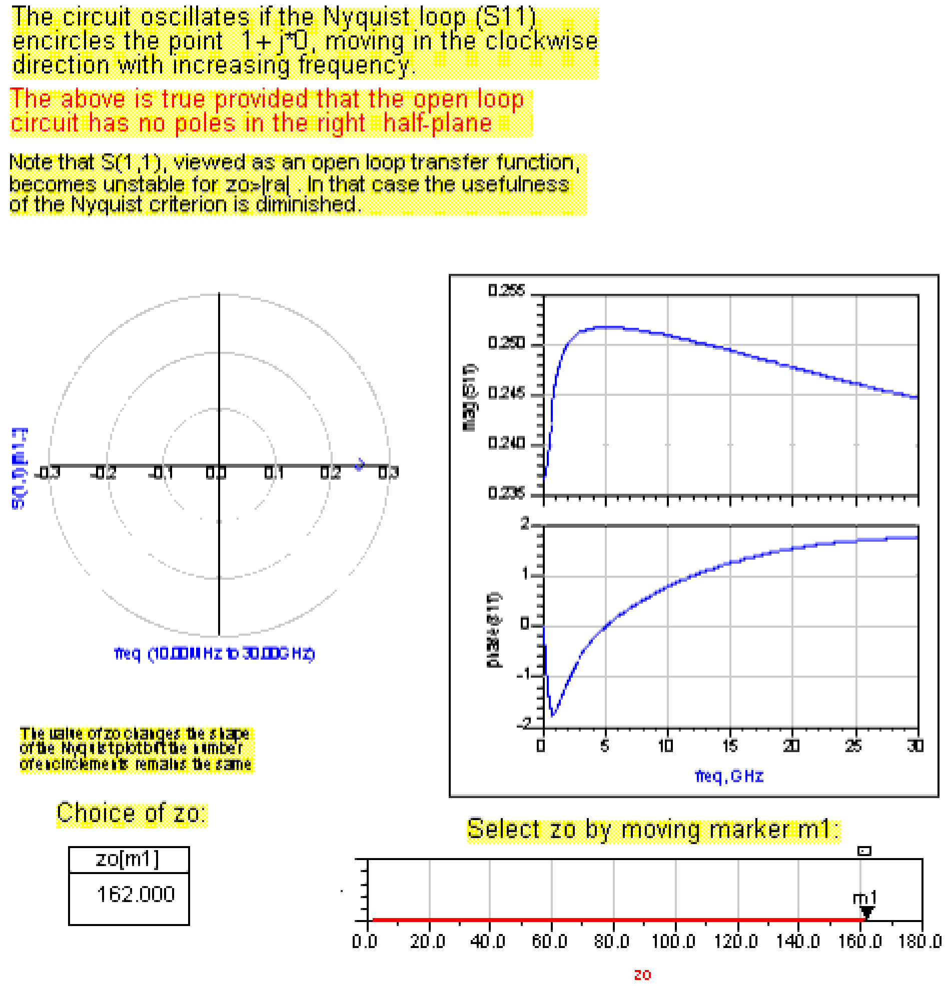
and

`|ra| < rr < Zo`

The circuit is obviously unstable. Consequently, the Nyquist loop, shown in the plots on the left in the following image, encircles the *point 1+j0* for every value of Zo. However, if we turn to the magnitude-phase plots (shown to the right), then the circuit instability will be hard to deduce. Finally the intuitive condition *(1)(SnSr >1 for arg(SnSr)= 0)* obviously fails for *Zo = 82.0 and Zo=162.0* ohm.

The *SrSn* contours clearly show that it is the encirclement of *1+j0* that matters (as we know from Nyquist theorem) and not the value of *SnSr* at the crossing of the real axis.

The Nyquist criterion is most useful when the open loop system is stable. In that case, the stability of the feedback system is determined by the closing of the feedback loop. The following image shows a variation of our original circuit.



The circuit oscillates if the Nyquist loop (S11) encircles the point 1+j*0, moving in the clockwise direction with increasing frequency.

The above is true provided that the open loop circuit has no poles in the right half-plane

Note that S(1,1), viewed as an open loop transfer function, becomes unstable for zo>|ra| . In that case the usefulness of the Nyquist criterion is diminished.

The value of zo changes the shape of the Nyquist plot but the number of encirclements remains the same

Choice of zo:

zo[m1]
162.000

Select zo by moving marker m1:

Nyquist Plot for Simple Circuit

In the circuit shown, the resistances were interchanged, resulting in an active resonator, for which adding the 100 ohms in parallel (i.e. loop closing) does not change its instability. Obviously, in this system, the Nyquist loop does not encircle the *1+j0* . This is because the open loop transfer function $Yn\ Zr = (s/rrC)/(s^2 + s/raC + 1/LC)$ has two poles in the right half plane and loop closing does not add any new poles.

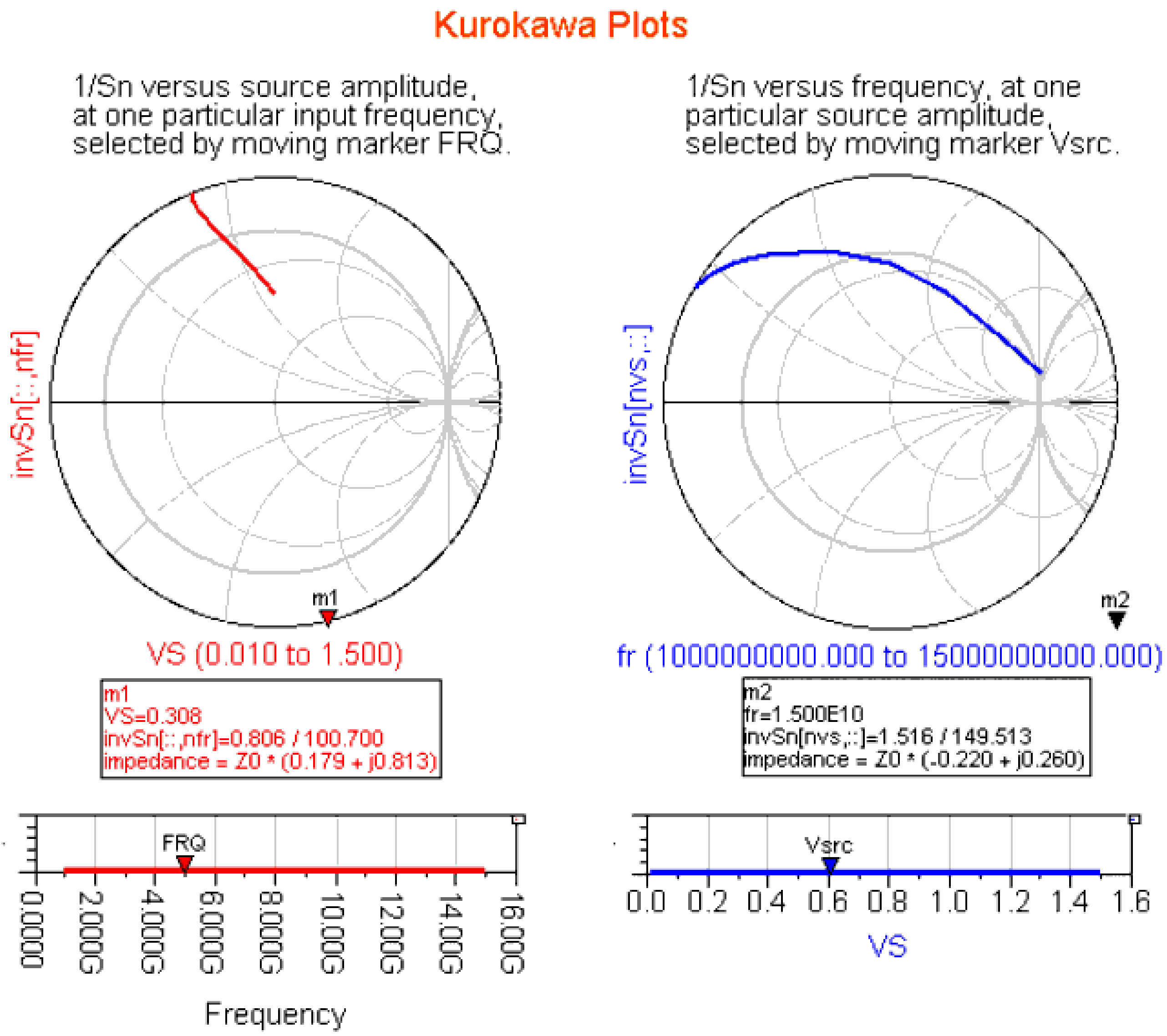
Therefore the position of the OscTest probe (which automatically computes *SrSn* in the simulator) should be carefully chosen. It should be placed between the resonator and the active circuit so that the open-loop system is stable.

The plots of S11 for *Zo = 2.0,82.0,162.0* need to be considered.For *Zo = 82.0, 162.0* the Nyquist loop does not encircle *1+j0,* as expected. However, for *Zo=2.0* it does, which seems contrary to the fact that the resonator circuit is active. The explanation for this is that the open loop S-parameter transfer function:

$Sr(s) = (Zr-Zo)/(Zr + Zo) = - (s^2 + s((1/ra)-(1/Zo))/C+1/LC)/(s^2 + s((1/ra)+(1/Zo))/C+1/LC)$

has all poles in the left hand plane for *Zo<|ra|*. Only closing the loop makes the system unstable.

Using Nonlinear Design Tools



Large-Signal S-Parameter Design

In the steady-state, because of intrinsically nonlinear behavior of oscillators, signals are no longer sinusoidal. Consequently, the concepts of impedance and S-parameters are not obvious. However, in high-Q circuits, in which signals are represented by their fundamental components, there is a natural way to define the large-signal impedance, or large-signal S-parameters. In this section, we define the large signal S-parameters and demonstrate that the equations

$$\arg(Sn) = - \arg(Sr)$$

$$| Sn\ Sr | = 1$$

determine amplitude and phase of oscillations.

The steady state periodic oscillations can be represented by their Fourier series:

$$v(t) = \Sigma |vn| \cos(n\omega t + jn) \quad i(t)= \Sigma |in| \cos(n\omega t + \psi n)$$

For a high-Q resonator the higher harmonics are negligibly small and voltage and current can be approximated by:

$v(t) \approx |V| \cos(\omega t + j)$

$i(t) \approx |I| \cos(\omega t + \psi)$

where

$V = |V| \exp(jj)$, and $I = |I| \exp(j\psi)$

denote the fundamental components of voltage and current.

Thus the signals are represented by their complex amplitudes V and I, for which we define the large-signal incident and reflected waves:

$a = (V + Zo \times I)/(2 \sqrt{Zo})$, $b = (V - Zo \times I)/(2 \sqrt{Zo})$

On the resonator side, we have $a = Sr\, b$, with $b = b(a)$ on the active circuit side. These two relationships provide us with the steady-state equations $a = Sr\, b(a)$. After defining the large signal S-parameter: $Sn = b(a)/a$ the steady state equation can be represented as $a = Sr\, Sn\, a$, which leads to: $1 = Sr\, Sn$, which is equivalent to the equations.

Solving Harmonic Balance Convergence Problems

Harmonic balance simulation in Advanced Design System is an excellent way to analyze many oscillators in the frequency domain. Occasionally, you might have an oscillator that converges in a time-domain simulation, but the harmonic balance oscillator algorithm is unable to find the solution. There are two techniques in ADS for solving those oscillators:

- Analyzing the large-signal loop gain in harmonic balance to find the point of oscillation and using that as an initial guess for the full harmonic balance oscillator analysis
- Using a transient analysis to produce an initial guess for harmonic balance oscillator analysis.

In the DesignGuide > Oscillator menu, select *Solving Harmonic Balance Conversion Problems* for several useful design examples.

For more information on these techniques, refer to *Simulation Techniques for Recalcitrant Oscillators* (cktsimhb).

Oscillator Core Examples

Oscillator Cores (cClappCore, cHartleyCore, cModifiedClappCore, and cModifiedColpittsCore) are compatible with simulation and measurement setups outlined by the Generic Oscillator Example. These core oscillator circuits are configured for low resistance loads. (50ohm is used for these four oscillator cores, although other load values are possible.) The design and display filenames for these examples follow a naming convention that indicates oscillator type and simulation setup as follows:

- OscillatorTypeSimulationType
- OscillatorTypeSimulationType.dds

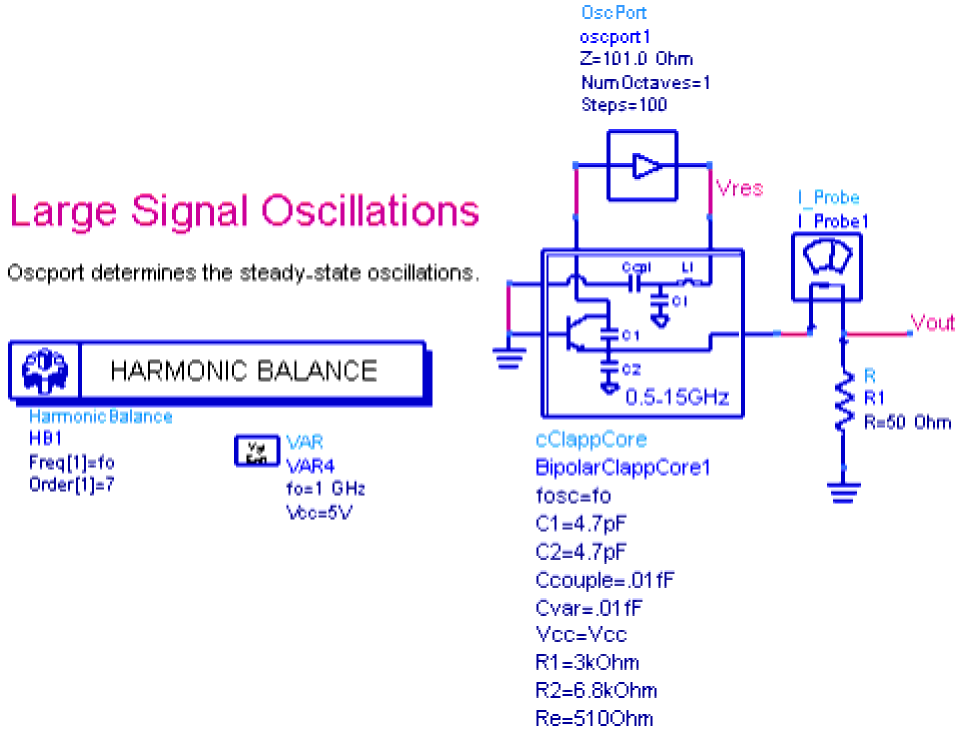
where:

OscillatorType indicates one of the following topologies: Clapp, Hartley, Modified Clapp, or Modified Colpitts.

SimulationType provides one of the following simulation set-ups or measurements: FixedFreqOsc, FreqPull, FreqPush, FreqTune, LSSpar, MapInput, MapOutput, NyqStab, or Phase Noise.

For example, the simulation that determines oscillator frequency of a Clapp oscillator has the design filename *ClappFixedFreqOsc* and a data display filename *ClappFixedFreqOsc.dds*. This simulation predicts oscillation frequency, output power, and calculates tank components *Lt* and *Ct*. The accompanying data display presents these results.

The *cClappCore* circuit is used to illustrate available simulation setups offered in the design guide.

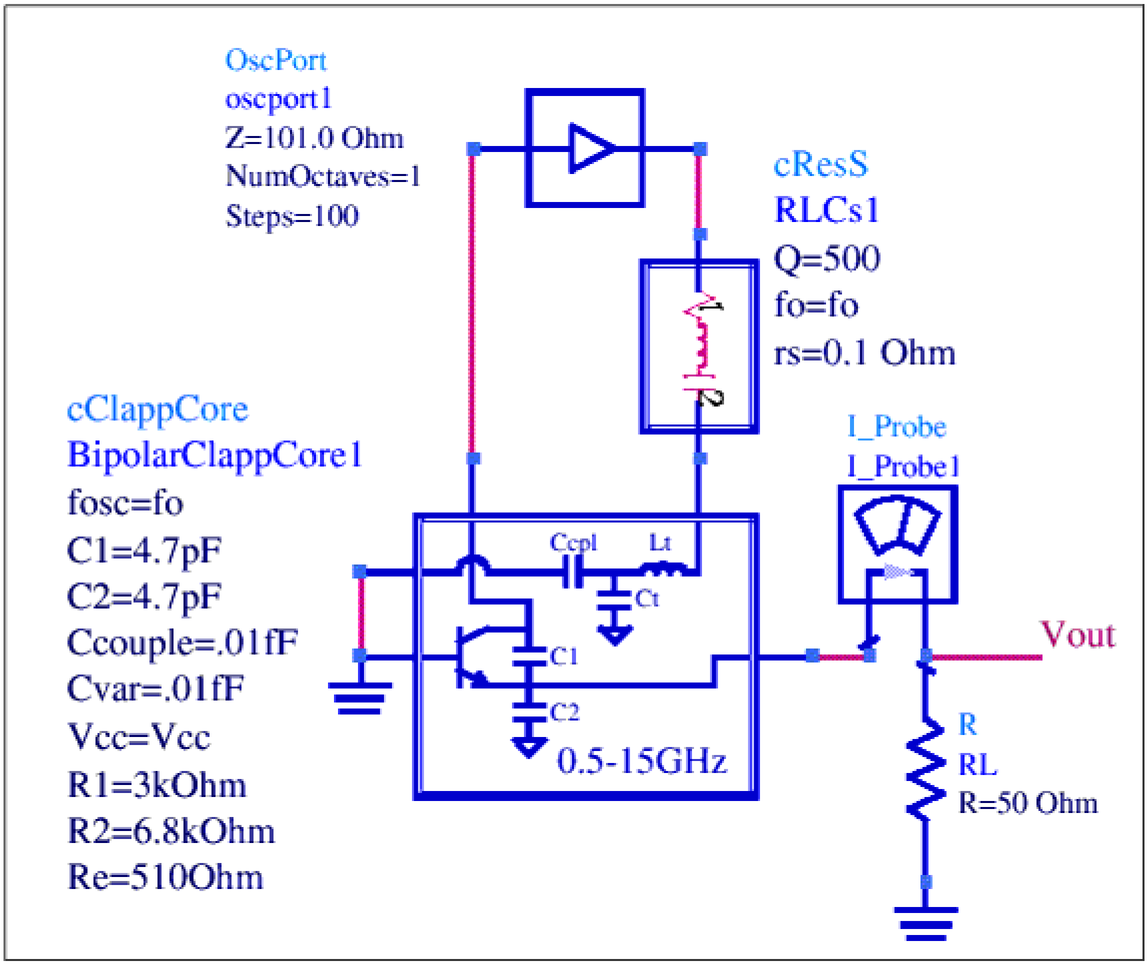


Oscillator Core Circuit Simulation Connections

The preceding image shows the Clapp oscillator harmonic balance simulation ClappFixedFreqOsc that determines oscillation frequency, output power and tank component values at 1GHz. The *OscPort* probe is shown connected between the active device and tank resonator. Resonator tank components *Lt* and *Ct* are computed and shown on the companion data display page.

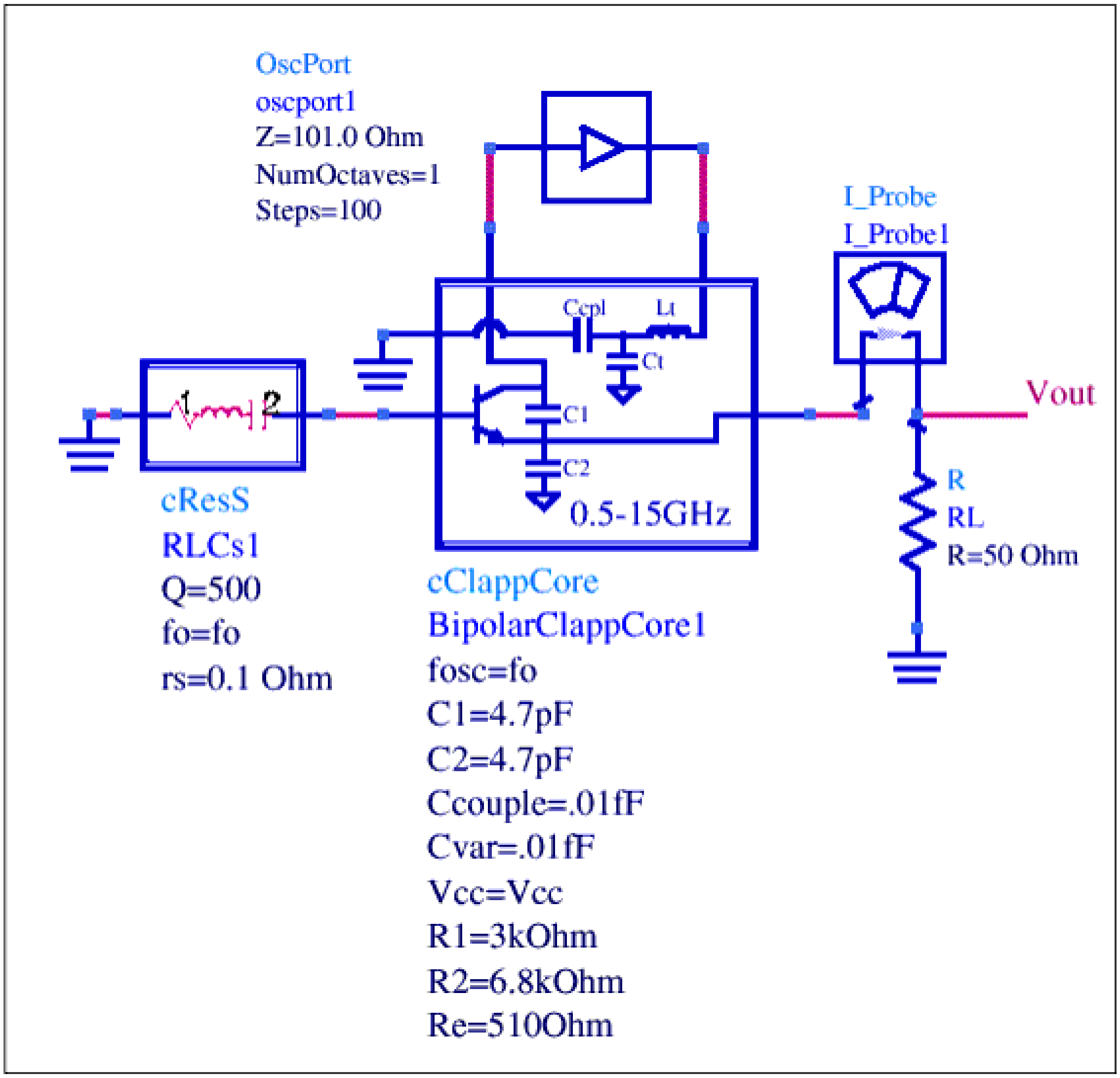
Oscillator Core Series Resonator Connection Alternatives

Oscillator core circuits (cClappCore, cHartleyCore, cModifiedClappCore, and cModifiedColpittsCore) are compatible with series resonators in two ways.



Series Resonator Connection Between Tank and Active Circuit

The first possibility shown in the preceding image connects the series resonator between the tank circuit and active device. If resonator losses are low, the series resonator can also be connected between the bipolar base terminal and ground as shown in the following image.



Series Resonator Connection Transistor Base and Ground

Additional Examples

Following are examples in addition to the primary example, as described in Generic Oscillator Example.

The design and display filenames for these examples follow the generic oscillator naming convention with 3-letter prefixes attached to the generic names, as follows:

- xxxgenericoscillator name
- xxxgenericoscillator name.dds

where xxx stands for one of: saw, vco, xto, or yto.
For example, VCO Large Signal S-Parameters have the filenames *vcoLSSpar* and *vcoLSSpar.dds*.

Crystal Oscillator (XTO)

These oscillators are notable for their high frequency stability and low cost. Typical structure is that of a Colpitts oscillator with quartz crystal resonator introduced into feedback path. Mechanical vibrations of the crystal stabilize the oscillations frequency. Vibration frequency is sensitive to temperature. Therefore, temperature compensation circuits are often used to improve frequency stability. Crystal resonators are typically used in the range up to 100 MHz (to a few hundreds of MHz if resonating on overtones).

SAW Resonator Oscillator (SAW)

Principle of operation is similar to that of crystal oscillator with the quartz resonator replaced by a Surface Acoustic Wave oscillator. SAW resonators are used in frequency range up to 2 GHz.

Voltage Controlled Oscillator (VCO)

In any of the preceding structures, frequency tuning can be provided by adding a varactor diode to the resonator. The varactor diode serves as a voltage controlled capacitor. It has very fast tuning speed (GHz/nsec) and low Q. Consequently, the varactor can be used with LC elements to provide wide tuning (with poor frequency stability) or with a crystal, SAW or DRO resonator for narrow tuning with better frequency stability.

At microwave frequencies, the device capacitances become significant, resulting in a different (often simpler) circuit. The operation principles, remain the same.

YIG Tuned Oscillator (YTO)

For a very wide band (that can reach decade) tuning with high frequency stability and for frequency range of 1 GHz to 50 GHz. YIG (Yttrium-Iron-Garnett) resonators are used. The YIG sphere behaves like a resonator with 1000-to-8000 unloaded Q resulting in very good frequency stability. The resonator are tunable over wide bandwidth with excellent linearity (~0.05%). For fine tuning (for phase-lock), or frequency modulation an FM coil can be added.

Bibliography

✓ **Note**
The EESof Application website contains additional reference material for oscillator
http://eesof.tm.agilent.com/applications/oscillators-b.html

1. G.R.Basawapatna, R.B.Stancliff, "A Unified Approach to the Design of Wide-Band Microwave Solid State Oscillators," IEEE Trans. MTT-27, May 1979, pp 379-385.
2. Fundamentals of Quartz Oscillators" HP Application Note 2002.
3. R. T. Jackson, "Criteria for the Onset of Oscillations in Microwave Circuits," IEEE Trans. MTT-40, Mar 1992, pp 566-568, "Comments on ``Criteria...,'' IEEE Trans. MTT-40, Sept 1992, pp1850-1851.
4. A. P. S. Khanna, "Oscillators" in Chapter 9 of "Microwave Solid State Circuit Design" by I. Bahl P. Bhartia, J.Wiley 1988.
5. Khanna and Obregon - "Microwave Oscillator Analysis" - IEEE Trans.vol. MTT-29, June 1981, pp 606-607.

6. M.Odyniec, "Nonlinear Synchronized LC Oscillators, Theory and Simulation," IEEE Transactions, MTT-41, May 1993, pp 774-780.
7. M.Odyniec, "Benchmarks for Nonlinear Simulation," APMC'94, Tokyo, Japan, Dec. 6-9 1994.
8. M.Odyniec, "Stability Criteria via S-parameters," EUMC'95, Bologna, Italy, Sept. 4-8, 1995.
9. V.Rizzoli, F. Mastri, and D, Maroti, "General noise analysis," IEEE Trans. MTT-37, Vol.1, No. 1, January 1989.
10. C. Schiebold, "Getting Back to the Basics of Oscillator Design," Microwave Journal, May 1998.
11. J.C. Slater, "Microwave Electronics," Van Nostrand, Princeton, 1950.
12. "S-parameter Design," HP Application Note 159 .
13. A. Sweet, "MIC and MMIC Amplifier and Oscillator Circuit Design," Artech House, Boston 1990.
14. D. Warren et al., "Large and Small Signal Oscillator Analysis," Microwave Journal, May 1989.
15. J.J. D'Azzo, C. H. Houpis, "Feedback Control Systems," McGraw-Hill New York 1960.
16. J. W. Boyles, "The Oscillator as a Reflection Amplifier..." Microwave Journal, June 1986.
17. J.L.J. Martin and F.J.O.Gonzalez, "Accurate Linear Oscillator Analysis and Design," Microwave Journal, June 1996.
18. Z. Nativ and Y. Shur, "Push-Push VCO Design with CAD Tools. "
19. Microwave Journal, February 1989.
20. M. Odyniec, "Stability Criteria via S-parameters "European Microwave Conference, Bologna, Italy, September 1995.
21. A. Przedpelski, "Simple, Low Cost UHF VTOs "RF Design," May 1993.
22. N. Ratier et al. "Automatic Formal Derivation of the Oscillation Condition", IEEE Int. Frequency Control Symp. 1997.
23. T. Razban et al.,"A Compact Oscillator..." Microwave Journal, February 1994.
24. S. Savaria and P. Champagne, "Linear Simulators..." Microwave Journal, May 1995.

Selected Papers

1. Basawapatna and Stancliff - "A Unified Approach to the Design of Wideband Microwave Solid State Oscillators" - IEEE trans. vol MTT-27, May 1979, pp 379-385.
2. Khanna and Obregon - "Microwave Oscillator Analysis" - IEEE Trans. vol. MTT-29, June 1981, pp 606-607.
3. M.Odyniec, "Nonlinear Synchronized LC Oscillators, Theory and Simulation," IEEE Transactions, MTT-41, May 1993, pp 774-780.
4. M.Odyniec, "Benchmarks for Nonlinear Simulation," APMC'94, Tokyo, Japan, Dec. 6-9 1994.
5. M.Odyniec, "Stability Criteria via S-parameters," EUMC'95, Bologna, Italy, Sept. 4-8, 1995.