



Agilent EEsof EDA

Overview on Low-Power Mixer Design Example using ADS

This document is owned by Agilent Technologies, but is no longer kept current and may contain obsolete or inaccurate references. We regret any inconvenience this may cause. For the latest information on Agilent's line of EEsof electronic design automation (EDA) products and services, please go to:

www.agilent.com/find/eesof

Low-Power Mixer Design Example using HP Advanced Design System

Technical Note

Introduction

This note describes a method for designing a low-power single-transistor active mixer using HP EEsof Advanced Design System (ADS). It includes details on the design steps, simulation setups and data displays. The project file is available in the ADS example directory */examples/RF_Board/MixerPager_prj*. Topics covered include:

Page

2	Device selection
2	Device model DC verification (use of swept-parameter DC analysis, importing measured data)
4	Bias network resistor design (DC analysis, use of equation in data display page)
6	Device model RF verification (DC back annotation, S-parameter simulation)
7	Device model large-signal verification (swept-parameter harmonic balance simulation, S-parameter simulation)
8	Matching circuit design (use of equation-based 1-port Z-parameters)
11	Mixer conversion gain versus LO drive level
14	Mixer conversion gain versus RF signal level
14	Creating the mixer layout
16	Simulation from layout

Note on Running examples projects:

Unix users must copy example projects to their local working directory to be able run simulations. In addition, the example described in this note uses a bipolar junction transistor (BJT) from the ADS RF Transistor Library and surface-mount technology (SMT) passive components from the RF Passive SMT Library. You must have these libraries installed to be able to run simulations for this example, although you can view and edit the schematics and data displays in any case.

Circuit Specifications

The mixer is a upper-sideband downconverter, with an RF of 900 MHz RF, and a 45 MHz IF. The simplified specifications supplied for this design call for it provide 10dB conversion gain, operating from a 1 volt DC supply at 600uA current. This very low power consumption is typical of applications such as pagers and cellular phones, where battery lifetime is critical. Low cost is another driving factor in such applications. Other typical specifications a mixer would have to meet in a “real-world” design, such as linearity, port-to-port isolation, spurious response and noise figure, are not included in this particular example. See */examples/RFIC/Mixers_prj* for examples of how to include these simulations in your design.

Device Selection

One of the first steps in the design process is to select the device. The device used for this example is the Motorola MMBR941, a bipolar junction transistor (BJT) packaged in a standard SOT-23 plastic package. While bipolar devices do not generally have as good mixing properties as field-effect transistors, the low operating voltage precludes using FETs in this case. The chosen device has acceptable performance for this application, and offers several other advantages: it is extremely low cost, and accurate models are readily available. Since a general rule of thumb in high-volume, low-cost applications is to use the least expensive device that will accomplish the job, the MMBR941 is a good choice for this mixer. It is equally true that, no matter how good a device is, if there are no models to simulate it with, it becomes impossible to use in a design.

The device model, taken from the ADS RF Transistor Library, is a Gummel-Poon model where the parameters were extracted by the manufacturer, Motorola. Initially, the model's DC performance is verified by comparing DC I-V curves. Next, a bias network will be designed to establish the desired operating point. The model's RF behavior will then be checked by comparing the simulated S-parameters with measured S-parameters taken at the same bias conditions. Finally, the model's nonlinear performance is verified by simulating gain compression and comparing to measured results.

Device Model DC Verification (DC_curves.dsn)

DC_curves.dsn (see Figure 1) shows one way to set up a swept-parameter DC analysis. The DC voltage supply at the collector is set to a variable, VCE, which is initialized in the VAR block. The VAR block also initializes the variable, IBB, used in the DC current source at the base of the BJT. The actual values used for VCE are determined in the DC simulation controller (DC1). In this example, VCE is swept from 0V to 6V, so that the model can be verified over a relatively wide operating range. The DC controller can only sweep a single variable, so the values for IBB are swept using the ParamSweep component. The range chosen for the base current, IBB, is set to 50uA to 350uA.

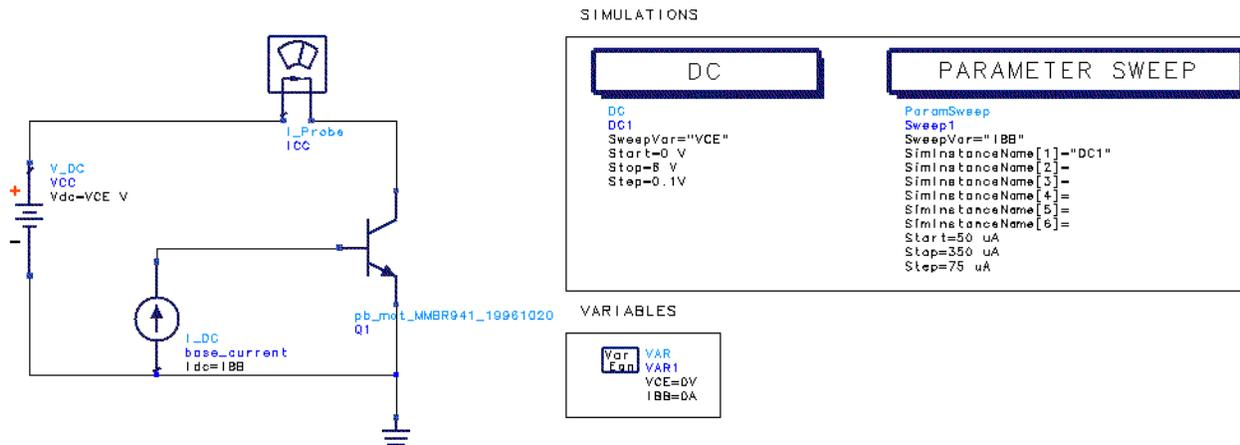


Figure 1. Transistor Swept-Parameter DC Analysis Setup

Results from this simulation are displayed in “DC_curves.dds”. The available output variables can be viewed by either placing a new plot or selecting the existing plot for edit, which open the **Insert Plot** dialog window shown in Figure 2. Notice that voltages at each of the named nodes are automatically supplied, as is the current at the DC supply (VCC.i). The data from the current probe, ICC.i, is redundant in this case. The numbered nodes are used to store information for DC back annotation, discussed in the section on Device Model RF Verification (page 6).

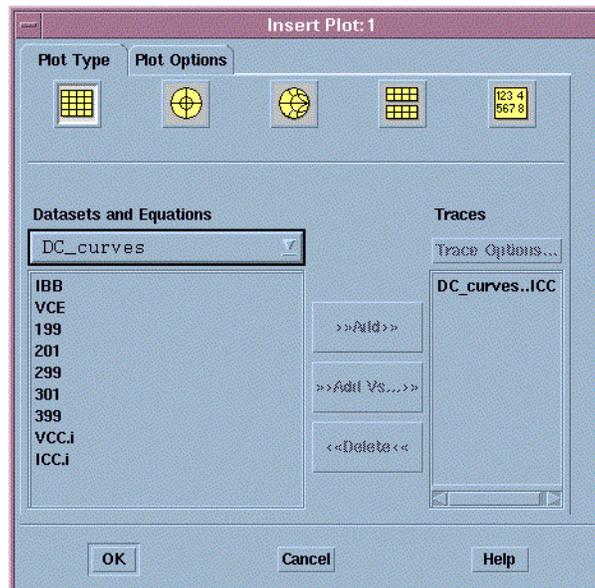


Figure 2. Insert Plot Dialog Box

Figure 3 shows there is good agreement between simulated and measured results. Measured data may be read in to ADS from either datafiles or instruments by selecting **Window>New File/Instrument Server**. ADS will convert files in Touchstone, MDIF, Citifile or ICCAP formats to ADS datasets, which can then be displayed alongside

simulated results. The I-V curves clearly show that, at the specified operating point of $V_{CE}=1V$, $I_{CE}<0.6mA$, the device will be operating in a very low current regime.

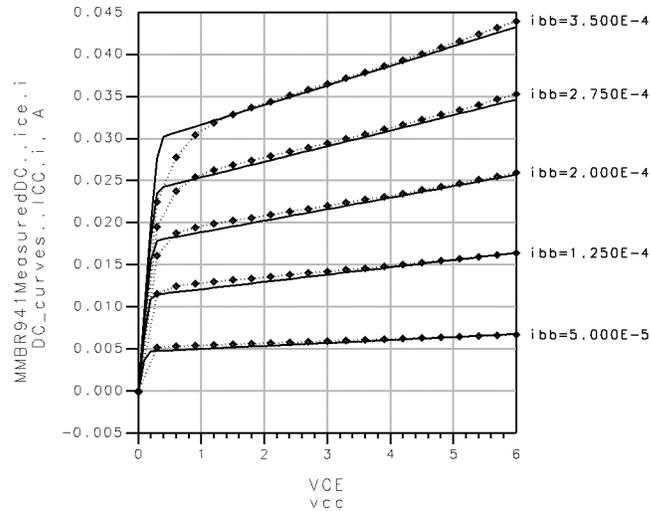


Figure 3. Comparison of Measured (symbols) and Simulated (Solid line) DC I-V Curves for MMBR941

Bias Network Design (BiasPoint.dsn)

The next step, selecting the device operating point and calculating the required bias resistors, is done using the set-up in BiasPoint.dsn as shown in Figure 4. Since the collector voltage and current have been specified, only the base current needs to be determined. In the schematic, V_{CC} is fixed at 1V and I_{BB} is swept from 1uA to 10uA, using the DC controller.

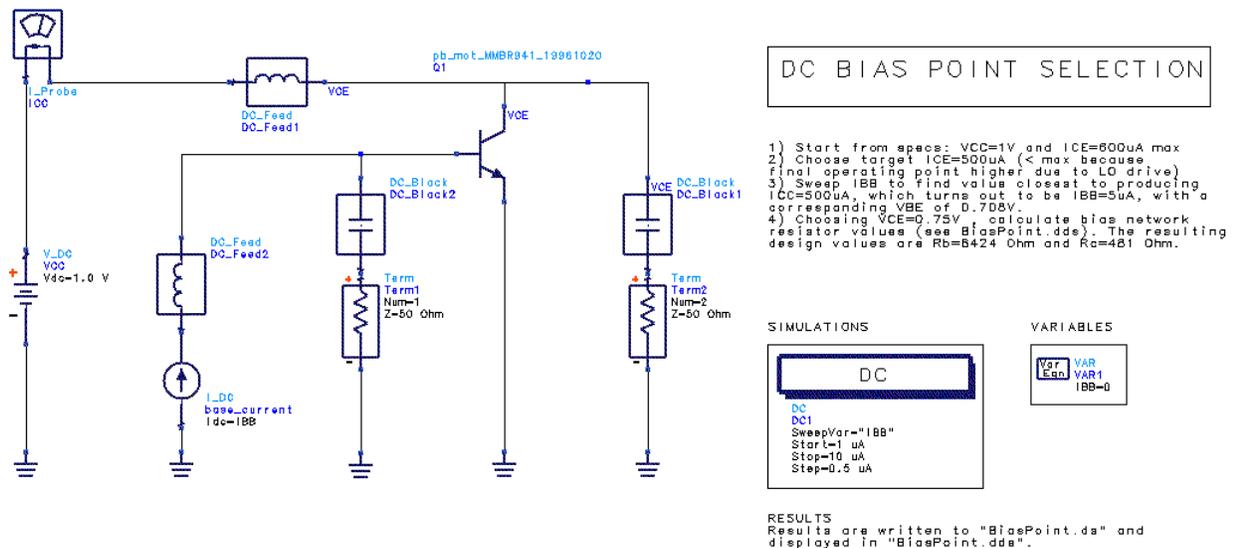


Figure 4. Calculation of Bias Point

The schematic contains bias tee components (the DC_Feed and DC_Block components) and 50 Ω terminations that mimic the actual test setup used to measure the device. However, since the DC simulation does not include any RF signals, they are not necessary at this point, and can be omitted without changing the results.

The results are displayed in tabular format in BiasPoint.dds (see Figure 5), so the appropriate base current can be selected. Note that the bias point current is actually lower than the specified final value. This is because the device will be pumped with a relatively large LO signal, causing a shift in the DC component of the collector current. This shift will be calculated more precisely later on but, for now, IBB is selected to be 5uA so that the corresponding collector current (514uA) is well below the specification.

Table 1: ICE and VBE vs. IBB, VCC=1V

IBB	...Point...ICC.i	BiasPoint...VBE
1.000E-6	102.707uA	665.470mV
1.500E-6	154.533uA	676.186mV
2.000E-6	206.251uA	683.768mV
2.500E-6	257.865uA	689.642mV
3.000E-6	309.381uA	694.438mV
3.500E-6	360.808uA	698.493mV
4.000E-6	412.152uA	702.006mV
4.500E-6	463.420uA	705.105mV
5.000E-6	514.615uA	707.879mV
5.500E-6	565.743uA	710.389mV
6.000E-6	616.807uA	712.682mV

Figure 5. Device Operating Point Selection

The bias resistor values, shown in Figure 6, are calculated next. Base current, collector current and VCC are known, but the designer must make an assumption about the voltage drop across Rc to be able to solve for Rc and Rb. In this case, a collector-emitter voltage of 0.75V is chosen, providing a reasonable working voltage at the output and realizable resistor values. The equations, written in the data display page, calculate the exact values required for each value of base current, but of course the nearest standard values must be chosen. The next step is to confirm bias operation using these standard values and then verify the S-parameters of the model against measured values.

Table 2: Calculated Rb and Rc

IBB	Rb	Rc
1.000E-6	84529.635	2410.638
1.500E-6	49209.429	1602.222
2.000E-6	33115.916	1200.472
2.500E-6	24143.233	960.191
3.000E-6	18520.552	800.304
3.500E-6	14716.285	686.232
4.000E-6	11998.583	600.741
4.500E-6	9976.677	534.280
5.000E-6	8424.296	481.125
5.500E-6	7202.032	437.642
6.000E-6	6219.680	401.408

$$Eqn Rb = (0.75 - VBE) / IBB$$

$$Eqn Rc = (0.25) / (ICC.i + IBB)$$

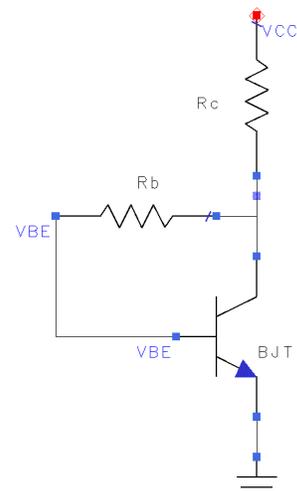
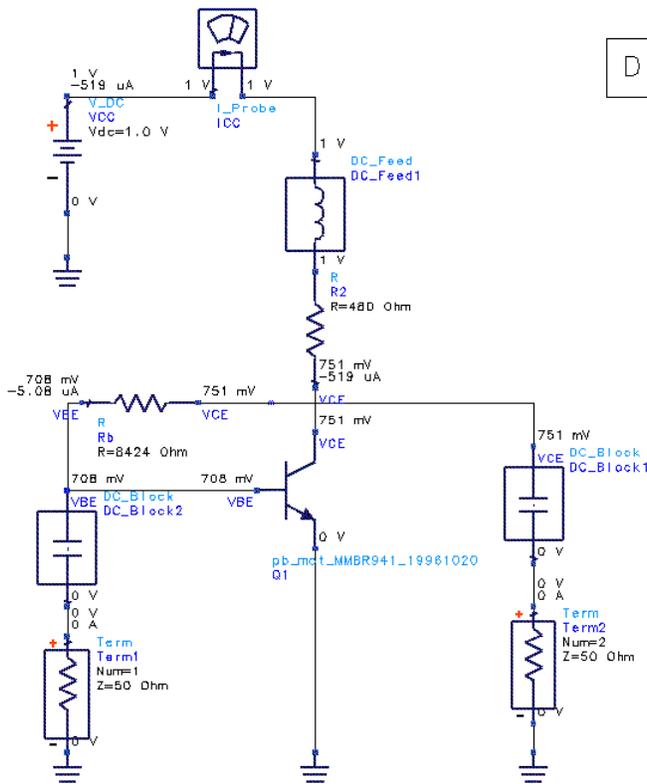


Figure 6. Calculation of Bias Network Resistors



DC BIAS NETWORK VERIFICATION

- 1) Compare operating point of device using exact calculated values of R_c and R_b from BiasPoint.dsn: $I_{CC}=519\mu A$, $V_{BE}=0.708V$. This can be done by choosing "Annotate DC Solution" from the "Simulate" menu once simulation is finished.
- 2) Compare operating point of device using closest standard resistor values. For $R_c=470\ \Omega$, $R_b=8200\ \Omega$: $I_{CC}=529\mu A$, $V_{BE}=0.709V$.
- 3) Calculate S-parameters.

SIMULATIONS

DC	S PARAMETERS
DC DC1	S_Param SP1 Start=50 MHz Stop=3 GHz Step=10 MHz

RESULTS
 Results are written to "Sparams.ds" and displayed in "Sparams.dds".

Figure 7. DC and S-Parameter Simulation Setup

Device Model RF Verification (BiasNet.dsn)

BiasNet.dsn, shown in Figure 7, includes both DC and S-parameter simulations so, in this case, bias tee components (DC feeds and blocks) are required to ensure proper RF performance. DC results are displayed directly on the schematic page, using the DC back annotation feature: once the simulation has been run, select **Simulate>Annotate DC Solution** to see the DC voltages and currents at each node. This simulation can be done with both the exact resistor values and nearest standard values ($R_c=470\ \Omega$, $R_b=8.2\ k\Omega$) to confirm that the operating point is correct.

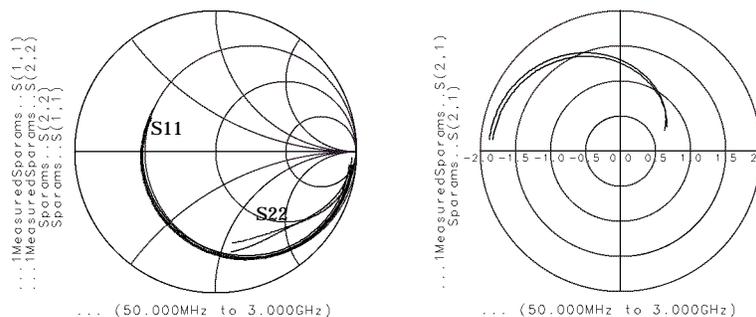


Figure 8. Comparison of Measured and Simulated S-Parameters for MMBR941

The device S-parameters are calculated at this operating point and displayed, together with measured data, in Figure 8. The good agreement obtained here verifies the small-

signal RF performance. The device compression point will be simulated next to confirm large-signal operation.

Device Model Large-Signal Verification (Compression.dsn)

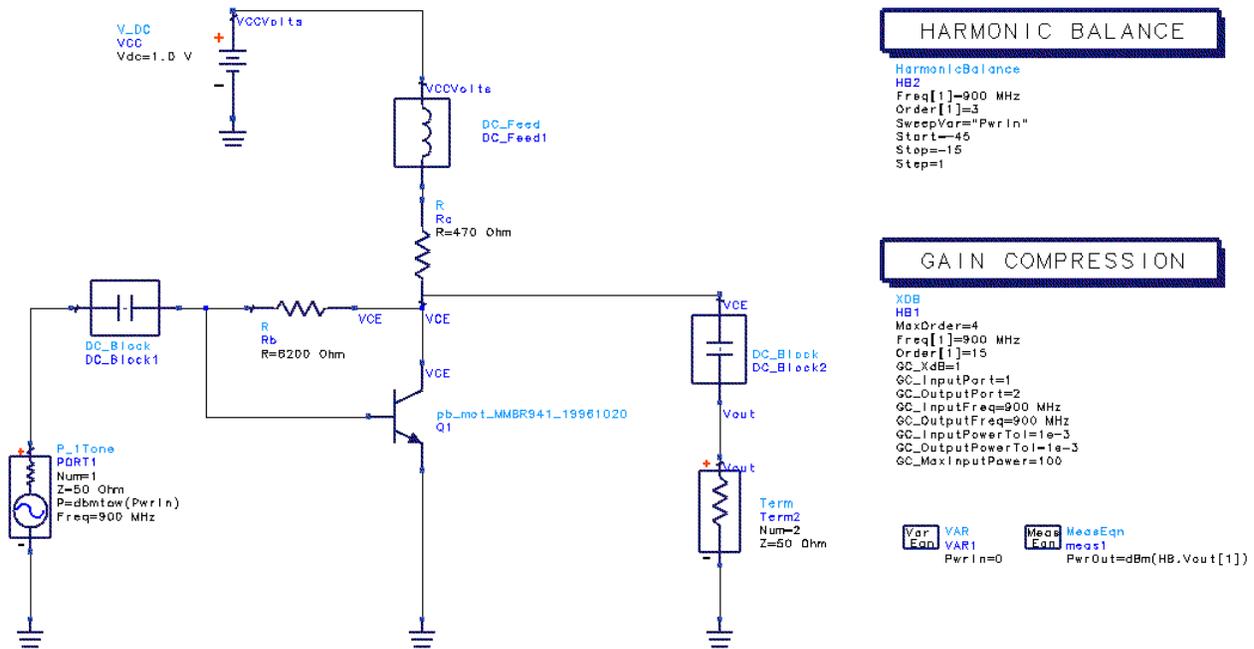
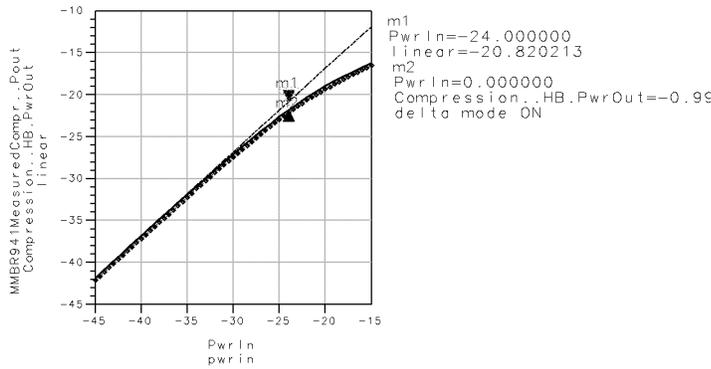


Figure 9. Device Compression Measurement Setup

Compression.dsn (Figure 9) shows two ways of calculating the device output compression at the RF frequency of 900 MHz. The conventional way, implemented here with the Harmonic Balance controller, is to sweep the input power level from low (i.e. small-signal) to high values until the output power compresses (the ratio P_{out}/P_{in} starts to fall off from its small-signal value). The input power variable, "PwrIn" is swept from -45 to -15dBm and a Measurement Equation component is used to define the output power at 900 MHz, in dBm. Notice that the dBm function assumes the power is being delivered to a 50 Ω load, unless otherwise specified by the user. The argument of the function, "HB.Vout[1]", specifies the fundamental frequency. Figure 10 shows the equation and graph used to determine the 1dB compression point, and includes the measured results as well.

The second method, unique to ADS, is more direct and does not require graphs or sweeping variables. The Gain Compression controller "XDB" performs a harmonic balance analysis that directly calculates and outputs the input and output power levels at the specified compression point. The default setting is 1dB, but the user can specify any amount of compression. Figure 10 also shows the output from this method: the input and output power levels at 1dB compression are listed in dBm.

Method 1: Pin versus Pout



Eqn | linear=Compression.HB2.HB2.HB.PwrIn+Gain[1]

Method 2: “XDB” simulation results:

inpwr[1]	outpwr[1]
-24.007	-21.816

Figure 10. Two Methods of Determining 1dB Compression - both show input P1dB is -24 dBm

ADS offers a great deal of flexibility in where and how output data are defined. To take a simple example, “PwrOut” has been defined on the schematic page using a MeasEqn component, but it could equally well have been defined on the data display page as an equation. An advantage of defining outputs on the schematic is that they can be used in optimizations. On the other hand, defining them on the data display page is useful for setting up templates (where complex calculations can be easily applied to many different schematics). Also, any outputs that were overlooked before the simulation was run can be calculated afterwards by adding them on the data display page.

Notice that, at this point, the design still uses ideal bias tee components to isolate the DC and RF signal paths. These will be replaced with the real components that make up the matching networks in the next stage of the design.

Mixer Matching Circuit Design (RFIFmatch1.dsn)

An important step in mixer design is determining what impedances are seen at each port for both the RF and IF. The finished input network will match the device base to 50 Ω at the RF and present a short circuit at the IF (to prevent any noise at the input being amplified and interfering with the IF at the output). Similarly, the output network will match the collector to 50 Ω at the IF, while presenting a short circuit to the RF. Thus, for each frequency, the terminations seen at the input and output of the device are completely different. Since the device is not unilateral, the presence of a short circuit on one side of the device will affect the impedance seen at the other side for matching purposes.

The first step in designing the input matching network, then, is to determine the device input impedance at the RF when the output is terminated in a short circuit. For the output matching network, the designer needs to know the BJT’s output impedance at the IF when the input is terminated in a short circuit. In ADS, equation-based 1-port Z-parameter components are used to simulate this sort of idealized frequency-dependent termination, as seen in RFIFmatch1.dsn (see Figure 11).

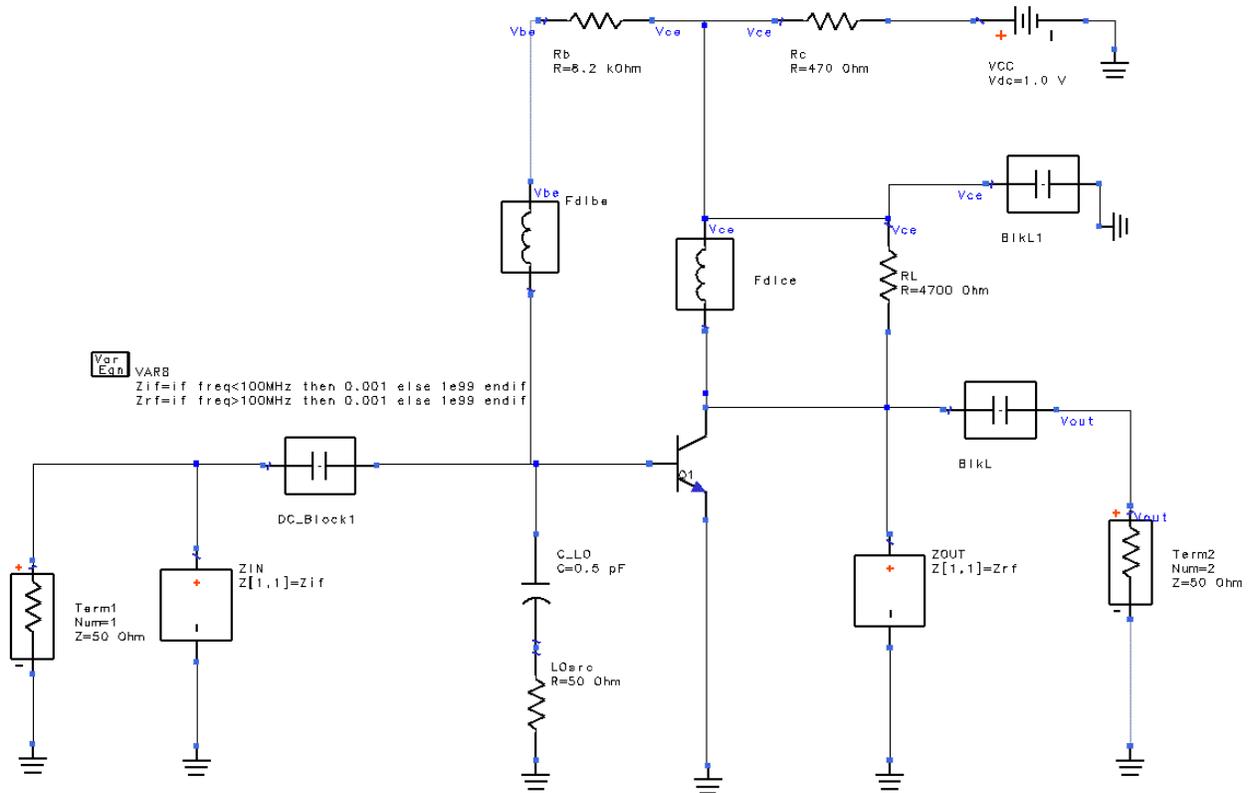


Figure 11. Calculating Device Impedance for Matching Network Design

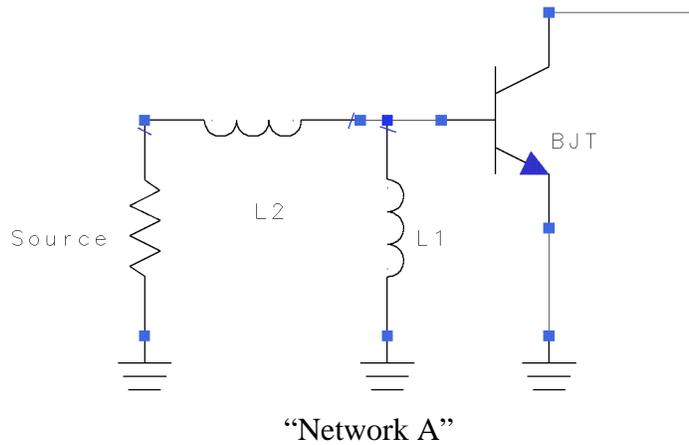
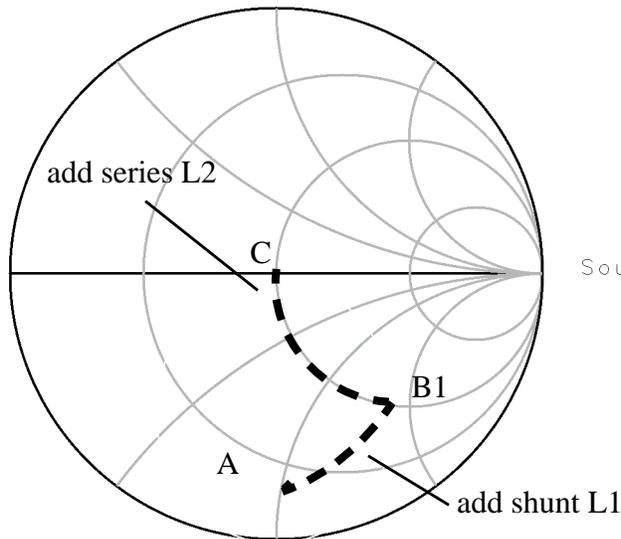
The Z1P_Eqn components are defined in a VAR block. The one at the input, ZIN, is set to be a short-circuit at the IF and an open at the RF. This provides the required termination for S22 at the IF, while leaving S11 unperturbed at the RF. Similarly, ZOUT, at the output, is set to be a short at the RF and an open at the IF. Notice also that the LO source is represented at this point as an ideal 50Ω termination, coupled to the mixer through a 0.5pF capacitor. The capacitor was chosen to be so small in order to isolate the LO source from the RF input signal. The return loss looking back through the capacitor towards the LO source is only 0.33dB at RF, so that it almost appears like an open circuit to the incoming RF signal. The penalty is that the LO, which is close in frequency to the RF, is also isolated from the circuit, meaning that a higher LO drive level is required. For example, when the LO source is set at -10dBm, only -22dBm reaches the mixer.

The resulting S-parameters at the RF show that the input impedance is $(11.5 - j51.4)\Omega$ with a short-circuit on the output. At the IF, the output impedance is $(2065 - j2010)\Omega$. These values can be used to decide on matching network topologies and component values. The designer always has several topologies to choose from in developing a matching network, and which one is best will depend on factors such maximizing yield (some topologies are more sensitive to component variation than others), minimizing component count (to reduce cost) and combining functions where possible (incorporating the bias decoupling components into the matching, in this case).

To illustrate, Figure 12 shows that, starting at the device input impedance (A), a shunt inductor followed by a series inductor will move the circuit impedance successively from B1 to 50 Ω. The resulting network “A” has some advantages: the shunt inductor will

provide a short to the IF at the input, as required, and it can be used in the bias decoupling network (to replace the ideal DC_feed). However, network “B” is even better: using a smaller value of shunt inductance brings the impedance to B2, where a match is achieved using a series capacitor. C2 can also serve as the DC blocking capacitor, thereby saving a component, so this network is used for the mixer.

S11 at 900MHz



S11 at 900MHz

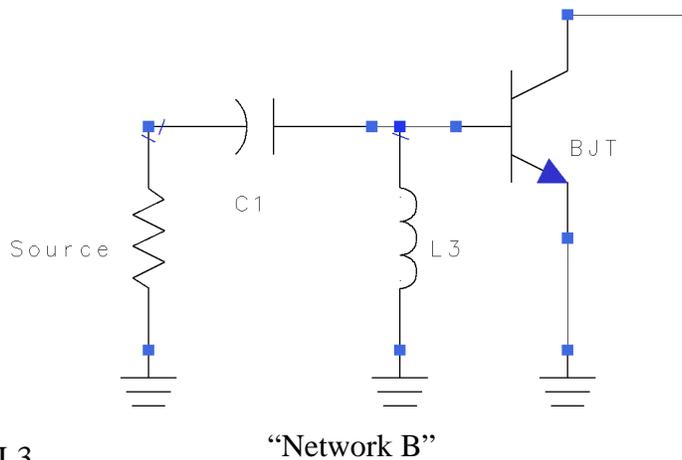
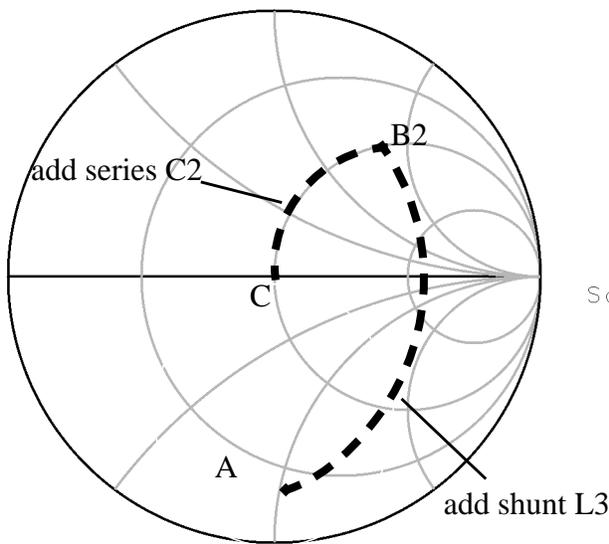


Figure 12. Choosing an Input Matching Network Topology

The output matching network was developed using a similar approach: starting from the Smith Chart, a matching network consisting of a shunt inductor followed by a series capacitor was designed. However, this topology would result in any RF at the output being dumped to the load instead of being short circuited as intended. To solve this, the shunt inductor (originally nearly 910nH, a very high impedance at RF) is replaced by an equivalent parallel LC combination. The capacitor must be large enough to provide a near-short for the RF, and a value of 33pF is chosen. The shunt inductor is then decreased, so the total reactance provided by the LC pair at the IF the same as that of the original inductor.

Although it was not done in this example, the actual component values for the network can be calculated using ADS, as illustrated in examples like */examples/MWCKts/LNA_1GHz_prj*. In this case, components were calculated manually from the Smith Chart, and the resulting circuit is shown in LOdrive.dsn. The final matching networks are shown in Figure 13. Notice that, in addition to components for the matching and bias networks, a load resistor, R_L has been added to control the mixer's conversion gain. The initial value of $4.7\text{k}\Omega$ was chosen to be high enough not to have an effect on the mixer's performance, and will be adjusted as required once the conversion gain is known. Also, two large RF bypass capacitors (BlkL1 and BlkL2) are added to provide RF ground to the output load resistor and inductor and to the input shunt inductor, respectively.

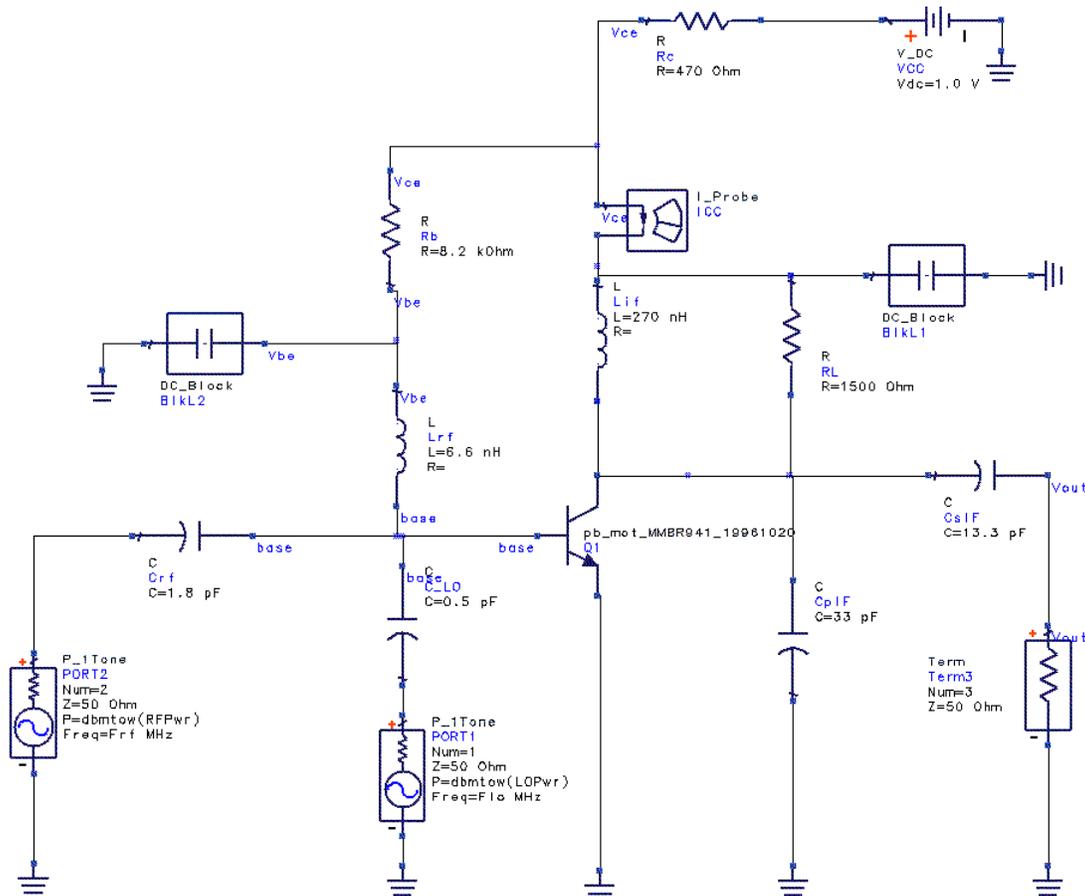


Figure 13. Mixer Matching Networks

Mixer Conversion Gain versus LO Drive Level (LOdrive.dsn)

LOdrive.dsn (Figure 14) shows how to simulate conversion gain for the mixer and how to determine the effect of LO drive level on gain and DC bias. The RF and LO frequencies and the LO power level have been defined as variables. The RF drive level is specified at -50dBm , while the harmonic balance controller is set up to sweep the LO drive level from -30m to -5dBm . (The controller has many parameters, and the user can control which are visible on the schematic by editing the component and choosing the "Display" page in the edit dialog window). A simulation measurement equation defines the output power, in dBm , at the IF. Defining it here instead of the data display page makes it possible to optimize for output IF power, if needed. The "mix" function will return the component of

the V_{out} spectrum defined by $\{-1, 1\}$, meaning $\{-Freq[1]+Freq[2]\}$ or $-LO+RF=IF$ (45MHz). The P_{IF} equation calculates the dBm value of the mix function.

LO DRIVE LEVEL AND RF/IF MATCH

These input and output matching networks were designed based on the results of RFIFmatch.dsn.
 The input match consists of a shunt L (Lrf) and series C (Crf). Several topologies are possible, but this one was chosen because: i) It provided the correct impedance of the IF (almost short-circuit); ii) It eliminates need for extra DC blocking components; iii) cost is a key driver in this design and capacitors are typically less expensive than inductors.
 The output match consists of Lrf, CpIF, CeIF. Again, the topology was chosen both to match the IF and the RF requirements.

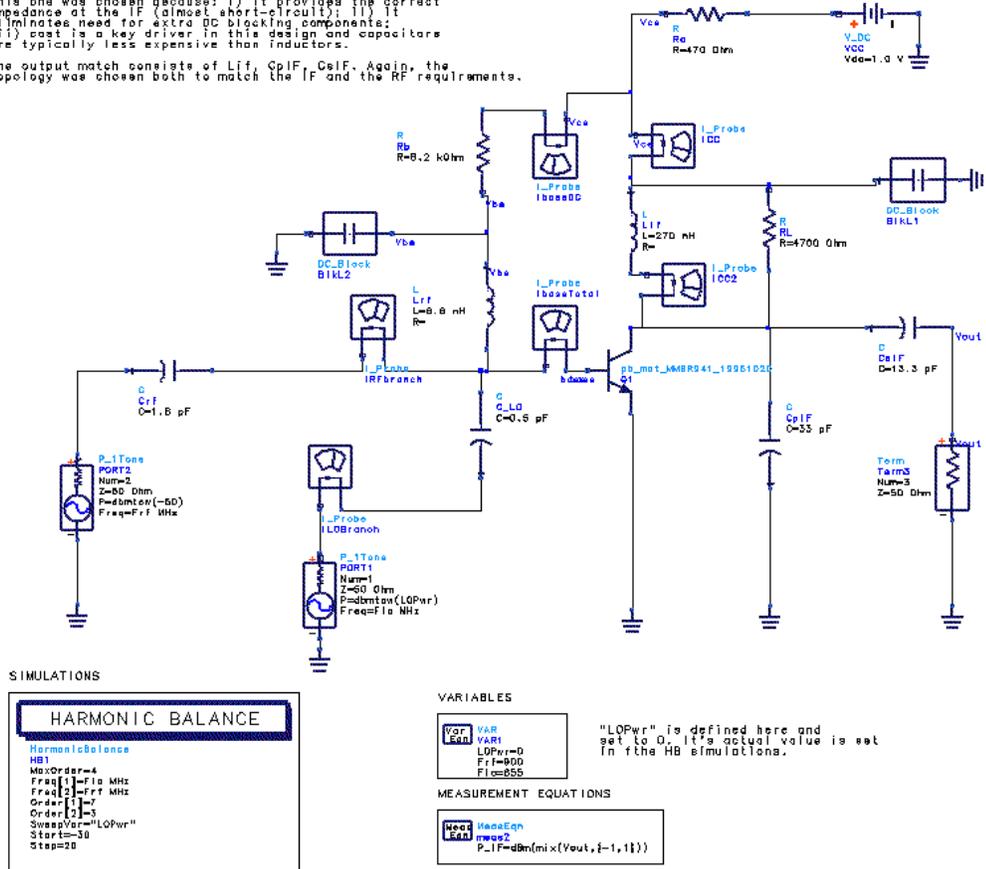


Figure 14. Set-up for Swept LO Drive Level Mixer Simulation

The data display shows the effect of the load resistor (Figure 15). Since the conversion gain is the difference between P_{IF} and the RF, and the RF power is fixed at -50dBm , the conversion gain can be calculated with a simple expression. Note that the default dataset, LOdrive, contains results for a $4.7\text{k}\Omega$ load resistor, and the conversion gain for this simulation is calculated by the equation "ConvGain". The conversion gain for a -10dBm LO drive is 17dB , which is unacceptably high. A second simulation was run with the load resistor reduced $1.5\text{k}\Omega$, which creates a lossy mismatch on the output. The results for that simulation were output to dataset LOdrive15, and equation "ConvGain_R15kOhm" shows the conversion gain is reduced to 13.7dB . This is still higher than the specification of 10dB , but will be left at this value for now since conversion gain can be expected to decrease further when non-ideal surface mount components replace the ideal components.

The second graph in the data display shown in Figure 15 illustrates the effect of the LO drive level on DC bias. Increasing the LO signal at the base drives the output swing on the collector harder, shifting the DC component higher (see Figure 16). In practice, a 5 to 15 percent shift in collector bias current typically gives good performance for a mixer of this type.

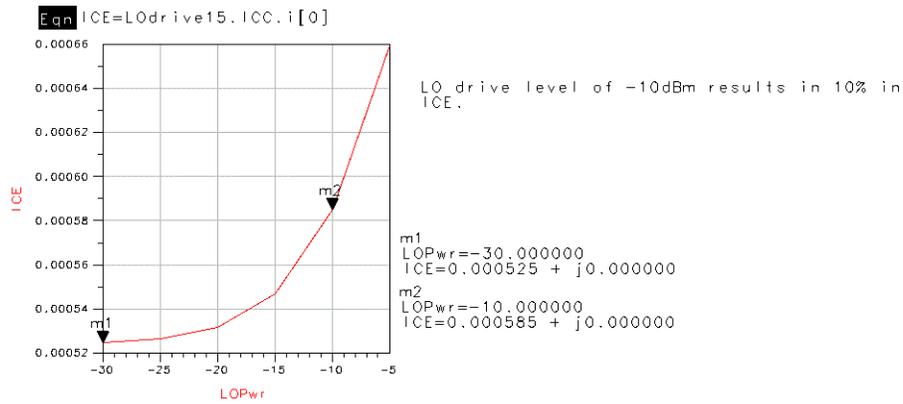
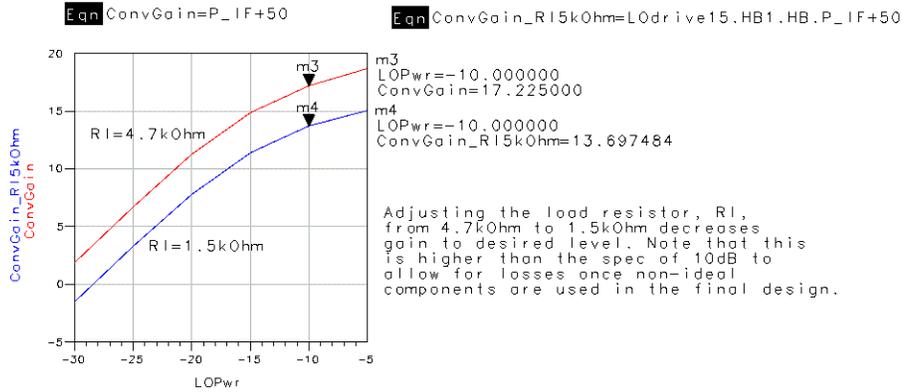


Figure 15. Conversion Gain and Bias Current Vary with LO Drive Level

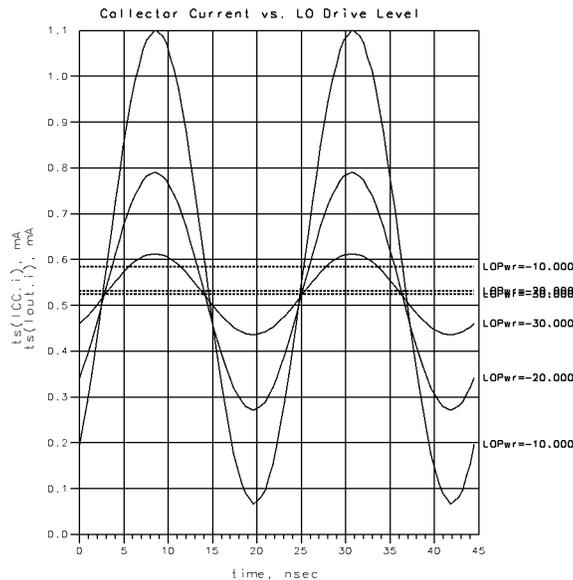


Figure 16. Variation in Output Collector Current with LO Drive Level

Mixer Conversion Gain versus RF Signal Level (MixCompr.dsn)

The set-up for measuring mixer compression used in MixCompr.dsn is very similar to LOdrive.dsn except that the LO power level is now held constant at -10dBm, while the RF power is swept from -50dBm to 0dBm. As the results in Figure 17 show, the mixer's conversion gain reaches 1dB compression at an input signal level of -27dBm.

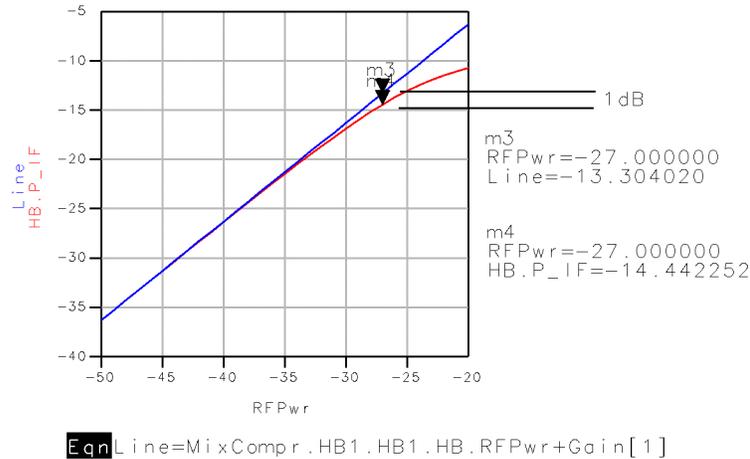


Figure 17. Mixer Conversion Gain Compression

Now that the mixer's performance is verified, the next step is to replace the ideal passive components with realistic models of the surface-mount resistors, capacitors and inductors that will be used in the actual circuit.

Creating the Mixer Layout (MixerLayout.dsn)

The design file MixerLayout.dsn contains a layout as well as a schematic. There are many possible ways to create layouts, and the best method will depend on the application. In this example, the first step was to convert all the components in the schematic to their nearest equivalent SMT part from the Passive Component Library. Next, these parts were placed in the layout window in their approximate locations. Interconnects were made in the layout window using the Trace command or microstrip components, and the final positioning of the components was adjusted. Finally, the schematic was updated using the design synchronization function.

MixerLayout.dsn was created by saving MixCompr.dsn under a new name and modifying it. Since the finished circuit will be simulated using the layout representation, it will have to be placed as a subnetwork in another schematic. This is because the layout file cannot contain simulator controllers, sources or terminations. The first step is to remove those components from the schematic and add ports to each point in the circuit that will be connected externally, either to sources, grounds or other circuits. The labels for each port will appear on the schematic symbol used when the design is placed in another schematic, so meaningful names should be provided. At this stage, the designer may also create a custom symbol for the circuit by selecting **View>Create/Edit Schematic Symbol**.

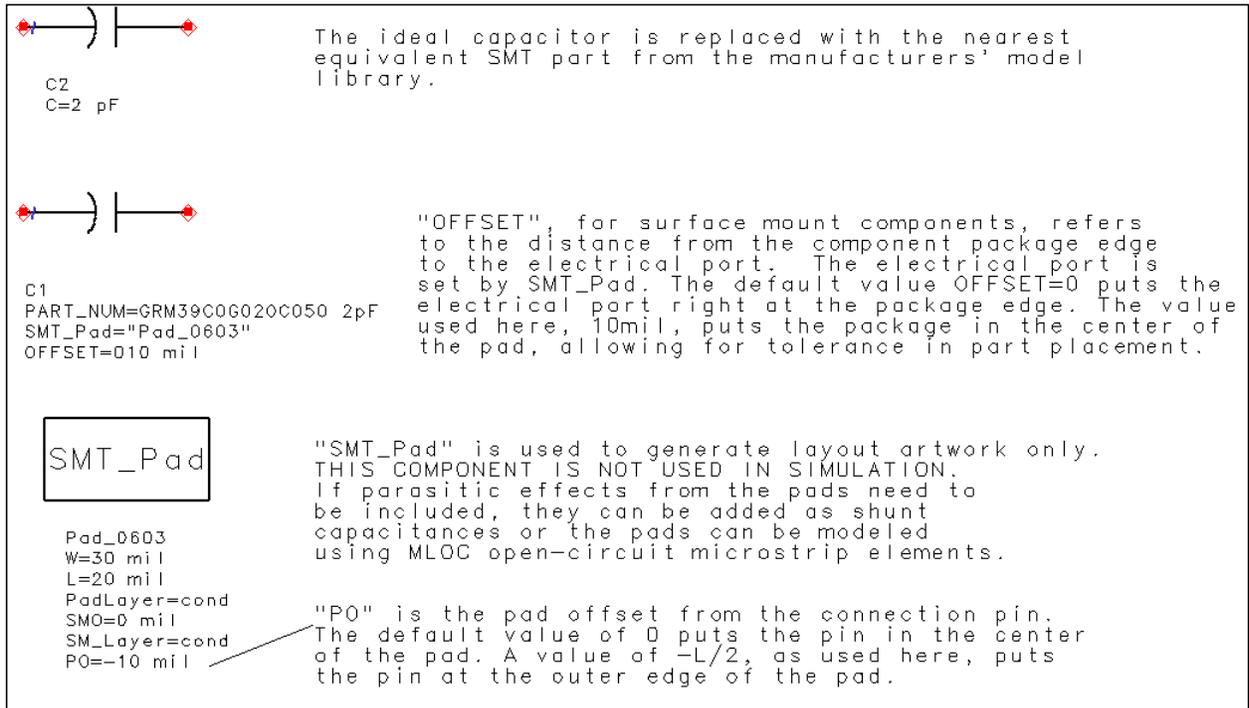


Figure 18. Substitution of SMT component for ideal component requires use of SMT_Pad

The next step is to replace each resistor, capacitor and inductor with a model of the SMT component that will be used in the actual circuit. The models are all found in the SMT Component Libraries by selecting the **Browse and Search** function in the Component Library List window. In this case, all the capacitors are MuRata Erie parts: the matching and bias capacitors are all MuRata Erie series GRM39 parts, while the RF bypass capacitors are GRM36 series. The resistors are taken from the Dale CRCW series and the inductors are Coilcraft parts. Where possible, parts are chosen to have a standard 0.060" x 0.030" footprint, although the inductors and RF bypass capacitors have different dimensions. Note that each SMT component specifies the name of the SMT_Pad component it uses. This SMT_Pad defines the pad-size to be used in layout, as shown in Figure 18. The designer must define these on the schematic page to ensure the pads appear correctly in the layout. Since each user will define the pads to suit their own board-fabrication process, the models do not include pad parasitics.

Once the components are placed and the pads defined, the designer can select **Layout>Place Components from Schematic to Layout** and place each part in its approximate location in the layout window. Traces are a convenient way to create the interconnects. They can be converted to equivalent microstrip components using the **Edit>Path/Trace/Convert Traces** command. In general, when moving back and forth between the schematic and layout representations, it is best to work on small sub-sections and synchronize the two representations manually. Synchronization ensures that both layout and schematic describe the same circuit: for example, if the designer has made some changes to the layout, the schematic can be updated to reflect them by selecting **Schematic>Generate/Update Schematic** in the layout window. Changes made to the schematic can be similarly transferred to the layout by choosing **Layout>Generate/Update Layout** in the schematic window.

Figure 19 shows the finished layout. A ground-plane has been added to the top-side metallization to eliminate the need for vias, thus reducing fabrication costs. This can be easily created in HP ADS by drawing a rectangle the size of the final circuit board and using the **Edit>Create Clearance** feature to generate the required spacing around transmission lines and component footprints.

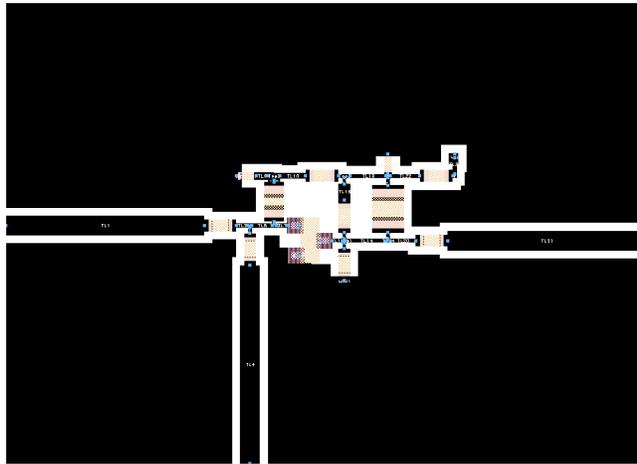


Figure 19. Finished Layout for Mixer Circuit

Finally, in this example the design will be simulated from layout, so the “SimLay” option is selected in the **File>Design/Parameters** dialog box. This allows the designer to see the effects of changes in the layout directly, without having to re-enter parameters in the schematic. Notice that components in the schematic can be modified (or even deleted entirely) without affecting the simulation, as long as the layout remains intact.

Simulation from Layout (SimFromLayout.dsn)

SimFromLayout.dsn contains MixerLayout, together with the simulation controller, sources and terminations required to simulate it. The simulation setup is identical to the one used in LOdrive.dsn so the results using the non-ideal components may be compared directly. MixerLayout uses microstrip lines, so an “MSub” component is also included (Figure 20).

Figure 21 shows the mixer conversion gain as a function of LO drive level when simulated using both the ideal components and the SMT model components. As expected, the conversion gain has drops significantly, due mainly to the resistive losses in the inductors. This can be verified by replacing individual components with their ideal counterparts and re-simulating. The load resistor can now be adjusted to compensate for these losses: changing Rl from 1.5k Ω to 3.3k Ω restores the simulated conversion gain to 10.76 dB, providing a 0.76dB margin over the specification. Note that these changes must be made in the layout file in order to be reflected in the simulation results. Once any such final corrections have been made to the layout, the circuit board is ready to be exported for fabrication.

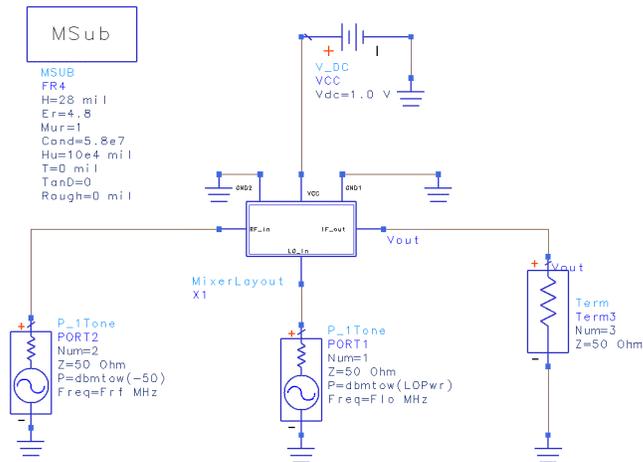


Figure 20. MixerLayout called as a sub-network in SimFromLayout.dsn

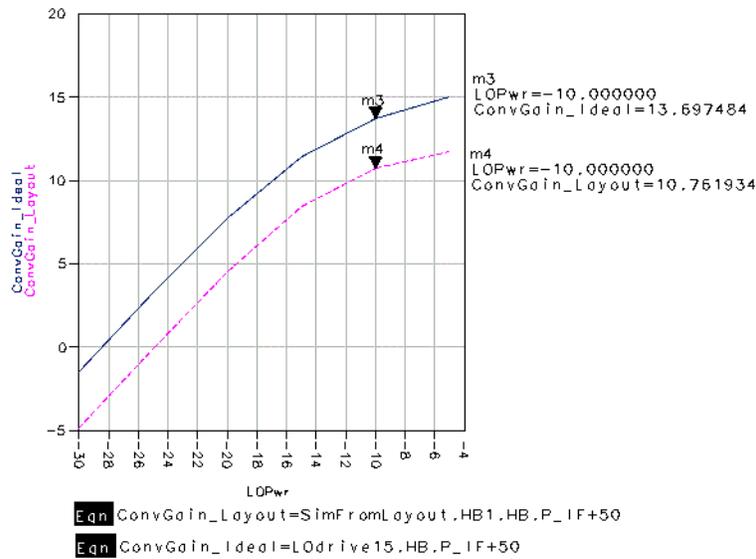


Figure 21. Comparison of Mixer Conversion Gain Using Ideal and SMT Components

Summary

An example of mixer design using HP ADS has been presented, including details of the design process and simulation set-ups. This example is included with HP ADS 1.0 and can be readily copied and modified by users for their own projects.

For more information about Agilent EEs of EDA, visit:

www.agilent.com/find/eesof

 **Agilent Email Updates**

www.agilent.com/find/emailupdates
Get the latest information on the products and applications you select.

 **Agilent Direct**

www.agilent.com/find/agilentdirect
Quickly choose and use your test equipment solutions with confidence.

www.agilent.com

For more information on Agilent Technologies' products, applications or services, please contact your local Agilent office. The complete list is available at:

www.agilent.com/find/contactus

Americas

Canada	(877) 894-4414
Latin America	305 269 7500
United States	(800) 829-4444

Asia Pacific

Australia	1 800 629 485
China	800 810 0189
Hong Kong	800 938 693
India	1 800 112 929
Japan	0120 (421) 345
Korea	080 769 0800
Malaysia	1 800 888 848
Singapore	1 800 375 8100
Taiwan	0800 047 866
Thailand	1 800 226 008

Europe & Middle East

Austria	0820 87 44 11
Belgium	32 (0) 2 404 93 40
Denmark	45 70 13 15 15
Finland	358 (0) 10 855 2100
France	0825 010 700*
	*0.125 €/minute
Germany	01805 24 6333**
	**0.14 €/minute
Ireland	1890 924 204
Israel	972-3-9288-504/544
Italy	39 02 92 60 8484
Netherlands	31 (0) 20 547 2111
Spain	34 (91) 631 3300
Sweden	0200-88 22 55
Switzerland	0800 80 53 53
United Kingdom	44 (0) 118 9276201

Other European Countries:

www.agilent.com/find/contactus

Revised: March 27, 2008

Product specifications and descriptions in this document subject to change without notice.

© Agilent Technologies, Inc. 2008

Printed in USA, January 1995

5989-8980EN