RFIC Design Using the Agilent RFIC Dynamic Link to Cadence

Combine capable tools to efficiently design RFICs.

Andy Howard, Agilent Technologies

The design process for RFICs can be more efficient when you combine capable tools. By combining the Cadence Analog Design Environment with the Advanced Design System (ADS) from Agilent EEsof EDA, you end up with a very powerful suite. In this case study of RFIC design, Cadence was used for schematic entry, layout, and verification while the ADS RF simulation and analysis environment helped to quickly determine what circuit parameters needed to be changed to improve performance. ADS was also used to determine the noise performance of the circuit.

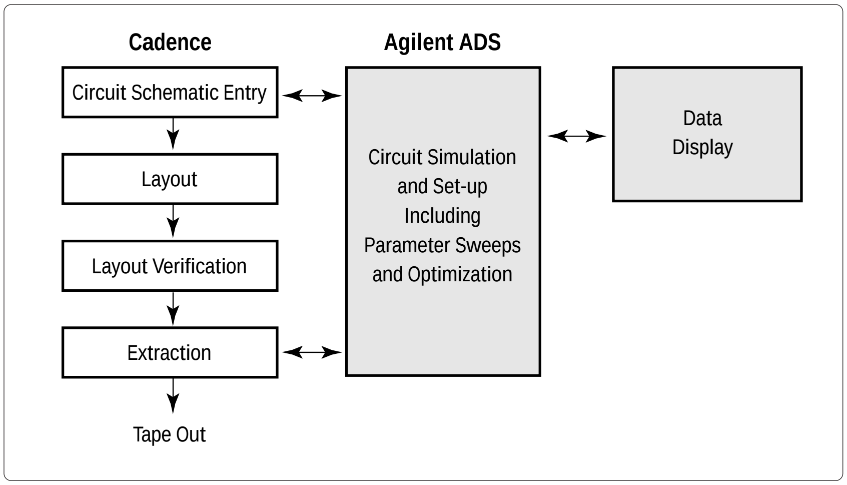


Figure 1. A design flow for RFICs using Cadence and ADS

The case study was the design of a prescaler (frequency divider) RFIC using a methodology incorporating Cadence and ADS. Each circuit to be simulated was entered into Cadence. The simulations were then carried out in ADS. Various parameters were swept in the simulations to examine their effects. **Figure 1** shows a design flow for RFICs using

Cadence and ADS. Some key points of the design flow are:

- using a single schematic entry in Cadence
- running all simulations in ADS, including parameter sweeps and optimizations
- using ADS templates to speed up simulation setups
- performing swept frequency analysis and phase-noise simulation using Transient-Assisted Harmonic Balance
- using device models from the IBM SiGe design kit
- creating the layout in Cadence, using parameterized cells from the IBM SiGe design kit.

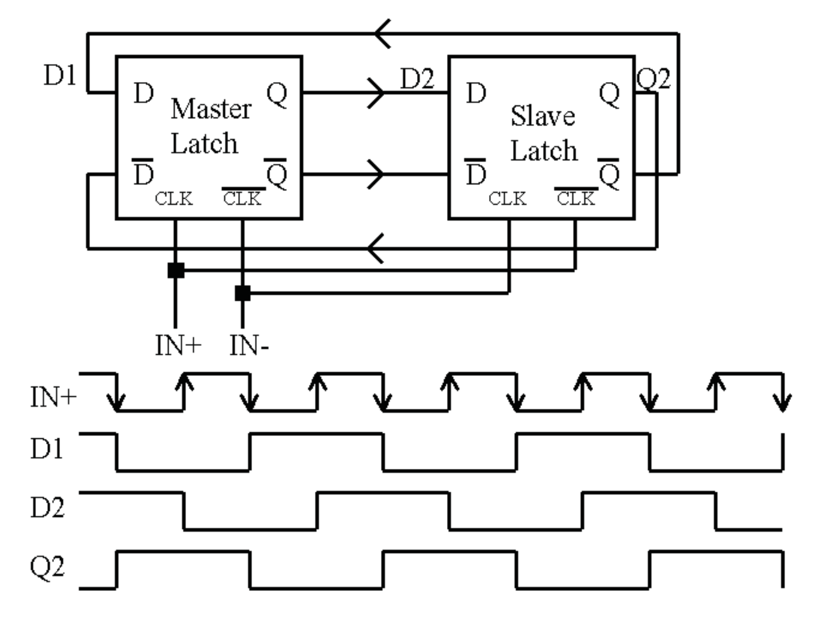


Figure 2. A prescaler block diagram, with signals (voltages versus time)

Design Flow and Methodology

The schematic of each circuit to be simulated was entered into Cadence. Simulations were carried out in ADS, with the Cadence schematics simulated as subcircuits. The ADS simulation setups were used to sweep various parameters and examine their effects. Once acceptable performance was obtained, a layout was created in Cadence. A design rule checker (DRC) was run on the layout to verify that it satisfied all of the IBM process design rules. Then a layout-versus-schematic (LVS) check was run to verify that the final schematic matched the layout. A simulation of the extracted layout (required for the LVS check) was run to see if the prescaler still operated satisfactorily. Then the layout was submitted for fabrication.

The general approach was to start with very simple, ideal circuits, verify that they functioned, check how performance varied with adjustable parameters, and gradually replace ideal components in the circuit with non-ideal elements. The main variables to be adjusted were the bias currents and voltages in each latch, and the signal amplitudes at the inputs and outputs of the latches.

How a Prescaler Works

A prescaler is a “digital” circuit that outputs a signal at half the frequency of the input signal. In this design, a D-flip flop master latch drives a D-flip flop slave latch, with the outputs of the slave fed back to the inputs of the master. With the connections shown in **figure 2**, the output signal from the slave will be at half the frequency of the input clock. By combining these in series, you can implement 1/4, 1/8, and higher division factors.

A prescaler should work for a relatively large range of input signal amplitudes. The input signal amplitude just has to be large enough to switch the current in an emitter-coupled pair (ECP) from one side to the other. Since these are digital circuits, the output signal amplitude is nearly independent of the input signal amplitude.

Prescalers are often used in phase-locked loops (PLLs) or multiplexers. They are used in PLLs when the VCO (voltage-controlled oscillator) frequency is above the maximum operating frequency of a programmable divider. They are also used for multplexing signals in optical transmitters.

Key Prescaler Specifications

Part of the case study was to see how high in frequency the prescaler could be made to operate using the IBM SiGe 5HP process. Other important specifications were the range of input signal power levels over which the prescaler would still operate, and phase noise. Specifications that were not of concern (because the part would not be put into production) included die size, operating temperature range, and power consumption.

Prescalers are not simple parts to simulate or measure, because the output frequency is not the same as the input frequency, and because the input frequency and input signal power level must be swept. In addition, the circuits are nonlinear, so you must use a nonlinear simulator. Most specifications of interest are steady state, meaning that if you use a time-domain simulator, you have to run the simulation long enough for the turn-on transient to die out. The time required for the transient to die out is multiplied if you are running multiple simulations varying some parameter like clock frequency or amplitude. The use of harmonic balance helps to avoid this problem.

Design Sequence

The following steps were used to design and simulate the prescaler:

1) Simulate the DC I-V curves of a single transistor. This was done to verify that the RFIC dynamic link was working properly.
2) Simulate the S-parameters versus bias voltage for a single transistor.
3) Simulate the propagation delay of an emitter-coupled pair (ECP) versus collector resistance, bias current, and emitter length.
4) Simulate a single latch with an ideal current source.
5) Simulate a divide-by-two cell with ideal current sources.
6) Sweep divider input frequency, bias current, collector resistance, and other parameters to see how high an input frequency the divider will divide.
7) Design and simulate a current mirror.
8) Rerun the divide-by-two swept simulations with current mirrors instead of ideal current sources.
9) Simulate the divider with a sine-wave input signal instead of a pulse input signal.
10) Use Transient-Assisted Harmonic Balance to simulate phase noise.
11) Sweep parameters to determine what affects phase noise, and determine what tradeoffs can be made.
12) Design and simulate an input amplifier.
13) Rerun divider simulations with the input amplifier, including tests to verify over what range of input signal amplitudes the divider still divides. Simulate performance with a single-ended input signal.
14) Design and simulate an output amplifier.
15) Rerun divider simulations with both the input and output amplifiers, and again verify over what range of input signal frequencies the divider divides.

16) Simulate phase noise of overall divider circuit, including both amplifiers.

17) Readjust parameter values to improve phase noise, and check frequency range of divider circuit. It is assumed there will be a tradeoff between frequency range and phase noise.

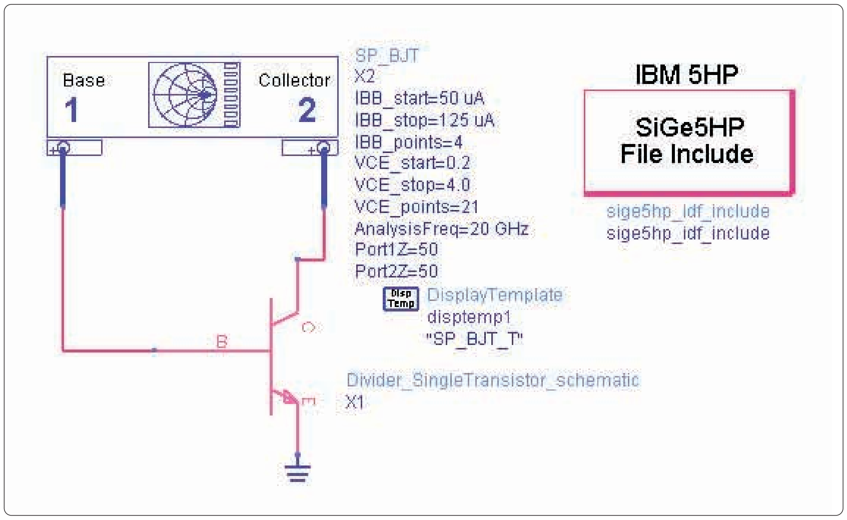
Design and Simulation Details

The first step in the design process was to simulate the I-V curves of a single transistor from the IBM SiGe design kit, and investigate how S21 varies with bias at a particular frequency. This is shown in **figure 3**. A Cadence schematic and then a Cadence symbol view were created for the single transistor. The same Cadence symbol was used in ADS.

Emitter-Coupled Pair Simulations

After simulating a single transistor, the next step was to simulate the basic building block of a frequency divider, the emitter-coupled pair. First the dc characteristics of this circuit and then its switching speed required simulation. **Figure 4** shows the ECP output voltages and emitter currents versus differential-mode input voltage. An approximately 200-mV change in the differential-mode input voltage was sufficient to completely switch the current from one device to the other.

After the dc transfer characteristics were examined, the switching delay speed was investigated, along with what could be done to improve it. A time-domain simulation was used for this, driving the ECP with a differential-mode input step. In the simulation, the emitter bias current, I_{EE}, was swept, but the collector resistance, Rcol, was varied along with I_{EE}, so voltage drop across the resistor, which sets the



SP_BJT
X2
IBB_start=50 uA
IBB_stop=125 uA
IBB_points=4
VCE_start=0.2
VCE_stop=4.0
VCE_points=21
AnalysisFreq=20 GHz
Port1Z=50
Port2Z=50

DisplayTemplate
disptemp1
"SP_BJT_T"

Divider_SingleTransistor_schematic
X1

IBM 5HP
SiGe5HP File Include
sige5hp_idf_include
sige5hp_idf_include

Figure 3. ADS simulation setup for a single transistor

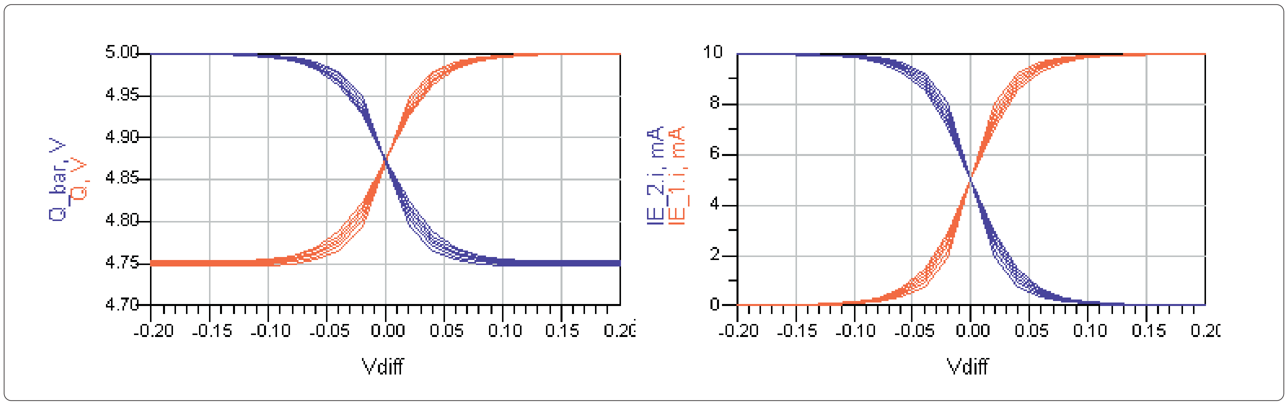


Figure 4. ECP output voltages (left) and emitter currents (right) versus differential-mode input voltage

logic "low" level, did not change. **Figure 5** shows the high-to-low and low-to-high propagation delays versus emitter bias current. This plot could be repeated versus other parameters. Based on these results, an emitter bias current of 10 mA was chosen, although 8 mA would perhaps produce about the same performance.

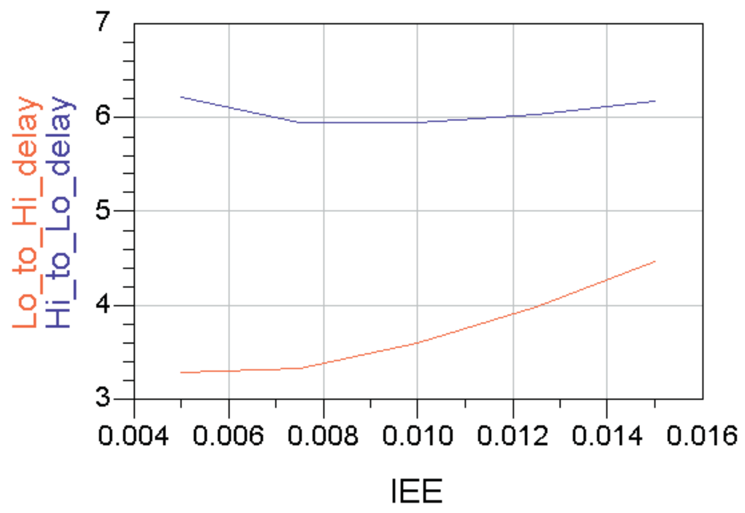


Figure 5. ECP propagation delay versus emitter bias current (The vertical axis is in picoseconds.)

Preliminary Divide-By-Two Simulations

After the ECP circuit was investigated, it was time to build a basic frequency divider (as shown in **figure 2**). Two identical latch circuits, with ideal bias current sources, were used as shown in the block diagram. The next step was to verify that this configuration operated as a frequency divider and to determine how high in frequency it could divide when the input signal was a differential-mode square-wave clock. Also requiring investigation was how the maximum frequency of operation varied with different circuit parameters such as bias currents, emitter lengths, collector resistances, and clock signal amplitude.

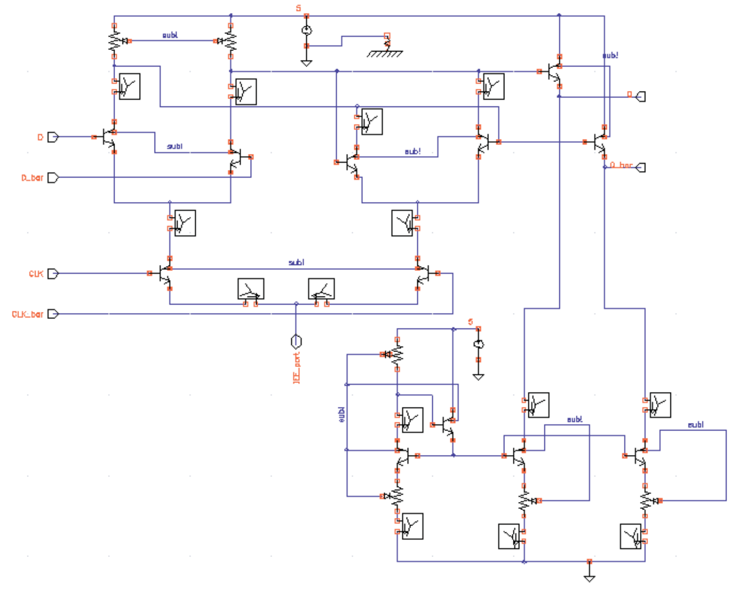
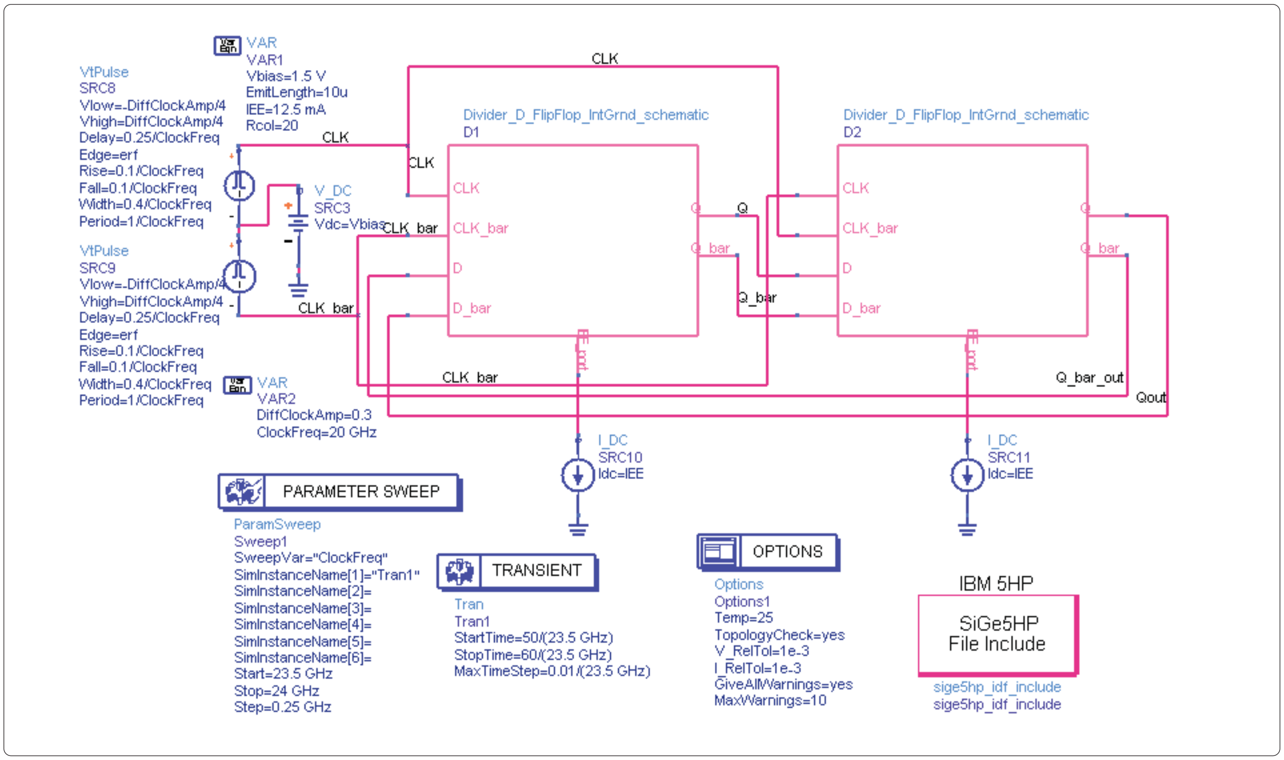


Figure 6. Cadence schematic of the latch circuit used in the preliminary divide-by-two simulations

Figure 6 shows the preliminary latch circuit that was simulated. Emitter followers were added at



VtPulse
SRC8
Vlow=-DiffClockAmp/4
Vhigh=DiffClockAmp/4
Delay=0.25/ClockFreq
Edge=erf
Rise=0.1/ClockFreq
Fall=0.1/ClockFreq
Width=0.4/ClockFreq
Period=1/ClockFreq

VtPulse
SRC9
Vlow=-DiffClockAmp/4
Vhigh=DiffClockAmp/4
Delay=0.25/ClockFreq
Edge=erf
Rise=0.1/ClockFreq
Fall=0.1/ClockFreq
Width=0.4/ClockFreq
Period=1/ClockFreq

VAR
VAR1
Vbias=1.5 V
EmitLength=10u
IEE=12.5 mA
Rcol=20

V_DC
SRC3
Vdc=Vbias

VAR
VAR2
DiffClockAmp=0.3
ClockFreq=20 GHz

Divider_D_FlipFlop_IntGrnd_schematic
D1

Divider_D_FlipFlop_IntGrnd_schematic
D2

I_DC
SRC10
Idc=IEE

I_DC
SRC11
Idc=IEE

ParamSweep
Sweep1
SweepVar="ClockFreq"
SimInstanceName[1]="Tran1"
SimInstanceName[2]=
SimInstanceName[3]=
SimInstanceName[4]=
SimInstanceName[5]=
SimInstanceName[6]=
Start=23.5 GHz
Stop=24 GHz
Step=0.25 GHz

Tran
Tran1
StartTime=50/(23.5 GHz)
StopTime=60/(23.5 GHz)
MaxTimeStep=0.01/(23.5 GHz)

Options
Options1
Temp=25
TopologyCheck=yes
V_RelTol=1e-3
I_RelTol=1e-3
GiveAllWarnings=yes
MaxWarnings=10

sige5hp_idf_include
sige5hp_idf_include

Figure 7. Divide-by-two simulation setup in ADS (Each of the “boxes” is the Cadence subcircuit of figure 6.)

the output to drive the inputs of the slave latch. These were necessary to keep the inputs' transistors for the slave latch out of saturation. These output emitter followers were biased with a current mirror. **Figure 7** shows the ADS setup for simulating the preliminary divide-by-two circuit. The differential input clock frequency was swept to determine the highest frequency of operation. Variables defined on this schematic (Rcol and EmitLength) were passed into the Cadence subcircuit.

You have considerable flexibility in defining the shape of the input pulse. One of the limitations of doing a swept transient pulse is that it can take many clock cycles before the divider reaches steady state. You have to restart from time = 0 each time the clock period is changed. With a technique called Transient-Assisted Harmonic Balance, this wait for the initial transient to die out is not necessary.

Figure 8 shows the simulation results, which indicate that the frequency divider works up to 23.75 GHz. A number of iterations were made to the circuit parameter values before this performance was achieved. Also, because parasitics were not included and because the current sources were ideal, it is not realistic to expect this level of performance from the fabricated ICs.

Phase-Noise Simulation

The phase noise contributed by a frequency-divider circuit is an important specification. Designers want to know ahead of time the phase-noise performance of their circuits as well as what changes they can make to improve noise performance. Time-domain simulators like Spice and its derivatives do not support phase-noise calculations. Harmonic balance does simulate phase noise, but needs some help to solve frequency-divider circuits. Transient-Assisted Harmonic Balance enables harmonic balance to solve circuits like these, and operates as follows. A transient simulation is run until the circuit reaches steady state. The spectra of the periodic node voltages and branch currents are computed and saved internally, and used as an initial guess for harmonic balance. In many cases, harmonic balance is then able to converge on the solution. Once harmonic balance has converged, a phase-noise simulation or various sweeps may be run.

For more-accurate noise simulation in this RFIC design, the ideal current sources that biased the latch circuits were replaced with a simple transistor current mirror. As an experiment, how phase noise varies with bias current was investigated. To make this easy to carry out, a current multiplication factor variable “ImultFactor” was defined, and the current mirror resistances and emitter sizes were made a function of this variable. **Figure 9** shows the amplitude and phase noise at one offset frequency,

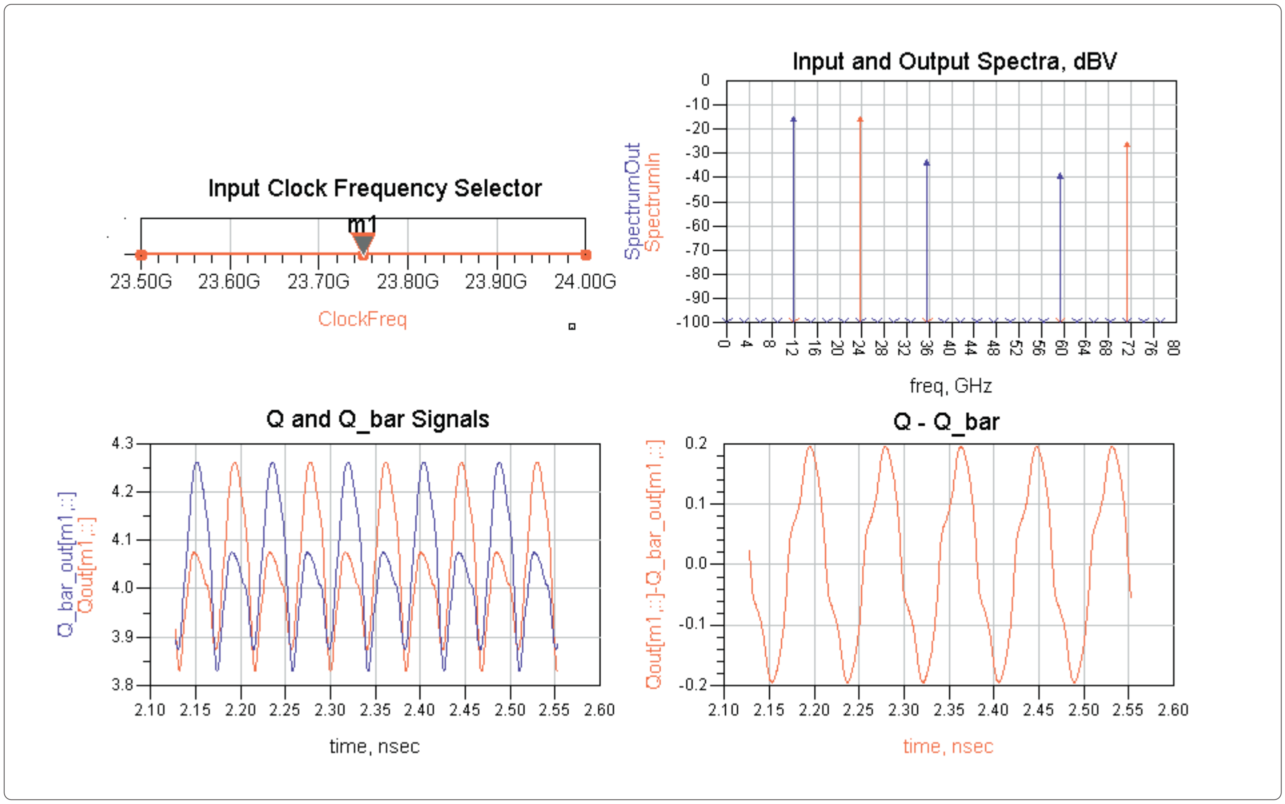
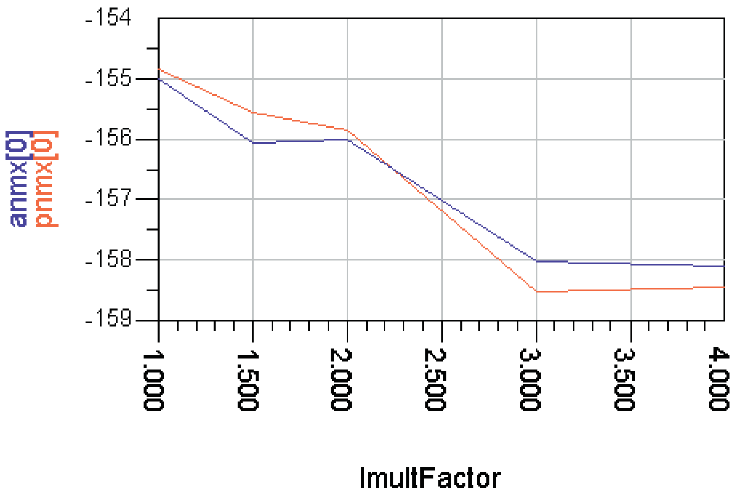


Figure 8. Preliminary frequency-divider simulation results

versus the current multiplication factor. This simulation did not include input or output amplifiers, so the actual phase noise would likely be worse. Other circuit parameters could easily be varied to determine their effect on phase noise.

Input Amplifier Simulations

The input amplifier was designed to provide a reasonably good match to 50 Ω, handle either differential or single-ended signals, and convert a reasonably large dynamic range of input signal amplitudes to a signal level sufficient to drive the divide-by-two circuit. The design was just an input-stage ECP driving two stages of emitter followers. The dc transfer curves, small-signal frequency response, and the large-signal gain compression were examined, and the bias currents were varied in the different stages. **Figure 10** shows the small-signal differential-mode gain simulation results, which indicate that the gain increases with bias current as expected.



ImultFactor	pnmx[0]	anmx[0]
1.000	-154.834	-155.007
1.500	-155.563	-156.059
2.000	-155.863	-156.006
3.000	-158.529	-158.035
4.000	-158.458	-158.116

Figure 9. Amplitude (anmx) and phase noise (pnmx) versus current multiplication factor (ImultFactor)

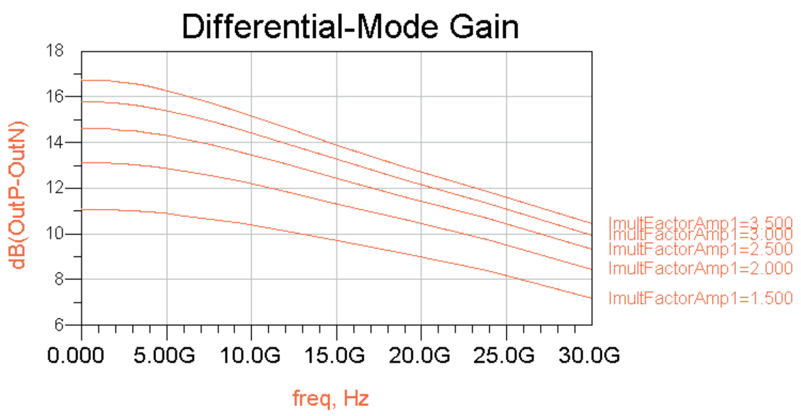


Figure 10. Small-signal differential-mode gain simulation results

It was also important to examine the large-signal characteristics of the amplifier to determine over what range of input signal amplitudes the divider could function. **Figure 11** shows the differential-mode output voltage waveforms and the output voltage in dB versus input signal power. These plots indicate that as long as the clock power level is greater than -10 dBm, the peak-to-peak differential-mode output signal will be greater than about 200 mV, which should be large enough to drive the divide-by-two core. There could be some signal degradation due to the loading of the divide-by-two core circuit input ECP when it is connected to the amplifier.

Simulating the Top-Level Divider

The output amplifier was designed with a topology similar to that of the input amplifier, using similar techniques. Then the overall performance of the frequency divider, including both amplifiers, needed to be tested. There was some degradation in the maximum operating frequency when the output amplifier was connected, but increasing the amount of available drive current at the output of the slave latch helped alleviate this. Transient-Assisted Harmonic Balance was used again, this time to determine

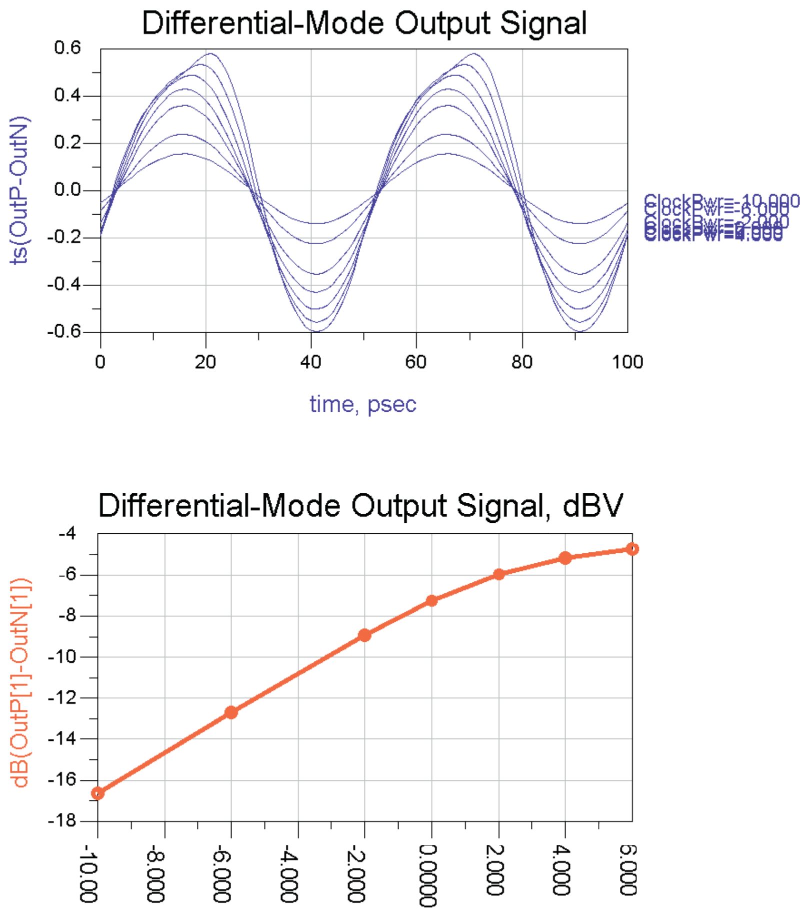
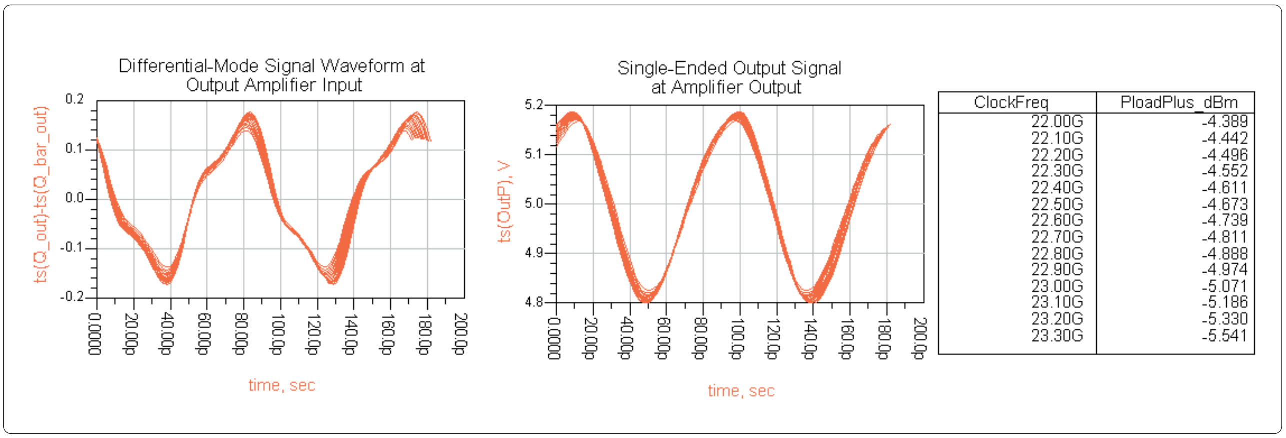


Figure 11. Differential-mode output voltage waveforms (top) and the output voltage in dB versus input signal power (bottom)



ClockFreq	PloadPlus_dBm
22.00G	-4.389
22.10G	-4.442
22.20G	-4.496
22.30G	-4.552
22.40G	-4.611
22.50G	-4.673
22.60G	-4.739
22.70G	-4.811
22.80G	-4.888
22.90G	-4.974
23.00G	-5.071
23.10G	-5.186
23.20G	-5.330
23.30G	-5.541

Figure 12. Top-level divider waveforms (left) and output power at the fundamental frequency (half the clock frequency) (right)

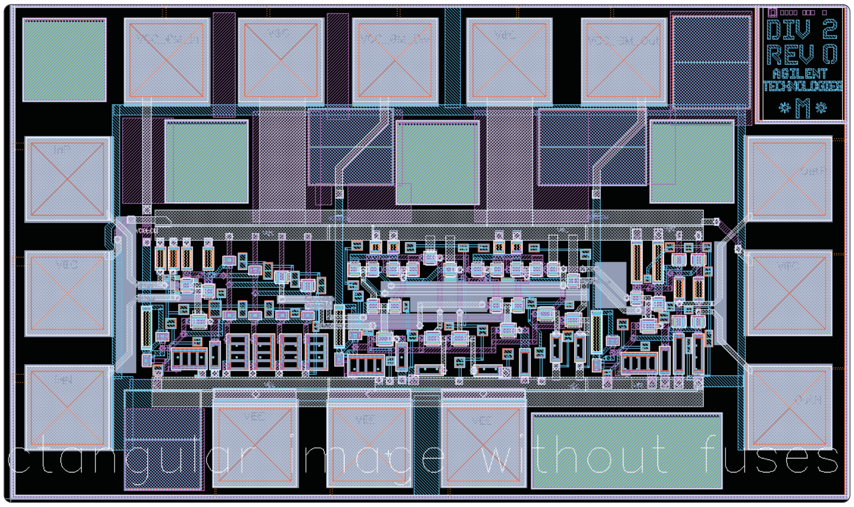


Figure 13. Completed layout of the prescaler IC

the maximum operating frequency and the fundamental output power versus output frequency response of the frequency divider. **Figure 12** shows the output waveforms from the divider core and from the output amplifier, and the fundamental output power in dBm. The input clock frequency was swept, and there was one trace per input clock frequency. The divider worked to a very high operating frequency in simulation, but the actual operating frequency will not be as high due to parasitic effects that were not included in these simulations.

Layout

The next step was to create a layout of the IC, using Cadence. The design was required to pass a Layout Versus Schematic (LVS) test before being submitted for fabrication, so any elements in the schematic such as voltage sources and current probes that were not to be in the layout had to be removed. Also, all variables used in component parameter definitions had to be replaced with constants.

The layout was created in three separate parts: input amplifier, divider core, and output amplifier; each of these passed LVS testing before the final layout was created. Trace lengths were minimized, especially the feedback path from the output of the slave latch to the input of the master latch. Trace widths were manually checked to make sure that they were sufficient to satisfy IBM's current capacity requirements. **Figure 13** shows the completed layout, including the input and output amplifiers, all the bondpads, pattern fill, and substrate contacts.

Comparison with Measured Results

After fabrication of the prescaler IC was completed, a number of them were measured, and their performance was found to be quite consistent. However, at higher frequencies, the simulations without any parasitic elements included were overly optimistic. Performing a parasitic extraction from the Cadence layout using the Columbus RF tool and including models for the bondwires used to bias the IC provided reasonably close agreement with the measurements. It is not sufficient to just compare the simulated and measured divided signal amplitudes at the fundamental frequencies when determining whether there is good agreement or not. The waveforms should be compared also.

Figure 14 shows the measured and simulated output waveforms with different input clock frequencies. With an input clock of up to 12 GHz, the waveforms agree quite well.

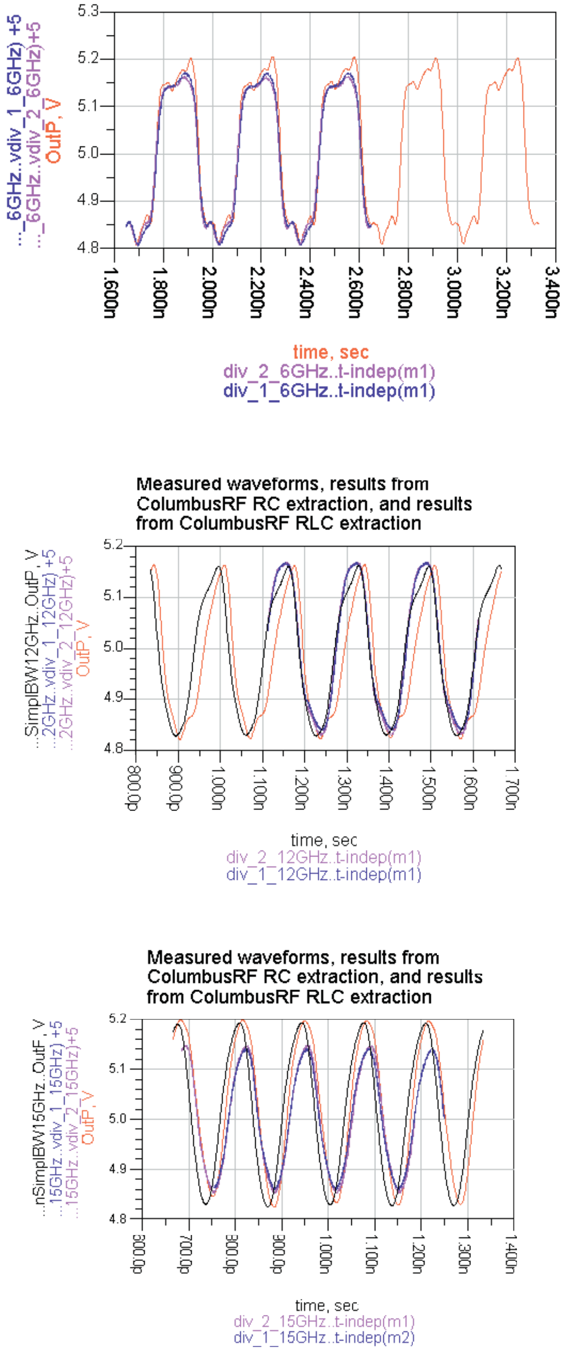


Figure 14. Measured and simulated waveforms, (a) for a 6-GHz input clock, (b) for a 12-GHz input clock, and (c) for a 15-GHz input clock

At 15 GHz, the shapes of the waveforms still agree, but the measured amplitude is somewhat less than the simulated amplitude. A possible explanation for this is that parasitic coupling to the substrate is not being included in the extraction, and at higher frequencies this coupling has a larger effect and needs to be modeled.

This case study shows that an RFIC design flow using Cadence for schematic entry and layout, and ADS for simulation and data display, works well. ADS offers valuable simulation and data display technology even for the design of what could be considered a digital IC. The IBM SiGe foundry produced ICs with good performance, especially considering that there was only a single pass through the foundry. However, results could have been improved by including parasitics in the simulations to accurately predict performance.

Expanded from original article in
Agilent Measurement Solutions
Volume 1. Issue 2

Printed in the USA April 15, 2002
5988-5981EN