Chapter 11

Optimizing Performance

Optimization, the automatic generation of model parameters and component values from a given set of electrical specifications or measured data, is available in Star-Hspice. With a user-defined optimization program and a known circuit topology, Star-Hspice automatically selects the design components and model parameters to meet DC, AC, and transient electrical specifications.

Star-Hspice optimization is the result of more than ten years of research in both the optimizing algorithms and user interface. The optimizing function is integrated into the core of Star-Hspice, resulting in optimum efficiency. The circuit result targets are part of the .MEASURE command structure, and the parameters to be optimized are Star-Hspice-defined parameter functions. A .MODEL statement sets up the optimization.

The most powerful feature of the Star-Hspice approach is its incremental optimization technique. Incremental optimization allows you to solve the DC parameters first, then the AC parameters, and finally the transient parameters. A set of optimizer measurement functions not only makes the task of transistor optimization easy, but significantly improves cell and whole circuit optimization.

This chapter covers the following topics:

Performing Optimization

To perform optimization, create an input netlist file specifying:

- Minimum and maximum parameter and component limits
- The variable parameters and components
- An initial estimate of the selected parameter and component values
- The circuit performance goals or model-versus-data error function

Given the input netlist file, optimization specifications, component limits, and initial guess, the optimizer reiterates the circuit simulation until the target electrical specification is met or an optimized solution is found.

For improved optimization and simulation time and to increase the likelihood of a convergent solution, the initial estimate of the component values should produce a circuit with specifications near those of the original target. This reduces the number of times the optimizer reselects component values and resimulates the circuit.

Optimization Control

The length of time to complete an optimization is a function of the number of iterations allowed, the relative input tolerance, the output tolerance, and the gradient tolerance. The default values are satisfactory for most applications. Generally, 10 to 30 iterations are sufficient to get accurate optimizations.

Simulation Accuracy

Set the simulator with tighter convergence options than normal for optimization. The following options are suggested:

For DC MOS model optimizations:

```
absmos=1e-8
relmos=1e-5
relv=1e-4
```

For DC JFET, BJT, and diode model optimizations:

```
absi=1e-10
reli=1e-5
relv=1e-4
```

For transient optimizations:

```
relv=1e-4
relvar=1e-2
```

Curve Fit Optimization

Use optimization to curve fit user-defined DC, AC, or transient data. In a curve fit optimization, the desired numeric data for curves is stored in the data file as in-line data using the .DATA statement. The variable circuit components and parameter values of the netlist are specified in the .PARAM xxx=OPTxxx statement. The optimization analysis statements call the in-line data using the DATA= keyword. The .MEASURE statement uses the simulation result and compares it with the values given in the data file. The .MEASURE statement controls the comparison of simulation results to the values given in the data file. This is usually done with the ERR1 keyword. If the calculated value is not within the error tolerances specified in the optimization model, a new set of component values are selected and the circuit is resimulated. This is repeated until the closest fit to the curve is obtained, or the error tolerances set is satisfied.

Avant!'s Device Model Builder (DMB) product automates curve-fit optimization for the development of most DC MOSFET models.

Goal Optimization

Goal optimization differs from curve fit optimization in that it usually only applies to the optimization of a particular electrical specification, such as rise time or power dissipation.

Goal optimizations are specified using the GOAL keyword with a choice of relational operator in the .MEASURE statement, where GOAL is the target electrical specification being measured. This choice of relational operator is useful in multiple-constraint optimizations, when the absolute accuracy of some criteria is less important than for others.

Understanding the Statements Syntax

Several Star-Hspice statements are required for optimization.

- .MODEL modname OPT ...
- .PARAM parameter=OPTxxx (init, min, max)
- A .DC, .AC, or .TRAN analysis statement with MODEL=modname, OPTIMIZE=OPTxxx, and RESULTS=measurename
- .MEASURE measurename ... <GOAL = | < | > val> – note that a space is required on either side of the relational operator =, <, or >

The .PARAM statement lets you specify initial, lower, and upper bound values. The types of .MEASURE statements available for optimization are described in *Chapter 4, Specifying Simulation Output*.

Output statements .PRINT, .PLOT, and .GRAPH must be associated with the analysis statements .DC, .AC, or .TRAN. An analysis statement with the keyword OPTIMIZE is used for optimization only. To generate output for the optimized circuit, another analysis statement (.DC, .AC, or .TRAN) must be specified, along with the output statements. The proper specification order is:

1. Analysis statement with OPTIMIZE
2. .MEASURE statements specifying optimization goals or error functions
3. Ordinary analysis statement
4. Output statements

Analysis Statement Syntax (.DC, .TRAN, .AC)

The syntax is:

```
.DC <DATA=filename> SWEEP OPTIMIZE=OPTxxx RESULTS=ierr1 ...
+  ierrn MODEL=optmod
```

or

```
.AC <DATA=filename> SWEEP OPTIMIZE=OPTxxx RESULTS=ierr1 ...
+  ierrn MODEL=optmod
```

or

```
.TRAN  <DATA=filename> SWEEP OPTIMIZE=OPTxxx RESULTS=ierr1
+  ...  ierrn MODEL=optmod
```

where:

DATA specifies the in-line file of parameter data to use in the optimization

OPTIMIZE indicates the analysis is for optimization. Specifies the parameter reference name used in the .PARAM optimization statement. All .PARAM optimization statements with the parameter reference name selected by OPTIMIZE will have their associated parameters varied during an optimization analysis.

MODEL the optimization reference name that is also specified in the .MODEL optimization statement

RESULTS the measurement reference name that is also specified in the .MEASURE optimization statement. RESULTS is used to pass analysis data to the .MEASURE optimization statement.

.PARAM Statement Syntax

The syntax is:

```
.PARAM parameter=OPTxxx (initial_guess, low_limit, upper_limit)
```

or

```
.PARAM parameter=OPTxxx (initial_guess, low_limit, upper_limit,
+  delta)
```

where:

OPTxxx specifies the optimization parameter reference name, referenced by the associated optimization analysis. This must agree with the OPTxxx name given in the analysis command associated with the keyname OPTIMIZE.

parameter specifies the parameter to be varied, the initial value estimate, the lower limit, and the upper limit allowed for the parameter. If the best solution does not exist within these constraints, the optimizer attempts to find the best solution.

delta the final parameter value is the initial guess ± (*n*·delta). If delta is not specified, the final parameter value can be anything between low_limit and upper_limit. This is useful for optimizing transistor drawn widths and lengths, which must be quantized.

Example

```
.PARAM vtx=OPT1(.7,.3,1.0) uox=OPT1(650,400,900)
```

In the above example, the parameters *uox* and *vtx* are the variable model parameters to optimize a model for a given set of electrical specifications. The parameter *vtx* is given an initial value estimate of 0.7 volts and can be varied within the limits of 0.3 and 1.0 volts for the optimization procedure. The optimization parameter reference name, OPT1, is used to reference the associated optimization analysis statement (not shown).

.MODEL Statement Syntax

For each optimization within a data file, specify a .MODEL statement to allow for more than one optimization per simulation run to be executed. The optimization .MODEL statement defines the convergence criteria, number of iterations, and derivative methods.

The syntax is:

```
.MODEL mname OPT <parameter=val ...>
```

You can specify the following OPT parameters in the .MODEL statement:

mname model name. Elements refer to the model by this name.

CENDIF represents the point at which more accurate derivatives are required. When the gradient of the RESULTS functions are less than CENDIF, the more time-consuming derivative methods are used. Values of 0.1 to 0.01 are suitable for most applications. If too large a value is used, the optimizer requires more CPU time. If too small a value is used, it might not find as accurate an answer. Default=1.0e-9.

CLOSE the initial estimate of how close the parameter initial value estimates are to the final solution. CLOSE multiplies the changes in the new parameter estimates. A large value for CLOSE causes the optimizer to take large steps toward the solution and a small value causes the optimizer to take smaller steps toward the solution. CLOSE should range from 0.01 for very close parameter estimates to 10 for rough initial guesses. Default=1.0.

If CLOSE is greater than 100, the steepest descent part of the Levenburg-Marquardt algorithm dominates. For CLOSE less than 1, the Gauss-Newton method dominates. For further details, see L. Spruiell, “Optimization Error Surfaces,” *Meta-Software Journal*, Vol. 1, No. 4, December 1994.

CUT modifies CLOSE, depending on how successful the iterations toward the solution become. If the last iteration was successful, descent toward the solution CLOSE is decreased by CUT. If the last iteration was not a successful descent to the solution, CLOSE is increased by CUT squared. CUT drives CLOSE up or down depending on the relative success in finding the solution. The CUT value must be greater than 1. Default = 2.0.

DIFSIZ determines the increment change in a parameter value for gradient calculations ($\Delta x = DIFSIZ \cdot max(x, 0.1)$). If delta is specified in a .PARAM statement, then Δx = delta. Default = 1e-3.

GRAD represents a possible convergence when the gradient of the RESULTS function is less than GRAD. Values of 1e-6 to 1e-5 are suitable for most applications. If too large a value is used, the optimizer could stop before the best solution is found. Too small a value requires more iterations. Default=1.0e-6.

ITROPT sets the maximum number of iterations. Typically no more than 20-40 iterations are needed to find a solution. Too many iterations can imply the values for RELIN, GRAD, or RELOUT are too small. Default=20.

LEVEL selects the optimizing algorithm to use. Currently, the only option is LEVEL=1, a modified Levenburg-Marquardt algorithm.

MAX sets the upper limit on CLOSE. Values greater than 100 are recommended. Default=6000.

PARMIN allows better control of incremental parameter changes during error calculations. This produces more control over the trade-off between simulation time and optimization result accuracy. Star-Hspice calculates parameter increments using the relationship:

$$\Delta par\_val = \mathrm{DIFSIZ} \cdot \mathrm{MAX}(par\_val, \mathrm{PARMIN})$$

Default=0.1.

RELIN specifies the relative input parameter variation for convergence. If all the optimizing input parameters vary by no more than RELIN from one iteration to the next, then the solution is declared convergent. Since RELIN is a relative variance test, a value of 0.001 implies that the optimizing parameters are varying by less then 0.1% from one iteration to the next. Default=0.001.

RELOUT represents the relative output RESULTS function variance for convergence. For RELOUT=0.001, the difference in the RMS error of the RESULTS functions should vary less than 0.001. Default=0.001.

Examining Optimization Examples

This section provides examples of the following types of Star-Hspice optimizations:

- MOS Level 3 Model DC Optimization
- MOS Level 13 Model DC Optimization
- RC Network Optimization
- CMOS Tristate Buffer Optimization
- BJT S-Parameters Optimization
- BJT Model DC Optimization
- GaAsFET Model DC Optimization
- MOS Op-amp Optimization

MOS Level 3 Model DC Optimization

This example shows an optimization of I-V data to a Level 3 MOS model. The data consists of gate curves (*ids* versus *vgs*) and drain curves (*ids* versus *vds*). The Level 3 parameters VTO, GAMMA, UO, VMAX, THETA, and KAPPA are optimized. After optimization, the model is compared separately to the data for the gate and drain curves. The option POST generates AvanWaves files for comparing the model to the data.

Level 3 Model DC Optimization Input Netlist File

```
$level 3 mosfet optimization
$..tighten the simulator convergence properties
.OPTION nomod post=2 newtol relmos=1e-5 absmos=1e-8
.MODEL optmod OPT itropt=30
```

Circuit Input

```
vds  30 0  vds
vgs  20 0  vgs
vbs  40 0  vbs
m1 30 20 0 40 nch w=50u l=4u
$..
$..process skew parameters for this data
```

```
.PARAM xwn=-0.3u xln=-0.1u toxn=196.6 rshn=67
$..the model and initial guess
.MODEL nch NMOS level=3
+  acm=2 ldif=0 hdif=4u tlev=1 n=2
+  capop=4 meto=0.08u xqc=0.4
$...note capop=4 is ok for H8907 and later, otherwise use
$...Capop=2
$...fixed parameters
+  wd=0.15u ld=0.07u
+  js=1.5e-04 jsw=1.8e-09
+  cj=1.7e-04 cjsw=3.8e-10
+  nfs=2e11 xj=0.1u  delta=0  eta=0
$...process skew parameters
+  tox=toxn rsh=rshn
+  xw=xwn xl=xln
```

Optimized Parameters

```
+ vto=vto gamma=gamma
+ uo=uo  vmax=vmax theta=theta kappa=kappa
.PARAM
+ vto    = opt1(1,0.5,2)
+ gamma  = opt1(0.8,0.1,2)
+ uo     = opt1(480,400,1000)
+ vmax   = opt1(2e5,5e4,5e7)
+ theta  = opt1(0.05,1e-3,1)
+ kappa  = opt1(2,1e-2,5)
```

Optimization Sweeps

```
.DC DATA=all  optimize=opt1  results=comp1  model=optmod
.MEAS DC comp1 ERR1 par(ids) i(m1) minval=1e-04 ignor=1e-05
```

DC Sweeps

```
.DC DATA=gate
.DC DATA=drain
```

Print Sweeps

```
.PRINT DC vds=par(vds) vgs=par(vgs) im=i(m1) id=par(ids)
.PRINT DC vds=par(vds) vgs=par(vgs) im=i(m1) id=par(ids)
```

DC Sweep Data

```
$..data
.PARAM vds=0 vgs=0 vbs=0 ids=0
```

```
.DATA all vds vgs vbs ids
1.000000e-01 1.000000e+00  0.000000e+00 1.655500e-05
5.000000e+00 5.000000e+00 0.000000e+00 4.861000e-03
.ENDATA
.DATA gate vds vgs vbs ids
1.000000e-01 1.000000e+00  0.000000e+00 1.655500e-05
1.000000e-01 5.000000e+00 -2.000000e+00 3.149500e-04
.ENDDATA
.DATA drain vds vgs vbs ids
2.500000e-01 2.000000e+00 0.000000e+00 2.302000e-04
5.000000e+00 5.000000e+00 0.000000e+00 4.861000e-03
.ENDDATA
.END
```

The Star-Hspice input netlist shows:

- Using .OPTIONS to tighten tolerances increases the accuracy of Star-Hspice. This is recommended for I-V optimization.
- “.MODEL optmod OPT itropt=30” limits the number of iterations to 30.
- The circuit is simply one transistor. VDS, VGS, VBS are parameter names that match names used in the data statements.
- The process variation parameters, XL, XW, TOX, RSH are specified as constants in a .PARAM statement. These are measured parameters for the device being characterized.
- The model contains references to parameters. In “GAMMA= GAMMA”, the left-hand side is a Level 3 model parameter name, while the right-hand side is a “.PARAM” parameter name.
- The long .PARAM statement specifies initial, min and max values for the optimized parameters. UO is initialized at 480 and kept within the range 400 to 1000 during optimization.
- The first .DC statement indicates that the data is in the in-line “.DATA all” block (which contains merged gate and drain curve data), optimization of parameters that were declared as OPT1 (in this case all of the optimized parameters), error function COMP1 (matches the name of a .MEASURE statement), and model OPTMOD (sets iteration limit).
- The .MEASURE statement specifies least-squares relative error. The difference between data par(ids) and model i(m1) is divided by either the

absolute value of par(ids), or by minval=10e-6, whichever is larger. Using minval keeps low current data from dominating the error.

- The remaining .DC and .PRINT statements are for printback after optimization. You can be place them anywhere in the netlist input file because they will be correctly assigned when the file is parsed.
- The “.PARAM VDS=0 VGS=0 VBS=0 IDS=0” statements simply declare these data column names as parameters.

The .DATA statements give data for IDS versus VDS, VGS, VBS. The selection of data should match the choice of model parameters to optimize. To optimize GAMMA, data with back bias must be provided (VBS= -2 in this case). To optimize KAPPA, the saturation region must have data. In this example, the data set “all” contain:

gate curves: vds=0.1 vbs=0,-2 vgs=1 to 5 in steps 0.25

drain curves: vbs=0 vgs=2,3,4,5 vds=0.25 to 5 in steps 0.25

The results are shown in Figure 11-1:.

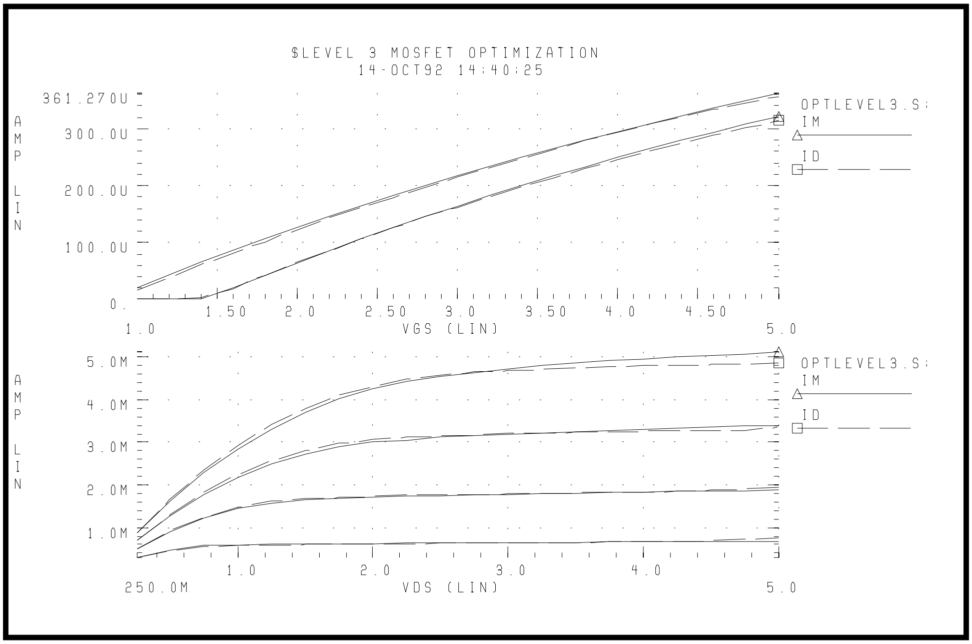

Figure 11-1: Level 3 MOSFET Optimization

MOS Level 13 Model DC Optimization

This example shows an optimization of I-V data to a Level 13 MOS model. The data consists of gate curves (*ids* versus *vgs*) and drain curves (*ids* versus *vds*). This example demonstrates two-stage optimization. The Level 13 parameters vfb0, k1, muz, x2m, and u00 are optimized to the gate data. Then the Level 13 parameters MUS, X3MS, U1, and the impact ionization parameter ALPHA are optimized to the drain data. After optimization, the model is compared to the data. The option POST generates AvanWaves files for comparing the model to the data.

The results are shown in Figure 11-2:.

Level 13 Model DC Optimization Input Netlist File

```
$level 13 mosfet optimization
$..tighten the simulator convergence properties
.OPTION nomod post=2
+       newtol relmos=1e-5 absmos=1e-8
.MODEL optmod OPT itropt=30
```

Circuit Input

```
vds  30 0  vds
vgs  20 0  vgs
vbs  40 0  vbs
m1 30 20 0 40 nch w=50u l=4u
$..
$..process skew parameters for this data
.PARAM xwn=-0.3u xln=-0.1u toxn=196.6 rshn=67
$..the model and initial guess
.MODEL nch NMOS level=13
+  acm=2 ldif=0 hdif=4u tlev=1 n=2 capop=4 meto=0.08u xqc=0.4
$...parameters obtained from measurements
+  wd=0.15u ld=0.07u js=1.5e-04 jsw=1.8e-09
+  cj=1.7e-04 cjsw=3.8e-10
$...parameters not used for this data
+  k2=0 eta0=0 x2e=0 x3e=0  x2u1=0 x2ms=0 x2u0=0 x3u1=0
$...process skew parameters
+  toxm=toxn rsh=rshn
+  xw=xwn xl=xln
$...optimized parameters
+   vfb0=vfb0 k1=k1 x2m=x2m muz=muz u00=u00
+   mus=mus x3ms=x3ms u1=u1
$...impact ionization parameters
+   alpha=alpha vcr=15
.PARAM
+ vfb0    = opt1(-0.5, -2, 1)
+ k1      = opt1(0.6,0.3,1)
+ muz     = opt1(600,300,1500)
+ x2m     = opt1(0,-10,10)
+ u00     = opt1(0.1,0,0.5)
+ mus     = opt2(700,300,1500)
+ x3ms    = opt2(5,0,50)
+ u1      = opt2(0.1,0,1)
+ alpha   = opt2(1,1e-3,10)
```

Optimization Sweeps

```
.DC DATA=gate  optimize=opt1  results=comp1  model=optmod
.MEAS DC comp1 ERR1 par(ids) i(m1) minval=1e-04 ignor=1e-05
.DC DATA=drain  optimize=opt2  results=comp2  model=optmod
.MEAS DC comp2 ERR1 par(ids) i(m1) minval=1e-04 ignor=1e-05
```

DC Data Sweeps

```
.DC DATA=gate
.DC DATA=drain
```

Print Sweeps

```
.PRINT DC vds=par(vds) vgs=par(vgs) im=i(m1) id=par(ids)
.PRINT DC vds=par(vds) vgs=par(vgs) im=i(m1) id=par(ids)
```

DC Sweep Data

```
$..data
.PARAM vds=0 vgs=0 vbs=0 ids=0
.DATA gate vds vgs vbs ids
1.000000e-01 1.000000e+00  0.000000e+00 1.655500e-05
1.000000e-01 5.000000e+00 -2.000000e+00 3.149500e-04
.ENDDATA
.DATA drain vds vgs vbs ids
2.500000e-01 2.000000e+00 0.000000e+00 2.809000e-04
5.000000e+00 5.000000e+00 0.000000e+00 4.861000e-03
.ENDDATA
.END
```

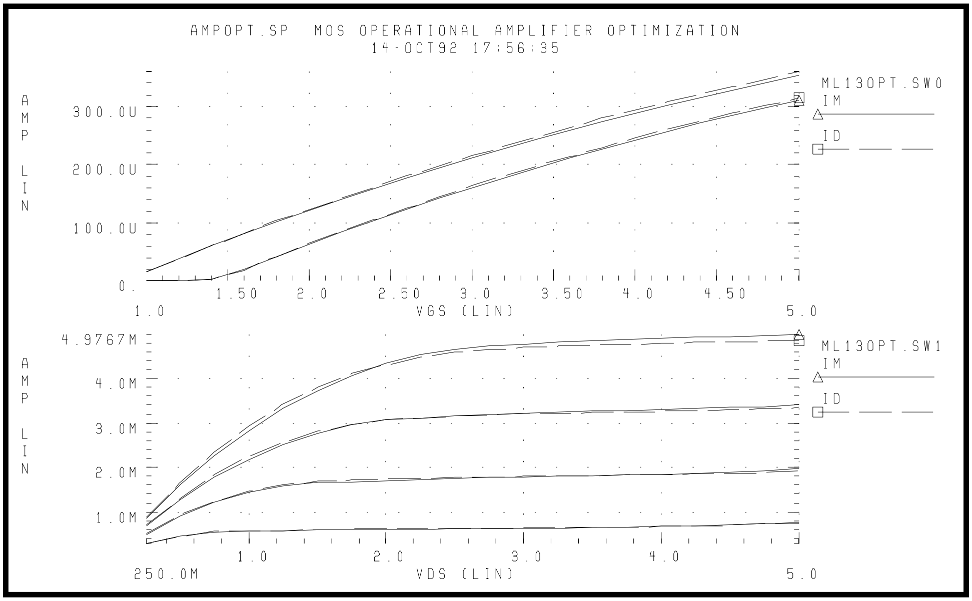

Figure 11-2: Level 13 MOSFET Optimization

RC Network Optimization

Following is an example of optimizing the power dissipation and time constant of an RC network. The circuit is a parallel resistor and capacitor. The following design targets are specified.

- 1 s time constant
- 50 mW rms power dissipation through the resistor

The Star-Hspice strategy is as follows:

- .MEASURE statement RC1 calculates RC time constant (GOAL of .3679 V corresponds to 1 s time constant e^{-rc}).
- .MEASURE statement RC2 calculates the rms power where the GOAL is 50 mW.
- OPTrc identifies RX and CX as optimization parameters and sets their starting, minimum, and maximum values.

Star-Hspice features used:

- Measure voltages and report times subject to goal
- Measure device power dissipation subject to goal
- Measure statements replace tabular or plot output
- Element value parameterization
- Parameter optimization function
- Transient with SWEEP optimize

RC Network Optimization Input Netlist File

```
.title RCOPT.sp
.option post

.PARAM RX=OPTRC(.5, 1E-2, 1E+2)
.PARAM CX=OPTRC(.5, 1E-2, 1E+2)

.MEASURE TRAN RC1 TRIG AT=0 TARG V(1) VAL=.3679 FALL=1
+ GOAL=1sec
.MEASURE TRAN RC2 RMS P(R1) GOAL=50mwatts

.MODEL OPT1 OPT

.tran .1 2        $ initial values
.tran .1 2 SWEEP OPTIMIZE=OPTrc RESULTS=RC1,RC2 MODEL=OPT1
.tran .1 2       $ analysis using final optimized values

.ic 1 1
R1 1 0 RX
c1 1 0 CX
```

Optimization Results

```
RESIDUAL SUM OF SQUARES     =  1.323651E-06
NORM OF THE GRADIENT        =  6.343728E-03
MARQUARDT SCALING PARAMETER =  2.799235E-06
NO. OF FUNCTION EVALUATIONS =       24
NO. OF ITERATIONS           =       12
```

Residual Sum of Squares

The residual sum of squares is a measure of the total error. The smaller this value is, the more accurate the optimization results are.

$$\text{residual sum of squares} = \sum_{i=1}^{ne} E_i^2$$

where *E* is the error function and *ne* is the number of error functions.

Norm of the Gradient

The norm of the gradient is another measure of the total error. The smaller this value is, the more accurate the optimization results are.

The gradient G is found by

$$G_j = \sum_{i=1}^{ne} E_i \cdot (\Delta E_i / \Delta P_j)$$

and

$$\text{norm of the gradient} = 2 \cdot \sqrt{\sum_{i=1}^{np} G_j^2}$$

where *P* is the parameter and *np* is the number of parameters to be optimized.

Marquardt Scaling Parameter

This parameter is used in the Levenberg-Marquardt algorithm to find the actual solution of the optimizing parameters. The search direction is a combination of the Steepest Descent method and the Gauss-Newton method.

The Steepest Descent method is used initially to approach the solution because it is fast, and then the Gauss-Newton method is used to find the solution. During this process, the Marquardt Scaling Parameter becomes very small, but starts to increase again if the solution starts to deviate. If this happens, the optimizer chooses between the two methods to work toward the solution again.

If the optimal solution is not attained, an error message is printed and a large Marquardt Scaling Parameter value is printed.

Number of Function Evaluations

This is the number of analyses (for example, finite difference or central difference) that were needed to find a minimum of the function.

Number of Iterations

This is the number of iterations needed to find the optimized or actual solution.

Optimized Parameters OPTRC

```
*                                    %NORM-SEN    %CHANGE
.PARAM RX            =   6.7937   $   54.5260    50.2976M
.PARAM CX            = 147.3697M  $   45.4740    33.7653M
```

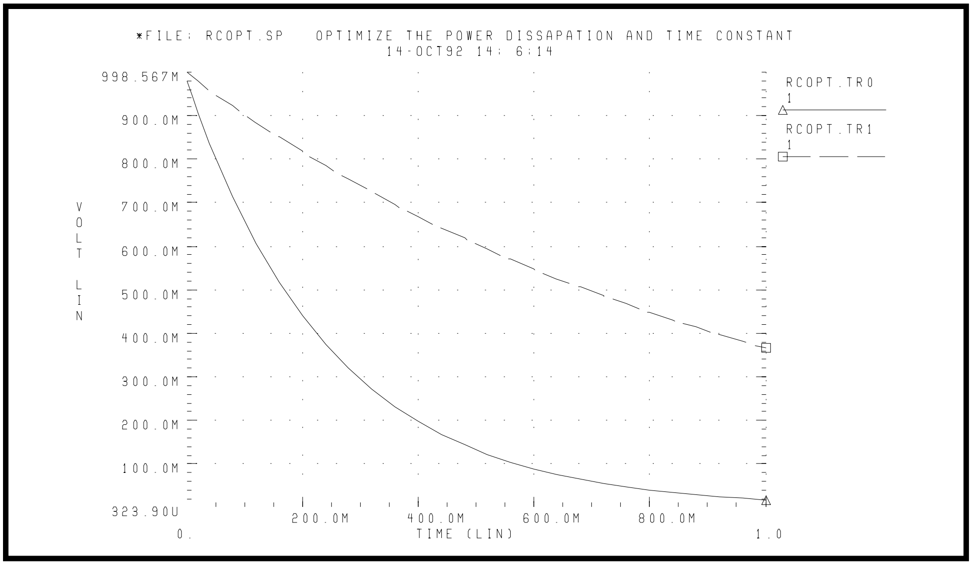

Figure 11-3: Power Dissipation and Time Constant (VOLT) RCOPT.TR0 = Before Optimization, RCOPT.TR1 = Optimized Result

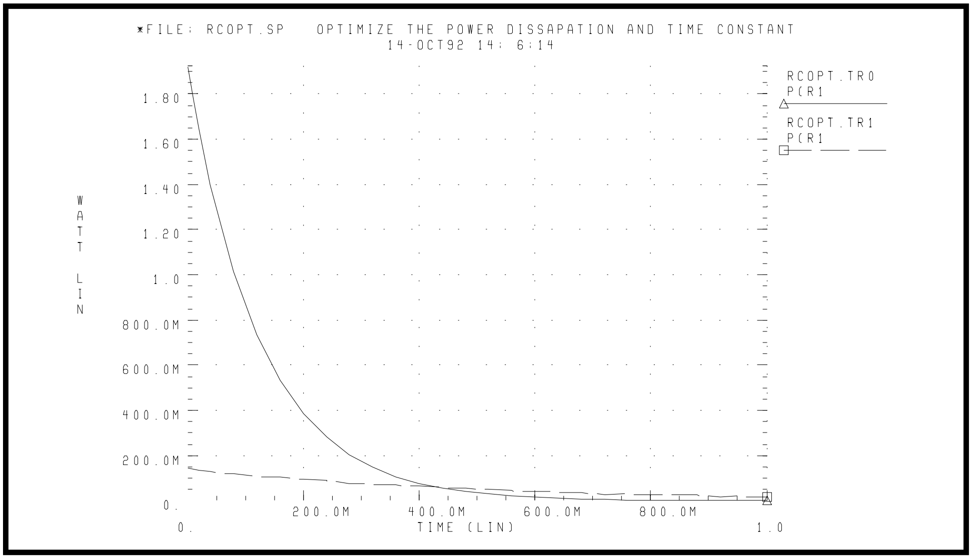

Figure 11-4: Power Dissipation and Time Constant (WATT) RCOPT.TR0 = Before Optimization, RCOPT.TR1 = Optimized Result

CMOS Tristate Buffer Optimization

The example circuit is an inverting CMOS tristate buffer. The following design targets are specified:

1. Rising edge delay of 5 ns (input 50% voltage to output 50% voltage)
2. Falling edge delay of 5 ns (input 50% voltage to output 50% voltage)
3. RMS power dissipation should be as low as possible
4. Output load consists of

   pad capacitance

   leadframe inductance

   50 pF capacitive load

The Star-Hspice strategy is as follows:

- Simultaneously optimize rising delay buffer and falling delay buffer.
- Set up internal power supplies and tristate enable as global nodes.
- Optimize all device widths except:
  - Initial inverter (this is assumed to be standard size)
  - Tristate inverter and part of tristate control (the optimization is not sensitive to this path)
- Perform initial transient analysis for plotting purposes, then optimize and perform a final transient for plotting purposes.
- Use a weighted RMS power measure by specifying an unrealistically low power goal and using MINVAL to attenuate the error.

CMOS Tristate Buffer Optimization Input Netlist File

```
*Tri-State I/O Optimization
.options defl=1.2u nomod post=2
+ relv=1e-3 absvar=.5 relvar=.01
```

Circuit Input

```
.global lgnd lvcc enb
.macro buff data out
mp1  DATAN DATA LVCC  LVCC p w=35u
mn1  DATAN DATA LGND  LGND n w=17u

mp2 BUS DATAN LVCC  LVCC  p w=wp2
mn2 BUS DATAN LGND  LGND  n w=wn2

mp3 PEN PENN  LVCC  LVCC  p  w=wp3
mn3 PEN PENN  LGND  LGND  n w=wn3

mp4 NEN NENN  LVCC  LVCC  p w=wp4
mn4 NEN NENN  LGND  LGND  n w=wn4

mp5 OUT PEN   LVCC  LVCC  p w=wp5 l=1.8u
mn5 OUT NEN   LGND  LGND  n w= wn5 l=1.8u

mp10 NENN  BUS  LVCC  LVCC p w=wp10
mn12 PENN  ENB  NENN LGND  n w=wn10
mn10 PENN  BUS  LGND  LGND n w=wn10
mp11 NENN  ENB  LVCC  LVCC p w=wp11
mp12 NENN  ENBN PENN LVCC p w=wp11
mn11 PENN  ENBN LGND  LGND n w=80u

mp13 ENBN ENB LVCC LVCC p w=35u
mn13 ENBN ENB LGND LGND n w=17u
cbus BUS  LGND  1.5pf
cpad OUT  LGND  5.0pf
.ends
* * input signals  *
vcc VCC GND 5V

lvcc vcc lvcc 6nh
lgnd lgnd gnd 6nh
vin  DATA    LGND pl (0v 0n, 5v 0.7n)
```

```
vinb DATAbar LGND pl (5v 0n, 0v 0.7n)
ven ENB GND 5V

** circuit **
x1 data out buff
cext1 out  GND  50pf
x2 databar outbar buff
cext2 outbar  GND  50pf
```

Optimization Parameters

```
.param
+      wp2=opt1(70u,30u,330u)
+      wn2=opt1(22u,15u,400u)
+      wp3=opt1(400u,100u,500u)
+      wn3=opt1(190u,80u,580u)
+      wp4=opt1(670u,150u,800u)
+      wn4=opt1(370u,50u,500u)
+      wp5=opt1(1200u,1000u,5000u)
+      wn5=opt1(600u,400u,2500u)
+      wp10=opt1(240u,150u,450u)
+      wn10=opt1(140u,30u,280u)
+      wp11=opt1(240u,150u,450u)
```

Control Section

```
.tran 1ns 15ns
.tran .5ns 15ns sweep optimize=opt1 results=tfopt,tropt,rmspowo
model=optmod
** put soft limit for power with minval setting (i.e. values
** less than 1000mw are less important)
.measure rmspowo rms  power goal=100mw  minval=1000mw
.mea tran tfopt  trig v(data) val=2.5 rise=1 targ v(out)
+ val=2.5 fall=1 goal 5.0n
.mea tran tropt trig v(databar) val=2.5 fall=1 targ
+ v(outbar) val=2.5 rise=1 goal 5.0n

.model optmod opt itropt=30  max=1e+5

.tran 1ns 15ns
* output section *
*.plot tran v(data) v(out)
.plot tran v(databar) v(outbar)
```

Model Section

```
.MODEL N NMOS LEVEL=3 VTO=0.7 UO=500 KAPPA=.25 KP=30U ETA=.03
+ THETA=.04 VMAX=2E5 NSUB=9E16 TOX=500E-10 GAMMA=1.5 PB=0.6
+ JS=.1M XJ=0.5U LD=0.0 NFS=1E11 NSS=2E10 CGSO=200P CGDO=200P
+ CGBO=300P
.MODEL P PMOS LEVEL=3 VTO=-0.8 UO=150 KAPPA=.25 KP=15U
+ ETA=.03 THETA=.04 VMAX=5E4 NSUB=1.8E16 TOX=500E-10 NFS=1E11
+ GAMMA=.672 PB=0.6 JS=.1M XJ=0.5U LD=0.0
+ NSS=2E10 CGSO=200P CGDO=200P CGBO=300P
.end
```

Optimization Results

```
residual sum of squares     =  2.388803E-02
norm of the gradient        =  0.769765
marquardt scaling parameter =   12624.2
no. of function evaluations =      175
no. of iterations           =       23
```

Optimization Completed

```
Parameters relin=  1.0000E-03 on last iterations
```

Optimized Parameters OPT1

```
*                                  %norm-sen    %change

.param wp2          =  84.4981u  $   22.5877  -989.3733u
.param wn2          =  34.1401u  $    7.6568  -659.2874u
.param wp3          = 161.7354u  $  730.7865m -351.7833u
.param wn3          = 248.6829u  $    8.1362    -2.2416m
.param wp4          = 238.9825u  $    1.2798    -1.5774m
.param wn4          =  61.3509u  $  315.4656m   43.5213m
.param wp5          =   1.7753m  $    4.1713     2.1652m
.param wn5          =   1.0238m  $    5.8506   413.9667u
.param wp10         = 268.3125u  $    8.1917    -2.0266m
.param wn10         = 115.6907u  $   40.5975  -422.8411u
.param wp11         = 153.1344u  $  482.0655m  -30.6813m
*** optimize results measure names and values
* tfopt          =   5.2056n
* tropt          =   5.5513n
* rmspowo        = 200.1808m
```

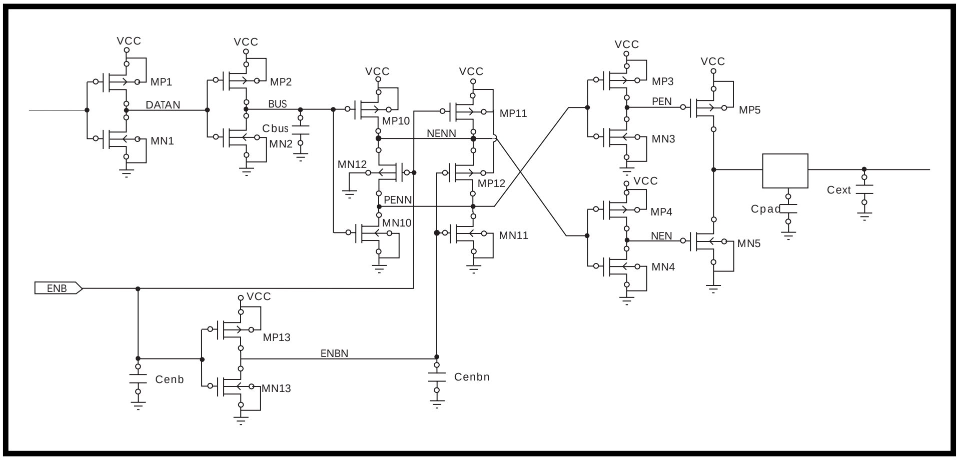

Figure 11-5: Tristate Buffer Optimization Circuit

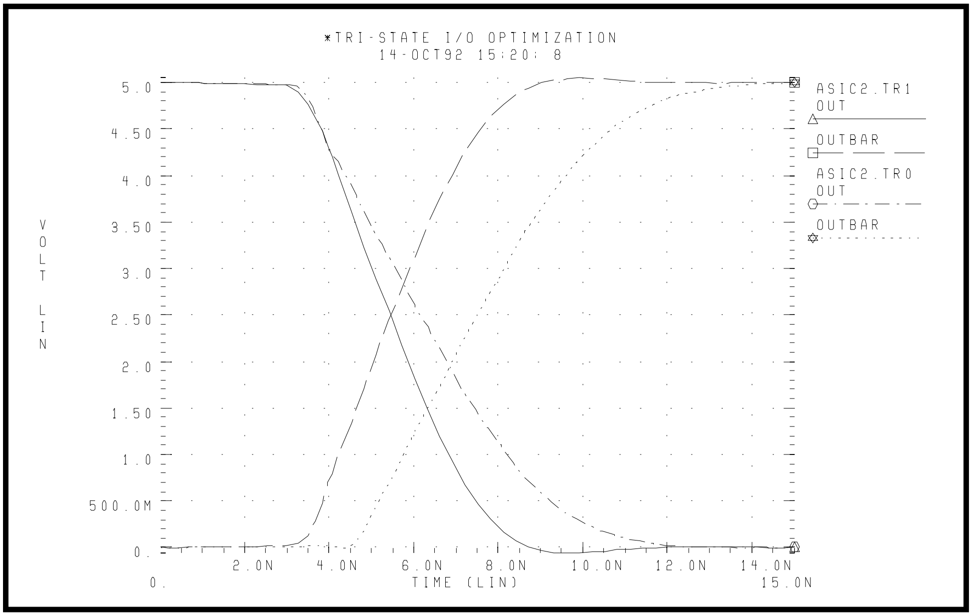

Figure 11-6: Tristate I/O Optimization
ACIC2B.TR0 = Before Optimization, ACIC2B.TR1 = Optimized Result

BJT S-Parameters Optimization

In the following example, the s-parameters are optimized to match those given for a set of measurements. These measured s-parameters, as a function of frequency, are in the ".DATA MEASURED" in-line data statement. The model parameters of the microwave transistor (LBB, LCC, LEE, TF, CBE, CBC, RB, RE, RC, and IS) are varied so that measured s-parameters in the .DATA statement matches the calculated s-parameters from the simulation results.

This optimization uses a 2n6604 microwave transistor and an equivalent circuit consisting of a BJT with parasitic resistances and inductances. The BJT is biased at a 10 mA collector current (0.1 mA base current at DC bias and bf=100).

Key Star-Hspice Features Used

- NET command to simulate network analyzer action
- AC optimization
- Optimization of element and model parameters
- Optimization comparing measured s-parameters versus calculated parameters
- S-parameters used in magnitude and phase (real and imaginary available)
- Data-driven frequency versus s-parameter table weighting used for phase domain

BJT S-Parameters Optimization Input Netlist File

```
* BJTOPT.SP BJT S-PARAMETER OPTIMIZATION
.OPTION ACCT NOMOD POST=2
```

BJT Equivalent Circuit Input

```
* THE NET COMMAND IS AUTOMATICALLY REVERSING THE SIGN OF
* THE POWER SUPPLY CURRENT FOR THE NETWORK CALCULATIONS
.NET I(VCE) IBASE ROUT=50 RIN=50
VCE  VCE 0 10V
IBASE 0 IIN   AC=1 DC=.1MA
LBB  IIN BASE  LBB
LCC  VCE COLLECT LCC
LEE  EMIT 0 LEE
```

```
Q1 COLLECT BASE EMIT T2N6604
.MODEL T2N6604 NPN RB=RB  BF=100 TF=TF CJE=CBE CJC=CBC
+ RE=RE RC=RC  IS=IS
.PARAM
+ LBB= OPT1(100P, 1P, 10N)
+ LCC= OPT1(100P, 1P, 10N)
+ LEE= OPT1(100P, 1P, 10N)
+ TF = OPT1(1N,  5P,  5N)
+ CBE= OPT1(.5P, .1P, 5P)
+ CBC= OPT1(.4P, .1P, 5P)
+  RB= OPT1(10, 1, 300)
+  RE= OPT1(.4, .01, 5)
+  RC= OPT1(10, .1, 100)
+  IS= OPT1(1E-15, 1E-16, 1E-10)
.AC DATA=MEASURED  OPTIMIZE=OPT1
+   RESULTS=COMP1,COMP3,COMP5,COMP6,COMP7
+   MODEL=CONVERGE
.MODEL CONVERGE OPT RELIN=1E-4 RELOUT=1E-4 CLOSE=100 ITROPT=25
.MEASURE AC COMP1 ERR1 PAR(S11M) S11(M)
.MEASURE AC COMP2 ERR1 PAR(S11P) S11(P) MINVAL=10
.MEASURE AC COMP3 ERR1 PAR(S12M) S12(M)
.MEASURE AC COMP4 ERR1 PAR(S12P) S12(P) MINVAL=10
.MEASURE AC COMP5 ERR1 PAR(S21M) S21(M)
.MEASURE AC COMP6 ERR1 PAR(S21P) S21(P) MINVAL=10
.MEASURE AC COMP7 ERR1 PAR(S22M) S22(M)
.AC DATA=MEASURED
.PRINT PAR(S11M) S11(M) PAR(S11P) S11(P)
.PRINT PAR(S12M) S12(M) PAR(S12P) S12(P)
.PRINT PAR(S21M) S21(M) PAR(S21P) S21(P)
.PRINT PAR(S22M) S22(M) PAR(S22P) S22(P)
.DATA MEASURED
FREQ   S11M   S11P   S21M    S21P   S12M   S12P S22M   S22P
100ME  .6    -52     19.75   148    .02     65    .87   - 21
200ME  .56   -95     15.30   127    .032    49    .69   - 33
500ME  .56   -149     7.69    97    .044    41    .45   - 41
1000ME .58   -174     4.07    77    .061    42    .39   - 47
2000ME .61    159     2.03    50    .095    40    .39   - 70
.ENDDATA
.PARAM FREQ=100ME S11M=0, S11P=0, S21M=0, S21P=0, S12M=0,
+  S12P=0, S22M=0, S22P=0
.END
```

Optimization Results

```
RESIDUAL SUM OF SQUARES     =  5.142639e-02
NORM OF THE GRADIENT        =  6.068882e-02
MARQUARDT SCALING PARAMETER =  0.340303
CO. OF FUNCTION EVALUATIONS =  170
NO. OF ITERATIONS           =  35
```

The maximum number of iterations (25) was exceeded. However, the results probably are accurate. Increase ITROPT accordingly.

Optimized Parameters OPT1– Final Values

```
***OPTIMIZED PARAMETERS OPT1 SENS  %NORM-SEN
.PARAM  LBB  =  1.5834N  $ 27.3566X  2.4368
.PARAM  LCC  =  2.1334N  $ 12.5835X  1.5138
.PARAM  LEE  =723.0995P  $254.2312X 12.3262
.PARAM  TF   = 12.7611P  $  7.4344G 10.0532
.PARAM  CBE  =620.5195F  $ 23.0855G  1.5300
.PARAM  CBC  =  1.0263P  $346.0167G 44.5016
.PARAM  RB   =  2.0582   $ 12.8257M  2.3084
.PARAM  RE   =869.8714M  $ 66.8123M  4.5597
.PARAM  RC   = 54.2262   $  3.1427M 20.7359
.PARAM  IS   = 99.9900P  $  3.6533X 34.4463M
```

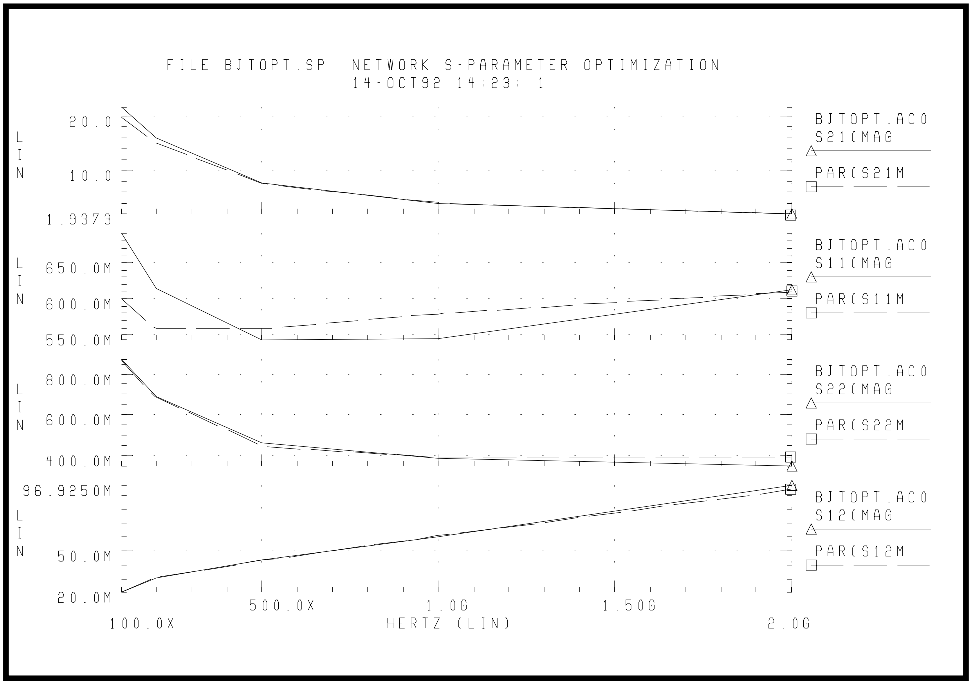

Figure 11-7: BJT-S Parameter Optimization

BJT Model DC Optimization

The goal is to match the forward and reverse Gummel plots obtained from a HP4145 semiconductor analyzer with the Star-Hspice LEVEL=1 Gummel-Poon BJT model. Since the Gummel plots are at low base currents, the base resistance is not optimized. The forward and reverse Early voltages (VAF and VAR) are not optimized, since no VCE data was measured.

The key feature used in this optimization is incremental optimization. First the forward Gummel data points are optimized. The forward optimized parameters are updated into the model and not allowed to change. Then the reverse Gummel data points are optimized.

BJT Model DC Optimization Input Netlist File

```
* FILE OPT_BJT.SP BJT OPTIMIZATION T2N3947
* OPTIMIZE THE DC FORWARD AND REVERSE CHARACTERISTICS FROM A
* GUMMEL PLOT
* ALL DC GUMMEL-POON DC PARAMETERS EXCEPT BASE RESISTANCE AND
* EARLY VOLTAGES OPTIMIZED
*
$..TIGHTEN THE SIMULATOR CONVERGENCE PROPERTIES
.OPTION NOMOD INGOLD=2 NOPAGE VNTOL=1E-10 POST
+ NUMDGT=5 RELI=1E-4 RELV=1E-4
$..OPTIMIZATION CONVERGENCE CONTROLS
.MODEL OPTMOD OPT RELIN=1E-4 ITROPT=30 GRAD=1E-5 CLOSE=10
+ CUT=2 CENDIF=1E-6 RELOUT=1E-4 MAX=1E6
```

Room Temp Device

```
VBER BASE 0 VBE
VBCR BASE COL VBC
Q1 COL BASE 0 BJTMOD
```

Model and Initial Estimates

```
.MODEL BJTMOD NPN
+ ISS = 0. XTF = 1. NS = 1.
+ CJS = 0. VJS = 0.50000 PTF = 0.
+ MJS = 0. EG = 1.10000 AF = 1.
+ ITF = 0.50000 VTF = 1.00000
+ FC = 0.95000 XCJC = 0.94836
+ SUBS = 1
+ TF=0.0 TR=0.0 CJE=0.0 CJC=0.0 MJE=0.5 MJC=0.5 VJE=0.6
+ VJC=0.6 RB=0.3 RC=10 VAF=550 VAR=300
$..THESE ARE THE OPTIMIZED PARAMETERS
+ BF=BF IS=IS IKF=IKF ISE=ISE RE=RE
+ NF=NF NE=NE
$..THESE ARE FOR REVERSE BASE OPT
+ BR=BR IKR=IKR ISC=ISC
+ NR=NR NC=NC

.PARAM VBE=0 IB=0 IC=0 VCE_EMIT=0 VBC=0 IB_EMIT=0 IC_EMIT=0
+ BF= OPT1( 100, 50, 350)
+ IS= OPT1( 5E-15, 5E-16, 1E-13)
+ NF= OPT1( 1.0, 0.9, 1.1)
+ IKF=OPT1( 50E-3, 1E-3, 1)
+ RE= OPT1( 10, 0.1, 50)
+ ISE=OPT1( 1E-16, 1E-18, 1E-11)
+ NE= OPT1( 1.5, 1.2, 2.0)

+ BR= OPT2( 2, 1, 10)
+ NR= OPT2( 1.0, 0.9, 1.1)
+ IKR=OPT2( 50E-3, 1E-3, 1)
+ ISC=OPT2( 1E-12, 1E-15, 1E-10)
+ NC= OPT2( 1.5, 1.2, 2.0)

.DC DATA=BASEF SWEEP OPTIMIZE=OPT1 RESULTS=IBVBE,ICVBE
+ MODEL=OPTMOD
.MEAS DC IBVBE ERR1 PAR(IB) I2(Q1) MINVAL=1E-14 IGNORE=1E-16
.MEAS DC ICVBE ERR1 PAR(IC) I1(Q1) MINVAL=1E-14 IGNORE=1E-16

.DC DATA=BASER SWEEP OPTIMIZE=OPT2 RESULTS=IBVBER,ICVBER
+ MODEL=OPTMOD
.MEAS DC IBVBER ERR1 PAR(IB) I2(Q1) MINVAL=1E-14 IGNORE=1E-16
.MEAS DC ICVBER ERR1 PAR(IC) I1(Q1) MINVAL=1E-14 IGNORE=1E-16

.DC DATA=BASEF
.PRINT DC PAR(IC) I1(Q1) PAR(IB) I2(Q1)
.DC DATA=BASER
.PRINT DC PAR(IC) I1(Q1) PAR(IB) I2(Q1)
```

Optimization Results

```
RESIDUAL SUM OF SQUARES = 2.196240E-02
```

Optimized Parameters OPT1

```
*                                %NORM-SEN %CHANGE

.PARAM BF = 1.4603E+02 $ 2.7540E+00 -7.3185E-07
.PARAM IS = 2.8814E-15 $ 3.7307E+00 -5.0101E-07
.PARAM NF = 9.9490E-01 $ 9.1532E+01 -1.0130E-08
.PARAM IKF = 8.4949E-02 $ 1.3782E-02 -8.8082E-08
.PARAM RE = 6.2358E-01 $ 8.6337E-02 -3.7665E-08
.PARAM ISE = 5.0569E-16 $ 1.0221E-01 -3.1041E-05
.PARAM NE = 1.3489E+00 $ 1.7806E+00 2.1942E-07
```

Optimization Results

```
RESIDUAL SUM OF SQUARES = 1.82776
```

Optimized Parameters OPT2

```
*                                %NORM-SEN %CHANGE

.PARAM BR = 1.0000E+01 $ 1.1939E-01 1.7678E+00
.PARAM NR = 9.8185E-01 $ 1.4880E+01 -1.1685E-03
.PARAM IKR = 7.3896E-01 $ 1.2111E-03 -3.5325E+01
.PARAM ISC = 1.8639E-12 $ 6.6144E+00 -5.2159E-03
.PARAM NC = 1.2800E+00 $ 7.8385E+01 1.6202E-03
```

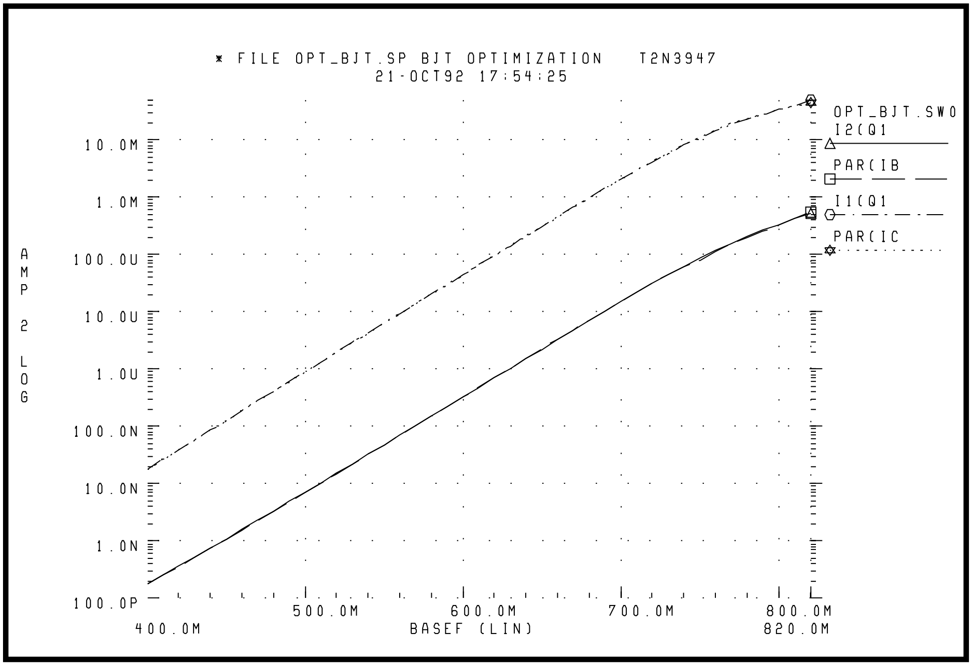

Figure 11-8: BJT Optimization Forward Gummel Plots

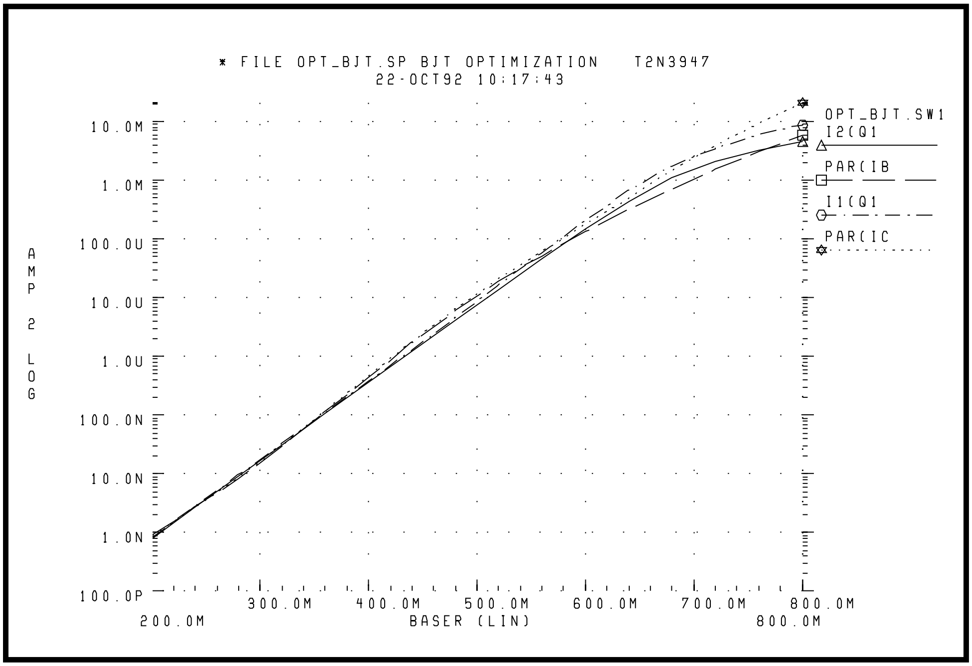

Figure 11-9: BJT Optimization Reverse Gummel Plots

GaAsFET Model DC Optimization

This example circuit is a high performance GaAsFET transistor. The design target is to match HP4145 DC measured data to the Star-Hspice LEVEL=3 JFET model.

The Star-Hspice strategy is as follows:

- MEASURE IDSERR is an ERR1 type function providing linear attenuation of the error results, starting at 20 mA and ignoring all currents below 1 mA. The high current fit is the most important for this model.
- The OPT1 optimization function allows all DC parameters to be simultaneously optimized.
- The .DATA statement merges raw data files *TD1.dat* and *TD2.dat* together.
- The graph plot model sets the parameter MONO=1 to remove the retrace lines from the family of curves.

GaAsFET Model DC Optimization Input Netlist File

```
*FILE JOPT.SP JFET OPTIMIZATION
.OPTIONS ACCT NOMOD POST
+ RELI=2E-4 RELV=2E-4

VG GATE 0 XVGS
VD DRAIN 0 XVDS
J1 DRAIN GATE 0 JFETN1

.MODEL JFETN1 NJF LEVEL=3 CAPOP=1 SAT=3
+ NG=1
+ CGS=1P CGD=1P RG=1
+ VTO=VTO BETA=BETA LAMBDA=LAMBDA
+ RS=RDS RD=RDS IS=1E-15 ALPHA=ALPHA
+ UCRIT=UCRIT SATEXP=SATEXP
+ GAMDS=GAMDS VGEXP=VGEXP

.PARAM
+ VTO=OPT1(-.8,-4,0)
+ VGEXP=OPT1(2,1,3.5)
+ GAMDS=OPT1(0,-.5,0)
+ BETA= OPT1(6E-3, 1E-5,9E-2)
+ LAMBDA=OPT1(30M,1E-7,5E-1)
+ RDS=OPT1(1,.001,40)
+ ALPHA=OPT1(2,1,3)
+ UCRIT=OPT1(.1,.001,1)
+ SATEXP=OPT1(1,.5,3)

.DC DATA=DESIRED OPTIMIZE=OPT1 RESULTS=IDSERR MODEL=CONV
.MODEL CONV OPT RELIN=1E-4 RELOUT=1E-4 CLOSE=100 ITROPT=25
.MEASURE DC IDSERR ERR1 PAR(XIDS) I(J1) MINVAL=20M IGNORE=1M

.DC DATA=DESIRED
.GRAPH PAR(XIDS) I(J1)
.MODEL GRAPH PLOT MONO=1
.PRINT PAR(XVGS) PAR(XIDS) I(J1)

.DATA DESIRED MERGE
+ FILE=JDC.DAT XVDS=1 XVGS=2 XIDS=3
.ENDDATA
.END
```

Optimization Results

```
RESIDUAL SUM OF SQUARES = 7.582202E-02
```

Optimized Parameters Opt1

```
*                                          %NORM-SEN    %CHANGE
.PARAM VTO           = -1.1067   $   64.6110      43.9224M
.PARAM VGEXP         = 2.9475    $   13.2024     219.4709M
.PARAM GAMDS         = 0.        $    0.            0.
.PARAM BETA          = 11.8701M  $   17.2347     136.8216M
.PARAM LAMBDA        = 138.9821M $    2.2766      -1.5754
.PARAM RDS           = 928.3216M $  704.3204M   464.0863M
.PARAM ALPHA         = 2.2914    $  728.7492M     168.4004M
.PARAM UCRIT         = 1.0000M   $   18.2438M   -125.0856
.PARAM SATEXP        = 1.4211    $    1.2241        2.2218
```

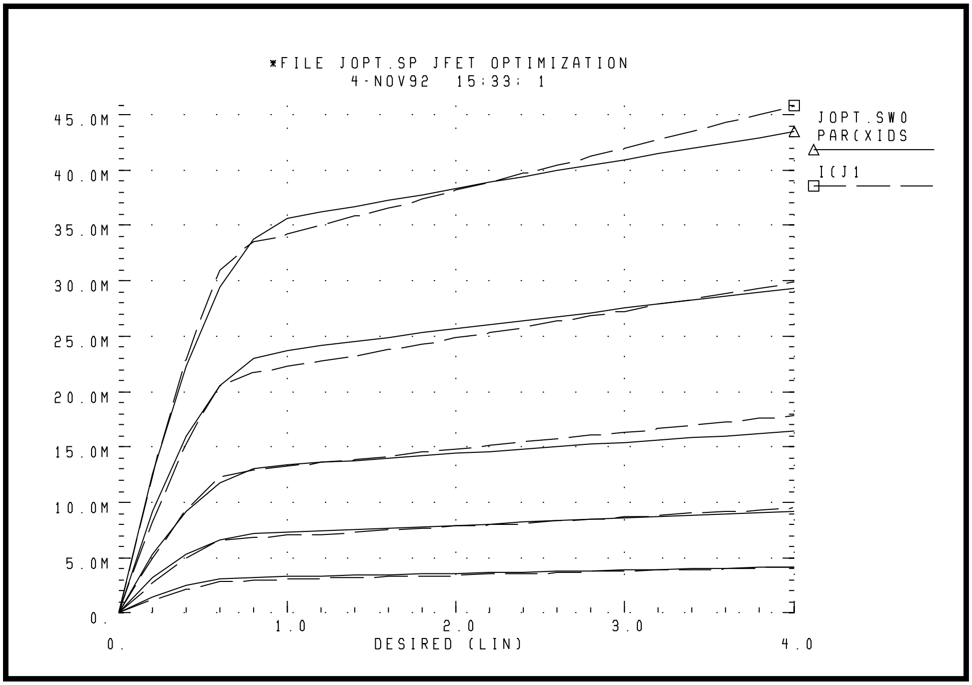

Figure 11-10: JFET Optimization

MOS Op-amp Optimization

The design goals for the MOS operational amplifier are:

- Minimize the gate area (and hence the total cell area)
- Minimize the power dissipation
- Open-loop transient step response of 100 ns for rising and falling edges

The Star-Hspice strategy is:

- Simultaneous optimization of two amplifier cells for rising and falling edges
- Total power is power for two cells
- The optimization transient analysis must be longer to allow for a range of values in intermediate results
- All transistor widths and lengths are optimized
- Transistor area is calculated algebraically, a voltage value is used, and the resulting voltage is minimized
- The transistor area measure statement uses MINVAL to give less weighting to the area minimization
- Bias voltage is optimized

MOS Op-amp Optimization Input Netlist File

```
AMPOPT.SP MOS OPERATIONAL AMPLIFIER OPTIMIZATION

.OPTION RELV=1E-3 RELVAR=.01 NOMOD ACCT POST
.PARAM VDD=5   VREF='VDD/2'
VDD VSUPPLY 0 VDD
VIN+    VIN+     0 PWL(0 ,'VREF-10M'   10NS 'VREF+10M' )
VINBAR+ VINBAR+  0 PWL(0 ,'VREF+10M'   10NS 'VREF-10M' )
VIN- VIN-  0  VREF
VBIAS VBIAS 0 BIAS
.GLOBAL VSUPPLY VBIAS

XRISE VIN+ VIN- VOUTR AMP
CLOADR VOUTR 0 .4P
XFALL VINBAR+ VIN- VOUTF AMP
CLOADF VOUTF 0 .4P
```

```
.MACRO AMP VIN+ VIN- VOUT
M1 2     VIN-   3       3       MOSN W=WM1 L=LM
M2 4     VIN+   3       3       MOSN W=WM1 L=LM
M3 2     2      VSUPPLY VSUPPLY MOSP W=WM1 L=LM
M4 4     2      VSUPPLY VSUPPLY MOSP W=WM1 L=LM
M5 VOUT  VBIAS  0       0       MOSN W=WM5 L=LM
M6 VOUT  4      VSUPPLY VSUPPLY MOSP W=WM6 L=LM
M7 3     VBIAS  0       0       MOSN W=WM7 L=LM
.ENDS
.PARAM AREA='4*WM1*LM + WM5*LM + WM6*LM + WM7*LM'
VX 1000 0 AREA
RX 1000 0 1K

.MODEL MOSP PMOS (VTO=-1 KP=2.4E-5 LAMBDA=.004
+ GAMMA =.37 TOX=3E-8 LEVEL=3)
.MODEL MOSN NMOS (VTO=1.2 KP=6.0E-5 LAMBDA=.0004
+ GAMMA =.37 TOX=3E-8 LEVEL=3)

.PARAM WM1=OPT1(60U,20U,100U)
+      WM5=OPT1(40U,20U,100U)
+      WM6=OPT1(300U,20U,500U)
+      WM7=OPT1(70U,40U,200U)
+       LM=OPT1(10U,2U,100U)
+     BIAS=OPT1(2.2,1.2,3.0)

.TRAN 2.5N 300N SWEEP OPTIMIZE=OPT1
+     RESULTS=DELAYR,DELAYF,TOT_POWER,AREA  MODEL=OPT
.MODEL OPT OPT CLOSE=100

.TRAN 2N 150N
.MEASURE DELAYR TRIG AT=0 TARG V(VOUTR) VAL=2.5 RISE=1
+ GOAL=100NS
.MEASURE DELAYF TRIG AT=0 TARG V(VOUTF) VAL=2.5 FALL=1
+ GOAL=100NS
.MEASURE TOT_POWER AVG POWER GOAL=10MW
.MEASURE AREA MIN PAR(AREA) GOAL=1E-9  MINVAL=100N
.PRINT V(VIN+) V(VOUTR) V(VOUTF)
.END
```

Optimization Results

```
RESIDUAL SUM OF SQUARES     =  4.654377E-04
NORM OF THE GRADIENT        =  6.782920E-02
```

Optimized Parameters Opt1

```
*                                 %NORM-SEN  %CHANGE

.PARAM WM1          =  47.9629U  $    1.6524  -762.3661M
.PARAM WM5          =  66.8831U  $   10.1048    23.4480M
.PARAM WM6          = 127.1928U  $   12.7991    22.7612M
.PARAM WM7          = 115.8941U  $    9.6104  -246.4540M
.PARAM LM           =   6.2588U  $   20.3279  -101.4044M
.PARAM BIAS         =   2.7180   $   45.5053     5.6001M

*** OPTIMIZE RESULTS MEASURE NAMES AND VALUES
* DELAYR         = 100.4231N
* DELAYF         =  99.5059N
* TOT_POWER      =  10.0131M
* AREA           =   3.1408N
```

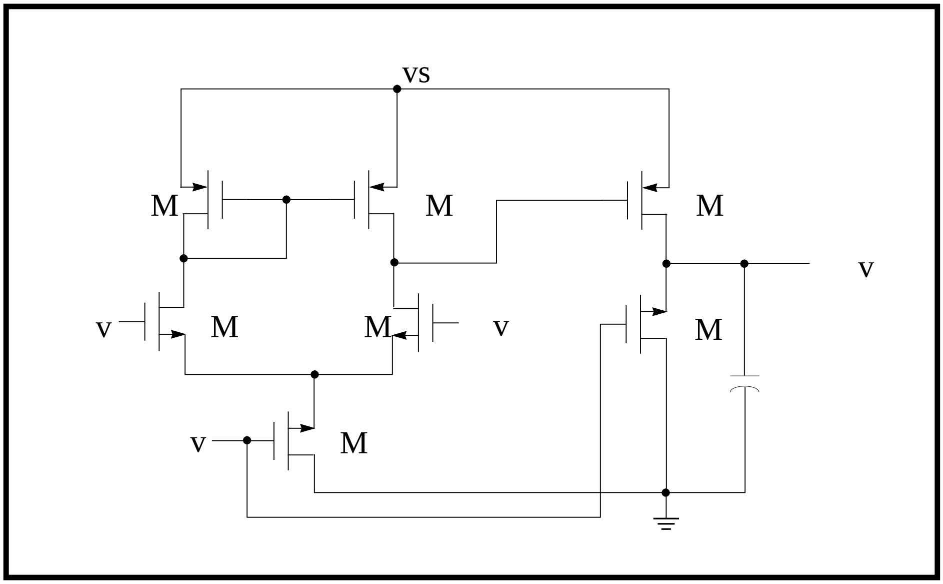

Figure 11-11: CMOS Op-amp

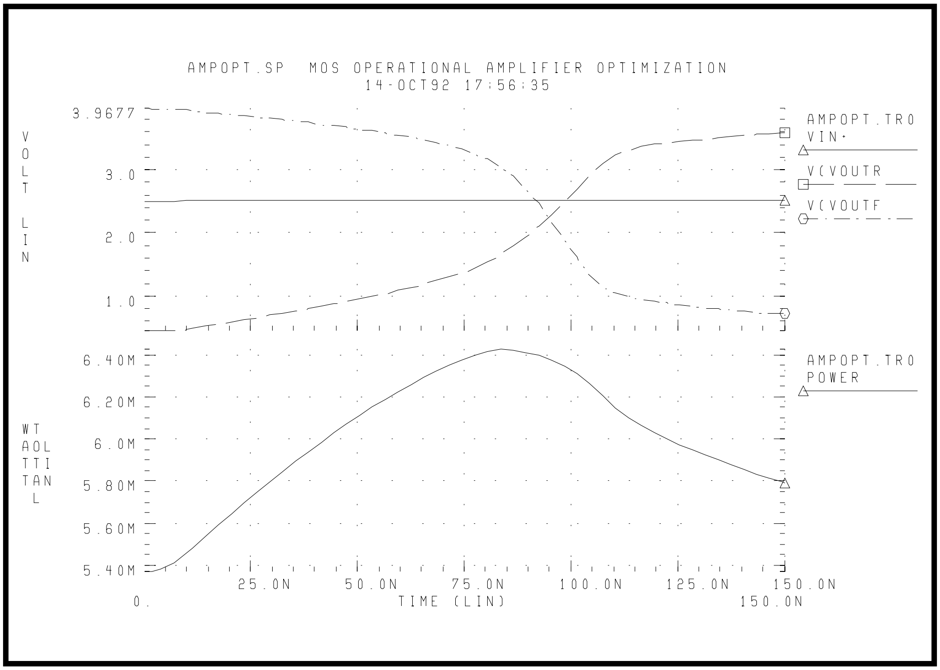

Figure 11-12: Operational Amplifier Optimization

Performing Timing Analysis

To analyze circuit timing violation, Star-Hspice uses a binary search algorithm to generate a set of operational parameters that produce a failure in the required behavior of the circuit. When a circuit timing failure occurs, you can identify a timing constraint that can lead to a design guideline. Typical types of timing constraint violations include:

- Data setup time before clock
- Data hold time after clock
- Minimum pulse width required to allow a signal to propagate to the output
- Maximum toggle frequency of the component(s)

Bisection is a method of optimization that finds the value of an input variable (target value) associated with a goal value of an output variable. The input and output variables can be of various types (for example, voltage, current, delay time, or gain) related by some transfer function. You can use the bisection feature in a pass-fail mode or a bisection mode. The process is largely the same in each case.