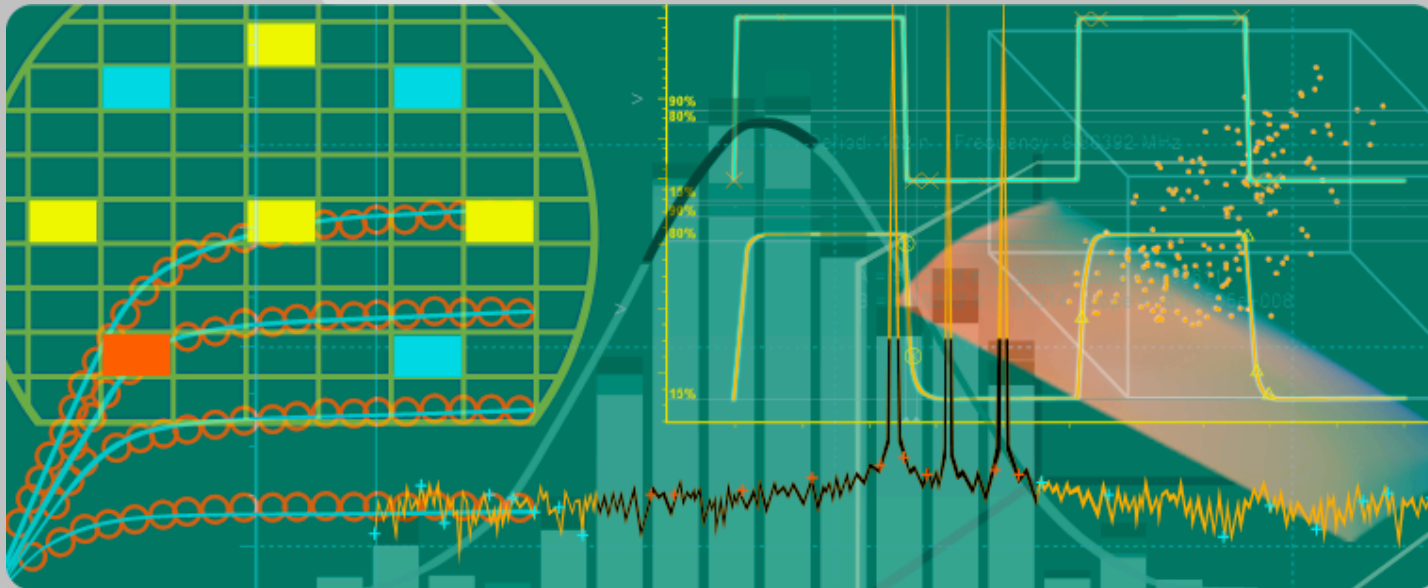


SmartSpice Training Program



Part 6: SmartSpice Lab Instructions

SmartSpice Optimizer Lab

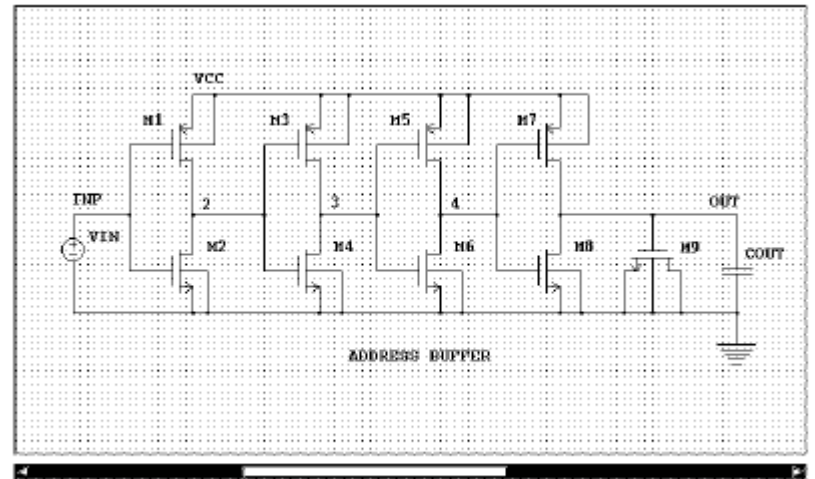


Figure 2-2: Address Buffer Circuit

- For the circuit shown in Figure 2-2, the optimization targets at the temperature 75oC are:
 - Rise delay of 3ns
 - Fall delay of 3ns

The parameters for optimization are the widths of transistors: m3, m4, m5, m6, m7 and m8. In the input file, the .MODIF statement sets the temperature to a value of 75oC, and the power supply voltage vccdc to 4.5V. The .MODIF statement then performs the performance measure optimization. The rise time del_r and the fall time del_f are calculated using the .MEASURE statements. The rise time is defined as the time difference between the mid-point of the input signal's rising edge and the corresponding point of the output signal. The fall time is calculated from the mid-points of the falling edges.

SmartSpice Optimizer Lab Example

```
Address Buffer
* Optimization of rise and fall delays
***** Circuit
vin inp 0 DC 1.5 PULSE(0.4 2.6 0 5N 5N 15N)
vcc vss 0 DC vccdc
m1 2 inp vss vss pm W=5.067U L=1.6U
m2 2 inp 0 0 nm W=34.82U L=2.0U
m3 3 2 vss vss pm W= 50u L=1.1U
m4 3 2 0 0 nm W=70.U L=1.0U
m5 4 3 vss vss pm W=320.U L=1.1U
m6 4 3 0 0 nm W=440.U L=1.1U
m7 out 4 vss vss pm W=parm7w L=1.1U
m8 out 4 0 0 nm W=parm8w L=1.1U
m9 0 out 0 0 nm W=380U L=1.2U
COUT out 0 700p
***** Models
.MODEL nm NMOS (LEVEL = 3 TOX=.02e-6)
.MODEL pm PMOS (LEVEL = 3 TOX=.02e-6)
***** Analysis statement
.TRAN 0.2NS 40NS CALLV SAVEV
***** Parameter labels
.PARAM parm7w=500u parm8w=500u vccdc=4.5

***** Measure statements
.MEASURE TRAN del_R DELAY v(inp) RISE=1 VAL=1.5
+ TARG=v(out) RISE=1 VAL= 'vccdc/2'
.MEASURE TRAN del_F DELAY v(inp) FALL=1 VAL=1.5
+ TARG=v(out) FALL=1 VAL= 'vccdc/2'
***** Optimization specifications
.MODIF TEMP= 75 vccdc=4.5
+ OPTIMIZE parm7w=opt(10u 3000u 500U)
+ parm8w=opt(10u 3000u 500U)
+ m3(w)=opt(10u 300u 50U)
+ m4(w)=opt(10u 300u 70U)
+ m5(w)=opt(100u 1000u 320U)
+ m6(w)=opt(50u 1000u 440U)
+ TARGETS del_f=3ns del_r=3ns
+ OPTIONS AVG=0.001 MEASOFF=0
***** Interactive plot
.OPTIONS iplot_one
.IPLOT v(out) v(inp)
.END
```

SmartSpice Optimizer Lab Results #1

The optimization results are shown in the following tables.

Parameters

Name	Final Value	Init. Value
parm7w	= 8.57060e-04	5.00000e-04
parm8w	= 1.24485e-03	5.00000e-04
m3(w)	= 3.12675e-05	5.00000e-05
m4(w)	= 5.27244e-05	7.00000e-05
m5(w)	= 2.49000e-04	3.20000e-04
m6(w)	= 2.08049e-04	4.40000e-04

TARGETS

Name	Final Value	Init. Value
del_f	= 3.02174e-09	5.75936e-09
del_r	= 3.00062e-09	4.80122e-09
Number of iterations	=	8
Number of func evals	=	21
Number of Jac evals	=	2
Residual sum of squares	=	2.821e-04
Norm of the gradient	=	2.165e-02
Marquardt parameter	=	1.000e-05
RMS relative error	=	5.126e-03
Average relative error	=	3.727e-03
Maximum relative error	=	7.247e-03
Termination code	=	7 : RMS criterion

SmartSpice Optimizer Lab Results Graph

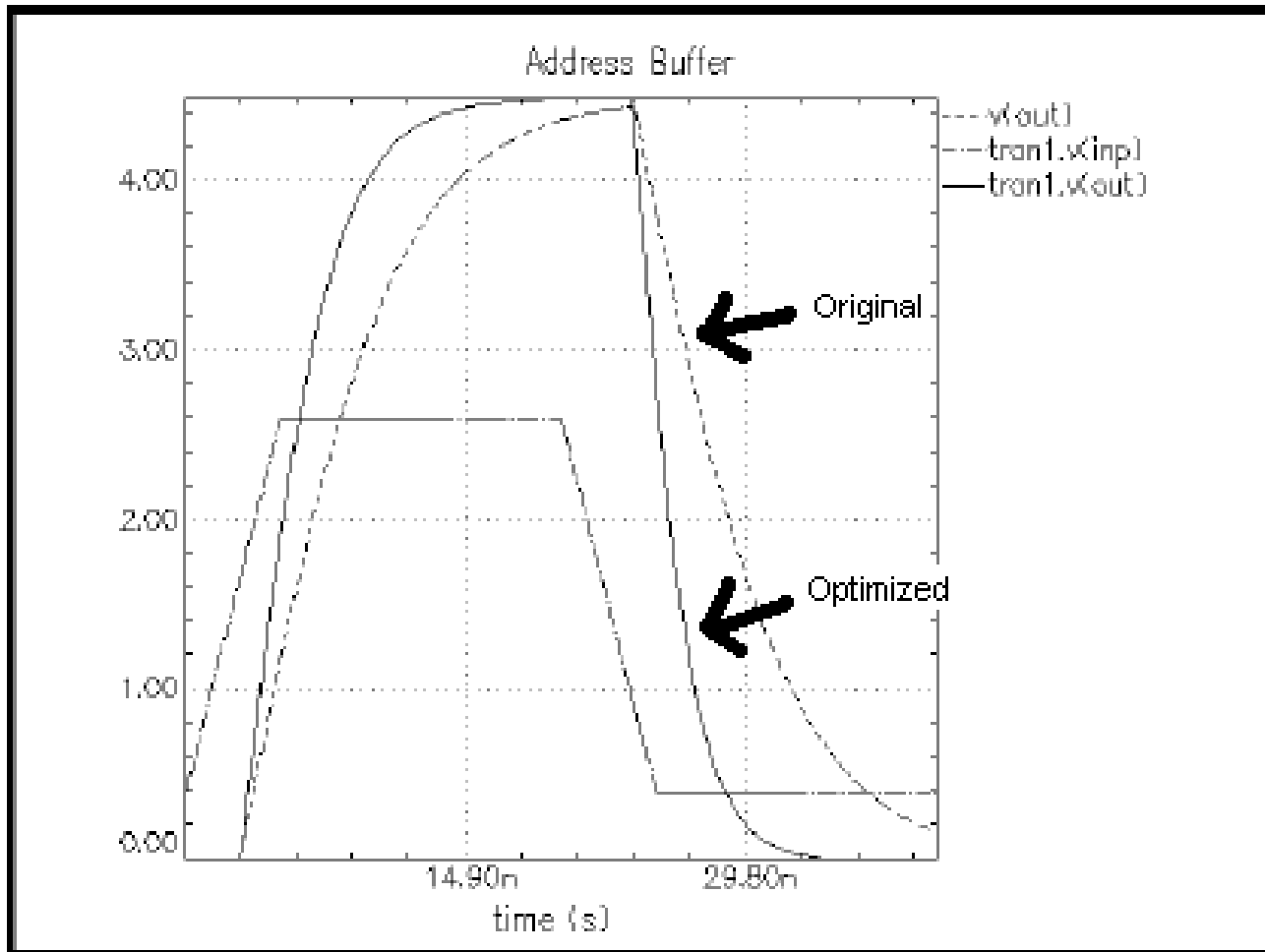


Figure 2-3: Delay Optimization Plot

SmartSpice Verilog-A Lab

- Scenario: You have taken over a partially completed project from another engineer and been asked to review and complete the task ASAP. Using only the example code provided earlier in this presentation complete the project by modifying the existing code and constructing a top level verilog-A model (VTOP) and evaluate what happens. Test a 10MHz ramp simulated for 257 clks as it passes through a system with the following architecture:
- DAC - 1MHz LPF - 10kHz HPF - Gain=2 amp - 2.5MHz Sample & Hold - 8bit ADC
- VDD=5V VREF=2V RESOLUTION=8bits
- Find and correct any existing design errors and maximize the signal quality by tuning the components of the system and note the runtime of the corrected design.
- Then create a SPICE top level (TOP) that utilizes the verilog-A components and note the runtime difference.
- Then replace ALL verilog-A modules of the SPICE top level circuit, (except the 8-bit src reg and the ADC) with the provided transistor designs, correct any existing design errors and compare the performance and runtime differences vs. the verilog-A models.
- Write a report highlighting the errors found and the runtimes of each top level circuit.
- EXTRA CREDIT!!! – using the optimizer to help you, modify the verilog-A model to better match the actual circuit performance of the corrected transistor design.

SmartSpice Verilog-A Lab SPICE deck

```
*file: smartspice_training.in
.VERILOG "smartspice_training.va"
.INCLUDE "smartspice_training.net"
.OPTIONS POST
.GLOBAL VDD VSS
.PARAM TDAC=100N TADC=400NS
.MODEL P PMOS LEVEL=3
.MODEL N NMOS LEVEL=3

.SUBCKT MTOP CLKDIN DIN0 DIN1 DIN2 DIN3 DIN4 DIN5 DIN6 DIN7 CLKDOUT MDOUT0 MDOUT1 MDOUT2 MDOUT3 MDOUT4 MDOUT5 MDOUT6
MDOUT7 AVDD AVSS
YVLG_DF0 Q0 QBAR0 CLKDIN DIN0 VDFF
YVLG_DF1 Q1 QBAR1 CLKDIN DIN1 VDFF
YVLG_DF2 Q2 QBAR2 CLKDIN DIN2 VDFF
YVLG_DF3 Q3 QBAR3 CLKDIN DIN3 VDFF
YVLG_DF4 Q4 QBAR4 CLKDIN DIN4 VDFF
YVLG_DF5 Q5 QBAR5 CLKDIN DIN5 VDFF
YVLG_DF6 Q6 QBAR6 CLKDIN DIN6 VDFF
YVLG_DF7 Q7 QBAR7 CLKDIN DIN7 VDFF
YVLG_DAC DACOUT Q0 Q1 Q2 Q3 Q4 Q5 Q6 Q7 VDAC
YVLG_LPF LPFOUT DACOUT VLPF
YVLG_HPF HPFOUT LPFOUT VHPF
YVLG_AMP AMPOUT HPFOUT VAMP
YVLG_SH SHOUT AMPOUT CLKDOUT VSH
YVLG_ADC DOUT0 DOUT1 DOUT2 DOUT3 DOUT4 DOUT5 DOUT6 DOUT7 SHOUT CLKDOUT VADC
.ENDS MTOP

.SUBCKT TOP CLKDIN DIN0 DIN1 DIN2 DIN3 DIN4 DIN5 DIN6 DIN7 CLKDOUT DOUT0 DOUT1 DOUT2 DOUT3 DOUT4 DOUT5 DOUT6 DOUT7 AVDD
AVSS
YVLG_DF0 Q0 QBAR0 CLKDIN DIN0 VDFF
YVLG_DF1 Q1 QBAR1 CLKDIN DIN1 VDFF
YVLG_DF2 Q2 QBAR2 CLKDIN DIN2 VDFF
YVLG_DF3 Q3 QBAR3 CLKDIN DIN3 VDFF
YVLG_DF4 Q4 QBAR4 CLKDIN DIN4 VDFF
```

SmartSpice Verilog-A Lab SPICE deck (cont'd)

```

YVLG_DF5 Q5      QBAR5 CLKDIN  DIN5  VDFF
YVLG_DF6 Q6      QBAR6 CLKDIN  DIN6  VDFF
YVLG_DF7 Q7      QBAR7 CLKDIN  DIN7  VDFF
XDAC      DACOUT Q0      Q1      Q2      Q3      Q4      Q5      Q6      Q7      AVDD  AVSS DAC
XLPF      LPFOUT DACOUT  AVSS    LPF
XHPF      HPFOUT LPFOUT  AVSS    HPF
XAMP      AMPOUT HPFOUT  AVDD    AVSS    AMP
XSH       SHOUT  AMPOUT CLKDOU  AVDD    AVSS    SH
YVLG_ADC DOUT0  DOUT1  DOUT2  DOUT3  DOUT4  DOUT5  DOUT6  DOUT7  SHOUT  CLKDOU  VADC
.ENDS TOP

AVDD      AVDD      0 5V
AVSS      AVSS      0 0V
VDD       VDD       0 5V
VSS       VSS       0 0V
VDINCLK  DINCLK  0 PULSE(0V 5V 1N 1N 1N ' TDAC/2'      TDAC )
VDIN0    DIN0    0 PULSE(0V 5V 0N 1N 1N ' 1*TDAC' ' 2*TDAC')
VDIN1    DIN1    0 PULSE(0V 5V 0N 1N 1N ' 2*TDAC' ' 4*TDAC')
VDIN2    DIN2    0 PULSE(0V 5V 0N 1N 1N ' 4*TDAC' ' 8*TDAC')
VDIN3    DIN3    0 PULSE(0V 5V 0N 1N 1N ' 8*TDAC' ' 16*TDAC')
VDIN4    DIN4    0 PULSE(0V 5V 0N 1N 1N ' 16*TDAC' ' 32*TDAC')
VDIN5    DIN5    0 PULSE(0V 5V 0N 1N 1N ' 32*TDAC' ' 64*TDAC')
VDIN6    DIN6    0 PULSE(0V 5V 0N 1N 1N ' 64*TDAC' ' 128*TDAC')
VDIN7    DIN7    0 PULSE(0V 5V 0N 1N 1N ' 128*TDAC' ' 256*TDAC')
VDOUTCLK DOUTCLK 0 PULSE(0V 5V 0N 1N 1N ' TADC/2'      TADC )

YVLG_TOP  CLKDIN DIN0  DIN1  DIN2  DIN3  DIN4  DIN5  DIN6  DIN7  CLKDOU  VDOUT0  VDOUT1  VDOUT2  VDOUT3  VDOUT4  VDOUT5  VDOUT6  VDOUT7
VTOP
XTOP_MIXED CLKDIN DIN0  DIN1  DIN2  DIN3  DIN4  DIN5  DIN6  DIN7  CLKDOU  MDOUT0  MDOUT1  MDOUT2  MDOUT3  MDOUT4  MDOUT5  MDOUT6  MDOUT7
AVDD AVSS MTOP
XTOP      CLKDIN DIN0  DIN1  DIN2  DIN3  DIN4  DIN5  DIN6  DIN7  CLKDOU  DOUT0  DOUT1  DOUT2  DOUT3  DOUT4  DOUT5  DOUT6  DOUT7
AVDD AVSS TOP

.TRAN 1N '257*TADC'
.END

```


SmartSpice Verilog-A Lab SPICE netlist (part I)

```
*file: smartspice_training.net
```

```
.SUBCKT INV OUT IN
MP OUT IN VDD VDD P W=2U L=1U
MN OUT IN VSS VSS N W=1U L=1U
.ENDS INV
```

```
.SUBCKT NAND Y A B
MPA Y A VDD VDD P W=2U L=1U
MPB Y B VDD VDD P W=2U L=1U
MNA Y A AB VSS N W=1U L=1U
MNB AB B VSS VSS N W=1U L=1U
.ENDS NAND
```

```
.SUBCKT DFF      Q QBAR CLK D
MPCLK           CLKN          CLK      VDD          VDD P W=2U L=1U
MNCLK           CLKN          CLK      VSS          VSS N W=1U L=1U
```

```
MPQNCLKMSTR    PQNCLKMSTR  CLK      VDD          VDD P W=2U L=1U
MPQN           QN          D        PQCLKMSTR    VDD P W=2U L=1U
MNQN           QN          D        NQCLKMSTR    VSS N W=1U L=1U
MNQNCLKMSTR    NQNCLKMSTR  CLKN     VSS          VSS N W=1U L=1U
```

```
MPQ            Q            QN       VDD          VDD P W=2U L=1U
MNQ            Q            QN       VSS          VSS N W=1U L=1U
```

```
MPQBAR         QBAR         Q         VDD          VDD P W=2U L=1U
MNQBAR         QBAR         Q         VSS          VSS N W=1U L=1U
```

```
MPQCLKSLV     PQCLKSLV     CLKN     VDD          VDD P W=2U L=1U
MPQ           Q            QN       PQCLKSLV     VDD P W=2U L=1U
MNQ           Q            QN       NQCLKSLV     VSS N W=1U L=1U
MNQCLKSLV     NQCLKSLV     CLK      VSS          VSS N W=1U L=1U
.ENDS DFF
```

```
.SUBCKT LPF OUT IN VSSA
R1 OUT IN 75
C1 OUT AVSS 270PF
.ENDS LPF
```

SmartSpice Verilog-A Lab SPICE netlist (part II)

```
.SUBCKT HPF OUT IN AVSS
C1 OUT IN 1U
R1 OUT AVSS 75
.ENDS HPF

.SUBCKT AMP OUT IN AVDD AVSS
.PARAM VBIAS=0.7V
MP1 OUTN OUTN AVDD AVDD P W=1U L=1U
MP2 OUT OUTN AVDD AVDD P W=1U L=1U
MPP VCMN IN OUT AVDD P W=1U L=1U
MPN VCMN OUT OUTN AVDD P W=1U L=1U
MNB VCMN NBIAS AVSS AVSS N W=1U L=1U
VBIAS NBIAS AVSS VBIAS
.ENDS AMP

.SUBCKT SH OUT IN CLK VDDA VSSA
MPS OUT CLKN IN AVDD P W=2U L=1U
MNS OUT CLK IN AVSS N W=1U L=1U

MPI CLKN CLK VDD VSS P W=2U L=1U
MNI CLKN CLK VSS VSS N W=1U L=1U
CHOLD OUT AVSS 10PF
.ENDS SH

.SUBCKT DIN0 DIN1 DIN2 DIN3 DIN4 DIN5 DIN6 DIN7 AVDD AVSS DAC
.PARAM VBIAS=0.7V
XINV0 NDIN0 DIN0 INV M=1
XINV1 NDIN1 DIN1 INV M=2
XINV2 NDIN2 DIN2 INV M=4
XINV3 NDIN3 DIN3 INV M=8
XINV4 NDIN4 DIN4 INV M=16
XINV5 NDIN5 DIN5 INV M=32
XINV6 NDIN6 DIN6 INV M=64
XINV7 NDIN7 DIN7 INV M=128

MPSW0 AVSS DIN0 I0 AVDD P W=1U L=1U M=1
MPSW1 AVSS DIN1 I1 AVDD P W=1U L=1U M=2
MPSW2 AVSS DIN2 I2 AVDD P W=1U L=1U M=4
MPSW3 AVSS DIN3 I3 AVDD P W=1U L=1U M=8
MPSW4 AVSS DIN4 I4 AVDD P W=1U L=1U M=16
MPSW5 AVSS DIN5 I5 AVDD P W=1U L=1U M=32
MPSW6 AVSS DIN6 I6 AVDD P W=1U L=1U M=64
MPSW7 AVSS DIN7 I7 AVDD P W=1U L=1U M=128
```

SmartSpice Verilog-A Lab SPICE netlist (part III)

```
MPSWB0 OUT NDIN0 I0 AVDD P W=1U L=1U M=1
MPSWB1 OUT NDIN1 I1 AVDD P W=1U L=1U M=2
MPSWB2 OUT NDIN2 I2 AVDD P W=1U L=1U M=4
MPSWB3 OUT NDIN3 I3 AVDD P W=1U L=1U M=8
MPSWB4 OUT NDIN4 I4 AVDD P W=1U L=1U M=16
MPSWB5 OUT NDIN5 I5 AVDD P W=1U L=1U M=32
MPSWB6 OUT NDIN6 I6 AVDD P W=1U L=1U M=64
MPSWB7 OUT NDIN7 I7 AVDD P W=1U L=1U M=128

MPI0 I0 VBIAS AVDD AVDD P W=1U L=1U M=1
MPI1 I1 VBIAS AVDD AVDD P W=1U L=1U M=2
MPI2 I2 VBIAS AVDD AVDD P W=1U L=1U M=4
MPI3 I3 VBIAS AVDD AVDD P W=1U L=1U M=8
MPI4 I4 VBIAS AVDD AVDD P W=1U L=1U M=16
MPI5 I5 VBIAS AVDD AVDD P W=1U L=1U M=32
MPI6 I6 VBIAS AVDD AVDD P W=1U L=1U M=64
MPI7 I7 VBIAS AVDD AVDD P W=1U L=1U M=128

ROUT OUT AVSS 75
VBIAS VBIAS AVSS VBIAS
.ENDS DAC
```

SmartSpice Verilog-A netlist (part I)

```
// file: smartspice_training.va
`include "discipline.h"
`include "constants.h"
module VINV(out, in);
  input    in;
  output   out;
  electrical in, out;
  parameter real vdd    = 5.0;    //supply voltage
  parameter real vss    = 0.0;    //gnd voltage
  parameter real vth    = vdd / 2; //threshold
  parameter real tdelay = 5n;     //propagation delay
  parameter real trise  = 8.5n;   //rise time
  parameter real tfall  = 10n;   //fall time
  real    val;
  analog begin
    @( initial_step )
    begin // initial condition
      if ( V(in) > vth ) val = vss;
      else val = vdd;
    end
    @(cross(V(in) - vth, +1)) val = vss; // Vin > vth => vss
    @(cross(V(in) - vth, -1)) val = vdd; // Vin < vth => vdd
    V(out) <+ transition(val, tdelay, trise, tfall);
  end
endmodule
module VDFF(q, qbar, clk, d);
  input    clk, d;
  output   q, qbar;
  electrical q, qbar, clk, d;
  parameter real tdelay  = 5n,
               ttransit = 5n,
               vdd       = 5.0,
               vss       = 0.0,
               vth       = vdd / 2;

  integer    x;
  analog begin
    @(initial_step)
    x = 0;
    @(cross(V(clk) - vth, +1 )) x = (V(d) > vth);
    V(q)    <+ transition( vdd * x + vss * !x, tdelay, ttransit);
    V(qbar) <+ transition( vdd * !x + vss * x, tdelay, ttransit);
  end
endmodule
```

SmartSpice Verilog-A netlist (part II)

```
module VLPF(out, in);
  input    in;
  output   out;
  electrical in, out;
  parameter real freq_p1 = 1MHz;
  analog V(out) <+ laplace_nd(V(in), {1}, {1, 1 / (`M_TWO_PI * freq_p1)});
endmodule

module VHPF(out, in);
  input    in;
  output   out;
  electrical in, out;
  parameter real freq_p1 = 10KHz;
  analog V(out) <+ laplace_nd(V(in), {0, 1 / (`M_TWO_PI * freq_p1)}, {1, 1 / (`M_TWO_PI * freq_p1)});
endmodule

// simple analog buffer
module VAMP(out, in);
  input    in;
  output   out;
  electrical in, out;
  //parameter real gain = 10;
  parameter avdd = 5.0; //analog supply voltage
  parameter avss = 0.0; //analog gnd voltage
  parameter vref = 0.0; //input reference voltage
  parameter real R1 = 150; //initial resistance
  parameter real R2 = 150; //initial resistance
  real A, gain, vout;
  analog begin
    @(initial_step) begin
      gain = R2 / R1;
      A = gain + 1; //non-invert signal
    end
    vout = A * (V(in) - vref) + vref;
    if (vout > avdd / 2)
      vout = avdd / 2;
    if (vout < avss)
      vout = avss;
    V(out) <+ vout;
  end
endmodule
```

SmartSpice Verilog-A netlist (part III)

```
module VSH(out, in, clk);
  input    in, clk;
  output   out;
  electrical in, out, clk;
  parameter real avdd = 5.0,      // analog supply voltage
             vth = avdd / 2; // threshold
  real      v;
  analog begin
    @(initial_step)
      v = V(in);
    if (analysis("static") || (V(clk) > vth))
      v = V(in); // passing phase
    @(cross(V(clk) - vth, 0))
      v = V(in); // sampling phase
    V(out) <+ v;
  end
endmodule
//----- DAC -----//
module VDAC(out, IN0, IN1, IN2, IN3, IN4, IN5, IN6, IN7);
  parameter real supply = 5.0 // supply voltage
             fullscale = 2.0; //reference voltage
  parameter integer maxbit = 8, bit = 8; //Resolution
  input    IN0, IN1, IN2, IN3, IN4, IN5, IN6, IN7;
  output   out;
  electrical IN0, IN1, IN2, IN3, IN4, IN5, IN6, IN7, out;
  electrical [0:bit-1] in;
  real      vlump[maxbit-1:0];
  real      vout[maxbit-1:0]; //voltage
  real      outv, vth;
  integer   i; //index loop
  integer   code[bit-1:0]; //digital code analog begin
  analog begin
    @(initial_step) begin
      in[0] = IN0;
      in[1] = IN1;
      in[2] = IN2;
      in[3] = IN3;
      in[4] = IN4;
      in[5] = IN5;
```

SmartSpice Verilog-A netlist (part IV)

```
in[6] = IN6;
  in[7] = IN7;
  vth = supply / 2;
  for (i = 0; i <= maxbit; i = i + 1) begin
    vlump[i] = fullscale / pow(2, i + 1);
  end
end
for (i = 0; i <= bit; i = i + 1) begin
  if (V(in[i-1]) > vth) begin
    code[i-1] = 1;
  end else begin
    code[i-1] = 0;
  end
  vout[i] = vlump[i] * code[i];
end
if (bit < maxbit) begin
  for (i = maxbit - 1; i > bit; i = i - 1) begin
    vout[i] = 1;
  end
end
outv = vout[0] + vout[1] + vout[2] + vout[3] + vout[4] + vout[5] + vout[6] + vout[7];
// V(out) <+ transition(outv, 50n, 50n, 50n);
V(out) <+ outv;
end
endmodule
//----- Pipelined ADC-----
module VADC (OUT0, OUT1, OUT2, OUT3, OUT4, OUT5, OUT6, OUT7, in, clk);
  parameter integer bit = 8; // ADC resolution
  parameter real supply = 5.0 // supply voltage
  parameter real fullscale = 2.0, // reference voltage
  vth = fullscale / 2, // threshold
  dly = 10n, // transition delay
  ttime = 1n; // transition rising time

  input in; // input analog voltage
  input clk; // input clock
  output OUT0, OUT1, OUT2, OUT3, OUT4, OUT5, OUT6, OUT7;
  electrical in, clk;
  electrical OUT0, OUT1, OUT2, OUT3, OUT4, OUT5, OUT6, OUT7;
  real sample;
```

SmartSpice Verilog-A netlist (part V)

```
integer    result[bit-1:0];          // integer array
integer    i;                        // index loop
analog begin
    @(cross(V(clk) - vth, +1) ) begin
        sample = V(in);
        for (i = bit - 1; i >= 0; i = i - 1) begin
            if( sample>vth ) begin
                result[i] = supply;
                sample = sample - vth;
            end else begin
                result[i] = 0.0;
            end
            sample = 2.0 * sample;
        end
    end
    V(OUT0) <+ transition(result[0], dly, ttime);
    V(OUT1) <+ transition(result[1], dly, ttime);
    V(OUT2) <+ transition(result[2], dly, ttime);
    V(OUT3) <+ transition(result[3], dly, ttime);
    V(OUT4) <+ transition(result[4], dly, ttime);
    V(OUT5) <+ transition(result[5], dly, ttime);
    V(OUT6) <+ transition(result[6], dly, ttime);
    V(OUT7) <+ transition(result[7], dly, ttime);
end
Endmodule
// ----- Top Level -----
module VTOP (CLKDIN,  DIN0,  DIN1,  DIN2,  DIN3,  DIN4,  DIN5,  DIN6,  DIN7,
             CLKDOUT, DOUT0, DOUT1, DOUT2, DOUT3, DOUT4, DOUT5, DOUT6, DOUT7);
    input    CLKDIN, DIN0,  DIN1,  DIN2,  DIN3,  DIN4,  DIN5,  DIN6,  DIN7, CLKDOUT;
    output   DOUT0,  DOUT1, DOUT2, DOUT3, DOUT4, DOUT5, DOUT6, DOUT7;
    electrical CLKDIN, DIN0,  DIN1,  DIN2,  DIN3,  DIN4,  DIN5,  DIN6,  DIN7, CLKDOUT;
    electrical DOUT0,  DOUT1, DOUT2, DOUT3, DOUT4, DOUT5, DOUT6, DOUT7;
    df0 VDFE (q0,    qbar0, CLKDIN, DIN0);
    df1 VDFE (q1,    qbar1, CLKDIN, DIN1);
    df2 VDFE (q2,    qbar2, CLKDIN, DIN2);
    df3 VDFE (q3,    qbar3, CLKDIN, DIN3);
    df4 VDFE (q4,    qbar4, CLKDIN, DIN4);
    df5 VDFE (q5,    qbar5, CLKDIN, DIN5);
    df6 VDFE (q6,    qbar6, CLKDIN, DIN6);
```

SmartSpice Verilog-A netlist (part VI)

```
df7 VDFE (q7,      qbar7, CLKDIN, DIN7);
  dac VDAC (dacout, q0,      q1,      q2,      q3,      q4,      q5,      q6,      q7);
  lpf VLPF (lpfout, dacout);
  hpf VHPF (hpfout, lpfout);
  amp VAMP (ampout, hpfout);
  sh VSH (shout,      ampout, CLKDOUT);
  adc VADC (DOUT0,      DOUT1,      DOUT2,      DOUT3,      DOUT4,      DOUT5,      DOUT6,      DOUT7, shout, CLKDOUT);
endmodule
```