ΔΣ ADC Design Examples by Schreier’s DS Toolbox

Derek Chen

5/18/2015

Resources

- Tutorial:

Brian Young's final report & presentation slides

- Schreier's Toolbox and manual:

http://www.mathworks.com/matlabcentral/fileexchange/19-delta-sigma-toolbox

- Simulink toolbox by Prof. Maloberti & Malcovati

  http://ims.unipv.it/Courses/Dataconv.php

- Prepare yourself

Project Assignment

ECE 627 PROJECT

Design of a Video Delta-Sigma A/D Converter

Due: June 11, 2010, 5 pm.

Design a delta-sigma ADC for the following specifications:

Signal bandwidth	0 – 10 MHz
Clock frequency	Less than 700 MHz
Accuracy	At least 16 bits

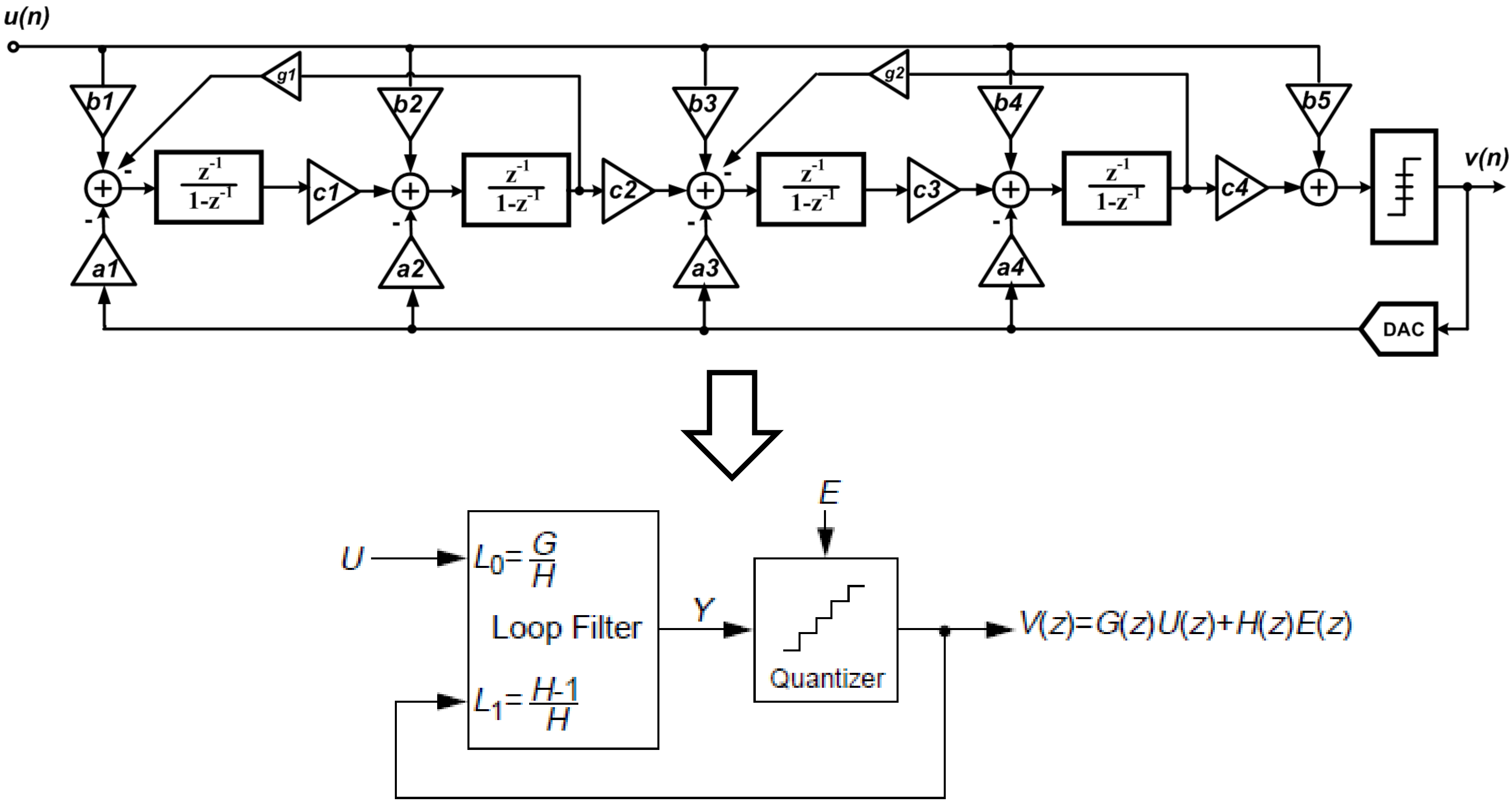
Pre-design

- SQNR, SNR
  SNR is limited by thermal noise (kT/C)
  SQNR> SNR +10 dB. Determined by Order, Quantizer's level.

- Architecture of the modulator
  feedback, feedforward.

- Simulation techniques
  number of simulation samples and spectra by FFT

Typical ΔΣ Modulators



$$V(z) = STF(z) \cdot U(z) + NTF(z) \cdot E(z) = \frac{L_0(z)}{1 + L_1(z)} \cdot U(z) + \frac{1}{1 + L_1(z)} \cdot E(z)$$

1. SQNR vs. Order, OSR

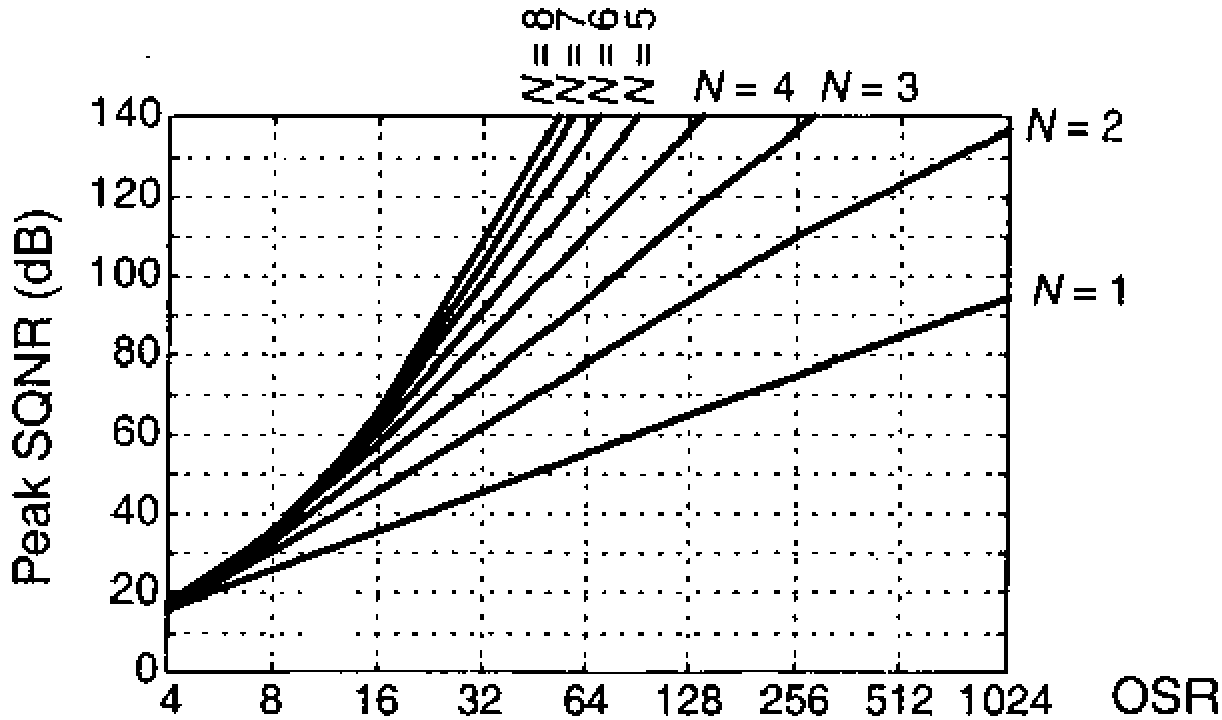
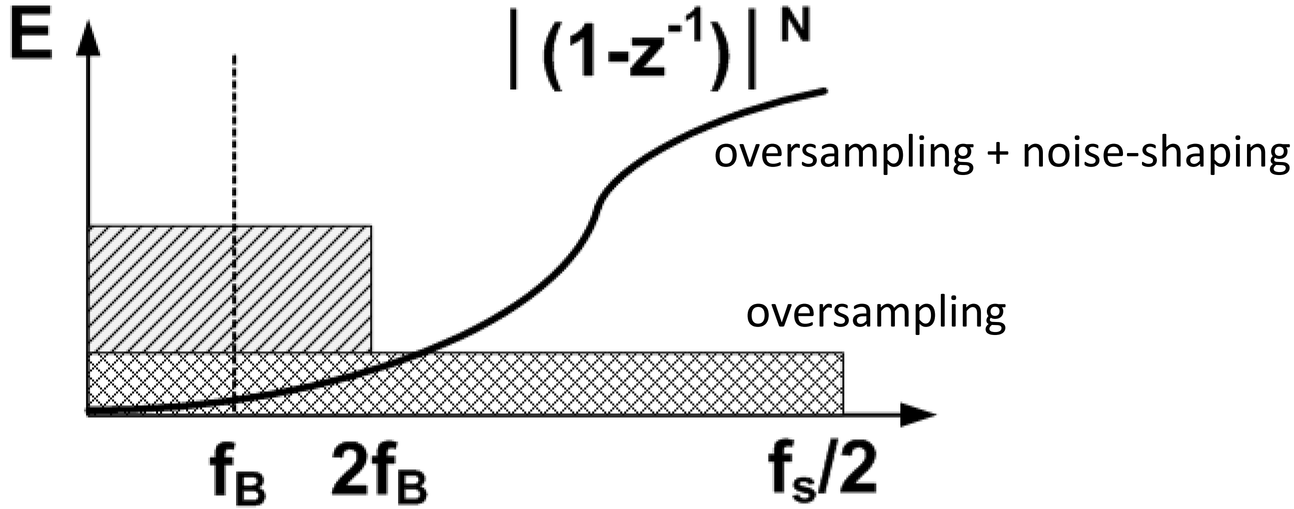


Figure 4.14: Empirical SQNR limit for 1-bit modulators of order *N*.

Peak SQNR estimation



$$P_E = \int \frac{V_{LSB}}{12} \cdot \frac{1}{f_s/2} \cdot \left|1 - z^{-1}\right|^N df = \frac{V_{LSB}}{12} \cdot \frac{1}{OSR} \cdot \left(\frac{\pi}{OSR}\right)^{2N} \frac{1}{2N+1}$$

$$SQNR = 10\log\left(\frac{V_{FS}^2/8}{P_E}\right)$$

$$SQNR \approx \underbrace{(6.02 \cdot B + 1.76)}_{\text{Internal quantization: B-bit}} + \underbrace{10\log(OSR)}_{\text{Plain oversampling}} + \underbrace{N \cdot 20\log\left(\frac{OSR}{\pi}\right)}_{\text{Noise-shaping } N^{th}\text{-order}}$$

Internal quantization: B-bit

Plain oversampling

Noise-shaping N^{th}-order

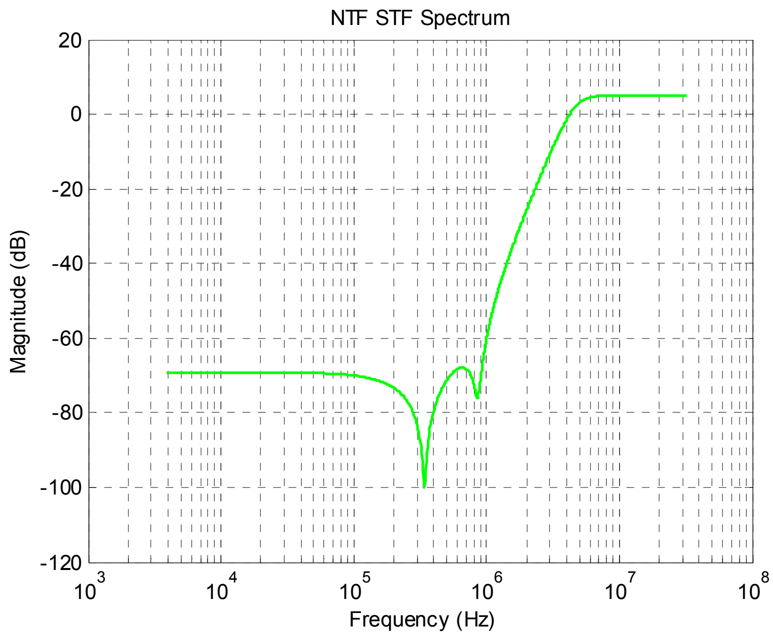
Example: 2-bit, OSR=32, 2^{nd}-order peak SQNR = 6*2 + 3*5 + 2*20 = 67 dB

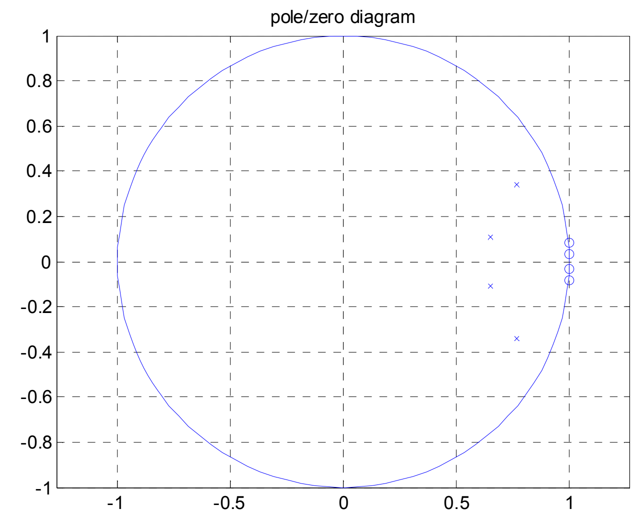
1.1 NTF Synthesis

NTF627 = synthesizeNTF (order, osr, 1, Hinf);

Zero/pole/gain:

$$\frac{(z^2 - 1.999z + 1)\,(z^2 - 1.993z + 1)}{(z^2 - 1.303z + 0.4368)\,(z^2 - 1.534z + 0.7037)}$$





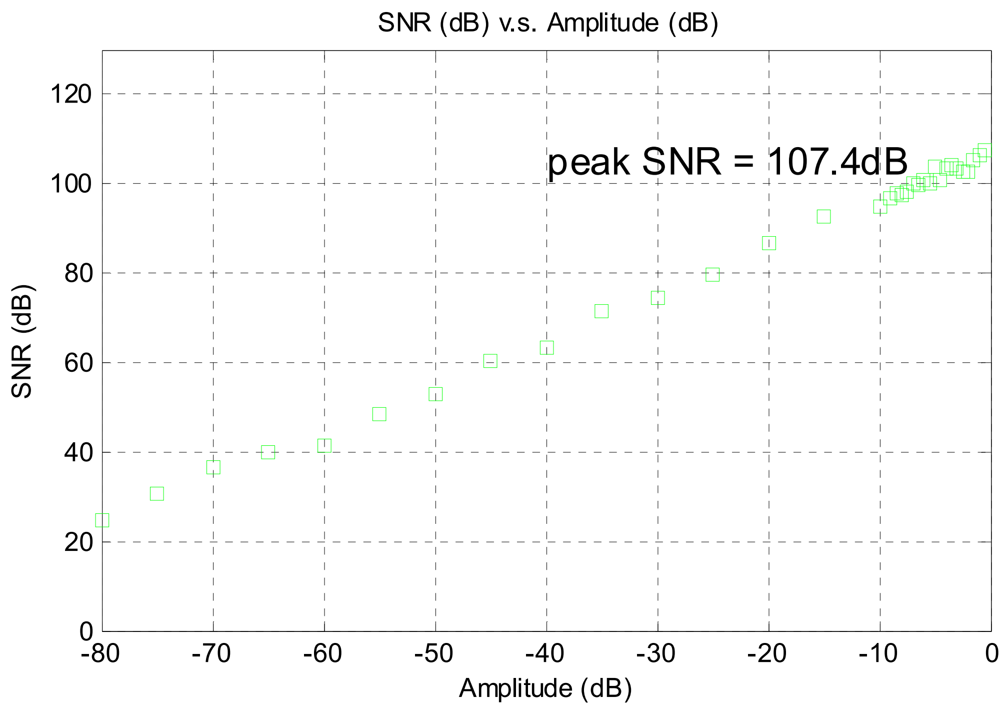
plotPZ(NTF627)

[num,den] = tfdata(NTF627, ‘v’);
[mag_ntf, wT] = freqz (num_ntf, den_ntf, 8192);
semilogx(wT/(2*pi*fs), 20*log10(abs(mag_ntf)));

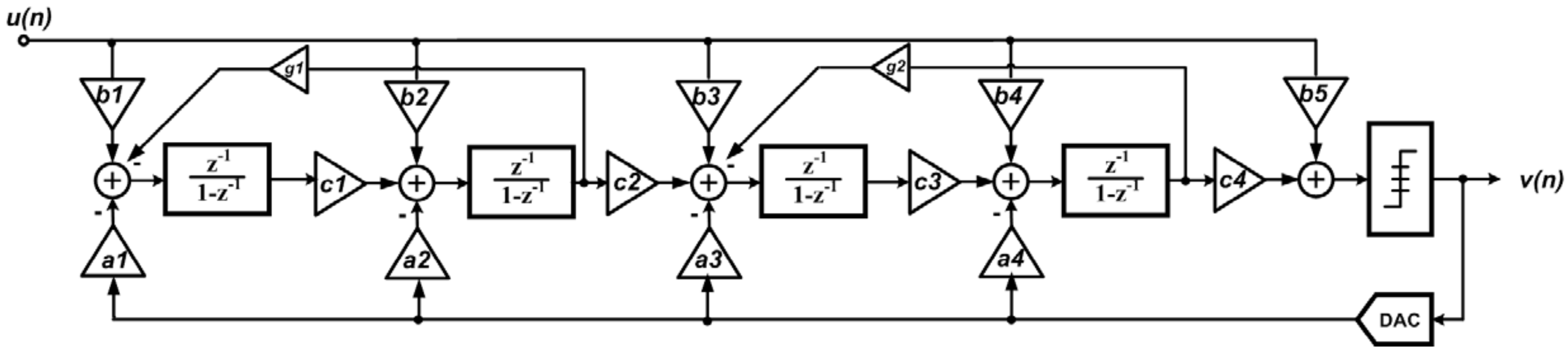
1.2 Verify SQNR by toolbox

```
u = vp*sin(2*pi*fsig/fs*[0:16383]);
v = simulateDSM (order, osr, 1, Hinf);

Amp = [-80:5:10  -9:0.5:0];
v = simulateSNR (ntf, osr, amp, f0, nlev);
```



2. Realize NTF into coefficient



$$NTF627 = \frac{(z^2 - 1.999z + 1)\ (z^2 - 1.993z + 1)}{(z^2 - 1.303z + 0.4368)\ (z^2 - 1.534z + 0.7037)}$$

⇩ [a, g, b, c] = realizeNTF(NTF627, CIFB);

[a,g,b,c]

⇩ ABCD = stuffABCD (a, g, b, c, CIFB);

ABCD Maxtix

⇩ [ABCDs, umax] = scaleABCD (ABCD, nlev, 0, 1, 7);

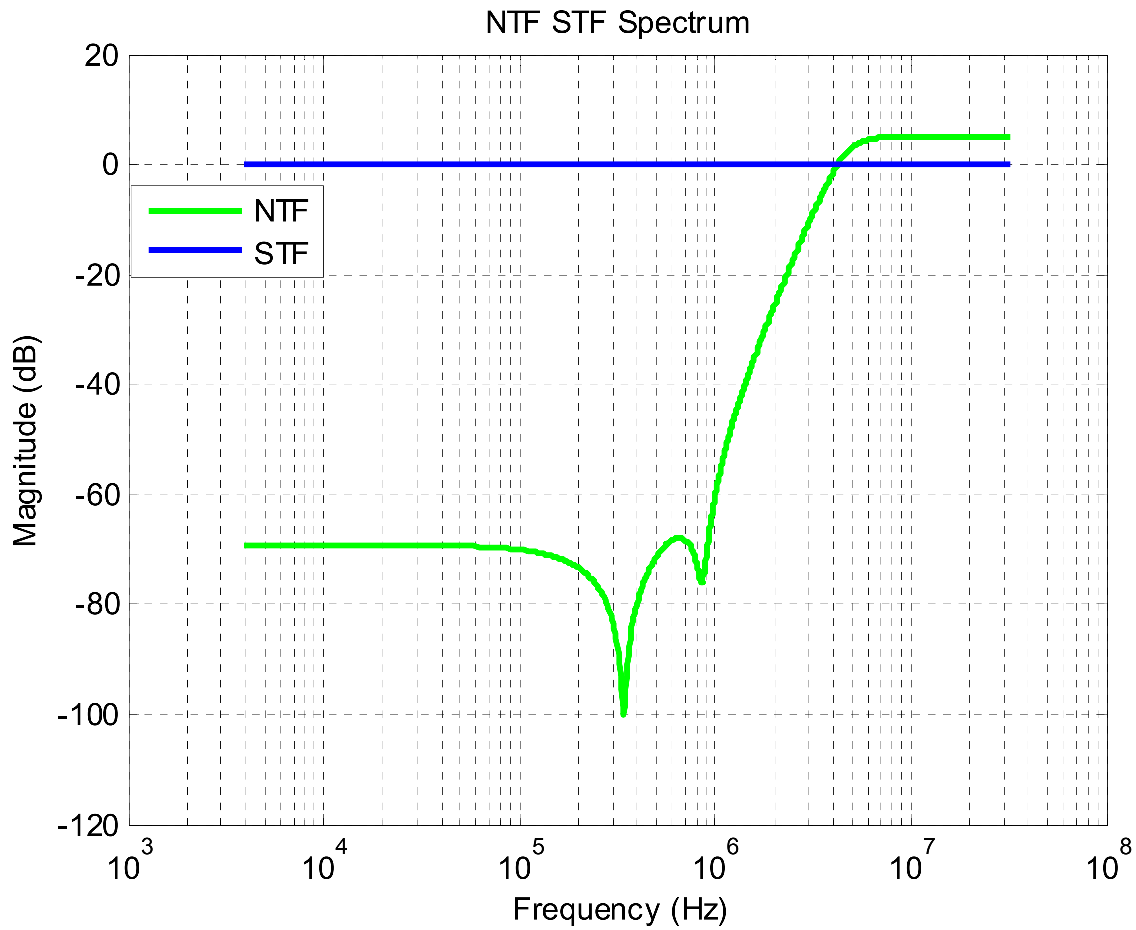
Scale the internal nodes

⇩ [a, g, b, c] = mapABCD (ABCDs, CIFB);

Adjust the coefficients a, g, b, c

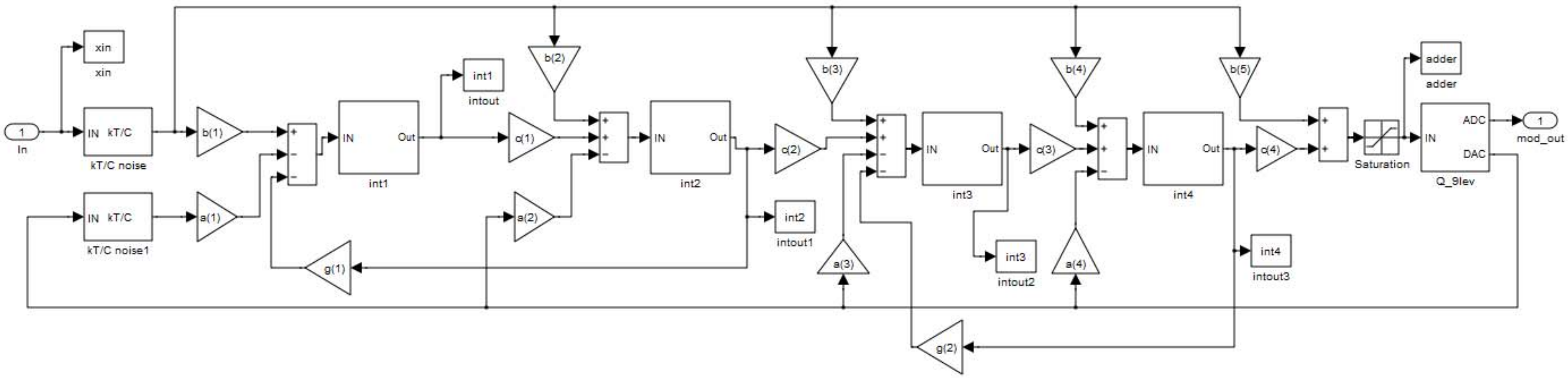
Verify your final NTF & STF

[ntf, stf] = calculateTF (ABCDs, 1);

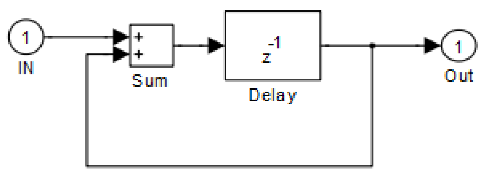


- Generate NTF from your ABCDs.
- Plot PSD (commands on page 2).

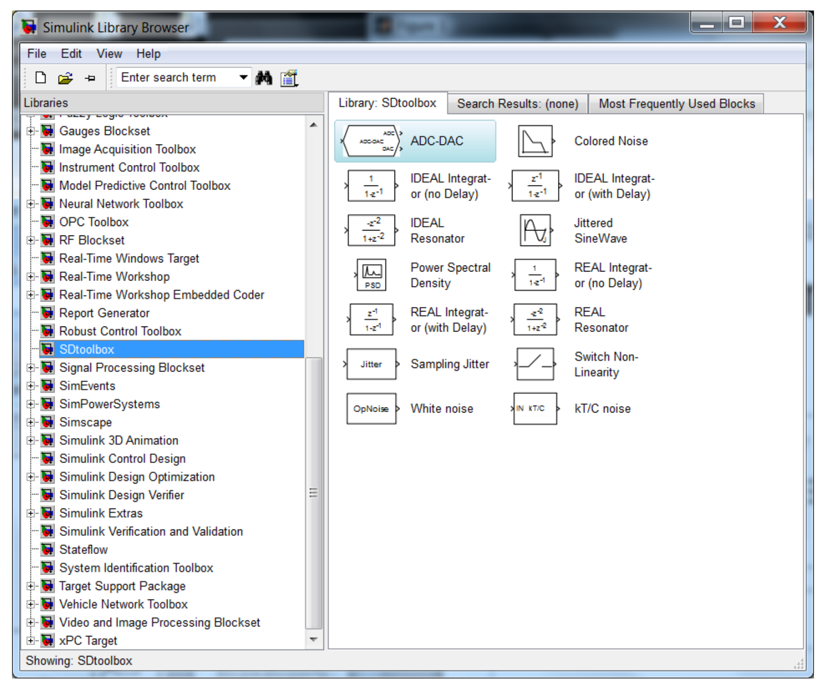
3. Simulink Models



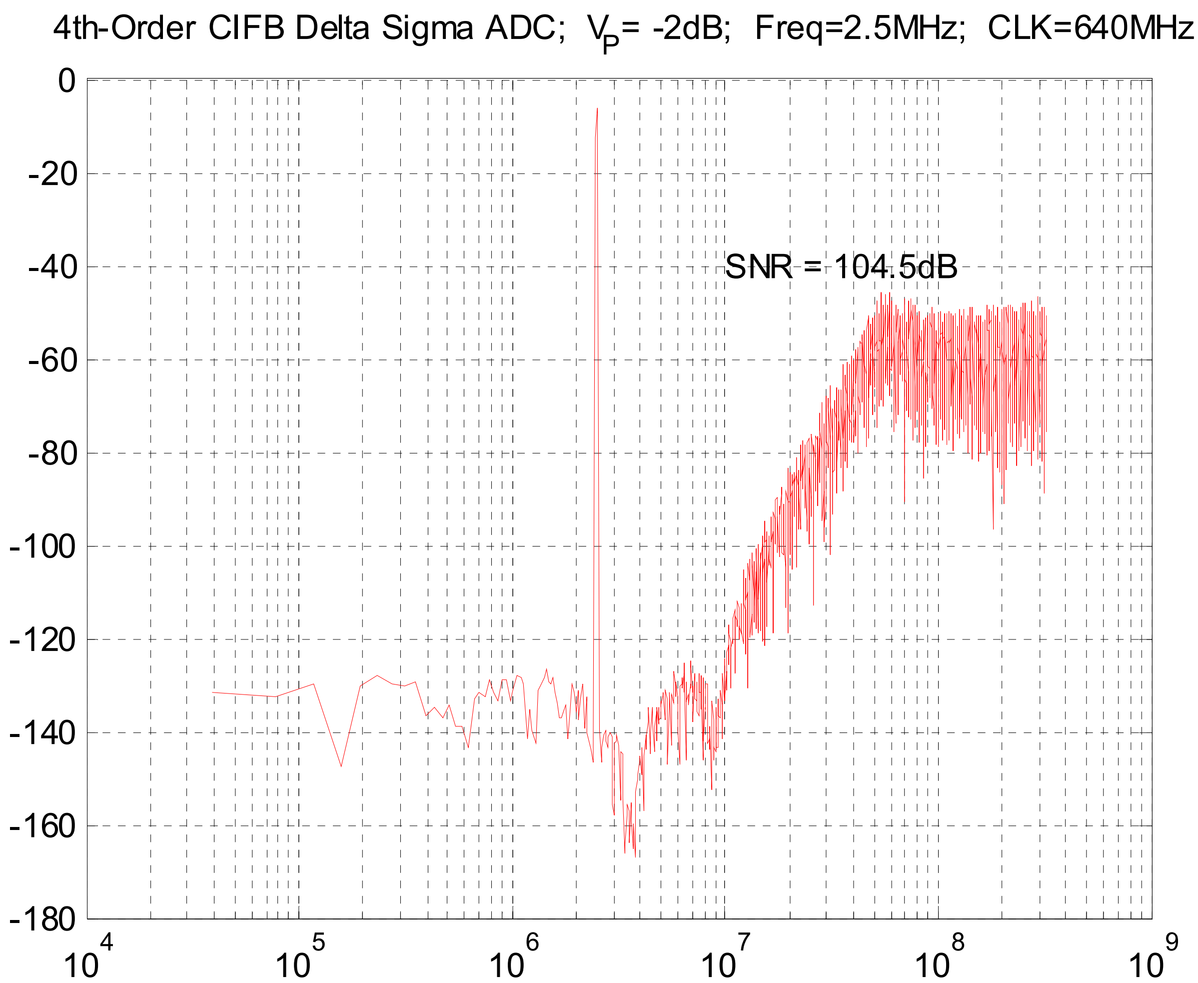
delay integrator



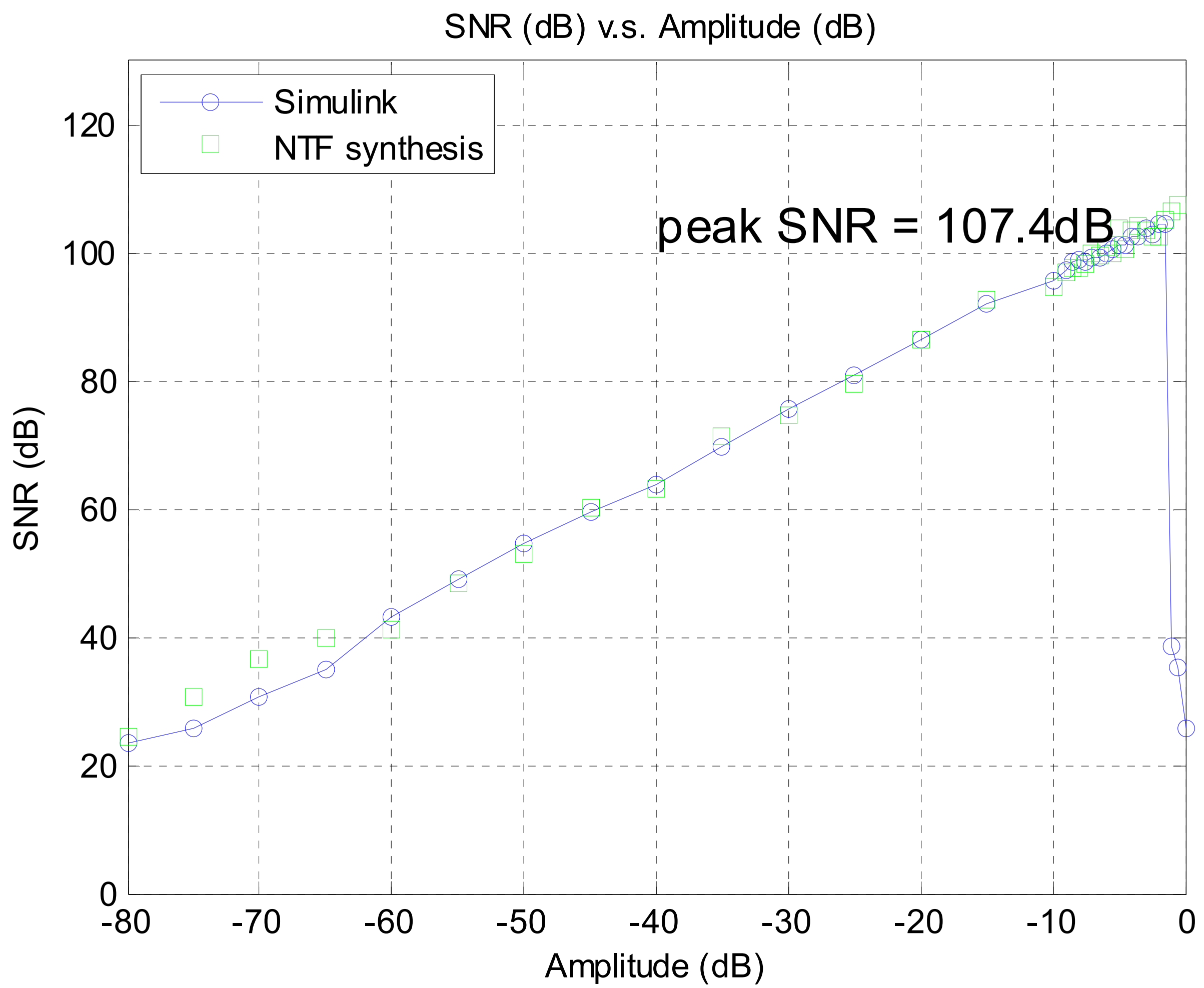
Built-in SDtool in Simulink Library Browser



Peak SQNR @ -2 dBFS



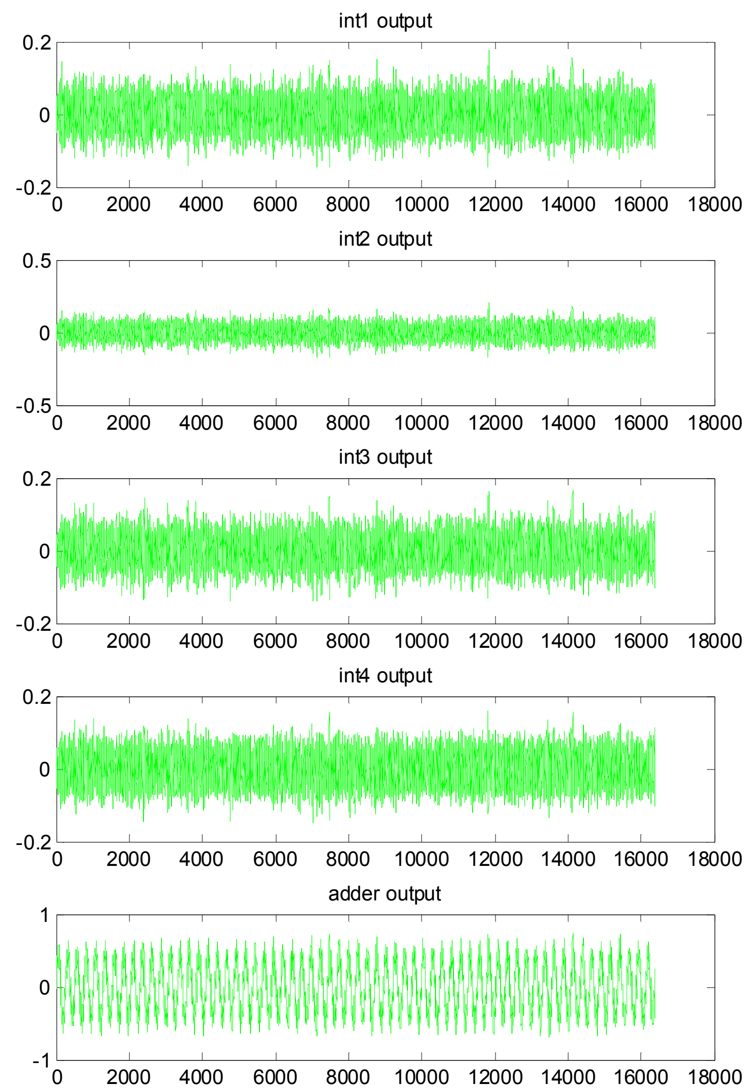
Simulink simulation vs. synthesized NTF



Simulink (p.11) vs. synthesized NTF (p. 8)

Simple Debug Techniques

- Swing at every integrator node should be bounded within VREF.
- Use a smaller input amplitude if not stable.
- Sweep the amplitude. Plot SQNR vs. amplitude.
- FFT points and window (ds_hann).
- Make everything right at Matlab before you start to build the circuits at Cadence.



Summary

1. Determine: order, OSR, quantizer level, modulator type.

2. Synthesize NTF.

3. Realize the coefficient [a, b, c, g] of the modulator.

4. Map the coefficient to internal states ABCD and scale the ABCD.

5. Realize again the coefficient [a, b, c, g] by ABCD. Round-off the [a, b, c ,g] manually by yourself.

6. Simulink simulation. Check the SQNR vs amplitude and integrator swing.

7. Circuit simulation at Candence.