Overview

• Building behavioral models in stages
• A 5th-order, 1-Bit ΣΔ modulator
  - Noise shaping
  - Complex loop filters
  - Stability
  - Voltage scaling

Building Models in Stages

• When modeling a complex system like a 5th-order ΣΔ modulator, model development proceeds in stages
  - Each stage builds on its predecessor

• The design goal is to detect and eliminate problems at the highest possible level of abstraction
  - Each successive stage consumes progressively more engineering time
Building Models in Stages

- Rework and reverification of early stage models because of problems found in later stages is expensive
  - Defective silicon is much more expensive (and often fatal)

- Don’t launch a multistage rework cycle every time you find a single bug

Building Models in Stages

- Our $\Sigma\Delta$ model development proceeds in stages:
  - Stage 0 gets to the starting line
  - Stage 1 develops a practical system built with ideal subcircuits
  - Stage 2 models key subcircuit nonidealities and translates the results into real-world subcircuit performance specifications
Building Models in Stages

- Real-world model development includes a critical stage 3:
  - Adding elements to earlier stages (hopefully only stage 2) to model significant surprises found in silicon

- The previous lecture introduced much of the stage 0 \( \Sigma\Delta \) model and 1-Bit quantization background
  - What other steps are needed to arrive at a successful design?

Stage 0

- Collect references
  - Important references
  - Readable references
  - Talk to veterans to find them and sort them

- Understand the readable references
  - Build a simple model of what you think you understand
  - Start building diagnostic infrastructure
Stage 0 Models

- You can’t just talk about stage 0 models with veterans and look at their stage 0 simulations
  - You’ve got to exercise and think with the model until you can begin to explain surprises by yourself
  - Then, in stage 1, you can ask a veteran more intelligent questions

- Stage 0 model code (download code used for last lecture) is 20% modulator loop code, 80% diagnostics
  - This ratio holds for all stages of modeling

Stage 1

- In stage 1, we’ll study a model for a practical $\Sigma\Delta$ modulator topology built with ideal blocks

- Stage 1 model focus
  - Signal amplitudes
  - Stability
    - Worst-case inputs
    - Unstable systems can’t graduate to stage 2
  - Quantization noise shaping
Stage 1 Models

Building the infrastructure to generate worst-case inputs and analyze model responses is of critical importance in stage 1

- You must tap into your organization’s technical wisdom to learn what those worst-case real world inputs are

**Models can only tell you the right answers if you ask them the right questions!**

ΔΣ Modulator Filter Design

- Procedure
  - Establish requirements
  - Design noise-transfer function, NTF
  - Determine loop-filter, H
  - Synthesize filter
  - Evaluate performance, stability

Modulator Specification

• Example: Audio ADC
  - Dynamic range DR 16 Bits
  - Signal bandwidth B 20 kHz
  - Nyquist frequency $f_N$ 44.1 kHz
  - Modulator order $L$ 5
  - Oversampling ratio $M = f_s/f_N$ 64
  - Sampling frequency $f_s$ 2.822 MHz

• The oversampling ratio M chosen based on
  - SQNR > 120dB (20dB below thermal noise)
  - Experience (e.g. Figure 4.14 in Adams & Schreier)

Modulator Block Diagram

Gain Block

Loop filter
Noise Transfer Function, NTF(z)

% stop-band attenuation ...  
% reduce if design is not stable

Rstop = 80;  
[b,a] = cheby2(L, Rstop, 1/M, 'high');

% normalize (for causality)
[b] = b/b(1);  
NTF = filt(b, a, 1/fs);

% check stability (mag < 1.5)
[mag] = bode(NTF, pi*fs)

>> mag = 1.32

Loop-Filter, H(z)

H = inv(NTF) - filt(1, 1, 1/fs);

% check causality ... y(1) should be 0
y = impulse(H);

>> y = 0
Filter Topology

Rounded Filter Coefficients

\[ \begin{align*}
    a_1 &= 1; & k_1 &= 1; & b_1 &= 1/1024; \\
    a_2 &= 1/2; & k_2 &= 1; & b_2 &= 1/16-1/64; \\
    a_3 &= 1/4; & k_3 &= 1/2; \\
    a_4 &= 1/8; & k_4 &= 1/4; \\
    a_5 &= 1/8; & k_5 &= 1/8;
\end{align*} \]

5\textsuperscript{th}-Order Noise Shaping

\begin{align*}
|A_m| (\text{dBWN}) & \text{ or Integrated Noise (dBV)} \\
0 & -40 & -80 & -120 & -160
\end{align*}

\begin{align*}
\text{Frequency [kHz]} & \\
0 & 300 & 600 & 900 & 1200 & 1500
\end{align*}

That's noise shaping! – let's look closer...

100mVrms, 30kHz input
30000 point DFT
30 averages

3 Q\text{noise} zeroes

only 82nVrms from DC to 20kHz
5th-Order Noise Shaping

- The 1Vrms 1-Bit quantization noise is shaped to sum to only 82nVrms in the audio band
  - That's over 140dB of dynamic range
- ΣΔ modulators are usually designed so that their quantization noise is negligible in the frequency band of interest
  - Thermal noise sources dominate
- Let's look at the loop filter transfer function...
5th-Order Loop Filter

- Lots of low frequency gain
- 0dB gain at 378kHz

Frequency [kHz]

upward phase jumps imply poles just outside the unit circle
5th-Order Loop Filter

- The fact that $H(z)$ has poles outside the unit circle doesn’t mean that the entire ΣΔ modulator is unstable
  - The modulator’s stability depends on its closed loop poles

- All loop variables ($\int_1, \int_2, \int_3, \int_4, \int_5$) have the same closed loop poles
  - If one is stable, they all are
Modulator Root-Locus

The nonlinear modulator system operates at some effective gain \( G \) between points A and B:

\[
\begin{align*}
\text{v}_{\text{IN}} & \rightarrow H(z) \rightarrow \text{d}_{\text{OUT}} \\
1 \text{ or } -1
\end{align*}
\]

- \( G \) may be a function of both \( v_{\text{IN}} \) and \( g \)
- The modulator closed loop poles are the zeroes of the function \( 1+HG \):
  \[
  \frac{\text{D}_{\text{OUT}}(z)}{\text{V}_{\text{IN}}(z)} = \frac{\text{HG}/g}{1 + \text{HG}}
  \]
- We'll plot closed loop poles in the z-plane as \( G \) varies from 0.1 to 10 in equal log steps ...
Modulator Root-Locus

unit circle

start (G=0.1) unstable
Modulator Root-Locus

Closed-loop poles move inside the unit circle for $G > 0.4$

stop (G=10) stable
Effective Gain

• If our linearized model is valid (a big if)
  - For $G > 0.4$, the modulator system is stable
  - For $G < 0.4$, it's unstable

• Presumably, the noise shapes in slides 18 and 23 were produced by a stable system
  - We'll evaluate $G$ for 5kHz and 20kHz sinusoidal inputs varying in amplitude from $-30$dBV to $+5$dBV…
  - While we're at it, we'll capture minimum and maximum signal levels throughout the modulator

**Sinewave Input Effective Gain**

Both 5kHz and 20kHz look "DC like" to a 3MHz modulator
Effective Gain

- As the input amplitude increases, the signal at the quantizer input grows, and G falls
  - Just over 1Vrms, G falls to below 0.4, and
    - The system becomes unstable
    - Loop variables grow without bound (opamps in a real analog circuit will just run up to power supply rails)
    - Noise shaping is lost

- It's highly unlikely that audio sinewaves provide the worst case inputs for stability
  - To evaluate any model, you've got to know what the worst case inputs are

- Let's look at inputs that aren't “dc-like” and aren't sinusoidal (square waves)...
Modulator Stability

- The sensitivity of ΣΔ modulators to high frequency square wave inputs was first discovered on breadboards
  - No one thought to provide such inputs to early modulator simulations

- Worst-case square wave frequencies are roughly equal to the frequency of the highest Q pole in the noise shape
  - A key job of the antialiasing filters used in front of ΣΔ modulators is to reduce out-of-band signals to safe levels
Modulator Stability

• 5000 point simulations such as those in the previous slides don’t guarantee stability
  – Sometimes millions of time points are required before an unstable modulator blows up
  – When it explodes, G falls very quickly

• Square wave tolerance is a fast, effective basis for comparing the relative stability of different modulator topologies

Voltage Scaling

• Given that the modulator is stable for 1Vrms inputs, let’s move on to look at the state variable voltages under various input conditions
  – Loop state variables and the filter output are labeled green on the next slide

• Peak signal levels and signal standard deviations are easy to obtain in MATLAB
  – We’ll examine voltages for a 5kHz sinusoidal input...
5th-Order Loop Filter

IN (from summer)

\[ k_1 (1 - z^{-1}) \]

\[ b_1 \]

\[ k_2 z^{-1} (1 - z^{-1}) \]

\[ 1 - z^{-1} \]

\[ a_1 \]

\[ k_3 z^{-1} (1 - z^{-1}) \]

\[ 1 - z^{-1} \]

\[ a_2 \]

\[ k_4 z^{-1} (1 - z^{-1}) \]

\[ 1 - z^{-1} \]

\[ a_3 \]

\[ k_5 z^{-1} (1 - z^{-1}) \]

\[ 1 - z^{-1} \]

\[ a_4 \]

\[ a_5 \]

\[ Q \]

OUT (to comparator)

5kHz Input Loop Voltages

<table>
<thead>
<tr>
<th>Input Amplitude [dBV]</th>
<th>Positive Peaks</th>
<th>Negative Peaks</th>
</tr>
</thead>
<tbody>
<tr>
<td>-30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

EECS 247 Lecture 20: 5th Order Architecture © 2002 B. Boser
Only the sign of $Q$ matters, so we can make $k_1$ whatever we want without changing the 1-Bit data at all.

5kHz Input Loop Voltages

- If we scale $k_1$ by 0.1,
  - All state variables and $Q$ scale by 0.1
  - But since the comparator output is fixed, $G$ increases 10X

- The change in $k_1$ doesn’t change the shape of the root locus, either
  - The effective gain for each root position is increased 10X
  - $G > 4$ is now required for stability
5kHz, $k_1=0.1$ Effective Gain

![Graph showing effective gain vs. sinewave input amplitude (dBV).]

$k_1=0.1$ Loop Voltages

![Graph showing loop peak voltages (V) vs. input amplitude (dBV).]
Loop Voltage Scaling

• Before we scale $k_1$ down any lower, we note that $\int_3$, $\int_4$, and $\int_5$ have substantially larger swings than $\int_1$ and $\int_2$

• Just about any filter topology allows scaling tricks which change internal state variable amplitudes without changing the filter output
  - The next slide shows an example
Input Range Scaling

- Slides 40 and 44 indicate inadequate stability margins for 1Vrms sinewave inputs

- Scaling the DAC output levels adjusts the modulator input range
  - If $V_{in}$ and the DAC outputs are scaled up by the same factor $g$, the 1-Bit data is completely unchanged
  - Of course, increasing the range also increases the quantization noise ... the dynamic range and peak SQNR stay the same!
  - If the DAC output levels are increased and the analog full scale is held constant, the stability margin improves ... at the expense of reduced SQNR

Increasing the DAC levels by $g$ reduces the analog to digital conversion gain:

$$\frac{D_{OUT}(z)}{V_{IN}(z)} = \frac{H}{1+gH} \approx \frac{1}{g}$$

Increasing $V_{IN}$ $g$ by the same factor leaves 1-Bit data unchanged
Stage 1 Modulator

• We’ll increase $g$ from 2.5 to 3.0 to provide a 2dB increase in stability margin for a 1Vrms full scale input

• We’ll also implement the loop voltage scaling changes suggested in slide 45

• The result is our first-pass stage 1 modulator, and its performance appears on the following slides …
Loop Voltages

VLSI-compatible voltages!

Input Amplitude [dBV]

Loop Peak Voltages [V]

1 2 3 4 5

-2 -4 -30 -25 -20 -15 -10 -5 0 +5

C

loop voltages

VLGH-compatible voltages!