EE247
Lecture 14

• Administrative issues:
  – Midterm date changed to Thurs. Oct. 21
  – Final exam moved to Wednesday, December 15, 12:30-3:30pm
  – Both date changes due to conflict with EE142
  – No homework next week

EE247
Lecture 14

• Data Converters
  • Practical aspects of converter testing
    • Signal source
    • Clock generator
    • Evaluation board considerations
  • D/A converter design
Converter Testing
Practical Aspects

Just Got Silicon Back...

• Now what?
• Practical aspects of converter testing
• Equipment requirements
• Pitfalls
Direct ADC-DAC Test

Device Under Test (DUT)

- Need a very good DAC
- Actually a good way to "get started"

- Beware of spectrum analyzer nonlinearities
- For high performance converter linearity test, may need to notch out the signal to measure the ADC linearity via spectrum analyzer
- Need to build or purchase notch filter/s
Filtering

Amplitude DAC Output

Notch (Band Reject) Filter

Amplitude Input to Spectrum Analyzer

0 \( f_{in} \) \( 2f_{in} \) \( 3f_{in} \) \( 4f_{in} \) \( \cdots \) \( f \)

Input to Spectrum Analyzer

0 \( f_{in} \) \( 2f_{in} \) \( 3f_{in} \) \( 4f_{in} \) \( \cdots \) \( f \)

ADC Test Setup

Specs?

Evaluation Board?

How to get data across?

Signal Generator

Clock Generator

ADC

Data Acquisition

PC

Specs?
Example: High Performance ADC

- Testing a high performance converter may be just as challenging as designing it!
- Key to success is to be aware of test setup and equipment limitations

<table>
<thead>
<tr>
<th>Resolution</th>
<th>14 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conversion Rate</td>
<td>75 MSPS</td>
</tr>
<tr>
<td>Input Range</td>
<td>2 Vpp differential</td>
</tr>
<tr>
<td>SNR @ Nyquist</td>
<td>73 dB</td>
</tr>
<tr>
<td>SFDR @ Nyquist</td>
<td>88 dB</td>
</tr>
<tr>
<td>DNL</td>
<td>0.6 LSB</td>
</tr>
<tr>
<td>INL</td>
<td>2.0 LSB</td>
</tr>
</tbody>
</table>


Example: ADC Linearity Test

[Graph showing linearity test results]

Signal Source

- Need: SFDR > 95dB @ \( f_{in} = f_s / 2 = 37.5 \text{MHz} \) & SFDR > 70dB to about 120MHz
- Let’s see, how about the "value priced" signal generator available in most labs...
  - \( f = 0 \ldots 15 \text{MHz} \)
  - Harmonic distortion (f > 1MHz): -35dBc
    → Does not cover the required frequency range & poor linearity

A Better Signal Source

- OK, now we’ve spent about $40k, this should work now... (?)
  - \( f = 100 \text{kHz} \ldots 3 \text{GHz} \)
  - Harmonic distortion (f > 1MHz): -30dBc!
  - No way to produce the sine wave we need without a filter!
Filtering Out Harmonics

- Given HD=-30dBc, we need a stopband rejection > 65dB to get SFDR>95dB

Available Filters

Elliptical Function Bandpass Filters 1kHz to 20MHz

<table>
<thead>
<tr>
<th>Series Number</th>
<th>BWR</th>
<th>Stopband Attenuation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q34</td>
<td>4.0:1</td>
<td>-60dBc</td>
</tr>
<tr>
<td>Q36</td>
<td>4.0:1</td>
<td>-60dBc</td>
</tr>
<tr>
<td>Q36</td>
<td>10.0:1</td>
<td>-60dBc</td>
</tr>
<tr>
<td>Q54</td>
<td>2.5:1</td>
<td>-60dBc</td>
</tr>
<tr>
<td>Q70</td>
<td>3.5:1</td>
<td>-60dBc</td>
</tr>
<tr>
<td>Q56</td>
<td>3.5:1</td>
<td>-60dBc</td>
</tr>
</tbody>
</table>

- Want to test at many frequencies → Need to have at least one filter per test frequency!
Tunable Filter

www.klmicrowave.com

Filter Distortion

- Beware: The filters themselves also introduce distortion
- Distortion is usually not specified, need to call manufacturer
- Often guaranteed: HD<-85dBc,
- Don't trust your filters blindly...
Clock Generator

- Let us check if for the clock a "value-priced" signal generator will suffice...
- No! The clock signal controls sampling instants – which we assumed to be precisely equi-distant in time (period T)
- Variability in T causes errors
  - "Aperture Uncertainty" or "Aperture Jitter"
- How much Jitter can we tolerate?

Clock Jitter

- Sampling jitter adds an error voltage proportional to the product of \((t_J - t_0)\) and the derivative of the input signal at the sampling instant
- Jitter doesn’t matter when sampling dc signals \((x'(t_0) = 0)\)
Clock Jitter

- The error voltage is
  \[ e = x'(t_0)(t_j - t_0) \]

Jitter Example

Sinusoidal input

- Amplitude: \( A \)
- Frequency: \( f_s \)
- Jitter: \( \frac{A_{FS}}{2} \)

\[ x(t) = A \sin \left( 2\pi f_s t \right) \]
\[ x'(t) = 2\pi f_s A \cos \left( 2\pi f_s t \right) \]
\[ \left| x'(t) \right|_{\text{max}} \leq 2\pi f_s A \]
\[ \left| e(t) \right| \leq \left| x'(t) \right| dt \]
\[ \left| e(t) \right| \leq 2\pi f_s A \left| dt \right| \]

Worst case

- \( A = \frac{A_{FS}}{2} \)
- \( f_s = \frac{f_s}{2} \)

\[ e(t) \leq \pi f_s \frac{A_{FS}}{2} \left| dt \right| \]
\[ \left| e(t) \right| \leq \frac{A_{FS}}{2 \pi f_s} \]

\[ dt \ll \frac{1}{2 \pi f_s} \]

<table>
<thead>
<tr>
<th># of Bits</th>
<th>( f_s )</th>
<th>( dt )</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>10 MHz</td>
<td>0.5 ps</td>
</tr>
<tr>
<td>12</td>
<td>100 MHz</td>
<td>0.8 ps</td>
</tr>
<tr>
<td>8</td>
<td>1000 MHz</td>
<td>1.2 ps</td>
</tr>
</tbody>
</table>
Law of Jitter

- The worst case looks pretty stringent … what about the “average”?
- Let’s calculate the mean squared jitter error (variance)
- If we’re sampling a sinusoidal signal
  \[ x(t) = A \sin(2\pi f_x t), \]
  then
  - \[ x'(t) = 2\pi f_x A \cos(2\pi f_x t) \]
  - \[ E\{[x'(t)]^2\} = 2\pi^2 f_x^2 A^2 \]
- Assume the jitter has variance \[ E\{(t_j - t_0)^2\} = \tau^2 \]

Law of Jitter

- If \( x'(t) \) and the jitter are independent
  - \[ E\{[x'(t)(t_j - t_0)]^2\} = E\{[x'(t)]^2\} \ E\{(t_j - t_0)^2\} \]
- Hence, the jitter error power is
  \[ E\{e^2\} = 2\pi^2 f_x^2 A^2 \tau^2 \]
  (compare to the worst case \( 4\pi^2 f_x^2 A^2 \tau^2 \))
- If the jitter is uncorrelated from sample to sample, this “jitter noise” is white
Law of Jitter

\[ DR_{jitter} = \frac{A^2}{2f_s^2 A^2 \tau^2} = \frac{1}{2f_s^2 \tau^2} = -20 \log_{10}(2\pi f_s \tau) \]

ADC under test:
SNR=73dB
\( f_{in} = 37.5\text{MHz} \)
\( \Rightarrow \tau \ll 1\text{ps rms}! \)

More on Jitter

- Once we have a good enough generator, other circuit and test setup related issues may determine jitter, but...
- Usually, clock jitter in the single-digit pico-second range can be prevented by appropriate design techniques
  - Separate supplies
  - Separate analog and digital clocks
  - Short inverter chains between clock source and destination
- Few, if any, other analog-to-digital conversion non-idealities have the same symptoms as sampling jitter:
  - RMS noise proportional to input frequency
  - RMS noise proportional to input amplitude
\( \Rightarrow \) In cases where clock jitter limits the dynamic range, it’s easy to tell, but may be difficult to fix...
Evaluation Board

- Planning begins with converter pin-out
  - Example of poor pin-out: clock pin right next to a digital output...
- Not "Black Magic", but weeks of design time and learning
- Key aspects
  - Supply/ground routing, bypass capacitors
  - Coupling between signals
- Good idea to look at ADC vendor datasheets for example layouts/schematics/application notes

Vendor Eval Board Layout

[Analog Devices AD9235 Data Sheet]
Issues

- A converter does not just have one "input"
  - Clock
  - Power Supply, Ground
  - Reference Voltage
- For good practices on how to avoid issues see e.g.:
  - Analog Devices Application Note 345: "Grounding for Low-and-High-Frequency Circuits"
  - Maxim Application Note 729: "Dynamic Testing of High-Speed ADCs, Part 2"

How to Get the Bits Off Chip?

- "Full swing" CMOS signaling works well for $f_{\text{CLK}} < 100\text{MHz}$
- But we want to build faster ADCs...
- Alternative to CMOS: LVDS – Low Voltage Differential Signaling
- LVDS vs. CMOS:
  - Higher speed, more power efficient at high speed
  - Two pins/bit!
LVDS Outputs

**Figure 1. LVDS Output Levels**

Analog Devices Application Note 586: "LVDS Data Outputs for High Speed ADCs"

---

LVDS Outputs

**Figure 4. LVDS Output Current**

Analog Devices Application Note 586: "LVDS Data Outputs for High Speed ADCs"
Data Acquisition

- Several options:
  - Logic analyzer with PC interface
  - FIFO board, interface to PC DAQ card
  - Vendor kit, simple interface to printer port:

![Diagram of data acquisition setup]

Post-Processing

- LabView (DAQ Software Toolbox)
- Matlab
- Some vendors provide example source code
- E.g. Maxim Application Note 1819: "Selecting the Optimum Test Tones and Test Equipment for Successful High-Speed ADC Sine Wave Testing"
- EE247 Matlab Examples, e.g. Sine Code Density Test
Complete Setup

[Maxim Application Note 729: *Dynamic Testing of High-Speed ADCs, Part 2]

Debugging

- State-of-the-art converters almost always yield surprises in silicon
  - If models anticipate everything, the application probably isn’t state-of-the-art
- Analog designers and mixed-signal architects often invent new circuits while measuring in the lab
- How do we debug converters?
  - Start with a simple time domain test. Does the captured digital waveform look like a sine wave?
  - Begin your DFT/INL signature analysis by scaling down sampling frequencies and signal input frequencies together
  - If you can’t explain performance with essentially infinite settling times, don’t add dynamic errors to the mix
Debugging

• Typical problems come from non-idealities never built into your "model"
  – E.g. half-circuit models for fully-differential circuits inherently can’t explain some types of differential-symmetry errors
• You can’t afford to rediscover old non-idealities in new silicon
  – Talking to veterans early in the modeling phase can be important

Debugging

• Design teams usually track down and fix single-cause problems quickly
• Interactions are much trickier
• Interaction examples:
  – Digital activity-dependent clock jitter
    • S/(N+D) degradation only happens when large amplitude, high frequency analog inputs coincide with the offending digital activity
  – Distortion cancellation
    • Nonlinear phenomena don’t obey superposition
Debugging

- Cancellation of Nonlinearities?

![SFDR vs. Input Frequency (f_{\text{clk}} = 75\text{MHz})](chart)

- Never assume all of your data is good
  - One bad data set can "rule out" the correct explanation, leading you astray forever
- "Compare measurements to themselves"
- But, noise is a random variable, and the noise power in 1000 time samples will vary from DFT to DFT
- How big an effect is this?
Debugging

• Can show that:
  – Variation of noise in 1000 samples yields a standard deviation in SNR of 0.2dB
  – This means that 68.3% of all DFTs will produce SNRs within 0.2dB of the average
  – 99.7% of 1000 point DFTs yield SNRs within ±0.6dB of the average
• If you’re seeing ADC noise variation of greater than ±0.6dB in the lab, some sort of interference is usually the culprit

Debugging

• Always try to use two independent measurement methods to verify important results
  – Correlate INL & SFDR, DNL & SNR
• Comparing time domain and frequency domain views of the same measurement is good practice
  – DC histogram test & thermal noise - DNL & SNR
D/A Converters

- D/A architecture examples
  - Unit element
  - Binary weighted
- Static performance
  - Component matching
  - Architectures
    - Unit element
    - Binary weighted
    - Segmented
    - Dynamic element matching
- Dynamic performance
  - Glitches
- DAC Examples

D/A Examples

- Voltage, Charge, or Current Based
- E.g.
  - Resistor string
  - Charge redistribution
  - Current source type
Example:
Input code 101

\[ V_{\text{out}} = \frac{5V_{\text{REF}}}{8} \]

\[ \tau_{\text{settling}} = \]

\[ = (3R||5R) \times C \]
\[ -0.23 \times 8RC \]
R-String DAC

- Advantages:
  - Simple, fast for <8-10 bits
  - Inherently monotonic
  - Compatible with purely digital technologies

- Disadvantages:
  - $2^B$ resistors & $2^B$ switches for $B$ bits → High element count & larger area for $B$>10 bits
  - High settling time for $B > 10$:
    $\tau_{\text{max}} = 0.25 \times 2^B RC$


Charge Redistribution DAC

- E.g. “Binary weighted”
- $B$+1 capacitors & switches (if unit elements used $2^B$ caps)
Charge Redistribution DAC

Example: 4Bit DAC - Input Code 1011

- Charge redistribution

\[ V_{out} = \sum_{i=0}^{3} b_i \cdot 2^i \cdot C \]

- Reset phase

- Charge phase

- Monotonicity depends on element matching
- Large area of caps for high DAC resolution (10bit DAC ratio 1:512)
- Sensitive to parasitic capacitor @ output
Charge Redistribution DAC

\[ V_{out} = \sum_{i=0}^{B-1} b_i 2^i C \cdot V_{ref} / C_f \]

- Opamp helps eliminate the parasitic capacitor effect
  - Issue: opamp offset & speed

Current Source DAC

- “Unit elements ”
- Monotonicity does not depend on element matching
- \(2^{B-1}\) current sources & switches
Current Source DAC
Binary Weighted

- “Binary weighted”
- Monotonicity depends on element matching
- B current sources & switches ($2^B - 1$ unit elements)

Static DAC INL / DNL Errors

- Component matching
- Systematic errors
  - Contact resistance
  - Edge effects in capacitor arrays
  - Process gradient
  - Finite current source output resistance
- Random errors
  - Lithography
  - Often Gaussian distribution (central limit theorem)

Gaussian Distribution

\[ p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \]

where standard deviation: \( \sigma = \sqrt{E(X^2) - \mu^2} \)

Yield

\[ P(-X \leq x \leq +X) = \int_{-X}^{X} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \]

\[ = \text{erf} \left( \frac{X}{\sqrt{2}} \right) \]
## Yield

<table>
<thead>
<tr>
<th>$X/\sigma$</th>
<th>$P(-X \leq x \leq X)$ [%]</th>
<th>$X/\sigma$</th>
<th>$P(-X \leq x \leq X)$ [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2000</td>
<td>15.8519</td>
<td>2.2000</td>
<td>97.2193</td>
</tr>
<tr>
<td>0.4000</td>
<td>31.0843</td>
<td>2.4000</td>
<td>98.3605</td>
</tr>
<tr>
<td>0.6000</td>
<td>45.1494</td>
<td>2.6000</td>
<td>99.0678</td>
</tr>
<tr>
<td>0.8000</td>
<td>57.6289</td>
<td>2.8000</td>
<td>99.4890</td>
</tr>
<tr>
<td>1.0000</td>
<td>68.2689</td>
<td>3.0000</td>
<td>99.7300</td>
</tr>
<tr>
<td>1.4000</td>
<td>83.8487</td>
<td>3.4000</td>
<td>99.9326</td>
</tr>
<tr>
<td>1.6000</td>
<td>89.0401</td>
<td>3.6000</td>
<td>99.9682</td>
</tr>
<tr>
<td>1.8000</td>
<td>92.8139</td>
<td>3.8000</td>
<td>99.9855</td>
</tr>
<tr>
<td>2.0000</td>
<td>95.4500</td>
<td>4.0000</td>
<td>99.9937</td>
</tr>
</tbody>
</table>