FUNDAMENTALS OF ANALOG TO DIGITAL CONVERTERS: PART I. 1

Jose Silva-Martinez

Many of these slides were provided by

Dr. Sebastian Hoyos

January 2020
Outline

- Fundamentals of Analog-to-Digital Converters
  - Introduction
  - Sampling and Quantization
  - Quantization noise and distortion
  - INL and DNL
  - Technological related issues
  - Sample and Hold
  - Switching issues
  - S/H Accuracy
  - Active S/H
  - Switch around S/H
What Does an Analog-to-digital Converter Do?

- It converts continuous-time signals to discrete-time binary-coded form. Two purposes are (1) to enable computer analysis of the signal, and, (2) to enable digital transmission of the signal.
  - Some examples of continuous-time signals:
    - speech, medical imaging, sonar, radar, electronic warfare, instrumentation, consumer electronics, telecommunications, ...
  - The conversion can be thought of as a two-step process:
    - sampling the input signal in time, usually at regularly-spaced intervals; \( f_{\text{samp}} = 1/T \), where \( T \) = sampling interval
      - Example, for \( f_{\text{samp}} = 1 \) gigasample per second, \( T = 1 \) ns.
    - quantizing (or digitizing) the samples in amplitude, usually voltage. The full-scale input voltage is divided into \( 2^N \) sub-ranges where \( N \) = the ADC’s resolution (number of output leads).
      - Example, for \( N = 12 \) bits, a 1-Volt full-scale range is divided into \( 2^N = 4096 \) levels. The size of the least-significant bit (LSB) is \( 1 \text{ V} / 2^N = 244 \mu\text{V} \).

R. Walden, 1999
What is an Analog-to-Digital Converter (ADC)?

**Analog**
- Continuous with no apparent discontinuities
- The way we interpret our surroundings: sound, light, temperature … etc

**Digital**
- Discrete with limited range; based on binary numbers with limited number of bits.
- The way we mathematically represent and process our world using electronic “brain” power

ADC
How does an ADC work?

**Analog**
- Continuous with no apparent discontinuities
- The way we interpret our surroundings: sound, light, temperature … etc

**Digital**
- Discrete with limited range; based on binary numbers with limited number of bits.
- The way we mathematically represent and process our world using electronic “brain” power
How does an ADC work?

**Analog**

- Input signal $x(t)$
- Sampling
  - $\delta(t-nT_s)$
  - $x(t-nT_s)$

**ADC**

- Quantization noise
- $x(nT_s) + q(nT_s)$
- $2^N$ Levels separated by 1 LSB, $1$ LSB = $V_{FS} \times 2^{-N}$

**Digital**

- $x(n)$
- Decoding
  - $N$ bits
  - 100101001
  - 010010100
  - 1001100010
  - 11001010

* $V_{FS} = \text{full scale range, } V_{max} - V_{min}$
ADCs: Yesterday vs. Today

Example: Digital photography (8-12b ADCs)

Yesterday: 2000

- 0.5-0.8µm CMOS with 5V supply (moderate gate density and speed in DSPs)
- 2M pixel CCD sensor (low pixel scanning speed)
- Some pre-ADC analog conditioning
- ~ 2.5mV / LSB

Today: 2009

- 90nm-180nm CMOS with 1.2-1.8V supplies (high gate density and speed in DSPs)
- 12M pixel CCD sensor (high pixel scanning speed)
- Minimal pre-ADC analog conditioning
- ~ 0.5mV / LSB

- Faster DSPs capable of performing numerous complex functions are developed thanks to advanced CMOS technologies.
- ADCs are indispensable, but now need to handle smaller signals at higher speeds with similar or higher resolutions.
- ADCs are becoming the bottleneck for advancement, and new design techniques need to be developed.
ADC IEEE literature survey: 2006-2008

- Pipeline ADC is currently the most published architecture.
- Pipeline ADC is breaking the trend set by Sigma-Delta and Flash ADCs, and driven by consumer electronics.
- Pipeline ADC is expected to be a key ADC architecture in future applications.

The development of new design techniques for high speed, low voltage and low power ADCs is crucial to stay on the future applications roadmap.
Multi-standard Wireless Systems

– Multiple services

– Reuse circuits as much as possible
  • Power
  • Area
  • Competitiveness

– Smaller Cell phone,
  stronger function,
  longer battery duration

– Use of digital (analog unfriendly)
  nanometric tecnologies
Super-heterodyne Receiver

- Invented by Armstrong in 1918
- Hardware specific radio architecture
- Extensive filtering to relax ADC specs
- Suitable for narrow-band applications
Design issues for multi-standard solutions

- Excessive power at the front-end (Linearity issues)
- Extensive down conversions: LO and mixers increase both noise and power consumption
- Extensive filtering: Area, Power and Noise issues
- Not fully compatible for the Telecoms roadmap

Limited by flicker noise
Not flexible
Hardware intensive
Current Multi-standard designs

- Minimum sharing of blocks
- Area and **power** consumption overhead
- Not Flexible at all
- Limited number of standards can be accommodated

Receiver for standard 1
- RF (1-2 GHz)
- BPF
- LNA
- VGA
- BPF
- LO1
- LO2

Receiver for standard 2
- RF (1-2 GHz)
- BPF
- LNA
- VGA
- BPF
- LO1
- LO2
Efficient radio transceiver: Direct Conversion

- Direct conversion + broadband ADC (1 receiver per service)
- Lowpass filter is required (~ 50-100 mW)
- 13-14 bits 80 MHz Lowpass ADC (500 mW from ADI)
- Bank of receivers, filters and ADCs
Roadmap for high-resolution Receivers

1. RF Filter → RF → Anti-Aliasing Filter → SCF, $G_m C$ → OP-RC → A/D → DSP

2. RF Filter → RF → Anti-Aliasing Filter → A/D → Dig. Filter → DSP

3. RF Filter → LNA → G → A/D → Dig. Mod. → Dig. Filter → DSP

- How much RF processing should be done before the ADC?
- The front-end must be scalable and configurable to fit multiple standards
The single-chip Transceiver Paradigm

• Modern technologies:
  “Digital intensive” System-on-Chip (SOC) environment
  ▪ Scaling of transistor dimensions in digital CMOS technologies
  ➢ Increased intra-die variability from device scaling
  ➢ Defect densities increase in newer technologies
  ➢ Yields decrease as SOC chip sizes increase
  ➢ Yield impact on analog specifications leads to process corner-based overdesign to allow for analog parameter variations
  ➢ Increased test cost

Critical Analog components must be minimized

M. Onabajo, 2011
Software radio transceiver: Design Issues

- Makes it sense to have a multi-standard solution based on this architecture?
- Bandwidth required?
- Dynamic range required?
- DTV $\Rightarrow$ SNR_{signal}=25 dB; Blockers > 45 dB; Crest factor > 20 dB
- LNA+VGA+ADC Dynamic Range over 90 dB (practical ?)
- Can you use tracking filters? (back to the past)
Ultimate goal: Reality or Dream

- Concept introduced in 1991
- Modulation/demodulation waveforms in software
- Flexible multi-standard software architecture
Analog-to-Digital Converter Data: Stated Resolution

Over 170 converters represented

slope: -1 bit/octave

R. Walden, 1999
Where we were in 99? Where we are?
This article discusses current commercially available analog-to-digital converter (ADC) technology, analyzes its key performance parameters, reviews its historical trends over the past two decades, and postulates future ADC capabilities. Based on an extensive data set including nearly 1,600 commercial products released over the past 20 years, our results show new trends compared to the well-known ADC survey by Walden [1]. It is also shown that advanced semiconductor technologies and emerging communication applications are strongly influencing the future of ADCs.

ADCs play an important role in almost all application fields, which makes it important to review the technology trends every few years. The communications industry has consistently pushed the boundaries of ADCs, and current advances in ultra-wideband (UWB) and software defined radio (SDR) continue the trend. Today, sensor technologies are becoming increasingly popular and are another area where ADCs play a major part. Here, we analyze ADCs with a general overview suitable for any area of need and present general technology trends apart from technology-specific areas.
A Little bit of History

A Little bit of History

[FIG8] Historical trends in (a) sampling speed and (b) number of bits.
Jitter and noise limitations on ENOB

Figure 1. Effective number of bits as a function of analog input frequency. The lines labeled 1999 and 2007 represent the state-of-the-art for those years. Notice the increase of 3+ ENOB over this interval.
The quantized signal presents a finite number of output values that are associated with digital codes.

- N bits used to quantize input signal.
What the problem is?

- Most common way of performing A/D conversion
  - Sample signal uniformly in time
  - Quantize signal uniformly in amplitude
- Key questions
  - How fast do we need to sample?
  - How much "noise" is added due to amplitude quantization?
  - How can we reconstruct the signal back into analog form?
Issues: Sampling, Holding and conversion

Quantization

- N bits used to quantize input signal.

The quantized signal presents a finite number of output values that are associated with digital codes.
## Properties of the Fourier Series

<table>
<thead>
<tr>
<th>$f(t)$</th>
<th>$F(\omega)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1 f_1(t) + a_2 f_2(t)$</td>
<td>$a_1 F_1(\omega) + a_2 F_2(\omega)$</td>
</tr>
<tr>
<td>$f(at)$</td>
<td>$\frac{1}{</td>
</tr>
<tr>
<td>$f(-t)$</td>
<td>$F(-\omega)$</td>
</tr>
<tr>
<td>$f(t-t_0)$</td>
<td>$F(\omega) e^{-j\omega t_0}$</td>
</tr>
<tr>
<td>$f(t) e^{j\omega_0 t}$</td>
<td>$F(\omega - \omega_0)$</td>
</tr>
<tr>
<td>$f(t) \cos \omega_0 t$</td>
<td>$\frac{1}{2} F(\omega - \omega_0) + \frac{1}{2} F(\omega + \omega_0)$</td>
</tr>
<tr>
<td>$f(t) \sin \omega_0 t$</td>
<td>$\frac{1}{2j} F(\omega - \omega_0) - \frac{1}{2j} F(\omega + \omega_0)$</td>
</tr>
</tbody>
</table>

### Convolution in time

$f_1(t) * f_2(t) = \int_{-\infty}^{\infty} f_1(x) f_2(t-x) \, dx$

$F_1(\omega) F_2(\omega)$

### Modulation properties
# Relevant properties of the Fourier Series

<table>
<thead>
<tr>
<th>$f(t)$</th>
<th>$F(\omega)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_1(t) f_2(t)$</td>
<td>$\frac{1}{2\pi} F_1(\omega) \ast F_2(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F_1(y) F_2(\omega - y) , dy$</td>
</tr>
<tr>
<td>$\delta(t)$</td>
<td>$1$</td>
</tr>
<tr>
<td>$\delta(t - t_0)$</td>
<td>$e^{-j\omega t_0}$</td>
</tr>
<tr>
<td>$\delta'(t)$</td>
<td>$j\omega$</td>
</tr>
<tr>
<td>$\delta^{(n)}(t)$</td>
<td>$(j\omega)^n$</td>
</tr>
<tr>
<td>$u(t)$</td>
<td>$\pi\delta(\omega) + \frac{1}{j\omega}$</td>
</tr>
<tr>
<td>$u(t - t_0)$</td>
<td>$\pi\delta(\omega) + \frac{1}{j\omega} e^{-j\omega t_0}$</td>
</tr>
<tr>
<td>$1$</td>
<td>$2\pi\delta(\omega)$</td>
</tr>
<tr>
<td>$t$</td>
<td>$2\pi j \delta'(\omega)$</td>
</tr>
</tbody>
</table>
### Relevant properties of the Fourier Series

<table>
<thead>
<tr>
<th>Function</th>
<th>Fourier Transform</th>
</tr>
</thead>
<tbody>
<tr>
<td>( e^{j\omega_0 t} )</td>
<td>( 2\pi \delta(\omega - \omega_0) )</td>
</tr>
<tr>
<td>( \cos \omega_0 t )</td>
<td>( \pi [\delta(\omega - \omega_0) + \delta(\omega + \omega_0)] )</td>
</tr>
<tr>
<td>( \sin \omega_0 t )</td>
<td>( -j\pi [\delta(\omega - \omega_0) - \delta(\omega + \omega_0)] )</td>
</tr>
<tr>
<td>( \sin \omega_0 t \ u(t) )</td>
<td>( \frac{\omega_0}{\omega_0^2 - \omega^2} + \frac{\pi}{2j} [\delta(\omega - \omega_0) - \delta(\omega + \omega_0)] )</td>
</tr>
<tr>
<td>( \cos \omega_0 t \ u(t) )</td>
<td>( \frac{j\omega}{\omega_0^2 - \omega^2} + \frac{\pi}{2} [\delta(\omega - \omega_0) + \delta(\omega + \omega_0)] )</td>
</tr>
<tr>
<td>( t \ u(t) )</td>
<td>( j\pi \delta'(\omega) - \frac{1}{\omega^2} )</td>
</tr>
<tr>
<td>( \frac{1}{t} )</td>
<td>( \pi j - 2\pi j \ u(\omega) )</td>
</tr>
<tr>
<td>( \frac{1}{t^n} )</td>
<td>( \frac{(-j\omega)^{n-1}}{(n-1)!} [\pi j - 2\pi j \ u(\omega)] )</td>
</tr>
<tr>
<td>( \text{sgn} \ t )</td>
<td>( \frac{2}{j\omega} )</td>
</tr>
</tbody>
</table>
Additional properties of the Fourier Series

\[ \delta_T(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) \]

\[ \omega_0 \delta_{\omega_0}(\omega) = \omega_0 \sum_{n=-\infty}^{\infty} \delta(\omega - n\omega_0) \]

\[ \int_{-\infty}^{\infty} f_1(t) f_2(t) \, dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} F_1(\omega) F_2^*(\omega) \, d\omega, \]

\[ \int_{-\infty}^{\infty} |f(t)|^2 \, dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} |F(\omega)|^2 \, d\omega, \]

\[ \int_{-\infty}^{\infty} f(x) G(x) \, dx = \int_{-\infty}^{\infty} F(x) \, g(x) \, dx. \]
Define the problem: Sampling Operation

\[ v_{\text{sig}}(t) = \cos(2\pi \cdot f_{\text{in}} \cdot t) \]

\[ t \rightarrow n \cdot T_s = \frac{n}{f_s} \]

\[ v_{\text{sig}}(n) = \cos \left( 2\pi \cdot \frac{f_{\text{in}}}{f_s} \cdot n \right) \]

\[ = \cos \left( 2\pi \cdot \frac{101}{1000} \cdot n \right) \]

\[ f_s = \frac{1}{T_s} = 1000\text{kHz} \]

\[ f_{\text{sig}} = 101\text{kHz} \]
Sampling Operation: Nyquist Rate

According to the sampling theorem: If no alias issues, then

**Ideal sampling does not add distortion but replicas of the original spectrum**
Signal Sampling Theorem

**Time domain sampling**

- Input Waveform
- Sampling Function
- Sampled Output
- Multiplication in Time Domain
- Unit Impulses
- Fourier Analysis

**Frequency Spectrum**

- Input Spectra
- Sampling Spectra
- Sampled Spectra
- Convolution in Frequency Domain

**Nyquist’s Theorem:** $f_s - f_1 > f_1 \Rightarrow f_s > 2f_1$
Signal Sampling employing a train of pulses

Time domain sampling with pulses

Spectrum

Input Waveform

Sampling Function

Sampled Output

Input Spectra

Sampling Spectra

Output Spectra

Square Wave

Period T

F(t)

A

τ/2 + τ/2

f(t) = \frac{\sin \pi f \tau}{\pi f \tau}

Input signals are not truly band limited

f_s > 2f_1

Sampling cannot be done with impulses so, amplitude of signal is modulated by

\[ \frac{\sin \pi}{\pi} \text{ envelope} \]

Because of input spectra and sampling there is aliasing and distortion
\[
\text{Sample & Hold}
\]

\[
\text{Sampling theorem:} \quad S_s(t) = \sum_{n=-\infty}^{\infty} S(t - nT_s)
\]

\[
S_s(t) = [S(t)] \cdot \left[ \sum_{n=-\infty}^{\infty} S(t - nT_s) \right]
\]

\[
S_s(f) = S(f) \ast \sum_{n=-\infty}^{\infty} \delta(n \omega_s)
\]

\[
= \frac{1}{T_s} \left[ \sum_{n=-\infty}^{\infty} S(f) \ast \delta(n \omega_s) \right] = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} S(f - n \omega_s)
\]

\[
\text{Replica of } S(f)
\]

\[
\text{Sampling period: } T_s = \frac{1}{f_s}
\]

\[
f_s = \text{Sampling frequency}
\]
Sampling

\[ x_s(t) = x_c(t) \cdot s(t) = x_c(t) \cdot \delta(t - nT) \]

\[ X_s(j\Omega) = \frac{1}{2\pi} X_c(j\Omega) \otimes S(j\Omega) \]

\[ s(t) \leftrightarrow \frac{2\pi}{T} \sum_k \delta(\Omega - k\Omega_s), \quad \Omega_s = \frac{2\pi}{T} \]

\[ X_s(j\Omega) = \frac{1}{T} \sum_k X_c(\Omega - k\Omega_s) \]
It can be shown that
\[ p_1(f) = \gamma \text{sinc} \left( \frac{\pi f T}{\gamma} \right) = \gamma \left( \frac{\sin \left( \pi f T \right)}{\pi f T} \right) \Rightarrow \text{nulls at } f = \frac{n}{\gamma} \]

"Hold" transfer function
\[ H(s) = \frac{\gamma e^{-j\omega T}}{s \text{sinc}(\pi f T)} \]
\[ S_{\text{s/H}}(f) = \frac{T}{T} \sum_{n=-\infty}^{\infty} (\text{sinc} \left( \frac{\pi f T}{\gamma} \right)) s(f - nf_s) \]
\[ S_{\text{s/H}}(f) = \sum_{n=-\infty}^{\infty} \text{sinc} \left( \frac{\pi f T}{\gamma} \right) \sum s(f - nf_s) \]
Notice in the afore equation that the delay due to $\exp(-j\pi fT)$ has being ignored.

In practice, this term corresponds to a signal delay of $T/2$ seconds!

Key observations:

- $S(f)$ is repeated (modulated by the sinc function) @ $f_s$
- $S(f)$ must be band limited to minimize alias effects
- Sinc function distorts the spectrum of $S(f)$. This effect, however, is systematic and can be compensated and usually not a major issue.
The sampling and **Held** operations generate alias frequency components and (sinc) signal distortion, respectively.

**Quantization** generates harmonic distortion components when sinusoidal input signals are used.
Distortion due to S/H errors

The S/H signal can be expressed as:

\[ V_d(t) = V_{PK} \sin(\omega_0 t) + V_{\text{error}}(t) \]

The first part of this equation does not generate any distortion since it is a pure sine wave function.

The error signal is input dependent and non-linear! Fundamental component is at the signal frequency. It can be expanded in a Fourier series and then the harmonic components can be found.
Distortion due to quantization errors: Ramp

In general, the quantization error can be expressed as:

\[ \varepsilon(v_{\text{in}}) = \frac{\Delta}{2} - (v_{\text{in}} - (N-1)\Delta) \quad \text{if} \quad (N-1)\Delta \leq v_{\text{in}} < N\Delta \]

For the case of a ramp input signal, then \( V_\text{in}(t) = Kt \), then

\[ \varepsilon(t) = -V_{\text{in}}(t) + \frac{\Delta}{2} -(N-1)\Delta \quad \text{if} \quad (N-1)\Delta \leq V_{\text{in}} < N\Delta \]

or

\[ \varepsilon(t) = -(Kt) + \frac{\Delta}{2} -(N-1)\Delta \quad \text{if} \quad (N-1)\Delta \leq Kt < N\Delta \]

You may want to find the Fourier series that represent this sawtooth error function to find the harmonic distortion components.
The problem is more complex for a sine function, the quantization error can be expressed as a kind of frequency modulation function See Van dePlassche, pp. 14:

$$\varepsilon(v_{in}) = \sum_{p=1}^{\infty} A_p \sin(p\Phi)$$

- $A_p = 0$ if $p$ is even
- $A_p = \text{complex Bessel function}$ if $p$ is odd

Interestingly, this complex expression can be simplified for n-bit converters, leading to a very simple result for the third order harmonic distortion:

$$\text{HD}_3 = \frac{\text{Power of 3rd harmonic component}}{\text{Power of fundamental component}} = 2^{-1.5n} \quad n \geq 1$$
Distortion due to quantization errors: Sinusoidal input

- For 10-Bit ADC, HD3 is in the range of -90.31 dB
- HD3 reduces at a rate of -9dB per additional Bit
- After 10 bits, this distortion can be easily ignored

\[ HD_3 \approx 2^{-1.5n} \]
For the case of intermodulation distortion using a couple of test tones, it is found

$$\text{IM}3 \approx 2^{-2n}$$

Notice that SNR is over 20dB more relevant than HD3 for $n \geq 7$
**Alias issue if undersampling**

$$f_s = \frac{1}{T_s} = 1000kHz$$

$$f_{sig} = 899kHz$$

$$v_{sig}(n) = \cos\left(2\pi \cdot \frac{899}{1000} \cdot n\right) = \cos\left(2\pi \left[ \frac{899}{1000} - 1 \right] \cdot n\right) = \cos\left(2\pi \cdot \frac{101}{1000} \cdot n\right)$$
Aliasing Error
(not enough samples per frequency cycle \( f_S < 2f_{MAX} \))

Original Signal

Reconstructed Waveform

First Order Interpolation
(Reconstruction Using Filter or Vector Generator)

Samples

Zero Order Interpolation
(Reconstruction Directly from D/A Converter)
Under-sampling of a broadband signal

In order to prevent aliasing, we need

$$f_{\text{sig, max}} < \frac{f_s}{2}$$

The sampling rate $f_s = 2 \cdot f_{\text{sig, max}}$ is called the Nyquist Rate

Two possibilities

- Sample fast enough to cover all spectral components, including "parasitic" ones outside band of interest
- Limit $f_{\text{sig, max}}$ through filtering
The sampling and Held operations generate alias frequency components and (sinc) signal distortion, respectively.

Error is an odd function (no even harmonic distortions, why?)

Quantization generates harmonic distortion components when sinusoidal input signals are used.

\[ S_{in}(t) = S_q(t) + \text{Error}(t) \]
Distortion due to quantization errors

Commensurate $f_s$ and $f_{in}$

- Periodic sampling points result in periodic quantization errors.
- Periodic quantization errors result in harmonic distortion.
**ADC metrics: Quantization error**

- Signal is sampled at given instants
- Signal is encoded to a limited number of codes resulting in quantization noise (random signals) and distortion (periodic signals)

\[ \Delta = \frac{V_{\text{full-scale}}}{2^N} \]

- **Binary code**
  - 00000
  - 00001
  - 00010
  - 00100
  - 01000
  - 10000

- **Input Signal**
  - Time
  - Code
  - Code values: 00111, 00110, 00101, 00100, 00111, 00110, 00101, 00100

- **Quantization error**
  - \( \pm \Delta/2 \)

- **Probability density**
  - \( p_e = \frac{\Delta^2}{12} \)
Elements of Transfer Diagram for an Ideal Linear ADC

- **Conversion Code Table**
  - **Range of Analog Input Values**
    - 4.5 - 5.5
    - 3.5 - 4.5
    - 2.5 - 3.5
    - 1.5 - 2.5
    - 0.5 - 1.5
    - 0 - 0.5
  - **Digital Output Code**
    - 0 ... 101
    - 0 ... 100
    - 0 ... 011
    - 0 ... 010
    - 0 ... 001
    - 0 ... 000

- **Ideal Straight Line**
  - Step Width (1 LSB)
  - Midstep Value of 0 ... 011

- **Quantization Error**
  - ±1/2 LSB

- **Inherent Quantization Error (± 1/2 LSB)**
What the fundamental problem is?

Mapping an infinite resolution analog signal into a digital but finite resolution representation.

Quantization Error

\[ \Delta = \frac{V_{FS}}{2^N} = \text{LSB} \]

\[ V_{in} \in [0, V_{FS}] \]

\[ \varepsilon = D_{out} \Delta - V_{in} = D_{out} \left( \frac{V_{FS}}{2^N} \right) - V_{in} \]

\[ -\frac{\Delta}{2} \leq \varepsilon \leq \frac{\Delta}{2} \]

"Random" quantization error is regarded as noise.
Quantization noise for Random (Ramp) input signal

Quantization Noise

Assumptions:
- N is large.
- \(0 \leq V_{in} \leq V_{FS}\) and \(V_{in} \gg \Delta\).
- \(V_{in}\) is active.
- \(\varepsilon\) is Uniformly distributed.
- Spectrum of \(\varepsilon\) is white.

\[
\sigma_{\varepsilon}^2 = \int_{-\Delta/2}^{\Delta/2} \varepsilon^2 \frac{1}{\Delta} \, d\varepsilon = \frac{\Delta^2}{12}
\]

Signal-to-Quantization Noise Ratio (SQNR)

Assume $V_{\text{in}}$ is sinusoidal with $V_{\text{p-p}} = V_{\text{FS}}$,

$$SQNR = \frac{V_{FS}^2}{\sigma^2} = \frac{(2^N \Delta)^2}{8} = \frac{\Delta^2}{12} = 1.5 \times 2^{2N}.$$  

$$SQNR = 6.02N + 1.76\,\text{dB}.$$  

- SQNR depicts the theoretical performance of an ideal ADC.
- In reality, ADC performance is limited by many other factors:
  - Electronic noise (thermal, 1/f, coupling, and etc.)
  - Distortion (measured by THD, SFDR)
**ADC metrics: SQNR**

The maximum Signal-to-Quantization Noise ratio (SQNR) for an N-bit ADC:

\[
\text{SQNR}_{\text{ideal}} = \frac{P_{\text{signal}}}{P_{\text{noise}}} = \frac{A^2 / 2}{\Delta^2 / 12} = \left(\Delta \cdot 2^N / 2\right)^2 = 6.02N + 1.76 \, \text{dB}
\]

- For an ADC with a measured SNDR, the effective number of bits is defined as:

\[
\text{ENOB} = \frac{\text{SNDR(dB)} - 1.76}{6.02}
\]
The dynamic range of a system is equal to the signal to noise ratio measured over a bandwidth equal to half of the sampling (Nyquist) frequency.

Then,

\[ \sigma^2 = \frac{q^2}{12} \]

Is the total while the quantization noise density (quantization noise measured in a bandwidth of 1 Hz)

\[ \text{Noisedensity} = \frac{2\sigma^2}{f_s} = \frac{q^2}{6f_s} \]
DFT Spectrum of Quantized Signal

- $N = 10$ bits.
- 8192 samples, only $f = [0, f_s/2]$ shown.
- Normalized to $V_{in}$.
- $f_s = 8192$, $f_{in} = 779$.
- $f_{in}$ and $f_s$ must be incommensurate.

$$ENOB = \frac{SQNR - 1.76}{6.02}$$

Incommensurate $f_s$ and $f_{in}$

- Sampling frequency $f_s$ is fixed.

- Input frequency $f_{in}$ is chosen to satisfy (a) integer number of cycles $C$ and (b) $N / C = f_s / f_{in}$ is incommensurate. An easy way is to make $N$ a power of 2 and $C$ a prime number.

- Windowing lifts the need to have an integer number of cycles. Good for measurements.

- Pick $N$ depending on noise floor requirements: The DFT noise floor is $10 \times \log_{10}(N/2)$ below the noise floor. Then DFT noise floor $= -\text{SNR}_{0\text{dBFS}} - 10 \times \log_{10}(N/2)$. 
Spectrum Leakage

- TD samples must include integer number of cycles of input signal.
- Easily ensured in simulation; in testing, windowing can be applied.
DFT Spectrum with Distortion

- SNDR = 59.16 dB
- ENOB = 9.535 bits
- THD = 63.09 dB
- SFDR = 64.02 dB

- Signal-to-noise plus distortion ratio (SNDR)
- Total harmonic distortion (THD)
- Spurious-free dynamic range (SFDR)

\[
ENOB = \frac{SNDR - 1.76}{6.02}
\]

- High-order harmonics alias back, visible in \([0, f_s/2]\) band.
- HD3 @ 779x3+1 = 2338, HD9 @ 8192-9x779+1 = 1182.
Practical Limitations

Dynamic Performance

- Peak SNDR limited by large-signal distortion of the converter.
- Dynamic range implies the “theoretical” SNR of the converter.

\[ SNR = 10 \log \left( \frac{V_{in}^2 / 2}{\Delta^2 / 12 + \sigma_N^2} \right) \approx LOG(V_{in}) \]
Digital to Analog Converters

D/A Converter Transfer Characteristic

\[ V_{out} = V_{FS} \cdot \sum_{i=1}^{N} b_i \cdot 2^{-i} = \Delta \cdot \sum_{i=1}^{N} b_i \cdot 2^{N-i} \]

- \(N = \) # of bits
- \(V_{FS} = \) Full-scale input
- \(\Delta = V_{FS}/2^N = 1\) LSB
- \(b_i = 0\) or 1
- Multiplication

**Note:** \(V_{out} \) (\(b_i = 1\), for all \(i\)) = \(V_{FS} - \Delta = V_{FS}(1-2^{-N}) \neq V_{FS}\)
Practical Definitions

- Definition

\[ SNDR = \frac{Signal\ Power}{Noise\ and\ Distortion\ Power} \]

- Noise and distortion power includes all bins except DC and signal

- Effective number of bits

\[ ENOB = \frac{SNDR(dB) - 1.76dB}{6.02dB} \]
Practical Limitations

Ideal DAC Transfer Characteristic

![Graph showing ideal DAC transfer characteristic. The graph plots \( V_{\text{out}} \) against \( D_{\text{in}} \), with \( V_{FS} \) and \( V_{FS}/2 \) marked on the vertical axis. The horizontal axis represents the input digital code from 000 to 111.]
Practical Limitations

Monotonicity

Quite critical issue!

Offset

Usually not a major issue
Practical Limitations: Offset error

(a) ADC
Offset error of a Linear 3-Bit Natural Binary Code Converter (Specified at Step 000)

(b) DAC

0 1 2 3
Analog Output Value

0 1 2 3
Analog Output Value (LSB)

0 0 1
Digital Input Code

0 0 1
Digital Input Code

000 001 010 011
Digital Output Code

000 001
Digital Output Code

0 1
Nominal Offset Point

0 1
Nominal Offset Point

Ideal Diagram
Actual Diagram
Ideal Diagram
Actual Diagram

Offset Error (+1 1/4 LSB)

+1/2 LSB

Actual Offset Point

Actual Offset Point

Texas A&M University 66 Spring, 2019
Practical Limitations

Gain Error

Differential and Integral Nonlinearity

\[
DNL(n) = \frac{V_{out}(n) - V_{out}(n-1) - \Delta}{\Delta}
\]

\[
INL(n) = \sum_{i=1}^{n} DNL(i)
\]

Usually not a major issue

Quite critical issue!
Practical Limitations: Gain Error

Gain Error of a Linear 3-Bit Natural Binary Code Converter (Specified at Step 111), After Correction of the Offset Error
Practical Limitations: Differential Error

(a) ADC

Digital Output Code

Analog Input Value (LSB)

0 ... 110
0 ... 101
0 ... 100
0 ... 011
0 ... 010
0 ... 001
0 ... 000

1 LSB

Differential Linearity Error (1/2 LSB)

Differential Linearity Error (−1/2 LSB)
Practical Limitations

DNL and INL

- DNL measures the incremental (local) nonlinearity.
- INL measures the cumulative (global) nonlinearity.
Practical Limitations: Integral error

(a) ADC
Practical Limitations

Measure DNL and INL

Method I (endpoint stretch)
\[ \Sigma(\text{INL}) \neq 0 \]

Method II (LS fit & stretch)
\[ \Sigma(\text{INL}) = 0 \]
Practical Limitations: Absolute Accuracy

Figure 7. Absolute Accuracy (Total) Error
Analog to Digital Converters

Ideal ADC Transfer Characteristic

Usually the effects of the systematic offsets can be minimized through calibration or accounted in digital domain.

Note the systematic offset!
Digital to Analog Converters

DNL and INL

\[ DNL(n) = \frac{V_{out}(n) - V_{out}(n-1) - \Delta}{\Delta} \]

\[ INL(n) = \sum_{i=1}^{n} DNL(i) \]
Practical Limitations

DNL and INL

\[ DNL(n) = \frac{V_{out}(n) - V_{out}(n-1) - \Delta}{\Delta} \]

\[ INL(n) = \sum_{i=1}^{n} DNL(i) \]
Practical Limitations

DNL and INL

\[
DNL(n) = \frac{V_{out}(n) - V_{out}(n-1)}{\Delta} \quad \text{INL}(n) = \sum_{i=1}^{n} DNL(i)
\]

- Nonmonotonic?
- DNL < -1?
- Missing code?
Practical Limitations

10-bit Pipeline ADC

- Code density test (CDT)
- 1024 codes
- No missing code!
- Plotted against the digital code, not $V_{in}$

DNL must be smaller or equal to 1 LSB
Practical Limitations

Nonmonotonic Codes

- Two transition steps for one code?! How to plot INL/DNL?
- CDT can be very misleading to determine the static nonlinearity
Offset Voltages

Max. Offset vs. Resolution

- DNL < 0.5 LSB
- Large $V_{FS}$ relaxes offset tolerance
- Small $V_{FS}$ benefits conversion speed (settling, linearity)
Practical Limitations

A/D Converter Architectures

- Nyquist-rate converters
- Oversampling converters

Sub-sampling

\[ |X(j\omega)| \]

\[ f_m = \frac{f_s}{2} \quad 2f_s \quad 3f_s \]

Bandpass oversampling

\[ |X(j\omega)| \]

\[ f_m \quad f_s/2 \]

\[ \text{OSR} = \frac{f_s}{2f_m} \]
Nyquist-Rate ADC’s

- The “black box” version of the quantization process
- Digitizes the input signal up to the Nyquist frequency \( (f_s/2) \)
- Minimum sampling frequency \( (f_s) \) for a given input bandwidth
- Each sample is digitized to the maximum resolution of the converter

![Diagram of A/D converter with analog input and digital output](image)
Nyquist-Rate ADC’s

- Word-at-a-time (1 step)†
  - Flash
  - Folding
- Level-at-a-time ($2^N$ steps)
  - Integration (Serial)
- Bit-at-a-time (N steps)
  - Successive approximation
  - Algorithmic (Cyclic)
- Partial word-at-a-time ($1<M\leq N$ steps)
  - Subranging
  - Multi-step
  - Pipeline

† the number in the parentheses is the “latency” of conversion, not “throughput”.
Oversampling ADC’s

- Sample rate is well beyond the signal bandwidth.
- Coarse quantization is combined with feedback to provide an accurate estimate of the input signal on an “average” sense.
- Quantization error in the coarse digital output can be removed by the digital decimation filter.
- The resolution/accuracy of oversampling converters is achieved in a sequence of samples (“average” sense) rather than a single sample; the usual concept of DNL and INL of Nyquist converters are not applicable.

![Diagram showing the oversampling ADC process: Analog input, reference voltage, A/D converter, digital output, decimation filter, and quantized output.](Image)
Oversampling ADC’s

• Predictive
  – Delta modulation

• Noise shaping
  – Sigma-delta modulation
  – Multi-level (quantization) sigma-delta modulation
  – Multi-stage (cascaded) sigma-delta modulation (MASH)
Building Blocks for Data Converters

- Comparators (Preamp and Latch)
- Sample-and-Hold (Track-and-Hold) Amplifier
- Operational Amplifier (ELEN 474, ELEN609)
- Switched-Capacitor Amplifiers, Integrators and Filters (ELEN622)
- Voltage and Current DAC’s
- Current Sources (ELEN 474)
- Voltage/Current/Bandgap References (ELEN474)