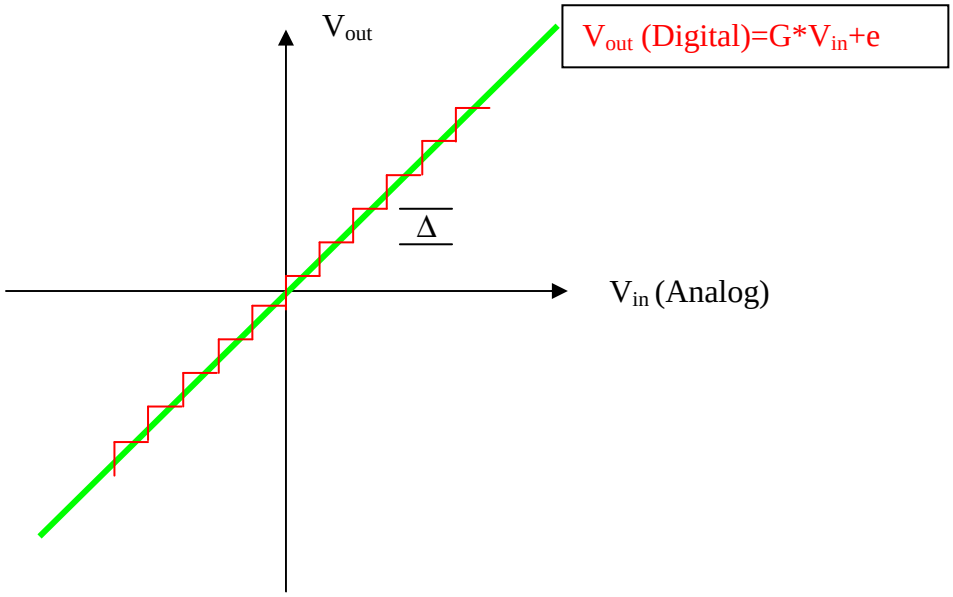
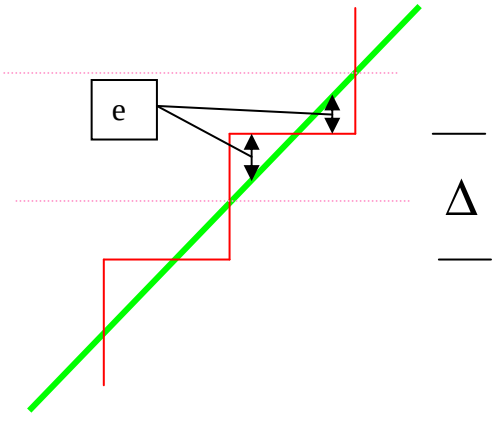
Quantization Noise

Quantization is the mapping of a range of analog voltage to a single value.



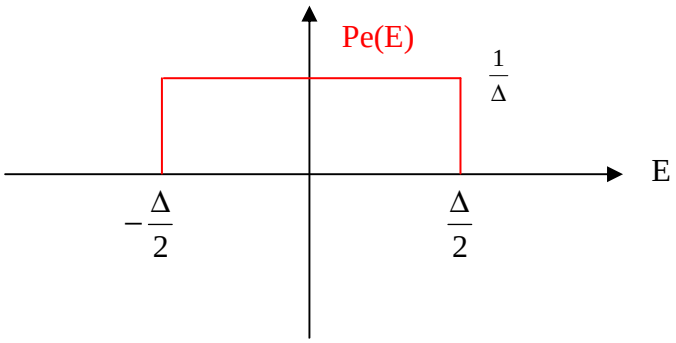
Staircase curve of a linear N Bit ADC Converter

- Assume that the input in “busy”, moderate signal level.
- Green curve is a scaled version of Vin without any quantization.
- Red curve is the ADC Output.
- Δ is the step size of the converter.



Zoom in of Staircase

- Pink dots show that analog range that maps to an ADC Value.
- Black arrows show the Quantization error for 2 points.



PDF of Quantization Error

- Quantization error is uniformly distributed.
- Integrates to 1
- $\sigma_{QNoise}^{\ 2} = E(e^2) = \int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} \frac{1}{\Delta} e^2 de = \frac{1}{\Delta}\frac{e^3}{3}\Big|_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} = \frac{1}{\Delta}\left[\frac{\left(\frac{\Delta}{2}\right)^3}{3} - \frac{\left(-\frac{\Delta}{2}\right)^3}{3}\right] = \frac{1}{\Delta}\left[\frac{\Delta^3}{24} - \frac{-\Delta^3}{24}\right] = \frac{\Delta^2}{12}$
- $\sigma_{QNoise}^{\ 2} = \text{Quantization Noise Power} = \frac{\Delta^2}{12}$
- $\sigma_{QNoise} = V_{QNoise\_rms} = \frac{\Delta}{\sqrt{12}}$
- RMS value for a full scale sinusoidal input is

$$V_{MaxSignal\_rms} = \frac{\left(\frac{2^N}{2}\right)}{\sqrt{2}}\Delta$$

- $$\text{Max SNR} = 20\log\left(\frac{\frac{\left(\frac{2^N}{2}\right)}{\sqrt{2}}\Delta}{\frac{\Delta}{\sqrt{12}}}\right) = 20\log\left(\frac{\sqrt{6}}{2}2^N\right) = 20\log\left(\frac{\sqrt{6}}{2}\right) + 20N\log(2)$$

$$= 1.76 + 6.02\text{N}$$

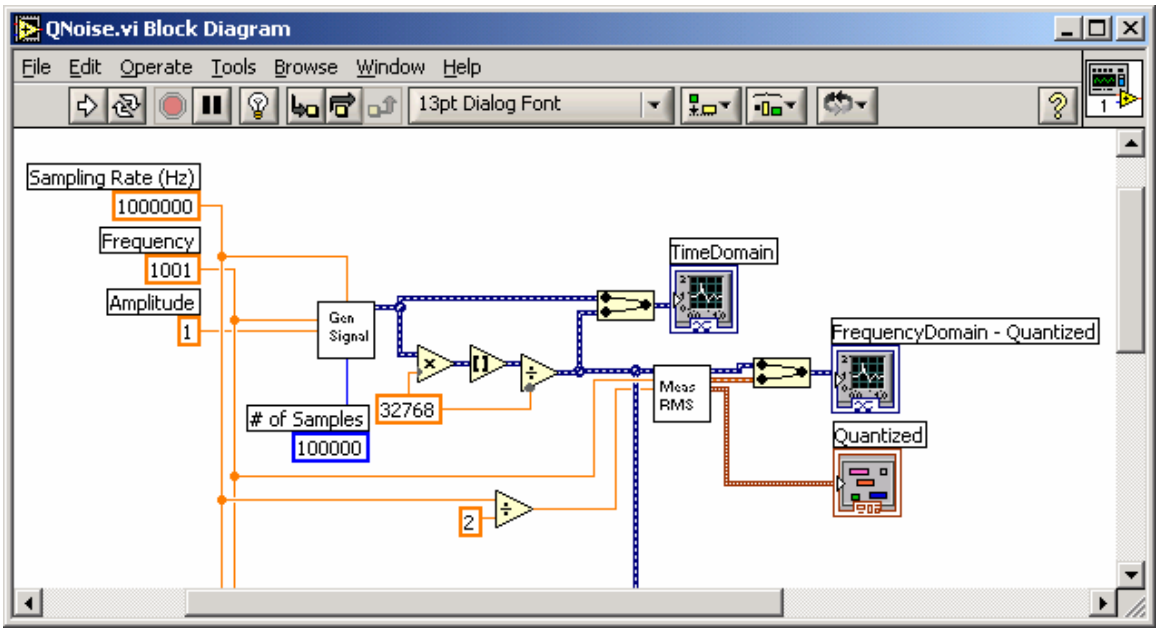
- $\text{N} = \text{Effective\# of Bits} = \frac{\text{Max SNR -1.76}}{6.02}$

For example, consider a 16 Bit, 1MHz ADC with ±1V Range, with a 1001 Hz input:

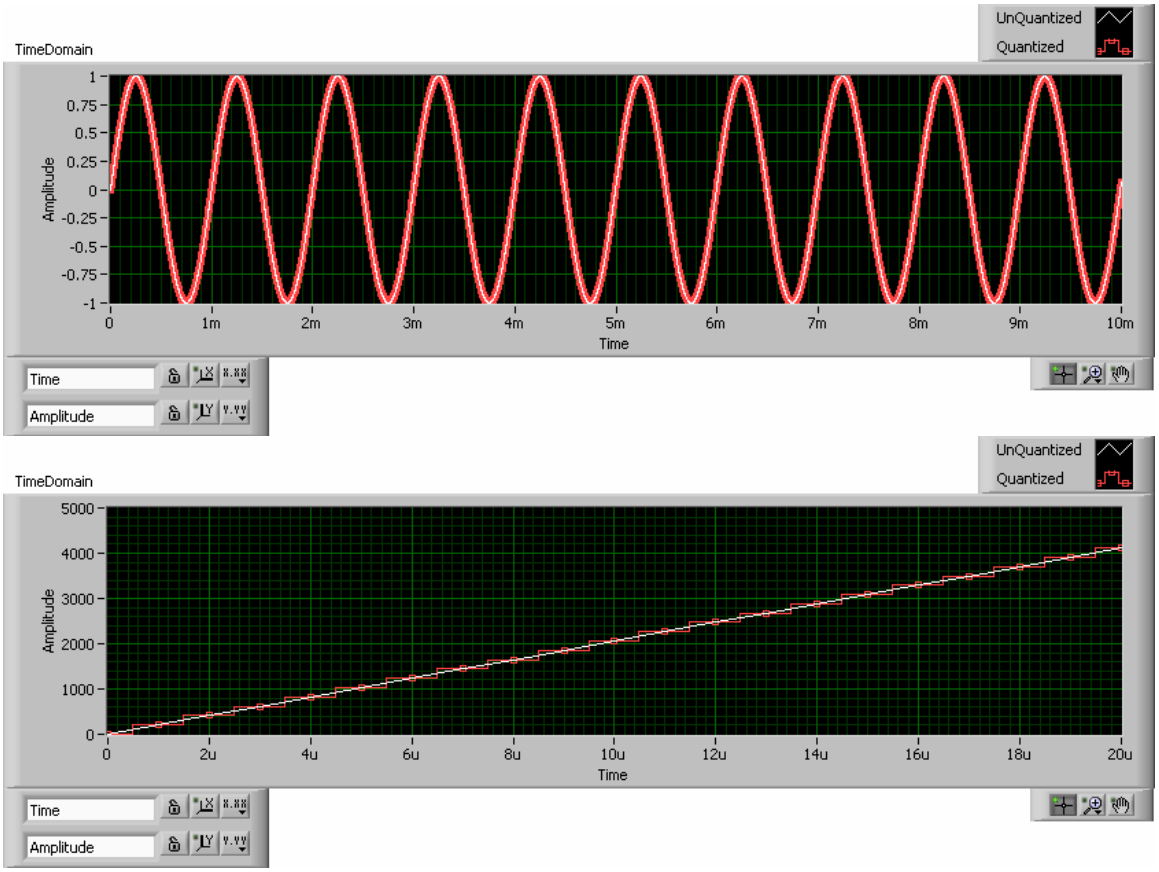
- $\Delta = \frac{1\,\mathrm{V}}{32768\,\mathrm{Steps}} = 30.51\frac{\mu V}{Step}$
- $Noise(Vrms) = \sigma = \frac{\Delta}{\sqrt{12}} = 8.81\mu\mathrm{V}$
- $MaxSNR = 20\log\left(\frac{\frac{1}{\sqrt{2}}}{8.81\mu}\right) = 98.1\,\mathrm{dB}$
- $\mathrm{Effective\,\#\,Bits} = \frac{\mathrm{Max\ SNR\ -1.76}}{6.02} = \frac{98.1-1.76}{6.02} = 16\ \mathrm{Bits}$

To simulate this 16 Bit converter

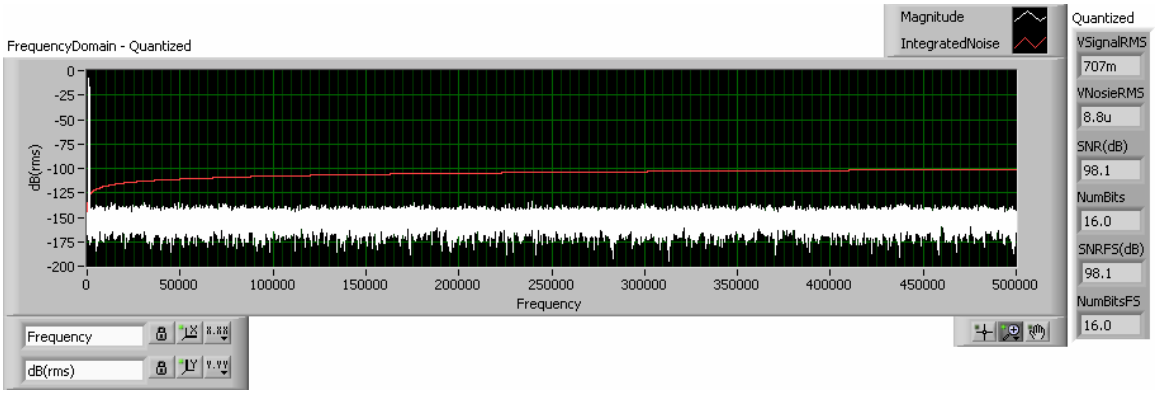
- Create a 100000 samples of a 1V Double Precision sine wave, and convert it to 16 bit integers by multiplying by 32768 and rounding the result. Divide the result by 32768 to refer it to the input of the ADC. This is the 16 bit quantized version of the input.
- Hodie Window the quantized signal, take the DFT, and integrate the Power Spectral Density in the 17 bins around the signal bin.
  - $\Delta f = \frac{F_s}{N} = \frac{1MHz}{100000} = 10Hz$
  - $\mathrm{Signal\ Bin} = \frac{Fsignal}{\Delta f} = \frac{1001Hz}{10Hz} = 100$
- The center of these 17 bins is bin 100, so we want to integrate from bin 92 to 108 inclusive to measure the Signal Power. The square root of this integral is the rms value of the signal.
- To measure the Quantization noise, we should replace the 17 signal bins with the value of the leftmost signal bin (92) to approximate the quantization noise in the signal bins. The integral of this modified PSD from 0 to $F_s/2$ is the noise power, σ^2. The square root of this integral is the rms value, σ.



LabVIEW to simulate the 16 bit ADC.



Time Domain Double Precision Input and Quantized ADC output (Bottom graph is zoomed in version)



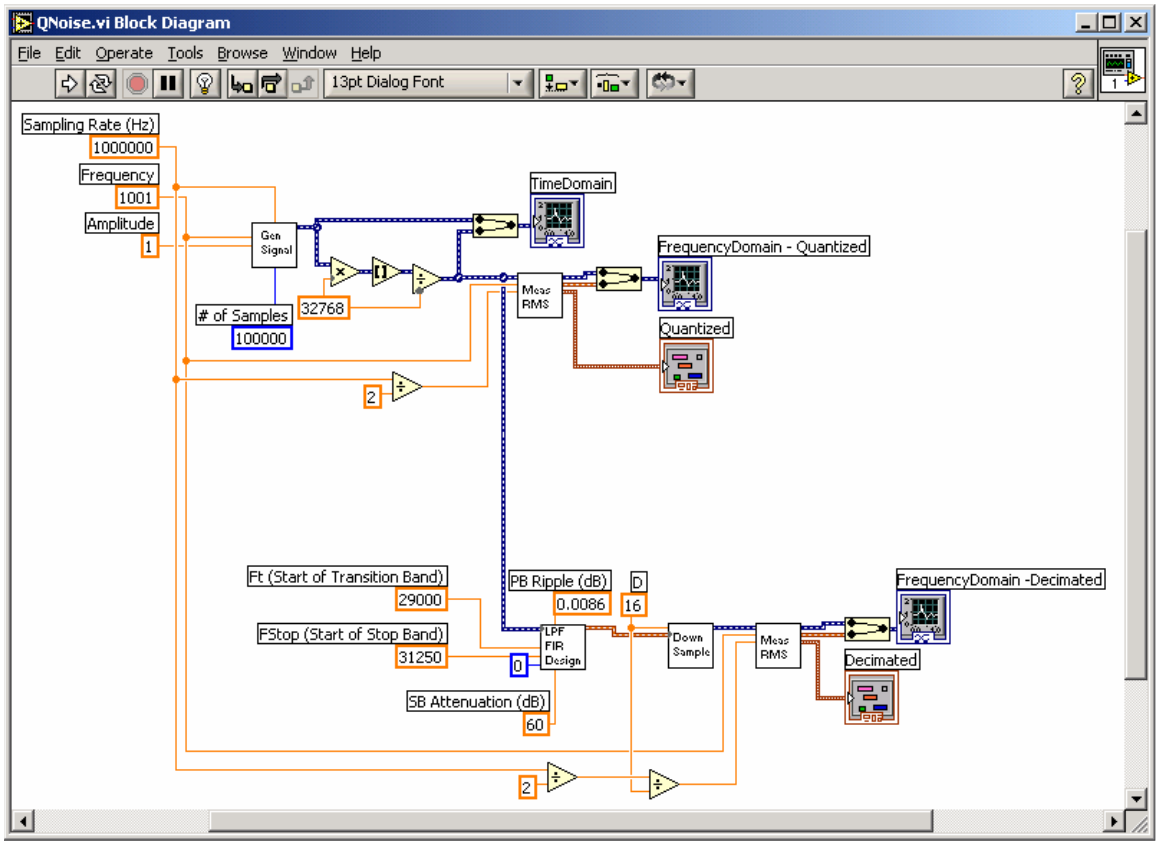
Magnitude Spectrum in dB (rms) and the Integrated Noise (dB rms) as a function of Bandwidth of the Quantized ADC output.

- The integral of the 17 signal bins in the PSD is shown in theVSignalRMS indicator and has the value $0.707 = \frac{1}{\sqrt{2}}$.
- The integral of the PSD, dodging the signal bins as described above is shown in the VNoiseRMS indicator and has the value $8.8\mu V$.
- These are the same values as predicted above, i.e., we correctly modeled a 16 Bit ADC.

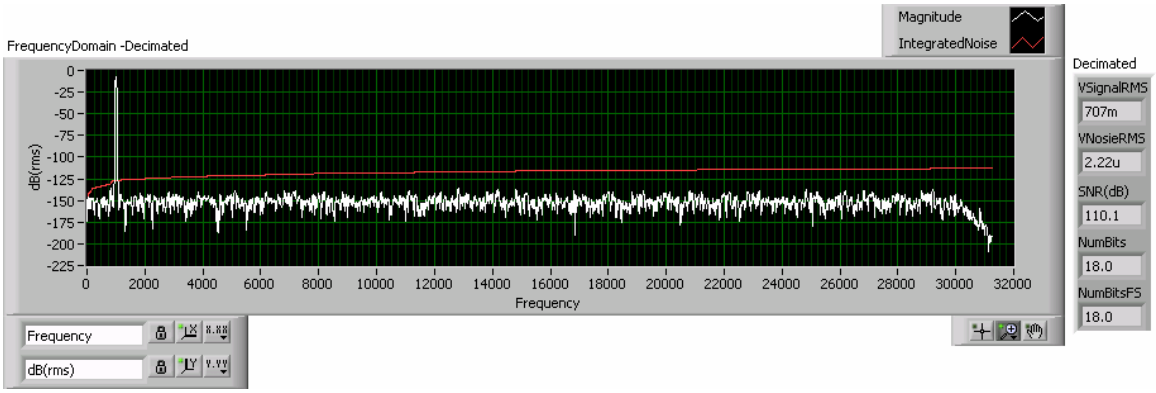
Oversampling a Non-Oversampling ADC

- Low Pass Filtering the ADC output will reduce the noise power and yield more effective bits.
- Downsample the ADC to the new rate. The new rate must be at least 2 times bigger than Fstop of the Low Pass Filter to prevent aliasing.
- Low Pass Filtering and Downsampling is called Decimation.
- Pick up ½ bit per octave of oversampling (Equation 8 of Hauser). Why?

For example, let's Decimate the above example by 16 to 62.5 KHz. We are reducing the bandwidth by 4 octaves which should give us an 18 bit ADC.



The lower portion of the code does the Decimation by 16.



- 18 Bits! Verify?

References:

- "Mixed-Signal System Design and Modeling", Eric Swanson, Fall Semester 2002.
- Dr. Morley's EE437 Lecture Notes, Fall 2003.