Sampled systems

Lecturer: Krébesz, Tamás

3.3 AD-DA converters, DAQ cards

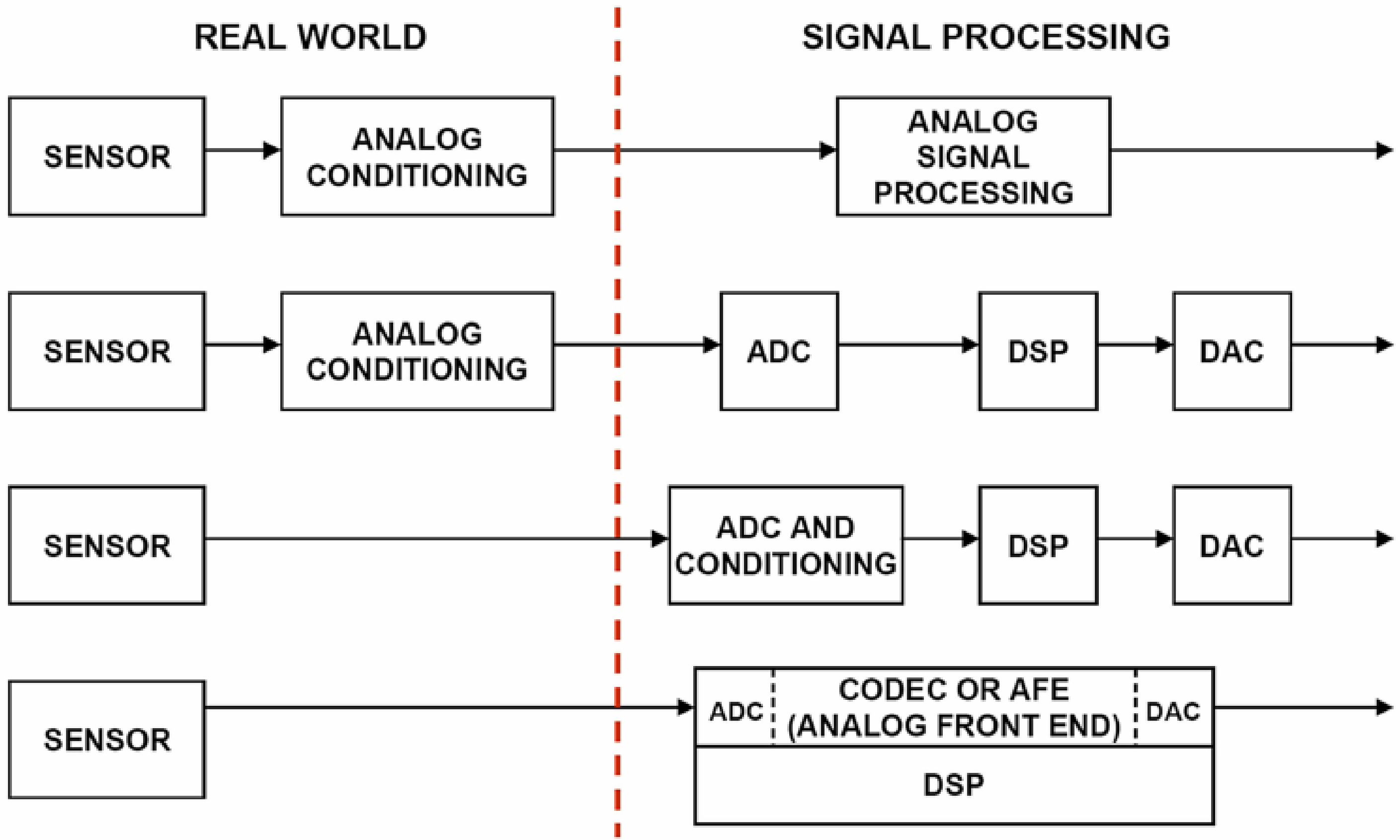
Webster's New Collegiate Dictionary defines a signal as "A **detectable** (or measurable) **physical quantity** or impulse (as voltage, current, or magnetic field strength) by which messages or **information** can be transmitted."

How a signal can be processed using digital computer?

Contents:

- Analog-to-digital converters (ADCs)
- Digital-to-analog converters (DACs)
- Decimation, interpolation
- Data Acquisition (DAQ) cards

3.3.1 Analog-to-digital converters - Real-world signal processing

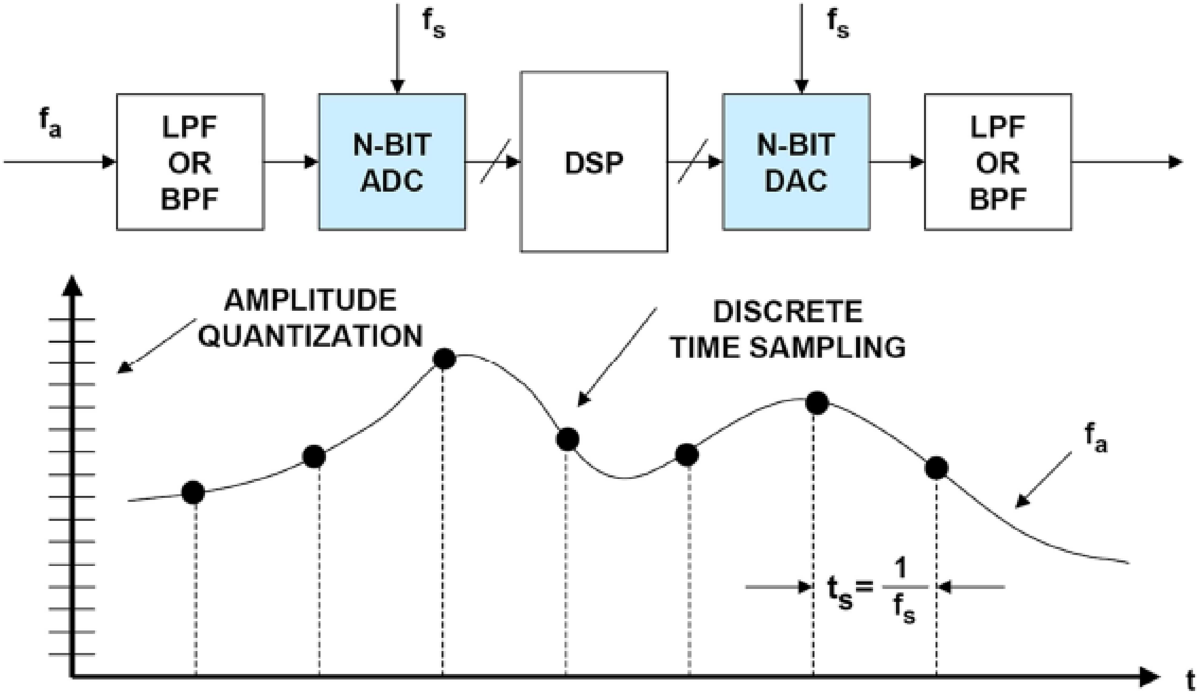


Trend to perform as much signal processing in the digital domain as possible

Sampled data systems

Prior to ADC, analog signal usually passes through signal conditioning (amplification, attenuation, filtering, etc.)

Sampled data system:



Operation of sampled data systems

- (input) LPF/BPF to remove unwanted signals and prevent *aliasing*
- signal to ADC is continuously sampled at a sampling rate f_s (ADC presents a new sample to the DSP at this rate)
- in case of real time operation, the DSP must perform all computation (e.g. filtering) within the sampling interval, $t_s = 1/f_s$ (new sample generated for DAC before the next sample arrives from ADC)
- (output) LPF/BPF to remove the *image* frequencies

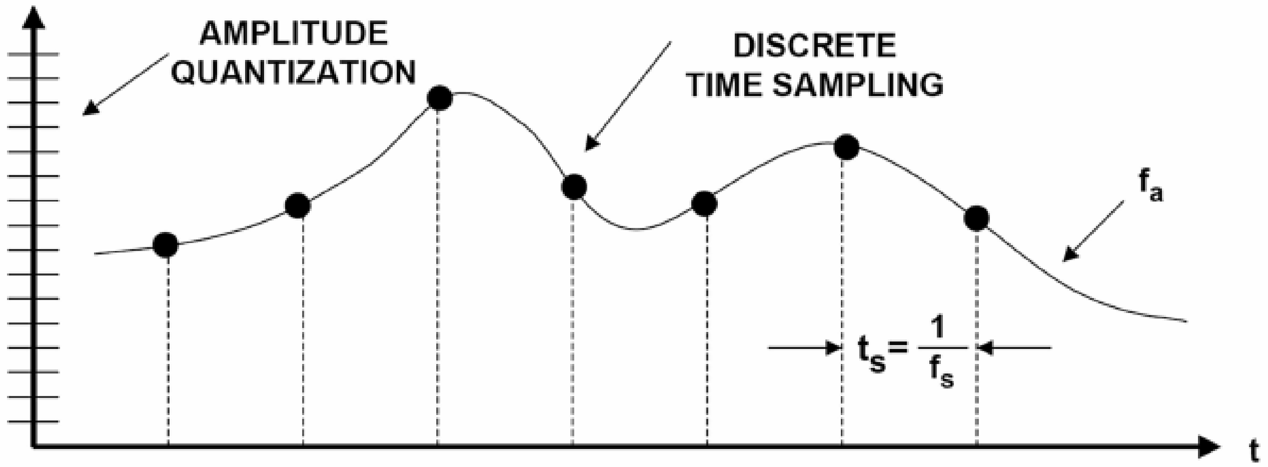
Some remarks:

- in case of FFT a block of data is first transferred to the DSP memory
- DAC is only required if DSP data must be converted back to analog (audio applications)
- no ADC is required when DSP is solely used for signal generation (CD player)

Two key concepts in ADCs (and DACs)

- discrete time sampling
- finite amplitude resolution due to quantization

Discrete time sampling



- a sample is taken from continuous analog signal at discrete intervals, $t_s = \frac{1}{f_s}$
- t_s must be chosen carefully to insure an accurate representation of the original analog signal $\rightarrow$ **Nyquist's criteria must be met**

Nyquist's criteria

The more samples taken (faster sampling rate), the more accurate the digital representation, but if fewer samples are taken, a point is reached where information about the signal is lost

Q: What is fewer? How can this lower limit be defined?
A: Using Nyquist's criteria: the sampling frequency, f_s must be at least twice of f_a, the highest frequency content of the signal to be sampled

Consequences of Nyquist's criteria:

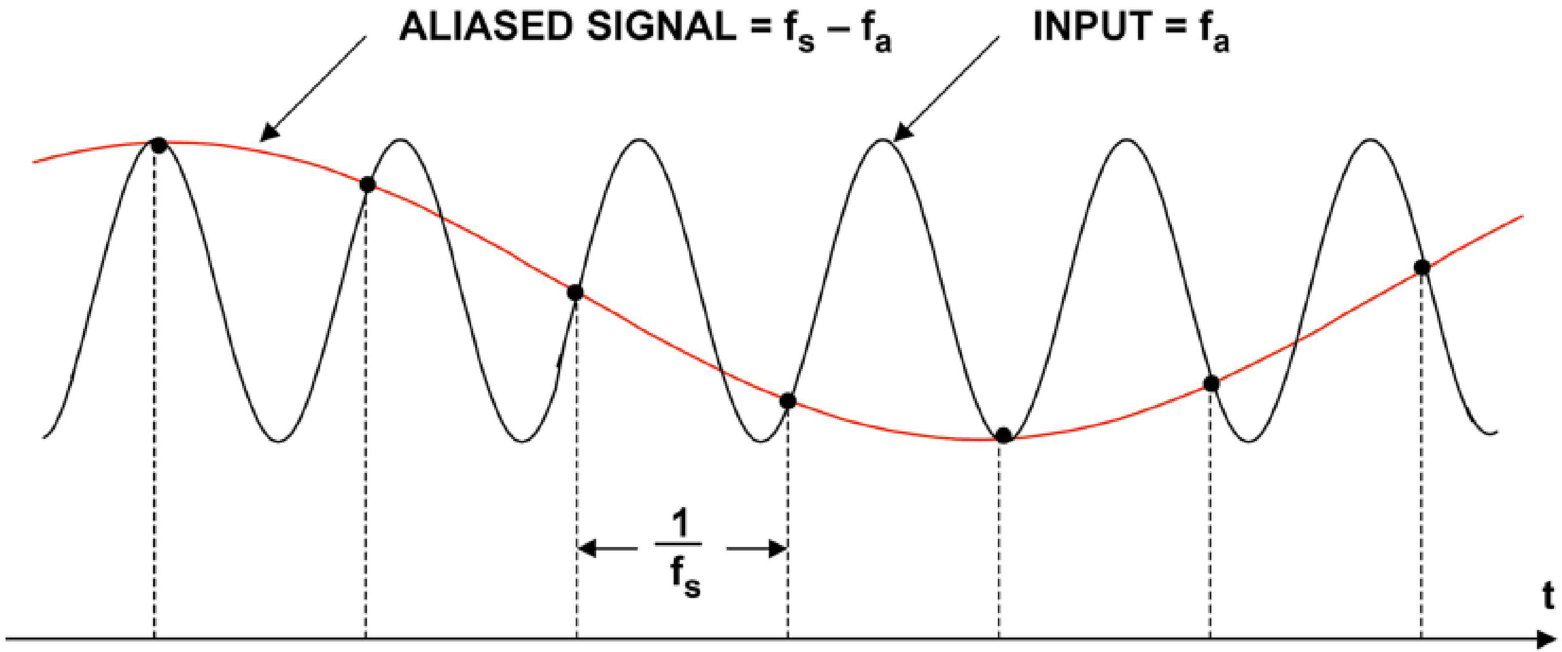
- if $f_s \geq 2f_a \Rightarrow$ no information loss, no aliasing
- if $f_s < 2f_a \Rightarrow$ information loss, aliasing

In summary: if Nyquist criteria is met the information can be reconstructed from the discrete time sampled signal without any loss on the original analog signal

Aliasing in the time domain

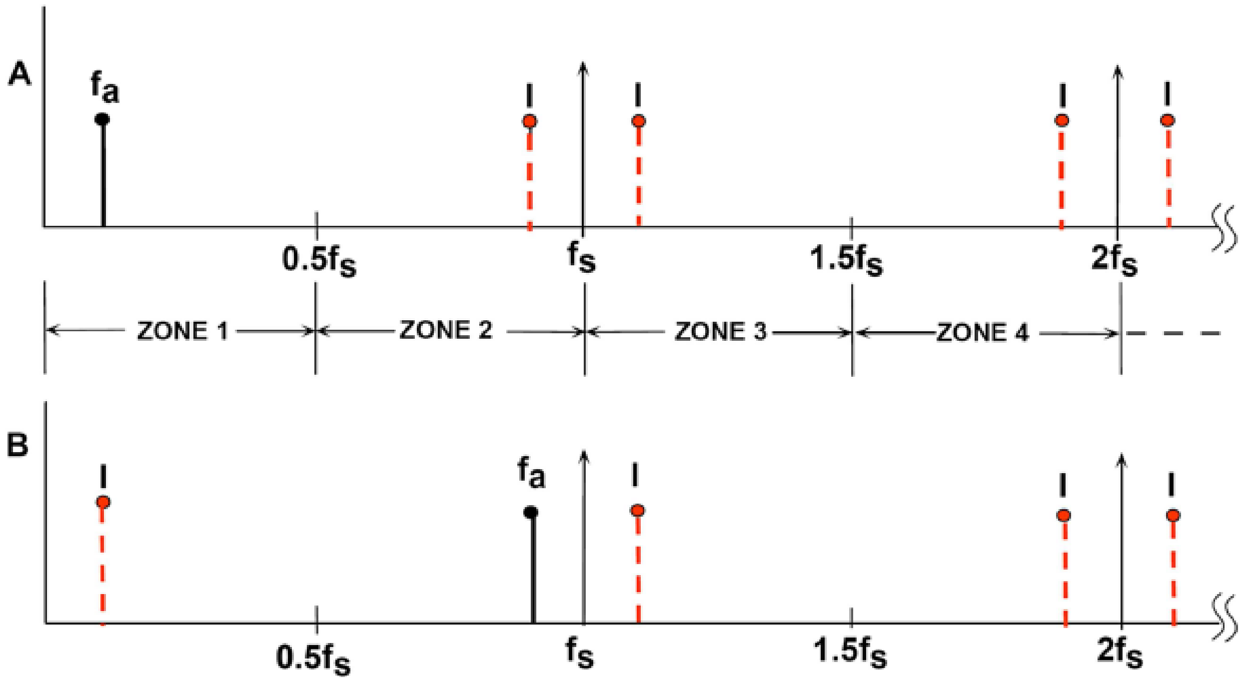
If the sampling frequency is less than twice the analog signal bandwidth, a phenomena known as aliasing will occur

Let assume $f_s \approx f_a$ but $(f_s > f_a) \wedge (f_s < 2f_a) \Rightarrow$ Nyquist's criteria is violated



Aliasing in the frequency domain

Analog signal f_a sampled at f_s using ideal sampler has images (aliases) at $|\pm K f_s \pm f_a|$, $K \in \mathbb{N}$

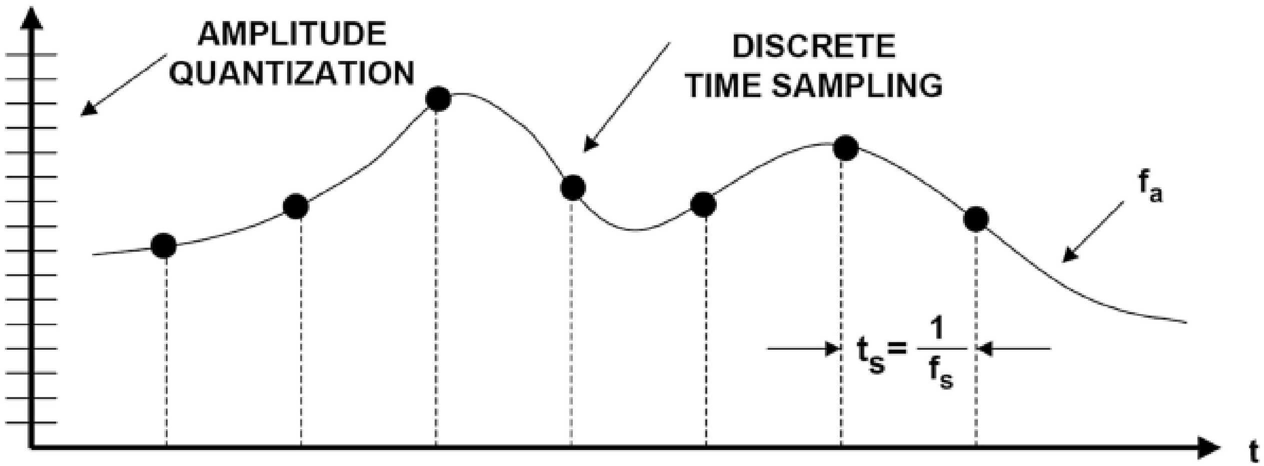


- Figure **A:** $f_s > 2f_a \Rightarrow$ Nyquist's criteria is met
- Figure **B:** $f_s \approx f_a$ but $f_s > f_a \wedge f_s < 2f_a \Rightarrow$ Nyquist's criteria is violated

Two key concepts in ADCs (and DACs)

- discrete time sampling
- finite amplitude resolution due to quantization

Finite amplitude resolution due to quantization



- The ADC output (DAC input) is digital $\Rightarrow$ the signal is quantized
- *N-bit ADC (DAC)* works with N-bit words representing one of 2^N possible analog levels (tipically voltages) each

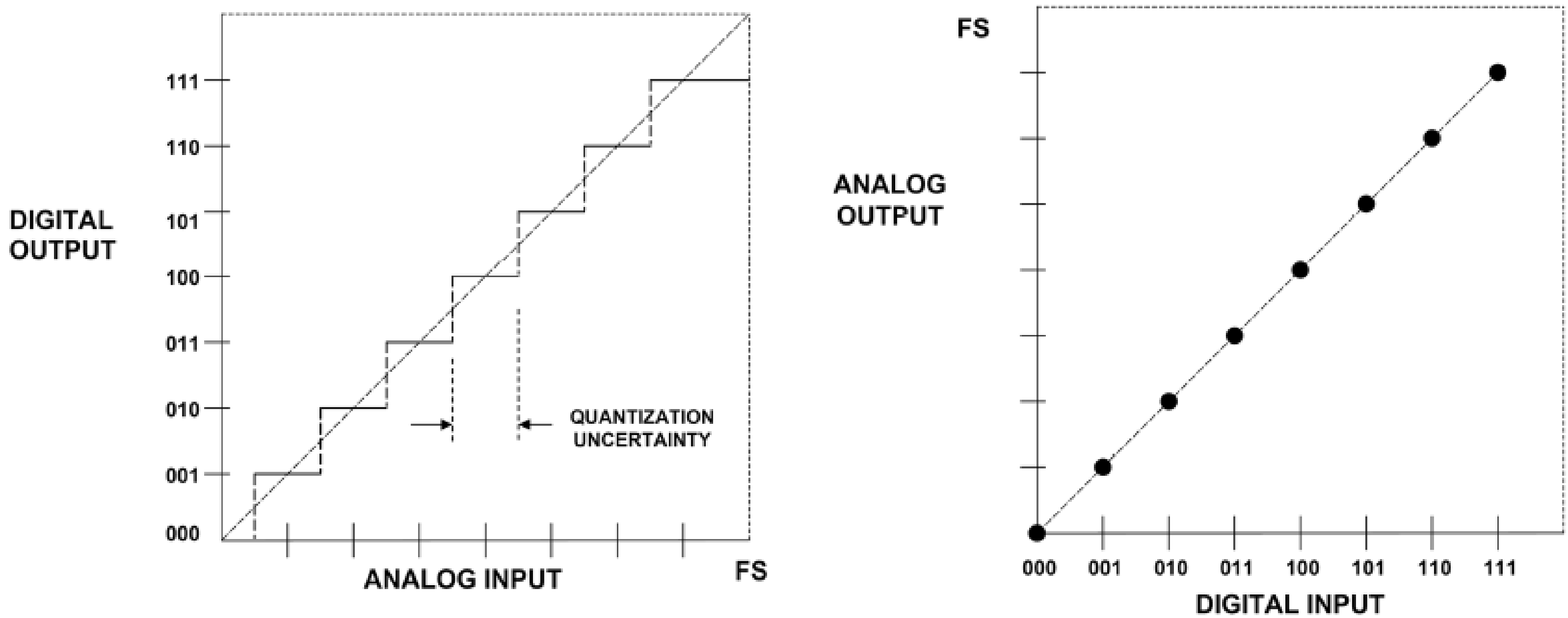
Quantization: the size of a least significant bit (LSB=Full Scale/2^N)

RESOLUTION N	2^N	VOLTAGE (10V FS)	ppm FS	% FS	dB FS
2-bit	4	2.5 V	250,000	25	-12
4-bit	16	625 mV	62,500	6.25	-24
6-bit	64	156 mV	15,625	1.56	-36
8-bit	256	39.1 mV	3,906	0.39	-48
10-bit	1,024	9.77 mV (10 mV)	977	0.098	-60
12-bit	4,096	2.44 mV	244	0.024	-72
14-bit	16,384	610 µV	61	0.0061	-84
16-bit	65,536	153 µV	15	0.0015	-96
18-bit	262,144	38 µV	4	0.0004	-108
20-bit	1,048,576	9.54 µV (10 µV)	1	0.0001	-120
22-bit	4,194,304	2.38 µV	0.24	0.000024	-132
24-bit	16,777,216	596 nV*	0.06	0.000006	-144

*600nV is the Johnson Noise in a 10kHz BW of a 2.2kΩ Resistor @ 25°C

- The ADC output (DAC input) is digital ⇒ the signal is quantized
- *N-bit ADC (DAC)* works with N-bit words representing one of 2^N possible analog levels (tipically voltages) each

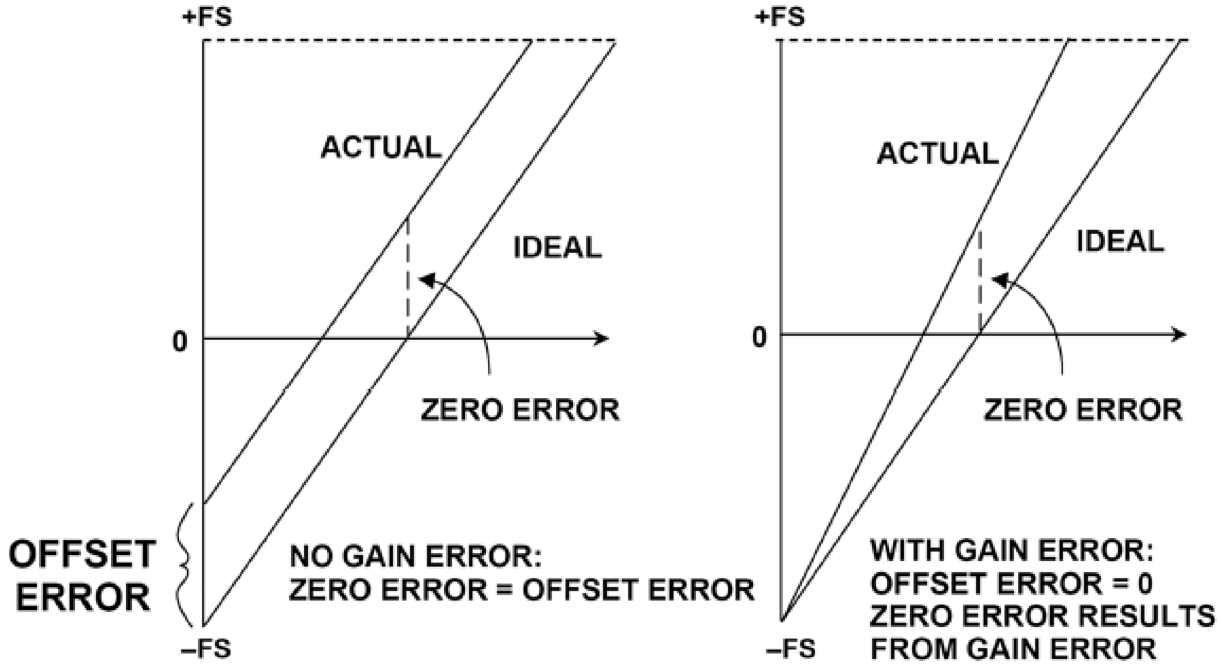
Transfer functions for ideal 3-bit unipolar converters



- $2^N|_{N=3} = 8$ possible levels
- the transfer characteristic is not a line but a number of discrete points

DC errors in data converters

- Integral non-linearity, differential non-linearity
- **Offset-error, zero-error, gain error:**

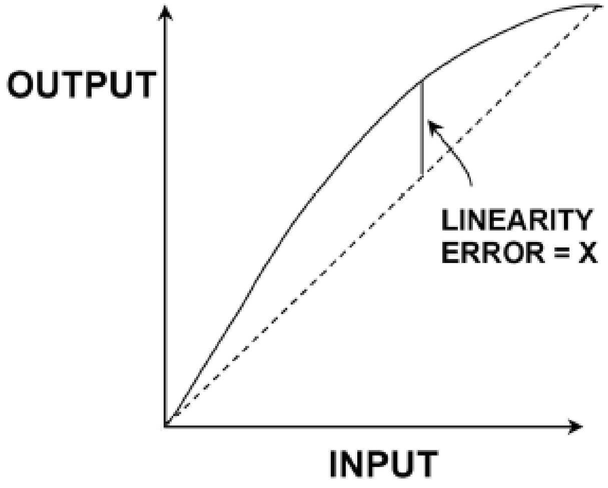


- These errors can usually be trimmed by the data converter user
- Amplifier offset is trimmed at zero input, and then the gain is trimmed near to full scale

Integral non-linearity (INL)

- The maximum deviation of the actual transfer characteristic from the ideal transfer characteristic of the converter
- It is generally expressed as a percentage of full scale (but may be given in LSBs)

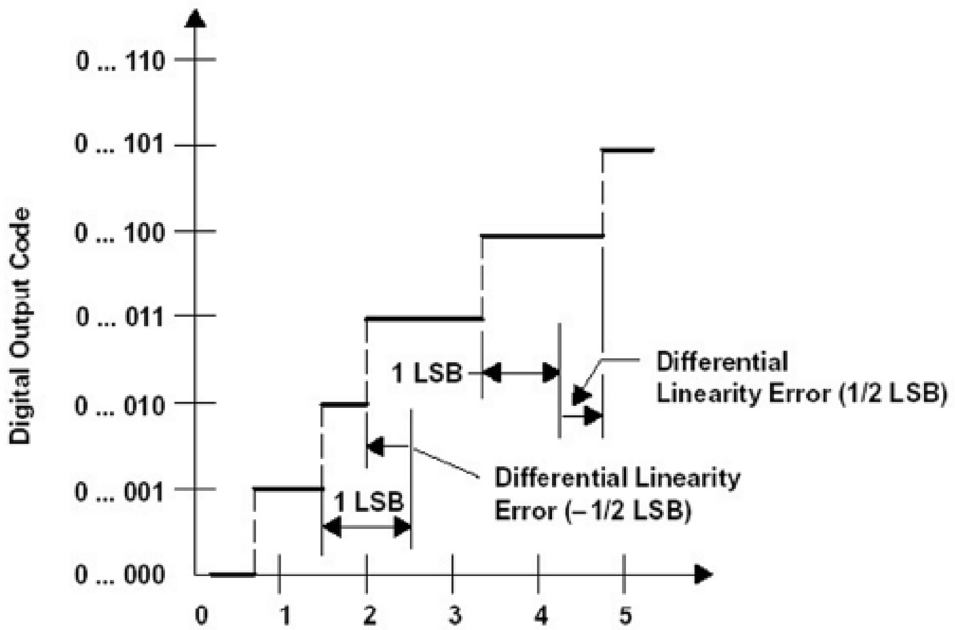
$$\text{INL}=\max\{|U(\text{code})_{act} - U(\text{code})_{id}|\}$$
$$\text{code} \in \{0 \cdots 2^N\}$$



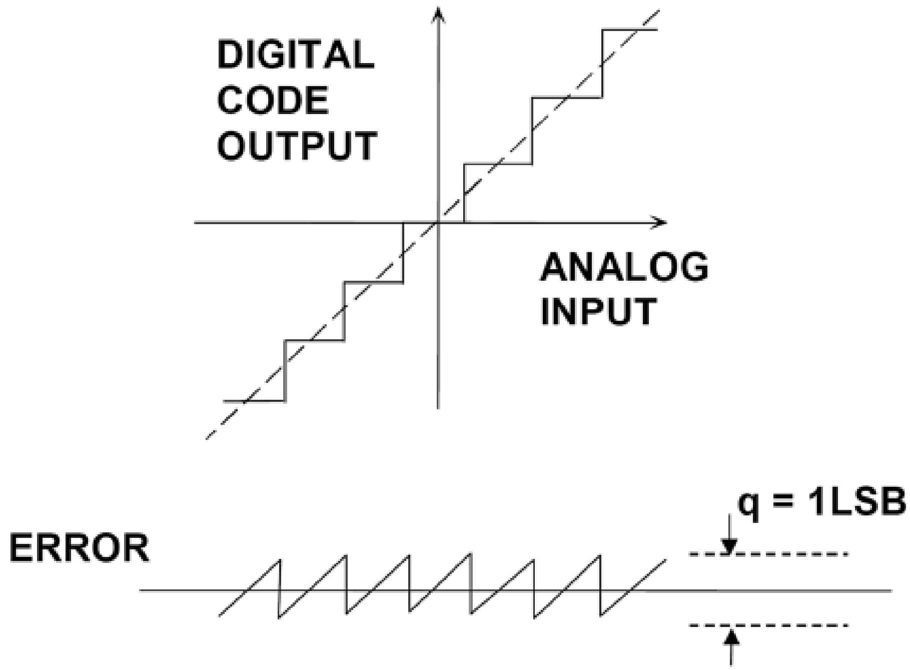
Differential non-linearity (DNL)

- DNL relates to the linearity of the code transitions of the converter
- In the ideal case, a change of 1 LSB in digital code corresponds to a change of exactly 1 LSB of analog signal
- DNL$=\max\{|U(\text{code}+1)_{act} - U(\text{code})_{act}| - LSB\}$ code $\in \{0 \cdots 2^N - 1\}$

DNL indicates the deviation from the ideal 1 LSB step size of the analog input signal corresponding to a code-to-code increment



Quantization noise in an Ideal N-Bit ADC - engineering approach

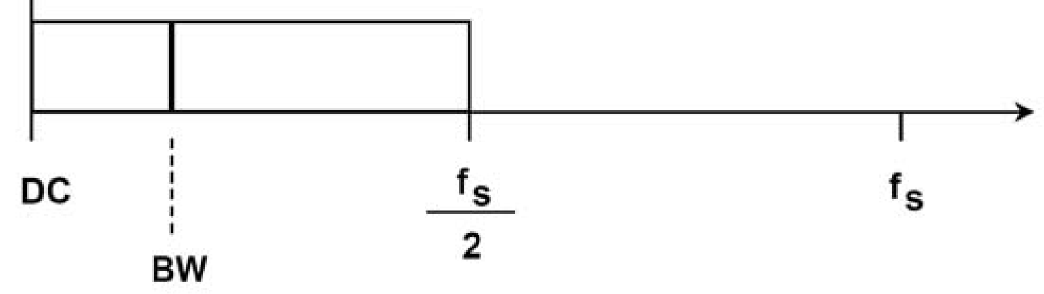


RMS ERROR = $q/\sqrt{12}$

SNR = $6.02N + 1.76dB + 10\log\left[\frac{f_S}{2 \cdot BW}\right]$ FOR FS SINEWAVE

Quantization noise in an Ideal N-Bit ADC - engineering approach

- RMS QUANTIZATION NOISE = $q/\sqrt{12}$,
  q = WEIGHT OF LSB
- SNR (FS RMS INPUT) = (6.02N + 1.76)dB, OVER $f_s/2$ BANDWIDTH
  WHERE N = NUMBER OF BITS
- ASSUME QUANTIZATION NOISE IS UNIFORMLY
  DISTRIBUTED: DC TO $f_s/2$
  (SPECIAL CASES OCCUR WHERE THIS IS NOT TRUE)

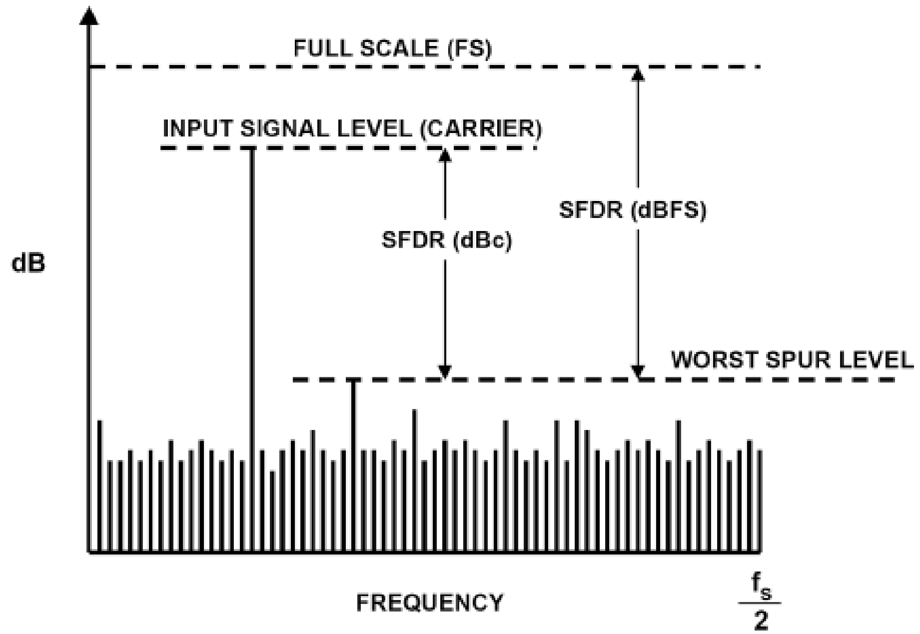


Quantization noise in an Ideal N-Bit ADC

- It can be shown that the ratio of the rms value of a full scale sinewave to the rms value of the quantization noise (expressed in dB) is:
  $SNR^{dB} = 6.02N + 1.76$
- The above equation is only valid if the noise is measured over the entire Nyquist bandwidth from DC to fs/2
- If the signal bandwidth, BW, is less than fs/2, then the SNR within the signal bandwidth BW is increased because the amount of quantization noise within the signal bandwidth is smaller:
  $SNR^{dB} = 6.02N + 1.76 + 10\log_{10}\left(\frac{f_s}{2BW}\right)$
- The above equation reflects the condition called oversampling, where the sampling frequency is higher than twice the signal bandwidth
- The last, correction term is often called processing gain
- Notice that for a given signal bandwidth, doubling the sampling frequency increases the SNR by 3dB

Dinamical properties - Spurious free dinamic range (SFDR)

SFDR is the ratio of the rms signal amplitude to the rms value of the peak spurious spectral content (measured over the entire first Nyquist zone, DC to fs/2)



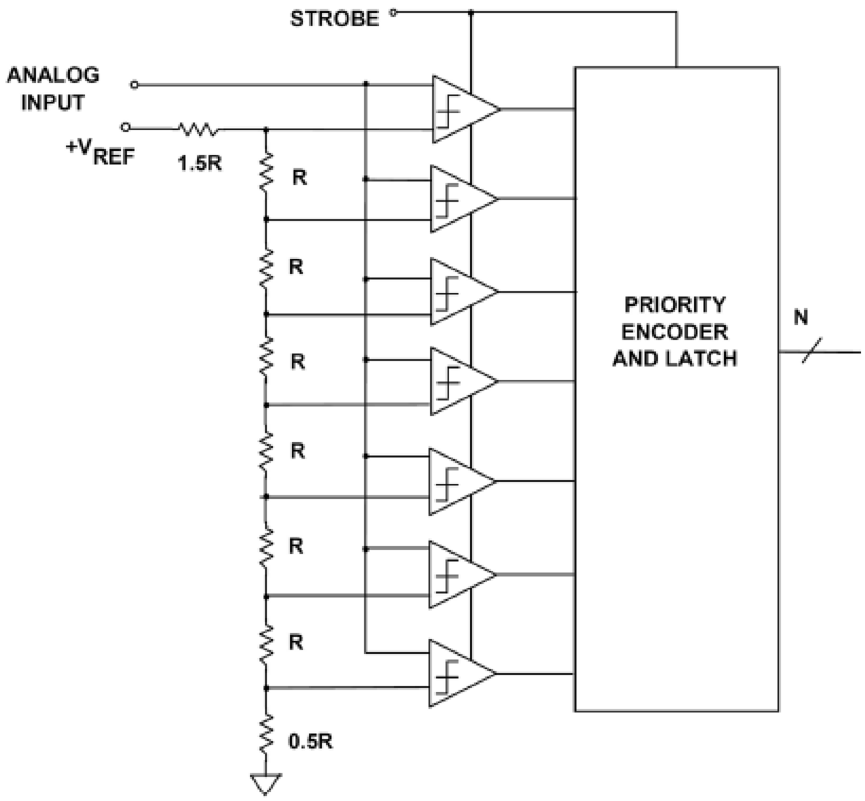
- dBc = amplitude relative to the carrier amplitude in decibel $= 20\log_{10}(A/A_{carrier})$
- dBc = amplitude relative to the full scale amplitude in decibel $= 20\log_{10}(A/A_{FS})$

Analog-to-digital conversion in a nutshell

1. Take samples in every time interval t_s from the analog signal (Sampling)
2. Quantize the discrete samples to the nearest possible level (Quantization)
3. Generate the digital code (Coding)

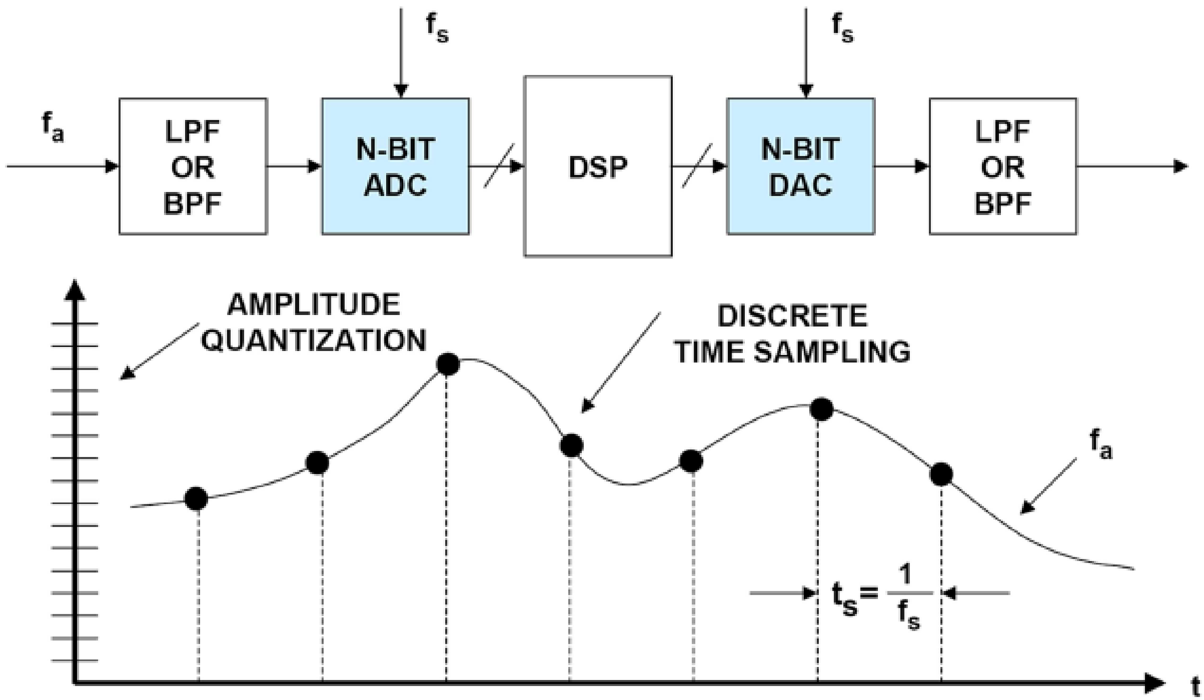
A simple ADC: Flash (Parallel) converter

- An N-bit Flash ADC consits of 2^N resistors and $2^N - 1$ comparators
- $2^N - 1$ comparator output are processed and decoded into N-bit binary output
- Amplitude resolution max. 8-10 bits (LSB= $\frac{FS}{2^N}$)
- Max. sampling rate 1 GHz
- Bandwidth 300 MHz



3.3.2 Digital-to-analog converters

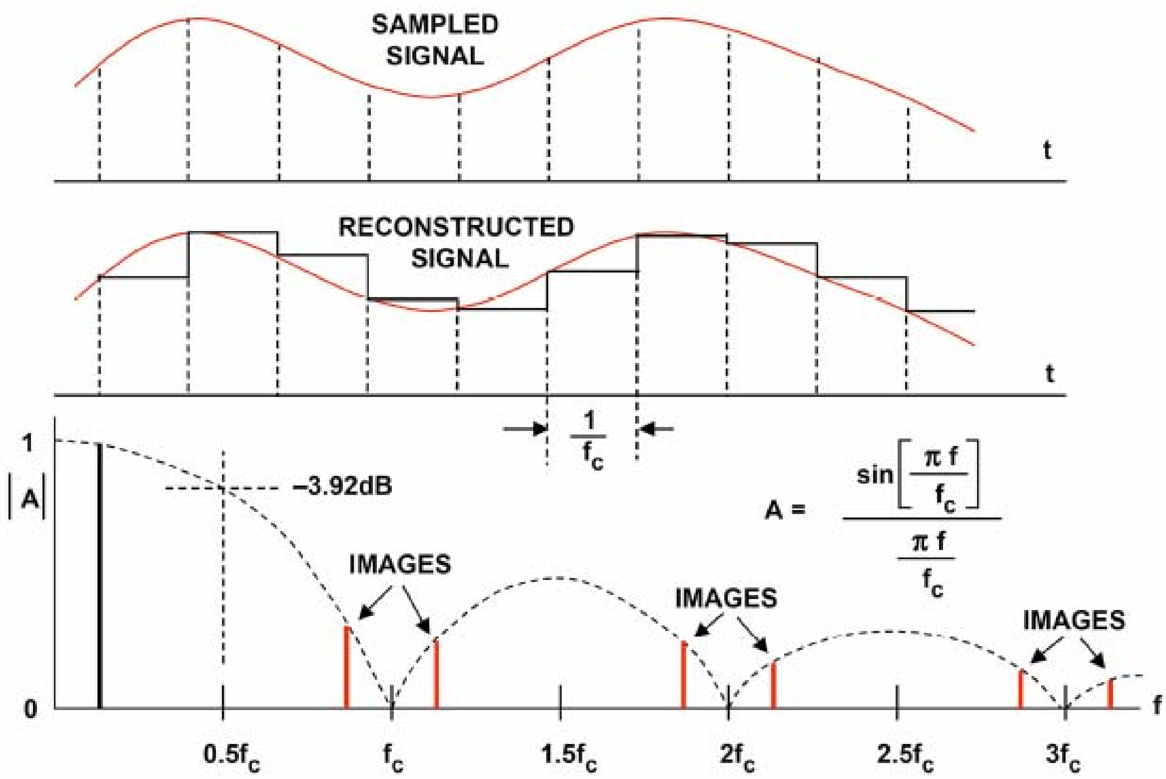
Where did we started from? Sampled data system:



How can a digital code (a series of "0"-s and "1"-s) be converted back into analog?

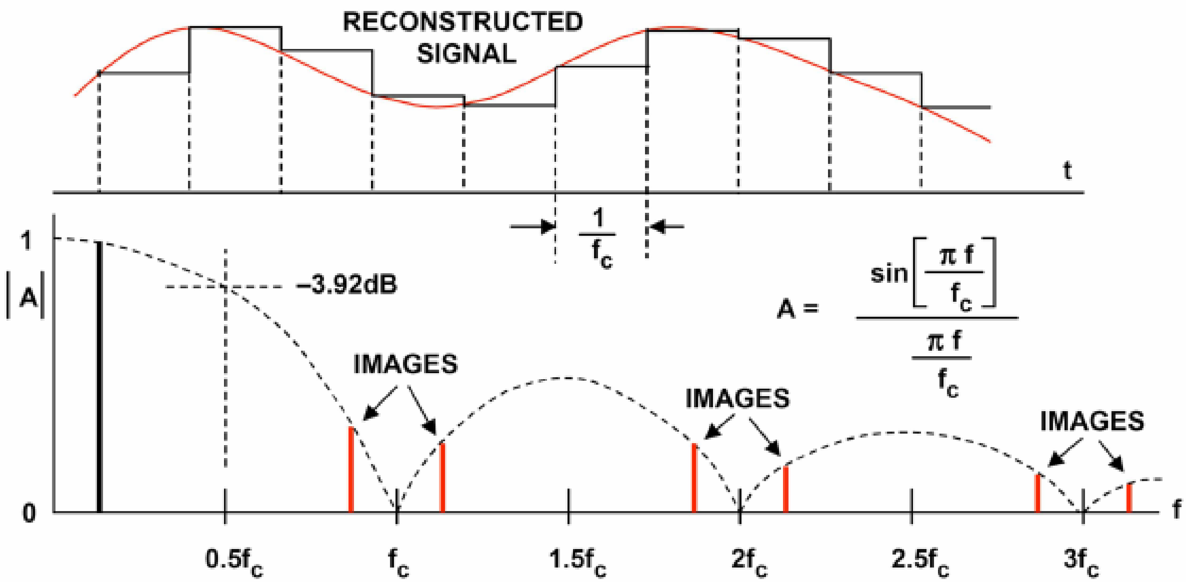
Converting codes into continuous analog signal

Digital code $\longrightarrow$ Discrete quantized levels $\longrightarrow$ Continuous signal in time and amplitude



DAC sin x/x roll off

- the output of a reconstruction DAC can be visualized as a series of rectangular pulses whose width is equal to the reciprocal of the clock rate
- the reconstructed signal amplitude is down 3.92dB at the Nyquist frequency, fc/2
- the images of the fundamental signal are also attenuated

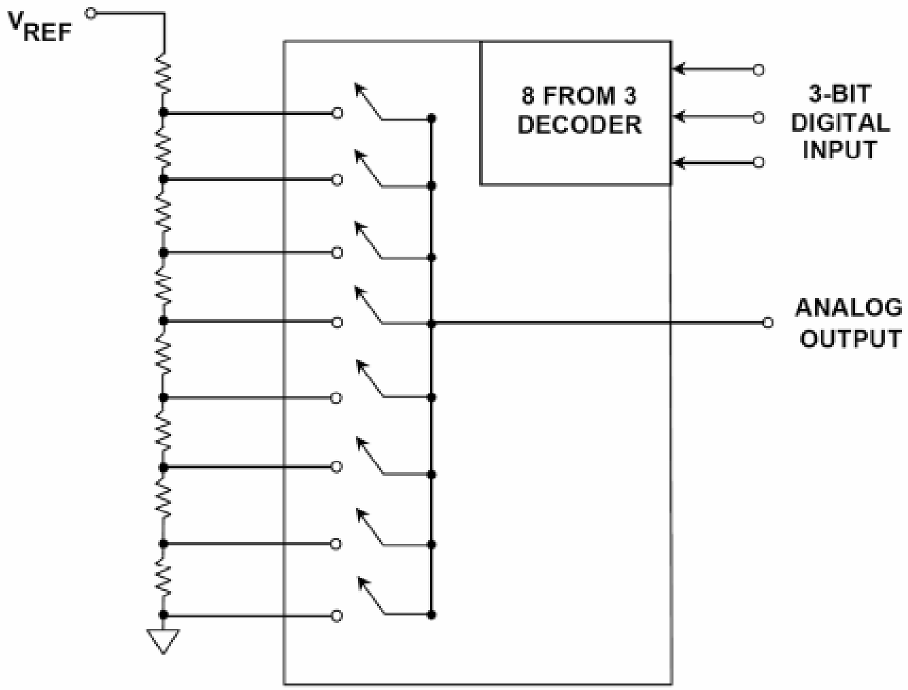


Digital-to-analog conversion in a nutshell

1. Convert the digital codes into discrete levels (Decoding)
2. Turn the series of discrete levels into time-continuous signal (ZoH)

A simple DAC: Kelvin divider ("string DAC")

- An N-bit Kelvin DAC consits of 2^N equal resistors in series
- The output is taken from the appropriate tap by closing one of the 2^N switches by decoding 1 of 2^N switches from the N-bit data
- Amplitude resolution max. 8-10 bits (LSB$=\frac{FS}{2^N}$)

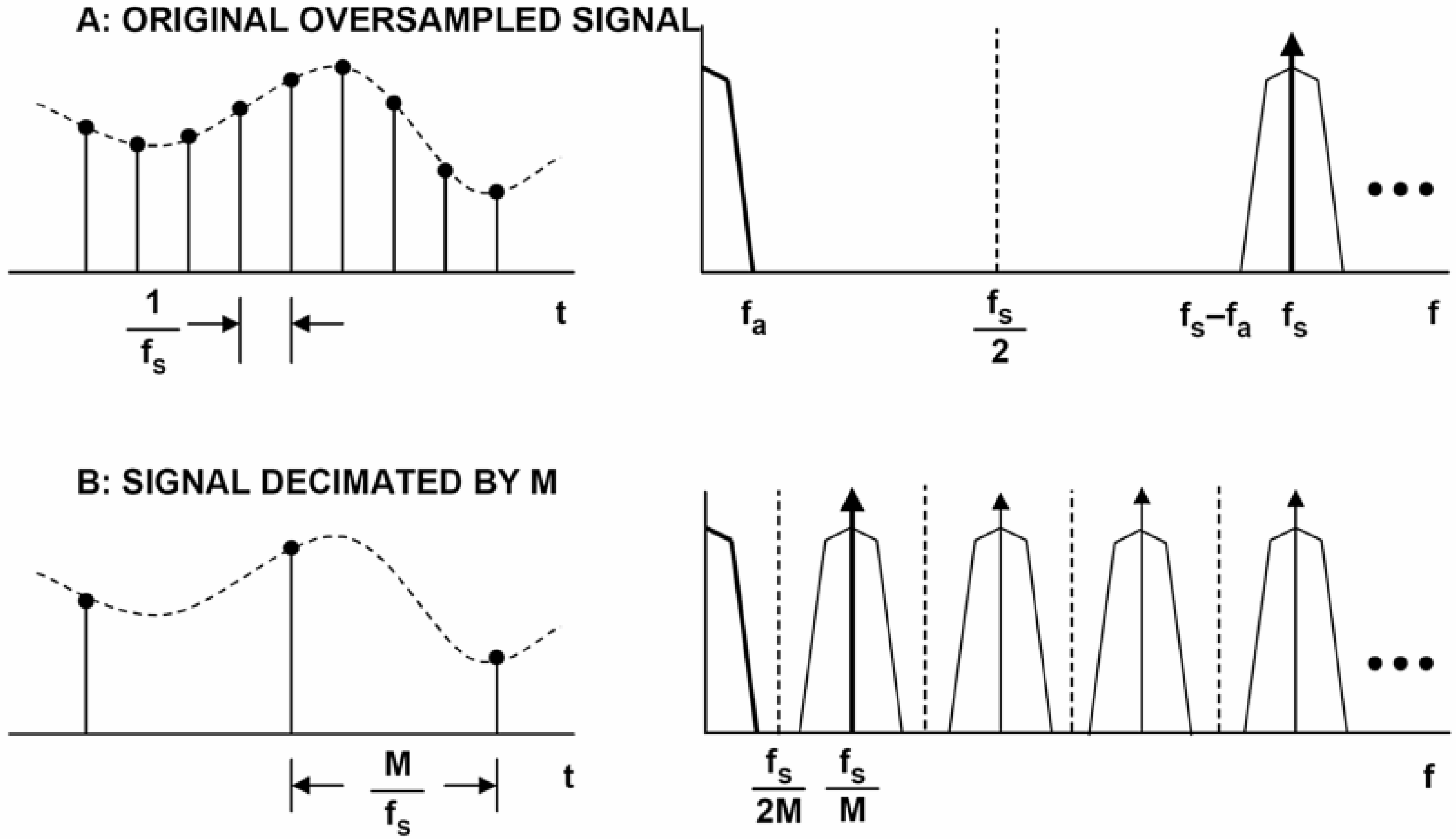


3.3.3 Decimation and interpolation - powerful tools for multirate filters

- In some cases the change of sampling rate is necessary in a sampled data system
  - change the sampling rate of the ADC/DAC
  - change the sampling rate after the signal has been digitized $\surd$
- Techniques to change the sample rate:
  - decimation $\rightarrow$ reducing the sampling rate by a factor of $M \in \mathbb{N}$
  - interpolation $\rightarrow$ increasing the sampling rate by a factor of $L \in \mathbb{N}$
- In a generalized sample-rate converter, it may be desirable to change the sampling frequency by a non-integer number

Example: converting the CD sampling frequency of 44.1 kHz to the digital audio tape (DAT) sampling rate of 48 kHz
$\Rightarrow$ interpolating by $L=160$ followed by decimation by $M=147$ accomplishes the desired result

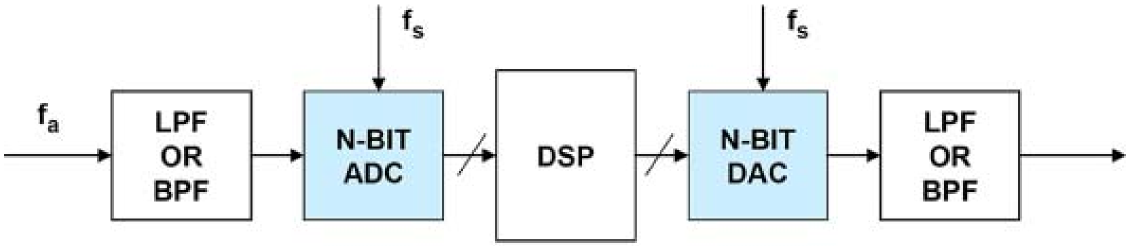
Decimation of a sampled signal by a factor of M



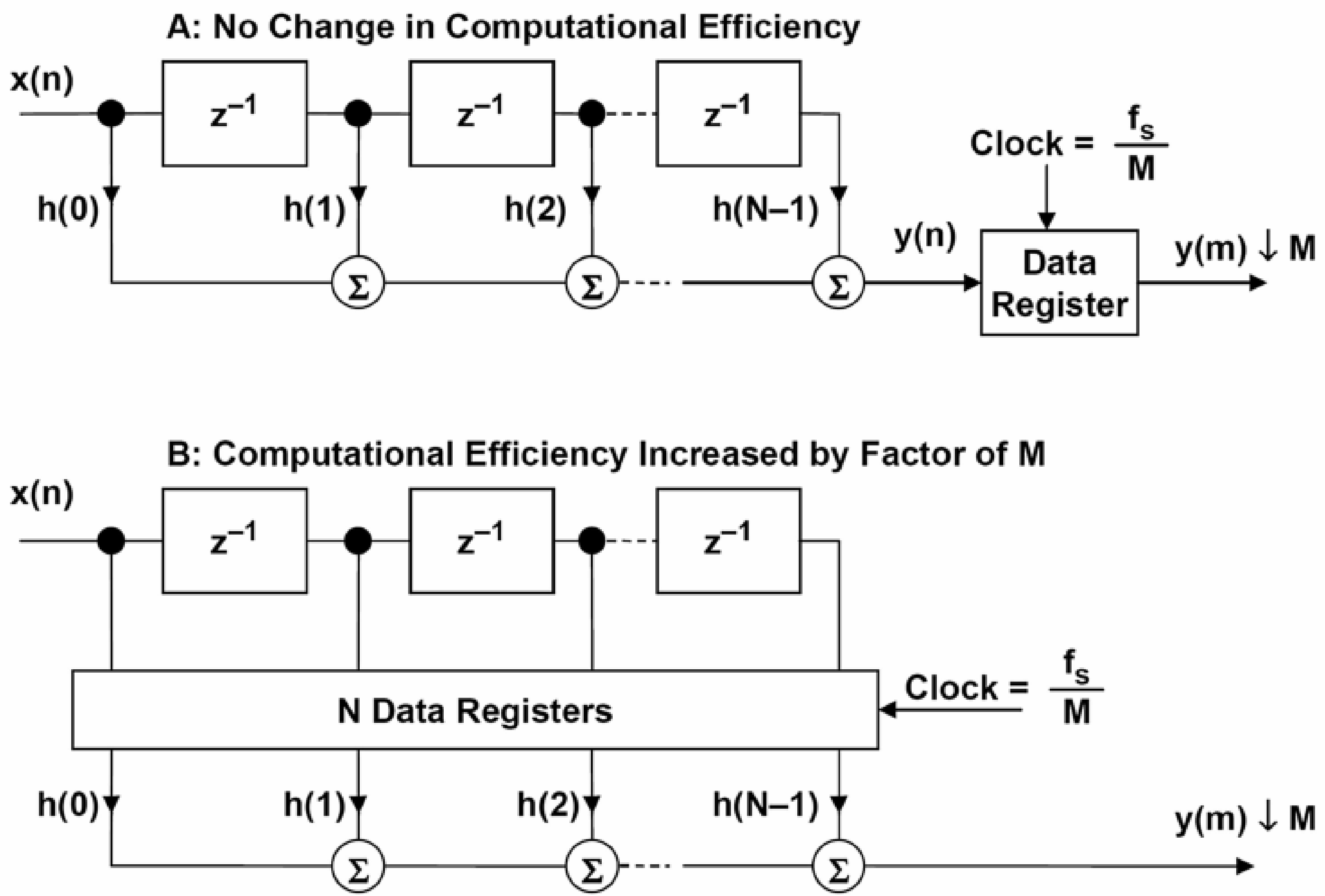
Decimation of a sampled signal by a factor of M

- **A:** original signal, f_a, sampled at a frequency f_s
  - frequency spectrum shows that f_s much higher than required
  - no information contained between f_a and $f_s - f_a$
- **B:** signal decimated by a factor of M
  - even though the sampling rate has been reduced, there is no aliasing and loss of information
  - decimation by a larger factor than M will cause aliasing

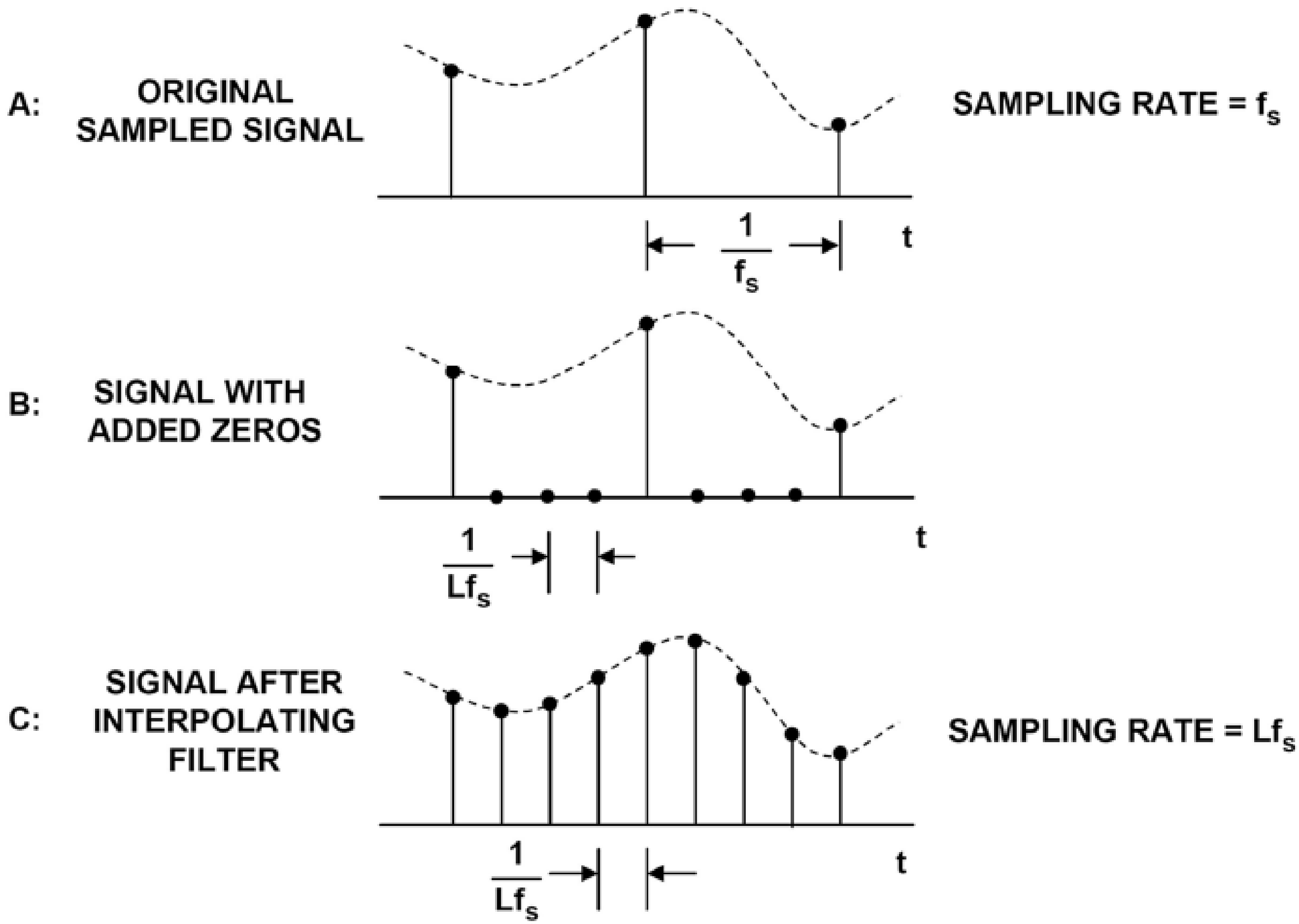
Note: Decimation and interpolation are performed in the DSP block



Decimation combined with FIR filtering



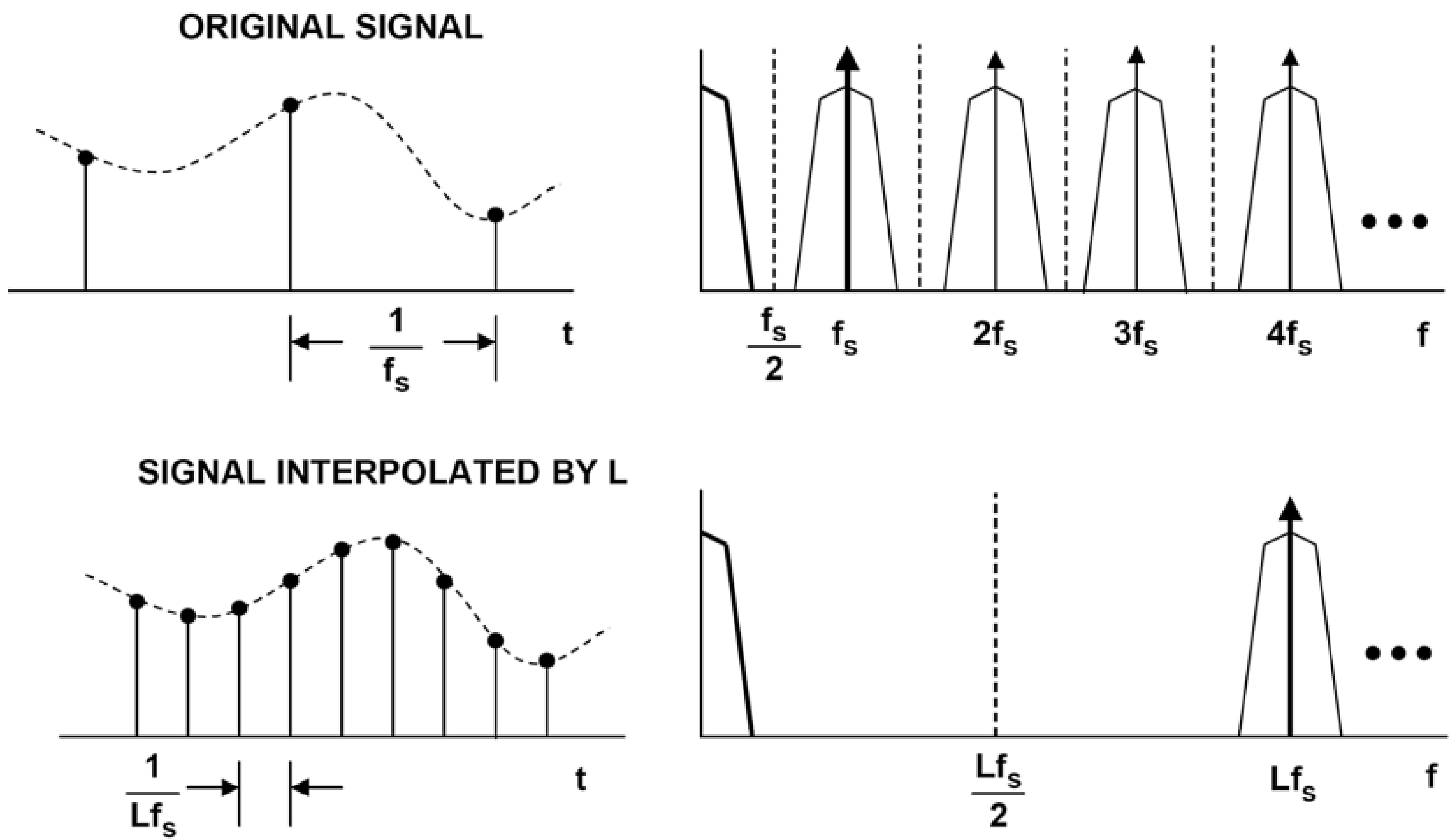
Interpolation of a sampled signal by a factor of L in the time domain



Interpolation of a sampled signal by a factor of L in the time domain

- **A:** original signal, f_a, sampled at a frequency f_s
- **B:** f_s has been increased by a factor of L, and zeros have been added to fill in the extra samples
- **C:** the signal with added zeros is passed through an interpolation filter which provides the extra data values

Effects of interpolation on the frequency spectrum



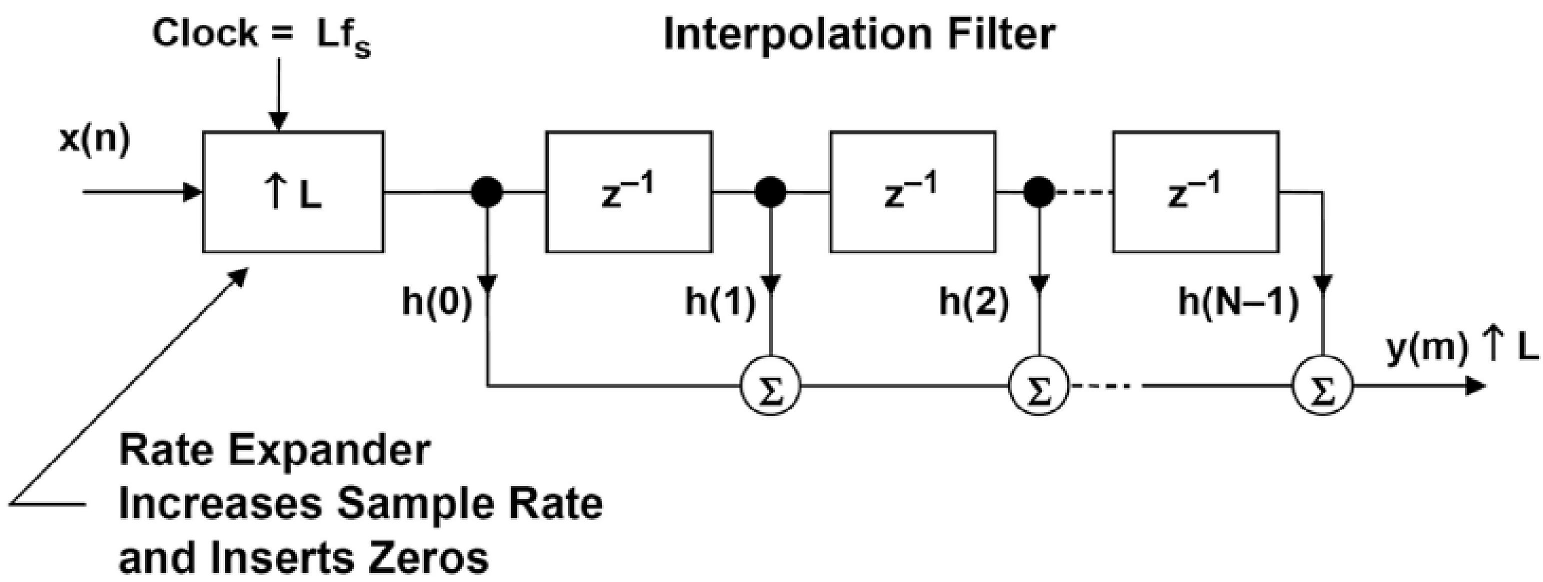
Effects of interpolation on the frequency spectrum

- **Top:** original signal, f_a, sampled at a frequency f_s
- **Bottom:** interpolated signal sampled at a frequency LF_s

Example: interpolation in the DAC of CD player (f_s =44.1 kHz) to relax the requirements of anti-imaging filter

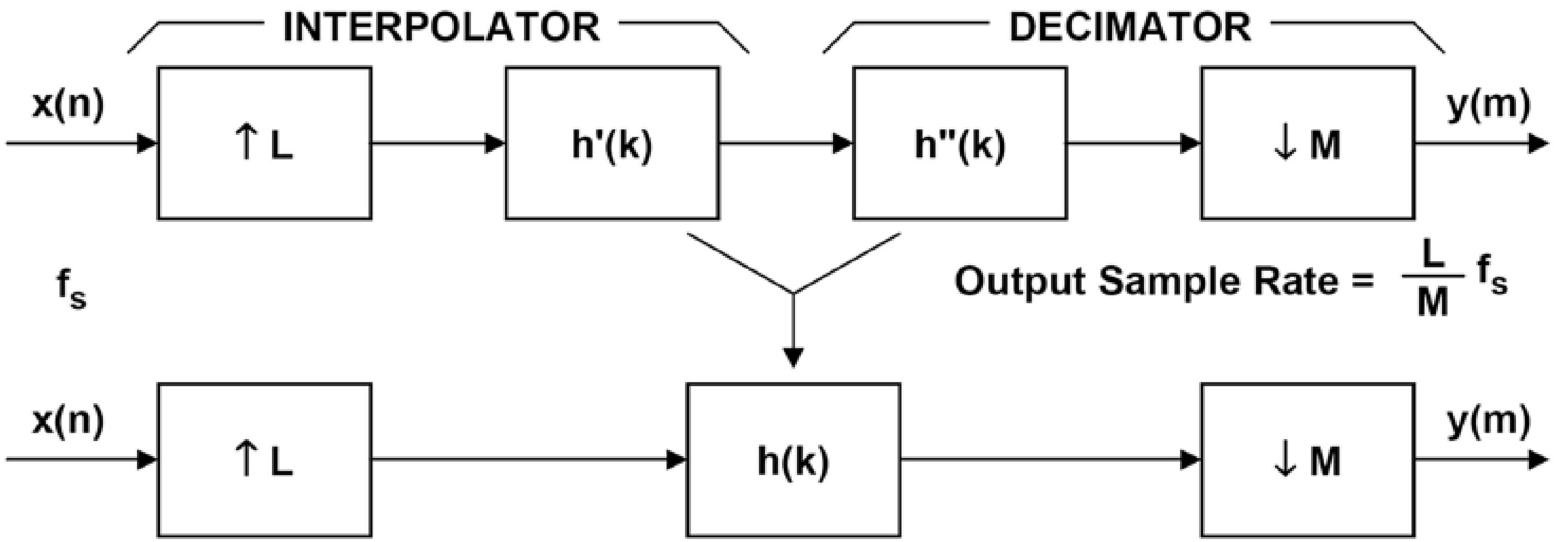
- no interpolation $\Rightarrow$ steep cut-off
- interpolation $\Rightarrow$ better spectrum separation, the filter is easier to realize, relatively linear phase, cost effective filter

Typical interpolation implementation



- the original signal x[n] is first passed through a rate expander which increases the sampling frequency by a factor of L and inserts the extra zeros
- the data then passes through an interpolation filter which smoothes the data and interpolates between the original data points
- the efficiency of this filter can be improved by using a filter algorithm which takes advantage of the fact that the zero-value input samples do not require multiply-accumulates

Multirate filter (Sample rate converter)

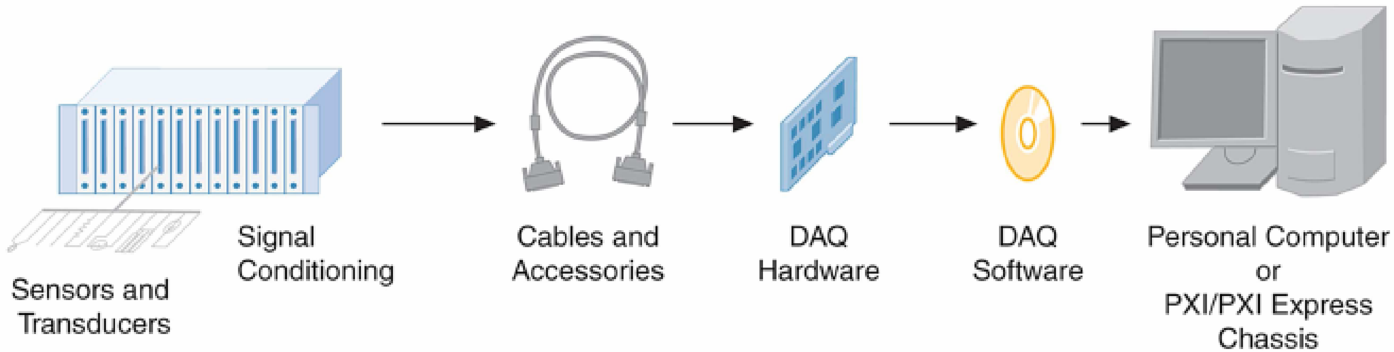


- CD f_s=44.1 kHz
  DAT f_s=48 kHz
- Use L=160, M=147
- $f_{out} = \frac{L}{M} f_s = \frac{160}{147} 44.1$ kHz $= 48$ kHz
- AD189x - family of sample rate converters

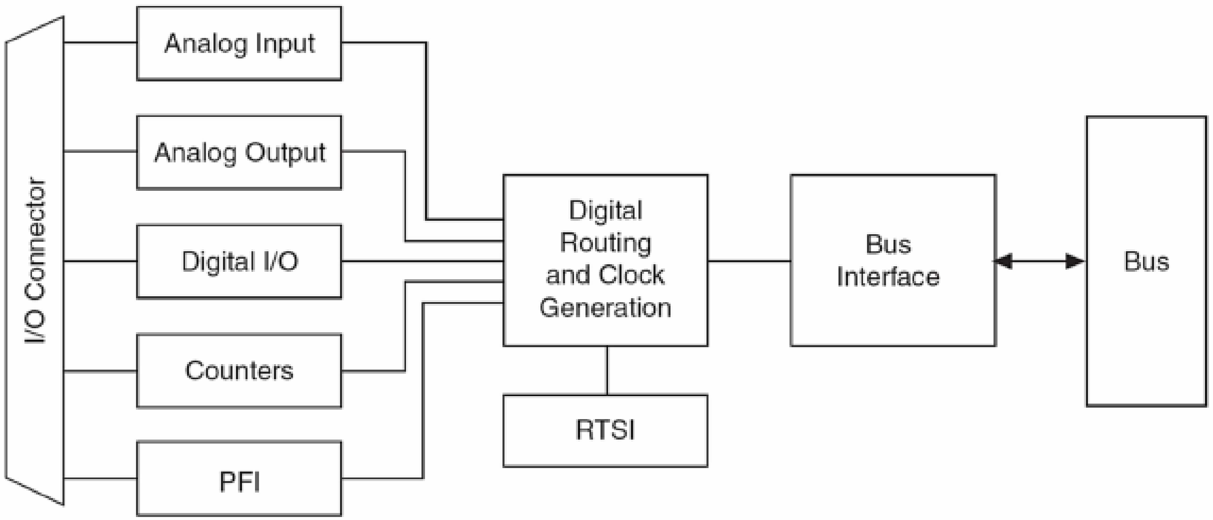
3.3.4 Computer based Data Acquisition Overview

- Traditionally, measurements are done on stand alone instruments of various types - oscilloscopes, multi meters, counters etc
- The need to *record* the measurements and *process* the collected data for visualization has become increasingly important
- There are several ways in which the data can be exchanged between instruments and a computer (serial, GPIB, LAN ports, etc.)
- Another way to measure signals and transfer the data into a computer is by using a Data Acquisition (DAQ) cards
- A typical commercial DAQ card contains ADC and DAC that allows input and output of analog and digital signals in addition to digital input/output channels

DAQ system



DAQ card block diagram



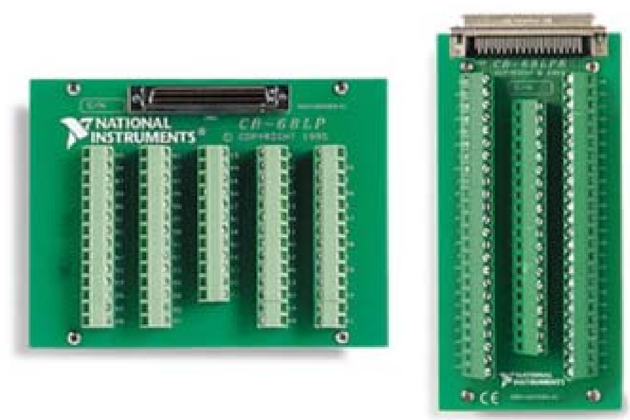
Data Acquisition Card - NI-PCI 6251 DAQ

- 1 MS/s (MegaSample/sec) $\Rightarrow 10^6$ sample /sec
- 16 bit resolution $\Rightarrow$ LSB=$U_{in_{swing}}/2^{16}$ e.g. LSB$_{FS} = \pm 10V/2^{16} = 306\mu V$
- 16 analog input
- 2 analog output

DAQ card

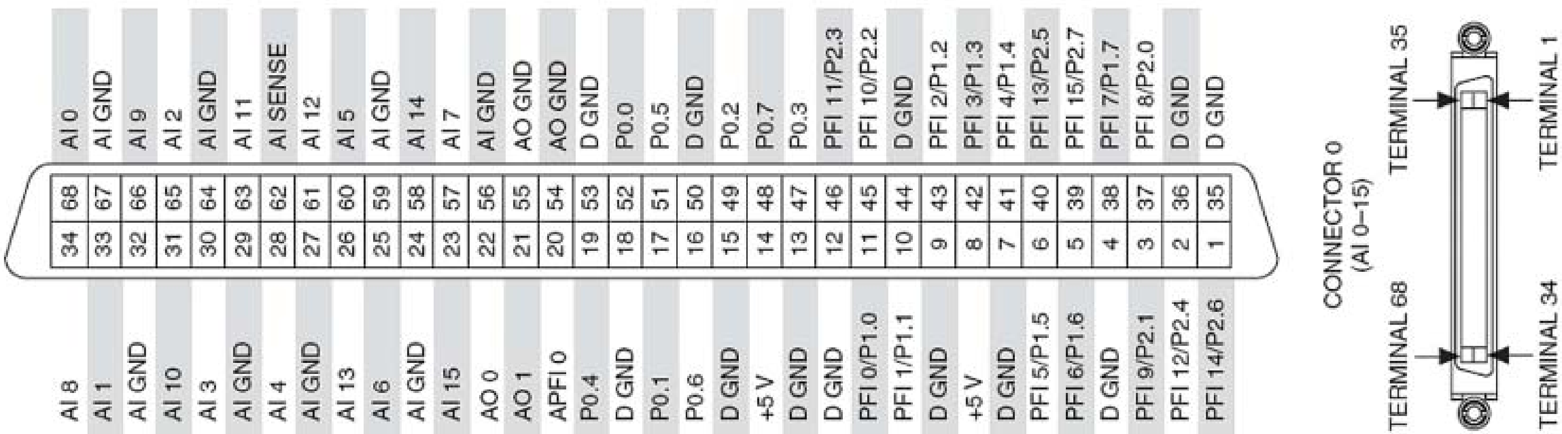
$\Longrightarrow$

Connector block



Data Acquisition Card - pinout

Use the manual of the DAQ card



References

- Analog Devices: Basic Linear Design
- Analog Devices: ADCs for DSP Application