Introduction to Signals and Sampling

S. J. Cox

s.j.cox@uea.ac.uk

Overview

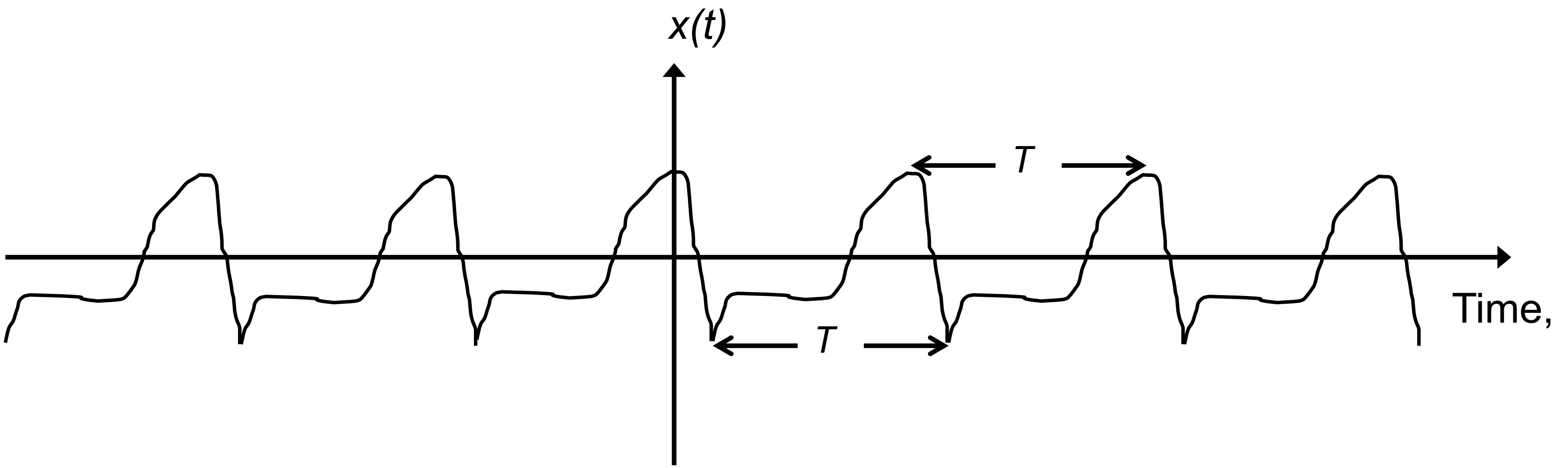
- What are signals?
- Periodic signals: the sinusoid
- Frequency domain representation of signals
- The Data Acquisition Chain
- The Sampling process
- How often do we need to Sample?
  - Aliasing and anti-aliasing filters
- The effect of quantization
- Different types of ADCs
- The MCP 3004/3008 ADC: explanation of key terms on datasheet

What are Signals?

- A signal is something that conveys information:
  - Flags, smoke-signals, bugle-calls etc.
  - Gestures
  - Text (language)
  - Audio (including speech)
  - Visual (images, video) …. etc. etc.
- Signals in the real-world are usually continuous in time and continuously valued
  - Sound is a continuous variation of air pressure and the pressure can take on an infinite number of values
  - A 2-d visual scene is a variation of colour with time on a plane. The colour can take on an infinite number of hues with an infinite number of intensities
- To convey information, a signal must vary. It can do this in a single dimension (e.g. audio = pressure variation) or in multiple dimensions (e.g. video = variation in intensity and colour in x and y directions)

Periodic Signals

- Periodic signals occur very frequently in the real world
- Below is an example of a 1-d periodic signal

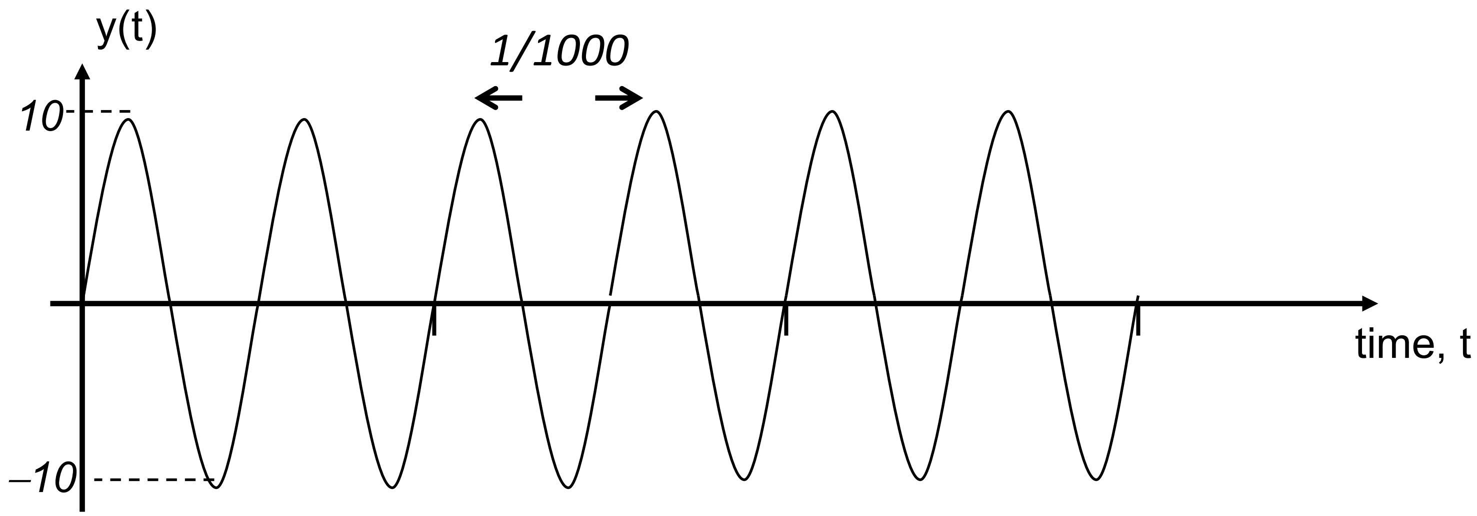


- The signal repeats every *T* seconds, and we say that the **period** of this signal is *T* seconds
- Note that *T* can be measured between *any two identical points in the cycle of the signal*
- Defining equation of a periodic signal:

$$x(t + nT) = x(t) \qquad \text{where } n \text{ is integer}$$

Sinusoidal Signals I

- The sinusoidal signal is a periodic signal that has a special status in signal analysis. Here is an example of a sinusoidal signal



- In general, the equation of a sinusoidal signal is given by:

$$y(t) = A \sin(2\pi f t + \phi)$$

- In this case: $A = 10$ units = Amplitude

  $T = 1/1000$ seconds = Period

  $f = 1/T = 1000$ Hertz = Frequency

  $\phi = 0$ radians = Phase

Sinusoidal Signals II

- An infinitely long sine wave contains **a single frequency**
- For sine waves whose frequency lies within the limits of human hearing (20 Hertz — 20000 Hertz):
  - They are perceived as a “pure tone” by the ear
  - *Amplitude* is correlated with *loudness* (larger amplitude = louder sound)
  - *Frequency* = 1/Period is correlated with *pitch* (higher frequency = higher pitch). Frequency is measured in *Hertz* (Hz)

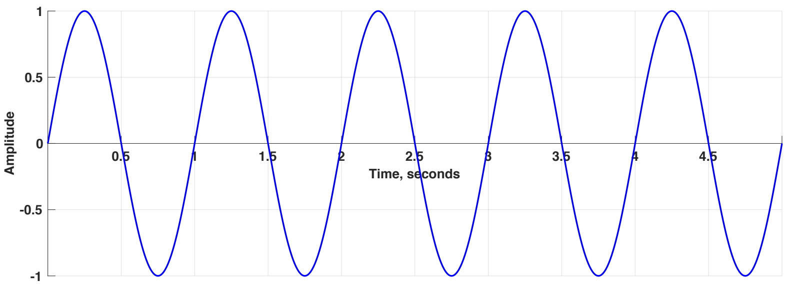
440 Hz

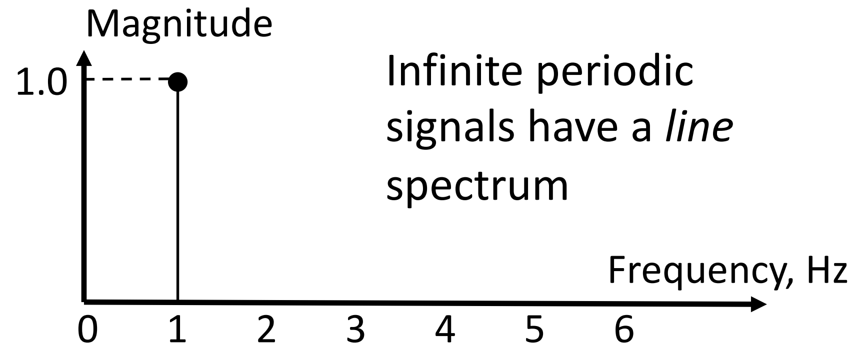
440 Hz
Lower amplitude

880 Hz

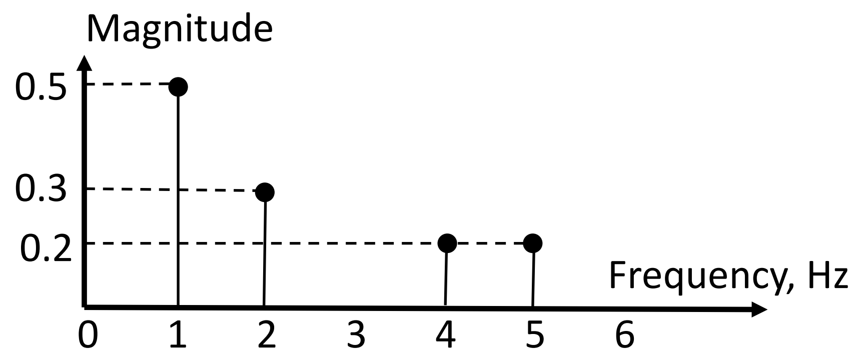
- Sinusoids possess a remarkable property: *any* periodic signal, no matter how complex, can be decomposed into a set of sinusoids (Fourier Series)
  - This property can be extended to non-periodic signals using the Fourier Transform

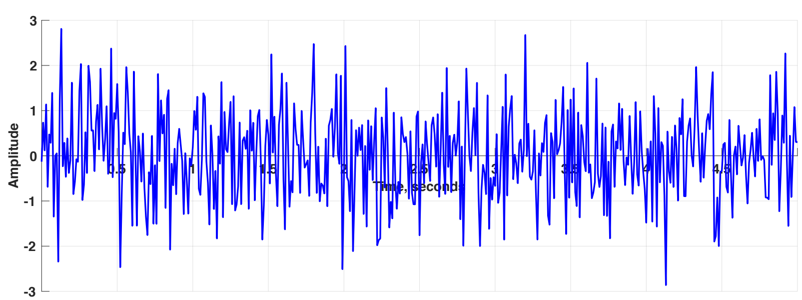
Representing a Signal in the Frequency Domain as a Spectrum

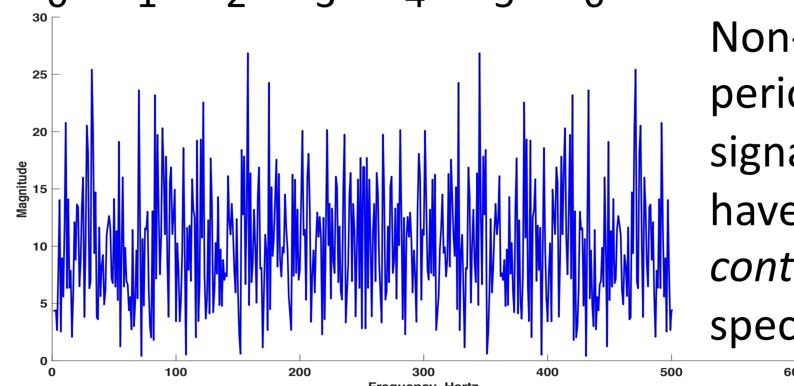




Infinite periodic signals have a *line* spectrum







Non-periodic signals have a *continuous* spectrum

Time domain

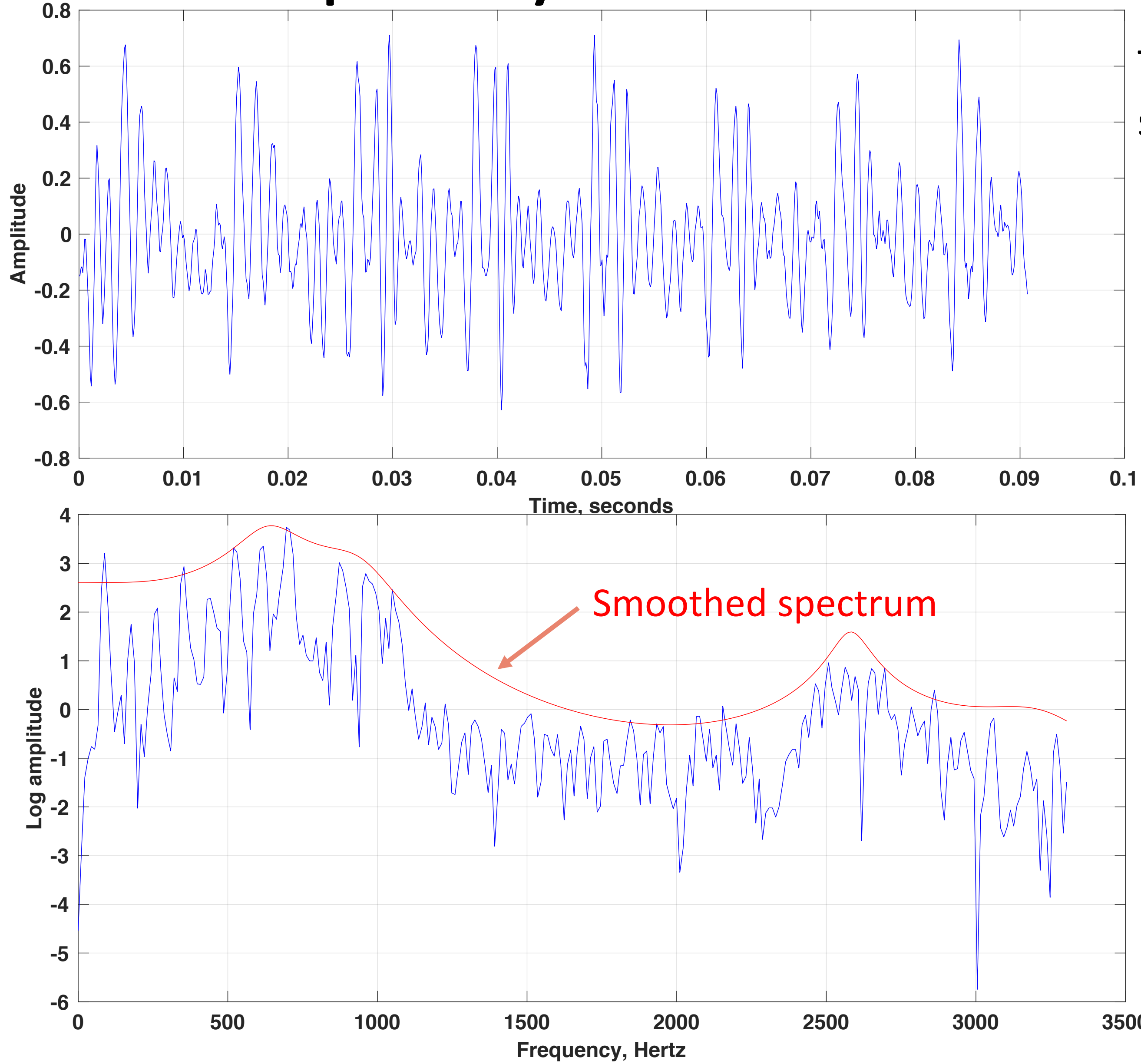
Frequency domain (spectrum)

Visualizing a Speech Sound in the Frequency Domain

Time domain

Frequency domain (spectrum)

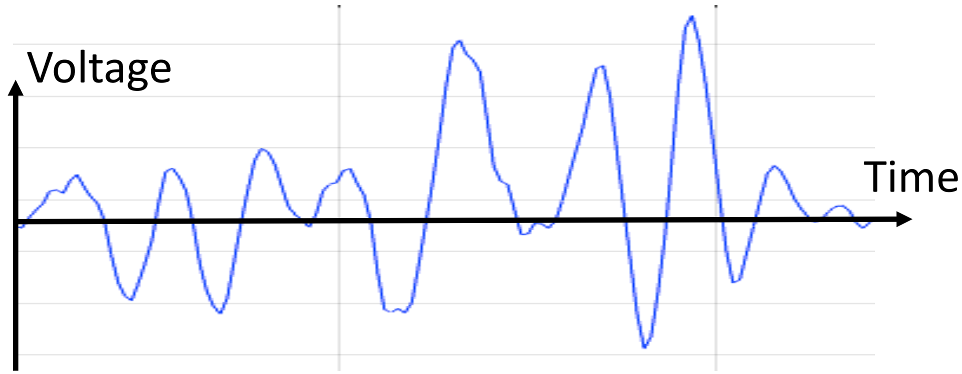
The vowel sound “ar”:



In many cases, visualizing in the frequency domain gives more insight into the signal

Sampling and Quantization

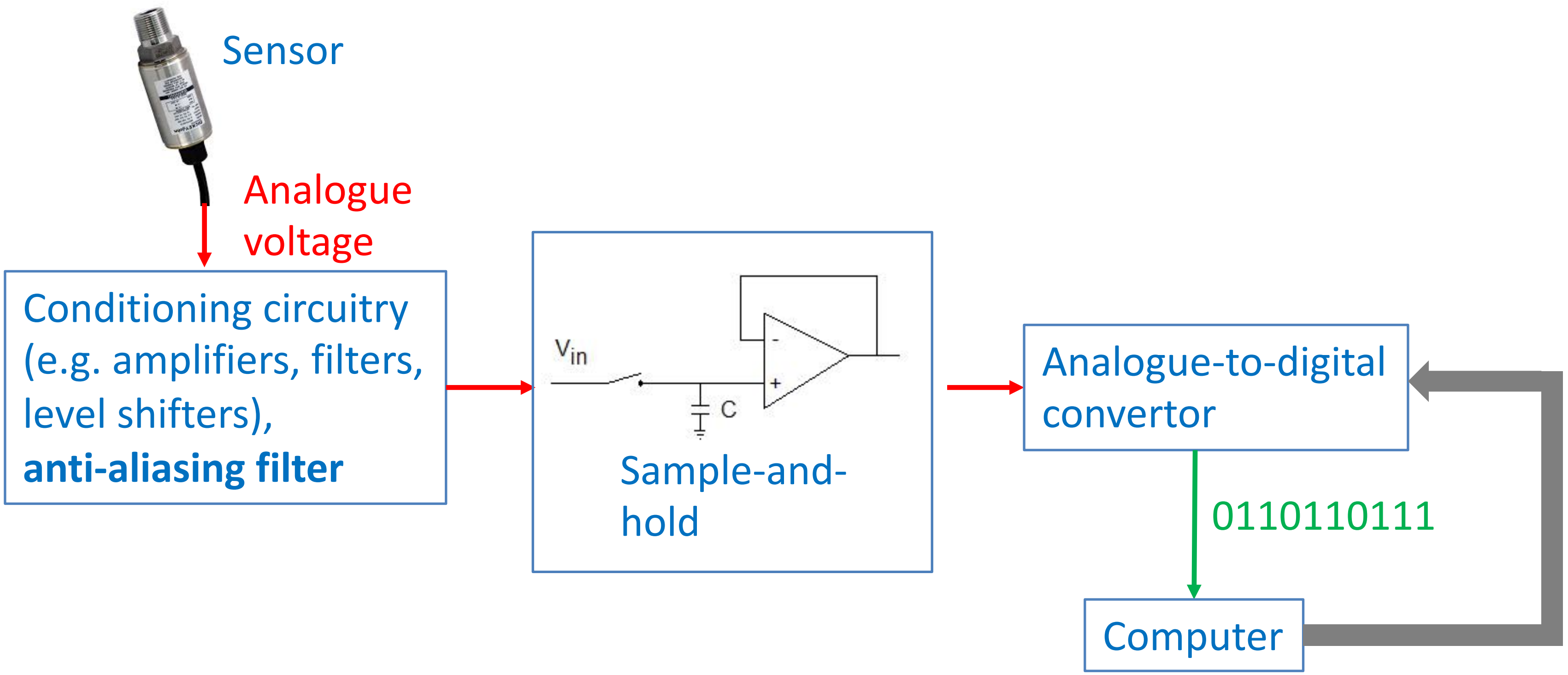
An analogue signal is **continuous in time** and can take on an infinite number of values i.e. it is also **continuous in amplitude**.



To represent it digitally:

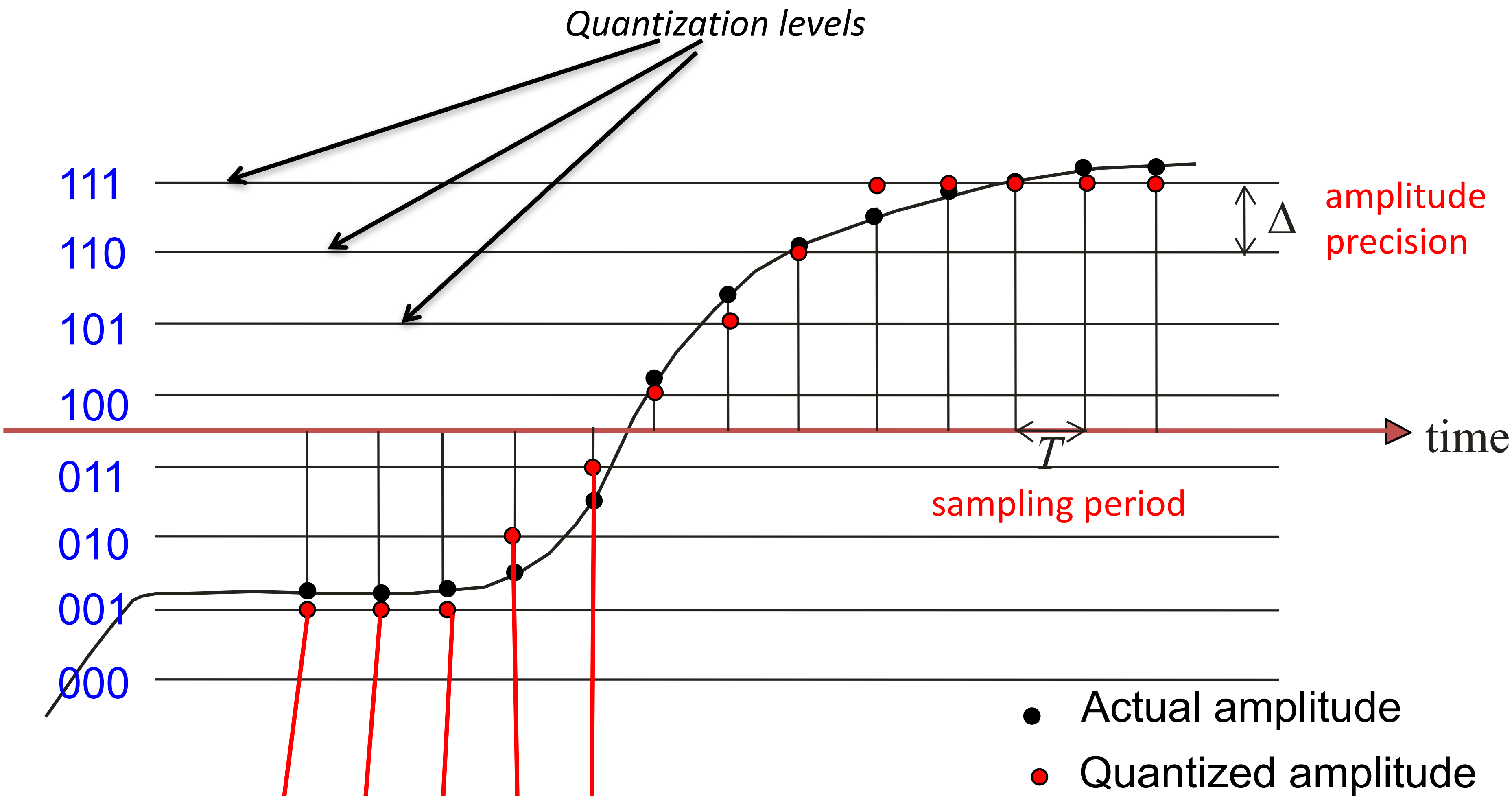
- Take a sample every *T* seconds. The more samples per second we take, the more accuracy we have, but the more storage cost/transmission cost we incur.
- The sample we have taken requires infinite precision, which we cannot handle on a computer. We must represent it as one of a finite number of *N* *quantization levels*.
  - Again, the more levels we have, the higher the accuracy, but the more overhead is incurred.

Technology of the Data Acquisition Chain



In many cases, ADCs incorporate an “on-board” sample-and-hold circuit

Sampling and Quantization



Output sequence: 001 001 001 010 011…

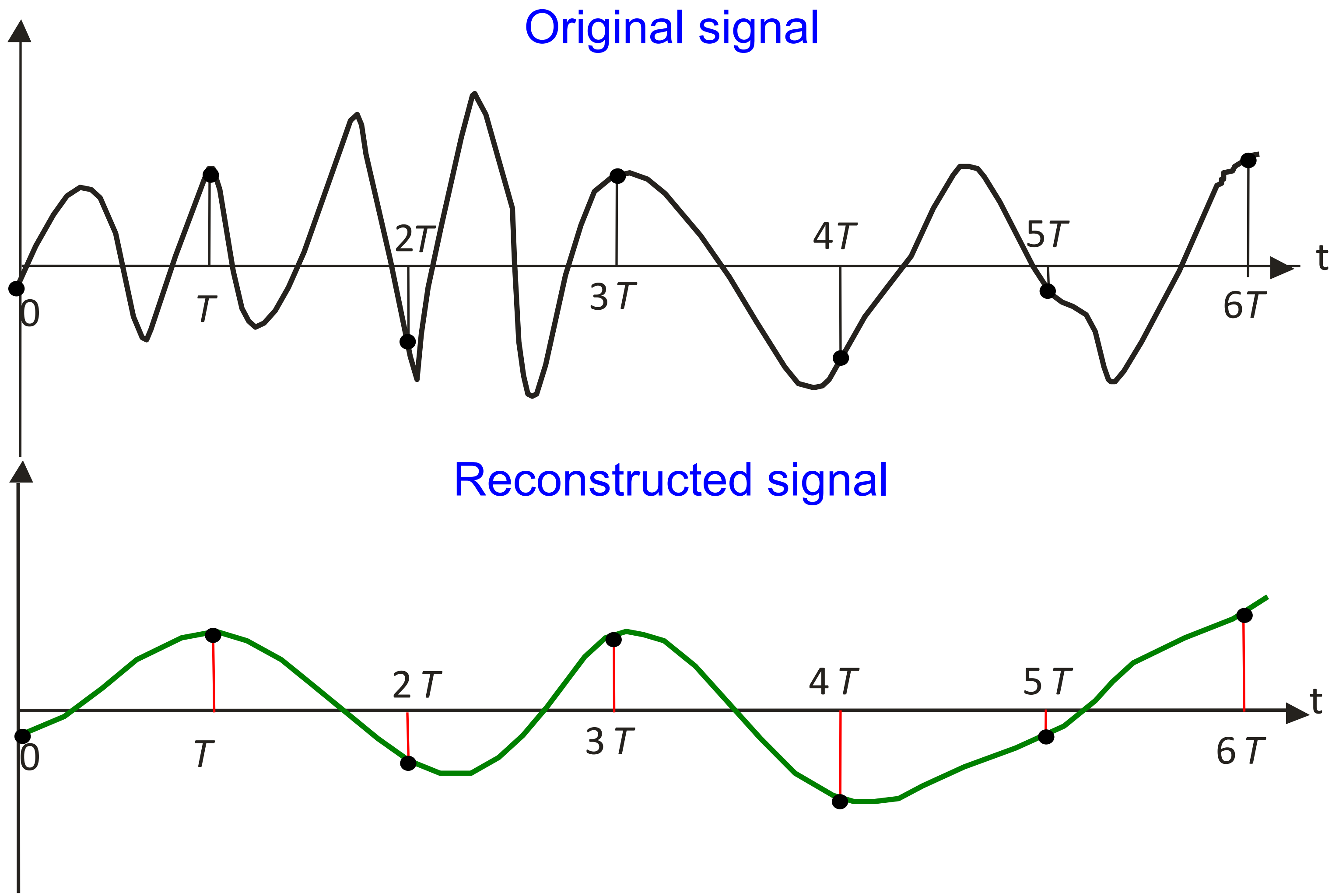
- This kind of representation of a waveform signal is known as *Pulse-code modulation* (PCM)
- It is carried out by an *analogue-to-digital convertor*

Points about Sampling and Quantization

- This kind of representation of a waveform signal is known as *pulse-code modulation* (PCM)
- It is carried out by an *analogue-to-digital convertor* (ADC)
- Sampling rates vary widely, from ~14 MHz for video to a few samples per second or even per hour for processes that change very slowly.
  - Choosing your sampling rate is a crucial decision for how much data you have to store and process
  - **But it depends on the frequency content of your data**! (See next slides)
- An ADC which represents each sample by B bits has $N = 2^B$ quantization levels
  - e.g. in last slide, $N = 8$ and $B = 3$
- 24 bits per sample (giving 16 777 216 quantization levels) is used in the highest quality audio nowadays.
  - Choosing your number of bits per sample is also crucial for storage/processing

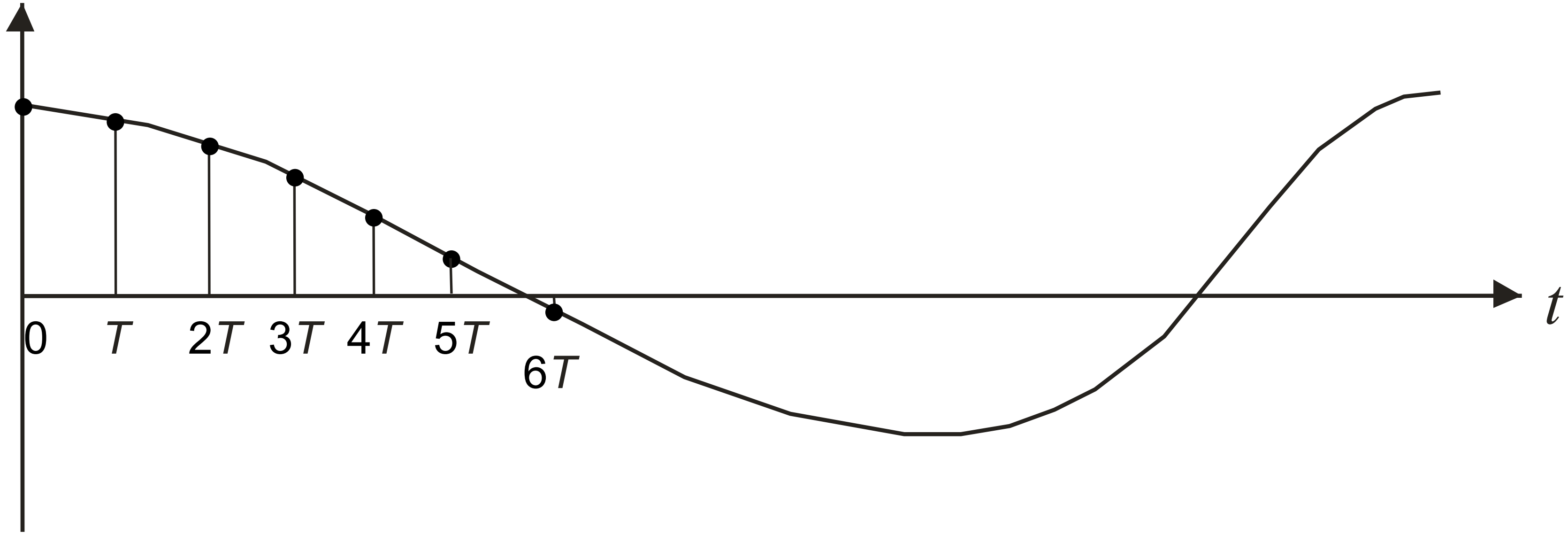
Sampling Rate I

How often do we need to take a sample?



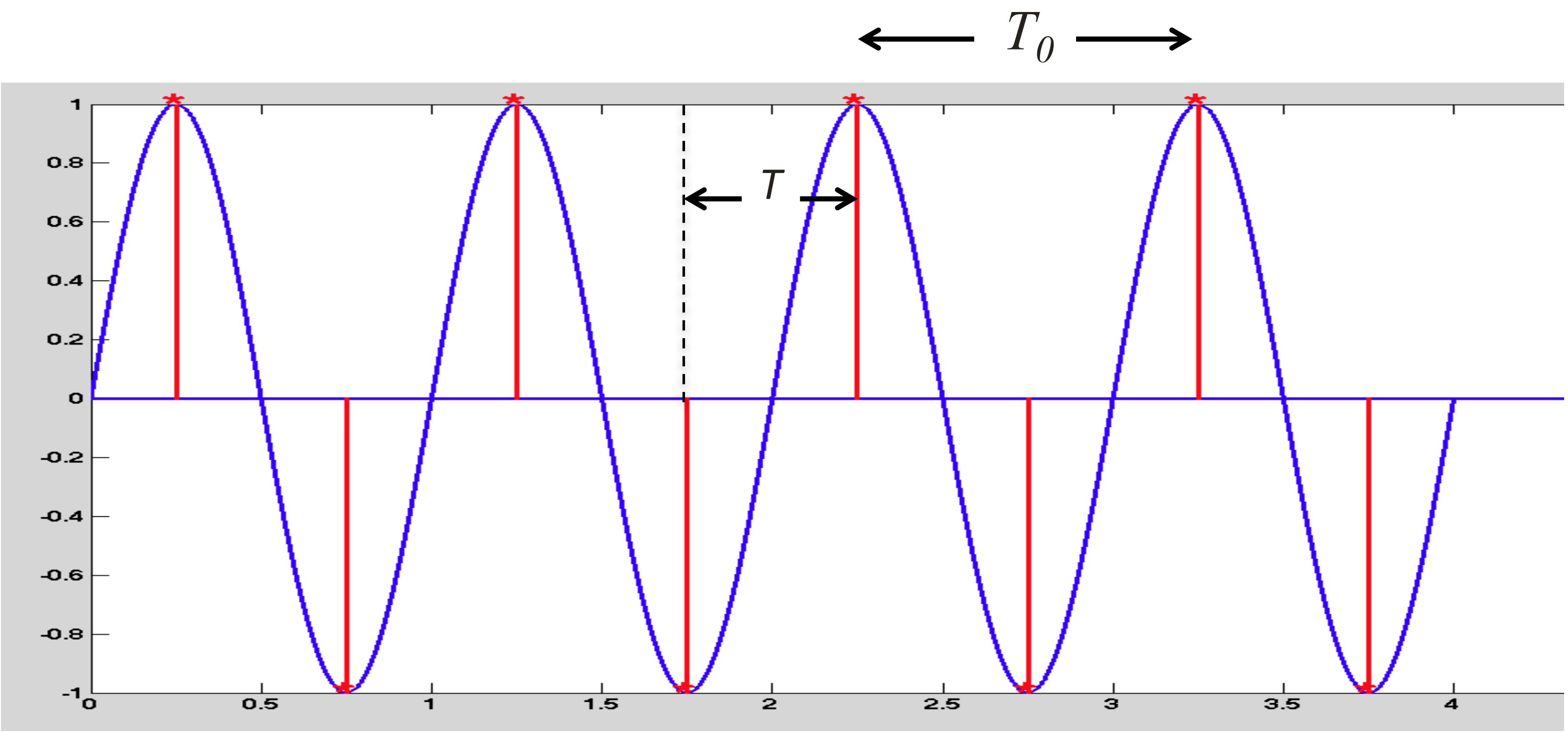
Undersampled: Insufficient detail. Signal is grossly distorted

Sampling Rate II



Oversampled: unnecessary detail, which wastes storage space and increases processing and data transfer time.

Sampling Rate III



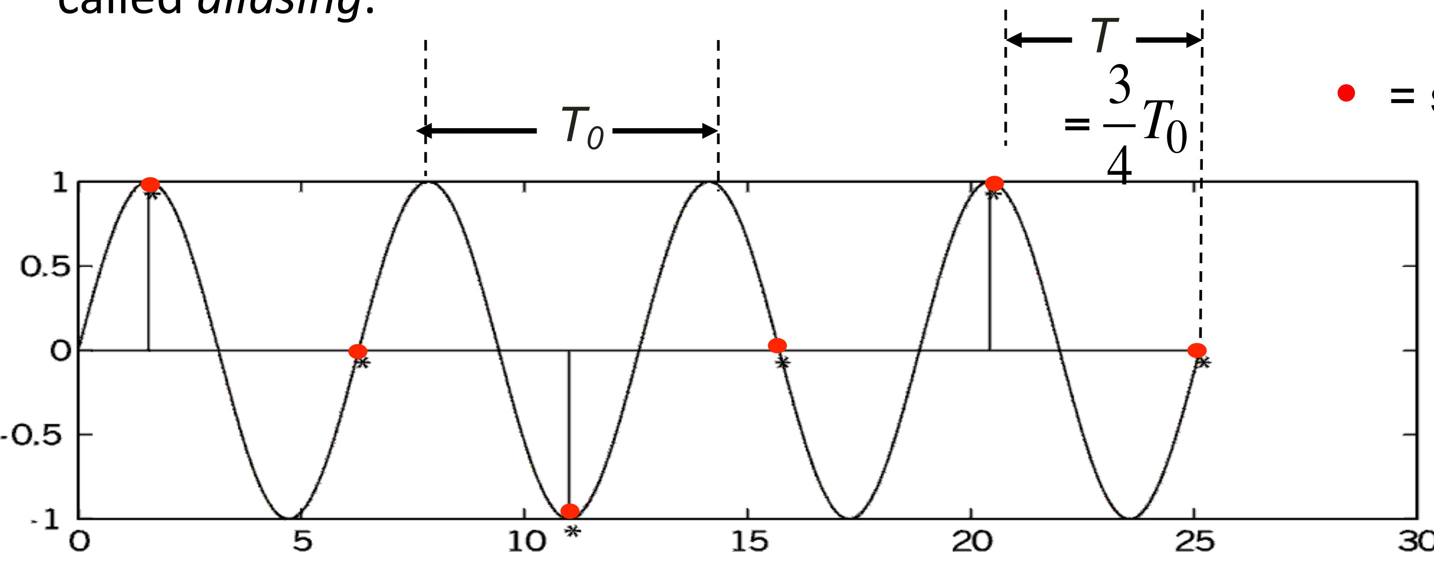
- It turns out we can capture a sine-wave with only two samples per period (an absolute minimum).
- Suppose the blue sine-wave above is the *highest* frequency wave contained in a signal, and this sine wave has a period of T_0 seconds ($f_0 = 1/T_0$ Hz.)
- Then the sampling period T must be *at most* $T_0 / 2$ seconds i.e. the sampling frequency f_s must be *at least* $2f_0$ Hz.
  - E.g. if $f_0 = 5000$ Hz, then f_s must be *at least* 10000 Hz i.e. a sample every 1/10000 *s*

The Nyquist Sampling Theorem

- Sampling theorem states : "When sampling a signal, the sampling frequency (f_s) must be greater than twice the signal's bandwidth in order to be able to reconstruct it perfectly from the sampled version."
- By *bandwidth*, we generally mean the highest frequency contained in the signal (remember: Fourier analysis tells us that any signal can be broken down into a set of sinusoids of different frequencies and phases)
- Example: if the highest frequency in a signal is 10 kHz, we must sample at least at 20 kHz, i.e. $f_s \geq 20$ kHz.
- This is known as the Nyquist Frequency.

Aliasing

If we don't take enough samples per second, we can encounter a phenomenon called *aliasing*.



● = sample point

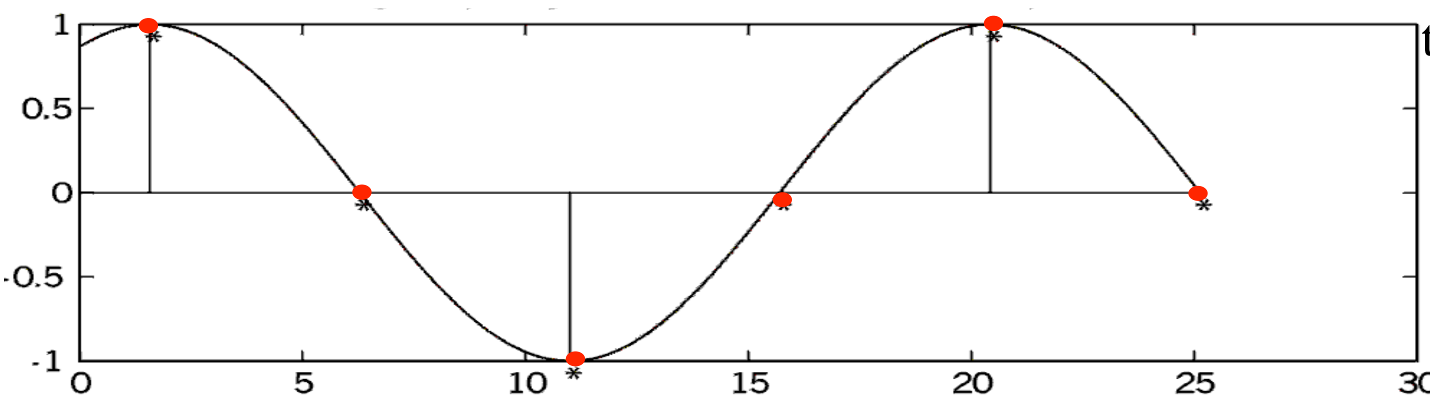
Undersampled, as:

$$T = \frac{3}{4}T_0$$

$$\Rightarrow f_s = \frac{1}{T} = \frac{4}{3T_0} = \frac{4}{3}f_0$$

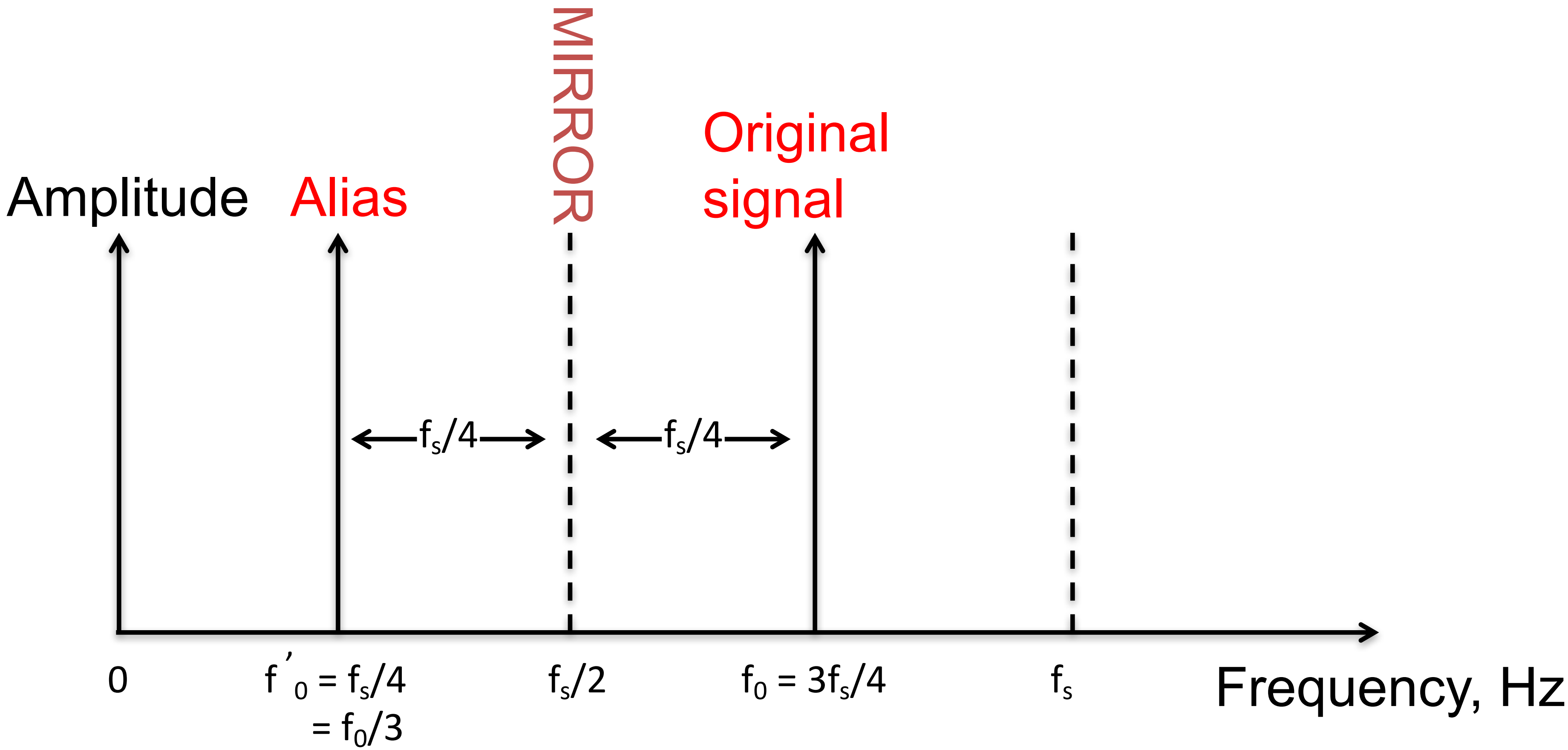
But $f_s \geq 2f_0$ for Sampling Theorem to be satisfied

On reconstruction, we obtain a signal with a frequency of f_0 / 3



We say that the original frequency of f_0 has acquired an 'alias' of f_0 / 3

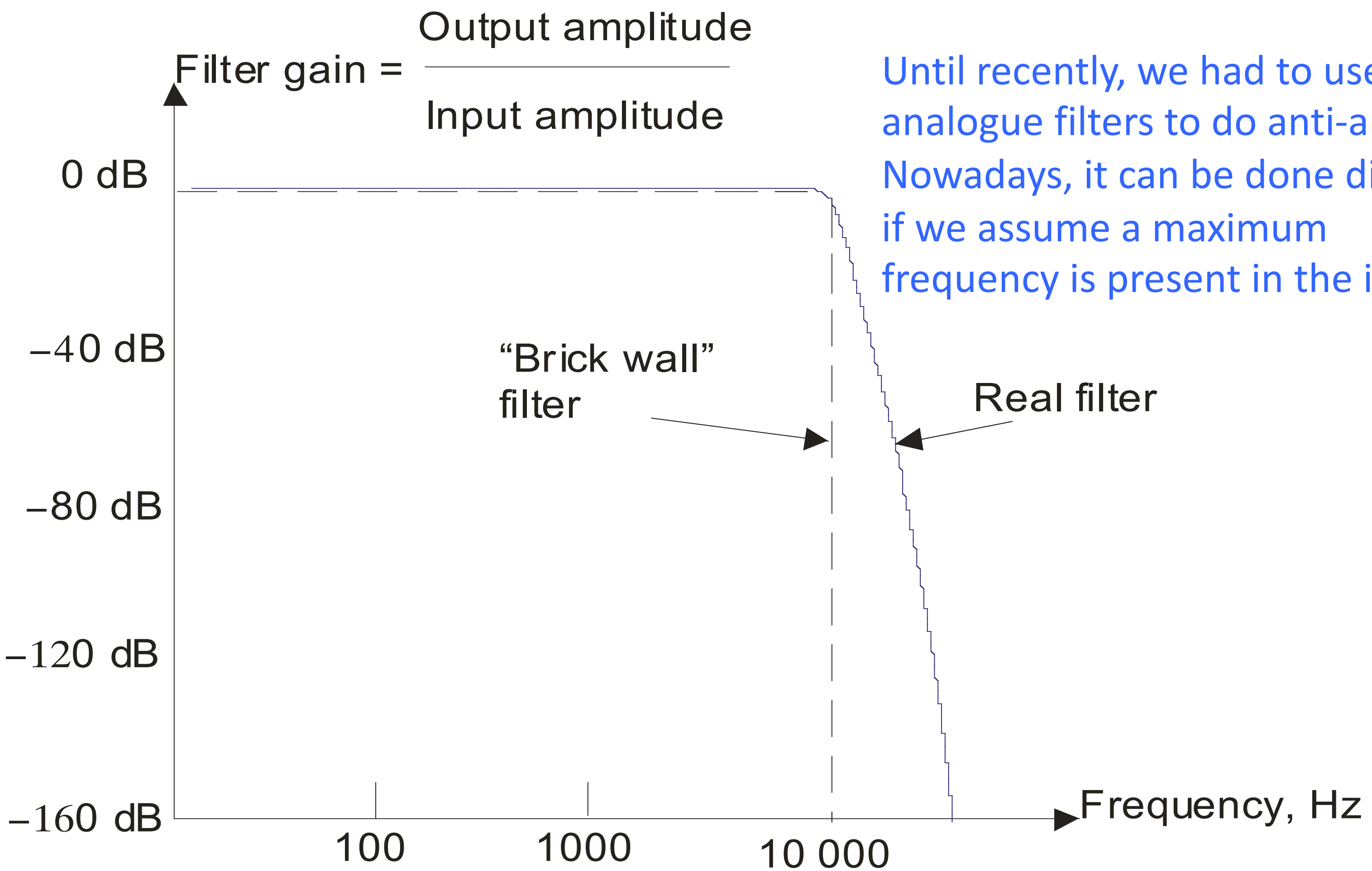
Effect on signal spectrum of aliasing



Aliasing causes a “mirroring” at $f_s/2$ in the frequency domain

Anti-aliasing filters

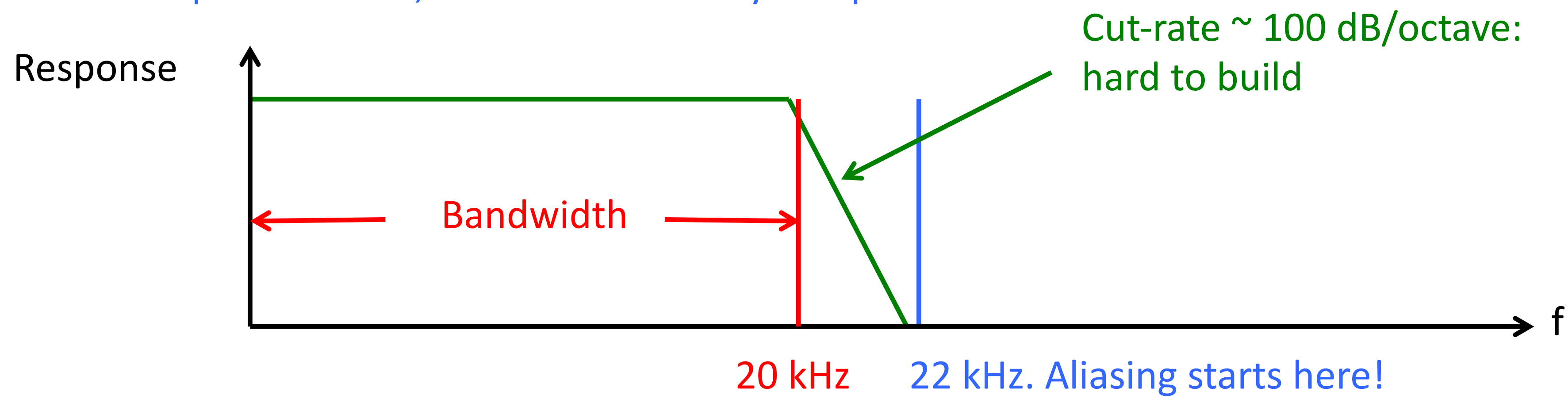
In practice, we cannot *choose* the highest frequency in our signal, so we have to remove all frequencies above $f_s/2$. To do this, we use an *anti-aliasing filter*. The response of a filter set for f_s = 20 kHz, is shown below.



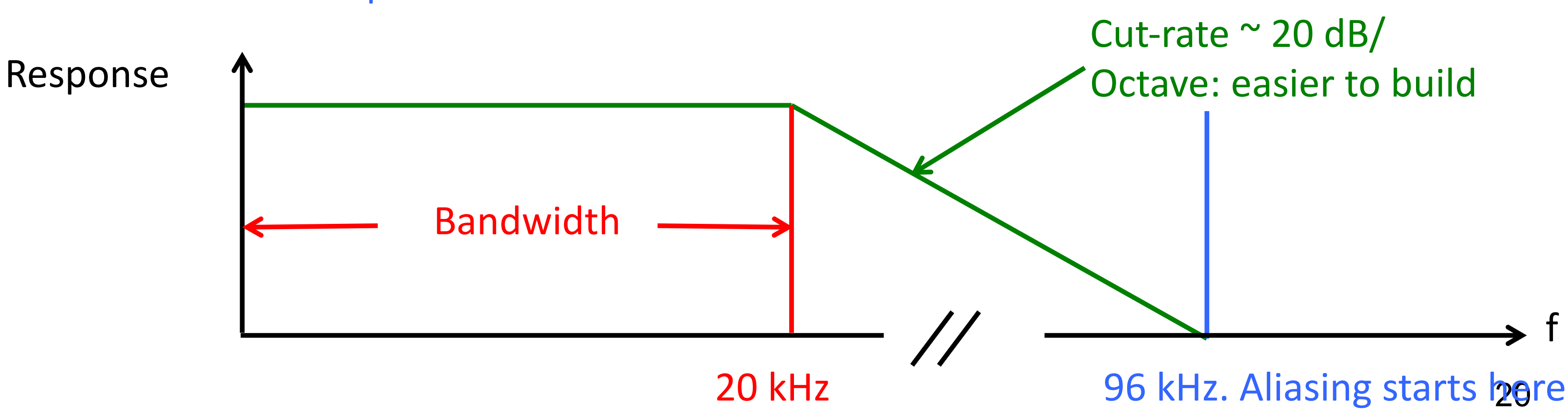
Until recently, we had to use analogue filters to do anti-aliasing. Nowadays, it can be done digitally, if we assume a maximum frequency is present in the input.

Over-sampling to make anti-aliasing filtering easier

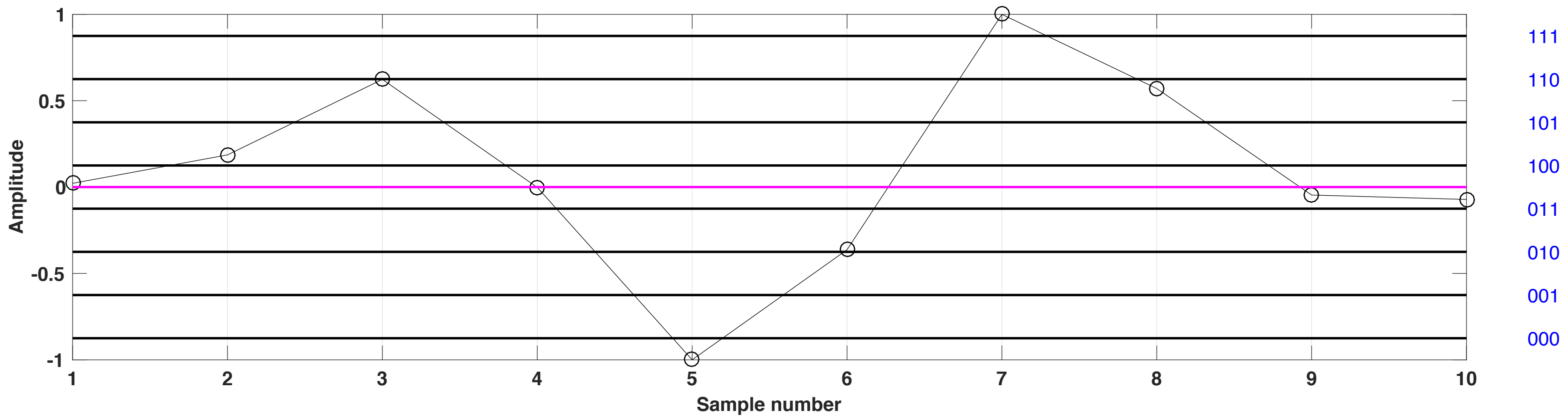
Let's assume that we are sampling audio and want a highest frequency of 20 kHz.
If we sample at **44 kHz**, we will need a very sharp filter:

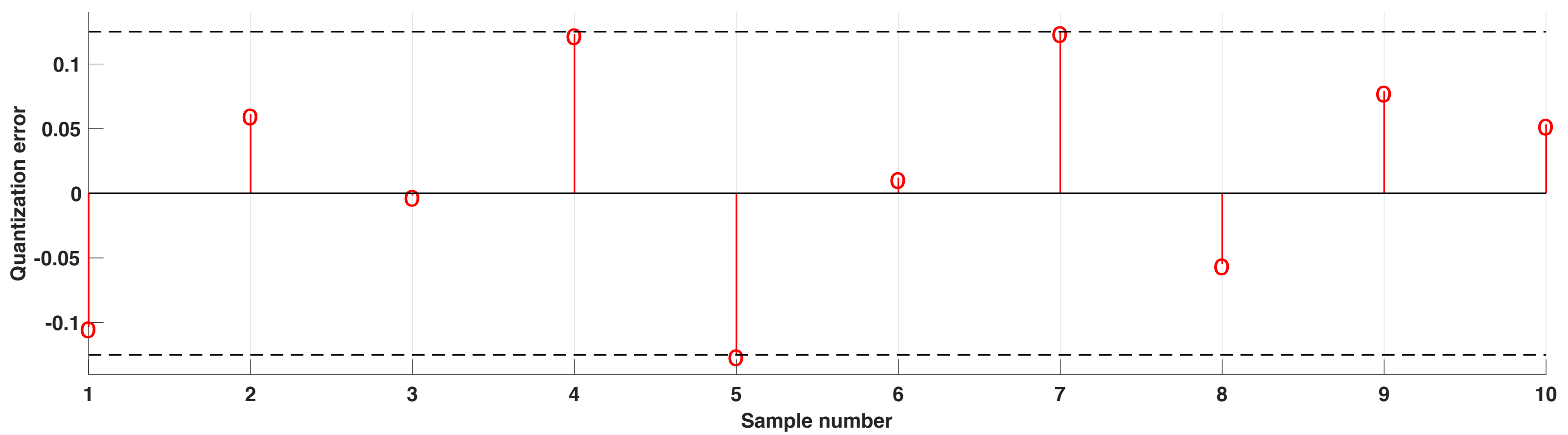


But if we over-sample at **192 kHz:**



Quantization Error





Sample value

Quantization level value

Quantization error $e_q(n) = x(n) - x_q(n)$.

$$\boxed{\text{N.B. } -\Delta/2 \leq e_q(n) \leq +\Delta/2}$$

ADC resolution: How Many Bits per Sample do I Need?

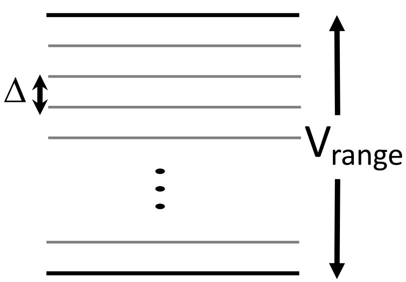
Example:

I have a thermometer that records a temperature range of -40^{O} C to 40^{O} C.
I wish to record this range to an accuracy of 0.5^{O} C.
How many bits to I need in my quantizer and what quantization accuracy will I actually get?

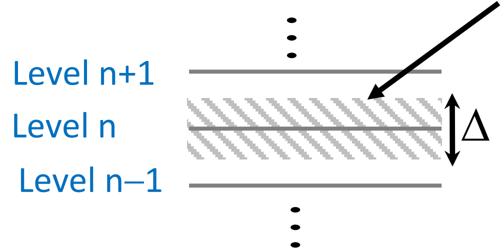
Solution:

The range is 80^{O}C in steps of 0.5^{O} C, so a minimum of 160 quantization levels are required.

If I use 7 bits, this gives $2^7 = 128$ levels, not enough, so I need to use **8 bits** which gives me $2^8 = 256$ levels.



$$\Delta = \frac{V_{range}}{(2^B - 1)}$$

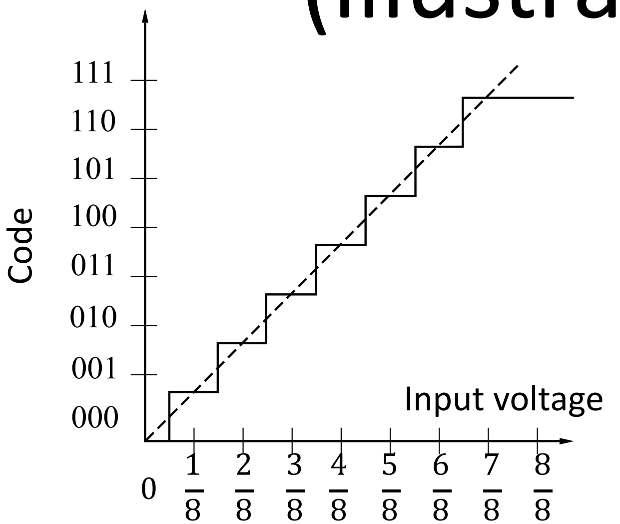


All samples lying in this region will be quantized to level n

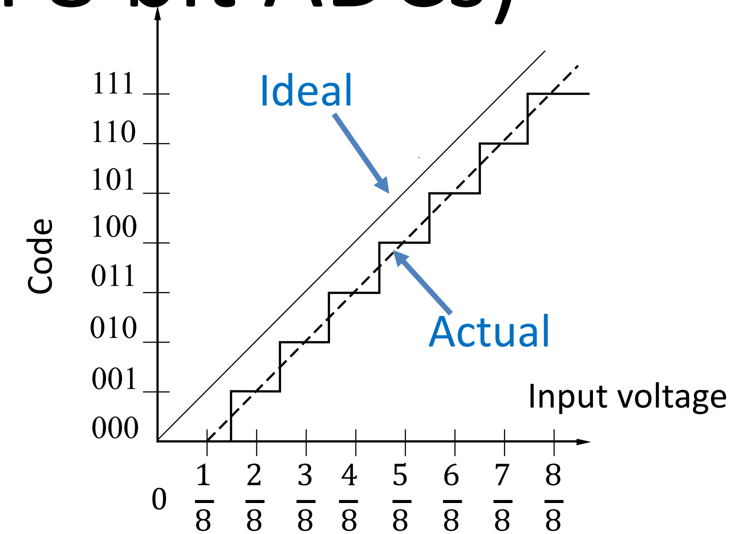
Hence the resolution of the quantization is $\pm\Delta/2$ volts.

$$\%\, Acc = \pm\frac{\Delta/2}{V_{range}}\times 100\%$$
$$= \pm\frac{1}{2(2^B - 1)}\times 100\% \approx \pm\mathbf{0.2\%} \quad \textbf{for B = 8}$$

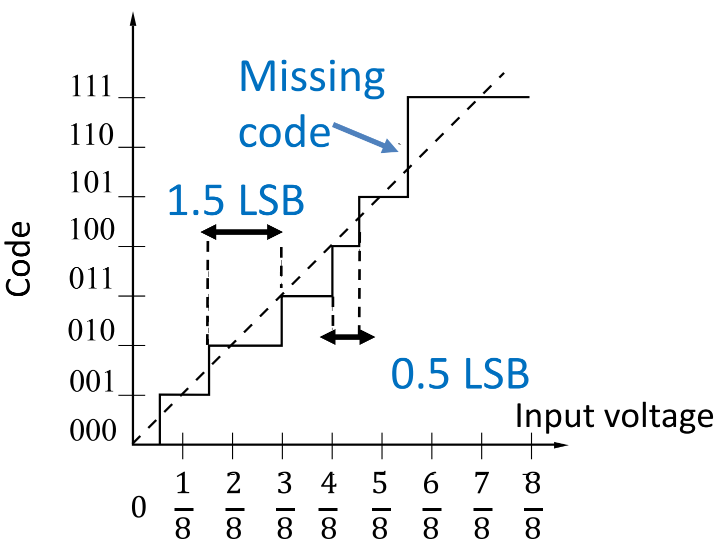
Accuracy Errors
(illustrated with 8 bit ADCs)



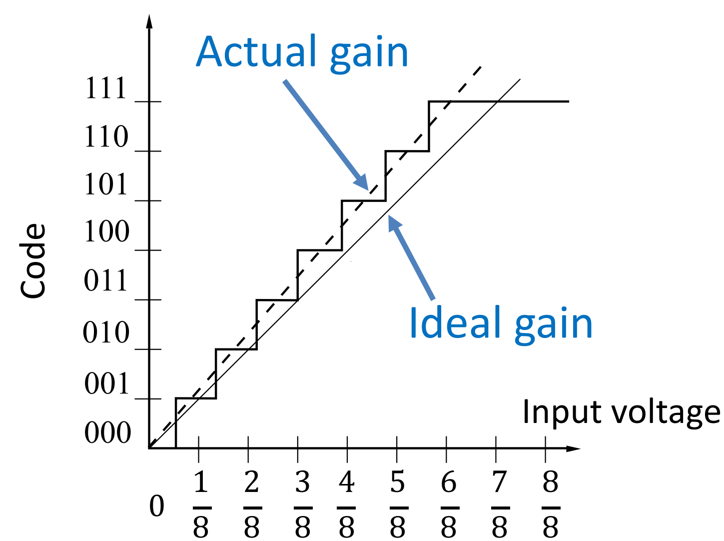
1. Ideal



2. Offset error of 1 LSB (least significant bit)



3. Differential Non-linearity

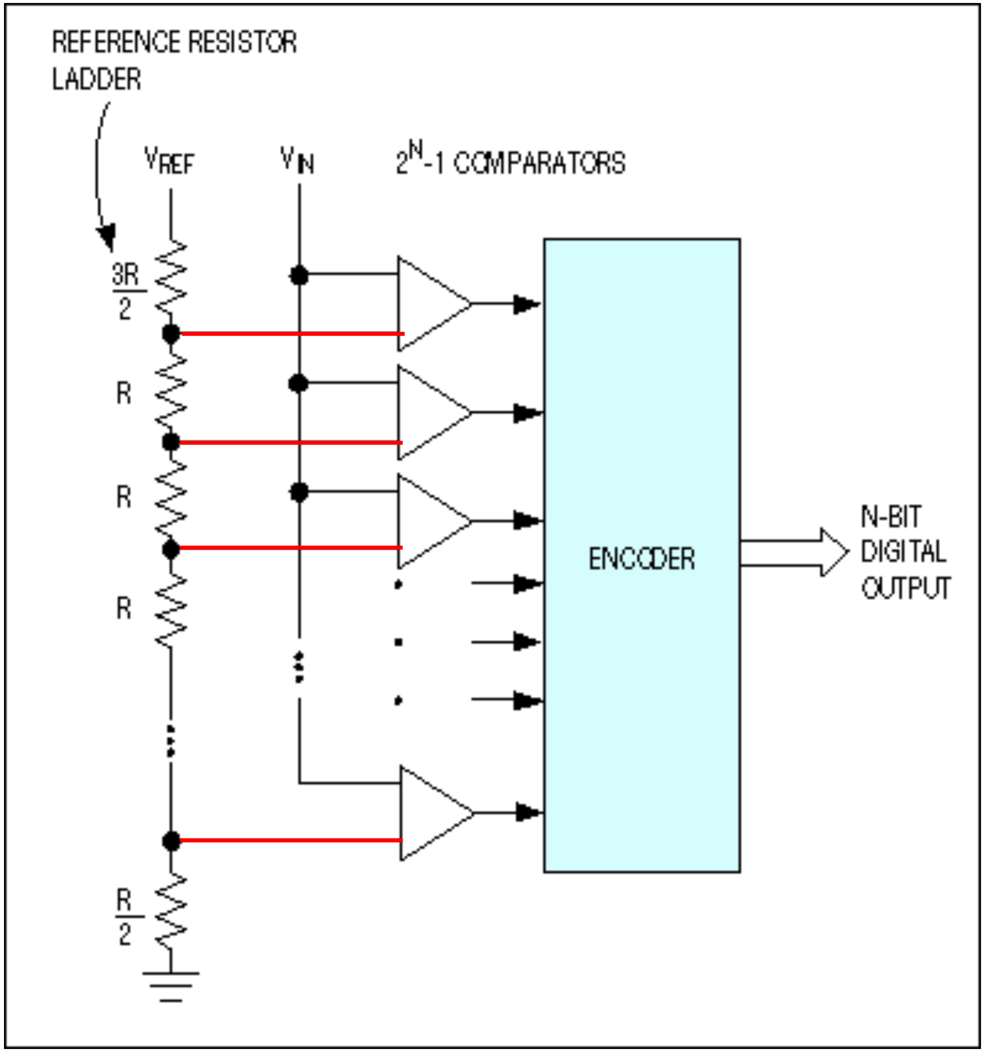


4. Gain error

Considerations when Choosing an Analogue to Digital Convertor

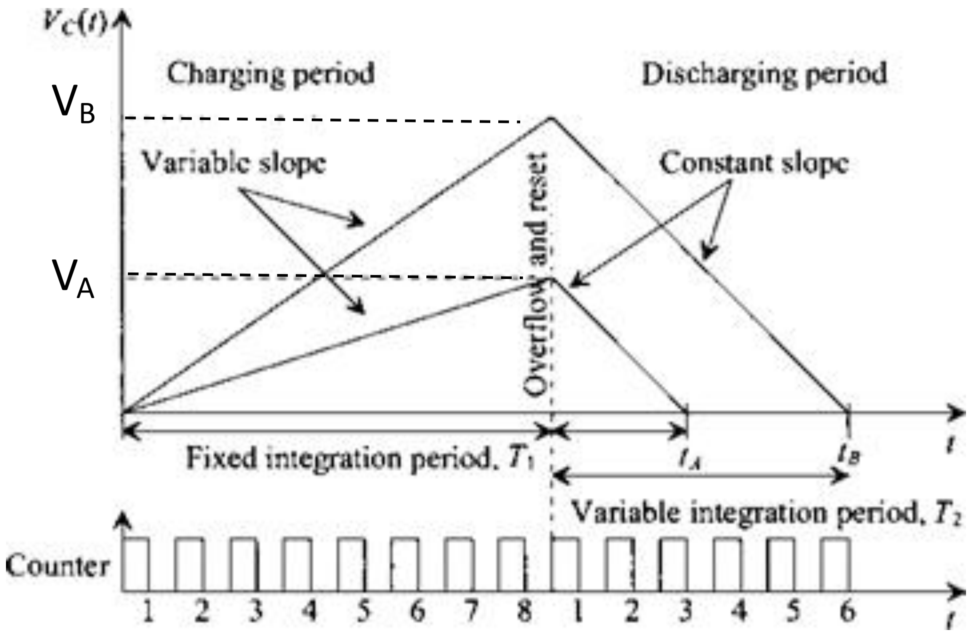
- There is a comprehensive discussion document available at a Texas Instruments site: http://www.ti.com/lit/an/sbaa004a/sbaa004a.pdf
- The main points are:
  - Resolution (covered in previous slide)
  - Accuracy: are there any missing codes? linearity?
  - Conversion speed (Maximum sampling rate)
    - Not usually an issue when taking measurements of the environment—a maximum sampling rate of 10 samples per second may be adequate.
  - Temperature range. Does it work in the cold and heat?
  - Input signal:
    - What *range* of voltage does the ADC accept?
    - Does the ADC accept *bipolar* voltages (in range ± V_{max} volts) or *unipolar* (in range 0 to V_{max} volts)
  - Does it have an on-board “sample and hold” circuit?
  - How many channels do you need?

Types of ADC I: Flash convertor



- PROS
  - Very Fast (Fastest)
  - Speed is only limited by gate and comparator propagation
- CONS
  - Expensive
  - Prone to produce glitches in the output
  - Each additional bit of resolution doubles the number of comparators.

Types of ADC II: Dual Slope Convertor

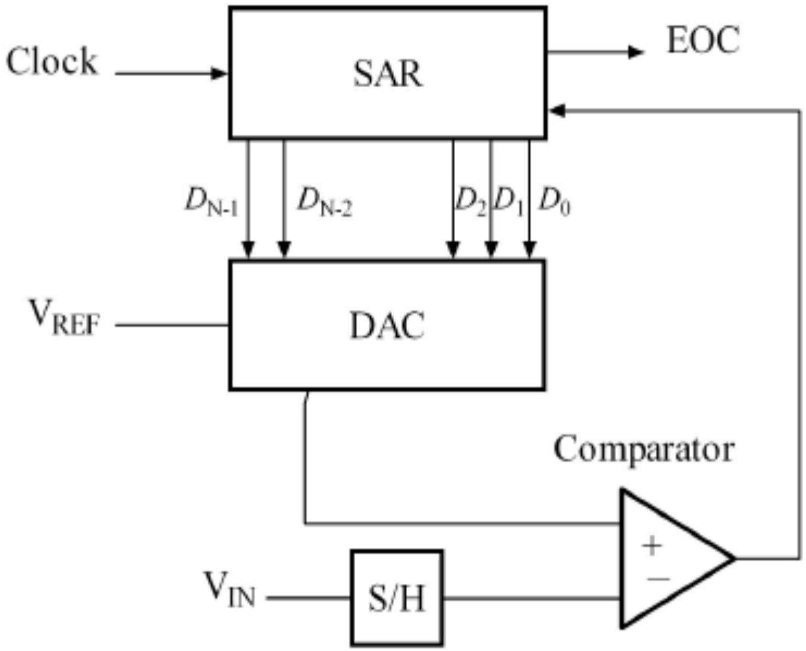


Period 1:
V_{in} is integrated for a fixed period:
Hence V_A or $V_B \propto V_{in}$

Period 2:
Voltage V_A or V_B ramps down to 0 V at a constant rate
Hence $T_2 \propto V_{in}$. Counter records time T_2.

- PROS:
  - Conversion result is insensitive to errors in the component values.
  - Fewer adverse affects from "noise"
  - High Accuracy
- CONS
  - Slow
  - Expensive in components

Types of ADC III: Successive Approximation Convertor



See e.g. http://ume.gatech.edu/mechatronics_course/ADC_F08.pdf for explanation of operation

- PROS:
  - Capable of high speed and reliable
  - Medium accuracy compared to other ADC types
  - Good tradeoff between speed and cost
  - Capable of outputting the binary number in serial (one bit at a time) format.
- CONS
  - Higher resolution successive approximation ADC's are slower
  - Speed limited to ~5Msps

The MCP 3004/3008 Convertor

Excerpt from datasheet available at
https://cdn-shop.adafruit.com/datasheets/MCP3008.pdf

- 10-bit resolution
- ± 1 LSB max DNL
- ± 1 LSB max INL
- 4 (MCP3004) or 8 (MCP3008) input channels
- Analog inputs programmable as single-ended or pseudo-differential pairs
- On-chip sample and hold
- SPI serial interface (modes 0,0 and 1,1)
- Single supply operation: 2.7V - 5.5V
- 200 ksps max. sampling rate at V_{DD} = 5V
- 75 ksps max. sampling rate at V_{DD} = 2.7V
- Low power CMOS technology
- 5 nA typical standby current, 2 µA max.
- 500 µA max. active current at 5V
- Industrial temp range: -40°C to +85°C
- Available in PDIP, SOIC and TSSOP packages

10-bit resolution

Low non-linearities

4 or 8 channels

Can convert voltage or *difference* of two voltages

No external sample/hold required

Output bits appear in a serial form (one after another)

One battery required

75000 samples per second if low voltage supply used

200000 samples per second if high voltage supply used

Very low current (hence power) consumption: nanoamps!

Good temperature operating range

Summary

- You need to understand something about signals before you can be confident of using ADCs
- The two key parameters for sampling signals are:
  - Sampling rate
  - Resolution (bits per sample)

  These should be considered carefully before choosing an ADC.
- It is important to visualise the *spectrum* of the signal before sampling it:
  - There is plenty of software to do this, including Python (e.g. `fft()` in the NumPy libraries).
  - MATLAB is also useful for this
- If connecting a signal directly to an ADC, you may need to use an anti-aliasing filter
- You will see in the lab that you will also need to condition the signal so that it lies in the right voltage range for the ADC.
- There are different types of ADCs available.
  - In general, there is a trade-off between speed (sampling rate) and resolution.
  - A Successive Approximation ADC is a good general purpose ADC