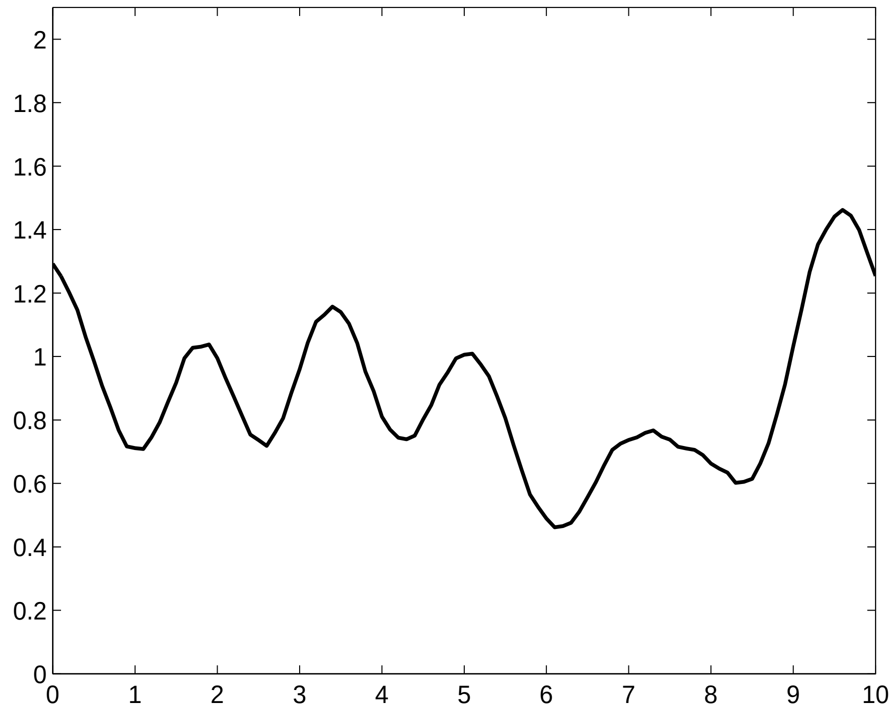
SIGNALS AND SYSTEMS: PAPER 3C1
HANDOUT 7.
Dr Anil Kokaram
Electronic and Electrical Engineering Dept.
anil.kokaram@tcd.ie www.mee.tcd.ie/∼ack

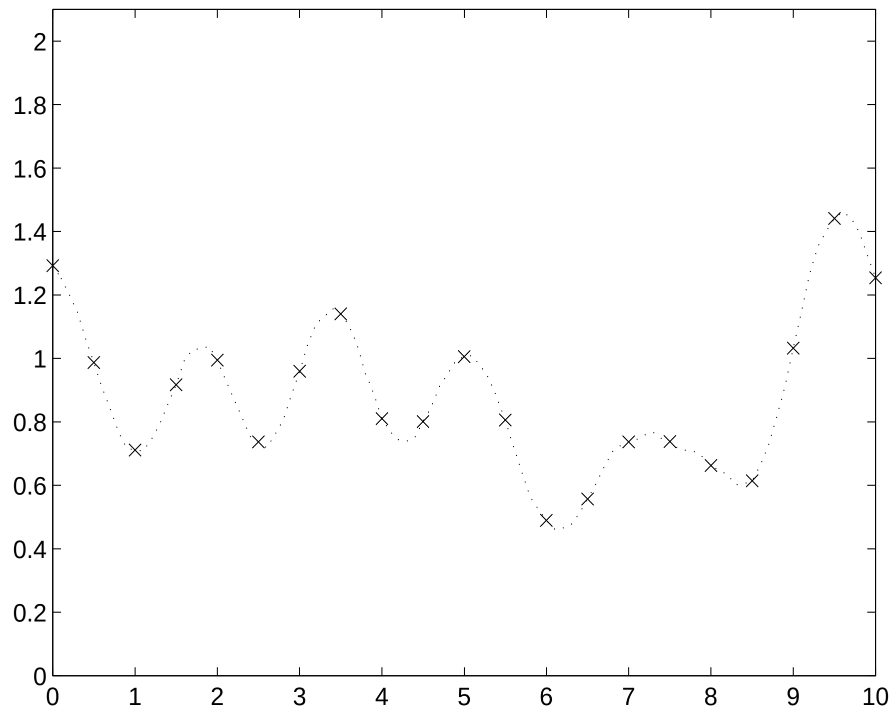
FROM ANALOGUE TO DIGITAL

To digitize signals it is necessary to define them by samples of discrete amplitude, taken at discrete instants in time. Each sample amplitude is quantized so that its 'voltage' is represented by a number which has a minimum (typically 0) and a maximum value (depends on the number of bits allocated for representing the number 8 bits = 255 max, 16bits = 65535 max).

- Much signal processing today is done in the digital domain. Digital signals can be manipulated by computers which are intrinsically very flexible i.e. programmable.
- Because digital signals can be easily manipulated one can make digital systems which implement much more complicated processes than would be possible using analogue techniques.
- Digital signal processing is encountered in everyday devices like mobiles, CD, .mp3 Players DVD Players (Digital Video Disc.), soundblaster cards for the PC, "web-cams", network cards etc. Even cars, microwaves ...
- For digital processing to be any use we must be confident that representing a signal by a set of discrete "samples" still retains all the information of the original analogue signal. We must then be also confident that it is possible to reconstruct the analogue signal from the samples after the samples have been processed.
- We'll first look at the effects of 'sampling' and then look at techniques for reconstructing the analogue signal from these samples.

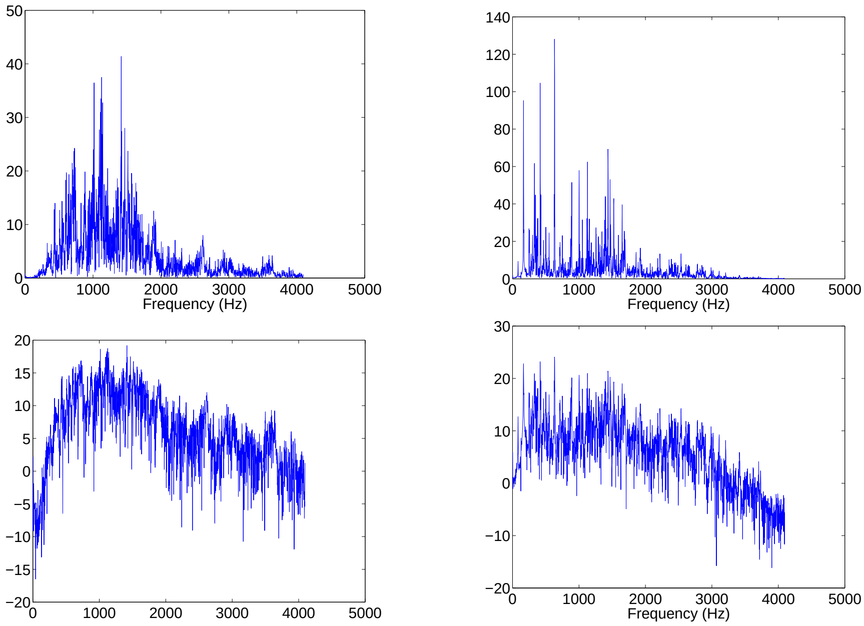
Sampling and Quantization



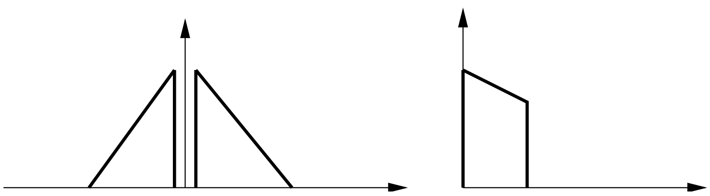


1 A note about representing spectra

- We will be deriving limits on sampling using spectal analysis. To do this we will be drawing a bunch of *representations* for signal spectra.
- Most naturally occuring (and interesting) signals have most of their spectral energy concentrated around the lower frequencies. In fact most natural signals (speech, music, pictures of natural scenes like landscapes) have spectra with energy that falls off with $1/f$ where f is frequency[1].



- So to visualise what is happening to signals in terms of frequency content, we tend to use pictures like below.

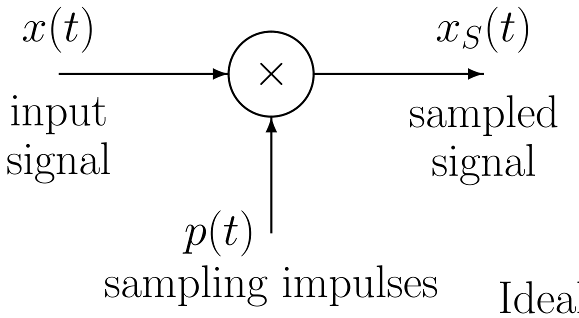


- Do not be alarmed by these pictures, they are just representations of signal spectra to allow us to visualise what would happen to the frequency content. In fact the most important thing about these pictures is that they allow us to visualise the bandwidth of signals without having to worry about a specific signal.

[1]A very weird phenomenon that has something to do with the fractal nature of . . . well . . . nature..

2 Sampling (Quantitative analysis)

Model sampling by an ideal sampler as below. The impulse train $p(t)$ selects the values of $x(t)$ only at the sampling instants. We wish to examine the spectrum of the sampled signal and compare it to the spectrum of the original analogue signal.



Ideal Sampler

$$x_S(t) = x(t) \times p(t) \tag{1}$$

The sampling pulses $p(t)$ are a periodic train of impulses of unit mean value and period T secs:

$$p(t) = \tag{2}$$

In order to make it easier to calculate the spectrum of $x_S(t)$, we use Fourier series analysis with $\omega_0 = 2\pi/T$ to express $p(t)$ instead as a sum of its Fourier components:

$$p(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\omega_0 t}$$

where

$$
\begin{aligned}
a_k &= \frac{1}{T} \int_{-T/2}^{T/2} T\delta(t)\; e^{-jk\omega_0 t}\; dt \\
&= e^{-jk\omega_0\,.\,0} \int_{-T/2}^{T/2} \delta(t)\; dt \\
&=
\end{aligned}
$$

$$
\Rightarrow \quad p(t) =
$$

Now we want to find $X_S(\omega)$ so:

$$
x_S(t) = x(t) \times p(t) = \tag{3}
$$

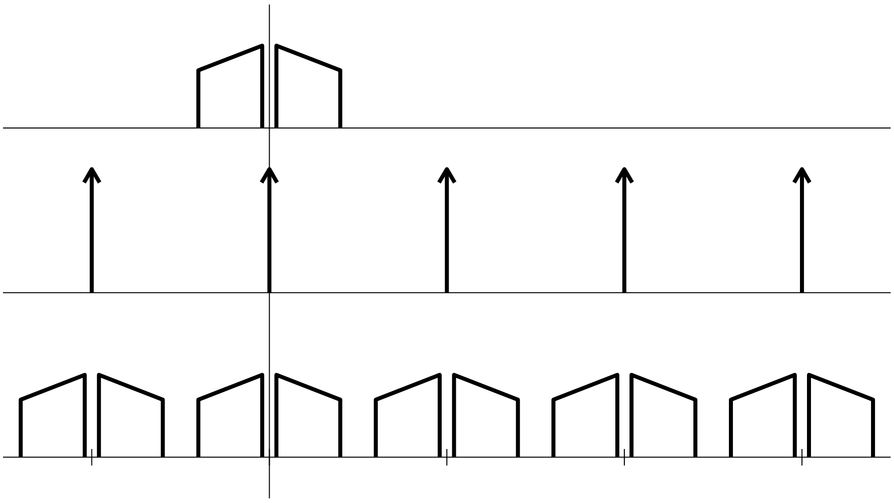
Note that the sampling frequency, $f_S = \dfrac{1}{T}$ Hz $= \dfrac{\omega_0}{2\pi}$ rad/sec. Taking Fourier transforms and using the frequency shift theorem:

$$
X_S(\omega) =
$$

Remember we derived the frequency shift theorem as

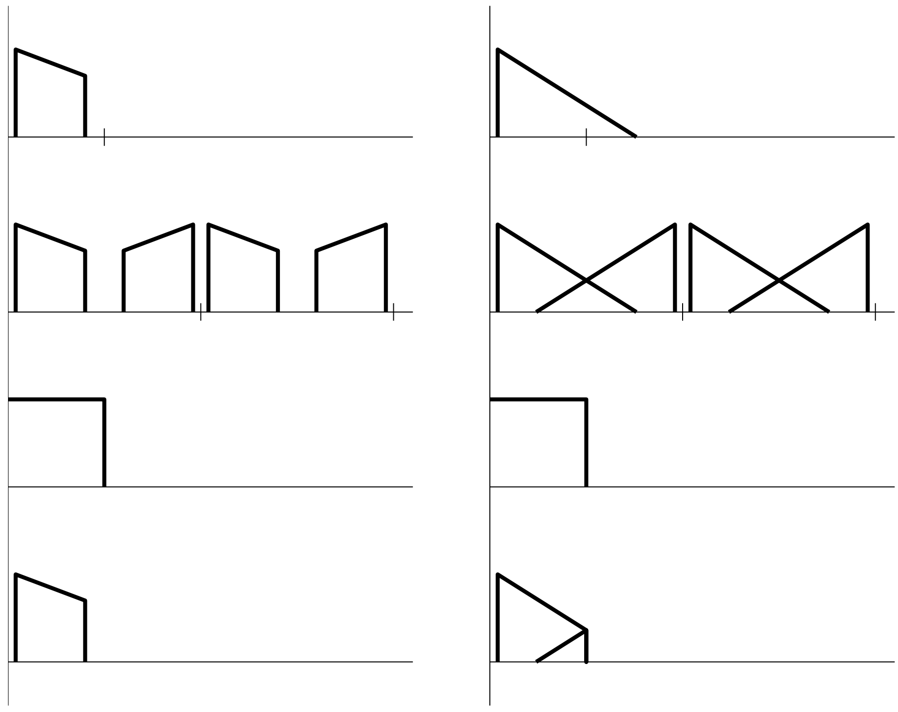
$$
\begin{gathered}
\text{If } x(t) \leftrightarrow X(\omega) \\
\text{Then } x(t)e^{jat} \leftrightarrow X(\omega - a)
\end{gathered}
$$

Hence the spectrum of a sampled signal is the original spectrum $X(\omega)$, repeated at multiples of the sampling freq. See below



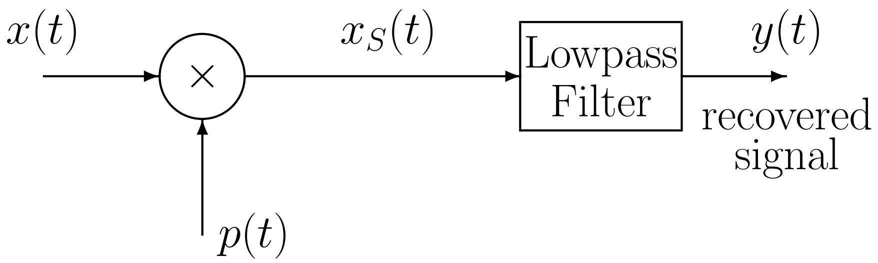
Spectral representation of sampling.

If the repeated spectra overlap, ALIASING occurs.



(a) No Aliasing (b) Aliasing

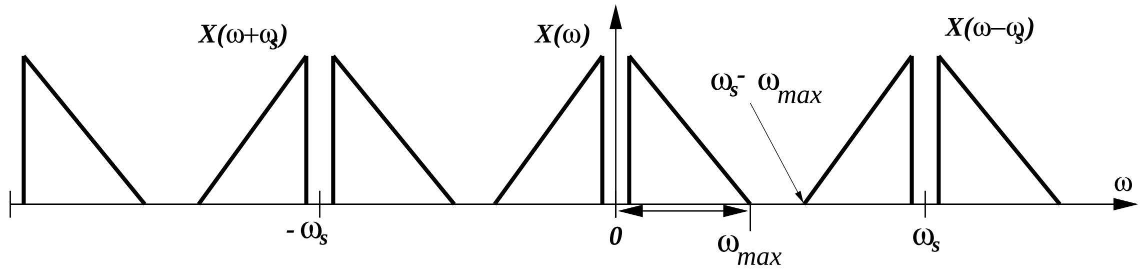
Signal recovery and the effect of aliasing.



Lowpass filter recovers the original signal.

If there is no aliasing, the original signal can be recovered perfectly from the sampled version by lowpass filtering, which removes all repeats of the original spectrum.

If aliasing occurs, perfect recovery is not possible - f_S is too low!



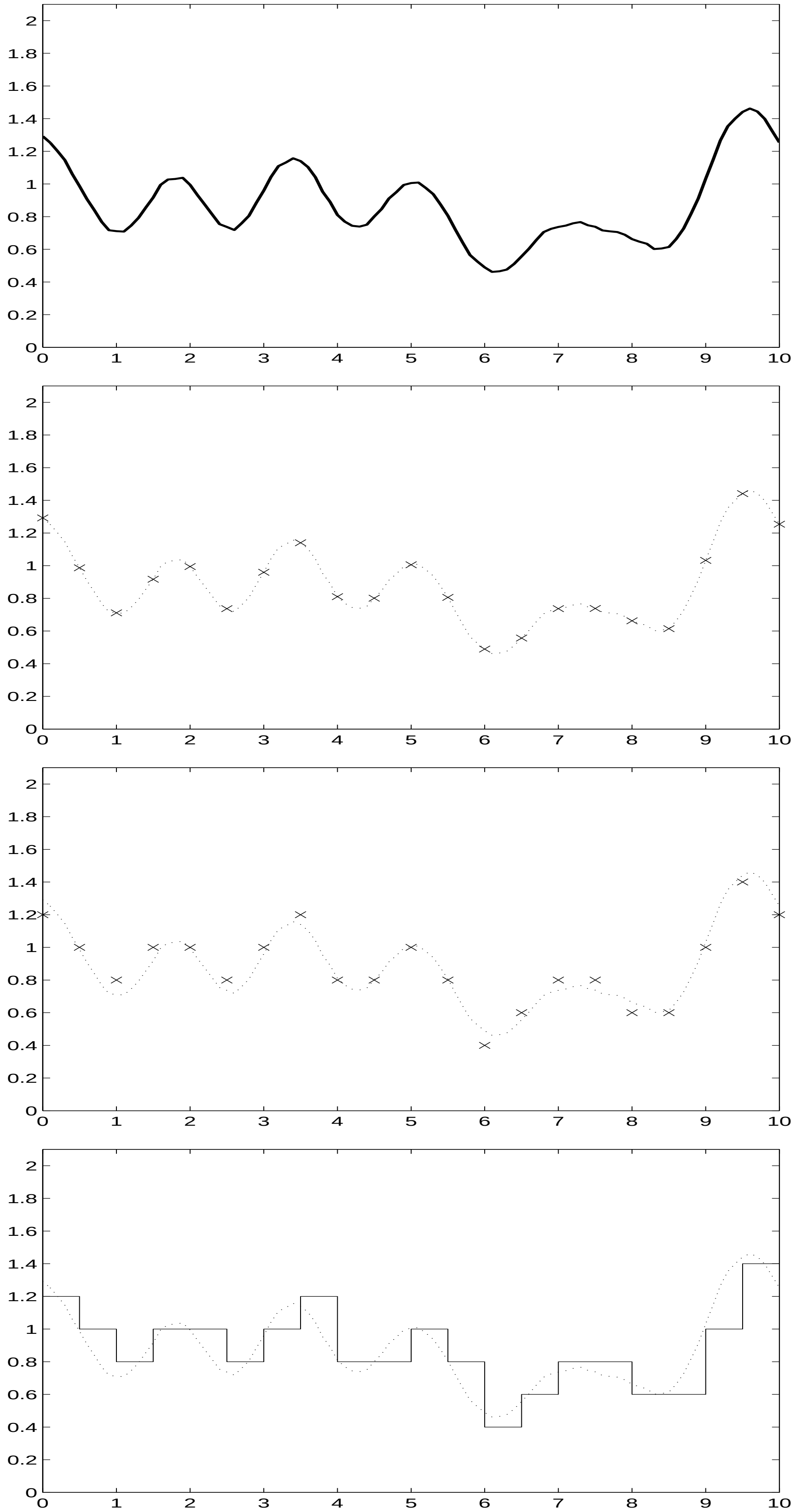
As long as $\omega_s - \omega_{\max} > \omega_{max}$ there is no aliasing. Therefore, minimum sampling frequency is when $\omega_s - \omega_{\max} = \omega_{max}$ which implies $\omega_s = 2\omega_{max}$ Same for Hz since $\omega_s = 2\pi f_S$.

The Sampling Theorem (Nyquist): If a signal has components with a maximum frequency of W Hz, then it is completely defined by samples which occur at intervals of $1/2W$ sec.

The Sampling Theorem (Nyquist): If a signal has components with a maximum frequency of W Hz, then it is completely defined by samples that are created by sampling at $2W$ Hz.

That is to say: if it is required to sample an analogue signal which has a spectrum containing frequency components up to W Hz, then the minimum sampling frequency required is $2W$ Hz. When this condition is satisfied it is possible to reconstruct the analogue signal exactly from the sampled data. This minimum sampling frequency required to prevent aliasing is called the “Nyquist” frequency.

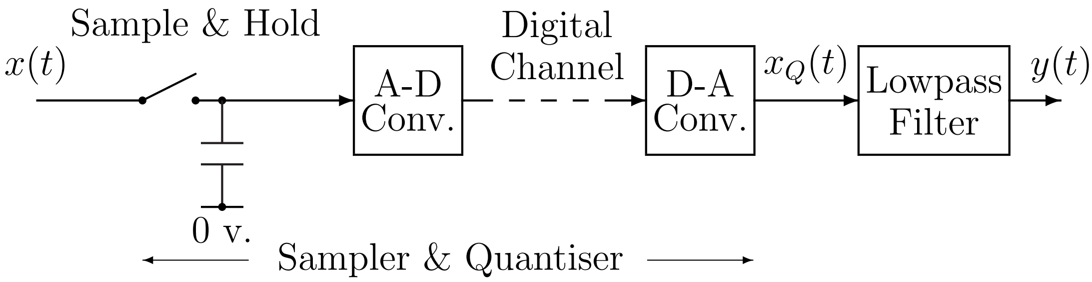
Sampling and Quantization



3 Quantization

Now we consider the effects of discrete quantization of the amplitude of our time samples.

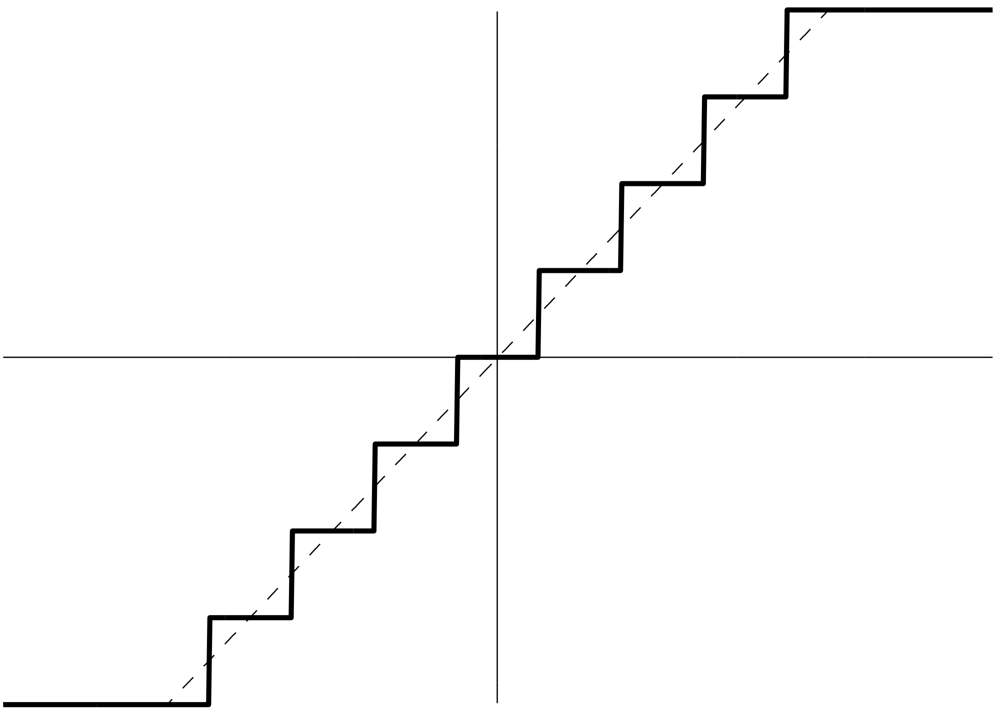
The figure below shows a digital coding system.



Digital coding system.

1. Sample and Hold (S and H): Samples the input and holds its value constant while the A-D converts it.
2. A-D Converter: Converts each input sample to a digital value. i.e. a binary number.
3. D-A Converter: Converts back to an analogue voltage. i.e. outputs a particular voltage when presented with a binary number.

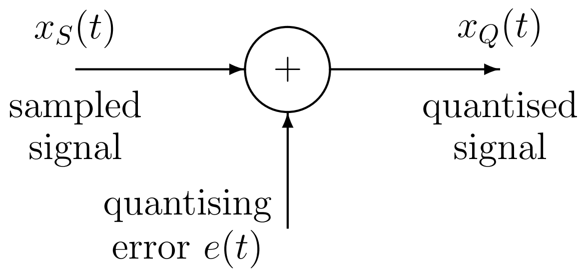
Quantization results in a transfer function similar to that shown in the figure below.



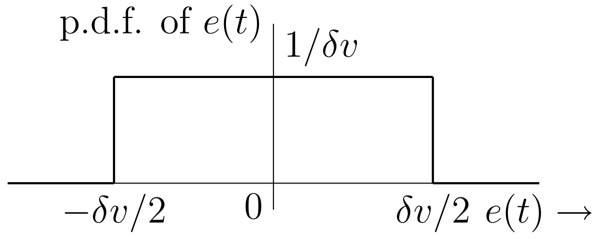
A-D + D-A transfer function (ideal linear quantizer).

If the quantizer has a small step size and it always rounds to the nearest output value, a given input voltage results in a random quantizing error that has a uniform probability of being anywhere from $-\delta v/2$ to $+\delta v/2$.

Hence the quantizing errors may be modelled as additive random noise with a uniform probability density function (p.d.f.). Using these ideas we can estimate the average error caused by quantization and so work out how many bits we would need to represent a particular signal properly. This will not be discussed further here.



Quantising noise model.



Quantising noise probability density function (p.d.f.).

Quantising image samples

Clockwise from top left: 8bit (256 Grey levels), 5bit (32 Grey Levels), 1bit (2 Grey levels), 2bit (4 Grey Levels) Quantisation.

4 Methods of A-D and D-A conversion:

4.1 D-A Conversion:

The most common type of D-A converter (DAC) is based on switched resistive ladders. A typical circuit is shown.

4.1.1 Resistive Ladder Digital-to-Analogue Converter:

D-A converter using R-2R ladder and FET switches.

Voltages and currents become halved as they progress from left to right via each stage of the ladder. The left-hand switch is controlled by the most significant bit of the digital input word, and the other switches are controlled by bits with decreasing significance. Each switch routes the current from the 2R resistor above it either to earth or to the virtual earth of the op-amp. This ensures that voltages in the ladder remain constant, independent of the states of the switches. The op-amp adjusts the analogue output voltage so that the current through the feedback resistor equals the sum of the currents switched to the virtual earth by the input data, and hence the output voltage is proportional to the value of the input word.

The R-2R ladder is effective because the resistors have only 2 different values so their ratios will match very accurately when fabricated on the same piece of silicon. High relative accuracy is required so that the input/output characteristic remains monotonic and linear even when the input code changes from 100000... to 011111.... The response time of the converter is determined largely by the decay time of capacitive transients in the FET switches and the settling time of the Op. Amp. (typically of the order of 0.1 to 1 μs.)

4.2 A-D Conversion:

There are 3 main types of A-D converter (ADC):

1. Dual slope integration ADC – based on charging and discharging a capacitor and timing this using a counter. It is capable of high accuracy ($\sim$ 16 bits) without needing very precise components, but it is slow ($\sim$ 10 ms. conversion time) due to the time taken for the counter to count through all the quantizer states. It is useful for low bandwidth instrumentation applications.

2. Successive approximation ADC – uses a fast DAC in a successive approximation feedback loop. It achieves fair accuracy ($\sim$ 12 bits), and is quite fast ($\sim 5\mu$s.). It is useful for audio and modem signals.

3. Flash ADC – uses many comparators to compare the input with all possible quantizer thresholds simultaneously. Due to practical limits on the number of comparators it only gives poor accuracy ($\sim$ 8 bits), but is very fast ($\sim$ 50 ns.). It is used mainly in wide bandwidth video applications.

4.3 Dual-slope Integration Analogue-to-Digital Converter:

Dual-slope integration A-D converter. Conversion method:

1. Step 1: C accumulates charge at a rate proportional to V_{analog} for a time $T_1 = N/f_C$, where $N = 2^n$ (In diagram n=4, so N=16) and T_1 is measured by the n-bit counter.

2. Step 2: The time $T = Q/f_C$ to discharge C back to zero at rate proportional to V_{REF} is measured by the n-bit counter.

   The output value is given by the count value Q, where:

   $$Q = T\ .\ f_C = \frac{V_{\text{analog}}}{V_{REF}}\ .\ N$$

   Hence Q is proportional to V_{analog} and does *not* depend on RC.

The dual-slope converter is guaranteed monotonic and provides good linearity without accurate components, BUT it is slow, because up to $2N$ clock cycles are needed and $N = 2^n$ for n-bit precision.

It is typically used for low-speed high-accuracy converters as in digital voltmeters or sensor monitors (thermometers, strain gauges etc.).

4.4 Successive Approximation Analogue-to-Digital Converter:

Successive approximation A-D converter. The control logic adjusts the D-A converter to make $V_{\mathrm{d/a}} \approx V_{\mathrm{analog}}$, using the following successive approximation procedure :

1. Step 1: Set $Q_{IN} = 10000\ldots$ (MSB ... LSB). (= 1000 above)
2. Step 2: If $V_{\mathrm{analog}} > V_{\mathrm{d/a}}$, set $Q_{IN} = 11000\ldots$ (= 1100 for cct above.)
   Else set $Q_{IN} = 01000\ldots$ (= 0100 for cct above.)
3. Step 3: If $V_{\mathrm{analog}} > V_{\mathrm{d/a}}$, set $Q_{IN} = x1100\ldots$ (= $x110$ above.)
   Else set $Q_{IN} = x0100\ldots$ (= $x010$ for cct above.)
   where x is the value of the bits determined in the previous step.
4. Steps 4 to n: Append the bit decision of the previous step to the least significant end of x and repeat step 3 for progressively less significant bits.
5. Step $n+1$: Determine the l.s.b. from V_{analog} and $V_{\mathrm{d/a}}$, and read out the result Q. (In cct shown, just apply step3 again to final bit.)

The time to convert is now only $n+1$ cycles for n-bit accuracy (instead of 2^{n+1} cycles for the dual-slope converter), BUT an accurate (and fast) n-bit D-A converter is needed (the D-A converter must be guaranteed monotonic to avoid missed codes.)

This is a general purpose method used for audio frequencies and bandwidths up to about 100 kHz.

For dual polarity inputs, an offset is added to the D-A converter output so it is symmetrical about zero, and if necessary the m.s.b. of the output Q is inverted to provide 2's complement coding.

For speech coding at 64 kbit/s in the public telephone network, the D-A converter is modified so that V_2 is approximately an exponential function of Q_{IN}, and this causes Q to be approximately a logarithmic function of V_1. Hence low amplitude speech samples are coded with finer resolution than large samples, which is a good way to minimize the audibility of quantizing noise.

4.5 Flash Analogue-to-Digital Converter:

Flash A-D converter with n-bit resolution.

For a word size of n bits, $2^n - 1$ comparators 'instantly' determine which quantizing interval V_{analog} is in, by simultaneously comparing it against $2^n - 1$ thresholds.

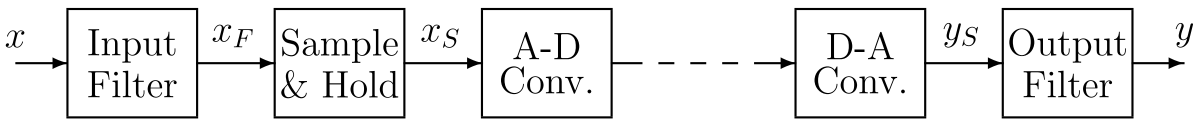
The comparator outputs must then be decoded into the n-bit output value (usually achieved by converting to the n-bit Gray unit-distance code first).

The flash converter is very fast, as it is limited only by comparator and logic propagation delays, BUT it becomes very complex if $n > 8$ (e.g. 1023 comparators are needed if $n = 10$).

It is used for video bandwidth signals at sampling rates up to 20 MHz, but usually $n \leq 8$ bits.

5 Anti-aliasing/smoothing filters

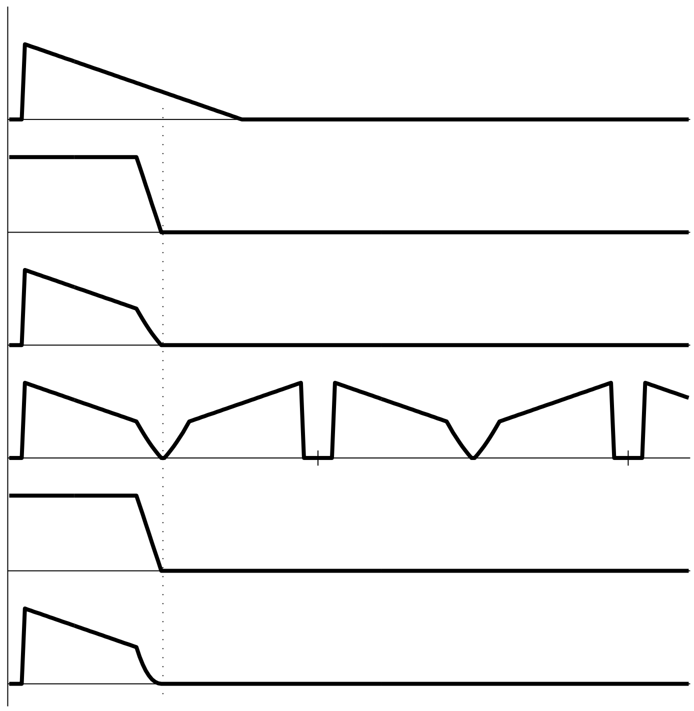
In most digitizing systems, 2 lowpass filters are required, as below.



Filters required for A-D and D-A conversion.

- Input filter – prevents possible aliasing distortion by eliminating any input frequencies $> \frac{1}{2}f_S$.
- Output filter – removes all components of the sampled signal spectrum $> \frac{1}{2}f_S$, to regenerate the original continuous signal.

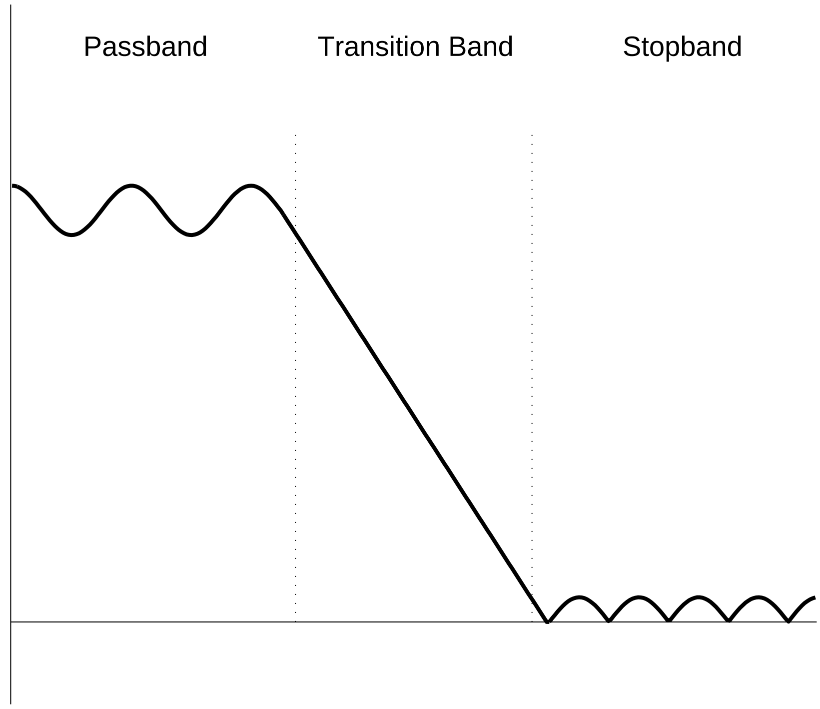
Next fig. shows the filter actions.



Spectra of signals in figure above

Let the filters have a passband of bandwidth f_P .

Realisable filters must also have a transition band of finite width f_T between the passband and the stopband. To make f_T narrow, the filter complexity (no. of poles and zeroes) must be high.



Typical filter response.

In both filters, the stopband must start at $\frac{1}{2}f_S$ to avoid aliasing and unwanted output components.

$$\text{Hence} f_P + f_T \leq \tfrac{1}{2}f_S \tag{4}$$

If the required signal bandwidth $= W$, then $f_P \geq W$ for negligible frequency distortion.

Let us assume that $f_T = \alpha f_P$ for a given complexity of filter.
Typically $\alpha = 0.1$ (high complexity) to 1.0 (low complexity).

Therefore the minimum sampling frequency f_S is given by:

$$f_S = 2(f_P + f_T) = \tag{5}$$

Therefore if low filter complexity is important, we must choose f_S to be significantly greater than $2W$ (e.g. $4W$).

Sampling in images

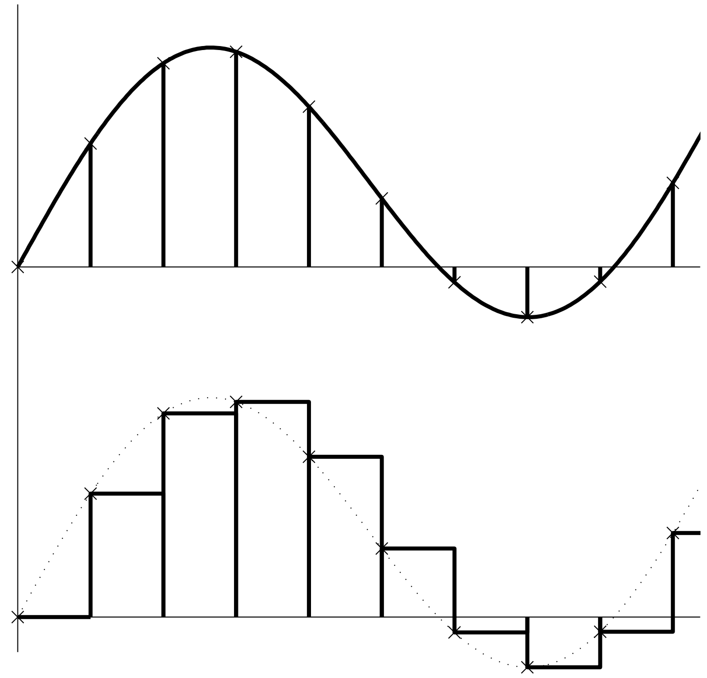
Top left: 4th Order butterworth filter (on rows then columns) for anti-aliasing

Top right: No anti-aliasing filter

Bottom: Analogue lenna [2]

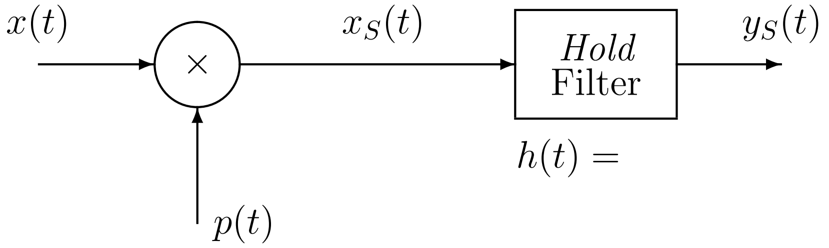
[2] In fact, that's a lie, this picture is also digital, but the sampling rate used was 3 times higher than the top two images and the Nyquist limit is satisfied (in this case), even though an anti-aliasing filter was not used.

6 Effect of D-A Hold action on the system frequency response:



D-A converter *hold* action.

The A-D + D-A system is equivalent to an ideal sampler followed by a filter which extends each sampling impulse into a pulse of width $T = 1/f_S$, as below.



Equivalent circuit for D-A *hold* action.

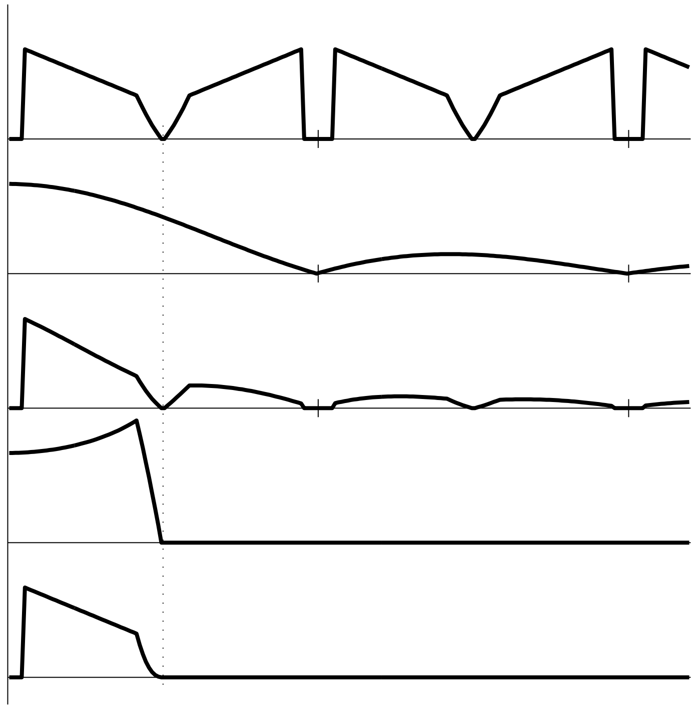
If the filter impulse response is given by:

$$\begin{aligned} h(t) &= 1 \qquad \text{for } 0 \leq t < T, \\ &= 0 \qquad \text{elsewhere,} \end{aligned}$$

$$\text{then} \quad |H(\omega)| = \tag{6}$$

See fig 4.4.6 for the effect of this filtering on the spectrum $Y_S(\omega)$ of the D-A output.

It is possible to modify the Output Filter response as shown so as to correct for the *hold* filter and obtain a flat response overall.



Spectra of signals modified by D-A *hold* action.