XILINX®

The **DSP** Primer 4

Introduction to DSP

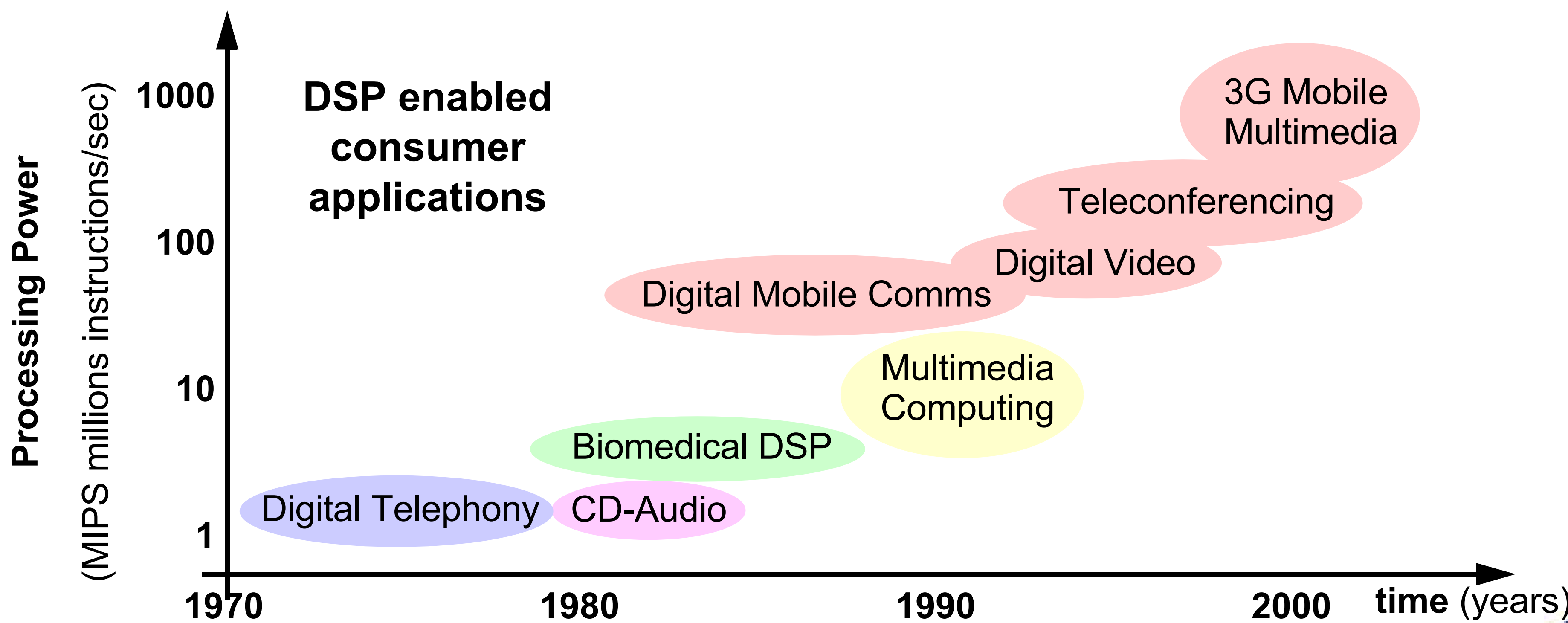
Return
DSP*primer* Home

Return
DSP*primer* Notes

THIS SLIDE IS BLANK

The DSP "Revolution"

- ***1980s:*** Special-purpose DSP microprocessors introduced.
- ***1990s:*** Processing power of DSP (micro-)processors has increased by an order of magnitude, and price decreased.
- The (digital) *reliability*, *repeatability*, and *programmability* of DSP has widely displaced analogue systems in both consumer and industrial markets.



Notes:

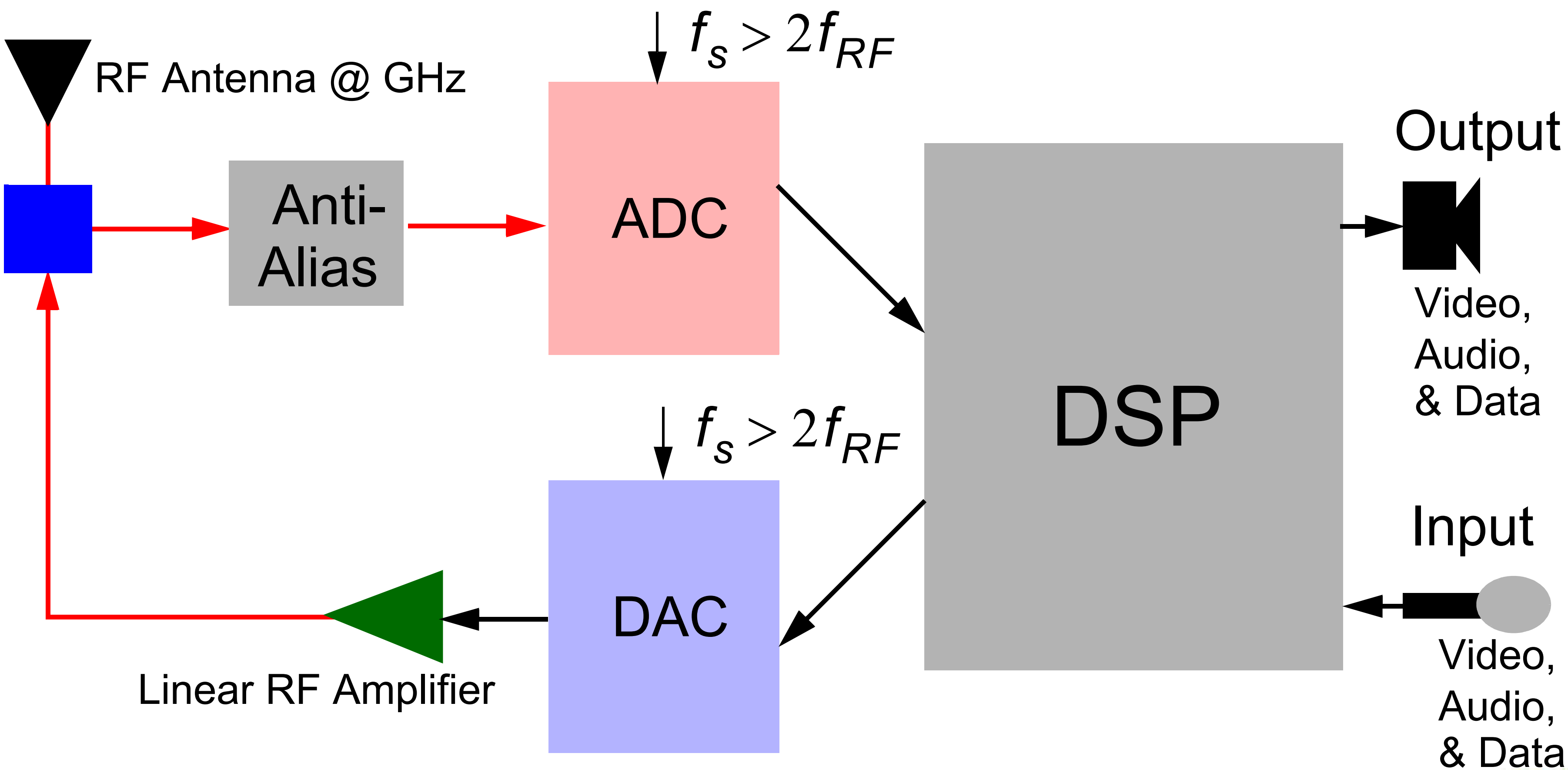
Digital Audio: Early 1980s DSP systems such as CD-audio did not require high level of processing power. Data was mainly be read from the CD and output via a digital to analogue converter (DAC). More recently CD Audio systems have been equipped with sound effects systems, recording capabilities and so on; however processing requirements are still relatively low for such operations. **The Modem:** In the 1990s the computer "fax-modem" became ubiquitous and from 1990 the speed of modems increased from a nominal 2400 bits/sec to a standard of 57200 bits/sec by the end of the decade. This communication was over the same copper wires that have existed in the ground for some 50 or more years; so why the order of magnitude increase in speed? The answer is again, DSP, or more specifically adaptive DSP algorithms in the form of the LMS (least mean squares). Via adaptive DSP enabled echo cancellation and data equalisation, bandlimited telephone channels were used with various signalling methods to reach data speeds approaching the theoretical limit by essentially using DSP to correct for any distortion that may have been present in the channel.

Digital Subscriber Loops: The last couple of years of the 1990s saw the introduction of DSL (digital subscriber loops) which bring data rates of millions (M) bits/sec to the home over conventional telephone lines. In summary DSL is using DSP techniques to allow the high frequency portions of a copper wire twisted pair (MHz) to be utilised. Prior to the requirements for fast data communications, most telephone lines were bandlimited between 300-3400Hz - sufficient for voice, but rather limiting for data communications - this is the reason that "voiceband" modems have a clearly calculable limit based on the available teleco bandwidth of a few kHz. The introduction of DSL equipment to telephone exchanges will bring a new lease of life to the copper wire infrastructure.

The application of **mobile multimedia** will allow consumers to communicate via teleconferencing (audio and video), and transmit documents (email/fax) from a small hand held communicator. Very high levels of processing power will be required for the audio/video coding/compression algorithms and for the DSP communications strategies. Third generation mobile communications (3G) will allow data rates of up to 2Mbits/sec to be available from hand held wireless devices (laptop/mobile communicator). To achieve these high rates a "chip" rate of some 5Mbits/sec to and from the basetation will be used. In order to process this speed of data transfer, a modulation scheme called CDMA (code division multiple access) will be used. DSP strategies for pulse shaping, channel equalisation, echo control, speech compression and so on, will require very powerful DSP devices capable of 1000's millions of instructions per second will be required. 2002 is the roll-out year....

DSP in 2000's - Towards Software Radio

- The ideal DSP enabled Software Radio receiver directly convert to/from RF frequencies (typically GHz, (G = 10^9).
- Initial implementations will work with IF (mixed down from GHz to a few MHz) frequencies.



Notes:

For the received signal, all downconversion/demodulation is done after the wideband ADC sampling (so called zero-IF or homdyne receiver). As in current 2nd generation mobile, all baseband processing (echo cancelling, speech coding, equalisation, despreading, channel decoding) is done in the DSP processor.

For current mobile 2nd generation technologies, a desirable software radio would be one that could operate in the 800 to 900MHz spectrum and can adapt between AMPs, GSM, DAMPs, CT2 and so on by simply changed the software on the DSP processor.

The next generation of UMTS, although being driven by cooperating worldwide standards is likely to be a major beneficiary from the implementation of software radio strategies. Currently multiband radios are available (particularly in the USA) that cover both GSM at 900MHz and DECT at 1300MHz, although this is essentially accomplished by having separate hardware rather than as a programmable SR architecture.

For next generation CDMA with 5MHz bandwidth if the ADC used 4 times bandpass sampling, then more than 20Msamples per second are required, with an analogue front end capable of integrating at a frequency range extending as high as 2GHz. At *least* 18 bits of resolution is likely to be required. Currently such devices are not available. The DAC would require to produce more than 20Msamples/sec at up to 18 bits, and input to a linear RF amplifier.

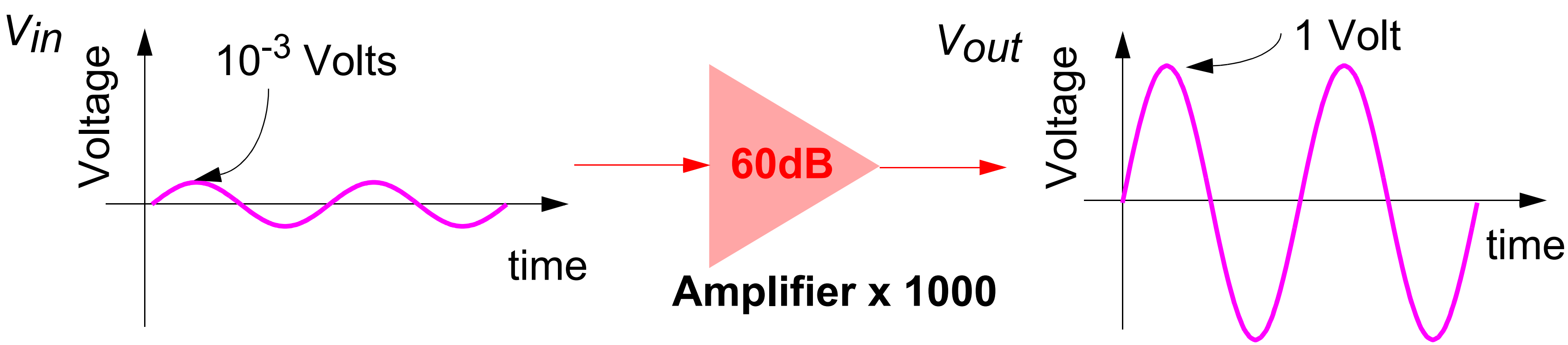
Some of the ideas for software radio include algorithms toolbox’s, where for example there is separate software for each standard stored within a handset or mobile terminal. Alternatively there may be a toolbox of algorithms (such as QPSK, equalisers etc) which can be called with appropriate parameters depending on the actual standard (GSM, W-CDMA etc) being used.

There is also the concept of over-air download whereby a mobile terminal can download required software to the generic hardware of the mobile terminal. This, however, is a concept that is still a long time away.

Amplification and Conditioning

- The voltage from a signal sensor is very small in magnitude. A microphone may produce **voltages** of the order of 10^{-6} volts. Similarly for ECG sensors, vibration sensors etc.

- Prior to recording the signal or reproducing with an actuator an amplifier should signal condition by linearly amplifying the signal by an appropriate factor.

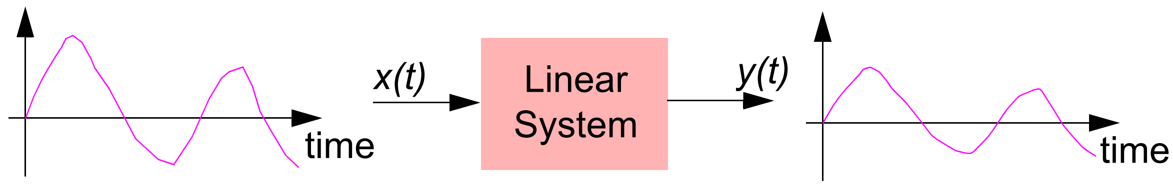


- The above amplifier adds 60dB of gain ($20\log_{10}1000 = 60$)

$$V_{out} = 1000\,V_{in} \text{ therefore } \frac{V_{out}}{V_{in}} = 1000$$

Notes:

A system is said to be linear if the output can formed as the convolution of the input and system transfer function



In general for a *linear system* $y(t) = f(x(t))$, if,

$$y_1(t) = f[x_1(t)]$$
$$y_2(t) = f[x_2(t)]$$

then by superposition: $y_1(t) + y_2(t) = f[x_1(t) + x_2(t)]$

One of the simplest ways to test the linearity of a system is to input a pure tone sine wave, and if, at all frequencies, the output is only a pure tone at the signal input frequency (i.e. no harmonics are produced), then the system is truely linear.

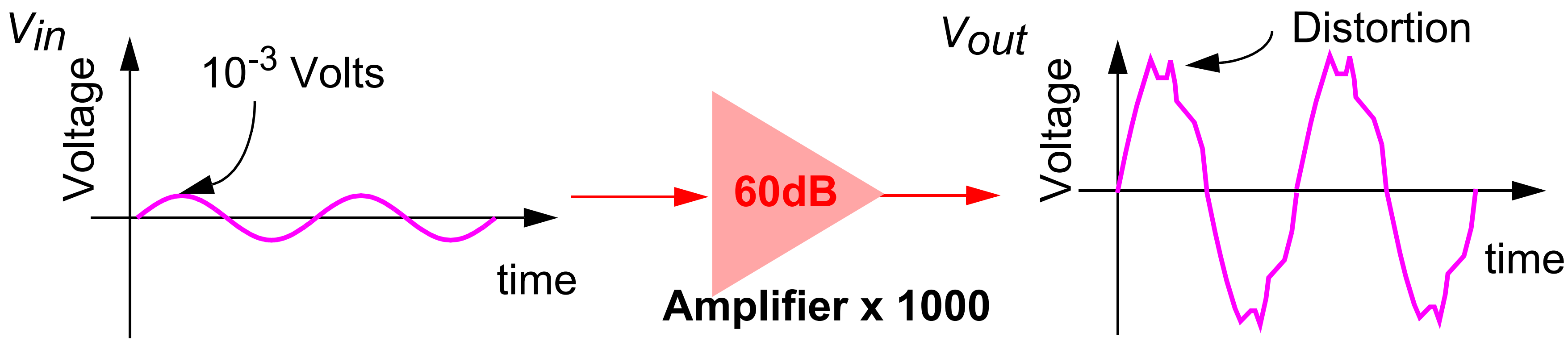
Amplification is often presented as a logarithmic measure of the power amplification ratio (P_{out}/P_{in}) given the large linear dymanic range. Recall that $P \propto V^2$, then:

$$A_{dB} = 10\log_{10}(P_{out}/P_{in}) = 10\log_{10}(V_{out}/V_{in})^2 = 20\log_{10}(V_{out}/V_{in}) \text{ decibels (dB)}$$

Therefore if an amplification $A = 1000$, then the power amplification is 60 dB. Similary an attenuation of a factor of 1000, (or a gain of 0.001) corresponds to -60dB of gain, or 60dB of attenuation! The placement of the -ve sign needs some care given the antonymit of the words gain and attenuation.

Amplifier Distortion

- An amplifier which introduces unwanted artifacts, is said to be non-linear and is, of course, very undesirable as it may mask signal components of interest.



- The above amplifier is non-linear and actually outputs the input signal plus a 3rd order harmonic:

$$V_{out} = (1000 \times V_{in}) + 10 \times (V_{in})^3$$

- Unlike noise it is essentially impossible to remove the effects of distortion. Therefore we try to minimize the possibility of distortion by using suitable components.

Notes:

Unlike in the above example, it is of course very unlikely that you will ever actually known the true non-linear equation of an amplifier. If you do, then you *might* be in a position to remove/address some of this non-linearity. However even the simplest non-linearity can be very difficult to remove. As an example, consider a system that adds a 2nd order harmonic according to the equation:

$$V_{out} = V_{in} + 0.001 V_{in}^2$$

Try to solve this equation for V_{in} as a function of V_{out}. ....not easy, and in fact there is not a unique solution. (Consider even the a simple squared equation:

$V_a = V_b^2$, then $V_b = \pm\sqrt{V_a}$ which is a non-unique solution.

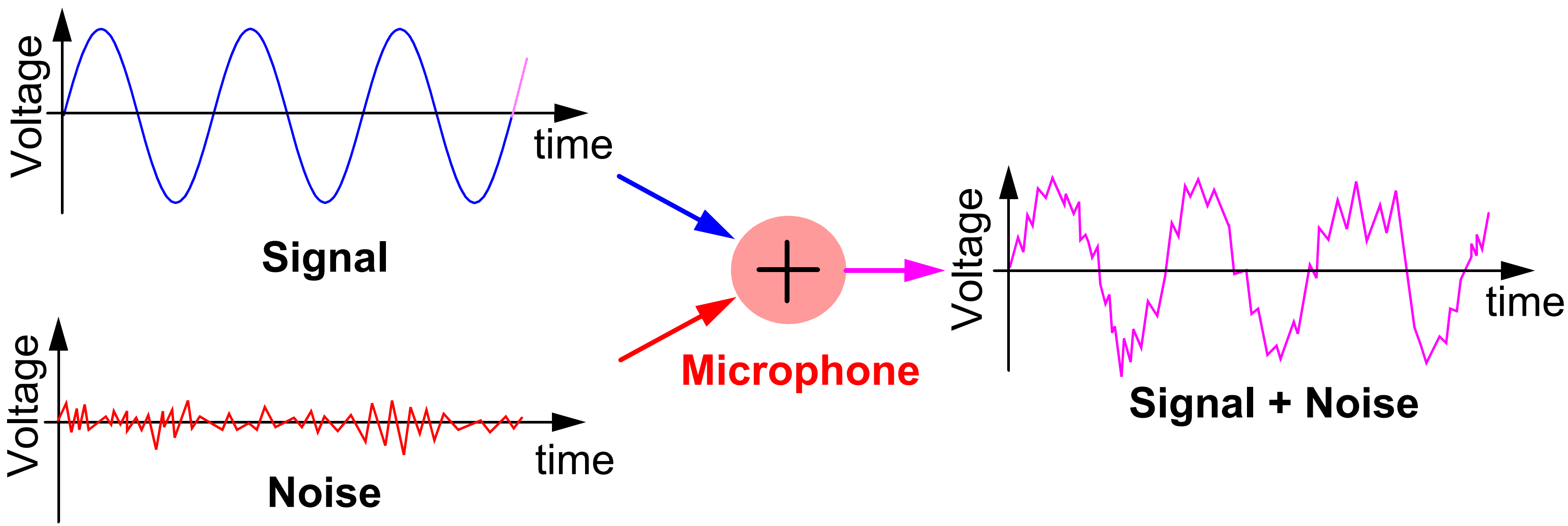
Non-linearities can cause serious problems in DSP by losing or masking desirable signal components.

It is probably true to say that every amplifier is non-linear to some extent. However if the power levels of the non-linear signal components are *very low*, then for pragmatic implementation purposes the amplifier can be considered linear; (the definition of "very low" of course depends on your application). In the above example where the non-linear second harmonic component has 1/5 of the voltage (1/25th of the power) of the fundamental, this is very high and the amplifier is at best "very" non-linear. Systems may often be classed as "weakly non-linear", "moderately non-linear", or "strong non-linear".

See system `c:\DSPedia\intro\non-linear_amplifer.svu` for implementation of the above slide and see `c:\DSPedia\intro\non-linear_amplifer_speech.svu` for a speech example.

Signals and Noise

- Most acquired signals are corrupted by some level of noise which can cause information to be lost.
- Signal processing techniques are often used in an attempt to remove or attenuate noise.
- Most noise can be considered as additive (linear superposition) which can be address by linear filtering techniques.



Notes:

One of the key strategies for DSP is too remove/attenuate noise from a signal of interest. There are some situations where it is very easy to remove the noise, for example in cases where the the signal and the noise are very disimilar with respect to some signal feature. If speech is corrupted by, say, a low rumbling single frequency, then removal is straightforward. However in situations where the signal of interest is very similar to the noise then things are less straightforward. One example where this type of problem occurs is the “cocktail party” noise effect. If a speech signal is corrupted by other speech signals, then extracting the desired signal can be very difficult. iIn general removing the noise from a signal requires that we know some fundamental characteristics about the signal and noise, i.e. the frequency range, typical power levels etc.

We should try to be clear about the difference between noise and distortion. Noise is usually the name given to an interfering signal, and which is usually in most cases additive noise. Using signal processing techniques we can address the effects of this noise and perhaps try and attenuate the noise using linear filtering or other techniques in order to improve the signal to noise ratio. Distortion on the other hand is caused by some non-linear process occuring in the signal acquisition or processing chain, and usually there is nothing that can be done to address distortion after it has occured.

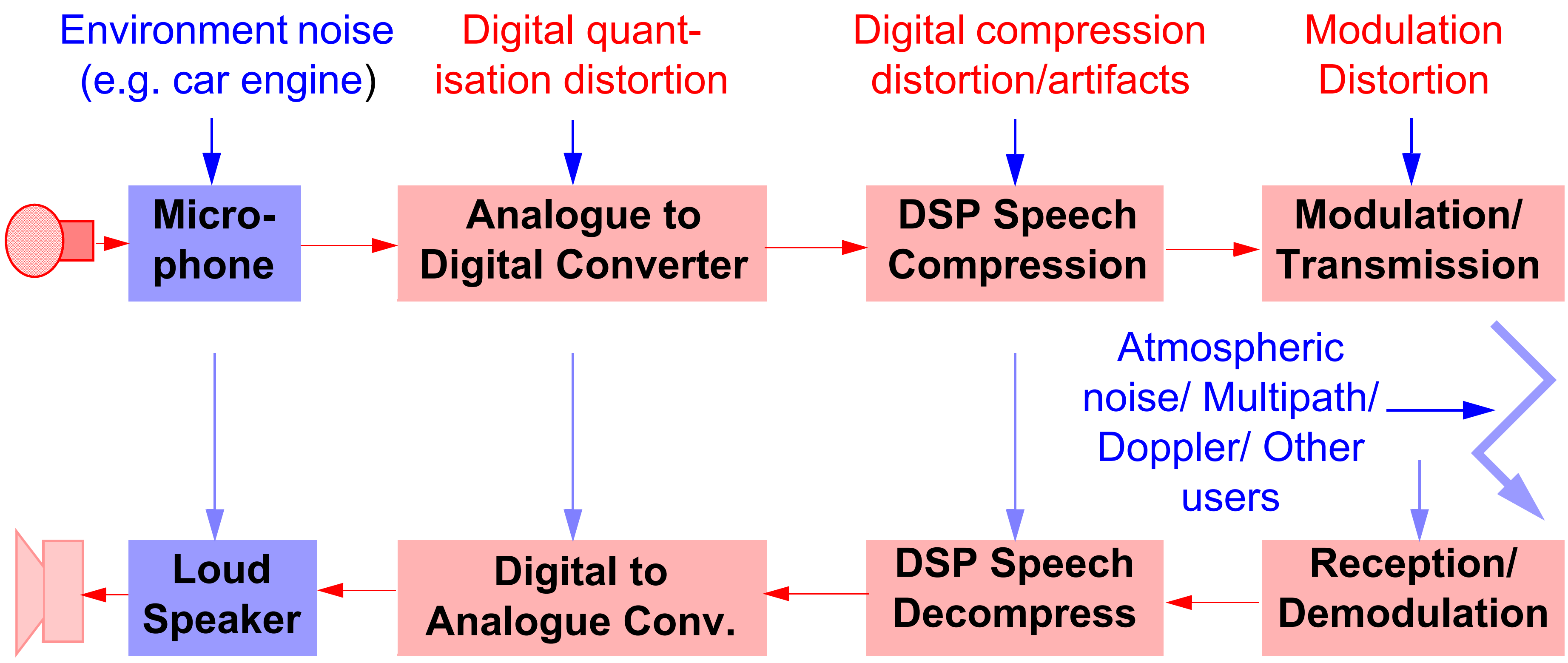
Additive Noise: Consider an acoustic signal source $s(t)$ received by a microphone which is corrupted by a nearby acoustic noise source $n(t)$. The composite signal recorded at the receiver is denoted $y(t)$. The simplest form of “additive noise” that could occur would be $y(t) = s(t) + n(t)$ A slightly more complex version of additive noise is $y(t) = s(t) + An(t)$ where the noise is attenuated by a factor of A perhaps because of the acoustic propogation path. Even more realistically the additive noise may take the form $y(t) = s(t) + An(t) + Bn(t - t_0)$ given that the noise may arrive at the receiver via a number of paths. And in its more general form, the received signal will take the form:

$$y(t) = s(t) + \int_0^{\infty} n(t)h(t - \tau)dt$$

where $h(t)$ is the “impulse response” of the acoustic path from noise source to receiver. Therefore although we may know exactly what the noise source is emitting, to “remove” this noise from $y(t)$ also requires that we have “information” about the acoustic transfer path.

The Noise/Distortion Chain....

- Consider the various levels of noise and distortion added in a digital mobile communications link:



- DSP must minimize the amount of noise/distortion input to the chain, and where possible attenuate other sources.

Notes:

Environment Noise: Noise coming from vehicle engine, wind noise etc. We may be able to address the noise at the microphone using DSP algorithms and techniques for linear or adaptive filtering. At the receiver, linear or adaptive filtering or perhaps active noise control could be introduced to improve the signal to noise ratio.

Quantisation Distortion: Quantisation distortion or noise is introduced by the analogue to digital converter (ADC). Ideally we use as many bits in the ADC converter as possible for quality, but as few bits as possible to keep bandwidth requirements low. To attempt to improve quality for a given low number of bits we may use psychoacoustic quantisation noise shaping, or dithering techniques.

Speech Coding/Compression Noise: To keep bandwidth requirements low we require to compress the signal. By compressing the signal we aim to maintain the signal quality and intelligibility but will accept some level of distortion or loss of fidelity.

Modulation: The modulation/demodulation process will introduce various levels of noise and distortion due to the various stages of modulation and filtering that are required.

Atmospheric/Multipath Noise: When the electromagnetic mobile signal is transmitted general signal reflection problems, interference from other users etc, will introduce a level of noise. Therefore it is desirable that the digital coding scheme is as resistant to noise as possible.

Signal to Noise Ratio

- Taking the logarithm of the linear signal power to noise power ratio (SNR) and multiplying by 10 gives the measure of decibels or dBs.

$$SNR = 10\log\frac{(\text{Signal Power})}{(\text{Noise Power})} = 10\log\frac{P_{\text{signal}}}{P_{\text{noise}}}$$

- Recalling that Power $\propto$ Volts^2, then:

$$SNR = 10\log\frac{V^2_{\text{signal}}}{V^2_{\text{noise}}} = 20\log\frac{V_{\text{signal}}}{V_{\text{noise}}}$$

SNR = 10dB, $P_{\text{signal}} = 10 \times P_{\text{noise}}$ and $V_{\text{signal}} = \sqrt{10} \times V_{\text{noise}}$
Very low quality telephone line

SNR = 60dB, $P_{\text{signal}} = 1000000 \times P_{\text{noise}}$ and $V_{\text{signal}} = 1000 \times V_{\text{noise}}$

Audio cassette deck

More....

Notes:

Note there are many forms of “decibel” and although all have specific definition, it is often the case that the meaning of dB is implied rather than explicity stated. For acoustic signals in particular there are a number of definitions of dB A, dBm, dB SPL, or dB HL. One of the most common uses of *dB is Sound Pressure Level* (*SPL*) is specified in decibels (dB) and is calculated as the logarithm of a ratio:

$$\text{SPL} = 10\log\left(\frac{I}{I_{\text{ref}}}\right)\text{dB}$$

where I is the sound intensity measured in Watts per square meter (W/m^2) and I_{ref} is the reference intensity of $10^{-12} W/m^2$ which is (or perhaps was!) the approximate lower threshold of hearing for a tone at 1000Hz. Alternatively (and more intuitively given the name *so*und “pressure” level) SPL can be expressed as a ratio of a measured sound pressure relative to a reference pressure, P_{ref}, of 2×10^{-5} N/m^2 = 20μPa:

$$\text{SPL} = 10\log\left(\frac{I}{I_{\text{ref}}}\right) = 10\log\left(\frac{P^2}{P_{\text{ref}}^2}\right) = 20\log\frac{P}{P_{\text{ref}}}\text{ dB}$$

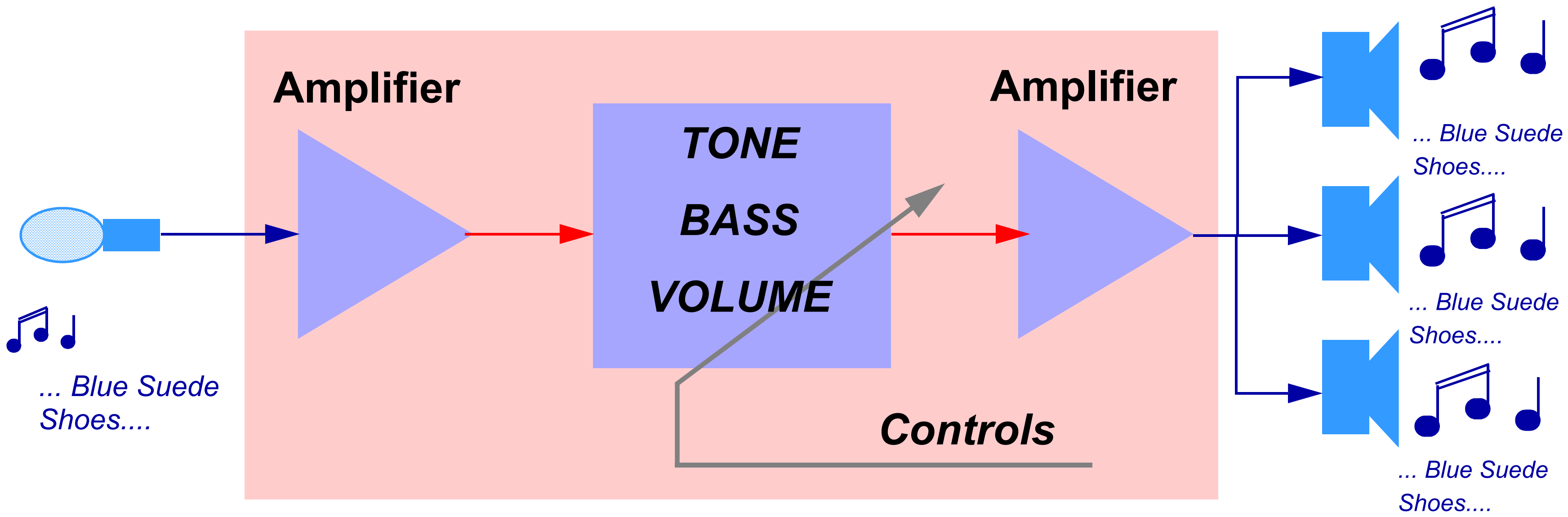
Intensity is proportional to the squared pressure. i.e. $I \propto P^2$. A logarithmic measure is used for sound because of the very large dynamic range of the human has a linear scale of more than 10^{12} and because of the logarithmic nature of hearing. Due to the nature of hearing, a 6dB increase in sound pressure level is not necessarily perceived as twice as loud in fact far from it. For example the difference in intensity between 110dB and 116dB seems much greater than the same difference between 40dB and 46dB. The threshold of pain is about 120dB. (See entry for *Sone*.)

It is worth noting that standard atmospheric pressure is around 101300 N/m^2 and the pressure exerted by a very small insect’s legs is around 10 N/m^2. Therefore the ear and other sound measuring devices are measuring extremely small variations on pressure.

Generic Analogue I/O Signal Processing

- In general an *analogue signal processing system* can be defined as a system that senses a signal to produce an analogue voltage, "process" this voltage, and reproduce the signal to its original analogue form.

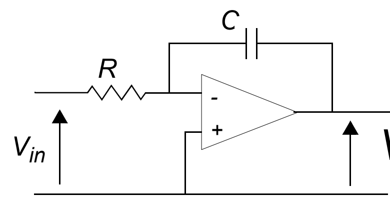
- A public address system is an example of an analogue signal processing system:



ANALOGUE SIGNAL PROCESSING

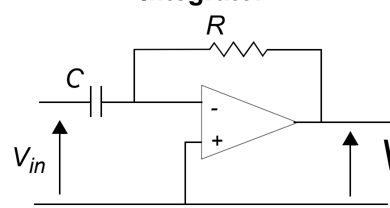
Notes:

Analogue systems have more than just the flexibility to amplify and filter. Before the availability of low cost, high performance DSP processors, analogue computers were used for analysis of signals and systems. The basic linear elements for analog computers were the summing amplifier, the integrator, and the differentiator. By the judicious use of resistor and capacitor values, and the input of appropriate signals, analogue computers could be used for solving differential equations, exponential and sine wave generation and the development of control system transfer functions.



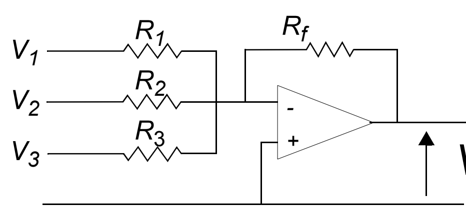
$$V_{out} = \frac{-1}{RC}\int_0^t V_{in}dt$$

Integrator



$$V_{out} = -RC\frac{dV_{in}}{dt}$$

Differentiator

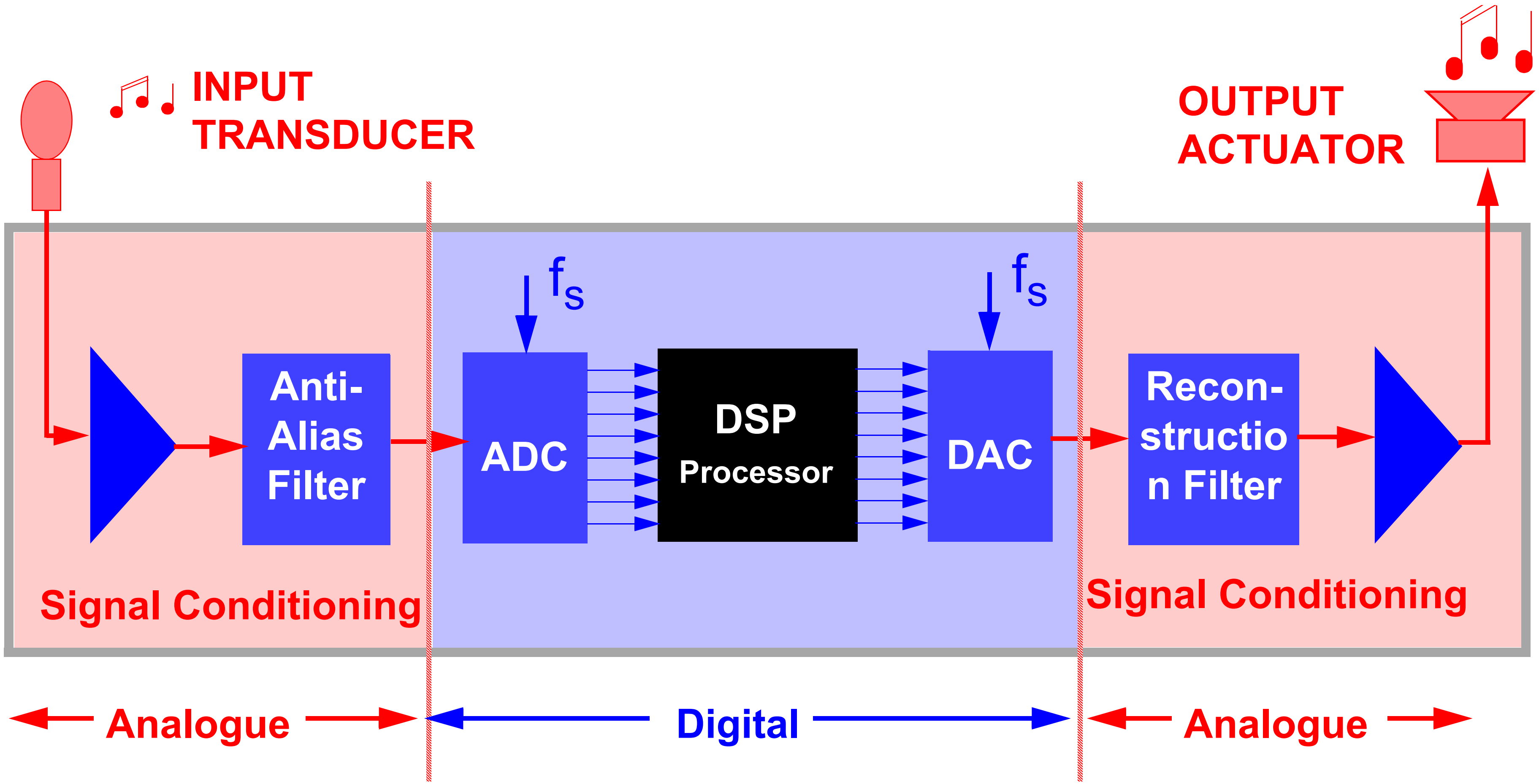


$$V_{out} = \frac{R_f}{R_1}V_1 + \frac{R_f}{R_2}V_1 + \frac{R_f}{R_3}V_1$$

Summer

A Generic Input/Output DSP System

- A single input, single output DSP system has the following components:



Notes:

Analogue components still play a very important part in DSP. In particular the input and output stages from and to the real world are of course analogue. Where possible however the design requirements and specification of the analogue components is being simplified in favour of more digital complexity, i.e. oversampling type strategies. Many systems will now use much higher than necessary sampling rates in order that the "complexity" of the analogue components can be reduced.

DSP systems can be classed as one of three general types:

(1) Real time Input/Output, e.g. a DSP enabled communications link;
(2) Real time Input only, e.g a speech recognition system;
(3) Real time Output only, e.g. a CD audio reproduction system.

The anti-alias filter is important in order to ensure that aliasing distortion is not introduced to the DSP system. The reconstruction filter is important to ensure that reconstruction high frequency noise is not present in the output signal.

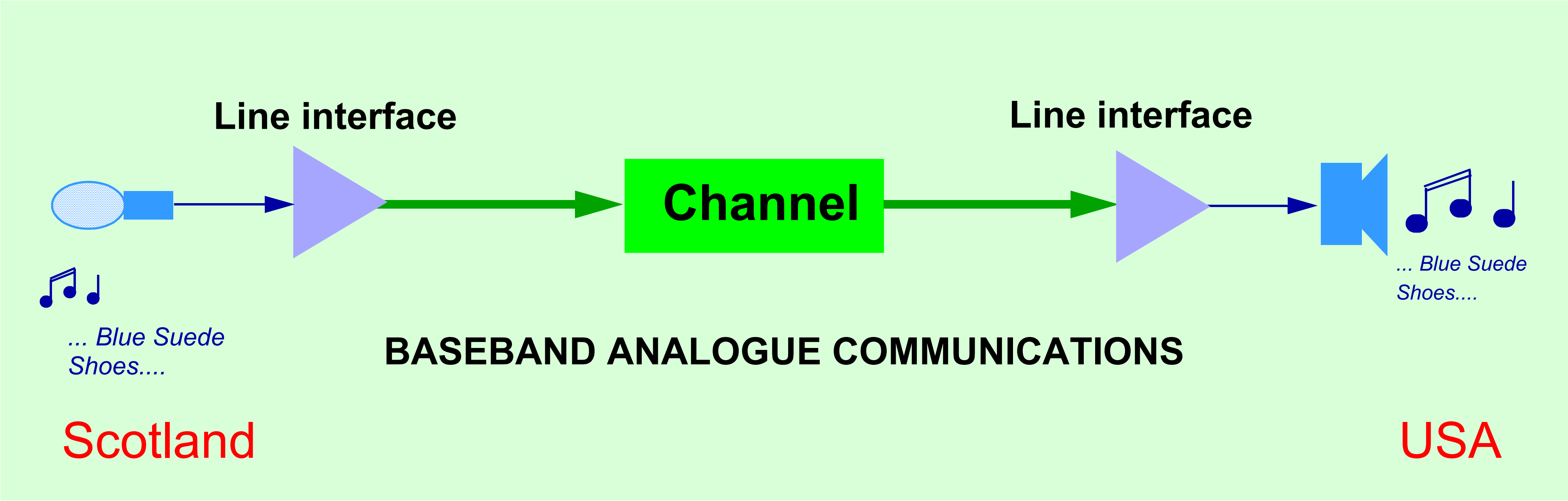
In a DSP system the analogue voltage is converted from a real world signal, to a voltage, to binary numerical values. DSP systems use binary numbers, or base 2 and usually with 2's complement (to allow +ve and -ve number representation.) Base 2 is used because it is easy to design electronic devices that only have two digits corresponding to two voltage levels. Binary adders, multipliers, memory elements and so on are widely available and together form the core element of every computer's ability to perform high speed arithmetic. Every DSP microprocessor/ASIC uses binary arithmetic. 2's complement arithmetic used by most DSP processors which allows a very convenient way of representing negative numbers, and imposes no overhead on arithmetic operations. In two's complement the most significant bit is given a negative weighting, e.g. for the 16 bit number.

$$1001\ 0000\ 0000\ 0001_2 = -2^{15} + 2^{12} + 2^{1}$$

$$= -32768 + 4096 + 1 = -28671$$

Generic Analogue Communications

- For most baseband telecommunications a voltage signal is transmitted over a cable.



- A simple example is a ***telephone***. The acoustic signal is converted to a voltage which is then directly transmitted over a twisted pair of wires to be received at a remote location.

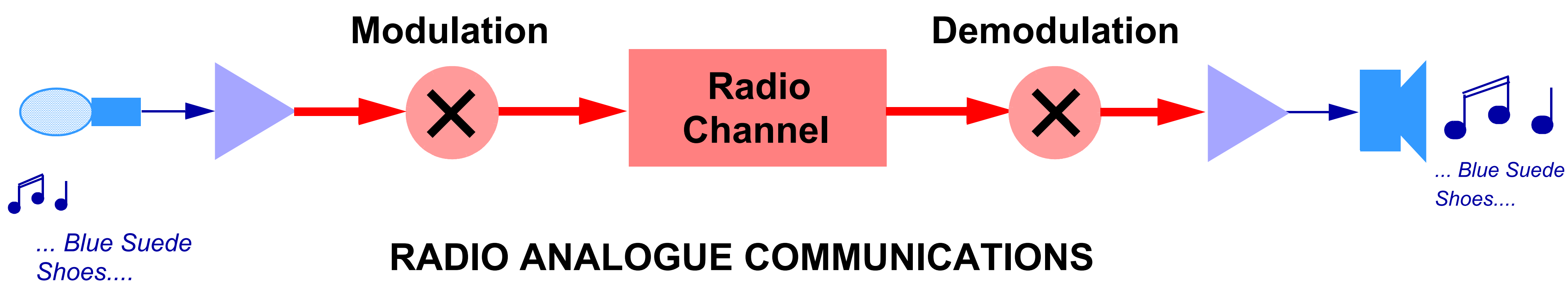
Notes:

In a telephone system the line interface is some form of driver/amplifier that is capable of converting the analogue voltage of the speech into a signal with sufficient power to be transmitted from transmitter to receiver (or transmitted to an intermediate point such as an exchange for subsequent switching).

Generic Analogue Radio Communication

For radio based analogue communications a modulator is required to convert from voltage to an electromagnetic radio frequency (RF) signal.

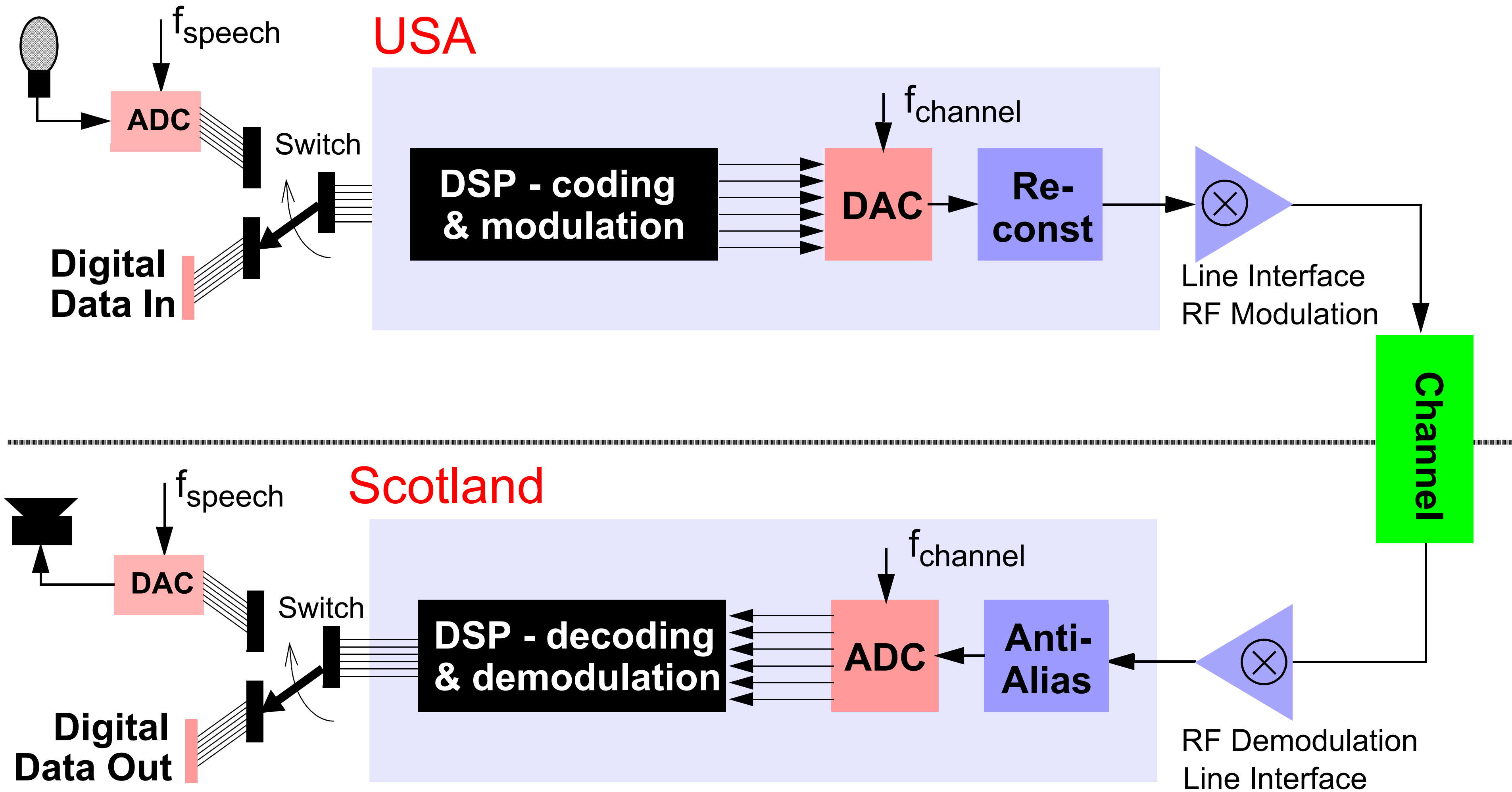
An example of (one way) analogue communication is an FM radio station (modulation to around 100MHz) or first generation mobile telephones which had 30kHz of bandwidth available for a speech channel.



RADIO ANALOGUE COMMUNICATIONS

Digital Data Communications

- Modern communications systems require that digital information is transmitted and received.



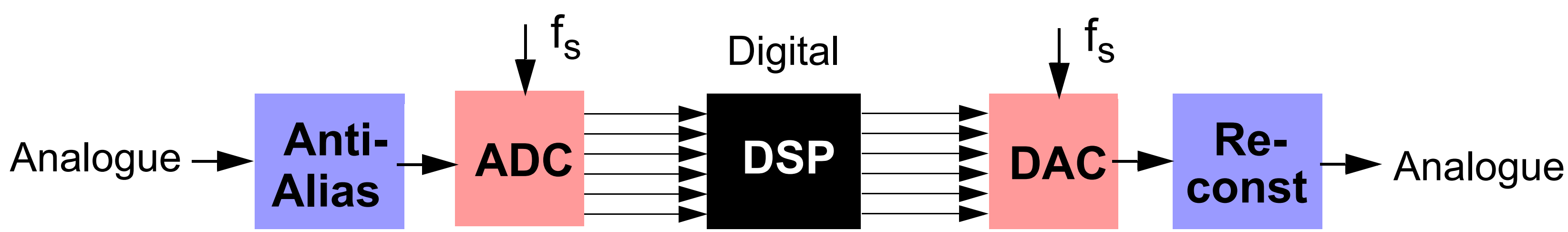
Notes:

Data communications requires that digital information (1's and 0's) is coded/modulated to produce an analogue signal that can be transmitted over a cable. If the information is to be sent over radio, then a transmitter and receiver is also required.

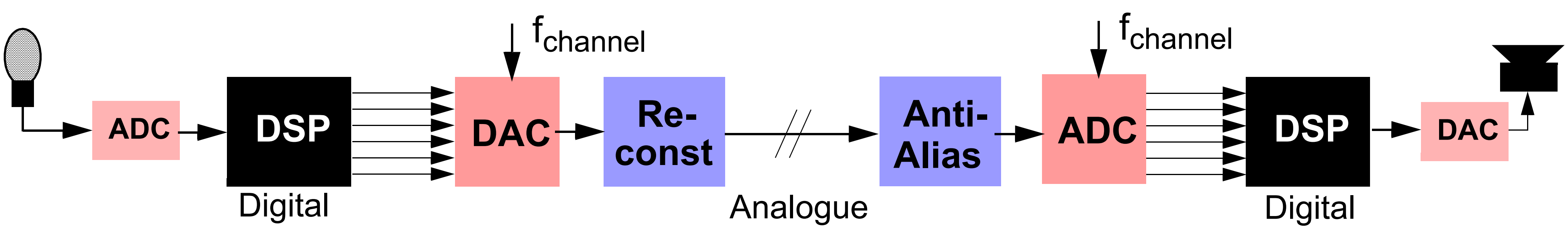
The digital coding/modulation may compress or coded the data in some way, then translate into QPSK or other QAM format. A final analogue signal is then produced by the DAC which is then suitable for sending directly over a cable, or it can be modulated by a carrier frequency (RF (radio frequency) or centre of the baseband). The sampling rate $f_{channel}$ is at least twice the channel bandwidth to satisfy the Nyquist sampling criteria.

At the receive side, the analogue signal is demodulated from RF, or otherwise, and then converted to digital by the ADC prior to the final stage of decoding/demodulation.

Note that with the ***generic input/output DSP system*** we have the signal "*domain*" sequence: *analogue-digital-analogue:*

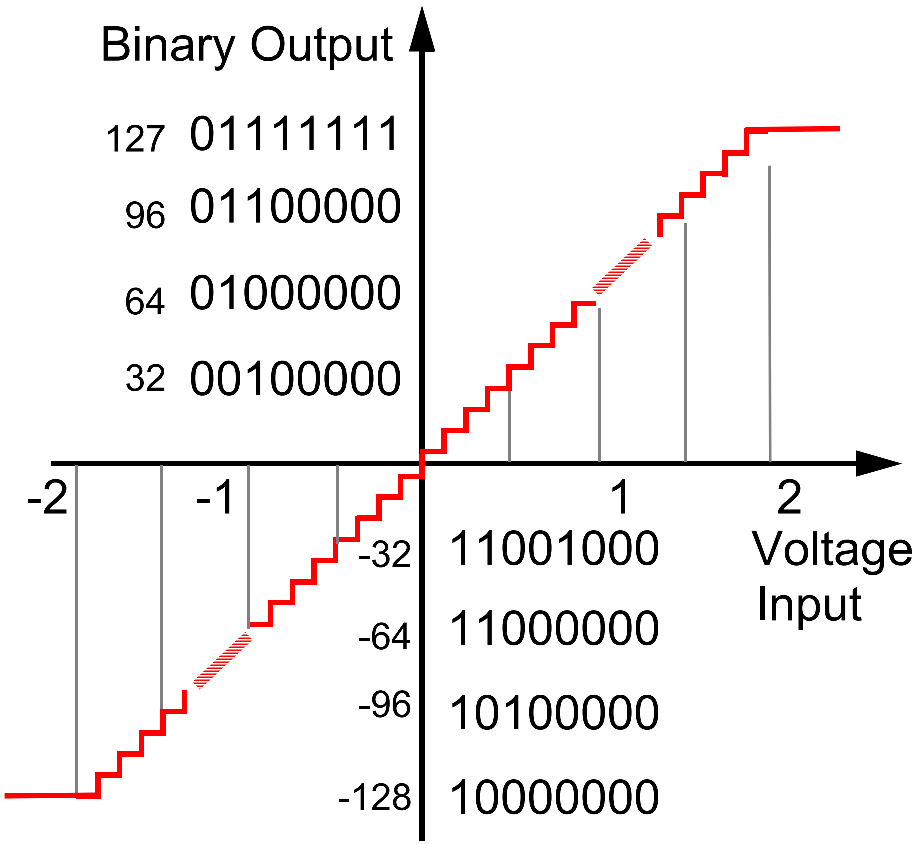


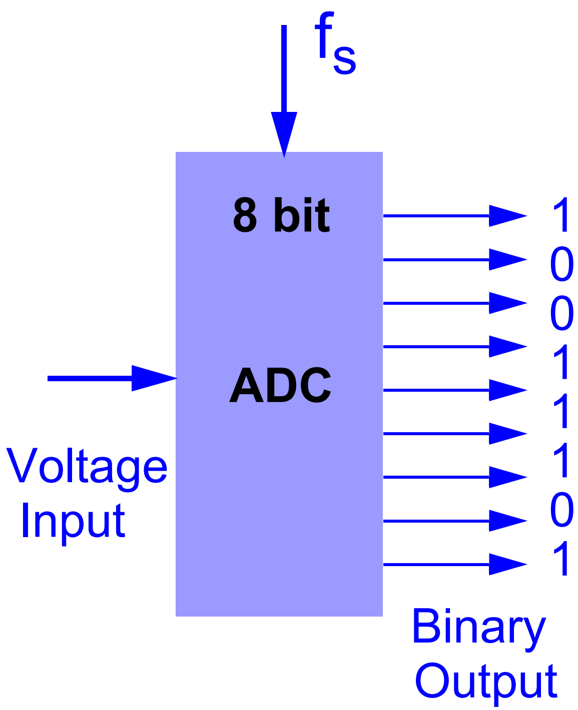
whereas with the ***DSP communications system*** we have the signal "*domain*" sequence: *digital-analogue-digital:*



Analogue to Digital Converter (ADC)

- An ADC is a device that can convert a voltage to a binary number, according to its specific input-output characteristic.





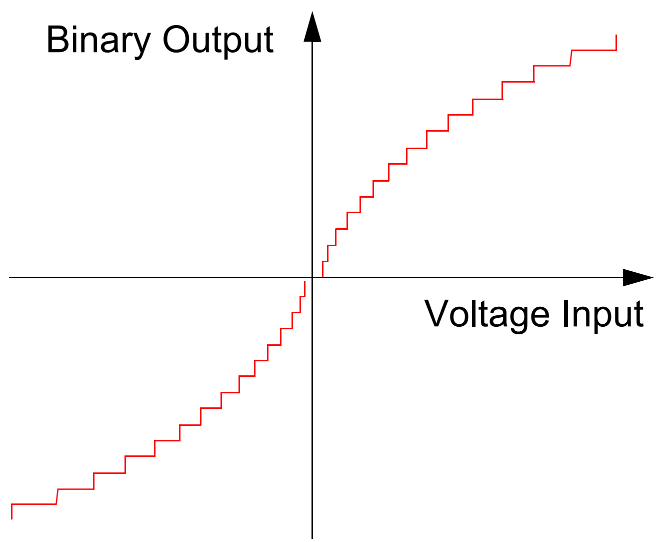
Generic DSP

- The number of digital samples converted per second is defined by the sampling rate of the converter, f_s Hz.

Notes:

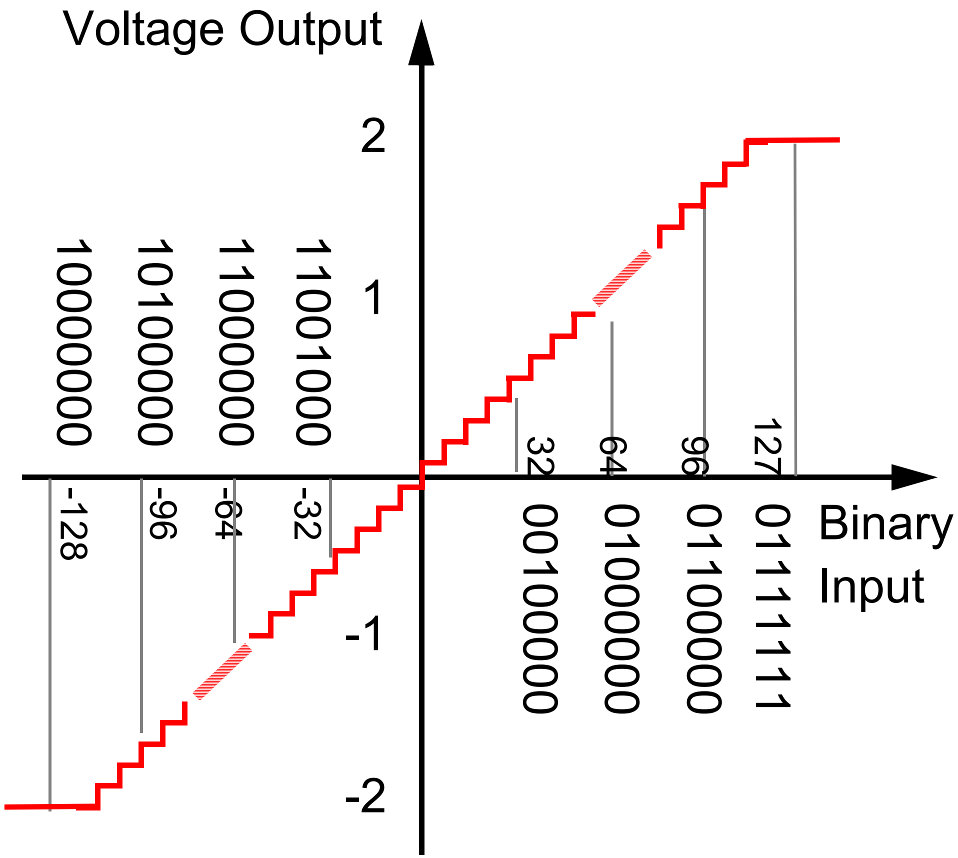
Viewing the straighline portion of the device e are tempted to refer to the characteristic as “linear”. However a quick consideration clearly shows that the device is non-linear (recall the definition of a linear system from before) as a result of the discrete (staircase) steps, and also that the device clips above and below the maximum and minumum voltage swings. However if the step sizes are small and the number of steps large, then we are tempted to call the device “***piecewise linear over its normal operating range***”.

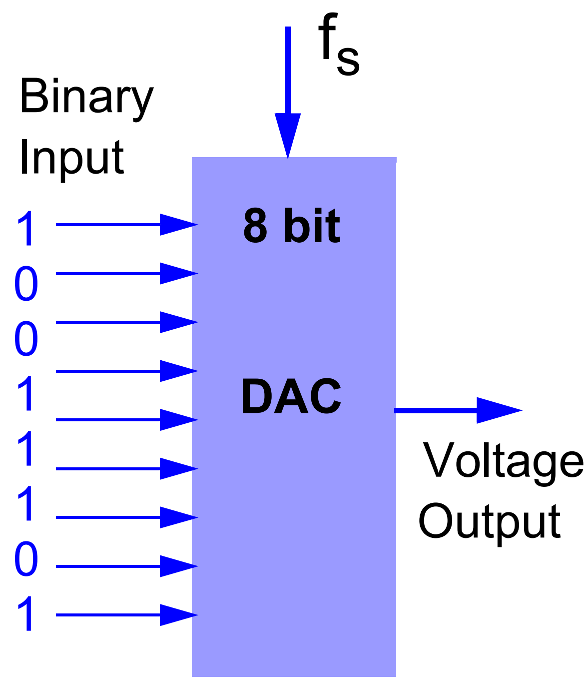
Note that the ADC does not necessarily have a linear (straight line) characteristic. In telecoms for example a defined standard nonlinear quantizer characteristic is often used (A-law and μ-law). Speech signals, for example, have a very wide dynamic range: Harsh “oh” and “b” type sounds have a large amplitude, whereas softer sounds such as “sh” have small amplitudes. If a uniform quantization scheme were used then although the loud sounds would be represented adequately the quieter sounds may fall below the threshold of the LSB and therefore be quantized to zero and the information lost. Therefore non-linear quantizers are used such that the quantization level at low input levels is much smaller than for higher level signals. A-law quantizers are often implemented by using a nonlinear circuit followed by a uniform quantizer. Two schemes are widely in use: the A-law in Europe, and the μ-law in the USA and Japan. Similarly for the DAC can have a non-linear characteristic..



Digital to Analogue Converter (DAC)

- A DAC is a device that can convert binary numbers to voltages, according to its specific input-output characteristic.





Notes:

Type of ADCs and DACs

ADCs (Quantizer):

Successive Approximation ADC - Uses a DAC and comparator to determine voltage

Dual Slope ADC - Uses a capacitor connected to a reference voltage, and time taken for capacitor to charge is counted by a digital counter.

Flash ADC - Accurately trimmed ladder of resistors.

Sigma-Delta - Oversampling single bit converter

DACs:

Multiplying DACs, accurately trimmed resistor is used to generate the output voltage via summing amplifier.

Sigma-Delta DAC - Single bit oversampled data

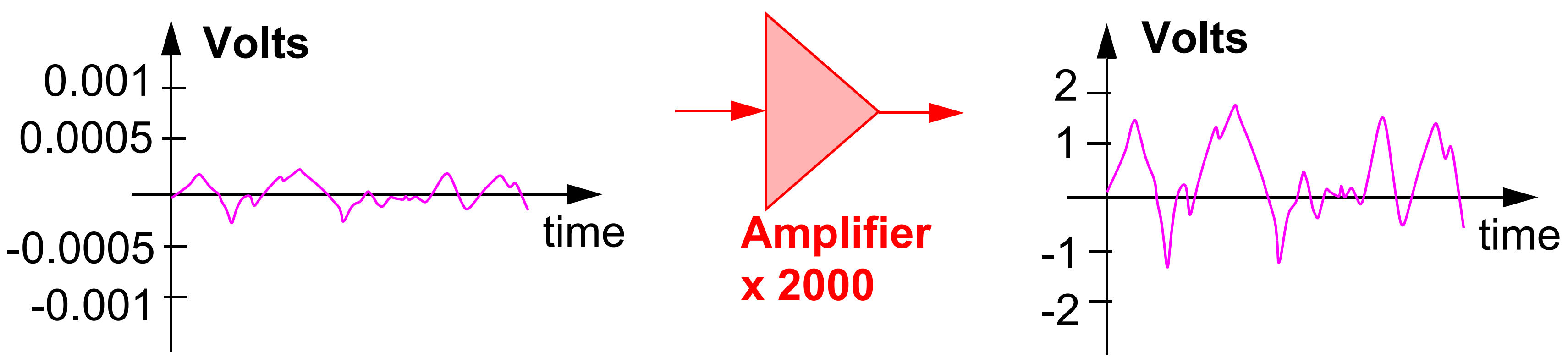
In this course we will specifically study different type of ADC or DAC. We will use them as functional components.

Because of the DSP processing within sigma delta devices (oversampling and noise shaping), later in the course we will look at sigma delta in more detail.

Signal Conditioning

- Note that prior to a signal being input to an ADC, an amplifier will be required to ensure that the full voltage range of the ADC is used - this is referred to as signal conditioning.

- For the above ADC with a maximum input and output of 2 volts we would require that the input signal to the ADC has a similar range:

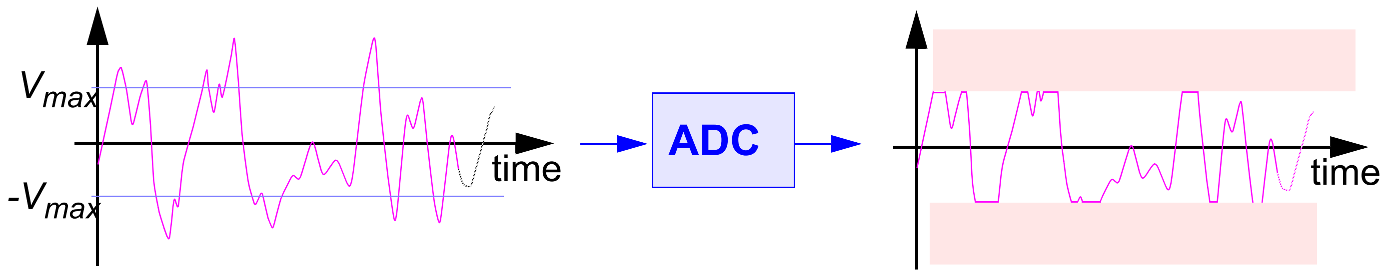


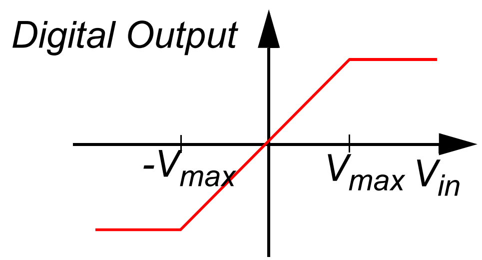
Generic DSP

•Depending on the output actuator, an amplifier, or at least a buffer amplifier will be required.

Notes:

An analog signal with a magnitude larger than the upper and lower bounds $\pm V_{max}$ of an ADC chip, will be clipped. Any voltage above V_{max} will be clipped and the information lost. Clipping effects frequently occur in amplifiers when the amplification of the input signal results in a value greater than the power rail voltages.





The DSP designer must ensure that for the particular application, clipping does not in general occur.

Sampling

- The speed at which an ADC generates binary numbers is called the sampling rate or sampling frequency, f_s.

- The time between samples is called the sampling period, t_s:

$$t_s = \frac{1}{f_s}$$

- Sampling frequency is quoted in samples per second, or simply as Hertz (Hz).

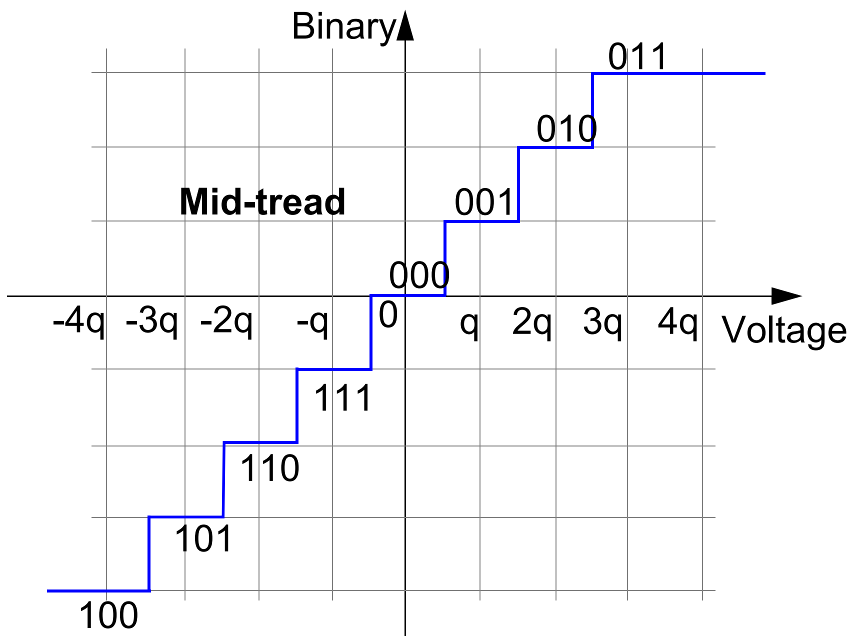
- The actual sampling rate will depend on parameters of the application. This may vary from:

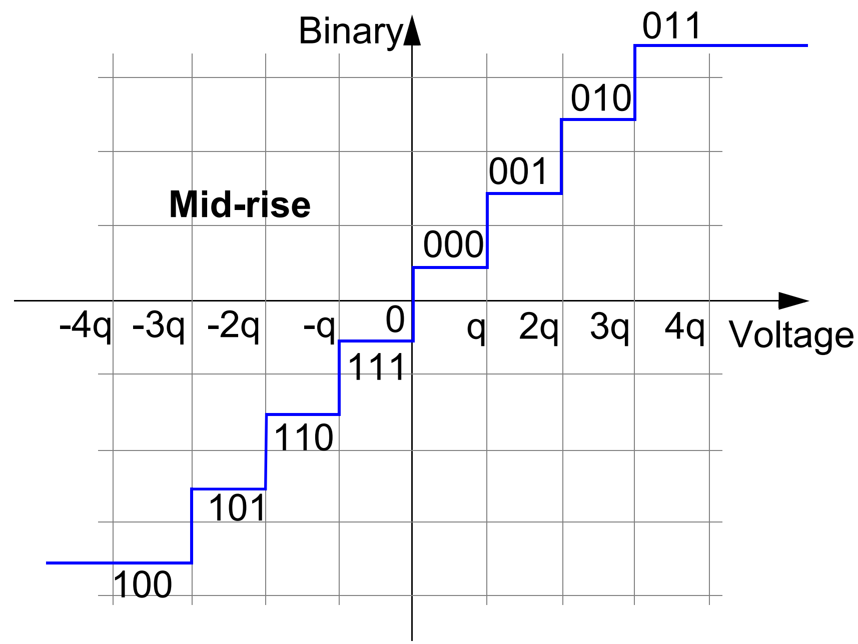
10's of Hz for control systems,
100's of Hz for biomedical,
1000's of Hz for audio applications,
1,000,000's of Hz for digital radio front ends.

Generic DSP

Notes:

Note that the ADC may or may not have a zero output voltage level. For example an ADC/quantizer could have a mid-tread or a mid-rise characteristic. Consider a 3 bit converter for the mid-tread and the mid-riser:





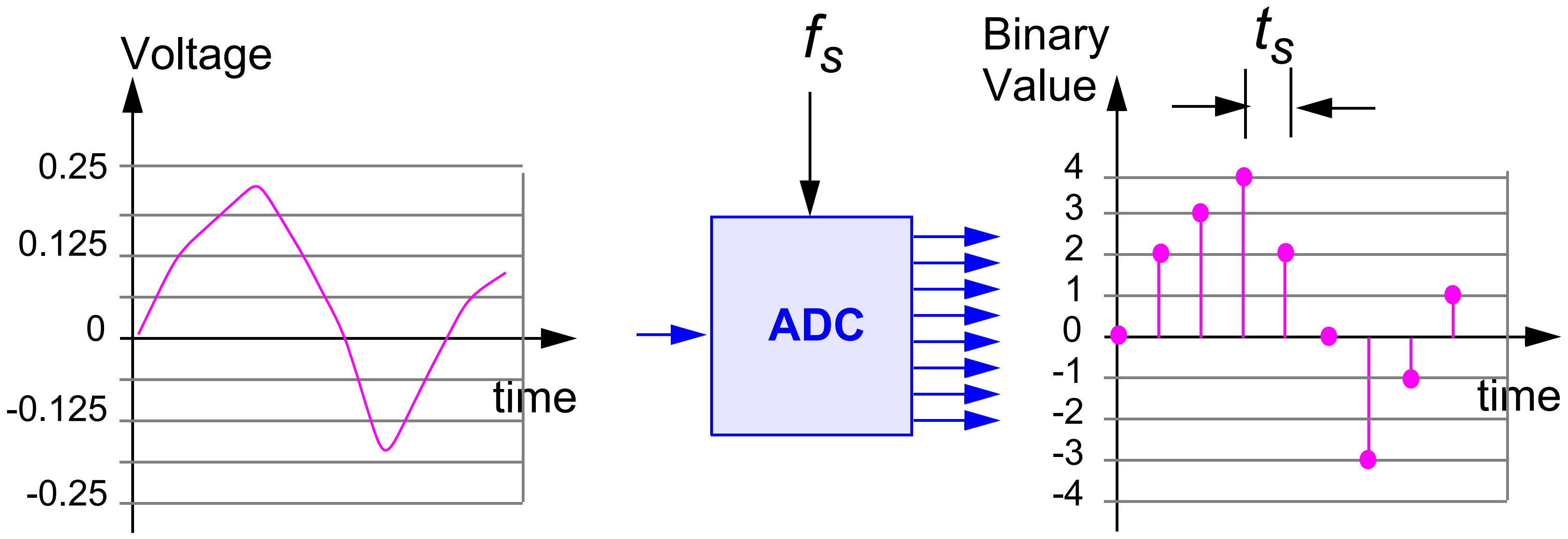
Note that the mid-riser does not have a zero output level, whereas the mid-tread does have a zero output level but there are more levels above the zero level than there are below, this being a feature of two's complement arithmetic.

For an ADC with a very low number of bits, the differences between the mid-tread and mid-rise quantizer may be noticeable, particularly in terms of the perceived quantisation noise. For example if a very small sine wave of amplitude $q/10$ was input to the above mid-rise quantiser, then the output would always be zero, 000. However inputting the same waveform to the midtread would produce a square wave of levels 000 and 111 at the same frequency as the sine wave. Hence the mid-tread registers something of the input signal, but the mid-rise has not!

Sampling an Analogue Signal

- After signal conditioning the ADC can produce binary number equivalents of the input voltage.

- If the ADC has finite precision due to a limited no. of discrete levels then there may be a “small” error associated with each sample.

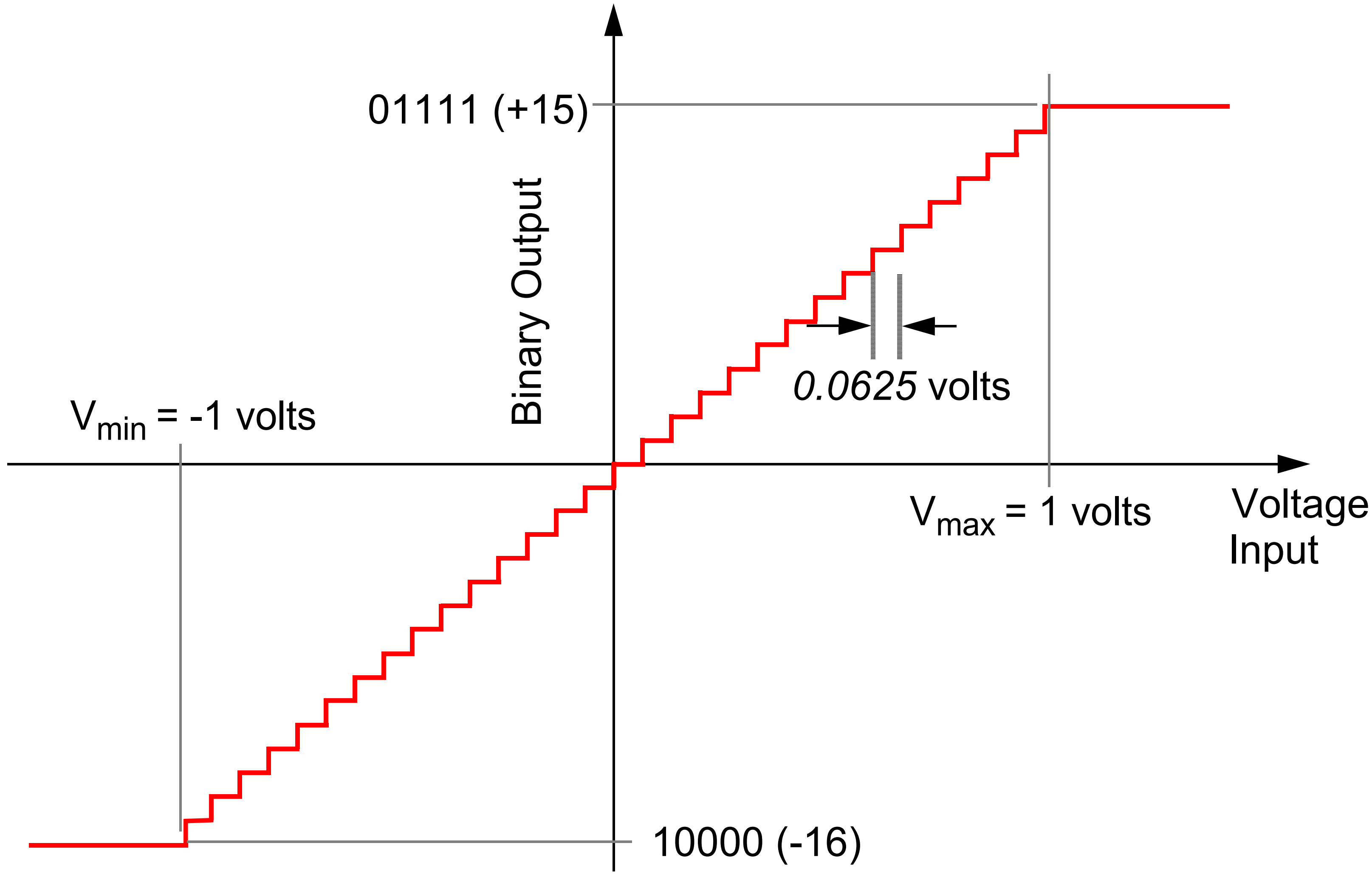


- The quantisation step size is 0.0625 volts. If an 5 bit ADC is used, then the max/min voltage input is approx 0.0625 x 16 = 1 volt.

Generic DSP

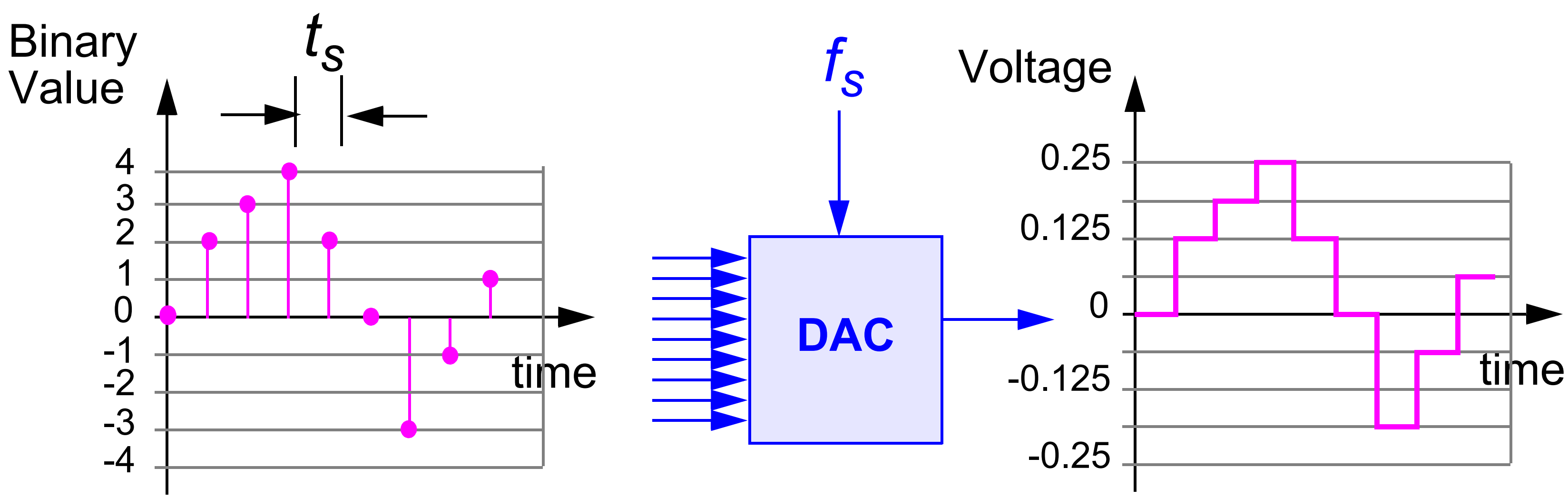
Notes:

For example purposes, we can assume our ADC or quantiser has 5 bits of resolution and maximum/minimum voltage swing of +1 and -1 volts. The input/output characteristic is shown below:



Reproducing an Analogue Signal

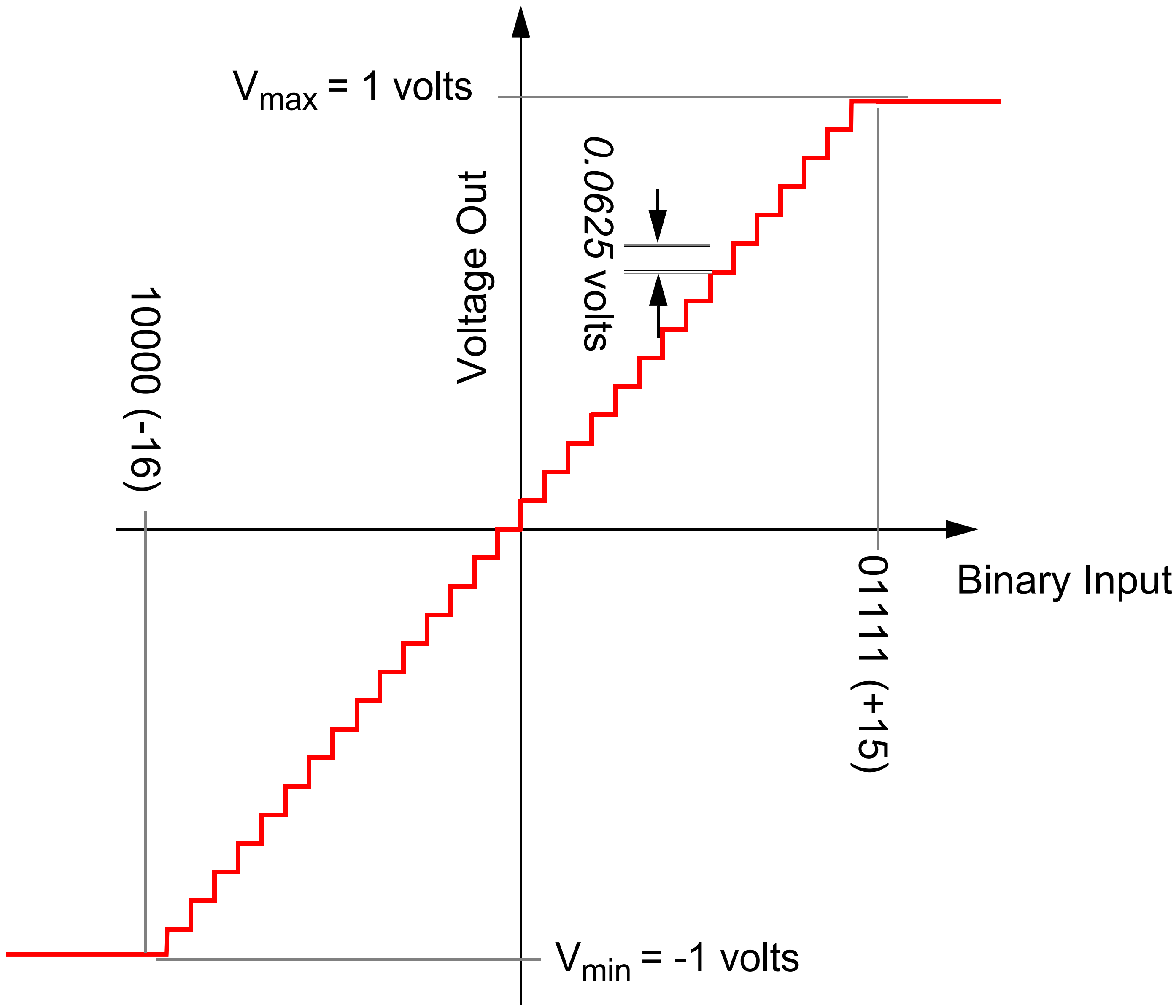
- Using a DAC at an appropriate sampling rate, we can reproduce an analogue signal:



- Note that the output is a little "*steppy*" caused by the zero order hold (step reconstruction);
  ....this artifact can however be removed with a reconstruction filter.

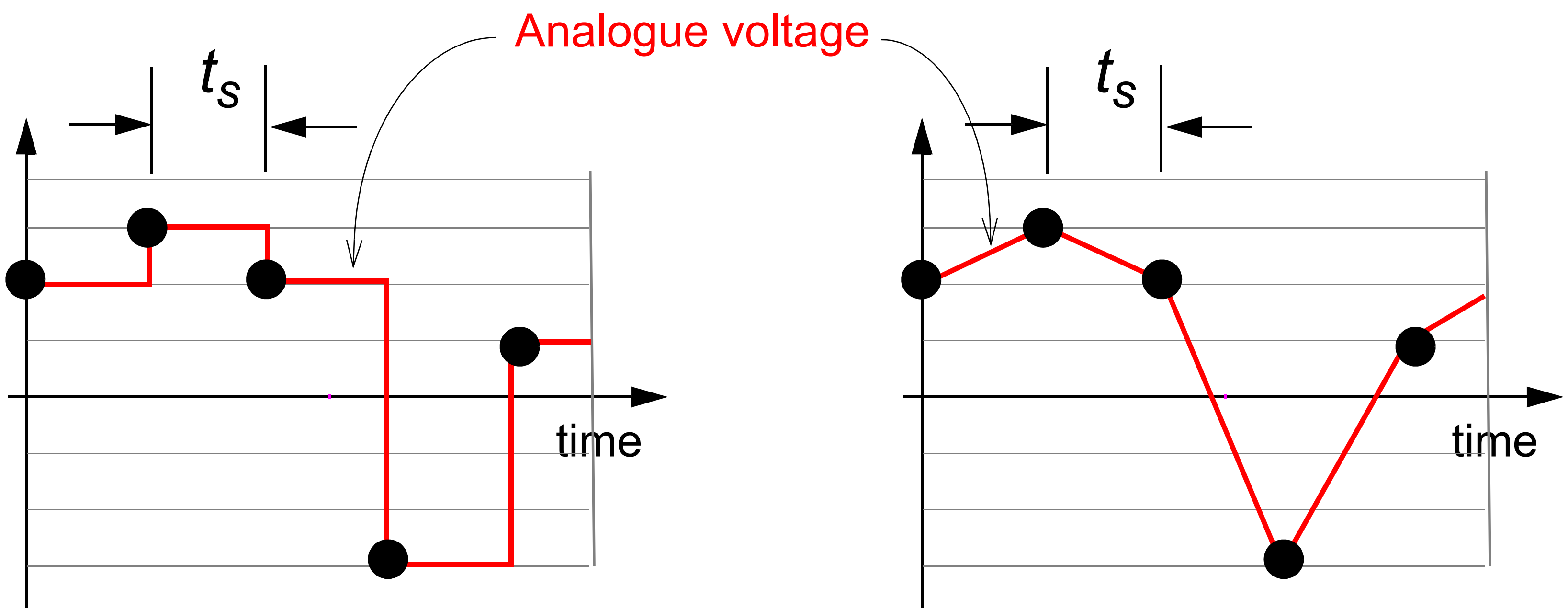
Notes:

For example purposes, we can assume our DAC or quantiser has 5 bits of resolution and maximum/minimum voltage output swing of +1 and -1 volts. The input/output characteristic is shown below:



First Order Hold

- Alternatively a first order hold could be attempted in the DAC. Here the voltage between two discrete samples is approximated by a straight line.



Zero Order Hold

First Order Hold

- A first order apparently produces a “more accurate” reproduction of the analogue signal. However implementation of a circuit to perform interpolation is not trivial and turns out not to be necessary.

Notes:

A zero order circuit is essentially a form of capacitive element whereby the input voltage is held (almost!) constant for one sampling period. This is a straightforward and low cost circuit to build and implement.

A first order hold voltage reconstruction does in fact produce a signal that is closer to the original in a mean squared error sense. However building an electronic circuit that will generate a linearly increasing voltage between two arbitrary input voltages is not trivial.

Hence we favour zero order hold. We can further note that the “problem” of the zero order hold (i.e the squareness of the output voltage) can be corrected for later using a reconstruction and a sinx/x compensating filter.

Binary Data Wordlengths

- Data wordlengths for DSP applications, typically:

Fixed Point Wordlengths: *Dynamic Range*

8 bits	–128	to	+127	$20\log 2^{8} \approx 48$ dB
16 bits	–32768	to	+32767	$20\log 2^{16} \approx 96$ dB
24 bits	–8388608	to	+8388607	$20\log 2^{24} \approx 154$ dB

Floating Point Wordlengths (for arithmetic only):

- **32 bits** (-10^{38} to $+10^{38}$)
  (24 bit mantissa, 8 bit exponent)

- Note that data input from an ADC, or output to a DAC will always be fixed point, although the internal DSP computation may be floating point.

Notes:

Not all applications will use ADCs or DACs that are 8, 16, or 24 bits. Some applications such as digital communications typically use between 10 and 12 bits linear quantising ADCs, and thereafter use DSP processors that use 16 bit wordlengths.

DSP processors used to process the data will use data wordlengths of 8 (more a microcontroller), 16, 24 or 32 bits. 32 bits is usually floating point, 24 bit mantissa, 8 bit exponent.

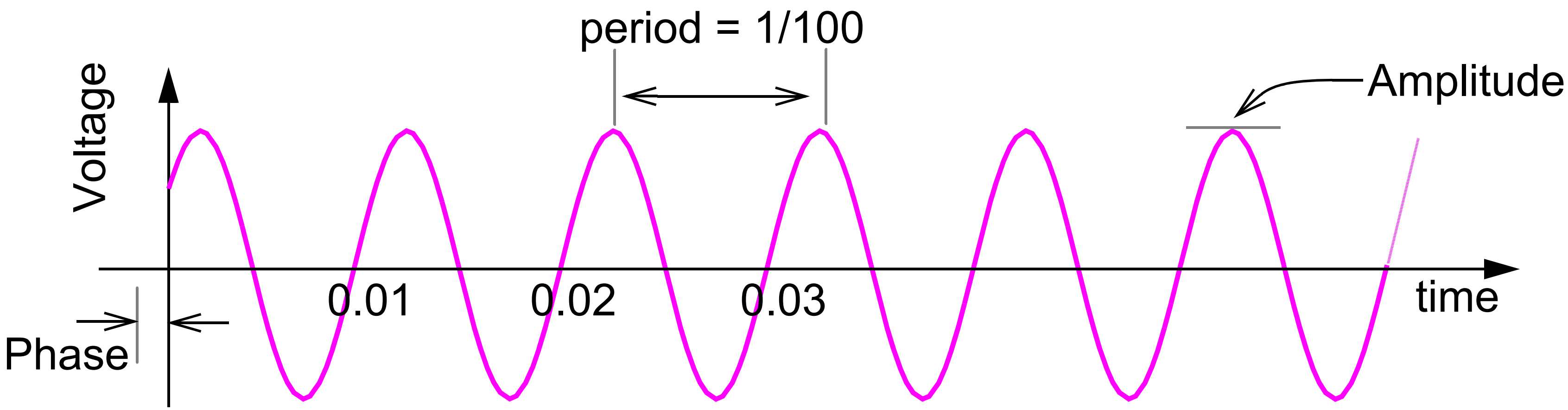
It is worth pointing out at this stage that the no. of bits resolution of an ADC or DAC can be increased by using oversampling techniques. For example a 3 bit converter, could be "oversampled" to produce a sampled signal that has resolution of 8 bits! Oversampling techniques will be discussed later in the course.

In general the arithmetic "modes" of DSP processors are straightforward unlike general purpose processors. In a general purpose processor, arithmetic modes of 16 bit integer, 32 bit integer, 32 bit floating point, 64 bit floating point etc are likely to be available. This is not the case for DSP processors. Although various arithmetic modes could be programmed, in general the mode is fixed point, where the sign bit is the MSB (most significant bit) and a binary point is placed after the MSB. Therefore for a 16 bit processor which has an integer numerical range of -32768 to 32767, the numerical range with the binary point at position 15 is -1 to 0.9999, i.e. $1.000\ 0000\ 0000\ 0000$ to $0.111\ 1111\ 1111\ 1111$.

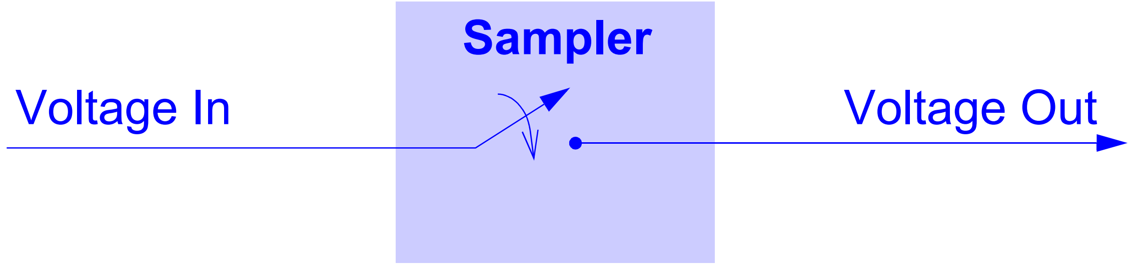
Binary data formats are discussed later in the course.

Sampling - How Fast?

- To intuitively derive the sampling theorem, consider first a pure sine wave of frequency 100Hz:



- In order to ensure that we retain all of the information in the signal what sampling rate should be used? (*no* quantisation)



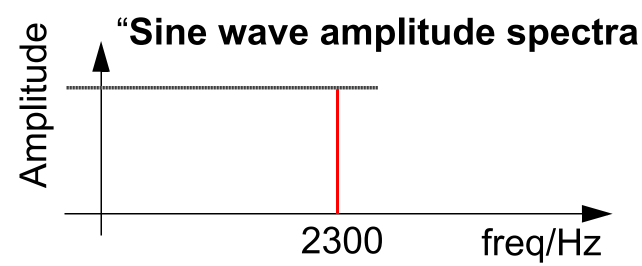
Generic DSP

Notes:

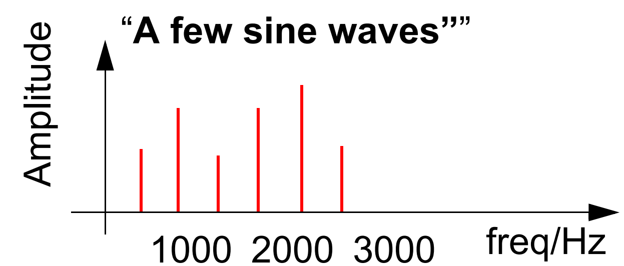
Note that the sampling being performed in this slide does ***not*** quantise the signal. We are assuming the existence of a sampler only at this stage.

In sampling this signal, we are clearly trying to ensure that we retain the information regarding the signal's amplitude, frequency and phase, the three components which characterise the signal.

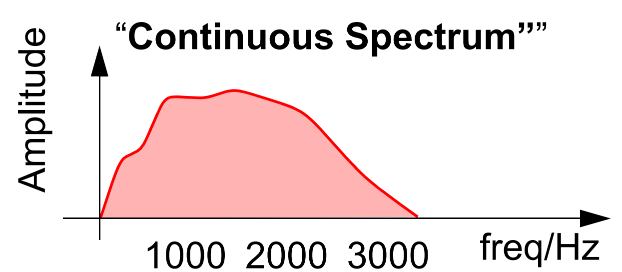
As an alternative to representing a sine wave as an equation, e.g. $y(t) = 10\sin(2\pi 2300t + \theta)$ for a 2300 Hz sine wave of amplitude 10 volts and phase θ we can use a simple frequency domain representation of a (sine wave amplitude) spectrum. (Note that this simple spectrum does not represent phase information.)



If a signal is composed of a "few sine waves" then the spectrum may be represented as illustrated, and if the signal is somewhat more aperiodic and we can only identify the "average" or typical frequency content, then we may use a continuous type spectrum

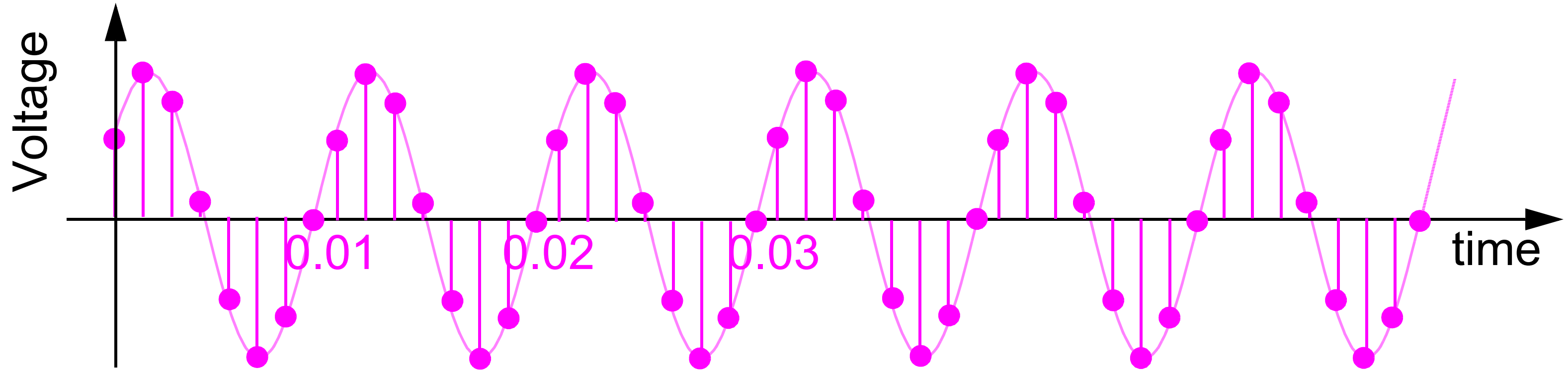


In effect the sine wave is the simplest type of time varying signal that exists, and ANY real signal can be produced from a sum of suitable sine and cosine waves. Later in the course when we discuss Fourier analysis it will be shown that any periodic signal can be decomposed into a sum of sine waves of suitable frequencies, amplitudes and phases. These frequency components actually form a mathematical basis for describing the periodic signal. Hence in attempting to derive a sampling theorem it is entirely appropriate that we start by finding a suitable rate for a fundamental single sinusoidal wave, and extend the theorem from this point.



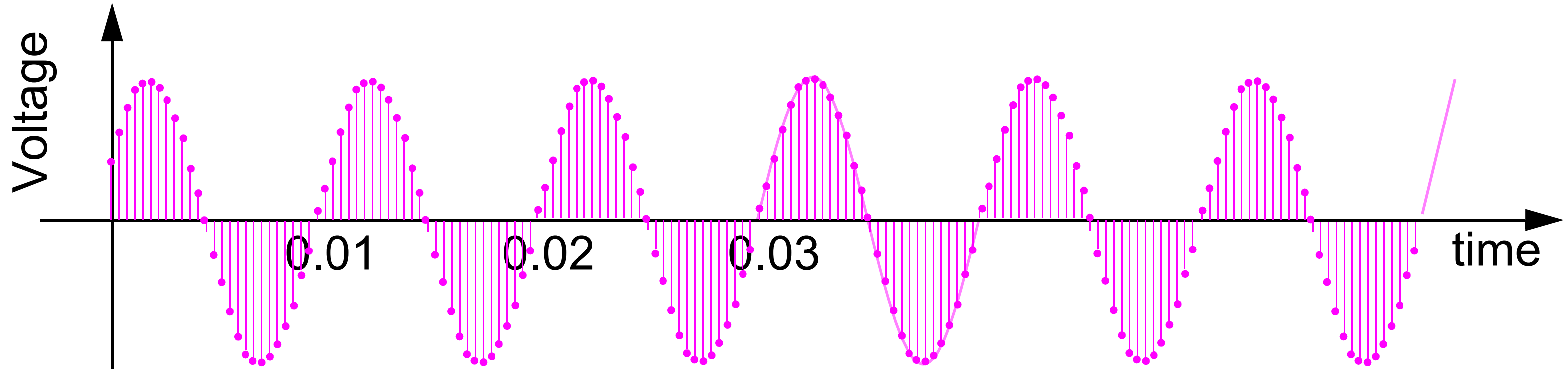
Sampling - Too Fast?

- Sampling at f_s = 800Hz, i.e. 8 samples per period:



Appears to be a "reasonable" sampling rate.

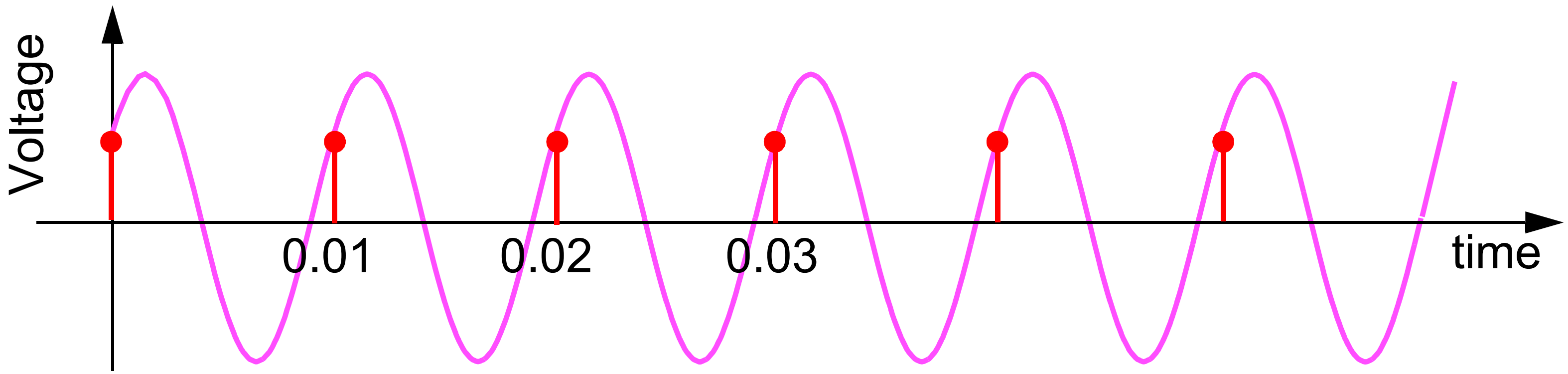
- Sampling at f_s = 3000Hz, i.e. 30 samples per period:



Perhaps higher than necessary sampling rate?

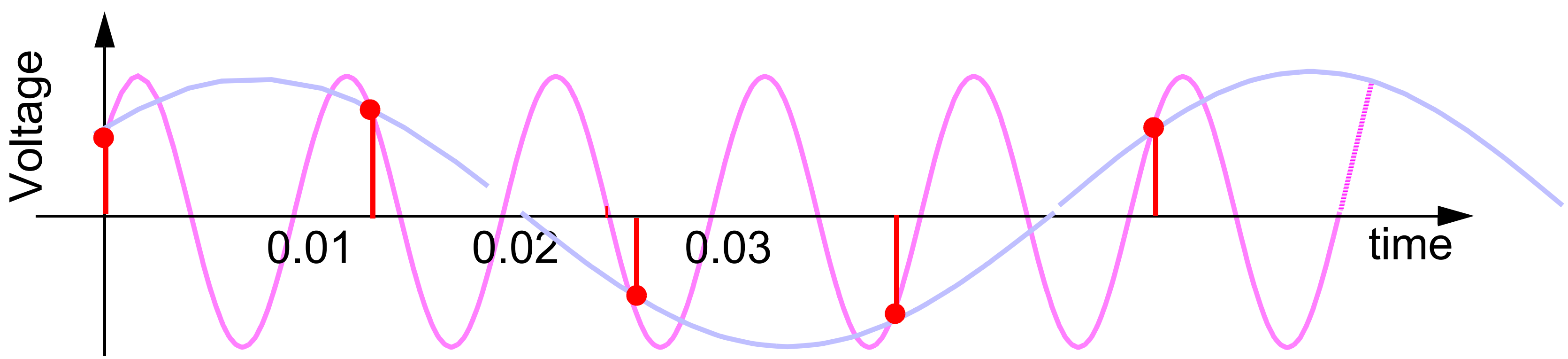
Sampling - Too Slow?

- Sampling at f_s = 100Hz, i.e. 1 sample per period:



Signal interpreted as DC!

- Sampling at f_s = 80Hz, i.e. 1 sample every 0.8 of a period:



Most of the signal features are "missed".

“Suitable” Sampling Rate

- From inspection of the above 100Hz digital waveforms at the four different sample rates:

  - f_s = 800Hz seems a *reasonable* sampling rate;
  - f_s = 3000Hz is perhaps higher than necessary;
  - f_s = 100Hz is too low and fails to correctly sample the waveform, and loses the signal parameter information;
  - f_s = 80Hz is too low and fails completely

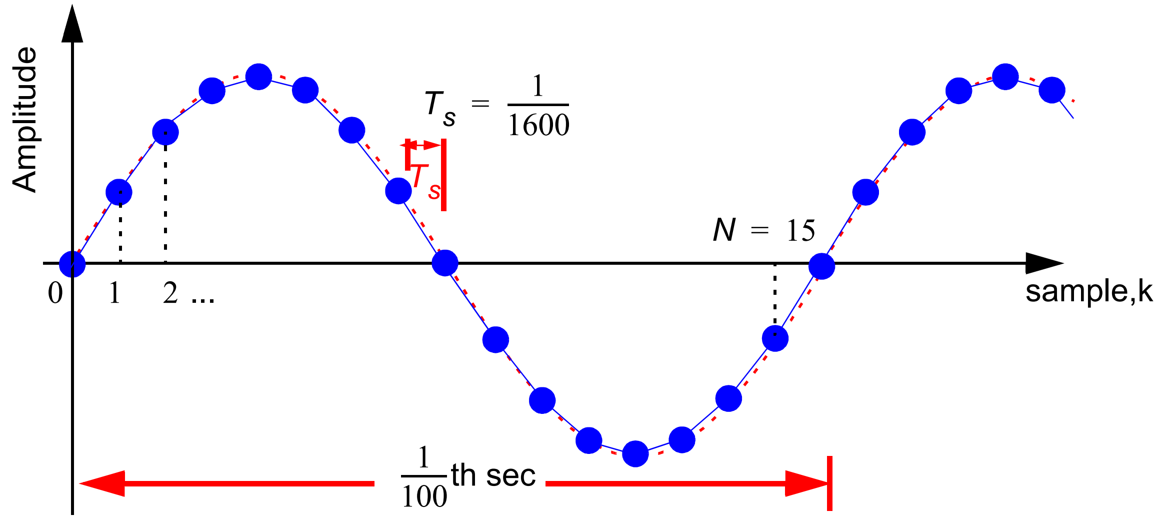
- From mathematical theory the minimum sampling rate to retain all information is:

  greater than **2 x f_{max}**

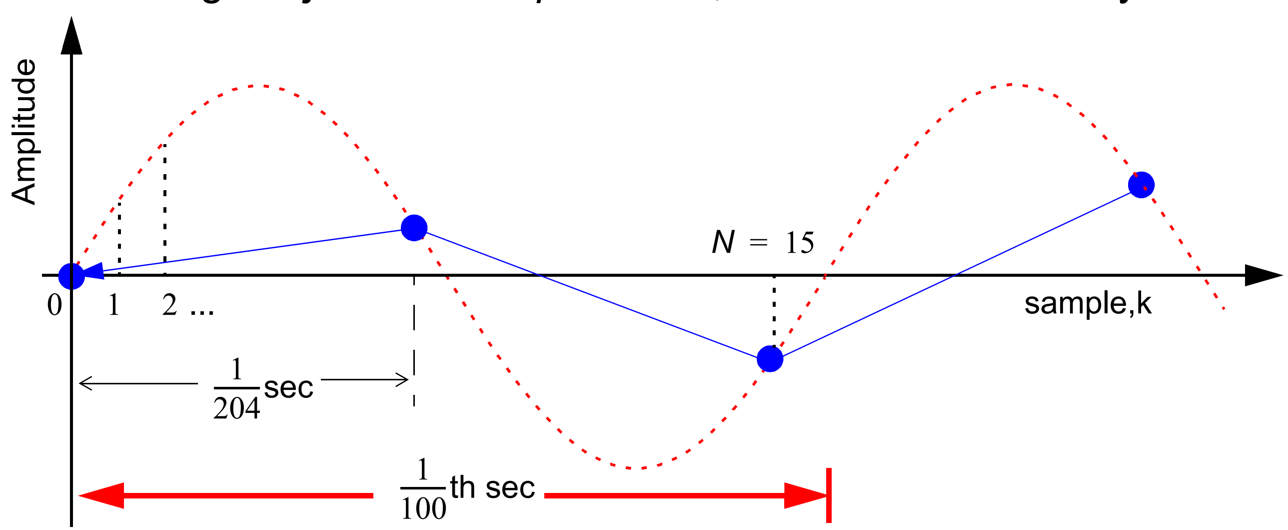
  where f_{max} is the maximum frequency component of a baseband, bandlimited signal.

Notes:

If we sample at a rate that is much higher than the signal frequency/bandwidth, such as sampling a 100Hz sine wave at 1600Hz, then the signal that can be viewed by simply joining the samples looks clearly like a sine wave.



However when we sample the 100Hz sine wave at just above Nyquist, i.e. $f_s = 204$Hz then this ***is*** high enough to mathematically retain all information about ALL sine wave components that may be present from 0 to 100Hz (actually 102 = $f_s/2$). But observing the *join the samples* view, it does not necessary ***look*** correct...... but it is.



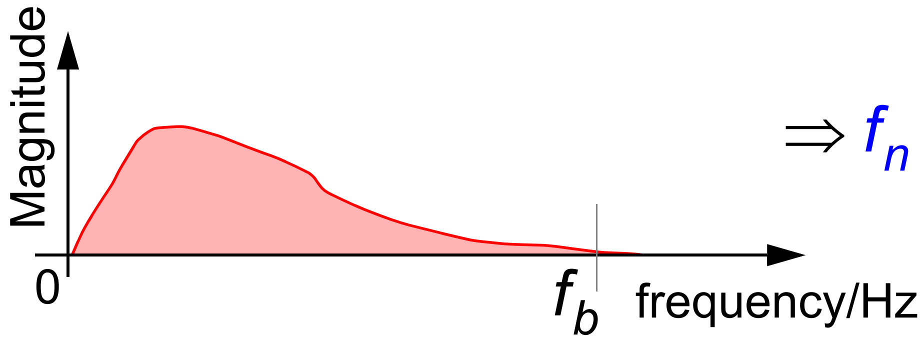
Top

Signal Frequency Range Terminology

- Nyquist frequency/rate: The Nyquist frequency, f_n is identified as twice the maximum frequency component present in a signal.

- Baseband: The lowest signal frequency present is around 0 Hz:

$$f_b = \text{bandwidth}$$



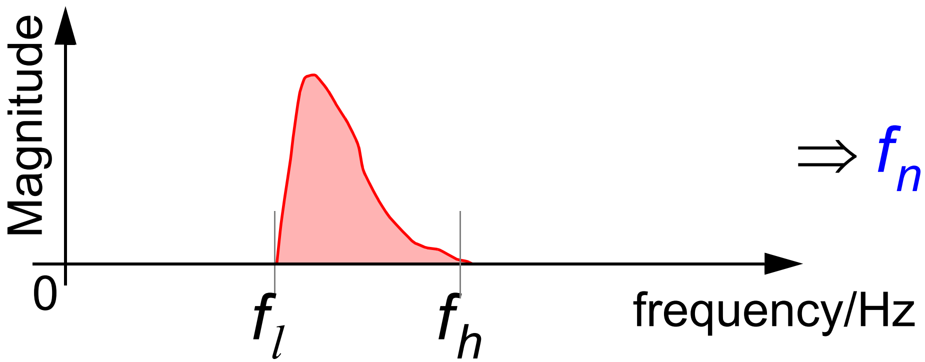
$$\Rightarrow f_n = 2f_b$$

- Bandlimited: For all frequencies in the signal $f_l < f < f_h$:

$$f_l = \text{lowest freq}$$

$$f_h = \text{highest freq}$$

$$f_b = f_h - f_l$$

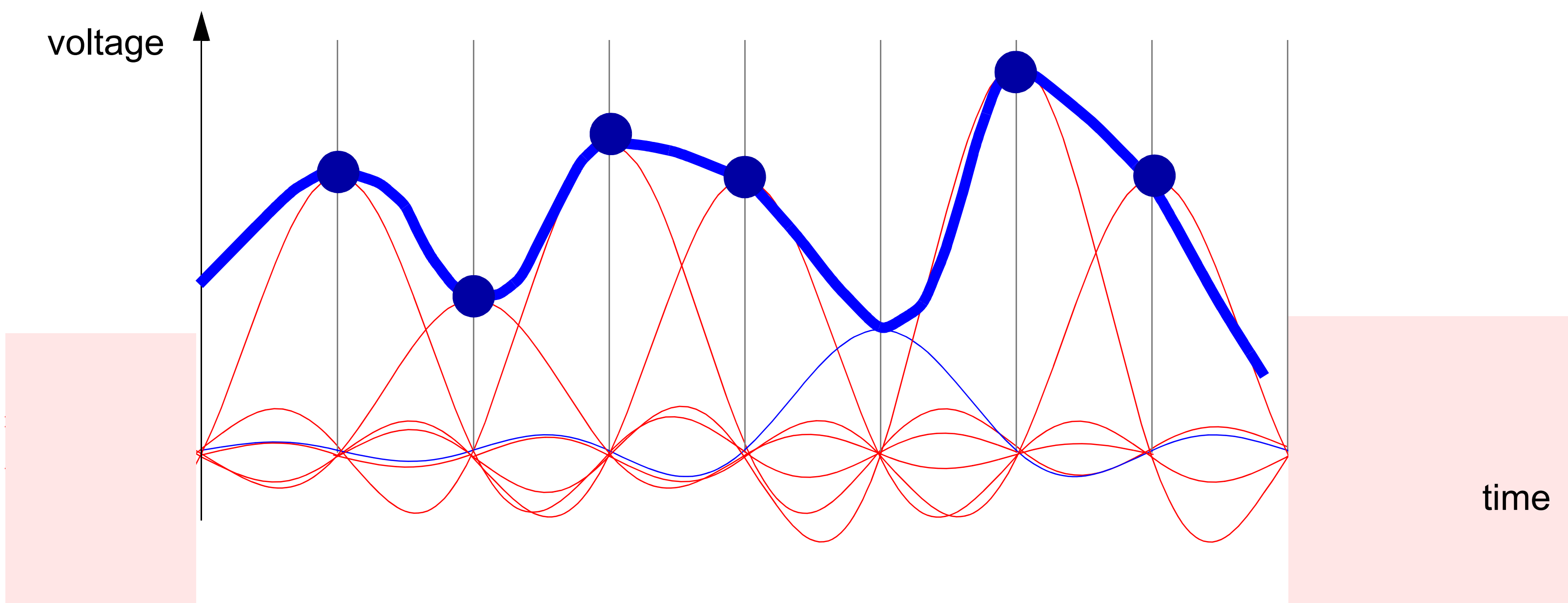


$$\Rightarrow f_n = 2f_h$$

Notes:

More precisely the Nyquist sampling theorem says that if a signal $x(t)$ was sampled at greater than twice the maximum frequency component present, then the signal can be exactly recovered from the sample values using the interpolation rule (or convolving a (non-causal) sinc function with the sampled signal):

$$x(t) = \sum_{k=-\infty}^{\infty} x(kt_s)h(t-kt_s) \qquad h(t) = \frac{\sin(\pi t/t_s)}{\pi t/t_s}$$

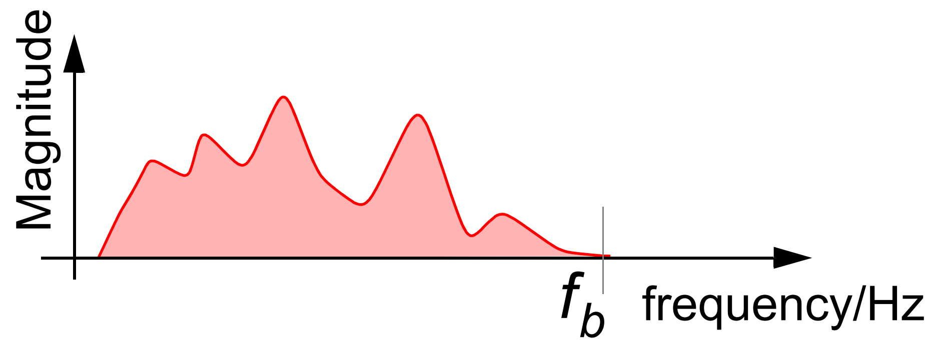


IIn practice true sinc interpolation is not possible to do (due to temporal contrants), hence as we will see later we usually perform zero order hold followed by low pass filtering to recover the original signal.

Nyquist Sampling Rate

- If a baseband, bandlimited signal is composed of “sine waves” up to a frequency f_b Hz, then

Nyquist frequency, $f_n = 2f_b$



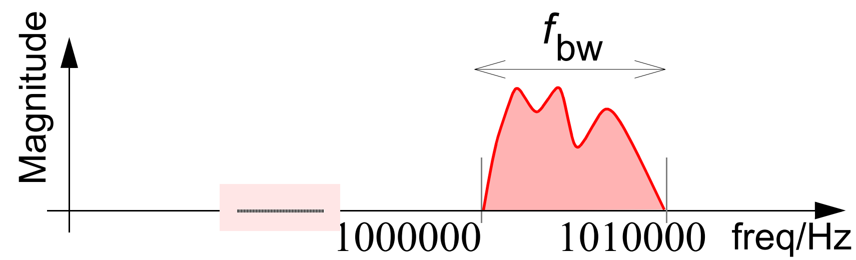
- In we require to sample this signal and retain all information, then the sampling rate, f_s must be chosen as:

$$f_s > f_n \quad \text{i.e.} \quad f_s > 2f_b$$

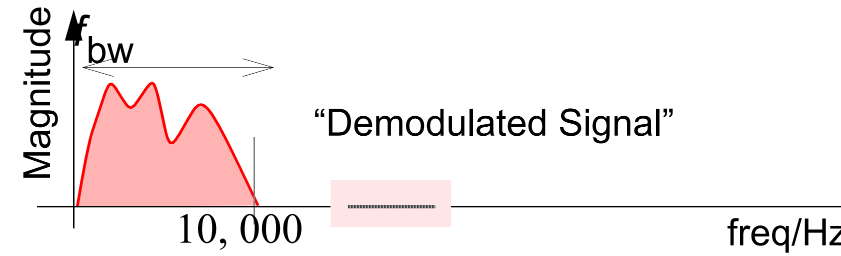
- This frequency is often refered to as the Nyquist sampling rate, (distinct from the Nyquist frequency!).

Notes:

If a signal bandwidth is from 1,000,000 Hz to 1,010,000 Hz: we might be tempted to suggest that the sampling frequency (according to the Nyquist criteria) is $f_s = 2f_{bw} = 2,020,000$ Hz .



However if we realise that we could "demodulate" this signal to baseband and *then* sample, it would seem that we can find some mathematical justification that the direct sampling rate be only $f_s = 2f_{bw} = 20,000$ Hz . Because the signal is bandlimited, if we sample at 20000 kHz then the signal from 1 - 1.01 MHz will alias down to baseband and all signal information retained.



Therefore we are effectively downsampling the signal by a factor of 50 and then sampling. Hence it is appropriate to state that the sampling frequency required to sample any signal can be $f_s = 2f_{bw}$ rather than that predicted by the Nyquist criteria. The above signal could be sampled at 20000Hz *(Note in this example for simplicity we have carefully chosen the maximum frequency in the bandlimited signal and bandwidth to be integer multiples of each other, in practice when this is not true the required sampling rate will be greater than $2f_{bw}$ in cases where demodulation to baseband is NOT performed, i.e. the demodulation is implicitly done by aliasing.)*

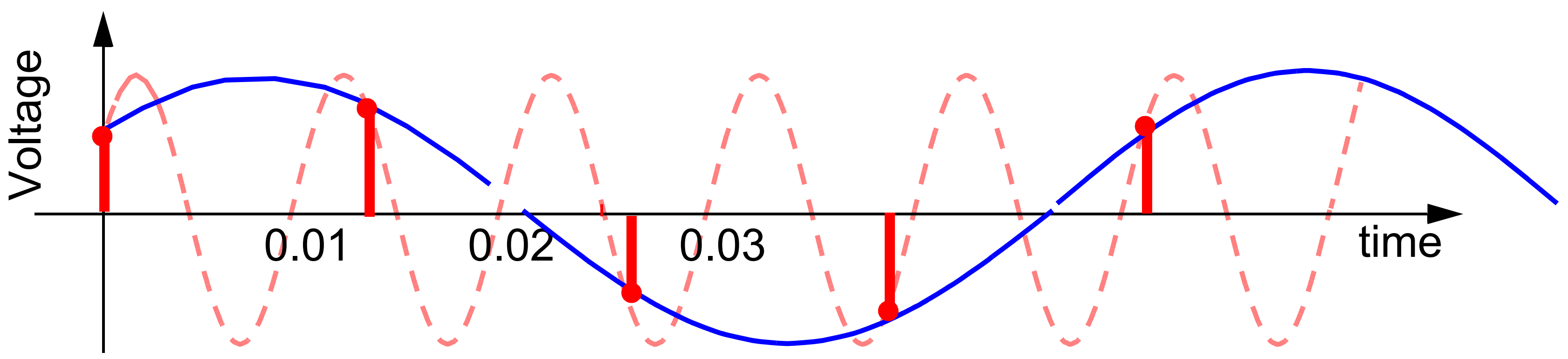
This general process is called **undersampling** and is an area of considerable "DSP" activity at present with respect to "software radio" for communication systems where the electromagnetic signal of interest (e.g. a mobile phone in the high 100's MHz's) is demodulated to an intermediate frequency of around 1 MHz and then sampled at a frequency rate of the order of $2f_{bw}$ rather than $2f_{max}$. It is important to point out however that the sampler required to accomplish undersampling, requires a bandwidth capability equivalent to f_{max} given that the obtained sample is essentially an integration over $1/f_{max}$.

Aliasing

- When a (baseband) signal is sampled at a frequency *below the Nyquist rate*, then we "lose" the signal frequency information and aliasing is said to have occured.

- Aliasing can be illustrated by sampling a sine wave at below the Nyquist rate and then "reconstructing". We note that it appears as a sine wave of a *lower* frequency
(aliasing - cf. impersonating).

- Consider again sampling the 100Hz sine wave at 80Hz:

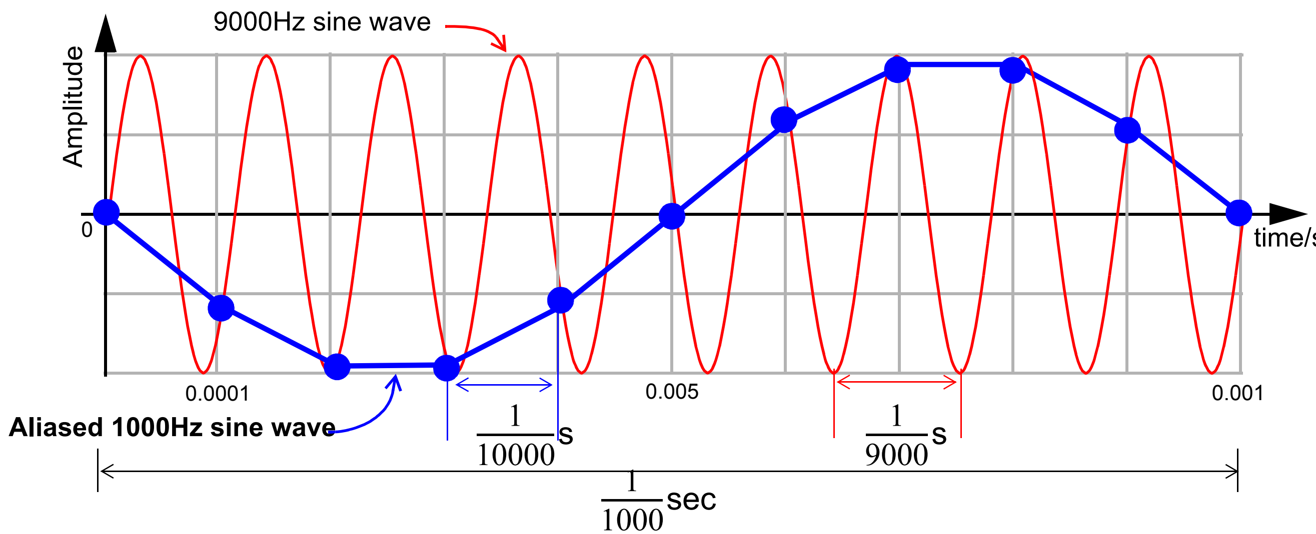


Reconstructed signal has a freq. of f_s - f_{signal} = 20Hz

Notes:

Clearly if a signal has frequency components greater than $f_s/2$ then aliasing will occur. Aliasing will manifest as distortion of a signal; for example if a 6000 Hz tone is input to a DSP system (without anti-alias and reconstruction filters) and sampled at 10000Hz, the sampled signal is interpreted as a 4000Hz tone. Clearly this is non-linear behavior! (One of the simplest ways of testing the linearity of any system at a particular frequency is to input a pure tone. If the output is not a pure tone (it may contain harmonics) then the system is NOT linear.)

Consider the following figure where we sample a 9000Hz tone at $f_s = 10000$Hz:

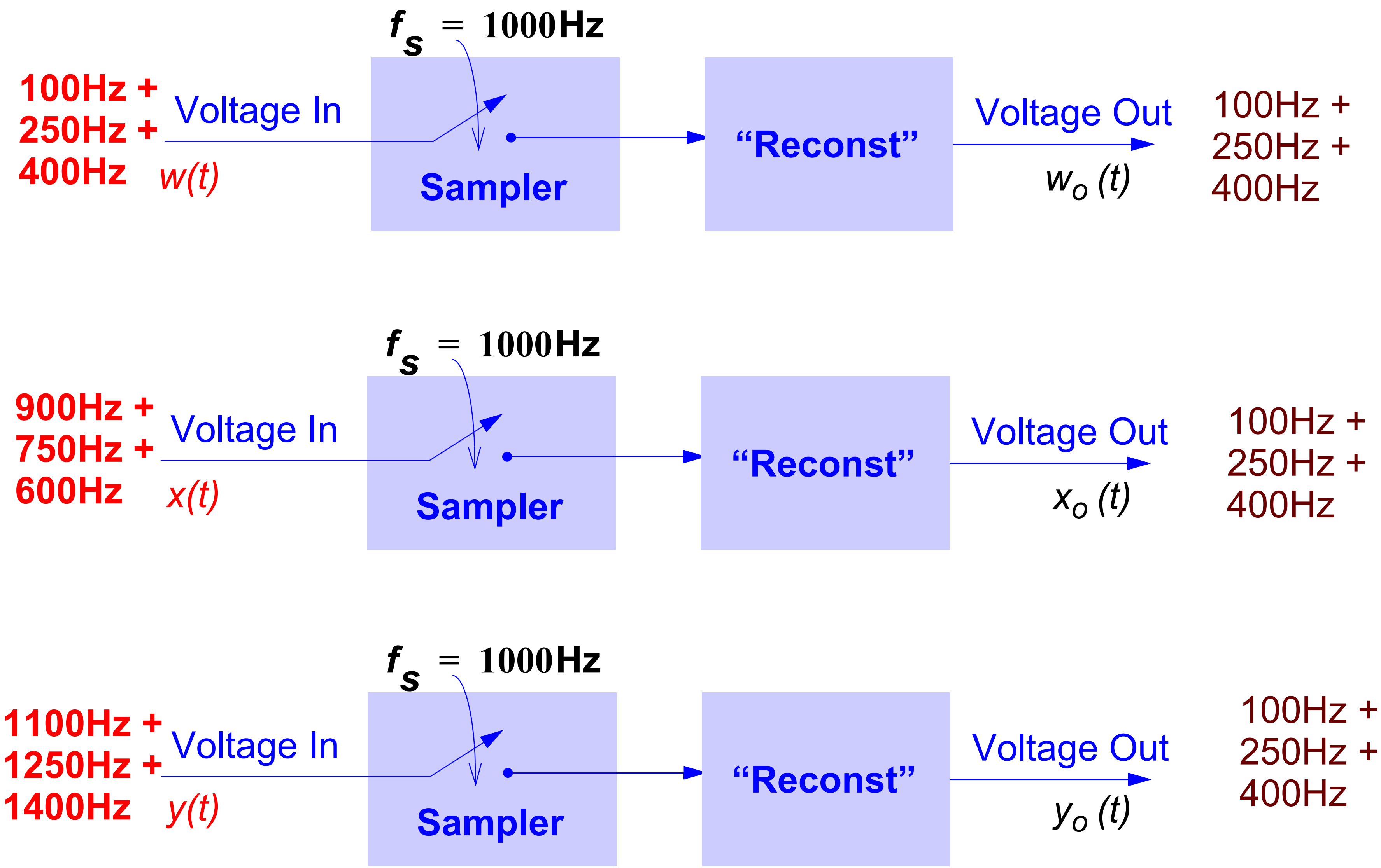


Clearly this is above $f_s/2 = 5000$Hz and the 9000Hz will alias. The diagram illustrates that when we reconstruct this signal we get a 1000Hz sine wave. Note that the phase has also shifted, and is now 180^o out of phase, compared to the phase of the input at 9000Hz.

From a knowledge of f_s and the input frequencies it is straightforward to establish the frequency of the aliased components for any input components above $f_s/2$. This is presented in the next two slides.

Aliasing Examples

- Consider the output from the following three systems:

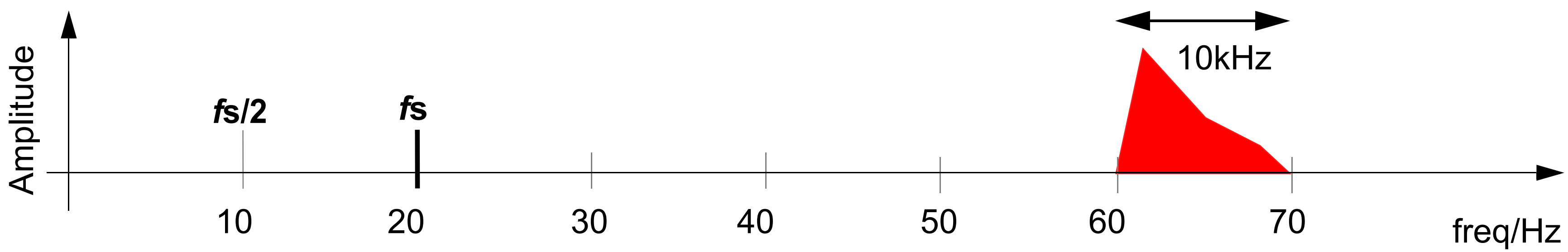


Notes:

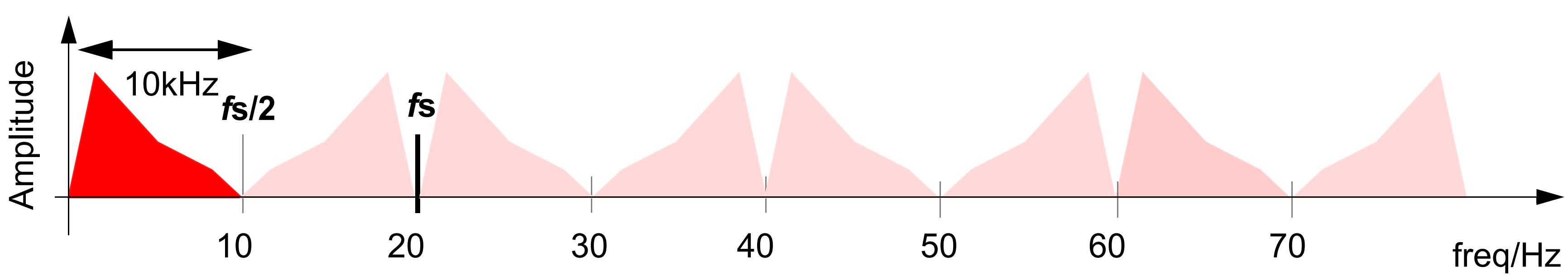
If you are viewing the output signal only, after sampling then a suitable reconstruction you cannot be sure what the input signal given that aliasing may have occured.

Clearly the above system sampling at 1000Hz would be best limited to only have input signals with frequencies below 500Hz.

Of course sometimes ***aliasing can be exploited*** in situations where perhaps we are aiming to demodulate a signal. As we will discuss in the Software Radio section of notes, ***direct digital downconversion*** could be used to "demodulate" (or alias down) from between 60 and 70kHz to 10kHz by taking only 20000 samples per second (although the front end of the ADC must still be capable on integrating over a time interval that is commensurate with the signal bandwidth).

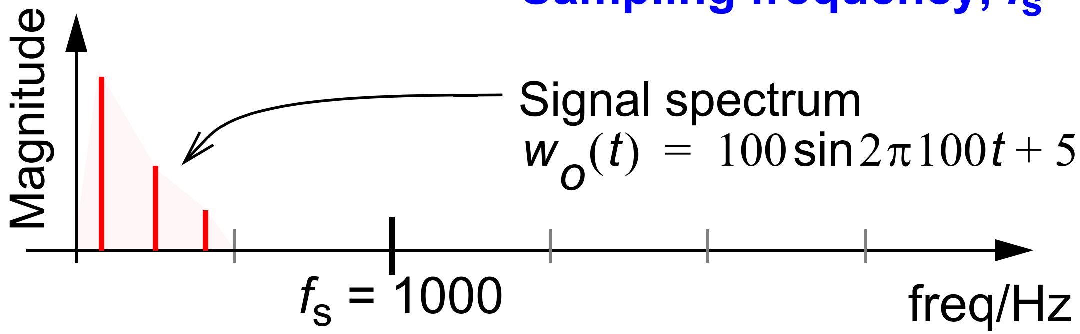


If this signal is appropriately bandlimited, the output signal will alias to the same "shape" at baseband frequencies:



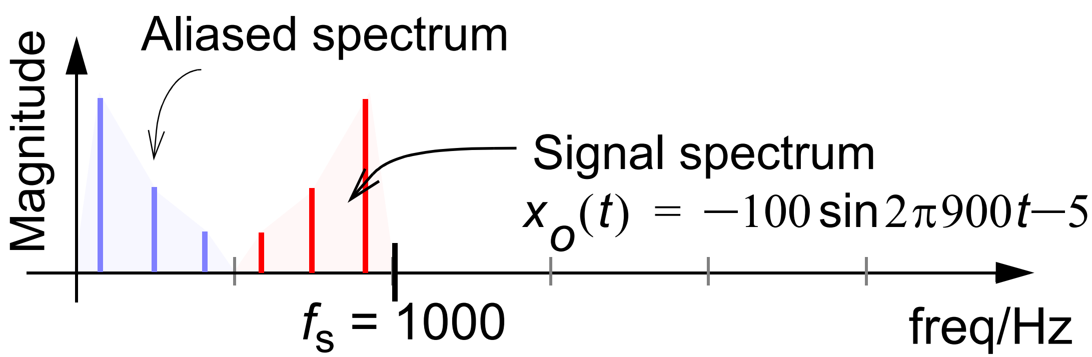
Aliased Spectra

Sampling frequency, f_s = 1000Hz



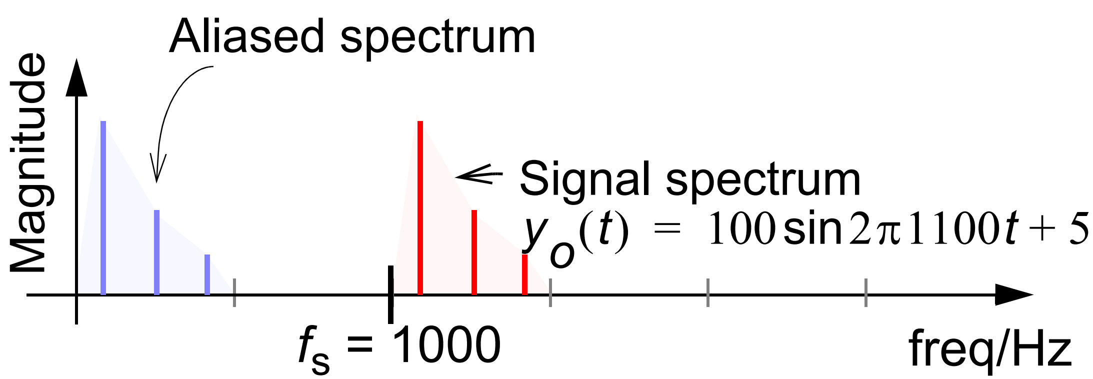
$$w_o(t) = 100\sin 2\pi 100t + 50\sin 2\pi 250t + 25\sin 2\pi 400t$$

NO ALIASING



$$x_o(t) = -100\sin 2\pi 900t - 50\sin 2\pi 750t + -25\sin 2\pi 600t$$

ALIASING

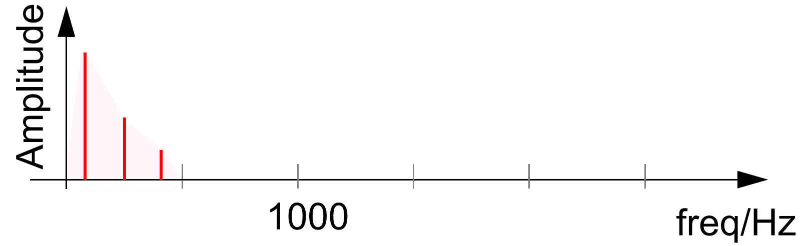


$$y_o(t) = 100\sin 2\pi 1100t + 50\sin 2\pi 1250t + 25\sin 2\pi 1400t$$

ALIASING

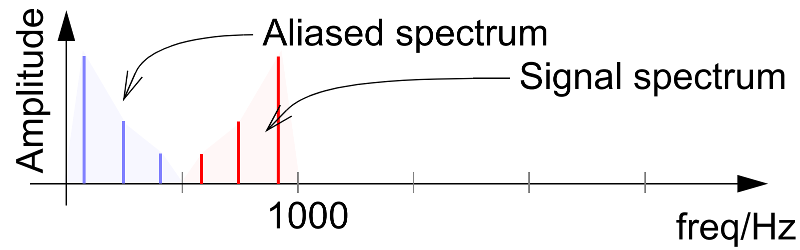
Notes:

If we sample a sum of a few sine waves signal, $w(t)$ at 1000 Hz we adhere to the Nyquist criteria. We can represent the signal to be sampled as a simple (sine wave amplitude) spectrum:



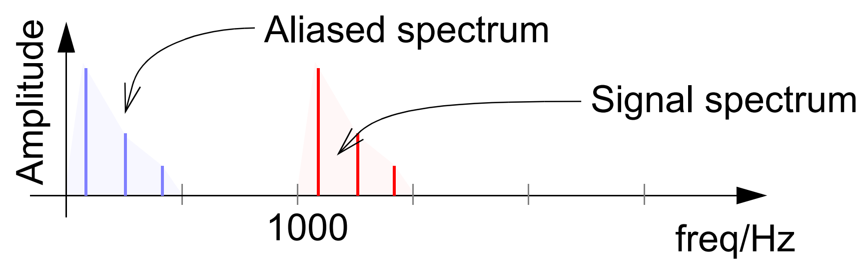
$$w(t) = 100\sin 2\pi 100t + 50\sin 2\pi 250t + 25\sin 2\pi 400t$$

Of course if we had sampled the the signal $x(t)$ (i.e. sine waves at 900Hz, 750Hz and 600Hz at 1000 Hz), then because Nyquist criteria is not preserved, all of these sine waves will alias respectively to 100, 250 and 400 Hz components.



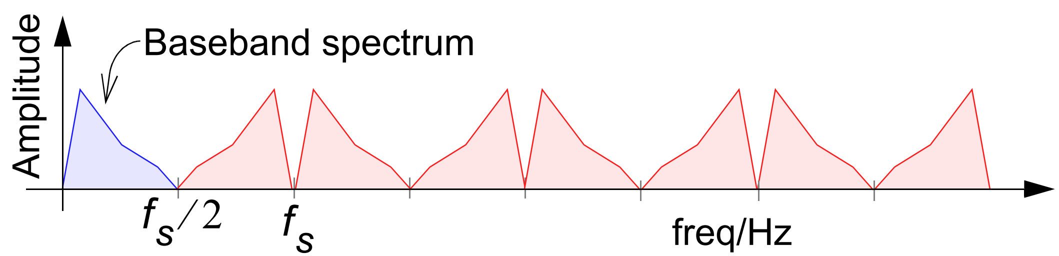
$$x(t) = 100\sin 2\pi 900t + 50\sin 2\pi 750t + 25\sin 2\pi 600t$$

Similarly if we sampled the signal $y(t)$ (i.e. sine waves at 1100Hz, 1250Hz and 1400Hz) at 1000 Hz, then because Nyquist criteria is not preserved, all of these sine waves will alias respectively to 100, 250 and 400 Hz components.



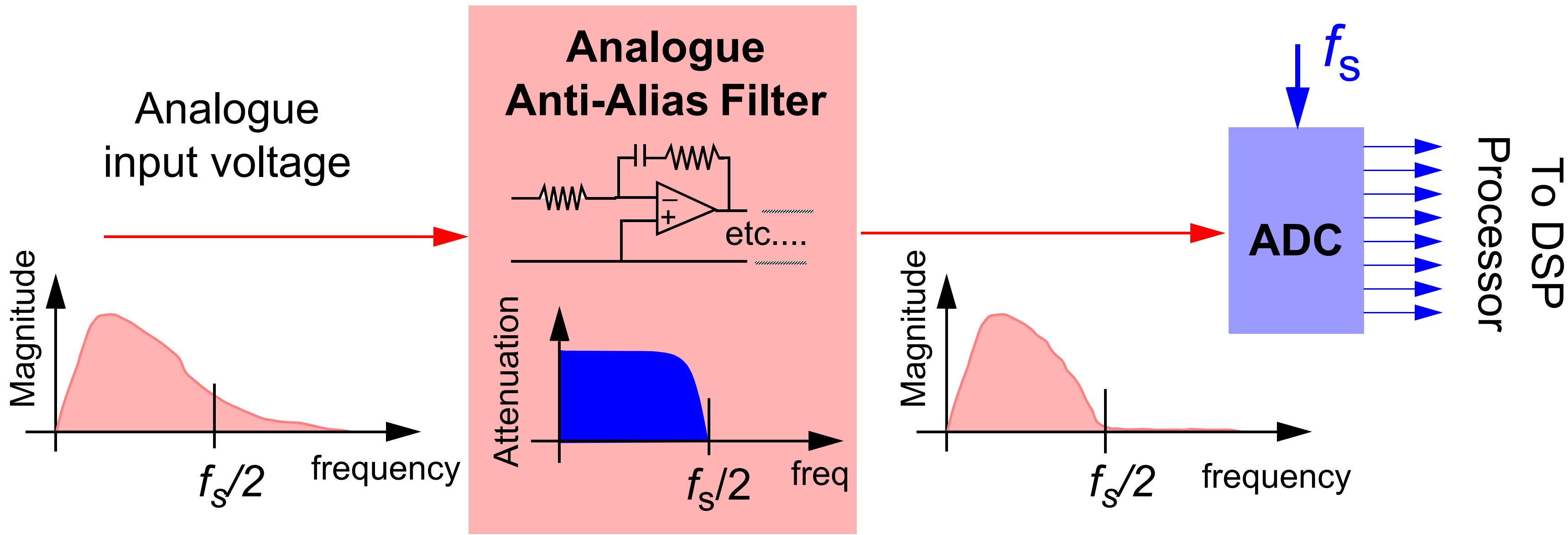
$$y(t) = 100\sin 2\pi 1100t + 50\sin 2\pi 1250t + 25\sin 2\pi 1400t$$

We can then of course note the "pattern" of spectrum aliasing as shown below:



Anti-Alias Filter

- Prior to the analogue to digital converter (ADC) all frequencies above fs/2 must be blocked or they will be interpreted as lower frequencies, i.e aliasing.

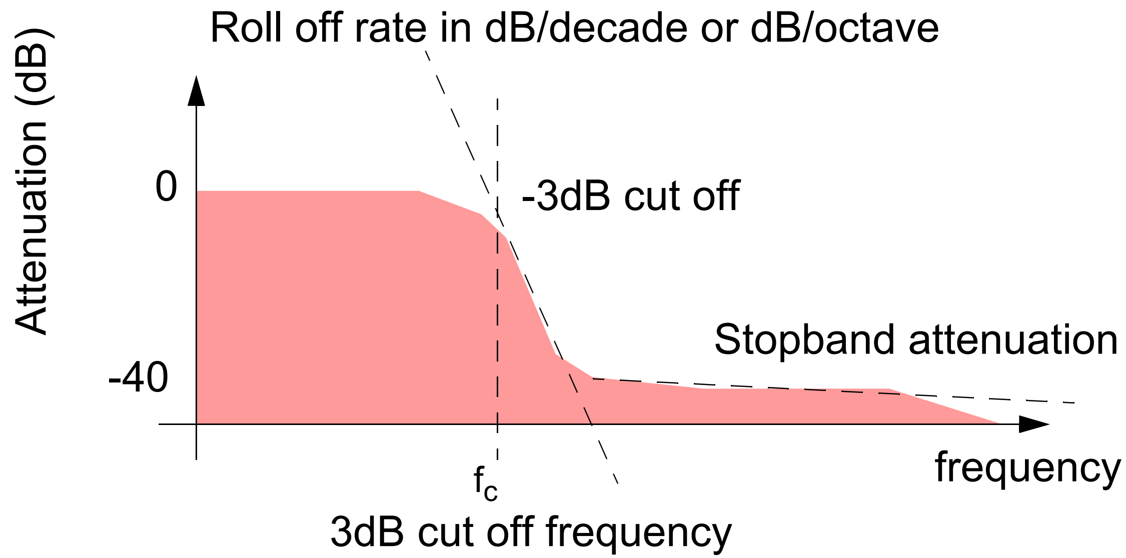


- The anti-alias filter is analogue (ideally a *brick wall filter*), cutting off just before $f_s/2$ Hz.

Generic DSP

Notes:

Clearly brick wall filters cannot be implemented, and therefore the analog filter designer must compromise to produce a filter “enough” roll-off, attenuation etc. The filter designed should also ideally have linear phase. Therefore the work of the anti-alias designed is less than trivial!

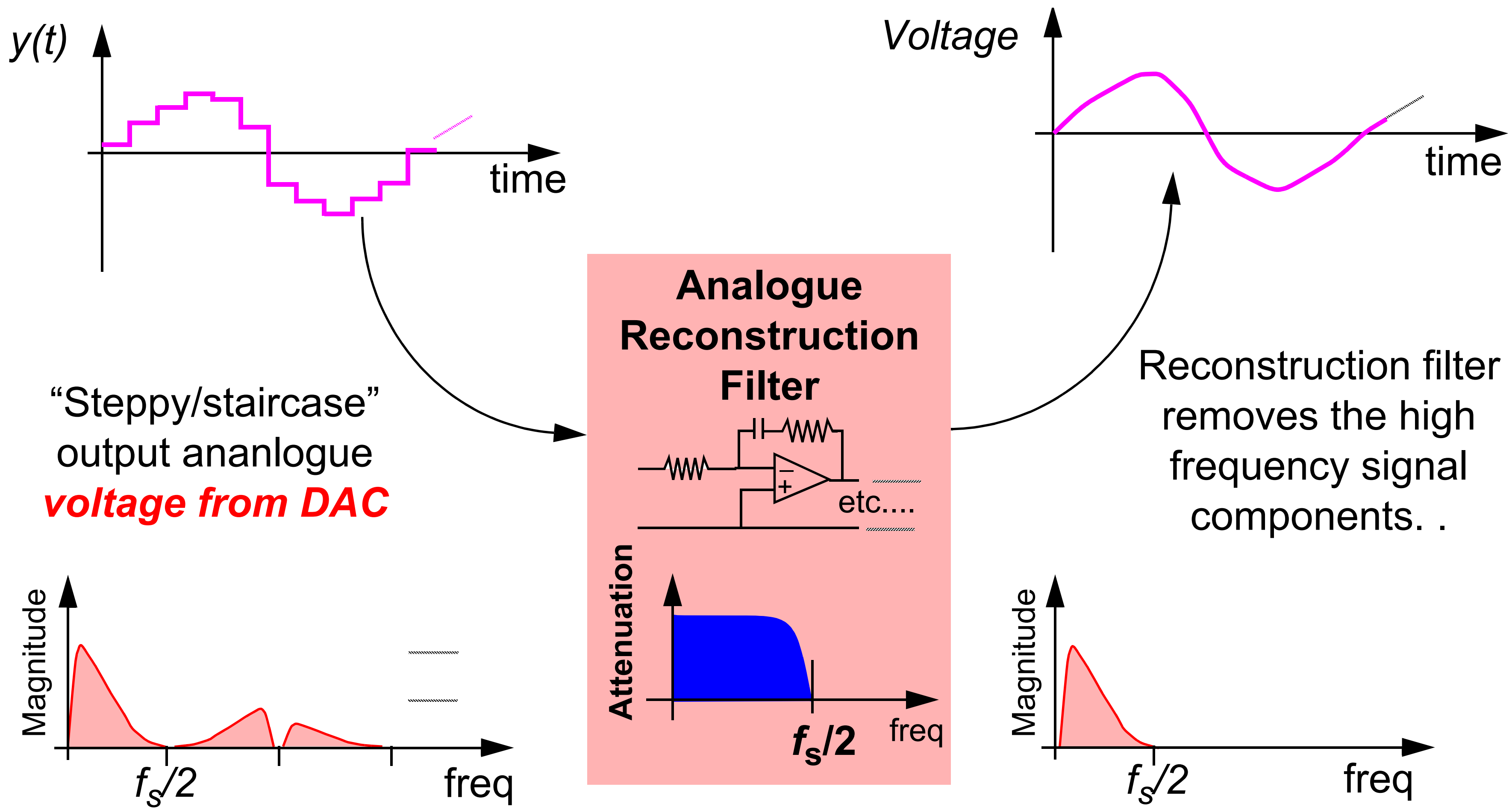


0dB corresponds to an attenunation of 1, i.e., if $20\log\frac{V_{out}}{V_{in}} = 0$, then $\frac{V_{out}}{V_{in}} = 1$ and therefore $V_{out} = V_{in}$.

-40dB corresponds to $20\log\frac{V_{out}}{V_{in}} = -40$, then $\frac{V_{out}}{V_{in}} = 0.01$ and therefore $V_{out} = 0.01V_{in}$.

Reconstruction Filter

- The analogue reconstruction filter at the output of a DAC removes the baseband image high frequencies present in the signal (in the form of the *steps* between the discrete levels).

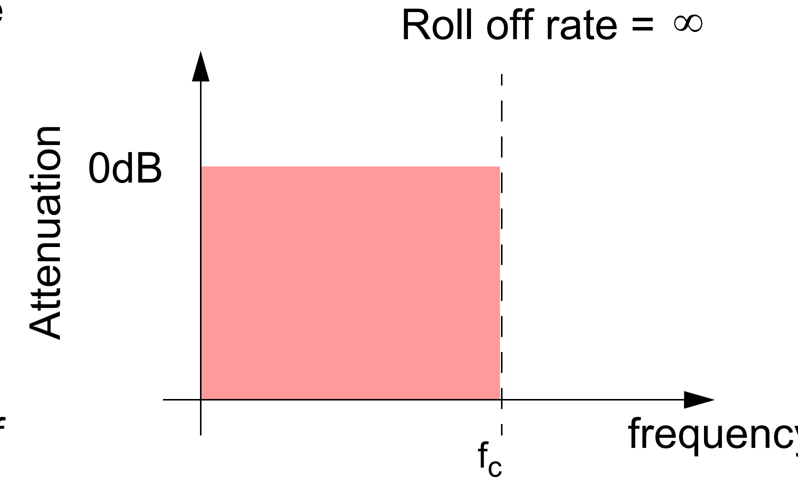


Notes:

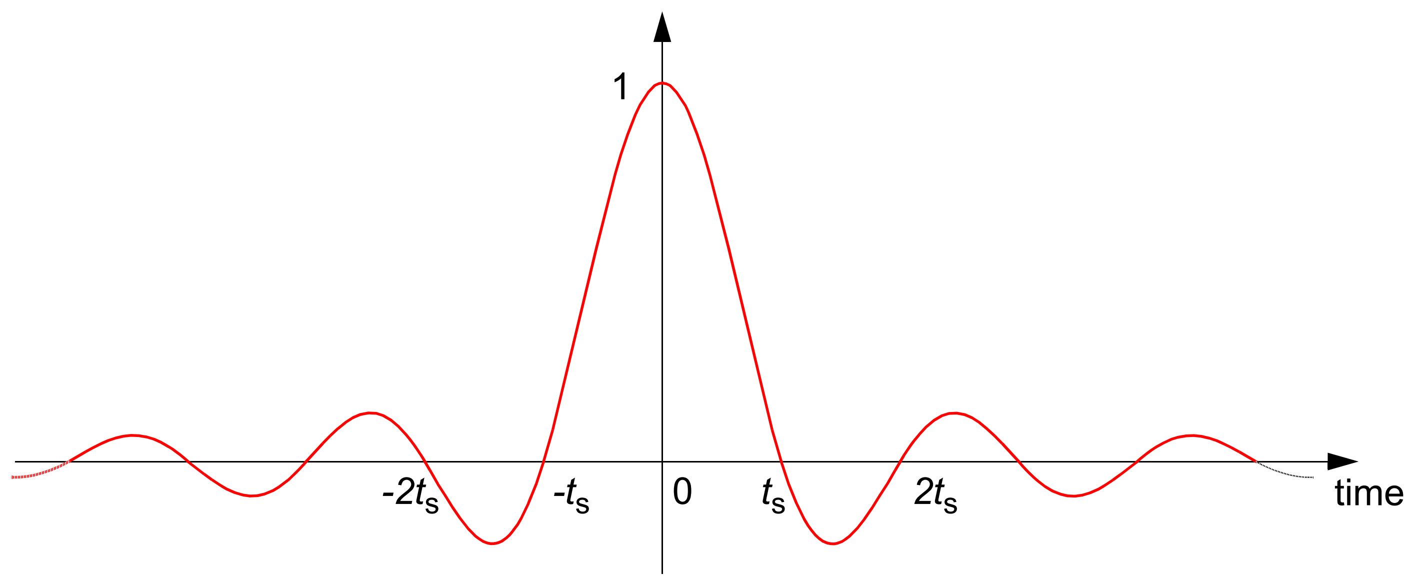
Note that the (time) impulse response (discussed later) of the perfect brick wall filter is in fact a sinc function

Note that the (time) impulse response (discussed later) of the perfect brick wall filter is in fact a sinc function:

$$\frac{\sin(\pi t / t_s)}{\pi t / t_s}$$

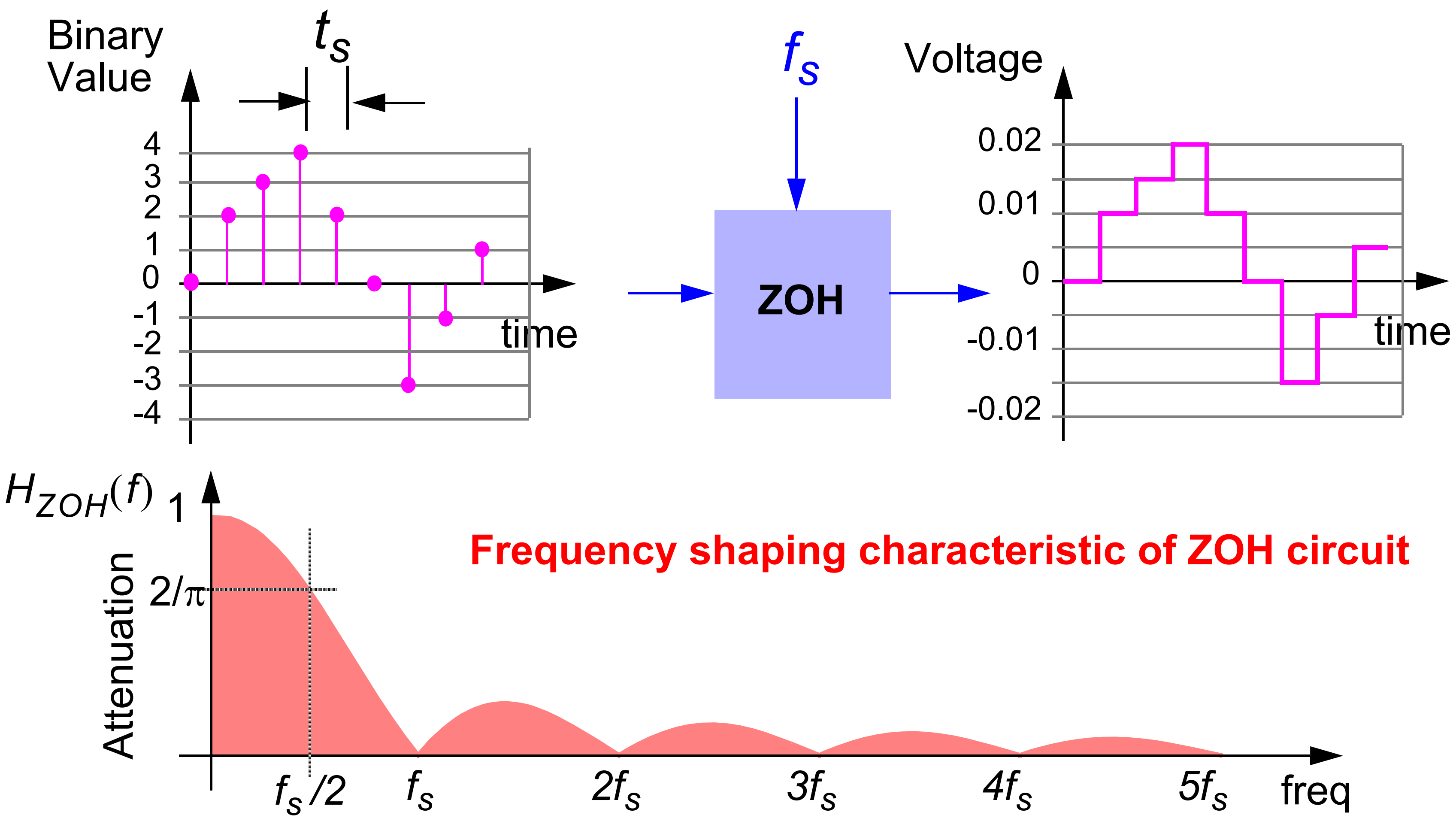


which starts at $t = -\infty$ and ends at $t = \infty$. Hence the existence of the sinc interpolation process.



Zero Order Hold (ZOH) "Filter"

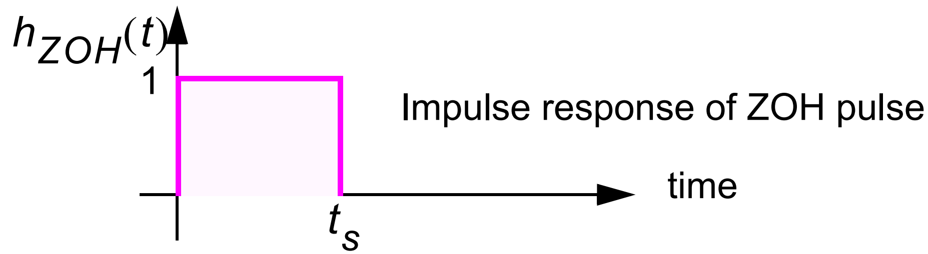
- Note that the operation of zero order hold can be interpreted as a simple "reconstructing" frequency filtering operation:



- The step reconstruction therefore causes a "droop" near $f_s/2$.

Notes:

The frequency response of the ZOH circuit can be calculated by simply noting that the impulse response is:



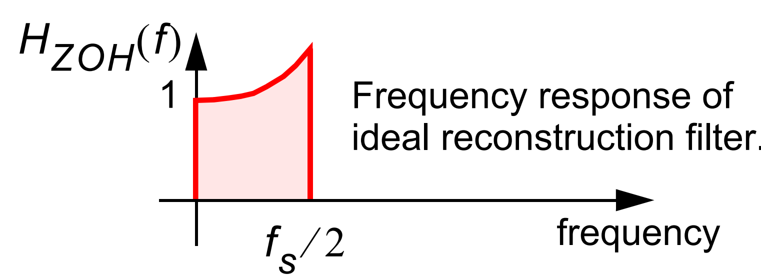
and finding the frequency response (as shown in diagram in above slide) via a Fourier transform;

$$H_{ZOH}(f) = \int_0^{t_s} 1.e^{-j2\pi ft}\,dt = \frac{1}{j2\pi ft_s}[1 - e^{-j2\pi ft_s}] = \left[\frac{e^{j\pi ft_s} - e^{-j\pi ft_s}}{2j(\pi ft_s)}\right]e^{-j\pi ft_s} = \frac{\sin \pi ft_s}{\pi ft_s}e^{-j\pi ft_s}$$

Hence the ideal reconstruction filter should compensate for the “sinx/x” “droop” at frequency $f_s/2$ by a factor of:

$$2/\pi = 0.637 \equiv 20\log 0.637 = -3.92 \text{ dB} .$$

Therefore the magnitude response for “perfect” reconstruction (a linear phase response is also ideally required) is a filter that compensates at $f_s/2$ with a gain of 1/0.637 = 1.569; above this value, all frequency components should be attenuated (in a brick wall fashion!).

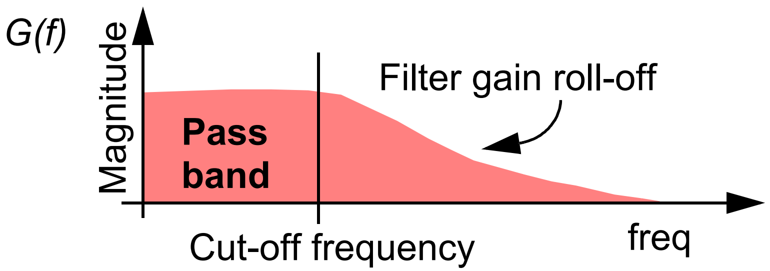


In practice of course such an analogue filter will not be produced, and as far as possible a close to ideal brick wall filter will be used. In order to compenstate for the droop, we can actually introduce a digital filter before the DAC to amplify the signal with the above inverse droop characteristic! And finally in the modern DSP world, oversampling will be used to reduce the analogue complexity (see later....)

Anti-alias and Reconstruction

- Anti-alias and reconstructions filters are analogue, i.e. made from resistors, capacitors, amplifiers, even inductors.

- Ideally they are both very sharp cut off filters at frequency $f_s/2$. In practice the roll off will be between 6dB/octave (a simple resistor and capacitor) to 96dB/octave (a 16th order filter).



- Steeper roll-off is more expensive, but clearly for many applications, good analogue filters are essential.

- In a DSP system the accurately trimmed analogue filters could actually be more costly than the other DSP components: i.e. DSP processor, ADC, DAC, memory etc.

Notes:

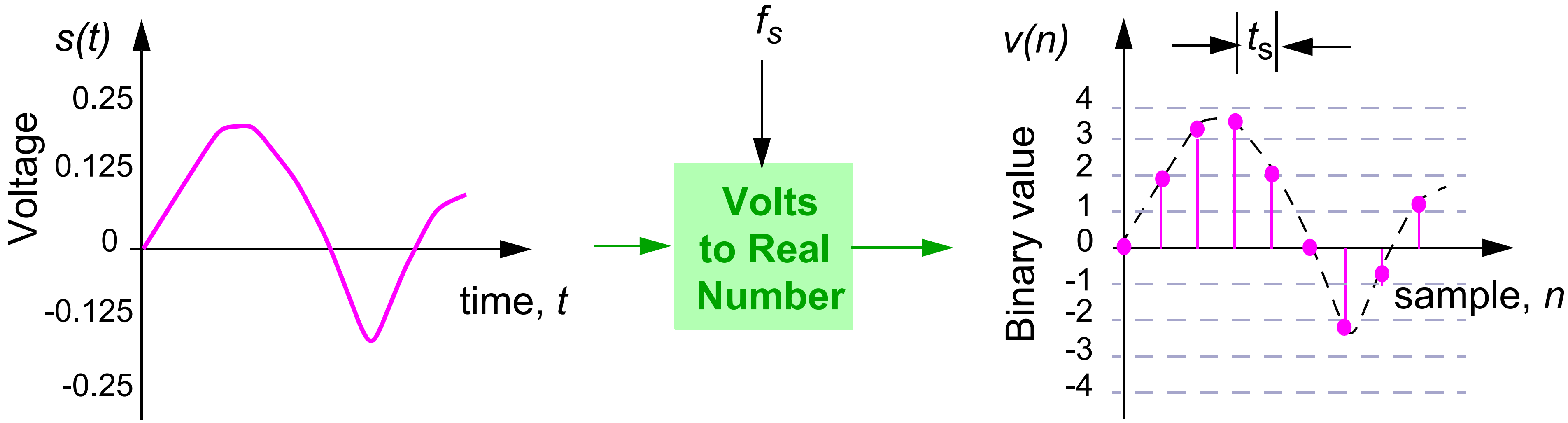
One of the key changes in signal processing over the last few years has been the introduction of oversampling (and sigma delta) DSP systems. The key strategy behind oversampling is to reduce the cost of the analog components at the expense of more DSP processing and higher sampling rates. With the reduction in cost of reliable easy to fabricate DSP systems, more modern DSP implementations use oversampling ADCs and DACs. Oversampling is discussed later in the multirate section of the course.

Perfect Nyquist Sampling

- The Nyquist sampling theorem states that a (baseband) signal should sampled at greater than twice the maximum frequency component present in the signal:

$$f_s > 2 \times f_{max}$$

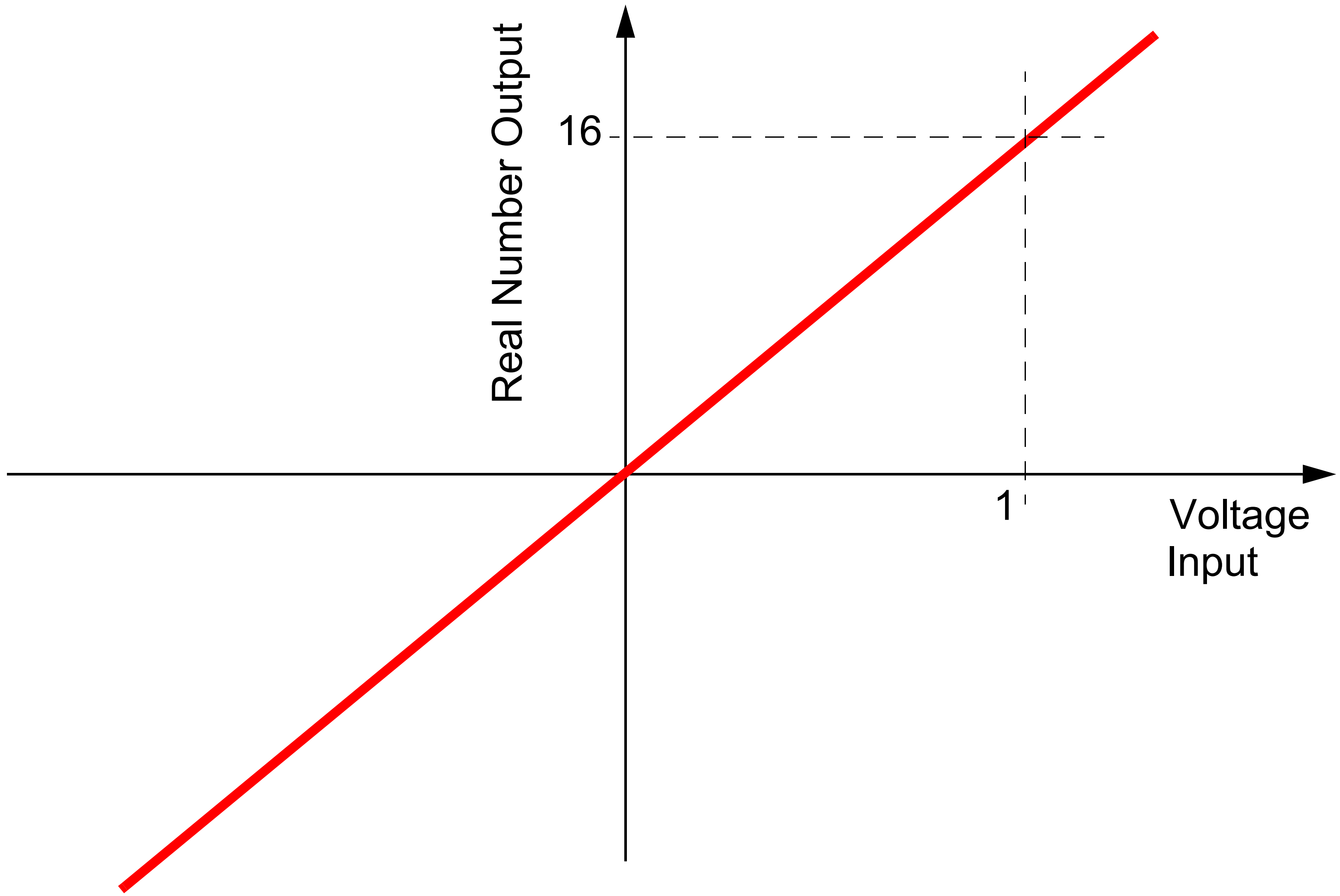
- The sampled signal can then be *perfectly* reconstructed to the original analogue signal with no added noise or distortion.



$$v(n) = s(nt_s), \quad \text{for } n = 0, 1, 2, \ldots$$

Notes:

In the above example the "converter" produces a real number value, i.e an input voltage of 1.10233 is output as the real number 1.10233. Therefore the input/output characteristic of this system is:

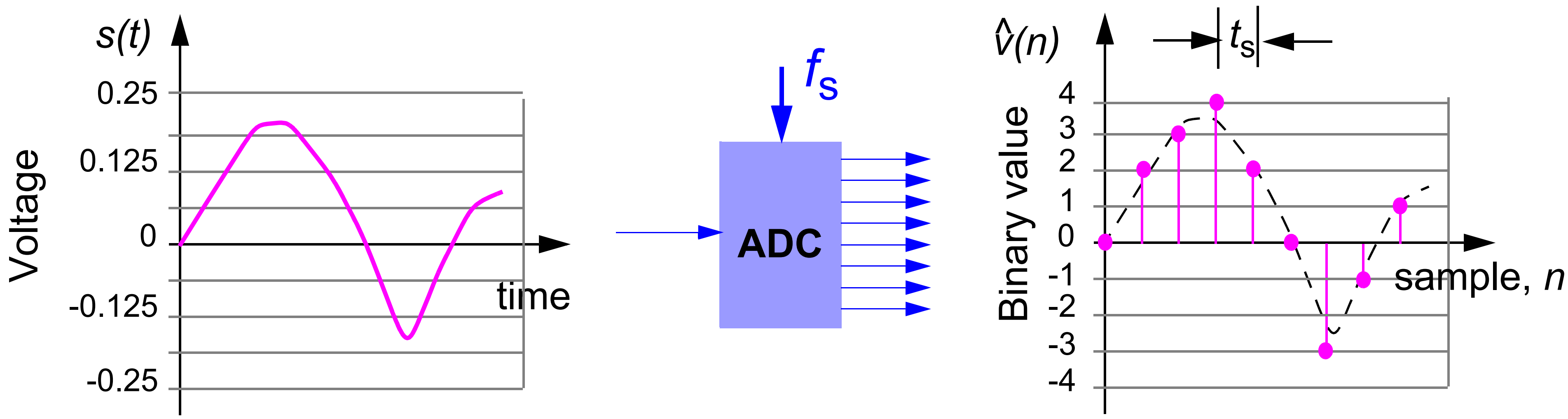


Clearly such real number "converters" do not exist.

ADC Sampling "Error"

- Perfect signal reconstruction assumes that sampled data values are exact (i.e. infinite precision real numbers). In practice they are not, as an ADC will have a number of discrete levels.

- The ADC samples at the Nyquist rate, and the sampled data value is the closest (discrete) ADC level to the actual value:

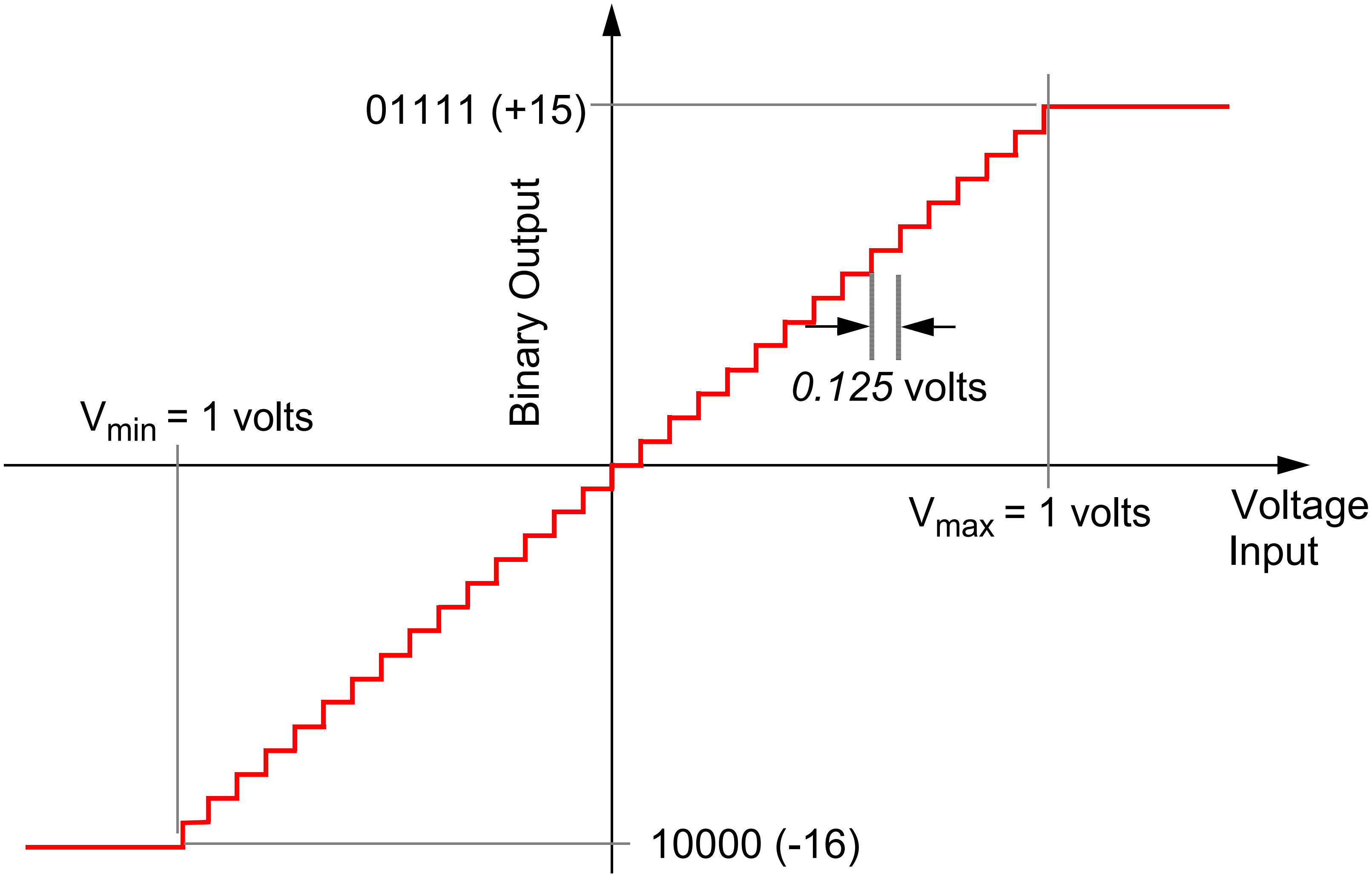


$$\hat{v}(n) = \text{Quantise}\{s(nt_s)\}, \quad \text{for } n = 0, 1, 2, \ldots$$

- Hence every sample has a "small" quantisation error.

Notes:

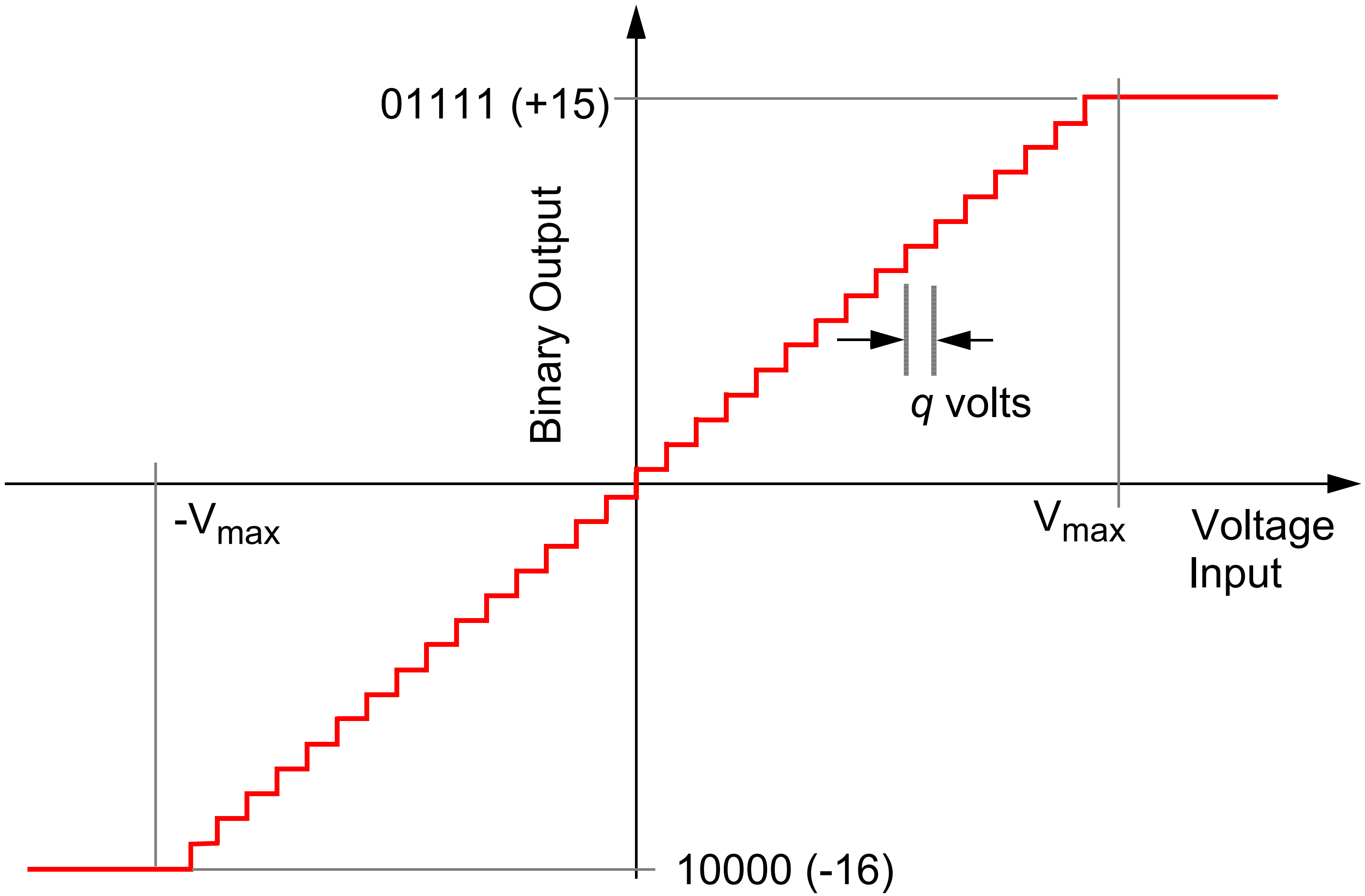
For example purposes, we can assume our ADC or quantiser has 5 bits of resolution and maximum/minimum voltage swing of +1 and -1 volts. The input/output characteristic is shown below:



In the above slide figure, the second sample, the true sample value is 1.589998..., however our ADC quantises to a value of 2.

Quantisation Error

- If the smallest step size of a linear ADC is ***q* volts**, then the error of any one sample is at worst ***q*/2** volts.



Notes:

The quantisation error is straightforward to calculate from:

$$q = \frac{V_{\max}}{2^{N-1}}$$

where N is the number of bits in the converter.

The dynamic range of an N bit converter is often quoted in dBs:

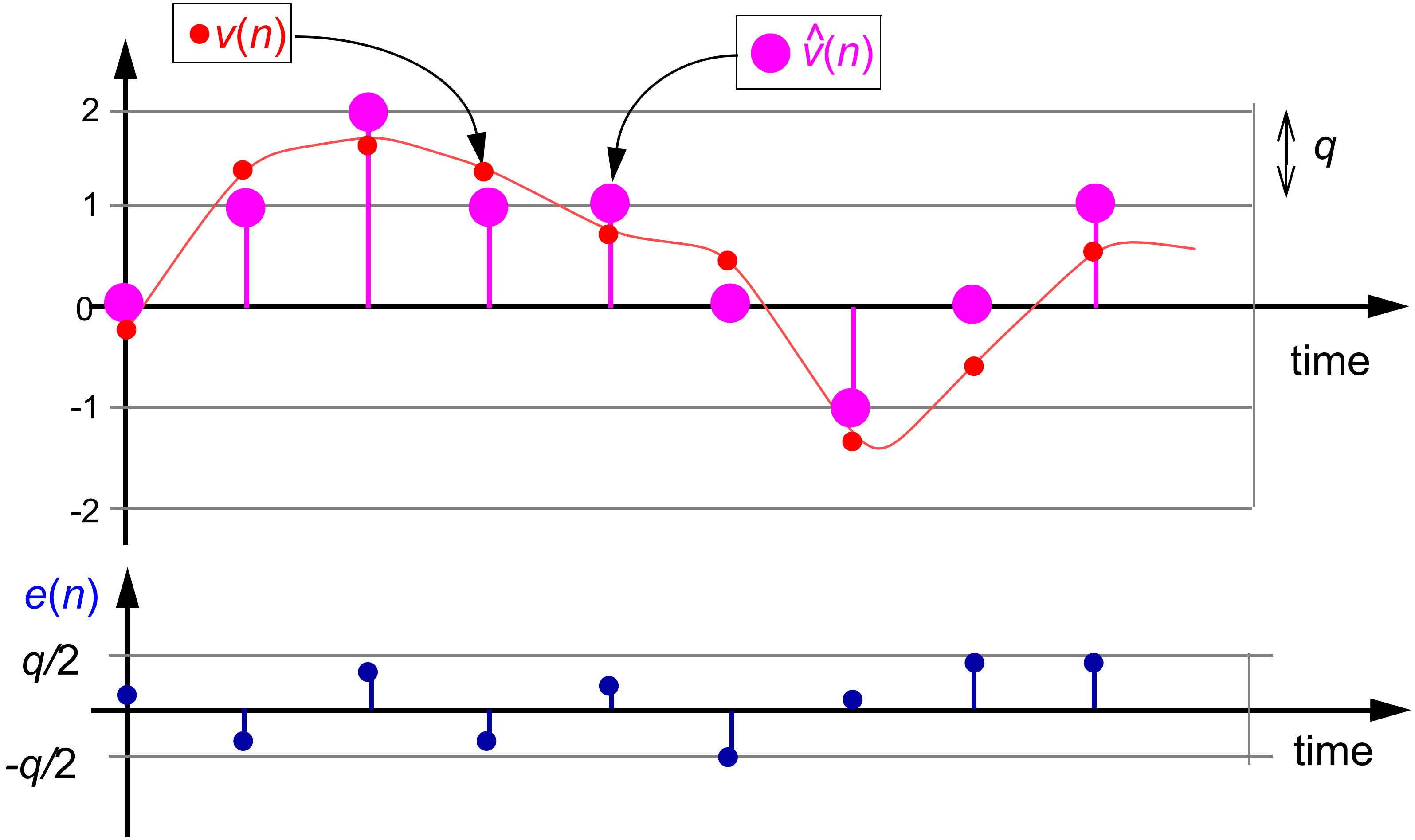
$$\text{Dynamic Range} = 20\log_{10}2^{N} = 20N\log_{10}2 = 6.02N$$

Therefore an 8 bit converter has a range of

Binary 10000000 to 01111111, or in decimal -128 to 127 has a dynamic range of approximately 48 dB.

Quantisation Error in a Signal

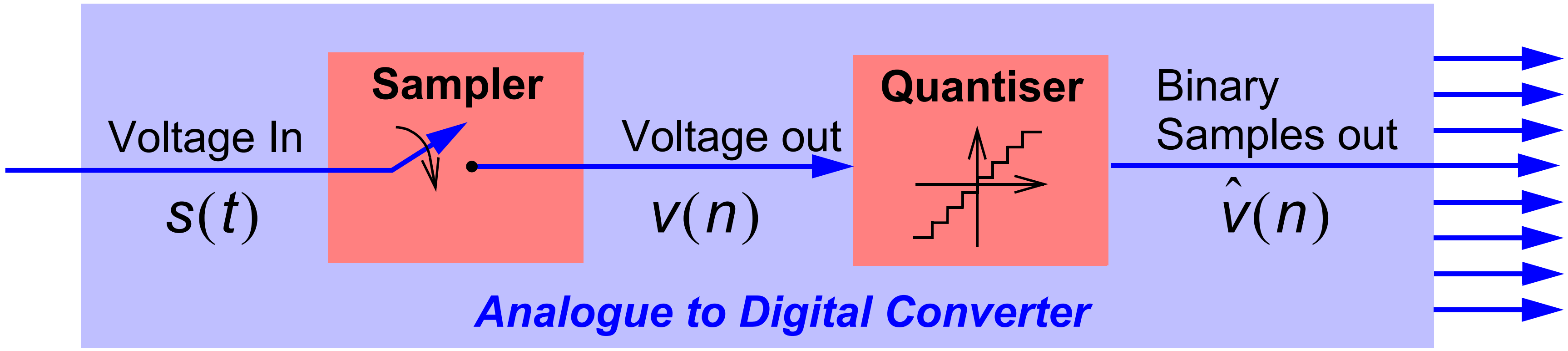
- The ADC output, $\hat{v}(n)$, can therefore be modelled as the perfectly sampled signal, $v(n)$, plus a quantisation error signal denoted, $e(k)$:



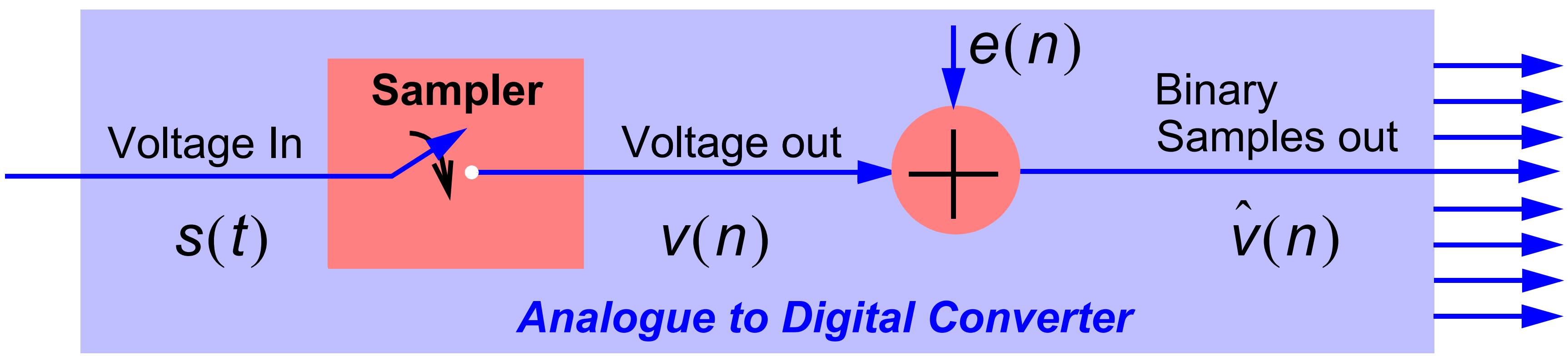
$$\boxed{\hat{v}(n) = v(n) + e(n)}$$

Notes:

The actual ADC can be represented by a sampler and a quantiser:



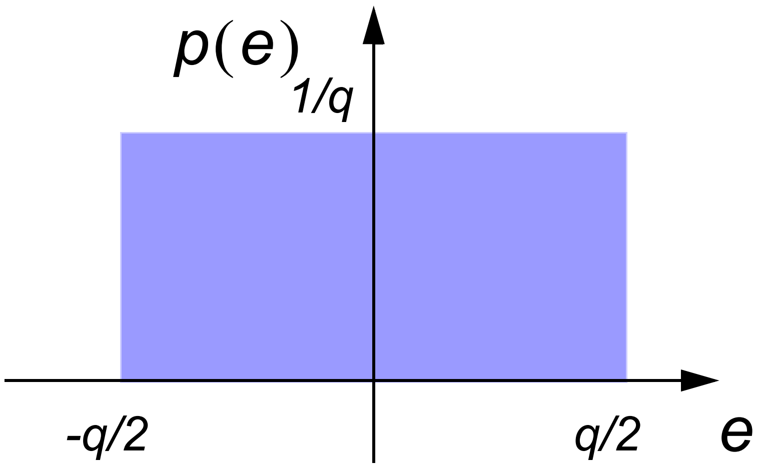
The quantisation error of each sample is in the range $\pm q/2$ and we can model the quantiser as a linear additive noise source.



We can therefore use this as a simple linear equation model of our quantiser where $\hat{v}(n) = v(n) + e(n)$ and $e(n)$ is a white noise source uncorrelated with the input signal $v(n)$. Note that for the actual quantiser it is not possible to write a simple set of mathematical equations to describe/define its input/output. (***Later in the course (Dithering) we will see that the "uncorrelated white quantisation noise" assumption is not actually generally true.....***but thats for later and is a reasonable assumption for now.)

Quantisation Error PDF

- Assuming the ADC rounds to the nearest digital level, the maximum error of any one sample is *q/2* volts.
- If we assume that the probability of the error being at a particular value between +*q/2* and -*q/2* is equally likely then the probability density function (pdf) for the error is flat.

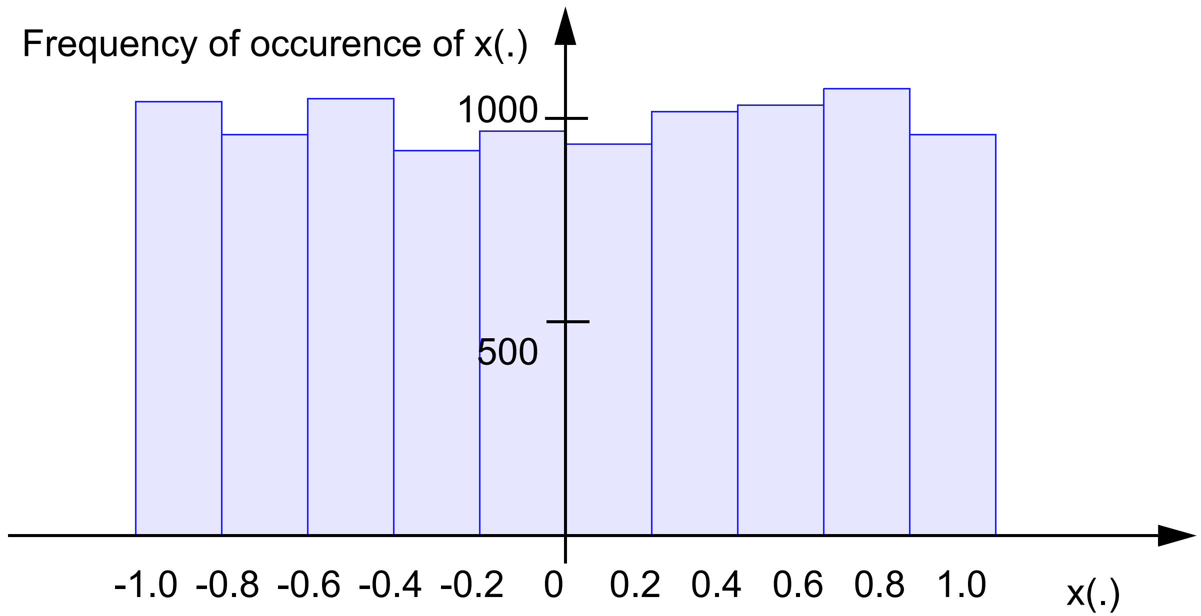


(In practice the quantisation error will be somewhat correlated with the input signal, particularly for low level periodic signals. This will be discussed and addressed later in the dithering section).

Notes:

By taking a uniformly distributed random variable, we can produce a histogram of the output values to confirm the probability density function (pdf).

For example, if a signal source outputs a random variable that is uniformly distributed over the range -1 to 1 and we take outputs and plot a histogram for ranges of 0.2, we may note that after observing, say 10000 samples the histogram of output values is:



The histogram is straightforward to interpret, e.g. in the last 10,000 samples, 983 have been in the range 0 to 0.2. In the limit if the signal is truely generated by a uniformly distributed random noise source, then as we reduce the interval width to 0, and increase the no. of samples to an infinite (!) number, the pdf is as shown in the slide.

Use SystemView to calculate the PDF of a Gaussian noise generator.

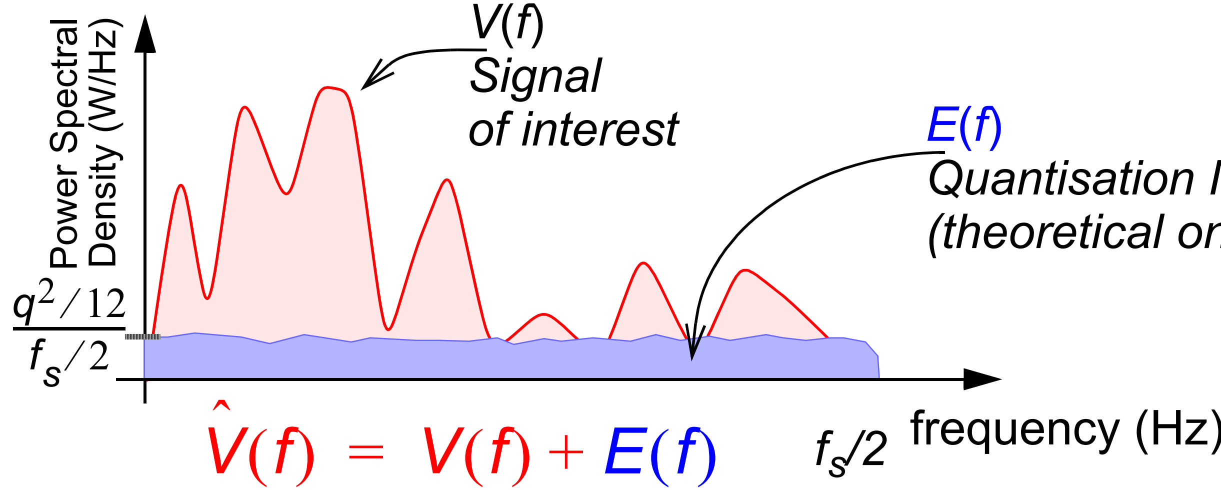
Quantisation Error Power (Statistical)

- Consider the noise power or variance (in a 1 ohm resistor for a 1Hz sampling rate) of the quantisation error signal from:

$$n_{adc} = \int_{-\infty}^{\infty} e^2 p(e)\,de = \int_{-q/2}^{q/2} e^2 p(e)\,de = \frac{1}{3q} e^3 \Big|_{-q/2}^{q/2} = \frac{q^2}{12}$$

- The quantisation error, $E(f)$, will extend over the frequency range 0 to $f_s/2$. , i.e. the full baseband. If $V(f)$ is the signal of interest that is being quantised, then:

Low level signals may be masked by the quantisation noise.



$$\hat{V}(f) = V(f) + E(f)$$

Return

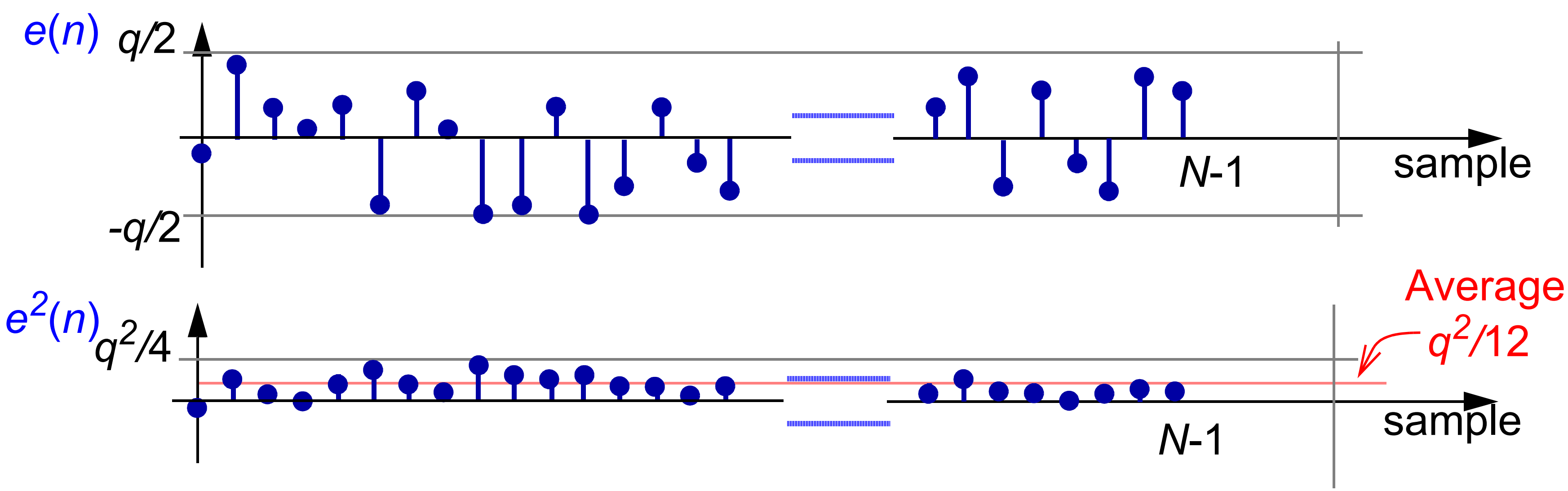
Notes:

The terms "quantisation power" and "quantisation error" are often used interchangeably within DSP. Strictly speaking the quantisation error is the error between one quantised sample, and its true value. Extending this we have the quantisation error signal which is a time versus voltage signal of the error samples - this signal does, of course, not actually exist and is only one that we use for analysis purposes. Collectively we then often refer to the quantisation error signal as the "quantisation noise", or "quantisation noise signal". Arguably the term noise is usually restricted to a signal "noise", which we might expect to be able to remove by filtering. Quantisation "noise" is more correctly termed quantisation "distortion" - once a signal has been quantised there is NO inverse process.

Quantisation Noise Power (Time Average - compare to above Statistical Average)

The noise power can also be calculated from a time average rather than a statistical average. Taking N samples (where N is large) of the quantisation error, the average power is simply:

$$\text{Noise Power} = \frac{\text{Signal Energy}}{\text{Time}} = \frac{1}{N}\sum_{n=0}^{N-1} e^2(k) \approx \frac{q^2}{12} \quad \text{for large } N$$

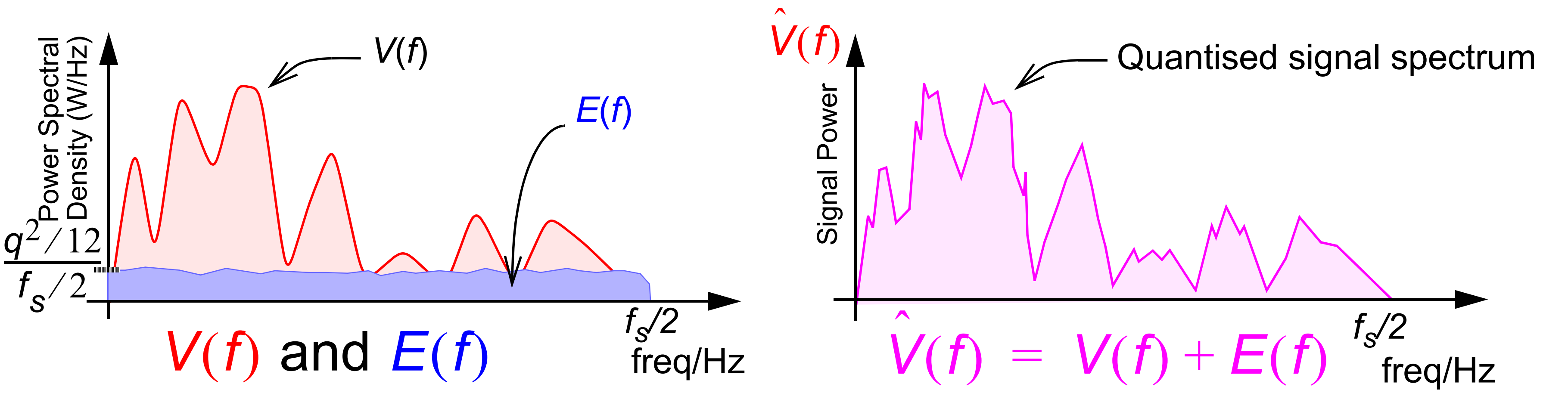


Quantisation Spectra

- After quantisation the signal spectrum will be:

$$\hat{V}(f) = V(f) + E(f) \qquad [\text{ recall } \hat{v}(n) = v(n) + e(n) \,]$$

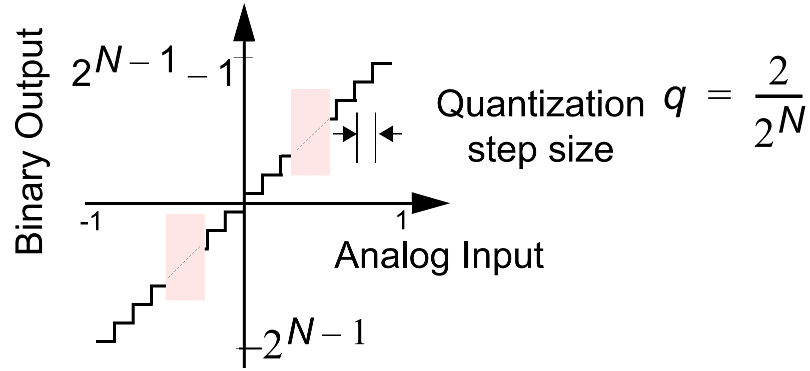
- The quantisation noise will therefore obscure/mask frequency components of interest in the true signal spectrum:



- Compare to previous spectrum on Slide 4.38 .

Notes:

For an *N*-bit signal, there are 2^N levels from the maximum to the minimum value of the quantiser:



Therefore the mean square value of the quantisation noise power can be calculated as:

$$Q_N = 10\log\left(\frac{(2/2^N)^2}{12}\right) = 10\log 2^{-2N} + 10\log\frac{4}{12} \approx -6.02N - 4.77 \text{ dB}$$

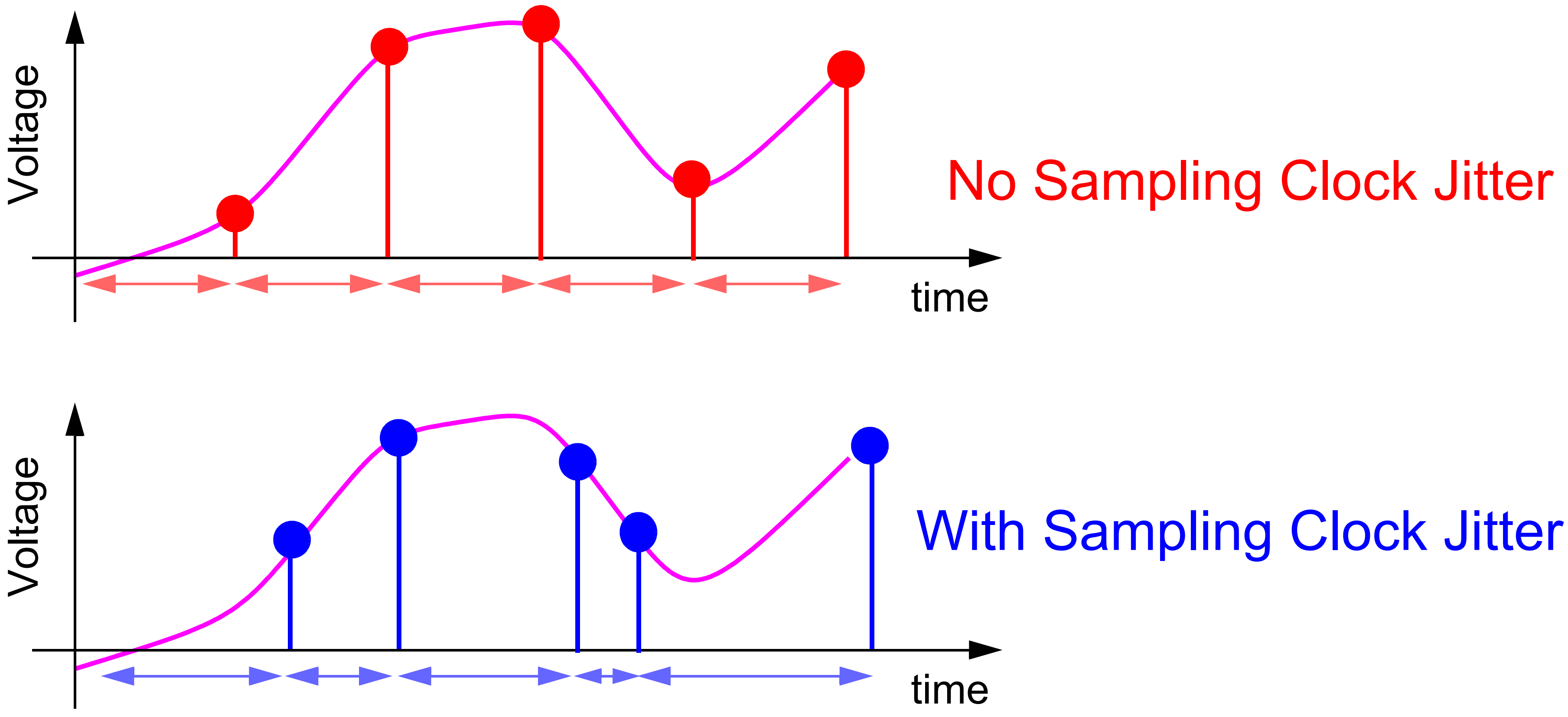
Another useful measurement is the signal to quantisation noise ratio (SQNR). For the above ADC with voltage input levels between -1 and +1 volts, if the input signal is the maximum possible, i.e. a sine wave of amplitude 1 volt, then the average input signal power is: $\text{Signal Power} = E[\sin^2 2\pi ft] = 1/2$. Therefore the maximum SQNR is:

$$\text{SQNR} = 10\log\frac{\text{Signal Power}}{\text{Noise Power}} = 10\log\frac{0.5}{((2/2^N)^2/12)} = 10\log 2^{-2N} + 10\log\frac{3}{2} = 6.02N + 1.76 \text{ dB}$$

For a *perfect* 16 bit ADC the maximum SQNR can be calcuated to be 98.08 dB.

Timing Jitter Error I

- Note that when a signal is sampled there may be some "jitter" on the sampling clock which will cause additional sample error.



- With jitter each sampling instant may be slightly offset, and therefore the sample value obtained and sent to the DSP will be in error.

Notes:

Conclusions

- This session has introduced them fundamental concepts of DSP:
  - Sampling Theorem;
  - Quantisation error and noise model;
  - Aliasing and Reconstruction;
  - The Generic DSP System

Notes: