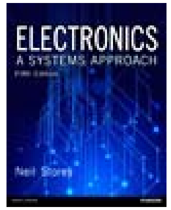
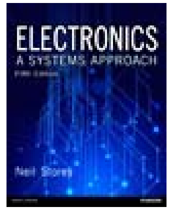


Data acquisition and conversion

Chapter 28

- Introduction
- Sampling
- Signal reconstruction
- Data converters
- Sample and hold gates
- Multiplexing



28.1

Introduction

- Digital techniques have several advantages over analogue methods:
  - They are less affected by noise
  - Processing, transmission and storage is often easier
- However, we often produce or use analogue signals
- Therefore, we often have the need to translate between analogue and digital representations

The sampling system

Abbreviations:

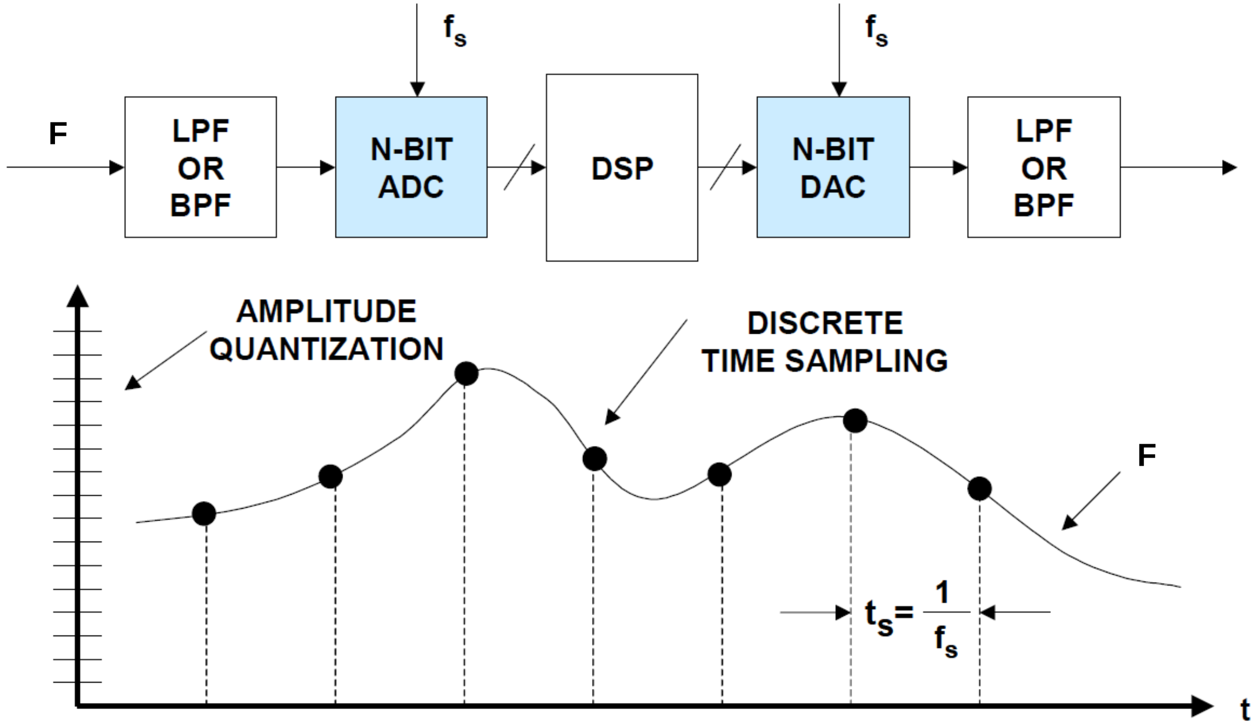
LPF = low pass filter

BPF = band pass filter

ADC = analogue to digital converter

DSP = digital signal processor

DAC = digital to analogue converter



Signal with maximum frequency F is filtered, sampled at frequency f_s, digitized, stored and then output as an analogue signal that is filtered.

The sampling system

Have talked about the DSP
Now we will first talk about the sampling and the filters
Then the ADC/DAC

Abbreviations:

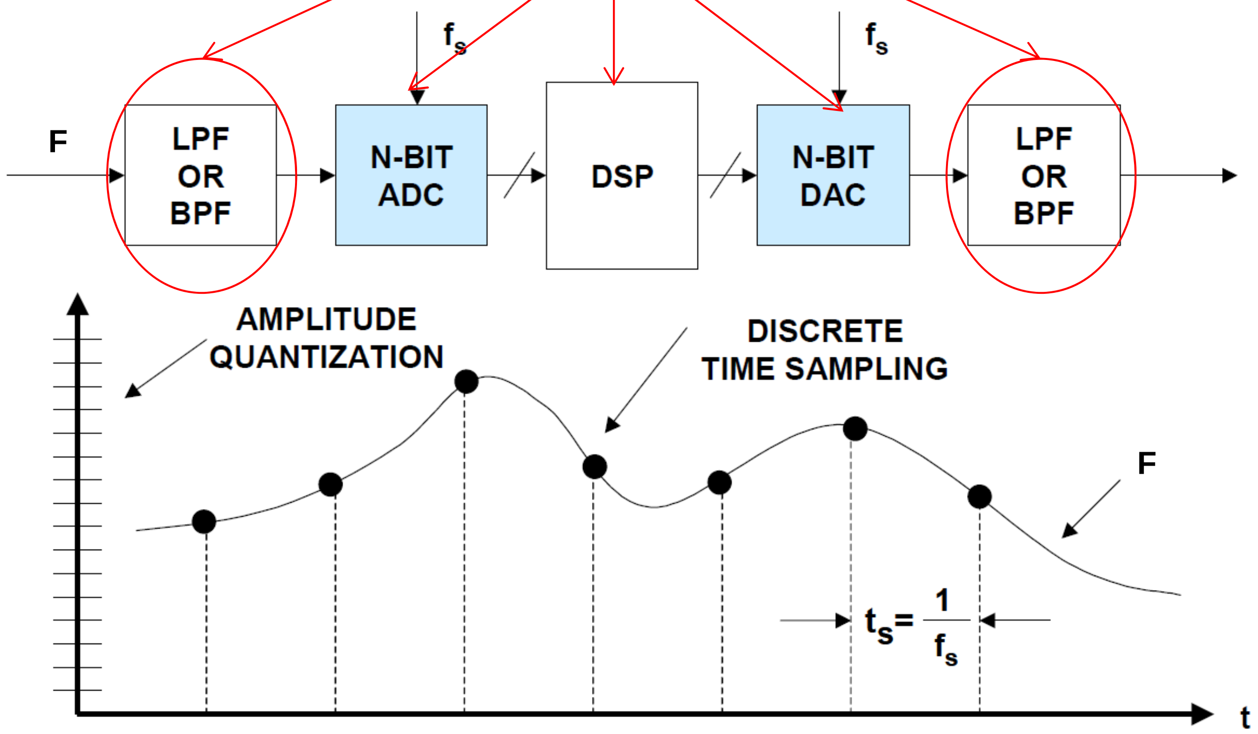
LPF = low pass filter

BPF = band pass filter

ADC = analogue to digital converter

DSP = digital signal processor

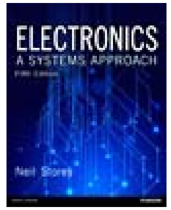
DAC = digital to analogue converter



Signal with maximum frequency F is filtered, sampled at frequency f_s, digitized, stored and then output as an analogue signal that is filtered.

Why the filters: Sampling

Video 28A

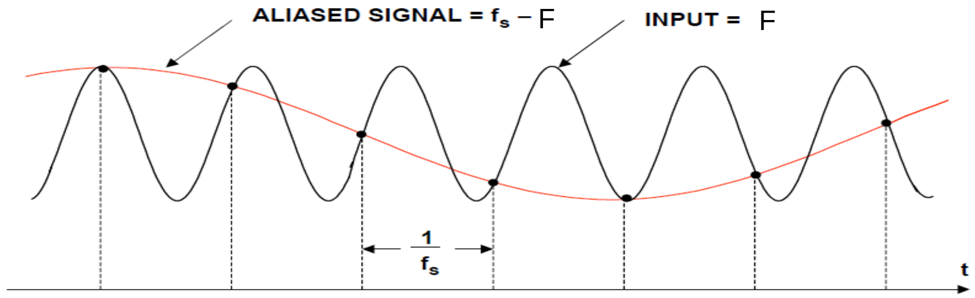


28.2

- In order to obtain a picture of a varying quantity we need to take regular measurements
  – this process is called **sampling**
  – but how often do we need to sample?
- The answer is given by the **Nyquist sampling theorem** which says that:

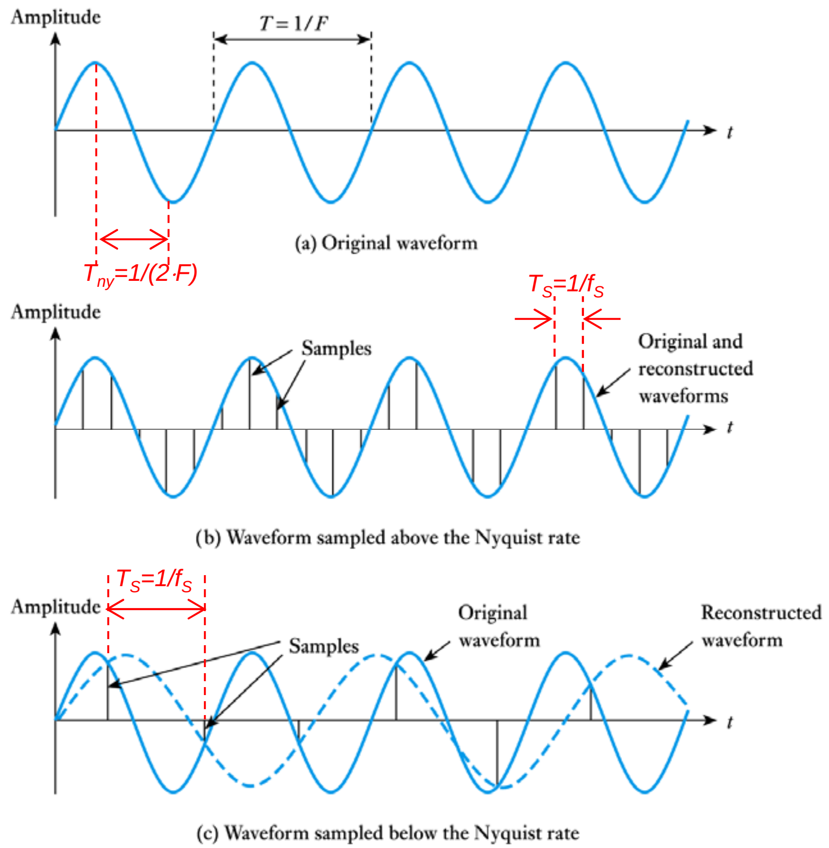
  the sampling rate must be greater than twice the highest frequency present in the signal being sampled.

  – this minimum sampling rate is the **Nyquist rate**



So a high-frequency signal at F
Is read as a low frequency signal at f_s-F

- The effects of sampling rate are illustrated here:
  - (a) shows the original signal
  - (b) shows the effects of sampling at a rate *above* the Nyquist rate
  - (c) shows the effects of sampling at a rate *below* the Nyquist rate (f_{ny}=F/2)
  - (d) Sample at the frequency of the sine wave, F, interpret as f=0 again! Second harmonic



(a) Original waveform

(b) Waveform sampled above the Nyquist rate

(c) Waveform sampled below the Nyquist rate

“Negative” frequencies or harmonics

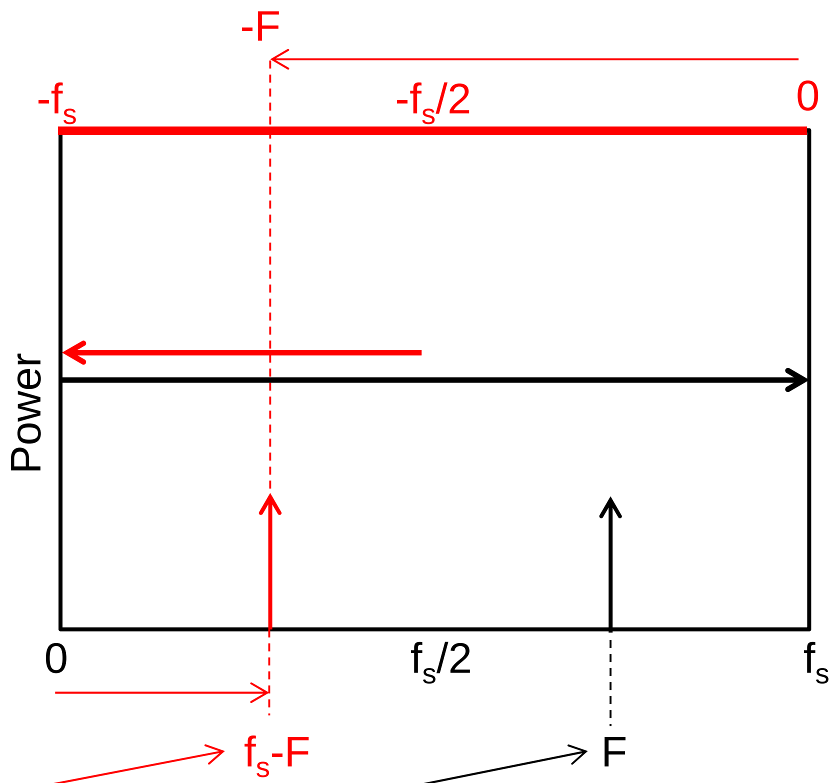
Fourier Transform

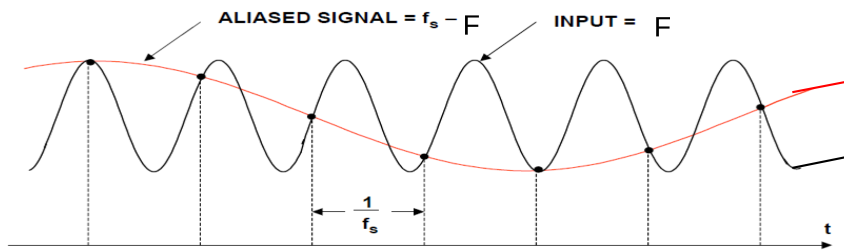
Shows power vs. frequency of sine wave at frequency F

Add a reverse scale of negative frequencies

A sine wave with high, negative frequency –F

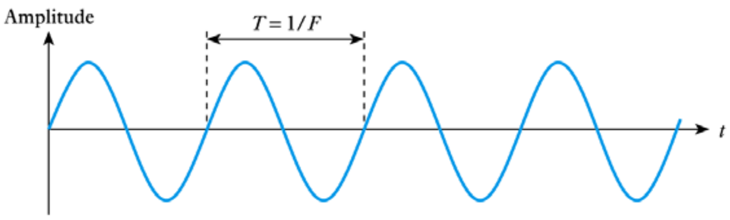
Is heard as a low frequency sine wave at f_s-F



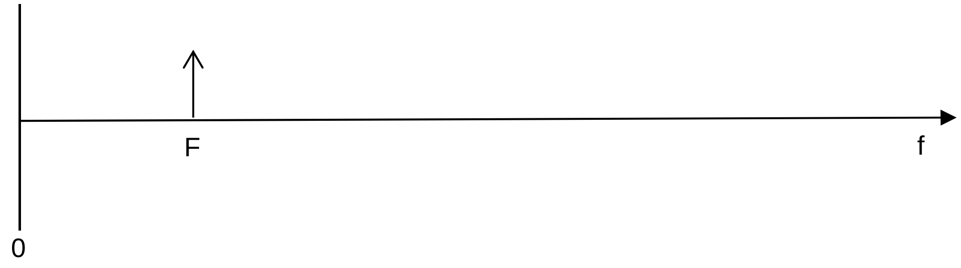


Sine sweep 0-44kHz sampled at 22kHz

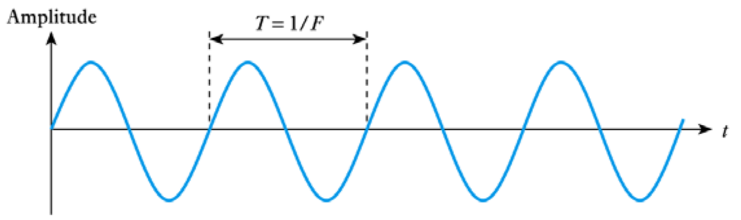
How does this look in frequency?



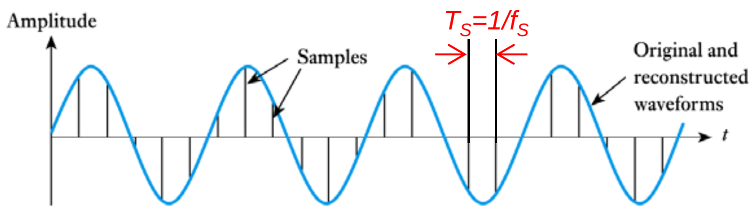
(a) Original waveform



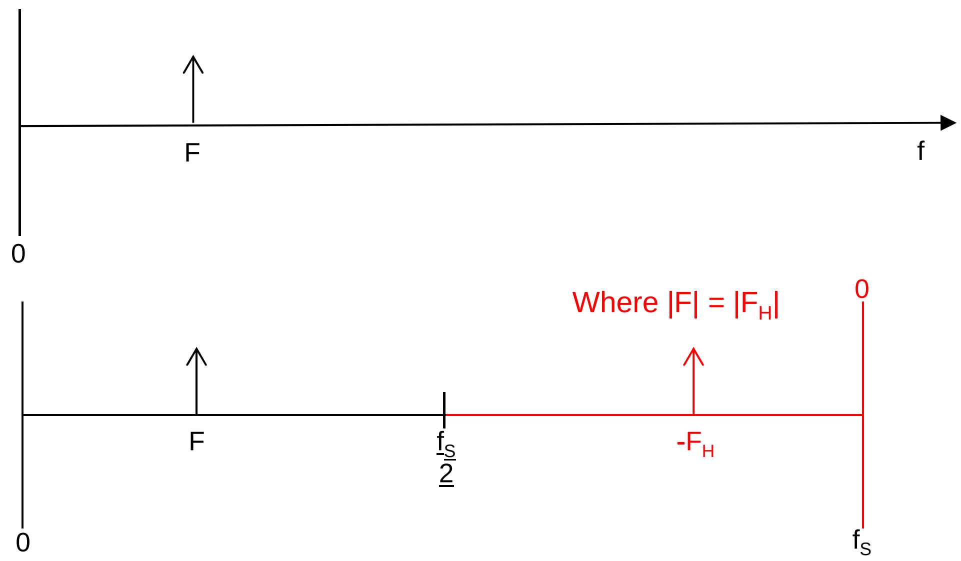
How does this look in frequency?



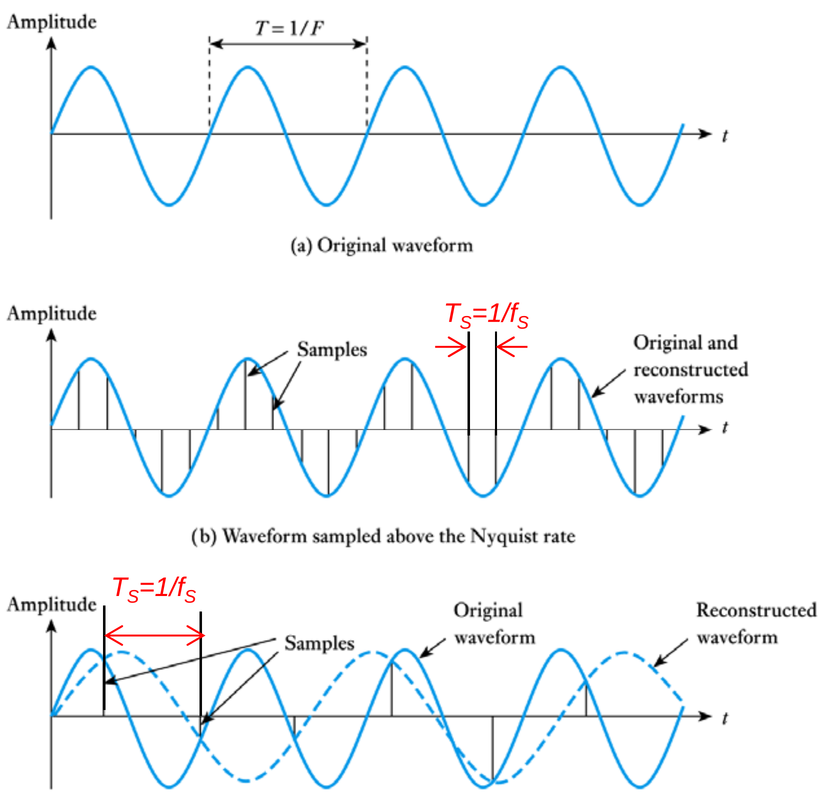
(a) Original waveform



(b) Waveform sampled above the Nyquist rate



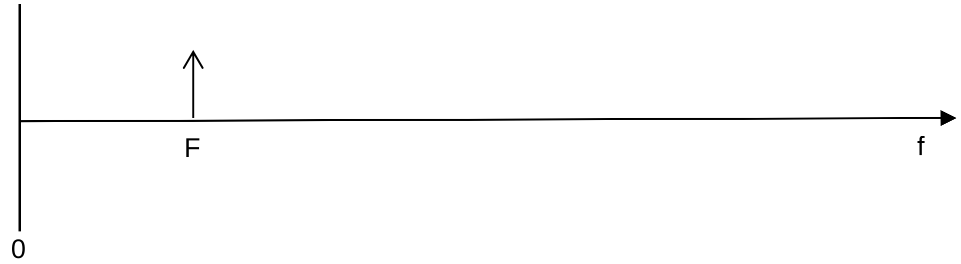
How does this look in frequency?

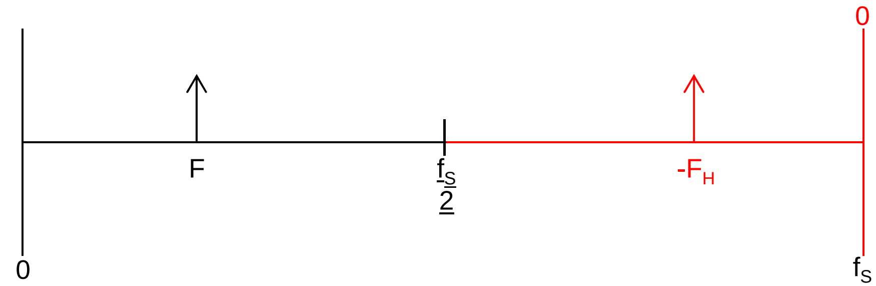


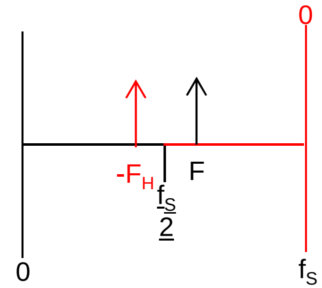
(a) Original waveform

(b) Waveform sampled above the Nyquist rate

(c) Waveform sampled below the Nyquist rate



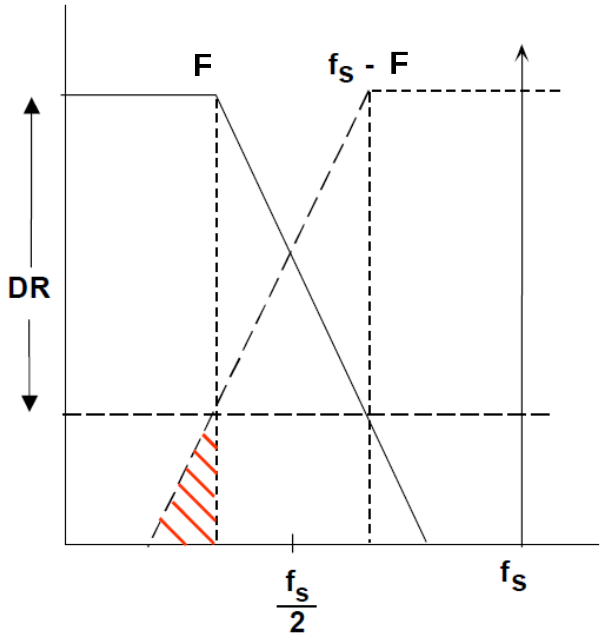




Now the 1st harmonic (at $-F_H$)
appears at $f_a = f_S - F$
It is *"aliased"* about the Nyquist
frequency = $f_S/2$
We only really see 0-f_{Ny}

- ***Note:*** the sampling rate is determined by the highest frequency present in the signal, *not* the highest frequency of interest
- If a signal contains unwanted high frequency components, *we will see these as aliased frequencies*. So they should be removed before sampling
  - this is done using a low-pass filter
  - such a filter is called an **anti-aliasing filter**
- It is common to sample at about 20% or more above the Nyquist rate to allow for imperfect filtering

Example of filtering



STOPBAND ATTENUATION = DR
TRANSITION BAND: F to f_s - F
CORNER FREQUENCY: F

Specify the attenuation of the first harmonic in our "signal" band, $0 \rightarrow F$. ($F \approx 0.8 \cdot f_s/2 = 0.8 \cdot f_{ny}$)

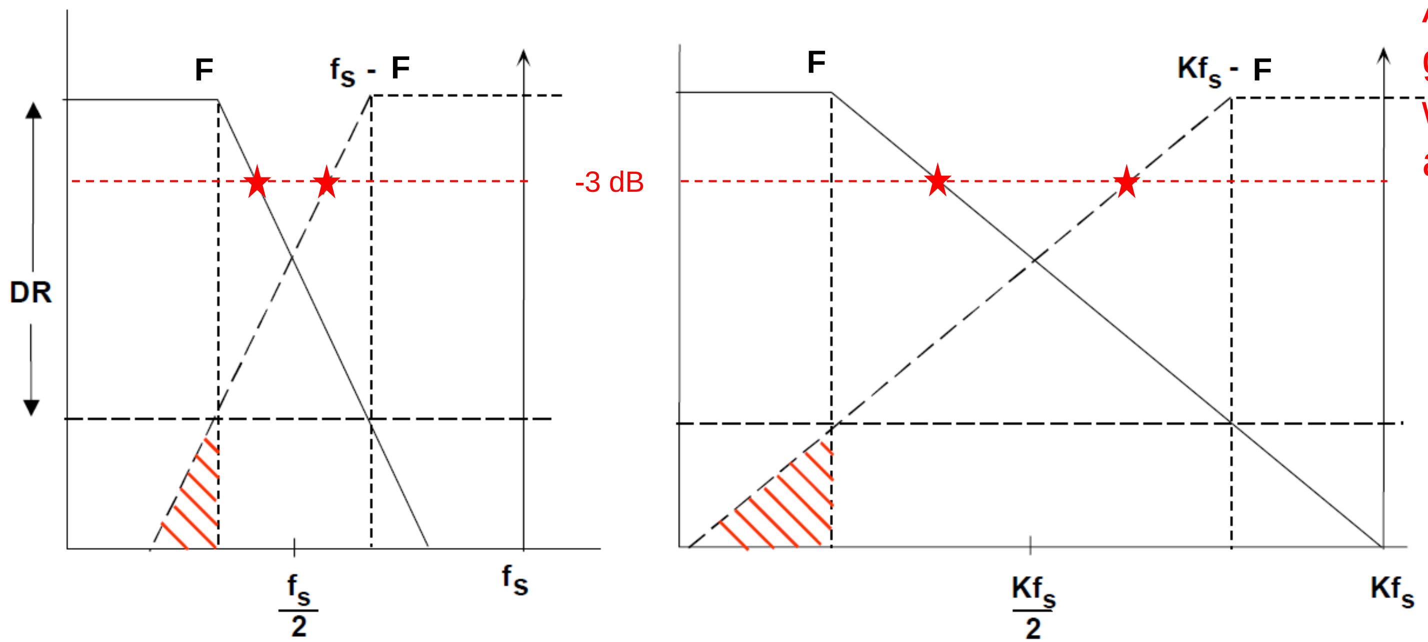
Will still have those frequencies, but at an attenuation of DR.

Usually want as many frequencies as possible ($F \rightarrow f_s/2$), but that would require a sharp filter ("brick wall"),

Relax to ~80% of f_{ny}

Example of filtering and "oversampling"

A more gentle filter with same attenuation



STOPBAND ATTENUATION = DR
TRANSITION BAND: F to f_s - F
CORNER FREQUENCY: F

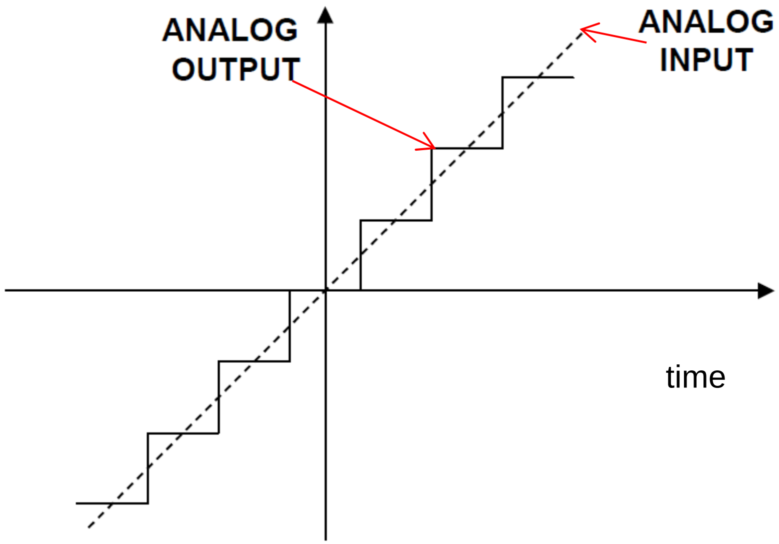
STOPBAND ATTENUATION = DR
TRANSITION BAND: F to Kf_s - F
CORNER FREQUENCY: F

Remember, a LPF will have a -45° phase shift at the -3 dB point

If you want the violin at the same time as the bass, move this as far out as possible

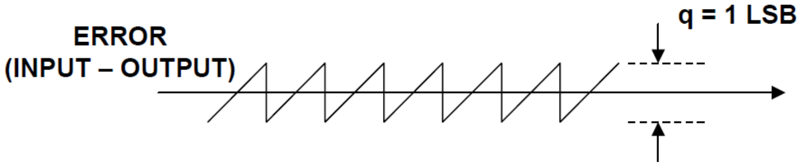
Assume we have digitized and now want to follow steps to reconstruct the signal.

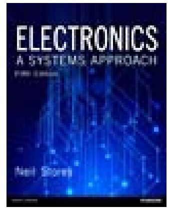
- Have anti-aliased the input (some band playing)
- Converted it to digital numbers (and stored on computer)
- Now reconvert to analogue signal (to play back on speakers)



The input ramp has been "quantized" into a digital signal

These "quanta" appear at the output giving us steps





Signal reconstruction

- In many cases it is necessary to reconstruct an analogue signal from a series of sample
  – typically after they have been processed, transmitted or stored
- This requires the removal of the step transitions in the sampled waveform (step $\Rightarrow$ high frequencies)
- Reconstruction is achieved using a low-pass filter to remove these unwanted frequencies
  – this filter is called a **reconstruction filter**

Hence, the sampling system

Abbreviations:

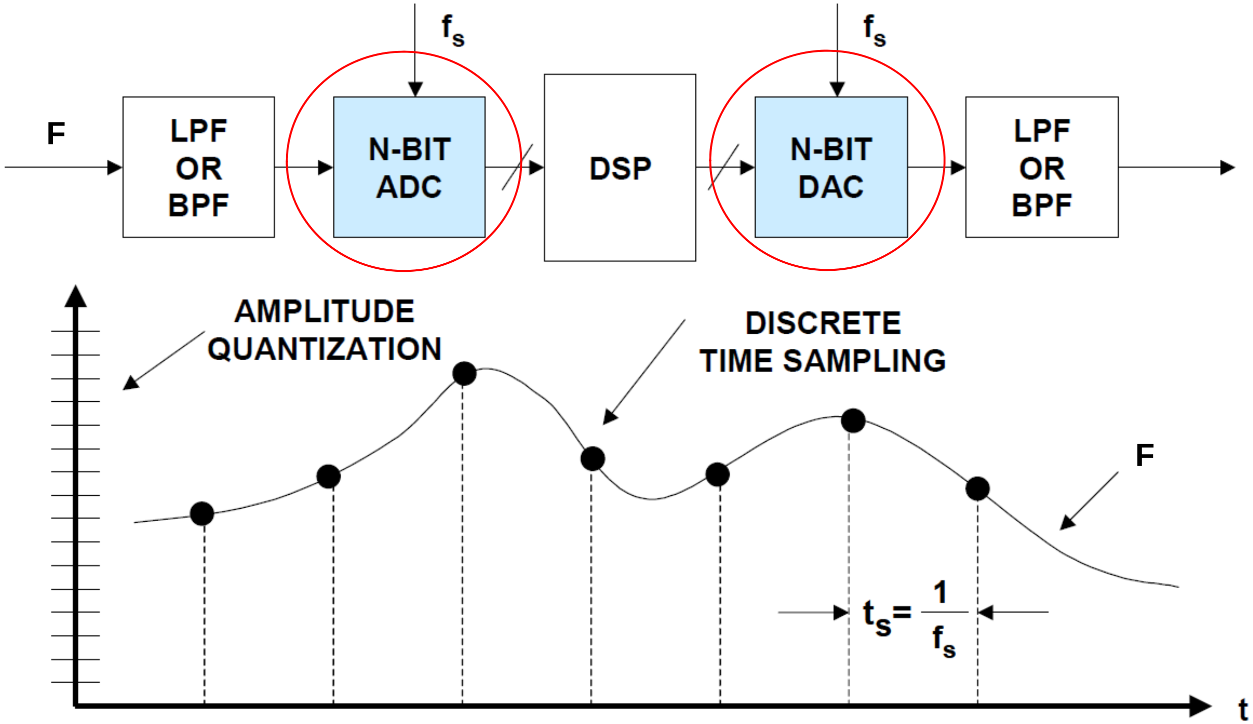
LPF = low pass filter

BPF = band pass filter

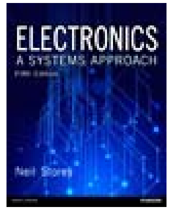
ADC = analogue to digital converter

DSP = digital signal processor

DAC = digital to analogue converter



Signal with maximum frequency F is filtered, sampled at frequency f_s, digitized, stored and then output as an analogue signal and filtered.



28.4

What about the data converters!

- *Sampling* involves taking a series of instantaneous measurements of a signal and converting these into a digital form
- *Reconstruction* involves taking a series of digital readings and converting these back into their analogue equivalents
- These two operations are performed by **data converters** which can be of two basic types:
  - **Analogue-to-digital converters (ADC'S)**
  - **Digital-to-analogue converters (DAC'S)**

Resolution of data converters

– a range of converters is available, each providing conversion to a particular **resolution**

– this determines the number of **quantization levels** used

– an *n*-bit converter uses 2^n discrete steps

e.g. an 8-bit converter uses 2^8 or 256 levels

a 10-bit converter uses 2^{10} or 1024 levels

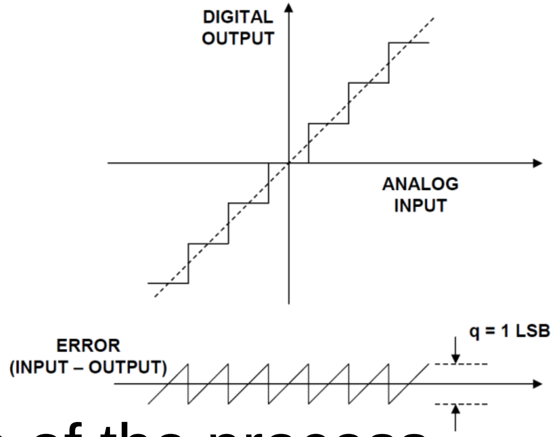
– an 8-bit converter gives a resolution of about 0.4%

(volts/step = FullScaleVolts/2^8. Resolution = volts/step/FSV *100)

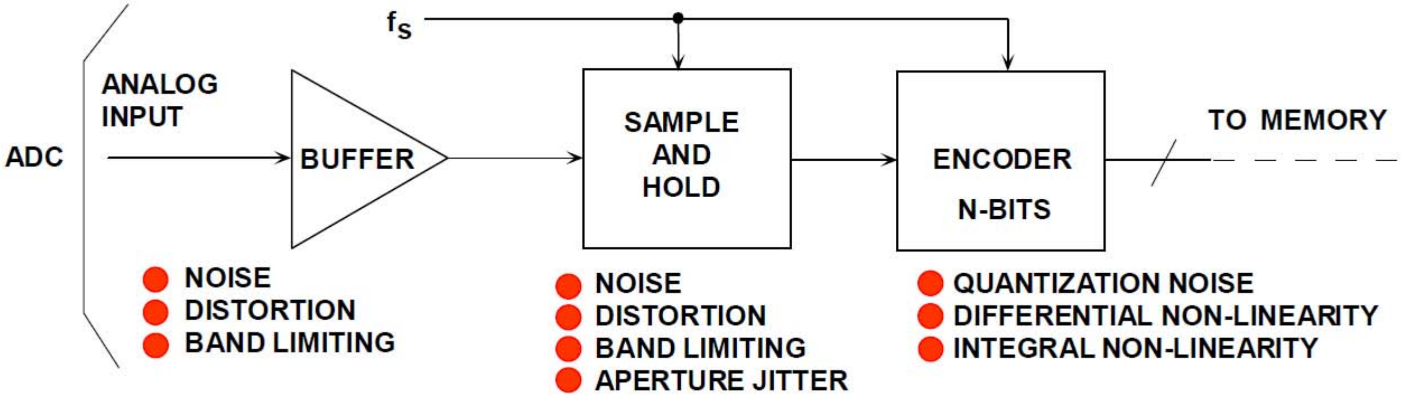
– where greater resolution is required converters with up to 20-bit resolution or more are available

Resolution vs. accuracy

- The resolution represents a "quantization" noise
- Frequently quoted as ±½ LSB



- Accuracy will depend upon the total noise of the process, which depends on the entire chain (typically ±1.5 LSB)



Reverse the chain through DAC for output

How fast can we digitize or reconstruct a signal?

- Determines what frequencies we can sample
- **Speed of conversion**
  - – conversions of ADC's or DAC's take a finite time
  - – this is referred to as the **conversion time** or **settling time** of the converter
  - – the time taken depends on the converter
  - – DACs are usually faster than ADCs

Video 28B

Digital-to-analogue converters (DACs)

- wide range of resolutions, but in general, conversion time *increases with resolution*
- a typical general-purpose 8-bit DAC would have a settling time of between 100 ns and 1 μs
- a typical 16-bit converter would have a settling time of a few milliseconds
- for specialist applications high-speed converters have settling times of a few nanoseconds
- a video DAC might have a resolution of 8 bits and a maximum sampling rate of 100 MHz

– a binary-weighted resistor DAC

A "1" on an input tries to draw current from the op amp junction (must come from V_o since input of op amp does not draw or source current)

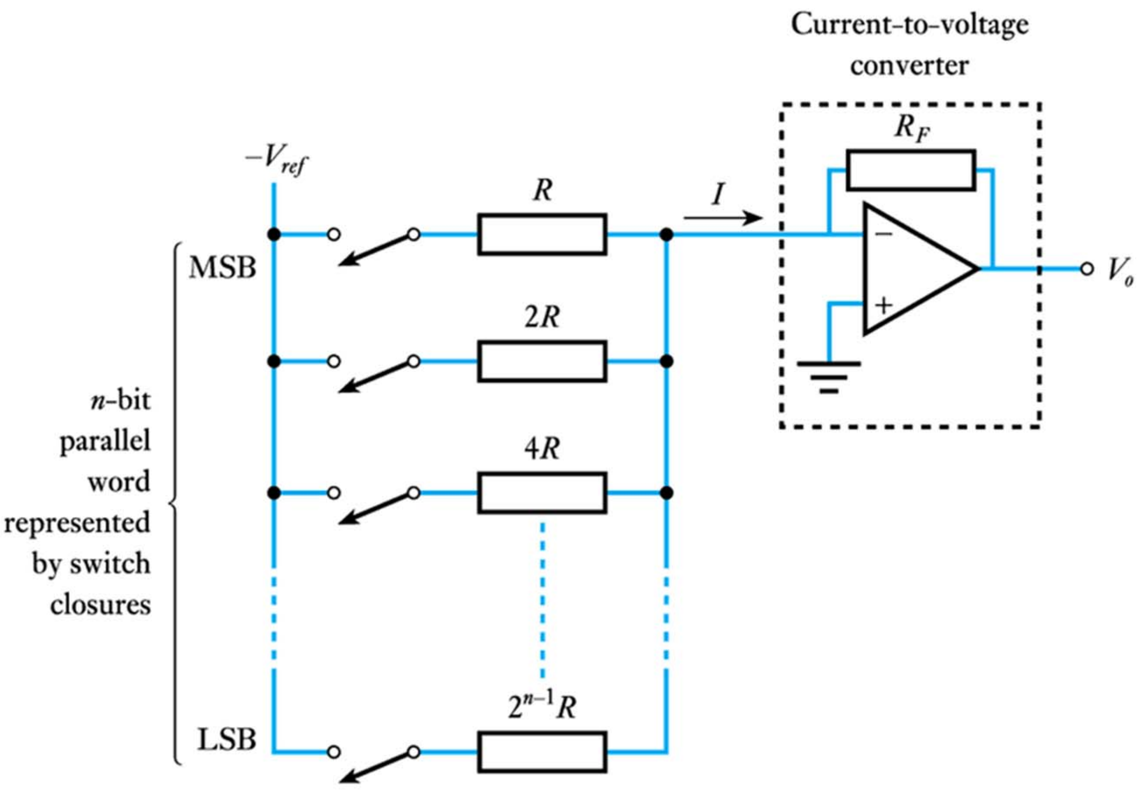
Op amp changes V_o to null the current at the input

Just an inverting amplifier whose gain depends on which bits are 1.

Gain decreases from MSB to LSB

So for a binary input number m:

$$V_o = m \times \frac{V_{ref} R_F}{2^{n-1} R}$$



Same idea, but same resistors for better thermal control

– an *R-2R* resistor chain DAC

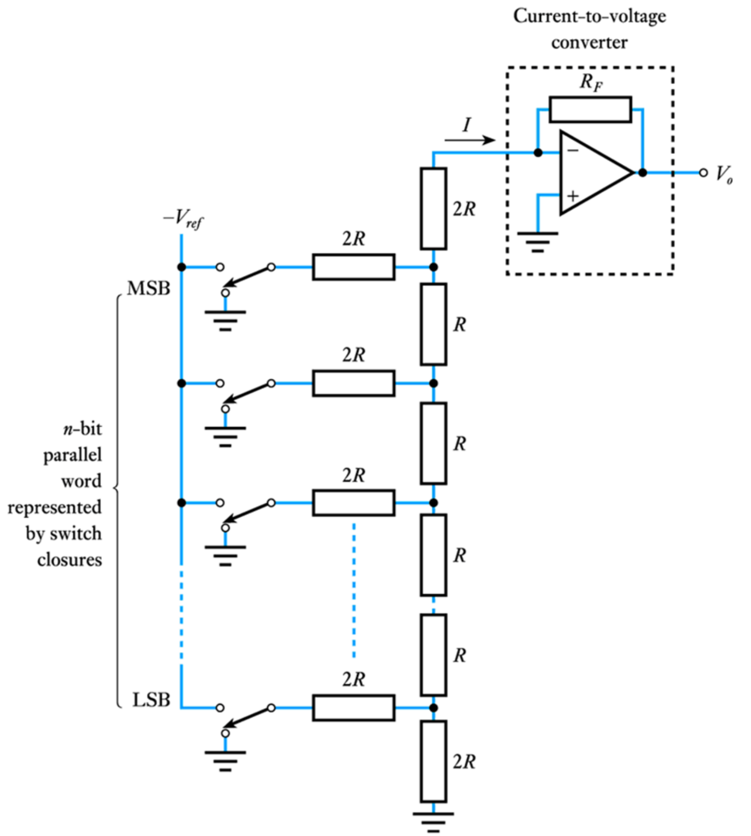
A 1 on an input will cause a current to flow through the 2R resistor attached to the switch.

At the junction, this current sees an equivalent resistance 2R in each direction and is split equally

As it travels up the chain, it is split again at each junction

Coming from LSB it sees the most splits⇒this gives the least current at op-amp

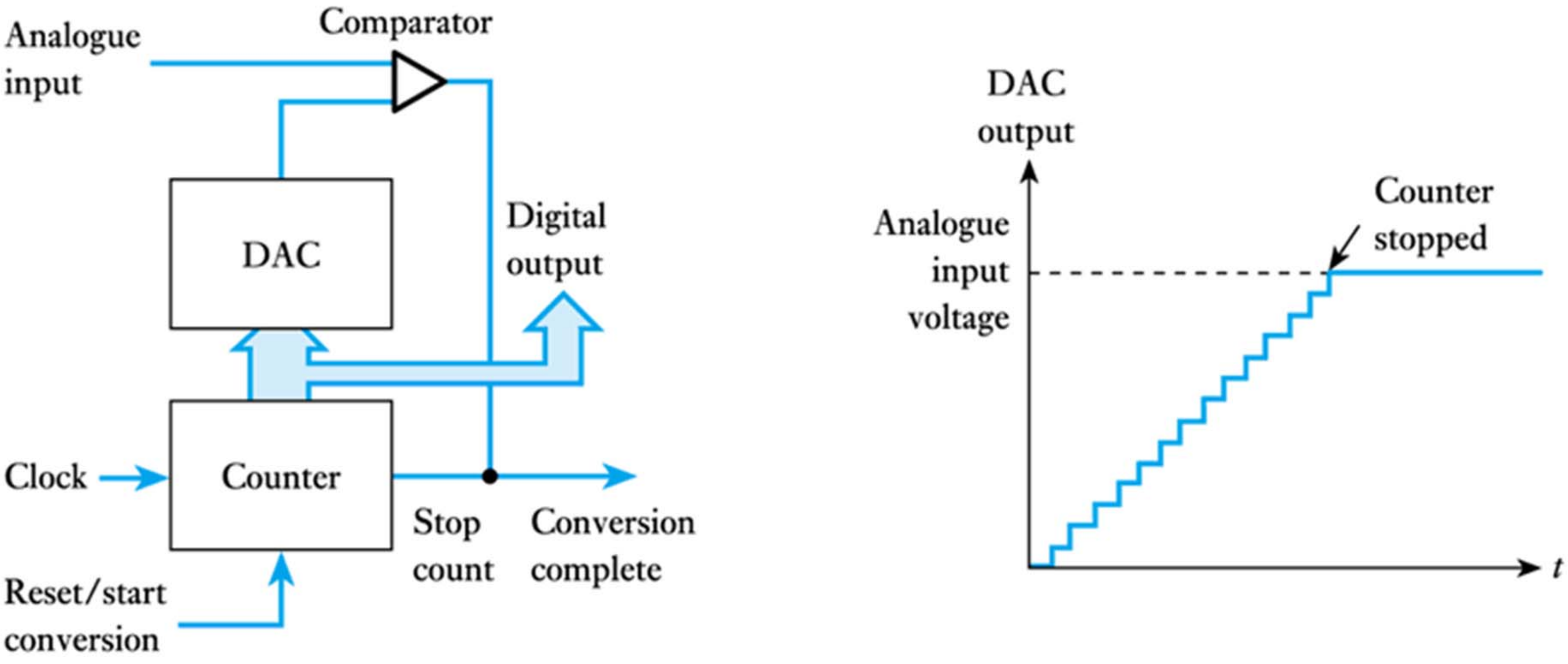
Op amp changes Vo to offset (null) the current at the input



- **Analogue-to-digital converters (ADCs)**
  - again available in a range of resolutions and speeds
  - a typical 8-bit converter might have a settling time of between 1 and 10 μs (~10x longer than a DAC)
  - a typical 12-bit converter might have a settling time of 10 to 100 μs
  - But, high speed converters can exceed 150 million samples per second

– a counter or servo ADC

Slow but cheap-few ms



(a) Block diagram

(b) DAC waveform

Fast and still cheap

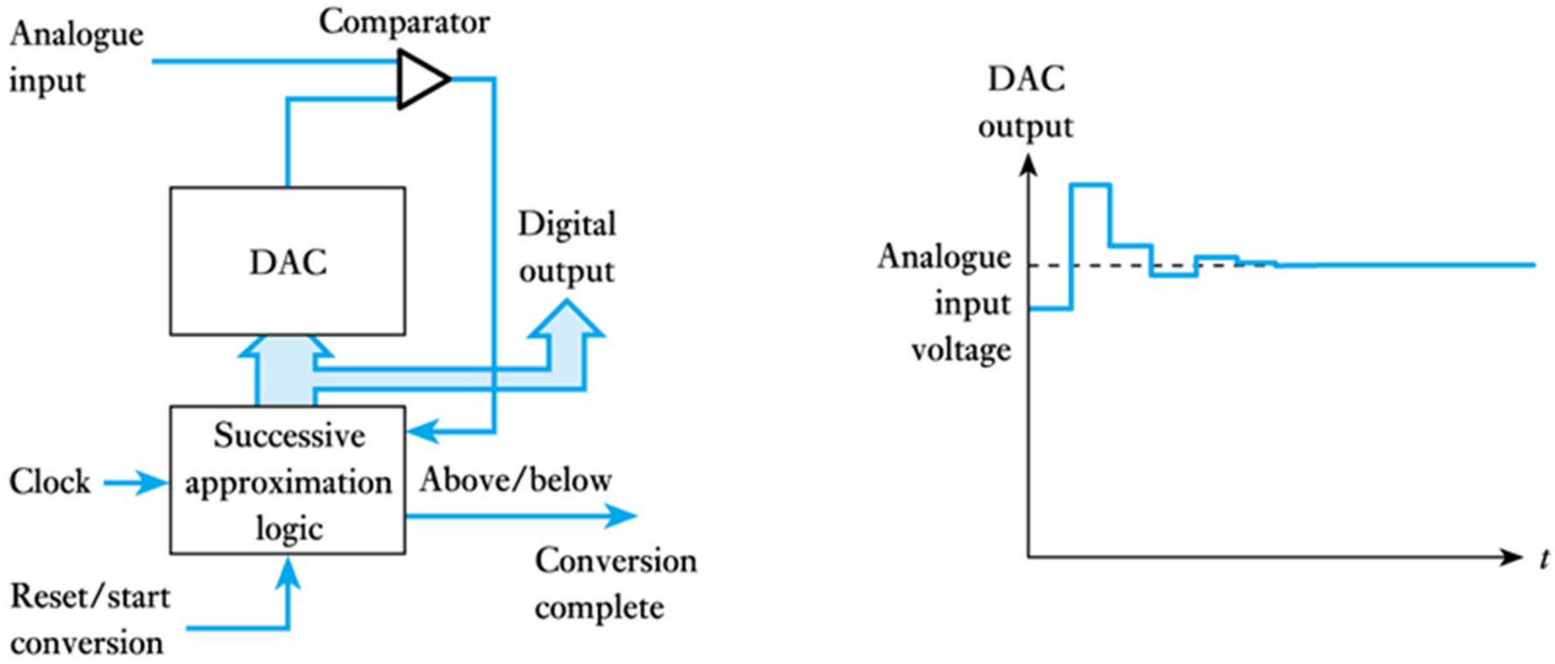
1-100 μs for 8-12 bits

Most common type

– a successive approximation ADC (bisection)

– MSB=1, DAC = ½ scale, if less than input, keep MSB=1 & turn on next bit

– If more than input, reset bit to 0 and turn next bit on, keep going



(a) Block diagram (b) DAC waveform

V_i charges capacitor

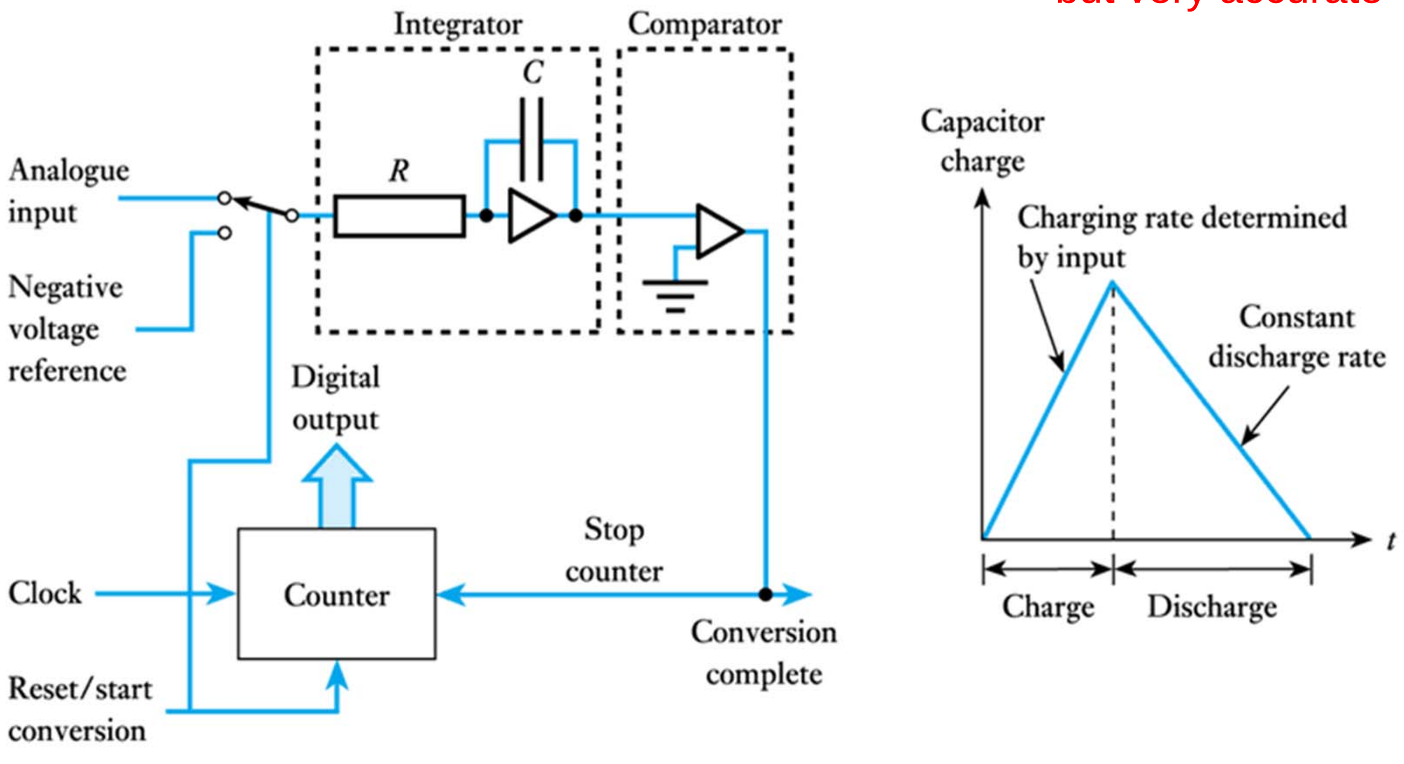
Discharge at constant rate

Counter starts at start of discharge and stops at full discharge

Time = count = V_i

– a dual-slope ADC

Slow (10-100 ms)
but very accurate

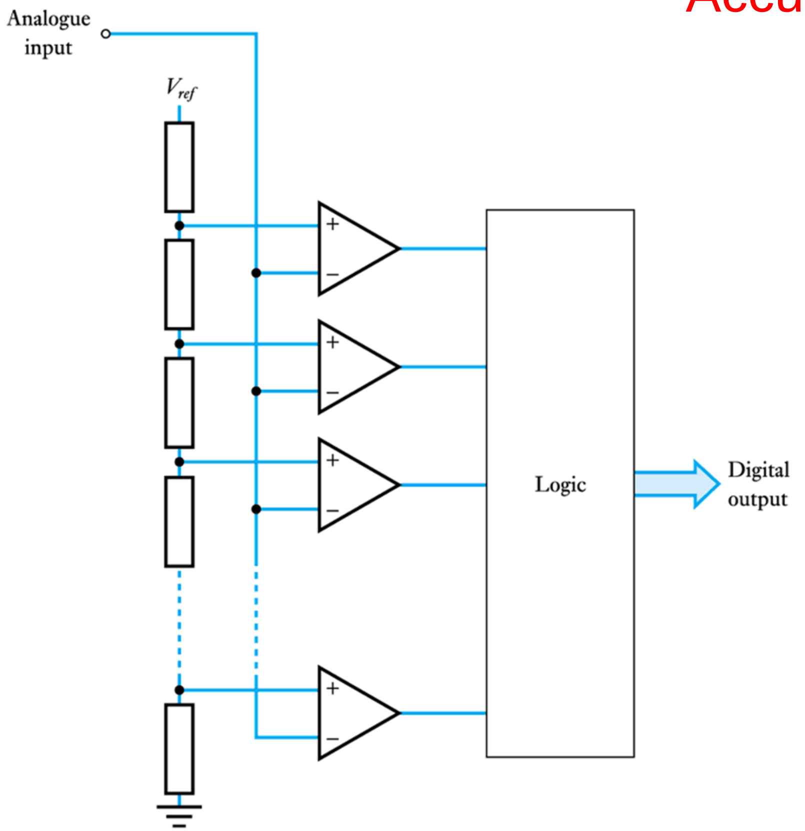


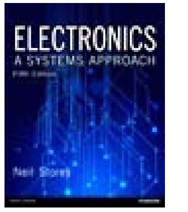
(a) Block diagram (b) Capacitor waveform

Comparators connected to voltages with $V > V_i$ produce one polarity,
All connected to $V < V_i$ produce the opposite polarity.
Combinational logic is then used to determine the value of the input voltage from this pattern.

– a parallel or flash ADC

Go fast converter (ns)
Accurate and expensive



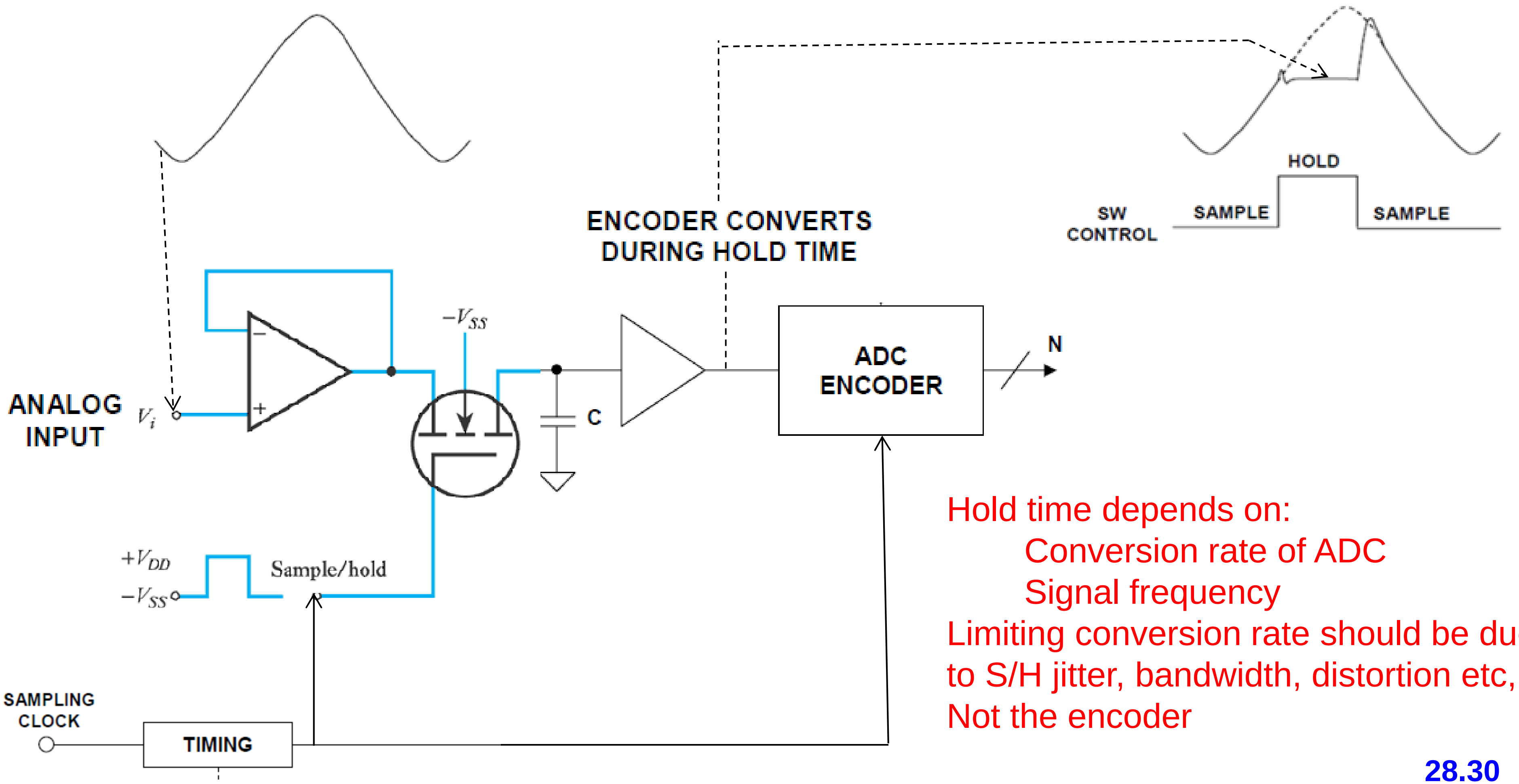


28.5

Sample and hold gates

- It is often useful to be able to *sample* a signal and then *hold* its value constant
  - this is useful when performing analogue-to-digital conversion so that the signal does not change during conversion
  - it is also useful when doing digital-to-analogue conversion to maintain the output voltage constant between conversions
- This task is performed by a **sample and hold gate**
  - we have looked at such circuits in earlier lectures

A FET sample and hold gate



Hold time depends on:

Conversion rate of ADC

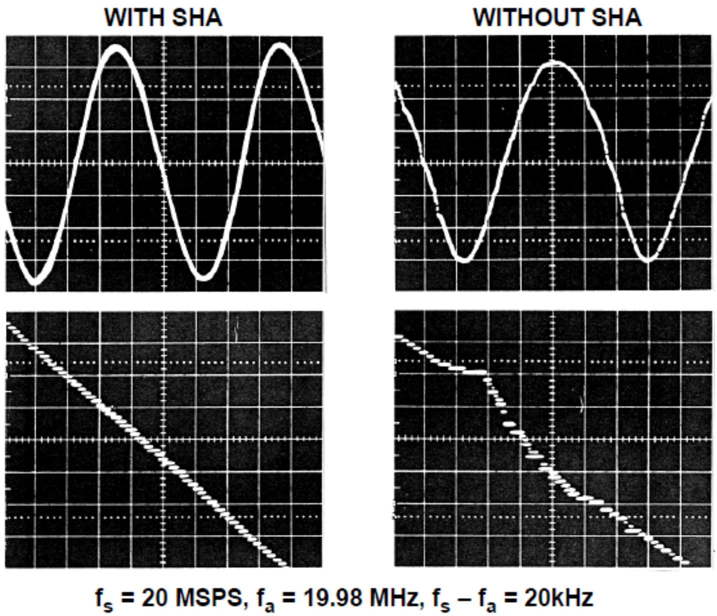
Signal frequency

Limiting conversion rate should be due to S/H jitter, bandwidth, distortion etc, Not the encoder

- Most sample and hold gates are constructed using integrated circuits
- Typical devices require a few microseconds to sample the incoming waveform, which then decays (or **droops**) at a rate of a few millivolts per millisecond
- High speed devices, such as those used for video applications, can sample an input in a few nanoseconds, so droop must be limited to a few millivolts per microsecond

Advantage of S/H

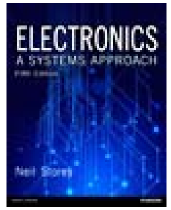
- If signal is changing during conversion, increases bit errors, missing codes, and accentuates timing mismatches in comparators of flash ADC.



Encoder output

Output vs. input.

Figure 2.28: *8-bit, 20-MSPS Flash ADC With and Without Sample-and-Hold*

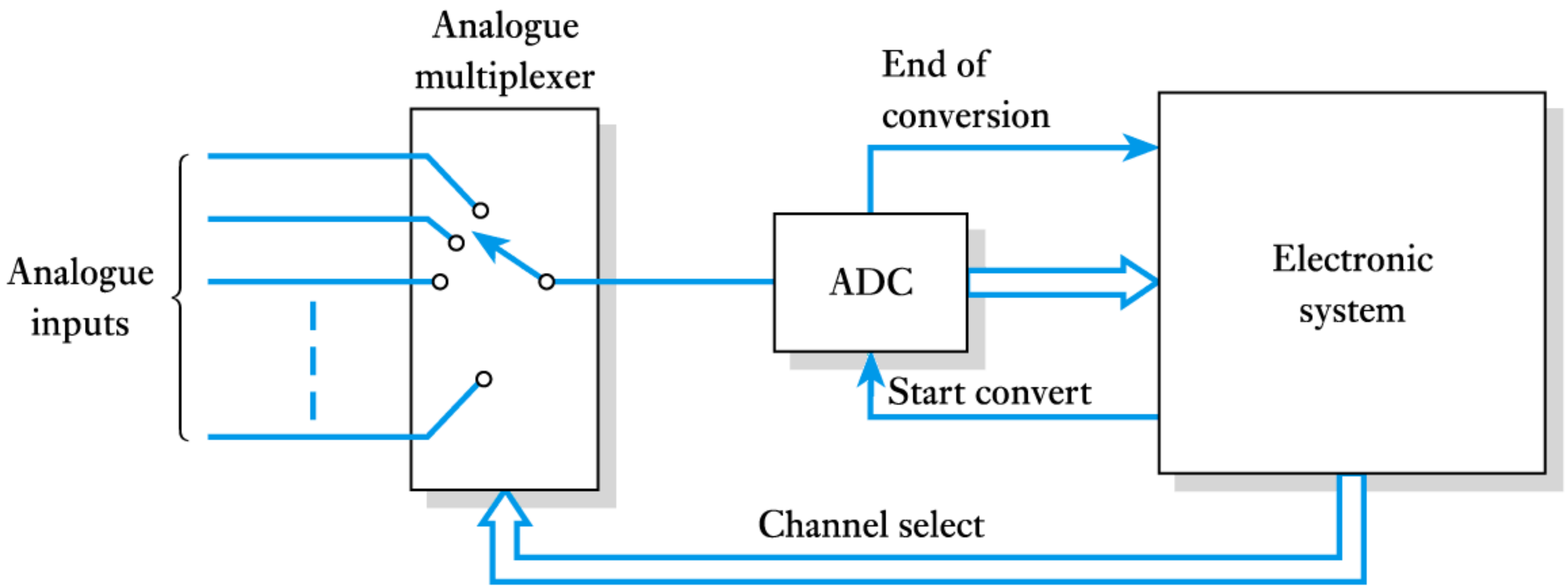


Multiplexing

- While some systems have a single input and a single output, often there are multiple inputs and outputs
- Rather than have separate converters for each input and output, we often use **multiplexing**
  - multiplexers make use of **electrically operated switches** to control the routing of signals
  - these can be used at the input or output of a system
  - normally separate anti-aliasing and/or reconstruction filters would be used with each input

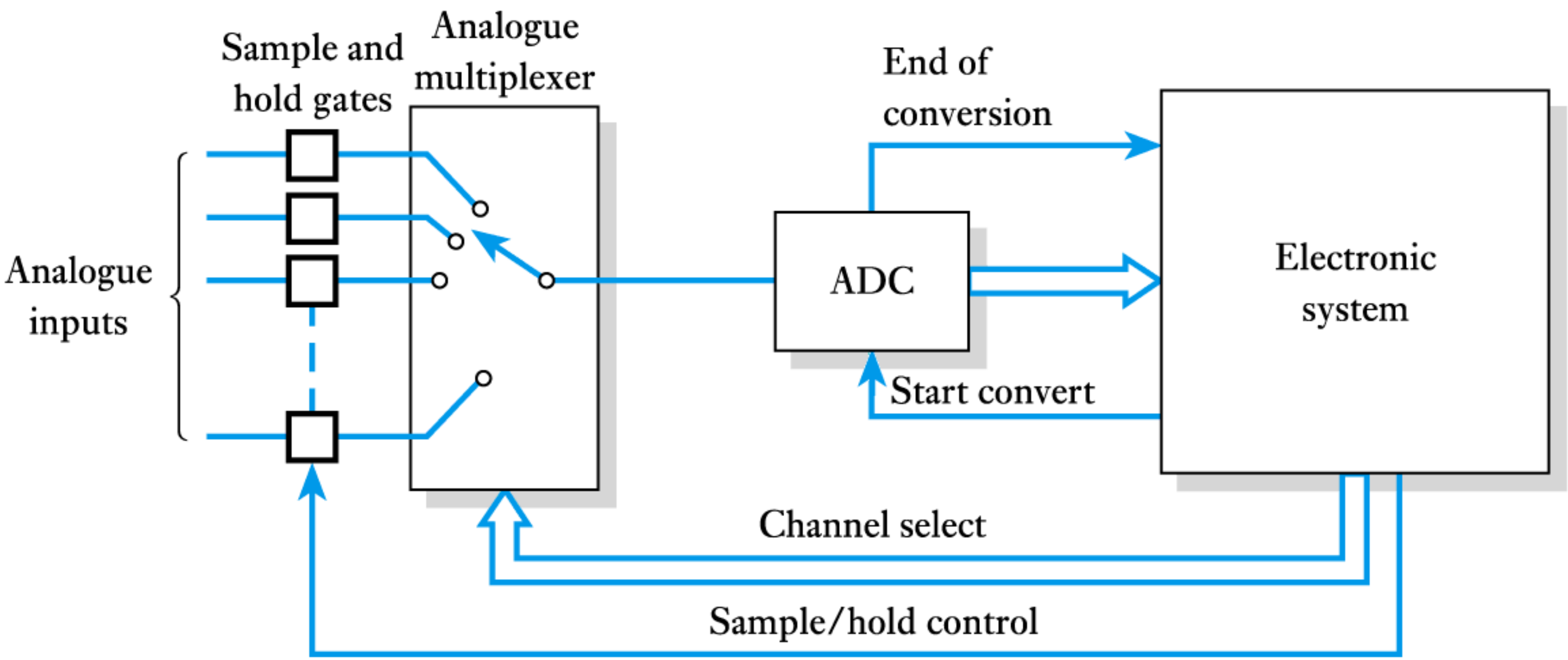
Time delay between sampled inputs

- **Input multiplexing**



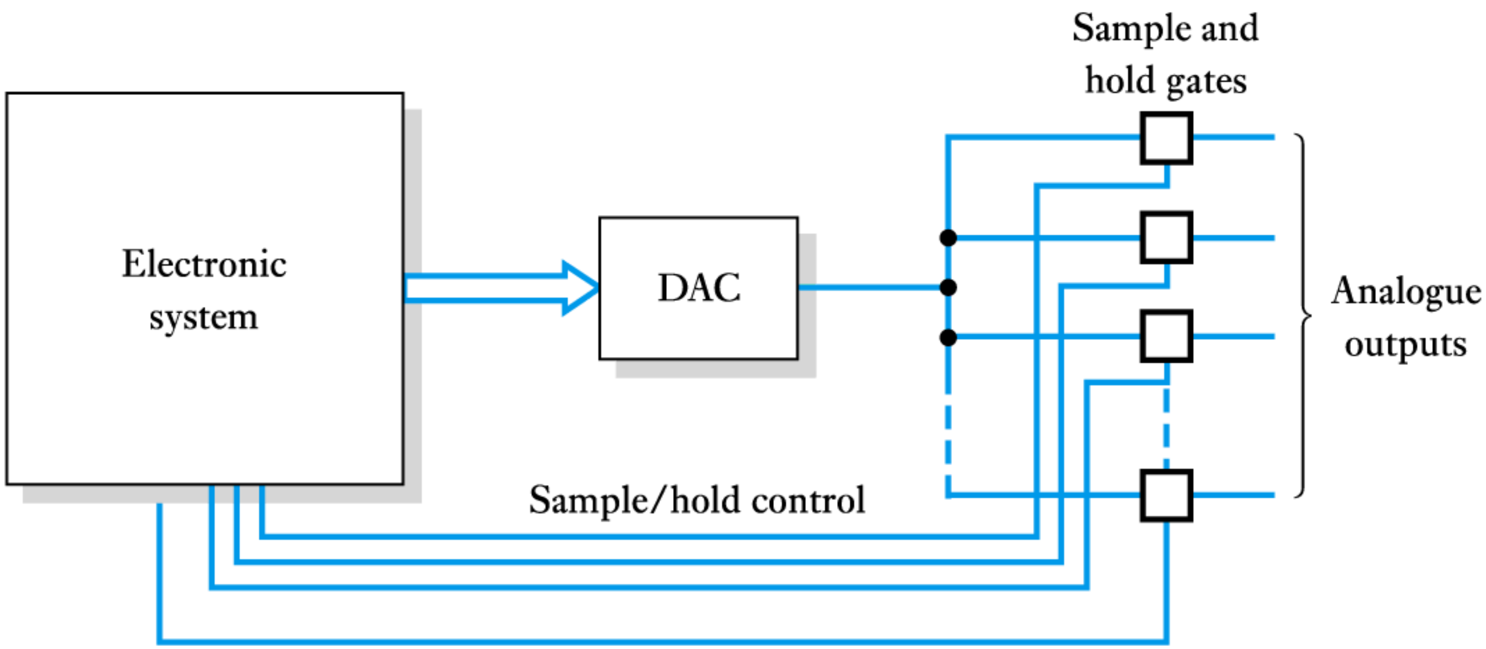
All inputs sampled simultaneously—only droop to worry about (last one read has largest droop)

- **Input multiplexing with sample and hold gates**



Convert input 1-hold output 1
Get new input, convert new input, hold output on ch 2 …

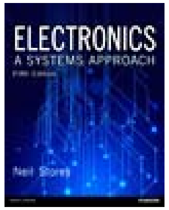
- **Output multiplexing**



- **Single-chip data acquisition systems**
  – these are a combination of an ADC and a multiplexer within a single integrated circuit
  – usually designed for use with microprocessor-based systems, and provide all the control lines necessary for simple interfacing

- More info in the analogue devices analogue to digital conversion handbook on It's Learning analog_digital_conversion.zip

Video 28D



Further Study

Further Study

- The Further Study section at the end of Chapter 28 looks at the use of analogue sensors within smartphones.
- Consider what novel sensors could be added to a smartphone and suggest some applications that might make use of them.
- When you have given this some thought, take a look at the video.

Key points

- Converting an analogue signal to a digital form is achieved by sampling the waveform and then performing analogue to digital conversion
- As long as the sampling rate is above the Nyquist rate (> twice the highest frequency present), no information is lost as a result of sampling
- When sampling broad spectrum signals we make use of anti-aliasing filters to remove unwanted components
- When reconstructing signals, filters are used to remove the effects of the sampling
- A wide range of ADCs and DACs is available
- Sample and hold gates may be useful at the input or output
- Multiplexers can reduce the number of converters required