Telecoms Systems (Week 1&2)

Dr Vincent (Yuanwei) Liu/
Dr Md Hasanuzzaman Sagor/Dr (Cindy) Sun

School of Electronic Engieering & Computer Science,

Queen Mary
University of London

Staff

Dr. Vincent (Yuanwei) Liu yuanwei.liu@qmul.ac.uk

- ✓ Telecom class 1-6
- ✓ E-commerce 11-16

Dr. Cindy Sun – Yan.sun@qmul.ac.uk

Md Hasanuzzaman Sagor m.h.sagor@qmul.ac.uk

- ✓ Telecom class 7-10

Course contents and schedule

The main topics covered by this course are organized as follows:

- Week 1&2: System Introduction, Digitization, Source coding and Channel coding

- Week 3&4: Baseband transmission, Digital modulations and radio propagation

Learning Outcomes

- Understand the foundations of digital communications systems
- Understand the principles of Information Theory and apply them to solve problems in telecommunications engineering.
- Understand the principles of digital transmission, specially baseband transmission, modulation, spread spectrum and multiple access.
- Understand the principles of radio propagation and apply them to solve problems in wireless telecommunications.

About This Course

- Content has not changed significantly
- Exam format – still 4 compulsory questions in 2 hours:
  - Questions from four teaching weeks
  - NO identical questions from past exams
- Important course information:
  - Check information on QMPlus regularly
  - Check your email daily

Assessment

- Exam: 88%
- Class tests: 12%
  - One on in week 2 and one in week 4
  - Each test counts 6%
  - Open book

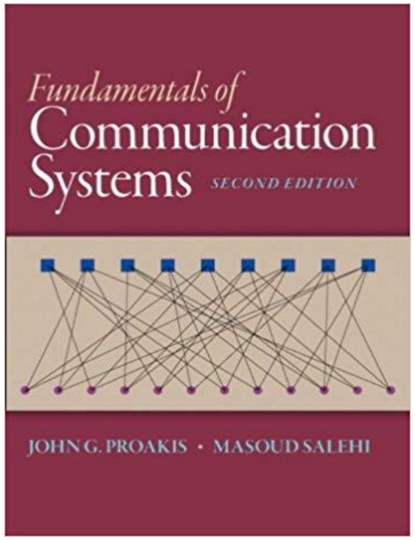
Friday Lecture Arrangement:

- Pattern: Class Split into Two Groups, each group one hour
- Venue: Same Lecture Room
- Contents: Problem Solving Practice

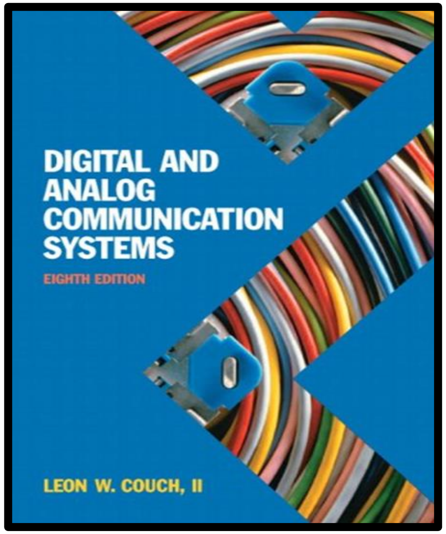
Recommended Text book and references

- There are plenty of books available on this topic.
- Majority of the content is available in 'Proakis and Salehi' book
  - Contents of week 1 & 2 are covered pretty well in this book.
- L Couch's book has better coverage for week 3 & Stallings book has better coverage for week 4.
- If needed, we will provide additional materials.

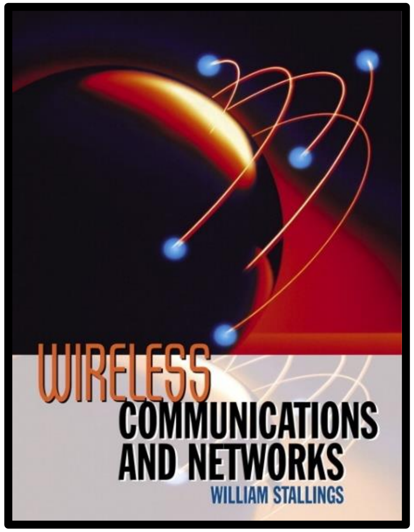
Recommended Text book and References



Week 1 &2



Week 3



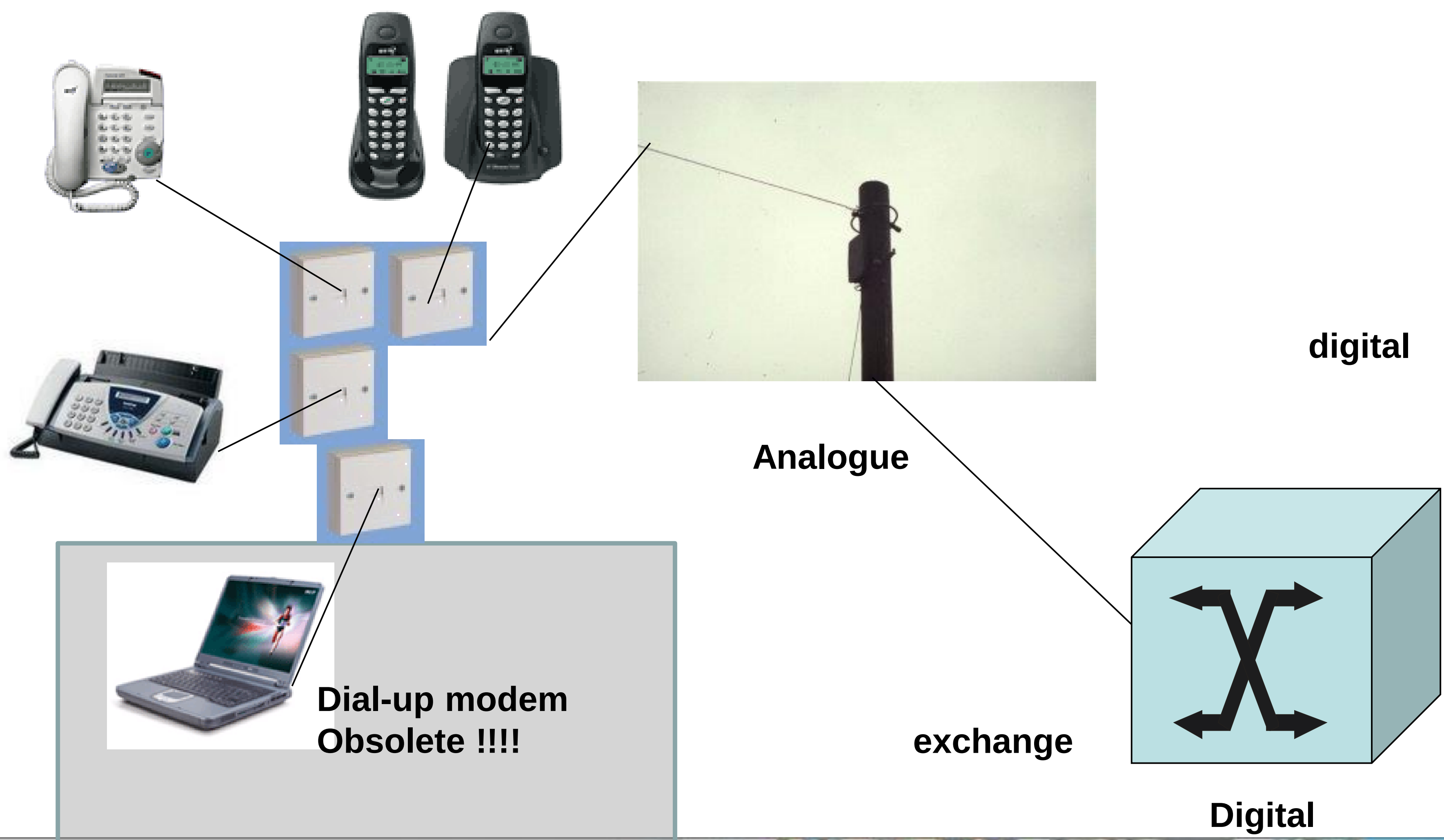
Week 4

MODERN TELECOMMUNICATIONS

"More communications than we know how to use"

- Many different technologies
- Developed in parallel
- Lead time to introduce new services decreasing
- "Reliability" of software decreasing
  - Online patches for mobile phones
- Remote working is now normal

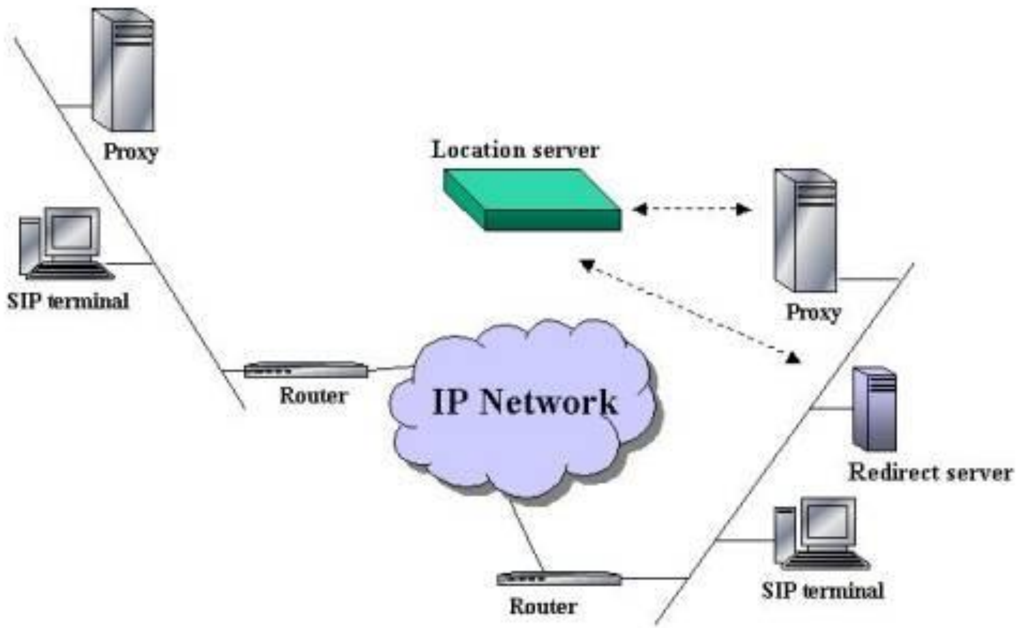
Legacy - wired communications using telephony



Voice is now going IP

SIP based telephony

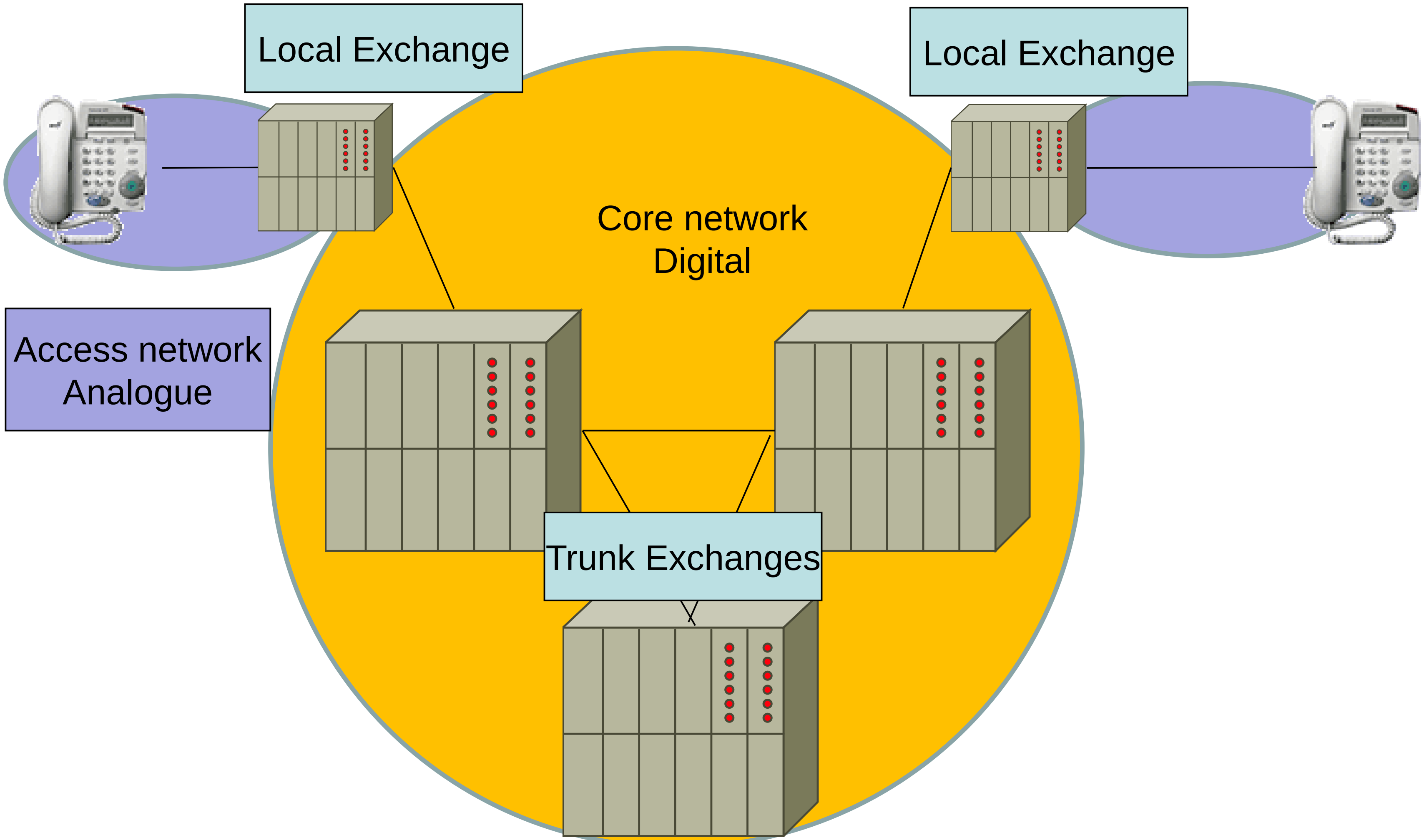
All telephony is going IP



Transport modes

- Traditionally telephony was circuit switched:
  - Call set up, conversation and clear-down phases
  - 64 kbit/s (in digital era) allocated in both directions
  - Much of the capacity wasted
  - Analogue to digital conversion in local exchange
  - Control very much centralised
- Now IP-based
  - Session Initiation Protocol (SIP) sets up and clears down connections
  - Transport RTP (Real-time Transport Protocol)
  - A-D conversion in the telephone
  - More distributed

Traditional network hierarchy

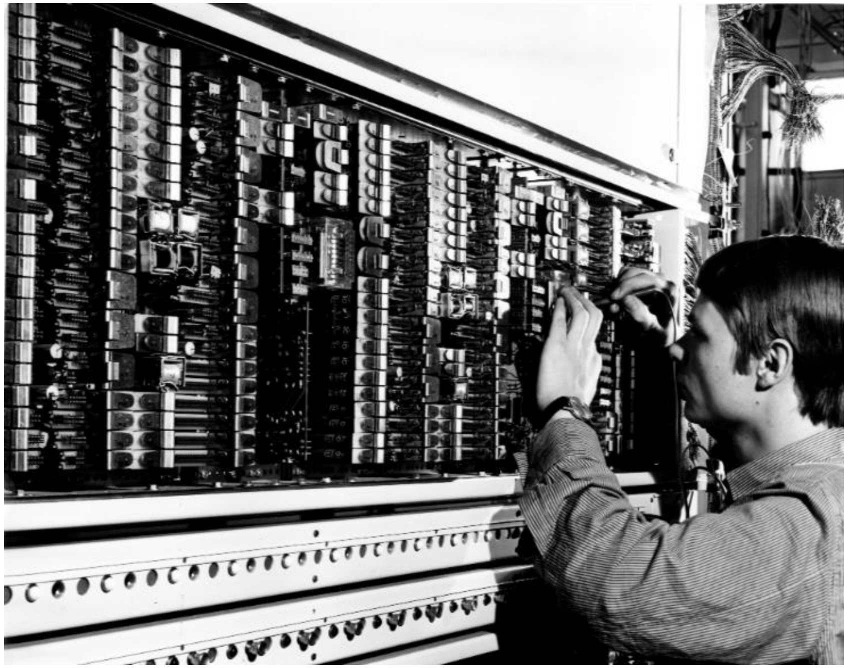


Transmission

- Copper
- Wireless
- Optical
- Increasing optical nearer the customer

Switching used to be manual

Then relays, then electronic – but specialised



Electromechanical exchange
picture courtesy of Nortel

1960s

Private electronic exchange
1983

Now just boxes of electronics – high volume

WLAN AP

IP router

Servers

All of these are just “boxes” of Electronics

IP switch

IP phone

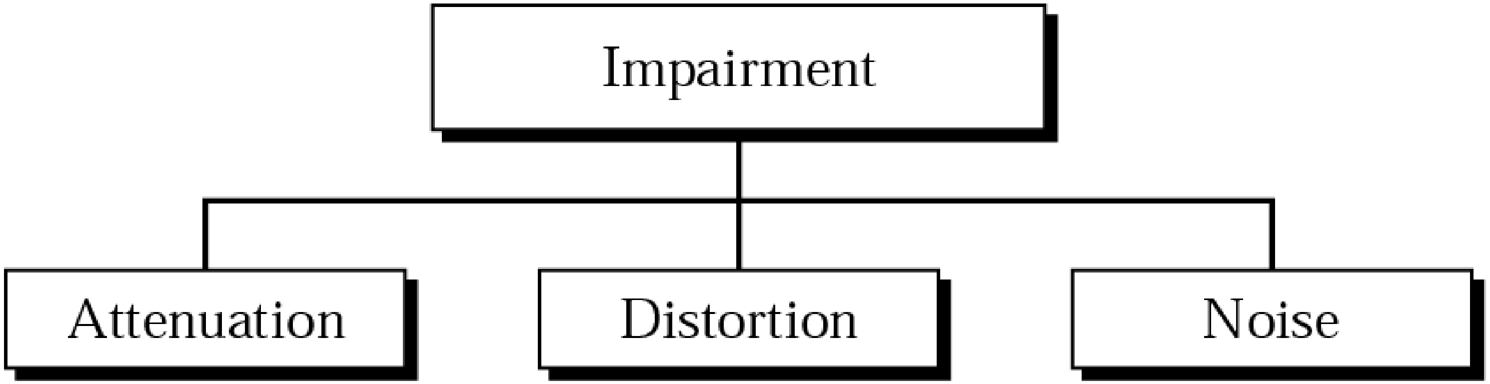
Queen Mary
University of London

Amplification vs Regeneration

When a signal travels through a channel, it suffers attenuation, distortion and noise contamination. Since their negative effects increase with the distance, special equipment called repeaters are inserted along the way.

- In analog systems continuously-varying waveforms are transmitted. In order to preserve the transmitted waveforms, repeaters essentially filter, equalise and amplify the signal.
- In digital systems sequences predefined waveforms (symbols) are transmitted. In this case, repeaters regenerate such waveforms.

Transmission impairment



- Signals that travel through transmission media will always be *corrupted* by attenuation, distortion and noise.

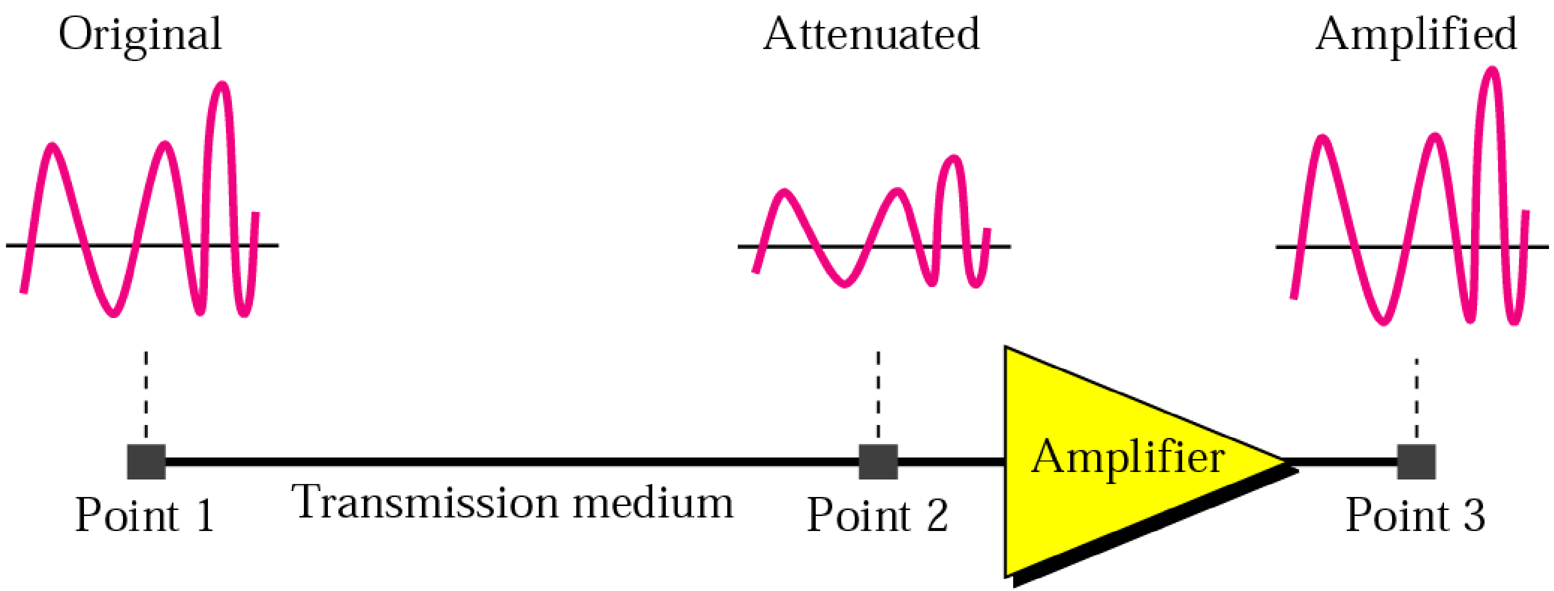
Attenuation

- Attenuation means the loss of energy.

- When signals travel through a medium they lose some energy so that they can **overcome** the ***resistance*** of this medium.

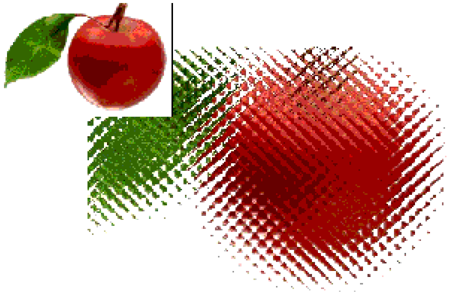
Attenuation

- To compensate for energy loss, **amplifiers** are used to boost the signal back up to its original level.



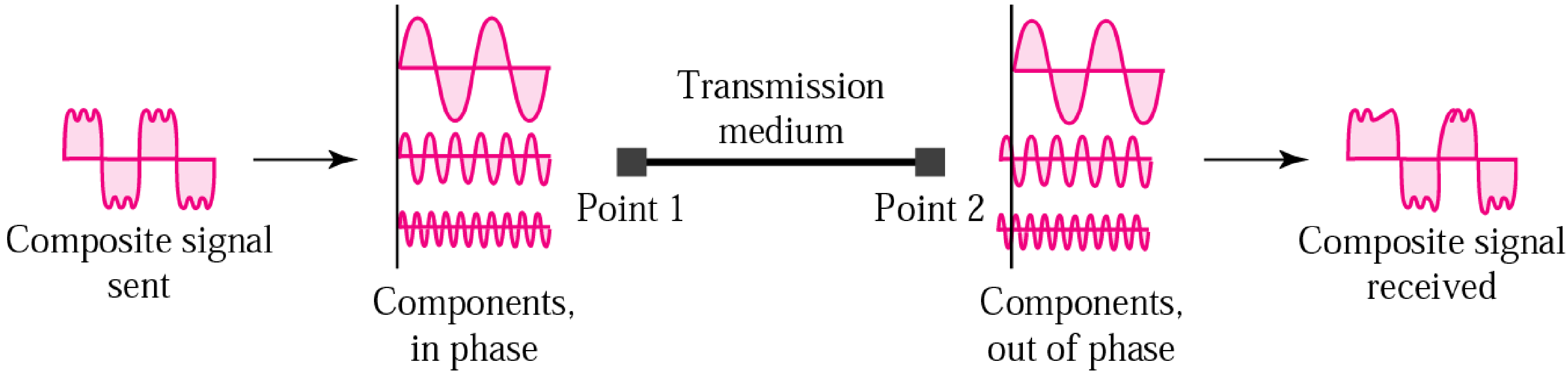
Distortion

- **Distortion**: signal *changes* in its **form** or **shape**.
- Typically effects complex or composite signals.
- Distortion takes place when a composite signal carrying different frequencies suffers from the delay of some of these frequencies.



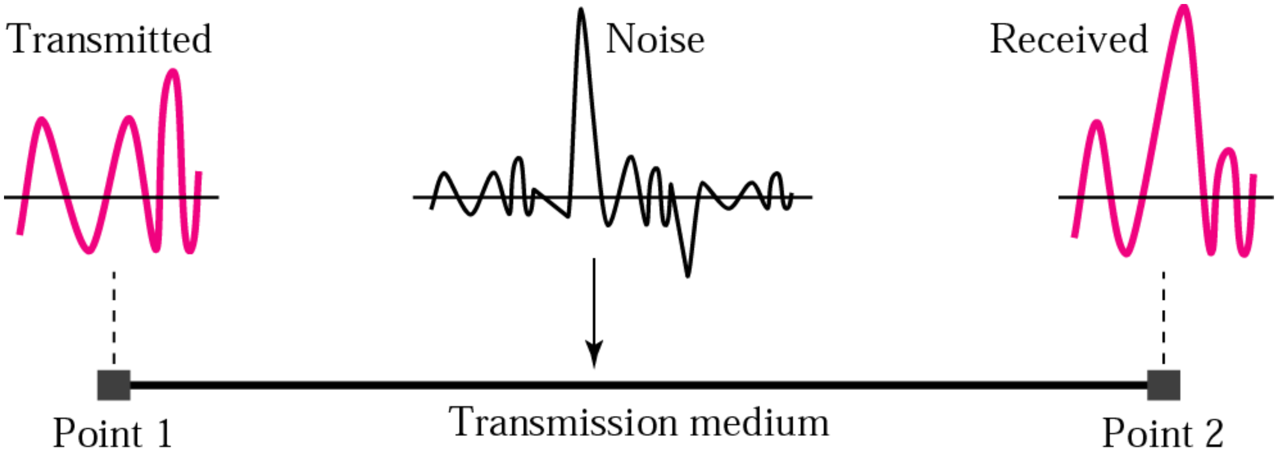
Distortion

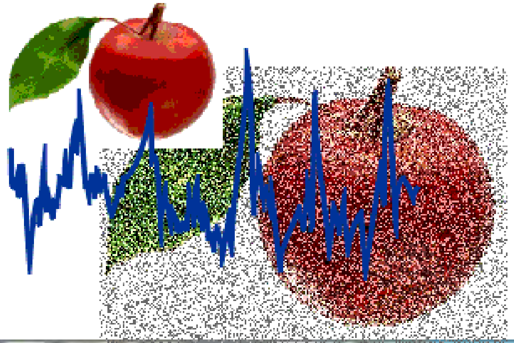
- Each frequency component has its own propagation **attenuation** through a medium.



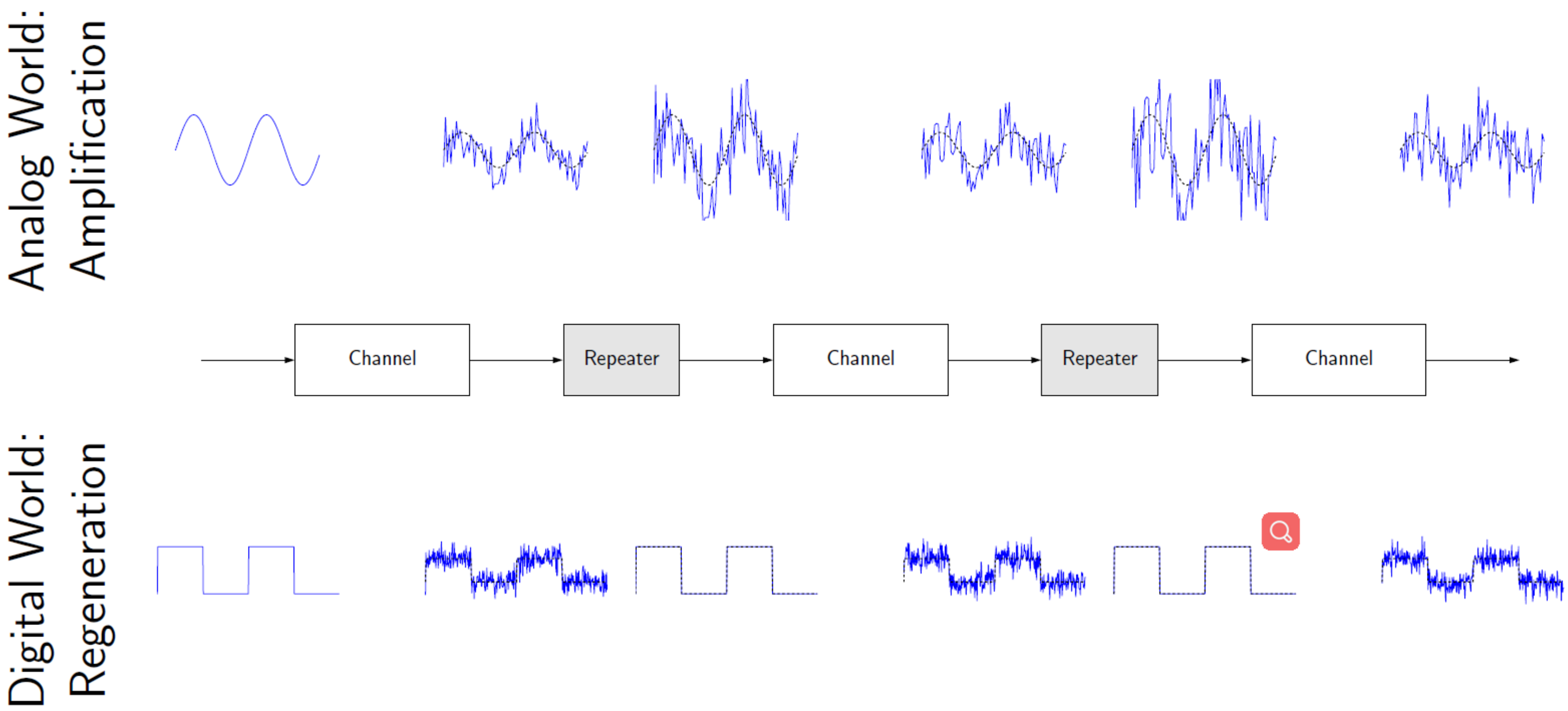
Noise

- Noise is the **main source** of a signal being <u>corrupted</u>.





Amplification vs Regeneration



The Public Switched Telephone Network

The PSTN has a dual analog-digital nature:

- Local loops are mainly analog (they are legacy increasingly structures), although digital (FTTH, Fiber To The Home).
- The core network (switching offices and trunks) is however digital. Local exchanges connected to the local loops digitize speech signals.

(By contrast, due to the lack of legacy structures, mobile phone systems are entirely digital)

So, why digital?

Efficiency: Circuit switching vs Packet switching

In communication networks such as the PSTN information sources and destinations are usually connected through intermediate systems such as switches and routers. Circuit switching and packet switching are two strategies to send the information from the source to the destination through them.

- ❖ In circuit switching a dedicated path connecting source and destination is established. Resources are guaranteed for the whole connection time but they might be wasted.
- ❖ In packet switching no resources are reserved and the information to be sent is split into packets that can follow different paths to reach destination. Since resources are not reserved, they can be shared.

Hence, packet switching makes a more efficient use of the existing resources.

Resilience: Circuit switching vs Packet switching

Resilience means that the system can recover quickly when parts of it fail. Originally, packet switching was proposed as an alternative to circuit switching in response to the need of building resilient communication networks.

- Traditionally, circuits were established in hierarchical networks by finding a common switching office. This means that the network could be split into different isolated networks if a few switching offices failed.
- By contrast, in mesh networks many different paths can be established. Combining mesh networks topologies with packet switching results in fault tolerant systems, that is, systems that are resilient.

Incidentally, digital data are more amenable to be implemented in packet switching than analog data.

(more in later weeks)

WIRELESS COMMUNICATIONS

History of wireless communications

- 1865 James Clerk Maxwell published his equations
- 1887 Heinrich Hertz demonstrated EM wave propagation
- 1893 Nicola Tesla demonstrated communication by radio
- 1895 Aleksandr Popov demonstrated a wireless system
- 1896 Guglielmo Marconi demonstrated wireless telegraphy
- 1901 First wireless signal sent across the Atlantic Ocean from Cornwall to St. John’s, Newfoundland (Canada)
- Marconi was not the ‘inventor’, but appreciated the commercial opportunities offered by the new medium.

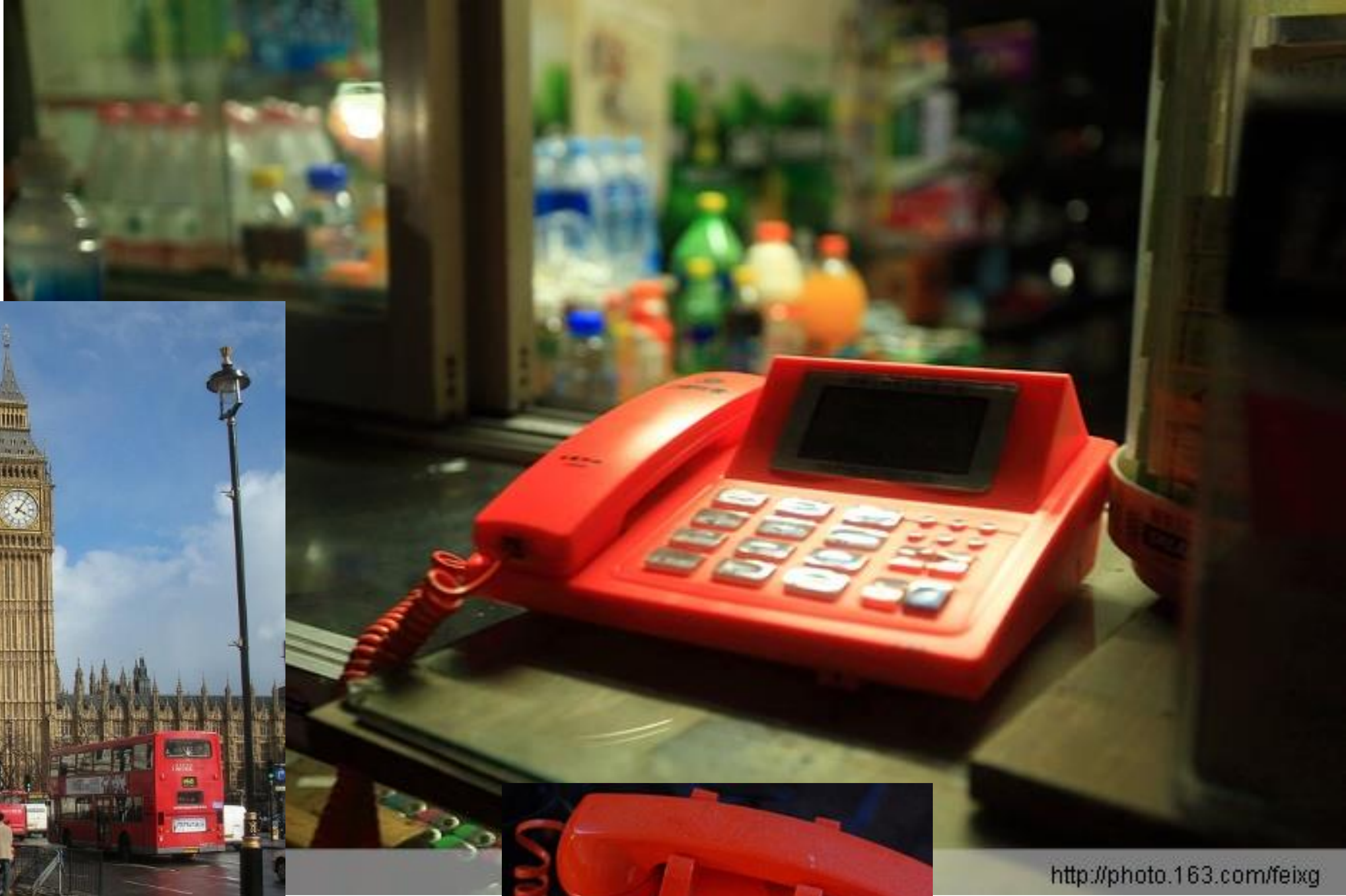
Types of wireless network

- WPAN (Wireless Personal Area Network)
  - typically operates within about 30 feet
- WLAN (Wireless Local Area Network)
  - operates within 300 yards
- WMAN (Wireless Metropolitan Area Network)
  - operates within tens of miles
- WWAN (Wireless Wide Area Network)
  - operates over a large geographical area, mobile phone, …

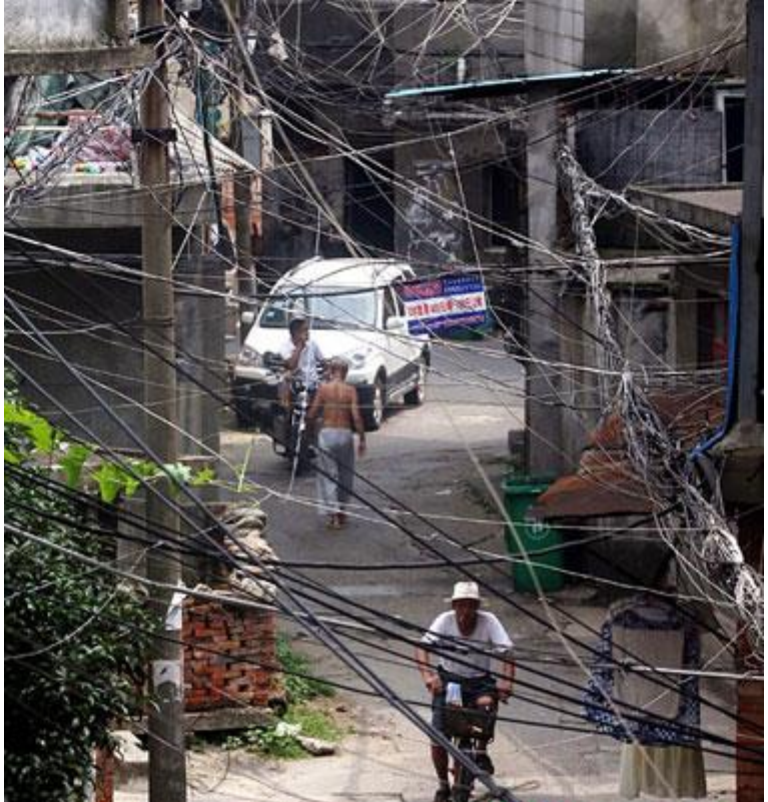
Why wireless?

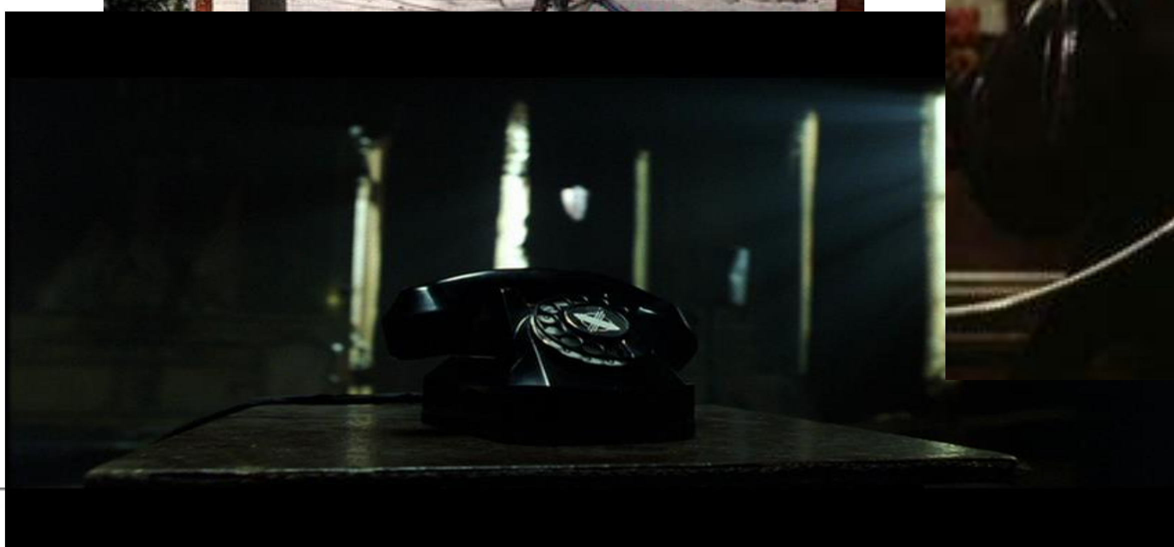
- No more cables
  - No cost for installing wires or rewiring
  - Wiring is infeasible or costly in some areas, e.g.. rural areas, old buildings…
- Mobility and convenience
  - Allows users to access services while moving: walking, in vehicles…
- Flexibility
  - Roaming allows connection any where and any time
- Scalability
  - Easier to expand network coverage compared to wired networks.

Touch-Tone dialling



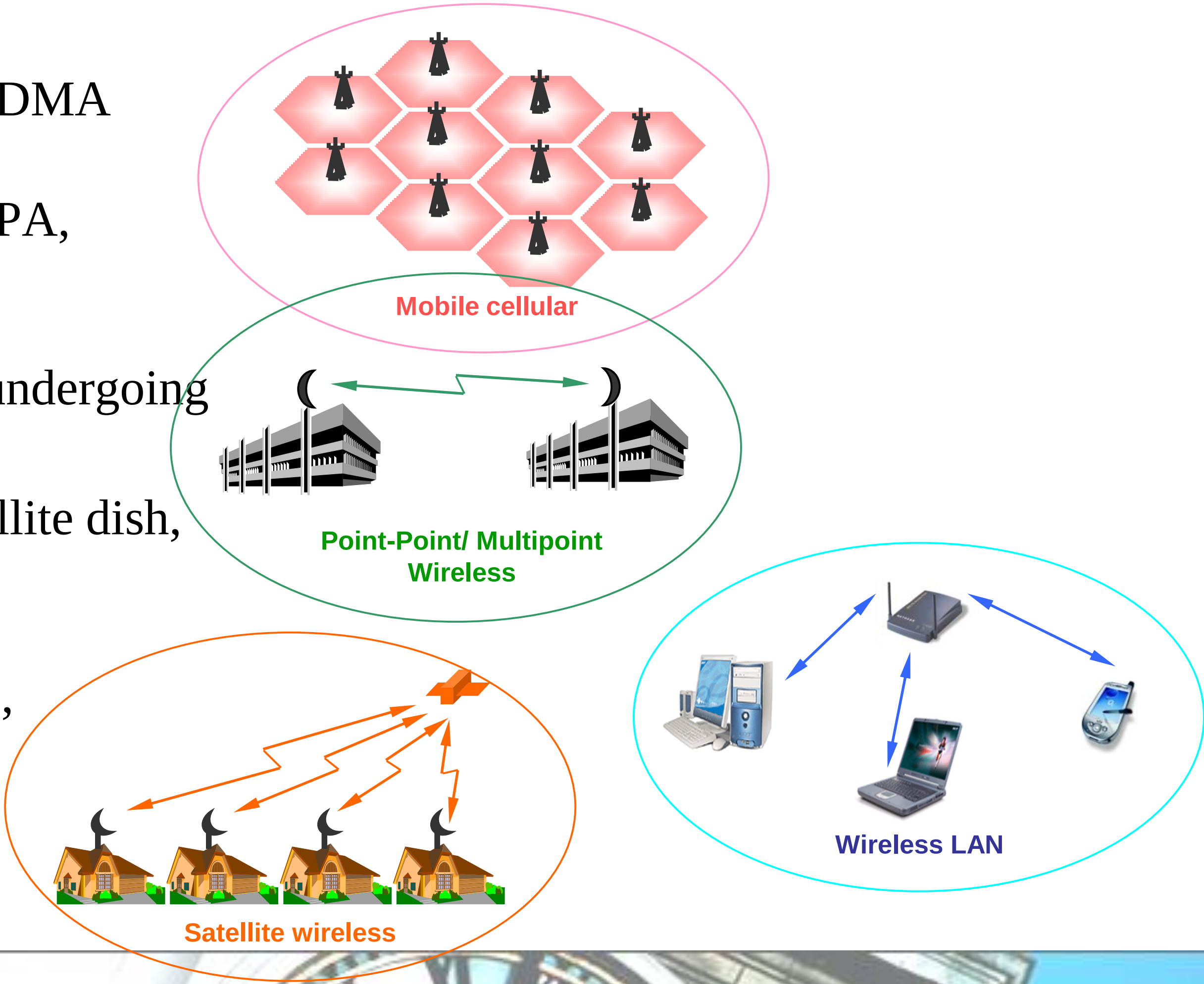
Pulse Phone



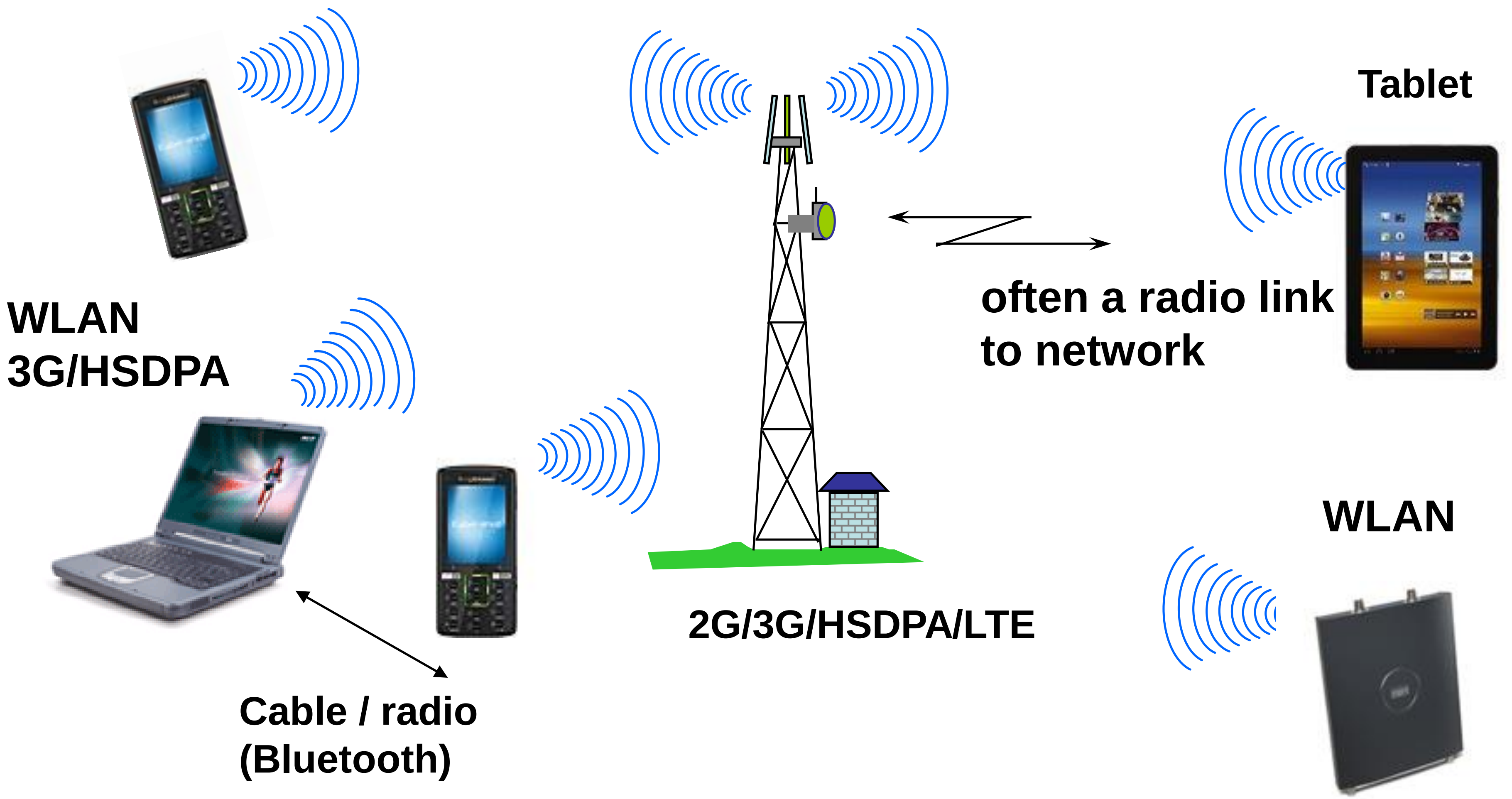


Emerging and existing wireless technology

- Mobile Wireless:
  - 1G: Analog
  - 2G: GSM, TDMA, CDMA
  - 2.5G: EDGE, GPRS
  - 3G: W-CDMA, HSDPA, HSUPA
  - 4G: LTE
  - 5G: Standardization undergoing
- Fixed Wireless:
  - MMDS, LMDS, Satellite dish, Microwave
- Wireless LAN:
  - IEEE 802.11, Ad-hoc, Bluetooth,
- WiMax

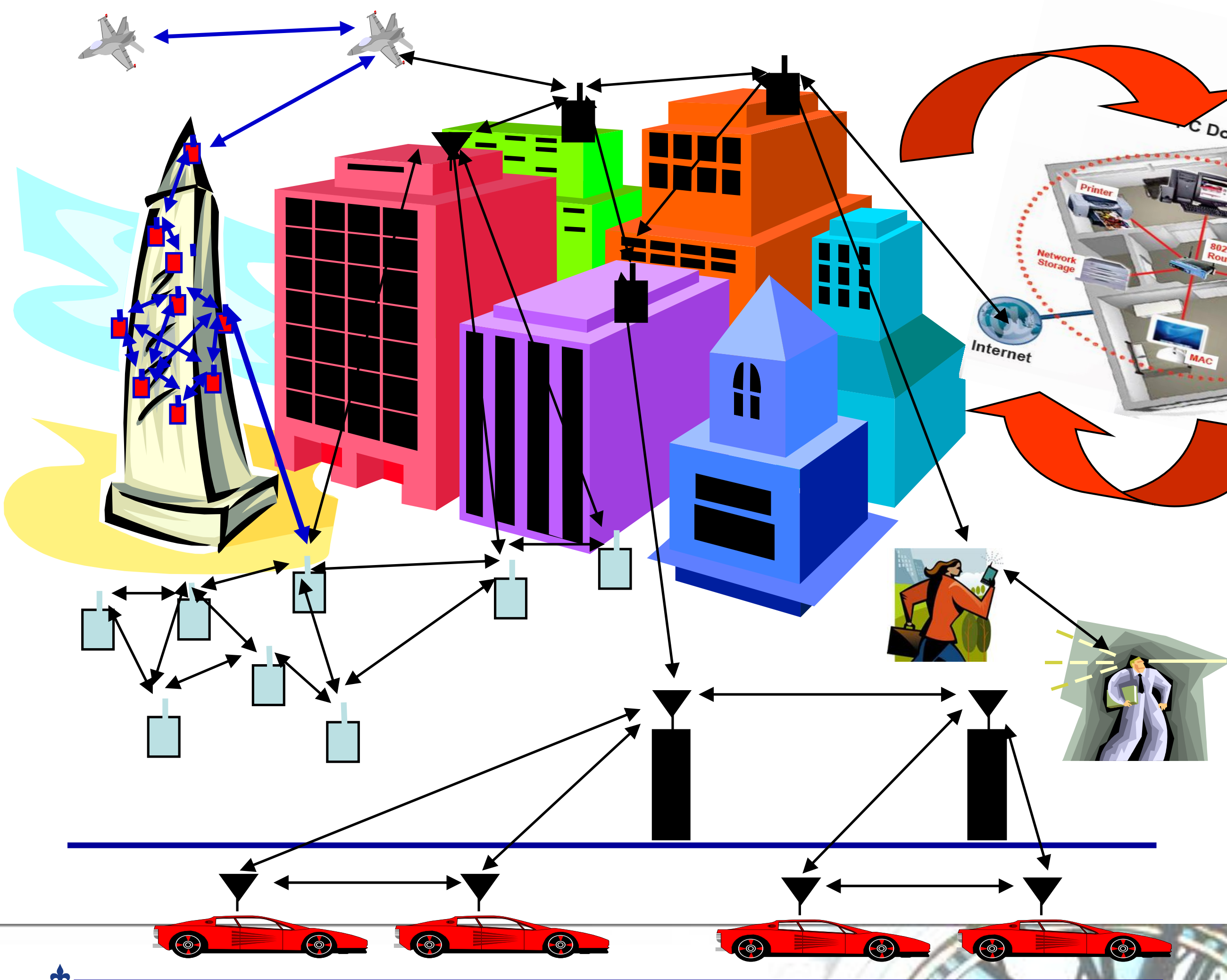


Mobile communications



Future Wireless Networks

Ubiquitous Communication Among People and Devices



Next-generation Cellular
Wireless Internet Access
Wireless Multimedia
Sensor Networks
Smart Homes/Spaces
Automated Highways
In-Body Networks
All this and more …

Roadmap for Wireless Standardization

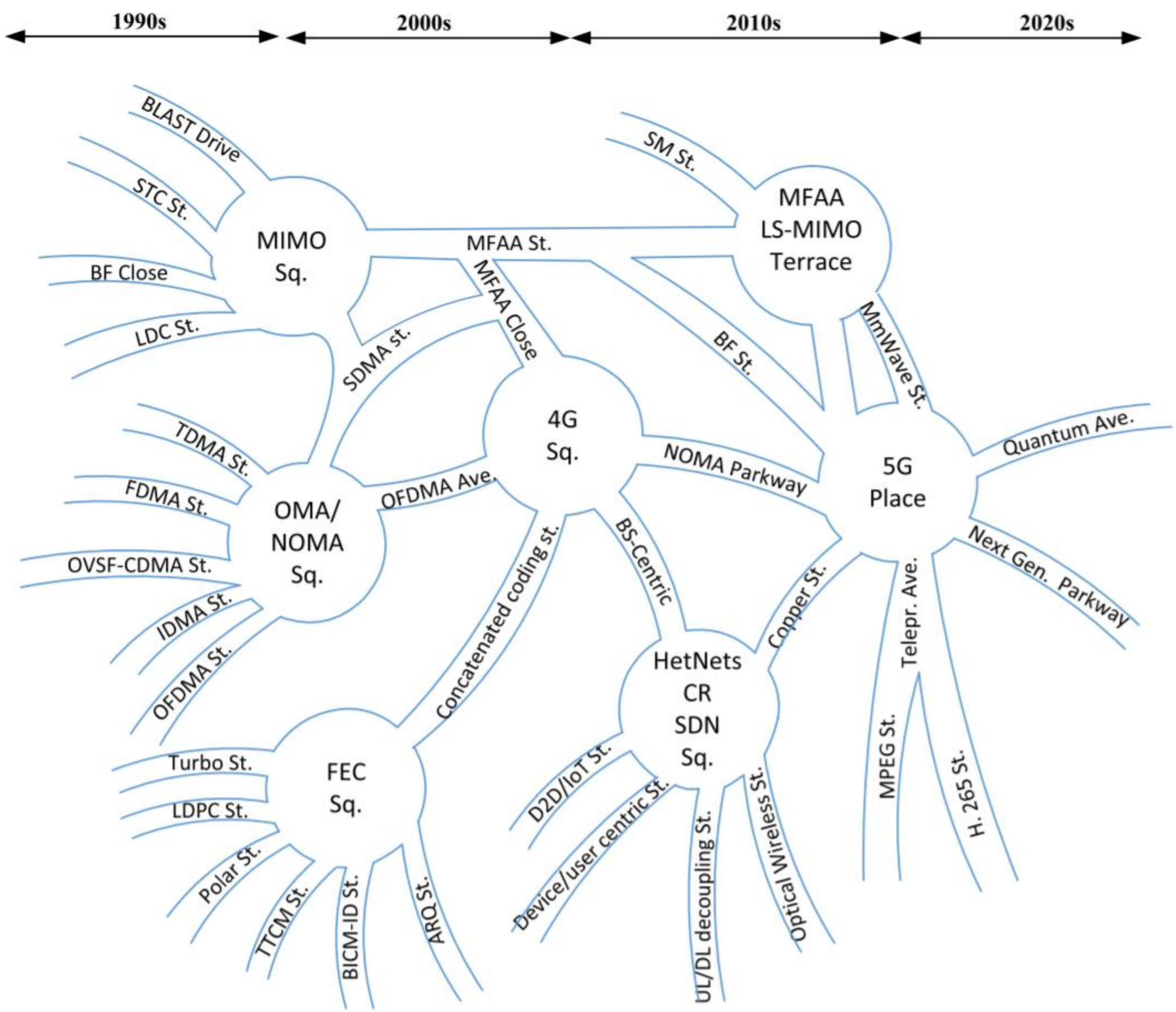
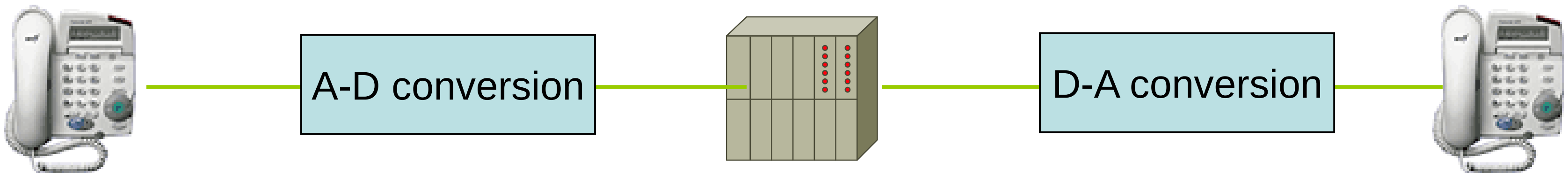


Fig. 1. ***The roadmap for illustrating the brief history of wireless standardization.***

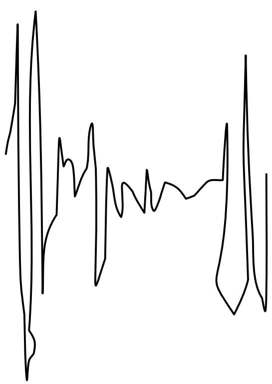
INFORMATION CONVERSION

Transmission of information



Information:
'Hello! How are you?'

You and I understand but not the telephone!



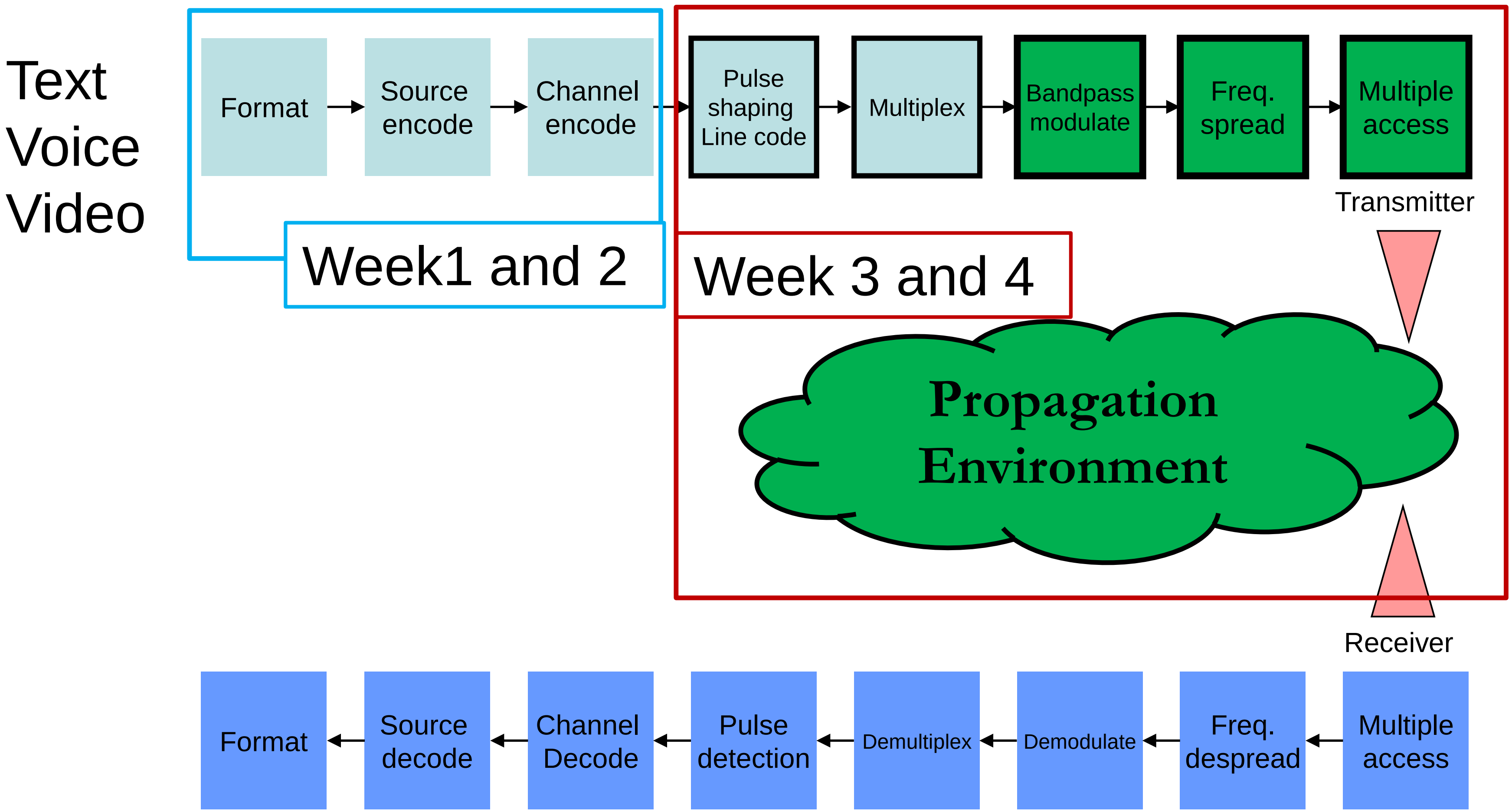
Analogue signal can be understood by electrical systems but problematic!

So all new systems digital

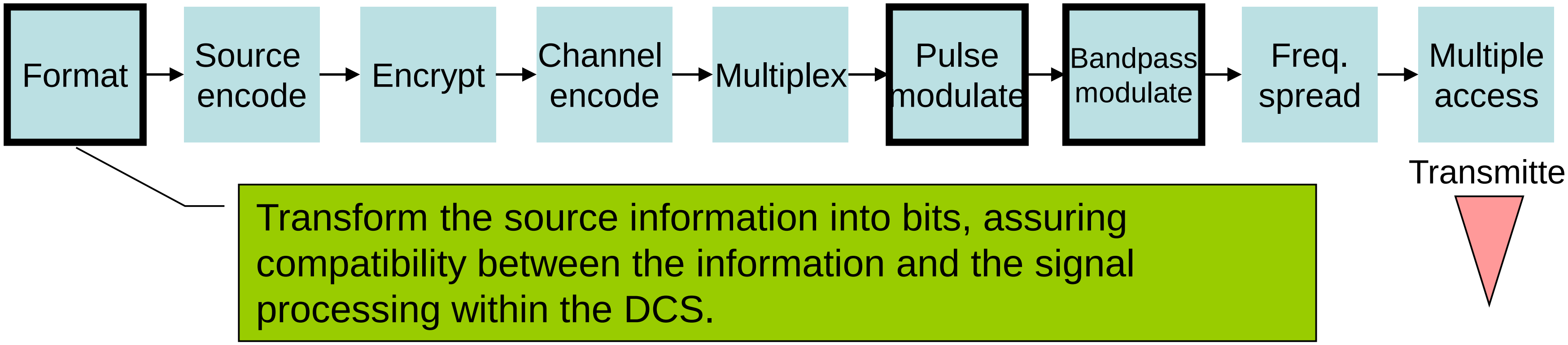
Information Conversion

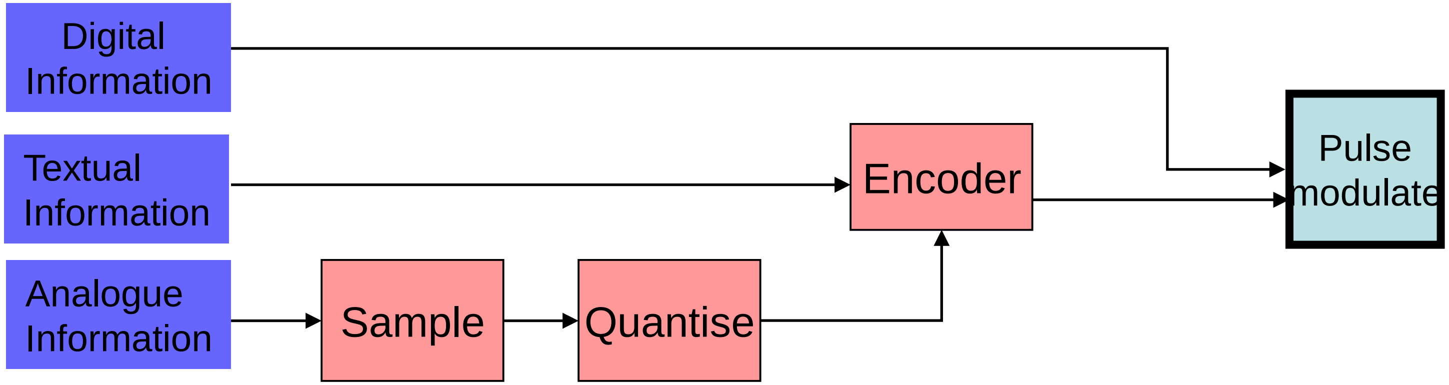
- Different sources of information need different methods to transform the source information to a digital format
  - Text – ASCII (used to be others)
  - Voice (PSTN) – Pulse Code Modulation (G711a/u) 64kps
  - Voice (GSM) – GSM codec (13kbps) EFR (improved quality)
  - 3G WCDMA – AMR (adaptive Multi Rate)
  - Picture – JPEG …..
  - Video – MPEG2, MPEG4, H264
- Aim is to minimise bitrate but maintain quality

Overview of Wireless Communication System



Transmission side





Formatting Textual Data

- Textual information compromises a sequence of alphanumeric characters.
- Each alphanumeric character is transformed into binary by character coding. Most popular character coding method is ASCII
- Encoded into sequence of k bits called [symbols]

SAMPLING

Introduction to A/D conversion

Analog information has the following properties:

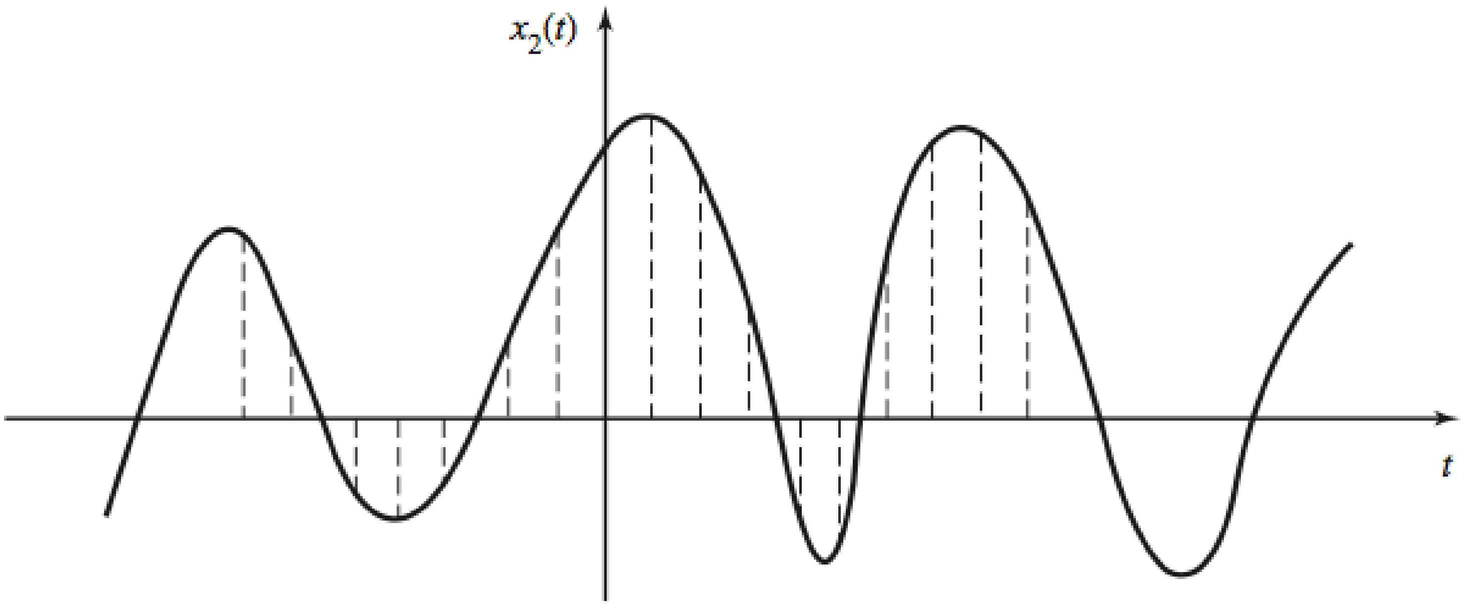
- Continuous in time.
- Continuous in amplitude.

By contrast, digital information consists of sequences of discrete values. In other words, digital information is:

- Discrete in time.
- Discrete in amplitude.

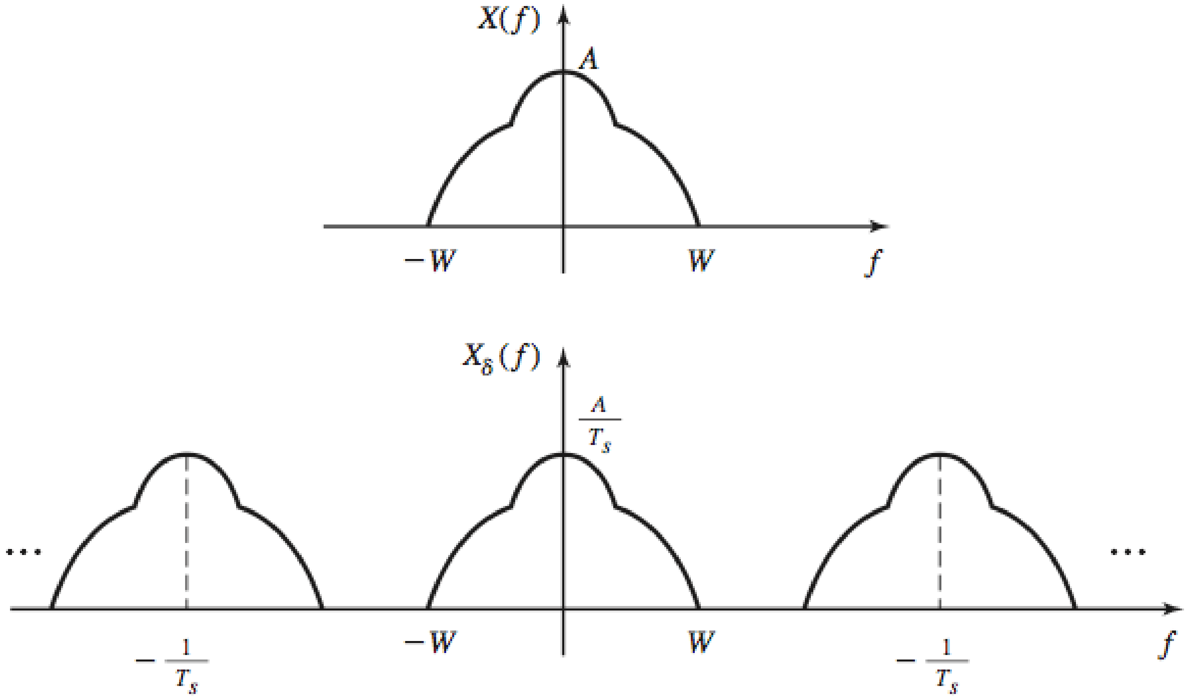
The process of converting analog information into digital form is called **digitisation**. Hence, digitisation consists of discretising analog information both in time (**sampling**) and in amplitude (**quantisation**).

Sampling in the Time Domain



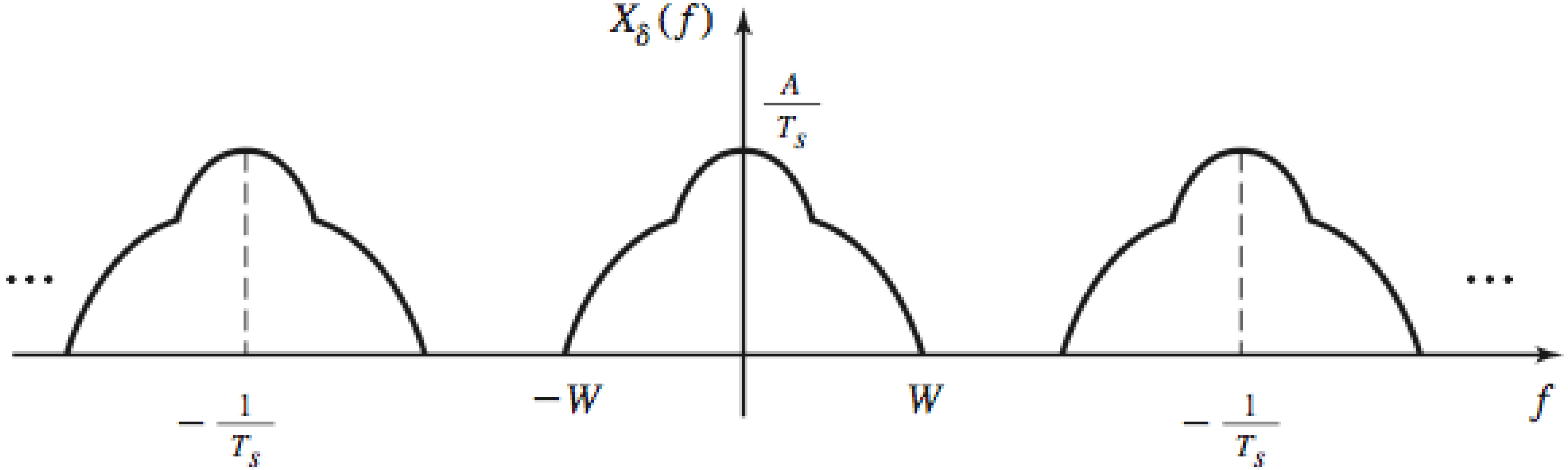
The process of sampling a signal x(t) can be mathematically modelled as the result of multiplying it by a train of impulses. The quantity T_s is known as the sampling period and $f_s = 1/T_s$ is the sampling frequency.

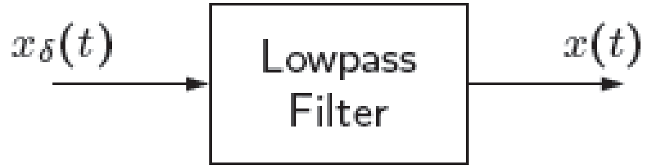
Sampling in the Frequency Domain



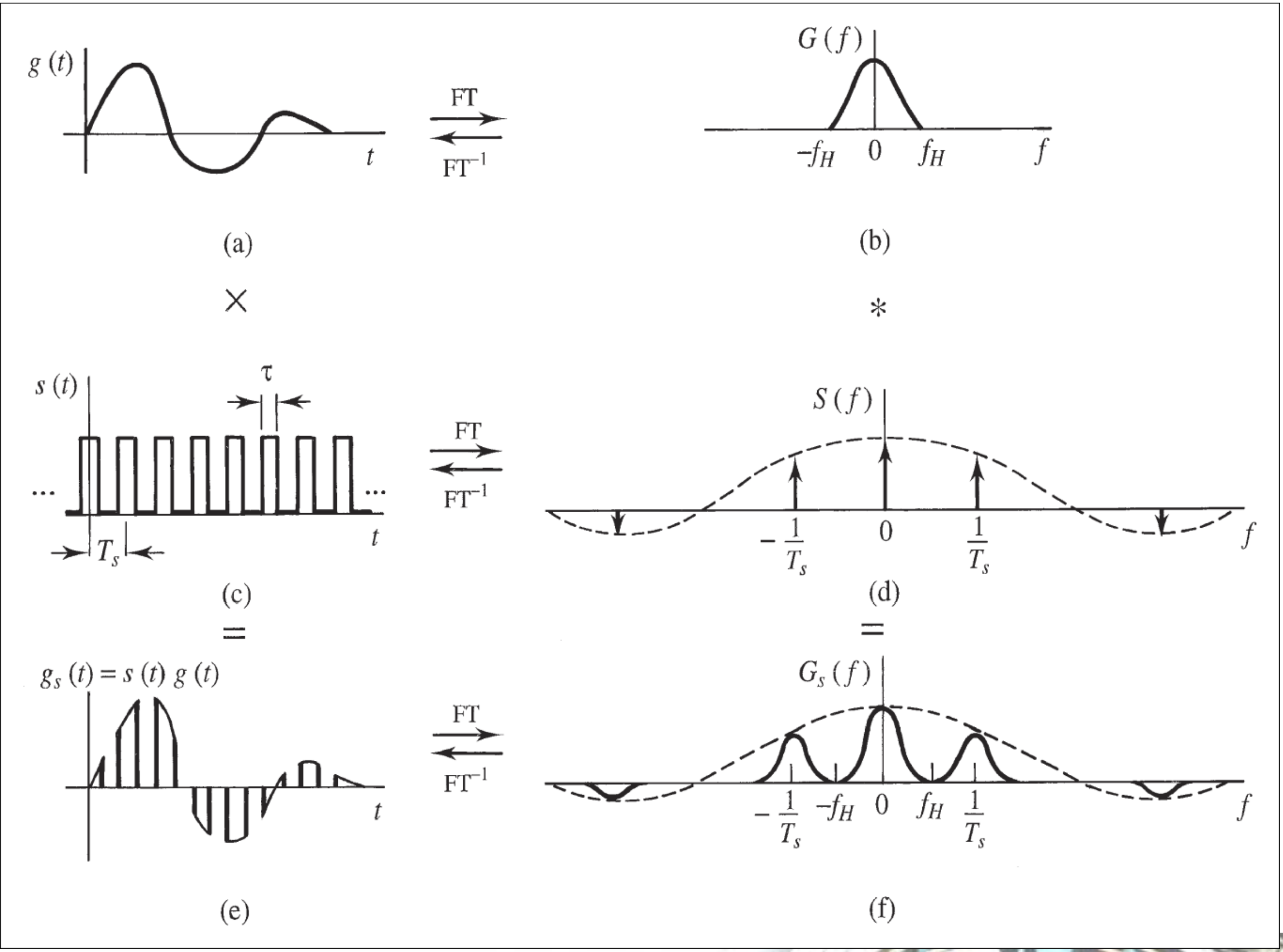
By using the convolution property, we can prove that the spectrum of a sampled signal consists of replicas of the original spectrum centred at multiples of the sampling frequency.

Interpolating for D/A conversion





Natural Sampling (Periodic Pulse Train)



(a) signal g(t);

(b) signal spectrum;

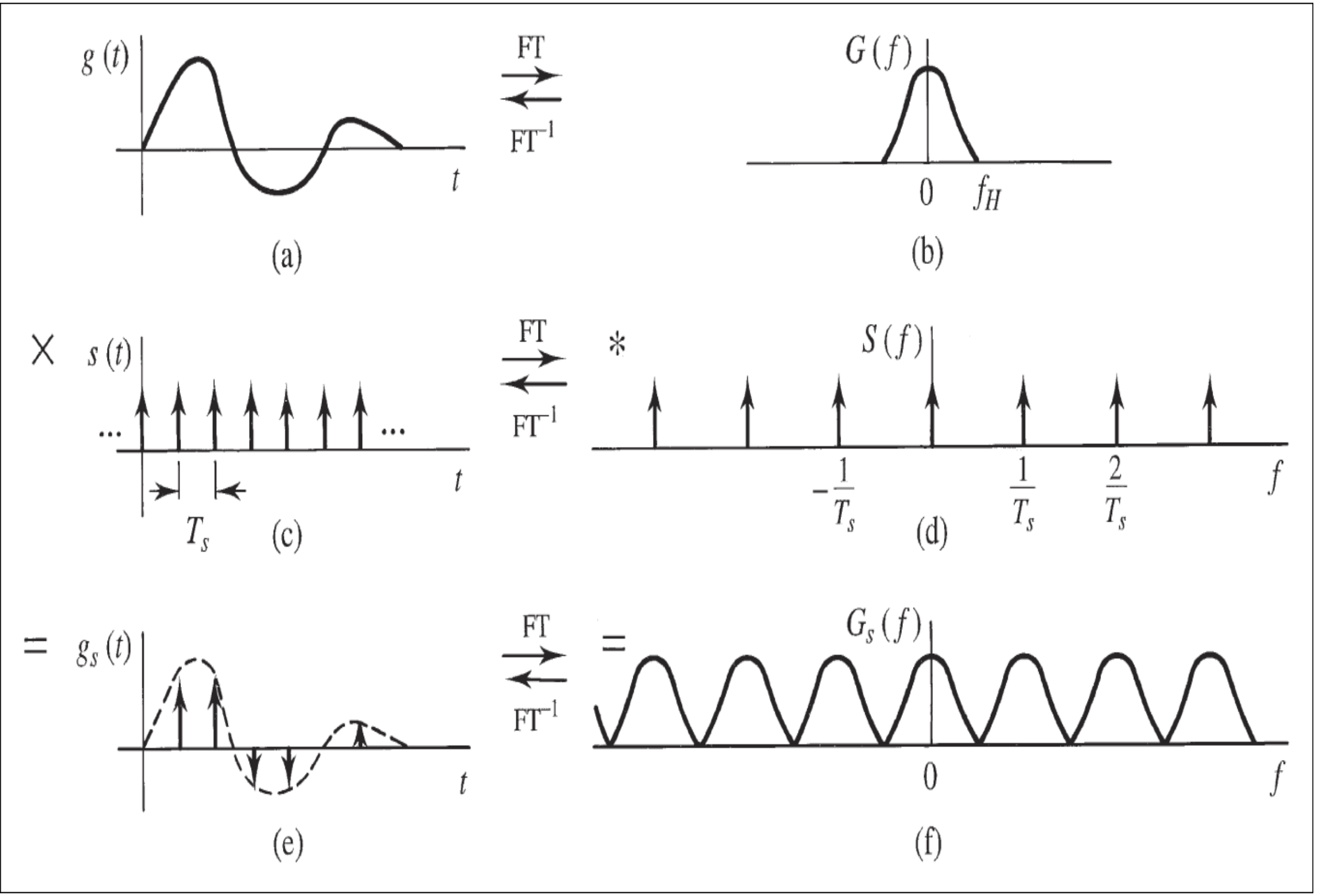
(c) sampling function;

(d) spectrum of sampling function;

(e) sampled signal;

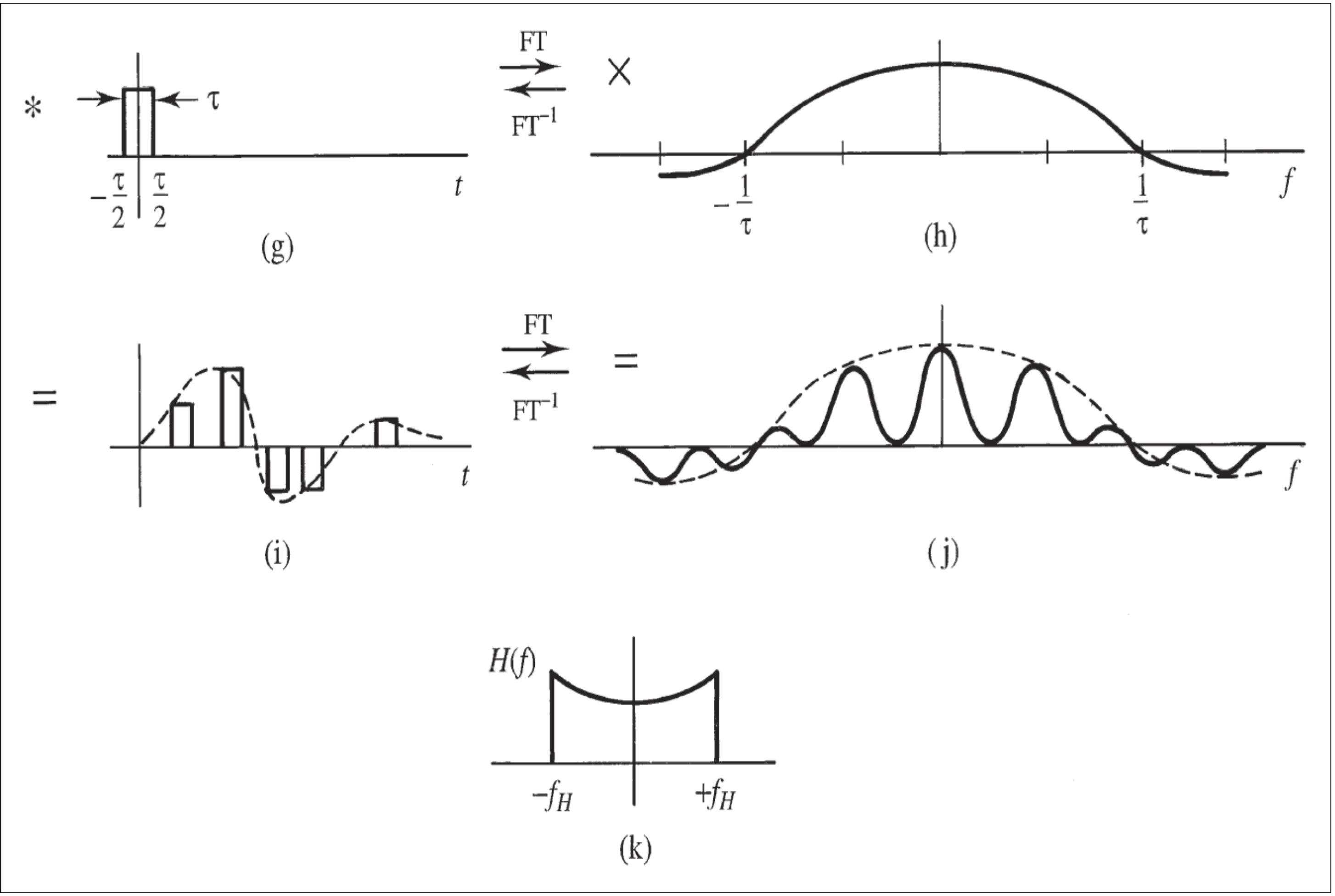
(f) spectrum of sampled signal

Ideal Sampling (Periodic Impulse Train)



a) signal;
b) signal spectrum;
c) sampling function;
d) spectrum of (c);
e) sampled signal;
f) spectrum of (e);

Impulse (Ideal) Sampling -- Equalization



(g) finite width sample;

(h) spectrum of (g);

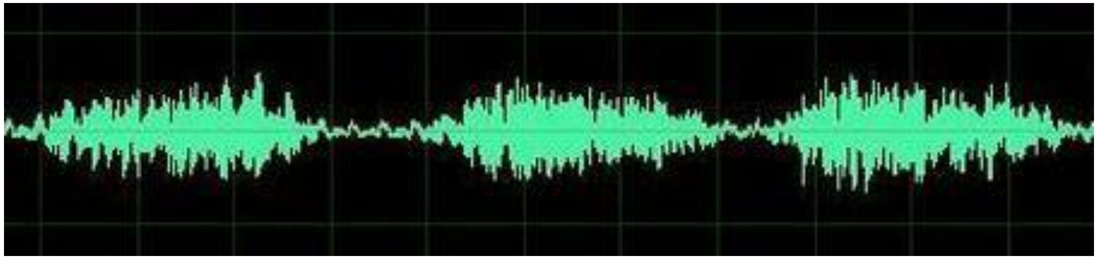
(i) sampled signal;

(j) spectrum of (i);

(k) receiver equalizing filter to recover g(t)

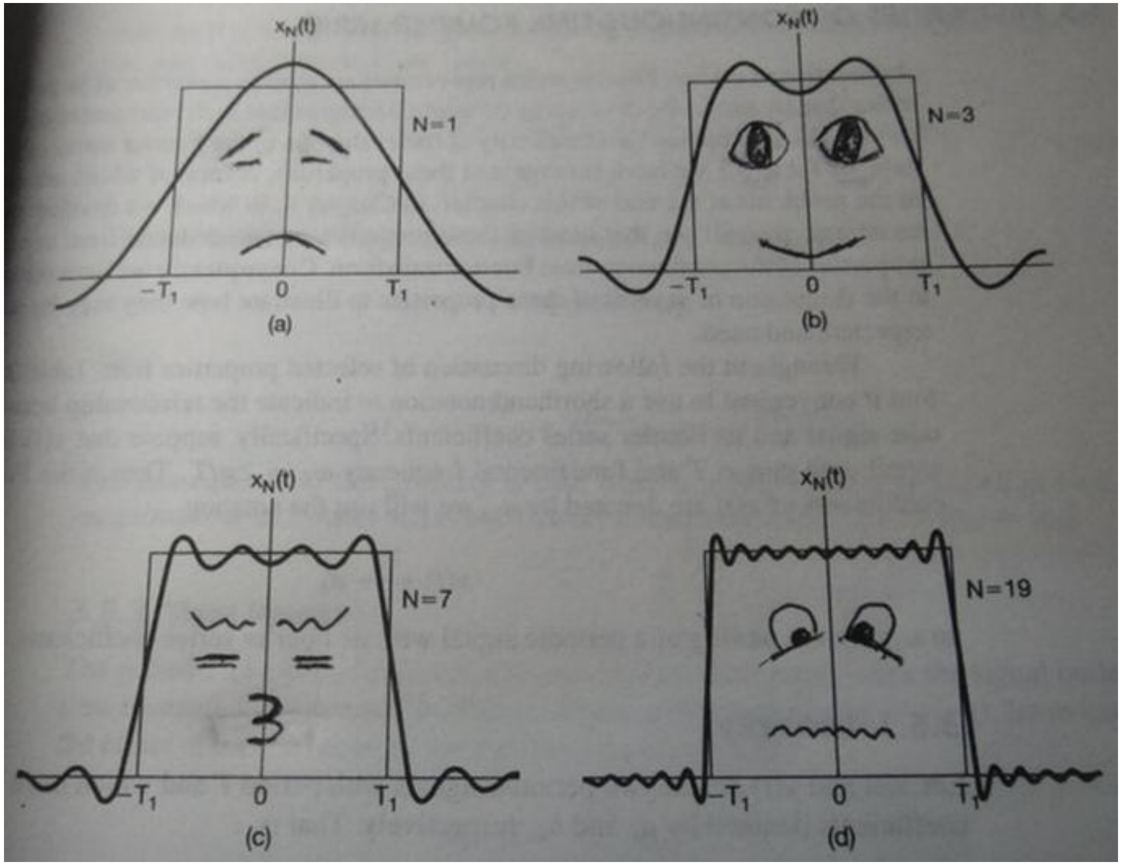
Time Domain VS Frequency Domain

Time Domain



Frequency Domain

Fourier Series of Square Wave



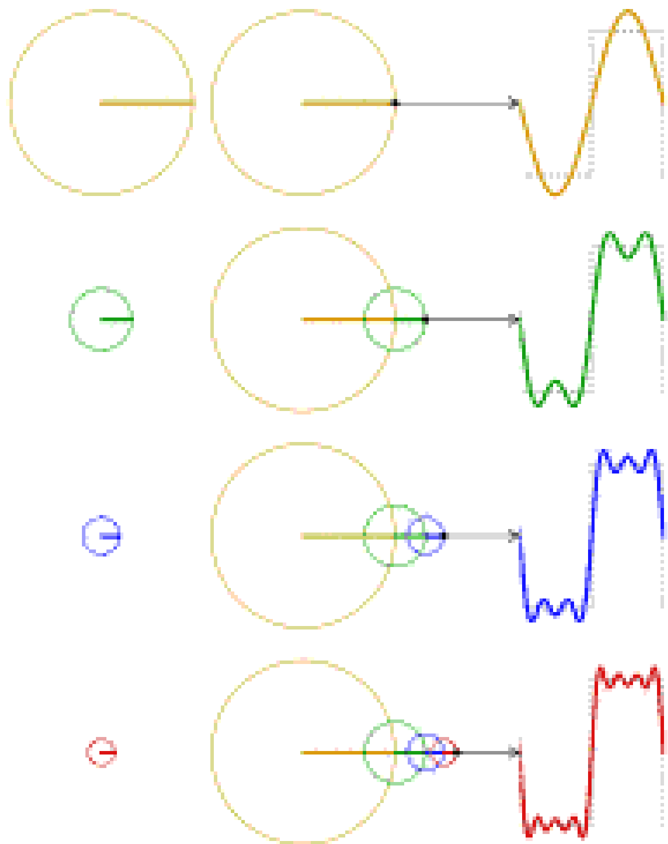
Fourier Series of Square Wave : Dynamic

$$\frac{4\sin\theta}{\pi}$$

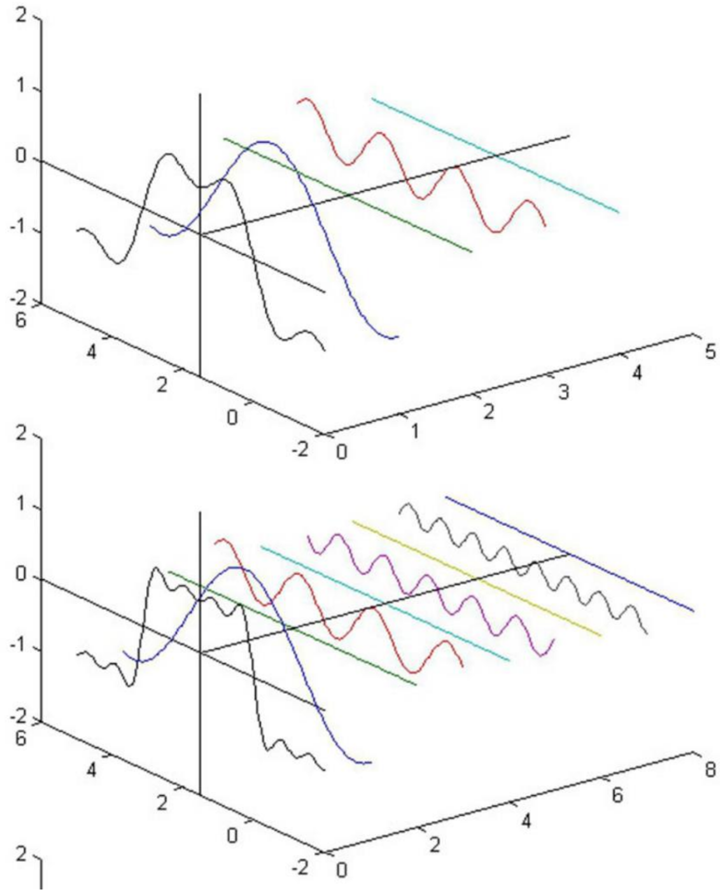
$$\frac{4\sin 3\theta}{3\pi}$$

$$\frac{4\sin 5\theta}{5\pi}$$

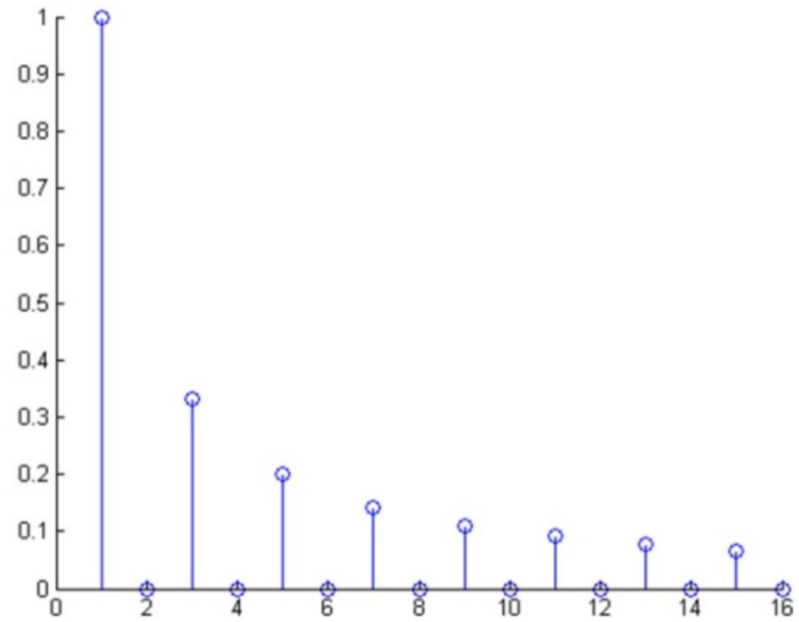
$$\frac{4\sin 7\theta}{7\pi}$$



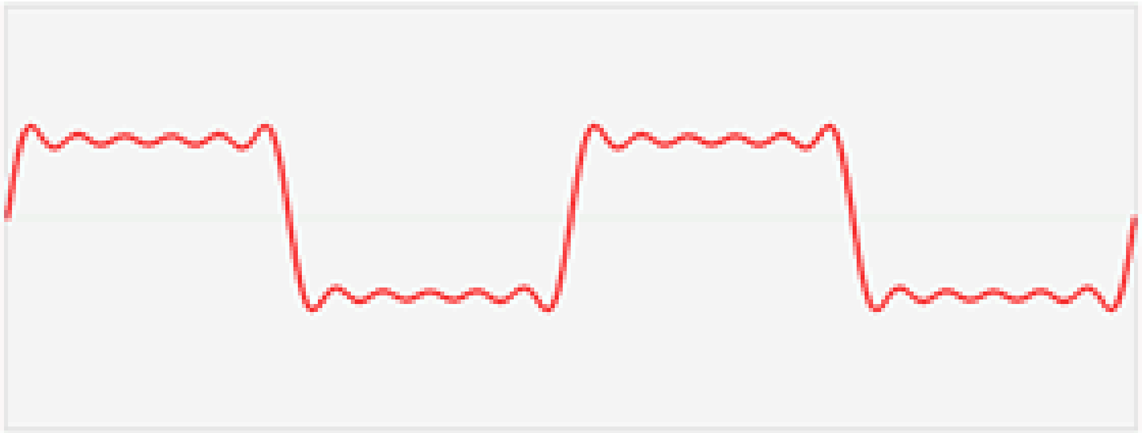
Time Domain VS Frequency Domain ： 3D



Time Domain VS Frequency Domain ： 3D

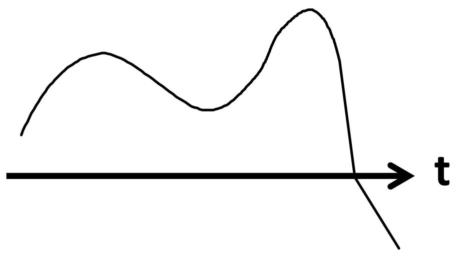


Time Domain VS Frequency Domain ： Dynamic

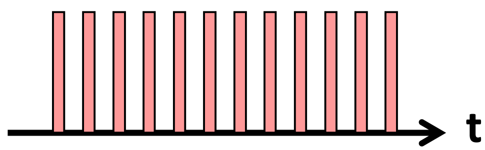


Sampling (Nature sampling)

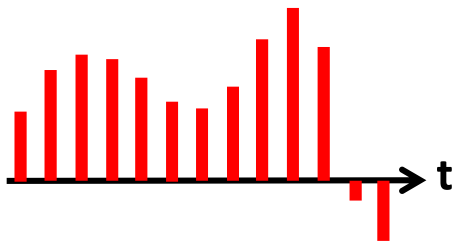
time domain



original signal $x(t)$

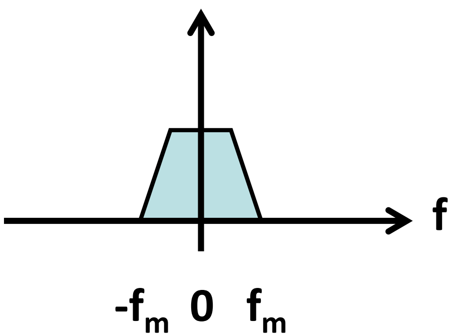


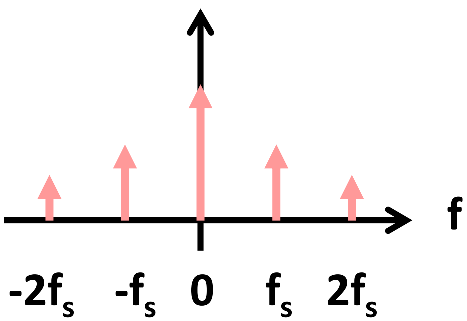
Sample pulse $x_p(t)$

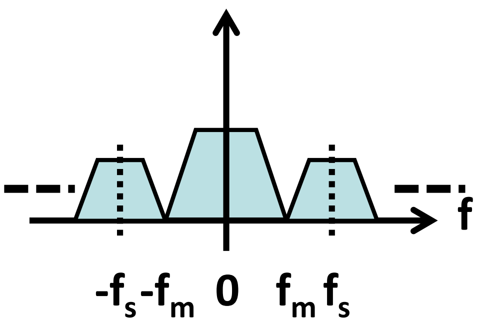


Sampled signal $x_s(t)$

frequency domain

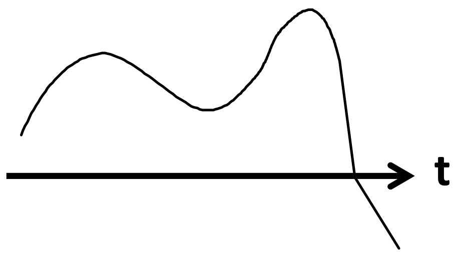




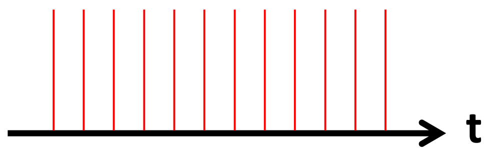


Sampling (ideal sampling)

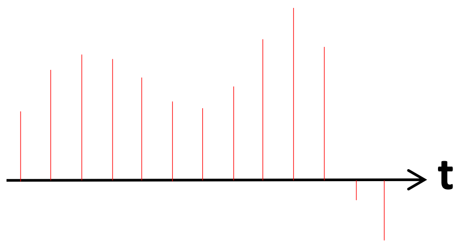
time domain



original signal $x(t)$

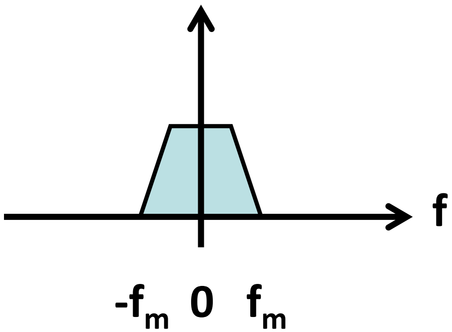


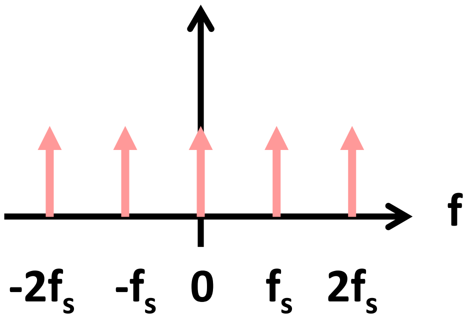
Sample pulse $x_p(t)$

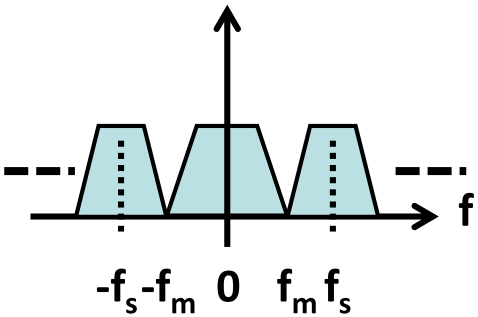


Sampled signal $x_s(t)$

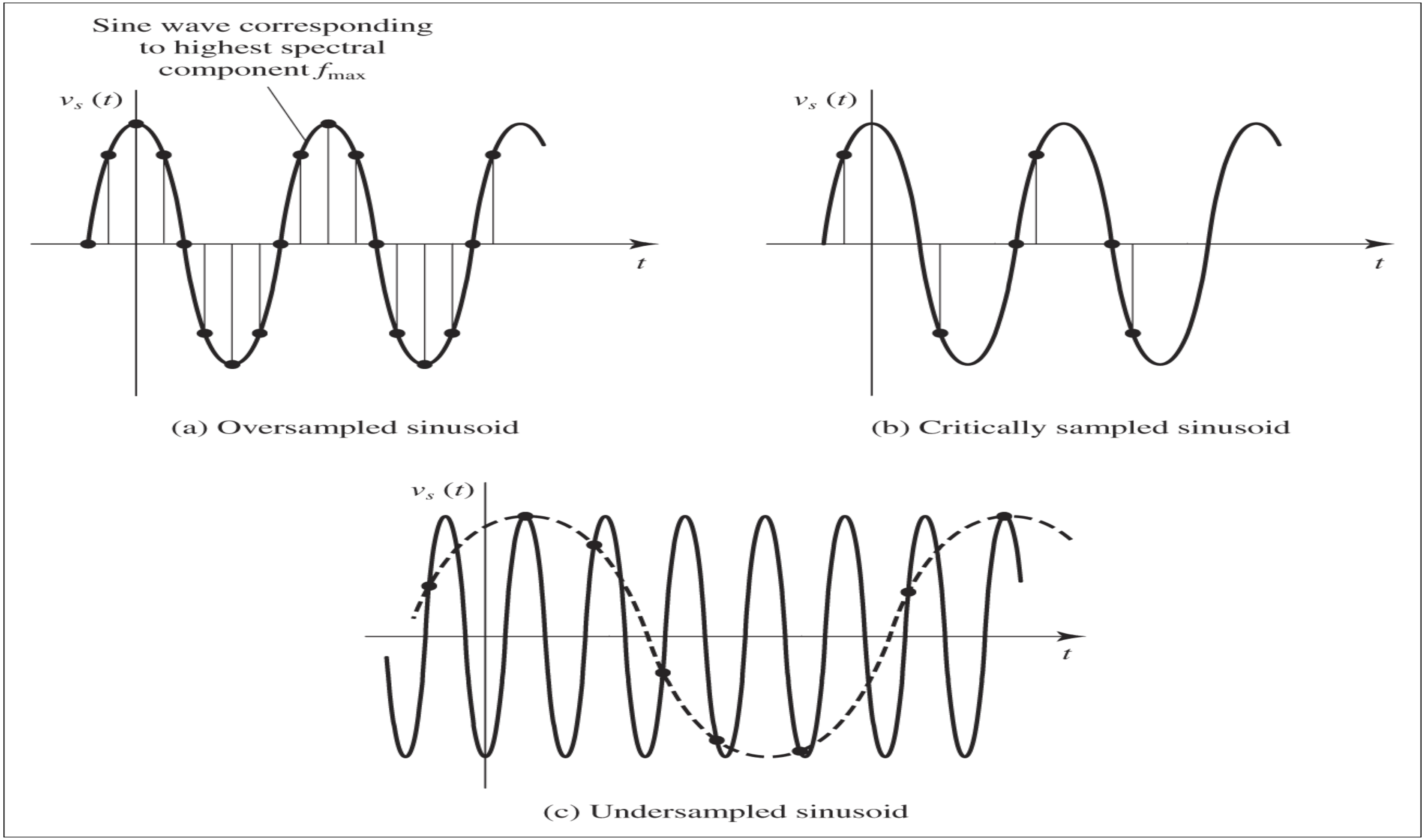
frequency domain





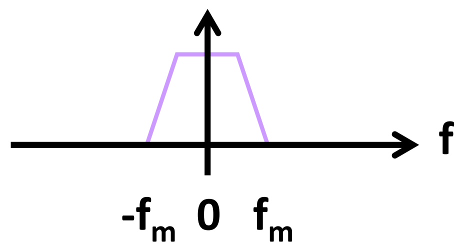


Sampling frequency (*fs*)

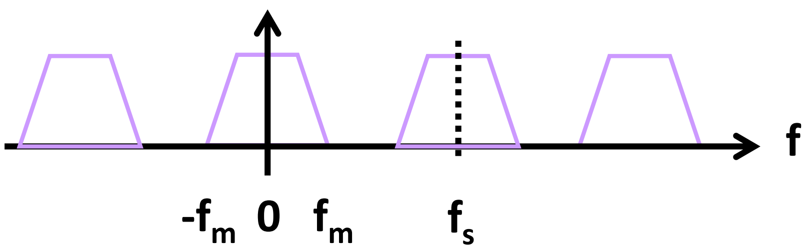


Aliasing (ideal sampling)

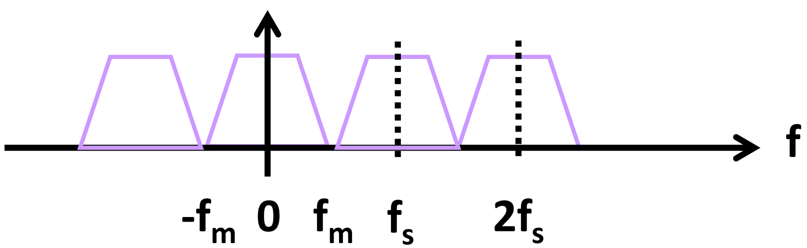
original signal



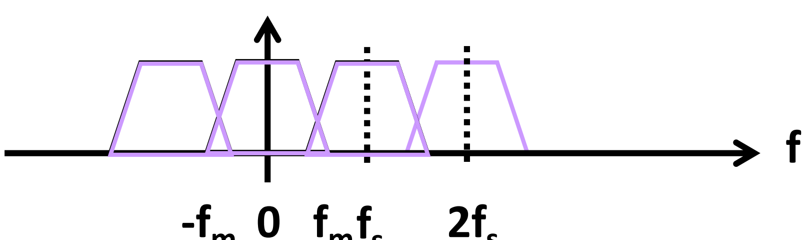
signal sampled with $f_s > 2\ f_m$



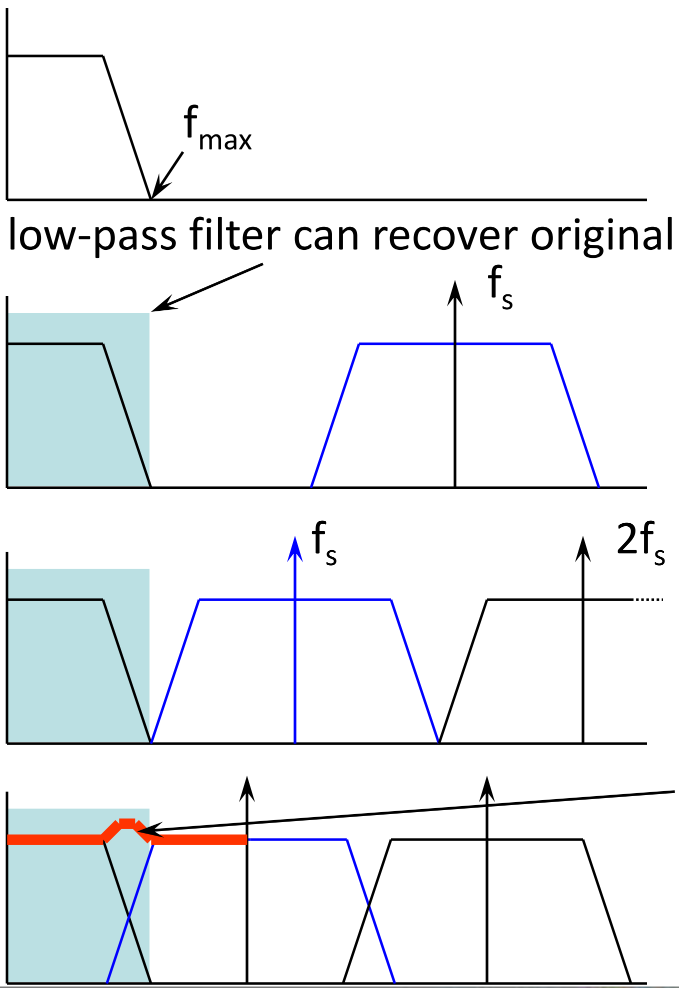
signal sampled with $f_s = 2\ f_m$



signal sampled with $f_s < 2\ f_m$
aliasing occurs



Aliasing in more detail



original signal

signal sampled with $f_s > 2\ f_{max}$

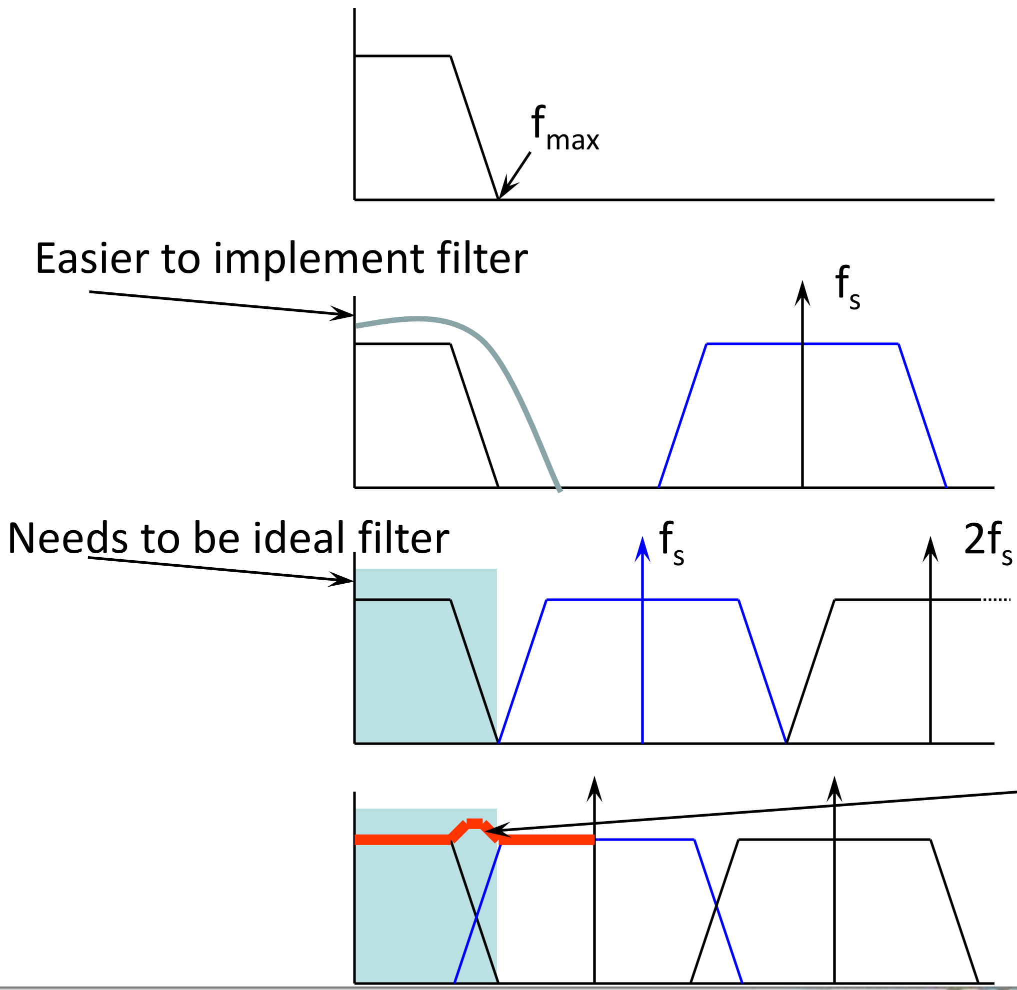
signal sampled with $f_s = 2\ f_{max}$

recovery not possible - spectra interfere

signal sampled with $f_s < 2\ f_{max}$
aliasing distortion occurs

Oversampling

low-pass filter can recover original signal



original signal

oversampling
signal sampled with $f_s > 2\ f_{max}$

signal sampled with $f_s = 2\ f_{max}$

recovery not possible - spectra interfere

signal sampled with $f_s < 2\ f_{max}$
aliasing distortion occurs

Sampling Theorem (Nyquist's Criterion)

- To prevent aliasing and hence to allow the original signal to be recovered the sampling frequency (f_s) must be given by:

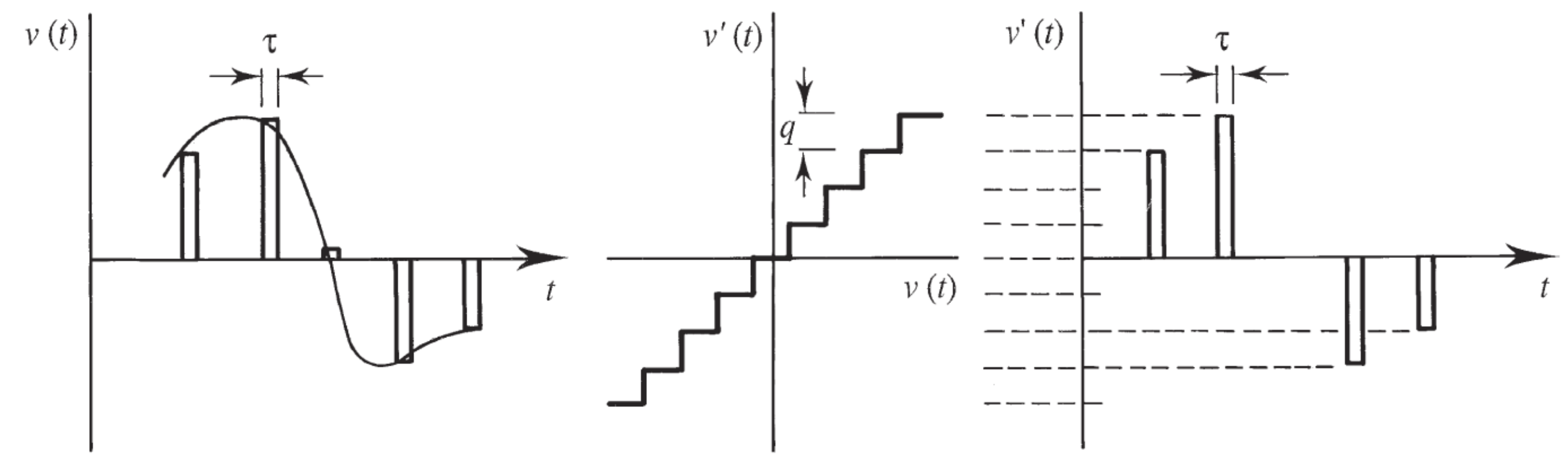
$$f_s \geq 2\, f_{max}$$

  where f_{max} is the highest frequency present in the original signal.
- This is the SAMPLING THEOREM and is a fundamental theorem.
- Notice that f_{max} is the highest frequency interest, NOT the highest frequency of present
- Oversampling makes it easier to design a simpler filter to recover the original signal

QUANTISATION

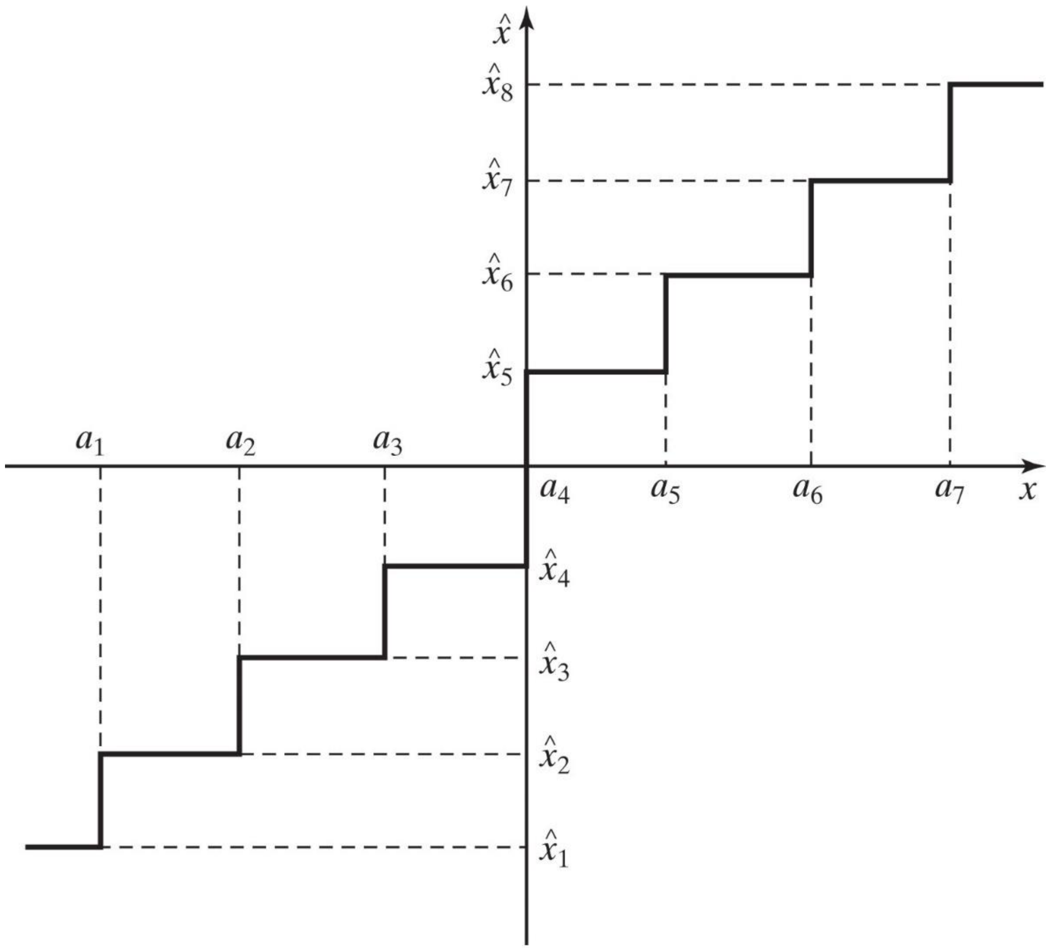
Quantisation

- Results from mapping continuous analogue values to discrete vales that can be represented digitally.
- May be uniform or nonuniform
- Pulse-code modulation (PCM) is a method used to digitally represent sampled analogue signals



(a) Analogue input signal (b) Quantiser characteristics (c) Quantised output signal

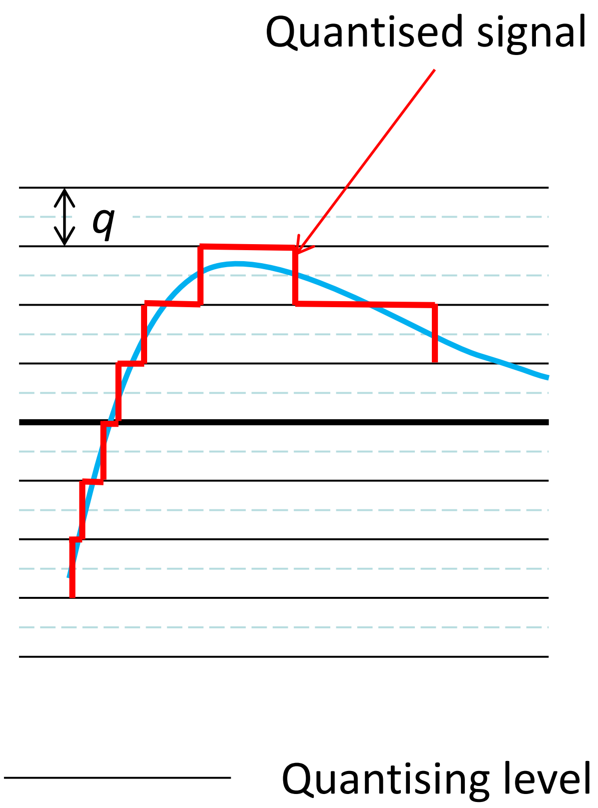
Quantisation



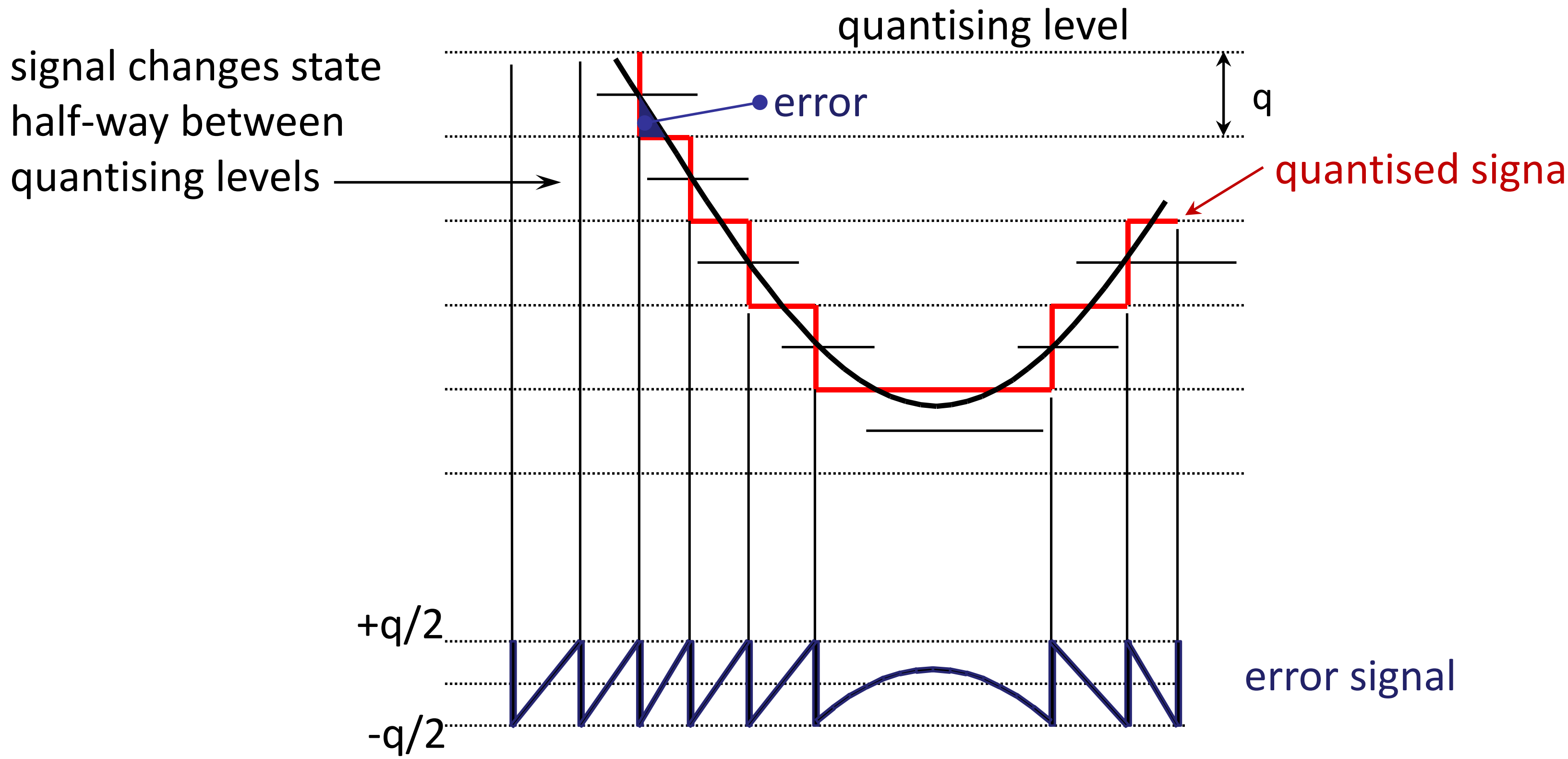
- ✓ Quantisation Region
- ✓ Quantized value of *x*
- ✓ Quantisation distortion

Uniform quantisation

- Peak-to-Peak Voltage, $V_{pp}=V_p-(-V_p) = 2V_p$
- Quantisation interval, q, (step size) uniformly distributed over the full range
- Approximation will result in an error no larger than $\pm q/2$

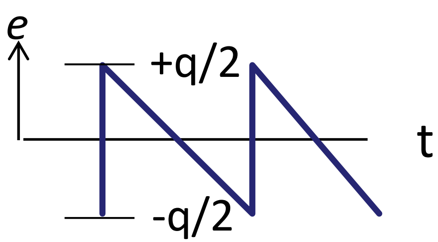


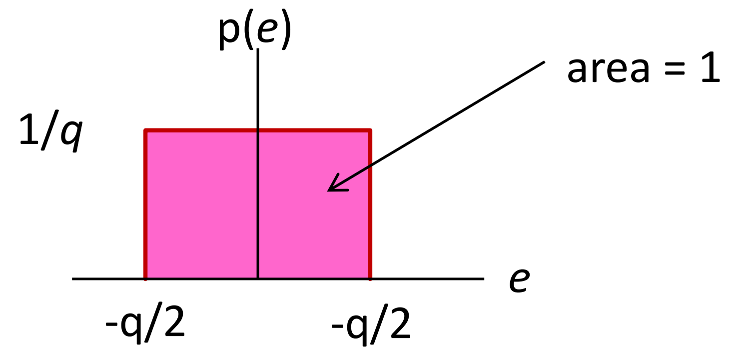
Quantising distortion



Quantising error power

Error (e) is approximately sawtooth over the quantisation region, apart from the dwell regions.





A sawtooth waveform has a uniform pdf: all values are equally likely.
The area under the pdf must be 1 so that the amplitude is $1/q$.
Note that p(e)=0 outside the range $+q/2$ to $-q/2$

the power of the quantisation error is $$P_Q = \int_{-\infty}^{\infty} e^2 p(e) de = \int_{-q/2}^{q/2} e^2 \frac{1}{q} de = q^2/12$$

Notes

- this holds for reasonable well-behaved signals without frequent dwell regions.
- quantising error leads to distortion, not noise, because the same input will always produce the same output.
- the statistics of the distortion are independent of the statistics of the input.
- this approximation shows that the distortion power is constant and depends only on the step size.

Signal to Quantisation Noise Ratio (SQNR)

The average signal to quantisation noise ratio (SQNR) is defined as by:

$$SQNR = \frac{P_s}{P_Q} \qquad P_s = \frac{M^2-1}{12} q^2 \ (\mathrm{V}^2)$$

$$SQNR = M^2 - 1 \cong M^2$$

where M is the number of quantisation levels and P_S is the signal's power:

$$SQNR = 20 \log_{10} M \ (dB)$$

$$SQNR = 4.8 + 6n - \alpha_{dB}$$

Where α is the ratio of peak to mean signal power, ${V^2}_{peak} / \overline{V^2}$

Queen Mary University of London

Review: What is dB?

- The intensity of sound is measured in decibels (dB).
- This is not a linear scale, as the human ear does not perceive volume changes in a linear way.
  - Doubling the sound energy arriving at the human ear is not perceived as being twice as loud.
  - The sound energy has to be increased more than twice before it is perceived as being twice as loud.
- The ear's response to sound changes is logarithmic and therefore audio volume controls are similarly logarithmic.

Logarithmic volume

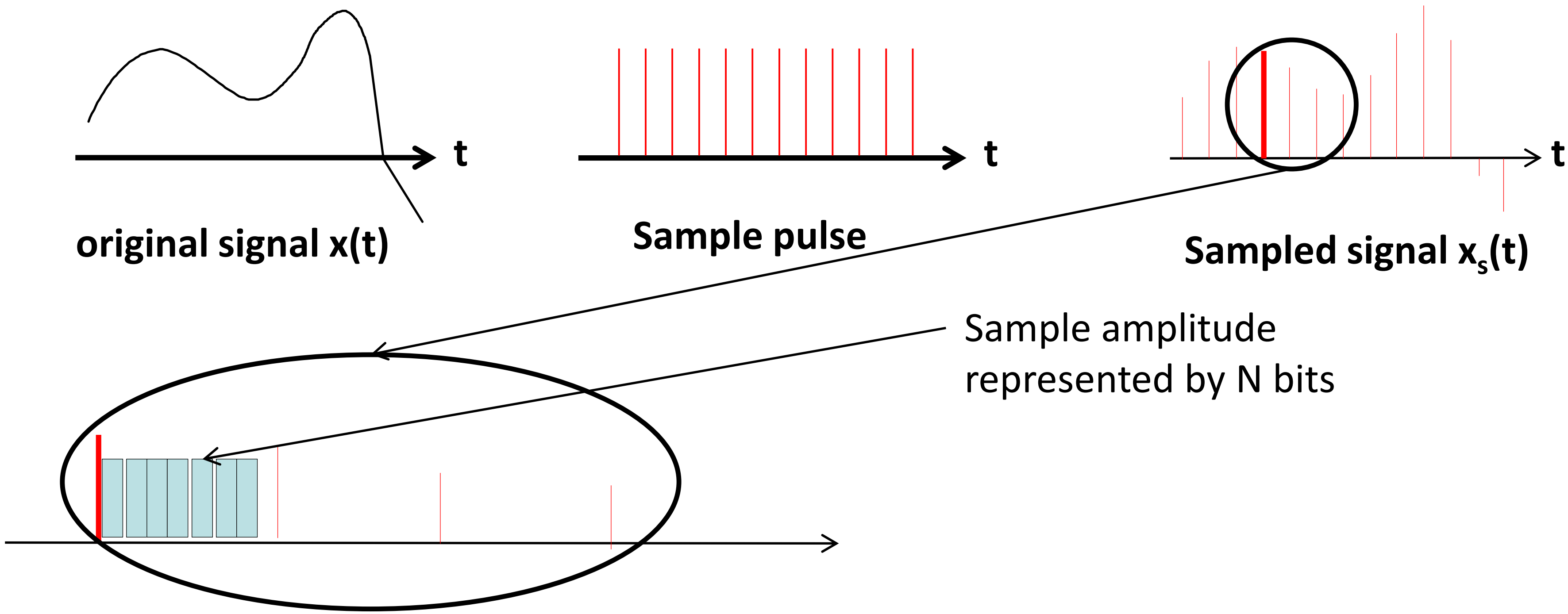
- How to calculate logarithmic volume
  - **$x=10*\log_{10}(y)$,** x in dB and y in decimal
  - $10*\log_{10}(1000)=10*3=30$dB
  - $10*\log_{10}(2)=10*0.3010=3$dB
- Turning up the audio volume results in logarithmic increases in output and not linear increases.
  - Doubling the volume is a 3dB increase, while quadrupling the volume is a 6dB increase.
  - Increasing the volume by eight times is a 9dB increase and a 30dB increase turns up the volume to 1,000 times its previous amplitude.

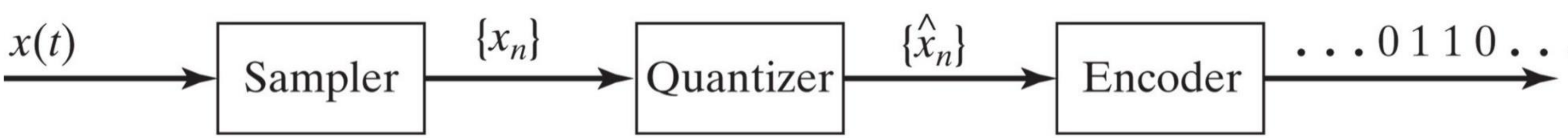
Everyday noise levels

DB Level	Example
160	Jet engine
130	Large orchestra at full blast
120	Start of pain threshold
110	Power tools
100	Loud rock music
90	Subway
80	Car/Truck
70	Normal conversation
60	Background noise in busy store
50	Background noise in house or office
40	Quiet conversation
30	Whisper
20	Quiet living room
10	Background noise in recording studio
0	Hearing threshold

Pulse Code Modulation (PCM)

time domain

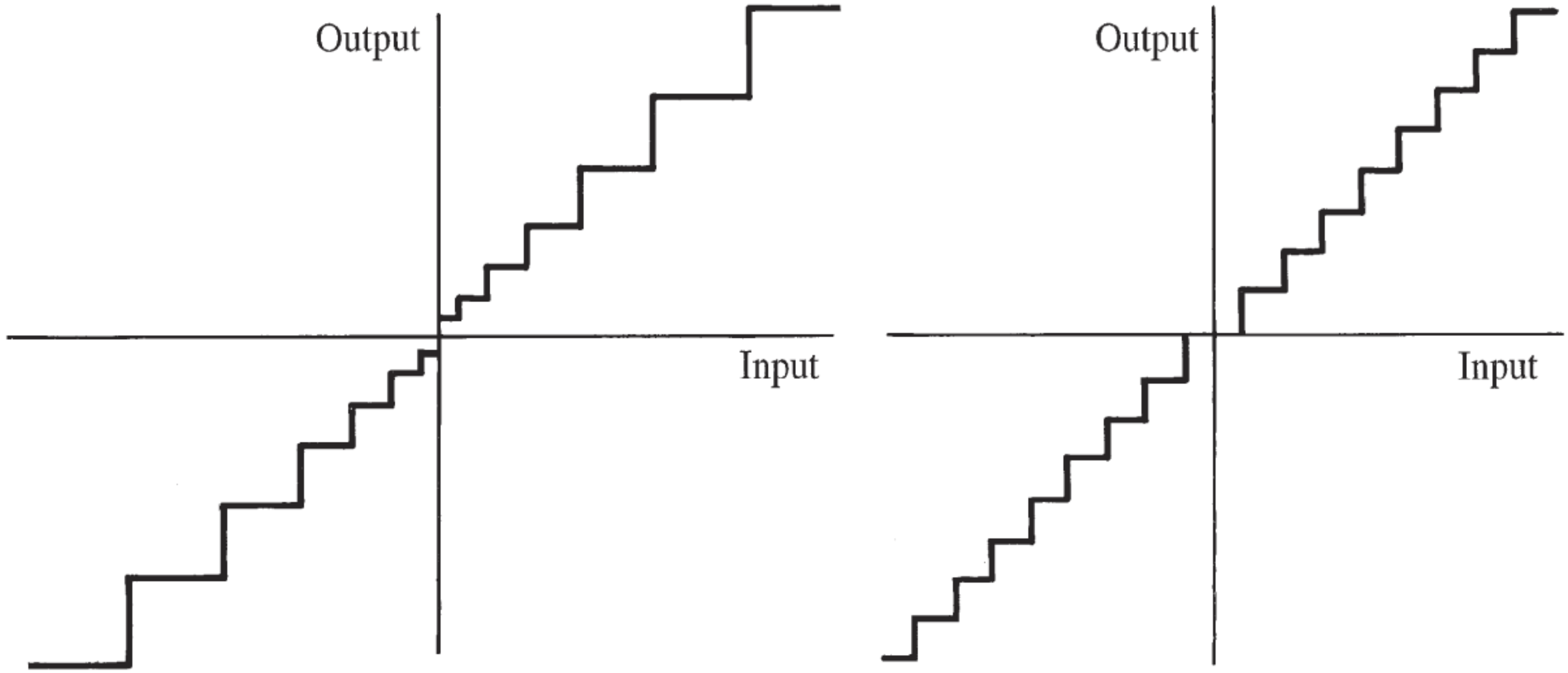




Speech and non-uniform Quantisation

- Speech power has a wide dynamic range – whispering, shouting
- SQNR is worse for lower powers as quantisation noise is the same for all signal magnitudes ($q^2/12$).
- Non-uniform quantisation can provide fine quantisation of the weak signals and coarse quantisation of the strong signals.
- 8-bits per sample not sufficient for good speech encoding with uniform quantisation.
- Solution is to use nonuniform quantisation:
  - Step-size varies with amplitude of sample.
  - For larger amplitudes, larger step-sizes are used
  - 'Nonuniform' because step-size changes from sample to sample.

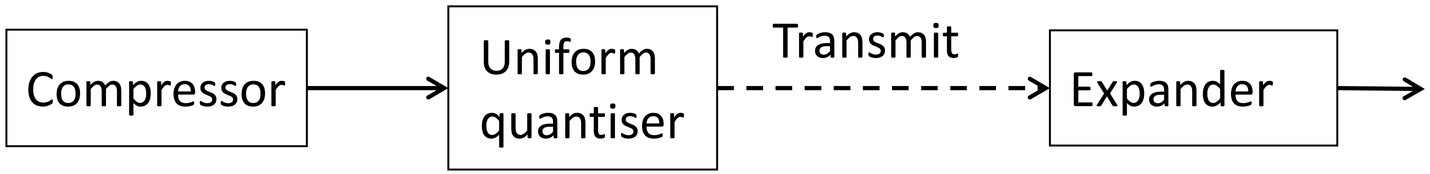
Uniform Quantisation and Nonunifrom Quantisation



(a) Non-linear quantisation (b) Linear quantisation

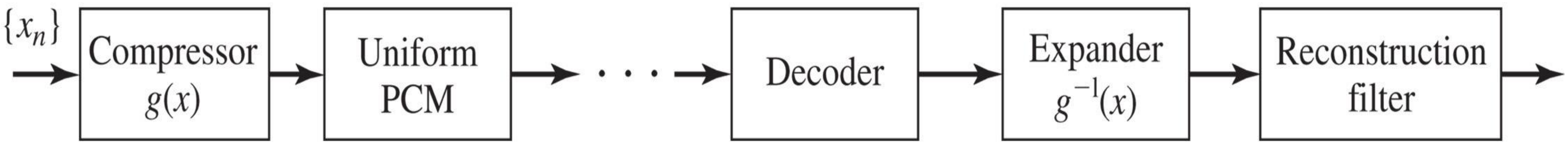
Nonuniform quantisation - principle

- Nonuniform quantisation uses logarithmic compression and expansion.
- Compress at the transmitter and expand at the receiver.
- Compression changes the distribution of the signal amplitude.
  - Lower amplitude signals strength to higher values of quantisation.
  - As the result the compressed speech signal is now more suitable for uniform quantisation.
- The logarithmic compression and expansion function is also called Companding.

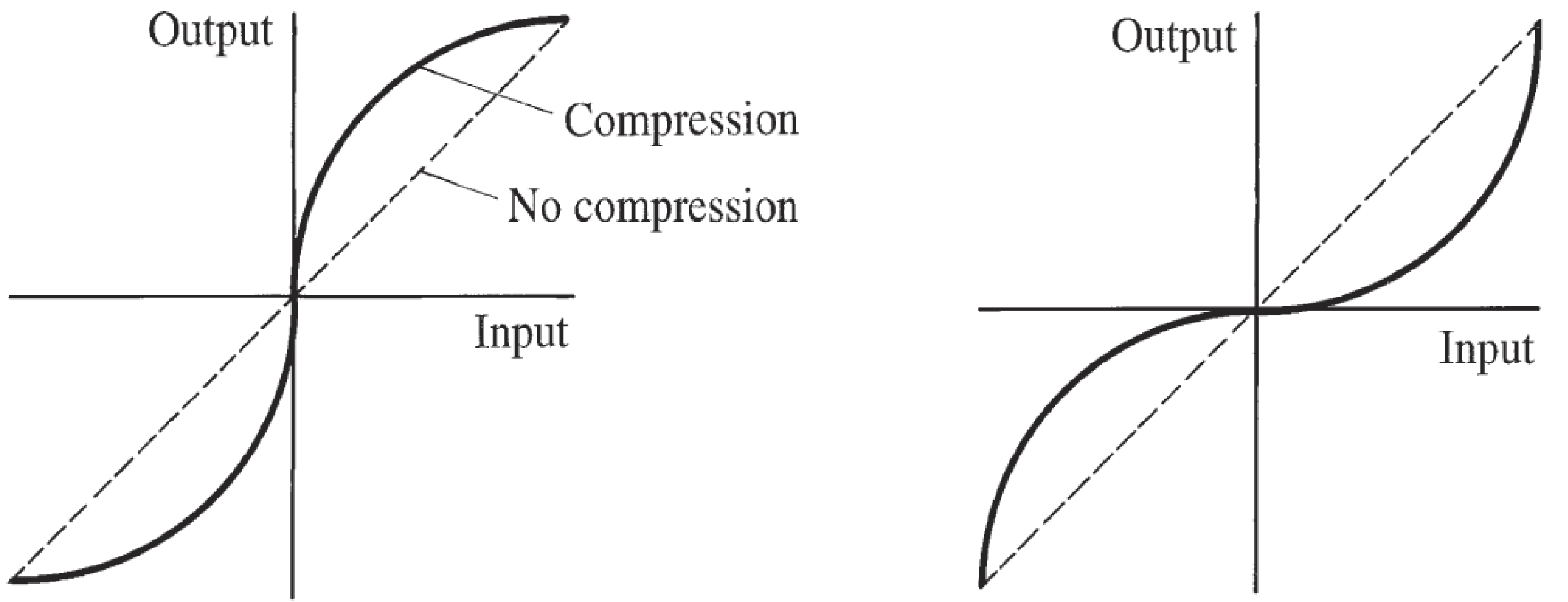


Implementation of companding (in principle)

- Pass x(t) thro' **compressor** to produce y(t).
- y(t) is quantised uniformly to give y'(t) which is transmitted or stored digitally.
- At receiver, y'(t) passed thro' **expander which** reverses effect of compressor.
- analogue implementation uncommon but shows concept well.



Typical Compander characteristics



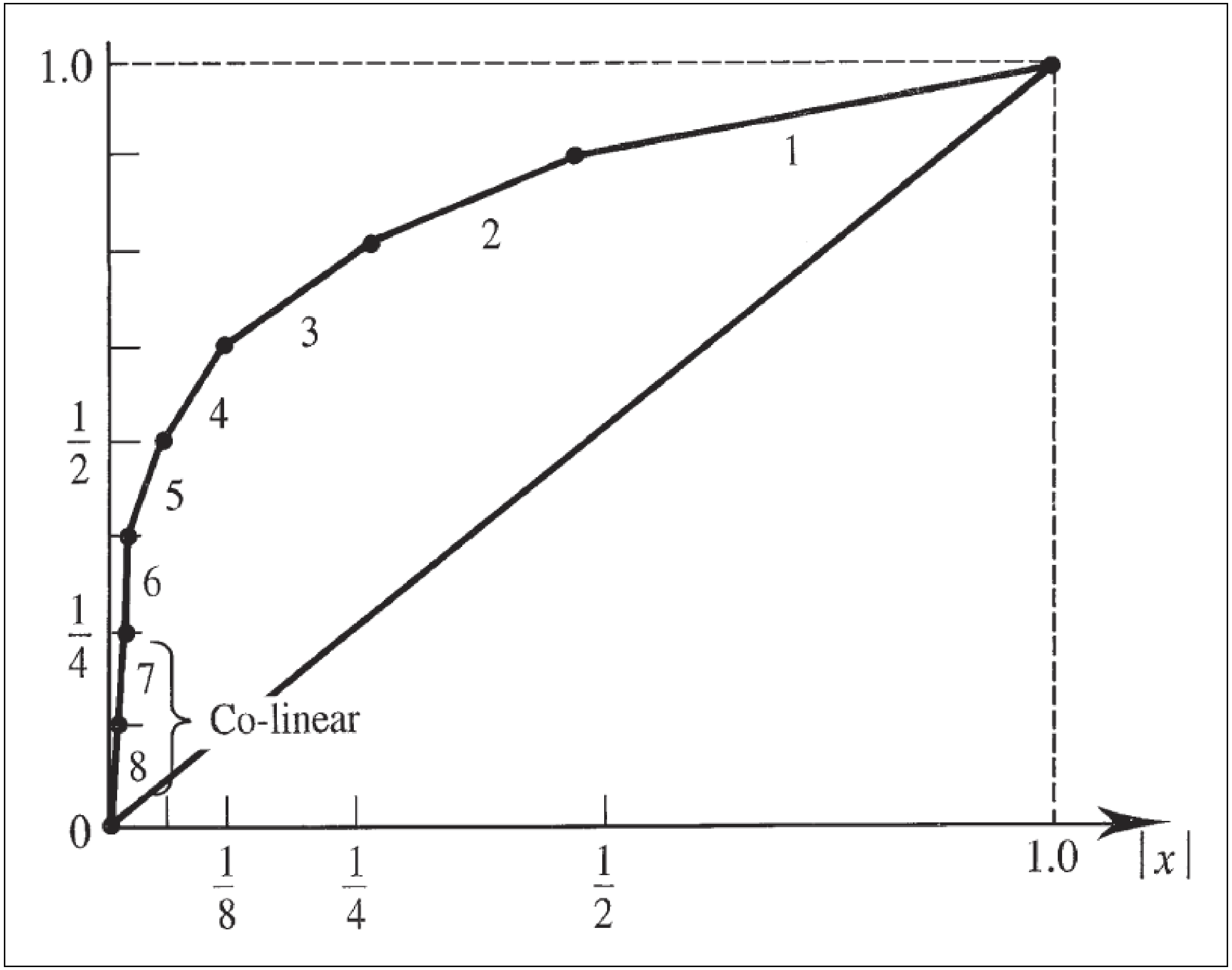
(a) Compression characteristic

(b) Expansion characteristic

Companding

- There are two companding standards for telephony: A-law (G711a - used mainly in Europe) and μ-law (G711u - used in North America and Japan).
- Implemented as a segmented, piece-wise linear approximation.
- 8-bit code consist of
  – i) polarity bit P (range is ±V)
  – ii) 3 segment decoding bits XYZ
  – iii) 4 bits (abcd) specifying intra segment value on a linear scale

13-segment compression A-law



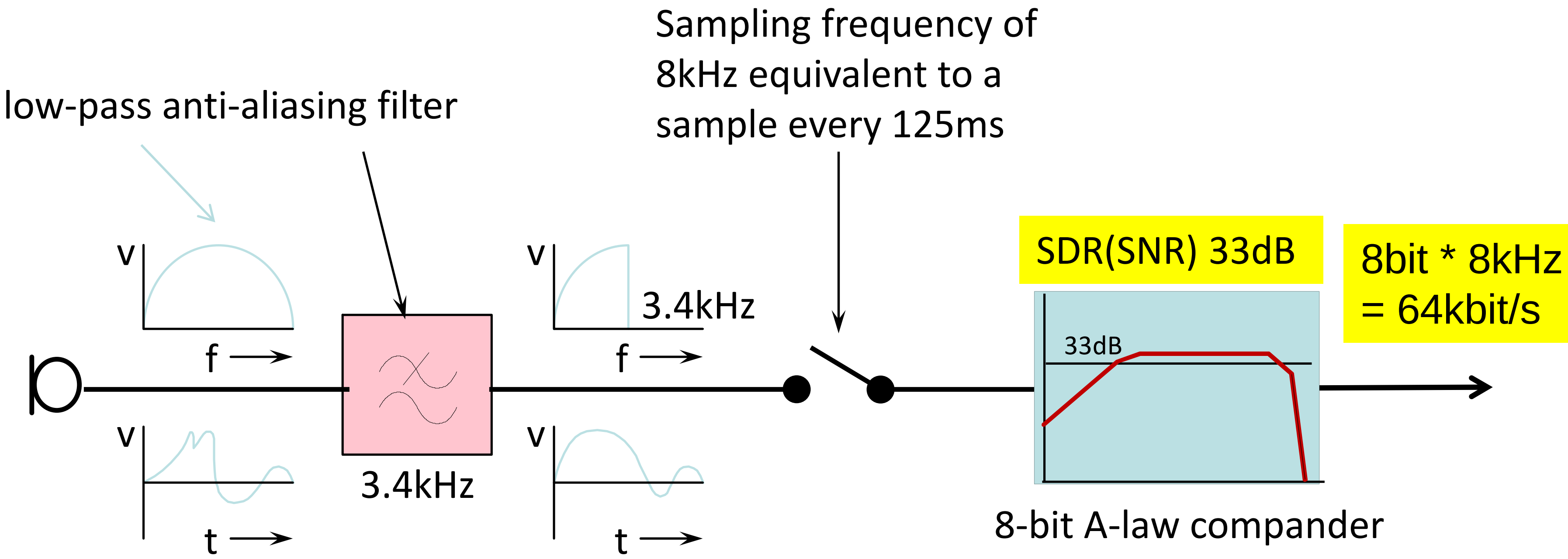
A-law encoding table

Segment	Coder input range	output code	quantum interval
0	0-V/128	P 000 abcd	V/2048
1	V/128-V/64	P 001 abcd	V/2048
2	V/64-V/32	P 010 abcd	V/1024
3	V/32-V/16	P 011 abcd	V/512
4	V/16-V/8	P 100 abcd	V/256
5	V/8-V/4	P 101 abcd	V/128
6	V/4-V/2	P 110 abcd	V/64
7	V/2-V	P 111 abcd	V/32

P is a polarity bit

abcd is a 4-digit *intra*-segment code – linear quantisation

PCM telephony

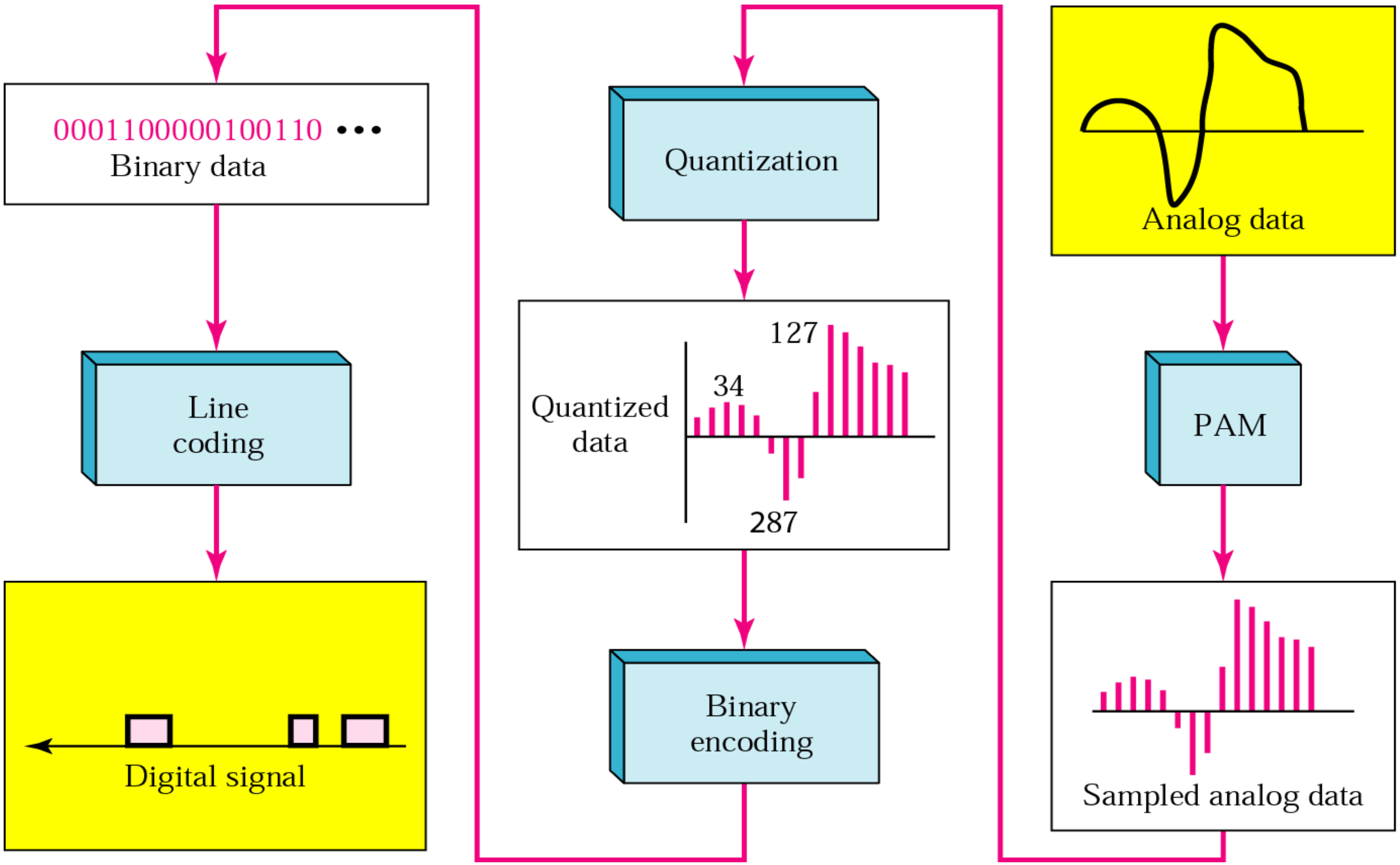


By bandlimiting the incoming speech signal to 3.4kHz
and sampling at 8kHz, the sampling theorem is satisfied

You need to remember this diagram:

- anti-aliasing filter gives bandwidth of 3.4kHz to the sampler
- sampling rate of 8kHz – 1 sample every 125 µs
- 8-bit (a-law) non-linear quantisation

Pulse-Code Modulation



Pulse-Code Modulation (PCM)

- In pulse-code modulation (PCM) we can identify three components: sampler, quantiser and encoder.
- The encoder converts the sequence of quantised amplitudes into a sequence of bits. Hence, the bit rate R_B can be calculated as

$$R_B = f_s \times N_B \ bps$$

Where f_s is the sampling frequency and N_B is the number of bits per quantisation level.The quantiser can be either uniform or non-uniform. In telephony, speech f_s = 8 kHz, N_B = 8; hence R_B = 64 kbps.

Review: Analogue & Digital Bandwidth

- **Analogue bandwidth** of a medium is expressed in cycles per second (Hz).

- **Digital bandwidth** is expressed in bits per second.

- Analogue bandwidth is the ***range of frequencies*** that a medium can pass.

- Digital bandwidth is the ***maximum bit rate*** that a medium can pass.

Analogue v Digital (bandwidth)

- Always depend on the ***application*** and the ***available bandwidth***.

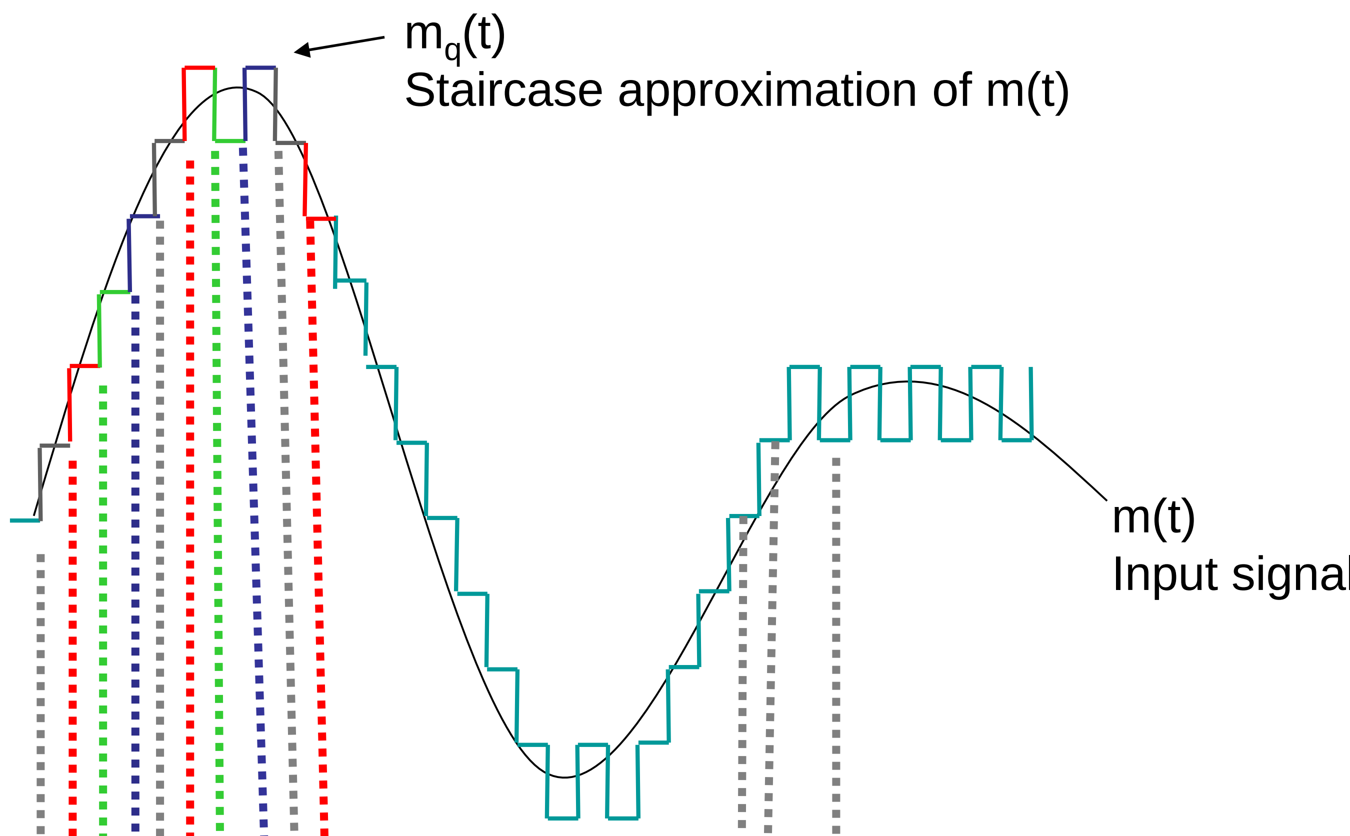
- In general, the bandwidth of the signal to be sent has to be smaller than the bandwidth the medium can have.
    - It is not good sending a 4000 Hz signal through a 1000 Hz medium

DELTA MODULATION

Delta modulation (DM)

- Simpler than PCM
- Provides a staircase version of the message signal by referring to the difference between the input signal and its approximation
- Quantization is done using 2 levels:
  - Positive difference: $+\Delta$
  - Negative difference: $-\Delta$
- Provided that the input signal does not change too rapidly from sample to sample, this approximation works well

DM illustration

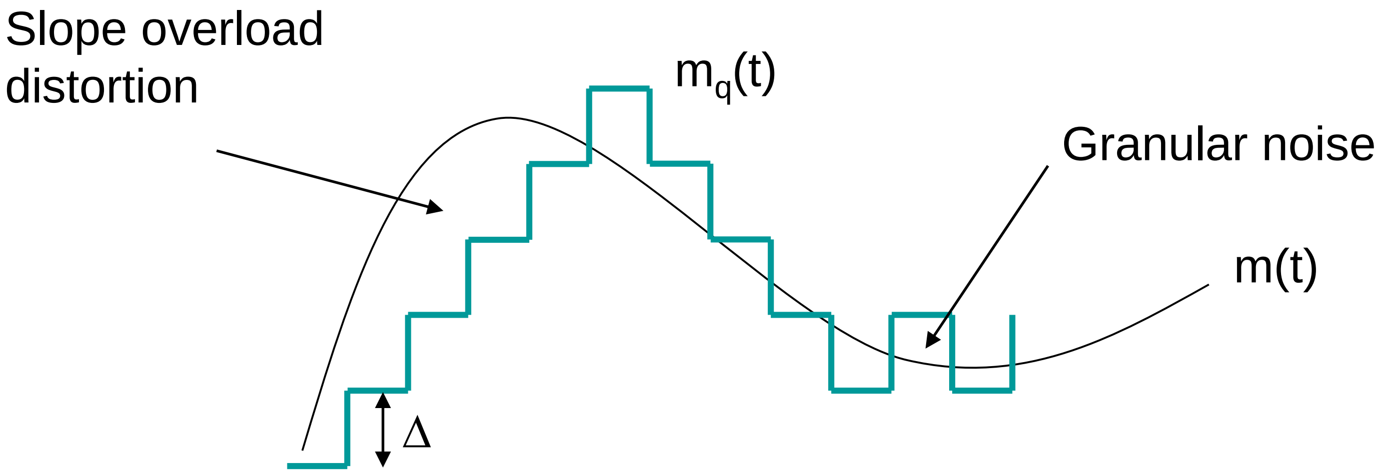


Binary sequence at modulator output

111111010000000010101110100101010

DM quantisation error

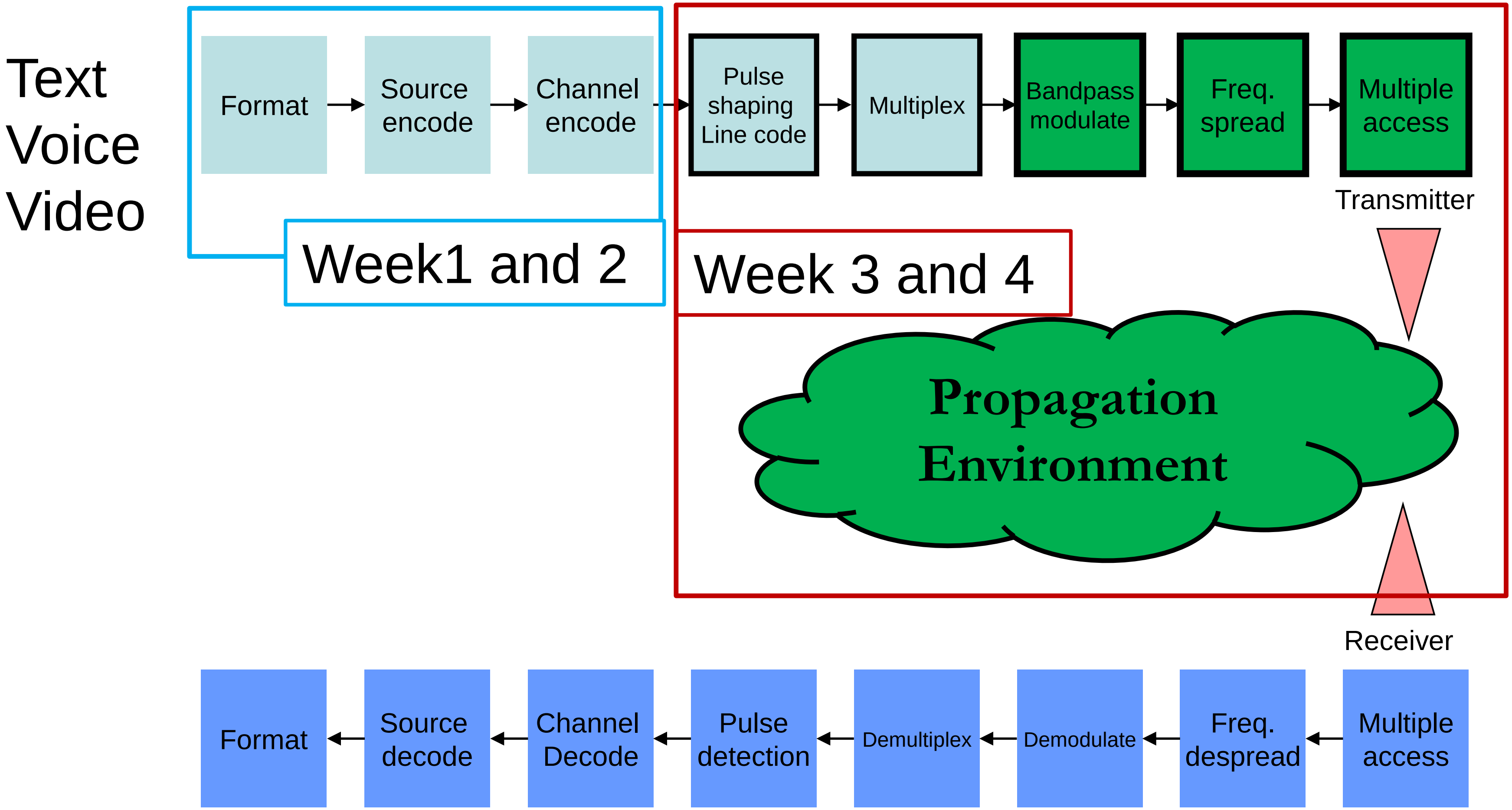
- Slope overload distortion
- Granular noise



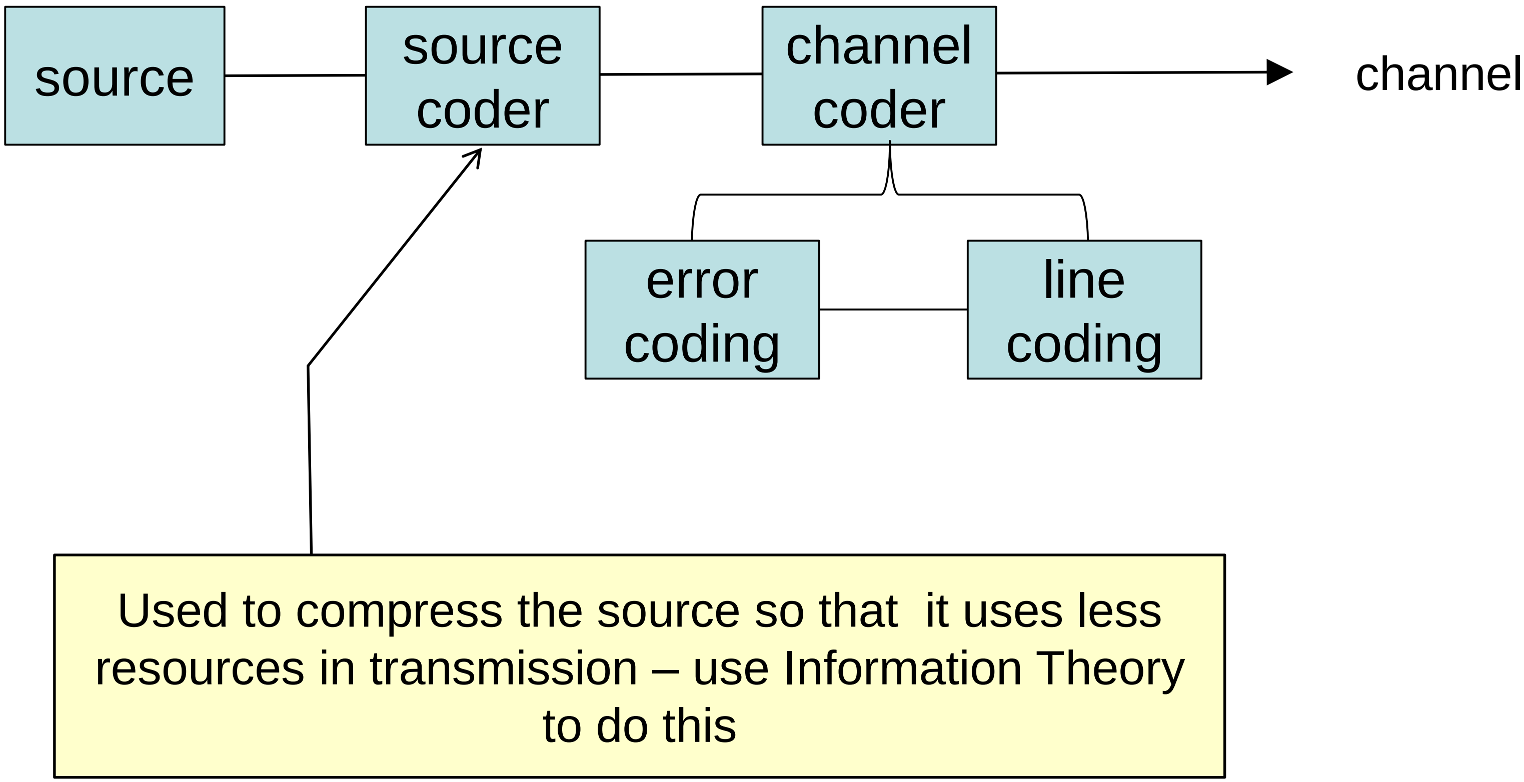
To minimise slope overload distortion $\dfrac{\Delta}{T_s} \geq \max\left|\dfrac{dm(t)}{dt}\right|$

INFORMATION THEORY

Overview of Wireless Communication System



Coding elements



Memoryless source

- Probability of an event occurring does not depend on what came before.
- Produces *symbols* (e.g. A, B, C, D) from an *alphabet*.
- Probability of a particular symbol is fixed
- Sum of probabilities is 1

Source with memory

- Probability of a symbol depends on previous symbol
- In English the probability of “u” occurring after “q” is higher than after any other letter
- Lots of real sources in this category and coding like JPEG and MPEG exploits this to produce smaller file sizes
- Can be very complex

Probability reminder

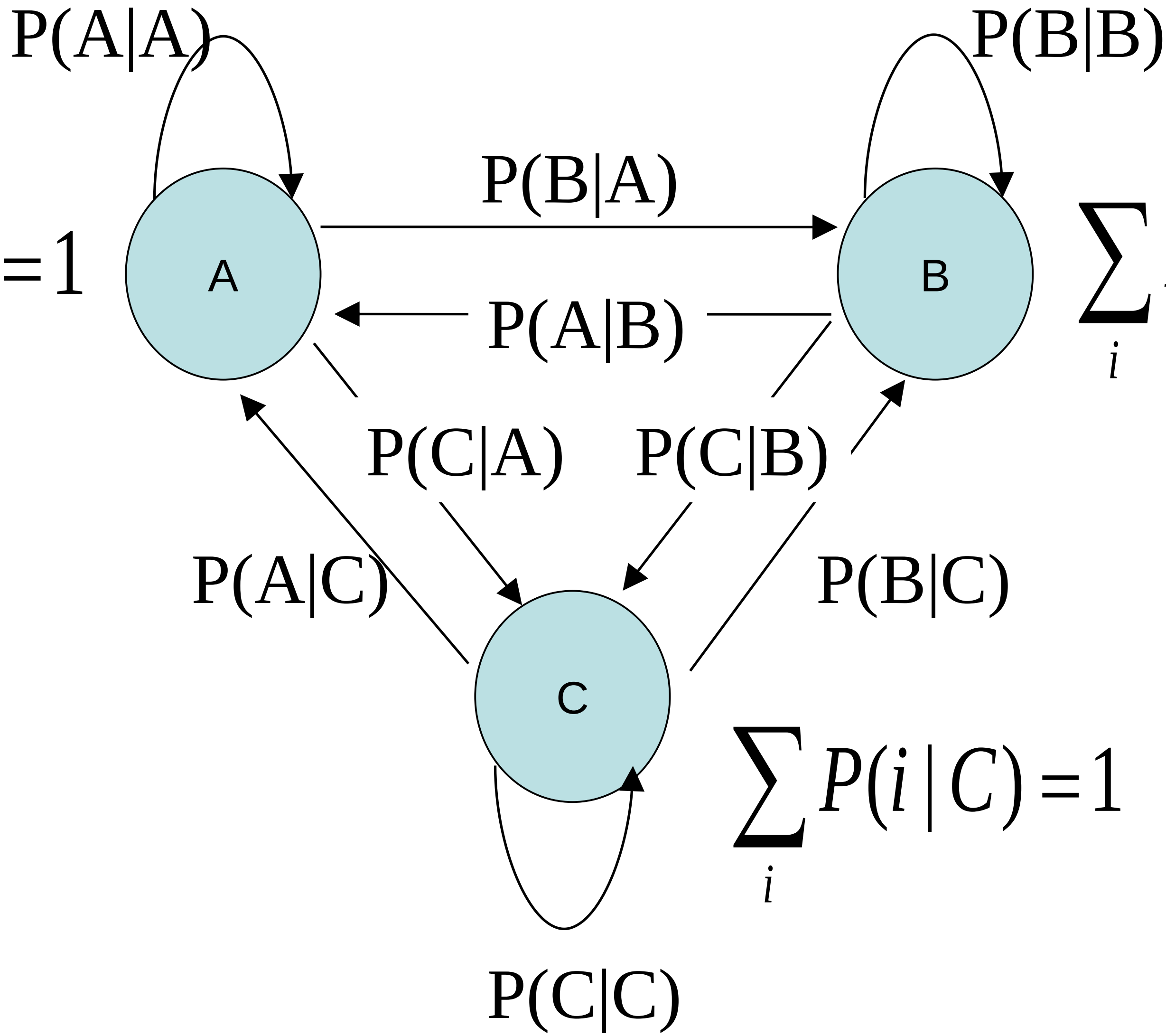
- P(A) is probability of event A
- P(B|A) is conditional probability – probability of event B occurring given that A has occurred.
- Joint probability $P(A \cap B)$ is probability that A and B occur – this is also written P(A,B)
- Here the notation P(A;B) is used to denote the probability of the event pair A followed by B – note this is not the same as (2)

Queen Mary
University of London

Probability reminder

- Independent sources
  - $P(A \cap B) = P(A)P(B)$
    - condition for being statistically independent
  - $P(B|A) = P(B)$
- Conditional probability
  - $P(A \cap B) = P(A|B)P(B)$
  - $P(A|B) = P(A)P(B|A)/P(B)$

$$\sum_i P(i \mid A) = 1$$



$$\sum_i P(i \mid B) = 1$$

$$\sum_i P(i \mid C) = 1$$

$$P(A) = P(A|A)P(A) + P(A|B)P(B) + P(A|C)P(C)$$

$$P(A)(1 - P(A|A)) = P(A|B)P(B) + P(A|C)P(C)$$

And similarly for the others

$$P(B)(1 - P(B|B)) = P(B|A)P(A) + P(B|C)P(C)$$

$$P(C)(1 - P(C|C)) = P(C|A)P(A) + P(C|B)P(B)$$

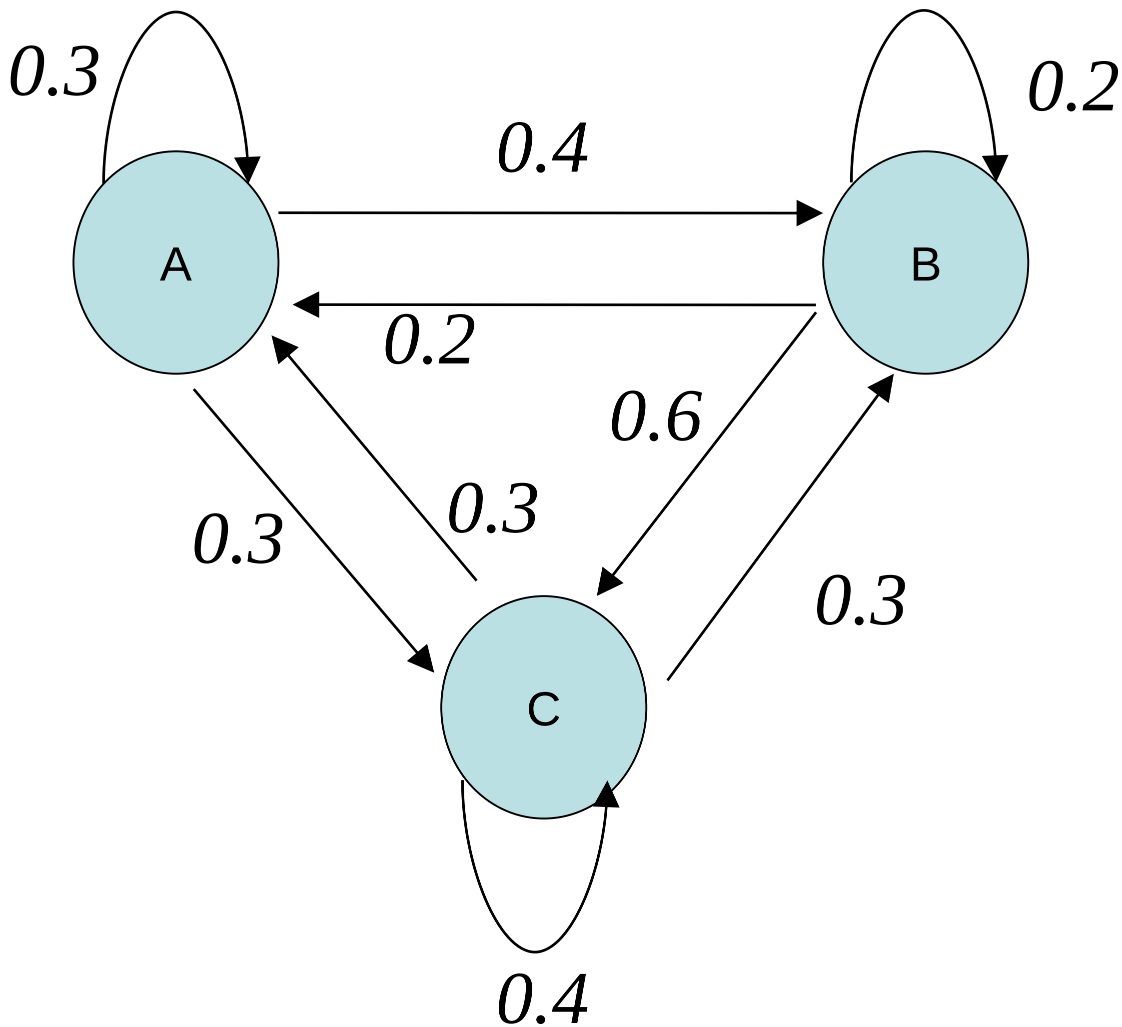
Also we must get one of the symbols so:

$$1 = P(A) + P(B) + P(C)$$

Taking this last equation plus 2 others gives 3 simultaneous equations that can be solved for P(A), P(B) and P(C)

Example

P(A|A)	0.3
P(B|A)	0.4
P(C|A)	0.3
P(A|B)	0.2
P(B|B)	0.2
P(C|B)	0.6
P(A|C)	0.3
P(B|C)	0.3
P(C|C)	0.4



$$P(A)(1-P(A|A)) = P(A|B)P(B) + P(A|C)P(C)$$

$$P(B)(1-P(B|B)) = P(B|A)P(A) + P(B|C)P(C)$$

$$P(C)(1-P(C|C)) = P(C|A)P(A) + P(C|B)P(B)$$

- 0 = - 0.7 P(A) + 0.2 P(B)+ 0.3 P(C)
- 0 = 0.4 P(A) - 0.8 P(B) + 0.3 P(C)
- 0 = 0.3 P(A) + 0.6 P(B) – 0.6 P(C)
- 1 = P(A) + P(B) + P(C)

To solve need to include this line plus 2 others

Solving gives:

P(A)	0.2703
P(B)	0.2973
P(C)	0.4324

- Need to calculate the probabilities of pairs of symbols.
- $P(B;A) = P(A|B)P(B)$ and similarly for all other pairs

$P(A;A)=P(A\|A)P(A)$	0.0811
$P(A;B)=P(B\|A)P(A)$	0.1081
$P(A;C)=P(C\|A)P(A)$	0.0811
$P(B;A)=P(A\|B)P(B)$	0.0595
$P(B;B)=P(B\|B)P(B)$	0.0595
$P(B;C)=P(C\|B)P(B)$	0.1784
$P(C;A)=P(A\|C)P(C)$	0.1297
$P(C;B)=P(B\|C)P(C)$	0.1297
$P(C;C)=P(C\|C)P(C)$	0.1730
Total	**1**

Information theory

- What is information ?
  - Everything has information.
  - Everywhere has information.
- Information in communication.
  - Signals: carry the information, a physical concept.
  - Symbols: describe the information by mathematics

Information theory..

- Measure of information is a measure of uncertainty - more certain data contains less information.
- Which has more information?
  - Now the weather outside is sunny .
  - You tell me it is sunny now. 100% happened
  - You tell me it is snowing now. Can't happen
- A lie contains infinite information.

Information theory

- Unit of information now generally in bits
  - p is the probability of the event and I is the information content.
  - Notice that we take logs to base 2 to get the units in bits.

$$I = \log_2 (1/p)$$

  - Other bases give other units - e is Nat, 10 is Det (or Hartley)

- Remember that $$\log_2(x) = \frac{\log_{10}(x)}{\log_{10}(2)}$$

- Excel function LOG(number,base) takes logs to any base.

Entropy

- If a source emits a number of symbols, the **entropy** (H) is the *average information content per symbol:*

$$H = \sum_i p_i \, log_2(1/p_i)$$

p_i is the probability of the i'th event occurring

$$\sum_i p_i = 1$$

Queen Mary University of London

Example

Determine the entropy of a source that transmits 4 symbols with the probability of each symbol being emitted being:

Symbol:	*A*	*B*	*C*	*D*
Probability	0.15	0.15	0.2	0.5

$$H = \sum p_i \log_2 (1/p_i)$$

$$H = 0.15\log_2(1/0.15) + 0.15\log_2(1/0.15) + 0.2\log_2(1/0.2) + 0.5\log_2(1/0.5)$$

which gives H=1.785 bit/symbol

Maximum Entropy

- This occurs when all events have the same probability.
- Max entropy for N symbols is

$$H = \sum_{i=1}^{N} p_i \log_2 \left(1/p_i \right)$$

but $p_i = 1/N$

$$\text{so } H = \sum_{\mathrm{i}=1}^{N} (1/N) \log_2 \left(N \right) = \log_2 \left(N \right)$$

- In the previous example this would be when the probability for each event is 0.25
  - which would give H= 4* 0.25 $\log_2(4)$ = 2 bits/symbol

Source coding

- Aims to reduce the number of bits transmitted.
- Previous example:

Symbol:	*A*	*B*	*C*	*D*
Probability	0.15	0.15	0.2	0.5

- 4 symbols means 2 bits required ($2^2 = 4$ combinations) if no account is taken of probability of event
- Entropy of 1.785 bit/symbol means that is all that is required
- So the code efficiency is 1.785/2=0.875 (87.5%)
- How??? - many methods of source coding such as
  - Video compression (MPEG)
  - Think of other examples …..

Principle of source coding

- The event with high probability uses short code
- The event with low probability uses long code
- Consider the same example:

Symbol:	*A*	*B*	*C*	*D*
Probability (p_i)	0.5	0.25	0.125	0.125
code	0	10	110	111
Number of bits (n_i)	1	2	3	3

- H=1.75 bit/symbol
- Average code length $=\sum p_i n_i = 0.5\times1+0.25\times2+0.125\times3\times2=1.75$

Huffman coding

- Put events in DESCENDING order of probability
- Combine the lowest 2 and re-order
- Repeat until only one value of "1"
- At each split, mark top as "0" and bottom as "1" (or vice-versa as long as same throughout)
- Go backwards from final "1" to each symbol in turn

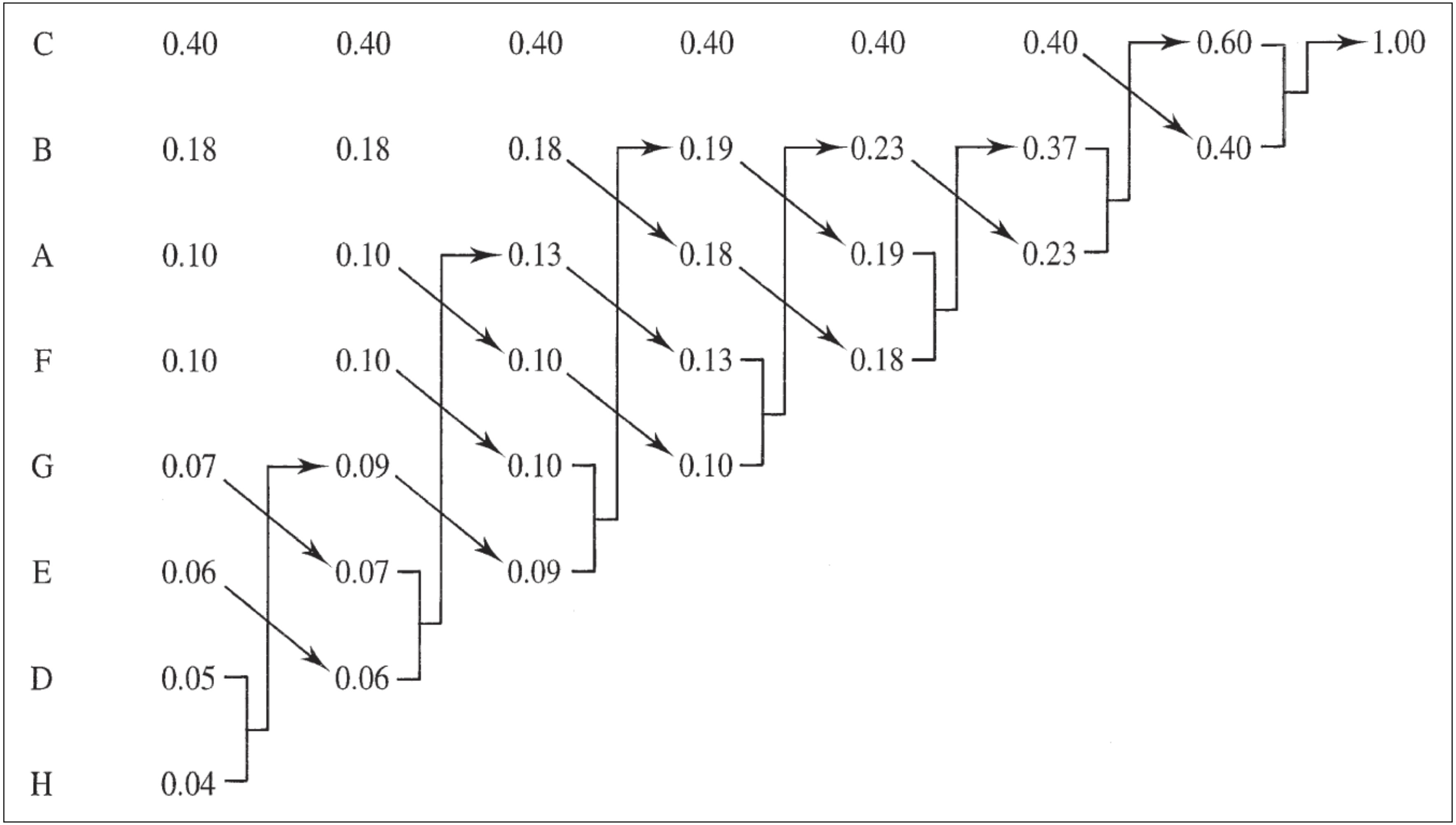
Huffman coding ..

- Add 1/0 at each split as appropriate
- Write down:
  - pattern: these are bits transmitted
  - no of bits for each symbol
  - probability*no of bits for each symbol (pi×Ni)
- $\Sigma (p_i \times N_i)$ is average no of bits transmitted
- Compression ratio= no source coding bits / average number of bits transmitted
- Efficiency=H / average code length

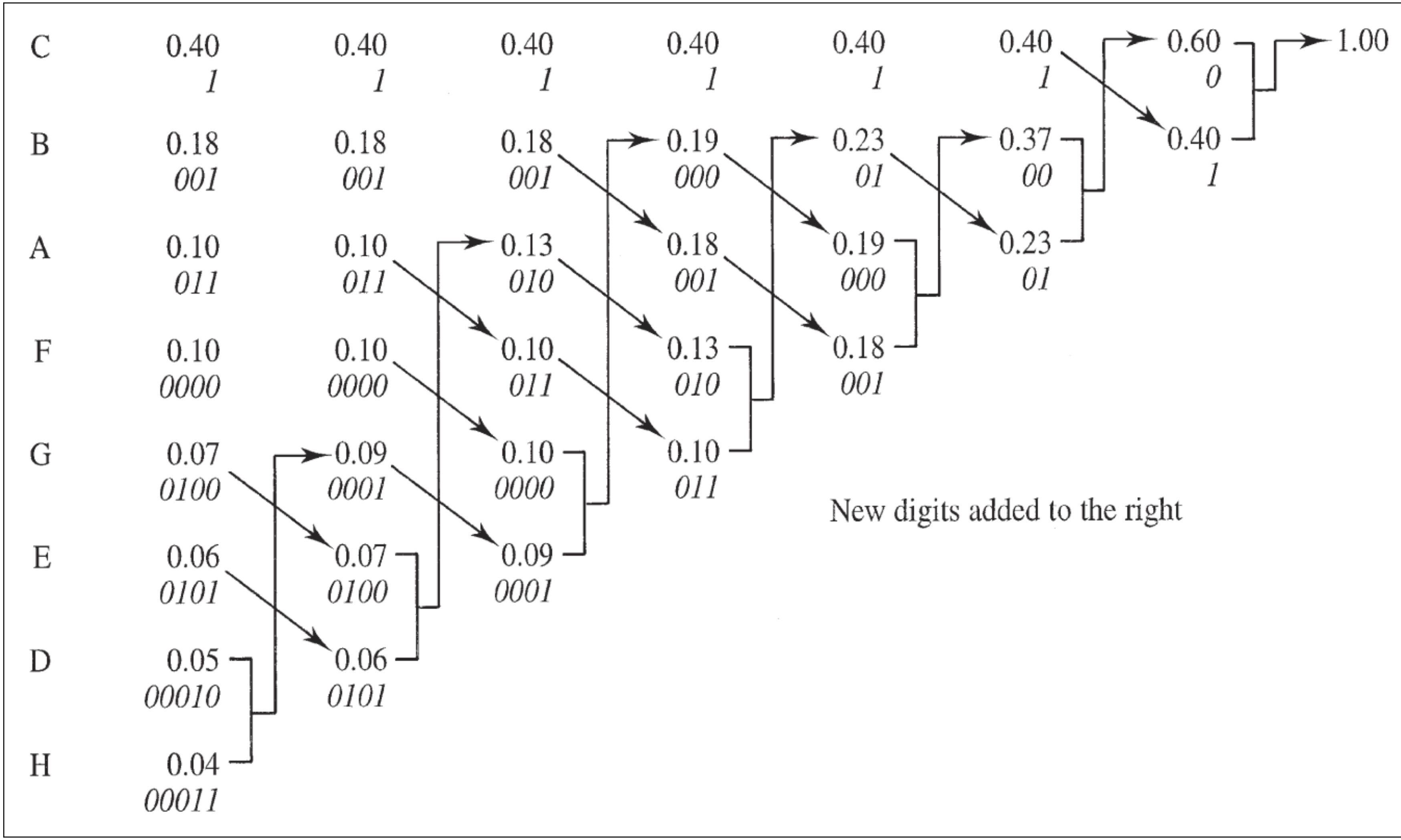
Huffman coding Procedure (8-symbol)

Symbol	C	B	A	F	G	E	D	H
Probability	0.40	0.18	0.10	0.10	0.07	0.06	0.05	0.04
Codeword								

Huffman coding – Reduction



Huffman coding — Splitting



Huffman calculation example

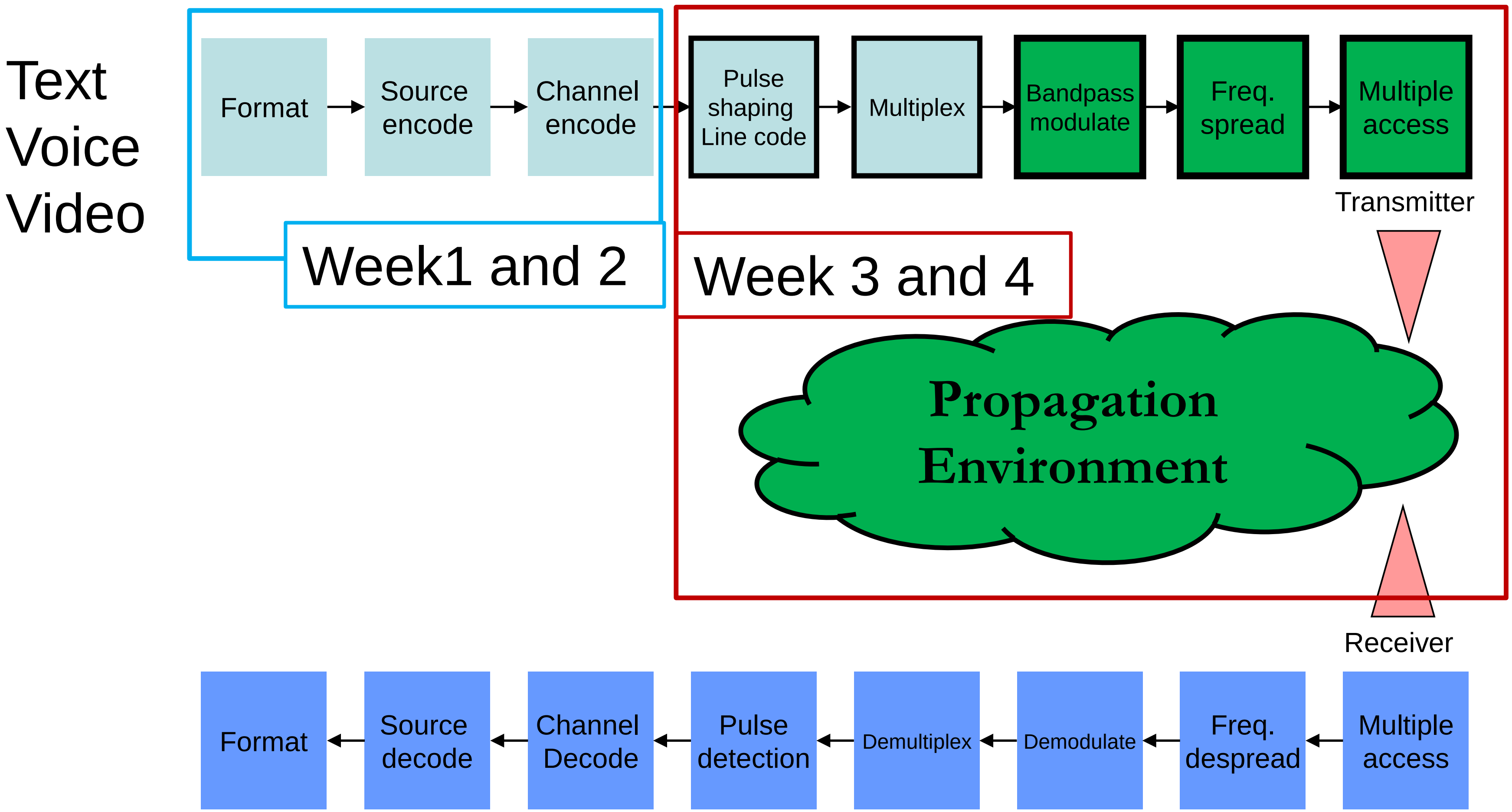
- What is the source entropy H?
- What is the source efficiency $\eta_{source} = \frac{H}{H_{max}}$?
- What is the code efficiency $\eta_{code} = \frac{H}{L}$?

Unique decodability

- Variable length codes must be unique
- Codes A=0 B=011 C=110 are NOT unique
  - 0110 could be AC or BA
- Codes A=0 B=10 C=110 are unique but errors are a problem
  - 0110 can only be AC
  - 0110 (AC) corrupted to 0010 is read as 0 0 10 AAB

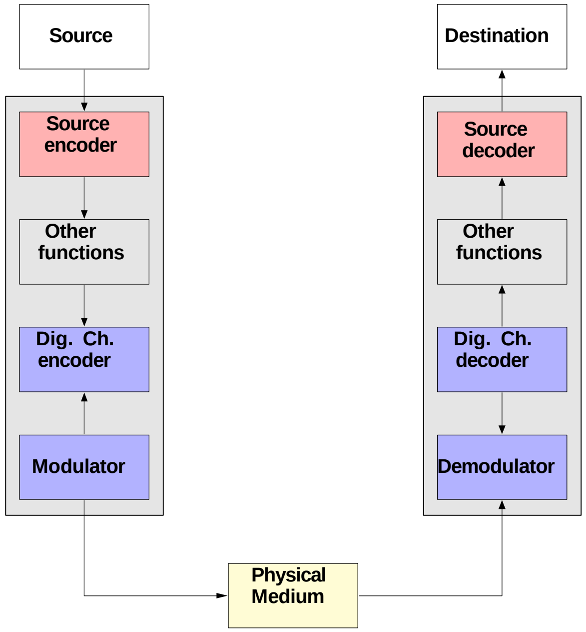
DIGITAL CHANNELS

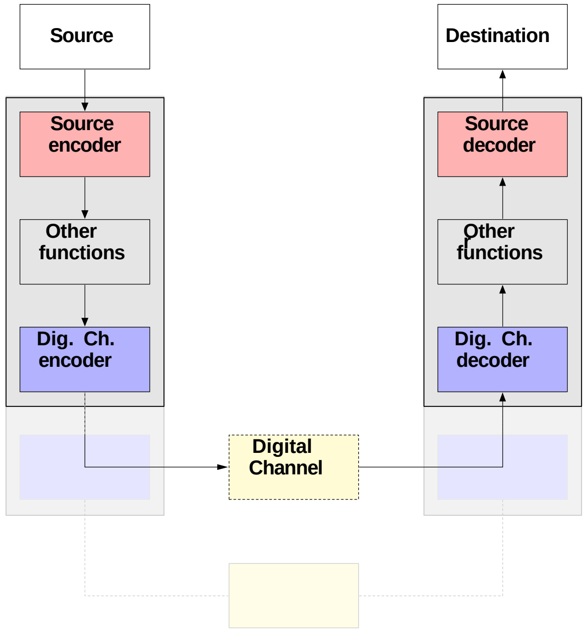
Overview of Wireless Communication System



The digital channel

The digital channel is a mathematical abstraction. It describes the relationship between the sequence of symbols that leaves the digital channel encoder and the one that reaches the digital channel decoder.





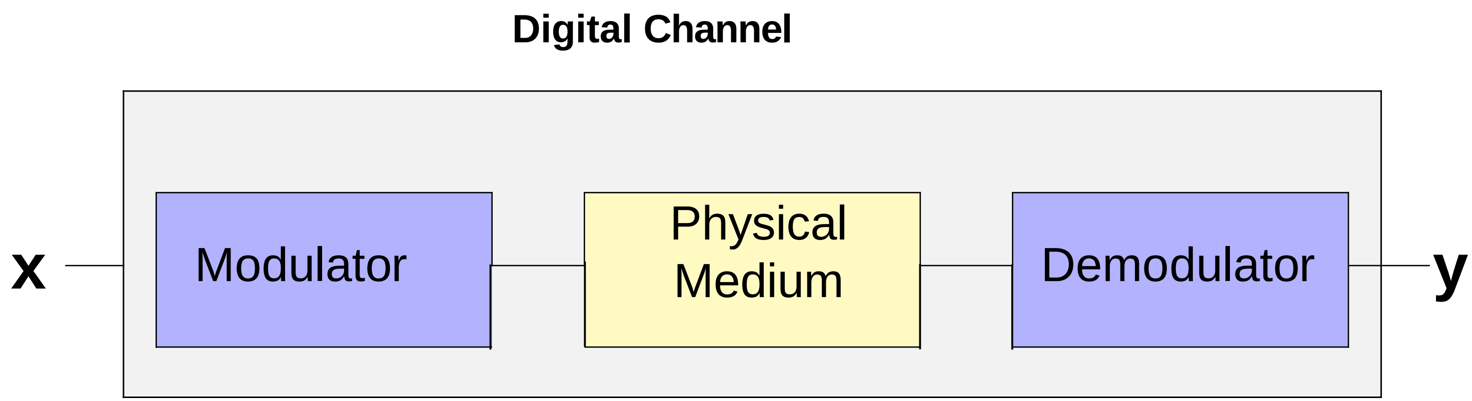
The digital channel

Mathematically, a digital channel establishes a relationship between two sequences of symbols, the input sequence x = $[x_1, x_2, \ldots, x_N]$ and the output sequence y = $[y_1, y_2, \ldots, y_N]$.

This relationship is described statistically by the probability of observing **y** at the output when **x** is at the input: $p(\mathbf{y}|\mathbf{x})$.

Mathematical definition of a digital channel

In digital communication technologies, the digital channel is an abstraction encompassing: (1) the modulator at the transmitter, (2) the physical medium and the (3) demodulator at the receiver.



Hence, the modulator, the physical medium and the demodulator determine the relationship between the input symbol sequence x and the output symbol sequence y.

Mathematical definition of a digital channel

We will say that there is an error in the output sequence y whenever $y \neq x$. There are many factors that can contribute to errors in the output sequence y, such as

- ❖ Attenuation
- ❖ Noise
- ❖ Bandwidth limitations.
- ❖ Multipath propagation.

Errors cannot be anticipated and hence, there is no deterministic relationship between the output sequence y and the input sequence x. A digital channel will be defined by a probabilistic relationship between the output and input sequence, namely the probability of observing y when we use x as the input, $p(y|x)$.

The digital memoryless channel

Digital memoryless channels have the property that the probability of observing a symbol y_i in the output sequence y, only depends on the input symbol at the same time instant, x_i.

As a consequence of this property, the conditional probability defining the channel can be expressed as follows:

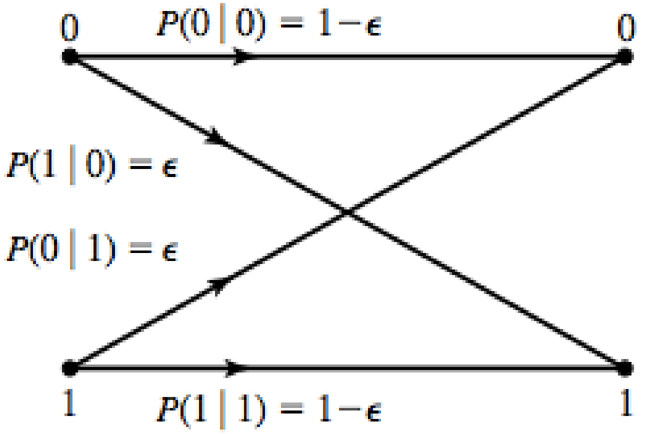
$$p(y|x) = \prod_{i=1}^{N} p(yi|xi)$$

where $p(y_i|x_i)$ is the probability of observing symbol y_i at the output when symbol x_i is present at the input.

The binary symmetric channel

The binary symmetric channel is a special case of memoryless channel for which the input and output alphabet are binary $X = 0, 1$ and $Y = 0, 1$ and the symbol conditional probabilities are defined as:

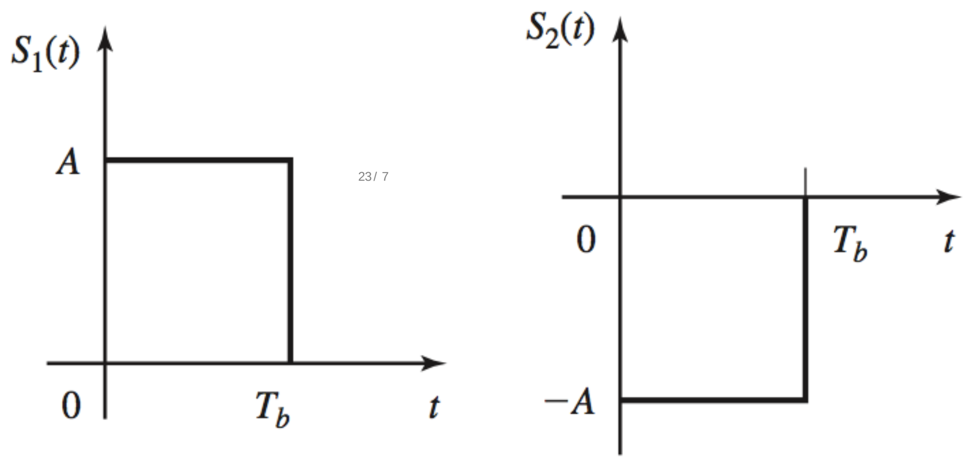
$$p(1|1) = p(0|0) = 1 - \varepsilon$$
$$p(1|0) = p(0|1) = \varepsilon$$



The value ε is known as the **crossover probability**. Note that $p(1|0) + p(0|0) = 1$ and $p(1|1) + p(0|1) = 1$.

Example: Binary PAM and AWGN channel - Modulator

Binary PAM is the simplest digital modulation method. Symbol 1 is converted into a pulse waveform $S_1(t)$ of amplitude A, whereas symbol 0 is converted into a pulse waveform $S_2(t)$ of amplitude $-A$.

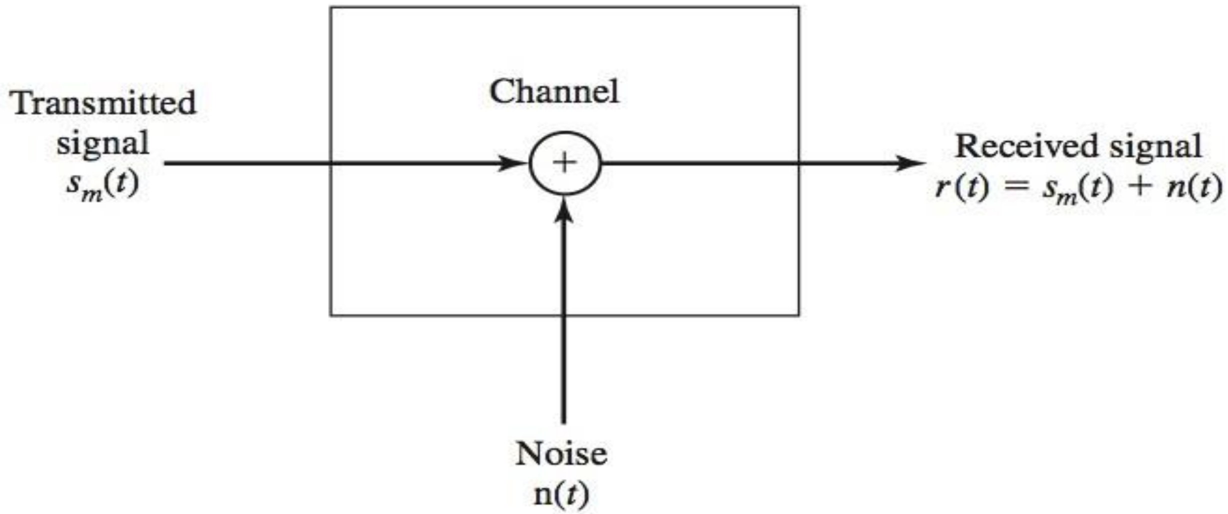


23 / 7

A sequence of bits is transmitted as a sequence of the corresponding waveforms at a bit rate $R_b = 1/T_b$, where T_b is the bit interval. The energy of a symbol can be obtained as $E_b = A^2 T_b$.

Example: Binary PAM and AWGN channel - Channel

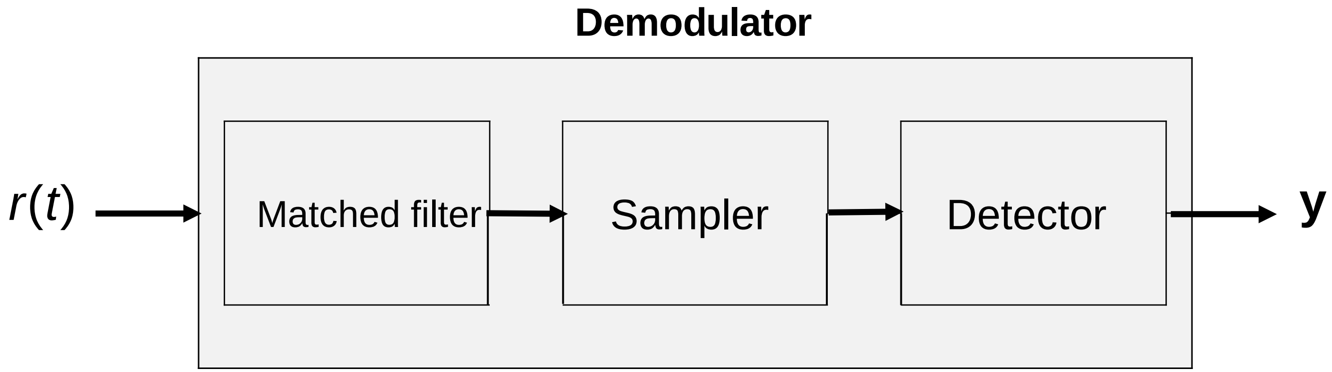
In the additive white gaussian noise (AWGN) channel, the signal at the output $r(t)$ can be expressed as the message signal at input $s_m(t)$ plus white gaussian noise, $n(t)$.



The white Gaussian noise $n(t)$ has a flat power spectrum characterized by a density $N_o/2$.

Example: Binary PAM and AWGN channel- Demodulator

The binary PAM demodulator consists of three steps, namely a matched filter, a sampler and a detector.



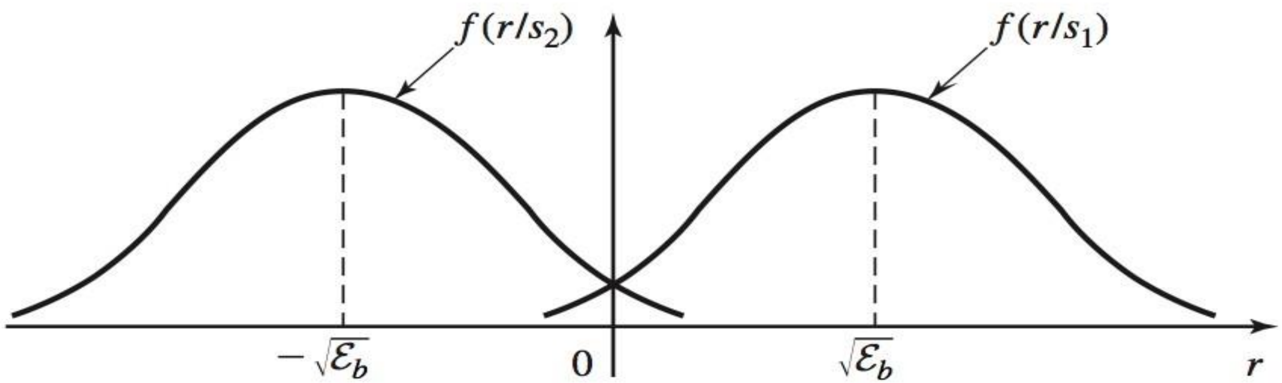
The matched filter compares the input waveforms with the expected waveform. The sampler provides a numerical value of how similar the received waveform and the expected one are. Based on the value, the detector will decide wether symbol 1 or symbol 0 was received.

Example: Binary PAM and AWGN channel - Demodulator

It can be proved that the sample values have the following probability density functions that depend on the transmitted waveform, $S_1(t)$ or $S_2(t)$, and the noise power density $N_0/2$:

$$f(r \mid s_1) = \frac{1}{\sqrt{\pi N_0}} e^{-(r-\sqrt{\mathcal{E}_b})^2/N_0}$$

$$f(r \mid s_2) = \frac{1}{\sqrt{\pi N_0}} e^{-(r+\sqrt{\mathcal{E}_b})^2/N_0}$$



Example: Binary PAM and AWGN channel - Demodulator

Based on the sample values PDF, the detector will decide symbol 1 was transmitted whenever the sample value is positive; otherwise it will decide symbol 0 was transmitted. The probability of error when $S_1(t)$ is transmitted will then be:

$$
\begin{aligned}
P(e \mid s_1) &= \int_{\infty}^{0} p(r \mid s_1)\, dr \\
&= \frac{1}{\sqrt{\pi N_0}} \int_{-\infty}^{0} e^{-(r-\sqrt{\mathcal{E}_b})^2/N_0}\, dr \\
&= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\sqrt{2\mathcal{E}_b/N_0}} e^{-x^2/2}\, dx \\
&= \frac{1}{\sqrt{2\pi}} \int_{\sqrt{2\mathcal{E}_b/N_0}}^{\infty} e^{-x^2/2}\, dx
\end{aligned}
$$

The same probability of error will be obtained when $S_2(t)$ is transmitted.

Mathematical definition of a digital channel

A digital channel is a system that establishes a relationship between two **sequences of symbols**, namely the **input sequence** $\mathbf{x} = [x_1, x_2, \ldots, x_N]$ and the **output sequence** $\mathbf{y} = [y_1, y_2, \ldots, y_N]$.

The symbols x_i in the input sequence belong to an alphabet X and the symbols at the output y_i to an alphabet Y.

Information-theory analysis of the digital channel

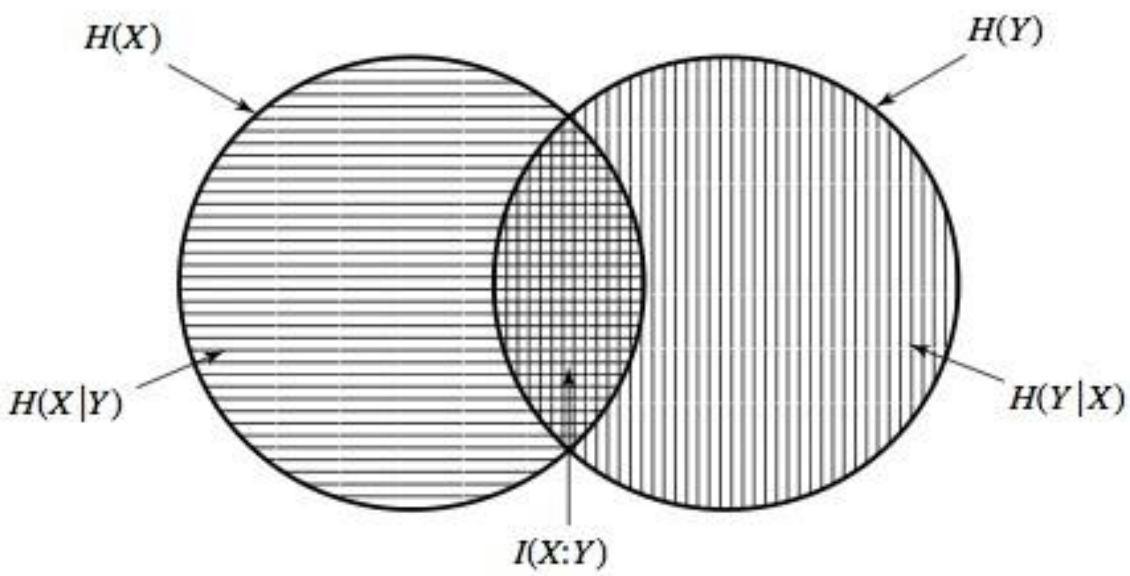
If we treat X and Y as information sources, we can define the following quantities:

- The entropies $H(X)$ and $H(Y)$: the information content of each source.
- The entropy $H(X,Y)$: the information content of both sources.
- The conditional entropies $H(X|Y)$ and $H(Y|X)$: the new information provided by one source if the other source is known.
- The mutual information $I(X;Y)$: the information shared by both sources.

These quantities can be used to describe the relationship between the input and the output of a digital channel.

Information-theory analysis of the digital channel

Information theory quantities $H(X)$, $H(Y)$, $H(X|Y)$, $H(Y|X)$ and $I(X;Y)$ can be represented as follows:



Question: Which quantity are we interested in when transmitting information?

Information-theory analysis of the digital channel

The entropy H(X) is defined as

$$H(X) = -\sum_{x} P(x) \log p(x)$$

and the conditional entropy $H(X|Y)$ is defined as

$$H(X|Y) = -\sum_{x,y} P(x,y) \log p(x|y)$$

The mutual information $I(X;Y)$ can be defined as

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X).$$

Question: It can be proved that $0 \leq I(X;Y) \leq \min(H(X), H(Y))$, can you see why?

Interlude: The medical diagnosis machine

You are a hospital manager and want to buy a diagnosis machine that tells whether a patient suffers from a certain illness or not. Three different companies offer you their machines, M_A, M_B and M_C.

Before buying any of the machines, you try them on patients whose diagnosis you know in advance and get the following percentage of correctly diagnosed patients:

- M_A: 80%.
- M_B: 50%.
- M_C: 2%.

Which machine would you buy? Why?

CHANNEL CAPACITY

Channel capacity

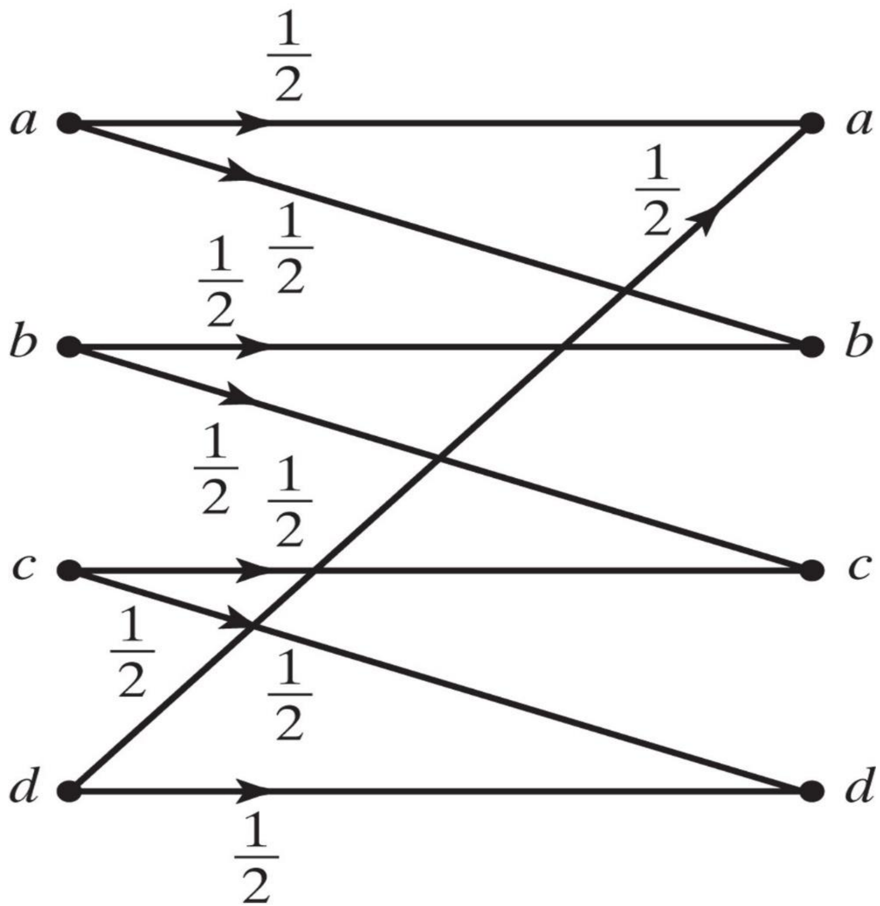
The main objective when transmitting information over a channel is **reliability**, which is measured by the probability of correct reception at the receiver.

Information theory tells us that this probability can be increased as much as we want as long as the transmission rate is less than the **channel capacity**.

Noisy channel coding theorem (Shannon 1948) says that the basic limitation that noise causes in a communication channel is not on the reliability of communication, but on the speed of communication

The channel capacity imposes a **theoretical limit** on the transmission speed. Hence, **the probability of error affects the speed of the communication**.

A Example of A Discrete Channel



❖ Four Inputs/Outputs

- What if *b* is received?
- What if *d* is received?

❖ Two Inputs/Outputs (a/c)

- What if *b* is received?
- What if *d* is received?

→ Using only those inputs whose corresponding possible outputs are disjoint, and thus do not cause ambiguity

Channel capacity of Binary Symmetric Channels

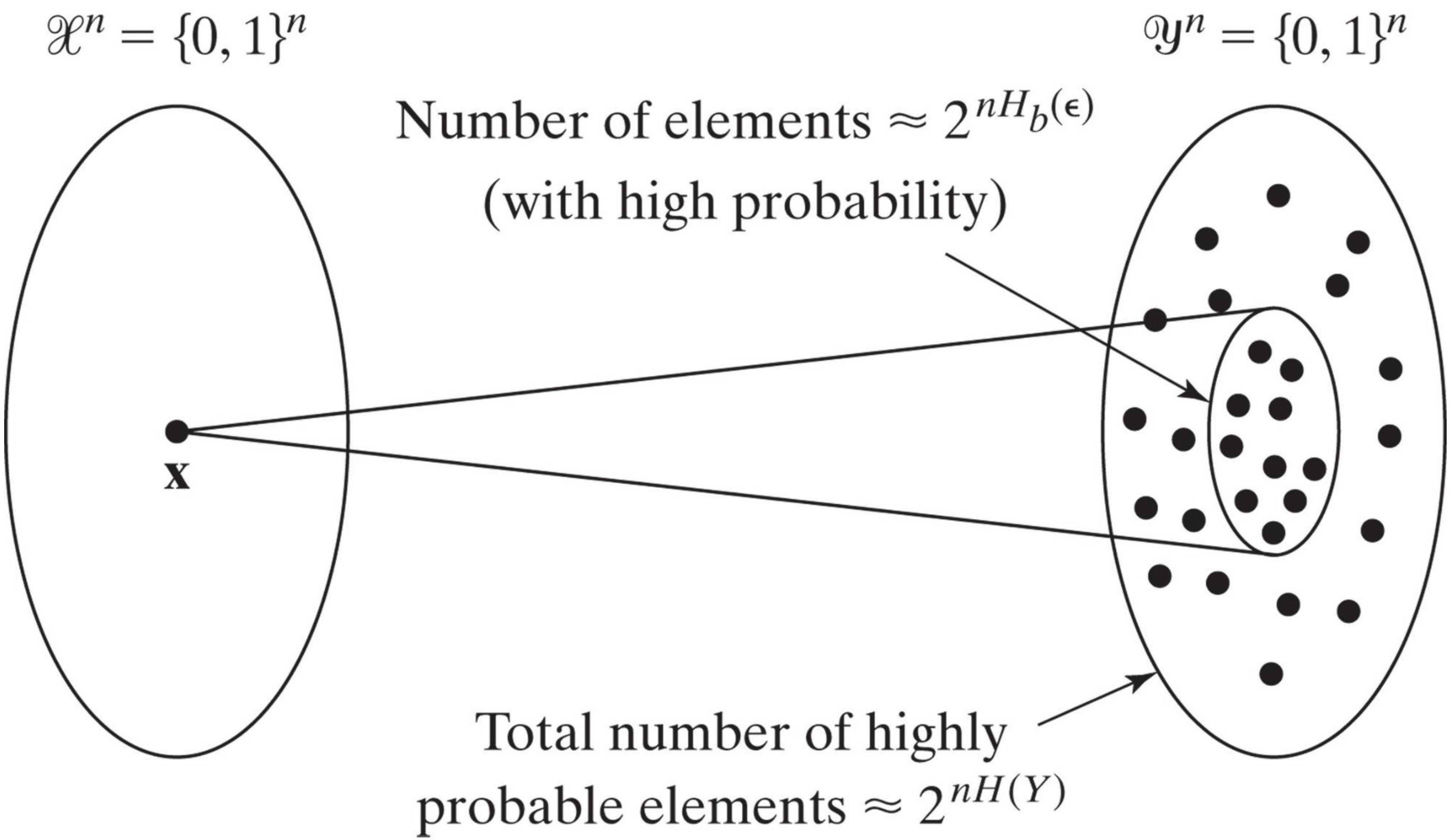
Let us consider a memoryless binary digital channel with input sequence $\mathbf{x} = [x_1, x_2, \ldots, x_N]$ and output sequence $\mathbf{y} = [y_1, y_2, \ldots, y_N]$. Let ε be the crossover (error) probability.

For N large enough, we will expect $N \times \varepsilon$ errors in the output sequence $\mathbf{y}$. Using Stirling's approximation for factorials, the number of possible output sequences $\mathbf{y}$ that disagree with $\mathbf{x}$ in $N \times \varepsilon$ positions is

$$\binom{N}{N\varepsilon} \approx 2^{NH_b(\varepsilon)}$$

where $H_b(\varepsilon) = -\varepsilon \log \varepsilon - (1-\varepsilon)\log(1-\varepsilon)$. This means that for every input sequence $\mathbf{x}$ of length N there will be approximately $2^{NH(\varepsilon)}$ different output sequences $\mathbf{y}$ of length N.

Channel capacity of Binary Symmetric Channels



Channel capacity of Binary Symmetric Channels

When we analyzed information sources, we concluded that for an information source Y with entropy $H(Y)$ there are $2^{NH(Y)}$ highly probable sequences $\mathbf{y}$ of length N.

Hence, the maximum number of input sequences that produce almost non-overlapping output sequences is at most equal to

$$M = \frac{2^{NH(Y)}}{2^{NH_b(\epsilon)}} = 2^{N(H(Y) - Hb(\epsilon))}$$

Then, in theory, **if we choose wisely M different input sequences we can always identify them without error by looking at the output sequence**.

Channel capacity of Binary Symmetric Channels

If we restrict ourselves to M different binary input sequences of length N, the transmission rate R will be:

$$R = \frac{\log M}{N} = H(Y) - Hb(\varepsilon) \text{ bits/transmission}$$

How can we increase the transmission rate? Either by reducing $H_b(\varepsilon)$ or by increasing $H(Y)$:

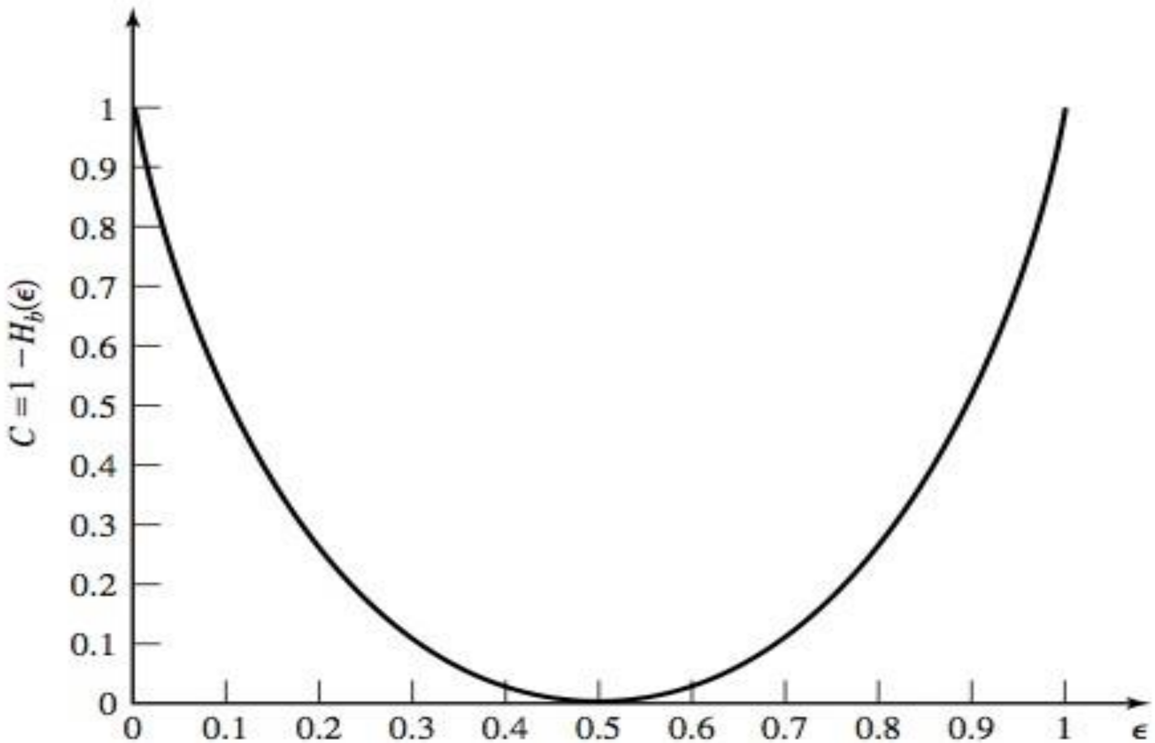
- ❖ The quantity $H_b(\varepsilon)$ cannot be controlled, since it is a property of the channel.
- ❖ The entropy $H(Y)$ can however be maximized by wisely choosing $p(x)$.

The resulting maximum transmission rate C will be:

$$C = 1 - H_b(\varepsilon) \text{ bits/transmission}$$

and it is known as the binary symmetric **channel capacity**.

Channel capacity of Binary Symmetric Channels



Question: Why is $C = 0$ when $\varepsilon = 0.5$? Why is $C = 1$ for both $\varepsilon = 0$ and $\varepsilon = 1$?

Queen Mary University of London

The Noisy Channel Coding Theorem

The capacity of a digital memoryless channel is given by

$$C = \max_{p(x)} I(X;Y)$$

$$I(X;Y) = \sum_{x,y} p(x,y) \log \left(\frac{p(x,y)}{p(x)p(y)}\right)$$

If the transmission rate R is less than the channel capacity C, there will exist a code that will result in an error probability as small as desired. If $R > C$, the error probability will be bounded away from 0.

CAPACITY OF AWGN CHANNEL

Shannon's formula

The capacity of a channel defines a limit for reliable communications. Only when the information rate is below the capacity of the channel, error-free transmission **can** be achieved (if we design our system properly!).

According to Shannon's formula, the capacity of an additive white Gaussian noise channel is

$$C_{bit/s} = Wlog(1 + SNR) = Wlog\left(1 + \frac{P}{N_0 W}\right) bit/\text{sec}$$

where W is the channel bandwidth, the noise spectral density is $N_0/2$ and P is the signal power.

Notice that the noise power is $P_N = N_0 W$, hence$SNR = \frac{P}{P_N} = \frac{P}{N_0 W}$

Transmission rate and spectral bandwidth

In digital systems, the speed of communication is measured by the **bit transmission rate** R_B (number of bits per second) or, in general, by the **symbol transmission rate** R_S (number of symbols per second).

Digital information is transmitted through a physical medium by means of analog waveforms that occupy a bandwidth W. It can be proved that **the maximum symbol transmission rate is** $2W$ **symbol/sec**. Hence, Shannon's formula can also be expressed as

$$C_{bit/sym} = \frac{C_{bit/s}}{2W} = \frac{1}{2}\log(1 + SNR)$$
$$= \frac{1}{2} log\left(1 + \frac{P}{N_0 W}\right) bits/symbol$$

Consequently, we can measure the channel's capacity both in bits/sec and bits/symbol.

Spectral efficiency

The spectral efficiency η is used to measure how efficiently the available bandwidth is used. It is defined as

$$\eta = \frac{R_B}{W} \; bps/Hz$$

where bps=bits/s.

(For instance, in GSM R_B = 104Kbps and W = 200KHz, hence η_{GSM} = 0.52bps/Hz. In LTE R_B = 81Mbps and W = 20MHz, hence η_{LTE} = 4.08bps/Hz.)

The maximum spectral efficiency η_{max} is then

$$\eta_{max} = \frac{C_{bit/s}}{W} = \log\left(1 + \frac{P}{N_0 W}\right) bps/Hz$$

Queen Mary University of London

Signal to noise ratio

In digital communication systems, the bit energy E_B is the average energy that we use to transmit one bit. Since $T_B = 1/R_B$ is the duration of one bit, the signal power P can be obtained as

$$P = E_B R_B$$

By using the bit energy, the SNR can be expressed as

$$SNR = \frac{P}{P_N} = \frac{E_B R_B}{N_0 W} = \eta \frac{E_B}{N_0}$$

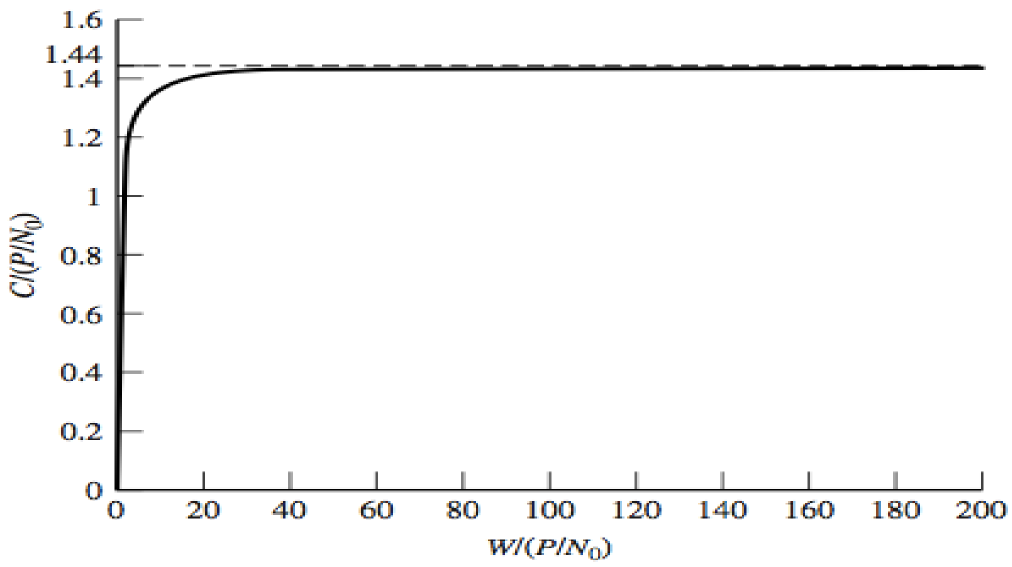
and Shannon's formula as

$$C_{bit/s} = W log(1 + \eta \frac{E_B}{N_0}) \text{ bit/sec.}$$

Bandwidth effects

With higher bandwidths the transmission rate can be increased. However, higher bandwidths also imply higher noise power. The following limit can be derived:

$$\lim_{W \to \infty} C = 1.44 \frac{P}{N_0}$$

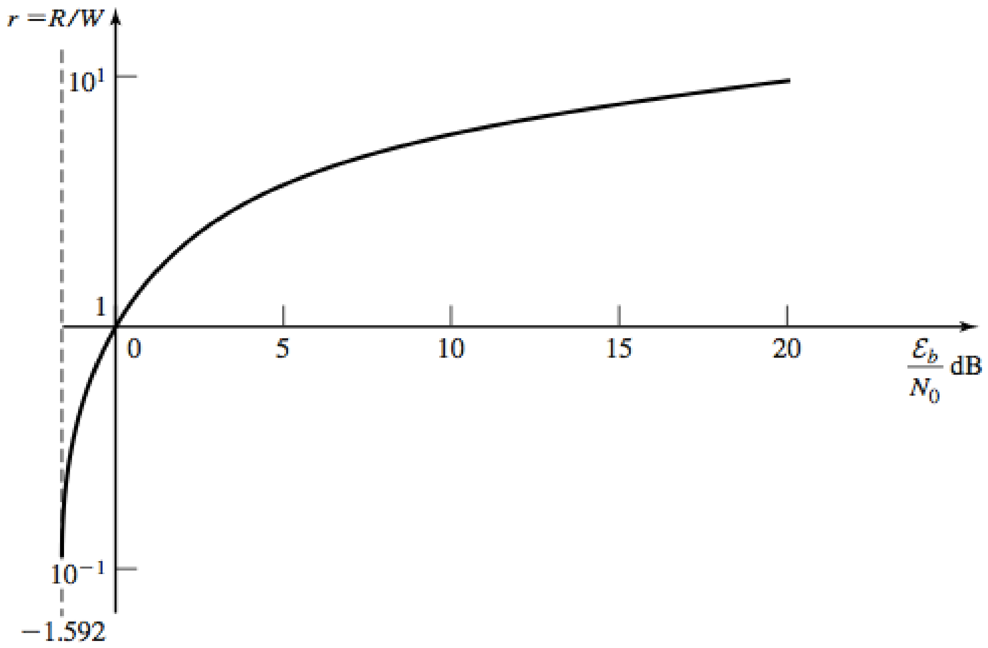


Queen Mary University of London

SNR effects

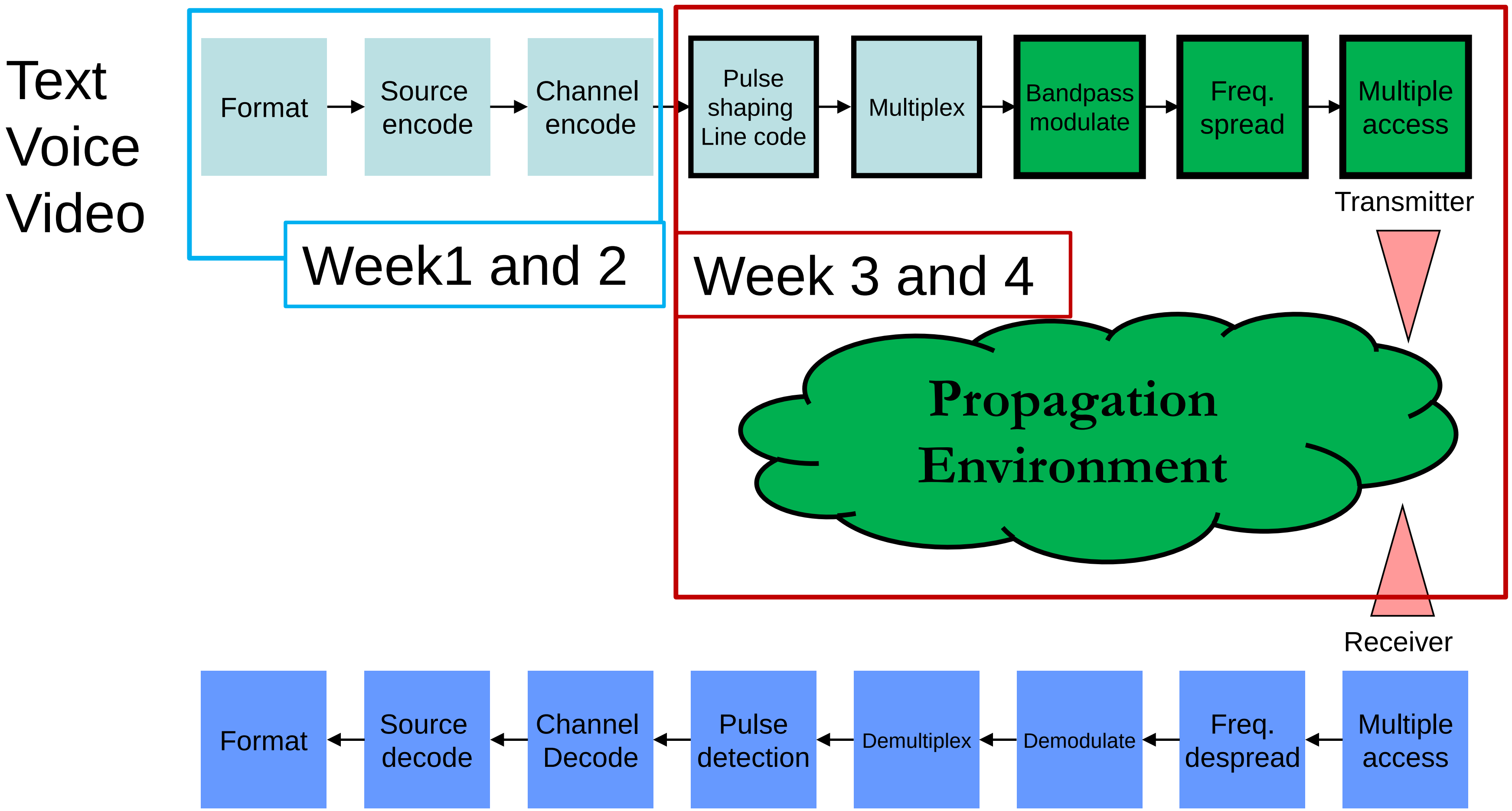
By increasing the signal to noise ratio (SNR) the maximum information rate for reliable communications increases.

$$r = \log(1 + r\frac{E_b}{N_0})$$



CHANNEL CODING

Overview of Wireless Communication System



Channel coding and redundancy

Coding is a process that produces a sequence of symbols from another sequence of symbols.

In **source coding**, given a sequence of symbols we produce a new, shorter sequence that contains the same information. Hence, **eliminate redundancy**.

By contrast, in **channel coding** our aim is to protect information against errors and for that we **introduce redundancy**, producing longer sequences of symbols.

Are we undoing in channel coding what we did in source coding? **Not really**, in the former we eliminate unnecessary redundancy and in the latter we introduce suitable redundancy!

A simple example of channel coding

Let us define the following code:

- $0 \to 00$.
- $1 \to 11$.

By using this code, we are sending every bit of our message twice. Hence, we would expect to see sequences of 00 and 11 but not 10 nor 01 since **they are not codewords**. Unless, of course, there is one error.

In the sequence $00 - 11 - 00 - 11 - 11 - \mathbf{10} - 00 - 00$ we have been able to detect one error, but we do not know whether the corrupted two bit sequence was 11 or 00.

Another simple example of channel coding

Let us define the following code:

- $0 \rightarrow 000$.
- $1 \rightarrow 111$.

Now we are sending every bit of our message three times. If we detect a three bit sequence that is not 000 nor 111 we will know there has been an error.

In the sequence 000 − 111 − 000 − 111 − 111 − **101** − 000 − 000 we have been able to detect one error. In this case, we will correct 101 to 111 and not to 000, because it is more probable to observe one error than two errors.

Coding rate

From now on, we will assume that we are dealing with binary sequences. Let k be the length of the original binary sequence and n the length of the sequence after coding. Hence, we are introducing $m = n - k$ redundancy bits.

We define the code rate R_C as

$$R_C = \frac{k}{n}$$

As we can see, only 2^k binary sequences out of 2^n binary sequences are valid code words! If we receive a sequence that is not a code word, it means that errors have occurred during transmission.

Question: What was the coding rate in the previous two examples?

Effects of coding on the bandwidth and the bit rate

If our code rate is $R_C = k/n$, for every k bits of our message, we will be transmitting n bits. Hence:

- If our bandwidth is fixed, then the information rate will decrease by R_C. In other words, we transmit fewer bits of our message per second.
- If our information rate is fixed, then the transmission rate will increase by $1/R_C$ and so will the necessary bandwidth. In other words, we need more bandwidth to accommodate more bits in the same time interval.

Error detection and correction

We have seen that the purpose of channel coding is to protect our information again errors. Assume that we are using a code rate R_C.

- **Forward error correction** (FEC) protects our message against up to N_C errors. It is convenient in those cases where we do not have a feedback link.
- If we detect errors, what should we do? We can ask for the message to be retransmitted. **Automatic repeat request** (ARQ) consists of asking the sender to retransmit the message.
- For the same code rate R_C, $N_D > N_C$.

Types of channel codes

Here exist two main families of channel coding techniques:

- **Block codes**. In a block code, an information sequence is broken into blocks of length k and each block is mapped into channel inputs of length n. Each block is independent from any other block.
- **Convolutional codes**. In a convolutional code, k bits of the information sequence enter a $k \times L$ shift registry. The bits are linearly combined to produce n bits. Hence, each n-bit output depends on the previos $k \times (L-1)$ bits. In other words, convolutional codes have memory.

In this lecture, we will restrict ourselves to block codes.

Definition of linear block codes

An (n, k) block code $C = \{\mathbf{c}_1, \mathbf{c}_2 \ldots, \mathbf{c}_M\}$ is defined by a collection of $M = 2^k$ binary sequences of length n called *code words*. Instead of sending the original block of k bits, we send a code word.

- **Definition**: A block code is said to be **linear** if any linear combination of code words is also a code word. In the binary case, linear combinations are defined as the component-wise modulo 2 addition (i.e. 0+0=0, 1+1=0, 1+0=1, 0+1=1).

One consequence is that the zero sequence $\mathbf{0}$ is always a code word of any linear block code.

Linear Block Code or Not?

Coding 2 bits information into 5 bits codewords:

00 → 00000,00001,00010,00011,00100,00101,00110,00111
01 → 01000,01001,01010,01011,01100,01101,01110,01111
10 → 10000,10001,10010,10011,10100,10101,10110,10111
11 → 11000,11001,11010,11011,11100,11101,11110,11111

The (5,2) code defined by the following mapping is linear or non-linear?

$$00 \rightarrow 00000$$
$$01 \rightarrow 01111$$
$$10 \rightarrow 10100$$
$$11 \rightarrow 11011$$

Hamming distance and Hamming weight

Four more definitions:

- **Definition**: The *Hamming distance* between two code words $\mathbf{c}_i$ and $\mathbf{c}_j$, $d(\mathbf{c}_i, \mathbf{c}_j)$ is the number of components at which they differ.
- **Definition**: The minimum distance of a code d_{min} s the minimum Hamming distance between any two code words.
- **Definition**: The *Hamming weight*, or the *weight* of a code word $\mathbf{c}_i$, $w(\mathbf{c}_i)$is the number of 1's in the code word.
- **Definition**: The minimum weight of a code is the minimum of the weights of the code words except the all-zero code word.

In the previous example, the Hamming distance between code words 01111 and 10100 is $d(01111, 10100) = 4$. However, the minimum distance of the code is $d_{min} = 2$.

Hamming distance

How can we use the Hamming distance for decoding?

- Given an output sequence y, we can obtain the distance between this sequence and all of the code words, $d(y, c_i)$.
- We will decode y as the sequence $c_{\min}$ whose distance to y is shortest, i.e. as the most similar sequence.

In terms of error correction, a good code book will then be one such that its minimum distance $d_{\min}$ is high. Why? Because the maximum number of errors that we are able to correct by this procedure, N_C, is related to $d_{\min}$ by $d_{\min} \geq 2N_C + 1$. The number of errors that can be detect, N_D, is by contrast related to $d_{\min}$ by $d_{\min} \geq N_D + 1$.

Generator matrix

Given an (n, k) linear block code, any information sequence $\mathbf{x}$ can be mapped into its code word $\mathbf{c}$ by multiplying it by the generator matrix $\mathbf{G}$

$$\boldsymbol{G} \stackrel{\text{def}}{=} \begin{bmatrix} \boldsymbol{g_1} \\ \boldsymbol{g_2} \\ \vdots \\ \boldsymbol{g_k} \end{bmatrix} = \begin{bmatrix} g_{11} & g_{11} & \cdots & g_{1n} \\ g_{21} & g_{22} & & g_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ g_{k1} & g_{k2} & \cdots & g_{kn} \end{bmatrix}$$

so that $\mathbf{c} = \mathbf{xG}$. It is easy to see that the code word for the sequence $10\ldots0$ is $\mathbf{g}_1$, for $010\ldots0$ is $\mathbf{g}_2$ and so on.

The generator matrix in our previous example is

$$G = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Systematic codes

In a systematic code, the code word consists of the information sequence followed by a sequence of $m = n - k$ bits, known as the parity bits. Hence, the generator matrix has the form

$$\mathbf{G} = [\,\mathbf{I}_k \mid \mathbf{P}\,]$$

where $\mathbf{I}_k$ is the $k \times k$ identity matrix and $\mathbf{P}$ is the parity matrix.

Parity check matrix

The parity check matrix **H** allows us to check whether a code word belongs to our code or not. It has the property that

$$cH^t = 0$$

If the code is systematic, the parity check matrix can be obtained as

$$\mathbf{H} = [\ \mathbf{P}^t \mid \mathbf{I}_{n-k}\ ]$$

where t denotes transposition. In the binary case, $-\mathbf{P}^t = \mathbf{P}^t$. Hence, parity check matrices allow us to **detect errors** by determining whether a given received sequence is a code word or not.

Parity check matrix

In our example:

$$G = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\mathbf{H} = \left[\begin{array}{c|c} 11 & 100 \\ 01 & 010 \\ 01 & 001 \end{array}\right]$$

Principles of block decoding

Decoding consists of recovering the initial information sequence based on the observation of a (probably) corrupted sequence of code words. During decoding we essentially **compare each received sequence with all the code words** defining the code and then **choose the most similar code word**.

The question is, how do we define similarity between binary sequences? Our definition of distance is given by the Hamming distance. The notion of similarity will also allow us to determine the maximum number of errors that we can correct.

Block decoding algorithm: syndrome decoding

Let us denote by **e** the error binary sequence. The output sequence y that we obtain when code word c is transmitted can be expressed as

$$y = c + e.$$

If there are no errors during transmission, e = 0, if there is an error in the first bit, e = (10 . . . 0), if there is an error in the first and third bite = (1010 . . . 0), and so on.

If we apply the parity check to y, we get:

$$y\boldsymbol{H^t} = c\boldsymbol{H^t} + e\boldsymbol{H^t} = e\boldsymbol{H^t}$$

Notice that the result of this operation **depends on the error sequence e and not on the code word c that we have transmitted.**

Block decoding algorithm: syndrome decoding

The sequence $\mathbf{s} = \mathbf{e}\boldsymbol{H}^{\boldsymbol{t}}$ is called the syndrome. How many different syndrome sequences are there? For a *(n, k)* code, $\boldsymbol{H}^{\boldsymbol{t}}$ has n rows and m = n − k columns. Hence, s is a $\mathbf{1} \times \mathbf{m}$ vector and there will exist $\mathbf{2}^{\boldsymbol{m}}$different syndrome sequences **s**.

If we can relate an error sequence **e** to one syndrome sequence ***s***, we can determine **e** based on the calculation $\mathbf{s} = \mathbf{y}\boldsymbol{H}^{\boldsymbol{t}}$ and the transmitted code word will be obtained as **c = y + e.**

Hence, we need as many syndrome sequences **s** as error sequences **e** we want to identify. For instance, if we want to be able to correct one single error in our sequence we need

$$\mathbf{2}^{\boldsymbol{m}} \geq n + 1.$$