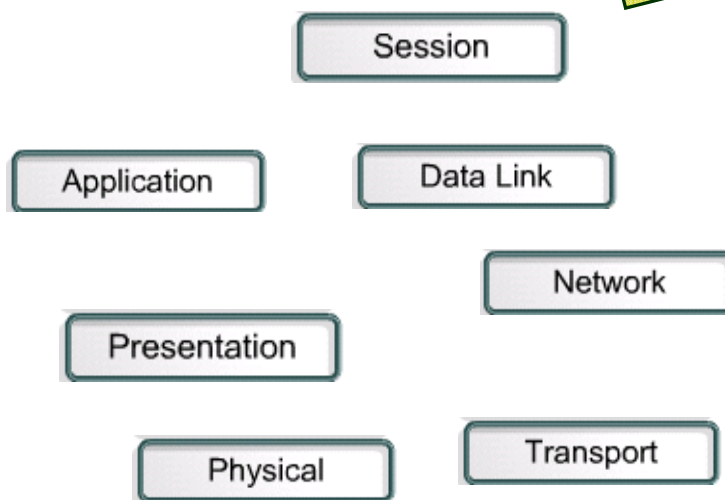


# Datacommunication I

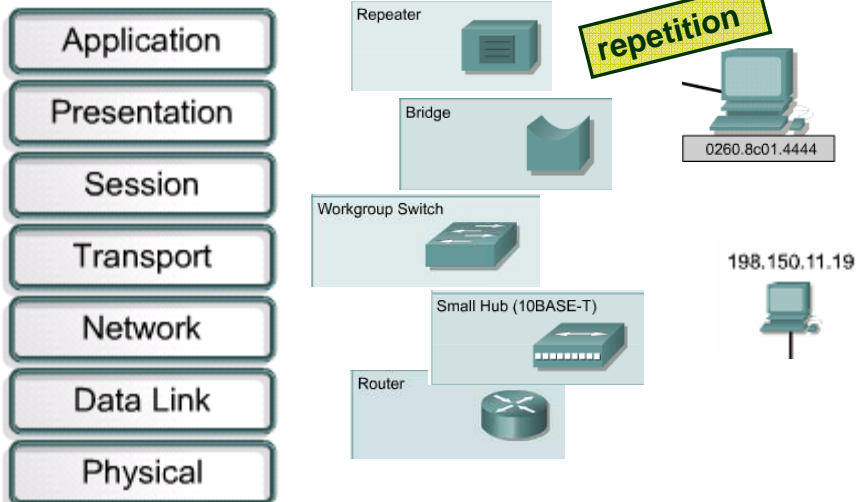
Lecture 3 –  
signal encoding, error detection/correction

## Layers of the OSI-model

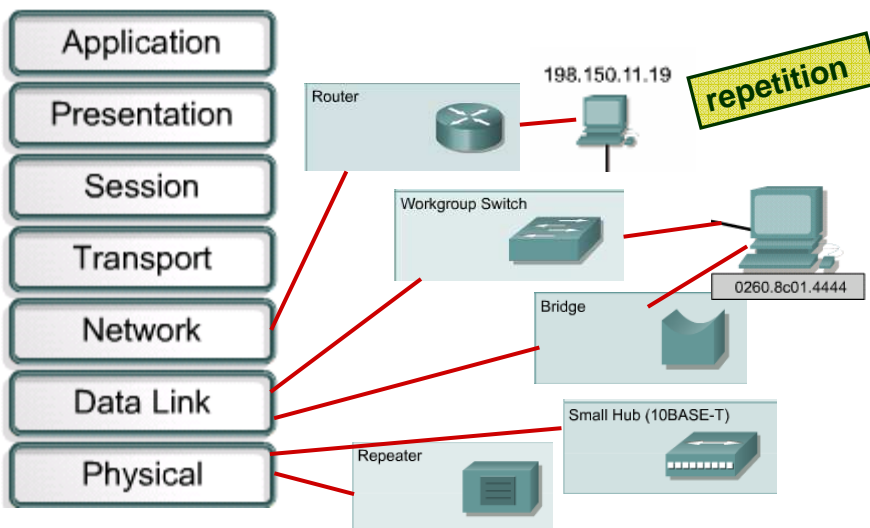
repetition



## The OSI-model and its networking devices



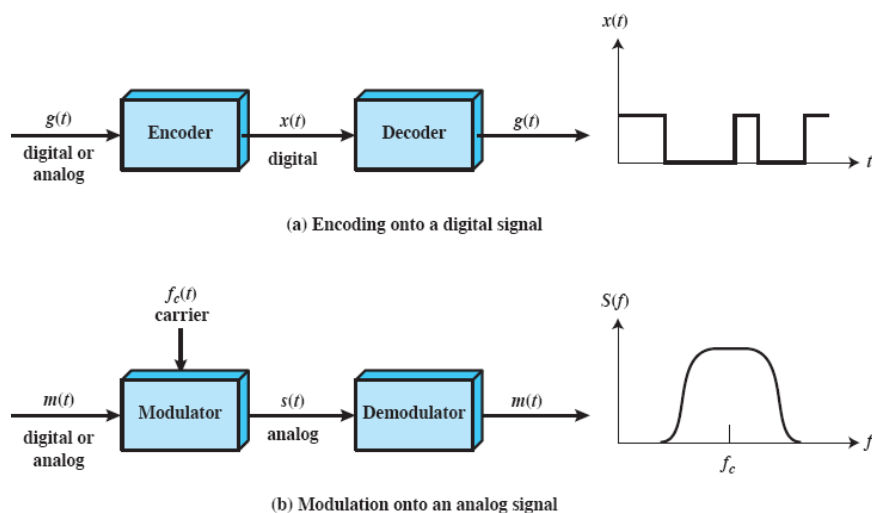
## The OSI-model and its networking devices



## Data encoding

- Turning data (digital or analog) into signals (digital or analog)
  - Encoders and decoders ("codec") are used for digital signals
  - Modulators and demodulators ("modem") are used for analog signals
- Why change between analog and digital?
  - Copper wires e.g. can handle both digital and analog signals. But cheap and fast networking equipment (e.g. switching technology) only works on digital signals
  - Unguided media and optical fibers only operate with analog signals
- In this course: focus on digital data as input

## Data encoding



## Repetition of some terms

- Simplest form:
  - One data element <-> one signal element
- Data rate (measured in bits/s)
- Signal rate
  - measured in signal elements per second or "baud"
  - Also called modulation rate or baud rate
- Bit error rate (BER)
  - the number of erroneous bits received divided by the total number of bits transmitted
- Signal to Noise Ratio (SNR)
  - Ratio of a signal power to the noise power corrupting the signal

## Bit rate vs. Signal rate

Term	Units	Definition
Data element	Bits	A single binary one or zero
Data rate	Bits per second (bps)	The rate at which data elements are transmitted
Signal element	Digital: a voltage pulse of constant amplitude Analog: a pulse of constant frequency, phase, and amplitude	That part of a signal that occupies the shortest interval of a signaling code
Signaling rate or modulation rate	Signal elements per second (baud)	The rate at which signal elements are transmitted

**Some general truths about data rate etc.**

An increase in data rate \_\_\_\_\_ the BER

An increase in SNR \_\_\_\_\_ the BER

An increase in bandwidth allows an \_\_\_\_\_ in data rate

*Increase????*

*Decrease?????*

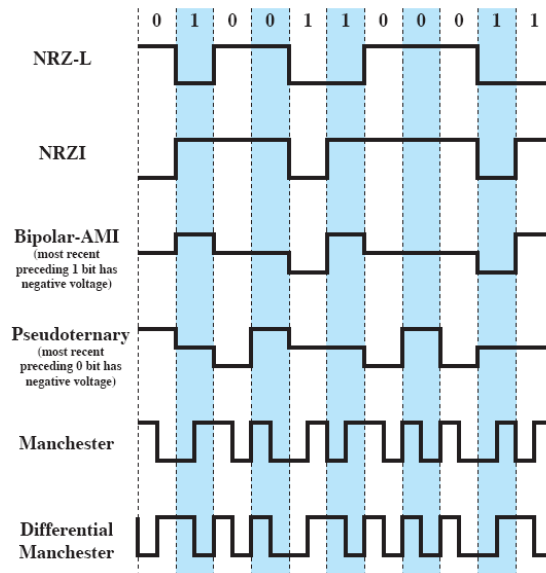
**Some general truths about data rate etc.**

An increase in data rate increases the BER

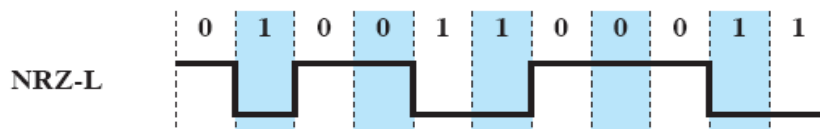
An increase in SNR decreases the BER

An increase in bandwidth allows an increase in data rate

### Digital signal encoding schemes

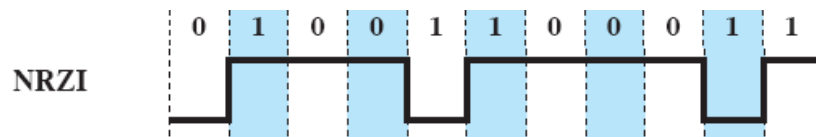


### NRZ(-L) – Nonreturn to Zero (-Level)



- Different types:
  - E.g. no voltage = 0, +voltage = 1
  - NRZ-L: -voltage = 1, +voltage = 0
- Data rate = signal rate
- Constant value during the whole bit duration
- Problems
  - Lack of synchronization possibilities

## NRZI – Nonreturn to Zero, invert on ones



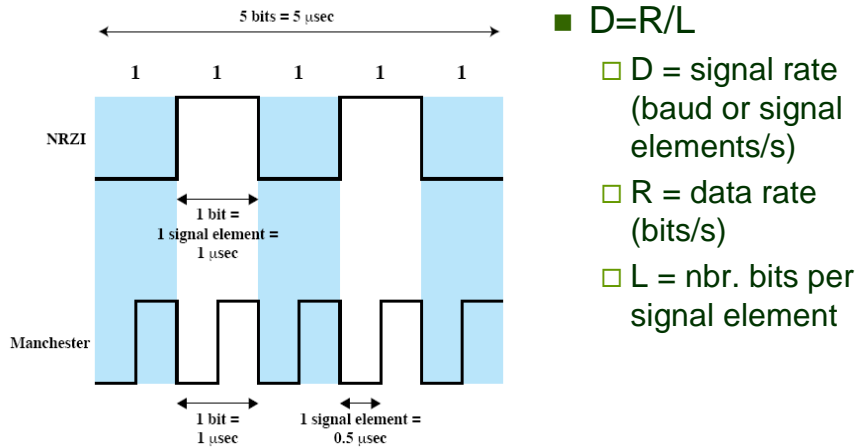
- Bits are encoded not through the voltage level itself but through the existence or absence of transitions between the levels ("Differential encoding")
  - Transition = 1
  - No transition = 0
- Easier to detect transition than a signal level (especially in the presence of noise)

## Manchester encoding ("Biphase encoding")

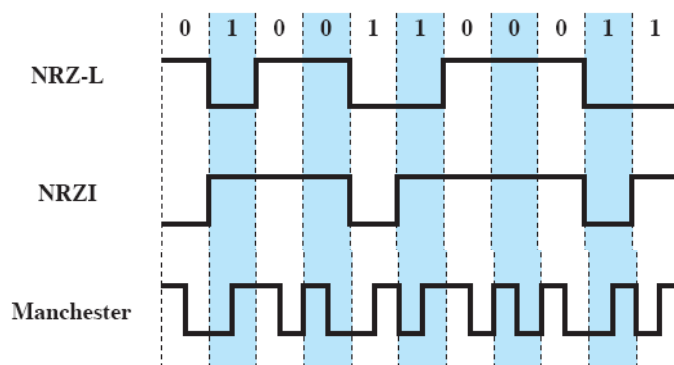


- Transition in the middle of the bit
  - High to low for 0
  - Low to high for 1
- Offers a chance to clock synchronize at every bit ("self-clocking code")
- Offers "error detection"
  - No transition during a bit -> error
- Signal rate = 2 x data rate
- Popular (used e.g. in Ethernet standard)

## More about the signal rate



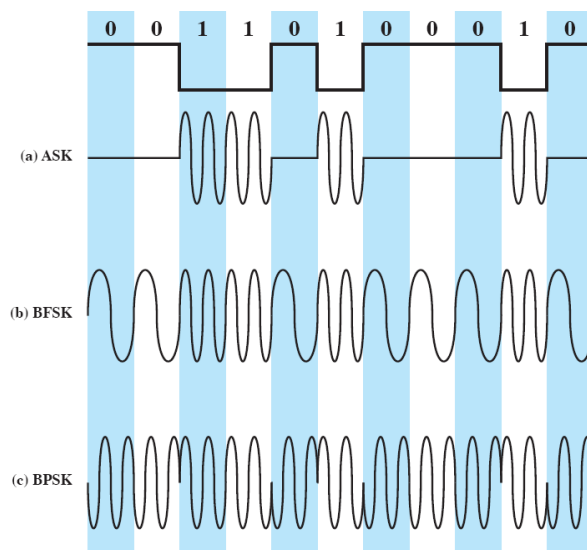
## Signal transition rate



## Signal transition rate

	min	101010...	max
<b>NRZ-L</b>	0 (all 0 or 1)	1	1
<b>NRZI</b>	0 (all 0)	0,5	1 (all 1)
<b>Manchester</b>	1	1	2 (all 0 or 1)

## Digital data, analog signals



## Amplitude Shift Keying (ASK)

- E.g. used for optical fibers
- No amplitude = 0
- Amplitude = 1

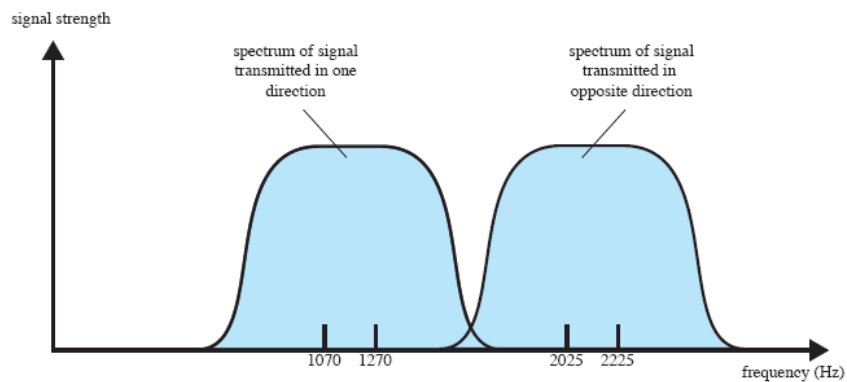
$$s(t) = \begin{cases} A \cos(2\pi f t) \\ 0 \end{cases}$$

## Binary Frequency Shift Keying (BPSK)

- Used frequently
- 2 different frequencies used to encode 0 and 1
- Less susceptible to error than ASK
- Full duplex possible
  - Use separated frequency bands, one for each transmission direction

$$s(t) = \begin{cases} A \cos(2\pi f_1 t) \\ A \cos(2\pi f_2 t) \end{cases}$$

## Full duplex in BFSK



## Phase Shift Keying (PSK)

### ■ Binary Phase Shift Keying (BPSK)

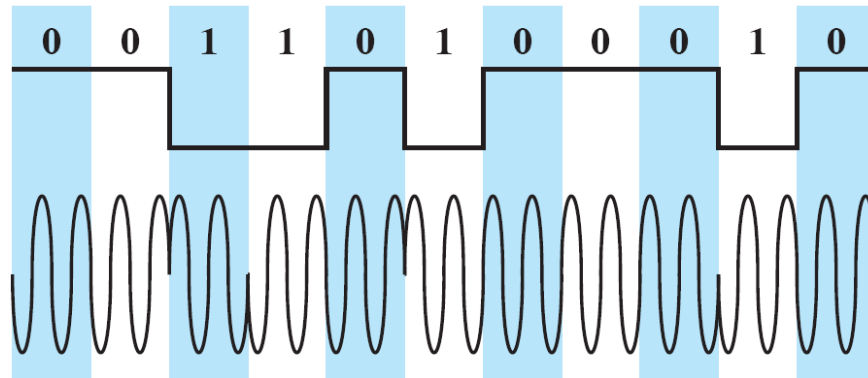
- A phase denotes 1, another phase denotes 0

### ■ Differential PSK (DPSK)

- A transition denotes 1
- No transition denotes 0

$$s(t) = \begin{cases} A \cos(2\pi ft) \\ A \cos(2\pi ft + \pi) \end{cases}$$

## Example DPSK



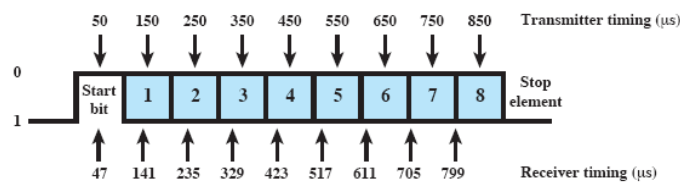
## Quadrature Phase Shift Keying (QPSK)

- 4 different phases used
- Each phase denotes 2 bits

$$s(t) = \begin{cases} A \cos(2\pi ft + \frac{\pi}{4}) \longrightarrow 11 \\ A \cos(2\pi ft + \frac{3\pi}{4}) \longrightarrow 01 \\ A \cos(2\pi ft - \frac{3\pi}{4}) \longrightarrow 00 \\ A \cos(2\pi ft - \frac{\pi}{4}) \longrightarrow 10 \end{cases}$$

## Synchronization

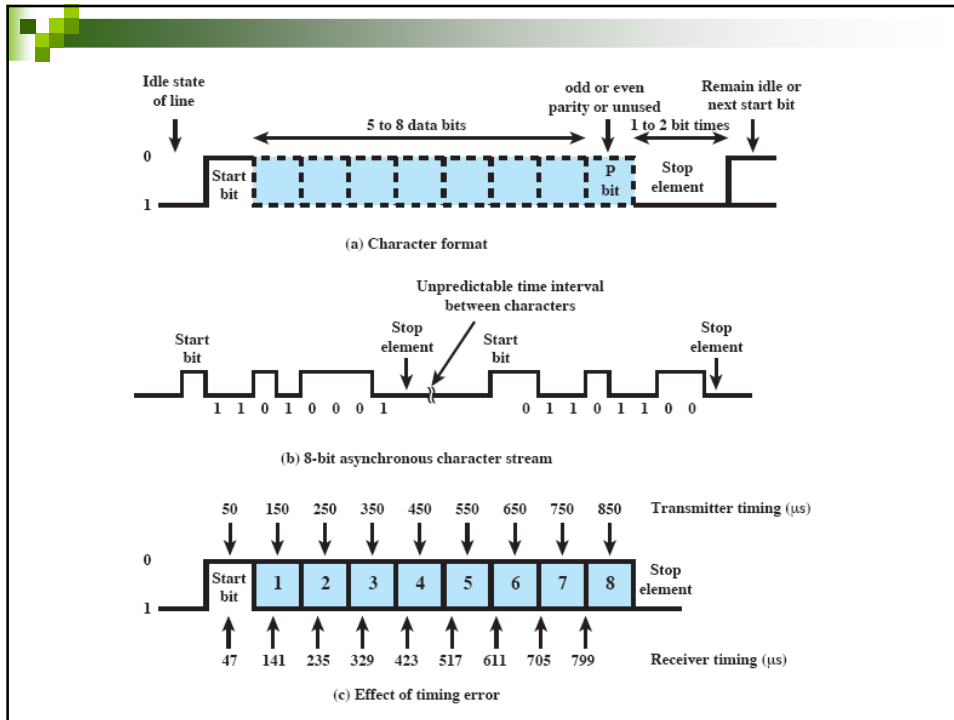
- Needed between transmitter and receiver
  - Receiver needs to "read" the incoming stream once a signal
  - Clock drifting leads to synchronization errors
- Two types of transmission
  - Asynchronous transmission
  - Synchronous transmission



(c) Effect of timing error

## Asynchronous transmission

- New definitions
  - "character" = block of a couple of bits (5-8)
  - "idle" = a state where no data is sent (or ready to be sent and the equipment is in a waiting state)
- Synchronization done before every character
  - Start bit at the beginning of each character
  - Stop element ends a character
  - Possible idle state between character transmissions
- Disadvantages
  - Simple and cheap but much overhead



## Synchronous transmission

- Blocks of bits are sent in a steady stream
  - No start and stop bits
- Synchronization through
  - self-clocking code (e.g. Manchester)
  - Separate clock line between transmitter and receiver
- Data "frame"
  - Preamble to let the receiver know that a frame start
  - Block of data
- Far less overhead than asynchronous transnm.



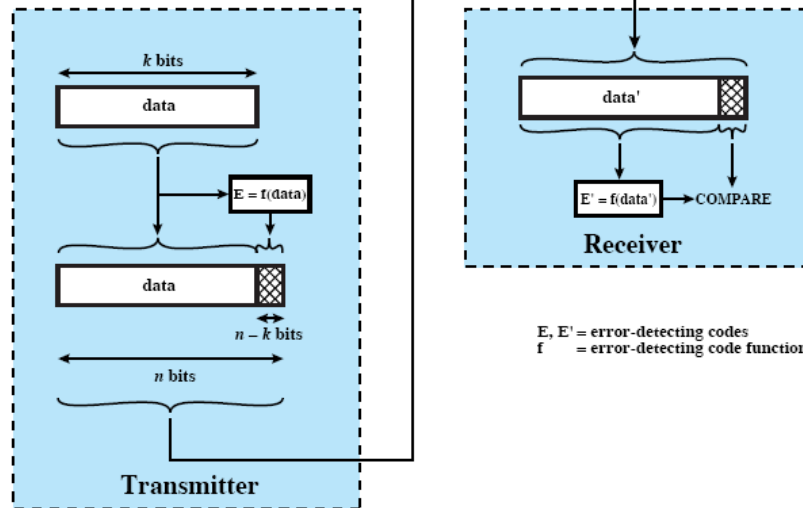
## Error detection and correction

- Bit errors occur as
  - Single bit errors
  - Error bursts
    - Large number of bits affected
    - Due to impulse noise
- Bit errors can be detected by error detecting code or detected and corrected by error correcting code

## Error detection

- A function is applied to a block of data
- The result is sent together with the data
- The receiver applies the same function to the data and compares the results
- Transmitter is informed about the erroneous data transmission (and asked for retransmission)
- Receiver cannot correct the errors
- Examples:
  - Parity check
  - Cyclic Redundancy Check (CRC)

## Error detection



## Parity Check

- Add a parity bit to the data block
- Two types
  - Even parity
    - Create an even number of "ones" by adding a parity bit
  - Odd parity
    - Create an odd number of "ones" by adding a parity bit

Even parity:	100100011110 <b>0</b>
	110111010010 <b>1</b>
Odd parity:	100100011110 <b>1</b>
	110111010010 <b>0</b>

## Cyclic Redundancy Check (CRC)

- Commonly used
- Able to detect most errors
- Transmitter creates a frame check sequence (FCS) and sends it along with the data
  - The FCS gives no remainder when the data block is divided by it
  - Gives the receiver a chance to check for errors
- How to create a suitable FCS?
  - Several methods
  - E.g. through XOR with a predetermined bit pattern

## CRC example

Data block: 1101001110100

Bit pattern: 1011

1101001110100

1011 **xor**

0110001110100

\_1011

0011101110100

\_\_1011

...

0000000001110

\_\_\_\_\_1011

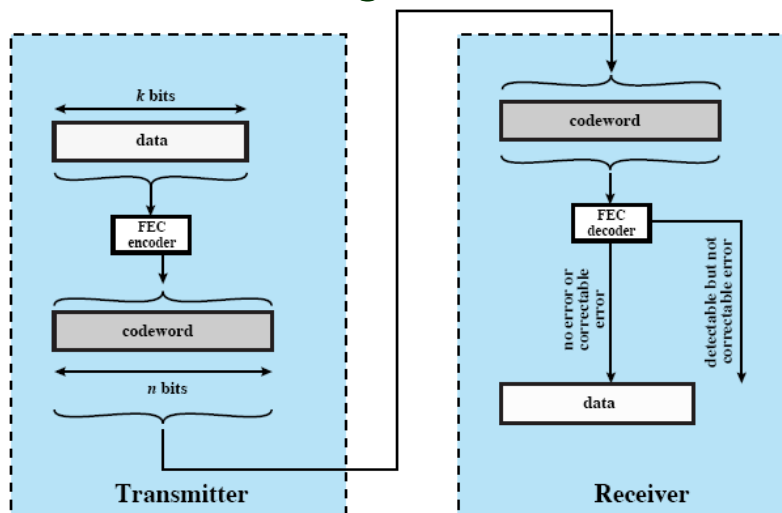
0000000000**101**

**FCS: 101**

## Error correcting codes

- Receiver can detect and correct the error by itself
  - No retransmission required
  - Saves resources
- Codeword is created out of data block
- Receiver decodes the codeword and sees correctable and non-correctable errors
  - Some errors can be corrected using the codeword
  - Some errors are not detected or only detected without being corrected
- Always a compromise between error correction abilities and limiting overhead
  - "Best error correcting code" = sending the whole block of data twice, or multiple times

## Error correcting codes



## Block codes

### ■ Definition:

- Hamming distance  $d$  = number of bits in which two binary sequences disagree
  - Example:
    - S1: 001101
    - S2: 110000
    - Hamming distance  $d = 5$
- Code rate =  $k/n$  (number of bits in the original data / number of bits in the codeword)
  - The code rate is a measurement for how much extra resources (bandwidth) that a certain code requires

## Block codes

$k$  = number of bits in the original data block  
 $n$  = number of bits in the codeword

### Example:

$k=2$ ,  $n=5$

Data block	codeword	received
00	00000	← Most probable 00100
01	00111	
10	11001	
11	11110	

- Receive wrong (non-existing) codeword → must be error
- Chose codeword with least Hamming distance to recreate correct data
- Never 100% sure the right codeword is chosen, but you chose the codeword with the highest probability to correspond to the correct data block

## Key terms

- Encoder-decoder
- Modulator-demodulator
- Signal rate (baud rate, modulation rate)
- NRZ encoding
- Manchester encoding
- Differential coding
- Amplitude Shift Keying (ASK)
- Binary Frequency Shift Keying (BFSK)
- Binary Phase Shift Keying (BPSK)
- Differential Phase Shift Keying (DPSK)
- Quadrature Phase Shift Keying (QPSK)
- Asynchronous transmission
- Synchronous transmission
- Error correction vs detection
- Clock synchronization
- Idle state
- Self-clocking code
- Parity check
- Odd vs even parity
- Block coding
- Codeword
- Hamming distance
- Code rate