

Prof. Dr.-Ing. Dr. h.c. Jürgen Becker (*becker@kit.edu*)

Dr.-Ing. Jens Becker (*jens.becker@kit.edu*)

Institut für Technik der Informationsverarbeitung (ITIV)

Communication Systems and Protocols

Session 4: Physical Principles

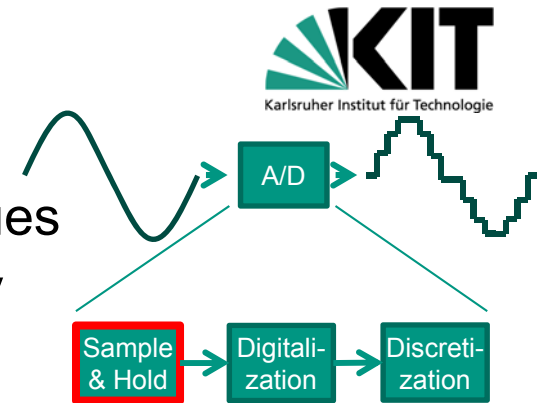
Clicker Session: Recapitulation

■ <https://arsnova.eu/mobile/#id/77482045>



Recapitulation: Sampling of Signals

- Transforming a continuous signal into discrete values
 - Signal shall be reconstructable unambiguously
- Prerequisites:
 - Signal limited in frequency with a maximum frequency f_{max}
 - Equidistant sampling points



■ Nyquist-Shannon-Sampling Theorem

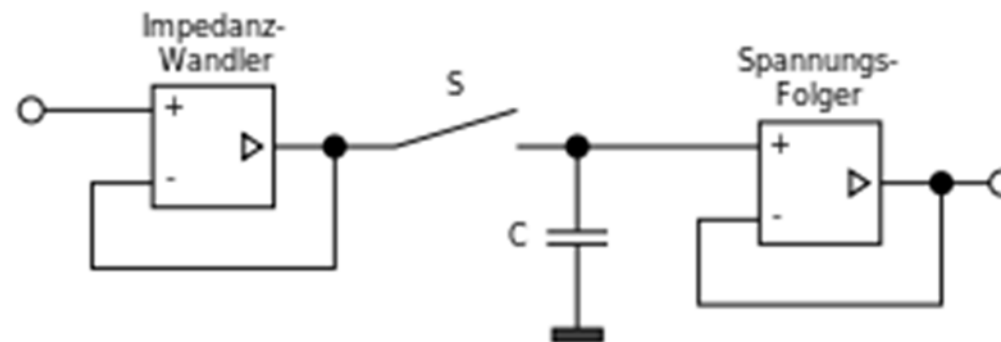
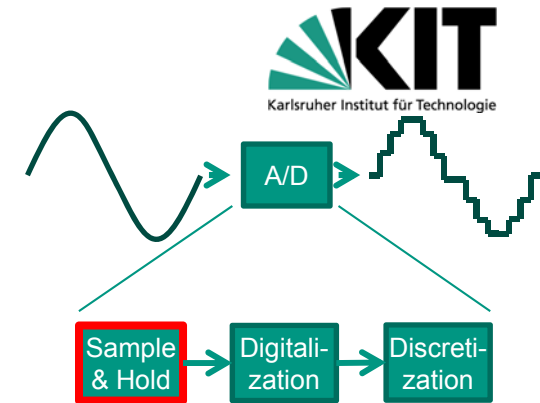
A continuous signal has to be sampled with a frequency at least double the maximum frequency of the signal itself f_{max} in order to be able to unambiguously reconstruct the original signal.

$$f_{Sample} \geq 2 \cdot f_{max}$$

To keep this condition it may be necessary to filter the signal prior to sampling with a low-pass filter. This prevents artefacts otherwise caused by high frequencies.

Sample & Hold

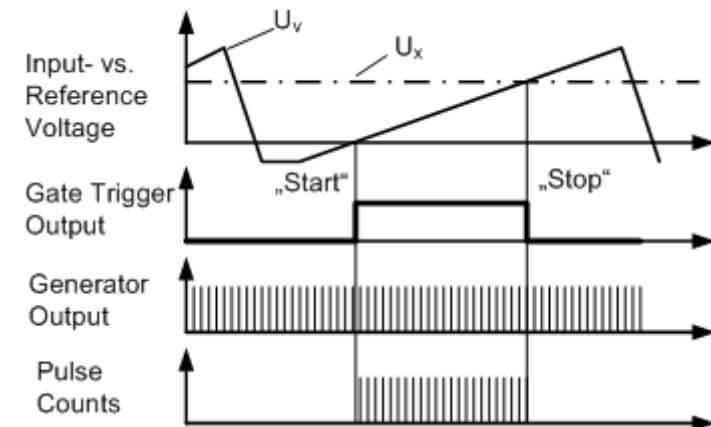
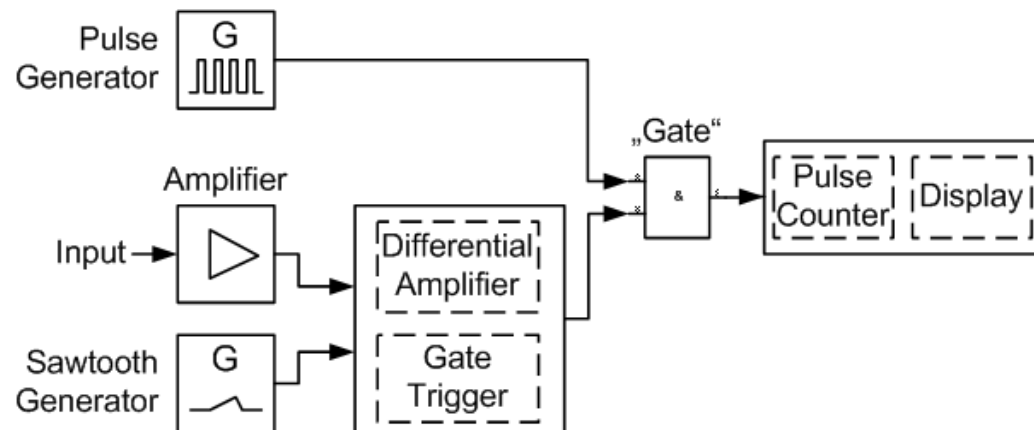
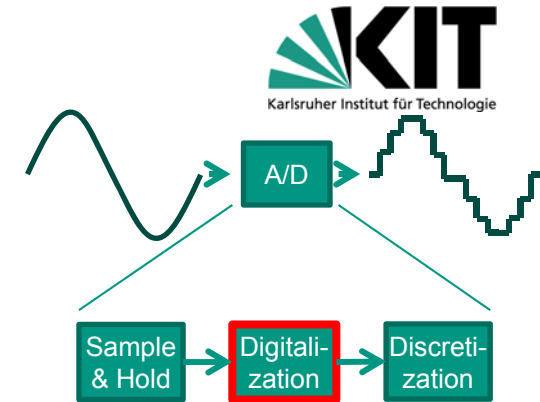
- Analogue signals change over time
- Analogue signals have to be kept constant for duration of analog-digital conversion



Digitalization

■ Single-Slope ADC (counter-ramp)

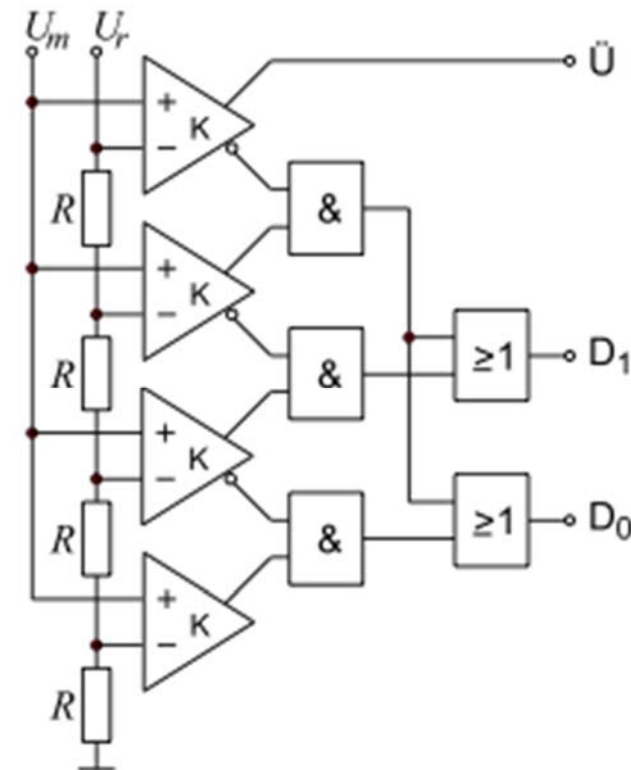
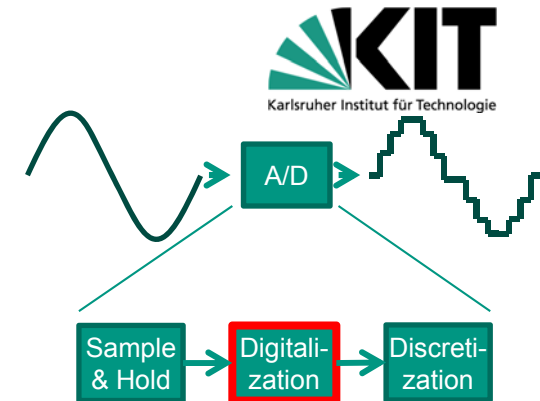
- Sawtooth Signal rises slowly from GND to U_e .
- Counter is incremented until U_e is reached
- Counter value is proportional to U_e
- At the end of the sampling phase Counter value is stored into a register



→ More information see lecture “Elektronische Schaltungen”

Flash Converter

- Direct conversion of analogue input signal into digital signal
- Here: 2-bit full parallel Converter with binary encoding
- Comparison of signal with reference voltage(s)
 - resistor cascade
 - Comparator “Cascade”
- Digital Glue Logic for generation of binary value

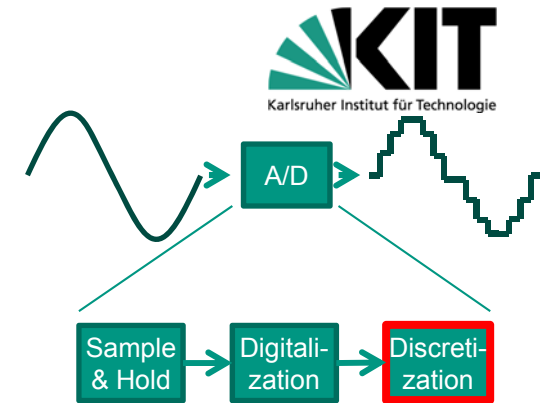


Source: <http://de.wikipedia.org>

Discretization

In theory:

- After the first two steps, the signal can still be reconstructed perfectly
 - Signal value is sampled with unlimited precision



In reality:

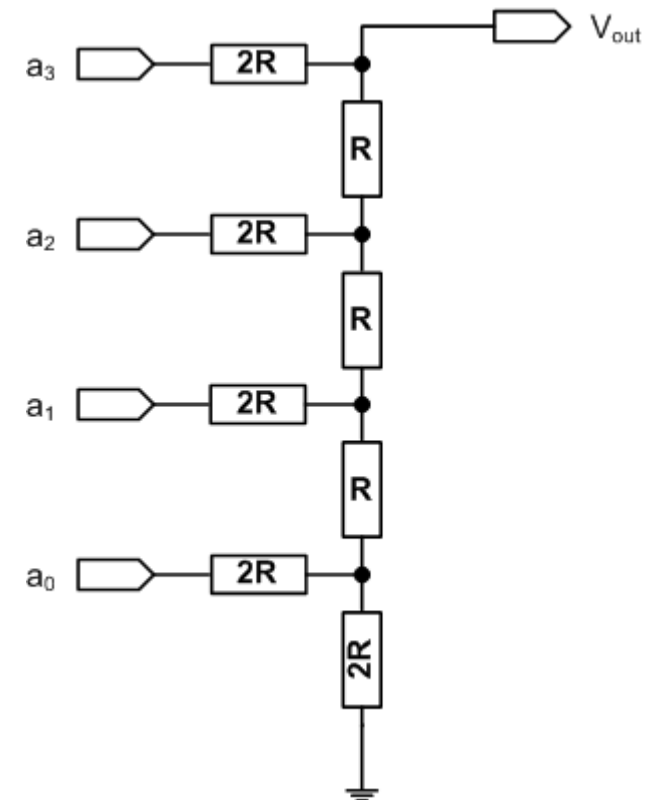
- Technical realization is restricted in the number of possible signal values
 - Quantization required to fit existing system
 - Only limited precision available
 - Cost
 - Amount of data
- Discretization step inherently included in AD Converters (see previous examples)

→ Signal can not be reconstructed perfectly because of errors introduced by quantification

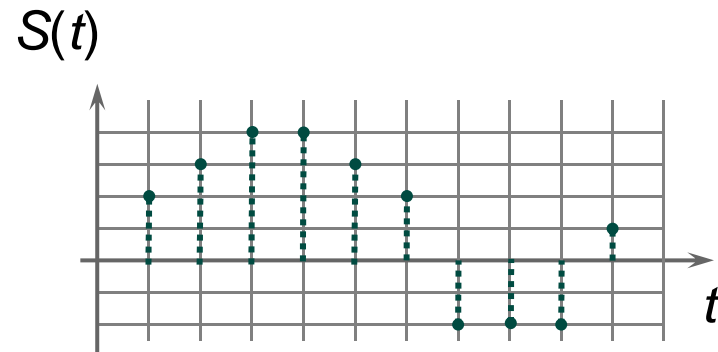
Digital Analog Conversion (DAC) – Example

- Digital Values are converted into analog signal
- Reconstruction Filters can be used to reconstruct original analog signal in some cases (reconstruction filters) using interpolation
 - Bandwidth has to be large enough
- Quantization Errors
 - Discretization step in ADC always introduces noise (relatively low)

R-2R Ladder

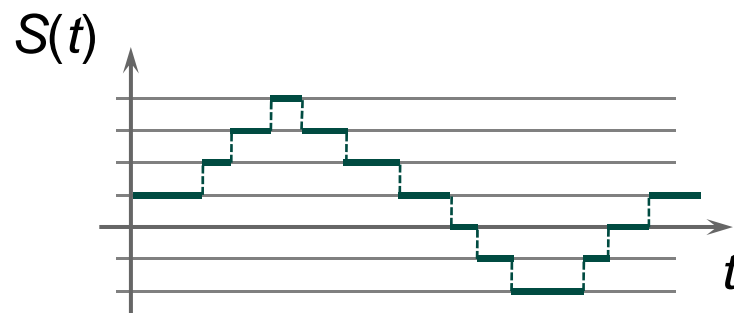


Motivation Oversampling



- Digital Signals are assumed to have discrete values only at discrete points in time

- High/Low
- -1 , 0 , +1



- Digital signals are often represented by analog signals

- discrete values
- time continuous

- Under ideal conditions, a single sample during a given interval is sufficient to reconstruct the original digital signal

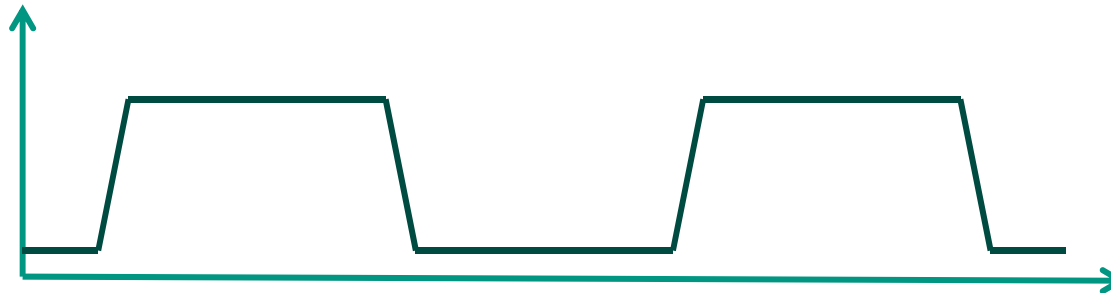
Representation of digital Signals



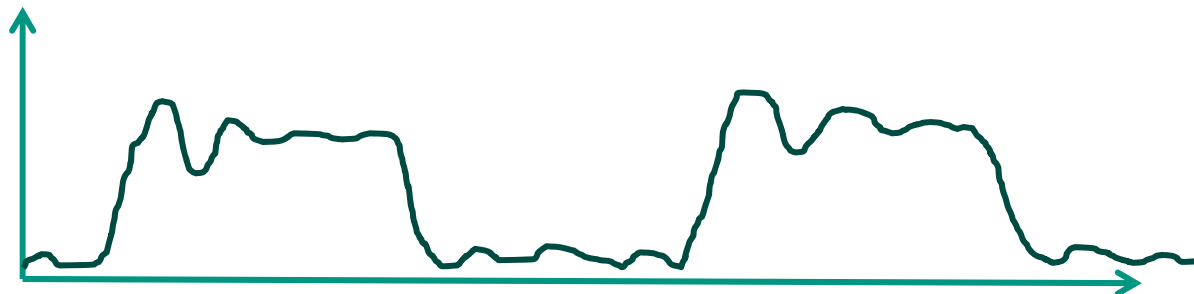
Data



Digital Signal



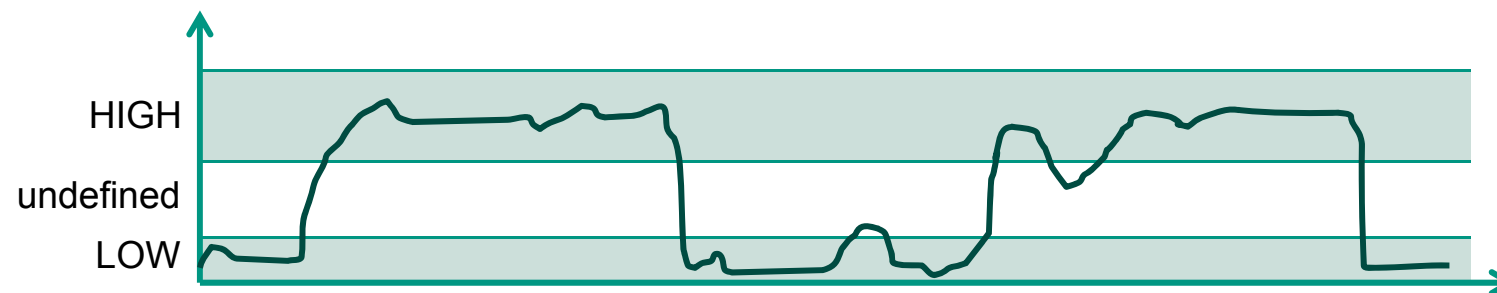
Ideal Analog
Representation



Analog
Representation

Acceptance Bands for Digital Signals

- Absolute value of the analog representation is not of interest only the signal interval is of importance
- Distortion of digital signals is possible if the analog signal is distorted
→ Definition of Acceptance Bands



- Reconstruction of digital signal is possible as long as analog signal is within the acceptance bands
- If signal drops into undefined:
 - interpret a signal change only if and when a valid acceptance band is entered
 - Hysteresis (see Schmitt Trigger in the analogue domain)

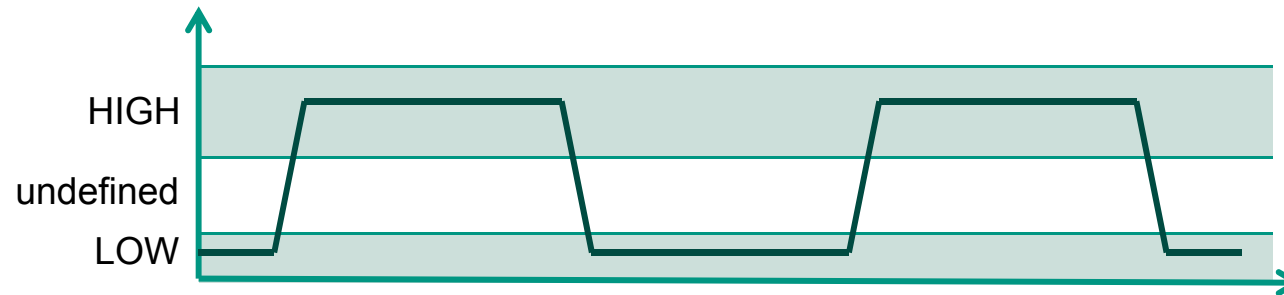
Representation of digital Signals



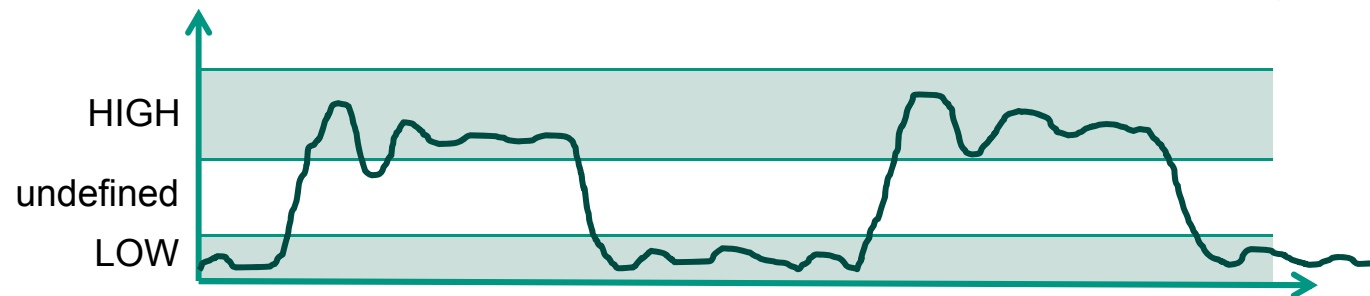
Data



Digital Signal



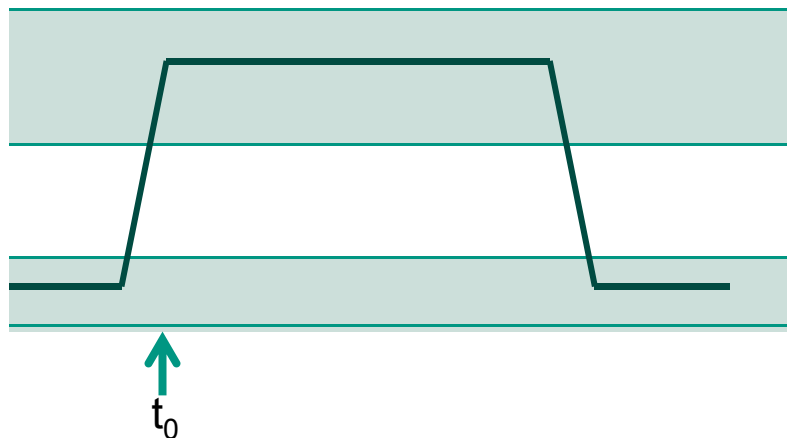
Ideal Analog Representation



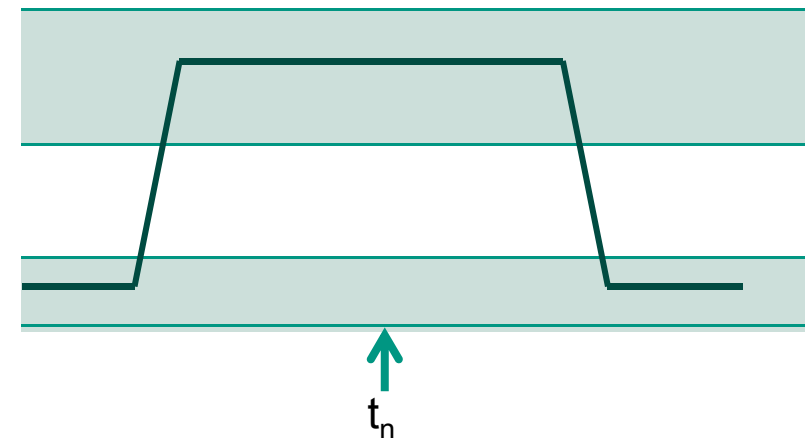
Analog Representation

Sample Points for digital Signals

- Goal in a communication system: get valid data as early as possible
→ Sampling of signals as early as possible

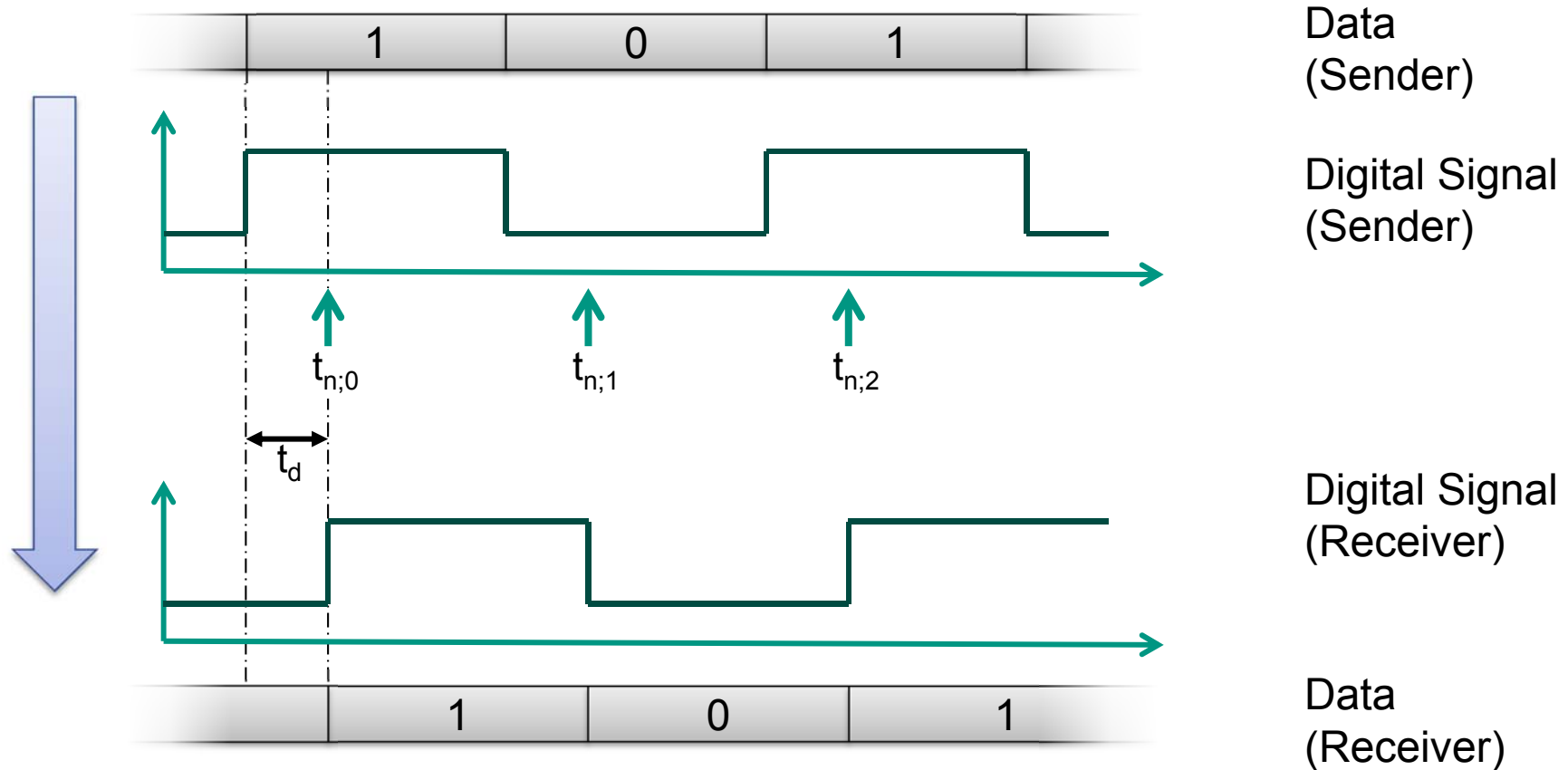


- Sampling at t_0 (beginning of a bit)
 - + no latency
 - possibly high error rate
 - over-/underswing
 - slopes
 - timing (detection of bit start)



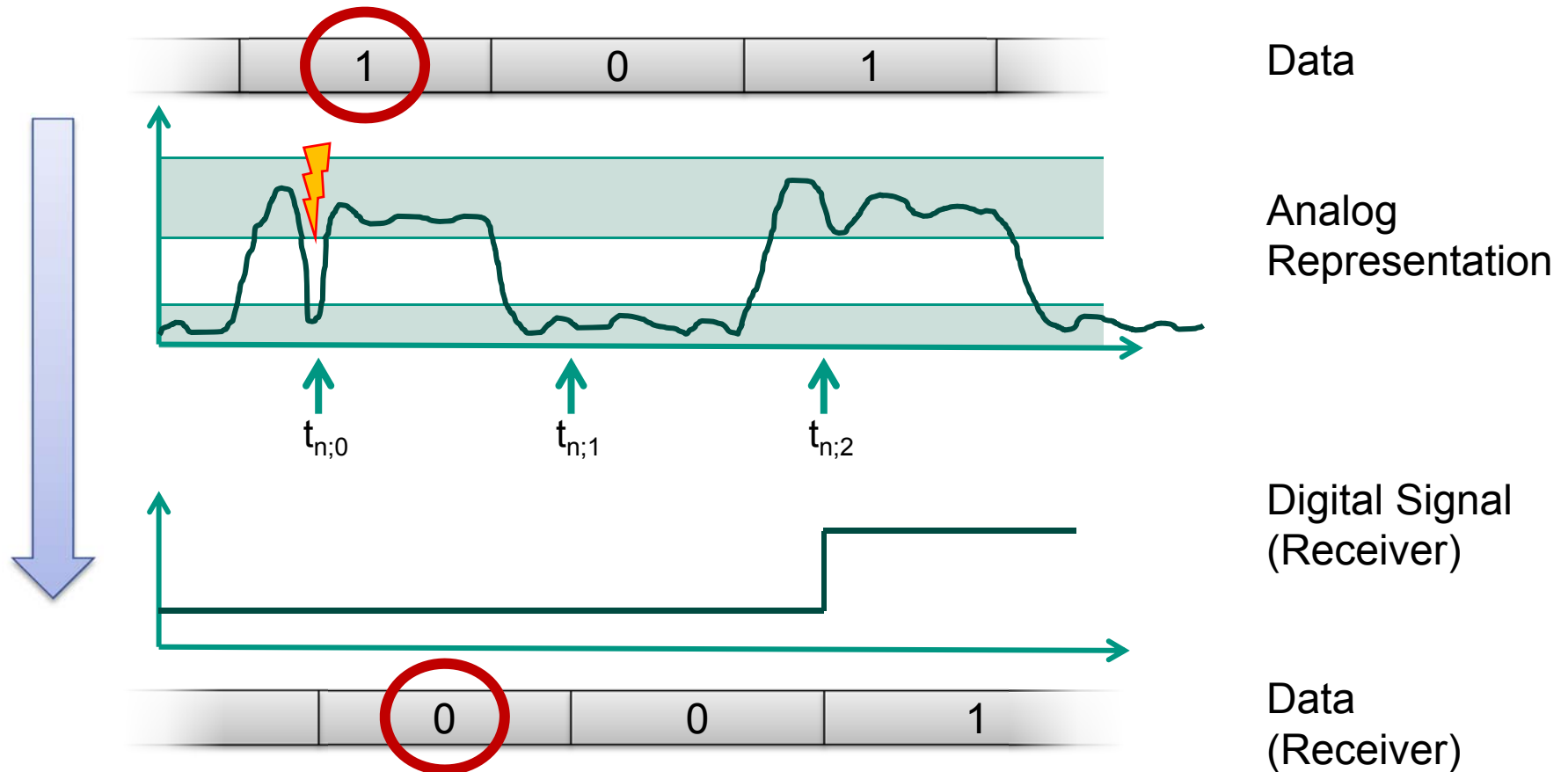
- Sampling at t_n
 - + Signal is stable
 - high latency with late t_n

Time Delay of Samples



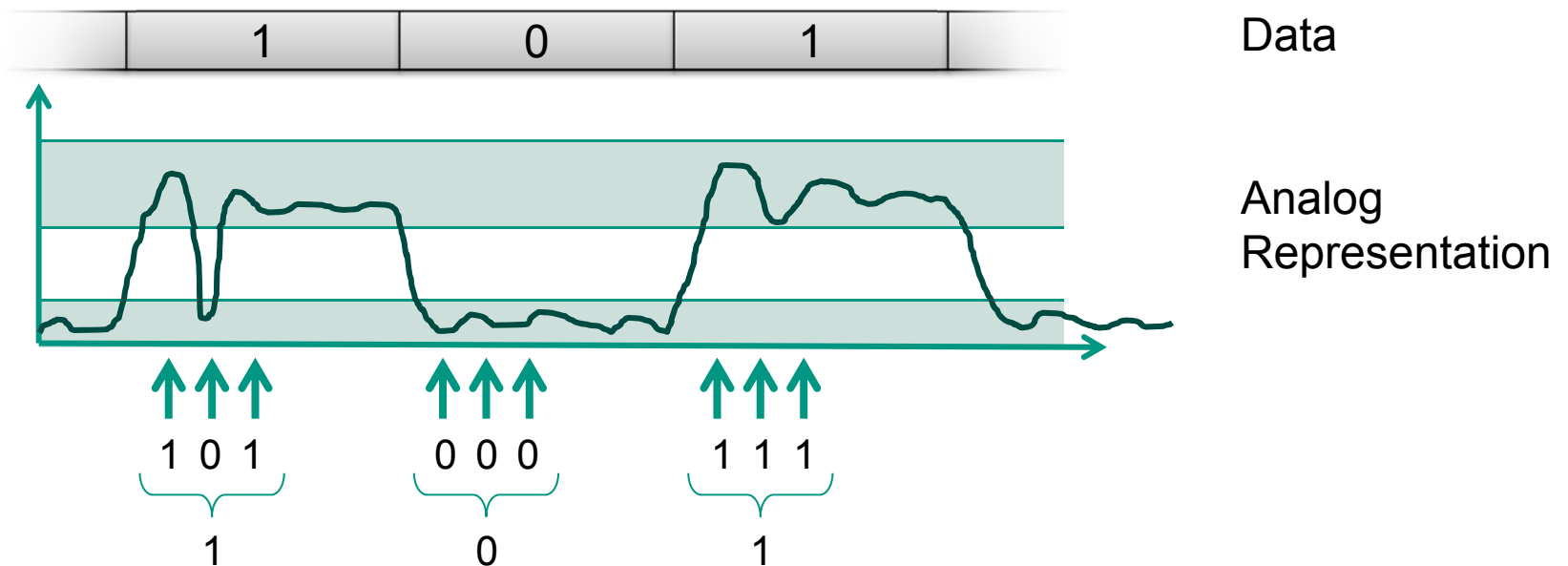
- Sampling at later time points increases latency (t_d) on receiver side

Error Case: Sampling of Digital Signals



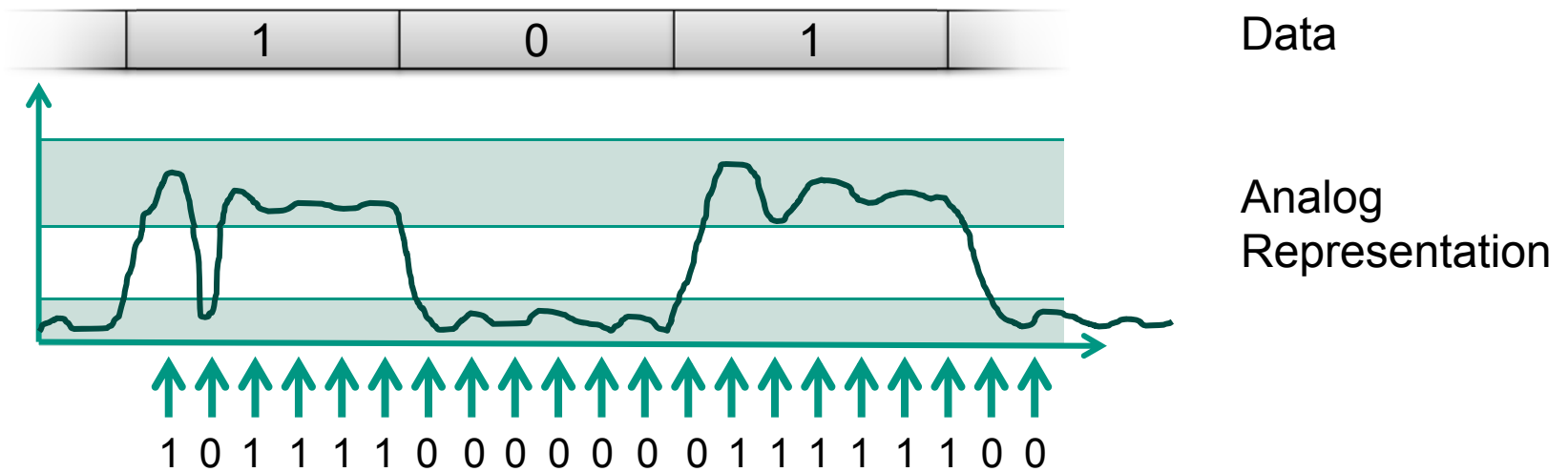
- Problem: Interferences (glitches) could change value of this single sample → wrong signal value will be assumed

Oversampling of Digital Signals



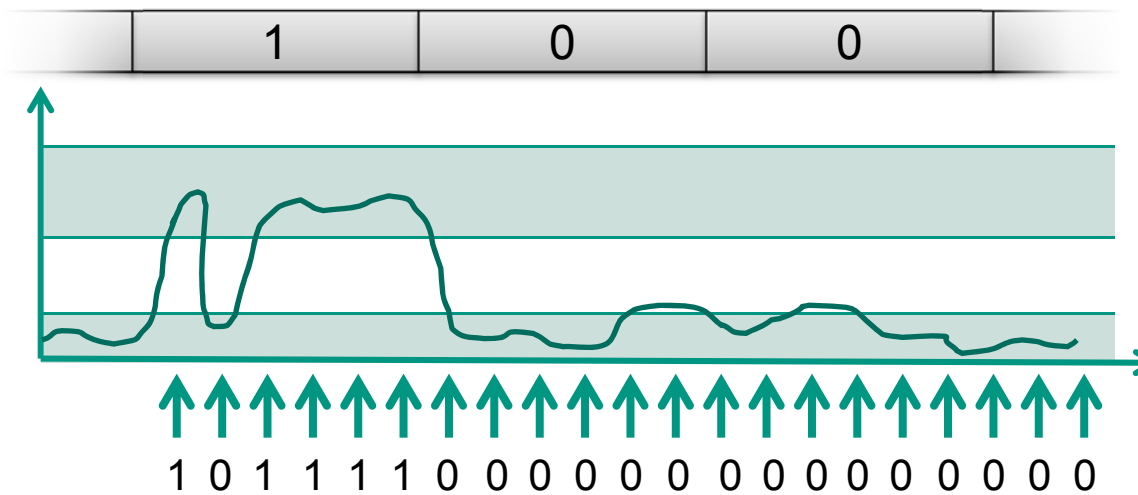
- Oversampling:
 - n sample points per bit (n-times oversampling)
 - Majority voting among the samples → value that appears most is taken
- Advantages:
 - Filtering of short distortions is possible
- Disadvantages:
 - Final signal is available only after the last sample has been taken

Oversampling of Digital Signals



- Oversampling can be done continuously
 - only first n sample points of a bit are used to read value
 - remaining samples are not needed
- If no information about the start of a bit is available:
 - sliding window
 - each sample value is used
 - allows for edge detection

Ambiguity of Digital Signals

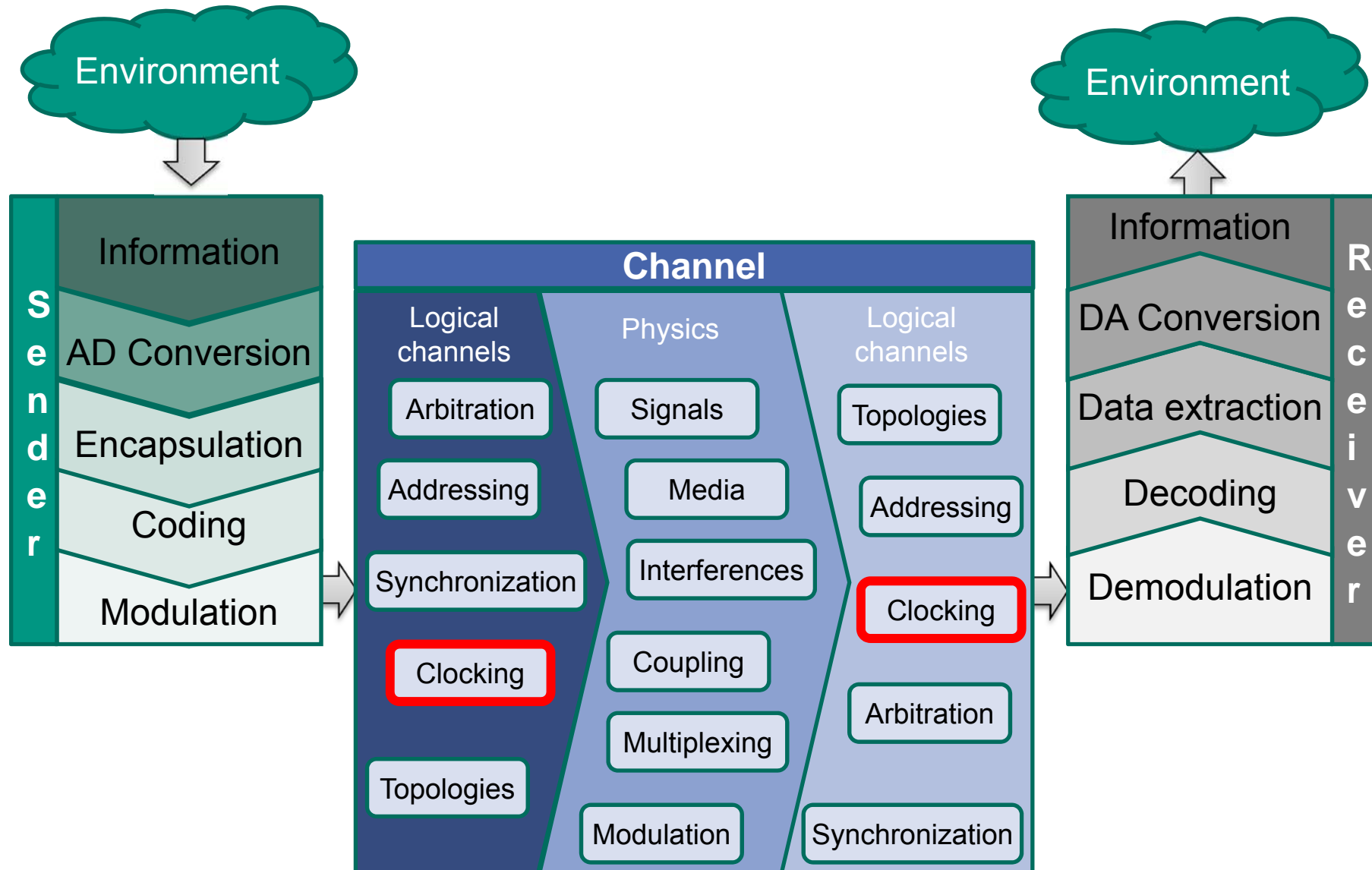


Data

Analog
Representation

- What happens with series of equal data?
 - Where does a bit begin and where does it end?
 - Additional information needed!

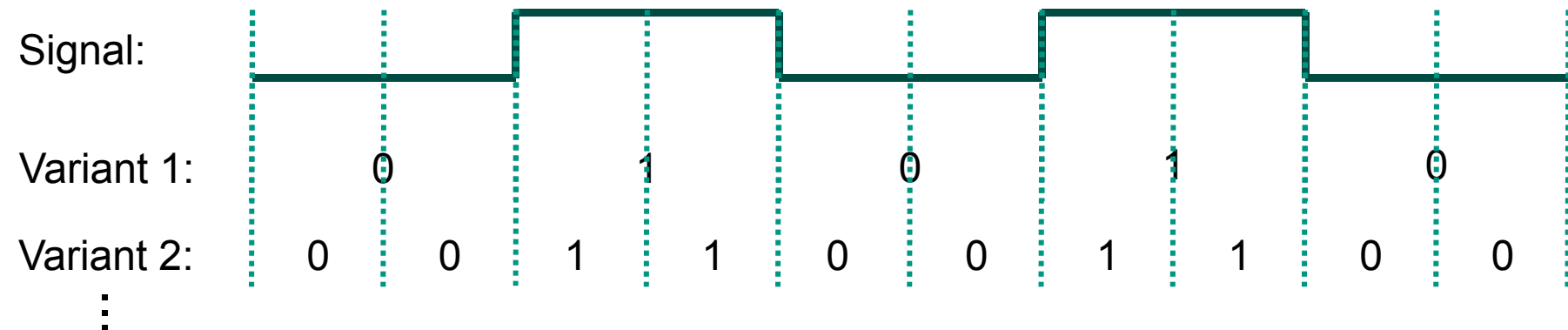
Transmission System – Clocking



Clocking

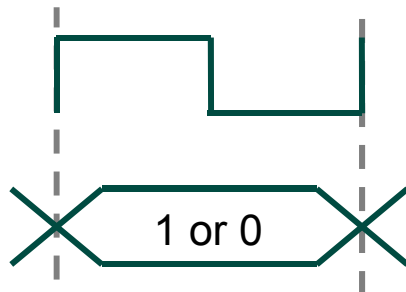
- For data transmission it is important to separate individual bits
 - Where is the end of a bit, when does a new one start?
 - Possible ambiguity

→ Clocking is used to determine the borders of a bit

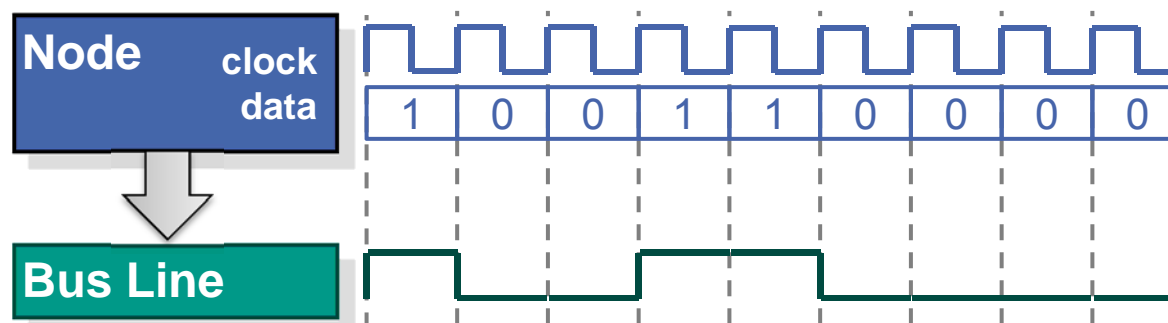


Using a Clock Signal

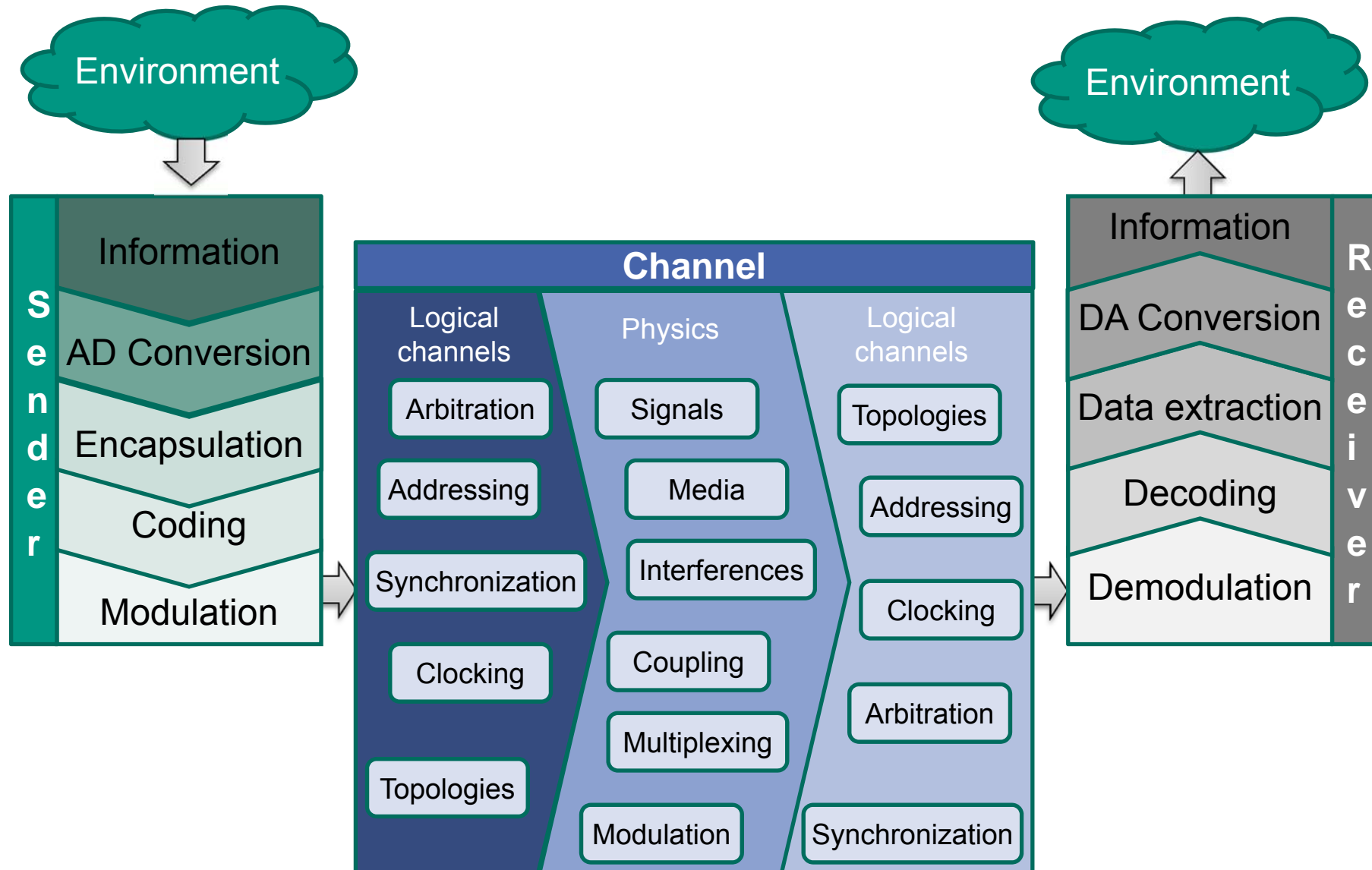
- Clock signal acts as time measure



- Transmission of information is based on time slots (clock cycles)



Transmission System – Overview



Definitions

■ Bit

- A **bit** (a contraction of **binary digit**) is the basic unit of information in computing and telecommunications. A bit can only have two values: either 1 or 0. A bit can be implemented in hardware by means of a two state device.
- The symbol for bit, as unit of information, is either simply “bit” or lowercase “b”.

■ Bit rate

- In telecommunications and computing, **bit rate** is the number of bits that are conveyed or processed per unit of time.
- The bit rate is quantified using the bits per second (bit/s or bps) unit.

Source: Wikipedia

Definitions II

■ Symbol

- A **symbol** can be described as either a pulse (in digital baseband transmission) or a “tone” (in passband transmission using modems) representing an integer number of bits. There may be a direct correspondence between a symbol and a small unit of data (for example, each symbol may encode one or several binary digits or 'bits') or the data may be represented by the transitions between symbols or even by a sequence of many symbols.

■ Symbol rate

- In digital communications, **symbol rate** (also known as baud or modulation rate) is the number of symbol changes (waveform changes or signaling events) made to the transmission medium per second using a digitally modulated signal or a line code.
- The Symbol rate is measured in baud (Bd) or symbols/second.

Source: Wikipedia

Relation Between Bit Rate and Symbol Rate

Bit rate is related to, but should not be confused with the symbol rate!

- Only equal when there are two levels per symbol, representing 1 and 0.
--> binary signals
- For most line codes and modulation methods:
 - Symbol rate \leq Bit rate
 - A transmission scheme with 2^N signal levels can transfer N bit/symbol
 - Bit rate = symbol rate * N
- For some self-synchronizing line codes, where each bit is represented by two pulses
 - Bit rate = symbol rate / 2

Think pair share

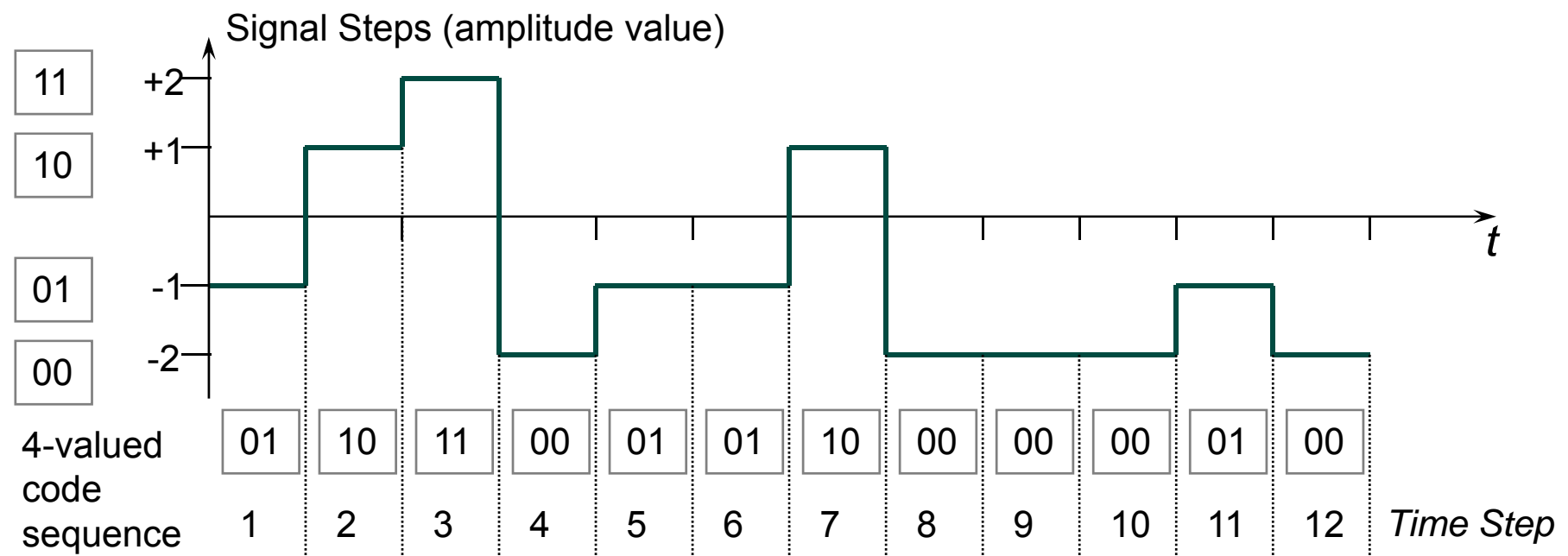
**Do you know examples for systems where bit-
and baudrate are not equal?**



**Discuss with your neighbor first, then give an
example to the audience.**

Example: Multi-Value digital Signal

- The (discrete) coordinate of a signal can have more than two values
 - Example: DIBIT: two bit per coordinate value (4-valued Signal)
- One Symbol encodes two bits



Representation of Signals

- What is an electrical representation of binary signals?
 - current, no current
 - High voltage, low voltage (e.g. amplitude)
 - High frequency, low frequency
 - bright light, dim light

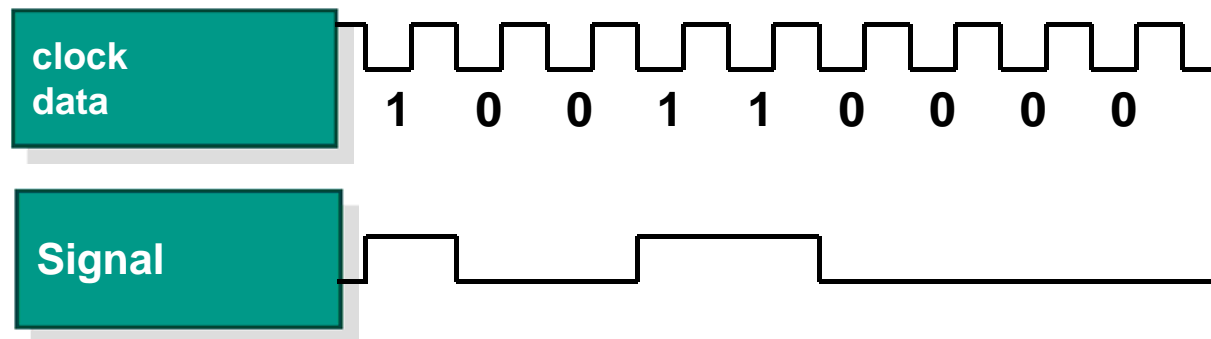
- Digital Baseband Representation
 - Signals contain no modulation, in general it contains $f=0$.
Baseband signal = sequence of (square-)Impulses

- Modulation Representation
 - Signals are modulated onto a carrier frequency
 - it does not need to contain $f=0$

Baseband Representation

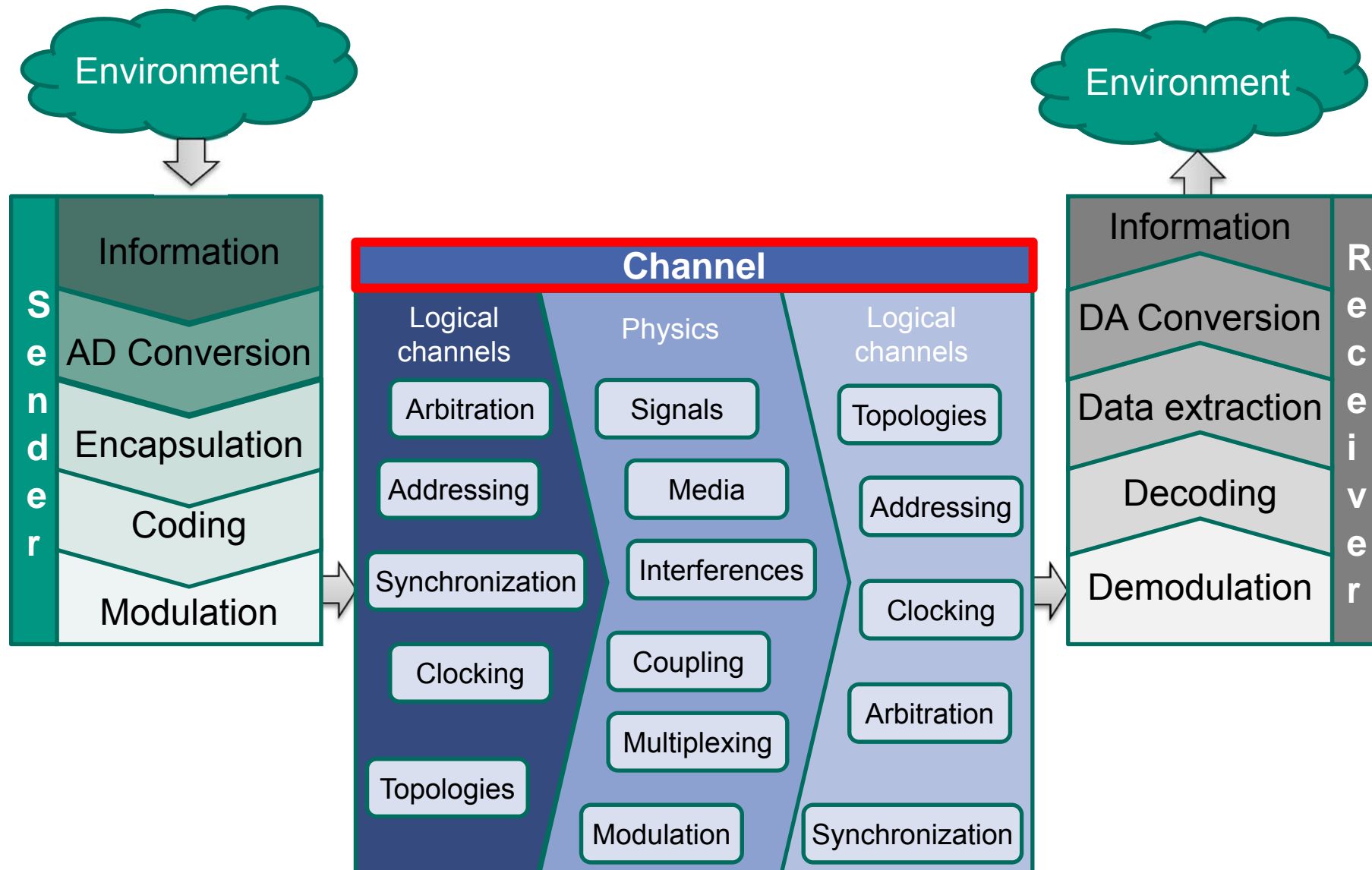
■ Baseband Representation

- Signals contain no modulation, in general it contains $f=0$.
Baseband signal = sequence of (square-)pulses



- Intuitive representation of data
- Requires theoretically unlimited bandwidth (due to square-pulses)
- Capacitive or inductive coupling of nodes infeasible

Transmission System – Overview



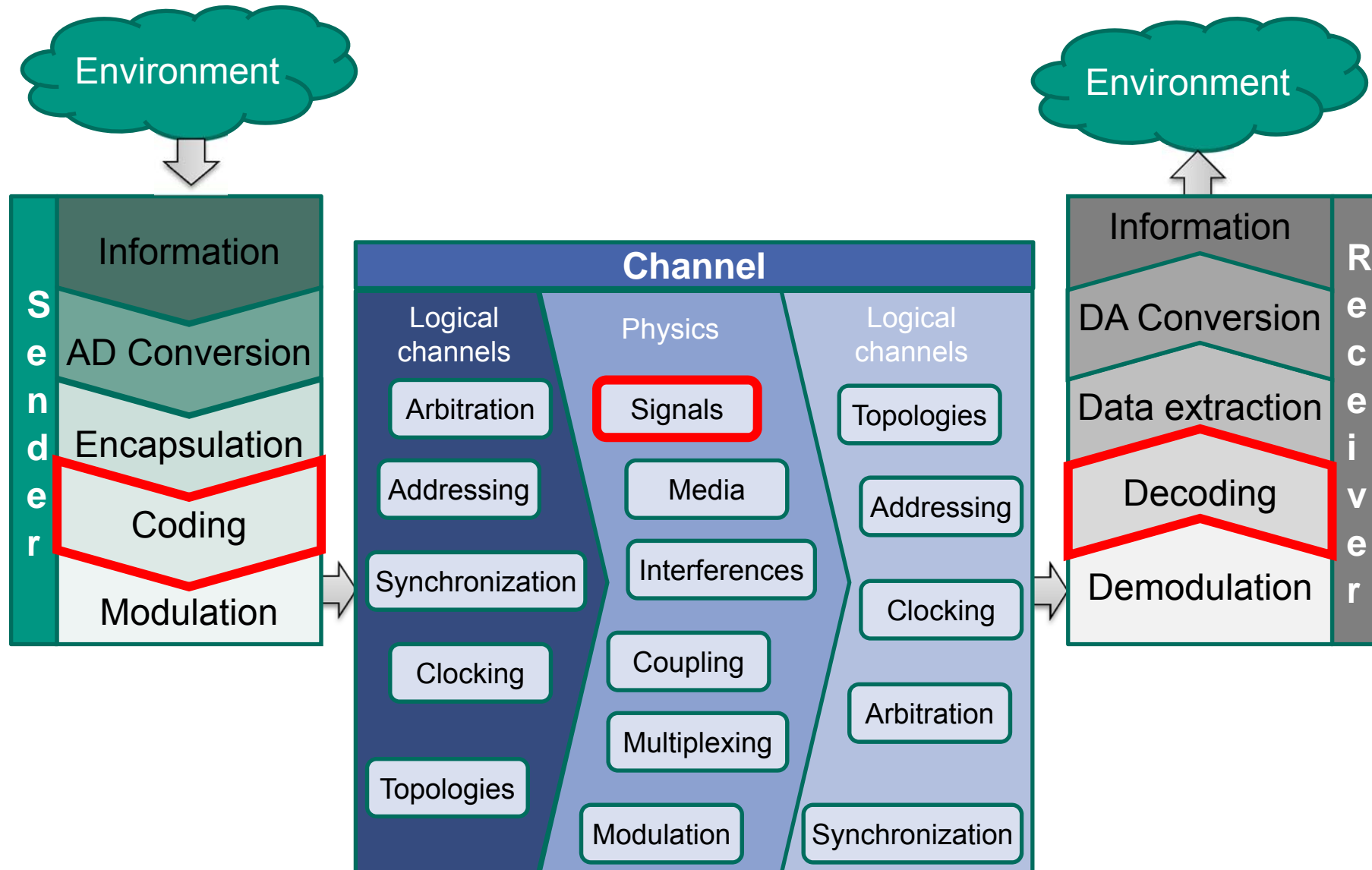
Transmission Capacity of an ideal channel

- How many bits can be transmitted error free over a band limited channel?
- Following the sampling theorem:
 - a channel limited in frequency to f_{limit} can transmit at maximum $2 \cdot f_{\text{limit}}$ symbols per second.
- H. Nyquist, 1924:
 - Noise-free channel
 - V = amount of discrete signal steps
 - Maximum data rate: $2 \cdot f_{\text{limit}} \cdot \log_2 V \text{ [bit/s]}$
 - Data rate increases with the number of signal steps.
- Example:
 - $F_{\text{limit}} = 3000 \text{ Hz}$, 8 signal steps
 - $D_{\text{limit}} = 2 \cdot 3000 \cdot \log_2(8) = 18000 \text{ bit/s}$

Transmission Capacity of a real channel

- C. Shannon, 1948:
 - Channel with random noise
 - How many different signal steps can be distinguished correctly?
- Signal/Noise ratio S/N limits the number of bits per symbol
- Number of transmittable bits per second:
 - Shannon Limit = $W \cdot \log_2(1 + S/N) [bit/s]$
 - $S/N=0$: no information (infinite amount of noise)
 - $S/N=1$: ~ 1 bit/s
 - $S/N>1$: multiple bits/s
- Example: analog telephone line
 - $S_{\text{norm}} = -10\text{dB}$, $N_{\text{norm}} = -34\text{dB}$, $W = 3000\text{Hz}$
→ 24,000bit/s

Transmission System – Line codes

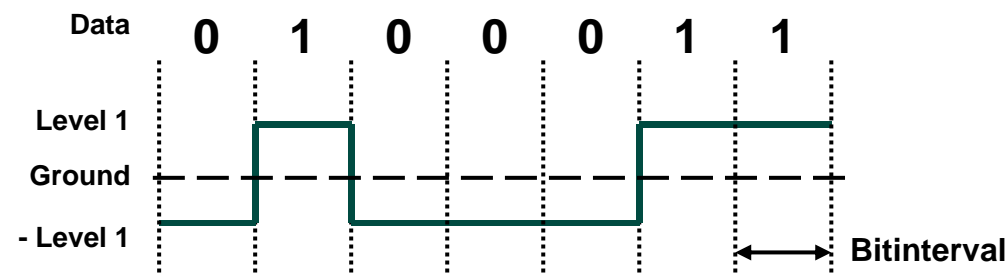


Motivation: Line Codes

- Is baseband representation of bits sufficient?
- Value of bit can be encoded, but how about other features?
 - Clock recovery
 - Multi-valued signals
 - Simple signal reconstruction
 - Error detection
- There should be other encoding schemes besides baseband representation → Line codes

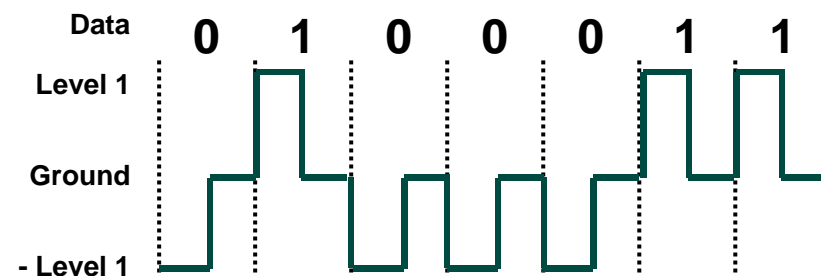
Non-Return-to-Zero (NRZ)

- '1' is represented by a HIGH-signal and '0' as a LOW-Signal
 - Invariant signal level during a bit interval
 - Signal transitions occur at the interval boundary
 - NRZ is the de-facto standard in digital systems
-
- Easy to implement
 - DC Component can be substantial
 - Unsuitable for timing recovery



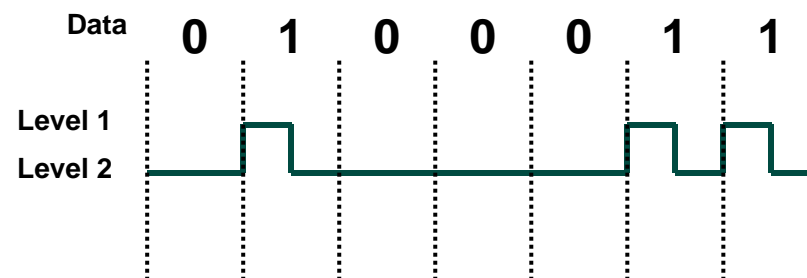
Return-to-Zero (RZ)

- Logic '1' is represented by a HIGH-signal in the first half of the bit interval
 - Logic '0' is represented by a LOW-signal in the first half of the bit interval
 - In the second half of the bitinterval code returns to the basic state (Zero)
-
- Easy Clock Recovery possible
 - The baud rate is double the bit rate → doubled bandwidth is necessary
 - DC Component can be substantial



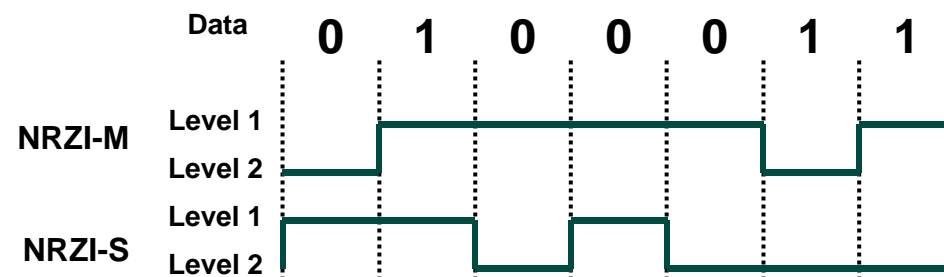
Return-to-Zero (RZ) Unipolar

- '1' is represented by a HIGH-signal and '0' as a LOW-Signal
 - Square pulse within the first half of a bit interval stands for a '1'
 - Afterwards code falls back to the basic state (zero)
-
- Simple implementation in digital hardware
 - In the extreme case (long sequences of '1's) the baud rate is double the bit rate → doubled bandwidth is necessary
 - No timing recovery possible for long sequences of '0's
 - DC Component can be substantial



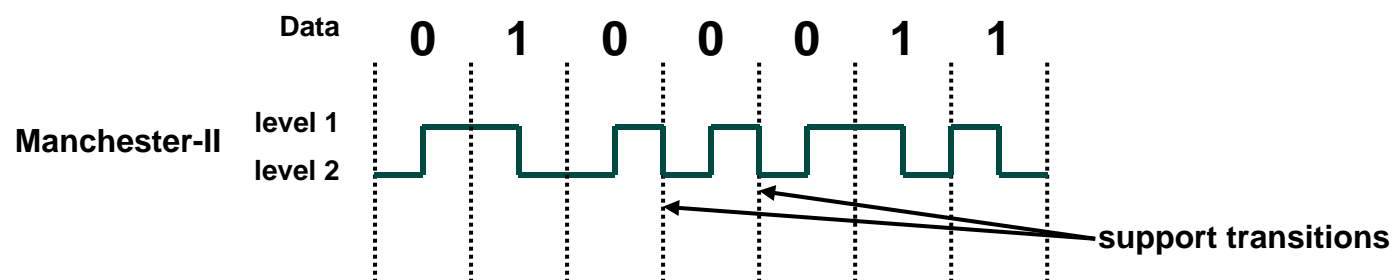
NRZI Coding (Differential coding)

- Not the absolute signal value is used for coding, but instead a signal value depending on the polarity of the previous signal element.
- Two variants of NRZI codes
 - NRZI-M (Mark)
 - Signal transition to opposite signal value when transmitting '1'
 - No signal transition when transmitting '0'
 - NRZI-S (Space)
 - Identical to NRZI-M, but signal transitions when sending a '0'
- For both variants:
 - Edges more easily detectable than absolute levels
 - DC Component can be substantial
 - Timing recovery not always possible



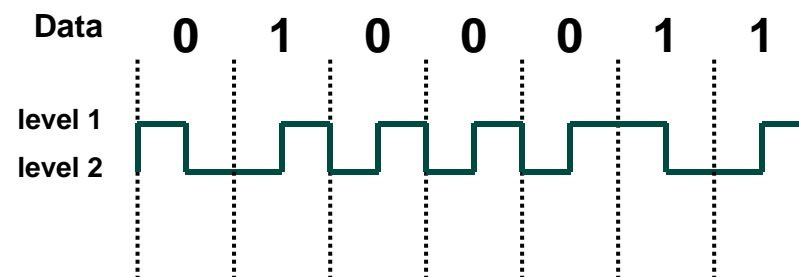
Manchester Code

- At least one signal transition per bit interval, maximum two transitions per bit
- One possible definition (Manchester-II):
 - '1' = Signal transition from HIGH level to LOW level in the middle of the bit interval
 - '0' = Signal transition LOW level to HIGH level in the middle of the bit interval
 - Can be created via XORing of NRZ data and clock signal
- Easy timing recovery, because at least one signal transition per bit interval exists
- no DC component exists for bipolar encoding
- Error detection on signal level: Missing an expected signal transition
- Support transitions increase baud rate → higher band width necessary



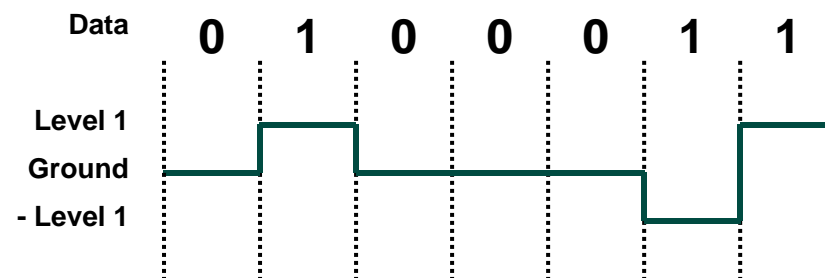
Differential Manchester Code

- Signal transitions in the middle of each bit interval
- Signal transition at the start of a bit interval only when encoding '0'
- This can be seen as a phase change when encoding '1' and no change in phase when encoding a '0'
- Output signal dependent on initial signal level at t_0
- Independent of polarity



AMI-Code (Alternate Mark Inversion)

- Encoding with more than two signal values
- '1' is represented by alternating positive or negative pulses
 - No DC component
 - Long '0'-sequences impede timing recovery



Classification of Line Codes

- Level orientated Line Codes
 - Symbols are represented by signal value
- Phase orientated Line Codes
 - Symbols are coded through phase jumps
- Unipolar Line Codes
 - only one active Signal level is used in addition to GND level
- Bipolar Line Codes
 - two different active Signal levels are use, GND is not used for coding
- Ternary Line Codes
 - Both symbol values 0 und 1 are encoded using three signal values (level 1, ground, level 2)
- Multi-valued Line Codes
 - Multiple bits are assembled within a single coding symbol

Clicker Session: Line Codes

■ <https://arsnova.eu/mobile/#id/77482045>



Important properties of Line Codes I

- Number of collectively coded symbols
 - In a single value more than one symbol can be encoded
- Direct Current Component
 - Sum V_{DC} of voltages on a wire over a longer period of time

$V_{DC} = 0$ \rightarrow DC balanced (or: zero-DC, zero-bias, DC equalized)
 $V_{DC} \neq 0$ \rightarrow a DC current is transported over the line

- Problem, if not DC balanced:
 - When transmitting data and power supply voltage over the same line, the supply voltage is affected
 - When using transducers as couplers, varying DC affects data interpretation on receiver's end
- In practice, DC balancing is only achievable in the statistic average of long transmission sequences.

Important properties of Line Codes II

- Error detection
 - Signal errors shall be detectable on the signal level itself.

- Required Bandwidth
 - Is correlated to attenuation
 - Higher frequencies are attenuated more than lower frequencies

- Clock Recovery
 - From received signals the value and clock of the send data can be recovered.
 - Clock recovery is needed if no separate clock line is available.
 - The clock portion shall not be correlated to the data content of the transmission.

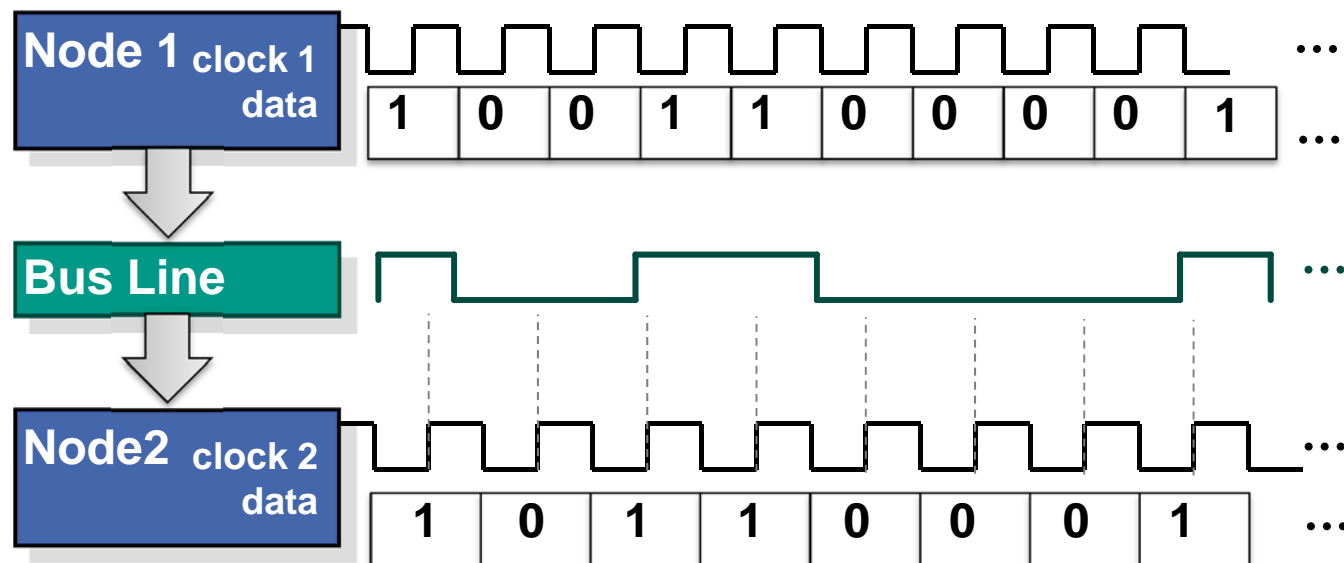
Homework

- Classify the Line Codes according to the properties given in the last two slides (+, ++, -, --, O)

| Line Code | DC Current | Required Bandwidth | Clock Recovery |
|------------|------------|--------------------|----------------|
| NRZ | | | |
| RZ | | | |
| NRZI | | | |
| AMI | | | |
| Manchester | | | |

Importance of Clock Signal

- Transmission of information is based on time slots (clock cycles).
- Clock signal acts as time measure
- For correct transmission of data sender and receiver have to have a common understanding of the current time (clock)



- Clock Recovery from Data allows
 - Detection of correct bit intervals
 - Synchronization

Block Codes

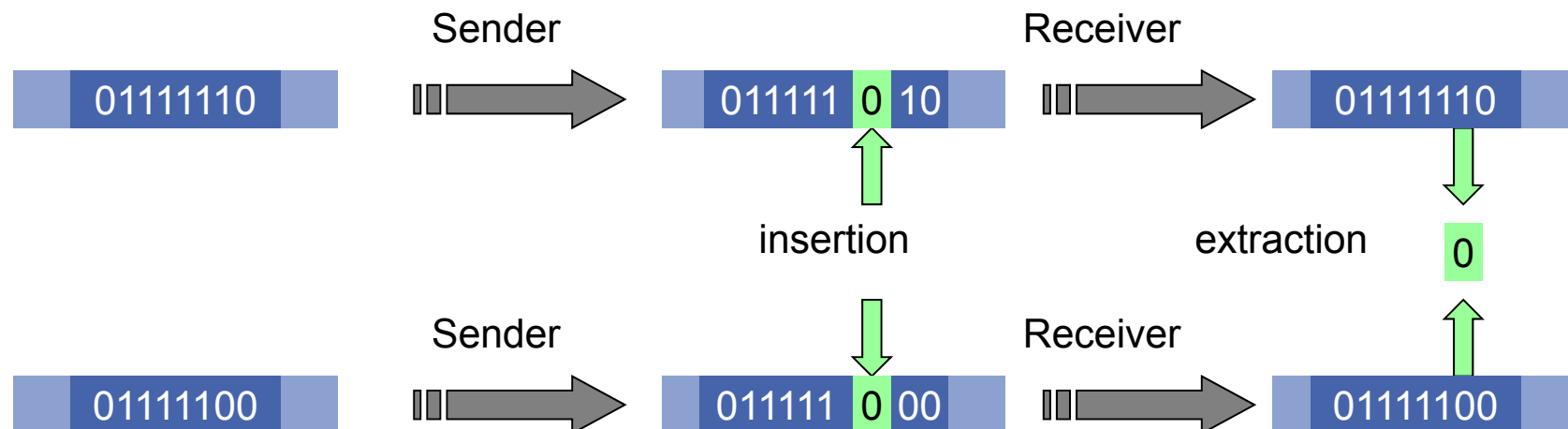
■ 4B/5B Code

- Avoiding the inefficiency of Manchester encoding
- No long sequences of ,0's or ,1's
- Addition of an extra bit to avoid such sequences
- 4-Bit data is encoded in 5-Bit blocks
 - Not more than one leading ,0's
 - Not more than two trailing ,0's
- Transmission is done using NRZ-I-Code
- 80% efficiency

| 4-Bit Data | 5-Bit Code |
|------------|------------|
| 0000 | 11110 |
| 0001 | 01001 |
| 0010 | 10100 |
| 0011 | 10101 |
| 0100 | 01010 |
| 0101 | 01011 |
| 0110 | 01110 |
| 0111 | 01111 |
| 1000 | 10010 |
| 1001 | 10011 |
| 1010 | 10110 |
| 1011 | 10111 |
| 1100 | 11010 |
| 1101 | 11011 |
| 1110 | 11100 |
| 1111 | 11101 |

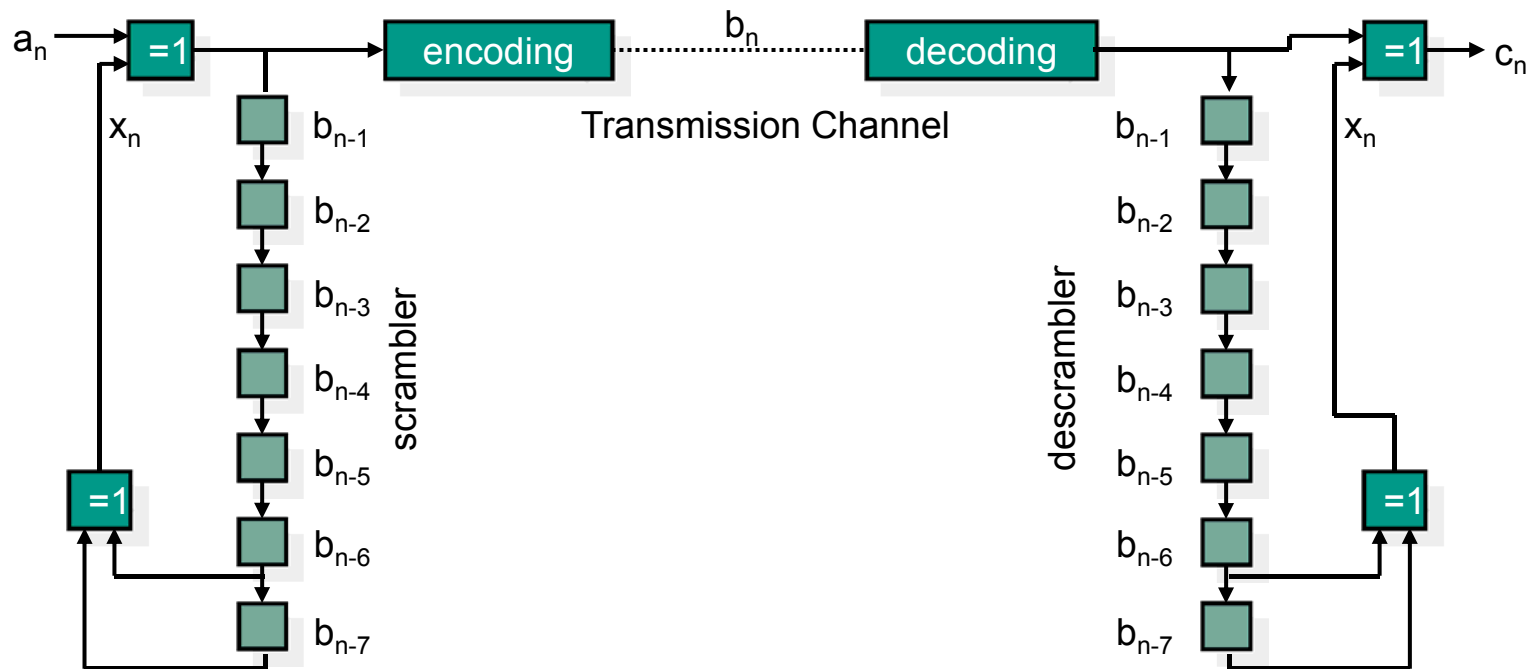
Scrambler: Bitstuffing

- Goal: Ensure timing recovery using frequent signal transitions, if no proper line encoding is being used.
- Bit Stuffing
 - Sender always inserts a ,0' after n ,1's (or vice versa)
 - Receiver interprets every sequence of 111...110 as 111...11 (or vice versa)
 - Example: $n=5$



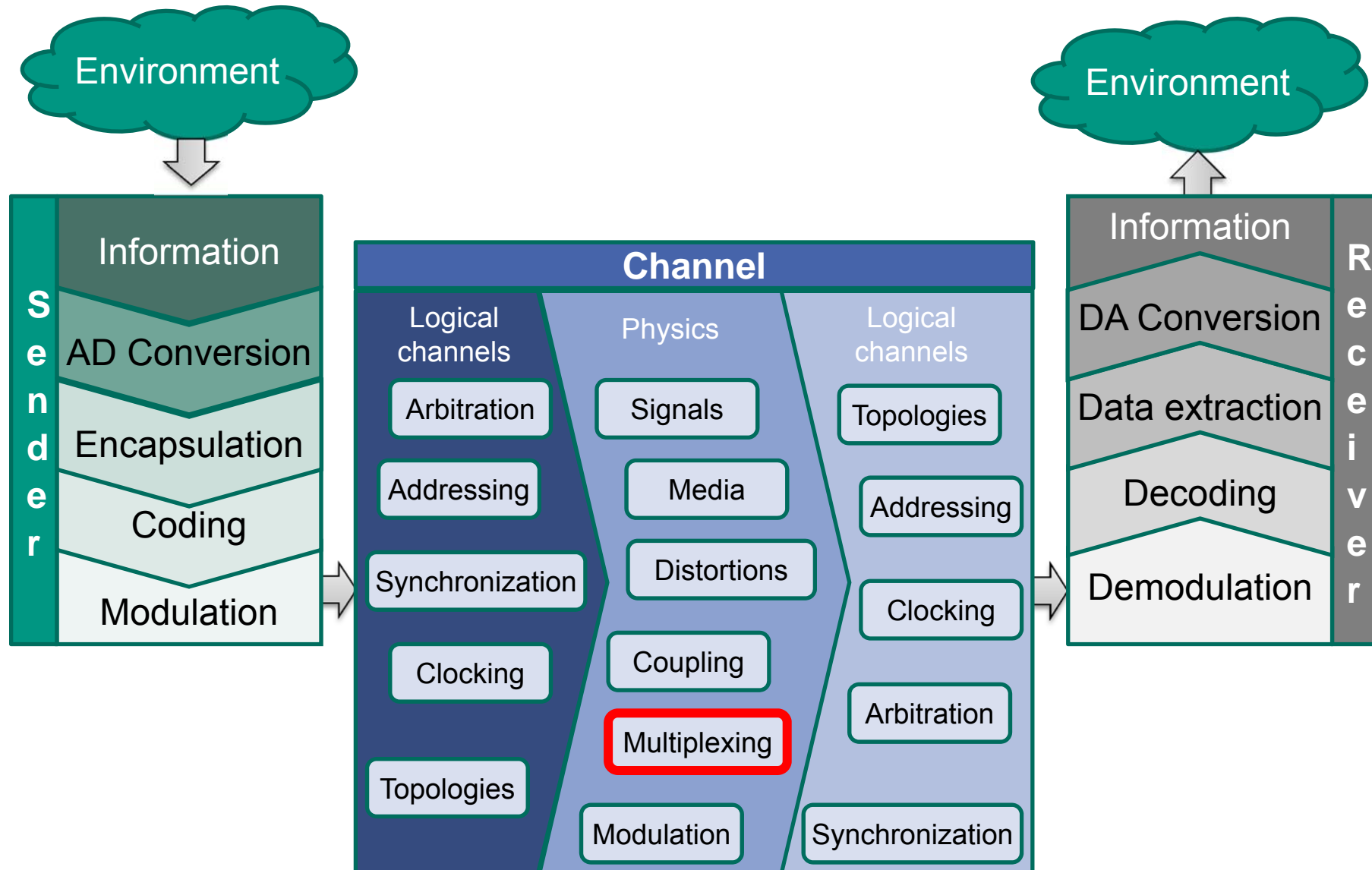
Scrambler: Use of Linear Feedback Shift Register

- Goal: Creating a seemingly chaotic sequence of bit values from any given data stream (including long sequences of '1's or '0's)
- Data stream is linked with sequence of a linear feedback shift register (LFSR)
- Resulting bit is send to receiver and also fed back into LFSR



- Descrambler build symmetrically to scrambler
- It holds: $c_n = b_n \otimes x_n = (a_n \otimes x_n) \otimes x_n = a_n$ with $x_n = b_{n-6} \otimes b_{n-7}$ (on both sides)

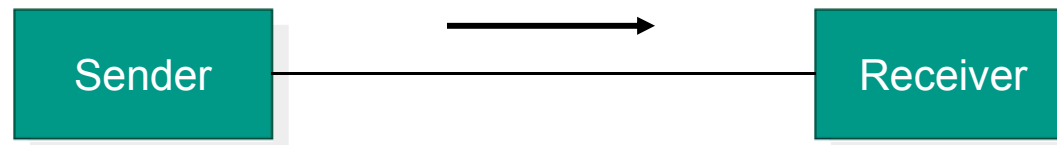
Transmission System – Multiplexing



Types of Transmissions

A transmission on a channel can be classified according to the direction of the data transmitted, as well as according to the degree of parallelism.

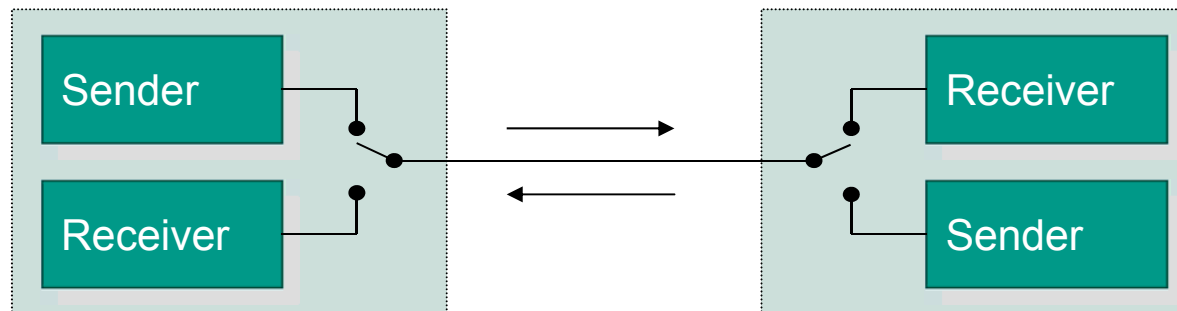
- Simplex transmission (respectively directional operation)
 - Only a single direction
 - Application: television, terrestrial radio, process data acquisition



Types of Transmissions

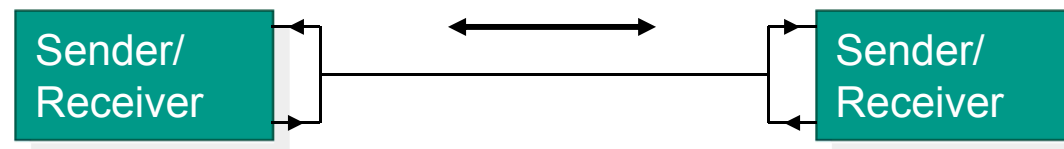
■ Half duplex transmission (also: alternating operation)

- In both directions, but not simultaneously
- Switching in terminal equipment
- Application: radio communication, 10base-T Ethernet , ...



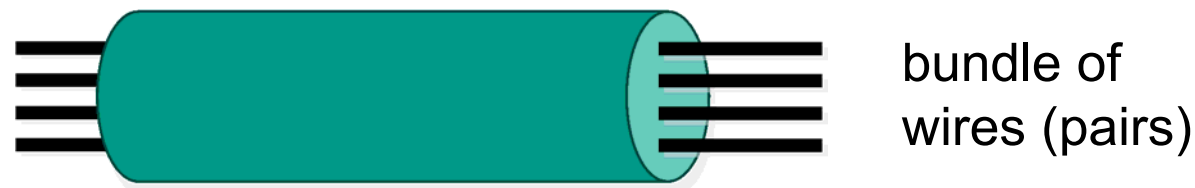
■ Full Duplex transmission

- In both directions simultaneously
- Application: Telephone, RS 232, 100Mbit Ethernet,...

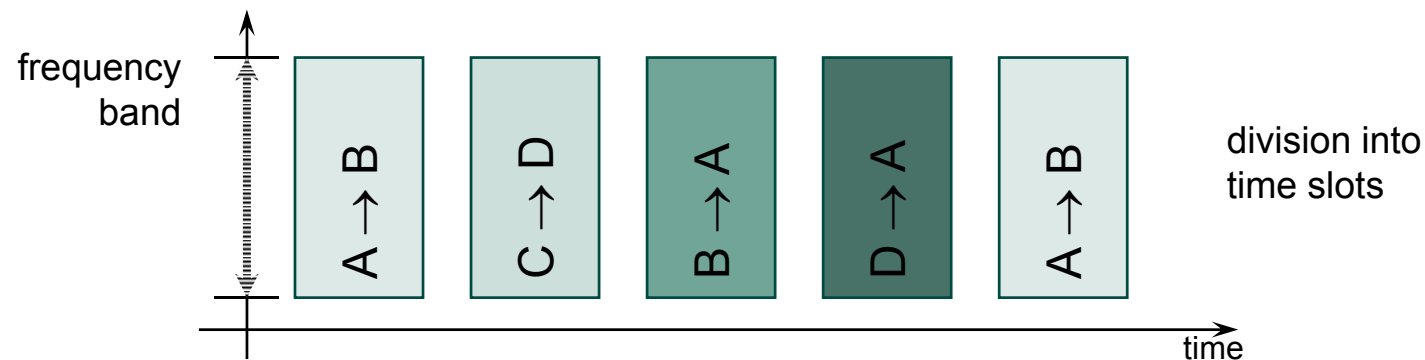


Multiple use of Media by multiplexing

- Multiple signals can be transmitted in multiplex mode
- Space division („copper multiplex“)
 - Representation of signals on multiple lines, baseband



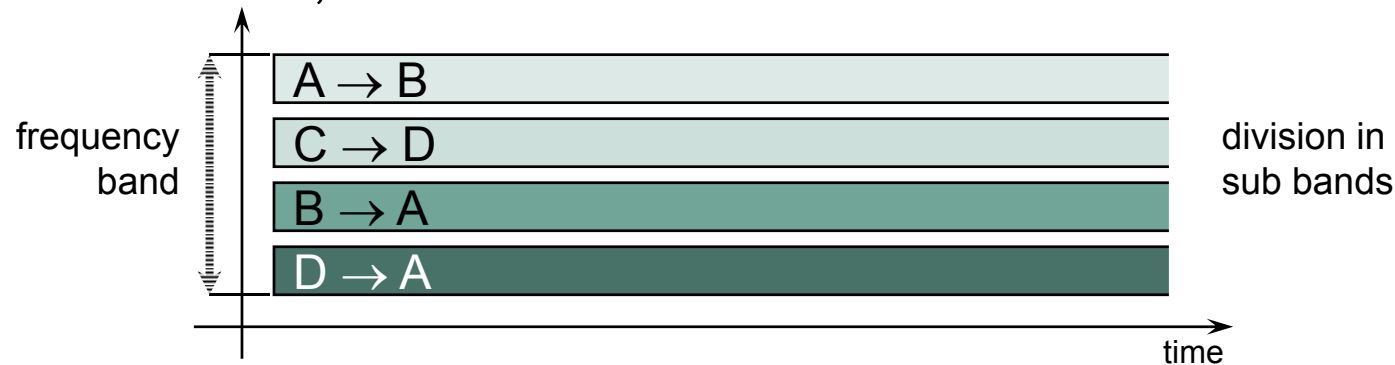
- Time division
 - Representation using temporal shifts (slots), base band



Multiple Use of media

■ Frequency division multiplex

- Representation using multiple frequencies at the same location, at the same time, modulation



■ Wavelength Division multiplex (special form of frequency division)

- transmission of multiple wave lengths (colors) over a single optical waveguide.
- currently up to 80 individual wavelengths possible
- especially suitable for wide area networks (WAN)
- within a single wavelength time multiplex possible – currently data rates of 80 GBits/s feasible (theoretically resulting in 6.4 Tbit/s per fiber)