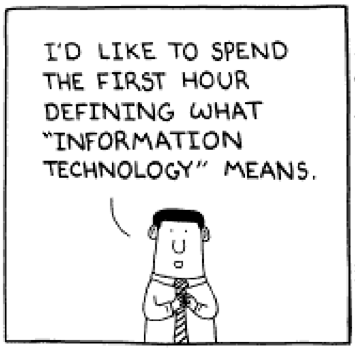


Communication Basics

Principles and Dogmas

 2010/02/15

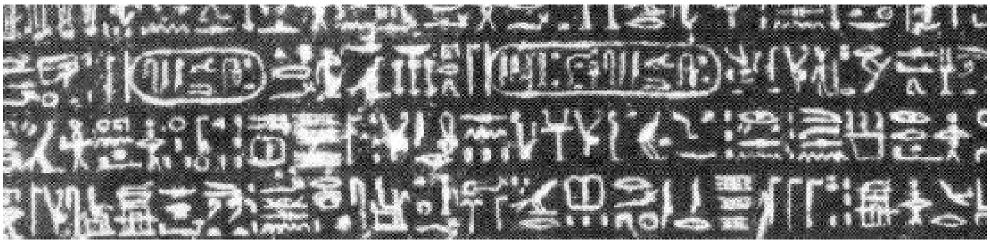
In this chapter we discuss basic communication issues, such as synchronization, coding, scrambling, modulation, and so on.

“Everything
should be made
as simple as possible,
...but not simpler.”

Albert Einstein

Information

- ***What is* information?**
  - **Carried by symbols**
  - **Recognized by receiver (hopefully)**
  - ***Interpretation* is the key…**



(C) Herbert Haas 2010/02/15 3

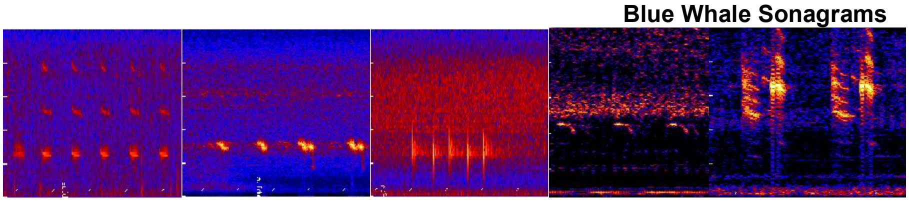
What is information? This question may sound quite easy but think a bit about it. Obviously we need symbols to represent information. But these symbols must also be recognized as symbols by the receiver. In fact, philosphical considerations conclude that information can only be defined through a receiver. The same problem is with art. What is art? Several decades and centuries had their own definitions. Today most critics use a general definition: art can only be defined in context with the viewer.

In the following chapters – throughout the whole data communication – we will deal with symbols representing information. A symbol is not a 0 or a 1. But this binary information can be represented by symbols. Be patient...

Symbols

- **Symbols (may) represent information**
  - **Voice patterns (Speech)**
  - **Sign language, Pictograms**
  - **Scripture**
  - **Voltage levels**
  - **Light pulses**

Blue Whale Sonagrams



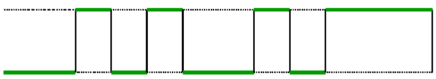
What is a good information source? From a theoretical point of view a random pattern is the best because you'll never know what comes next. On the other hand, if you receive a continous stream of the same symbol this would be boring. More than boring: there is no information in it, because you can predict what comes next! From this we conclude that a sophisticated coding – representing the information as efficient as possible using symbols – is a critical step during the communication process.

Throughout these chapters we will mainly deal with symbols such as voltage levels or light pulses.

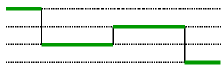
Look at the Blue Whale Sonograms. The x-axis represents time, the y-axis frequency and the color represents power density. This communication pattern is very complex (those of dolphins is even more complex). It is known that each herd has their own traditional hymn. And: they like to communicate!

Symbols on Wire

- Discrete voltage levels = "Digital"
  - Resistant against noise
- How many levels?
  - Binary (easiest)
  - M-ary: *More information per time unit!*



Binary



M-ary
(here 4 levels, e. g. ISDN)

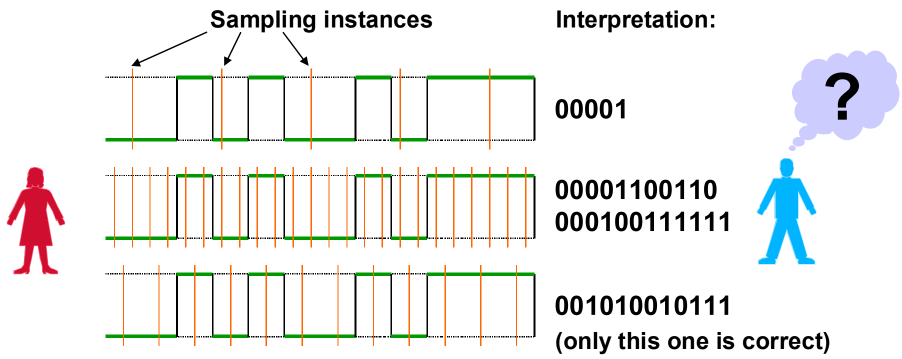
What symbols do we encounter on wire? Digital binary symbols are commonly known and widely in use. Why? Consider information transmissions in groups of symbols (for example the group of 8 binary symbols is called a byte). We have two parameters: the number base B and the group order C. If you calculate the "costs" that you get for arbitrary variations of B and C, and if we assume a linear progress (so that cost $=k\times B\times C$) then for any given (constant) cost the perfect base would be B=e, that is B=2.7182...

In other words: the perfect base is a number between 2 and 3. The technical easiest solution is to use B=2. Note that these considerations assume a linear cost progression.

In many cases we pay the price of higher efforts and use a larger base. This leads us to m-ary symbols and later to PAM and QAM.

Synchronization

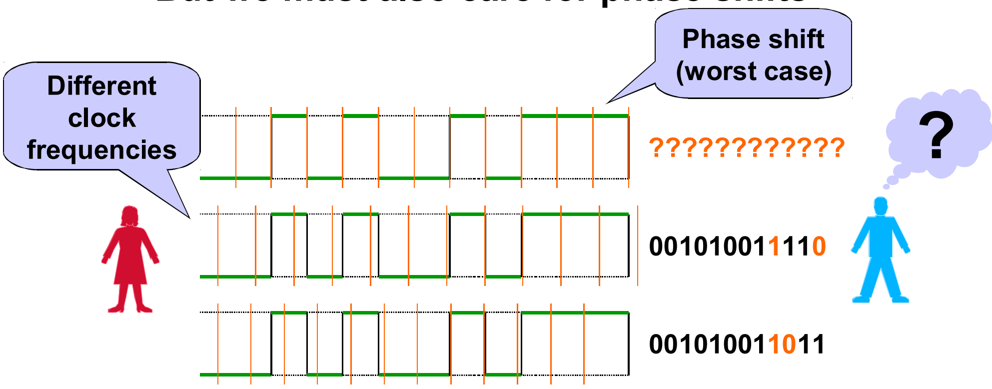
- Sender sends symbol after symbol...
- *When* should receiver pick the signal samples?
  - => Receiver must sync with sender's clock !



One of the most important issues among communication is that of synchronization. Nature forbids absolute synchronization of clocks. Suppose you are a receiver and you see alternating voltage levels on your receiving interface. If you had no idea about the sending clock then you would never be able to interprete the symbols correctly. When do you make a sample?

Synchronization

- In reality, two independent clocks are NEVER precisely synchronous
  - We always have a frequency shift
  - But we must also care for phase shifts



So we must assume that the receivers clock is approximately identical to the senders clock. At least we must deal with small phase and frequency gaps. As you can see in the slide above, we still cannot be sure when to make samples.

What we need is some kind of synchronization method.

Serial vs Parallel

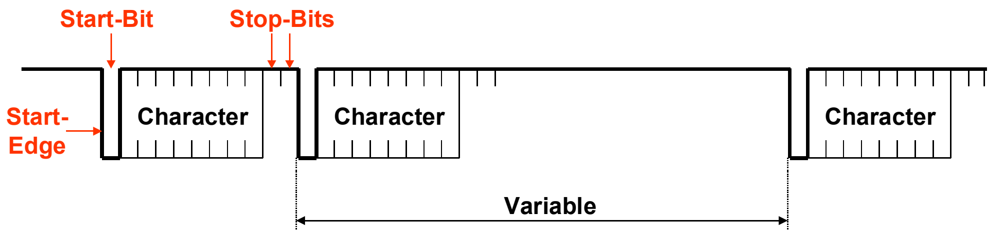
- Parallel transmission
  - Multiple data wires (fast)
  - Explicit clocking wire
  - Simple Synchronization but not cost-effective
  - Only useful for small distances
- Serial transmission
  - Only one wire (-pair)
  - No clocking wire
  - Most important for data communication

In case of parallel transmissions there is always a dedicated clock line. This is a very comfortable synchronization method. A symbol pattern on the data lines should be sampled by the receiver each time a clock pulse is observed on the clock line. But unfortunately, parallel transmissions are too costly on long links. In LAN and WAN data communication there are practically no parallel lines.

The most important transmission technique is the serial. Data is transmitted over a single fiber or wire-pair (or electromagentic wave). There is no clock line. How do we synchronize sender and receiver?

Asynchronous Transmission

- Independent clocks
  - Oversampling: Much faster than bitrate
- Only phase is synchronized
  - Using Start-bits and Stop-bits
  - Variable intervals between characters
  - Synchronity only during transmission
- Inefficient



One synchronization method is the Asynchronous Transmission. Actually this method cannot provide real synchronization (hence the name) but at least a short-time quasi-synchronization is possible. The idea is to frame data symbols using start and stop symbols (lets sloppy call them start- and stop bits). Using oversampling, the receiver is able to get a sample approximately in the middle of each bit – but only for short bit-sequences.

Asynchronous transmission is typically found in older character-oriented technologies.

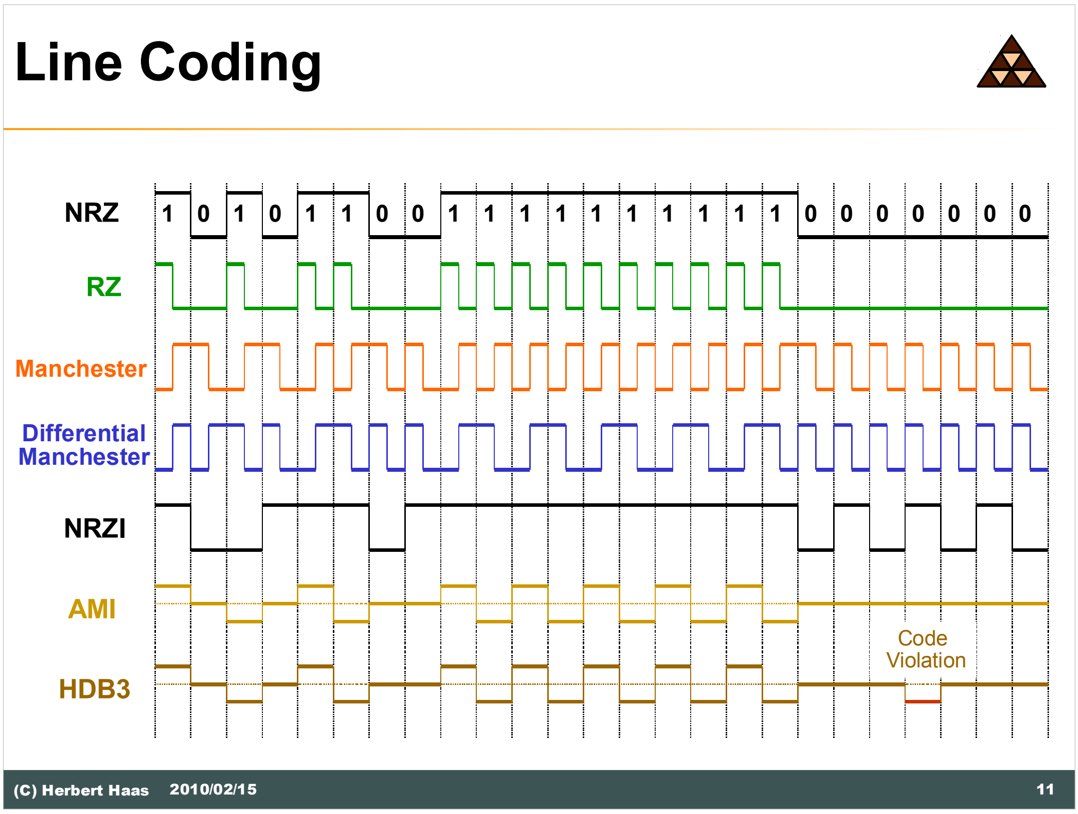
Example application: RS-232C
Relative overhead: 3/11

Synchronous Transmission

- Synchronized clocks
  - Most important today!
  - Phase and Frequency synchronized
- Receiver uses a Phased Locked Loop (PLL) control circuit
  - Requires frequent signal changes
  - => *Coding* or *Scrambling* of data necessary to avoid long sequences without signal changes
- Continous data stream possible
  - Large frames possible (theoretically endless)
  - Receiver remains synchronized
  - Typically each frame starts with a short "training sequence" aka "preamble" (e. g. 64 bits)

The most important method is the Synchronous Transmission. Don't confuse this with synchronous multiplexing–we are still on the physical layer! Two things are necessary: a control circuit called Phased-Locked-Loop (PLL) and a signal that consists of frequent transitions. How do we ensure frequent transitions in our data stream? Two possibilities: coding and scrambling our data.

Synchronous Transmission is found in most modern bit-oriented technologies–nearly anything you know.



The trivial code is Non Return to Zero (NRZ) which is usually the human naive approach.

Remarks:

RZ codes might also use a negative level for logical zeroes, a positive level for logical ones and a zero Volt level inbetween to return to. RZ is for example used in optical transmissions (simple modulation).

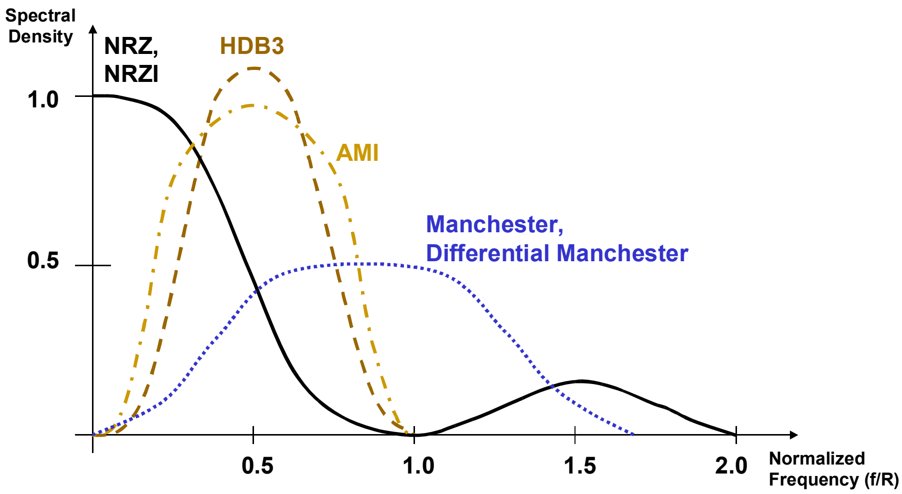
NRZI codes either modulate for logical ones or zeros. In this slide we modulate the zeroes, that is each logical zero requires a transition at the beginning of the interval.

NRZI means Non-return to zero inverted or interchanged.

B8ZS: same as bipolar AMI, except that any string of eight zeros is replaced by a string of two code violations.

Manchester is used with 10 Mbit Ethernet. Token Ring utilizes Differential Manchester. Telco backbones (PDH technology) use AMI (USA) or HDB3 (Europe). Of course there are many many other coding styles.

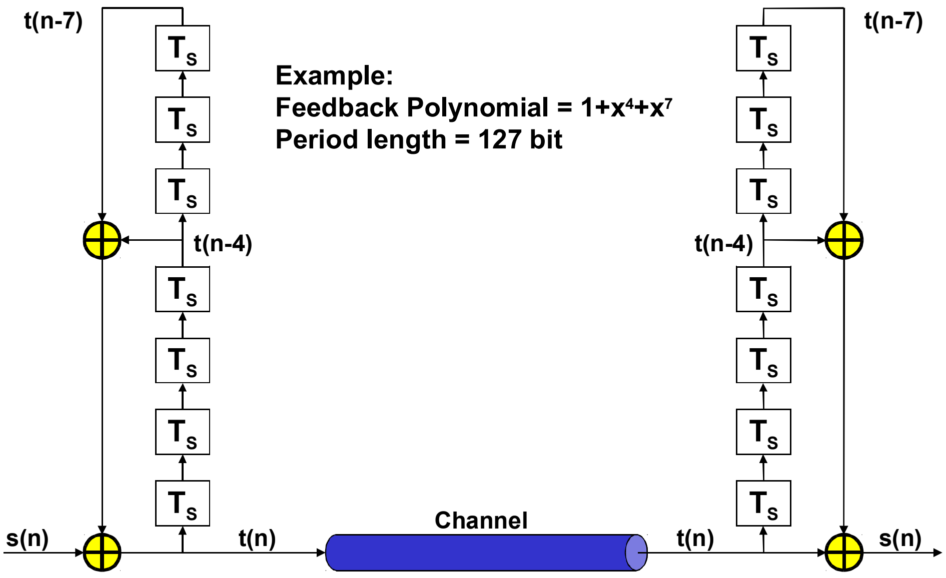
Power Spectrum Density



The slide above compares the power density distribution of some codes mentioned before. Obvously the code must match the spectral characteristics of the transmission channel.

Note that these codes are still kinds of baseband transmissions. Each one can be modulated using a carrier signal at higher frequency to comply to a specific channel characteristic.

Scrambling Example



Another method to guarantee frequent transitions is scrambling. Scramblers are used with ATM, SONET/SDH for example.

The feedback polynomial above can be written as

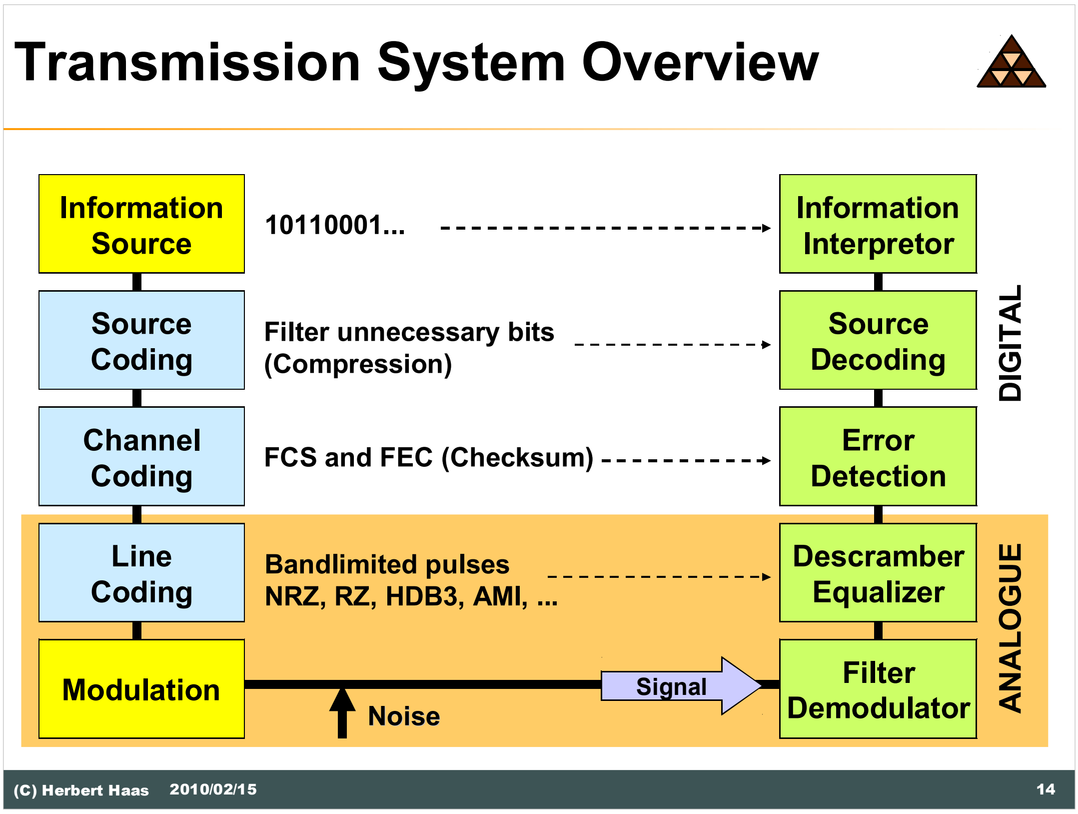
t(n) = s(n) XOR t(n-4) XOR t(n-7)

The descrambler recalculates the original pattern with the same function (change s(n) with t(n))

Period length = 2^R –1 , where R is the number of shift registers

That is, even a single 1 on the input (and all registers set to 0) will produce a 127-bit sequence of pseudo random pattern.

This scrambler is used with 802.11b (Wireless LAN).



Coding is not coding. The above slide gives you an overview about different coding purposes. Even modulation is sometimes called coding.

Source coding tries to eliminate redundancy within the information. Source coders must know well about the type of information that is delivered by the source.

Channel coding protects the non-redundant data stream by adding calculated overhead. Typically a Frame Check Sequence (FCS) is added. Only on very errourness and/or long-delay links a Forward Error Correction (FEC) method might be useful. FEC requires too much overhead in most terrestial applications.

Line coding focuses on the line, that is we want the symbols to be received correctly, even if noise and distortions are present. Furthermore line coding provides clock synchronization as discussed earlier.

Finally modulation might be necessary in case the channel has better properties at higher frequencies.

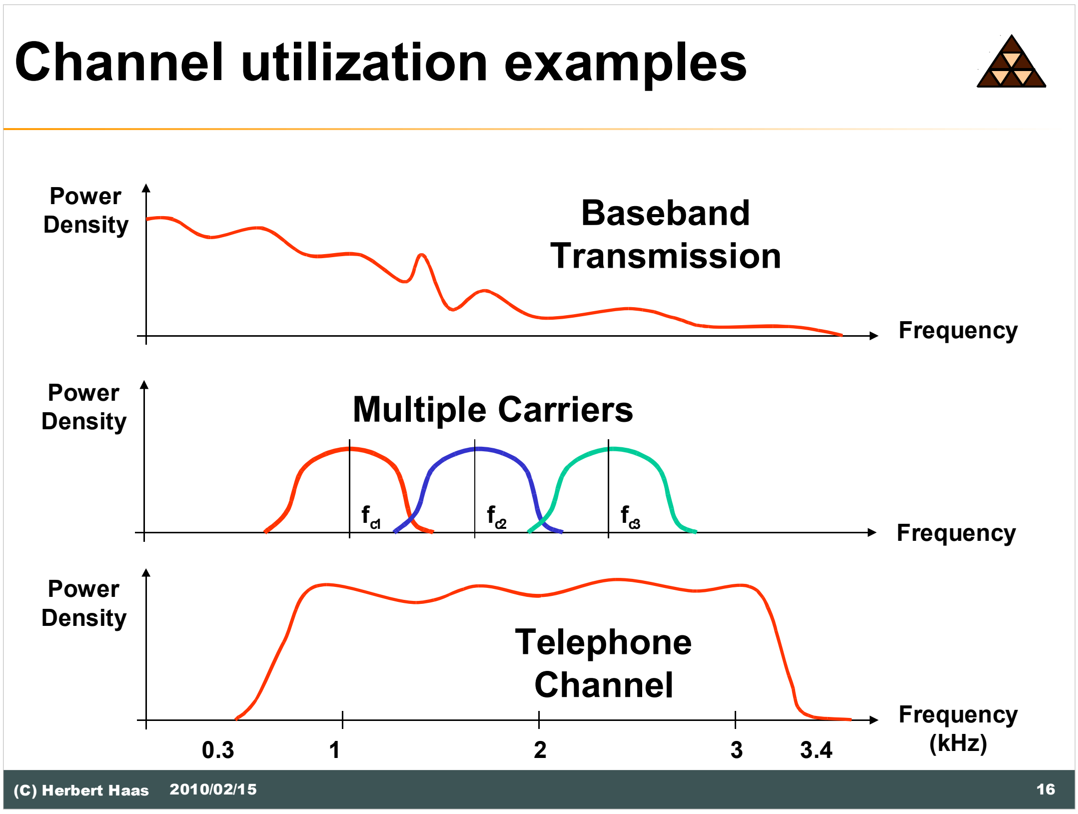
Communication Channels

- Usually *Low-Pass* behavior
  - Higher frequencies are more attenuated than lower
- Baseband transmission
  - Signal without a dedicated carrier
  - Example: LAN technologies (Ethernet etc)
- Carrierband transmission
  - The baseband signal modulates a carrier to match special channel properties
  - Medium can be shared for many users (different carriers) – e. g. WLAN

Each communication channel exhibits a low-pass behavior—at least beyond a very high frequency. Not only is the signal attenuated; phase shifts occur and even nonlinear effects sometimes rise with higher frequencies. The result is a smeared signal with little energy.

In most cases the signals do not need to be modulated onto a carrier. That is, all the channel bandwidth can be used up for this signal. We call this baseband transmission.

Carrier and transmission put the baseband signal onto a carrier with higher frequency. This is necessary with radio transmissions because low frequencies have a very bad radiation characteristic. Another example is fiber optics, where special signal frequencies are significantly more attenuated and scattered than others.

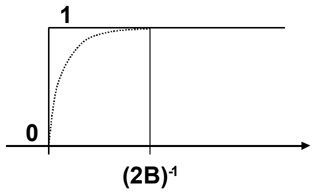


The above slide shows some examples for baseband and carrierband transmission. In case we use multiple carriers we may also call it broadband-transmission.

The third picture (bottom of slide) shows the spectral characteristic of a telephony channel (signal). The ITU-T defined an "attenuation-hose" in great detail (dynamics, ripples, edge frequencies, etc). As a rule of thumb we can expect low attenuation between 300 Hz and 3400 Hz.

Maximal Signal-Rate

- **Maximal data rate proportional to channel-bandwidth B**
  - **Raise time of Heavyside T=1/(2B)**
  - **So the maximum rate is R=2B, also called the Nyquist Rate**
  - **Note: We assume an ideal channel here – without noise!**
- **Bandwidth decreases with cable length**
  - **As a dirty rule of thumb: BW × Length ≅ const**
  - **But note that the reality is much more complex**
  - **Solitons are remarkable exceptions…**



Maximum signal rate: At least the amplitude must be reached

Since each channel is a low-pass, and some channels even damp (very) low frequencies, data can only be transmitted within a certain channel bandwidth B.

If we put a 0 to 1 transition on the line (with ideally zero transition time), the receiver will see a slope with a rise time of T=1/(2B).

So the maximal signal rate is T=1/(2B) – in theory. In practice we need some budget because there is noise and distortion and imperfect devices.

The longer the cable the more dramatically the low-pass behaviour. In other words: on the same cable type we can transmit (let's say) 1,000,000,000 bits/s if the cable is one meter in length, or only 1 bit/s if the cable is one million kilometers in length.

It is very interesting to mention that some modern fiber optic transmission methods violate this basic law. This methods base on so-called Soliton-Transmission.

The Maximum Information Rate

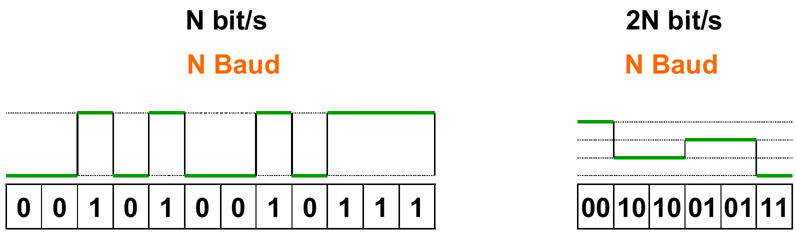
- What about a real channel? What's the maximum achievable information rate in presence of noise?
- Answer by C. E. Shannon in 1948
  - Even when noise is present, information can be transmitted ***without errors*** when the information rate is below the ***channel capacity***
- Channel capacity depends only on channel bandwidth AND SNR
  - Example: AWGN-channel

$$C = B \log (1 + S/N)$$

The great information theory guru Claude E. Shannon made a great discovery in 1948. Before 1948, it was commonly assumed, that there is no way to guarantee an error-less transmission over a noisy channel. However, Shannon showes that transmission without errors is possible when the information rate is below the so-called channel capacity, which depends on bandwidth and signal-to-noise ratio. This discovery is regarded as one of the most important achievements in communication theory.

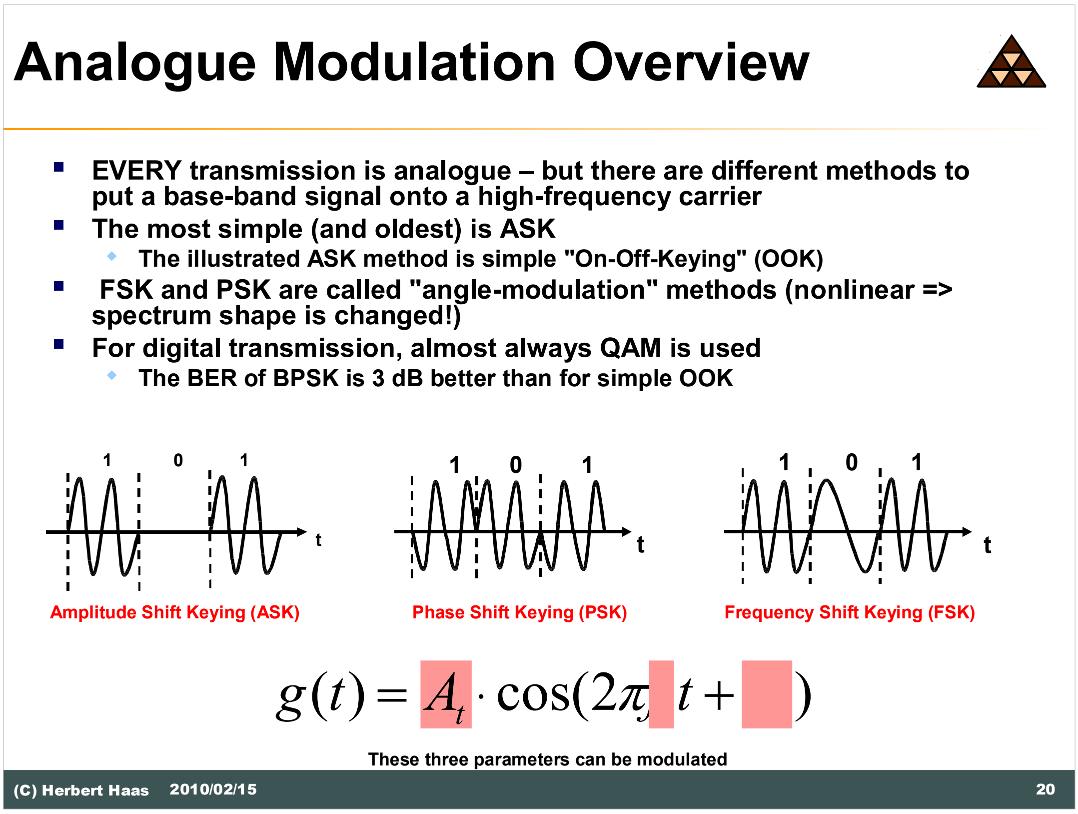
Bitrate vs Baud

- Information Rate: Bit/s
- Symbol Rate: Baud
- The goal is to send many (=as much as possible) bits per symbol
  - => QAM (see next slides)



Baud is named after the 19th centurey French inventor Baudot, originally referred to the speed a telegrapher could send Morse Code.

Today the symbol rate is measured in Baud whereas the information rate is measured in bit/s.



The slide shows a general modulation equation. The 3 parameters of the equation describe the 3 basic modulation types. All 3 parameters, the amplitude A_t, the frequency f_t and the phase φ_t can be varied, even simultaneously. In nature, there is no real digital transmission; the binary data stream needs to be converted into an analog signal. As first step, the digital data will be “directly” transformed into a analog signal (0 or 1), which is called a baseband signal. In order to utilize transmission media such as free space (or cables and fibers) the base signal must be mixed with a carrier signal. This analog modulation shifts the center frequency of the baseband signal to the carrier frequency to optimize the transmission for a given attenuation/propagation characteristic.

Amplitude Shift Keying

A binary 1 or 0 is represented through different amplitudes of a sinus oscillation. Amplitude Shift Keying (ASK) requires less bandwidth than FSK or PSK since *natura non facit saltus*. However ASK is interference prone. This modulation type also used with infrared-based WLAN.

Frequency Shift Keying

Frequency Shift Keying (FSK) is often used for wireless communication. Different logical signals are represented by different frequencies. This method needs more bandwidth but is more robust against interferences. To avoid phase jumps, FSK uses advanced frequency modulators (Continuous Phase Modulation, CPM).

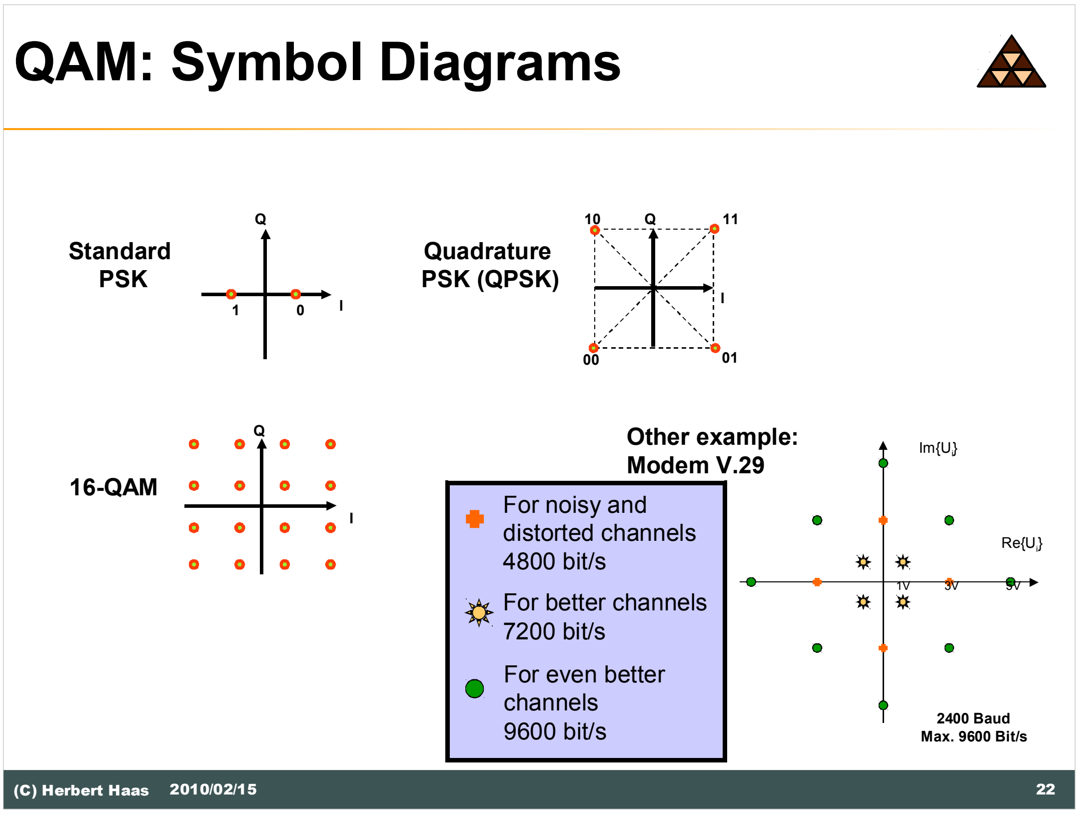
Phase Shift Keying

The 3rd basic modulation method is the Phase Shift Keying (PSK). The digital signal is coding through phase skipping. In the picture above you see the simplest variation of PSK, using phase jumps of 180°. In practice, to reduce BW, phase jumps must be minimized, and therefore PSK is implemented using advanced phase modulators (e. g. Gaussian Minimum Shift Keying, etc). The receiver must use same frequency and must be perfectly synchronized with the sender using a Phase Locked Loop (PLL) circuit. PSK is more robust as FSK against interferences, but needs complex devices.

After understanding these modulation methods QAM shall be introduced, which is the most important modulation scheme today – for both wired and wireless transmission lines.

QAM: Idea

- "Quadrature Amplitude Modulation"
- Idea:
  1. Separate bits in groups of words (e. g. of 6 bits in case of QAM-64)
  2. Assign a dedicated pair of Amplitude and phase to each word (A,φ)
  3. Create the complex amplitude $Ae^{j\varphi}$
  4. Create the signal $Re\{Ae^{j\varphi}e^{j\omega t}\}$
     $= A (\cos \varphi \cos \omega t - \sin \varphi \sin \omega t)$ which represents one (of the 64) QAM symbols
  5. Receiver can reconstruct (A,φ)



Worth to know: Simple Phase Shift Keying (PSK) which only uses two symbols, each representing either 0 and 1. Quadrature Phase Shift Keying (QPSK) with four symbols.

Usually the assignment of bit-words to symbols is such that the error probability due to noise is minimized. For example the Gray-Code may be used between adjacent symbols to minimize the number of wrong bits when an adjacent symbol is detected by the receiver.

The above slide also shows the symbol distribution over the complex plane for the V.29 protocol which is/was used by modems. Depending on the noise-power of the channel, different sets of symbols are used.

14,400 bit/s requires 64 points

28,800 bit/s requires 128 points

Example QAM Applications

- One symbol represents a bit pattern
  - Given N symbols, each represent ld(N) bits
- Modems, 1000BaseT (Gigabit Ethernet), WiMAX, GSM, …
- WLAN 802.11a and 802.11g:
  - BPSK @ 6 and 9 Mbps
  - QPSK @ 12 and 18 Mbps
  - 16-QAM @ 24 and 36 Mbps
  - 64-QAM @ 48 and 54 Mbps

 23

It is important to understand that spread spectrum (or OFDM) techniques are always combined with a symbol modulation scheme. Quadrature Amplitude Modulation (QAM) is a general method where practical methods such as BPSK, QPSK, etc are derived from.

The main idea of QAM is to combine phase and amplitude shift keying. Since orthogonal functions (sine and cosine) are used as carriers, they can be modulated separately, combined into a single signal, and (due to the orthogonality property) de-combined by the receiver.

And since A*cos(wt + phi) = A/2{cos(wt)cos(phi) – sin(wt)sin(phi)} QAM can be easily represented in the complex domain as Real{ A*exp(i*phi)*exp(i*wt)}.

The standard PSK method only use phase jumps of 0° or 180° to describe a binary 0 or 1. In the right picture above you see a enhanced PSK method, the Quadrature PSK (QPSK) method. While using Quadrature PSK each condition (phase shift) represent 2 bits instead of 1. Now it is possible to transfer the same datarate by halved bandwidth.

The QSK signal uses (relative to reference signal)

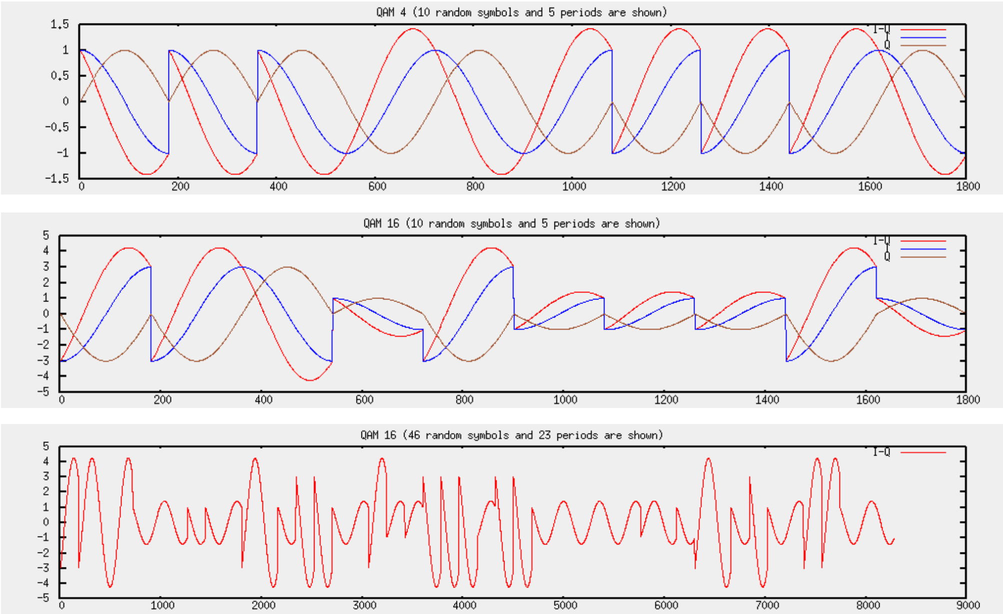
- 45° for a data value of 11
- 135° for a data value of 10
- 225° for a data value of 00
- 315° for a data value of 01

To reconstruct the original data stream the receiver need to compare the incoming signal with the reference signal. The synchronization is very important.

Why not coding more bits per phase jump ?

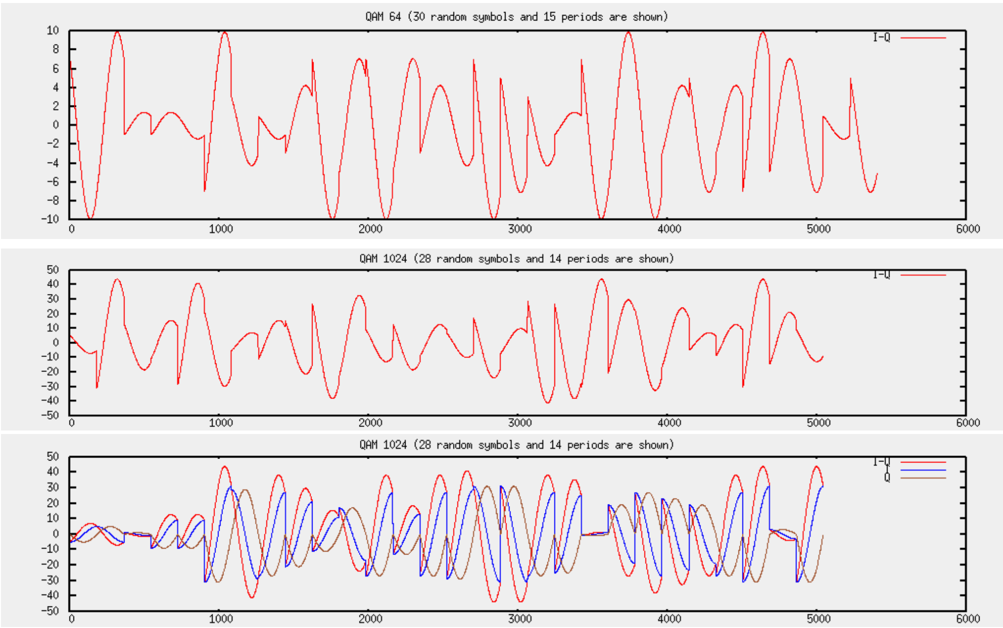
Especial in the mobile communication there are to much interferences and noise to encode right. As more bits you use per phase jump, the signal gets more "closer". It is getting impossible to reconstruct the original data stream. In the wireless communication the QPSK method has proven as a robust and efficient technique.

QAM Example Symbols (1)



Note that the above QAM signals show different successive QAM-symbols for illustration purposes. In reality each symbol is transmitted many hundred/thousand times

QAM Example Symbols (2)



Note that the above QAM signals show different successive QAM-symbols for illustration purposes. In reality each symbol is transmitted many hundred/thousand times

These diagrams have been generated using Octave, a free Matlab clone.

“The biggest problem with communication is the illusion that it has occured.”

Married?