Facoltà di Scienze Matematiche, Fisiche e Naturali
Dipartimento di Scienze dell'Informazione
Corso di Laurea Specialistica in Scienze di Internet (SdI) e Informatica (Inf)

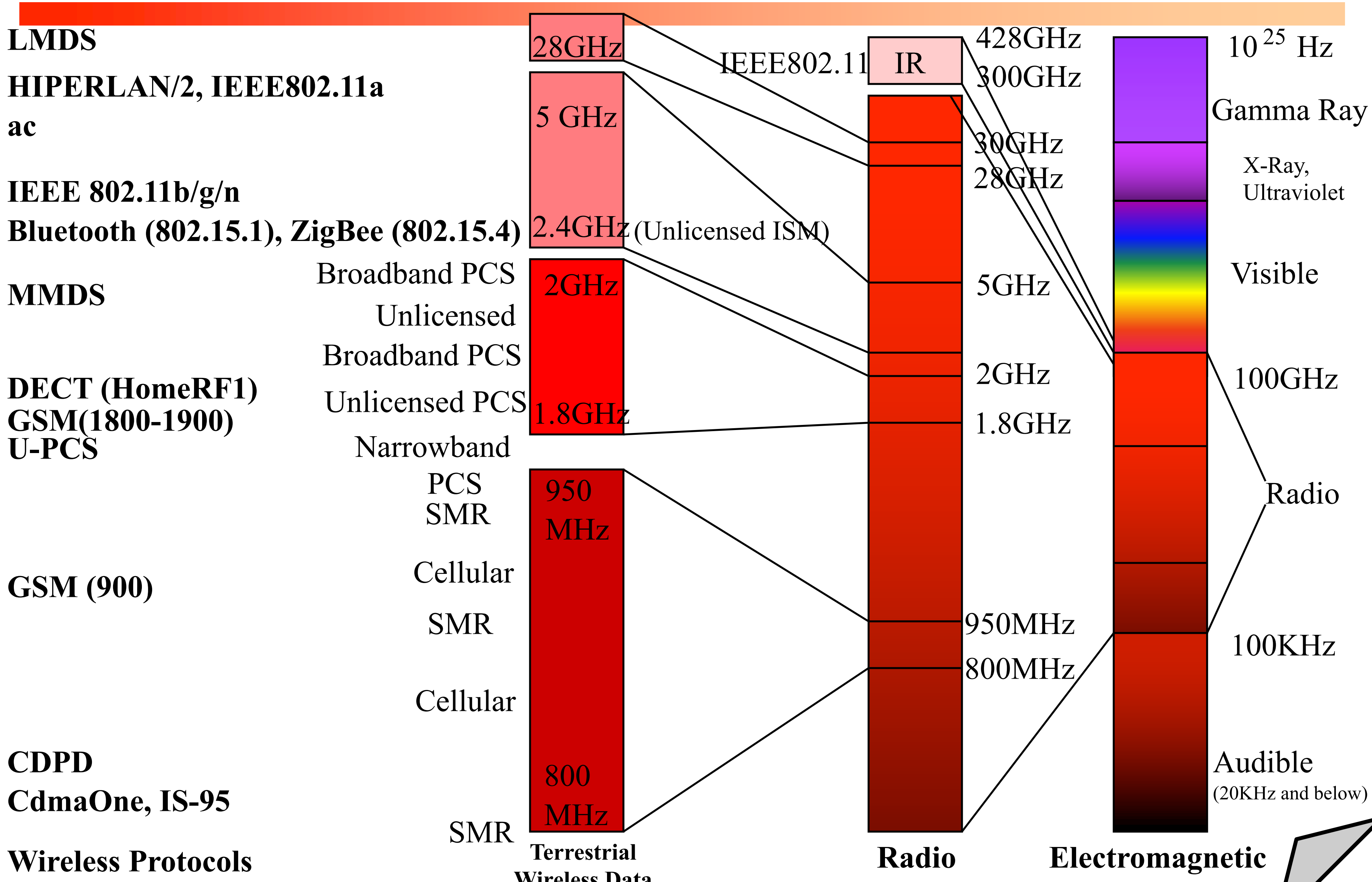
Wireless Systems (2) – Physical Spectrum, Logical Channels, Digital Modulation

Luciano Bononi
(bononi@cs.unibo.it)
http://www.cs.unibo.it/~bononi/

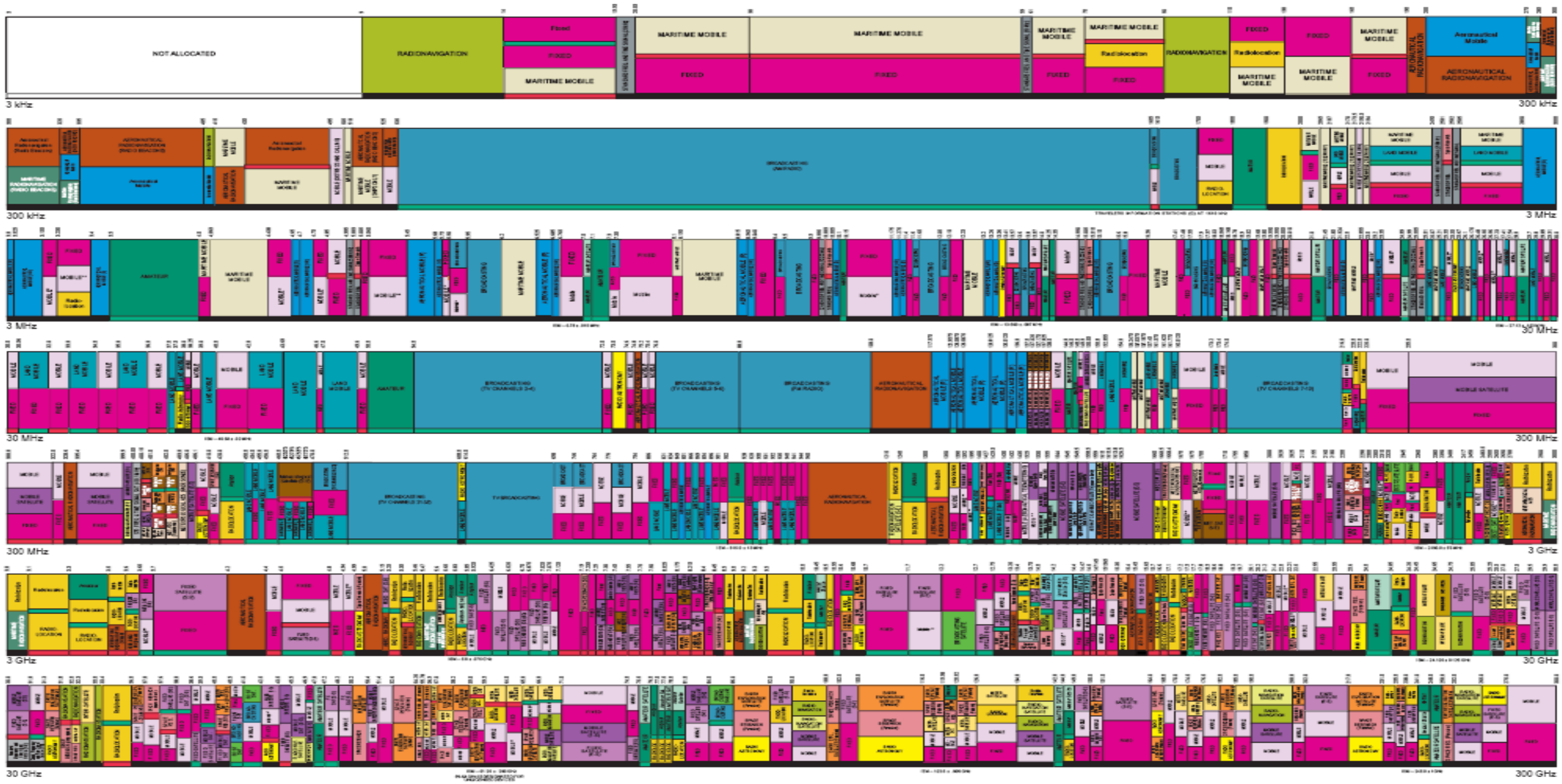
Ricevimento: sempre aperto .
Si consiglia di concordare via e-mail almeno un giorno prima
(informazioni in tempo reale sulla home page personale)

Figure-credits: some figures have been taken from slides published on the Web, by the following authors (in alfabethical order):
J.J. Garcia Luna Aceves (ucsc), James F. Kurose & Keith W. Ross, Jochen Schiller (fub), Nitin Vaidya (uiuc)

Wireless networks' spectrum



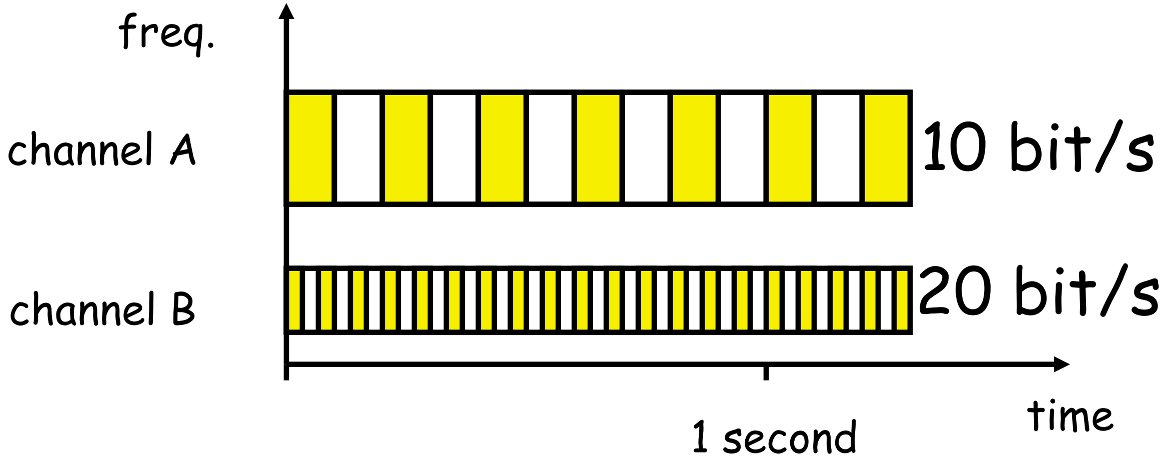
Fixed spectrum assignment



Slide credits: IFA'2007, prof. Ian Akyildiz @ Gtech

Wireless networks Bandwidth and Spectrum

- **how can wireless channels have different bandwidth?**
  - bits run less or more faster? (NO)
    - Light speed: ˜ <300.000 Km/s for every bit
  - the channel pipe (spectrum) is bigger (YES/NO)
  - the channel requires less time to accomodate (i.e. to code) one bit on the channel (YES)

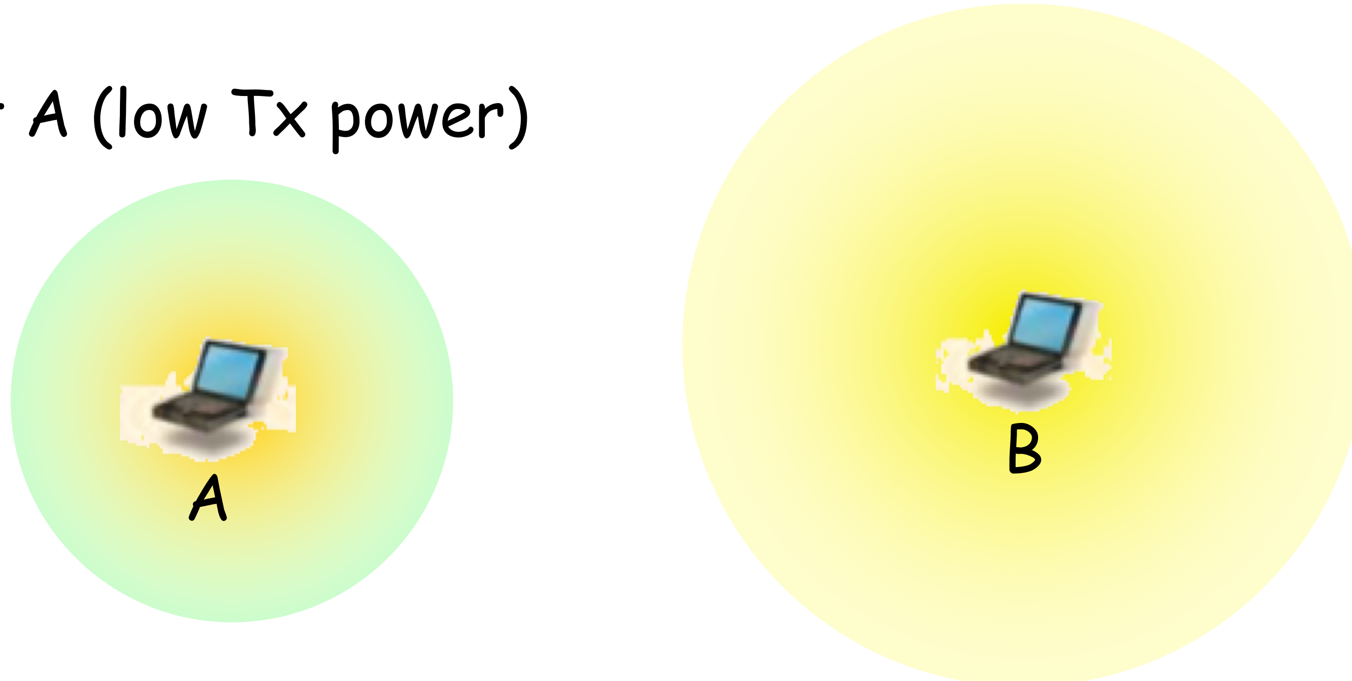


Wireless networks' technology

- **Radio transmission coverage**

host B (high Tx power)

host A (low Tx power)

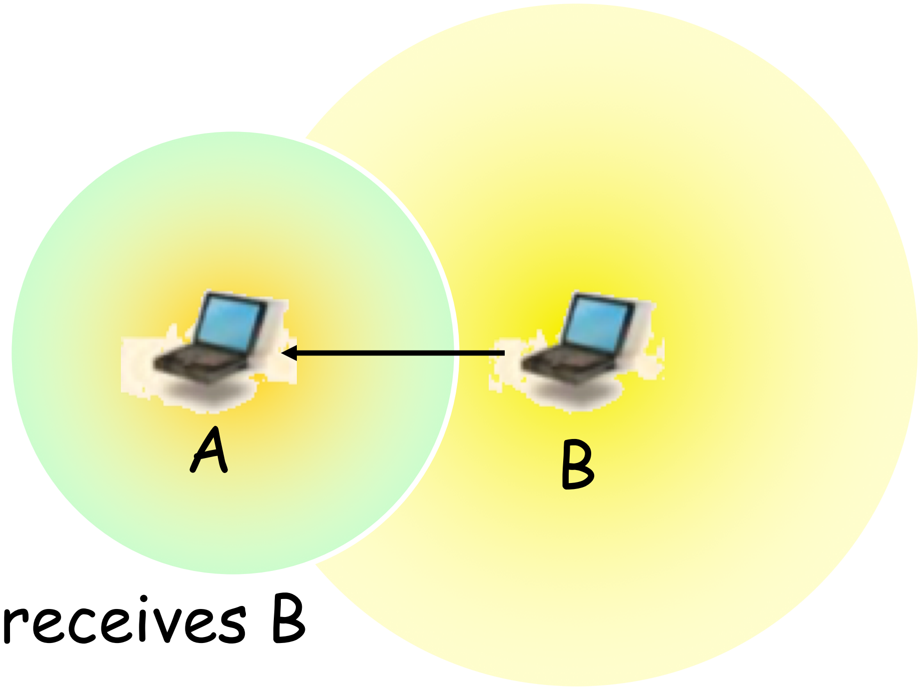


"...is there anybody outhere?"

both isolated

Wireless networks' technology

- **Radio transmission coverage**



A receives B

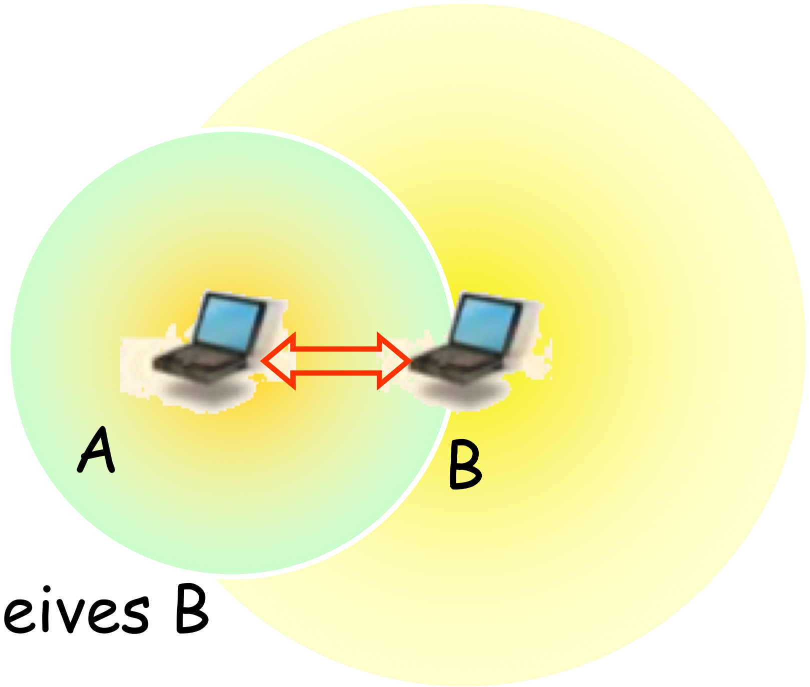
B cannot receive A

unidirectional(*) link

(*) sometimes improperly referred to as "asymmetric link"

Wireless networks' technology

- **Radio transmission coverage**



A receives B

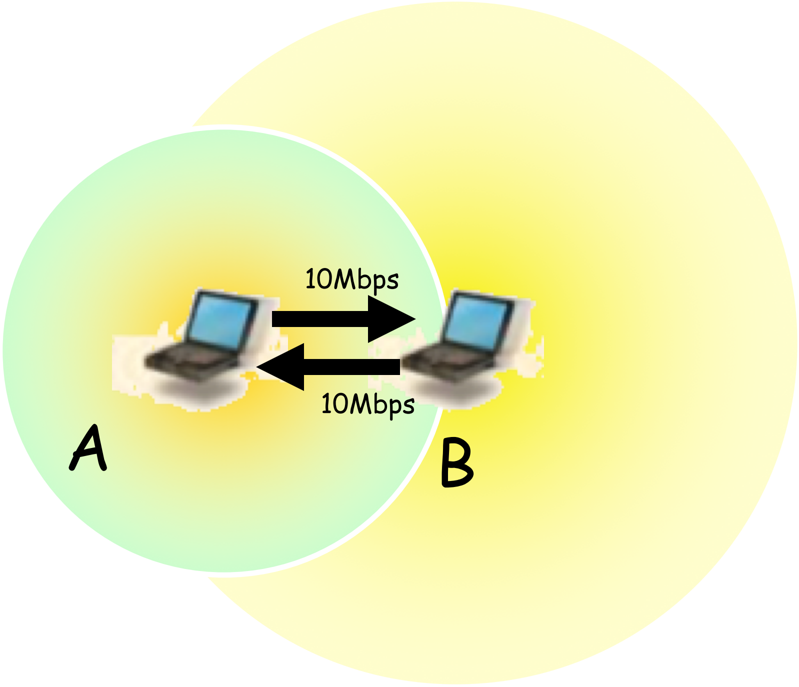
B receives A

bidirectional(*) link

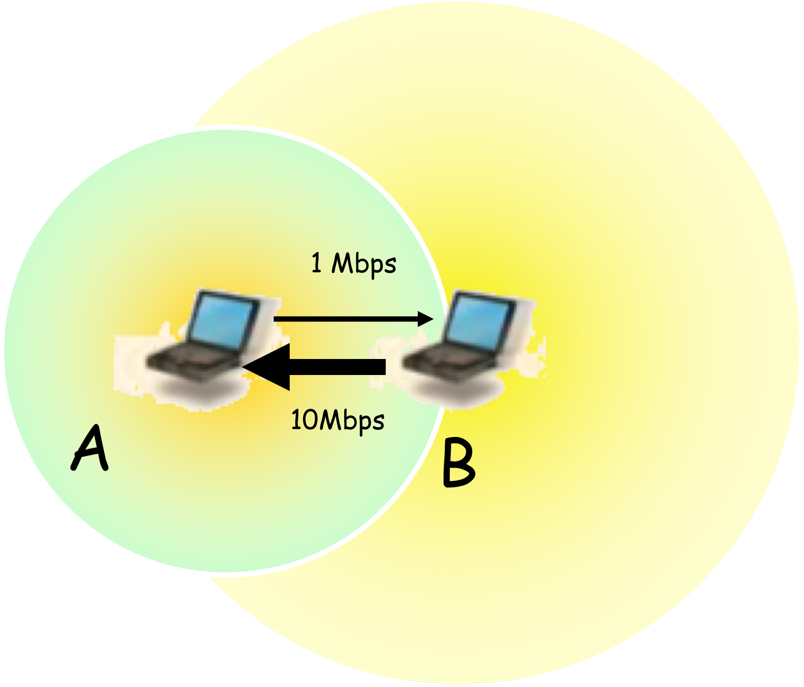
(*) sometimes improperly referred to as "symmetric link"

Wireless networks' technology

- **Radio transmission coverage**



bidirectional symmetric link



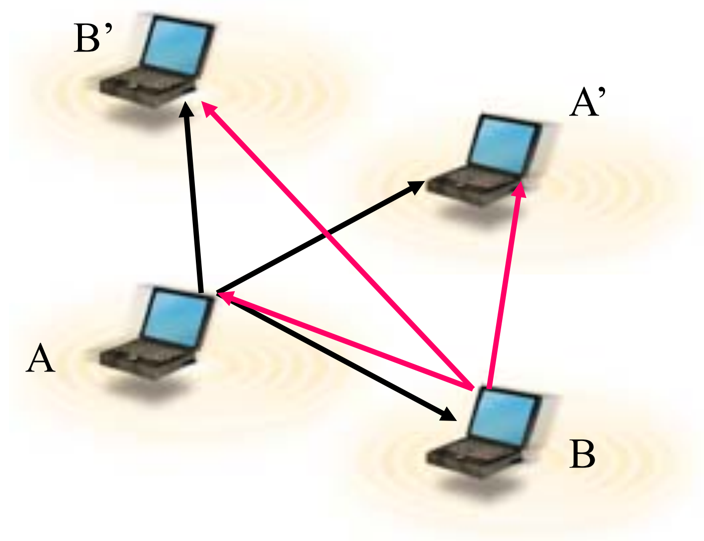
bidirectional asymmetric link

Wireless networks' technology

- **Narrowband radio system**
    - transmit/receive using a single radio frequency
- **Spread Spectrum technology**
    - bandwidth efficiency vs. reliability and security
  - Frequency Hopping Spread Spectrum
    - narrowband carrier hopping in a pattern sequence
  - Direct Sequence Spread Spectrum
    - bit coding and transmission spreading over the spectrum
- **Infrared technology**
    - line of sight or diffused, short range (in room)

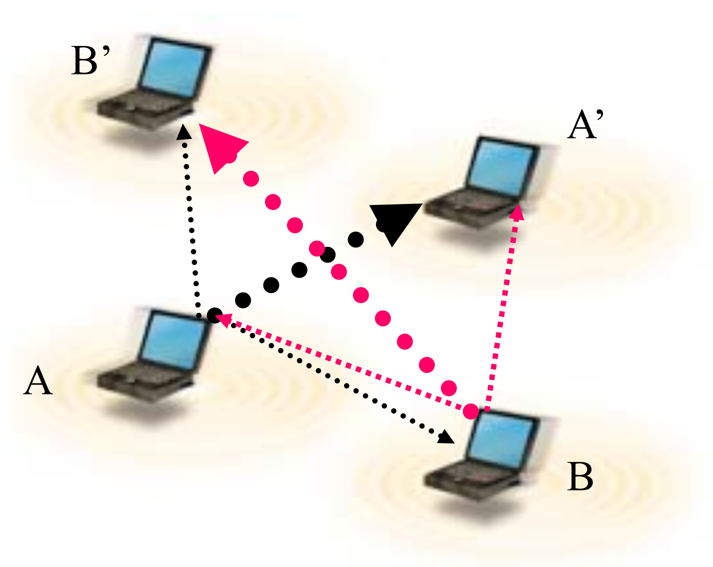
Wireless networks' technology

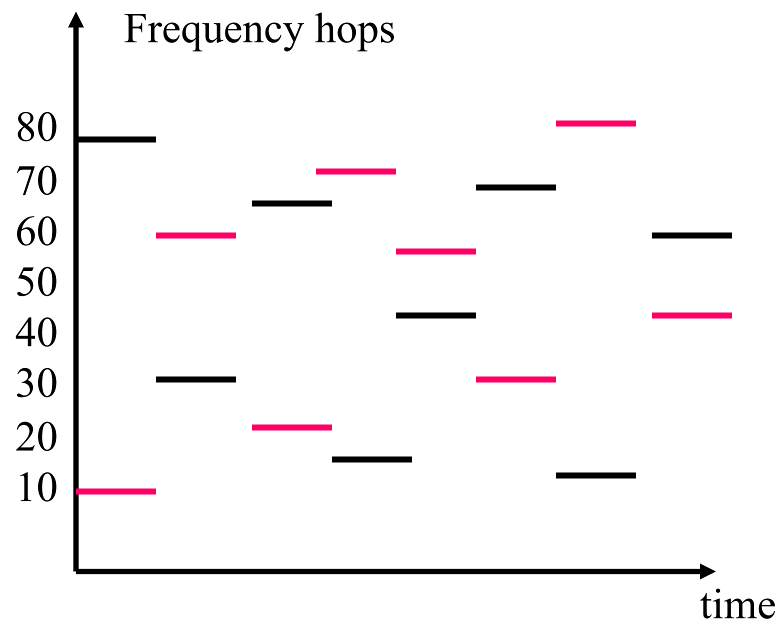
- **Narrowband radio system**
    - transmit/receive using a single, licensed, as narrow as possible radio frequency
    - undesired cross-talk between channels requires coordination and license for each site
    - low data-rates
    - e.g. ⟶ frequency X
    - e.g. ⟶ frequency Y



Wireless networks' technology

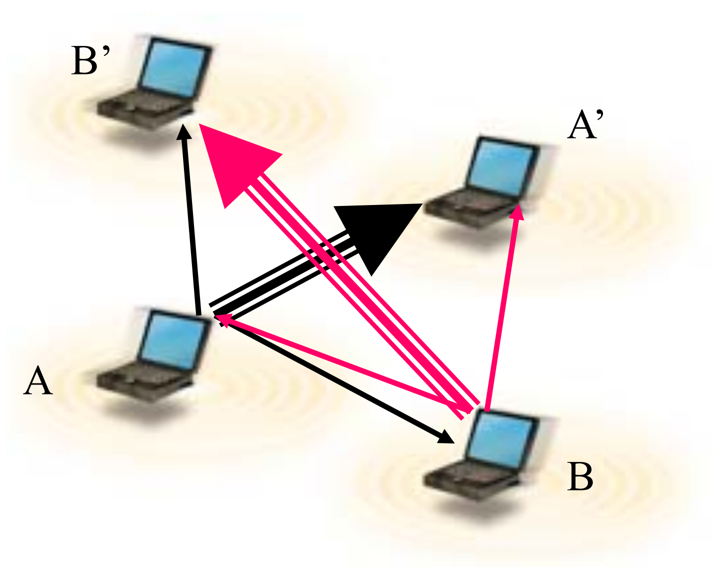
- **Frequency Hopping Spread Spectrum**
  - narrow band carrier changes frequency in a pattern known by both transmitter and receiver (single logical channel)
  - to unintended receiver FHSS appears as impulse noise

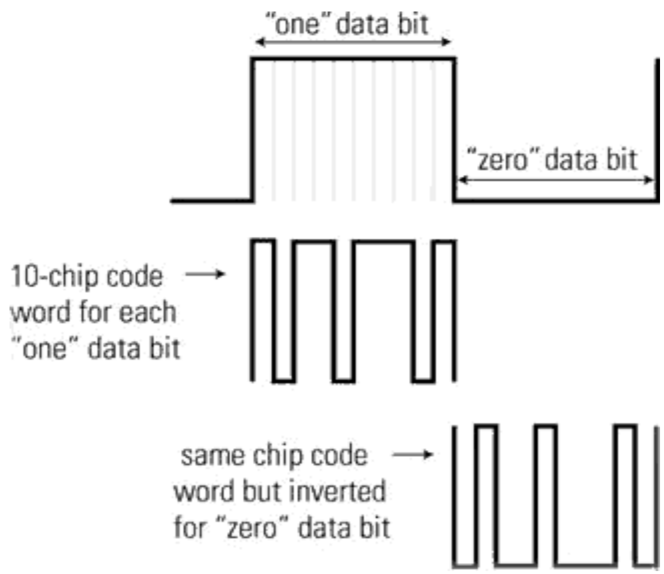




Wireless networks' technology

- **Direct Sequence Spread Spectrum**
  - redundant bit pattern (chipping code) spreaded over a large spectrum. Long chips increase probability of recovering the original bit (with no retransmission)
  - to unintended receiver DSSS appears as low power wideband noise

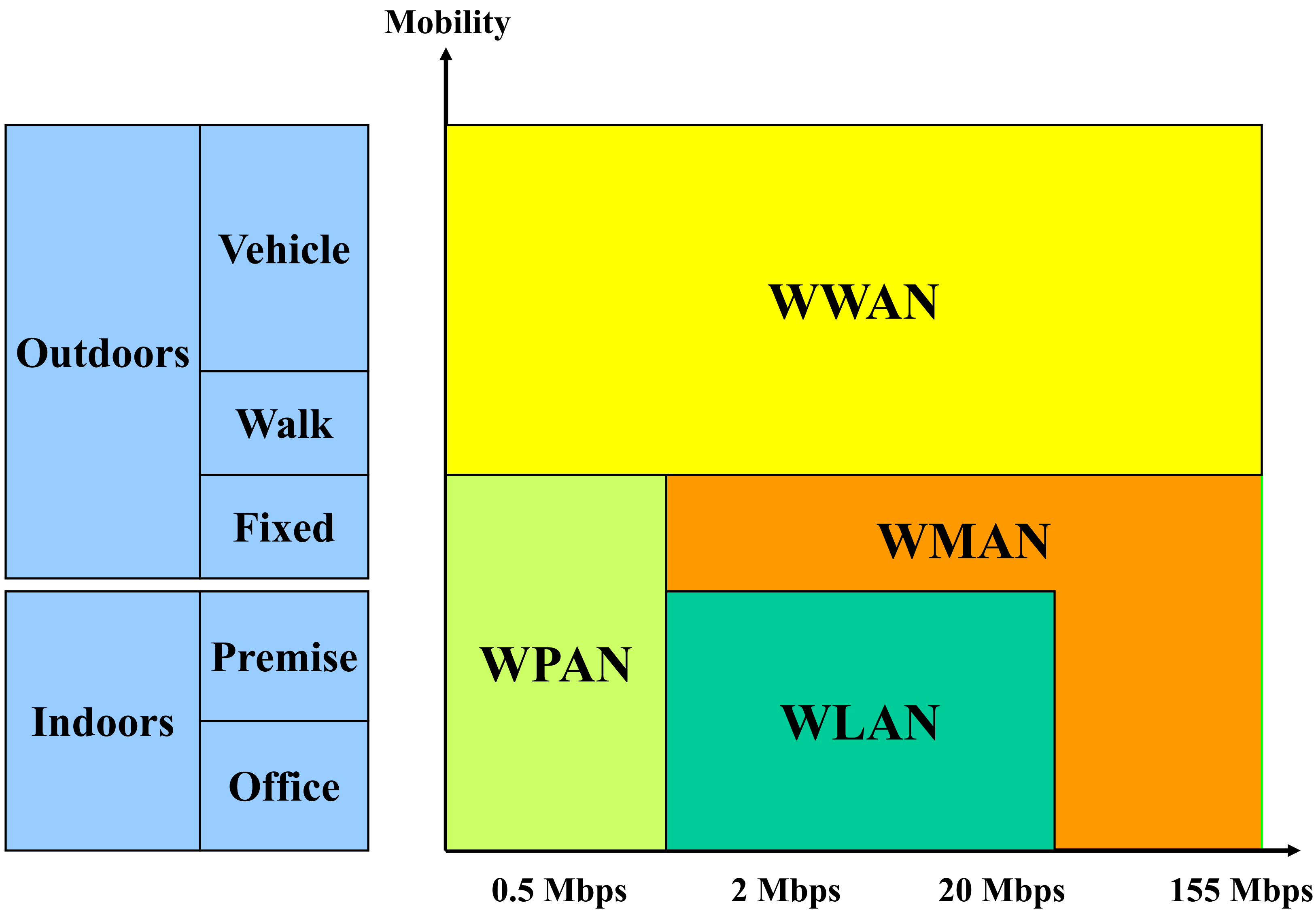




Wireless networks’ coverage classification

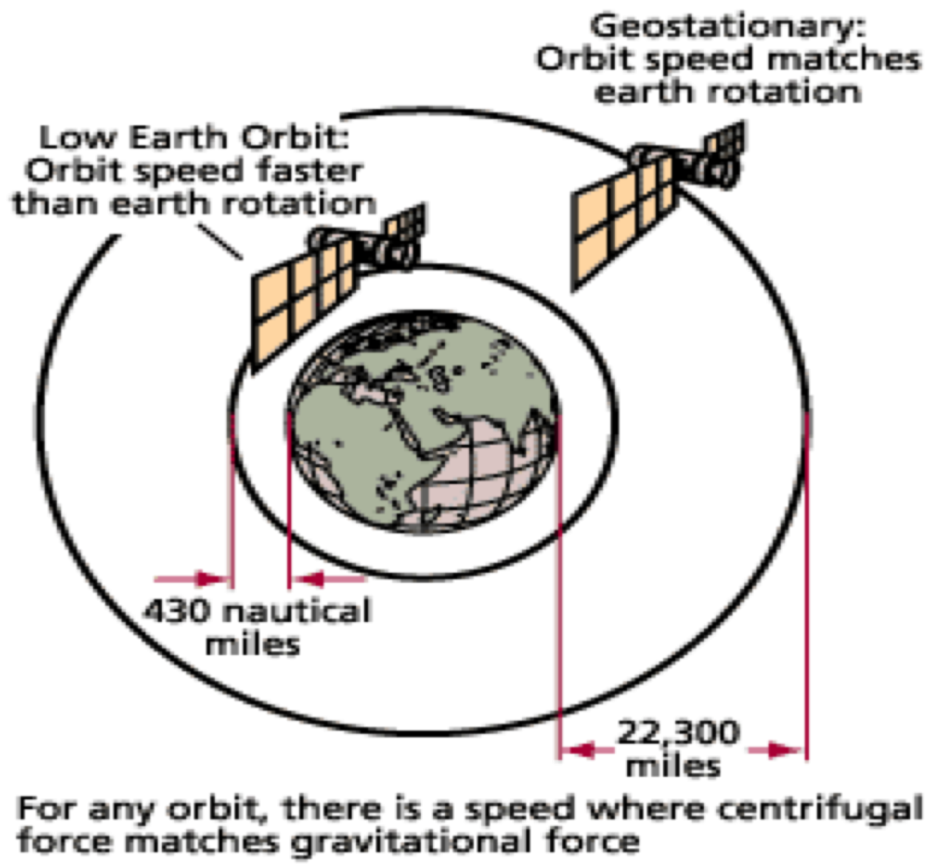
- **Wireless Wide Area Network (WWAN)**
  - geographic coverage (e.g. satellite, cellular)
- **Wireless Metropolitan Area Net. (WMAN)**
  - Metropolitan coverage (e.g. town, large campus)
- **Wireless Local Area Network (WLAN)**
  - local area coverage (e.g. campus, building, home)
- **Wireless Personal Area Network (WPAN)**
  - reduced local area coverage (e.g. house, office)
- **Wireless Indoor Area Network (indoor)**
  - short range coverage (e.g. room, office)

Wireless network positioning

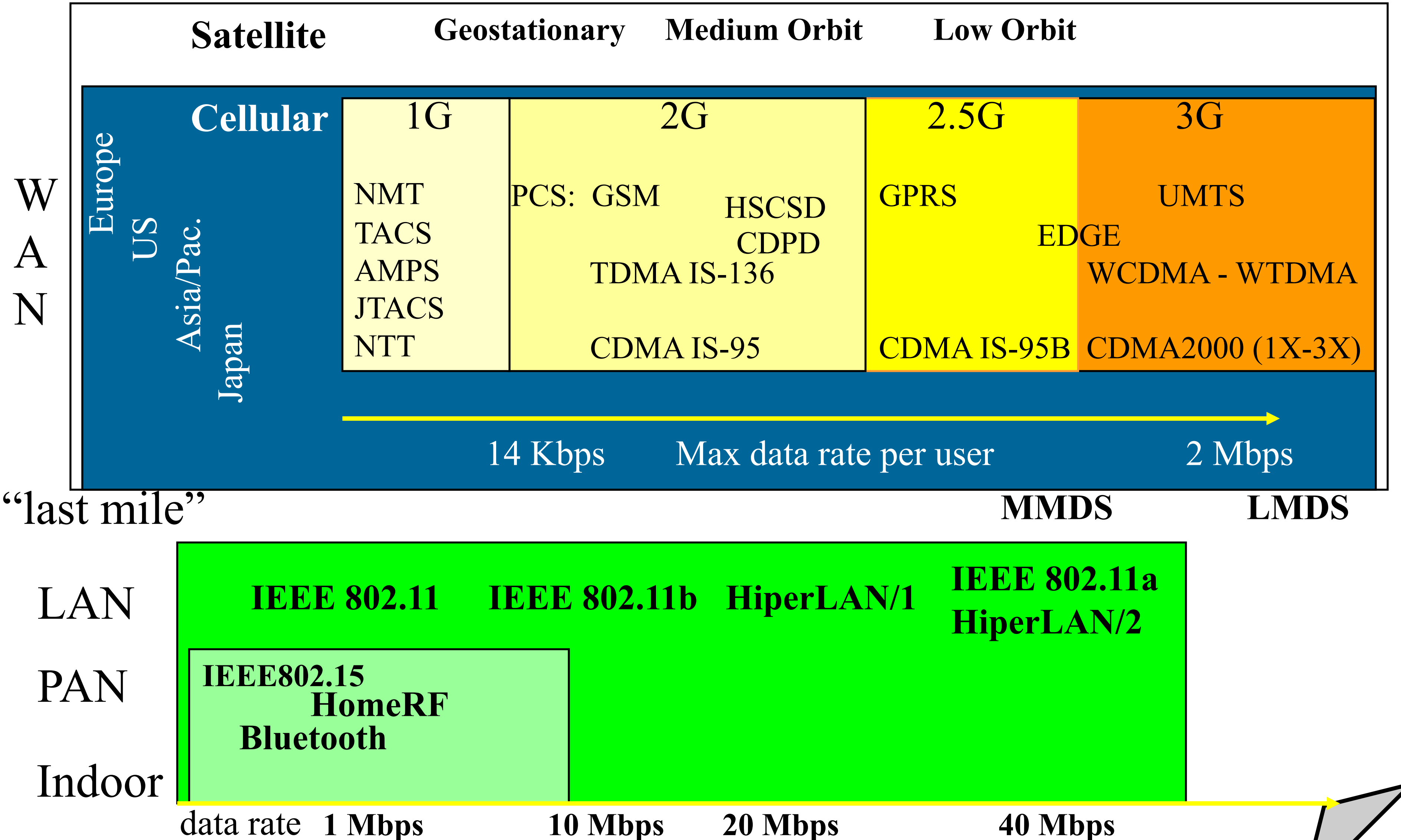


Wireless network structures

- **WWAN and WMAN**
  - Satellite (low orbit, geo-stationary)

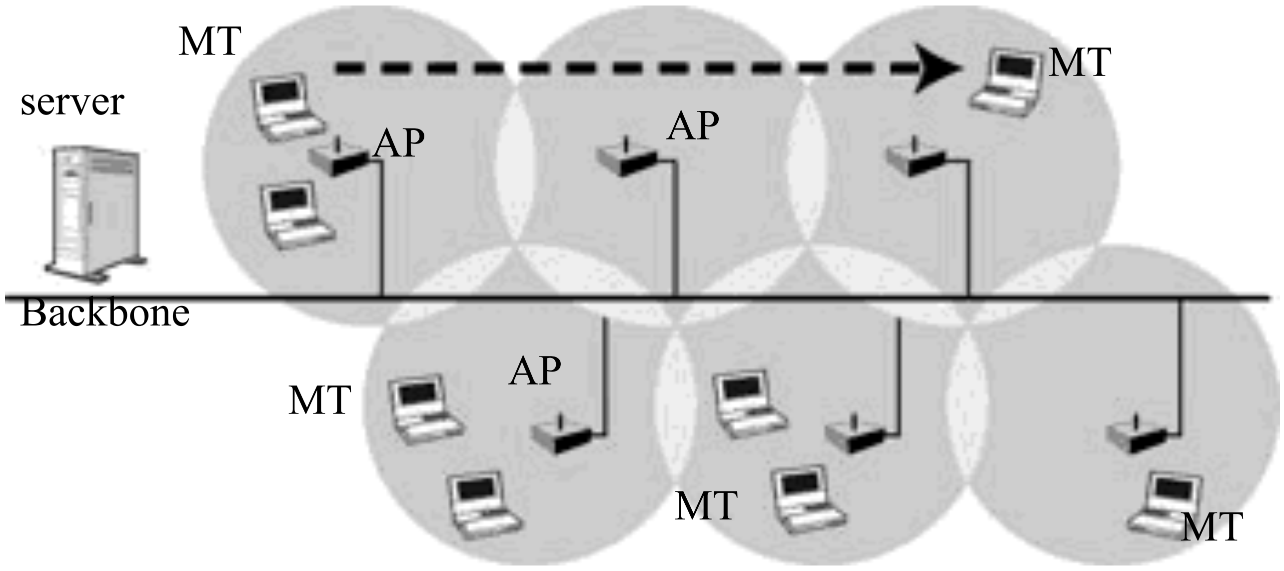


Wireless networks' structure (taxonomy)



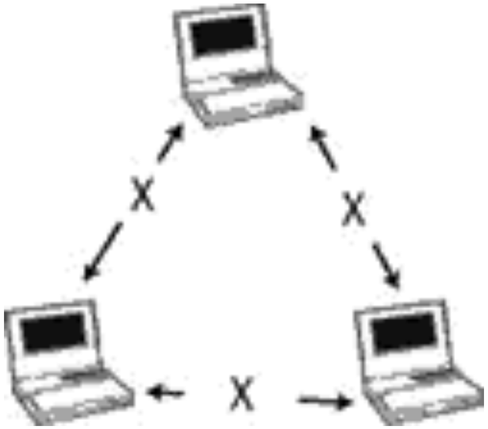
Wireless network structures

- **WWAN and WMAN**
  - Cellular or multi-Infrastructure WLAN
    - grid of Access Points (AP), managing local Mobiles terminals (MT), and connected to Backbones

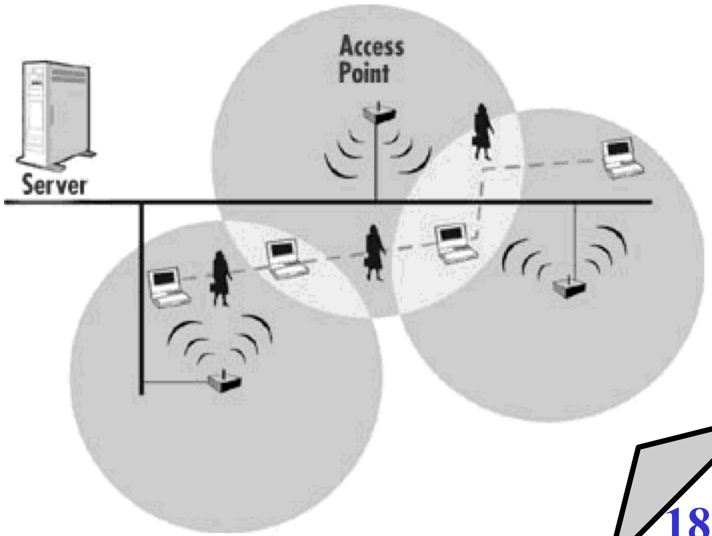


Wireless network structures

- **WLAN:**
  - Ad Hoc:
    - peer-to-peer (P2P) “on the fly” communication
    - the network “is” the set of computers
    - no administration, no setup, no cost?

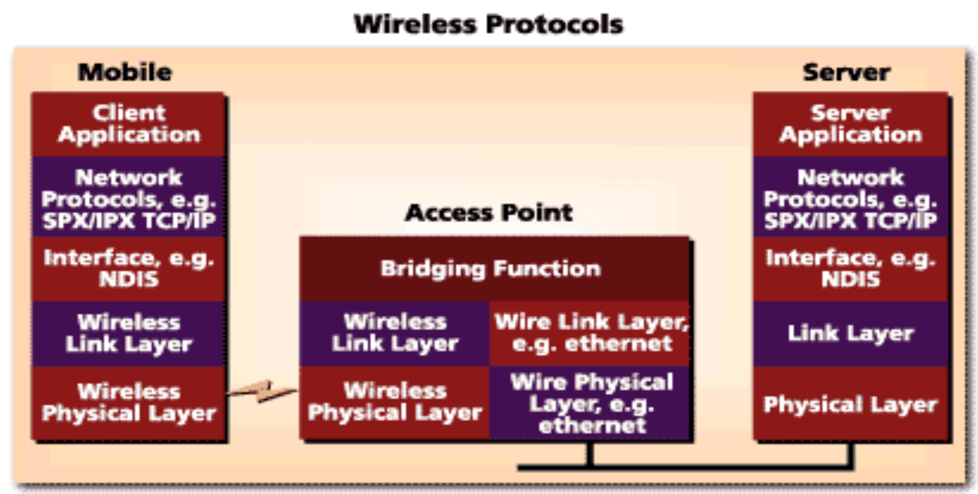


  - Infrastructure:
    - Centralized control unit (Access Point, local server)
    - Roaming between cells
    - resource sharing and backbone connection



Wireless/Wired extension

- **Wireless protocols' design, integration, optimization**
  - layering, bridging functions
  - mobile IP
  - support and management for QoS
- **support for Wired-like applications**
  - Internet connectivity, DB access, e-mail
  - value added services



Wireless World integration

- **One possible solution for Integraton with wired world:**
  - to uncouple wired and wireless networks
  - protocol integration, maintaining services and protocols view from both sides
  - protocols and SW structures to adapt the contents transferred to etherogeneous devices
  - adaptive behavior of network protocols (from the wireless side)
    - the wired host does not know if the other host is wireless and dialogue with it in the standard wireless way (protocol transparency)
    - the wireless host know it is wireless and implements adaptive behavior

Wireless drawbacks

- **reduced Channel Capacity (1 or 2 order of magnitude)**
  - e.g. 54 Mbps vs. Gigabit Ethernet
- **Limited spectrum (etherogeneous frequency windows) available**
  - need for international frequency-allocation plans
  - need for frequency reuse
- **Limited energy (batteries): +20% every 5 years**
  - Moore law: SoC transistors double every year
- **Noise and Interference have great impact on performances and system design**
  - need for high power, bit error correction
- **Security: sensible information travels “on the air”**
  - need for protection based on cyphering, authentication, etc.

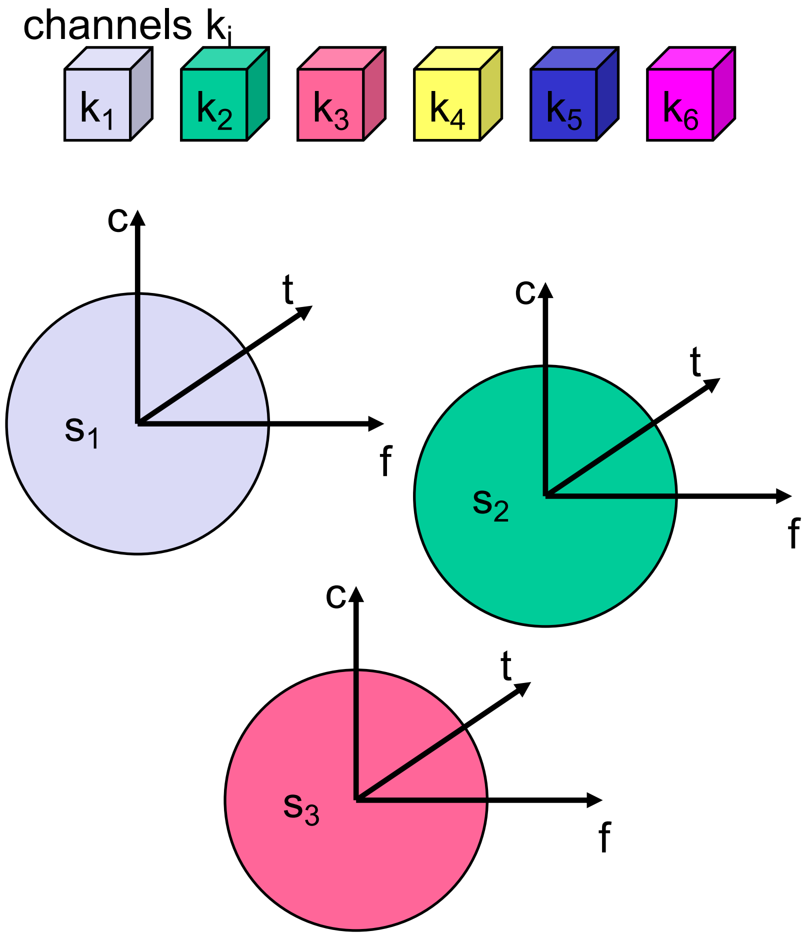
Wireless drawbacks

- Mobility management
  - addressing and routing (eg. Mobile IP)
- Location Tracking
  - Broadcasting (paging) to find users/hosts
  - support for Location Based Services
- QoS Management
  - not a single layer management (application, transport, network, MAC)
  - depends on the system/user/application scenario
  - managed for the wireless cell only (no multi-hop)
  - advance reservation, admission control policies (centralized, distributed)
  - scheduling (centralized, distributed) for resources' allocation
- Best effort services

Logical wireless channel

Multiplexing: multiple use of shared medium

- **Multiplexing in 4 dimensions**
  - space (s_i)
  - time (t)
  - frequency (f)
  - code (c)

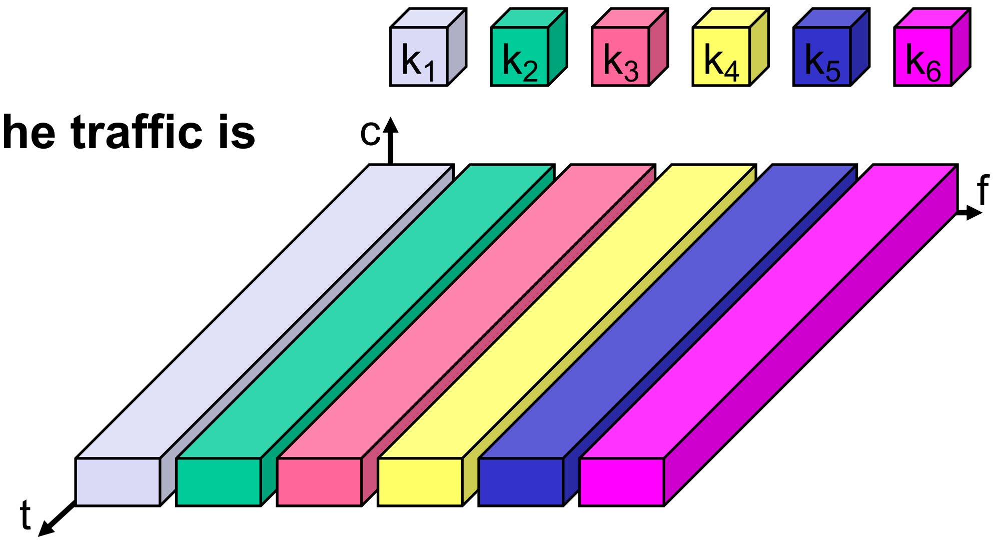


- **Goal: multiple use of a shared medium**

- **Important: guard spaces needed!**

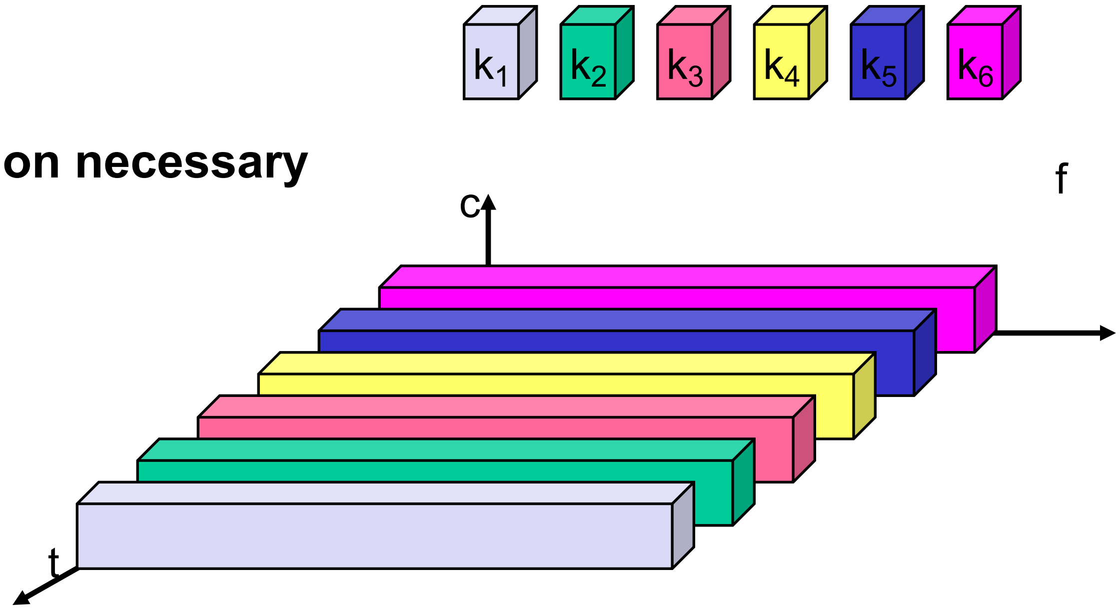
Frequency multiplex

- Separation of the whole spectrum into smaller frequency bands
- A channel gets a certain band of the spectrum for the whole time
- Advantages:
  - no dynamic coordination necessary
  - works also for analog signals
- Disadvantages:
  - waste of bandwidth if the traffic is distributed unevenly
  - inflexible
  - guard spaces



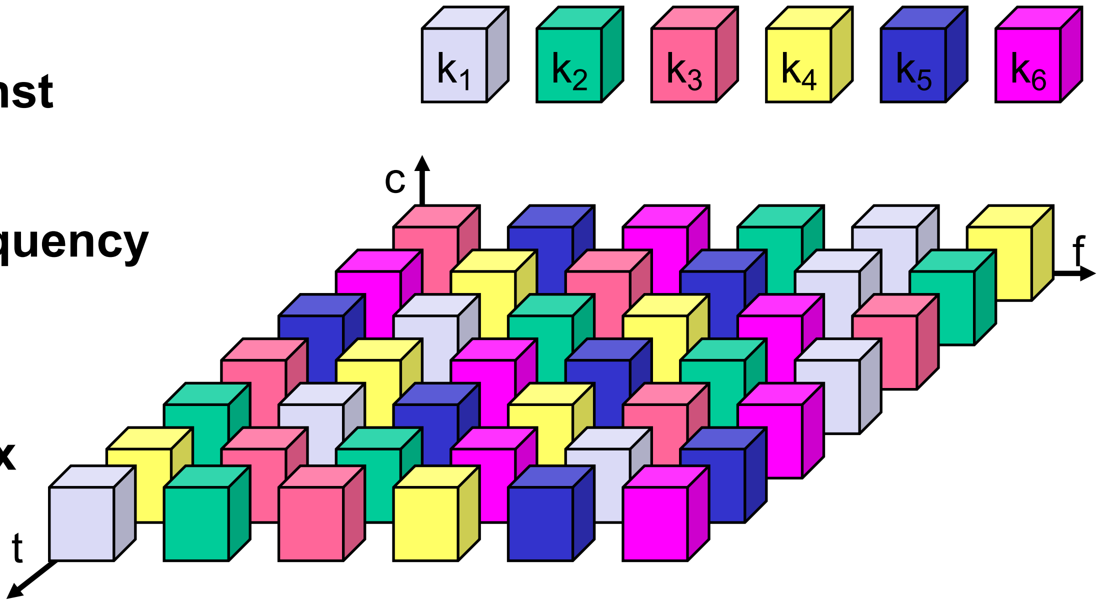
Time multiplex

- **A channel gets the whole spectrum for a certain amount of time**
- **Advantages:**
  - **only one carrier in the medium at any time**
  - **throughput high even for many users**
- **Disadvantages:**
  - **precise synchronization necessary**



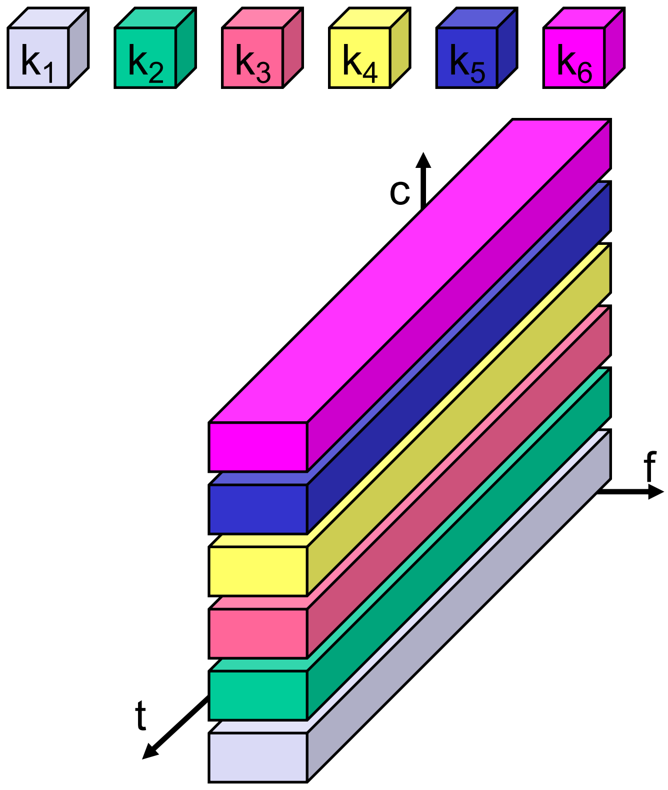
Time and frequency multiplex

- **Combination of both methods**
- **A channel gets a certain frequency band for a certain amount of time**
  - **Example: GSM**
- **Advantages:**
  - **better protection against tapping**
  - **protection against frequency selective interference**
  - **higher data rates compared to code mux**
- **but:**
  - **precise coordination required**



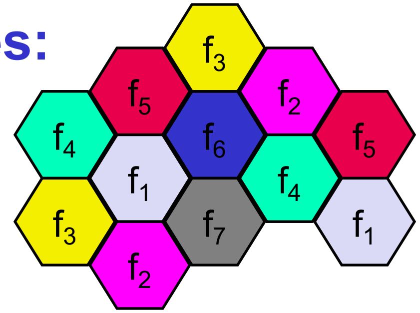
Code multiplex

- **Each channel has a unique code**
- **All channels use the same spectrum at the same time**
- **Advantages:**
  - **bandwidth efficient**
  - **no coordination and synchronization necessary**
  - **good protection against interference and tapping**
- **Disadvantages:**
  - **lower user data rates**
  - **more complex signal regeneration (€)**
- **Implemented using spread spectrum technology**



Frequency planning I

- **Frequency reuse only with a certain distance between the base stations**
- **Standard model using 7 frequencies:**



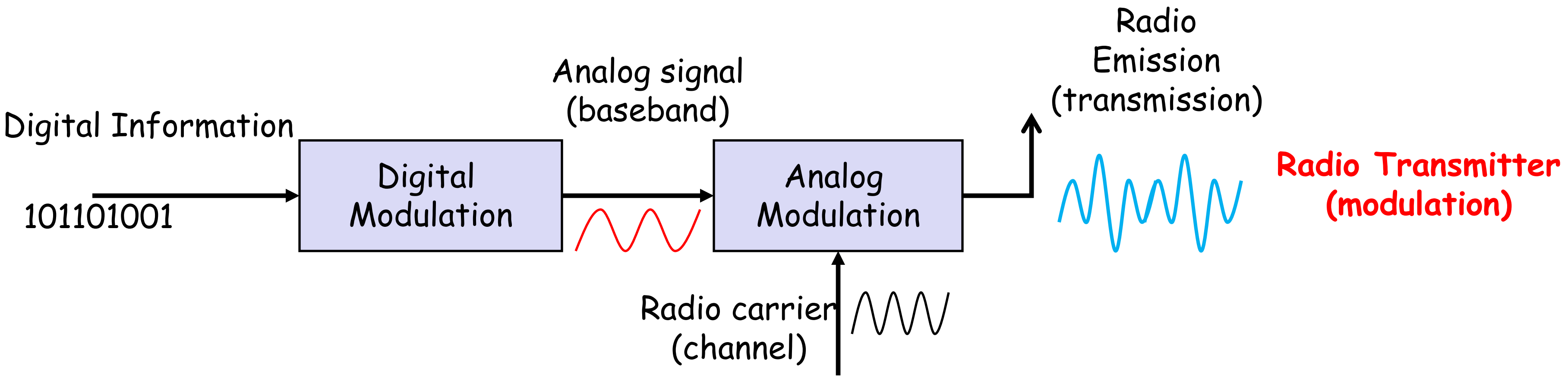
- **Fixed frequency assignment:**
  - certain frequencies are assigned to a certain cell
  - problem: different traffic load in different cells
- **Dynamic frequency assignment:**
  - base station chooses frequencies depending on the frequencies already used in neighbor cells
  - more capacity in cells with more traffic
  - assignment can also be based on interference measurements

Modulation

- **Digital modulation**
  - digital data is translated into an analog signal (baseband)
  - ASK, FSK, PSK differences in spectral efficiency, power efficiency, robustness
- **Analog modulation**
  - shifts center frequency of baseband signal up to the radio carrier (i.e. FM)
- **Motivation**
  - smaller antennas (e.g., $\lambda/4$)
  - Frequency Division Multiplexing
  - medium characteristics
- **Basic schemes**
  - Amplitude Modulation (AM)
  - Frequency Modulation (FM)
  - Phase Modulation (PM)

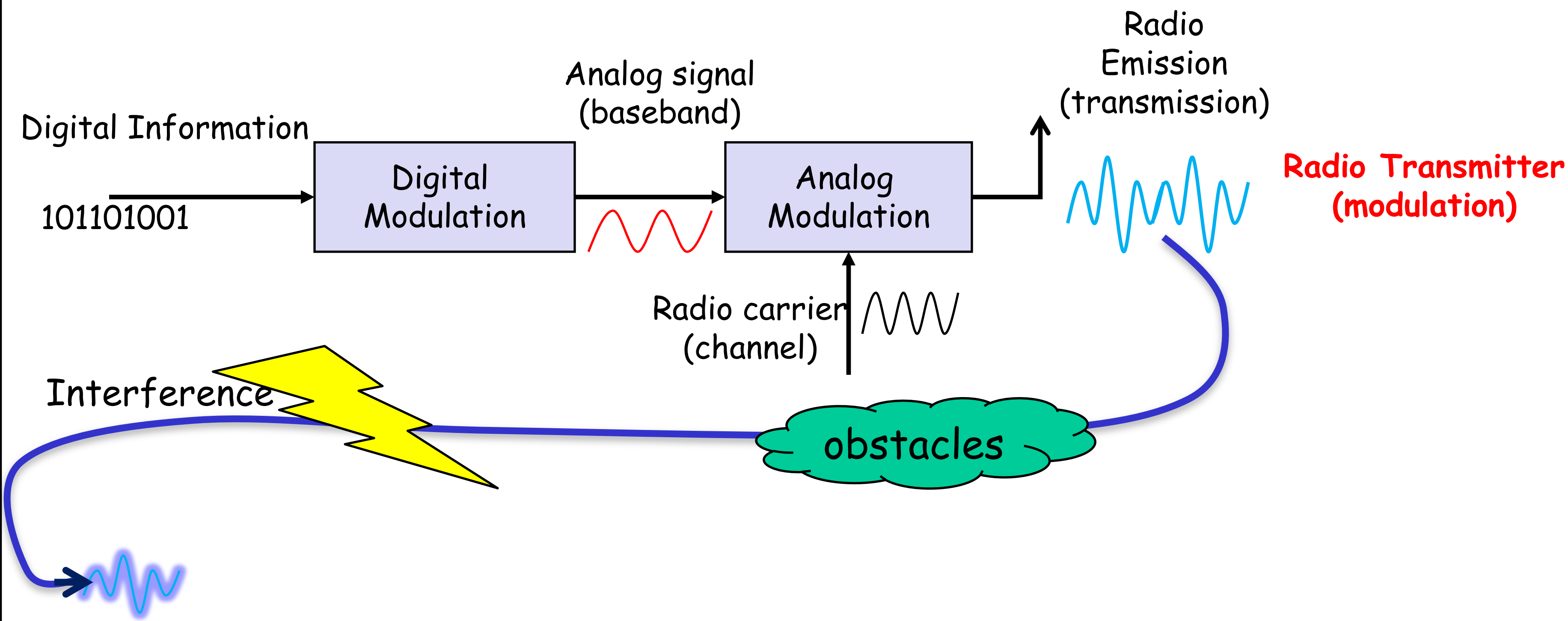
How to transmit bits with radio waves?

Digital Modulation and Demodulation



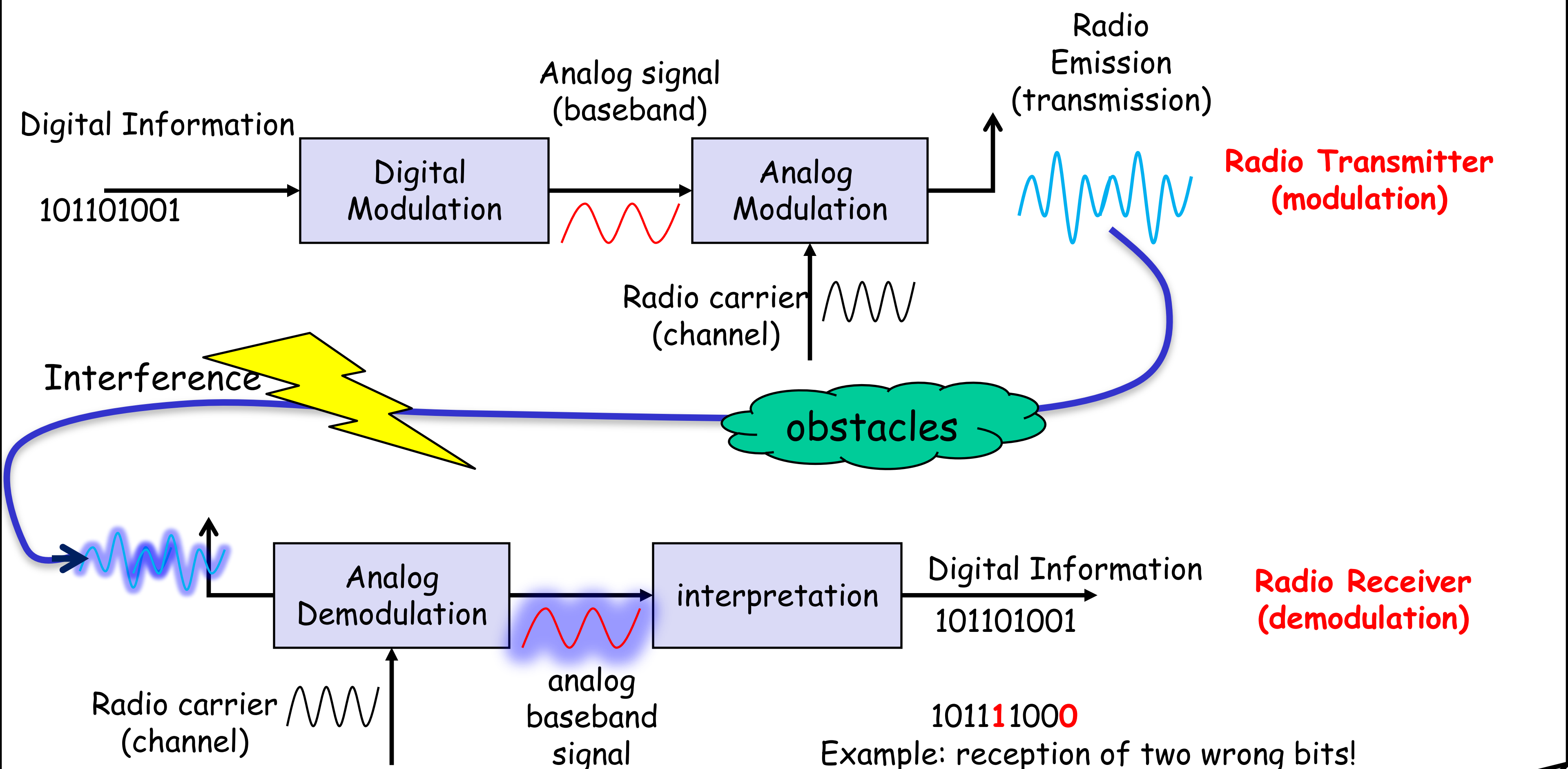
How to transmit bits with radio waves?

Digital Modulation and Demodulation

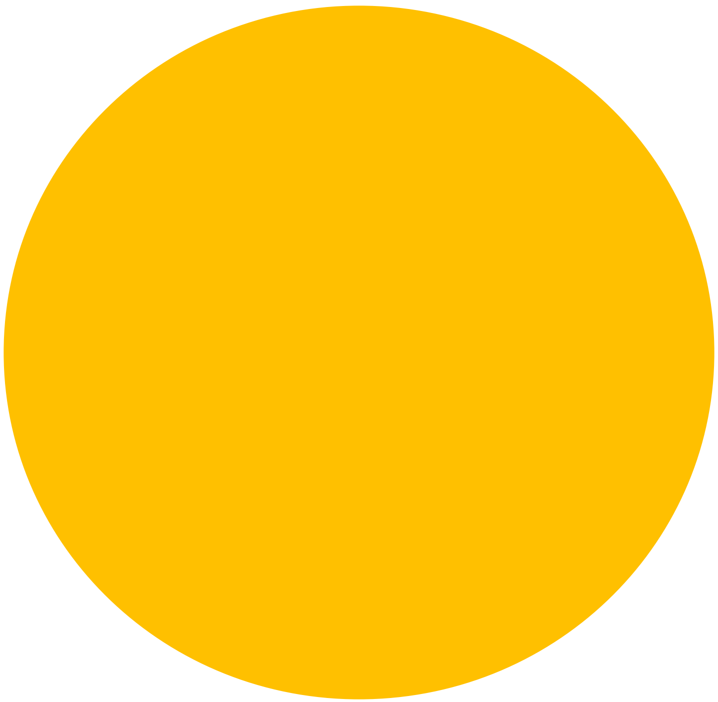


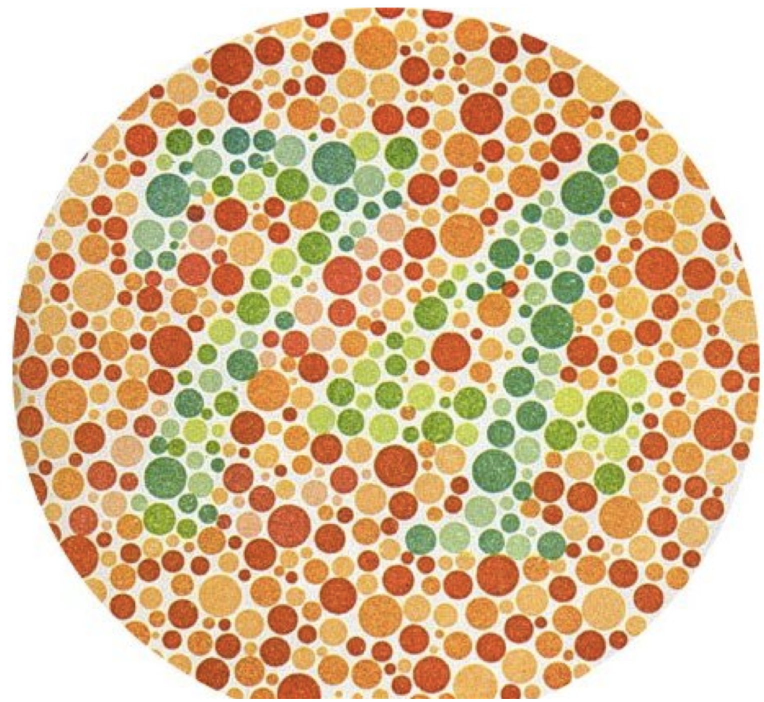
How to transmit bits with radio waves?

Digital Modulation and Demodulation

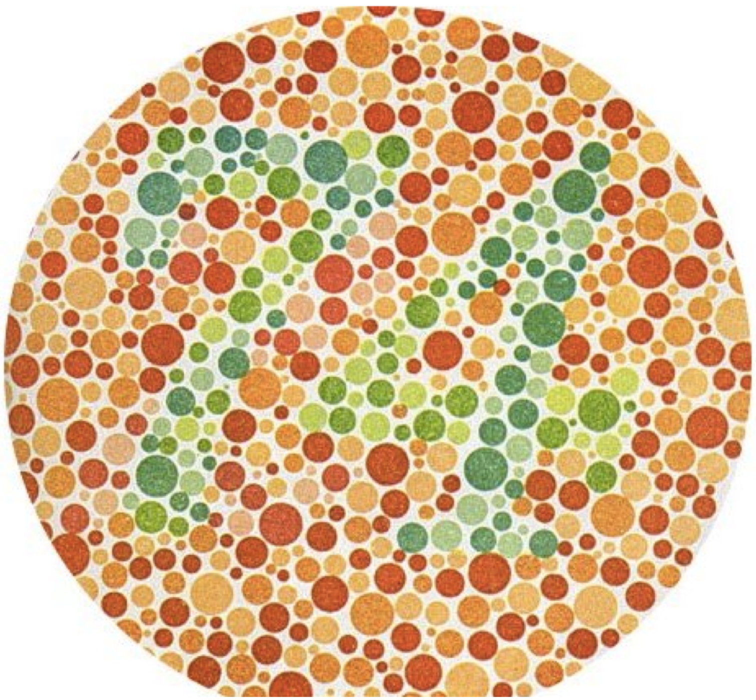


The signal to noise effect (interpretation possible?)



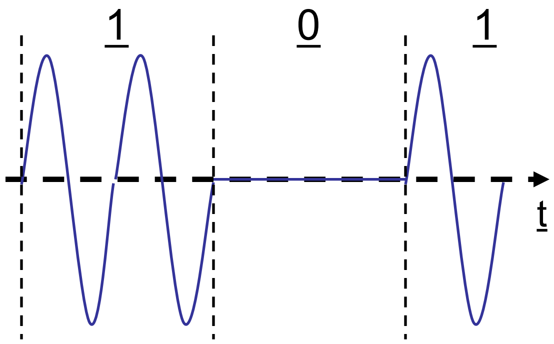


The signal to noise effect (interpretation possible?)



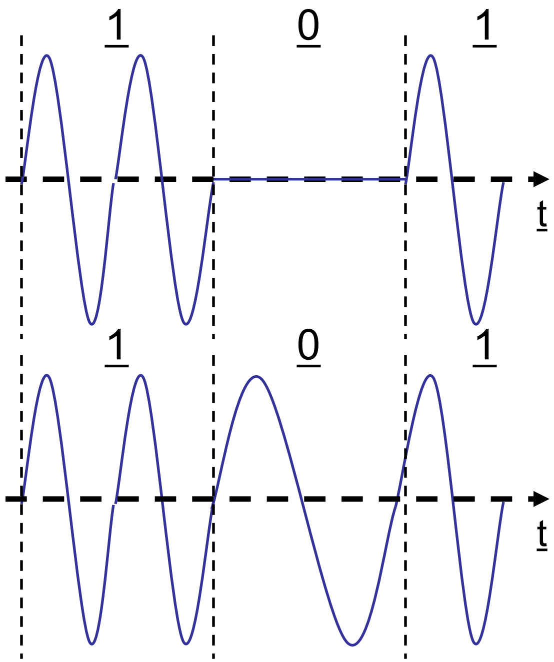
Digital Modulation Techniques

- Signal Modulation (Shift Keying)
- Amplitude Shift Keying (ASK):
  - Simple (on/off)
  - Uses few spectrum resources
  - But subject to high interference



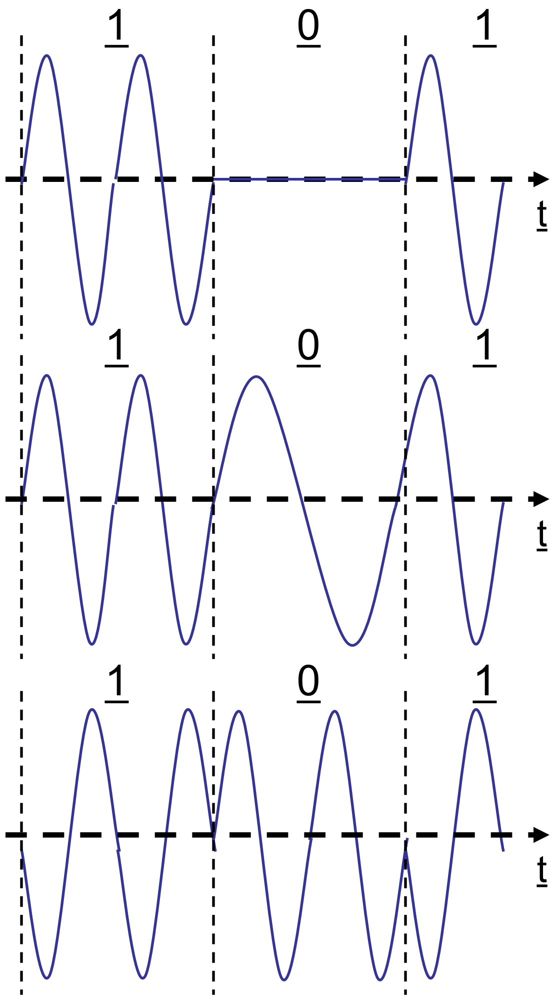
Digital Modulation Techniques

- Signal Modulation (Shift Keying)
- Amplitude Shift Keying (ASK):
  - Simple (on/off)
  - Uses few spectrum resources
  - But subject to high interference
- Frequency Shift Keying (FSK):
  - Uses more spectrum
  - “high” and “low” frequencies



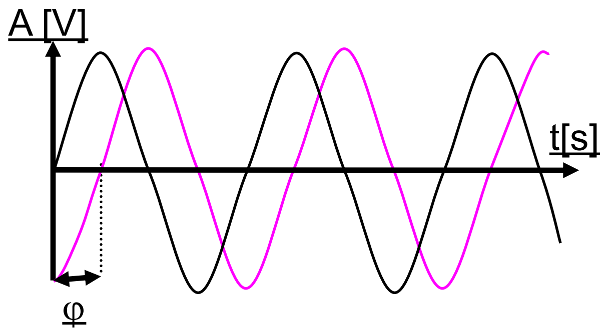
Digital Modulation Techniques

- Signal Modulation (Shift Keying)
- Amplitude Shift Keying (ASK):
  - Simple (on/off)
  - Uses few spectrum resources
  - But subject to high interference
- Frequency Shift Keying (FSK):
  - Uses more spectrum
  - “high” and “low” frequencies
- Phase Shift Keying (PSK):
  - More complex to implement
  - More robust against interference
  - Many phase levels of signal possible



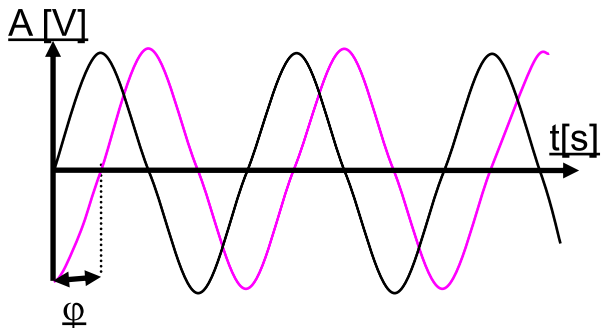
Signal Representation

- There are different ways to graphically represent the characteristics of a radio signal:
  - (a) Amplitude Domain



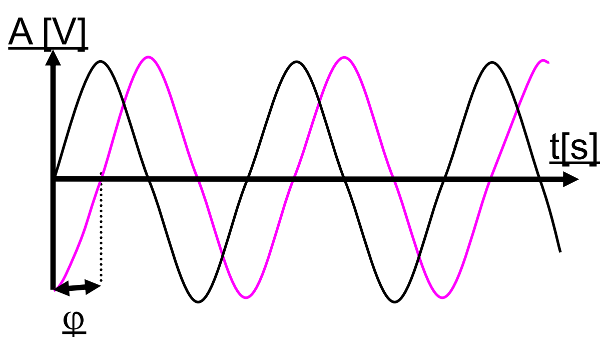
Signal Representation

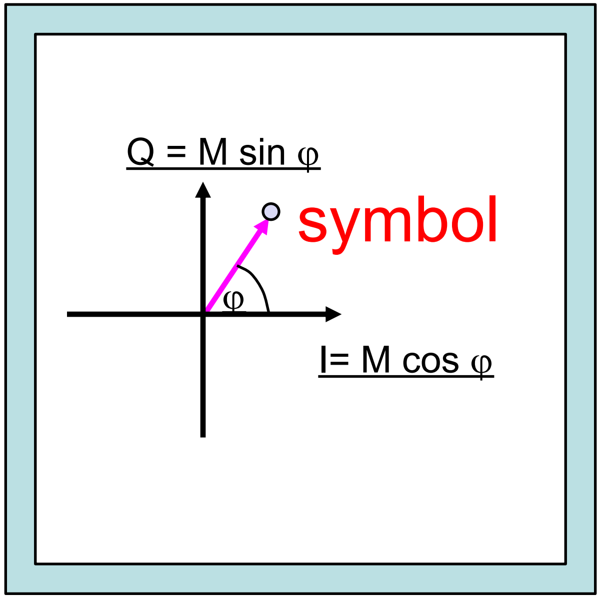
- There are different ways to graphically represent the characteristics of a radio signal:
  - (a) Amplitude Domain
  - (b) Frequency Domain



Signal Representation

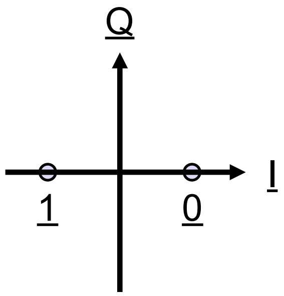
- There are different ways to graphically represent the characteristics of a radio signal:
  - (a) Amplitude Domain
  - (b) Frequency Domain
  - (c) Stat diagram of phase and amplitude (amplitude M and phase φ in polar coordinates)
    - Every SYMBOL represents a possible state (phase and amplitude) of the transmitted (and received) radio frequency.





One example?

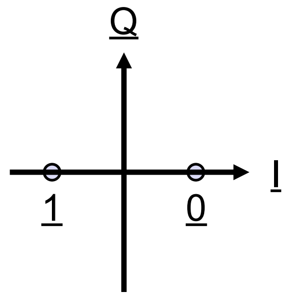
- **BPSK (Binary Phase Shift Keying):**
  - **Every symbol represents a bit value:**
    - **Bit 0: transmitted signal sin(t) (in phase 0)**
    - **Bit 1: transmitted signal sin(t) in phase 180°**
  - **Simple and robust example of PSK**
  - **Es. Used in satellite communications**
  - **But has low spectral efficiency (few bits per spectrum unit)**



Can we do better?

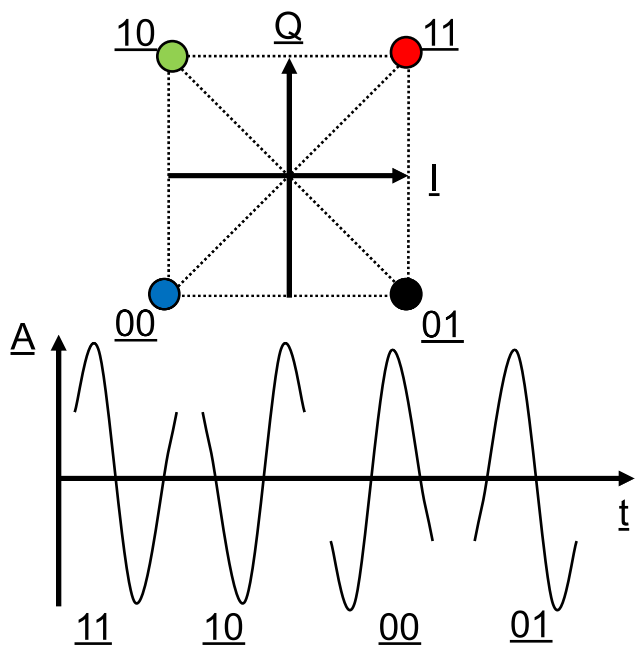
One example?

- **BPSK (Binary Phase Shift Keying):**
  - **Every symbol represents a bit value:**
    - **Bit 0: transmitted signal sin(t) (in phase 0)**
    - **Bit 1: transmitted signal sin(t) in phase 180°**
  - **Simple and robust example of PSK**
  - **Es. Used in satellite communications**
  - **But has low spectral efficiency (few bits per spectrum unit)**



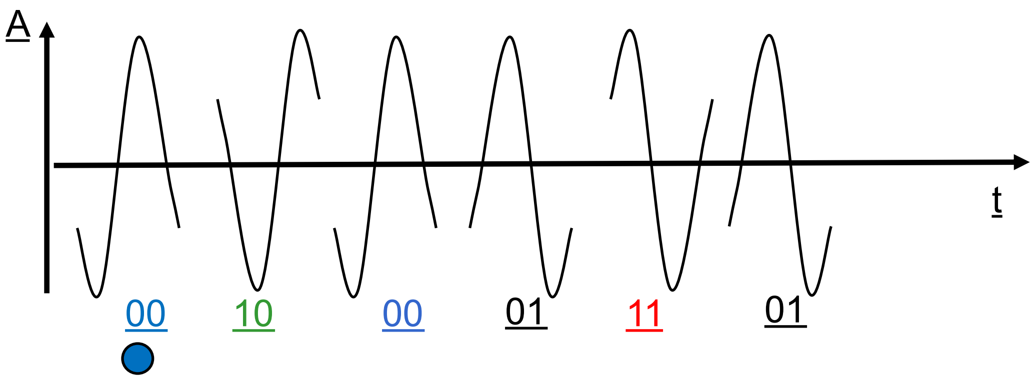
Can we do better?

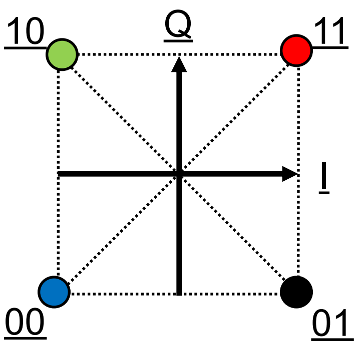
- **QPSK (Quadrature Phase Shift Keying):**
  - **Every symbol represents a value on two bits:**
    - **Bit 11: transmitted signal sin(t) in phase +45°**
    - **Bit 10: transmitted signal sin(t) in phase +135°**
    - **Bit 11: transmitted signal sin(t) in phase +225°**
    - **Bit 10: transmitted signal sin(t) in phase +315°**
  - **More complex and vulnerable**
    - **How much interference is needed to realize a wrong interpretation of a symbol on the receiver?**



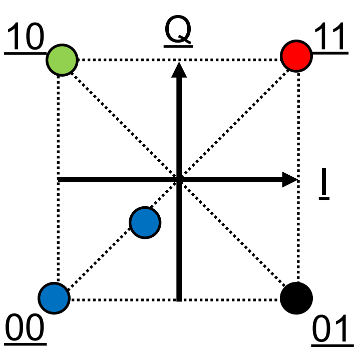
Imagine a target shooting…

- es. How do I "launch" in transmission these bits?
- 001000011101… = 00 10 00 01 11 01….
- Emitting a wave assuming in sequence the characteristics of the symbols associated to pairs in the bit sequence:



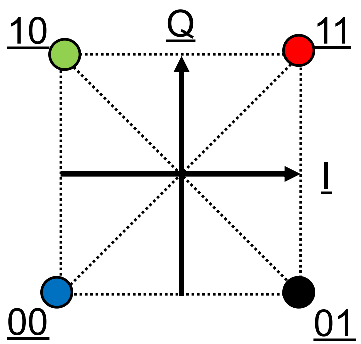


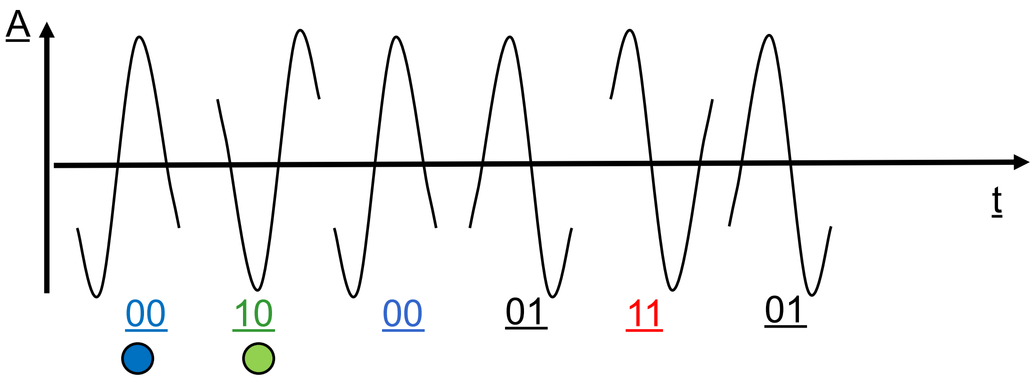
- …the receiver tries to understand the symbols
- Despite the symbols falls out of the target it would be possible to interpret them as the "nearest" target



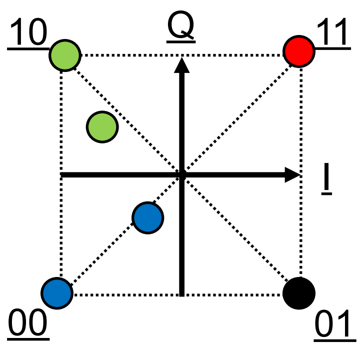
Imagine a target shooting…

- es. How do I "launch" in transmission these bits?
- 001000011101… = 00 10 00 01 11 01….
- Emitting a wave assuming in sequence the characteristics of the symbols associated to pairs in the bit sequence:



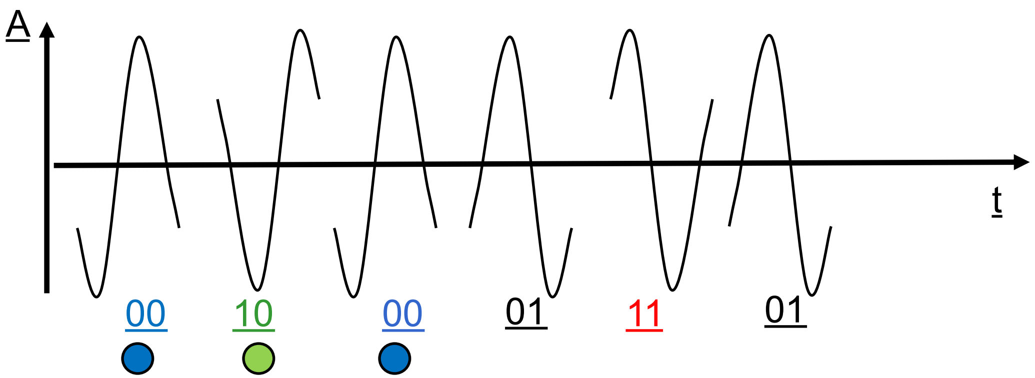


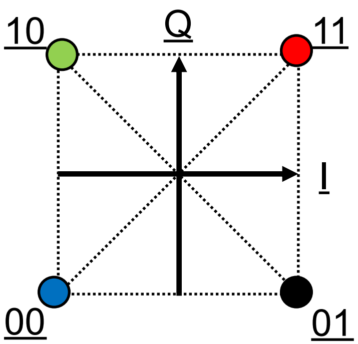
- …the receiver tries to understand the symbols
- Despite the symbols falls out of the target it would be possible to interpret them as the "nearest" target



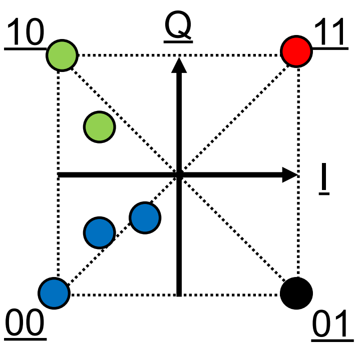
Imagine a target shooting…

- es. How do I “launch” in transmission these bits?
- 001000011101… = 00 10 00 01 11 01….
- Emitting a wave assuming in sequence the characteristics of the symbols associated to pairs in the bit sequence:



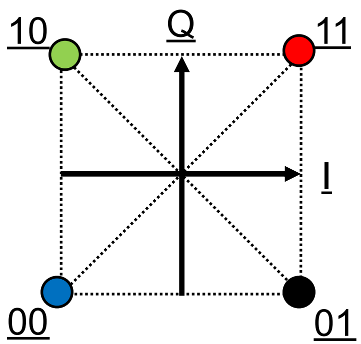


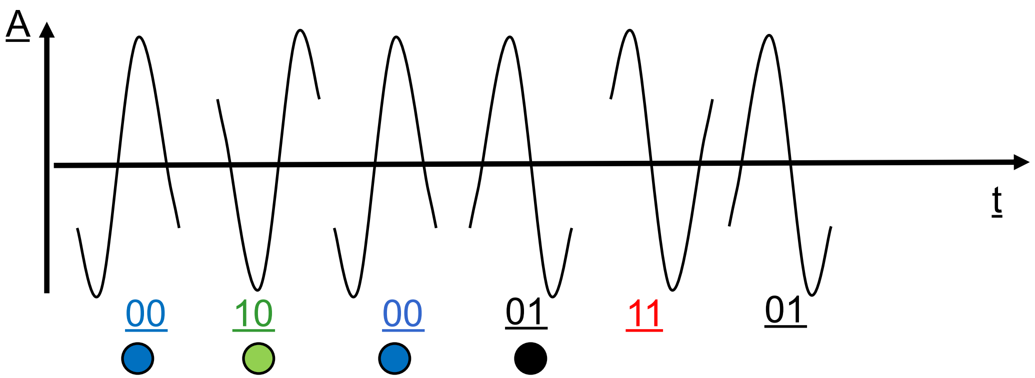
- …the receiver tries to understand the symbols
- Despite the symbols falls out of the target it would be possible to interpret them as the “nearest” target



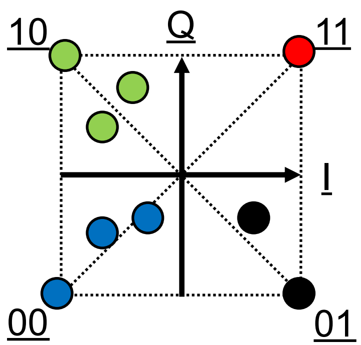
Imagine a target shooting…

- es. How do I “launch” in transmission these bits?
- 001000011101… = 00 10 00 01 11 01….
- Emitting a wave assuming in sequence the characteristics of the symbols associated to pairs in the bit sequence:



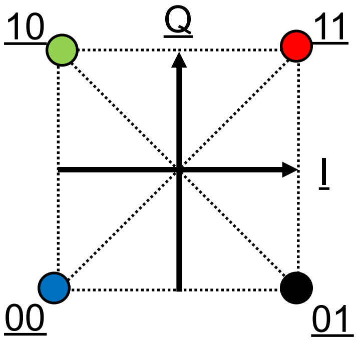


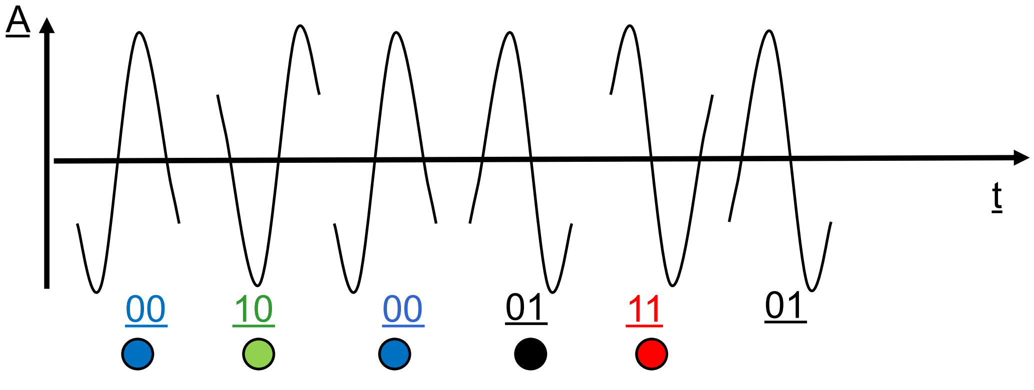
- …the receiver tries to understand the symbols
- Despite the symbols falls out of the target it would be possible to interpret them as the “nearest” target

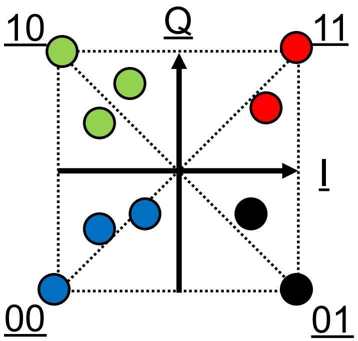


Imagine a target shooting…

- es. How do I “launch” in transmission these bits?
- 001000011101… = 00 10 00 01 11 01….
- Emitting a wave assuming in sequence the characteristics of the symbols associated to pairs in the bit sequence:

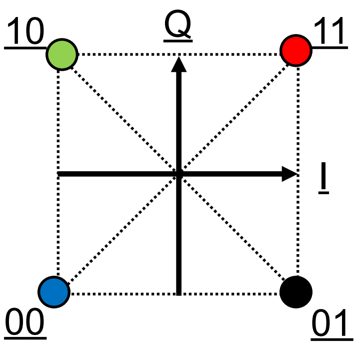




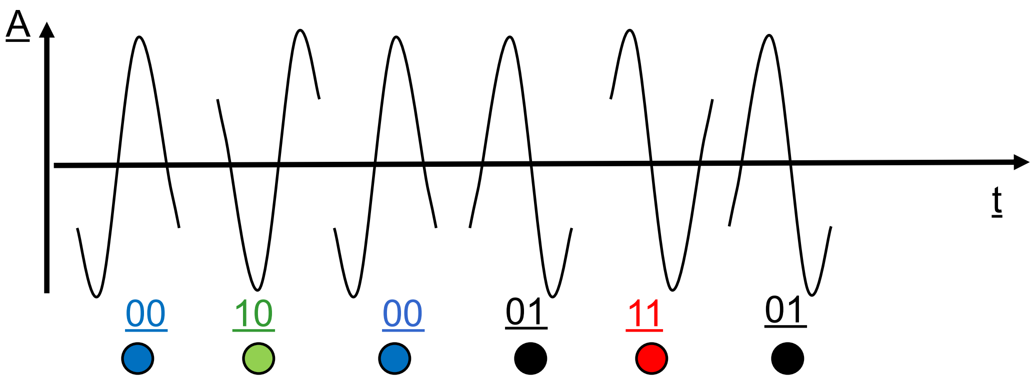


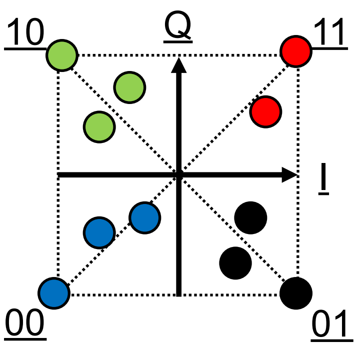
- …the receiver tries to understand the symbols
- Despite the symbols falls out of the target it would be possible to interpret them as the “nearest” target

Imagine a target shooting…



- es. How do I "launch" in transmission these bits?
- 001000011101… = 00 10 00 01 11 01….
- Emitting a wave assuming in sequence the characteristics of the symbols associated to pairs in the bit sequence:

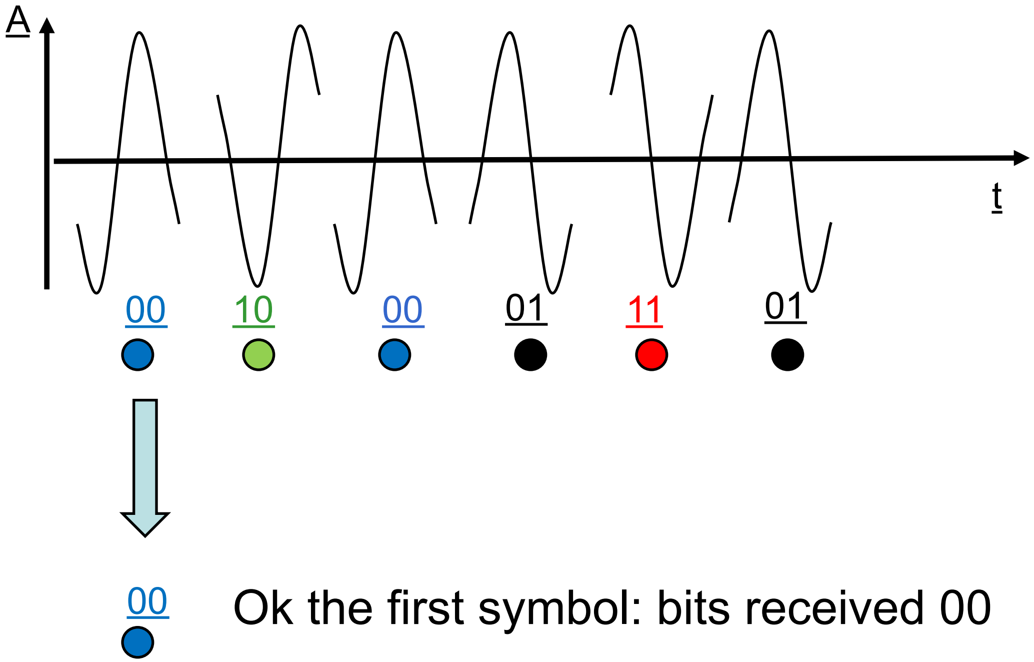


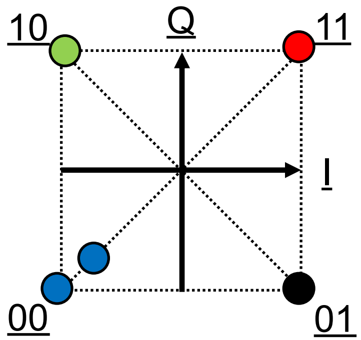


- …the receiver tries to understand the symbols
- Despite the symbols falls out of the target it would be possible to interpret them as the "nearest" target

Imagine a target shooting…

- Q: Who can say when errors are possible?
- **Let's see one example step by step**

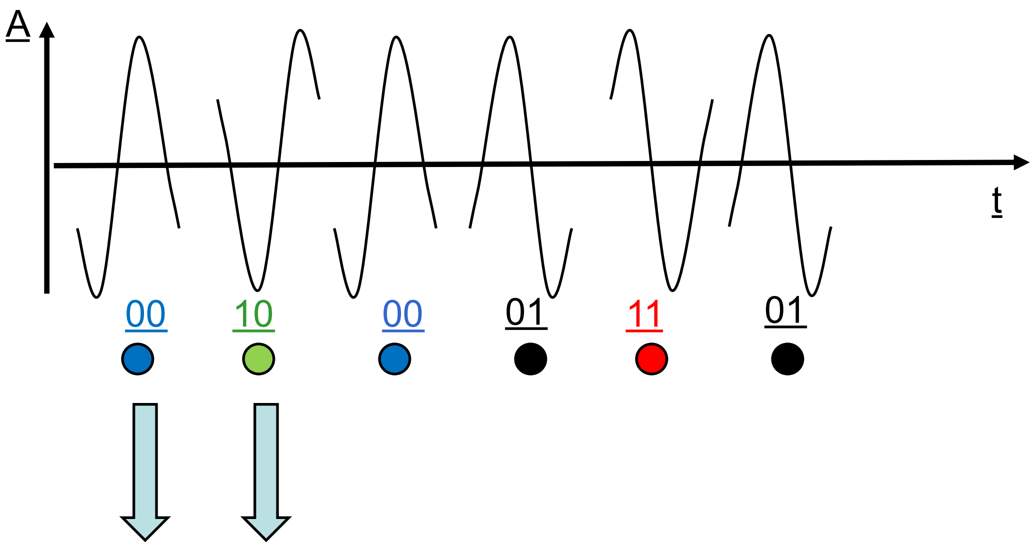




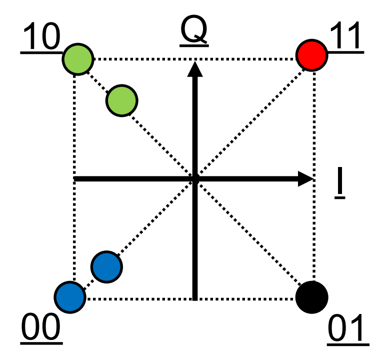
Ok the first symbol: bits received 00

Imagine a target shooting…

- Q: Who can say when errors are possible?
- Let's see one example step by step

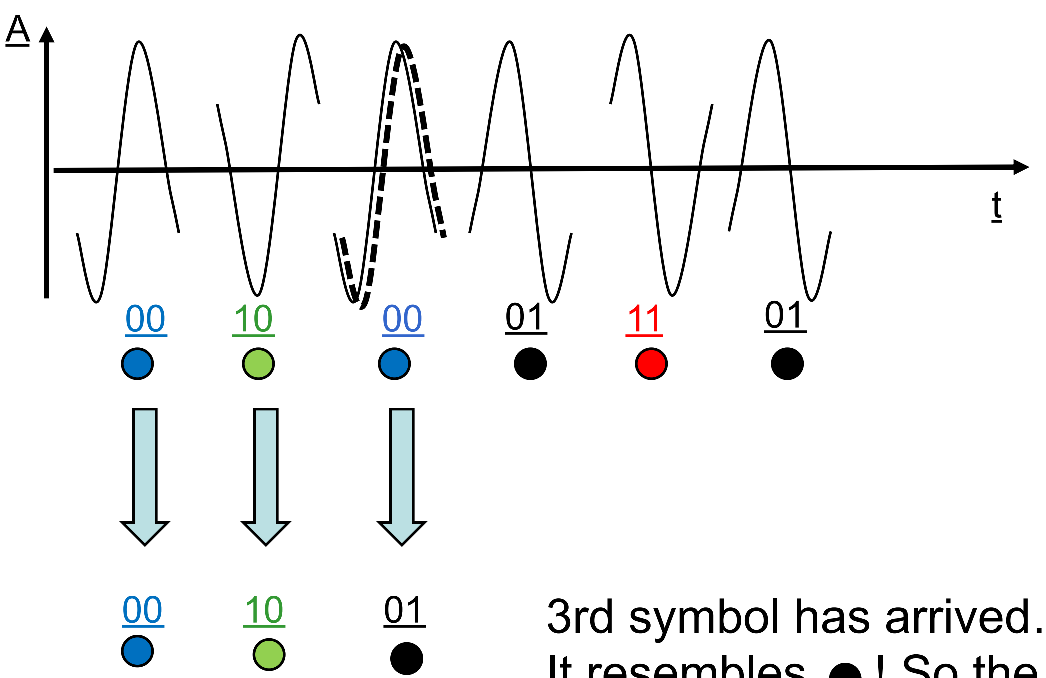


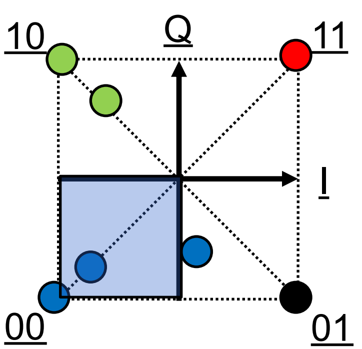
All is fine with 2nd symbol! Received bits 00 10



Imagine a target shooting…

- Q: Who can say when errors are possible?
- **A: when the changes of phase and amplitude are so high that the limits of the target are exceeded, that is, the area of the target is not hit.**

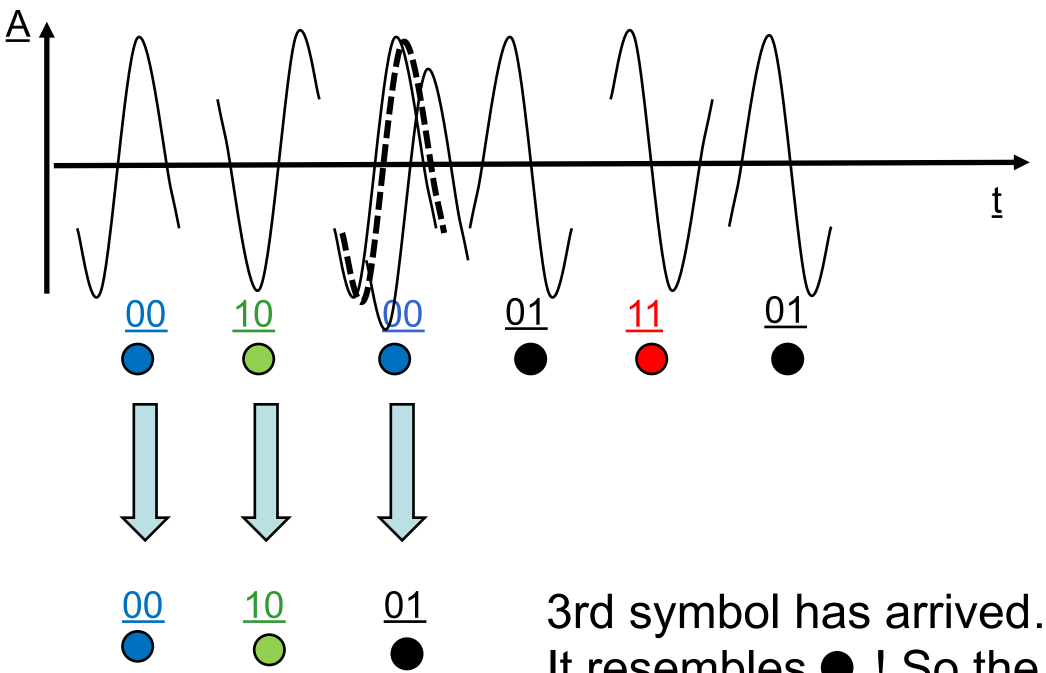


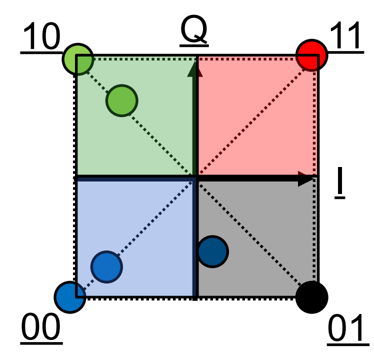


3rd symbol has arrived…
It resembles ● ! So the received bits are 00 10 01: **Error!**

Imagine a target shooting…

- Q: Who can say when errors are possible?
- **A: when the changes of phase and amplitude are so high that the limits of the target are exceeded, that is, the area of the target is not hit.**





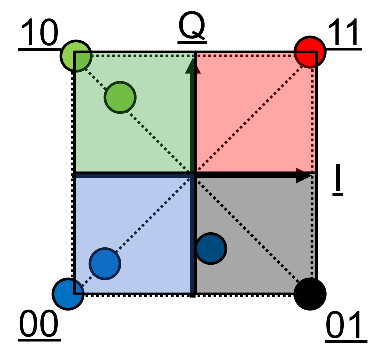
3rd symbol has arrived…
It resembles ● ! So the received bits are 00 10 0**1: Error!**

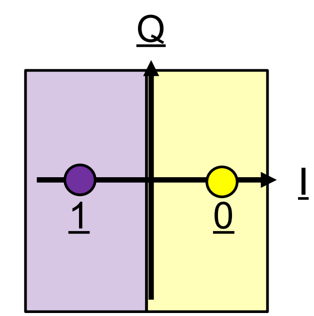
The area of the target…

- **Useful observation:** when the area of the target is small, a small error is sufficient to cause some wrong bits!

- **How could we change this fact in positive?**

- **1) when the channel is noisy, we can use a BPSK with just 2 symbols (distance 180°) , i.e. we increase the target area!**
  - **advantage: all the bits sent are correct despite the noise**
  - **disadvantage: we get half of the nominal bitate (1 bit per symbol)**
  - **00 10 00 01 11 01… = 0 0 1 0 0 0 0 1 1 1 0 1**





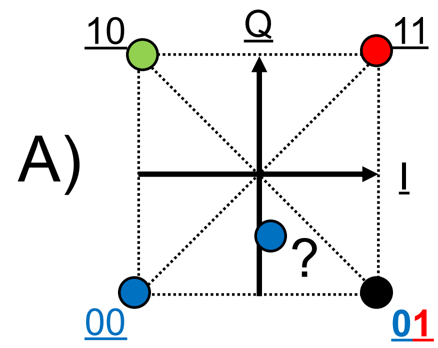
- **2) when the channel is good (low noise, a majority of bits received correctly) we can "push on the encoding accelerator" by increasing the nominal bitrate of the channel.**

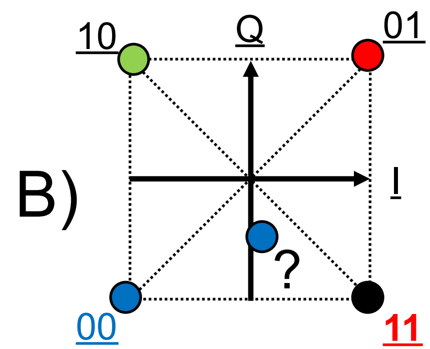
Bits and symbols associations

- **Second observation:** it is highly probable that a noisy channel would cause a symbol to be wrong **but misplaced with the adjacent ones (near to the border).**
- **How to exploit this assumption?**
- **1) we can decide to use a more "intelligent" labelling of the symbols.**
- **Example: what happens if the same symbol is wrong in case A and B?**
- **Case A: sent 00** ⇒ **received 01: 1 bit error!**
- **Case B: sent 00** ⇒ **received 11: 2 bit errors!**

N.B. given the same channel noise errors are doubled!!!

- **So, not all the same labelling associations of bits with symbols are equivalent…**





- **The best labelling are those where the number of different bits between adjacent symbols is minimum. So we must find those labelling and use them!**

Detecting wrong bits

- **Third useful observation:** why it is important that max 1 bit is wrong?
- Because we invented a nice algorithm to reveal the wrong bits: **Parity bits.**
- *"given a sequence of bits to transmit, we add a final bit which will make even the number of 1s"*
- **Example. 10010101 0 (4 bits have value one, and 4 is already even)**

Sender

10010101 0

→

Receiver

10010101 0

Note: whatever different bit would make the number of ones odd:
This allows to detect the existence of a wrong bit!

Q: what if 2 bits are wrong?

Detect and correct wrong bits

- **Third useful observation:** why it is important that max 1 bit is wrong?
- Because we invented a nice algorithm to reveal the wrong bits, and in case the wrong bit is just one, also to correct it! : **Parity bits matrix.**
- *"Given a sequence of bits to be transmitted we organize the bits in a matrix structure MxN and we put a parity bit after each row (M bits) and column (N bits)."*

Sender

1 0 0 1 0
0 1 1 0 0
0 0 0 1 1
1 1 0 1 1
0 0 1 1

Receiver

1 0 0 1 0
0 0 1 0 0 !
0 0 0 1 1
1 1 0 1 1
0 0 1 1
!

Like in a battleship game, if we detect a row and a column with errors we identify the wrong bit!
Then we can also correct it!

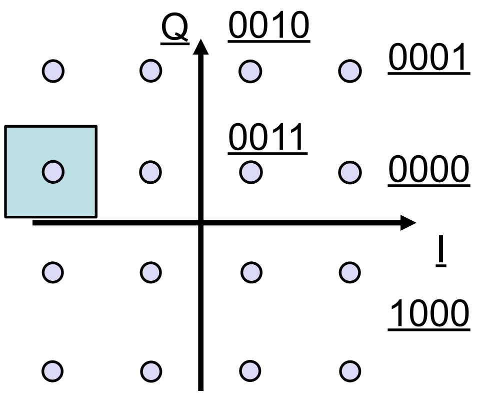
Q: what if the wrong bits are 2?

Quadrature Amplitude Modulation

- **Fourth observation:** what if the channel is even better quality?
- **We can push the encoding even more!**
- **Quadrature Amplitude Modulation (QAM):** it combines modulation of both amplitude and phase of the signal for each transmitted symbol.
- 2^n defined symbols: every symbol identifies by itself a combinatin of n bits!
- …however, be careful, since the area of the target always reduces when n grows!

Area of the target (see figure)

Example in figure: **16-QAM (16 symbols, 1 symbol = 4 bit)**



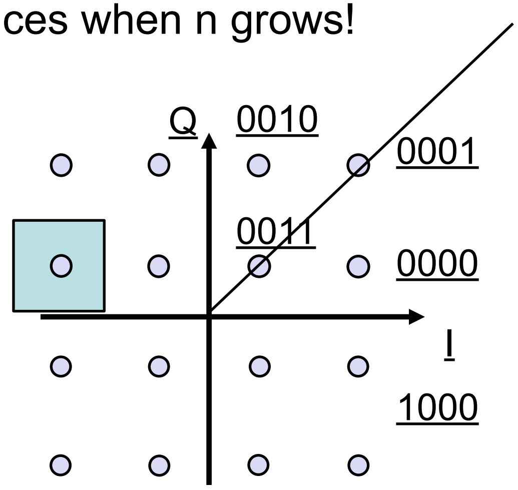
Quadrature Amplitude Modulation

- **Fourth observation:** what if the channel is even better quality?
- **We can push the encoding even more!**
- **Quadrature Amplitude Modulation (QAM):** it combines modulation of both amplitude and phase of the signal for each transmitted symbol.
- 2^n defined symbols: every symbol identifies by itself a combinatin of n bits!
- …however, be careful, since the area of the target always reduces when n grows!

Area of the target (see figure)

Example in figure: **16-QAM (16 symbols, 1 symbol = 4 bit)**

- Note. the symbols 0011 and 0001 have same phase but different amplitude

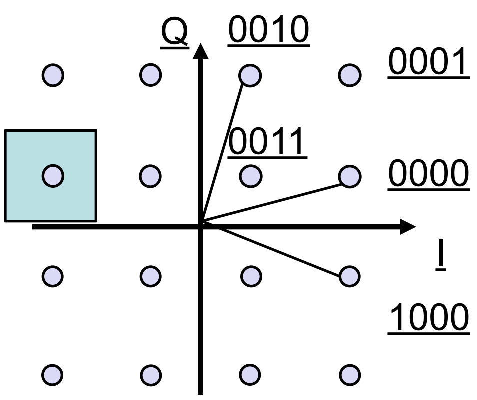


Quadrature Amplitude Modulation

- **Fourth observation:** what if the channel is even better quality?
- **We can push the encoding even more!**
- **Quadrature Amplitude Modulation (QAM):** it combines modulation of both amplitude and phase of the signal for each transmitted symbol.
- 2^n defined symbols: every symbol identifies by itself a combinatin of n bits!
- …however, be careful, since the area of the target always reduces when n grows!

Area of the target (see figure)

Example in figure: **16-QAM (16 symbols, 1 symbol = 4 bit)**

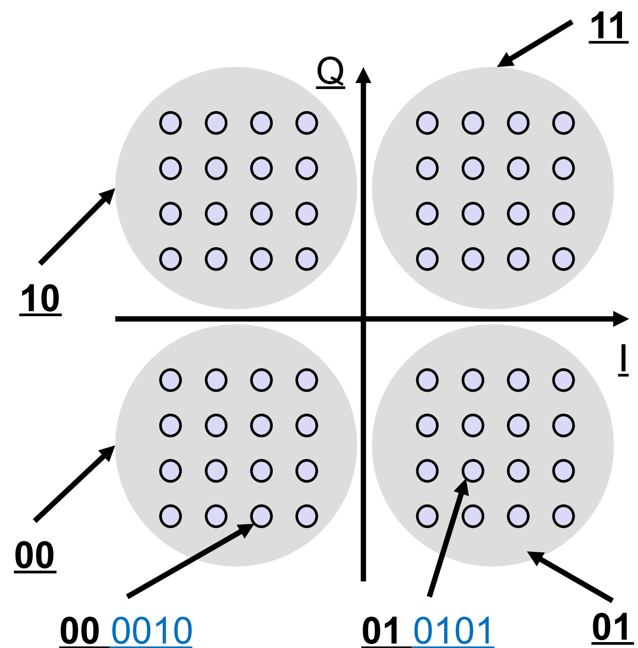


- Note. the symbols 0011 and 0001 have same phase but different amplitude
- This enconding was used in the early 9600 bit/s modems, and also in Digital TV, in Wi-max (multicarrier OFDM)…etc.

- Let's have a look at a simulation of the modulation (with variable channel errors): file:///Users/Luciano/Desktop/Didattica/Mambo_20140315/QAM16_demo/QAM16.html

Hierarchical Modulation

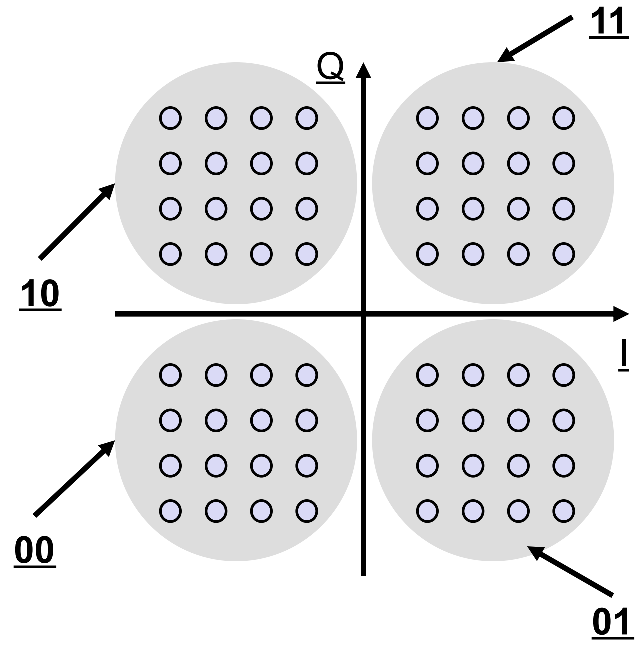
- **Fifth observation:** let's introduce a new magic thing!
- Q1: with the QAM encoding could I modulate two different sequences of bits?
- Q2: and could I give them different priority in transmission (protection from errors)?
- Example: **64-QAM with Hierarchical Modulation**
- **Each symbol oncodes 6 bits!**
  - Each "gray cloud" contains 16 symbols
    - Used to encode the bit sequence with LOW priority
    - …as an example: the video info of a video-call
  - Each "gray cloud" is labelled with the value of a combination of 2 bits
    - Used to encode the bit sequence with HIGH priority
    - …as an example: the voice info of a video-call



  - Q: what happens when channel has low noise?
  - Q: what happens when channel starts having high noise?

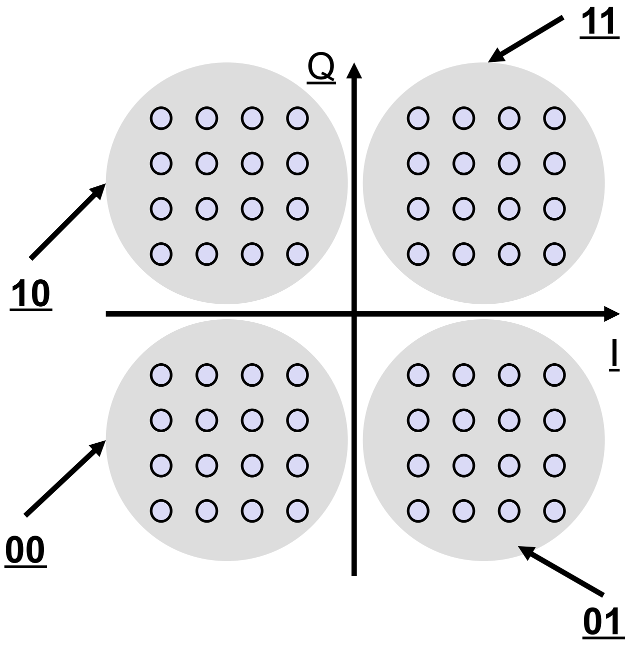
The mobile Video-call

- Ex. Let's assume a video-call generates these sequences of bits to transmit:
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…



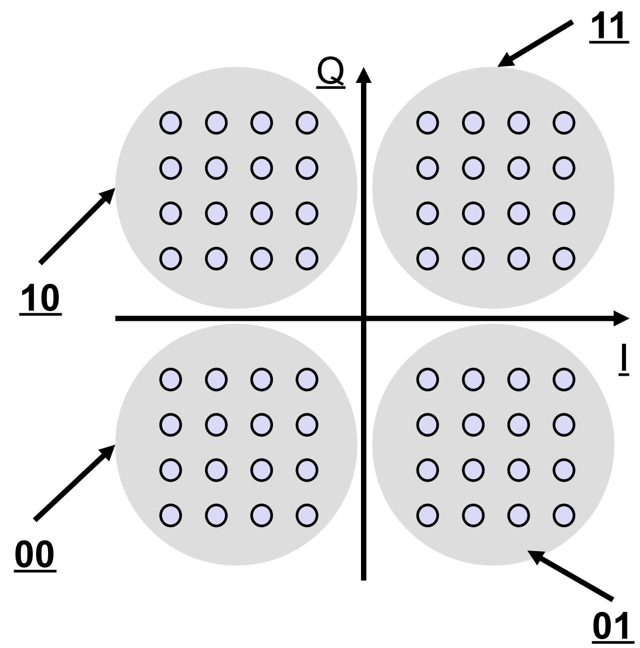
The mobile Video-call

- Ex. Let's assume a video-call generates these sequences of bits to transmit (which will be merged into a unique sequence as follows)
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- 100010



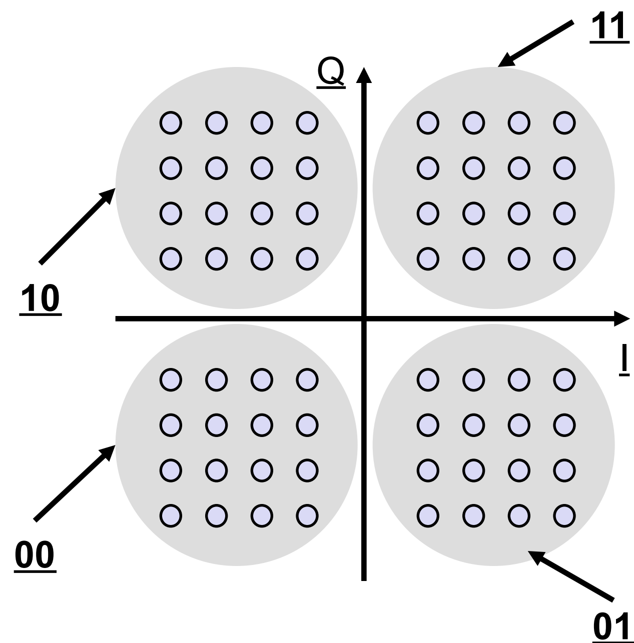
The mobile Video-call

- Ex. Let's assume a video-call generates these sequences of bits to transmit (which will be merged into a unique sequence as follows)
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- 100010 011001



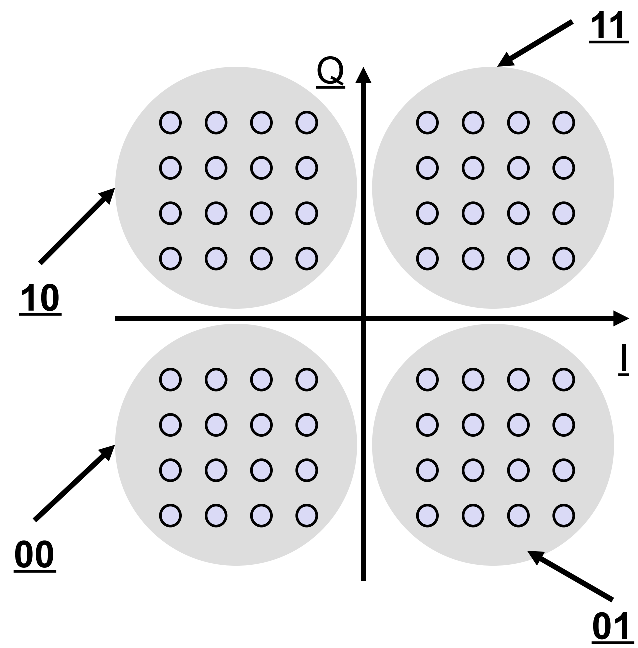
The mobile Video-call

- Ex. Let's assume a video-call generates these sequences of bits to transmit (which will be merged into a unique sequence as follows)
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- 100010 011001 111100



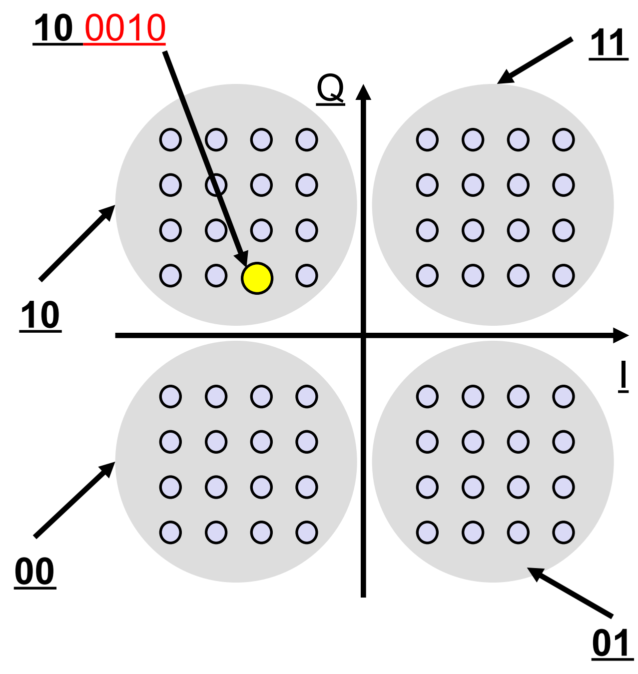
The mobile Video-call

- Ex. Let's assume a video-call generates these sequences of bits to transmit (which will be merged into a unique sequence as follows):
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- 100010 011001 111100 000101



The mobile Video-call

- Ex. Let's assume a video-call generates these sequences of bits to transmit:
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- Which will be merged in the following unique sequence:
- 100010 011001 111100 000101

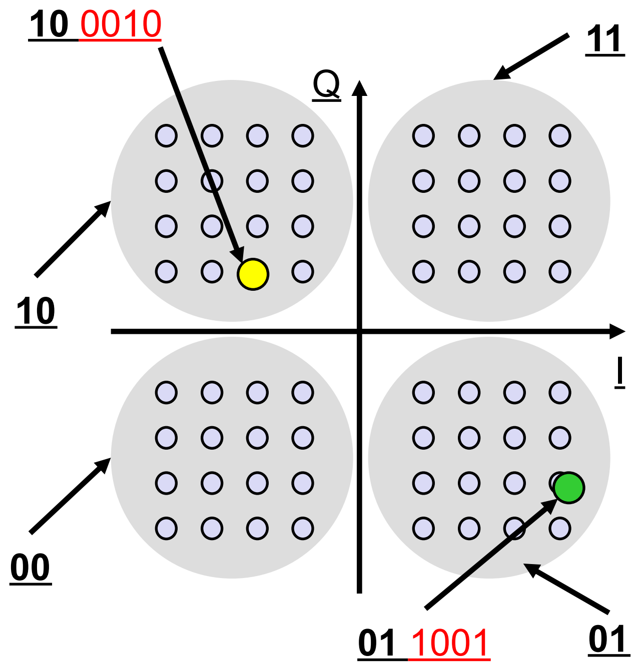


Q: what happens if the channel has low noise?

- All the symbols are hit correctly with high probability!

The mobile Video-call

- Ex. Let's assume a video-call generates these sequences of bits to transmit:
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- Which will be merged in the following unique sequence:
- 100010 011001 111100 000101



Q: what happens if the channel has low noise?

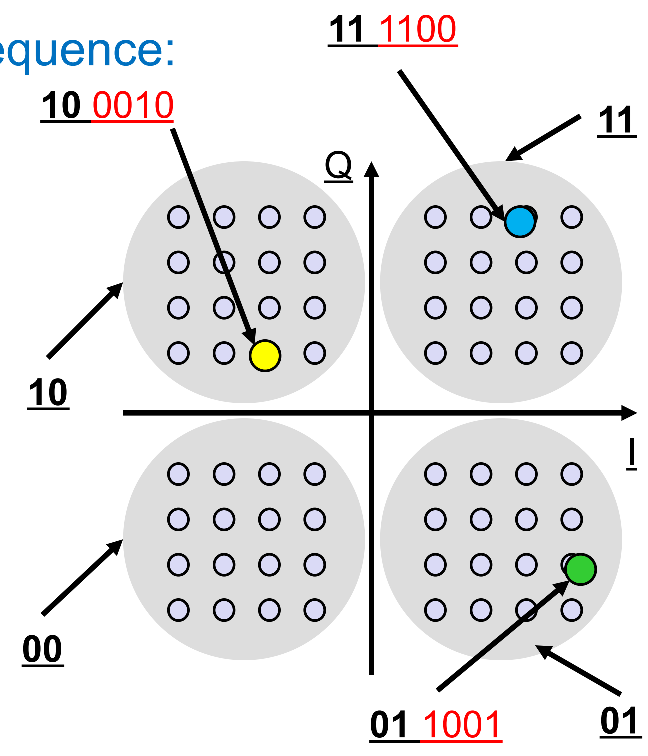
- All the symbols are hit correctly with high probability!

The mobile Video-call

- Ex. Let's assume a video-call generates these sequences of bits to transmit:
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- Which will be merged in the following unique sequence:
- 100010 011001 111100 000101

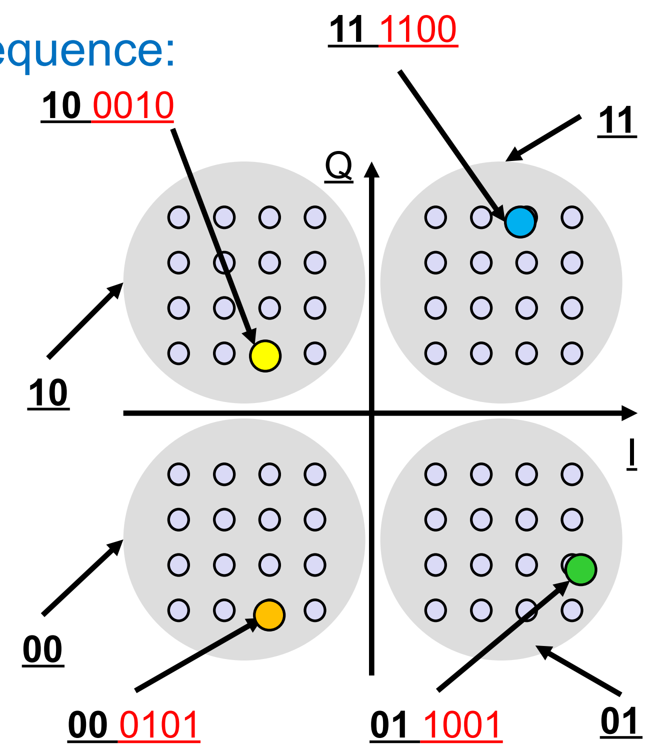
Q: what happens if the channel has low noise?

- All the symbols are hit correctly with high probability!



The mobile Video-call

- Ex. Let's assume a video-call generates these sequences of bits to transmit:
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- Which will be merged in the following unique sequence:
- 100010 011001 111100 000101

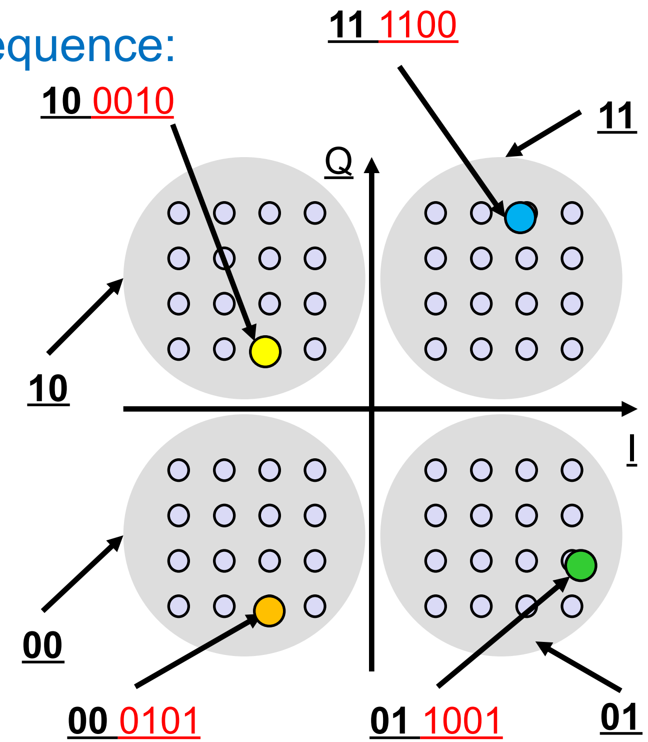


Q: what happens if the channel has low noise?

- All the symbols are hit correctly with high probability!

The mobile Video-call

- Ex. Let's assume a video-call generates these sequences of bits to transmit:
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- Which will be merged in the following unique sequence:
- 100010 011001 111100 000101



Q: what happens if the channel has low noise?

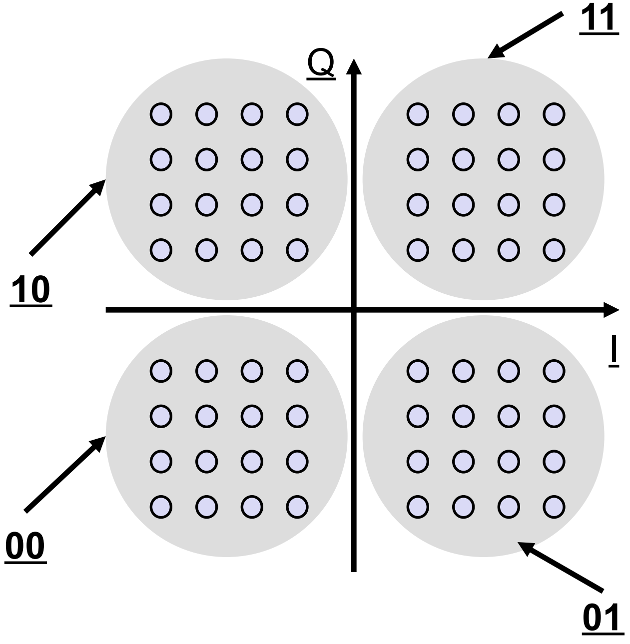
- All the symbols are hit correctly with high probability!
- ...so the receiver can correctly interprete both voice and video

Supercalifragilisticexpialidocious!!!!

The mobile Video-call

- Ex. Let's assume the call generates the following bit sequence to transmit:
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- Which will be merged as the following sequence:
- 100010 011001 111100 000101

Q: **what if the channel has a high noise?**

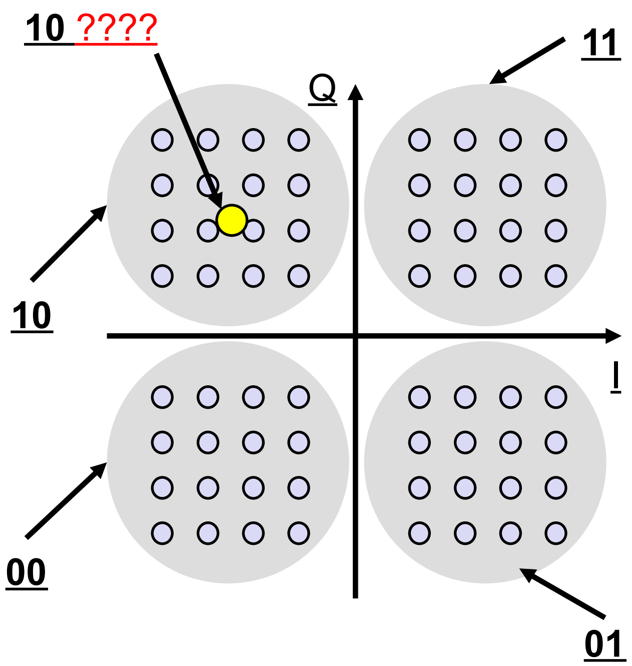


The mobile Video-call

- Ex. Let's assume the call generates the following bit sequence to transmit:
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- Which will be merged as the following sequence:
- 100010 011001 111100 000101

Q: what if the channel has a high noise?

- Many symbols are NOT hit correctly with high probability! ?

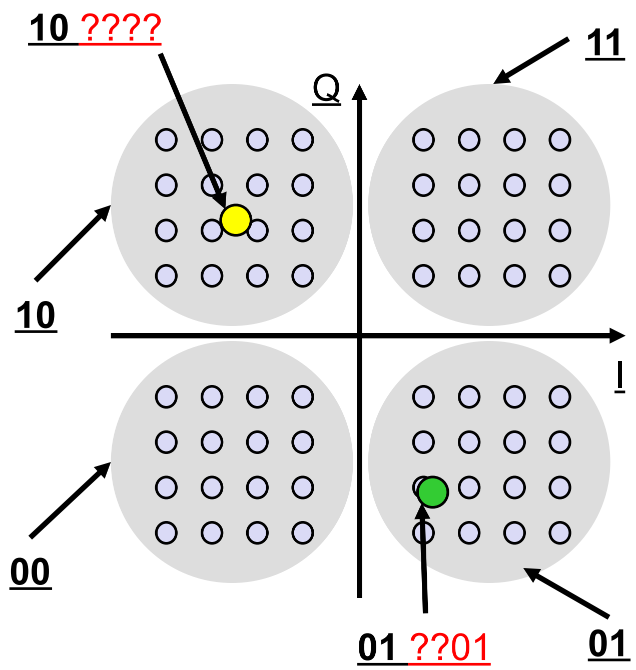


The mobile Video-call

- Ex. Let's assume the call generates the following bit sequence to transmit:
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- Which will be merged as the following sequence:
- 100010 011001 111100 000101

Q: what if the channel has a high noise?

- Many symbols are NOT hit correctly with high probability! ? ?

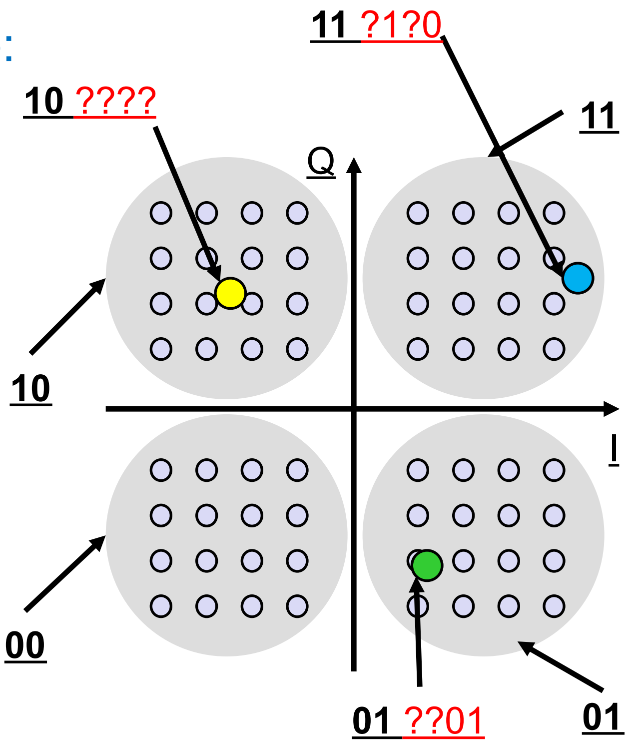


The mobile Video-call

- Ex. Let's assume the call generates the following bit sequence to transmit:
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- Which will be merged as the following sequence:
- 100010 011001 111100 000101

Q: what if the channel has a high noise?

- Many symbols are NOT hit correctly with high probability! ? ? ?

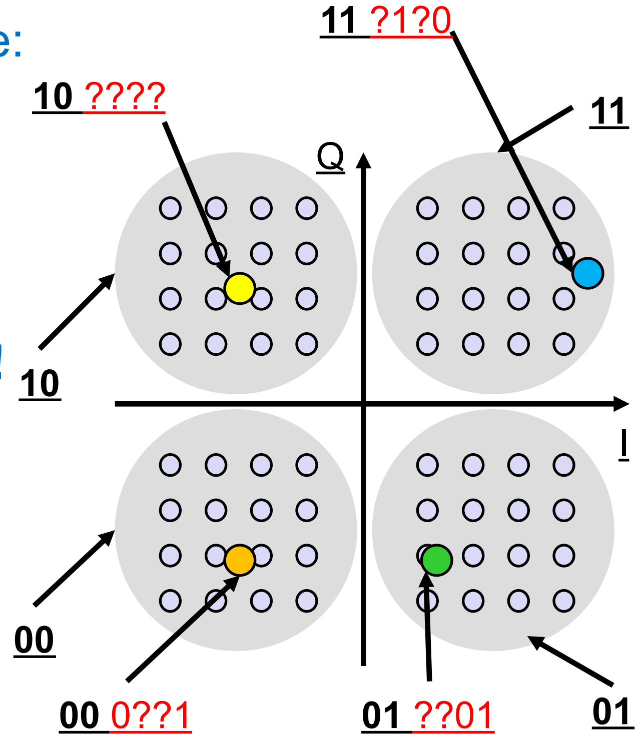


The mobile Video-call

- Ex. Let's assume the call generates the following bit sequence to transmit:
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- Which will be merged as the following sequence:
- 100010 011001 111100 000101

Q: what if the channel has a high noise?

- Many symbols are NOT hit correctly with high probability! ? ? ? ?
  BUT the right CLOUD is always hit correctly!

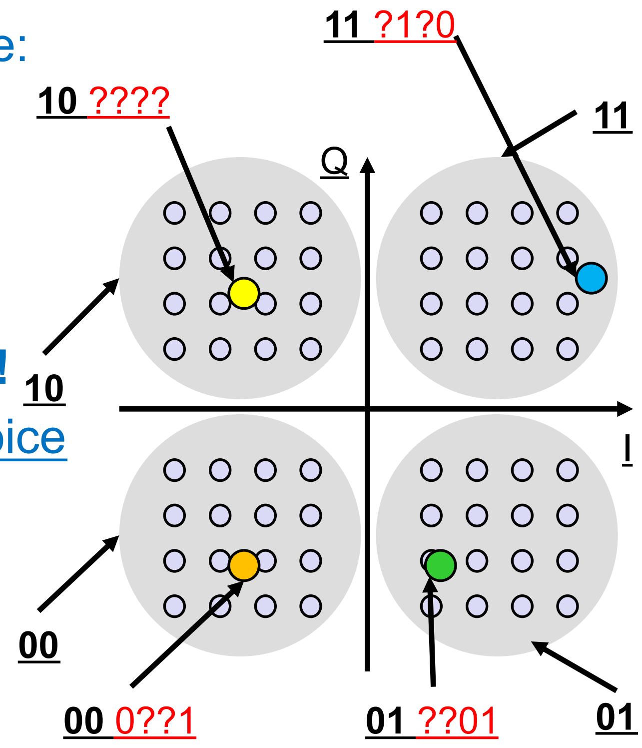


The mobile Video-call

- Ex. Let's assume the call generates the following bit sequence to transmit:
- Voice: 10 01 11 00…
- Video: 0010 1001 1100 0101…
- Which will be merged as the following sequence:
- 100010 011001 111100 000101

Q: what if the channel has a high noise?

- Many symbols are NOT hit correctly with high probability! ? ? ? ?
  BUT the right CLOUD is always hit correctly!
- So the receiver is able to correctly detect the voice by sacrifying the video quality!



Supercalifragilisticexpialidocious!!!!

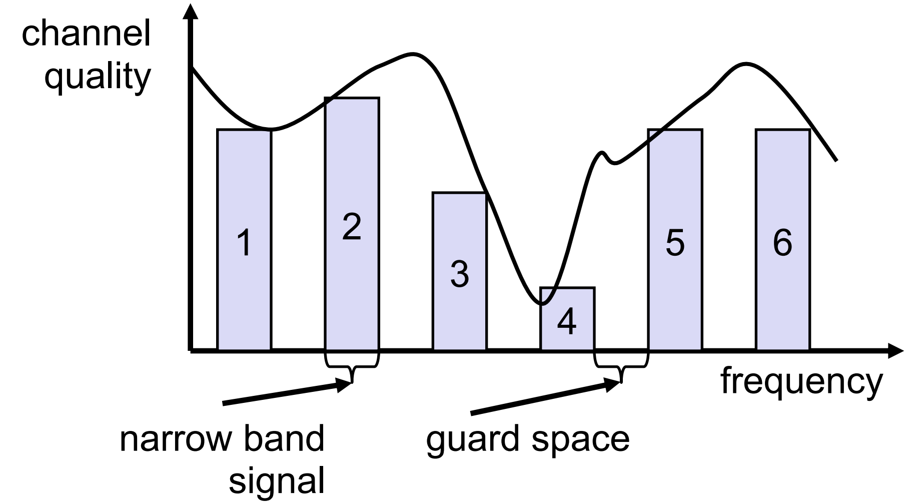
Conclusion

Where is the difference between the Good and the Bad in the wireless transmissions, given the same physical conditions?

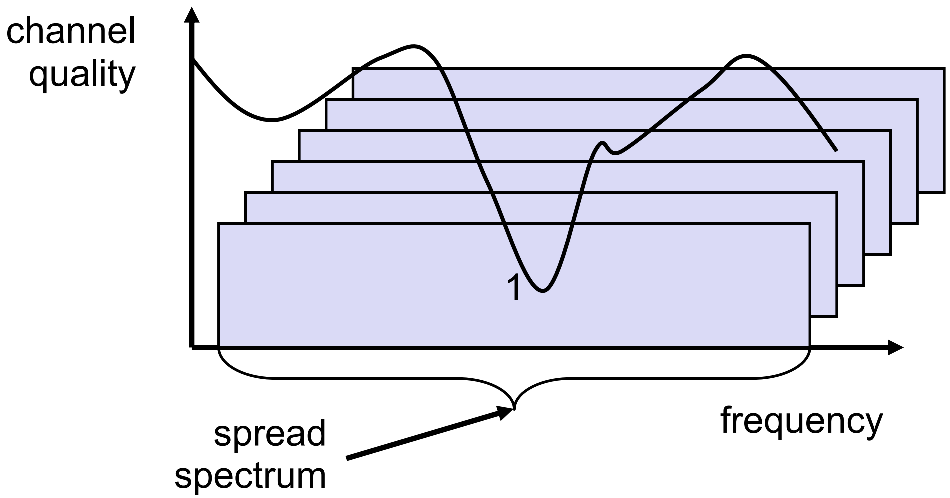
"It is mostly in the the choices about efficient and effective protocol components, data structures and algorithms and HW advances, used in a way based on correct assumptions and by exploiting in the most practical and "intelligent" way the opportunities to turn a drawback or limits into a practical advantage or synergy".

Previous examples clarify the active and relevant role of protocols to achieve a transmission potential in a harsh world.

Spreading and frequency selective fading



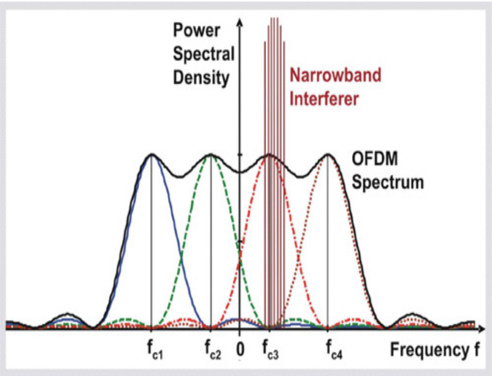
narrowband channels

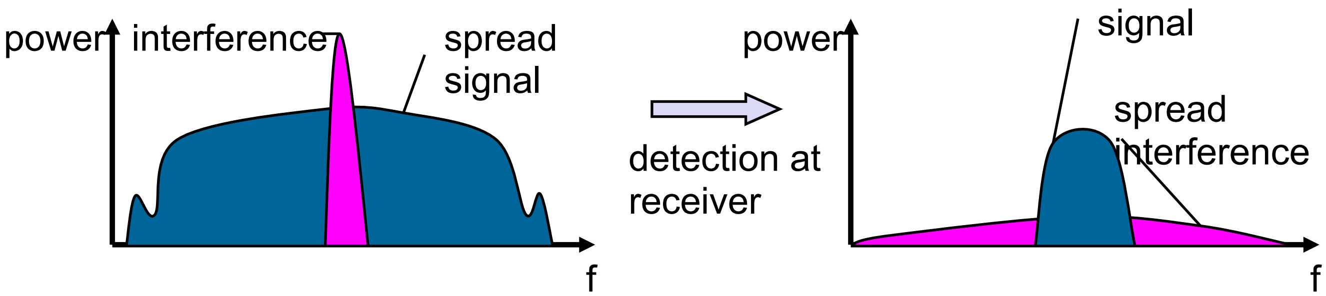


spread spectrum channels

Spread spectrum technology

- **Problem of radio transmission: frequency dependent fading can wipe out narrow band signals for duration of the interference**
- **E.g. DSSS modulation and correspondent CDMA access technique spread narrowband signal into a broadband signal using special code**
- **protection against narrow band interference**





protection against narrowband interference

- **Side effects:**
  - coexistence of several signals without dynamic coordination
  - tap-proof (cannot be detected without knowing the code)
- **Spread spectrum modulation Alternatives: Direct Sequence, Frequency Hopping**

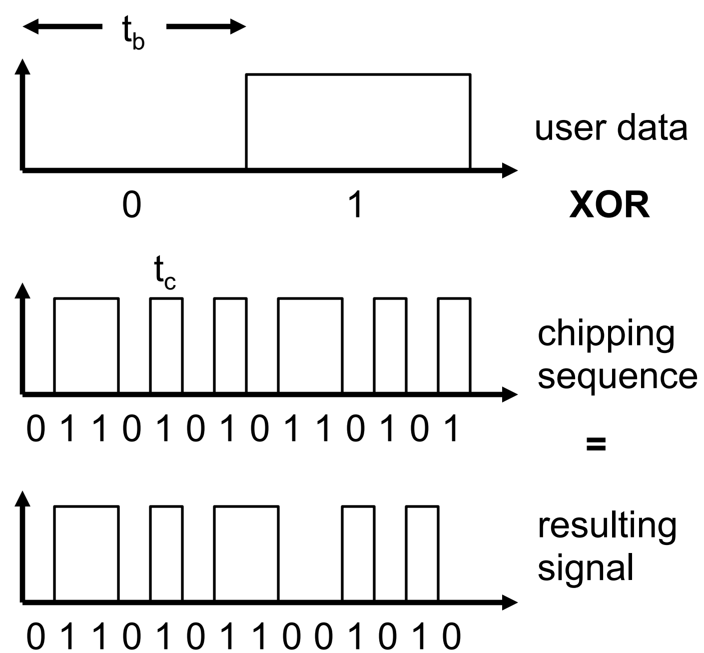
Spread spectrum technology

- **intuitive example: narrowband interference effect on transmission:**
  - transmit “Hello World” coded using narrowband “yellow” frequency and broadband “many colors” frequencies

  - a burst of yellow interference adds to the signal for a significant time: what is the result at the receiver?

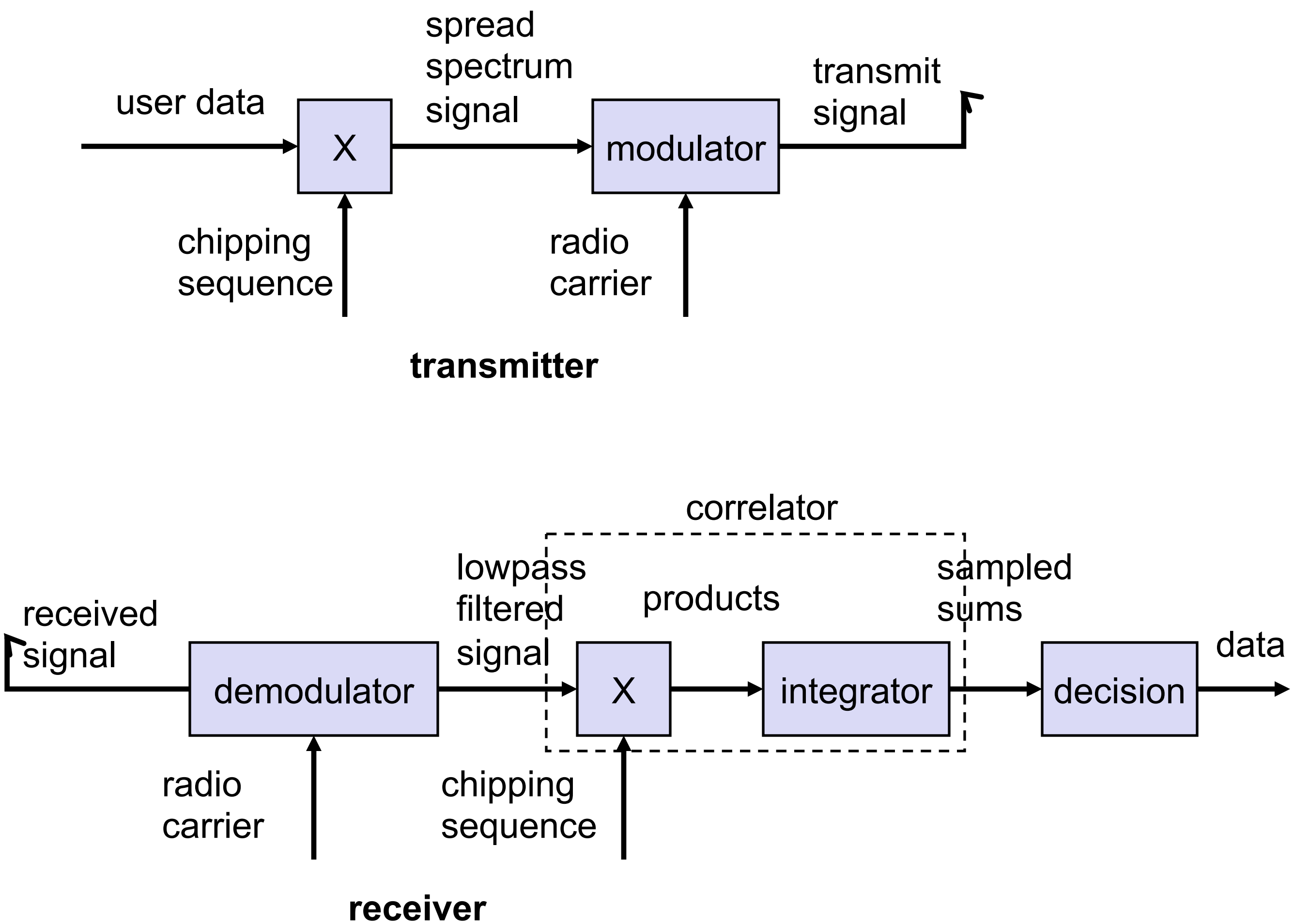
DSSS (Direct Sequence Spread Spectrum) I

- **XOR of the signal with pseudo-random number (chipping sequence, or Barker sequence)**
  - many chips per bit (e.g., 128) result in higher bandwidth of the signal (low throughput)
- **Advantages**
  - reduces frequency selective fading
  - in cellular networks
    - base stations can use the same frequency range
    - several base stations can detect and recover the signal
    - soft handover
- **Disadvantages**
  - precise power control and synchronization necessary



t_b: bit period
t_c: chip period

DSSS (Direct Sequence Spread Spectrum) II



DSSS (Direct Sequence Spread Spectrum) III

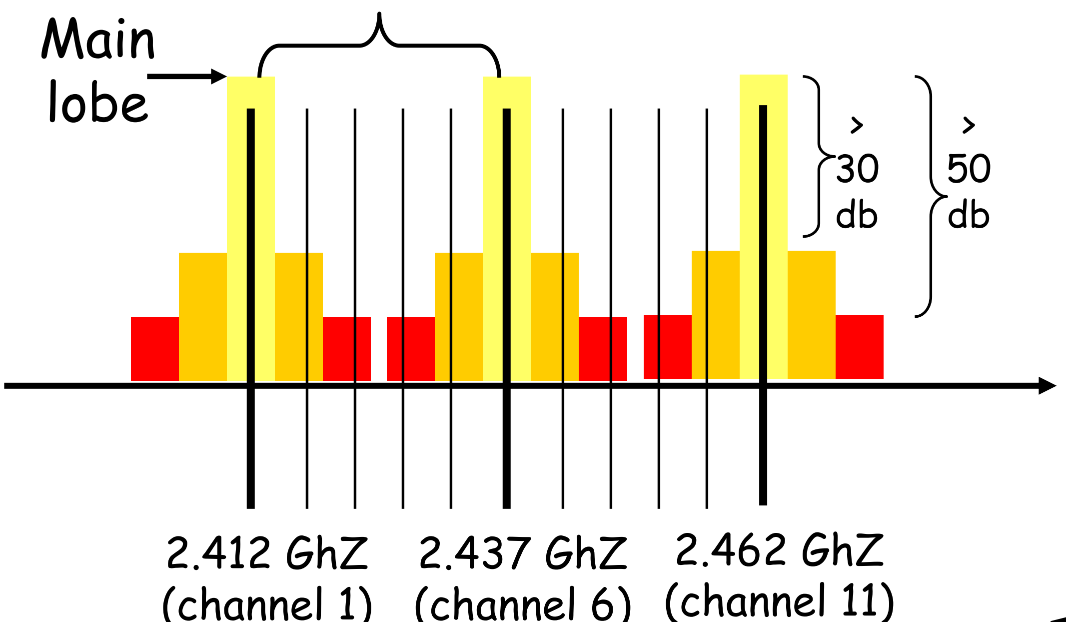
IEEE 802.11b DSSS channel frequency assignment

Channel ID	Channel (center) frequencies (GhZ)
1	2.412
2	2.417
3	2.422
4	2.427
5	2.432
6	2.437
7	2.442
8	2.447
9	2.452
10	2.457
11	2.462
12	2.467
13	2.472
14	2.484

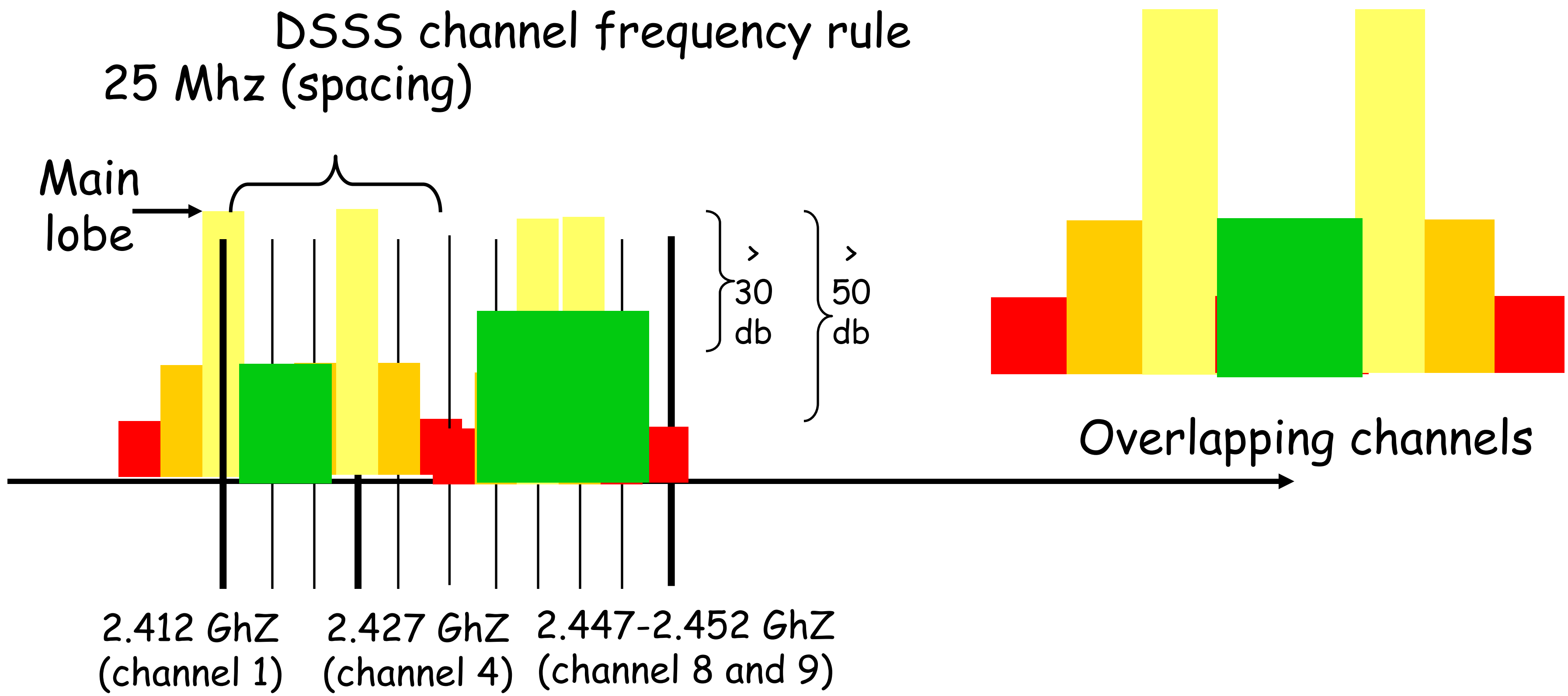
Non Overlapping channels

DSSS channel frequency rule

25 Mhz (spacing)



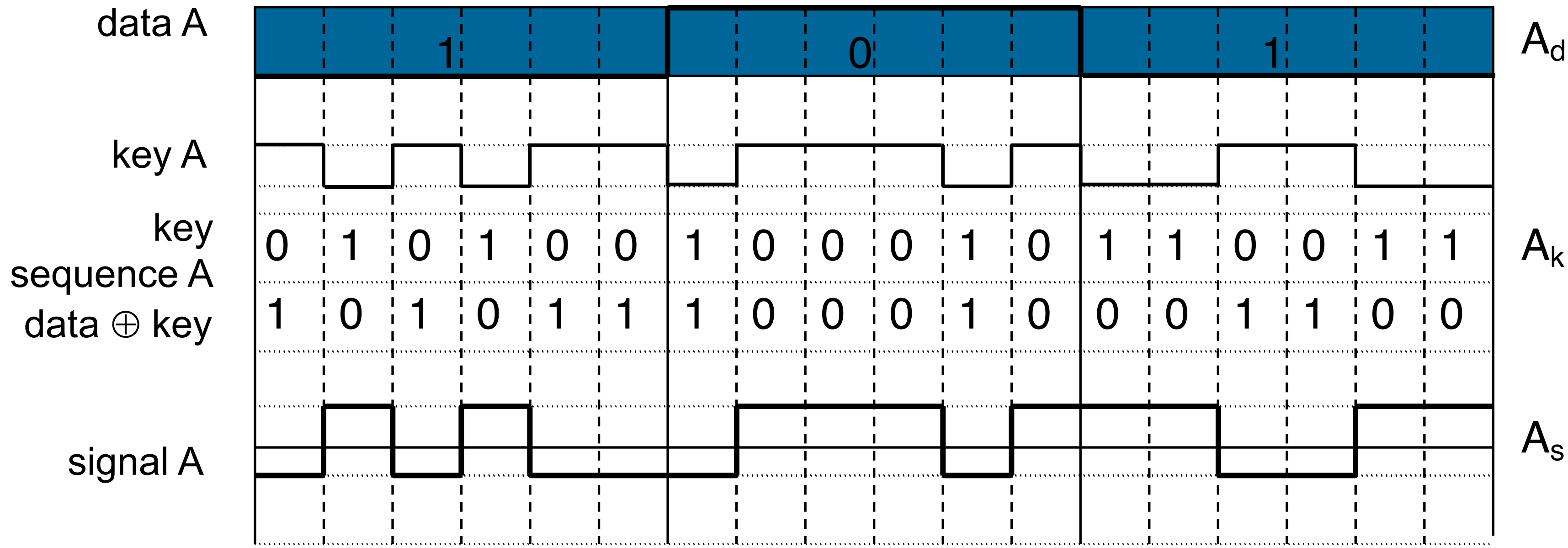
DSSS (Direct Sequence Spread Spectrum) IV



Code Division Multiple Access (CDMA)

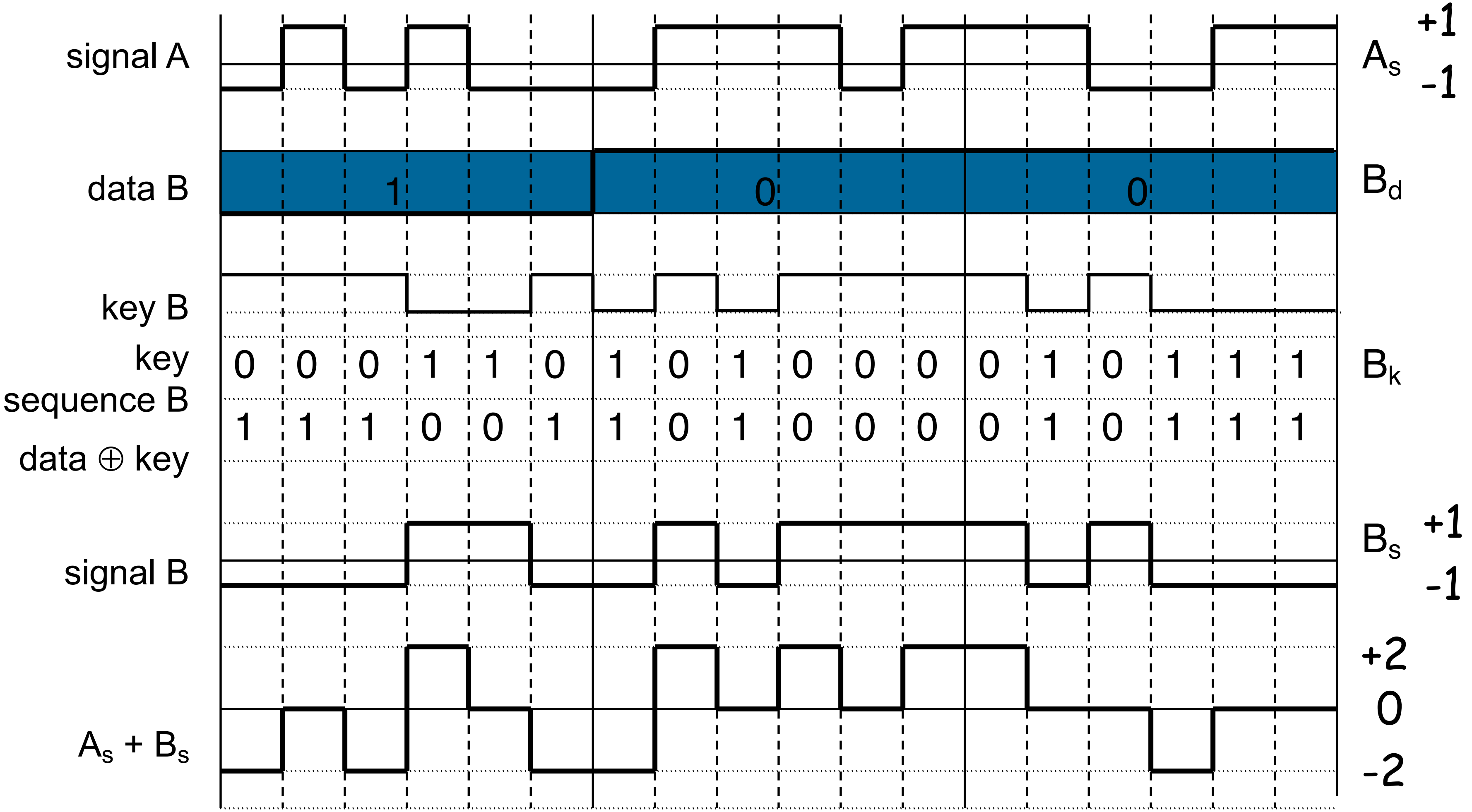
- **Sender A**
  - sends $A_d = 1$, key $A_k = 010011$ (assign: „0“= -1, „1“= +1) = (-1, +1, -1, -1, +1, +1)
  - sending signal $A_s = A_d * A_k = (-1, +1, -1, -1, +1, +1)$
- **Sender B**
  - sends $B_d = 0$, key $B_k = 110101$ (assign: „0“= -1, „1“= +1) = (+1, +1, -1, +1, -1, +1)
  - sending signal $B_s = B_d * B_k = (-1, -1, +1, -1, +1, -1)$
- **Both signals superimpose in space**
  - interference neglected (noise etc.)
  - $A_s + B_s = (-2, 0, 0, -2, +2, 0)$
- **Receiver wants to receive signal from sender A**
  - apply key A_k bitwise (inner product)
    - $A_e = (-2, 0, 0, -2, +2, 0) \bullet A_k = 2 + 0 + 0 + 2 + 2 + 0 = 6$
    - result greater than 0, therefore, original bit was „1“
  - receiving B
    - $B_e = (-2, 0, 0, -2, +2, 0) \bullet B_k = -2 + 0 + 0 - 2 - 2 + 0 = -6$, i.e. „0“

CDMA on signal level I

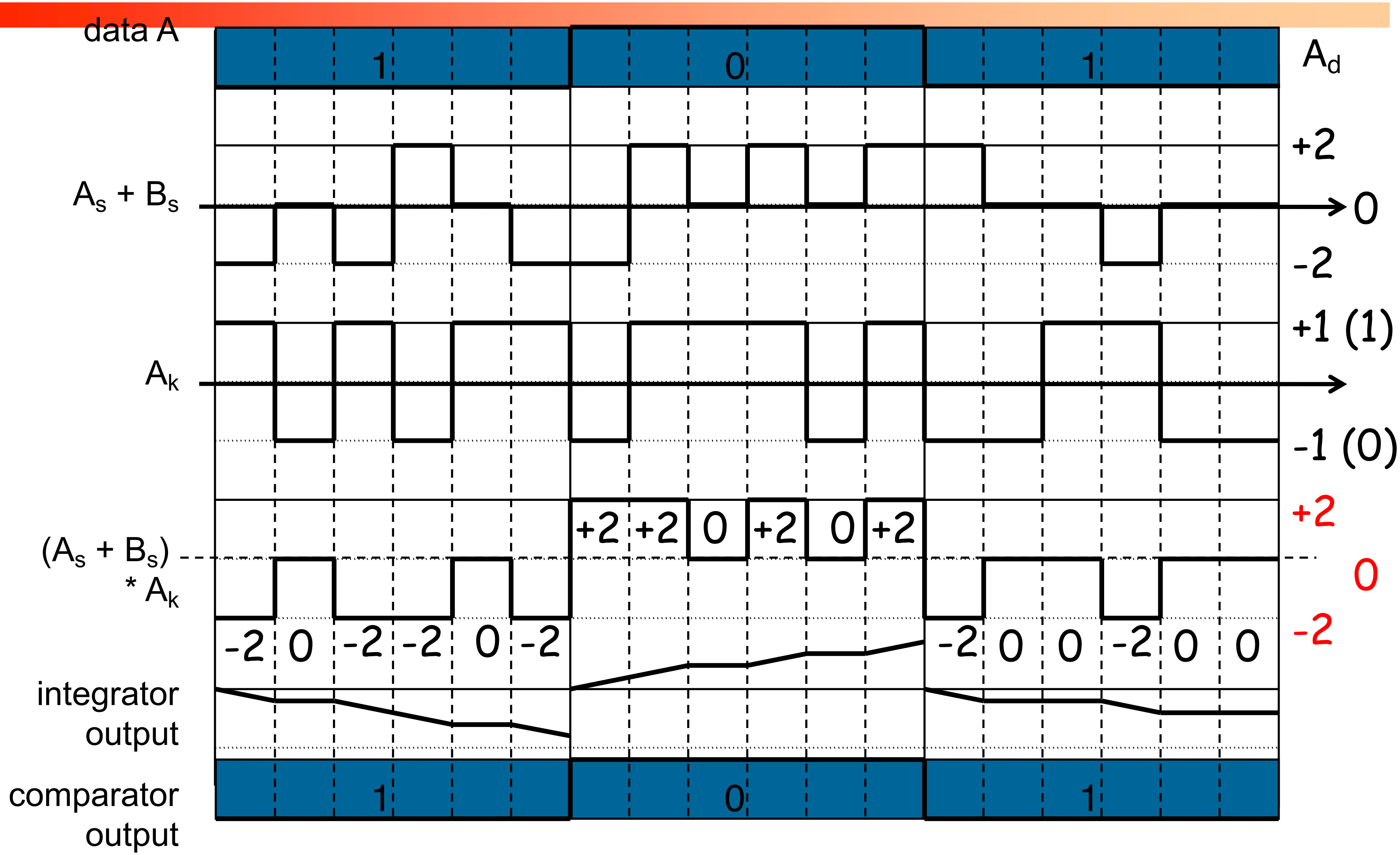


Real systems use much longer keys resulting in a larger distance between single code words in code space.

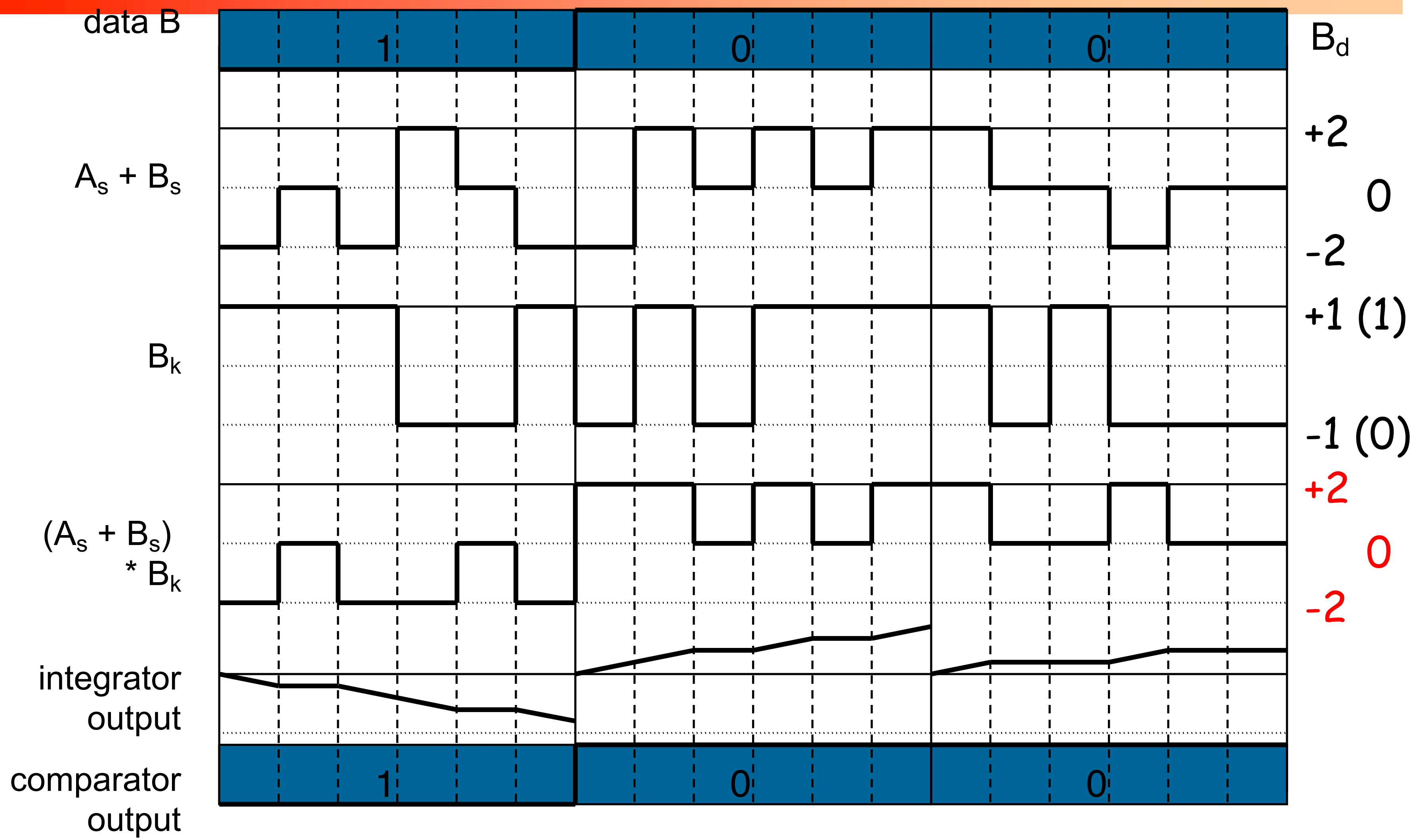
CDMA on signal level II



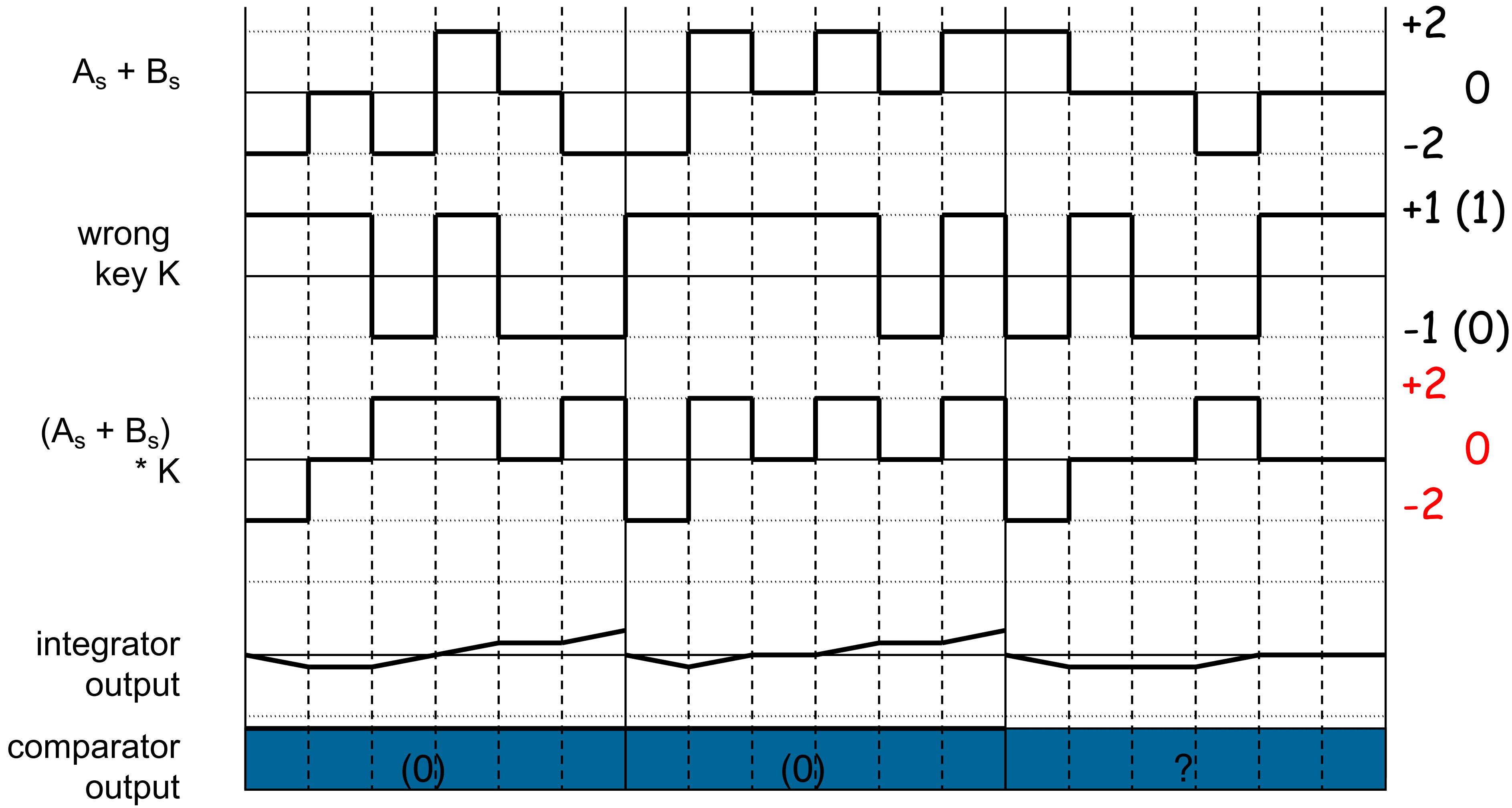
CDMA on signal level III



CDMA on signal level IV



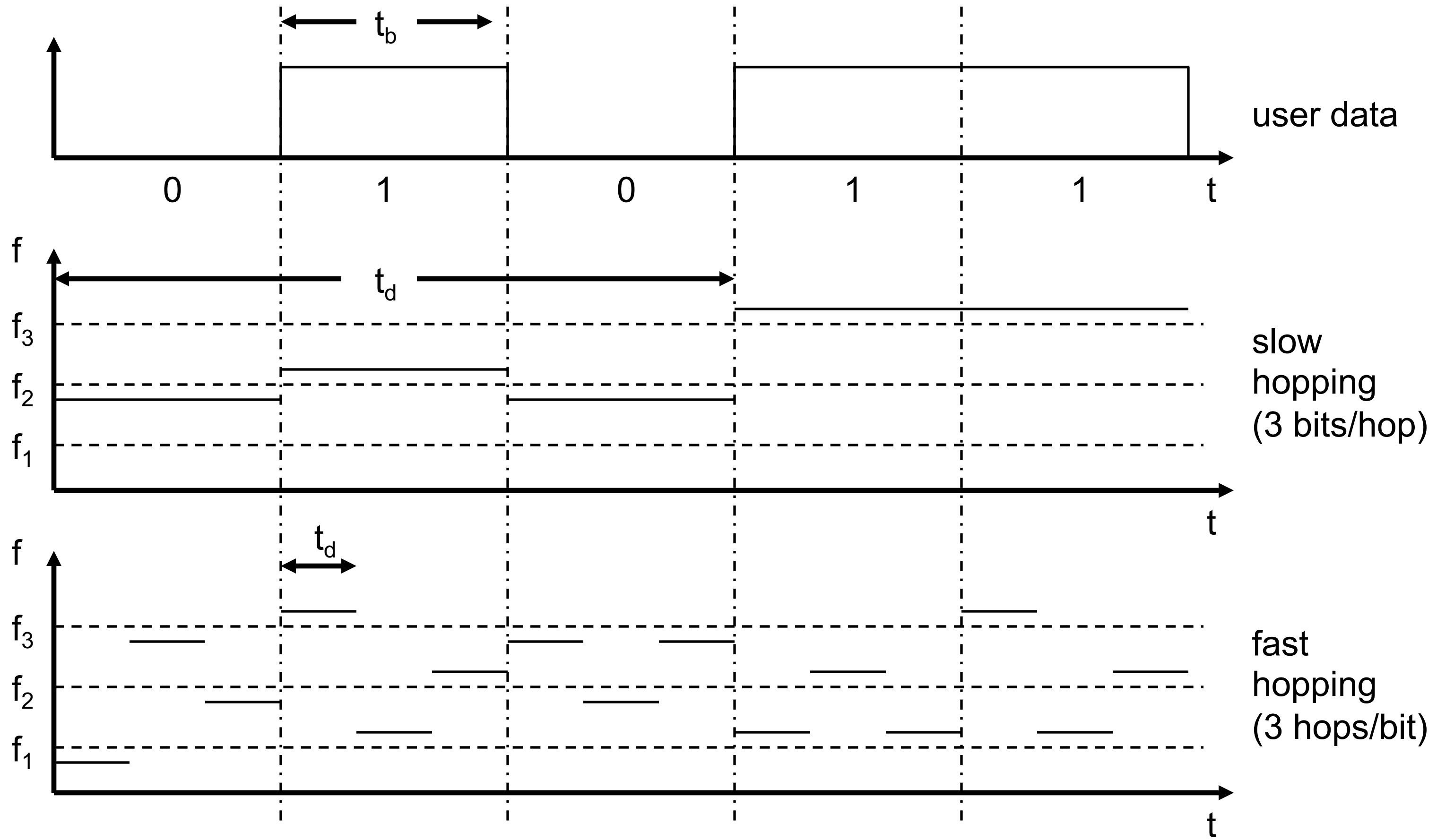
CDMA on signal level V



FHSS (Frequency Hopping Spread Spectrum) I

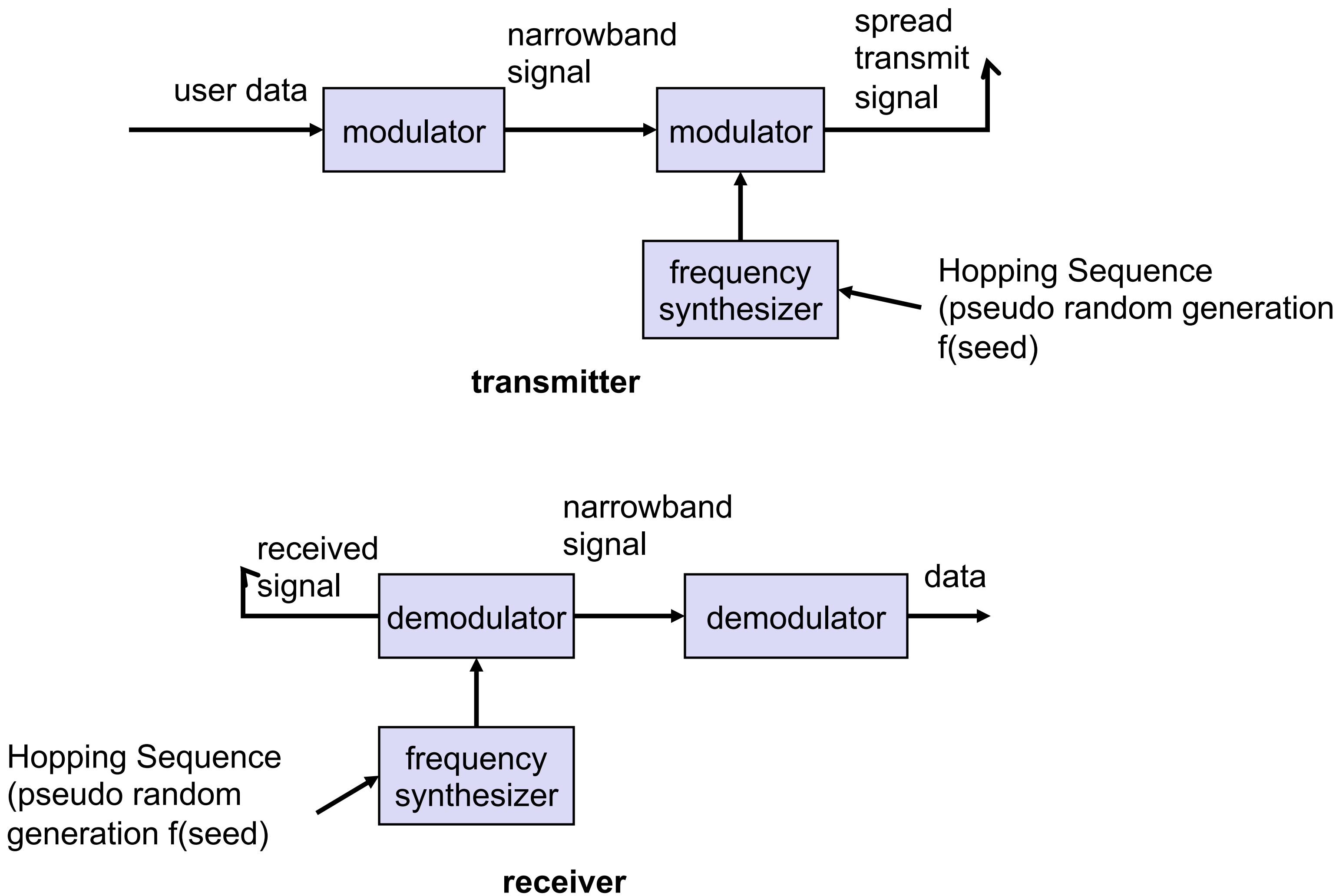
- **Discrete changes of carrier frequency**
  - sequence of frequency changes determined via pseudo random number sequence (e.g. seed = f(host identifier in Bluetooth))
- **Two versions**
  - Fast Hopping:
    several frequencies per user bit
  - Slow Hopping:
    several user bits per frequency
- **Advantages**
  - frequency selective fading and interference limited to short period
  - simple implementation
  - uses only small portion of spectrum at any time
- **Disadvantages**
  - not as robust as DSSS
  - simpler to detect

FHSS (Frequency Hopping Spread Spectrum) II



t_b: bit period t_d: dwell time

FHSS (Frequency Hopping Spread Spectrum) III

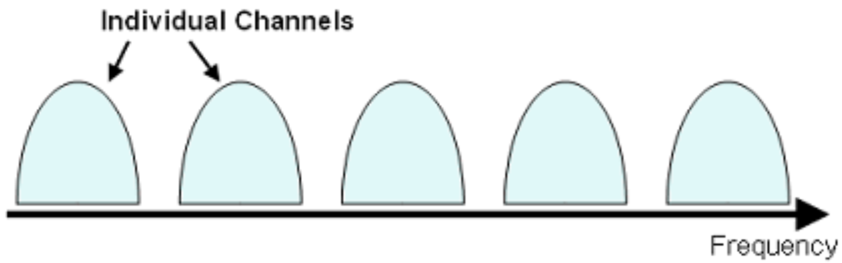


OFDM

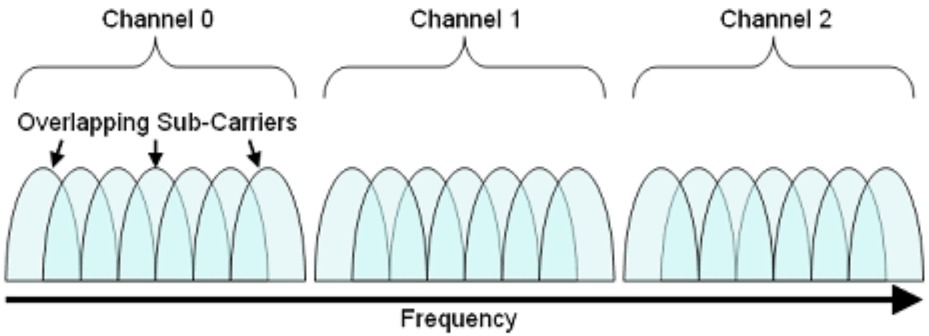
- **Very accurate adjacent communication channels**
  - Transmit data concurrently in parallel subcarriers
    - No need for separate filter for each sub-channel (like in FDM)
    - High bandwidth efficiency (see Nyquist rate formula)
    - Problem: doppler shift (at high speed) of subcarrier frequencies
  - Harmonics cancelation, low cost Fast Fourier Transform chips
  - Convolution coding (error correction with redundant information)
    - More or less similar to: subcarriers transmit “parity bit”
  - OFDM channels: 20 Mhz divided in 52 sub-carriers (300 Khz)
    - 4 subcarriers used as pilot (management)
    - 48 subcarriers used for data (symbols coding = 1 symbol per subcarrier at a time) = 48 concurrent symbols
  - OFDM in 802.11g is not compatible with DSSS in 802.11b!

OFDM

- Reprise: Frequency division multiplexing (FDM)
    - non overlapping channels



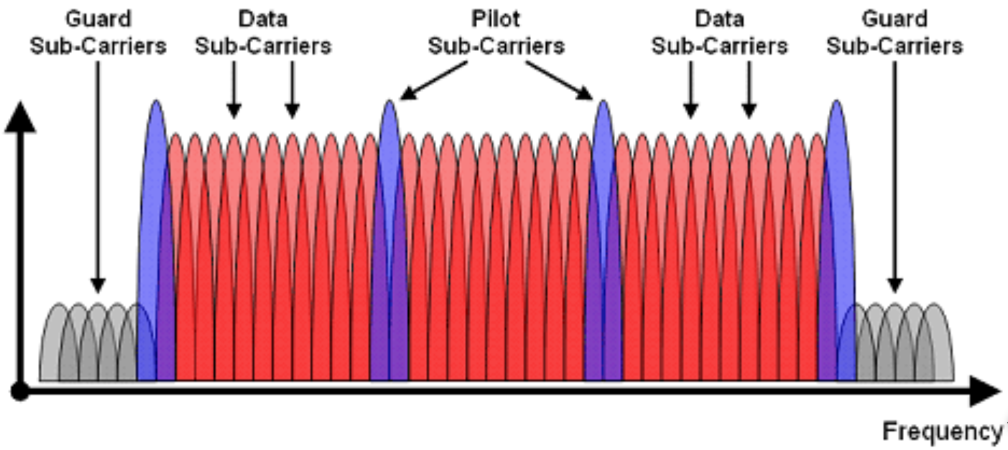
- OFDM: frequency division multiplexing in which a single channel utilizes multiple sub-carriers on adjacent overlapping frequencies



- Spectral efficiency (no guard space)
- Better symbol rate

e.g. OFDM

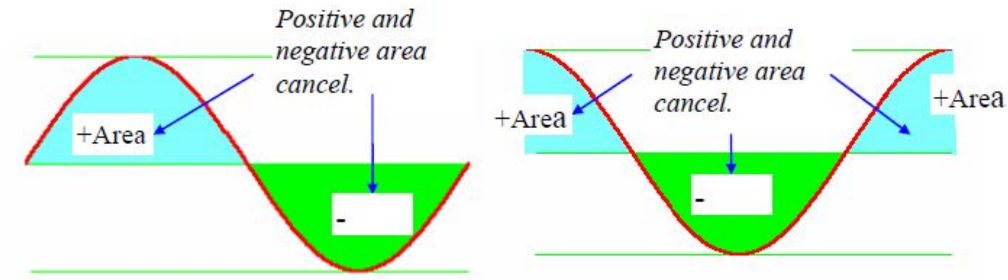
- **E.g. IEEE 802.16 (WiMAX):** internet access across long wireless communications links (up to 30 miles)
- 1 OFDM channel = 128 to 2048 sub-carriers
  - 1 sub-carrier bandwidth: 9.76 Khz (11.16 kHz in practice)
    - In 1.25 MHz there is space up to 128 subcarriers
    - …Up to 20 MHz (2048 subcarriers)
  - BPSK, QPSK, 16-QAM, or 64-QAM modulation

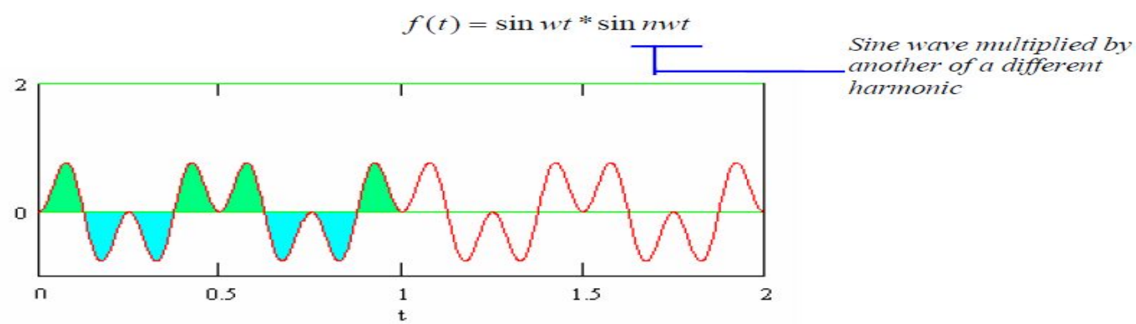


N.B. Frequency

How OFDM works

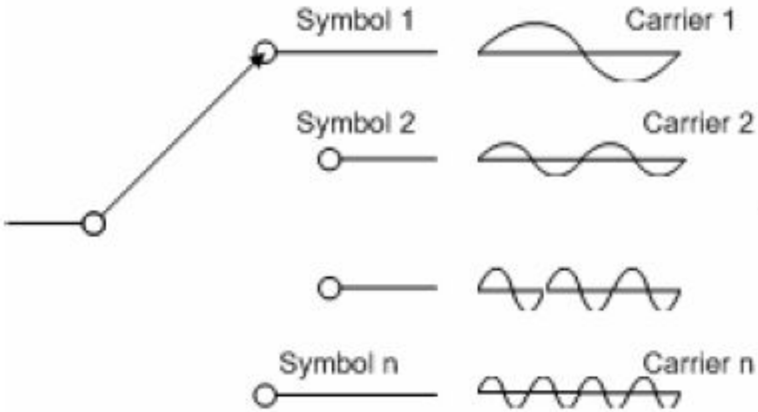
- **1- The importance of orthogonal subcarriers**
  - **Sin(x) * Sin(kx) = orthogonal signal (Harmonics orthogohality)**
  - **Cos(x) * Cos(kx) = orthogonal signal**
  - **In general, all sin(mx), sin(nx), cos(nx), cos(mx) are orthogonal**
  - Orthogonal means that integral of signal (t) is zero over period T. This allows simultaneous transmissions on different carriers with no interference

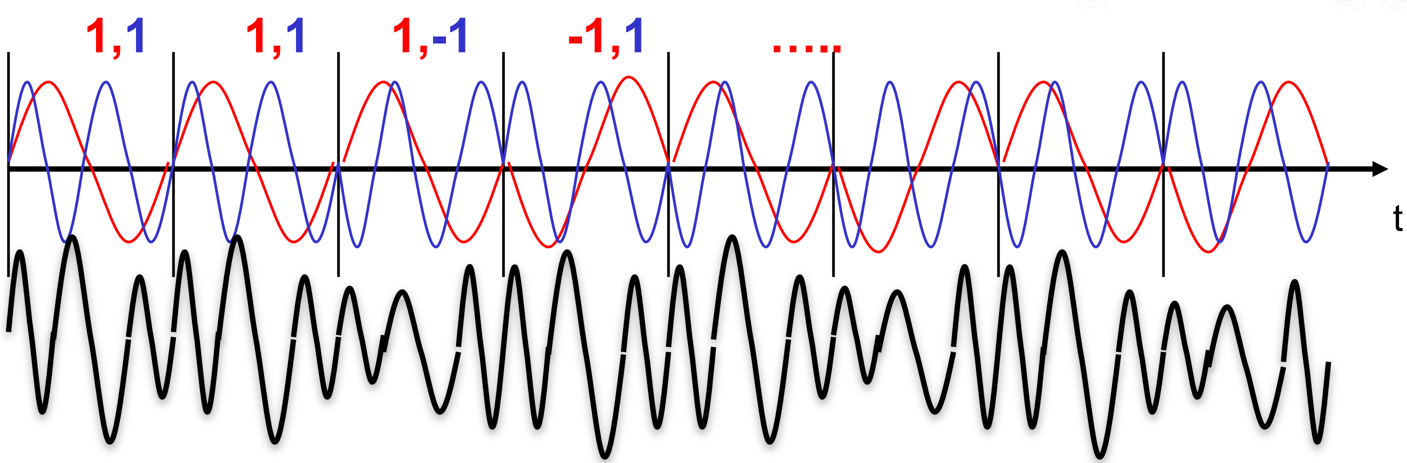




How OFDM works

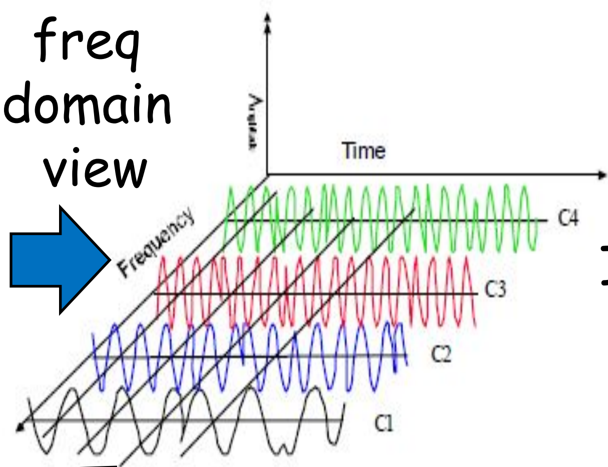
- E.g. OFDM with 4 carriers, 1 symbol per second (total for 4 carriers)
- Bit stream to be modulated (replace 0 with -1): 1 1 -1 -1 1 1 1 -1 1 -1….
- Split the bit sequence in 4 sub-sequences
  - C1 (1 Hz): 1 1 1 -1 1 -1 1 -1…
  - C2 (2 Hz): 1 1 -1 1 -1 -1 -1 1….
  - C3 (3 Hz): -1 1 1 1 1 -1 -1 1….
  - C4 (4 Hz): -1 -1 1 -1 -1 1 1 -1 ….





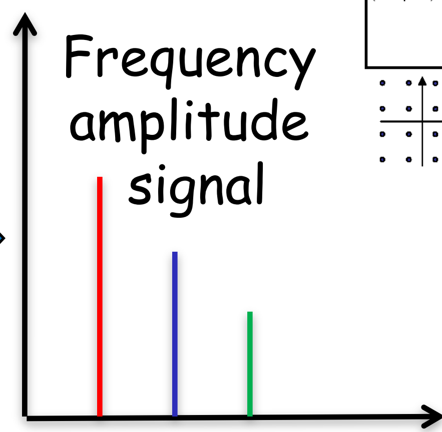
How OFDM works

freq domain view



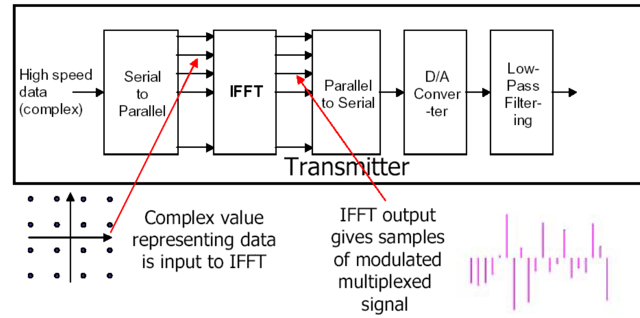
time domain view

IFFT / FFT



Frequency amplitude signal

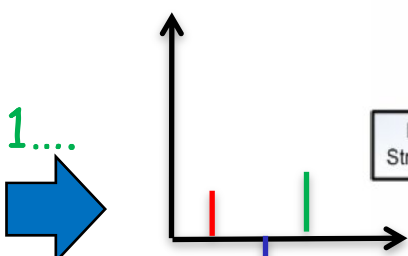
freq



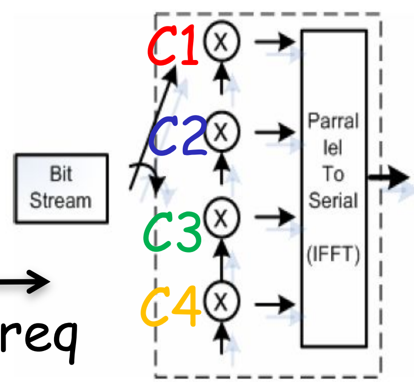
Now think at C1, C2… bits like if they are variable amplitudes of frequencies…

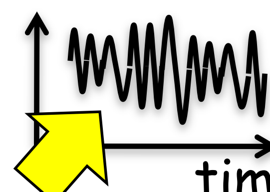
…and apply the IFFT to transform it in a time domain signal:

1 -1 1….



freq





time

time domain signal

…and viceversa!

Summary of OFDM

- **OFDM encoding: ≈ 250.000 phase modulations per second**

Data Rate (Mbps)	modulation	Bits coded per phase transition	R = fraction of carriers used for convolution	Length of 1 symbol at the given data rate (#subcarriers * bits coded per symbol)	Data bits encoded in 1 symbol
6	DBPSK	1	1/2	48	24
9	DBPSK	1	3/4	48	36
12	DQPSK	2	1/2	96	48
18	DQPSK	2	3/4	96	72
24	16-QAM	4	1/2	192	96
36	16-QAM	4	3/4	192	144
48	64-QAM	6	2/3	288	192
54	64-QAM	6	3/4	288	216