N) Wireshark/OSI

Friday, 3 January 2025 09.28

Husk der findes et søge felt.

Man kan udvide de forskellige dele og se mere information når man gå ind og kigger.

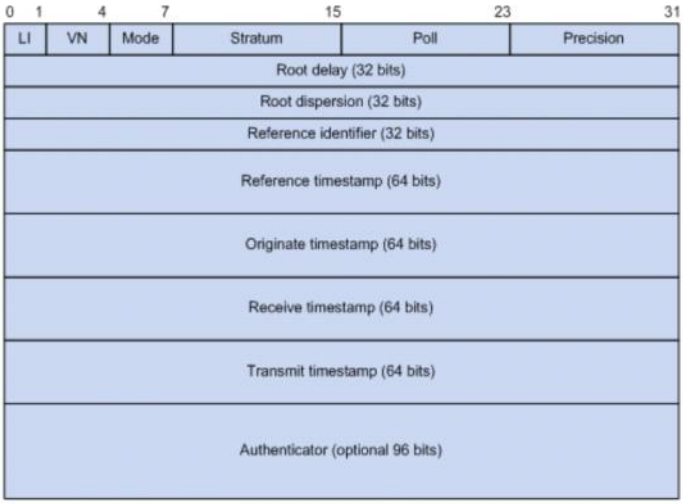
NTP:

NTP message format

NTP uses two types of messages: clock synchronization and NTP control messages. All NTP messages mentioned in this document refer to NTP clock synchronization messages. NTP control messages are used in environments where network management is needed. Because NTP control messages are not essential for clock synchronization, they are not described in this document.

A clock synchronization message is encapsulated in a UDP message, in the format shown in Figure 18.

Figure 18: Clock synchronization message format



Main fields are described as follows:

- **LI (Leap Indicator)**—A 2-bit leap indicator. When set to 11, it warns of an alarm condition (clock unsynchronized); when set to any other value, it is not to be processed by NTP.
- **VN (Version Number)**—A 3-bit version number that indicates the version of NTP. The latest version is version 4.
- **Mode**—A 3-bit code that indicates the work mode of NTP. This field can be set to these values:
  - **0**—reserved
  - **1**—symmetric active
  - **2**—symmetric passive
  - **3**—client
  - **4**—server
  - **5**—broadcast or multicast
  - **6**—NTP control message
  - **7**—reserved for private use.
- **Stratum**—An 8-bit integer that indicates the stratum level of the local clock, with the value ranging from 1 to 16. Clock precision decreases from stratum 1 through stratum 16. A stratum 1 clock has the highest precision, and a stratum 16 clock is not synchronized and cannot be used as a reference clock.
- **Poll**—An 8-bit signed integer that indicates the maximum interval between successive messages, which is called the poll interval.
- **Precision**—An 8-bit signed integer that indicates the precision of the local clock.
- **Root Delay**—Roundtrip delay to the primary reference source.
- **Root Dispersion**—The maximum error of the local clock relative to the primary reference source.
- **Reference Identifier**—Identifier of the particular reference source.
- **Reference Timestamp**—The local time at which the local clock was last set or corrected.
- **Originate Timestamp**—The local time at which the request departed from the client for the service host.
- **Receive Timestamp**—The local time at which the request arrived at the service host.
- **Transmit Timestamp**—The local time at which the reply departed from the service host for the client.
- **Authenticator**—Authentication information.

Find DNS information fra DHCP

```
    1 0.000000000   192.168.122.1     224.0.0.251       MDNS      82 Standard query 0x0000 PTR _googlecast._tcp.local, "QM" question
    2 0.000160947   192.168.122.1     224.0.0.251       MDNS      82 Standard query 0x0000 PTR _googlecast._tcp.local, "QM" question
    3 1.000552726   192.168.122.1     224.0.0.251       MDNS      82 Standard query 0x0000 PTR _googlecast._tcp.local, "QM" question
    4 1.000693942   192.168.122.1     224.0.0.251       MDNS      82 Standard query 0x0000 PTR _googlecast._tcp.local, "QM" question
    5 2.343419534   0.0.0.0           255.255.255.255   DHCP     335 DHCP Discover - Transaction ID 0xbb0bc02e
    6 2.343618497   192.168.122.1     192.168.122.43    DHCP     342 DHCP Offer    - Transaction ID 0xbb0bc02e
    7 2.348326426   0.0.0.0           255.255.255.255   DHCP     347 DHCP Request  - Transaction ID 0xbb0bc02e
    8 2.348490619   192.168.122.1     192.168.122.43    DHCP     345 DHCP ACK      - Transaction ID 0xbb0bc02e
    9 3.002276047   192.168.122.1     224.0.0.251       MDNS      82 Standard query 0x0000 PTR _googlecast._tcp.local, "QM" question
   10 3.002614385   192.168.122.1     224.0.0.251       MDNS      82 Standard query 0x0000 PTR _googlecast._tcp.local, "QM" question
   11 3.760636291   192.168.122.1     192.168.122.255   UDP       86 57621 → 57621 Len=44
   12 5.699378825   52:54:00:7f:e3:df Broadcast          ARP       42 Who has 192.168.122.1? Tell 192.168.122.43
   13 5.699405510   52:54:00:26:63:d3 52:54:00:7f:e3:df  ARP       42 192.168.122.1 is at 52:54:00:26:63:d3
   14 5.699509813   192.168.122.43    192.168.122.1     DNS       87 Standard query 0x8c45 A api.snapcraft.io OPT
   15 5.699565537   192.168.122.43    192.168.122.1     DNS       87 Standard query 0x4e48 AAAA api.snapcraft.io OPT
   16 5.699803865   192.168.122.1     192.168.122.43    DNS       87 Standard query response 0x4e48 AAAA api.snapcraft.io OPT
   17 5.700205858   192.168.122.1     192.168.122.43    DNS      183 Standard query response 0x8c45 A api.snapcraft.io A 91.189.92.20 A
   18 5.824067213   192.168.122.43    192.168.122.1     DNS       87 Standard query 0x99ea AAAA api.snapcraft.io OPT
   19 5.824134404   192.168.122.1     192.168.122.43    DNS       87 Standard query response 0x99ea AAAA api.snapcraft.io OPT
   20 34.377316123  192.168.122.43    192.168.122.1     DNS       85 Standard query 0xf238 A ntp.ubuntu.com OPT
   21 34.377652061  192.168.122.43    192.168.122.1     DNS       85 Standard query 0xe4df AAAA ntp.ubuntu.com OPT
   22 34.378541941  192.168.122.1     192.168.122.43    DNS      149 Standard query response 0xf238 A ntp.ubuntu.com A 91.189.91.157 A
   23 34.378644112  192.168.122.1     192.168.122.43    DNS      141 Standard query response 0xe4df AAAA ntp.ubuntu.com AAAA 2001:67c:1
   24 34.380693425  192.168.122.43    91.189.91.157     NTP       90 NTP Version 4, client
   25 34.480819221  91.189.91.157     192.168.122.43    NTP       90 NTP Version 4, server
   26 60.287131027  192.168.122.1     192.168.122.255   UDP       86 57621 → 57621 Len=44
   27 66.657850615  192.168.122.43    91.189.91.157     NTP       90 NTP Version 4, client
   28 66.758738234  91.189.91.157     192.168.122.43    NTP       90 NTP Version 4, server
   29 78.772120824  192.168.122.43    192.168.122.1     UDP       72 38384 → 5678 Len=30
```

```
Seconds elapsed: 1
▸ Bootp flags: 0x0000 (Unicast)
Client IP address: 0.0.0.0
Your (client) IP address: 192.168.122.43
Next server IP address: 192.168.122.1
Relay agent IP address: 0.0.0.0
Client MAC address: 52:54:00:7f:e3:df (52:54:00:7f:e3:df)
Client hardware address padding: 00000000000000000000
Server host name not given
Boot file name not given
Magic cookie: DHCP
▸ Option: (53) DHCP Message Type (Offer)
▸ Option: (54) DHCP Server Identifier (192.168.122.1)
▸ Option: (51) IP Address Lease Time
▸ Option: (58) Renewal Time Value
▸ Option: (59) Rebinding Time Value
▸ Option: (1) Subnet Mask (255.255.255.0)
▸ Option: (28) Broadcast Address (192.168.122.255)
▸ Option: (3) Router
▾ Option: (6) Domain Name Server
    Length: 4
    Domain Name Server: 192.168.122.1
▸ Option: (255) End
  Padding: 000000000000000
```

```
0000  52 54 00 7f e3 df
0010  01 48 26 3a 00 00
0020  7a 2b 00 43 00 44
0030  c0 2e 00 01 00 00
0040  7a 01 00 00 00 00
0050  00 00 00 00 00 00
0060  00 00 00 00 00 00
0070  00 00 00 00 00 00
0080  00 00 00 00 00 00
0090  00 00 00 00 00 00
00a0  00 00 00 00 00 00
00b0  00 00 00 00 00 00
00c0  00 00 00 00 00 00
00d0  00 00 00 00 00 00
00e0  00 00 00 00 00 00
00f0  00 00 00 00 00 00
0100  00 00 00 00 00 00
0110  00 00 00 00 00 00
0120  a8 7a 01 33 04 00
0130  04 00 00 0c 4e 01
0140  ff 03 04 c0 a8 7a
0150  00 00 00 00 00 00
```

DNS: DHCP

DHCP offer har al information

- option 6 har 99.9% af tiden, DNS address

OSI model.

OSI (Open Source Interconnection) 7 Layer Model

Layer	Application/Example		Central Device/ Protocols			DOD4 Model
Application (7) Serves as the window for users and application processes to access the network services.	**End User layer** Program that opens what was sent or creates what is to be sent Resource sharing • Remote file access • Remote printer access • Directory services • Network management		**User Applications** SMTP		**GATEWAY** Can be used on all layers	Process
Presentation (6) Formats the data to be presented to the Application layer. It can be viewed as the "Translator" for the network.	**Syntax layer** encrypt & decrypt (if needed) Character code translation • Data conversion • Data compression • Data encryption • **Character Set Translation**		JPEG/ASCII EBDIC/TIFF/GIF PICT			
Session (5) Allows session establishment between processes running on different stations.	**Synch & send to ports** (logical ports) Session establishment, maintenance and termination • Session support - perform security, name recognition, logging, etc.		**Logical Ports** RPC/SQL/NFS NetBIOS names			
Transport (4) Ensures that messages are delivered error-free, in sequence, and with no losses or duplications.	**TCP** Host to Host, Flow Control Message segmentation • Message acknowledgement • Message traffic control • Session multiplexing	PACKET FILTERING	TCP/SPX/UDP			Host to Host
Network (3) Controls the operations of the subnet, deciding which physical path the data takes.	**Packets** ("letter", contains IP address) Routing • Subnet traffic control • Frame fragmentation • Logical-physical address mapping • Subnet usage accounting		**Routers** IP/IPX/ICMP			Internet
Data Link (2) Provides error-free transfer of data frames from one node to another over the Physical layer.	**Frames** ("envelopes", contains MAC address) [NIC card — Switch — NIC card] (end to end) Establishes & terminates the logical link between nodes • Frame traffic control • Frame sequencing • Frame acknowledgment • Frame delimiting • Frame error checking • Media access control		**Switch Bridge WAP** PPP/SLIP	Land Based Layers		Network
Physical (1) Concerned with the transmission and reception of the unstructured raw bit stream over the physical medium.	**Physical structure** Cables, hubs, etc. Data Encoding • Physical medium attachment • Transmission technique - Baseband or Broadband • Physical medium transmission Bits & Volts		**Hub**			

N) bit sending

Friday, January 3, 2025 10:30 AM

Through put er alt det du sender som kommer igennem, succesfult.

Good put, er det data vi så kan bruge

Protocol effciency $e_p = \frac{e_G}{e_T} = \frac{64}{67} = 0.96$

Good put $e_G = 1000 \cdot 64 \cdot 0.96 = 960 \left[\frac{bit}{s}\right]$

data rate
1k bit/s

things we will do to deal with errors

- introduce a mechanism to detect errors
  - Bob will know if the received packet has errors or not
- introduce a feedback link
  - Bob can tell Alice if the packet has errors
- improve the reliability of the preamble
  - the packet is not missed if 1 is interpreted as 0
- enable Alice to retransmit a packet again
  - eventually the packet arrives at Bob

error detection (2)

- parity check is a very weak error detection code
  - if errors occur in two different bit positions, then the packet is accepted as correct
- CRC (Cyclic Redundancy Check) codes
  - come in different bit lengths, e. g. 8, 16, 32
  - very high probability to detect errors
    - the packet is in error if bit errors occur in the CRC bits or the other bits in the packet
  - yet, it is not perfect
    - there is always a probability of undetected error
  - the value of the K check bits is a function of the other M bits

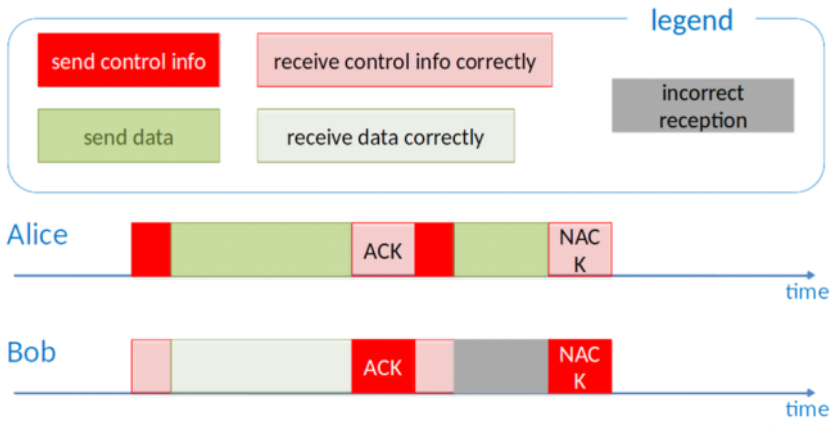
error detection (2)

- parity check is a very weak error detection code
  - if errors occur in two different bit positions, then the packet is accepted as correct
- CRC (Cyclic Redundancy Check) codes
  - come in different bit lengths, e. g. 8, 16, 32
  - very high probability to detect errors
    - the packet is in error if bit errors occur in the CRC bits or the other bits in the packet
  - yet, it is not perfect
    - there is always a probability of undetected error
  - the value of the K check bits is a function of the other M bits
    - it must happen that $2^{(M-K)}$ bit combinations have exactly the same parity check

N) protocol types

Friday, 3 January 2025 13.06

how should the TDD protocol work



feedback link from Bob (2)

- in a TDD operation, Alice and Bob agree upon the following protocol
  - after each packet sent by Alice, Bob starts his transmission
- Bob needs to send 1 bit (ACK or NACK)
  - but this is highly unreliable, Alice is not even able to detect if there is an error
- we assume that the ACK/NACK packet sent by Bob has a length of 1 byte more reliable transmission (to be elaborated later)
  - still only 1 data bit for Alice (ACK or NACK)
    - Q: does the ACK/NACK bit carry one bit of information?

Automatic Retransmission ReQuest (ARQ) protocol (1)

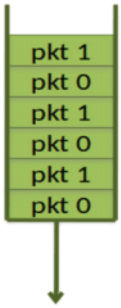
- upon receiving NACK, Alice resends the same packet
  - repeats this until receiving ACK for this packet
  - this is a stop-and-wait ARQ protocol
- note that NACK can be implicit
  - if no ACK arrives, Alice resends the same packet
  - this solves the situation when Bob erroneously detects the packet length

Automatic Retransmission reQuest (ARQ) protocol (2)

- a different challenge
  - error in ACK/NACK reception
- Example
  - Alice sends (1)(0101)(1)(011101)
  - Bob receives it correctly and sends ACK
  - error occurs and Alice receives NACK
  - Alice retransmits the packet (1)(0101)(1)(011101)
  - the problem: Bob thinks this is a new packet with data bits (011101), not a retransmission of the old one

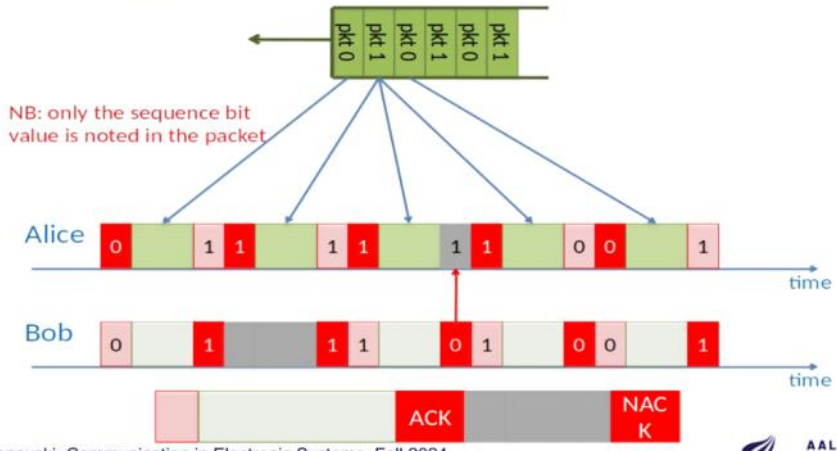
ARQ with sequence number (1)

- the problem is solved by a sequence number
- using a single-bit sequence number
  - Alice assigns modulo-2 numbers to the data packets in its queue



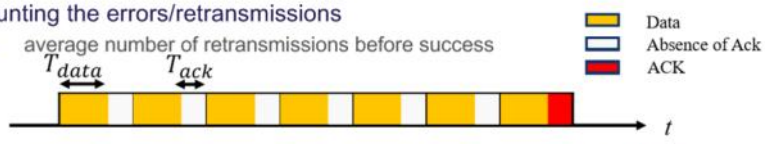
- Bob does not send ACK/NACK, but sends a bit to say which sequence bit value he expects next
- Alice increases the sequence number modulo 2 only when it is sure that Bob has received it

ARQ with sequence number (2)



ARQ performance

- counting the errors/retransmissions
  - average number of retransmissions before success



- probability of successful reception at the k-th attempt using the p_{PER}

$$p_k = \underbrace{p_{\mathrm{PER}} \cdots p_{\mathrm{PER}}}_{k-1 \text{ times}}(1 - p_{\mathrm{PER}}) = p_{\mathrm{PER}}^{k-1}(1 - p_{\mathrm{PER}})$$

- average number of delay

$$\bar{T} = (T_{data} + T_{ack})p_1 + 2(T_{data} + T_{ack})p_2 + \cdots = \sum_{k=0}^{\infty} k(T_{data} + T_{ack})p_k = \frac{T_{data} + T_{ack}}{1 - p_{\mathrm{PER}}}$$

medium access methods

1. all users are known?
2. is there any coordination among users?

scheduling (reserving)

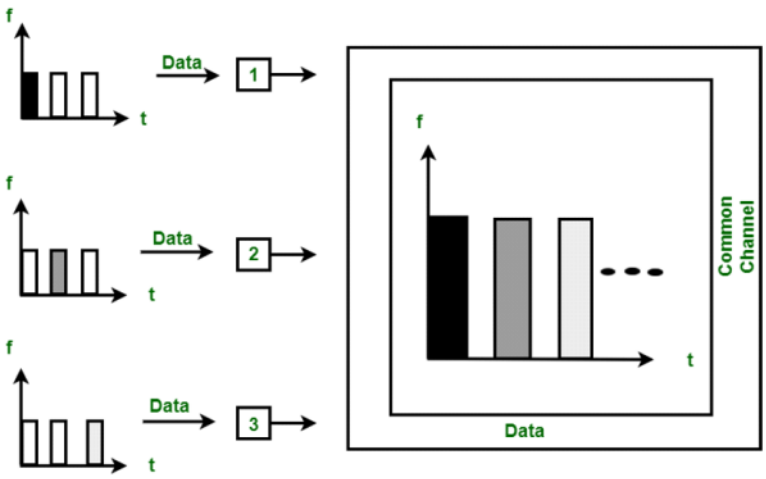
- scheduling: TDMA, FDMA
- token based, round-robin

free-for-all

- random access
  - ALOHA
  - slotted ALOHA
  - carrier sensing

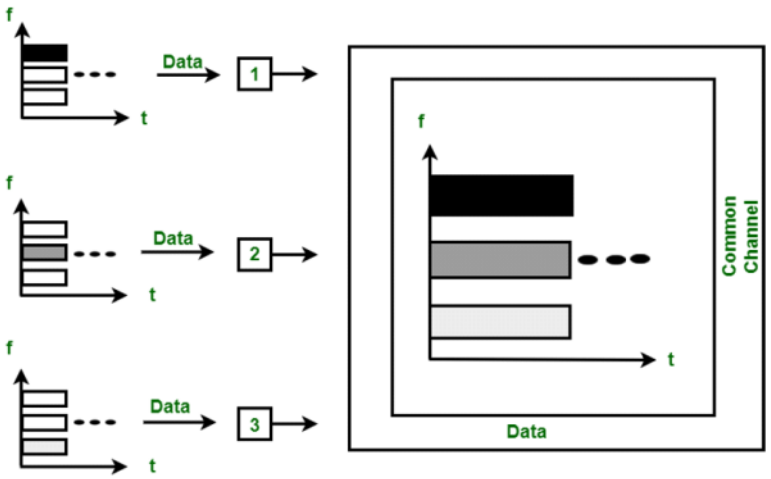
scheduling (1)

- TDMA
  - time-division multiple access



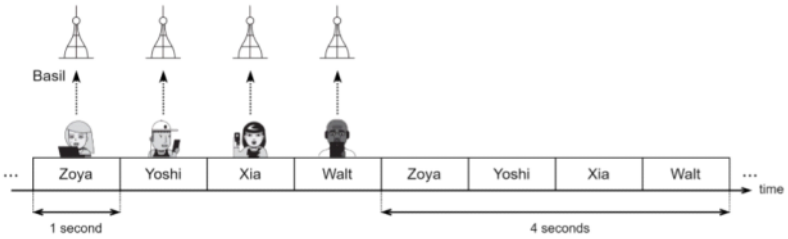
scheduling (2)

- FDMA
  - frequency-division multiple access



TDMA with periodic reservation (1)

- 4 users send packets to Basil (uplink)
  - packet rate $R = 1\ \text{kbit/s}$
  - one packet contains $1{,}000\ \text{bits}$
- Zoya's effective data rate
  - 2 frames $(8\ \text{s})$ to transmit $2{,}000\ \text{bits}$
  - $\frac{2{,}000}{8} = 0.25\ \text{kbit/s} < 1\ \text{kbit/s}$
- system throughput
  - $G = \frac{4{,}000}{4} = 1\ \text{kbit/s}$



TDMA with periodic reservation (2)

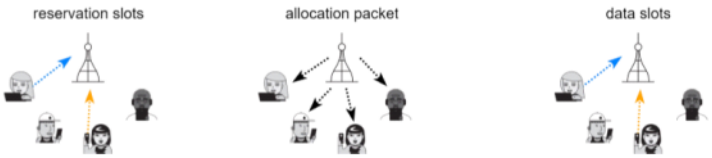
- the network traffic has changed
  - only Xia had data packets to send during 10 frames
  - system throughput: $G = \frac{10{,}000}{80} = 0.25\ \text{kbit/s} < 1\ \text{kbit/s}$

dynamic reservation and the cost of overhead (1)

- 4 users send packets to Basil (uplink)
  - reservation slots (users)
  - allocation packet (Basil)
  - data slots (users)

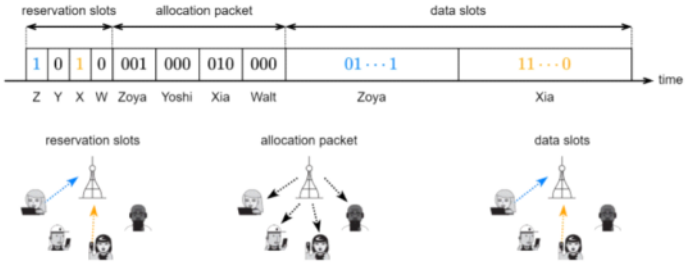
allocation packet	
000	No data slot reserved
001	1st data slot reserved
010	2nd data slot reserved
	...

reservation slots	
0	User has no packet to transmit
1	User has a packet to transmit



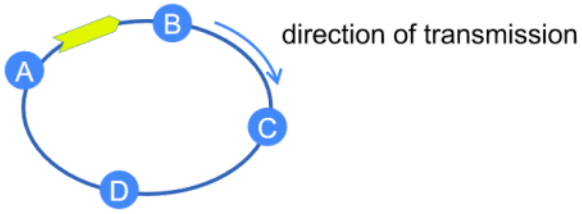
dynamic reservation and the cost of overhead (2)

- system throughput
  - 4 users transmitting / data packet: $1{,}000$ bits / $R = 1$ kbit/s
  - overhead: 16 bits or $\frac{16}{1{,}000} = 0.016$ s
  - throughput: $G = \frac{4{,}000}{0.016+4} = 0.996$ kbit/s ≈ 1 kbit/s
- 1 user transmitting
  - throughput: $G = \frac{1{,}000}{0.016+1} = 0.984$ kbit/s
- 1 sensor transmitting
  - data packet: 4 bits
  - throughput: $G = \frac{4}{0.016+0.004} = 0.2$ kbit/s

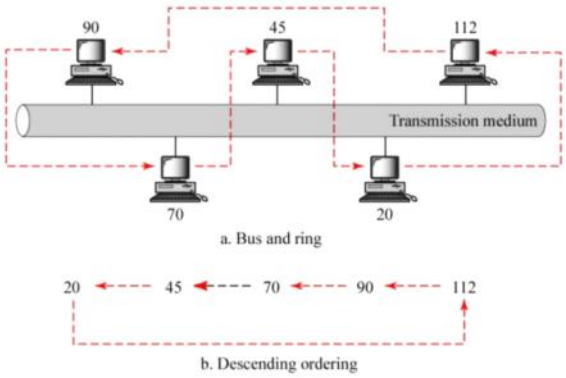


token based (1)

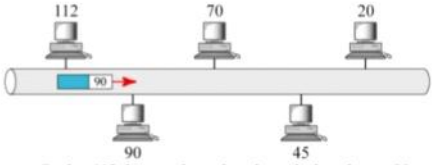
- control token passed from one node to next sequentially
- point-to-point links can be fast
- problems:
  - token overhead
  - latency
  - single point of failure (token)



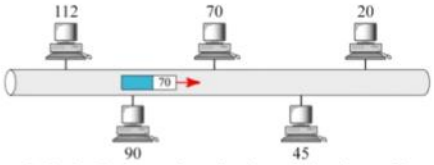
token based (2)



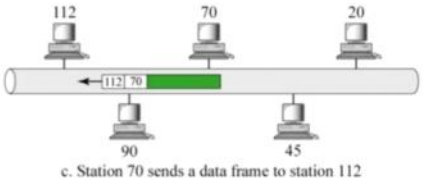
token based (3)



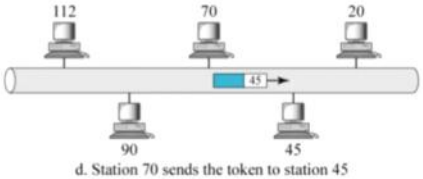
a. Station 112 does not have data; it sends the token to 90



b. Station 90 does not have data; it sends the token to 70



c. Station 70 sends a data frame to station 112



d. Station 70 sends the token to station 45

token based (4)

- under light load – delay is added due to waiting for the token
- under heavy load – ring is “round robin”

advantage:

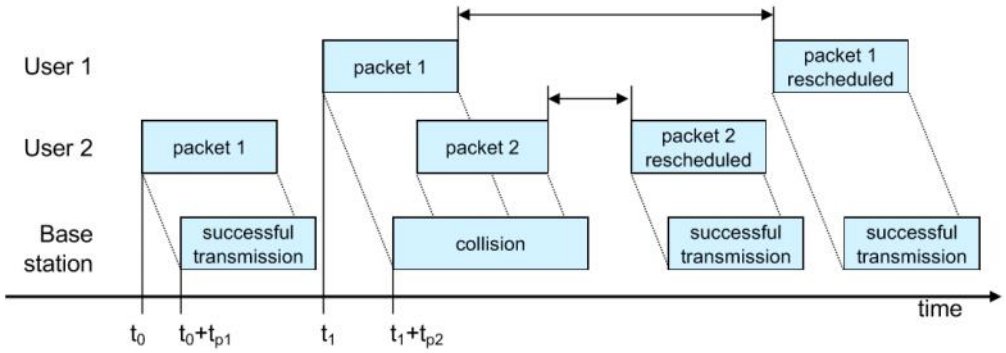
a) fair access

disadvantages:

a) ring is sensitive to points of failure

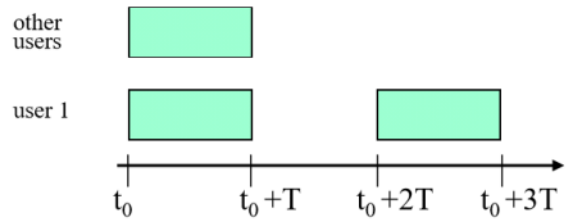
b) added issues due to token maintenance

ALOHA



- transmit messages immediately – use entire bandwidth
- in case of collision – delay retransmission for a **random** time interval
- special case of ”stop-and-wait” ARQ (Automatic Repeat Request)

slotted ALOHA



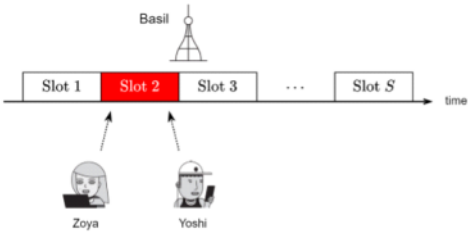
vulnerable period: T

throughput

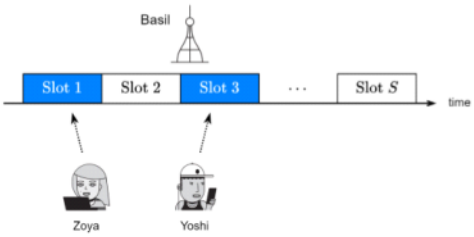
$$S = \lambda T \exp(-\lambda T) = G \exp(-G)$$

framed ALOHA (1)

- 2 users sends packets to Basil
  - S data slots are available
  - Zoya and Yoshi pick a single slot with probability $\frac{1}{S}$

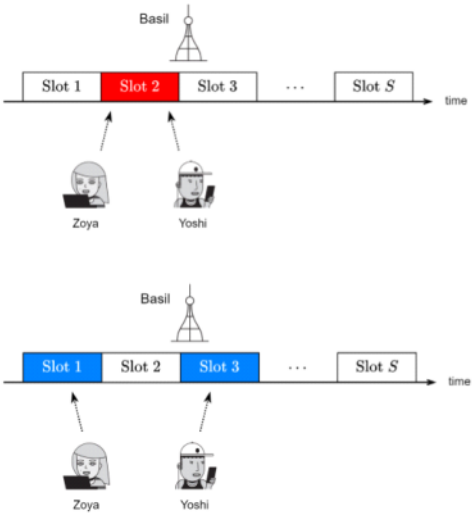


- probability of successful transmission in a packet
  - $P(S) = \frac{1}{S}\left(1 - \frac{1}{S}\right) + \left(1 - \frac{1}{S}\right)\frac{1}{S} = \frac{2}{S}\left(1 - \frac{1}{S}\right)$
  - optimal number of slots: $S^* = 2$
  - $P(S = 2) = \frac{1}{2}$



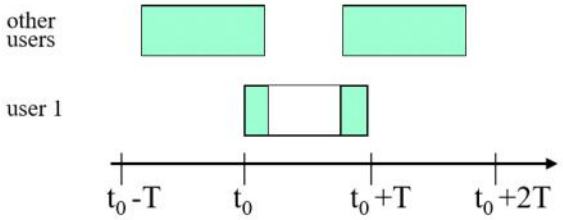
framed ALOHA (2)

- generalization for K users
  - $P(S) = \frac{K}{S}\left(1 - \frac{1}{S}\right)^{K-1}$
  - optimal number of slots: $S^* = K$
  - $P(S = K) = \left(1 - \frac{1}{K}\right)^{K-1}$
- lower bound for the probability
  - $\lim_{K \to \infty} \left(1 - \frac{1}{K}\right)^{K-1} = e^{-1}$

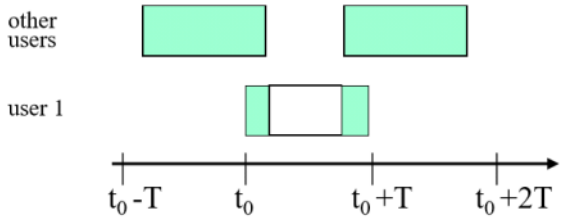


ALOHA analysis (1)

- packet arrival process is modelled as a Poisson process with average arrival rate λ
- λ - arrival rate of new and retransmitted packets



ALOHA analysis (2)



$$\Pr[n] = \frac{(d\lambda)^n}{n!}\exp(-d\lambda)$$

- vulnerable period: *(t-T, t+T)*
- probability that no packet starts *T* seconds before and *T* seconds after the start time of a given packet

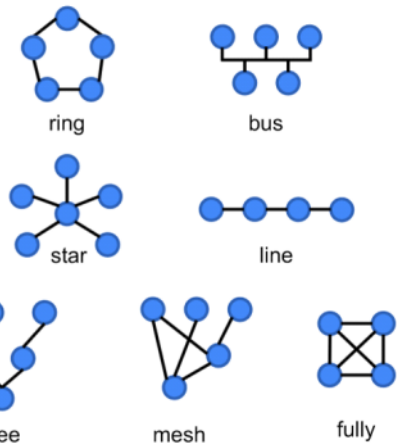
$$\Pr[success] = \Pr[0] = \exp(-2T\lambda)$$

- the throughput is

$$S = \lambda T \exp(-2\lambda T) = G\exp(-2G)$$

network topologies: some classifications

- how to connect the network nodes?
- why is this important?
  - throughput and reliability
- common layouts:
  - bus, star, ring, mesh, fully, tree



overlay network:
virtual network built on top of another network

internet is a network of networks

N) security

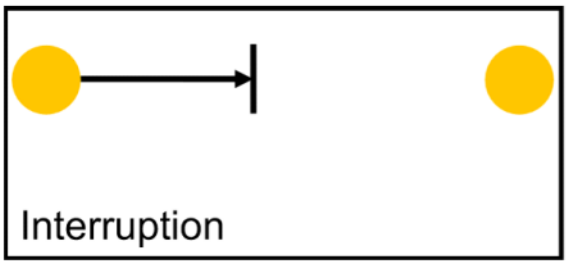
Friday, 3 January 2025 13.06

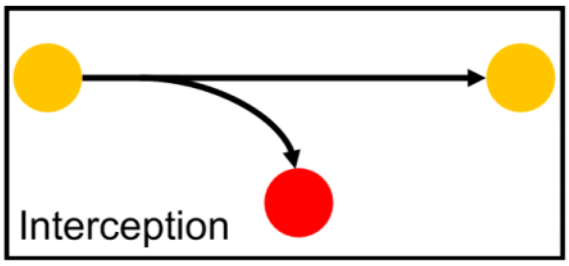
SEE LECTURE 9

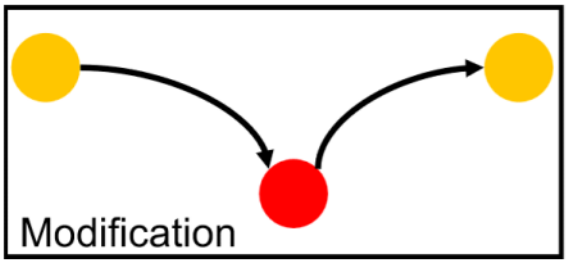
security: main requirements

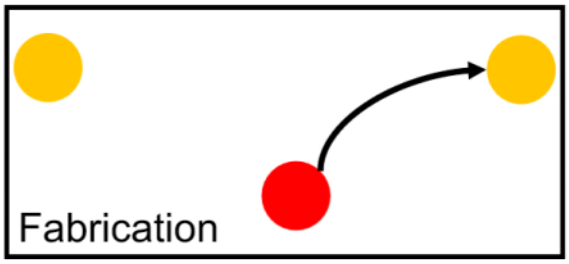
- authentication: the communication parties want to be sure that the other party is indeed the one claimed.
- confidentiality: only sender and receiver shall be able to read the transferred data.
- privacy/anonymity: personal information (including own identity) should not be revealed.
- integrity: assurance that data has not been changed on the way from the sender to the receiver.
- availability: network and services shall be available whenever needed.
- non-repudiation: a user cannot deny having used a certain service.
- legal requirements: country specific legal security requirements (e.g. Legal interception, etc.)
- double spending: ensure that digital money is spent only once (the main invention in Bitcoin).

security attacks









passive vs active attacks

- passive attacks
  - difficult to detect -> prevention, not detection
  - release of message contents
  - traffic analysis

- active attacks
  - masquerade
  - replay
  - modification of messages
  - denial of service

determining severity of the risk

Likelihood and Impact Levels	
0 to <3	LOW
3 to <6	MEDIUM
6 to 9	HIGH

Overall Risk Severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
		Likelihood		

terminology

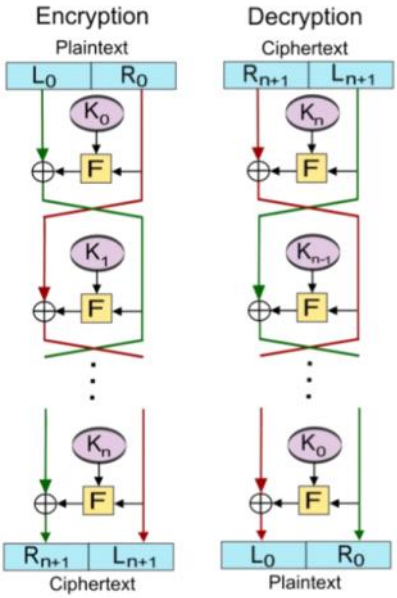
- plaintext - original message
- ciphertext - coded message
- cipher - algorithm for transforming plaintext to ciphertext
- key – secret information used in cipher known only to sender/receiver
- encipher (encrypt) - converting plaintext to ciphertext
- decipher (decrypt) - recovering ciphertext from plaintext
- cryptography - study of encryption principles/methods
- cryptanalysis (code breaking) –
  study of principles/methods of deciphering ciphertext without knowing key

attack type	what is known by the cryptanalyst
Ciphertext only	•Encryption algorithm •Ciphertext to be decoded
Known plaintext	•Encryption algorithm •Ciphertext to be decoded •One or more plaintext-ciphertext pairs formed with the secret key
Chosen plaintext	•Encryption algorithm •Ciphertext to be decoded •Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key
Chosen ciphertext	•Encryption algorithm •Ciphertext to be decoded •Purported ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key
Chosen text	•Encryption algorithm •Ciphertext to be decoded •Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key •Purported ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key

- computationally secure scheme
  - the cost of breaking the cipher exceeds the value of the encrypted information
  - the time required to break the cipher exceeds the useful lifetime of the information

Feistel cipher structure

- all conventional block encryption algorithms have this structure
  - not a specific block cipher, but rather a blueprint
- parameters for a concrete realization:
  - block size, e.g. 64 bits
  - key size, e.g. 128 bits
  - number of rounds, e.g. 16
  - subkey generation algorithm
  - round (inner) function F



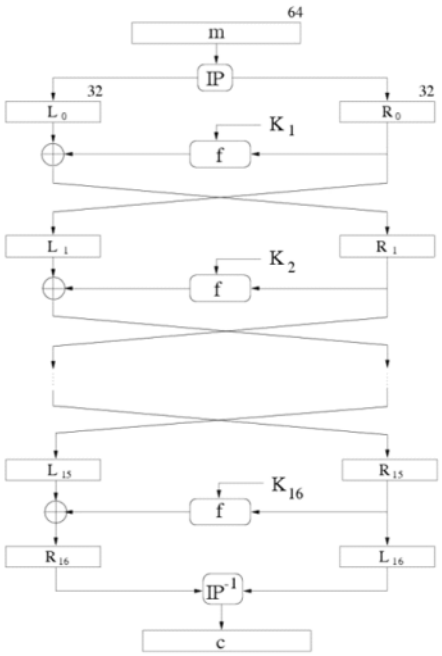
symmetric block encryption algorithms

- Data Encryption Standard (DES)
- Triple DES (3DES)
- Advanced Encryption Standard (AES)
- Blowfish
- RC5

data encryption standard (DES)

- DES is a Feistel cipher with…
  - 64 bit block length
  - 56 bit key length
  - 16 rounds
  - 48 bits of key used each round (subkey)

- Advanced Encryption Standard (AES)
  - block size: 128 bits
  - key length: 128, 192 or 256 bits
  - 10 to 14 rounds (depends on key length)



Electronic Code Book (ECB)

- notation: C = E(P, K)
- given plaintext P0, P1, …, Pm, …
- most obvious way to use a block cipher:

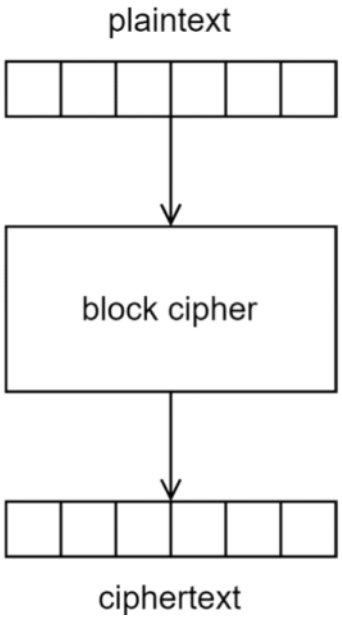
Encrypt	Decrypt
C0 = E(P0, K)	P0 = D(C0, K)
C1 = E(P1, K)	P1 = D(C1, K)
C2 = E(P2, K) …	P2 = D(C2, K) …

the trouble with ECB

- if the same b-bit block of plaintext appears more than once in the message, it always produces the same ciphertext

- due to this, for lengthy messages, the ECB mode may not be secure

- if the message is highly structured, it may be possible for a cryptanalyst to exploit these regularities



cipher block chaining (CBC)

- blocks are “chained” together
- a random initialization vector (IV), is required to initialize CBC mode
- IV is random, but not secret
- trouble: if a block/packet is lost, then all subsequent cannot be decrypted
  - recall our flow control and transport protocols from the last time

Encryption	Decryption
$C0 = E(IV \oplus P0, K)$,	$P0 = IV \oplus D(C0, K)$,
$C1 = E(C0 \oplus P1, K)$,	$P1 = C0 \oplus D(C1, K)$,
$C2 = E(C1 \oplus P2, K),\ldots$	$P2 = C1 \oplus D(C2, K),\ldots$

public-key cryptography

- two keys, one to encrypt, another to decrypt
  - Alice uses Bob’s public key to encrypt
  - Only Bob’s private key decrypts the message
- based on “trap door, one way function”
  - “One way” means easy to compute in one direction, but hard to compute in other direction
  - Example: Given p and q, product N = pq easy to compute, but hard to find p and q from N
  - “Trap door” is used when creating key pairs
- encryption
  - Suppose we encrypt M with Bob’s public key
  - Bob’s private key can decrypt C to recover M
- digital signature

public-key cryptography

- each party has a pair of keys: K1 is the public key and K2 is the secret key, such that $D_{K2}(E_{K1}(M))=M$
- knowing the public-key and the cipher, it is computationally infeasible to compute the private key
  - thereby **asymmetric** crypto system
- the public-key K1 may be made publicly available
  - many can encrypt, only one can decrypt
- two parties who do not share any private information through communications arrive at some secret not known to any eavesdroppers
  - use it to share a secret key

RSA (Rivest, Shamir and Adleman 1978)

- based on difficulty of determining prime factors of large numbers
- approach
  - select secret primes p,q (>100 decimal digits)
  - communicate N=pq, the modulus
  - choose e relatively prime to (p−1)(q−1)
  - find d such that ed = 1 mod (p−1)(q−1)
  - public key is (N,e)
  - private key is d
- encryption and decryption
  - Encryption: $c = m^e \bmod N$
  - Decryption: $m = c^d \bmod N$

cryptographic protocols: key agreement

- Task: Agreement on joint, secret session key k
- Diffie-Hellman key exchange
  - invented by Williamson (GCHQ) and, independently, by D and H (Stanford)
  - a “key exchange” algorithm, used to establish a shared symmetric key
  - based on discrete log problem
    - **given:** g, p, and $g^k \bmod p$**, find:** exponent k
    - this problem is hard
  - Steps
    1. A and B agree on prime number p and integer g (can be publicly known)
    2. A selects secret a, B selects secret b
    3. A computes $\alpha = g^a \bmod p$ and sends α to B,
       B computes $\beta = g^b \bmod p$ and sends β to A
    4. A computes $k = \beta^a \bmod p$, B computes $k = \alpha^b \bmod p$
    5. A and B can communicate with secret session key k

examples of security threats

- eavesdropping messages
- modifying messages on their path from sender to receiver
- using somebody else's identity
- manipulate charging
  - use services without payment or with payment from third person's account
  - 'overcharge' third persons account (without use of services)
- block certain functionality (Denial of Service Attacks)

- possible origin/point of attack
  - via external Interfaces: e.g., connection to Internet
  - while passing through un-trusted intermediate networks (e.g. backbone connecting site networks)
  - air interface/wireless links
  - malicious processes/users within the distributed system
    - viruses, worms, etc.
  - distributed attacks, e.g. via botnets
  - network management/administration

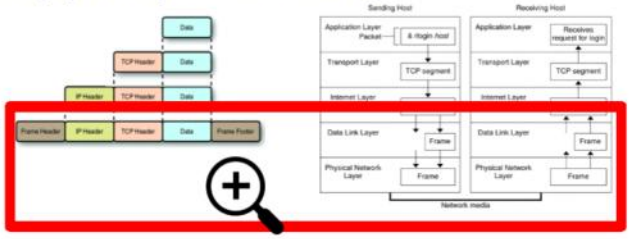
M) coding

Saturday, 4 January 2025 09.00

today

- focus on the data-link and physical layer communication

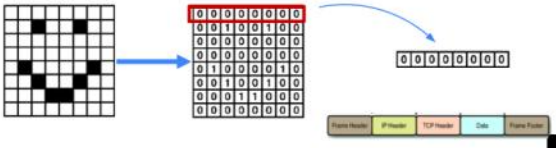
we abstract this away, and consider it as our packet



- better define the strategies to provide a reliable communication
  - we have seen a bit of error detection and error correction in the first lecture
  - our goal is to recap and go a little bit further

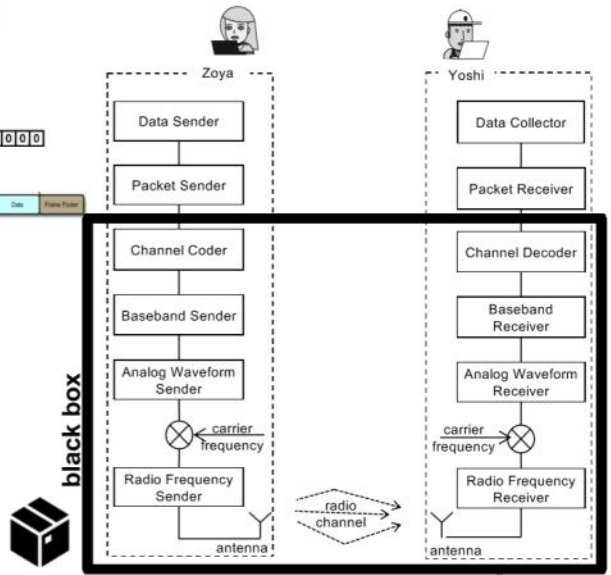
Petar Popovski, Communication in Electronic Systems, Fall 2024. AALBORG UNIVERSITY 4

reliable communication



main question for today:

how do we guarantee a reliable communication?

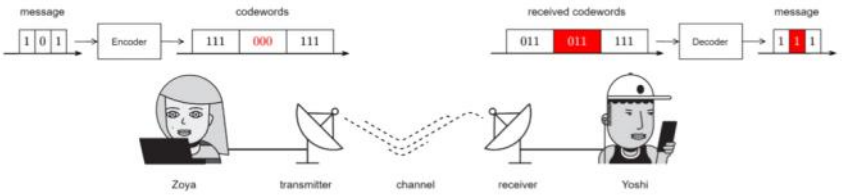


Petar Popovski, Communication in Electronic Systems, Fall 2024. AALBORG UNIVERSITY 5

definition of a code rate

- we use the binary channel n times to send $b < n$ bits
- the code rate is then $R = \frac{b}{n}$ [bits/channel use]
- number of redundant bits is $n - b$
- both error detection and error correction require redundancy
- in general, we can send M possible messages by using the channel n times
  - here $M < 2^n$
  - code rate $R = \frac{\log_2 M}{n}$

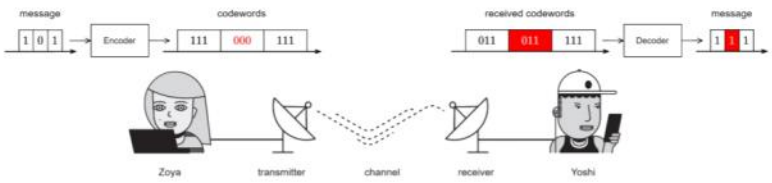
repetition code and majority voting (1)



- encoder
  - bit $0 \rightarrow$ codeword 000
  - bit $1 \rightarrow$ codeword 111
- decoder
  - majority voting
  - most frequent bit in the received codeword
  - lookup table
- bit error rate analysis
  - BER uncoded communication (BSC): p_U
  - tx codeword: 111
  - prob. rx 000: p_U^3
  - prob. rx 001: $p_U^2(1-p_U)$
  - prob. rx 010: $p_U^2(1-p_U)$
  - prob. rx 100: $p_U^2(1-p_U)$
  - BER coded communication: $p_C = 3p_U^2(1-p_U) + p_U^3 < p_U$

Received codeword	Decoded bit
000	0
001	0
010	0
011	1
100	0
101	1
110	1
111	1

repetition code and majority voting (2)



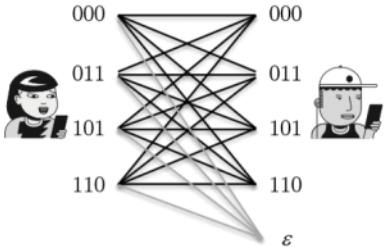
- throughput analysis
  - b information bits + c check bits
- uncoded communication
  - $T_U = \frac{b}{b+c}(1-p_U)^{b+c}$
- coded communication
  - $T_C = \frac{b}{3(b+c)}(1-p_C)^{b+c}$
  - $T_C = T_U \frac{1}{3}\left(\frac{1-p_C}{1-p_U}\right)^{b+c}$
- throughput degrades significantly when the BSC BER is low

BSC BER (p_U)	Number of bits ($b+c$)	Throughput
0.495	100	$T_C = 0.549T_U$
0.495	200	$T_C = 0.895T_U$
0.495	300	$T_C = 1.47T_U$
10^{-6}	100	$T_C = 0.333T_U$

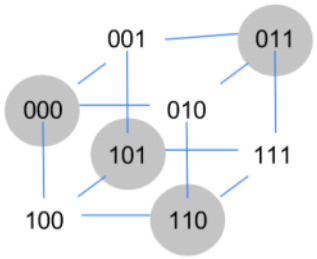
forward error correction (FEC): a simple code

- block code: b bits supplied as a **message** $\mathbf{d} = (d_1, \ldots, d_b)$
- each **message** is associated a **codeword** $\mathbf{x} = (x_1, \ldots, x_n)$
- $n > b$ and both d_i and x_j assumed binary
- we call this an $(\boldsymbol{n}, \boldsymbol{b})$**-code** with rate $R = \frac{b}{n}$ [bit/c.u.] and corresponds to c-channel with 2^b inputs and 2^n outputs
- Hamming distance of 2

message $\mathbf{d}$	codeword $\mathbf{x}$
00	000
01	011
10	101
11	110



(d)



Hvert bit skal endres 2 ganger for at vi "forkert"

FEC: linear block code

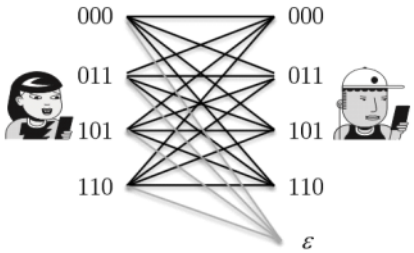
the code provided as example belongs to the class of *linear block codes*, where linearity wrt. addition and multiplication of binary numbers, i.e., within the *GF*(2)

encoding has particularly simple interpretation

$$\mathbf{x} = \mathbf{d} \cdot \mathbf{G}$$

where $\boldsymbol{G} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}$ is the Generator matrix
the additional bits are called **parity bits**

- in a good code **one bit** should influence **several symbols**, and **one symbol** should be influenced by **several bits**



(d)

message $\mathbf{d}$	codeword $\mathbf{x}$
00	000
01	011
10	101
11	110

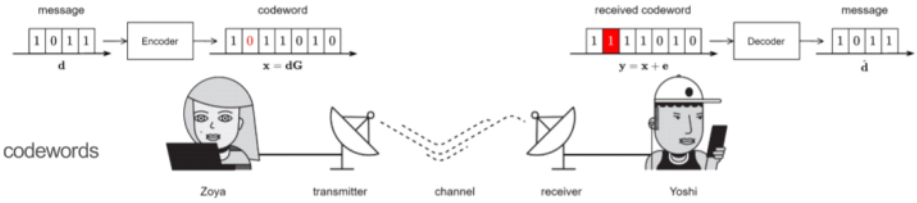
Hamming code (1)

- linear block code
- detect 1- and 2-bit errors
- correct 1-bit errors

- (l, b) code
  - b-bits blocks mapped into l-bits codewords
  - code rate: $R = \frac{b}{l}$ bit/c.u.

- encoder
  - $d \in \{0,1\}^{1\times b}$: message
  - $x \in \{0,1\}^{1\times l}$: codeword
  - tx codeword: $\boldsymbol{x} = \boldsymbol{dG}$ (binary addition!)
- receiver
  - rx codeword: $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{e}$
  - syndrome vector: $\boldsymbol{s} = \boldsymbol{yH}^T$
  - $s = 0 \Rightarrow$ no error in the received codeword
  - $s \neq 0 \Rightarrow$ error in the received codeword

- (7,4) Hamming code
  - higher rate than the repetition code $\left(\frac{4}{7} > \frac{1}{3}\right)$

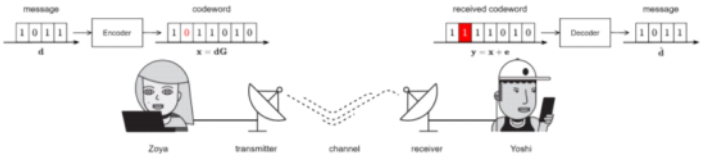


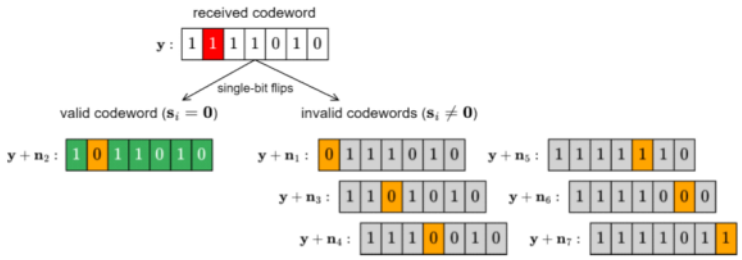
generator and parity-check matrices:

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

Hamming code (2)

- syndrome decoding
  - compute syndrome vector for each bit flip
  - $s_i = (y + n_i)H^T$
  - the message is the valid codeword
  - no valid codeword ⇒ 2-bit error detected
  - the message is in the 4 initial bits of the corrected codeword
- codewords
  - in the Hamming code the minimum distance between valid codewords is always 3
  - this is why we can detect up to 2-bit errors and correct 1-bit errors
- other Hamming codes
  - m parity bits, with $m \geq 3$
  - we can create $(m, 2^m - 1)$ codes
  - if we choose a large m, the code rate approaches 1
  - rate improves with large m, but we cannot correct more erroneous bits





digital modulation

an **analog carrier** is modulated by a **discrete signal**

advantages vs. analog modulation:

- higher data rate (bits per second) given a fixed bandwidth
- more robust to channel impairments
  - advanced coding and decoding to combat fading and noise
  - spread spectrum techniques to deal with multipath and to resist interference
- allows for multiple access
  - signals from multiple users in the same bandwidth can be decoded simultaneously
- security and privacy: encryption

going from bits to symbols

three steps

Baseband

1. map bits to complex-valued baseband symbols (2 dimensions)

example: with two complex values, the symbols can be $1-j$ to represent 0 and $1+j$ for 1

The system transmits one symbol each T seconds: symbol period

$$0110101\ldots \longrightarrow X_0, X_1, X_2, \ldots$$

2. assign a pulse waveform (pulse shape) to each symbol

$$X_0, X_1, X_2, \ldots \longrightarrow \sum_k X_k p(t-kT)$$

3. modulate to a high frequency carrier

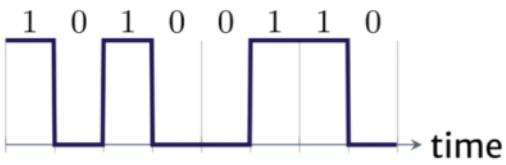
higher data rate?

a complex symbol can carry more than one bit

example:

1 bit per symbol: $0 \to -A$ and $1 \to A$

vector $[1,0,1,0,0,1,1,0] \to [A,-A,A,-A,-A,A,A,-A]$



2 bits per symbol

$$00 \to A$$
$$01 \to Aj$$
$$10 \to -A$$
$$11 \to -Aj$$

pulse shape examples

sampling function

$$p(t) = \frac{1}{\sqrt{T}}\text{sinc}\left(\frac{\pi t}{T}\right) = \begin{cases} \frac{1}{\sqrt{T}}, & \text{for } t = 0 \\ \frac{1}{\sqrt{T}}\frac{T\sin\left(\frac{\pi t}{T}\right)}{\pi t}, & \text{otherwise} \end{cases}$$

rectangular function

$$p(t) = \begin{cases} \frac{1}{\sqrt{T}}, & \text{for } t \in (0, T] \\ 0, & \text{otherwise} \end{cases}$$

IQ representation of a signal

a bandpass signal can be represented as

$$s(t) = s_I(t)\cos(2\pi f_c t) + j s_Q(t)\sin(2\pi f_c t)$$
$$= I(t) + jQ(t)$$

I : in-phase component $s_I(t)$ **Q : quadrature component** $s_Q(t)$

canonical form of a bandpass signal

the carrier is normally thought as the cosine term, so, the I term is “in-phase” with the carrier

amplitude: $A(t) = \sqrt{s_I^2(t) + s_Q^2(t)}$ Phase: $\phi(t) = \tan^{-1}\left(\frac{s_Q(t)}{s_I(t)}\right)$

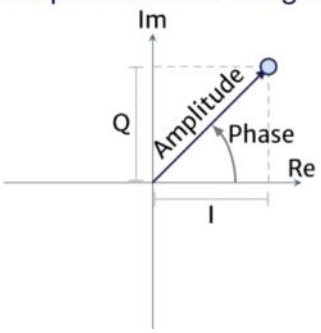
constellation diagram

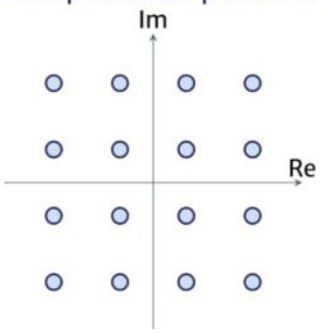
representation of a digitally modulated signal

uses a collection of points to represent all the different symbols that can be transmitted

the **angle** represents the shift of the carrier wave from a reference phase

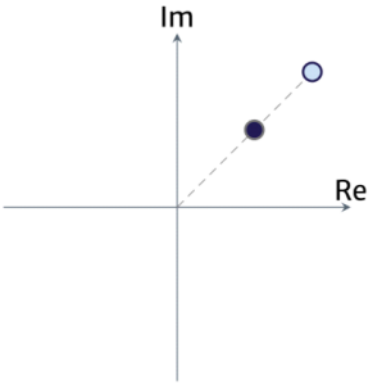
the **distance** of a point from the origin represents the amplitude or power of the signal



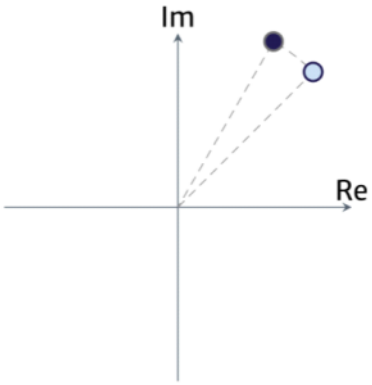


effects of the channel on the symbols

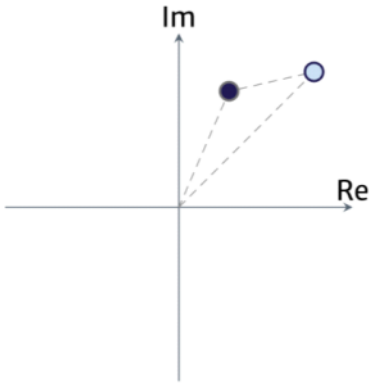
amplitude change



phase change



amplitude and phase change



bit rate and symbol rate

in a constellation with M symbols, each symbol represents $N = \log_2 M$ bits

symbol period: T_{sym}

a new symbol is represented once every T_{sym} seconds by shifting the pulse

symbol rate: $R_{sym} = \dfrac{1}{T_{sym}}$ symbols/second (bauds)

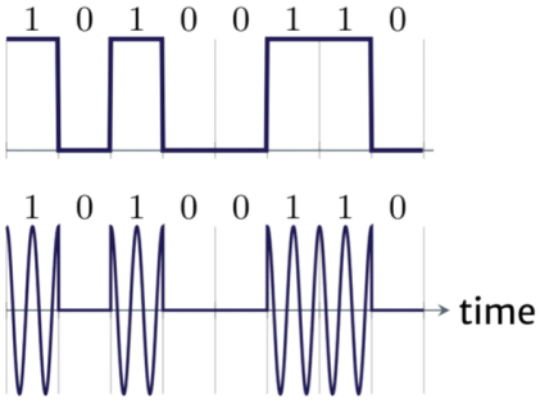
the bit rate is $R_b = NR_{sym} = R_{sym} \log_2 M$ bits per second (bps)

amplitude-shift keying (ASK)

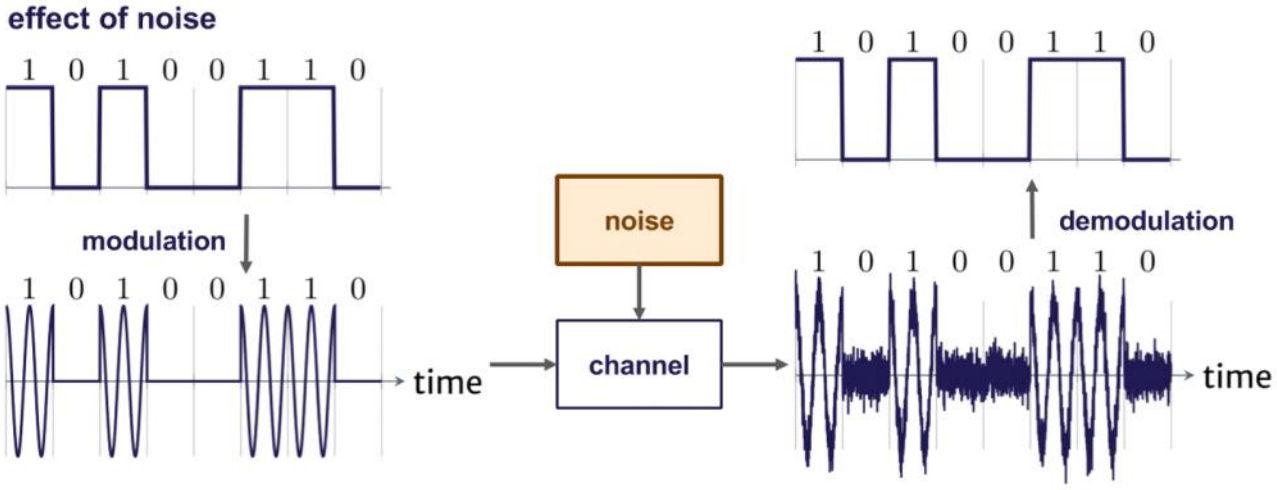
bit stream is encoded with the amplitude of the transmitted signal

simplest form: on-off keying (OOK)

$$\begin{cases} s_0(t) = 0 \\ s_1(t) = A\cos(2\pi f_c t) \end{cases}$$



on-off keying (OOK)

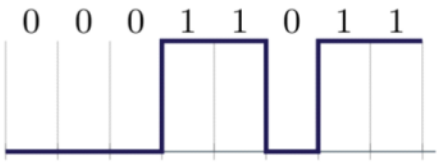


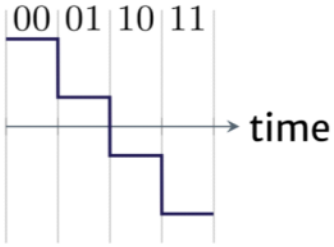
M-ary amplitude-shift keying (M-ASK)

there is an amplitude for each symbol (bit pattern)

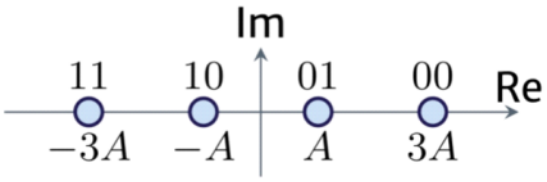
$m \in \{1, 2, \ldots, M\}$

$s_m(t) = A_m \cos(2\pi f_c t) \quad t \in [0, T_{sym})$





example: 4-ASK

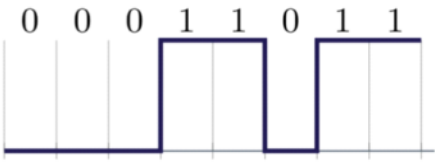


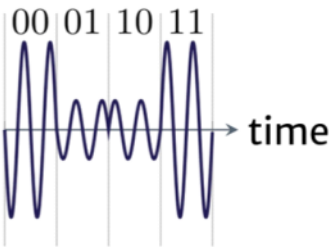
M-ary amplitude-shift keying (M-ASK)

there is an amplitude for each symbol (bit pattern)

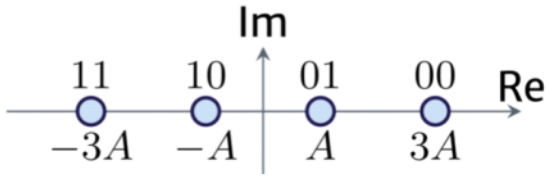
$m \in \{1, 2, \ldots, M\}$

$s_m(t) = A_m \cos(2\pi f_c t) \quad t \in [0, T_{sym})$





example: 4-ASK



phase-shift keying (PSK)

bit streams are encoded in the phase of the transmitted signal

simplest: Binary Phase-Shift Keying (BPSK)

$$\begin{cases} s_0(t) = A\cos(2\pi f_c t + 0) \\ s_1(t) = A\cos(2\pi f_c t + \pi) \end{cases} \qquad f_c = \frac{n_c}{T_{sym}}$$

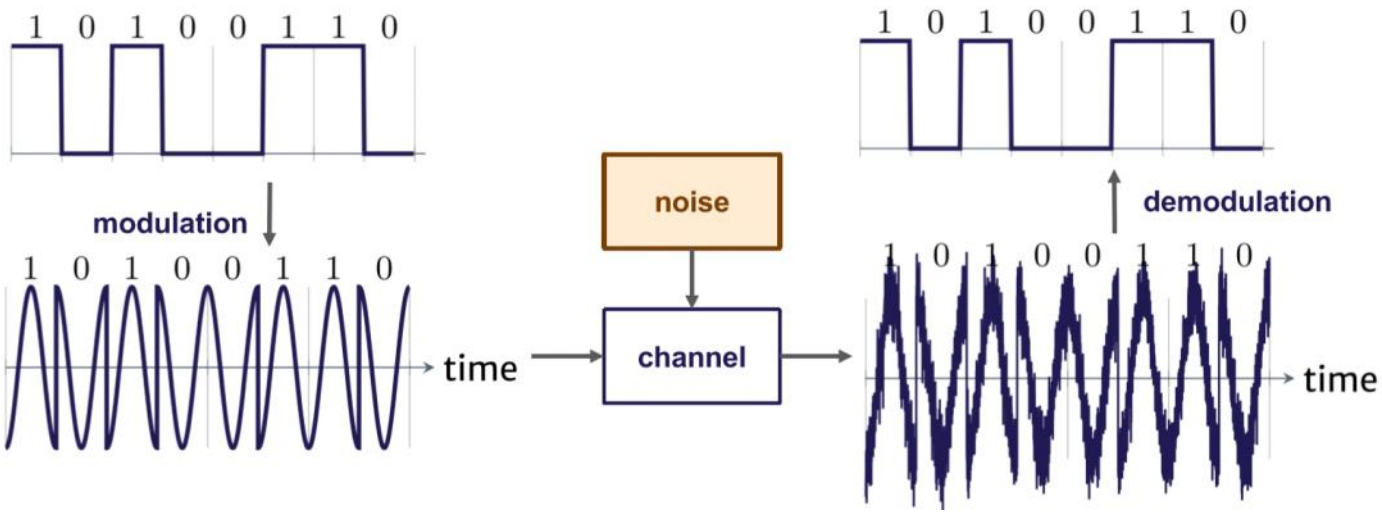
bandwidth $\mathrm{BW} \approx 2R_{sym}$ Hz

modulated signal with rectangular pulse: bandwidth approximated by

m-ary Phase-Shift Keying (M-PSK)

$$s_m(t) = A\cos(2\pi f_c t + \theta_m) \qquad \theta_m = \frac{2\pi(m-1)}{M} \qquad m \in \{1, 2, \ldots, M\} \qquad t \in [0, T_{sym})$$

binary phase-shift keying (BPSK)

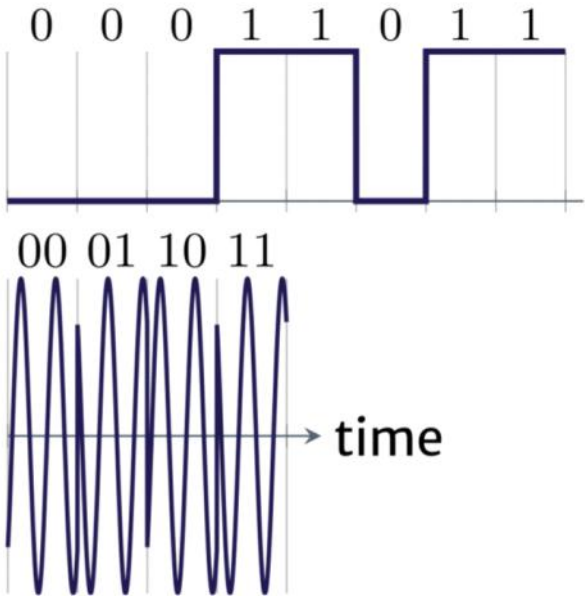


quaternary phase-shift keying (QPSK)

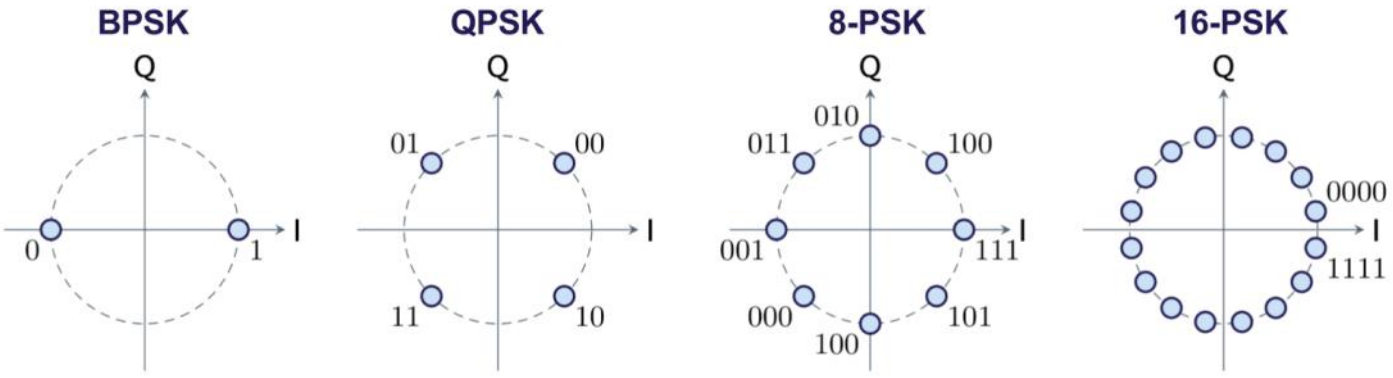
phases

$$\theta_m = \frac{2\pi(m-1)}{M} + \frac{\pi}{4}$$

like 4-PSK but rotated $\frac{\pi}{4}$



M-PSK



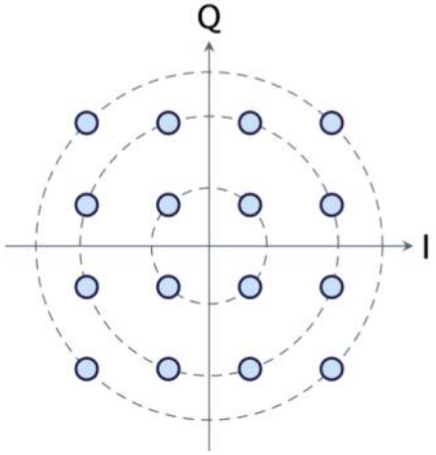
Gray coding

quadrature amplitude modulation (QAM)

QAM has two carriers, each having the same frequency but differing in phase by 90 degrees

the information symbol modulates both the amplitude and phase of the carrier

combination of PSK and ASK

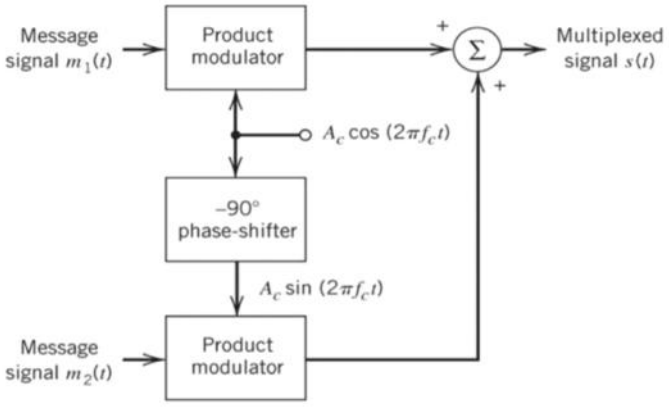


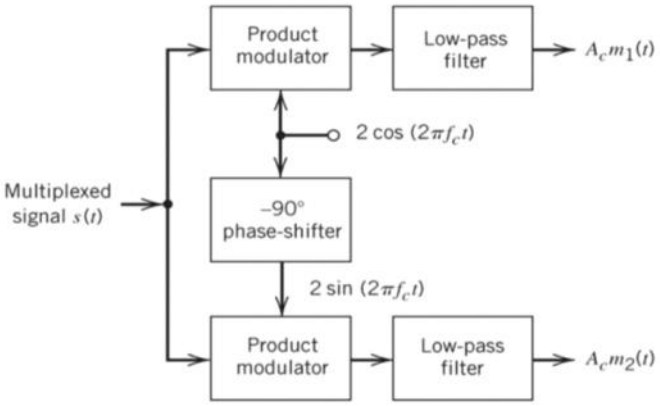
receiver:
the two waves can be coherently separated (demodulated) because of their orthogonality property

widely used

- IEEE 802.11
- optical fiber
- cellular systems

QAM modulation and demodulation





frequency-shift keying (FSK)

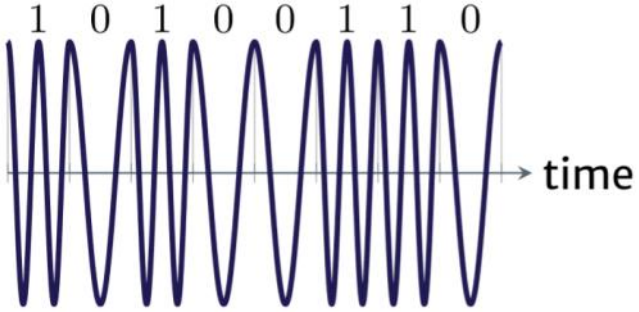
the digital message is carried in the discrete frequency changes of the carrier

simplest: Binary FSK (BFSK)

two frequencies:

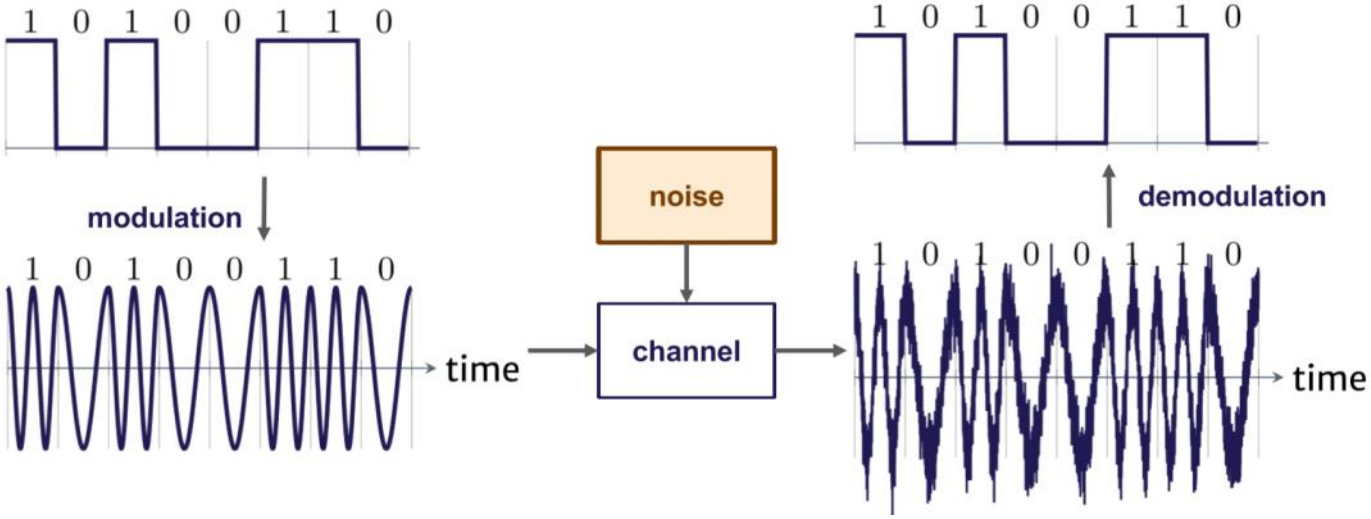
- 1: “mark frequency”
- 0: “space frequency”

$$\begin{cases} s_0(t) = A\cos(2\pi f_{space}\, t) \\ s_1(t) = A\cos(2\pi f_{mark}\, t) \end{cases}$$



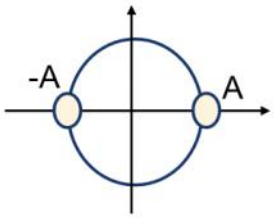
Gaussian FSK (GFSK)

binary frequency-shift keying (BPSK)



error probability for BPSK

- Bit 1: $s_1(t) = Ag(t)\cos(2\pi f_c t)$
- Bit 0: $s_2(t) = -Ag(t)\cos(2\pi f_c t)$
- Minimum distance: $A - (-A) = 2A$
- Energy per bit:



$$E_b = \int_0^{T_b} s_1(t)^2\,dt = \int_0^{T_b} s_2(t)^2\,dt = \int_0^{T_b} A^2 g(t)^2 cos^2(2\pi f_c t)\,dt = A^2$$

- Bit Error Probability

$$P_b = Q\left(\frac{d_{min}}{\sqrt{2N_0}}\right) = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) = Q\left(\sqrt{2\gamma_b}\right)$$

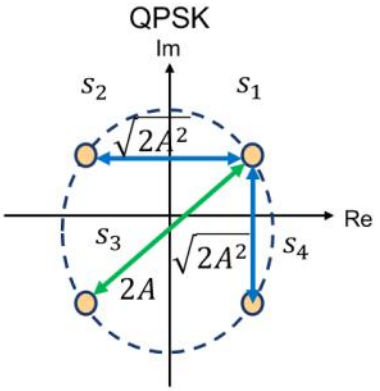
error probability of QPSK

- union bound of error probability

$$P_e \le 2Q\left(\frac{A}{\sqrt{N_0}}\right) + Q\left(\frac{A\sqrt{2}}{N_0}\right)$$

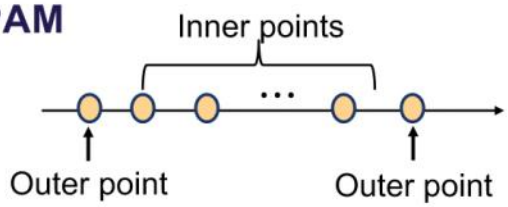
- $\gamma_s = 2\gamma_b = 2\frac{A^2}{N_0}$

$$P_e \le 2Q\left(\sqrt{\frac{\gamma_s}{2}}\right) + Q\left(\sqrt{\gamma_s}\right)$$



error Probability for ASK/PAM

- two types of symbols
  - $M-2$ inner points
  - 2 Outer points



Error probability for inner point

$$p_{e,i} = Pr\left[|n| > \frac{d_{min}}{2}\right] = 2Q\left(\frac{d_{min}}{\sqrt{2N_0}}\right)$$

Error probability for outer point

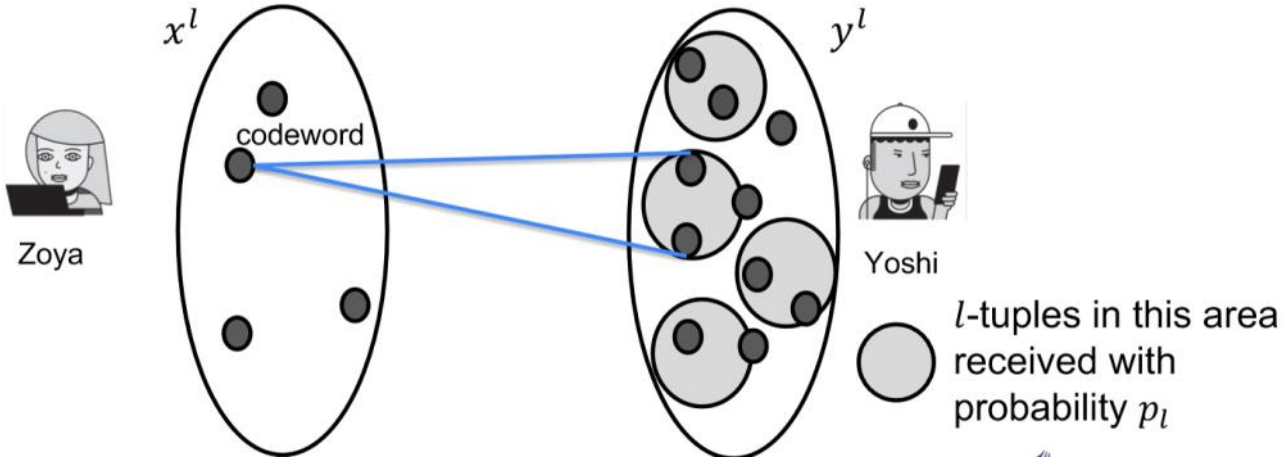
$$p_{e,o} = \frac{1}{2}p_{e,i} = 2Q\left(\frac{d_{min}}{\sqrt{2N_0}}\right)$$

- Expected error probability

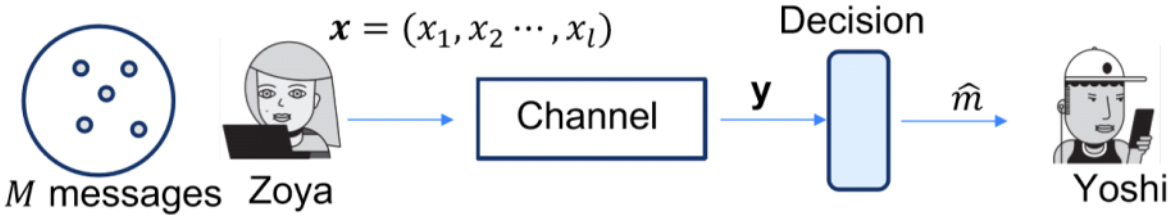
$$P_e = \frac{1}{M}\left[(M-2)p_{e,i} + 2p_{e,o}\right] = \frac{2(M-1)}{M}\,Q\left(\frac{d_{min}}{\sqrt{2N_0}}\right)$$

Generalization of coding idea

- How to select as many as possible inputs for the expanded channel
- Guarantee high reliability for the transmission of a single input in the expanded channel.



Maximum likelihood detector



Likelihood

$$\Pr(\boldsymbol{x_i}|\boldsymbol{y}) = \frac{\Pr(\boldsymbol{y}|\boldsymbol{x_i})P(\boldsymbol{x_i})}{p(\boldsymbol{y})}$$

e.g., memoryless channel

$$\Pr(\boldsymbol{y}|\boldsymbol{x_i}) = \prod_{j=1}^{l} \Pr(y_j|x_{i\,j})$$

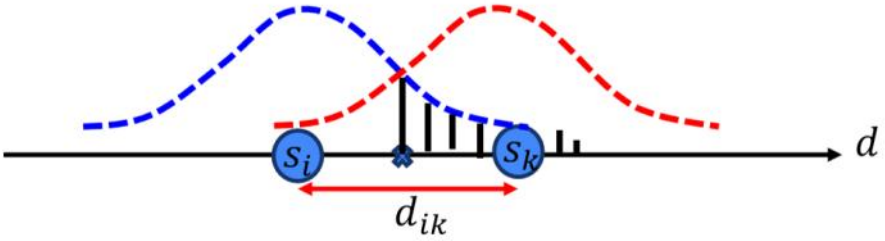
Maximum log likelihood:

$$\hat{m} = \underset{i \in \{1,2,\cdots,M\}}{\operatorname{argmax}} \log \Pr(\boldsymbol{y}|\boldsymbol{x}_i)$$

basics of error probability

$$Q(z) = \int_z^{\infty} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) dx$$

- $y = x + n$
  - ***x:*** input symbol
  - ***y:*** **received signal**
  - ***n:*** zero mean Gaussian random variable with variance σ^2

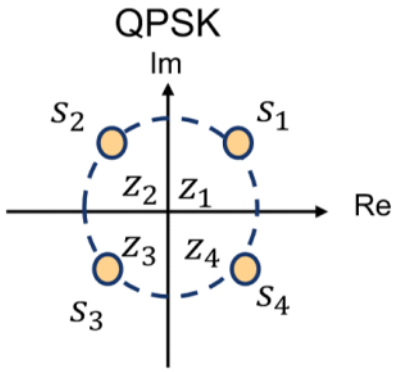


$$\Pr(A_{ik}) = \Pr\left(n > \frac{d_{ik}}{2}\right) = \int_{\frac{d_{ik}}{2}}^{\infty} \frac{1}{\sqrt{\pi N_0}} \exp\left(-\frac{v^2}{N_0}\right) dv = Q\left(\frac{d_{ik}}{\sqrt{2N_0}}\right)$$

decision regions and error probability

- decision region

$$z_i = (x: \|x - s_i\| < \|x - s_j\|, j = 1, \cdots M, j \neq i)$$



- probability of symbol error

$$P_e = \sum_{i=1}^{M} \Pr(\boldsymbol{x} \neq \boldsymbol{Z_i} | \boldsymbol{m}_i \, sent) \Pr(\boldsymbol{m}_i \, sent)$$

lower bound of error analysis

- union Bound of probability

$$P_e(m_i) = Pr\left(\bigcup_{k=1,k\neq i}^{M} A_{i,k}\right) \leq \sum_{k=1,k\neq i}^{M} \Pr(A_{i,k}) = \sum_{k=1,k\neq i}^{M} Q\left(\frac{d_{i,k}}{\sqrt{2N_0}}\right)$$

- expectation of error probability

$$P_e = \sum_{i=1}^{M} \Pr(m_i) P_e(m_i) \leq \frac{1}{M} \sum_{i=1}^{M} \sum_{k=1,k\neq i}^{M} Q\left(\frac{d_{i,k}}{\sqrt{2N_0}}\right)$$

- looser bound
  - $d_{min} = \min_{i,k} d_{ik}$

$$P_e \leq (M-1) Q\left(\frac{d_{min}}{\sqrt{2N_0}}\right)$$

M) mod2

Saturday, 4 January 2025 14.31

recap (2): baseband signals

three steps

1. map bits to complex-valued baseband symbols (2 dimensions) **Baseband**

example: with two complex values, the symbols can be $1-j$ to represent 0 and $1+j$ for 1
The system transmits one symbol each T seconds: symbol period

$$0110101\ldots \longrightarrow X_0, X_1, X_2, \ldots$$

2. assign a pulse waveform (pulse shape) to each symbol

$$X_0, X_1, X_2, \ldots \longrightarrow \sum_k X_k p(t-kT)$$

3. modulate to a high frequency carrier

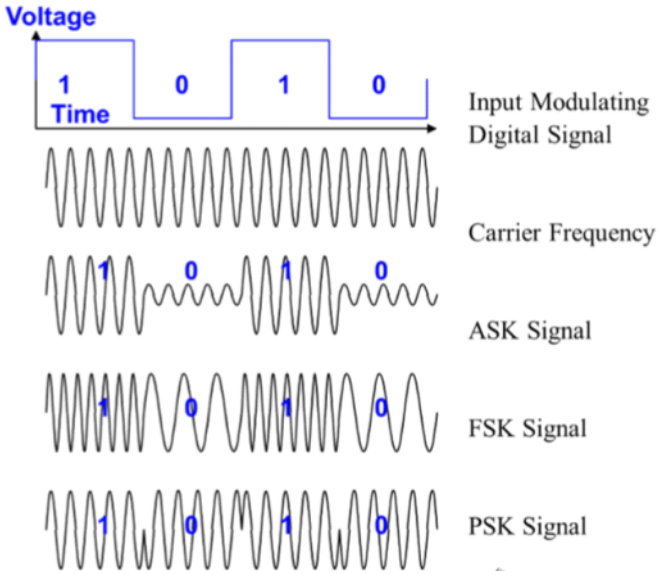
recap (3): digital modulations

amplitude-shift keying (ASK)

phase-shift keying (PSK)

frequency-shift keying (FSK)

quadrature amplitude modulation (QAM)



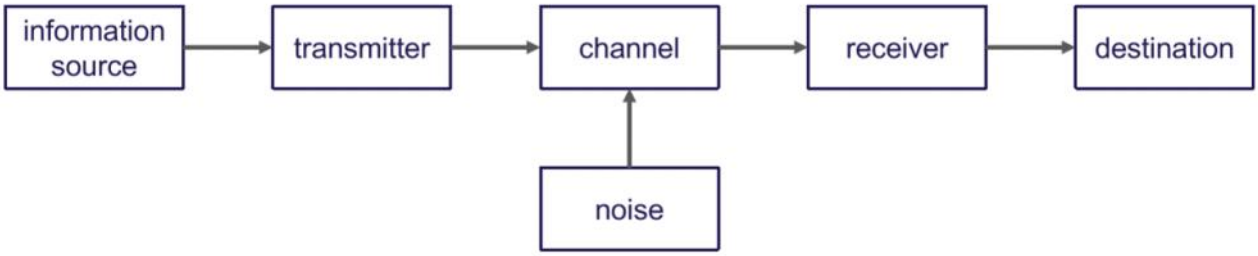
it's all about the information

recall: the purpose of a communication system is **to convey information**

this information must be **converted into** an appropriate format for the transmission medium

signal: carries information

energy must be spent to create a structured signal



from time to frequency and *vice versa*

time domain: how the properties change over time

frequency domain: how the properties change over frequency

Fourier transform: from time to frequency domain

$$W(f) = \mathcal{F}\left[w(t)\right] = \int_{-\infty}^{\infty} w(t) e^{-j2\pi ft} \mathrm{d}t$$

inverse Fourier transform: back from frequency to time domain

$$w(t) = \mathcal{F}^{-1}\left[W(f)\right] = \int_{-\infty}^{\infty} W(f) e^{j2\pi ft} \mathrm{d}f$$

a reminder of Euler's formula

link between sine/cosine and $e^{j2\pi ft} = e^{j\omega t}$

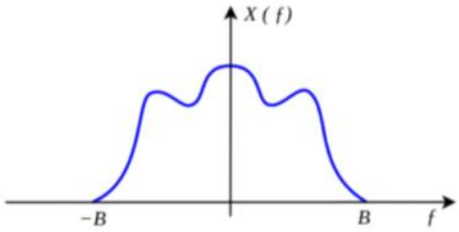
$$e^{j2\pi ft} = \cos(2\pi ft) + j\sin(2\pi ft) \qquad \cos(2\pi ft) = \frac{e^{j2\pi ft} + e^{-j2\pi ft}}{2}$$

$$e^{-j2\pi ft} = \cos(2\pi ft) - j\sin(2\pi ft) \qquad \sin(2\pi ft) = \frac{e^{j2\pi ft} - e^{-j2\pi ft}}{2j}$$

transmitted power and channel bandwidth

two of the main resources in wireless communications

expensive, so these must be used efficiently



transmitted power

average power of the transmitted signal

channel bandwidth

range of frequencies allocated to the transmission of the message

M) analog to digital

Saturday, 4 January 2025 14.34

from analog to digital

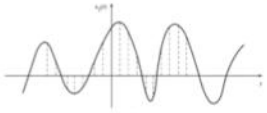
digitalization: analog-to-digital conversion (ADC)

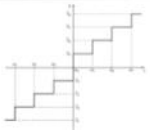
from continuous time and amplitude to discrete time and amplitude

Step 1: sampling: make the signal discrete in time

Step 2: quantization: make the signal discrete in amplitude

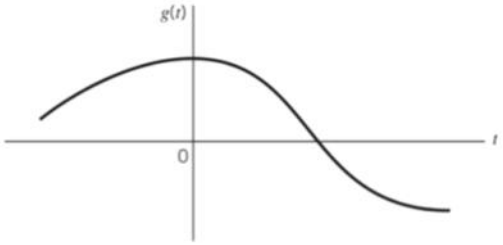
$x(t)$ —sampling→ $x(nT_s)$ —quantization→ ...01101010010010010... (bits)



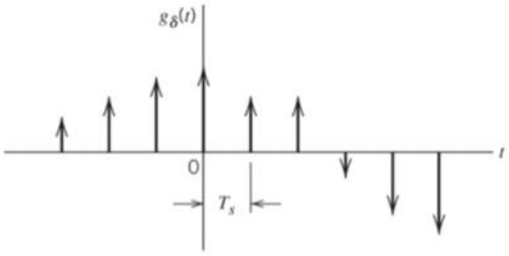


the sampling process

analog signal



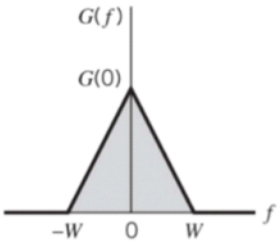
instantaneous sampled version of the signal



in math $g_\delta(t) = \sum_{n=-\infty}^{\infty} g(nT_s)\delta(t - nT_s)$

how often do I need to sample?

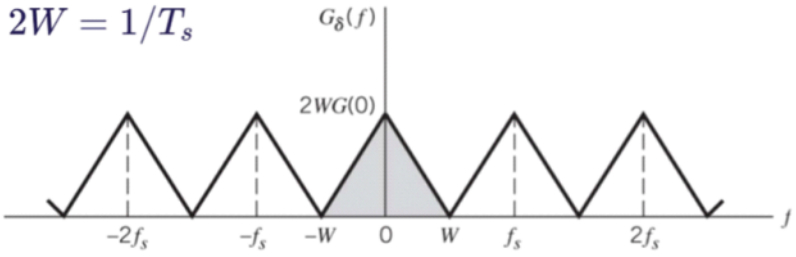
spectrum of band-limited signal $g(t)$



spectrum of $g(t)$ sampled at $f_s = 2W = 1/T_s$

this is the **Nyquist rate**

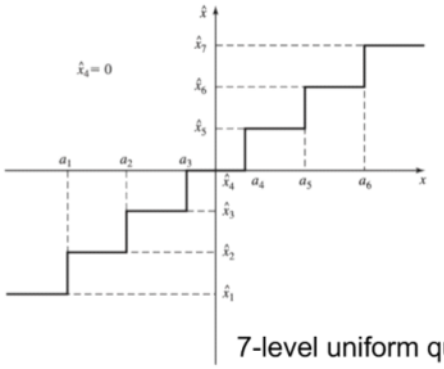
$$G_\delta(f) = \sum_{n=-\infty}^{\infty} g(nT_s)e^{-j2\pi nfT_s}$$



after sampling, quantize!

quantization

map voltage values from a continuous set into a smaller and countable set that will represent the voltage as levels



7-level uniform quantizer.

- the interval is proportional to the resolution that we can represent and after reconstruct the analog signal

- always some level of distortion will be introduced

now what?

- we have a digital signal, what should we do?

$x(t)$ —sampling→ $x(nT_s)$ —quantization→ ...0 1 1 0 1 0 1 0 0 1 0 0

- now, we need to introduce signal waveforms that represents the symbols (group of bits) that are going to be transmitted through the channel medium over time
- the idea is that each waveform is mapped to a symbol
- e.g., we have two waveforms  $= 0$  $= 1$

some notation

- let M be the number of waveforms
- denote the m-th waveform as $s_m(t)$
- the number of different binary symbols that can be represented by M is $k = \log_2 M$
- for example,

$M = 2$

$s_1(t) \Rightarrow 0 \quad s_2(t) \Rightarrow 1$

$M = 4$

$s_1(t) \Rightarrow 00 \quad s_2(t) \Rightarrow 01$
$s_3(t) \Rightarrow 10 \quad s_4(t) \Rightarrow 11$

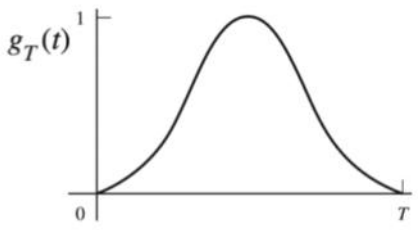
one-dimension: PAM

M-ary pulse amplitude modulation (PAM)

baseband

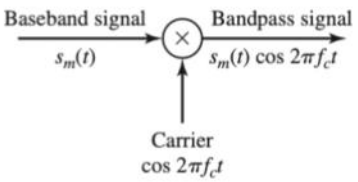
$$s_m(t) = A_m g_T(t)$$

A_m - amplitude, $g_T(t)$ – is a pulse of some arbitrary shape



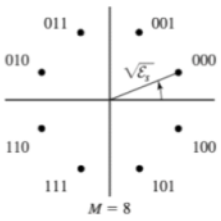
bandpass

$$u_m(t) = A_m g_T(t) \cos 2\pi f_c t$$



two-dimensions: QPSK

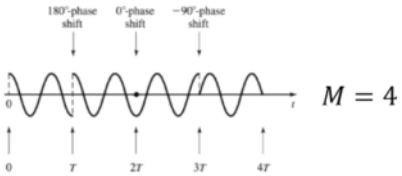
***M*-ary quadrature phase-shift keying**
assumptions: points have same energy



baseband
$M = 8$ signal-point constellation in two dimensions

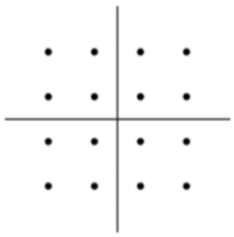
bandpass
$u_m(t) = s_t(t)\cos 2\pi f_c t$

$u_m(t) = g_T(t)\cos\left(2\pi f_c t + \frac{2\pi m}{M}\right)$

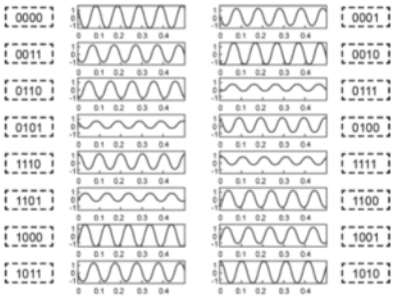


two-dimensions: QAM

***M*-ary quadrature amplitude modulation**
assumptions: points can have different energy



baseband
$M = 16$ signal-point constellation in two dimensions



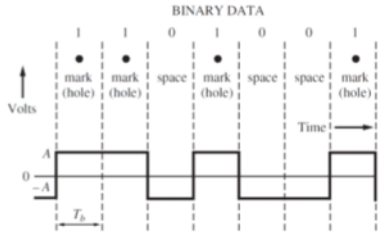
bandpass
$u_{mn}(t) = A_m g_T(t)\cos(2\pi f_c t + \theta_n)$,

for $m = 1, \ldots, M_1, n = 1, \ldots, M_2, M = M_1 + M_2$

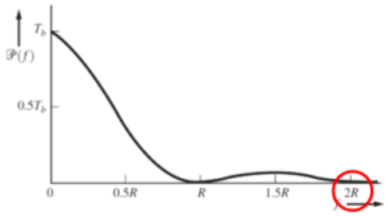
bandwidth of a baseband waveform (1)

- power spectral density (PSD) – power content in the freq. domain

$$P_w(f) = \lim_{T\to\infty}\left(\frac{|W_T(f)|}{T}\right)$$



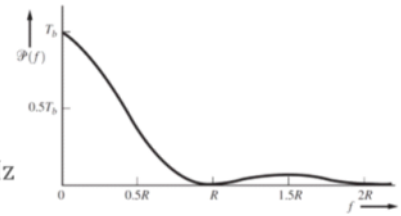
- BPSK waveform with bitrate $R = 1/T_b$
- worst case: rectangular wave of freq. $R/2$
- equivalent PSD: $P(f) = A^2T_b\left(\frac{\sin \pi f T_b}{\pi f T_b}\right)^2$
- bandwidth definition based on the PSD second null – $2R$



bandwidth of a baseband waveform (2)

example 1

- an RS-232 link operates at a rate of 230,400 baud
- what is the bandwidth of the communication signal?
  - $230{,}400 \text{ baud} \equiv 230{,}400 \text{ bit/s} \quad \Rightarrow \quad B = 2{\times}230{,}400 = 460.8 \text{ kHz}$



example 2

- a BPSK baseband signal has a bandwidth of 20 MHz
- 5% of the bits contain link control information
- what is the system throughput?
  - bitrate: $R = \frac{B}{2} = \frac{20{\times}10^6}{2} = 10 \text{ Mbit/s}$
  - throughput: $T = (100\% - 5\%)R = 95\%{\times}10{\times}10^6 = 9.5 \text{ Mbit/s}$

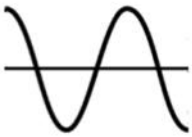
transmitting the bits

- now we have the M-ary signal waveforms, how do we transmit a given sequence of bits

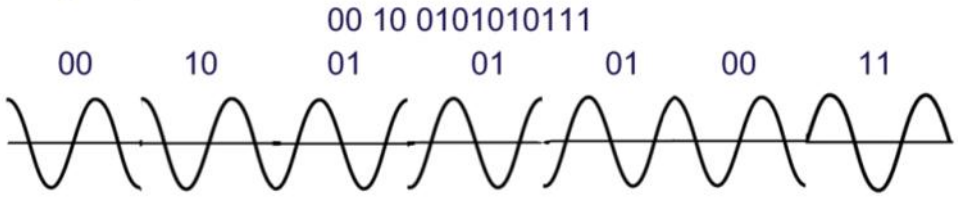
1. break the sequence of bits into k-bit blocks or symbols
2. associate each symbol to its corresponding waveform
3. transmit this waveform for a symbol (signaling) interval of T seconds

an example

- assume a 4-QPSK with waveform $g(t)$:



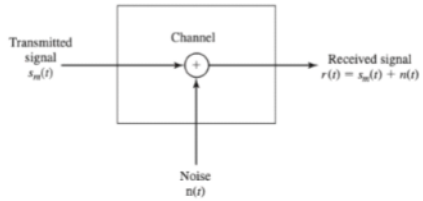
we have the following sequence of bits:



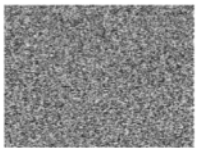
additive white Gaussian noise channels

assuming baseband, we consider the following model:

$$r(t) = s_m(t) + n(t), 0 \le t \le T$$



- what is white Gaussian noise:
  - model the imperfections introduced by the channel
    -- things that we are uncertain about
  - "white" comes from the fact that
    the power of the noise is constant over time and frequency
    $S_n(f) = \frac{N_0}{2}$ W/Hz, where N_0 is the noise power
  - Gaussian comes from a theorem from statistics that is known as the central limit theorem



white noise image

how do we received data?

we have defined very well the transmitter side, but now how do we receive data?

we divide the functions of the received into two parts:

- **signal demodulator:** converts the received waveform $r(t)$ into a vector with length equal to the dimension of the modulation
- **signal detector:** from the vector, guess which of the M waveforms was received

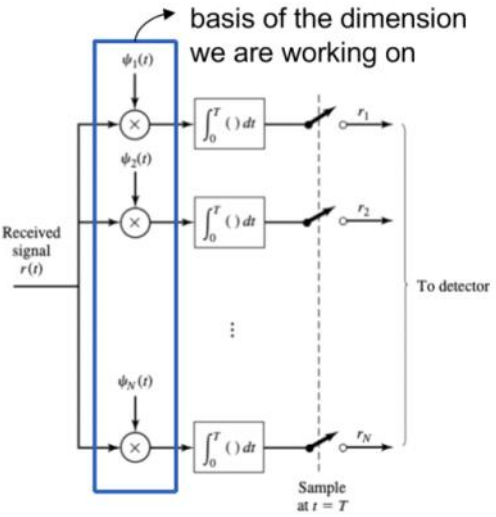
signal demodulator

$\int_0^T (\,)\,dt$

integrator block

correlation-type demodulator

basis of the dimension we are working on



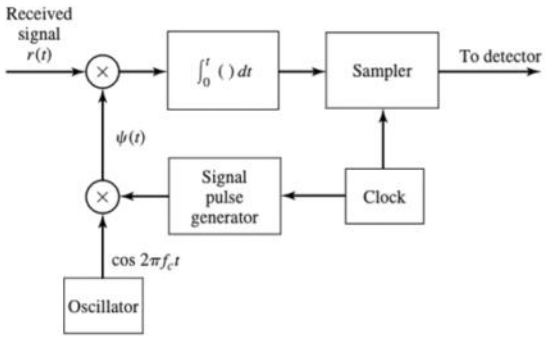
intuition

1. first you correlate the received signal in time to break into one of the components that represents the dimension of the signal waveform space
2. then you integrate over time to extract the energy received over that dimension
3. finally, you sample and get a vector representation

signal demodulator: PAM example

PAM case

we just have one-dimension



transmitted signal

$$u_m(t) = A_m g_T(t) \cos 2\pi f_c t, \quad 0 \le t \le T$$

received signal

$$r(t) = A_m g_T(t) \cos 2\pi f_c t + n(t), \quad 0 \le t \le T$$

noise signal

$$n(t) = n_c(t) \cos 2\pi f_c t - n_s(t) \sin 2\pi f_c t$$

integration

$$\int_0^T r(t)\psi(t)\,dt = A_m\sqrt{\frac{2}{\mathcal{E}_g}}\int_0^T g_T^2(t)\cos^2 2\pi f_c t\,dt + \int_0^T n(t)\psi(t)\,dt$$

$$= A_m\sqrt{\mathcal{E}_g/2} + n$$

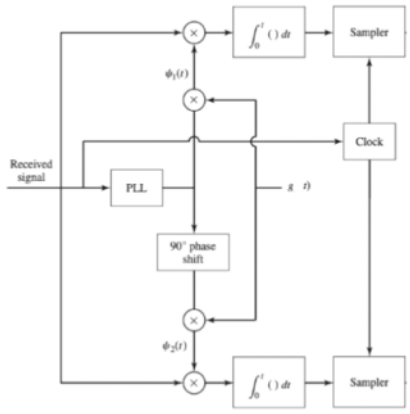
$\downarrow$

energy of g_T

signal demodulator: QAM example

QAM case

we have two-dimensions



received signal

$$r(t) = A_{mc} g_T(t) \cos(2\pi f_c t + \phi) + A_{ms} g_T(t) \sin(2\pi f_c t + \phi) + n(t)$$

basis

$$\psi_1(t) = \sqrt{\frac{2}{\mathcal{E}_g}} g_T(t) \cos(2\pi f_c t + \hat{\phi})$$

$$\psi_2(t) = \sqrt{\frac{2}{\mathcal{E}_g}} g_T(t) \sin(2\pi f_c t + \hat{\phi})$$

integration

$$r_1 = A_{mc}\sqrt{\mathcal{E}_s} \cos(\phi - \hat{\phi}) + A_{ms}\sqrt{\mathcal{E}_s} \sin(\phi - \hat{\phi}) + n_c \sin\hat{\phi} - n_s \cos\hat{\phi}$$
$$r_2 = A_{mc}\sqrt{\mathcal{E}_s} \sin(\phi - \hat{\phi}) + A_{ms}\sqrt{\mathcal{E}_s} \cos(\phi - \hat{\phi}) + n_c \sin\hat{\phi} - n_s \cos\hat{\phi}$$

signal detector

maximum-a-posteriori principle

- after the detector, we end up with a vector $\boldsymbol{r} = [r_1, r_2, \dots, r_N]$ where N is the number of dimensions used to represent the waveforms

- from $\boldsymbol{r}$ we need to decide which $s_m(t)$ was transmitted

- one way to do so is by using a decision rule based on posterior probabilities:

$$P(signal\ s_m(t)\ was\ transmitted|\boldsymbol{r})\ for\ m = 1, 2, \dots, M$$

intuition: find the $s_m(t)$ with the largest probability of being transmitted given that $\boldsymbol{r}$ was received

M) Modulation3

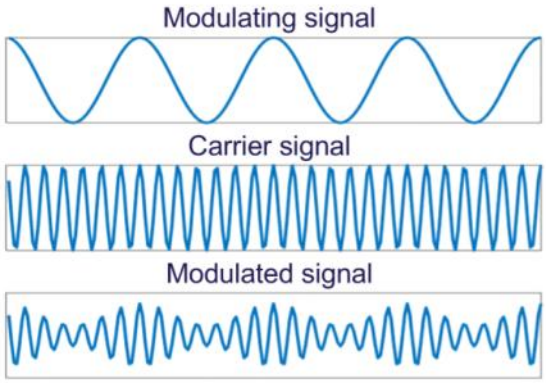
Saturday, 4 January 2025 14.37

analog and digital modulation

analog modulation

modulating signal is **continuous**

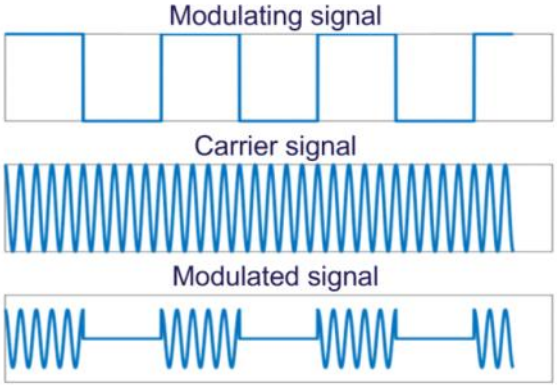
first implemented: AM and FM radio



digital modulation

modulating signal is **discrete**

used in modern communication systems



analog modulation: amplitude, frequency, and phase

$$s(t) = \underbrace{A}_{\text{amplitude}} \cos(\omega_c t + \underbrace{\phi}_{\text{phase}})$$

ω_c: carrier angular frequency

- amplitude modulation (AM)
- angle modulation/phase modulation (PM)
- frequency modulation (FM)

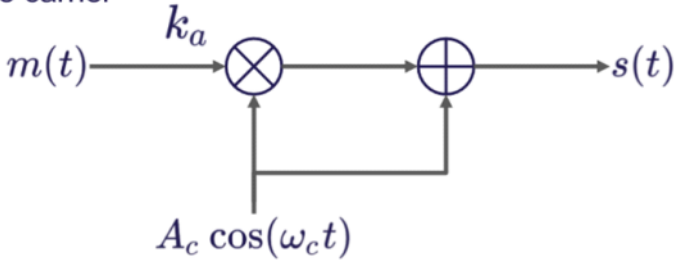
amplitude modulation (AM)

$m(t)$: the baseband signal that carries the information

$s(t)$: the modulated signal with angular carrier frequency $\omega_c = 2\pi f_c$

classical AM

1. multiply the baseband signal by a sinusoid carrier signal
2. add the carrier



k_a: amplitude sensitivity

AM applications and principle

applications
broadcast and radio (540 to 1600 kHz). **cheap user equipment**

principle
the information is carried by the amplitude of the modulated signal

why adding the carrier? Equivalent to a DC shift

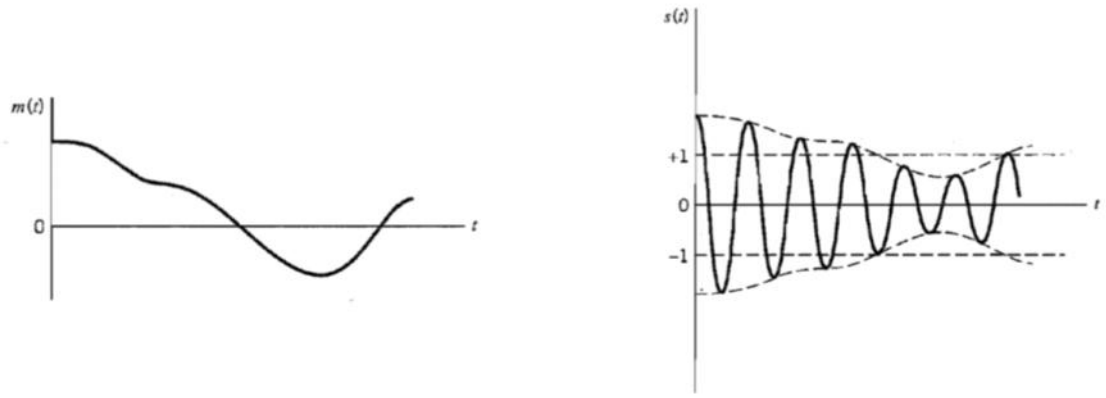
makes the message easy to decode: envelope – DC shift

time domain: the signal modulates the amplitude of the carrier

frequency domain: the carrier shifts the frequency of the signal by ω_c

AM signal in time

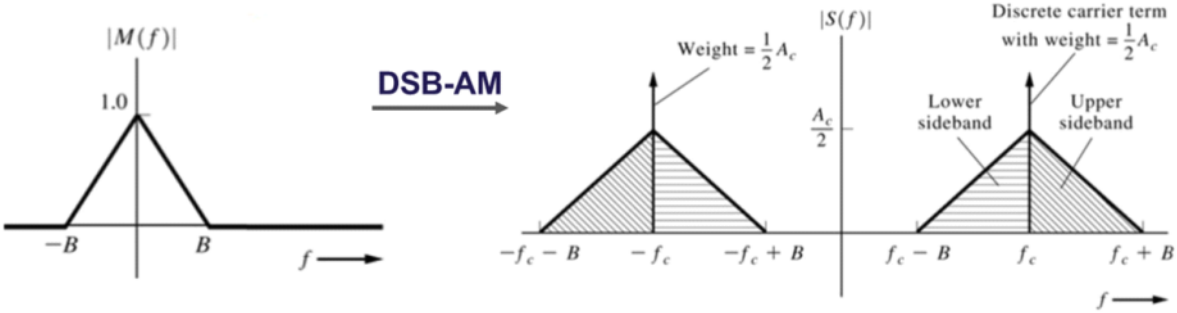
$$s(t) = A_c\left[1 + k_a m(t)\right]\cos(\omega_c t)$$



AM signal in frequency (spectrum)

traditional AM is actually double sideband with carrier AM (DSB-AM)

$$S(f) = \frac{A_c}{2}\left[\delta(f - f_c) + \delta(f + f_c)\right] + \frac{k_a A_c}{2}\left[M(f - f_c) + M(f + f_c)\right]$$



example: single-tone modulation

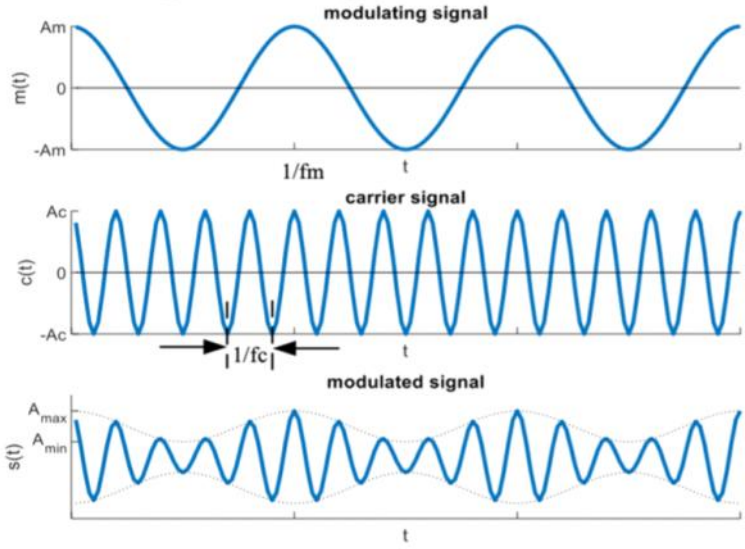
single-tone modulating signal: $m(t) = A_m \cos(\omega_m t)$

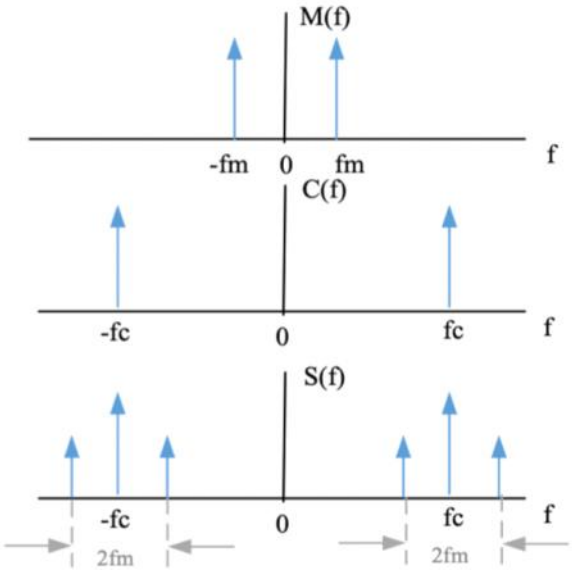
modulated signal

$$\begin{aligned} s(t) &= A_c \left[1 + k_a m(t)\right] \cos(\omega_c t) \\ &= A_c \left[1 + k_a A_m \cos(\omega_m t)\right] \cos(\omega_c t) \end{aligned}$$

modulation factor $\mu = k_a A_m$

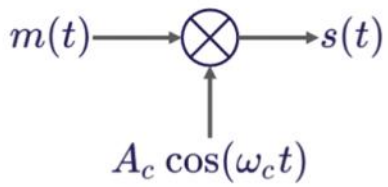
example: single-tone modulation



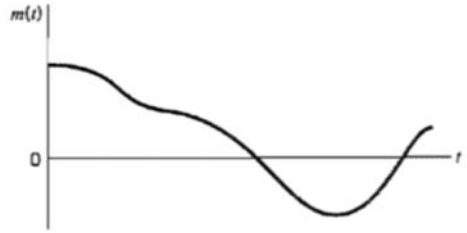


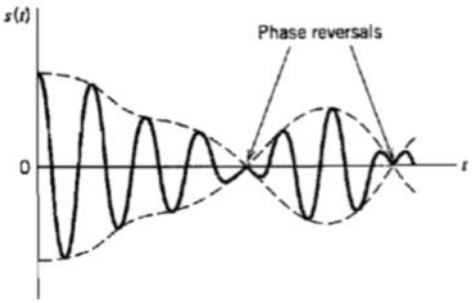
double-sideband suppressed carrier (DSB-SC)

multiply the signal by the carrier but skip its addition



phase reversal whenever the signal crosses 0

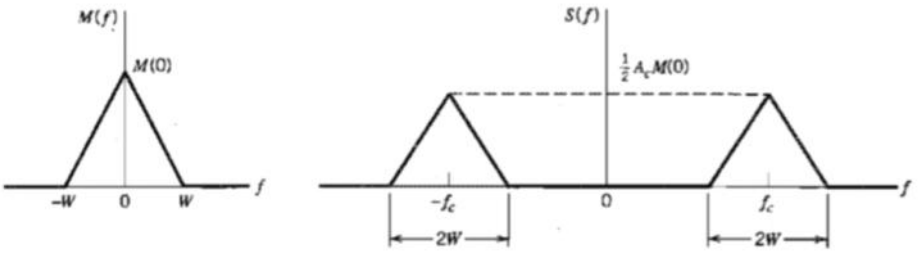




double-sideband suppressed carrier (DSB-SC)

spectrum

$$S(f) = \frac{1}{2}A_c\Big(M(f - f_c) + M(f + f_c)\Big)$$



single sideband (SSB)

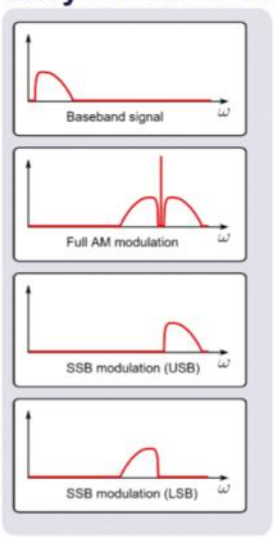
why occupying double the bandwidth if both sides carry the same information?

SSB

similar to DSB, just change settings of band pass filter

pro: only uses half the bandwidth

con: recovering the original signal

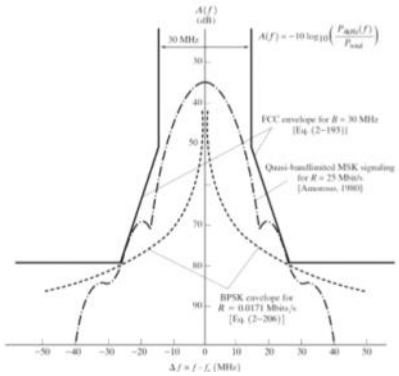


Source: Wikipedia

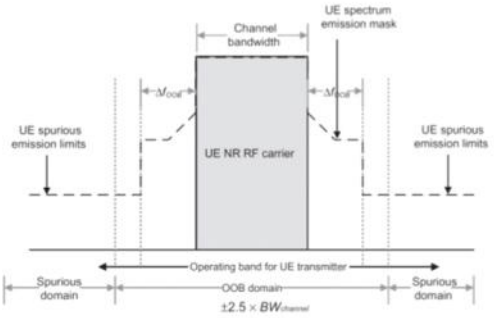
spectrum emission mask (SEM)

- regulatory agencies control how bands of the radio spectrum are used
- spectral masks are defined to limit interference at neighboring bands

FCC SEM for freqs. below 15 GHz:



SEM for 5G devices:



M) amplitude demodulation

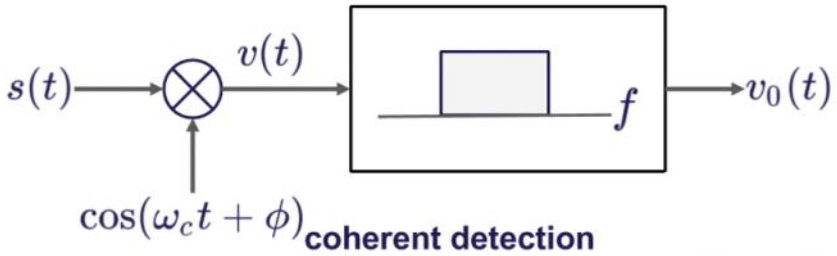
Saturday, 4 January 2025 15.25

synchronous demodulation

recover the signal by multiplying by the carrier and then using a low pass filter

$$s(t) = m(t)\cos(\omega_c t)$$

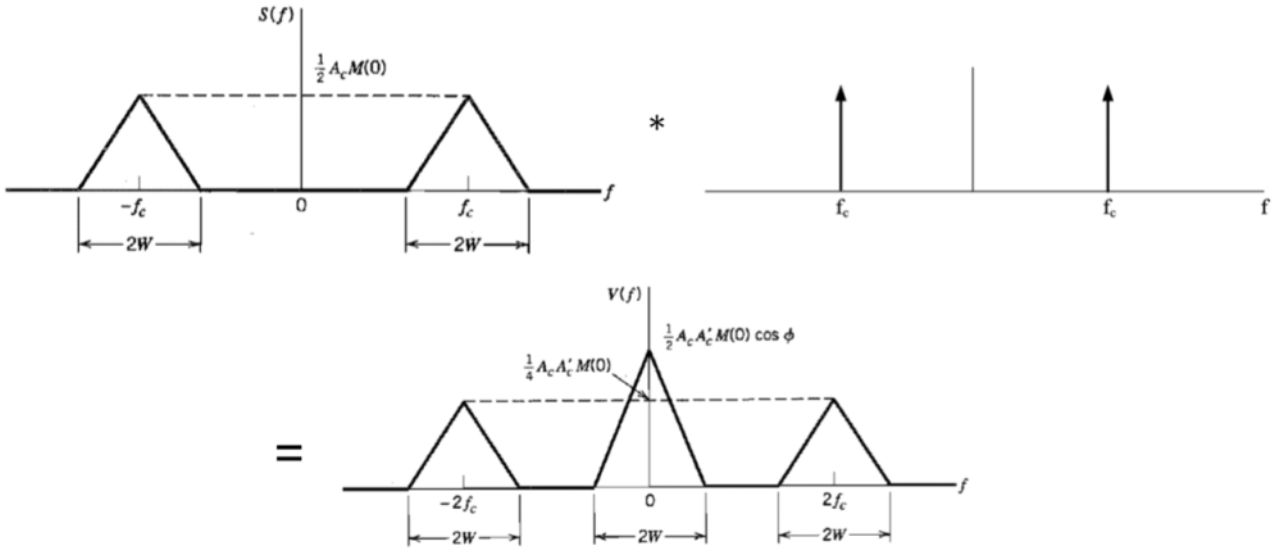
$$v(t) = s(t)\cos(\omega_c t) = m(t)\left[\frac{1}{2} + \frac{1}{2}\cos(2\omega_c t)\right]$$



coherent detection

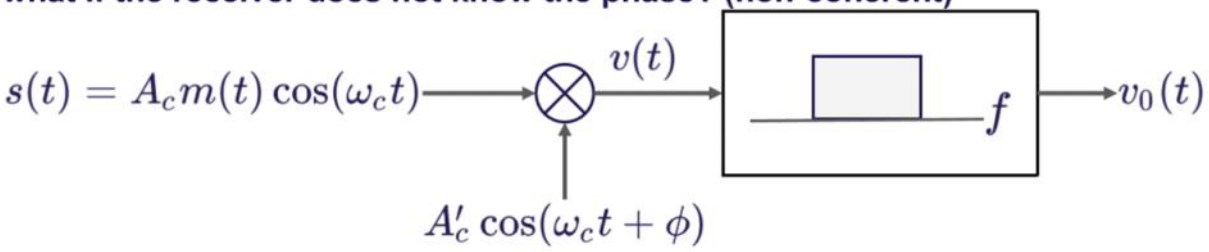
problem: the receiver must know the carrier perfectly

synchronous demodulation



synchronous demodulation

what if the receiver does not know the phase? (non-coherent)



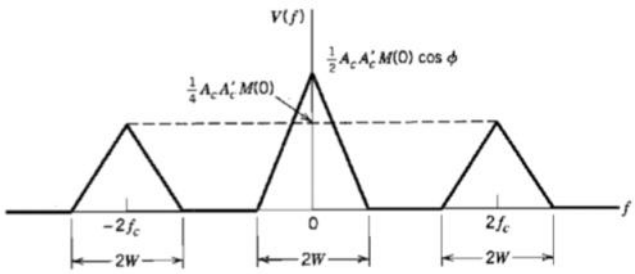
$$v(t) = s(t)A'_c\cos(\omega_c t + \phi)$$
$$= m(t)A_c A'_c \cos(\omega_c t)\cos(\omega_c t + \phi)$$
$$= \frac{m(t)}{2}A_c A'_c\Big(\cos(2\omega_c t + \phi) + \cos(\phi)\Big)$$

remember:
$2\cos(\theta)\cos(\phi) = \cos(\theta - \phi) + \cos(\theta + \phi)$

synchronous demodulation

after low-pass filter

$$v_0(t) = \frac{1}{2}A_c A'_c \cos(\phi)m(t)$$



if the phase is constant and $\neq \pm\pi/2$

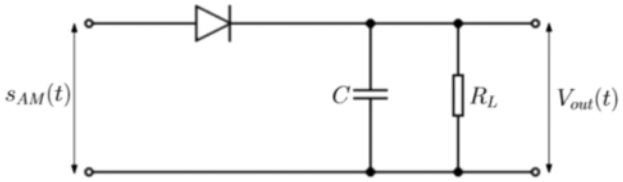
$v_0(t)$ is proportional to $m(t)$

if the phase is $= \pm\pi/2$

$v_0(t) = 0$

envelope detector

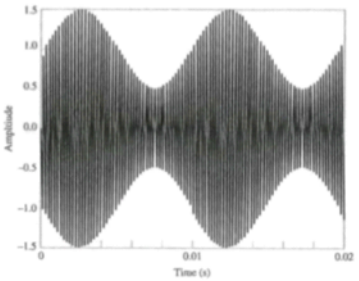
non-coherent detection



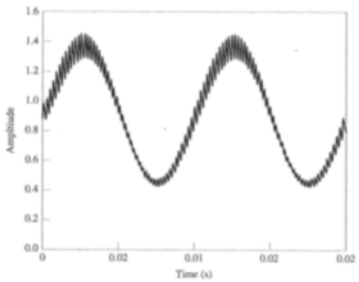
the capacitor charges in the positive half-cycle up to the peak value of the input

the capacitor discharges slowly in the negative half-cycle

cheap receiver that requires $k_a < 1$



AM wave input



Envelope detector output

demodulation in AM: wrap-up

coherent (synchronous) detection

- the receiver uses a local carrier of the same frequency and phase to detect the signal
- cross-correlation of replica signals at the receiver

non-coherent detection

- no replica signals required
- does not exploit phase reference information
- **pro:** lower complexity at the receiver
- **con:** worse performance

M) phase and frequency mod

Saturday, 4 January 2025 15.38

demodulation in AM: wrap-up

coherent (synchronous) detection

- the receiver uses a local carrier of the same frequency and phase to detect the signal
- cross-correlation of replica signals at the receiver

non-coherent detection

- no replica signals required
- does not exploit phase reference information
- **pro:** lower complexity at the receiver
- **con:** worse performance

phase modulation (PM)

the instantaneous **angle** $\theta_i(t)$ changes linearly as a function of $m(t)$

$$\theta_i(t) = 2\pi f_c t + k_p m(t)$$

where k_p is the **phase sensitivity** of the modulator

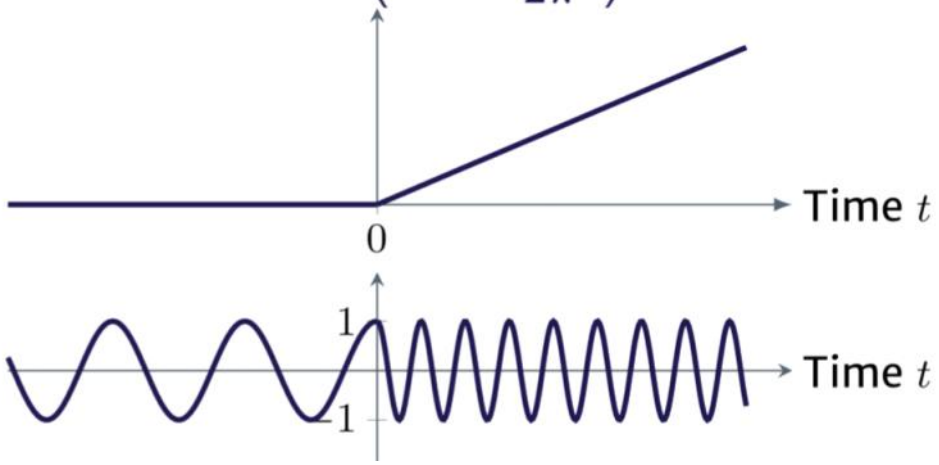
the modulated signal is $s(t) = A_c \cos\left(2\pi f_c t + k_p m(t)\right)$

if $m(t)$ is a first-order function, the modulated signal will be shifted from f_c to $f_c + f_m$

$$\begin{aligned}\theta_i(t) &= 2\pi f_c t + k_p a t \\ &= 2\pi\left(f_c + \frac{k_p a}{2\pi}\right) t = 2\pi\left(f_c + f_m\right) t\end{aligned}$$

phase modulation (PM)

$$\theta_i(t) = 2\pi f_c t + k_p a t = 2\pi \left(f_c + \frac{k_p a}{2\pi} \right) t = 2\pi \left(f_c + f_m \right) t$$



frequency modulation (FM) principle

the instantaneous **frequency** $f_i(t)$ changes linearly as a function of $m(t)$

$$f_i(t) = f_c + k_f m(t)$$

where k_f is the **frequency sensitivity** of the modulator

the angle is

$$\theta_i(t) = 2\pi \int_0^t f_i(\tau) \mathrm{d}\tau = 2\pi \left(f_c t + k_f \int_0^t m(\tau) \mathrm{d}\tau \right)$$

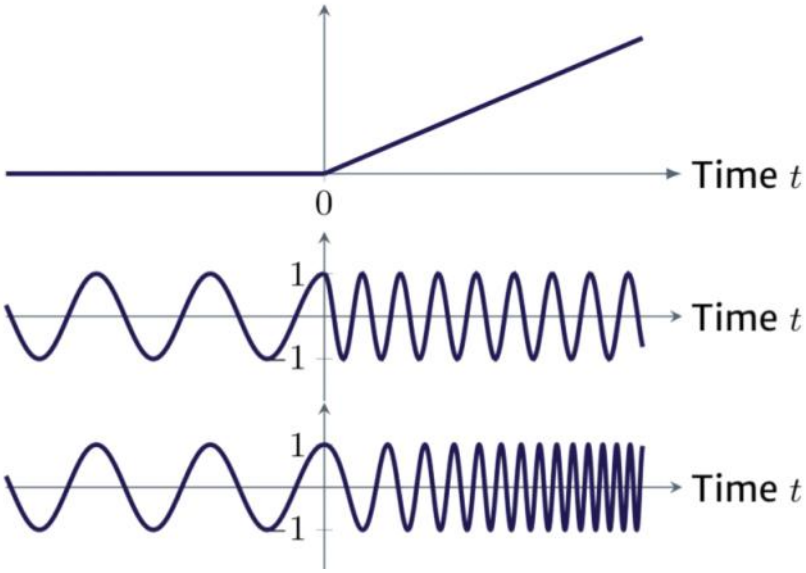
the modulated signal is

$$s(t) = A_c \cos\left[2\pi \left(f_c t + k_f \int_0^t m(\tau) \mathrm{d}\tau \right) \right]$$

frequency modulation (FM)

$$\theta_i(t) = 2\pi \left(f_c + f_m \right) t$$

$$\theta_i(t) = 2\pi \left(f_c t + k_f \int_0^t m(\tau) \mathrm{d}\tau \right)$$



angle modulation: PM vs. FM

		Phase modulation (PM)	Frequency modulation (FM)
instantaneous phase	$\theta_i(t)$	$2\pi f_c t + k_p m(t)$	$2\pi\left(f_c t + k_f \int_0^t m(\tau)\mathrm{d}\tau\right)$
instantaneous frequency	$f_i(t)$	$f_c + \frac{k_p}{2\pi}\frac{dm(t)}{dt}$	$f_c + k_f m(t)$
modulated wave	$s(t)$	$A_c \cos\left(2\pi f_c t + k_p m(t)\right)$	$A_c \cos\left[2\pi\left(f_c t + k_f \int_0^t m(\tau)\mathrm{d}\tau\right)\right]$

k_p: phase-sensitivity factor; k_f: frequency-sensitivity factor

properties of angle modulation

constant terms in the transmitted wave

the amplitude of the PM and FM waves A_c

the average transmitted power $\bar{P} = \frac{A_c}{2}$

non-linearity of the modulation process

$m(t) = m_1(t) + m_2(t)$ $\qquad s(t) = A_c \cos[2\pi f_c t + k_p\,(m_1(t) + m_2(t))]$

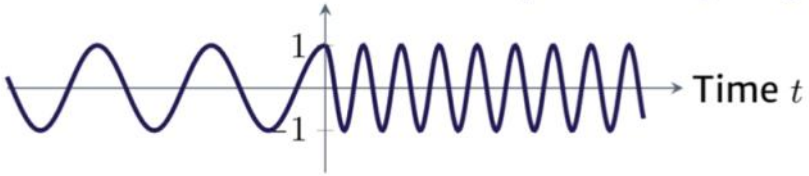
$s_1(t) = A_c \cos[2\pi f_c t + k_p m_1(t)]$ $\qquad s_2(t) = A_c \cos[2\pi f_c t + k_p m_2(t)]$

$$\boxed{s(t) \neq s_1(t) + s_2(t)}$$

properties of angle modulation

irregularity of zero-crossing

zero-crossing: points in time where the waveform amplitude changes sign



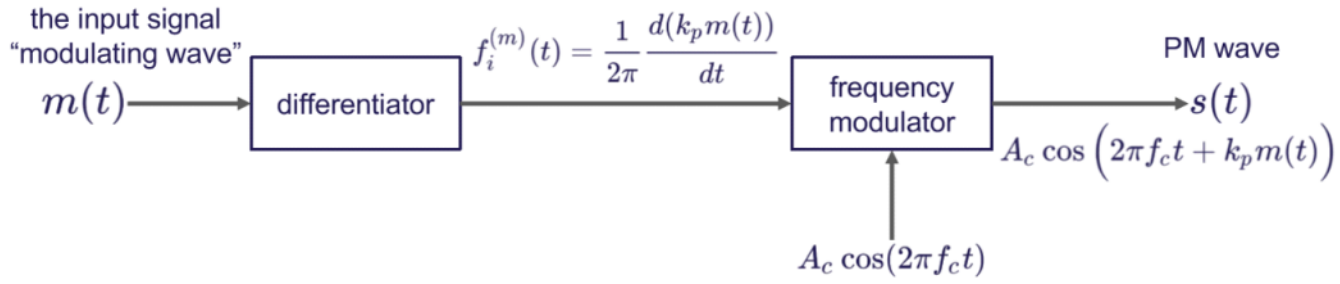
the zero-crossings of the modulated wave carry the information of the signal $m(t)$

visualization difficulty: attributed to the non-linear nature

trade-off: increased transmission bandwidth for improved noise performance

less sensitive to additive noise obtained at the expense of increased Tx. bandwidth

modulation process in PM



$f_i^{(m)}(t) = \frac{1}{2\pi}\frac{d(k_p m(t))}{dt}$ is the instantaneous frequency of the modulating wave