3 Data Link Layer

arrangement

The lowest layer of the OSI Model enables the transmission of bits via media and establishes the corresponding specifications. This layer is not discussed further here because computer scientists do not necessarily deal directly with the details of this layer.

An important task for the second layer of the OSI Model, namely the Data Link Layer, which is located above the first layer, is regulating access of end systems to a shared transmission medium. Such regulation is necessary to ensure reliable transmission of bit sequences. At this layer, the bit sequences have predefined structures and the resulting data units are called frames.

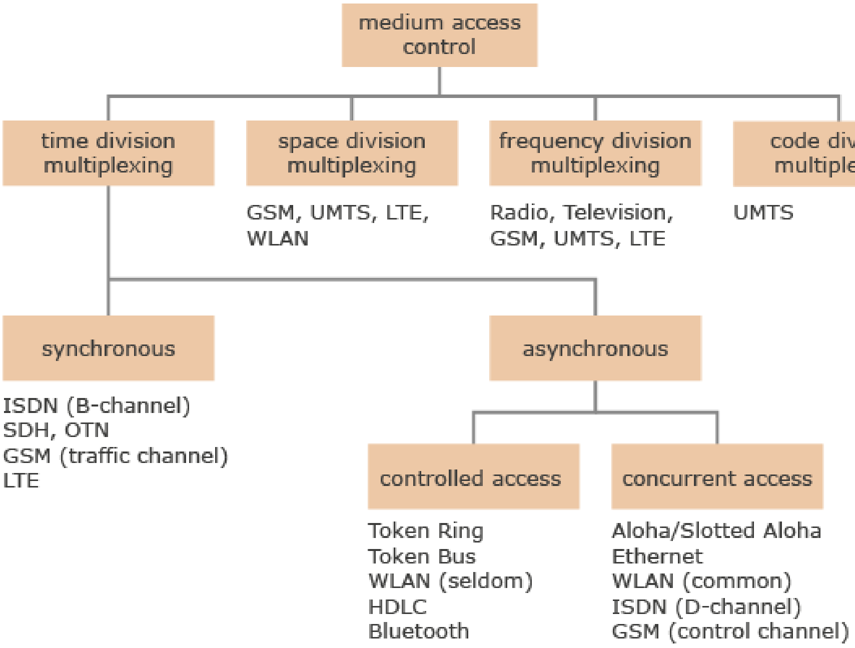
These frames must, however, also be protected against transmission errors, which is another essential task of the Data Link Layer. It must be determined whether frames have arrived incorrectly or have not arrived at all at the receiving end. Such transmission errors must then be dealt with, which is often done by frame retransmission.

3.1 Multiplexing

arrangement

3.1.5 Code Division Multiplexing

In many networks, you find a shared medium, i.e., a set of end systems that would like to access a commonly used medium. In this situation, rules are needed so that either parallel access is possible or algorithms are used to make conflicts impossible or try avoid them. The task to be solved is called **medium access control**.



Multiplexing techniques and classification of technologies

There are **four multiplexing** techniques which are depicted as basic options in the figure. However, these do not have to be used exclusively, but can be used in combination with each other. You can also see this in the allocation of technologies where, for example, the mobile telecommunications technologies GSM and LTE represent combinations of time division, frequency division and space division multiplexing.

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/G1LqxrQ38-s
Multiplexing

Time division multiplexing also contains another distinction between synchronous and asynchronous. **Synchronous** means that a coordination of clocks between the systems is necessary. Which system is permitted to send at a certain moment is determined according to the time. If the systems work **asynchronous**, however, the time is not coordinated. In this case, you can further distinguish between controlled access and concurrent access. With **controlled access**, no data collisions can happen on the medium; with **concurrent access**, the aim is to avoid collisions, but they can occur.

Before looking at multiplexing techniques in more detail, the concepts of simplex, half-duplex and duplex will be introduced as **three operation modes**.

3.1.1 Operation Modes

arrangement

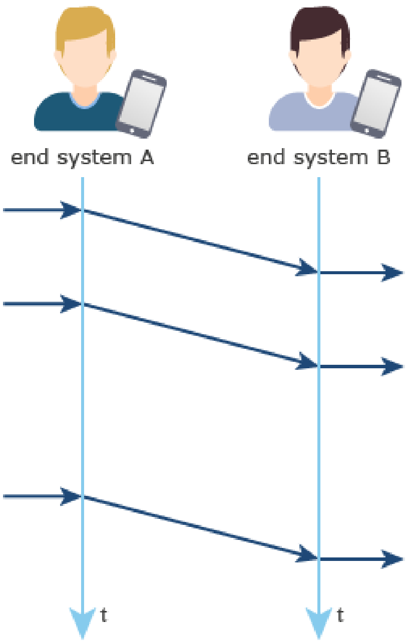
The so-called **operation modes** reflect three different scenarios. Unidirectional (simplex) operation for communication networks is a rather special case. In contrast it is often an important question whether a full duplex implementation is possible or whether half-duplex is unavoidable.

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/nYIHVjM9nm0
Operation modes

3.1.1.1 Simplex

The first scenario is called **simplex**. In this situation the communication between two systems is in principle only possible in one direction, i.e. system A can send data units to system B, but system B cannot respond. This is especially critical for transmission errors because system B cannot provide feedback about whether the data arrived correctly or not. In this case, it is only possible to use so-called Forward Error Correction, where the data contain redundant information so the receiver can correct transmission errors on its own.

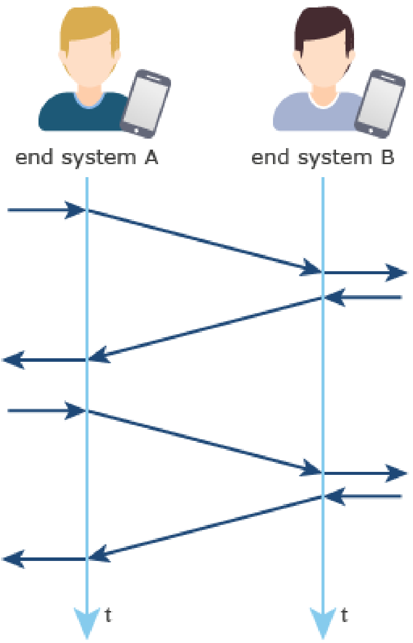


Simplex communication between two end systems

In practice such scenarios exist for broadcasting of radio or television programs, i.e., when the broadcasting occurs via satellites, radio towers or cable television networks. Here many participants simultaneously receive the same program and cannot provide feedback.

3.1.1.2 Half-duplex

When using **half-duplex**, communication can in principle occur in two directions between systems A and B but only in one direction at a time. If communication needs to take place in the other direction, switchover times have to be followed. If both systems attempt to send data units at the same time, then there is interference on the medium, which leads to bit errors.



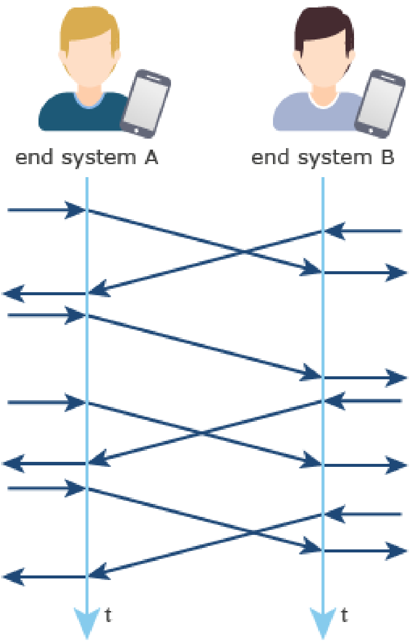
Half-duplex communication between two end systems

Half-duplex scenarios can also be present with more than two systems. Then only one out of n systems can send a data unit at a time.

Examples of half-duplex operation mode are walkie-talkies, wireless transmissions (including Wireless LAN) and bus topologies such as the original Ethernet.

3.1.1.3 Full Duplex

In **full duplex** operation mode, communication can take place simultaneously between two systems A and B. There are two independent transmission channels where it does not matter whether communication occurs in the opposite direction at the same time. In such a scenario, the switchover times are eliminated and resynchronization is not necessary.



Duplex communication between two end systems

Communication is restricted to two systems if the full duplex operation mode is used.

Examples of it are transmissions via DSL, where different frequency ranges are used for both directions, or modern Ethernet installations.

3.1.2 Time Division Multiplexing

arrangement

The multiplexing techniques that are most important for local networks are time division multiplexing techniques. As already mentioned, a distinction can be made between **synchronous** and **asynchronous** time division multiplexing. A further distinction can also be made if asynchronous time division multiplexing is used. **Controlled access** prevents that collisions happen on the medium. When using **concurrent access** collisions on the medium can occur, but the aim is to avoid collisions as much as possible. The possibility of collisions is of course the drawback of this technique. It means that you cannot guarantee that the data frames will arrive. However, management is significantly easier, and if you look at the method over a longer period of time ever-repeating collisions are very unlikely.

If time division multiplexing is used, only one system is able to send at a certain time. Therefore, we have a half-duplex situation. In contrast, the use of other multiplexing techniques enables full duplex operation.

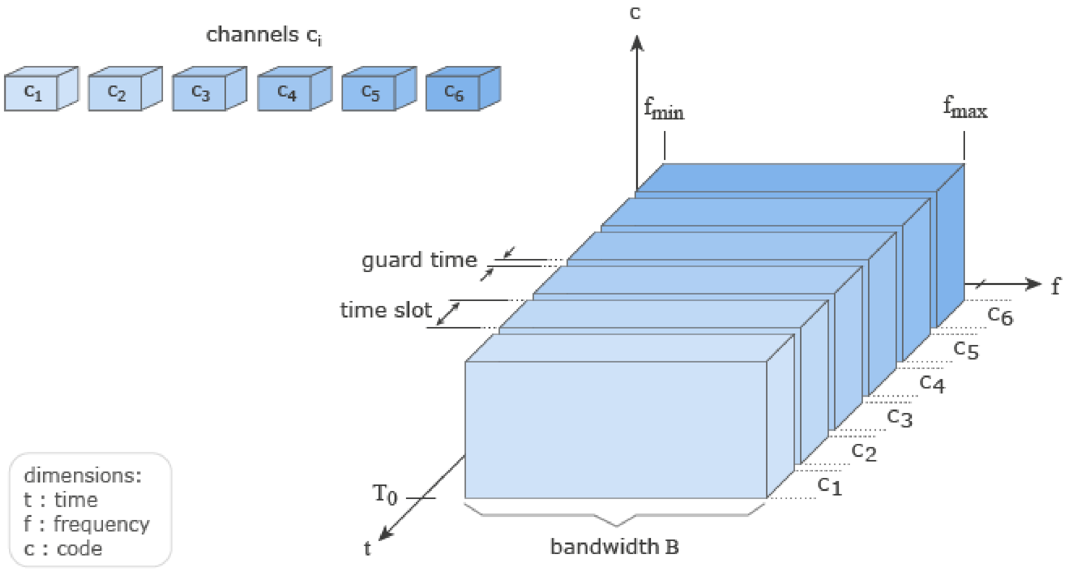
In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/ETYWs1Jnq5c

Time division multiplexing

3.1.2.1 Fixed Arbitration Strategy

In the fixed arbitration strategy, which requires permanently synchronized clocks, all system gets the possibility to transmit bits over the medium one after the other. In each time interval, a system can make use of the entire bandwidth. This means it can use all frequencies between the lowest frequency f_{min} and the highest frequency f_{max} for data transmission. The time intervals are called **time slots**. If all time slots have passed, then the first time slot will start again, i.e., the turn of each system can be found periodically.



Technique with fixed allocation with 6 time slots

The allocation of time slots is done by the network, e.g., through a base station in mobile telecommunications. Alongside mobile telecommunications, another application is wireless telephony using the DECT standard.

Using fixed arbitration you can provide guarantees for the end systems that they get a minimum bit rate and that there is a maximum waiting time. However, the whole system is not very flexible especially if bit rate requirements vary. It may happen then that time slots elapse unused even if other systems have high transmission demands that exceed their time slots.

This technique is not used in local area networks for data transmission.

notice

> In some publications, only this technique is regarded as time division multiplexing (and not the asynchronous techniques).

3.1.2.2 Variable Arbitration Strategy

There are basically two options how controlled access methods can work: by polling or token-based. In a **polling** scheme there is a special coordinator that controls the access to the medium. WLAN can operate in such a way in one (rarely used) variant. In this case, the access point acts as the coordinator. All terminals may then only send data if the access point has explicitly granted them transmission rights. In this way the access point queries all terminals one after the other (i.e., gives them the opportunity for transmission and also sends them data). Although this technique seems to be relatively simple, there are some difficulties to be aware of: All end systems must register with the access point in order to be included into the polling sequence. Even end systems that currently have nothing to send have to announce that they have nothing to send. Otherwise, the access point cannot proceed to the next end system. End systems that have not logged out properly also remain in the polling sequence.

Other controlled access methods are **token-based techniques**. They are used in networks with ring topologies or bus topologies. The explanation that follows refers to a ring topology. Here a so-called **token** that is passed from one station to the next controls access rights to the transmission medium. A token is an identifier in the form of a special bit pattern; a distinction is made between a free and occupied token.

The station that currently owns a free token has the right to send. If it currently has no transmission needs, it passes the free token to the next station. But if it wants to transmit, it marks the token as occupied and transmits it along with the data it wants to send (see video).

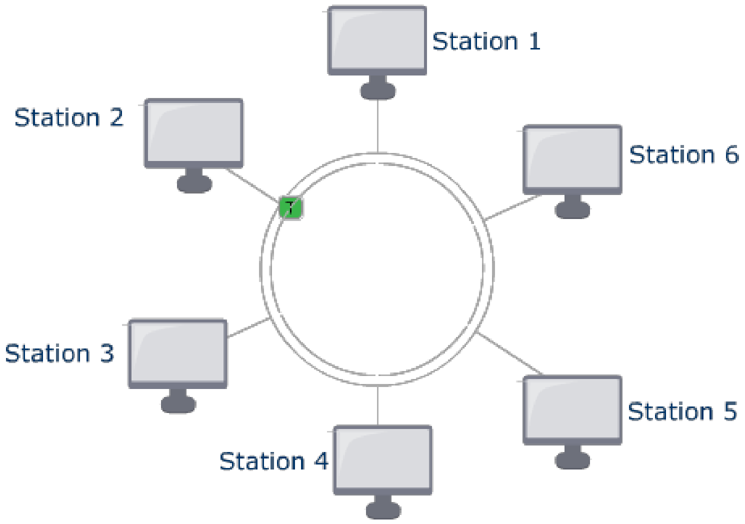
All connected stations use an address comparison to check whether they are the intended destination station. The destination station copies the data intended for it in its receive buffer and sends the occupied token including the sent data further along the ring until they have reached the source again.

The source now has the task to remove the data from the ring and to pass on a new token marked as free, even if it wants to continue to send data. This means every station gets the right to send a single time per token cycle. This technique leads to a fair allocation of the transmission medium for all stations that wish to transmit.

In the online version an animation is shown here.

Token-based arbitration in a ring topology

Begin printversion



The station which currently holds the free token has the right to send. If it does not have anything to send right now, it forwards the free token to the neighboring station.

In our example station 2 wants to send a data unit to station 5. Once it receives the free token, it marks the token as occupied. Then it sends the data unit and adds the occupied token to the end of the data unit. Now all stations on the ring check the target address whether they are the destination of the data unit. The destination station copies the data which is intended for it into its receive buffer. It then forwards the occupied token including the data via the ring. Once the data unit including the token has again reached the source, the source has to take care to remove the data unit from the ring and to generate a new token which is marked as a free token. This is also the case if the source wants to continue to send data.

Therefore, each station gets a free token exactly once in one token cycle. This way of operation leads to a fair assignment of the transmission media to all stations that would like to send data units.
End printversion

In principle this technique does not require a central facility because the process is regulated solely by the connected stations. To avoid errors such as token loss or token duplication and to verify the proper operation of the entire process, it makes sense to select one station as the monitor station. The monitor station checks if the ring operation works as expected.

The effective transmission time for the data consists of the waiting period for transmission authorization (token) and the pure physical transmission time in the network. The waiting period is strongly dependent on the network load, but it cannot be higher than an upper limit defined by the maximum token circulation time. This worst case is present if all connected stations want to transmit a maximum amount of data. Since a maximum waiting time can be guaranteed, the method can be regarded as a deterministic procedure.

To increase the performance, enhancements have been developed where several circulating tokens are possible or where it is allowed to attach own data to an occupied token.

annotation

Token-based procedures are standardized for ring structures (Token Ring, IEEE 802.5, sponsored by IBM) and bus systems (Token Bus, IEEE 802.4, sponsored by General Motors) for application in local fixed networks. The products based on this were, however, completely pushed out of the market around 1990 by Ethernet. Nevertheless, a basic understanding of the technique is relevant because you use related techniques in environments with hard real-time conditions (such as in factory automation or vehicles).

3.1.2.3 Random Access Strategy

When using random access strategies, each end system implements an algorithm internally. The end systems then attempt to access the transmission medium according to the algorithm. Because the time when the access will take place is not predictable, but occurs randomly, this technique is also called a **stochastic** strategy. In contrast to the controlled access strategies, in this technique the access times to the medium are reduced, and better channel utilization is achieved. However, the drawback of it is the possibility of **collisions** that lead to a distortion of the transmitted data.

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/9RbGEz_voeQ

Aloha method

The first random access strategy was the Aloha method developed in the late 1960s/ early 1970s on the Hawaiian Islands. With this technique you had two frequencies available in order to communicate wirelessly between a computer on the main island and the other islands. Transmission occurred from the main island to the other islands on one frequency, which was not a problem because there was only one transmitter. The other frequency was used so the other islands could transmit to the main island. Because in this case several senders share one frequency, collisions can occur. One difficulty was also that you could not determine whether a data transmission took place.

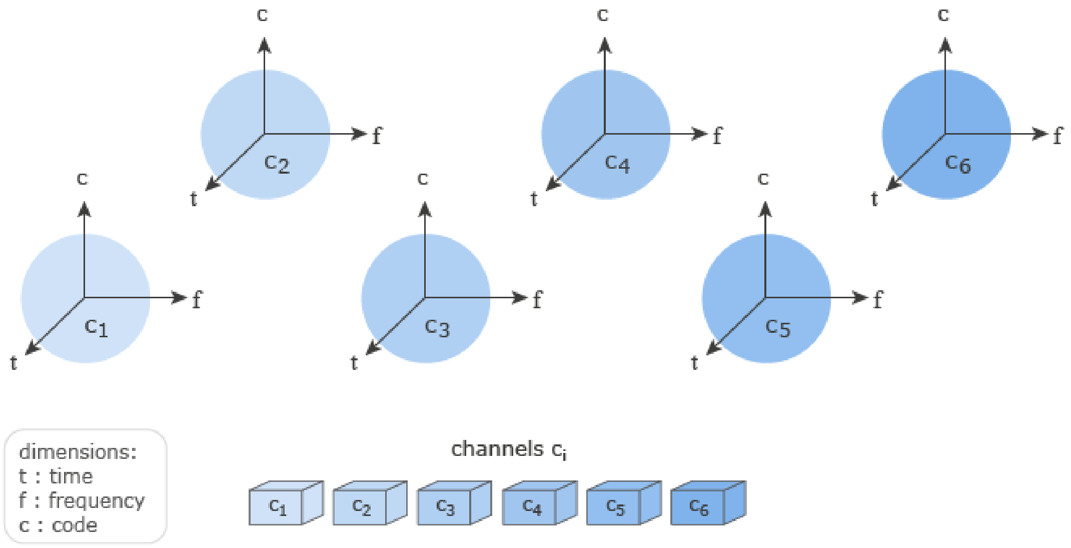
An improvement of Aloha is Slotted Aloha. In this case there is a general, predefined time-slot structure. The transmitter then either uses a time slot or lets it pass completely unused. This leads to an improvement of the situation with respect to possible collisions, so that this method (under certain assumptions; see TaWe12) works twice as good.

The idea of the wireless Aloha technique was taken up during the 1970s for fixed-line networks, which gave rise to the CSMA/CD technique and Ethernet technology (see Ethernet).

Later the CSMA/CD technique was adapted again for wireless networks, which resulted in the CSMA/CA technique. This is the basis for WLAN technology (see Wireless LAN).

3.1.3 Space Division Multiplexing

Space division multiplexing means that transmissions are separated spatially from each other. For wired systems this just means that communication occurs in parallel in different cables. For example, each end system can be connected to a switch with its own cable.



Space division multiplexing in wireless transmission

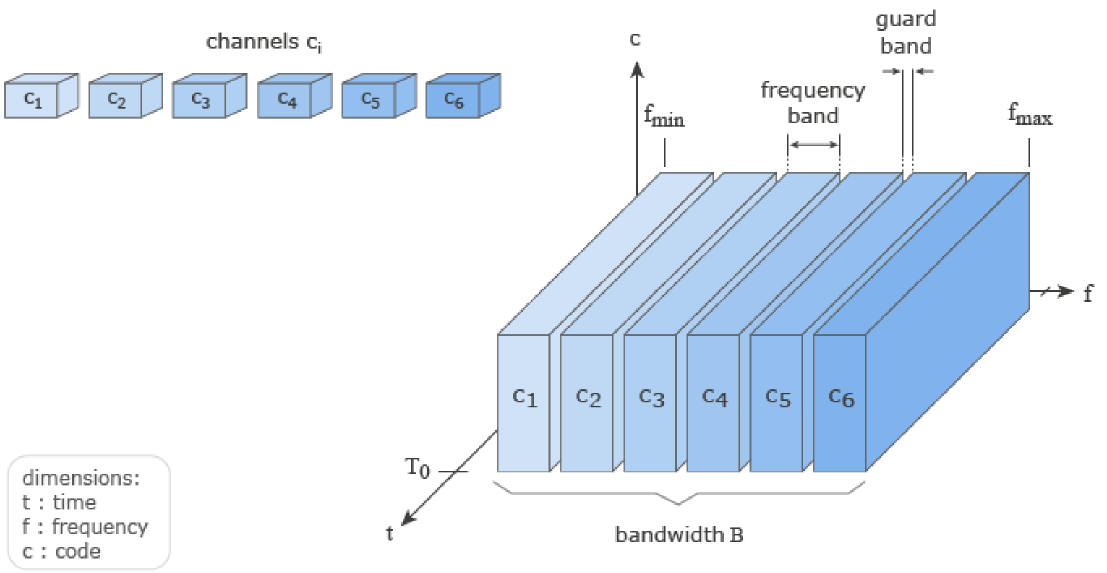
In wireless transmission, there is often the problem that you only have a limited number of frequencies available. For example, you would like to configure mobile telecommunications cells by setting up base stations. The cells should provide mobile telecommunications for a larger urban area. However, the number of base stations required is much larger than the number of available frequencies. Therefore, it is necessary to use the same frequencies in different cells. This is possible if the cells are far away from each other. Then parallel communication in these cells is possible without causing interference. In the figure you find six locations where transmissions can take place independently from each other due to the separation in space.

A related problem also occurs in WLAN planning for buildings. Especially in the use of the 2.4 GHz band, you only have three overlap-free frequency bands available. As a consequence, if you want to install more than three access points for the 2.4 GHz band, then is is necessary to reuse frequencies and to place the respective access points at a certain minimum distance.

An additional possibility to separate transmission in space is by using directed antennas.

3.1.4 Frequency Division Multiplexing

When using **frequency division multiplexing** each system is given a fixed allocation of a specific frequency range from the entire available frequency band of bandwidth B (f_{min} to f_{max}) for the entire transmission period (see figure). These individual frequency ranges are referred to as channels. Between these channels, there is a gap for easy separation of the individual frequency ranges with appropriate filters.



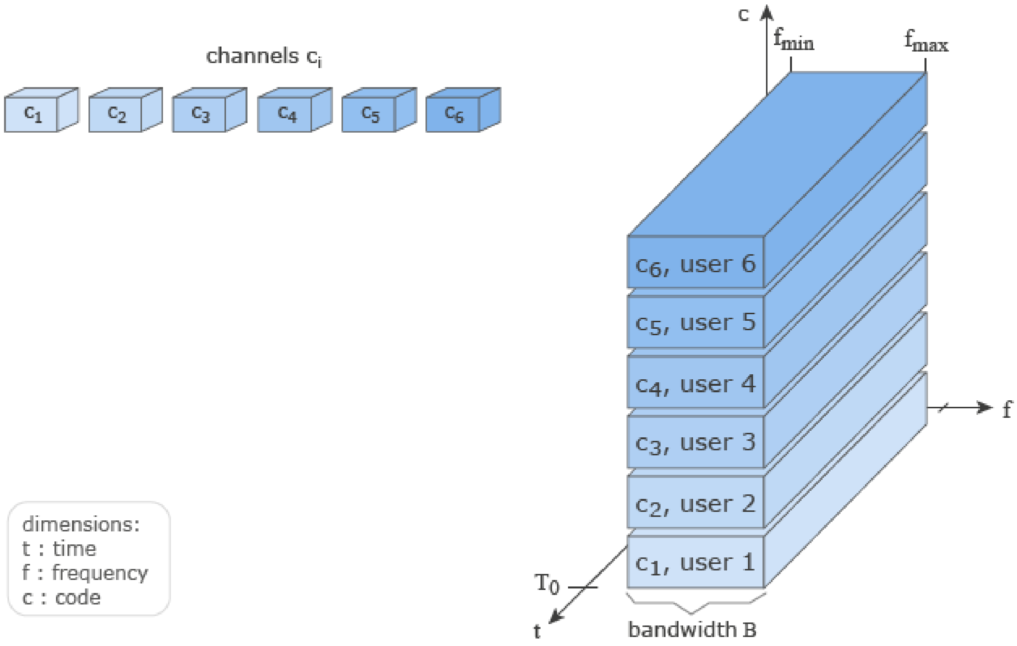
Frequency division multiplexing with six channels

Frequency division multiplexing is used when transmitting radio or television programs. You can see this also in the choice of some radio station names, since some radio stations sometimes put their frequency as part of their name (e.g., "Radio 90.3" refers to frequency 90.3 MHz). Frequency division multiplexing is well suited for this purpose because the programs are sent all the time and with approximately constant bandwidth needs.

Frequency division and time division multiplexing are combined in the second (GSM) and fourth generation (LTE) of mobile telecommunications. This means that terminals may use certain frequencies for a specified time.

3.1.5 Code Division Multiplexing

The final multiplexing option is **code division multiplexing**, which is the main technique used in the third generation of mobile telecommunications (UMTS). With this technique all stations can use the entire available bandwidth of the transmission medium all the time.



Code-division multiplexing with six codes

This is initially surprising because the stations can transmit simultaneously on identical frequencies. But it is possible to make a distinction between them on the basis of codes that are assigned to the individual transmitters. A code consists, for example, of 32 bits. If a transmitter would like to transmit a logical 1, it transmits the code sequence instead; for a logical 0 it transmits the inverse of the code. You can set up receivers so that the received signal is evaluated according to the transmitter code. By using this technique, the signals sent simultaneously by others become a kind of background noise, and correct evaluation still remains possible. Only if the background noise is too strong, then the technique does no longer work.

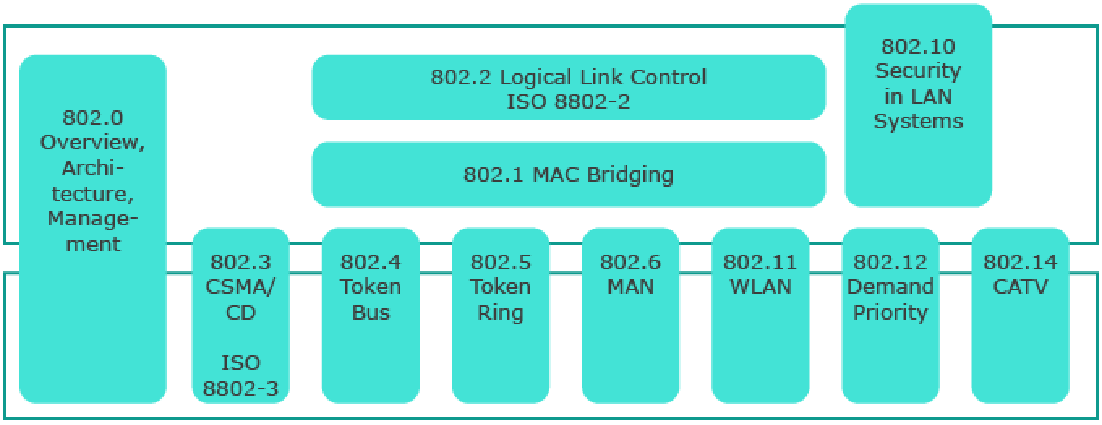
The technique can also be explained with an analogy. Imagine that you are standing with a group of other people on a crowded train platform. You are speaking English with the group that is nearest to you, while another group near you is speaking German and another is speaking Chinese. If you pay attention to the English conversation, the other conversations become a kind of background noise. Now let us assume you know German but not Chinese. In this case the Chinese conversation actually is background noise for you, but you can tune into the German conversation and evaluate what you hear as German words. You can imagine the evaluation by receivers the same way.

annotation

> Code division multiplexing comes from the military and offers the advantage that many frequencies can be used at the same time, whereby transmissions are made on each frequency with a low amplitude. In conventional transmissions, however, the transmissions are only made on a few frequencies with high amplitude. So for code division multiplexing, it may happen that you cannot recognize whether a communication is occurring because it is considered background noise. This is also known as spread spectrum.

3.2 IEEE Working Group 802

For local area networks, the specifications established within the IEEE 802 working group are very important. As already explained in the first chapter, IEEE is an organization of companies from the electrical engineering sector that agree on industry standards. The working group, whose name actually refers to its founding in February 1980, distinguishes between two sub-layers in the Data Link Layer. The lower layer (Layer 2A) is called **Medium Access Control** because a technique for medium access control is established here. The additional tasks in this sub-layer are the specification of frame formats, addressing, error recognition and correction. Layer 2B above is called **Logical Link Control** and serves as a uniform interface to higher layers. This layer thus abstracts from the fact that different technologies can be found below. It also defines three service types: unconfirmed and connectionless (this is the typical service), confirmed and connection-oriented, confirmed and connectionless.



IEEE 802 standardization groups

The figure shows the structure of IEEE 802. You can see that the general parts such as overview, architecture and management as well as the Logical Link Control apply to all technologies. On the other hand, for each technology it is determined how it should operate on the Physical Layer as well as the MAC Layer.

Here is a selection of interesting working sub-groups:

- 802.3: The Ethernet technology for wired local networks is being further developed in this very active working group. It is named after the media access control method CSMA/CD.
- 802.4/802.5: These working groups were established as competitors to the 802.3 working group because Token Bus and Token Ring are also standards for wired local networks. Because both techniques have been forced out of the market, these groups are no longer active.
- 802.11: The Wireless LAN technology for wireless local networks is standardized in this working group. This technology had no competition from the beginning (a competitor named HiperLAN was not developed into marketable products).
- 802.15.1: This group works on Bluetooth (the standard for wireless personal area networks). The work is, however, being continued now in an industry group established for this purpose (the Bluetooth Special Interest Group).

3.3 Ethernet

arrangement

Ethernet is the dominant technology for wired local networks. For usual office buildings there are no longer alternatives on the market. The original Ethernet specification had two main points. First, it established a relatively simple frame format and second, it specified the media access technique CSMA/CD for a bus topology. While the frame format has been preserved during evolution, the access technique CSMA/CD became less important. The reason for this has to do with the development of switches and the use of full duplex transmission. The latest standards with bit rates higher than 10 Gbit/s also no longer specify any variants with CSMA/CD and half-duplex.

3.3.1 Ethernet Frames

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/WgcQKsHWQn4

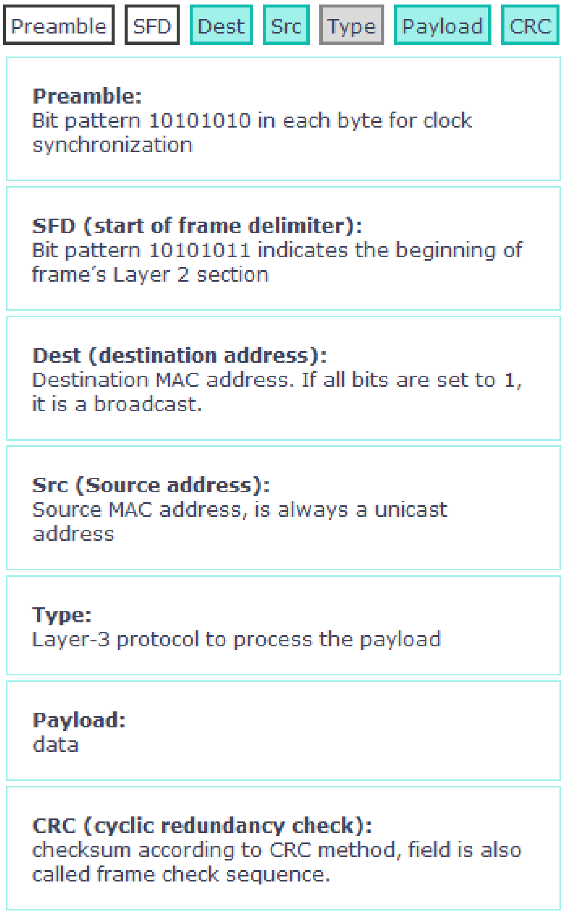
Ethernet frames

Due to the history of its development, there are two formats for Ethernet frames: The frame format according to **Ethernet II**, which was defined by the vendors Digital Equipment, Intel and Xerox (also called DIX Ethernet), and the **IEEE 802.3 Ethernet** frame format. The Ethernet II format is used primarily. It is depicted in the following figure.

In the online version an click interaction is shown here.

Structure of an Ethernet II frame (click on the boxes)

Begin printversion



Preamble:
Bit pattern 10101010 in each byte for clock synchronization

SFD (start of frame delimiter):
Bit pattern 10101011 indicates the beginning of frame´s Layer 2 section

Dest (destination address):
Destination MAC address. If all bits are set to 1, it is a broadcast.

Src (Source address):
Source MAC address, is always a unicast address

Type:
Layer-3 protocol to process the payload

Payload:
data

CRC (cyclic redundancy check):
checksum according to CRC method, field is also called frame check sequence.

End printversion

In the beginning of the frame, two fields belong to Layer 1. The 7-byte preamble always includes the bit pattern 10101010. This is followed by the start-of-frame delimiter field with the bit pattern 10101011. This initial sequence is used for synchronization. Because the receiver already knows which bits will be arriving, it can adjust its internal clock so that it can reliably recognize subsequent unknown bits.

The addresses in the frames are named as **MAC addresses** or as Ethernet addresses, physical addresses or hardware addresses. They are always 6 bytes long. The first 3 bytes represent a company code (**OUI**, organizationally unique identifier), which is assigned by IEEE to the vendor ("**OUI listing**"). The remaining 3 bytes of the address are assigned by the vendors themselves for each Ethernet card. In doing so, the vendor must ensure that each card really gets a unique address.

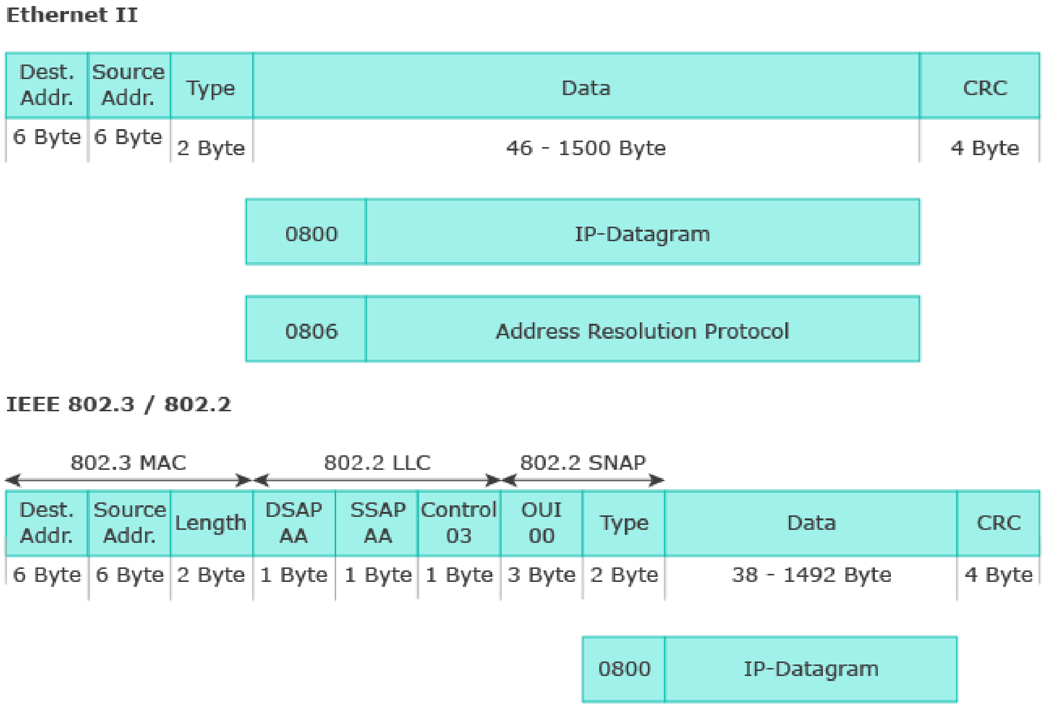
The first bit of the address indicates whether it is a unicast address (0) or a broadcast/ multicast address (1). The source address is always a unicast address. The second bit indicates whether the address is used globally (0) or locally (1). Purchased Ethernet cards always have a global unique address. Ethernet addresses can be overwritten if they are only to be used locally.

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/f2_lGr109Cg

MAC addresses

The essential difference between the Ethernet II and the IEEE 802.3 format can be seen in the following figure. In the Ethernet II format, the destination and source address are followed by the **type field**, which indicates the protocol on the next higher layer, i.e., the Network Layer. This information is needed in order to know how the contents of the payload data field should be evaluated. The type field is specified in a hexadecimal form and is always larger than 0x0600, which corresponds to a decimal value of 1536. For example, the value 0x0800 means that an IPv4 datagram is transmitted. 0x0806 is used for the Address Resolution Protocol.



Ethernet II and IEEE 802.3

The Ethernet II format is not compatible with the OSI standards because among other things it does have an **LLC header**. Therefore it was redefined by IEEE to be OSI compliant, with the essential difference that the type field in the Ethernet II format is processed as a **length field**.

The additionally contained LLC header consists of 3 bytes: DSAP (destination service access point), SSAP (source service access point) and a control field. There are two difficulties with this format: The data to be transmitted are no longer aligned with straight word boundaries, which has negative effects on the processing speed, and the type field is no longer present. For these reasons an additional header with a length of 5 bytes was introduced (**SNAP header**, subnetwork access protocol). The first 3 bytes were intended for the OUI code; the two subsequent bytes indicate the type, which gets the same value as the type field in the Ethernet II format. DSAP and SSAP always have the value 'AA': this means that a SNAP header follows the LLC header. The control field has the value '03' and the first 3 bytes of the SNAP header in general have the value '00'.

This IEEE 802.3 standardization of the Ethernet frame also results in the fact that 8 data bytes less can be transmitted, which means a maximum of 1492 bytes compared with 1500 bytes in the Ethernet II format. The maximum size of user data is also called the **maximum transmission unit** (MTU).

How can you determine whether an Ethernet frame has to be processed as Ethernet II or IEEE 802.3 Ethernet frame? If you find values greater or equal to 1536 (0x0600) in the length field, it must be an Ethernet II frame. For values up to 1500, it is an IEEE 802.3 Ethernet frame, and the number is regarded as a frame length. The values between (1501 to 1535) are not defined.

For Ethernet-based end systems, the rule is that they have to send and understand the Ethernet II format, that they should understand IEEE 802.3 Ethernet, and that they do not need to send IEEE 802.3 Ethernet frames. This is why you hardly find IEEE 802.3 Ethernet frames in Ethernet networks today. As a consequence, the **Ethernet II** format must be supported without restrictions.

The Ethernet frames are protected by a checksum, which is called the FCS (frame check sequence). A Cyclic Redundancy Check with 32 bit length (**CRC-32**) is used as algorithm for it.

3.3.2 CSMA/CD

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/zoaIEF1GtqQ

CSMA/CD method

The essential point of the original Ethernet specification was the media access technique CSMA/CD. All end systems were connected with each other via a bus topology where each system used the algorithm internally.

In the abbreviation the MA stands for **multiple access**, which means that all connected end systems are in competition to access the transmission medium (the bus).

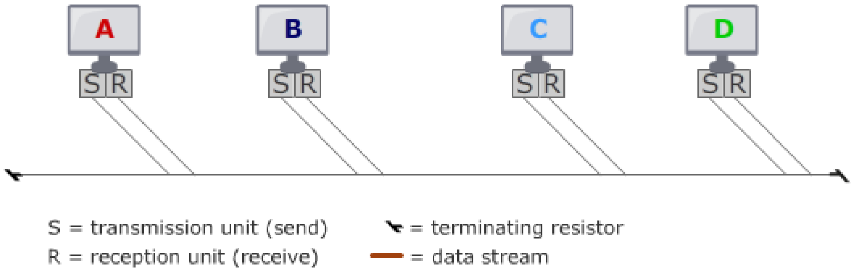
The starting point for the algorithm is an end system that would like to send a data unit. The end system checks before starting to send whether a transmission is already ongoing on the bus. This check is called **listen before talk** or **carrier sense**, where carrier is another name for medium. There can be two situations here. Either the medium is already free, or a transmission is ongoing. If a transmission is ongoing, the system waits until the medium becomes free. If the medium is free, the transmission starts immediately.

During transmission the end system still listens to the medium and compares the signals on the bus with the signals that the end system itself sends. This is called **listen while talk**. If there are deviations, the signals must come from (at least) one other end system that has begun transmitting at approximately the same time. This means a collision has taken place and has been recognized in this way. That is why the abbreviation contains CD, which means **collision detection**. In this case a special jamming signal is sent for safety's sake so all participating end systems take note of the collision. The transmission is aborted, and the end system will attempt to transmit the data again at a later time. If no collision is recognized during the entire transmission period, it is assumed that the frame arrived successfully at the receiver.

In the online version an animation is shown here.

CSMA/CD procedure

Begin printversion



End system A wants to send a data unit to end system D. For doing so, it first checks whether the medium is free. However, computer C checks the medium at the same time because it wants to send a data unit as well. Both of them determine in a completely correct manner that the medium is free and consequently start to transmit their data units.

Since the data units are provided to all end systems which are connected to the bus, also the transmitting end system receives its data unit via its reception unit. Now, as you have guessed, the data units of end system A and end system C collide. The data units overlap and consequently corrupted data are received by the connected end systems. In this situation, end system A recognizes that the received data is no longer identical to the transmitted data and therefore concludes that a collision has happened. It therefore immediately stops the transmission of the data unit. The same holds for end system C. End system A which has recognized the collision at first sends a jam signal, that is, a specific bit pattern to all other end systems to inform them about the collision. Due to the immediate transmission stops after detecting the collision, the medium therefore becomes available again after a relatively short time.

After a waiting time determined by the internal use of backoff algorithm, end system A repeats its transmission attempt. End system C wants to send as well, but it has internally determined a longer waiting time by the backoff algorithm. C then detects that the medium is in use and waits until it becomes free again. Therefore, A's transmission is successful this time. Every connected end system receives A's data unit and checks whether it is the destination. End system D which is the destination copies the data to process them internally.
End printversion

If a collision has occurred, then the question arises how retransmission attempts should be made. At this point you have to use different waiting times because at least two end systems were involved in the collision. The idea at this point is that the end systems select random times. With the so-called **backoff procedure**, you make the waiting times in addition dependent on how many collisions have previously occurred. The number of possibilities increases exponentially with the number of collisions. It is therefore very unlikely that collisions will continuously occur.

3.3.3 Switches

Current Ethernet networks are no longer based on bus topologies. Tree topologies composed of stars have been used for many years now. In the center of the stars, there are **switches**. Compared to older network components called hubs, switches have the advantage that they can forward data units in a targeted way. If, for example, a switch has 24 connection ports and an end system that is connected to port 5 is supposed to receive a data frame from another end system, the switch will forward the data frame only to port 5. This requires, however, that the switch already knows that the end system is there (see Learning Mechanism for Bridges/Switches).

Through this way of operation the switch allows for **parallel communication** between different ports. For example, it enables a frame to be received at port 1 and forwarded to port 21. At the same time, a frame can be received at port 7 and forwarded to port 4. It is also possible that the switch stores frames for an interim period, if, for example, two frames are received at two different ports, but should be forwarded to the same output port. As a side remark, it should be noted that there are switches with different internal performance capabilities, which means that not every kind of parallel communication between ports may be possible.

An important point in the development of switches was the change from half-duplex to **full duplex** at the ports. With half-duplex, problems occur at a connection port when a frame is simultaneously sent to the port and also sent by the switch via the port. Then a collision of frames occurs. However, if the connection is realized with full duplex operation mode, this problem no longer occurs.

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/OLL4_t-BjX0
Switches

The change of networks from bus topologies to tree topologies with switches and full duplex results in the fact that collisions can no longer occur on the medium. This eliminates the need for CSMA/CD, although it also does not hurt if it continues to be implemented. In this case a free medium will always be recognized, the transmission will start immediately, and it will never result in a collision.

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/LTdt7xqDKTs

Development LAN topologies

notice

> The reason why CSMA/CD is still presented here is because the CSMA/CA method for Wireless LAN is derived from it. It is easier to understand CSMA/CA if you are familiar with CSMA/CD.

3.3.4 Learning Mechanism for Bridges/Switches

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/HLpMoCRcHgY

Learning mechanism for bridges/switches

If you look at a local area network, which today is constructed based on switches, the question arises which steps are necessary to configure the network. You could imagine that you have to provide configuration information to the switches which end systems are connected to them.

But this is not the case because a learning mechanism was introduced for the switches (as well as with previous models, namely bridges). Here the switches learn which MAC addresses they can reach via which ports on the basis of the frames that they receive. It is important to note here that switches only learn from source MAC addresses and not from the destination MAC addresses. Each switches maintains a **MAC address table** (also called bridge table), which consists of the mapping of MAC addresses to ports. The table also records when the entry has been learned for the last time. Entries that are not updated from time to time are removed from the MAC address table in order to eliminate wrong mappings.

If a frame is sent to the switch, there can be three different situations.

- Destination address still unknown or broadcast: When a destination MAC address is not yet contained in the bridge table, or the frame is supposed to be sent to all devices in the local area network, **flooding** is used. This means that the switch forwards the frame to all ports (except the input port). The idea behind flooding for unknown destination addresses is that you definitely want the frame to arrive at the destination even if the frame is then forwarded to many places that it actually should not go to. As already mentioned, the correct port for the destination address still remains unknown afterwards as long as no response is sent because switches only learn from source MAC addresses.

- Destination address is known and can be reached via another port: The destination MAC address is contained in the table, which means it is known that it can be reached via a port different from the input port. The switch then sends the frame only to this port; this is called **forwarding**.
- Destination address is known and can be reached via input port: The destination MAC address is contained in the table; here the port via which the frame has been received is contained as the destination port. In this case the switch doesn't have to do anything except dropping the frame; this is called **filtering**. This may be surprising, but in a way the frame already previously reached its destination before it arrived at the switch.

3.3.5 Spanning Tree Protocol

If you build local networks on the basis of switches, then you also want to prepare for error situations in which, for example, a cable is damaged or a switch no longer works. You want to achieve fault tolerance so that the network will continue to function as good as possible in such scenarios. Therefore, redundant links are desirable.

However, at this point there is a problem with the way how switches operate. In a simple scenario there are switches A, B, and C. A and B, B and C, as well as A and C are connected with each other. If you assume now that an end system connected with A sends a broadcast frame, it will be forwarded by A to B and C. B and C in turn will forward the broadcast to the respective other output ports such that they mutually send the frame to each other. However, at this point it is not recognized that there are duplicates, and the switches send the frame in turn to A. A then sends the frame to B and C again and so on. This means the frames will circulate in the network all the time, and the network will be completely paralyzed for other communications.

The spanning tree protocol (IEEE 802.1d) was introduced to prevent this. As the name “tree” suggests, a logical tree structure is established. All areas of the network can still be reached via the tree structure, but only through unique and especially loop-free paths. Ports that have not been selected for the tree structure are logically deactivated but can be re-activated in case of an error.

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/TlFPPah-iBI

Spanning Tree Protocol

3.3.6 Ethernet Evolution

Since the beginning of the 1970s to today, a number of Ethernet variants have been developed. When increasing the bit rates, which initially took place in jumps by a factor of 10, backward compatibility was always respected. This means that you do not need to completely replace an existing network if an increase in the bit rate is to happen. You can initially only increase bit rates by implementing new variants where this is necessary (e.g., install new switches only in these areas), but other parts of the network can be left unchanged.

The notation for the variants first indicates the **transmission bit rate**. This is followed by word **base** for baseband (a variant with a broadband transmission where "broad" was put in this position was not successful on the market). At the end there is the **segment length** or the **type of cable**. In the following a brief overview of the Ethernet evolution is provided (for more details see Wikipedia).

10 Mbit/s Ethernet

The original Ethernet has the abbreviation 10Base5. 10 stands for the data rate 10 Mbit/s and 5 stands for the possible segment length of the bus, which was 500 meters. Using up to four repeaters, a maximum size of 2500 meters was possible in this system. 1 cm thick coaxial cable was used, which was difficult to install. A variant that used thinner coaxial cable was introduced with 10Base2, which was called Cheapernet. However, the segment length was reduced to about 200 m.

In 1990 two significant changes were made with the introduction of 10BaseT. The T stands for twisted pair, i.e., the wiring was changed from coaxial cables to twisted copper conductors, which still represent the usual network cable today (but there are different cable categories which have to be distinguished). In addition the topology was changed from a bus topology to a star topology.

The first fiber-optic variants were introduced with 10BaseF. This is more of a special case if you can no longer bridge a long distance with copper cable due to its attenuation properties.

100 Mbit/s Ethernet

There are essentially two variants of 100 Mbit/s Ethernet which is also called Fast Ethernet. A variant based on the twisted-pair cables and a fiber-optic variant.

Two innovations were introduced. First there is so-called auto-negotiation: When two devices are connected with each other via Ethernet, this means that they automatically negotiate the highest common bit rate as well as the half-duplex or full duplex mode. So in the configuration you do not have to consider which specifications the devices exactly have. The other innovation is flow control (802.3x): Here devices can signal with so-called pause frames that they are internally overloaded and that the other device should not send any more data units for the time being. At this point you can also see that flow control does not mean that a steady flow of data should be achieved (this exists elsewhere but is called traffic shaping).

1 Gbit/s Ethernet

Gigabit Ethernet specifies two twisted-pair cable variants and two fiber-optic variants that are relevant in practice.

There were significant problems with continuing to support the half-duplex mode at a bit rate of 1 Gigabit/s because possible collisions on the medium still had to be detected. To ensure this, the transmission of frames must continue until a collision can be detected in any case. With higher bit rates and constant minimum frame lengths (64 byte), the difficulty arises that the transmission is completed much faster. To prevent this, the minimum frame length had to be increased to 512 bytes. Despite certain countermeasures, this is extremely inefficient because in extreme cases 7/8 of the frame consists only of meaningless dummy bytes in order to reach the minimum frame length.

10 Gbit/s Ethernet

Therefore in the specification of 10 Gigabit/s Ethernet, a clear cut had to be made such that with this bit rate and with all higher bit rates half-duplex and CSMA/CD are no longer supported. The network must therefore be built up using star or tree topologies or point-to-point connections in such a way that collisions are no longer possible on the medium.

As just presented the Ethernet bit rate up to this point has always been increased by the factor of 10. In contrast, an increase factor of 4 was always used in backbone networks such that the bit rates there were 155 Mbit/s, 622 Mbit/s and 2.5 Gbit/s. With the next increase in backbone networks, the bit rates were approximately the same (not exactly the same because the backbone data rate of the SDH technology was 9.953 Gbit/s). Such a compatibility became interesting in the meantime when you wanted to link two locations (such as two data centers) with each other via the backbone network.

The relevance of Ethernet for wide area networks is reflected in the fact that for 10 Gbit/s there are two twisted-pair variants specified whereas for fiber-optic there are eight.

40 Gbit/s / 100 Gbit/s Ethernet

With the establishment of the next higher bit rate, it was not clear which increase factor should be used to continue (x4 or x10). Therefore as a compromise variants for both 40 Gbit/s and 100 Gbit/s were specified. With copper cables there are great difficulties with attenuation at these high bit rates, so these variants only have a reach up to 10 m. Otherwise, implementation has to be done with fiber-optic cables, which require the use of multiple wavelengths within the cable for larger distances.

Variants with 200 Gbit/s and 400 Gbit/s are going to be specified in 2017. The standardization of 1 Tbit/s is planned for 2020 or later.

3.3.7 Summary - Ethernet

We may sum up the most important properties of Ethernet from today's perspective again.

- Ethernet is the dominant technology for wired local networks. It also plays an important role in the MAN and WAN areas with more recent variants.
- In wired local networks, Ethernet is constructed as a tree topology with switches in the center of the stars. Transmission is full duplex. This means collisions can no longer occur in the network and the CSMA/CD method is no longer relevant.
- The Ethernet II format is used almost exclusively.
- The addresses used are MAC addresses, which are 6 bytes long.
- Up to 1500 bytes of payload data can be transmitted.
- With the Ethernet header and trailer, the data are extended by 18 bytes. An Ethernet frame can have a maximum length of 1518 bytes.
- The checksum, which is calculated according to the CRC 32 method, provides good error detection. In case of errors, the frames must be retransmitted. Due to the low bit error rates in today's wired networks, this happens very rarely.

3.4 Wireless LAN

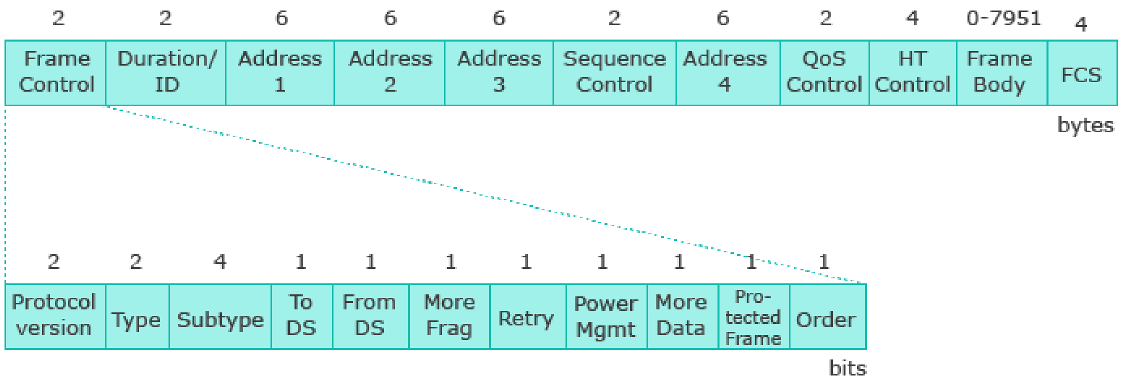
arrangement

Similar to Ethernet, Wireless LAN is specified by IEEE, which is done in the 802.11 working group. This section briefly presents the considerably more complicated frame format before it looks at the adaptation of CSMA/CD. This adaptation is necessary due to different properties of wireless transmission. In the Wireless LAN specification two scenarios are defined.

- In **infrastructure networks**, the wireless network (here also referred to as basic service set) is created by an access point and end systems that communicate with it in a wireless manner. The access point selects a channel (i.e., a specific frequency range), and each terminal must use this channel in order to participate at the basic service set. In case of simultaneous transmissions in this network, there are collisions, so that time-division multiplexing is required. The infrastructure network also comprises an interconnection of access points and the transition to other networks. This additional networking is almost always implemented with wires.
- Another possibility is a so-called **ad hoc network**, which can be configured in the short term. Here the end systems form a wireless network, which is independent from an infrastructure, especially independent from access points. None of the end systems has a special role here; all have equal rights. Because this mode is rarely used in WLAN, it is not considered further here. In contrast, Bluetooth only knows an ad hoc mode.

3.4.1 WLAN Frames

The Wireless LAN frame format is considerably more complicated than the Ethernet frame format, as can be seen in the figure.



Frame format for wireless LAN

At this point the details are not discussed so only a few selected fields are explained (see section in Wikipedia for more details).

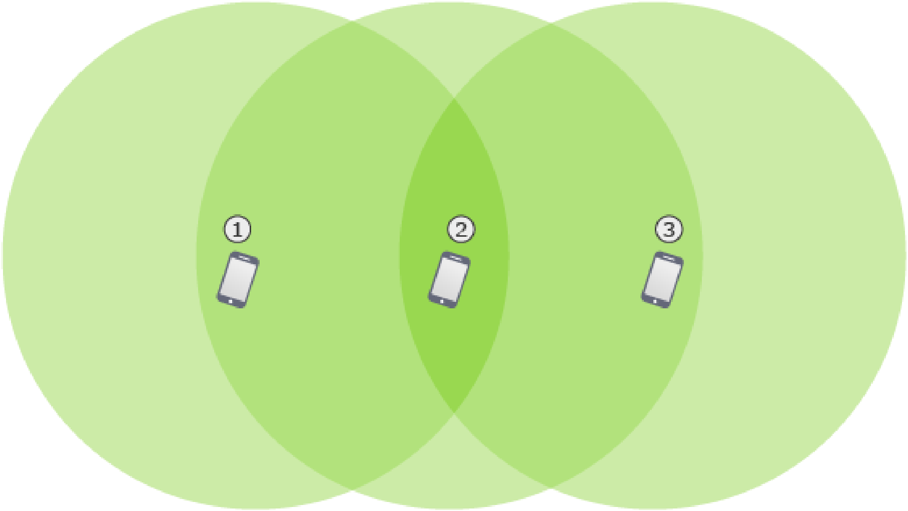
- Type/subtype: These fields indicate what kind of frame it is. A distinction can be made in particular between usual data frames and control frames.
- ToDS/FromDS and address fields: There are a total of up to four address fields for WLAN fames. This has to do with the fact that pieces of information for forwarding are also included. These pieces of information are the MAC addresses of access points that are used as intermediate systems. The exact use is defined by the ToDS and FromDS bits. DS stands for distribution system. This term refers to a distribution infrastructure between the access points, which is usually implemented as a wired network.
- PowerMgmt: In Wireless LAN there are mobile end systems that are powered by batteries. This means that achieving long battery use is an interesting goal. So energy saving mechanisms are implemented in WLAN so that the devices can sometimes save energy. This can be achieved by informing the end system that no data will be sent to it in the near future. The access point, which is constantly up, stores the data frames temporarily if needed so the end system can from time to time enter a sleep state. This means the energy consumption of the transmitting/ receiving unit in the device is lowered. This is controlled among other things with this bit.
- WEP: The abbreviation of this bit stands for wired equivalent privacy, i.e., it is an encryption method specified for WLAN. When using WLAN all devices in the area can read the frames transmitted by other devices. This means the encryption of data is essential, so WLAN established the WEP standard for this purpose. However, it had significant weaknesses, so it was first replaced by WPA and relatively soon afterwards by WPA2.

- Duration: An intended transmission time can be indicated in the duration field. This is relevant for the RTS/CTS mechanism (see CSMA/CA with RTS/CTS).
- CRC: Similar to Ethernet, there is a checksum at the end in accordance with the CRC method.

3.4.2 Challenges of Wireless Transmission

When transmitting frames in wireless networks, there are situations that do not exist in wired networks. Therefore the CSMA/CD method cannot be used without modifications.

One problem here is called the **hidden terminal problem**, which is illustrated in the figure.



Three stations in a wireless LAN with their transmission areas

Station 2 can receive all signals from neighboring stations 1 and 3, but stations 1 and 3 cannot communicate directly with each other because of the limited range of the emitted signal. There could be a situation that station 1 is already transmitting a data frame to station 2. During this transmission station 3 listens to the medium because station 3 would also like to transmit a frame to station 2. However, station 3 cannot detect the use of the medium, so it also begins to transmit. This leads to a collision at station B, so neither data frame can be evaluated properly by B.

With regard to CSMA/CD, you could say that the carrier sense (CS) failed in this case. A free medium was detected although it was in use. Collision detection (CD) also does not work in the same way here as in wired networks. In the bus topology the transmitter can detect that it has caused a collision because it can compare its own signal with the signals on the medium. If there is a deviation, a collision has occurred. However, in wireless networks this is not the case, which means collisions are not detected by the transmitter during the transmission. This implies that the frame will in any case be sent until the end. In addition, it is usually the case in wireless networks that an antenna is used at a specific time either as a transmitting or receiving antenna. This means that while transmitting via the antenna no signal can be received from the other participant anyway. As a consequence, a collision can only be detected indirectly via the receiver. It should confirm the successful reception.

3.4.3 CSMA/CA

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/

CSMA/CA method

Video to be added!

Due to the different characteristics of wireless transmissions, CSMA/CD was modified for Wireless LAN and is called CSMA/CA in this context. The end of the abbreviation stands for **collision avoidance**. The aim was to try to avoid collisions as much as possible through the specifications of this method. There are two modes: the **distribution coordination function** (DCF) and the point coordination function (PCF). Only DCF is discussed here because this mode must be implemented by all end systems. PCF in contrast is optional and is almost never supported in reality.

In general, it should be considered that WLAN always performs carrier sense. Because of the hidden terminal problem, it can incorrectly indicate a free medium, but its indication is correct in many situations.

The basic idea in CSMA/CA is to specify different waiting times that reflect different priorities for media access. When a terminal is permitted to use a shorter waiting time than other terminals, it has a higher priority. Three waiting times have been defined (at the beginning of standardization):

- SIFS: The short interframe space is the shortest waiting time and represents the highest priority. Devices only need to wait for this time in particular if they want to send control messages.
- PIFS: The PCF interframe space is the medium waiting time. It is only used for the PCF mode.

- DIFS: The DCF interframe space is the longest waiting time. This time must always be waited for normal data frames. An additional random waiting time may also be added. This depends on whether the medium was previously free or occupied. If an end system would like to transmit, and the medium was already free when it began to listen to the medium, only the DIFS period is waited. If the medium was previously occupied and just becomes free, then DIFS plus an additional random time must be waited. The reason for the difference lies in the consideration of how likely it is that a collision will occur. If the medium was previously free, then there is probably currently not so much need to use the medium. If the medium was previously occupied, then it may be the case that not only one device but several others are also waiting to transmit. If these only waited for DIFS to pass, then there would be collisions, which you often avoid by the addition of a random waiting time (a collision occurs only if both select an identical waiting time). You can also look at it this way: you proceed here as if a collision had occurred in CSMA/CD.

Within CSMA/CA it is essential that unicast frames are always acknowledged. This works so that the receiver immediately tries to acknowledge the frame reception and must only wait the SIFS time for this special acknowledgment frame. Such acknowledgment frames do not exist within CSMA/CD. One reason for this is that with wireless transmission the transmitter cannot determine on its own that a collision has happened. Therefore, it can indirectly determine a collision by a missing ACK frame. However, there may be other reasons for a missing ACK frame such as a bit error. In addition, WLAN is in general significantly more susceptible to errors than wired networks because also other WLANs, Bluetooth or microwave ovens use its frequencies (this is particularly true for the 2.4 GHz band).

After collisions a backoff algorithm is used by CSMA/CA similar to CSMA/CD.

3.4.4 CSMA/CA with RTS/CTS

The basic CSMA/CA method can be enhanced with the optional RTS/CTS mechanism. RTS (**request to send**) and CTS (**clear to send**) are short control frames for reserving the medium.

A transmitter who wants to send a data frame initially transmits an RTS control frame to the receiver. The receiver indicates its readiness to receive frames with the CTS control message; here only the SIFS period must be waited. The transmitter begins with the transmission of the data frame after receiving this control frame and waiting the SIFS time. The receiver confirms a successful receipt with an acknowledgment frame (again only the SIFS time is waited). The advantage of this mechanism is that the RTS/CTS frames always contain time periods for the reservation of the medium. Other devices store these time periods in the so-called **network allocation vector** and do not access the medium for this time even if they cannot observe the data frame transmission. As explained before this holds for the hidden terminal problem. The RTS/CTS mechanism thereby improves the handling of the hidden terminal problem considerably. In particular it becomes very unlikely that a collision will occur during the transmission of data frames.

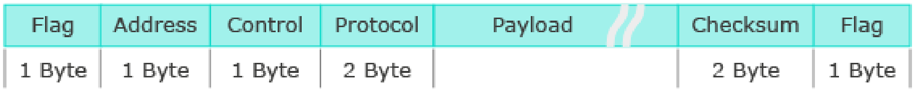
However, collisions are still possible especially for the control frames. This likelihood is, however, not so high because they only occupy the medium for a short time. The damage is also not as important as for the collision of data frames.

The University of Innsbruck offers an animation for trying out CSMA/CA with RTS/CTS. You can choose here between a situation with or without a hidden terminal.

3.5 Point-to-Point Protocol

The **Point-to-Point Protocol** (PPP) is mentioned here as another Data Link Layer protocol. Its primary use is as a Layer 2 protocol between private households, more specific DSL modems, and the respective Internet service provider.

The general purpose of this protocol is to transmit data via point-to-point connections. The packets from the Layer 3 protocol are put in PPP frames with an appropriate protocol type indication. A PPP frame has the following structure:



PPP frame format

The format is based on the HDLC frame. HDLC is an older Layer 2 protocol from which many other protocols have been derived (for example, in mobile telecommunications). As in HDLC, there is a start and end flag. This is a name for the bit sequence 01111110, which marks the beginning and end of the frame. The subsequent field for the address is filled with 1111 1111, but it has no meaning because PPP does not support addressing. The control field is also not relevant and is always filled with 0000 0011. However, the protocol field is more interesting. Similar to the type field in the Ethernet frame, it denotes which Layer 3 protocol is used, e.g., IP or NetBIOS. Then the actual data follows as the payload, which has a length of up to 1500 bytes. The checksum, which is calculated again according to the CRC method (but with 16 bits), serves for error detection.

PPP is used for synchronous and asynchronous data transmission as well as for modem connections and dedicated lines. It is independent from the physical interface.

When connecting to an ISP from home, PPP includes several phases:

- Connection setup and negotiation of communication parameters via **LCP**.
- Configuration of the Internet protocol via **NCP**.
- Use of the connection
- Disconnect

Parameters for communication are exchanged with the **Link Control Protocol (LCP)**, such as the maximum length of the data frame to be transmitted (default is 1500 bytes), an optional authentication protocol (PAP or CHAP) and parameters for monitoring the connection quality.

The parameters necessary for the operation of the Layer 3 protocol (generally IP) are transmitted by the **Network Control Protocol (NCP)**. When IP is used, the parameters are transmitted by the **Internet Protocol Control Protocol (IPCP)** which realizes NCP in this case. IPCP transmits the IP address allocated by the provider, the default router (default gateway) and the IP address of the DNS server. The participant can use the Internet only after these parameters have been provided. NCP has a task similar to DHCP (see DHCP - Dynamic Host Configuration Protocol), which is used in LANs.

3.6 Error Detection and Error Correction

arrangement

The most important task of Layer 2 in addition to medium access control is the protection of entire **frames** against faulty transmission.

Bit errors in the transmission must be detected with appropriate methods. For this purpose **check bits** are added to the frame, which are transmitted along with the data and enable verification of the correct transmission on the receiver side. Parity bits for this purpose are discussed briefly in this section, but the cyclic redundancy check method is presented in a detailed manner.

However, possible transmission errors not only need to be detected, they also need to be corrected. A distinction is made between backward and forward error correction. When using **backward error correction**, frames are retransmitted in case of bit errors. When using **forward error correction**, a frame contains enough redundant information so that the reconstruction of the right bit sequence is possible without retransmission.

What is important in this context is also how such techniques are implemented in the data transfer protocols.

3.6.1 Bit Error Rates for Media

As you can easily imagine, the effort that has to be spent to increase transmission safety depends on both the requirements of a given application as well as on the quality of the transmission channel. While a **bit error ratio** of 10^{-2} is still tolerable for voice transmission, it should be smaller than 10^{-7} if possible for data transmission. Without additional measures, the following values arise for the listed transmission media:

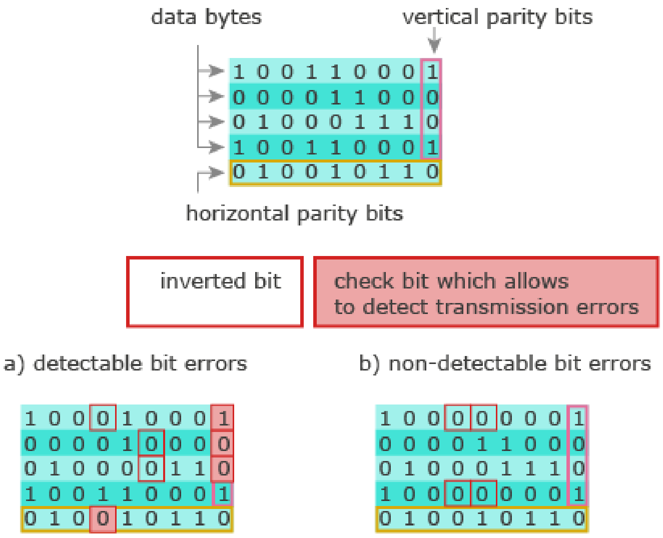
- Bit error ratio $< 10^{-6}$ to 10^{-7} for digital telephone networks,
- Bit error ratio $< 10^{-9}$ with coaxial cables in the local area,
- Bit error ratio $< 10^{-12}$ for fiber-optic cables

The bit error rate (BER) refers to the number of errors in a unit of time, typically per second.

3.6.2 Transverse or Longitudinal Redundancy Check

The simplest check method is to attach a bit to the data unit to be protected. The value of the bit is determined so that the entire data block (data + check bit) should always contain an odd number (**odd parity**) or an even number (**even parity**) of bits with the value “1.” This method is very simple, but it has the disadvantage that an even number of bit errors cannot be detected. In such a case there is no difference in the result when checking for the even or odd number of ones.

An improvement of the method in based on a two-dimensional representation of the data to be transferred together with transverse and longitudinal parity checks. The difference between the transverse and longitudinal parity checks consists in the fact that the parity bit is determined either from the rows or the columns of the data fields to be protected. The following figure contains an example of the simultaneous use of both checks:



Redundancy check with parity bits

Some detectable bit errors are shown in figure a). The bit errors in the 1st byte can be located precisely and can be corrected. The bit errors in the 2nd and 3rd bytes can only be detected by the transverse parity, which means they cannot be located precisely.

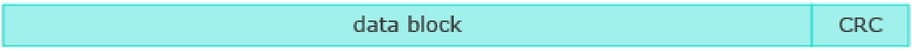
Figure b) shows bit errors, which cannot be detected.

The transverse redundancy check is often used in the transmission of single characters, such as in the **ASCII code**, where the 7-bit code is extended by a length of one byte with a parity bit. The longitudinal redundancy check is preferably used to secure entire data blocks.

3.6.3 Cyclic Redundancy Check

arrangement

The cyclic redundancy check is considerably more efficient than parity checks, which only raises the bit error ratio by 2 to 4 orders of magnitude. With this technique, the serial data stream to be transmitted is first divided into blocks whose length should be as large as possible for efficiency reasons. As previously mentioned, Ethernet frames can contain up to 1500 bytes of payload data. The checksum is calculated based on the Ethernet header and the payload data.



Block formation with cyclic redundancy check

In general, a checksum (FCS, **frame check sequence**) is derived from the frame fields with the help of certain algorithms. The length of the checksum depends on the algorithm used. The checksum is transmitted along with the other frame fields.

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/w_TUHoA1vAo
Cyclic redundancy check

3.6.3.1 Polynomial Representation

In the following sections the CRC method is presented. Before we discuss it in more detail, it is useful to become familiar with a special representation of bit sequences for this method. A bit sequence is represented here as a polynomial on the basis of x.

For example, you can represent the bit sequence 1001101 as a **polynomial** in the following way:

$1 * x^6 + 0 * x^5 + 0 * x^4 + 1 * x^3 + 1 * x^2 + 0 * x^1 + 1 * x^0$

Using usual calculation rules, this can be abbreviated to:

$x^6 + x^3 + x^2 + x^0$

This polynomial representation is used for the data to be transmitted as well for the bit sequence called the **generator polynomial**, which is used for calculating the checksum.

In addition, the so-called **modulo 2 arithmetic** or, in other words, the exclusive-or function is used for calculations in the CRC method. This means that carryovers are ignored in the binary system calculations. So 0 + 0 = 0 and 0 + 1 = 1 + 0 = 1 as usual, but 1 + 1 no longer equals 10 but 0. This way of calculating is advantageous for implementation.

3.6.3.2 Calculations on the Transmitter Side

The calculation on the transmitter side works as follows. A bit sequence that is to be transmitted is first converted to the polynomial representation. It is referred to as the **message polynomial M(x)**.

Then the **generator polynomial G(x)**, which is used for protecting each bit sequence and is known not only to the transmitter but also to the receiver, is considered. Its highest term is taken, and the message polynomial is multiplied by it. For example, if the generator polynomial is $x^3 + x^2 + x^0$, then the highest term is x^3 and the M(x) is multiplied by it. You can also take a look at this in the representation as bit sequence. This means then that one zero less than the number of bits that G(x) has is attached to M(x). In the example, the G(x) written as bit sequence is 1101, so that three zeros are attached to the message polynomial.

This $M(x) * x^r$ is divided by G(x), which results in a quotient and a remainder. Only the remainder R(x) is important. The **transfer polynomial T(x)** to be transmitted is obtained from the addition of $M(x) * x^r$ and R(x). T(x) is therefore constructed so that its division by G(x) has the result zero. This is used for the check on the receiver side.

3.6.3.3 Calculations on the Receiver Side

The receiver side gets a message with an attached checksum T'(x). You want to check here if the message has been received correctly. To determine this a calculation is performed similar to the one on the transmitter side. However, here it is not necessary to attach additional zeros. You directly divide T'(x) by G(x) and consider the remainder.

There can be two cases.

- R(x) = 0: There is no remainder after the division. In this case you can assume that there is no transmission error. There is a very high probability that this assumption holds (for details see CRC Accuracy).
- R(x) != 0: In this case one or more bit errors have occurred during transmission, but you cannot use this method to determine which bits are incorrect. So the frame has to be retransmitted.

3.6.3.4 CRC Accuracy

The CRC method is very reliable overall. A more accurate statement can be made regarding generator polynomials, which meet two relatively easy-to-fulfill conditions.

- G(x) must consist of more than two terms with a coefficient of 1.
- G(x) must be divisible by (x + 1).

Then the following applies:

- Single bit errors: Residual error probability = 0
- Double bit errors: Residual error probability = 0
- Odd-numbered bit errors: Residual error probability = 0
- Error burst b < r: Residual error probability = 0
- Error burst b < r: Residual error probability = 0.5^{r-2}
- Error burst b < r: Residual error probability = 0.5^{r-1}

Here b is the length of a block containing bit errors and r is the degree of the polynomial.

Extensive tests have been performed to determine which generator polynomial is most appropriate for bit error patterns that occur in reality. As a result there is a list of standardized polynomials that are intended for different purposes (see list on Wikipedia)

3.6.3.5 CRC Implementation

Although the CRC method may initially seem relatively complicated, it can be implemented relatively easily in hardware with shift registers. Shift registers are small memory elements that record a bit and in the next cycle transmit the value of the bit again and receive a new value as input.

In practice to check whether the remainder is zero has a disadvantage, namely that zeros additionally added to the transmission cannot be detected. So in implementation the storage elements in the shift register are not initialized with zeros but with ones. As a consequence, a special results pattern becomes the correct checksum, and you can also reliably detect additional zeros.

3.6.4 Backward Error Correction

With the CRC method or other techniques, you can determine that a bit sequence has not been received correctly. This is recognized on the receiver side, but the transmitter ultimately has to retransmit the data. You can imagine that the receiver could transmit a negative acknowledgment and thereby informs the transmitter about the error. But this is usually not done in practice. Instead, just positive feedback is provided. The transmitter waits for positive feedback, and if it does not arrive, it retransmits the data.

Negative acknowledgments would have the advantage of faster error correction, but the following considerations speak against them: As it is discussed in the Protocol Mechanisms section, you have to compensate for frame loss, too. Frames can not only have bit errors, they can also be lost completely. In the latter case the receiver cannot send feedback because it has not received anything. If you do not use negative acknowledgments, then the bit error case is handled similar to frame loss. When frame loss occurs, the transmitter has to respond to the missing acknowledgment and retransmit the frame. So two situations have been merged. An additional but not so important reason why negative acknowledgements are not used is that there could be a bit error in the transmitter's address. A negative acknowledgment would not reach the transmitter in this case.

3.6.5 Forward Error Correction

In addition to backward error correction, there is also the option to use error-correcting codes. Here so much **redundancy** is added to the data that the receiver can not only detect a certain number of errors but also correct them on its own. Because the retransmission of data is not necessary, this is also called forward error correction.

A simple example of such a code is when you transmit ones and zeros three times each. This means a 1 becomes 111 and a 0 becomes 000, so only 111 and 000 are valid codes. If the receiver receives the code 101, then it performs a majority vote and corrects the code to 111 and thereby detects a 1. The idea here is that it is more probable that only a single bit arrived incorrectly than that two bits were altered. But you can also see that this correction may be flawed.

More complex error-correcting codes are among others the Reed-Solomon codes.

3.6.6 Protocol Mechanisms

arrangement

When two end systems communicate with each other, you want to ensure that all transmitted data arrives at the receiver. But the data should also not arrive multiple times as duplicates, and the correct sequence of data has to be maintained. This is not ensured by the Physical Layer. It has to be realized by the Data Link Layer with the help of protocol mechanisms.

We can consider a situation here, for which the following assumptions hold. An end system A would like to transmit a data stream to an end system B. Within end system A, the Layer 2 implementation receives an appropriate set of payload data from Layer 3, i.e. it matches to the MTU. This Layer 2 instance adds control information to ensure safe transmission and passes the frame to Layer 1 for transmission. At Layer 1 errors are possible during transmission, i.e., frames can be transferred incorrectly, so they are dropped by the receiver's error detection at Layer 2, or the frames can be lost completely.

In the online version an video is shown here.

Link to video : http://www.youtube.com/embed/rfDNvS3QJyY
Data transfer protocols

3.6.6.1 Stop-and-Wait Protocol

The **stop-and-wait protocol** is a simple transmission mechanism. The basic principle in this protocol is that the receiver acknowledges each received data unit. The transmitter waits until it receives an acknowledgment and then continues with the transmission.

However, this simple principle cannot handle transmission errors. In this case no acknowledgment would arrive, and the transmission would come to a stop. **Timeouts** are used to prevent this. This means the transmitter has an expectation how long it may take until it receives an acknowledgment. If this acknowledgment does not arrive, an internal clock reaches a time limit, and it retransmits the data. To do this it must temporarily buffer the data internally.

When using this method, deadlocks no longer occur, but a new problem arises. You have to take into account that not only data units themselves but also the acknowledgment of the data unit can be lost. The transmitter cannot distinguish between these situations, i.e., from its perspective it is unclear whether the data unit or the acknowledgment was lost. In either case it retransmits the data unit. However, this presents a problem if the data unit previously arrived correctly. After being retransmitted it has now been received twice by the receiver. It is important to detect this. The data unit then has to be discarded internally by the receiver, but an acknowledgment must be sent to the transmitter so the data unit is not transmitted again. Assume for example that a text is transmitted where there are multiple sentences in each frame. If the receiver did not recognize this as a **duplicate**, the sentences would be doubled in the text.

Sequence numbers are required in order to prevent this. These are numbers that are inserted into the data units to make them distinguishable. Ascending numbers can simply be assigned to the data units, i.e., data unit 1, 2, 3, etc. In the present case of the stop-and-wait protocol, even the numbers 0 and 1 are enough.

You can see clearly with an example calculation that the stop-and-wait protocol is very **inefficient** for today’s networks. Assume the following realistic conditions: The signal propagation time between transmitter and receiver is 10 ms; the possible data rate is 1 Gbit/s (full duplex); the frame size is 1500 bytes; the stop-and-wait protocol is used, and there are no frame losses during the period observed. With some simplification (e.g., ignoring the processing time at the receiver), you can assume that 1500 bytes will be transmitted every 20 ms. The bit rate is therefore 1000/20 * 1500 * 8 bit/s = 600,000 bit/s. If you calculate the percentage of the bit rate of the medium that will be used, you can see that this percentage is only 600,000 bit/s / 1,000,000,000 bit/s = 0.06%. This means that the medium is hardly used, and most of the time is spent waiting for the acknowledgment from the other end.

3.6.6.2 Credit Method and Sliding Window Techniques

The basic idea to improve the efficiency of transmissions is to allow the transmitter to send a series of data units without the need to wait for an acknowledgment for the respective previous data unit.

One method is the **credit method**. Here a certain number of data frames are defined, which the transmitter may send without intermediate acknowledgments by the receiver. For example, there could be a credit of eight frames. This means the transmitter may send eight frames and then waits for an acknowledgment that indicates the correct reception of the eight frames. When using a full duplex channel, however, there are still times where the medium is not used.

Another improvement is the **sliding window technique**. Here the transmitter also receives a credit for a certain number of frames. When the receiver receives the frames, it already begins to acknowledge them. These acknowledgments result in an increase of credits for the transmitter so in the best-case-scenario it can continue to transmit all the time. For example, there could be a credit of eight frames at the beginning. If the transmitter sends five frames, the credit is lowered to three frames. However, if the first two frames are acknowledged, the credit rises again to five frames. This makes it possible that the credit will never drop to zero frames.

However, what is more difficult with this technique is handling frame losses such as when eight frames are sent in a sequence and the second frame is lost. For this case there are the simpler method called **Go-Back-N**, where all frames beginning with the first lost frame are retransmitted, and **selective repeat** method. With the latter method, only the frames that are actually lost are retransmitted, which is, however, more difficult to manage on the receiving end. Go-Back-N also has the disadvantage that in some cases frames that have already arrived correctly are retransmitted. You can look at an animation on this subject by the University of Innsbruck. You can click on the frames so they are lost and then see the reaction.

notice

As a side note it should be mentioned that the mechanisms presented here are important beyond the Data Link Layer and are especially relevant for TCP on the Transport Layer. The difference, however, is that the mechanisms at the Data Link Layer are implemented between two systems that communicate with each other directly, whereas they are implemented end-to-end on the Transport Layer. The end systems communicate with each other at this layer via a series of intermediate systems.

3.7 Exercises - Data Link Layer

task

Tasks for beginners

Task 1:
A computer has an Ethernet interface and a WLAN interface. Their MAC addresses are 00:14:0b:61:73:d2 and 00:21:5D:38:04:C4. Find the vendor of the network interface cards with the help of the Wireshark website. Alternatively, you could find this information at IEEE itself.

Task 2:
Try out the CSMA animation from the University of Innsbruck. Then answer the following questions.

- What does "channel considered busy" mean?
- Can collisions be avoided completely in a situation with hidden terminals?

Task 3:
The CRC technique can be used to ensure that a frame has arrived correctly. The received frame looks the following way represented as a bit sequence: . The generator polynomial is: . Answer the following questions:

- What are the quotient and the remainder in the CRC calculation?
- Was the frame received correctly (on what grounds)?

Tasks for advanced students

Task 1:
Install the network simulator eNSP as well as Wireshark on a Windows computer. If you cannot do this, you will be provided with corresponding images, configuration files and Wireshark records.
In the network topology stp.topo, you can see a network that consists of different switches. Start one or more terminals and observe the changes in the MAC address tables on the switches (Note: After being switched on, the terminals transmit search queries in the network with the DHCP protocol; this enables the MAC addresses to be learned). Answer the following question about this:

- What is the entry for client 1 in the MAC address table?

The spanning tree protocol is also already active in the network. Answer the following questions in this context.

- Which switch is the root bridge?
- Which ports were disabled?

Also consider the recording of DHCP and STP with Wireshark.

- Can you see ethernet II or IEEE ethernet there?

Learning MAC address table.

Task 2:
Download the ethernet standard (registration with IEEE required).
http:// standards.ieee.org/about/get/802/802.3.html
pingTools network utilities StreamSoft, http://pingtools.org, Ekahau Heatmapper

3.8 Summary - Data Link Layer

In this chapter you have seen a wide range of topics in communication networks, which include issues such as safe data transmission as well as collision-free and collision-prone medium access. Ethernet, WLAN and PPP were presented as important protocols.

I Bibliography

Doyl06 Jeff Doyle: "OSPF and IS-IS: Choosing an IGP for Large-Scale Networks", Addison Wesley, Upper Saddle River, NJ, 2006.

KuRo14 James F. Kurose und Keith W. Ross: "Computernetzwerke - Der Top-Down-Ansatz", 6. Auflage, Pearson Studium, München, 2014.

Rech12 Jörg Rech: "Wireless LANs", 4. Auflage, Heise Zeitschriften Verlag, 2012.

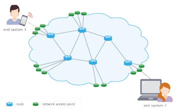
Rech14 Jörg Rech: "Ethernet", 3. Auflage, Heise Zeitschriften Verlag, 2014.

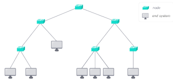
Roth10 Jörg Roth: "Prüfungstrainer Rechnernetze", Vieweg+Teubner, 2010.

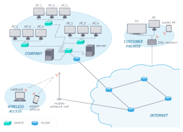
Schi03 Jochen Schiller: "Mobilkommunikation", 2. Auflage, Pearson Studium, München, 2003.

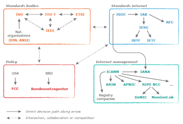
TaWe12 Andrew S. Tanenbaum und David J. Wetherall: "Computernetzwerke", 5. Auflage, Pearson Studium, München, 2012.

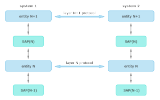
II List of figures

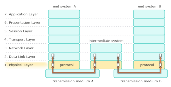


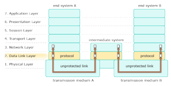


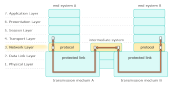


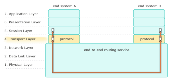


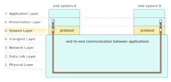


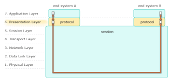


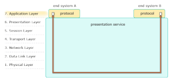


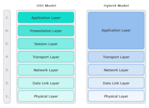


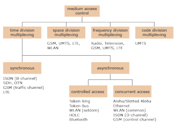


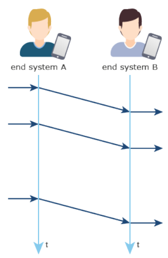


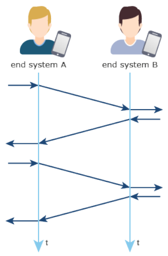


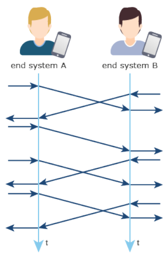


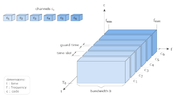


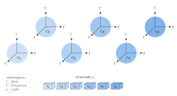


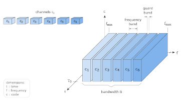

end system A
end system B
t
t

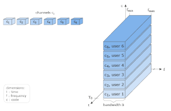

end system A
end system B
t
t

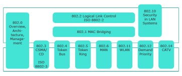

end system A
end system B
t
t

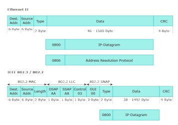


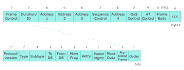


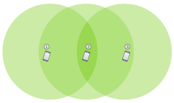


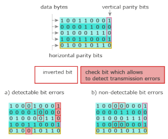











data bytes
vertical parity bits
horizontal parity bits
inverted bit
check bit which allows to detect transmission errors
a) detectable bit errors
b) non-detectable bit errors

data block
CRC

III List of tables

IV List of media

V Index