

Solution of Exercise Sheet 3

Exercise 1 (Bridges and Switches)

1. Explain what the purpose of **Bridges** in computer networks is.

For connecting different physical networks, Bridges are required because they forward frames from one physical network to another one.

Bridges and Switches check the correctness of the frames via checksums.

2. Give the number of **interfaces** („Ports“) a Bridge provides.

2 ports.

3. Explain what the major difference between **Bridges** and **Layer-2-Switches** is.

Bridges with > 2 ports are called Multiport Bridge or Layer-2-Switch.

4. Explain why Bridges and Layer-2-Switches do not require **physical or logical addresses**.

Bridges do not need addresses for filtering and forwarding the frames, because they do not actively participate in the communication. They work transparent, just like the devices of the Physical Layer.

5. Name at least two **examples** of Bridge implementations.

WLAN Bridges and Laser Bridges.

6. Explain what the advantage of **Learning Bridges** is in contrast to „dumb“ Bridges.

Learning Bridges learn which network devices are accessible via which port.

7. Name the information that is stored in the **forwarding tables** of Bridges.

The information, which network devices are accessible via which port in local forwarding tables.

8. Explain what happens, if for a network device, no entry exists in the **forwarding table** of a Bridge.

This is not a problem because the table is only used for optimization. If for a network device no entry in the forwarding table exists, the Bridge forwards the frame to every port, which is connected to a physical network.

9. Explain why Bridges try to avoid **loops**.

Loops can cause malfunctions and reduce the performance of the network or even lead to a network failure.

10. Name the protocol that Bridges implement for **handling loops**.

Spanning Tree Protocol (STP).

11. Explain what a spanning tree is.

It is a subgraph of the graph, which covers all nodes, but it is cycle-free, because edges have been removed.

12. Give the information, the **Bridge ID** according to the IEEE contains.

The Bridge ID consists of the Bridge priority (2 bytes) and MAC address (6 bytes) of the Bridge port with the lowest port ID.

13. Explain what the difference between the **Bridge ID** according to the IEEE and the **Cisco extended version** of the Bridge ID is.

Cisco subdivides the original 2 bytes long part for the Bridge priority. 4 bits now represent the Bridge priority. The remaining 12 bits are used to encode the VLAN ID.

14. Give the number of priority values that can be encoded with the **Bridge ID** according to the IEEE.

65,536 priority values can be represented.

15. Give the number of priority values that can be encoded with the **Cisco extended version** of the Bridge ID.

4 bits represent the Bridge priority \implies only 16 values can be represented.

16. Explain what a **Bridge Protocol Data Unit** (BPDU) message is and for what purpose it is used.

Bridges exchange information about Bridge IDs and path costs via special data frames, called Bridge Protocol Data Unit (BPDU). These messages are used by the Spanning Tree Protocol (STP).

17. Give the selection criteria for determining, whether a Bridge becomes the **Root Bridge**.

First, the Bridges have to determine the Bridge with the lowest Bridge Priority in the Bridge ID. This Bridge is the Root Bridge of the spanning tree to be generated.

18. Explain what a **Designated Bridge** is and what its task is.

For each physical network, a single one of the directly connected Bridges needs to be selected as responsible for forwarding the frames towards in the direction of the Root Bridge. This Bridge is called Designated Bridge for this network.

19. Give the number of **Designated Bridges**, a computer network contain.

For each physical network, a single Designated Bridge exists.

20. Give the selection criteria for determining, whether a Bridge becomes a **Designated Bridge**.

The Bridge with the lowest path costs to the Root Bridge is selected as Designated Bridge.

21. Explain what the impact of Bridges and Layer-2-Switches on the **collision domain** is.

If a physical network is subdivided via a Bridge or Switch, also the collision domain is divided and the number of collisions decreases.

For Bridges and Switches, each port forms its own collision domain.

22. Explain what a switched network is.

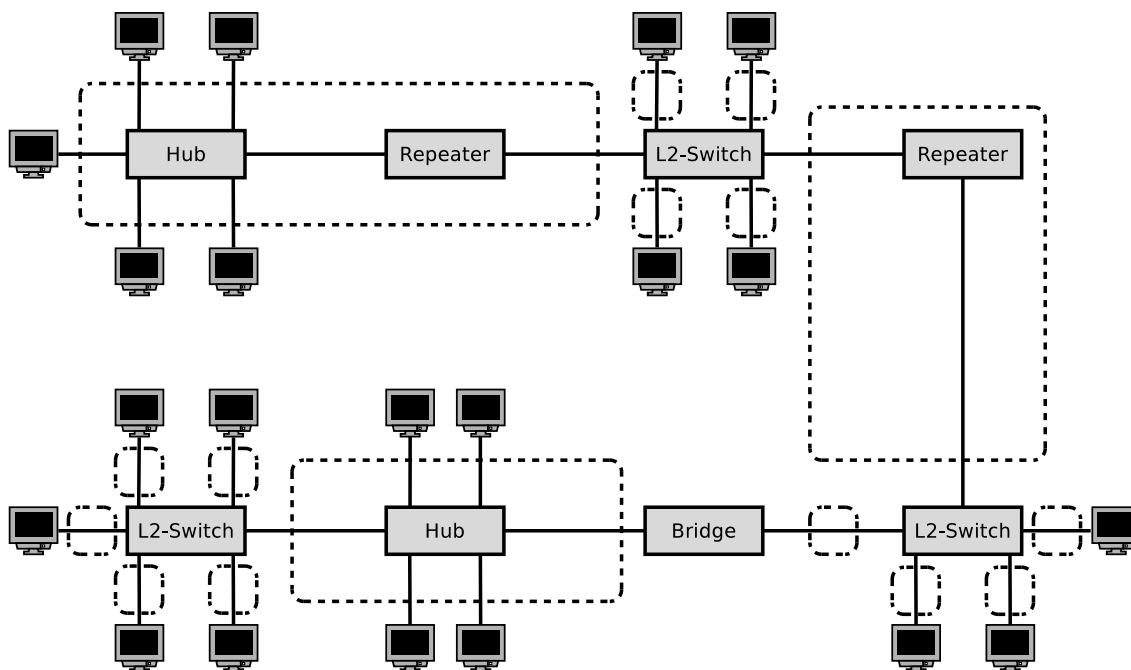
In a switched network, each port of the switches is connected with just a single network device.

23. Explain the benefit of a switched network compared to a non-switched network.

Such a network is free from collisions and state of the art.

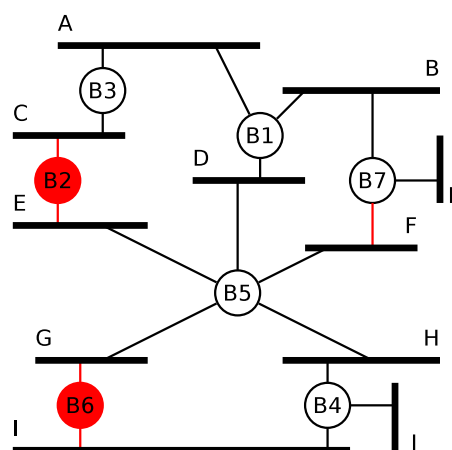
Exercise 2 (Collision Domain)

Sketch in the diagram all **collision domains**.



Exercise 3 (Spanning Tree Protocol)

The figure shows the physical connections of a network topology. A-J are physical networks (LANs). B1-B7 are Bridges (L2-Switches). All Bridges boot up at the same time after a power failure. Highlight in the figure which ports and Bridges are not used when the Spanning Tree Protocol is used.



Attention: If multiple paths from a network to the root bridge have the same distance, then take the bridge IDs as decision criterion. The smaller the ID of a bridge is, the higher is its priority.

Non used Bridges and ports are highlighted with red color.

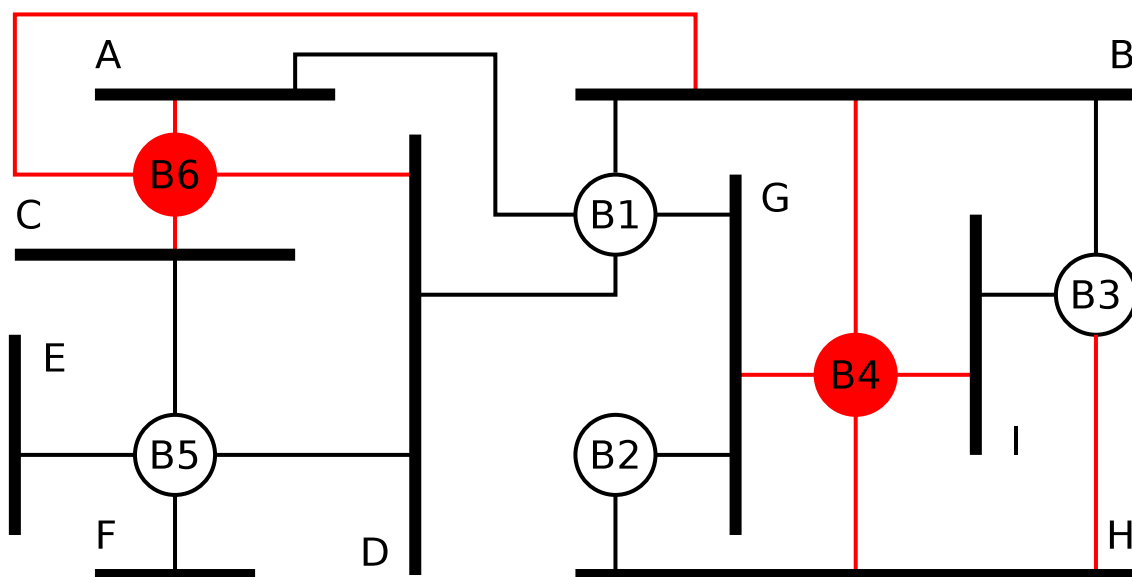
Hint: For solving this question, identify the Root Bridge first. It is the Bridge with the lowest Bridge Priority value inside the Bridge ID. In this network topology, we

just have Bridge ID values 1-7. Therefore the Root Bridge is B1. As the next step, for each physical network A-J, a single one of the directly connected Bridges needs to be selected as responsible for forwarding the frames into the Root Bridge's direction. The Bridge is called Designated Bridge for this network. The Bridge with the lowest path costs (minimum number of hops) to the Root Bridge becomes the Designated Bridge. If multiple Bridges exist for a physical network that have the same distance (number of hops) to the Root Bridge, the Bridge with the lowest Bridge Priority value inside the Bridge ID becomes the Designated Bridge.

Exercise 4 (Spanning Tree Protocol)

The figure shows the physical connections of a network topology. A-I are physical networks (LANs). B1-B6 are Bridges (L2-Switches). All Bridges boot up at the same time after a power failure. Highlight in the figure which ports and Bridges are not used when the Spanning Tree Protocol is used.

Attention: If multiple paths from a network to the root bridge have the same distance, then take the bridge IDs as decision criterion. The smaller the ID of a bridge is, the higher is its priority.



Non used Bridges and ports are highlighted with red color.

Hint: For solving this question, identify the Root Bridge first. It is the Bridge with the lowest Bridge Priority value inside the Bridge ID. In this network topology, we just have Bridge ID values 1-6. Therefore the Root Bridge is B1. As the next step, for each physical network A-I, a single one of the directly connected Bridges needs to be selected as responsible for forwarding the frames into the Root Bridge's direction. The Bridge is called Designated Bridge for this network. The Bridge with the lowest path costs (minimum number of hops) to the Root Bridge becomes the Designated Bridge. If multiple Bridges exist for a physical network that have the same distance

(number of hops) to the Root Bridge, the Bridge with the lowest Bridge Priority value inside the Bridge ID becomes the Designated Bridge.

Exercise 5 (Addressing in the Data Link Layer)

1. Data Link Layer protocols specify the **format** of...
 physical network addresses logical network addresses
2. Give the name (technical term) of **physical network addresses** (Data Link Layer addresses).
MAC addresses (Media Access Control).
3. Name the protocol that is used by Ethernet for the **address resolution**.
Address Resolution Protocol (ARP).
4. Which devices receive a frame with the **destination address FF-FF-FF-FF-FF-FF**.
This address is the MAC broadcast address. Every participant in the physical network receives this frame.
5. Explain what **MAC spoofing** is.
MAC addresses can be modified via software. The method is called MAC spoofing.

Exercise 6 (Framing)

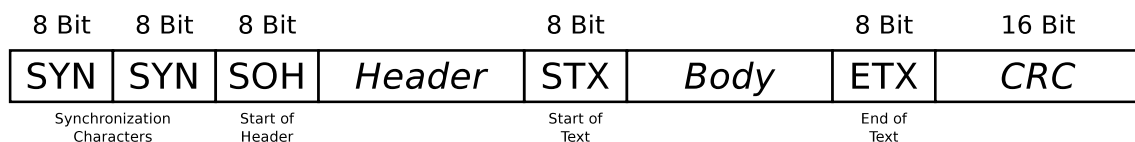
1. One way to mark the frames' borders is via **character count in the frame header**. Name a potential issue that can arise from this method.
If the field, which contains the number of bytes payload inside the frame is modified during transmission, the receiver is unable to correctly detect the end of the frame.
2. One way to mark the frames' borders is via **Byte Stuffing**. Name a drawback of this method.
The strong relationship with the ASCII character encoding. The network t
3. Explain why up-to-date Data Link Layer protocols, such as Ethernet and WLAN, work **bit-oriented and not byte-oriented**.
Because this allows using any character encoding.

4. Mark the information that an **Ethernet frame** contains.

- Sender IP address
- Sender MAC address
- Hostname of the receiver
- Information about the Transport Layer protocol used
- Preamble to synchronize the receiver
- Port number of the receiver
- CRC checksum
- Information about the Application Layer protocol used
- VLAN tag
- Receiver MAC address
- Receiver IP address
- Information about the Network Layer protocol used
- Hostname of the sender
- Signals, which are transmitted via the transmission medium
- Port number of the sender

Exercise 7 (Byte Stuffing)

The Data Link Layer splits the bit stream from the Physical Layer into frames. The character-oriented protocol BISYNC uses control characters to mark the structure of the frames. The start of a frame highlights the character **SYN**. The start of the header highlights the character **SOH** (*Start of header*). The payload is located between **STX** (*Start of text*) and **ETX** (*End of text*). The figure shows the structure of BISYNC frames:



| | | | | | |
|-----------------------------|-----|-----|-----|-----|-----|
| Control character | SOH | STX | ETX | DLE | SYN |
| Hexadecimal notation | 01 | 02 | 03 | 10 | 16 |

If the payload (body) contains the control characters **ETX** and **DLE** (*Data Link Escape*), they are protected (*escaped*) by the Data Link Layer protocol with a stuffed **DLE** character. A single **ETX** in the payload area is represented by the sequence **DLE ETX**. The **DLE** character itself is represented by the sequence **DLE DLE**.

Mark the payload inside the following BISYNC frames.

1. 16 16 01 99 98 97 96 95 02 A1 A2 A3 A4 A5 03 A0 B7

Payload: A1 A2 A3 A4 A5

2. 16 16 01 99 98 97 96 95 02 05 04 10 03 02 01 03 76 35

Payload: 05 04 03 02 01

3. 16 16 01 99 98 97 96 95 02 10 03 10 10 10 03 03 92 55

Payload: 03 10 03

4. 16 16 01 99 98 97 96 95 02 10 10 10 10 10 03 01 02 A1 03 99 B2

Payload: 10 10 03 01 02 A1

Source: Jörg Roth. *Prüfungstrainer Rechnernetze. Vieweg (2010) and Wikipedia*

Exercise 8 (Bit Stuffing)

The Data Link Layer protocol HDLC (High-Level Data Link Control) uses Bit Stuffing. If the sender discovers 5 consecutive 1 bits in the bitstream from the Network Layer, it *stuffs* a single 0 bit into the outgoing bit stream. If the receiver discovers 5 consecutive 1 bits, followed by a single 0 bit in the bit stream from the Physical Layer, it removes (*destuffs*) the 0 bit.

Give the encoding for each one of the following bit sequences, when the sender *stuffs* after 5 consecutive 1 bits a single 0 bit into the bit stream from the Network Layer.

1. 01111110 10100111 11111000 11110010 10011111 10111111 11100101

Bit stream with stuffed 0 bits:

011111010 10100111 110111000 11110010 100111110 101111101 11100101

2. 00111111 01110001 11110011 11111100 10101010 11001111 11100001

Bit stream with stuffed 0 bits:

001111101 01110001 111100011 111011100 10101010 11001111 101100001

3. 11111111 11111111 11111111 11111111 11111111 11111111 11111111

Bit stream with stuffed 0 bits:

111110111 1101111101 111101111 1011111011 1110111110 111110111 1101111101

Exercise 9 (Error Detection – CRC)

1. Calculate the frame to be transferred.

Generator polynomial: 100101

Payload: 11010011

The generator polynomial has 6 digits \implies five 0 bits are appended

Frame with appended 0 bits: 1101001100000

```
1101001100000
100101|||||
-----v|||||
 100011|||||
 100101|||||
  -----vvv|||
    110100|||
    100101|||
    -----v||
      100010||
      100101||
      -----vv
        11100 = Remainder
```

Remainder: 11100

Transferred frame: 1101001111100

2. Check, if the received frame was transmitted correctly.

Transferred frame: 1101001110100

Generator polynomial: 100101

```
1101001110100
100101|||||
-----v|||||
 100011|||||
 100101|||||
  -----vvv|||
    110110|||
    100101|||
    -----v||
      100111||
      100101||
      -----vv
        1000 => Error
```

3. Check, if the received frame was transmitted correctly.

Transferred frame: 1101001111100

Generator polynomial: 100101

```
1101001111100
100101|||||
-----v|||||
```

```
100011|||||
100101|||||
-----vvv|||
 110111|||
 100101|||
-----v||
 100101||
 100101||
-----vv
      00 => Transmission was error-free
```

4. Calculate the frame to be transferred.

Generator polynomial: 100101
Payload: 10110101

The generator polynomial has 6 digits \implies five 0 bits are appended.

Frame with appended 0 bits: 1011010100000

```
1011010100000
100101|||||
-----vv||||
 100001||||
 100101||||
-----vv|||
   100000||
   100101||
-----vv
       10100 = Remainder
```

Remainder: 10100
Transferred frame: 1011010110100

5. Check, if the received frame was transmitted correctly.

Transferred frame: 1011010110110
Generator polynomial: 100101

```
1011010110110
100101|||||
-----vv||||
 100001||||
 100101||||
-----vvv||
   100101||
   100101||
-----vv
```

10 => Error

6. Check, if the received frame was transmitted correctly.

Transferred frame: 1011010110100
Generator polynomial: 100101

```
1011010110100
100101|||||
-----vv||||
 100001||||
 100101||||
  -----vvv|
    100101|
    100101|
  -----vv
      00 => Transmission was error-free
```

7. Check, if the received frame was transmitted correctly.

Transferred frame: 1010010110100
Generator polynomial: 100101

```
1010010110100
100101|||||
-----vv||||
 110001||||
 100101||||
  -----v|||
    101001|||
    100101|||
  -----vv|
    110001||
    100101||
  -----v|
    101000|
    100101|
  -----v
      11010 => Error
```

8. Calculate the frame to be transferred.

Generator polynomial: 100000111
Payload: 1101010101110101

The generator polynomial has 9 digits \implies eight 0 bits are appended.

Frame with appended 0 bits: 110101010111010100000000

```

110101010111010100000000
100000111|||||
-----v|||||
 101011011|||||
100000111|||||
-----vv|||||
 101110011|||||
100000111|||||
-----vv|||||
 111010001|||||
100000111|||||
-----v|||||
 110101100|||||
100000111|||||
-----v|||||
 101010111|||||
100000111|||||
-----vv|||||
 101000000|||||
100000111|||||
-----vv|||||
 100011100|||||
100000111|||||
-----vvvv
 110110000
 100000111
-----
10110111 = Remainder

```

Remainder: 10110111

Transferred frame: 110101010111010110110111

9. Check, if the received frame was transmitted correctly.

Transferred frame: 110101010111110110110111

Generator polynomial: 100000111

```

110101010111110110110111
100000111|||||
-----v|||||
 101011011|||||
100000111|||||
-----vv|||||
 101110011|||||
100000111|||||
-----vv|||||

```

```

111010011|||||
100000111|||||
-----v|||||
110101000|||||
100000111|||||
-----v|||||
101011111|||||
100000111|||||
-----vv|||||
101100010|||||
100000111|||||
-----vv|||
110010111|||
100000111|||
-----v|||
100100000|||
100000111|||
-----vvv
100111111
100000111
-----
111000 => Error
    
```

10. Check, if the received frame was transmitted correctly.

Transferred frame: 110101010111010110110111
 Generator polynomial: 100000111

```

110101010111010110110111
100000111|||||
-----v|||||
101011011|||||
100000111|||||
-----vv|||||
101110011|||||
100000111|||||
-----vv|||||
111010001|||||
100000111|||||
-----v|||||
110101100|||||
100000111|||||
-----v|||||
101010111|||||
100000111|||||
-----vv|||||
101000010|||||
    
```

```

1000001111111111
-----vv1111
 1000101111111111
1000001111111111
-----vvvv
 100000111
 100000111
-----
0 => Transmission was error-free

```

Exercise 10 (Error Correction – Simplified Hamming Code)

Transmission errors can be detected via CRC checksums. If it is important to not only recognize errors, but also to be correct them, then the data to be transmitted must be encoded in a way, that error-correction is possible. Error correction can be realized e.g. via the **Simplified Hamming Code** we discussed in the computer networks course.

1. A message of 8 bits payload (10011010) needs to be transferred. Calculate the message, that will be transmitted (payload inclusive parity bits).

Step 1: Determine parity bit positions:

```

          Position:  1  2  3  4  5  6  7  8  9 10 11 12
Data to be transmitted:  ?  ?  1  ?  0  0  1  ?  1  0  1  0

```

Step 2: Calculate parity bit values:

```

0011 Position 3
0111 Position 7
1001 Position 9
XOR 1011 Position 11
-----
0110 = parity bit values

```

Step 3: Insert parity bit values into the transmission:

```

          Position:  1  2  3  4  5  6  7  8  9 10 11 12
Data to be transmitted:  0  1  1  1  0  0  1  0  1  0  1  0

```

2. The following messages have been received. Verify, if they were transmitted correctly.

a) 00111101

Received data: 1 2 3 4 5 6 7 8
 0 0 1 1 1 1 0 1

0011 Position 3
0101 Position 5
XOR 0110 Position 6

0000 Parity bits calculated
XOR 0011 Parity bits received

0011 => Bit 3 ist defective!

b) 101110100010

Received data: 1 2 3 4 5 6 7 8 9 10 11 12
 1 0 1 1 1 0 1 0 0 0 1 0

0011 Position 3
0101 Position 5
0111 Position 7
XOR 1011 Position 11

1010 Parity bits calculated
XOR 1010 Parity bits received

0000 => Correct transmission

c) 001101100100

Received data: 1 2 3 4 5 6 7 8 9 10 11 12
 0 0 1 1 0 1 1 0 0 1 0 0

0011 Position 3
0110 Position 6
0111 Position 7
XOR 1010 Position 10

1000 Parity bits calculated
XOR 0010 Parity bits received

1010 => Bit 10 ist defective!

d) 0001101100101101

Received data: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
 0 0 0 1 1 0 1 1 0 0 1 0 1 1 0 1

```
00101 Position 5
00111 Position 7
01011 Position 11
01101 Position 13
XOR 01110 Position 14
-----
01010 Parity bits calculated
XOR 00111 Parity bits received
-----
01101 => Bit 13 ist defective!
```

Exercise 11 (Media Access Control)

1. Explain why computer networks use protocols for media access control.

With Ethernet and WLAN, the network devices or stations use a shared transmission medium. To coordinate media access and to avoid collisions, media access control methods are required.

2. Explain why do Ethernet and WLAN use different media access control methods.

With wireless networks, it is not guaranteed that all stations can detect all collisions.

In wired networks with a shared transmission medium, each participant receives the transmissions of all other participants.

3. Explain how Ethernet devices react, when they detect a collision.

If a collision is detected, the sender stops the frame transmission and sends the jam signal to announce the collision. If the maximum number of transmission attempts is not yet reached, the sender tries to transmit the frame again after a random time.

4. Explain why it is important that the transmission of a frame is not completed when a collision occurs in an Ethernet network.

Otherwise, the network device might already be finished with the transmission and believes the transmission was successful.

5. Explain what is done to ensure that the transmission of a frame is not completed when a collision occurs in an Ethernet network.

Each frame must have a certain minimum length. It must be dimensioned in a way, that the transmission duration for a frame with minimum length does not fall below the maximum RTT (round trip time).

This ensures that a collision reaches the sender before its transmission is finished. If a sender detects a collision, it knows that its frame has not arrived correctly at the receiver, and can try the transmission again later.

6. Name the two special characteristics of the transmission medium in wireless networks that cause undetected collisions at the receiver.

Hidden terminal problem and Fading.

7. Describe both special characteristics of subtask 6.

Hidden terminal problem (problem caused by invisible or hidden terminal device). Because of obstacles, not all stations can detect all transmissions, although they interfere each other at the Access Point.

Fading (decreasing signal strength). The electromagnetic waves of the wireless network are weakened by obstacles and in free space. Caused by the positions of stations to each other, their signals are so weak, that the stations cannot detect each others transmissions.

8. Explain what the Network Allocation Vector (NAV) is and for what purpose it is used.

The NAV is a counter variable which is maintained by each node itself. It contains the expected time when the transmission medium will be occupied. It reduces the number of collisions when CSMA/CA is used.

9. Explain what the Contention Window (CW) is and for what purpose it is used.

If the NAV and another DIFS with an idle transmission medium has expired, a backoff time is created from the CW. The backoff time is calculated by using a random value between the minimum CW and maximum CW and multiplying this random value with the slot time. After the backoff time has expired, the frame is transmitted. The CW prevents that all stations which wait for a free transmission medium, start their transmissions at the same time.

10. Name a benefit and a drawback of using the control frames Request To Send (RTS) and Clear To Send (CTS).

Advantage: It reduces collisions because it solves the problem of hidden terminals.

Drawbacks: Delays occur, which are caused by the reservation of the transmission medium. The RTS and CTS frames, which are used to reserve the transmission medium, are overhead.

Exercise 12 (Address Resolution Protocol)

1. Explain what the function of the Address Resolution Protocol is.

The Address Resolution Protocol (ARP) is used to convert IP address of the Network Layer to MAC address of the Data Link Layer.

2. Explain what the ARP cache is.

The ARP cache is a table, which contains IP addresses and MAC addresses, that belong together. It is used to speed up the address resolution.