HIRSCHMANN

Basics manual

Local Area Network (LAN)

HIRSCHMANN

Basics manual
Local Area Network (LAN)

The naming of copyrighted trademarks in this manual, even when not specially indicated, should not be taken to mean that these names may be considered as free in the sense of the trademark and tradename protection law and hence that they may be freely used by anyone.

This manual has been created by Hirschmann electronics GmbH & Co. KG according to the best of our knowledge. Hirschmann reserves the right to change the contents of this manual without prior notice. Hirschmann can give no guarantee in respect of the correctness or accuracy of the details in this manual.

Hirschmann can accept no responsibility for damages, resulting from the use of the network components or the associated operating software. For the rest, we refer to the conditions of use specified in the license contract.

Printed in Germany

Hirschmann Electronics GmbH & Co. KG
Automation and Network Solutions
Stuttgarter Straße 45-51
72654 Neckartenzlingen
Telefon (07127) 14-1538

039 6xx-001-01-0900

Hirschmann worlwide:

■ Germany

Hirschmann Electronics GmbH & Co. KG
Automation and Network Solutions
Stuttgarter Straße 45-51
D-72654 Neckartenzlingen
Tel. ++49-7127-14-1527
Fax ++49-7127-14-1542
email: ans-hi-line@nt.hirschmann.de
Internet: www.hirschmann.de

■ Austria

Hirschmann Austria GmbH
Oberer Paspelweg 6-8
A-6830 Rankweil-Brederis
Tel. ++43-5522-3070
Fax ++43-5522-307555
A-1230 Wien
Tel. ++43-1-6174646
Fax ++43-1-6174646
email: cherz@nt.hirschmann.de

■ Switzerland

Hirschmann Electronics GmbH & Co. KG, Neckartenzlingen
Niederlassung Uster
Seestr. 16
CH-8610 Uster
Tel. ++41-1905-8282
Fax ++41-1905-8289
email: ans_ch@hirschmann.ch

■ France

Hirschmann Electronics S.A.
24, rue du Fer à Cheval, Z.I.
F-95200 Sarcelles
Tel. ++33-1-39330280
Fax ++33-1-39905968
email: ans@hirschmann.fr

■ Great Britain

Hirschmann Electronics Ltd.
St. Martins Way
St. Martins Business Centre
GB-Bedford MK42 OLF
Tel. ++44-1234-345999
Fax ++44-1234-352222
email: enquiry@hirschmann.co.uk

■ Netherlands

Hirschmann Electronics B.V.
Pampuslaan 170
NL-1382 JS Weesp
Tel. ++31-294-462555
Fax ++31-294-480639
email: ans@hirschmann.nl

■ Spain

Hirschmann Electronics S.A.
Calle Traspaderne, 29
Barrio del Aeropuerto
Edificio Barajas 1,2 Planta
E-28042 Madrid
Tel. ++34-1-7461730
Fax ++34-1-7461735
email: hes@hirschmann.es

■ Hungary

Hirschmann Electronics Kft.
Rokolya u. 1-13
H-1131 Budapest
Tel. ++36-1-3494199
Fax. ++36-1-3298453
email: hirh.vez@nap-szam.hu

■ USA

Hirschmann Electronics Inc.
30 Hook Mountain Road _ Unit 201
USA-Pine Brook, N. J. 07058
Tel. ++1-973-8301470
Fax ++1-973-8302000
email: ans@hirschmann-usa.com

■ Singapore

Hirschmann Electronics Pte. Ltd.
3 Toh Tuck Link
# 04-01 German Districentre
Singapore 596228
Tel: (65)463 5855
Fax:(65) 463 5755
email: hirschmann.ap@pacific.net.sg

■ China (PRC)

Hirschmann Electronics
Shanghai Rep. Office
Room 518, No. 109 Yangdang Road
Lu Wan District, 200020
SHANGHAI, PRC
Tel +86-21 63 58 51 19
Fax +86-21 63 58 51 25
E-mail: hirschsh@public4.sta.net.cn

Hirschmann worlwide:

Content

1 Overview

This chapter allows to overview the historical development of Ethernet and his important charactaristics.

1.1 Historical Development of Ethernet

The constant growth in the use of data processing systems and their introduction into many areas (office communications, scientific-technical applications, construction, manufacturing, etc.) make high-capacity and high-function data networks mandatory. One possible way to economically solve this problem is the use of local area networks.

In 1972 Xerox began to develop the bus-connected local area network (LAN) at its Palo Alto Research Center using the **CSMA/CD** access method. The name of the access method stands for three actions that characterize it::

- **C**arrier **S**ense
- **M**ultiple **A**ccess
- **C**ollision **D**etection.

The increasing significance of LANs caused three companies, Digital Equipment Corporation (DEC), Intel Corporation, and Xerox, to found the DIX consortium, whose goal was to continue development, building on the good results already achieved by Xerox.

In 1980, DIX published the first specifications for Ethernet Version 1.0.

At the same time, working group 802 of the Institution of Electronical and Electronic Engineers (IEEE) began to develop a standard for a CSMA/CD bus LAN. The Ethernet Version 1.0 specifications formed the base for this work. The T24 Committee on Communication Protocols of the **E**uropean **C**omputer **M**anufacturers **A**ssociation (ECMA) also provided useful input for the development of the standard.

The result of this work is IEEE recommendation 802.3. In 1982, the DIX group modified their Ethernet Version 2.0 specifications to conform with the IEEE recommendations. In 1985, the recommendation was raised to the status of a standard.

The standard was submitted to the **I**nternational **S**tandardization **O**rganization / **I**nternational **E**lectrotechnical **C**ommission (ISO / IEC) with the goal of creating an international standard. This resulted in its being published as the ISO/IEC 8802-3 International Norm in 1988. No significant technical changes were made to the original standard as a result of this.

A further significant step for the success of local area networks was the creation of the ISO/OSI reference model (International **S**tandardization **O**rganisation / **O**pen **S**ystem **I**nterconnection).

This reference model had the following goals:

- To define a standard for information exchange between open systems;
- To provide a common basis for developing additional standards for open systems;
- To provide international teams of experts with functional framework as the basis for independent development of every layer of the model;
- To include in the model developing or already existing protocols for communications between heterogeneous systems;
- To leave sufficient room and flexibility for the inclusion of future developments.

The reference model consists of 7 layers, ranging from the application layer to the physical layer.
The model was published in October 1984 in international standard ISO 7498.

7	Application	Access to communication services from an application program
6	Presentation	Definition of the syntax for data communication
5	Session	Set up and breakdown of connections by synchronization and organization of the dialog
4	Transport	Specification of the terminal connection, with the necessary transport quality
3	Network	Transparent data exchange between two transport entities
2	Data-Link	Access to physical media and detection and resolution of transmission errors
1	Physical	Transmission of bit strings via physical media

Fig. 1: OSI reference model

However, the data rate and the transmission media were permanently adapted. The next data rate - 100 Mbit - was actually already attained by FDDI. The transition from 10 MBit Ethernet to FDDI was, however, not a very smooth one for users. Standardization of FDDI was also very sluggish, and the data terminal equipment never fell to price level that might have made them competitive in the market.

Thus, the development of 100 MBit Ethernet began. On the physical level, FDDI components were adopted. Since 1994 FDDI has at times been implemented with TP cable. Initially there were two approaches to finding a solution. The first one, Fast Ethernet, simply adapted all transmission parameters to the new speed. The other approach defined a new access method - demand priority - and from that time on was referred to as project group 802.12 by the IEEE. The sole disadvantage of the first proposal - reducing the spatial extent of a network to one tenth of its size - became insignificant due to the widespread availability of bridges and switches. Consequently, it became the new standard. It was adopted in 1995. Although 802.12 was also adopted, it hardly plays a role anymore.

The next level of speed appeared once and for all to belong to another form of transmission - ATM - which promised data rates in excess of 622 MBit.

This is why the idea of 1 GBit Ethernet, presented in 1995, was not taken very seriously. As it turned out, this appeared to be a quite premature. Work on the standard proceeded very quickly. For example, it was possible to adopt transmission components from Fiberchannel. Products already became available far before the standard was adopted in 1998. The first chips appeared at the end of 1996, and functional devices hit the market a year later. In 1999 even twisted pair transmission was standardized at this speed.

Ever since Gigabit Ethernet has become commonplace and the digitalization has continued its torrid pace, calls for even more bandwidth have become increasing louder. This has led to work on developing a 10 Gigabit Ethernet standard that got underway in 1999.

1.2 The ISO/IEC 8802-3 Standard

The most significant characteristic of a local area network conforming to ISO/IEC 8802-3 is that all network users have equal access to the transmission medium. In order to handle the inevitable collisions, reliable collision detection and unambiguous resolution are mandatory elements of any implementation of this norm.

1.2.1 CSMA/CD Access Method

There is no central station to monitor or control access to the local area network. Each member of the network monitors traffic on the network and, if the network is free, can start transmitting data immediately.

Sequence of a transmission occurrence:

1 Carrier Sense: Network members check to see if the transmission medium is free.

2 Multiple Access: If the transmission medium is free, any network member can start transmitting data.

3 Collision Detection: If more than one member of the network start transmitting data simultaneously, a data collision will result. The transmitting members will detect the collision and terminate transmission. A backoff strategy determines when the members can retry the data transmissions.

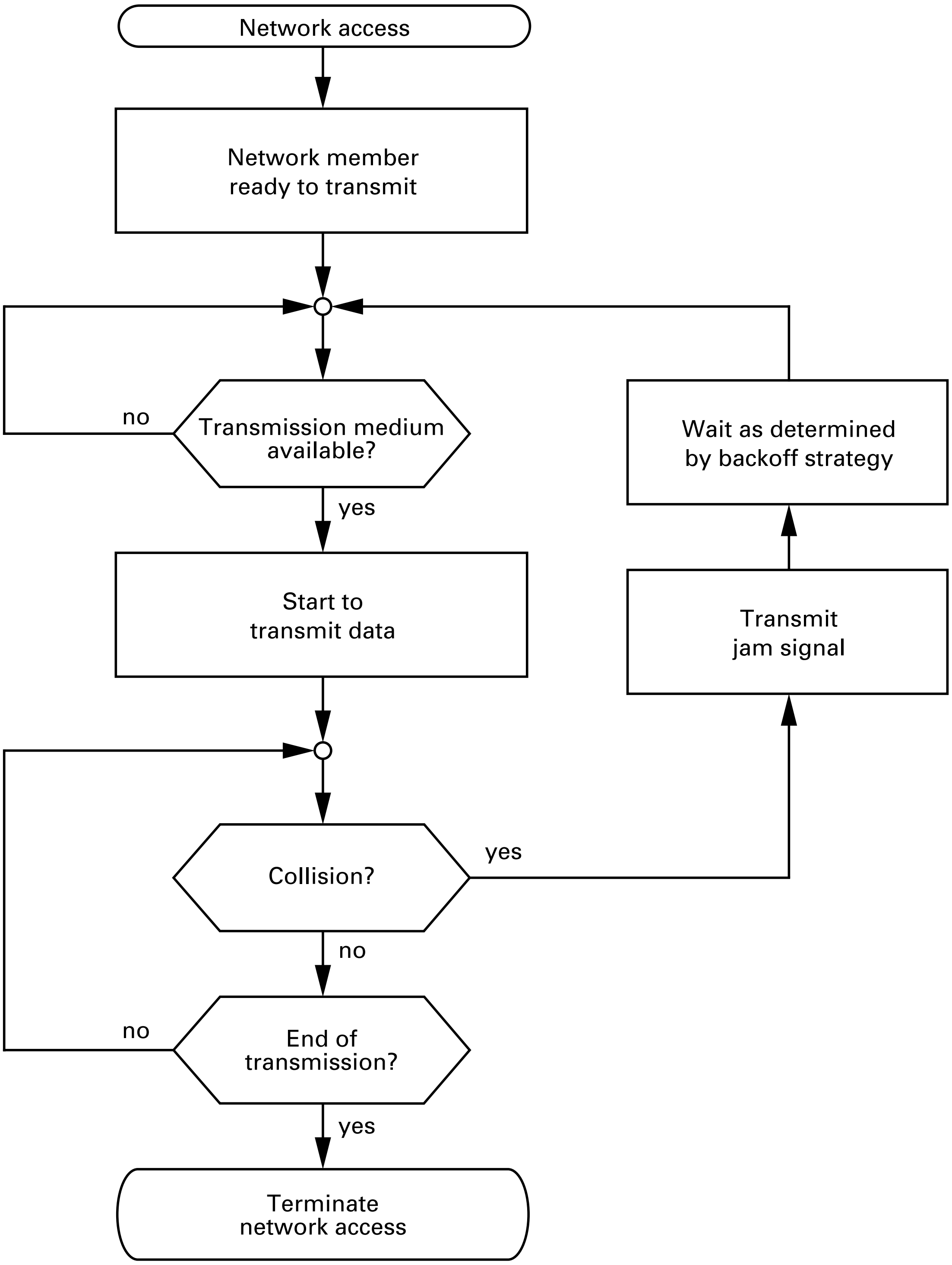


Fig. 2: CSMA/CD Access

1.2.2 Collisions

The logical result of the CSMA/CD method is that there is a finite probability that multiple users could attempt to access the medium simultaneously. For that reason, the access method must have a mechanism for dealing with any collisions as they occur.

Requirements for this mechanism:

- Detection of each collision by the participating network members.
- Termination of the transmission attempt in case of a collision.
- Renewed transmission attempt if the previous attempt has failed due to a collision.

1 The following conventions have been agreed to for meeting these requirements:
The signal transmission time depends on the minimum data packet length. ISO/IEC 8802-3 defines the time window (slot time) to be the time it takes from the beginning of the transmission until a collision at the far end of the transmission medium occurs. The slot time is 51.2 µs.

The minimum data packet length is equal to the slot time. This insures that the transmitting station can detect a collision while the transmission is still taking place and therefore knows that the transmission has failed. “Collision detection as a function of time and location. The jam (collision notification) signal from member 2 reaches member 1 while member 1 is still sending.” on page 24 shows this relationship. Network member 1 starts to transmit. Just before the transmission reaches network member 2, member 2 also begins to transmit. The signal from member 1 then reaches member 2 who detects the collision and transmits the 32-bit jam signal before terminating its own transmission. The jam signal arrives at member 1 within the slot time interval, that is, while member 1 is still transmitting. Member 1 is thus also able to detect the collision.

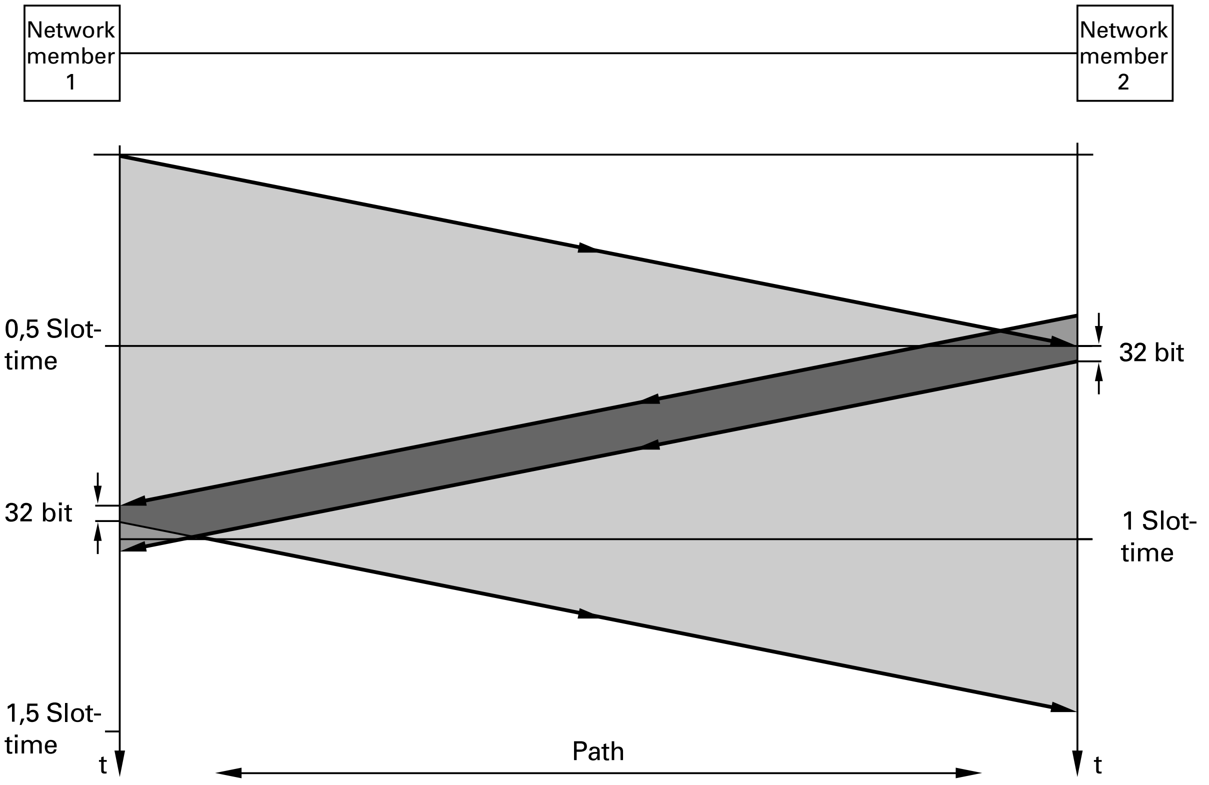


Fig. 3: Collision detection as a function of time and location. The jam (collision notification) signal from member 2 reaches member 1 while member 1 is still sending.

If a data packet is shorter than the slot time, it is possible that a transmitting member of the network might not be able to detect that a data packet it had just sent has been damaged by a collision. In that case, there would be no re-transmission of the damaged packet (see Fig. 4).

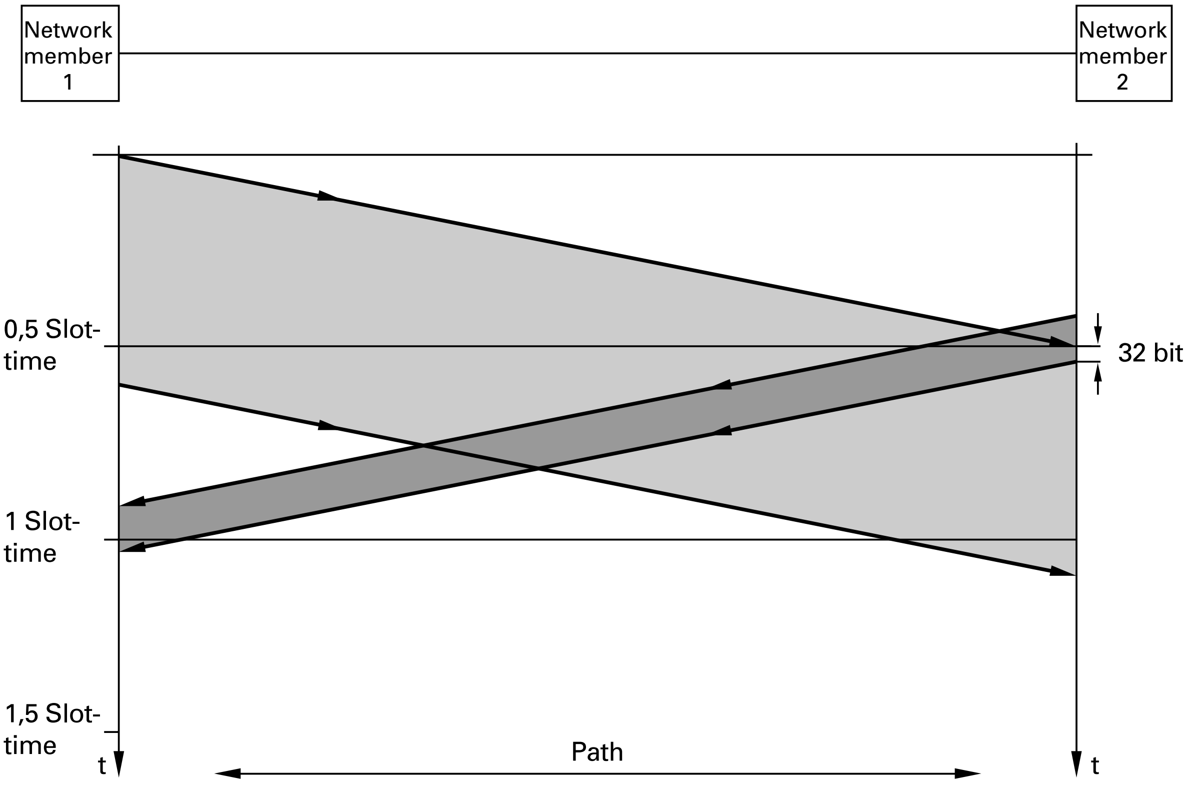


*Fig. 4: Data packet too short.
Network member 1 is not able to detect that the data packet just sent has been damaged by a collision.*

ISO/IEC specifies a slot time of 51.2 µs. Taking into consideration the repeater propagation times of a network of maximum size, this results in a maximum network size of 2500 meters. “Network Planning” on page 41 describes how this maximum can be extended.

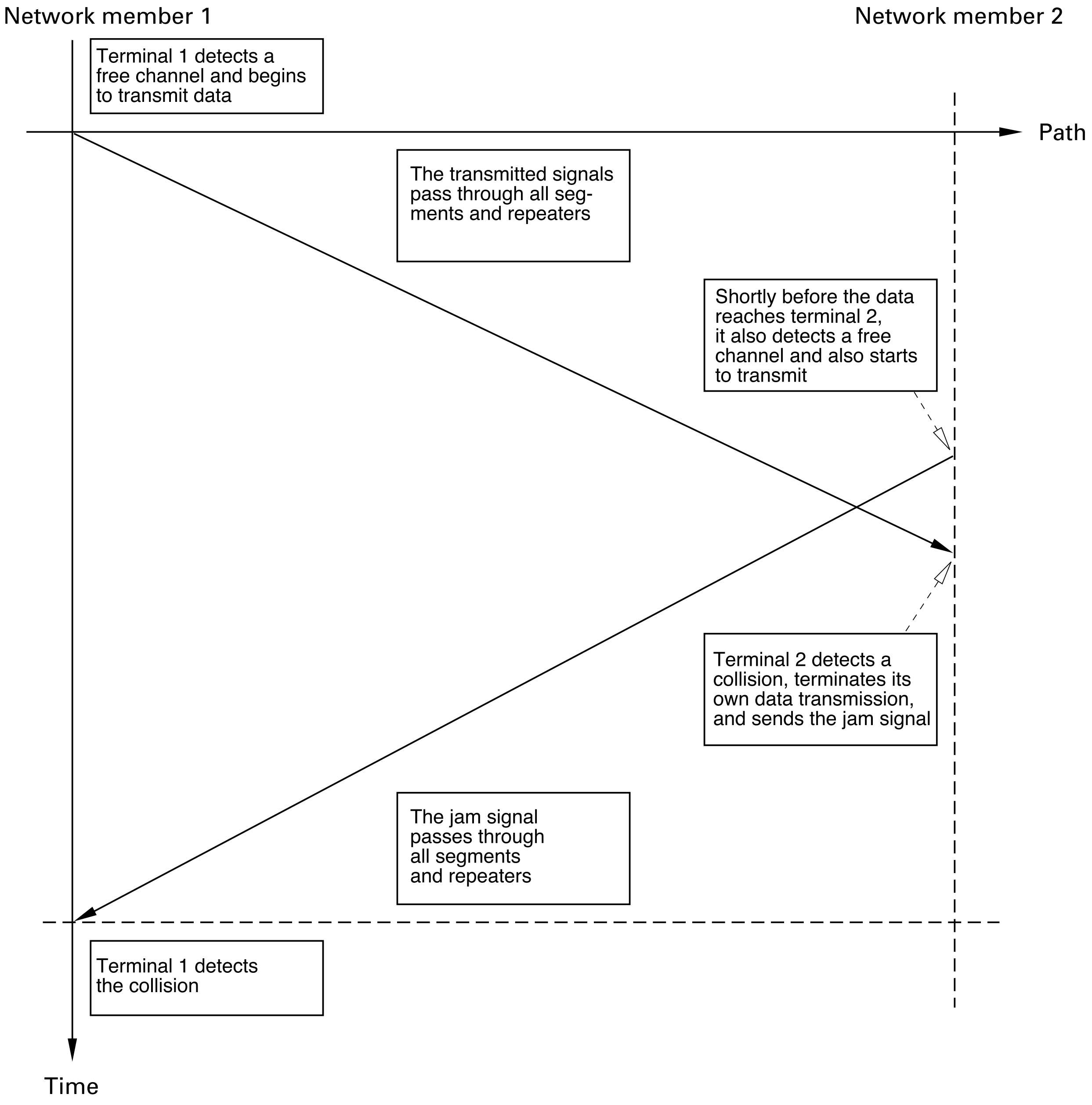

Network member 1
Network member 2
Terminal 1 detects a free channel and begins to transmit data
Path
The transmitted signals pass through all segments and repeaters
Shortly before the data reaches terminal 2, it also detects a free channel and also starts to transmit
Terminal 2 detects a collision, terminates its own data transmission, and sends the jam signal
The jam signal passes through all segments and repeaters
Terminal 1 detects the collision
Time

Fig. 5: Collision detection model

2 If a transmitting network station detects a collision, it must send at least 32 more bits (jam size) before finally terminating its transmission attempt. This minimum collision duration of 3.2 µs insures that each station in the network detects the collision.

3 If a station in the network is unable to transmit its data packet completely due to a collision, then it must wait a predetermined length of time and re-attempt the transmission.

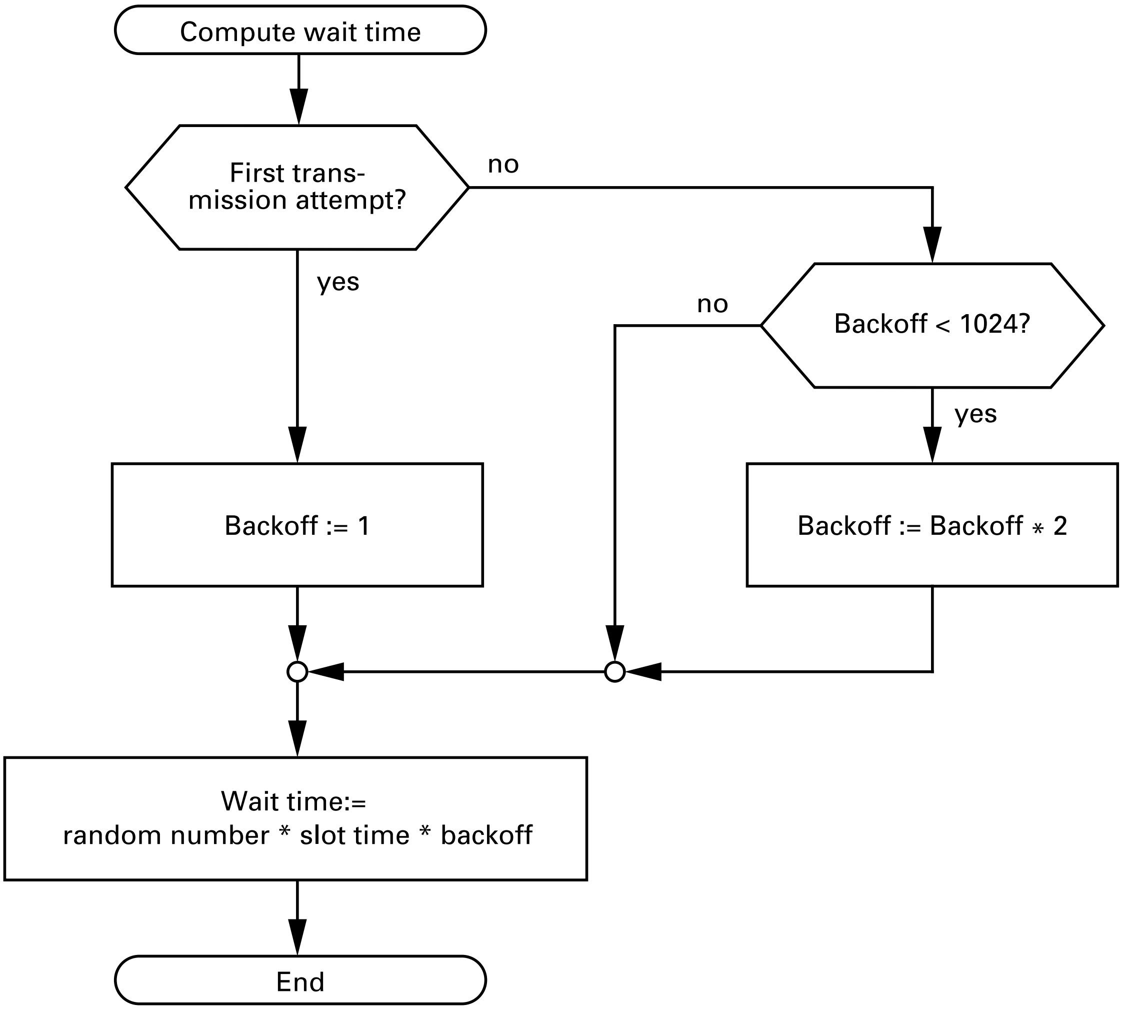


Fig. 6: Backoff algorithm

ISO/IEC 8802-3 allows up to 16 transmission attempts before finally giving up trying to send a data packet. The maximum value that "backoff" can assume is 1024. This means that "backoff" will not be increased after the tenth attempt.

1.2.3 Interpacket Gap

A minimum gap between packets is required as recovery time for CSMA/CD sub-layers and the physical medium. ISO/IEC 8802-3 defines this minimum interpacket gap to be 9.6 µs.

Transmission rate	Interpacket gap
10 MBit/s	9,6 µs
100 MBit/s	960 ns
1000 MBit/s	96 ns

Tab. 1: Interpacket gap in dependency of the transmission rate

The varying bit loss (preamble loss) of two successive data packets on the same path can cause the interpacket gap to shrink. A repeater regenerates the lost preamble bits of any packet passing through it. This gap shrinkage is called interpacket gap shrinkage.

If the first data packet (frame) loses more preamble bits on reception than the subsequent packet - A and B (see Fig. 7), then the gap will be reduced after the preamble has been regenerated by the repeater - C (see Fig. 7).

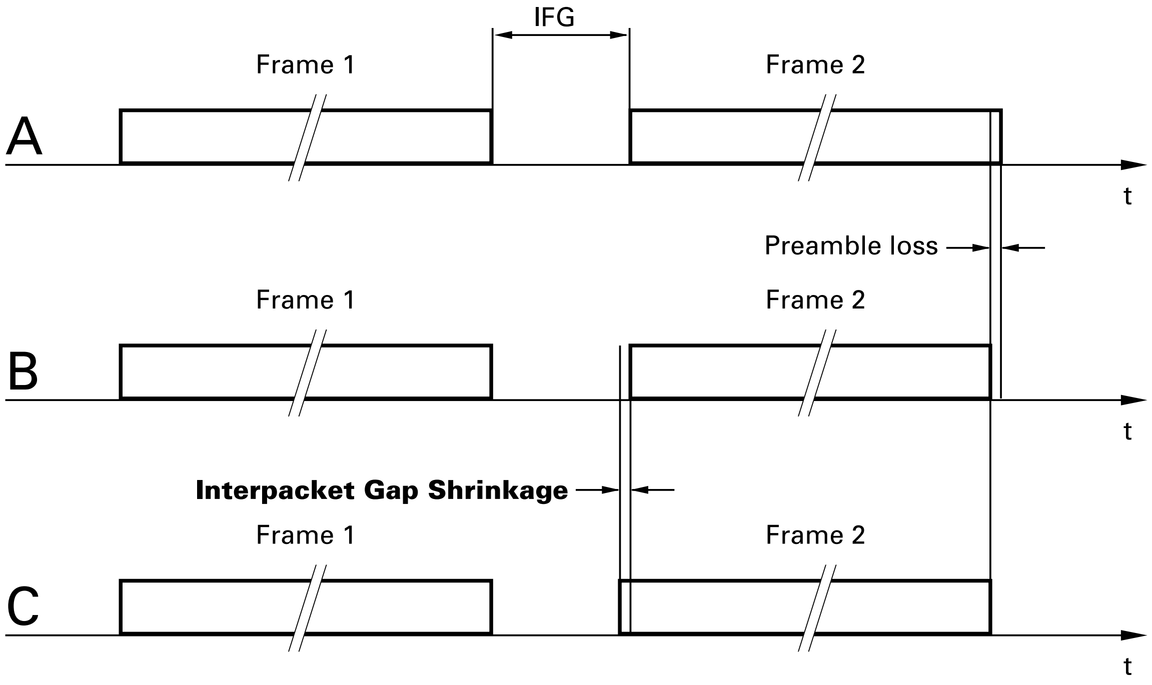


Fig. 7: Schematic representation of Interpacket Gap Shrinkage

1.2.4 Full duplex

In half duplex operation, the port of the switch is connected to an Ethernet. All rules prescribed by the CSMA/CD access method must be followed. For example, it is not possible to send and receive data at the same time, and, in order to be sure of detecting collisions, the propagation time is limited.

These restrictions are removed for full duplex operation. Instead of the well known bus structure, a point-to-point or bridge-to-bridge connection is used. Transmitting and receiving are conducted over two separate lines so that CSMA/CD rules can be ignored. Transmitting and receiving can take place at the same time at a particular port, which means that twice the bandwidth is available. By eliminating collisions it is possible to increase the effective data throughput by a factor of almost 10. The effective maximum through-put for Ethernet is 2 to 3 Mbit/s. Full duplex provides a bandwidth of 2 * 10 Mbit/s = 20 Mbit/s. The propagation time limitations needed for collision detection no longer apply. This makes it possible to extend networks to much greater distances than are possible with ordinary Ethernet connections.

Certain PC controller cards, such as that of Compaq and IBM, also supports this full duplex function, so that the 20 Mbit/s bandwidth can be achieved in a direct connection between these devices and a switch.

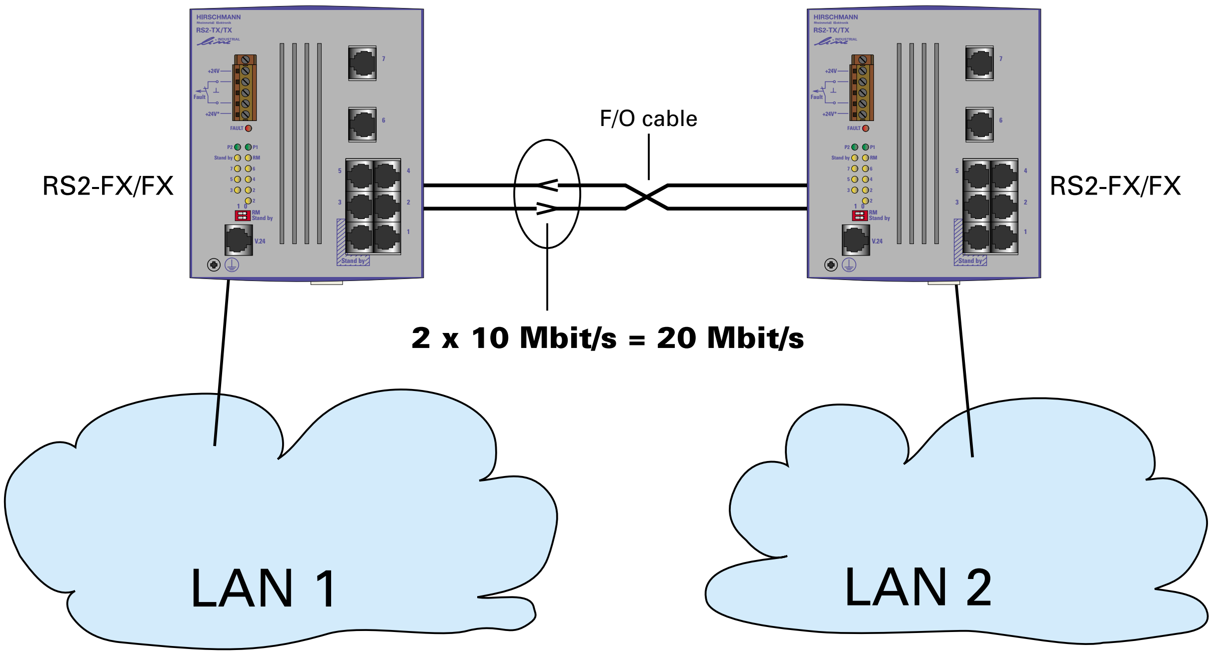


Abb. 8: Full duplex connection of two LANs

1.3 Frame structure

Within the scope of the CSMA/CD access principle, data frames specified in ISO/IEC 8802-3 are used for data transfer.

A **frame** is a data packet with a defined form and length that consists of signals in Manchester code. A frame is transferred serially with a data rate of 10 Mbit/second, whereby the individual bits are combined in octets (bytes of 8 bits each). All octets of one frame, with the exception of the Frame Check Sequence Field (FCS), are transferred with the least significant bit (LSB) first.

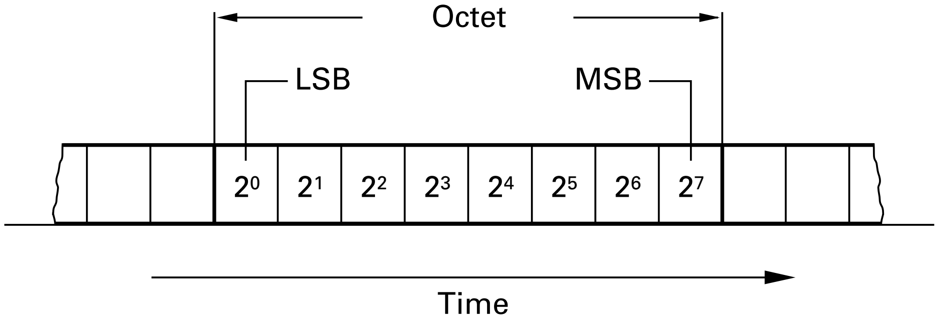


Fig. 9: Data transfer direction

A frame has a minimum length of 64 octets and a maximum of 1518 (see Fig. 10). The length of the frame is calculated without a preamble and the Start Frame Delimiter (SFD).

A frame consists of:

- **Preamble Field**
  The preamble, consisting of a string of alternate ones and zeros, serves to stabilize and synchronize the respective recipient to the incoming frame.

  Length: 7 octets (10101010...10101010)

- **Start Frame Delimiter Field**
  (SFD Field)
  The SFD marks the start of the destination address (Destination Address Field).

  Length: 1 octet (10101011)

- **Destination Address Field**
  Contains the frame's destination address.

  Length: 6 octets (48 bits)

- **Source Address Field**
  Contains the frame's source address.

  Length: 6 octets (48 bits)

- **Length/Type Field**
  The ISO/IEC 8802-3 standard uses this field as the length field to specify the number of octets in the subsequent data field that are to be transferred (values between 0 and 1518).
  Ethernet Version 2.0 specifies this field as the type field, which contains parameters specific to the manufacturer (values > 1518).

  Length: 2 octets (16 bits)

Data- und Pad Field

The user data is transferred in the data field.

Lengthmin:46 octets(368 bits)
max: 1500 octets(12 000 bits)

A data field that is less than 46 octets is filled up by a pad field containing additional octets.

Frame Check Sequence Field (FCS Field)

A check value is calculated on the basis of the contents of the previous fields (without the preamble and SFD) and is written into the FCS field. On the receiving end, a check value is also calculated on the basis of the same principle. This value agrees with the one in the FCS field if the data was transferred without errors occurring.

Length: 4 octets (32 bits)

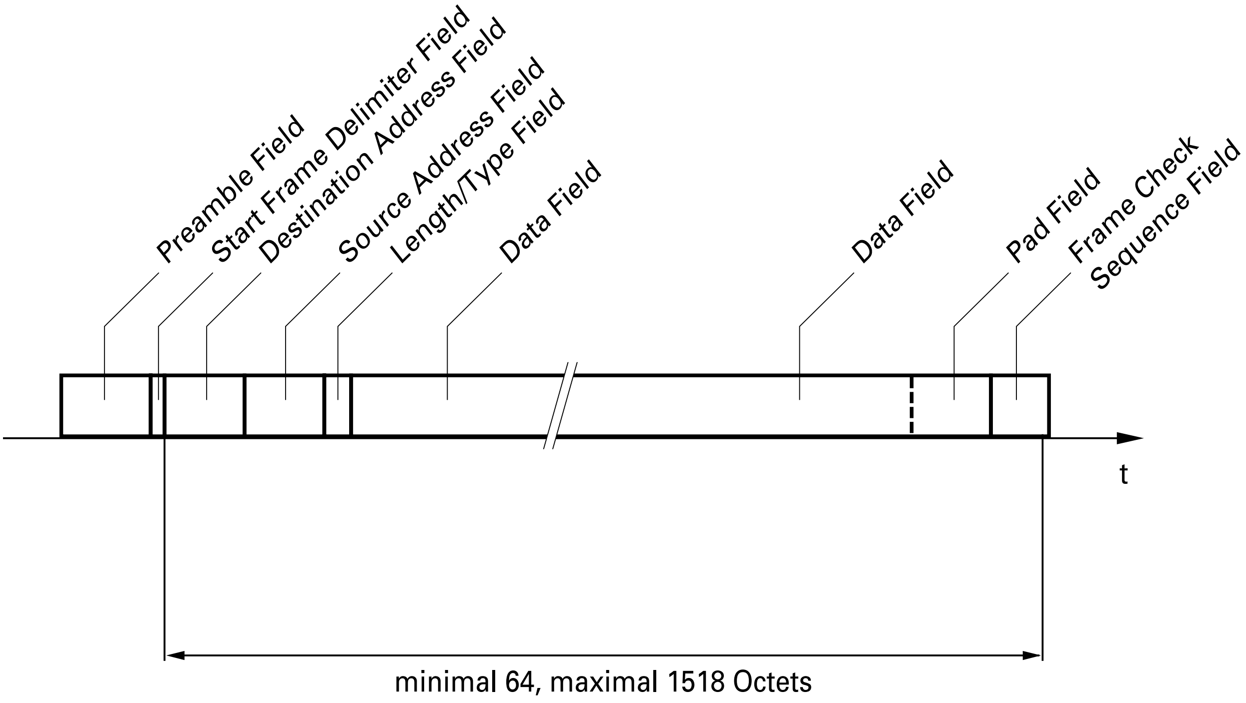

Preamble Field
Start Frame Delimiter Field
Destination Address Field
Source Address Field
Length/Type Field
Data Field
Data Field
Pad Field
Frame Check Sequence Field
t
minimal 64, maximal 1518 Octets

Fig. 10: Frame structure

1.4 IP address

The IP address consists of 4 bytes. These 4 bytes are written in decimal notation, each separated by a dot.

Since 1992, five classes of IP addresses have been defined in RFC 1340. The most frequently used address classes are A, B and C.

Class	network address	Host address
A	1 Byte	3 Bytes
B	2 Bytes	2 Bytes
C	3 Bytes	1 Byte

Table 2: IP address classification

The network address represents the permanent part of the IP address. It is as-signed by the DoD (Department of Defense) Network Information Center.

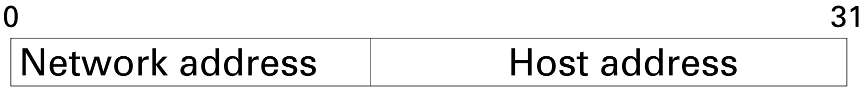


Fig. 11: Bit notation of the IP address

All IP addresses belong to **class A** when their first bit is a zero, i.e. the first decimal number is less than 128.
The IP address belongs to **class B** if the first bit is a one and the second bit is a zero, i.e. the first decimal number is between 128 and 191.
The IP address belongs to **class C** if the first two bits are a one, i.e. the first decimal number is higher than 191.

Assigning the host address (host id) is the responsibility of the network operator. He alone is responsible for the uniqueness of the IP addresses he assigns.

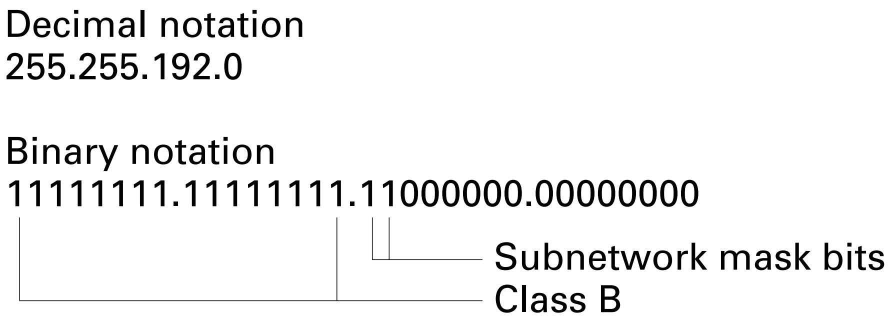
1.4.1 Network mask

Routers and gateways subdivide large networks into subnetworks. The network mask assigns the individual devices to particular subnetworks.

The subdivision of the network into subnetworks is performed in much the same way as IP addresses are divided into classes A to C (network id).

The bits of the host address (host id) that are to be shown by the mask are set to one. The other host address bits are set to zero in the network mask (see following example).

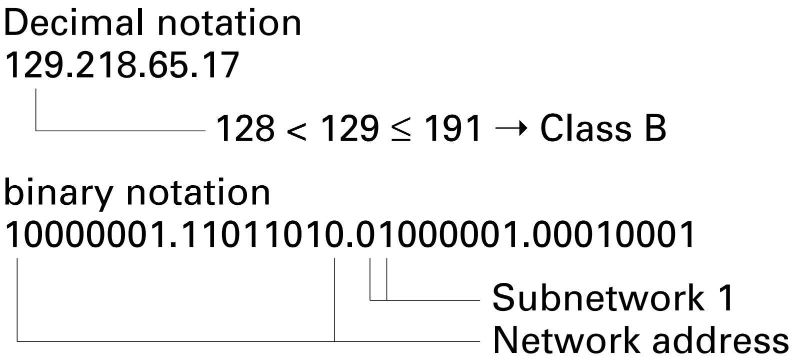
Example of a network mask:



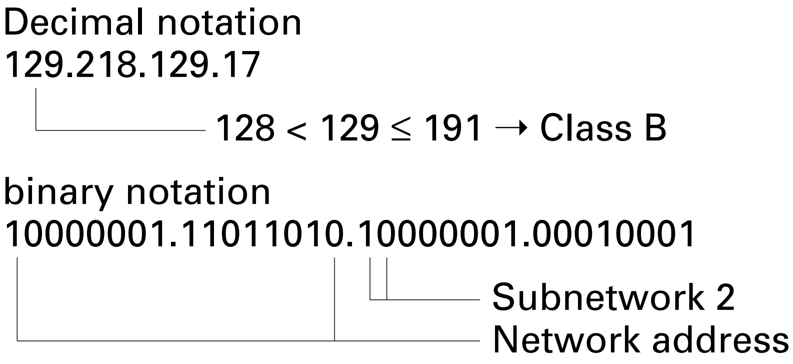
Decimal notation
255.255.192.0

Binary notation
11111111.11111111.11000000.00000000

Example of IP addresses with subnetwork allocation in accordance with the network mask from the above example:



Decimal notation
129.218.65.17
128 < 129 ≤ 191 → Class B
binary notation
10000001.11011010.01000001.00010001
Subnetwork 1
Network address



Decimal notation
129.218.129.17
128 < 129 ≤ 191 → Class B
binary notation
10000001.11011010.10000001.00010001
Subnetwork 2
Network address

1.4.2 Example of how the network mask is used

In a large network it is possible that gateways and routers separate the management agent from its management station. How does addressing work in such a case?

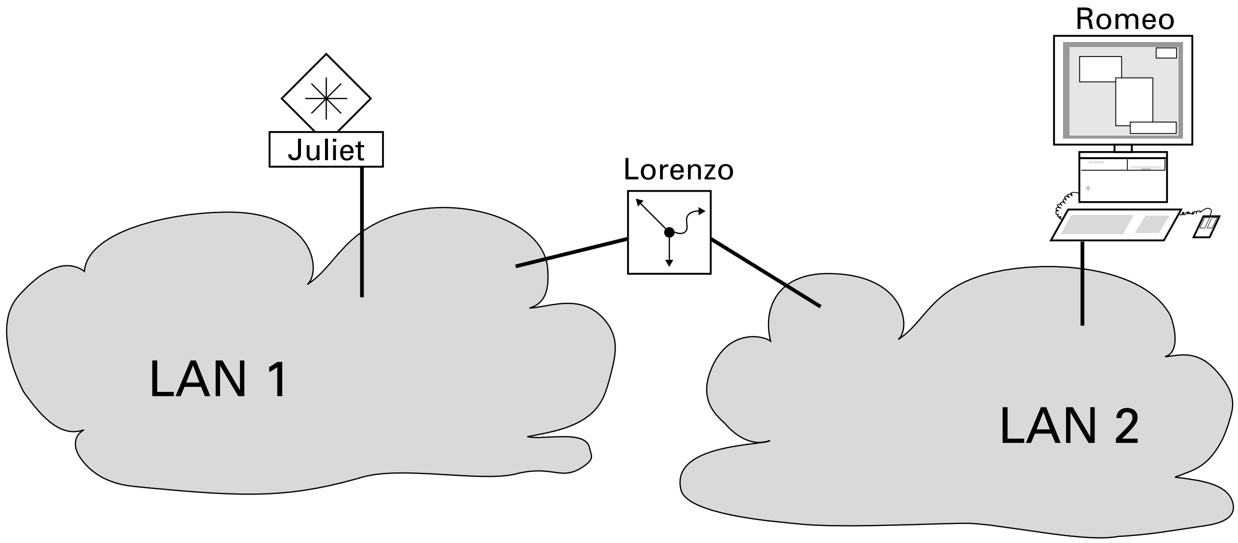


Fig. 12: Management agent that is separated from its management station by a router

The management station ”Romeo” wants to send data to the management agent ”Juliet”. Romeo knows Juliet's IP address and also knows that the router ”Lorenzo” knows the way to Juliet.

Romeo therefore puts his message in an envelope and writes Juliet's IP address on the outside as the destination address. For the source address, he writes his own IP address on the envelope.

Romeo then places this envelope in a second one with Lorenzo’s MAC address as the destination and his own MAC address as the source. This process is comparable to going from Layer 3 to Layer 2 of the ISO/OSI base reference model.

Finally, Romeo puts the entire data packet into the mailbox. This is comparable to going from Layer 2 to Layer 1, i.e. to sending the data packet via the Ethernet.

Lorenzo receives the letter and removes the outer envelope. From the inner envelope he recognizes that the letter is meant for Juliet. He places the inner envelope in a new outer envelope and searches his address list (the ARP ta-- ble) for Juliet’s MAC address. He writes her MAC address on the outer envelope as the destination address and his own MAC address as the source address. He then places the entire data packet into the mail box.

Juliet receives the letter and removes the outer envelope, exposing the inner envelope with Romeo’s IP address. Opening the letter and reading its con-tents corresponds to transferring the message to the higher protocol layers of the ISO/OSI layer model.

Juliet would now like to send a reply to Romeo. She places her reply in an envelope with Romeo’s IP address as destination and her own IP address as source. The question then arises, where should she send the letter, since she did not receive Romeo’s MAC address. It was lost when Lorenzo replaced the outer envelope.

By comparing her IP address to Romeo's with the aid of the **network mask**, Juliet would immediately recognize that Romeo lives nowhere close by, and that to call out the window would be pointless.

In the MIB, Juliet finds Lorenzo listed under the variable `hmNetGatewayIPAddr` as a means of communicating with Romeo. The envelope with the IP addresses is therefore placed in a further envelope with the MAC destination address of Lorenzo.

The letter then travels back to Romeo via Lorenzo the same way the first letter traveled from Romeo to Juliet.

2 Network Planning

The collision mechanism of an ISO/IEC 8802-3 LAN makes it necessary to limit signal delay value (see “The ISO/IEC 8802-3 Standard” on page 21). As a consequence, the physical size of the network is also limited. The signal delay value limitation means that the distance between any two stations in the network cannot exceed 4520 meters. ISO/IEC 8802-3 allows a maximum distance of only 2500 meters, however. This reduction is due to the delays introduced by the transmission components, primarily the repeaters.

The path variability value (see “Variability” on page 59) is just as important for correct network functioning as signal delay value. Until now, it was necessary to limit the number of repeaters to four in order to guarantee a minimum interpacket gap size. The limitation of four repeaters was dropped from the standard with the publication of Chapter 13 of ISO/IEC 8802-3. Instead of limiting the number of repeaters in a network, Chapter 13 specifies the maximum amount by which the interpacket gap can shrink and how this amount can be calculated for a particular signal path.

If a network requires more repeaters, Hirschmann repeaters are available that help you overcome this barrier. When passing through the repeater, the distance between packets shrinks by a smaller amount than permitted by the standard (see “Variability” on page 59).

2.1 Planning Rules

2.1.1 Planning Guidelines for 10 MBit/s Ethernet

When CSMA/CD networks were first being standardized, ISO/IEC 8802-3 limited itself to specifying standards that used the thick yellow coaxial cable (10BASE5 or yellow cable) as a transmission medium. The characteristics of this cable allow a maximum segment length of 500 meters.

Individual network stations are connected to the cable by transceivers. A maximum of 100 transceivers with a minimum spacing of 2.5 meters can be connected to a segment. A transceiver cable (AUI cable) with a maximum length of 50 meters connects the network station to the transceiver.

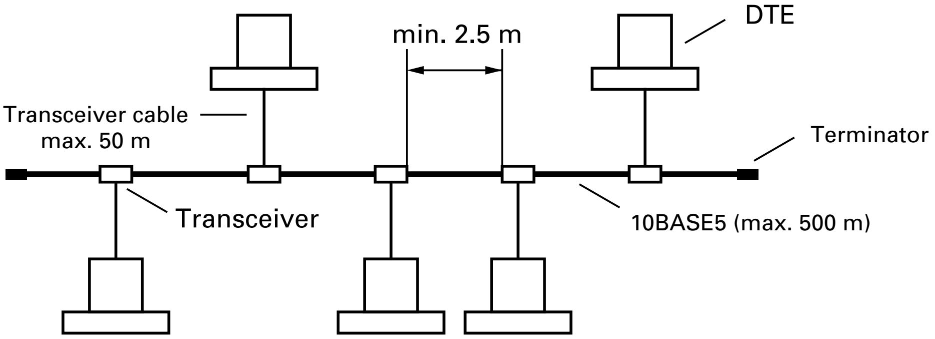


Fig. 13: Ethernet base segment as specified in ISO/IEC 8802-3

In order to expand the network, the norm prescribes the use of repeaters for coupling two segments together.

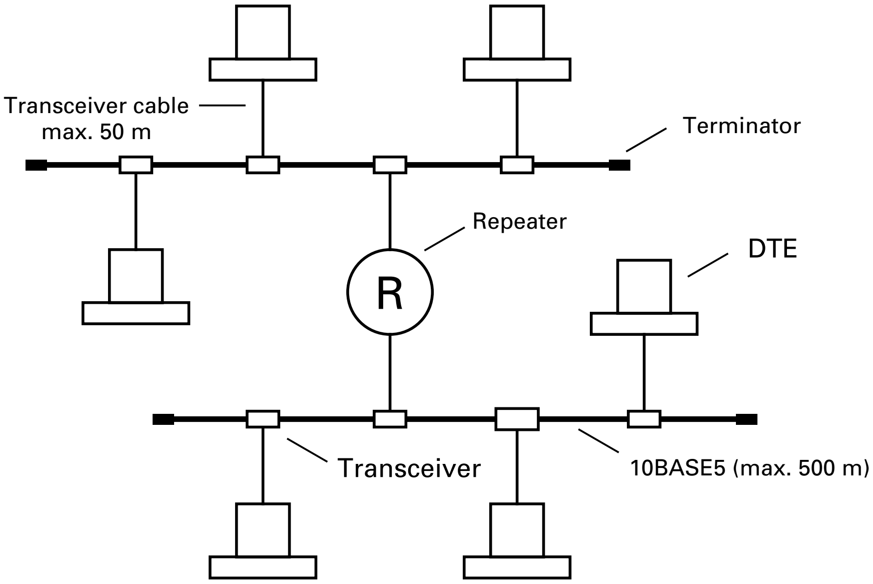


Fig. 14: Two basic Ethernet segments linked with repeaters

The delay value through a repeater is approximately the same as the delay value through a 500 meter coax segment.

This means that a signal path can contain a maximum of **four repeaters** and **five coax segments**. Of the five coax segments, at least two must be pure connecting segments (**link segments**) to which no stations are attached.

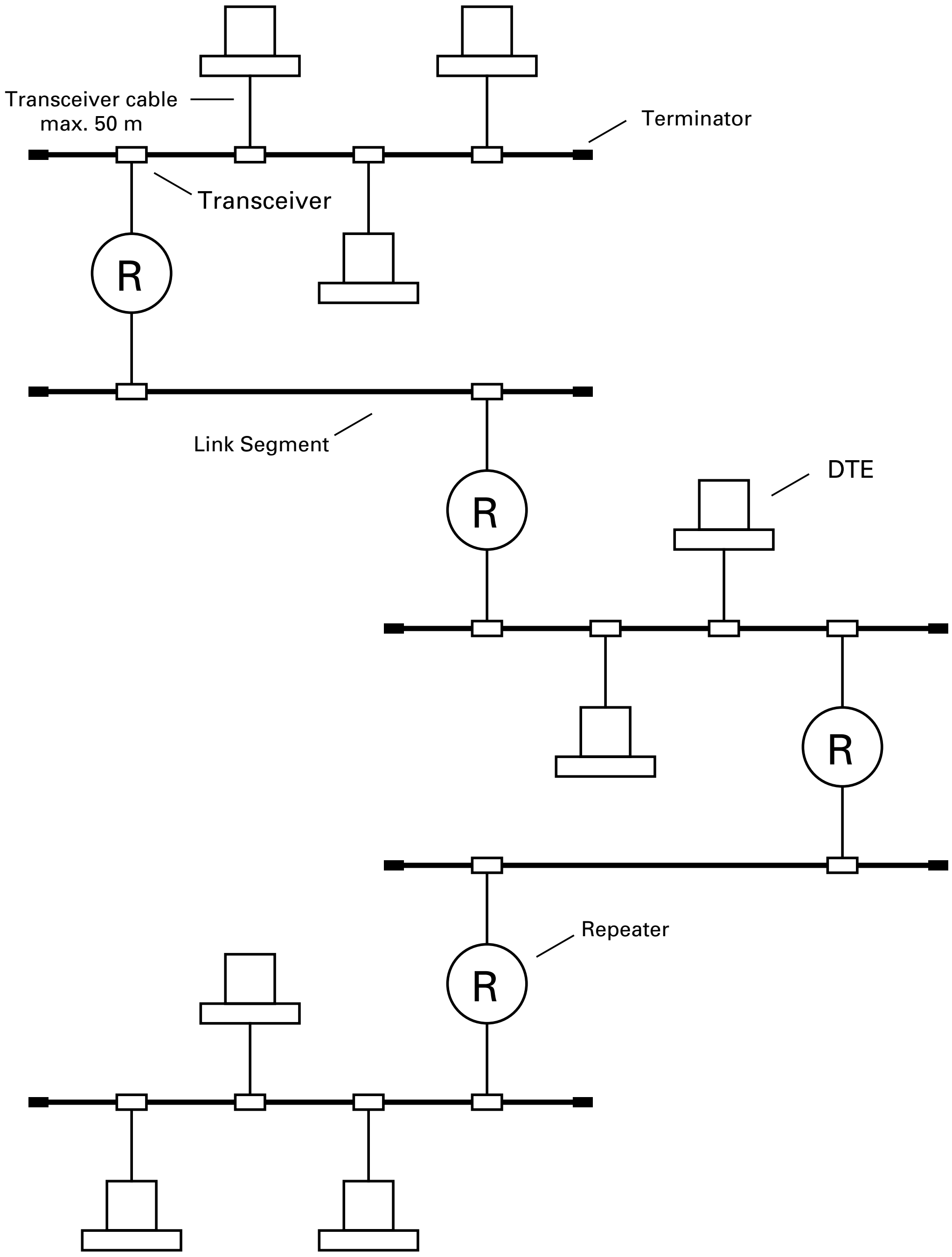


Fig. 15: Maximum signal path with repeaters

Because the thick yellow coax cable was too costly and too difficult to handle, the standard was extended to include the RG 58 coax cable. This was named the **Cheapernet** or thin wire Ethernet (10BASE2).

With a few restrictions and changes, the general specifications for the old standard were also applied to the newer standard. The trans-mission quality of this cable allows a maximum segment length of only **185 meters**. Also, each segment can have a maximum of only **30 transceivers** attached with a minimum spacing of **0.5 meters**.

It is possible to mix standard Ethernet and Cheapernet in a single configuration.

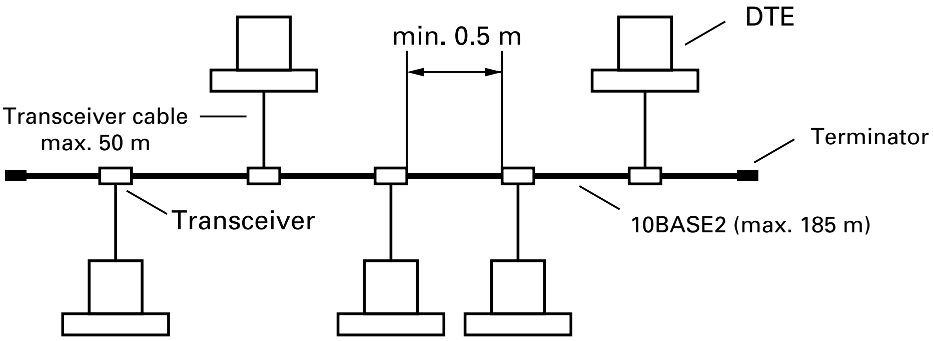


Fig. 16: Cheapernet base segment

The increased popularity of this LAN type led to user requirements for more flexible connection capabilities:

- thinner cable diameters
- lower cost cable
- use of already installed telephone wiring
- use of already installed IBM Type 1 cabling
- better transmission characteristics
- better interception security
- reduction of the problems due to potential differences
- greater distance
- resistance to electromagnetic interference

The standards bodies met these demands by extending the standard to 10BASE-T for twisted pair cables and 10BASE-F for fiber optic cables. In contrast to the bus connection provided by 10BASE5 and 10BASE2, these two new cable types are able to offer pure point-to-point connections.

With the standards as a framework, Hirschmann developed its network concept based on star distribution points, namely the active star couplers. Interface cards which can be plugged into the star couplers are available for all different transmission media, thus making it possible to operate a mixed network. The right medium is available for every requirement.

Characteristic	10BASE2	10BASE5	10BASE-F	10BASE-T
Maximum cable length	185 m	500 m	2000 m	100m
Termination	50 Ω	50 Ω	–	100 Ω
Maximum transceivers	30	50	2	2
Minimum transceiver spacing	0,5 m	2,5 m	–	–
Signal velocity	0,65 * c	77 * c	66 * c	59 * c

Table 3: The most important parameters for the media allowed by ISO/IEC 8802-3

2.1.2 Planning Guidelines for 100 MBit/s Ethernet

Using half duplex segments the propagation delay between the terminal equipments is 512 bit times (BT) maximum. Add all components of the signal path plus a safety margin.

Components	Delay
RT2-TX/FX	84 BT
Class II Repeater	92 BT
Terminal equipment with TP connection	50 BT
Terminal equipment with F/O connection	50 BT
Kat. 5 TP cable	1,112 BT/m
F/O cable	1 BT
Savety margin	4 BT

Table 4: Signal delay in the sgnal path

2.1.3 Planning Guidelines for 100 MBit/s Ethernet

Using 1000 Mbit/s Ethernet generally the terminals are connected with Switches directly. Thus the values in Table 5, “Length area in dependency of F/O at 850 nm,” on page 48 and Table 6, “Length area in dependency of F/O at 1300 nm,” on page 48 apply.

F/O type	bandwith length product	minimum distance	maximum distance
62,5 nm multimode F/O	160 MHz * km	2 m	220 m
62,5 nm multimode F/O	200 MHz * km	2 m	275 m
50 nm multimode F/O	400 MHz * km	2 m	500 m
50 nm multimode F/O	500 MHz * km	2 m	550 m

Table 5: Length area in dependency of F/O at 850 nm

F/O type	bandwith length product	minimum distance	maximum distance
62,5 nm multimode F/O	500 MHz * km	2 m	550 m
50 nm multimode F/O	400 MHz * km	2 m	550 m
50 nm multimode F/O	500 MHz * km	2 m	550 m
10 nm singlemode F/O	–	2 m	5000 m

Table 6: Length area in dependency of F/O at 1300 nm

2.2 Maximum Network Size

2.2.1 Hub Networks

Chapter 13 of ISO/IEC 8802-3 describes the system requirements for a local area network that uses mixed transmission media. The standard assumes, however, that the communications components used exploit the tolerances that they are allowed.

The model 1 transmission system of Chapter 13 conforms basically with the currently valid configuration guidelines and only extends them with regard to the media that can be used. This model is un-suitable, however, if one wants to test the maximum limits of what is possible. For such a case, Model 2 provides a much more precise description of how to calculate the maximum network range.

Model 2 takes the delay values of all the network components in the signal path into consideration. Considering all delay values is a very complex task, so Model 2 uses a simplification and defines fixed delay values for the individual segments. The disadvantage of this simplification is the invariance of the delay values for the various segments.

The following model for calculating the maximum network range is derived from Model 2. It has been optimally tailored to calculate a LAN made up of Hirschmann network components. Just like Model 2, it includes all network components found in the signal path. Only the form of the simplification has been changed, which leads to a much more precise calculation of maximum network range.

The basis for calculating the maximum network range is the maximum allowable signal delay value in the signal path between any two network stations. The critical case most frequently encountered can be found in the following situation (see Fig. 17).

Station 1 transmits data to station 2, which is quite close. The data is then sent into the rest of the network. Just before the data reaches station 3, station 3 starts to send data to station 1 (cf. Appendix B, System Guidelines, B1.1 IEEE Std. 802.3-1985 Baseband Systems). Because station 3 must be sure to detect the data collision, the maxi-mum distance between stations 1 and 3 is 4520 meters given the standard minimum packet length and delay value through an ideal transmission link (fiber optic only).

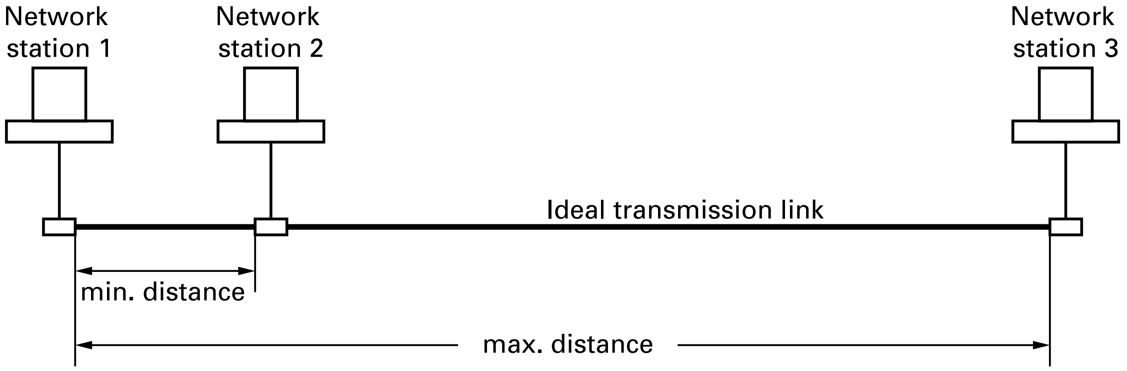


Fig. 17: Critical case for maximum network range

Signal delays in the individual components of the signal path form a significant part of the total signal delay value. The delay value for a component is a simple way to determine the effect the delay value through the component has on the maximum range of the network.

Definition of "Propagation Equivalent":
The propagation equivalent describes the signal delay of a component located in the signal path. The signal delay is specified in terms of distance (meters) rather than time (seconds). The specifi-cation in meters indicates the distance that the signal could have traveled in the same time if it had been moving through a cable instead of the component.

Note: The conversion from time units to distance units assumes a cable propagation delay of 5 ns/m. A UTP cable has a signal delay of 5.6 ns/m(See “The most important parameters for the media allowed by ISO/IEC 8802-3” on page 47.)

□ In order to determine if you comply with the standard, calculate the signal propagation time between the two members of the network that are farthest apart from one another.
Add up the values for all of the components within the signal path. The table on page 47 lists all of the components belonging to the Hirschmann network concept. Signal delay for each component is specified in terms of the "propagation equivalent".
This total plus the length of the cable in the signal path must not exceed 4520 meters. In order to correctly compensate for the slower propagation velocity in UTP cable segments, you should add 10% to the length of those cables..

$$n_1 * Ü_1 + ... + n_x–1 * Ü_x–1 + n_x * Ü_x + \Sigma l \leq 4520\ m$$

n_x = Number of ports in the signal path of the transmission components with the index x

$Ü_x$ = propagation equivalent of a transmission component with the index x

Σl = Sum of the lengths of all segments in the signal path

Example 1:
The figure (see Fig. 18) shows a local area network that consists of twisted pair segments.
Seven Rail Hubs RH1-TP/FL lie in the signal path between the two network stations shown in the figure.

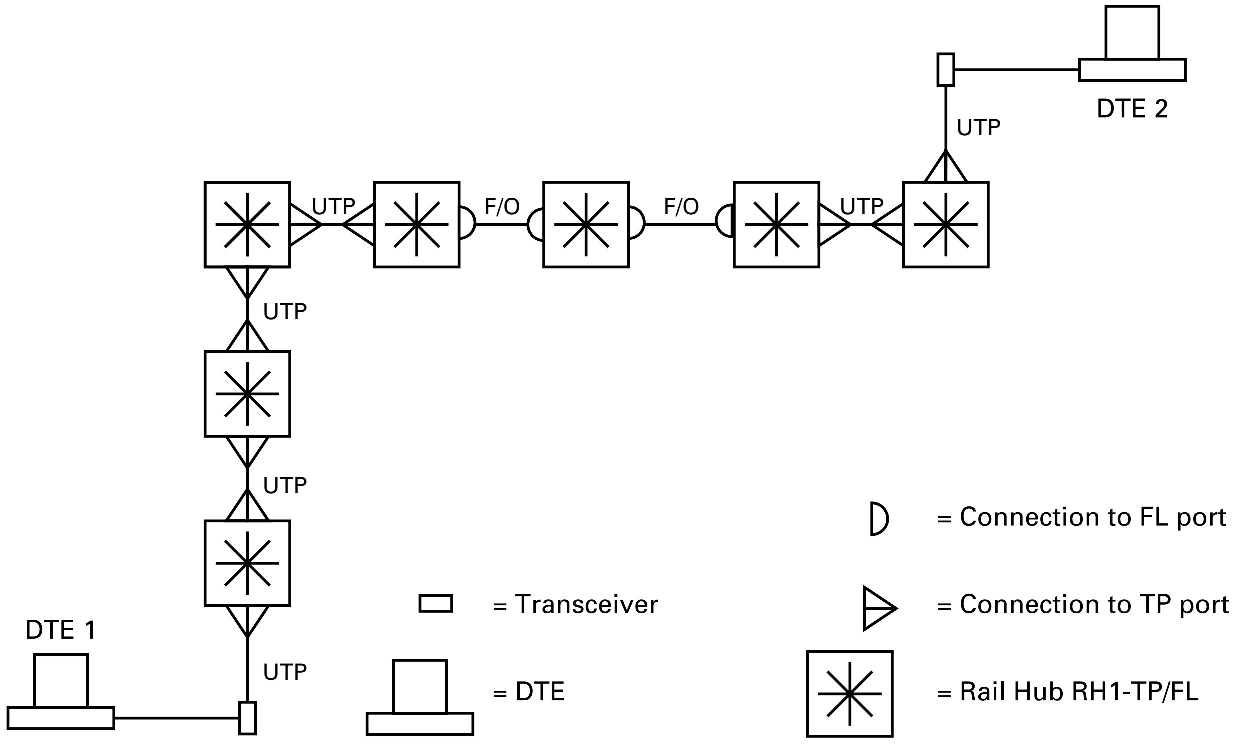


Fig. 18: Example of a reduced network range

The propagation equivalent for the interface cards and transceivers is calculated as follows:

DTE 1		
Transceiver	Mini-UTDE	140 m
Rail Hub 1	TP	95 m
	TP	95 m
Rail Hub 2	TP	95 m
	TP	95 m
Rail Hub 3	TP	95 m
	TP	95 m
Rail Hub 4	TP/FL	180 m
	TP/FL	180 m
Rail Hub 5	FL	130 m
	FL	130 m
Rail Hub 6	TP/FL	180 m
	TP/FL	180 m
Rail Hub 7	TP	95 m
	TP	95 m
Transceiver	Mini-UTDE	140 m
DTE 2		
		Σ 2020 m

The total number of cable segment lengths may amount to

4520 m – 2020 m = 2500 m.

Example 2:
The figure (see Fig. 19) shows a LAN whose coax segments are connected together by fiber optic cables and star couplers. There are 5 star couplers equipped with 2 KYDE-S µC coax interface cards and 8 OYDE-S µC optical interface cards located between the network stations.

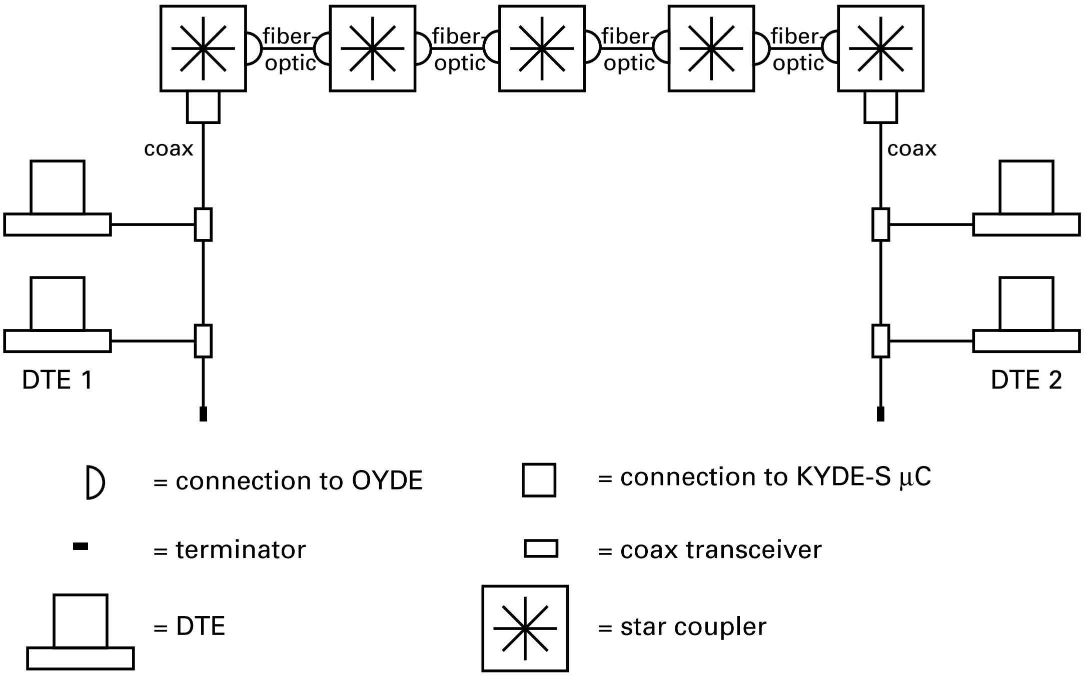


Fig. 19: Coax segments connected with fiber optic segments

The propagation equivalent for the interface cards and transceivers is calculated as follows:

DTE 1		
Transceiver	KTDE-S	205 m
Star coupler 1	KYDE-S µC	50 m
	OYDE-S µC	40 m
Star coupler 2	OYDE-S µC	40 m
	OYDE-S µC	40 m
Star coupler 3	OYDE-S µC	40 m
	OYDE-S µC	40 m
Star coupler 4	OYDE-S µC	40 m
	OYDE-S µC	40 m
Star coupler 5	OYDE-S µC	40 m
	KYDE-S µC	50 m
Transceiver	KTDE-S	205 m
DTE 2		
		Σ 830 m

The total of the cable segment lengths may amount to

4520 m – 830 m = 3690 m.

Example 3:
The figure (see Fig. 20) shows a mixed LAN consisting of coax, fiber optic and twisted pair segments. Because more than five star couplers are cascaded in the network, it is necessary to use a repeater. Interface cards with clock regeneration - in this case the ECFL2 - realise all repeater functions.

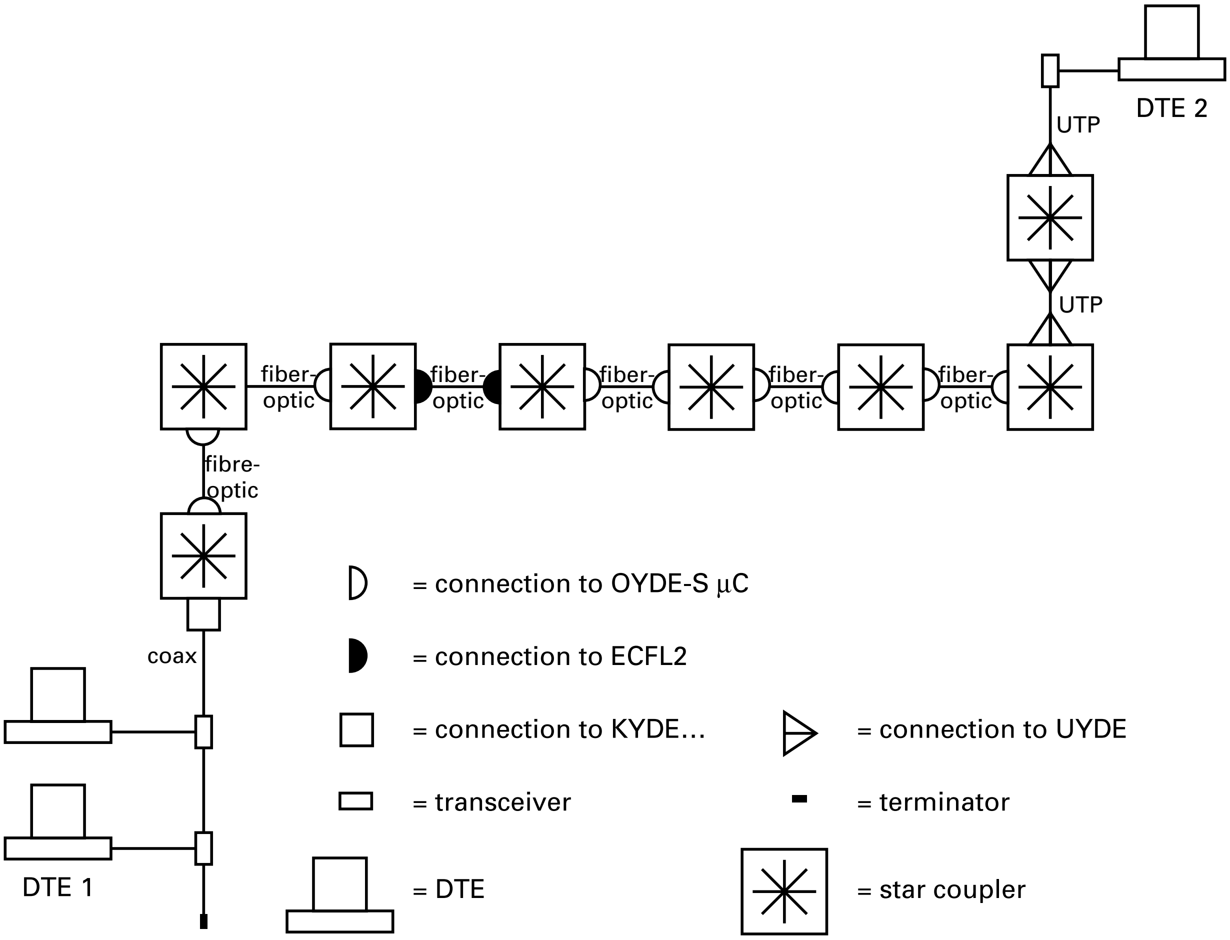


Fig. 20: Cascading of more than five star couplers possible using a retiming path

The propagation equivalent for the interface cards and transceivers is calculated as follows:

DTE 1		
Transceiver	KTDE-S	205 m
Star coupler 1	KYDE-S µC	50 m
	OYDE-S µC	40 m
Star coupler 2	OYDE-S µC	40 m
	OYDE-S µC	40 m
Star coupler 3	OYDE-S µC	40 m
	ECFL2	170 m
Star coupler 4	ECFL2	170 m
	OYDE-S µC	40 m
Star coupler 5	OYDE-S µC	40 m
	OYDE-S µC	40 m
Star coupler 6	OYDE-S µC	40 m
	OYDE-S µC	40 m
Star coupler 7	OYDE-S µC	40 m
	UYDE	170 m
Star coupler 8	UYDE	170 m
	UYDE	170 m
Transceiver	Mini-UTDE	140 m
DTE 2		
		Σ 1645 m

The total of the cable segment lengths may amount to

4520 m – 1645 m = 2875 m.

Network components	Propagation equivalent	Propagation time	Medium
OYDE-S µC, ECSM1	40 m	4 BT	10BASE-FL
ECFL2	170 m	17 BT	
ECFL4	130 m	13 BT	
RH1-TP/FL for FL-FL link	130 m	13 BT	
RH1-TP/FL for TP-FL link	180 m	18 BT	
RT1-TP/FL	50 m	50 BT	
Optical Transceiver	100 m	10 BT	
KYDE-S µC	50 m	5 BT	10BASE5
Coax Transceiver	205 m	20,5 BT	
CYDE	210 m	21 BT	10BASE2
Coax Transceiver	205 m	20,5 BT	
ECTP3	120 m	12 BT	10BASE-T
UYDE	170 m	17 BT	
RH1-TP	95 m	9,5 BT	
RH1-TP/FL for TP-TP link	95 m	9,5 BT	
RH1-TP/FL for TP-FL link	180 m	18 BT	
RT1-TP/FL	50 m	5 BT	
Twisted-Pair Transceiver	140 m	14 BT	
ECAUI	165 m	16,5 BT	MAU

Table 7: Length and propagation time table per port

2.2.2 Switch networks

Switch networks can be virtually extended to no end.

Possible limiting factors for extending them are:

- Response times
  Some user applications expect an answer from another partner within a specified time period.
- Redundancy protocols:
  In redundant networks, the connected switches exchange status information by means of a redundancy protocol. In the event of an error, this information must be exchanged within a certain period of time so that the network can be correctly reconfigured.
  When using up to 50 Rail Switches in a redundant ring, the reconfiguration time is less than one-half of a second.

2.3 Variability

Just as it is necessary to check the signal run time, there is also a need to check the **P**ath **V**ariability **V**alue (PVV).

The PVV along the path between network stations must not exceed 49 BT.

Definition of the variability value:
The run time (start up delay) of a data packet through a component fluctuates from one packet to another. The amount of this fluctuation is the variability value of this component.

Definition of the path variability value:
The total of the variability values of all components along a data path between two network stations is the PVV.

Suggestions for calculation of the PVV can be found in Chapter 13 of ISO/ IEC 8802-3. This standard defines upper limits for various kinds of components (e.g. coax and fiber optic etc.). To some extent, components from Hirschmann possess narrower tolerance limits than the upper limits specified by the standard.
This is why high cascading depths can be achieved with transmission components from Hirschmann.

On the basis of 49 bit times, the PVV can be calculated as follows.

1 The influence of the clock tolerance,

- the variability value of the first MAU
- transmit start-up delay variability + transmit start-up delay variability correction)

and

- a safety reserve

reduce the budget for the remaining transmission components.

Clock tolerance (Clock Skew)	2,5 BT
Transmit Start-up Delay Variability	2,0 BT
Transmit Start-up Delay Variability Correction	1,5 BT
Reserve	3 BT
	9 BT

The transceiver connected to the second network station does not contribute towards shrinkage of the packet interval. A value of 40 BT remains as the budget for the other transmission components in the signal path

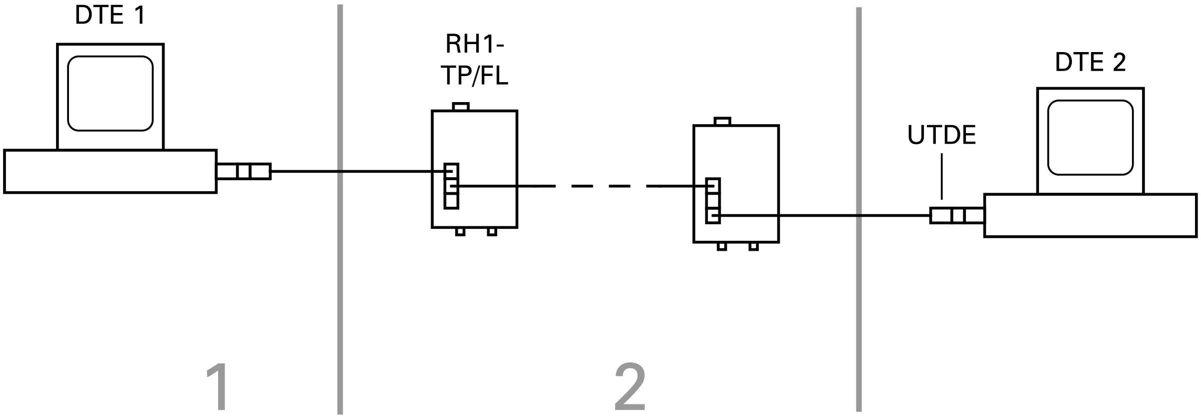


Fig. 21: Calculating the PVV

2 Add up the bit times

– of the interface card pairs from the table (see table 8 on page 61) and

– of the components from the table (see table 8 on page 61)

that are located in the signal path between a network station and the last repeater before any other network station. These are the components that lie between the grey lines in the figure (see Fig. 21).

	OYDE-S µC	KYDE-S µC	CYDE	ECAUI	ECFL4
OYDE-S µC	2	2	4	2	3
KYDE-S µC	-	2	4	2	3
CYDE	-	-	5	4	6
ECAUI	-	-	-	2	3
ECFL4	-	-	-	-	3

Table 8: Variability value in bit times for interface card pairs

	Variability Value
RH1-TP (TP ÷ TP)	3 BT
RH1-TP/FL (TP ÷ TP)	3 BT
RH1-TP/FL (TP ÷ FL)	6 BT
RH1-TP/FL (FL ÷ FL)	3 BT
RT1-TP/FL	3 BT
Mini OTDE	2 BT
KTDE-S	6 BT
Mini KTDE	6 BT
Mini UTDE	2 BT

Table 9: Variability value in bit times for single components

Example

The path variability value for the example in the figure (see Fig. 22) is calculated, starting from DTE 1, as follows:

Pair CYDE/ECFL4	6 BT
RH1-TP/FL, FL to TP	6 BT
RH1-TP/FL, TP to TP	3 BT
RH1-TP	3 BT
RH1-TP	3 BT
RH1-TP/FL, TP to FL	6 BT
RT1-TP/FL	1 BT
	28 BT

Thus: $28\ \text{BT} \leq 40\ \text{BT}$

Note: Exclusive this components which are above the grey lines contribute to the PVV. The both transceivers below the grey lines are just taken into consideration within the budget and are not longer calculated to the PVV.

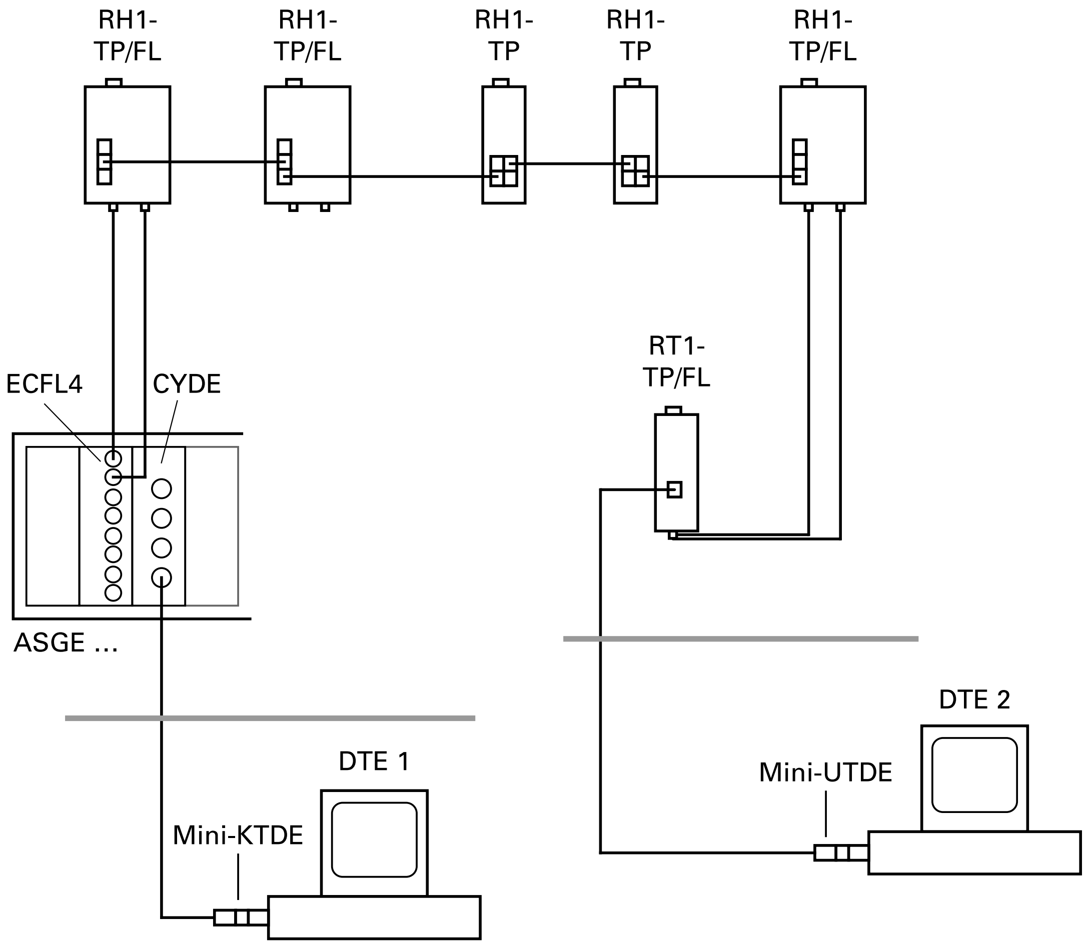


Fig. 22: Example of calculating the PVV

2.4 Redundancy

There are particularly critical areas in which data security is assigned absolute priority. To circumvent any possible failure of the transmission medium or of a concentrator in such areas, a standby line is frequently laid in a separate cable line. The interface cards and units featuring a redundancy function enable automatic changeover between one main line and a standby line.

Depending on the card/unit type, the following redundancy modes are available:

- Normal mode ('as-delivered' setting)
- Frame redundancy
- Switch redundancy

2.4.1 Normal mode ('as-delivered' setting)

Standard operation is realised between two normally linked interface cards. Such a connection represents a part of the main link through which data communication takes place during regular operation.

2.4.2 Frame redundancy (10 Mbit/s Ethernet)

Redundancy mode is based on monitoring the data flow within a network structure featuring a redundant design.
Redundancy permits the creation of networks structured in a ring. The occurrence of one single error can be bypassed.

Rules for creating redundant structures

1 **Components for links featuring a redundant structure:**
Links safeguarded by redundancy must contain only the following components from Hirschmann:

- RH1-TP/FL (F/O ports)
- ECFL4.

2 **Ring with redundancy mode:**
A ring is produced through a cross-link within the bus structure. The link in redundancy mode RM in the figure (see Fig. 23) represents this cross-link.
The advantage of this structure is that, with the aid of the redundant link, in the event of failure of a star coupler link or of a star coupler itself every other star coupler remains accessible.

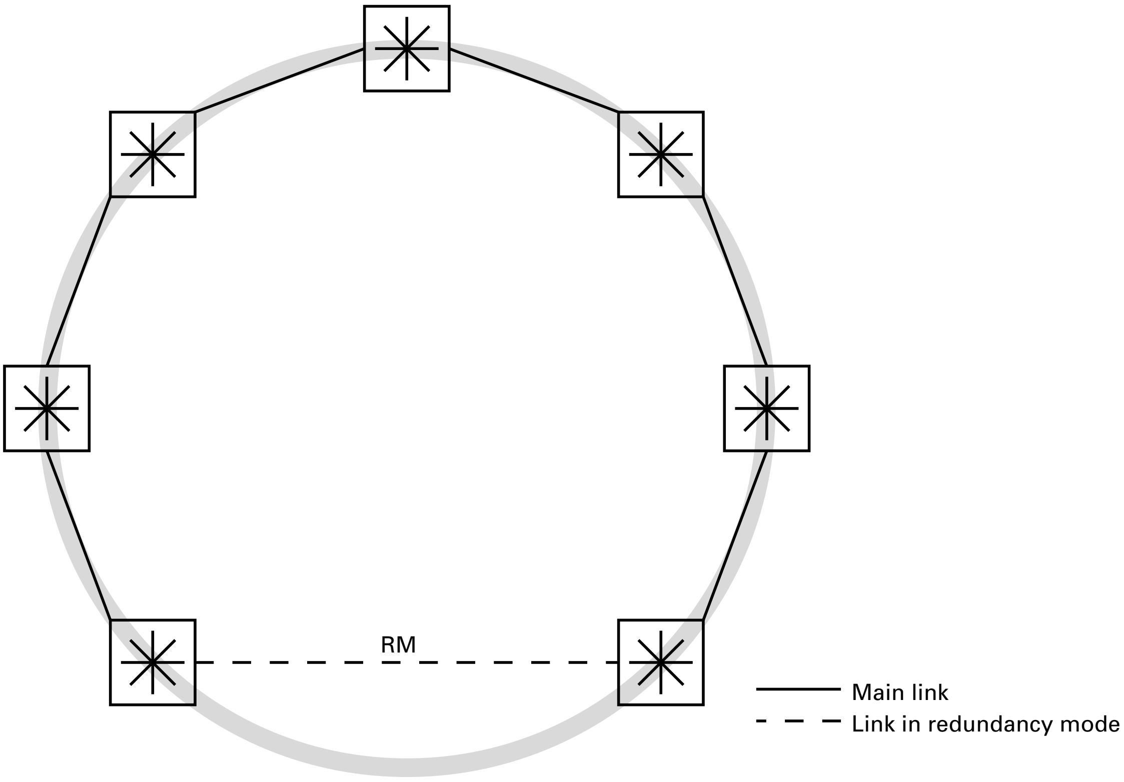


Fig. 23: Example of a singly redundant ring

3 **Network size:**
In redundantly structured networks, the failure of links results in new network topologies.

□ With regard to every conceivable network topology, check whether the largest distance between two network stations is less than the maximum network size
(cf. “Maximum Network Size” on page 49).

4 **Variability:**
In redundantly structured networks, the failure of links results in new network topologies.

□ With regard to every conceivable network topology, check whether the PVV between two network stations assumes a permissible value (cf. “Variability” on page 59).

5 If Rules 1 to 4 have been obeyed, then any number of redundant transmission links may occur in one network.

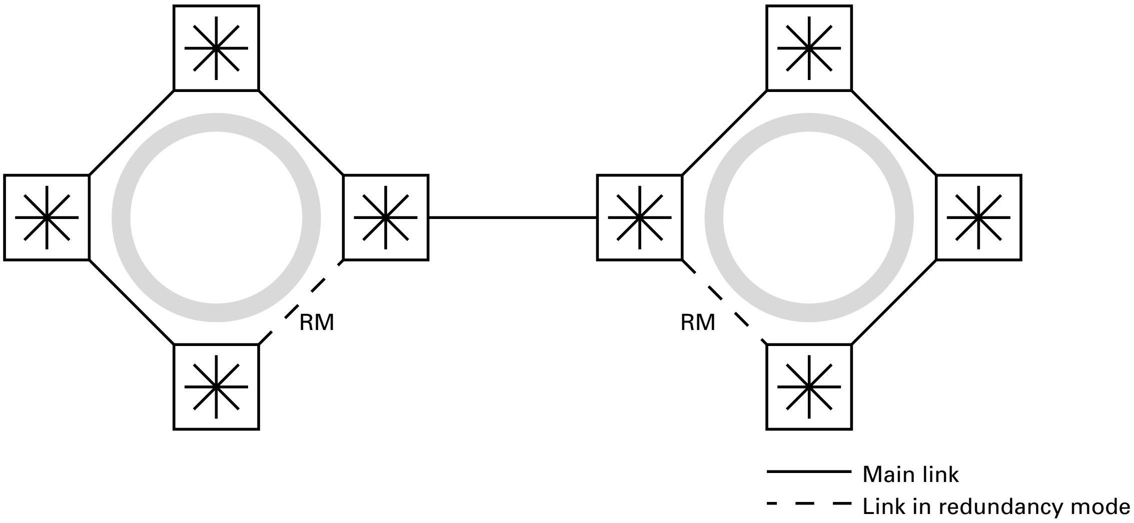


Fig. 24: Example of redundant links conforming to rule 6

2.4.3 Switch redundancy (100 Mbit/s Ethernet)

Line configuration

The RS2-../.. enables the setup of backbones in the line configurations. Cascading takes place via the backbone ports.

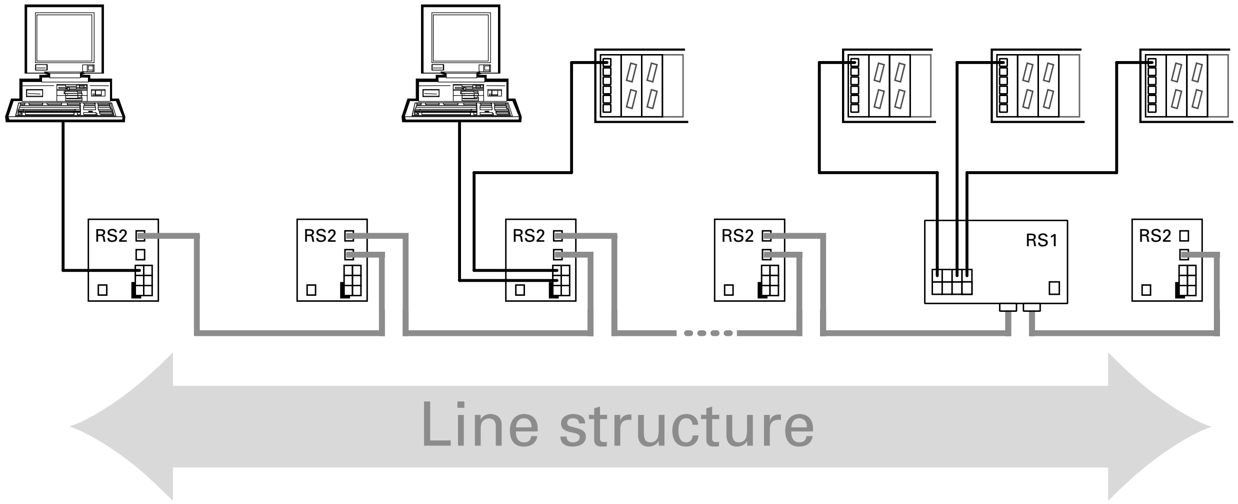


Fig. 25: Line configuration

Redundant ring structure

The two ends of a backbone in a line configuration can be closed to form a redundant ring by using the RM function (**R**edundancy **M**anager) of the RS2-../.. or RM1.

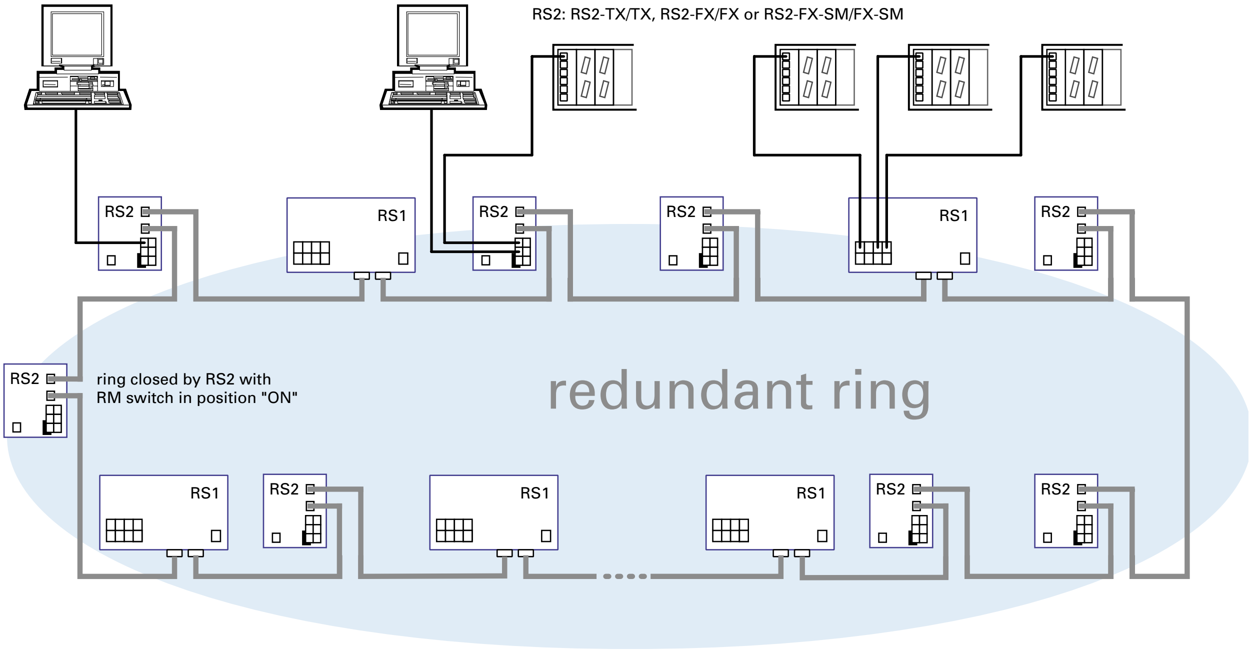


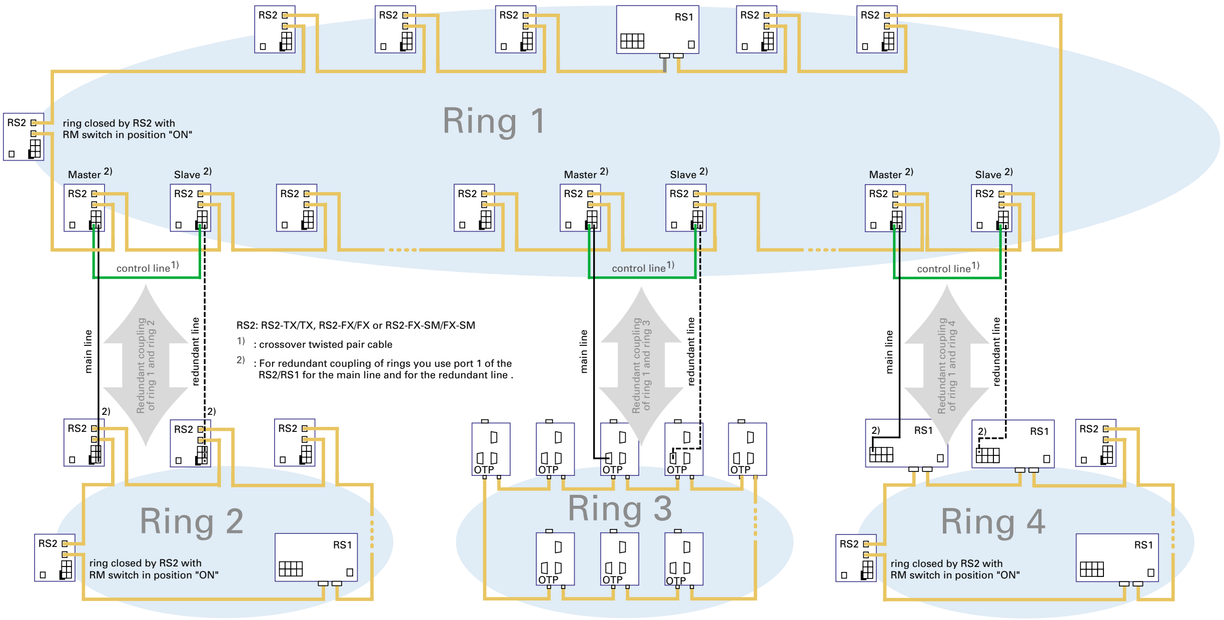
Fig. 26: Redundant ring structure

The RS2-../.. is integrated into the ring via the backbone ports (ports 6 and 7). It is possible to mix the RS1 and RS2-../.. in any combination within the redundant ring. If a line section fails, the ring structure of up to 50 RS1/ RS2-../.. transforms back to a line configuration within 0.5 seconds.

Note: The function “Redundant ring” requires the following setting for ports 6 and 7: 100 Mbit/s, full duplex and autonegotiation (state on delivery).

Redundant coupling of network segments

The control intelligence built into the RS2-../.. allows the redundant coupling of network segments. The figure on page 70 illustrates the possible configurations.


Ring 1
Ring 2
Ring 3
Ring 4
RS2
RS1
OTP
ring closed by RS2 with
RM switch in position "ON"
Master 2)
Slave 2)
control line 1)
main line
redundant line
Redundant coupling of ring 1 and ring 2
Redundant coupling of ring 1 and ring 3
Redundant coupling of ring 1 and ring 4
2)

RS2: RS2-TX/TX, RS2-FX/FX or RS2-FX-SM/FX-SM

1) : crossover twisted pair cable

2) : For redundant coupling of rings you use port 1 of the RS2/RS1 for the main line and for the redundant line .

Fig. 27: Redundant coupling of rings

Two network segments are connected over two separate paths with one RS2-../.. each.
The redundancy function is assigned to the RS2-../.. in the redundant link via the standby DIP switch setting.
The RS2-../.. in the redundant line and the RS2-../.. in the main line inform each other about their operating states via the control line (crossed twisted pair cable).

Note: The main and redundant lines must be connected to port 1 of the respective RS2-../..s.

Immediately after the main line fails, the redundant RS2-../.. switches to the redundant line. As soon as the main line is restored to normal operation, the RS2-../.. in the main line informs the redundant RS2-../... The main line is activated, and the redundant line is re-blocked.
An error is detected and eliminated within 0.5 seconds.

3 Network management

When people started using heterogeneous computer networks, nobody at the time devoted any thoughts to the fact that they would need to be managed. Today, however, management of networks is gaining increasingly in importance. The size and complexity of networks are increasing along with the number of nodes involved. This makes planning, control and error localization difficult because failure of a network can be tolerated only in the rarest of cases.
A management system enhances clarity and allows users to check the network.

3.1 Management principles

In the mid 70's the International Organization for Standardization ISO began developing a model within the framework of Open Systems Interconnection OSI that defined communication interfaces between devices in a computer network, thus enabling the use of hardware from different manufacturers in one single network.

The ISO/OSI basic reference model was issued as a standard in 1984. See “Historical Development of Ethernet” on page 17.

The protocols ensure communication between devices in different layers.

7	Application Layer		Gateway
6	Presentation Layer		
5	Session Layer		
4	Transport Layer		
3	Network Layer		Router
2	Data Link Layer	2b Logical Link Control	LLC Level Bridge
		2a Medium Access Control	MAC Level Bridge
1	Physical Layer		Star coupler, Repeater

Fig. 28: OSI reference model

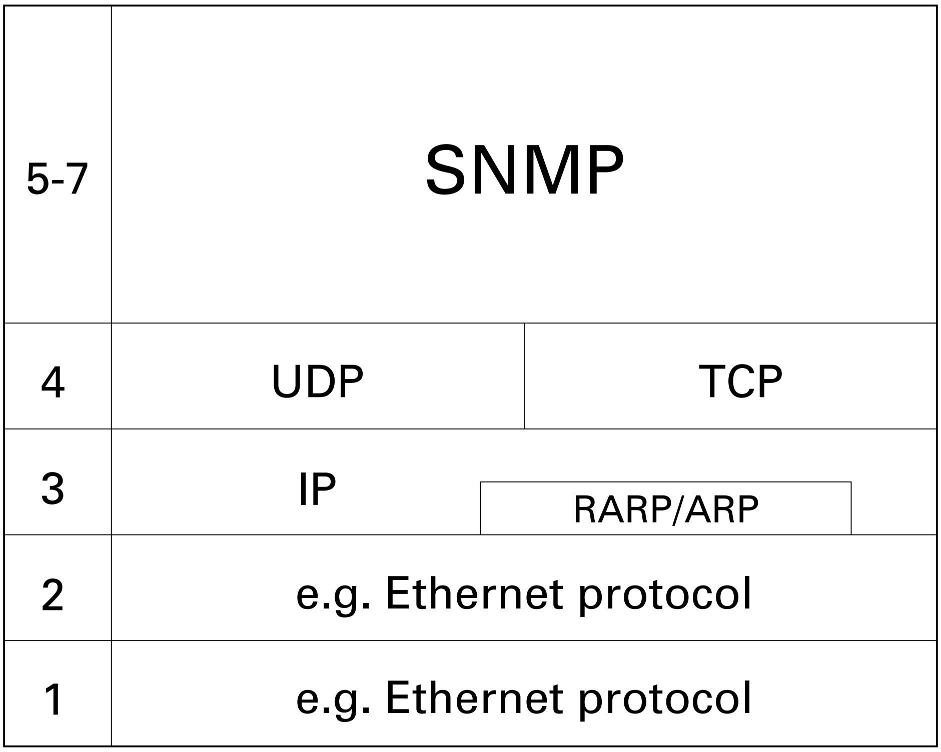


Fig. 29: Affiliation of the SNMP protocol stack to the OSI reference model

The functions of network management

The functions performed in network management can be assigned to 5 groups:

- Configuration management
  – Modifying parameters
  – Starting and ending actions
  – Registering the status of the network components
  – Configuring the network
- Fault management
  – Fault and error messages
  – Fault and error statistics
  – Fault and error diagnosis
  – Thresholds for alarms
  – Tests
- Performance management
  – Real-time statistics
  – RMON statistics

- Security management
  – Password management
  – Privilege management
  – Access management
  – Detecting unauthorized network users
- Accounting management
  – Aids to accounting
  – Verifying invoices
  – Registering cost shares (each user's communication volume)
  – Distributing these costs

Simple Network Management Protocol (SNMP)

A common communication protocol between terminal devices and the network management station (NMS) is needed to manage heterogeneous network environments.
SNMP is one such protocol, which has been adopted and implemented by a large number of manufacturers, therefore representing a de facto standard.

A management system generally consists of the following components:

- An **agent** in a node of the network
  An agent is an item of equipment in the network components (star couplers, concentrators, switches, routers or gateways) that provides information for the manager and influences the components of the network.
- A **manager**, a program running on a management station Working from this station, the person responsible for a network is able to communicate with agents in each of the managed nodes to obtain an overview of their states and to influence the network. The manager itself may be an agent and may be managed, in turn, from a higher instance. The structure is upwardly open.
- A **management protocol** through which the management station exchanges management information with the agents.
- A **Management Information Base** MIB
  The MIB embraces all objects, i.e.
  – agents and managers (managed and managing instances), contained in an open system,
  – including their attributes. The MIB is therefore distributed over the components of the network. The network is checked by reading

and modifying the attributes. Objects may be network components, instances of the components or even software modules.

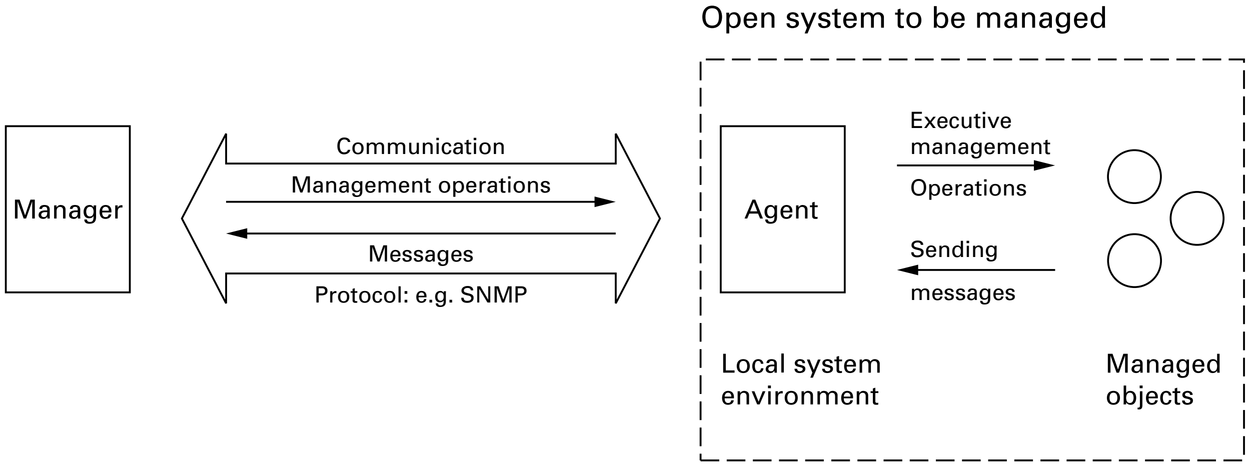


Fig. 30: Communication between manager, agent and objects

3.2 Statistic tables

It is not sufficient for the network manager merely to be given the information that an error has occurred in the network. Conclusions about network reliability can only be drawn when the error frequency is known.

The management card records errors and events in statistics tables based on statistics counters.

Modern agents therefore use the standardized Remote Monitoring (RMON).

RMON is a facility used to manage networks remotely while providing multi-vendor interoperability between monitoring devices and management stations. RMON is defined by an SNMP MIB. This MIB is divided into nine different groups, each gathering specific statistical information or performing a specific function.
RMON-capable devices gather network traffic data and then store them locally until downloaded to an SNMP management station.
Four of the nine groups of RMON defined for Ethernet networks on a per segment basis are:

- RMON 1 – Statistics
  a function that maintains counts of network traffic statistics such as number of packets, broadcasts, collisions, errors, and distribution of packet sizes.
- RMON 2 – History
  a function that collects historical statistics based on user-defined sampling intervals. The statistical information collected is the same as the Statistics group, except on a time stamped basis.
- RMON 3 – Alarm
  a function that allows managers to set alarm thresholds based on traffic statistics. Alarms trigger other actions through the Event group.
- RMON 9 – Event
  a function that operates with the Alarm group to define an action that will be taken when an alarm condition occurs. The event may write a log entry and/or send a trap message.

3.3 Security

3.3.1 SNMP

The Hirschmann agent communicates with the network management station via the Simple Network Management Protocol. Therefore the network management station uses the HiVision network management software or the web based interface.
Every SNMP packet contains the IP address of the sending computer and the community under which the sender of the packet will access the Hirschmann agent MIB.

The Hirschmann agent receives the SNMP packet and compares the IP address of the sending computer and the community with the entries in the access table for communities and the access table for hosts of its MIB. If the community has the appropriate access right, and if the IP address of the sending computer has been entered, then the Hirschmann agent will allow access.

In the delivery state, the Hirschmann agent is accessible via the community ”public” (read only) and ”private” (read and write) from every computer.

To protect your Hirschmann agent from unwanted access:

□ First define a new community which you can access from your computer with all rights.

Note: make a note of the community name and the associated index. For reasons of security, the community name cannot be read later. Access to the community access, trap destination and trap configuration table is made via the community index.

□ Treat this community **with discretion** since everyone who knows the community can access the switch MIB with the IP address of your computer.
□ Limit the access rights of the known communities or delete their entries.

3.3.2 SNMP traps

If unusual events occur during normal operation of the switch, they are reported immediately to the management station. This is done by means of so-called **traps**- alarm messages - that bypass the polling procedure ("Polling" means to query the data stations at regular intervals). Traps make it possible to react quickly to critical situations.

Examples for such events are:

- a hardware reset
- changing the basic device configuration
- link down

Traps can be sent to various hosts to increase the transmission reliability for the messages. A trap message consists of a packet that is not acknowledged.
The management agent sends traps to those hosts that are entered in the trap destination table. The trap destination table can be configured with the management station via SNMP.

4 Switching Functions

A switch contains different functions:

- Frame switching
- Parallel link

4.1 Frame switching

4.1.1 Store and Forward

All data received by an RS2-../.. is stored, and its validity is checked. Invalid and defective data packets (> 1,502 Bytes or CRC errors) as well as fragments (< 64 Bytes) are dropped. Valid data packets are forwarded by an RS2-../...

4.1.2 Multi-address capability

A RS2-../.. learns all the source addresses for a port. Only packets with
– unknown addresses
– these addresses or
– a multi-/broadcast address
in the destination address field are sent to this port.

A RS2-../.. can learn up to several thousand addresses. This becomes necessary if more than one terminal device is connected to one or more ports. It is thus possible to connect several independent subnetworks to a RS2-../.. .

4.1.3 Learning addresses

A RS2-../.. monitors the age of the learned addresses. Address entries which exceed a certain age (aging time), are deleted by the RS2-../.. from its address table. The aging time is set via the management.

4.1.4 Prioritization

The received data packets are assigned to priority queues (traffic classes in compliance with IEEE 802.1D) by the priority of the data packet contained in the VLAN tag.
This function prevents high priority data traffic being disrupted by other traffic during busy periods. The traffic of lower priority will be dropped when the memory or transmission channel is overloaded.

4.1.5 Tagging

The VLAN tag is integrated into the MAC data frame for the VLAN and prioritization functions in accordance with the IEEE 802.1 Q standard. The VLAN tag consists of 4 bytes. It is inserted between the source address field and the type field.

When a data packet is being read, the two bytes are interpreted as type field according to the source address. The content of these two bytes ”81 00” identifies this data packet as a data packet with an embedded tag.

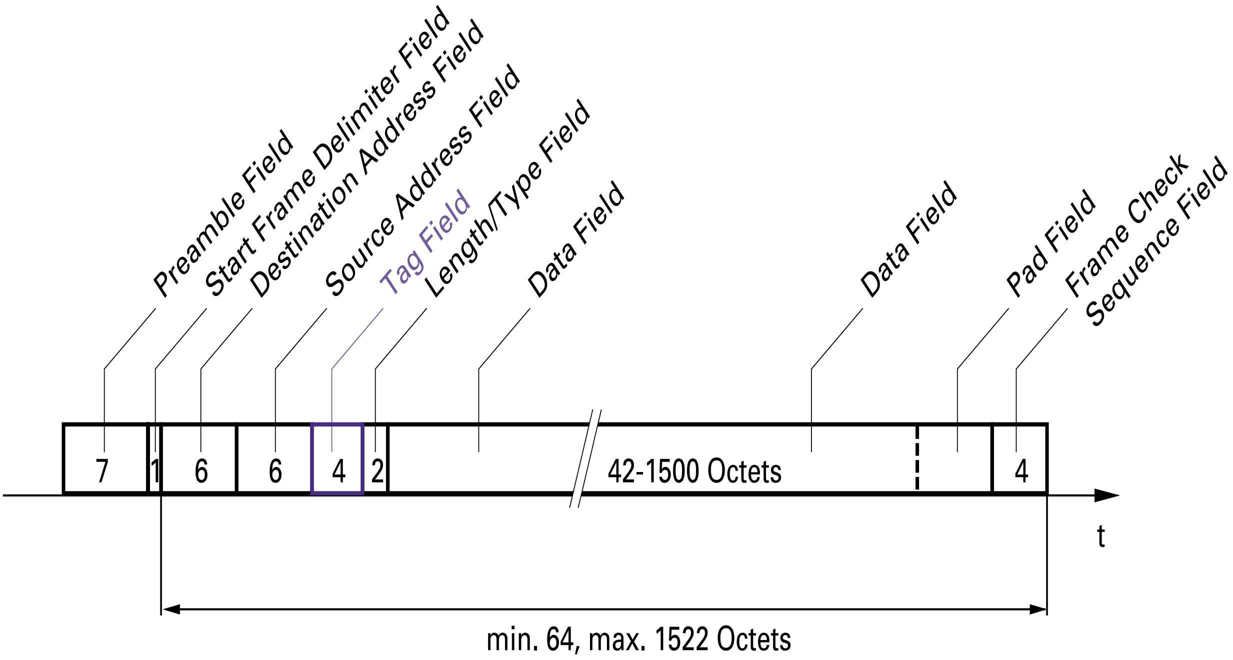


Fig. 31: Ethernet data packet with tag

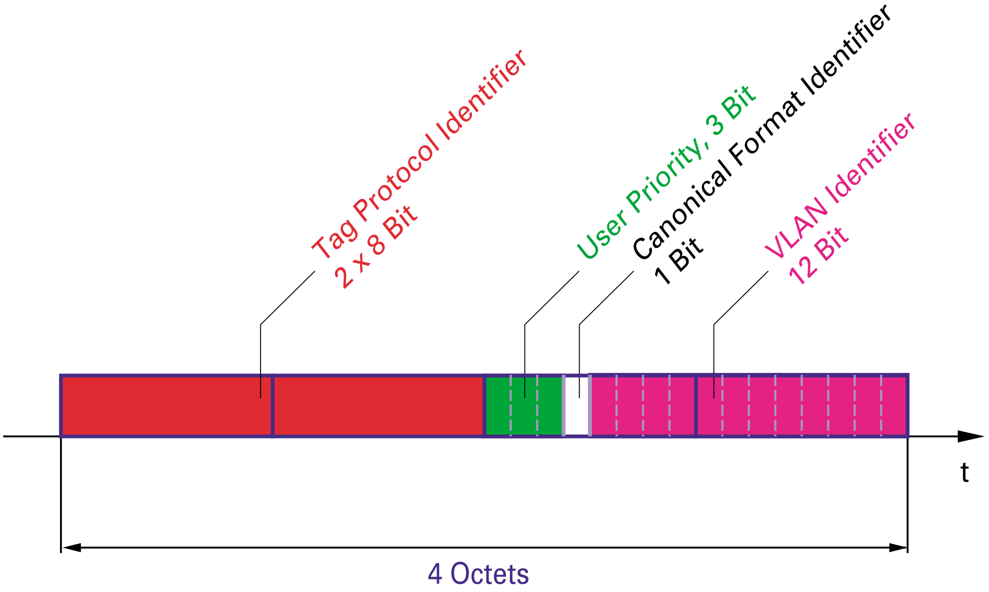


Fig. 32: Tag format

Data packets with VLAN tag, the RS2 evaluates the 3 Bit priority field within the VLAN tag.

The MAC data frame is transferred unchanged by the RS2.

4.2 Parallel Connection

The RS2-../.. is capable of simultaneously receiving data, checking it for errors, and sending it again over several ports.

This makes it possible to move data between several networks in parallel.

If all eight ports are set to full duplex operation (see "Full duplex" on page 29), this results in a theoretical data throughput of 80 Mbit/s for the RS2-../...

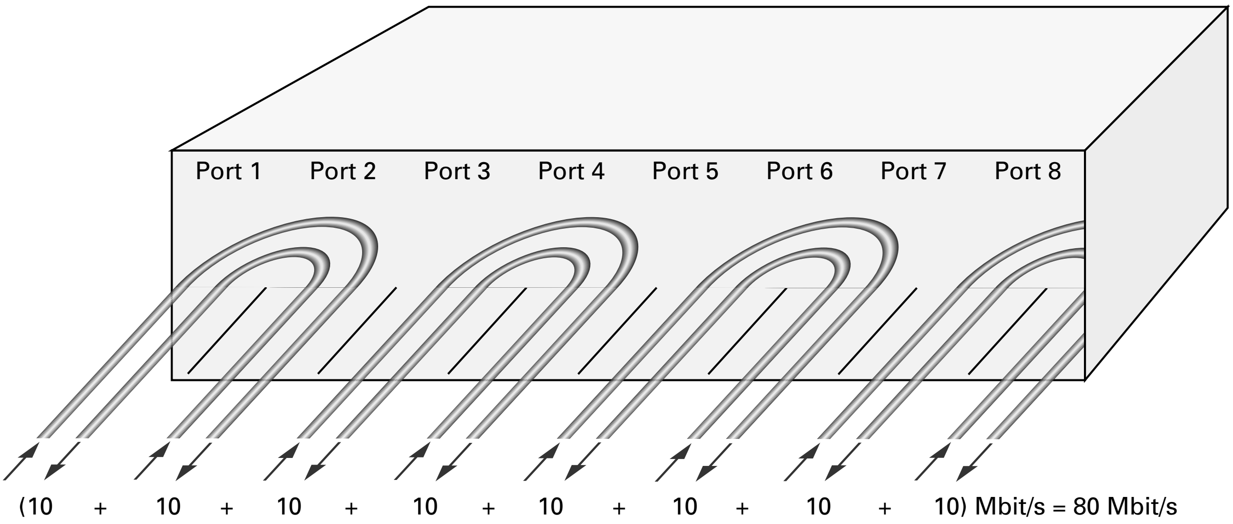
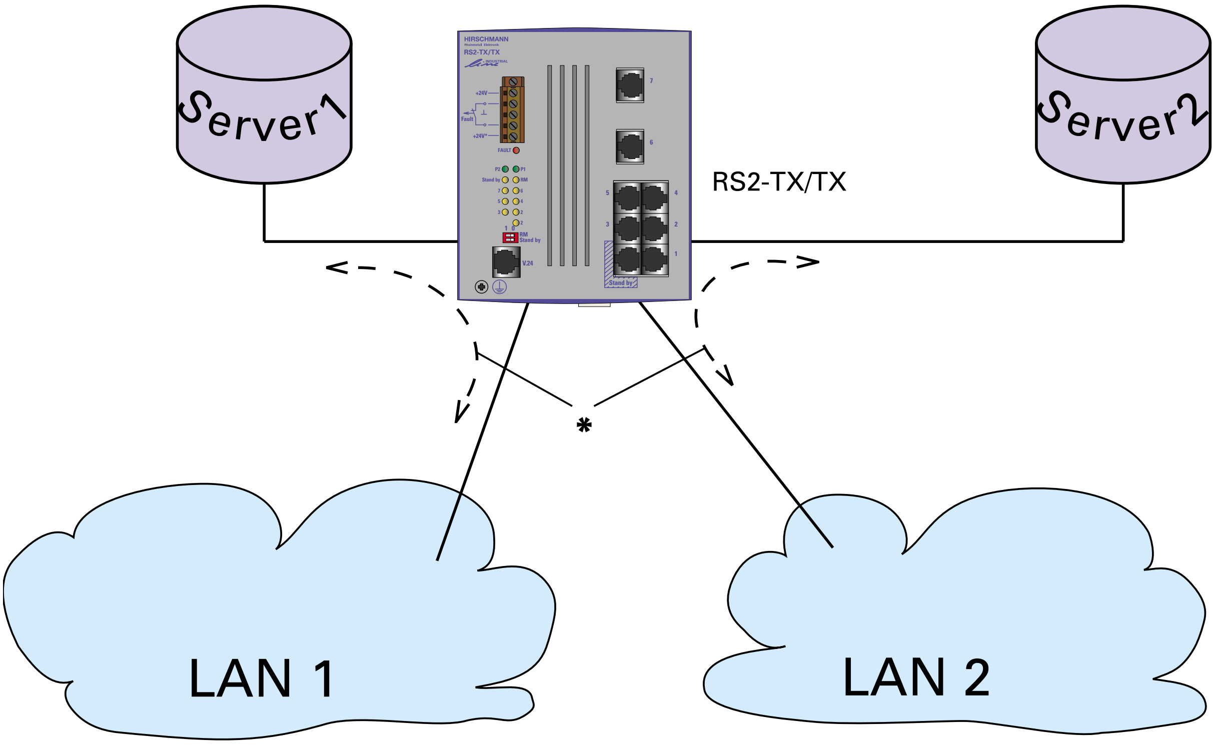


Fig. 33: Example for a data throughput of 80 Mbit/s for an 8-port RS2-../.. in full duplex operation



* **Simultaneous** transmission possible

Fig. 34: Parallel connection of servers and LANs

5 Spanning tree algorithm

Local area networks are becoming ever larger. This is true both for their geographic size as well as for the number of stations they include. As the networks become larger, there are reasons why it often makes sense to implement several bridges:

- reduce network load in subnetworks
- create redundant connections and
- overcome distance limitations

Using many bridges with multiple connections between the subnetworks can lead to considerable problems, possibly even to total network failure if the bridges are configured incorrectly. The spanning tree algorithm described in IEEE 802.1D was developed to prevent this.

Note: The standard demands, that all bridges of a mash have to work with the spanning tree algorithm.

5.1 Tasks

The spanning tree algorithm reduces the topology of any network that is connected using bridges to a single tree structure. The root bridge forms the origin of the tree structure. Any rings that could occur are broken according to pre-defined rules. If there should be a path failure, the algorithm will repeal the loop breakage in order to maintain the data traffic. It is thus possible to increase data reliability by redundant connections.

The following requirements must be met by the algorithm:

- It must automatically reconfigure the tree structure in case of a bridge failure or break in a data path.
- It must stabilize the tree structure for any size network.
- It must stabilize within a short, known time.
- It must produce a reproducible topology that can be pre-defined by management.
- It must be transparent to the terminal equipment.
- By creating a tree structure it must result in a low network load compared to the available transmission capacity.

5.2 Rule for creating the tree structure

Each bridge is uniquely described by the following parameters:

- Bridge identification
- Root path costs
- Port identification

5.2.1 Bridge identification

The bridge identification is 8 bytes long. The 6 low-value bytes are formed by the 48-bit Ethernet address. This ensures that each bridge has a unique identification. The higher-value parts of the bridge identification are formed by the priority number which can be changed by the management administrator when configuring the network. The bridge with the numerically lowest-value bridge identification has the highest priority.

The MAC address and priority are kept in the Management Information Base (see “dot1dBridge (1.3.6.1.2.1.17)” on page 117):

– `dot1dBaseBridgeAddress` (1.3.6.1.2.1.17.1.1.0)
– `dot1dStpPriority` (1.3.6.1.2.1.17.2.2.0)

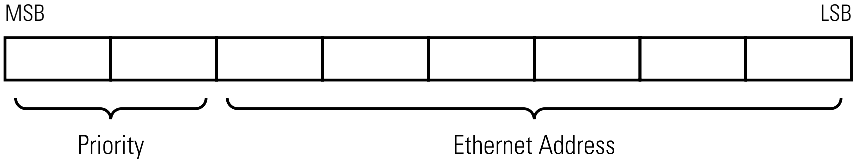


Fig. 35: Bridge identification

5.2.2 Root path costs

Each path connecting two bridges has transmission costs assigned to it. The management administrator sets this value and specifies it for each path when configuring a bridge. The recommended default value is:

Data rate	Recommended value	Recommended range	Possible range
10 MBit/s	100	50-600	1-65.535
100 MBit/s	19	10-60	1-65.535
1 GBit/s	4	3-10	1-65.535
10 GBit/s	2	1-5	1-65.535

Table 10: Recommended path costs dependence on data rate

Because the management administrator essentially has a free hand in specifying this value, he has a tool for ensuring that in case of redundant paths one path will be favored over the others.

The root path costs are calculated by adding up of the individual path costs for the paths that a data packet must traverse between the port of a bridge and the root bridge.

The root path costs and individual path costs are stored in the Management Information Base (see “dot1dBridge (1.3.6.1.2.1.17)” on page 117):

– `dot1dStpRootCost` (1.3.6.1.2.1.17.2.6.0)
– `dot1dStpPortPathCost` (1.3.6.1.2.1.17.2.15.1.5.Index)

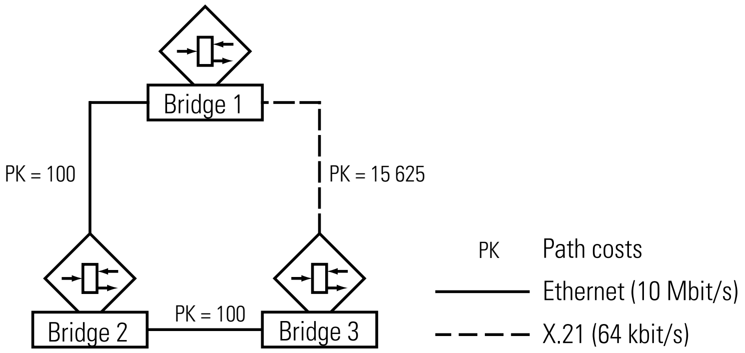


Fig. 36: Path costs

5.2.3 Port identification

The port identification consists of two parts of 8 bits each. One part, the lower-value byte, reflects a fixed relationship to the physical port number. This part ensures that no port in a bridge receives the same designation as another port in the same bridge. The second part contains the priority number which is set by the management administrator. It is also true here that the port with the lowest numerical value for its port identifier is the one with the highest priority.
The port number and port priority number are stored in the Management Information Base (see “dot1dBridge (1.3.6.1.2.1.17)” on page 117):

– `dot1dStpPort` (1.3.6.1.2.1.17.2.15.1.1.Index)
– `dot1dStpPortPriority` (1.3.6.1.2.1.17.2.15.1.2.Index)

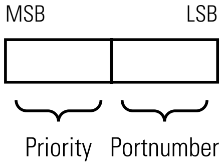


Fig. 37: Port identification

In order to compute their tree structures, the bridges need information about other bridges that are present in the network. This information is obtained by each bridge sending a BPDU (Bridge Protocol Data Unit) to all other bridges.

Along with other information, the BPDU contains the

- bridge identification,
- root path costs, and
- port identification

(see IEEE 802.1D).

- The bridge with the numerically smallest bridge identification is made the root bridge. It forms the root of the tree structure.
- The structure of the tree depends upon the root path costs. The structure that is chosen is the one that provides the lowest path costs between each individual bridge and the root bridge.
- If there are multiple paths with the same root path costs, the priorities of the bridge identifications for the bridges connected to this path determine which bridge is blocked.
- If there are two paths leading away from a single bridge with the same root path costs, the port identification is used as the last criterion for determining which path is used (see Fig. 38). It decides which port is selected.

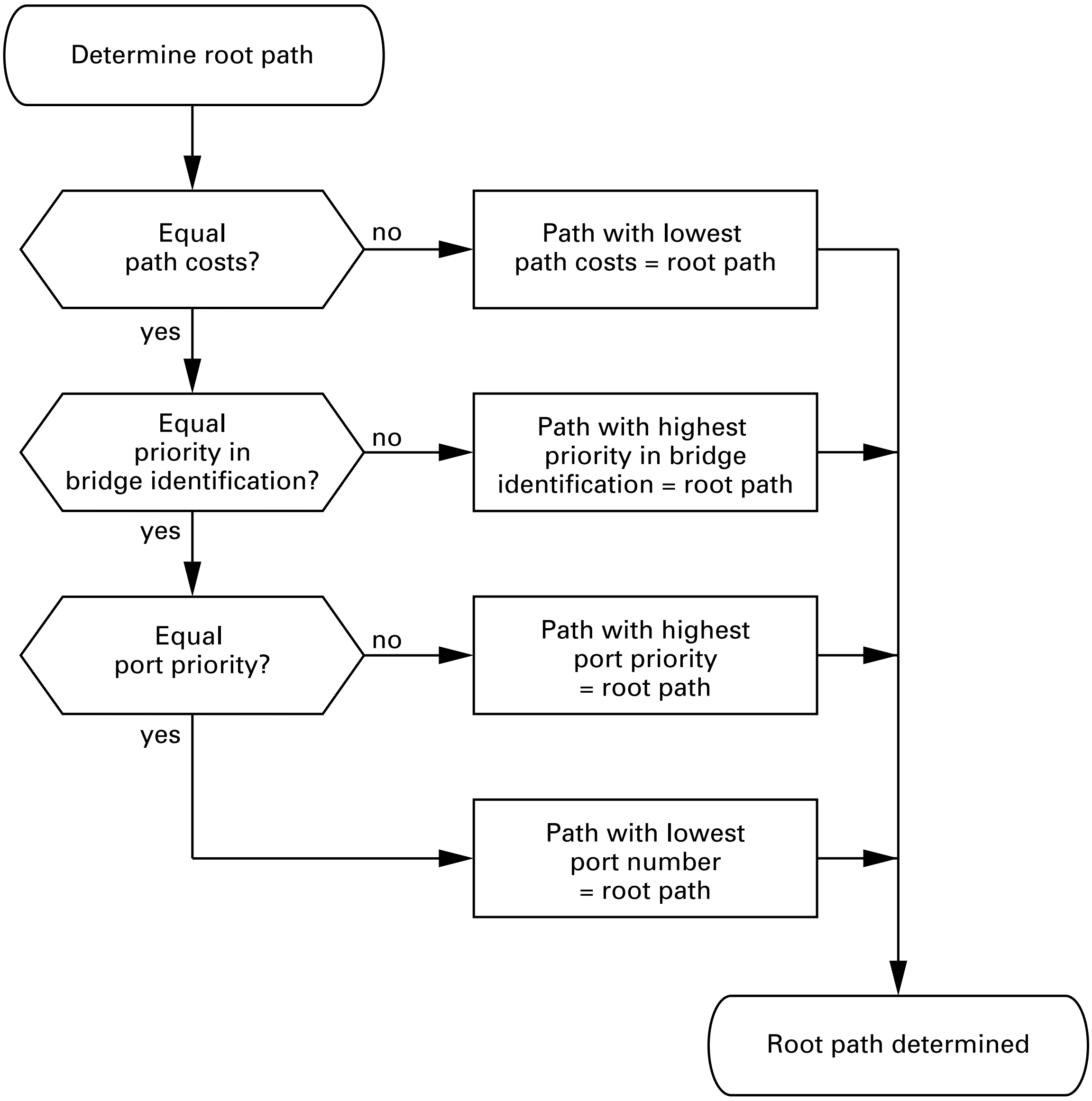

Determine root path
Equal path costs?
no
yes
Path with lowest path costs = root path
Equal priority in bridge identification?
no
yes
Path with highest priority in bridge identification = root path
Equal port priority?
no
yes
Path with highest port priority = root path
Path with lowest port number = root path
Root path determined

Fig. 38: Flow chart for determining root path

Using the network diagram (see Fig. 39), it is possible to follow the logic in the flow chart (see Fig. 38) for determining the root path. The bridge with the numerically smallest bridge identification (in this case, bridge 1) is selected as the root bridge. In this example the partial paths all have the same path costs. The path between bridge 2 and bridge 3 is removed because a connection from bridge 3 to the root bridge via bridge 2 would result in twice the path costs.

The path from bridge 6 to the root bridge is interesting:

- The path via bridges 5 and 3 generates the same root path costs as the path via bridges 4 and 2.
- The path via bridge 4 is selected because the bridge identifier 40 is numerically less than 50.

There are however two paths between bridge 6 and bridge 4. In this case, the larger port priority is decisive.

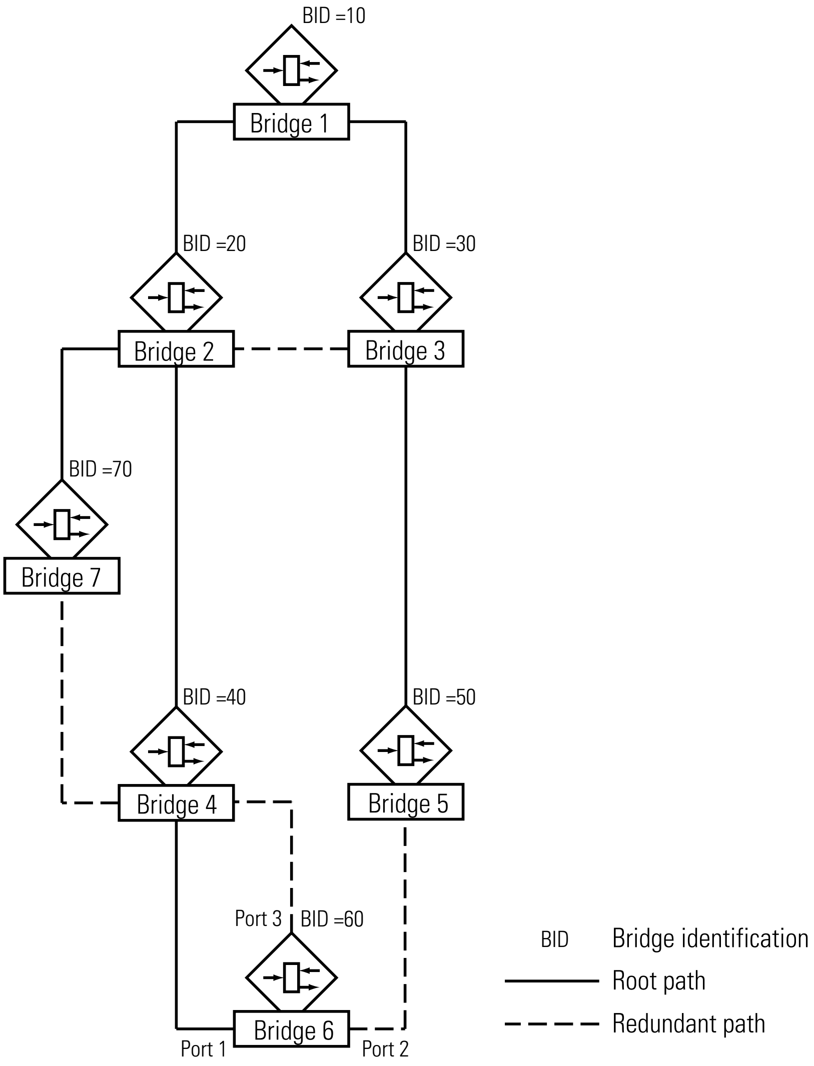


Fig. 39: Root path determination example

5.2.4 Example: manipulation of a tree structure

The management administrator of the network (see Fig. 39) soon discovers that this configuration, with bridge 1 as its root bridge, is unfavorable. The control packets that bridge 1 sends to the other bridges are concentrated on the paths between bridge 1 and bridge 2 and between bridge 1 and bridge 3. If the management administrator raises bridge 2 to the root bridge, the load caused by the control packets will be more evenly distributed among the subnetworks. This would result in the configuration shown (see Fig. 40). The paths between the individual bridges and the root bridge have become shorter.

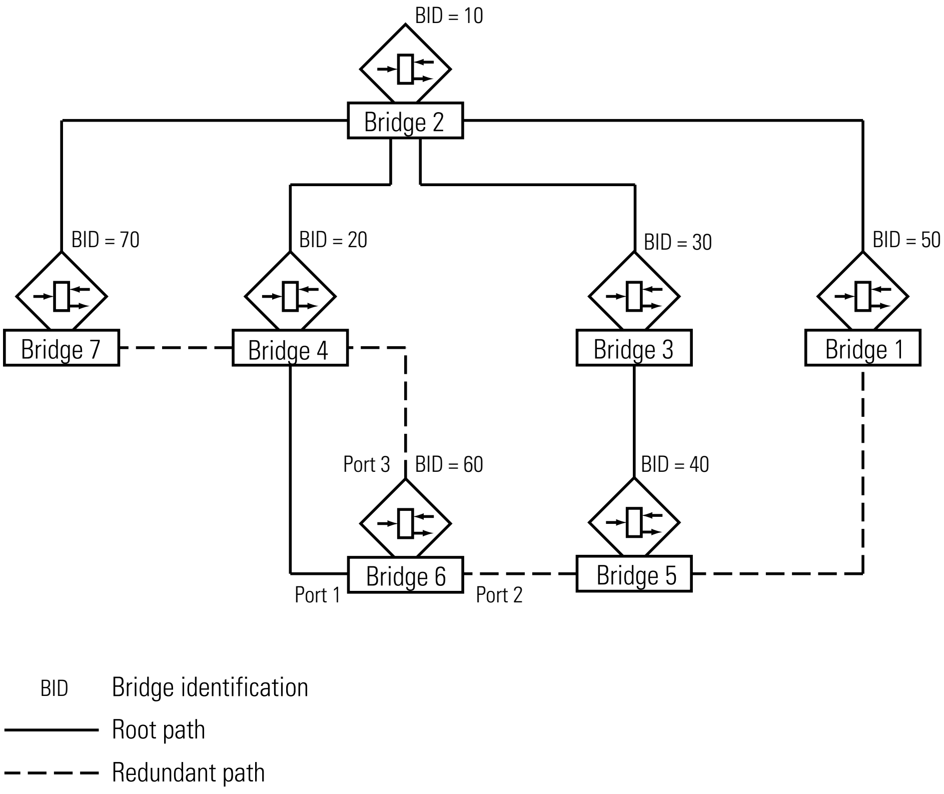


Fig. 40: Example of a root path manipulation

6 Management Information Base MIB

The **M**anagement **I**nformation **B**ase MIB is designed in the form of an abstract tree structure.
The branching points are the **object classes**. The “leaves” of the MIB are called **generic object classes**. Wherever necessary for unambiguous identification, the generic object classes are **instantiated**, i.e. the abstract structure is imaged on the reality, by specifying the port address or the source address.
Values (integers, timeticks, counters or octet strings) are assigned to these instances; these values can be read and, in some cases, modified. The **object description** or **object ID** (OID) identifies the object class. The subidentifier (SID) is used for instantiation.

Example:
The generic object class

```
  dot1dStpPortState (OID = 1.3.6.1.2.1.17.2.15.1.3)
```

is the description of the abstract information „current port state“. It is, however, not possible to read any information from this, as the system does not know which port is meant.
images this abstract information on the reality (instantiates it), which means that it refers to port state of port 4. A value is assigned to this instance and can then be read. The instance „get `1.3.6.1.2.1.17.2.15.1.3.4`“ for example, returns the response „5“, this means that port 4 is forwarding data.

The following abbreviations are used in the MIB:

Comm	Group access rights
con	Configuration
Descr	Description
Fan	Fan
ID	Identifier
Lwr	Lower (e.g. threshold)
PS	Power supply
Pwr	Supply voltage
sys	System

Stp	Spanning Tree Protocol
UI	User Interface
Upr	Upper (e.g. threshold)
ven	Vendor (Hirschmann)

Definition of the syntax terms used:

Integer	An integer in the range 0 - 2^{32}
IP address	xxx.xxx.xxx.xxx (xxx = integer in the range 0-255)
MAC address	12-digit hexadecimal number in accordance with ISO / IEC 8802-3
Object Identifier	x.x.x.x… (e.g. 1.3.6.1.1.4.1.248…)
Octet String	ASCII character string
PSID	Power supply identifier (power supply number)
TimeTicks	Stopwatch Elapsed time (in seconds) = numerical value / 100 Numerical value = integer in the range 0 - 2^{32}
Timeout	Time value in hundredths of a second Time value = integer in the range 0-2^{32}
Typefield	4-digit hexadecimal number in accordance with ISO / IEC 8802-3
Counter	Integer (0 - 2^{32}) whose value is incremented by 1 when certain events occur.

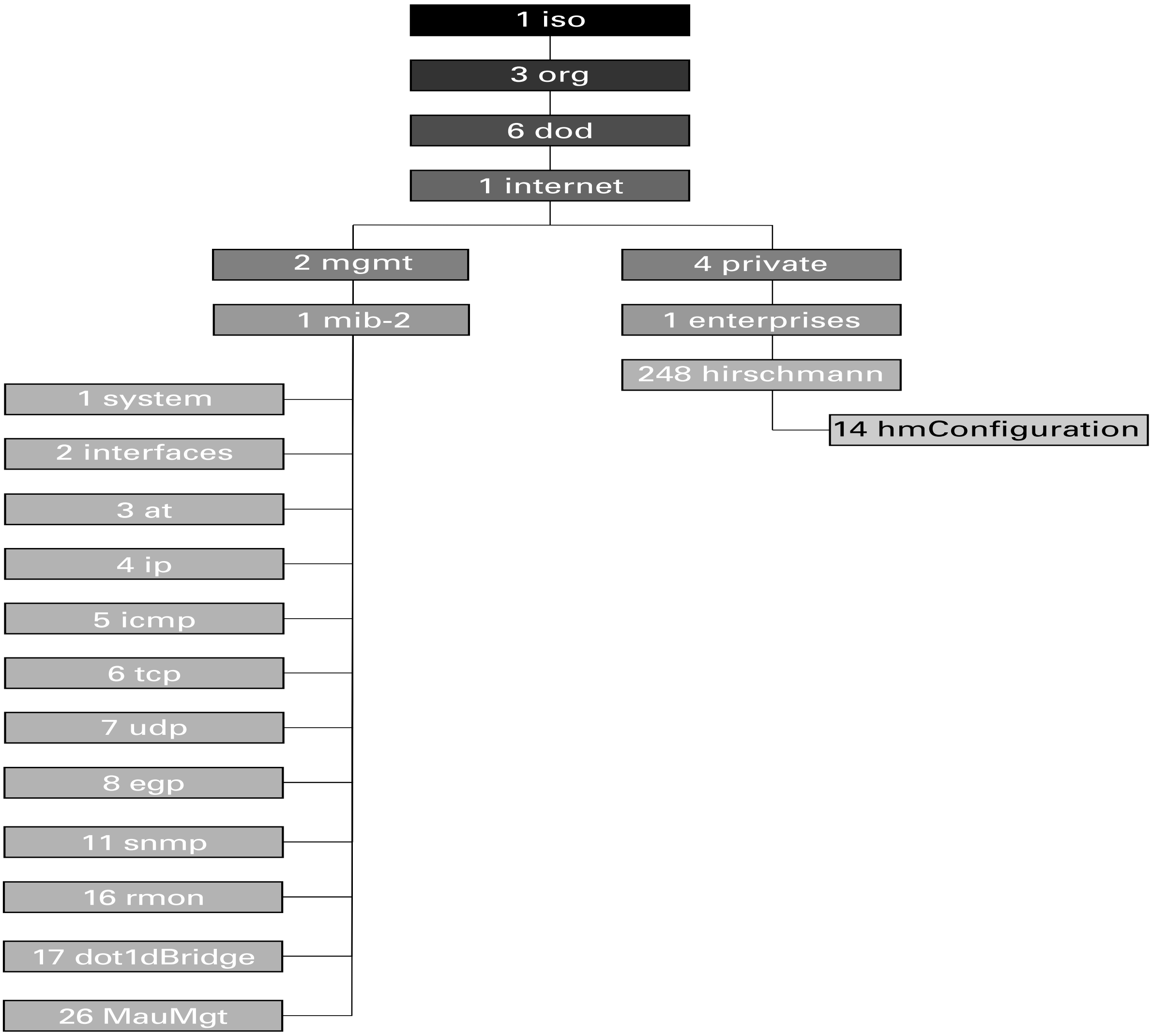


Fig. 41: Baumstruktur der Hirschmann-MIB

6.1 MIB II

6.1.1 System Group (1.3.6.1.2.1.1)

The system group is a required group for all systems. It contains system-related objects. If an agent has no value for a variable, then the response returned includes a string of length 0.

```
(1) system
    |-- (1) sysDescr
    |-- (2) sysObjectID
    |-- (3) sysUpTime
    |-- (4) sysContact
    |-- (5) sysName
    |-- (6) sysLocation
    |-- (7) sysServices
```

sysDescr

OID	1.3.6.1.2.1.1.1.0
Syntax	ASCII String (Größe: 0-255)
Access	Read
Description	A verbal description of the entry. This value should contain the full name and version number of – type of system hardware, – operating system software, and – network software. The description must consist only of printable ASCII characters.

sysObjectID

OID1.3.6.1.2.1.1.2.0

Syntax Object identifier

Access Read

Description The authorization identification of the manufacturer of the network management system that is integrated in this device. This value is placed in the SMI enterprises subtree (1.3.6.1.4.1) and describes which type of device is being managed. For example: if the manufacturer "Hirschmann GmbH" is assigned the subtree 1.3.6.1.4.1.248, then he can assign his bridge the identifier 1.3.6.1.4.1.2.248.2.1.

sysUpTime

OID 1.3.6.1.2.1.1.3.0

Syntax Time ticks

Access Read

Description The time in 1/100 seconds since the last reset of the network management unit.

sysContact

OID1.3.6.1.2.1.1.4.0

Syntax ASCII string (size: 0-255)

Access Read and write

Description The clear-text identification of the contact person for this managed node along with the information about how that person is to be contacted.

sysName

OID1.3.6.1.2.1.1.5.0

Syntax	ASCII string (size: 0-255)
Access	Read and write
Description	A name for this node for identifying it for administra-tion. By convention, this is the fully qualified name in the domain.

sysLocation

OID	1.3.6.1.2.1.1.6.0
Syntax	ASCII string (size: 0-255)
Access	Read and write
Description	The physical location of this node (e.g. “staircase, 3rd floor”)

sysServices

OID	1.3.6.1.2.1.1.7.0
Syntax	Integer (0-127)
Access	Read

Description

This value designates the set of services offered by this device. It is the sum of several terms. For each layer of the OSI reference model there is one term of the form (2^{L-1}), where L identifies the layer.
For example:
For a node that primarily performs routing functions the value would be $(2^{3-1}) = 4$.
For a node that is a host and which offers application services the value would be $(2^{4-1}) + (2^{7-1}) = 72$.

6.1.2 Interface Group (1.3.6.1.2.1.2)

The interface group contains information about the device interfaces.

```
(2) interfaces
   |-- (1) ifNumber
   |-- (2) ifTable
      |-- (1) ifEntry
         |-- (1) ifIndex
         |-- (2) ifDescr
         |-- (3) ifType
         |-- (4) ifMtu
         |-- (5) ifSpeed
         |-- (6) ifPhysAddress
         |-- (7) ifAdminStatus
         |-- (8) ifOperStatus
         |-- (9) ifLastChange
         |-- (10) ifInOctets
         |-- (11) ifInUcastPkts
         |-- (12) ifInNUcastPkts
         |-- (13) ifInDiscards
         |-- (14) ifInErrors
         |-- (15) ifInUnknownProtos
         |-- (16) ifOutOctets
         |-- (17) ifOutUcastPkts
         |-- (18) ifOutNUcastPkts
         |
               |-- (19) ifOutDiscards
               |-- (20) ifOutErrors
               |-- (21) ifOutQLen
               |-- (22) ifSpecific
```

6.1.3 Address Translation Group (1.3.6.1.2.1.3)

The Address Translation Group is required for all systems. It contains information about the assignment of addresses.

```
(3) at
     |-- (1) atTable
          |-- (1) atEntry
               |-- (1) atIfIndex
               |-- (2) atPhysAddress
               |-- (3) atNetAddress
```

6.1.4 Internet Protocol Group (1.3.6.1.2.1.4)

The Internet Protocol Group is required for all systems. It contains information affecting IP switching.

```
(4) ip
     |-- (1) ipForwarding
     |-- (2) ipDefaultTTL
     |-- (3) ipInReceives
     |-- (4) ipInHdrErrors
     |-- (5) ipInAddrErrors
     |-- (6) ipForwDatagrams
     |-- (7) ipInUnknownProtos
     |
|-- (8) ipInDiscards
|-- (9) ipInDelivers
|-- (10) ipOutRequests
|-- (11) ipOutDiscards
|-- (12) ipOutNoRoutes
|-- (13) ipReasmTimeout
|-- (14) ipReasmReqds
|-- (15) ipReasmOKs
|-- (16) ipReasmFails
|-- (17) ipFragOKs
|-- (18) ipFragFails
|-- (19) ipFragCreates
|-- (20) ipAddrTable
|   |-- (1) ipAddrEntry
|       |-- (1) ipAdEntAddr
|       |-- (2) ipAdEntIfIndex
|       |-- (3) ipAdEntNetMask
|       |-- (4) ipAdEntBcastAddr
|       |-- (5) ipAdEntReasmMaxSize
|-- (21) ipRouteTable
|   |-- (1) ipRouteEntry
|       |-- (1) ipRouteDest
|       |-- (2) ipRouteIfIndex
|       |-- (3) ipRouteMetric1
|       |-- (4) ipRouteMetric2
|       |-- (5) ipRouteMetric3
|       |-- (6) ipRouteMetric4
|       |-- (7) ipRouteNextHop
|       |-- (8) ipRouteType
|       |-- (9) ipRouteProto
|       |-- (10) ipRouteAge
|       |-- (11) ipRouteMask
|       |-- (12) ipRouteMetric5
|       |-- (13) ipRouteInfo
|-- (22) ipNetToMediaTable
|   |-- (1) ipNetToMediaEntry
|       |-- (1) ipNetToMediaIfIndex
|       |-- (2) ipNetToMediaPhysAddress
|       |-- (3) ipNetToMediaNetAddress
|       |-- (4) ipNetToMediaType
|-- (23) ipRoutingDiscards
```

6.1.5 ICMP Group (1.3.6.1.2.1.5)

The Internet Control Message Protocol group is obligatory for all systems. It contains all the information on error handling and control for data exchange in the Internet.

```
(5) icmp
    |-- (1) icmpInMsgs
    |-- (2) icmpInErrors
    |-- (3) icmpInDestUnreachs
    |-- (4) icmpInTimeExcds
    |-- (5) icmpInParmProbs
    |-- (6) icmpInSrcQuenchs
    |-- (7) icmpInRedirects
    |-- (8) icmpInEchos
    |-- (9) icmpInEchoReps
    |-- (10) icmpInTimestamps
    |-- (11) icmpInTimestampReps
    |-- (12) icmpInAddrMasks
    |-- (13) icmpInAddrMaskReps
    |-- (14) icmpOutMsgs
    |-- (15) icmpOutErrors
    |-- (16) icmpOutDestUnreachs
    |-- (17) icmpOutTimeExcds
    |-- (18) icmpOutParmProbs
    |-- (19) icmpOutSrcQuenchs
    |-- (20) icmpOutRedirects
    |-- (21) icmpOutEchos
    |-- (22) icmpOutEchoReps
    |-- (23) icmpOutTimestamps
    |-- (24) icmpOutTimestampReps
    |-- (25) icmpOutAddrMasks
    |-- (26) icmpOutAddrMaskReps
```

6.1.6 Transfer Control Protocol Group (1.3.6.1.2.1.6)

The Transfer Control Protocol group is required for all systems that have implemented TCP. Instances of objects that describe information about a particular TCP connection exist only as long as the connection exists.

```
(6) tcp
  |-- (1) tcpRtoAlgorithm
  |-- (2) tcpRtoMin
  |-- (3) tcpRtoMax
  |-- (4) tcpMaxConn
  |-- (5) tcpActiveOpens
  |-- (6) tcpPassiveOpens
  |-- (7) tcpAttemptFails
  |-- (8) tcpEstabResets
  |-- (9) tcpCurrEstab
  |-- (10) tcpInSegs
  |-- (11) tcpOutSegs
  |-- (12) tcpRetransSegs
  |-- (13) tcpConnTable
  |    |-- (1) tcpConnEntry
  |         |-- (1) tcpConnState
  |         |-- (2) tcpConnLocalAddress
  |         |-- (3) tcpConnLocalPort
  |         |-- (4) tcpConnRemAddress
  |         |-- (5) tcpConnRemPort
  |-- (14) tcpInErrs
  |-- (15) tcpOutRsts
```

6.1.7 User Datagram Protocol Group (1.3.6.1.2.1.7)

The User Datagram Protocol group is required for all systems that have implemented UDP.

```
(7) udp
    |-- (1) udpInDatagrams
    |-- (2) udpNoPorts
    |-- (3) udpInErrors
    |-- (4) udpOutDatagrams
    |-- (5) udpTable
        |-- (1) udpEntry
            |-- (1) udpLocalAddress
            |-- (2) udpLocalPort
```

6.1.8 Exterior Gateway Protocol Group (1.3.6.1.2.1.8)

The EGP (Exterior Gateway Protocol) routing method will be used to exchange reachability information between the NSFNET backbone and the regional networks.

```
(8) egp
    |-- (1) egpInMsgs
    |-- (2) egpInErrors
    |-- (3) egpOutMsgs
    |-- (4) egpOutErrors
    |-- (5) egpNeighTable
        |-- (1) egpNeighEntry
            |-- (1) egpNeighState
            |-- (2) egpNeighAddr
            |-- (3) egpNeighAs
            |-- (4) egpNeighInMsgs
            |-- (5) egpNeighInErrs
            |-- (6) egpNeighOutMsgs
            |-- (7) egpNeighOutErrs
```

```
           |-- (8) egpNeighInErrMsgs
           |-- (9) egpNeighOutErrMsgs
           |-- (10) egpNeighStateUps
           |-- (11) egpNeighStateDowns
           |-- (12) egpNeighIntervalHello
           |-- (13) egpNeighIntervalPoll
           |-- (14) egpNeighMode
           |-- (15) egpNeighEventTrigger
       |-- (6) egpAs
```

6.1.9 Simple Network Management Protocol Group (1.3.6.1.2.1.11)

The Simple Network Management Protocol group is required for all systems. In SNMP installations that have been optimized to support either just one agent or one management station, some of the listed objects will contain the value “0”.

```
(11) snmp
    |-- (1) snmpInPkts
    |-- (2) snmpOutPkts
    |-- (3) snmpInBadVersions
    |-- (4) snmpInBadCommunityNames
    |-- (5) snmpInBadCommunityUses
    |-- (6) snmpInASNParseErrs
    |-- (7) not used
    |-- (8) snmpInTooBigs
    |-- (9) snmpInNoSuchNames
    |-- (10) snmpInBadValues
    |-- (11) snmpInReadOnlys
    |-- (12) snmpInGenErrs
    |-- (13) snmpInTotalReqVars
    |-- (14) snmpInTotalSetVars
    |-- (15) snmpInGetRequests
    |-- (16) snmpInGetNexts
    |-- (17) snmpInSetRequests
|-- (18) snmpInGetResponses
|-- (19) snmpInTraps
|-- (20) snmpOutTooBigs
|-- (21) snmpOutNoSuchNames
|-- (22) snmpOutBadValues
|-- (23) not used
|-- (24) snmpOutGenErrs
|-- (25) snmpOutGetRequests
|-- (26) snmpOutGetNexts
|-- (27) snmpOutSetRequests
|-- (28) snmpOutGetResponses
|-- (29) snmpOutTraps
|-- (30) snmpEnableAuthenTraps
```

6.1.10 RMON-Gruppe (1.3.6.1.2.1.16)

This part of the MIB provides a continuous flow of current and historical network component data to the network management. The configuration of alarms and events controls the evaluation of network component counters. The agents inform the management station of the evaluation result by means traps depending on the configuration.

```
(16) rmon
   |--(1) statistics
      |--(1) etherStatsTable
         |--(1) etherStatsEntry
            |--(1) etherStatsIndex
            |--(2) etherStatsDataSource
            |--(3) etherStatsDropEvents
            |--(4) etherStatsOctets
            |--(5) etherStatsPkts
            |--(6) etherStatsBroadcastPkts
            |--(7) etherStatsMulticastPkts
            |--(8) etherStatsCRCAlignErrors
            |--(9) etherStatsUndersizePkts
            |--(10) etherStatsOversizePkts
            |
            |--(11) etherStatsFragments
            |--(12) etherStatsJabbers
            |--(13) etherStatsCollisions
            |--(14) etherStatsPkts64Octets
            |--(15) etherStatsPkts65to127Octets
            |--(16) etherStatsPkts128to255Octets
            |--(17) etherStatsPkts256to511Octets
            |--(18) etherStatsPkts512to1023Octets
            |--(19) etherStatsPkts1024to1518Octets
            |--(20) etherStatsOwner
            |--(21) etherStatsStatus
|--(2)  history (2)
    |--(1) historyControlTable
        |--(1) historyControlEntry
            |--(1) historyControlIndex
            |--(2) historyControlDataSource
            |--(3) historyControlBucketsRequested
            |--(4) historyControlBucketsGranted
            |--(5) historyControlInterval
            |--(6) historyControlOwner
            |--(7) historyControlStatus
    |--(2) etherHistoryTable
        |--(1) etherHistoryEntry
            |--(1) etherHistoryIndex
            |--(2) etherHistorySampleIndex
            |--(3) etherHistoryIntervalStart
            |--(4) etherHistoryDropEvents
            |--(5) etherHistoryOctets
            |--(6) etherHistoryPkts
            |--(7) etherHistoryBroadcastPkts
            |--(8) etherHistoryMulticastPkts
            |--(9) etherHistoryCRCAlignErrors
            |--(10) etherHistoryUndersizePkts
            |--(11) etherHistoryOversizePkts
            |--(12) etherHistoryFragments
            |--(13) etherHistoryJabbers
            |--(14) etherHistoryCollisions
            |--(15) etherHistoryUtilization
|--(2)  alarm
    |--(1) alarmTable
        |--(1) alarmEntry
            |--(1) alarmIndex
            |--(2) alarmInterval
            |
                |--(3) alarmVariable
                |--(4) alarmSampleType
                |--(5) alarmValue
                |--(6) alarmStartupAlarm
                |--(7) alarmRisingThreshold
                |--(8) alarmFallingThreshold
                |--(9) alarmRisingEventIndex
                |--(10) alarmFallingEventIndex
                |--(11) alarmOwner
                |--(12) alarmStatus
    |--(9)  event
        |--(1) eventTable
            |--(1) eventEntry
                |--(1) eventIndex
                |--(2) eventDescription
                |--(3) eventType
                |--(4) eventCommunity
                |--(5) eventLastTimeSent
                |--(6) eventOwner
                |--(7) eventStatus
        |--(2) logTable
            |--(1) logEntry (1)
                |--(1) logEventIndex
                |--(2) logIndex
                |--(3) logTime
                |--(4) logDescription
```

6.1.11 dot1dBridge (1.3.6.1.2.1.17)

This part of the MIB contains bridge-specific objects.

```
(17) dot1dBridge
    |--(1) dot1dBase
        |--(1) dot1dBaseBridgeAddress
        |--(2) dot1dBaseNumPorts
        |--(3) dot1dBaseType
        |
          |--(4) dot1dBasePortTable
              |--(1) dot1dBasePortEntry
                  |--(1) dot1dBasePort
                  |--(2) dot1dBasePortIfIndex
                  |--(3) dot1dBasePortCircuit
                  |--(4) dot1dBasePortDelayExceededDiscards
                  |--(5) dot1dBasePortMtuExceededDiscards
      |--(2) dot1dStp
          |--(1) dot1dStpProtocolSpecification
          |--(2) dot1dStpPriority
          |--(3) dot1dStpTimeSinceTopologyChange
          |--(4) dot1dStpTopChanges
          |--(5) dot1dStpDesignatedRoot
          |--(6) dot1dStpRootCost
          |--(7) dot1dStpRootPort
          |--(8) dot1dStpMaxAge
          |--(9) dot1dStpHelloTime
          |--(10) dot1dStpHoldTime
          |--(11) dot1dStpForwardDelay
          |--(12) dot1dStpBridgeMaxAge
          |--(13) dot1dStpBridgeHelloTime
          |--(14) dot1dStpBridgeForwardDelay
          |--(15) dot1dStpPortTable
              |--(1) dot1dStpPortEntry
                  |--(1) dot1dStpPort
                  |--(2) dot1dStpPortPriority
                  |--(3) dot1dStpPortState
                  |--(4) dot1dStpPortEnable
                  |--(5) dot1dStpPortPathCost
                  |--(6) dot1dStpPortDesignatedRoot
                  |--(7) dot1dStpPortDesignatedCost
                  |--(8) dot1dStpPortDesignatedBridge
                  |--(9) dot1dStpPortDesignatedPort
                  |--(10) dot1dStpPortForwardTransitions
      |--(3) dot1dSr
      |--(4) dot1dTp
          |--(1) dot1dTpLearnedEntryDiscards
          |--(2) dot1dTpAgingTime
          |--(3) dot1dTpFdbTable
              |--(1) dot1dTpFdbEntry
                  |--(1) dot1dTpFdbAddress
                  |--(2) dot1dTpFdbPort
                  |--(3) dot1dTpFdbStatus
```

```
    |--(4) dot1dTpPortTable
        |--(1) dot1dTpPortEntry
            |--(1) dot1dTpPort
            |--(2) dot1dTpPortMaxInfo
            |--(3) dot1dTpPortInFrames
            |--(4) dot1dTpPortOutFrames
            |--(5) dot1dTpPortInDiscards
|--(5) dot1dStatic
    |--(1) dot1dStaticTable
        |--(1) dot1dStaticEntry
            |--(1) dot1dStaticAddress
            |--(2) dot1dStaticReceivePort
            |--(3) dot1dStaticAllowedToGoTo
            |--(4)  dot1dStaticStatus
|--(6) pBridgeMIB
    |--(1) pBridgeMIBObjects
        |--(1) dot1dExtBase
            |--(1) dot1dDeviceCapabilities
            |--(2) dot1dTrafficClassesEnabled
            |--(3) dot1dGmrpStatus
            |--(4) dot1dPortCapabilitiesTable
                |--(1) dot1dPortCapabilitiesEntry
                    |--(1) dot1dPortCapabilities
        |--(2) dot1dPriority
            |--(1) dot1dPortPriorityTable
                |--(1) dot1dPortPriorityEntry
                    |--(1) dot1dPortDefaultUserPriority
                    |--(2) dot1dPortNumTrafficClasses
            |--(2) dot1dUserPriorityRegenTable
                |--(1) dot1dUserPriorityRegenEntry
                    |--(1) dot1dUserPriority
                    |--(2) dot1dRegenUserPriority
            |--(3) dot1dTrafficClassTable
                |--(1) dot1dTrafficClassEntry
                    |--(1) dot1dTrafficClassPriority
                    |--(2) dot1dTrafficClass
            |--(4) dot1dPortOutboundAccessPriorityTable
                |--(1) dot1dPortOutboundAccessPriorityEntry
                    |--(1) dot1dPortOutboundAccessPriority
        |--(3) dot1dGarp
            |--(1) dot1dPortGarpTable
                |--(1) dot1dPortGarpEntry
                    |--(1) dot1dPortGarpJoinTime
                    |
              |--(2) dot1dPortGarpLeaveTime
              |--(3) dot1dPortGarpLeaveAllTime
        |--(4) dot1dGmrp
          |--(1) dot1dPortGmrpTable
            |--(1) dot1dPortGmrpEntry
              |--(1) dot1dPortGmrpStatus
              |--(2) dot1dPortGmrpFailedRegistrations
              |--(3) dot1dPortGmrpLastPduOrigin
    |--(7) qBridgeMIB
      |--(1) qBridgeMIBObjects
        |--(1) dot1qBase
          |--(1) dot1qVlanVersionNumber
          |--(2) dot1qMaxVlanId
          |--(3) dot1qMaxSupportedVlans
          |--(4) dot1qNumVlans
          |--(5) dot1qGvrpStatus
        |--(2) dot1qTp
          |--(1) dot1qFdbTable
            |--(1) dot1qFdbEntry
              |--(1) dot1qFdbId
              |--(2) dot1qFdbDynamicCount
          |--(2) dot1qTpFdbTable
            |--(1) dot1qTpFdbEntry
              |--(1) dot1qTpFdbAddress
              |--(2) dot1qTpFdbPort
              |--(3) dot1qTpFdbStatus
          |--(3) dot1qTpGroupTable
            |--(1) dot1qTpGroupEntry
              |--(1) dot1qTpGroupAddress
              |--(2) dot1qTpGroupEgressPorts
              |--(3) dot1qTpGroupLearnt
          |--(4) dot1qForwardAllTable
            |--(1) dot1qForwardAllEntry
              |--(1) dot1qForwardAllPorts
              |--(2) dot1qForwardAllStaticPorts
              |--(3) dot1qForwardAllForbiddenPorts
          |--(5) dot1qForwardUnregisteredTable
            |--(1) dot1qForwardUnregisteredEntry
              |--(1) dot1qForwardUnregisteredPorts
              |--(2) dot1qForwardUnregisteredStaticPorts
              |--(3)
dot1qForwardUnregisteredForbiddenPorts
        |--(3) dot1qStatic
```

```
            |--(1) dot1qStaticUnicastTable
                |--(1) dot1qStaticUnicastEntry
                    |--(1) dot1qStaticUnicastAddress
                    |--(2) dot1qStaticUnicastReceivePort
                    |--(3) dot1qStaticUnicastAllowedToGoTo
                    |--(4) dot1qStaticUnicastStatus
            |--(2) dot1qStaticMulticastTable
                |--(1) dot1qStaticMulticastEntry
                    |--(1) dot1qStaticMulticastAddress
                    |--(2) dot1qStaticMulticastReceivePort
                    |--(3) dot1qStaticMulticastStaticEgressPorts
                    |--(4)
dot1qStaticMulticastForbiddenEgressPorts
                    |--(5) dot1qStaticMulticastStatus
        |--(4) dot1qVlan
            |--(1) dot1qVlanNumDeletes
            |--(2) dot1qVlanCurrentTable
                |--(1) dot1qVlanCurrentEntry
                    |--(1) dot1qVlanTimeMark
                    |--(2) dot1qVlanIndex
                    |--(3) dot1qVlanFdbId
                    |--(4) dot1qVlanCurrentEgressPorts
                    |--(5) dot1qVlanCurrentUntaggedPorts
                    |--(6) dot1qVlanStatus
                    |--(7) dot1qVlanCreationTime
            |--(3) dot1qVlanStaticTable
                |--(1) dot1qVlanStaticEntry
                    |--(1) dot1qVlanStaticName
                    |--(2) dot1qVlanStaticEgressPorts
                    |--(3) dot1qVlanForbiddenEgressPorts
                    |--(4) dot1qVlanStaticUntaggedPorts
                    |--(5) dot1qVlanStaticRowStatus
            |--(4) dot1qNextFreeLocalVlanIndex
            |--(5) dot1qPortVlanTable
                |--(1) dot1qPortVlanEntry
                    |--(1) dot1qPvid
                    |--(2) dot1qPortAcceptableFrameTypes
                    |--(3) dot1qPortIngressFiltering
                    |--(4) dot1qPortGvrpStatus
                    |--(5) dot1qPortGvrpFailedRegistrations
                    |--(6) dot1qPortGvrpLastPduOrigin
            |--(6) dot1qPortVlanStatisticsTable
                |--(1) dot1qPortVlanStatisticsEntry
                    |--(1) dot1qTpVlanPortInFrames
```

```
                 |--(2) dot1qTpVlanPortOutFrames
                 |--(3) dot1qTpVlanPortInDiscards
                 |--(4) dot1qTpVlanPortInOverflowFrames
                 |--(5) dot1qTpVlanPortOutOverflowFrames
                 |--(6) dot1qTpVlanPortInOverflowDiscards
         |--(8) dot1qLearningConstraintsTable
             |--(1) dot1qLearningConstraintsEntry
                 |--(1) dot1qConstraintVlan
                 |--(2) dot1qConstraintSet
                 |--(3) dot1qConstraintType
                 |--(4) dot1qConstraintStatus
         |--(9) dot1qConstraintSetDefault
         |--(10) dot1qConstraintTypeDefault
```

6.1.12 MAU Management Group (1.3.6.1.2.1.26)

The MAU Management Group is responsible for setting the autonegotiation parameters.

```
(26) snmpDot3MauMgt
    |--(2) dot3IfMauBasicGroup
        |--(1) ifMauTable
            |--(1) ifMauEntry
                |-- (1) ifMauIfIndex
                |-- (2) ifMauIndex
                |-- (3) ifMauType
                |-- (4) ifMauStatus
                |-- (5) ifMauMediaAvailable
                |-- (6) ifMauMediaAvailableStateExits
                |-- (7) ifMauJabberState
                |-- (8) ifMauJabberingStateEnters
                |-- (9) ifMauFalseCarriers
                |-- (10)ifMauTypeList
                |-- (11)ifMauDefaultType
                |-- (12)ifMauAutoNegSupported
    |--(5) dot3IfMauAutoNegGroup
```

```
|--(1) ifMauAutoNegTable
    |-- (1) ifMauAutoNegEntry
    |    |-- (1) ifMauAutoNegAdminStatus
         |-- (2) ifMauAutoNegRemoteSignaling
         |-- (4) ifMauAutoNegConfig
         |-- (5) ifMauAutoNegCapability
         |-- (6) ifMauAutoNegCapAdvertised
         |-- (7) ifMauAutoNegCapReceived
         |-- (8) ifMauAutoNegRestart
```

6.2 Private MIB

The private MIB is for configuring the device-specific properties of the RS2-../...
The groups below are implemented in the RS2-../.. from the private MIB hm-Configuration (OID = 1.3.6.1.4.1.248.14).

- hmChassis (OID = 1.3.6.1.4.1.248.14.1)
- hmAgent (OID = 1.3.6.1.4.1.248.14.2)

6.2.1 Device Group

The device group contains information on the status of the RS2-../.. hardware.

```
(14) hmConfiguration
    |--(1) hmChassis
        |--(1) hmSystemTable
            |--(1) hmSysProduct
            |--(2) hmSysVersion
            |--(3) hmSysGroupCapacity
            |--(4) hmSysGroupMap
            |--(5) hmSysMaxPowerSupply
            |--(6) hmSysMaxFan
            |--(7) hmSysGroupModuleCapacity
            |--(8) hmSysModulePortCapacity
            |--(9) hmSysGroupTable
                |--(1) hmSysGroupEntry
                    |--(1) hmSysGroupID
                    |--(2) hmSysGroupType
                    |--(2) hmSysGroupDescription
                    |--(4) hmSysGroupHwVersion
                    |--(5)  hmSysGroupSwVersion
                    |--(6) hmSysGroupModuleMap
        |--(10) hmSysModuleTable
            |--(1) hmSysModuleEntry
                |--(1) hmSysModGroupID
                |--(2) hmSysModID
                |--(3) hmSysModType
                |--(4) hmSysModDescription
                |--(5) hmSysModVersion
                |--(6) hmSysModNumOfPorts
                |--(7) hmSysModFirstMauIndex
        |--(11) hmInterfaceTable
            |--(1) hmIfEntry
                |--(1) hmIfaceGroupID
                |--(2) hmIfaceID
                |--(3) hmIfaceStpEnable
                |--(4) hmIfaceLinkType
                |--(5) hmIfaceAction
                |--(6) hmIfaceNextHopMacAddress
                |--(7) hmIfaceFlowControl
        |--(20) hmSysChassisName
        |--(21) hmSysStpEnable
    |--(2) hmPSTable
        |--(1) hmPSEntry
            |--(1) hmPSSysID
            |--(2) hmPSID
            |--(3) hmPSState
    |--(3) hmFanTable
        |--(1) hmFanEntry
            |--(1) hmFanSysID
            |--(2) hmFanID
            |--(3) hmFanState
```

6.2.2 Management Group

The management group contains parameters for configuring the management agent.

```
(14) hmConfiguration
   |--(2) hmAgent
      |--(1) hmAction
      |--(2) hmActionResult
      |--(3) hmNetwork
         |--(1) hmNetLocalIPAddr
         |--(2) hmNetLocalPhysAddr
         |--(3) hmNetGatewayIPAddr
         |--(4) hmNetMask
      |--(4) hmFSTable
         |--(1) hmFSUpdFileName
         |--(2) hmFSConfFileName
         |--(3) hmFSLogFileName
         |--(4) hmFSUserName
         |--(5) hmFSTPPassword
         |--(6) hmFSAction
         |--(8) hmFSActionResult
         |--(9) hmFSConfigState
      |--(5) hmTempTable (5)
         |--(1)  hmTemperature
         |--(2)  hmTempUprLimit
         |--(3) hmTempLwrLimit
      |--(6) hmNeighbourAgentTable
         |--(1) hmNeighbourEntry
            |--(1) hmNeighbourSlot
            |--(2) hmNeighbourIpAddress
      |--(7) hmAuthGroup
         |--(1) hmAuthHostTableEntriesMax
         |--(2) hmAuthCommTableEntriesMax
         |--(3) hmAuthCommTable
            |--(1) hmAuthCommEntry
               |--(1) hmAuthCommIndex
               |--(2) hmAuthCommName
               |--(3) hmAuthCommPerm
               |--(4) hmAuthCommState
         |--(4) hmAuthHostTable
            |--(1) hmAuthHostEntry
```

```
                |--(1) hmAuthHostIndex
                |--(2) hmAuthHostName
                |--(3) hmAuthHostCommIndex
                |--(4) hmAuthHostIpAddress
                |--(5) hmAuthHostIpMask
                |--(6) hmAuthHostState
    |--(8) hmTrapGroup
        |--(1) hmTrapCommTableEntriesMax
        |
        |--(2) hmTrapDestTableEntriesMax
        |
        |--(3) hmTrapCommTable
            |--(1) hmTrapCommEntry
                |--(1) hmTrapCommIndex
                |--(2) hmTrapCommCommIndex
                |--(3) hmTrapCommColdStart
                |--(4) hmTrapCommLinkDown
                |--(5) hmTrapCommLinkUp
                |--(6) hmTrapCommAuthentication
                |--(7) hmTrapCommBridge
                |--(8) hmTrapCommRMON
                |--(9) hmTrapCommUsergroup
                |--(10)hmTrapCommDualHoming
                |--(11)hmTrapCommChassis
                |--(12)hmTrapCommState
        |--(4) hmTrapDestTable
            |--(1) hmTrapDestEntry
                |--(1) hmTrapDestIndex
                |--(2) hmTrapDestName
                |--(3) hmTrapDestCommIndex
                |--(4) hmTrapDestIpAddress
                |--(6) hmTrapDestState
    |--(9) hmLastAccessGroup
        |--(1) hmLastIpAddr
        |--(2) hmLastPort
|--(3) userGroup
    |--(1) userGroupTable
        |--(1) userGroupEntry
            |--(1) userGroupID
            |--(2) userGroupDescription
            |--(3) userGroupRestricted
            |--(4) userGroupSecAction
    |--(2) userGroupMemberTable
        |--(1) userGroupMemberEntry
```

```
            |--(1) userGroupMemberGroupID
            |--(2) userGroupMemberUserID
      |--(3) userTable
         |--(1) userEntry
            |--(1) userID
            |--(2) userRestricted
      |--(4) portSecurityTable
         |--(1) portSecurityEntry
            |--(1) portSecSlotID
            |--(2) portSecPortID
            |--(3) portSecPermission
            |--(4) portSecAllowedUserID
            |--(5) portSecAllowedGroupIDs
            |--(6) portSecConnectedUserID
            |--(7) portSecAction
      |--(5) userGroupSecurityAction
   |--(4) hmDualHoming
      |--(1) hmDualHomingTable
         |--(1) hmDuHmEntry
            |--(1) hmDuHmPrimGroupID
            |--(2) hmDuHmPrimIfIndex
            |--(3) hmDuHmPrimIfOpState
            |--(4) hmDuHmRedGroupID
            |--(5) hmDuHmRedIfIndex
            |--(6) hmDuHmRedIfOpState
            |--(7) hmDuHmDesiredAction
            |--(8) hmDuHmOperState
            |--(9) hmDuHmPortRevivalDelay
            |--(10) hmDuHmLinkMode
            |--(11) hmDuHmRedCheckEnable
            |--(12) hmDuHmRedCheckState
```

A Appendix

A.1 FAQ

Answers to frequently asked questions can be found at the Hirschmann Web site:

`www.hirschmann.com`

Inside Network and Automaition Solutions is located on the pages `SERVICES` the area `FAQ`.

A.2 Literature references

[1] „Optische Übertragungstechnik
in der Praxis“
Christoph Wrobel
Hüthig Buch Verlag Heidelberg
ISBN 3-7785-2238-3

[2] Hirschmann Manual
“Management MIKE”
943 416-001

[3] Hirschmann Manual
“Management FCMA”
943 378-002

[4] Hirschmann Manual
“MultiLAN Switch”
943 309-001

[5] Hirschmann Manual
“ETHERNET”
943 320-001

[6] Hirschmann Manual
“FDDI”
943 395-001

[7] Hirschmann Manual
“Token Ring”
943 397-001

[8] Hirschmann Manual
“ATM LAN Switch”
943 470-102

A.3 Reader's comments

What is your opinion of this manual? We constantly strive to provide as comprehensive a description as possible of our product and provide important information to ensure trouble-free operation. Your comments and suggestions help us improve the quality of our documentation.

Your **assessment** of this manual:

	excellent	good	satisfactory	adequate	poor
Accuracy	O	O	O	O	O
Readability	O	O	O	O	O
Comprehensibility	O	O	O	O	O
Examples	O	O	O	O	O
Structure/Layout	O	O	O	O	O
Completeness	O	O	O	O	O
Graphics	O	O	O	O	O
Drawings	O	O	O	O	O
Tables	O	O	O	O	O

Did you discover an error in the manual?
If so, on what page?

....

....

....

....

....

....

....

Suggestions for improvement and additional information:

....

....

....

....

General comments:

....

....

....

....

Company / Department ....

Name / Telephone number ....

Street ....

Zip code / City ....

Date / Signature ....

Dear user,

Please fill out and return this page

- as a fax to the number +49 (0)7127/14-1542 or
- by mail to

  Hirschmann Electronics GmbH & Co. KG
  Department AID
  Stuttgarter Str. 45 - 51
  72654 Neckartenzlingen
  Germany

A.4 Stichwortverzeichnis