KNX Association

IP Communication

Table of Contents

1 IP Communication

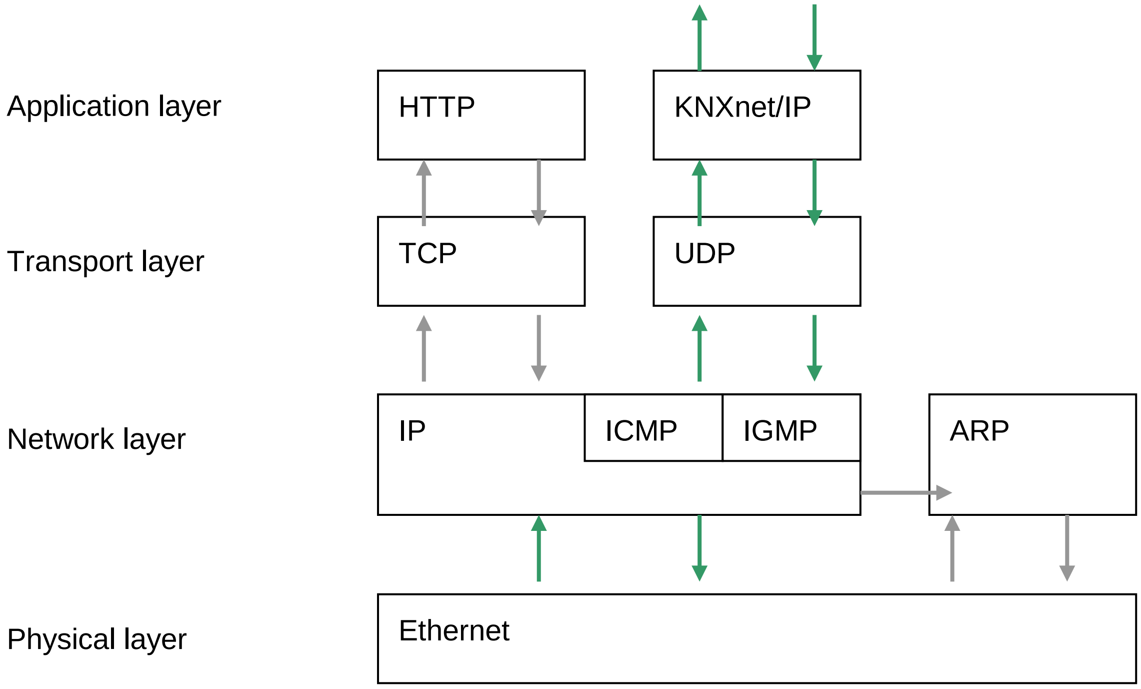
The data transfers described in the “Serial Data Communication” chapter refer to communication via TP1. So-called IP couplers or IP routers are however increasingly used in current KNX installations. The basic functionality and areas of application are already described in the “Couplers” chapter of the advanced training course. The implementation of the KNXnet/IP protocol is explained in more detail in the following chapter.

1.1 Protocol

IP communication in KNX is also structured according to the OSI reference model. Communication takes place via the application layer which generates the KNXnet/IP telegram, the transport layer (UDP), the network layer (IP) and the Ethernet as a physical layer.

Additional information for the respective layer (the header) is always added to the KNXnet/IP information in a similar way as for the TP1 protocol.

Ethernet header	IP header	UDP header	KNXnet/IP



1.1.1 HTTP (Hypertext Transfer Protocol):

The “http” protocol is mainly used to load web pages (text, images and other multimedia data) from the World Wide Web (WWW) into a web browser.

1.1.2 TCP (Transmission Control Protocol):

TCP is located in the transport layer and is used for connection-oriented data transfer.

1.1.3 UDP (User Datagram Protocol):

UDP like TCP is located in the transport layer. It is a minimal, connectionless network protocol with the task of assigning data to the correct application.

1.1.4 IP (Internet Protocol):

IP is located in the network layer and thus forms the first layer which is independent of the transmission medium. Various devices can be classified into logical units (subnets) via their IP addresses and subnet masks. Via this classification, it is possible to address devices in larger networks and establish links to them, as logical addressing is the basis for routing.

1.1.5 ICMP (Internet Control Message Protocol):

ICMP is used for the exchange of information and error messages via the Internet Protocol (IP).
ICMP bases on the IP protocol i.e. an ICMP package is stored in the data part of the IP protocol.

1.1.6 IGMP (Internet Group Management Protocol):

IGMP is based on the IP protocol and enables group communication via IPv4-multicasting, i.e. the simultaneous distribution of IP packages via one IP address. Groups can be managed dynamically. Management takes place in the routers and/or switches, to which receivers of a multicast group are connected. IGMP is mainly needed in larger networks. In this way, it is avoided that the transmission of multicast addresses leads to a network overload.
Example: When ETS in the “ETS Connection Manager” polls for available IP routers or IP gateways, this will be done via the KNX multicast IP. Subsequently, all KNX IP devices within reach will reply. The router / switch located in between will now bundle all replying devices to a multicast group address. This implies that future multicast transmissions will always be routed to the KNX devices only and not anymore to all network devices.

1.1.7 ARP (Address Resolution Protocol):

ARP enables the assignment of network addresses to hardware addresses.

1.2 Addressing of Network Devices

A network device has at least two addresses:

- The MAC address (Media Access Control) is linked to the hardware and cannot be modified. It is globally unique. It consists of 6 bytes with the following format: 00-50-C2-55-40-00
- The IP address (Internet Protocol) is assigned by the administrator or automatically via DHCP. In the case of IPv4 (Internet Protocol version 4), it consists of 4 bytes (IPv6 is currently not used in KNXnet/IP) and has the format: 192.168.1.1.

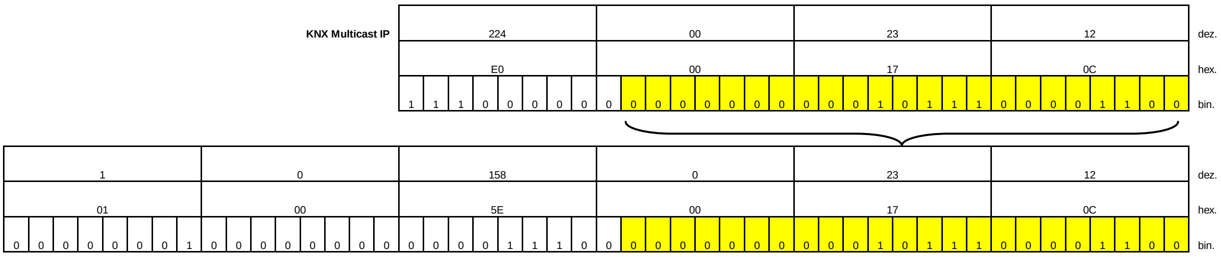
The IP address can be compared to the physical address in KNX. It is entered in the memory of the network device.
The MAC address represents a unique “serial number”. It is the IP address of the hardware and permanently linked to it.

1.3 KNX Multicast address

In case of multicast, a defined MAC address consisting of 6 bytes is sent in the network layer as destination address. This address always starts with 01-00-5E- and is then supplemented by a so-called 4 byte long multicast IP. In IPv4 the address range from 224.0.0.0 to 239.255.255.255 has been defined for a wide variety of network – applications[1]. The multicast IP address **224.0.23.12** is reserved for KNXnet/IP.

A MAC address however always consists of 6 bytes. That’s why a 4 byte multicast IP address can’t be padded to the predefined 3 bytes (01-00-5E-) as otherwise the address would be 7 bytes long. The padding is therefore realised as shown below:

The lower 23 bits of the multicast IP address are inserted in the MAC address, resulting in MAC addresses ranging from 01-00-5E-00-00-00 to 01-00-5E-7F-FF-FF. It is possible that several multicast IP addresses are mapped on the same MAC address (for example 224.0.0.1 and 233.128.0.1).

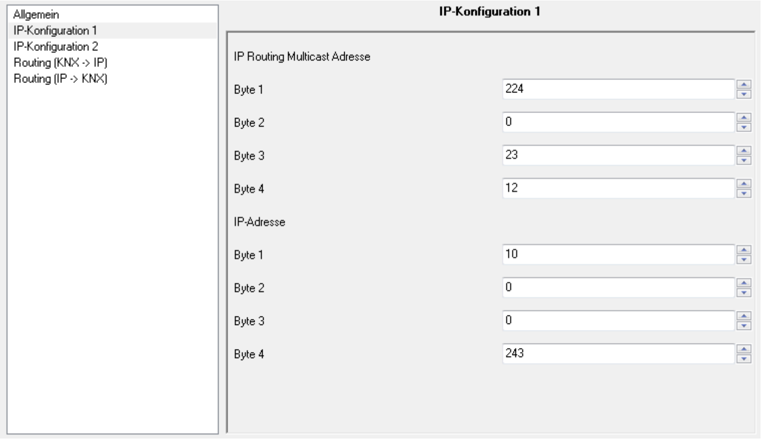


KNX Multicast IP																	224								00								23								12								dez.
																	E0								00								17								0C								hex.
																	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	1	1	0	0	bin.
	1								0								158								0								23								12								dez.
	01								00								5E								00								17								0C								hex.
	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	1	1	0	0	bin.

The complete multicast MAC address in KNXnet/IP is therefore 01-00-5E--00-17-0C.

[1] For the coordination the protocol IGMP is used in IPv4

It is possible to change the predefined KNX multicast IP addresses in the parameters of KNXnet/IP devices.



IP-Konfiguration 1

IP Routing Multicast Adresse

Byte 1 224

Byte 2 0

Byte 3 23

Byte 4 12

IP-Adresse

Byte 1 10

Byte 2 0

Byte 3 0

Byte 4 243

In this way it is possible to establish communication between different, from each other separated KNX installations within the same IP network. Please note that all IP routers of an installation are listening in on the same multicast IP address.

1.4 KNXnet/IP

The KNXnet/IP telegram contains some additional information compared to the TP1 telegram.

KNXnet/IP				
Header Length	Protocol Version	Service Type Identifier	Total Length	KNXnet/IP Body

1.4.1 Header Length (1 byte)

The header size is always the same. It is transmitted nevertheless as the size may change with further versions of the protocol. It is used to find the start of the total KNXnet/IP frame. The header size is indicated in bytes.
The transmitted value is 06_{hex}.

1.4.2 Protocol Version (1 byte)

The protocol version indicates the status of the KNXnet/IP protocol. It is binary coded and is currently version 1.0.
The transmitted value is 10_{hex}.

1.4.3 KNXnet/IP Service Type Identifier (2 byte)

The KNXnet/IP type identifier indicates which action should be carried out. The following table shows defined address ranges and actions.

Address range in hex		KNXnet/IP action
From	To	
0200	020F	KNXnet/IP Core
0310	031F	KNXnet/IP Device Management
0420	042F	KNXnet/IP Tunnelling
0530	053F	KNXnet/IP Routing
0600	06FF	KNXnet/IP Remote Logging
0740	07FF	KNXnet/IP Remote Configuration and Diagnosis
0800	08FF	KNXnet/IP Object Server

A detailed list of all the defined addresses can be found in the KNX specifications, volume 3 (System Specifications), part 8 (KNXnet/IP), chapter 1.

1.4.4 Total Length (2 bytes)

The total length of the KNXnet/IP frame is indicated in bytes in the “Total Length” field. The bytes of the previous fields (Header Length, Protocol Version and Service Type Identifier) are also part of the total length. If the total number of bytes transmitted is greater than 252 bytes, the first “Total Length” byte is set to FF (255). Only in this case the second byte includes additional length information.

1.4.5 KNXnet/IP Body (variable)

The KNXnet/IP Body describes the useful data within a KNXnet/IP frame. The useful data adopts a different structure depending on the information content which should be sent. It consists of a connection header, including the Message Code (1 byte) and an additional length information (1 byte). Furthermore, the IP Body contains the so-called cEMI frame.

KNXnet/IP-Body	
Connection Header	cEMI Frame

The cEMI frame basically contains the TP1 telegram structure. The check field is not used in IP communication as error detection is done via the IP protocol. Moreover a second control field (1 byte) is included behind the TP1 control field. In the second control field the Type of the destination address (1 bit) and the routing counter (3 bits) are included. In the last 4 bits there the so-called Extended Frame format (EFF) can be given, e.g:

0000_{bin}: for standard frames

$01xx_{bin}$: for LTE frames (see serial data communication and KNX protocol)

As the type of the destination address and the routing counter are already described in the second control field, the first 4 bits of the N_PDU are set to 0000.

									Kontrollfeld		Quelladresse		Zieladresse		Typ Zieladr.(1b); Rout.Zähl.(3b); Längenf.(4b)	TPCI/APCI & data		Prüffeld
TP1:									BC		11	04	00	01	E1	00	80	CS
IP:	06	10	05	30	00	11	29	00	BC	D0	11	04	00	01	01	00	80	
	Header lebgth.	Protokoll Version.	Servive Type identifier.		Total length (KNXnet/IP).		Message Code.	Länge zusätzlicher Infos (zukünftige Anw.).	Kontrollfeld.	Typ Zieladr.(1b); Rout.Zähl.(3b); Ext. Frame Format(4b).	Quelladresse.		Zieladresse.		0000 + Längenfeld(4b).	TPCI/APCI & data.		

Example: TP1 Telegram compared to KNXnet/IP telegrams

1.5 Tunnelling

Tunnelling is necessary for the connection-oriented transmission of KNX telegrams in an IP frame with ETS. This is always the case if a physical address should be addressed as a target address i.e. when programming the physical address or downloading the application program of KNX devices. Communication in tunnelling is always carried out via the IP address of the KNXnet/IP device that is to be “tunnelled”.

1.5.1 Address Range

The tunnelling variant is always used when a KNXnet/IP device should be addressed directly via Ethernet e.g. to load the application into a TP1 device. In this case, there is an address range of 0420_{hex} to $042F_{hex}$ in the KNXnet/IP type identifier field. Two addresses are currently defined:

- 0421_{hex} : TUNNELLING_REQUEST
- 0421_{hex} : TUNNELLING_ACK

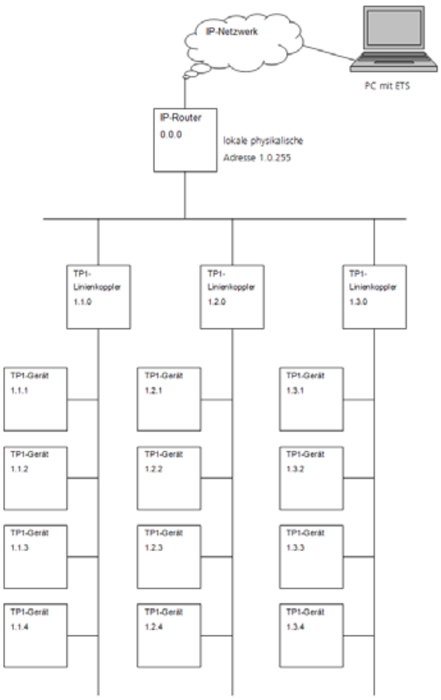
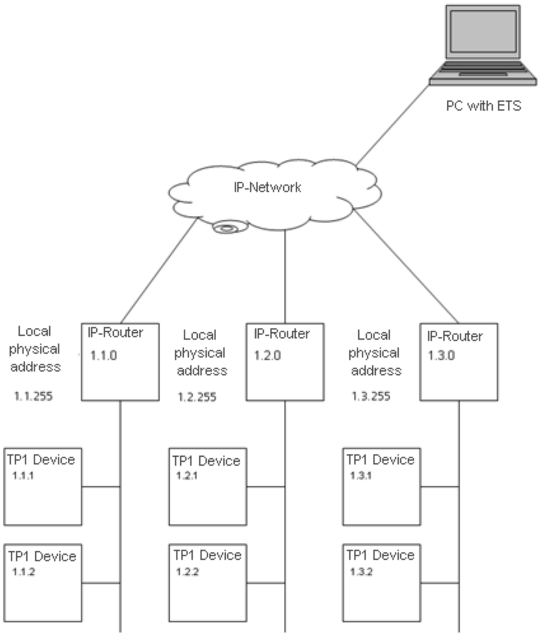
The TUNNELLING_REQUEST address is used to send and receive individual KNX telegrams between KNXnet/IP Client and Server.

The TUNNELLING_ACK address is sent by a KNXnet/IP device to confirm the receipt of TUNNELLING_REQUEST information.

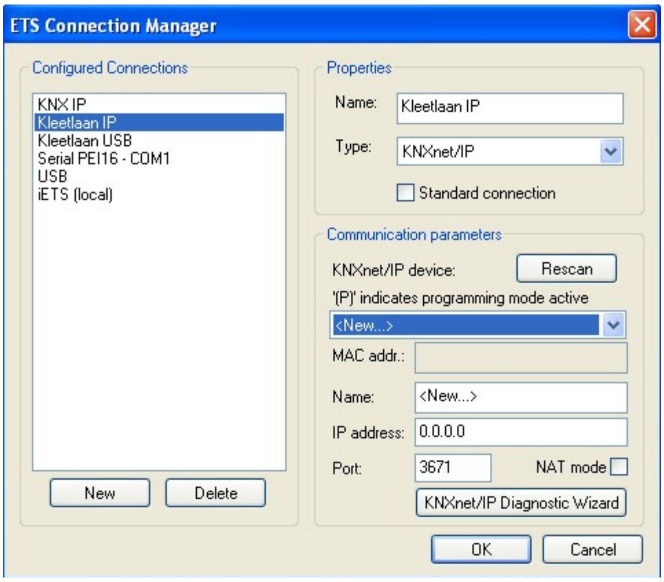
1.5.2 Tunnelling Examples

Two examples of tunnelling are shown in the following section.

- Example 1: IP gateway in the main/backbone line as a universal coupler
- Example 2: IP router as a coupler between TP1 lines

Example 1	Example 2
	

If a device in a TP1 line should be programmed by ETS via Ethernet in the two examples above, the settings of the interface should always be performed at first.



ETS Connection Manager

Configured Connections

KNX IP
Kleetlaan IP
Kleetlaan USB
Serial PEI16 - COM1
USB
iETS (local)

Properties

Name: Kleetlaan IP

Type: KNXnet/IP

☐ Standard connection

Communication parameters

KNXnet/IP device:

'(P)' indicates programming mode active

<New...>

MAC addr.:

Name: <New...>

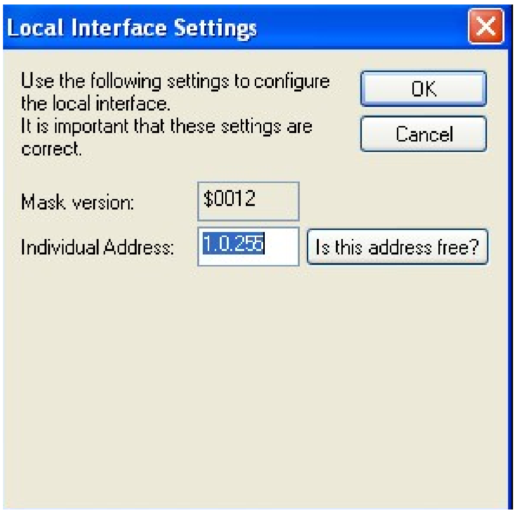
IP address: 0.0.0.0

Port: 3671 NAT mode ☐

Figure 1: ETS Connection Manager

To do so, the ETS Connection Manager shall be opened, select type “KNXnet/IP” (“KNXnet/IP”) and press the “Scan” button. ETS will list all connected KNXnet/IP devices.

If several KNXnet/IP devices are displayed (e.g. the different IP routers, if they are functioning as main/backbone couplers), it is now important to select precisely the KNXnet/IP device that is connected to the TPI line (area) of the device to be programmed. The local physical address must then be set.



Local Interface Settings

Use the following settings to configure the local interface.
It is important that these settings are correct.

Mask version: $0012

Individual Address: 1.0.255

Figure 2: Local Interface Settings

It is important that the address matches the line or area of the device to be addressed. Moreover, the address shall currently not be used in the installation or project. It is advised that to select 255 as device number as it probably does not exist in the installation.

An IP router of line 1.1 thus has physical address 1.1.0 as coupler address and address 1.1.255 as local physical address.

1.6 Routing

Routing is necessary for connectionless and simultaneous transmission of KNX telegrams to several devices via an IP coupler or IP router. This corresponds to group communication in TP1 communication.

1.6.1 Address Range

There is an address range of 0530_{hex} to $053F_{hex}$ within a routing telegram in the KNXnet/IP type identifier field. Two addresses are currently defined:

- 0530_{hex} : ROUTING_INDICATION
- 0531_{hex} : ROUTING_LOST_MESSAGE

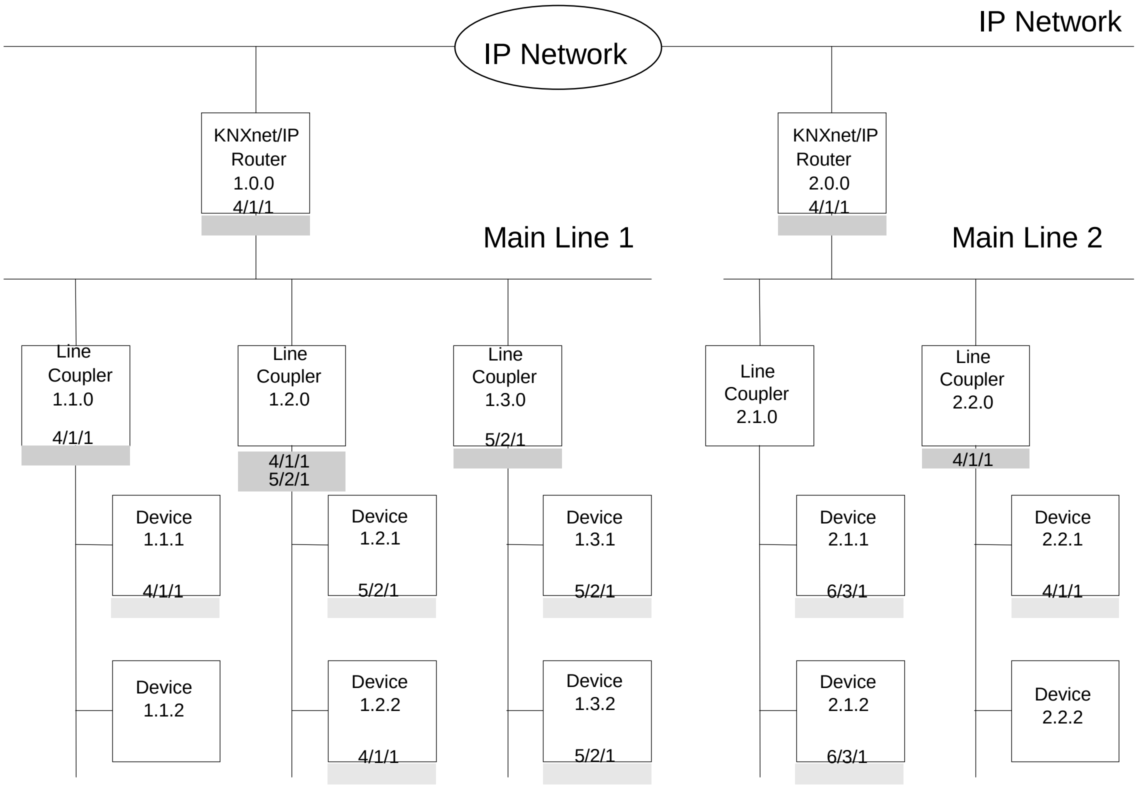
ROUTING_INDICATION is used to send KNX telegrams via IP networks without a connection.

ROUTING_LOST_MESSAGE is used to display any KNXnet/IP routing information that has been lost (connectionless).

1.6.2 Routing Example

If IP routers are used as line or backbone couplers in a KNX installation, they route a TP1 telegram to the primary IP side if the corresponding group address is entered in the filter table and the parameter is set to “filter”. The telegram is then sent via UDP with the multicast address 224.0.23.12 (port 3671) defined for KNX. A successful transmission is not confirmed. Data loss on the IP side is however relatively unlikely.

An IP router that receives a KNXnet/IP telegram on its primary IP side via the KNX multicast address will evaluate the KNXnet/IP telegram and only route it on its secondary TP1 side if the corresponding group address is entered in the filter table and the parameter is set to “filter”.



Note:
Direct communication between two IP devices is normally only possible via the respective IP addresses of the IP devices. That means that if an IP device sends information to another IP device, the target address is always the IP address of the recipient.

In the case of KNXnet/IP routing, the network structure which was pre-assigned by the IP and subnet addresses is removed by using the multicast IP. If two networks should now be linked together via the KNX multicast IP, it should be ensured that intermediate network routers also let the multicast IP through. By default, a network router will first block all multicast addresses so that no attacks can be carried out externally. This also applies to newer network switches. Preset filter functions can prevent the multicast IP from being let through.