

DIN EN 13321-2



ICS 35.240.99; 97.120

Ersatz für
DIN EN 13321-2:2007-01

**Offene Datenkommunikation für die Gebäudeautomation und
Gebäudemanagement –
Elektrische Systemtechnik für Heim und Gebäude –
Teil 2: KNXnet/IP-Kommunikation;
Deutsche Fassung EN 13321-2:2012**

Open Data Communication in Building Automation, Controls and Building Management –
Home and Building Electronic Systems –
Part 2: KNXnet/IP Communication;
German version EN 13321-2:2012

Réseau ouvert de communication de données pour l'automatisation, la régulation et la
gestion technique du bâtiment –
Systèmes électroniques pour la maison et le bâtiment –
Partie 2: Communication KNXnet/IP;
Version allemande EN 13321-2:2012

Gesamtumfang 181 Seiten

Normenausschuss Heiz- und Raumluftechnik (NHRS) im DIN
DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE

Nationales Vorwort

Dieses Dokument (EN 13321-2:2012) wurde vom Technischen Komitee CEN/TC 247 „Gebäudeautomation und Gebäudemanagement“ erarbeitet, dessen Sekretariat vom SNV (Schweiz) gehalten wird.

Für die deutsche Mitarbeit ist der Arbeitsausschuss NA 041-03-68 AA „Gebäudeautomation und -kommunikation (SpA CEN/TC 247/WG 4)“ des Normenausschusses Heiz- und Raumluftechnik (NHRS) verantwortlich.

Änderungen

Gegenüber DIN EN 13321-2:2007-01 wurden folgende Änderungen vorgenommen:

- a) Die Norm wurde redaktionell überarbeitet.

Frühere Ausgaben

DIN V EN V 13321-2: 2000-05
DIN EN 13321-2: 2007-01

Deutsche Fassung

Offene Datenkommunikation für die Gebäudeautomation und
Gebäudemanagement —
Elektrische Systemtechnik für Heim und Gebäude —
Teil 2: KNXnet/IP-Kommunikation

Open Data Communication in Building Automation,
Controls and Building Management —
Home and Building Electronic Systems —
Part 2: KNXnet/IP Communication

Réseau ouvert de communication de données pour
l'automatisation, la régulation et la gestion technique du
bâtiment —
Systèmes électroniques pour la maison et le bâtiment -
Partie 2: Communication KNXnet/IP

Diese Europäische Norm wurde vom CEN am 30. September 2012 angenommen.

Die CEN-Mitglieder sind gehalten, die CEN/CENELEC-Geschäftsordnung zu erfüllen, in der die Bedingungen festgelegt sind, unter denen dieser Europäischen Norm ohne jede Änderung der Status einer nationalen Norm zu geben ist. Auf dem letzten Stand befindliche Listen dieser nationalen Normen mit ihren bibliographischen Angaben sind beim Management-Zentrum des CEN-CENELEC oder bei jedem CEN-Mitglied auf Anfrage erhältlich.

Diese Europäische Norm besteht in drei offiziellen Fassungen (Deutsch, Englisch, Französisch). Eine Fassung in einer anderen Sprache, die von einem CEN-Mitglied in eigener Verantwortung durch Übersetzung in seine Landessprache gemacht und dem Management-Zentrum mitgeteilt worden ist, hat den gleichen Status wie die offiziellen Fassungen.

CEN-Mitglieder sind die nationalen Normungsinstitute von Belgien, Bulgarien, Dänemark, Deutschland, der ehemaligen jugoslawischen Republik Mazedonien, Estland, Finnland, Frankreich, Griechenland, Irland, Island, Italien, Kroatien, Lettland, Litauen, Luxemburg, Malta, den Niederlanden, Norwegen, Österreich, Polen, Portugal, Rumänien, Schweden, der Schweiz, der Slowakei, Slowenien, Spanien, der Tschechischen Republik, der Türkei, Ungarn, dem Vereinigten Königreich und Zypern.



EUROPÄISCHES KOMITEE FÜR NORMUNG
EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION

Management-Zentrum: Avenue Marnix 17, B-1000 Brüssel

Inhalt

Seite

Vorwort	4
Einleitung.....	5
1 Anwendungsbereich	7
2 Normative Verweisungen	8
3 Begriffe	8
4 Symbole und Abkürzungen	10
5 Anforderungen	11
5.1 Abschnitt 1: Überblick.....	11
5.1.1 Abschnitte des KNXnet/IP-Dokuments.....	11
5.1.2 Obligatorische und optionale Implementierungen von IP-Protokollen.....	13
5.1.3 Sicherheitsbetrachtungen	15
5.2 Abschnitt 2: Core	17
5.2.1 Anwendungsbereich	17
5.2.2 KNXnet/IP-Frames.....	18
5.2.3 Host-Protokoll-Unabhängigkeit.....	19
5.2.4 Erkennung und Selbstbeschreibung	20
5.2.5 Kommunikationskanäle	22
5.2.6 Allgemeine Implementierungsrichtlinien	25
5.2.7 Datenpaketstrukturen.....	29
5.2.8 IP-Netzwerke.....	43
5.2.9 Zertifizierung	49
5.3 Abschnitt 3: Device-Management-Spezifikation.....	50
5.3.1 Anwendungsbereich	50
5.3.2 KNXnet/IP Device Management.....	50
5.3.3 Implementierungsregeln und -richtlinien	62
5.3.4 Datenpaketstrukturen.....	64
5.3.5 Zertifizierung	67
5.4 Abschnitt 4: Tunnelling	68
5.4.1 Anwendungsbereich	68
5.4.2 Tunnelling von KNX-Telegrammen	68
5.4.3 Konfiguration und Management.....	72
5.4.4 Frame-Strukturen.....	72
5.4.5 Zertifizierung	75
5.5 Abschnitt 5: Routing.....	75
5.5.1 Anwendungsbereich	75
5.5.2 KNXnet/IP-Routing von KNX-Telegrammen	76
5.5.3 Implementierungsregeln und -richtlinien	83
5.5.4 Konfiguration und Management.....	85
5.5.5 Datenpaketstrukturen.....	86
5.5.6 Zertifizierung	88
5.6 Abschnitt 6: Ferndiagnose und Konfiguration	89
5.6.1 Anwendungsbereich	89
5.6.2 Ferndiagnose von KNXnet/IP-Geräten.....	89
5.6.3 Konfiguration und Management.....	90
5.6.4 Datenpaketstrukturen.....	90
5.6.5 Zertifizierung	96
Anhang A (normativ) Liste der Codes.....	97
Anhang B (informativ) Beispiele von KNXnet/IP-IP-Frames im Binärformat.....	106
Anhang C (normativ) KNXnet/IP-Parameterobjekt.....	126

Anhang D (normativ) cEMI (Common External Messaging Interface)	129
Anhang E (normativ) Koppler-Ressourcen	165
Literaturhinweise	179

Vorwort

Dieses Dokument (EN 13321-2:2012) wurde vom Technischen Komitee CEN/TC 247 „Gebäudeautomation und Gebäudemanagement“ erarbeitet, dessen Sekretariat vom SNV gehalten wird.

Diese Europäische Norm muss den Status einer nationalen Norm erhalten, entweder durch Veröffentlichung eines identischen Textes oder durch Anerkennung bis Juni 2013, und etwaige entgegenstehende nationale Normen müssen bis Juni 2013 zurückgezogen werden.

Es wird auf die Möglichkeit hingewiesen, dass einige Texte dieses Dokuments Patentrechte berühren können. CEN [und/oder CENELEC] sind nicht dafür verantwortlich, einige oder alle diesbezüglichen Patentrechte zu identifizieren.

Dieses Dokument ersetzt EN 13321-2:2006.

Während sich ENV 13321-2:2000 mit der Übertragung von EIB-Paketen über Ethernet einschließlich der Frame-Codierung befasste, beschreibt dieses Dokument die Übertragung von ESHG-Paketen mittels Internet Protocol. Einzelheiten über die ESHG-Paketframes sind in EN 13321-1 „*Offene Datenkommunikation für die Gebäudeautomation und Gebäudemanagement — Elektrische Systemtechnik für Heim und Gebäude — Teil 1: Produkt- und Systemanforderungen*“ enthalten, sodass die ESHG-Paketframes in diesem Dokument nicht ausdrücklich beschrieben werden müssen.

Dieses Dokument ist Teil 2 der Reihe EN 13321 Europäischer Normen unter dem allgemeinen Titel „*Offene Datenkommunikation für die Gebäudeautomation und Gebäudemanagement — Elektrische Systemtechnik für Heim und Gebäude*“, die aus den folgenden Teilen besteht:

— *Teil 1: Produkt- und Systemanforderungen*

— *Teil 2: KNXnet/IP-Kommunikation*

Entsprechend der CEN/CENELEC-Geschäftsordnung sind die nationalen Normungsinstitute der folgenden Länder gehalten, diese Europäische Norm zu übernehmen: Belgien, Bulgarien, Dänemark, Deutschland, die ehemalige jugoslawische Republik Mazedonien, Estland, Finnland, Frankreich, Griechenland, Irland, Island, Italien, Kroatien, Lettland, Litauen, Luxemburg, Malta, Niederlande, Norwegen, Österreich, Polen, Portugal, Rumänien, Schweden, Schweiz, Slowakei, Slowenien, Spanien, Tschechische Republik, Türkei, Ungarn, Vereinigtes Königreich und Zypern.

Einleitung

Diese Europäische Norm wurde für den Entwurf neuer Gebäude und die Sanierung vorhandener Gebäude entwickelt, um ein angenehmes Raumklima sowie Energieerhaltung und Energieeffizienz zu erzielen.

Diese Norm legt die Einbindung von KNX-Protokoll-Implementierungen innerhalb von Internet Protocol (IP) namens KNXnet/IP fest.

Sie legt ein Standardprotokoll fest, das innerhalb von KNX-Geräten, der ETS (Engineering Tool Software) und anderen Implementierungen eingesetzt wird, um den KNX-Datenaustausch über IP-Netzwerke zu unterstützen. KNXnet/IP stellt im Grunde genommen ein allgemeines System bereit, das mehrere spezialisierte „Dienstprotokolle“ in einer modularen und erweiterbaren Form einschließt.

Die KNXnet/IP-Norm umfasst die folgenden Abschnitte:

- Abschnitt 1, Überblick;
- Abschnitt 2, Core;
- Abschnitt 3, Device Management;
- Abschnitt 4, Tunnelling;
- Abschnitt 5, Routing;
- Abschnitt 6, Ferndiagnose und Konfiguration.

In der Zukunft werden der KNXnet/IP-Norm möglicherweise zusätzliche Abschnitte hinzugefügt; dann müssen auch der Abschnitt 1 „Überblick“ sowie Anhang A aktualisiert werden.

KNXnet/IP unterstützt unterschiedliche Software-Implementierungen, die auf dem Protokoll aufsetzen. Im Einzelnen können diese Software-Implementierungen Gebäudemanagement-, Facility-Management-, Energiemanagement- oder einfach Datenbank- und SCADA-Pakete (Supervisory Control and Data Acquisition) sein.

Die meisten dieser Pakete müssen für die spezifische Benutzeranwendung konfiguriert werden. Um diesen Prozess zu vereinfachen und Engineering-Kosten zu senken, stellt KNXnet/IP einfache Engineering-Schnittstellen, nämlich eine „Beschreibungssprache“ für das zugrunde liegende KNX-System, bereit. Die Konfiguration kann offline durchgeführt werden, z. B. indem eine ETS-Exportdatei generiert wird, oder online mithilfe eines Mechanismus, der das zugrunde liegende KNX-System selbst beschreibt (d. h. Daten aus dem System selbst liest).

Im Zusammenhang mit der Abbildung von EIB/KNX auf BACnet, die in EN ISO 16484-5 beschrieben wird, können EIB/KNX-Installationen auf einfache Weise in BACnet-Systemumgebungen integriert werden.

KNXnet/IP unterstützt:

- die schnelle Umschaltung zwischen Betriebsmodi (Konfiguration, Betrieb);
- ereignisgesteuerte Mechanismen;
- Verbindungen mit einer Verzögerungszeit größer als $t_{\text{EIB_Übertragung_Zeitüberschreitung}}$ (z. B. eine Netzwerkverbindung über Satellit).

Abschnitt 1, Überblick

Abschnitt 1 „Überblick“ gibt einen allgemeinen Überblick über KNXnet/IP und beinhaltet Sicherheitsbetrachtungen.

Abschnitt 2, Core

Abschnitt 2 „Core“ legt ein Standardprotokoll fest, das innerhalb von KNXnet/IP-Geräten und der Engineering Tool Software implementiert wird, um den KNX-Datenaustausch über IP-Netzwerke zu unterstützen.

Diese spezifische Implementierung des Protokolls über Internet Protocol (IP) wird „KNXnet/IP“ genannt.

Diese Norm befasst sich mit:

- der Festlegung von Datenpaketen, die über das IP-Host-Protokoll-Netzwerk für KNXnet/IP-Kommunikation gesendet werden;
- der Erkennung und Selbstbeschreibung von KNXnet/IP-Servern;
- der Konfiguration und der Einrichtung eines Kommunikationskanals zwischen einem KNXnet/IP-Client und einem KNXnet/IP-Server.

Abschnitt 3, Device Management

Abschnitt 3 „Device Management“ legt Dienste für die Fernkonfiguration und die Fernverwaltung von KNXnet/IP-Servern fest.

Abschnitt 4, Tunnelling

Abschnitt 4 „Tunnelling“ legt Dienste für einen Punkt-zu-Punkt-Austausch von KNX-Telegrammen über ein IP-Netzwerk zwischen einem KNXnet/IP-Gerät, das als Server agiert, und einem KNXnet/IP-Client fest. Dieser Punkt-zu-Punkt-Austausch kann mithilfe eines übergeordneten Systems für Funktionen der Gebäudeautomation oder des Gebäudemanagements oder mithilfe einer Engineering Tool Software eingerichtet werden. Er unterstützt alle ETS-Funktionen für den Download, die Prüfung und die Analyse von KNX-Geräten in KNX-Netzwerken, die über KNXnet/IP-Server verbunden sind. Dies schließt Änderungen einzelner Objekteigenschaften der KNX-Geräte ein.

Tunnelling setzt voraus, dass ein Datenübertragungsrundlauf zwischen einem KNXnet/IP-Tunnelling-Client und KNXnet/IP-Servern weniger als $t_{\text{KNX_Übertragung_Zeitüberschreitungen}}$ benötigt.

Abschnitt 5, Routing

Abschnitt 5 „Routing“ legt Dienste für einen Punkt-zu-Multipunkt-Austausch von KNX-Telegrammen über ein IP-Netzwerk zwischen KNXnet/IP-Routern und/oder KNX IP-Geräten fest.

Abschnitt 6, Ferndiagnose und Konfiguration

Abschnitt 6 „Ferndiagnose und Konfiguration“ legt Dienste für einen Punkt-zu-Punkt-Austausch von KNX-Telegrammen über ein IP-Netzwerk zwischen KNXnet/IP-Routern und/oder KNX IP-Geräten fest. Die Dienste stellen ein Mittel zur Ermittlung von Kommunikationseinstellungen und ihrer Änderung bei Fernbetrieb bereit.

1 Anwendungsbereich

Diese Europäische Norm legt die Einbindung von KNX-Protokoll-Implementierungen, die auf IP-Netzwerken (Internet-Protocol) aufsetzen, namens KNXnet/IP, fest. Sie beschreibt ein Standardprotokoll für KNX-Geräte, die mit einem IP-Netzwerk verbunden sind, sogenannte KNXnet/IP-Geräte. Das IP-Netzwerk agiert in KNX-Installationen als schneller Backbone (im Vergleich zur KNX-Übertragungsgeschwindigkeit).

Der weit verbreitete Einsatz von Datennetzwerken über Internet Protocol (IP) stellt eine Möglichkeit zur Erweiterung der Kommunikation im Bereich Gebäudesteuerung über den lokalen KNX-Steuerbus hinaus dar, wodurch Folgendes möglich wird:

- Fernkonfiguration;
- Fernbetrieb (einschließlich Steuerung und Meldung);
- schnelle Schnittstelle von LAN auf KNX und umgekehrt;
- WAN-Verbindungen zwischen KNX-Systemen (bei denen ein installiertes KNX-System mindestens eine Linie darstellt).

Ein KNXnet/IP-System besteht mindestens aus den folgenden Elementen:

- einer EIB-Linie mit bis zu 64 (255) EIB-Geräten,
ODER
einem KNX-Segment (KNX-TP1, KNX-TP0, KNX-RF, KNX-PL110, KNX-PL132);
- einem Netzwerkverbindungsgerät von KNX zu IP (einem sogenannten KNXnet/IP-Server),

und üblicherweise aus zusätzlicher

- Software für entfernte Funktionen, z. B. auf einer Workstation (dies kann eine Datenbankanwendung, ein BACnet-Gebäudemanagementsystem, ein Browser usw. sein).

Bild 1 zeigt ein typisches Szenario, in dem ein KNXnet/IP-Client (z. B. eine installierte ETS) auf mehrere installierte KNX-Systeme oder KNX-Subnetze über ein IP-Netzwerk zugreift. Der KNXnet/IP-Client kann auf mehrere KNXnet/IP-Server gleichzeitig zugreifen. Durch Subnetz-Routing ist eine Kommunikation zwischen Servern möglich.

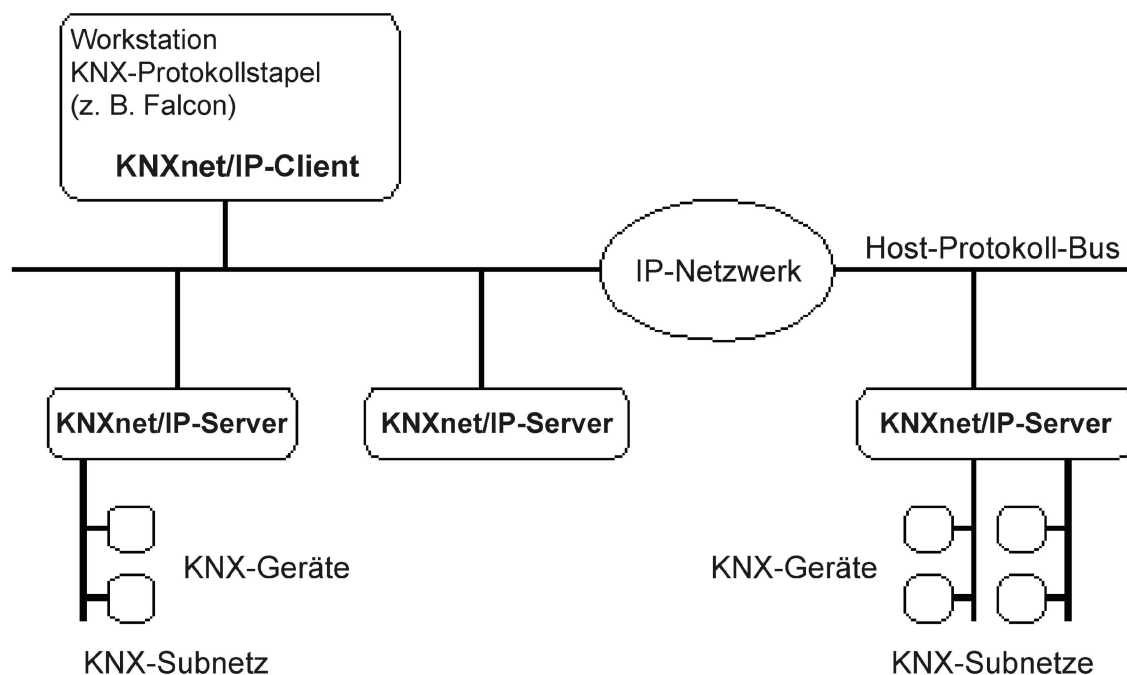


Bild 1 — Gerätetypen und Konfigurationsbeispiele

2 Normative Verweisungen

Nicht anwendbar.

3 Begriffe

Für die Anwendung dieses Dokuments gelten die folgenden Begriffe.

3.1

Subnetz

Teil eines Netzwerkes, der eine gemeinsame Adresskomponente verwendet, die sogenannte „Subnetzadresse“

Anmerkung 1 zum Begriff: Die verschiedenen Netzwerkprotokolle geben die Subnetzadresse auf unterschiedliche Weise an.

3.2

Engineering Tool Software

ETS

Software, die für die Konfiguration von KNX-Geräten verwendet wird

3.3

Host Protocol Address Information

HPAI

Struktur, die die Adressinformationen des IP-Host-Protokolls enthält, die für die Adressierung eines KNXnet/IP-Endpunkts auf einem anderen KNXnet/IP-Gerät verwendet werden

3.4

Kommunikationskanal

logische Verbindung zwischen einem KNXnet/IP-Client und einem KNXnet/IP-Server (oder, im Fall eines Routings, zwischen zwei oder mehreren KNXnet/IP-Servern)

Anmerkung 1 zum Begriff: Ein Kommunikationskanal besteht aus einer oder mehreren Verbindungen basierend auf der Definition im Host-Protokoll, die für KNXnet/IP verwendet wird.

3.5

KNX-Knoten

Gerät, das einen KNX-Protokollstapel implementiert und die Anforderungen für die Zertifizierung nach dem KNX-Standard erfüllt

3.6

KNXnet/IP-Server

KNX-Gerät mit physikalischem Zugriff auf ein KNX-Netzwerk, das das KNXnet/IP-Server-Protokoll implementiert, um mit KNXnet/IP-Clients oder (im Fall eines Routings) mit anderen KNXnet/IP-Servern über einen IP-Netzwerkkanal zu kommunizieren

Anmerkung 1 zum Begriff: Ein KNXnet/IP-Server ist seiner Konstruktion nach immer auch ein KNX-Knoten.

3.7

KNXnet/IP-Client

Anwendung, die das KNXnet/IP-Client-Protokoll verwendet, um Zugriff auf ein KNX-Subnetz über einen IP-Netzwerkkanal zu erhalten

3.8

KNXnet/IP-Gerät

Implementierung von KNXnet/IP-Diensten auf einem KNX-Knoten (KNXnet/IP-Server) oder anderer Hardware (KNXnet/IP-Client)

3.9

KNXnet/IP-Router

spezieller Typ eines KNXnet/IP-Gerätes, das KNX-Protokollpakete zwischen KNX-Subnetzen hin- und herleitet

3.10

Time To Live (de: Lebenszeit)

TTL

maximale Anzahl von IP-Routern, durch die ein Multicast-UDP/IP-Datagramm geleitet werden kann

Anmerkung 1 zum Begriff: Jeder IP-Router, durch den das Datagramm geleitet wird, verringert die TTL um eins; der lokale Hostadapter führt dies gleichermaßen durch. Wenn die TTL null erreicht, verwirft der Router das Datagramm. Wenn ein Datagramm von einem lokalen Hostadapter mit einer TTL von null gesendet wird, bedeutet dies, dass das Datagramm den Host nie verlässt. Eine TTL von eins bedeutet, dass das Datagramm das lokale Netzwerk nie verlässt (es wird nicht weitergeleitet).

3.11

KNXnet/IP Tunnelling

Dienste für einen Punkt-zu-Punkt-Austausch von KNX-Telegrammen über ein IP-Netzwerk zwischen einem KNXnet/IP-Gerät, das als Server agiert, und einem KNXnet/IP-Client

3.12

Internet Control Message Protocol

ICMP

Erweiterung von Internet Protocol (IP) um Fehler-, Steuer- und Informationsmeldungen

Anmerkung 1 zum Begriff: ICMP ist in RFC¹⁾ 92 definiert und unterstützt Pakete, die Fehler-, Steuer- und Informationsmeldungen enthalten. Der Ping-Befehl verwendet ICMP beispielsweise, um eine Internetverbindung zu testen.

1) Request for Comment: Internetstandards, die durch die Internet Engineering Task Force (IETF) festgelegt werden, werden zunächst als RFCs veröffentlicht.

3.13

Internet Group Management Protocol IGMP

Erweiterung von Internet Protocol (IP) um die Verwaltung von IP-Multicasting im Internet

Anmerkung 1 zum Begriff: IGMP ist in RFC 1112 als Standard für das IP-Multicasting im Internet festgelegt. Es wird für die Einrichtung von Host-Mitgliedschaften in bestimmten Multicast-Gruppen in einem einzelnen Netzwerk verwendet. Unter Verwendung von Host-Mitgliedschaftsberichten erlaubt es der Mechanismus des Protokolls einem Host, seinen lokalen Router darüber zu informieren, dass er Nachrichten, die an eine bestimmte Multicast-Gruppe adressiert sind, empfangen möchte. Alle Hosts, die der Stufe 2 der IP-Multicast-Spezifikation entsprechen, benötigen IGMP.

3.14

IP-Kanal

logische Verbindung zwischen zwei IP-Host/Port-Endpunkten

Anmerkung 1 zum Begriff: IP-Kanäle sind entweder garantierte, vertrauenswürdige TCP-Verbindungen (Transmission Control Protocol) oder nicht vertrauenswürdige Punkt-zu-Punkt- oder (im Fall von Routing) Multicast-UDP-Verbindungen (User Datagram Protocol).

3.15

Kommunikationskanal

Wie in der KNXnet/IP-Core-Spezifikation festgelegt, wird dieser Kanal durch einen oder zwei IP-Kanäle repräsentiert.

3.16

common External Message Interface cEMI

generische Struktur für medienunabhängige KNX-Nachrichten

Anmerkung 1 zum Begriff: cEMI-Frames (übliche EMI-Frames) werden verwendet, um KNX-Nachrichten innerhalb von IP-Paketen einzukapseln.

4 Symbole und Abkürzungen

Für die Anwendung dieses Dokuments gelten die folgenden Symbole und Abkürzungen.

In den Tabellen zu den Implementierungsanforderungen werden die folgenden Abkürzungen verwendet.

Symbol	Beschreibung
M	Obligatorisch
Cn	Bedingungen sind in der Anmerkung „n“ angegeben.
O	Optional
X	Nicht zulässig
n/a	Nicht anwendbar
R	Erforderlich
MC	Nachrichtencode
AddIL	Länge der zusätzlichen Informationen

4.1

DHCP

Dynamic Host Configuration Protocol — Kommunikationsprotokoll für die automatische Zuordnung von IP-Adresseinstellungen

4.2

DNS

Domain Name Service — weist IP-Adressen Internetnamen zu

4.3

EIB

Europäischer Installationsbus — Standard für die Gebäudesteuerung (EN 50090)

4.4

IP

Internet Protocol

4.5

KNX

Standard für die Gebäudesteuerung (EN 50090)

4.6

TCP/IP

Transmission Control Protocol over Internet Protocol — verbindungsorientierte Kommunikation über das Internet

4.7

UDP/IP

User Datagram Protocol over Internet Protocol — verbindungslose Kommunikation über das Internet

5 Anforderungen

5.1 Abschnitt 1: Überblick

5.1.1 Abschnitte des KNXnet/IP-Dokuments

5.1.1.1 Allgemeines

Diese Europäische Norm legt die Einbindung von KNX-Protokoll-Implementierungen innerhalb von Internet Protocol (IP) namens KNXnet/IP oder EIBnet/IP fest, um die ENV 13321-2:2000 (EIBnet) fortzuführen, wie durch CEN/TC 247 festgelegt. EIBnet wurde in den Fachbereich Informationstechnologie als Erweiterung von EIB eingeführt und als Technologie für die Gebäudesteuerung in CEN/TC 247 eingegliedert. EIBnet/IP ist der logische Nachfolger von EIBnet. Da EIB ein Teil von KNX geworden ist, wird dieser Standard „KNXnet/IP“ genannt.

Diese Europäische Norm legt ein Standardprotokoll fest, das innerhalb von KNX-Geräten, der Engineering Tool Software und anderen Implementierungen eingesetzt wird, um den KNX-Datenaustausch über IP-Netzwerke zu unterstützen. KNXnet/IP stellt im Grunde genommen ein allgemeines System bereit, das mehrere spezialisierte „Dienstprotokolle“ in einer modularen und erweiterbaren Form einschließt.

Die KNXnet/IP-Norm umfasst die folgenden Abschnitte:

- Abschnitt 1, Überblick;
- Abschnitt 2, Core;
- Abschnitt 3, Device Management;
- Abschnitt 4, Tunnelling;
- Abschnitt 5, Routing.

In der Zukunft werden der KNXnet/IP-Norm möglicherweise zusätzliche Abschnitte hinzugefügt; dann muss auch der Abschnitt 1 „Überblick“ aktualisiert werden.

KNXnet/IP unterstützt unterschiedliche Software-Implementierungen, die auf dem Protokoll aufsetzen. Im Einzelnen können diese Software-Implementierungen Gebäudemanagement-, Facility-Management-, Energiemanagement- oder einfach Datenbank- und SCADA-Pakete (Supervisory Control and Data Acquisition) sein.

Die meisten dieser Pakete müssen für die spezifische Benutzeranwendung konfiguriert werden. Um diesen Prozess zu vereinfachen und Engineering-Kosten zu senken, stellt KNXnet/IP einfache Engineering-Schnittstellen, nämlich eine „Beschreibungssprache“ für das zugrunde liegende KNX-System, bereit. Die Konfiguration kann offline durchgeführt werden, z. B. indem eine ETS-Exportdatei generiert wird, oder online mithilfe eines Mechanismus, der das zugrunde liegende KNX-System selbst beschreibt (d. h. Daten aus dem System selbst liest).

KNXnet/IP unterstützt:

- die schnelle Umschaltung zwischen Betriebsmodi (Konfiguration, Betrieb);
- ereignisgesteuerte Mechanismen;
- Verbindungen mit einer Verzögerungszeit größer als $t_{\text{EIB_Übertragung_Zeitüberschreitung}}$ (z. B. eine Netzwerkverbindung über Satellit).

5.1.1.2 Abschnitt 1, Überblick

Abschnitt 1 „Überblick“ beschreibt die Inhalte der einzelnen Abschnitte dieser Norm.

5.1.1.3 Abschnitt 2, Core

Abschnitt 2 „Core“ legt ein Standardprotokoll fest, das innerhalb von KNXnet/IP-Geräten und der Engineering Tool Software (ETS) implementiert wird, um den KNX-Datenaustausch über IP-Netzwerke zu unterstützen.

Diese spezifische Implementierung des Protokolls über Internet Protocol (IP) wird „KNXnet/IP“ genannt.

Diese Norm befasst sich mit:

- der Festlegung von Datenpaketen, die über das IP-Host-Protokoll-Netzwerk für KNXnet/IP-Kommunikation gesendet werden;
- der Erkennung und Selbstbeschreibung von KNXnet/IP-Servern;
- der Konfiguration und des Aufbaus eines Kommunikationskanals zwischen einem KNXnet/IP-Client und einem KNXnet/IP-Server.

5.1.1.4 Abschnitt 3, Device Management

Abschnitt 3 „Device Management“ legt Dienste für die Fernkonfiguration und die Fernverwaltung von KNXnet/IP-Servern fest.

5.1.1.5 Abschnitt 4, Tunnelling

Abschnitt 4 „Tunnelling“ legt Dienste für einen Punkt-zu-Punkt-Austausch von KNX-Telegrammen über ein IP-Netzwerk zwischen einem KNXnet/IP-Gerät, das als Server agiert, und einem KNXnet/IP-Client fest. Dieser Punkt-zu-Punkt-Austausch kann mithilfe eines übergeordneten Systems für Funktionen der Gebäudeautomation oder des Gebäudemanagements oder mithilfe einer Engineering Tool Software eingerichtet werden. Er unterstützt alle ETS-Funktionen für den Download, die Prüfung und die Analyse von KNX-Geräten in KNX-Netzwerken, die über KNXnet/IP-Server verbunden sind. Dies schließt Änderungen einzelner Objekteigenschaften der KNX-Geräte ein.

Tunnelling setzt voraus, dass ein Datenübertragungsrundlauf zwischen der ETS oder einem KNXnet/IP-Tunnelling-Client und KNXnet/IP-Servern weniger als $t_{\text{KNX_Übertragung_Zeitüberschreitungen}}$ benötigt.

5.1.1.6 Abschnitt 5, Routing

Abschnitt 5 „Routing“ legt Dienste fest, die KNX-Telegramme zwischen KNXnet/IP-Servern über das IP-Netzwerk hin- und herleiten.

5.1.2 Obligatorische und optionale Implementierungen von IP-Protokollen

5.1.2.1 Allgemeines

KNXnet/IP verwendet, wo möglich, vorhandene IP-Protokolle, sofern ihre Verwendung nicht eine unzulässige Belastung im Hinblick auf den Speicherplatz und die Implementierungsanforderungen für den vorgesehenen Dienst einschließt.

Die folgende Tabelle zeigt obligatorische (M) und optionale (O) Implementierungen von IP-Protokollen nach KNXnet/IP-Dienstarten. Obwohl sich diese Tabelle auf den KNXnet/IP-Server bezieht, ist ebenfalls angegeben, welche IP-Protokolle vom KNXnet/IP-Client implementiert werden müssen. Nicht anwendbare IP-Protokolle sind mit „n/a“ markiert.

Tabelle 1 — KNXnet/IP-Dienstarten und IP-Protokolle

IP-Protokoll	Dienstart			
	Core	Device Management	Tunnelling	Routing
ARP	M	M	M	M
RARP	O	O	O	O
Unterstützung einer festen IP-Adresse	M	M	M	M
BootP (Client) ^a	M	M	M	M
DHCP (Client) ^a	M	M	M	M
UDP	M	M	M	M
TCP	O	O	O	n/a
ICMP	M	M	M	M
IGMP	M	M	n/a	M
^a BootP/DHCP: Es muss einer der beiden Dienste auf einem KNXnet/IP-Gerät implementiert sein.				

Andere Internetprotokolle wie NTP (Network Time Protocol), FTP (File Transfer Protocol), HTTP (Hypertext Transfer Protocol), SMTP (Simple Message Transfer Protocol), DNS (Domain Name System) und SNMP (Simple Network Management Protocol) können verwendet werden, liegen aber nicht im Anwendungsbereich des KNXnet/IP-Protokolls.

5.1.2.2 Mindestanforderungen an KNXnet/IP-Geräte

KNXnet/IP-Dienstarten, wie in dieser Norm festgelegt, erfordern die Implementierung einer Mindestanzahl von IP-Protokollen für die Interaktion.

KNXnet/IP-Server müssen folgende IP-Protokolle implementieren: ARP, BootP, UDP, ICMP und IGMP. Für spezifische Dienste können weitere IP-Protokolle erforderlich sein.

5.1.2.3 Netzwerkumgebung

Da KNXnet/IP-Server IP verwenden, ist nach dieser Norm kein spezifisches Medium, das die IP-Datagramme überträgt, erforderlich.

Es wird empfohlen, ein Medium zu verwenden, das mindestens die zweifache Bitrate aller KNXnet/IP-Router, die mit diesem Medium verbunden sind, überträgt. Bei einer Punkt-zu-Punkt-Verbindung, z. B. einer PPP-Verbindung, müsste diese mindestens 19 200 bit/s betragen; bei einem Netzwerk mit bis zu 400 KNXnet/IP-Servern müsste diese mindestens 8 Mbit/s betragen.

Für KNXnet/IP-Server, die Ethernet als physikalisches Medium verwenden, wird 10BaseT als Minimum empfohlen.

5.1.2.4 Adressierung

KNXnet/IP-Server sind üblicherweise mit einem KNX-Subnetz und mit einem IP-Netzwerk verbunden. Aus diesem Grund verfügen KNXnet/IP-Server für jedes Netzwerk, mit dem sie verbunden sind, über eine eindeutige Adresse: eine individuelle KNX/EIB-Adresse und eine IP-Adresse.

Zusätzlich werden KNXnet/IP-Server einer KNXnet/IP-Projektinstallation zugewiesen, wobei für alle KNXnet/IP-Server in einer KNXnet/IP-Installation dieselbe IP-Multicast-Adresse verwendet wird.

5.1.2.5 KNXnet/IP-Geräteklassen

Ein KNXnet/IP-Gerät kann eine Softwareinstallation sein, die auf einem PC ausgeführt wird. Eine ETS oder eine andere Software, die KNXnet/IP-Dienste implementiert, wird daher als KNXnet/IP-Gerät betrachtet.

Die Festlegung von KNXnet/IP-Geräteklassen stellt die Interoperabilität zwischen KNXnet/IP-Geräten sowie eine Mindestmenge an unterstützten KNXnet/IP-Diensten für eine bestimmte KNXnet/IP-Geräteklasse sicher.

Tabelle 2 listet obligatorische und optionale Dienstarten für Geräteklassen auf.

Die Geräteklasse A umfasst Tools für die Konfiguration und Systemwartung. Mit der Ausnahme von Object-Server-Diensten müssen alle anderen KNXnet/IP-Dienste implementiert werden. ETS gehört zu dieser Geräteklasse.

Die Geräteklasse B legt eine Mindestmenge von KNXnet/IP-Diensten für KNXnet/IP-Router fest.

Die Geräteklasse C legt eine Mindestmenge von KNXnet/IP-Diensten für alle anderen KNXnet/IP-Geräte fest. Zu dieser KNXnet/IP-Geräteklasse gehören Gebäudemanagement-, Energiemanagement- und Facility-Management-Systeme.

Tabelle 2 — KNXnet/IP-Geräteklassen

Gerätekategorie	Dienststart			
	Core	Device Management	Tunnelling	Routing
A (Tools für die Konfiguration und Systemwartung)	M	M	M	M
B (KNXnet/IP-Router)	M	M	M	M
C (alle anderen KNXnet/IP-Geräte)	M	M	O	O

5.1.3 Sicherheitsbetrachtungen

5.1.3.1 Einleitung

Für KNX war und ist das Thema Sicherheit von keiner großen Bedeutung, da man für eine Verletzung der Sicherheit lokal Zugriff auf das Netzwerk haben muss. Im Fall von KNX TP (EIB) und KNX PL bedeutet das, dass man dafür sogar den physikalischen Zugriff auf die Netzkabel benötigt, was in fast allen Fällen unmöglich ist, da die Kabel innerhalb des Gebäudes oder unter der Erde verlegt sind.

Aus diesem Grund spielen Sicherheitsaspekte für KNX-Medien auf der Feldebene eine untergeordnete Rolle.

Das Ziel von KNXnet/IP ist, ein Mittel zur Verwendung der vorhandenen Datennetzwerktechnologie, d. h. Internet Protocol, für die schnelle Kommunikation zwischen verschiedenen KNX-Subnetzen am selben Ort oder an unterschiedlichen Orten bereitzustellen.

Sofern das Datennetzwerk nicht vollständig abgeschirmt ist und nur für KNXnet/IP verwendet wird, muss eine gewisse Anzahl von potentiellen Bedrohungen betrachtet werden.

5.1.3.2 Kategorien von Bedrohungen

5.1.3.2.1 Allgemeines

Dieser Abschnitt umschreibt kurz verschiedene Arten von Sicherheitsbedrohungen und zeigt typische Gegenmaßnahmen auf.

5.1.3.2.2 Passive Angriffe

Passive Angriffe schließen eine Verletzung der Privatsphäre oder die Weitergabe von Informationen an nicht autorisierte Dritte ein, die das Netzwerk „anzapfen“ oder abhören. Passive Angriffe sind schwer aufzudecken, da sie den Netzwerkverkehr in keiner Weise verändern. Die primäre Abwehr von passiven Angriffen besteht in der Abschirmung des Netzwerkverkehrs (man kann keinen Verkehr abhören oder analysieren, den man nicht sieht).

- Lauschangriff — Abfangen und Untersuchen von Paketinhalten durch unautorisierte Dritte. Zusätzlich zur Abschirmung des Netzwerks kann dieser Bedrohung durch das Verschlüsseln von Paketen entgegengewirkt werden.
- Verkehrsanalyse — Abfangen und Analysieren von Verkehrsmerkmalen, ohne die Inhalte einzelner Pakete zu untersuchen. Diese Bedrohung besteht auch dann, wenn Pakete verschlüsselt sind, und kann nur durch Abschirmung des Netzwerkverkehrs abgewehrt werden.

5.1.3.2.3 Aktive Angriffe

Aktive Angriffe schließen in irgendeiner Form eine Veränderung des Netzwerkverkehrs ein: Pakete werden hinzugefügt, gelöscht oder verzögert oder die Inhalte von Paketen werden in irgendeiner Weise verändert. Arten aktiver Angriffe sind folgend aufgeführt.

- Maskerade — nicht autorisierte Instanzen, die den Verkehr derart verfälschen, dass er von einer legitimen Quelle zu stammen scheint. Dieser Bedrohung kann mithilfe von Authentisierungsmechanismen entgegengewirkt werden.
- Verletzung der Zugriffskontrolle — ein Benutzer greift auf eine Ressource zu, für deren Verwendung er nicht autorisiert ist. Dieser Bedrohung kann mithilfe von Mechanismen für die Nachrichtenintegrität, Authentisierung und Autorisierung entgegengewirkt werden (die Richtlinien zur Zugangskontrolle müssen auch gesichert werden).
- Modifikation — Abfangen und Ändern der Inhalte von Paketen im Netzwerk. Dieser Bedrohung kann mithilfe von Mechanismen für die Nachrichtenintegrität entgegengewirkt werden.
- Löschung — Verhindern, dass Pakete am beabsichtigten Ziel ankommen (indem z. B. ein Router derart böswillig konfiguriert wird, dass er Pakete verwirft, oder indem Pakete während ihrer Übertragung in der Leitung absichtlich abgefangen werden). Diese Bedrohung kann nur sehr schwer abgewehrt werden, insbesondere dann, wenn der Angreifer Paketübertragungen im physikalischen Netzwerk stören kann.
- Verzögerung — Verzögern der Ankunft von Paketen an ihrem beabsichtigten Ziel. Dieser Angriff kann erzielt werden, indem ein Router oder eine Brücke beeinträchtigt oder überlastet wird oder indem ein zeitlich begrenzter Angriff auf die Paketübertragung durchgeführt wird (und somit Verzögerungen bei der erneuten Übertragung verursacht werden). Diese Bedrohung kann ebenfalls nur schwer abgewehrt werden.
- Wiedergabe – Abfangen von Paketen und erneutes Senden der Pakete im Netzwerk zu einem späteren Zeitpunkt. Dieser Bedrohung kann durch Authentisierung und Nachrichtenintegrität sowie Zeitstempel oder Nachrichtenzähler entgegengewirkt werden.
- DoS (Denial Of Service; Dienstablehnung) — diese Art von Angriff schließt üblicherweise das Einschleusen einer unnützen Verkehrsflut in das Netzwerk ein, um Ressourcen wie die Bandbreite oder die Verarbeitungszeit zu verbrauchen, sodass diese nicht verfügbar sind, um den legitimen Verkehr zu regeln. Diese Bedrohung ist nur schwer abzuwehren, insbesondere dann, wenn der Angriffsverkehr nur schwer von dem legitimen Verkehr unterschieden werden kann.

5.1.3.3 Gegenmaßnahmen

Im Folgenden sind Gegenmaßnahmen für KNXnet/IP aufgelistet.

- a) Üblicherweise werden Gebäudesteuerungsdaten nur innerhalb von geschlossenen oder streng überwachten Netzwerken ausgetauscht. KNXnet/IP sollte daher nur in einem Intranet oder in einem VPN (Virtual Private Network) verwendet werden.
- b) Für das Filtern von KNXnet/IP-Datagrammen aus dem Netzwerk sind Netzwerkanalysertools sowie Fachkenntnisse erforderlich. Der Inhalt einer KNXnet/IP-Nachricht ist nicht selbsterklärend, sondern erfordert semantisches Wissen, das in der ETS enthalten ist. Der Zugriff auf die ETS darf nur durch autorisierte Personen erfolgen.
- c) Beim Öffnen von Punkt-zu-Punkt-Verbindungen, z. B. für das Tunnelling oder die Fernprotokollierung, sollte ein Authentisierungsmechanismus verwendet werden.
- d) Standardmäßige KNXnet/IP-Multicast-Adressen dürfen nicht veröffentlicht werden.

- e) Bei allen Punkt-zu-Punkt-Verbindungen, z. B. für das Tunnelling oder die Fernprotokollierung, sollten Folgezähler verwendet werden.
- f) Es sollten UDP/IP- oder TCP/IP-Portnummern verwendet werden, die von einer Firewall gefiltert werden.

Tabelle 3 — Bedrohungen und Gegenmaßnahmen

Bedrohung	Gegenmaßnahme(n)
Lauschangriff	a), b), d)
Verkehrsanalyse	a), d)
Maskerade	a), c), d)
Verletzung der Zugriffskontrolle	a), c), d)
Modifikation	a), e)
Löschung	a)
Verzögerung	a)
Wiedergabe	c), e)
DoS	a)

5.1.3.4 Schlussfolgerung

Den meisten der aufgeführten Bedrohungen kann entgegengewirkt werden, indem man sich nur innerhalb eines Netzwerks aufhält, das den Netzwerkzugriff von außen beschränkt und streng überwacht.

Die Verwendung eines Authentisierungsmechanismus beim Aufbau von Punkt-zu-Punkt-Verbindungen ist hierfür ein zusätzliches Mittel.

Es ist eher unwahrscheinlich, dass legitimierte Benutzer eines Netzwerks über Mittel zum Abfangen und Entschlüsseln verfügen, um KNXnet/IP anschließend zu verfälschen, ohne die KNX-Normen intensiv studiert zu haben.

Daher wird die verbleibende Sicherheitsbedrohung als sehr gering eingeschätzt und rechtfertigt nicht die Verwendung einer Verschlüsselung, die beachtliche Computerressourcen erfordern würde.

5.2 Abschnitt 2: Core

5.2.1 Anwendungsbereich

Dieser Abschnitt „Core“ der KNXnet/IP-Norm stellt ein allgemeines, vom Host-Protokoll unabhängiges System bereit, das mehrere spezialisierte „Dienstprotokolle“ in einer modularen und erweiterbaren Form einschließt.

Diese Norm befasst sich mit:

- der Festlegung von Datenpaketen, die für die KNXnet/IP-Kommunikation über das „Host-Protokoll“-Netzwerk (nicht KNX) gesendet werden;
- der Erkennung und Selbstbeschreibung von KNXnet/IP-Servern und;
- der Konfiguration, des Aufbaus und der Wartung eines Kommunikationskanals zwischen einem KNXnet/IP-Client und einem KNXnet/IP-Server.

5.2.2 KNXnet/IP-Frames

5.2.2.1 Allgemeine Festlegungen

5.2.2.1.1 Datenformat

Die in dieser Norm beschriebenen KNXnet/IP-Frames sind binär codiert.

Datenstruktur-Spezifikationen sind immer im Binärformat angegeben.

5.2.2.1.2 Byte-Reihenfolge

Bei binären Strukturen muss die Byte-Reihenfolge dem Big-Endian-Modus (Motorola, keine Vertauschung) entsprechen, wenn nichts anderes angegeben ist. Bei Klartextformaten müssen die Byte-Reihenfolge und die Formatierung den Angaben in den zugehörigen Protokoll-Spezifikationen entsprechen.

5.2.2.1.3 Strukturen

Alle KNXnet/IP-Strukturen folgen einer gemeinsamen Regel: das erste Oktett entspricht immer der Länge der gesamten Struktur (einige Strukturen können über Felder mit variabler Länge, z. B. für Zeichenketten, verfügen) und das zweite Oktett stellt immer eine Kennung dar, die den Typ der Struktur angibt. Ab dem dritten Oktett folgen die Strukturdaten. Übersteigt die Menge der Daten 252 Oktette, muss das Längen-Oktett den Wert FFh erhalten und die nächsten beiden Oktette entsprechen der Länge eines 16-Bit-Werts. Die Strukturdaten beginnen dann im fünften Oktett.

5.2.2.2 Frame-Format

Die Kommunikation zwischen KNXnet/IP-Geräten beruht auf KNXnet/IP-Frames. Ein KNXnet/IP-Frame ist ein Datenpaket, das über das Netzwerkprotokoll (nicht KNX) gesendet wird und aus einer Kopfzeile (vergleichbar mit dem IP-Header eines IP-Datenpakets) und einem optionalen Textkörper (Body) von variabler Länge besteht.

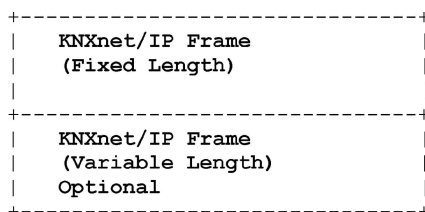


Bild 2 — KNXnet/IP-Frame im Binärformat

Der Typ des KNXnet/IP-Frames wird mithilfe einer KNXnet/IP-Dienststartkennung in der Kopfzeile beschrieben. KNXnet/IP-Dienste schließen unter anderem Informationen bezüglich der Erkennung und Beschreibung, des (Verbindungs-)Managements des Kommunikationskanals sowie der KNX-Datenübertragung ein.

5.2.2.3 Header

5.2.2.3.1 Beschreibung

Jeder KNXnet/IP-Frame umfasst ausnahmslos mindestens den allgemeinen KNXnet/IP-Header, der Informationen zur Protokollversion, dem Header und der Gesamtpaketlänge sowie die KNXnet/IP-Dienststartkennung enthält. Dem KNXnet/IP-Header kann ein KNXnet/IP-Body folgen; dies hängt von dem KNXnet/IP-Dienst ab.

Zeitstempelinformationen und Frame-Zähler sind nicht in dem allgemeinen Header des KNXnet/IP-Frames enthalten, da diese Informationen sehr stark mit bestimmten KNXnet/IP-Dienststarten verknüpft sind und daher als zusätzliche Informationen für bestimmte Kommunikations- und Kanalarten in den Body dieser Dienste eingeschlossen werden.

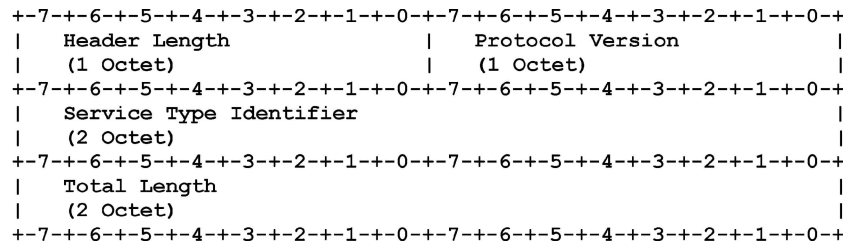


Bild 3 — KNXnet/IP-Header im Binärformat

5.2.2.3.2 Header-Länge

Obwohl die Länge des Headers stets festgelegt ist, kann sich die Größe des Headers mit einer neuen Version des Protokolls ändern. Die Header-Länge kann als Anzeiger der KNXnet/IP-Frame-Daten verwendet werden, um den Anfang des KNXnet/IP-Bodys zu finden.

5.2.2.3.3 Protokollversion

Die Informationen zur Protokollversion geben die Überarbeitung des KNXnet/IP-Protokolls an, auf der der sich anschließende KNXnet/IP-Frame basiert. Sie werden im binär codierten Dezimalformat abgespeichert. Die derzeit einzige gültige Protokollversion ist 1.0 (als Hexadezimalwert 10h dargestellt).

5.2.2.3.4 KNXnet/IP-Dienst

Die KNXnet/IP-Dienststartkennung definiert die Art der Aktion, die ausgeführt werden soll, und den Typ der Nutzdaten, die im KNXnet/IP-Body enthalten sind, wenn anwendbar. Das hohe Oktett der KNXnet/IP-Dienststartkennung kennzeichnet die Dienststartfamilie und das niedrige Oktett die eigentliche Dienststart innerhalb dieser Familie. Für eine ausführliche Beschreibung der Dienste, siehe unten.

5.2.2.3.5 Gesamtlänge

Die Gesamtlänge drückt die gesamte Länge des KNXnet/IP-Frames in Oktetten aus. Die Länge schließt den gesamten KNXnet/IP-Frame, d. h. die Header-Länge des KNXnet/IP-Headers und den gesamten KNXnet/IP-Body, ein.

5.2.3 Host-Protokoll-Unabhängigkeit

5.2.3.1 Host-Protokoll

Die KNXnet/IP-Core-Spezifikation legt die Einbindung von KNX-Protokoll-Implementierungen, die auf Internet Protocol (IP) aufsetzen, fest. Sie beschreibt allgemeine und vom Host-Protokoll unabhängige sowie die bezüglich des Host-Protokolls spezifischen Teile der KNXnet/IP-Kommunikation.

5.2.3.2 Endpunkte

KNXnet/IP legt die HPAI-Struktur (Host Protocol Address Information) fest, die die Kombination aus der IP-Adresse und der Portnummer ist. HPAI sind Daten, die erforderlich sind, um einen KNXnet/IP-Frame an ein anderes Gerät zu senden. Die KNXnet/IP-Norm verwendet die Bezeichnung „KNXnet/IP-Endpunkt“ als logische Sicht einer HPAI, um ein anderes KNXnet/IP-Gerät für bestimmte eindeutige Zwecke zu adressieren.

Jedes KNXnet/IP-Gerät muss genau einen zum Gerät gehörenden, bidirektionalen und verbindungslosen Endpunkt für die Erkennung unterstützen, wenn für das Host-Protokoll Erkennungsdienste erforderlich sind. Es muss mindestens einen bidirektionalen und verbindungslosen Endpunkt für die Kontrolle und mindestens einen bidirektionalen und verbindungsorientierten Endpunkt für die Übertragung von zur Dienststart gehörenden Daten unterstützen.

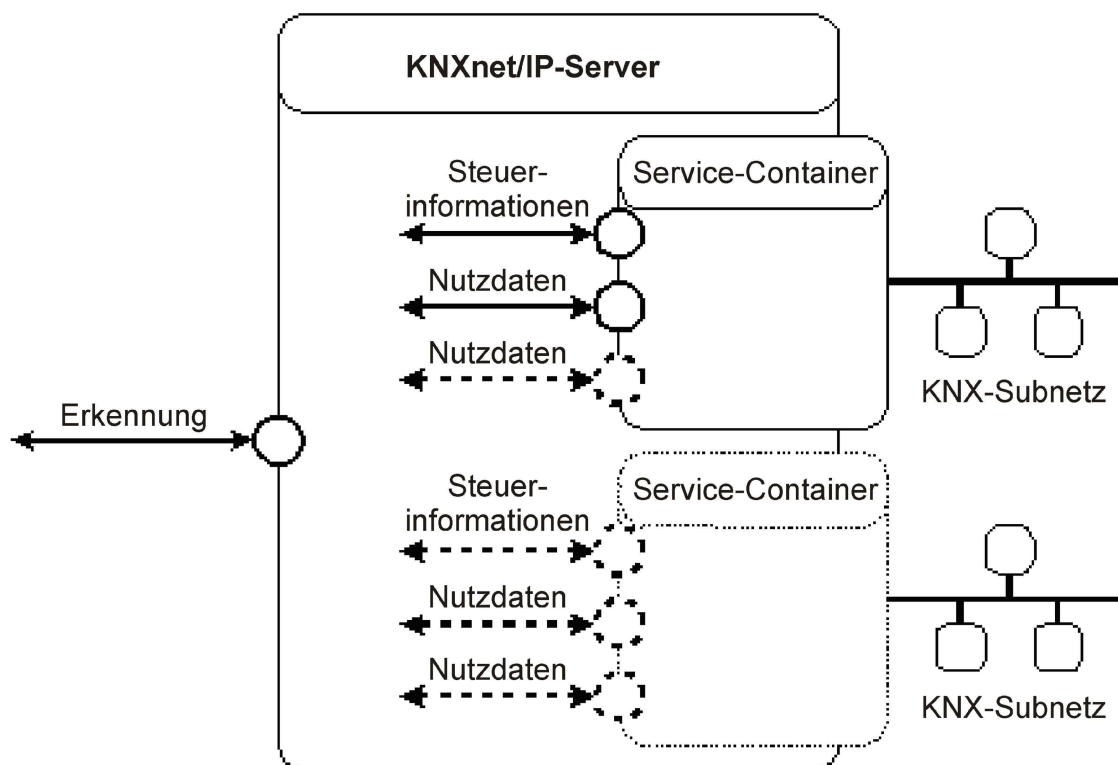


Bild 4 — Beispielkonfiguration für KNXnet/IP-Server-Endpunkte

Der Kontrollendpunkt adressiert eindeutig eine Instanz innerhalb des KNXnet/IP-Servergerätes, die mindestens eine KNXnet/IP-Dienst bereitstellen muss.

Diese Instanz, der sogenannte Dienst-Container, kann mit einem KNX-Subnetz verbunden sein. Wenn das KNXnet/IP-Servergerät mehr als eine KNX-Subnetzverbindung unterstützt, ist es erforderlich, dass jedes KNX-Subnetz durch einen anderen Kontrollendpunkt abgebildet wird. Der KNXnet/IP-Client betrachtet somit jeden Dienst-Container, der durch einen Kontrollendpunkt abgebildet wird, als unabhängige Instanz, unabhängig davon, ob die Dienst-Container in nur einem oder in zwei getrennten KNXnet/IP-Servergeräten implementiert sind.

Diese KNXnet/IP-Endpunkte stellen eine logische Sicht auf die Kommunikation eines KNXnet/IP-Gerätes dar. Für die eigentliche Implementierung dieser Endpunkte mit verschiedenen Host-Protokollen können transportmedienunabhängige Ansätze, die sich von der logischen Sicht unterscheiden, verwendet werden. Die bidirektionalen KNXnet/IP-Endpunkte könnten z. B. mithilfe von zwei unidirektionalen Kanälen mit dem Host-Protokoll implementiert werden. Es ist daher möglich, dass ein KNXnet/IP-Endpunkt durch mehrere HPAs repräsentiert wird.

5.2.4 Erkennung und Selbstbeschreibung

5.2.4.1 Einleitung

Insbesondere bei Netzwerken, die Hot-Plug unterstützen und bei denen selbst die Adresszuweisung während der Laufzeit stattfinden kann (z. B. IP-Adresszuweisung über BootP oder DHCP), ist es von großer Bedeutung, nach Geräten innerhalb eines Subnetzes suchen zu können, ohne Netzwerkparameter auf unkonventionellem Weg abrufen und manuell in das Client-Tool eingeben zu müssen, um eine Verbindung aufzubauen. Des Weiteren ist ein Selbstbeschreibungsmechanismus eine wichtige Funktion, um ein genaues Bild der vom KNXnet/IP-Server unterstützten Dienste zu erhalten, ohne das Prinzip „Versuch und Irrtum“ anwenden zu müssen.

5.2.4.2 Erkennung

Jeder KNXnet/IP-Server muss einen Erkennungsmechanismus nach der folgenden Vorgehensweise implementieren. Wenn es für das Host-Protokoll anwendbar ist, wird empfohlen, einen KNXnet/IP-Client mit Unterstützung der Suche nach KNXnet/IP-Servern anstelle der manuellen Eingabe zu implementieren.

Die Erkennungsoperation besteht aus einem SEARCH_REQUEST-Datenpaket, das über Multicast mithilfe des Erkennungsendpunktes gesendet wird und das die HPAI des Erkennungsendpunktes des KNXnet/IP-Clients enthält. Die HPAI können eine Unicast-IP-Adresse enthalten, um die Antworten von den verschiedenen KNXnet/IP-Servern direkt in Punkt-zu-Punkt-Form zu empfangen. Normalerweise sollten sie die KNXnet/IP-Multicast-Adresse der Systemkonfiguration enthalten, um den Empfang von KNXnet/IP-Servern sicherzustellen, die sich in einem anderen Subnetz befinden.

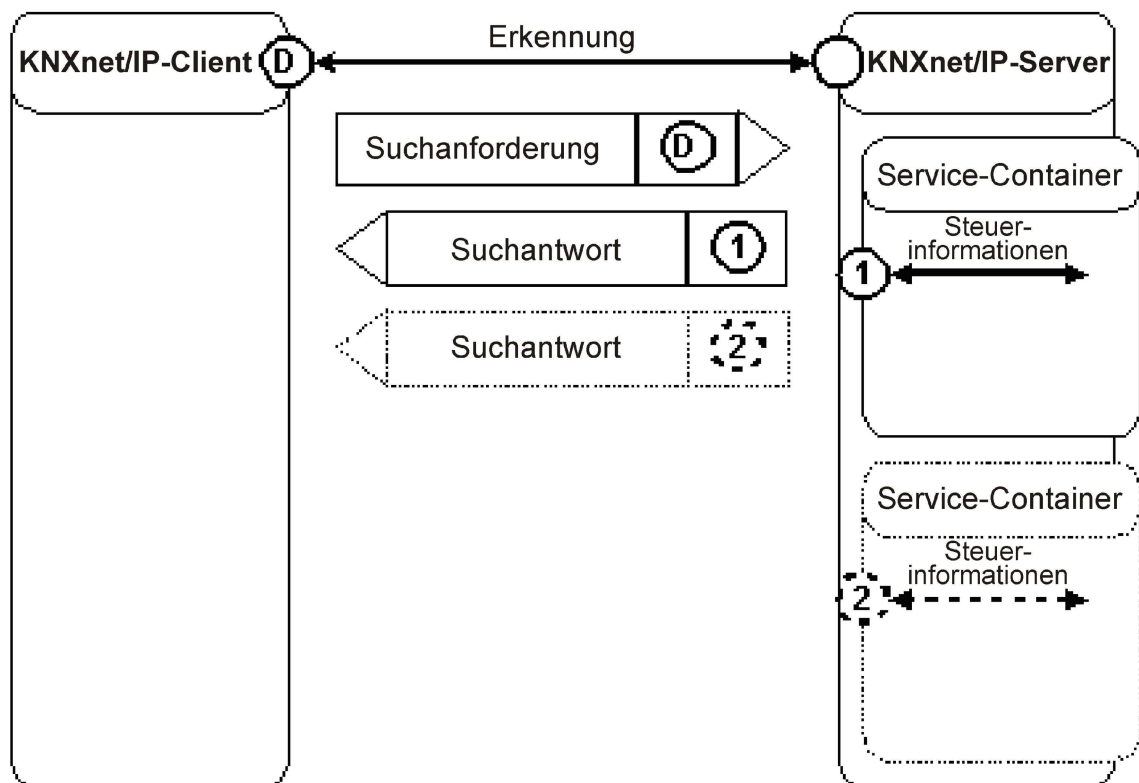


Bild 5 — Erkennungsprozedur

Nach dem Senden der Anforderung muss der KNXnet/IP-Client während der SEARCH_TIMEOUT-Zeit auf SEARCH_RESPONSE-Frames von KNXnet/IP-Servern warten. Nach dieser Zeitdauer muss jeder empfangene SEARCH_RESPONSE-Frame von diesem Client ignoriert werden, bis dieser einen neuen Erkennungszyklus startet. SEARCH_RESPONSE-Frames, die Clients von anderen Clients empfangen, müssen ignoriert werden.

Jeder KNXnet/IP-Server, der einen SEARCH_REQUEST-Dienst empfängt, muss der angegebenen HPAI mithilfe seines Erkennungsendpunktes unverzüglich mit einem SEARCH_RESPONSE-Datenpaket antworten. Eine derartige Antwort enthält nur die HPAI des Kontrollendpunktes des KNXnet/IP-Servers für alle weitere Kommunikation.

Jeder KNXnet/IP-Server muss die Erkennung unterstützen, indem er Suchanforderungen verarbeitet und korrekte Antworten sendet. Ein KNXnet/IP-Server kann Verbindungen mit mehreren KNX-Subnetzen unterstützen; er muss jedoch ein SEARCH_RESPONSE-Datenpaket an den Kontrollendpunkt jedes KNX-Subnetzes senden, mit dem er Verbindungen aufbauen kann, auch dann, wenn er jeweils nur eine Datenverbindung unterstützt.

5.2.4.3 Selbstbeschreibung

Nach dem Erkennen eines KNXnet/IP-Servers sendet der KNXnet/IP-Client normalerweise einen DESCRIPTION_REQUEST über eine Unicast- oder Punkt-zu-Punkt-Verbindung an alle Kontrollendpunkte des KNXnet/IP-Servers. Dazu ist es erforderlich, dass alle KNXnet/IP-Implementierungen Beschreibungsanforderungen unterstützen. Des Weiteren sollte ein KNXnet/IP-Client vor dem Kommunizieren mit einem KNXnet/IP-Server mithilfe des Selbstbeschreibungsmechanismus prüfen, ob der Server die vom Client angeforderten Dienste unterstützt.

Wenn ein KNXnet/IP-Server eine gültige Beschreibungsanforderung empfängt, muss er mit einem DESCRIPTION_RESPONSE-Paket antworten, das Informationen zu dem Bereich unterstützter Protokollversionen und den eigenen Funktionen sowie Statusinformationen und optional einen Anzeigenamen für diese KNX-Verbindung des KNXnet/IP-Servers bereitstellt. Da ein KNXnet/IP-Server Verbindungen mit mehreren KNX-Subnetzen unterstützen kann, muss er das Antworten auf Erkennungsanforderungen für jede potentielle KNX-Subnetzverbindung unterstützen, die durch die Erkennungsantworten gemeldet wird.

5.2.5 Kommunikationskanäle

5.2.5.1 Einleitung

Ein Kommunikationskanal ist die Datenendpunktverbindung zwischen einem KNXnet/IP-Client und einem KNXnet/IP-Server. Datenendpunktverbindungen werden für Dienste aufgebaut, die eine Punkt-zu-Punkt-Kommunikation erfordern, z. B. Tunnelling oder Device Management. Jede Punkt-zu-Punkt-Verbindung zwischen einem KNXnet/IP-Client und einem KNXnet/IP-Server muss vom Client initiiert werden. Jeder KNXnet/IP-Server muss mindestens jeweils eine Client-Verbindung unterstützen. Er kann auch mehrere Client-Verbindungen gleichzeitig unterstützen, muss jedoch sicherstellen, dass vorhandene Verbindungen durch das Annehmen neuer Verbindungen nicht beeinflusst werden (ein KNXnet/IP-Server darf z. B. keine Tunnelling-Verbindung mit demselben physikalischen Zugriff auf ein KNX-Subnetz in verschiedenen Modi, Sicherungsschicht oder Busmonitor-Schicht, annehmen).

5.2.5.2 Verbindung herstellen

Um eine Verbindung zwischen einem KNXnet/IP-Client und einem KNXnet/IP-Server herzustellen, muss der Client einen CONNECT_REQUEST-Frame an den Kontrollendpunkt des Servers senden. Diese Anforderung stellt Informationen zu der angeforderten Verbindungsart (z. B. Daten-Tunnelling oder Fernprotokollierung), allgemeine und in Bezug auf die Verbindungsart spezifische Optionen (z. B. Sicherungsschicht oder Busmonitor-Modus²⁾) sowie die HPAI des Datenendpunktes, die der Client für diesen Kommunikationskanal verwenden möchte, bereit.

Vor dem Senden einer Verbindungsanforderung einer spezifischen Art (mit spezifischen Optionen) sollte der KNXnet/IP-Client mithilfe der vom KNXnet/IP-Server empfangenen Selbstbeschreibungsinformationen prüfen, ob der Server die angeforderte Verbindungsart und/oder alle angeforderten Optionen unterstützt.

Der KNXnet/IP-Server muss dann in jedem Fall einen CONNECT_RESPONSE-Frame an den KNXnet/IP-Client zurücksenden, der den Aufbau der Verbindung anfordert, in dem der Status der Anforderung bereitgestellt wird (ACK/NACK mit erweiterten Statusinformationen, wenn anwendbar). Wenn die Anforderung vom Server angenommen wurde, enthält der CONNECT_RESPONSE-Frame zusätzlich eine Kennung sowie die HPAI des Datenendpunktes, die der Server nun für diesen Kommunikationskanal³⁾ vorbereitet hat.

Nach dem Senden der Verbindungsanforderung muss der KNXnet/IP-Client während der vom Host-Protokoll abhängigen CONNECT_REQUEST_TIMEOUT-Zeit (= 10 s) auf den Antwort-Frame vom KNXnet/IP-Server warten. Nach dieser Zeitdauer muss jeder empfangene Antwort-Frame von diesem Client ignoriert werden, bis dieser eine neue Verbindungsanforderung startet.

2) Die Liste der unterstützten Schichten und Dienste sollte in zukünftigen Überarbeitungen erweitert werden.

3) Es sollte beachtet werden, dass eine EIBnet/IP-Verbindung aus technischen Gründen aus mehreren logischen Verbindungen in der Host-Protokoll-Schicht bestehen kann.

Die aktuelle Protokoll-Norm setzt voraus, dass eine Verbindung nicht von mehreren Clients gemeinsam genutzt werden darf. Ein KNXnet/IP-Server darf daher nicht mehrere Verbindungsanforderungen derselben Art, z. B. auf derselben physikalischen KNX-Verbindung, annehmen, obwohl er durchaus mehrere physikalische Verbindungen implementieren kann, die logisch jeweils als unabhängiger KNXnet/IP-Server dargestellt werden, wenn dies von dem verwendeten Host-Protokoll unterstützt wird. Die Client-Implementierung kann auf dieser Beschränkung aufbauen und muss ein derartiges Szenario einer gemeinsam genutzten Verbindung nicht unterstützen. Durch bestimmte Dienstfamilien bedingte Ausnahmen zu dieser allgemeinen Regel sind in dem KNXnet/IP-Kapitel der entsprechenden Dienstfamilie beschrieben.

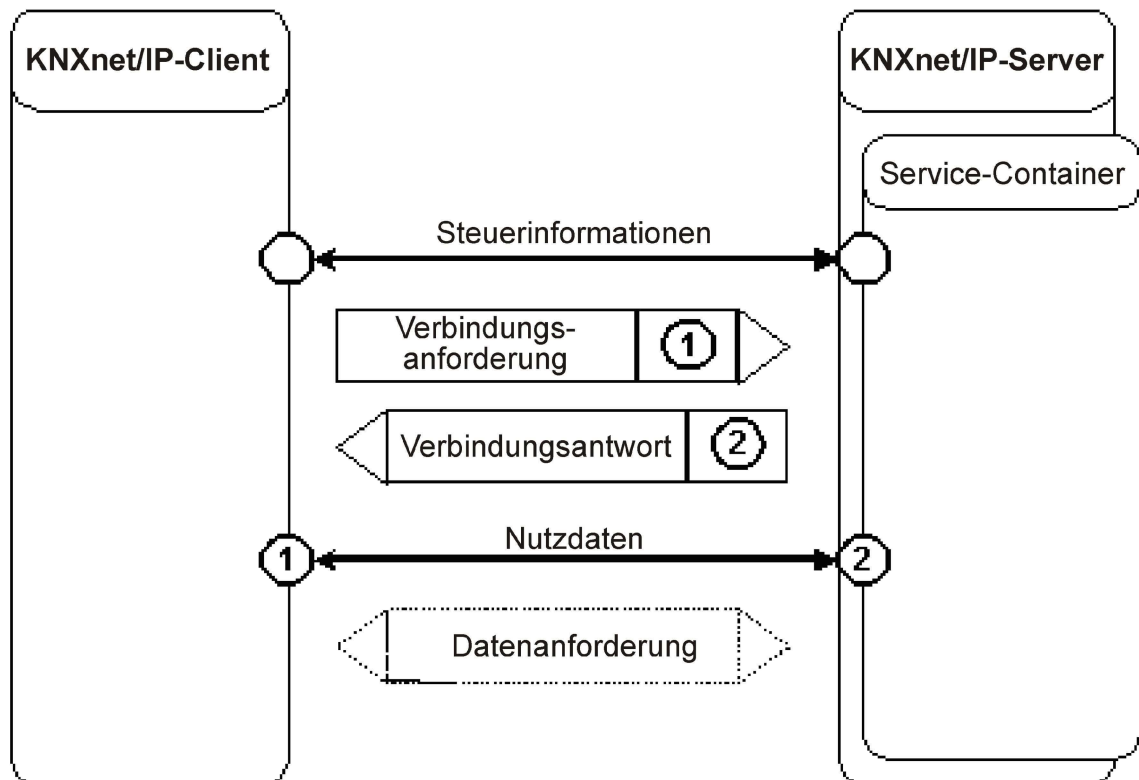


Bild 6 — Aufbau einer Datenverbindung

5.2.5.3 Connection-Header

5.2.5.3.1 Beschreibung

Der Body eines jeden KNXnet/IP-Frames, der über einen aufgebauten Kommunikationskanal gesendet wird, beginnt mit Datenfeldern, die zusätzliche allgemeine Informationen zur Datenverbindung enthalten. Die Menge dieser Daten und die Art der eingeschlossenen Informationen wird durch verschiedene Optionen während der Verbindungsphase eines Kommunikationskanals bestimmt. Die Gesamtheit dieser Datenfelder wird als „Connection-Header“ bezeichnet und sein Aufbau unterscheidet sich stark in Abhängigkeit von den bereits erwähnten Verbindungsoptionen. Nur die Reihenfolge, in der die verschiedenen Datenfelder im Connection-Header gespeichert werden, ist festgelegt.

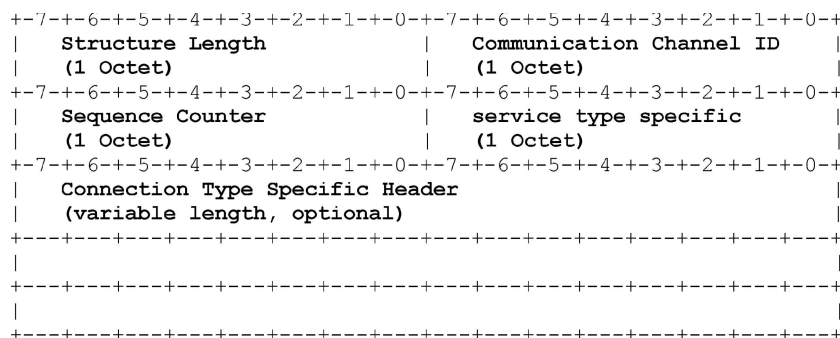


Bild 7 — Üblicher Connection-Header

5.2.5.3.2 Strukturlänge

Die Strukturlänge bezeichnet die Gesamtlänge des Connection-Headers.

5.2.5.3.3 Kommunikationskanal-ID

Der KNXnet/IP-Server weist jedem aufgebauten Kommunikationskanal eine Kommunikationskanal-ID zu.

5.2.5.3.4 Folgezähler

Der Folgezähler wird für jeden Kommunikationskanal verwaltet. Er muss sich für jeden Kommunikationskanal einzeln bei jedem KNXnet/IP-Frame, der über die Datenverbindung gesendet wird, um eins erhöhen. Beide KNXnet/IP-Geräte, die den Kommunikationskanal aufrechterhalten, verfügen über unabhängige Folgezähler.

Jedes Mal, wenn eine Verbindung aufgebaut wird, muss der Zähler für diese Verbindung auf 0 gesetzt werden, sodass der erste KNXnet/IP-Frame, der über einen aufgebauten Kommunikationskanal gesendet wird, einen Folgezähler von 0 aufweist.

5.2.5.3.5 Verbindungsartspezifische Header-Elemente

Die verbindungsartspezifischen Header-Elemente sind optional und in ihrer Länge in Abhängigkeit von der Art der Verbindung variabel.

5.2.5.4 Heartbeat-Überwachung

Host-Protokolle, die keine Mechanismen zur Prüfung der Lebensdauer wie UDP/IP bereitstellen, benötigen eine Vorgehensweise, um Kommunikationsausfälle im KNX- oder Tunnelling-Netzwerk zu ermitteln. Um derartige Situationen aufzudecken, wurde die Heartbeat-Überwachung entwickelt und muss sowohl von KNXnet/IP-Clients als auch KNXnet/IP-Servern implementiert werden.

Der Client muss regelmäßig, d. h. alle 60 s, einen CONNECTIONSTATE_REQUEST-Frame an den Kontrollendpunkt des Servers senden, der unverzüglich mit einem CONNECTIONSTATE_RESPONSE-Frame antworten muss (dies zählt als Heartbeat-Antwort).

Wenn der KNXnet/IP-Client die Heartbeat-Antwort nicht innerhalb der CONNECTIONSTATE_REQUEST_TIMEOUT (= 10 s) empfängt oder der Status der empfangenen Heartbeat-Antwort irgendeinen Fehlerzustand meldet, muss der Client den CONNECTIONSTATE_REQUEST dreimal wiederholen und dann die Verbindung beenden, indem er einen DISCONNECT_REQUEST an den Kontrollendpunkt des Servers sendet.

Wenn der KNXnet/IP-Server eine Heartbeat-Anforderung nicht innerhalb von 120 s des letzten korrekt empfangenen Nachrichten-Frames empfängt, muss der Server die Verbindung beenden, indem er einen DISCONNECT_REQUEST an den Kontrollendpunkt des Clients sendet. Der Server darf die Zeitüberschreitung nicht erneut auslösen, nachdem Nachrichten mit einer unerwarteten Folgenummer empfangen wurden.

5.2.5.5 Verbindung trennen

Normalerweise beendet der Client die Verbindung. Während des normalen Betriebs muss der Client einen DISCONNECT_REQUEST an den Kontrollendpunkt des Servers senden, um die Beendigung der Datenkanalverbindung anzufordern.

Der Client sollte, wenn möglich, versuchen, die Verbindung nicht abrupt zu beenden, auch bei Fehlerbedingungen. Der Server kann die Verbindung mit dem Client im Fall von internen Problemen oder wenn er ungültige Datenpakete empfängt trennen, indem er einen DISCONNECT_REQUEST sendet; es wird jedoch empfohlen, den Client die Verbindung beenden zu lassen.

Das KNXnet/IP-Gerät, das den DISCONNECT_REQUEST vom Kommunikationspartner empfängt, muss die Operation mit einem DISCONNECT_RESPONSE-Frame bestätigen. Dieses Datenpaket meldet die endgültige Beendigung des zuvor aufgebauten Kommunikationskanals.

5.2.6 Allgemeine Implementierungsrichtlinien

5.2.6.1 Einleitung

Dieser Abschnitt legt die Programmierungsrichtlinien fest, die bei der Implementierung von KNXnet/IP-Servern bzw. KNXnet/IP-Clients beachtet werden müssen. Die Übereinstimmung mit diesen Richtlinien ist eine Anforderung für die Zertifizierung der Protokoll-Implementierung.

5.2.6.2 KNXnet/IP-Server

- Wenn ein Server ein Datenpaket mit einem nicht unterstützten Versionsfeld empfängt, muss er mit einer NACK-Nachricht antworten, die im Statusfeld „E_VERSION_NOT_SUPPORTED“ angibt.
- Wenn ein ungültiges Datenpaket empfangen wird, muss die Implementierung das Datenpaket ignorieren, ohne eine weitere Aktion auszuführen.
- Wenn eine Verbindung aufgebaut wird, müssen alle Datenpakete mit derselben Protokollversion gesendet werden.
- Wenn eine Verbindung aufgebaut wird und sich die Protokollversion innerhalb der empfangenen Datenpakete ändert, muss der Server die Verbindung beenden.

5.2.6.3 KNXnet/IP-Clients

- Wenn ein Client ein Datenpaket mit einem nicht unterstützten Versionsfeld empfängt, muss er mit einer NACK-Nachricht antworten, die im Statusfeld „E_VERSION_NOT_SUPPORTED“ angibt. Wenn eine Verbindung mit diesem Server, der Datenpakete mit einer nicht unterstützten Protokollversion sendet, aufgebaut wird, muss der Client die Verbindung trennen. Der Client kann anschließend versuchen, erneut eine Verbindung mit dem Server aufzubauen.
- Wenn ein ungültiges Datenpaket empfangen wird, muss die Implementierung das Datenpaket ignorieren, ohne eine weitere Aktion auszuführen.
- Wenn eine Verbindung aufgebaut wird, müssen alle Datenpakete mit derselben Protokollversion gesendet werden.
- Wenn eine Verbindung aufgebaut wird und sich die Protokollversion innerhalb der empfangenen Datenpakete ändert, muss der Client die Verbindung zum Server trennen. Der Client kann anschließend versuchen, erneut eine Verbindung mit dem Server aufzubauen.

5.2.6.4 Einstellungen für KNXnet/IP-Router

5.2.6.4.1 Werksseitige Standardeinstellungen für KNXnet/IP-Router

Eine wichtige Funktion von KNXnet/IP-Routern ist die Bereitstellung eines ordnungsgemäßen KNXnet/IP-Routings ohne Benutzereingriff. Dieses sofort betriebsbereite Routing-Verhalten („Plug and Play“) setzt eine standardisierte Werkskonfiguration voraus.

- Router müssen mit der individuellen Standardadresse FF00h ausgeliefert werden.
- Die Routing-Multicast-Adresse entspricht der Multicast-Adresse der Systemkonfiguration.
- KNXnet/IP-Broadcast-Telegramme müssen von einem KNX-Subnetz zu einem anderen geleitet werden, auch wenn die KNXnet/IP-Routing-Geräte nach wie vor mit ihrer Standard-Werkskonfiguration verwendet werden.
- Das KNXnet/IP-Routing muss bereits funktionieren, auch wenn noch keine gültige Unicast-IP-Adresse vom KNXnet/IP-Routing-Gerät bezogen oder diesem zugewiesen wurde.
- Wenn einem KNXnet/IP-Router eine individuelle Adresse über ein KNX-Subnetz zugewiesen wurde, muss das Routing bereits funktionieren und das Tunnelling muss für die (weitere) Konfiguration des KNXnet/IP-Routers verfügbar sein.

5.2.6.4.2 Zuweisung einer IP-Adresse zu einem KNXnet/IP-Router

KNXnet/IP-Router müssen ein sofort betriebsbereites KNXnet/IP-Routing („Plug and Play“) unterstützen, auch wenn noch keine gültige Unicast-IP-Adresse von einem DHCP-Server, durch manuelle Eingabe oder durch Konfiguration über ETS bezogen wurde. Dies kann erfordern, dass der IP-Protokollstapel Multicast-IP-Nachrichten senden und empfangen können muss, obwohl noch keine gültige Unicast-IP-Adresse bezogen wurde.

Wenn ein IP-Protokollstapel Multicasting nicht ohne eine zugewiesene Unicast-IP-Adresse unterstützt, muss der KNXnet/IP-Router eine Unicast-IP-Adresse über AutoIP beziehen oder sich die standardmäßige Unicast-IP-Adresse für KNXnet/IP-Quellen 0.0.0.0 selbst zuweisen.

Wenn keine gültige Unicast-IP-Adresse bezogen wurde, muss der KNXnet/IP-Router für das Routing die standardmäßige Unicast-IP-Adresse für KNXnet/IP-Quellen 0.0.0.0 verwenden, darf jedoch kein KNXnet/IP-Tunnelling unterstützen.

5.2.6.5 Ersteinrichtungsmaßnahmen für KNXnet/IP-Server

5.2.6.5.1 Allgemeines

KNXnet/IP-Geräte müssen in derselben Art und Weise wie herkömmliche KNX-Geräte konfigurierbar sein. In unkonfiguriertem Zustand müssen KNXnet/IP-Router mit Default-Werten arbeiten, damit KNX-Telegramme von einem KNX-Subnetz zu einem anderen gelangen können; dadurch können KNX-Router (Linien- und Bereichskoppler) nahtlos durch KNXnet/IP-Router ersetzt werden.

Diese Anforderung setzt voraus, dass individuelle KNX-Adressen über das gesamte IP-Netzwerk hinweg eindeutig sind. Wenn KNXnet/IP-Geräte zweier unabhängiger Installationen mit demselben IP-Netzwerk verbunden werden oder wenn ein KNX-Projekt aus mehreren Installationen besteht, kann die Verwendung einer werksseitigen Default-Routing-Konfiguration nicht gewährleisten, dass KNX-Telegramme an ihr beabsichtigtes Ziel übertragen werden, da dieselbe KNX-Adresse (individuelle oder Gruppenadresse) in den verschiedenen Installationen vorhanden sein könnte.

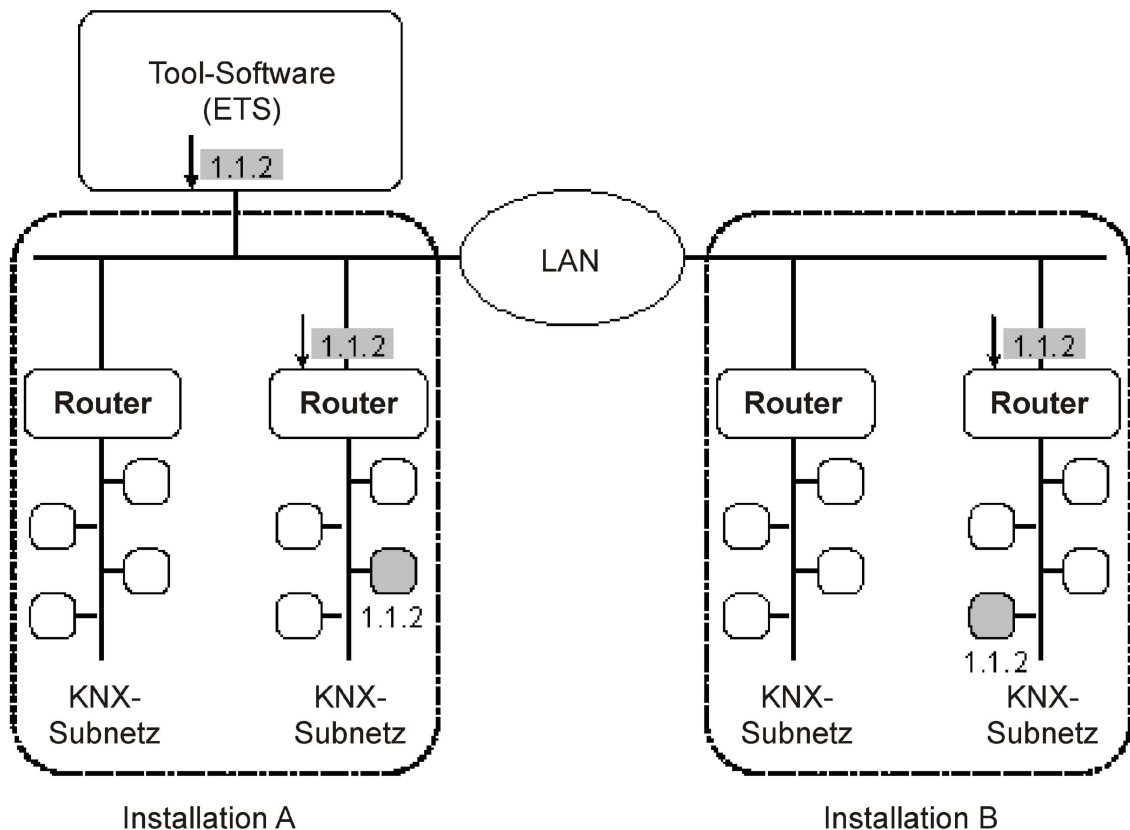


Bild 8 — KNX-Projekt mit mehreren Installationen

Unter der Voraussetzung, dass nur eine Installation mit dem IP-Netzwerk verbunden ist, gelten folgende Regeln:

- 1) Nach dem Einschalten durchläuft das KNXnet/IP-Gerät die im Kapitel „KNXnet/IP Core“ unter 5.2.8.5 „IP-Adresszuweisung“ beschriebene Prozedur, um eine individuelle IP-Adresse zu beziehen.
- 2) Wenn die Tool-Software mit einem seriellen Schnittstellengerät (RS232) oder einer KNX-USB-Schnittstelle verbunden ist, kann sie solange nur auf KNX-Geräte im lokalen KNX-Subnetz (einschließlich des lokalen KNXnet/IP-Routers) zugreifen, bis alle zu einem Projekt gehörenden KNXnet/IP-Router-Geräte erfolgreich konfiguriert wurden.
- 3) Die Tool-Software sollte stets versuchen, zuerst die KNXnet/IP-Router-Geräte zu konfigurieren, um Zugriff auf KNX-Geräte hinter möglicherweise noch unkonfigurierten KNXnet/IP-Routern zu erhalten.
- 4) Wenn die Tool-Software direkt mit dem IP-Netzwerk verbunden ist, kann sie auf alle KNXnet/IP-Geräte und über die konfigurierten Router auf die entsprechenden KNX-Subnetze zugreifen.
- 5) Die Tool-Software sollte das IP-Netzwerk für die Einrichtung von Projekten verwenden, die KNXnet/IP-Geräte enthalten.
- 6) KNXnet/IP-Geräte können in einer einzigen Konfigurationsprozedur (Zuweisung individueller KNX-Adressen und Parameter-Download) vollständig konfiguriert werden.
- 7) Zuweisung von individuellen KNX-Adressen: Wenn die Tool-Software über ein seriell KNX-Schnittstellengerät (RS232) oder über eine KNX-USB-Schnittstelle mit dem KNX-Netzwerk verbunden ist, findet die herkömmliche KNX-Management-Prozedur für KNXnet/IP-Geräte sowie für andere KNX-Knoten Anwendung. Geräte hinter unkonfigurierten Routern sind nicht erreichbar. Wenn die Tool-Software mit dem IP-Netzwerk verbunden ist, findet die nachfolgend beschriebene Konfigurationsprozedur für KNXnet/IP-Geräte Anwendung. Vor dem Einrichten der KNX-Medien auf der Feldebene muss die Tool-Software zuerst alle KNXnet/IP-Router-Geräte konfigurieren.

- 8) Parameter-Download: Wenn die Tool-Software über eine serielle Schnittstelle mit dem KNX-Netzwerk verbunden ist, findet die herkömmliche KNX-Management-Prozedur für KNXnet/IP-Geräte sowie für andere KNX-Knoten Anwendung. Wenn die Tool-Software mit dem IP-Netzwerk verbunden ist, wird zu jedem KNXnet/IP-Gerät eine Management-Verbindung aufgebaut, um die Geräteparameter herunterzuladen (siehe Konfigurationsprozedur). Nachdem alle KNXnet/IP-Router-Geräte konfiguriert wurden, kann die Tool-Software für die weitere Einrichtung und den Download über eine KNXnet/IP-Tunnelling-Verbindung auf alle anderen KNX-Knoten des Projekts zugreifen.
- 9) Wenn aus den SEARCH_RESPONSE-Antworten ersichtlich wird, dass die KNXnet/IP-Geräte unterschiedliche IP-Adressbereiche verwenden, muss die Tool-Software eine Fehlermeldung anzeigen.
- 10) Wenn aus den SEARCH_RESPONSE-Antworten ersichtlich wird, dass sich die IP-Adresse eines KNXnet/IP-Gerätes von den Netzwerkeinstellungen der Tool-Software unterscheidet und die IP-Adresse des KNXnet/IP-Gerätes eine AutoIP-Adresse ist, muss die Tool-Software eine Fehlermeldung sowie eine Bedienschaltfläche anzeigen, mit der der Benutzer das KNXnet/IP-Gerät über KNXnet/IP-Routing mit dem verbindungslosen KNX-Neustartdienst zurücksetzen kann.

5.2.6.5.2 Konfigurationsprozedur für die Konfiguration über KNXnet/IP-Routing

Die Tool-Software ist über ein serielltes KNX-Schnittstellengerät (RS232) oder eine KNX-USB-Schnittstelle mit einem KNX-Subnetz (1.1) verbunden. Ein zweites KNX-Subnetz (1.2) ist mit dem ersten KNX-Subnetz über zwei KNXnet/IP-Router und ein LAN verbunden. Alle Geräte einschließlich der Router sind zunächst nicht konfiguriert.

Die Tool-Software verwendet KNX-Management-Prozeduren und (indirekt) KNXnet/IP-Routing, um die beiden KNXnet/IP-Router zu konfigurieren.

Im Folgenden wird die Konfigurationsprozedur für die Zuweisung der individuellen KNX-Adresse und die Konfiguration der Parameter eines KNXnet/IP-Gerätes beschrieben:

- 1) Die Tool-Software fordert den Benutzer auf, den Programmiermodus des KNXnet/IP-Gerätes zu aktivieren.

ANMERKUNG Dies kann durch Drücken der Programmiertaste auf dem KNXnet/IP-Gerät erfolgen.

- 2) Die Tool-Software sendet eine KNX-Leseanforderung (Broadcast). Der KNXnet/IP-Router in diesem KNX-Subnetz sendet diesen Broadcast an weitere KNXnet/IP-Router, die diesen wiederum an ihre KNX-Subnetze weiterleiten und das Telegramm selbst verarbeiten, sofern der Programmiermodus aktiviert ist.
- 3) Die Tool-Software muss prüfen, ob mehr als ein Gerät mit „Programmiermodus aktiv“ antwortet, und die Prozedur in einem solchen Fall mit einer beschreibenden Meldung über die Ursache abbrechen. Der KNXnet/IP-Router in diesem KNX-Subnetz leitet die Antwort(en) an das KNX-Subnetz weiter.
- 4) Die Tool-Software muss eine KNX-Schreibanforderung (Broadcast) senden, um die individuelle KNX-Adresse in das KNXnet/IP-Gerät zu schreiben. Der KNXnet/IP-Router in diesem KNX-Subnetz sendet diesen Broadcast an weitere KNXnet/IP-Router, die diesen wiederum an ihre KNX-Subnetze weiterleiten und das Telegramm selbst verarbeiten, sofern der Programmiermodus aktiviert ist.
- 5) Die Tool-Software muss eine KNX-Verbindung mit dem KNXnet/IP-Gerät aufbauen, um die Konfigurationsparameter (Eigenschaften) herunterzuladen. Dafür ist es erforderlich, dass der KNXnet/IP-Router, der in demselben KNX-Subnetz wie die Tool-Software vorhanden ist, zuerst konfiguriert wird, bevor weitere KNXnet/IP-Geräte konfiguriert werden können.
- 6) Nach dem Herunterladen der Parameter muss die Tool-Software das KNXnet/IP-Gerät zurücksetzen, damit die Parameter wirksam werden.

5.2.6.5.3 Konfigurationsprozedur für die Konfiguration über KNXnet/IP Device Management

Die Tool-Software ist direkt mit einem LAN verbunden. Zwei KNX-Subnetze, (1.1) und (1.2), sind mit dem LAN über zwei KNXnet/IP-Router verbunden. Alle Geräte einschließlich der Router sind zunächst nicht konfiguriert.

Die Tool-Software muss KNXnet/IP Device Management verwenden, um die beiden KNXnet/IP-Router zu konfigurieren.

Im Folgenden wird die Konfigurationsprozedur für die Zuweisung der individuellen KNX-Adresse und die Konfiguration der Parameter eines KNXnet/IP-Gerätes beschrieben:

- 1) Die Tool-Software muss den Benutzer auffordern, den Programmiermodus des KNXnet/IP-Gerätes zu aktivieren.

ANMERKUNG Dies kann durch Drücken der Programmier Taste auf dem KNXnet/IP-Gerät erfolgen.

- 2) Die Tool-Software muss einen KNXnet/IP-SEARCH_REQUEST-Frame senden.
- 3) Die Tool-Software muss prüfen, ob mehr als ein KNXnet/IP-Gerät (Dienst-Container) mit dem Gerätestatus „Programmiermodus aktiv“ antwortet und die Prozedur in einem solchen Fall abbrechen.
- 4) Die Tool-Software muss die IP-Adresse von dem SEARCH_RESPONSE-Frame des KNXnet/IP-Gerätes verwenden, um eine Device-Management-Verbindung mit dem Gerät aufzubauen.
- 5) Die Tool-Software muss die individuelle KNX-Adresse, die Projektkennung sowie Subnetzinformationen festlegen, wenn anwendbar.
- 6) Zu diesem Zeitpunkt kann die Tool-Software zusätzliche Parameter herunterladen, wenn erforderlich.
- 7) Nach dem erfolgreichen Trennen der Device-Management-Verbindung oder dem Senden eines reset_req-Dienstes an das Gerät, müssen die geänderten Werte in das Gerät geschrieben und das Gerät neu gestartet werden.

5.2.7 Datenpaketstrukturen

5.2.7.1 Einleitung

Alle KNXnet/IP-Frames müssen einen gemeinsamen Header aufweisen, der aus Informationen zur Header-Länge, der Protokollversion, der KNXnet/IP-Dienststartkennung und der Gesamtlänge des KNXnet/IP-Frames besteht.

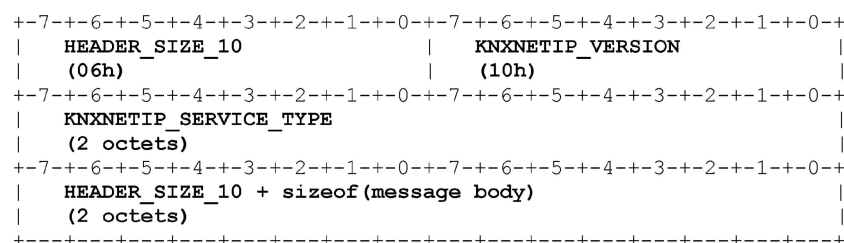


Bild 9 — KNXnet/IP-Nachrichten-Header

Anforderungen, die an verbindungslose KNXnet/IP-Endpunkte gesendet werden, müssen Informationen zur Rückkehradresse enthalten. Diese HPAI für den Empfang der Antwortinformationen stellen stets die ersten Daten des KNXnet/IP-Bodys derartiger Anforderungen dar. Zusätzliche zum KNXnet/IP-Dienst gehörende Daten können sich anschließen. Antwortpakete enthalten diese Art von Informationen zur Rückkehradresse nicht.

5.2.7.2 Übliche Konstanten

5.2.7.2.1 HEADER_SIZE_10

Diese Konstante mit dem Wert 06h kennzeichnet den KNXnet/IP-Header, wie in der Protokollversion 1.0 definiert.

5.2.7.2.2 KNXNETIP_VERSION_10

Diese Konstante mit dem Wert 10h kennzeichnet die KNXnet/IP-Protokollversion 1.0.

5.2.7.3 Übliche Fehlercodes

5.2.7.3.1 E_NO_ERROR

Diese Konstante mit dem Wert 00h kennzeichnet eine erfolgreiche Operation.

5.2.7.3.2 E_HOST_PROTOCOL_TYPE

Diese Konstante mit dem Wert 01h kennzeichnet, dass das angeforderte Host-Protokoll nicht von dem KNXnet/IP-Gerät unterstützt wird.

5.2.7.3.3 E_VERSION_NOT_SUPPORTED

Diese Konstante mit dem Wert 02h kennzeichnet, dass die angeforderte Protokollversion nicht von dem KNXnet/IP-Gerät unterstützt wird.

5.2.7.3.4 E_SEQUENCE_NUMBER

Diese Konstante mit dem Wert 04h kennzeichnet, dass die empfangene Folgeznummer fehlerhaft ist.

5.2.7.4 KNXnet/IP-Dienste

5.2.7.4.1 KNXnet/IP-Core-Dienste

5.2.7.4.1.1 SEARCH_REQUEST

Diese Konstante mit dem Wert 0201h kennzeichnet die Art des KNXnet/IP-Dienstes, der von einem KNXnet/IP-Client gesendet wird, um nach verfügbaren KNXnet/IP-Servern zu suchen.

5.2.7.4.1.2 SEARCH_RESPONSE

Diese Konstante mit dem Wert 0202h kennzeichnet die Art des KNXnet/IP-Dienstes, der von einem KNXnet/IP-Server gesendet wird, wenn er auf einen KNXnet/IP-SEARCH_REQUEST antwortet.

5.2.7.4.1.3 DESCRIPTION_REQUEST

Diese Konstante mit dem Wert 0203h kennzeichnet die Art des KNXnet/IP-Dienstes, der von einem KNXnet/IP-Client an einen KNXnet/IP-Server gesendet wird, um Informationen zu den Funktionen und unterstützten Diensten abzurufen.

5.2.7.4.1.4 DESCRIPTION_RESPONSE

Diese Konstante mit dem Wert 0204h kennzeichnet die Art des KNXnet/IP-Dienstes, der vom KNXnet/IP-Server als Antwort auf einen DESCRIPTION_REQUEST gesendet wird, um Informationen zur Server-Implementierung bereitzustellen.

5.2.7.4.1.5 CONNECT_REQUEST

Diese Konstante mit dem Wert 0205h kennzeichnet die Art des KNXnet/IP-Dienstes, der vom KNXnet/IP-Client gesendet wird, um einen Kommunikationskanal mit einem KNXnet/IP-Server aufzubauen.

5.2.7.4.1.6 CONNECT_RESPONSE

Diese Konstante mit dem Wert 0206h kennzeichnet die Art des KNXnet/IP-Dienstes, der vom KNXnet/IP-Server als Antwort auf ein CONNECT_REQUEST-Telegramm gesendet wird.

5.2.7.4.1.7 CONNECTIONSTATE_REQUEST

Diese Konstante mit dem Wert 0207h kennzeichnet die Art des KNXnet/IP-Dienstes, der vom KNXnet/IP-Client gesendet wird, um den Verbindungsstatus einer mit einem KNXnet/IP-Server aufgebauten Verbindung anzufordern.

5.2.7.4.1.8 CONNECTIONSTATE_RESPONSE

Diese Konstante mit dem Wert 0208h kennzeichnet die Art des KNXnet/IP-Dienstes, der vom KNXnet/IP-Server gesendet wird, wenn er einen CONNECTIONSTATE_REQUEST für eine aufgebaute Verbindung empfängt.

5.2.7.4.1.9 DISCONNECT_REQUEST

Diese Konstante mit dem Wert 0209h kennzeichnet die Art des KNXnet/IP-Dienstes, der vom KNXnet/IP-Gerät, normalerweise dem Client, gesendet wird, um eine aufgebaute Verbindung zu beenden.

5.2.7.4.1.10 DISCONNECT_RESPONSE

Diese Konstante mit dem Wert 020Ah kennzeichnet die Art des KNXnet/IP-Dienstes, der vom KNXnet/IP-Gerät, normalerweise dem Server, als Antwort auf einen DISCONNECT_REQUEST gesendet wird.

5.2.7.5 Platzhalter

5.2.7.5.1 HPAI (Host Protocol Address Information)

Die HPAI-Struktur muss den Adressinformationen entsprechen, die für die eindeutige Identifizierung eines Kommunikationskanals auf dem Host-Protokoll erforderlich sind. Ihre Größe kann bei verschiedenen Host-Protokollen variieren. Für eine spezifische Definition der HPAI siehe die entsprechenden Anhänge der KNXnet/IP-Norm zu den Host-Protokollen.

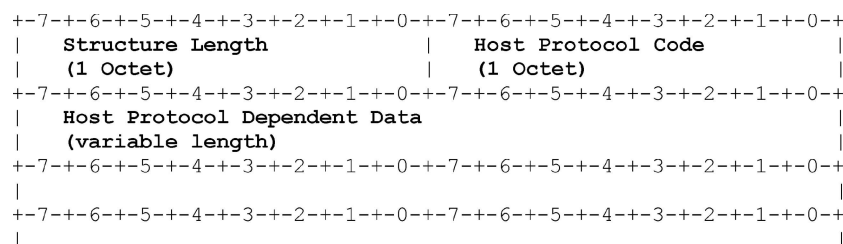


Bild 10 — HPAI-Struktur im Binärformat

5.2.7.5.2 CRI (Connection Request Information)

Die CRI-Struktur muss den zusätzlichen Informationen entsprechen, die für die verschiedenen Arten von Kommunikationskanälen zur Erfüllung einer Verbindungsanforderung notwendig sind. Da diese Struktur zwei Unterstrukturen umfassen muss, die sowohl vom Host-Protokoll unabhängige Daten als auch vom Host-Protokoll abhängige Informationen enthalten, müssen für die spezifische Definition der CRI die Beschreibung der Verbindungsart sowie die entsprechenden Teile der KNXnet/IP-Norm zu den Host-Protokollen konsultiert werden.

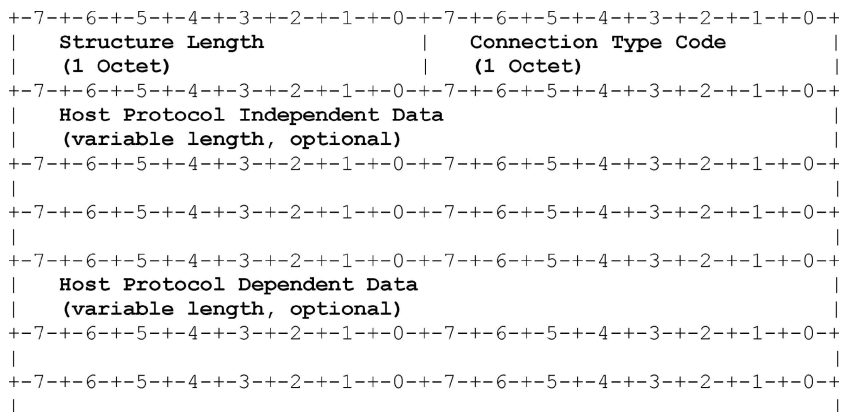


Bild 11 — CRI-Struktur im Binärformat

5.2.7.5.3 CRD (Connection Response Data Block)

Die CRD-Struktur muss dem Datenblock entsprechen, der mit dem CONNECT_RESPONSE-Frame zurückgegeben wird. Da diese Struktur zwei Unterstrukturen umfassen muss, die sowohl vom Host-Protokoll unabhängige Daten als auch vom Host-Protokoll abhängige Informationen enthalten, müssen für die spezifische Definition der CRD die Beschreibung der Verbindungsart sowie die entsprechenden Teile der KNXnet/IP-Norm zu den Host-Protokollen konsultiert werden.

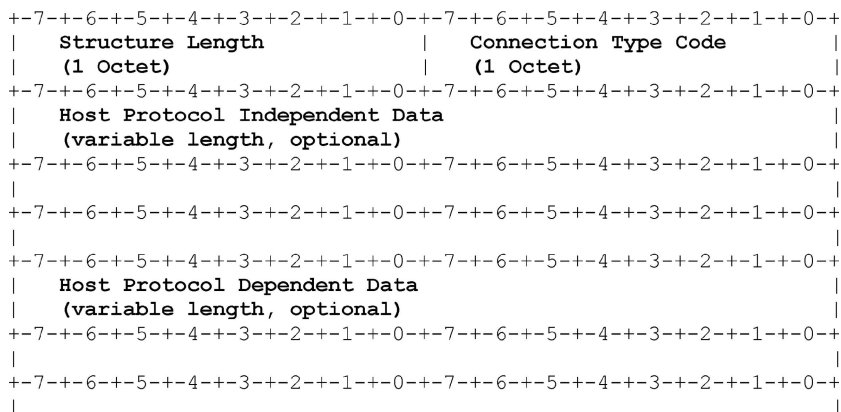


Bild 12 — CRD-Struktur im Binärformat

5.2.7.5.4 DIB (Description Information Block)

5.2.7.5.4.1 Verwendung, Format und allgemeine Anforderungen

Die DIB-Struktur muss von einem KNXnet/IP-Server verwendet werden, um einen spezifischen Block von Geräteinformationen zurückzugeben, wenn er auf einen DESCRIPTION_REQUEST antwortet.

Es müssen mindestens zwei DIB-Strukturen mit Informationen zu den Gerätefunktionen in Bezug auf (1) die Geräte-Hardware und (2) die unterstützten Dienstfamilien zurückgegeben werden.

Es können mehr als zwei DIB-Strukturen in einem einzigen DESCRIPTION_RESPONSE-Frame zurückgegeben werden.

Das erste Oktett jedes DIB muss die Länge der DIB-Struktur enthalten. Das zweite Oktett muss den Typ der DIB-Struktur angeben. Anschließend werden die eigentlichen Daten des DIB angehängt. Die Struktur muss stets eine gleiche Anzahl von Oktetten aufweisen, die gegebenenfalls dadurch erreicht wird, dass das letzte Oktett der DIB-Struktur mit 00h aufgefüllt wird.

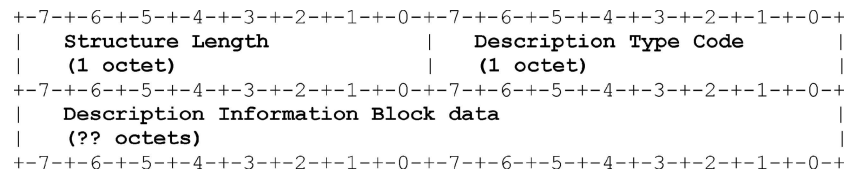


Bild 13 — DIB-Struktur im Binärformat

In Tabelle 4 sind die gültigen Codes für Beschreibungstypen aufgeführt.

Tabelle 4 — Codes für Beschreibungstypen

Beschreibungstyp	Wert	V.	Beschreibung
DEVICE_INFO	01h	1	Geräteinformationen, z. B. KNX-Medium
SUPP_SVC_FAMILIES	02h	1	Vom Gerät unterstützte Dienstfamilien
IP_CONFIG	03h	1	IP-Konfiguration des Gerätes
IP_CUR_CONFIG	04h	1	Aktuelle IP-Konfiguration des Gerätes
IP_CONFIG	05h	1	Vom Gerät verwendete/dem Gerät zugewiesene KNX-Adressen
Reserved	06h-FDh	1	Für zukünftige Verwendung reserviert
MFR_DATA	FEh	1	DIB-Struktur für weitere vom Gerätehersteller festgelegte Daten
not used	FFh	1	Nicht verwendet

5.2.7.5.4.2 DIB für Geräteinformationen

Der DIB für Geräteinformationen muss die folgende Struktur aufweisen.

[illegible]

Bild 14 — DIB-Struktur für Geräteinformationen

— **KNX-Medium**

Die Codes für das KNX-Medium sind in Tabelle 5 angegeben.

Tabelle 5 — Codes für KNX-Medium

Code für KNX-Medium	KNX-Medium
01h	reserviert
02h	TP1
04h	PL110
08h	reserviert
10h	RF
20h	KNX IP

Es muss genau ein einziges Bit gesetzt werden.

— **Gerätestatus**

Das Gerätestatus-Oktett muss folgendermaßen angegeben werden:

Tabelle 6

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
reserviert	reserviert	reserviert	reserviert	reserviert	reserviert	reserviert	Programmiermodus

— **Projektinstallationskennung**

Die Projektinstallationskennung muss folgendermaßen festgelegt werden:

Tabelle 7

Bit 15-4	Bit 3-0
Projektnummer	Installationsnummer

Die Projektinstallationskennung darf ausschließlich von der ETS zugewiesen werden und muss verwendet werden, um KNXnet/IP-Geräte in einem Projekt mit mehreren KNX-Installationen, d. h. mit mehr als 15 Bereichen mit 12 Linien, oder in einer Umgebung mit mehreren KNX-Projekten eindeutig identifizieren zu können.

— **KNX-Seriennummer des KNXnet/IP-Gerätes**

Die Seriennummer des KNXnet/IP-Gerätes muss der KNX-Seriennummer des KNXnet/IP-Gerätes entsprechen. Diese Information kann zur Identifizierung des Gerätes oder zur Festlegung der individuellen Adresse verwendet werden.

— **Routing-Multicast-Adresse des KNXnet/IP-Gerätes**

Die Routing-Multicast-Adresse des KNXnet/IP-Gerätes muss die Multicast-Adresse sein, die von einem KNXnet/IP-Router für KNXnet/IP-Routing verwendet wird. Bei KNXnet/IP-Geräten, die kein KNXnet/IP-Routing implementiert haben, muss dieser Wert auf 0.0.0.0 festgelegt werden. KNX-Seriennummer des KNXnet/IP-Gerätes. Diese Information kann verwendet werden, wenn KNXnet/IP-Routing-Nachrichten an KNXnet/IP-Router gesendet werden müssen, die nicht die standardmäßige KNXnet/IP-Routing-Multicast-Adresse verwenden, die der Multicast-Adresse der KNXnet/IP-Systemkonfiguration entsprechen muss.

— **MAC-Adresse des KNXnet/IP-Gerätes**

Die MAC-Adresse des KNXnet/IP-Gerätes ist die Ethernet-MAC-Adresse des KNXnet/IP-Gerätes. Diese Information kann zur Identifizierung des Gerätes auf Ethernet gegenüber einem Server verwendet werden, der Netzwerkressourcen, im Speziellen die Unicast-IP-Adresse für das KNXnet/IP-Gerät, zuweist.

— **Anzeigenname des Gerätes**

Der Anzeigenname des Gerätes kann eine beliebige Zeichenfolge nach ISO/IEC 8859-1 sein, die mit null (00h) endet; sie darf maximal 30 Oktette lang sein. Dieser Name kann zur Identifizierung des Gerätes durch einen Benutzer dienen. Nicht verwendete Oktette werden mit dem Null-Zeichen (00h) aufgefüllt.

5.2.7.5.4.3 DIB für unterstützte Dienstfamilien

Der DIB für unterstützte Dienstfamilien muss die folgende Struktur aufweisen.

Structure Length (1 octet)	Description Type Code (1 octet)
Service Family ID (1 Octet)	Service Family version (1 Octet)
Service Family ID (1 Octet)	Service Family version (1 Octet)
....	
Service Family ID (1 Octet)	Service Family version (1 Octet)

Bild 15 — DIB für unterstützte Dienstfamilien

Die Dienstfamilien-IDs müssen durch das hohe Oktett der Dienstart-ID repräsentiert werden. Eine Liste der Dienstart-IDs ist in Anhang A, Tabelle A.2, enthalten.

Die Version einer Dienstfamilie muss sich auf die Version des entsprechenden KNXnet/P-Kapiteldokuments beziehen. Diese Version wird nur aktualisiert, wenn das Dokument selbst aktualisiert wird, nachdem es den KNX-Prozess durchlaufen hat. Jede Version einer Dienstfamilie muss rückwärts kompatibel mit den vorhergehenden Versionen sein, d. h. alle Dienste müssen implementiert und unterstützt werden. Die Dienstfamilienversion entspricht einer ganzen Zahl. Sie stellt nicht die Implementierungsversion des Herstellers dar.

5.2.7.5.4.4 DIB für Herstellerdaten

Der DIB für Herstellerdaten weist die folgende Struktur auf.

Structure Length (1 octet)	Description Type Code (1 octet)
KNX Manufacturer ID (2 Octets)	
Any manufacturer specific data (?? Octets)	

Bild 16 — DIB für Herstellerdaten

Die KNX-Hersteller-ID muss zur eindeutigen Identifizierung des Herstellers hinzugefügt werden. Diese Information ist nicht unbedingt innerhalb der KNXnet/IP-Seriennummer (6 Oktette) verschlüsselt.

Der DIB für Herstellerdaten kann beliebige herstellersistenspezifische Daten enthalten.

5.2.7.6 Erkennung

5.2.7.6.1 SEARCH_REQUEST

Der SEARCH_REQUEST-Frame wird von einem KNXnet/IP-Client über Multicast an die Erkennungsendpunkte eines empfangenden KNXnet/IP-Servers gesendet. Da die Kommunikation mit dem Erkennungsendpunkt verbindungs- und zustandslos sein muss, müssen die Adressinformationen des Erkennungsendpunktes des Clients in den KNXnet/IP-Body eingeschlossen werden.

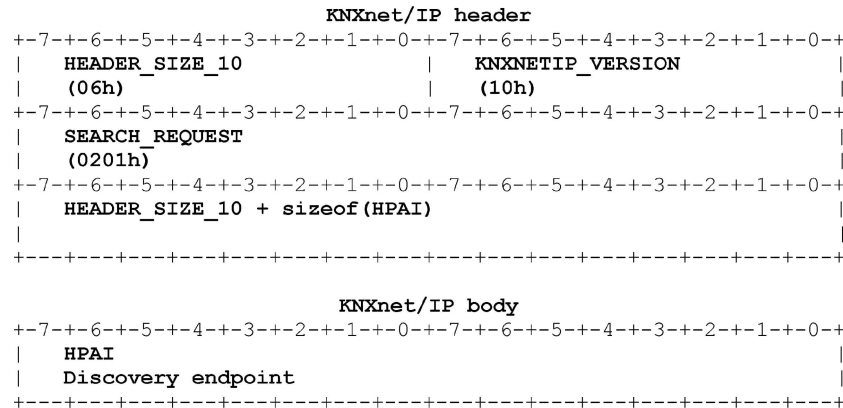


Bild 17 — SEARCH_REQUEST-Frame im Binärformat

5.2.7.6.2 SEARCH_RESPONSE

Der SEARCH_RESPONSE-Frame wird vom KNXnet/IP-Server als Antwort auf einen empfangenen SEARCH_REQUEST-Frame gesendet. Er muss an den Erkennungsendpunkt des Clients unter Verwendung der in dem empfangenen Frame enthaltenen HPAI adressiert werden.

Die HPAI des eigenen Kontrollpunktes des Servers müssen zusammen mit der Beschreibung der Geräte-Hardware und der unterstützten Dienstfamilien im KNXnet/IP-Body des SEARCH_RESPONSE-Frames übertragen werden. Wenn der KNXnet/IP-Server mehrere KNX-Verbindungen unterstützt, muss der Server jeden der eigenen Kontrollendpunkte in einem einzelnen SEARCH_RESPONSE-Frame melden.



Bild 18 — SEARCH_RESPONSE-Frame im Binärformat

5.2.7.7 Selbstbeschreibung

5.2.7.7.1 DESCRIPTION_REQUEST

Der DESCRIPTION_REQUEST-Frame wird vom KNXnet/IP-Client an den Kontrollendpunkt des KNXnet/IP-Servers gesendet, um eine Selbstbeschreibung des KNXnet/IP-Servergerätes zu erhalten.

Der KNXnet/IP-Body muss Informationen zur Rückkehradresse des Kontrollendpunktes des Clients enthalten.

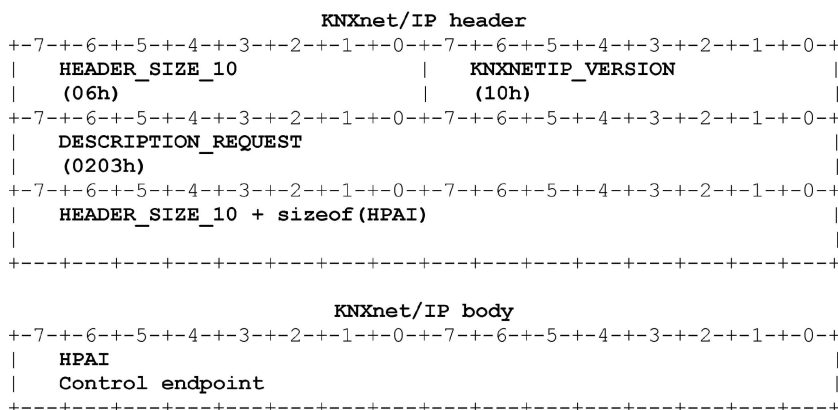


Bild 19 — DESCRIPTION_REQUEST-Frame im Binärformat

Der DESCRIPTION_RESPONSE-Frame wird vom KNXnet/IP-Server als Antwort auf einen empfangenen DESCRIPTION_REQUEST-Frame gesendet. Er muss an den Kontrollendpunkt des Clients unter Verwendung der in dem empfangenen Frame enthaltenen HPAI adressiert werden.

Die Größe des KNXnet/IP-Bodys variiert in Abhängigkeit von der Anzahl der DIB-Strukturen, die vom Server als Antwort auf den DESCRIPTION_REQUEST des Clients gesendet werden.

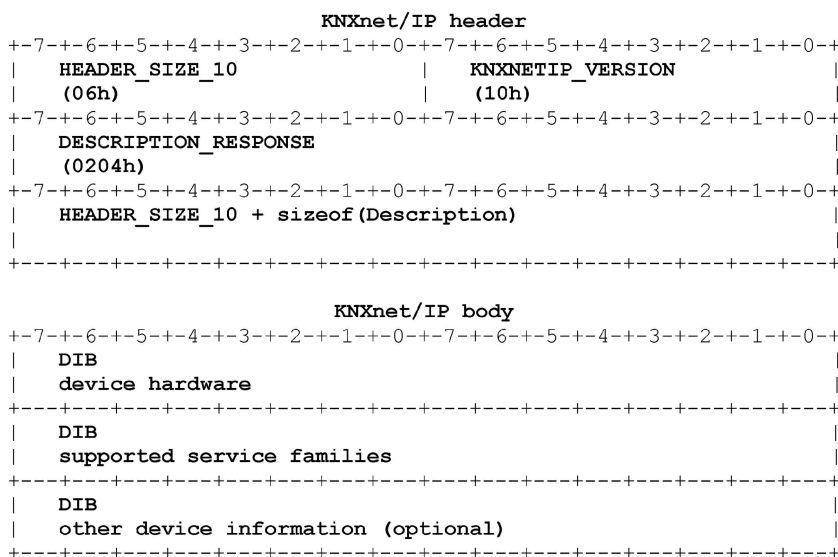


Bild 20 — DESCRIPTION_RESPONSE-Frame im Binärformat

5.2.7.8 Verbindungsmanagement

5.2.7.8.1 CONNECT_REQUEST

Der CONNECT_REQUEST-Frame wird vom KNXnet/IP-Client an den Kontrollendpunkt des KNXnet/IP-Servers gesendet. Wie bei jeder Anforderung mittels Kontrollkommunikation muss der KNXnet/IP-Body mit den Informationen zur Rückkehradresse des Kontrollendpunktes des Clients beginnen.

Anschließend folgt eine variable Datenstruktur, die CRI, die alle zusätzlichen, für die angeforderte Verbindungsart (und das zugrunde liegende Host-Protokoll) spezifischen Informationen enthält. Die genaue Festlegung dieser Struktur ist in der jeweiligen Beschreibung der spezifischen Verbindungsart enthalten.

Tabelle 8 — Verbindungsarten

Verbindungsart	Wert	V.	Beschreibung
DEVICE_MGM_CONNECTION	03h	1	Datenverbindung, die für die Konfiguration eines KNXnet/IP-Gerätes verwendet wird
TUNNEL_CONNECTION	04h	1	Datenverbindung, die für die Weiterleitung von EIB-Telegrammen zwischen zwei KNXnet/IP-Geräten verwendet wird
REMLOG_CONNECTION	06h	1	Datenverbindung, die für die Konfiguration und die Datenübertragung mit einem Server für die Fernprotokollierung verwendet wird
REMCONF_CONNECTION	07h	1	Datenverbindung, die für die Datenübertragung mit Server für die Fernkonfiguration verwendet wird
OBJSVR_CONNECTION	08h	1	Datenverbindung, die für die Konfiguration und die Datenübertragung mit einem Object-Server in einem KNXnet/IP-Gerät verwendet wird

Innerhalb der CRI legt ein Oktett den Typ des Kommunikationskanals fest, der von diesem Frame angefordert wird, und zwei Oktette sind für die Optionen für diesen Kanal reserviert. In zusätzlichen KNXnet/IP-Dokumenten sind möglicherweise weitere Verbindungsarten und zusätzliche verbindungsartspezifische Optionen festgelegt.

Die HPAI der Datenendpunkte des Clients, die für die angeforderte Datenverbindung vorgesehen sind, vervollständigen den Body des CONNECT_REQUEST-Frames.

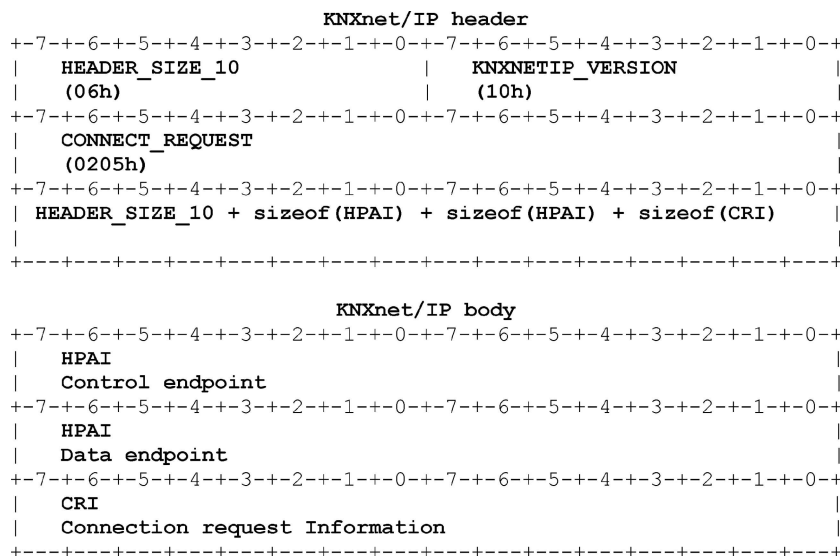


Bild 21 — CONNECT_REQUEST-Frame im Binärformat

5.2.7.8.2 CONNECT_RESPONSE

Der CONNECT_RESPONSE-Frame wird vom KNXnet/IP-Server als Antwort auf einen empfangenen CONNECT_REQUEST-Frame gesendet. Er muss an den Kontrollendpunkt des Clients unter Verwendung der in dem empfangenen Frame enthaltenen HPAI adressiert werden.

Die Größe des KNXnet/IP-Bodys variiert in Abhängigkeit davon, ob der CONNECT_REQUEST des Clients erfolgreich war oder nicht.

Wenn die Verbindungsanforderung mit allen angeforderten Optionen erfolgreich erfüllt werden konnte, enthält der Body eine Kommunikationskanal-ID, die diese Verbindung mit dem KNXnet/IP-Server eindeutig identifiziert. Die Kommunikationskanal-ID muss das erste Oktett des Bodys darstellen.

Das zweite Oktett des Bodys muss die Statusinformationen der Verbindungsanforderung enthalten. Diese Statusinformationen können Fehlerinformationen bezüglich der Anforderung selbst oder in Bezug auf die verbindungsartspezifischen Informationen enthalten.

Tabelle 9 — Übliche CONNECT_RESPONSE-Statuscodes

Fehlerkonstante	Wert	V.	Beschreibung
E_NO_ERROR	00h	1	Die Verbindung wurde erfolgreich hergestellt.
E_CONNECTION_TYPE	22h	1	Die angeforderte Verbindungsart wird von dem KNXnet/IP-Servergerät nicht unterstützt.
E_CONNECTION_OPTION	23h	1	Eine oder mehrere angeforderte Verbindungsoptionen werden von dem KNXnet/IP-Servergerät nicht unterstützt.
E_NO_MORE_CONNECTIONS	24h	1	Das KNXnet/IP-Servergerät konnte die neue Datenverbindung nicht annehmen, da die maximale Anzahl von gleichzeitigen Verbindungen bereits erreicht ist.

Die HPAI des Datenendpunktes des Servers, die für diese Datenverbindung vorbereitet wurden, stellen den nächsten Datenblock im Body eines erfolgreichen CONNECT_RESPONSE-Frames dar.

Der CRD mit verbindungsartspezifischen Antwortdaten vervollständigt den Body eines erfolgreichen CONNECT_RESPONSE-Frames.

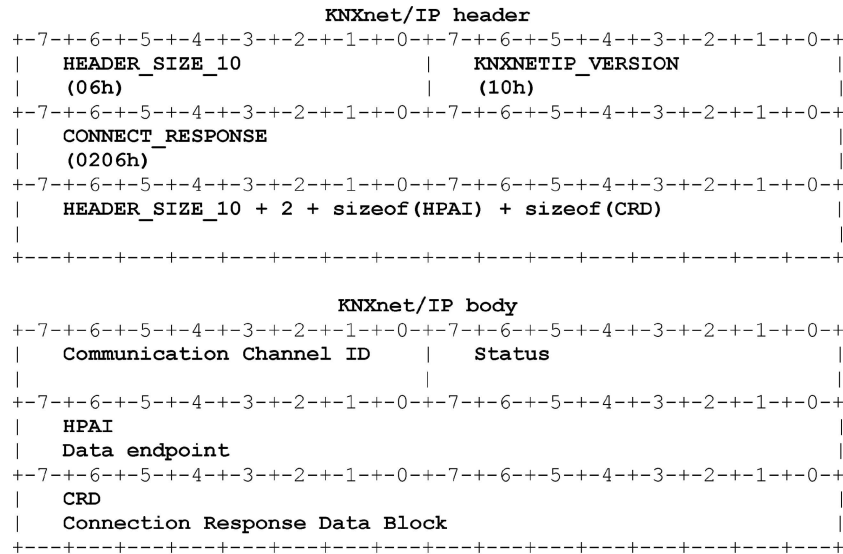


Bild 22 — CONNECT_RESPONSE-Frame im Binärformat

5.2.7.8.3 CONNECTIONSTATE_REQUEST

Der CONNECTIONSTATE_REQUEST-Frame wird vom KNXnet/IP-Client an den Kontrollendpunkt des KNXnet/IP-Servers gesendet. Das erste Oktett des KNXnet/IP-Bodys muss die Kommunikationskanal-ID enthalten, die der KNXnet/IP-Server für diese Verbindungsstatusanforderung für die eindeutige Identifizierung der Datenverbindung verwendet. Das zweite Oktett ist für die zukünftige Verwendung reserviert.

Die HPAI wird zusammen mit der Rückkehradresse des Kontrollendpunktes des Clients nach der Kommunikationskanal-ID hinzugefügt.

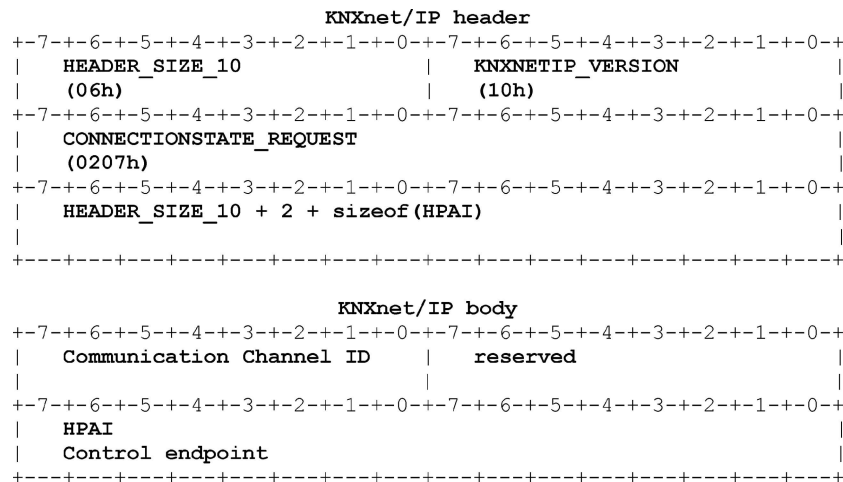


Bild 23 — CONNECTIONSTATE_REQUEST-Frame im Binärformat

5.2.7.8.4 CONNECTIONSTATE_RESPONSE

Der CONNECTIONSTATE_RESPONSE-Frame wird vom KNXnet/IP-Server als Antwort auf einen empfangenen CONNECTIONSTATE_REQUEST-Frame gesendet. Er muss an den Kontrollendpunkt des Clients unter Verwendung der in dem empfangenen Frame enthaltenen HPAI adressiert werden.

Das erste Oktett des KNXnet/IP-Bodys muss die Kommunikationskanal-ID enthalten, die der KNXnet/IP-Client an den KNXnet/IP-Server mit dem CONNECTIONSTATE_REQUEST-Frame gesendet hat.

Das zweite Oktett des KNXnet/IP-Bodys enthält die Statusinformationen der Verbindungsstatusanforderung. In der folgenden Tabelle sind die Statuscodes aufgelistet, die für den CONNECTIONSTATE_RESPONSE-Frame festgelegt sind:

Tabelle 10 — CONNECTIONSTATE_RESPONSE-Statuscodes

Fehlerkonstante	Wert	V.	Beschreibung
E_NO_ERROR	00h	1	Der Verbindungsstatus ist normal.
E_CONNECTION_ID	21h	1	Das KNXnet/IP-Servergerät konnte keine aktive Datenverbindung mit der angegebenen ID finden.
E_DATA_CONNECTION	26h	1	Das KNXnet/IP-Servergerät hat bezüglich der Datenverbindung mit der angegebenen ID einen Fehler festgestellt.
E_KNX_CONNECTION	27h	1	Das KNXnet/IP-Servergerät hat bezüglich der EIB-Bus-/KNX-Subsystem-Verbindung mit der angegebenen ID einen Fehler festgestellt.

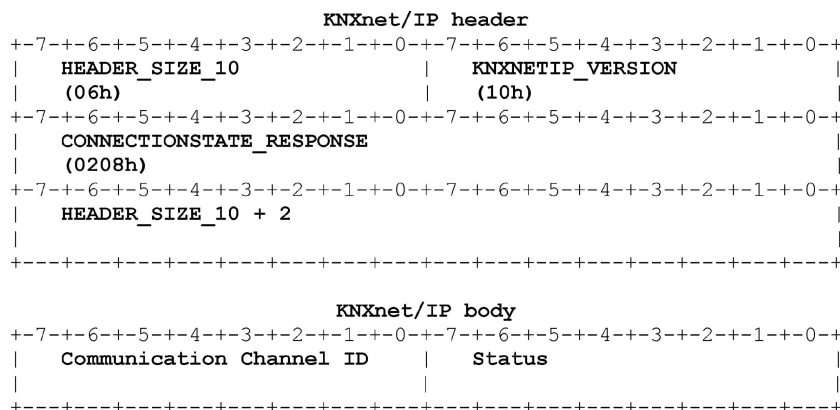


Bild 24 — CONNECTIONSTATE_RESPONSE-Frame im Binärformat

5.2.7.8.5 DISCONNECT_REQUEST

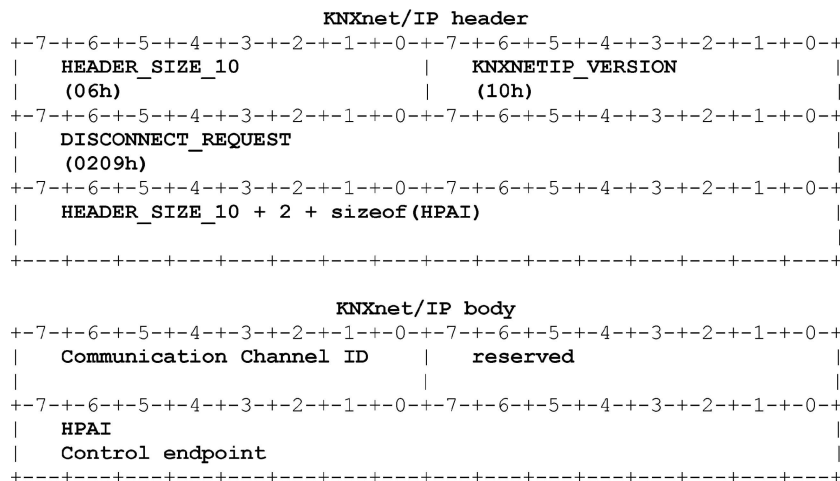


Bild 25 — DISCONNECT_REQUEST-Frame im Binärformat

5.2.7.8.6 DISCONNECT_RESPONSE

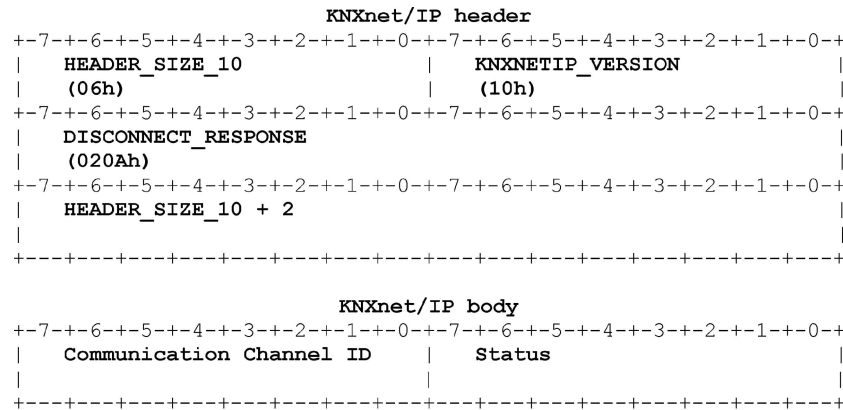


Bild 26 — DISCONNECT_RESPONSE-Frame im Binärformat

5.2.8 IP-Netzwerke

5.2.8.1 Einleitung

Das KNXnet/IP-Protokoll wird für das Tunneln oder Routen von KNX-Daten über das weit verbreitete Internet Protocol (IP) verwendet und ermöglicht den Fernzugriff und die Fernwartung über große Entfernungen hinweg sowie die Nutzung als Hochgeschwindigkeits-Backbone für KNX-Netzwerke.

Dieser Teil der Norm legt fest, welche IP-Parameter und Funktionen von KNXnet/IP unterstützt werden.

Es wird vorausgesetzt, dass der Anwender dieser Norm mit den Internetprotokollen TCP und UDP vertraut ist.

5.2.8.2 Physikalisches Netzwerk und logisches Netzwerk

IP-Netzwerke unterscheiden sich von KNX-Netzwerken: KNX-Netzwerke sind von Natur aus physikalische Busse. Dies impliziert, dass alle Geräte auf dem Kanal standardmäßig alle Pakete empfangen, die über das Netzwerk übermittelt werden. Des Weiteren ist es bei einem neuen Gerät, das dem Netzwerk hinzugefügt wird, nicht erforderlich, dass andere Geräte im Netzwerk dieses Gerät erst erkennen, bevor sie mit ihm Pakete austauschen können. Um ein Telegramm über KNX zu übertragen, ist es nur erforderlich, dass ein Gerät das Telegramm physikalisch an den Bus übermitteln kann. Es reicht aus, wenn ein Gerät mit einem KNX-Netzwerk physikalisch verbunden ist, damit es in der Lage ist, Telegramme mit anderen Geräten auf dem Kanal auszutauschen.

Im Gegensatz dazu sind IP-Netzwerke von logischer, nicht von physikalischer Natur. Es gibt eine Reihe von unterschiedlichen physikalischen Medien, die die IP-Kommunikation unterstützen, und alle diese Medien sollten für das Tunneling von KNXnet/IP-Frames geeignet sein. Da es sich um einen logischen Kanal handelt, ist es erforderlich, den Kanal zu „bauen“, indem jedes Gerät auf dem Kanal über die Existenz der anderen Geräte auf diesem Kanal informiert wird. Das heißt, bevor ein Gerät ein Paket an ein anderes Gerät auf einem IP-Kanal übermitteln kann, muss es erst mitgeteilt bekommen, wie ein Paket an jenes Gerät im Speziellen gesendet werden muss, d. h. es muss die IP-Adresse des Gerätes kennen.

Ein weiterer bedeutender Unterschied zwischen physikalischen und logischen Netzwerken liegt in der Tatsache, dass es im Fall von typischen physikalischen Netzwerken möglich ist, feste obere Grenzen für die Zeitdauer festzulegen, die ein Paket für die Übertragung von einem Gerät auf ein anderes Gerät benötigt, sobald es auf dem Kanal transportiert wird. Dies ist bei IP-Netzwerken nicht immer möglich. Die Paketübertragungszeiten zwischen KNXnet/IP-Geräten auf einem IP-Kanal weichen erfahrungsgemäß wesentlich stärker von denen bei nativen KNX-Geräten ab.

Der IP-Kanal wird von einer Vielzahl von KNXnet/IP-Geräten als dazwischenliegender Transportmechanismus für KNX-Telegramme verwendet. Wenn ein KNXnet/IP-Paket auf einem IP-Kanal transportiert wird, wird eine IP-Nachricht, in die das KNX-Telegramm eingekapselt ist, an die anderen KNXnet/IP-Geräte auf diesem IP-Kanal gesendet. Der IP-Kanal wird durch eine Liste von Unicast-IP-Adressen definiert, wobei für jedes KNXnet/IP-Gerät genau eine Unicast-IP-Adresse festgelegt ist. Es gibt keine Begrenzung für die Anzahl der KNXnet/IP-Geräte in einem einzelnen IP-Netzwerk.

5.2.8.3 Transportmechanismus

IP ist ein Protokoll der Vermittlungsschicht. Es ist mit einer Vielzahl von physikalischen Medien und Sicherungsschicht-Protokollen einsetzbar. Dieses Dokument enthält jedoch keine Festlegungen zu den Sicherungs- oder physikalischen Schichten des IP-Protokollstapels.

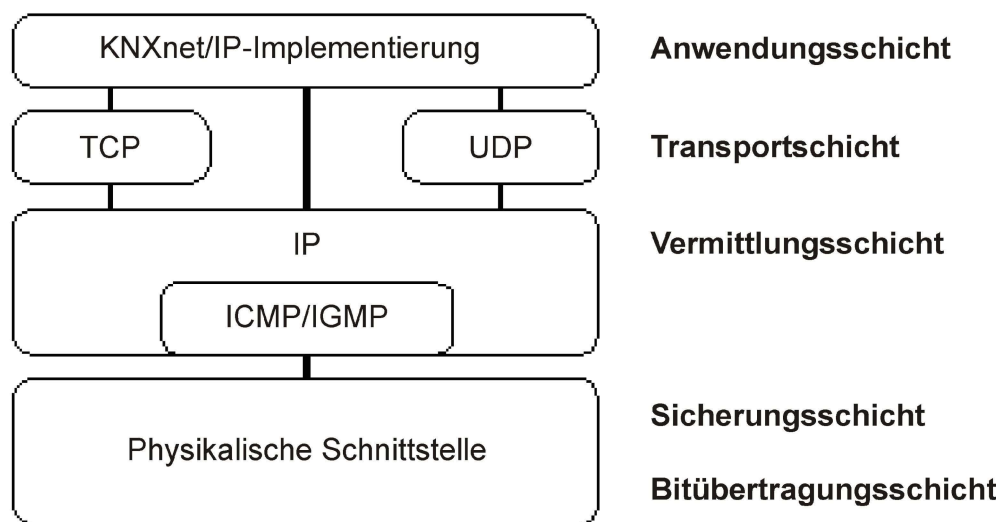


Bild 27 — IP-Protokollstapel

Die drei Mechanismen, die am häufigsten für den Transport von IP-Paketen eingesetzt werden, sind Raw IP, TCP und UDP. TCP und UDP sind Transportprotokolle, die auf IP aufsetzen.

TCP implementiert einen zuverlässigen, verbindungsorientierten Transportdienst zwischen Endpunkten. Es gewährleistet die korrekte Übertragung und behält die Reihenfolge des empfangenen Datenstroms bei.

Im Gegensatz dazu implementiert UDP einen bestmöglichen Datagrammdienst. Raw IP ist ein Mechanismus, der von vielen Betriebssystemen implementiert wird, um Host-Anwendungen mit direktem Zugriff auf die IP-Schicht bereitzustellen, ohne einen Transportdienst anzubieten.

Obwohl es möglich ist, jedes dieser Protokolle als KNXnet/IP-Host-Protokoll zu verwenden, ist es anwendungstechnisch sehr viel einfacher, Raw IP auszuklammern und die Spezifikation auf TCP und UDP zu beschränken.

5.2.8.4 UDP und TCP

Da das KNX-Protokoll selbst Empfangsbestätigungen zwischen Endpunkten vorsieht, ist es nicht erforderlich zu gewährleisten, dass getunnelte IP-Pakete, die KNX-Datentelegramme enthalten, zwischen den KNXnet/IP-Geräten empfangen werden. In diesem Fall erzeugt die Verwendung eines zuverlässigen Transportdienstes einen unnötigen Protokoll-Overhead, wodurch TCP an Effizienz gegenüber UDP oder Raw IP verliert. Auf der anderen Seite werden zwischen KNXnet/IP-Geräten Konfigurations- und Statusmeldungen ausgetauscht, die keine getunnelten KNX-Daten enthalten und dennoch zuverlässig empfangen werden müssen.

Ein Vorteil von TCP ist der zuverlässige Übertragungsdienst, durch den die Beibehaltung der Reihenfolge der empfangen Pakete gewährleistet wird. Nachteile sind die fehlende Unterstützung der Multicast-Adressierung und die geringe Effizienz gegenüber UDP. TCP verbraucht für die Implementierung auch mehr Ressourcen des KNXnet/IP-Gerätes als UDP. UDP ist bei der Übertragung getunnelter KNX-Datennachrichten effizienter, kann aber aufgrund des fehlenden zuverlässigen Übertragungsdienstes nicht gewährleisten, dass die Paketreihenfolge erhalten bleibt.

Aufgrund der erhöhten Effizienz in Bezug auf den Transport von KNX-Datennachrichten und der Unterstützung der Multicast-Adressierung wird UDP standardmäßig für die Kommunikation zwischen KNXnet/IP-Geräten eingesetzt. Alle KNXnet/IP-Geräte müssen daher UDP unterstützen. TCP kann aufgrund der Zuverlässigkeitsvorteile zusätzlich zu UDP unterstützt werden. Die Unterstützung von TCP in KNXnet/IP-Geräten ist jedoch optional.

Um die Beibehaltung der Paketreihenfolge dennoch sicherzustellen, gibt es die Folgezähler-Option, die dem Connection-Header hinzugefügt werden kann, damit diese in eine Reihenfolge gebracht werden können.

Mit UDP können Datagramme entweder mithilfe der Unicast-Adressierung oder der Multicast-Adressierung gesendet werden. Unter Unicast versteht man die Übertragung zwischen zwei Punkten, d. h. ein Datagramm wird von einem IP-Host an einen einzigen anderen IP-Host gesendet. Wenn dasselbe Datagramm an mehrere IP-Hosts gesendet wird, wie es für das Routing von KNX-Daten erforderlich ist, ist es effizienter, die Multicast-Adressierung zu verwenden. Es wird daher empfohlen, dass KNXnet/IP-Geräte sowohl die Unicast- als auch die Multicast-IP-Adressierung unterstützen, wobei es nicht erforderlich ist, dass ein KNXnet/IP-Gerät Multicasts unterstützt, um sich mit einem anderen multicast-fähigen KNXnet/IP-Gerät austauschen zu können.

5.2.8.5 IP-Adresszuweisung

5.2.8.5.1 IP-Unicast-Adresse

5.2.8.5.1.1 Feste IP-Adresse

Eine feste IP-Adresse wird dem Gerät über eine Benutzerschnittstelle, über die ETS oder ein anderes Tool zugewiesen. Diese IP-Adresszuweisung ist fest.

Jedes KNXnet/IP-Gerät muss fest zugewiesene IP-Adressen unterstützen.

5.2.8.5.1.2 BootP/DHCP

Ein BootP- oder DHCP-Server weist einem Gerät automatisch eine IP-Adresse zu. Die Konfiguration beider Server-Typen ist Teil der Netzwerkadministration und ist auf allen Netzwerk-Server-Plattformen verfügbar (Windows NT Server, Windows 2000 Server, Windows XP Server, Unix, Linux). Vorkonfigurierte DHCP-Server wie DSL-Modems oder ISDN-Router können auch eingesetzt werden.

Auf einem KNXnet/IP-Gerät muss entweder ein BootP- oder ein DHCP-Client implementiert werden.

5.2.8.5.1.3 AutoIP

Ein Gerät mit AutoIP kann sich selbst eine Unicast-IP-Adresse im Bereich von 169.254.1.0 bis 169.254.254.255 zuweisen.

Ein KNXnet/IP-Gerät kann AutoIP implementieren.

Prozedur der IP-Adresszuweisung

Tabelle 11 beschreibt die Prozedur der Adresszuweisung für KNXnet/IP-Geräte.

Tabelle 11 — Prozedur der Adresszuweisung

1	WENN	KNX-Gerät eine feste IP-Adresse zugewiesen wurde,	DANN	→ 2
			SONST	→ 6
2	WENN	Adresszuweisung durch BootP oder DHCP aktiviert ist,	DANN	→ 3
			SONST	muss bereits zugewiesene feste IP-Adresse verwenden werden.
3	WENN	Adresszuweisung durch BootP oder DHCP erfolgreich war,	DANN	kann neu zugewiesene IP-Adresse verwenden werden.
			SONST	→ 4
4	WENN	Adresszuweisung durch AutoIP aktiviert ist,	DANN	→ 5
			SONST	muss bereits zugewiesene feste IP-Adresse verwenden werden.
5	WENN	Adresszuweisung durch AutoIP erfolgreich war,	DANN	kann neu zugewiesene IP-Adresse verwenden werden.
			SONST	muss bereits zugewiesene feste IP-Adresse verwenden werden.
6	WENN	Adresszuweisung durch BootP oder DHCP aktiviert ist,	DANN	→ 7
			SONST	→ 8
7	WENN	Adresszuweisung durch BootP oder DHCP erfolgreich war,	DANN	kann neu zugewiesene IP-Adresse verwenden werden.
			SONST	→ 8
8	WENN	Adresszuweisung durch AutoIP aktiviert ist	DANN	→ 9
			SONST	wurde keine IP-Adresse zugewiesen; 0.0.0.0 muss im IP-Adressfeld der HPAI eingegeben werden.
9	WENN	Adresszuweisung durch AutoIP erfolgreich war	DANN	kann neu zugewiesene IP-Adresse verwenden werden.
			SONST	wurde keine IP-Adresse zugewiesen; 0.0.0.0 muss im IP-Adressfeld der HPAI eingegeben werden.

5.2.8.5.2 IP-Multicast-Adressen

5.2.8.5.2.1 Multicast-Adresse der KNXnet/IP-Systemkonfiguration

Um sicherzustellen, dass jedes KNXnet/IP-Gerät von den KNXnet/IP-Core-Erkennungsdiensten erreicht werden kann, wird eine „Multicast-Adresse der Systemkonfiguration“ festgelegt. Der Wert der „Multicast-Adresse der Systemkonfiguration“ muss 224.0.23.12 sein.

5.2.8.5.2.2 Multicast-Adresse für KNXnet/IP-Routing

Jedes KNXnet/IP-Gerät, das KNXnet/IP Routing implementiert hat, muss über eine „Routing-Multicast-Adresse“ verfügen. Diese Adresse wird von der „Multicast-Adresse der Systemkonfiguration“ abgeleitet, indem ein Adressabstand hinzugefügt wird. Dieser Adressabstand hat einen Default-Wert von Null. Sind mehrere Installationen in einem Projekt vorhanden, müssen KNXnet/IP-Routern in separaten Installationen unterschiedliche Routing-Multicast-Adressen zugewiesen werden.

5.2.8.6 KNXnet/IP-Host-Protokoll

5.2.8.6.1 Gerätespezifikation

Ein KNXnet/IP-Gerät muss sich wie jeder Standard-IP-Host verhalten, der IP-Pakete mit jedem anderen IP-Host entweder in demselben IP-Subnetz oder irgendwo anders in der IP-Netzwerk-Cloud austauschen kann. Ein KNXnet/IP-Gerät muss über eine einzelne Unicast-IP-Adresse verfügen und muss zu mindestens einer IP-Multicast-Gruppe gehören, kann aber zu bis zu 16 Multicast-Gruppen gehören. KNXnet/IP-Geräte müssen Multicasting unterstützen.

KNXnet/IP-Geräte müssen folgende IP-Protokolle unterstützen: ARP, ICMP, IGMP, BootP/DHCP⁴⁾ und UDP. Zusätzlich können sie andere Protokolle wie RARP, TCP, NTP, FTP, HTTP, SMTP, DNS oder SNMP unterstützen.

Um die Kommunikation mit einem KNXnet/IP-Gerät über TCP/UDP zu starten, muss zusätzlich zu der individuellen IP-Adresse eine feste Portnummer für die Erkennungsprozedur festgelegt werden.

Um das Routing von IP-Paketen zwischen Subnetzen oder durch das Internet zu unterstützen, müssen KNXnet/IP-Geräte mit den entsprechenden Standardmechanismen (IP-Router, Switches usw.), die für die Ausführung der IP-Routing-Funktionen erforderlich sind, kompatibel sein.

5.2.8.6.2 HPAI (Host Protocol Address Information)

Die HPAI enthalten die Informationen, die für die eindeutige Identifizierung des Endpunktes einer IP-Transportverbindung erforderlich sind. Dazu gehören die Adresse der Vermittlungsschicht sowie die Kennung der Transportschicht, die sogenannte Portnummer. Sowohl die IP-Adresse als auch die Portnummer werden binär in der Oktett-Reihenfolge des Netzwerks gespeichert.

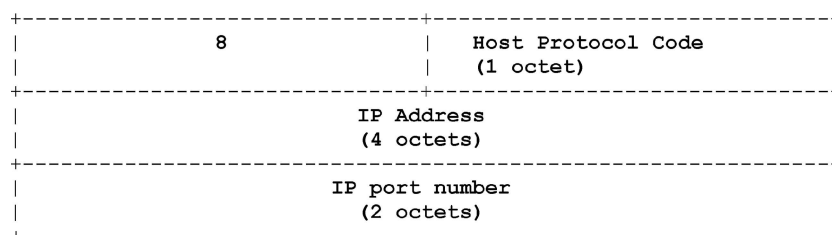


Bild 28 — IP-HPAI im Binärformat

4) BootP/DHCP: Es muss einer der beiden Dienste implementiert werden.

Tabelle 12 — Host-Protokoll-Codes für das IP-Netzwerk

Name der Konstante	Wert	V.	Beschreibung
IPV4_UDP	01h	1	Identifiziert eine IPv4-Adresse (Internet Protocol Version 4) und eine Portnummer für die Kommunikation über UDP.
IPV4_TCP	02h	1	Identifiziert eine IPv4-Adresse (Internet Protocol Version 4) und eine Portnummer für die Kommunikation über TCP.

5.2.8.6.3 KNXnet/IP-Endpunkte

5.2.8.6.3.1 Allgemeines

Die Verwendung der KNXnet/IP-Endpunkte als logische Sicht der Kommunikation zwischen KNXnet/IP-Geräten erzeugt unter Verwendung des spezifischen Host-Protokolls eine hohe Flexibilität der eigentlichen Implementierung.

Wie bereits erwähnt, können KNXnet/IP-Geräte UDP, oder optional TCP, als Transportschichtprotokoll für die IP-Kommunikation unterstützen. Diese IP-Kanäle können zwischen den KNXnet/IP-Geräten mithilfe der HPAI-Struktur der KNXnet/IP-Frames dynamisch verhandelt werden. Aufgrund der Punkt-zu-Multipunkt-Anforderung der Erkennungskommunikation ist UDP am Port 3671 der einzige zulässige Transportmechanismus.

Für den Erkennungsendpunkt muss die KNXnet/IP-Portnummer 3671 verwendet werden. Für die Daten- und Kontrollendpunkte kann sie ebenfalls verwendet werden. Dies ist aber nicht obligatorisch. Die Portnummer, die vom KNXnet/IP-Server als Antwort auf einen CONNECT_REQUEST zurückgegeben wird, kann eine beliebige gültige Portnummer sein.

Es ist daher möglich, ein KNXnet/IP-Gerät zu implementieren, das nur einen einzigen bidirektionalen UDP-Port für die gesamte Kommunikation verwendet, oder ein KNXnet/IP-Gerät, das zwei unidirektionale UDP-Ports für den Erkennungsendpunkt und für jeden Kontrollendpunkt sowie TCP-Ports für Datenverbindungen verwendet.

5.2.8.6.3.2 Erkennungsendpunkt

Für die Kommunikation zwischen Erkennungsendpunkten ist nur UDP zulässig. Der Versuch, eine Kommunikation über TCP anzufordern, führt zu einem Fehler des Host-Protokoll-Typs.

Der KNXnet/IP-Client muss einen SEARCH_REQUEST-Frame von einem beliebigen Quellport aus mittels lokalem UDP-Broadcast an das feste Ziel des Erkennungsendpunktes am Port 3671 senden. Jeder KNXnet/IP-Server, der diese Anforderung empfängt, muss die Quellinformationen auf IP-Ebene ignorieren. Es ist nur die HPAI des empfangenen KNXnet/IP-Frames relevant. Die KNXnet/IP-Server müssen dann ihre SEARCH_RESPONSE-Frames von einem beliebigen Quellport aus mittels UDP-Unicast an das angeforderte Ziel senden.

5.2.8.6.3.3 Kontrollendpunkt

Für die Kommunikation zwischen Kontrollendpunkten sind UDP und TCP zulässig. Jedes KNXnet/IP-Gerät muss daher die Kommunikation über UDP unterstützen. Die Kommunikation über TCP ist optional.

Der KNXnet/IP-Client muss die Portinformationen zu den Kontrollendpunkten des KNXnet/IP-Servers über die HPAI der SEARCH_RESPONSE-Frames empfangen. Ein KNXnet/IP-Server darf diesen Port nicht mehr ändern, sobald er gemeldet wurde.