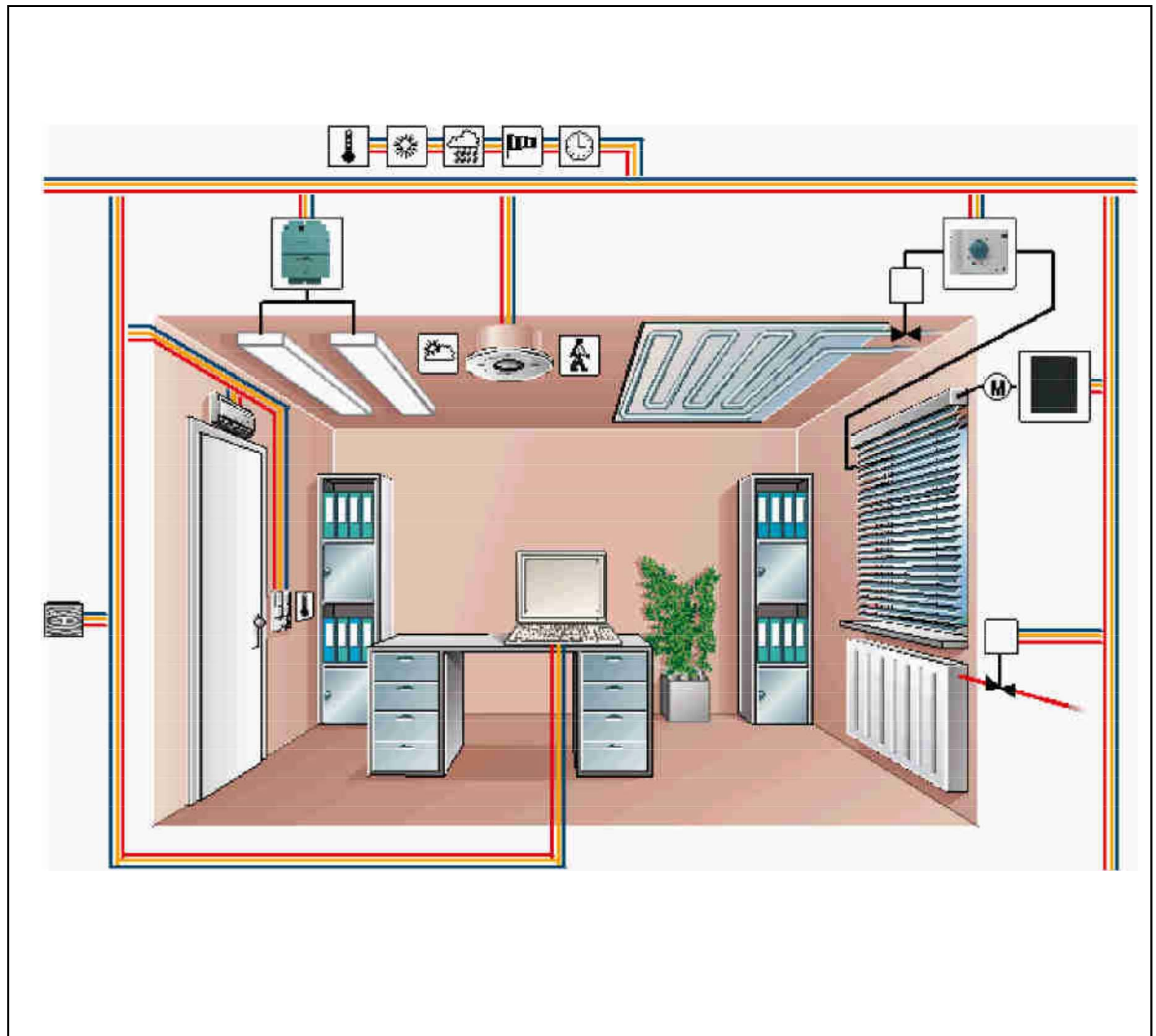


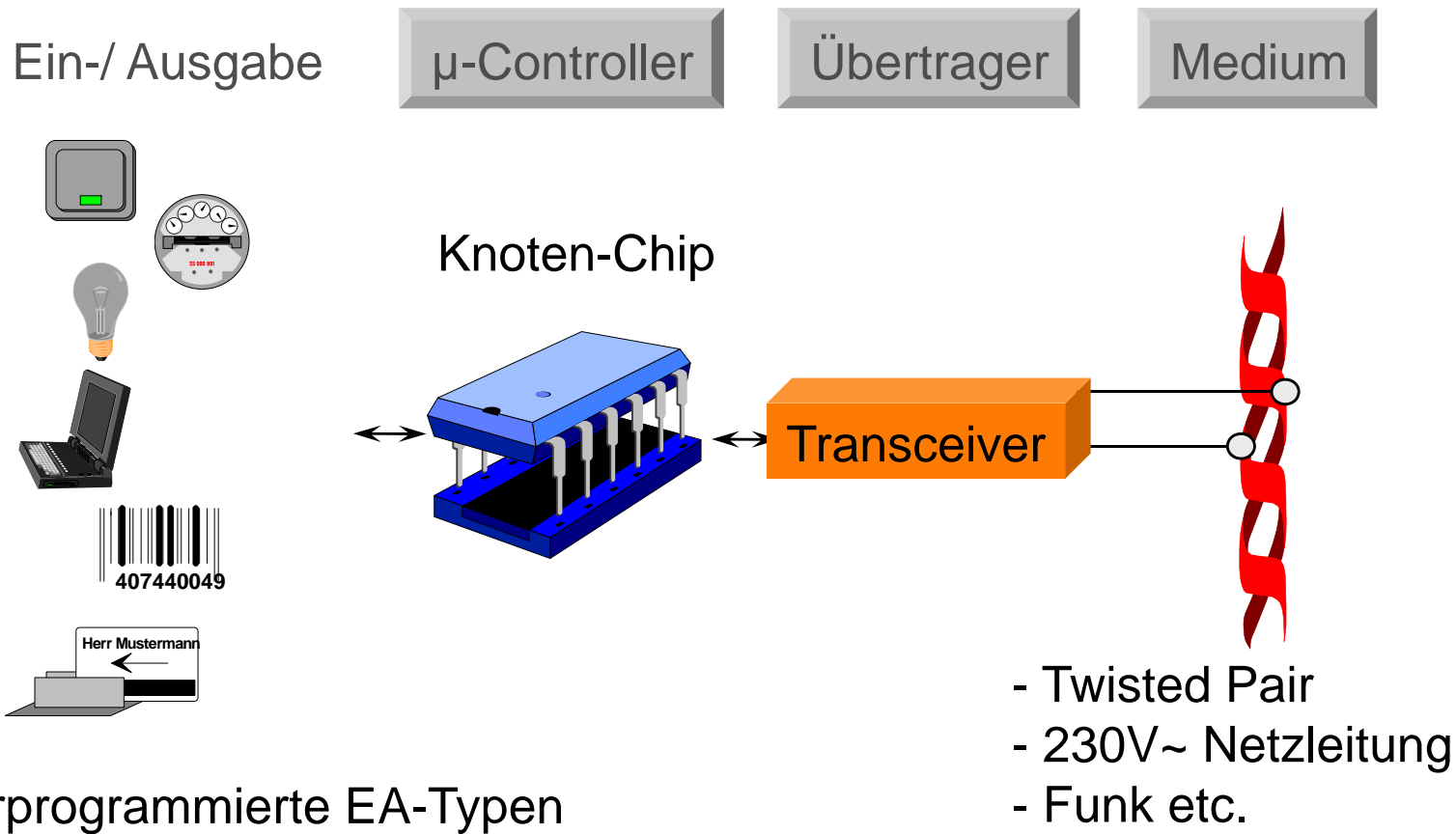
Cyber Physical Systems:

Bottom-Up

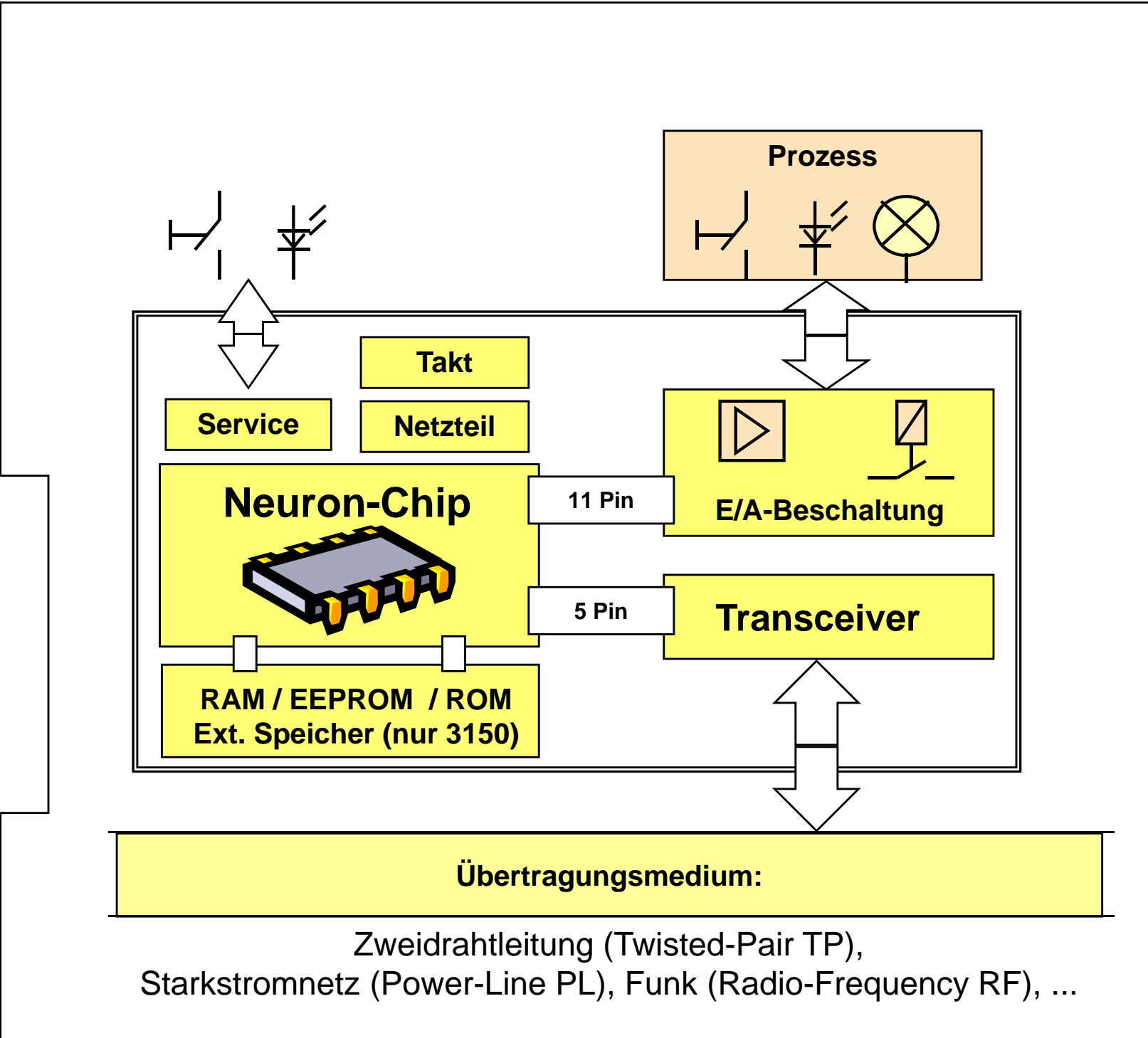


Technik/ Knoten

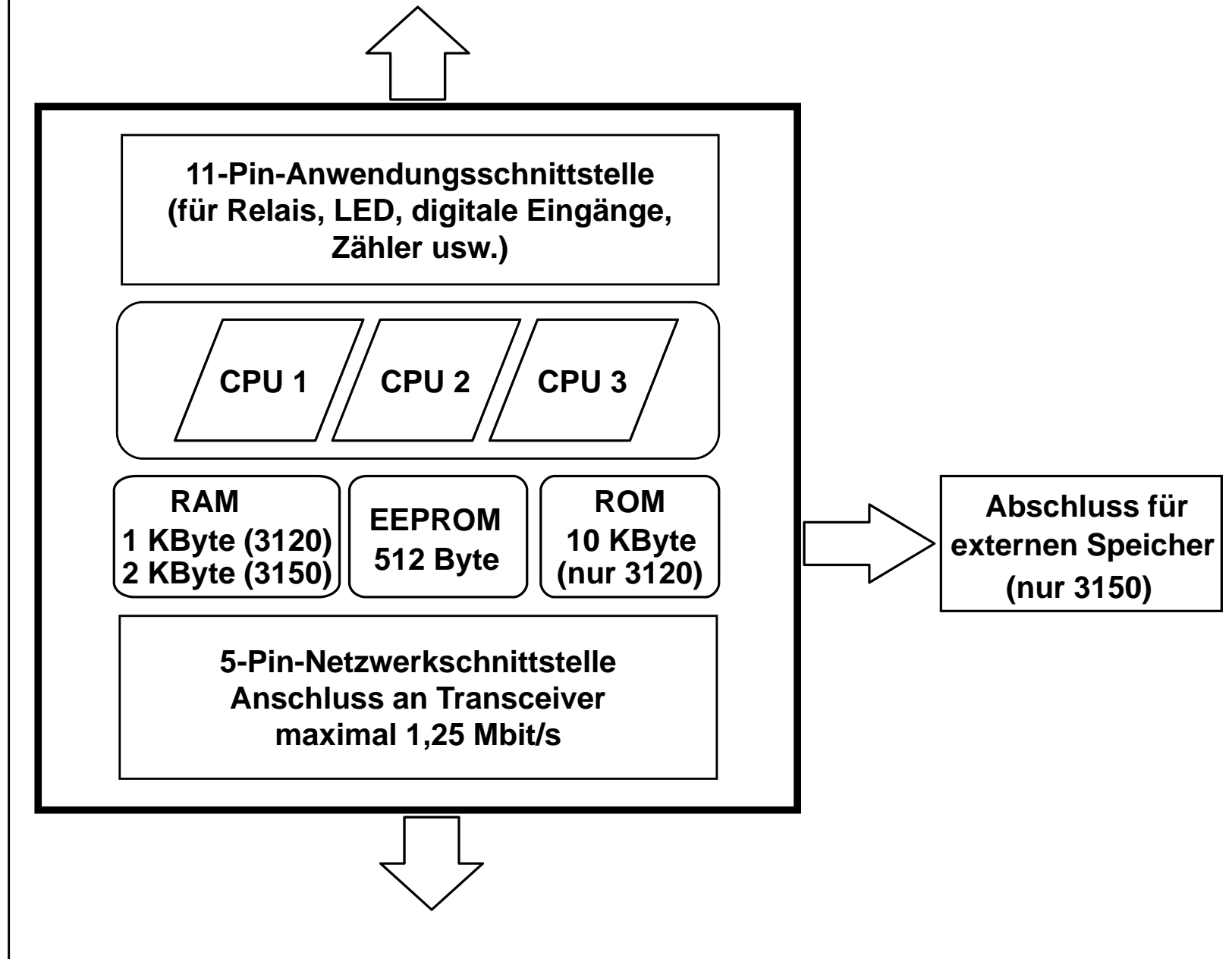
Ein CPS-Knoten und seine Umgebung



□ Knotenaufbau
(konkretes
Beispiel LON)

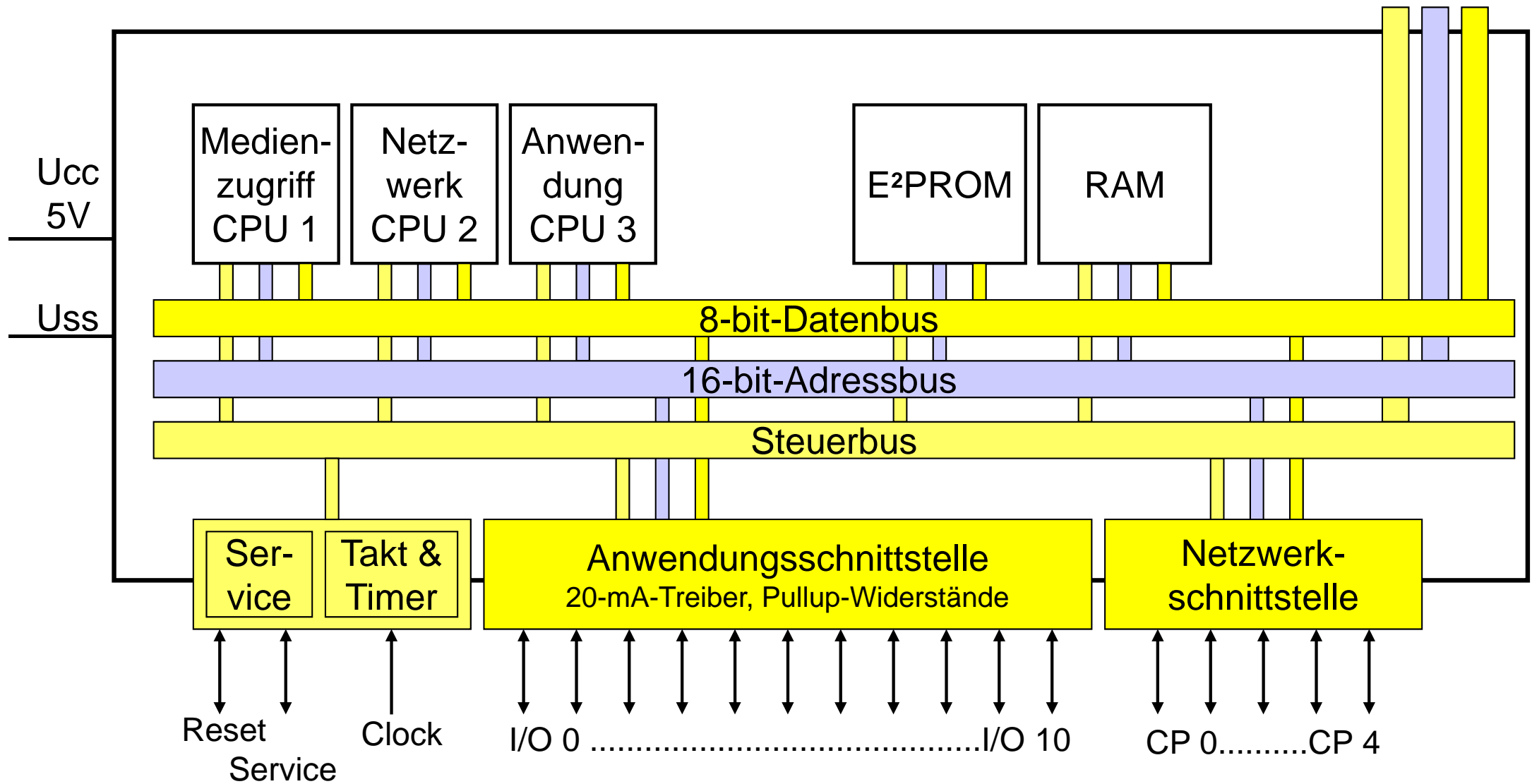


Neuron 3120 und Neuron 3150



Neuron 3150

Erweiterungsbus



Neuron Chip

CPU 2:
Netzwerk-
management,
Adressierung

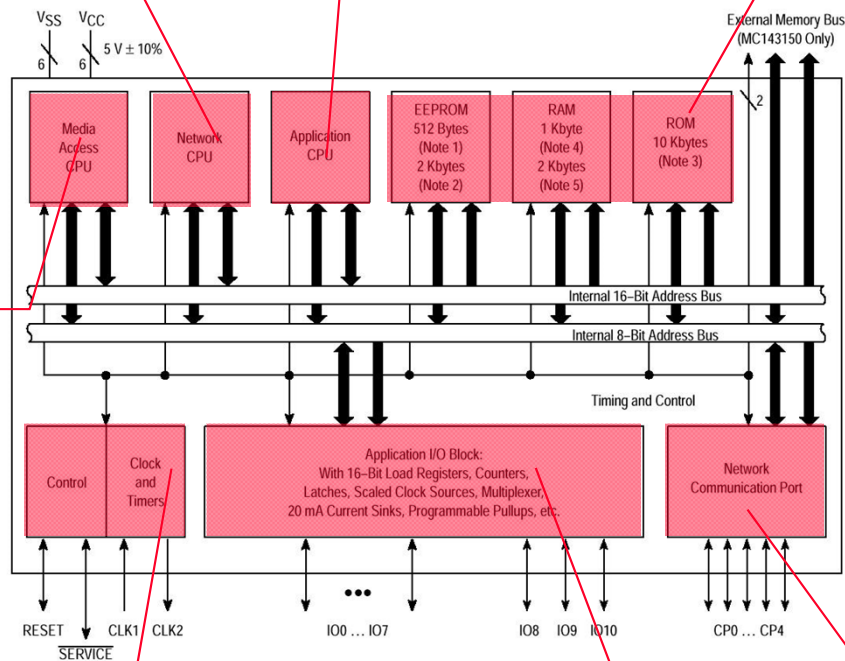
CPU 3:
Anwendungs-
programm

Speicher für:

- Neuron-ID (MAC)
- Firmware

- Anwendungsprogramm
- Konfiguration
- Verknüpfungen
- Betriebsdaten
- + externer Speicher

CPU 1:
Buszugriff
Kollisions-
erkennung



Kontrolle & Timer

I/O

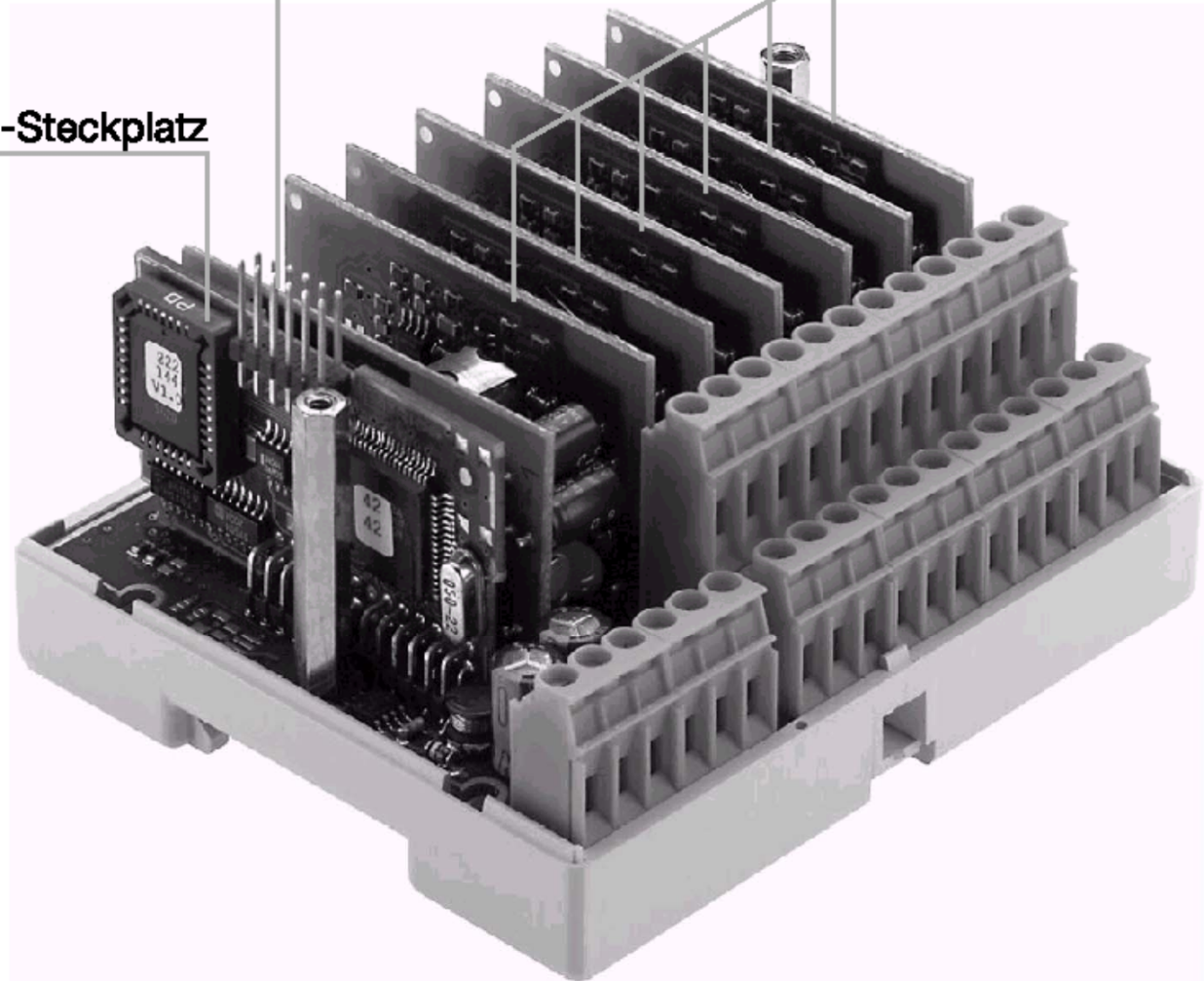
Transceiveranschluß

Knotenaufbau: Beispiel

Transceiver-Steckplatz

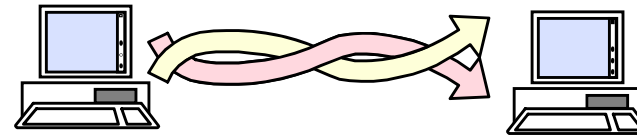
Neuron-Steckplatz

6 E/A-Steckplätze



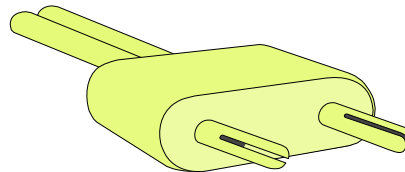
Übertragungs -medien

Twisted-Pair (TP)



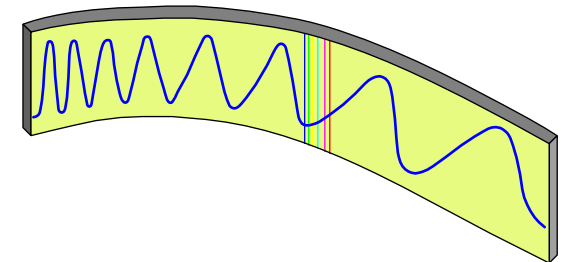
Verdrillte Zweidraht-Leitung

Power Line (PL)

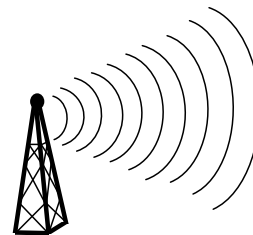


~~~~~ + 230 V

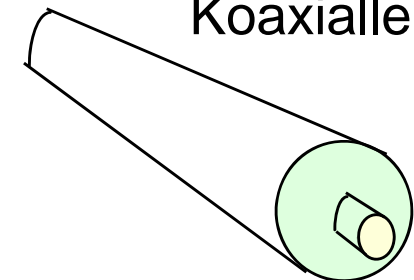
Lichtwellenleiter  
(Fiber Optic)



Funk  
(Radio Frequency RF)



Koaxialleiter



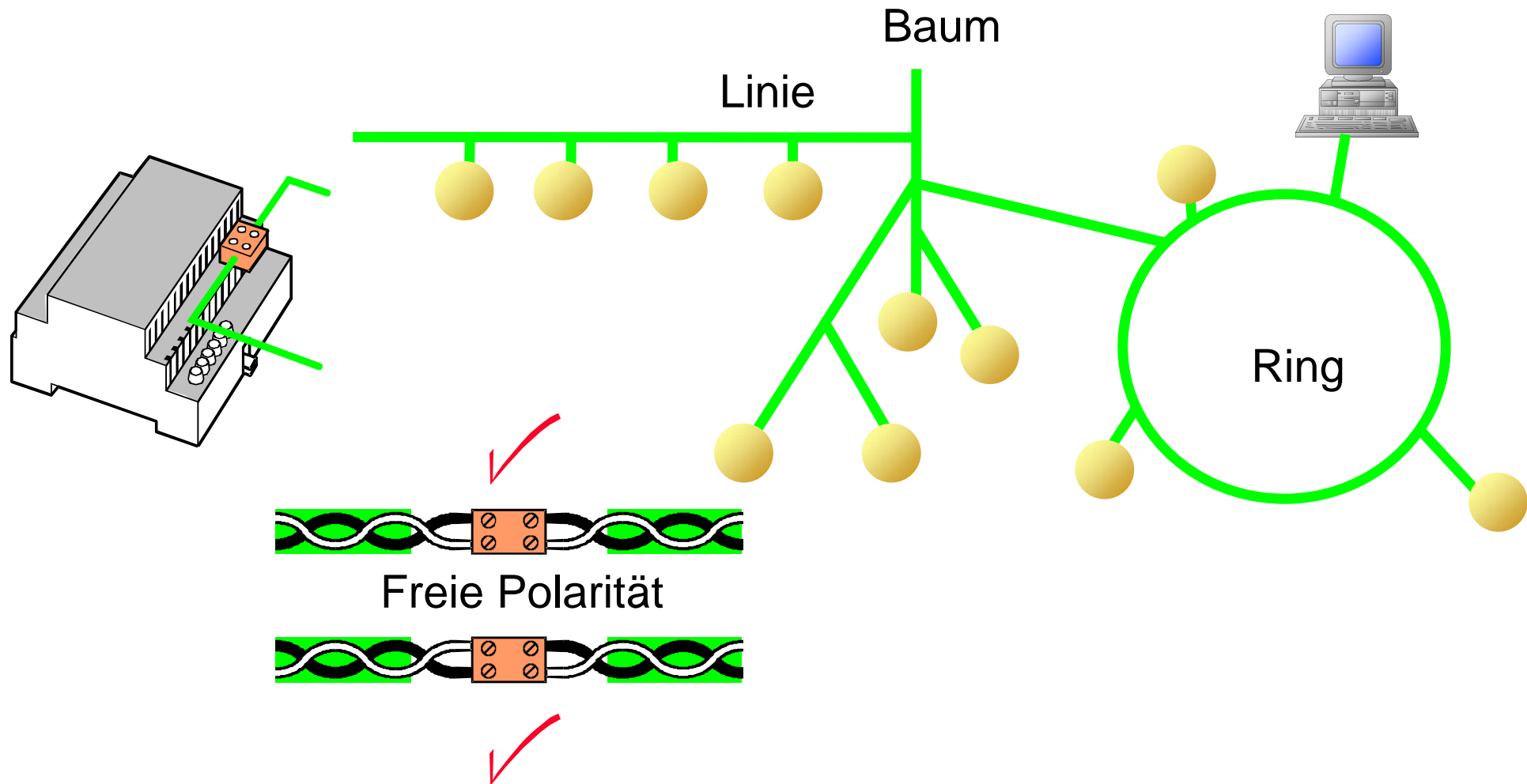
Übertragungs-  
medien

Übersicht

# Physikalische Netzwerk- Strukturierung

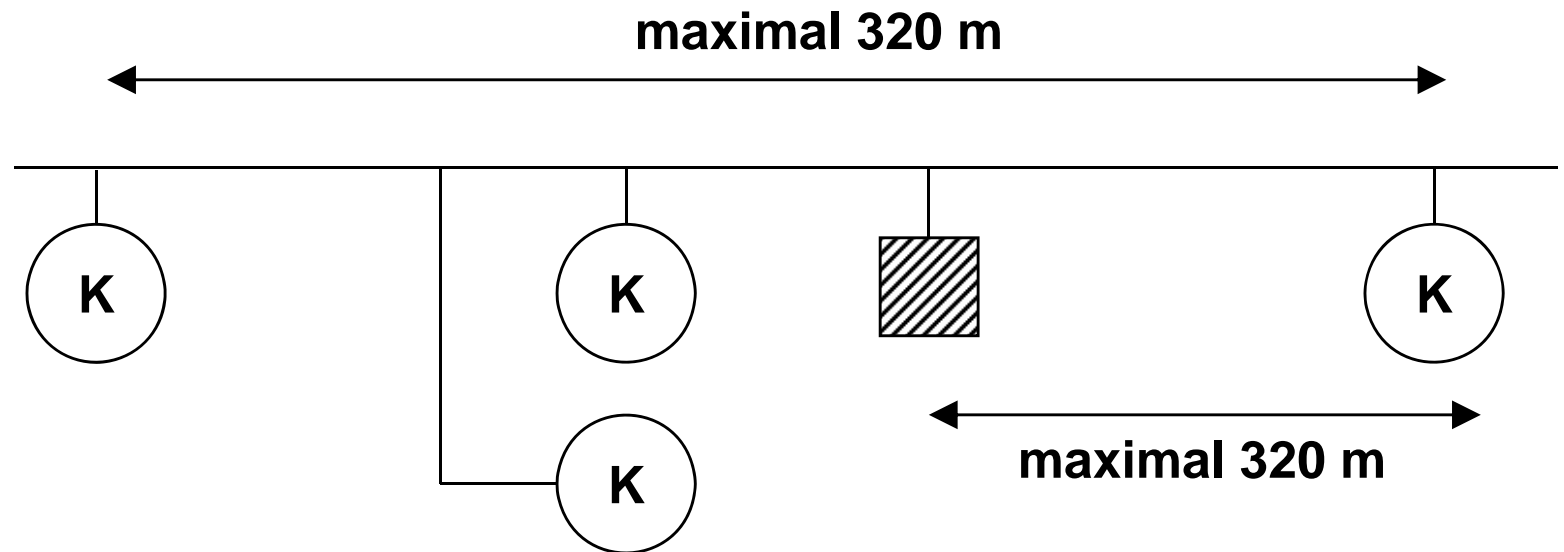


# Freie Netztopologie: bei eingebetteten CPS weit verbreitet



## FTT10-A / LPT-10A in freier Topologie mit JY(St)Y 2×2×0,8

|                                                                               |            |
|-------------------------------------------------------------------------------|------------|
| Abstand zwischen beliebigen Knoten:                                           | max. 320 m |
| Abstand zwischen beliebigen Knoten<br>und Busabschluß bzw. Link-Power-Supply: | max. 320 m |
| Gesamtleitungslänge:                                                          | max. 500 m |

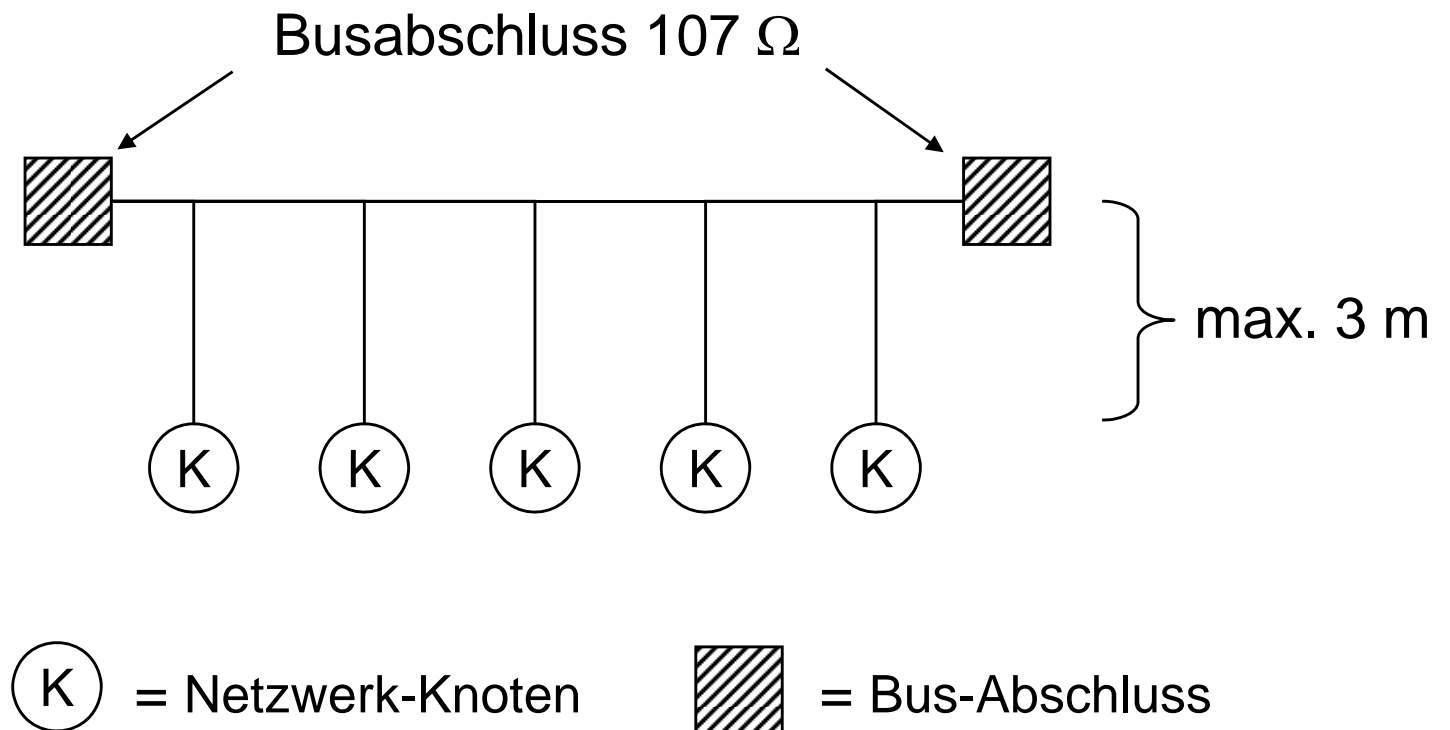


⊙ K = Netzwerk-Knoten

▨ = Bus-Abschluss

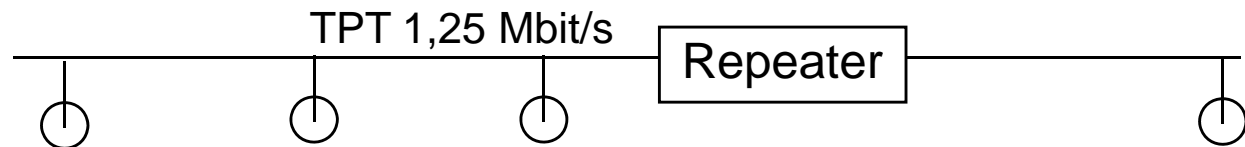
## FTT / LPT in Bus- / Linientopologie

- Große Leitungslänge
- Zwei Busabschlüsse an den Leitungsenden erforderlich
- Länge der Stichleitungen = maximal 3 m
- keine beliebige Verzweigung zulässig
- daher schwer erweiterbar, insbesondere bei Unterputz-Installation



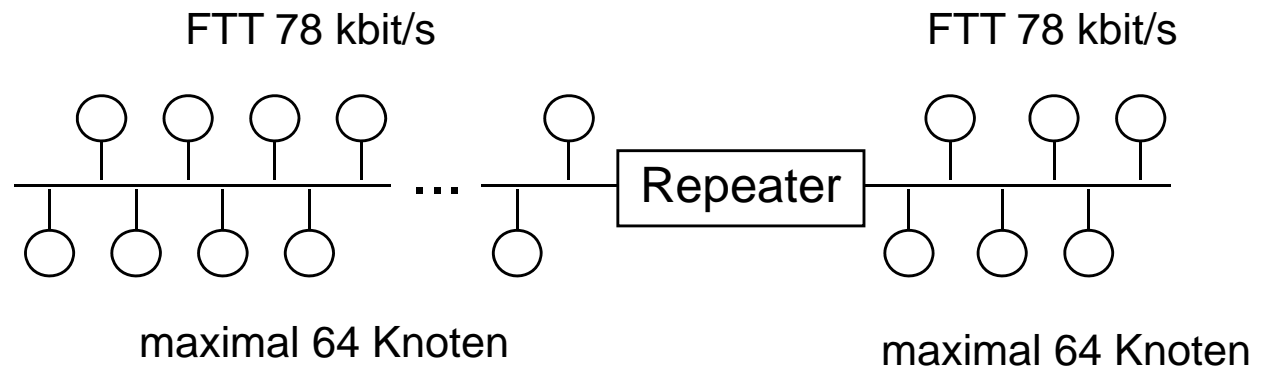
# Repeater als Verstärker

Erhöht die maximale Netzausdehnung



□ Repeater

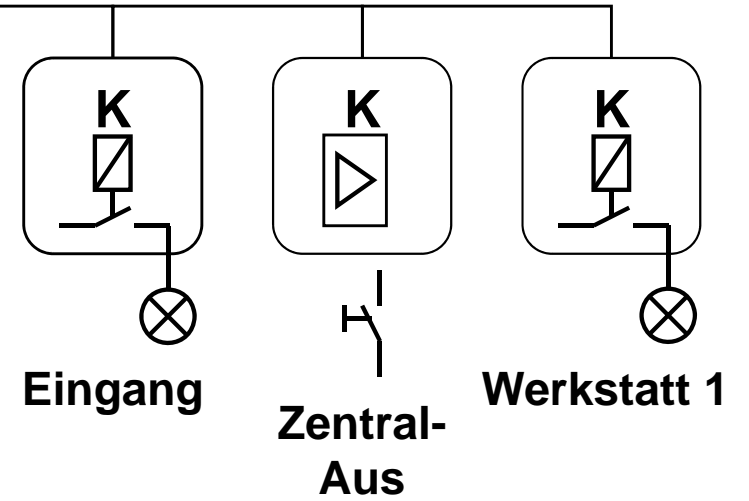
Erhöht die maximale Knotenanzahl



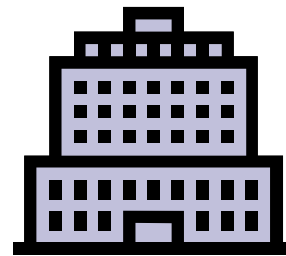


**Neubau**  
Twisted-Pair FTT

*Bus-Beleuchtungssteuerung  
über Zweidrahtleitung (Twisted-Pair)*

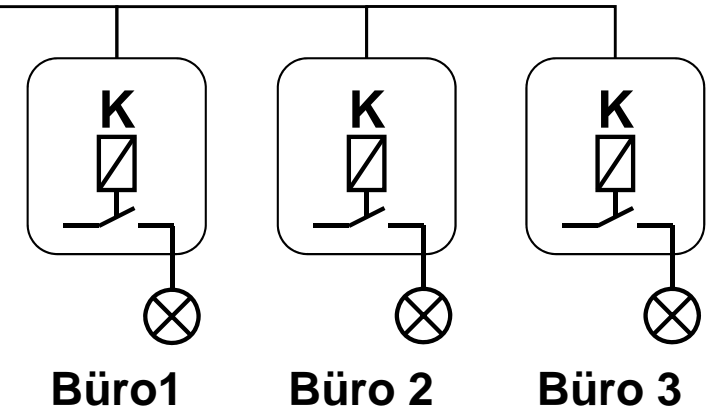


**Router**



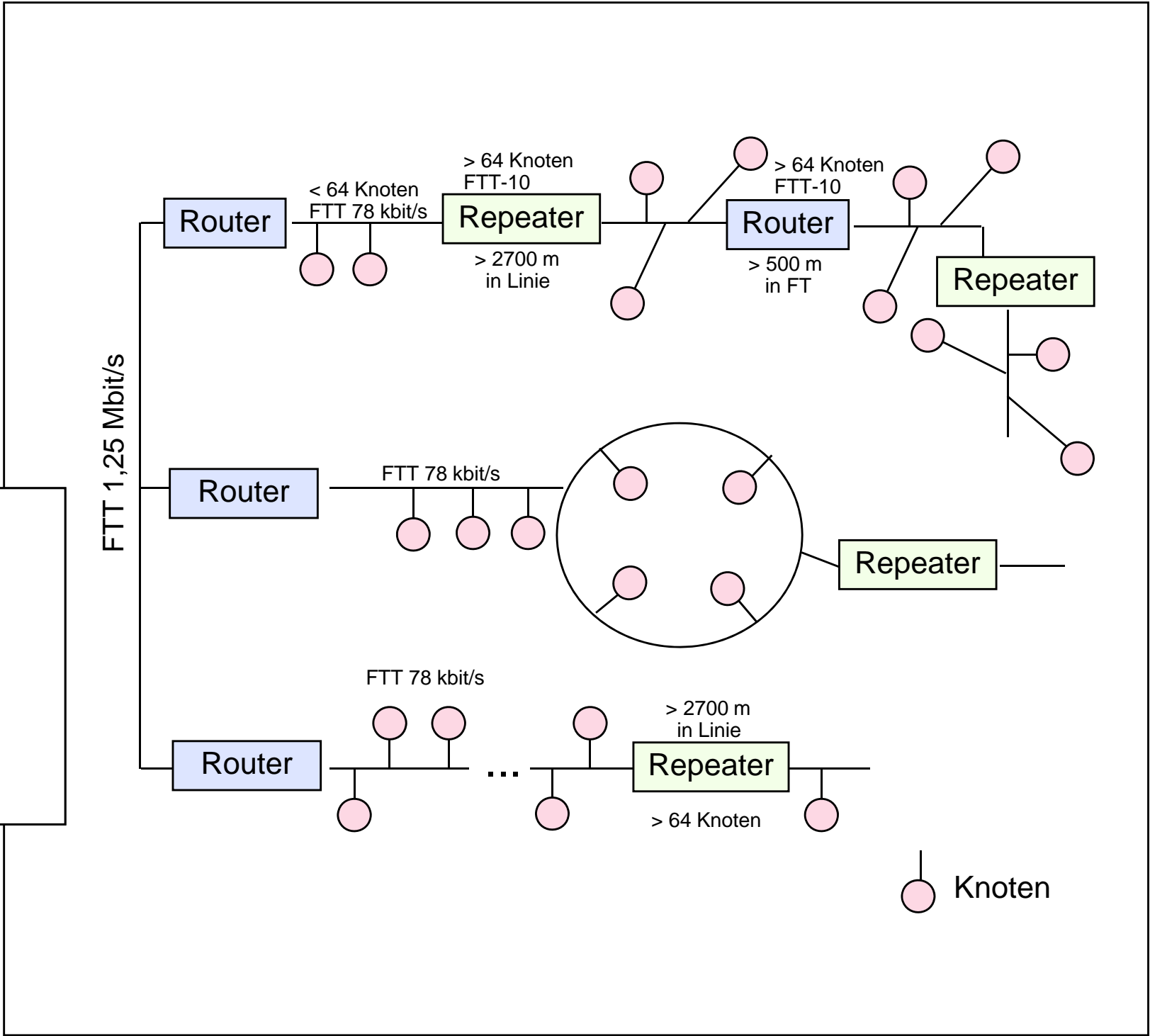
**Altbau**  
Power-Line PLT

*Bus-Beleuchtungssteuerung  
über 230-V-Netzleitung (Power-Line)*

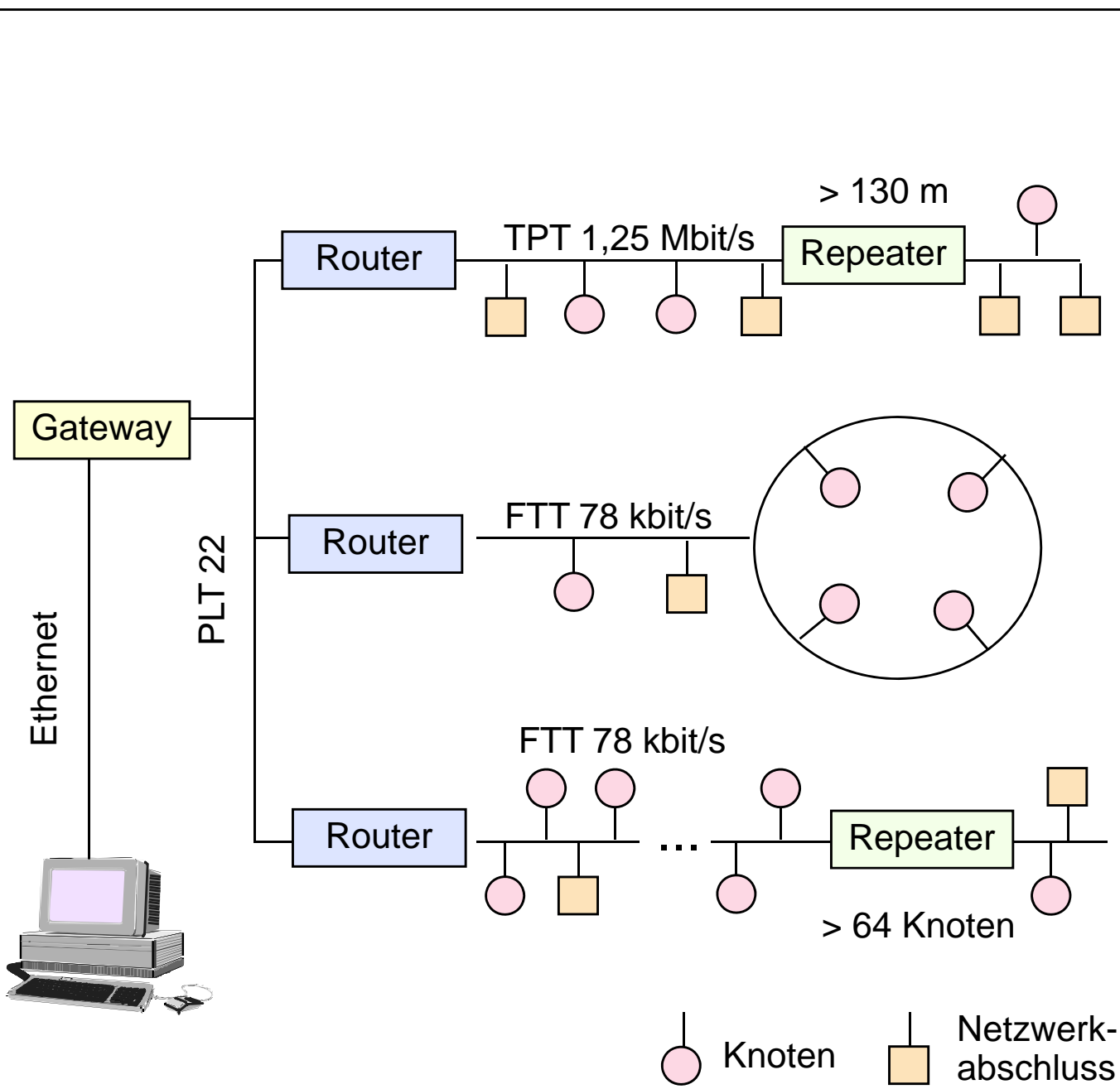


Anwendungs-  
beispiel:  
**Router als  
Koppelement**

□ **Infrastruktur großer Netze**

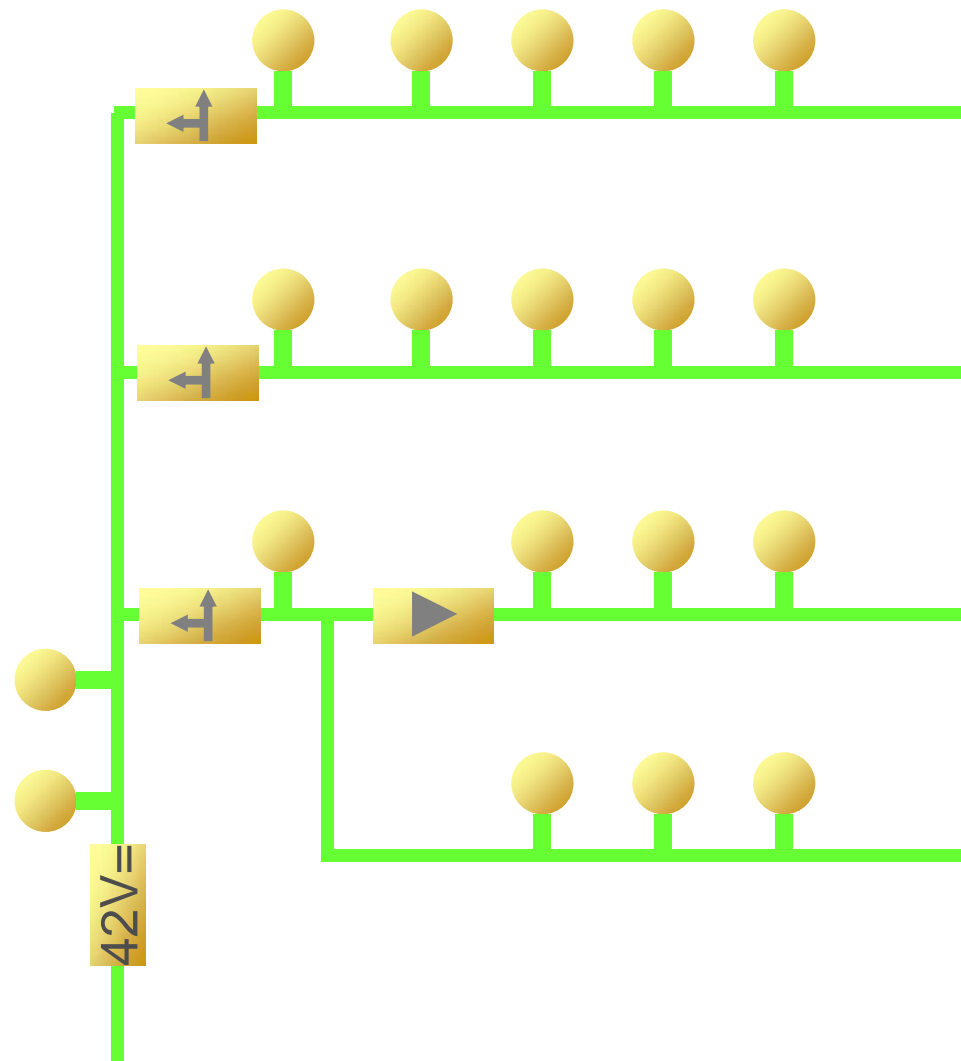


□ **Infrastruktur großer Netze**



# Topologie: Backbone

- Prinzipiell hierarchiefrei
- Senkrechter Channel (“Backbone”):
  - Allgemeine Funktionen wie PC oder Uhr
  - Spannungsversorgung, jedoch mind. Busabschluß
- Router trennt logische Einheiten;  
z.B. 1 Channel / Etage



## **Backbone: Wo gehört was hin?**

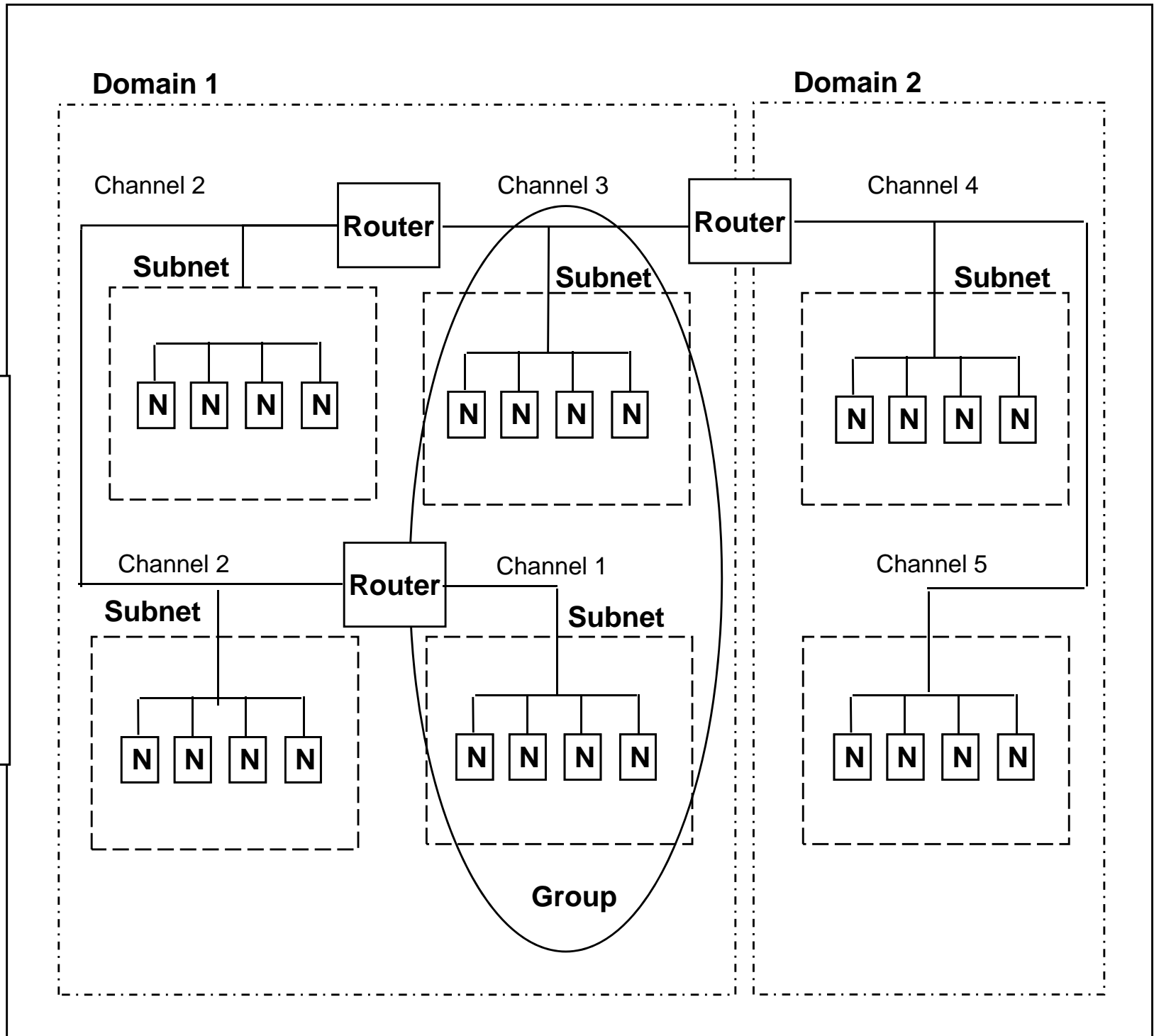
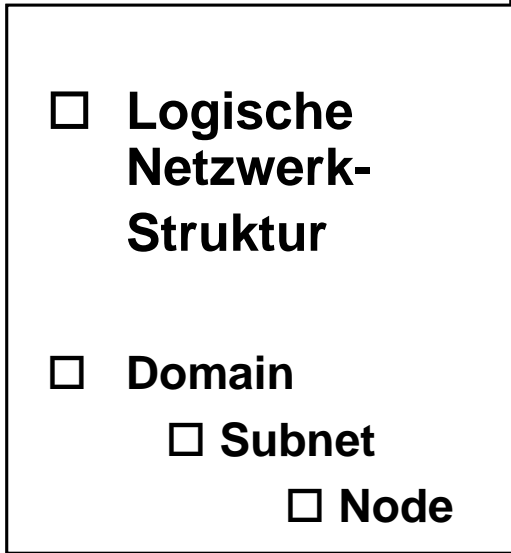
### **□ Je zentraler die Funktion eines Nodes, desto zentraler ist er auch anzuschließen:**

- Die Systemuhr mit dem Zeitsignal für alle Nodes gehört auf den Backbone
- Die Zutrittserfassung für die Etage gehört auf den Etagenbus

### **□ Allgemeiner formuliert:**

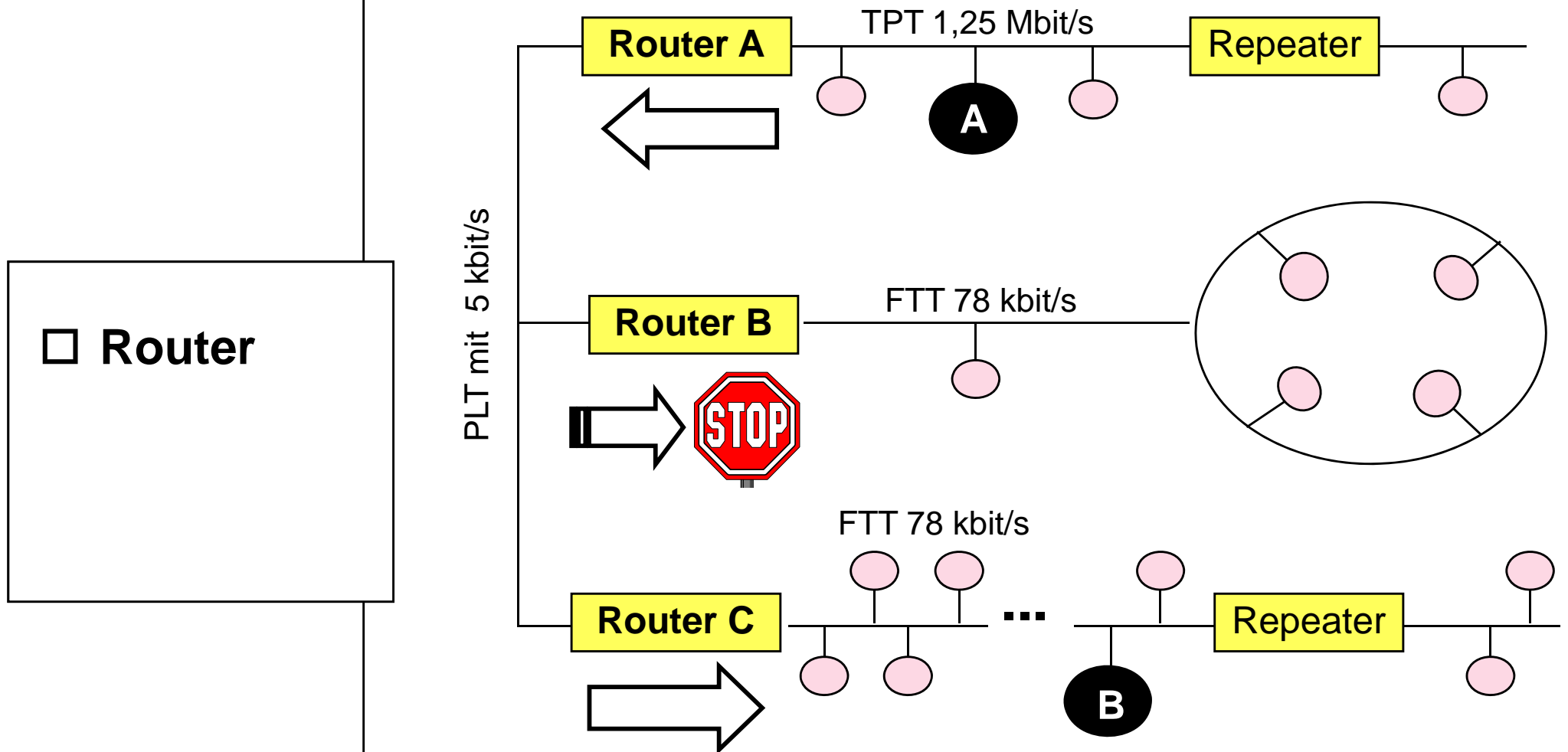
- Signale sollen möglichst wenige Router passieren
- Zu allen Empfängern sollen die Signale möglichst gleich viele Router passieren

# **Logische Netzwerk- Strukturierung**



# Router mit Filterfunktionen

übernimmt Adress-Filterfunktionen und schafft somit eine Buslastentkopplung

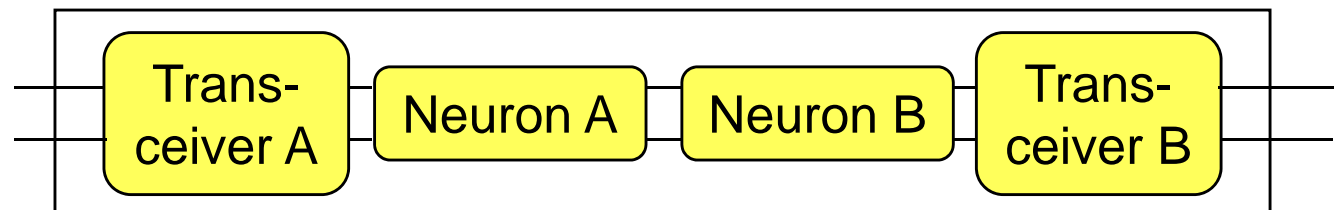


# Routertypen

- **Vorkonfigurierte (*Configured*) Router**
  - filtern Telegramme anhand der Subnet-ID
  - werden vom Inbetriebnahme-Tool mit einer Filtertabelle versorgt, die das Tool auf Basis der eingetragenen Projektdaten selbsttätig erstellt
- **Selbstlernende (*Learning*) Routers**
  - filtern Telegramme anhand der Subnet-ID
  - lassen nach Reset alle Telegramme passieren
  - erstellen selbsttätig ihre Filtertabelle durch Beobachtung des Telegrammverkehrs auf beiden Seiten
- **Bridges**
  - filtern auf Basis der **Domain-ID** im Telegramm
  - bilden die „Brücke“ zwischen zwei Domains
  - verhalten sich wie Repeater in Netzen mit nur einer Domain (keine Filterung)

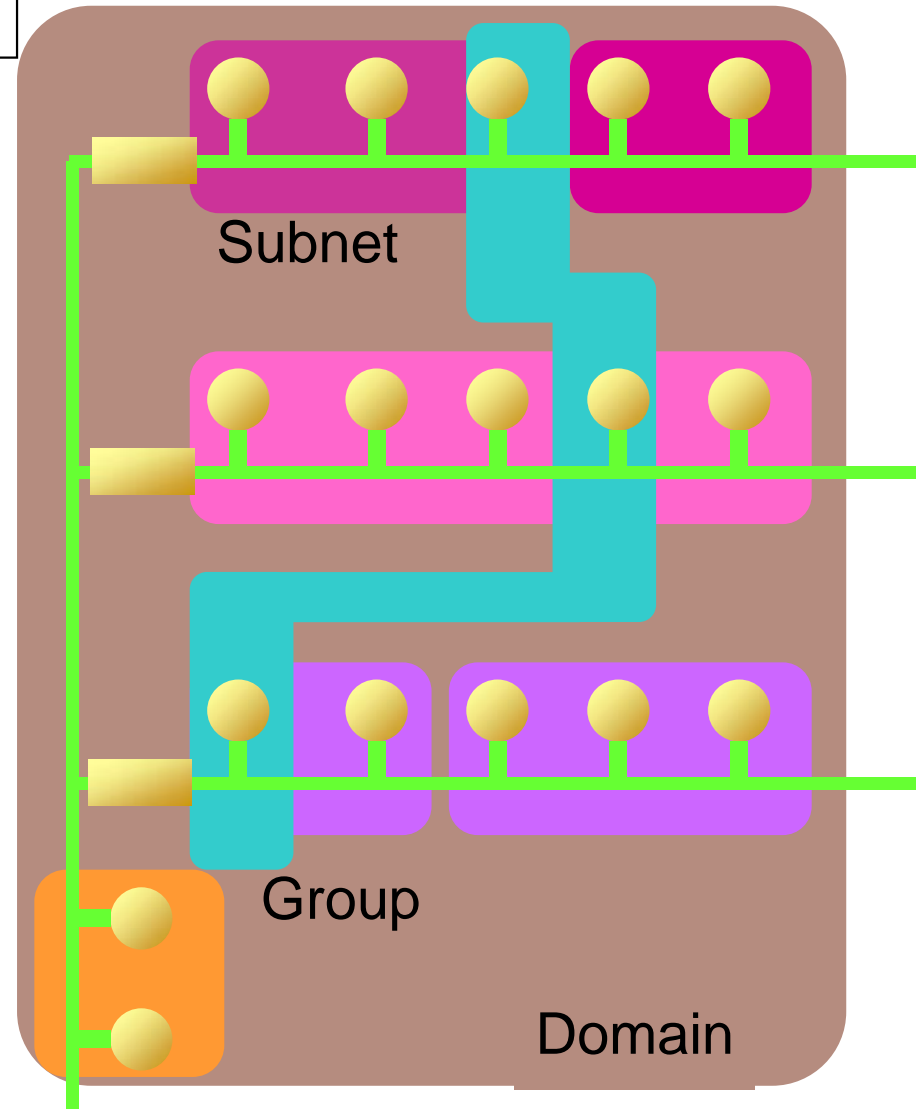
## □ Router-Typen

### Routeraufbau:



# Logische Adressierung

- Nodes  
*Einwohner*
- Subnet  
*Zustellbezirk*
- Domain  
*Stadt*
- Group  
*Verein*

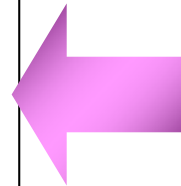


# Eindeutige Adresse

- Physikalisch:  
Node-ID bzw. MAC

- Logisch:  
Subnet + Node  
8 bit + 7 bit

- Group + Member  
8 bit + 6 bit



Meistverwendet;  
Automatische Vergabe  
möglich

## Anwendung von Subnet & Group

### □ Subnet

- Alle Nodes eines Büroflures, einer Etage etc.
- Räumlich zusammen gehörende Nodes

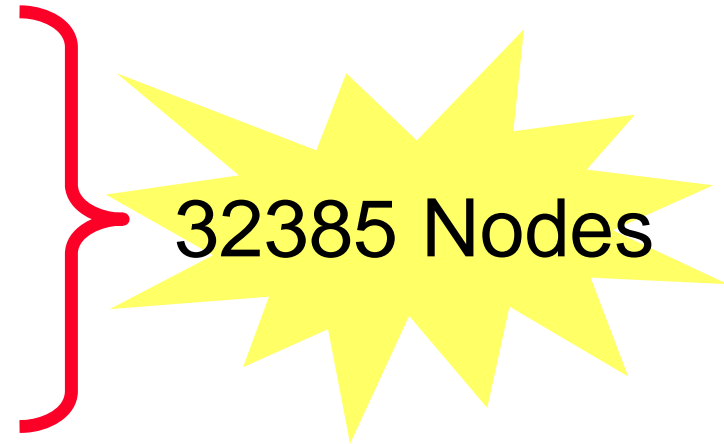
### □ Group

- Alle Rauchmelder im Gebäude
- Alle Jalousien auf einer Gebäudeseite
- Verteilte, aber funktional zusammen gehörende Nodes

# Adressierung in Zahlen

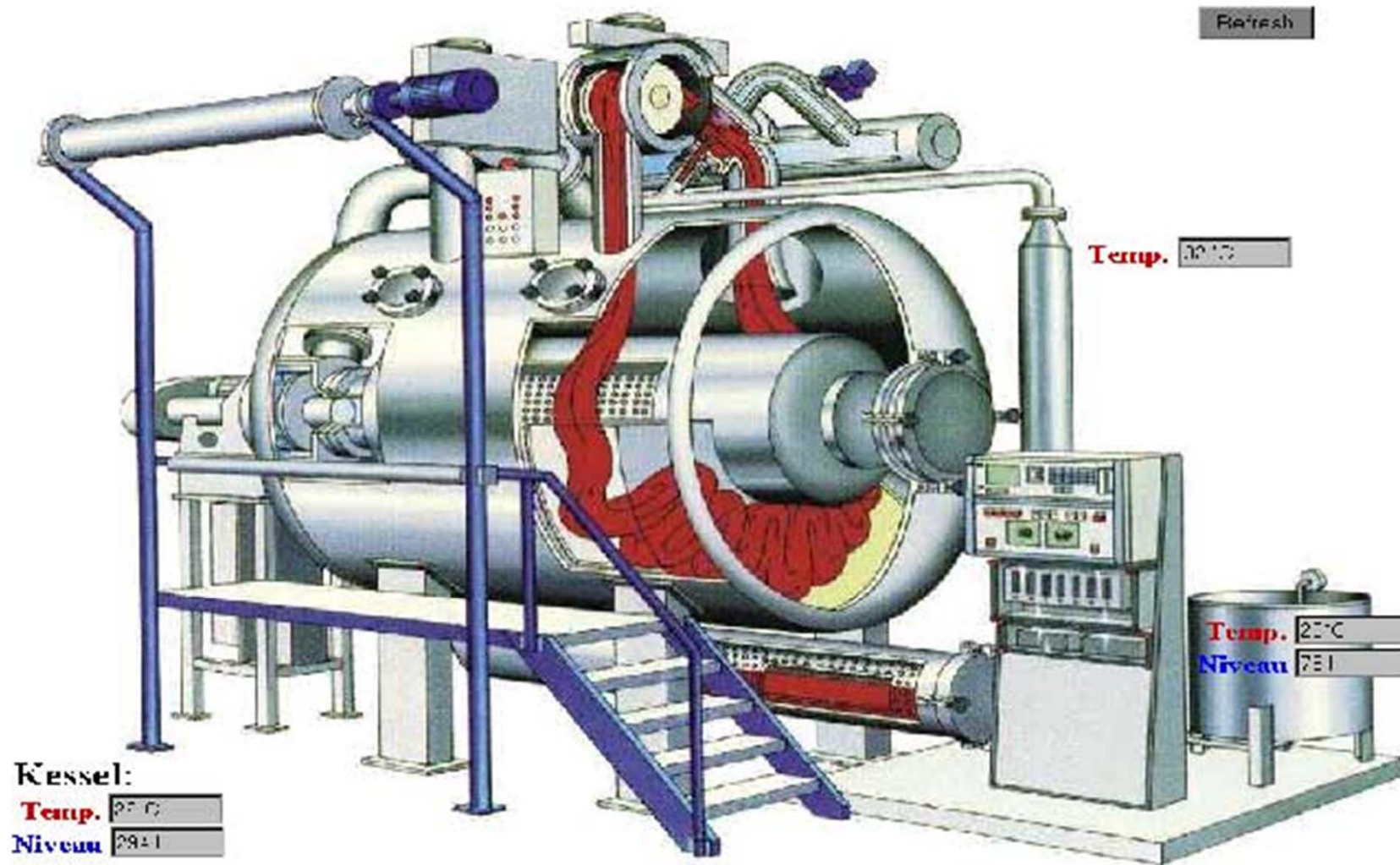
- 255 Subnets je Domain  
*Zustellbezirke je Stadt*
- 127 Nodes je Subnet (egal ob FTT  
oder LPT)  
*Einwohner je Zustellbezirk*
- 256 Groups je Domain  
*Vereine je Stadt*
- 64 Nodes je Group bei bestätigtem Dienst  
*Vereinsmitglieder*
- 15 Groups je Node  
*Einwohner gehört max. 15 Vereinen an*

32385 Nodes

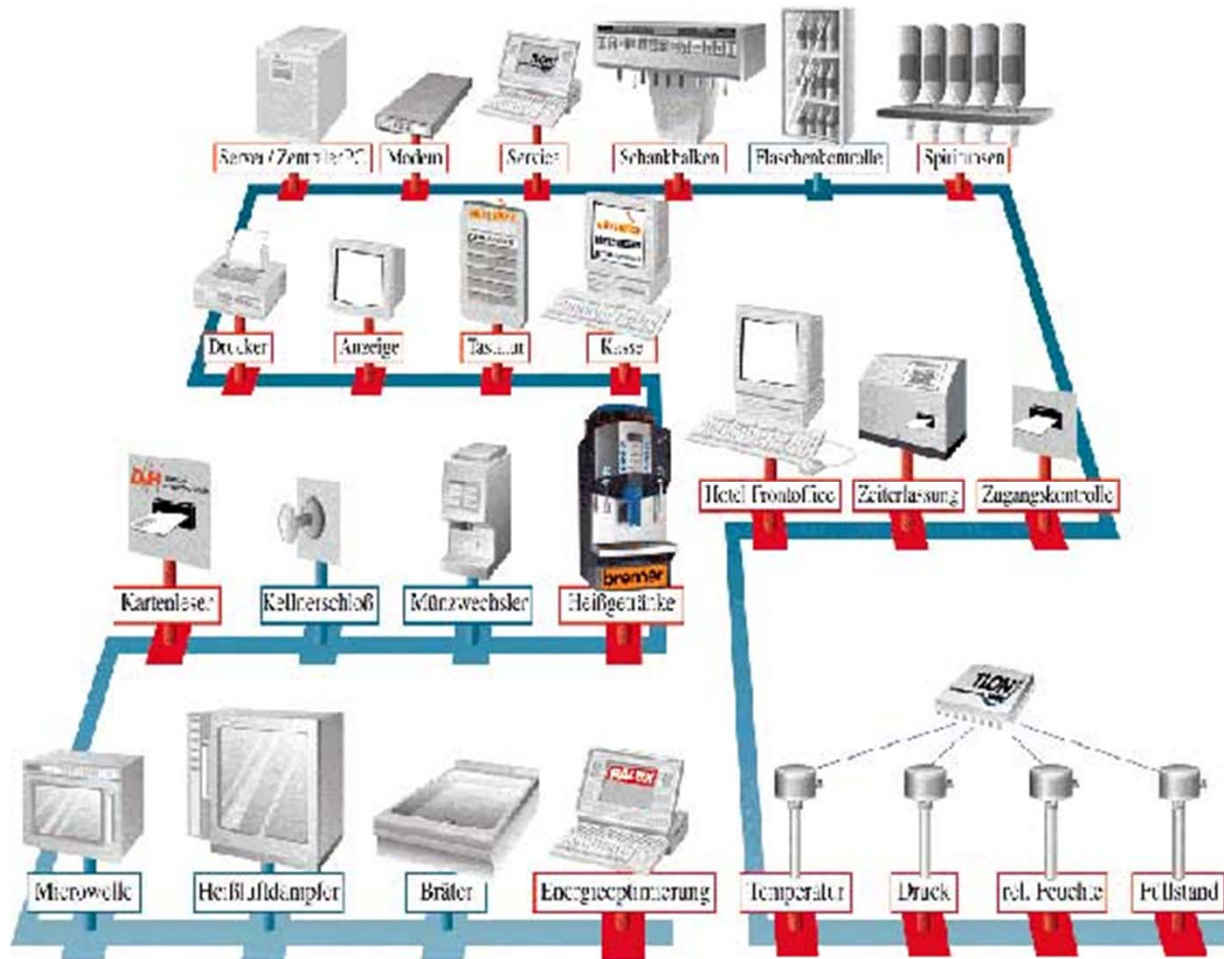


# **Wirtschaftliche Anwendung**

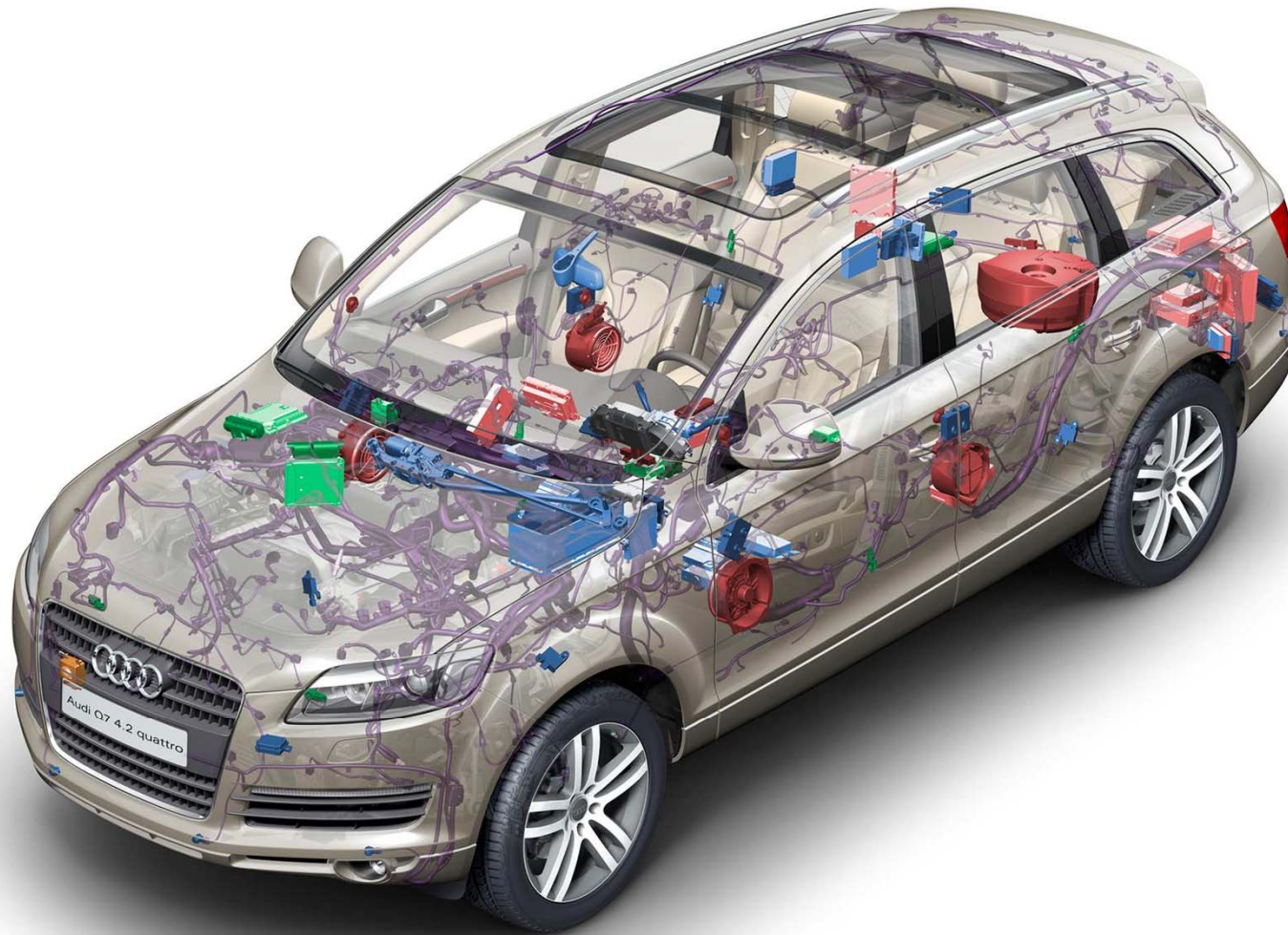
# Maschine



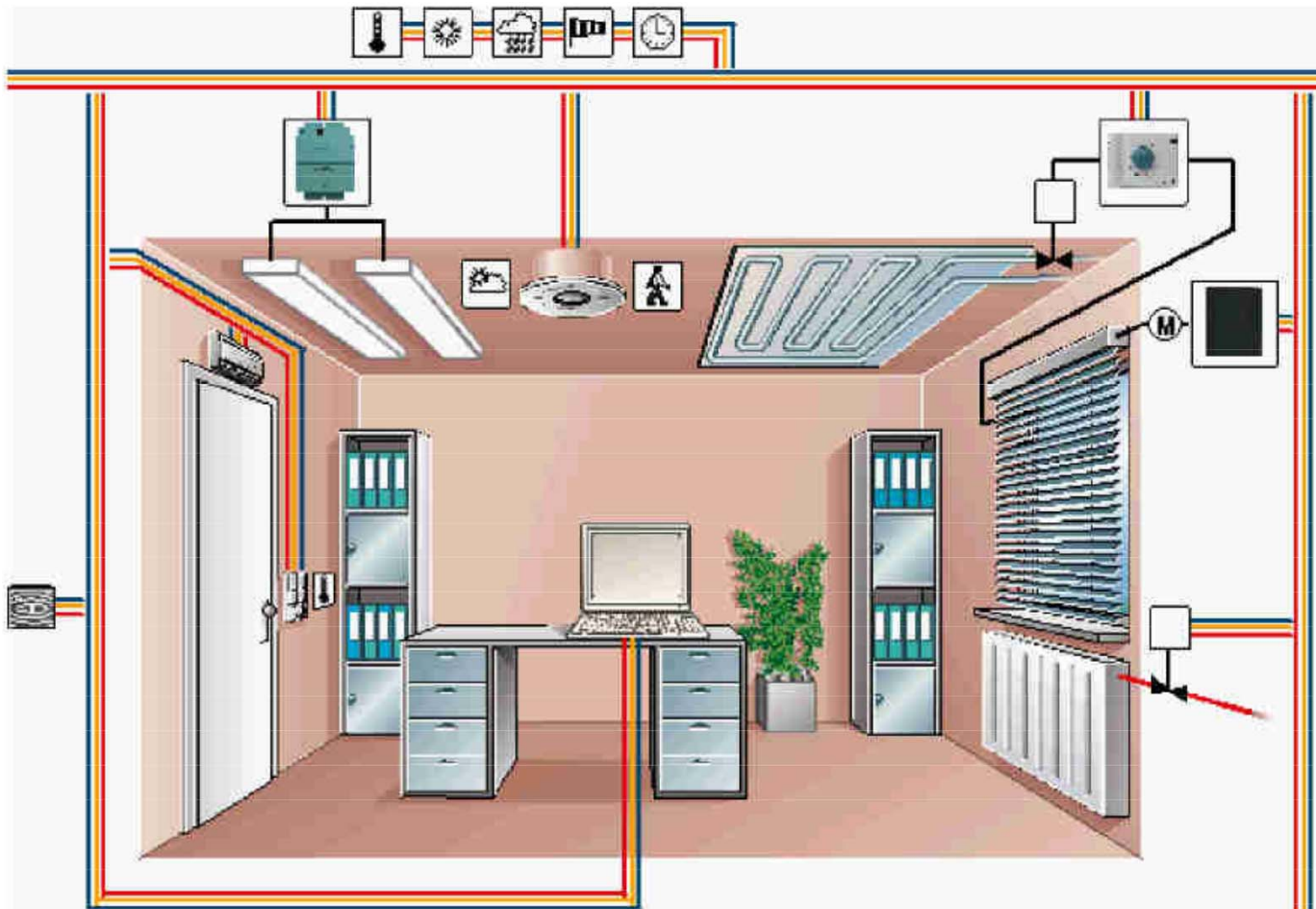
# Restaurant (Systemgastronomie)



# Fahrzeug



# Gebäude



# Standardbüro: Funktionen

## Raumtemperaturregelung

- Frostschutz, Sollwertverst.
- Anwesenheitsgesteuert
- Fensterkontakt

## Licht

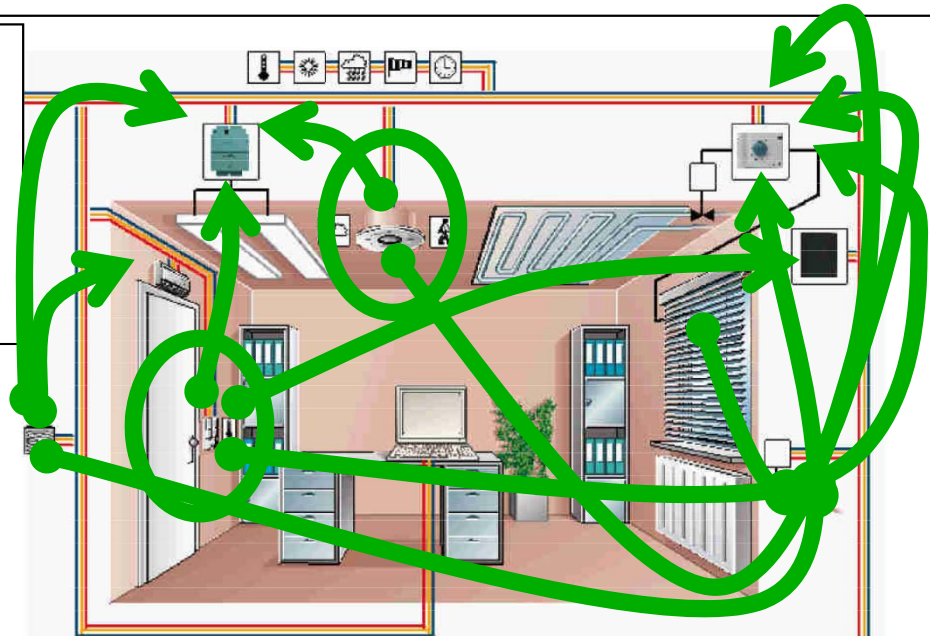
manuell, Sollwert Lichtreg.  
Anwesenheitsgesteuert

## Zutritt

Berührungslose Identifikation,  
Person-, Zeit- und Raumabh.

## Sonnenschutz

Manuell und automatisch



## Gemeinsame Funktionen

- Manuelle Bedienung
- Anwesenheit - Klima - Licht

# CPS-Einsatzgebiete

- Agrartechnik
- Automatisierungstechnik
- Büromaschinen
- Bürosysteme
- Consumer-Elektronik
- Diagnose, Test und Inspektion
- Energietechnik
- Fabrikautomation
- Fahrzeugtechnik
- Förder- und Lagertechnik
- Gebäudeautomation
- Heimautomation
- Identifikationssysteme
- Lichttechnik
- Luftfahrt/Raumfahrt
- Maschinenbau
- Medizintechnik
- Prozess-Steuerung
- Rechner/Peripherie
- Restaurant-Automation
- Robotertechnik
- Sicherheitstechnik
- Steuerungstechnik
- Telekommunikation
- Umwelttechnik
- Verkehrstechnik
- ...

# CPS-Einsatz

Wozu brauchen wir die neuen  
Anwendungen ?

Welche Vorteile bieten CPS bei  
etablierten Anwendungen ?

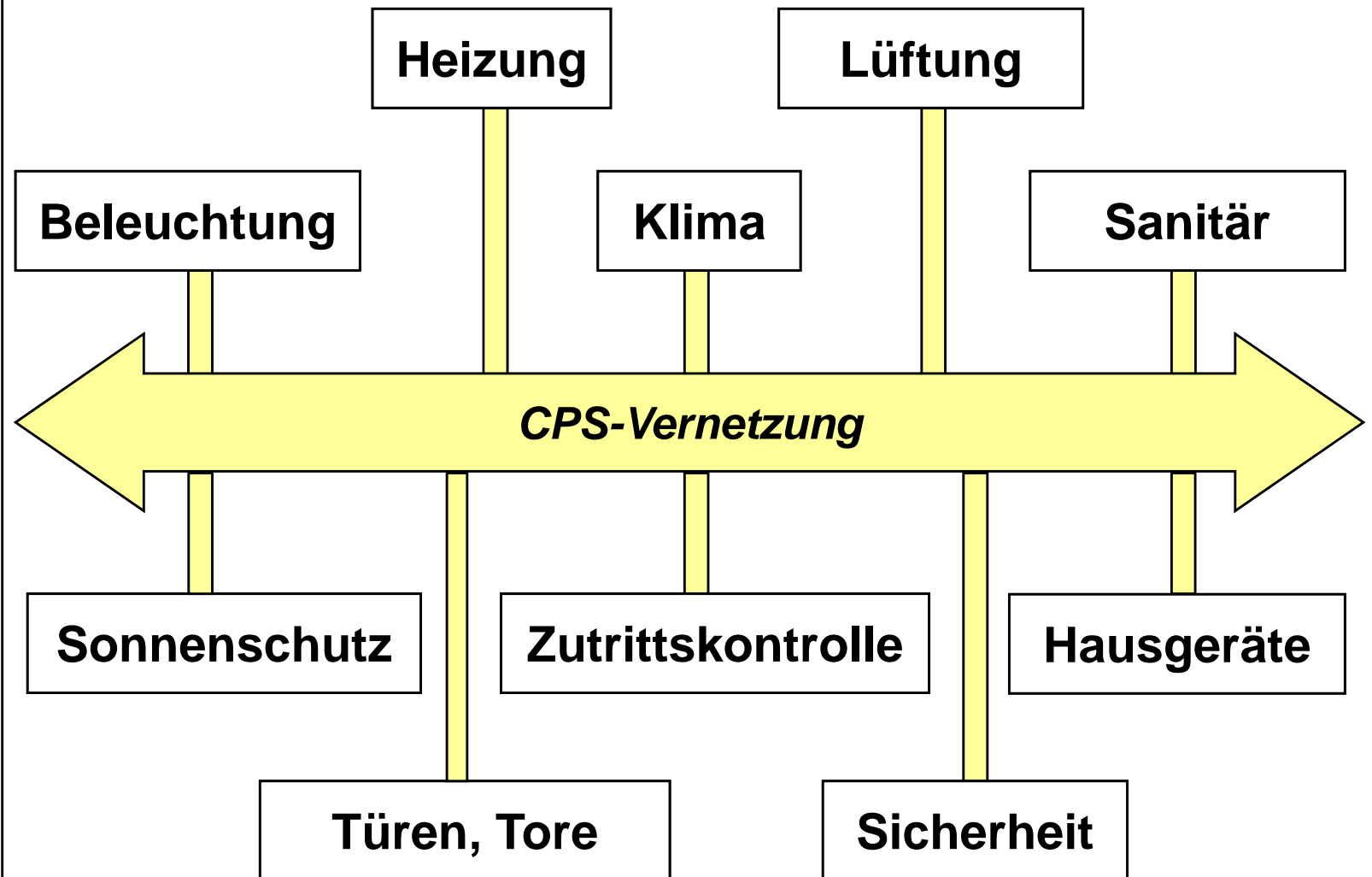
# CPS-Einsatz

Wozu brauchen wir die neuen  
Anwendungen ?

Google Glass ?  
vernetzte Zahnbürste ?

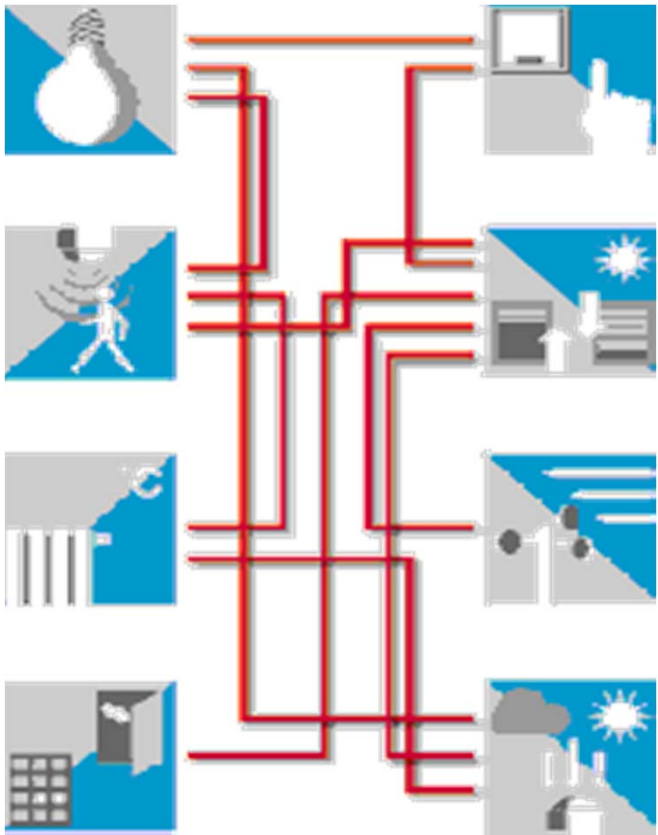
Welche Vorteile bieten CPS bei  
etablierten Anwendungen ?

# Beispiel Gebäude: Integration der Gewerke



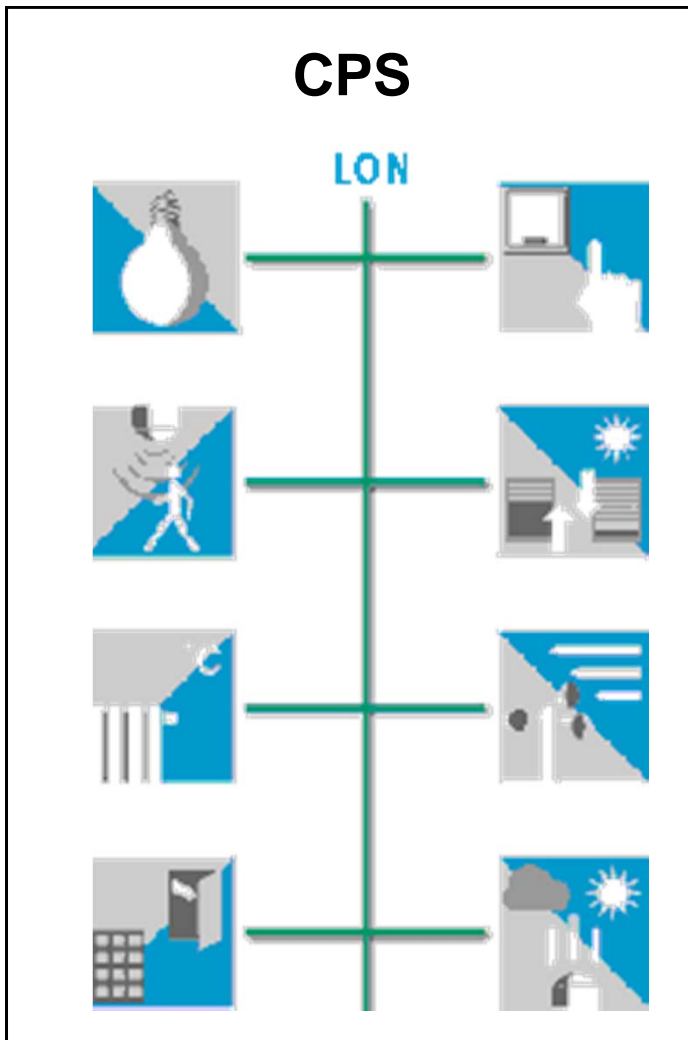
# Integration der Gewerke

## Konventionell



- Diskrete Verkabelung
- Separate Bedienelemente für jedes Gewerk
- Anpassung an Nutzungsänderungen nur durch Umverdrahtung

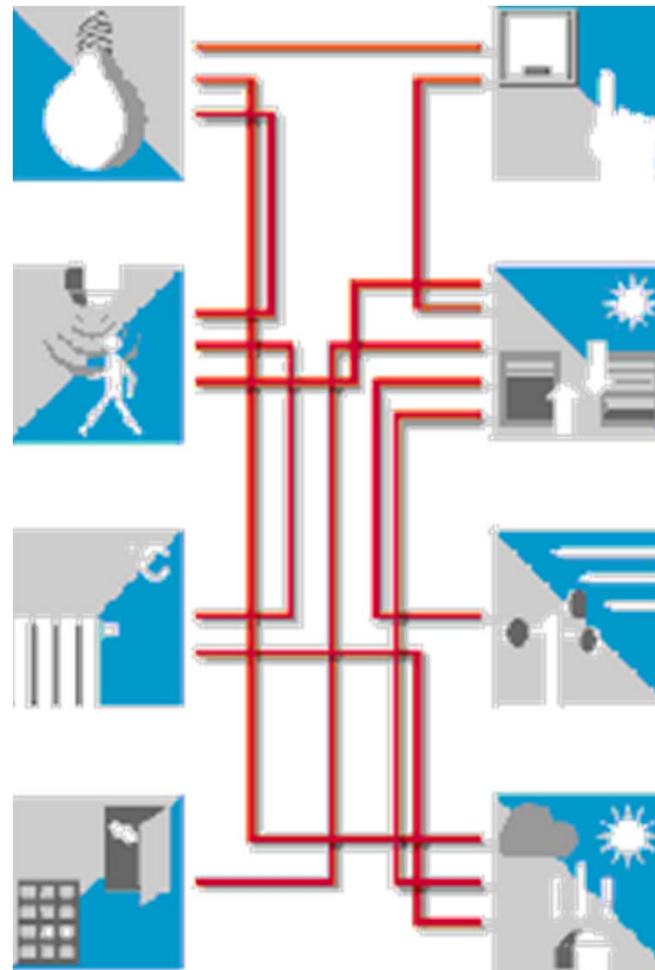
# Integration der Gewerke



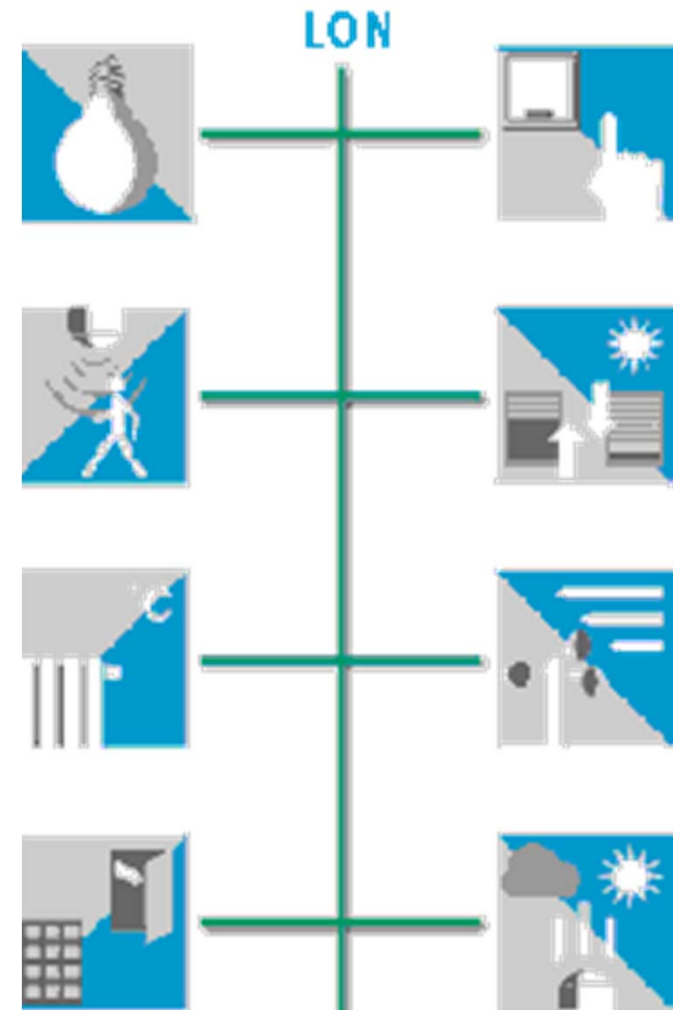
- **Gemeinsames Netzwerk**
  - Reduzierte Verkabelung
  - Geringere Brandlast
- **Gemeinsame Bediengeräte und gemeinsame Sensoren**
  - Einheitliche Bedienphilosophie
  - Reduzierter Geräteaufwand
- **Verknüpfung durch Software**
  - Flexible Anpassung
  - Zukunftssicherheit

# Integration der Gewerke

## Konventionell



## CPS-Vernetzung



# Vorteile und Nutzen für Bauherren und Betreiber:

*Investitionskostenersparnis ...*

*Betriebskostenersparnis ...*

*Komfort ...*

*Einheitliche Bedienung*

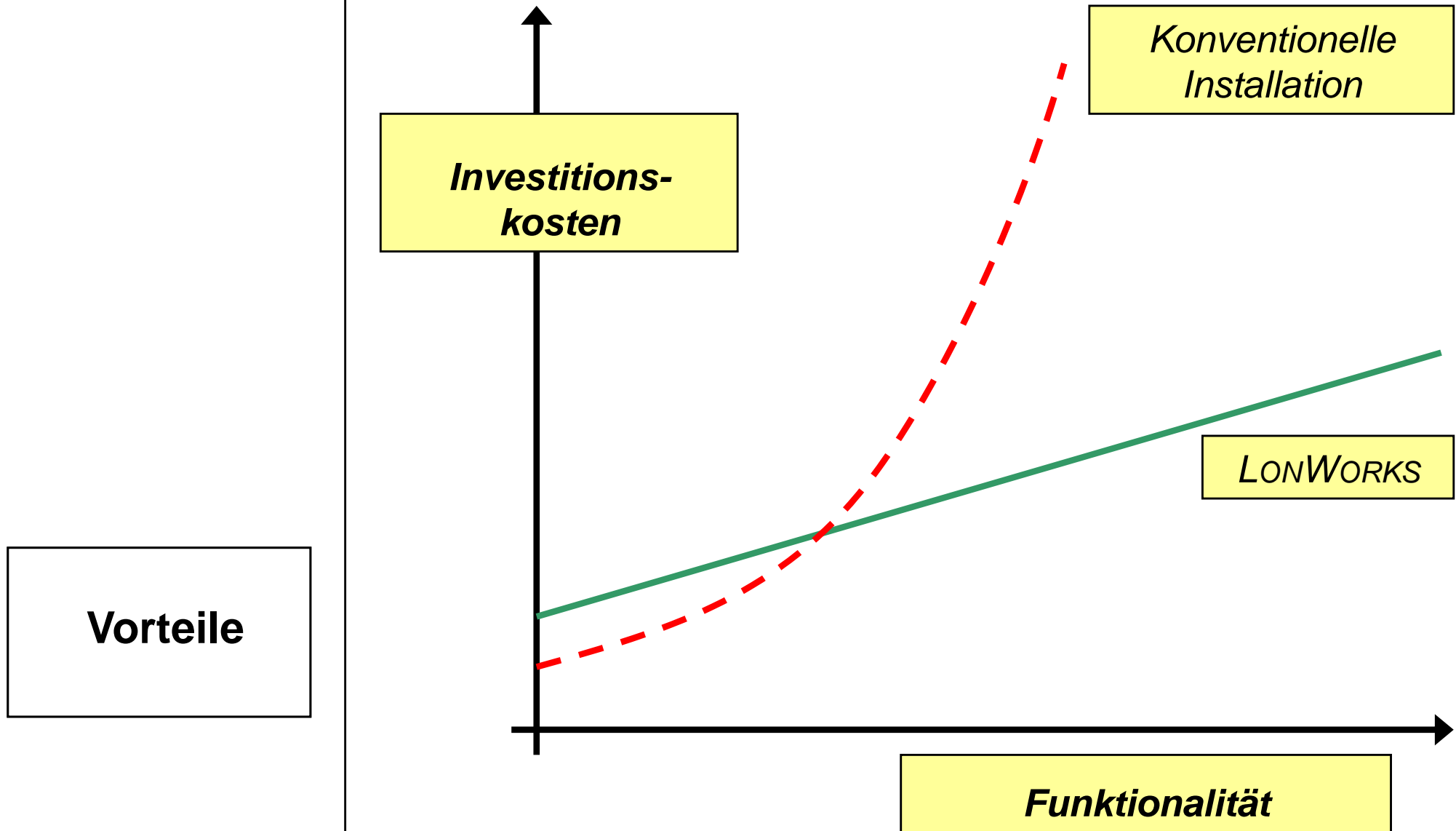
*Flexibilität  
bei Änderungen und Erweiterungen*

*Gebäude-Transparenz  
(durch Visualisierung, Fernüberwachung)*

*Zukunftssicherheit  
weil Hersteller-Übergreifend*

**Vorteile  
(Überblick)**

# ... Investitionskostenersparnis



## ... Investitionskostenersparnis

**Wieso kann man trotz teurerer Komponenten Investitionskosten sparen?**



**Vorteile**

## ... Investitionskostenersparnis

**Wieso kann man trotz teurerer Komponenten Investitionskosten sparen?**

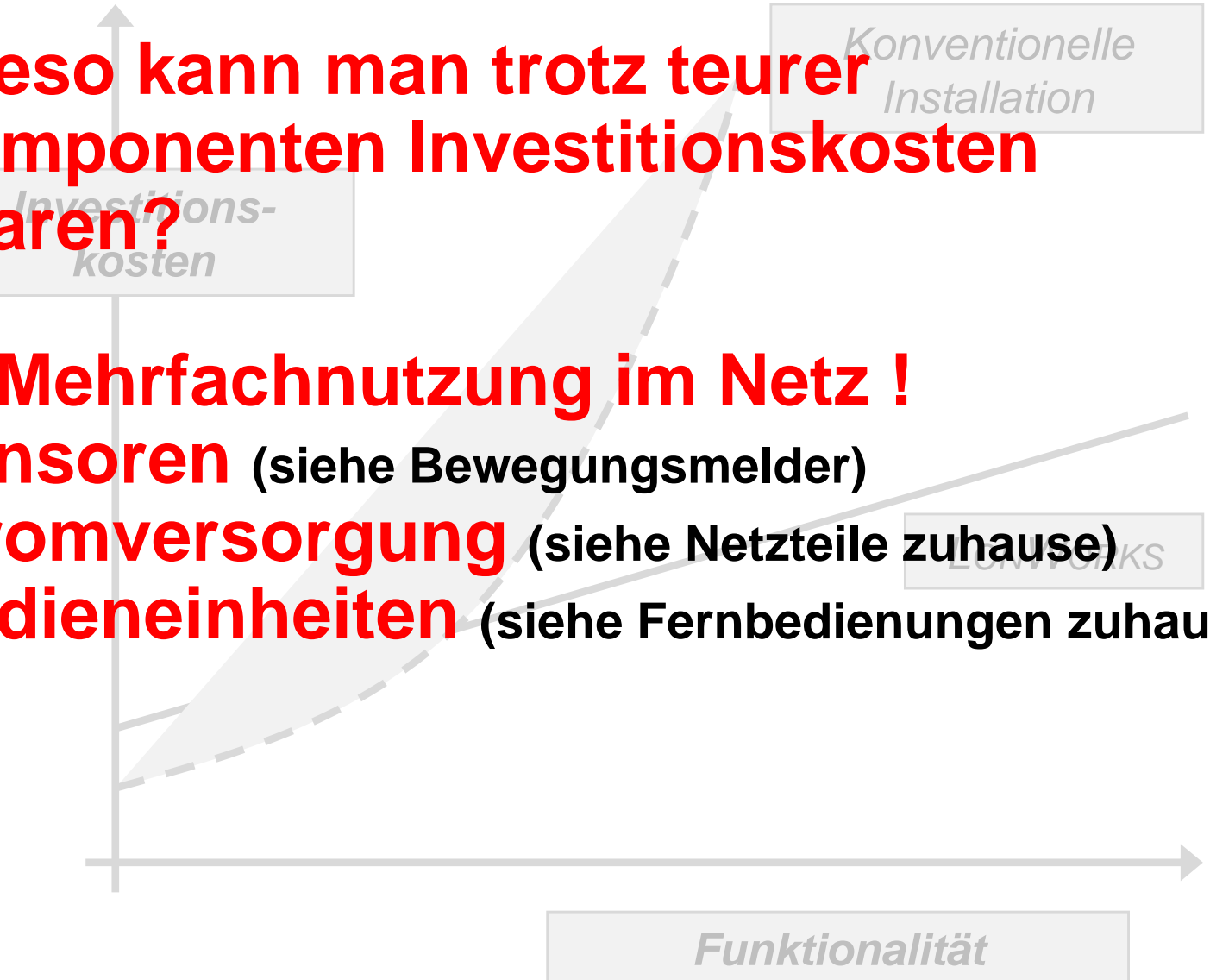
**→ Mehrfachnutzung im Netz !**

**Sensoren** (siehe Bewegungsmelder)

**Stromversorgung** (siehe Netzteile zuhause)

**Bedieneinheiten** (siehe Fernbedienungen zuhause)

Vorteile



# ... Betriebskostensparnis

***Zentralfunktionen, Zeitsteuerung***

***Einzelraumregelung***

***Konstantlichtregelung***

***Nutzungsabhängige Heizungs-- und  
Beleuchtungsregelung***

***Energiemanagement***

***Visualisierung***

***Wartung und Instandhaltung***

***Flexible Anpassung bei Änderung der Raumnutzung***

**Vorteile**

## ... Komfort

*IR-Fernbedienung*

*Zentralfunktionen*

*Zeitprofile*

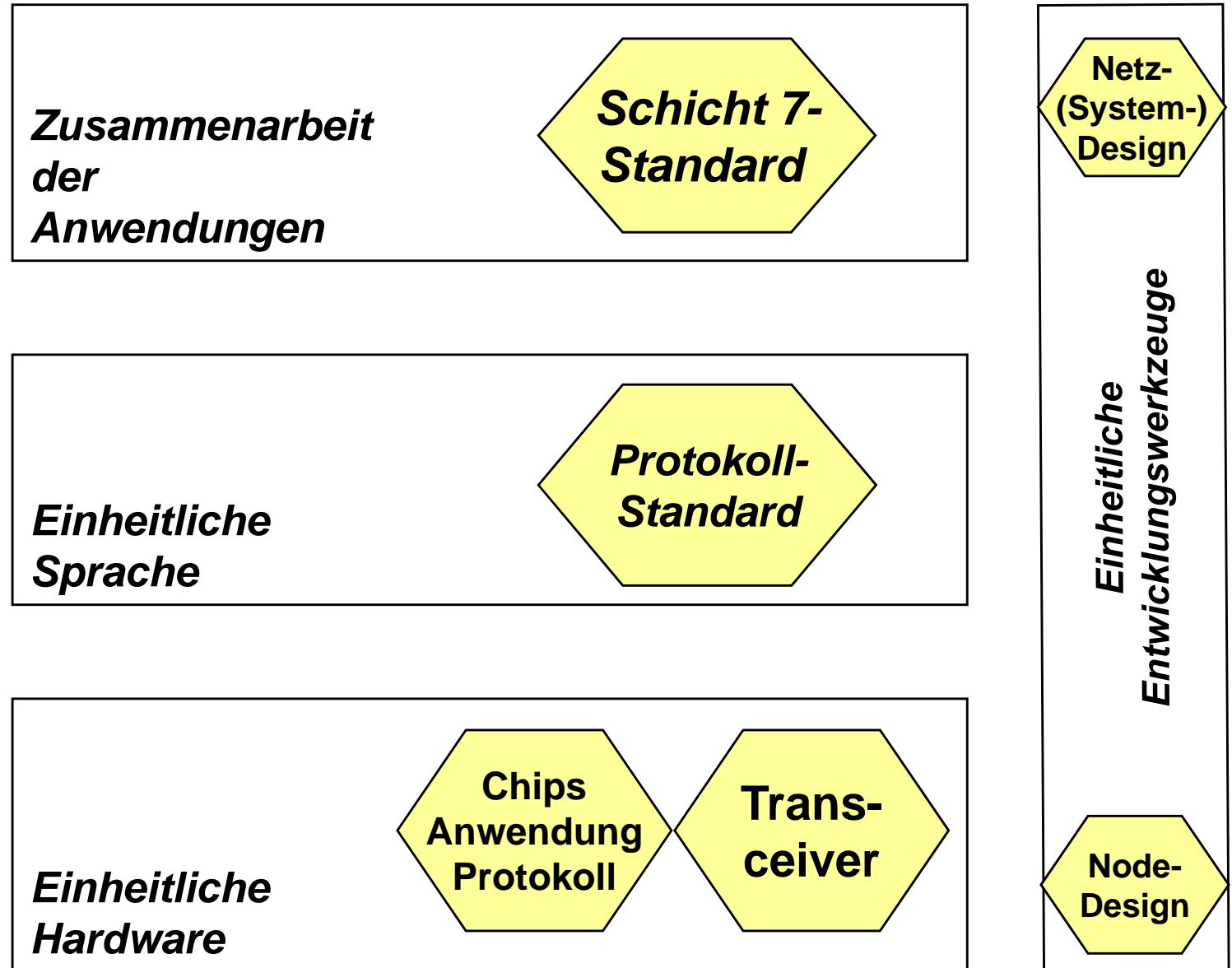
*Fernsteuern, Alarmierung über Telefon*

*Internet-Anbindung*

*USW.*


**Vorteile**

# Bausteine einer CPS-Technologie




# Bausteine der LONWORKS-Technologie

**Zusammenarbeit  
der  
Anwendungen**



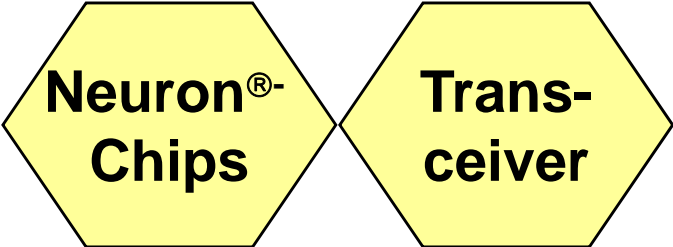
**LonMark®**

**Einheitliche  
Sprache**



**LonTalk®-  
Protokoll**

**Einheitliche  
Hardware**



**Neuron®-  
Chips**      **Trans-  
ceiver**



**LON-  
Builder®**

**Einheitliche  
Entwicklungswerkzeuge**



**Node-  
Builder®**

## **Echelon**

## **Technologiegeber**

**Firmensitz**

**Palo Alto, Californien, USA**

**Gründer**

**Mike Markkula (Intel, Apple), 1986**

**Web-Site**

**<http://www.echelon.com>**

**Vertretungen**

**ca. 200 Mitarbeiter weltweit:**

**USA, China, Frankreich, Deutschland,  
Italien, Japan, Niederlande, Schweden,**

**...**

**Entwicklung**

**250 Millionen US-\$, 2000 Mannjahre,  
67 angemeldete Patente**

**Historie**

**1986 Beginn der Entwicklungsarbeit**

**1990 Vorstellung des Neuron-Chips**

**2013 150 Mio. Neurons verkauft**

**2014 Internet of Things (IzoT™): IPv6,**

**WiFi, 4-Core-ARM-Chips, Linux, C++,**

**C, Python, 128 Bit Security,**

**Technologie-  
geber Echelon**



# **Schicht-7- Kommuni- kation & Software**

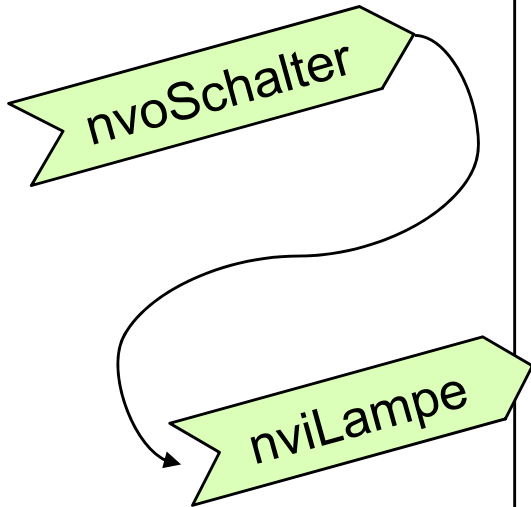
# **Bedeutung der Netzwerkvariablen (NV)**

*Datenaustausch  
zu anderen Knoten*

*Logische Schnittstelle  
des Knotens zum Netzwerk*

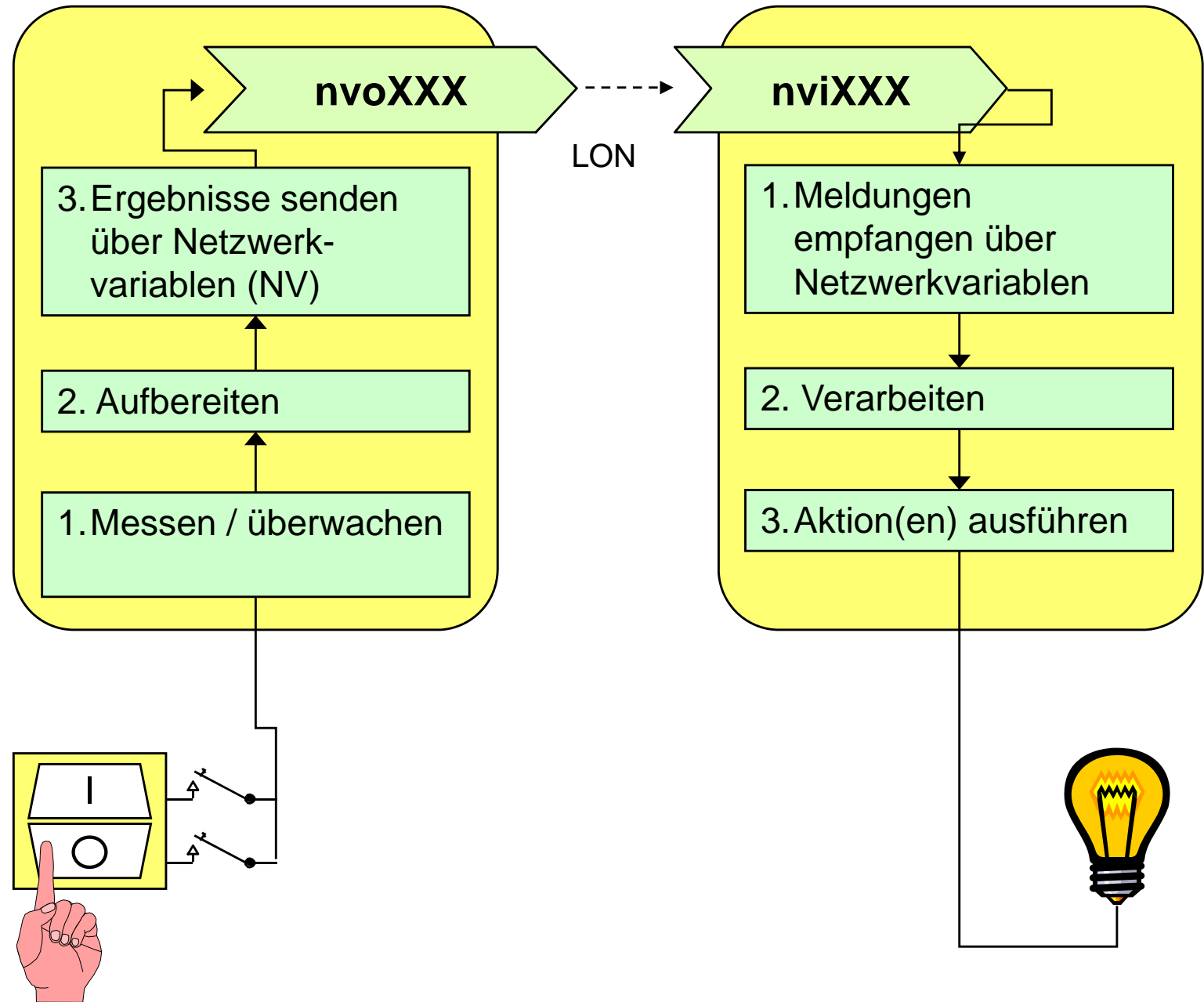
*Herstellen der Gerätebeziehungen  
über sog. „BINDEN“  
von Ein- und Ausgangs-NV (am PC)*

*Äußeres „Erscheinungsbild“  
eines LON Knotens am PC*

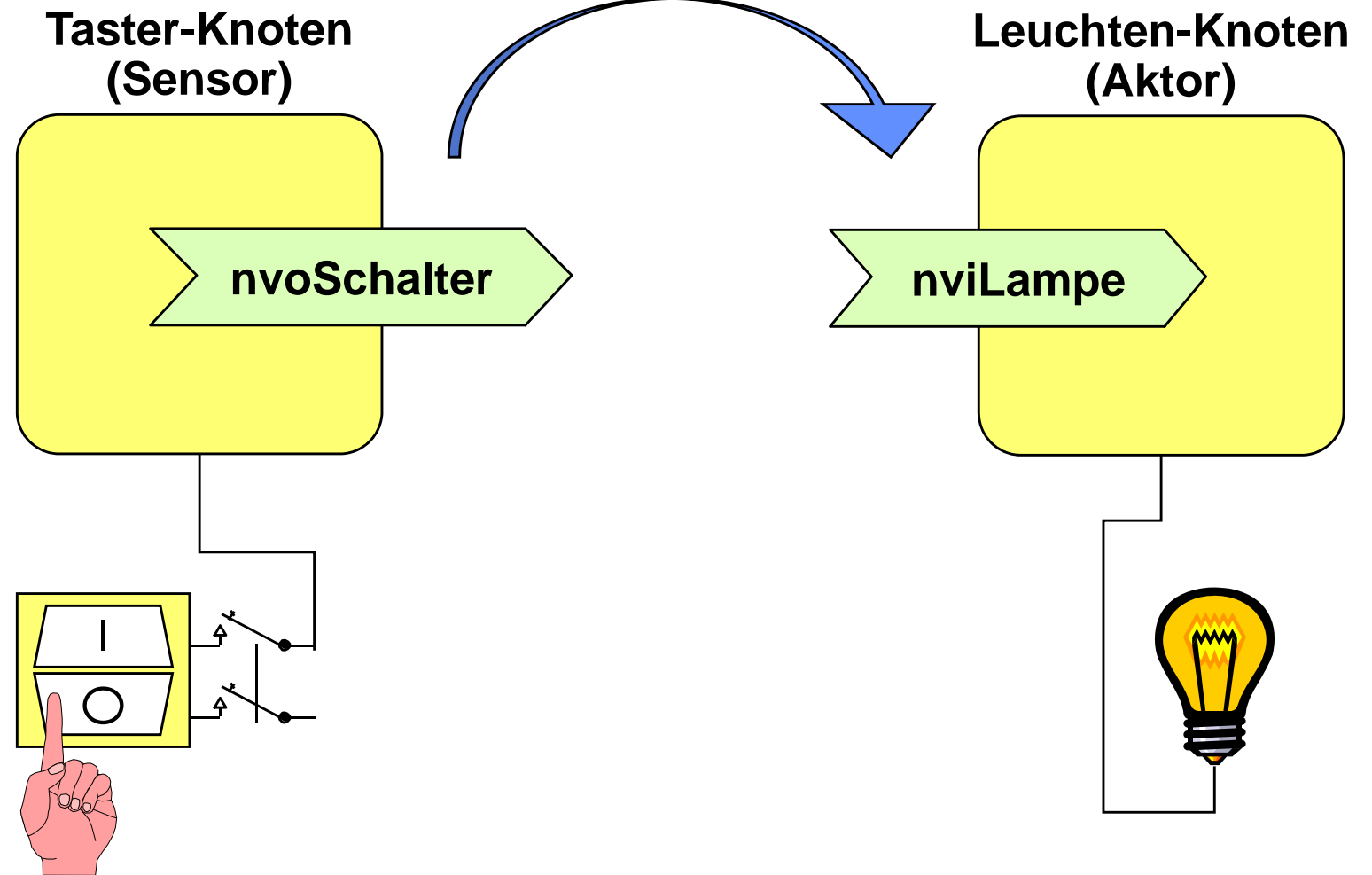


# Sensor

# Aktor

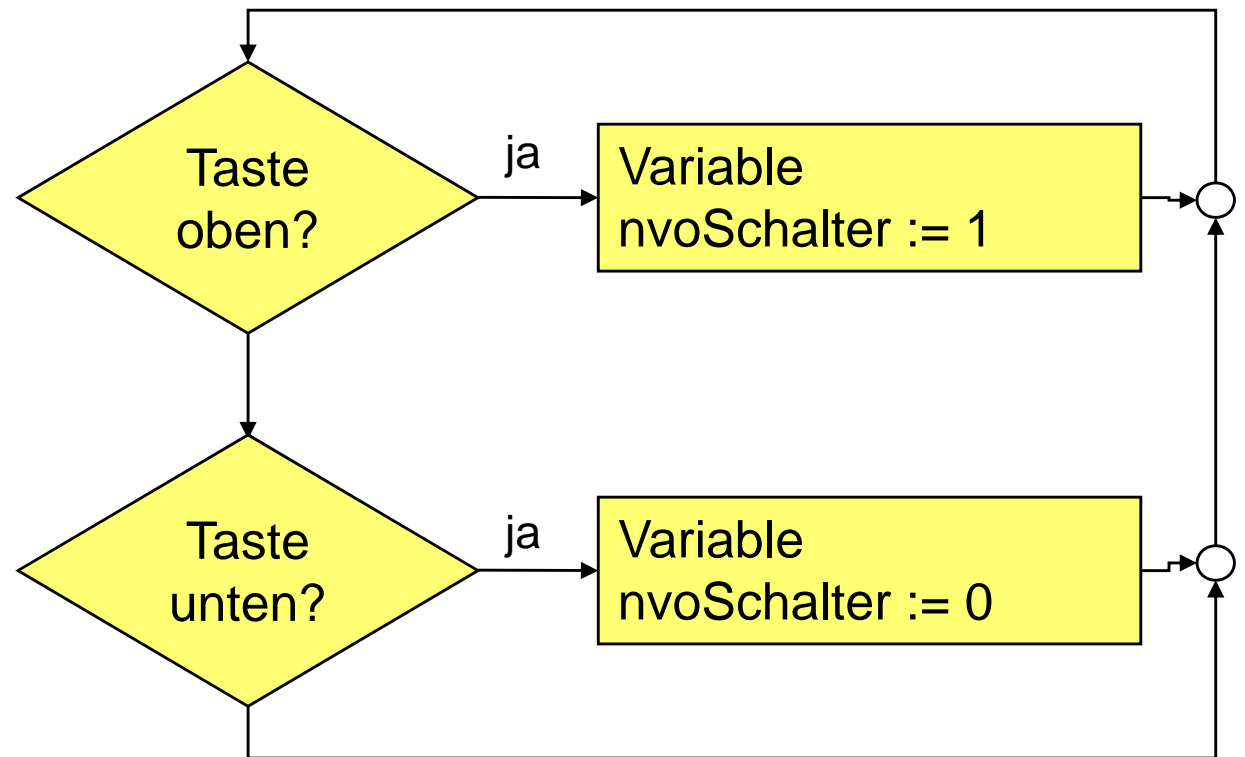
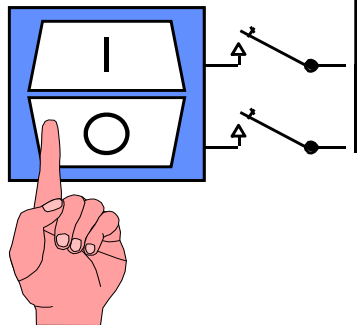
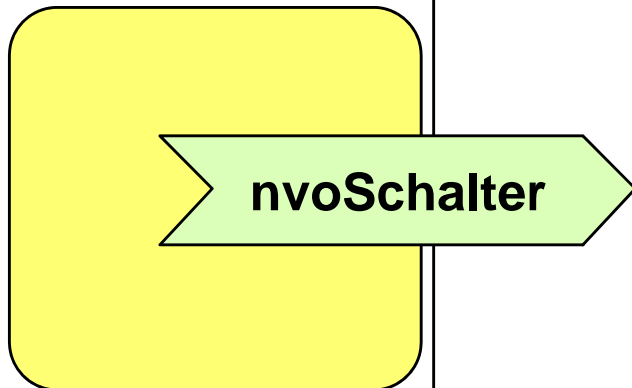


# Kommunikation über Netzwerkvariablen (NV)



# Anwendungsprogramm Taster-Knoten

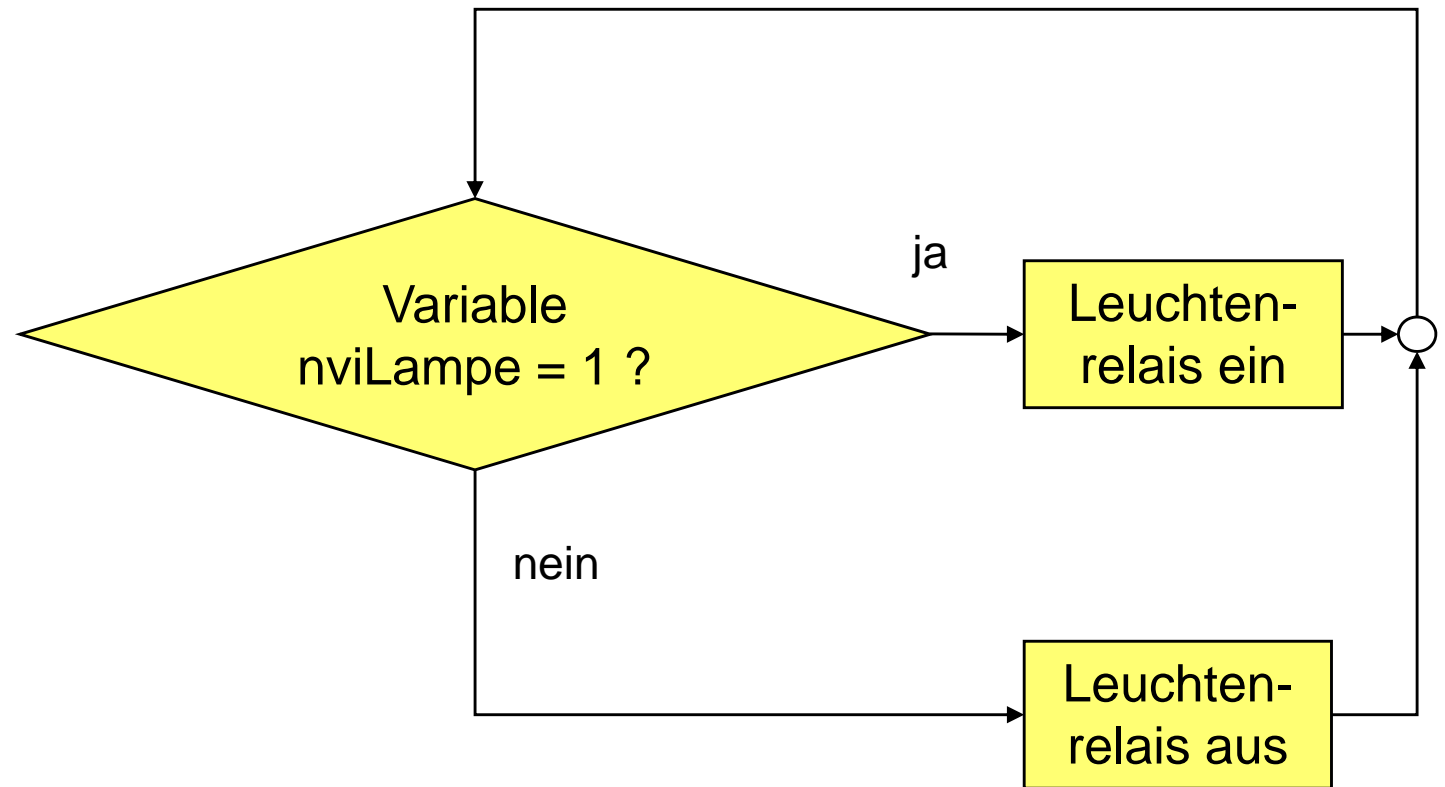
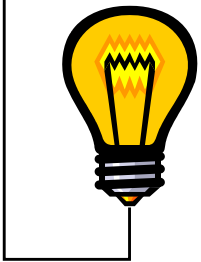
Taster-Knoten  
(Sensor)



# Anwendungsprogramm Leuchten-Knoten

Leuchten-Knoten  
(Aktor)

nviLampe



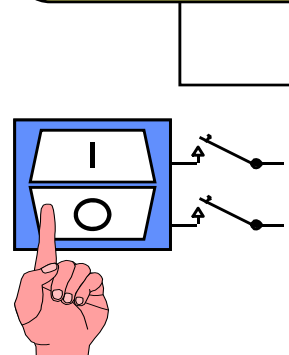
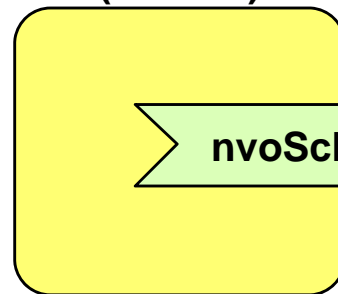
# Binding und Inbetriebnahme

PC mit LON-  
Inbetriebnahme-Tool

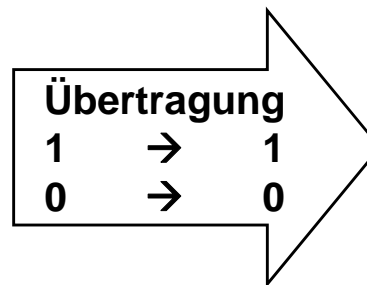
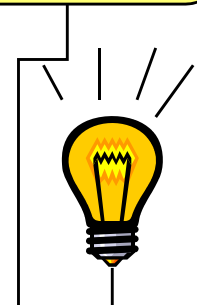
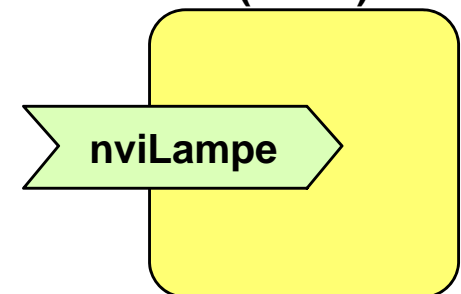


Nach der Konfiguration der Knoten über das Netzwerk mittels PC und Inbetriebnahme-Tool ...

Taster-Knoten  
(Sensor)



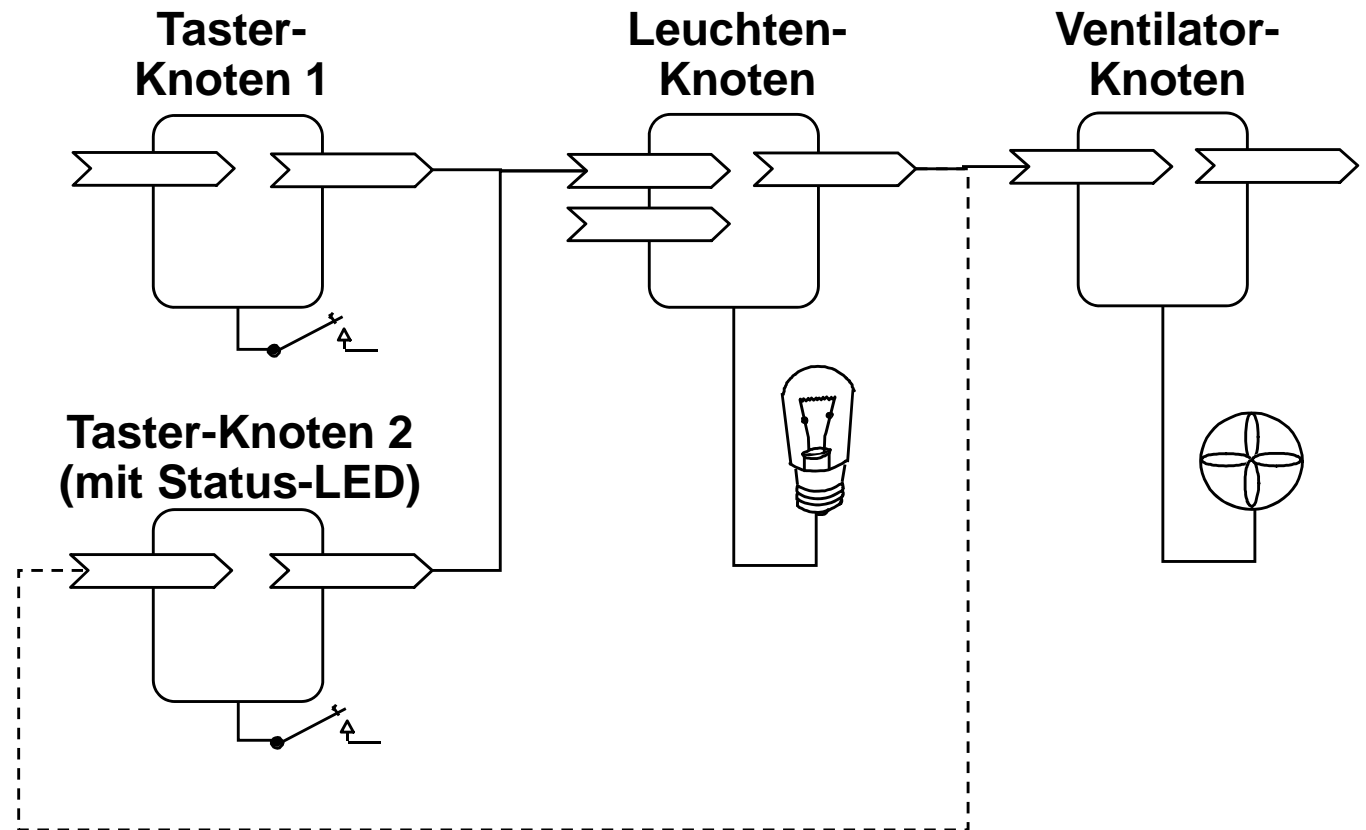
Leuchten-Knoten  
(Aktor)



... sendet der Taster-Knoten jede Änderung seiner Ausgangs-NV automatisch an die Eingangs-NV des Leuchten-Knotens

## Binding:

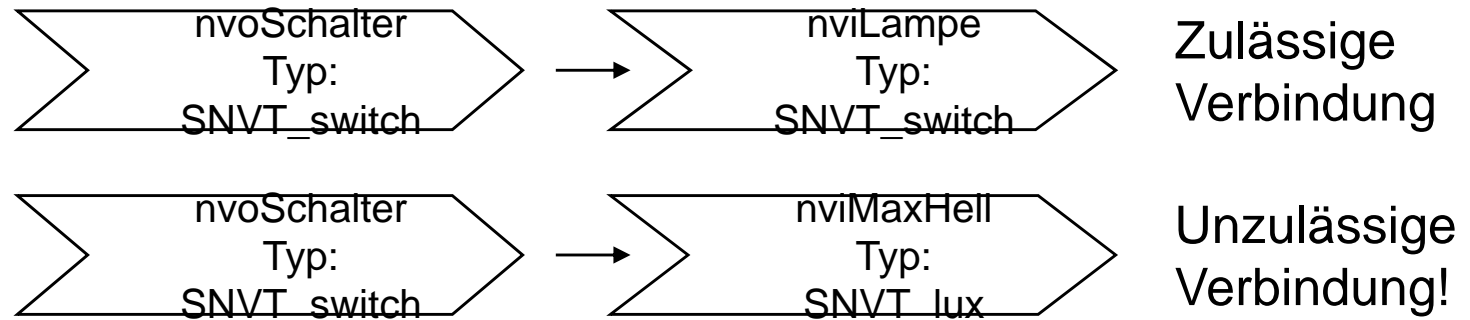
Herstellen der Kommunikationsbeziehungen  
zwischen den Netzwerkknoten



## Regeln für Netzwerkvariablen

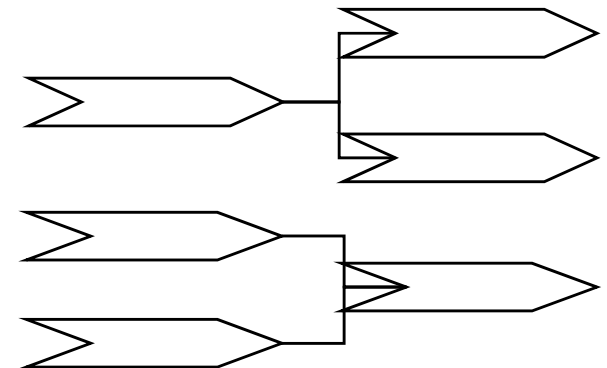
□ Ausgangsvariablen können nur auf Eingangsvariablen gebunden werden

□ Nur Variablen desselben Typs können verbunden werden

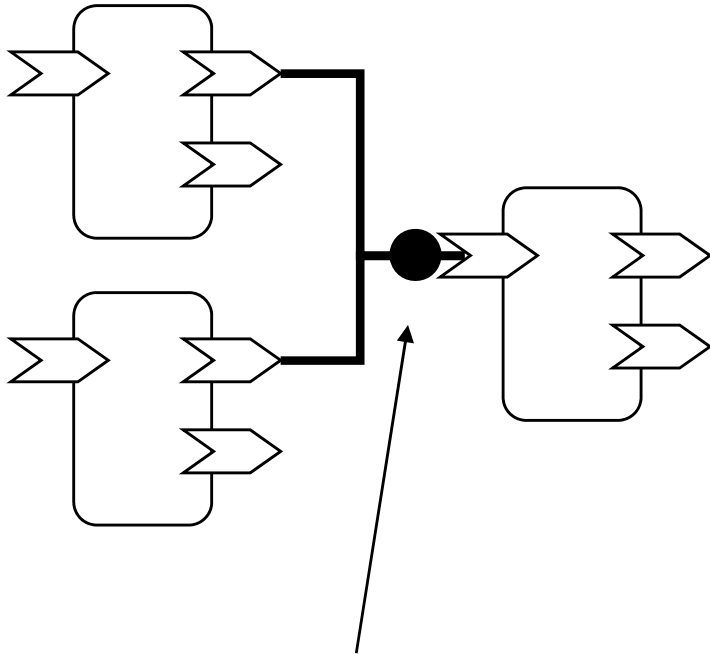


□ **Mehrfach-Bindungen sind möglich**

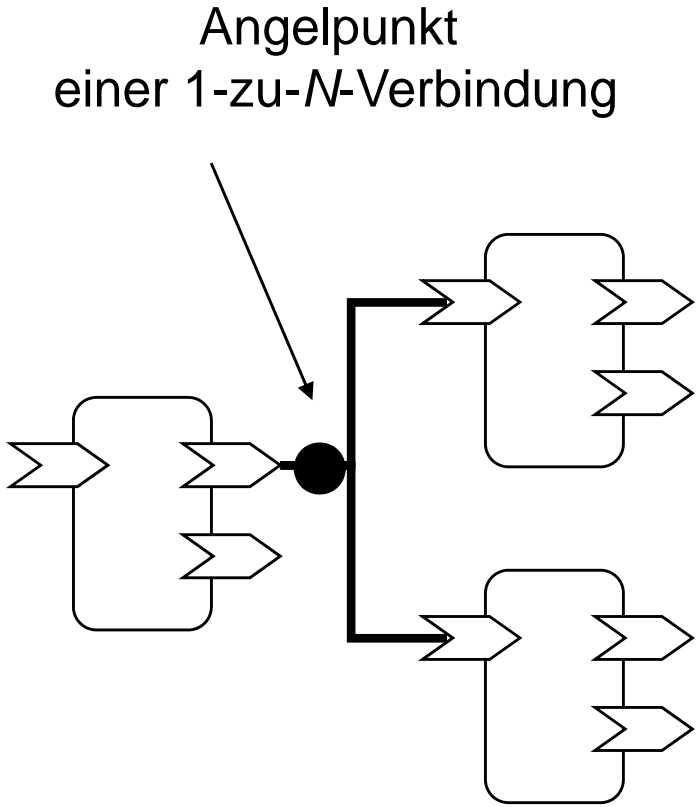
- 1-zu-N Verbindungen, Beispiel:  
Ein Taster-Knoten steuert  
zwei Leuchten-Knoten
- N-zu-1 Verbindungen, Beispiel:  
Zwei Taster-Knoten steuern  
einen Leuchtenaktor



# Verbindungen und Angelpunkte



Angelpunkt  
einer  $N$ -zu-1-Verbindung



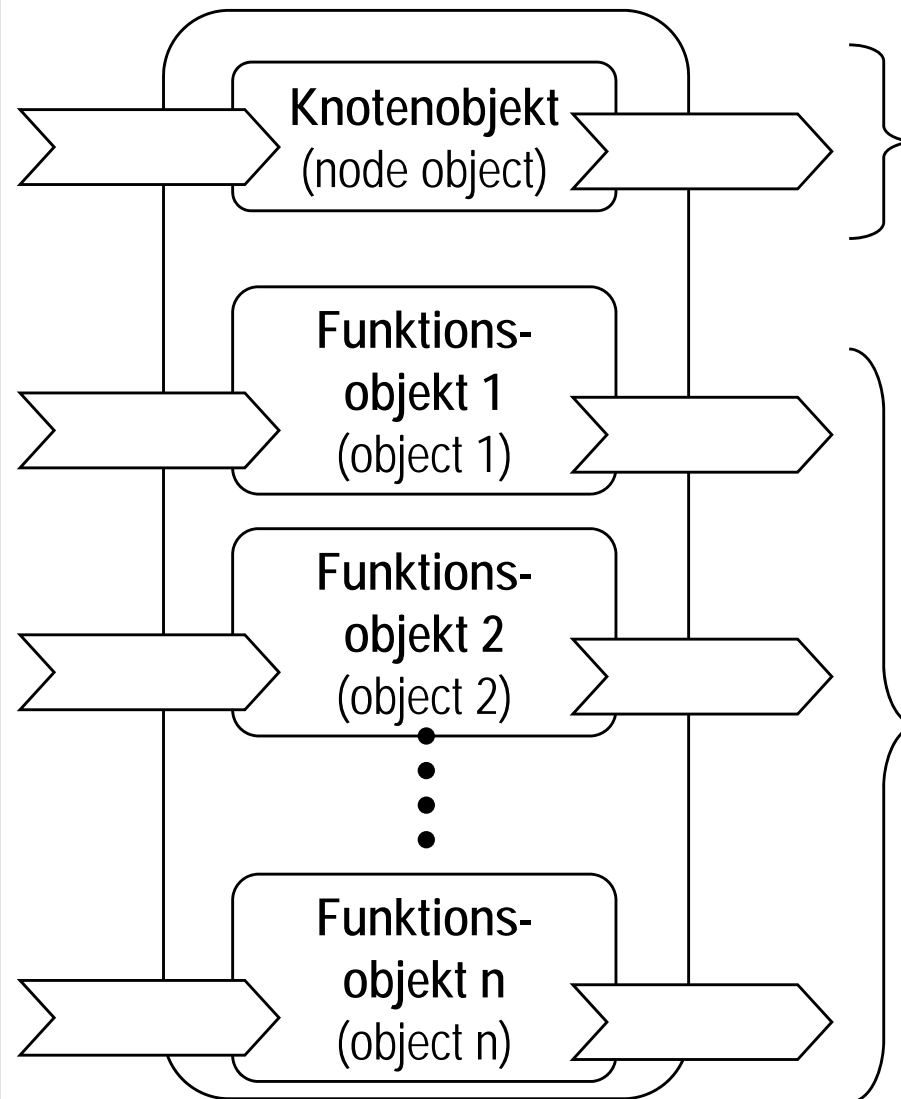
Angelpunkt  
einer 1-zu- $N$ -Verbindung

## Namensgebung von Netzwerkvariablen

| Vorsilbe | Klartext Englisch                      | Klartext Deutsch                                              | Speicherort                                      |
|----------|----------------------------------------|---------------------------------------------------------------|--------------------------------------------------|
| nvi      | network<br>variable<br>input           | Netzwerk-<br>variable<br>(Eingang)                            | Arbeitsspeicher des<br>Neuron<br>(RAM, flüchtig) |
| nvo      | network<br>variable<br>output          | Netzwerk-<br>variable<br>(Ausgang)                            |                                                  |
| nci      | network<br>variable<br>config<br>input | Netzwerk-<br>variable für<br>Konfigurationsdaten<br>(Eingang) | EEPROM des Neuron<br>(nichtflüchtig)             |
| nro      | network<br>variable<br>rom             | wie nvo                                                       | Programm-Speicher<br>(ROM)                       |

# Darstellung eines Knotens

## Knoten allgemein



Systemfunktionen des Knotens (Gerät aktivieren, Fehler auslesen, Fehler zurücksetzen usw.)

Ein Objekt pro Funktionseinheit

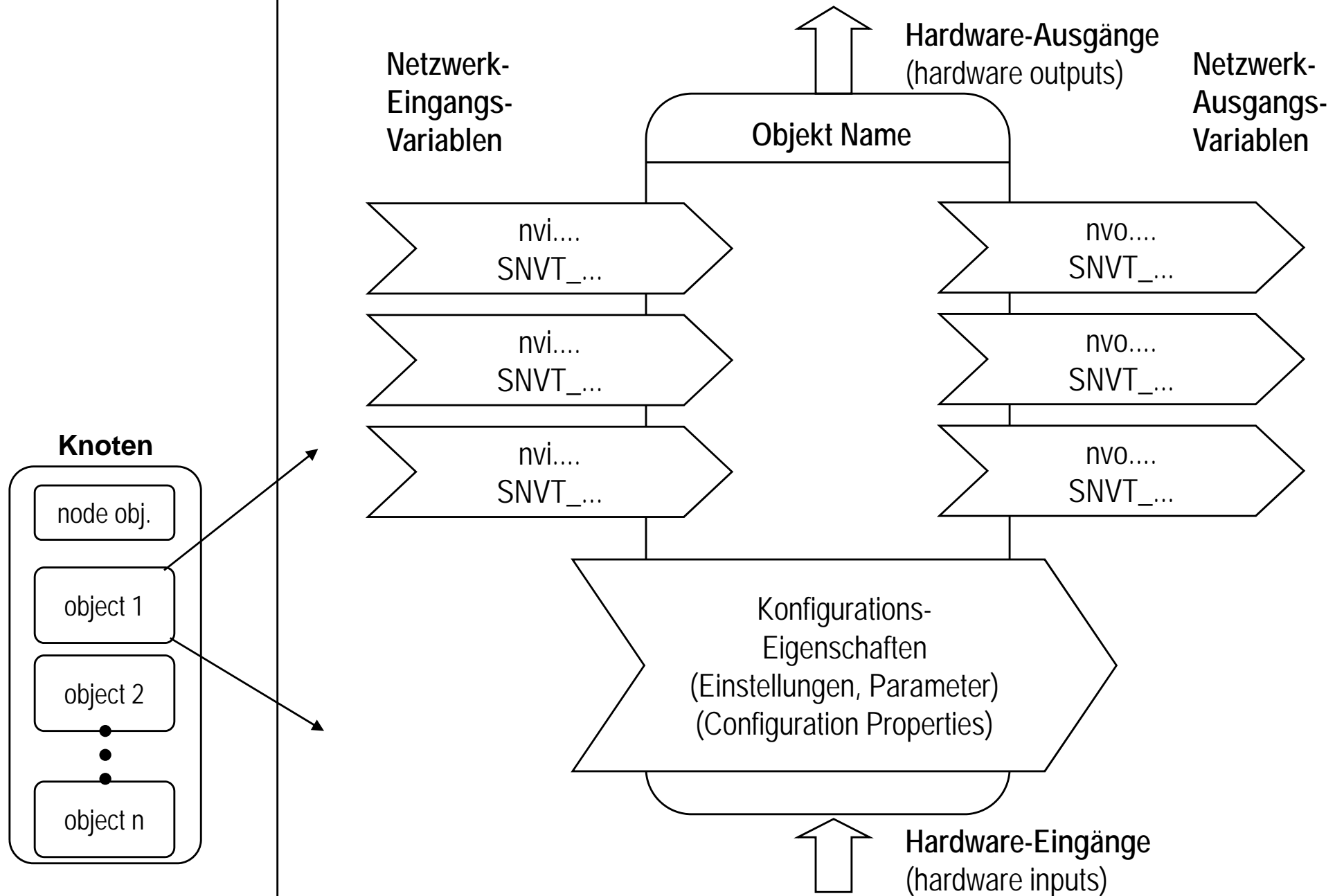
Beispiele:

achtfach Digitalausgang:  
=> 8 Objekte „Schaltaktor“

zweifach Jalousieausgang:  
=> 2 Objekte  
„Jalousieaktor“

vierfach Tasterwippe:  
=> 4 Objekte „Tastsensor“

# Allgemeine Darstellung eines Objekts



# Standard-Netzwerk-Variablen-Typen (SNVT)

□ **Standardisierung  
nach LonMark**

- **dienen der Normung des Datenaustauschs über Netzwerkvariablen zwischen Knoten verschiedener Hersteller**
- **zusammengefasst in der LonMark-SNVT-Master-List**  
<http://www.lonmark.org>
- **derzeit über 150 verschiedene Typ-Definitionen**
- **Jede Typ-Definition umfasst**
  - ▲ Name des Typs
  - ▲ Anwendungsbereich des Typs
  - ▲ Gesamtlänge in Bytes
  - ▲ Angaben, wie die Werte zu interpretieren sind (Wertebereich von ... bis, Schrittgröße / Auflösung)
- **Größere Typdefinitionen sind möglich: sog. Strukturen verbinden mehrere Einzelwerte zu einem Datenblock (struct)**

# Standard-Netzwerkvariablen-Typen (SNVT) – Beispiele –

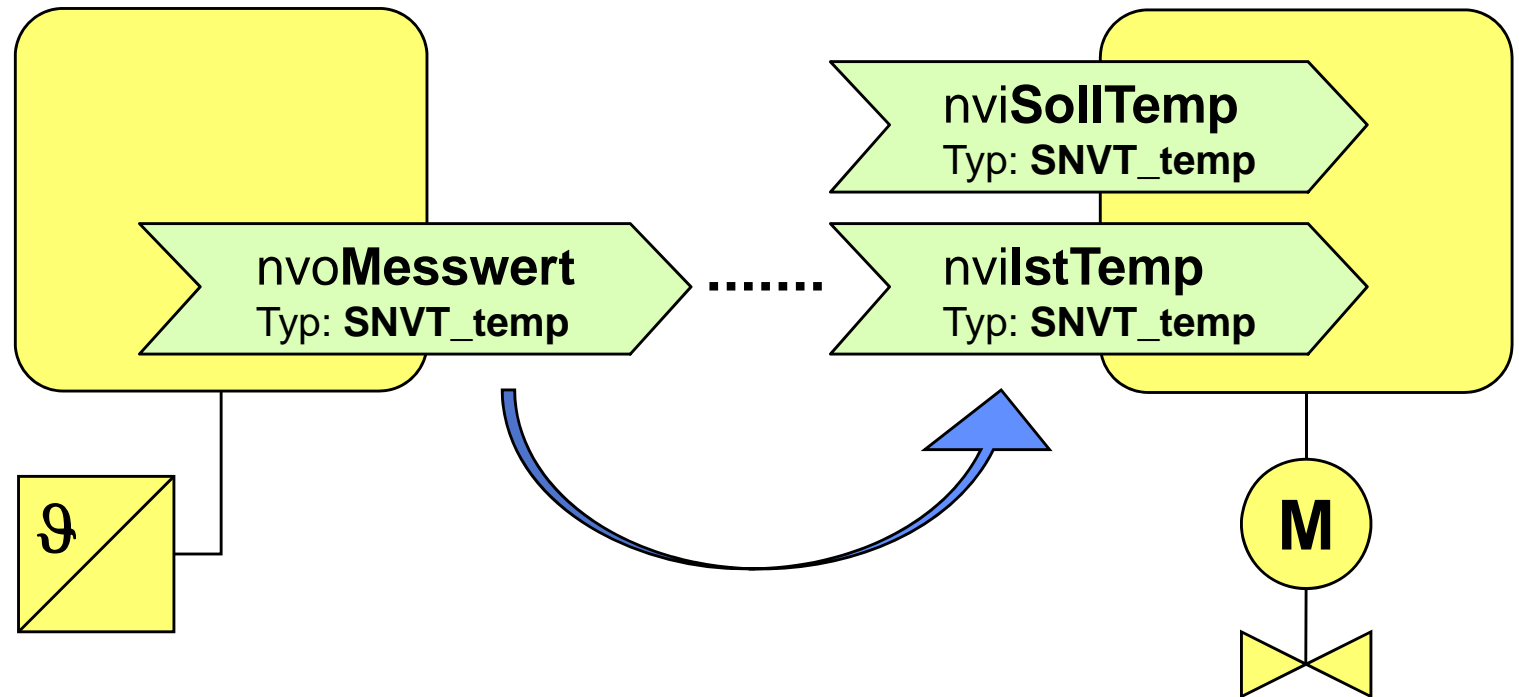
|                                 | Name des Typs           |
|---------------------------------|-------------------------|
|                                 | SNVT_temp               |
| Anwendungsbereich des Typs      | Temperatur allgemein    |
| Basistyp(en)                    | unsigned long (integer) |
| Gesamtlänge in Bytes            | 2 Byte                  |
| Wertebereich von..bis           | -274 °C... +6279,5 °C   |
| Schrittgröße / Auflösung        | 0,1 °C                  |
| Beispiel Messwert               | 0 °C                    |
| entspr. Übertragung (Roh-Daten) | 2740                    |

**Standardisierung nach LonMark**

# SNVT - Beispiel

## Temperatur-Sensor

## Raumtemperatur-Regler



Gemessene  
Raumtemperatur:  
**20 °C**

Übertragener  
Zahlenwert:  
**2940**  
(= 2740 + 20 °C : 0,1 °C)

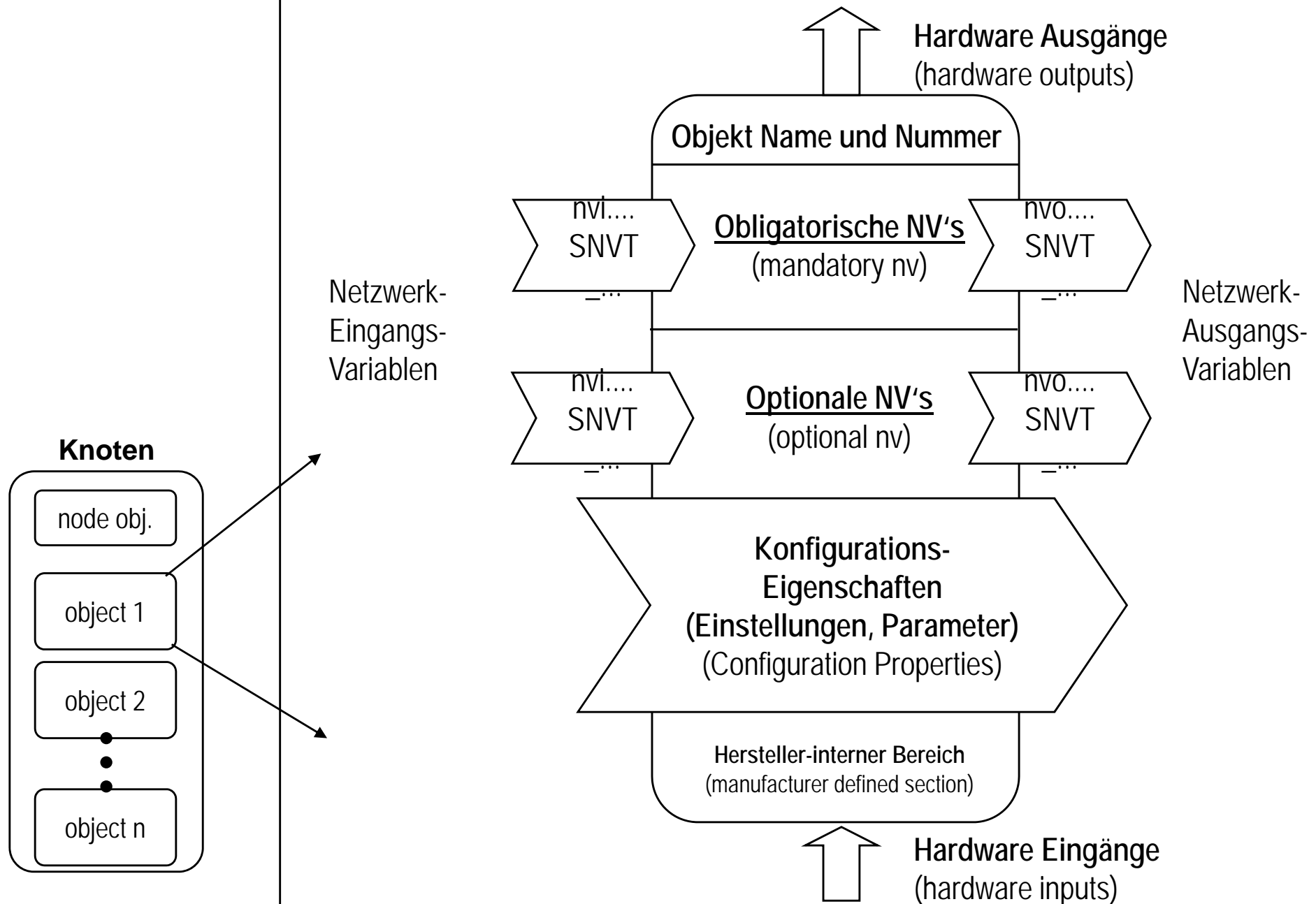
Empfangene  
Temperatur-  
information:  
**20 °C**

# Standard-Netzwerkvariablen-Typen (SNVT) – Beispiele –

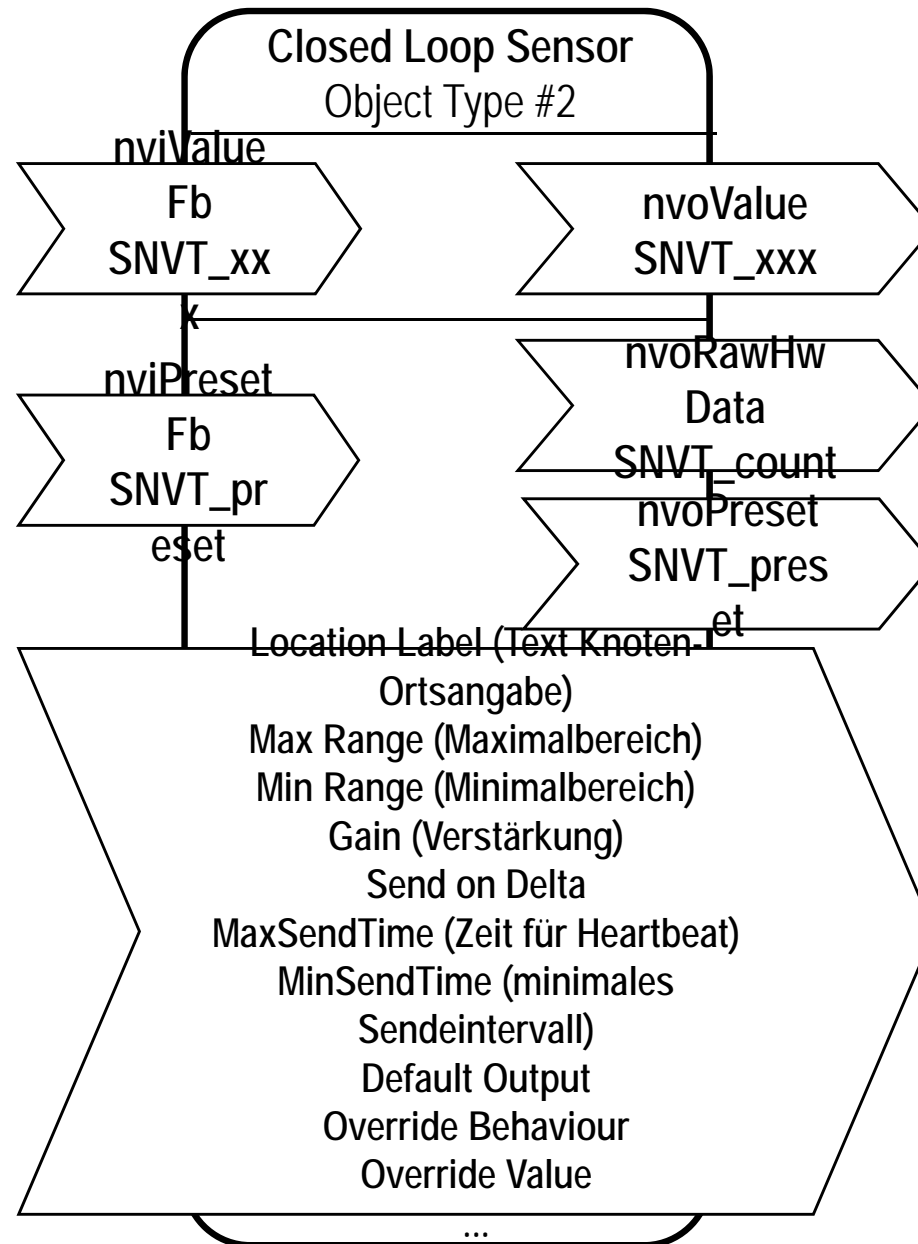
|                                 | Name des Typs            |                          |                                           |                         |  |
|---------------------------------|--------------------------|--------------------------|-------------------------------------------|-------------------------|--|
|                                 | SNVT_lev_cont            | SNVT_temp                | SNVT_temp_p                               | SNVT_lux                |  |
| Anwendungsbereich des Typs      | Prozentwerte             | Temperatur allgemein     | Temperatur für Heizung, Lüftung und Klima | Helligkeit              |  |
| Basistyp(en)                    | unsigned short (integer) | unsigned long (integer)  | signed long (integer)                     | unsigned long (integer) |  |
| Gesamtlänge in Bytes            | 1 Byte                   | 2 Byte                   | 2 Byte                                    | 2 Byte                  |  |
| Wertebereich von..bis           | 0 % ...100 %             | -274 °C...<br>+6279,5 °C | -273,17 °C...<br>+327,66 °C               | 0 lux ...<br>65535 lux  |  |
| Schrittgröße / Auflösung        | 0,5 %                    | 0,1 °C                   | 0,01 °C                                   | 1 lux                   |  |
| Beispiel Messwert               | 10 %    100 %            | 0 °C                     | -10 °C    100 °C                          | 500 lux                 |  |
| entspr. Übertragung (Roh-Daten) | 20        200            | 2740                     | -1000    10000                            | 500                     |  |

□ **Standardisierung nach LonMark**

# Darstellung eines Objekts nach LonMark



# Sensor mit Rückkopplung



**LonMark-Objekt**