Internet of Things (AIoT)

Vorlesungsüberblick und Lernstruktur

Dieser Foliensatz bietet einen strukturierten Überblick über die zentralen Themen der Vorlesung: IoT-Begriffe und Anwendungsfälle, Hardware und Einplatinenmaschinen, Infrastruktur und MING-Stack sowie drahtlose Kommunikation als größten Theorieblock. Ziel ist sowohl ein fachlicher Gesamtüberblick als auch eine klare Orientierung für die Prüfungsvorbereitung.

KAPITEL 0 — LERNSTRATEGIE

Klausur-Tipp: Wie Sie sich am besten vorbereiten

Besonders wirksam ist meist: Stoff eingrenzen, in Themenblöcke aufteilen, die wichtigen Punkte selbst zusammenfassen und das Gelernte regelmäßig ohne Unterlagen wiedergeben.

Planung

- Verschaffen Sie sich zuerst einen Überblick: Was ist klausurrelevant, was ist Randstoff?
- Teilen Sie den Stoff in inhaltliche Blöcke auf und planen Sie Lerntage realistisch (ca. 10 % Puffer)
- Setzen Sie Prioritäten auf Kerninhalte, Zusammenhänge und Anwendung — nicht auf jedes Detail

Lernmethoden

- Fassen Sie den Stoff selbst zusammen — handschriftliche Notizen zwingen zur aktiven Verdichtung
- Nutzen Sie Mindmaps, Skizzen und Farbsysteme, um Zusammenhänge sichtbar zu machen
- Fragen Sie sich ständig selbst ab: Karteikarten, lautes Erklären in eigenen Worten

Tagesrhythmus

- Lernen Sie in konzentrierten Einheiten von 1–2 Stunden, dann ~20 Min. Pause
- Halten Sie Ablenkungen fern: Handy, Fernsehen, zielloses Surfen sind auch in Pausen kontraproduktiv
- Erst allein durcharbeiten, dann mit anderen gegenseitig abfragen

Arten von Klausurfragen

Technologievergleiche

Vergleiche von Technologien miteinander — technisch (Protokoll, Overhead, Reichweite) und nach Einsatzmöglichkeiten (wann eignet sich was?).

Wie funktioniert X?

Erklärung von Mechanismen und Verfahren, z. B. der Medienzugriff in WLAN (CSMA/CA, Backoff, Interframe Spaces).

Use-Case-Betrachtungen

Gegeben ist ein konkreter Anwendungsfall — welche Technologien eignen sich, und warum? Begründung ist entscheidend.

Rechenaufgaben

Was in den Übungen gerechnet wurde (z. B. Frequenz, Wellenlänge, Dämpfung, QAM, Datenraten), muss auch in der Klausur sicher hergeleitet werden können.

Überblick und Zusammenhänge

Fehlt Detailwissen zu einer Technologie, so zumindest die übergeordneten Eigenschaften kennen und einordnen können.

KAPITEL 0 — LERNSTRATEGIE

Kurz vor der Klausur & Technische Fächer

Kurz vor der Klausur

- Am letzten Tag nichts Neues anfangen — nur wiederholen und verdichten
- Einen kleinen Lernzettel als Zusammenfassung schreiben (natürlich nicht in die Prüfung mitnehmen)
- Ausreichend Schlaf ist deutlich sinnvoller als eine Nachtschicht

Für technische Fächer

- Themen top-down ordnen: erst Hauptthemen, dann Unterthemen, dann typische Verfahren, Vor- und Nachteile, Standardanwendungen
- Rechenwege und Anwendungsaufgaben aktiv üben, bis sie ohne Unterlagen sauber hergeleitet werden können
- Nicht nur lernen, was eine Technologie ist — sondern auch wann man sie einsetzt und warum man Alternativen verwirft

KAPITEL 0 — LERNSTRATEGIE

Wie Studenten für diese Vorlesung lernen sollten

1 Überblick

Alle Hauptthemen zunächst im Überblick erfassen und top-down in Themen, Unterthemen und Technologien gliedern. Themen, die in der Vorlesung mehr Raum hatten, verdienen mehr Lernzeit — insbesondere drahtlose Kommunikation.

2 Technische Einordnung üben

Zu jedem Unterthema die Grundidee, die typischen Trade-offs sowie die Einsatzszenarien verstehen und vergleichen können. Bei Use-Case-Aufgaben Plattformen, Kommunikationstechnologien und Architekturen (Cloud, Fog, Edge, On-Device) mit Vor- und Nachteilen einordnen.

3 Aktiv verdichten

Handschriftliche Lernzettel anfertigen und den Stoff aktiv komprimieren. Übungen erneut durchgehen — Rechenaufgaben (insbesondere zu Frequenz, Wellenlänge, Dämpfung, QAM) sicher beherrschen.

4 Selbstabfrage mit KI-Tools

KI-gestützte Wissensabfrage nutzen, z. B. NotebookLM zur Generierung von Beispielaufgaben auf Basis der Vorlesungsfolien. Aktives Abrufen von Wissen ist effektiver als passives Lesen.

Playlist der Vorlesungsvideos: https://www.youtube.com/playlist?list=PLFJGmP04pevnjjsw3qFHNbsp9CAXeqin6

Kalman Graffi

LA
1.2.03
Zusammenfassung

1) Logik und Mengen

Indirekte Beweis: $(p \Rightarrow q) \Leftrightarrow (\neg q \Rightarrow \neg p) \Leftrightarrow \neg(p \wedge \neg q)$

Potenzmenge: $P(A) := \{B : B \subseteq A\}$ alle Teilmengen von A

n-Tupel: Kartesisches Produkt von X und Y: $X \times Y := \{(x,y) : x \in X, y \in Y\}$ geordnete Liste $(a_1 \dots a_n)$ ist der n-Tupel

Abbildung: $\varphi : M \to N$ ist eine Relation (M, N, Γ_φ) wobei M und N so dass gilt: $(\forall x \in M)(\exists! y \in N)(x,y) \in \Gamma_\varphi$ (Graph)

Gleichheit: zwei Abbildungen (M, N, Γ_φ) und $(M', N', \Gamma_{\varphi'})$ sind gleich, wenn $M = M'$ und $N = N'$ und $\Gamma_\varphi = \Gamma_{\varphi'}$

Einschränkung der Abbildung: ist $\varphi : M \to N$ und $A \subseteq M$ so kann man die φ auf A einschränken $\varphi|_A : A \to N$ $x \to \varphi(x)$

Inverse Relation: Vorraussetzung: φ: ist bijektiv $\Gamma^{-1} = \{(y,x) \in N \times M : (x,y) \in \Gamma\}$ (ein Inverse Funktion von φ)

Komposition: $\varphi \circ \psi : M \to P$, $x \to \psi(\varphi(x))$ für $\varphi : M \to N$ und $\psi : N \to P$ ist $\psi \circ \varphi : M \to P$ ~~Multiplikat~~

Mächtigkeit: Kardinalität von M: $|M| := n$ Anzahl der Elemente. Permutation: bijektive Abbildung einer Menge in sich.

2) Zahlen

Gegeben: $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$, dazu 2 binäre Operationen + und · => Asso z iativ, neutr. Elem, inv. Element, Kommutativ, Distr. Gesetz

Spaltenschreibweise: Weitere Schreibform: Elemente $x = (x_1, \dots, x_n)$ des n-fachen kartesischen Produktes $\mathbb{R}^n$ $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{R}^n$ => Addition und Skalarmultiplikation

Lineares Gleichungssystem: $a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1n}x_n = b_1$, $a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + \dots + a_{mn}x_n = b_m$ mit Matrizenprodukt: $Ax = b$ für $b = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}$ (n-Spalten, m-Zeilen) Lösung ist dann n-Tupel.

Gauß-Jordan-Verfahren: in Stufenform bringen dazu nutze: (E1) Addition eines Skalarvielfachen einer anderen Zeile (E2) Zeilenvertauschen (E3) Zeile ·λ

=> Pivotvariablen: die x_i deren Zeile nicht $= 0$ ist x_i^{Pivot} => Lösung: 1) $\lambda = 0$ für $x_i \neq 0$ unlösbar 2) Nicht Pivotvar. beliebig

Matrizen: Tabelle der Koeffizienten und Konstanten $A = (a_{ij}) = (a_{ij})_{m \times n} = (a_{ij})_{\substack{i=1,\dots,m \\ j=1,\dots,n}}$ | a Zeile Spalte

Einheitsmatrix: $E_{n \times n} = E_n = E = (\delta_{ij})_{n \times n}$ wobei $\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$

Matrizenprodukt: $C_{i \times k} = \sum_{j=1}^{n} a_{ij} b_{jk} = a_{i1}b_{1k} + a_{i2}b_{2k} + \dots + a_{in}b_{nk}$ dazu: $A_{n \times m} \cdot B_{m \times p} = C_{n \times p}$ $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

Matrizenprodukt Regeln: $(AB)C = A(BC)$; $(A + A')B = AB + A'B$, $A(B + B') = AB + AB'$; $E_m A = A = A E_n$, $(\lambda A)B = \lambda(AB) = A(\lambda B)$

Homogene Lösung: $Ax = b$ ist homogen wenn $b = 0$, x_h: $Ax_h = 0$ inhomogen: $b \neq 0$ x_s: $Ax_s = b$ => $x_{allg} = x_h + x_s$ da $A(x_h + x_s) = b$

Invertierbarkeit: $A \in M_n(\mathbb{R})$ heißt invertierbar, falls ex. $B \in M_n(\mathbb{R})$ so dass $AB = E = BA$ => $B, A \in GL_n(\mathbb{R})$

Bist dann: $B = A^{-1}$; $(AB)^{-1} = B^{-1}A^{-1}$; 2x2 Matrix: $A^{-1} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{ad - bc}\begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$

Berechnung des Inversen über den Gauß-Algo: Ergänzt: $Ax = y \Leftrightarrow x = A^{-1}y$ => $\underline{A|E} \xrightarrow{\text{Gauß-Algo}} \underline{E|A^{-1}}$

Transponierte: A^T: i-te Zeile von A wird i-te Spalte von A^T, die j-te Spalte von A wird j-te Zeile von A^T

Regeln: $(A^T)^T = A$; $(A + A')^T = A^T + (A')^T$; $(\lambda A)^T = \lambda \cdot (A')$; $(AB)^T = B^T A^T$

3) Algebraische Strukturen

binäre Operation: Abbildung $S \times S \xrightarrow{*} S$, $(a,b) \to a * b$ entweder additiv oder multiplikativ

Halbgruppe: $(M, *)$: Menge M und assoziative Operation * auf M: $(a * b) * c = a * (b * c)$

Kommutativität: $a * b = b * a$, falls erfüllt => kommutative <Strukturbezeichnung>

Potenz: $(S, *)$ ist Halbgruppe, $a \in S$ => $a^n = \underbrace{a * a * \dots * a}_{n\text{-mal}}$

Monoid: $(M, *, e)$ Halbgruppe mit neutralem Element $e * a = a = a * e$ (multiplikativ meist 1, additiv meist 0)

Morphismen: Abbildungen, die die mathematische Struktur respektieren.

Halbgruppenhomom: Sei $(S_1, *_1)$ und (S_2, Δ) Halbgruppen, $f : S_1 \to S_2$; f ist Halbhomom wenn $f(s * t) = f(s) \Delta f(t)$

Monoidhomomorphismus: $(S_1, *_1, e_1)$ und (S_2, Δ, e_2), $f : S_1 \to S_2$ => $f(s * t) = f(s) \Delta f(t)$ und $f(e_1) = e_2$

Isomorphismus: $f : S_1 \to S_2$, f ist ein Isom. wenn: ex. $g : S_2 \to S_1$: $g \circ f = id_{S_1}$ und $f \circ g = id_{S_2}$ d.h. $g(f(s) \Delta f(t)) = s * t$

Endomorphismus: Homom. in sich selbst: $f : S \to S_1$ $f(s * t) = f(s) * f(t)$ | bei Monoid: $f(e) = e$

Automorphismus: Endom. und Isom gleichzeitig: $f : S_1 \to S_1$ ex. $g : S_1 \to S_1$, $f(s * t) = f(s) * f(t)$ und $g(f(s) * f(t)) = s * t$

Invertierbare Elem.: Im Monoid $(M, *, e)$: $a \in M$ ist invertierbar, wenn ex. $b \in M$ dass $a * b = e = b * a$ => $b = a^{-1}$ multiplikativ

Gruppen: Monoid $(G, *, e)$, indem jedes $a \in G$ invertierbar ist. (G1*, e): (G1) assoziativ (G2) neutr. Elem (G3) $(\forall a \in G)(\exists b \in G)$: $a * b = e = b * a$

abelsche Gruppe: (G4): $(\forall a, b \in G)$: $a * b = b * a$ (G3*) $(\forall a \in G)(\exists b \in G)$: $a * b = e = b * a$

Gruppenhomom: $(G_1, *_1, e_1)$ und $(G_2, *_2, e_2)$ sind Gruppen dann äquivalent: $(G_1, *_1)$, (G_2, Δ) Homom $\Leftrightarrow$ (GH1) $(\forall a, b \in G_1)$ $f(a * b) = f(a) \Delta f(b)$

(GH2) $f(e_1) = e_2$ (GH3) $f(a^{-1}) = f(a)^{-1}$

Einheitengruppe: Monoid $(M, *, e)$ => $(M^\times, *)$ die Einheitengruppe, die alle invertierbaren Elemente aus M enthält

Ring (mit 1): $(G, M, +, \cdot)$ bzw. Ring mit 1): $(M, +, \cdot, 0, 1)$: 1) $(M, +, 0)$ ist abelsche Gruppe (kommutativ, assoziativ, neutr. Elem, inverses Element) 2) Dist. Distributivgesetze 3) $(M, \cdot)$ ist eine Halbgruppe bzw $(M, \cdot, 1)$ ist ein Monoid mit 1 ≠ 0 (kommutativ, assoziativ, neutr. Elem)

Körper: kommutativer Ring mit 1, in dem $0 \neq 1$ und $K^\times = K \setminus \{0\}$, d.h. keine Nullteiler => $ab = 0 \Rightarrow a = 0 \vee b = 0$ (Nullteilerfrei) (kommutativ, assoziativ, neutr. Elem, inverses Element)

Korpus $(M, +, \cdot, 0, 1)$ (K1) assoziativ (K2) neutr. Elem. (K3) invertierbar (K4) kommutativ $a + b = b + a$, $ab = ba$ (K5) $a(b + c) = ab + ac$

Ring/Korpushomom: $f(x + y) = f(x) + f(y)$; $f(x \cdot y) = f(x) \cdot f(y)$ Iso: $f : R \to S$ ex. $g : S \to R$ $g \circ f = id_R$ $f \circ g = id_S$

Komplexe Zahlen: $\mathbb{C} := \mathbb{R}^2$ $(\mathbb{C}, +, \cdot, (0,0), (1,0))$ $z = x + iy$, $i := (0,1)$ $(x,y) + (x',y') = (x + x', y + y')$; $(x,y) \cdot (x',y') = (xx' - yy', xy' + x'y)$ $i^2 = -1$

Ringe $\mathbb{Z}/n\mathbb{Z}$: Operationen: + und · auf die Menge $\mathbb{Z}/n\mathbb{Z} = \{\bar{0}, \bar{1}, \dots, \overline{n-1}\}$ => einen kommutativen Ring, $\bar{0} = e_1$, $e_2 = \bar{1}$

Äquivalenzrelation: Relation $R \subseteq M \times M$ auf der Menge M heißt Äq. wenn 1) reflexiv: $(\forall x \in M)\, xRx$ 2) symmetrisch $(\forall x, y \in M)\, xRy \Rightarrow yRx$ 3) transitiv $(\forall x, y, z \in M)\, xRy \wedge yRz \Rightarrow xRz$

Klasseneinteilung/Partitionierung: Menge von Teilmengen $P = \{P_i : i \in I\} \subseteq P(M)$ (I: Indexmenge) : 1) $M = \bigcup_{i \in I} P_i$ 2) $(\forall i, j \in I)\, i \neq j \Rightarrow P_i \cap P_j = \emptyset$

$\bar{x} = \{y \in M : xRy\} \subseteq M$ => für $x, y \in M$ Pictotome: 1) $xRy \Rightarrow \bar{x} = \bar{y}$ 2) $\neg(xRy) \Rightarrow \bar{x} \cap \bar{y} = \emptyset$ Lemma: $(\forall x \in M)(\exists! i \in I)\, x \in P_i$

$\mathbb{Z}/n\mathbb{Z}$ ist Körper: Falls n Primzahl => $\mathbb{Z}/n\mathbb{Z}$ ist Körper (dann alles invertierbar)

Kalman Graffi

LA
1.2.03

4) $\mathbb{K}$-Vektorraum: "$(M, +, \cdot, 0, 1)$" (Mult. => M, $\mathbb{K} \times M \to M$) Besonders im Vergleich zum Körper: $\cdot$ Skalarmultiplikation! $\mathbb{K} \times M \to M$ Zusammenfassung

(V1) $(M, +)$ assoziativ: $(\forall v, u, w \in M)\, u + (v + w) = (u + v) + w$ (V2) $(M, +, 0)$ Monoid: $(\forall v \in M)\, v + 0 = v$ (V3) $(M, +, 0)$ Gruppe: $(\forall v \in M)\, v + (-v) = 0$

(V4) $(M, +, 0)$ abelsche Gruppe: $(\forall v, w \in M)\, u + w = v + u$ (V5) $(M, \cdot, 1)$ (kein Inverses) HG assoziativ: $(\forall v \in M)(\forall \lambda, \mu \in \mathbb{K})\, \lambda(\mu v) = (\lambda \mu) v$ (V6) $(M, \cdot, 1)$ Monoid: $(\forall v \in M)\, 1v = v$

(V7) Mit Distributiv: $(\forall u, v \in V)(\forall \lambda \in \mathbb{K})\, \lambda(v + w) = \lambda v + \lambda w$ (V8) $(\forall v \in M)(\forall \lambda, \mu \in \mathbb{K})\, (\lambda + \mu) v = \lambda v + \mu v$

Abgeleitete Regeln: (V9) $(\forall u, v, w \in M)\, u + w = v + w \Rightarrow u = v$ (V10) $0 \cdot v = 0$ (V11) $(\forall v \in M)\, 0v = 0$ $(\forall \lambda \in \mathbb{K})\, \lambda \cdot 0 = 0$ $(\forall v \in M)(\forall \lambda \in \mathbb{K})\, \lambda v = 0 \Rightarrow \lambda = 0 \vee v = 0$

$M_{m,n}(\mathbb{K})$: Menge aller $m \times n$ - Matrizen mit Koeffizienten aus $\mathbb{K}$. Die Addition (Multiplikation in $\mathbb{K}$ gibt die komponentenweise Addition (Skalarmultipl.) $M_{m,n}(\mathbb{K})$

Lineare Abbildungen: (Abbildungen, die die algebraische Struktur erhalten) 1) V, W sind $\mathbb{K}$-VR, $\varphi : V \to W$ linear => VR-Homo-Morph.; $\varphi : V \to W$ ex. linear $\varphi : W \to V$ $\varphi \circ \varphi = id_V$ und $\varphi \circ \varphi = id_V$ => VR-iso-Morph.; $\varphi : V \to V$ linear => VR-Endo-Morph. $\forall v \in \mathrm{End}(V)$

Morphismen: $\varphi : V \to W$ linear => VR-Homo-Morph.; $\varphi : V \to W$ ex. linear $\varphi : W \to V$ $\varphi \circ \varphi = id_V$ und $\varphi \circ \varphi = id_V$ => VR-iso-Morph.; $\varphi : V \to V$ linear => VR-Endo-Morph. $\forall v \in \mathrm{End}(V)$

$\mathrm{Hom}(V, W)$: Menge aller linearen Abbildungen $\varphi : V \to W$, Automorphismus $\varphi : V \to V$: iso+Endo $= \mathrm{Aut}(V)$

$(VR, +)$ abelsche Gruppe: => $(\varphi + \psi)(v) := \varphi(v) + \psi(v)$ und $(\varphi \circ \psi) := \varphi(\psi(v))$ für $v \in VR$

Unterpektorraum von V: $U \subseteq V$: (U1) $0 \in U$ (U2) $\forall u, v \in U \Rightarrow v + w \in U$ (U3) $\forall v \in U, \lambda \in \mathbb{K} \Rightarrow \lambda v \in U$ d.h. additiv und Skalarmultiplikativ abgeschlossen

=> (da abgeschlossen) $v_1, \dots, v_n \in U$, $\lambda_1, \dots, \lambda_n \in \mathbb{K}$ => $\lambda_1 v_1 + \lambda_2 v_2 + \dots + \lambda_n v_n$ => Span(U) = auch UVR. UVR ist auch ein VR, da die Axiome gelten!

Prüfen ob UVR: Überprüfung der Axiome (V1)-(V8) oder zeigen: U ist UVR eines VR V => U ist selber VR

$U = \bigcap_{i \in I} U_i = \{v \in V : (\forall i \in I)\, v \in U_i\}$ => U ist UVR, wenn $(U_i)_{i \in I}$ UVR

$\mathrm{span}(B) := \bigcap \{U \subseteq V : B \subseteq U,\ U \text{ ist UVR von } V\}$ der von B aufgespannte erzeugte UVR von V => span(B) ein UVR von V

Summe von UVR: $U_1, \dots, U_n$ seien UVR von V => $U_1 + U_2 + \dots + U_n = \{u_1 + \dots + u_n : u_1 \in U_1, \dots, u_n \in U_n\}$ ebenfalls UVR von V

direkte Summe: $U_1, \dots, U_n$ seien UVR von V => $U_1 \oplus U_2 \oplus \dots \oplus U_n = V$ Bedingungen: 1) $V = U_1 + \dots + U_n$ 2) $U_i \cap U_j = \{0\}$

Affine UVR: $A \subseteq VR$ V heißt affines UVR wenn ex. UVR U von V, $a \in V$ so dass $A = a + U = \{a + u, u \in U\}$ (parallel-verschoben)

Lineare Abbildungen und UVR: Sei $\varphi : V \to W$ linear: U UVR => $\varphi(U)$ UVR; U UVR => $\varphi^{-1}(U)$ UVR; A UVR => $\varphi(A)$ UVR; A UVR => $\varphi^{-1}(A)$ UVR

$\varphi^{-1}(b) := \{x \in V : \varphi(x) = b\}$ für $\varphi : V \to W$ linear und $b \in W$ $\varphi^{-1}(b)$ Menge aller Lösungen von $\varphi(x) = b$ entweder leer oder UVR

Direkte Produkte: kartesisches Produkt auf $(\mathbb{K}V)^n$: $V = V_1 \times V_2 \times \dots \times V_n$ Dazu komponentenweise Addition, Skalarmultiplik. $\to V$ VR, das direkte Produkt von V_i

Linearkombination: $v_i \in V$; V VR ($\mathbb{K}$): $v = \sum_{i=1}^{n} \lambda_i v_i$ => v ist LK aus v_i Menge aller LK: $\mathrm{span}(v_1, \dots, v_n) = \mathrm{span}(\{v_1, \dots, v_n\})$

$\mathrm{span}(v_1, \dots, v_n)$: $\mathbb{K}v_1 + \dots + \mathbb{K}v_n$ => UVR für $w_i \in \mathrm{span}(v_i)$ => $\mathrm{span}(w_i) \subseteq \mathrm{span}(v_i)$ | $\mathrm{span}(B) = \bigcup \mathrm{span}(E)$ falls $E \subseteq B$ endlich

Lineare Unabhängigkeit: $\lambda_1 v_1 + \dots + \lambda_n v_n = 0 \Rightarrow \lambda_1 = \dots = \lambda_n = 0$ B ist l.u. wenn alle Teilmengen von B l.u.

Basis: von V: $B \subseteq V$: $\mathrm{span}(B) = V$ und B l.u. d.h. minimal Erzeugendensystem <=> maximal linear unabhängig, ist B

Finden einer Basis: $U = \mathrm{span}(v_1, \dots, v_n) \subseteq \mathbb{K}^n$ => $m \times n$ Matrix mit v_i als Zeilen! => Gauß-Algo => Basisvektoren als Zeilen! (+ Ein dazu => Abhängigkeitsbeweis)

Koordinaten: sei $B := \{v_1, \dots, v_n\}$ eine Basis von V => $(\forall v \in V)(\exists! \lambda_i \in \mathbb{K})$ $v = \lambda_1 v_1 + \lambda_2 v_2 + \dots + \lambda_n v_n$ eindeutige $\lambda_1, \lambda_2, \dots, \lambda_n$, da v_i l.u. n

Koordinaten: $\{v_1, \dots, v_n\} \subseteq V$ Basis von V; $v_1, \dots, v_n$ halten eine best Anordnung fest => $B = (v_1, \dots, v_n)$ angeordnete Basis => $(\forall v \in V)(\exists \lambda_i \in \mathbb{K})$ $v = \sum \lambda_i v_i$

=> Eindeutig bestimmte Skalare $\lambda_1, \dots, \lambda_n$ heißen Koordinaten von v bzgl. des angeordneten Basis B: $[v]_B = \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{pmatrix} \in \mathbb{K}^n$

$\varphi_B : \mathbb{K}^n \to V$ $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \to \lambda_1 v_1 + \lambda_2 v_2 + \dots + \lambda_n v_n$ ist bijektiv (da λ_i eindeutig) φ_B linear => VR-iso-Morphismus zwischen $\mathbb{K}^n$ und V

Beziehung ex: $v = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$ Koord bzg Standardbasis e $[v]_E = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ => $v = x_1 e_1 + x_2 e_2$ (b_1, b_2 Basisvektoren von B) d.h. $\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = x_1 \begin{pmatrix} a_{11} \\ a_{21} \end{pmatrix} + x_2 \begin{pmatrix} a_{12} \\ a_{22} \end{pmatrix}$ => $x_i = \sum_{j=1}^{n} a_{ij} x_j$ => allgemeine Matrix

Beispiel: $e : \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ $b_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ $b_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$ => $\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = x_1 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + x_2 \begin{pmatrix} 1 \\ -1 \end{pmatrix}$, da $[v]_B = \begin{pmatrix} x_1' \\ x_2' \end{pmatrix}$ => $\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = x_1' \begin{pmatrix} 1 \\ 0 \end{pmatrix} + x_2' \begin{pmatrix} 1 \\ -1 \end{pmatrix}$ d.h. $\begin{pmatrix} x_1' \\ x_2' \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ => $\begin{pmatrix} x_1' \\ x_2' \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix}^{-1} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = -\frac{1}{1 \cdot (-1) - 1 \cdot 0} \begin{pmatrix} -1 & -1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$

Dimension: $\dim V = n$ V ist ein VR, n die Anzahl der Basisvektoren von V. Jede Basis hat die gleiche Mächtigkeit, da nur n Vektoren

Basissätze: 1) Jedes endliche Erzeugendensystem B eines VR enthält eine Basis: B l.u. $(B_1 \cup B_2) = V \Rightarrow B_2' \subseteq B_2$, so dass $\mathrm{span}(B_1 \cup B_2') = V$ und Basis von V

2) k l.u. Vektoren in V => k-te weitere (u. wenn $\dim V = n$ 3) B = Basis von V, C l.u. Teilmenge von V => ex. $B' \subseteq B$ => $B' \cup C$ Basis von V

Dimensionssatz: $\dim(U_1 + U_2) + \dim(U_1 \cap U_2) = \dim U_1 + \dim U_2$ (U_i UVR von V) Es gilt für $v_1, \dots, v_n$ in $\dim V = n$: v_i l.u $\Leftrightarrow \mathrm{span}(v_i) = V \Leftrightarrow v_i$ Basis

5) Lin Abb & Matrizen: Lineare Abbildungen (zwischen endlich-dimensionalen $\mathbb{K}$-Vektorräumen (nach Auswahl von Basen)) kann man durch Matrizen beschreiben: $x \to Ax$

$\mathrm{im}\,\varphi := \varphi(V) = \{\varphi(v) | v \in V\} \subseteq$ für $\varphi : V \to W$ $\ker \varphi := \varphi^{-1}(0) = \{v \in V | \varphi(v) = 0\} \subseteq V$, wenn $\ker \varphi = \{0\} \Rightarrow \varphi$ injektiv

Dimensionssatz: $\dim(\mathrm{im}\,\varphi) + \dim(\ker \varphi) = \dim V$ für lin Abb. $\varphi : V \to W$ injektiv: $\ker \varphi = \{0\}$ d.h. $\dim \ker \varphi = 0$

rank φ $= \dim(\mathrm{im}\,\varphi) = \dim(\varphi(V))$ Anzahl der l.u. Vektoren in im φ d.h. Anzahl der Elemente von der Basis von im φ

φ(Basis) Satze: $v_1, \dots, v_n$ ist Basis von V, $\varphi : V \to W$ 1) $\mathrm{span}(\varphi(v_i)) = \mathrm{im}\,\varphi$ (2) $\mathrm{rank}\,\varphi = \dim(\mathrm{im}\,\varphi) \leq$ (3) φ surjektiv => $\mathrm{rank}\,\varphi = \dim W$ (4) φ injektiv => $\mathrm{rank}\,\varphi = \dim V$

=> Möglichkeit zur Überprüfung, ob φ injektiv/surjektiv

φ(Basis) => beliebige Elemente: V, W VR, $v_1, \dots, v_n$ Basis von V, w_i beliebig $\in W$ => Ex. genau eine Abb. $\varphi : V \to W$ so dass $\varphi(v_1) = w_1, \varphi(v_2) = w_2, \dots, \varphi(v_n) = w_n$

Isomorph: Jeder n-dimensionale $\mathbb{K}$-Vektorraum V ist isomorph zu $\mathbb{K}^n$, da lin Abb. $\mathbb{K}^n \to V$ er so dass $\varphi(\text{Basis } \mathbb{K}^n)$ = Basis V

Operationen auf lineare Abbildungen: $\mathrm{Hom}(V, W)$ ist ein $\mathbb{K}$-Vektorraum. $\mathrm{Hom}(V, W)$ ist die Menge aller lin $\varphi : V \to W$ => $(\varphi + \psi)v = \varphi v + \psi v$; $(\lambda \varphi) v = \varphi(\lambda v)$

Komposition: lin. Abb. ex: $\varphi \circ \varphi : U \to W$ linear für $\varphi : U \to V$, $\psi : V \to W$ linear. $\varphi : V \to W$ linear, $\psi : V \to W$ bijektiv => $\varphi^{-1} : W \to V$ linear, wenn φ bijektiv

Regeln für die Komposition: $\varphi \circ \psi$ lin $\lambda \in \mathbb{K}$: (1) assoziativ (2) id $\circ \varphi = \varphi = \varphi \circ id_U$ (3) distributiv bezgl. + $(\varphi)(\lambda(\psi)\varphi) = (\varphi \circ \psi) = \varphi \circ (\lambda \psi)$ $\mathrm{Aut}(V) = \mathrm{End}(V)^\times$

$\mathrm{End}(V) = \mathrm{Hom}(V, V)$ bildet $\mathbb{K}$-Algebra: $(V, +, \circ, 0, 1, \cdot, id_V)$ Eine $\mathbb{K}$-Algebra ist ein $\mathbb{K}$-VR mit einer zus. bin. Op $\bullet$ (Matrizenmultiplik.) welche assoz., neutr. Elem. hat, Skalarmultiplik. und distributiv + erfüllt

$\mathbb{K}$-Algebra: Bicher VR: $(M, +, \cdot, 0, 1)$ $\mathbb{K}$-Algebra: $(M, +, \cdot, 0, 1, E_n)$ Skalarmultiplik. $\lambda \cdot M = \lambda E_n M$ $M_n(\mathbb{K})$ ist eine $\mathbb{K}$-Algebra Im + (M, ·, E_n) Monoid

$\varphi \to$ Matrix $A \in \mathbb{K}$: $\varphi : \mathbb{K}^n \to \mathbb{K}^m$ und zu φ $A(\varphi) = A$ x für A $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$ x: Spaltenvektor $\in \mathbb{K}^n$ Ax: Spaltenvektor $\in \mathbb{K}^m$ Es gilt: $\varphi_A(e_j) = Ae_j = j$-te Spalte von A

Im φ_A: Es gilt: $v_1, \dots, v_n$ Basis von V, $\varphi : V \to W$ => $\varphi(v_i)$ spannen W auf => $\varphi_A(e_j \in$ Standardvektor von $\mathbb{K}^n$) = Spalten von A spannen W auf => im φ_A = span(Spalten von A)

rank A: $\mathrm{rank}\,A = \dim(\mathrm{Span}(\text{Spalten von } A))$ => Gauß-Algo mit Spalten von A als Zeilen => Basis von im φ_A und Zeilenrang = als = Rang von A

Surjektiv/Injektiv/Bijektiv: $A \in M_{m,n}(\mathbb{K})$ sei $\varphi_A : \mathbb{K}^n \to \mathbb{K}^m$ $\varphi_A = A$ => Surjektiv <=> $\mathrm{rank}\,A = m$ φ_A injektiv <=> $\mathrm{rank}\,A = n$ φ_A bijektiv <=> $n = m$ A rank A = n

ker φ_A: $= \{x \in \mathbb{K}^n : \varphi_A(x) = 0\} = \{x \in \mathbb{K}^n : Ax = 0\}$ homogene Lösungen Fundamentalsystem: Zeilen von A => Matrix => Gauß-Algo => x_i: x_i die Nicht Pivot Variablen

=> dim V_h (Vektor Basis von ker φ_A) zu jeden weitere $x_i = 1$ die anderen $x_j = 0$ => V_1, ... dann für V_k $[\varphi] := A = \begin{pmatrix} \varphi(e_1) & \dots & \varphi(e_n) \end{pmatrix}$ $\varphi(e_i) = \varphi_A(e_i)$ => $\varphi : \mathrm{Hom}(V, W) \to \mathrm{Hom}(\mathbb{K}^n, \mathbb{K}^m)$ $A \to \varphi_A$

Kalmen Grafi
Mobile und sensorgestützte Robotersysteme Ü.
10.Feb.04 Seite 1

Teil0: Robotik: Sense-Plan-Act, Robotics is the intelligent connection of perception and action. Robot = Fronarbeit

Teil1: 3 Robotergebote: 1. Niemanden verletzen (aktiv, passiv) 2. Gehorchen (außer §1) 3. Eigene Existenz schützen (außer §1/2)

Teilsysteme eines autonomen Roboters: Sense: interne/externe Sensoren, Act: Mechanik, Kinematik, Antrieb, Achsregelung

Plan: Programmierung, Steuerung, Energieversorgung, Kommunikation; Kern innerer Ring: mechatronisch-informationstechnische Bausteine

Kinematik: Anzahl, Art, Anordnung der Achsen → Form des Arbeitsraumes. Roboterprogrammierung: Imperative: Achse von Anfahrpunkten. Adaptiv: Einbezug von Sensoren. Komplex: autonome, interaktive Robotersysteme.

3 Robotertypen: Industrieroboter: Aufgaben fest definiert, Objekte/Umwelt fest, automatische Ausführung (automat. explizite Programme). Serviceroboter: Umweltdaten → Weltmodell, aufgabenspezifische Befähigung, Auswertung der Sensorinfo, automatische Bahnplanung. Personal Robot: Kommunikation mit der Umwelt, Verständnis, selbstständige Aktionsauswahl, Lernfähigkeit, aufgabenspezifisch.

Teil2: Industrieroboter, Komponenten: Robotersteuerung: Onboard Computer, Kommunikationsverbindungen (Teach Box / Host), Schnittstelle zu Sensoren (Kamera), aufgaben-Planung, Bahnplanung, Bewegungsregelung. Sensoren: Kameras, Systeme, Fuzzy Sensoren.

Serviceroboter, Komponenten: Personal Robots, Komponenten: Äußerer Ring: Funktionale Steuerung der Roh. Architektur. Mechanik (Arme, Beine, Antriebe) ⇄ Extern: Umwelt, Prozess. Elektronik (Motoren, CPU, Schnittstelle) ⇄ Sensorik: intern, extern. Regelung (Modelle). Planung (Sensorfusion, Welt-Aufbau, Weltmodell). Mobile Roboter: Echtzeit-PC, Sensoren, Motorcontroller, Motoren. Echtzeitaufgaben: Kommunikation, Sensorabfrage, Kollisionsvermeidung, Ansteuerung der Motoren, Positionierung im Umweltmodell.

Mobilität von Robotern: Laufen, Fahren, Kriechen, Schwimmen/Tauchen, Fliegen. Kette offenbar gut.

Sensorstruktur: Elementarsensor (unbearbeitet), Integrierter Sensor (Signalaufbereitung), Intelligenter Sensor (Auswertung).

Interne Sensoren: Position, Geschwindigkeit, Beschleunigung, Inertial Navigations Systeme (Beschleunigung in 3 Raumachsen). Optische Codiere: Info über: Gelenkposition. Prinzip: LED (auf Scheibe codiert), Photodioden: Unterbrechungen werden gemessen. Wahrnehmung (Perception): Umwelt → Sensor, Verhaltenssteuerung. Anwendungen: Gelenkpositionen, Telerobotik, Medizinrobotik, Programmierung, Autonome Sys.

Teil3: Sensoren: aktiv (Laser), passiv (Tast), intern/extern. Anforderungen: Genau, Auflösung, schnell, empfindlich, billig, zuverlässig, fehlerfreie. Kraft/Moment Sensor: 3 el. Erfassung der Kräfte/Momente zw. Endeffektor und Objekt. Dehnungsmessstreifen → Widerstand ändert. Kraft/Moment Sensor: Compliance Sensor: 2 bewegliche Platten für Effektor & Arm, Bei Kraftbeeinflussung: Ausgleichsbewegung, Messung durch Infrarot.

Näherungssensoren: Bestimmende Anwesenheit eines Objekts im spezifischen Abstand. Typen: induktiv, kapazitiv, optisch, akustisch. Induktiv: Spule erzeugt magnetisches Feld, Metallische Objekte entziehen ihm Energie. Kapazitiv: Gleichg. RC-Schwingkreis, Objekt bildet Kondensator. Lichtschranke: Binäre Auswahl, Vorteile weit, berührungslos, kalibrierbar. Echoprinzip im Ultraschall: Emitter & Detektor in einem, Rauschen.

Abstandssensoren: weiter, Erfasst Umwelt geometrisch. Typen: opt. sich (mist Empfangzeit), akustisch (Ultraschall zeitmessung), Radar (Reflektor magnetisch).

Teil4: Triangulation, Stereo, Interferometrie (Überlagerung zweier Lichtwellen, Messung der Verschiebung), Stereoskopie, Ultraschall → 40 kHz. Typische Situation. Probleme. Visuelle Vorverarbeitung: technische Auge, CCD, Bildverarbeitung, Extraktion der Sensor Gemäß Licht. CCD Matrix. 2) Prozessor. Bildverarbeitung: Sensor → Bildaufnahme → Vorverarbeitung → Segmentierung (Partitionierung) → Beschreibung (Merkmalextraktion) → Objekterkennung → Bildverstehen. Bildverstehen/Semantik: Ungelöstes Turing Test: PC testet, ob A = Mensch oder PC, PCs schützen: Unlesbarer Text erkennen, CAPTCHA (Bild Semantik).

Teil5: Minerva: Aufbau der Umgebungskarte durch Hereinfahren, Sensorauswertung, Deckenbild. Konfigurationsraum $\in \mathbb{R}^2(x,y,\text{Richtung})$. Sensorfehler notwendig. Bahnplanung: Schnittlinie schließt Abstände Sensorgenauigkeit. Sensor-Umwelt ← Aktuatoren, Hardwareschnittstelle, Navigation (Selbstlokalisierung, Bahnplan, Kollision) Wissen (Umgebungskarte). Robot.

Teil6: Sensorfusion & Weltmodellierung. Ebenen der Sensorik: physikalische Ebene (ADC), Messungsebene (digit. Signale → Werte, Gefahrenerkennung/Kollision, Wahrnehmungsebene (Aufgabenmodell, Weltzustand, Handlungskonsequenzen). Modell-Ebene (Umwelt, Roboter, Sensorfusion, Datenkomprimierung).

Sensorfusion → einige Messungen sind ungenau, Zuverlässigkeit, schnell, Erkennung unterschiedlicher Materials Probs: Messen unterschiedlicher Sensorwerte, Tatsächlich Rauschen. Komponenten: (Sensormodelle, Fusionsmethoden (numerisch, geometrisch) (Entscheidung, Berechnung, Führung), Weltmodell (Kriterien, Abstraktionsniveau, Operationsniveau, Umweltbed. des R., Anwendungsgebiet des R., Informationsgehalt (Pfade, Freiraum, Objekte, geometrische Modelle).

Teil7: Aufgabenplanung, semantische Planung, topologische Wegplanung, geometrische Bahnplanung, Stat. kinemat. Trajektorienplanung, dyn. kinemat. Trajektorienplanung, dynamische Regelung. Weltmodell: Externe Sensoren, Aufgabe und Welt, Realisierung, Detektion, Konsequenz, Welt, Aufgabe, Strategie.

Aufgaben: Aufgaben + Steuerung → unterschiedl. Ebenen. Programmierung: Anfahrpunkte, Sensorbezug, autonom. Programmierebenen: Mission → Aufgabe → Handlung → Roboter, Gelenk → Physikalisch, Elementar, Roboterprogramm. Kinematisch Probleme bei kontr. Bahnvorgabe: Unterbrechung zwischenpunkte, Hohe Gelenkgeschwindigkeit, Unterschiedliche Gelenkpositionen möglich. Industrieroboter, Programmierung: Teach In, Text, Anfahrpunkte, Grafisch (CAD).

Aufgaben/Handlungsebene: User beschreibt Rob bildet Auftrag in Roboter, befüllt die über mehrere Stufen, entsprechende Sprachen.

Planen: Zerlegung der Aufgabe in einzelne Tasks (Mission), Analyse der Task nach Bedingungen, zerlegen in Elementaroperationen (Task), Symbolische Parameter → memorierte Werte, Sprachen vom Sensoren (Action). Auftragsverarbeitung: Auftragseingang → semantische Planung → topol. Weg → geometr. Bahnpl. → Kollisionsvermeidung.

Teil8: Navigation, Verhaltenssteuerung, Urteilsfähigkeit: Kartographierung, Selbstlokalisierung, Bahnplanung & -auswahl, Bahnausführung, Echtzeitsteuerung, Kollisionsvermeidung. Steuerungsarchitekturen: Hierarchisch (Sense → Plan → Act) (aufwändig), Reaktiv (Sense ⇔ Act) (schnell, robust), Hybrid deliberativ (Planung) reaktiv (Sense ⇔ Plan ⇔ Act) (Pfad planen, beste Unterteilung einer Aufgabe, Auswahl geeigneter Verhalten, zur Lösung der Teilaufgaben, Anzahl Verhalten, reaktiv). Szenario: Haushaltsroboter, kann die Einsatzumgebung lernen. Lernverfahren notwendig für Autonome: lernen von Karten, kinem. Polynom. Modelle, Verfahren.

Teil9: Robotersysteme in Fahrzeugführung 1) Fahrassistenz (Informierend, Servosystem, Automatisch intervenierend, Automatisch agierend) 3 Ebenen Modell: Planung, Bahnführung, Stabilisierung. 2) Elektronische Deichsel: Hinterer LKW durch vorderen gesteuert: Entfernungsmessung, Kameras, Radar, Fuzzy Sensoren, Steer/Drive by wire, Funkkommunikation. 3) Automatisierte Schnellstraße: Auto + Straße (Signale, Streckenführung): intelligent: Sichere Erkennung von Straßen & Spurverlauf. Autonome Fahrzeuge: a) öffentliche Straßen (autonome Fahrmodi, Verkehrsspezifikation, Zustandserhaltungsgenauigkeit). b) Off-Road → Weltraumfahrzeuge. Sim. der Fahrverhaltens: genauig, schnelle, direkt. Teilmodelle: Fahrzeug Fahrer, Fahrbahn (Selbstrotation, Beschleunigung des Fahrzeugschwerpunkts, Stabilisierung).

Teil10: 6/4 Beinige: Aufwendig, aber universeller. Statisch stabil ≥ 3 Beine auf Boden, Regulierte Gangart: Wiederholte Muster. 6-Beinig: Bein steuerung autonom. Kinematik: Stabilisierung > 4-Beinig. Dynamisch stabiles Laufen notwendig, Hindernisüberwindung vor Beinkontakt notwendig, zentrale Steuerung. Reflexphase: mind. 3 Freiheitsgrade notwendig. **Teil11:** 2-Beinige: Menschl. Hüftgelenk = 2 = Beingelenk. PAtron & LPatron, CARBOT (Mechanik-Ansteuerung). Verhalten der Wirbelsäule unterstützt Fuß ermöglicht Dehnmuster. [Passiv dynamisches Laufen: angetrieben durch schiefe Ebene]. Fazit 6 Drehgelenke Bein. Stabilitätsregelung, inertial Pendel, Zero Moment Point, Schalter, Laufen: Steuerung, Antriebe + Sensoren, Ansätze zum Kennenlernen, Laufen, Kern: passiv, Aktiv (Arm Beine, Kopf, Arbeitsraum, Intelligent Pattern, Aufgaben, Computer, Terminal, Kommunikation, Schnittstelle, Neuronale Netzen, Reizen, Motion, Intelligenz).

Kalmen Grafi
Comm Net Abk.
31.7.2005 Seite 1

WWW - World Wide Web	WWA - World Wide Access
WPP - World Wide Peer-To-Peer	WWSC - World Wide Seamless Communication
ISO - International Standards Organisation	OSI - Open System Interconnection
ES - End system	DTE - Data Terminal Equipment / DSE - Data Switching Exchange
DCE - Data Circuit terminating System	LAN - Local Area Network 70 m–1 km
MAN - Metropolitan Area Network 10 km	WAN - Wide Area Network 10 km+
ARPA - Advanced Research Projects Agency	DoD - Department of Defense ab 1969
UUCP - UNIX to UNIX Copy	dfn - Deutsches Forschungsnetz
MAC - Medium Access Control	IP - Internet Protocol
SAP - Service Access Point	SDU - Service Data Unit
PCI - Protocol Control Information	PDU - Protocol Data Unit = PCI + SDU
IDU - Interface Data Unit = ICI + SDU	ICI - Interface Control Information
FCS - Frame Check Sequence	ARP - Address Resolution Protocol
FTP - File Transfer Protocol	HTTP - Hypertext Transfer Protocol
ICMP - Internet Control Message Protocol	LLC - Logical Link Control
MAC - Media Access Control	NFS - Network File System
SMTP - Simple Mail Transfer Protocol	Telnet - Remote Login Protocol
UDP - User Datagram Protocol	TCP - Transmission Control Protocol
SCTP - Stream Control Transmission Protocol	ANSI - American National Standards Institute
WD - Working Draft / DP - Draft Proposal	DIS - Draft International Standard
IS - International Standard	ITU - International Telecommunication Union
CCITT - Comite Consultatif International de Telegraphie et Telephonique	
ECMA - European Computer Manufacturers Assoc.	IEEE - Institute of Electrical and Electronics Engineers
IAB - Internet Architecture Board	IETF - Internet Engineering Taskforce
RFC - Requests for Comments	GW - Gateway

BPS - Bits per Second	UART: Universal asynchronous receiver and transmitter
NRZ-L - Non Return to zero-level	RZ: Return to Zero
FDM - Frequency Division Multiplexing	TDM: Time Division Multiplexing
DLE - Data Link Escape	CRC - Cyclic Redundancy Code (CRC-CCITT) $x^{16}+x^{12}+x^5+1$
DSAP - Data Link (Layer) Service Access Point	PAR - Positive Acknowledgement with Retransmit
ARQ - Automatic Repeat reQuest	SDLC - Synchronous Data Link Control
SNA - System Network Architecture	ADCCP - Advanced Data Communication Control Procedure
HDLC - High-Level Data Link Control	LAP - Link Access Procedure
LAPB - Link Access Procedure, Balanced	LLC - Logical Link Control
FCS - Frame Check Sequence / Sum	P/F - Poll / Final
DISC - Disconnect	SNRM - Set Normal Response Mode
FRMR - Frame Reject	UA - Unnumbered Acknowledgement
SLIP - Serial Line Internet Protocol	PPP - Point 2 Point Protocol
LCP - Link Control Protocol	NCP - Network Control Protocol

LAN - Local Area Network	MAC - Medium Access Control
EOT - End of Transmission	TDMA - Time Division Multiple Access
CSMA - Carrier Sence Multiple Access	CD - Collision Detection
SD - Starting Delimiter / AC - Access Control	FC - Frame Control / ED - End Delimiter / FS - Frame Status

PMD - Physical Media Dependent	WIS - WAN Interface Sublayer
WWDM - Wide Wave Division Multiplexing	IVD - Integrated Voice Data
UTP - Unshielded Twisted Pair	HIPPI - High Performance Parallel Interface

MAN - Metropolitan Area Network DQDB - Distributed Queue Dual Bus MST - Multiplexed Slot and Token Ring
QPSX - Queued Packet Switch FDDI - Fiber Distributed Data Interface CDDI - Copper Distributed Data Interface
TRT - Token Rotation Time TTRT - Target Token Rotation Time THT - Token Holding Time

$$\frac{}{\Gamma, A, \Delta \vdash A}\,(Ax)$$

$$\frac{\Gamma, A, B, \Delta \vdash C}{\Gamma, B, A, \Delta \vdash C}\,(X) \qquad \frac{\Gamma, A, A, \Delta \vdash C}{\Gamma, A, \Delta \vdash C}\,(C)$$

$$\frac{\Gamma \vdash A \quad \Delta \vdash B}{\Gamma, \Delta \vdash A \wedge B}\,(\wedge\mathcal{I}) \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\,(\wedge\mathcal{E}) \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}\,(\wedge\mathcal{E})$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}\,(\vee\mathcal{I}) \qquad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}\,(\vee\mathcal{I}) \qquad \frac{\Gamma \vdash A \vee B \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash C}\,(\vee\mathcal{E})$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\,(\to\mathcal{I}) \qquad \frac{\Gamma \vdash A \quad \Delta \vdash A \to B}{\Gamma, \Delta \vdash B}\,(\to\mathcal{E})$$

$$\frac{\Gamma \vdash \neg\neg A}{\Gamma \vdash A}\,(\neg\neg) \qquad \frac{\Gamma \vdash \bot}{\Gamma \vdash A}\,(\bot\mathcal{E})$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash \forall x.A}\,(\forall\mathcal{I}),\ x \notin FV(\Gamma) \qquad \frac{\Gamma \vdash \forall x.A}{\Gamma \vdash A[t/x]}\,(\forall\mathcal{E})$$

$$\frac{\Gamma \vdash A[t/x]}{\Gamma \vdash \exists x.A}\,(\exists\mathcal{I}) \qquad \frac{\Gamma \vdash \exists x.A \quad \Gamma, A \vdash C}{\Gamma \vdash C}\,(\exists\mathcal{E}),\ x \notin FV(\Gamma, C)$$

$$\frac{}{\Gamma \vdash x = x}\,(R) \qquad \frac{\Gamma \vdash s = t \quad \Delta \vdash A[s/x]}{\Gamma, \Delta \vdash A[t/x]}\,(Eq)$$

$$\frac{\Gamma, A \vdash \bot}{\Gamma \vdash \neg A}\,(\neg\mathcal{I}) \qquad \frac{\Gamma \vdash A \quad \Delta \vdash \neg A}{\Gamma, \Delta \vdash \bot}\,(\neg\mathcal{E})$$

$$\frac{\Gamma \vdash A}{\Gamma, B \vdash A}\,(W) \qquad \frac{\Gamma \vdash A \quad \Delta, A \vdash B}{\Gamma, \Delta \vdash B}\,(Cut)$$

$$\frac{}{\Gamma \vdash A \vee \neg A}\,(TND) \qquad \frac{\Gamma, \neg A \vdash \bot}{\Gamma \vdash A}\,(RAA)$$

$$\frac{}{\Gamma \vdash t = t}\,(R') \qquad \frac{\Gamma \vdash s = t}{\Gamma \vdash t = s}\,(S) \qquad \frac{\Gamma \vdash s = t \quad \Delta \vdash t = u}{\Gamma, \Delta \vdash s = u}\,(T)$$

Kalman Graph

23.2.03

RT Kommuz

Grundlagen $U = R \cdot I$ (V, Ω, A) Knot. Knotenregel: $\sum I_{ein} = \sum I_{aus}$ Maschenregel: $\sum U_{ec} = 0$

$L = \frac{Vs}{A} = \frac{U t}{I}$ $C = \frac{As}{V} = \frac{I(t)}{U_c}$ $(\tau = R \cdot C)$

Widerstände	Kondensator	Induktivität $\tau = L/R$
Reihenschaltung $R_{ges} = \sum R_i$	$\frac{1}{C_{ges}} = \sum \frac{1}{C_i}$	$L_{ges} = \sum L_i$
Parallelschaltung $\frac{1}{R_{ges}} = \sum \frac{1}{R_i}$	$C_{ges} = \sum C_i$	$\frac{1}{L_{ges}} = \sum \frac{1}{L_i}$
Spannungsteiler U_i: $U_{R_i} = U_0 \cdot \frac{R_i}{R_{ges}}$	$U_{C_i}(t) = \frac{C_{ges}}{C_i} \cdot U_{ges}(t)$	$U_{L_{ges}} = I_L(t) \cdot \sum L_i$; $U_{L_i} = \frac{L_i}{L_{ges}} \cdot U_{ges}$
Stromteiler I_i: $I_{R_i} = \frac{U_{R_i}}{R_i} = I_0 \cdot \frac{R_{ges}}{R_i}$	$I_{C_i} = \frac{C_i}{C_{ges}} \cdot I_{ges}$	$I_{L_{ges}} = U_L / L_{ges}$; $I_{L_i} = \frac{L_{ges}}{L_i} \cdot I_{ges}$

1-Speicher-Formel

$x(t) = x_{end} - (x_{end} - x_{anf}) \cdot e^{-\frac{\Delta t}{\tau}}$ $\tau = R \cdot C$, $\tau = L/R$

Feldaufbau / Entladen / Feldabbau

Kondensator:
$U_C = U_{end} - (U_{end} - U_{anf}) \cdot e^{-\frac{t}{\tau}}$
$I_C = \frac{U_{end} - U_{anf}}{R} \cdot e^{-\frac{t}{\tau}}$

Induktivität:
$U_L = -R \cdot (I_{anf} - I_{end}) \cdot e^{-\frac{t}{\tau}}$
$I_L = I_{end} - (I_{end} - I_{anf}) \cdot e^{-\frac{t}{\tau}}$

Induktivität: kein Stromsprung $I(t=0) = 0$ $I(t=\infty) = \max$ $U(t=0) = \max$ $U(t=\infty) = 0$ Kondensator: kein Spannungssprung $U(t=0) = U(t=\infty) = \max$

Ersatzquelle: R_i: bzgl. Leistung U_L R_{ges} bzgl. U_0 berechnen! I_L: Klemmenkurzschlußstrom, Strom durch Klemmen $= I_K$

Superposition: Ideale Quellen werden unabhängig von einander: 1) Alle bis auf eine Quelle durch Innenwiderstand ersetzen 2) Betreffende Beitrag der Quelle 3) Summieren

Maschenstrom: 1) Vollständiger Baum 2) Maschenströme aufzeichnen 3) Maschengleichung aufstellen 4) Gleichungssystem lösen 5) Berechnen mit $I_{Zweig} = I_{Masch}$ addieren

Maschengleichung $(R) \cdot (I) = (U)$ I: enthält Maschenströme U: enthält alle Spannungsquellen der Masche: Vorzeichen ⚠ $\sum R I = \sum U$

Widerstandsmatrix: Symmetrisch! Diagonale: R_S in der Masche, Nicht Diagonale: Kopplungswiderstände: R_S auch in anderen Maschen

Vermeiden: $+ I_a \uparrow \uparrow I_b$ $- I_a \uparrow \downarrow I_b$

Diode: gesperrt! leitend ϕU_a Vorgehen: 1) Annahme 2) Ersatzschaltbild ⇒ Berechnung von U_D ⇒ Bestätigung / Annahme

Bipolartransistor $I_E = I_B + I_C$ $I_C = B_n \cdot I_B = \frac{A_n}{1 - A_n}$ $I_E = (B_n + 1) \cdot I_B$

Zustände $I_E = 0$: 1) Gesperrt 2) Normaldaktiv 3) gesättigt 4) invers aktiv

Grundschaltungen: Emitterschaltung BGS, KGS, BGS

Grafische Arbeitspunkt Bestimmung: $U_{Ein} = U_{RE} + U_{BE}$ $U_{RE} = I_B \cdot R_E$ ⇒ $U_{Ein} = I_B \cdot R_E + U_{BE}$ U_{Ein} I_B bekannt

MOS-Fets G=1 leitet, wenn G=1 G=0 leitet, wenn G=0 CMOS

Operationsverstärker: U_B Betriebsspannung U_d Differenzspannung $U_d = U_+ - U_-$

Ideal, OP

Schmitt Trigger

I pos übersteuert: $U_a = U_B^+$ wenn $U_d > 0$ oder $\frac{U_B^+}{A_d} \le U_d \le \frac{U_B^-}{A_d}$

II OP aktiv $U_a = A_d \cdot U_d$ für $U_d = 0$ oder

III neg $U_a = U_B^-$ für $U_d < \frac{U_B^-}{A_d}$ oder 0

Falls positiv rückgekoppelt: Schmitt Trigger

Falls negativ rückgekoppelt

Allgemeines Vorgehen Bestimme U^{ein} und U^{ar} 1) Schaltung ansehen 2) a) pos → aktiv: pos: $U_{aus} = U_B^+$ 2) b) neg → aktiv: aktiv $U_+ = U_-$ neg: $U_{aus} = U_B^-$ ⇒ U_{ein}

Beispiel: Bipolartransistor

$R_E = 1\,k\Omega$, $R_C = 10\,k\Omega$, $U_{CC} = 10\,V$

$A_n = 0{,}99$, $A_i = 0{,}90$

$B_n = \frac{I_C}{I_B}$ $A_B = \frac{I_C}{I_B + I_C}$ $B_n = \frac{I_C}{I_B} = \frac{I_C \cdot (I_B + I_C)}{(I_B + I_C) \cdot I_B} = A_n \cdot \frac{(I_B + I_C)}{I_B} = A_n \cdot \left(1 + \frac{I_C}{I_B}\right) = A_n$

$B_n = A_n \cdot A_n \cdot B_n$ $B_n = \frac{A_n}{1 - A_n} = \frac{0{,}99}{0{,}01} = 99$

Bestimme B_n, U_{ein} $U_C = U_{CC} - I_C \cdot R_C = U_{ein} + U_{BC}$ $U_E = I_E \cdot R_E = U_{ein} - U_{BE}$

$U_{BE} = 0{,}7\,V$ $I_E^{**} = A_i \cdot I_E^*$ $I_E^{**} = I_E^* + U_{ein}$ aktiv, gesättigt $A_B = \frac{I_C}{I_B + I_C} = \frac{I_E}{I_E}$ $U_{CC} - R_C \cdot I_C = U_{ein} - U_{BC}$ $R_E \cdot I_E = U_{ein} - U_{BE}$

$R_E \cdot I_E = U_{ein} - U_{BE}$ $(U_{CC} + U_{BC}) - R_C \cdot I_C = U_{ein} = (U_{CC} + U_{BC}) - A_n \cdot R_C \cdot I_E$

$I_E = \frac{U_{ein} - U_{BE}}{R_E}$ $I_E = \frac{(U_{CC} + U_{BC}) - U_{ein}}{A_B \cdot R_C}$ $R_E \cdot (U_{CC} + U_{BC}) - R_E \cdot U_{ein} = A_n \cdot R_C \cdot U_{ein} - A_n \cdot R_C \cdot U_{BE}$

$U_{ein} \cdot (A_n \cdot R_C + R_E) = R_E \cdot (U_{CC} + U_{BC}) + A_n \cdot R_C \cdot U_{BE}$ $U_{ein} = \frac{R_E \cdot (U_{CC} + U_{BC}) + A_n \cdot R_C \cdot U_{BE}}{(A_n \cdot R_C + R_E)}$

$U_C = U_{ein} - U_{BC} = U_{ein} - \frac{(A_n \cdot R_C + R_E) \cdot U_{BC}}{(A_n \cdot R_C + R_E)} = \frac{R_E \cdot (U_{CC} + A_n \cdot R_C \cdot (A_n \cdot R_C + R_E) \cdot U_{BC})}{(A_n \cdot R_C + R_E)}$

Kalman Grafik Robotik 1 SS 2005

9 NAVIGATION MOBILER ROBOTER

9.1 Physikalische Modellierung a) Odometrie

b) Inertial Sys.

c) Aktive Landmarken

d) Erkennung

e) Sensorauswertung

f) Modellerkennung

g) Lokalisierung

h) Monte-Carlo Lokalisierung

Selbstlokalisierung: Best. der Aufenthaltsortes relativ zu einem Ausgangspunkt / absolut bezogen auf ein Weltsystem

Messungsungenauigkeiten: Lange aus Verhältnisbildung über den Raum

Klassifikation von Verfahren: Karte / Topologie / Topologisch / Metrisch / Topologisch / Hybrid / Anzahl der Hypothesen: einer / mehrere / Aufgabenstellung: Lokale Bestimmung der Lage / Globale Lokalisierung / Entführungsproblem / Verfahren: deterministisch oder zufällig / Verwendete Sensoren

Topologische Karte: Graph, Knoten, Kanten

Lokalisation: Koppelnavigation: Steuerung, Fehler, Odometrie

Kalman Filter: Mittelwertfunktion, Kovarianzfunktion, Messung, Kalman Filter

Kinematik: Annahme von Geschwindigkeit, Winkelgeschwindigkeit

Time-Update: Schätze Folgezustand, Korrigiere Schätzung

Gewichtung: a-priori Schätzung, Kalman-Gain, a-posteriori Schätzung

Messungsupdate: Varianz a-priori Schätzung, Kalman-Gain, Messung, Zustandsgleichung

Nicht modellierbare Effekte: Bewegungen von Menschen, Fehlerhafte Messungen

Normalverteilung: Wahrscheinlichkeit keine Abbildung von Hindernissen

Lösung durch Hindernisse bewegt werden

Multiple Hypothesen: Hypothesen, Kommunikation, Lokalisierung, Sensorik, Hypothesen, Messung

Time-Update auf alle Hypothesen anwenden

Konkurrenzhypothesen, Hypothesen, Lokalisierung

Distinktive Merkmale: Gemeinsamkeiten, Unterschiede, Schätzung der Verteilung

Monte Carlo: Approx. der Wahrscheinlichkeitsverteilung durch Partikel, Lokalisierung, Lage des Roboters, Partikel, Bewegung, Messung, Gewichtung der Partikel, Bewegung, Lage der Partikel, Gewichtung, Messung, Gewichtung der Partikel, Vorgehensweise: Initialisierung, Bewegung und Messung

10 ARCHITEKTUR AUTONOMER ROBOTER

Paradigma: Annahmen und Verfahren zur Problemlösung

Hierarchisch: Sense → Plan → Act

Reaktives Paradigma: Sense → Act

Hybrid: Sensor → Act

Subsumption Architektur: Verhalten, Schichten, Aufgabe, Sensoren, Robotertyp

Kompetenzebenen: 0: Kollisionsvermeidung, 1: Ziellosen Herumfahren, 2: Umwelt folgen, 3: Karte erstellen, 4: Karten planen

Kommunikation: Kommunikation, Kommunikation, Kommunikation

Autonomous Flying Robots: TUB Rob Blimp

6. Roboterregelung 7. Bahnplanung und autonome Robotersysteme (11. Simulation in der Robotik)

6 Roboterregelung

Antriebssysteme: Pneumatisch (Gasdruck), Hydraulisch (Inkompressibel), Elektrisch Motoren (DC, Bürsten), Schrittmotor

Pneumatik: Gasdruck, Ventil, Regelventil, Positionsregelung

Hydraulik: Inkompressibel, Hydraulikpumpe, Temperaturregelung

Elektrisch: Schnell, Präzise, Regelung, Regelungssystem, Motor

DC Motor: Bürstenmotor

Gelenke: Motor schnell, geringes Drehmoment, Gewichtsreduzierung, Drehmoment, Gewichtsreduzierung

Kleinere Gewinde: Kugelgewinde, Gewinde, Steigung, Steigung, Gewinde, Kugelgewinde, Gewinde, Kugelgewinde

Radarschnecke: Spielgetriebe, Zahnrad, Getriebe, Drehmoment, Kompakt, Kein Leistung, Kein Drehmoment, Kompakt, Kein Leistung

Harmonic Drive Getriebe: Kompakt, Kein Spiel, Kein Drehmoment, Kompakt, Kein Leistung

Sensor: Intern, Extern, Aktiv, Passiv, Sensor, Signale, Sensor, Sensor, Sensor, Sensor

Messgrößen: Position, Geschw. (Geschw. Sensor), Kraft (Kraft Sensor), Sensor, Sensor, Sensor, Sensor

Optisch: Absolut, Inkrementell, Stopp, Sensor, Sensor, Sensor, Sensor

Optische Codes: Absolut, Inkrementell, Gray-Code, Sensor, Sensor, Sensor, Sensor

Rotationsgeber: Sensor, Sensor, Sensor, Sensor, Sensor, Sensor, Sensor

Beschleunigungssensor: Silizium Beschl. Sensor, Trägheitsmomente, Sensor, Sensor, Sensor, Sensor

Inertial Sensor: Gyroskop, Sensor, Sensor, Sensor, Sensor, Sensor, Sensor

Kraft Momente Sensor: Sensor, Sensor, Sensor, Sensor, Sensor, Sensor, Sensor

Regelung: Kontrolle, Steuerung, Regelung, Steuerung, Regelung, Regelung, Regel

Kalman Graffi
Netzwerksicherheit Z.
6.2.2005 Seite 1

Netzwerksicherheit

(ISBN 3827413737 Autor: Stephen Northcutt)

Kapitel 1: Einführung

Historie, ARPANET, IMP, IP, TCP, Morris-Wurm

ARPANET (1967): Ziel: Vernetzung der teuren DoD/ARPA Mainframes zur besseren Auslastung
Verbindungsorientiertes Netz: (Telefon-Intelligenz im Netz, Hierarchisch, Bandbreitenanforderungen vorhersagen zu stark)
Paketbasierte Netze: Daten → Datagramme / Pakete, unabhängig durchs Netz geleitet, Knoten kennt nur next-hop, Intelligenz im Endgerät
Topologien: zentral, dezentral, distributed

ARPANET: Mainframes ohne Netzwerkinterface → IMPs (Interface Message Prozessor), von MF handshaken, Addon
IMPs kommunizieren untereinander über Telefonstandleitung (50kBit/s) (Honeywell)
IMP-Mainframe Kommunikation über „Host Special Interface", muss an MF angepasst werden
NetControl Protocol NCP, Paketbasierte Komm, NCP: Ziel, Absender, message id, payload, kurze Pakete, der Kab. fehlerhaft
ARPANET-Grenzen: max. 64 IMPs mit 16 Host = 256 Host maximal!
Hauptunterschied zwischen ARPANET und Snet: Verlagerung der Weiterverkehrskomm. von spezialisierten Zwischenknoten auf einzelne Hosts.

TCP/IP: Ziel: Internetting, d.h. verschiedene Netze verknüpfen, Best Effort-Basis, keine Informationen über Datenflüsse (Payload) → resend falls Paketverlust, => sliding window + ack. TCP
TCP/IP kompakt implementiert, funkte schnell zu interoperablen Systemen Transmission Control Protocol
Durchbruch von TCP/IP: Integration in UNIX. Dann Integration in 4.2BSD → kostenlos → in allen Unix.
sehr liberale Lizenzpolitik => TCP/IP-Code in allen weiteren UNIX-Impl.

Sicherheitsaspekte: Ziel war: möglichst einfach, pragmatisch, keine Verbindungsverschlüsselung
Nachträgliche Einbringung von Sicherheitsmechanismen => fehlerhaften oder ineffizient

Morris-Wurm: 1988 6000/60000 Knoten befallen 3 potentielle Angriffswege: Shell-Zugang, SMTP (sendmail), finger
Shell-Zugang: rsh an „.rhosts und /etc/hosts.equiv" mit gegenseitiges Vertrauen (kein Passwd) aus
sendmail: Debug-Feature beliebige Shellbefehle mit sendmail-Rechten ausführen (meist root) standardmäßig an.
finger: Buffer Overrun Morriswurm = Infektionsvektor und Wurm Hauptprogrammen
Wurm Hauptprogramm: Erschweren der Erkennung des Wurms

Brückenkopfidee etabliert: näheres Netzwerk besser auszuspähen, häufig gegenseitiges Vertrauen
Problem: Betriebssystem Monokultur.

Kapitel 2: Risikoanalysen und Sicherheitspolitiken

Risikoanalyse (Erstellung), Sicherheitspolitiken, positive Fragestellung

Methodik der Risikoanalyse: Identifikation der zu schützenden Werte → Id. der Bedrohungen und ihre Wahrscheinlichkeit
Bewertung der Schäden (Bedrohungsschaden * Wahrsch.) → Priorisierung von Gegenmaßnahmen → Auswahl von Gegenmaßnahmen
→ Umsetzung + Restrisiko

zu schützende Werte: Vertrauliche Informationen, Verfügbarkeit von Ress. und Diensten, Integrität von Info., Ausrüstung, Personal
Bedrohungswahrscheinlichkeitsbewertung, Folgenabschätzung: qualitativ (Coming-hoch), Risiko = Bedrohungslast × Folgeb.

Begriffe der Fehleranalyse: Fehler-Mechanismus (Ursache), Fehlermodus (fehlerhafte Teilkomponente), Fehlereffekt (auf das Syst.)
Fehlerbäume: Wolke Top Event aus und untersuchen unmittelbare Ursachen für das Eintreten
AND, OR, Main Event □, Grund Event ◇ (nicht tief), Grund Event 2 ○ (abbrechen), Fallbaum, Switch, Inhibit

Sicherheitspolitik: rechtlich notwendig ab best. Größe. Problem: Balance zwischen Abstraktion und Umsetzbarkeit
Inhalt: Definition der Zielsetzung, Umfang, (Aussagen der Geschäftsführung dazu), Erläuterungen der Sicherheitspolitik, Richtlinien, Standards (→ Verträge, gesetze, Mitarbeiterschulung), Malware vorbeugung / Viren
Verfügbarkeitsplanung, Konsequenzen und Sanktionen bei Verletzung der Politik, Verantwortlich bei

Organisation (Sicherheit): Verantwortungen in der IT-Sec festlegen. Festlegen von Rollen, Verantwortung einzelner Stellen / Personen,
Methoden, Verfahren (Risiko Anal.), Schulungen, Integration neuer IT-Verfahren, Bewertung der techn. Realisierung der Sicherheitsrichtlinien

Kalman Graffi
Netzwerksicherheit.
6.2.2005 Seite 2

Verfahren für Umsetzung, Beurteilung der Wirksamkeit der Sec.pol., Vorgehen für Aufnahme und Analyse von vorsätzlichen Angriffen
Klassifizierungsmechanismen: Verknüpfen von identifizierten Werten mit Verantwortlichen:
Werte-Typen: Informations-Werte, Software, Physikalische Werte, Externe Dienstleistungen, also Info, SW, HW, Versorgung

Personal: IT-Sec-Richtlinien Bestandteil des Arbeitsvertrages, Admins gegenüber misstrauisch: Angaben, Referenzen
Finanzen (abundzu) checken, Alle Mitarbeiter einbeziehen: Regeln begründen, Schulen!! → Erkennen, Vermeiden
Physikalische Sicherheit: Sicherheitszonen räumlich trennen, Kabel schützen, Plänen für Versorgungsausfälle, Regelungen für Wartungsmitarbeiter (-externe)
Laufender Betrieb: Wichtiges dokumentieren, Änderungen über Revisionsmechanismus einfügen (d.h. evtl undo)
Betriebsvorgänge (zu schützen): pers. bezogene oder legale Daten, Wartungsarbeiten → Abhängigkeiten, zeitfenster,
Fehlerszenarien, Kontakte in Notfällen. Änderungen am IT-Sys kontrollieren. Alles Protokollieren!
Verfolgung von Änderungen am System: Wer, Was, Auswirkungen, wer erlaubt, verkündung der Änd., Notfallplan → verwaltung
Vorgehen bei Auftreten, Sec-probs: Art bestimmen, (Analyse der Ursache, Wiederholung verhindern, evtl. Juristen zeugen)
Sichern der Protokoll- und Revisionsdaten, Betrieb wieder aufnehmen (Forensik ↔ weiter laufen lassen)

Zugriffskontrollmechanismen: need-to know, all deny, nur falls notwendig Rechte vergeben, 1 Konto = 1 User
Proto tollieren!, zeitlich begrenzen, mobile Geräte (Notebooks...) sind unkontrollierbar

Rechtliche Aspekte: KonTraG: Gesetz zur Kontrolle und Transparenz im Unternehmensbereich.
=> Gesetzliche Verpflichtung für Unternehmer „ein Überwachungssys zur Frühzeitigkeitserkennung existenzgefährdender Entwicklungen" einzurichten. => rechtliche Verpflichtung zum Risikomanagment. → AGs, GmbHs
Umsetzungsphasen modell vorgegeben, unabh. Wirtschaftsprüfer prüft Umsetzung

Bundesdatenschutzgesetz: für öffentliche Stellen und Private, personenbezogene Daten nur erhebbar, falls für Vertragserfüllung notwendig

TKG: Telekommunikationsgesetz (für Betreiber), TDG: Teledienstgesetz TDDSG: Teledienstdatenschutzgesetz
TDDSG: z.B. private Inet-Nutzung im Betrieb darf nicht protokolliert werden. Betr.verf.G Betriebsverfassungsgesetz
Betriebsrat hat Akteneinsicht, soll Durchsetzung der Arbeitnehmerrechte kontrollieren

Kapitel 3: Grundlagen des Internet Protocol

IP, IP-Header (-Optionen), IP-Adr., -Routing, ICMP, UDP, TCP

OSI: Application (Protokollanfragen (NFS)) – [Presentation (Problembehandl. von heterogenen Systemen)] – Session (Sync von Partnerinhalt tragend) – Transport (abstrahierte P2P-Verbindung) – Network (NetFlow Routenberechnung) – Data Link (Checksum gegen Übertragungsfehler) – [Physical (Elektrotechnik)]
Gateway (inter-net Kopplung), Router (mit Routingtabelle), Bridge/Switch (Mac), Repeater, Hub

IP (Internet Protocol): Paketbasiert, best effort Protokoll, Pakete variable Länge, Routing der Pakete unabhängig von einander
ICMP (Inet Control Message Prot) → Info- und Fehlerübermittlung) IGMP: Internet Group Management Prot: → zugehörigkeit zu multicast g.
UDP (User Datagram Protocol): verbindungslos, best-effort TCP (Transmission Control Protocol): Verbindungsorientiert, gesichertes Kanal gegen Paketverlust, Sequenzierung (Reihenfolge) – und Prüfsummenfehler.

IP Header (mit Bits): Version (4), Header-Länge (in Byte (4)), Type of Service TOS (8) (Precedence (3), was kann Router zu mir sagen), Gesamtlänge (16), Ident. (16),
Dont fragment (1), More Fragment (1), Fragment-Offset (13) (Ursprung der alt. Flag im Gesamt Datagramm), Time to Live TTL (8) (Hop-Protokoll (8), Header Prüfsumme (16), Source Adr (32), Dest Adr. (32), OPTIONEN, Payload (Nutzdaten)

Optionen: Sicherheits-Optionen (was zu tun bei Fehlern), Routen-Katalogisierung (Router auf dem Weg tragen sich ein), Zeitstempel (max 9)
(Router tragen ihr timestamp ein), Loose Source Routing (Liste der Router, die auf dem Weg liegen müssen)
Strict Source Routing (nur diese Router dürfen verwendet werden)

Fragmentierung: Ethernet 576B – ~1522 Bytes (MTU) => evtl Pakete fragmentieren, Fragmente (bis auf letzt) MF-Flag
mit Fragment Offset, eindeutig wegen Ident, Source, Dest, Protocol, alle Fragmente mit selber Ident MF=0 => Ende

Aufbau der Vorlesung

Block 1: IoT-Begriffe und Use Cases

Begriffsklärung im IoT-Umfeld, Abgrenzung verwandter Konzepte, systematische Analyse von Anwendungsszenarien aus verschiedenen Domänen.

Block 2: Sensoren und Einplatinenmaschinen

Hardware-Trade-offs, Mikrocontroller vs. Single-Board-Computer, platinennahe Schnittstellen und typische Entwicklungsplattformen.

Block 3: Infrastruktur, Systeme und MING-Stack

Anwendungsprotokolle (REST, CoAP, MQTT), Datenformate, Systemintegration und der MING-Stack als Ende-zu-Ende-Infrastruktur.

Block 4: Drahtlose Kommunikation

Physikalische Grundlagen, Modulation, Multiplexing, Medienzugriff, Short-Range- und LPWAN-Technologien sowie Cellular IoT — der umfangreichste Theorieblock.

Didaktische Grundidee der Vorlesung

Top-Down-Vorgehen

Die Vorlesung folgt einem konsequent anwendungsorientierten Vorgehen: Zunächst werden Anforderungen und Use Cases geklärt, dann werden daraus die technischen Entscheidungen abgeleitet.

- Anforderungen zuerst klären
- Technologien danach auswählen
- Kein technologiegetriebenes Vorgehen

Vom Use Case zur Systemarchitektur

Aus einem konkreten Anwendungsfall werden systematisch abgeleitet:

- Sensorik: Welche Daten werden benötigt?
- Kommunikationstechnologie: Reichweite, Energie, Datenrate
- Verarbeitungsarchitektur: On-Device, Edge, Fog oder Cloud
- Aktorik: Welche Reaktionen sollen ausgelöst werden?

Prüfungserfolg beruht auf dem Überblick über alle Themen *und* der Fähigkeit zur technischen Einordnung.

KAPITEL 1 — BEGRIFFE & PROBLEMRAUM

Begriffsabgrenzung im IoT-Umfeld

Embedded Systems

Rechenfunktion ist fest in ein Gerät integriert. Der Mikrocontroller übernimmt eine spezifische Aufgabe, oft ohne Betriebssystem. Grundlage nahezu aller IoT-Knoten.

Cyber-Physische Systeme (CPS)

Kopplung von Rechenfunktion mit einem physischen Regelkreis. Sensoren erfassen die Umgebung, Aktoren greifen ein. Typisch für Industrie 4.0 und autonome Systeme.

Smart Systems

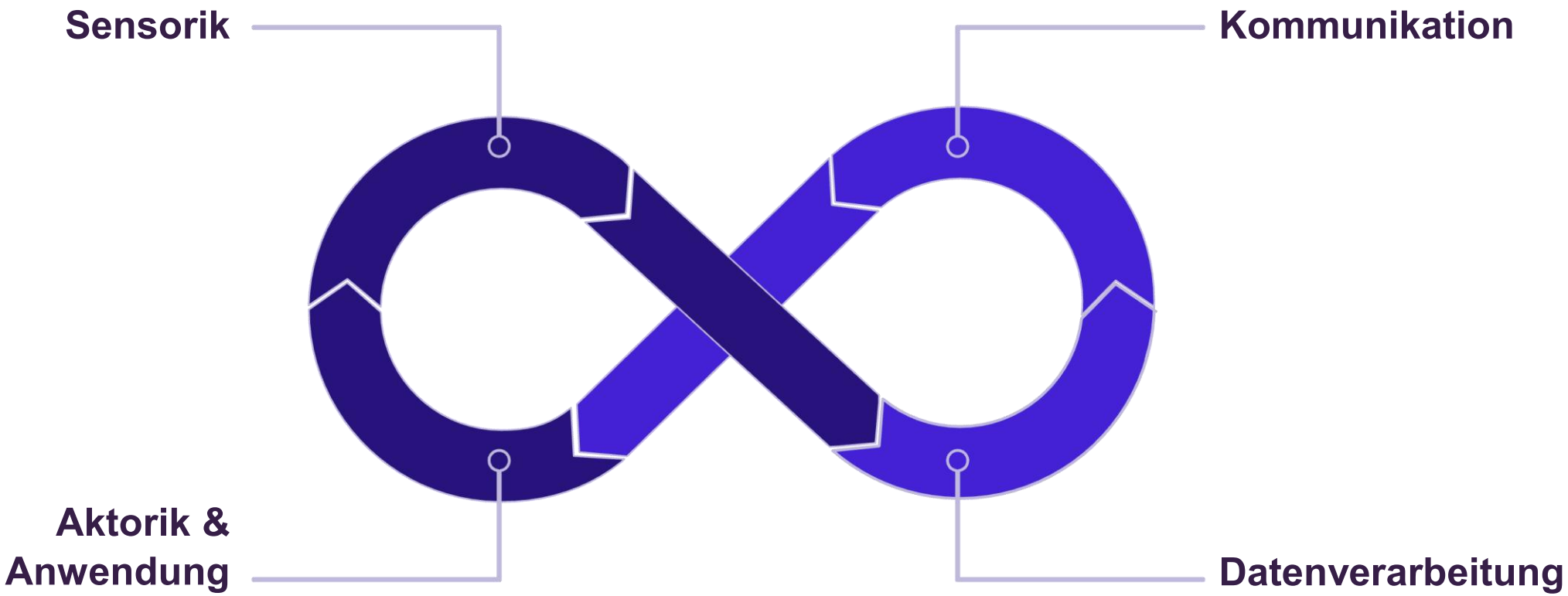
Adaptive Systeme mit integrierter Intelligenz. Lernen aus Daten, passen ihr Verhalten an und optimieren Prozesse autonom. Grundlage für AIoT (AI + IoT).

IoT / Ambient Intelligence / Ubiquitous Computing

Vernetzte, allgegenwärtige Computerwelt. IoT betont Vernetzung von Geräten, Ambient Intelligence die nahtlose Einbettung, Ubiquitous Computing die Überallverfügbarkeit — unterschiedliche Abstraktionsgrade desselben Grundgedankens.

Video: AIoT 01 – Begriffe https://youtu.be/ehFyfNd0tK0

Die 4 IoT-Funktionsebenen



Die vier Ebenen bilden die konzeptionelle Grundstruktur jedes IoT-Systems. Entscheidend ist die Rückkopplung: Die Anwendungsebene löst Aktionen aus, die wiederum auf die physische Welt wirken und neue Sensordaten erzeugen. Wo die Verarbeitung stattfindet — on device, am Edge, im Fog oder in der Cloud — ist eine zentrale Designentscheidung mit Auswirkungen auf Latenz, Energiebedarf und Datenschutz.

Verwandte Gebiete und Technologietrends

Sicherheit und Regulierung

- **Datensicherheit:** Authentifizierung, Verschlüsselung, sichere Firmware-Updates — kritisch für vernetzte Geräte ohne direkte Nutzerinteraktion.
- **Datenschutz / DSGVO:** Erhebung, Speicherung und Verarbeitung personenbezogener Sensordaten unterliegen strengen rechtlichen Anforderungen.
- **Interoperabilität:** Fragmentierte Protokoll- und Plattformlandschaft erfordert Standards wie Matter oder OPC UA.

Technologische Trends

- **Zeitreihenanalyse:** IoT-Daten sind inhärent zeitlich strukturiert. Spezialisierte Datenbanken (z. B. InfluxDB) und Analysemethoden sind erforderlich.
- **Blockchain / Dezentrale Strukturen:** Manipulationssichere Protokollierung von Sensordaten und Transaktionen ohne zentrale Instanz.
- **Digital Twin:** Digitale Echtzeit-Nachbildung eines physischen Systems. Ermöglicht Simulation, Optimierung und Predictive Maintenance ohne Eingriff in das Realsystem.

Video: AIoT 02 – Verwandte Gebiete https://youtu.be/5gHmkDJcioU

Use Cases systematisch analysieren

Leitfragen zur Use-Case-Analyse

Jede IoT-Anwendung beginnt mit systematischen Fragen: Welche Daten entstehen, und in welchem Rhythmus? Wer oder was ist Rezipient der Daten (Mensch, System, Aktuator)? Welche Verarbeitungsschritte sind erforderlich — Filterung, Aggregation, Klassifikation? Welche Latenzanforderungen bestehen?

Vom Use Case zur Systemarchitektur

Die Antworten auf diese Fragen bestimmen die technischen Entscheidungen: Art und Anzahl der Sensoren, Wahl der Kommunikationstechnologie (Reichweite, Energiebudget, Datenrate), Verarbeitungsort (On-Device bis Cloud) sowie die Anforderungen an Aktorik und Feedback. Dieses Analyseschema ist ein zentrales Prüfungsformat für Transferaufgaben — Anwenden auf unbekannte Szenarien ist prüfungsrelevant.

Video: AIoT 03 – Use Cases https://youtu.be/deYlb4kgSzg

IoT-Use-Case-Landschaft

Konsumer & Infrastruktur

- Smart Home (Beleuchtung, Heizung, Sicherheit)
- Smart Grid (Netzstabilisierung, Verbrauchsmessung)
- Connected Healthcare / Remote Patient Monitoring
- Retail (Regalmonitoring, Kundenverfolgung)

Mobilität & Logistik

- Logistik / Asset Tracking (Standort, Temperatur, Schock)
- Vehicle-to-Everything (V2X): Fahrzeuge kommunizieren mit Infrastruktur und anderen Fahrzeugen
- Facility Management (Raumklima, Energieoptimierung)

Industrie & Umwelt

- Smart Farming (Bodenfeuchtigkeit, Bewässerungssteuerung)
- Umwelt- und Gewässermonitoring
- Öl- und Gasindustrie (Leckdetektion, Drucküberwachung)
- Industrielle Predictive Maintenance

Die Breite der Anwendungsbereiche zeigt: IoT ist kein monolithisches Feld — Anforderungen an Kommunikation, Energie und Verarbeitung variieren drastisch zwischen den Domänen.

IoT-Hardware als Kompromissraum

Es gibt keine universell optimale Hardware-Plattform für IoT-Anwendungen. Jede Entscheidung ist ein Trade-off zwischen konkurrierenden Zielen:

Rechenleistung vs. Energiebedarf

Mehr Prozessorleistung erfordert mehr Energie — für batteriebetriebene Knoten oft inakzeptabel.

Konnektivität vs. Systemkomplexität

Integrierte Funkmodule vereinfachen den Aufbau, erhöhen aber Komplexität und Stromverbrauch.

Echtzeitfähigkeit vs. Flexibilität

Deterministische Reaktionszeiten erfordern spezialisierte Hardware oder RTOS; universelle Linux-Systeme bieten keine garantierten Latenzen.

Kosten vs. Funktionsumfang

Günstige MCUs (< 5 €) bieten wenig RAM und keine Konnektivität; vollständige Module kosten mehr, sparen aber Entwicklungszeit.

Speicher vs. Programmierbarkeit

Wenig Flash und RAM begrenzen Firmware-Komplexität; OTA-Updates erfordern Mindest-Speicher.

Formfaktor vs. Wärmeabgabe

Kleine Gehäuse erschweren Wärmeabfuhr — kritisch bei dauerhaft aktiven Geräten.

Sicherheit vs. Ressourcenverbrauch

Kryptographische Operationen (TLS, AES) sind rechenintensiv — auf MCUs oft nur mit Hardware-Beschleuniger realisierbar.

Standardisierung vs. Optimierung

Generische Plattformen (Arduino, ESP32) sind schnell einsetzbar, aber anwendungsspezifische ASICs bieten bessere Effizienz.

Mikrocontroller vs. Single-Board-Computer

Mikrocontroller (MCU)

- CPU, RAM und Flash-Speicher auf einem Chip eng integriert
- Bare Metal oder RTOS (z. B. FreeRTOS, Zephyr)
- Extrem geringer Energieverbrauch (µA-Bereich im Sleep)
- Hohe GPIO-Nähe und direkte Hardwaresteuerung
- Deterministisches Echtzeitverhalten
- Typische Anwendung: Sensorknoten, Aktorsteuerung

Single-Board-Computer (SBC)

- Linux-Systeme mit vollständigem Betriebssystem
- Deutlich höhere Rechenleistung und mehr RAM
- Höherer Energiebedarf (hunderte mW bis mehrere W)
- Gut geeignet für komplexe Software, Container, Gateways
- Keine native Echtzeitfähigkeit ohne Sondermaßnahmen
- Typische Anwendung: Edge-Gateway, Datenvorverarbeitung

Video: AIoT 04 – Einplatinenmaschinen https://youtu.be/Sr8blj_gQeU

Wichtige Hardware-Plattformen

ESP32

Dual-Core-MCU mit integriertem WLAN und Bluetooth. Unterstützt tiefe Schlafmodi (<10 µA). Ideal für batteriebetriebene, sensornahe Knoten. Sehr kostengünstig und gut dokumentiert. Weit verbreitet in Maker- und Prototyp-Projekten sowie in der Industrie.

Raspberry Pi 5

Leistungsstarker SBC mit ARM-Prozessor, mehreren GB RAM und vollständigem Linux. Geeignet für Edge-Gateways, Datenaggregation, Docker-Container und lokale Machine-Learning-Inferenz. Hoher Energiebedarf macht Batteriebetrieb unpraktisch.

Arduino

Einfaches AVR- bzw. ARM-basiertes MCU-System. Ausgezeichnet für Einstieg und Lehre. Kein nativer IP-Stack; Netzwerkanbindung erfordert externe Module. Breites Ökosystem an Shields und Bibliotheken.

STM32 / BeagleBone

STM32: leistungsfähige ARM-Cortex-M-MCUs für industrielle Anwendungen mit HAL und umfangreichen Peripheriefunktionen. BeagleBone: Linux-SBC mit Echtzeit-I/O durch dedizierte PRUs (Programmable Real-time Units) — Sonderrolle für zeitkritische Aufgaben unter Linux.

Hybrider Aufbau mit Feldknoten und Gateway

Feldknoten: ESP32

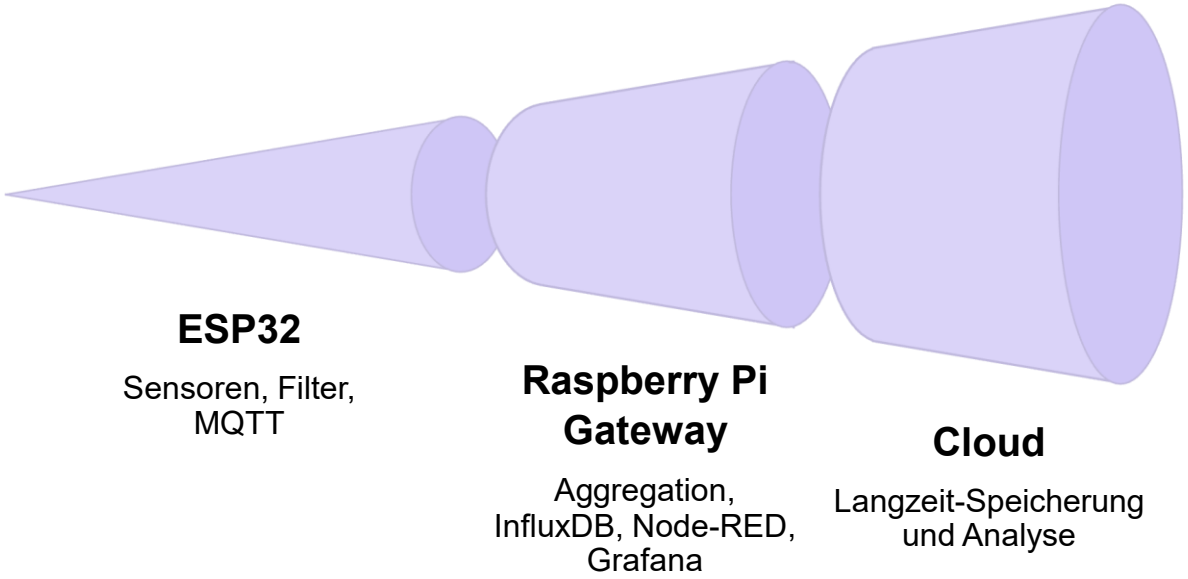
Aufgaben im System:

- Sensordaten erfassen (Temperatur, Feuchte, Druck)
- Digitale und analoge Aktoren schalten
- Lokale Vorfilterung und Schwellwertprüfung
- Datenpakete energieeffizient per MQTT oder REST senden
- Schlafmodi nutzen, um Batterielaufzeit zu maximieren

Gateway: Raspberry Pi

Aufgaben im System:

- Datenaggregation von mehreren Feldknoten
- Persistierung in lokaler Zeitreihendatenbank (InfluxDB)
- Protokollübersetzung (z. B. MQTT zu REST)
- Lokales Dashboard und Alerting (Grafana)
- Optional: Cloud-Weiterleitungslogik via Node-RED



Kommunikationsschnittstellen auf Platinenebene

Der ESP32 bietet alle gängigen Schnittstellen auf einem Chip.

Digitale I/O

- **GPIO:** LEDs, Taster und Relais direkt steuern
- **Interrupts:** Schnelle Reaktion auf Events ohne permanentes Polling
- **PWM:** Für LED-Dimmen oder einfache Motorsteuerung

Serielle Busse

- **I2C:** Typisch für Sensoren und kleine Erweiterungsmodule
- **SPI:** Gut für Displays und schnelle Peripherie
- **UART:** Häufig für GPS-Module und Debugging

Analog

- **ADC:** Der ESP32 hat einen 12-bit ADC
- **ADC-Kanäle:** Bis zu 18 Kanäle für analoge Sensoren

Kommunikationsprotokolle auf Anwendungsebene

Auf der Anwendungsschicht konkurrieren verschiedene Protokolle mit unterschiedlichen Designzielen. Die Wahl hängt von Kommunikationsmodell, Protokoll-Overhead, Ressourcenbeschränkungen und Einsatzszenario ab:

01

REST / HTTP

Zustandsloses Request-Response-Modell über TCP. Ressourcenorientiert, weit verbreitet, aber hoher Header-Overhead.

02

CoAP

UDP-basiertes Leichtgewicht-Protokoll für constrained devices. Ähnliche Semantik wie REST, aber minimaler Overhead.

03

MQTT

Publish/Subscribe über TCP. Asynchron, entkoppelt, Broker-zentriert. Standard für IoT-Datenpipelines.

04

Datenformate

JSON (leichtgewichtig, lesbar) und XML (formalisiert, schema-validierbar) als komplementäre Serialisierungsformate.

Video: AIoT 05 – MING Stack https://youtu.be/vzUiZzl8uaA

Request/Response im IoT: REST vs. CoAP

REST über HTTP

- **Transport:** TCP (verbindungsorientiert, zuverlässig)
- **Zustand:** Zustandslos — jede Anfrage enthält alle nötigen Informationen
- **Adressierung:** Ressourcenorientierte URLs
- **Methoden:** GET, POST, PUT, DELETE
- **Overhead:** Relativ hoher TCP- und HTTP-Header-Overhead
- **Einsatz:** Backend-APIs, Dashboards, wenig ressourcenbeschränkte Geräte

CoAP (Constrained Application Protocol)

- **Transport:** UDP (verbindungslos, geringer Overhead)
- **Zustand:** Ähnliche Semantik wie REST, aber ohne TCP-Handshake
- **Header:** Nur 4 Byte Minimalheader
- **Übertragungstypen:** Confirmable, Non-confirmable, Acknowledgement, Reset
- **Overhead:** Sehr gering — geeignet für 6LoWPAN und IEEE 802.15.4
- **Einsatz:** Stark ressourcenbeschränkte Geräte (MCUs mit wenig RAM)

MQTT als Publish/Subscribe-Protokoll

Grundarchitektur und Transport

MQTT ist TCP-basiert und arbeitet asynchron über einen zentralen Broker (z. B. Mosquitto). Publisher senden Nachrichten an Topics, Subscriber empfangen alle Nachrichten der abonnierten Topics. Clients und Broker sind vollständig entkoppelt. Header sind sehr klein (2 Byte Minimalheader), was MQTT auf ressourcenbeschränkten Geräten effizient macht. Topics sind hierarchisch organisiert (z. B. `home/wohnzimmer/temperatur`) und unterstützen Wildcards (`+` für eine Ebene, `#` für alle Unterebenen).

Erweiterte Funktionen

- **QoS 0:** At most once — keine Garantie, kein Overhead
- **QoS 1:** At least once — Empfangsbestätigung, Duplikate möglich
- **QoS 2:** Exactly once — Four-Way-Handshake, höchste Zuverlässigkeit
- **Retained Messages:** Broker speichert letzte Nachricht je Topic; neue Subscriber erhalten sofort den letzten Wert
- **Last Will:** Vordefinierte Nachricht, die der Broker bei unerwartetem Verbindungsabbruch publiziert — für Gerätestatusüberwachung

JSON und XML im Vergleich

JSON (JavaScript Object Notation)

Leichtgewichtiges, textbasiertes Format auf Basis von Key-Value-Paaren, Arrays und verschachtelten Objekten. Gut lesbar für Menschen, einfach zu parsen. De-facto-Standard für REST-APIs und MQTT-Payloads.

```
{
  "sensor": "temp_01",
  "value": 22.5,
  "unit": "°C"
}
```

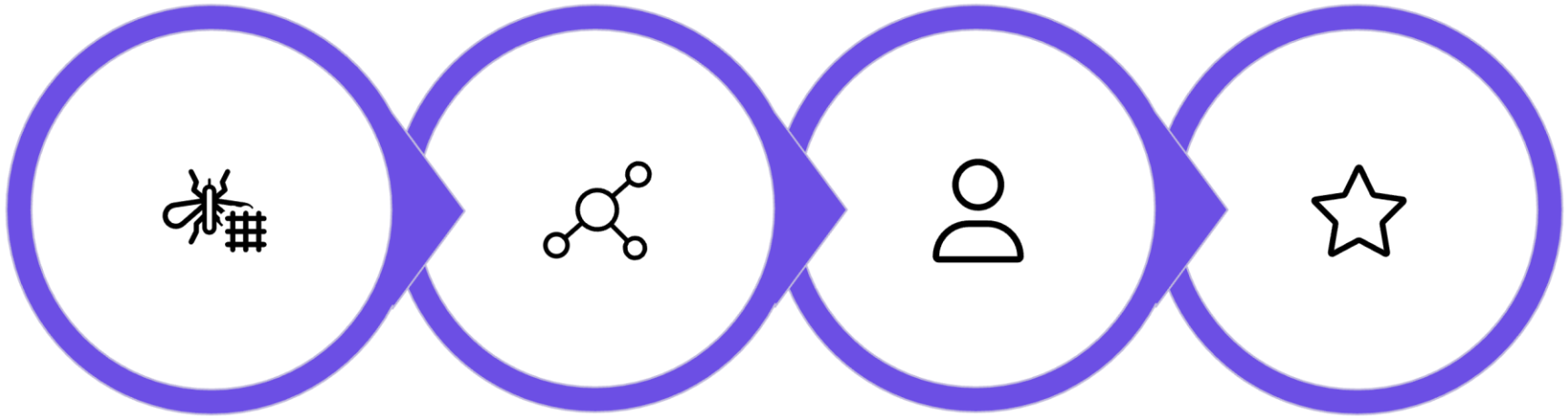
XML (eXtensible Markup Language)

Stärker formalisiertes, baumstrukturiertes Format mit expliziten öffnenden und schließenden Tags. Schema-Validierung möglich (XSD). Typisch in Enterprise-Integrationssystemen und OPC UA.

```
<sensor id="temp_01">
  <value unit="°C">22.5</value>
</sensor>
```

Kriterium	JSON	XML
Lesbarkeit	Hoch	Mittel
Overhead	Gering	Höher
Schema-Validierung	Optional (JSON Schema)	Stark (XSD)
Typische Nutzung	REST, MQTT, Web-APIs	OPC UA, Enterprise, SOAP

Der MING-Stack



MQTT **Node-RED** **InfluxDB** **Grafana**

- **MQTT / Mosquitto:** Nachrichtentransport per Publish/Subscribe
- **Node-RED:** Visuelle Datenverarbeitung und Routing
- **InfluxDB:** Zeitreihendatenbank für Sensordaten
- **Grafana:** Dashboard und Visualisierung

Aufgaben der Systemkomponenten im MING-Stack

MQTT / Mosquitto

Asynchroner Nachrichtentransport per Publish/Subscribe. Mosquitto ist ein leichtgewichtiger Open-Source-Broker. Entkoppelt Produzenten (Sensoren) von Konsumenten (InfluxDB, Node-RED). Mehrere Subscriber für denselben Topic möglich.

InfluxDB

Spezialisierte Zeitreihendatenbank. Optimiert für hochfrequentes Schreiben und zeitbasierte Abfragen (InfluxQL / Flux). Speichert Messwerte mit Zeitstempel, Tags und Fields. Retention Policies für automatisches Datenlöschen.

Node-RED

Visuelles, flussbasiertes Programmierframework. Verarbeitet Daten als JSON-basierte `msg`-Objekte. Verbindet MQTT-Eingänge, Transformationslogik, Datenbankschreiber und HTTP-Endpoints ohne klassischen Code. Gut geeignet für Prototypen und Integrationsaufgaben.

Grafana

Visualisierungsplattform für Zeitreihendaten. Dashboards mit Graphen, Gauges und Heatmaps. Direktanbindung an InfluxDB. Alerting bei Schwellwertüberschreitungen. Für Monitoring, Betrieb und Präsentation von IoT-Systemzuständen.

Videos: AIoT 06 – Node-RED / InfluxDB / Grafana / Matter https://youtu.be/szx19dkxhAQ | AIoT Ü7 – MING Stack https://youtu.be/pGz2K8z9WRc

Standards und Interoperabilität

Matter

IP-basierter, plattformübergreifender Anwendungsstandard für das Smart Home. Getragen von Apple, Google, Amazon und Samsung. Arbeitet über Thread (Mesh-IP) oder WLAN. Ziel: einheitliche App-Steuerung über Hersteller hinweg. Relevanz steigt ab 2023 stark an.

Zigbee

Etablierter Smart-Home-Funkstandard auf Basis von IEEE 802.15.4. Mesh-Netzwerk mit geringem Energiebedarf. Häufig über Bridges (z. B. Philips Hue Hub) integriert. Vielzahl von Geräteherstellern, aber historisch fragmentiert.

OPC UA

Industrieller Kommunikationsstandard (IEC 62541). Plattformunabhängig, transportagnostisch (TCP, MQTT, HTTPS). Starkes semantisches Informationsmodell: Maschinen beschreiben ihre eigene Struktur und Daten. Standard für Industrie 4.0 und IIoT.

Physikalische Grundlagen elektromagnetischer Wellen

Grundzusammenhänge

Frequenz f, Wellenlänge λ und Lichtgeschwindigkeit c sind verknüpft durch:

$$c = f \quad \lambda$$

Niedrigere Frequenzen bedeuten längere Wellenlängen. Elektrische Antennen müssen auf die Wellenlänge abgestimmt sein — niedrige Frequenzen erfordern größere Antennen.

- Niedrigere Frequenzen: höhere Reichweite, bessere Gebäudedurchdringung
- Wellenlänge im Sub-GHz-Band: 30 cm bis mehrere Meter

Frequenz und Datenrate

Höhere Trägerfrequenzen ermöglichen breitere Kanäle und damit höhere Datenraten — erkauft durch kürzere Reichweite und schlechtere Penetration.

- **Sub-GHz (z. B. 868 MHz):** Lange Reichweite, geringer Datendurchsatz — typisch für LoRa, SigFox, NB-IoT
- **2,4 GHz:** Mittlere Reichweite, höhere Datenraten — typisch für WLAN (802.11b/g/n), Bluetooth, Zigbee
- **5 GHz / 6 GHz:** Sehr hohe Datenraten, kurze Reichweite — typisch für WLAN 5/6

Videos: AIoT 07 https://youtu.be/K17RqDwXSog | AIoT 08 https://youtu.be/xl95w5s8X9Q

Störeffekte bei Funkübertragung

Dämpfung (Path Loss)

Freiraumdämpfung wächst quadratisch mit der Distanz (Friis-Formel). Materialdämpfung durch Wände, Wasser und menschliche Körper. Beschrieben durch den Pfadverlust-Exponenten *n* (Freiraum: n=2, Gebäude: n=3–5).

Reflexion

Signale werden an glatten Oberflächen (Metall, Beton, Glas) reflektiert. Führt zu Mehrwegeausbreitung: dasselbe Signal trifft auf verschiedenen Pfaden mit unterschiedlichen Laufzeiten am Empfänger ein.

Beugung und Streuung

Beugung ermöglicht Signalausbreitung um Hindernisse herum (Huygensches Prinzip). Streuung tritt an rauen Oberflächen oder kleinen Objekten auf und verteilt Signalenergie in viele Richtungen.

Fading und Doppler-Effekt

Mehrwegeausbreitung erzeugt destruktive Interferenz (Multipath Fading). Schnelles Fading (Rayleigh) bei Bewegung, langsames Fading (Shadowing) durch Hindernisse. Doppler-Verschiebung bei bewegten Sender/Empfänger-Paaren verändert die Empfangsfrequenz und degradiert das Signal.

Reale Funkkanäle verhalten sich nicht wie ideale Leitungen. Die Kenntnis dieser Effekte ist Grundlage für die Bewertung von Modulationsverfahren und Diversitätstechniken.

Modulation und Bitabbildung

Basismodulationsverfahren

- **ASK (Amplitude Shift Keying):** Bits werden auf verschiedene Amplituden abgebildet. Einfach, aber störanfällig gegenüber Amplitudenschwankungen.
- **FSK (Frequency Shift Keying):** Bits werden auf verschiedene Frequenzen abgebildet. Robuster gegenüber Amplitudenrauschen; typisch für LoRa (als Chirp Spread Spectrum eine FSK-Variante).
- **PSK (Phase Shift Keying):** Bits werden auf Phasenlagen abgebildet (BPSK: 1 Bit/Symbol, QPSK: 2 Bit/Symbol). Gute Störfestigkeit bei hoher spektraler Effizienz.

QAM und Prüfungsrelevanz

- **QAM (Quadrature Amplitude Modulation):** Kombiniert Amplituden- und Phasenmodulation. 16-QAM: 4 Bit/Symbol; 64-QAM: 6 Bit/Symbol; 256-QAM: 8 Bit/Symbol. Höhere QAM-Stufen = höhere Effizienz, aber empfindlicher gegenüber Rauschen.
- **Absolut vs. differenziell:** Absolutes Mapping bezieht sich auf feste Referenzphasen; differenzielles Mapping kodiert Bit-Änderungen relativ zum Vorgängersymbol — robuster bei Phasendrift.

Prüfungsrelevant: Die Wahl des Modulationsverfahrens beeinflusst direkt Robustheit, spektrale Effizienz und Reichweite.

Ressourcenteilung im Funkkanal: Multiplexing

Klassische Multiplexverfahren

- **FDMA (Frequency Division Multiple Access):** Jeder Nutzer erhält ein eigenes Frequenzband. Einfach, aber unflexible Ressourcennutzung.
- **TDMA (Time Division Multiple Access):** Nutzer teilen sich einen Kanal in Zeitschlitzen. Effizient bei gleichmäßigem Verkehr.
- **SDMA (Space Division Multiple Access):** Räumliche Trennung durch Richtantennen oder MIMO. Grundlage moderner Mobilfunksysteme.
- **CDMA (Code Division Multiple Access):** Nutzer senden gleichzeitig auf derselben Frequenz, unterschieden durch orthogonale Spreizsequenzen. Robust, aber komplex in der Demodulation.

OFDM — Orthogonal Frequency Division Multiplexing

OFDM unterteilt einen breiten Kanal in viele schmale, orthogonale Subträger, die parallel übertragen werden. Jeder Subträger trägt einen Teil der Daten mit niedrigerer Symboldauer.

- Spektral sehr effizient durch orthogonale Überlappung
- Robust gegenüber Mehrwegeausbreitung (Cyclic Prefix)
- Grundlage für WLAN (802.11a/g/n/ac/ax), LTE und 5G
- Implementierung via FFT/IFFT effizient realisierbar

Video: AIoT 09 – Multiplex, Codierung https://youtu.be/Q4hB87OD9G0

MAC-Layer und Kanalkoordination

CSMA/CA — Collision Avoidance

Carrier Sense Multiple Access with Collision Avoidance ist das Kernverfahren in WLAN (IEEE 802.11). Anders als bei CSMA/CD (Ethernet) werden Kollisionen nicht erkannt, sondern aktiv vermieden: Vor der Übertragung wird der Kanal abgehört (Carrier Sense). Ist er belegt, wartet der Knoten einen zufälligen Backoff. Interframe Spaces (SIFS, PIFS, DIFS) definieren Prioritäten: kurze SIFS-Lücken erlauben ACK-Frames höchste Priorität; DIFS ist der normale Abstand für Datenpakete.

Koordinierte Verfahren

Polling-basierte Verfahren wie PCF (Point Coordination Function) geben einer zentralen Instanz (Access Point) die Kontrolle über den Kanalzugriff. Einzelne Stationen werden nacheinander abgefragt — deterministische Latenz, kein Zufalls-Backoff. Relevant für Echtzeitanwendungen. Ziel aller MAC-Verfahren: geregelter, fairer Zugriff auf das geteilte Medium bei gleichzeitiger Minimierung von Kollisionen und Overhead.

Video: AIoT 10 – Medienzugriff https://youtu.be/3ATDeSb2FKU

WLAN, Bluetooth und BLE

WLAN / IEEE 802.11

- Hohe Bandbreite (bis mehrere Gbit/s bei Wi-Fi 6)
- 2,4 GHz und 5 GHz / 6 GHz Betrieb
- Hoher Energiebedarf — nur bedingt für batteriebetriebene IoT-Knoten
- Ideal für Gateway-Anbindung und datenintensive Anwendungen
- ESP32 integriert 802.11b/g/n nativ

Bluetooth Classic vs. BLE

- **Bluetooth Classic:** Durchgehende Verbindung, höhere Datenrate, Audio-Streaming — hoher Energiebedarf
- **BLE (Bluetooth Low Energy):** Sporadische, kurze Übertragungen; sehr geringer Energiebedarf; Verbindungsaufbau schnell (ca. 3 ms)
- BLE-Profil: GATT-basierte Services und Characteristics

Mesh-orientierte Smart-Home-Protokolle

- **Zigbee:** IEEE 802.15.4, 2,4 GHz Mesh-Netz, bis zu 65.000 Knoten, sehr geringer Energiebedarf
- **Z-Wave:** Sub-GHz (868 MHz in Europa), Mesh, max. 232 Knoten, weniger Interferenz durch 2,4-GHz-Geräte
- Einsatz: Wearables, Proximity-Erkennung, Asset Tracking, Smart Home

Video: AIoT 11 – WIFI, Bluetooth https://youtu.be/RfXXtFfNxP0

Long-Range-IoT mit wenig Energie: LPWAN

LoRa und LoRaWAN

LoRa ist eine physikalische Modulationstechnik (Chirp Spread Spectrum), die auf sub-GHz-Frequenzen (868 MHz in Europa) operiert. Sehr robuste Übertragung auch bei schlechtem SNR. **LoRaWAN** ist das darüber liegende Kommunikationsprotokoll (MAC + Netzwerkschicht). Sternnetztopologie mit Gateways, die Pakete an einen Network Server weiterleiten.

- Reichweite: bis 15 km im Freien, 2–5 km urban
- Datenrate: sehr gering (0,3–50 kbit/s)
- Energieverbrauch: extrem gering — Jahre auf Batterie möglich
- Geräte-Klassen A, B, C mit unterschiedlichem Downlink-Verhalten

SigFox

Ultra-Narrowband-Ansatz: Sehr schmale Kanäle (100 Hz) auf sub-GHz-Frequenzen. Proprietäres Netzwerk mit zentralisierter Infrastruktur.

- Sehr geringer Hardware-Aufwand auf Knotenseite
- Reichweite vergleichbar mit LoRa
- Extrem niedrige Datenraten (12 Bytes je Nachricht, max. 140 Nachrichten/Tag)
- Einsatz: Asset Tracking, Zählerwesen, einfache Alarmsysteme
- Nachteil: proprietäre Infrastruktur, begrenzte Netzverfügbarkeit

NB-IoT und LTE-M: Cellular IoT

LTE-M (LTE Cat-M1)

- Höhere Datenraten als NB-IoT (bis ca. 1 Mbit/s)
- Unterstützt Mobilität und Handover — für bewegliche Objekte geeignet
- Voice over LTE (VoLTE) möglich
- Einsatz: Wearables, Fahrzeugtracking, Logistik mit Mobilität

NB-IoT (Narrowband IoT)

- Schmalbandig (180 kHz), besonders gute Gebäude- und Kellerdurchdringung
- Keine Mobility-/Handover-Unterstützung — stationäre Geräte
- Betriebsmodi: In-Band (in LTE-Ressourcen), Guard Band, Standalone
- Einsatz: Smart Metering, Zählerwesen, stationäre Sensoren

Merkmal	LTE-M	NB-IoT	LoRaWAN
Reichweite	Mittel	Hoch (gut in Gebäuden)	Sehr hoch
Datenrate	Bis 1 Mbit/s	Bis 250 kbit/s	<50 kbit/s
Mobilität	Ja (Handover)	Nein	Begrenzt
Energiesparen	PSM, eDRX	PSM, eDRX	Geräteklassen A/B/C
Infrastruktur	Mobilfunknetz	Mobilfunknetz	Eigenes/Community-Netz

PSM (Power Saving Mode) und eDRX (extended Discontinuous Reception) sind zentrale Energiesparmechanismen in 3GPP-basiertem Cellular IoT — prüfungsrelevant.

Video: AIoT 12 – LPWAN, Long Range Communication https://youtu.be/-s22aPy0WyI

Grundkonflikte drahtloser Kommunikation

Drahtlose Kommunikation im IoT ist geprägt von fundamentalen physikalischen und technischen Zielkonflikten, die keine universelle Lösung zulassen:

LPWAN (LoRa, Sigfox)

- Sehr große Reichweite (km)
- Sehr geringer Energiebedarf
- Niedrige Datenrate (< 50 kbit/s)

Mobilfunk-IoT (NB-IoT, LTE-M)

- Mittlere bis große Reichweite
- Mobilfunknetz-Infrastruktur
- Höhere Datenrate, PSM/eDRX

WLAN / Bluetooth / BLE

- Kurze bis mittlere Reichweite
- Hohe Datenrate
- Höherer Energiebedarf

Dimension	LPWAN	Mobilfunk-IoT	WLAN/BLE
Reichweite	Sehr hoch	Hoch	Gering
Datenrate	Sehr niedrig	Mittel	Hoch
Energiebedarf	Sehr gering	Mittel	Hoch
Infrastruktur	Eigen/Community	Mobilfunknetz	Lokal
Mobilität	Begrenzt	Ja (LTE-M)	Ja