



# Digitale Systeme

vormals "Technische Grundlagen der Informatik 1"

## Kapitel 1: Motivation und Einführung

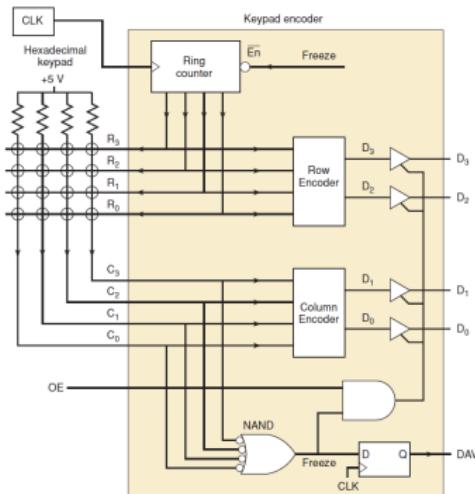
basierend auf den Unterlagen der Vorlesung von  
Prof. Dr.-Ing. Carsten Gremzow

**Prof. Dr.-Ing. Sebastian Möller**

Technische Universität Berlin  
Quality and Usability Lab  
Institut für Theoretische Informatik und Softwaretechnik  
Fakultät IV – Elektrotechnik und Informatik



1. Motivation und Einführung
  - Organisatorisches
  - Digitale Systeme vs. Rechnerorganisation
  - Themen Digitale Systeme am Beispiel Keypad Encoder
  - Streifzug durch die Computertechnik
2. Logik und Boolesche Algebra
3. Gatter und CMOS-Technik
4. Zahlendarstellung und Codes
5. Schaltnetze und Normalformen
6. Optimierung von Schaltnetzen
7. Standard-Schaltnetze
8. Speicherelemente und programmierbare Logik
9. Synchrone Schaltwerke
10. Register-Transfer-Entwurf und Mikroprogrammierung
11. Anwendungen Digitaler Systeme



Aus: Tocci, Widmer, Moss, 2010, Abb. 10-11.

- Organisation und Ansprechpartner
  - Vorlesung
  - Übung & Hausaufgaben & ISIS
  - Klausur & Prüfungsform
- Worum geht es in "Digitale Systeme"?
- Streifzug durch die Computertechnik



## Digitale Systeme

- Modul 40413
- regelmäßig jedes Sommersemester
- Integrierte Veranstaltung, im Mittel 2 SWS VL, 2 SWS UE (6 ECTS)
- Pflichtveranstaltung Studiengang Technische Informatik, BSc, und Medientechnik, BSc
- Wahlpflichtveranstaltung Studiengang Informatik, BSc
- Wahlpflichtveranstaltung in weiteren Studiengängen

## Vorlesung

- Dienstag, 8h15 – 9h45 Uhr ab 16.04.2024 Raum H 1012

## Übung

- Übungen in Tutorien  
4-5 Gruppen anwesend, 1 Gruppen Live-Streaming Zoom
- **Anmeldung zu den Tutorien via ISIS bis Mo, 21.04.2024**



## Dozent

- Prof. Dr.-Ing. **Sebastian Möller**
- E-Mail: [sebastian.moeller@tu-berlin.de](mailto:sebastian.moeller@tu-berlin.de)
- Gebäude: TEL, 6. Etage

## Übungsleitung

- **Maurizio Vergari, Maximilian Warsinke,**  
E-Mail: [maurizio.vergari@tu-berlin.de](mailto:maurizio.vergari@tu-berlin.de), [warsinke@tu-berlin.de](mailto:warsinke@tu-berlin.de)  
Gebäude: TEL, 3. Etage
- **Tutoren** (Kontakt & Sprechzeit auf ISIS)

## Sekretariat

- **Frau Yasmin Hillebrenner**
  - E-Mail: [yasmin.hillebrenner@tu-berlin.de](mailto:yasmin.hillebrenner@tu-berlin.de)
  - Gebäude: TEL, 3. Etage



- **Prüfungsform:** Schriftliche Prüfung mit Vorleistung
  - **schriftliche Prüfung = Klausur**
  - Klausurnote = Modulnote
  - **Vorleistung = Hausaufgaben**
- **QISPOS-Termine für Pflicht- und Wahlpflichtmodule**  
**(aktuelle Planung, Änderung möglich!):**

|                    | 1.Klausur  | 2.Klausur  |
|--------------------|------------|------------|
| Anmeldebeginn      | 26.04.2024 | 26.04.2024 |
| Anmeldeende im MTS | tba        | tba        |
| Rücktrittsende     | tba        | tba        |
| Prüfungsdatum      | tba        | tba        |

- Die Anmeldung zur Prüfung ist nur nach **erfolgreichem Erbringen der Vorleistung** möglich.

**Die endgültigen Daten und Aktualisierungen werden  
in ISIS bekannt gegeben**



## Vorleistung = Hausaufgaben

- Es werden **2 Hausaufgaben** zur Einzelarbeit (online im ISIS)
- Die Aufgabenstellungen orientieren sich an den bis dahin vermittelten Übungsinhalten.
- Es müssen mindestens **50% der Hausaufgabepunkte in JEDER Hausaufgabe** erreicht werden, um zur Klausur zugelassen zu werden.

|     | Punkte    | Online | Abgabe                |
|-----|-----------|--------|-----------------------|
| HA1 | 100 / 200 | 13.05. | bis <b>10.06.2024</b> |
| HA2 | 100 / 200 | 03.06. | bis <b>30.06.2024</b> |



- **Weitere Information über ISIS**

<https://www.isis.tu-berlin.de>

- Prüfungsmodalitäten (im Detail)
- Anmeldetermine (Modulprüfung, Klausur)
- Klausurbedingungen
- Vorlesungsfolien, Vorlesungsvideo, Q/A Forum
- Übungsblätter, Aufzeichnung der Lösung , Q/A Forum
- Hausaufgaben
- Angaben zu Räumlichkeiten (Tutorien, Sprechstunden, Klausur)

**Q/A für Übungen: 18.04.2024 um 15:00 Uhr (online)**

**Einschreibung:**

Kursname:

**[SoSe 2024] Digitale Systeme, QU, VL+UE**

Einschreibeschlüssel:

**Digsys2024**



- Leider gibt es auch bei dieser grundlegenden Lehrveranstaltung immer noch signifikante Durchfallquoten
- Deshalb: Regelmäßige Teilnahme an den Übungen/Tutorien
- Nehmen Sie die Sprechstunden der Übungsleiter/Tutoren in Anspruch...
- ... nicht erst 14 Tage vor der Klausur
- **lösen Sie Übungsaufgaben selbst !**
- ... nicht erst 2 Tage vor der Klausur
- ... sondern vorlesungsbegleitend.



## 1. Motivation und Einführung

- ✓ Organisatorisches
- Digitale Systeme vs. Rechnerorganisation
- Themen Digitale Systeme am Beispiel Keypad Encoder
- Streifzug durch die Computertechnik

## 2. Logik und Boolesche Algebra

## 3. Gatter und CMOS-Technik

## 4. Zahlendarstellung und Codes

## 5. Schaltnetze und Normalformen

## 6. Optimierung von Schaltnetzen

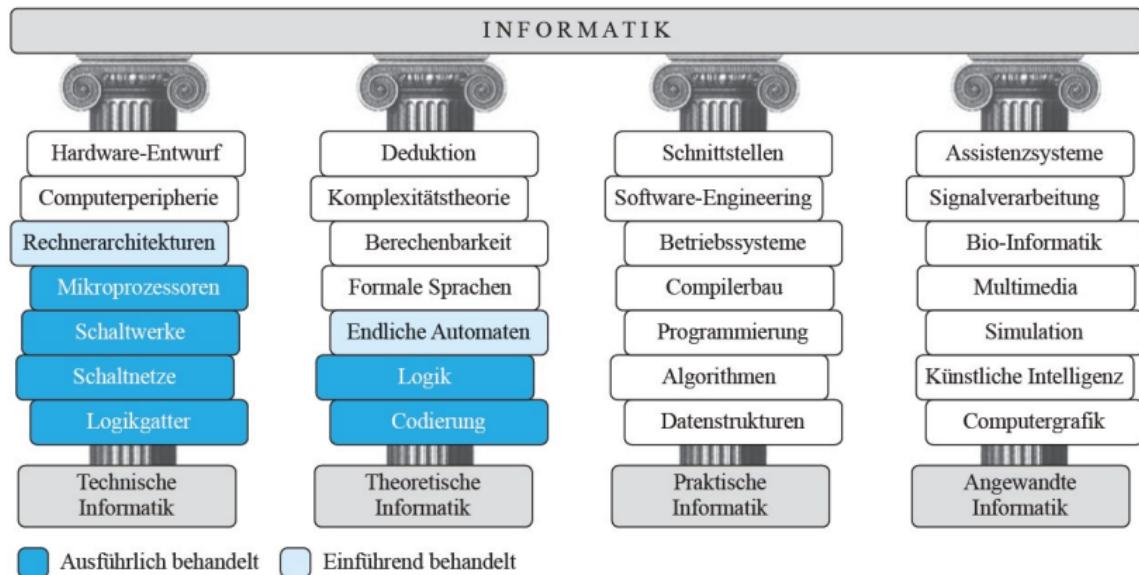
## 7. Standard-Schaltnetze

## 8. Speicherelemente und programmierbare Logik

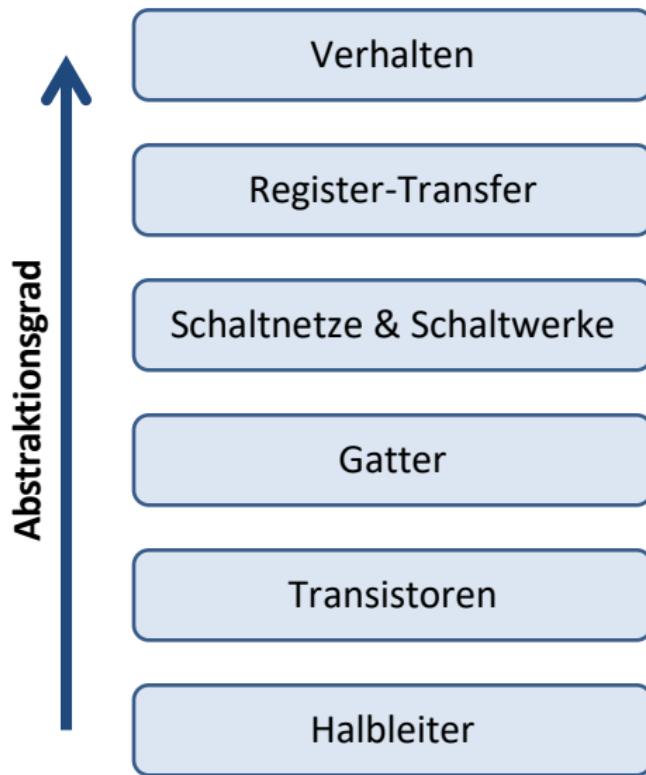
## 9. Synchrone Schaltwerke

## 10. Register-Transfer-Entwurf und Mikroprogrammierung

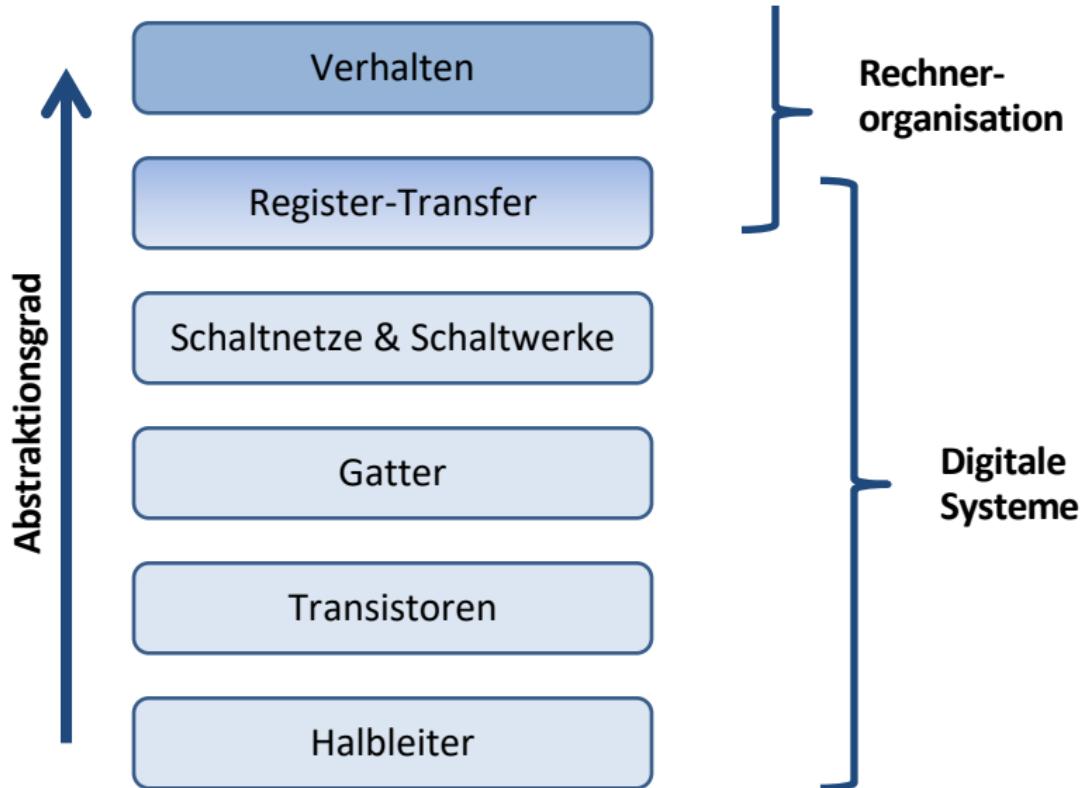
## 11. Anwendungen Digitaler Systeme



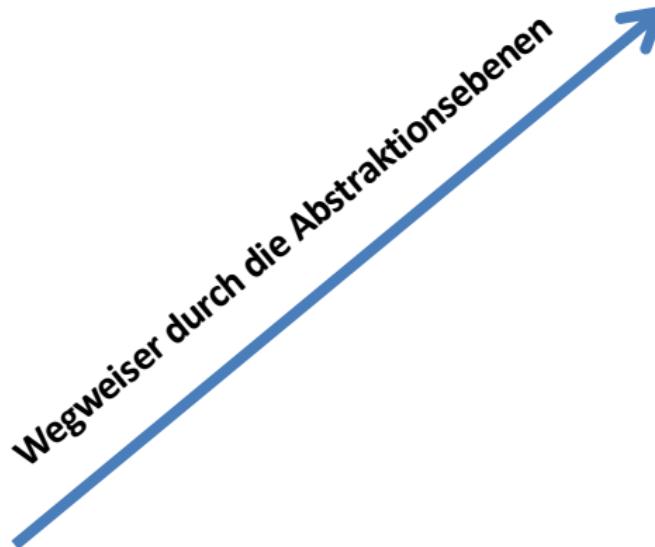
(Hoffmann, Grundlagen der Technischen Informatik, Hanser, 2014, Abb. 1.1)



(nach: Hoffmann, Grundlagen der Technischen Informatik, Hanser, 2014, Abb. 10.1)



(nach: Hoffmann, Grundlagen der Technischen Informatik, Hanser, 2014, Abb. 10.1)



- Verhalten
- Register-Transfer
- Schaltnetze & Schaltwerke
- Gatter
- Transistoren
- Halbleiter

(nach: Hoffmann, Grundlagen der Technischen Informatik, Hanser, 2014, Abb. 10.1)



## Gegenstand

- Organisationsprinzipien von Hardware und Software u. deren Zusammenspiel
- Beschreibung von **Struktur** und **Funktion** digitaler Rechner

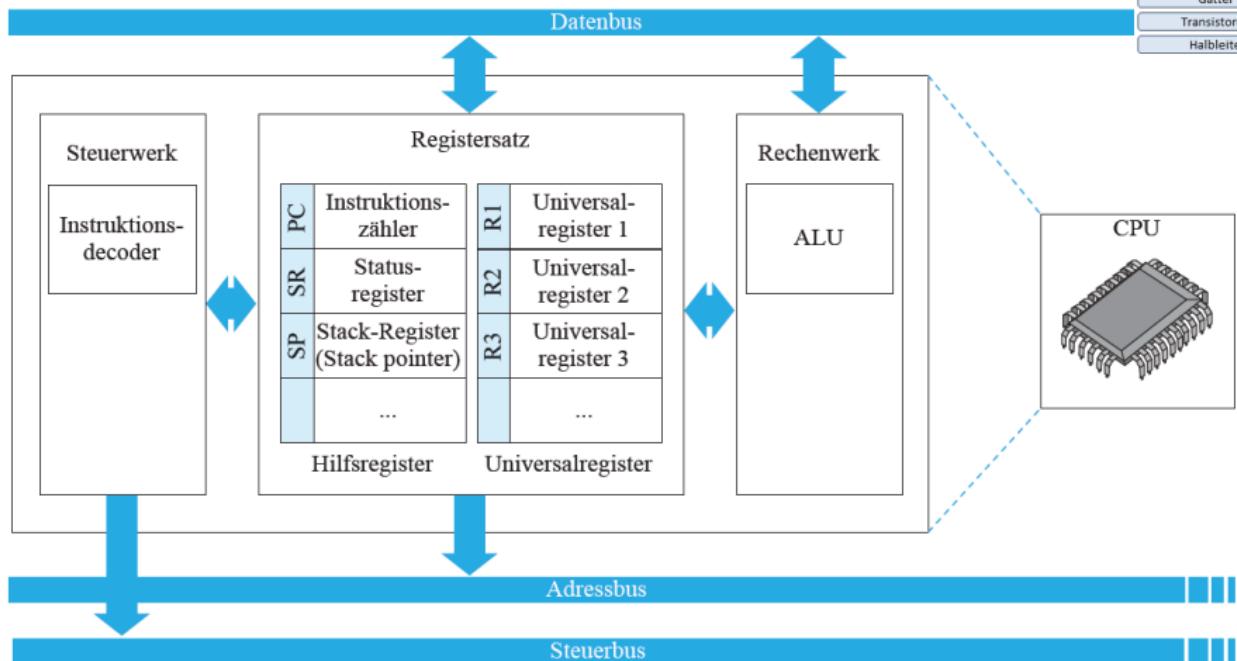
## Motivation

- Entwurf eines Rechners erlaubt für jede Funktionseinheit die Möglichkeit der Realisierung in Hardware oder Software
- Ingenieur hat zu entscheiden, wie er unter Abwägung von Hardware und Software effiziente Systeme „baut“

## Ziel

- Erwerb von Grundkenntnissen über Aufbau und Funktionsweise von programmierbaren digitalen Systemen
- Verständnis grundlegender Konzepte der Rechnerorganisation und der Mikroprozessortechnik
- Fertigkeit im Umgang mit der Mikro- und Assemblerprogrammierung

## Aufbau einer CPU



(Hoffmann, Grundlagen der Technischen Informatik, Hanser, 2014, Abb. 11.5)

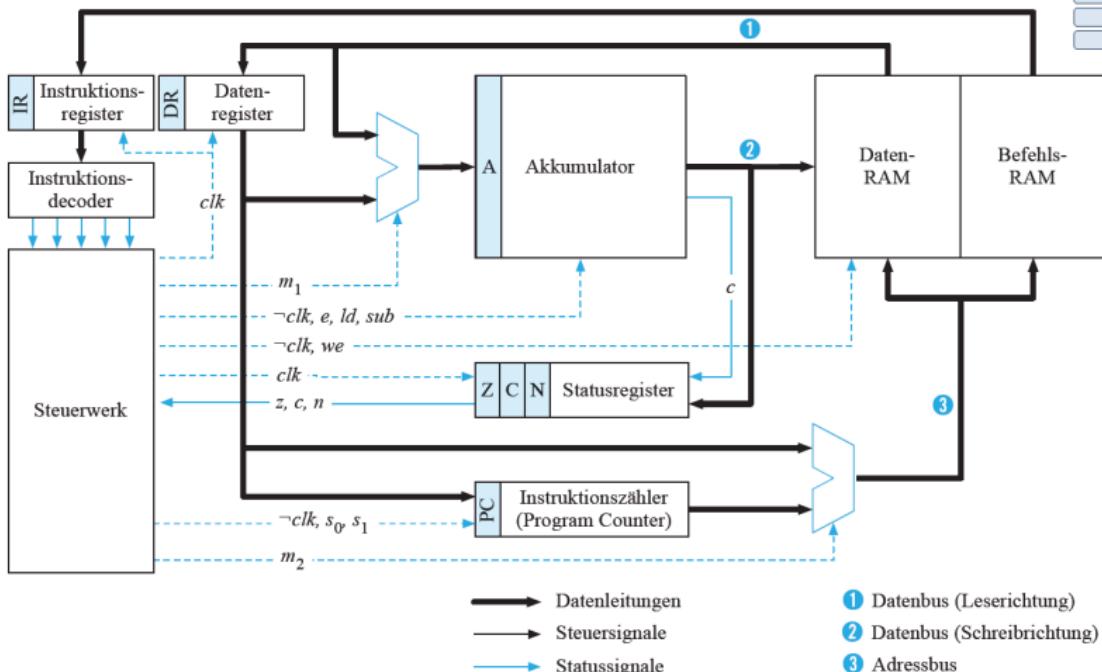


- **Rechnerorganisation** betrachtet u. A. die **Struktur** eines Rechners
- Zusammenspiel von Funktionseinheiten, Speicher, Verbindungen und Steuerung -> **Register-Transfer-Ebene**
- Funktionseinheiten sind *Black Boxes*
  - Addierer, Subtrahierer, Multiplizierer etc.
  - Vergleicher, Codierer, Decodierer etc.
- Speicherelemente sind *Black Boxes*
  - Register, Register-Files
  - statischer bzw. dynamischer Speicher
- Steuerungen (finite Zustandsautomaten) sind *Black Boxes*
- Verbindungen sind *Black Boxes*
  - Multiplexer
  - Demultiplexer



**Digitale Systeme schaut nun in die Black Boxes!**

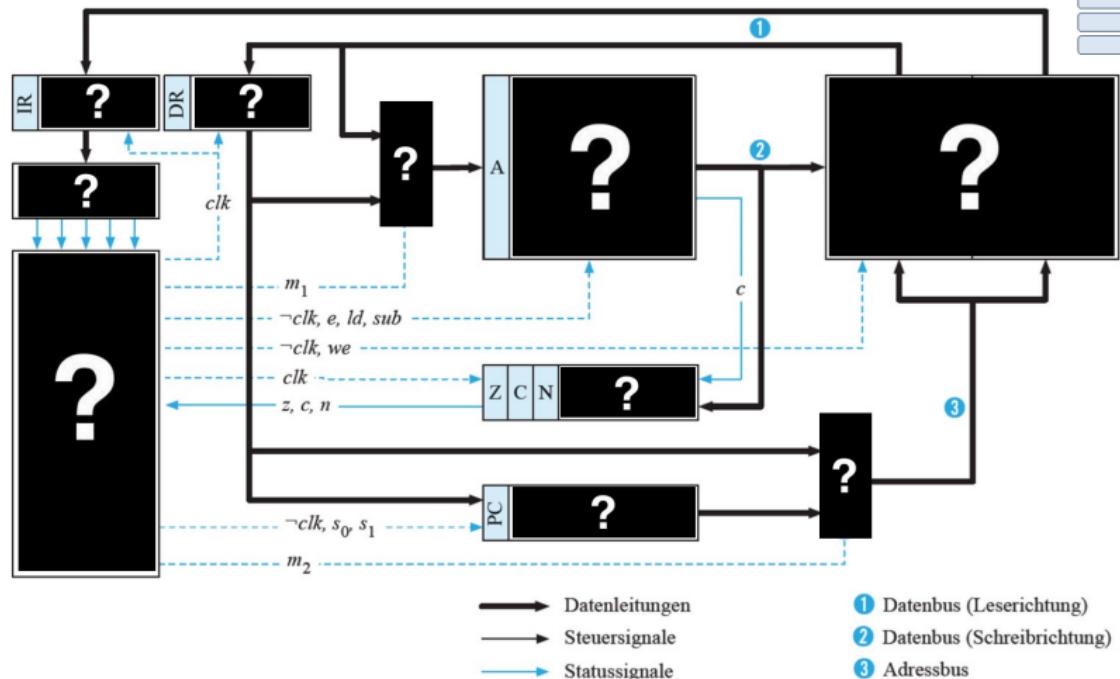
## Aufbau einer CPU



(Hoffmann, Grundlagen der Technischen Informatik, Hanser, 2014, Abb. 11.11)



## Aufbau einer CPU



(Hoffmann, Grundlagen der Technischen Informatik, Hanser, 2014, Abb. 11.11)



**Digitale Systeme schaut in die Black Boxes und...**

- erklärt wie sie aufgebaut sind,
- welche Alternativen man beim Aufbau hat,
- wie komplex der Aufbau ist...
- ...was er kostet...
- ...ob man die Kosten reduzieren kann...
- ...wie man es automatisieren kann.

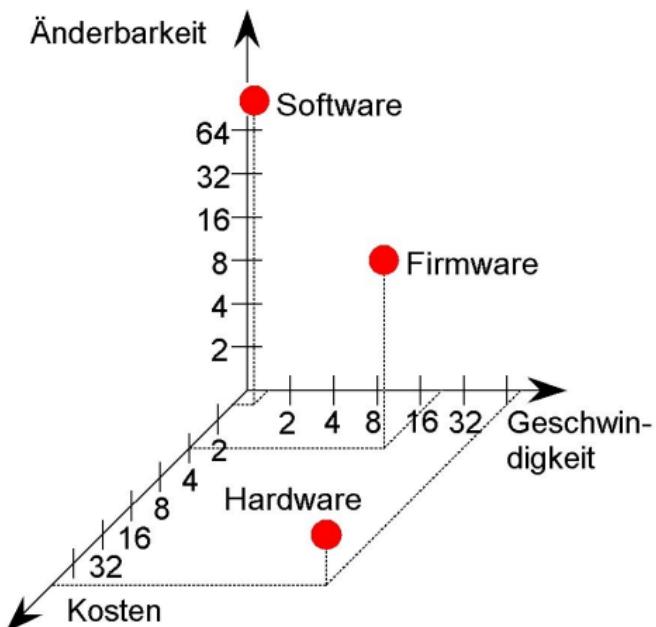
Parameter, die für Digitale Systeme wichtig sind

- Entwurfskosten
- Entwurfskomplexität
- Entwurfsoptimierung
- Entwurfsautomatisierung / Entwurfsproduktivität
- Entwurfsalternativen



Verhalten  
Register-Transfer  
Schaltnetze & Schaltwerke  
Gatter  
Transistoren  
Halbleiter

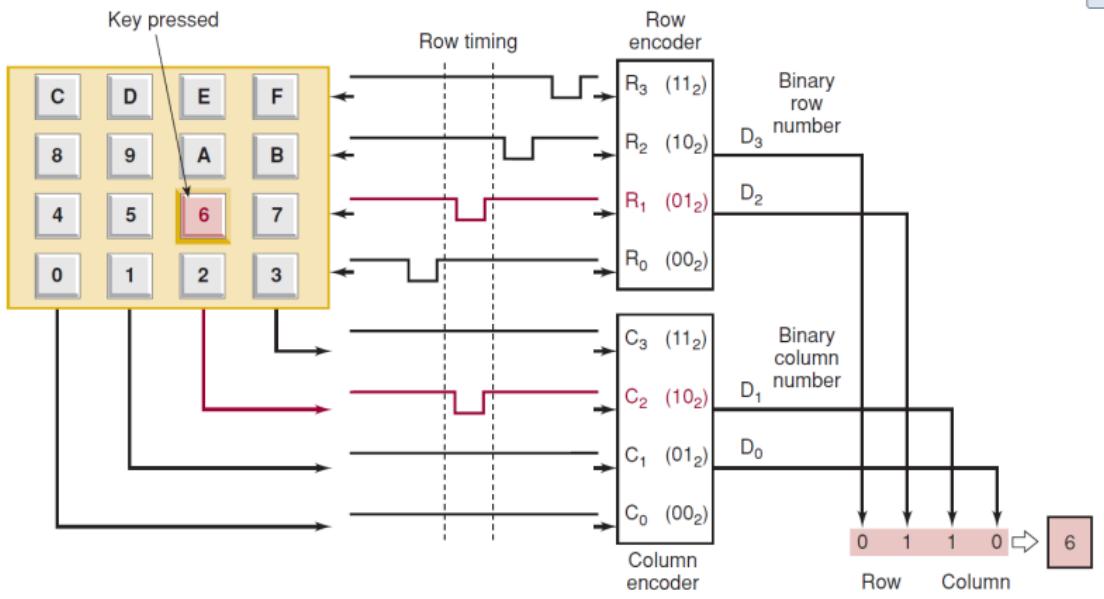
## Entwurfsalternativen: "Löten" versus "Programmieren"





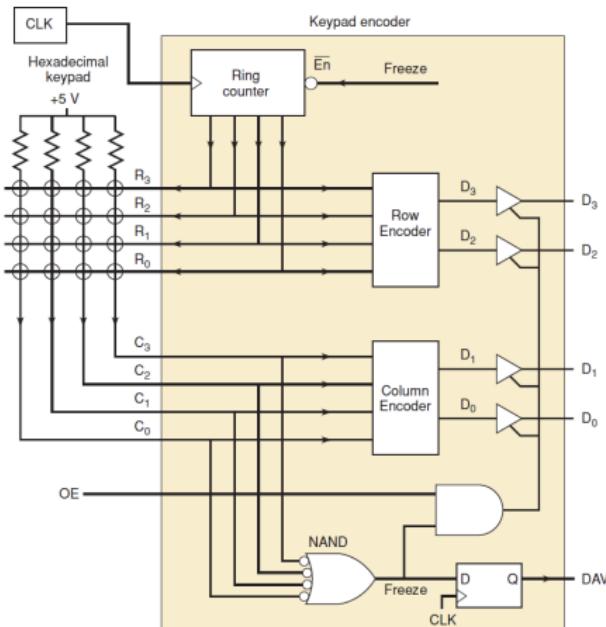
1. Motivation und Einführung
  - ✓ Organisatorisches
  - ✓ Digitale Systeme vs. Rechnerorganisation
  - ❑ Themen Digitale Systeme am Beispiel Keypad Encoder
  - ❑ Streifzug durch die Computertechnik
2. Logik und Boolesche Algebra
3. Gatter und CMOS-Technik
4. Zahlendarstellung und Codes
5. Schaltnetze und Normalformen
6. Optimierung von Schaltnetzen
7. Standard-Schaltnetze
8. Speicherelemente und programmierbare Logik
9. Synchrone Schaltwerke
10. Register-Transfer-Entwurf und Mikroprogrammierung
11. Anwendungen Digitaler Systeme

## ... am Beispiel eines Keypad Encoders



Tocci, Widmer, Moss, 2010, Abb. 10-12.

## ... am Beispiel eines Keypad Encoders

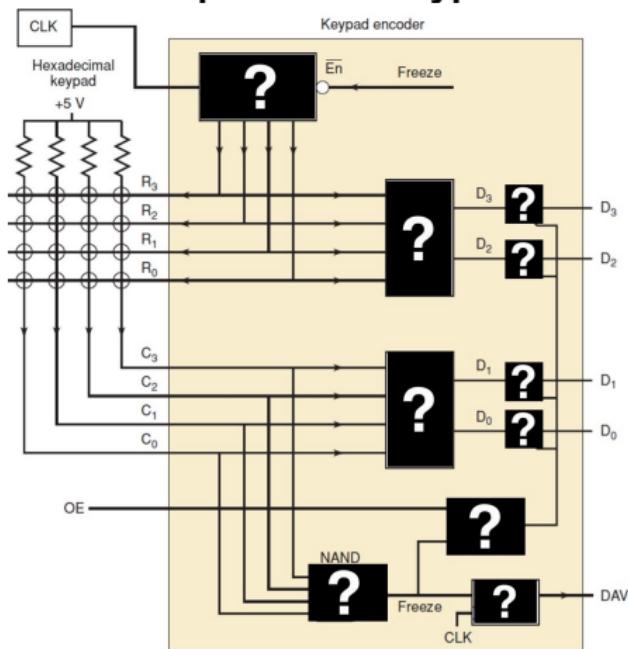


## Funktion und Struktur von digitaler Hardware

- Gatter
- (De-)Kodierer
- FlipFlops
- Zähler
- ...

Schaltung. Aus: Tocci, Widmer, Moss, 2010,  
Abb. 10-11.

## ... am Beispiel eines Keypad Encoders

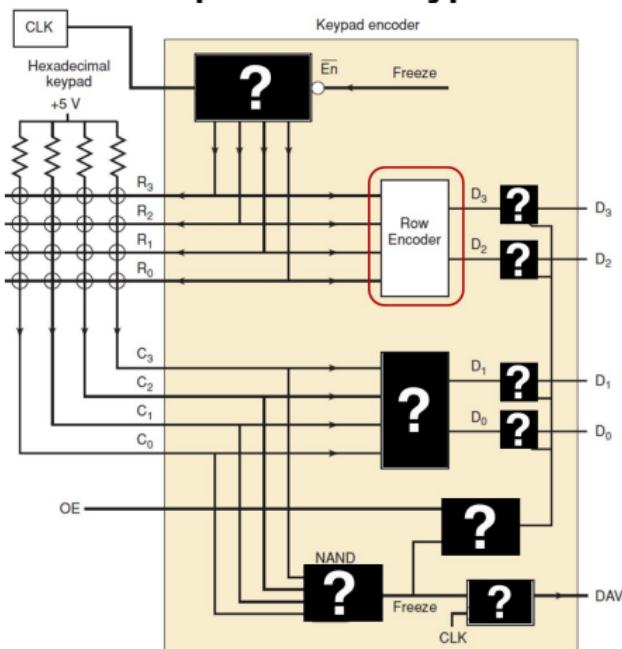


## Funktion und Struktur von digitaler Hardware

- Gatter
- (De-)Kodierer
- FlipFlops
- Zähler
- ...

Schaltung. Nach: Tocci, Widmer, Moss,  
2010, Abb. 10-11.

## ... am Beispiel eines Keypad Encoders



### Blick in den Aufbau digitaler Hardwareelemente

- Transistoren
- CMOS-Technologie
- Schaltnetze
- Speicherelemente

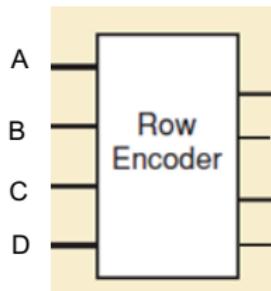
Schaltung. Nach: Tocci, Widmer, Moss,  
2010, Abb. 10-11.



## ... am Beispiel eines Row Encoders

### Spezifikation der Funktion von Hardware

z.B. als logische Funktionen



$$a = D + \overline{A} \overline{C} + A \overline{C} + A B$$

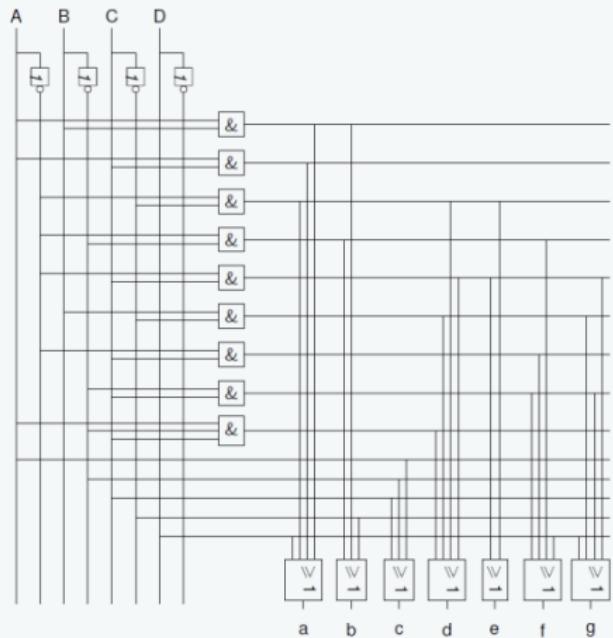
$$b = \overline{C} + A B + \overline{A} \overline{B}$$

$$c = \dots$$

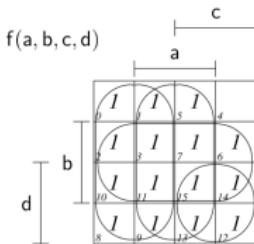
... oder Wahrheitstabellen

| Input |   |   |   | Output |   |   |     |
|-------|---|---|---|--------|---|---|-----|
| D     | C | B | A | a      | b | c | ... |
| 0     | 0 | 0 | 0 | 1      | 1 | 1 | ... |
| 0     | 0 | 0 | 1 | 0      | 1 | 1 | ... |
| 0     | 0 | 1 | 0 | 1      | 1 | 0 | ... |
| 0     | 0 | 1 | 1 | 1      | 1 | 1 | ... |

## ... am Beispiel eines Row Encoders



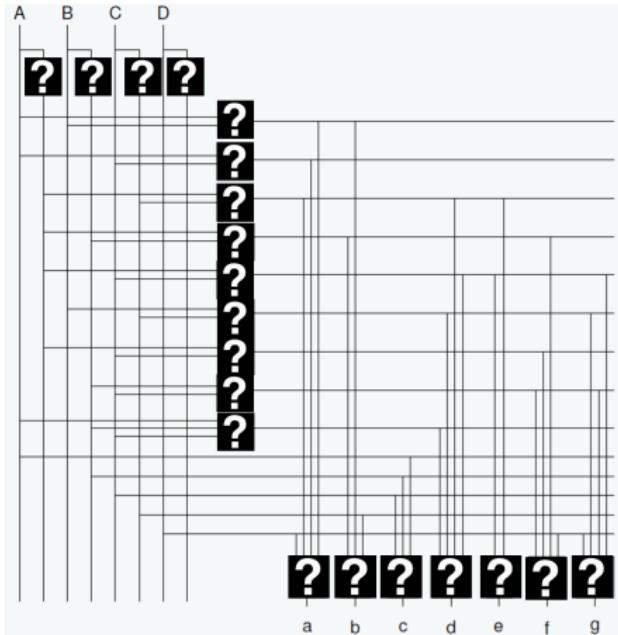
## Umsetzung und Optimierung der Spezifikation



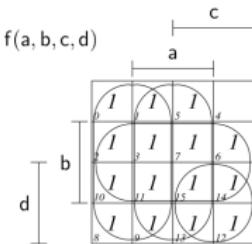
- KV-Tafeln
- kanonische DNF und KNF
- Implikanten, Primimplikanten
- Quine/McCluskey  
bzw. Tison Verfahren



## ... am Beispiel eines Row Encoders

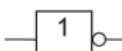
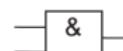
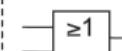
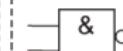
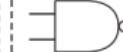


## Umsetzung und Optimierung der Spezifikation



- KV-Tafeln
- kanonische DNF und KNF
- Implikanten, Primimplikanten
- Quine/McCluskey  
bzw. Tison Verfahren

## Gattersymbolik

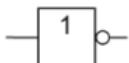
| Negation  | Konjunktion   | Disjunktion   | NAND  | NOR   | Äquivalenz   | Antivalenz  |
|---|---|---|---|---|--|---|
|  |  |  |  |  |  |  |
| DIN 40900  | DIN 40900   |
|  |  |  |  |  |  |  |
| alte Darstellung   | alte Darstellung  |
|  |  |  |  |  |  |  |
| US-Norm   | US-Norm   | US-Norm   | US-Norm   | US-Norm   | US-Norm  | US-Norm   |

Symbolische Darstellung verschiedener Logikgatter. Aus D.W. Hoffmann, 2014, Abb. 5.16.

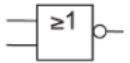
## Gatter als CMOS-Grundschaltungen

Verhalten  
Register-Transfer  
Schaltnetze & Schaltwerke  
Gatter  
Transistoren  
Halbleiter

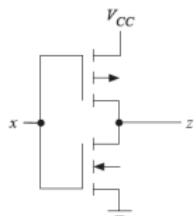
Negation



NOR

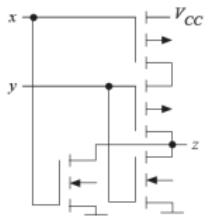


NAND



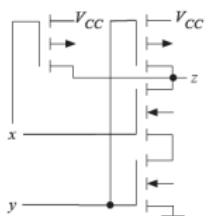
■ Positive Logik (NOT)

| $x$ | $z$ |
|-----|-----|
| 0   | 1   |
| 1   | 0   |



■ Positive Logik (NOR)

| $x$ | $y$ | $z$ |
|-----|-----|-----|
| 0   | 0   | 1   |
| 0   | 1   | 0   |
| 1   | 0   | 0   |
| 1   | 1   | 0   |



■ Positive Logik (NAND)

| $x$ | $y$ | $z$ |
|-----|-----|-----|
| 0   | 0   | 1   |
| 0   | 1   | 1   |
| 1   | 0   | 1   |
| 1   | 1   | 0   |

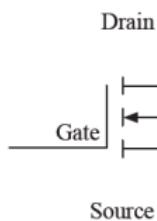
Aus: D.W. Hoffmann, 2014, Abb.5.11



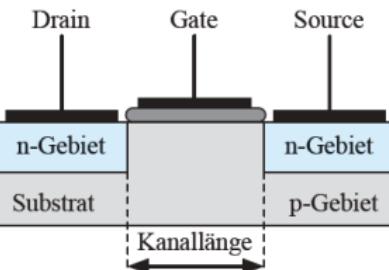
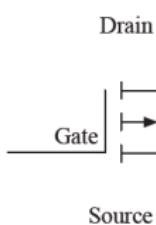
## Transistoren, z.B. MOSFET

Verhalten  
Register-Transfer  
Schaltnetze & Schaltwerke  
Gatter  
Transistoren  
Halbleiter

n-Kanal-MOSFET



p-Kanal-MOSFET



Links: Schaltsymbole; rechts: Interner Aufbau.  
Aus: D.W. Hoffmann, 2014, Abb. 2.25 und 2.26.

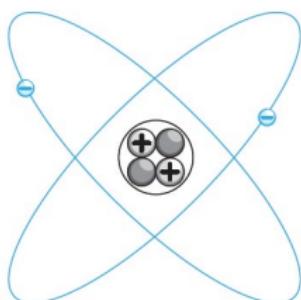
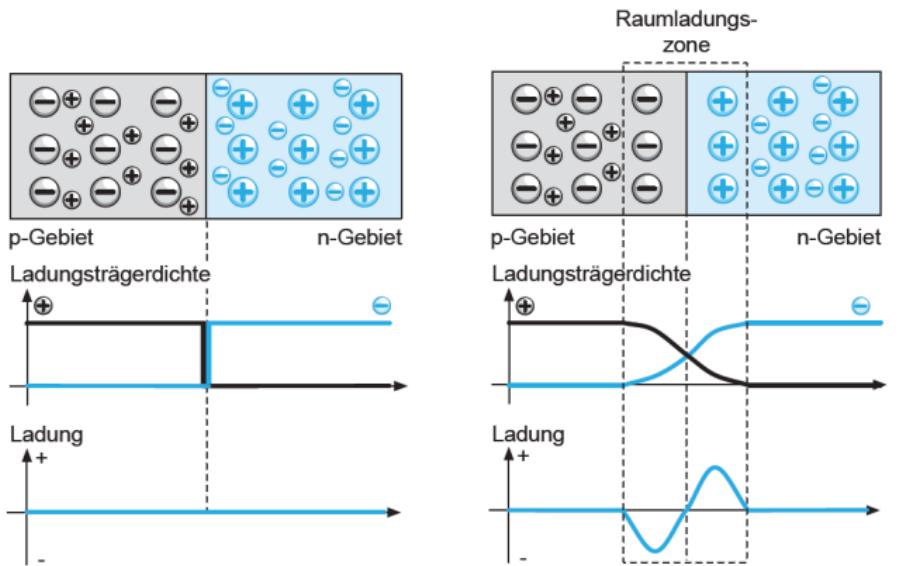
n-Kanal-MOSFET: p-Substrat mit 2 n-dotierten Gebieten

Gate auf isolierendem Dielektrikum



## Kurzer Abstecher in die Halbleitertechnik

Verhalten  
Register-Transfer  
Schaltnetze & Schaltwerke  
Gatter  
Transistoren  
Halbleiter



Hableiterdiode vor (links) und nach (rechts) Diffusion.  
Aus: D.W. Hoffmann, 2014, Abb. 2.2 & 2.12.



## 1. Motivation und Einführung

- ✓ Organisatorisches
- ✓ Digitale Systeme vs. Rechnerorganisation
- ✓ Themen Digitale Systeme am Beispiel Keypad Encoder
- ❑ Streifzug durch die Computertechnik

## 2. Logik und Boolesche Algebra

## 3. Gatter und CMOS-Technik

## 4. Zahlendarstellung und Codes

## 5. Schaltnetze und Normalformen

## 6. Optimierung von Schaltnetzen

## 7. Standard-Schaltnetze

## 8. Speicherelemente und programmierbare Logik

## 9. Synchrone Schaltwerke

## 10. Register-Transfer-Entwurf und Mikroprogrammierung

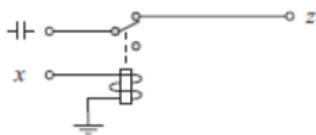
## 11. Anwendungen Digitaler Systeme



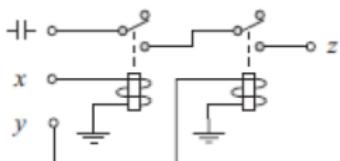
“Chinese-abacus” by Shieldforyoureyes Dave Fischer - Own work. Licensed under CC BY-SA 3.0 via Commons.



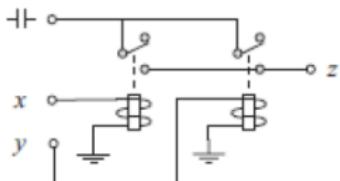
Die NOT-Verknüpfung



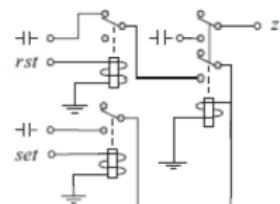
Die AND-Verknüpfung



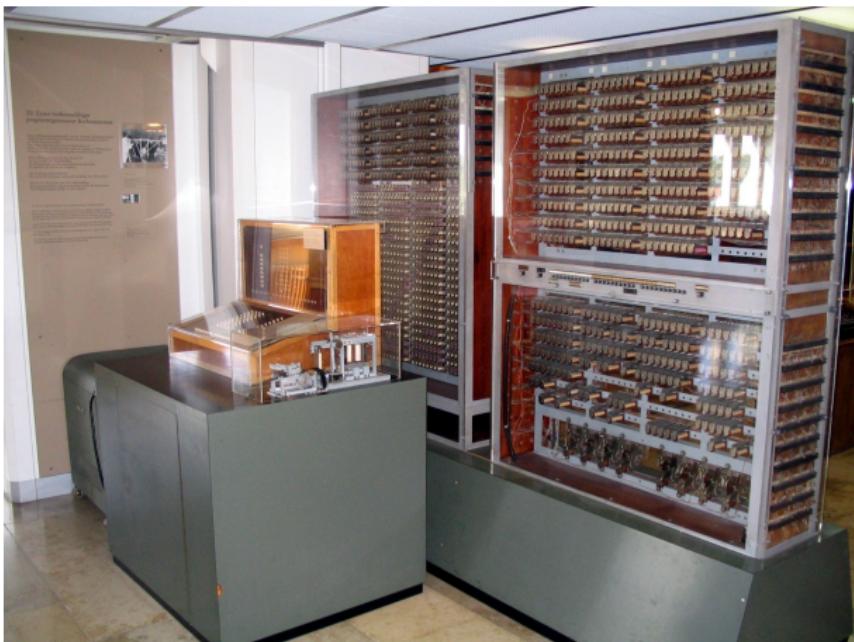
Die OR-Verknüpfung



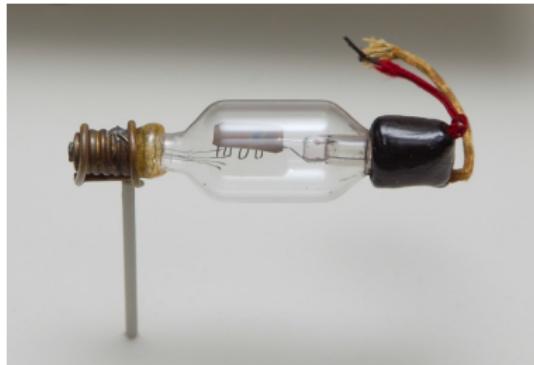
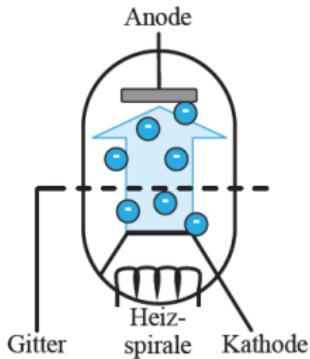
Der Relais-basierte Zustandsspeicher



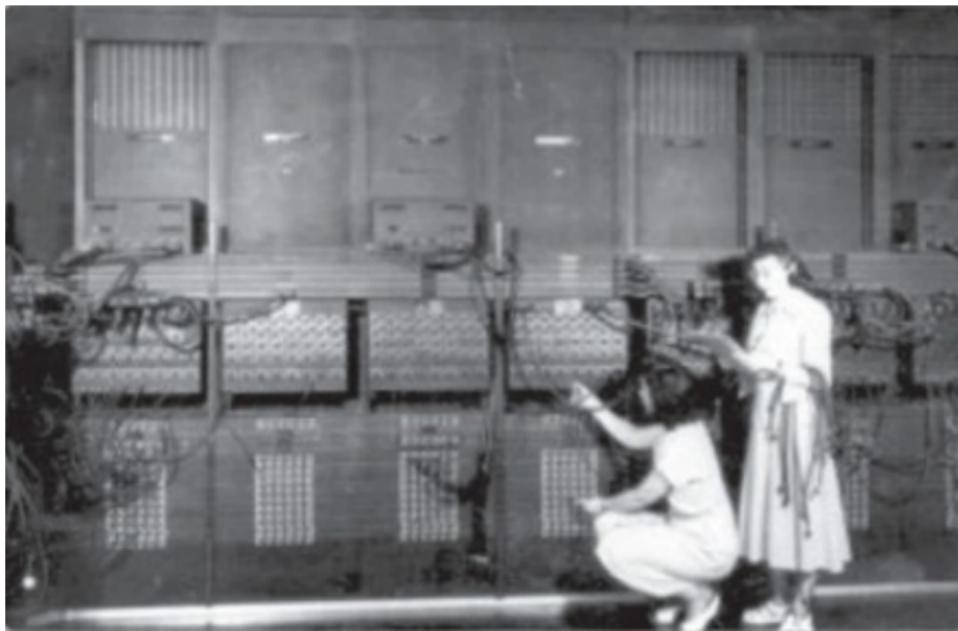
Basiskomponenten Relais-Rechner. Aus: D.W. Hoffmann, 2014, Abb. 1.8.



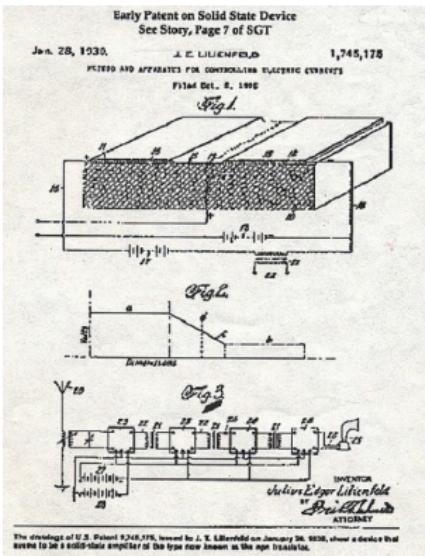
“Z3 Deutsches Museum” von Venusianer aus der deutschsprachigen Wikipedia.  
Lizenziert unter CC BY-SA 3.0 über Wikimedia Commons.



Basiskomponente Trioden-Röhre. Links: aus D.W. Hoffmann, 2014, Abb. 1.14. Rechts: "Triode tube 1906" von Gregory F. Maxwell - Photo by uploader, taken at The History of Audio: The Engineering of Sound, an exhibition of the San Francisco Airport Museums in SFO Airport, Terminal 3 from 2006-09 to 2007-05. Lizenziert unter GFDL 1.2 über Wikimedia Commons.



“Two women operating ENIAC” von United States Army - Image from [1].  
Lizenziert unter Gemeinfrei über Wikimedia Commons.



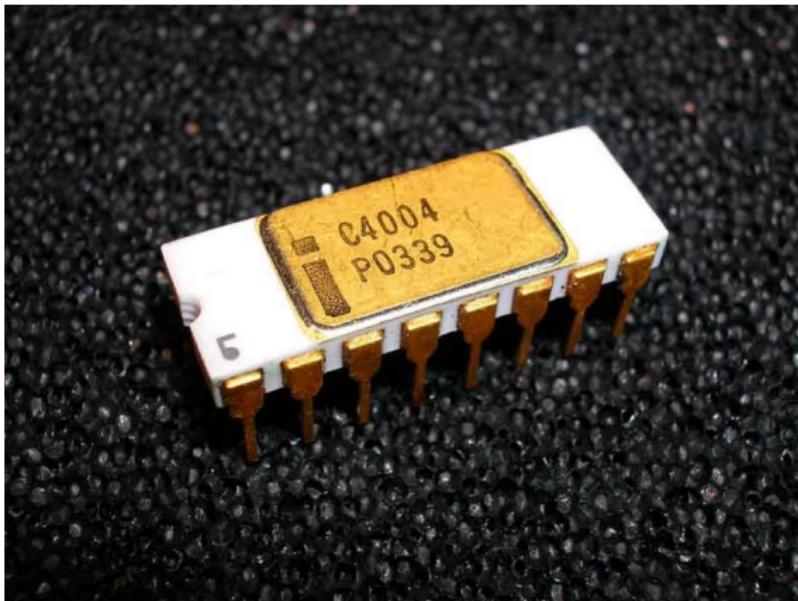
Basiskomponente Transistor. Links: "Replica-of-first-transistor". Licensed under Public Domain via Commons. Rechts: Patentschrift Julius Lilienfeld, 1928.



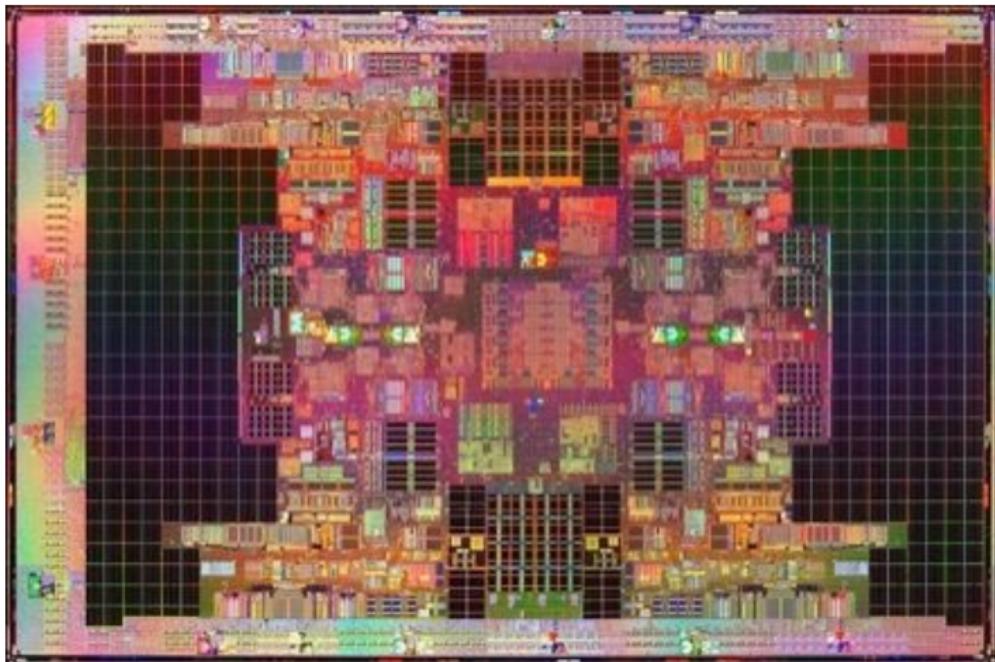
| Erste Generation        |                         | Zweite Generation       | Dritte und vierte Generation |
|-------------------------|-------------------------|-------------------------|------------------------------|
| 1940 – 1954             |                         | 1955 – 1964             | ab 1965                      |
| Relais                  | Vakuumröhre             | Transistor              | Integrierter Schaltkreis     |
| Schaltzeit: $10^{-1}$ s | Schaltzeit: $10^{-4}$ s | Schaltzeit: $10^{-6}$ s | Schaltzeit: $< 10^{-9}$ s    |

The table compares four generations of electronic components. The first generation (1940-1954) includes Relais and Vakuumröhre, with a switching time of  $10^{-1}$  s. The second generation (1955-1964) includes Transistor, with a switching time of  $10^{-6}$  s. The third and fourth generations (from 1965 onwards) include Integriertes Schaltkreis (Integrated Circuit), with a switching time of  $< 10^{-9}$  s.

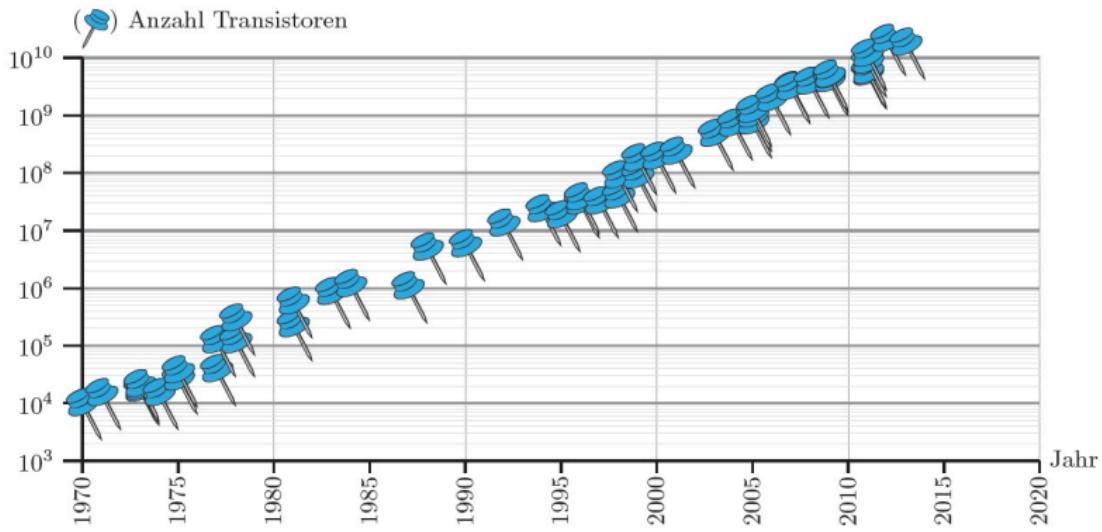
(Hoffmann, Grundlagen der Technischen Informatik, Hanser, 2014, Tab. 1.3)



“C4004 two lines” von Peter1912 in der Wikipedia auf Deutsch - Selbst fotografiert, übertragen aus de.wikipedia nach Commons durch Wdwd mit Hilfe des CommonsHelper. Lizenziert unter Gemeinfrei über Wikimedia Commons.

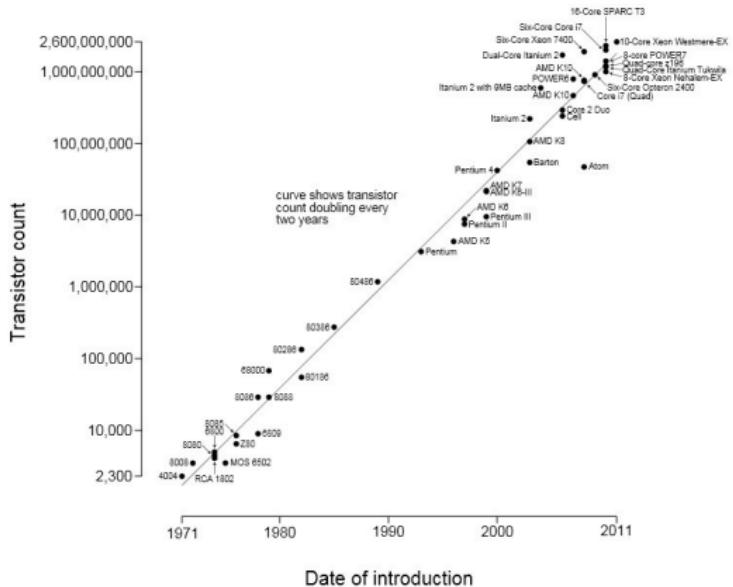


Intel Itanium 2.



Entwicklung der Hardware-Komplexität. Aus: Hoffmann, Grundlagen der Technischen Informatik, Hanser, 2014, Abb. 1.25

Microprocessor Transistor Counts 1971-2011 & Moore's Law



“Transistor Count and Moore’s Law - 2011” by Wgsimon - Own work. Licensed under CC BY-SA 3.0 via Commons.



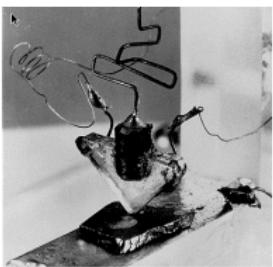


- Beispiel **Intel Itanium 2 Prozessor**
- Design Team 90 Personen,  $1.72 \cdot 10^9$  Transistoren
- $\frac{1.72 \cdot 10^9 \text{ Transistoren}}{90 \text{ Entwickler}} = \text{ca. } 19 \cdot 10^6 \frac{\text{Transistoren}}{\text{Entwickler}}$
- Design-Start: 24. September 2001, erste Samples Ende 2004
- ca. 36 Monate Entwurfszeit
- ca.  $531.000 \frac{\text{Transistoren}}{\text{Entwickler} \cdot \text{Monat}}$ , ca.  $26500 \frac{\text{Transistoren}}{\text{Entwickler} \cdot \text{Tag}}$
- 1 Transistor, TO92 Gehäuse, ca.  $\frac{0.000125 \text{ Liter}}{\text{Transistor}}$
- $\frac{0.000125 \text{ Liter}}{\text{Transistor}} \cdot 531.000 \frac{\text{Transistoren}}{\text{Entwickler} \cdot \text{Monat}} = 66 \frac{\text{Liter}}{\text{Entwickler} \cdot \text{Monat}}$
- das sind über 18 Müllsäcke a 120 Liter pro Entwickler
- oder 2160 Liter Transistoren pro Entwickler



## Entwurfskosten

“[..] transistors were difficult to produce. Only one in five transistors that were produced worked as expected (only a **20%** yield) and as a result the price remained extremely high. When it was released in 1954, the Regency TR-1 cost \$49.95 (roughly **\$364** in 2006 USD) and sold about 150,000 units. In 1955, Raytheon’s 8-TR-1 was priced at \$80 (approximately **\$425** in 1994 USD).”

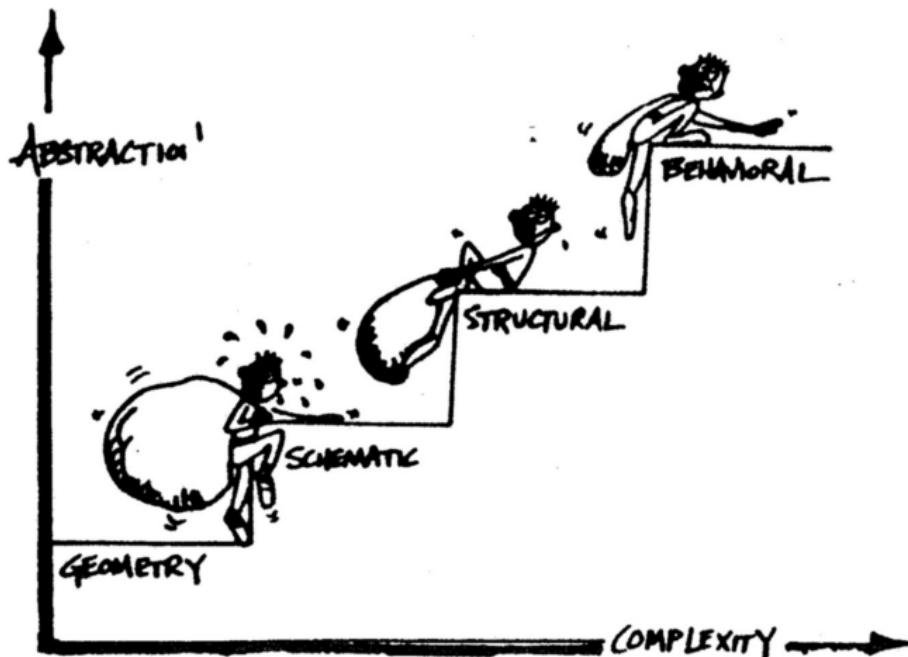




- Entwurfskosten
  - Kosten eines einzelnen Transistors heute unerheblich
  - Intel Xeon X7560 mit 8 Kernen
  - ca. 2.300.000.000 Transistoren
  - ca. \$3700 Listenpreis
  - \$0.000001605 pro Transistor
- Problematisch heute: Kosten für die Fertigung (Lithographie)
  - eine “Fab” kostet > \$4 Milliarden
  - ein “Maskensatz” kostet in 65 nm ca. > \$3 Millionen
  - im Durchschnitt sind 2.5 “Spins” notwendig
  - Funktionstüchtiges Design > \$7 - \$8 Millionen
  - Entwicklungskosten (Personal, Werkzeuge) nicht enthalten
- Moderne Technologie bietet Platz für mehrere Milliarden Transistoren
- Chip-Größe trotzdem Gegenstand der Optimierung



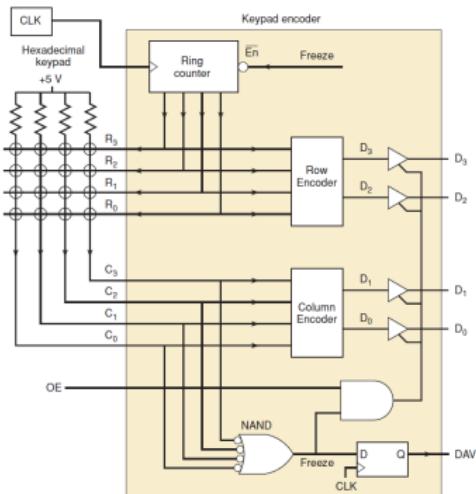
Komplexitätsbeherrschung durch Abstraktion





## Komplexitätsbeherrschung durch Abstraktion

| Ebene                                   | Verhaltensdomäne   | Strukturdomäne   | Physikalische Domäne                 |
|---|--|--|--------------------------------------|
| System                                  | Spezifikation  | Controller, Speicher, Busse, Schnittstellen              | Rack, Board, Multi-Chip-Module, Chip |
| Chip                                    | Algorithmus  | Mikroprozessor, RAM, ROM, UART, Ethernet, DMA-Controller | Chips, Floorplan                     |
| Registertransfer<br>Rechnerorganisation | Datenfluss   | Register, ALU, Zähler, Multiplexer, Dekoder              | Zell-Layout                          |
| Gatter<br>Digitale Systeme              | Boolesche Gleichungen  | AND, OR, NAND, NOR, XOR, DFF, JKFF                       | Gatter-Layout                        |
| Schaltung <sup>a</sup>                  | Differentialgleichungen, Ersatzschaltbilder                          | Transistor, R, C, L                                      | Symbolisches Layout                  |
| Transistor/Layout                       | Lösung der Schrödinger-Gleichung beschreibt Elektronen-/Lochbewegung | Geometrische Objekte, Lagen                              | Masken-Layout                        |



Aus: Tocci, Widmer, Moss, 2010, Abb. 10-11.

- ✓ Organisation und Ansprechpartner
  - ✓ Vorlesung
  - ✓ Übung & Hausaufgaben & ISIS
  - ✓ Klausur & Prüfungsform
  
- ✓ Worum geht es in "Digitale Systeme"?
  
- ✓ Streifzug durch die Computertechnik



<http://www.magicbluesmoke.org>