

# Grundlagen der Rechnerarchitektur und Betriebssysteme

## Logikgatter

**Florian Knoch** · Lehrstuhl für Praktische Informatik,  
insbes. Systemnahe Programmierung, Universität Bamberg

# Vorab: Silikon != Silizium



Damit ihr diesen Fehler vermeidet (oder korrigiert)

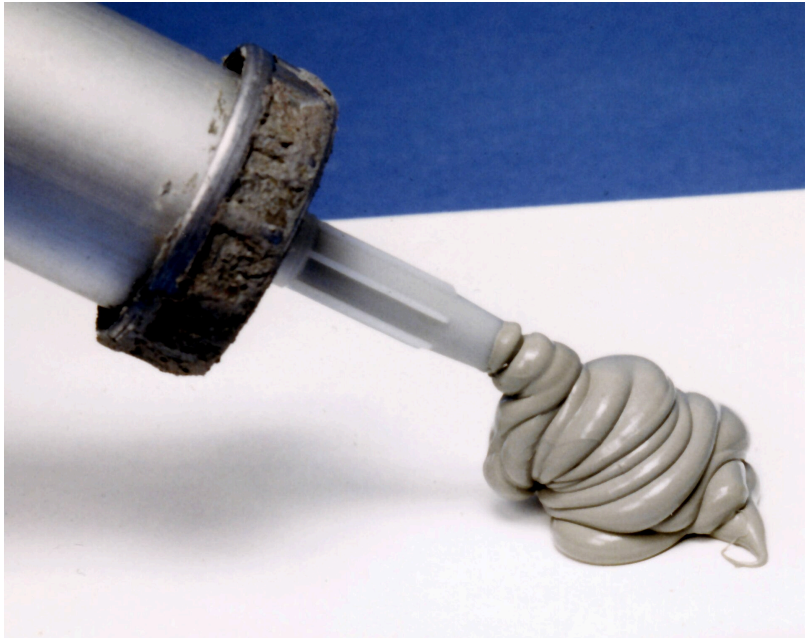


Abbildung 1: Silikon

Achim Hering – Own work, CC BY 3.0



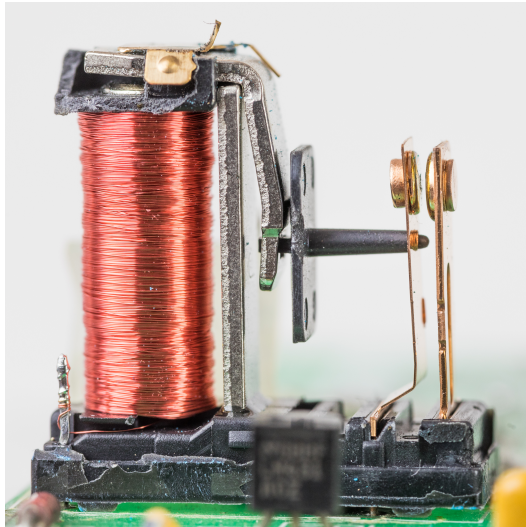
Abbildung 2: Silizium (engl. 'silicon')

**Was ist ein Bit?**

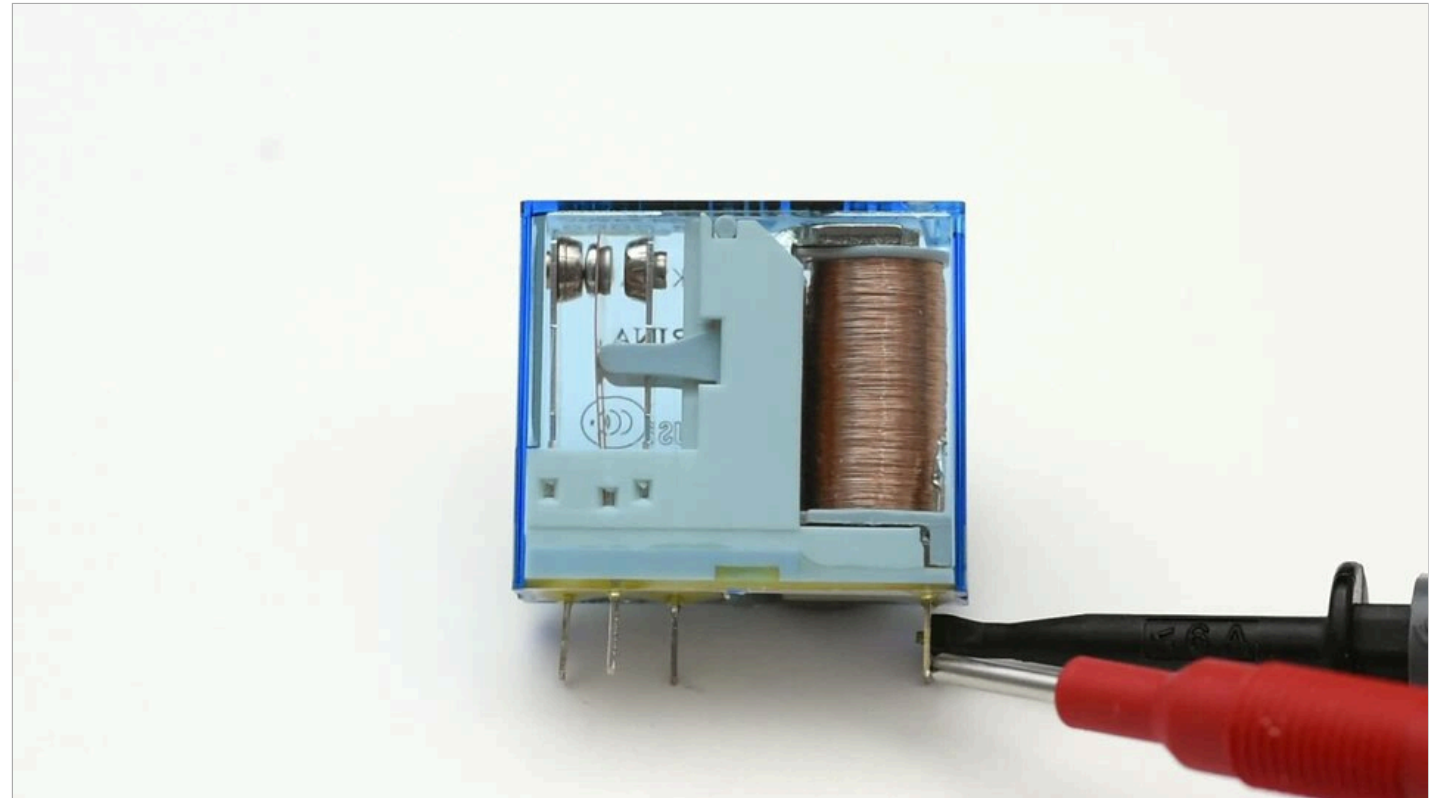
# Relais



Ein durch elektrischen Strom betriebener, fernbetätigter Schalter.



© Raimond Spekking  
via Wikimedia Commons (CC BY SA 4.0)



© Simon A. Eugster via Wikimedia Commons (CC BY-SA 4.0)

# Feldeffekttransistor



Ein Bauteil, bei dem eine Spannung am Gate die Leitfähigkeit zwischen Source und Drain beeinflusst.

Heute verbreitet: metal-oxide-semiconductor field-effect transistor (MOSFET)

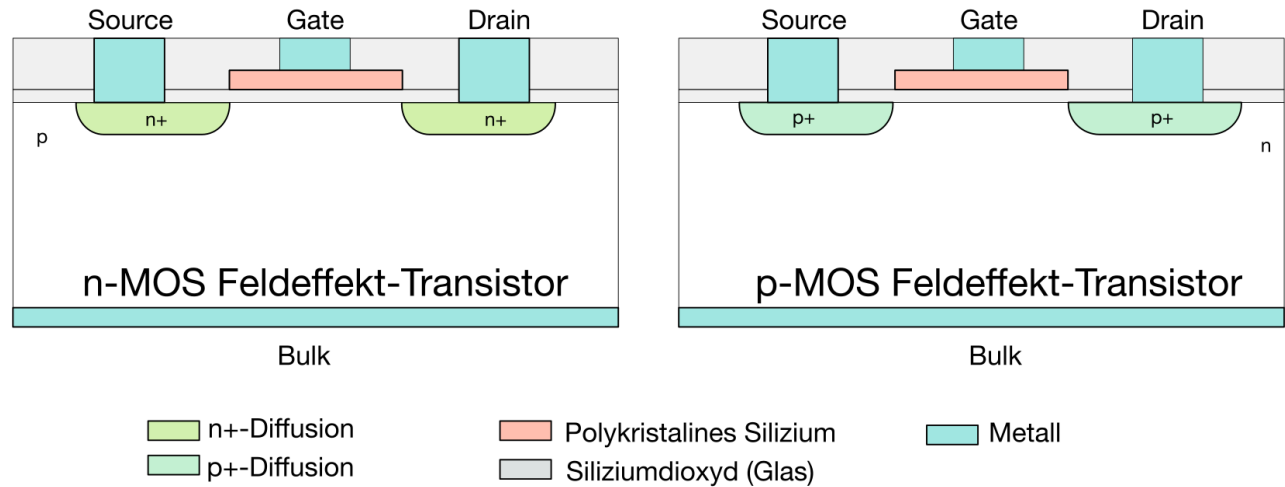


Abbildung 3: Struktureller Aufbau der MOS-Feldeffekttransistoren

© Abbildung 4.33 aus: Frank Slomka & Michael Glaß: [Grundlagen der Rechnerarchitektur](#). Springer Vieweg, 2023.

# Moderne(re) Halbleiterfertigung

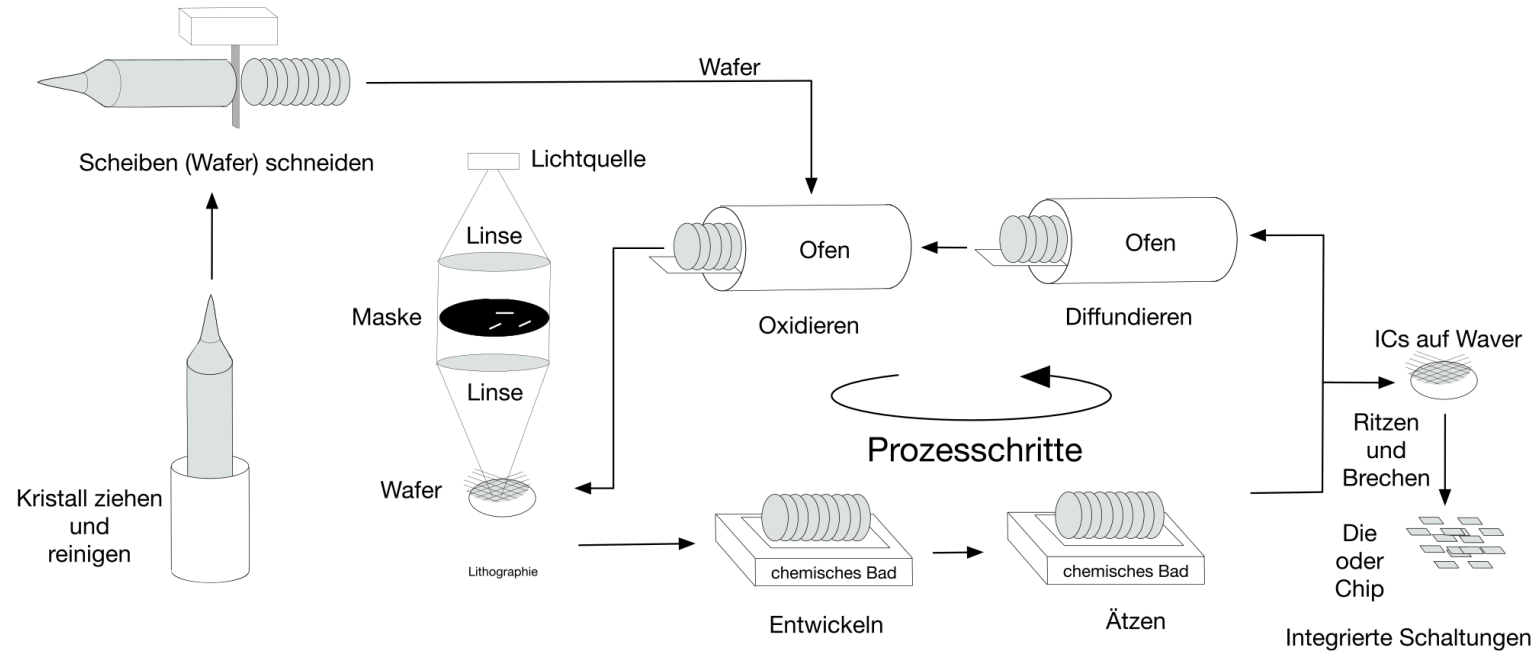


Abbildung 4: Schritte in der Halbleiterfertigung

© Abbildung 4.33 aus: Frank Slomka & Michael Glaß: [Grundlagen der Rechnerarchitektur](#). Springer Vieweg, 2023.

# Moderne(re) Halbleiterfertigung

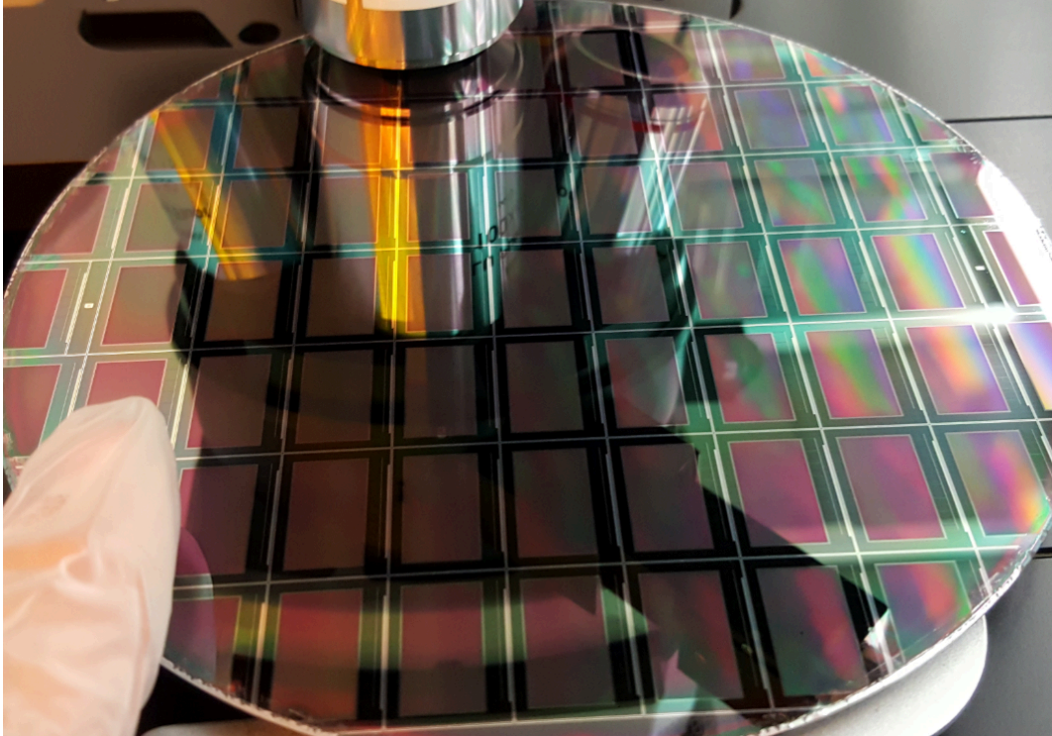


Abbildung 5: Ein einfacher Wafer

© Palee93 via [Wikimedia Commons](#) (CC BY-SA 4.0)

# Moderne(re) Halbleiterfertigung

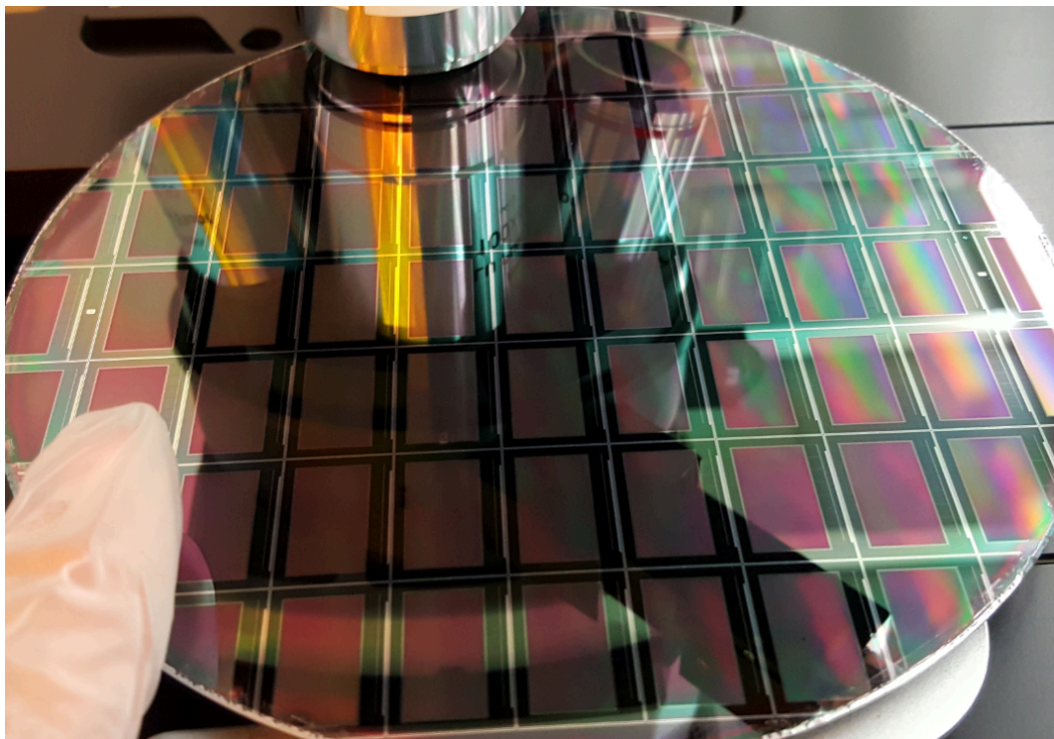


Abbildung 6: Ein einfacher Wafer

© Palee93 via [Wikimedia Commons](#) (CC BY-SA 4.0)

How to draw an owl

1.



1. Draw some circles

2.



2. Draw the rest of the fucking owl

# Ein einfacher Inverter

Nichts Besonderes, oder?

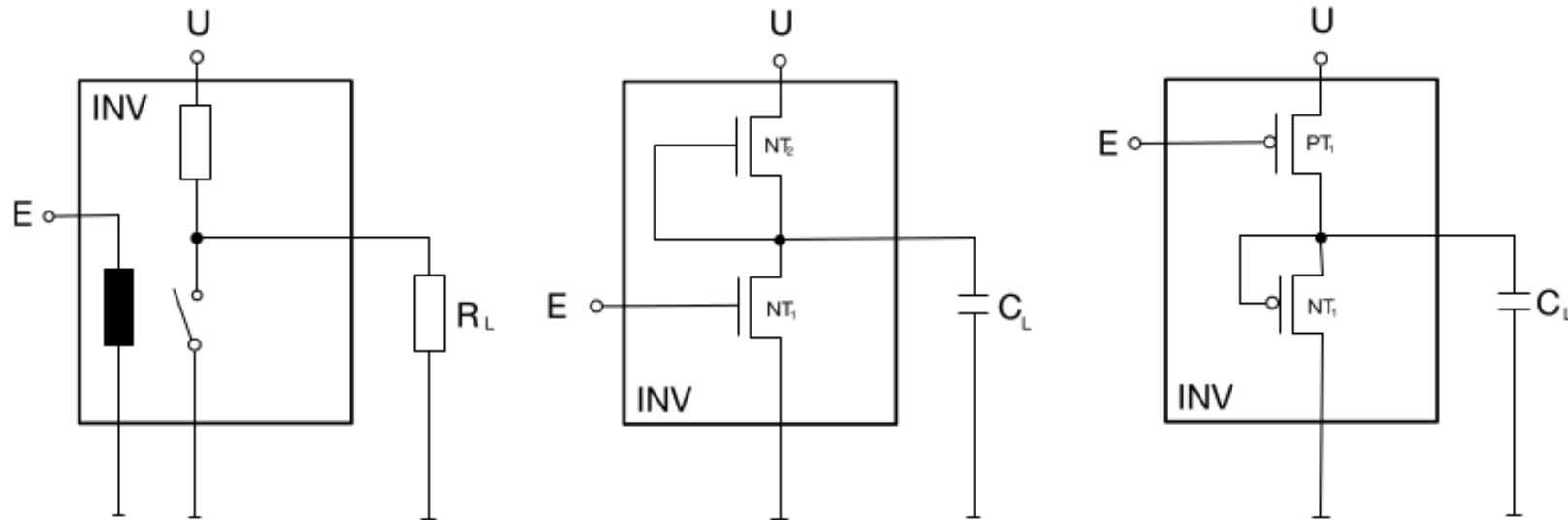


Abbildung 7: Inverter mit einem Relais, in NMOS- und PMOS-Technik

© Abbildung 4.39 aus: Frank Slomka & Michael Glaß: [Grundlagen der Rechnerarchitektur](#). Springer Vieweg, 2023.

# Doch eher Magie?



1

**Bei NMOS und PMOS fließt  
in jedem Zustand Strom!**

eine Lösung: CMOS

**WEITERFÜHREND**

EESYS-GEI-B

# Doch eher Magie?



1

Bei NMOS und PMOS fließt  
in jedem Zustand Strom!

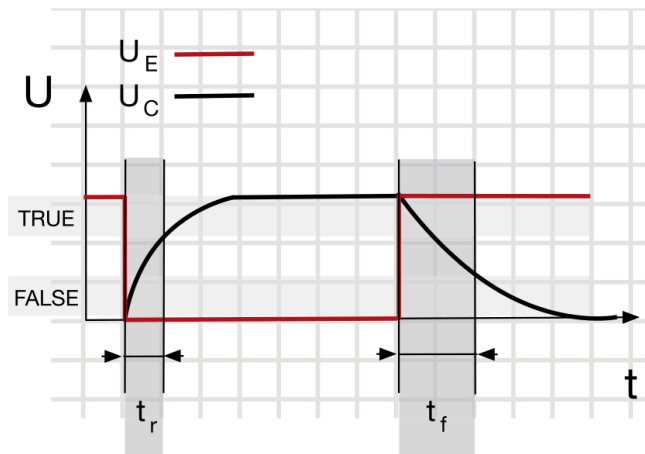
eine Lösung: CMOS

WEITERFÜHREND

EESYS-GEI-B

2

Der Wechsel zwischen 0 und  
1 ist gar nicht binär.



© Abbildung 4.45 aus: Frank Slomka & Michael Glaß:  
**Grundlagen der Rechnerarchitektur**. Springer Vieweg, 2023.

# Doch eher Magie?



1

Bei NMOS und PMOS fließt  
in jedem Zustand Strom!

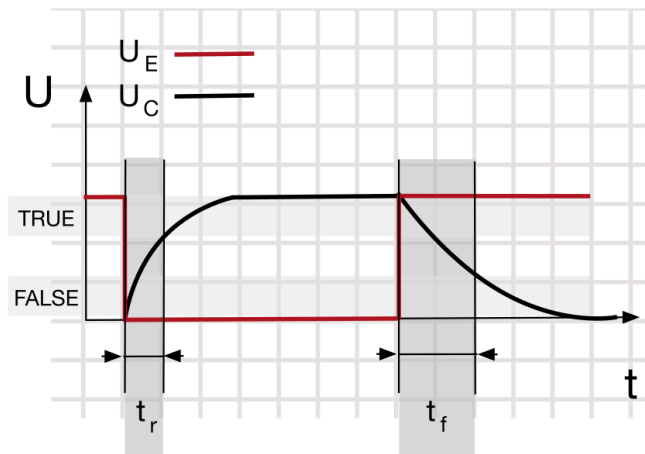
eine Lösung: CMOS

WEITERFÜHREND

EESYS-GEI-B

2

Der Wechsel zwischen 0 und  
1 ist gar nicht binär.



© Abbildung 4.45 aus: Frank Slomka & Michael Glaß:  
[Grundlagen der Rechnerarchitektur](#). Springer Vieweg, 2023.

3

Wir arbeiten auf  
Nanometerebene.

Das ist sehr klein.

# Einfache Schaltungen

# Gatter-Baukasten

## Grundbausteine



### AUFGABE 1A

Erstellen Sie die vier grundlegenden Logikelemente NOT (auch Inverter oder INV), AND, OR und XOR mit Hilfe von NAND-Gattern. Rufen Sie dazu die Webseite <https://nandgame.com> auf und lösen die ersten fünf Level.

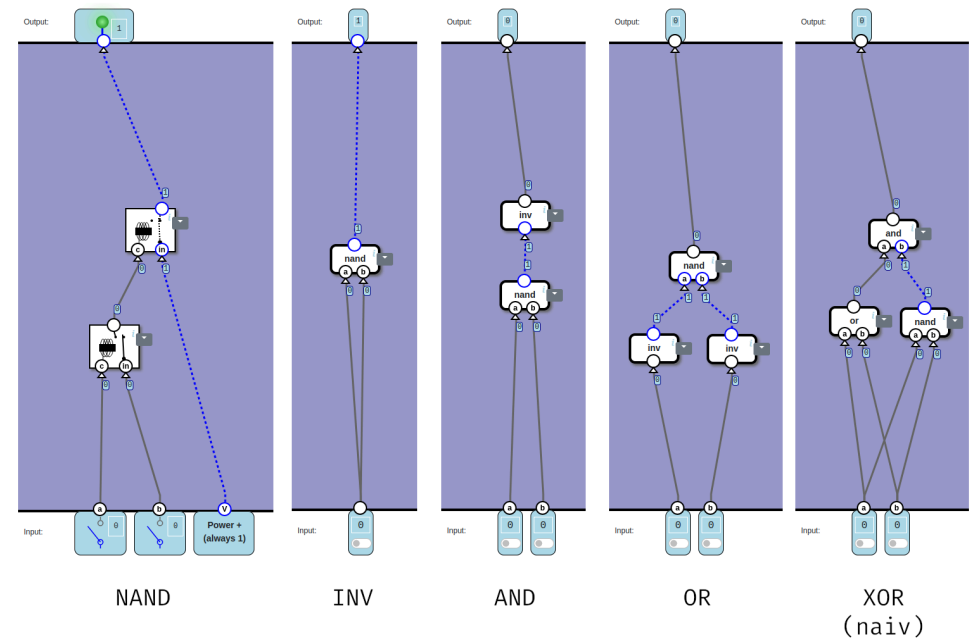
# Gatter-Baukasten

## Grundbausteine



### AUFGABE 1A

Erstellen Sie die vier grundlegenden Logikelemente NOT (auch Inverter oder INV), AND, OR und XOR mit Hilfe von NAND-Gattern. Rufen Sie dazu die Webseite <https://nandgame.com> auf und lösen die ersten fünf Level.



© Olav Junker Kjær, The Nand Game

# Gatter-Baukasten

Leichte Optimierung

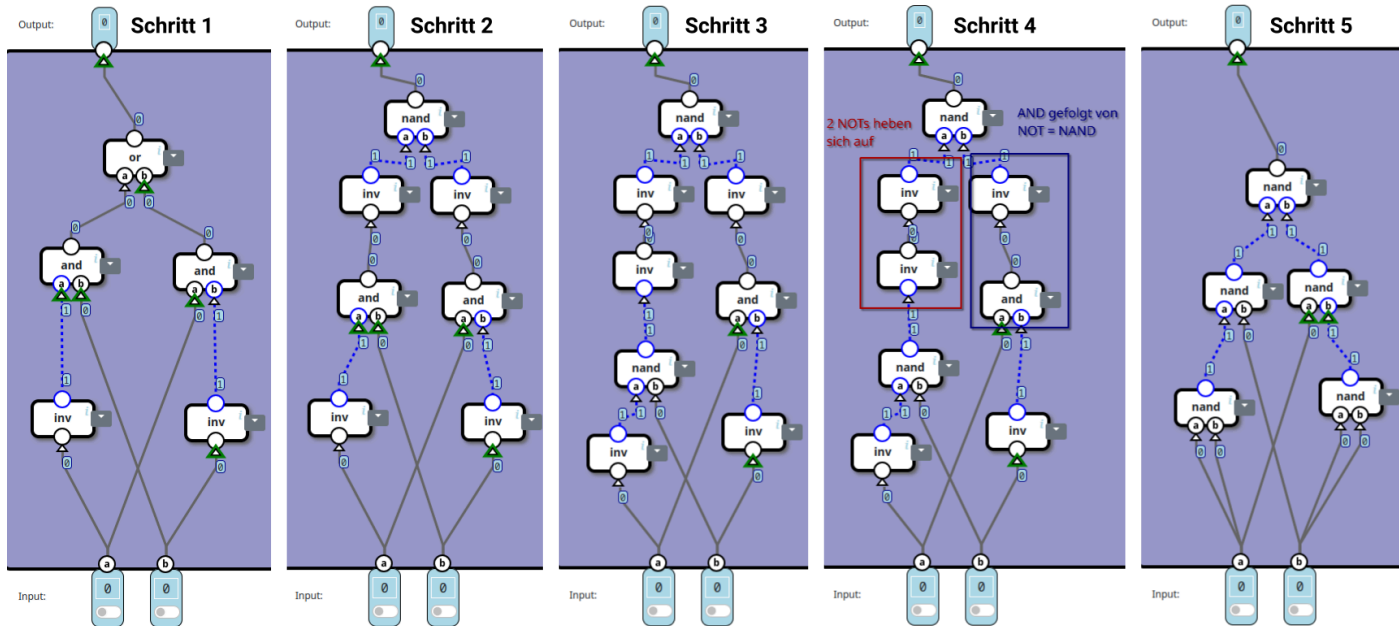


## AUFGABE 1B

Jedes NAND-Gatter ist ein Kostenfaktor bei der Chipherstellung. Minimieren Sie, falls nötig, ihren Entwurf für das XOR-Gatter, sodass höchstens 5 NAND-Gatter dafür nötig sind.

# Gatter-Baukasten

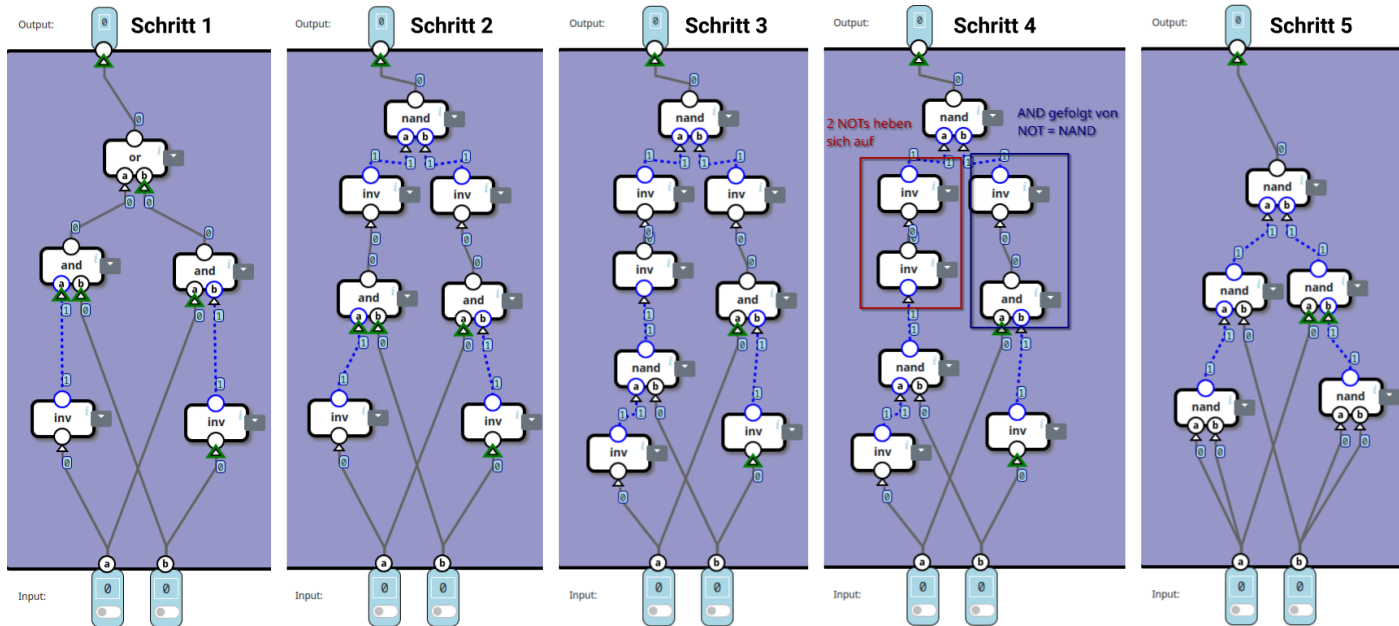
## Leichte Optimierung



© Olav Junker Kjær, The Nand Game

# Gatter-Baukasten

## Leichte Optimierung



**WEITER GEDACHT**

Geht da noch mehr?  
Wie wäre es mit einem  
XOR aus 4 NANDs?

# Organisatorisches



- **Lehrbücher:** Harris & Harris jetzt **online** verfügbar, bald im Handapparat
- **Prüfungswiederholung:** Nicht bestanden im Erstversuch? Wir unterstützen gern persönlich, damit es dieses Mal klappt. Buch gern einen Sprechstundentermin.
- **Tutorienumfrage:** Abstimmung bis Samstagnachmittag, Ergebnisse ab Samstagabend

# Semesterplan



	Montag	Dienstag	Mittwoch	Donnerstag	Freitag	
KW 16	[V00]			[V01] [Ü01]		13.04.–17.04.
KW 17	[V02] [T1.01]	[T2.01]	[T3.1]	[V03] [Ü02]		20.04.–24.04.
KW 18	[V04] [T1.02]	[T2.02]	[T3.2]	[V05] [Ü03]	Tag der Arbeit	27.04.–02.05.
KW 19	[V06] [T1.03]	[T2.03]	[T3.3]	[V07] [Ü04]		04.05.–08.05.
KW 20	[V08] [T1.04]	[T2.04]	[T3.4]	Himmelfahrt		12.05.–15.05.
KW 21	[V09] [T1.05]	[T2.05]	[T3.5]	[V10] [Ü05]		18.05.–22.05.
KW 22	Pfingstmontag	Frei		[V11] [Ü06]		25.05.–29.05.
KW 23	[V12] [T1.06]	[T2.06]	[T3.6]	Fronleichnam	Frei	02.06.–05.06.
KW 24	[V13] [T1.07]	[T2.07]	[T3.07]	[V14] [Ü07]		08.06.–12.06.
KW 25	[V15] [T1.08]	[T2.08]	[T3.08]	[V16] [Ü08]		15.06.–19.06.
KW 26	[V17] [T1.09]	[T2.09]	[T3.09]	[V18] [Ü09]	WIAI 25	22.06.–26.06.
KW 27	[V19] [T1.10]	[T2.10]	[T3.10]	[V20] [Ü10]		29.06.–03.07.
KW 28	[V21] [T1.11]	[T2.11]	[T3.11]	[V22] [Ü11]		06.07.–10.07.
KW 29	[V23] [T1.12]	[T2.12]	[T3.12]	[V24] [Ü12]		13.07.–17.07.
KW 30						20.07.–24.07.
KW 31		Klausur				27.07.–32.07.

[V] = Vorlesung, [Ü] = Übung, [T] = Tutorium)

# Hinweise zur Nutzung von LLMs



- Informationsmaterial unter <https://www.uni-bamberg.de/cio/ki/>
- in GRABS gibt es keine explizite KI-Policy
- **Folien dürfen an LLMs gegeben werden, Video- und Tonaufzeichnungen nicht**
- in der Klausur steht selbstverständlich kein LLM zur Verfügung
- starke Empfehlung: **Arbeitet weitestgehend ohne LLMs.**
- bei Fragen steht euch das Lehrstuhl-Team zur Verfügung

**Fragen?**



## [Kursmaterialien](#) · [Quelltext](#)

Diese Präsentation wurde zuletzt am 22.04.2026 bearbeitet. Sie basiert auf den Folien der zugehörigen Vorlesung von Prof. Dr. Michael Engel. Sofern nicht anders angegeben, sind die Inhalte unter der **Lizenz CC BY-SA 4.0** verfügbar. Das Universitätslogo sowie die Schriftart UB Scala Sans sind Eigentum der Otto-Friedrich-Universität Bamberg.

Folgende **Open-Source-Komponenten** sind Teil der Präsentation:

[typst](#) (Apache 2.0), [typst-ccicons](#) (MIT), [typst-polylux](#) (MIT), [fontawesome](#) (SIL).

