



Digitalelektronik 4

Vom Transistor zum Bit

Stefan Rothe

2015-04-21



Rechtliche Hinweise


Dieses Werk von Thomas Jampen und Stefan Rothe steht unter einer *Creative Commons Attribution-Non-Commercial-ShareAlike*-Lizenz.





Zudem verzichten die Autoren auf sämtliche Urheberrechtsansprüche für die in diesem Werk enthaltenen Quelltexte.




Quellenangaben

 Diagramm eines NMOS-FET von Olivier Deleage, angepasst durch Stefan Rothe, Quelle: [Wikimedia Commons](#), Lizenz: [Creative Commons Attribution ShareAlike](#) 5

 Diagramm eines NMOS-FET von Olivier Deleage, angepasst durch Stefan Rothe, Quelle: [Wikimedia Commons](#), Lizenz: [Creative Commons Attribution ShareAlike](#) 5

 Diagramm eines NMOS-FET von Olivier Deleage, angepasst durch Stefan Rothe, Quelle: [Wikimedia Commons](#), Lizenz: [Creative Commons Attribution ShareAlike](#) 6

 Diagramm eines PMOS-FET von Olivier Deleage, angepasst durch Stefan Rothe, Quelle: [Wikimedia Commons](#), Lizenz: [Creative Commons Attribution ShareAlike](#) 6

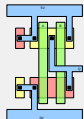


Die shot of Intel Pentium MMX with P55C core (166 MHz SL27K) von [Pauli Rautakorpi](#), Quelle: [Wikimedia Commons](#), Lizenz: [Creative Commons Attribution](#) 10



George Boole, ca. 1860 von Unbekanntem Autor, Quelle: [Wikimedia Commons](#), Lizenz: [Public](#)

Domain 11

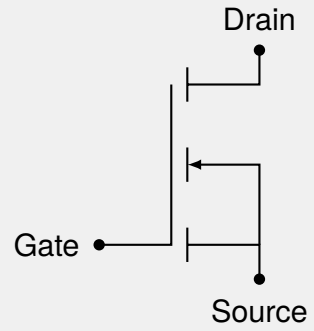


The physical layout of a NAND circuit von [Jamesm76](#), Quelle: [Wikimedia Commons](#), Lizenz: [Public](#)

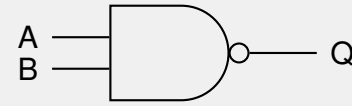
Domain 20



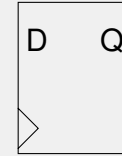
Inhalt



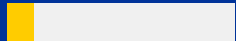
Transistor



Logikgatter



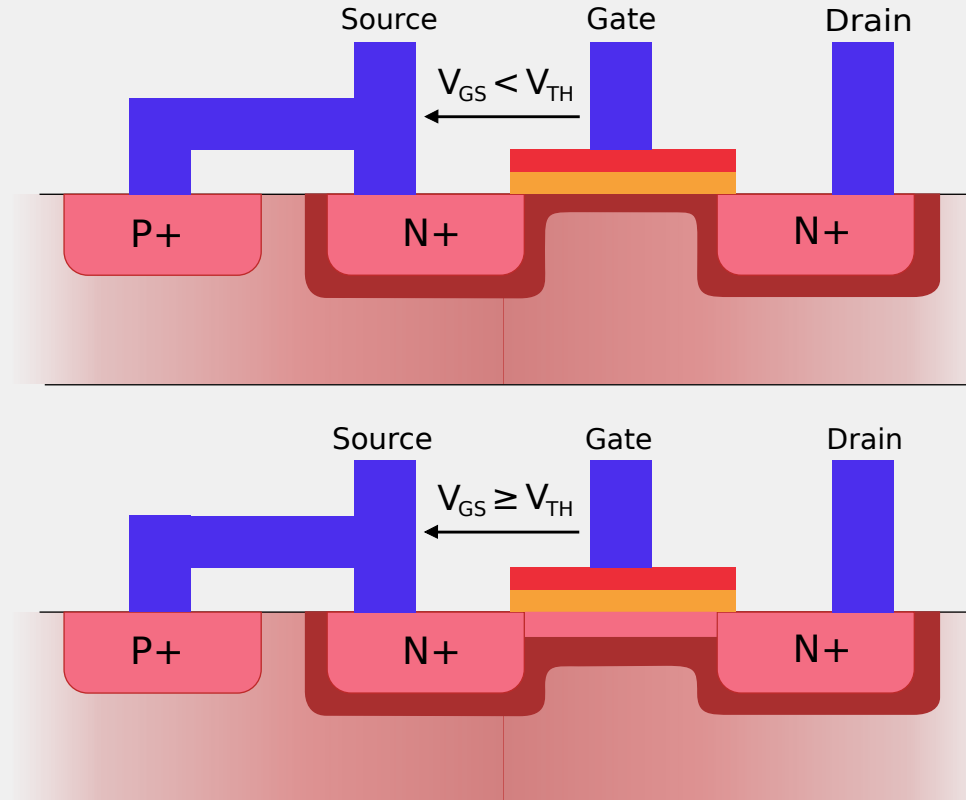
Flipflop





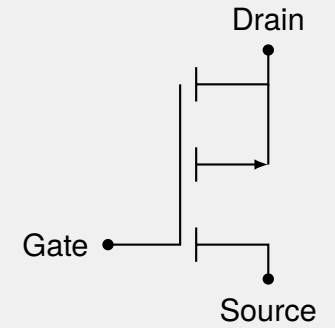
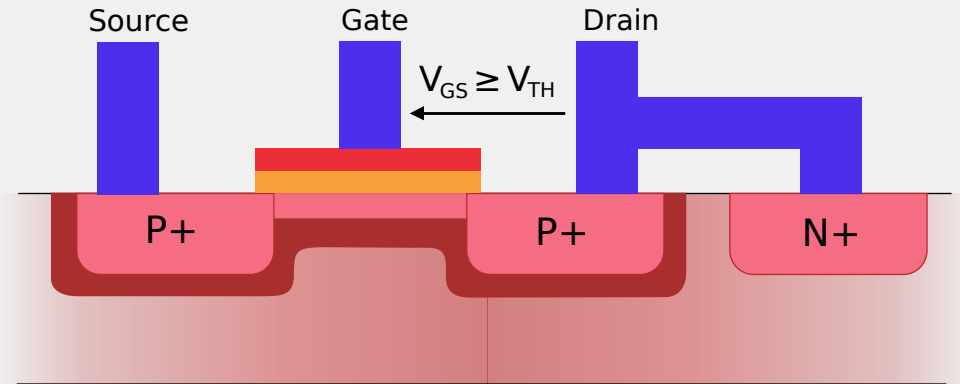
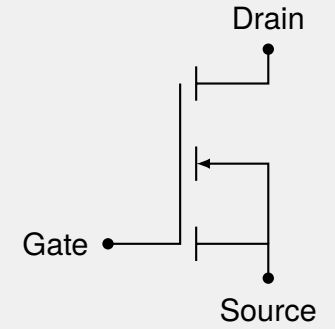
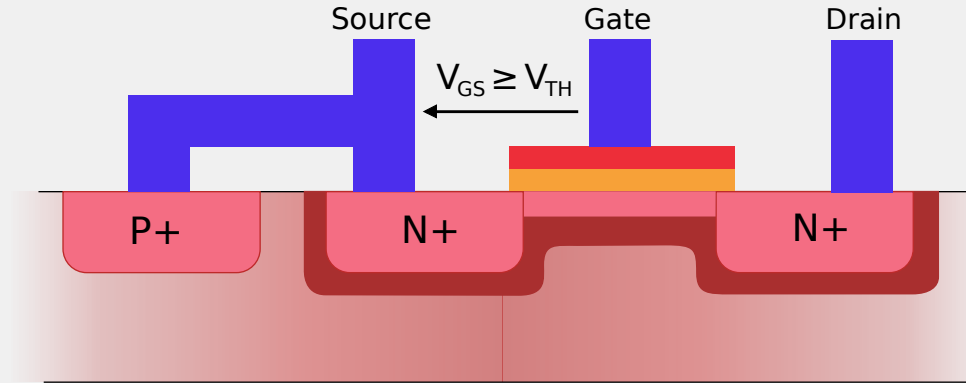
Metalloxid-Halbleiter-Feldeffekttransistor

englisch: *Metal Oxide Semiconductor Field Effect Transistor (MOS FET)*





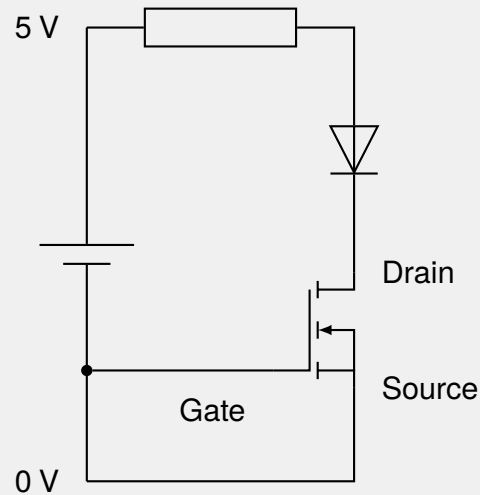
NMOS und PMOS



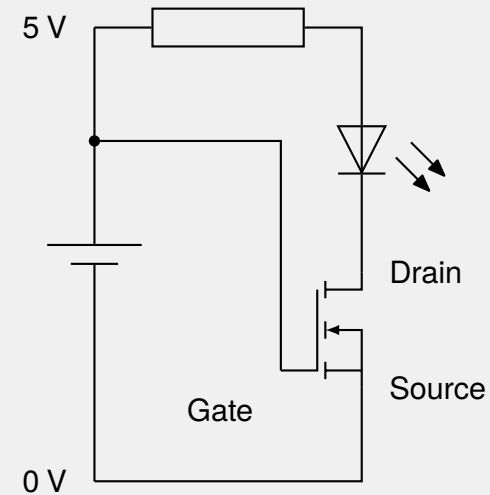


NMOS-FET: Funktionsweise

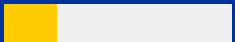
- Wenn zwischen dem *Gate* und *Source* eine genügend hohe Spannung vorhanden ist, so wird die Strecke zwischen *Source* und *Drain* leitend.



LED leuchtet nicht



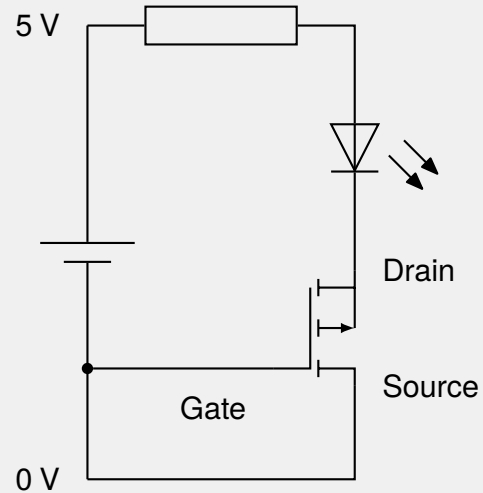
LED leuchtet



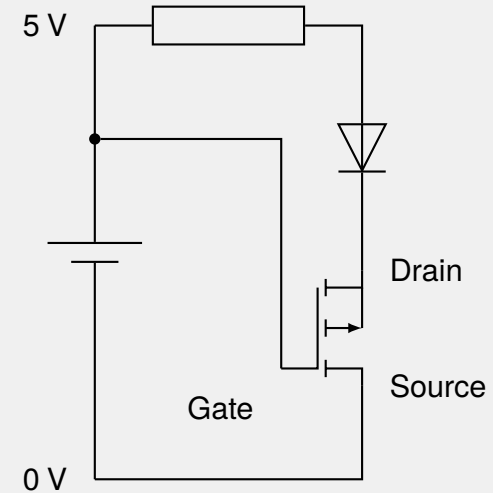


PMOS-FET: Funktionsweise

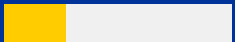
- Wenn zwischen dem *Gate* und *Drain* eine genügend hohe Spannung vorhanden ist, so wird die Strecke zwischen *Source* und *Drain* leitend.



LED leuchtet



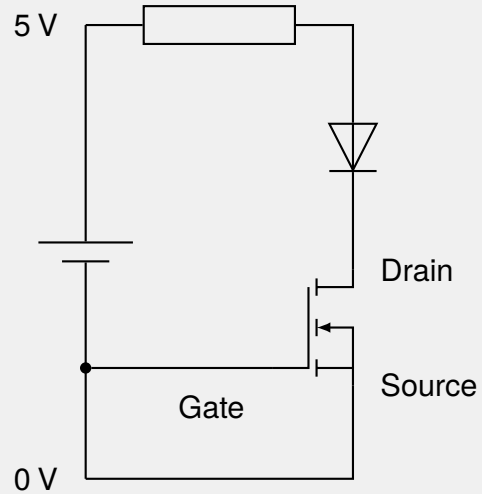
LED leuchtet nicht



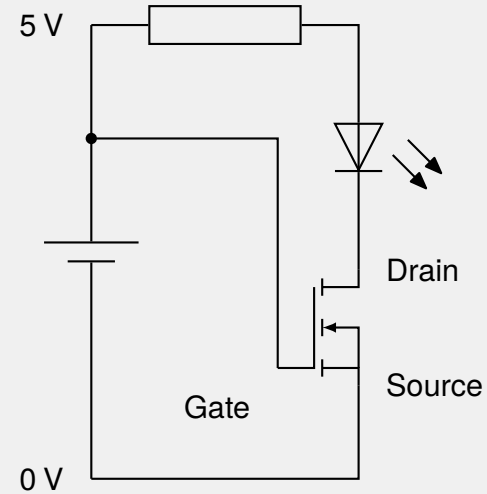


Aufgabe: NMOS-FET

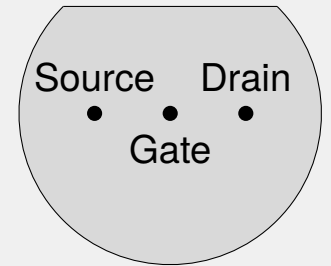
Testen Sie diese NMOS-FET-Schaltung:



LED leuchtet nicht



LED leuchtet

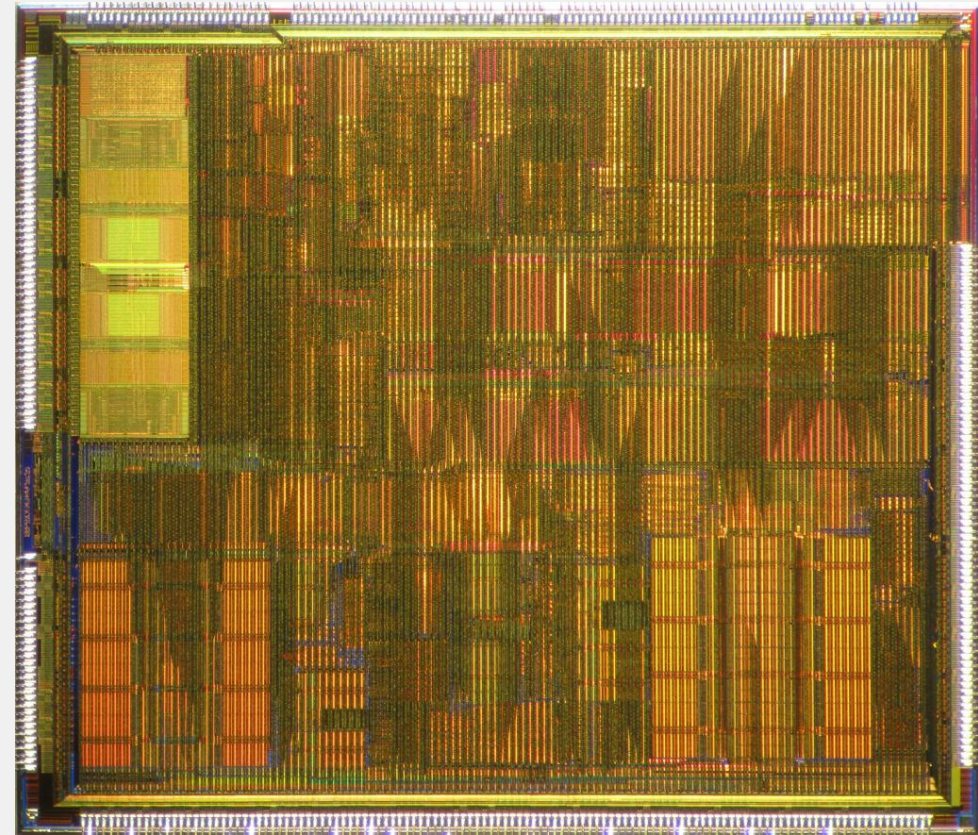


Pin-Belegung

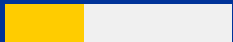


Integrierte Schaltkreise (IC)

- englisch: *Integrated Circuit*
- Prozessoren, Grafikprozessoren, Speicherchips, Mikrocontroller, ...
- Heute 1'400'000'000 Transistoren auf 177 mm^2 (Intel Haswell GT2 4C, 2014)
- Heute Polysilizium statt Metall (Aluminium)



Rechts: Foto eines Intel Pentium MMX Prozessors von 1995 mit 4'500'000 Transistoren auf 141 mm^2



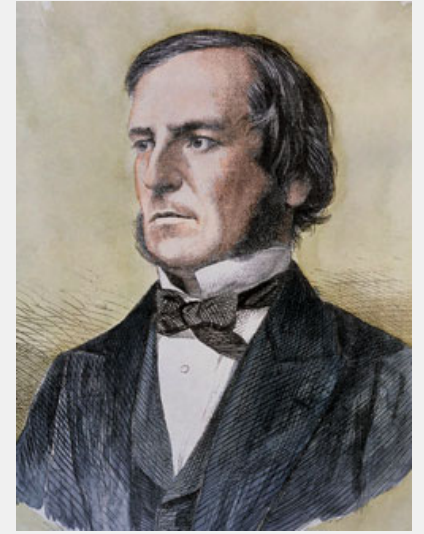


Boolesche Algebra

Boolesche Algebren sind eine mathematische Strukturen, die eine Verallgemeinerung der Mengenlehre darstellen. Die für uns wichtige Boolesche Algebra besitzt eine Grundmenge von zwei Elementen:

- 1: wahr, HIGH
- 0: falsch, LOW

Auf dieser Menge sind drei Grundoperationen definiert: UND, ODER, NICHT.



George Boole

Bezeichnung	Symbol
Konjunktion, UND	\wedge
Disjunktion, ODER	\vee
Negation, NICHT	\neg



Boolesche Algebra: Grundoperationen

A	B	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

Konjunktion
(UND)

A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

Disjunktion
(ODER)

A	$\neg A$
0	1
1	0

Negation
(NICHT)





NAND-Operation

Die *NAND*-Operation (von engl. *not and*) in einer Schaltalgebra ist definiert als eine Konjunktion mit anschließender Negation:

$$A \uparrow B = \neg(A \wedge B)$$

Durch die *NAND*-Operation lassen sich die drei Grundoperationen ausdrücken:

$$A \wedge B = (A \uparrow B) \uparrow (A \uparrow B)$$

$$A \vee B = (A \uparrow A) \uparrow (B \uparrow B)$$

$$\neg A = A \uparrow A$$

A	B	A \uparrow B
0	0	1
0	1	1
1	0	1
1	1	0

Somit lassen sich sämtliche Operationen in der Booleschen Algebra durch die *NAND*-Operation ausdrücken.



NAND-Gatter

- Physikalische Realisierung einer *NAND*-Operation
- Grundbaustein für alle digitalen Schaltungen



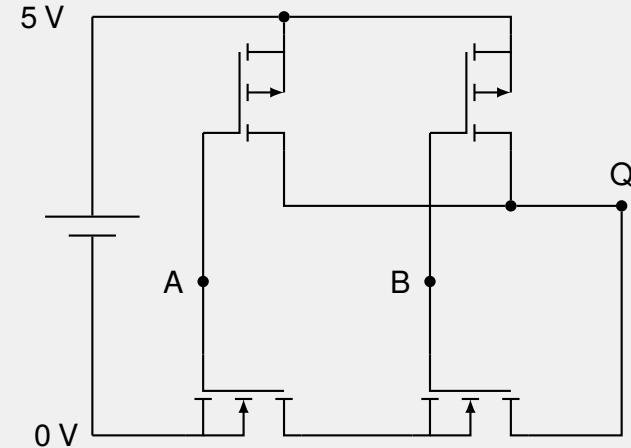


NAND-Gatter: Schaltung

Ein *NAND*-Gatter wird aus je zwei NMOS- und PMOS-Transistoren aufgebaut.

Die NMOS-Transistoren verbinden den Ausgang Q mit dem 0 V-Anschluss und sind in Serie geschaltet. Die PMOS-Transistoren sind parallel geschaltet und verbinden den Ausgang Q mit dem 5 V-Anschluss.

Die beiden Eingänge A und B steuern je ein Transistor von jedem Typ über dessen *Gate*.



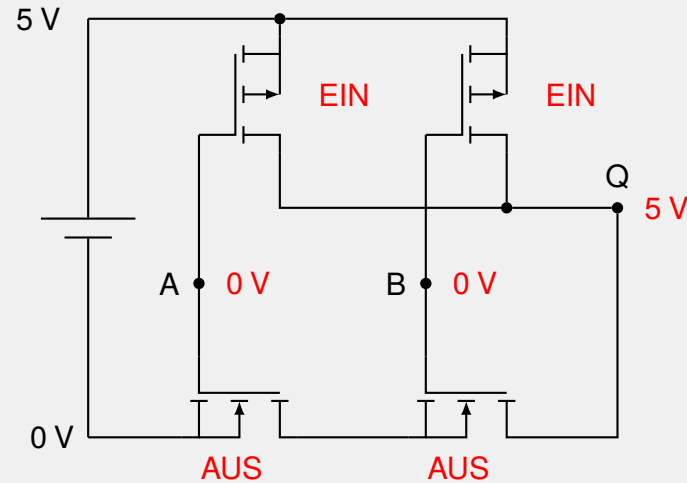


NAND-Gatter: Funktionsweise 1

Wenn an den Eingängen A und B 0 V anliegt, so sind die beiden NMOS-Transistoren ausgeschaltet und somit die Verbindung zwischen dem 0 V-Anschluss und Q unterbrochen.

Die PMOS-Transistoren sind eingeschaltet, sie stellen eine Verbindung zwischen dem 5 V-Anschluss und Q her.

Am Ausgang Q liegt somit eine Spannung von 5 V an.



A	B	Q
0	0	1

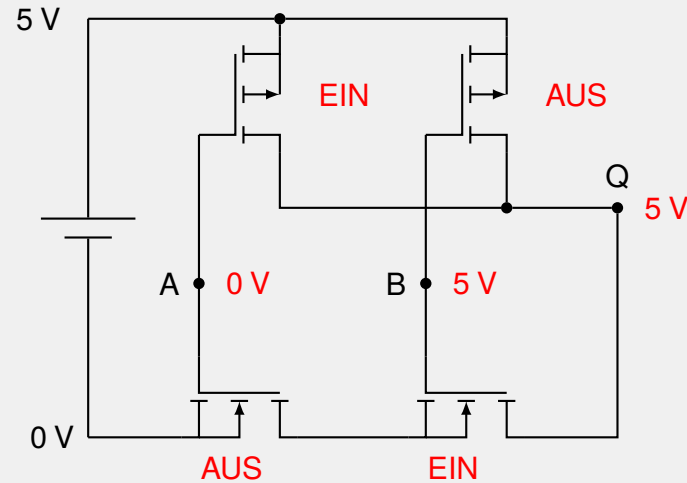


NAND-Gatter: Funktionsweise 2

Wenn am Eingang A 0 V und am Eingang B 5 V anliegen, so ist der rechte NMOS-Transistor ausgeschaltet und somit keine Verbindung zwischen dem 0 V-Anschluss und Q vorhanden.

Der linke PMOS-Transistor ist eingeschaltet und stellt eine Verbindung zwischen dem 5 V-Anschluss und Q her.

An Q liegt deshalb eine Spannung von 5 V an.



A	B	Q
0	0	1
0	1	1

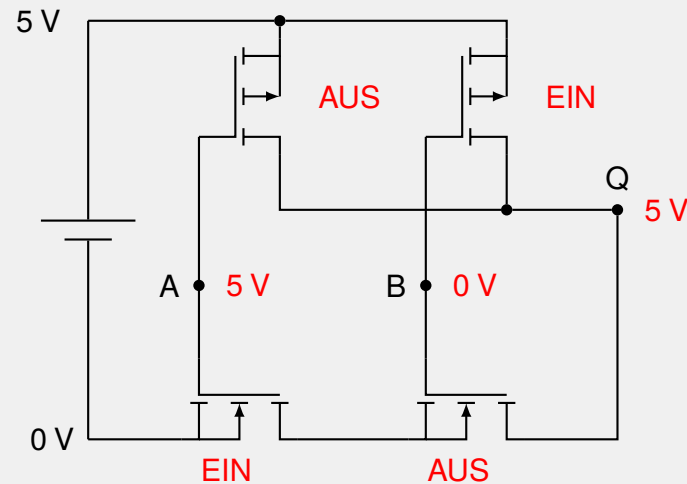


NAND-Gatter: Funktionsweise 3

Wenn am Eingang A 5 V und am Eingang B 0 V anliegen, so ist der linke NMOS-Transistor ausgeschaltet und somit keine Verbindung zwischen dem 0 V-Anschluss und Q vorhanden.

Der rechte PMOS-Transistor ist eingeschaltet und stellt eine Verbindung zwischen dem 5 V-Anschluss und Q her.

An Q liegt deshalb eine Spannung von 5 V an.



A	B	Q
0	0	1
0	1	1
1	0	1

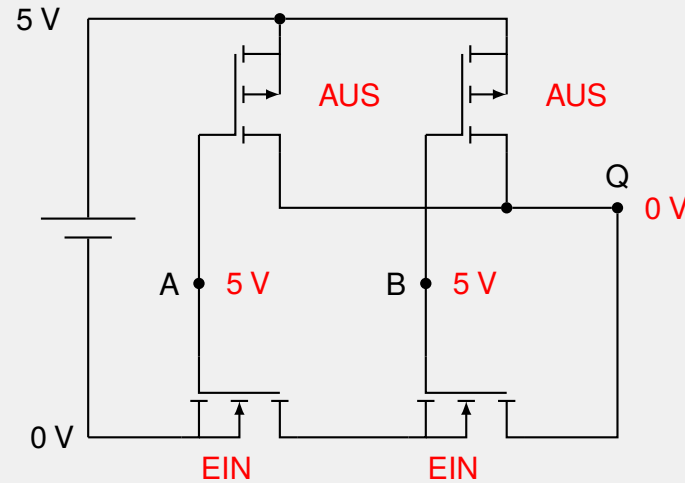


NAND-Gatter: Funktionsweise 4

Wenn an den Eingängen A und B 5 V anliegt, so sind die beiden NMOS-Transistoren eingeschaltet und stellen die Verbindung zwischen dem 0 V-Anschluss und Q her.

Die PMOS-Transistoren sind ausgeschaltet und somit die Verbindung zwischen dem 5 V-Anschluss und Q unterbrochen.

Am Ausgang Q liegt somit eine Spannung von 0 V an.

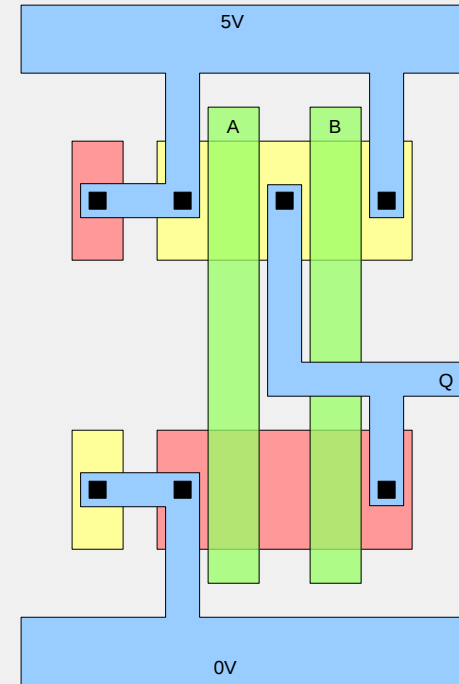
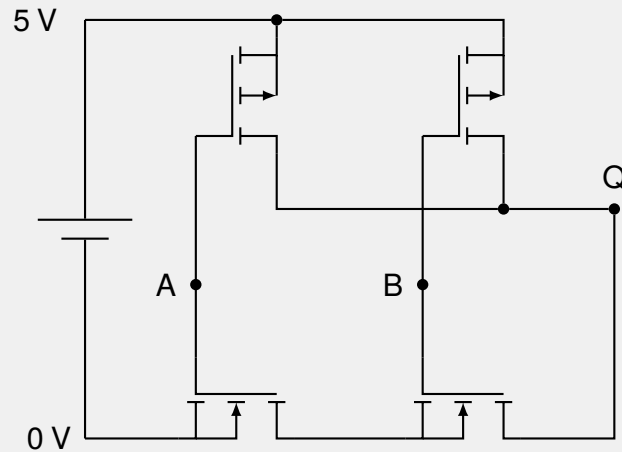


A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0



NAND-Gatter: Physikalischer Aufbau

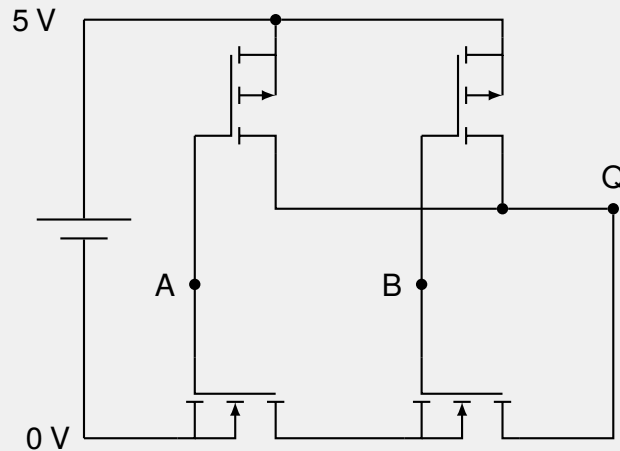
Ein *NAND*-Gatter kann sehr physikalisch effizient aufgebaut werden:





NAND-Gatter: Zusammenfassung

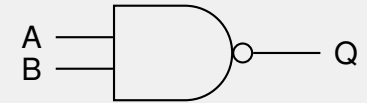
- Das *NAND*-Gatter hat zwei Eingänge und einen Ausgang.
- Am Ausgang liegt nur eine hohe Spannung vor, wenn an beiden Eingängen keine Spannung anliegt.



Schaltung

A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

Funktionstabelle



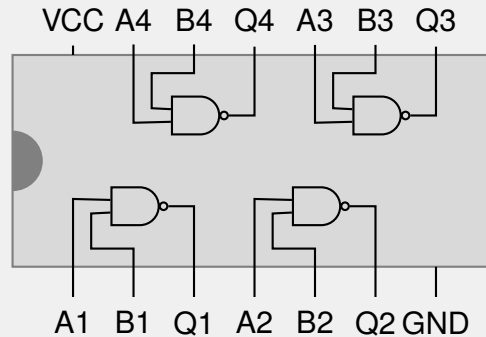
Symbol





74HC00 NAND-Chip

Der 74HC00-Chip enthält vier NAND-Logikgatter.



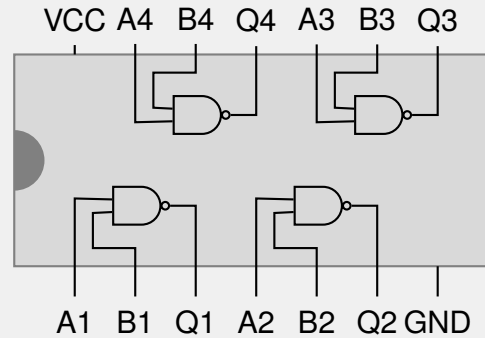
Symbol	Bedeutung
VCC	Versorgungsspannung 5 V
GND	Masse 0 V
A1 bis A4	erster Eingang des NAND-Gatters
B1 bis B4	zweiter Eingang des NAND-Gatters
Q1 bis Q4	Ausgang des NAND-Gatters

Bei der Darstellung von Logikschaltungen wird die Spannungsversorgung weggelassen.



Aufgabe 7: NAND-Chip

Steuern Sie eine Leuchtdiode über ein NAND-Gatter. Die Leuchtdiode wird am Ausgang des NAND-Gatters angeschlossen. Sie sollte nur leuchten, wenn beide Eingänge auf 0 V geschaltet sind.

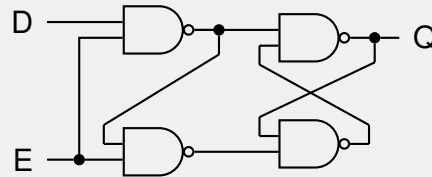


A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0



D-Flipflop

- Das Signal auf Eingang D (*data*) gibt den Wert des Bits an.
- Wird der Eingang E (*enable*) auf 1 gesetzt, so wird der Wert von Eingang D gespeichert.
- Der Wert kann bei Ausgang Q ausgelesen werden.
- Speichert ein Bit



Schaltung

D	E	Q
0	0	unverändert
0	1	0
1	0	unverändert
1	1	1

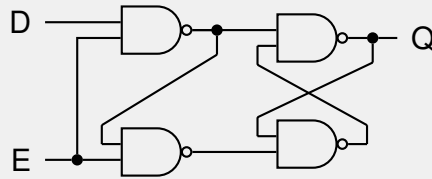
Funktionstabelle



Aufgabe 8: D-Flipflop

Bauen Sie eine D-Flipflop-Schaltung. Der Wert des Bits soll durch eine Leuchtdiode dargestellt werden.

Zusatzaufgabe a: Schliessen Sie das D-Flipflop am Arduino an. Schreiben Sie ein Programm, welches das kleinste Bit einer Zahl im D-Flipflop speichert.



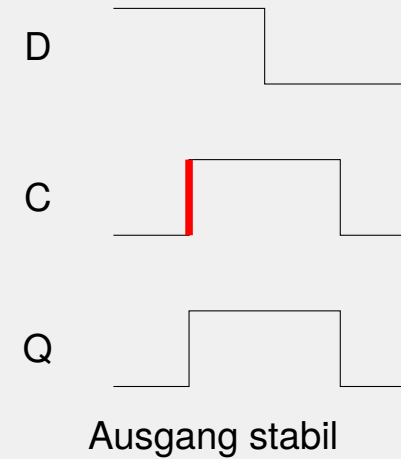
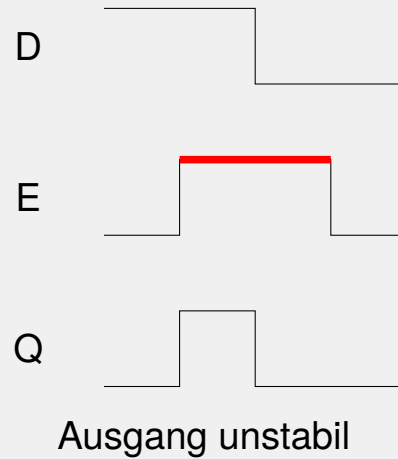
D	E	Q
0	0	unverändert
0	1	0
1	0	unverändert
1	1	1



Taktflankensteuerung

Problem: bei D-Flipflop: Ausgang unstabil, wenn Signal an E anliegt.

Lösung: Nur Speichern bei Übergang von 0 nach 1.

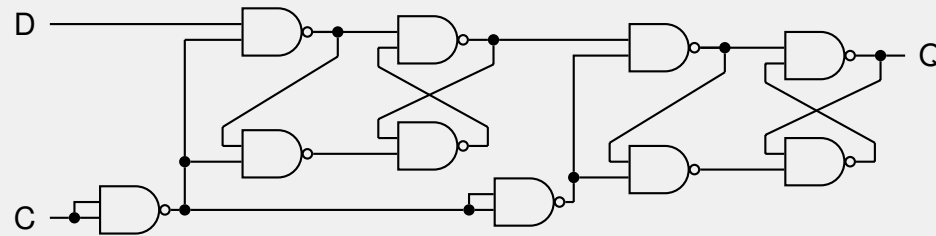
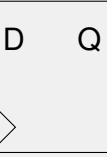




taktflankengesteuertes D-Flipflop

Grundbaustein für Datenspeicherung

- Ein Signal auf Eingang D (*data*) gibt den Wert des Bits an.
- Ein Übergang von 0 zu 1 auf Eingang C (*clock*) gibt an, dass der Wert von D gespeichert werden soll.
- Das Bit kann bei Ausgang Q ausgelesen werden.
- Speichert ein Bit



Schaltung

D	C	Q
0	↑	0
0	-	unverändert
1	↑	1
1	-	unverändert

Funktionstabelle