

Logikgatter verstehen

Workshop zur 15. Landestagung der Informatiklehrerinnen und -lehrer des Landes MV

Peter Böttcher 

22. April 2023

Alexander-von-Humboldt-Gymnasium Greifswald

1. Rahmenplan
2. Logikgatter
 - 2.1 Grundlagen
 - 2.2 Aufgaben
3. Simulation von Gattern
 - 3.1 DSImWeb und Logigator
 - 3.2 Vertiefende Aufgaben
 - 3.3 Arbeiten mit Modulen
4. Zusammenfassung

Rahmenplan

Klasse 10

ca. 20 Unterrichtsstunden

Digitalisierung in meiner Umgebung untersuchen

ca. 20 Unterrichtsstunden

[MD] [BO] [BNE] [DRF] [PG]

Grundlagen der Digitalisierung [MD] [BO] [BNE] [DRF] [PG]

ca. 8 Unterrichtsstunden

Eignung binärer Signale für die maschinelle Verarbeitung erläutern

- Bits logisch verknüpfen
- binäre Addition

Die Prinzipien sind anschaulich mit einem Verweis für Möglichkeiten einer technischen Umsetzung zu vermitteln.

...

Sekundarstufe 2 - Leistungskurs

Rechnerarchitektur

ca. 6/10 Unterrichtsstunden

von-Neumann-Rechner [MD5] [Geschichte]

ca. 6/10 Unterrichtsstunden

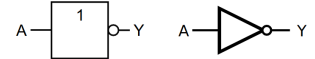
Verbindliche Ziele und Inhalte	Hinweise und Anregungen
<p>von-Neumann-Rechner als Struktur- und Arbeitsmodell</p> <ul style="list-style-type: none">• maschinennahe Programme in einer Simulation analysieren und das Systemverhalten erklären	
<i>zusätzlich für den Leistungskurs</i>	
<ul style="list-style-type: none">• maschinennahe Programme in einer Simulation erweitern und entwickeln• Gemeinsamkeiten und Unterschiede zur Harvard-Architektur beschreiben• elementare, binär arbeitende Hardware-Komponenten auf Basis von Logik-Gattern mithilfe einer Simulation in Aufbau und Funktionsweise beschreiben<ul style="list-style-type: none">- Halbaddierer- Flip-Flop	<p>Der innere Aufbau eines Logik-Gatters ist nicht zu thematisieren.</p>

	Nicht Bestandteil der schriftlichen Abiturprüfung	Sonstige Hinweise
Grundkurs	<ul style="list-style-type: none">- Meilensteine der Informatik sowie Kooperatives und kollaboratives Arbeiten- Relationenalgebra- Turingmaschine- Netzzugangsschicht- Lineare Datenstruktur Liste- Analyse und Erklärung maschinennaher Programme	Der Bereich „Sichere Kommunikation“ wird auf die Analyse sowie die objektorientierte Modellierung und Programmierung klassischer symmetrischer Verschlüsselungsverfahren mithilfe von Stringmanipulationen reduziert.
Leistungskurs	<ul style="list-style-type: none">- Meilensteine der Informatik sowie Kooperatives und kollaboratives Arbeiten- Relationenalgebra- Netzzugangsschicht- Kellerautomat- Hardwarekomponenten auf Basis von Logikgatter- Thema „Formale Sprachen und Grammatiken“	Die Sortieralgorithmen „Sortieren durch Austausch“ und „Sortieren durch Mischen“ sind im Kontext des objektorientierten Modellierens und Programmierens zu thematisieren. Der Bereich „Sichere Kommunikation“ wird auf das Prinzip der asymmetrischen Verschlüsselung mithilfe der Einwegfunktion reduziert.

	Nicht Bestandteil der schriftlichen Abiturprüfung	Sonstige Hinweise
Grundkurs	<ul style="list-style-type: none">- Meilensteine der Informatik sowie Kooperatives und kollaboratives Arbeiten- Relationenalgebra- Turingmaschine- Thema „Sichere Kommunikation“- Analyse und Erklärung maschinennaher Programme	In der objektorientierten Modellierung und Programmierung sollten Stringmanipulationen und Listenoperationen thematisiert werden.
Leistungskurs	<ul style="list-style-type: none">- Meilensteine der Informatik sowie Kooperatives und kollaboratives Arbeiten- <u>Relationenalgebra</u>- Kellerautomat- Thema „Formale Sprachen und Grammatiken“	In der objektorientierten Modellierung und Programmierung sollten Stringmanipulationen und Listenoperationen sowie Polymorphie und Vererbung thematisiert werden.

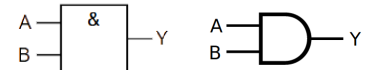
Logikgatter

Siehe auch: Tafelwerksergänzung
Auch im Abitur verfügbar!

Name	Bezeichnung	Symbole
NOT	Nicht	

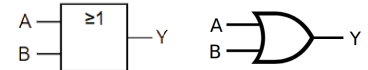
Wertetabelle

A	Y
0	1
1	0

Name	Bezeichnung	Symbole
AND	Und	

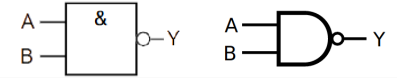
Wertetabelle

A	B	Y
0	0	0
1	0	0
0	1	0
1	1	1

Name	Bezeichnung	Symbole
OR	Oder	 The image shows two symbols for an OR gate. On the left is a rectangular symbol with two input lines labeled 'A' and 'B' on the left side, and one output line labeled 'Y' on the right side. Inside the rectangle is the symbol '≥1'. On the right is a D-shaped symbol with two input lines labeled 'A' and 'B' on the left side, and one output line labeled 'Y' on the right side.

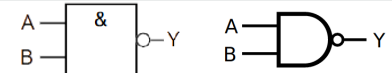
Wertetabelle

A	B	Y
0	0	0
1	0	1
0	1	1
1	1	1

Name	Bezeichnung	Symbole
NAND	Nicht-Und	

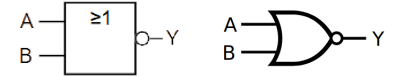
Wertetabelle

A	B	Y
0	0	
1	0	
0	1	
1	1	

Name	Bezeichnung	Symbole
NAND	Nicht-Und	

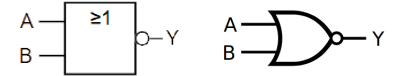
Wertetabelle

A	B	Y
0	0	1
1	0	1
0	1	1
1	1	0

Name	Bezeichnung	Symbole
NOR	Nicht-Oder	

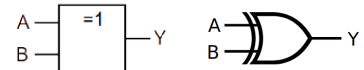
Wertetabelle

A	B	Y
0	0	
1	0	
0	1	
1	1	

Name	Bezeichnung	Symbole
NOR	Nicht-Oder	

Wertetabelle

A	B	Y
0	0	1
1	0	0
0	1	0
1	1	0

Name	Bezeichnung	Symbole
XOR	Exklusiv-Oder	

Wertetabelle

A	B	Y
0	0	0
1	0	1
0	1	1
1	1	0

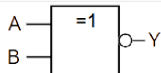
Name

Bezeichnung

Symbole

XNOR

Exklusiv-Nicht-Oder



Wertetabelle

A	B	Y
0	0	1
1	0	0
0	1	0
1	1	1

Aufgabe

Beschreibe die Funktionsweise des Gatters in eigenen Worten. Nutze dazu Sätze wie: „Die Lampe von Gatter 1 ist nur dann aus, wenn...“

Digitalo ist ein Kartenspiel, bei welchem zwei Spieler*innen versuchen, aus Gatter-Karten eine pyramidenförmige logische Schaltung aufzubauen.
informatik.mygymer.ch/m23c/010.circuits/09.digitalo.html

Gartenbewässerung (1)

Eine Bewässerungsanlage soll nur wässern, wenn der Wettersensor gutes Wetter meldet (gekennzeichnet mit „s“) und der Bewegungssensor niemanden im Garten erkennt (gekennzeichnet mit „g“). Die beiden Signale werden jeweils mit 0 (Schlechtes Wetter/Garten besetzt) oder 1 (Gutes Wetter/Garten frei) kodiert.

Aufgabe 1.1

Interpretiere folgende Zustände:

- $s=0, g=1$
- $s=0, g=0$
- $s=1, g=1$

Gartenbewässerung (2)

Aufgabe 1.2

Ergänze die Wertetabelle so, dass in der letzten Spalte das in der Beschreibung gewünschte Verhalten der Bewässerungsanlage abgebildet wird ($W=1$ - Bewässerung, $W=0$ - Keine Bewässerung).

s	g	W
0	0	
1	0	
0	1	
1	1	

Aufgabe 1.3

Welches Logikgatter würdest du wählen, um das gewünschte Verhalten abzubilden? Begründe deine Auswahl.

Eigentlich müssten spätestens jetzt folgende
Begriffe geklärt werden:

Schaltung

Eingang

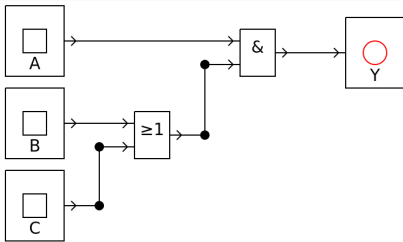
Ausgang

Verzweigungen im Schaltplan

Verknüpfung von Gattern

Aufgabe 2.1

Betrachte die folgende Schaltung und fülle die dazugehörige Wahrheitstabelle aus. Bei welcher Belegung entsteht welches Ergebnis?

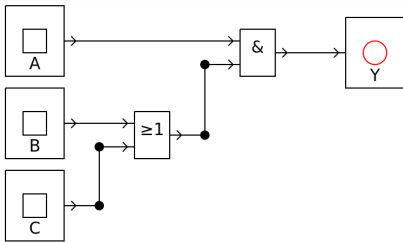


A	B	C	Y
1	1	1	
1	1	0	
1	0	1	
1	0	0	
0	1	1	
0	1	0	
0	0	1	
0	0	0	

Verknüpfung von Gattern

Aufgabe 2.1

Betrachte die folgende Schaltung und fülle die dazugehörige Wahrheitstabelle aus. Bei welcher Belegung entsteht welches Ergebnis?



A	B	C	Y
1	1	1	1
1	1	0	1
1	0	1	1
1	0	0	0
0	1	1	0
0	1	0	0
0	0	1	0
0	0	0	0

Aufgabe 2.2

Überlege dir einen Anwendungsfall aus der realen Welt, der mit einer solchen Schaltung abgedeckt werden könnte.

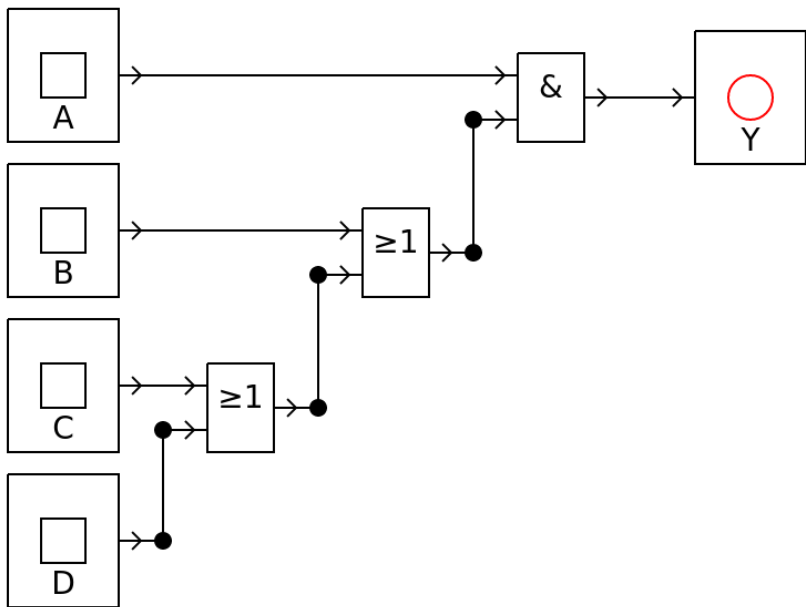
Verknüpfung von Gattern (3)

Angenommen, diese Schaltung würde eine Alarmanlage mit einem Hauptschalter (A) und zwei Sensoren (B und C) abbilden. Wenn der Hauptschalter gedrückt ist, und einer der Sensoren aktiviert wird, schlägt die Anlage Alarm.

Aufgabe 2.3

Füge einen dritten Sensor D hinzu, der gleichberechtigt zu den Sensoren B und C anschlagen kann.

Verknüpfung von Gattern (3)



Simulation von Gattern

Warum simulieren?

- Stärkt Verständnis
- Lebensweltbezug
- Änderungen werden direkt sichtbar
- (Wird im Abitur benötigt)

- Auf IO-Stick verfügbar
- Wird aktiv entwickelt
- Soll LogicSim ersetzen
- <https://logigator.com/> als Web-App
- <https://logigator.com/download/> als Download für Linux/Windows
- **Achtung:** Nicht kompatibel mit LogicSim!

Logicator-Interface

The screenshot displays the Logicator software interface for designing a Half Adder circuit. The window title is "Half Adder" and the user is identified as "Leo-P". The interface includes a menu bar (File, Edit, View, Help), a toolbar with various design tools, and a "Start Simulation" button.

The left sidebar contains a component library with the following categories:

- Basic:** NOT Gate (!), AND Gate (&), OR Gate (≥ 1), XOR Gate (=1), Delay (1), and Clock (clk).
- Advanced:** Half Adder (HA), Full Adder (FA), ROM, D Flip Flop (D), JK Flip Flop (JK), and SR Flip Flop (SR).
- IO:** (Partially visible)

Two circuit diagrams are shown in the main workspace:

- Top Diagram:** A logic-level implementation of a Half Adder. It features two input buses labeled "A" and "B". The output of the "A" bus is connected to the top input of an XOR Gate (=1) and the top input of an AND Gate (&). The output of the "B" bus is connected to the bottom input of both the XOR Gate and the AND Gate. The output of the XOR Gate is labeled "Sum", and the output of the AND Gate is labeled "Carry".
- Bottom Diagram:** A block-level implementation of a Half Adder. It features two input buses labeled "A" and "B" connected to the "A" and "B" inputs of a block labeled "HA" (Half Adder). The "S" output of the "HA" block is labeled "Sum", and the "C" output is labeled "Carry".

The status bar at the bottom indicates "Selecting Elements" and "Saved Sel: 0".

- Noch nicht auf IO-Stick
- Wird aktiv entwickelt
- Sehr intuitive Bedienung
- Web-App auch offline verfügbar

Projektname: project *

- Gatter
- AND
- OR
- NOT
- NAND
- NOR
- XOR
- Ein- und Ausgabe
- Schalter
- LED
- 0 konstant
- 1 konstant
- 7-Segmentanzeige
- Binäreingabe
- Binärdisplay
- Schaltnetze
- Halbaddierer
- Volladdierer
- 7-Segmentdecoder
- Multiplexer
- Demultiplexer
- Flipflops
- RS async

+

```
graph LR; I1[ ] --> AND[&]; I2[ ] --> AND; AND --> O1(( ))
```

Module

neues Modul

neues Modul Modul löschen

gewähltes Modul:

neues Modul

Abkürzung:

M1

Eingänge

Ausgänge

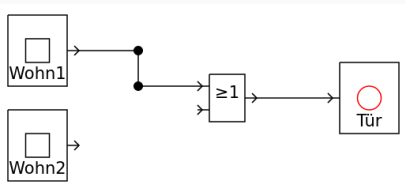
Zoom: 100%

X: 267 Y:628

Der Türöffner in einem Mehrfamilienhaus muss reagieren, sobald der entsprechende Knopf in einer der Wohnungen gedrückt wird.¹

Aufgabe 3.1

Überprüfe die gegebene Schaltung in DSImWeb für ein Zweifamilienhaus auf Korrektheit und korrigiere ggf. Fehler, indem du Signalleitungen von einem Ausgang an einen Eingang legst.



¹Nach Inf-Schule.de - Grundgatter sowie T. Hempel, CC BY-SA

Aufgabe 3.2

Vervollständige folgenden Satz mit den Begriffen „und“, „oder“, „entweder-oder“ bzw. „nicht“:

Wenn der Türknopf in Wohnung 1 _____ der Türknopf in Wohnung 2 gedrückt wird, dann _____.

Aufgabe 3.3

Erweitere die Schaltung so, dass sie auch in einen Haus mit 4 Wohnungen genutzt werden kann.

¹Nach Inf-Schule.de - Grundgatter sowie T. Hempel, CC BY-SA

Zum Start einer Rakete benötigt ein Raketenkontrollsystem eine Starterlaubnis. Wenn die Starterlaubnis erteilt ist, kann einer von zwei Startknöpfen gedrückt werden, um die Rakete zu starten.

Aufgabe 4.1

Erstelle in DSIMWeb eine Schaltung mit zwei Logikgattern, die diesen Sachverhalt abbildet. Das Abheben der Rakete soll durch das Leuchten einer Lampe symbolisiert werden.

Aufgabe 4.2

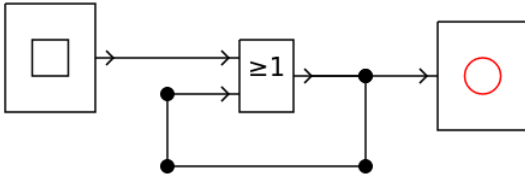
Füge der Schaltung einen „Präsidentenknopf“ hinzu. Dieser kann sämtliche Regeln überschreiben und die Rakete auch ohne Starterlaubnis starten lassen.

Aufgabe 4.3

Zur Wartung der Rakete ist ein Testmodus nötig. Wird der Schalter für den Testmodus betätigt werden sämtliche Eingaben ignoriert (auch der Präsidentenknopf) und die Rakete kann unter keinen Umständen starten.

Aufgabe 5.1

Baue folgende Schaltung in DSimWeb nach. Was passiert, wenn du den Schalter betätigst?



Aufgabe 5.2

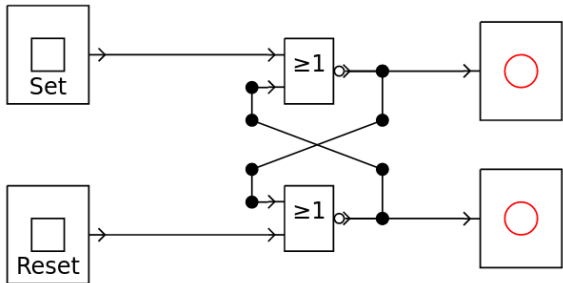
Wo könnte eine solche Schaltung nützlich sein? Welche Möglichkeiten siehst du zur Erweiterung?

Baue eine Alarmanlage, die manuell zurückgesetzt werden muss, damit der Alarm aufhört.

Speichern von Zuständen - RS-FlipFlop

Aufgabe 5.3

Diese Schaltung nennt sich „R(ese)t-S(et)-FlipFlop“. Baue sie in DSimWeb nach. Wie verhalten sich die beiden Ausgänge zueinander?



Man könnte an dieser Stelle thematisieren, warum ein FlipFlop besser ist als die Schaltung auf der vergangenen Folie (NAND-Gatter lassen sich leichter produzieren), aber das ist im Rahmenplan explizit nicht vorgesehen. Außerdem: Prima Einstieg in Automatentheorie!

DSimWeb bietet (wie fast alle Logiksimulatoren) die Verwendung von Modulen an.

Die Realisierung von Addition erfolgt im Rechenwerk des von-Neumann-Rechners mittels Logikgattern. Aber wie?²

Aufgabe 6.1

Für das Addieren zweier Bits gilt folgende Wertetabelle:

A		B	Y	Ü
0	+	0	0	0
1	+	0	1	0
0	+	1	1	0
1	+	1	0	1

Wofür steht das Ü? Kann es ein einzelnes Logikgatter geben, das uns diese Wertetabelle erzeugt? Gibt es ein Gatter, das unter Berücksichtigung der Eingänge A und B den Ausgang Y erzeugt?

²Nach T. Hempel 2023, CC BY-SA 4.0

Aufgabe 6.2

Baue eine Schaltung, die die folgende Wertetabelle implementiert.

A		B	Y	Ü
0	+	0	0	0
1	+	0	1	0
0	+	1	1	0
1	+	1	0	1

Aufgabe 6.2 (alternative)

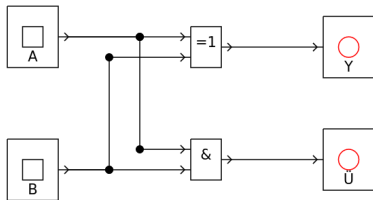
Baue eine Schaltung, die die folgende Wertetabelle implementiert, ohne ein XOR-Gatter zu verwenden.

²Nach T. Hempel 2023, CC BY-SA 4.0

Halbaddierer (2)

Aufgabe 6.2

Baue eine Schaltung, die die folgende Wertetabelle implementiert.



Aufgabe 6.2 (alternative)

Baue eine Schaltung, die die folgende Wertetabelle implementiert, ohne ein XOR-Gatter zu verwenden.

²Nach T. Hempel 2023, CC BY-SA 4.0

Ein Volladdierer arbeitet wie ein Halbaddierer, berücksichtigt aber zusätzlich noch einen ggf. vorhandenen Übertrag C.

Aufgabe 7.1

Ergänze die Wertetabelle des Halbaddierers um einen weiteren Input „C“ und passe die Ausgaben entsprechend an.

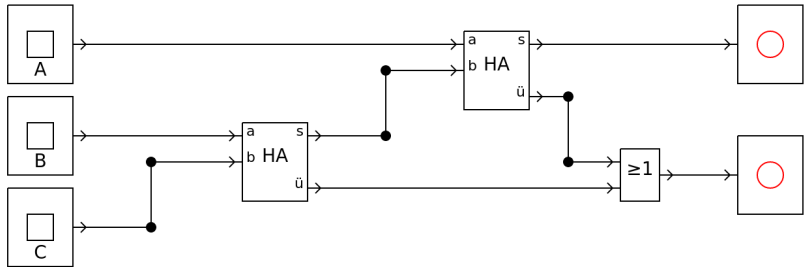
Tipp: Deine Tabelle muss 5 Zeilen und 8 Spalten besitzen.

²Nach T. Hempel 2023, CC BY-SA 4.0

Volladdierer (2)

Aufgabe 7.2

Überprüfe die Korrektheit der Schaltung für einen Volladdierer anhand deiner Wertetabelle.



²Nach T. Hempel 2023, CC BY-SA 4.0

Zusammenfassung

Fragen?

Multiplexer (1)

Ein Multiplexer (kurz: MUX oder Mux) ist eine Selektionsschaltung, mit der aus einer Anzahl von Eingangssignalen eines ausgewählt und an den Ausgang durchgeschaltet werden kann.

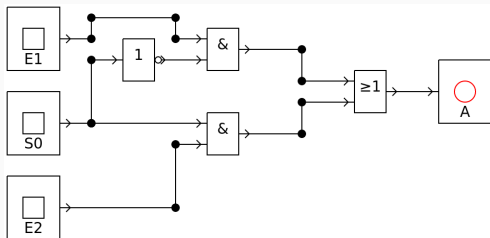
Aufgabe 8.1

Beschreibe einen möglichen Anwendungsbereich für einen Multiplexer. Informiere dich mittels einer Internetrecherche über weitere Anwendungsbereiche.

Multiplexer (2)

Aufgabe 8.2

Unten siehst du das Schaltbild eines Multiplexers. Erstelle eine Wahrheitstafel für die 3 Eingänge und den Ausgang.



Aufgabe 8.3

Erweitere den Multiplexer um 4 weitere Eingänge ($E_3 - E_6$). Wie viele Schalteinträge S werden dafür benötigt?

-  <https://www.inf-schule.de/rechner/digitaltechnik>
-  https://www.inf-schule.de/rechner/digitaltechnik/Simulatoren/DSimWeb/DSimWeb_Vollversion
-  <https://www.inf-schule.de/rechner/digitaltechnik/gatter/digitalo>
-  <https://logigator.com/>
-  <https://schule.informatik.uni-rostock.de/mod/folder/view.php?id=3848>
-  <https://www.hackster.io/ToniTaste/logik-gatter-und-schaltungen-simulieren-ea2e17>