

Mit Johnny die von-Neumann-Prinzipien verstehen

ILTIS 20.03.2019

TAJA ZUCHTMANN



Gliederung

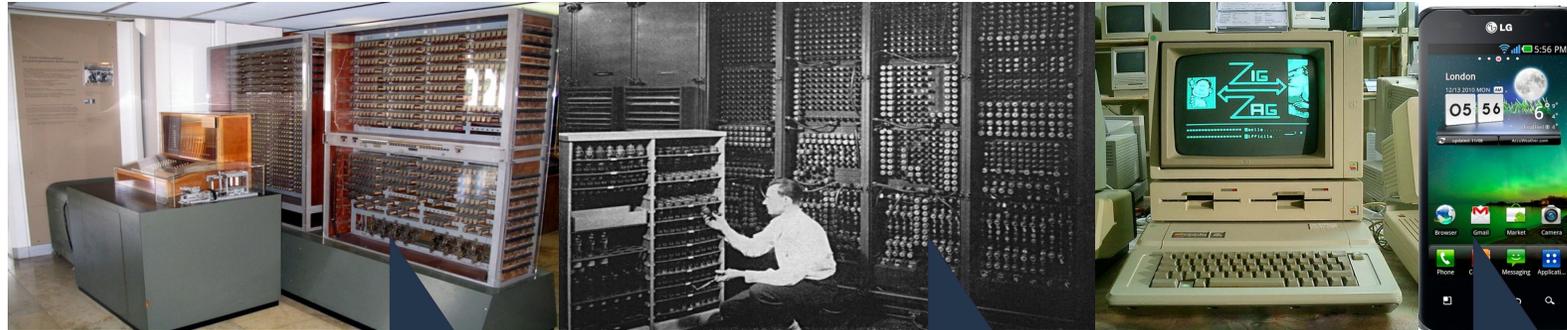
- 1) Motivierung
- 2) Der Von-Neumann-Rechner
- 3) Logikgatter
- 4) Inspirationen



1. Motivierung



1. Motivierung



geringere
Störanfälligkeit

geringerer
Energiebedarf

geringerer
Platzbedarf



Worin liegt der Unterschied?



Das Innere eines Rechners

Quelle: <https://wiki.zum.de/wiki/Datei:RIMG0014a.JPG>



Worin liegt der Unterschied?



Der Unterschied liegt
in der Funktions-
weise.

Es kommt darauf an,
was im Inneren
passiert!

Mit bloßem Auge ist
dies nicht zu sehen.

Das Innere eines Rechners

Quelle: <https://wiki.zum.de/wiki/Datei:RIMG0014a.JPG>



Wie funktioniert das?



Abiturvorbereitung

Themenfelder	
Rechner und Netze	<ul style="list-style-type: none">- Filius- Netemul- Johnny

Quelle: https://www.bildung-mv.de/export/sites/bildungsserver/downloads/Vorbereitung_ABI_AB_B_2018.pdf, S. 40

Kompetenzen und Inhalte

„Grundlage für die Anforderungen in den Prüfungsaufgaben sind die Einheitlichen Prüfungsanforderungen der **KMK für das Fach Informatik**, das **Kerncurriculum Informatik** für die Qualifikationsphase der gymnasialen Oberstufe sowie **der Rahmenplan Informatik** für die Jahrgangsstufen 7-10 des gymnasialen Bildungsgangs in M-V.“



Abiturvorbereitung

Themenfelder	
Rechner und Netze	- Filius - Netemul - Johnny

Quelle:
https://www.bildung-mv.de/export/sites/bildungsserver/downloads/Vorbereitung_ABI_AB_2018.pdf, S. 40

Kompetenzen und Inhalte zu den grundsätzlichen Funktionsweisen von Computersystemen

- EVAS-Prinzip
- John-Von-Neumann-Aufbau
- Von-Neumann-Befehlszyklus



Rahmenplan für Informatik und Medienbildung

Thema: Grundlagen der Digitalisierung

Klasse 10

Mit dem von-Neumann-Modell lernen die Schülerinnen und Schüler den Aufbau und die Funktionsweise von Informatiksystemen kennen und verstehen die Programmierbarkeit als zentrales Wirkprinzip von Informatiksystemen.

Verbindliche Ziele und Inhalte	Hinweise und Anregungen
den Aufbau und die grundlegende Funktionsweise eines Informatiksystems nach dem von-Neumann-Modell beschreiben	Die Schülerinnen und Schüler nutzen ein Werkzeug zur Simulation maschinennaher Programmierung mit Hilfe eines vereinfachten von-Neumann-Modells. Sie erkennen die Zweckmäßigkeit der Verwendung höherer Programmiersprachen im Vergleich zur Maschinensprache.



2. Der Von-Neumann-Rechner



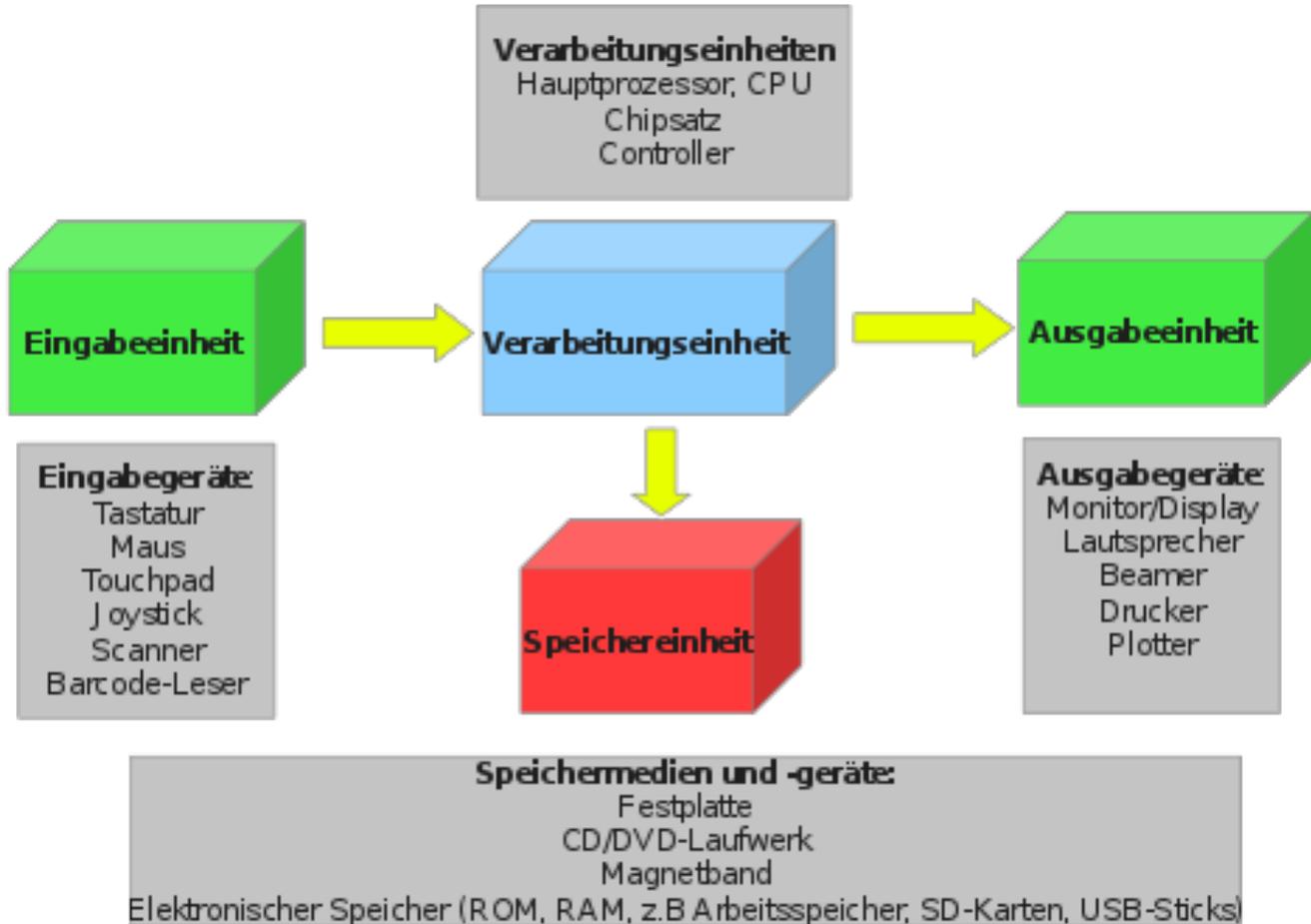
Kahoot.it

2.1 Reaktivierung

Das EVAS-Prinzip:

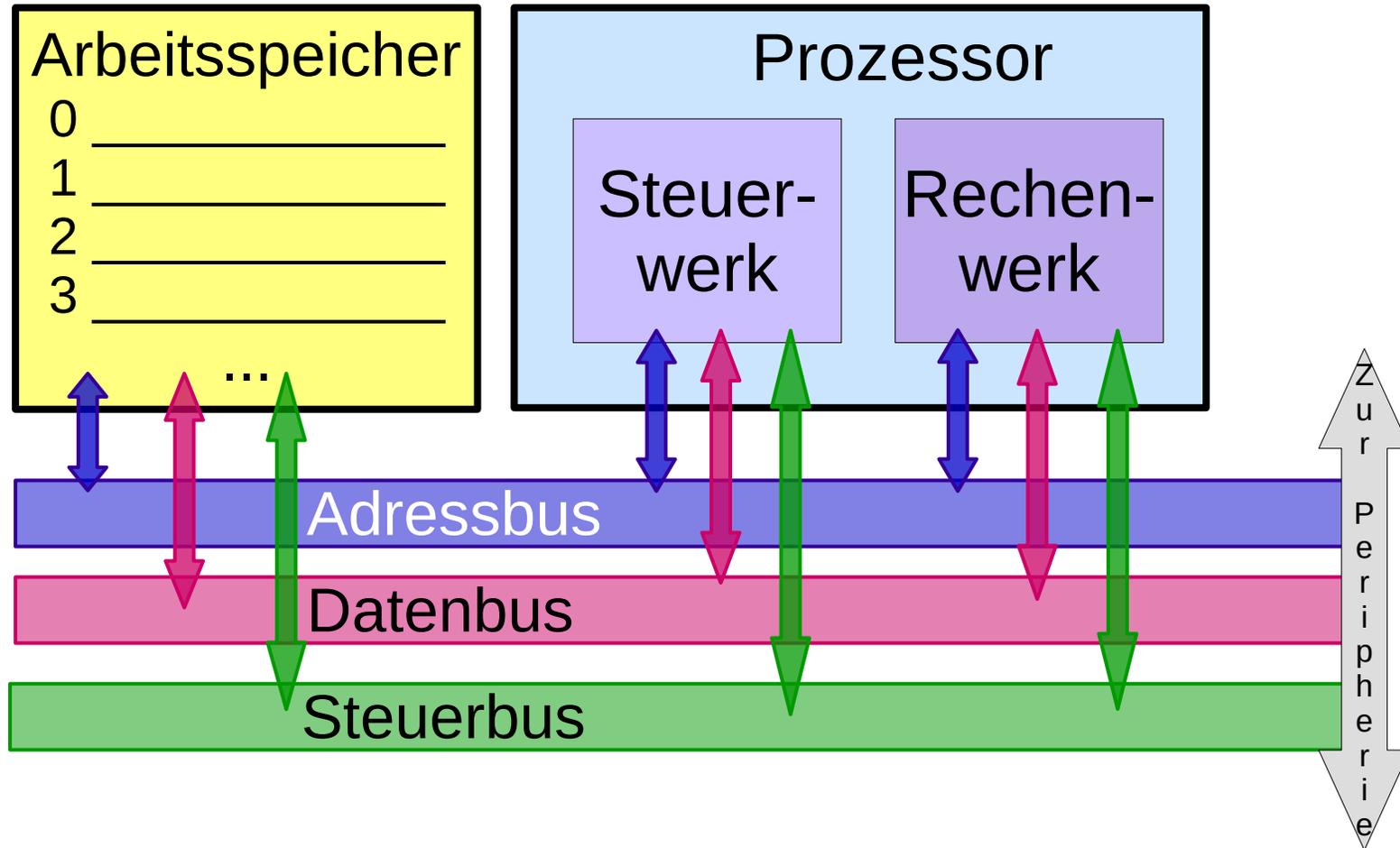
Eingabe, Verarbeitung,
Ausgabe, Speicherung

Quelle: <https://de.wikipedia.org/wiki/Datei:EVA-Prinzip.svg>

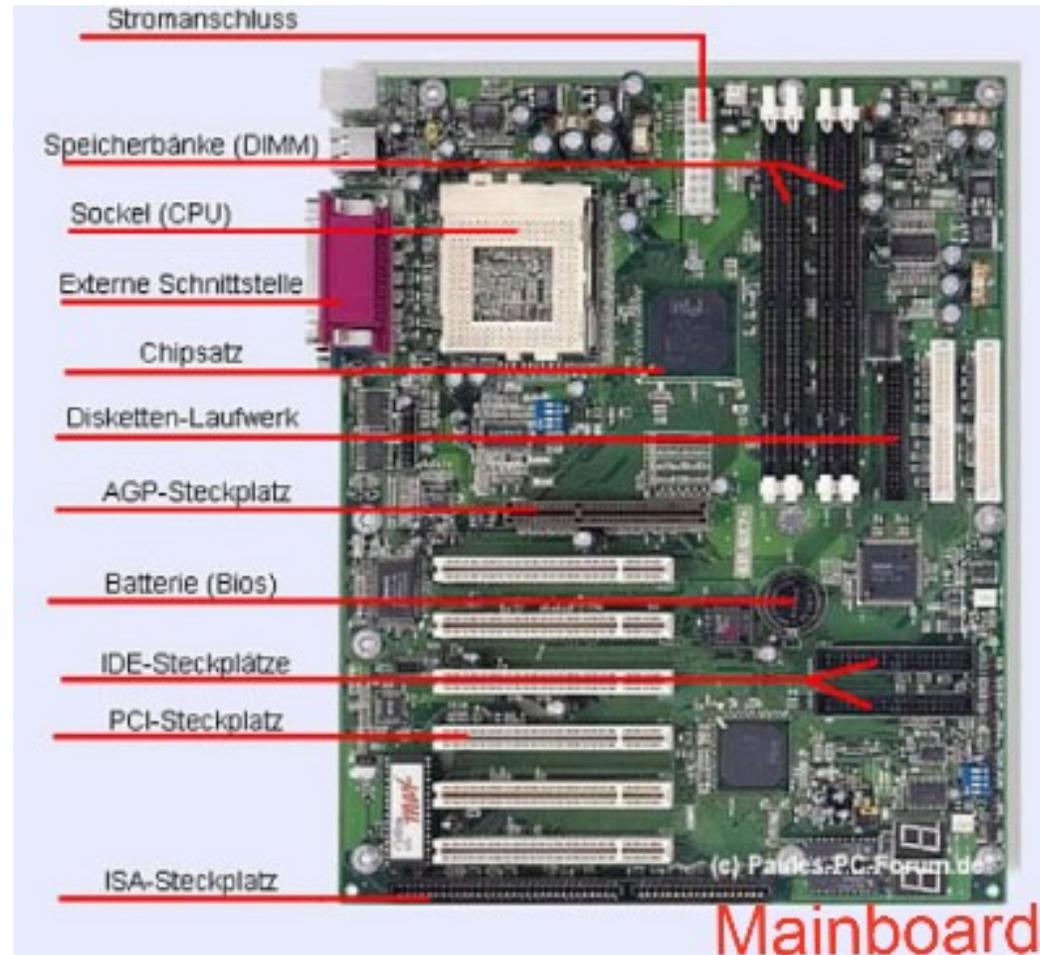


2.3 Der Aufbau eines Computersystems

Die Verarbeitungseinheit



2.3 Der Aufbau eines Computersystems

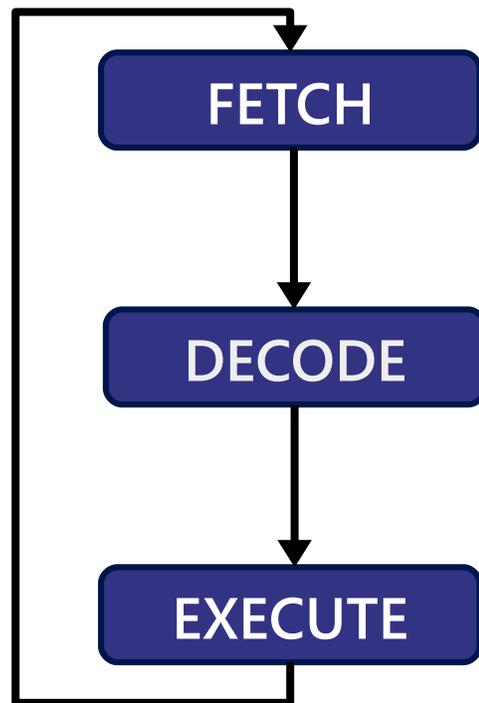


Quelle: <https://f2s12d1.beepworld.de/software.htm>



2.4 Der von-Neumann Befehlszyklus

Der Ablauf des Zyklus kann in mindestens drei Phasen eingeteilt werden:



Befehlsholphase:

Laden eines Befehls aus dem Speicher in das Steuerwerk

Dekodierphase:

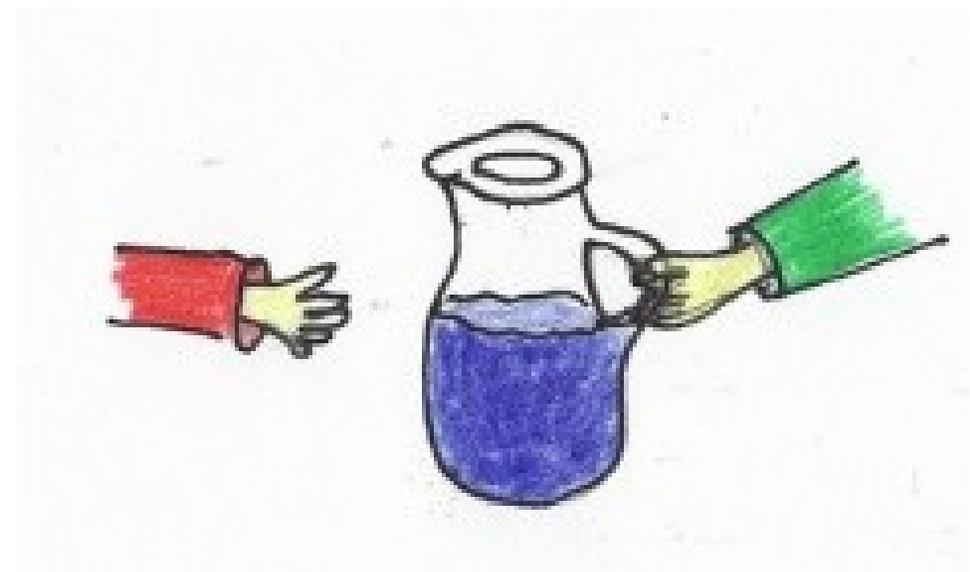
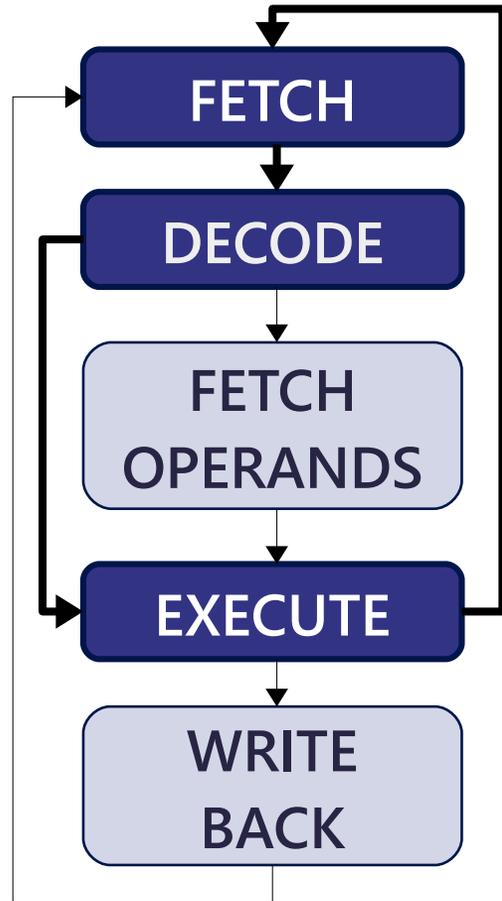
Interpretation des Befehls im Steuerwerk

Ausführungsphase:

Ausführung des Befehls



2.4 Der von-Neumann Befehlszyklus

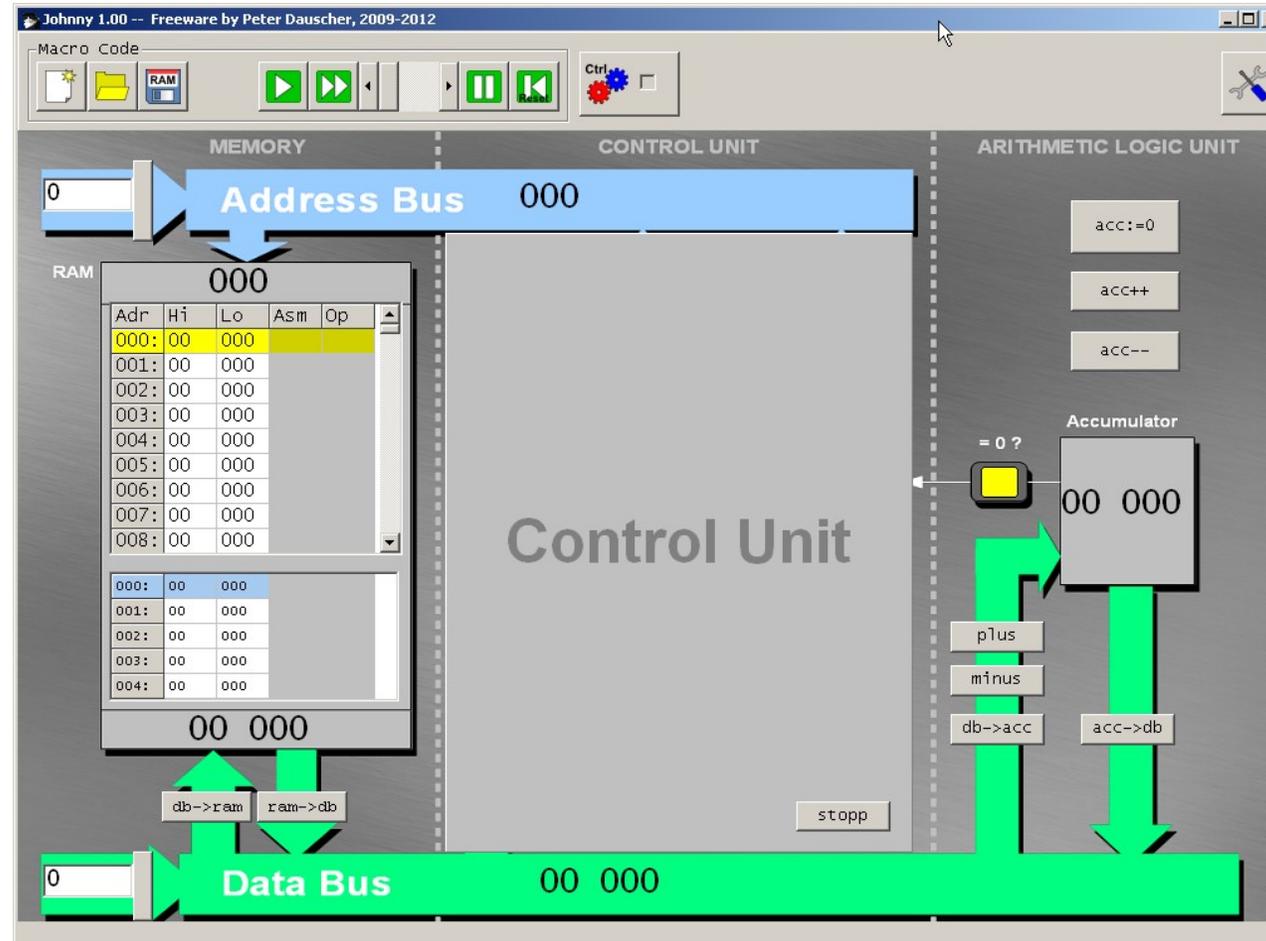


Quelle:

https://media.4teachers.de/images/thumbs/image_thumb.2642.jpg



2.5 Die Simulationssoftware JOHNNY



2.5 JOHNNY - Aufgaben

1. Aufbau Von-Neumann-Rechner
2. Der Von-Neumann Befehlszyklus
3. Sprunganweisungen
4. Zyklen in der Programmabarbeitung
5. Erweiterungen



2.5 JOHNNY – Aufgaben von Dirk Schwenn

Nicht nur Implementieren von Vorgaben, sondern auch

- Beschreibung von Wirkungen von Befehlen und Programmen
- Testen von Programmen
- Modifizieren, Erweitern und Entwickeln von Programmen
- Notieren von Quelltext und Speicherbelegungen
- Vergleich von korrekten und berechneten Ergebnissen



3. Logik-Gatter



3.1 Rahmenplan für Informatik und Medienbildung

Thema: Grundlagen der Digitalisierung

Klasse 10

Verbindliche Ziele und Inhalte	Hinweise und Anregungen
<p>den Aufbau und die grundlegende Funktionsweise eines Informatiksystems nach dem von-Neumann-Modell beschreiben</p> <p>Eignung binärer Signale für die maschinelle Verarbeitung erläutern</p> <ul style="list-style-type: none">• Bits logisch verknüpfen• binäre Addition	<p>Die Schülerinnen und Schüler nutzen ein Werkzeug zur Simulation maschinennaher Programmierung mit Hilfe eines vereinfachten von-Neumann-Modells. Sie erkennen die Zweckmäßigkeit der Verwendung höherer Programmiersprachen im Vergleich zur Maschinensprache.</p> <p>Die Prinzipien sind anschaulich mit einem Verweis für Möglichkeiten einer technischen Umsetzung zu vermitteln.</p>



**Austausch zu
Erfahrungen mit
Logikgattern**

3.2 Überblick zu Logikgattern

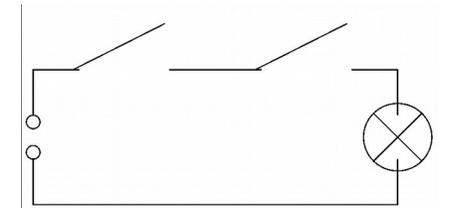
Logik

wahr \leftrightarrow falsch

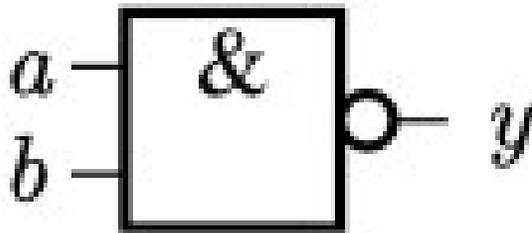
1 \leftrightarrow 0

Schaltungen

Strom fließt \leftrightarrow
Strom fließt nicht



Gatter



Gesetze

$$\overline{A \wedge B} = \overline{A} \vee \overline{B}$$

$$A \wedge \overline{B} = \overline{\overline{A} \vee B}$$

$$\overline{A} \vee \overline{B} = \overline{A \wedge B}$$

$$\overline{A} \vee B = \overline{A \wedge \overline{B}}$$

$$\overline{A} \wedge B = \overline{A \vee \overline{B}}$$

$$A \vee \overline{B} = \overline{\overline{A} \wedge B}$$

3.2 Überblick zu Logikgattern

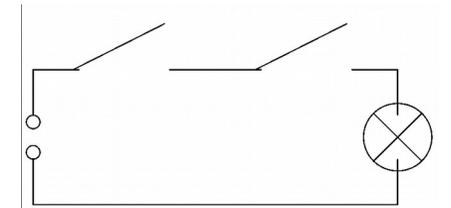
Logik

wahr \leftrightarrow falsch

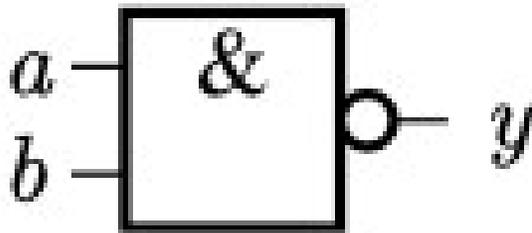
1 \leftrightarrow 0

Schaltungen

Strom fließt \leftrightarrow
Strom fließt nicht



Gatter



Gesetze

$$\bar{A} \wedge \bar{B} = \overline{A \vee B}$$

$$A \wedge \bar{B} = \overline{\bar{A} \vee B}$$

$$\bar{A} \vee \bar{B} = \overline{A \wedge B}$$

$$\bar{A} \vee B = \overline{A \wedge \bar{B}}$$

$$\bar{A} \wedge B = \overline{A \vee \bar{B}}$$

$$A \vee \bar{B} = \overline{\bar{A} \wedge B}$$

Logikbeispiel

Der Fahrstuhl bewegt sich, wenn
ein Knopf gedrückt wurde
und
die Tür zu ist.



Logikgatter-Tabelle

Legende



Öffner



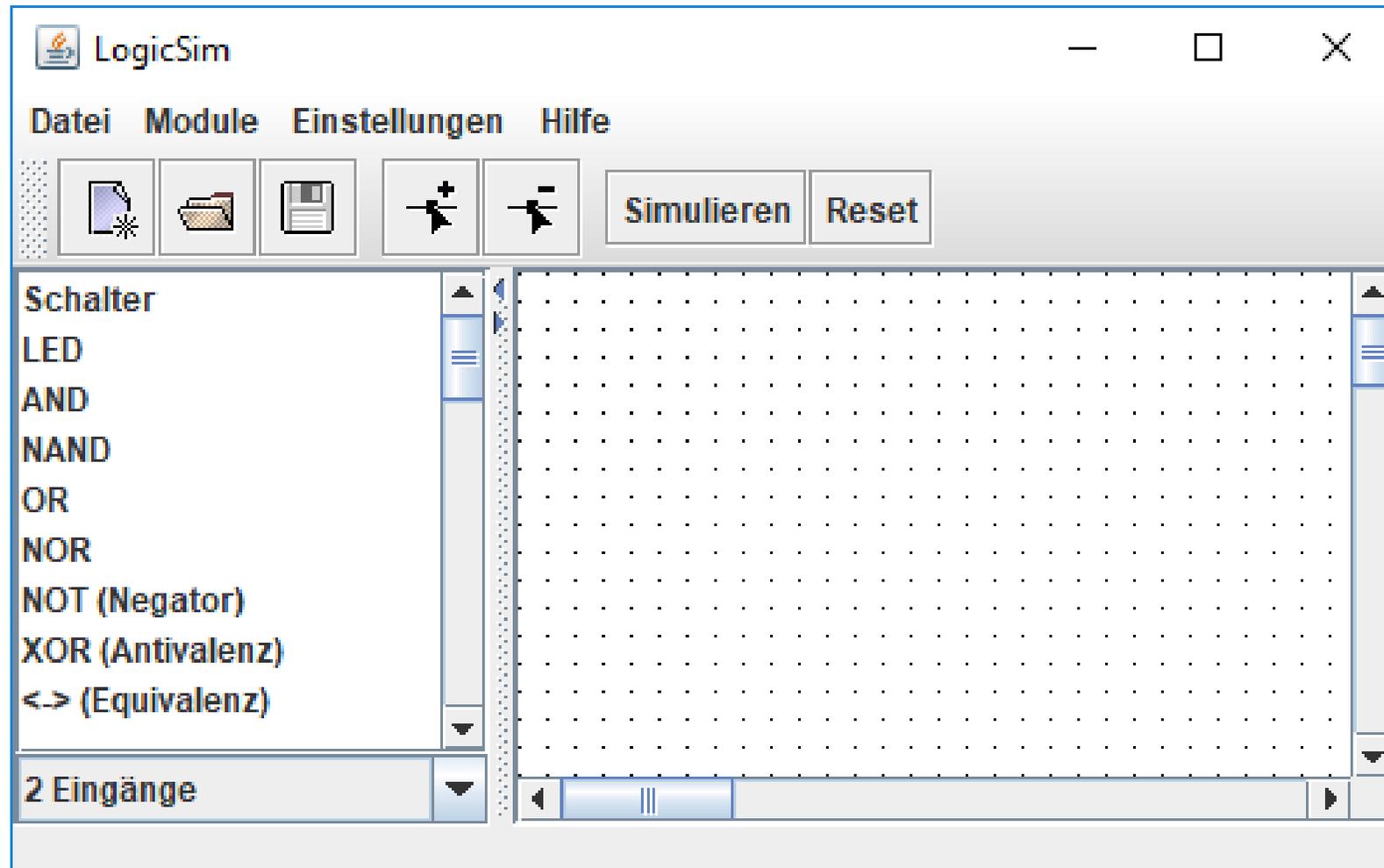
Wechselschalter



Schalter

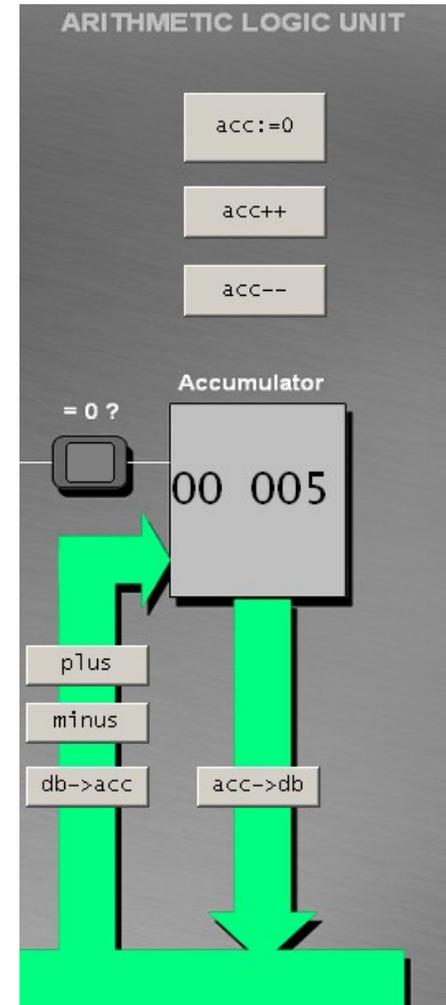


3.3 LogicSim



3.4 Das Rechenwerk

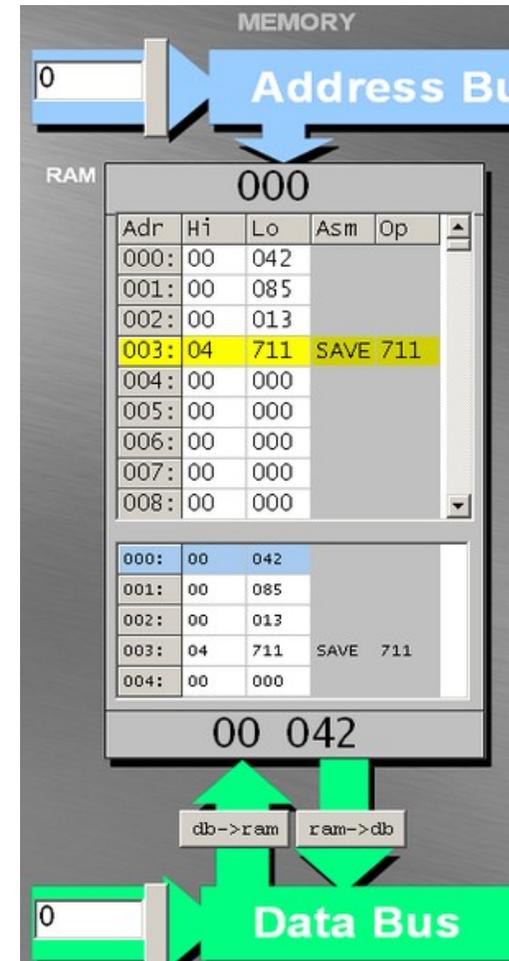
⇒ Ausführen von mathematischen, arithmetisch Rechenfunktionen und logischen Verknüpfungen.



**Wie kann das RECHENWERK
mittels Logikgattern
dargestellt werden?**

3.4 Das Speicherwerk

⇒ Speichert Programme und Daten,
die dem Rechenwerk zugänglich sind.



**Wie kann das Speichern
mittels Logikgattern
dargestellt werden?**

Alles ist Sprache

Maschinensprache

→ Sprache, deren Alphabet nur aus Gruppen fester Länge der Zeichen 0 und 1 besteht und vom Computer direkt verarbeitet werden kann

Bsp.:

1011 1000 000 1001 = B809_[16]

Assemblersprache

→ maschinenorientierte Programmiersprache, mit der ein Programm durch die Benutzung symbolischer Namen für Operanden und Operationen übersichtlicher wird

Bsp.:

→ add A, 0 (Addition zweier Zahlen)

Alle Befehle einer Maschinensprache werden durch den Prozessor nach dem gleichen Grundprinzip ausgeführt.



4. Inspirationen

- Dauscher, Peter: ZUM-Wiki: „**Rechenarchitektur mit Simulator JOHNNY**“, Interaktive Website: https://wiki.zum.de/wiki/Rechnerarchitektur_mit_Simulator_JOHNNY
- Dauscher, Peter: Inf-Schule: **7. Funktionsweise eines Rechners** – „Johnny-Modellrechner“, <https://www.inf-schule.de/rechner/johnny>
- Dauscher, Peter: „**Aufbau und Funktionsweise eines Von-Neumann-Rechners**“. In LOG IN, 33 Jg. (2013), Heft Nr. 175, S. 54 - 61
- The Computer Tutor, **Video-Interaktion** auf Englisch: <http://orionstarmedia.com/inc/sites/TheComputerTutor//interactive.html>
- GCFGlobal: „**Inside a computer**“ (Englisch), <https://edu.gcfglobal.org/en/computerbasics/inside-a-computer/1/>



4. Inspirationen

- App-Empfehlung: Suborbital Games: „**Circuit Scramble**“
<https://play.google.com/store/apps/details?id=com.Suborbital.CircuitScramble&hl=de>
- Dietz, Samuel: Inf-Schule: „**5. Digitaltechnik**“ (vom 11.12.2018)
<http://www.inf-schule.de/rechner/digitaltechnik>
- Schaltkreise in **GeoGebra**
<https://www.geogebra.org/classic/FeuwyUjj>
- Video: „**Film Zuse Babbage und der Computer 15Min**“ (vom 11.08.2011)
<https://www.youtube.com/watch?v=8Kc-g6gwbv8>

