Automatentheorie und Formale Sprachen

Sommersemester 2022
(LV 4110)

mittwochs, 11:45 bis 13:15

Prof. Dr. Bernhard Geib

Worum geht es in der Einführung?

- Womit beschäftigt sich die Theoretische Informatik?
  - ✓ Ziele und Merkmale
  - ✓ Themenbereiche und Abgrenzung
- Vorlesungsübersicht
  - ✓ Gliederung und Inhalte
  - ✓ Anwendungsbeispiele
- Organisation der Lehrveranstaltung
  - ✓ Vorlesung, Seminar und Klausur
  - ✓ Ablauf und Vereinbarung zur Leistungsbewertung
  - ✓ Hilfsmittel und Unterrichtsmaterial
  - ✓ Quellen- und Literaturangabe

Die vier wesentlichen Teilgebiete der Informatik:

Praktische Informatik	Angewandte Informatik	Technische Informatik
Theoretische Informatik		

- Die Theoretische Informatik ist das wissenschaftliche Fundament – sozusagen der theoretische Unterbau – der Informatik.
- Die Konzepte der Theoretischen Informatik sind ebenso fundamental wie abstrakt und anspruchsvoll in der Vermittlung.
  - ✓ prinzipielle Lösbarkeit von Problemen
  - ✓ Grenzen der Automatisierung
  - ✓ Modelle, mit deren Hilfe Problemlösungen auf grundsätzliche Machbarkeit hin überprüft und miteinander verglichen werden können

Teilgebiete der Informatik:

Praktische Informatik	Angewandte Informatik	Technische Informatik
Algorithmen, Datenstrukturen, Programmiermethoden	Graphik	Hardwarekomponenten
Programmiersprachen und Compiler	Datenbanken	Schaltnetze, Schaltwerke, Prozessoren
Betriebssysteme und Softwaretechnik	Künstliche Intelligenz	Mikroprogrammierung
	Simulation und Modellierung	Rechnerorganisation und -architekturen, Rechnernetze
	Textverarbeitung	
	Spezifische Anwendungen	

Theoretische Informatik

Automatentheorie und Formale Sprachen

Theorie der Berechenbarkeit

Komplexitätstheorie und Formale Semantik

Womit beschäftigt sich die Theoretische Informatik?

- **Automatentheorie** (Modellierung der prinzipiellen Funktion einer informationsverarbeitenden Maschine)
- **Algorithmentheorie** (Präzisierung der Begriffe Berechenbarkeit und Algorithmus)
- **Berechenbarkeitstheorie** (Gibt es zu jeder Problemstellung einen Lösungsalgorithmus?)
- **Komplexitätstheorie** (Einschätzung des Aufwandsverhaltens bezüglich Rechenzeit und Speicherplatz)
- **Theorie formaler Sprachen** (Abstraktion von Lexik, Syntax und Semantik)

Ziele und Merkmale der Theoretische Informatik

- Kennenlernen der grundsätzlichen Begriffe, Methoden und Beweistechniken
- Eindringen in die grundlegende Konzepte
- Erkenntnisse im Hinblick auf die praktische Lösbarkeit

und zwar **un**abhängig von konkreten Rechnern und aktuellen Technologien.

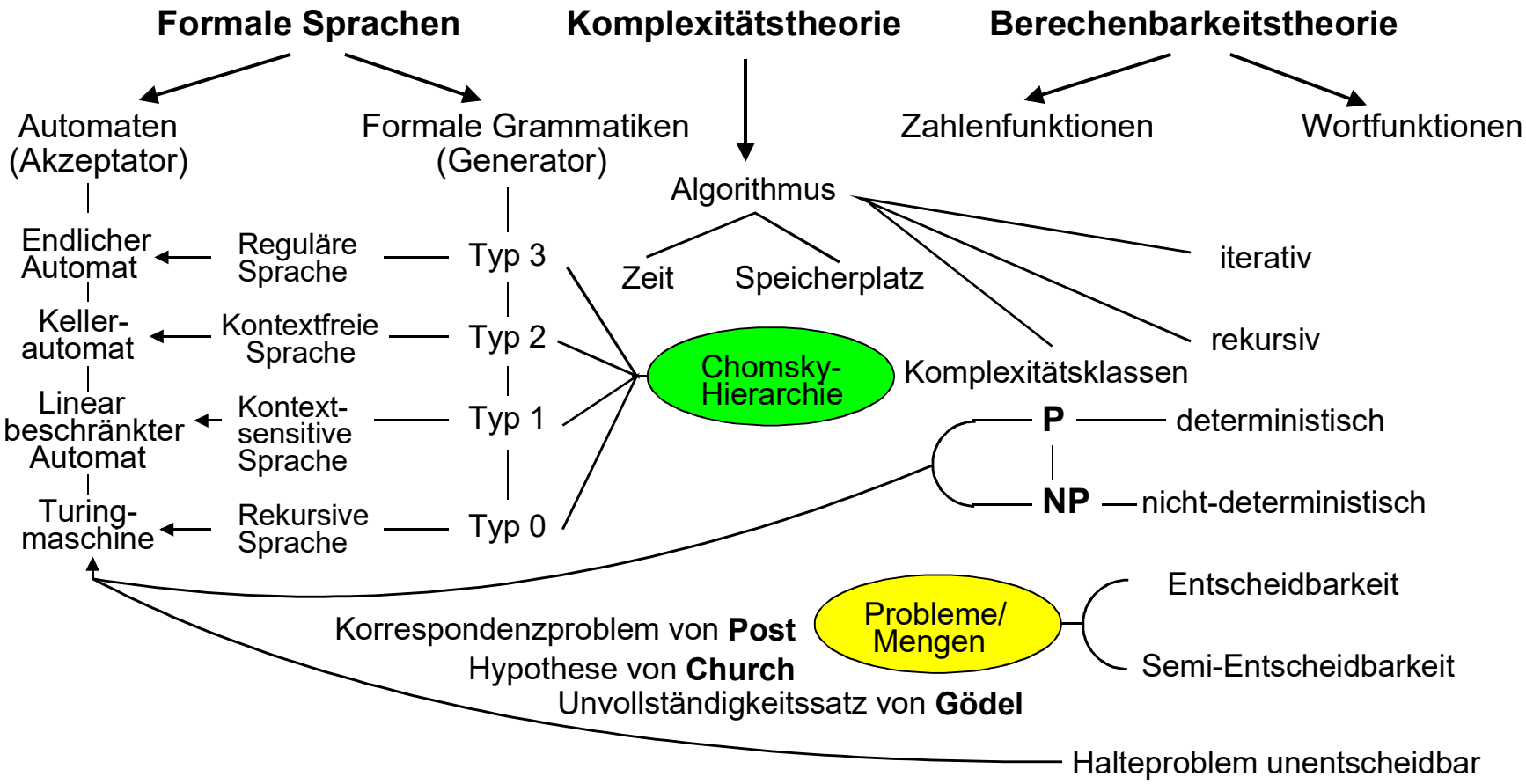
Wie beschäftigen uns mit Abstraktionen und Modellbildungen im Zusammenhang mit Problemen, die in irgendeiner Weise mit Hilfe von Computern gelöst werden sollen.

Grundziel: Unter dem Aspekt der Anwendung werden wichtige Hauptzweige der **Theoretischen Informatik** vorgestellt und anhand ausführlich behandelter Beispiele deren Bezug zu Problemlösungen der Praxis erläutert → **Praktische Informatik**

Gebiete:

- Lexikalische Analyse und Mustererkennung
- Definition von höheren Programmiersprachen (BNF, Syntaxgraphen)
- Compilerbau (Syntaxanalyse und Algorithmierbarkeit)
- Parallele Algorithmen und Berechenbarkeits- bzw. Komplexitätstheorie
- Sicherheitstechnik (Formale Verifikation von Sicherheitseigenschaften)

Zusammenhänge



Gliederung

1. Endliche Automaten (Automatentheorie, Modellierung und Überführungsfunktion, Zustandsgraphen und Funktionstafeln)
2. Reguläre Sprachen und Mengen (Reguläre Ausdrücke, Mengenoperationen und Verknüpfungen, Konstruktion von Automaten, Suchalgorithmen)
3. Grammatiken und Formale Sprachen (Semi-Thue-Systeme, Chomsky-Grammatiken und -Hierarchie, Ableitungsbäume)
4. Kellerautomaten und Kontextfreie Sprachen (Syntaxanalyse von Programmier-sprachen, Pumping-Lemma, Funktionsweise eines Kellerspeichers)
5. Turingmaschinen und Kontextsensitive Sprachen (Monotonie, Funktionsweise der Turingmaschine)
6. Entscheidbarkeit von Problemen und Berechenbarkeit von Algorithmen
7. Problemklassen und Komplexitätstheorie

Aufgabenstellung:

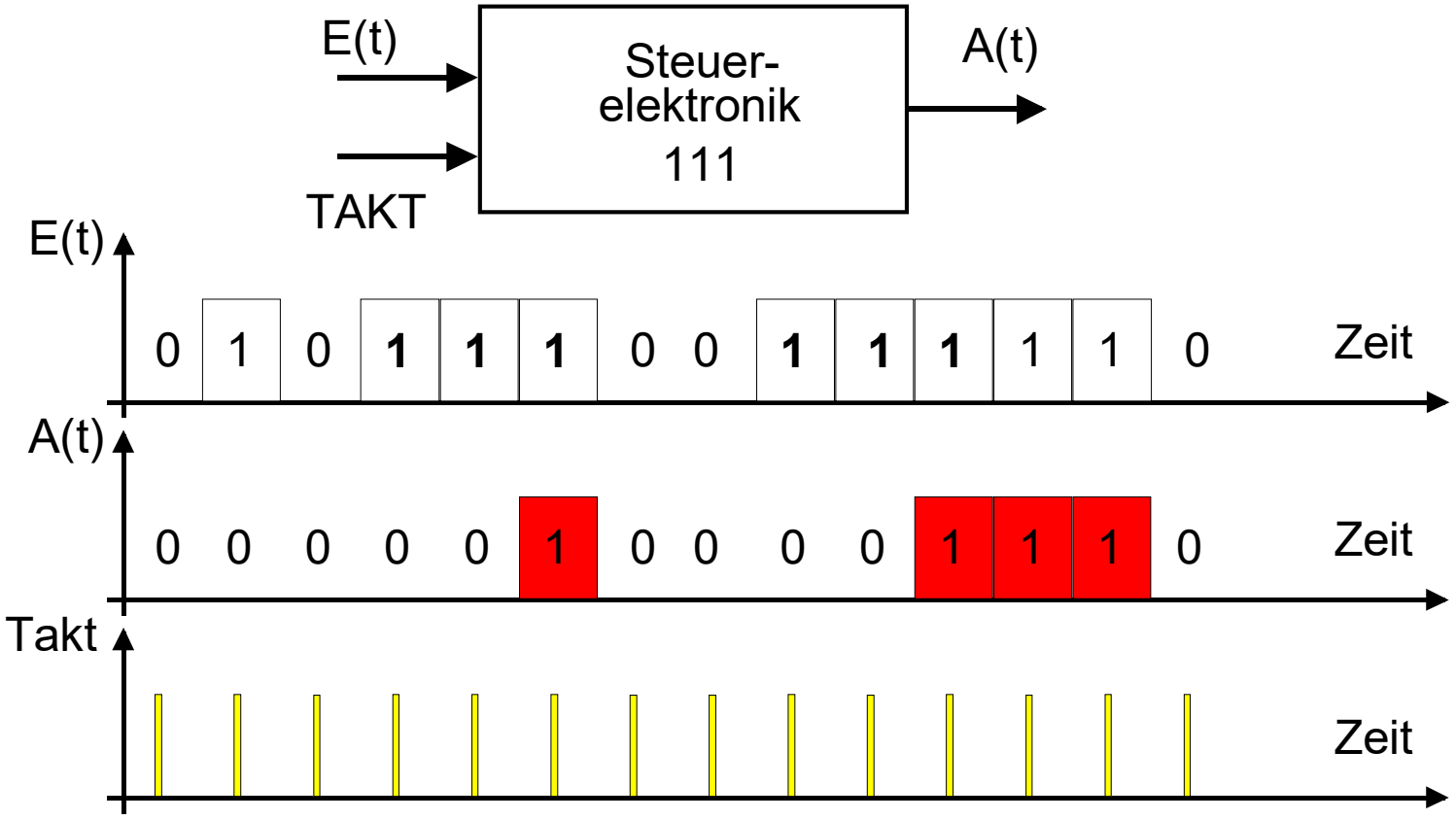
Entwerfen und Realisieren Sie unter Zuhilfenahme der Automatentheorie eine Steuerelektronik, die in einem binären Eingabestrom **E(t)** $\in \Sigma^*$ die Sequenz **111**, d. h. drei hintereinanderfolgende Einsen, erkennt. Am Ausgang der Steuereinheit **A(t)** $\in \Sigma^*$ soll dabei **A = 1** ausgegeben werden, sobald die Sequenz erkannt wurde, ansonsten soll **A = 0** sein.

Zur Lösung der Aufgabe bedienen wir uns dem Modell des deterministischen endlichen Automaten mit einer Ausgabefunktion, kurz **DFAwO**, der aufgrund der Ausgabefunktionalität nun folgende formale Beschreibung erfährt:

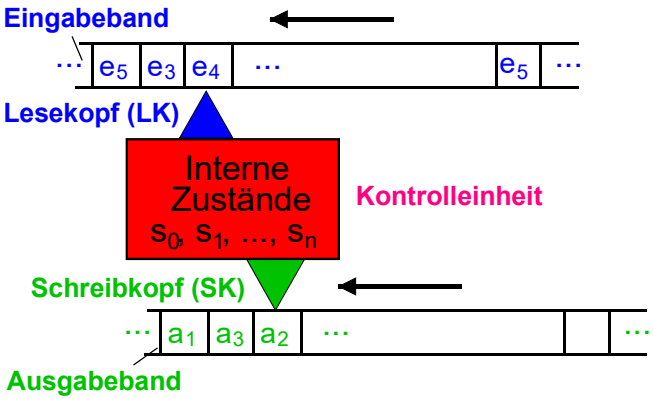
$$\mathbf{DFAwO} = (\Sigma = \{0, 1\}, \mathbf{S} = \{S_0, S_1, S_2, S_3\}, \delta, \{S_0\}, \mathbf{F} = \{S_3\}, \mathbf{A} = \{0, 1\}, \alpha)$$

Neben einem Taktgenerator und einigen elementaren Logikgattern (Negation, Konjunktion und Disjunktion) möge Ihnen zur Problemlösung zwei flankengesteuerte JK-Flip-Flops zur Verfügung stehen.

Veranschaulichung:



Zustandsautomat:

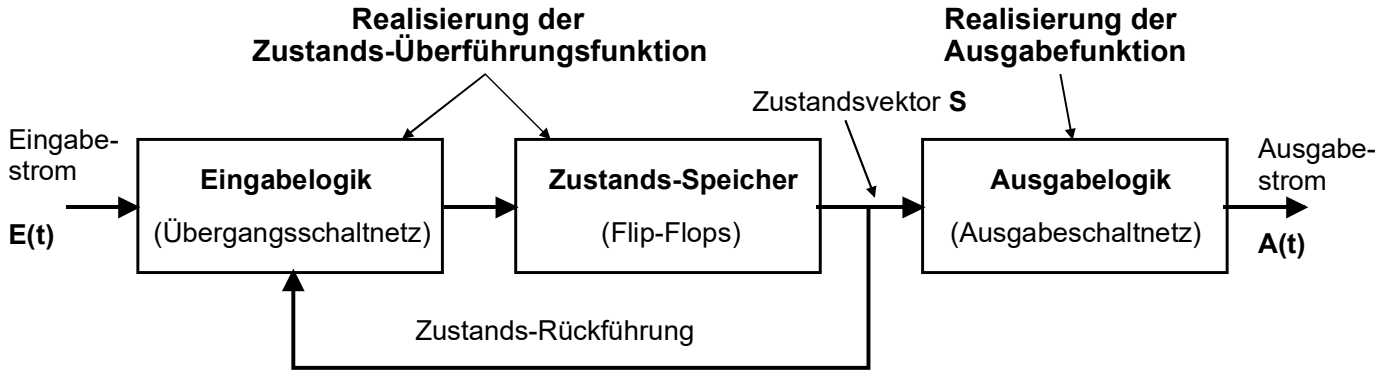


$\delta : S \times \Sigma \rightarrow S$

$\alpha : S \rightarrow A$

Lösungsidee:

Lese vom Eingabeband E(t) und schreibe auf das Ausgabeband A(t)



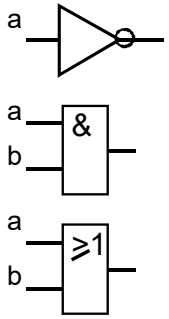
Zustandskodierung:

Logikgatter:

Negation

UND

ODER



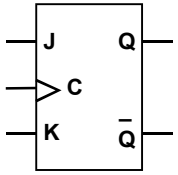
a	b		a & b	a | b	¬a
0	0		0	0	1
0	1		0	1	
1	0		0	1	0
1	1		1	1	

Speicherglieder:

JK-Flip-Flop
(flankengesteuertes)

J = Jump **K** = Kill

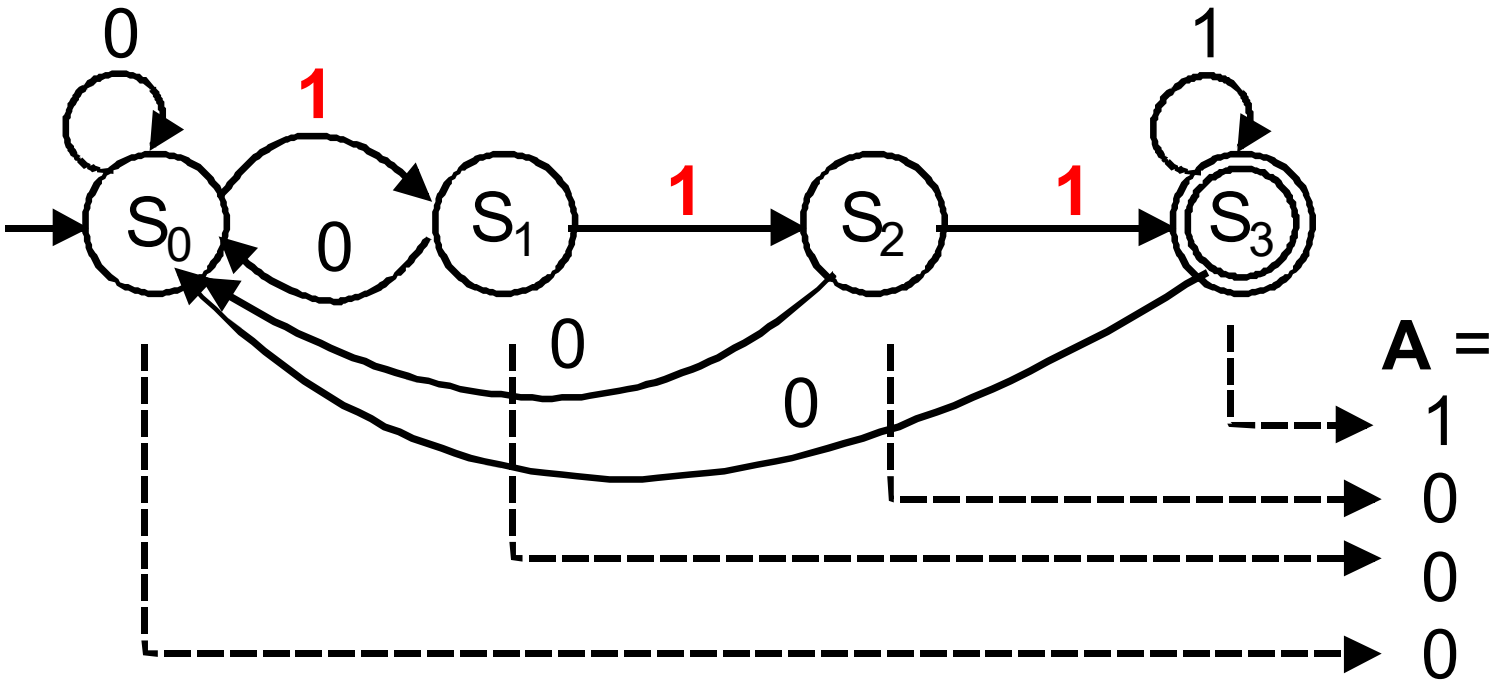
C = Clock



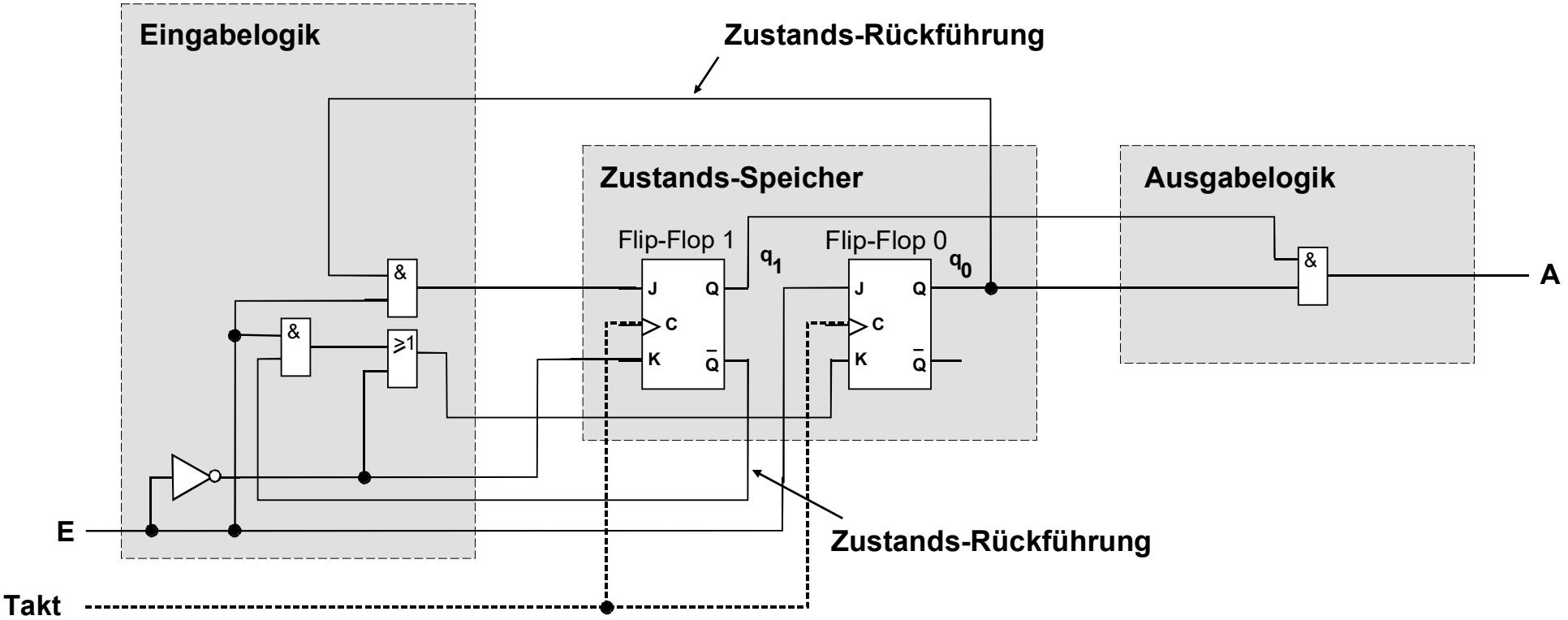
J	K		Q_{neu}	Wirkung
0	0		Q_{alt}	Speichern
0	1		0	Rücksetzen
1	0		1	Setzen
1	1		$\neg Q_{alt}$	Invertieren

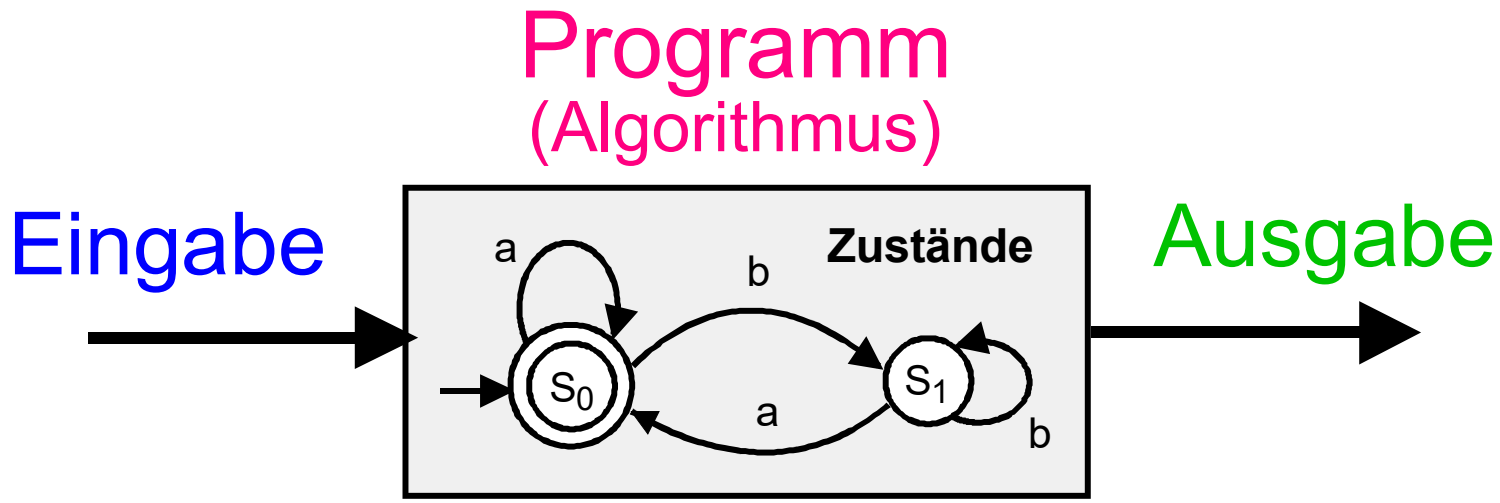
Zustandsdiagramm:

DFAwO = (Σ = {0, 1}, **S** = {S_0, S_1, S_2, S_3}, δ, {S_0}, **F** = {S_3}, **A** = {0, 1}, α)



Ergebnis:


Eingabelogik
Zustands-Rückführung
Zustands-Speicher
Ausgabelogik
Flip-Flop 1
Flip-Flop 0
q_1
q_0
J
C
K
Q
$\bar{Q}$
&
≥1
A
E
Takt
Zustands-Rückführung



- die **Eingabe** (Übernahme von Daten von außen)
- die **Wertzuweisung** (Zwischenrechnung, Zustände)
- die **Ausgabe** (Übertragung von Variablen nach außen)

Organisation

- Lehrform: 2 SWS Vorlesung, 2 SWS Seminar
- Credits / SWS: **5 cp / 4**
- Gesamtaufwand: **150 h** (etwa 8 h pro Woche)
  - Anwesenheit Vorlesung und Seminar 60 h
  - Vorbereitung und Nachbereitung Vorlesung 30 h
  - Bearbeitung der Übungsaufgaben (Seminar) 60 h
- Ort und Zeit:
  - Vorlesung findet mittwochs um 11:15 Uhr im Raum B 002 statt
  - Seminar erfolgt in kleinen Übungsgruppen gemäß Ankündigung und erfolgter Belegung

- Beim Seminar besteht **anwesenheitspflicht**
  - Eine 75%ige Anwesenheit muss **mindestens** erreicht worden sein
  - Bewerkstelligung von ca. 12 Übungsblättern (je 3 bis 5 Aufgaben)
  - Die Lösungen zu den zur Verfügung gestellten Übungsaufgaben sind zu den jeweilgen Seminarterminen anzufertigen
  - Das Seminar wird benotet (schriftlicher Test plus Übungsaufgaben)
- Klausur am Ende der Vorlesung
  - 90-minütige Klausur (Hilfsmittel: Merkblatt 2 DIN A4 Seiten)
  - Bearbeitung von 6 bis 8 unabhängigen Aufgaben aus dem Themengebiet „Automatentheorie und Formale Sprachen“
  - Die Klausur wird benotet

Folien und Übungsblätter zur Lehrveranstaltung

Befinden sich passwortgeschützt auf dem FB-Server und sind ausschließlich im Rahmen dieser Lehrveranstaltung zu verwenden.

www.cs.hs-rm.de/~rnlab/LVaktuell/AFS/Vorlesung/
www.cs.hs-rm.de/~rnlab/LVaktuell/AFS/Seminar/

Sprechstunde

Außerhalb der Lehrveranstaltungszeiten jeweils

mittwochs zwischen 10:30 und 11:30 Uhr im Raum C 210
oder nach Vereinbarung

Literatur

[1] J. Albert, Th. Ottmann: *Automaten, Sprachen und Maschinen für Anwender*, B.I.-Wissenschaftsverlag, Reihe Informatik/38, Zürich 1983

[2] Uwe Schöning: *Theoretische Informatik kurz gefaßt*, B.I.-Wissenschaftsverlag, Mannheim 1992

[3] Sander, Stucky, Herschel: *Automaten – Sprachen – Berechenbarkeit*, B.G. Teubner, Stuttgart 1992

[4] Ingo Wegner: *Theoretische Informatik – eine algorithmische Einführung*, B.G. Teubner, Stuttgart 1999

[5] M. Broy: *Informatik – Eine grundlegende Einführung*, Teil IV, Theoretische Informatik, Springer, 1995

[6] Daniel I. A. Cohen: *Introduction to Computer Theory*, John Wiley & Sons, Inc., 1997