UNI

Informatik I: Einführung in die Programmierung

19. Automaten, Zustandsmodellierung mit abstrakten und generischen Klassen



Prof. Dr. Peter Thiemann



Deterministische endliche Automaten

Deterministische endliche Automaten

Motivierendes Beispiel

Formale

Grundlagen

Automaten

Teilwort-Erkennung

Transdukto-

Modellierung von Automaten

Automaten

Welt & Modell

Vorweg ...





Vor kurzem fand ich auf dem Dachboden eines Kollegen einen Würfel und einen Brief.

Deterministische endliche Automaten

Motivierendes Beispiel

Formale Grundlagen

Verhalten eines Automaten Teilwort-Erkennung

reliwort-Erkerinul

Transduktoren

Modellierung von Automaten

Welt & Modell

Vorweg ...



BUR

Vor kurzem fand ich auf dem Dachboden eines Kollegen einen Würfel und einen Brief.



Deterministische endliche Automaten

Motivierendes Beispiel

Beispiel Formale

Grundlagen
Verhalten eines

Teilwort-Erkennung

Transduktoren

Modellierung von Automaten

Welt & Modell

Vorweg ...



RURG

H

Vor kurzem fand ich auf dem Dachboden eines Kollegen einen Würfel und einen Brief.



Lieber Finder dieses Briefes. der Würfel ist gefallen: 5-1-5-6 ergibt x, 5-4-2-5 ergibt v. Bei N 48° 00,x' 0 7° 50,y' in 15 Meter Tiefe wirst du einen sagenhaften Schatz finden, der das Bernsteinzimmer wie eine Studentenbude aussehen läßt. Ich wünsche Dir viel Glück bei deiner Suche! Fmil Nebel Freiburg, 1980

Deterministische endliche Automaten

Motivierendes Beisniel

Formale

Grundlagen Verhalten eine

Teilwort-Erkennun

Transdukto-

Modellierung von Automaten

Welt &



In dem Würfel gibt es einen Mechanismus, der die Abfolge der nach oben gerichteten Würfelseiten erkennt. Determinis sche endliche

Automaten Motivierendes

Beispiel

Grundlagen Verhalten eines

Teilwort-Erkennur

Transdukto-

Modellierung von Automaten

Welt & Modell





- In dem Würfel gibt es einen Mechanismus, der die Abfolge der nach oben gerichteten Würfelseiten erkennt.
- Nachdem die richtige Folge "gewürfelt" wurde, klopft von innen ein Hämmerchen die Koordinaten.

Determinist sche endliche Automaten

Motivierendes Beispiel

Grundlagen Verhalten eines

Teilwort-Erkennung

Transdukto-

Modellierung von

Automaten

Welt & Modell

ren



sche

Motivierendes Beispiel

Transdukto-

ren

Modellierung Automaten

Welt & Modell

- In dem Würfel gibt es einen Mechanismus, der die Abfolge der nach oben gerichteten Würfelseiten erkennt.
- Nachdem die richtige Folge "gewürfelt" wurde, klopft von innen ein Hämmerchen die Koordinaten.
- Wie erkennt der Würfel solche Folgen von Ereignissen?





- In dem Würfel gibt es einen Mechanismus, der die Abfolge der nach oben gerichteten Würfelseiten erkennt.
- Nachdem die richtige Folge "gewürfelt" wurde, klopft von innen ein Hämmerchen die Koordinaten.
- Wie erkennt der Würfel solche Folgen von Ereignissen?
- Hierfür ist ein endlicher Automat geeignet.

Determinist sche endliche Automaten

Motivierendes Beispiel

Grundlagen Verhalten eine

Teilwort-Erkennu

Transdukto-

Modellierung von Automaten

Welt &



SH -

In dem Würfel gibt es einen Mechanismus, der die Abfolge der nach oben gerichteten Würfelseiten erkennt.

Nachdem die richtige Folge "gewürfelt" wurde, klopft von innen ein Hämmerchen die Koordinaten.

- Wie erkennt der Würfel solche Folgen von Ereignissen?
- Hierfür ist ein endlicher Automat geeignet.
- Ein endlicher Automat ist ein sehr einfaches und eingeschränktes Berechnungsmodell, das für viele Anwendungen adäquat ist.

Determinist sche endliche Automaten

Motivierendes Beispiel

Grundlagen Verhalten eine Automaten

Teilwort-Erkennu

Transdukto-

Modellierung von Automaten

Welt & Modell





- In dem Würfel gibt es einen Mechanismus, der die Abfolge der nach oben gerichteten Würfelseiten erkennt.
- Nachdem die richtige Folge "gewürfelt" wurde, klopft von innen ein Hämmerchen die Koordinaten.
- Wie erkennt der Würfel solche Folgen von Ereignissen?
- Hierfür ist ein endlicher Automat geeignet.
- Ein endlicher Automat ist ein sehr einfaches und eingeschränktes Berechnungsmodell, das für viele Anwendungen adäquat ist.
- Wir können ihn durch eine Klasse definieren

Motivierendes Beispiel

Transduktoren

Automaten

Welt &

fassung & Ausblick

Automaten akzeptieren Sprachen



Z

Ein deterministischer Automat ist ein Quintupel (Q, E, δ, q_0, F)

Die accept Methode nimmt ein Eingabewort, lässt den Automaten vom Startzustand bis zum Ende des Worts laufen. Das Wort wird akzeptiert, wenn der Automat einen Endzustand erreicht

Determinist sche endliche Automaten

> Formale Grundlagen

Verhalten eines Automaten

Transdukto-

Modellierung von Automaten

Welt &



NE SE

Die Transitionsfunktion wird entweder durch eine Transitionstabelle oder durch ein Transitionsdiagramm (Transitionsdiagramm) angegeben.

Beispiel: Transitionsfunktion zur Erkennung von 5156

 $(q_e$ bezeichnet einen Fehlerzustand und $F = \{q_f\}$).

Determinis sche endliche Automaten

> Beispiel Formale

Grundlagen

Verhalten eines Automaten

Teilwort-Erkennung

Transdukto-

Modellierung von Automaten

Welt & Modell



EIBUR

Die Transitionsfunktion wird entweder durch eine Transitionstabelle oder durch ein Transitionsdiagramm (Transitionsdiagramm) angegeben.

Beispiel: Transitionsfunktion zur Erkennung von 5156

 $(q_e$ bezeichnet einen Fehlerzustand und $F = \{q_f\}$).

	1	2	3	4	5	6
q_0	q _e	q _e	q _e	q _e	<i>q</i> ₁	q _e
<i>q</i> ₁	q_2	qe	qe	qe	qe	qe
q_2	q _e	q _e	q _e	q _e	q_3	q _e
q_3	q _e	q_f				
q_f	qe	q _e	q _e	qe	q _e	q _e
a _o	a _o	a.	a _o	a.	a.	a.

Determinist sche endliche Automaten

> Beispiel Formale

Grundlagen

Verhalten eines Automaten

Teilwort-Erkennu

Transduktoren

Modellierung von Automaten

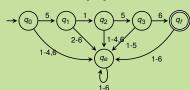
Welt & Modell

Die Transitionsfunktion wird entweder durch eine Transitionstabelle oder durch ein Transitionsdiagramm (Transitionsdiagramm) angegeben.

Beispiel: Transitionsfunktion zur Erkennung von 5156

 $(q_e$ bezeichnet einen Fehlerzustand und $F = \{q_f\}$).

	1	2	3	4	5	6
q_0	q _e	q _e	q _e	q _e	<i>q</i> ₁	q _e
<i>q</i> ₁	q_2	q _e	q _e	q _e	q _e	qe
q_2	q _e	q _e	q _e	q _e	<i>q</i> ₃	q _e
q_3	q _e	q_f				
q_f	qe	q _e	qе	q _e	q _e	qe
a _o	ao	a _o	a	a _o	a _o	ao



Determinist sche endliche Automaten

> Beispiel Formale

Formale Grundlagen

Verhalten ein

Teilwort-Erkennu

Transdukto-

Modellierung von

Automaten
Welt &

Modell

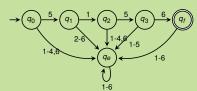


Die Transitionsfunktion wird entweder durch eine Transitionstabelle oder durch ein Transitionsdiagramm (Transitionsdiagramm) angegeben.

Beispiel: Transitionsfunktion zur Erkennung von 5156

 $(q_e$ bezeichnet einen Fehlerzustand und $F = \{q_f\}$).

	1	2	3	4	5	6
90	q _e	q _e	q _e	q _e	q_1	q _e
<i>q</i> ₁	q_2	q _e	qe	qe	qe	qe
q_2	q _e	q _e	q _e	q _e	q_3	q _e
q_3	q _e	q_f				
q_f	qe	q _e	qe	qe	qe	qe
q _e	q _e	q _e	q _e	q _e	q _e	q _e



Beachte: Im Transitionsdiagramm wird der absorbierende Fehlerzustand q_e mit allen Übergängen dorthin in der Regel nicht angegeben.

Determinist sche endliche Automaten

> Beispiel Formale

Formale Grundlagen

Verhalten eines Automaten

Teilwort-Erkennur

Transduktoren

Modellierung von Automaten

Welt &



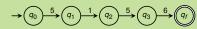
Die Transitionsfunktion wird entweder durch eine Transitionstabelle oder durch ein Transitionsdiagramm (Transitionsdiagramm) angegeben.

Beispiel: Transitionsfunktion zur Erkennung von 5156

 (q_e) bezeichnet einen Fehlerzustand und $F = \{q_f\}$).

	1	2	3	4	5	6
$\overline{q_0}$	q _e	q _e	q _e	q _e	q_1	q _e
	q_2	q _e	q _e	qe	q _e	q _e
q_2	q _e	q _e	q _e	q _e	q_3	q _e
q_3	q _e	q_f				
q_f	qe	qe	qe	qe	qe	qe
q _e	q _e	q _e	qe	q _e	q _e	qe

nicht angegeben.



Beachte: Im Transitionsdiagramm wird der absorbierende Fehlerzustand q_e mit allen Übergängen dorthin in der Regel

sche endliche

Formale

Grundlagen

Transduktoren

Modellierung Automaten

Welt & Modell

fassung & Ausblick

Implementierung der Transitionsfunktion



```
class State (Enum):
 Q0 = auto(); Q1 = auto(); Q2 = auto(); Q3 = auto()
 Qf = auto(); Qe = auto()
type Alphabet = Literal['1','2','3','4','5','6']
def delta_dice (q:State, c:Alphabet) -> State:
 match (q, c):
    case (State.Q0, '5'):
      return State.Q1
    case (State.Q1, '1'):
      return State.02
    case (State.Q2, '5'):
      return State.Q3
    case (State.Q3, '6'):
     return State.Of
    case :
     return State.Qe
```

sche endliche Automaten

Formale

Grundlagen

Automaten Teilwort-Erkennu

Transdukto-

Modellierung von Automaten

Welt &

Zusammenfassung & Ausblick

Der Typ Literal ['1', '2', '3', '4', '5', '6'] bestimmt das

Eingahaalnhahatl 22 01 2025

P Thiemann - Info I





■ Anfänglich befindet sich der Automat im Startzustand q_0 .

Deterministische endliche Automaten

Motivierendes Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten
Teilwort-Erkennung

Transdukto-

ren

Modellierung von Automaten

Welt & Modell





- Anfänglich befindet sich der Automat im Startzustand q_0 .
- Der Automat erhält ein Wort $w = "a_1 a_2 ... a_n"$ als Eingabe (darf auch leer sein, d.h. $n \ge 0$).

Determinist sche endliche Automaten

> Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten Teilwort-Erkennun

Transdukto-

ren

Modellierung von Automaten

Welt & Modell



9/44



- Anfänglich befindet sich der Automat im Startzustand q_0 .
- Der Automat erhält ein Wort $w = "a_1 a_2 ... a_n"$ als Eingabe (darf auch leer sein, d.h. $n \ge 0$).
- Der Automat liest im Zustand q_i das Eingabesymbol a_{i+1} und wechselt in den Folgezustand $q_{i+1} = \delta(q_i, a_{i+1})$.

Determinist sche endliche Automaten

> Motivierendes Beispiel

Formale Grundlagen

Verhalten eines Automaten

Teilwort-Erkennung

Transdukto-

Modellierung von Automaten

Welt & Modell



- Anfänglich befindet sich der Automat im Startzustand q_0 .
- Der Automat erhält ein Wort $w = "a_1 a_2 ... a_n"$ als Eingabe (darf auch leer sein, d.h. n > 0).
- Der Automat liest im Zustand q_i das Eingabesymbol a_{i+1} und wechselt in den Folgezustand $q_{i+1} = \delta(q_i, a_{i+1})$.
- Das macht der Automat, so lange Eingabezeichen gelesen werden können, das heißt für $i \in \{0, 1, ..., n-1\}$.

Determinis sche endliche Automaten

> Motivierendes Beispiel Formale

Verhalten eines

Teilwort-Erkennung

Transdukto-

Modellierung von Automaten

Welt &



- Anfänglich befindet sich der Automat im Startzustand q_0 .
- Der Automat erhält ein Wort $w = "a_1 a_2 ... a_n"$ als Eingabe (darf auch leer sein, d.h. n > 0).
- Der Automat liest im Zustand q_i das Eingabesymbol a_{i+1} und wechselt in den Folgezustand $q_{i+1} = \delta(q_i, a_{i+1})$.
- Das macht der Automat, so lange Eingabezeichen gelesen werden können, das heißt für $i \in \{0, 1, ..., n-1\}$.
- Am Ende der Eingabe befindet sich der Automat in einem Zustand q_n . Das Eingabewort w wird genau dann akzeptiert, wenn $q_n \in F$ ein Endzustand ist.

Determinis sche endliche Automaten

> Motivierendes Beispiel Formale

Grundlagen
Verhalten eines

Teilwort-Erkennung

Transdukto-

Modellierung von Automaten

Welt & Modell



- FRE
- Anfänglich befindet sich der Automat im Startzustand q₀.
- Der Automat erhält ein Wort $w = "a_1 a_2 ... a_n"$ als Eingabe (darf auch leer sein, d.h. n > 0).
- Der Automat liest im Zustand q_i das Eingabesymbol a_{i+1} und wechselt in den Folgezustand $q_{i+1} = \delta(q_i, a_{i+1})$.
- Das macht der Automat, so lange Eingabezeichen gelesen werden können, das heißt für $i \in \{0, 1, ..., n-1\}$.
- Am Ende der Eingabe befindet sich der Automat in einem Zustand q_n . Das Eingabewort w wird genau dann akzeptiert, wenn $q_n \in F$ ein Endzustand ist.
- Die Sprache von A, $\mathcal{L}(A)$, ist die Menge aller von A akzeptierten Worte.

Determinist sche endliche Automaten

> Motivierendes Beispiel Formalo

Grundlagen
Verhalten eines

Teilwort-Erkennung

Transdukto-

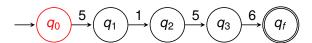
Modellierung von Automaten

Welt & Modell





Eingabe: 5156



Determinist sche endliche Automaten

> Motivierendes Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten Teilwort-Erkennung

Transduktoren

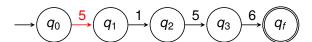
Modellierung von Automaten

Welt & Modell



A SEC

Eingabe: 5156



Determinist sche endliche Automaten

Motivierendes Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten Teilwort-Erkennung

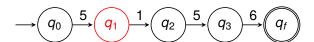
Transduktoren

Modellierung von Automaten

Welt & Modell



Resteingabe: 156



Determinist sche endliche Automaten

> Motivierendes Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten Teilwort-Erkennung

Transduktoren

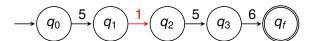
Modellierung von Automaten

Welt & Modell



RE --

Resteingabe: 156



Determinist sche endliche Automaten

> Motivierendes Beispiel Formale

Formale Grundlagen

Verhalten eines Automaten

Teilwort-Erkennung

Transduktoren

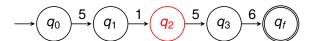
Modellierung von Automaten

Welt & Modell



FREE E

Resteingabe: 56



Determinist sche endliche Automaten

Motivierendes Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten Teilwort-Erkennung

Transdukto-

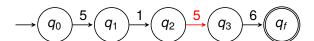
ren

Modellierung von Automaten

Welt & Modell



Resteingabe: 56



Determinist sche endliche Automaten

Motivierendes Beispiel

Formale Grundlagen Verhalten eines

Automaten Teilwort-Erkennung

Teilwort-Erkennur

Transduktoren

Modellierung von Automaten

Welt & Modell

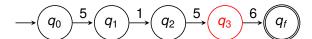


REIBUR

Resteingabe:

6

,



Determinist sche endliche Automaten

Motivierendes Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten Teilwort-Erkennung

Transdukto-

Modellierung von Automaten

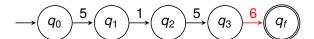
Welt &

ren



Resteingabe:

6



Determinist sche endliche Automaten

Motivierendes Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten Teilwort-Erkennung

Transduktoren

Modellierung von Automaten

Welt & Modell



REIBL

Resteingabe:

Eingabe akzeptiert



Determinist sche endliche Automaten

> Motivierendes Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten
Teilwort-Erkennung

Transdukto-

Transduktoren

Modellierung von Automaten

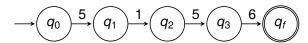
Welt & Modell



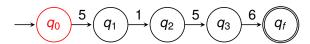
FREIBL

Resteingabe:

Eingabe akzeptiert



Eingabe: 515156



Deterministische endliche Automaten

> Motivierendes Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten

Teilwort-Erkennung

Transduktoren

Modellierung von Automaten

Welt & Modell

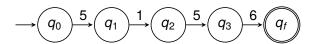


10 / 44

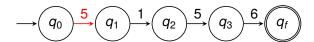
Z W

Resteingabe:

Eingabe akzeptiert



Eingabe: 515156



Deterministische endliche Automaten

Motivierendes Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten
Teilwort-Erkennung

Transduktoren

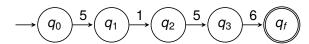
Modellierung von Automaten

Welt & Modell

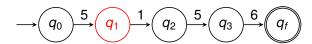


Resteingabe:

Eingabe akzeptiert



Resteingabe: 15156



Deterministische endliche Automaten

> Motivierendes Beispiel Formale

Verhalten eines

Automaten Teilwort-Erkennung

Transduktoren

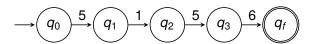
Modellierung von Automaten

Welt & Modell

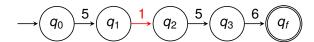


Resteingabe:

Eingabe akzeptiert



Resteingabe: 15156



Deterministische endliche Automaten

Motivierendes Beispiel

Formale

Verhalten eines Automaten

Teilwort-Erkennung

Transdukto-

ren Modellierung

von Automaten

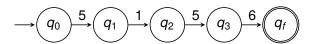
Welt & Modell



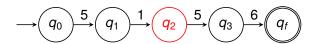
AR A

Resteingabe:

Eingabe akzeptiert



Resteingabe: 5156



Deterministische endliche Automaten

> Motivierendes Beispiel Formale

Formale Grundlagen

Verhalten eines Automaten

Teilwort-Erkennung

Transduktoren

Modellierung von Automaten

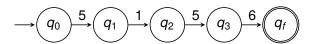
Welt & Modell



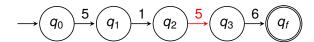
FREIB

Resteingabe:

Eingabe akzeptiert



Resteingabe: 5156



Deterministische endliche Automaten

Motivierendes Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten

Teilwort-Erkennung

Transduktoren

Modellierung von Automaten

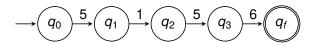
Welt & Modell



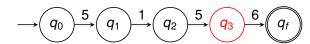
REIBUR

Resteingabe:

Eingabe akzeptiert



Resteingabe: 156



Deterministische endliche Automaten

> Motivierendes Beispiel

Formale Grundlagen

Verhalten eines Automaten

Teilwort-Erkennung

Transduktoren

Modellierung von Automaten

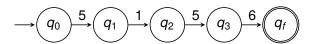
Welt & Modell



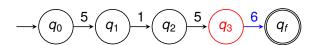
X

Resteingabe:

Eingabe akzeptiert



Resteingabe: 156



Deterministische endliche Automaten

> Motivierendes Beispiel Formale

Formale Grundlagen Verhalten eines

Automaten
Teilwort-Erkennung

Teilwort-Erkennu

Transduktoren

Modellierung von Automaten

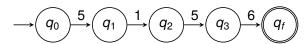
Welt & Modell



REB

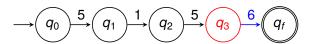
Resteingabe:

Eingabe akzeptiert



Resteingabe: 156

Kein Transition von q_3 aus möglich! Eingabe nicht akzeptiert.



Determinist sche endliche Automaten

> Motivierendes Beispiel

ormale Grundlagen

Verhalten eines Automaten

Teilwort-Erkennu

Transduktoren

Modellierung

von Automaten

Welt & Modell

Implementierung des Automaten



```
dice : Automaton[State, Alphabet] = Automaton(
  delta = delta_dice,
  start = State.QO,
  finals = frozenset ({State.Qf})
print(dice.accept("5156"))
```

Ausgabe: True

```
print(dice.accept("5155"))
```

Ausgabe: False

sche endliche Automaten

Verhalten eines Automaten Teilwort-Erkennung

Transdukto-

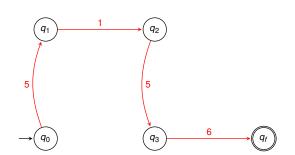
ren

Modellierung von Automaten

Welt & Modell



Der Würfel soll letztlich alle Folgen akzeptieren, die 5156 als Teilwort enthalten



Deterministische endliche Automaten

Motivierendes

Verhalten eines Automaton

Teilwort-Erkennung

Transdukto-

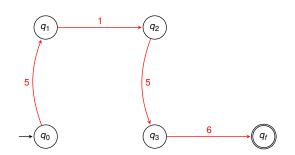
ren

Modellierung von Automaten

Welt & Modell

NE B

Der Würfel soll letztlich alle Folgen akzeptieren, die 5156 als Teilwort enthalten, z.B. auch 55156



Deterministische endliche Automaten

Motivierendes Beispiel

Formale Grundlagen

Verhalten eines Automaten

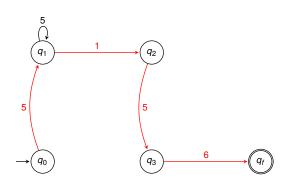
Teilwort-Erkennung

Transduktoren

Modellierung von Automaten

Welt & Modell

Der Würfel soll letztlich alle Folgen akzeptieren, die 5156 als Teilwort enthalten, z.B. auch 55156



Deterministische endliche Automaten

> Motivierendes Beispiel

Formale Grundlagen

Verhalten eines Automaten

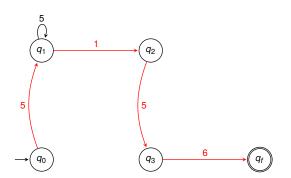
Teilwort-Erkennung

Transduktoren

Modellierung von Automaten

Welt &

Der Würfel soll letztlich alle Folgen akzeptieren, die 5156 als Teilwort enthalten, z.B. auch 55156, oder 5155156



Deterministische endliche Automaten

Beispiel

Formale Grundlagen

Verhalten eines Automaten

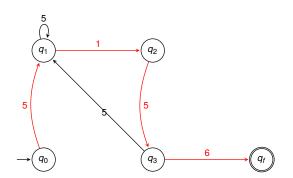
Teilwort-Erkennung

Transdukto-

Modellierung von Automaten

Welt &

Der Würfel soll letztlich alle Folgen akzeptieren, die 5156 als Teilwort enthalten, z.B. auch 55156, oder 5155156



Deterministische endliche Automaten

Beispiel

Formale Grundlagen

Automaten

Teilwort-Erkennung

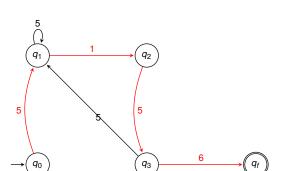
Transdukto-

Modellierung von Automaten

Welt & Modell



Der Würfel soll letztlich alle Folgen akzeptieren, die 5156 als Teilwort enthalten, z.B. auch 55156, oder 5155156, oder 515156



Deterministische endliche Automaten

Beispiel

Formale Grundlagen

Automaten

Teilwort-Erkennung

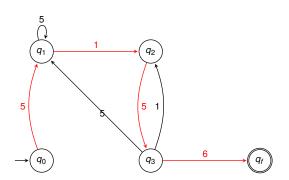
Transdukto-

Modellierung von Automaten

Welt &



Der Würfel soll letztlich alle Folgen akzeptieren, die 5156 als Teilwort enthalten, z.B. auch 55156, oder 5155156, oder 515156



Deterministische endliche Automaten

Beispiel

Formale Grundlagen

Verhalten ei

Teilwort-Erkennung

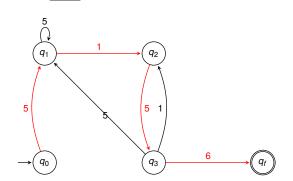
Transdukto-

Modellierung von Automaten

Welt &



Der Würfel soll letztlich alle Folgen akzeptieren, die 5156 als Teilwort enthalten, z.B. auch 55156, oder 5155156, oder 515156 oder ... 5156...



Deterministische endliche Automaten

> Motivierendes Beispiel

Formale Grundlagen

Verhalten ei

Teilwort-Erkennung

Transdukto-

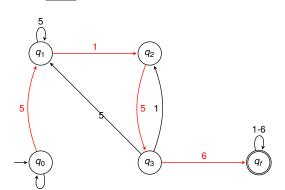
Modellierung von Automaten

Welt &

1-4.6



Der Würfel soll letztlich alle Folgen akzeptieren, die 5156 als Teilwort enthalten, z.B. auch 55156, oder 5155156, oder 515156 oder ... 5156...



Deterministische endliche Automaten

> Motivierendes Beispiel

Formale

Verhalten ein

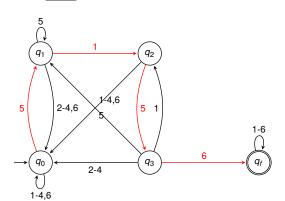
Teilwort-Erkennung

Transdukto-

Modellierung von Automaten

Welt &

Der Würfel soll letztlich alle Folgen akzeptieren, die 5156 als Teilwort enthalten, z.B. auch 55156, oder 5155156, oder 515156 oder ... 5156...



Deterministische endliche Automaten

Beispiel

Formale Grundlagen

Automaten

Teilwort-Erkennung

Transdukto-

Modellierung von Automaten

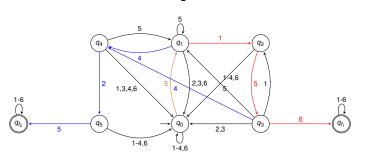
Welt &

Nord- und Ostkode integrieren



FREIBU

Zusätzlich muss der Würfel noch 5425 als Teilstring erkennen! Der Automat ändert sich wie folgt:



Deterministische endliche Automaten

Beispiel Formale

Grundlagen Verhalten eine

Teilwort-Erkennung

Transdukto-

Modellierung von Automaten

Welt & Modell



sche endliche Automaten

Transduktoren

Moore-Automat Python-Skript für Beispiel

> Modellierung von Automaten

Welt & Modell

Zusammen-Ausblick

Transduktoren

Nach der Akzeptanz ist vor der Akzeptanz!



 Der bisher konstruierte Automat akzeptiert alle Wörter, die 5156 oder 5425 als Teilwort enthalten. Determinis sche

sche endliche Automaten

Transduktoren

Moore-Automat
Umsetzung
Python-Skript fü
Beispiel

Modellieru

von Automaten

Automate Welt &

Modell
Zusammen-



Determini

- Der bisher konstruierte Automat akzeptiert alle Wörter, die 5156 oder 5425 als Teilwort enthalten.
- Eigentlich ist eine Maschine gesucht, die "ewig" läuft und die jeweils nach einem akzeptierten Teilwort eine Ausgabe macht.

Determinis sche endliche Automaten

Transduktoren

Moore-Automa

Umsetzung
Python-Skript fi

Modellierun von

Automaten

Welt & Modell

- Der bisher konstruierte Automat akzeptiert alle Wörter, die 5156 oder 5425 als Teilwort enthalten.
- Eigentlich ist eine Maschine gesucht, die "ewig" läuft und die jeweils nach einem akzeptierten Teilwort eine Ausgabe macht.
- Eine solche Maschine heißt Transduktor, das ist ein Automat, der auch Ausgaben macht.

Determinis sche endliche Automaten

Transduktoren

Moore-Automat Umsetzung

Python-Skript f Beispiel

VON

Automaten

Welt &



- Der bisher konstruierte Automat akzeptiert alle Wörter, die 5156 oder 5425 als Teilwort enthalten.
- Eigentlich ist eine Maschine gesucht, die "ewig" läuft und die jeweils nach einem akzeptierten Teilwort eine Ausgabe macht.
- Eine solche Maschine heißt Transduktor, das ist ein Automat, der auch Ausgaben macht.
- Transduktoren werden oft verwendet um das Verhalten eingebetteter Systeme zu beschreiben.

Moore-Automaten



```
Determin
```

```
@dataclass(frozen=True)
class MooreAutomaton[Q,
                            # Menge von Zuständen
                             # Eingabealphabet
                    A]:
                             # Ausgabealphabet
 dl : Callable[[Q, E], Q]
                             # Transitionsfunktion
 lm : Callable[[Q], A]
                            # Ausqabefunktion
                             # Startzustand
 st: Q
 def translate (self, input: Iterable[E]) -> Iterator[A]:
   state = self st
   yield self.lm(state)
   for c in input:
      state = self.dl(state. c)
     yield self.lm(state)
   return
```

Determinist sche endliche Automaten

Transduktoren

Moore-Automat Umsetzung

Umsetzung
Python-Skript f
Beispiel

Modellierung von Automaten

Welt & Modell

Verhalten des Moore-Automaten



■ Im Startzustand q_0 gibt der Automat $\lambda(q_0)$ aus.

Deterministische endliche Automaten

Transduktoren

Moore-Automat

Umsetzuna Beispiel

Modellierung von

Automaten

Welt & Modell

Zusammen-Ausblick

Verhalten des Moore-Automaten





- Im Startzustand q_0 gibt der Automat $\lambda(q_0)$ aus.
- Der Automat erhält ein Wort $w = "a_1 a_2 \dots"$ als Eingabe.

Determinist sche endliche Automaten

Transduktoren

Moore-Automat

Umsetzung Python-Skript f Beispiel

Modellierun von Automaten

Automate

Modell Zusammen-



REIBURG

- Im Startzustand q_0 gibt der Automat $\lambda(q_0)$ aus.
- Der Automat erhält ein Wort $w = a_1 a_2 \dots$ als Eingabe.
- Der Automat liest im Zustand q_i das Eingabesymbol a_{i+1} , wechselt in den Folgezustand $q_{i+1} = \delta(q_i, a_{i+1})$ und gibt dann $\lambda(q_{i+1})$ aus.

Determinist sche endliche Automaten

Transdukto-

Moore-Automat Umsetzung

Umsetzung
Python-Skript fi

Modellierung von Automaten

Welt & Modell



- Im Startzustand q_0 gibt der Automat $\lambda(q_0)$ aus.
- Der Automat erhält ein Wort $w = "a_1 a_2 \dots"$ als Eingabe.
- Der Automat liest im Zustand q_i das Eingabesymbol a_{i+1} , wechselt in den Folgezustand $q_{i+1} = \delta(q_i, a_{i+1})$ und gibt dann $\lambda(q_{i+1})$ aus.
- Das macht der Automat, so lange wie Eingabezeichen verfügbar sind. (Ggf. mit offenem Ende.)

sche Automaten

Transdukto-

Moore-Automat

Automaten

Welt &

fassung & Ausblick

Verhalten des Moore-Automaten



Determinis

sche endliche Automaten

Transduktoren

Moore-Automat Umsetzung

Umsetzung Python-Skript fi Beispiel

Modellierung von Automaten

Welt &

Zusammer fassung & Ausblick

Im Startzustand q_0 gibt der Automat $\lambda(q_0)$ aus.

■ Der Automat erhält ein Wort $w = a_1 a_2 \dots$ als Eingabe.

- Der Automat liest im Zustand q_i das Eingabesymbol a_{i+1} , wechselt in den Folgezustand $q_{i+1} = \delta(q_i, a_{i+1})$ und gibt dann $\lambda(q_{i+1})$ aus.
- Das macht der Automat, so lange wie Eingabezeichen verfügbar sind. (Ggf. mit offenem Ende.)
- Dabei wird das Eingabewort w in das Ausgabewort $\lambda(q_0)\lambda(q_1)\dots$ übersetzt. Daher auch der Name Transduktor.

Beispiel: Ein hypothetische Motorsteuerung



 $E = \{e, a, g, b, n\}$, wobei e für "ein", a für "aus", g für "Gas geben", b für "bremsen", n für "nicht drehende Räder" steht. $A = \{ off, low, full \}.$

sche endliche Automaten

> Transduktoren

Moore-Automat

Umsetzuna Beispiel

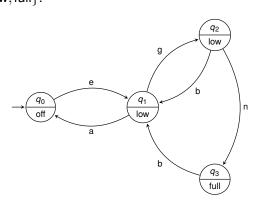
Automaten

Welt & Modell

Beispiel: Ein hypothetische Motorsteuerung



 $E = \{e, a, g, b, n\}$, wobei e für "ein", a für "aus", g für "Gas geben", b für "bremsen", n für "nicht drehende Räder" steht. $A = \{\text{off,low,full}\}.$



Determinist sche endliche Automaten

Transduktoren

Moore-Automat

Umsetzung Python-Skript

Modellierun von

Automaten
Welt &
Modell

Implementierung der Motorsteuerung

class MState (Enum):



```
Determin
```

Deterministi sche endliche Automaten

Transduktoren

Moore-Automat

Umsetzung
Python-Skript f

Modellierun von

Automaten
Welt &

Modell
Zusamme
fassung &
Ausblick

```
Q0 = auto(): Q1 = auto(): Q2 = auto(): Q3 = auto()
type Input = Literal['e','g','a','b','n']
type Output = Literal['off','low','full']
def delta_m (q : MState, c : Input) -> MState:
 match (q, c):
    case (MState.QO, "e"): return MState.Q1
    case (MState.Q1, "g"): return MState.Q2
    case (MState.Q1, "a"): return MState.Q0
    case (MState.Q2, "b"): return MState.Q1
    case (MState.Q2, "n"): return MState.Q3
    case (MState.Q3, "b"): return MState.Q1
    case :
                          return q
def lambda m (q : MState) -> Output:
 match q:
    case MState.Q0:
                    return "off"
    case MState.Q1 | MState.Q2: return "low"
                             return "full"
    case MState.03:
```

Implementierung der Motorsteuerung (2)



```
mcontrol : MooreAutomaton[MState, Input, Output] = MooreAutomaton(
 dl= delta m.
 lm= lambda m,
  st= MState.Q0
Ein Testlauf:
output = mcontrol.translate ("egbgnba")
print (list (output))
Ausgabe:
['off', 'low', 'low', 'low', 'full', 'low', 'off']
```

sche endliche Automaten

Transduktoren

Moore-Automat

Umsetzuna Beispiel

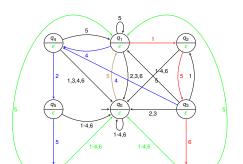
von Automaten

Welt & Modell

Ausblick

Beispiel: Der Würfel-Moore-Automat

Sei $A = \{n, o, \varepsilon\}$, dann könnte der Würfelautomat so ausschauen (die grünen Teile sind neu):



Deterministische endliche Automaten

Transduktoren

Moore-Automat

Umsetzung Python-Skript f Beispiel

Modellierung von Automaten

Welt &

Die Innereien des Würfels





■ Wie sieht die Implementierung eines solchen abstrakten Automaten aus?

Deterministische endliche Automaten

Transdukto-

Moore-Automat Umsetzung

Python-Skript fü Beispiel

Modellierung von

Automaten

Welt &

Die Innereien des Würfels

- Deterministische endliche Automaten
- Transduktoren
- Moore-Automat Umsetzuna
- Beispiel
- Modellierung Automaten
- Welt & Modell
- Zusammenfassung & Ausblick

- Wie sieht die Implementierung eines solchen abstrakten Automaten aus?
- Das Innere des Würfels:



Die Innereien des Würfels

- - - Determinist sche endliche Automaten
 - Transdukto-
 - Moore-Automat Umsetzung
 - Python-Skript fi Beispiel
 - Modellierur
 - von Automaten
 - Welt &
 - Modell
 Zusammenfassung &
 Ausblick

- BURG
- Wie sieht die Implementierung eines solchen abstrakten Automaten aus?
- Das Innere des Würfels:



■ Batterien (4×AA-Akkus, also 4,5-6 Volt),

Die Innereien des Würfels

- - - Determinist sche endliche Automaten
 - Transduktoren
 - Umsetzung
 - Python-Skript fü Beispiel
 - Modellierung von
 - Automaten
 Welt &
 - Modell Zusammen
 - Zusammenfassung & Ausblick

- Wie sieht die Implementierung eines solchen abstrakten Automaten aus?
- Das Innere des Würfels:



- Batterien (4×AA-Akkus, also 4,5-6 Volt),
- Servomotor,

Die Innereien des Würfels

sche endliche Automaten

Transdukto-

Moore-Automat Umsetzuna

Beispiel

Automaten Welt &

fassung & Ausblick

- Wie sieht die Implementierung eines solchen abstrakten Automaten aus?
- Das Innere des Würfels:



- Batterien (4×AA-Akkus, also 4,5-6 Volt),
- Servomotor,
- pyboard (mit einem ARM-5 Prozessor, Beschleunigungsmesser, usw.), auf dem Micropython läuft



■ side_up():

Bestimmt mit Hilfe des Beschleunigsmessers, welche Seite oben liegt. Bei unklaren Werten wartet die Funktion, bis eine stabile Lage eingetreten ist.

Determinist sche endliche Automaten

Transdukto-

Moore-Automat Umsetzung

Python-Skript für Beispiel

Modellierui von

Automaten

Welt &

Transdukto-

Umsetzuna

Automaten Welt &

Modell

fassung & Ausblick

- side up(): Bestimmt mit Hilfe des Beschleunigsmessers, welche Seite oben liegt. Bei unklaren Werten wartet die Funktion. bis eine stabile Lage eingetreten ist.
- new input(): Erzeugt ein neues Eingabesymbol (Zahl zwischen 1 und 6), wenn der Würfel 500 Millisekunden stabil lag.



■ side_up():

Bestimmt mit Hilfe des Beschleunigsmessers, welche Seite oben liegt. Bei unklaren Werten wartet die Funktion, bis eine stabile Lage eingetreten ist.

new_input():Erzeugt ein neues Eingabesymbol (Zahl zwischen 1 und 6), wenn der Würfel 500 Millisekunden stabil lag.

■ next_state(state, input):
Die Transitionsfunktion

Determinis sche endliche Automaten

Transduktoren

Moore-Automat Umsetzung

Python-Skript für Beispiel

Modellierun

Automaten
Welt &

Modell Zusammer



NE NE

■ side_up():

Bestimmt mit Hilfe des Beschleunigsmessers, welche Seite oben liegt. Bei unklaren Werten wartet die Funktion, bis eine stabile Lage eingetreten ist.

new_input():Erzeugt ein neues Eingabesymbol (Zahl zwischen 1 und 6), wenn der Würfel 500 Millisekunden stabil lag.

■ next_state(state, input):
Die Transitionsfunktion.

output_symbol(state): Berechnet das zum Zustand gehörige Ausgabesymbol. Determinis sche endliche Automaten

Transduktoren

Umsetzung

Python-Skript für Beispiel

Modellierung von Automaten

Welt &

Modell



A

■ side_up():

Bestimmt mit Hilfe des Beschleunigsmessers, welche Seite oben liegt. Bei unklaren Werten wartet die Funktion, bis eine stabile Lage eingetreten ist.

- new_input():
 Erzeugt ein neues Eingabesymbol (Zahl zwischen 1 und 6), wenn der Würfel 500 Millisekunden stabil lag.
- next_state(state, input):
 Die Transitionsfunktion.
- output_symbol(state): Berechnet das zum Zustand gehörige Ausgabesymbol.
- automaton():Die Endlosschleife zur Ausführung des Automaten.

Determinis sche endliche Automaten

Transduktoren

Umsetzung

Python-Skript fü Beispiel

Modellierung von Automaten

Welt &



24 / 44

N N N

■ side_up():

Bestimmt mit Hilfe des Beschleunigsmessers, welche Seite oben liegt. Bei unklaren Werten wartet die Funktion, bis eine stabile Lage eingetreten ist.

new_input():
 Erzeugt ein neues Eingabesymbol (Zahl zwischen 1 und 6), wenn der Würfel 500 Millisekunden stabil lag.

■ next_state(state, input):
Die Transitionsfunktion.

- output_symbol(state): Berechnet das zum Zustand gehörige Ausgabesymbol.
- automaton():Die Endlosschleife zur Ausführung des Automaten.
- code_knock(code):
 Klopft entsprechend dem angeforderten Code.

Determinis sche endliche Automaten

Transdukto-

Moore-Automat Umsetzung

Python-Skript für Beispiel

Modellierung von Automaten

Welt &

Der Seitenerkenner mittels Beschleunigungssensor



REBU

Die Erdbeschleunigung von 1g entspricht einem Messwert von rund 20.

```
Seitenerkenner
```

```
thres = 12
def side_up():
    while True:
        x = acc.x(); y = acc.y(); z = acc.z()
        if x > thres: return 5 #x up
        if x < -thres: return 2 #x down
        if y > thres: return 6 #y up
        if y < -thres: return 1 #y down
        if z > thres: return 3 #z up
        if z < -thres: return 4 #z down
        # no stable situation yet</pre>
```

Determinist sche endliche Automaten

Transduktoren

Moore-Automat Umsetzung Python-Skript für Beispiel

Modellierun von Automaten

Welt & Modell

Symbolerzeugung

Erzeugt ca. alle 0,5 Sekunden ein neues Eingabesymbol. Nicht nur, wenn die Seite gewechselt wird. Daher muss der Automat etwas anders aussehen!

Die Transitionsfunktion



Transitionsfunktion

```
def next state(state, input):
    if state == 0: # intial state
        if input == 5: return 1
        return 0
    elif state == 1: # '5' read
        if input == 5: return 1
        if input == 1: return 2
        if input == 4: return 4
        return 0
    elif state == 2: # '51' read
        if input == 1: return 2 # repetition!
        if input == 5: return 3
        return 0
    elif ...
```

Beachte: Jeder Zustand hat eine Schleife für das Zeichen, das dafür notwendig war, in den Zustand zu kommen.

Determinist sche endliche Automaten

Transduktoren

Moore-Automat
Umsetzung
Python-Skript für
Beispiel

Modellierung von

Automaten
Welt &

Modell Zusamme



REIBU

Der Automat & die Ausgabefunktion

```
def automaton():
    state = 0
    while True:
        if sw(): return # if switch is pressed, exit
        state = next state(state, new input())
        code knock(output symbol(state))
def output_symbol(state):
    if state == 10:
        return "north"
    elif state == 11:
        return "east"
    else:
        return None
```

Determinist sche endliche Automaten

> Transduktoren Moore-Automat

Umsetzung
Python-Skript für
Beispiel

Modellierun von Automaten

Welt &



Modellierung von Automaten

Deterministische endliche Automaten

Transduktoren

Modellierung von Automaten

Welt & Modell



#

- Im Vergleich zu Standard-Python hat Micropython einige Einschränkungen.
- Die Modellierung der Zustände durch Zahlen ist kompakt und effizient, aber fehleranfällig.
- Besserer Ansatz: Verwende Aufzählungstypen.

Determinist sche endliche Automaten

Transduktoren

Modellierung von Automaten

Welt & Modell

Zustandstyp



```
ZHZ
```

```
from enum import Enum, auto

class State(Enum):
    INIT = auto()
    AFTER_5 = auto()
    AFTER_51 = auto()
    AFTER_515 = auto()
    AFTER_5156 = auto()
    AFTER_54 = auto()
    AFTER_542 = auto()
    AFTER_5425 = auto()
```

Determinist sche endliche Automaten

Transduktoren

Modellierung von Automaten

Welt &

- Bessere Dokumentation für die Zustände.
- enum.auto() erzeugt automatisch einen neuen internen Code, sodass die Zustände durchnummeriert werden.

Transitionsfunktion



```
S = State
def next state(q: State, i: int) -> State:
   match (q, i):
       case (S.INIT, 5): return S.AFTER 5
       case (S.AFTER 5, 1): return S.AFTER 51
       case (S.AFTER 5, 4): return S.AFTER 54
       case (S.AFTER 5, 5): return S.AFTER 5
       case (S.AFTER 51, 5): return S.AFTER 515
       case (S.AFTER 515, 6): return S.AFTER 5156
       case (S.AFTER 515, 1): return S.AFTER 51
       case (S.AFTER_515, 4): return S.AFTER_54
       case (S.AFTER 54, 2): return S.AFTER 542
       case (S.AFTER 542, 5): return S.AFTER 5425
```

■ Hier ist ein Fehler!

return S.INIT

Determinist sche endliche Automaten

Transduktoren

Modellierung von Automaten

Welt & Modell



- Falls die Zustände selbst noch weitere Information beinhalten, können sie als "richtige" Objekte betrachtet werden.
- Die Zustandsfunktion wird nun als Methode implementiert.
- So ist auch die Menge der Zustände einfach erweiterbar.
- Wegen des Speicherbedarfs allerdings für eingebettete Systeme weniger geeignet.
- Das Zustandsmuster ist ein sogenanntes design pattern.
- Siehe Design Patterns: Elements of Reusable Object-Oriented Software von der "Gang of Four" (Gamma, Helm, Johnson, Vlissides).

Determinist sche endliche Automaten

Transdukto-

Modellierung von Automaten

Welt & Modell



```
Determinis
```

Deterministi sche endliche Automaten

Transduktoren

Modellierung von Automaten

Welt &

```
# base class for all states
@dataclass
class State:
    def next(self, input: int) -> 'State':
        return self
    def output(self) -> str:
        return ""
```

- Die Basisklasse State dient nur als Superklasse, von der konkrete Zustandsklassen erhen
- Sie soll nicht instanziert werden, d.h. State() soll nicht verwendet werden.
- Auch die Methoden sollen eigentlich nicht verwendet werden, sondern nur anzeigen, was in den Subklassen implementiert werden soll.
- Eine solche Klasse heißt abstrakte Klasse und die Methoden abstrakte Methoden.



```
FREIB
```

```
@dataclass
class S_Init(State):
    def next(self, input: int) -> State:
        match input:
        case 5:
        return S_After([5])
    return self
```

- Subklasse von State.
- Methode next() wird überschrieben.
- Methode output() wird aus der Superklasse übernommen (keine Ausgabe).

Determinist sche endliche Automaten

Transduktoren

Modellierung von Automaten

Welt &



Determin

sche endliche Automaten

Transduktoren

Modellierung von Automaten

Welt &

Zusammen fassung & Ausblick

@dataclass

```
class S After(State):
    prefix: list[int]
    def next(self, input: int) -> State:
        self.prefix = self.prefix + [input]
        match self.prefix:
            case ([5, 1] | [5, 1, 5] | [5, 1, 5, 6] |
                  [5, 4] \mid [5, 4, 2] \mid [5, 4, 2, 5]:
                return self
        match self.prefix[-1:]:
            case [5]:
                return S_After(self.prefix[-1:])
        match self.prefix[-2:]:
            case [5, 1] | [5, 4]:
                return S After(self.prefix[-2:])
        return S Init()
```



```
def output(self) -> str:
    match self.prefix:
        case [5, 1, 5, 6]:
        return "north"
        case [5, 4, 2, 5]:
            return "east"
        case _:
        return ""
```

- Die Klasse S_After verwaltet das bisher gelesene Präfix der Eingabe im Feld prefix.
- Zustandsübergang durch Änderung des Präfixes (soweit möglich).

Determinist sche endliche Automaten

Transdukto-

Modellierung von Automaten

Welt & Modell



Deterministische endliche Automaten

Transduktoren

Modellierung von Automaten

Welt &

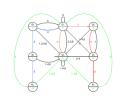
Zusammenfassung & Ausblick

Welt & Modell









Determinist sche endliche Automaten

Transduktoren

Modellierung von Automaten

Welt & Modell









Determinist sche endliche Automaten

Transduktoren

Modellierun von Automaten

Welt &

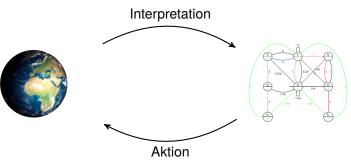
Modell

Zusammenfassung & Ausblick

Vor dem Einsatz formaler Modelle (wie Moore-Automaten) müssen die Messwerte/Eingaben interpretiert und in Symbole umgesetzt werden. Die Interpretation und das Modell beeinflussen sich dabei gegenseitig (Beispiel: Würfelseitenerkennung und Automat).







Vor dem Einsatz formaler Modelle (wie Moore-Automaten) müssen die Messwerte/Eingaben interpretiert und in Symbole umgesetzt werden. Die Interpretation und das Modell beeinflussen sich dabei gegenseitig (Beispiel: Würfelseitenerkennung und Automat).

Determinis sche endliche Automaten

Transdukto-

Modellierun von Automaten

Welt & Modell





Interpretation

Transduktoren Modellierun von

sche endliche Automaten

Automaten
Welt &

Welt & Modell

Zusammen fassung & Ausblick

Vor dem Einsatz formaler Modelle (wie Moore-Automaten) müssen die Messwerte/Eingaben interpretiert und in Symbole umgesetzt werden. Die Interpretation und das Modell beeinflussen sich dabei gegenseitig (Beispiel: Würfelseitenerkennung und Automat).

⇒ Siehe Vorlesung Embedded Systems.



Zusammenfassung & Ausblick

Deterministische endliche Automaten

Transduktoren

Modellierung von Automaten

Welt &

Zusammenfassung



■ Eine formale Sprache ist eine Menge von Wörtern.

Determinist sche endliche Automaten

> Transduktoren

Modellierung von Automaten

Welt & Modell



- Eine formale Sprache ist eine Menge von Wörtern.
- Automaten sind ein einfaches Berechnungsmodell.

Deterministische endliche Automaten

Transduktoren

Modellierung von Automaten

Welt &



Transdukto-

Modellierung von Automaten

Welt &

- Eine formale Sprache ist eine Menge von Wörtern.
- Automaten sind ein einfaches Berechnungsmodell.
- Sie können Sprachen erkennen (Akzeptoren) und übersetzen (Transduktoren).

Zusammenfassung



Z W

- Eine formale Sprache ist eine Menge von Wörtern.
- Automaten sind ein einfaches Berechnungsmodell.
- Sie können Sprachen erkennen (Akzeptoren) und übersetzen (Transduktoren).
- Determinstische endliche Automaten (DEAs) und Transduktoren (Moore-Automaten) werden zur Beschreibung von eingebetteten Systemen verwendet.

Determinis sche endliche Automaten

Transduktoren

Modellierung von Automaten

Welt & Modell

Zusammenfassung



- Determinis sche endliche
 - Automaten
 Transduktoren
 - Modellierung von Automaten
 - Welt & Modell

- Eine formale Sprache ist eine Menge von Wörtern.
- Automaten sind ein einfaches Berechnungsmodell.
- Sie k\u00f6nnen Sprachen erkennen (Akzeptoren) und \u00fcbersetzen (Transduktoren).
- Determinstische endliche Automaten (DEAs) und Transduktoren (Moore-Automaten) werden zur Beschreibung von eingebetteten Systemen verwendet.
- Einfache Implementierung.