2.6 Verdeutlichung verwendeter Begriffe

- endlich/finit: die Mengen der Zustände und der Ein- bzw. Ausgabezeichen sind endlich
- synchron: die Ausgabezeichen erscheinen synchron mit dem Einlauf der Eingabezeichen
- sequentiell: die Eingabezeichen werden seriell verarbeitet und bilden eine Eingangsfolge

3. Darstellung

3.1 Graphische Darstellung

- ein Automat kann durch den so genannten Zustandsübergangs-Graphen dargestellt werden
- die Knoten des Graphen entsprechen den Zuständen
- jede Kante des Graphen entspricht einem Arbeitsschritt
- am Beginn der Kante wird/werden das/die Eingangssymbol/e eingetragen, für welche dieser Übergang gültig ist
- bei Mealy-Automaten wird das Ausgangssymbol an der Kante angetragen, welches bei Benutzung dieses Überganges ausgegeben wird, bei Moore-Automaten wird das Symbol in den Zielknoten geschrieben
- der Startzustand wird durch einen einlaufenden Pfeil (Pfeil, der im leeren Raum beginnt) gekennzeichnet
- Endzustände werden durch einen Doppelkreis dargestellt

3.2 Tabellarische Darstellung

- die Menge der möglichen Zustände und Eingangssymbole ist endlich, alle möglichen Kombinationen von Eingangssymbolen und Zuständen lassen sich aufschreiben
- jeder solchen Kombination wird durch die Ausgabefunktion ein Ausgabesymbol und durch die Zustandsübergangsfunktion ein Folgezustand zugeordnet
- diese Abhängigkeiten werden tabellarisch dargestellt
- diese Darstellung eignet sich für maschinelle Bearbeitung natürlich besser als eine graphische Schreibweise
- ist die Zustandsübergangsfunktion nur partiell gegeben, so kann sie durch Einführung eines Fangzustandes total gemacht werden

3.3 Mathematische Darstellung

- Eingabe- und Ausgabezeichen, sowie Zustände und Endzustände werden als endliche Mengen von Symbolen geschrieben
- Startzustand ist ein Symbol aus der Menge der Zustände
- die Ausgabefunktion/-relation und die Zustandsübergangsfunktion/-relation sind mathematische Funktionen bzw. Relationen

4. Deterministische endliche Automaten

Definition 27

Ein deterministischer endlicher Automat (englisch: deterministic finite automaton, kurz DFA) wird durch ein 7-Tupel $M = (Q, \Sigma, \delta, \Gamma, g, q_0, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $q_0 \in Q$ ist der Startzustand.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to Q$ heißt Übergangsfunktion.
6. $g : Q \to \Gamma$ (bzw. $g : Q \times \Sigma \to \Gamma$) ist die Ausgabefunktion, Γ das Ausgabealphabet.

Die von M akzeptierte Sprache ist

$$L(M) := \{w \in \Sigma^*;\ \hat{\delta}(q_0, w) \in F\}\ ,$$

wobei $\hat{\delta} : Q \times \Sigma^* \to Q$ induktiv definiert ist durch

$$\begin{array}{rcll} \hat{\delta}(q,\epsilon) & = & q & \text{für alle } q \in Q \\ \hat{\delta}(q,ax) & = & \hat{\delta}(\delta(q,a),x) & \text{für alle } q \in Q, a \in \Sigma \\ & & & \qquad \text{und } x \in \Sigma^* \end{array}$$

Bemerkung: Endliche Automaten können durch (gerichtete und markierte) Zustandsgraphen (aka Cayley-Diagramme) veranschaulicht werden:

- Knoten $\widehat{=}$ Zuständen
- Kanten $\widehat{=}$ Übergängen
- genauer: die mit $a \in \Sigma$ markierte Kante (u, v) entspricht $\delta(u, a) = v$

Zusätzlich werden der Anfangszustand durch einen Pfeil, Endzustände durch doppelte Kreise gekennzeichnet.

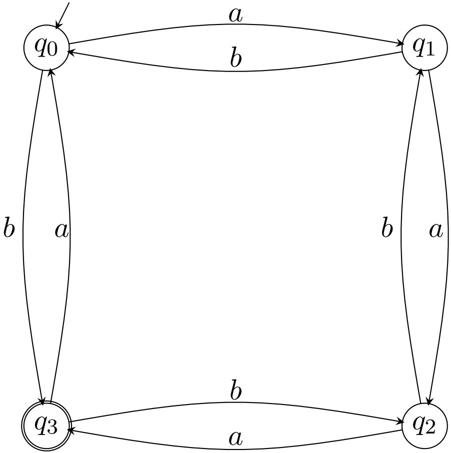
Weitere Informationen zu Arthur Cayley und zu Cayley-Diagrammen, deren ursprüngliches Anwendungsgebiet ja die algebraische Gruppentheorie ist, finden sich unter

1. Arthur Cayley (weitere Bilder von Arthur Cayley)
2. Grundbegriff des Cayley-Diagramms

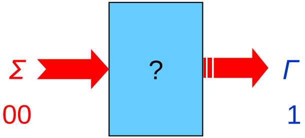
Beispiel 28 (DEA ohne Ausgabe)

Sei $M = (Q, \Sigma, \delta, q_0, F)$, wobei

$$
\begin{aligned}
Q &= \{q_0, q_1, q_2, q_3\}\\
\Sigma &= \{a, b\}\\
F &= \{q_3\}\\
\delta(q_0, a) &= q_1\\
\delta(q_0, b) &= q_3\\
\delta(q_1, a) &= q_2\\
\delta(q_1, b) &= q_0\\
\delta(q_2, a) &= q_3\\
\delta(q_2, b) &= q_1\\
\delta(q_3, a) &= q_0\\
\delta(q_3, b) &= q_2
\end{aligned}
$$



Beispiel 29



- **eingegebene Zeichenfolge ($e_i \in \Sigma$ Eingabe)**

	e_6	e_5	e_4	e_3	e_2	e_1
=	01	10	10	00	11	00

- **Ausgabezeichenfolge ($a_i \in \Gamma$ Ausgabe)**

	a_6	a_5	a_4	a_3	a_2	a_1
=	1	1	1	1	0	1

- **Eingabe und Ausgabe erfolgen synchron**

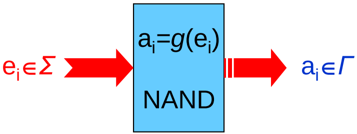
NAND-Automat

- **Endlicher Automat:** $\mathcal{A} = (Q,\Sigma,\delta,\Gamma,g,q_0,F)$
  - Eingabealphabet Σ
  - Ausgabealphabet Γ
  - Ausgabefunktion: $g: \Sigma \to \Gamma$

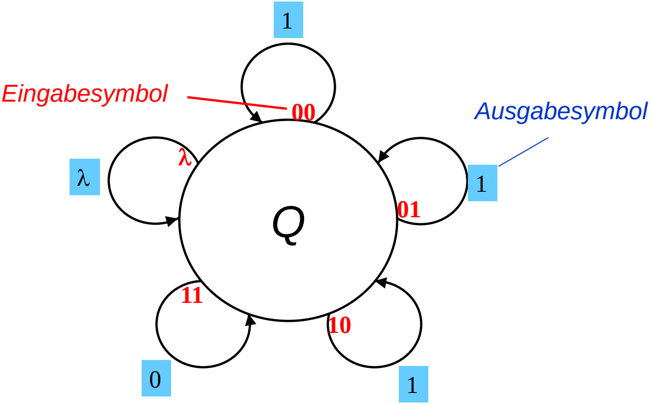
e	01	10	10	00	11	00
a	1	1	1	1	0	1

g Σ	Γ
00	1
01	1
10	1
11	0

NAND-Funktion



Cayley-Diagramm für den NAND-Automaten



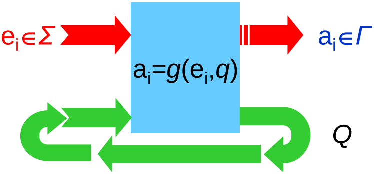
Beispiel 30

- **eingegebene Zeichenfolge ($e_i \in \Sigma$)**

...	λ	λ	e_6	e_5	e_4	e_3	e_2	e_1
= ...	λ	λ	01	10	10	11	00	00

- **Ausgabezeichenfolge ($a_i \in \Gamma$)**

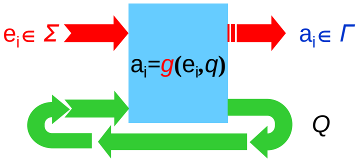
...	λ	a_7	a_6	a_5	a_4	a_3	a_2	a_1
= ...	λ	1	0	0	0	0	0	0



Bitserielle Addition

- **Automat:** $\mathcal{A} = (Q, \Sigma, \delta, \Gamma, g, q_0, F)$

Q	endliche Zustandsmenge
q_0	Anfangszustand
F	Menge der End- oder Finalzustände $F \subseteq Q$
$g : \Sigma \times Q \rightarrow \Gamma$	Ausgabefunktion
$\delta : \Sigma \times Q \rightarrow Q$	Zustandsübergangsfunktion



Funktion g
bit-serielle Addition
→ ADD-Funktion

Bitserielle Addition

- **Für den Automaten ADD gilt**

λ : leeres Zeichen

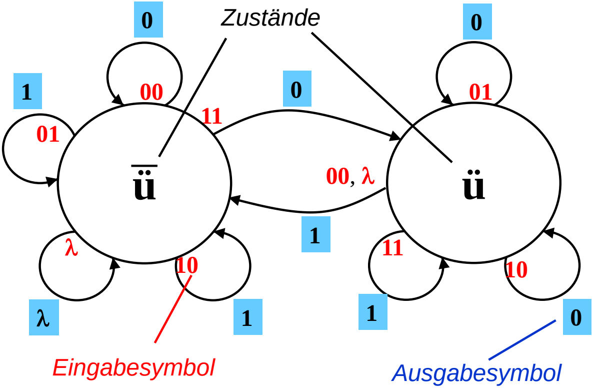
Σ	$\{00,01,10,11\} \cup \{\lambda\}$
Γ	$\{0,1\} \cup \{\lambda\}$
q_0	$\bar{ü}$
F	$\{\bar{ü}\}$
$g : \Sigma \times Q \rightarrow \Gamma$	*Ausgabefunktion*
$\delta : \Sigma \times Q \rightarrow Q$	*Zustandsübergangsfunktion*

Übertrag vorhanden (ü=1)

oder nicht (ü=0)

Bitserielle Addition

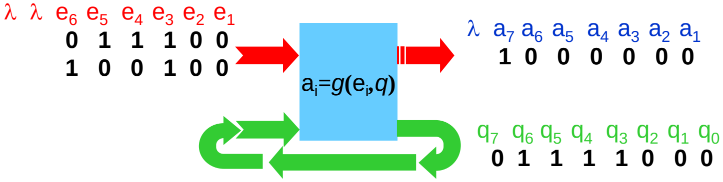
- **Cayley-Diagramm für den ADD-Automaten**



Addition binärer Zahlen

	1 1 1 1 0 0 0	*Übertrag - Zustand*
	0 1 1 1 0 0	
+	1 0 0 1 0 0	
	1 0 0 0 0 0 0	

Realisierung als endlicher Automat



$Q = \{\text{ü}, \bar{\text{ü}}\}$