

Einführung in die technische Informatik

Christopher Kruegel chris@auto.tuwien.ac.at

<http://www.auto.tuwien.ac.at/~chris>

Administratives

- 3 Vorträge zur Übung
 - Boolesche Funktionen
 - VHDL (Hardware Description Languages)
 - Schaltwerke (Moore / Mealy)
- Termine
 - Jeweils Dienstags, am 20.04.04, 27.04.04 und 04.05.04
 - 14¹⁵ - 15⁴⁵ Uhr, AudiMax
- Folien
 - verfügbar im Web, unter <http://www.auto.tuwien.ac.at/~chris>

Allgemeines (Wiederholung)

Automation Systems Group

- Was kommt zum Test?
 - Buch „Einführung in die technische Informatik“
Kapitel 1 bis *inklusive* Kapitel 4.3 (Mealy Schaltwerk)
 - Vorlesung und Material auf Homepage
 - Wissensfragen und Rechenbeispiele (Verhältnis: 60% - 40%)
- Wissensfragen
 - detailliert (!)
 - stichwortartig zu beantworten
 - Akronyme (z.B., ASIC, BCD)
 - Aufzählungen (z.B., Schaltkreisfamilien)
 - Schaltungen verstehen und reproduzieren können (!)

Schaltwerke

Automation Systems Group

- Realisieren logische Funktionen **mit Speicherwirkung**
 - Ausgang hängt vom Eingang *und* vom internen Zustand ab
 - logische Funktionen realisieren
 - Übergänge zwischen Zuständen
 - Ausgangsfunktion
 - Unterschied zu Schaltnetzen (Anhängigkeit nur vom Eingang)
 - Speicher (z.B. Flip-Flops) realisieren Zustände
 - Schaltungen (z.B. PLA, ROM) realisieren Übergangsfunktionen
- Synchron vs. asynchrone Schaltwerke
 - synchron -- Änderungen erfolgen nur bei Taktsignal
 - asynchron -- Änderungen folgen unmittelbar (auf neuen Eingang)

Schaltwerke

Automation Systems Group

- Konzept - Eingänge werden auf Ausgänge abgebildet

Eingang
→

Ausgang
→

Schaltwerke

Automation Systems Group

- Dabei wird ein interner Zustand gespeichert

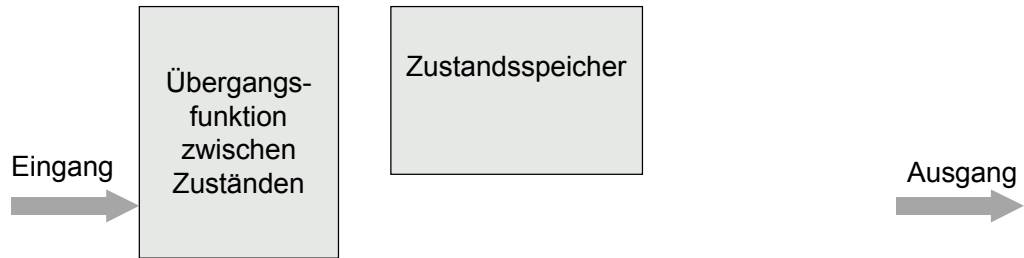
Eingang
→

Zustandsspeicher

Ausgang
→

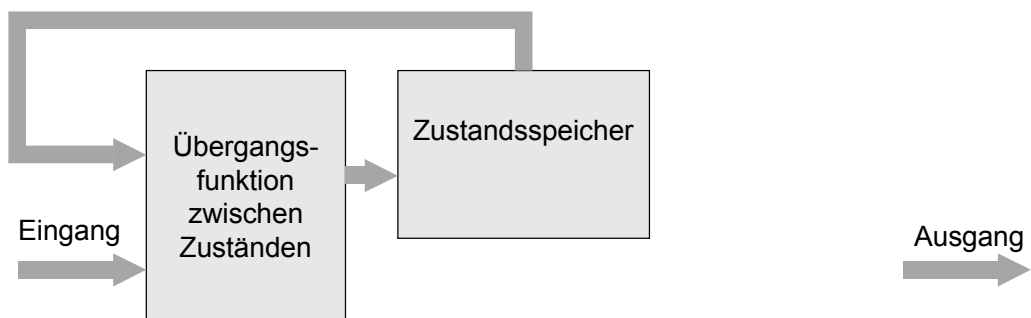
Schaltwerke

- Zustandswechsel hängen von den Eingängen ...



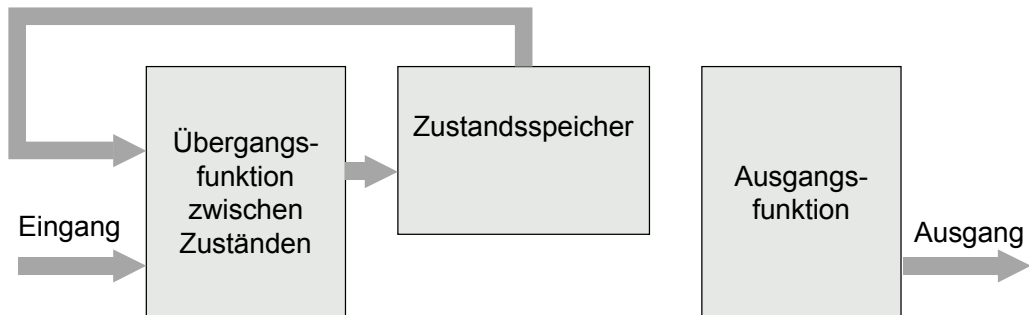
Schaltwerke

- ... und vom aktuellen Zustand ab



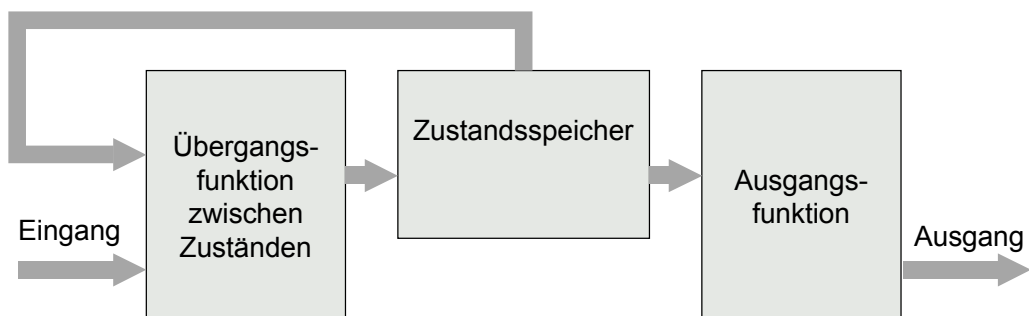
Schaltwerke

- Ausgänge werden von Ausgangsfunktion gesteuert



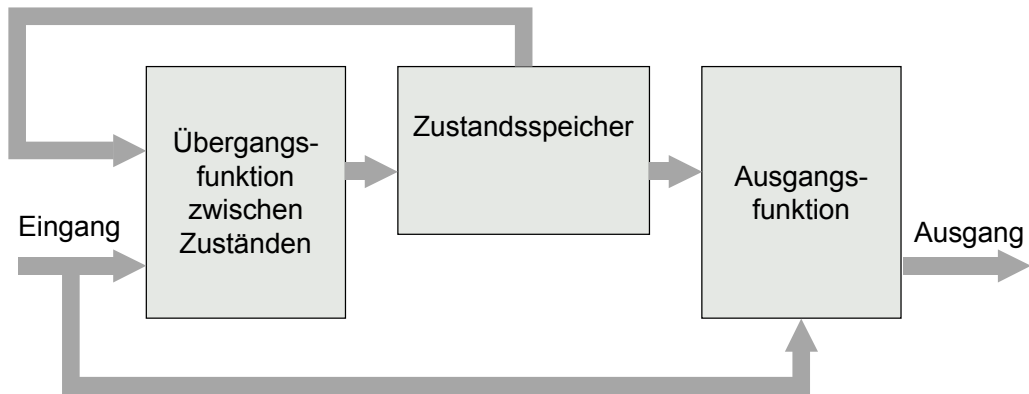
Schaltwerke

- Ausgangsfunktion hängt vom Zustand ab
→ Moore-Schaltwerk



Schaltwerke

- Ausgangsfunktion hängt vom Zustand und Eingang ab
→ Mealy-Schaltwerk

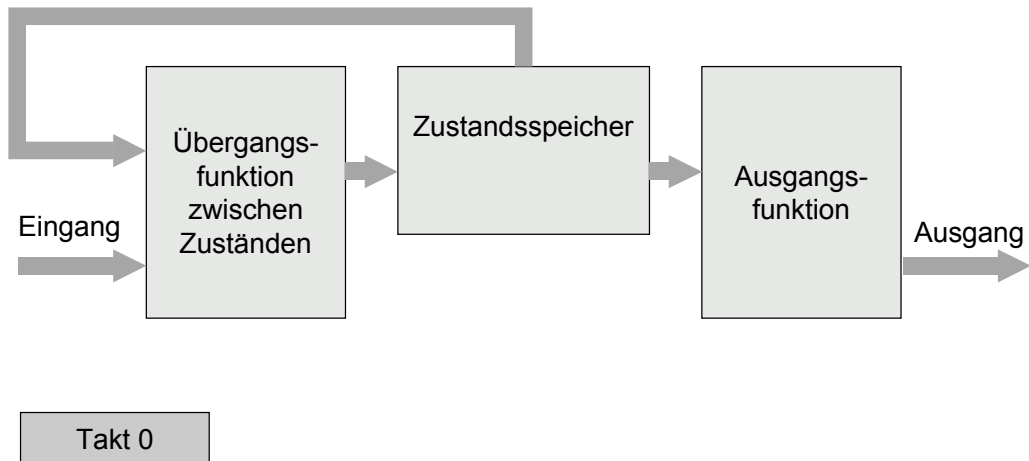


Zeitverhalten

- Zeitliches Verhalten
 - Zustandsspeicher ist synchron mit dem Takt
 - falls Eingangsänderungen zusammen mit der Taktflanke auftreten, kann es zu Problemen bei den Flip-Flops kommen
- Daher
 - Eingangssignale sind auch synchron mit dem Takt
- Konsequenz
 - Moore-Schaltwerk *kann nicht* sofort auf Eingangsänderungen reagieren
 - Mealy-Schaltwerk *kann* sofort auf Eingangsänderungen reagieren

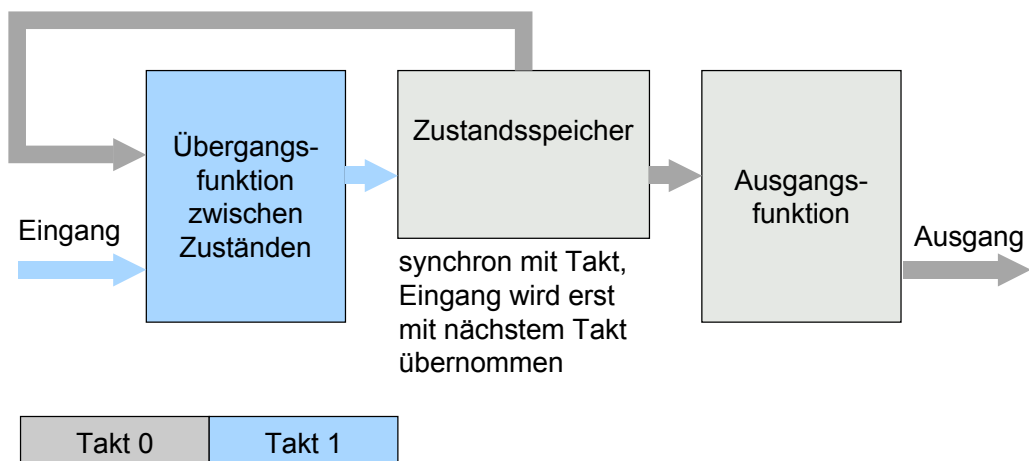
Reaktionszeit

- Moore-Schaltwerk



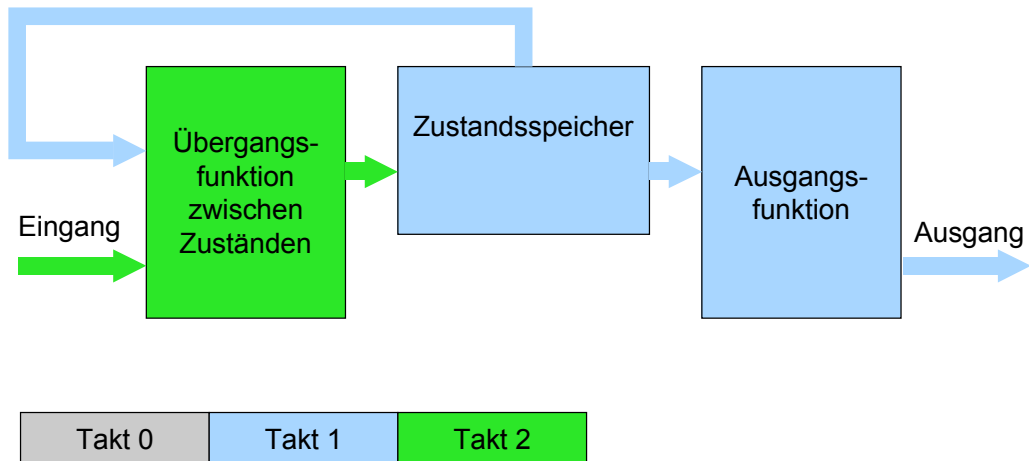
Reaktionszeit

- Moore-Schaltwerk



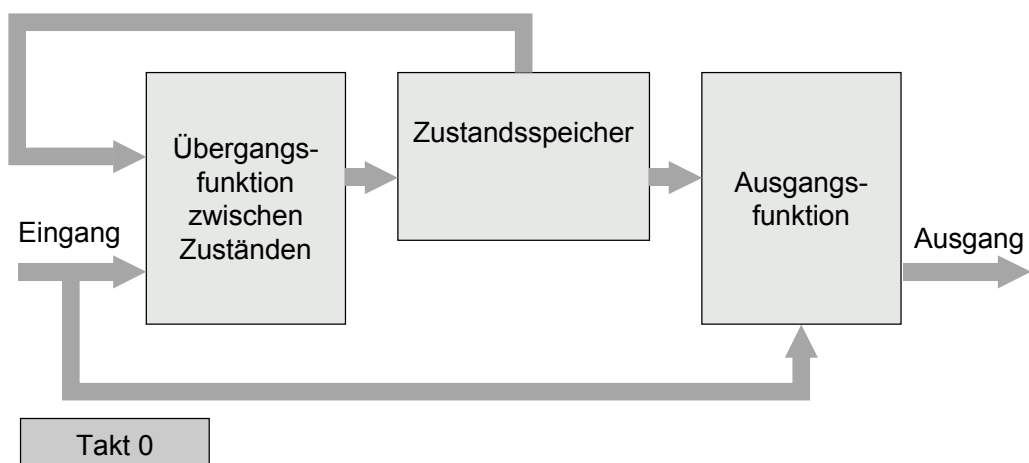
Reaktionszeit

- Moore-Schaltwerk



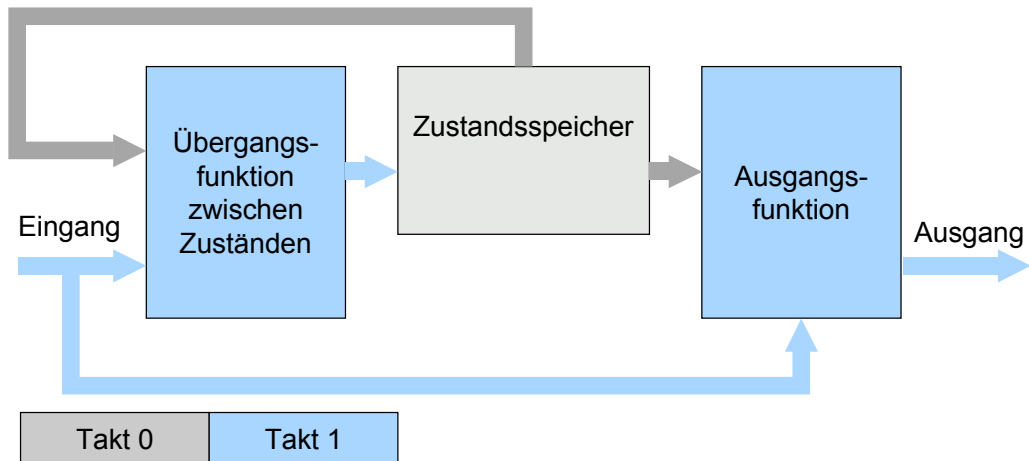
Reaktionszeit

- Mealy-Schaltwerk



Reaktionszeit

- Mealy-Schaltwerk



Takt

- Takt
 - muss langsam genug sein, damit Schaltung fehlerfrei arbeiten kann
 - maximaler Takt f_{\max} berechnet sich mittels minimalem Taktabstand

$$f_{\max} = \frac{1}{T_{\min}}$$

- Minimaler Taktabstand T_{\min}

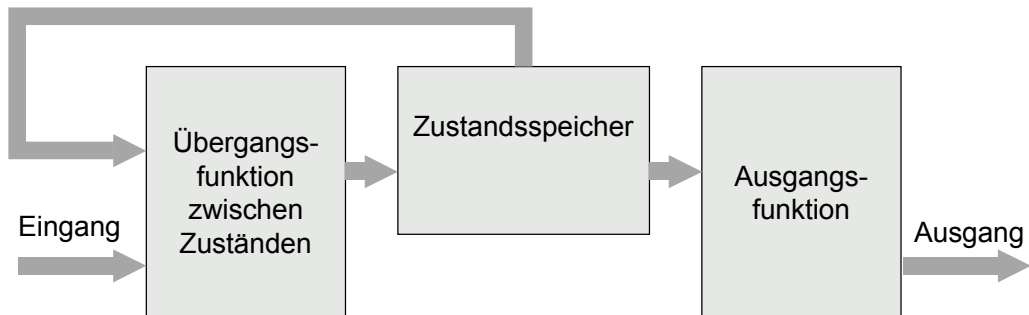
$$T_{\min} = t_{FF} + t_G + t_{Setup}$$

- Maximaler Takt f_{\max}

$$f_{\max} = \frac{1}{t_{FF} + t_G + t_{Setup}}$$

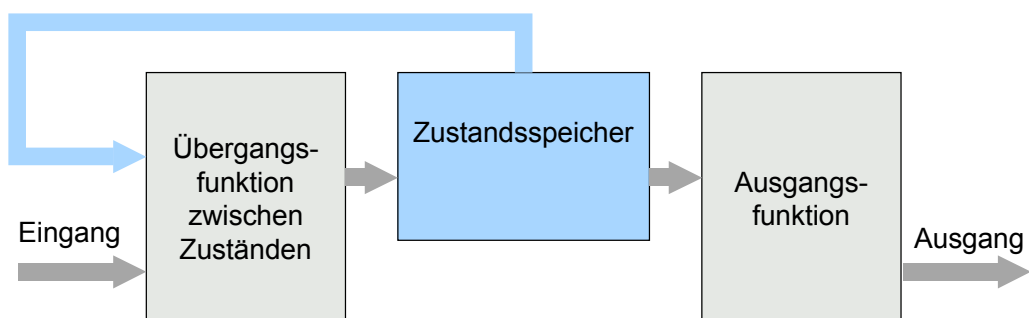
Minimaler Taktabstand

- Minimaler Taktabstand T_{\min}



Minimaler Taktabstand

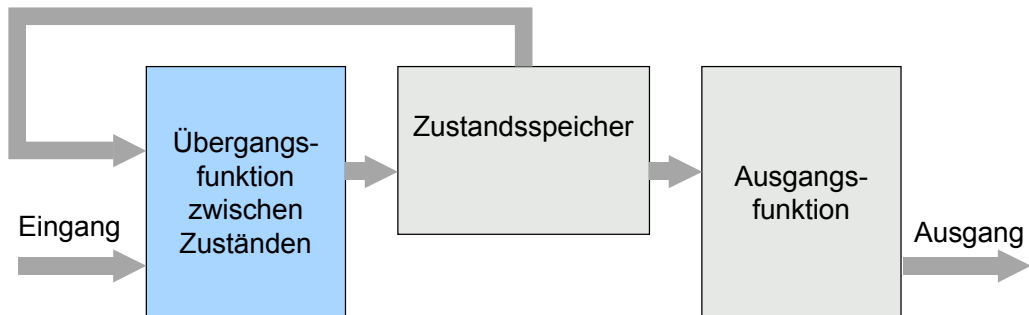
- Minimaler Taktabstand $T_{\min} = t_{FF}$



t_{FF} = Verzögerungszeit der Flip-Flops

Minimaler Taktabstand

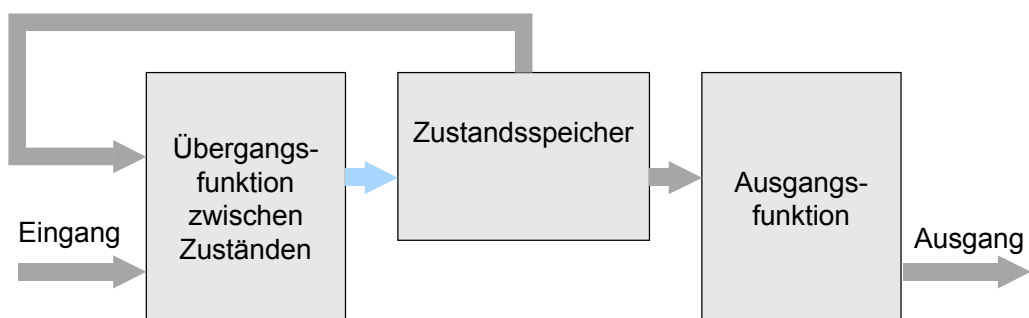
- Minimaler Taktabstand $T_{\min} = t_{FF} + t_G$



t_G = Durchlaufzeit der Übergangsfunktion

Minimaler Taktabstand

- Minimaler Taktabstand $T_{\min} = t_{FF} + t_G + t_{\text{Setup}}$



t_{Setup} = Vorbereitungszeit der Flip-Flops

Schaltwerke

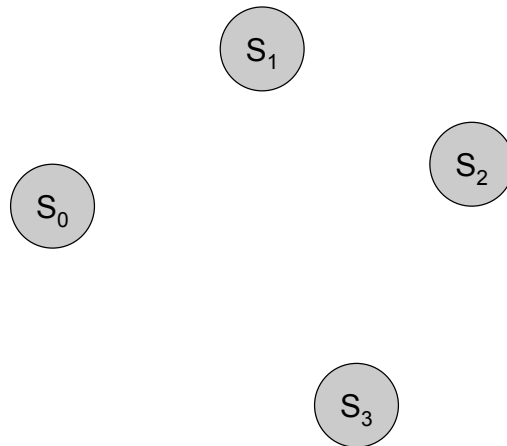
- Schritte zur Realisierung eines Schaltwerks
 - Verstehen der Aufgabenstellung
 - Aufbau des passenden Zustandsgraphen
 - Festlegen der Zustandskodierung
 - Übergangsfunktion bestimmen
 - Ausgabefunktion bestimmen
 - Berechnen der Taktfrequenz

Zustandsgraph

- Systeme mit Zustand können praktisch durch Zustandsgraphen beschrieben werden
- Zustandsgraph
 - Automaten
 - Knoten (Zustände)
 - Kanten (Übergänge zwischen Zuständen)
 - mit *endlicher* Anzahl von Zuständen und
 - *eindeutigem* Nachfolgezustand für jede Eingabe
 - deterministischer, endlicher Automat
 - = DFA (deterministic, finite automaton)

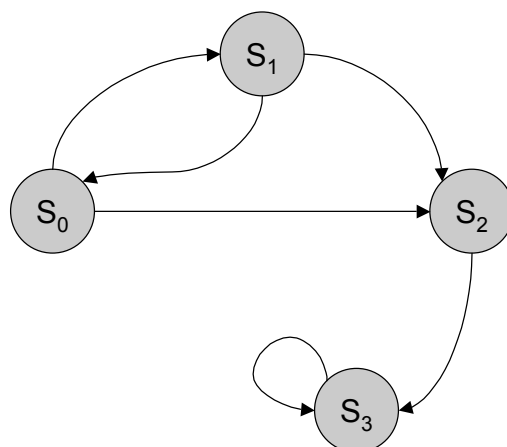
Zustandsgraph

Knoten für Zustände



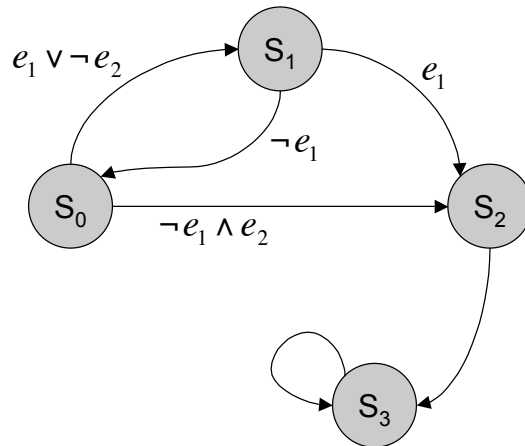
Zustandsgraph

Kanten für Zustandsübergänge



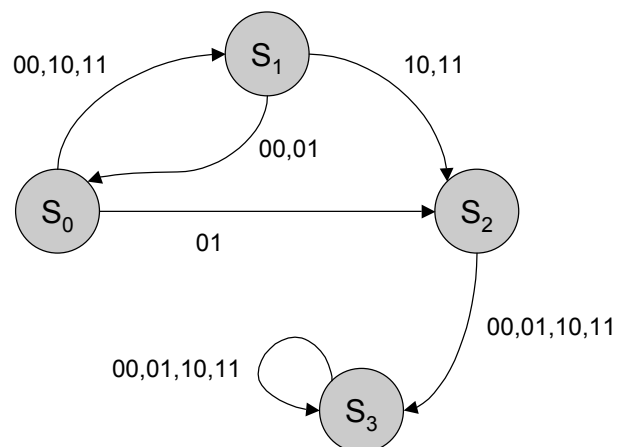
Zustandsgraph

Übergänge sind auch abhängig vom Eingang (hier e_1 und e_2)



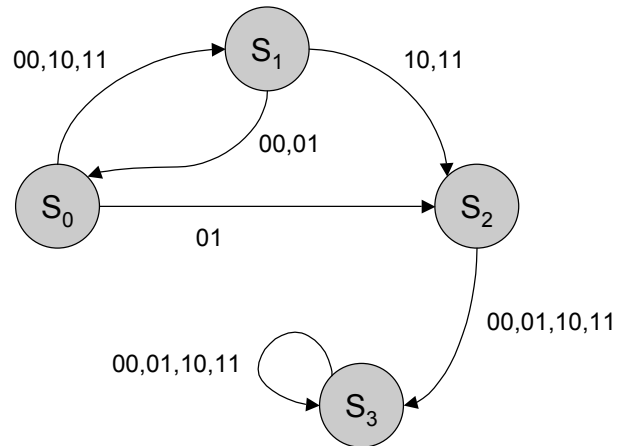
Zustandsgraph

Alternative Darstellung der Eingangswerte durch Bitmuster



Zustandsgraph

Alternative Darstellung der Eingangswerte durch Bitmuster



Reihenfolge der Bitmuster muss angegeben werden!

Reihenfolge: $e_1 e_2$

$e_1 \vee \neg e_2$ 00,10,11

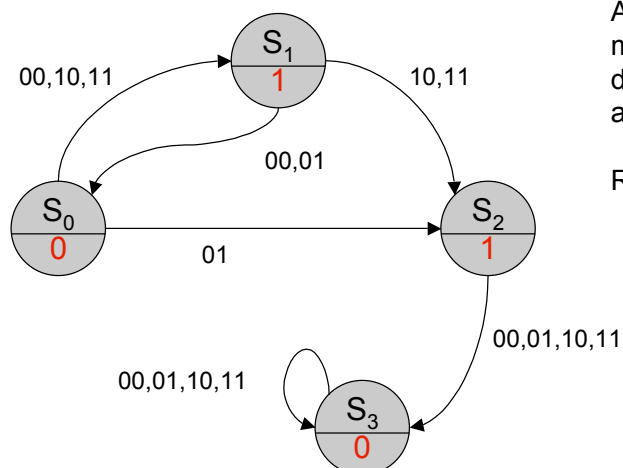
$\neg e_1 \wedge e_2$ 00

e_1 10,11

$\neg e_1$ 00,01

Zustandsgraph

Ausgabe kann *nur* vom Zustand abhängen (Moore-Schaltwerk)

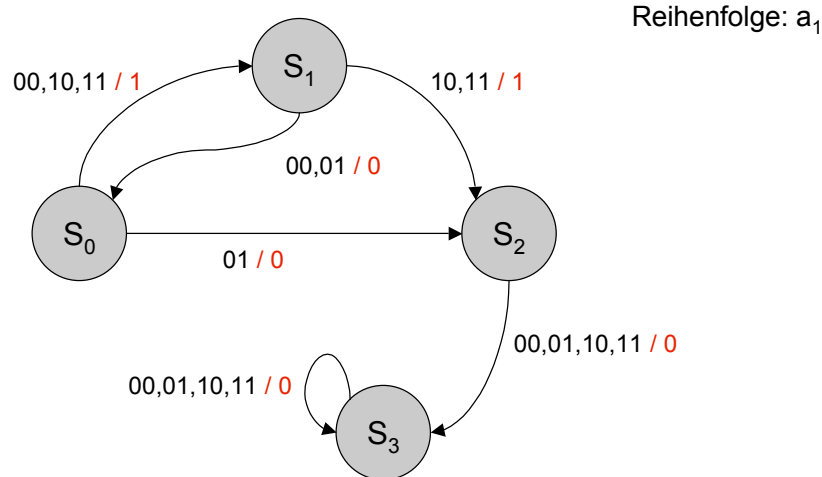


Auch bei der Ausgabe muss die Reihenfolge der Bitmuster angegeben werden!

Reihenfolge: a_1

Zustandsgraph

Ausgabe kann vom Zustand *und* Eingang abhängen (Mealy-Schaltwerk)



Zustandsgraph

- Notwendige Bedingungen für Korrektheit
 - Vollständigkeit
aus jedem Zustand sind Übergänge für alle Eingangskombinationen vorhanden
 - Eindeutigkeit
aus jedem Zustand gibt es nur einen Übergang für eine bestimmte Eingangskombinationen
- Notwendige Bedingung natürlich nicht hinreichend

Zustandskodierung

- Abbildung der Zustandsknoten auf Speicher
 - jedem Zustand muss ein Bitmuster der Flip-Flop Ausgänge zugeordnet werden
- Möglichkeiten
 - 1-aus-n Codierung
 - n Flip-Flops für n Zustände
 - immer genau ein Flip-Flop hat den Wert 1
 - Dichte Codierung
 - minimale Anzahl an Flip-Flops
 - f Flip-Flops für n Zustände
 - $2^f \geq n$

Zustandskodierung

- Beispiel mit 4 Zuständen S_0 bis S_3

- 1-aus-n Codierung
 - 4 Flip-Flops (F_0 bis F_3)

	F_0	F_1	F_2	F_3
S_0	1	0	0	0
S_1	0	1	0	0
S_2	0	0	1	0
S_3	0	0	0	1

- Dichte Codierung
 - 2 Flip-Flops (F_0 bis F_1)

	F_0	F_1
S_0	0	0
S_1	0	1
S_2	1	0
S_3	1	1

Übergangsfunktion

- Übergangsfunktion
 - logische Funktion, die Eingangswerte und Zustand auf einen neuen Zustand abbildet
 - für Moore-Schaltwerk und Mealy-Schaltwerk gleich
- Darstellung als Wahrheitstabellen
- Beispiel von vorhin, mit dichter Codierung
 - 2 Flip-Flops (F_0 und F_1) und 2 Eingänge (e_1 und e_2)

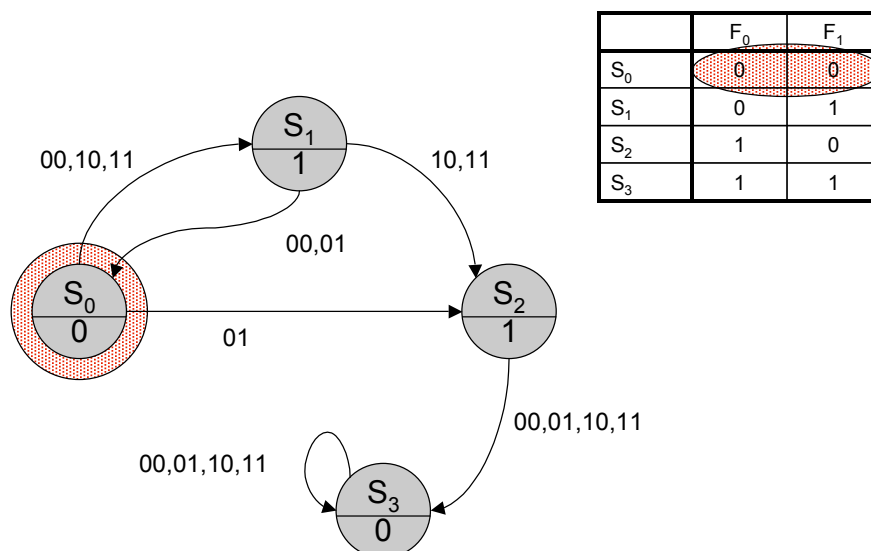
Übergangsfunktion

	S_0				S_1				S_2				S_3			
F_0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F_1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e_1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e_2	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F_0'																
F_1'																

Übergangsfunktion

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '																
F ₁ '																

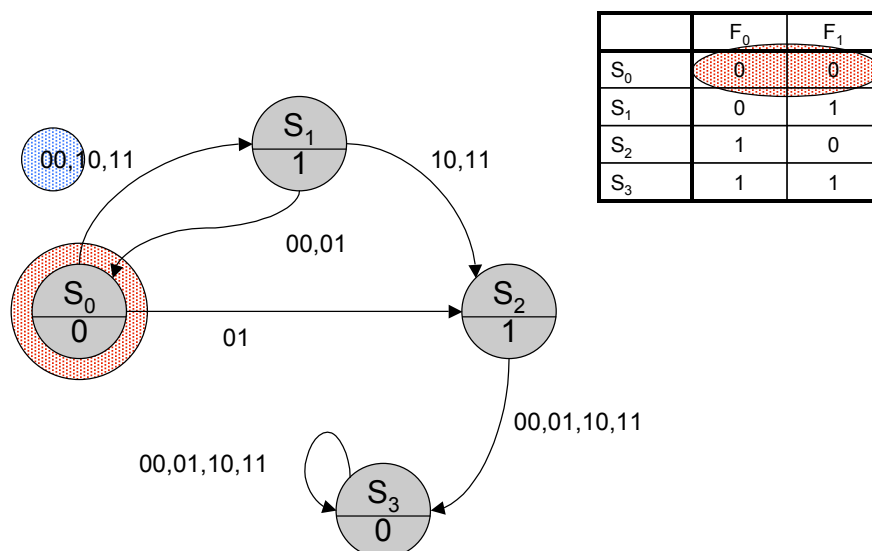
Übergangsfunktion



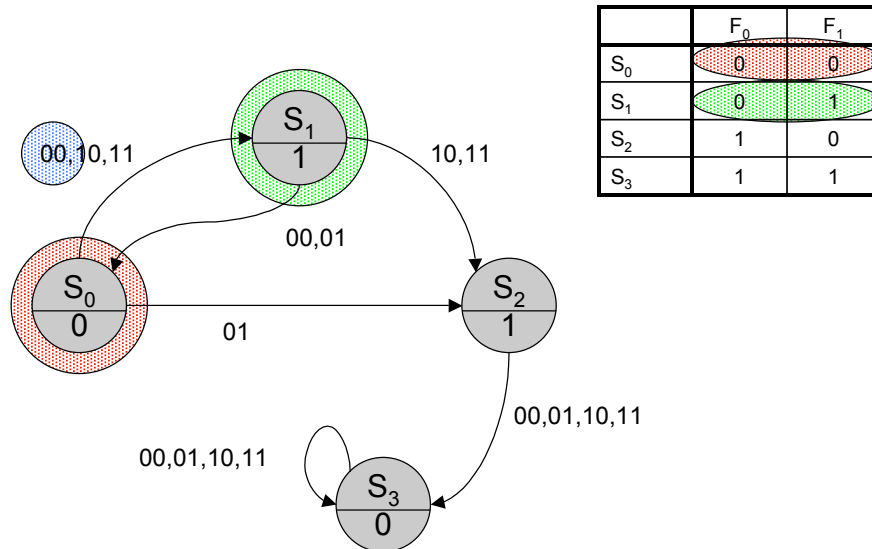
Übergangsfunktion

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '																
F ₁ '																

Übergangsfunktion



Übergangsfunktion



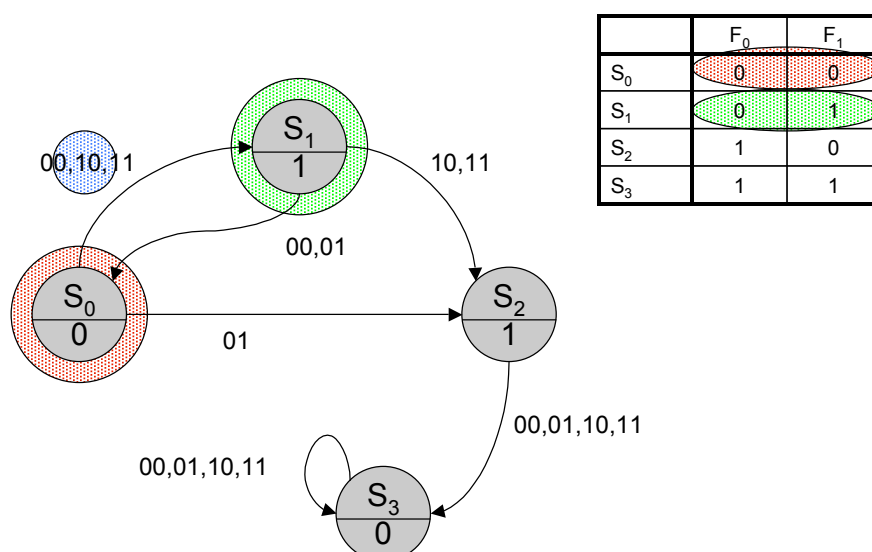
Übergangsfunktion

	S_0				S_1				S_2				S_3			
F_0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F_1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e_1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e_2	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F_0'	0															
F_1'	1															

Übergangsfunktion

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0															
F ₁ '	1															

Übergangsfunktion



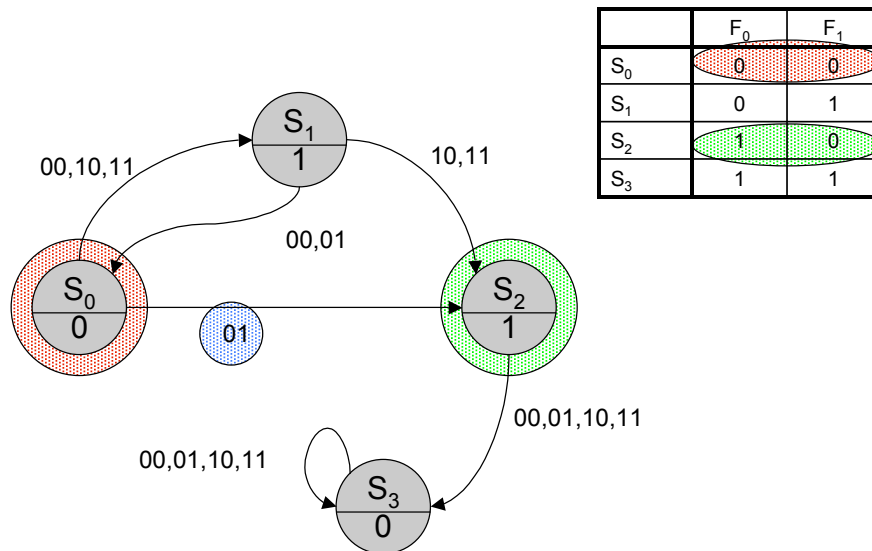
Übergangsfunktion

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0														
F ₁ '	1	1														

Übergangsfunktion

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0														
F ₁ '	1	1														

Übergangsfunktion



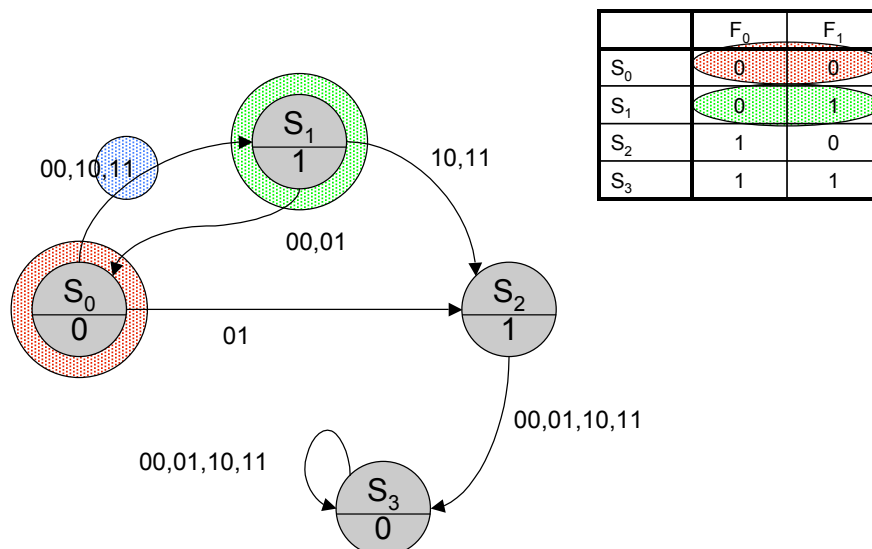
Übergangsfunktion

	S_0				S_1				S_2				S_3			
F_0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F_1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e_1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e_2	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F_0'	0	0	1													
F_1'	1	1	0													

Übergangsfunktion

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0	1													
F ₁ '	1	1	0													

Übergangsfunktion



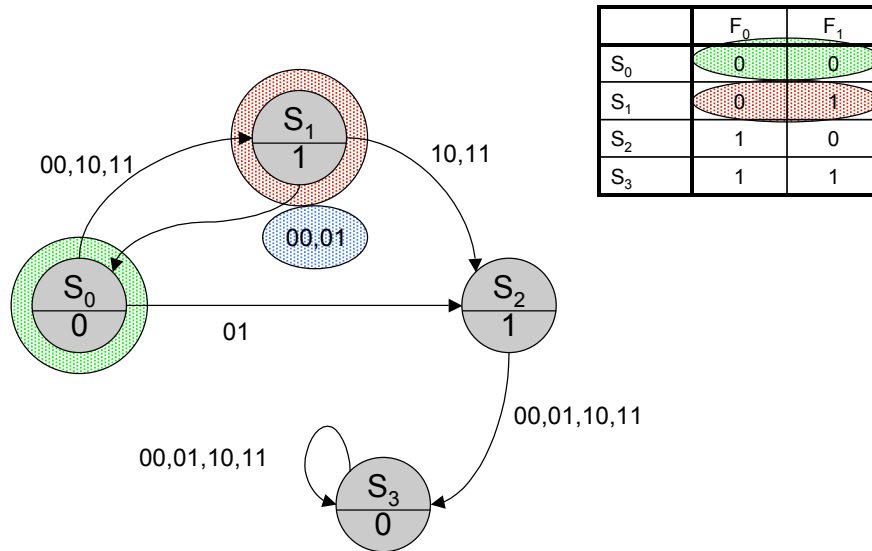
Übergangsfunktion

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0	1	0												
F ₁ '	1	1	0	1												

Übergangsfunktion

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0	1	0												
F ₁ '	1	1	0	1												

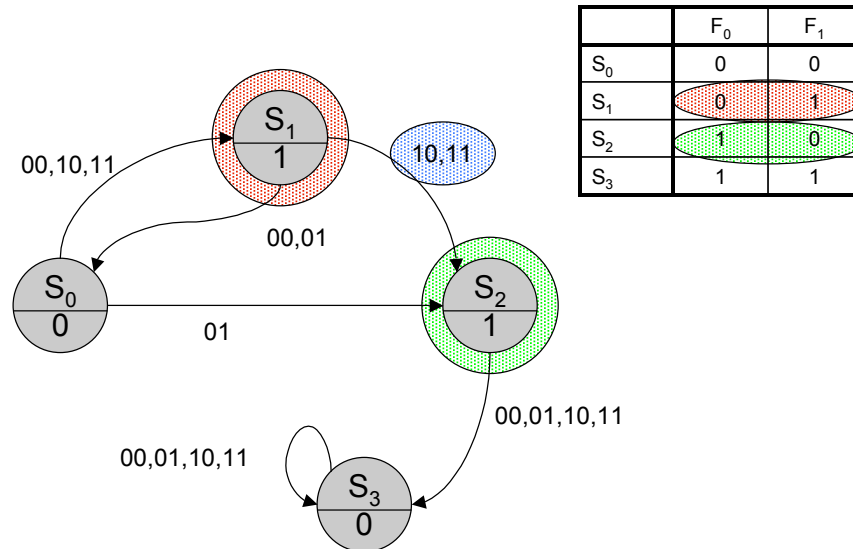
Übergangsfunktion



Übergangsfunktion

	S_0				S_1				S_2				S_3			
F_0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F_1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e_1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e_2	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F_0'	0	0	1	0	0		0									
F_1'	1	1	0	1	0		0									

Übergangsfunktion



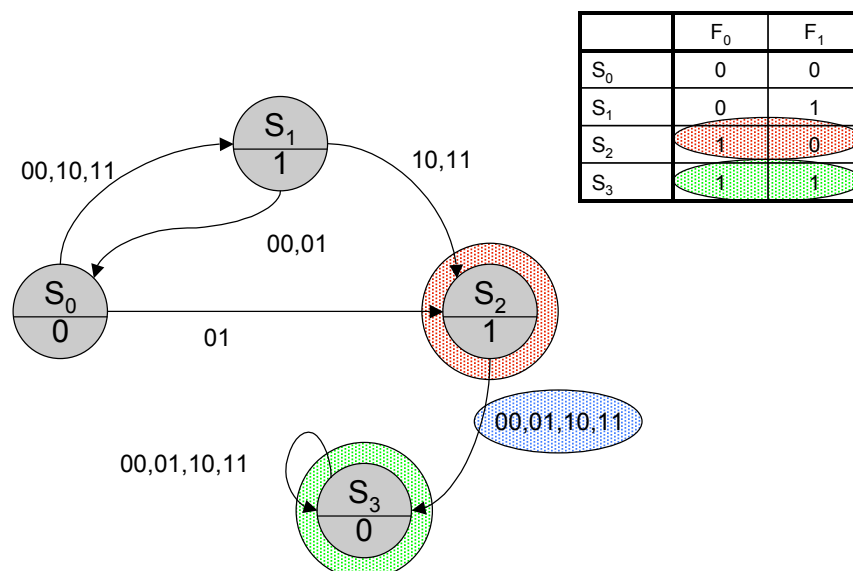
Übergangsfunktion

	S_0				S_1				S_2				S_3			
F_0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F_1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e_1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e_2	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F_0'	0	0	1	0	0	1	0	1								
F_1'	1	1	0	1	0	0	0	0								

Übergangsfunktion

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0	1	0	0	1	0	1								
F ₁ '	1	1	0	1	0	0	0	0								

Übergangsfunktion



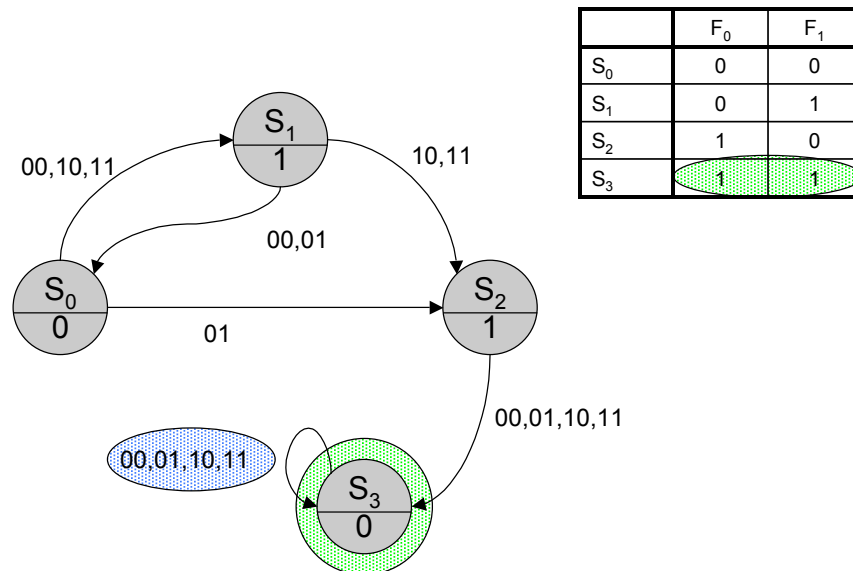
Übergangsfunktion

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0	1	0	0	1	0	1	1	1	1	1				
F ₁ '	1	1	0	1	0	0	0	0	1	1	1	1				

Übergangsfunktion

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0	1	0	0	1	0	1	1	1	1	1				
F ₁ '	1	1	0	1	0	0	0	0	1	1	1	1				

Übergangsfunktion



Übergangsfunktion

	S_0				S_1				S_2				S_3			
F_0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F_1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e_1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e_2	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F_0'	0	0	1	0	0	1	0	1	1	1	1	1	1	1	1	1
F_1'	1	1	0	1	0	0	0	0	1	1	1	1	1	1	1	1

Übergangsfunktion

- Übergangsfunktion mit 4 Eingängen und 2 Ausgängen
- kann minimiert werden (KV Diagram)

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0	1	0	0	1	0	1	1	1	1	1	1	1	1	1
F ₁ '	1	1	0	1	0	0	0	0	1	1	1	1	1	1	1	1

Ausgabefunktion

- Übergangsfunktion
 - für Moore-Schaltwerk und Mealy-Schaltwerk unterschiedlich
 - Moore-Schaltwerk
 - logische Funktion, die Zustand auf den Ausgang abbildet
 - Mealy-Schaltwerk
 - logische Funktion, die Eingangswerte und Zustand auf den Ausgang abbildet
- Darstellung als Wahrheitstabellen
- Fortsetzung des Beispiels von vorhin

Ausgabefunktion

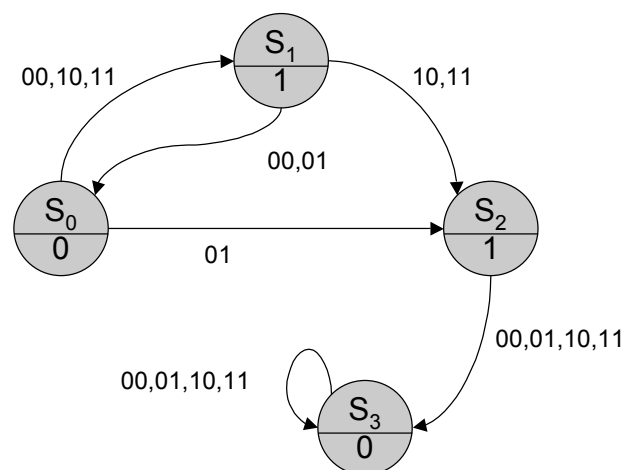
- Moore-Schaltwerk

	S ₀	S ₁	S ₂	S ₃
F ₀	0	0	1	1
F ₁	0	1	0	1
a ₁				

Ausgabefunktion

- Moore-Schaltwerk

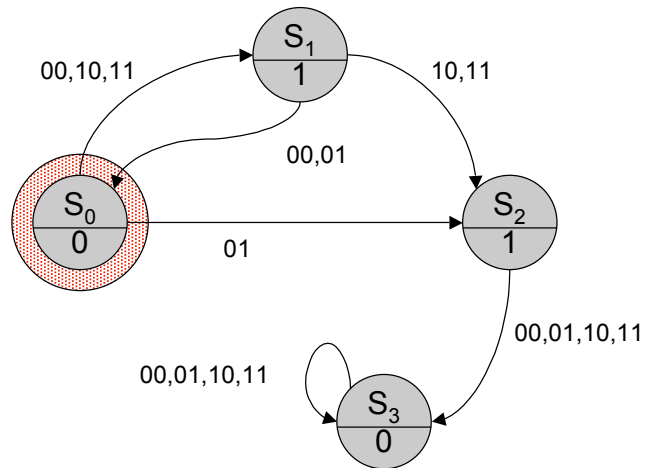
	S ₀	S ₁	S ₂	S ₃
F ₀	0	0	1	1
F ₁	0	1	0	1
a ₁				



Ausgabefunktion

- Moore-Schaltwerk

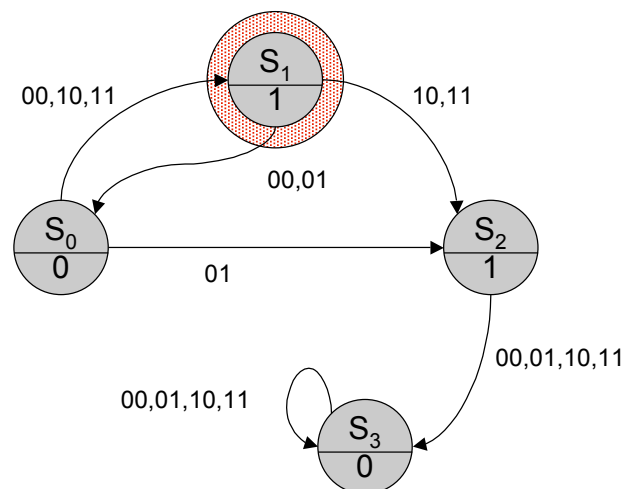
	S ₀	S ₁	S ₂	S ₃
F ₀	0	0	1	1
F ₁	0	1	0	1
a ₁	0			



Ausgabefunktion

- Moore-Schaltwerk

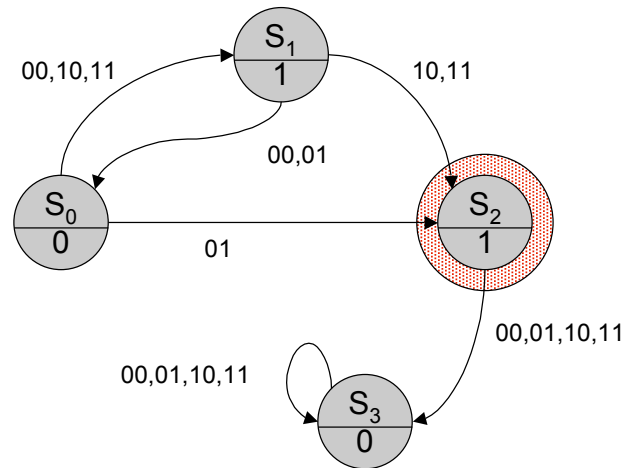
	S ₀	S ₁	S ₂	S ₃
F ₀	0	0	1	1
F ₁	0	1	0	1
a ₁	0	1		



Ausgabefunktion

- Moore-Schaltwerk

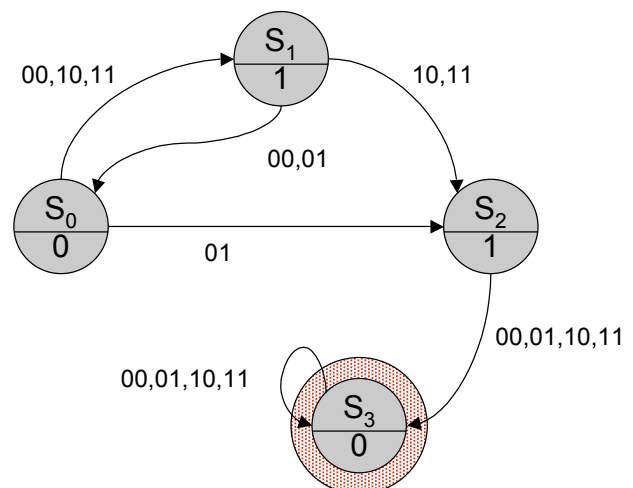
	S ₀	S ₁	S ₂	S ₃
F ₀	0	0	1	1
F ₁	0	1	0	1
a ₁	0	1	1	1



Ausgabefunktion

- Moore-Schaltwerk

	S ₀	S ₁	S ₂	S ₃
F ₀	0	0	1	1
F ₁	0	1	0	1
a ₁	0	1	1	0



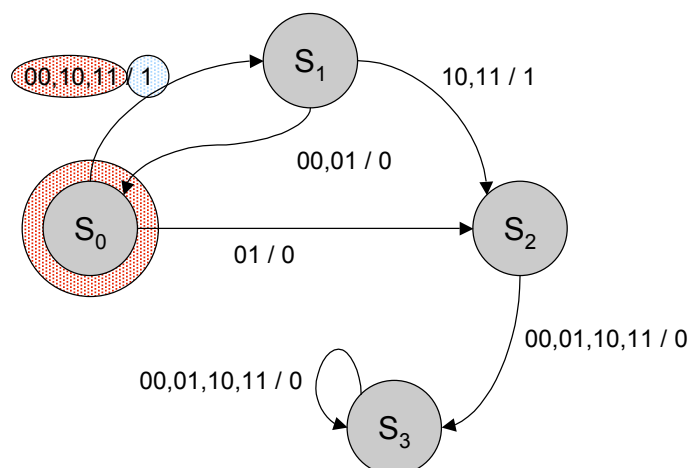
Ausgabefunktion

- Mealy-Schaltwerk

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
a ₁																

Ausgabefunktion

- Mealy-Schaltwerk



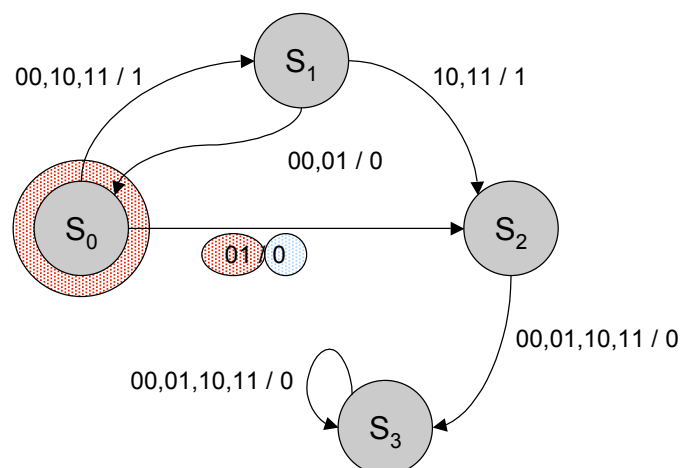
Ausgabefunktion

- Mealy-Schaltwerk

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
a ₁	1	1		1												

Ausgabefunktion

- Mealy-Schaltwerk



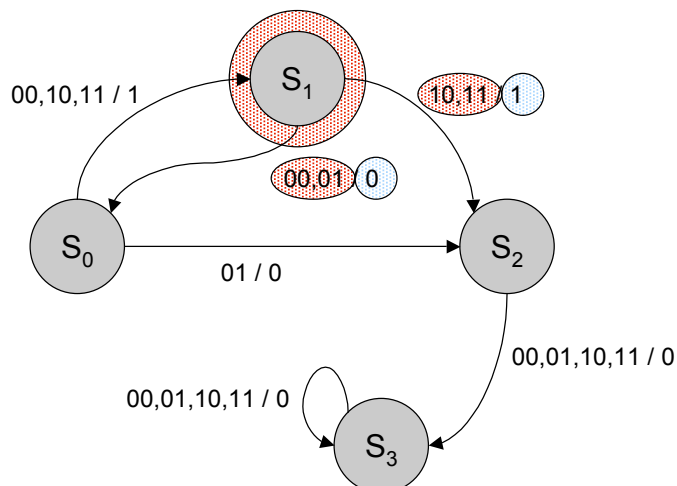
Ausgabefunktion

- Mealy-Schaltwerk

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
a ₁	1	1	0	1												

Ausgabefunktion

- Mealy-Schaltwerk



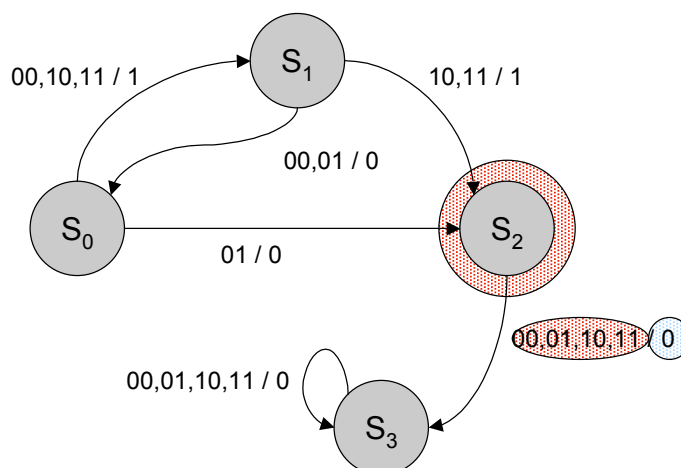
Ausgabefunktion

- Mealy-Schaltwerk

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
a ₁	1	1	0	1	0	1	0	1								

Ausgabefunktion

- Mealy-Schaltwerk



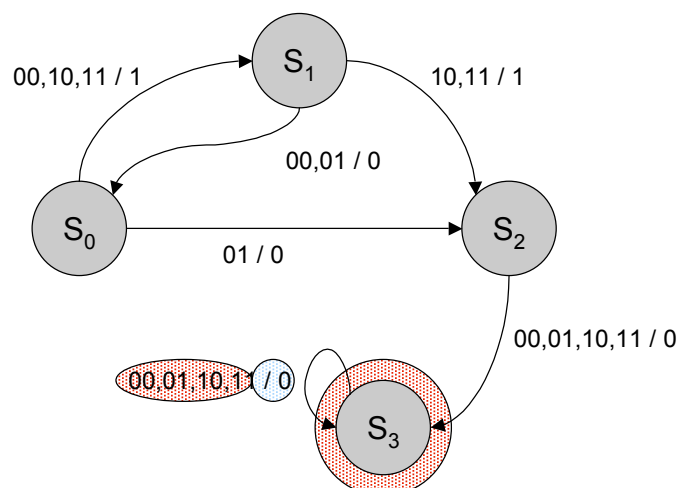
Ausgabefunktion

- Mealy-Schaltwerk

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
a ₁	1	1	0	1	0	1	0	1	0	0	0	0				

Ausgabefunktion

- Mealy-Schaltwerk



Ausgabefunktion

- Mealy-Schaltwerk

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
e ₁	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
e ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
a ₁	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0

Schaltwerk Beispiel

- Schritte zur Realisierung eines Schaltwerks
 - Verstehen der Aufgabenstellung
 - Aufbau des passenden Zustandsgraphen
 - Festlegen der Zustandskodierung
 - Übergangsfunktion bestimmen
 - Ausgabefunktion bestimmen

Schaltwerk Beispiel

Automation Systems Group

- Aufgabenstellung

Realisieren Sie einen 2-bit Zähler mit 2 Eingängen (up, down). Der Zähler fängt bei 0 an. Wenn up den Wert 1 hat, soll der Zählerstand um eins erhöht werden, wenn down den Wert 1 hat, soll der Zählerstand um eins erniedrigt werden. Wenn beide Eingänge 0 oder beide 1 sind, ändert sich der Zustand nicht. Es gibt keinen Overflow und Underflow. Die Ausgabe soll der aktuelle Stand des Zählers sein.

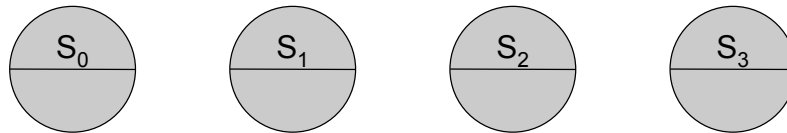
Schaltwerk Beispiel

Automation Systems Group

- Schritte zur Realisierung eines Schaltwerks
 - Verstehen der Aufgabenstellung
 - Aufbau des passenden Zustandsgraphen
 - Festlegen der Zustandscodierung
 - Übergangsfunktion bestimmen
 - Ausgabefunktion bestimmen

Schaltwerk Beispiel

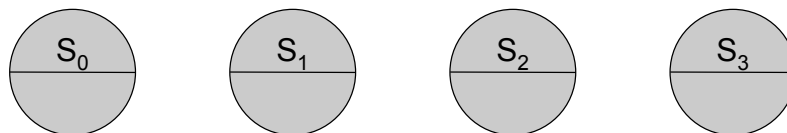
Automation Systems Group



Schaltwerk Beispiel

Automation Systems Group

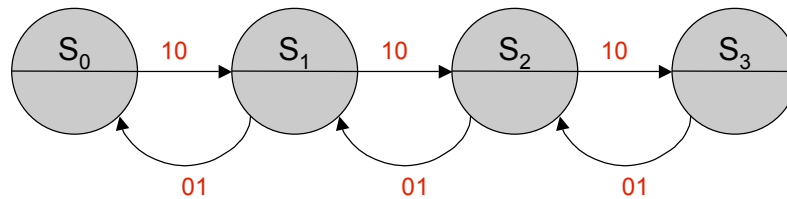
Eingangsbitmuster: up, down



Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: up, down

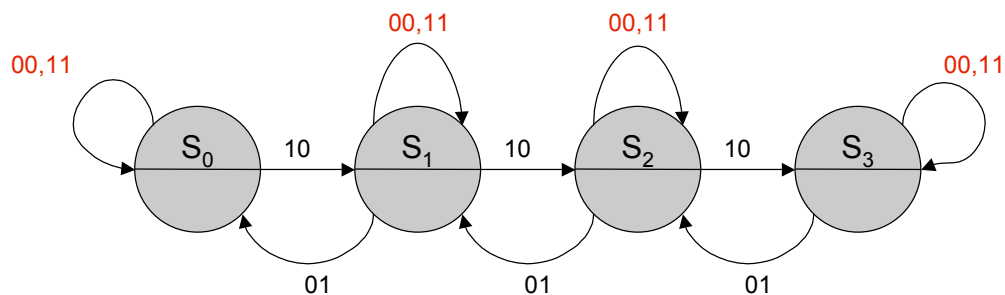


Zählerfunktion

Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: up, down

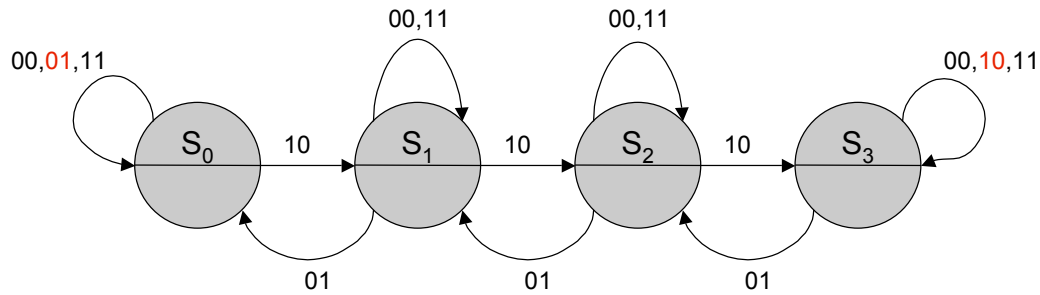


Keine Änderung wenn beide
Eingänge gleich sind

Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: up, down



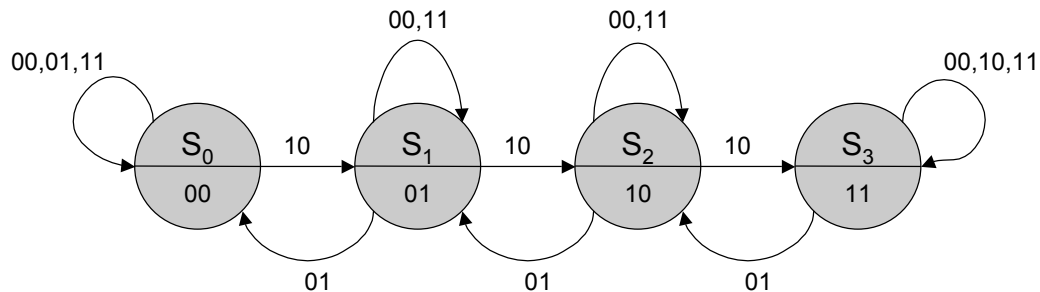
Kein Overflow und kein Underflow

Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: up, down

Ausgabebitmuster: msb, lsb



Schaltwerk Beispiel

- Schritte zur Realisierung eines Schaltwerks
 - Verstehen der Aufgabenstellung
 - Aufbau des passenden Zustandsgraphen
 - Festlegen der Zustandskodierung
 - Übergangsfunktion bestimmen
 - Ausgabefunktion bestimmen

Schaltwerk Beispiel

- Zustandskodierung
 - dichte Codierung
 - 4 Zustände benötigen 2 Flip-Flops

	F_0	F_1
S_0	0	0
S_1	0	1
S_2	1	0
S_3	1	1

Schaltwerk Beispiel

- Schritte zur Realisierung eines Schaltwerks
 - Verstehen der Aufgabenstellung
 - Aufbau des passenden Zustandsgraphen
 - Festlegen der Zustandskodierung
 - Übergangsfunktion bestimmen
 - Ausgabefunktion bestimmen

Schaltwerk Beispiel

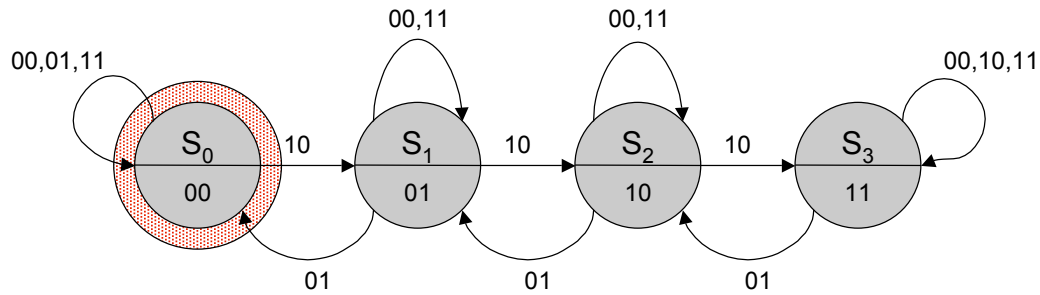
	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
up	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
down	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '																
F ₁ '																

Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: up, down

Ausgabebitmuster: msb, lsb



Schaltwerk Beispiel

Automation Systems Group

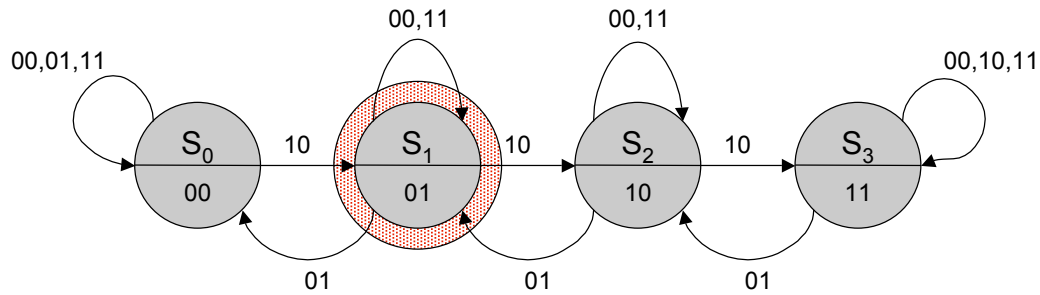
	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
up	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
down	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0	0	0												
F ₁ '	0	1	0	0												

Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: up, down

Ausgabebitmuster: msb, lsb



Schaltwerk Beispiel

Automation Systems Group

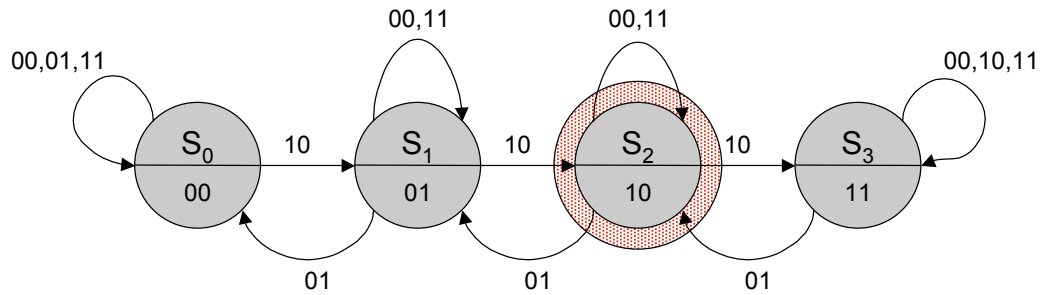
	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
up	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
down	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0	0	0	0	1	0	0								
F ₁ '	0	1	0	0	1	0	0	1								

Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: up, down

Ausgabebitmuster: msb, lsb



Schaltwerk Beispiel

Automation Systems Group

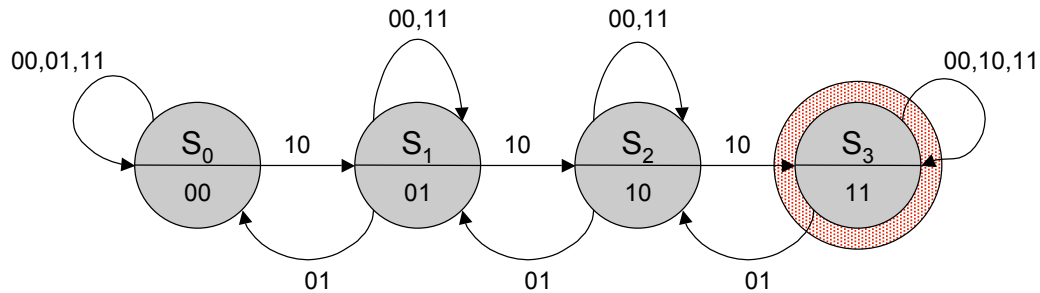
	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
up	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
down	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0	0	0	0	1	0	0	1	1	0	1				
F ₁ '	0	1	0	0	1	0	0	1	0	1	1	0				

Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: up, down

Ausgabebitmuster: msb, lsb



Schaltwerk Beispiel

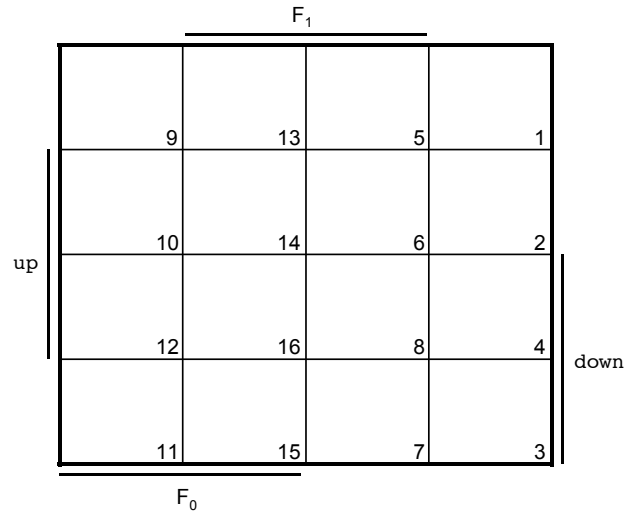
Automation Systems Group

	S ₀				S ₁				S ₂				S ₃			
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
up	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
down	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
F ₀ '	0	0	0	0	0	1	0	0	1	1	0	1	1	1	1	1
F ₁ '	0	1	0	0	1	0	0	1	0	1	1	0	1	1	0	1

Schaltwerk Beispiel

Automation Systems Group

F_0	F_1	down	up	F_0'	F_1'
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	0	0
0	0	1	1	0	0
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	0	0
0	1	1	1	0	1
1	0	0	0	1	0
1	0	0	1	1	1
1	0	1	0	0	1
1	0	1	1	1	0
1	1	0	0	1	1
1	1	0	1	1	1
1	1	1	0	1	0
1	1	1	1	1	1



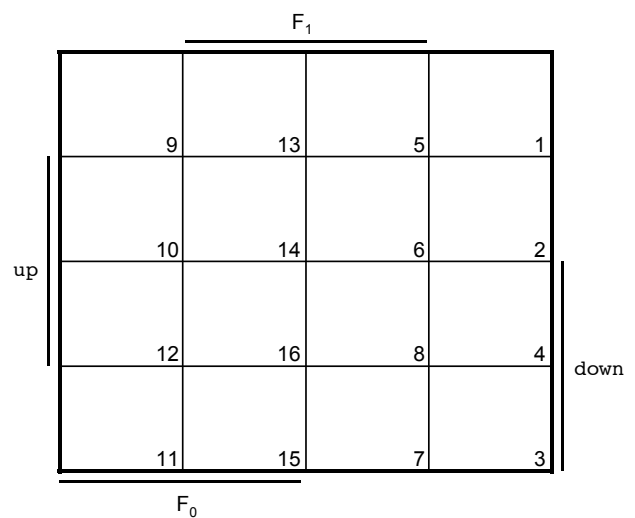
Einführung in die technische Informatik

103

Schaltwerk Beispiel

Automation Systems Group

F_0	F_1	down	up	F_0'	F_1'
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	0	0
0	0	1	1	0	0
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	0	0
0	1	1	1	0	1
1	0	0	0	1	0
1	0	0	1	1	1
1	0	1	0	0	1
1	0	1	1	1	0
1	1	0	0	1	1
1	1	0	1	1	1
1	1	1	0	1	0
1	1	1	1	1	1



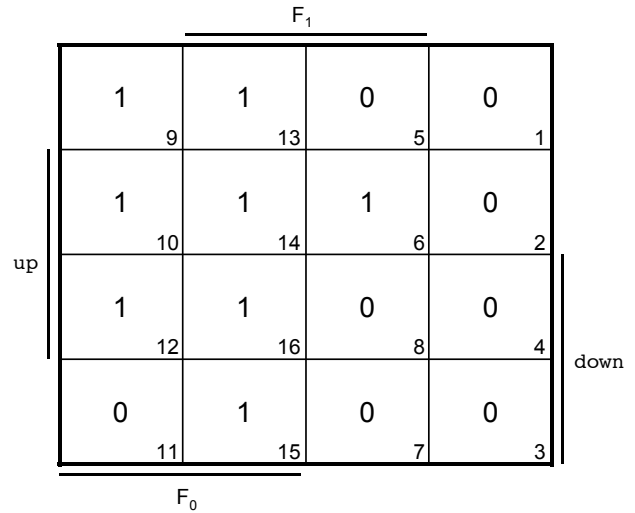
Einführung in die technische Informatik

104

Schaltwerk Beispiel

Automation Systems Group

F_0	F_1	down	up	F_0'	F_1'
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	0	0
0	0	1	1	0	0
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	0	0
0	1	1	1	0	1
1	0	0	0	1	0
1	0	0	1	1	1
1	0	1	0	0	1
1	0	1	1	1	0
1	1	0	0	1	1
1	1	0	1	1	1
1	1	1	0	1	0
1	1	1	1	1	1



Einführung in die technische Informatik

105

Schaltwerk Beispiel

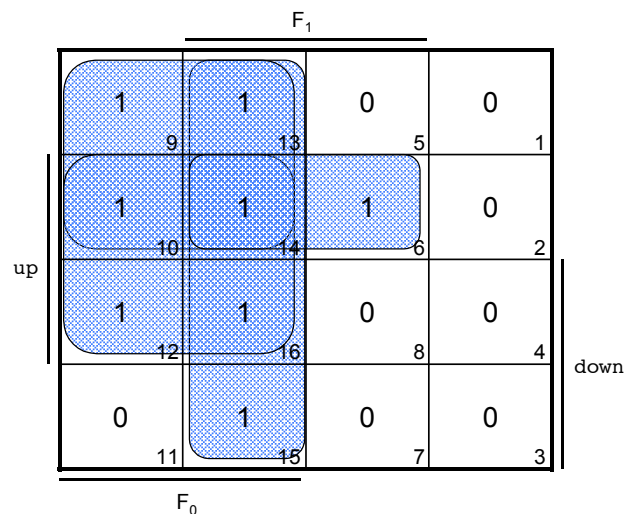
Automation Systems Group

$$(F_0 \wedge up) \vee$$

$$(F_0 \wedge \neg down) \vee$$

$$(F_0 \wedge F_1) \vee$$

$$(F_1 \wedge up \wedge \neg down)$$



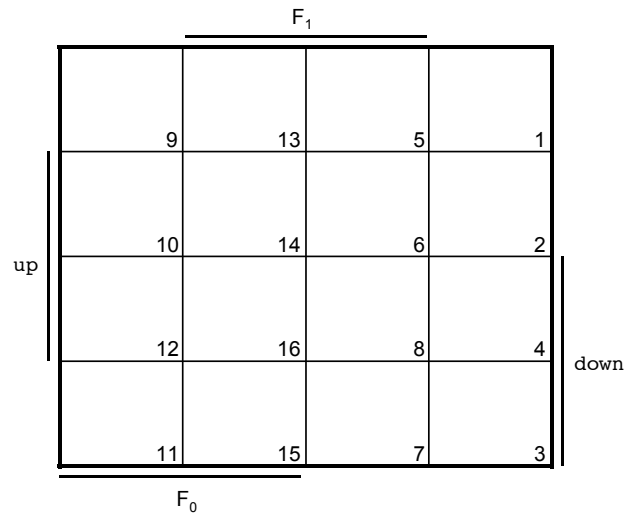
Einführung in die technische Informatik

106

Schaltwerk Beispiel

Automation Systems Group

F_0	F_1	down	up	F_0'	F_1'
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	0	0
0	0	1	1	0	0
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	0	0
0	1	1	1	0	1
1	0	0	0	1	0
1	0	0	1	1	1
1	0	1	0	0	1
1	0	1	1	1	0
1	1	0	0	1	1
1	1	0	1	1	1
1	1	1	0	1	0
1	1	1	1	1	1



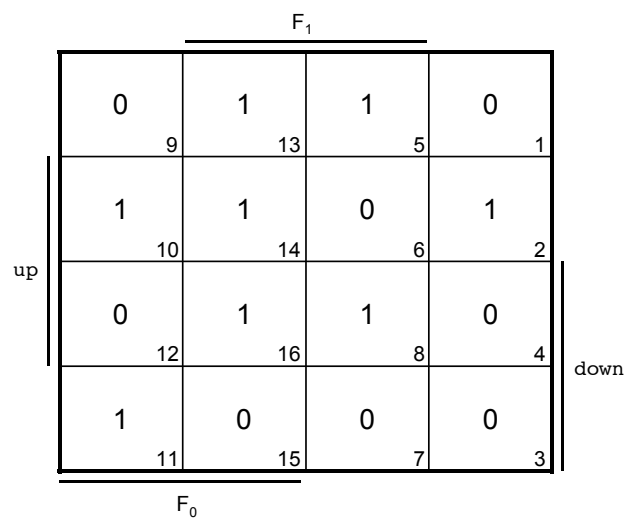
Einführung in die technische Informatik

107

Schaltwerk Beispiel

Automation Systems Group

F_0	F_1	down	up	F_0'	F_1'
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	0	0
0	0	1	1	0	0
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	0	0
0	1	1	1	0	1
1	0	0	0	1	0
1	0	0	1	1	1
1	0	1	0	0	1
1	0	1	1	1	0
1	1	0	0	1	1
1	1	0	1	1	1
1	1	1	0	1	0
1	1	1	1	1	1



Einführung in die technische Informatik

108

$$\begin{aligned} & (F_1 \wedge \neg up \wedge \neg down) \vee \\ & (F_1 \wedge up \wedge down) \vee \\ & (F_0 \wedge up \wedge \neg down) \vee \\ & (\neg F_1 \wedge up \wedge \neg down) \vee \\ & (F_0 \wedge \neg F_1 \wedge \neg up \wedge down) \end{aligned}$$

- Schritte zur Realisierung eines Schaltwerks
 - Verstehen der Aufgabenstellung
 - Aufbau des passenden Zustandsgraphen
 - Festlegen der Zustandscodierung
 - Übergangsfunktion bestimmen
 - Ausgabefunktion bestimmen

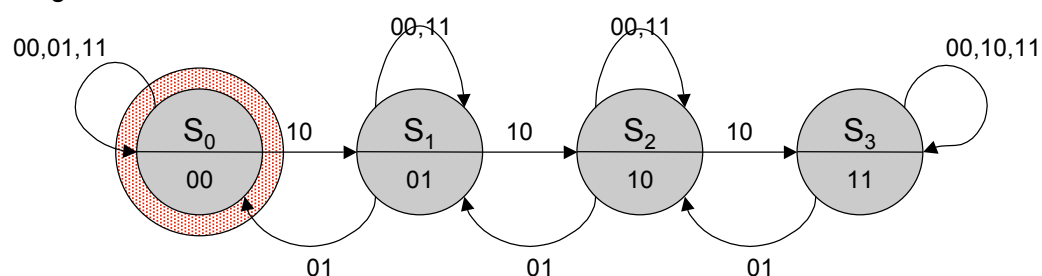
Schaltwerk Beispiel

- Ausgabefunktion des Moore-Schaltwerks hängt nur vom Zustand ab

	S ₀	S ₁	S ₂	S ₃
F ₀	0	0	1	1
F ₁	0	1	0	1
msb				
lsb				

Schaltwerk Beispiel

Ausgabebitmuster: msb, lsb

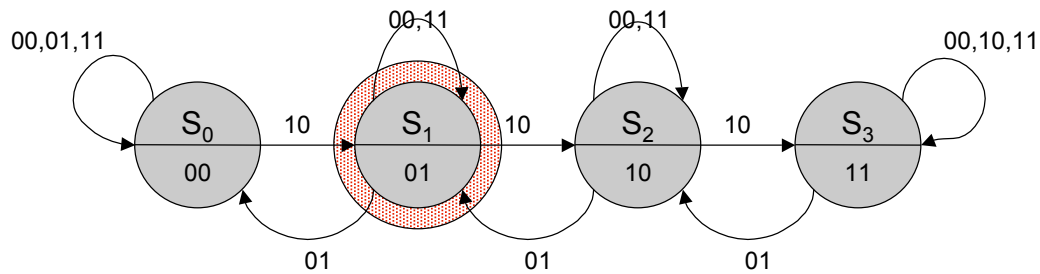


	S ₀	S ₁	S ₂	S ₃
F ₀	0	0	1	1
F ₁	0	1	0	1
msb	0			
lsb	0			

Schaltwerk Beispiel

Automation Systems Group

Ausgabebitmuster: msb, lsb

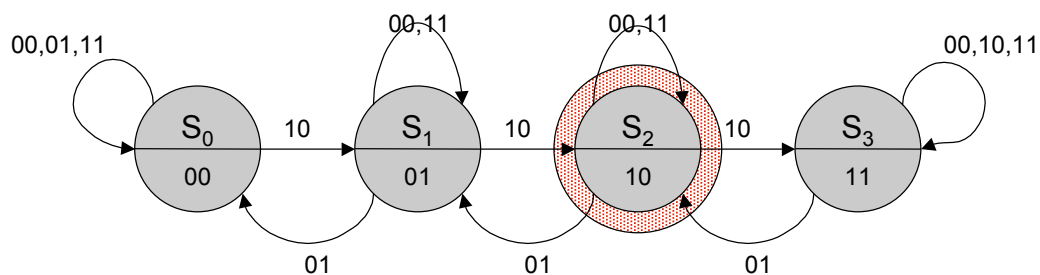


	S ₀	S ₁	S ₂	S ₃
F ₀	0	0	1	1
F ₁	0	1	0	1
msb	0	0		
lsb	0	1		

Schaltwerk Beispiel

Automation Systems Group

Ausgabebitmuster: msb, lsb

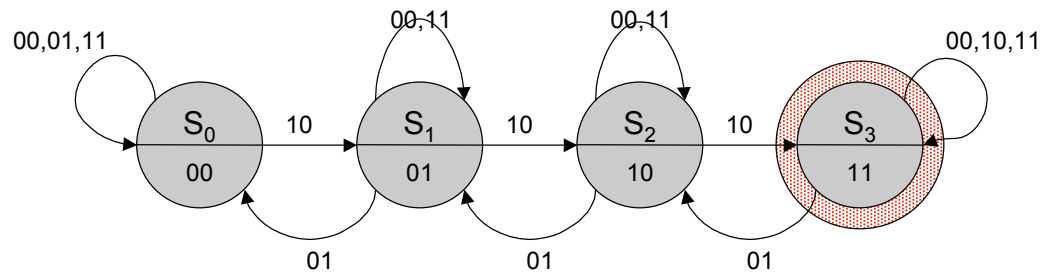


	S ₀	S ₁	S ₂	S ₃
F ₀	0	0	1	1
F ₁	0	1	0	1
msb	0	0	1	
lsb	0	1	0	

Schaltwerk Beispiel

Automation Systems Group

Ausgabebitmuster: msb, lsb



	S ₀	S ₁	S ₂	S ₃
F ₀	0	0	1	1
F ₁	0	1	0	1
msb	0	0	1	1
lsb	0	1	0	1

Schaltwerk Beispiel

Automation Systems Group

- Ausgabefunktion ist mit Zustandswerten identisch
→ keine zusätzliche Logik (und Minimierung) erforderlich

	S ₀	S ₁	S ₂	S ₃
F ₀	0	0	1	1
F ₁	0	1	0	1
msb	0	0	1	1
lsb	0	1	0	1

Schaltwerk Beispiel

Automation Systems Group

- Bis jetzt wurden alle Zustände der Flip-Flops benötigt
- Bei dichter Codierung kann der Fall auftreten, dass manche Zustände nicht benötigt werden
- Diese Zustände werden mit „don't care“ Werten in der Übergangsfunktion und der Ausgabefunktion behandelt

Schaltwerk Beispiel

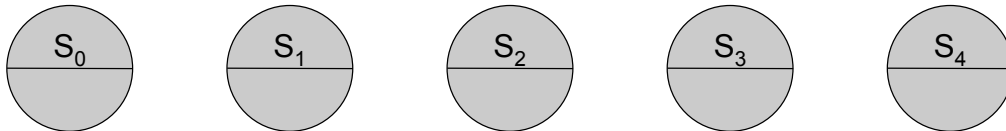
Automation Systems Group

- Aufgabenstellung

Realisieren Sie ein Schaltwerk, welches 1 ausgibt, wenn am Eingang e die Bitfolge 1010 aufgetreten ist. Der Eingang e liefert jeden Takt ein neues Bit.

Schaltwerk Beispiel

Automation Systems Group

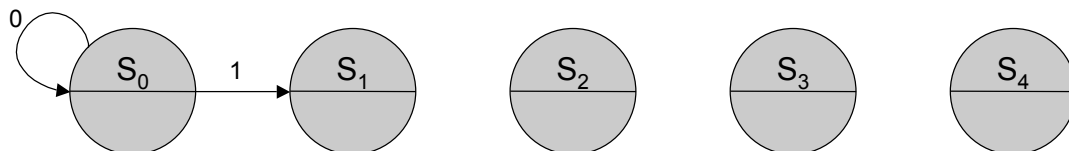


	Zustand
S_0	-
S_1	1
S_2	10
S_3	101
S_4	1010

Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: e

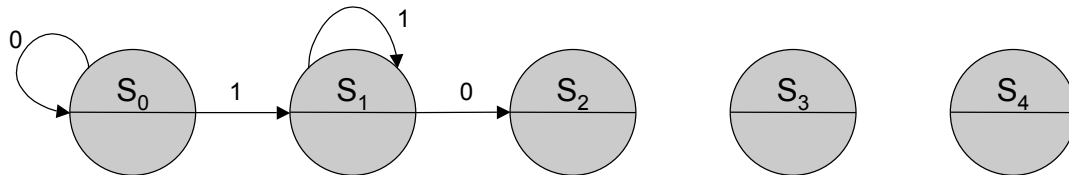


	Zustand
S_0	-
S_1	1
S_2	10
S_3	101
S_4	1010

Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: e

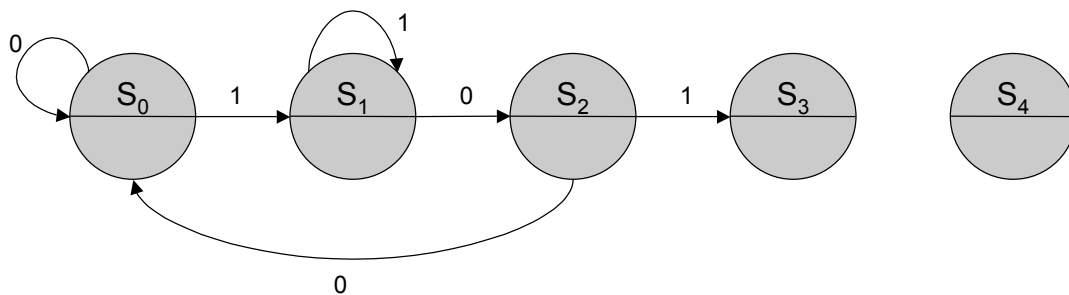


	Zustand
S ₀	-
S ₁	1
S ₂	10
S ₃	101
S ₄	1010

Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: e

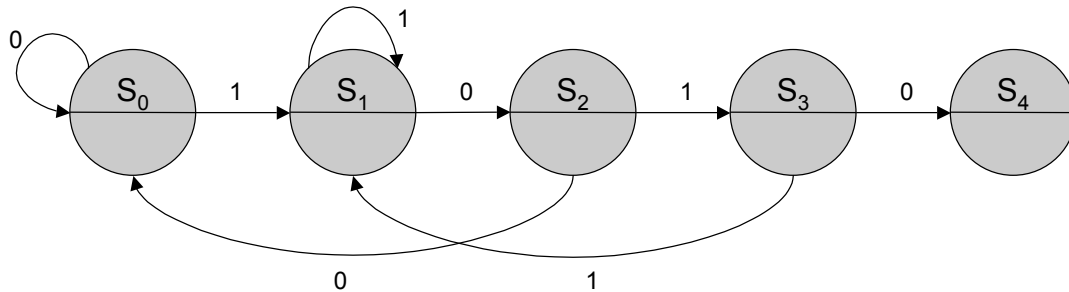


	Zustand
S ₀	-
S ₁	1
S ₂	10
S ₃	101
S ₄	1010

Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: e

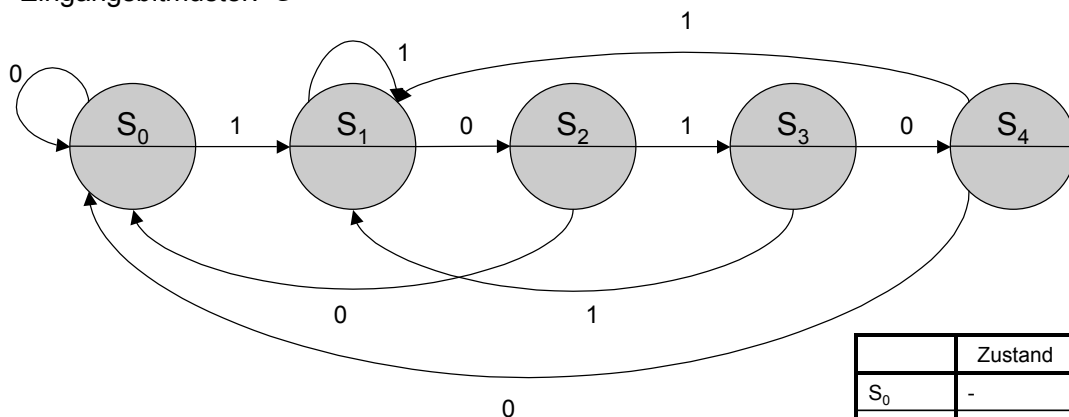


	Zustand
S ₀	-
S ₁	1
S ₂	10
S ₃	101
S ₄	1010

Schaltwerk Beispiel

Automation Systems Group

Eingangsbitmuster: e

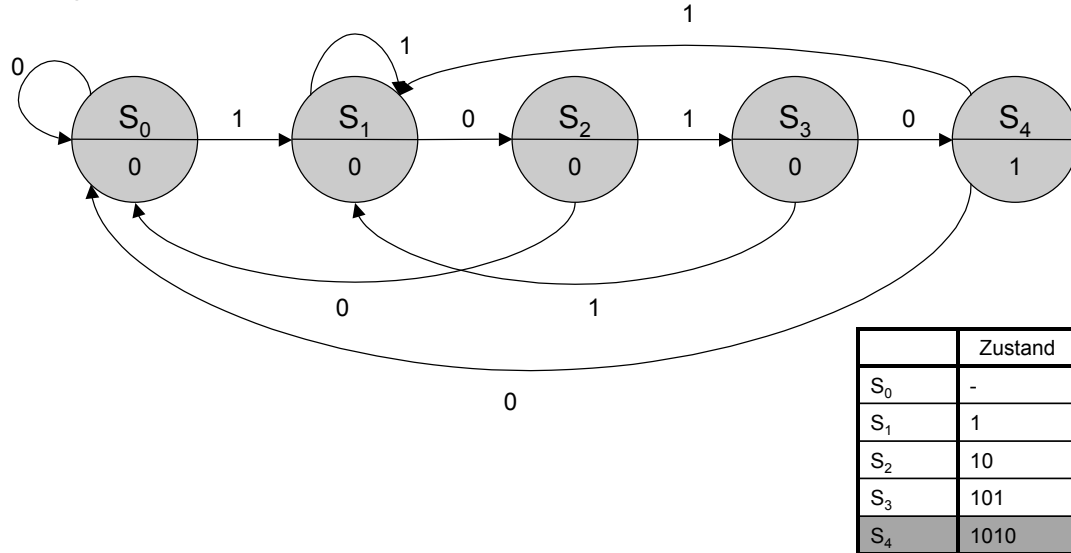


	Zustand
S ₀	-
S ₁	1
S ₂	10
S ₃	101
S ₄	1010

Schaltwerk Beispiel

Automation Systems Group

Ausgabebitmuster: a



Einführung in die technische Informatik

125

Schaltwerk Beispiel

Automation Systems Group

- Zustandskodierung
 - dichte Codierung
 - 5 Zustände benötigen 3 Flip-Flops
 - diese könnten jedoch 8 Zustände darstellen
 - einige sind daher unbelegt

	F ₀	F ₁	F ₂
S ₀	0	0	0
S ₁	0	0	1
S ₂	0	1	0
S ₃	0	1	1
S ₄	1	0	0
	1	0	1
	1	1	0
	1	1	1

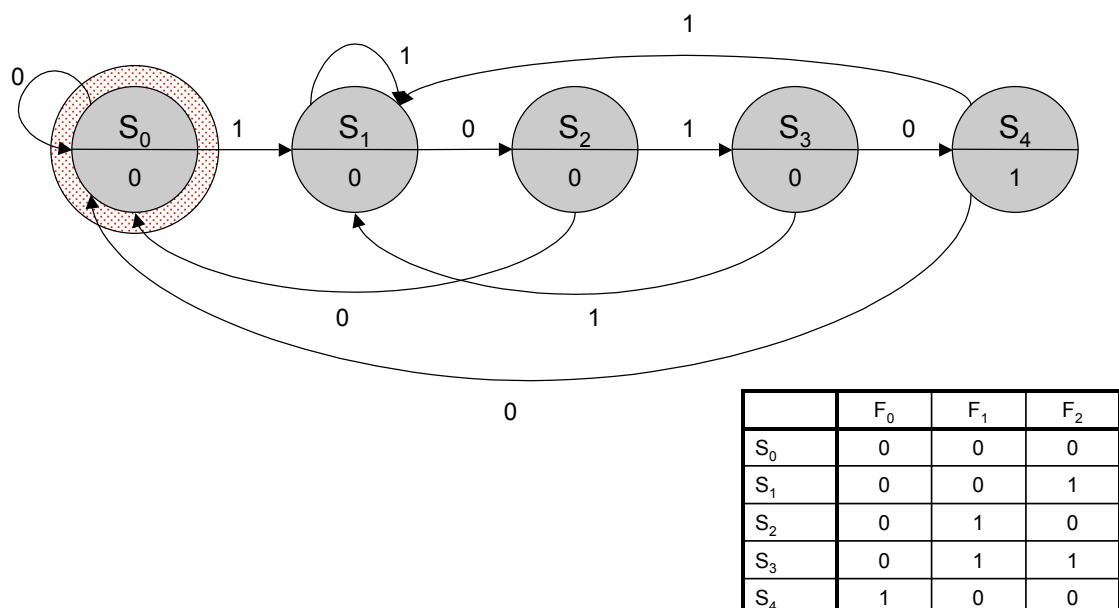
Einführung in die technische Informatik

126

Schaltwerk Beispiel

	S ₀		S ₁		S ₂		S ₃		S ₄							
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
F ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
e	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
F ₀ '																
F ₁ '																
F ₂ '																

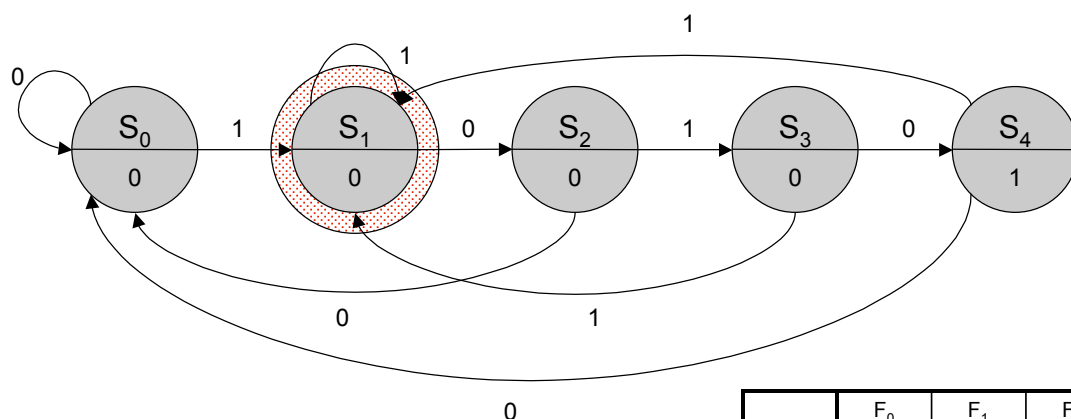
Schaltwerk Beispiel



Schaltwerk Beispiel

	S ₀		S ₁		S ₂		S ₃		S ₄							
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
F ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
e	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
F ₀ '	0	0														
F ₁ '	0	0														
F ₂ '	0	1														

Schaltwerk Beispiel

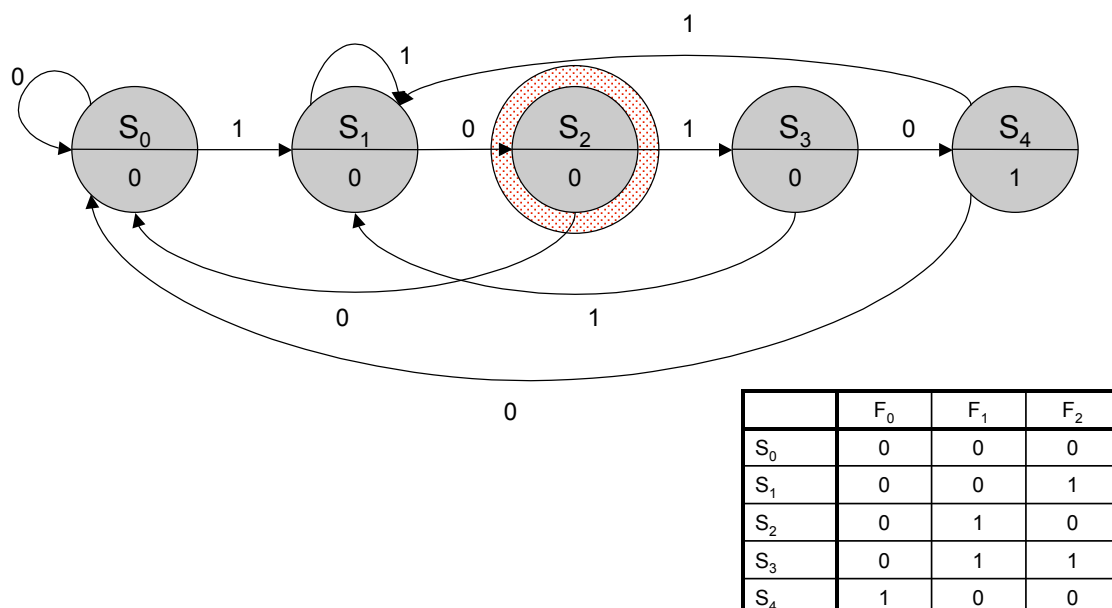


	F ₀	F ₁	F ₂
S ₀	0	0	0
S ₁	0	0	1
S ₂	0	1	0
S ₃	0	1	1
S ₄	1	0	0

Schaltwerk Beispiel

	S ₀		S ₁		S ₂		S ₃		S ₄							
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
F ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
e	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
F ₀ '	0	0	0	0												
F ₁ '	0	0	1	0												
F ₂ '	0	1	0	1												

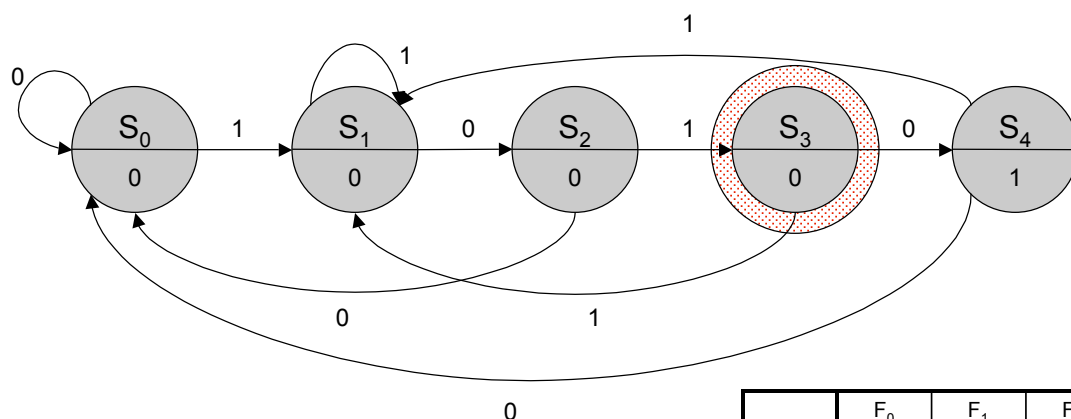
Schaltwerk Beispiel



Schaltwerk Beispiel

	S ₀		S ₁		S ₂		S ₃		S ₄							
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
F ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
e	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
F ₀ '	0	0	0	0	0	0										
F ₁ '	0	0	1	0	0	1										
F ₂ '	0	1	0	1	0	1										

Schaltwerk Beispiel

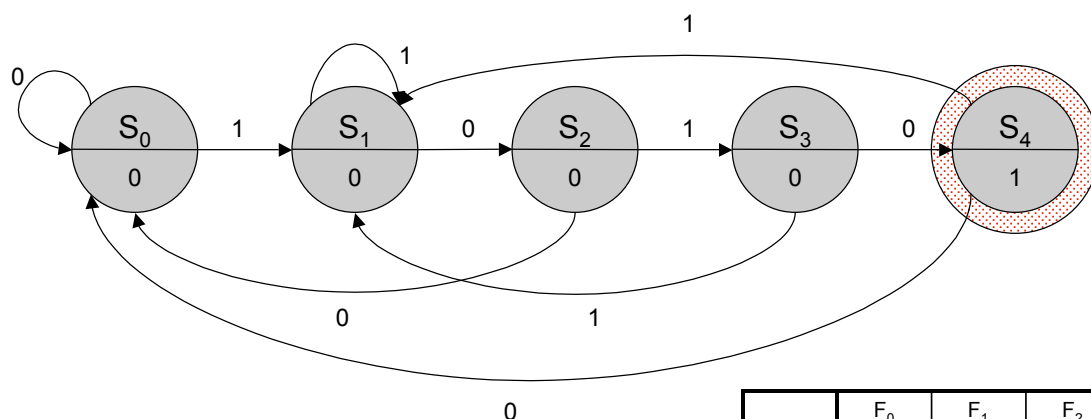


	F ₀	F ₁	F ₂
S ₀	0	0	0
S ₁	0	0	1
S ₂	0	1	0
S ₃	0	1	1
S ₄	1	0	0

Schaltwerk Beispiel

	S ₀		S ₁		S ₂		S ₃		S ₄							
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
F ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
e	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
F ₀ '	0	0	0	0	0	0	1	0								
F ₁ '	0	0	1	0	0	1	0	0								
F ₂ '	0	1	0	1	0	1	0	1								

Schaltwerk Beispiel



	F ₀	F ₁	F ₂
S ₀	0	0	0
S ₁	0	0	1
S ₂	0	1	0
S ₃	0	1	1
S ₄	1	0	0

Schaltwerk Beispiel

	S ₀		S ₁		S ₂		S ₃		S ₄							
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
F ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
e	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
F ₀ '	0	0	0	0	0	0	1	0	0	0						
F ₁ '	0	0	1	0	0	1	0	0	0	0						
F ₂ '	0	1	0	1	0	1	0	1	0	1						

Schaltwerk Beispiel

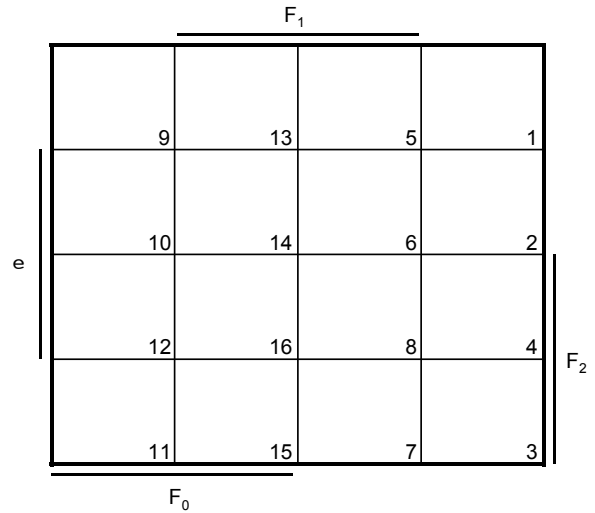
	S ₀		S ₁		S ₂		S ₃		S ₄							
F ₀	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
F ₁	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
F ₂	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
e	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
F ₀ '	0	0	0	0	0	0	1	0	0	0	X	X	X	X	X	X
F ₁ '	0	0	1	0	0	1	0	0	0	0	X	X	X	X	X	X
F ₂ '	0	1	0	1	0	1	0	1	0	1	X	X	X	X	X	X

Don't care Einträge

Schaltwerk Beispiel

Automation Systems Group

F_0	F_1	F_2	e	F_0'	F_1'	F_2'
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	0	1
0	1	0	0	0	0	0
0	1	0	1	0	1	1
0	1	1	0	1	0	0
0	1	1	1	0	0	1
1	0	0	0	0	0	0
1	0	0	1	0	0	1
1	0	1	0	X	X	X
1	0	1	1	X	X	X
1	1	0	0	X	X	X
1	1	0	1	X	X	X
1	1	1	0	X	X	X
1	1	1	1	X	X	X



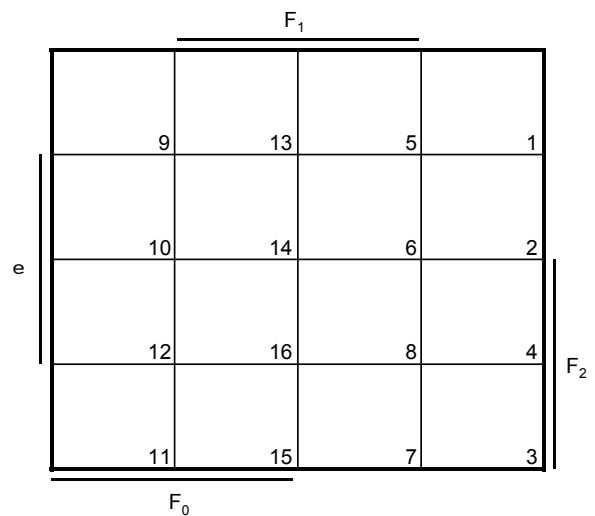
Einführung in die technische Informatik

139

Schaltwerk Beispiel

Automation Systems Group

F_0	F_1	F_2	e	F_0'	F_1'	F_2'
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	0	1
0	1	0	0	0	0	0
0	1	0	1	0	1	1
0	1	1	0	1	0	0
0	1	1	1	0	0	1
1	0	0	0	0	0	0
1	0	0	1	0	0	1
1	0	1	0	X	X	X
1	0	1	1	X	X	X
1	1	0	0	X	X	X
1	1	0	1	X	X	X
1	1	1	0	X	X	X
1	1	1	1	X	X	X



Einführung in die technische Informatik

140

Schaltwerk Beispiel

Automation Systems Group

F_0	F_1	F_2	e	F_0'	F_1'	F_2'
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	0	1
0	1	0	0	0	0	0
0	1	0	1	0	1	1
0	1	1	0	1	0	0
0	1	1	1	0	0	1
1	0	0	0	0	0	0
1	0	0	1	0	0	1
1	0	1	0	X	X	X
1	0	1	1	X	X	X
1	1	0	0	X	X	X
1	1	0	1	X	X	X
1	1	1	0	X	X	X
1	1	1	1	X	X	X

F_1			
F_0	0	X	0
0	9	13	5
1	10	14	6
2	X	X	0
3	X	X	1
4	12	16	8
5	X	X	0
6	X	X	0
7	11	15	7
8	X	X	0
9	X	X	0
10	X	X	0
11	X	X	0
12	X	X	0
13	X	X	0
14	X	X	0
15	X	X	0

Einführung in die technische Informatik

141

Schaltwerk Beispiel

Automation Systems Group

$$(F_1 \wedge F_2 \wedge \neg e)$$

F_1			
F_0	0	X	0
0	9	13	5
1	10	14	6
2	X	X	0
3	X	X	1
4	12	16	8
5	X	X	0
6	X	X	0
7	11	15	7
8	X	X	0
9	X	X	0
10	X	X	0
11	X	X	0
12	X	X	0
13	X	X	0
14	X	X	0
15	X	X	0

Einführung in die technische Informatik

142

Schaltwerk Beispiel

Automation Systems Group

F_0	F_1	F_2	e	F_0'	F_1'	F_2'
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	0	1
0	1	0	0	0	0	0
0	1	0	1	0	1	1
0	1	1	0	1	0	0
0	1	1	1	0	0	1
1	0	0	0	0	0	0
1	0	0	1	0	0	1
1	0	1	0	X	X	X
1	0	1	1	X	X	X
1	1	0	0	X	X	X
1	1	0	1	X	X	X
1	1	1	0	X	X	X
1	1	1	1	X	X	X

F_1			
9	13	5	1
10	14	6	2
12	16	8	4
11	15	7	3

F_0

Einführung in die technische Informatik

143

Schaltwerk Beispiel

Automation Systems Group

F_0	F_1	F_2	e	F_0'	F_1'	F_2'
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	0	1
0	1	0	0	0	0	0
0	1	0	1	0	1	1
0	1	1	0	1	0	0
0	1	1	1	0	0	1
1	0	0	0	0	0	0
1	0	0	1	0	0	1
1	0	1	0	X	X	X
1	0	1	1	X	X	X
1	1	0	0	X	X	X
1	1	0	1	X	X	X
1	1	1	0	X	X	X
1	1	1	1	X	X	X

F_1			
0	X	0	0
9	13	5	1
0	X	1	0
10	14	6	2
X	X	0	0
12	16	8	4
X	X	0	1
11	15	7	3

F_0

Einführung in die technische Informatik

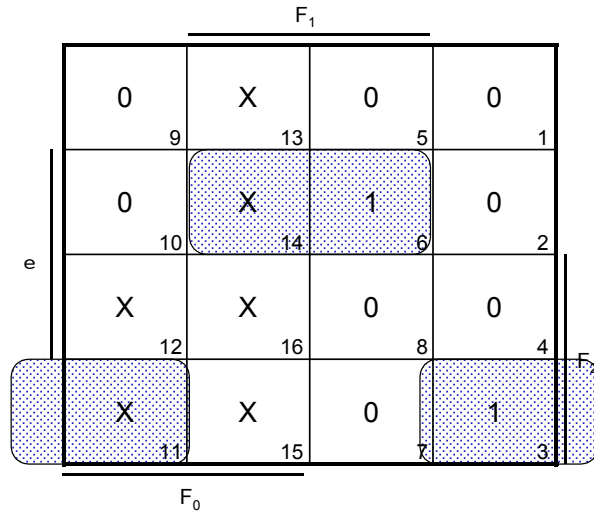
144

Schaltwerk Beispiel

Automation Systems Group

$$(F_1 \wedge \neg F_2 \wedge e) \vee$$

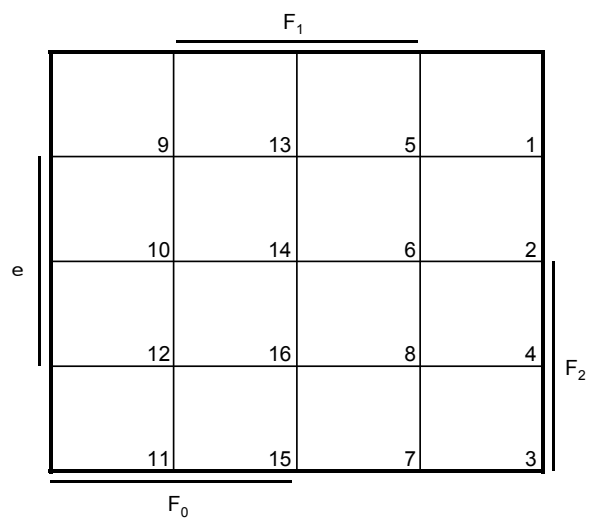
$$(\neg F_1 \wedge F_2 \wedge \neg e)$$



Schaltwerk Beispiel

Automation Systems Group

F_0	F_1	F_2	e	F_0'	F_1'	F_2'
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	0	1
0	1	0	0	0	0	0
0	1	0	1	0	1	1
0	1	1	0	1	0	0
0	1	1	1	0	0	1
1	0	0	0	0	0	0
1	0	0	1	0	0	1
1	0	1	0	X	X	X
1	0	1	1	X	X	X
1	1	0	0	X	X	X
1	1	0	1	X	X	X
1	1	1	0	X	X	X
1	1	1	1	X	X	X

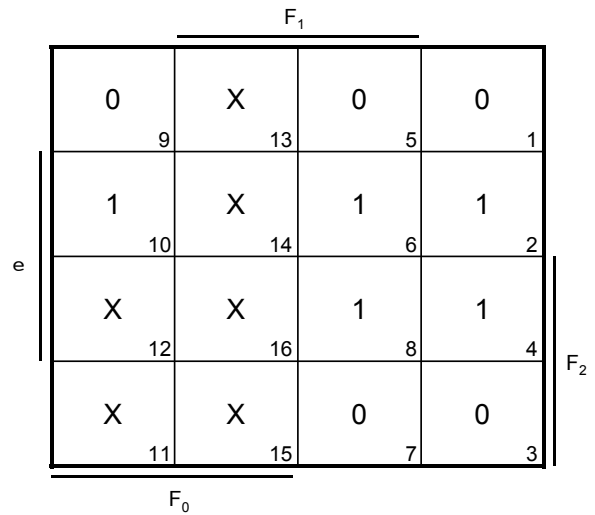


Schaltwerk Beispiel

Automation Systems Group

F_0	F_1	F_2	e	F_0'	F_1'	F_2'
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	0	1
0	1	0	0	0	0	0
0	1	0	1	0	1	1
0	1	1	0	1	0	0
0	1	1	1	0	0	1
1	0	0	0	0	0	0
1	0	0	1	0	0	1
1	0	1	0	X	X	X
1	0	1	1	X	X	X
1	1	0	0	X	X	X
1	1	0	1	X	X	X
1	1	1	0	X	X	X
1	1	1	1	X	X	X

Einführung in die technische Informatik

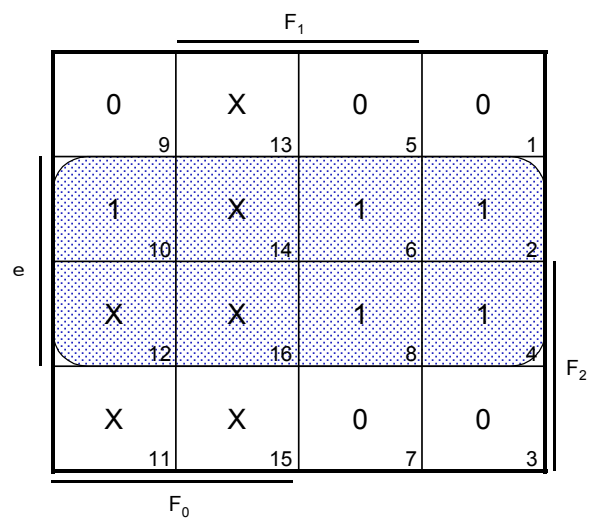


147

Schaltwerk Beispiel

Automation Systems Group

e



Einführung in die technische Informatik

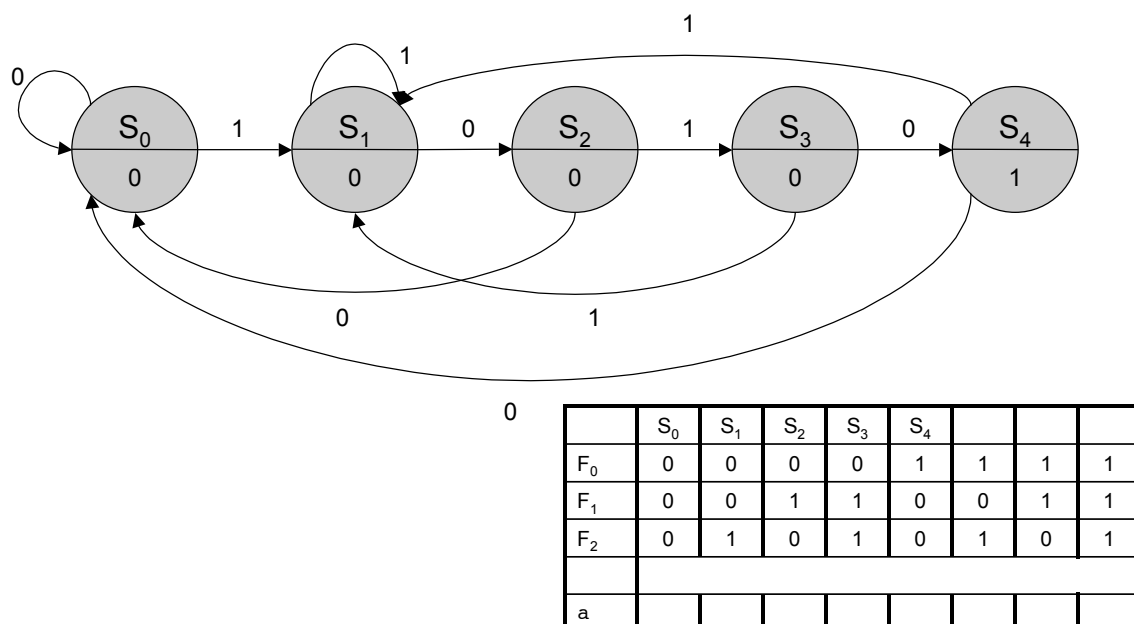
148

Schaltwerk Beispiel

- Ausgabefunktion des Moore-Schaltwerks

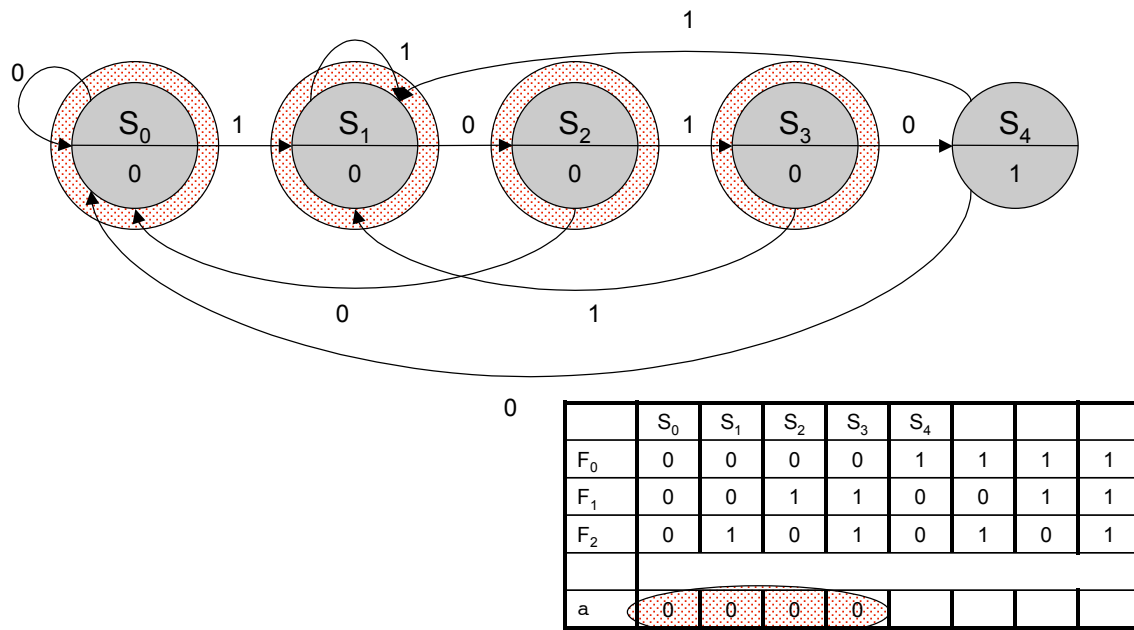
	S ₀	S ₁	S ₂	S ₃	S ₄			
F ₀	0	0	0	0	1	1	1	1
F ₁	0	0	1	1	0	0	1	1
F ₂	0	1	0	1	0	1	0	1
a								

Schaltwerk Beispiel



Schaltwerk Beispiel

Automation Systems Group

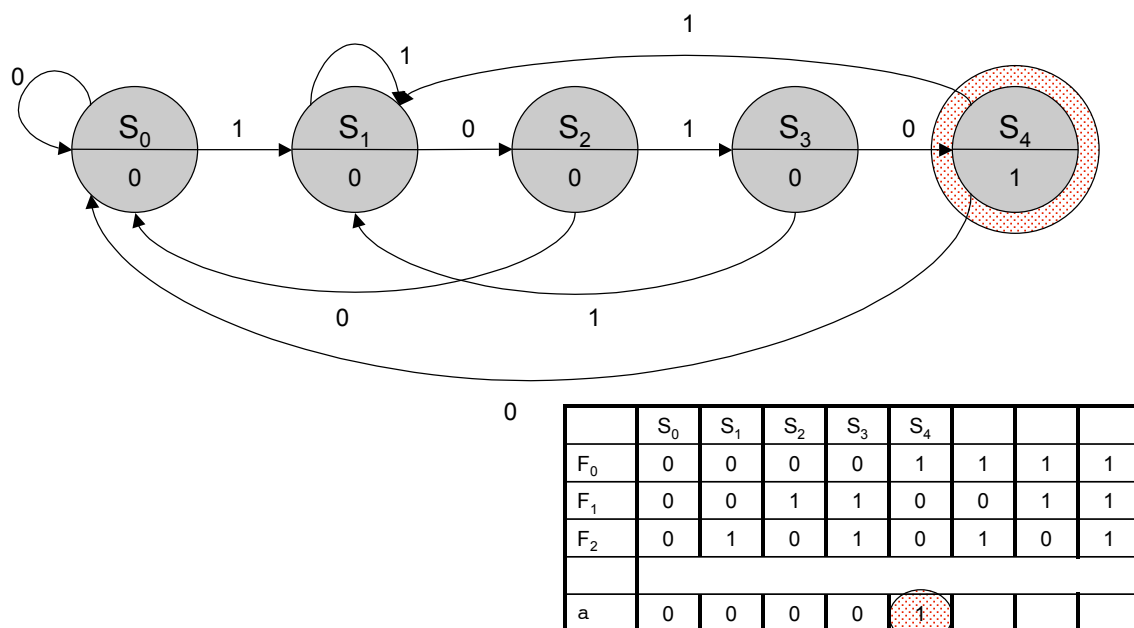


Einführung in die technische Informatik

151

Schaltwerk Beispiel

Automation Systems Group

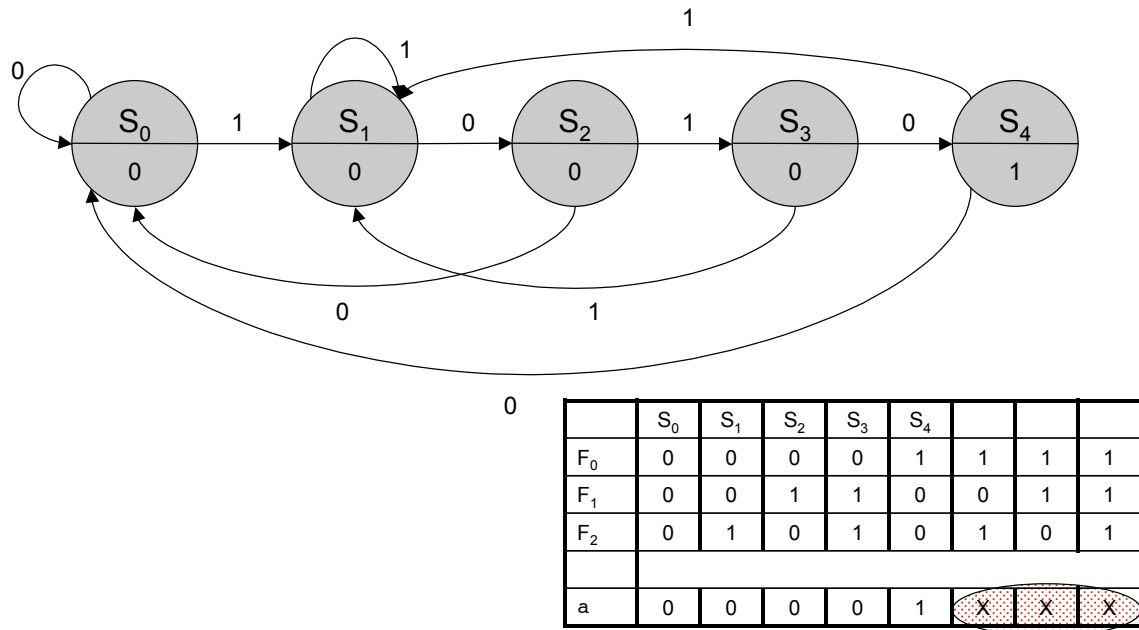


Einführung in die technische Informatik

152

Schaltwerk Beispiel

Automation Systems Group



Einführung in die technische Informatik

153

Schaltwerk Beispiel

Automation Systems Group

- Übergangsfunktion ist identisch mit F_0
 - keine weitere Minimierung erforderlich

	S_0	S_1	S_2	S_3	S_4			
F_0	0	0	0	0	1	1	1	1
F_1	0	0	1	1	0	0	1	1
F_2	0	1	0	1	0	1	0	1
a	0	0	0	0	1	X	X	X

- Übergangsfunktion und Ausgabefunktion können realisiert werden als
 - logische Schaltung (Gatter)
 - PLA
 - ROM

Einführung in die technische Informatik

154

Schaltwerk Beispiel

- Berechnung des maximalen Takts

Flip-Flops

Durchlaufzeit	37 ns
Setup-Time	12 ns
Hold-Time	3 ns
Maximaler Takt	25 MHz

PLA

Durchlaufzeit	22 ns
---------------	-------

$$f_{\max} = \frac{1}{T_{\min}} = \frac{1}{(37 + 22 + 12) \cdot 10^{-09}} \approx 14.08 \cdot 10^6 \approx 14 \text{ MHz}$$

Zusammenfassung

- Schaltwerke
 - logische Schaltungen mit Zustand
 - Ausgang hängt vom Eingang und vom Zustand ab
- Moore-Schaltwerk
 - Ausgabefunktion wird nur vom Zustand bestimmt
- Mealy-Schaltwerk
 - Eingang kann direkt auf Ausgabefunktion wirken

Zusammenfassung

- Schritte zur Realisierung eines Schaltwerks
 - Verstehen der Aufgabenstellung
 - Aufbau des passenden Zustandsgraphen
 - Festlegen der Zustandscodierung
 - Übergangsfunktion bestimmen
 - Ausgabefunktion bestimmen
 - Berechnen der Taktfrequenz
- Zustandsgraph
 - endlicher, deterministischer Automat
 - Knoten repräsentieren Zustände
 - Kanten repräsentieren Übergänge