

Digitaltechnik

Wintersemester 2021/2022

9. Übung



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Prof. Dr.-Ing. Thomas Schneider, M.Sc. Daniel Günther, M.Sc. Amos Treiber
LÖSUNGSVORSCHLAG

KW02

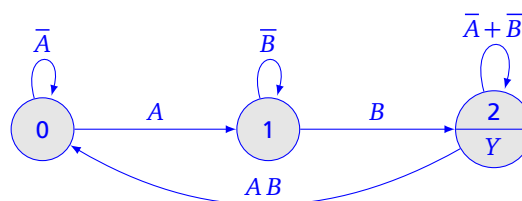
Bitte bearbeiten Sie die Übungsblätter bereits im Voraus, sodass Sie Ihre Lösungen zusammen mit Ihren Kommilitonen und Tutoren während der wöchentlichen Übungsstunde diskutieren können.
Mit der angegebenen Bearbeitungszeit für die einzelnen Aufgaben können Sie Ihren Leistungsstand besser einschätzen.

Übung 9.1 Realisierung endlicher Automaten EX10-C-1 EX10-C-2 EX10-C-3 [25 min]

Gegeben ist eine Zustandsübergangs- und Ausgangstabelle eines endlichen Automaten:

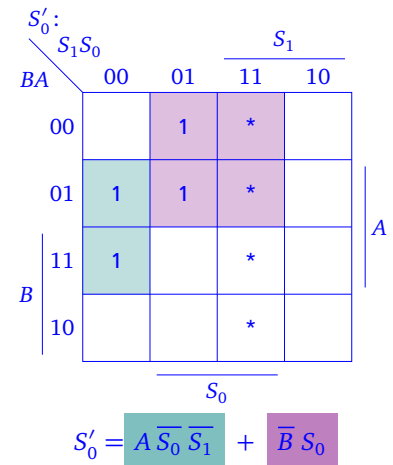
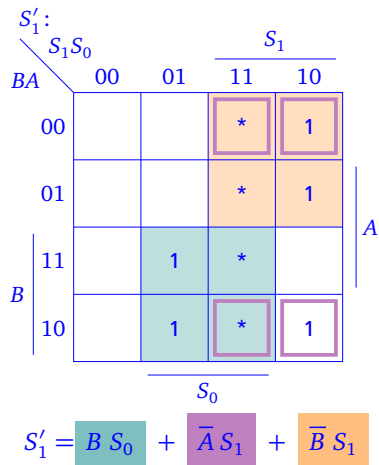
Zustand		Eingänge		Nächster Zustand		Zustand		Ausgang
S_1	S_0	B	A	S'_1	S'_0	S_1	S_0	Y
0	0	*	0	0	0	0	0	0
0	0	*	1	0	1	0	1	0
0	1	0	*	0	1	1	0	1
0	1	1	*	1	0			
1	0	*	0	1	0			
1	0	0	*	1	0			
1	0	1	1	0	0			

- a) Welche Art von Automat wird durch die Tabellen beschrieben? Wie sind die Zustände kodiert?
Es handelt sich um einen Moore-Automat, da das Ausgangssignal Y ausschließlich vom aktuellen Zustand bestimmt wird. Die Zustände sind "durchnummeriert", sie entsprechen den (vorzeichenlosen) 2-Bit-Zahlen 0, 1 und 2.
- b) Zeichnen Sie das zugehörige FSM Diagramm.

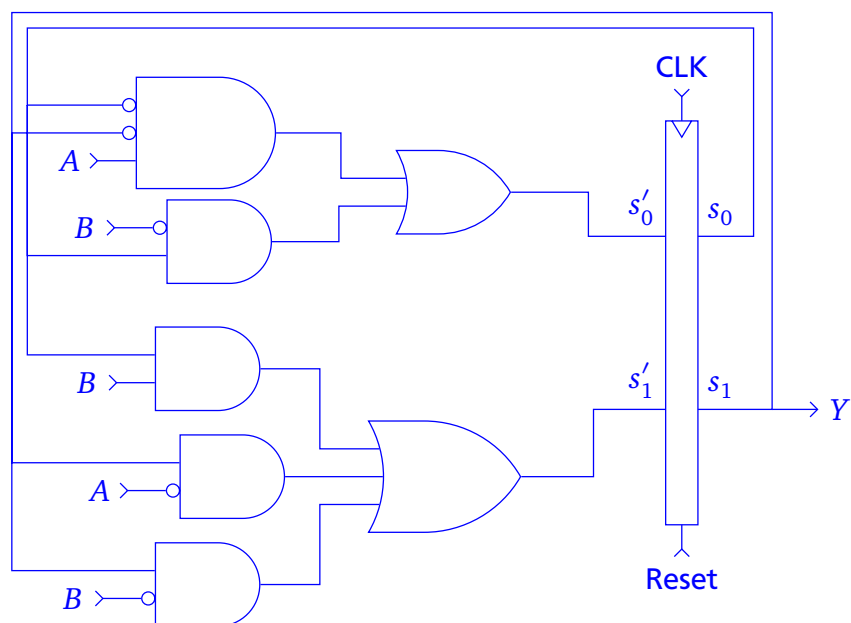


- c) Setzen Sie den Automaten als synchrone sequentielle Schaltung um. Nutzen Sie dafür kombinatorische Logik und D-Flip-Flops für die Zustände. Es bietet sich an, die Zustandsübergangsfunktionen zuerst zu minimieren.
Vereinfache zunächst die Zustandsübergangsfunktionen für S'_1 und S'_0 mittels KV Diagramm. Die nicht genutzte Zustandskodierung 11 kann dabei als "Don't Care" angesehen werden:

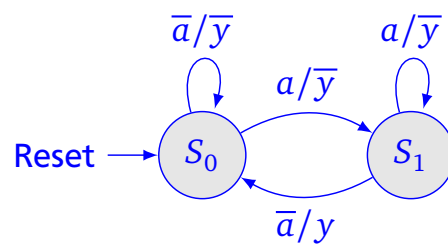
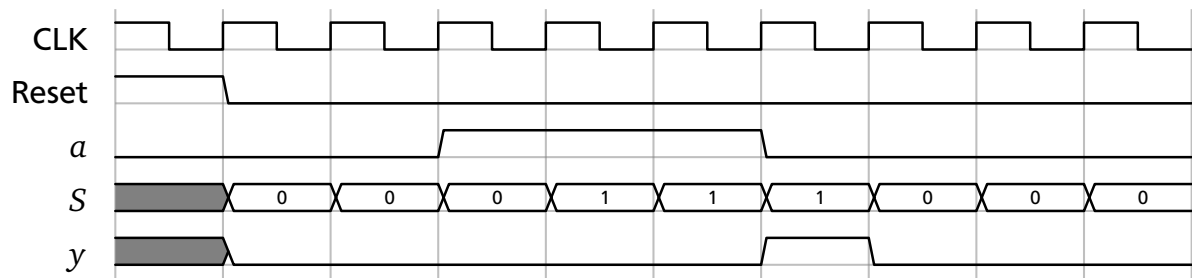
Zustand		Eingänge		Nächster Zustand	
S_1	S_0	B	A	S'_1	S'_0
0	0	*	0	0	0
0	0	*	1	0	1
0	1	0	*	0	1
0	1	1	*	1	0
1	0	*	0	1	0
1	0	0	*	1	0
1	0	1	1	0	0
1	1	*	*	*	*



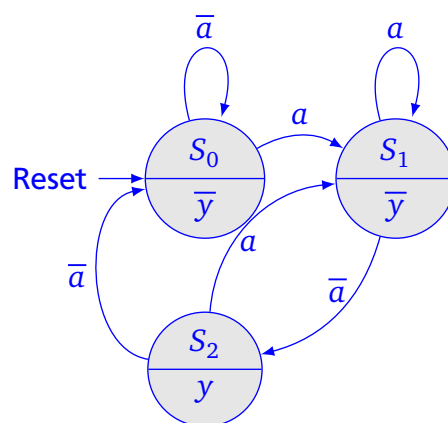
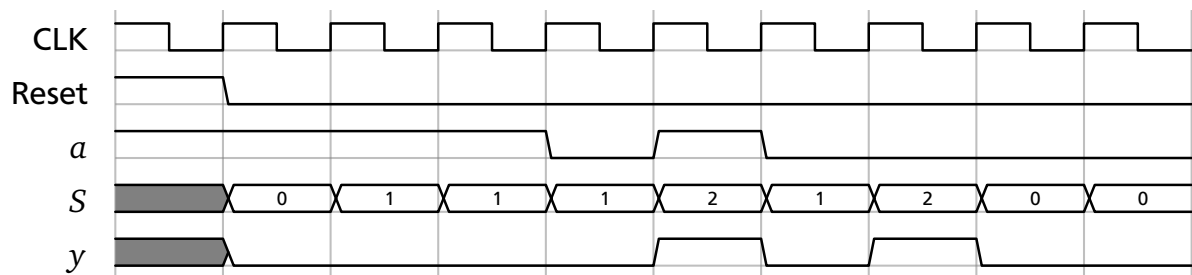
Laut Ausgangstabelle gilt $Y = S_1$. Anders als bei einem Mealy-Automat muss die Ausgangslogik die Eingänge nicht gesondert berücksichtigen. Zur Realisierung des Zustands werden 2 Flip-Flops als Zustandsregister benötigt.



- a) Zeichnen Sie aus dem gegebenen Timing-Diagramm das zugehörige FSM-Diagramm eines Mealy-Automaten.

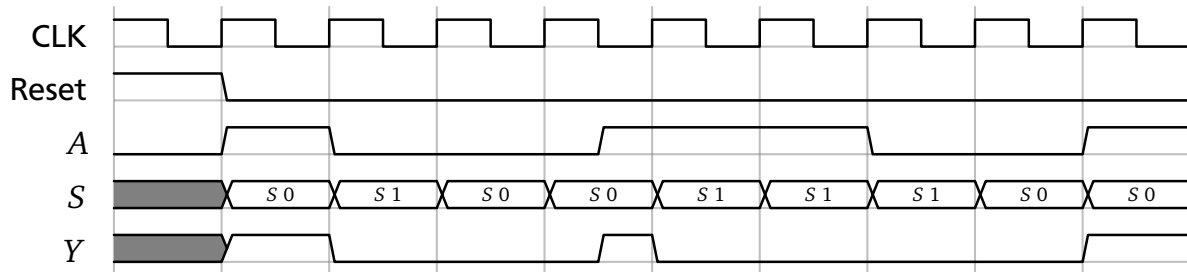


- b) Zeichnen Sie aus dem gegebenen Timing-Diagramm das zugehörige FSM-Diagramm eines Moore-Automaten.



Übung 9.3.1 Analyse eines Timing-Diagramms

Das folgende Timing-Diagramm zeigt das Verhalten eines Zustandsautomaten unbekannten Typs mit dem Eingang A , einem Ausgang Y und dem Zustand S :



- a) Handelt es sich um einen Mealy- oder Moore-Automaten? Woran haben sie dies erkannt?

Es handelt sich um einen Mealy-Automaten. Der Automat verändert innerhalb eines Taktes die Ausgabe und in Takten mit Zustand $S = 0$ erfolgt nicht immer die gleiche Ausgabe. Dies ist bei Moore-Automaten nicht möglich.

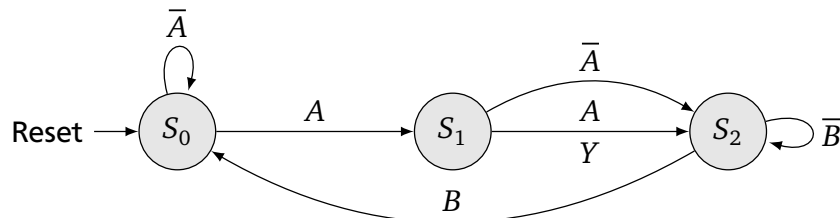
- b) Beschreiben sie die Unterschiede zwischen beiden Automatentypen.

Die Ausgänge eines Mealy-Automaten sind vom Zustand und den Eingängen abhängig. Daher werden die Ausgänge an den Kanten notiert. Daher können sich die Ausgänge von Mealy-Automaten innerhalb eines Taktes mehrmals ändern falls sich Eingangssignale ändern. Mealy-Automaten benötigen somit meist weniger Zustände als Moore-Automaten und reagieren schneller auf Änderungen der Eingangssignale.

Die Ausgänge bei Moore-Automaten sind hingegen ausschließlich vom aktuellen Zustand abhängig und werden daher auch direkt an den Knoten notiert. Moore-Automaten sind besonders geeignet, wenn viele statische Ausgaben erfolgen sollen, z.B. bei einer Ampelschaltung.

Übung 9.3.2 Automat umwandeln EX10-C-4

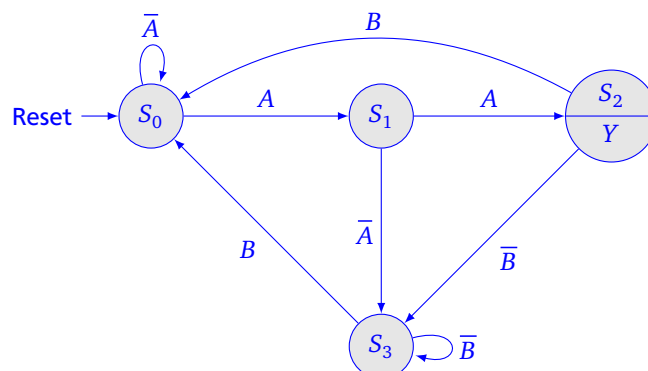
Gegeben ist folgender endlicher Automat mit den Eingängen A und B , sowie einem Ausgang Y :



- a) Handelt es sich um einen Moore- oder einen Mealy-Automat?

Es handelt sich um einen Mealy-Automat, da die Ausgabe Y an einer Kante annotiert ist.

- b) Entwerfen Sie einen äquivalenten Automaten des anderen Typs.



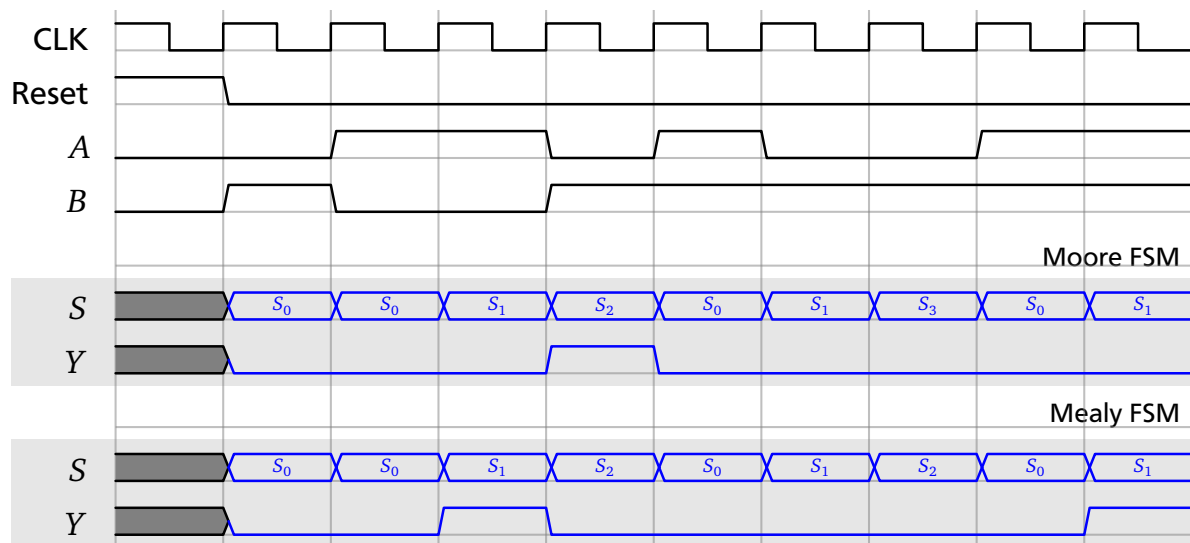
Die Überführung eines Mealy- in einen Moore-Automaten ist meist komplexer als umgekehrt, da ein Mealy-Automat in der Regel weniger Zustände als ein äquivalenter Moore-Automat besitzt.

Allgemeines Vorgehen:

- Alle Ausgabesymbole der Kanten (Zustandsübergänge) in ihren Zielzustand übertragen
- Knoten (Zustände) mit mehreren unterschiedlichen Ausgabesymbolen auf dessen eingehenden Kanten vervielfältigen (inklusive ausgehender Kanten), bis jeder Knoten genau eine Ausgabe hat

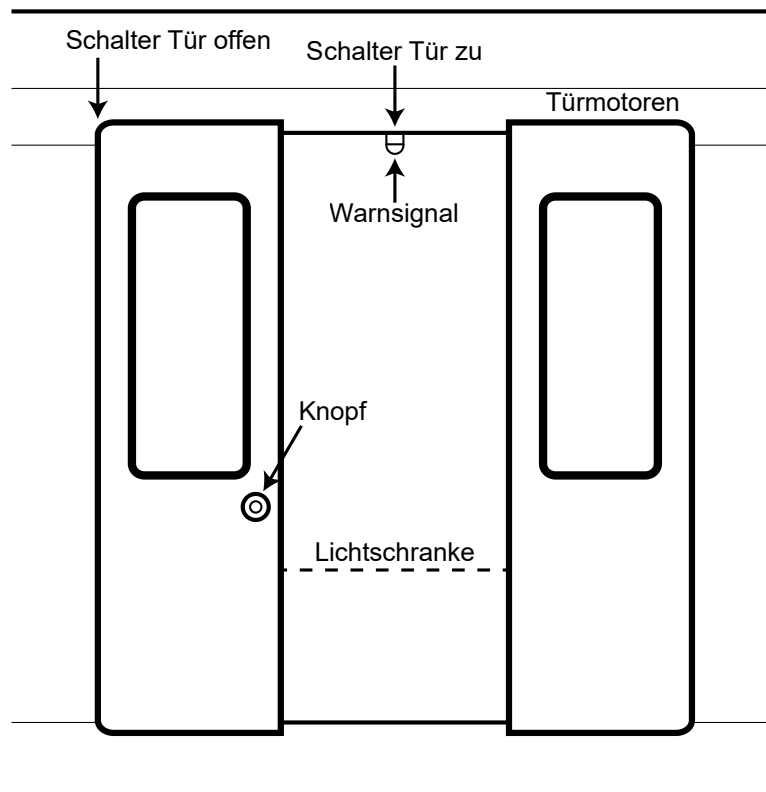
In diesem Fall muss zudem noch beachtet werden, dass der Zustand S_2 im Moore-Automat nach einem Takt auch bei der Eingabe \bar{B} verlassen werden muss, da sonst konstant Y ausgegeben wird, bis B gilt. Dies würde nicht dem Verhalten des Mealy-Automaten entsprechen.

c) Vergleichen Sie das zeitliche Verhalten der beiden Automaten im folgenden Timing-Diagramm:



Der Ausgang des Mealy-Automaten reagiert bereits jeweils einen Takt früher als der des Moore-Automaten.

Die folgende Abbildung zeigt eine übliche Zugtür:



Die Steuerung der Zugtür soll folgendermaßen umgesetzt werden:

Im Startzustand ist die Tür gesperrt. Hält der Zug, so kann der Zugführer die Tür freigeben, indem er ein Freigabesignal (**F**) sendet. Ist die Tür freigegeben, aber noch geschlossen, so soll die grüne Beleuchtung des Knopfs an der Tür (**B**) aufleuchten. Ein Drücken des Knopfes (**K**) durch einen Passagier leitet dann das Öffnen der Tür ein. Um die Tür zu öffnen, muss an den Motoren ein entsprechendes Signal (**M1**) anliegen, bis ein Schalter (**T1**) signalisiert, dass die Tür vollständig offen ist. Zur Vereinfachung können Sie annehmen, dass die Tür nicht automatisch wieder schließt und eine Freigabe der Tür nur einmal pro Halt stattfindet.

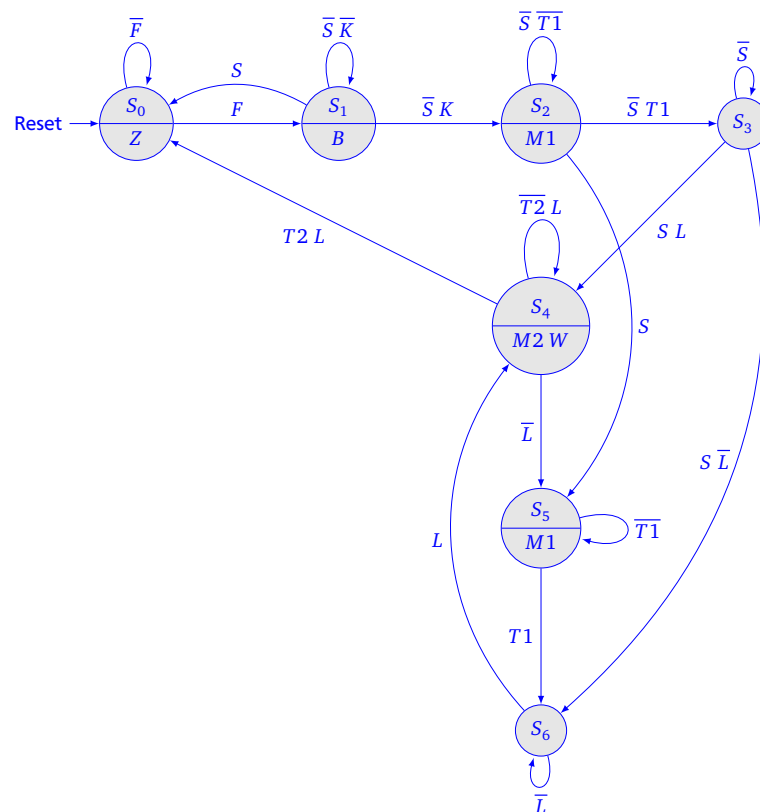
Ist die Tür freigegeben, kann der Zugführer sie jederzeit wieder sperren, indem er ein Sperrsignal (**S**) sendet. Ist die Tür noch geschlossen, so lässt sich diese nun nicht mehr öffnen. Das Schließen der Tür geschieht ähnlich wie das Öffnen durch das Anlegen eines Signals (**M2**) an die Türmotoren, bis ein Schalter (**T2**) signalisiert, dass die Tür zu ist. Ist die Tür gerade noch dabei sich zu öffnen, so wird sie erst nach dem vollständigen Öffnen geschlossen.

Aus Sicherheitsgründen soll die Tür ein Warnsignal während des Schließens ausgeben (**W**). Außerdem ist eine Lichtschranke (**L**) installiert, die solange eine 1 ausgibt, wie sie nicht unterbrochen wird. Die Tür darf nur geschlossen werden, wenn die Lichtschranke nicht unterbrochen wird, ebenso muss ein Unterbrechen der Lichtschranke während eines Schließvorgangs zum sofortigen Öffnen der Tür führen.

Ist die Tür gesperrt und geschlossen, soll dem Zugführer ein entsprechendes Signal gesendet werden (**Z**).

- Entwerfen Sie das FSM-Diagramm eines Moore-Automaten, der die Türsteuerung umsetzt.
- Geben Sie die Zustandsübergangs- und die Ausgabetablelle an. Sie müssen die Zustände nicht kodieren.

a) Lösung:



b) Lösung:

Zustand S_i	Eingänge						Nächster Zustand S'_i
	F	S	K	L	$T1$	$T2$	
S_0	0	*	*	*	*	*	S_0
S_0	1	*	*	*	*	*	S_1
S_1	*	0	0	*	*	*	S_1
S_1	*	0	1	*	*	*	S_2
S_1	*	1	*	*	*	*	S_0
S_2	*	0	*	*	0	*	S_2
S_2	*	0	*	*	1	*	S_3
S_2	*	1	*	*	*	*	S_5
S_3	*	0	*	*	*	*	S_3
S_3	*	1	*	0	*	*	S_6
S_3	*	1	*	1	*	*	S_4
S_4	*	*	*	0	*	*	S_5
S_4	*	*	*	1	*	0	S_4
S_4	*	*	*	1	*	1	S_0
S_5	*	*	*	*	0	*	S_5
S_5	*	*	*	*	1	*	S_6
S_6	*	*	*	0	*	*	S_6
S_6	*	*	*	1	*	*	S_4

Zustand S_i	Ausgänge				
	B	$M1$	$M2$	W	Z
S_0	0	0	0	0	1
S_1	1	0	0	0	0
S_2	0	1	0	0	0
S_3	0	0	0	0	0
S_4	0	0	1	1	0
S_5	0	1	0	0	0
S_6	0	0	0	0	0

Gewisse Spezialfälle wie “was passiert in S_4 wenn die Lichtschranke blockiert ist ($L = 0$) und gleichzeitig die Tür in diesem Moment schon geschlossen ist ($T2 = 1$)” könnten auch anders behandelt werden.

Wichtig ist, implizite Selbstschleifen (immer wenn kein Zustandswechsel stattfindet) beim Erstellen der Zustandsübergangstabelle nicht zu vergessen, sofern diese im FSM-Diagramm weggelassen wurden.

Alle anderen Zustands/Eingabe-Kombinationen sind hingegen Don't Cares.

Neben der Kodierung als Binärzahl wird der Zustand von Automaten auch oft als One-Hot-Kodierung umgesetzt.

- a) Wie viele Bits (und damit Speicherelemente) benötigt man für die Kodierung der Zustände eines Automaten mit 8 Zuständen bei einer Kodierung als Binärzahl? Wie viele bei einer One-Hot-Kodierung?

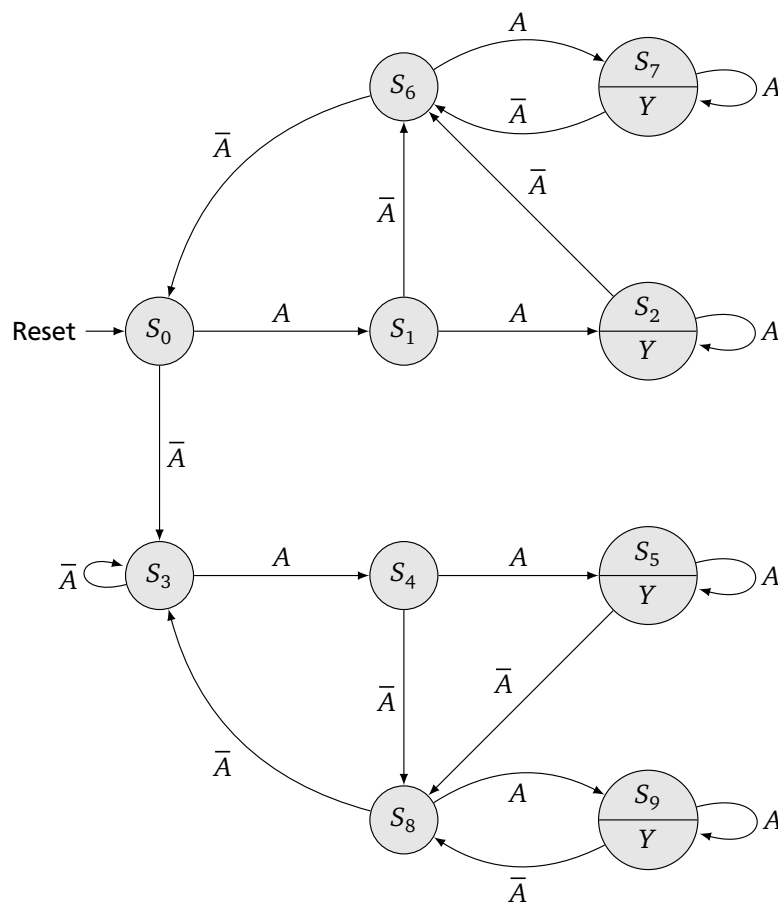
Für die Kodierung von 8 Zuständen benötigt man bei der Kodierung als Binärzahl 3 Bits ($2^3 = 8$), für die One-Hot-Kodierung benötigt man 8 (ein Bit für jeden Zustand).

- b) Welchen großen Nachteil und welchen großen Vorteil hat die One-Hot-Methode gegenüber einer Kodierung als Binärzahl?

Die One-Hot-Kodierung benötigt mehr Speicherelemente, ist jedoch sehr einfach umzusetzen. Daher kann die Zustandsübergangs- und Ausgangslogik oft deutlich einfacher ausfallen.

Übung 9.6 Reduzierung der Anzahl der Zustände – Zusatzaufgabe

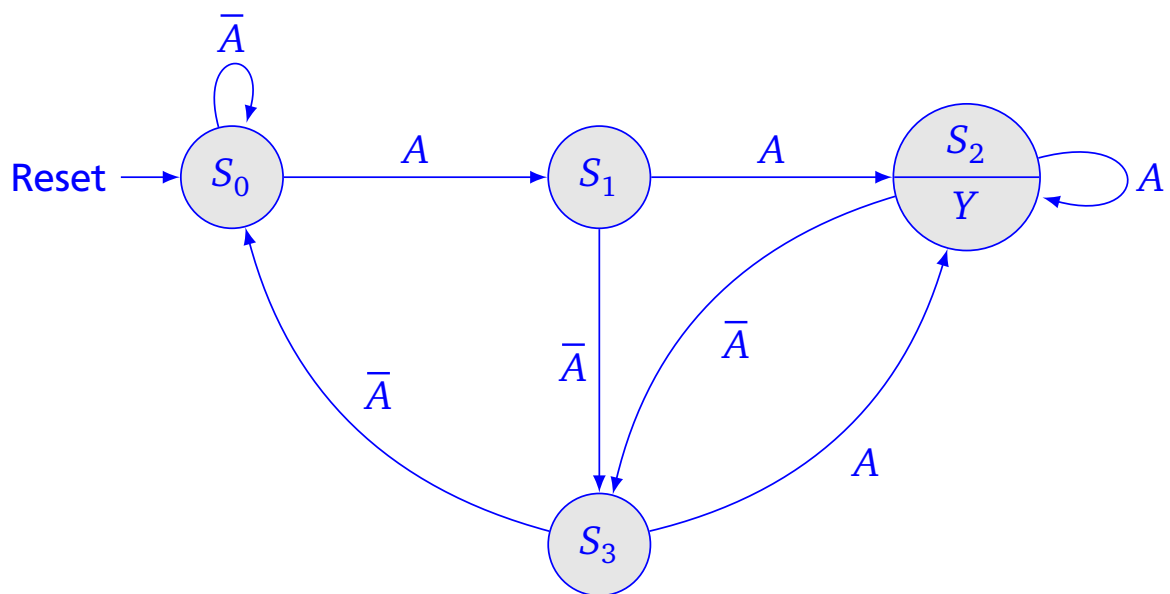
Der folgende Automat besitzt deutlich mehr Zustände als für seine Funktion unbedingt notwendig wären. Geben sie einen Moore-Automaten an, der dieselbe Funktion mit möglichst wenigen Zuständen umsetzt.



Man kann nach folgendem Algorithmus¹ vorgehen, um den Automaten zu minimieren:

- Stelle eine Tabelle aller Zustandspaare z, z' mit $z \neq z'$ auf.
- Markiere alle Paare z, z' bei denen z Endzustand und z' kein Endzustand ist (oder umgekehrt).
- Für jedes unmarkierte Paar z, z' und für jede Eingabe a teste, ob das Paar der Folgezustände bereits markiert ist. Wenn ja, markiere auch z, z' .
- Wiederhole Schritt c), bis sich keine Änderung in der Tabelle mehr ergibt.
- Alle nicht markierten Paare z, z' können jeweils zu einem Zustand verschmolzen werden.

Lösung (4 statt 10 Zuständen):



¹ Quelle: Schönig, Theoretische Informatik – kurzgefasst, 4. Auflage, S.46