Digitaltechnik
Wintersemester 2017/2018
8. Übung

TECHNISCHE UNIVERSITÄT DARMSTADT

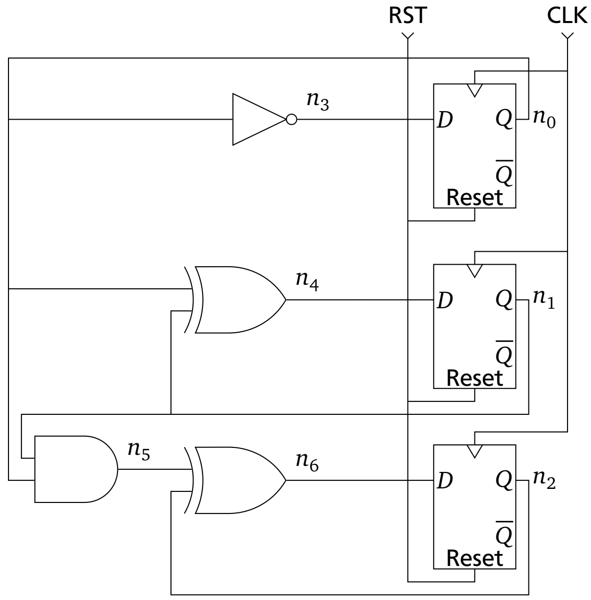
Andreas Engel, Raad Bahmani **LÖSUNGSVORSCHLAG** **KW50**

Die Präsenzübungen werden in Kleingruppen während der wöchentlichen Übungsstunde bearbeitet. Bei Fragen hilft Ihnen Ihr Tutor gerne weiter. Mit der angegebenen Bearbeitungszeit für die einzelnen Aufgaben können Sie Ihren Leistungsstand besser einschätzen.
Die mit **„Zusatzaufgabe"** gekennzeichneten Aufgaben sind zur zusätzlichen Vertiefung für interessierte Studierende gedacht und daher nicht im Zeitumfang von 90 Minuten einkalkuliert.

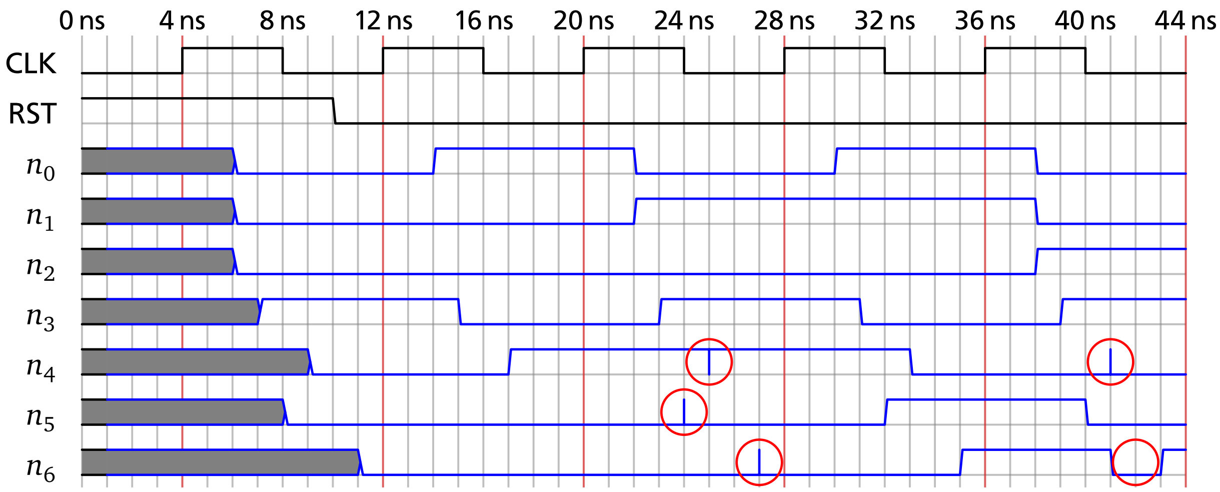
Übung 8.1 Flip-Flops - Wiederholung [15 min]

Gegeben ist folgendes Schaltwerk mit synchron zurücksetzbaren Daten-Flip-Flops (DFF) und den Verzögerungszeiten

$$t_{\text{pd,NOT}} = t_{\text{cd,NOT}} = 1\,\text{ns}, \quad t_{\text{pd,AND}} = t_{\text{cd,AND}} = 2\,\text{ns}, \quad t_{\text{pd,XOR}} = t_{\text{cd,XOR}} = 3\,\text{ns}, \quad \text{und} \quad t_{\text{pd,DFF}} = t_{\text{cd,DFF}} = 2\,\text{ns}$$



a) Vervollständigen Sie folgendes Timing-Diagramm.



b) Welche Funktionalität realisiert dieses Schaltwerk?
3 bit Zähler in $u_3(n_2, n_1, n_0)$

c) Markieren Sie alle Störimpulse im Timing-Diagramm. Müssen diese verhindert werden, um die Korrektheit der Schaltung sicher zu stellen?
Die markierten Störimpulse finden nicht zu den steigenden Taktflanken statt. Sie beeinflussen das nach außen sichtbare Verhalten der Schaltung daher nicht.

d) Ließe sich das gleiche Schaltverhalten auch mit Daten-Latches realisieren?
Nein, denn der Inverter ändert seinen Ausgang noch innerhalb der 1-Phase des Taktes. n_0 bliebe in der Speicher-Phase (0-Phase des Taktes) daher immer 0.

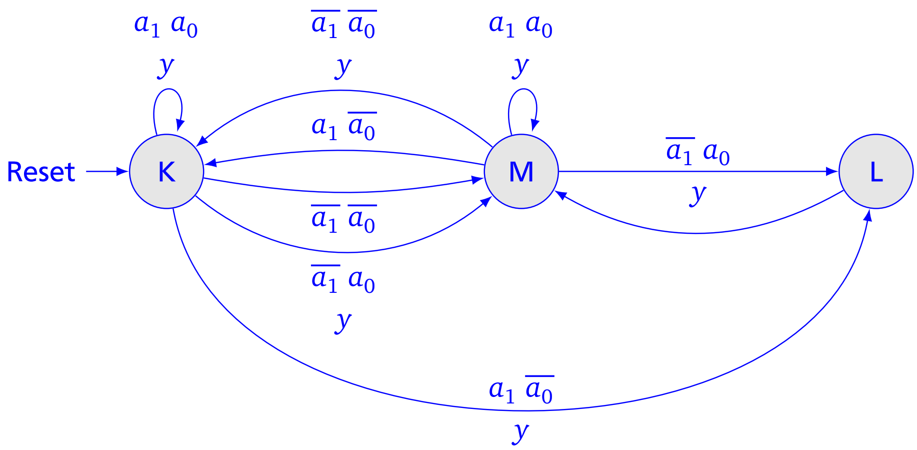
Übung 8.2 Realisierung endlicher Automaten [25 min]

Gegeben ist folgende Zustandsübergangs- und Ausgangstabelle:

S	a_1	a_0	S'
K	0	*	M
K	1	0	L
K	1	1	K
L	*	*	M
M	*	0	K
M	0	1	L
M	1	1	M

S	a_1	a_0	y
K	0	0	0
K	0	1	1
K	1	*	1
L	*	*	0
M	0	*	1
M	1	0	0
M	1	1	1

a) Zeichnen Sie das zugehörige FSM-Diagramm.



b) Zeichnen Sie das zugehörige Schaltwerk. Verwenden Sie dabei eine Zustandskodierung, die dem Zustand mit den meisten eingehenden Kanten die wenigsten Einsen zuordnet. Jede eingehende Kante muss dabei mit der Anzahl der Eingangs-Minterme ihrer Bedingung gewichtet werden. Dies reduziert die Anzahl der Minterme, welche durch die next state logic abgedeckt werden muss.

Für drei Zustände werden zwei Zustandsbits benötigt. Der Zustand M hat vier eingehende Kanten, von denen drei Kanten jeweils einen Eingangs-Minterm repräsentieren. Die vierte Kante (L → M) repräsentiert sogar 4 Minterme. Dem Zustand M wird daher die Codierung „00“ zugeordnert. Die Codierung „11“ sollte möglichst nicht verwendet werden, da dies unnötig viele Einsen erzeugt, die von der next state logic abgedeckt werden müssen. Den beiden verbleibenden Zuständen kann daher die Codierung „01“ und „01“ beliebig zugeordnet werden:

S	eingehendes Kantengewicht	s_1	s_0
K	$1+1+1=3$	0	1
L	$1+1=2$	1	0
M	$1+1+1+4=7$	0	0

Die zu realisierenden Wahrheitswertetabellen erhält man durch einsetzen der Zustandskodierung in die gegebene Zustandsübergangs- und Ausgangstabelle. Die ungenutzte Zustandskodierung „11“ wird als Don’t Care eingetragen:

s_1	s_0	a_1	a_0	s_1'	s_0'
0	1	0	*	0	0
0	1	1	0	1	0
0	1	1	1	0	1
1	0	*	*	0	0
0	0	*	0	0	1
0	0	0	1	1	0
0	0	1	1	0	0
1	1	*	*	*	*

s_1	s_2	a_1	a_0	y
0	1	0	0	0
0	1	0	1	1
0	1	1	*	1
1	0	*	*	0
0	0	0	*	1
0	0	1	0	0
0	0	1	1	1
1	1	*	*	*

Die drei zu realisierenden Funktionen s_1', s_0', y können per Espresso oder Karnaugh-Diagramm minimiert werden:

gen/state.esp

```
.i 4
.o 2
010- 00
0110 10
0111 01
10-- 00
00-0 01
0001 10
0011 00
11-- --
```

gen/state.min.esp

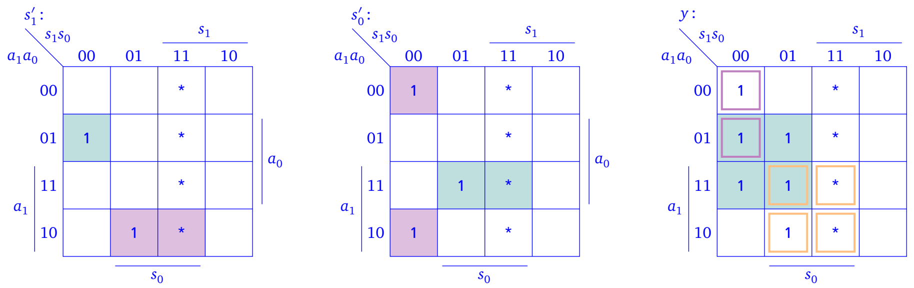
```
.i 4
.o 2
.p 4
0001 10
-110 10
-111 01
00-0 01
.e
```

gen/output.esp

```
.i 4
.o 1
0100 0
0101 1
011- 1
10-- 0
000- 1
0010 0
0011 1
11-- -
```

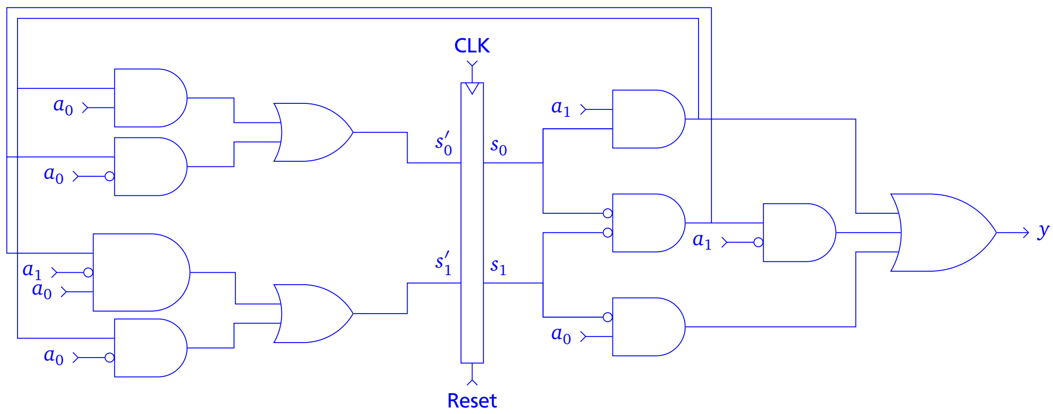
gen/output.min.esp

```
.i 4
.o 1
.p 3
0--1 1
000- 1
-11- 1
.e
```



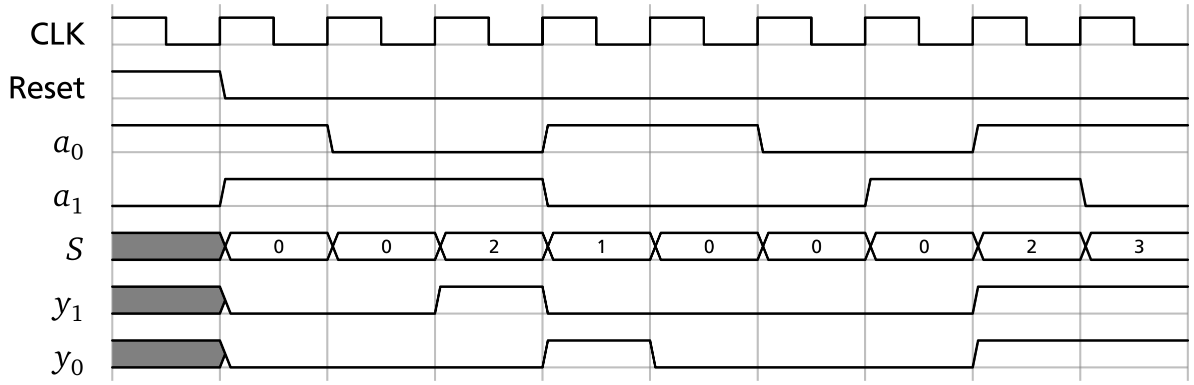
$$s_1' = \overline{s_1}\ \overline{s_0}\ \overline{a_1}\ a_0 + s_0\ a_1\ \overline{a_0}$$
$$s_0' = s_0\ a_1\ a_0 + \overline{s_1}\ \overline{s_0}\ \overline{a_0}$$
$$y = \overline{s_1} a_0 + \overline{s_1}\ \overline{s_0}\ \overline{a_1} + s_0\ a_1$$

Das Schaltwerk, welches die FSM realisiert, benötigt zwei Zustandsregister. Außerdem können die Terme $s_0\ a_1$ und $\overline{s_1}\ \overline{s_0}$ für mehrere Funktionen verwendet werden:



Übung 8.3 Analyse eines Timing-Diagramms [3 min]

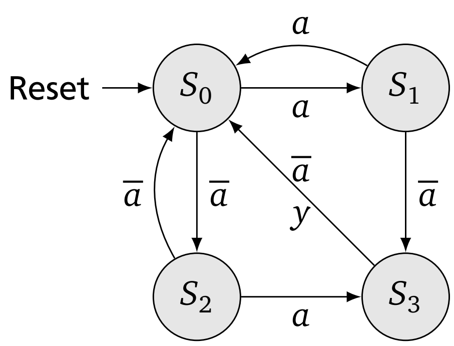
Das folgende Timing-Diagramm beschreibt das Verhalten eines Zustandsautomaten mit zwei Eingängen (a_1, a_0) und zwei Ausgängen (y_1, y_0). Handelt es sich dabei um einen Mealy- oder Moore-Automaten? Begründen Sie Ihre Antwort.



Es handelt sich um einen Mealy-Automaten, da in den beiden Takten mit Zustand S=2 unterschiedliche Ausgaben erzeugt werden.

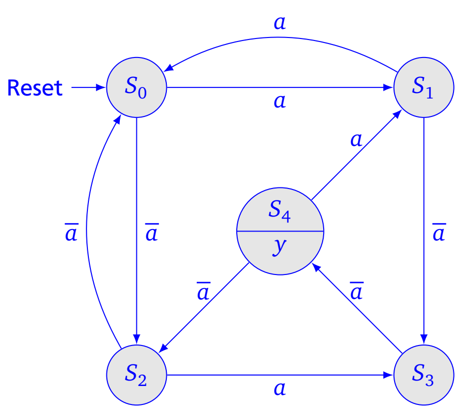
Übung 8.4 Mealy vs. Moore Automat [15 min]

Gegeben ist folgender endlicher Automat mit einem Eingang a und einem Ausgang y:



a) Handelt es sich um einen Moore- oder einen Mealy-Automat? Begründen Sie Ihre Antwort.
Es handelt sich um einen Mealy-Automat, da die Ausgabe an einer Kante annotiert ist.

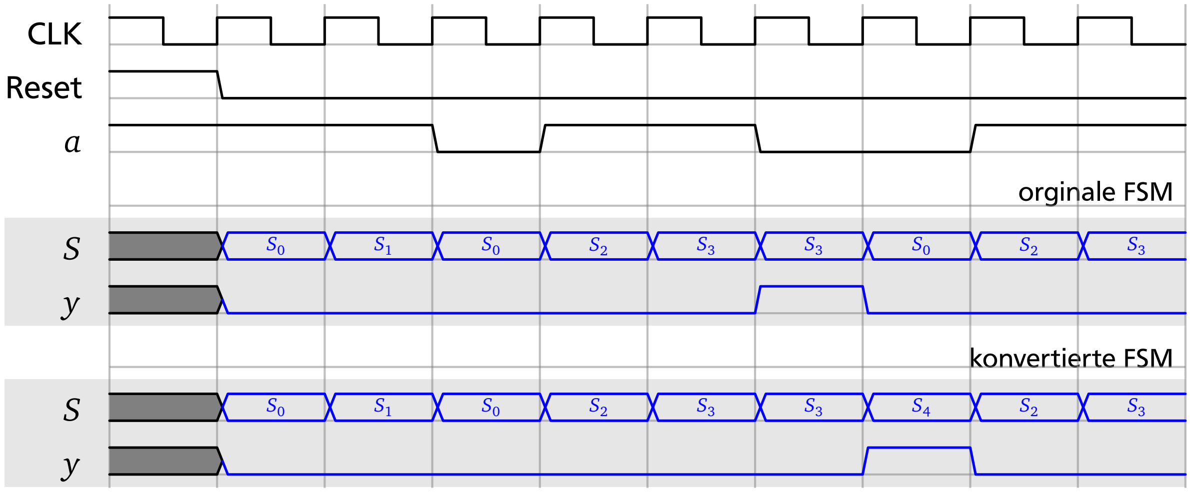
b) Transferieren Sie den gegebenen Automaten in einen äquivalenten Automaten des anderen Typs.



Die Überführung eines Mealy-Automaten in einen Moore-Automaten ist komplexer als umgekehrt, da ein Mealy-Automat idR. weniger Zustände hat als der äquivalente Moore-Automat:

- alle Ausgabesymbole der Zustandsübergänge in ihren Zielzustand übertragen
- Knoten mit mehreren unterschiedlichen Ausgabesymbolen vervielfältigen (inkl. ausgehender Kanten), bis jeder genau eine Ausgabe hat

c) Vergleichen Sie das zeitliche Verhalten der beiden Automaten in folgendem Timing-Diagramm.



Übung 8.5 Entwurf endlicher Automaten	[20 min]

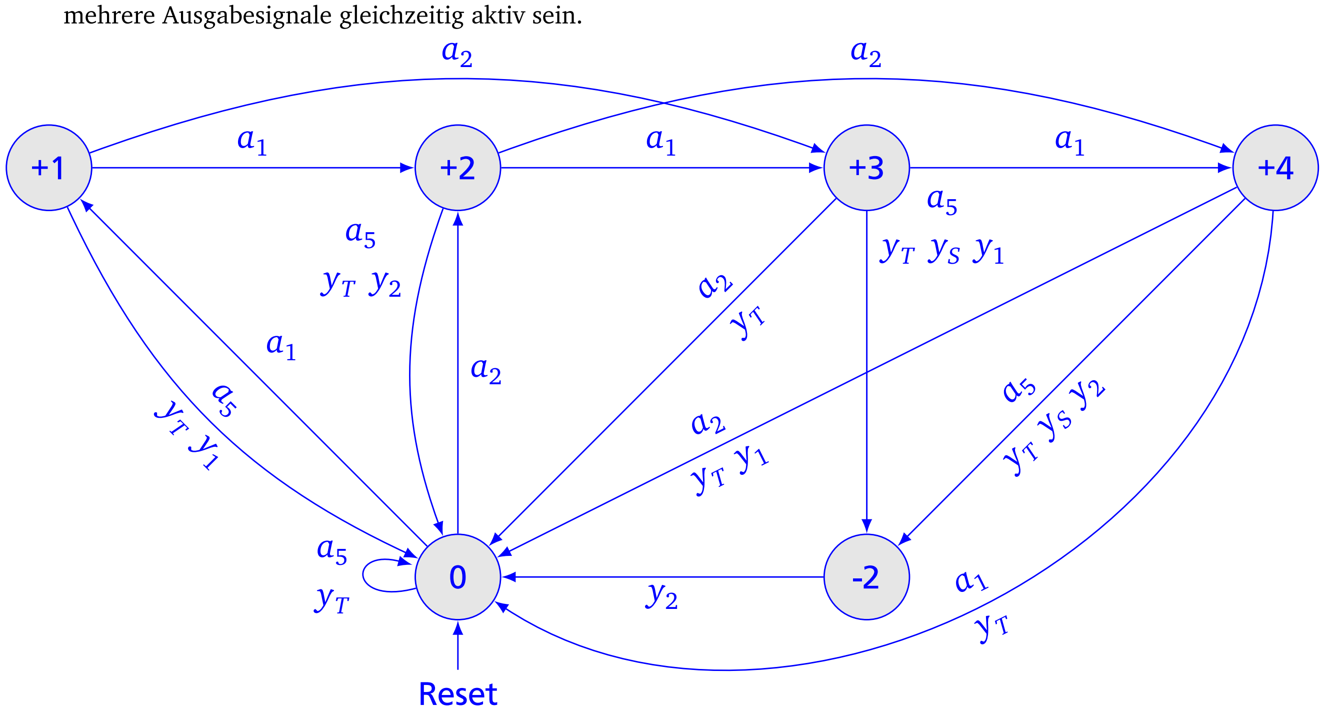
Entwerfen Sie einen endlichen Automaten für einen Fahrkartenverkauf mit Geldeinwurf. Dieser hat drei Ein- und vier Ausgaben mit folgender Spezifikation:

- $a_1 = 1 \Leftarrow$ eine 1 € Münze eingeworfen
- $a_2 = 1 \Leftarrow$ eine 2 € Münze eingeworfen
- $a_5 = 1 \Leftarrow$ ein 5 € Schein eingezogen

- $y_1 = 1 \Rightarrow$ eine 1 € Münze auswerfen
- $y_2 = 1 \Rightarrow$ eine 2 € Münze auswerfen
- $y_T = 1 \Rightarrow$ ein Ticket auswerfen
- $y_S = 1 \Rightarrow$ Geldeingabe sperren

Nach Einwurf von insgesamt 5 € soll das Ticket und ggf. überschüssiges Wechselgeld ausgegeben werden. Solange die Geldrückgabe nicht abgeschlossen ist, muss die Eingabe von weiterem Geld verhindert werden. Für das Auszahlen ist von beiden Münzensorten beliebig viel vorrätig.

a) Entwerfen Sie ein FSM Diagramm (Mealy oder Moore) mit möglichst wenigen Zuständen für diesen Automaten. Sie können dabei annehmen, dass niemals zwei Eingabesignale gleichzeitig aktiv sein können. Es dürfen aber mehrere Ausgabesignale gleichzeitig aktiv sein.



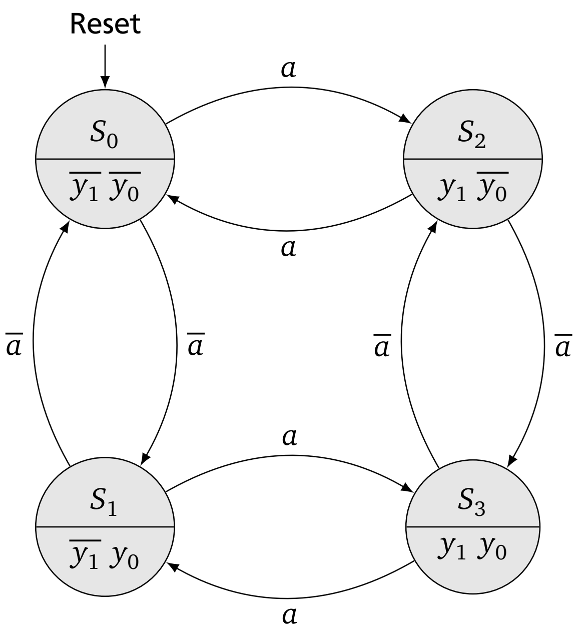
b) Geben Sie die Zustandsübergangs- und Ausgabetabellen (ohne Zustandskodierung) an.

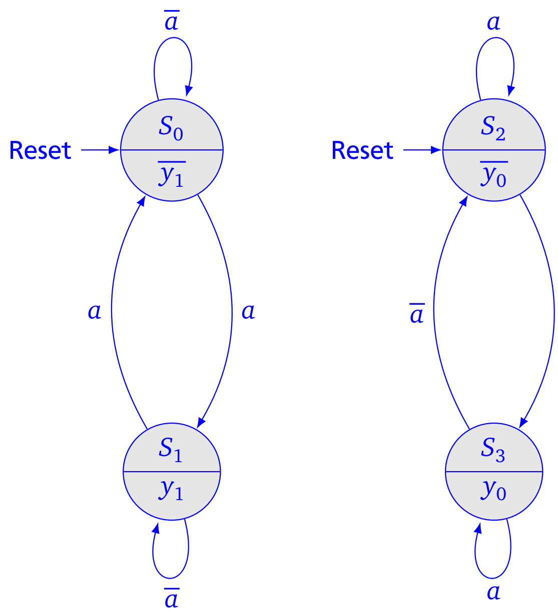
S	a_1	a_2	a_5	S'	y_1	y_2	y_T	y_S
0	0	0	0	0	0	0	0	0
0	1	0	0	+1	0	0	0	0
0	0	1	0	+2	0	0	0	0
0	0	0	1	0	0	0	1	0
+1	0	0	0	+1	0	0	0	0
+1	1	0	0	+2	0	0	0	0
+1	0	1	0	+3	0	0	0	0
+1	0	0	1	0	1	0	1	0
+2	0	0	0	+2	0	0	0	0
+2	1	0	0	+3	0	0	0	0
+2	0	1	0	+4	0	0	0	0
+2	0	0	1	0	0	1	1	0
+3	0	0	0	+3	0	0	0	0
+3	1	0	0	+4	0	0	0	0
+3	0	1	0	0	0	0	1	0
+3	0	0	1	-2	1	0	1	1
+4	0	0	0	+4	0	0	0	0
+4	1	0	0	0	0	0	1	0
+4	0	1	0	0	1	0	1	0
+4	0	0	1	-2	0	1	1	1
-2	0	0	0	0	0	1	0	0

Wichtig ist, die impliziten Selbstschleifen (für $a_1 = a_2 = a_5 = 0$) nicht zu vergessen. Alle anderen Zustands/Eingabe-Kombinationen sind hingegen Don't Cares.

Übung 8.6 Zerlegung endlicher Automaten - **Zusatz**

Zerlegen Sie folgenden Moore-Automaten in zwei unabhängige Moore-Automaten mit jeweils nur einem Ein- und Ausgang.





*