

Mealy-Automaten zur Modellierung von Schaltwerken

Wir haben bisher technische Systeme durch Schaltnetze oder mithilfe von Mealy-Automaten dargestellt. Im Folgenden geht es darum, einen Zusammenhang zwischen einem Mealy-Automaten und einer technischen Realisierung als digitale Schaltung herzustellen. Dazu unterscheiden wir zwischen Mealy-Automaten mit nur einem Zustand und Mealy-Automaten mit mehreren Zuständen.

Beispiel 1: Mealy-Automat mit einem einzigen Zustand und die zugehörige Realisierung als Schaltnetz

Ein sensorgesteuertes Nachtlicht¹ verfügt über zwei Sensoren: einen Helligkeitssensor H und einen Entfernungssensor E. Der Helligkeitssensor gibt den Wert 1 zurück, wenn es hell ist. Der Entfernungssensor gibt den Wert 1 zurück, wenn in einer Entfernung weniger als 2m ein Objekt erkannt wird. Das Nachtlicht soll nur angehen, wenn es dunkel ist und sich jemand in der Nähe befindet.

Dieses technische System lässt sich auf verschiedene Arten modellieren. Vielleicht haben Sie bereits eine Schaltung hierzu realisiert. Wir wollen das Nachtlicht an dieser Stelle zunächst durch einen Mealy-Automaten modellieren. Da wir zwei Sensoren mit ihren jeweiligen Sensorwerten 0 und 1 als Eingaben haben, gibt es vier verschiedene Eingabewerte: „E=0 und H=0“, „E=0 und H=1“, „E=1 und H=0“ sowie „E=1 und H=1“. Das Ausgabealphabet ist $\Omega = \{\text{Nachtlicht an}, \text{Nachtlicht aus}\}$.

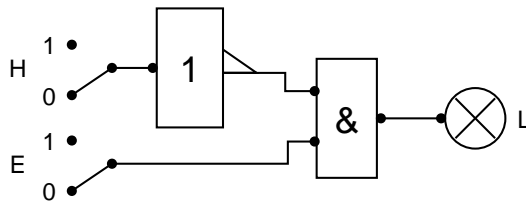
Aufgaben

- 1) Stellen Sie den Mealy-Automaten, der die Steuerung des Nachtlichtes modelliert, in Form eines Zustandsübergangsgraphen dar.
- 2) Die folgende Wahrheitstabelle ist ebenfalls eine Modellierung des Nachtlichtes. Vergleichen Sie diese mit Ihrem Zustandsübergangsgraphen aus Aufgabe 1:

E (Entfernung)	H (Helligkeit)	Ausgabe Nachtlicht
0	0	0
0	1	0
1	0	1
1	1	0

¹ vgl. Aufgabe 10, S. 5, Infsi-Material „Schaltnetze“, Ylva Brandt, Stand November 2020, veröffentlicht unter cc-by-nc-sa 4.0-Lizenz

- 3) Begründen Sie, warum die folgende Schaltung sowohl zur Modellierung des Nachtlichtes mittels eines Mealy-Automaten (vgl. Aufgabe 1) als auch durch eine Wahrheitstabelle (vgl. Aufgabe 2) passt.



Beispiel 2: Mealy-Automaten mit mehreren Zuständen und die Realisierung als Schaltwerk

Für die Steuerung einer Pumpe zur Befüllung eines Wassertanks ist eine digitale Schaltung gesucht. Dazu ist der Wassertank mit zwei Sensoren S1 und S2 ausgestattet (vgl. Abbildung 1). Die Sensoren liefern jeweils eine 1, wenn in ihrer Höhe noch Wasser vorhanden ist, bzw. 0, falls das nicht der Fall ist.

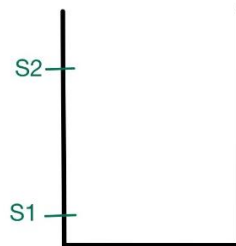


Abbildung 1: Wassertank mit zwei Sensoren

Die Funktionalität der Pumpe wird durch den Mealy-Automaten in Abbildung 2 dargestellt. Das Eingabealphabet lautet dabei $\Sigma = \{ S1=0 \text{ und } S2=0, S1=1 \text{ und } S2=0, S1=1 \text{ und } S2=1 \}$, Ausgabealphabet ist $\Omega = \{ \text{Pumpe an}, \text{Pumpe aus} \}$.

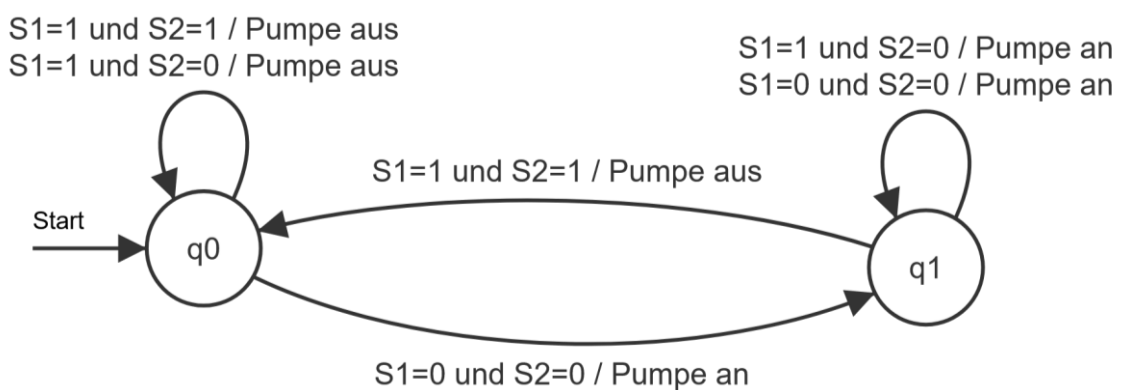


Abbildung 2: Mealy-Automat zur Modellierung einer Pumpe für einen Wassertank

Aufgaben

- 4) Beschreiben Sie, unter welchen Umständen die Pumpe läuft und der Wassertank befüllt wird.
- 5) Erläutern Sie, warum ein Zustand zur Modellierung der Pumpe nicht ausreicht.
- 6) Begründen Sie, warum in der Anwendung eine Eingabe „S1=0 und S2=1“ nicht sinnvoll ist. Diskutieren Sie, wie mit einer solchen unsinnigen Eingabe bei der Modellierung durch einen Mealy-Automaten umgegangen werden könnte, wenn das Eingabealphabet $\Sigma = \{ S1=0 \text{ und } S2=0, S1=1 \text{ und } S2=0, S1=1 \text{ und } S2=1, S1=0 \text{ und } S2=1 \}$ lautet und das Ausgabealphabet mit $\Omega = \{ \text{Pumpe an, Pumpe aus} \}$ beibehalten wird.
- 7) Ergänzen Sie basierend auf dem Zustandsübergangsgraph aus Abbildung 2 die folgende Schaltwerttabelle. Hierin wurde der Startzustand q0 mit 0 und der Zustand q1 mit 1 kodiert. Die Ausgabe „Pumpe an“ wurde entsprechend mit 1, die Ausgabe „Pumpe aus“ mit 0 kodiert. Erläutern Sie die Bedeutung der bereits ausgefüllten Zeilen der Wahrheitstabelle.

S1	S2	Zustand Q	Folgezustand Q	Ausgabe
0	0	0		
0	0	1		
0	1	0	0	0
0	1	1	1	1
1	0	0		
1	0	1		
1	1	0		
1	1	1		

- 8) Begründen Sie, warum in dieser Schaltwerttabelle, anders als in Aufgabe 2 oder allgemein bei Schaltnetzen, auch der aktuelle Zustand als Ein- und Ausgabe berücksichtigt werden muss.

Kodierung der Eingaben, Ausgaben und Zustände

In Aufgabe 7) wurde der aktuelle Zustand durch eine Dualzahl kodiert. Eine solche Kodierung ist notwendig, da wir mittels Flipflops nur Dualzahlen speichern können. Auch die Ein- und Ausgaben des Mealy-Automaten werden binär kodiert, so dass mit allen Angaben zusammen eine Schaltwerttabelle erstellt und anschließend ein Schaltterm abgeleitet werden kann. In der digitalen Schaltung benötigen wir dann neben den aus Schaltnetzen bekannten Standardgattern ein oder mehrere Bauteile zur Speicherung des aktuellen Zustands. Hierfür verwenden wir Flipflops, die Sie bereits als Möglichkeit zur Speicherung eines Bits kennengelernt haben.

Anzahl an Flipflops zur Speicherung des aktuellen Zustandes

Da der Zustandsübergangsgraph für unser Beispiel aus Abbildung 2 nur aus zwei Zuständen besteht, genügt hier ein einzelnes Bit zur Speicherung des Zustands, wir benötigen also nur ein Flipflop. Bei mehr Zuständen müssen diese zunächst binär kodiert werden. Je nach Anzahl der verwendeten Bits benötigt man dann entsprechend viele Flipflops zur Speicherung des aktuellen Zustandes.

Setzen des aktuellen Zustandes

Unser Beispiel der Steuerung der Pumpe eines Wassertanks muss nicht durch einen synchronen Takt gesteuert werden. Deswegen können wir hier zur Speicherung des aktuellen Zustandes ein (leicht modifiziertes) RS-Flipflop verwenden. Der aktuelle Zustand wird durch ein eingehendes Signal gesetzt. Damit auch der R-Eingang des RS-Flipflops korrekt belegt ist führen wir das gleiche Signal negiert auf den R-Eingang:

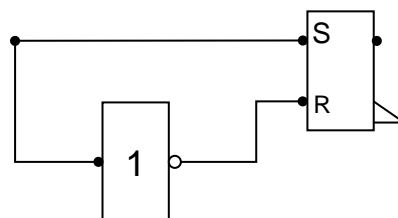
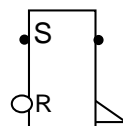


Abbildung 3: Setzen des aktuellen Zustands abhängig von einem Eingangssignal

Für ein derart modifiziertes RS-Flipflop gibt es ein eigenes Schaltsymbol:



Anmerkung: In vielen Anwendungen werden häufig andere Flipflops verwendet, die zusätzlich einen Takt zur Synchronisierung berücksichtigen. Beispiele wären etwa JK-Flipflops oder D-Flipflops in verschiedenen Formen.

Entwicklung eines zugehörigen Schaltwerks unter Verwendung von FlipFlops

Bei Schaltnetzen kann aus einer gegebenen Schaltwerttabelle systematisch die zugehörige Schaltung konstruiert werden. Der einzige Unterschied zwischen unserer Wahrheitstabelle in Aufgabe 7 und Wahrheitstabellen zu Schaltnetzen ist der, dass die Ausgabe nicht mehr nur von der Eingabe, sondern auch vom aktuellen Zustand abhängt (und wir damit auch für jeden Zustand den Folgezustand neu berechnen müssen). Wenn wir aber ein Bauteil wie ein Flipflop zur Speicherung des aktuellen Zustandes haben, können wir dieses in unserem Schaltwerk genauso wie bisher die Eingangswerte verwenden. Daher gehen wir bei der Entwicklung eines Schaltwerkes aus einer Wahrheitstabelle ganz analog zur Entwicklung von Schaltnetzen aus Wahrheitstabellen vor. Schauen wir uns dazu die Lösung von Aufgabe 7 an:

S1	S2	Zustand Q	Folgezustand Q_{neu}	Ausgabe
0	0	0	1	1
0	0	1	1	1
0	1	0	0	0
0	1	1	1	1
1	0	0	0	0
1	0	1	1	1
1	1	0	0	0
1	1	1	0	0

Abbildung 4

Zunächst erkennt man, dass der Folgezustand in diesem Beispiel genauso kodiert wird wie die Ausgabe. Daher konzentrieren wir uns zunächst auf eine Entwicklung einer Schaltung für den Folgezustand Q.

Es gibt viele verschiedene Möglichkeiten, aus dieser Wahrheitstabelle einen gültigen Schaltterm für Q abzuleiten. Wie Sie bereits von den Schaltnetzen wissen, ist es eine gute Strategie, sich auf die Terme mit einer 1 in der Ausgabe bzw. hier im Folgezustand Q zu konzentrieren. Die entsprechenden Zeilen wurden in der Tabelle zur besseren Lesbarkeit bereits hervorgehoben.

Aufgaben:

- 9) Leiten Sie aus der Wahrheitstabelle einen Schaltterm für Q_{neu} in der disjunktiven Normalform her. (Tipp: der erste Minterm lautet $(\overline{S1} \wedge \overline{S2} \wedge \overline{Q})$.)
- 10) Zeigen Sie, dass der Schaltterm $Q_{\text{neu}} = (\overline{S1} \wedge \overline{S2}) \vee (Q \wedge (S1 \oplus S2))$ ebenfalls geeignet ist zur Beschreibung der Wahrheitstabelle aus Abbildung 4. Zur Erinnerung: der Operator \oplus beschreibt die logische Verknüpfung XOR (gleichbedeutend mit „entweder – oder“).

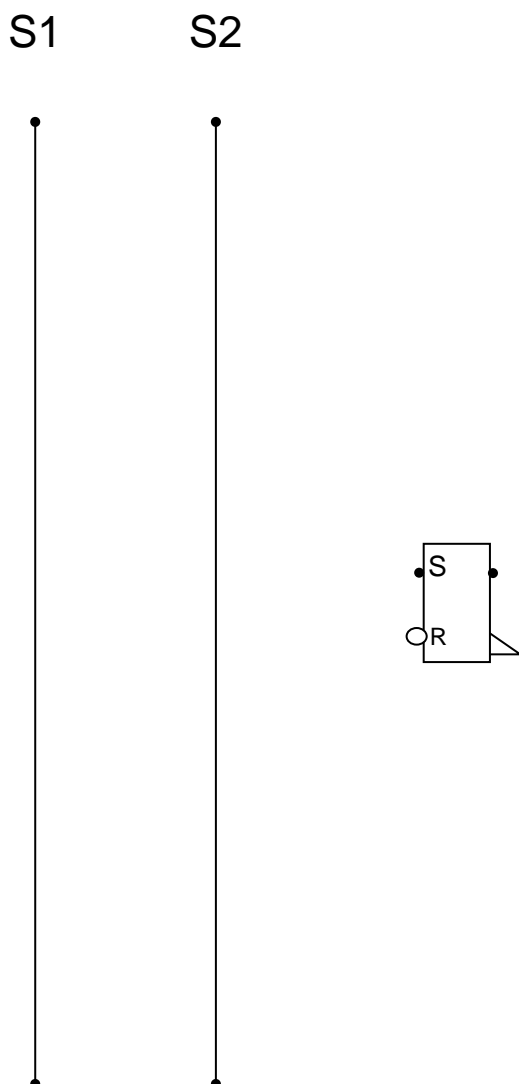
Zeichnen des Gatternetzes

11) Versuchen Sie jetzt, das Gatternetz zum Schaltterm $Q_{neu} = (\overline{S1} \wedge \overline{S2}) \vee (Q \wedge (S1 \oplus S2))$ zu zeichnen. Dazu finden Sie im Folgenden drei verschiedene Vorlagen, aus der Sie auswählen können:

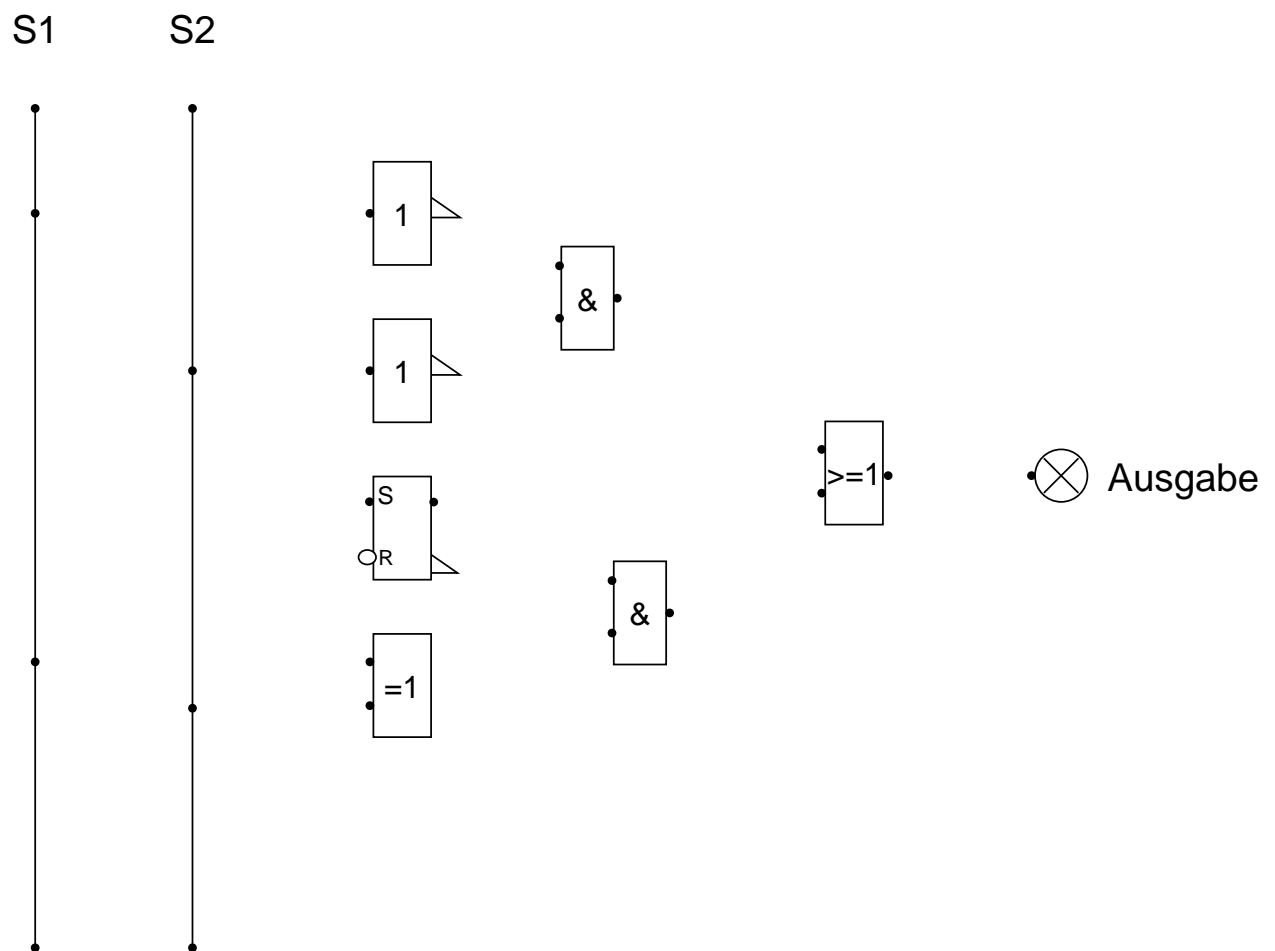
- Vorlage A: mit bereits zur Verfügung gestellten Eingängen S1 und S2 sowie einem RS-Flipflop zur Speicherung des Zustandes Q
- Vorlage B: mit deutlich größerer Vorstrukturierung, in der bereits alle benötigten Gatter eingezeichnet sind
- Vorlage C, in der das Gatternetz bereits fast fertig eingezeichnet ist und nur noch um eine geeignete Rückkopplung zum Aktualisieren des Zustandes Q ergänzt werden muss

Alternativ können Sie auch das fertige Gatternetz (vgl. Abbildung 5) analysieren und mit einer geeigneten Simulationssoftware testen.

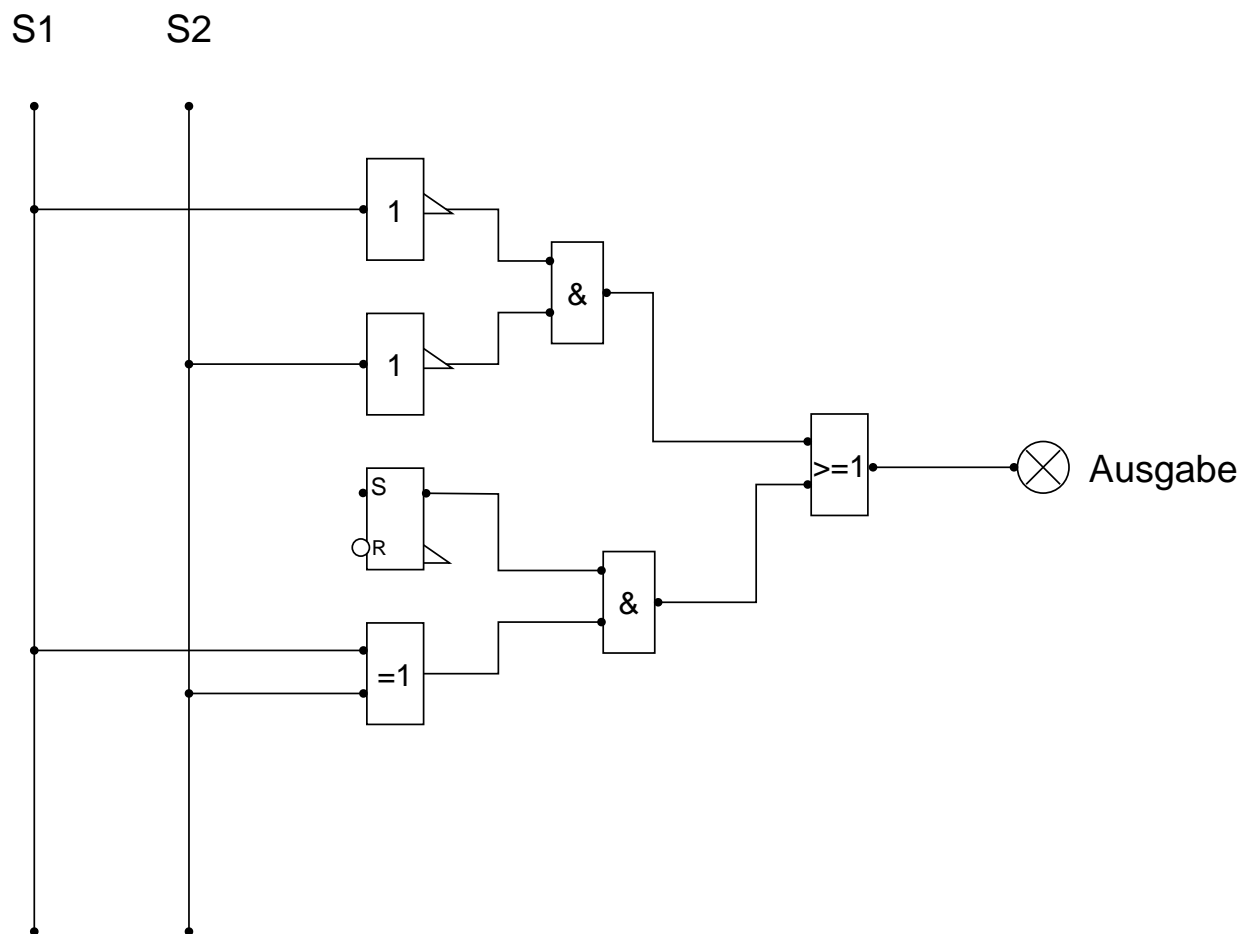
Vorlage A



Vorlage B



Vorlage C



Lösung

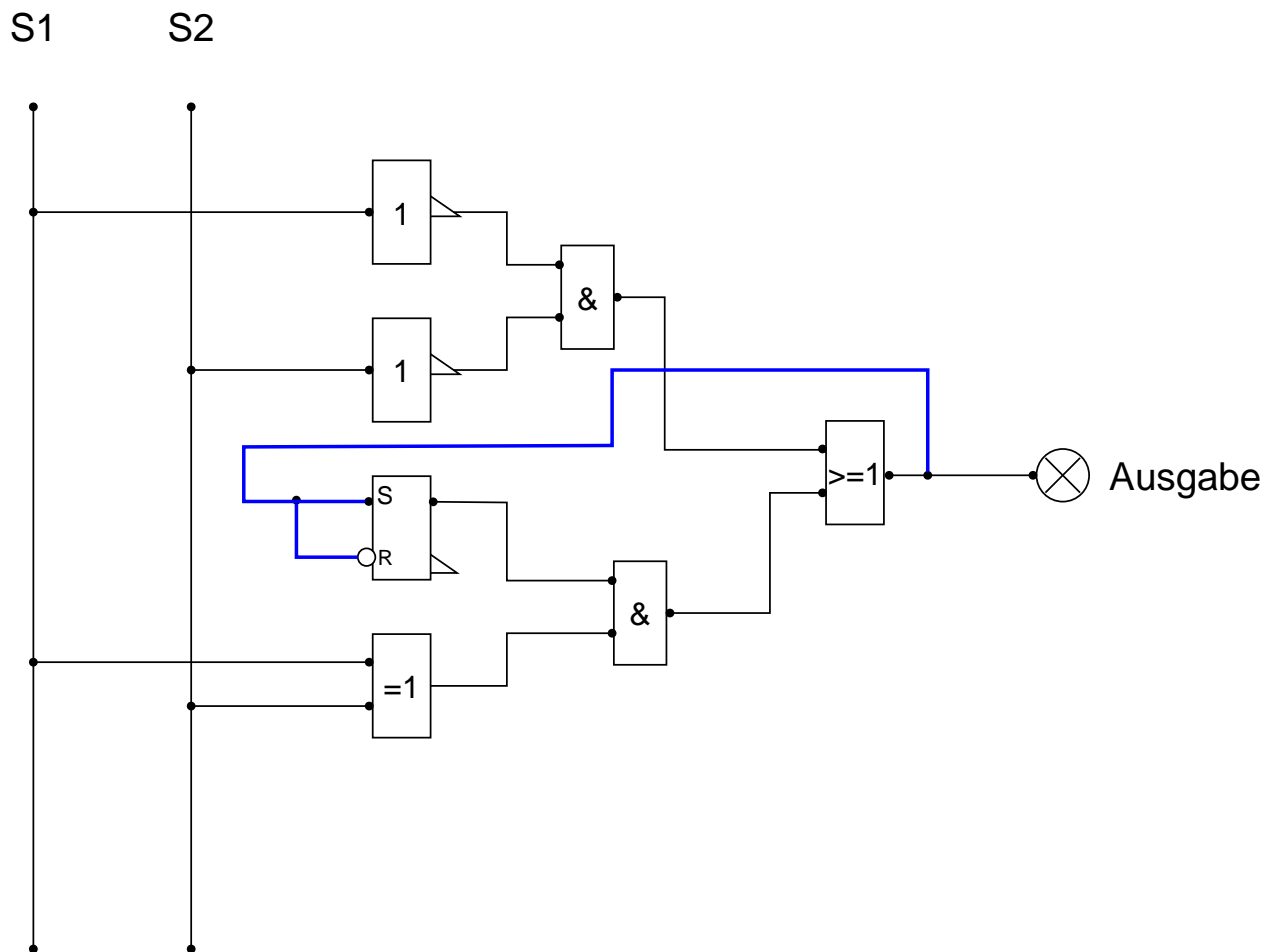


Abbildung 5: fertiges Gatternetz

Zusammenfassung:

Technische Systeme lassen sich gut durch Mealy-Automaten modellieren. Sucht man die zugehörige technische Schaltung, so kann man ähnlich zur Entwicklung von Schaltnetzen vorgehen. Der einfachste Fall, nämlich ein Mealy-Automat mit nur einem Zustand, kann durch eine Übersetzung der Übergangsfunktion in eine Schaltwerttabelle realisiert werden. Wie daraus systematisch eine Schaltung konstruiert werden kann, haben Sie im Leitfaden zu Schaltnetzen gelernt.

Bei Mealy-Automaten mit mehr als einem Zustand hängen die Ausgaben des technischen Systems nicht mehr nur von den aktuellen Eingaben, sondern jeweils auch vom aktuellen Zustand ab. Daher benötigt man bei der technischen Realisierung Flipflops zur Zwischenspeicherung und spricht nicht mehr von einem Schaltnetz, sondern von einem Schaltwerk. Die systematische Entwicklung dieses Schaltwerks besteht dann aus den folgenden Schritten:

1. **Schritt:** Entwicklung des Mealy-Automaten (Angabe des Eingabealphabets, des Ausgabealphabets sowie des Übergangsgraphen) zum gewählten technischen System
2. **Schritt:** Kodierung der Zeichen des Eingabealphabets, der Zeichen des Ausgabealphabets sowie der Zustände, jeweils durch Dualzahlen.
3. **Schritt:** Übersetzung des Zustandsübergangsgraphen in eine Schaltwerttabelle unter Verwendung der Kodierungen aus Schritt 2.
4. **Schritt:** Entwicklung eines Schaltterms aus der Wahrheitstabelle. Dies kann auf verschiedene Arten geschehen: zum Beispiel mittels der disjunktiven Normalform und anschließender systematischer Vereinfachung oder anderen Strategien zur Vereinfachung von Schalttermen
5. **Schritt:** Zeichnen des zugehörigen Gatternetzes

Aufgaben zur Vertiefung

- 12) Mithilfe einer CO₂-Ampel soll die Raumluft eines Raumes überwacht und davon abhängig die Lüftungsanlage ein- oder ausgeschaltet werden. Dabei gibt die CO₂-Ampel drei verschiedene Signale: grün für sehr gute Raumluft, gelb für zu beobachtende Werte und rot für kritische Werte. Damit die Lüftungsanlage nicht ständig an- und ausschaltet, soll sie erst eingeschaltet werden, wenn die CO₂-Ampel „rot“ signalisiert. Dann soll sie aber so lange in Betrieb sein, bis erneut „grüne“ Werte erreicht werden. Entwickeln Sie systematisch ein zugehöriges Schaltwerk.
- 13) (Für Fortgeschrittene) Alternativ zur Modellierung in Abbildung 2 der Wasserpumpe kann die unsinnige Eingabe $S_1=0$ und $S_2=1$ auch in einen Fehlerzustand führen: misst der obere Sensor Wasser, der untere dagegen nicht, soll die Pumpe in einen Fehlerzustand übergehen und sich abschalten. Ein entsprechender Übergangsgraph ist in Abbildung 6 dargestellt. Wandeln Sie diesen Zustandsgraphen systematisch in ein Gatternetz um. Geben Sie dazu geeignete Kodierungen der Zustände an und zeichnen Sie das zugehörige Gatternetz.

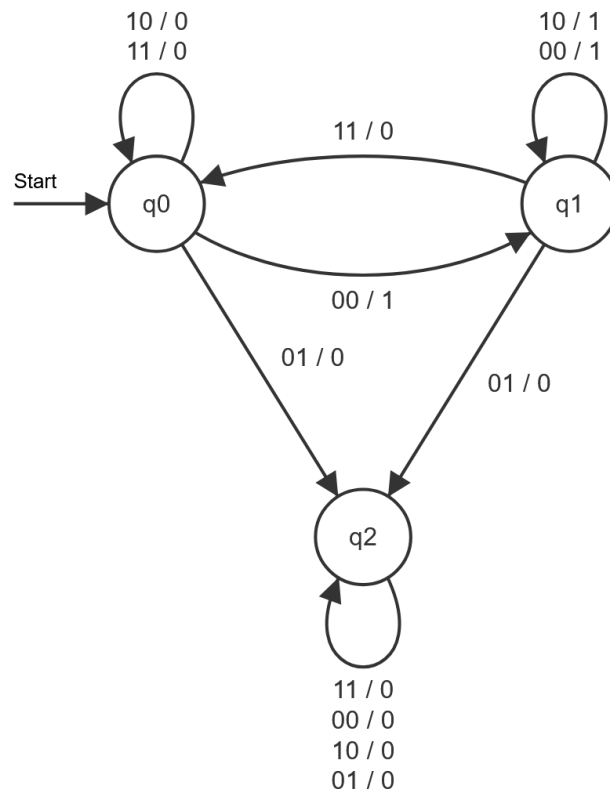


Abbildung 6: Berücksichtigung eines Fehlerzustandes

Dieses Werk ist lizenziert unter einer [Creative Commons Namensnennung - Nicht-kommerziell - Weitergabe unter gleichen Bedingungen 4.0 International Lizenz](https://creativecommons.org/licenses/by-nc-sa/4.0/). Sie erlaubt Bearbeitungen und Weiterverteilung des Werks unter Nennung meines Namens und unter gleichen Bedingungen, jedoch keinerlei kommerzielle Nutzung.