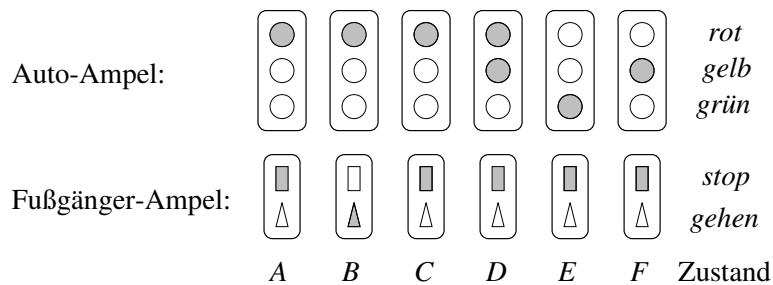


Aufgabe 6.1 Ampel - Schaltwerk (5 Punkte)

Verkehrsampelsteuerungen lassen sich direkt als einfache Schaltwerke realisieren. Die folgende Aufgabe besteht in der Realisierung einer vereinfachten Ampelanlage für einen Fußgängerüberweg mit Bedarfsknopf b . Eine symbolische Zustandskodierung könnte folgendermaßen aussehen:

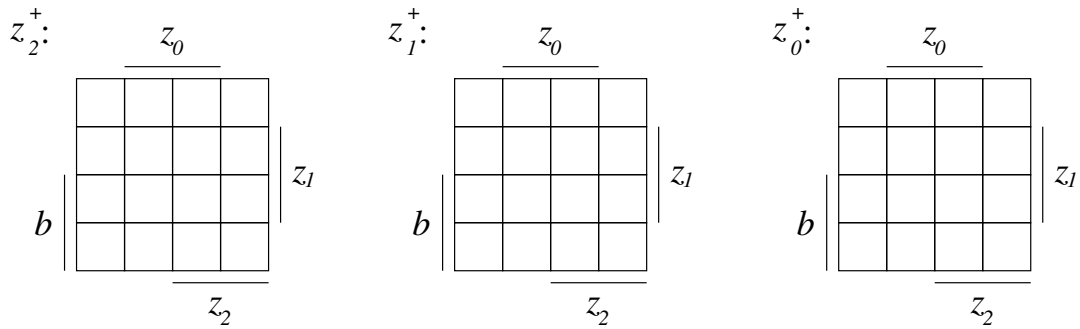


Der Automat wartet solange im Zustand E , bis der Bedarfsknopf gedrückt wird. Wird der Bedarfsknopf gedrückt, so durchläuft der Automat die Zustände F, A, B, C, D, E und wartet wieder im Zustand E .

- a) Übertragen Sie die dargestellten Ampelphasen ausgehend von der Zustandsmenge in einen Zustandsgraphen mit den sechs symbolisch bezeichneten Zuständen (A, \dots, F).
- b) Definieren Sie für die sechs Zustände (A, B, C, D, E, F) eine Zustandskodierung unter Verwendung der Variablen $z_2 z_1 z_0$ und übertragen Sie Ihren Zustandsgraphen in eine Zustandstabelle der folgenden Form:

b	z_2	z_1	z_0	z_2^+	z_1^+	z_0^+	$Kante (z \rightarrow z^+)$
0	0	0	0				
0	0	0	1				
0	1	0	0				
0	1	1	0				
1	0	0	0				
1	0	1	0				
1	1	0	0				
1	1	1	0				

Verwenden Sie zur Minimierung des Übergangsschaltnetzwerks folgende KV-Diagramme:

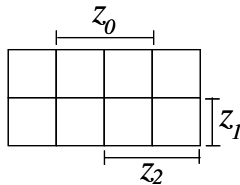


Nutzen Sie, wenn möglich *dont' cares* (*) bei der Minimierung. Geben Sie die minimierten Terme an. Erstellen Sie anschließend eine Wertetabelle für die Ausgangsfunktion in folgender Form:

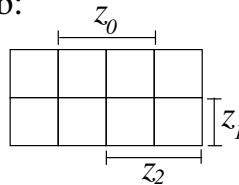
Zustand	z_2	z_1	z_0	rot	gelb	grün	stop	gehen
	0	0	0					
	0	0	1					
	0	1	0					
	0	1	1					
	1	0	0					
	1	0	1					
	1	1	0					
	1	1	1					

und verwenden Sie dazu folgende KV-Diagramme zur Minimierung:

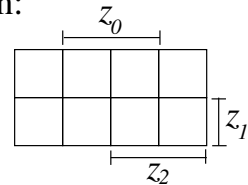
rot:



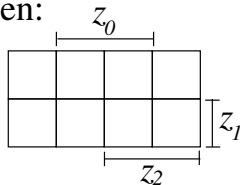
gelb:



grün:



gehen:



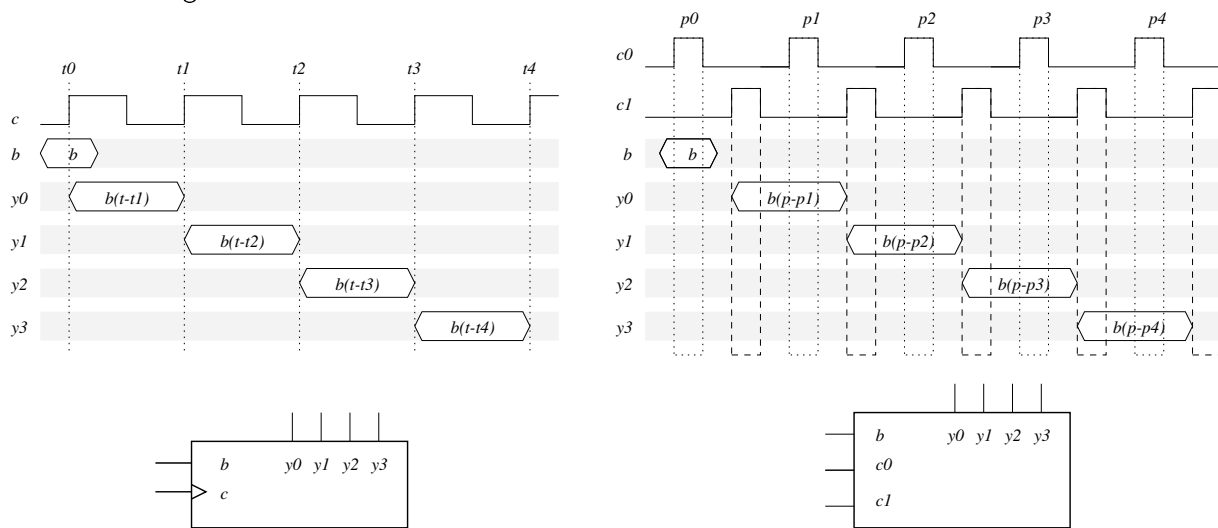
Nutzen Sie auch hierbei, wenn möglich *dont' cares* (*) bei der Minimierung. Geben Sie die minimierten Terme an.

- c) Realisieren Sie ausgehend von Ihrer Zustandstabelle ein Schaltwerk mit D-Flipflops zur Ampelsteuerung. Um welchen Automatentyp handelt es sich?
- d) Was passiert, wenn der Automat, etwa auf Grund von elektromagnetischen Störungen, in einen nicht kodierten Zustand gerät? Wie könnte in einem solchen Fall eine Fehlerbehandlung aussehen?

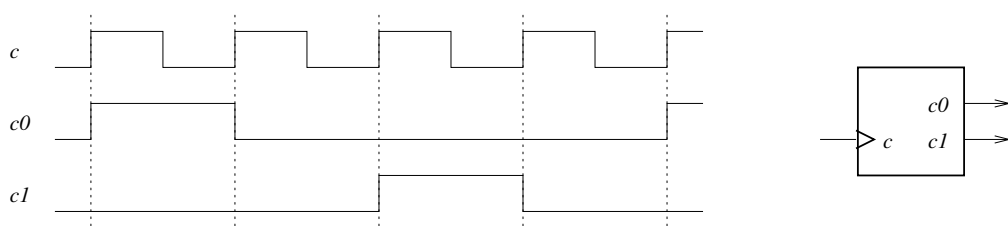
- e) In welchem Zustand befindet sich der Automat nach dem Einschalten? Wie kann der Automat in einen definierten Anfangszustand gebracht werden?

Aufgabe 6.2 Taktflanken- und taktzustandgesteuerte Schieberegister (4 Punkte)

Es wird grundsätzlich zwischen taktflankengesteuerten und taktzustandsgesteuerten Flipflops unterschieden. Der Unterschied liegt im zeitlichen Verhalten. Dieser Unterschied wird z.B. beim Aufbau von Schieberegistern besonders deutlich. Bei den Impulsdigrammen sind zwar für die Eingangssignale Setup- und Hold- Zeiten skizziert, jedoch nicht Verzögerungszeiten für die Ausgangssignale. Der linke Teil der folgenden Abbildung zeigt das Impulsdigramm eines Ein-Phasen-Takt Schieberegisters, der rechte Teil zeigt das Impulsdigramm eines Zwei-Phasen-Takt Schieberegisters:



- a) Realisieren Sie mit taktflankengesteuerten D-Flipflops (wie z.B. 7474) ein 4-Bit-Schieberegister, das mit einem Ein-Phasen-Takt angesteuert wird.
- b) Warum kann man mit taktzustandsgesteuerten D-Flipflops (wie z.B. 7475) kein Ein-Phasen-Takt Schieberegister wie in Aufgabe a) realisieren?
- c) Realisieren Sie mit taktzustandsgesteuerten D-Flipflops ein 4-Bit-Schieberegister, dass mit einem nicht überlappenden Zwei-Phasen-Takt angesteuert wird.
- d) Realisieren Sie ein Moore-Schaltwerk mit taktflankengesteuerten D-Flipflops, welches einen nicht überlappenden Zwei-Phasen-Takt aus einem einfachen Taktsignal generiert. Verwenden Sie dabei eine einschrittige Zustandskodierung.



- e) Welche besondere Eigenschaft ergibt sich aus der einschrittigen Zustandskodierung in Aufgabe d) bezüglich der Ausgangssignale?

Aufgabe 6.3 Automat (4 Punkte)

Es ist ein getaktetes Moore-Schaltwerk zu entwerfen, welches in Abhängigkeit der (einzigen) Eingangsvariablen x folgende zyklische Ausgabe der drei Ausgangsvariablen (y_2, y_1, y_0) liefert:

$$\begin{aligned} x = 0 & : (000) \Rightarrow (100) \Rightarrow (011) \Rightarrow (000) \Rightarrow \dots \\ x = 1 & : (000) \Rightarrow (110) \Rightarrow (111) \Rightarrow (100) \Rightarrow (000) \Rightarrow \dots \end{aligned}$$

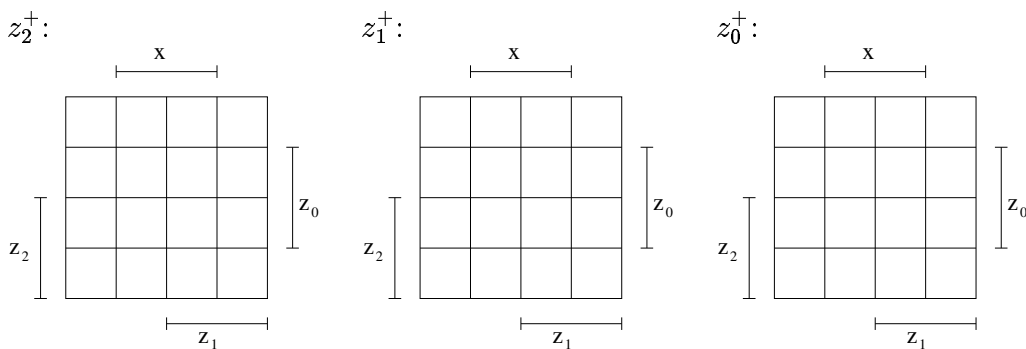
Der Übergang von einem Zyklus in den anderen beim Wechsel von x soll spätestens beim Erreichen der Ausgabe (000) erfolgen. Da fünf verschiedene Ausgabevektoren vorkommen, sind wenigstens drei Zustandsvariablen

$$(z_2, z_1, z_0)$$

erforderlich. Andererseits sind die Zyklen maximal vier Takte lang. Deshalb sind drei Zustandsvariablen auch ausreichend. Überdies kommt dieselbe Ausgabe in jedem Zyklus maximal einmal vor. In solch einem Fall ist es meist günstig, als Zustandskodierung direkt die Ausgabe zu verwenden, weil dann das Lambda-Schaltnetz fortfällt. Setzen Sie also

$$(z_2, z_1, z_0) = (y_2, y_1, y_0).$$

- a) Stellen Sie ein Zustandsdiagramm auf und ermitteln Sie die zugehörige Übergangstabelle.
- b) Führen Sie eine Minimierung des Delta-Schaltnetzes mit den KV-Diagrammen unter Ausnutzung nicht spezifizierter Stellen durch.



- c) Durch die Realisierung werden nicht spezifizierte Übergänge festgelegt. Zeichnen Sie das realisierte Zustandsdiagramm und prüfen Sie (a) ob alle Übergänge mit der Spezifikation im Einklang sind und (b) ob das Schaltwerk nach dem Einschalten aus jedem beliebigen Zustand heraus in den gewünschten Zyklus mündet. Dokumentieren Sie evtl. gefundene Designfehler.
- d) Beseitigen Sie in 6 c) gefundene Designfehler.