

Codes

- **Abbildung eines Zeichensatzes auf einen anderen Zeichensatz**
 - ASCII-Code stellt Ziffern und Buchstaben dar
 - „Dualcode“ stellt Zahlen dar
 - Maschinencode stellt Befehle dar

Hexadezimale Darstellung 41

Code	Bedeutung
ASCII	„A“
Zahl	65 (dezimal)
Maschinencode (8081)	AJMP <adresse>

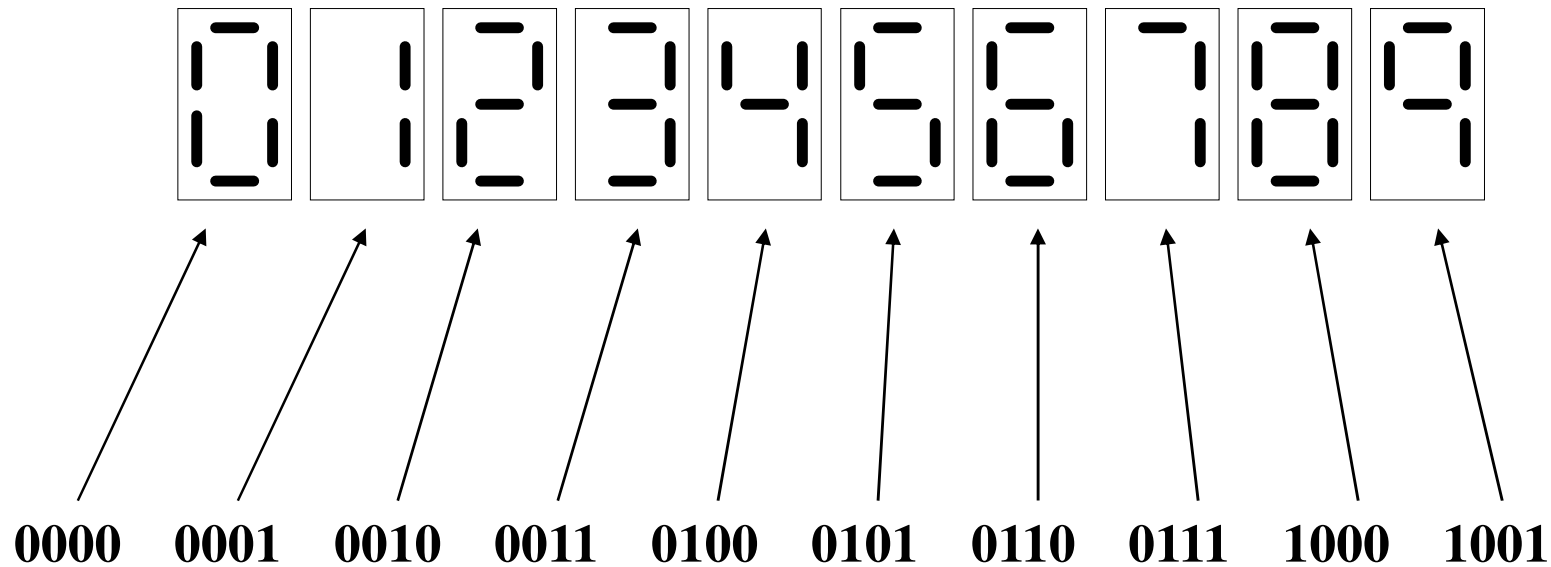
Codes, Wahrheitstabellen, Funktionen

Beispiel: Siebensegment-Dekoder

- Betrachtung als Code: Abbildung von vierstelligen Dualzahlen auf siebenstellige Siebensegmentanzeigemuster
- Betrachtung als Wahrheitstabelle: Auflistung aller zehn Wertemuster und Angabe der resultierenden Wertemuster für die zugehörigen Segmente der Anzeige
- Betrachtung als Funktion: Aufstellen von sieben Einzelfunktionen (je eine für ein Segment der Anzeige) in Abhängigkeit der vier gleichen Eingangsvariablen (der vierstelligen Dualzahlen)

Siebensegmentdekoder (Code)

Zeichenvorrat des Codes „Siebensegment“



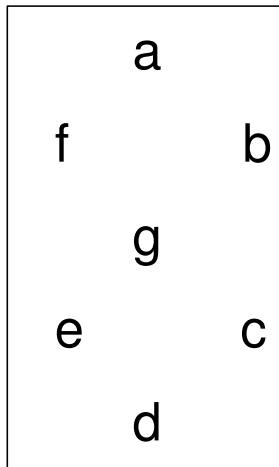
Zeichenvorrat des Codes „Dualzahl“

Siebensegmentdekodeer (Tabelle)

Eingänge

$$Z = \sum_{i=0}^3 x_i \times 2^i$$

Ausgänge



Eingänge				Ausgänge						
x ₃	x ₂	x ₁	x ₀	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1		1	1				
0	0	1	0	1	1		1	1		1
0	0	1	1	1	1	1	1			1
0	1	0	0		1	1			1	1
0	1	0	1	1		1	1		1	1
0	1	1	0	1		1	1	1	1	1
0	1	1	1	1	1	1				
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1			1	1

Siebensegmentdecoder (Funktionen)

Ableitung der Einzelfunktionen aus der Tabelle

$$a = \dots$$

$$b = \dots$$

$$c = \dots$$

$$d = \dots$$

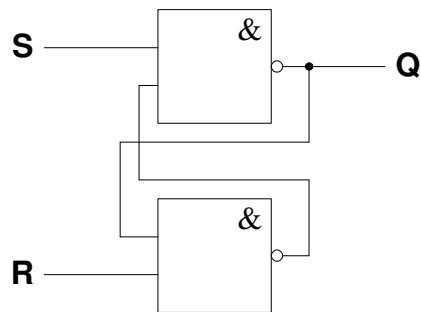
$$e = \bar{x}_3 \bar{x}_2 \bar{x}_1 \bar{x}_0 + \bar{x}_3 \bar{x}_2 x_1 \bar{x}_0 + \bar{x}_3 x_2 x_1 \bar{x}_0 + x_3 \bar{x}_2 \bar{x}_1 \bar{x}_0$$

$$f = \dots$$

$$g = \dots$$

Bistabile Kippstufen (FlipFlops)

- **Bisher: Schaltnetze (kombinatorische Schaltungen)**
 - Eingangsbelegung legt nach Einschwingen Ausgangsbelegung eindeutig fest
 - Schaltung hat kein Gedächtnis
- **Gesucht: Schaltung mit Gedächtnis**



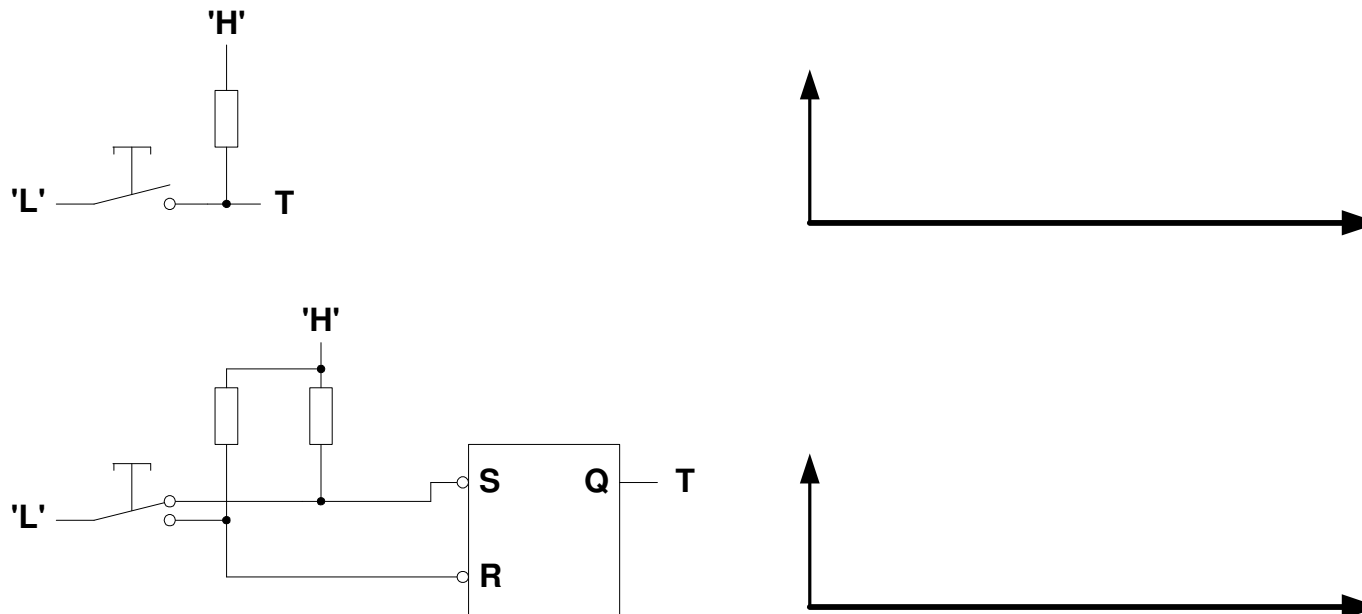
S	R	Q_n	Q_{n+1}
0	-	-	1
1	0	-	0
1	1	0	0
1	1	1	1

Bistabile Kippstufen

- **Eine bistabile Kippstufe (Flipflop) hat zwei stabile Zustände**
 - neuer Zustand abhängig von Eingangssignalen **und** dem derzeitigen Zustand
 - zustandsgesteuerte Flipflops ändern ihren Zustand aufgrund eines Signalpegels (statisch)
 - RS-Flipflop
 - Latch
 - flankengesteuerte Flipflops ändern ihren Zustand aufgrund eines Pegelwechsels (dynamisch)
 - D-Flipflop

Signalentprellung mit RS-FF

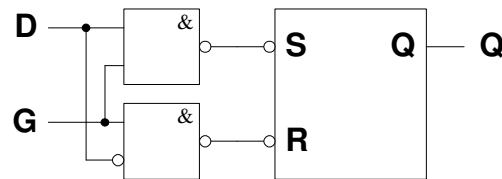
- **Beliebte Anwendung eines RS-Flipflops ist die Entprellung eines mechanischen Tasters**



Latch

- **Ein Latch („Riegel“)**

- schaltet während eines Taktsignalpegels den Eingang durch (das Latch ist transparent)
- hält während des anderen Taktsignalpegels den Ausgang auf dem zuletzt angenommenen Pegel (das Latch ist gesperrt)



G	D	Q_{n+1}
0	-	Q_n
1	0	0
1	1	1