

Übung 4 Schaltnetze

4.0 Theorie

- Warum werden bei Schaltnetzen gerne nur NAND bzw. NOR Gatter verwendet?
- Wozu vereinfacht man logische Funktionen?
- Vereinfachung einer Schaltfunktion mit Hilfe eines KV-Diagrammes
 - Wodurch zeichnen sich zwei benachbarte Felder im KV-Diagramm aus?
 - Was sind *Primiplikanten*?
 - Was sind *wesentliche Primimplikanten*?

4.1 Realisieren Sie folgendes Schaltnetz als Gatterschaltung

- unter ausschließlicher Verwendung von NAND-Gattern,
- unter ausschließlicher Verwendung von NOR-Gattern:

a	b	s	c
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

4.2 a) Erstellen Sie DKNF und KKNF der folgenden Schaltfunktion:

a	b	c	d	f
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1

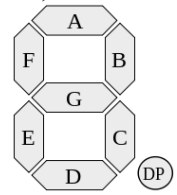
b) Erstellen Sie DKNF und KKNF der folgenden Schaltfunktion:

a	b	c	d	f
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

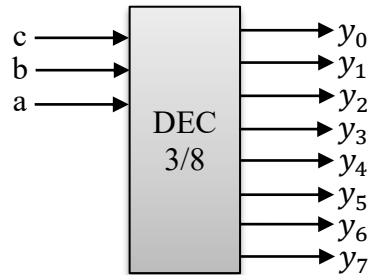
- Vereinfachen Sie die Schaltfunktion aus 4.2a mit Hilfe eines KV-Diagrammes.
- Vereinfachen Sie die Schaltfunktion aus 4.2b mit Hilfe eines KV-Diagrammes

4.4 Entwickeln Sie eine Logikschaltung mit drei Eingängen und einem Ausgang. Falls die Anzahl der Eingänge, an denen '1' anliegt, ungerade ist, soll am Ausgang '1' ausgegeben werden, ansonsten '0'. Wie heißt die logische Funktion, die sich derart verhält? Wie sieht diese Funktion im KV-Diagramm aus?

- 4.5 a) Entwickeln Sie eine Ansteuerungslogik für **den mittleren Querbalken** (Segment G) einer 7-Segment-Anzeige. Die gesamte Anzeige soll binär codierte Zahlen von 0-9 darstellen.
 b) Minimieren Sie die Ansteuerungslogik mittels KV-Diagramm und realisieren Sie die Schaltung mit NAND Gattern (Hinweis: Fassen Sie die „1“ und „don't-care“ Werte zusammen).



- 4.6 Erstellen Sie die Wertetabelle eines Schaltnetzes (Decoder 3/8), das binäre Zahlen an den Eingängen a, b, c auf Ausgänge y_0, y_1, \dots, y_7 abbildet, so dass jeweils nur der Ausgang y_i den Wert '1' ausgibt, falls an abc der Wert i (binär codiert) anliegt.



- 4.7 Realisieren Sie mit Hilfe des Decoders aus 4.6 folgendes Schaltnetz:

a	b	c	x	y	z
0	0	0	1	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	1
1	0	0	0	0	0
1	0	1	1	0	1
1	1	0	1	1	0
1	1	1	0	0	0

- 4.8 Realisieren Sie mit Hilfe des Decoders aus 4.6 einen Decoder 4/16. Sie dürfen dazu zusätzlich annehmen, dass der Decoder 3/8 über einen zusätzlichen Eingang „Enable“ verfügt; Liegt an diesem Eingang '0' an, liefert der Decoder an jedem Ausgang '0'.

