Prüfung

Prof. Dr.-Ing. J. Becker

Digitaltechnik

--

Institut für Technik der Informationsverarbeitung, KIT

Klausur

Do., 30.08.2012

Lösungsblätter

Hinweise zur Klausur

Hilfsmittel

Als Hilfsmittel zur Prüfung sind vier Seiten vorgegebene und **ein DIN A4 Blatt** selbst geschriebene Formelsammlung zugelassen. Nicht erlaubt hingegen sind die Verwendung eines Taschenrechners, zusätzliche Unterlagen und jegliche Kommunikation mit anderen Personen.

Prüfungsdauer

Die Prüfungsdauer beträgt für die Klausur 120 Minuten.

Prüfungsunterlagen

Die Prüfungsunterlagen bestehen aus insgesamt 27 Seiten Aufgabenblättern (einschließlich diesem Titelblatt und zwei zusätzlichen Lösungsblättern). Weiterhin sind 4 zusätzliche Seiten Formelsammlung enthalten.

Bitte überprüfen Sie vor der Bearbeitung der Aufgaben auf jeder Seite oben Ihren vorgedruckten Namen und ihre Matrikelnummer!

Bei Bedarf erhalten Sie zusätzliche Lösungsblätter. Auf jedes zusätzliche Lösungsblatt ist neben dem Namen auch die Aufgabennummer mit einzutragen. Vermeiden Sie das Beschreiben der Rückseiten.

Am Ende der Prüfung sind die 27 Seiten Aufgaben- und Lösungsblätter und alle verwendeten zusätzlichen Lösungsblätter abzugeben.

Verwenden Sie zum Bearbeiten der Aufgaben lediglich dokumentenechte Schreibgeräte – keinen Bleistift sowie Rotstifte!

Aufgabe 1 Fehlererkennung- und Korrektur

Aufgabe 1.1 Hamming Codes

Bei einer seriellen Datenübertragung wird zur Sicherung ein Hamming-Code verwendet. Dieser fasst vier Datenbits zusammen mit drei Prüfbits zu einem Codewort zusammen.

Die Generierung der Prüfstellen y0, y1 und y2 erfolgt anhand folgender boolscher Verknüpfungen:

$$\boldsymbol{y0 = x0 \oplus x2 \oplus x3}$$

$$\boldsymbol{y1 = x0 \oplus x1 \oplus x2}$$

$$\boldsymbol{y2 = x0 \oplus x1 \oplus x3}$$

Hinweis: $\oplus$ ist eine XOR oder Antivalenz-Verknüpfung.

A) Konstruieren Sie den 1-F-korrigierbaren Hammingcode, indem Sie die untenstehende Tabelle 1-1 ergänzen. Bestimmen Sie die korrekte Zuordnung von Prüf- und Datenbits entsprechend der in Tabelle 1-1 gegebenen dualen Kennzahlen. Tragen Sie ein, welche Datenbits durch welche Prüfbits geschützt werden.

lfd. Nr. **duale Kennzahl**	1 **001**	2 **010**	3 **011**	4 **100**	5 **101**	6 **110**	7 **111**
1. Prüfstelle	y0		x2		x3		x0
2. Prüfstelle		y1	x2			x1	x0
3. Prüfstelle				y2	x3	x1	x0

Tabelle 1-1: 1-F-korrigirbarer Hammingcode

Es wurde folgende Bitfolge empfangen:

0111001 0001000 0001011 0010110

Die Anordnung der Daten- und Prüfbits innerhalb eines Codeworts ist in Abbildung 1-1 dargestellt.

X3	X2	X1	X0	Y2	Y1	Y0

Abbildung 1-1: Hamming-Codewort

B) Rekonstruieren Sie die gesendeten Datenwörter. Beachten Sie dabei, dass eine fehlerhafte Übertragung nicht ausgeschlossen ist. Gehen Sie davon aus, dass gerade Parität verwendet wird und pro Codewort maximal 1 Bit fehlerhaft ist. Markieren Sie ggf. die korrigierten Bits und geben Sie sowohl die korrigierte Bitfolge, als auch die resultierenden (Nutz-)Datenwörter an.

*Korrigierte Bitfolge: 0**0**11001 000**0**000 0001**1**11 0010110*

Datenwörter: 0011 0000 0001 0010

C) Berechnen Sie die Wortfehlerrate $\left(\frac{\text{Fehlerhaft Empfangenes CW}}{\text{Empfangene CW}}\right)$ der Übertragung aus B). Geben Sie außerdem den Overhead $\left(\frac{\text{Übertragene Daten} - \text{Nutzdaten}}{\text{Übertragene Daten}}\right)$ der Datenübertragung an.

Wortfehlerrate = 3 / 4

Overhead Hamming-Code = k / (m+k) = 3 / 7

D) Berechnen Sie nun für einen Vergleich mit der Hamming-Codierung den Overhead für eine Datenübertragung bei der maximal ein Fehler erkannt werden kann. Ein Datenwort soll dabei wieder vier Bit lang sein.

Paritätsbit nötig für 1-F Erkennbar: Nutzdaten: 4 Bit,

Übertragene Daten: 5 Bit

Overhead 1-F Erkennbar = 1 / 5

Aufgabe 1.2 Blocksicherung

Die folgende ASCII-Zeichenkette soll über einen störanfälligen Kanal übertragen werden:

„DT_2012“

Abbildung 1-2: ASCII-Zeichenkette

Dabei soll eine Blocksicherung mit <u>ungerader Parität</u> vorgesehen werden.

A) Tragen Sie die entsprechenden ASCII Codes in die folgende Tabelle 1-2 ein und ergänzen Sie die Paritätsbits und das Prüfwort.
Hinweis: nehmen Sie die ASCII-Tabelle aus der Formelsammlung zur Hilfe.

Zeichen	MSB			**Codeworte**			LSB	**Parität**
D	1	0	0	0	1	0	0	1
T	1	0	1	0	1	0	0	0
_	1	0	1	1	1	1	1	1
2	0	1	1	0	0	1	0	0
0	0	1	1	0	0	0	0	1
1	0	1	1	0	0	0	1	0
2	0	1	1	0	0	1	0	0
Prüfwort	0	1	1	0	0	0	1	0

Tabelle 1-2: Codewort-Tabelle

B) Sie wollen einen 40-Bit langen Bitstrom über eine störanfällige Übertragungsstrecke mit einer Übertragungsrate von 1000 Bit/s senden. Es sind Störungen mit einer Dauer von max. 6ms zu erwarten. Der minimale Abstand zwischen zwei Störungen beträgt 100ms. Ein Bit gilt als fehlerhaft, wenn es zu mehr als 50% der Übertragungsdauer gestört ist.

Eine Blocksicherung mit Scrambling soll zur Übertragung verwendet werden. Der Bitstrom soll dabei nach Möglichkeit in einem Block übertragen werden. Wie wählen Sie die Anzahl von Zeilen und Spalten, sodass Bündelfehler maximaler Länge gerade noch erkannt werden? Geben Sie Werte an.

Maximale Bitlänge von Burstfehler = 6ms / (1bit/ms) = 6 bit

Anzahl an Zeilen/Spalten für Nutzdaten = 6 – 1 = 5

Anzahl an Spalten/Zeilen für Nutzdaten = 40/5 = 8

Spalten/Zeilen gesamt = 8 +1 = 9

Zeilen/Spalten gesamt = 5 + 1 = 6

Aufgabe 2 Mengen und Relationen

Aufgabe 2.1 Allgemein: Mengen, Graphen, Relationen

A) Geben Sie für die nachstehenden Aussagen an, ob sie wahr oder falsch sind. Fehlerhafte Antworten resultieren in Punktabzug. Die Aufgabe wird mit mindestens 0 Punkten bewertet.

Aussage	Wahr	Falsch
Eine Menge kann mehrere gleiche Elemente besitzen.	☐	☒
Die Elemente einer Menge sind sortiert.	☐	☒
Eine Menge kann unendliche viele Elemente besitzen.	☒	☐
Eine Relation kann zur Beschreibung einer Menge verwendet werden.	☒	☐
Jeder ungerichtete, zyklenfreie, zusammenhängende Graph ist ein Baum.	☒	☐
Für jede Menge A gilt $\lvert \mathrm{P}(A) \rvert \geq 1$	☒	☐
„>“ ist eine Äquivalenzrelation	☐	☒
Eine Ordnungsrelation ist transitiv, symmetrisch und reflexiv	☐	☒
Ein Graph und dessen dualer Graph haben immer die gleiche Anzahl an Knoten.	☐	☒
Es gibt einen Baum mit mindestens zwei Knoten, dessen Graph entartet ist.	☐	☒
Es gibt einen Graphen, dessen dualer Graph isomorph zum Graph ist.	☒	☐
Jeder zusammenhängende zyklische Graph kann durch weglassen von Kanten in einen Baum transformiert werden.	☒	☐

Tabelle 2-1: Allgemein: Mengen, Graphen, Relationen

Aufgabe 2.2 Eigenschaften von Relationen

Sie M eine Menge und $\alpha \subseteq M \times M$ eine Relation auf M. Geben Sie für jede der drei folgenden Relationen an, welche Eigenschaften zutreffen. Fehlerhafte Antworten resultieren in Punktabzug. Jede Teilaufgabe wird mit mindestens 0 Punkten bewertet.

A) Äquivalenzrelation.

Aussage	Wahr	Falsch
$\forall X \in M : X\alpha X$ (Reflexiv)	☒	☐
$\forall X, Y \in M : X\alpha Y \Rightarrow Y\alpha X$ (Symmetrisch)	☒	☐
$\forall X, Y \in M : X\alpha Y \wedge Y\alpha X \Rightarrow X = Y$ (Anti-Symmetrisch)	☐	☒
$\forall X, Y, Z \in M : X\alpha Y \wedge Y\alpha Z \Rightarrow X\alpha Z$ (Transitiv)	☒	☐

B) Verträglichkeitsrelation.

Aussage	Wahr	Falsch
$\forall X \in M : X\alpha X$ (Reflexiv)	☒	☐
$\forall X, Y \in M : X\alpha Y \Rightarrow Y\alpha X$ (Symmetrisch)	☒	☐
$\forall X, Y \in M : X\alpha Y \wedge Y\alpha X \Rightarrow X = Y$ (Anti-Symmetrisch)	☐	☒
$\forall X, Y, Z \in M : X\alpha Y \wedge Y\alpha Z \Rightarrow X\alpha Z$ (Transitiv)	☐	☒

C) Ordnungsrelation.

Aussage	Wahr	Falsch
$\forall X \in M : X\alpha X$ (Reflexiv)	☒	☐
$\forall X, Y \in M : X\alpha Y \Rightarrow Y\alpha X$ (Symmetrisch)	☐	☒
$\forall X, Y \in M : X\alpha Y \wedge Y\alpha X \Rightarrow X = Y$ (Anti-Symmetrisch)	☒	☐
$\forall X, Y, Z \in M : X\alpha Y \wedge Y\alpha Z \Rightarrow X\alpha Z$ (Transitiv)	☒	☐

Aufgabe 3 Boolesche Algebra

Aufgabe 3.1 Boolesche Algebra

Gegeben sei das folgende Schaltnetz:

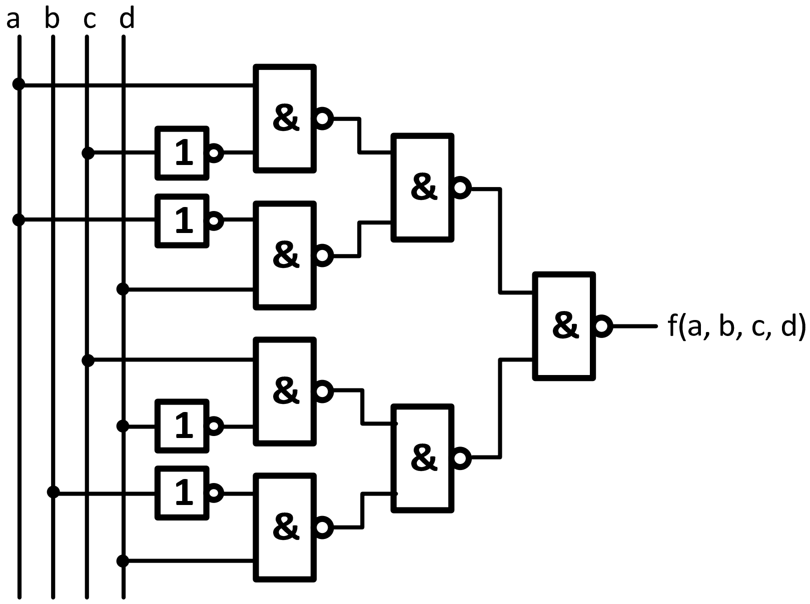


Abbildung 3-1: Schaltnetz

A) Bestimmen Sie die Boolesche Funktion $f(a,b,c,d)$ aus Abbildung 3-1 und stellen Sie sie in Disjunktiver Minimalform (DMF) dar.

$$f(a,b,c,d) = \overline{\left(\overline{\left(\overline{(a\wedge\bar{c})}\wedge\overline{(\bar{a}\wedge d)}\right)}\right)\wedge\left(\overline{\left(\overline{(c\wedge\bar{d})}\wedge\overline{(\bar{b}\wedge d)}\right)}\right)}$$

$$= \overline{\left(\overline{\left(\overline{(a\wedge\bar{c})}\wedge\overline{(\bar{a}\wedge d)}\right)}\right)}\vee\overline{\left(\overline{\left(\overline{(a\wedge\bar{c})}\wedge\overline{(\bar{a}\wedge d)}\right)}\right)}$$

$$= \left(\overline{(a\bar{c})}\wedge\overline{(\bar{a}d)}\right)\vee\left(\overline{(c\bar{d})}\wedge\overline{(\bar{b}d)}\right) = (\bar{a}\vee c)(a\vee\bar{d})\vee(\bar{c}\vee d)(b\vee\bar{d})$$

$$= \bar{a}\,\bar{d}\vee a\,c\vee c\,\bar{d}\vee b\,\bar{c}\vee\bar{c}\,\bar{d}\vee b\,d$$

$$= \bar{d}\vee ac\vee\bar{c}b\vee bd$$

$$= \bar{d}\vee ac\vee\bar{c}b\vee b = \bar{d}\vee ac\vee b$$

B) Wie viele Gatter konnten Sie durch die Umformung aus Teilaufgabe A) einsparen, wenn Sie davon ausgehen, dass für die Realisierung UND, ODER und NICHT-Gatter mit maximal 2 Eingängen zur Verfügung stehen?

Es werden 2 ODER + 1 UND + 1 NICHT, also 4 Gatter benötigt.

⇨ *Es können 11 – 4 = 7 Gatter eingespart werden.*

Aufgabe 3.2 Entwicklungssatz

Geben sei nun folgende boolesche Funktion:

$$\boldsymbol{y(d,c,b,a) = (\overline{\bar{c} \vee \bar{d}}) \vee (\overline{\bar{c} \vee d \vee a}) \vee \bar{c}d(a \equiv b)}$$

A) Entwickeln Sie den Ausdruck y mit Hilfe des Booleschen Entwicklungssatzes in der Reihenfolge d, c, b, a. Geben Sie alle Zwischenergebnisse an.

Hinweis: Bringen Sie den Funktionsausdruck zuerst in eine geeignete Form.

$$y(d,c,b,a) = (\overline{\bar{c} \vee \bar{d}}) \vee (\overline{\bar{c} \vee d \vee a}) \vee \bar{c}d(a \equiv b)$$

De Morgan: $y(d,c,b,a) = \mathrm{cd} \vee (\mathrm{c\bar{d}\bar{a}}) \vee \bar{c}d(a \equiv b)$

$$y(d,c,b,a) = \mathrm{cd} \vee (\mathrm{c\bar{d}\bar{a}}) \vee \bar{c}d(a\mathrm{b} \vee \mathrm{\bar{a}\bar{b}})$$

Lösungsweg A:

Entwicklung nach d:

$$y(0,c,b,a) = \mathrm{c\bar{a}}$$

$$y(1,c,b,a) = \mathrm{c} \vee \bar{c}\left(a\mathrm{b} \vee \mathrm{\bar{a}\bar{b}}\right)$$

Entwicklung nach c:

$y(0,0,b,a) = 0$ $\qquad$ $y(0,1,b,a) = \mathrm{\bar{a}}$

$y(1,0,b,a) = a\mathrm{b} \vee \mathrm{\bar{a}\bar{b}}$ $\qquad$ $y(1,1,b,a) = 1$

Entwicklung nach b:

$$y(0,1,-,a) = \bar{a}$$

$y(1,0,0,a) = \bar{a}$ $\qquad$ $y(1,0,1,a) = a$

Entwicklung nach a:

$y(0,1,-,0) = 1$ $\qquad$ $y(0,1,-,1) = 0$

$y(1,0,0,0) = 1$ $\qquad$ $y(1,0,0,1) = 0$

$y(1,0,1,0) = 0$ $\qquad$ $y(1,0,1,1) = 1$

Lösungsweg B:

Entwicklung nach d:

$$y(d,c,b,a) = \mathrm{d}[\mathrm{c} \vee \bar{c}\left(a\mathrm{b} \vee \mathrm{\bar{a}\bar{b}}\right)] \vee \mathrm{\bar{d}}[\mathrm{c\bar{a}}]$$

Entwicklung nach c:

$$y(d,c,b,a) = \mathrm{d}[\mathrm{c} \vee \bar{c}\left(a\mathrm{b} \vee \mathrm{\bar{a}\bar{b}}\right)] \vee \mathrm{\bar{d}}[\mathrm{c}(\mathrm{\bar{a}}) \vee \bar{c}(0)]$$

Entwicklung nach b:

$$y(d,c,b,a) = \mathrm{d}[\mathrm{c} \vee \bar{c}\left(b(a) \vee \mathrm{\bar{b}}(\mathrm{\bar{a}})\right)] \vee \mathrm{\bar{d}}[\mathrm{c}(\mathrm{\bar{a}}) \vee \bar{c}(0)]$$

Entwicklung nach a:

$$y(d,c,b,a) = \mathrm{d}\left[\mathrm{c} \vee \bar{c}\left(b\left(a(1)\right) \vee \mathrm{\bar{b}}(\mathrm{\bar{a}}(1))\right)\right] \vee \mathrm{\bar{d}}[\mathrm{c}(\mathrm{\bar{a}}(1)) \vee \bar{c}(0)]$$

Aufgabe 4 Zahlensysteme und Codierung

Aufgabe 4.1 Binary Coded Decimal

A) Addieren Sie die beiden im Dezimalsystem gegebenen Zahlen 4596_D und 8573_D im BCD System und im Dezimalsystem Geben Sie sowohl den Lösungsweg als auch alle notwendigen Korrekturschritte an.

	BCD	Dec
	0100 0101 1001 0110	4596
+	1000 0101 0111 0011	+8573
	1 11 111 11	
=	1100 1011 0000 1001	
Korr.	PT PT Ü	
	1100 1011 0000 1001	
+	0110 0110 0110	
	1 1 1 1 1	
= 1	0011 0001 0110 1001	13169

B) Warum ist die BCD Codierung im Vergleich zur Gleitkommadarstellung weniger effizient und wird daher nur noch selten benutzt?

Höherer Speicherverbrauch

C) Welchen wichtigen Vorteil bietet die BCD Codierung gegenüber der Gleitkommadarstellung?

Jeder Dezimalzahl kann exakt kodiert werden ODER Lesbarkeit

(Die Zahl 0,3 lässt sich in Gleitkommadarstellung nicht genau darstellen)

D) Wie viel Bit würden Sie einsparen, wenn Sie die Zahl „256“ Binär statt im BCD-Format kodieren?

Binär: 9 Bit, BCD: 12 Bit => Sie sparen 3 Bit

Aufgabe 4.2 Polyadische Zahlensysteme

Vervollständigen Sie die offenen Felder in Tabelle 4-1 indem Sie jeweils die entsprechenden Konvertierungen durchführen.

Dezimal	**Binär**	**Oktal**	**Hex**
1865_D	111 0100 1001B	3511_O	749_H
183_D	$1011\ 0111_B$	267_O	$B7_H$
258_D	$1\ 0000\ 0010_B$	402_O	102_H
55_D	$11\ 0111_B$	67_O	37_H

Tabelle 4-1: Konvertierung von Zahlensystemen.

Aufgabe 4.3 Rechenoperationen im Binärsystem

A) Durch welche Operation lassen sich im Binärsystem Multiplikationen und Divisionen mit Zweierpotenzen leicht realisieren? Geben Sie ein einfaches Zahlenbeispiel für eine entsprechende Division im Binärsystem.

Durch Bitshift-Operationen (Verschiebungen)

Bsp: 101 1011 0010 / 10 = 10 1101 1001

B) Führen Sie im Binärsystem die Subtraktion $140_D – 250_D$ durch. Verwenden Sie die Zweierkomplementdarstellung und geben Sie alle Zwischenschritte an.

$140_D = 1000\ 1100_B$

$250_D = 1111\ 1010_B$

Für eindeutige Darstellung müssen Zahlen auf 9bit erweitert werden

Zweierkomplementdarstellung von $1111\ 1010_B$: $0000\ 0101_B + 1_B = 1\ 0000\ 0110_B$

Addition des Zweierkomplements:

	$0\ 1000\ 1100_B$
+	$1\ 0000\ 0110_B$
=	$1\ 1001\ 0010_B$

Negatives Ergebnis => Zweierkomplement bilden um Betrag zu ermitteln:

$0\ 0110\ 1101_B + 1_B = 0\ 0110\ 1110_B \rightarrow -110_D$

Aufgabe 5 Minimierung

Aufgabe 5.1 Schaltnetzanalyse

Gegeben sei die folgende Schaltung:

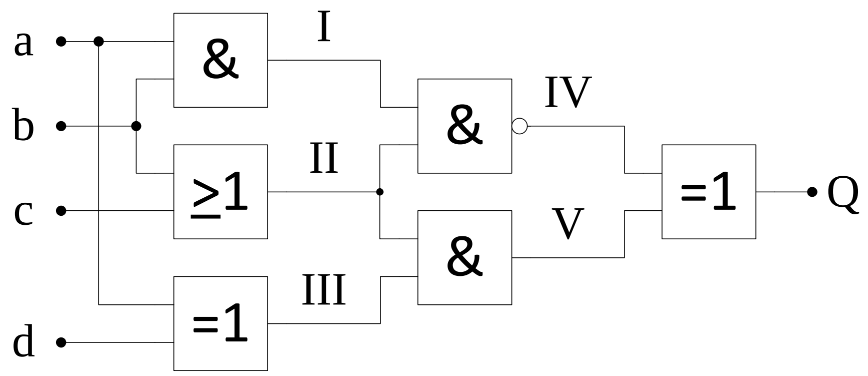


Abbildung 5-1: Schaltnetz der Funktion f(d,c,b,a) = Q

A) Vervollständigen Sie die Wahrheitstabelle der in Abbildung 5-1 angegebenen Schaltung. Tragen Sie dazu alle Zwischenergebnisse und den Ausgang Q in die Tabelle ein.

a	b	c	d	I	II	III	IV	V	Q
0	0	0	0	0	0	0	1	0	1
0	0	0	1	0	0	1	1	0	1
0	0	1	0	0	1	0	1	0	1
0	0	1	1	0	1	1	1	1	0
0	1	0	0	0	1	0	1	0	1
0	1	0	1	0	1	1	1	1	0
0	1	1	0	0	1	0	1	0	1
0	1	1	1	0	1	1	1	1	0
1	0	0	0	0	0	1	1	0	1
1	0	0	1	0	0	0	1	0	1
1	0	1	0	0	1	1	1	1	0
1	0	1	1	0	1	0	1	0	1
1	1	0	0	1	1	1	0	1	1
1	1	0	1	1	1	0	0	0	0
1	1	1	0	1	1	1	0	1	1
1	1	1	1	1	1	0	0	0	0

Tabelle 5-1: Wahrheitstabelle der Schaltfunktion f(d,c,b,a) = Q

B) Betrachten Sie nun die Konjunktive Normalform der Schaltfunktion G:

$$G: f(d,c,b,a) = (a \vee b \vee c \vee \bar{d}) \wedge (a \vee \bar{b} \vee c \vee d) \wedge (a \vee \bar{b} \vee c \vee \bar{d})$$
$$\wedge (a \vee \bar{b} \vee \bar{c} \vee d) \wedge (\bar{a} \vee b \vee c \vee \bar{d})$$

Übertragen Sie $f(d,c,b,a)$ in das unten stehende Symmetriediagramm und vervollständigen Sie alle Einträge. Beachten Sie hierbei die Reihenfolge der Variablen.

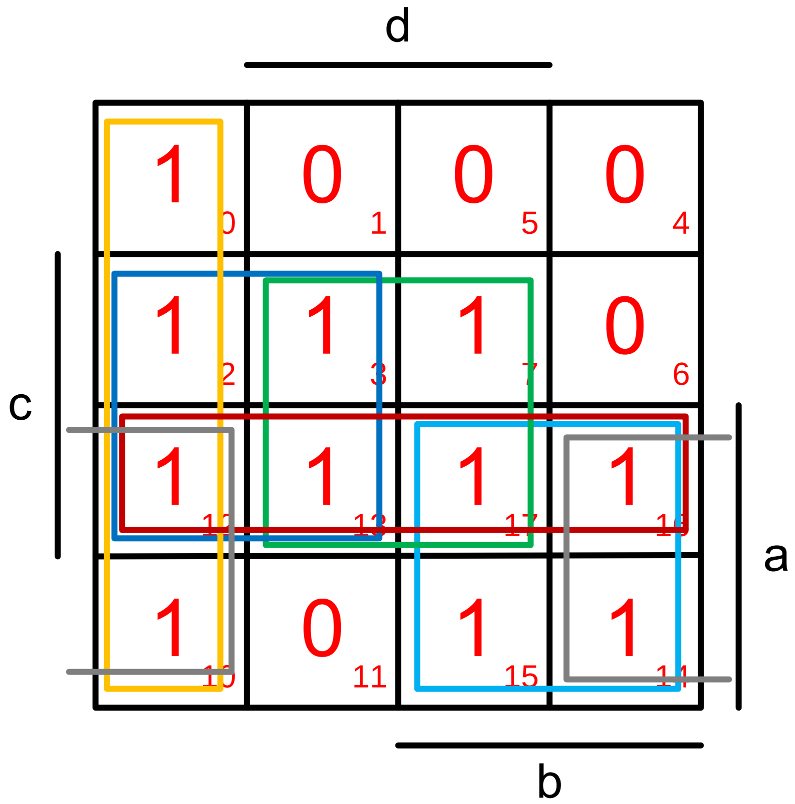


Abbildung 5-2: Symmetriediagramm

C) Geben sie **<u>alle</u>** möglichen Primterme für eine vollständige Einsstellenüberdeckung der Funktion G aus Abbildung 5-2 an. Geben Sie außerdem die resultierende Minimalüberdeckung an.

$\bar{b}\bar{d}, ab, ac, cd, c\bar{b}, a\bar{d}$

Minimalüberdeckung: $\bar{b}\bar{d}, ab, cd$

Aufgabe 5.2 Verfahren nach Petrick

In den folgenden Teilaufgaben sollen verschiedene Schritte des Petrick-Verfahrens durchgeführt werden.

A) Wenden Sie die Spaltendominanzregel auf Tabelle 5-2 an. Welche Spalte(n) können gestrichen werden? Streichen Sie die entsprechende(n) Spalte(n) und geben Sie die dominierte(n) und zugehörigen dominierende(n) Spalte(n), sowie die streichbaren Spalte(n) an.

pi/Ei	~~E_1~~	~~E_2~~	E_3	E_4	~~E_5~~	E_6	E_7	~~E_8~~
a	X					X		X
b		X					X	
c			X		X			
d		X		X			X	
e	X					X	X	X
f	X		X		X			
g		X		X				X
h					X			

Tabelle 5-2: Überdeckungstabelle 2

Dominierende Spalte(n):	E_1	E_2	E_5	E_8			
Dominierte Spalte(n):	E_6	E_4	E_3	E_6			
Streichbare Spalte(n):	E_1	E_2	E_5	E_8			

<u>Hinweis:</u> Die folgenden Teilaufgaben können und sollen vollkommen unabhängig von der vorherigen Aufgabe gelöst werden.

B) Wenden Sie nun die <u>Zeilendominanzregel</u> auf die bereits reduzierte Tabelle 5-3 an. Welche Zeile(n) können gestrichen werden? <u>Streichen Sie</u> die entsprechende(n) <u>Zeile(n) und die Spalte(n)</u> die durch dominierende Zeilen überdeckt werden. Geben Sie die dominierte(n) und zugehörigen dominierende(n) Zeile(n), sowie die streichbaren Zeile(n) in der Tabelle an. Beachten Sie dabei, dass minimale Kosten (gegeben in *gate equivalent* (*GE*)) entstehen sollen. Neu entstandene Kerne dürfen mitverwendet werden. Führen Sie also das Patrick-Verfahren zu Ende.

$p_i \backslash N_i$	E_1	E_2	E_4	E_7	E_{10}	E_{13}	E_{14}	Kosten
a				X				~~6 GE~~
c		X	X			X	X	~~4 GE~~
d			X			X		~~1 GE~~
g	X	X						~~5 GE~~
j	X						X	~~4 GE~~
n		X		X	X	X		~~3 GE~~
p	X						X	~~2 GE~~
r			X		X		X	~~8 GE~~

Tabelle 5-3: Reduzierte Überdeckungstabelle

Dominierende Zeile(n):	n	p/j	p	d/c	d		
<u>Dominierte</u> Zeile(n):	a	g	j	r	c		
Streichbare Zeile(n):	a	g	j	r	c		

C) Geben Sie die Präsenzvariablen (p_i) für die in Teilaufgabe B) gefundenen Überdeckung an.

Überdeckung:

p, n, d

Aufgabe 6 Optimale Codes

Aufgabe 6.1 Codierung

Der am ITIV eingesetzte Kaffeeautomat soll modernisiert und dabei so modifiziert werden, dass neben einer personenbezogenen Abrechnung des Kaffeeverbrauchs auch eine Statistik erzeugt und dem Benutzer angezeigt wird. Um eine Abschätzung der hierzu benötigten Hardware treffen zu können, wurde unter anderem eine empirische Ermittlung des Kaffeetassenaufkommens durchgeführt, welche in Tabelle 6-1 dargestellt ist.

Anhand des in Tabelle 6-1 gegebenen Kaffeetassenverbrauch wurde eine Codierung erstellt um den Speicher für die Aufzeichnung des Kaffeeverbrauchs möglichst effizient zu nutzen. Der zugehörige Codierbaum ist in Abbildung 6-1 dargestellt.

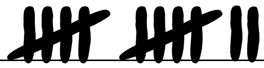
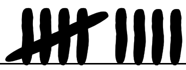
Mitarbeiter	Kaffeetassenverbrauch pro Woche	Ermittelte Codierung
A		0110
B		010
C		00
D		1011
E		110
F		01111
G		1110
H		1010
I		100
J		1111
K		01110

Tabelle 6-1: Kaffeetassenverbrauch

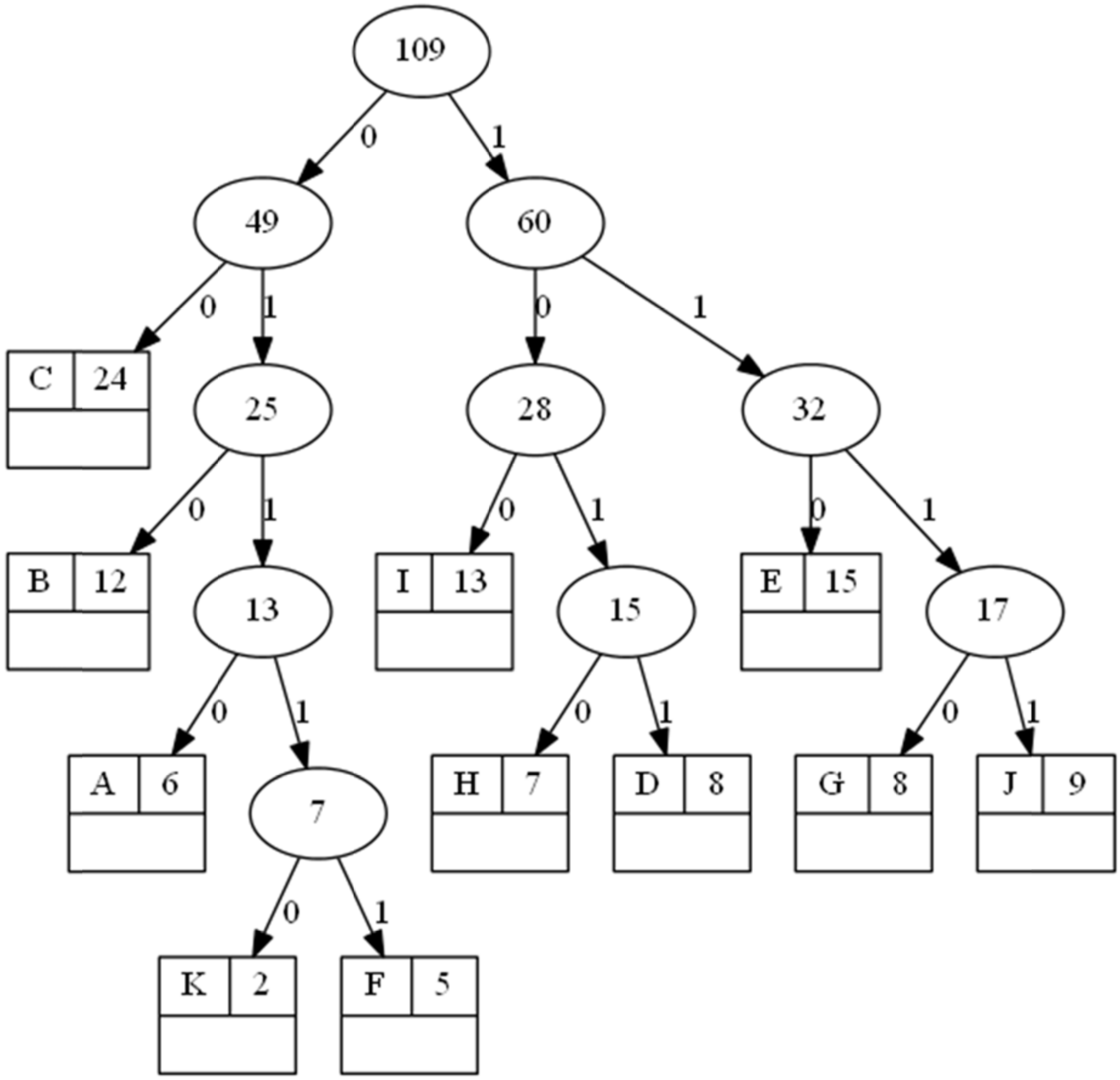


Abbildung 6-1: Codierbaum des Kaffeetassenverbrauch

A) In der Vorlesung haben Sie mehrere Codierungsverfahren kennengelernt. Welches Verfahren kam hier zur Anwendung? Begründen Sie Ihre Antwort.

Es handelt sich um eine Huffmann-Codierung.

Begründung: Es werden immer die beiden Knoten mit der geringsten Auftrittshäufigkeit zusammengefasst.Bei Shannon-Fano wäre eine eindeutige Sortierreihenfolge in den Blättern zu erkennen

B) Bestimmen Sie die Codierungen der einzelnen Mitarbeiter anhand des in Abbildung 6-1 erstellten Codierbaums und tragen Sie diese in Tabelle 6-1ein.

C) Geben Sie die Formel zur Berechnung der mittleren Codewortlänge für die Codierung an. Setzen Sie in die Formel die Werte zur Berechnung der Mittleren Codewortlänge für die in Abbildung 6-1 gegebene Codierung ein.

$$\bar{m} = \sum_{i=1}^{N} p(x_i) * m(x_i)$$

= (24*2 + 12*3 + 6*4 + 2*5 + 5*5 + 13*3 + 7*4 + 8*4 + 15*3 + 8*4 + 9*4) / (24+2+6+2+5+13+7+8+15+8+9)

= (6*4+12*3+24*2+8*4+15*3+5*5+8*4+7*4+13*3+9*4+2*5) / 109
= 355 / 109

D) Welche Länge hat die Historie der Kaffeetassenbezügen die im Mittel im Speicher abgelegt werden kann? Der Speicher hat eine Größe von 256 Byte. *(Die Angabe einer Formel ist ausreichend)*

Mittlere Anzahl an gespeicherten Kaffeetassenbezüge = (256 Byte * 8) / $\bar{m}$

E) Welche Eigenschaft der in Abbildung 6-1 gegebene Codierung erlaubt die eindeutige Decodierung eines Datenstroms, wie er im Aufgabenteil F) gegeben ist?

Die Präfixfreiheit

F) Nach einer kurzen Testphase enthält der Speicher der Kaffeemaschinensteuerung den folgenden Datenstrom:

0111001100111101111110110

Rekonstruieren Sie aus dem Datenstrom vollständig die im Speicher abgelegten Mitarbeiter. Das erste Bit entspricht dem Beginn eines Codeworts.

K, A, F, F, E, E

Aufgabe 6.2 Shannon-Fanø

Zur Optimierung einer Aufzugssteuerung soll eine Historie der Fahrziele der Aufzugskabine aufgezeichnet werden. Diese Historie soll über die Steuerelektronik in einem Speicher im Steuerungsrechner ablegen werden.

Um eine effiziente Codierung zu realisieren wurden die Fahrziele während eines Jahres gezählt und hieraus die in Tabelle 6-2 dargestellte Statistik erstellt.

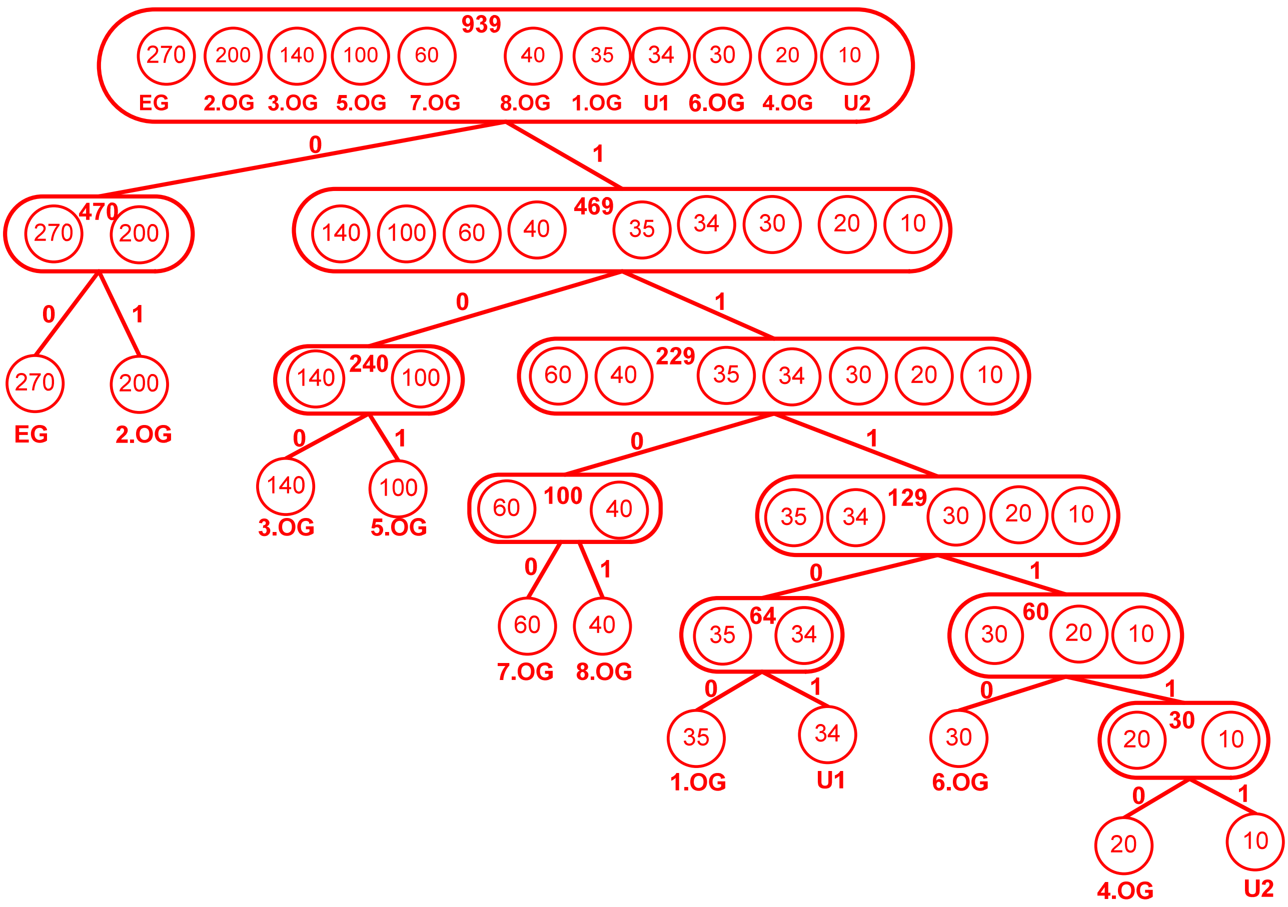
Etage	Anzahl der Fahrten zur Etage (x 1000)
8. OG	40
7. OG	60
6. OG	30
5. OG	100
4. OG	20
3. OG	140
2. OG	200
1. OG	35
EG	270
U 1	34
U 2	10

Tabelle 6-2: Fahrzielstatistik

A) Erstellen Sie einen Codierbaum nach dem Shannon-Fanø-Verfahren für die Codierung der Etagen anhand der in Tabelle 6-2 gegebenen Statistik.

Konventionen:

- *Sortieren Sie die Elemente entsprechend den Auftrittshäufigkeiten* ***aufsteigend von rechts nach links****. Falls unterschiedliche Knoten dieselbe Auftrittshäufigkeiten haben, sortieren Sie diese alphabetisch von links nach rechts.*
- *Teilen Sie eine Menge immer so auf, dass die Differenz zwischen den Summen der Auftrittshäufigen der Teilmengen minimiert wird.*
- *Weisen Sie den linken Ästen des entstehenden Baumes die „0“ zu, den rechten Ästen die „1“.*



Aufgabe 7 Automaten

Aufgabe 7.1 Entwurf von Automaten

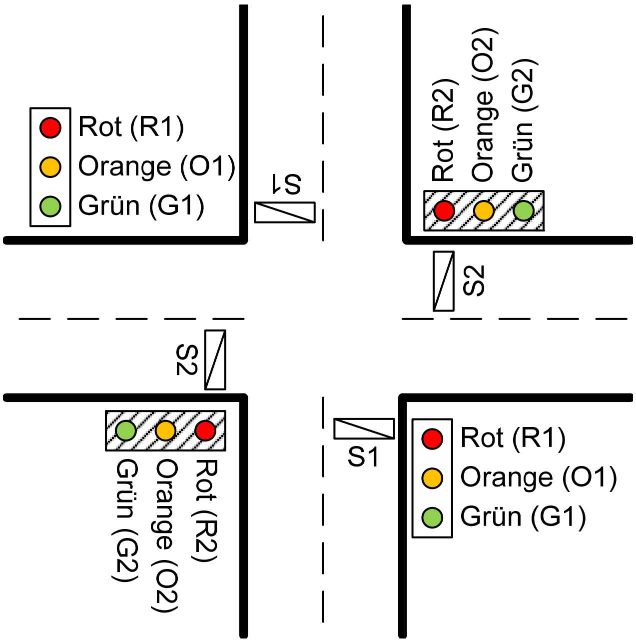


Abbildung 7-1: Ampelschaltung

Eine vielbefahrene Kreuzung soll eine Verkehrsampel erhalten (siehe Abbildung 7-1). Die Ansteuerung soll über einen Automaten erfolgen. Der Automat unterliegt folgenden Bedingungen:

- Beide Ampeln pro Fahrspur (FS_1, FS_2) werden identisch angesteuert.
- Signale: R_1, R_2 (rot), O_1, O_2 (orange), G_1, G_2 (grün) für Fahrspur 1 bzw. 2.
- Trägt ein Signal den Wert 1 so leuchtet die entsprechende Lampe der Ampel.
- Es gibt folgende Ampelphasen: „Rot“, „RotOrange“, „Orange“, „Grün“
- Die Dauer einer Ampelphase wird durch einen Timer bestimmt, Der Ablauf wird mit einem Timersignal T=1 angezeigt. Bei einem Phasenwechsel wird T auf T=0 zurückgesetzt.
- Eine Ampelsequenz ist ein Abfolge von Ampelphasen: „Rot“ (R_i=1), „Rot und Orange“ (R_i=1, O_i=1), „Grün“ (G_i=1), „Orange“ (O_i=1) „Rot“ (R_i=1).
- Eine Ampelsequenz startet nur wenn beide Fahrspuren „Rot“ zeigen (R_i=1, R_j=1).
- Während einer Ampelsequenz für FS_i zeigt die Ampel der zweiten Fahrspur FS_j immer Rot (R_j = 0).
- Es gibt zwei Sensoren S1 und S2 die anzeigen ob sich Fahrzeuge auf der Fahrspur befinden (S_i=1) oder nicht (S_i=0).
- Sobald sich ein Fahrzeug im Sensorbereich befindet soll es schnellstmöglich „Grün“ bekommen um die Weiterfahrt zu garantieren.
- FS_1 wird immer bevorzugt wenn beide Sensoren ein Fahrzeug detektieren
- Wenn alle Ampeln „Rot“ zeigen wird diese Phase erst verlassen sobald mindestens ein Fahrzeug im Sensorbereich ist, unabhängig vom Timersignal T.

A) Vervollständigen Sie die in Tabelle 7-1 dargestellte Liste der notwendigen Signalausgaben des Automaten. Im Zustand A_1 sind beide Ampeln rot.
Hinweis: nicht benötigte Zeilen oder Spalten können gestrichen werden

	R_1	O_1	G_1	R_2	O_2	G_2
A_1	1			1		
A_2	1	1		1		
A_3			1	1		
A_4		1		1		
A_5	1			1	1	
A_6	1					1
A_7	1				1	
A_8						
A_9						
A_{10}						
A_{11}						

Tabelle 7-1: Signalausgabe des Automaten

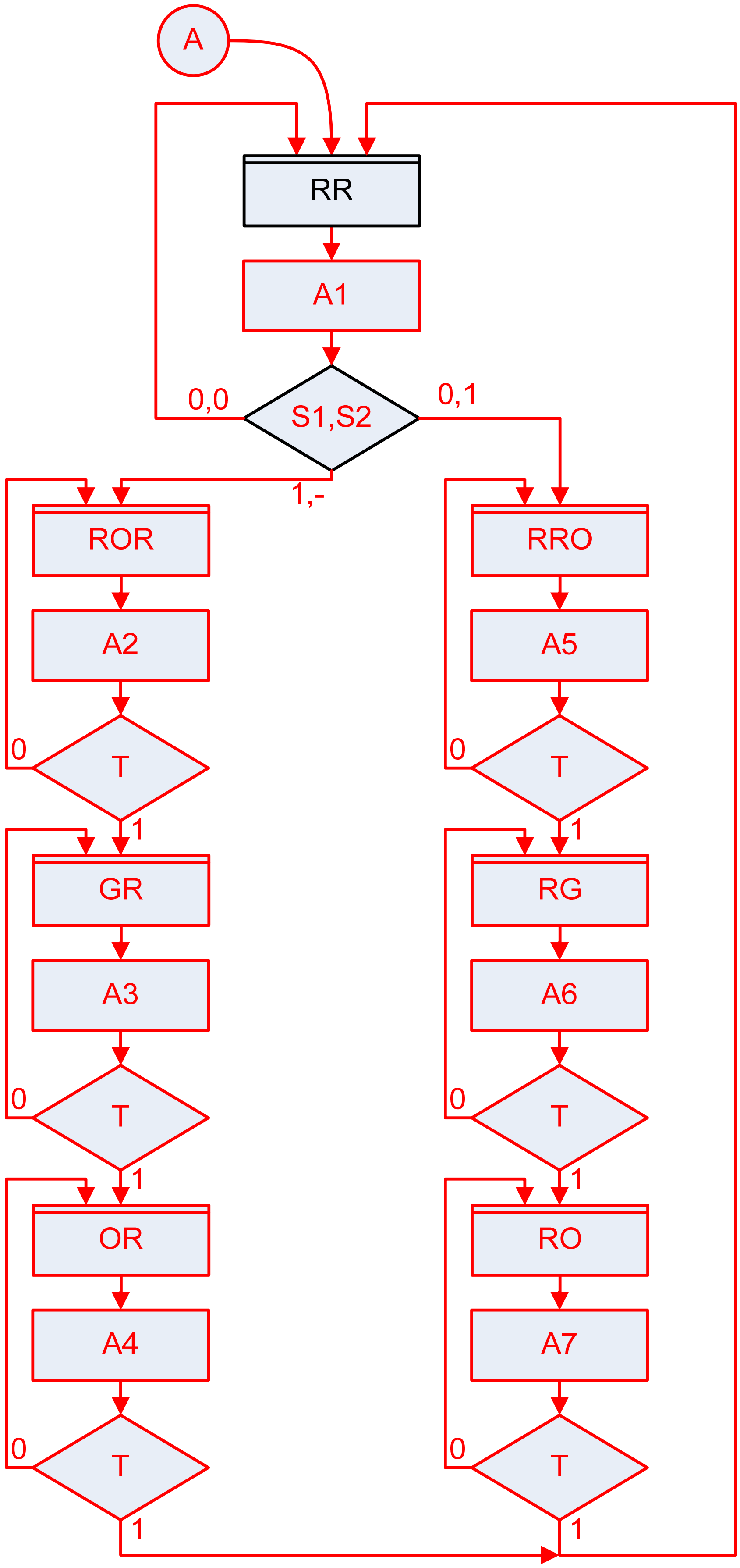
B) Geben Sie alle in der Vorlesung vorgestellten Automatentypen an.

Moore-, Mealy-, Medwedew-Automat

C) Welcher Automatentyp ermöglicht die schnellst mögliche Umschaltung der Lichtsignale als Reaktion auf die Sensoreingaben S_1 und S_2 sowie T? Begründen Sie Ihre Antwort.

Eine Realisierung als Mealy-Automat hat die schnellste Reaktionszeit, da sämtliche Eingaben sofort auf die Ausgabe durchschlagen, die im Vorliegenden Fall der Lampenansteuerung entspricht

D) Zeichnen Sie die Ampelschaltung als Moore-Automat mit einer minimalen Anzahl von Zuständen.



Aufgabe 7.2 Technische Realisierung von Automaten

Untenstehende Ablauftabelle definiert einen Automaten mit einem Eingang.

Q_3^n	Q_2^n	Q_1^n	in	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	y_3,y_2,y_1	T_3	T_2	D_1
0	0	0	0	0	0	1	000	0	0	1
0	0	1	0	0	1	0	001	0	1	0
0	1	0	0	0	1	1	010	0	0	1
0	1	1	0	1	0	1	011	1	1	1
1	0	1	0	1	1	1	101	0	1	1
1	1	1	0	0	0	0	111	1	1	0
0	0	0	1	1	1	1	000	1	1	1
0	0	1	1	0	0	0	001	0	0	0
0	1	0	1	0	0	1	010	0	1	1
0	1	1	1	0	1	0	011	0	0	0
1	0	1	1	0	1	1	101	1	1	1
1	1	1	1	1	0	1	111	0	1	1

Tabelle 7-2: Ablauftabelle eines Automaten

A) Welche Aufgabe hat „in“?

„in“ legt fest ob die Zahlen in aufsteigender Reihenfolge oder absteigender Reihenfolge ausgegeben werden

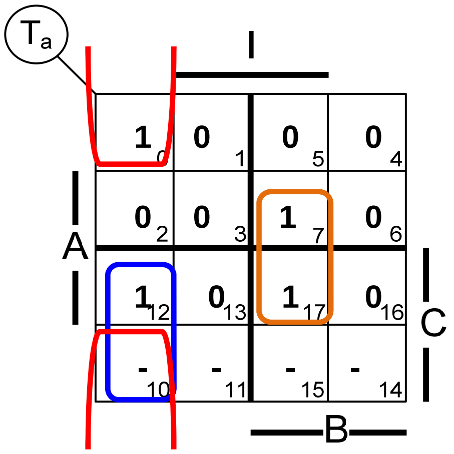
B) Welcher Automatentyp liegt vor? Begründen Sie Ihre Antwort.

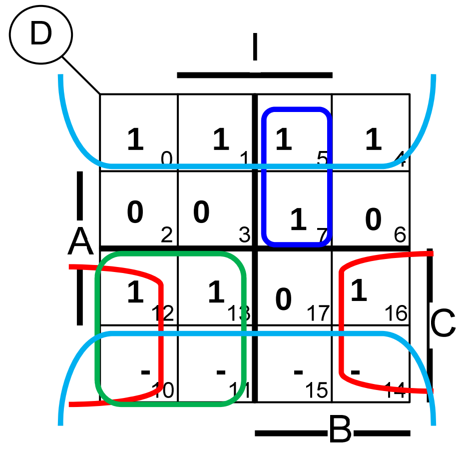
Medwedew-Automat, das Ausgabe identisch zur Zustandscodierung ist

C) Vervollständigen Sie in Tabelle 7-2 die Ansteuerfunktionen zur Realisierung des Automaten mittels zweier T-FlipFlops und eines D-FlipFlops.

Aufgabe 7.3 Ansteuerfunktionen

A) Gegeben sind nun die beiden unten dargestellten Symmetriediagramme zweier FlipFlop Eingänge **T_a und D**. Geben Sie die Ansteuerfunktionen von **T und D** in disjunktiver Minimalform an. Verwenden Sie „don't-cares" wenn immer möglich.





$$T_a = \bar{I}\,\bar{B}\,\bar{A} + C\,\bar{B}\,\bar{I} + IBA$$

$$D = \bar{A} + C\,\bar{B} + C\bar{I} + \bar{C}BI$$

B) Nehmen Sie an dass die Realisierung der Ansteuerfunktion des T-FlipFlops T_b wie in Abbildung 7-2 gezeigt aussieht. Durch einen Fehler in der Produktion steht anstelle eines Toggle FlipFlops ein JK FlipFlop zur Verfügung.
Ergänzen Sie das bereits vorhandene Schaltnetz mit einer minimalen Anzahl von Elementen um das JK FlipFlop verwenden zu können.

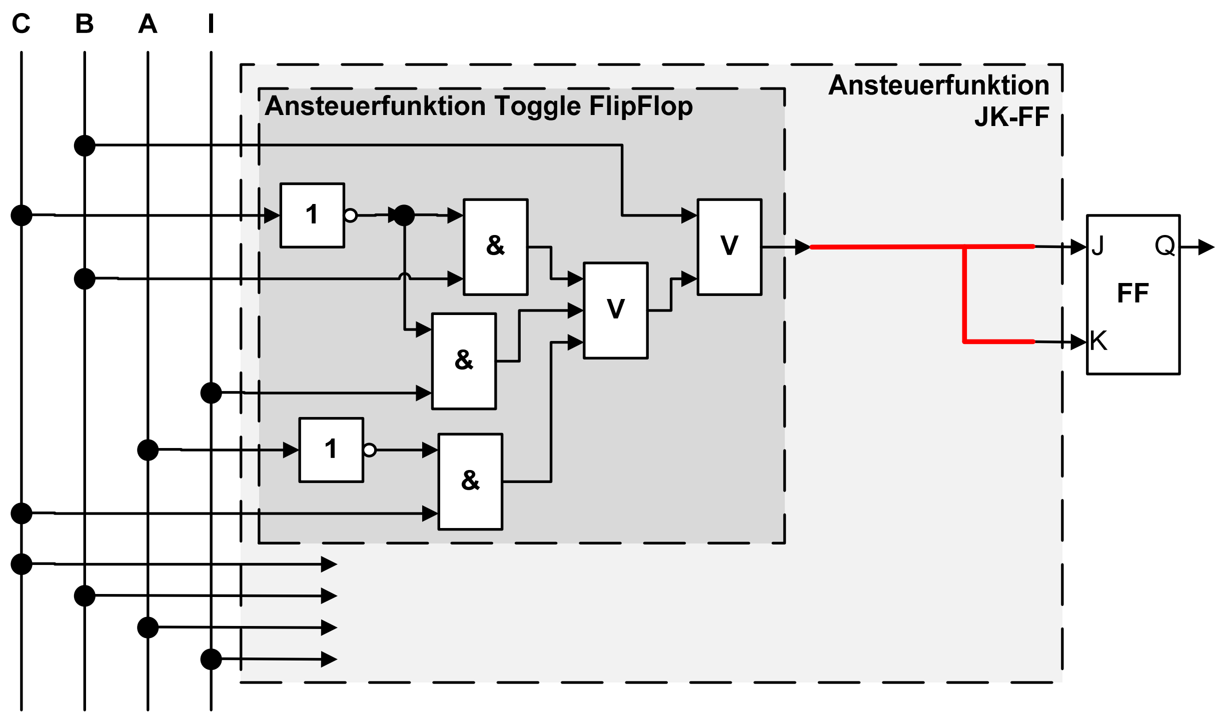


Abbildung 7-2: Ansteuerschaltung FF_2

Aufgabe 8 CMOS

Aufgabe 8.1 CMOS-Schaltungen

Abbildung 8-1 zeigt das Pull-Down Netz (G) einer CMOS-Schaltung.

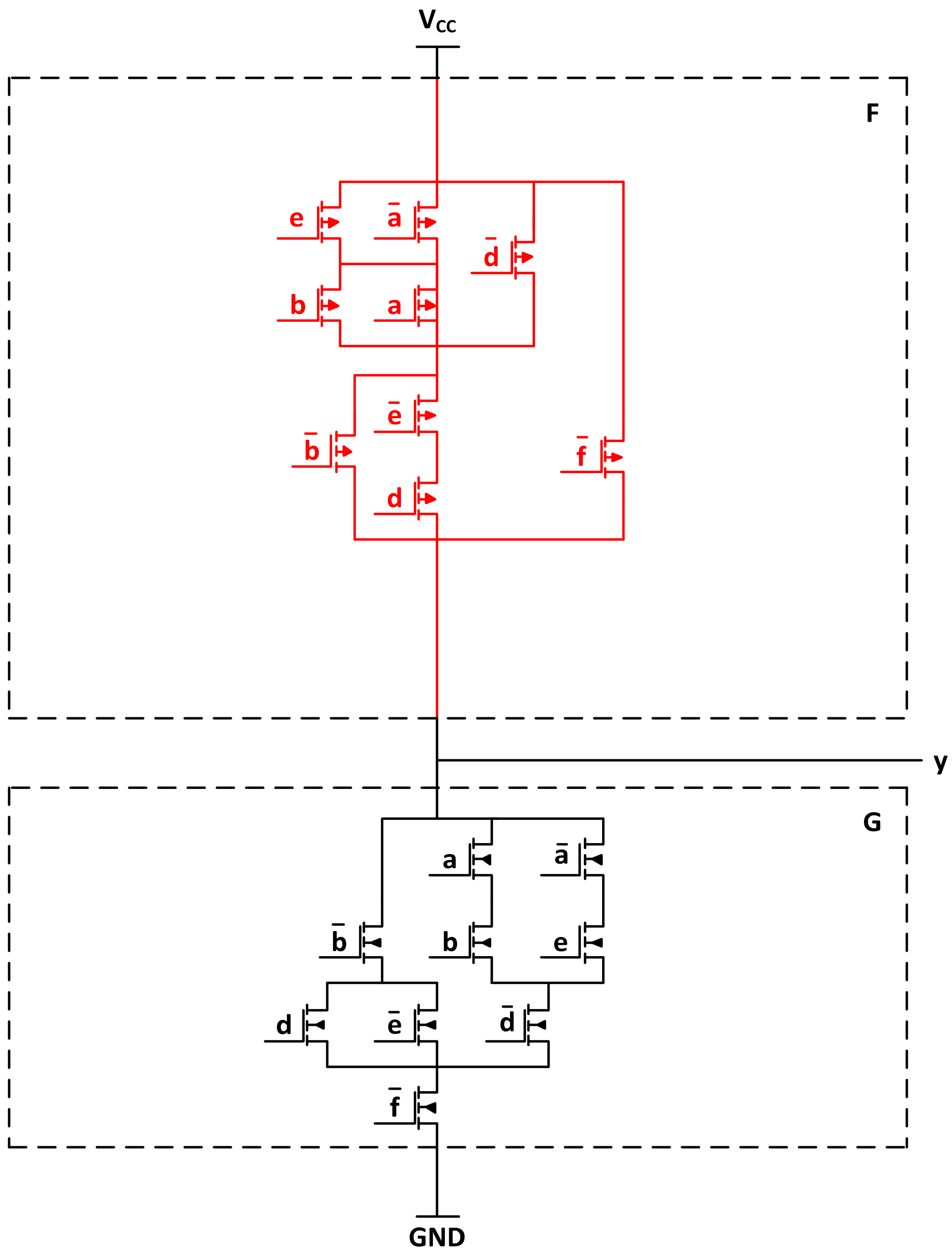


Abbildung 8-1: CMOS Schaltung

A) Ergänzen sie Abbildung 8-1 um ein wohldefiniertes, kurzschlussfreies Pull-Up Netz (F).

B) Zeigen sie rechnerisch dass ihr in Teilaufgabe A entwickeltes Netz Kurzschlussfrei ist.

$G = \overline{f} * [\, \overline{b} * (d + \overline{e}) + \overline{d} * (ab + \overline{a}e) \,] = 0$

$F = (b + \overline{d}e) * [\, (\overline{b} + \overline{a}) * (a + \overline{e}) + d \,] + f = 1$

Kurzschlussfreiheit: (F & G) = 0

$\{ \overline{f} * [\, \overline{b} * (d + \overline{e}) + \overline{d} * (ab + \overline{a}e) \,] \} * \{ (b + \overline{d}e) * [\, (\overline{b} + \overline{a}) * (a + \overline{e}) + d \,] + f \} =$

$\{\overline{fb}d + \overline{fbe} + ab\overline{df} + \overline{ade}\,\overline{f}\} * \{ (b + \overline{d}e) * [\, a\overline{b} + \overline{be} + a\overline{a} + \overline{ae} + d \,] + f \} =$

$\{\overline{fb}d + \overline{fbe} + ab\overline{df} + \overline{ade}\,\overline{f}\} * \{ ab\overline{b} + b\overline{be} + \overline{a}b\overline{e} + bd + a\overline{bd}e + \overline{bd}e\overline{e} + \overline{ad}e\overline{e} + d\overline{d}e + f\} =$

$\overline{fb}d\overline{a}b\overline{e} + \overline{fb}dbd + \overline{fb}da\overline{bd}e + \overline{fb}df + \overline{fbe}ab\overline{e} + \overline{fbe}bd + \overline{fbe}a\overline{bd}e + \overline{fbe}f + ab\overline{df}ab\overline{e} +$
$ab\overline{df}bd + ab\overline{df}a\overline{bd}e + ab\overline{df}f + \overline{ade}\,\overline{f}ab\overline{e} + \overline{ade}\,\overline{f}bd + \overline{ade}\,\overline{f}a\overline{bd}e + \overline{ade}\,\overline{f}f = 0$

ODER:

$F = \overline{G}$

Aufgabe 8.2 Wohldefinierte CMOS-Schaltungen

In Abbildung 8-2 hat sich ein Fehler in die Schaltung geschlichen, sie ist nicht wohldefiniert:

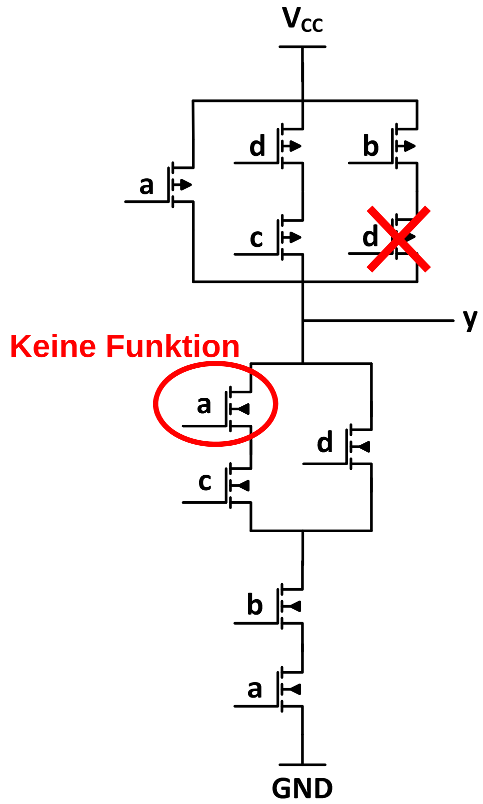


Abbildung 8-2: Fehlerhafte Schaltung

A) Geben Sie die Wahrheitstabelle der Schaltung aus Abbildung 8-2 an. *Verwenden Sie für undefinierte Zustände „UD" und für Kurzschlüsse „KS".*

a	b	c	d	y
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1

a	b	c	d	y
1	0	0	0	1
1	0	0	1	UD
1	0	1	0	1
1	0	1	1	UD
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

B) Streichen sie genau einen Transistor um die Vollständigkeit der Schaltung herzustellen.