

# GGI II Großübung

Arno Bücken, Thomas Jung, Thomas Steil  
Mitschrift: Marius Geis

17. April 2011

## Inhaltsverzeichnis

Blatt 1 . . . . .	2
Blatt 2 . . . . .	4
Blatt 3 . . . . .	6
Blatt 4 . . . . .	8
Blatt 5 . . . . .	9
Blatt 6 . . . . .	11
Blatt 7 . . . . .	14
Blatt 8 . . . . .	14
Blatt 9 . . . . .	16
Blatt 10 . . . . .	17
Blatt 11 . . . . .	18
Blatt 12 . . . . .	19
Intensivübung . . . . .	20

# Übungsblatt 1

## Aufgabe 1.1

- (a) Die Fano-Bedingung verlangt, dass kein Wort des Codes der Anfang eines anderen Codewortes sein darf (S. 12)
- (b) Ein Optimalcode ist ein Code mit möglichst kleiner mittlere Wortlänge (S. 16)
- (c) Welcher Code ist kein Block-Code? Der Morse-Code. (S. 24)
- (d) Die Hamming-Distanz  $d_m$  eines Codes lässt sich bestimmen durch Vergleich der Hamming-Distanzen aller Paare von Codewörtern. (S. 24)
- (e) Die Zahl der erkennbaren Fehler berechnet sich aus der Hamming-Distanz  $d_m$  zu

$$e = d_m - 1$$

(S. 24)

- (f) Die Anzahl der korrigierbaren Fehler bei einem Binärcode mit der Hamming-Distanz  $d_m (d_m > 1)$  ist

$$k = \left\lfloor \frac{d_m - 1}{2} \right\rfloor$$

(S. 24)

## Aufgabe 1.2

Code:  $a = 1, b = 01, c = 001, \dots, f = 000001$   
 $p_a = \frac{1}{16}, p_b = 0, p_c = \frac{1}{8}, p_d = \frac{1}{16}, p_e = \frac{1}{2}, p_f = \frac{1}{4}$

(a)

$$\bar{l} = \sum_{j=1}^n p_j l_j$$

$$\bar{l} = \frac{1}{16} \cdot 1 + 0 \cdot 2 + \frac{1}{8} \cdot 3 + \dots + \frac{1}{4} \cdot 6 = 4,6875 \text{ [bit]}$$

(b)

$a = 0001$   
 $b = 000001$   
 $c = 001$   
 $d = 00001$   
 $e = 1$   
 $f = 01$

$p_a$  und  $p_d$  sind vertauschbar

$$\bar{l} = 1,9375 \text{ [bit]}$$

(c)

$$H := \sum_{j=1}^n p_j \lg \frac{1}{p_j} = - \sum_{j=1}^n p_j \lg p_j$$

$$H = - \left( \frac{1}{16} \lg \frac{1}{16} + \frac{1}{8} \lg \frac{1}{8} + \frac{1}{16} \lg \frac{1}{16} + \frac{1}{2} \lg \frac{1}{2} + \frac{1}{4} \lg \frac{1}{4} \right) = \frac{15}{8}$$

$$= 1,875 \text{ [bit]}$$

Redundanz:  $R = H_0 - H$

$$H_0 = \lg n$$

$$H_0 = \lg 6 \approx 2,585$$

$$R = 2,585 - 1,875 \approx 0,71 \text{ [bit]}$$

relative Redundanz

$$r = \frac{R}{H_0} = \frac{0,71}{2,585} \approx 0,275 \quad (1)$$

### Aufgabe 1.3

Addiere 978 und 761 im BCD-Code

978:	0000	1001	0111	1000
761:	0000	0111	0110	0001
Übertrag:	1	1111	1100	
1. Sumation:	0001	0001	1101	1001
Pseudotetraden? :	nein	nein	ja	nein
Tetradenübertrag:	nein	ja	-	nein
Korrektur:	-	+0110	+0110	-
Übertrag:			11000	-
2. Summation:	0001	0111	0011	1100
Ergebnis:	1	7	3	9

also ist  $978 + 761 = 1739$

### Aufgabe 1.4

(a)	Note	Codewort	Paritätsbit
	1	0011	1
	2	0101	1
	3	1001	1
	4	0110	1
	5	1010	1
	6	1100	1

(b)

$$R = H_0 - H, \text{ mit } H_0 = \lg n, n = 16 = 2^4$$

$$H_0 = \lg 2^4 = 4 \text{ [bit]}$$

$$H = \sum_{j=1}^n p_j \cdot \lg p_j$$

$$= -6 \left( \frac{1}{6} \cdot \lg \frac{1}{6} \right) = -\lg \frac{1}{6} \approx 2,585$$

$$R = 4 - 2,585 \approx 1,415 \text{ [bit]}$$

Hemming Distanz  $d_m = 2$

$$\text{erkennbare Fehler } e = d_m - 1 = 2 - 1 = 1$$

$$\text{korrigierbare Fehler } k = \left\lfloor \frac{e}{2} \right\rfloor = \lfloor 0,5 \rfloor = 0$$

(c)

$$R = 5 - 2,585 \approx 2,415 \text{ [bit]}$$

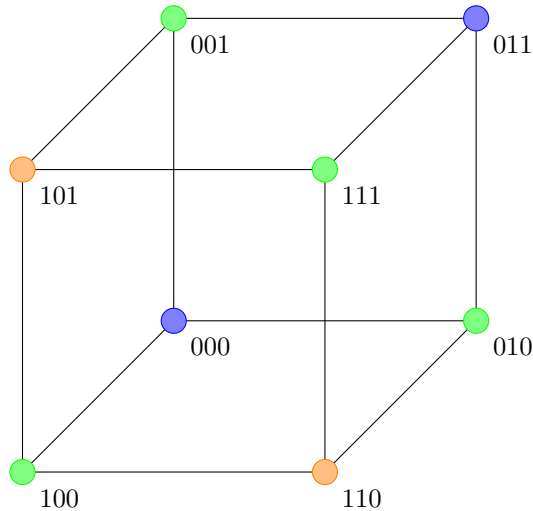
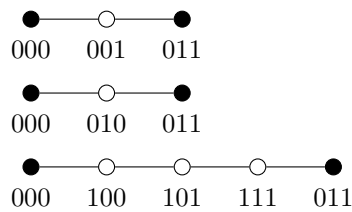
$$d_m = 2 \Rightarrow e = 1, k = 0$$

### Aufgabe 1.5

(a) 3-bit code 000 010

(b)  $d(000, 011) = 2$

(c) 001, 010, 100, 111



(d) Ja, 101 und 110

(e)  $d_m = 3$ 

000 111  
 oder 001 110  
 oder 010 101  
 oder 100 011

## Übungsblatt 2

### Aufgabe 2.1

(a) Die Auftretswahrscheinlichkeit eines  $k$ -fachen Fehlers in einem  $m$ -stelligen Codewort beträgt ...

$$\binom{m}{k} p^k (1-p)^{m-k},$$

mit Einzelfehlerwahrscheinlichkeit  $p$ 

(b) Gerade Parität bedeutet, dass...

die Summe aller gesetzten Bits (1-en) in einem Codewort gerade ist.

(c) Das Zweierkomplement eine Binärzahl erhält man durch ...

Invertieren aller Bits mit anschließender Addition von 1.

(d) Nicht darstellbar, da mit dem Einerkomplement und 7-Bit nur die Zahlen  $[-63, 63]$  darstellbar sind:

1000000, 0111111

(e) Ja! Ein Überlauf zeigt ein positives Ergebnis an

(f)  $-2^{n-1}$

**Aufgabe 2.2**

$$(a) (10011)_2 = 2^4 + 2^1 + 2^0 = (19)_{10}$$

$$(b) (101.01011)_2 = 2^2 + 2^0 + 2^{-2} + 2^{-4} + 2^{-5} = (5.34375)_{10}$$

$$(c) (234.72)_8 = 2 \cdot 8^2 + 3 \cdot 8^1 + 4 \cdot 8^0 + 7 \cdot 8^{-1} + 2 \cdot 8^{-2} = (156.90625)_{10}$$

$$(d) (ACD.81F)_{16} = 10 \cdot 16^2 + 12 \cdot 16 + 13 + 8 \cdot 16^{-1} + 16^{-2} + 15 \cdot 16^{-3} = 2765.507568$$

**Aufgabe 2.3**

Wandel um in Dual- und Hexadezimal-zahlen!

$$(a) (43406)_{10}$$

- Methode 1

$$\begin{array}{rcll}
 43406 & : & 43768 & = 1 \\
 -32768 & & & \\
 10648 & : & 16384 & = 0 \\
 & & : & 8132 = 1 \\
 -3192 & & & \\
 2446 & : & 4096 & = 0 \\
 -2048 & & & \\
 398 & : & 1024 & = 0 \\
 & & : & 512 = 0 \\
 & & : & 256 = 0 \\
 & & : & 256 = 1 \\
 -256 & & & \\
 142 & : & 128 & = 1 \\
 -128 & & & \\
 15 & : & 64 & = 0 \\
 & & : & 32 = 0 \\
 & & : & 32 = 0 \\
 & & : & 16 = 0 \\
 & & : & 8 = 1 \\
 -8 & & & \\
 6 & : & 4 & = 1 \\
 -4 & & & \\
 2 & : & 2 & = 1 \\
 -2 & & & \\
 0 & : & 1 & = 0
 \end{array}$$

1010100110001110

- Methode 2 (Horner-Schema)

$$\begin{array}{rcll}
 43406 & : 2 = & 21703 & R = 0 \\
 21703 & : 2 = & 10851 & R = 1 \\
 10851 & : 2 = & 5425 & R = 1 \\
 5425 & : 2 = & 2712 & R = 1 \\
 & \vdots & & 
 \end{array}$$

$$(43406)_{10} = (1010100110001110)_2 = (A98E)_{16}$$

$$(b) (0.78581)_{10}$$

$$\begin{array}{rcll}
 0,78581 & \cdot 2 = & 1,57152 & \rightarrow 1 \\
 0,57162 & \cdot 2 = & 1,14324 & \rightarrow 1 \\
 0,14324 & \cdot 2 = & 0,28648 & \rightarrow 0 \\
 0,28648 & \cdot 2 = & 0,572986 & \rightarrow 0 \\
 0,572986 & \cdot 2 = & 1,14592 & \rightarrow 1 \\
 0,145922 & \cdot 2 = & 0,29184 & \rightarrow 0 \\
 0,29184 & \cdot 2 = & 0,58308 & \rightarrow 0 \\
 & \vdots & & 
 \end{array}$$

$$\Rightarrow (0.78581)_{10} = (0.11001001\dots)_2 = (0.C9\dots)_{16}$$

## Aufgabe 2.4

- (a)
- das erste Bit wird als Vorzeichen interpretiert
  - zwei Darstellungen für die Zahl 0

$$[-2^{n-1} - 1; 2^{n-1} - 1] = [-31; 31]$$

- (b)
- Bitweises Komplementieren
  - zwei Darstellungen für die Zahl 0
  - Vorteil: Subtraktion durch Addition des  $K_1$

$$[-2^{n-1} - 1; 2^{n-1} - 1] = [-31; 31]$$

- (c)
- Addiere von 1 zum Einerkomplement
  - eindeutige Darstellung aller Zahlen
  - Vorteil: Subtraktion durch Addition des  $K_2$

$$[-2^{n-1}; 2^{n-1} - 1] = [-31; 31]$$

## 0.1 Rechne im jeweiligen Zahlensystem

(a)

$$\begin{array}{r} 1 \ 0 \ 1 \ 1 \ 0 \\ + \ 1 \ 1 \ 1 \ 0 \ 0 \\ 1 \ 1 \ 1 \\ \hline 1 \ 1 \ 0 \ 0 \ 1 \ 0 \end{array}$$

(b) Direkt:

$$\begin{array}{r} 1 \ 0 \ 1 \ 1 \ 0 \\ - \ 1 \ 1 \ 1 \ 0 \ 0 \\ 1 \ 1 \\ \hline - \ 1 \ 1 \ 0 \ 1 \ 0 \end{array}$$

Als Zweierkomplement:

$$\begin{array}{r} 1 \ 0 \ 1 \ 1 \ 0 \\ + \ 0 \ 0 \ 1 \ 0 \ 0 \\ 0 \ 1 \\ \hline - \ 1 \ 1 \ 0 \ 1 \ 0 \end{array}$$

(c)

$$\begin{array}{r} 1 \ 0 \ 1 \ 1 \ 0 \ . \ 1 \ 1 \ 1 \ 0 \ 0 \\ \quad 1 \ 0 \ 1 \ 1 \ 0 \\ \quad \quad 1 \ 0 \ 1 \ 1 \ 0 \\ \quad \quad \quad 1 \ 0 \ 1 \ 1 \ 0 \\ + \ 1 \ 1 \ 1 \ 1 \ 1 \\ \hline 1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \end{array}$$

(d)  $10110:111 = 11, \overline{001}$

$$\begin{array}{r} 10110:111 \\ - \underline{111} \\ 1000 \\ - \underline{111} \\ 1000 \\ - \underline{111} \\ 1 \end{array}$$

## Übungsblatt 3

### Aufgabe 3.1

### Aufgabe 3.2

(a)

$$C = E + \frac{1}{2}B^C - 1$$

VZ	M	C

$$C = E + \frac{1}{2}2^3 - 1$$

$$C = E + 3$$

(b) Größte

$$0|1111|111$$

$$Z = (0,1111)_2 \cdot 2^{7-3}$$

$$Z = (0,1111)_2 \cdot 2^4$$

$$Z = (1111)_2 \cdot 2_0$$

$$Z = 15_{10}$$

Kleinste positive

$$Z = 0|1000|000$$

$$Z = +(0,1)_2 \cdot 2^{0-3}$$

$$= (0,0001)_2$$

$$= 1 \cdot 2^{-4} - \left(\frac{1}{16}\right)_{10}$$

(c) Kleinste positive

$$0|0001|000$$

$$Z = (0,0001)_2 \cdot 2^{0-3}$$

$$= (0,000\,000\,1)_2$$

$$= 1 \cdot 2^{-7} = \frac{1}{128}$$

(d)

$$(D7)_{16} = (1101|0111)_2$$

VZ	M	C
1	1010	111

$$Z = -(0,101)_2 \cdot 2^{7-3}$$

$$= -(0,101)_2 \cdot 2^4$$

$$= (1010) \cdot 2^0 = -10$$

$$(29)_{16} \hat{=} \frac{VZ}{0} \mid \frac{M}{0101} \mid \frac{C}{001}$$

$$\hat{=} \frac{VZ}{0} \mid \frac{M}{1010} \mid \frac{C}{000} \text{ (Normalisiert)}$$

$$= (0,101)_2 \cdot 2^{-3}$$

$$= (101)_2 \cdot 2^{-6} = \frac{5}{64}$$

(e) 1,75 - 3,25

$$1,75 = (1,11)_2 = (0,111)_2 \cdot 2^1$$

$$= 0|1110|100$$

$$3,25 = (11,01)_2 = (0,1101)_2 \cdot 2^2$$

$$= 0|1101|101$$

$$1,75 = (0,111)_2 = (0,0111)_2 \cdot 2^2$$

$$= 0|0111|101$$

$$-3,25 \Rightarrow M = 1101$$

$$K_1(M) = 0010$$

$$K_2(M) = 0011$$

$$\begin{array}{r} 0111 \\ + 0011 \\ \hline 1010 \end{array}$$

kein Übertrag  $\Rightarrow \text{Erg} < 0 \Rightarrow K_2, \text{VZ}$ 

$$K_2(1010) = 0110$$

$$= 1|0110|101$$

$$= 1|1100|100 \text{ (Normalisiert)}$$

$$Z = -(0,11)_2 \cdot 2^{4-3}$$

$$= (-1,1)_2 \cdot 2^0 = -1,5$$

**Aufgabe 3.3**

(a)

$$Z = \begin{cases} 0 & \text{alle Bit} = 0 \\ (-1)^{VZ} \cdot (M)_2 \cdot 2^{C-3} & \text{sonst} \end{cases}$$

(b) Bit-Signifikanz nimmt von links nach rechts ab  $\Rightarrow$  bitweises Vergleichen von links nach rechts möglich.(c)  $1,0 = 0|011|0000$ **Aufgabe 3.4** $Z = 0,6 \Rightarrow$  Maschinenwort

$$0,6 \cdot 2 = 1,2 \rightarrow 1$$

$$0,2 \cdot 2 = 0,4 \rightarrow 0$$

$$0,4 \cdot 2 = 0,8 \rightarrow 0$$

$$0,8 \cdot 2 = 1,6 \rightarrow 1$$

$$0,6 \cdot 2 = \text{periodisch}$$

$$0,6 = (0,\overline{1001})_2$$

**Übungsblatt 4****Aufgabe 4.1**

1. Kommutativ + Distributiv
2. disjunktiv verknüpfte Konjunktionen
3. und Verknüpfung in der Schaltalgebra
4. Äquivalenz:  $a \leftrightarrow b = ab + \bar{a}\bar{b}$



## Aufgabe 4.2

$$\begin{aligned}
 b \cdot \bar{c} + \underbrace{(\bar{b} + c) \cdot c}_{\text{Absorption}} &= b \cdot \bar{c} + c \\
 &= b \cdot \underbrace{\bar{c} + c}_{\text{Reduktion}} \\
 &= b + c
 \end{aligned}$$

## Aufgabe 4.3

$$\begin{aligned}
 f(a, b, c) &= ((a \cdot b) + \bar{c}) \cdot \underbrace{((a + b) \cdot a)}_a \\
 &= ((a \cdot b) + \bar{c}) \cdot a \\
 &= \underbrace{(a \cdot b) \cdot a}_1 + \underbrace{\bar{c}a}_2 \\
 &= ab + a\bar{c} \\
 &= ab(c + \bar{c}) + a(b + \bar{b})\bar{c} \\
 &= abc + ab\bar{c} + a\bar{b}\bar{c}
 \end{aligned}$$

Absorption

Distributiv

1. Idempotenz, 2. Kommutativ

Neutrales Element, Komplement

Distributiv, Idempotenz

	a	b	c	f
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1

$$f = (a + b + c) \cdot \underbrace{(a + b + \bar{c})}_c \cdot (a + \bar{b} + c) \cdot (a + \bar{b} + \bar{c}) + (\bar{a} + b + \bar{c})$$

	0	0	0	1
0	1	5	4	
2	0	0	1	1
	3	7	6	

$\underbrace{\hspace{10em}}_a$

$\left. \begin{array}{c} \hline \\ \hline \end{array} \right| b$

$$\underbrace{\underbrace{((a \cdot b) + \bar{c})}_3 \cdot \underbrace{((a + b) \cdot a)}_5}_6$$

$$f = abc + ab\bar{c} + a\bar{b}\bar{c}$$

## Aufgabe 4.4

## Übungsblatt 5

### Aufgabe 5.1

- Sie können, müssen aber nicht zur Bildung größerer Blöcke herangezogen werden. S 60.
- Frage falsch: Es gib nur 1 Index  $i$ .  $i$  ist Anzahl nicht negierter Variablen.
- Er überdeckt einen Minterm, der von keinem anderen Primimplikanten überdeckt S 61.

## Aufgabe 5.2

$$f : \{0, \dots, 9\} \rightarrow \{0, 1\}$$

$$f(x) = \begin{cases} 1 & \text{falls } x \in \{0, 5, 6, 9\} \\ 0 & \text{sonst} \end{cases}$$

Dez	a	b	c	d	x	f(x)
0	0	0	0	0		*
1	0	0	0	1		*
2	0	0	1	0		*
3	0	0	1	1	0	1
4	0	1	0	0	1	0
5	0	1	0	1	2	0
6	0	1	1	0	3	0
7	0	1	1	1	4	0
8	1	0	0	0	5	1
9	1	0	0	1	6	1
10	1	0	1	0	7	0
11	1	0	1	1	8	0
12	1	1	0	0	9	1
13	1	1	0	1		*
14	1	1	1	0		*
15	1	1	1	1		*

- $f(x)$  ist nicht definiert für die Pseudo Tetraden.
- Nutzen diese Fälle als Don't-Cares für Minimierung

	a				
	*	1		*	
		1	*		
		*	*		
d	*	1		1	b
	c				

$$f(a, b, c, d) = a\bar{c} + \bar{a}b$$

## Aufgabe 5.3

Minimiere  $f(x)$  mittels Quine-McCluskey.

- Bestimmung aller Primimplikanten. Berücksichtige alle Fälle  $f(x) = 1$ .

$$f(x) = \bar{a}\bar{b}cd + a\bar{b}\bar{c}\bar{d} + \bar{a}\bar{b}cd + ab\bar{c}\bar{d}$$

$$\text{DC's: } \bar{a}\bar{b}\bar{c}\bar{d}, \bar{a}\bar{b}\bar{c}d, \bar{a}\bar{b}c\bar{d}, ab\bar{c}\bar{d}, acd\bar{d}, abcd$$

	Minterme		3 Variablen		2 Variablen	
$K_0$	$\bar{a}\bar{b}\bar{c}\bar{d}(0)$	✓	$\bar{b}\bar{c}\bar{d}(0, 8)$	✓	$\bar{b}\bar{c}(0 - 8, 1 - 9)$	
			$\bar{a}\bar{b}\bar{c}(0, 1)$	✓	$\bar{b}\bar{c}(0 - 1, 8 - 9)$	
			$\bar{a}\bar{b}\bar{d}(0, 2)$	✓	$\bar{a}\bar{b}(0 - 1, 2 - 3)$	
					$\bar{a}\bar{b}(0 - 2, 1 - 3)$	
$K_1$	$a\bar{b}\bar{c}\bar{d}(8)$	✓	$a\bar{b}\bar{c}(8, 9)$	✓	$a\bar{c}(8 - 9, 12 - 13)$	
	$\bar{a}\bar{b}\bar{c}\bar{d}(1)$	✓	$a\bar{c}\bar{d}(8, 12)$	✓	$a\bar{c}(8 - 12, 9 - 13)$	
	$a\bar{b}\bar{c}\bar{d}(2)$	✓	$\bar{a}\bar{b}\bar{d}(1, 3)$	✓		
			$\bar{b}\bar{c}\bar{d}(1, 9)$	✓		
			$\bar{a}\bar{b}\bar{c}(2, 3)$	✓		
$K_2$	$\bar{a}\bar{b}\bar{c}\bar{d}(3)$	✓	$a\bar{c}\bar{d}(9, 13)$	✓	$ab(12 - 13, 14 - 15)$	
	$\bar{a}\bar{b}\bar{c}\bar{d}(9)$	✓	$ab\bar{c}(12, 13)$	✓	$ab(12 - 14, 13 - 15)$	
	$a\bar{b}\bar{c}\bar{d}(12)$	✓	$a\bar{b}\bar{d}(9, 12)$	✓		
$K_3$	$a\bar{b}\bar{c}\bar{d}(13)$	✓	$abd(13, 15)$	✓		
	$a\bar{b}\bar{c}\bar{d}(14)$	✓	$abc(14, 15)$	✓		
$K_4$	$abcd(15)$	✓				

Prim-Implikanten:  $\bar{b}\bar{c}, \bar{a}\bar{b}, a\bar{c}, ab$

2. Bestimmung der wesentlichen Primimplikanten. DC's nicht! berücksichtigen.

	3	8	9	12
$\bar{a}\bar{b}$	×			
$\bar{b}\bar{c}$		×	×	
$a\bar{c}$		×	×	×
$ab$				×

wesentlich:  $\bar{a}\bar{b}$

3. Restmatrix

	8	9	12
$\bar{b}\bar{c}$	×	×	
$a\bar{c}$	×	×	×
$ab$			×

$$f(x) = \underbrace{\bar{a}\bar{b}}_{\text{wesentlich}} + \underbrace{a\bar{c}}_{\text{Rest}}$$

## Übungsblatt 6

### Aufgabe 6.1

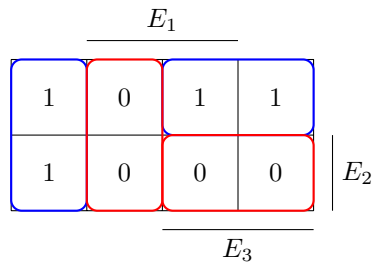
- (a) mehrere hintereinander geschaltete Schaltnetze gleicher Struktur
- (b)  $s(a, b, \ddot{u}) = (a \oplus b) \oplus \ddot{u}$
- (c) schaltet 1 von  $2^n$  Eingängen auf den Ausgang
- (d) synchron: alle Eingangsvariablen werden erst zu einem von dem Takt definierten Zeitpunkt gültig

### Aufgabe 6.2

- a) Aufstellen der Funktion

$$\begin{aligned}
 A &= M_1 \cdot M_2 \cdot M_3 \\
 M_1 &= \overline{E_1 \cdot E_2} \\
 M_2 &= \overline{E_1 \cdot \overline{E_2} \cdot \overline{E_3}} \\
 M_3 &= \overline{\overline{E_1} \cdot E_2 \cdot E_3} \\
 A &= \overline{E_1 \cdot E_2 \cdot E_1 \cdot \overline{E_2} \cdot \overline{E_3} \cdot \overline{E_1} \cdot E_2 \cdot E_3} \\
 &= (\overline{E_1} + \overline{E_2}) \cdot (\overline{E_1} + E_2 + E_3) \cdot (E_1 + \overline{E_2} + \overline{E_3}) \\
 &= (\overline{E_1} + \overline{E_2} + E_3) \cdot (\overline{E_1} + \overline{E_2} + \overline{E_3}) \cdot (\overline{E_1} + E_2 + E_3) \cdot (E_1 + \overline{E_2} + \overline{E_3}) \text{ (KNF)}
 \end{aligned}$$

Minimiere mittels KV-Diagramm



DF

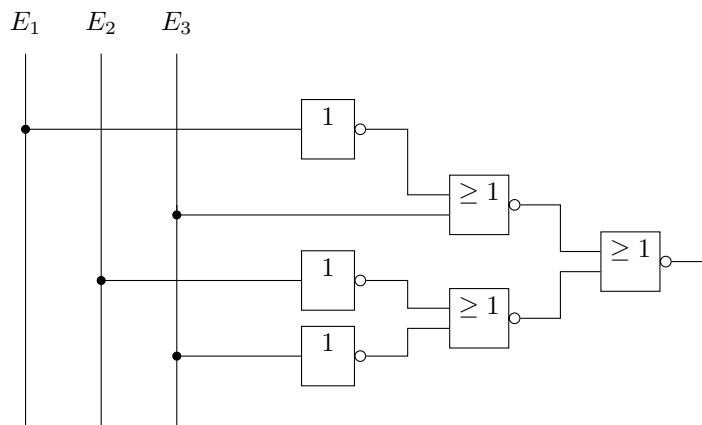
$$A = \overline{E_1} \cdot \overline{E_3} + \overline{E_2} \cdot E_3$$

KF

$$A = (\overline{E_1} + E_3) \cdot (\overline{E_2} + E_3)$$

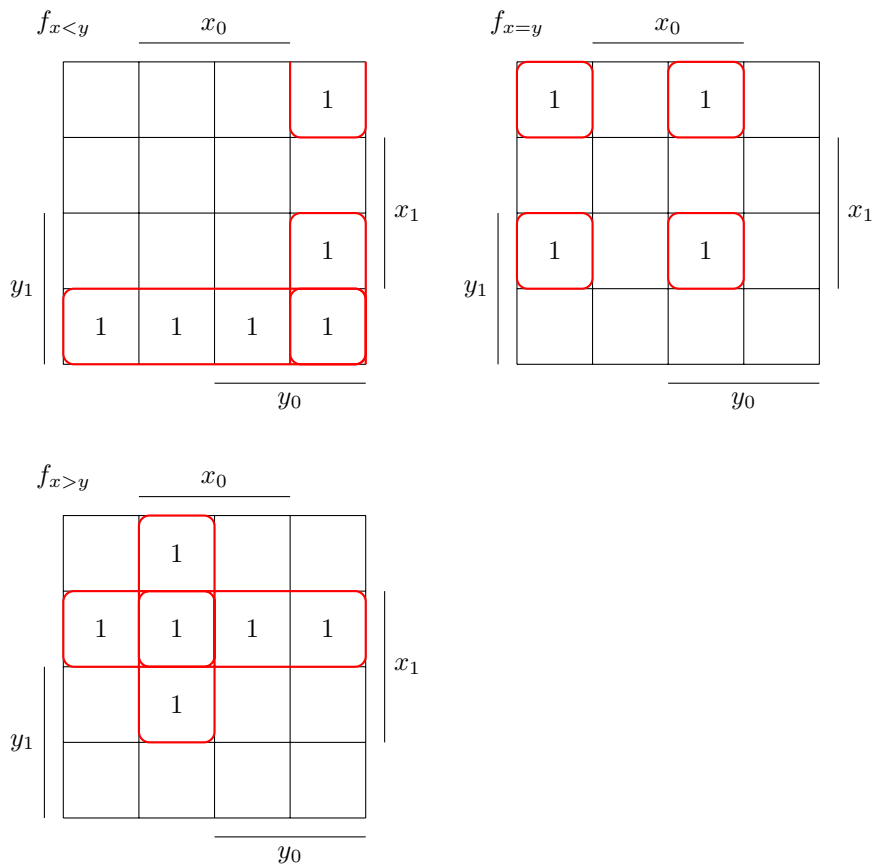
b)

$$\begin{aligned} A &= (\overline{E_1} + E_3)(\overline{E_2} + E_3) \\ &= \overline{(\overline{E_1} + E_3)(\overline{E_2} + E_3)} \quad \text{Doppelte Negation} \\ &= \overline{(\overline{E_1} + E_3) + (\overline{E_2} + E_3)} \end{aligned}$$



### Aufgabe 6.3

$y_1$	$y_0$	$x_0$	$x_1$	$x < y$	$x = y$	$x > y$
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

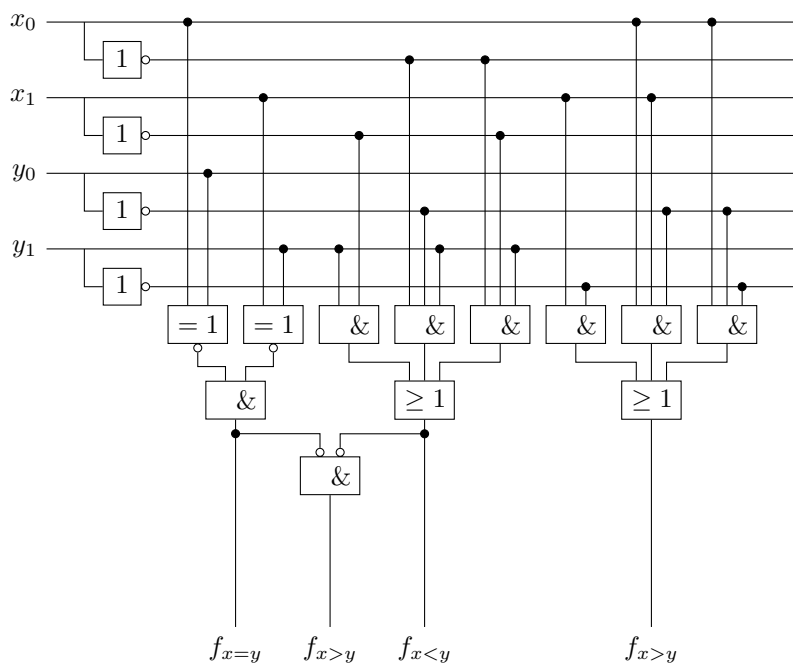


$$f_{x<y} = \bar{x}_1 y_1 + \bar{x}_0 y_0 y_1 + \bar{x}_0 \bar{x}_1 y_0$$

$$f_{x=y} = \overline{(x_0 \oplus y_0)} \cdot \overline{(x_1 \oplus y_1)}$$

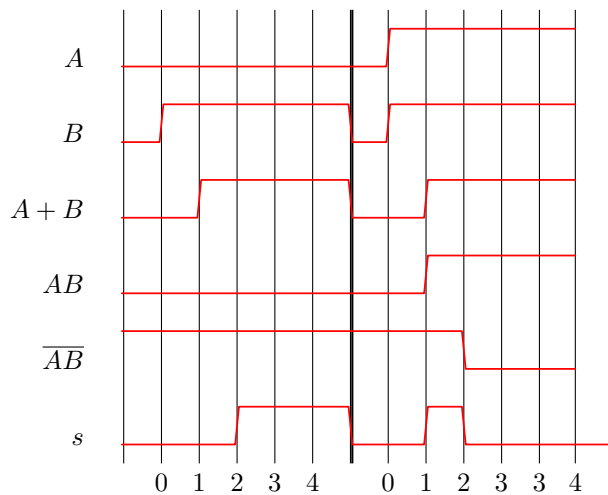
$$f_{x>y} = \overline{f_{x=y}} \cdot \overline{f_{x<y}}$$

$$= x_1 \bar{y}_1 + x_0 x_1 \bar{y}_0 + x_0 \bar{y}_0 \bar{y}_1$$



## Aufgabe 6.4

a)

b) Fehler: ungültig zwischen  $t = 1$  und  $t = 2$ Lösung: In dem der Ausgang  $s$  nur synchronisiert mit einem zusätzlichen Taktsignal weitergeleitet wird

## Übungsblatt 7

### Aufgabe 7.1

(a) Schaltwerke enthalten FlipFlops, Schaltnetze nicht

(b) auf fallender oder auf steigender Taktflanke

(c) Sie wechseln den Zustand bei  $J = K = 1$ 

(d) 
$$\begin{array}{c|c|c} Q_n & \rightarrow & Q_{n+1} & D \\ \hline * & & 1 & 1 \\ * & & 0 & 0 \end{array}$$

### Aufgabe 7.2

Weitere Funktion: Seriell-Parallel-Wandler

### Aufgabe 7.3

## Übungsblatt 8

### Aufgabe 8.1

(a) Nur von seinem Zustand

(b) Eingabemenge  $X$  enthält nur 1 Element,  $|X| = 1$ (c) Jeder Mealy als Moore darstellbar und umgekehrt  $\rightarrow$  keine!(d)  $12 \leq 16 = 2^4 \Rightarrow 4$  Flipflops

### Aufgabe 8.2

$n$	$n + 1$	
	$X_0$	$X_1$
$Z_0$	$Z_1/Y_0$	$Z_1/Y_1$
$Z_1$	$Z_1/Y_0$	$Z_0/Y_0$

(a) **1. Schritt:** Zustandsmenge?Mealy:  $X^{n=1} \times Z_{\text{Mealy}}^n \rightarrow Y_{\text{Mealy}}^{n-1}$ Moore:  $Z^{n=1} \rightarrow Y_{\text{Moore}}^{n+1}$  $\Rightarrow$  Moore Zustand Bsp: $Z_{1,\text{Mealy}} \times X_0 \rightarrow Z_{10,\text{Moore}}$ **2. Schritt:** Ausgabe zu  $Z_{ij,\text{Moore}}$ ? $= Z_i \times X_j$  (Mealy)**3. Schritt:** Folgezustände

$n$	$n+1$		$n$
	$X_0$	$X_1$	
$Z_{00}$	$Z_{10}$	$Z_{11}$	$Y_0$
$Z_{01}$	$Z_{10}$	$Z_{11}$	$Y_1$
$Z_{10}$	$Z_{10}$	$Z_{11}$	$Y_0$
$Z_{11}$	$Z_{00}$	$Z_{01}$	$Y_0$

(b)

$n$	$n+1$		$n$	$K$
	$X_0$	$X_1$		
$Z_{00}$	$Z_{10}/0$	$Z_{11}/0$	$Y_0$	$K_0^0$
$Z_{10}$	$Z_{10}/0$	$Z_{11}/0$	$Y_0$	
$Z_{11}$	$Z_{00}/0$	$Z_{01}/1$	$Y_0$	
$Z_{01}$	$Z_{10}/0$	$Z_{11}/0$	$Y_1$	$K_1^0$

$n$	$n+1$		$n$	$K$
	$X_0$	$X_1$		
$Z_{00}$	$Z_{10}/0$	$Z_{11}/1$	$Y_0$	$K_0^1$
$Z_{10}$	$Z_{10}/0$	$Z_{11}/1$	$Y_0$	
$Z_{11}$	$Z_{00}/0$	$Z_{01}/2$	$Y_0$	$K_1^1$
$Z_{01}$	$Z_{10}/0$	$Z_{11}/1$	$Y_1$	$K_2^1$

$n$	$n+1$		$n$
	$X_0$	$X_1$	
$Z_0^*$	$Z_0^*$	$Z_1^*$	$Y_0$
$Z_1^*$	$Z_0^*$	$Z_2^*$	$Y_0$
$Z_2^*$	$Z_0^*$	$Z_1^*$	$Y_1$

$Z$	$X$	$q_1$	$q_0$	$x_0$	$Z$	$q_1$	$q_0$	$J_1$	$K_1$	$J_0$	$K_0$
$Z_0^*$	$x_0$	0	0	0	$Z_0^*$	0	0	0	*	0	*
$Z_0^*$	$x_1$	0	0	1	$Z_0^*$	0	1	0	*	1	*
$Z_1^*$	$x_0$	0	1	0	$Z_0^*$	0	0	0	*	*	1
$Z_1^*$	$x_1$	0	1	1	$Z_0^*$	1	0	1	*	*	1
$Z_2^*$	$x_0$	1	0	0	$Z_0^*$	0	0	*	1	0	*
$Z_2^*$	$x_1$	1	0	1	$Z_0^*$	0	1	*	1	1	*
$\emptyset$	$x_0$	1	1	0	$\emptyset$	*	*	*	*	*	*
$\emptyset$	$x_1$	1	1	1	$\emptyset$	*	*	*	*	*	*

$J_1$   $q_1$

	*	*	
	*	*	1

$x$

$q_0$

$J_2$   $q_1$

		1	1
*	*	*	*

$x$

$q_0$

$$J_1 = xq_0$$

$$J_0 = x$$

(d) Äquivalenz zu gegebenen Automaten?

- $Z_2^*$  muss  $Z_0'$  entsprechen. ( $Y_1$ )
- $Z_1'$  geht mit  $x_0$  in sich selbst über  $\Rightarrow Z_1' \hat{=} Z_0^*$   
 $\Rightarrow Z_1^* \hat{=} Z_2'$ 

$Z_0'$	$Z_1'$	$Z_2'$	$Y_1$	✓
$Z_1'$	$Z_1'$	$Z_2'$	$Y_0$	✓
$Z_2'$	$Z_1'$	$Z_0'$	$Y_0$	✓

- Moore  $\rightarrow$  Mealy
- Menge der Zustände bleibt gleich
- Zustandsüfkt. bleibt gleich
- Ausgabefkt.: Einfach jeder Kante die Ausgabe zuordnen

$Z_0^*$	$Z_0^*/Y_0$	$Z_1^*/Y_0$
$Z_1^*$	$Z_0^*/Y_0$	$Z_2^*/Y_1$
$Z_2^*$	$Z_0^*/Y_0$	$Z_1^*/Y_0$

## Übungsblatt 9

### Aufgabe 9.1

- (a)
  - Wortaddierer
  - Schieberegister
  - parametrische Eingangsschaltung
- (b)
  - Arithmetische Operationen
  - Logische Operationen
- (c) V: Verlassen der Zahlenbereiche im 2-er Komplement  
C: Überlauf
- (d)

$$7_{10} + 5_{10} = 12_{10}$$

	0	1	1	1
+	0	1	0	1
	0	1	0	1
	1	1	0	0

(e)  $N = V \Rightarrow x \geq y \quad Z = 1 \Rightarrow x = 7$

(f)  $n + k - 1$

### Aufgabe 9.2

1.

Schritt	Mikrobefehl	Codierung				Register	Flags				Bemerkung
		$F_2F_1F_0$	$X_1X_0$	$Y_1Y_0$	$R_1R_0$		$C$	$N$	$Z$	$V$	
1	$4_{10} \rightarrow R_1$	000	00	**	01	$R_1 = 0100_2$	0	0	0	0	Eingabe = $4_{10}$
2	$3_{10} \rightarrow R_2$	000	00	**	10	$R_2 = 0011_2$	0	0	0	0	Eingabe = $3_{10}$
3	$\overline{R_1} \rightarrow R_1$	110	01	**	01	$R_1 = 1011_2$	0	1	0	0	$\} K_2$
4	$R_1 + 1 \rightarrow R_1$	101	01	**	01	$R_1 = 1100_2$	0	1	0	0	
5	$R_2 + R_1 \rightarrow R_3$	010	10	01	11	$R_3 = 1111_2$	0	1	0	0	Ergebnis = $-1$

2.

Schritt	Mikrobefehl	Codierung				Register	Flags				Bemerkung
		$F_2F_1F_0$	$X_1X_0$	$Y_1Y_0$	$R_1R_0$		$C$	$N$	$Z$	$V$	
1	$5_{10} \rightarrow R_3$	000	00	**	11	$R_3 = 0101_2$	0	0	0	0	Eingabe = $5_{10}$
2	$\overline{R_3} \rightarrow R_3$	110	11	**	11	$R_3 = 1010_2$	0	1	0	0	$\} K_2$
3	$R_3 + 1 \rightarrow R_3$	101	11	**	11	$R_3 = 1011_2$	0	1	0	0	
4	$R_3 + E \rightarrow$	010	11	00	00	$R = 0000_2$	1	0	1	0	Eingabe = $5_{10}$

3.

Schritt	Mikrobefehl	Codierung				Register	Flags				Bemerkung
		$F_2F_1F_0$	$X_1X_0$	$Y_1Y_0$	$R_1R_0$		$C$	$N$	$Z$	$V$	
1	$-8_{10} \rightarrow R_1$	000	00	**	01	$R_1 = 1000_2$	0	1	0	0	Eingabe = $-8_{10}$
2	$7 \rightarrow R_2$	000	00	**	10	$R_2 = 0111_2$	0	0	0	0	Eingabe = $7_{10}$
3	$\overline{R_2} \rightarrow R_2$	110	10	**	10	$R_2 = 1000_2$	0	1	0	0	$\} K_2$
4	$R_2 + 1 \rightarrow R_2$	101	10	**	10	$R_2 = 1001_2$	0	1	0	0	
5	$R_1 + R_2 \rightarrow$	010	01	10	00		1	0	0	1	$x < y$



4.

Schritt	Mikrobefehl	Codierung				Register	Flags				Bemerkung
		$F_2 F_1 F_0$	$X_1 X_0$	$Y_1 Y_0$	$R_1 R_0$		$C$	$N$	$Z$	$V$	
1	$11_{10} \rightarrow R_2$	000	00	**	10	$R_2 = 1011_2$	0	1	0	0	Eingabe = $11_{10}$
2	$R_2 + R_2 \rightarrow R_2$	010	10	10	10	$R_2 = 0110_2$	1	0	0	1	Links-Shift
3	$R_2 + R_2 \rightarrow R_2$	010	10	10	10	$R_2 = 1100_2$	0	1	0	1	Links-Shift

## Übungsblatt 10

### Aufgabe 10.1

- a)
  - maschinennah
  - maschinenspezifisch
  - „lesbar“
- b) LDS Rd, k : 10001 000d dddd 0000 kkkkkkkkkkkkkkkk  
16 bit
- c) Akku
- d) IJMP, ICALL, NOP (16 bit)

### Aufgabe 10.2

- a) Lade die CPU-spezifischen Informationen
- b) Initialisiere den Stack Pointer (Voreinstellung vom SP: \$0000)
- c)
  - 1. Durchlauf:  $R17 = 1 \rightarrow 2\text{er Komplement} \rightarrow R19 = -1$
  - 2. Durchlauf:  $R17 = 2 \rightarrow \text{kein 2er Komplement} \rightarrow R19 = -1 + 2 = 1$
  - 3. Durchlauf:  $R17 = 3 \rightarrow 2\text{er Komplement} \rightarrow R19 = 1 - 3 = -2$
- d)  $\sum_{i=1}^{R16} (-1)^i \cdot i \rightarrow R16$

### Aufgabe 10.3

```

249: LDI R16,10
250: LDI R17,0
251: ADD R17,R16
252: DEC R16
253: ORI R16,0
254: BREQ 1 (end)
255: RJMP -5 (loop)
256: MOV R16,R17
257: LDI R17,0

```

$$\sum_{i=1}^{R16} i$$

### Aufgabe 10.4

- a)  $R0 = R1$
- ```

        CP      R0, R1
        BREQ    if
else:    ...
        RJMP    end
if:      ...
end:     ...

```

b)  $R0 \geq R1$ 

```

        CP    R0, R1
        BRLO  else
if:     ...
        RJMP  end
else:   ...
end:    ...

```

c)  $R0 \leq R1$ 

```

        CP    R0, R1
        BREQ  if
        BRLO  if
else:   ...
        RJMP  end
if:     ...
end:    ...

```

alternativ:

```

        CP    R1, R0
        BRLO  else
if:     ...
        RJMP  end
else:   ...
end:    ...

```

d)  $R0 > R1$ 

```

        CP    R1, R0
        BRLO  if
else:   ...
        RJMP  end
if:     ...
end:    ...

```

## Übungsblatt 11

### Aufgabe 11.1

- a)
  - Indirekte Adressierung
  - Relative Adressierung

b) LIFO

- c)
  - Register
  - Stack
  - Speicher
  - (Flags)

### Aufgabe 11.2

```

.equ vec = 0x0100
    LDI ZH, 0          ; Z-Register inisialisieren
    LDI ZL, 0          ;
loop: STD Z+vec, ZL    ; An Stelle Z-Rg + vec ZL schreiben
    INC ZL             ; ZL erhöhen
    CPI ZL, 5          ; ZL mit Obergrenze vergleichen
    BREQ end          ; Schleifenende?
    RJMP loop         ; Sprung zum Schleifenanfang
end:  RJMP -1

```

>>Flow Charts für Pre-checked und Post-checked loops

## Übungsblatt 12

### Aufgabe 12.1

- a) 29.7.2010 14:30
- b) Keine
- c)
  - Freitextfragen (9 Fragen)
  - Aufgabenteil (3 Aufgaben 45 Punkte) ALU/Assembler, Automaten/Flipflops, Codierung/Zahlendarstellung/Logik
- d) 90 Minuten + 5 Minuten Einlesezeit
- e)
  - Kein Bleistift
  - Kein roten/grünen Stift
  - Kein Tipp-Ex

### Aufgabe 12.2

```
.include    "m8def.inc"
.def       function    = r16
.def       input       = r17
.def       stacksize   = r18
.def       temp        = r19
.def       error       = r20
.def       retadr1     = r21
.def       retadr2     = r22
.def       operand1    = r23
.def       operand2    = r24
.equ       func_nop     = 0
.equ       func_input   = 1
.equ       func_add     = 2
.equ       func_sub     = 4
.equ       func_mul     = 9
.equ       func_reset   = 16
.equ       max_stack   = 50

        LDI    temp,LOW(RAMEND)
        OUT    SPL, temp
        LDI    temp,HIGH(RAMEND)
        OUT    SPH, temp

        LDI    stacksiez, 0
        LDI    error, 0

Start:   IN     function, 37
        CPI    function, function_nop
        BREQ   Start
        CPI    function, function_input
        BREQ   Call_Punput
        RJMP   Post_Input
Call_Input: RCALL Sub_Input
Post_Input: CPI    function, function_add
        ...

        CPI    error, 0
        BREQ   Output_Val
        OUT    43,error
        LDI    error, 0
        RJMP   Start
Output_Val: OUT    43, error
        POP    temp
```

```

        OUT    42, temp
        PUSH   temp
Sub_Input: CPI    stacksize, max_stack
        BRLO   Cont_Input
        LDI    error, 1
        RET
Cont_Input: POP    retadr1
        POP    retadr2
        IN     temp, 38
        PUSH   temp
        INC    stacksize
        PUSH   retadr2
        PUSH   retadr1
        RET

Sub_Rset:  POP    retadr1
        POP    retadr2
        LDI    temp, 0
Loop_Reset: CPI    temp, stacksize
        BREQ   End_Reset
        POP    operand1
        INC    temp
        RJMP   Loop_Reset
End_Reset: LDI    stacksize, 0
        PUSH   retadr2
        PUSH   retadr1
        RET

```

## Intensivübung

### Teilaufgabe 1

1) 2-Adress

2)  $Y$

3)

| Schritt | Mikrobefehl             | Codierung   |          |          | Flag |     |     |     | Register       | Bemerkung          |
|---------|-------------------------|-------------|----------|----------|------|-----|-----|-----|----------------|--------------------|
|         |                         | $F_2F_1F_0$ | $X_1X_0$ | $Y_1Y_0$ | $C$  | $N$ | $Z$ | $V$ |                |                    |
| 1       | $R_2 = 5_{10}$          | 010         | 00       | 10       | 0    | 0   | 0   | 0   | $R_2 = 0101_2$ | Eingabe = $5_{10}$ |
| 2       | $R_2 = \text{NOT } R_2$ | 110         | 10       | 10       | 0    | 1   | 0   | 0   | $R_2 = 1010_2$ |                    |
| 3       | $R_2 = R_2 + R_2$       | 101         | 10       | 10       | 1    | 0   | 0   | 1   | $R_2 = 0100_2$ | Linksshift         |
| 4       | $R_2 = R_2 + R_2$       | 101         | 10       | 10       | 0    | 0   | 0   | 1   | $R_2 = 1000_2$ | Linksshift         |

4)

| Maschinenbefehl     | Mnemonic            | Kommentar                                     |
|---------------------|---------------------|-----------------------------------------------|
| 0000 0000 0000 0000 | NOP                 | Tue nichts                                    |
| 0000 1100 0011 0011 | ADD R3, R3 / LSL R3 | Addiere R3 zu R3, Ergebnis in R3              |
| 1001 0101 0000 1001 | ICALL               | Rufe Unterprogramm an Adresse in Z steht auf, |
|                     |                     | Rücksprungadresse auf Stack                   |

5) ICALL sichert die Rücksprungadresse auf den Stack und kann deswegen zum Sprung in Unterprogramme verwendet werden. Mit IJMP kann immer nur zur selben Stelle zurückgesprungen werden.

```

6)      LDI      zaehler , 1          ; Zähler Initialisieren
loop:   OUT      37, zaehler          ; Zähler an Port 37 ausgeben
        INC      zaehler              ; Zähler erhöhen
        CPI      zaehler, 6           ; Abbruchbedingung
        BREQ     end                  ; Falls ja: Ende
        RJMP     loop                 ; Falls nein: weiter in Schleife
end:    RJMP     end

```

## Teilaufgabe 2 - Zahlendarstellung, Codierung, Logik

1)

$$\begin{aligned}
 (((1 \cdot 6 + 2) \cdot 6 + 3) \cdot 6 + 4) \cdot 6 + 5 &= ((8 \cdot 6 + 3) \cdot 6) + 5 \\
 &= (51 \cdot 6 + 4) \cdot 6 + 5 \\
 &= (306 + 4) \cdot 6 + 5 = 1860 + 5
 \end{aligned}$$

2)

$$\begin{aligned}
 (a + b) \cdot (\bar{a} + c) &= a\bar{a} + ac + a\bar{b} + bc && \text{(Distributivgesetz)} \\
 &= 0 + ac + \bar{A}b + bc && \text{(Komplement)} \\
 &= ac + \bar{a}b + bc(a + \bar{a}) && \text{(Komplement/NE)} \\
 &= ac + \bar{a}b + abc + \bar{a}bc && \text{(Distributivgesetz)} \\
 &= ac + abc + \bar{a}bc + \bar{a}bc && \text{(Kommutativ)} \\
 &= ac(1 + b) + \bar{a}(1 + b) && \text{(Distributiv/NE)}
 \end{aligned}$$

Nebenbeweis S.49

$$\begin{aligned}
 &= ac(1) + \bar{a}b(1) && a + 1 = 1 \\
 &= ac + \bar{a}b && \text{NE}
 \end{aligned}$$

QED

3)

| $a$ | $b$ | $c$ | $f$ | Ungerades Paritätsbit |
|-----|-----|-----|-----|-----------------------|
| 0   | 0   | 0   | 0   | 1                     |
| 0   | 0   | 1   | 0   | 0                     |
| 0   | 1   | 0   | 1   | 1                     |
| 0   | 1   | 1   | 1   | 0                     |
| 1   | 0   | 0   | 1   | 1                     |
| 1   | 0   | 1   | 0   | 1                     |
| 1   | 1   | 0   | 1   | 0                     |
| 1   | 1   | 1   | 1   | 1                     |

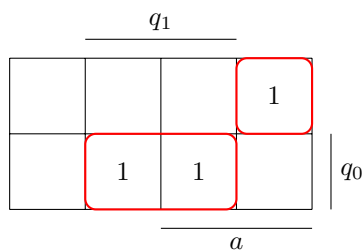
4)  $d = 4$ 

## Teilaufgabe 3 - Schaltungen, Automaten und Flipflops

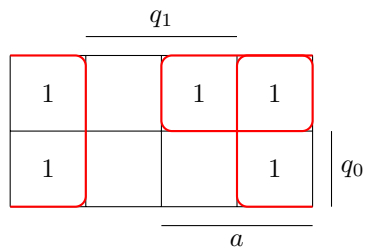
1)

| $n+1$ | $n$  |      | $n+1$ |      | $n$  |      |
|-------|------|------|-------|------|------|------|
| $a$   | $q0$ | $q1$ | $q0$  | $q1$ | $D0$ | $D1$ |
| 0     | 0    | 0    | 0     | 1    | 0    | 1    |
| 0     | 0    | 1    | 0     | 1    | 0    | 1    |
| 0     | 1    | 0    | 0     | 0    | 0    | 0    |
| 0     | 1    | 1    | 1     | 0    | 1    | 0    |
| 1     | 0    | 0    | 1     | 1    | 1    | 1    |
| 1     | 0    | 1    | 0     | 1    | 0    | 1    |
| 1     | 1    | 0    | 0     | 1    | 0    | 1    |
| 1     | 1    | 1    | 1     | 0    | 1    | 0    |

2)



$$D_0 = q_0q_1 + a\bar{q}_0\bar{q}_1$$



$$D_1 = \bar{q}_0 + a\bar{q}_1$$

$$3) f = abc + ab\bar{c} + a\bar{b} + \bar{a}bc + \bar{a}b\bar{c} + \bar{a}\bar{b}\bar{c}$$

| Anzahl nicht negierter Variable | 3er Terme                                                      | 2er Terme                                   | 1er Terme                                |
|---------------------------------|----------------------------------------------------------------|---------------------------------------------|------------------------------------------|
| 0                               | $\bar{a}\bar{b}\bar{c}$ (0)✓                                   | $\bar{a}\bar{c}$ (0, 2)                     |                                          |
| 1                               | $\bar{a}b\bar{c}$ (2)✓                                         | $b\bar{c}$ (2, 6)✓<br>$\bar{a}b$ (2, 3)✓    | $b$ (2 - 6, 3 - 7)<br>$b$ (2 - 3, 6 - 7) |
| 2                               | $ab\bar{c}$ (6)✓<br>$\bar{a}b\bar{c}$ (5)✓<br>$\bar{a}bc$ (3)✓ | $ab$ (6, 7)✓<br>$ac$ (5, 7)<br>$bc$ (3, 7)✓ |                                          |
| 3                               | $abc$ (7)✓                                                     |                                             |                                          |

Primimplikanten:

$\bar{a}\bar{c}, b, ac$

|                  | 0 | 2 | 3 | 5 | 6 | 7 |
|------------------|---|---|---|---|---|---|
| $b$              |   | × | ⊗ |   | × | × |
| $ac$             |   |   |   | ⊗ |   | × |
| $\bar{a}\bar{c}$ | ⊗ | × |   |   |   |   |

Alle Primimplikanten sind wesentlich!

$$f(a, b, c) = b + \bar{a}\bar{c} + ac$$

4) Moore  $\rightarrow$  Mealy

- Menge der Zustände bleibt unverändert
- Zustandsänderungsfunktion bleibt unverändert

