

# KLAUSUR Informationstechnik

Wintersemester 2015/2016

## MUSTERLÖSUNG

Prüfungsfach:	Informationstechnik
Studiengang:	Wirtschaftsinformatik, Softwaretechnik
Semestergruppe:	WKB 1, SWB 1
Fachnummer:	1051002
Erlaubte Hilfsmittel:	keine
Zeit:	90 min.

### **Wichtiger Hinweis für die Bearbeitung der Aufgaben:**

Schreiben Sie bitte Ihre Lösungen möglichst auf die Aufgabenblätter.

Sollte der vorgesehene Platz nicht reichen, verwenden Sie bitte jeweils die Rückseite.

Viel Erfolg wünscht Ihnen.

Reiner Marchthaler

# 1 Boolesche Algebra

## 1.1 Schaltungsanalyse

(5 Punkte)

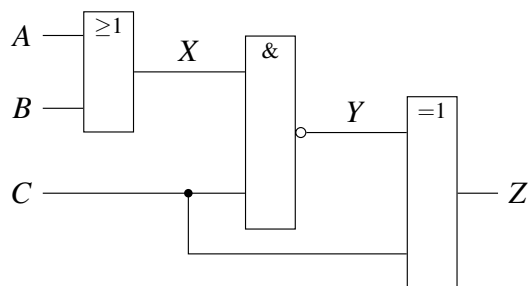


Abbildung 1: Zu untersuchende Schaltung

Geben Sie zu der Schaltung in Abbildung 1 die dazugehörige Boolesche Gleichung an.

$$Y = \overline{(A \vee B)} \wedge C$$

(3 Punkte)

Wie ist die Funktionslänge  $l$  und die Schachteltiefe  $k$  der Schaltung aus Abbildung 1?

$$l = 6$$

(1 Punkte)

$$k = 3$$

(1 Punkte)

## 1.2 Funktionstabelle

(4 Punkte)

Bestimmen Sie die Funktionstabelle der Schaltung aus Abbildung 1?

(4 Punkte)

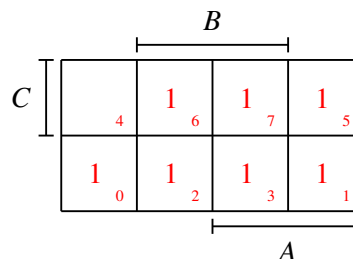
	C	B	A	$X = A \vee B$	$X \wedge C$	$Y = \overline{X} \wedge \overline{C}$	$Z = Y \leftrightarrow C$
0	0	0	0			1	1
1	0	0	1	1		1	1
2	0	1	0	1		1	1
3	0	1	1	1		1	1
4	1	0	0			1	
5	1	0	1	1	1		1
6	1	1	0	1	1		1
7	1	1	1	1	1		1

### 1.3 Minimierung

(9 Punkte)

Bestimmen Sie die disjunktive Minimalform  $Z_{DMF}$  der Schaltung aus Abbildung 1 mit Hilfe der erstellten Funktionstabelle. Übertragen zuerst Ihr Ergebnis aus Aufgabe 1.2 in die Tabelle 1 und füllen Sie dann das KV-Diagramm aus.

	C	B	A	Z
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1



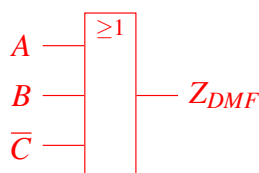
$$Z_{DMF} = A \vee B \vee \bar{C}$$

(5 Punkte)

Tabelle 1: Ergebnis aus Aufgabe 1.2

Zeichnen Sie Schaltung der oben bestimmten disjunktiven Minimalform  $Z_{DMF}$ ?

(2 Punkte)



Wie ist die Funktionslänge  $l$  und die Schachteltiefe  $k$  der zur disjunktiven Minimalform  $Z_{DMF}$  gehörenden Schaltung?

$$l = 3$$

(1 Punkte)

$$k = 1$$

(1 Punkte)

Prüfungsfach: <b>Informationstechnik</b>	Wintersemester 2015/2016	<div style="border: 1px solid black; padding: 5px; display: inline-block;"> <b>Hochschule Esslingen</b>  University of Applied Sciences </div>
Name, Vorname:	Mat.-Nr.:	

## 2 Zahlendarstellung und Codierung

### 2.1 Subtraktion in Festkommadarstellung

(10 Punkte)

Gegeben sind die beiden Hexadezimal-Zahlen  $Z_1 = (\mathbf{AF})_{16}$  und  $Z_2 = (\mathbf{81})_{16}$ .

Wandeln Sie die Hexadezimal-Zahlen  $Z_1$  und  $Z_2$  in eine Zahl zur Basis 10 um, falls

1. eine **Dualcodierung** (Betragszahl) zugrundeliegt:

(2 Punkte)

$$Z_1 = 1010\ 1111 = (\mathbf{175})_{10}$$

$$Z_2 = 1000\ 0001 = (\mathbf{129})_{10}$$

2. eine **2er-Komplement-Codierung** zugrundeliegt:

(2 Punkte)

$$Z_1 = 175 - 256 = (\mathbf{-81})_{10}$$

$$Z_2 = 129 - 256 = (\mathbf{-127})_{10}$$

3. eine **Offset-Code-Codierung** zugrundeliegt:

(2 Punkte)

$$Z_1 = 175 - 128 = (\mathbf{+47})_{10}$$

$$Z_2 = 129 - 128 = (\mathbf{+1})_{10}$$

4. eine **Vorzeichen-Betrags-Codierung** zugrundeliegt:

(2 Punkte)

$$Z_1 = 1010\ 1111 = (\mathbf{-47})_{10}$$

$$Z_2 = 1000\ 0001 = (\mathbf{-1})_{10}$$

Berechnen Sie die Subtraktion der beiden Zahlen  $Z = Z_1 - Z_2$  und stellen Sie das Ergebnis als Hexadezimal-Zahl dar.

(2 Punkte)

$$Z_1 = 1010\ 1111$$

$$Z_2 = 1000\ 0001$$

-----

$$Z = \mathbf{0010\ 1110} = (\mathbf{2E})_{16}$$

## 2.2 Zahlendarstellung nach IEEE 754

(10 Punkte)

Eine Gleitkommazahl in einfacher Genauigkeit (32 Bit) ist nach IEEE 754 wie folgt codiert:

Bits	1	8	23
	VZ von $M$	$E + 127$	$ M $ ohne $m_0$

- Das Bit 31 (MSB) kennzeichnet das Vorzeichen.
- Die nächsten 8 Bit 30...23 geben den Exponenten an (Offsetdarstellung um 127).
- Die nächsten 23 Bit 22...0 geben die normalisierte Mantisse ohne die Vorkomma-Eins an.

Abbildung 2: Darstellung von Gleitkommazahl in einfacher Genauigkeit (32 Bit) nach IEEE 754

<b>normalisierte Zahl</b>	$\pm$	$0 < \text{Exponent} < \text{max}$	Mantisse beliebig
<b>denormalisierte Zahl</b>	$\pm$	0000 0000	Mantisse nicht alle Bits 0
<b>Null</b>	$\pm$	0000 0000	0...0
<b>Unendlich</b>	$\pm$	1111 1111	0...0
<b>NaN</b>	$\pm$	1111 1111	Mantisse nicht alle Bits 0

Tabelle 2: Sonderfälle Gleitkommazahl in einfacher Genauigkeit (32 Bit) nach IEEE 754

Wie sieht die Zahl  $-\infty$  (minus unendlich) in der Gleitkommazahlendarstellung in einfacher Genauigkeit nach IEEE 754 in hexadezimaler Schreibweise aus?

VZ      Exp.      Mant.

1 111 1111 1000 0000 0000 0000 0000 0000

1111 1111 1000 0000 0000 0000 0000 0000

F      F      8      0      0      0      0      0

(3 Punkt)

Ergebnis: (FF80 0000) Hex

(2 Punkte)

Geben Sie ein Beispiel für die „Zahl“ NaN (Not a Number) in der Gleitkommazahlendarstellung in einfacher Genauigkeit nach IEEE 754 in hexadezimaler Schreibweise an?

VZ      Exp.      Mant.

1 111 1111 1111 0000 0000 0000 0000 0000

1111 1111 1111 0000 0000 0000 0000 0000

F      F      F      0      0      0      0      0

(3 Punkt)

Ergebnis: (FFF0 0000) Hex

(2 Punkte)

Prüfungsfach: <b>Informationstechnik</b>	Wintersemester 2015/2016	<b>Hochschule Esslingen</b> University of Applied Sciences
Name, Vorname:	Mat.-Nr.:	

## 2.3 Blockcodes

(18 Punkte)

Gegeben ist die Generatormatrix

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

Wie viele Nachrichtenstellen  $m$  haben Codewörter die mit der obigen Generatormatrix  $\mathbf{G}$  erzeugt werden können?

Die Generatormatrix hat allgemein die Dimension:  $m \cdot (m + k)$ .

In diesem Fall ist die Dimension von  $\mathbf{G}$ :  $2 \cdot 6$ .

Somit ist  $m = 2$

(2 Punkte)

Bestimmen Sie alle mit der Generatormatrix  $\mathbf{G}$  erzeugbare Codewörter?

Mit  $m = 2$  lassen sich  $2^m = 4$  Codewörter erzeugen

		1	0	0	0	1	1	
		0	1	1	0	1	0	
0	0	0	0	0	0	0	0	$\mathbf{Y_1 = 000000}$
0	1	0	1	1	0	1	0	$\mathbf{Y_2 = 011010}$
1	0	1	0	0	0	1	1	$\mathbf{Y_3 = 100011}$
1	1	1	1	1	0	0	1	$\mathbf{Y_4 = 111001}$

(6 Punkte)

Wie groß ist die Hammingdistanz des mit der Generatormatrix  $\mathbf{G}$  erzeugten Codes?

$$d(\mathbf{Y_1}, \mathbf{Y_2})=3, \quad d(\mathbf{Y_2}, \mathbf{Y_3})=4, \quad d(\mathbf{Y_3}, \mathbf{Y_4})=3$$

$$d(\mathbf{Y_1}, \mathbf{Y_3})=3, \quad d(\mathbf{Y_2}, \mathbf{Y_4})=3$$

$$d(\mathbf{Y_1}, \mathbf{Y_4})=4$$

$$\text{Hammingdistanz } h = \min_{\substack{(Y_i), (Y_j) \in M_2 \\ i \neq j}} \{ d(X_i, X_j) \} = \underline{3}$$

(3 Punkte)

Prüfungsfach: <b>Informationstechnik</b>	Wintersemester 2015/2016	<b>Hochschule Esslingen</b> University of Applied Sciences
Name, Vorname:	Mat.-Nr.:	

Wie viele Bitfehler können sicher erkannt werden?

$$e^* = h - 1 = \underline{2}$$

(2 Punkt)

Wie viele Bitfehler können sicher erkannt und korrigiert werden?

$$e = \frac{h-1}{2} = \underline{1}$$

(2 Punkte)

Bestimmen Sie die Parity-Check-Matrix  $\mathbf{H}^T$

Hinweis:  $\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$

$$\mathbf{H}^T = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(3 Punkte)

### 3 Hardware

Die in Abbildung 3 dargestellte 8 Bit-ALU enthält neben einem 8 Bit Addierer, eine 8 Bit-Logik-Einheit, ein 8-faches AND-Gatter sowie einen Block „Status“ zur Bildung des Carry-Flags (CF), Overflow-Flags (OF), Zero-Flags (Z) und Negativ-Flags (N).

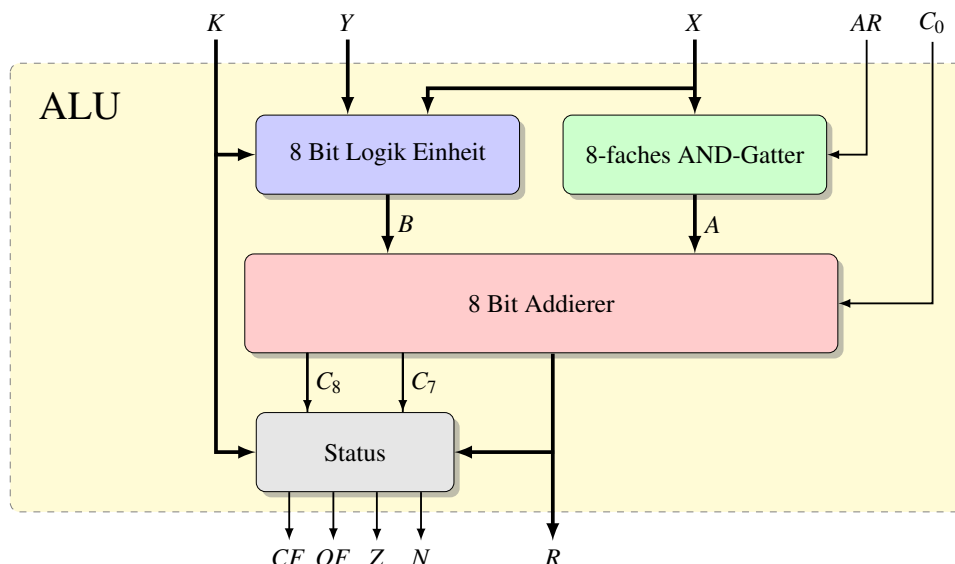


Abbildung 3: Aufbau 8-Bit ALU

Die Signale haben folgende Bitbreite:

Signalname	A	B	X	Y	R	K	AR	C <sub>0</sub>	C <sub>7</sub>	C <sub>8</sub>	CF	OF	Z	N
Breite in Bit	8	8	8	8	8	4	1	1	1	1	1	1	1	1

Tabelle 3: Bitbreite der Signale

Die gültigen Steuerworte des Steuersignals **K** sind der Tabelle 4 zu entnehmen.

Steuerwort (K)	Ergebnis für Stelle $B_i$	Logik-Funktion
(0000) = $0_H$	$B_i = 0$	Kontradiktion
(0001) = $1_H$	$B_i = 1$	Tautologie
(0010) = $2_H$	$B_i = X_i$	Identität X
(0011) = $3_H$	$B_i = Y_i$	Identität Y
(0100) = $4_H$	$B_i = \bar{X}_i$	Bitweise Invertierung X
(0101) = $5_H$	$B_i = \bar{Y}_i$	Bitweise Invertierung Y
(1000) = $8_H$	$B_i = X_i \vee Y_i$	OR
(1001) = $9_H$	$B_i = X_i \wedge Y_i$	AND

Tabelle 4: Wirkung des Steuersignals (K) auf  $B_i$  in Abhängigkeit von  $X_i$  und  $Y_i$  ( $i = 0, \dots, 7$ ).

Hinweis: AR=0 sperrt das 8-Bit AND-Gatter und AR=1 schaltet **X** nach **A** durch!





### 3.2 Speicher

(8 Punkte)

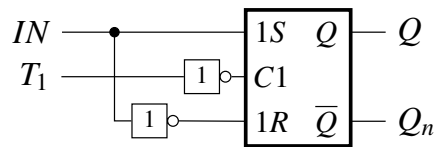
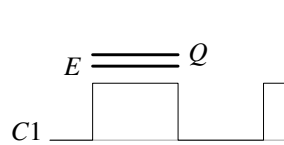


Abbildung 4: Schaltung mit einem RS-Flipflop

Hinweise zu RS-FF und MS-FF:

$1S^k$	$1R^k$	$Q^{k+1}$
0	0	$Q^k$
0	1	0
1	0	1
1	1	vermeiden



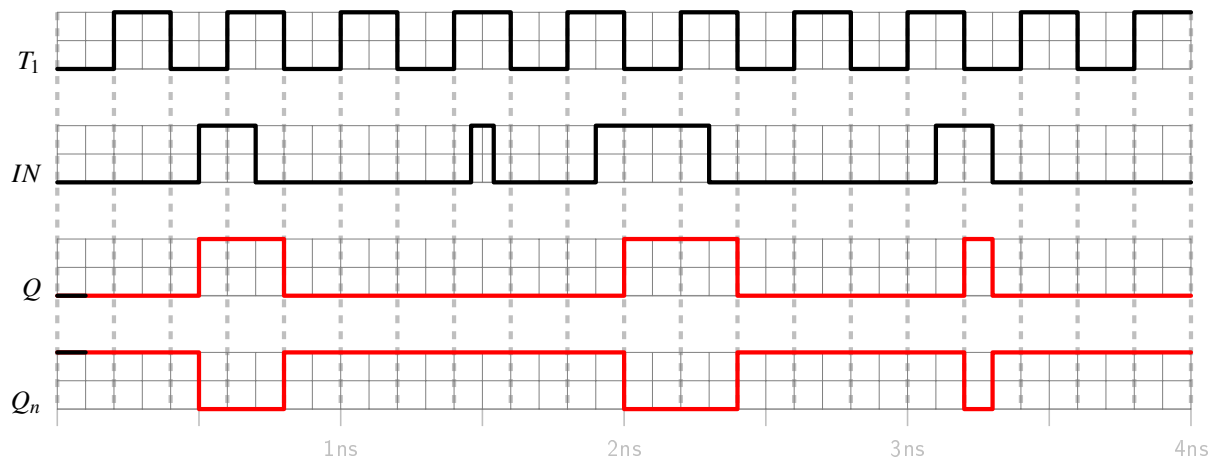
Änderung am Eingang (E) wird nur während der **aktiver Taktphase** am Ausgang wirksam

Tabelle 7: Vereinfachte Funktionstabelle RS-FF

Abbildung 5: Funktionsweise bei Änderung am Eingang eines taktzustandgesteuerten Flipflops

Vervollständigen Sie im nachfolgenden Impulsdiagramm die Signale **Q** und **Q<sub>n</sub>** der Schaltung aus Abbildung 4.

Die Gatterlaufzeiten sind zu vernachlässigen ( $t_{P,clk \rightarrow Q,LH} = t_{P,clk \rightarrow \bar{Q},LH} = t_{P,clk \rightarrow Q,HL} = t_{P,clk \rightarrow \bar{Q},HL} = 0ns$ )



Wie nennt man den Typ (FF-Typ) der Schaltung aus Abbildung 4?

(2 Punkte)

Delay-Flip-Flop

Prüfungsfach: <b>Informationstechnik</b>	Wintersemester 2015/2016	<b>Hochschule Esslingen</b> University of Applied Sciences
Name, Vorname:	Mat.-Nr.:	

## 4 Offene Fragen

### 4.1 Rechnertypen

(6 Punkte)

- (1) Erklären Sie kurz was bei einem Mainframe und einem Thin Client gleich ist.
- (2) Erklären Sie kurz worin sich ein Mainframe und ein Thin Client unterscheiden.

- (1) Es können mehrere Betriebssysteme gleichzeitig betrieben werden. Dem Benutzer wird ein eigener virtueller Rechner vorgegaukelt. Dient nur zur Anzeige der Verarbeitungsergebnisse. Verarbeitung und Speicherung finden auf dem Server statt. Beide Rechnertypen haben Vorteile bei der Wartung, der Datensicherheit und der Verarbeitung großer Datenmengen. (4 Punkte)
- (2) Mainframe ist deutlich teurer. In der Regel ist auch die Anzahl der Terminals bei einem Mainframe deutlich größer. (2 Punkte)

### 4.2 Rechnerarchitekturen

(6 Punkte)

- (1) Worin unterscheiden sich eine „von-Neumann-Architektur“ von einer „Harvard-Architektur“?
- (2) Was war die Motivation auf Basis eines CISC-Prozessor einen RISC-Prozessor zu entwickeln?

- (1) von-Neumann-Architektur: besser **Speicherausnutzung** bei Anwendungen mit unterschiedlich großer Menge an Programmcode und Daten. Nur ein **Bussystem**, verlangsamt sich die Ausführung von Programmen mit Datenzugriffen im Gegensatz zu der Harvard-Architektur. (3 Punkte)
- (2) manche Befehle wurden nur sehr selten aufgerufen und Speicher wurden günstiger und schneller. (3 Punkte)

Prüfungsfach: <b>Informationstechnik</b>	Wintersemester 2015/2016	<b>Hochschule Esslingen</b> University of Applied Sciences
Name, Vorname:	Mat.-Nr.:	

### 4.3 Betriebssystem

(5 Punkte)

Welche Aufgaben hat ein Betriebssystem?

- Die zentrale Aufgabe des Betriebssystems ist die **Verwaltung der Ressourcen**, insbesondere wenn **viele Benutzer** und deren **Programme** gleichzeitig auf diese Ressourcen zugreifen. (3 Punkte)
- Konkret gehören zu den Aufgaben des Betriebssystems die Verwaltung der: Prozesse, Speicher, Dateien (2 Punkte)

### 4.4 Vorgehensmodelle/Prozessentwicklung

(5 Punkte)

- (1) Welche Teilschritte beinhaltet ein **Spiralmodell** zusätzlich zu einem V-Modell?
- (2) Worin unterscheidet sich ein Lastenheft von einem Pflichtenheft?

- (1) Risikoanalyse und Entwicklung mehrerer Prototypen (2 Punkte)
- (2) Lastenheft beinhaltet die Anforderungen aus Kundensicht.  
Pflichtenheft beinhaltet die Realisierung der Produkts aus Herstellersicht.  
Pflichtenheft ist meist viel ausführlicher als das Lastenheft. (3 Punkte)