

KLAUSUR

Informationstechnik

Wintersemester 2016/2017

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

1.2 Kombinatorische Schaltung

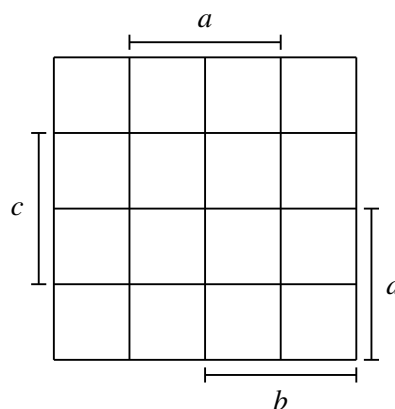
(12 Punkte)

Gegeben ist eine kombinatorische Schaltung. Diese wird durch eine Funktionstabelle (siehe Tabelle 1) beschrieben.

	d	c	b	a	Y	\bar{Y}
0	0	0	0	0	1	
1	0	0	0	1	1	
2	0	0	1	0	1	
3	0	0	1	1	1	
4	0	1	0	0	0	
5	0	1	0	1	1	
6	0	1	1	0	0	
7	0	1	1	1	1	
8	1	0	0	0	1	
9	1	0	0	1	1	
10	1	0	1	0	X	
11	1	0	1	1	1	
12	1	1	0	0	0	
13	1	1	0	1	0	
14	1	1	1	0	0	
15	1	1	1	1	0	

Tabelle 1: Funktionstabelle

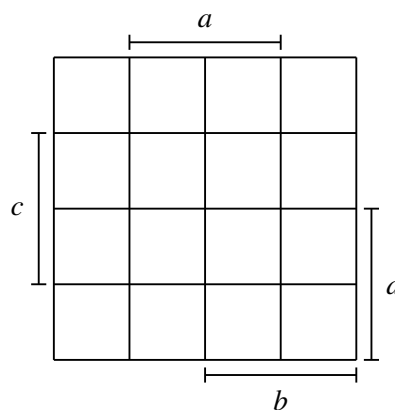
1. Bestimmen Sie die DMF des Signals **Y** mit Hilfe des KV-Diagramms und die Funktionslänge I_{DMF} .



$Y_{DMF} =$

$I_{DMF} =$

2. Bestimmen Sie die KMF des Signals **Y** mit Hilfe des KV-Diagramms und die Funktionslänge I_{KMF} .



$Y_{KMF} =$

$I_{KMF} =$

Prüfungsfach: Informationstechnik	Wintersemester 2016/2017	Hochschule Esslingen University of Applied Sciences
Name, Vorname:	Mat.-Nr.:	

2 Zahlendarstellung und Codierung

2.1 Festkommadarstellung

(15 Punkte)

Gegeben sind die Zahlen $Z_1 = (-127)_{10}$ und $Z_2 = (64)_{10}$.

Wandeln Sie die beiden Zahlen Z_1 und Z_2 jeweils in eine Zahl zur **Basis 16** um, indem Sie eine

1. **Dualcodierung** (Betragzahl) verwenden:

2. **2er-Komplement-Codierung** verwenden:

3. **Offset-Code-Codierung** verwenden:

4. **Vorzeichen-Betrags-Codierung** verwenden:

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

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

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

Gegeben sind die drei Gleitkommazahlen $Z_1 = (\mathbf{0000\ 0000})_{16}$, $Z_2 = (\mathbf{FFFF\ FFFF})_{16}$ und $Z_3 = (\mathbf{3F00\ 0000})_{16}$ in einfacher Genauigkeit nach IEEE 754. Welchen „Zahlen“ entsprechen diese Gleitkommazahlen im Dezimalsystem?

Prüfungsfach: Informationstechnik	Wintersemester 2016/2017	Hochschule Esslingen University of Applied Sciences
Name, Vorname:	Mat.-Nr.:	

2.3 Blockcodes

(9 Punkte)

Das Nachrichten-Codewörter $\mathbf{X}_1 = [111]$, $\mathbf{X}_2 = [001]$ und $\mathbf{X}_3 = [010]$ sollen zu einem Empfänger übertragen werden. Um Datenmanipulation zu verhindern werden mit Hilfe der Generatormatrix

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

weitere Kontrollbits hinzugefügt.

Wie lautet das mit Hilfe der Generatormatrix \mathbf{G} gezeugte Codewörter \mathbf{Y}_1 , \mathbf{Y}_2 und \mathbf{Y}_3 ?

Gegeben ist der Code \mathbf{Z} bestehend aus den drei folgenden Codewörtern:

$\mathbf{Z}_1 = 10101011$ $\mathbf{Z}_2 = 00010111$ $\mathbf{Z}_3 = 11110000$

Welche Hammingdistanz besitzt der Code \mathbf{Z} ?

Wie viele Einzelbitfehler können bei dem Code \mathbf{Z} sicher erkannt und korrigiert werden?

3 Hardware

Die in Abbildung 2 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).

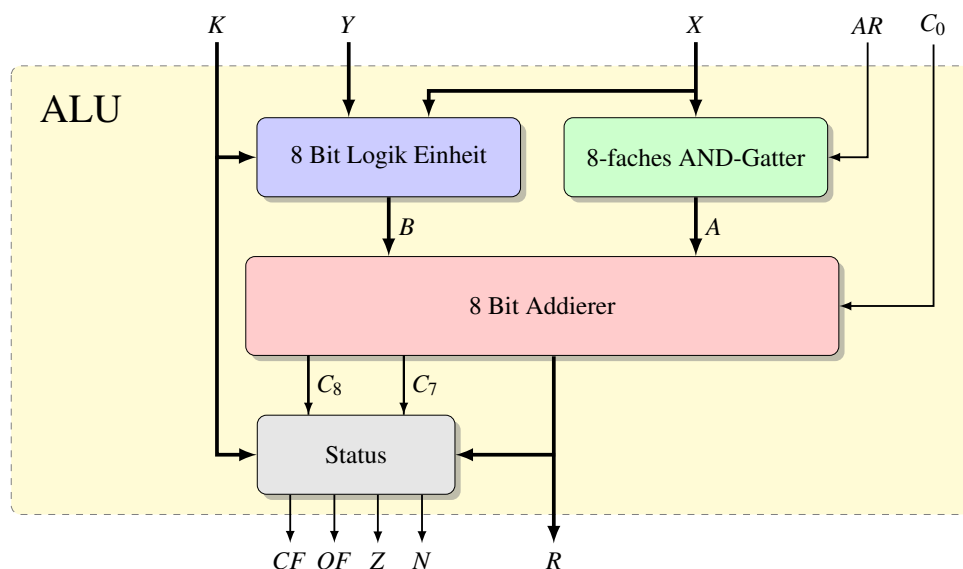


Abbildung 2: Aufbau 8-Bit ALU

Die Signale haben folgende Bitbreite:

Signalname	A	B	X	Y	R	K	AR	C ₀	C ₇	C ₈	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!

3.3 Speicher

(8 Punkte)

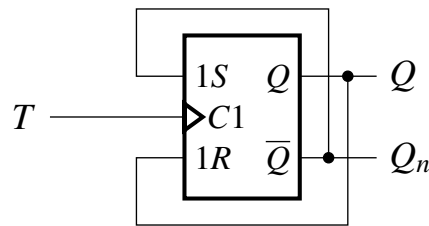


Abbildung 3: Schaltung mit einem RS-Flipflop

Hinweise zu Flip-Flops:

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

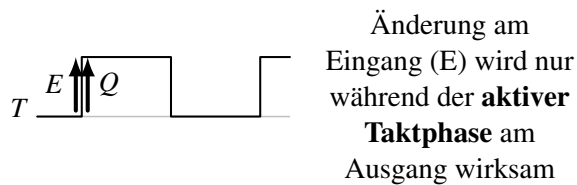
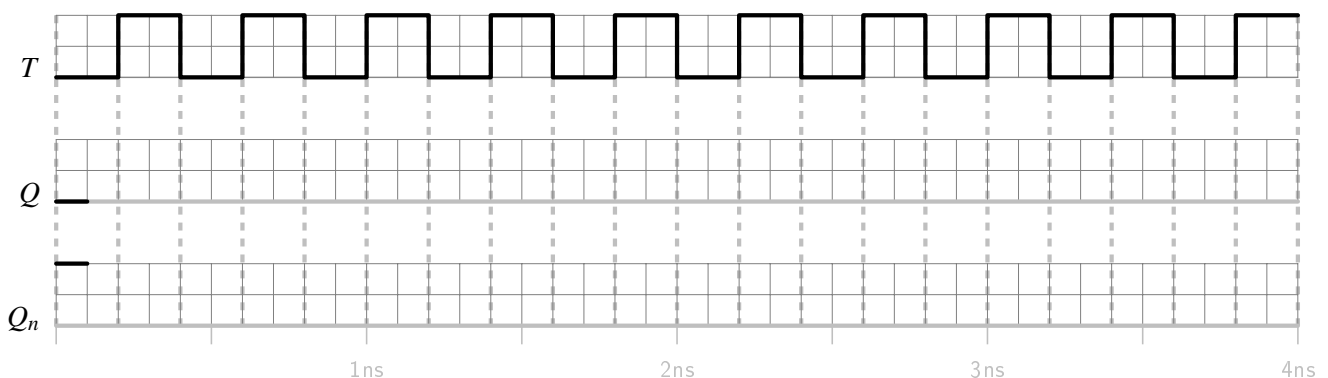


Tabelle 7: Vereinfachte Funktionstabelle RS-FF

Abbildung 4: Funktionsweise bei Änderung am Eingang eines pos. taktflankengesteuerten Flipflops

Vervollständigen Sie im nachfolgenden Impulsdigramm die Signale **Q** und **Q_n** der Schaltung aus Abbildung 3.

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



Prüfungsfach: Informationstechnik	Wintersemester 2016/2017	Hochschule Esslingen University of Applied Sciences
Name, Vorname:	Mat.-Nr.:	

4 Offene Fragen

4.1 Rechnertypen

(6 Punkte)

Erklären Sie den Unterschied zwischen einem *PC* und einem „*Embedded System*“.

4.2 Rechnerarchitekturen

(4 Punkte)

- (1) Was ist die Aufgabe des Steuerwerks?
- (2) Was ist die Aufgabe des Rechenwerks?

Prüfungsfach: Informationstechnik	Wintersemester 2016/2017	Hochschule Esslingen University of Applied Sciences
Name, Vorname:	Mat.-Nr.:	

4.3 Compiler

(10 Punkte)

Programme durchlaufen mehrere Schritte in einer Werkzeugkette (Toolchain), bevor sie von einer CPU ausgeführt werden können. Beschreiben Sie diese Schritte, gerne auch per Skizze.