

Universität Koblenz–Landau
Institut für Naturwissenschaften
Abteilung Physik

Name: **Musterlösung**
Vorname:
Matr. Nr.:
Studiengang:

Klausur

zur Vorlesung
“Technische Informatik C”

Donnerstag 20.12.2007

Lösen Sie die Aufgaben 1 - 6!
Verwenden Sie keinen Bleistift!
Es sind keine Hilfsmittel zugelassen.
Schalten Sie Ihr Handy aus!

GUTEN ERFOLG !!!

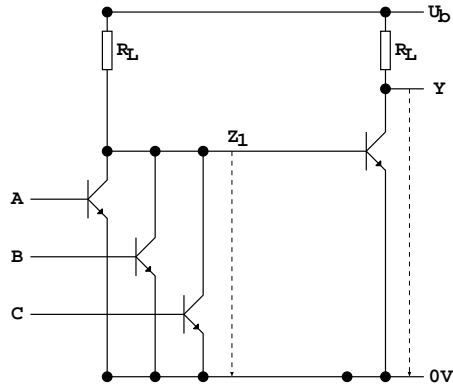
Aufgabe	1	2	3	4	5	6
max. Punkte	8	9	10	13	12	8
err. Punkte						

Summe Pkte.:

Note.:

Aufgabe 1: Diskrete Logikschaltungen

Gegeben ist die folgende Schaltung.



Hinweis: Die Transistoren schalten, wenn H an der Basis anliegt. Der Widerstand eines leitenden Transistors ist dann gegenüber dem Lastwiderstand R_L sehr klein: $R_{leit} \ll R_L$. Ein sperrender Transistor (Basis = L) hat hingegen einen sehr viel höheren Widerstand als der Lastwiderstand $R_{sperr} \gg R_L$.

- a) Analysieren Sie das elektrische Schaltverhalten der Schaltung. Die Eingangspegel der drei Eingänge A, B und C können die beiden Werte $H \simeq U_b$ und $L \simeq 0$ annehmen. Bestimmen Sie zunächst die Pegel (H oder L), die sie an den Punkten Z_1 und Y gegen Masse abgreifen können.

A	B	C	Z_1	Y
L	L	L	H	L
L	L	H	L	H
L	H	L	L	H
L	H	H	L	H
H	L	L	L	H
H	L	H	L	H
H	H	L	L	H
H	H	H	L	H

() / 4

- b) Untersuchen Sie das logische Verhalten der Schaltung, zunächst mit positiver Logik ($H = 1, L = 0$) und mit negativer Logik ($H = 0, L = 1$). Geben Sie jeweils die ermittelten Funktionen (auch Zwischenfunktionen) an.

positive Logik ($H = 1, L = 0$):

A	B	C	Z_1	Y
0	0	0	1	0
0	0	1	0	1
0	1	0	0	1
0	1	1	0	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	0	1
Funktion			$A \vee B \vee C$	$A \vee B \vee C$

negative Logik ($H = 0, L = 1$):

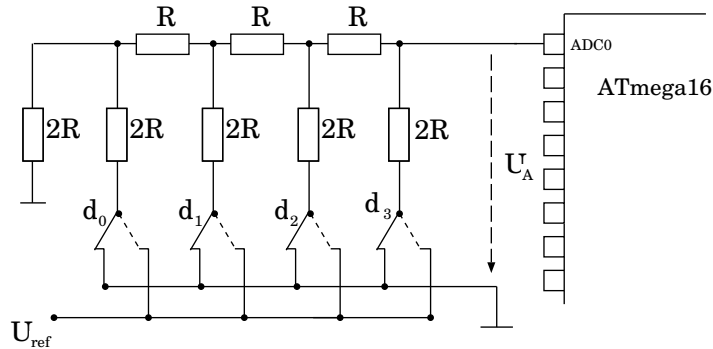
A	B	C	Z_1	Y
1	1	1	0	1
1	1	0	1	0
1	0	1	1	0
1	0	0	1	0
0	1	1	1	0
0	1	0	1	0
0	0	1	1	0
0	0	0	1	0
Funktion			$A \wedge B \wedge C$	$A \wedge B \wedge C$

() / 4

Aufgabe 2: DA-Wandler

Digital-Analog Wandler können in Verbindung mit einem Analogeingang eines Mikrocontrollers dazu verwendet werden, mehrere digitale Schalter gleichzeitig auszuwerten. Der Vorteil dieser Vorgehensweise liegt darin, dass nur ein einziger Anschlusspin am Mikrocontroller belegt wird.

Gegeben sei dazu folgende Schaltung:



Je nach Stellung der Schalter ($d_3 \cdots d_0$) wird eine definierte Ausgangsspannung U_A erzeugt, die vom AD-Wandler des Mikrocontrollers in eine 10 Bit Zahl umgewandelt wird. Aus diesem gemessenen Wert können die Schalterstellungen rekonstruiert werden.

- a) Wieviele verschiedene Ausgangsspannungen (U_A) können mit dem obigen DA-Wandler erzeugt werden?

Es können 16 verschiedene Spannungen erzeugt werden. () / 1

- c) Welche Ergebnisse finden sich im ADC-Ergebnisregister, wenn folgende Schalterstellungen eingestellt sind ($d_n = 0$ Schalter n auf Masse, $d_n = 1$ Schalter n auf U_{ref})

d_3	d_2	d_1	d_0	Messergebnis im 10 Bit AD-Register
0	0	0	0	0
0	0	0	1	64
0	0	1	0	128
0	1	0	0	256

() / 4

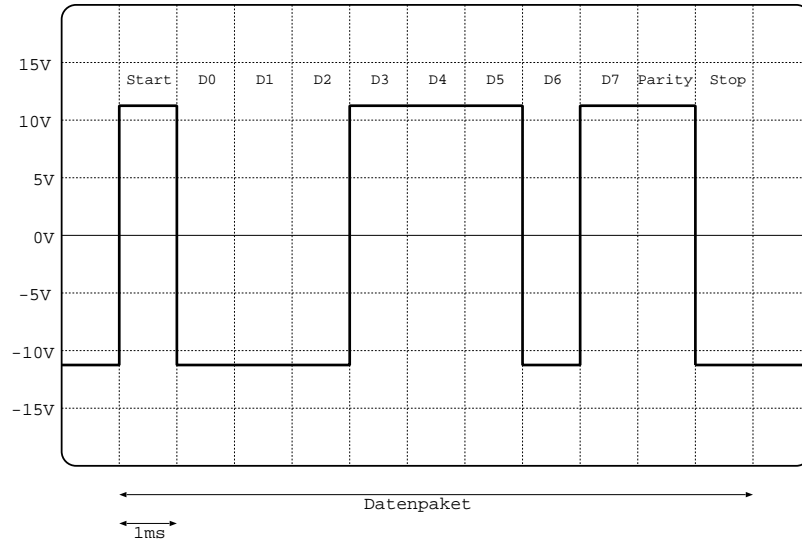
- c) Tragen Sie für folgende Messergebnisse die entsprechenden Schalterstellungen in die Tabelle ein ($d_n = 0$ Schalter n auf Masse, $d_n = 1$ Schalter n auf U_{ref})

Messergebnis im 10 Bit AD-Register	d_3	d_2	d_1	d_0
640	1	0	1	0
320	0	1	0	1
960	1	1	1	1
512	1	0	0	0

() / 4

Aufgabe 3: serielle Schnittstelle

An der RS232 Schnittstelle eines PCs wurde mit dem Oszilloskop folgendes Datenpaket mitgeschnitten:



Hinweis: Das Startbit ist logisch 0, ein Stopbit logisch 1. Weiterhin ist bekannt, dass genau 8 Datenbits übertragen wurden.

Analysieren Sie den abgebildeten Frame:

- a) Notieren Sie im obigen Bild die einzelnen Datenbit ($D_0 \dots D_7$) und die Steuerbits (Start, Stop, Parity) soweit vorhanden. () / 5

- b) Wie wurden die folgenden Parameter zwischen den Kommunikationspartnern vereinbart?

- Anzahl Stopbits = 1
- Paritätsbit (No, Even, Odd) = Even

- c) Welche Hexadezimalzahl wird im obigen Packet verschickt? () / 2

$0b01000111 = 0x47$ () / 2

- b) Wie hoch ist die Datenübertragungsrate D (einschliesslich der Steuerbits) in Bit/s?

$$D = 1\text{Bit}/1\text{ms} = 1000\text{Bit/s}$$

() / 1

Aufgabe 5: Programmierung eines Leitwerks

Gegeben sei die letzte Ausbaustufe unseres 4Bit-Miniprozessor (s. Anhang):

Analysieren Sie die folgenden Auszüge aus dem Leitwerk

- a) Erklären Sie jede Zeile des Leitwerkprogramms und folgern Sie daraus auf den realisierten Assemblerbefehl.

	AC	BC	OUT	DW1	DW0	AS2	AS1	AS0	StC	JZ	JNZ	ARC	ABW	PCC	RMW	IRC
1)	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
2)	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1
3)	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0	0
4)	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0
5)	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0
6)	0	1	0	1	1	0	1	1	0	0	0	0	1	0	0	0
7)	0	0	0	0	1	0	1	1	1	0	0	0	0	0	0	0
8)	1	0	0	0	1	0	1	1	0	0	0	0	0	1	0	0

- 1) Ram legt den aktuellen Befehl auf den Datenbus
 - 2) IR uebernimmt den Befehl vom Datenbus
 - 3) PC wird incrementiert und das Ram legt ein Datum (Adresse) auf den Datenbus
 - 4) Adressregister uebernimmt diese Adresse
 - 5) Adressregister legt die Adresse auf den Adressbus, und das RAM legt das adressierte Datum auf Datenbus
 - 6) RegisterB uebernimmt den Operanden, gleichzeitig wird die ALU auf SUB eingestellt
 - 7) ALU legt das Ergebniss auf den Datenbus, Statusregister sichert den Status
 - 8) Accu uebernimmt das Ergebniss ($A = A - B$). Gleichzeitig wird der PC incrementiert und zeigt auf den nächsten Befehl
- SUB

() / 8

- b) dito

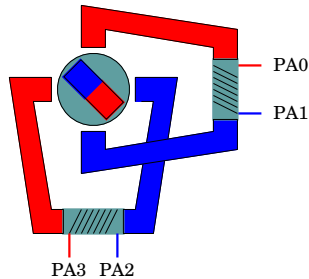
	AC	BC	OUT	DW1	DW0	AS2	AS1	AS0	StC	JZ	JNZ	ARC	ABW	PCC	RMW	IRC
1)	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
2)	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1
3)	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0	0
4)	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0
5)	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0

- 1) Ram legt den aktuellen Befehl auf den Datenbus
- 2) IR uebernimmt den Befehl vom Datenbus
- 3) PC wird incrementiert und das Ram legt ein Datum (Adresse) auf den Datenbus
- 4) Adressregister uebernimmt diese Adresse
- 5) Ist das Statusregister nicht NULL, so uebernimmt der PC den Inhalt des Adressregisters → JNZ

() / 4

Aufgabe 6: C Programmierung

Entwerfen Sie ein C-Programm zur Ansteuerung eines Schrittmotors. Der Schrittmotor soll im **Halbschritt** betrieben werden.



- a) Geben Sie zunächst die logische Steuerfolge (in 1/0 Notation) für den **Halbschrittbetrieb** in beliebiger Drehrichtung an:

Schritt Nr.	PA3	PA2	PA1	PA0
0	0	1	0	1
1	0	1	0	0
2	0	1	1	0
3	0	0	1	0
4	1	0	1	0
5	1	0	0	0
6	1	0	0	1
7	0	0	0	1

() / 4

- b) Schreiben Sie ein möglichst kurzes C-Programm zur Ansteuerung. Verwenden Sie PortA als Ausgabe. Eine Konfiguration des Ports ist nicht gefordert, ebensowenig wie Delays zwischen den Ausgaben.

() / 4