



Interessante Altklausur Aufgaben

Mikroprozessoren (Hochschule Darmstadt)



messages.pdf_cover_qr_code_label

h_da Günther Fröhlich	Mikroprozessorsysteme SS 2016 Klausur am 7.7.2016 10:15-11:45 D14/104-204	Name: _____ Matrikel-Nr.: _____
------------------------------------	--	--

Aufgabe 4: Cyclic Redundancy Check

8 Punkte

Die Korrektheit einer Datenübertragung soll mit dem Cyclic Redundancy Check sichergestellt werden. Hierzu wird folgendes Generatorpolynom verwendet:

$$x^5 + x^2 + 1$$

Die zu übertragende Nachricht stellt die Bitfolge 1 1 1 1 1 1 1 0 0 0 dar.

Bestimmen Sie die CRC-Prüfbits (6 Punkte).

$$\begin{array}{r}
 11111110000000 \\
 1001011 \\
 \hline
 110101 \\
 100101 \\
 \hline
 1000000 \\
 100101 \\
 \hline
 101000 \\
 100101 \\
 \hline
 110100 \\
 100101 \\
 \hline
 100010 \\
 100101 \\
 \hline
 1110
 \end{array}$$

Prüfbit: 01110

Wie lautet der übertragende Code bestehend aus Datenbits und Prüfbits in hexadezimaler Form bei rechtsbündiger Anordnung?

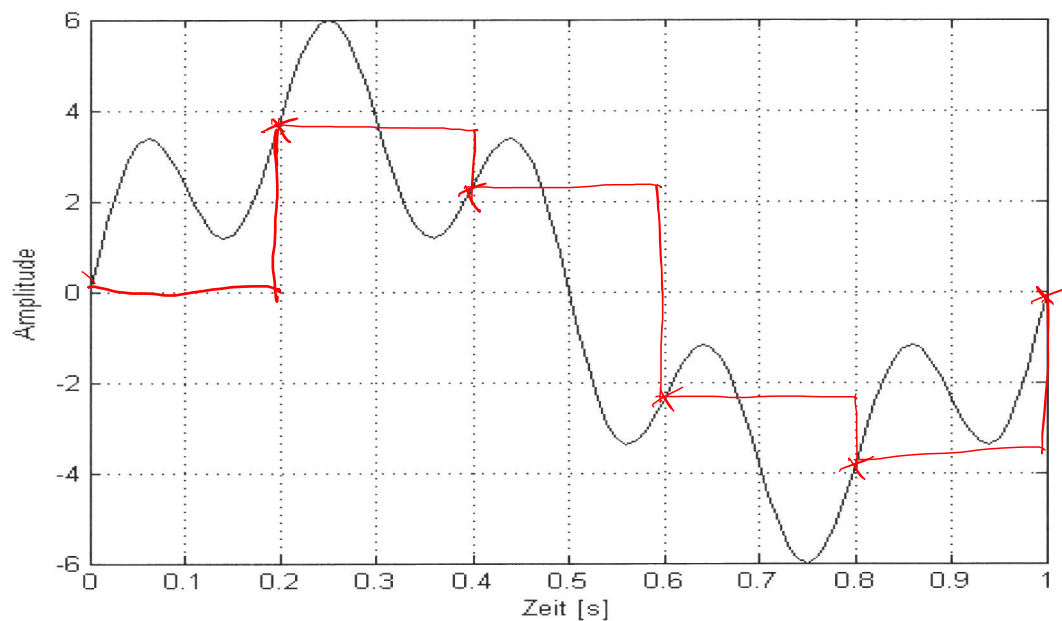
1 1 1 | 1 1 1 1 | 0 0 0 | 1 1 1 0
7 F 0 E

h_da Günther Fröhlich	Mikroprozessorsysteme SS 2016 Klausur am 7.7.2016 10:15-11:45 D14/104-204	Name: _____ Matrikel-Nr.: _____
------------------------------------	--	--

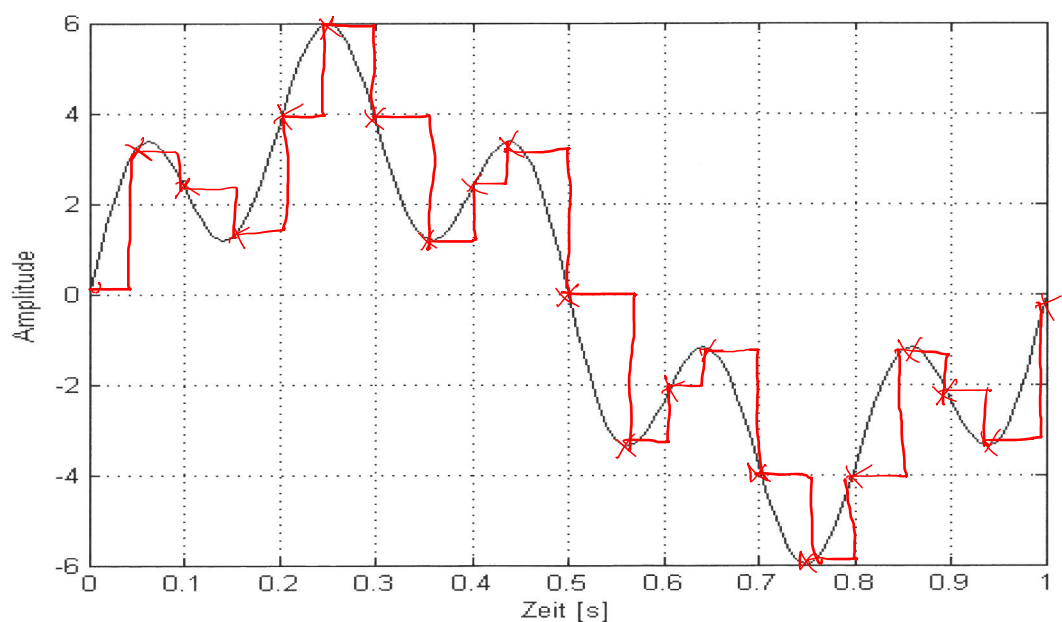
Aufgabe 6: Signalverarbeitung

4 Punkte

Tasten Sie das gegebene Signal mit 5 Hz ab und skizzieren Sie im Graph das abgetastete und gehaltene Signal (2 Punkte)



c) Tasten Sie das gegebene Signal mit 20 Hz ab und skizzieren Sie im Graph das abgetastete und gehaltene Signal (2 Punkte)



h_da Günther Fröhlich	Mikroprozessorsysteme SS 2016 Klausur am 7.7.2016 10:15-11:45 D14/104-204	Name: _____ Matrikel-Nr.: _____
------------------------------------	--	--

Aufgabe 7: Interruptverarbeitung

8 Punkte

Beschreiben sie die folgenden 3 Methoden der Ereignisbehandlung und grenzen Sie sie gegeneinander ab

A) Busy Waiting

Wartet SD lange auf das Eintreten einer Bedingung im Programmfluss stehen.

B) Polling

Programm fragt alle Schnittstellen in einer Schleife ab und reagiert auf auftretende Ereignisse. Problem: Die Wahrscheinlichkeit Ereignis zu verpassen ist hoch.

C) Interrupt Handling

Laufendes Programm wird unterbrochen und eine Reaktionsroutine wird gestartet.

D) Welche Hardware Komponenten des Arm unterstützen speziell die Interruptbehandlung von Ereignissen ?

- ① Alle möglichen Interruptquellen.
- ② Alle außer CPU und AIC
- ③ ? ? ?

h_da Günther Fröhlich	Mikroprozessorsysteme SS 2016 Klausur am 7.7.2016 10:15-11:45 D14/104-204	Name: _____ Matrikel-Nr.: _____
------------------------------------	--	--

Aufgabe 8: serielle Schnittstelle - DMA Transfer 6 Punkte

Welche Parameter werden bei der Initialisierung einer seriellen Schnittstelle definiert?

Welche Aufgabe hat der DMA Kanal der seriellen Schnittstelle?

--

Warum kann ein DMA Kanal diese Aufgabe besser ausführen als eine CPU?

h_da Günther Fröhlich	Mikroprozessorsysteme WS 2016 Klausur am 27.1.2017 14:15-15:45 D14/104-204 Unterschrift:	Name: _____ Matrikel-Nr.: _____
------------------------------------	--	--

r f Kreuzen Sie bei Aussagen mit Kreisen an, ob die Aussage richtig oder falsch ist

oo

(1)

4 Punkte

Die Programmiersprache C ist für hardwarenahe Programmierung besonders geeignet weil:

- a)
- b)
- c)
- d)

(2)

16 Punkte

- oo *Memory Mapped IO* ist ein bewährtes Verfahren, um Ein-/Ausgaben zu beschleunigen
- oo *Memory Mapped IO* entlastet die CPU bei Ein-/Ausgaben
- oo *Memory Mapped IO* ist ein Verfahren zur Kommunikation einer CPU mit Peripheriebausteinen
- oo Bei *Memory Mapped IO* werden die Steuer- und Status- Register der Peripherie in den Hauptspeicher kopiert
- oo Bei *Memory Mapped IO* kann die Programmierung der Peripherie ohne privilegierte Befehle erfolgen
- oo *Port Mapped IO* schützt die Peripherie vor Zugriff im *user mode*, weil auf ports nur mit dem privilegierten *in/out* Befehl/zugegriffen werden kann
- oo Bei *Memory Mapped IO* kann nicht der gesamte Adressraum vom Hauptspeicher belegt werden
- oo *Memory Mapped IO* ermöglicht es, Hardwareregister wie Variablen zu behandeln. Das kann zu Dateninkonsistenzen führen, die mit der Deklaration *Volatile* verhindert werden müssen
- oo Sicherheitshalber sollten im embedded Bereich alle Variablen als *Volatile* deklariert werden, weil man einer Variablen nicht ansieht, ob sie ein Hardwareregister repräsentiert
- oo Mikrocontroller sind Mikroprozessoren auf einem Chip
- oo Mikrocontroller werden zu Steuerungsaufgaben eingesetzt, die für Mikroprozessoren zu aufwendig sind
- oo Das Projekt *Selbstfahrende Autos* verlangt von Mikrocontrollern neben Hochleistungs-Mehrkernprozessoren und viel mehr Speicher als bisher auch leistungsfähigere Peripheriebausteine
- oo Powermanagement, das Abschalten von gerade nicht benötigter Peripherie ist wichtig bei batteriebetriebenen Systemen, der Takt der CPU des At91M63200 kann reduziert, aber nicht abgeschaltet werden, da die Steuerung des PMC die CPU erfordert.
- oo Ein-Ausgaben (Taster und LEDs) werden vom PIO-Controller gesteuert. Direkte Ausgaben auf einzelne Port-Pins sind für andere Bausteine (wie Timer etc) nicht möglich
- oo Zur Ausgabe von Daten besitzt die PIO ein SetOutputDataRegister (SODR). Dies wird bei gewünschter Ausgabe von der CPU ebenso wie von den Peripheriebausteinen beschrieben
- oo Die PIO ermöglicht, dass die CPU und mehrere Peripheriebausteine auf den ihnen zugewiesenen Pins gleichzeitig Ein-/Ausgaben durchführen

h_da Günther Fröhlich	Mikroprozessorsysteme WS 2016 Klausur am 27.1.2017 14:15-15:45 D14/104-204 Unterschrift:	Name: _____ Matrikel-Nr.: _____
------------------------------------	--	--

(8)

5 Punkte

Was bestimmt die Zeit eines Timers bis zum Überlauf ?

- ☐ Die Taktfrequenz der CPU
- ☐ Die Taktfrequenz des Timers
- ☐ Die Anzahl Zählregister im Timer
- ☐ Die Einstellung des Prescalers
- ☐ Ob ein Interrupt zwischenzeitlich eintritt oder nicht
- ☐ Ob er auf raising oder falling edge triggert
- ☐ Die Breite der Zähler Register in Bit

Wie kann man mit einem Timer die Leuchstärke einer LED modifizieren?

.....
.....

(9)

3 Punkte

Zusätzlich zum 32-Bit ARM-Befehlssatzes wurde auch ein auf 16-Bit reduzierter THUMB-Befehlssatz auf den Arm Prozessoren implementiert

A) Welcher Vorteil ergibt sich durch diese Einschränkung des Arm Befehlssatzes?

B) Welcher Nachteil ergibt sich durch diese Einschränkung des Arm Befehlssatzes?

C) Wodurch kompensiert der Thumb 2 Befehlssatz die Nachteile?

h_da Günther Fröhlich	Mikroprozessorsysteme WS 2016 Klausur am 27.1.2017 14:15-15:45 D14/104-204 Unterschrift:	Name: _____ Matrikel-Nr.: _____
------------------------------------	--	--

(10)

8 Punkte

- Busy Waiting ist angemessen, wenn nur dieses eine Ereignis erwartet wird, bis zum Eintreffen des Ereignisses nichts getan werden kann und dieses so schnell wie möglich verarbeitet werden soll
- Busy Waiting ist im embedded Bereich ungünstig, weil die CPU mit höchster Last wartet
- Werden unterschiedliche Ereignisse erwartet, führt Busy Waiting leicht zum verpassen von Ereignissen
- Polling ist am einfachsten zu Implementieren
- Polling kann wie Busy Waiting die CPU vollkommen auslasten, auch wenn kein Ereignis eintrifft
- Interruptverarbeitung ist programmieraufwendig, das Programm aber zur Laufzeit Ressourcen schonend
- Interruptverarbeitung benötigt ein asynchrones Programmiermodell
- Interruptverarbeitung erfordert viel mehr Aufwand als ein Funktionsaufruf, der notwendige Kontextswitch kann bei zu vielen Interrupts zu Verklemmungen führen

h_da Günther Fröhlich	Mikroprozessorsysteme WS 2016 Klausur am 27.1.2017 14:15-15:45 D14/104-204 Unterschrift: _____	Name: _____ Matrikel-Nr.: _____
------------------------------------	--	--

(11)

5 Punkte

Welche Parameter werden bei der Initialisierung einer seriellen Schnittstelle definiert?

- ☒ Baudrate ☐ Synchronisations-Mode ☐ Aktiv-Mode
- ☐ Taktfrequenz ☐ Datenrichtung ☐ Bits pro Zeichen
- ☒ Anzahl Parity Bits ☐ DMA Übertragung ☐ Anzahl Stop Bits
- ☒ Anzahl CRC Bits

(12)

6 Punkte

- ☐ Die CPU des Atmel arbeitet asynchron zur USART und synchronisiert sich mit ihr über das Channel Status Register, unabhängig davon, ob die USART synchron- oder asynchron betrieben wird.
- ☐ DMA ermöglicht eine beschleunigte Datenübertragung zwischen RAM und Peripherie
- ☐ DMA ist eine auf schnelle Datenübertragung optimierte Hardware, die im Wesentlichen aus einem Zählerregister (das in jedem Takt um eins dekrementiert wird) und zwei Adressregistern (die in jedem Takt um vier erhöht werden) besteht
- ☐ Die synchrone Datenübertragung zwischen USART und externem Empfänger kann schneller erfolgen als die asynchrone

h_da Dr. Horsch Fachbereich I	Mikroprozessortechnik SS 2009 Klausur vom 13.07.2009	Name: _____ Matrikel-Nr.: _____
--	--	------------------------------------

Aufgabe 5:

6 Punkte

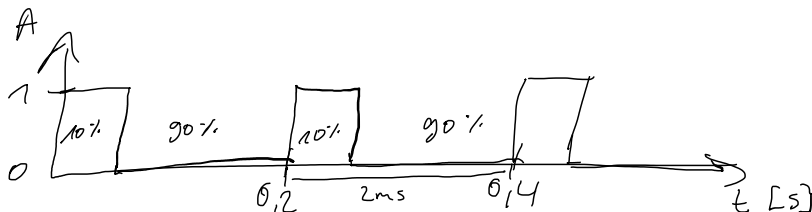
Schauen Sie sich die Initialisierung des Timer 3 bei einem MCKI von **16 MHz** an und beantworten Sie die gestellten Fragen:

```
void Timer3_init( void ) {
.....
// Initialisieren des Modus von Timer 3
timerbase3->TC_CMR =
    TC_ACPC_CLEAR_OUTPUT | // Register C Vergleich löscht TIOA
    TC_ACPA_SET_OUTPUT | // Register A Vergleich setzt TIOA
    TC_WAVE | // Waveform Modus
    TC_CPCTRG | // Register C Vergleich setzt Zähler zurück
    TC_CLKS_MCK32; // TCCLKS: MCKI / 32

// Initialisieren der Compare-Register des Timers:
timerbase3->TC_RA = 100; // Wert im Compare RA Register
timerbase3->TC_RC = 1000; // Wert im Compare RC Register
// Start des Timers:
timerbase3->TC_CCR = TC_CLKEN; // Aktiviere Timertakt
timerbase3->TC_CCR = TC_SWTRG; // Zurücksetzen des Timers
.....
}
```

a) Wie sieht das erzeugte Signal aus und welche Frequenz hat es? Beschreiben Sie den Weg zu Ihrer Lösung und skizzieren Sie das Signal.

$$T_A = \frac{\text{Prescaler} \cdot R_c}{\text{MCKI}} = \frac{2^5 \cdot 10^{-3}}{2^4 \cdot 10^6} = 2 \cdot 10^{-3} = 0,002 \text{ s} = 2 \text{ ms}$$



b) Ändern Sie die Initialisierung so, dass ein symmetrisches Rechtecksignal entsteht mit der Frequenz von 1 kHz.

h_da Dr. Horsch Fachbereich I	Mikroprozessortechnik SS 2009 Klausur vom 13.07.2009	Name: _____ Matrikel-Nr.: _____
--	--	------------------------------------

Aufgabe 6: (Cyclic Redundancy Check)

8 Punkte

Die Korrektheit einer Datenübertragung soll mit dem Cyclic Redundancy Check sichergestellt werden. Hierzu wird folgendes Generatorpolynom verwendet:

$$x^5 + x^3 + 1$$

Die zu übertragende Nachricht stellt die Bitfolge 1 0 0 1 1 1 1 0 0 1 dar. Bestimmen Sie die CRC-Prüfbits.

Was sind die Vorteile des CRC-Verfahrens?

$$\begin{array}{r}
 1001111001 \quad 11001 : 101001 \\
 \underline{101001} \\
 111011 \\
 \underline{101001} \\
 100100 \\
 \underline{101001} \\
 110100 \\
 \underline{110100} \\
 000000
 \end{array}$$

Vorteil:

Sicherung der Datenübertragung
 ↳ Fehlerüberprüfung

Aufgabe 3: Timer Wave-Mode

8 Punkte

Mit einem Timerbaustein des im Praktikum eingesetzten AT91M63200-Prozessor wird ein sich wiederholendes Signal erzeugt.

Das System habe einen **Systemclock von 24MHz**. Es sei die folgende Initialisierung gegeben.

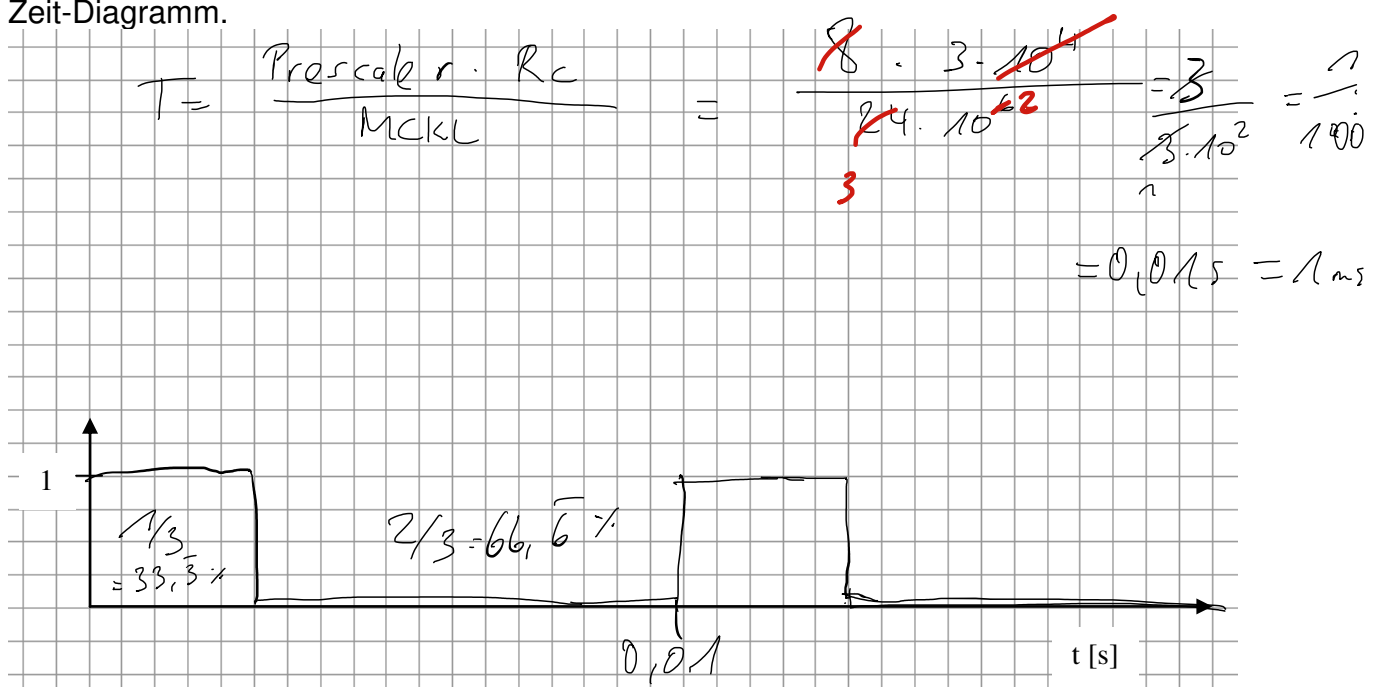
```
// Timer3 initialisieren
void Timer3_init( void )
{
    StructTC* timerbase3 = TCB3_BASE;           // Basisadresse TC Block 1
    timerbase3->TC_CCR = TC_CLKDIS;             // Disable Clock

    // Initialize the mode of the timer 3
    timerbase3->TC_CMR =
    TC_ACPC_CLEAR_OUTPUT | //ACPC : Register C clear TIOA
    TC_ACPA_SET_OUTPUT   | //ACPA : Register A set TIOA
    TC_WAVE               | //WAVE : Waveform mode
    TC_CPCTRG             | //CPCTRG : Register C compare trigger enable
    TC_CLKS_MCK8          | //TCCLKS : MCKI / 8

    // Initialize the counter:
    timerbase3->TC_RA = 10000;                    // Capture RA mit 10000 beschreiben
    timerbase3->TC_RC = 30000;                    // Capture RC mit 30000 beschreiben

    // Start the timer :
    timerbase3->TC_CCR = TC_CLKEN ;              // Enable Clock
    timerbase3->TC_CCR = TC_SWTRG ;              // Timer starten
}
```

Berechnen (Lösungsweg angeben) und zeichnen Sie von dem sich wiederholenden Signal, welches der Timer erzeugt, mindestens eine Periode im beschrifteten Pegel-Zeit-Diagramm.



Aufgabe 6: Cyclic Redundancy Check

8 Punkte

Die Korrektheit einer Datenübertragung soll mit dem Cyclic Redundancy Check sichergestellt werden. Hierzu wird folgendes Generatorpolynom verwendet:

$$x^5 + x^2 + x + 1$$

Die zu übertragende Nachricht stellt die Bitfolge 1 0 0 1 0 1 1 1 0 0 1 dar.
Bestimmen Sie die CRC-Prüfbits.

$$\begin{array}{r}
 1001011100100000 : 100111 \\
 \underline{100111} \\
 101100 \\
 \underline{100111} \\
 101110 \\
 \underline{100111} \\
 100100 \\
 \underline{100111} \\
 1100
 \end{array}$$

Wie lautet der übertragende Code bestehend aus Datenbits und Prüfbits in binärer und hexadezimaler Form?

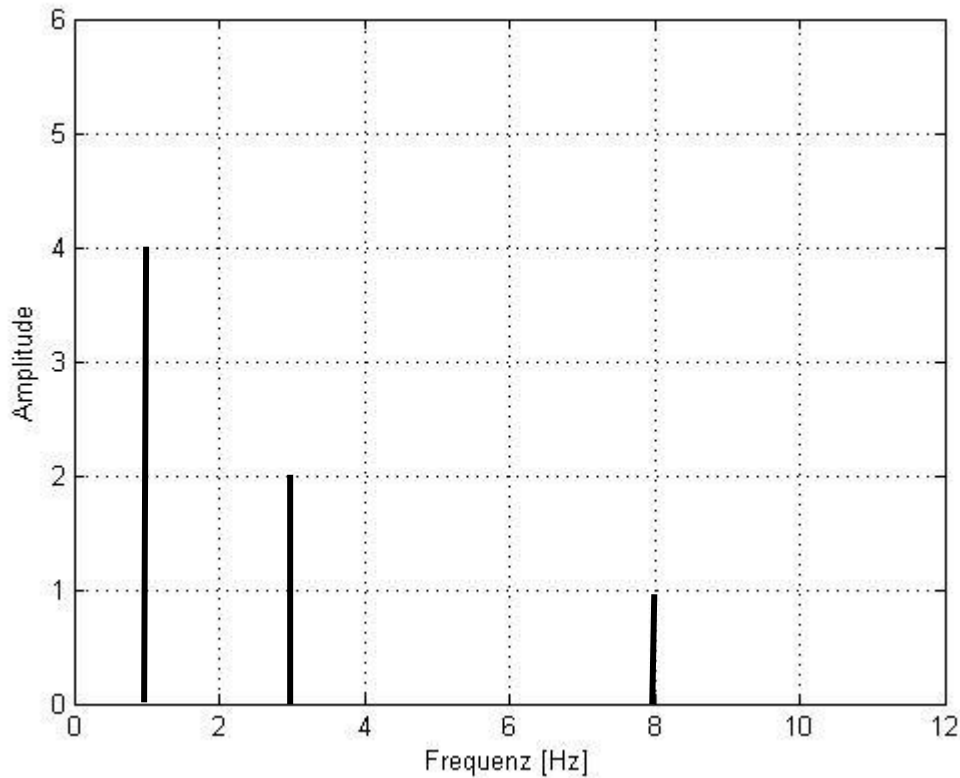
Binär: 100101110010/1100

Hexadezimal: 0X 972C

Aufgabe 7: Signalverarbeitung

9 Punkte

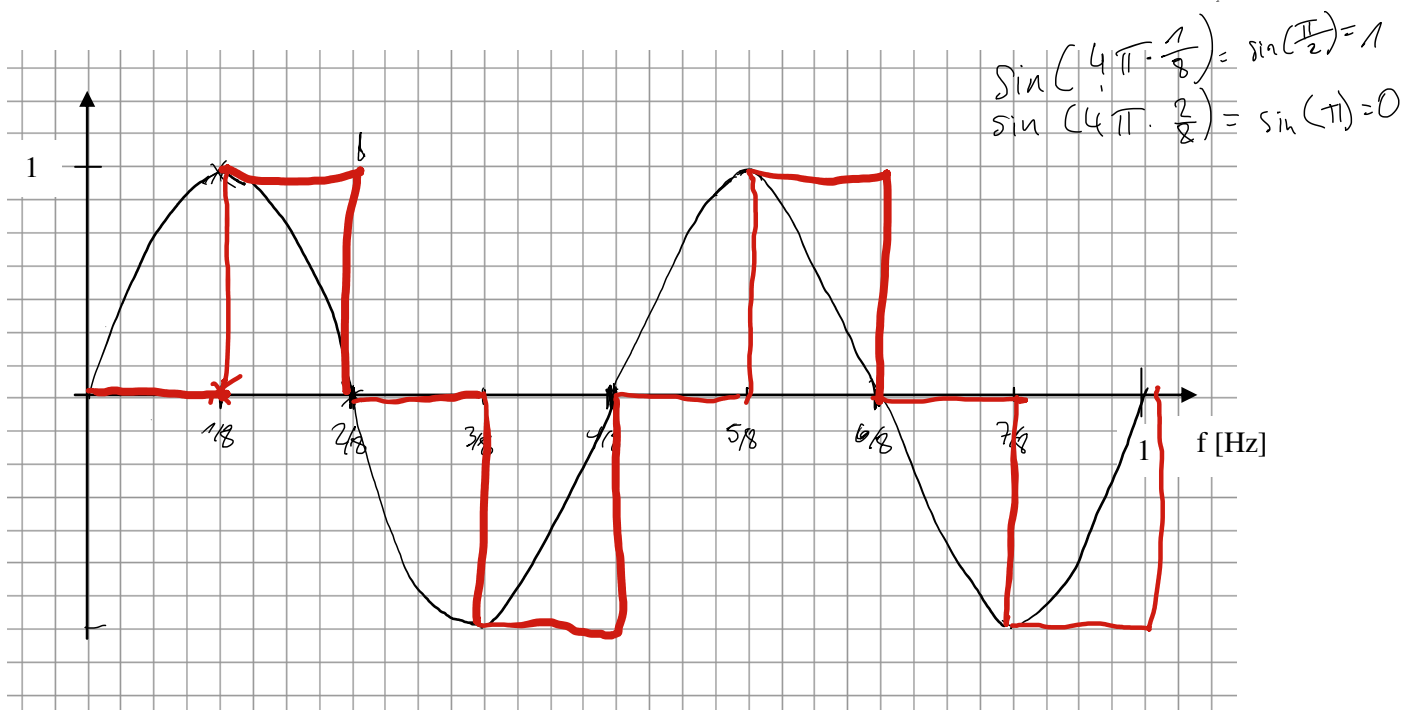
Gegeben sei ein Signal, das aus einer Überlagerung von drei periodischen Signalen besteht und im Frequenzbereich folgendermaßen beschrieben ist:



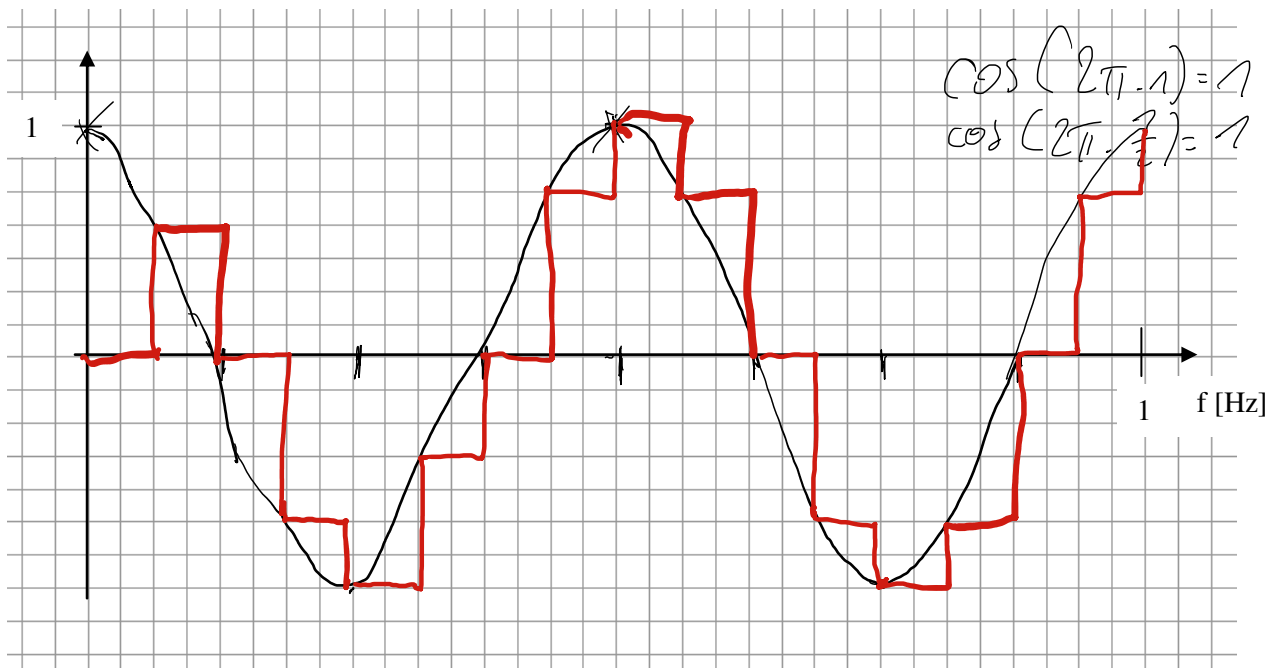
a) Beschreiben Sie das entsprechende Signal im Zeitbereich (3 Punkte)

$$f(t) = 4 \sin(2\pi t) + 2 \sin(6\pi t) + \sin(16\pi t)$$

b) Skizzieren Sie das Signal: $f(t) = \sin(4 \cdot \pi \cdot t)$, tasten es mit 8 Hz ab und skizzieren das entsprechende abgetastete Signal als Treppenfunktion (3 Punkte).



c) Skizzieren Sie das Signal: $f(t) = \cos(2 \cdot \pi \cdot t)$, tasten es mit 16 Hz ab und skizzieren das entsprechende abgetastete Signal als Treppenfunktion (3 Punkte).



Aufgabe 6: Cyclic Redundancy Check

8 Punkte

Die Korrektheit einer Datenübertragung soll mit dem Cyclic Redundancy Check sichergestellt werden. Hierzu wird folgendes Generatorpolynom verwendet:

$$x^5 + x^3 + x$$

Die zu übertragende Nachricht stellt die Bitfolge 1 0 0 1 0 1 1 1 0 0 1 dar.
Bestimmen Sie die CRC-Prüfbits.

$$\begin{array}{r}
 10010111001 : 101010 \\
 101010 \mid 1 \\
 \hline
 1111111 \\
 1010100 \\
 \hline
 101010 \\
 101010 \\
 \hline
 0001
 \end{array}$$

Wie lautet der übertragende Code bestehend aus Datenbits und Prüfbits in binärer und hexadezimaler Form?

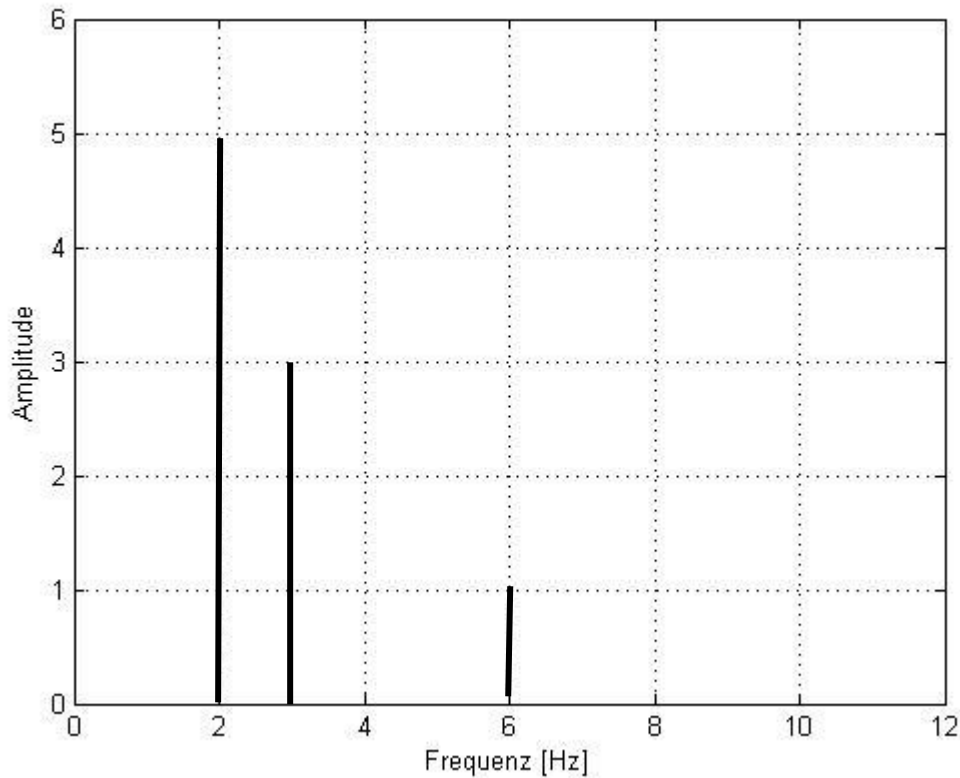
Binär: 100101110010001

Hexadezimal: 0x 9721

Aufgabe 7: Signalverarbeitung

10 Punkte

Gegeben sei ein Signal, das aus einer Überlagerung von drei periodischen Signalen besteht und im Frequenzbereich folgendermaßen beschrieben ist:



a) Beschreiben Sie das entsprechende Signal im Zeitbereich (2 Punkte)

$$f(t) = 5 \sin(4\pi t) + 3 \sin(6\pi t) + \sin(12\pi t)$$

Nachname: _____ Vorname: _____ Matr.-Nr.: _____

Hiermit bestätige ich, dass ich das Praktikum bestanden habe: _____

(Unterschrift)

Punkte in Aufgabe	1. (15)	2. (18)	3. (5)	4. (14)	5. (16)	6. (11)	7. (13)	Σ (max. 92)

Note: _____

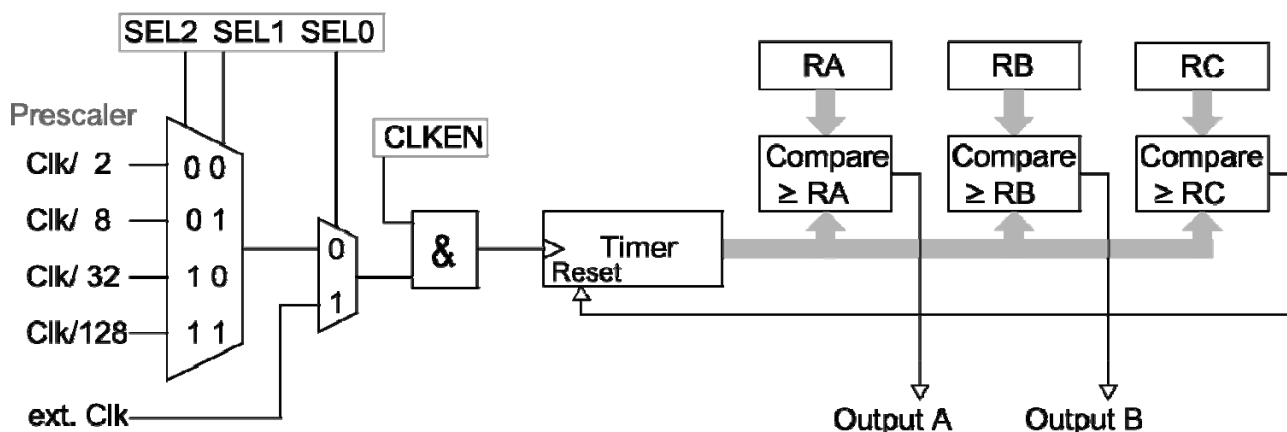
Zugel. Hilfsmittel: 1 Blatt DIN A4 handschriftlich mit Namen. (kein Druck!, keine Fotokopie!)

Bitte füllen Sie diese Klausurbogen in **deutlich lesbarer** Schrift aus, Antworten → hier, ggf Rückseite.

Eventuelle Punktabzüge erfolgen nur innerhalb der jeweiligen Teilaufgabe!

Aufgabe 1: Timer (18 Punkte)

Gegeben sei die folgende Prozessor-Architektur



Der **Timer** hat eine Auflösung von **16 Bit** ($2^{16} = 65536$), der CPU-Takt (**Clk**) ist **16 MHz**.

1) Der Timer soll mit einer Wiederholfrequenz von **100 Hz** laufen. Welcher **Prescale-Faktor** ist für eine **gute Auflösung** zu wählen und mit welchem Wert muss **RC** geladen werden.

a) Berechnen Sie den Wert **RC** für den **Prescale-Faktor 8**. Schreiben Sie die Gleichung hin.

$$RC = \frac{f_{clk}}{\text{Prescaler} \cdot f_{ziel}} = \frac{2^4 \cdot 10^6}{2^3 \cdot 10^2} = 2 \cdot 10^4 = \underline{\underline{20.000}}$$

b) Berechnen Sie den Wert **RC** für den **Prescale-Faktor 32**

$$RC = \frac{f_{clk}}{\text{Prescaler} \cdot f_{ziel}} = \frac{2^4 \cdot 10^6}{2^5 \cdot 10^2} = 2^{-1} \cdot 10^4 = \frac{1}{2} \cdot 10^4 = 0,5 \cdot 10^4 = \underline{\underline{5000}}$$



Nachname: _____

Vorname: _____

(Heftung darf nicht gelöst werden)

c) Welcher Prescale-Faktor führt zu einer besseren Auflösung des Timers? *Begründen Sie Ihre Antwort stichwortartig.*

Je kleiner der Abstand zwischen jedem Inkrement des Timers, umso näher kommen wir an unsere Zielperiode. In diesem Fall ist der Prescaler 8 die genaueste Auflösung.

2. Wie lange ist die Periodendauer bei 100 Hz?

$$t = \dots \quad T_{\text{Ziel}} = \frac{1}{f_{\text{Ziel}}} = \frac{1}{100} = 0,01s = 10 \text{ ms}$$

3. Mit der 100 Hz Grundfrequenz soll ein pulswidenmoduliertes (PWM) Signal mit dem Register RA am Ausgang Output A mit dem Tastverhältnis von **50%** erzeugt werden.

Mit welchem Wert muss der Zähler RA geladen werden? *Erläutern Sie stichwortartig Ihre folgenden Berechnungen.*

$$20.000 : 2 = 10.000$$

Da wir 50% High und 50% Low Signale erzeugen wollen, müssen wir die Hälfte von R_c für R_A nehmen.

$$RA = \underline{10.000}$$

4. Wie müssen im Programm aus der Aufgabenstellung 1.-3. die Register SEL (SEL0...2) und CLKEN gesetzt werden, damit an **Output A 50% Pulsweitenmodulation (PWM)** ausgegeben wird? *Geben Sie alle Werte hexadezimal an und erläutern Sie unbedingt stichwortartig die Bedeutung der einzelnen Bits!*

$$SEL_{*1} = \underline{0x2}; \quad // \dots \quad SEL2 = 0 \quad SEL1 = 1 \quad SEL0 = 0$$

$$CLKEN_{*2} = \underline{0x1}; \quad // \dots \text{ & - Gatter}$$

*₁ SEL 0 = keine externe Clock

*₂

Eingang	CLKEN	Q
0	0	0
0	1	0
1	0	0
1	1	1

Wir wollen die Eingänge so übernehmen wie sie sind.

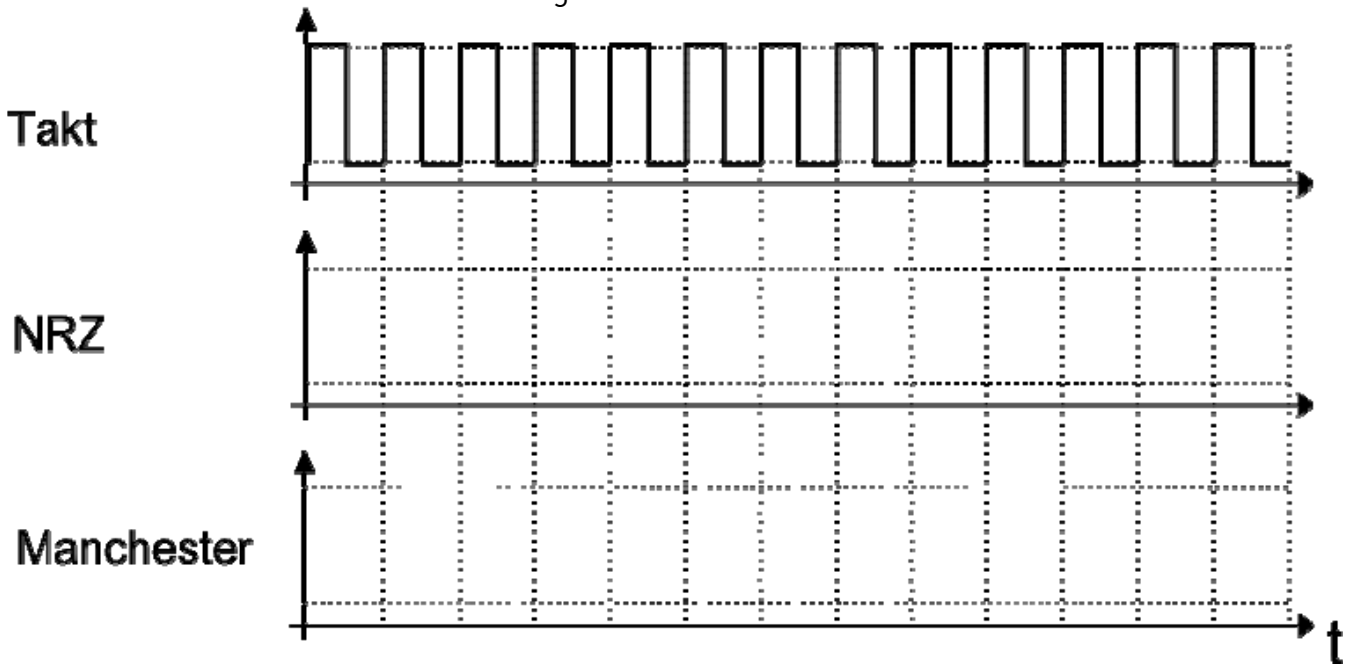


Nachname: _____ Vorname: _____ (Heftung darf nicht gelöst werden)

Aufgabe 2: Codierung (18 Punkte)

A. Gegeben sei die Bitfolge **0 1 0 1 1 0 0 0 1 0**

1. Zeichnen Sie die gegebene Bitfolge in das Diagramm unter Berücksichtigung des Taktes für die NRZ- und Manchester-Codierung ein.



2. Bei welcher Codierung kann nur von den Daten auf die Taktrate geschlossen werden?
Antworten und begründen Sie in Stichworten.

B. In der Vorlesung wurde die **serielle Schnittstelle** besprochen und im Praktikum verwendet. Beantworten Sie dazu die folgenden Fragen.

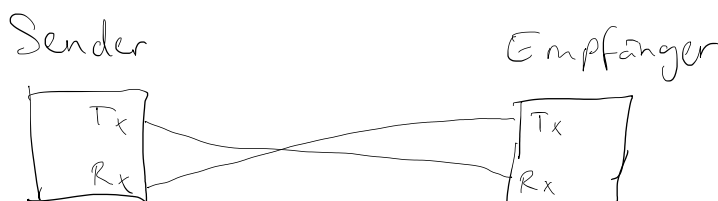
1) Wie viele Leitungen außer der Masseleitung werden bei dieser Schnittstelle mindestens benötigt?

Anzahl: 2

2) Wie heißen die Leitungen und was ist deren Funktion?

Transmit = Senden
Receive = Empfangen

3) Es seien zwei Teilnehmer A und B am seriellen Bus. Wie werden die Leitungen beider Teilnehmer miteinander verbunden? *Skizzieren Sie oder beschreiben Sie stichwortartig.*





Nachname: _____ Vorname: _____ (Heftung darf nicht gelöst werden)

4) Handelt es sich um eine synchrone oder asynchrone Übertragung? *Begründen Sie Ihre Antwort.*

asynchron

5) Bei einer seriellen Schnittstelle sei die Angabe 9600 8N1 gegeben? *Erläutern Sie die Bedeutung der Angaben in Stichworten.*

9600 = Baudrate. Die Zustandsänderungen der Bits pro Sekunde.

8 = Nutzbits (Datenbits)

N = NOT $\Rightarrow$ Keine Parität

1 = Stopbit

Aufgabe 3 Allgemeine Fragen (5 Punkte)

Mikrocontroller. Kreuzen Sie an, falsche Antworten führen zu Abzug

	ja	nein
Bei Mikrocontrollern sind Ein-/Ausgabe-, Interruptsteuerung, A/D-, D/A-Wandler, Arbeits- und Programmspeicher auf einem Chip integriert.	☒	☐
Bei Mikrocontrollern verbrauchen z.B. Timer und Counter vergleichsweise viel CPU-Zeit.	☐	☒
Zähler und Zeitgeber müssen bei Mikrocontrollern extern angeschlossen werden	☐	☒
Größe und Typ des Speichers unterscheiden oft verschiedene Untertypen desselben Mikrocontrollers	☒	☐
Mikrocontroller enthalten integrierten Speicher, jedoch nur einen Typ. Entweder RAM oder FLASH oder EEPROM	☐	☒

h_da Dr. Horsch Fachbereich I	Mikroprozessortechnik „Halbzeittest“	Name: _____ Matrikel-Nr.: _____
--	--	------------------------------------

Aufgabe 5:

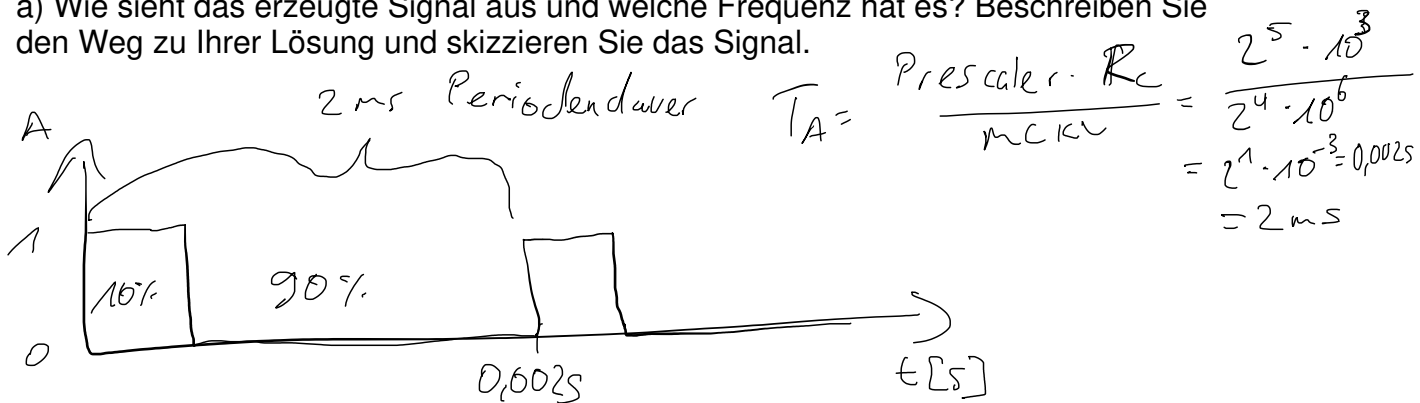
6 Punkte

Schauen Sie sich die Initialisierung des Timer 3 bei einem Prozessortakt (MCKI) von **16 MHz** an und beantworten Sie die gestellten Fragen:

```
void Timer3_init( void ) {
.....
// Initialisieren des Modus von Timer 3
timerbase3->TC_CMR =
    TC_ACPC_CLEAR_OUTPUT | // Register C Vergleich löscht TIOA
    TC_ACPA_SET_OUTPUT | // Register A Vergleich setzt TIOA
    TC_WAVE | // Waveform Modus
    TC_CPCTRG | // Register C Vergleich setzt Zähler zurück
    TC_CLKS_MCK32; // TCCLKS: MCKI / 32

// Initialisieren der Compare-Register des Timers:
timerbase3->TC_RA = 100; // Wert im Compare RA Register
timerbase3->TC_RC = 1000; // Wert im Compare RC Register
// Start des Timers:
timerbase3->TC_CCR = TC_CLKEN; // Aktiviere Timertakt
timerbase3->TC_CCR = TC_SWTRG; // Zurücksetzen des Timers
.....
}
```

a) Wie sieht das erzeugte Signal aus und welche Frequenz hat es? Beschreiben Sie den Weg zu Ihrer Lösung und skizzieren Sie das Signal.



b) Ändern Sie die Initialisierung so, dass ein symmetrisches Rechtecksignal entsteht mit der Frequenz von 1 kHz.