

Musterlösung

Automatisierungstechnik I – SS18

Name:	Vorname:	Matrikel-Nr:
-------	----------	--------------

- Prüfungstag:** 24.08.2018
- Prüfungsdauer:** 120 Minuten
- Maximalpunktzahl:** 120 Punkte
- Prüfungsumfang:** 6 Aufgaben (21 Seiten)
- Hilfsmittel:** Alle, außer Kommunikationsgeräte
- Hinweise:**
- 1.) Verlangt sind alle Aufgaben.
 - 2.) Verwenden Sie **keinen Rotstift / Grünstift / Bleistift**.
 - 3.) Tragen Sie auf dem **Deckblatt** Ihren **vollen Namen** und Ihre **Matrikelnummer** ein.
 - 4.) Tragen Sie auf **allen Blättern** Ihre **Initialen** (1. Buchstabe von Vor- und Nachnamen) ein.
 - 5.) Es werden **nur** die Blätter **gewertet**, die mit Ihren **Initialen** versehen sind.
 - 6.) Verwenden Sie für **Ihre Lösungen unbedingt** die jeweiligen Lösungsfelder.
 - 7.) Geben Sie dieses **Deckblatt zusammen** mit Ihren in der **richtigen Reihenfolge sortierten Aufgabenblättern** ab.

Alle Angaben ohne Gewähr.

Stand: 04.08.2020

----- Nicht von Studierenden auszufüllen -----

Aufgabe	1	2	3	4	5	6		
Maximal	14,5	30,5	16	17	17,5	24,5		
Erreicht								

Aufgabe 1: Prozessperipherie (14,5 Punkte)

Ein Autokonzern entwickelt einen PKW speziell für Fahranfänger. Das Fahrzeug soll besonders sicher sein und verschiedene Anfängerfehler zu vermeiden helfen.

Ein wichtiges Eingangssignal vieler Fahrassistenzsysteme ist der jeweils aktuelle Lenkwinkel. Dieser soll mittels eines smarten Winkelsensors auf Basis eines Schleifwiderstands erfasst werden. Am Ende der A/D-Wandlung innerhalb des Sensors wird ein Rohwert y in Abhängigkeit vom Lenkwinkel α und dem maximalen Rohwert $y_{\max}$ gemäß der folgenden Gleichung ausgegeben:

$$y_{\alpha,R}(\alpha) = \frac{100 \, \Omega}{600 \, \Omega + 0,9 \frac{\Omega}{^\circ} \cdot \alpha} \cdot y_{\max} \cdot \frac{9 \, V}{12 \, V}$$

Aufgabe 1.1: Auswahl eines A/D-Wandlers (6,5 Punkte)

Der Lenkwinkelsensor soll in der Lage sein, den Lenkwinkel mit einer Genauigkeit von $\pm 0,1^\circ$ auszugeben. Wie viel Bit müssten die vom A/D-Wandler generierten Rohwerte mindestens lang sein, um diese Genauigkeit ermöglichen zu können? Füllen Sie zur Beantwortung dieser Frage Tabelle 1 vollständig aus.

Tabelle 1: Vergleich unterschiedlicher A/D-Wandler-Auflösungen

	8 Bit	12 Bit	16 Bit
$y_{\alpha,R}(-180^\circ) =$	43	701	11.221
$y_{\alpha,R}(180^\circ) =$	25	403	6.450
$ y_{\alpha,R}(-180^\circ) - y_{\alpha,R}(180^\circ) =$	18	298	4.771

Reichen die in Tabelle 1 angestellten Berechnungen prinzipiell aus, um für mindestens eine der betrachteten Auflösungen eine Genauigkeit von $\pm 0,1^\circ$ über den gesamten Wertebereich (-180° bis 180°) zu garantieren? Bitte begründen Sie Ihre Antwort.

Nein, da $y_{\alpha,R}(\alpha)$ nicht linear ist und die Anzahl der zur Verfügung stehenden Rohwerte damit noch keine Aussage über die Schrittweite bzw. Genauigkeit zulässt.

Diese Teilaufgabe ist nicht mehr prüfungsrelevant.

Aufgabe 2: Scheduling (30,5 Punkte)

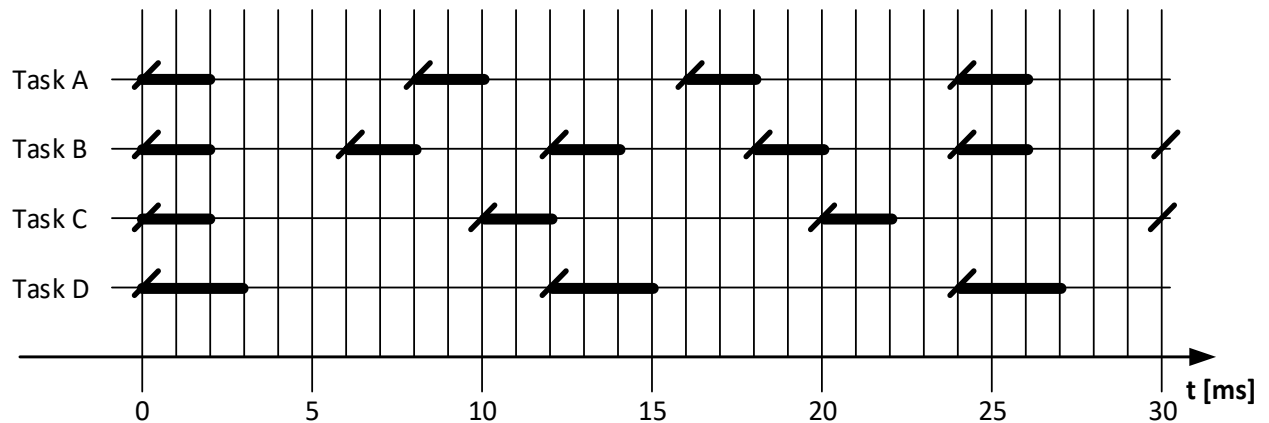


Abbildung 1: Soll-Ablauf der Tasks A, B, C und D

Gegeben sei ein periodisches Taskset mit vier Tasks und einem Soll-Ablaufverhalten entsprechend Abbildung 1. Die Periode ist gleichzeitig als Deadline für die Ausführung zu interpretieren. Die Prioritäten der Tasks können der Tabelle 3 entnommen werden. Der Systemtakt beträgt 1 ms, es steht ein Rechenkern zur Verfügung.

Tabelle 3: Prioritäten

Task	Priorität
Task A	3
Task B	1
Task C	2
Task D	4

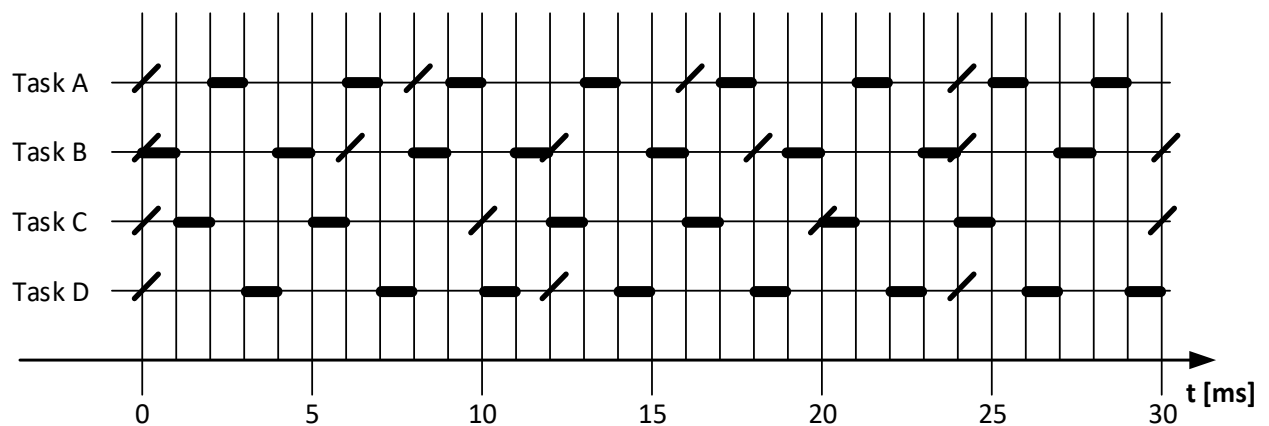


Abbildung 2: Ist-Ablauf der Tasks A, B, C und D (Zeitscheibenverfahren)

Aufgabe 2.1: Gleichzeitigkeit und Rechtzeitigkeit (4 Punkte)

a) Erfüllt die Abarbeitung der einzelnen Tasks im Bereich $0 \leq t < 30$ ms gemäß Abbildung 2 die Rechtzeitigkeits-Bedingung? Wenn nein, welche Task(s) verletzt bzw. verletztten die Rechtzeitigkeits-Bedingung wann zum ersten Mal?

Task	Rechtzeitige Abarbeitung?		Wenn nein, zu welchem Zeitpunkt wird die ersten Deadline verletzt?
Task A	Ja ☒	Nein ☐	$t = \underline{\hspace{2cm}}$ ms
Task B	Ja ☐	Nein ☒	$t = \underline{\hspace{1cm}}18\underline{\hspace{1cm}}$ ms
Task C	Ja ☒	Nein ☐	$t = \underline{\hspace{2cm}}$ ms
Task D	Ja ☒	Nein ☐	$t = \underline{\hspace{2cm}}$ ms

b) Erfüllt die Abarbeitung der einzelnen Tasks im Bereich $0 \leq t < 30$ ms gemäß Abbildung 2 die Anforderungen für Gleichzeitigkeit? Wenn nein, nennen Sie den Zeitpunkt der ersten auftretenden Verletzung der Gleichzeitigkeits-Bedingung.

Nein, bei $t = 8$ ms findet die erste Überholung statt.

c) Bestimmen Sie die Auslastung des Rechenkerns bei einer Beauftragung entsprechend Abbildung 1 und Tabelle 3.

Raum für Nebenrechnungen:

Rechenkernauslastung $U = 1,033$

Aufgabe 2.2: Abweichungen (6 Punkte)

Hinweis: Diese Teilaufgabe kann unabhängig von den vorhergehenden bearbeitet werden.

Wie oft werden die einzelnen Tasks im Bereich $0 \leq t < 30$ ms gemäß Abbildung 2 jeweils beauftragt? Wie viele Beauftragungen gibt es in diesem Zeitraum insgesamt?

Anzahl Beauftragungen von Task A:	4	Anzahl Beauftragungen von Task B:	5
Anzahl Beauftragungen von Task C:	3	Anzahl Beauftragungen von Task D:	3
Anzahl Beauftragungen insgesamt:	15		

Wie groß sind die jeweiligen Summen der Abweichungen zwischen Soll- und Ist-Ablauf der einzelnen Tasks sowie des Tasksets im Bereich $0 \leq t < 30$ ms gemäß Abbildung 2?

Hinweis: Bei unvollendeten Beauftragungen sind auch und nur die bis zum Ende des Betrachtungszeitraums entstandenen Abweichungen zu betrachten.

Abweichungen Task A (Ausführungsbeginn):	5 ms	Abweichungen Task A (Ausführungsende):	16 ms
Abweichungen Task B (Ausführungsbeginn):	≥ 16 ms	Abweichungen Task B (Ausführungsende):	≥ 27 ms
Abweichungen Task C (Ausführungsbeginn):	3 ms	Abweichungen Task C (Ausführungsende):	12 ms
Abweichungen Task D (Ausführungsbeginn):	7 ms	Abweichungen Task D (Ausführungsende):	≥ 20 ms
Abweichungen Taskset (Ausführungsbeginn):	≥ 31 ms	Abweichungen Taskset (Ausführungsende):	≥ 75 ms

Wie groß sind die mittleren Abweichungen der einzelnen Tasks im Bereich $0 \leq t < 30$ ms gemäß Abbildung 2? Wie groß ist die mittlere Abweichung des Tasksets in diesem Zeitraum?

Mittlere Abweichung Task A (Ausführungsbeginn):	1,25 ms	Mittlere Abweichung Task A (Ausführungsende):	4 ms
Mittlere Abweichung Task B (Ausführungsbeginn):	$\geq 3,2$ ms	Mittlere Abweichung Task B (Ausführungsende):	$\geq 5,4$ ms
Mittlere Abweichung Task C (Ausführungsbeginn):	1 ms	Mittlere Abweichung Task C (Ausführungsende):	4 ms
Mittlere Abweichung Task D (Ausführungsbeginn):	2,33 ms	Mittlere Abweichung Task D (Ausführungsende):	$\geq 6,67$ ms
Mittlere Abweichung Taskset (Ausführungsbeginn):	$\geq 2,067$ ms	Mittlere Abweichung Taskset (Ausführungsende):	≥ 5 ms

Aufgabe 2.3: Last In, First Out (4,5 Punkte)

Hinweis: Diese Teilaufgabe kann unabhängig von den vorhergehenden bearbeitet werden.

Analog zum in der Übung eingeführten First-In-First-Out-Verfahren kann auch ein Last-In-First-Out-Verfahren gebildet werden. Dieses Verfahren räumt nicht der ältesten, sondern der jüngsten Beauftragung Vorrang ein. Etwaige nachrangige Entscheidungskriterien bleiben unverändert. Ermitteln Sie das zeitliche Ist-Verhalten der Tasks bei einer Beauftragung

entsprechend Abbildung 1 und Tabelle 3 beim Scheduling nach dem nicht-präemptiven Last-In-First-Out-Verfahren im Bereich $0 \leq t < 30$ ms. Zeichnen Sie Ihre Lösung in Abbildung 3 ein.

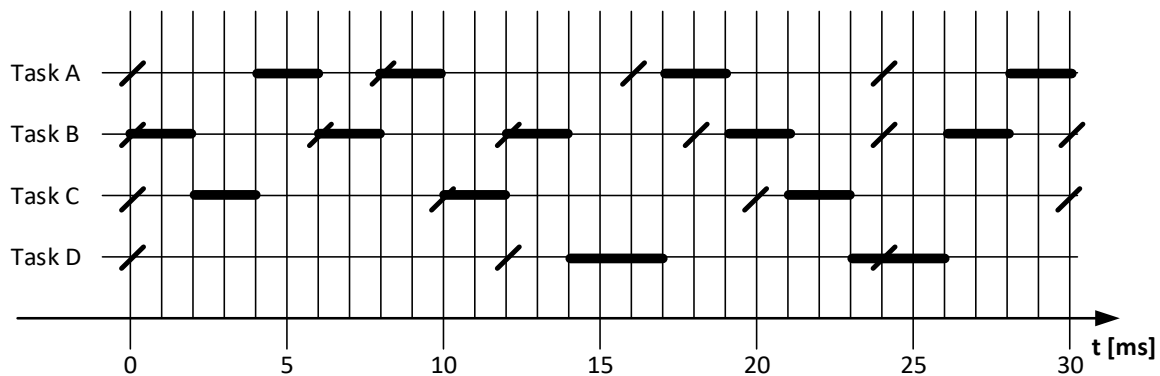


Abbildung 3: Ist-Ablauf der Tasks A, B, C und D (Last-In-First-Out)

Aufgabe 2.4: Kleinsten Spielraum (12 Punkte)

Hinweis: Diese Teilaufgabe kann unabhängig von den vorhergehenden bearbeitet werden.

Abbildung 4 soll das zeitliche Ist-Verhalten der Tasks bei einer Beauftragung entsprechend Abbildung 1 und Tabelle 3 beim Scheduling nach dem Verfahren des kleinsten Spielraums im Bereich $0 \leq t < 30$ ms darstellen. Es haben sich jedoch neun Fehler eingeschlichen. Markieren Sie die Fehler in Abbildung 4 und begründen Sie Ihre Wahl in Tabelle 4. Die Fehler können sämtliche Elemente des Diagramms betreffen, wobei jeder Fehler nur ein einzelnes Element betrifft (ein Fehler erzeugt also keine Folgefehler).

Hinweis: Ein Fehler wurde als Beispiel bereits markiert. Es verbleiben acht Fehler für Sie. Sie können Abbildung 5 zu Ihrer Unterstützung nutzen (nicht bepunktungsrelevant).

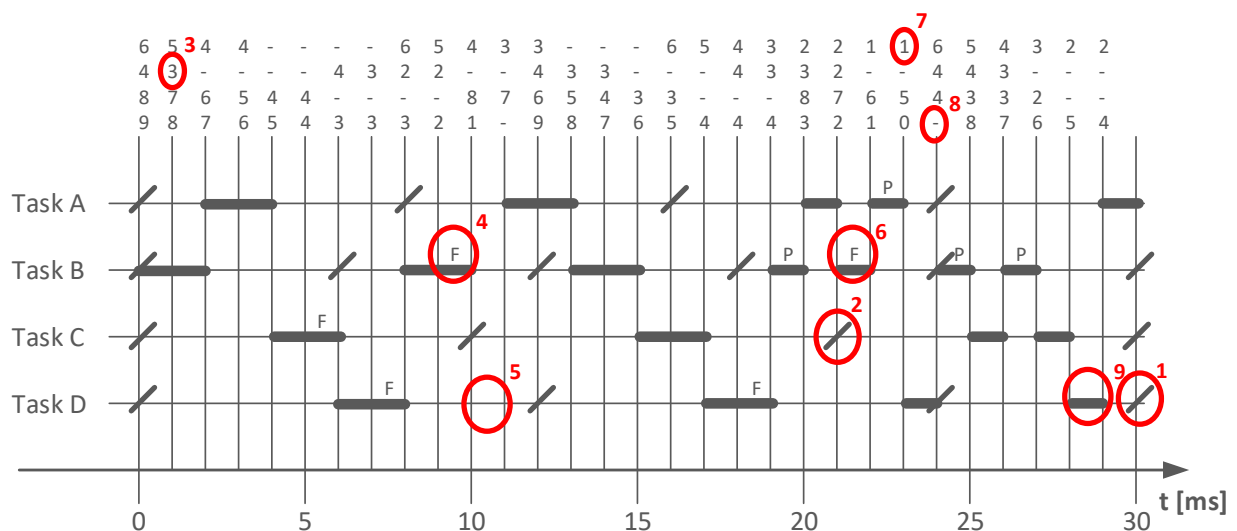
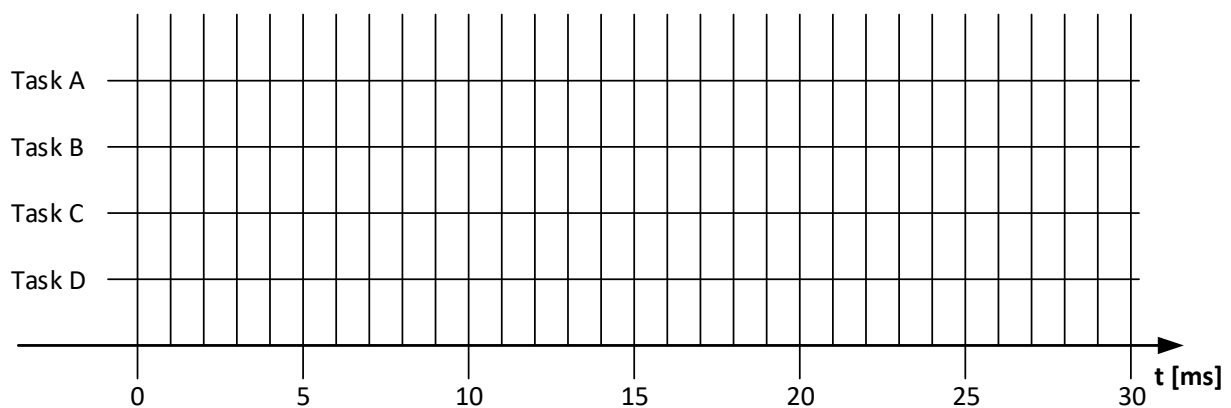


Abbildung 4: Ist-Ablauf der Tasks A, B, C und D (Kleinsten Spielraum)

Tabelle 4: Begründung bzw. Korrektur der gefundenen Fehler

Nr.	Begründung / Korrektur („Warum ist das ein Fehler?“ oder „Wie wäre es richtig?“)
1	Falscher Zeitpunkt (Korrekt: 24 ms oder 36 ms)
2	Falscher Zeitpunkt (Korrekt: 20 ms)
3	Falscher Spielraum (Korrekt: 4)
4	Falsches Kriterium (Korrekt: P)
5	Strich fehlt / Task D läuft nicht (Korrekt: Task D läuft)
6	Falsches Kriterium (Korrekt: P)
7	Falscher Spielraum (Korrekt: -)
8	Falscher Spielraum (Korrekt: 9)
9	Strich verrutscht / Falscher Task (Korrekt: Task A läuft)

**Abbildung 5: Leeres Diagramm zu Ihrer freien Verfügung**

Aufgabe 2.5: Bestimmung der Prozessparameter (4 Punkte)

Hinweis: Diese Teilaufgabe kann unabhängig von den vorhergehenden bearbeitet werden.

Gegeben ist der in Abbildung 6 dargestellte Ist-Ablauf von vier Tasks nach dem Scheduling-Verfahren mit festen Prioritäten (präemptiv). Gehen Sie davon aus, dass alle Tasks bei $t = 0$ ms zum ersten Mal beauftragt wurden und dass die Periode einer Task konstant und gleichzeitig als die Deadline für deren Ausführung zu betrachten ist. Der Systemtakt beträgt 2 ms.

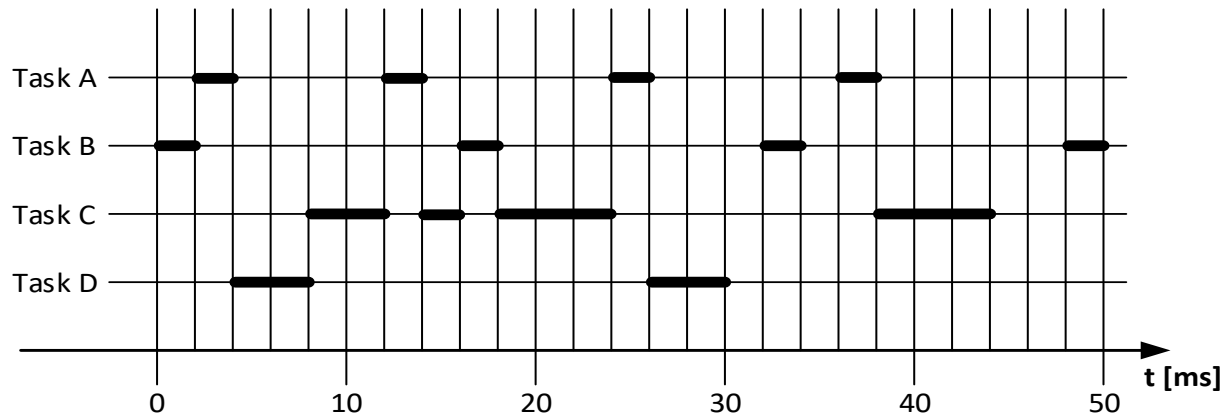


Abbildung 6: Ist-Ablauf der Tasks A, B, C und D (feste Prioritäten)

Ermitteln Sie Priorität, Ausführungsdauer und Periode der einzelnen Tasks und tragen Sie Ihre Ergebnisse in Tabelle 5 ein.

Tabelle 5: Parameter des Tasksets

Task	Ausführungsdauer	Periode	Priorität
Task A	2	12	2
Task B	2	16	1
Task C	6	18	4
Task D	4	26	3

Aufgabe 3: Cloudbindung (16 Punkte)

Aufgabe 3.1: Authentifizierung und Verschlüsselung (7 Punkte)

a) Alice (Client A) möchte Charly (Client C) eine verschlüsselte Botschaft senden. Die von ihr zu diesem Zweck generierte Nachricht hat folgenden Inhalt:

$$c || sig_{K_A^-, c} || C_A, \text{ wobei gilt } c = E(K_B^+, m)$$

Welches Problem hat Charly mit dieser Nachricht? Wie hätte die Nachricht korrekterweise aussehen müssen?

Alice hat den falschen öffentlichen Schlüssel (K_B^+) verwendet, um die Nachricht an Charly zu versenden. Korrekt wäre gewesen:

$$c = E(K_C^+, m)$$

b) In einem völlig anderen Szenario erhält Bob (Client B) von einer anderen Alice (Client A) die Nachricht

$$c'' || sig_{K_A^-, c'} || C_A, \text{ wobei gilt } c'' = E(K_B^+, m)$$

Zusätzlich gilt:

$$sig_{K_A^-, c'} = 10;$$

$$D(K_A^+, sig_{K_A^-, c'}) = 15;$$

$$D(K_B^-, c'') = 18;$$

$$c'' = 8;$$

$$D(K_B^-, sig_{K_A^-, c'}) = 14;$$

$$h(c'') = 14$$

Das Zertifikat C_A von Alice ist gültig. Beurteilen Sie ob $c' = c''$ gilt (und die Nachricht damit gültig ist). Bitte begründen Sie Ihre Antwort.

Nein $c' = c''$ gilt nicht, da ansonsten auch $h(c'') = D(K_A^+, sig_{K_A^-, c'})$ und damit $14 = 15$ gelten müsste.

Aufgabe 3.2: Vektor-Uhren (9 Punkte)

Hinweis: Diese Teilaufgabe kann unabhängig von den vorhergehenden bearbeitet werden.

a) Abbildung 7 zeigt ein Vektoruhrsystem mit 3 Prozessen aber ohne die Pfeile, die normalerweise die zeitlichen Beziehungen zwischen den Prozessen kennzeichnen. Ermitteln Sie die entsprechenden Informationen aus den Zeitvektoren und zeichnen Sie die Pfeile in die Abbildung ein.

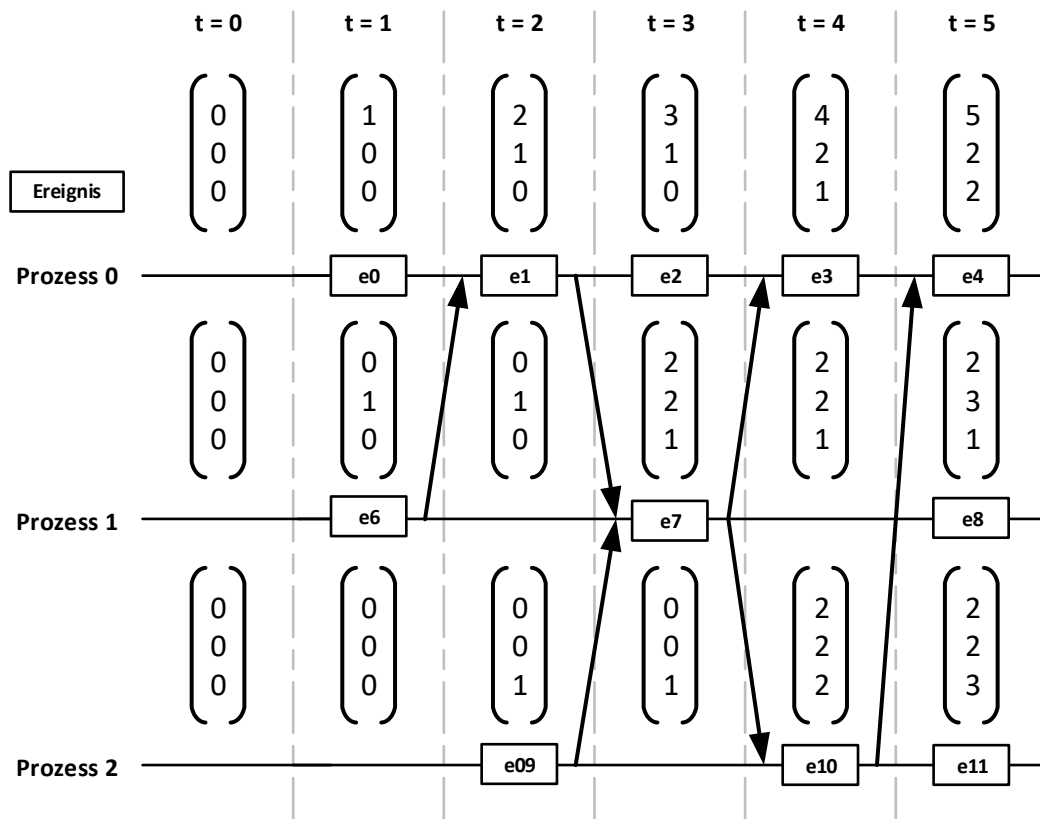


Abbildung 7: Vektoruhrsystem mit 3 Prozessen

b) Stimmt die Behauptung $C_2(e9) < C_0(e2)$ in Bezug auf das Vektoruhrsystem aus Abbildung 7? Bitte begründen Sie Ihre Antwort mathematisch.

Nötig wäre:

1. $\forall x: \vec{C}_i[x] \leq \vec{C}_j[x]$: Nicht erfüllt für $x = 3$ ($1 > 0$)
2. $\exists x: \vec{C}_i[x] < \vec{C}_j[x]$: Erfüllt für $x = 1$ ($0 < 3$) und $x = 2$ ($0 < 1$)

Daher: Nein, die Behauptung stimmt nicht.

Aufgabe 4: Semaphoren (17 Punkte)

Hinweis: Sollten Sie sich verzeichnen, so können Sie den Diagrammvordruck aus Abbildung 13 verwenden. Er dient als Ersatz.

Gegeben sei der Ablauf dreier Tasks A, B und C gemäß Abbildung 8. Das ausführende System verfügt über einen Rechenkern. Zu Beginn haben alle Semaphore S0, S1 und S2 den Wert 1. Ferner hat Task C die höchste Priorität und Task A die niedrigste.

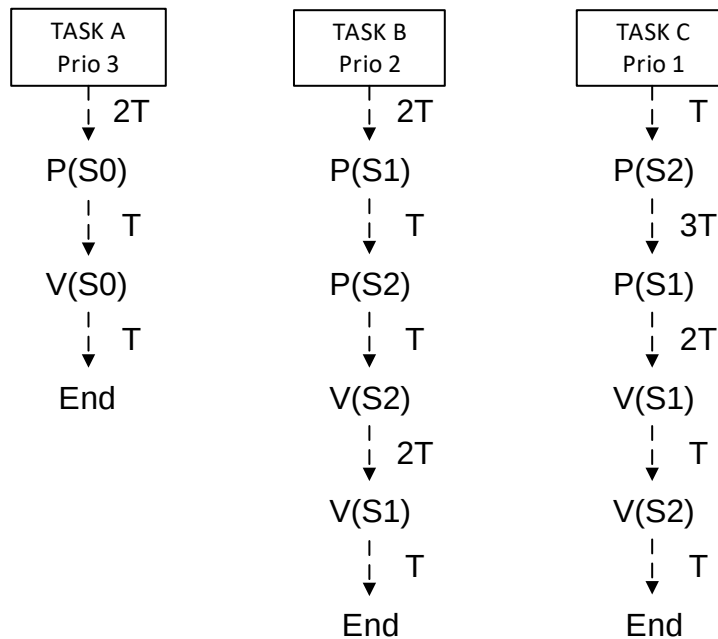


Abbildung 8: Struktur des Tasksets

Tabelle 6 nennt den Zeitpunkt der einmaligen Aktivierung der drei Tasks.

Tabelle 6: Aktivierungszeiten

Task	Aktivierungszeit
A	$t = 0 \text{ T}$
B	$t = 4 \text{ T}$
C	$t = 7 \text{ T}$

Nutzen Sie im Folgenden das präemptive Scheduling nach festen Prioritäten.

Aufgabe 4.1: Soll-Ablauf (2 Punkte)

Ermitteln Sie das zeitliche Soll-Verhalten der Tasks im Bereich $0 \leq t < 20 \text{ T}$. Zeichnen Sie Ihre Lösung in Abbildung 9 ein.

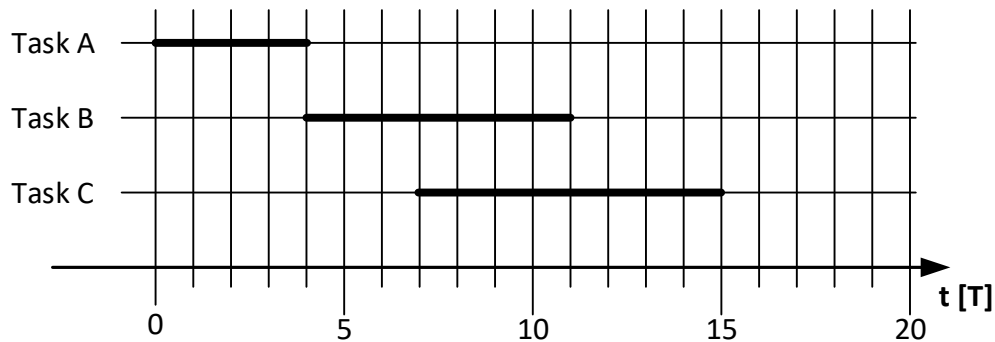


Abbildung 9: Soll-Ablauf der Tasks A, B und C

Aufgabe 4.2: Ist-Ablauf (6 Punkte)

Hinweis: Diese Teilaufgabe kann unabhängig von den vorhergehenden bearbeitet werden.

Hinweis: Die Kennzeichnung des Verlaufs der Semaphor-Variablen ist nicht bewertungsrelevant, kann aber im Fehlerfall die Vergabe von Folgefehler-Punkten erleichtern.

a) Ermitteln Sie das zeitliche Ist-Verhalten der Tasks im Bereich $0 \leq t < 20$ T. Zeichnen Sie Ihre Lösung in Abbildung 10 ein.

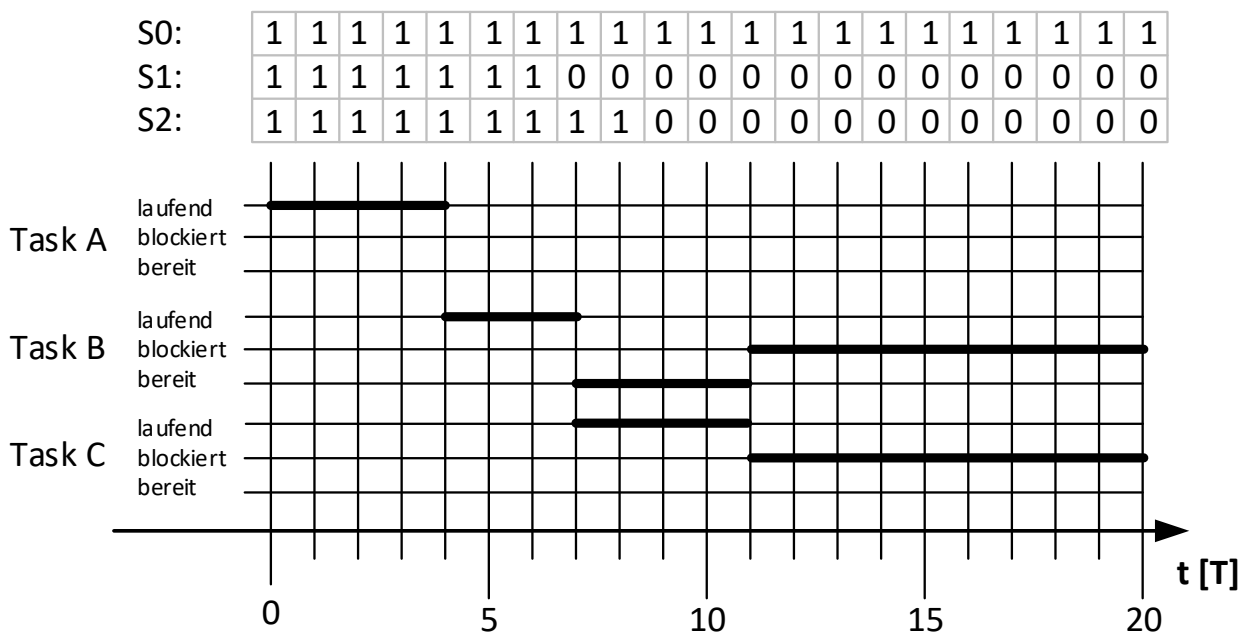


Abbildung 10: Ist-Ablauf der Tasks A, B und C

b) Wie lange wird Task A in ihrer Durchführung verzögert und warum ist das so?

Verzögerung:	0 T
Begründung:	Task A ist abgearbeitet, bevor andere Tasks starten.

Aufgabe 4.3: Priority Inheritance Protocol (3 Punkte)

Hinweis: Diese Teilaufgabe kann unabhängig von den vorhergehenden bearbeitet werden.

Hinweis: Die Kennzeichnung des Verlaufs der Semaphor-Variablen ist nicht bepunktungsrelevant, kann aber im Fehlerfall die Vergabe von Folgefehler-Punkten erleichtern.

Ein gängiger Ansatz zur Optimierung des Ablaufverhaltens eines Tasksets bei gemeinsamer Nutzung von Betriebsmitteln ist das Priority Inheritance Protocol. Ermitteln Sie das zeitliche Ist-Verhalten der Tasks im Bereich $0 \leq t < 20 T$ bei Anwendung des genannten Ansatzes. Zeichnen Sie Ihre Lösung in Abbildung 11 ein.

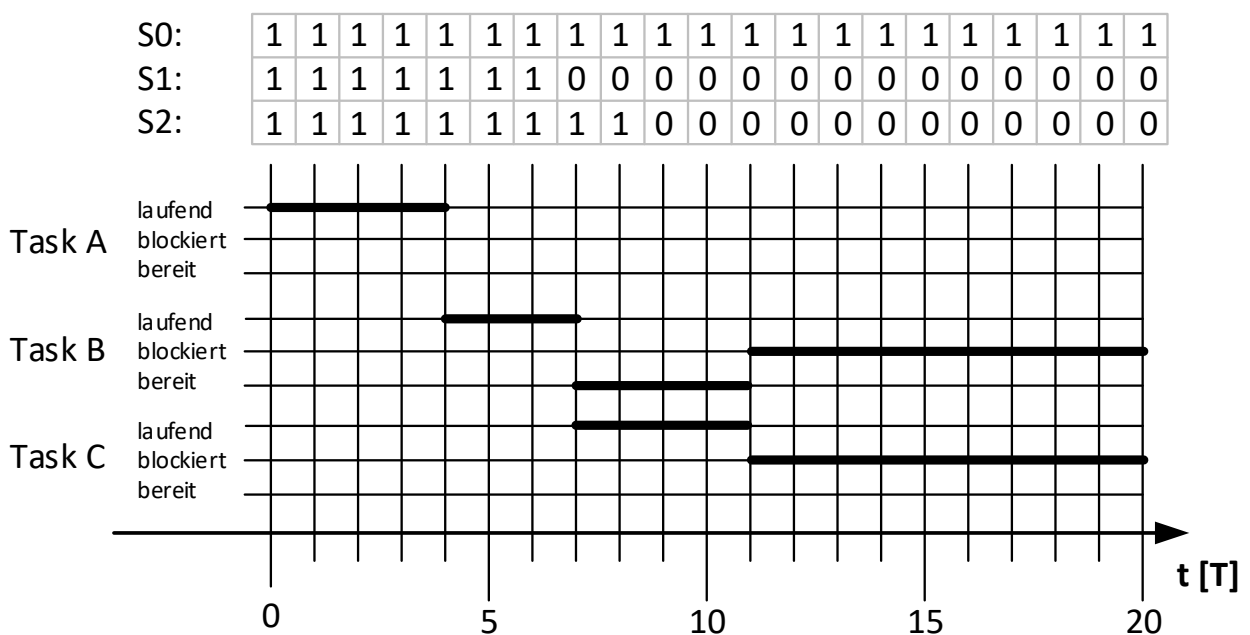


Abbildung 11: Ist-Ablauf der Tasks A, B und C (Priority Inheritance)

Aufgabe 4.4: Priority Ceiling Protocol (6 Punkte)

Hinweis: Die Kennzeichnung des Verlaufs der Semaphor-Variablen ist nicht bepunktungsrelevant, kann aber im Fehlerfall die Vergabe von Folgefehler-Punkten erleichtern.

a) Ermitteln Sie das zeitliche Ist-Verhalten der Tasks bei Anwendung des Priority Ceiling Protocols im Bereich $0 \leq t < 20 T$. Zeichnen Sie Ihre Lösung in Abbildung 12 ein.

Priority Ceiling S0	3
Priority Ceiling S1	1
Priority Ceiling S2	1

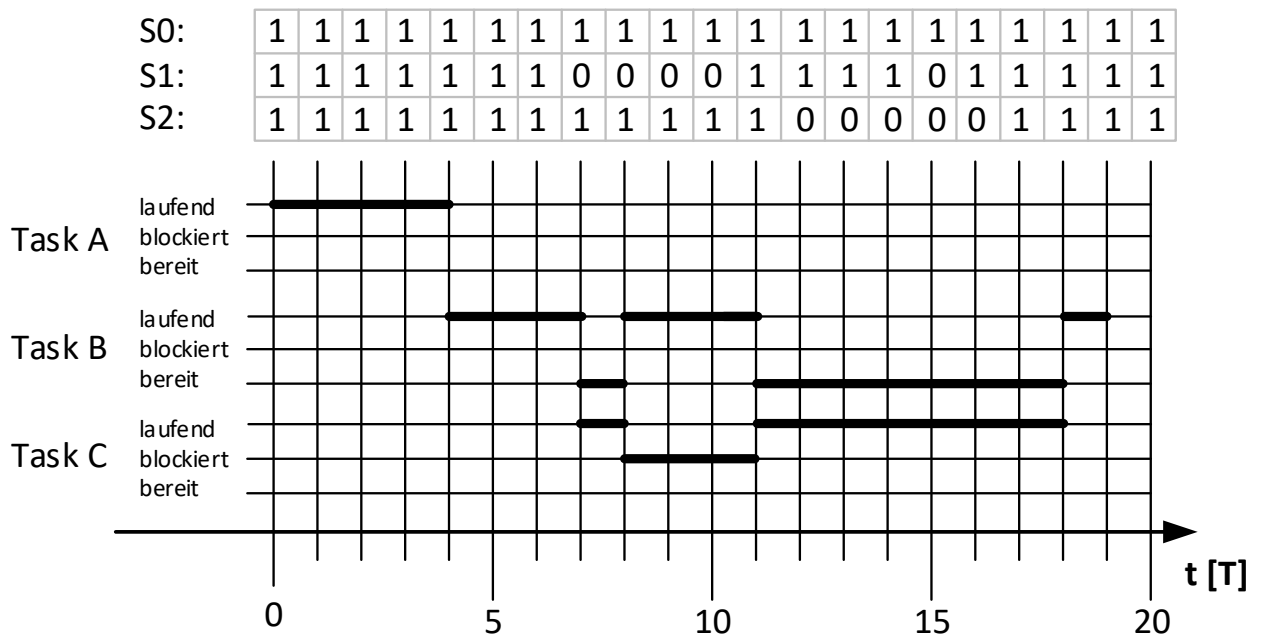


Abbildung 12: Ist-Ablauf der Tasks A, B und C (Priority Ceiling)

Aufgabe 4. _____:

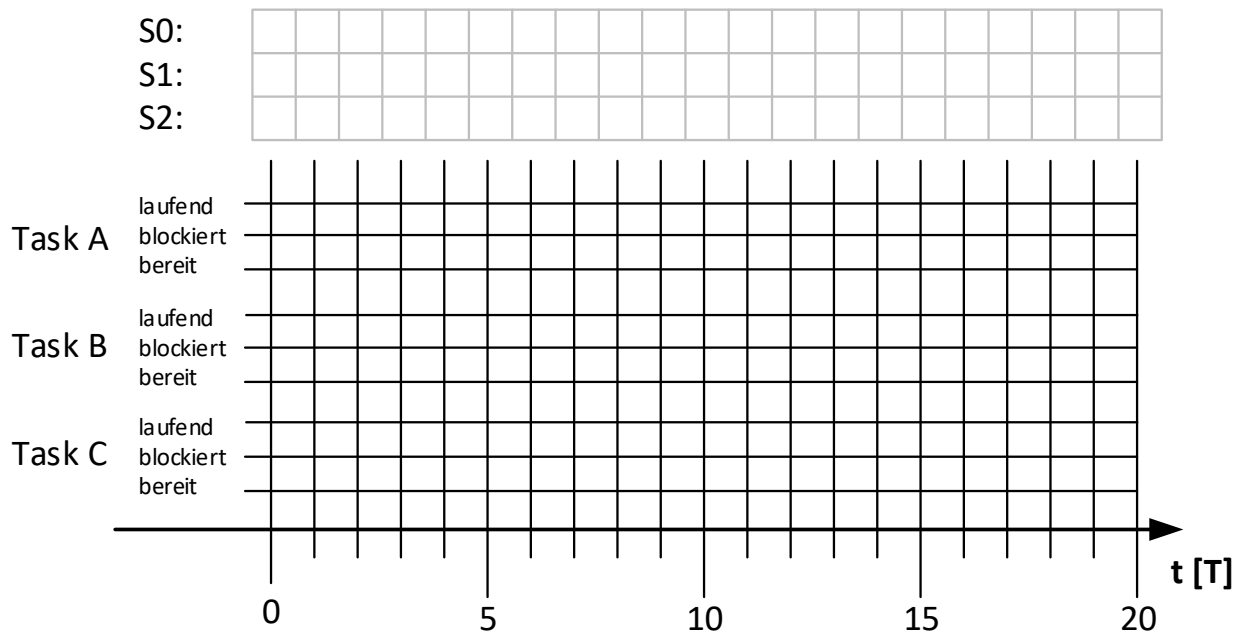


Abbildung 13: _____-Ablauf der Tasks A, B und C (Ersatz-Diagramm)

Aufgabe 5: Speicherprogrammierbare und Pneumatische Steuerungen (17,5 Punkte)

Aufgabe 5.1: Structured Text (10,5 Punkte)

Ein Kollege zeigt Ihnen folgendes SPS-Anwenderprogramm in der Sprache Structured Text und beklagt sich darüber, dass ihm zwar bekannt sei, dass es acht Fehler enthalte, er diese aber nicht finden könne. Markieren Sie die sieben Fehler in Abbildung 14 und begründen Sie Ihre Wahl in Tabelle 7.

Hinweis: Ein Fehler wurde als Beispiel bereits markiert. Es verbleiben sieben Fehler für Sie. Jeder Fehler ist einmalig (tritt also nur einmal auf).

```

1  NETWORK Tiefgaragenzufahrt:
2
3  VAR
4      Hauptschalter AT E0.0:BOOL;
5      Feuermelder_1 AT E0.1:BOOL;
6      Feuermelder_2 AT E0.2:BOOL;
7      Lueftung AT E1.0:BOOL;
8      Einfahrt_Ampel_Rot AT A0.0:BOOL;
9      Einfahrt_Ampel_Gruen AT A0.1:BOOL;
10     Ausfahrt_Ampel_Rot AT A1.0:BOOL;
11     Ausfahrt_Ampel_Gruen AT A0.1:BOOL;
12     Alarmhorn AT A2.0:BOOL;
13 END_VAR;
14
15 IF Hauptschalter = TRUE THEN
16     IF Feuermelder_1 = TRUE OR Feuermelder_2 = TRUE OR
17     Lueftung = FALSE
18         Einfahrt_Ampel_Rot := TRUE;
19         Einfahrt_Ampel_Gruen := FALSE;
20         Ausfahrt_Ampel_Rot := TRUE;
21         Ausfahrt_Ampel_Gruen := FALSE;
22     CASE Alarmton OF
23         0..498:
24         499:
25         IF Alarmhorn = TRUE THEN
26             Alarmhorn := FALSE;
27         ELSE
28             Alarmhorn := TRUE;
29         END_IF;
30     END_CASE;
31 ELSE

```

```

32      Einfahrt_Ampel_Rot := FALSE;
33      Einfahrt_Ampel_Gruen := TRUE;
34      Ausfahrt_Ampel_Rot := FALSE;
35      Ausfahrt_Ampel_Gruen := TRUE;
36  ELSEIF 7
37  8      Einfahrt_Ampel_Rot := TRUE;
38      Einfahrt_Ampel_Gruen := FALSE;
39      Ausfahrt_Ampel_Rot := TRUE;
40      Ausfahrt_Ampel_Gruen := FALSE;
41  END_IF;

```

Abbildung 14: SPS-Programm in ST

Tabelle 7: Begründung bzw. Beschreibung der gefundenen Fehler

Nr.	Begründung / Beschreibung („Warum ist das ein Fehler?“)
1	Falsches (Schlüssel-)Wort (Korrekt: „PROGRAM“)
2	Semikolon zu viel
3	Ausgang doppelt belegt
4	Semikolon zu viel
5	Falsches (Schlüssel-)Wort (Korrekt: „THEN“)
6	Variable nicht deklariert
7	If-Schleife nicht geschlossen („END_IF;“ fehlt) [kann an verschiedenen Stellen markiert werden]
8	Falsches (Schlüssel-)Wort (Korrekt: „ELSE“)

Aufgabe 5.2: Pneumatische Steuerung (7 Punkte)

Hinweis: Diese Teilaufgabe kann unabhängig von den vorhergehenden bearbeitet werden.

Schon die Abläufe in einer Dampfmaschine vor 200 Jahren waren eine Art automatisch ablaufende pneumatische Steuerung. Zur Unterhaltung und Ausbildung jüngerer Geschwister entscheiden Sie sich, eine solche Dampfmaschine nachzubauen.

Sie haben zu diesem Zweck einen doppelwirkenden Zylinder, eine Druckluftquelle sowie verschiedene Ventile zur Auswahl. Konstruieren müssen Sie noch einen geeigneten Ein- und Ausschalter.

Der Zylinder soll im Betrieb dauerhaft in Bewegung sein und abwechselnd seine beiden Endpunkte anfahren. Per Knopfdruck soll die Anlage in Bewegung gesetzt werden. Ein erneuter Knopfdruck soll den Stillstand der Anlage bewirken.

Zeichnen Sie den Ein-/Ausschaltmechanismus in das vorgesehene Feld. Nutzen Sie diesen Mechanismus, den Zylinder und mindestens eines der vorgedruckten Ventile um die gewünschte Funktionalität sicherstellen zu können. Bitte benennen oder streichen Sie die Signale.

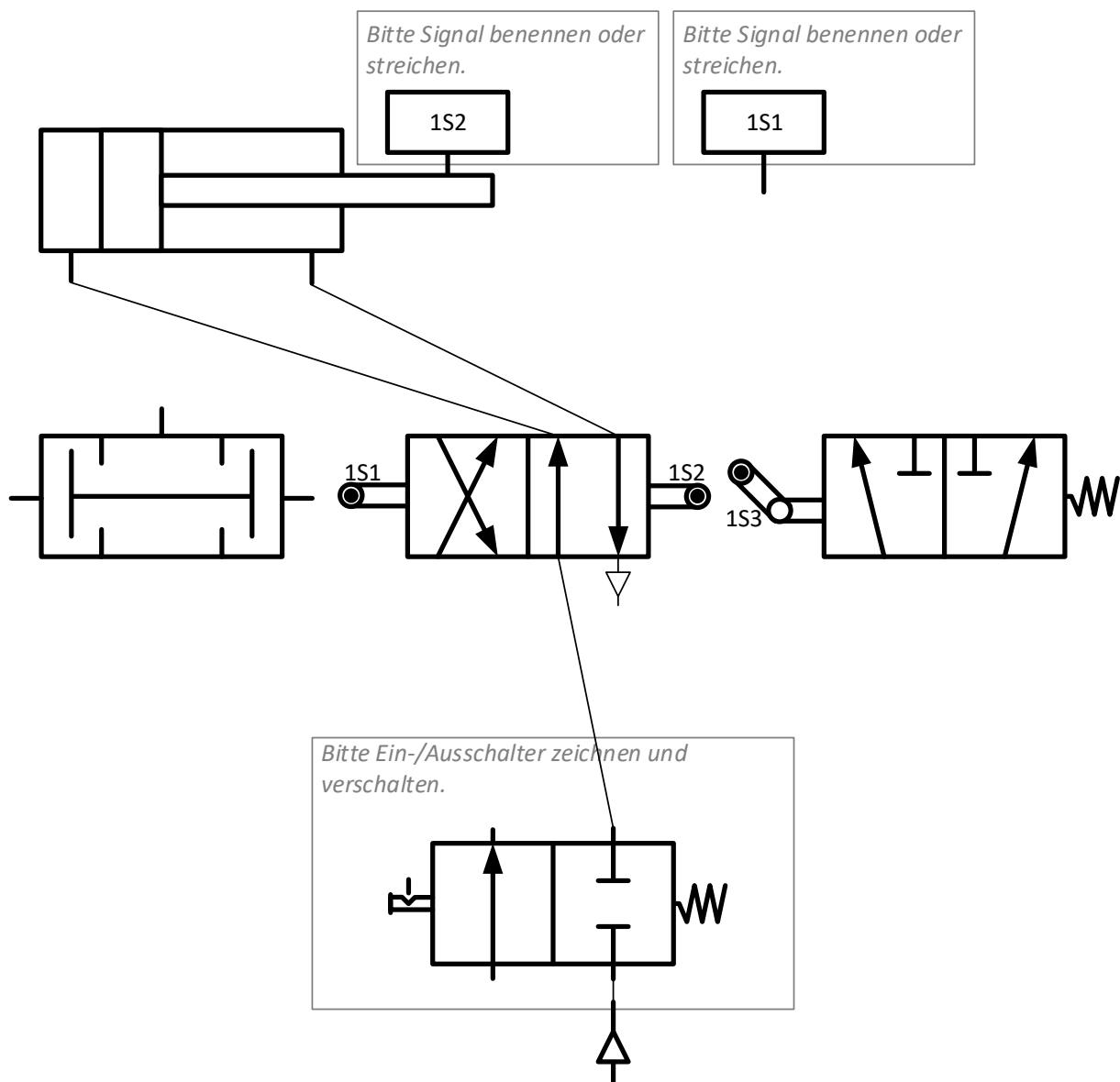


Abbildung 15: Pneumatische Steuerung

Anmerkung: Aufgabe 5.2 hatte mehrere Lösungen, die als korrekt gewertet wurden.

Aufgabe 6: Kurzfragen (24,5 Punkte)

Hinweis: Die Teilaufgaben in Aufgabe 6 können alle unabhängig voneinander bearbeitet werden.

Frage 6.1: CAN (3 Punkte)

Verfälscht Bit-Stuffing den Arbitrierungsvorgang, d.h. kommt es vor, dass aufgrund von Bit-Stuffing eine eigentlich niederpriore Nachricht Vorrang vor einer höherprioren bekommt? Begründen Sie Ihre Antwort.

Nein, da immer alle noch sendenden Knoten bzw. alle noch nicht abgebrochenen Nachrichten gleichermaßen betroffen wären.

Alternativ: Nein, da alle das Stuffing-Bit bekämen.

Frage 6.2: Amdahl (4 Punkte)

Gegeben ist ein Programm mit einem seriellen Anteil von 12,5 %. Bisher wurde es auf einem Ein-Kern-Prozessor ausgeführt und benötigte dafür 568 Sekunden.

a) Wie lange dauert die Ausführung des Programms bei Nutzung eines Sechs-Kern-Prozessors unter den Annahmen, dass

- (I) die Prozessoren sich nicht weiter unterscheiden,
- (II) beide Rechner während der Ausführung des Programms keinerlei andere Aufgaben bearbeiten müssen.

Raum für Nebenrechnungen:

Sequentieller Anteil: 71 s = 12,5 %

Parallelisierbarer Anteil: 568 s – 71 s = 497 s = 87,5 %

Rechnung nach Amdahl:

$$T_{6-Kern} = \left[\frac{71}{568} + \frac{497}{6} \right] \cdot 568 = 153,833 \text{ s}$$

Ausführungsdauer:

$T_{6-Kern} = 153,833 \text{ s}$

b) Was ist die maximale, prozentuale Zeitersparnis, die in dem genannten Szenario unter Verwendung eines hypothetischen Unendlich-Kern-Prozessors erreichbar wäre?

Raum für Nebenrechnungen:

Maximale, prozentuale Zeitersparnis = Parallelisierbarer Anteil

Maximale Zeitersparnis (prozentual):	0,875 oder 87,5 %
--------------------------------------	-------------------

c) Was ist die maximale, prozentuale Zeitersparnis, die in dem genannten Szenario unter Verwendung eines hypothetischen Ein-Kern-Prozessors und unter der Annahme, dass der serielle Anteil des Programms 0% beträgt, erreichbar wäre?

Maximale Zeitersparnis (prozentual):	0
--------------------------------------	---

Aufgabe 6.3: MQTT-Längenangabe (5,5 Punkte)

Hinweis: Diese Teilaufgabe kann unabhängig von den vorhergehenden bearbeitet werden.

Eine MQTT-PUBLISH-Nachricht mit einer Payload von 16.212 Byte und einem Topicname mit 13 Byte Länge soll mit QoS-L0 übertragen werden. Ermitteln Sie die MQTT-Längenangabe für diese Nachricht.

Wie lautet der Zahlenwert der zu übermittelnden Längenangabe (dezimal und binär)?

Raum für Nebenrechnungen:

Zu übermittelnder Zahlenwert (dezimal):	16.225
---	--------

Raum für Nebenrechnungen:

0 x 32.768	0 x 16.384	1 x 8.192	1 x 4.096	1 x 2.048	1 x 1.024	1 x 512	1 x 256
0 x 128	1 x 64	1 x 32	0 x 16	0 x 8	0 x 4	0 x 2	1 x 1

Zu übermittelnder Zahlenwert (binär):	11 1111 0110 0001
---------------------------------------	-------------------

Wie sieht die finale Längenangabe aus (binär und hexadezimal)?

Raum für Nebenrechnungen:

Codierter Binärwert bei 7 Informationsbits pro Byte: 111110 1100001

Codierter Binärwert bei 7 Informationsbits pro Byte

plus 1 Bit zur Signalisierung folgender Bytes: 01111110 11100001

Längenangabe (binär):

0111 1110 1110 0001

Raum für Nebenrechnungen:

0 = 0000 1 = 0001 2 = 0010 3 = 0011 4 = 0100 5 = 0101 6 = 0110 7 = 0111

8 = 1000 9 = 1001 A = 1010 B = 1011 C = 1100 D = 1101 E = 1110 F = 1111

Codierter Hexadezimalwert:

0x7E 0xE1

Ausgegebener Hexadezimalwert (Immer umgedreht, da niederwertigstes Byte zuerst codiert wird.):

0xE1 0x7E

Längenangabe
(hexadezimal):

0xE1 0x7E

Aufgabe 6.4: Single-Choice Fragen (12 Punkte)

Hinweis: Bitte kennzeichnen Sie Ihre Antworten eindeutig. Pro Frage ist genau eine Antwort korrekt. Pro korrekt beantworteter Frage werden 1,5 Punkte vergeben.

a) Welche der folgenden Aussagen ist wahr?

☒	Mikrocontroller enthalten üblicherweise Datenspeicher.
☐	Mikrocontroller und Mikroprozessoren sind üblicherweise ähnlich leistungsfähig.
☐	Mikroprozessoren haben üblicherweise keinen Speicher integriert.
☐	Mikroprozessoren werden vielfach auf Feldgerät-Ebene eingesetzt.

...