

Hochschule Esslingen

Sommersemester 2009	Blatt-Nr.	1 von 10
Fachbereich: Informationstechnik	Semester:	IT4
Prüfungsfach: Echtzeitsysteme	Fachnummer:	4061
Hilfsmittel: schriftliche Unterlagen, Taschenrechner	Zeit:	90 Minuten

Name, Vorname: _____

Aufgabe 1: Diverse Fragen (25 Minuten)

Hinweis: Die folgenden Teilaufgaben können unabhängig voneinander bearbeitet werden.

- 1.1 Erläutern Sie kurz den Begriff „präemptives Scheduling“ im Gegensatz zum „nicht-präemptiven Scheduling“.

Antwort:

- 1.2 Erläutern Sie kurz den Begriff „optimaler Scheduling-Algorithmus“.

Antwort:

Sommersemester 2009	Blatt-Nr.	2 von 10
Fakultät: Informationstechnik	Semester:	TI 7
Prüfungsfach: Echtzeitsysteme	Fachnummer:	4061
Hilfsmittel: schriftliche Unterlagen, Taschenrechner	Zeit:	90 Minuten

1.3 Erläutern Sie kurz den Unterschied zwischen einer Semaphore und einem Mutex.

Antwort:

1.4 Die Frequenz des RTI des im Labor verwendeten Mikrocontrollers lässt sich gemäß der Gleichung $f_{RTI} = f_{oscclk} / (2^{9+X} (Y+1))$ einstellen, wobei X drei Bit und Y 4 Bit groß ist. Damit sollen Sie eine Echtzeituhr mit einer Auflösung von ca. 0,1 Sekunden realisieren, die eine Abweichung von nie mehr als 1 Sekunde gegenüber einem globalen perfekten Referenzzeitgeber besitzt. Welche Werte verwenden Sie für X und Y, wenn der Oszillatortakt f_{oscclk} 4 MHz beträgt? Wie oft müssen Sie innerhalb von 24 Stunden die Uhr neu synchronisieren?

Antwort:

X = _____

Y = _____

Anzahl Resynchronisationen per 24 Stunden: _____

1.5 Im Echtzeit-Projekt haben Sie mit einem verteilten System mit zwischen drei und fünf Prozessoren gearbeitet. Nehmen Sie an, Sie lassen auf jedem Rechner des Komplettsystems mit fünf Prozessoren jeweils eine Echtzeituhr mit einer mittleren Gangabweichung von jeweils $25 \cdot 10^{-6}$ sec/sec laufen. Ausgehend von perfekt synchronisierten Uhren soll die Zeitdifferenz nie größer als 1 Sekunde sein.

Genügt es, wenn die Uhren einmal alle 24 Stunden resynchronisiert werden? Wie groß ist zu diesem Zeitpunkt die maximale Zeitdifferenz zwischen jeweils zwei Uhren?

Sommersemester 2009	Blatt-Nr.	3 von 10
Fakultät: Informationstechnik	Semester:	TI 7
Prüfungsfach: Echtzeitsysteme	Fachnummer:	4061
Hilfsmittel: schriftliche Unterlagen, Taschenrechner	Zeit:	90 Minuten

Antwort mit Rechenweg:

maximale Zeitdifferenz nach 24 Stunden: _____

- 1.6 Erläutern Sie die wesentlichen Unterschiede zwischen einem allgemeinen (General Purpose)-Betriebssystem und einem Echtzeitbetriebssystem.

Antwort:

Sommersemester 2009	Blatt-Nr.	4 von 10
Fakultät: Informationstechnik	Semester:	TI 7
Prüfungsfach: Echtzeitsysteme	Fachnummer:	4061
Hilfsmittel: schriftliche Unterlagen, Taschenrechner	Zeit:	90 Minuten

- 1.7 Schreibt die OSEK-Spezifikation für Semaphoren bzw. Ressourcen ein Ressource-Zugriffsprotokoll vor, und wenn ja, welches? Wie heißt der OSEK-Betriebssystemaufruf zum geschützten Belegen einer Ressource?

Antwort:

- 1.8 Welche Bedingungen sind erforderlich, damit es beim Zugriff auf Betriebsmittel zu einer Verklemmungssituation (Deadlock) kommt?

Antwort:

- 1.9 Kann mit Semaphoren ein gleichzeitiges Warten auf die Freigabe von mehr als einem Betriebsmittel realisiert werden? Bitte begründen.

Antwort:

Sommersemester 2009	Blatt-Nr.	5 von 10
Fakultät: Informationstechnik	Semester:	TI 7
Prüfungsfach: Echtzeitsysteme	Fachnummer:	4061
Hilfsmittel: schriftliche Unterlagen, Taschenrechner	Zeit:	90 Minuten

Aufgabe 2: Synchronisation (30 Minuten)

2.1 Betrachten Sie vier Tasks, die gemäß folgender Tabelle auf Ereignisse (Events) warten und dann die entsprechenden Aktivitäten ausführen, die jeweils *eine Zeiteinheit* für ihre Ausführung benötigen:

Task	Priorität	Event Flag Bits	Konj./disj.	Aktivität
τ_1	10	0 und 2	konjunktiv	A1
τ_2	20	3		A2
τ_3	30	4 und 5	disjunktiv	A3
τ_4	40	6 und 2	konjunktiv	A4

Der zugrunde liegende Scheduler ist präemptiv und arbeitet prioritätsorientiert.

Die Aktivitäten setzen die folgenden Eventflags ganz am Anfang (nach 0 Zeiteinheiten Laufzeit) oder ganz am Ende (mit 0 Zeiteinheiten Restlaufzeit) ihrer Laufzeit:

Aktivität	Zeitpunkt	Event Flag Bits
A1	Anfang	Bit 6
A2		
A3		
A4	Ende	Bit 4

Für die Modellierung betrachten Sie das Setzen *dieser* Eventflags als eine zusätzliche Aktivität mit null Laufzeit direkt vor oder direkt nach der Aktivität Ax.

Zeichnen Sie auf die folgende Seite das zu diesem Szenario gehörige Aktivitätsdiagramm unter der Annahme, dass alle Tasks vom Typ „run to completion“ sind.

Sommersemester 2009	Blatt-Nr.	6 von 10
Fakultät: Informationstechnik	Semester:	TI 7
Prüfungsfach: Echtzeitsysteme	Fachnummer:	4061
Hilfsmittel: schriftliche Unterlagen, Taschenrechner	Zeit:	90 Minuten

Lösung zu Aufgabe 2.1 (UML-Aktivitätsdiagramm):

Task τ_1

Task τ_2

Task τ_3

Task τ_4

Sommersemester 2009	Blatt-Nr.	7 von 10
Fakultät: Informationstechnik	Semester:	TI 7
Prüfungsfach: Echtzeitsysteme	Fachnummer:	4061
Hilfsmittel: schriftliche Unterlagen, Taschenrechner	Zeit:	90 Minuten

2.2 Zum Zeitpunkt 0 sind alle Eventflags zurückgesetzt. Es werden von außen nun zu den in der untenstehenden Tabelle dargestellten Zeitpunkten Eventflags gesetzt. Gehen Sie davon aus, dass alle einmal angelaufenen Aktionen zu Ende laufen können, bevor ein neues Ereignis auftritt.

Wenn eine Task ihren Zustand wechselt, tragen Sie den neuen Zustand sowie die evtl. ausgeführte Aktion in eine neue Zeile ein (a: ablaufbereit; w:wartend; l:laufend, r:ruhend):

Zeitpunkt	Setze Eventflag	Zustand τ_1	Zustand τ_2	Zustand τ_3	Zustand τ_4	Aktion
0		w	w	w	w	
1	0 und 3					
2	2					
3						
4						
5						
6						

2.3 Das unten dargestellte Szenario hat sich ergeben, wobei immer zu den angegebenen Zeitpunkten ein Eventflag gesetzt worden ist oder auch nicht. Rekonstruieren Sie *alle* möglichen Folgen von Eventflag-Setz-Ereignissen, die zu dem beobachteten Ablauf geführt haben können (a: ablaufbereit; w:wartend; l:laufend; r:ruhend):

Zeitpunkt	Gesetzte Eventflags	Zustand τ_1	Zustand τ_2	Zustand τ_3	Zustand τ_4
0		w	w	w	w
1				l	
2				r	l
3			l		r
4			r		
5		l			

Sommersemester 2009	Blatt-Nr.	8 von 10
Fakultät: Informationstechnik	Semester:	TI 7
Prüfungsfach: Echtzeitsysteme	Fachnummer:	4061
Hilfsmittel: schriftliche Unterlagen, Taschenrechner	Zeit:	90 Minuten

Aufgabe 3: Scheduling (35 Minuten)

Gegeben sind vier Tasks, die auf Ressourcen (A,B,C) zugreifen:

τ_1 ($r_0 = 10$, $C = 4$, $\text{Prio}=1$; [A;1]), wobei die Task zwei Zeiteinheiten läuft, bevor sie auf die Ressource A zugreift.

τ_2 ($r_0 = 7$, $C = 4$, $\text{Prio}=2$; [A;1][B;1]), wobei die Task eine Zeiteinheit läuft, bevor sie auf die Ressourcen A und dann B zugreift.

τ_3 ($r_0 = 4$, $C = 4$, $\text{Prio}=3$; [B;1][C;1]), wobei die Task eine Zeiteinheit läuft, bevor sie auf die Ressourcen B und dann C zugreift.

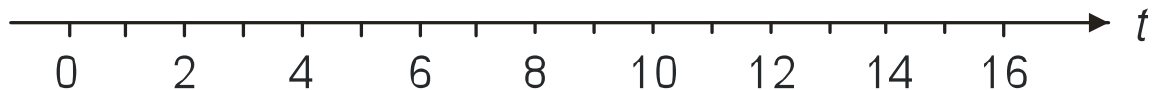
τ_4 ($r_0 = 4$, $C = 4$, $\text{Prio}=4$; [A;5[B;2]][C;1]), wobei die Task zwei Zeiteinheiten läuft, dann auf Ressource A zugreift, sie eine Zeiteinheit lang hält und dann einen verschachtelten Zugriff auf die Ressourcen B und danach C vornimmt.

Notation: r_0 ist die Release- oder Request-Zeit, C ist die Ausführungszeit, Prio ist die Priorität der Task (höhere Zahl gleich höhere Priorität). [A;5] bedeutet Zugriff auf Ressource 5 für fünf Zeiteinheiten. [A;a[B;b]] bedeutet ein verschachtelter Zugriff, wobei Benutzung der Ressource A die Benutzung von B beinhaltet, und die Zeit b in a enthalten ist.

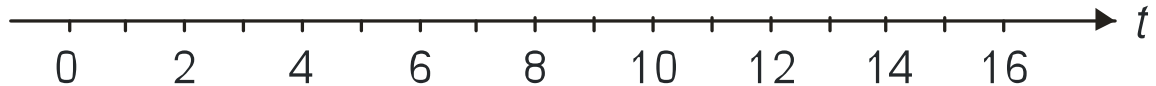
Das Task-Set soll mit zwei verschiedenen Ressource-Access-Protokollen geplant werden.

3.1 Zeichnen Sie das Zeitdiagramm für das obige Taskset unter Verwendung des **Priority Inheritance Protokolls** in das vorbereitete Diagramm. **Eigene Skizzen werden nicht akzeptiert!** Antwort:

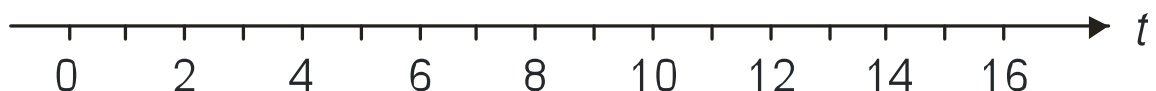
τ_1



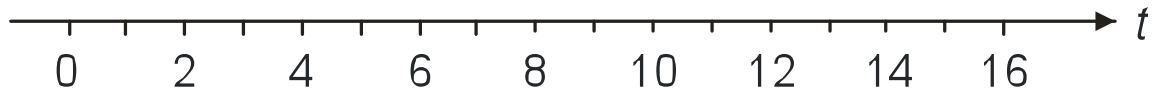
τ_2



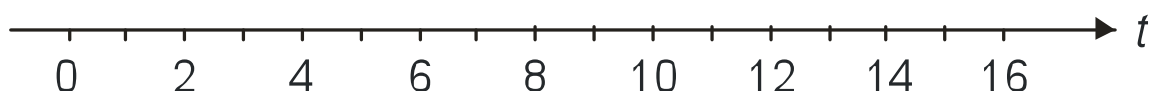
τ_3



τ_4

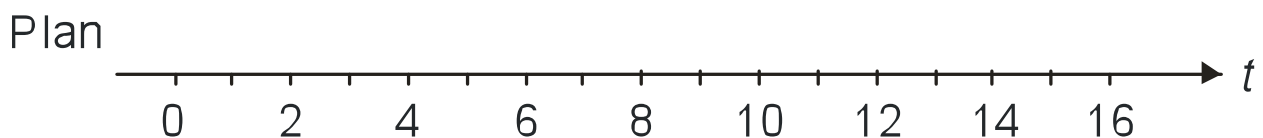
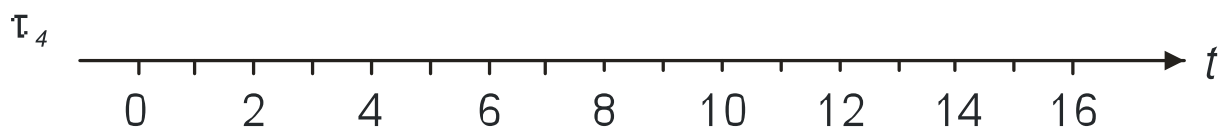
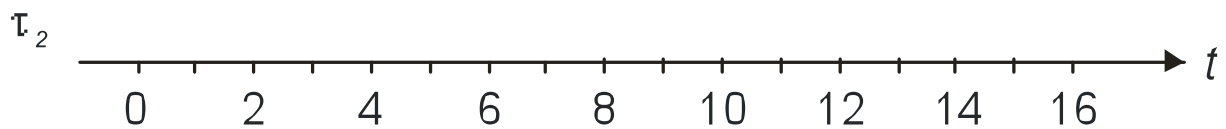
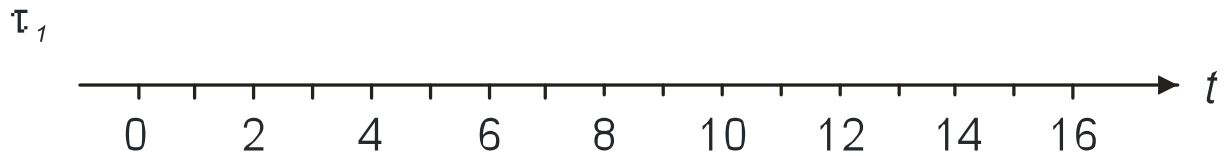


Plan



Sommersemester 2009	Blatt-Nr.	9 von 10
Fakultät: Informationstechnik	Semester:	TI 7
Prüfungsfach: Echtzeitsysteme	Fachnummer:	4061
Hilfsmittel: schriftliche Unterlagen, Taschenrechner	Zeit:	90 Minuten

3.2 Zeichnen Sie das Zeitdiagramm für das obige Taskset unter Verwendung des **Priority Ceiling Protokolls** in das vorbereitete Diagramm. **Eigene Skizzen werden nicht akzeptiert!** Antwort:



3.3 Betrachten Sie ein neues periodisches Taskset mit den folgenden Parametern:

τ_i	r_i	C_i	T_i
τ_1	0	4	36
τ_2	0	3	8
τ_3	0	2	24
τ_4	0	8	39
τ_5	0	1	27
τ_6	0	1	25

Ist es möglich, dieses Taskset mit dem Rate-Monotonic-Verfahren unter Einhaltung aller Fristen einzuplanen? Bitte begründen!

Sommersemester 2009	Blatt-Nr.	10 von 10
Fakultät: Informationstechnik	Semester:	TI 7
Prüfungsfach: Echtzeitsysteme	Fachnummer:	4061
Hilfsmittel: schriftliche Unterlagen, Taschenrechner	Zeit:	90 Minuten

Antwort:

3.4 Zeichnen Sie den Plan (Markierungen mit X) für das o.g. Taskset und Schedulingverfahren in die unten vorbereitete Tabelle (mit Umbruch, zweite Tabelle ist Fortsetzung der ersten):

τ_1																				
τ_2																				
τ_3																				
τ_4																				
τ_5																				
τ_6																				
	0	2	4	6	8	10	12	14	16	18	t									

τ_1																				
τ_2																				
τ_3																				
τ_4																				
τ_5																				
τ_6																				
	20	22	24	26	28	30	32	34	36	38	t									

3.5 Für das Taskset aus Aufgabe 3.3 soll nun noch die Kontextwechselzeit mit berücksichtigt werden, diese betrage 0,8 Zeiteinheiten, auch für den Start des ersten Jobs.

Ist es möglich, das Taskset unter dieser Bedingung einzuplanen? Bitte begründen.

Antwort: