

<p style="text-align: center;">Ostfalia Hochschule für angewandte Wissenschaften</p> <p>Fakultät Fahrzeugtechnik Prof. Dr.-Ing. V. von Holt Institut für Fahrzeugsystem- und Servicetechnologien</p>	<p>Modulprüfung Embedded Systems BPO 2011</p> <p style="text-align: right;">WS 2016/17 06.01.2017</p>	<p>Name:.....</p> <p>Vorname.....</p> <p>Matr.Nr.:.....</p> <p>Unterschrift.....</p>
---	---	--

Zugelassene Hilfsmittel: **Einfacher Taschenrechner**
Zeit: 60 Minuten

1 (8)	2 (38)	3 (14)	Summe (60)	Note

Aufgabe 1 (8 Punkte) – Kurzfragen

- a) (2 P) Was unterscheidet das Versetzen einer Task in den **Wait-Zustand** von dem Aufruf einer „klassischen“ **Delay-/Wait-Funktion**?
- b) (2 P) Warum darf in Multitaskingumgebungen eine Task **nicht ständig aktiv** sein?
- c) (2 P) Wann bezeichnet man ein **Schedulingverfahren** als „optimal“?
- d) (2 P) Was versteht man unter einer „**Task**“ und was unter dem „**Taskkontext**“?

Aufgabe 2 (38 Punkte) – Scheduling

Zur Steuerung eines Echtzeitsystems ist ein Taskset bestehend aus 4 Tasks (T1, T2, T3, T4) entworfen worden.

Die folgende Tabelle enthält die Zykluszeiten sowie die Laufzeiten der einzelnen Tasks.

Tasks	Zykluszeit [ms]	Laufzeit[ms]
T1	4...6	1
T2	8...10	1...2
T3	10...20	2
T4	15	3

- a) (8 P) Welche Annahmen nach o.a. Tabelle charakterisieren den **Worst-Case**, den **Best-Case** bzw. den **gemittelten** Wert der **Prozessorlast** durch die **einzelnen Tasks** sowie durch das **gesamte Taskset**?

Tasks	Minimal [%]	Maximal [%]	Gemittelt [%]
T1			
T2			
T3			
T4			
Gesamt			

- b) (1 P) Ist das gegebene Taskset nach den Berechnungen unter a) prinzipiell realisierbar?

- c) (5 P) Das Taskset soll durch ein **Rate-Monotonic-Scheduling** realisiert werden soll. Welche **Prioritäten** müssen den **Tasks** jeweils zugewiesen werden? (**Höchste Priorität : 0**)
Nach welcher **Regel** werden die **Prioritäten vergeben**?
Ist das **Taskset** mit RMS-Scheduling **umsetzbar** (Begründung)?

Das Taskset soll im Folgenden durch ein Least-Laxity-First-Scheduling realisiert werden. Hierbei sollen – abweichend zu oben – die folgenden Annahmen über die Zyklus- und Laufzeiten der Tasks gelten:

Tasks	Zykluszeit [ms]	Laufzeit[ms]
T1	5	1
T2	10	2
T3	10	3
T4	15	4

- d) (16 P) Geben Sie in **Tabelle 1** für den **Zeitraum 0...30ms** für alle Tasks die jeweilige **Restlaufzeit** sowie den **Spielraum** (laxity) an! Hierbei soll vereinfachend angenommen werden, dass das Betriebssystem **nur alle Millisekunde einen Taskwechsel** durchführt. Markieren Sie in jedem Zeitschritt durch **Einkreisen der Zeiten**, welche **Tasks** im jeweiligen Zeitschritt **abläuft**!
- e) (8 P) Stellen Sie den **Verlauf des Spielraums** (laxity) der Tasks in **Diagramm 1** für den **Zeitraum 0...15ms** grafisch dar! Geben Sie **farblich** oder durch **Annotationen** an, welcher **Tasks** die jeweiligen **Kurven entsprechen**!

Aufgabe 3 (14 Punkte) – Flusskontrolle

3 Producer-Tasks sollen **1 Consumer-Tasks** Daten zur Weiterverarbeitung übermitteln. Damit die Consumer-Tasks nicht mit Daten überflutet wird bzw. keine Daten durch Pufferüberläufe verlorengehen, soll zwischen den Producer-Tasks und der Consumer-Tasks eine **Flusskontrolle** implementiert werden. Die maximale Anzahl der zu puffernden Datensätze soll **N** betragen.

Anmerkung: Das Betriebssystem stellt Ihnen folgende Kommunikations-/Synchronisationskonstrukte zur Verfügung: **Message Queue, Mailbox, Semaphore, Mutex, Shared Memory**

- a) (6 P) Stellen Sie grafisch eine mögliche Lösung für eine Flusskontrolle dar.

- b) (8 P) Erläutern Sie die Funktionsweise Ihrer gewählten Anordnung anhand von Pseudocode oder eines Aktivitätsdiagramms!