

# Entwurf von Echtzeitsystemen

Wintersemester 2014

## 5. Übungsblatt

Übungsdatum: 05.12.2014

Gruppe: 1

Ausarbeitung von: Martin Winter

Wildon, am 4. Dezember 2014

# Task-Ausführung

Beschäftigen Sie sich mit der Ausführung von Tasks durch ein Echtzeitbetriebssystem. Nennen sie die Unterschiede zwischen synchroner und asynchroner Taskausführung. Überlegen Sie sich zu beiden Arten jeweils mindestens 2 Beispiele anhand derer sie die jeweiligen Funktionsweise genau erklären. Gehen sie in den Beispielen ebenfalls auf Minor und Major Frames ein, sowie auf die Zeitbedingungen.

# Schedulingverfahren

Klassifizieren Sie die Scheduling-Algorithmen **EDF**, **EDD** (Earliest Deadline Due), **RMS**, **DMS** (Deadline Monotonic Scheduling), **FCFS** und **RR** (RoundRobin) bezüglich folgender Kriterien:

- hart-echtzeitfähig vs zeit-unkritisch
- periodisch vs aperiodisch vs hybrid
- preemptive vs non-preemptive
- statisch vs dynamisch
- on-line vs off-line

Welche Kombination von Scheduling-Algorithmen würden Sie verwenden, um hart-echtzeitfähige periodische Tasks zusammen mit sporadischen (aperiodischen) auszuführenden, aber zeitunkritischen Tasks in sogenannten offenen Systemen zu planen?

**EDF** (Earliest Deadline first)

Ist ein **dynamisches** Schedulingverfahren, bei jedem Schedulingevent wird in der List der Task mit nächsten Deadline gesucht und dieser dann ausgeführt. Es ist eher **hart-echtzeitfähig**, aber es ist nicht leicht nachvollziehbar, ob Deadlines eingehalten werden oder nicht. Es handelt sich um ein **preemptives** Verfahren, welches mit periodischen, als auch aperiodischen Task gut klarkommt, also **hybrid**.

**EDD** (Earliest Deadline due) Ist ein **statisches** Schedulingverfahren, hierbei wird ähnlich wie bei EDF gearbeitet, nur dass keine dynamischen Änderungen möglich sind. Es ist also ebenso **hart-echtzeitfähig**, **preemptive** und **hybrid**.

**RMS** (Rate-Monotonic Scheduling)

Hier werden die Prioritäten zur Design-Zeit festgelegt nach zunehmenden Taskperioden, je kürzer die Periodendauer eines Prozesses, desto höher ist seine Priorität, es ist also **statisch**. Hierbei wird immer der Prozess mit der höchsten Priorität gescheduled, es ist also **hart-echtzeitfähig**, **preemptive** und kann sowohl mit periodischen, als auch aperiodischen Tasks umgehen, ist also **hybrid**.

**DMS** (Deadline Monotonic Scheduling)

Dies ist ein Verfahren, das **hart-echtzeitfähig** ist, es dient zur Verwaltung von Prozessen mit fester Priorität. Prozesse können hierbei zu jeder Zeit unterbrochen werden, es ist also **preemptive**. Es wird, wie bei RMS stets der Prozess der höchsten Priorität ausgeführt. Aperiodische Prozesse können auch eingebaut werden, indem man diese als fiktive periodische Prozesse integriert, das Verfahren ist also **hybrid**. Es vergibt dabei fixe Prioritäten, ist also **statisch**.

**FCFS** (FIFO)

Bei FIFO werden Prozesse einfach so angeordnet, wie sie in der Ready-Schlange ankommen, es ist ein Verfahren mit **statischen bzw. keinen** Prioritäten, wer zuerst bereit ist, wird gescheduled, es ist auch **non-preemptive**, Tasks laufen so lange, bis sie fertig sind,

kann somit zu Starvation führen. Dadurch ist auch das ganze Verfahren **dynamisch** und nur **zeitunkritisch**, da auch die Prozesse in jeder Reihenfolge dran kommen können, ist es **aperiodisch**.

**RR** (RoundRobin)

Jeder Prozess bekommt einen fixen Zeitslot und so wird durch rotiert zwischen den Prozessen, es handelt sich somit um ein **statisches Verfahren**. Ist ein Schedulingverfahren mit **statischen bzw. keinen** Prioritäten, Tasks werden immer im Kreislauf gescheduled, es handelt sich also um ein **periodisches** Verfahren, welches **preemptive** ist, Task werden nach einer gewissen Zeit unterbrochen und der nächste Task kommt dran. Dadurch lässt sich natürlich keine harte Echtzeitfähigkeit garantieren, es handelt sich also um ein **zeitunkritisches** Verfahren. Hierbei kann es aber zu keiner Starvation kommen, da keine Priorität vergeben wird.

Für dieses offene System würde sich entweder RMS oder DMS anbieten, da beide sehr gut mit hart echtzeitfähigen, periodischen Prozessen umgehen können, aber auch aperiodische gut integrieren können.

## Anwendungsbeispiel Scheduling-Verfahren

Bearbeiten Sie für die unten angegebenen Beispiele zur Prioritätsvergabe folgende Aufgaben mittels Rate-Monotonic Scheduling (RMS)

- Zeigen Sie, unter Zuhilfenahme des in der VO gezeigten Satzes, ob ein RM-Schedule sicher existiert oder nur möglicherweise.
- Zeigen Sie wie ein mögliches RM-Scheduling aussehen könnte, falls kein RM Schedule möglich ist, zeigen sie, wo es zu Problemen kommt.
- Versuchen Sie Problemfälle unter Zuhilfenahme eines anderen, in der VO gezeigten, dynamischen Schedulingverfahrens zu lösen.

Task	Bsp	1	Bsp	2	Bsp	3
1	C	T	C	T	C	T
A	1	5	1	4	1	4
B	1	7	1	7	1	5
C	2	10	2	8	2	8
D	2	15	2	13	2	10
E	1	16	1	14	1	12

## Anwendung von Scheduling

Gegeben seien folgende Tasks, deren Periode und Ausführungszeit

- Task 1 = (3;6)
- Task 2 = (2;9)
- Task 3 = (k, 12)

Wählen sie k so, dass die Tasks mittels EDF und RMS

- auf keinen Fall schedulbar sind
- eventuell schedulbar sind
- auf jeden Fall schedulbar sind

Analysieren Sie alle 3 Fälle für beide Algorithmen und begründen sie jeweils die Wahl von k.