

Script generated by TTT

Title: Grundlagen_Betriebssysteme (25.11.2015)

Date: Wed Nov 25 13:17:18 CET 2015

Duration: 43:00 min

Pages: 14

Eine wesentliche Aufgabe der Prozessorverwaltung besteht darin zu entscheiden, welcher der um den bzw. die Prozessor(en) konkurrierenden Prozesse (bzw. Threads) zu einem Zeitpunkt an den bzw. die Prozessor(en) gebunden wird. Dazu steht die BS-Komponente **Scheduler** zur Verfügung.

Prozessablauf besteht aus einer Sequenz von alternierenden CPU- und E/A-Perioden.

Zeitliche Verschränkung der Prozessbearbeitung bei einer CPU.

Unterscheidung zwischen Prozess-Scheduling und Thread-Scheduling.

[Kriterien](#)
[Scheduling-Strategien](#)
[Beispiel Unix Scheduling](#)
[Thread Scheduling](#)
[Mehrschichtiges Scheduling](#)
[Echtzeit Scheduling](#)

Generated by Targeteam



Echtzeit Scheduling



Prozesse in Multimedia-Anwendungen führen oft zeitkritische Operationen aus. Bzgl. des Scheduling existieren zwei gegensätzliche Ziele:

- ein **unkritischer Prozess** sollte nicht dauerhaft blockiert werden, weil zeitkritische Prozesse eine Ressource gänzlich auslasten.
- zeitkritische Prozesse** dürfen nicht durch Scheduling-Verfahren (Round-Robin oder Prioritäten) am zeitkritischen Fortschritt gehindert werden \Rightarrow Einhaltung von Zeitvorgaben.

Zuordnung von Kenngrößen zu zeitkritischen Prozessen

Bereitzeit (ready time): frühestmöglicher Ausführungsbeginn einer Aktivität.

Frist (deadline): spätestester Zeitpunkt für die Beendigung einer Aktivität.

Ausführungszeit: worst-case Abschätzung für das zur vollständigen Ausführung einer Aktivität notwendige Zeitintervall.

[Earliest Deadline First \(EDF\)](#)

[Rate-Monotonic Scheduling](#)

Generated by Targeteam



Earliest Deadline First (EDF)



Der Prozessor wird immer dem Prozess mit der am nächsten in der Zukunft liegenden Frist zugeordnet. Es existieren die beiden Varianten: nicht-unterbrechend und unterbrechend.

nicht-unterbrechend

Eine Prozesszuordnung bleibt bis der Prozess eine blockierende Systemfunktion aufruft oder freiwillig die CPU abgibt. Neue eintreffende Prozesse mit kürzeren Fristen werden erst beim nächsten Scheduling berücksichtigt.

unterbrechend

Diese Variante führt einen Kontextwechsel durch, wenn ein Prozess mit einer kürzeren Frist rechenwillig wird.

Generated by Targeteam





Earliest Deadline First (EDF)



Der Prozessor wird immer dem Prozess mit der am nächsten in der Zukunft liegenden Frist zugeordnet. Es existieren die beiden Varianten: nicht-unterbrechend und unterbrechend.

nicht-unterbrechend

Eine Prozessorzuordnung bleibt bis der Prozess eine blockierende Systemfunktion aufruft oder freiwillig die CPU abgibt. Neue eintreffende Prozesse mit kürzeren Fristen werden erst beim nächsten Scheduling berücksichtigt.

unterbrechend

Diese Variante führt einen Kontextwechsel durch, wenn ein Prozess mit einer kürzeren Frist rechenwillig wird.

Generated by Targeteam



Rate-Monotonic Scheduling



Rate-Monotonic Scheduling (RMS) ist für periodische Prozesse; RMS ordnet Prioritäten in Abhängigkeit von der Periode zu.

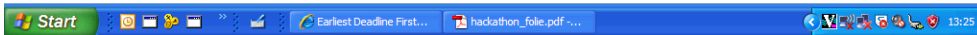
- 1) Prozesse mit der höchsten Frequenz (kleinste Periode) erhalten die höchste Priorität.
- 2) Prozesse mit der geringsten Frequenz (längste Periode) erhalten die niedrigste Priorität.

Auswahl der Prozesse anhand ihrer Priorität.

hochfrequente Prozesse werden minimal verzögert.

Zerstückelung niederfrequenter Prozesse, da sie häufig wegen hochfrequenter Prozesse unterbrochen werden.

Generated by Targeteam



Prozess- und Prozessorverwaltung



Ein Prozess ist der Ablauf eines Programms in einem Rechner. Dieser Ablauf ist eine Verwaltungseinheit im jeweiligen Betriebssystem.

Fragestellungen

Dieser Abschnitt gibt eine kurze Einführung in eine der wichtigen Verwaltungsaufgaben eines Betriebssystems:

Verwaltung von Prozessen.

Verwaltung des Prozessors, d.h. Zuteilung der CPU an rechenbereite Prozesse (Scheduling).

Unterbrechungskonzept.

[Prozessverwaltung](#)

[Prozessorverwaltung](#)

[Unterbrechungskonzept](#)

Generated by Targeteam



Unterbrechungskonzept



Die optimale Ausnutzung und Auslastung aller Geräte eines Rechner legt Mehrprogrammbetrieb nahe. Zerlegung der Ausführungsphasen eines Programms in viele Einzelteile;

- Aufbrechen der Ausführung eines Prozesses in mehrere Phasen: u.a. Rechnen, E/A, Synchronisieren mit Partner.
- Die Ausführung der Programme wird in der Regel mehrfach unterbrochen.

[Motivation](#)

[Unterbrechungsarten](#)

[Behandlung externer Unterbrechungen](#)

[Konflikte](#)

Generated by Targeteam



Motivation



Unterbrechungskonzept



Ursachen für Unterbrechungen

- zugeteilte Prozessorzeit ist aufgebraucht;
- benötigte Ressourcen stehen aktuell nicht zur Verfügung;
- ein E/A-Gerät meldet sich zurück;
- ein Fehler tritt auf, z.B. Division durch 0;
- Systemaufruf (wurde bereits als spezielle Unterbrechung eingeführt);

Bei einer Unterbrechung wird ein gerade aktiver Prozess unterbrochen und eine **Unterbrechungsbehandlung** durchgeführt.

Es ist erforderlich, bei der Unterbrechung den CPU Status des gerade aktiven Prozesses für die spätere Fortsetzung zu speichern.

Forderung: Eine Unterbrechung muss so **kontrolliert** erfolgen, dass ein definierter Prozessstatus festgehalten werden kann.

Generated by Targeteam

Die optimale Ausnutzung und Auslastung aller Geräte eines Rechensystems legt Mehrprogrammbetrieb nahe. Zerlegung der Ausführungsphasen eines Programms in viele Einzelteile;

- Aufbrechen der Ausführung eines Prozesses in mehrere Phasen: u.a. Rechnen, E/A, Synchronisieren mit Partner.
- Die Ausführung der Programme wird in der Regel mehrfach unterbrochen.

Motivation

Unterbrechungsarten

Behandlung externer Unterbrechungen

Konflikte

Generated by Targeteam



Interne, synchrone Unterbrechungen

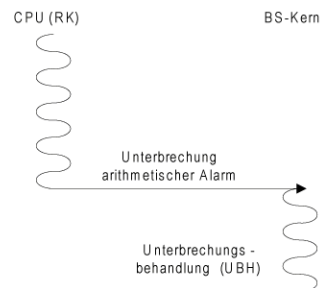


Behandlung externer Unterbrechungen



Dies sind Unterbrechungen (Alarmer, Exceptions), die durch den zu unterbrechenden Prozess selbst ausgelöst werden, z.B. Division durch 0.

Der Ablauf im RK wird unterbrochen und eine Unterbrechungsanfangsbehandlung des Systemkerns wird aktiv.



Beispiele von internen Unterbrechungen

Speicherschutzalarm: Prozess greift auf Speicherbereich zu, der nicht vorhanden ist oder auf den er nicht zugreifen darf.

Befehlsalarm: dem Operationscode des Maschinenbefehls ist keine Operation zugeordnet.

Seitenfehltalarm: Seite der virtuellen Adresse ist nicht im Arbeitsspeicher.

arithm. Alarm: arithm. Operation kann nicht ausgeführt werden, z.B. Division durch 0.

Systemaufruf: kontrollierter Übergang in das Betriebssystem.

Generated by Targeteam

Ablauf

Geräte-Controller meldet Unterbrechung über spezielle Interrupt-Leitung an CPU.

CPU prüft im Befehlszyklus nach jeder Befehlsausführung, ob eine Unterbrechung gemeldet wurde.

Falls Unterbrechung vorliegt: sichern u.a. des aktuellen Befehlszählers, des Programmstatusworts und Sprung zu einer Unterbrechungsanfangsbehandlung, die an festgelegter Speicheradresse steht.

Routine untersucht Unterbrechungsursache, die vom Controller über Datenleitung gemeldet wird (Unterbrechungsnummer).

über Unterbrechungsnummer erfolgt die Auswahl der benötigten Unterbrechungsbehandlungsroutine; Nummer ist i.a. Index in eine Tabelle, dem **Unterbrechungsvektor**.

Vektor enthält Speicheradresse der Unterbrechungsbehandlungsroutine.

Generated by Targeteam



Die optimale Ausnutzung und Auslastung aller Geräte eines Rechensystems legt Mehrprogrammbetrieb nahe. Zerlegung der Ausführungsphasen eines Programms in viele Einzelteile;

- Aufbrechen der Ausführung eines Prozesses in mehrere Phasen: u.a. Rechnen, E/A, Synchronisieren mit Partner.
- Die Ausführung der Programme wird in der Regel mehrfach unterbrochen.

[Motivation](#)

[Unterbrechungsarten](#)

[Behandlung externer Unterbrechungen](#)

[Konflikte](#)

Generated by Targeteam



Andere Unterbrechungen nicht zulassen, d.h. **Maskierung** von Unterbrechungen; anstehende Unterbrechung ignorieren oder vorläufig zurückstellen; Problem: u.a. Rechtzeitigkeit der Unterbrechungsbehandlung.

Interne Unterbrechungen erfolgen stets sofort und geben der zugehörigen Unterbrechungsbehandlung dieselbe **Priorität**, wie sie der unterbrochene Ablauf hatte.

Externe Unterbrechungen erhalten **Prioritäten** z.B. (0,...,31) zugeordnet. Die aufgerufene Unterbrechungsbehandlung erhält die **Priorität** (Ablaufpriorität) der auslösenden Unterbrechung.

Behandlung neu eintreffender externer Unterbrechung, falls **Priorität höher als Ablaufpriorität**; andernfalls zurückstellen.

Generated by Targeteam



- Prof. J. Schlichter
 - Lehrstuhl für Angewandte Informatik / Kooperative Systeme, Fakultät für Informatik, TU München
 - Boltzmannstr. 3, 85748 Garching
 - Email: schlichter@in.tum.de
 - Tel.: 089-289 18654
 - URL: <http://www11.informatik.tu-muenchen.de/>

[Übersicht](#)

[Einführung](#)

[Parallele Systeme - Modellierung, Strukturen](#)

[Prozess- und Prozessorverwaltung](#)

[Speicherverwaltung](#)

[Prozesskommunikation](#)

[Dateisysteme](#)

[Ein-/Ausgabe](#)

[Sicherheit in Rechensystemen](#)

[Entwurf von Betriebssystemen](#)

[Zusammenfassung](#)

Generated by Targeteam