

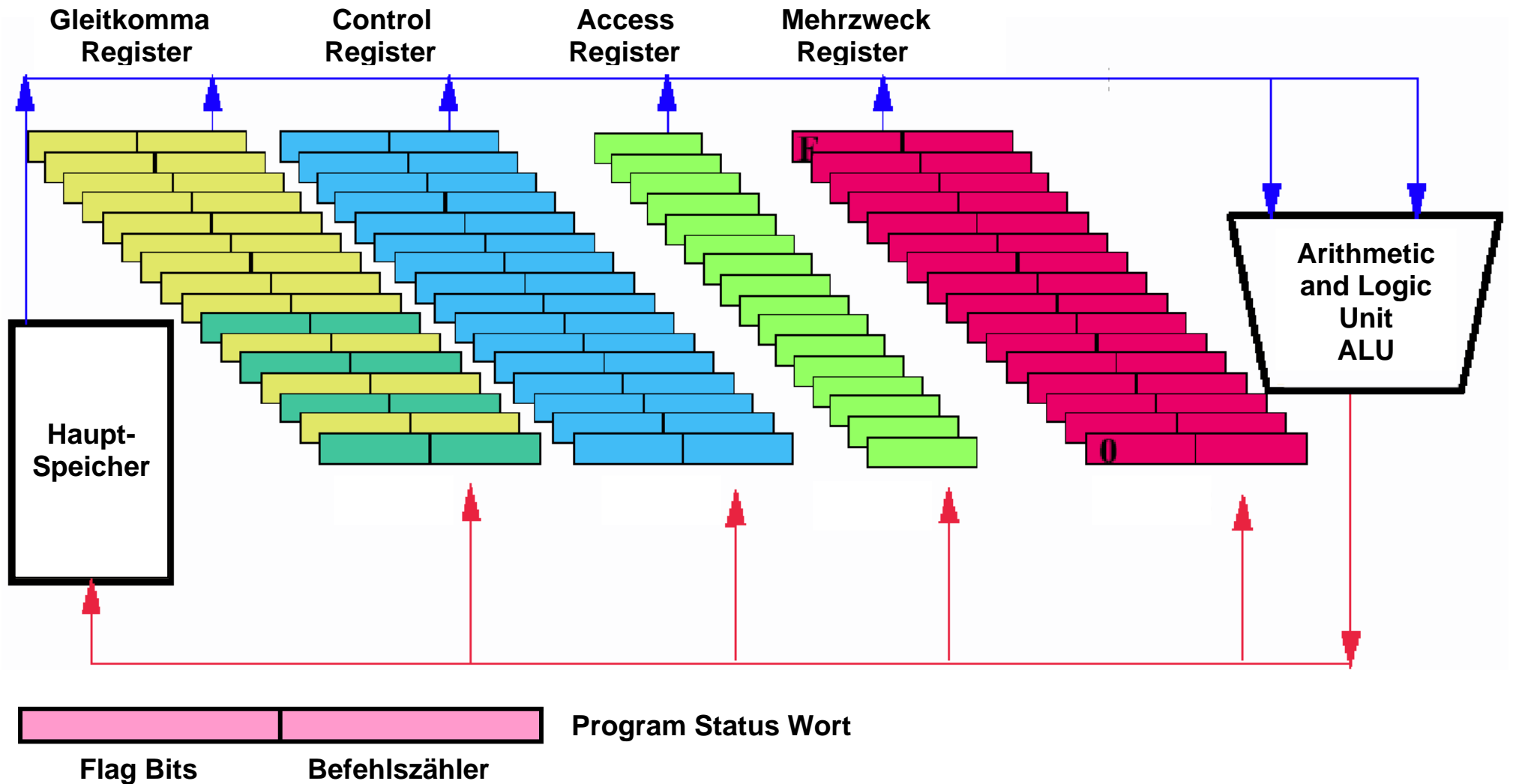
**Enterprise Computing
Einführung in das Betriebssystem z/OS**

**Prof. Dr. Martin Bogdan
Prof. Dr.-Ing. Wilhelm G. Spruth**

WS2012/13

Verarbeitungsgrundlagen Teil 3

Betriebssystem Überwacher



System z Programmiermodell

System z Programmiermodell

Die obige Abbildung zeigt die Bestandteile einer Mainframe CPU, die für den Programmierer sichtbar sind und die von Maschinenbefehlen manipuliert werden können. Man bezeichnet diese Teile auch als das Programmiermodell. Zu beachten ist, dass die CPU außerdem viele Teile und Funktionen enthält, die für einen Programmierer nicht zugänglich und unsichtbar sind.

Das Programmiermodell besteht aus dem Hauptspeicher, einer Verarbeitungseinheit (ALU) die Funktionen wie z.B. Addition ausführen kann und einer ganzen Reihe von Registern, die Daten, Hauptspeicheradressen oder Steuerfunktionen speichern können. Der Bestand an System z Registern umfasst:

- 16 Mehrzweck Register (General Purpose Register, GPR, 64 Bit) speichern Adressen oder Daten.
- 16 Gleitkommaregister (64 Bit) speichern Zahlen im Gleitkommaformat,
- 16 Control Register (64 Bit) speichern Steuerfunktionen. Ein Beispiel ist Steuerregister 1, welches die Anfangsadresse der Seitentafel im Hauptspeicher enthält,
- 16 Access Register (32 Bit) werden für spezielle Hauptspeicher Zugriffsfunktionen benutzt,

Zusätzlich existiert ein 64 Bit Befehlszähler Register und ein Flag Bit Register. Der Befehlszähler enthält die Adresse des nächstens auszuführenden Maschinenbefehls. Das Flag Register enthält individuelle Bits. Ein Beispiel ist ein Bit, welches den Rechner entweder in den Benutzerstatus oder in den Überwacherstatus versetzt.

Als eine Besonderheit der System z Architektur werden Befehlszähler und Flag Bits zu einem einzigen 128 Bit langen Register, dem „Programm Status Wort“ (PSW) Register zusammengefasst. Vom Standpunkt der Ausführung eines Programms definiert der Inhalt des PSW in jedem Augenblick den Status der CPU.

Bei einem CPU Chip mit mehreren Cores enthält jedes Core einen eigenen Satz der hier gezeigten Register.

Überwacherstatus Problemstatus

Eine CPU läuft in jedem Augenblick entweder im Überwacher Status (supervisor State) oder im Benutzer Status (user State).

Der Überwacher Status bzw. Benutzer Status wird durch 1 Bit im Flag Bit Register Teil des Programm Status Wortes der Zentraleinheit definiert.

Der Überwacher (Kernel) läuft im Überwacherstatus (Supervisor State, Kernel State).

Benutzerprogramme (Anwendungsprogramme) laufen im Problemstatus (Problem State, User State).

Auswirkungen sind:

- Bestimmte „**Privilegierte**“ Maschinenbefehle können nur im Überwacherstatus ausgeführt werden
- **Speicherschutz**. Im Problemstatus kann nur auf einen Teil des virtuellen und des realen Hauptspeichers zugegriffen werden

Unterbrechungen

Unterbrechungen (Interruptions) sind ein zentrales Steuerelement in einem jeden Rechner. Unterbrechungen bewirken eine Änderung des Status der Zentraleinheit als Folge von Bedingungen (Ursachen), die entweder

- **außerhalb der Zentraleinheit (CPU), oder**
- **innerhalb der Zentraleinheit**

auftreten.

Unterbrechungen bewirken den Aufruf und die Ausführung von speziellen Programmen (Unterbrechungsrouinen) außerhalb des normalen Programmablaufs.

Eine Unterbrechung bewirkt:

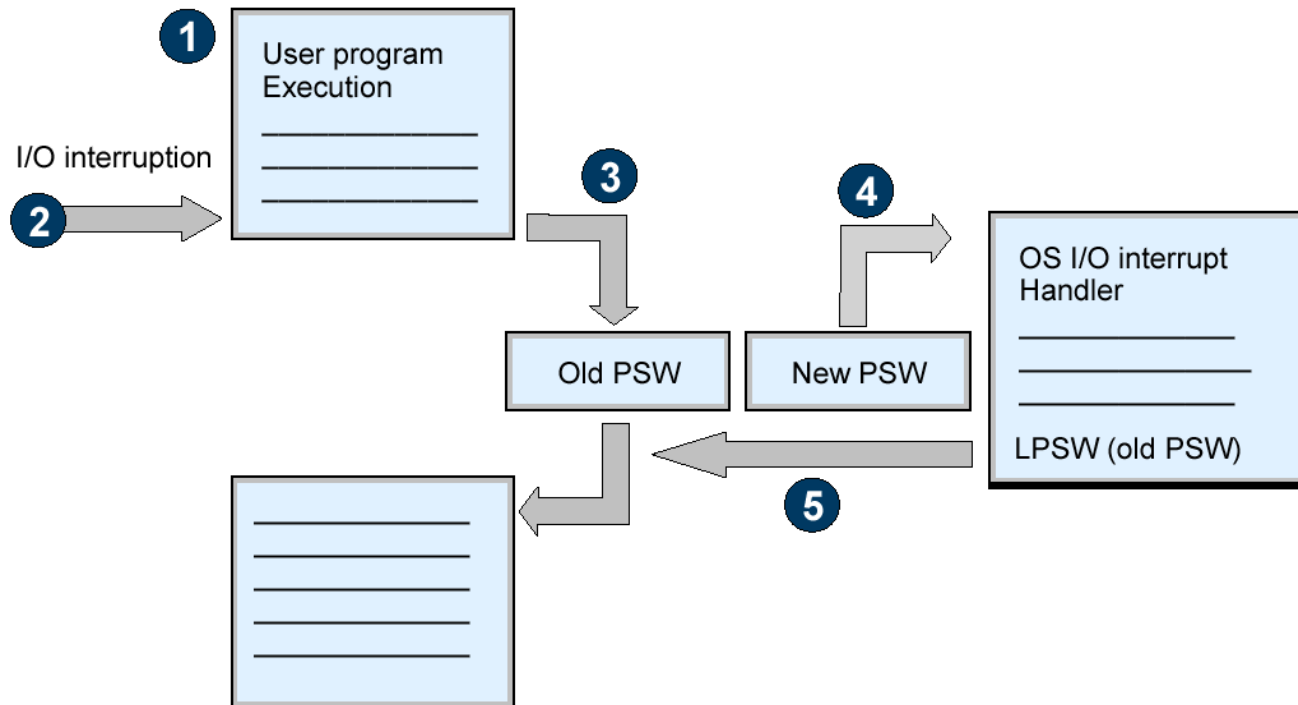
1. **Abspeichern des PSW (Programm-Status-Wort, Flag Bits und Befehlszählers) im Hauptspeicher.**
2. **Laden des Befehlszählers mit der Anfangsadresse einer Unterbrechungsroutine (Interrupt Handler).**
3. **Status-Initialisierung im PSW (z.B. Überwacherstatus setzen).**
4. **In der Regel: Abspeichern der Mehrzweckregister, evtl. auch Steuerregister, Gleitkommaregister durch die Unterbrechungsroutine.**

Typischerweise enthält ein Feld im Hauptspeicher zusätzliche Informationen über die Ursache der Unterbrechung.

Unterbrechungsklassen

Unterschiedliche Bedingungen können unterschiedliche Klassen von Unterbrechungen auslösen. Es existieren sechs Unterbrechungsklassen mit mehreren Unterbrechungsarten / Klasse

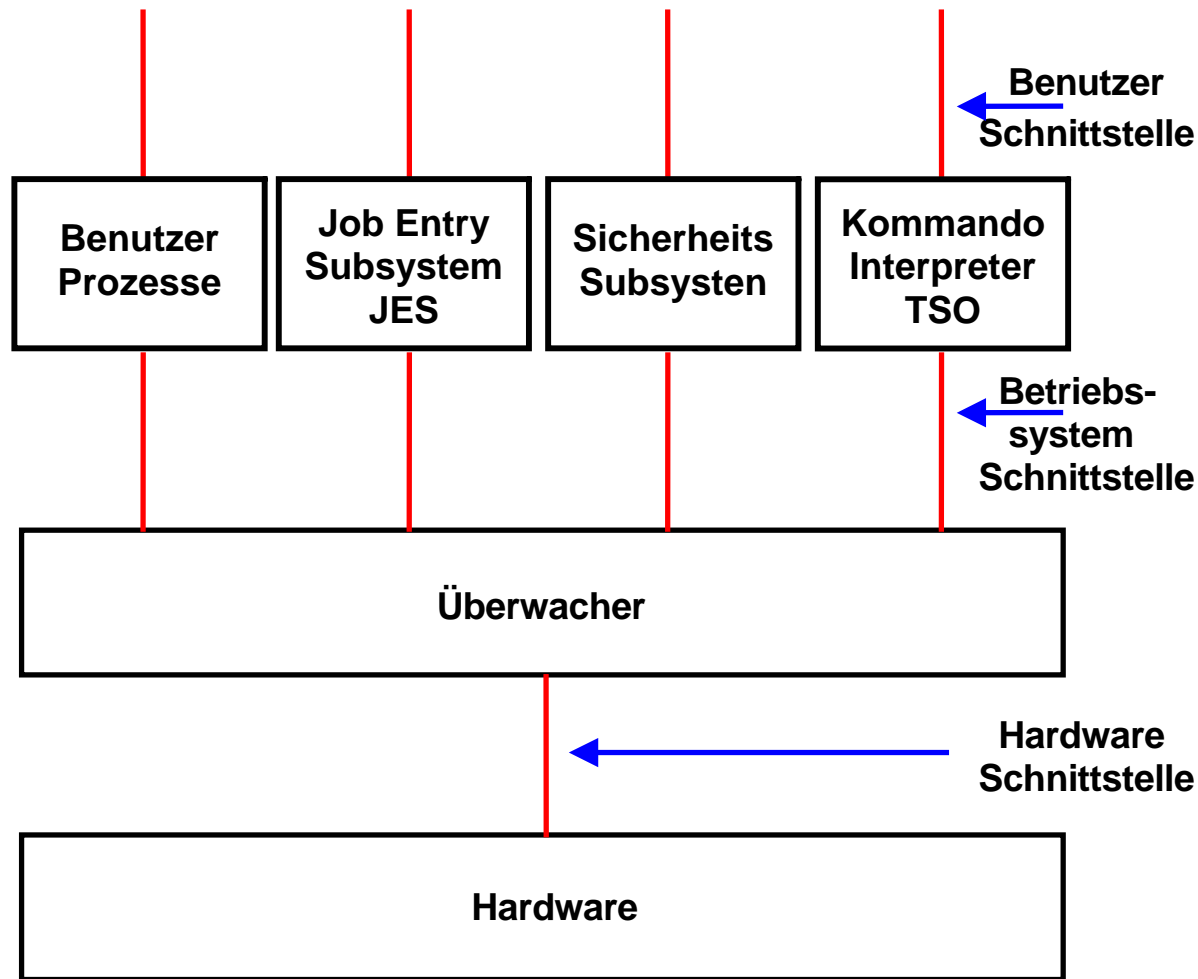
- | | |
|------------------------|---|
| Maschinenfehler | Beispiel Paritäts-Fehler (Bus, Hauptspeicher), fehlerhafte Addition |
| Reset | Beispiel: Setzt Zentraleinheit (und System) in jungfräulichen Zustand |
| I/O | Beispiel: Signalisiert den Abschluß einer E/A Operation |
| Extern | Beispiel: System externes Signal, z. B. Ablauf des Zeitgebers. |
| Programm | Beispiele: Division durch Null,
Fehlseitenunterbrechung,
illegaler OP Code,
illegale Adresse |
| Systemaufruf | (SVC) Aufruf des Überwachers im Benutzerprogramm (eigentlich ein Maschinenbefehl) |



Interruption Process Flow

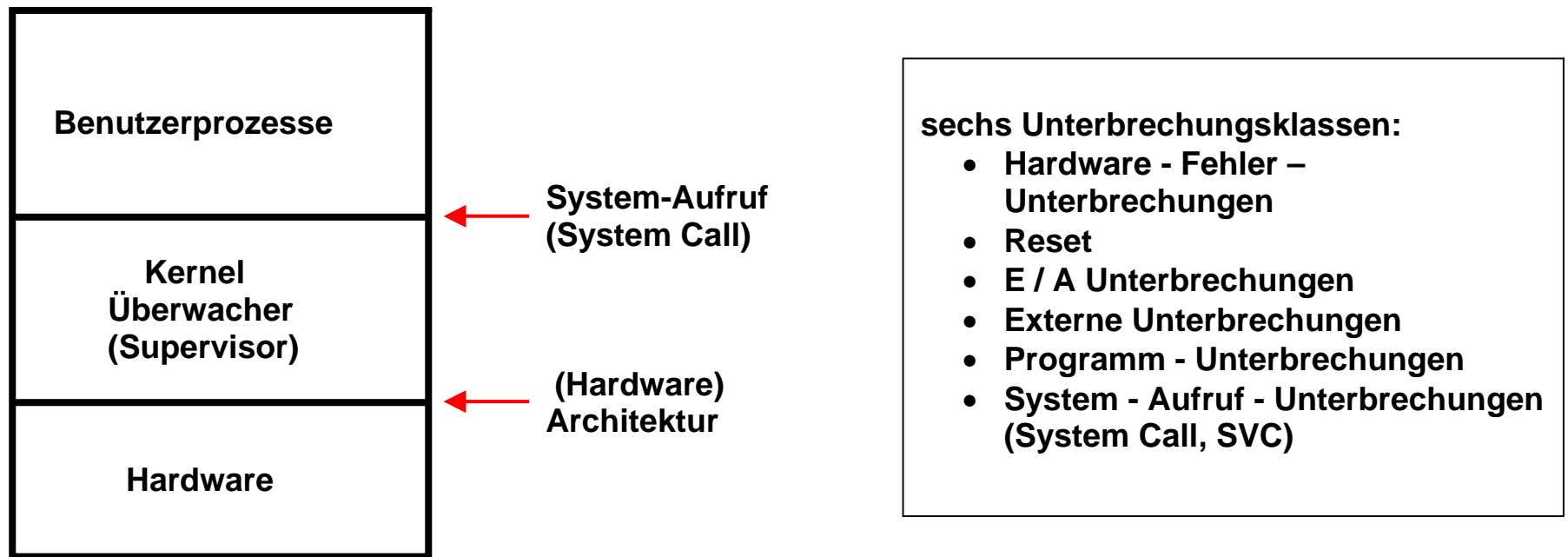
Eine Unterbrechung bewirkt das Abspeichern des derzeit gültigen PSW (als old PSW bezeichnet) im Hauptspeicher an einer hierfür vorgesehenen Adresse und ein Laden eines neuen PSW (new PSW) in das PSW Register. Der Befehlszählerteil des neuen PSW enthält die Adresse des ersten Befehls der Unterbrechungsroutine. Die CPU befindet sich automatisch im Überwacherstatus.

Als allerletzten Befehl führt die Unterbrechungsroutine den Maschinenbefehl „LPSW (Load Programm Status) aus. Dieser bewirkt, dass das old PSW in das PSW Register geladen wird. Der Befehlszählerteil des PSW enthält die Adresse des Maschinenbefehls, den die CPU ohne Eintreten der Unterbrechung als nächstes ausgeführt hätte.



Schichtenmodell der Rechnerarchitektur

Das Betriebssystem besteht aus einer zentralen Steuerungs- und Verwaltungsfunktion, dem Überwacher (Supervisor, Kernel), und weiteren Systemprogrammen oder Komponenten, die unter z/OS als „Subsysteme“ bezeichnet werden. Beispiele für Subsysteme sind der Kommandointerpreter (TSO), das Job Entry Subsystem (JES) oder das DB2 Datenbanksystem. Subsysteme sind Systemprozesse, die parallel zu den Benutzerprozessen laufen.



Betriebssystem = Überwacher + Systemprogramme

Das Betriebssystem besteht aus dem Überwacher und weiterer Software, welche die Subsysteme implementiert. Subsysteme können (müssen aber nicht) permanent Platz im Hauptspeicher belegen.

Die Hardware reagiert auf die Eingabe von Maschinenbefehlen und Unterbrechungen.

Der Überwacher kann nur über Unterbrechungen aufgerufen werden. Er läuft im Überwacherstatus.

System - Aufrufe (System Calls) sind die einzige Möglichkeit für Benutzerprozesse, mit dem Überwacher zu kommunizieren. Unter z/OS implementiert der SVC Maschinenbefehl die System Call Funktion. Beim x86 hat der INT Maschinenbefehl die gleiche Funktion. Ein System Call ist eine Routine, die im Benutzer Status läuft, u.a. den SVC (oder INT) Maschinenbefehl ausführt, und dadurch den Übergang vom Benutzerstatus in den Überwacherstatus herbeiführt.

Supervisor

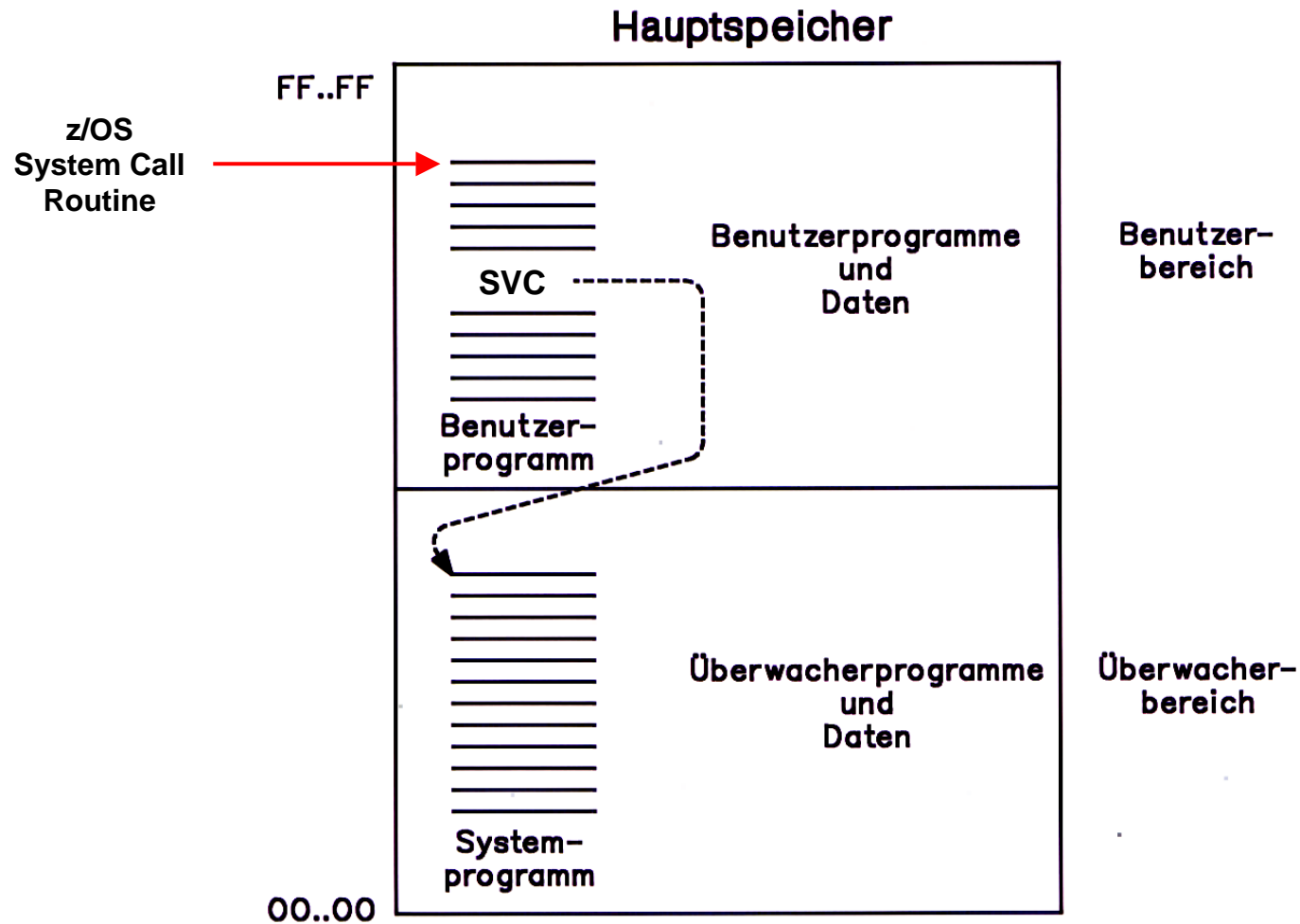
Der Supervisor besteht aus :

- 1. Datenbereichen (Control Blocks)**
- 2. Programmteilen, welche Controlblock Daten manipulieren**

In der Regel nicht strukturiert (Tanenbaum : The Big Mess).

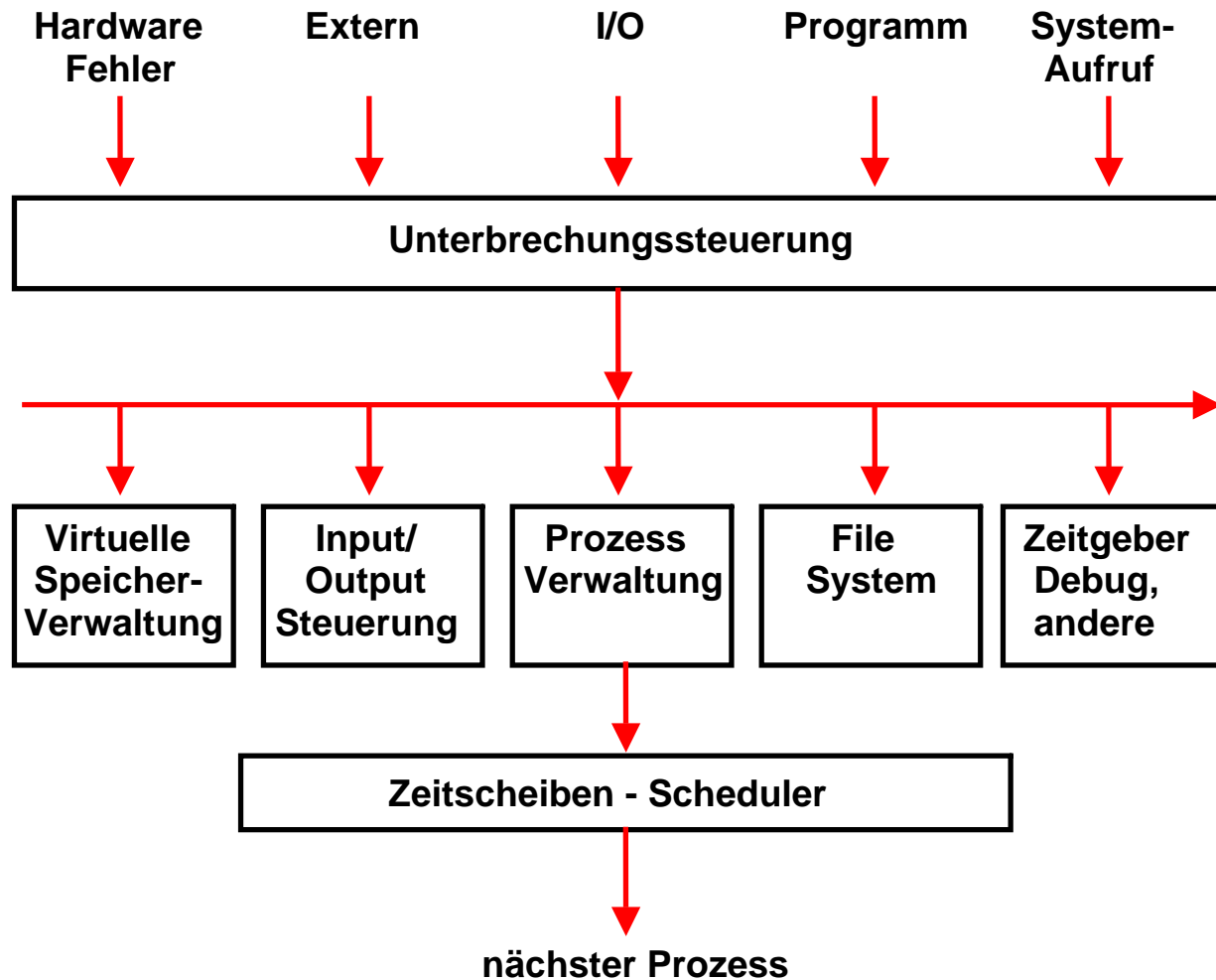
Der Überwacher enthält Funktionen die von vielen Prozessen gemeinsam genutzt werden.

Häufig verbringt ein Prozess 50 % der Ausführungszeit (Pfadlänge) mit der Ausführung von Überwacherfunktionen (läuft 50 % der Zeit im Überwacherstatus).



Systemaufruf (System Call)

Ein System z Benutzerprogramm kann eine Funktion des Überwachers durch Ausführung des SVC (Supervisor Call) Maschinenbefehls aufrufen. Der SVC Befehl übergibt hierzu einen Parameter, welcher die Art der gewünschten Funktion angibt. Ein Beispiel ist die Durchführung einer WRITE operation, welche Daten auf den Plattenspeicher schreibt. Die Ausführung des SVC Maschinenbefehls bewirkt eine Unterbrechung. Die Unterbrechungsroutine bewirkt unter anderem den Wechsel von Benutzerstatus in den Kernel Status.



Struktur des Überwachers

Der Aufruf des Überwachers (Supervisor, Kernel) erfolgt grundsätzlich über Unterbrechungen.

Je nach Art der Unterbrechung werden unterschiedliche Komponenten des Überwachers aufgerufen.

Benutzerprozesse nehmen Dienste des Überwachers über eine architekturierte Schnittstelle, den Systemaufruf / System Call, Supervisor Call, SVC –) in Anspruch. Ein System Call ist eine Routine, die u.a. den SVC Maschinenbefehl enthält.

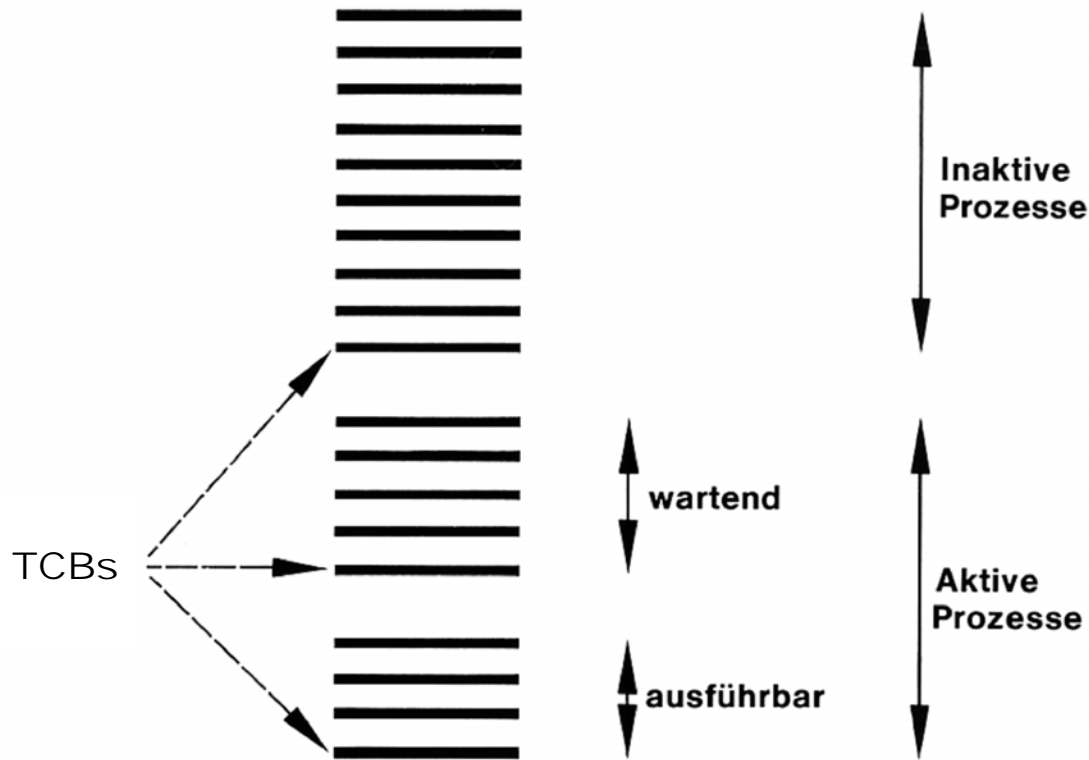
Der Scheduler (Zeitscheibensteuerung) sucht den nächsten auszuführenden Prozess aus.

Struktur des Überwachers

Der Überwacher wird über eine Unterbrechung aufgerufen. Dies ist der einzige Weg, über den man eine Funktion des Überwachers in Anspruch nehmen kann.

Die wichtigsten Überwacherfunktionen sind:

- Die Unterbrechungssteuerung untersucht die Art der aufgetretenen Unterbrechung und ruft je nach Art eine andere Komponente des Überwachers auf.
- Die virtuelle Speicherverwaltung ordnet virtuellen und realen Speicherplatz zu. Sie bestimmt, welche Seiten in Rahmen des Hauptspeichers abgebildet werden und welche Seiten auf den externen Seitenspeicher (Auxiliary Store) ausgelagert werden. Sie lädt bei Bedarf Seiten vom externen Seitenspeicher in den Hauptspeicher.
- Die Input/Output Steuerung (I/O Supervisor) wird aufgerufen wenn ein Benutzerprogramm eine Lese oder Schreiboperation auf ein I/O Gerät (z.B. Plattenspeicher) durchführen will. Sie nimmt auch eine I/O Unterbrechung entgegen, die z.B. besagt, dass ein Plattenspeicher eine I/O Operation erfolgreich abgeschlossen hat.
- Die Prozessverwaltung aktiviert und deaktiviert Prozesse. Aktive Prozesse verfügen über Ressourcen im Hauptspeicher. Bei deaktivierten Prozessen sind alle Ressourcen (vermutlich) auf einen Plattenspeicher ausgelagert.
- Datenstrukturen auf Plattenspeichern werden mit Hilfe des File Systems verwaltet. Windows verwendet die NTFS und FAT32 File Systems. VSAM, PDSe und zFS sind die wichtigsten z/OS File Systems.
- Zahlreiche weitere Funktionen wie Zeitscheibensteuerung und Funktionen zum Debuggen von Software sind vorhanden.
- Der Scheduler verwaltet die Warteschlangen der TCBs. Er versetzt Prozesse in den Zustand wartend, ausführbar oder laufend. Er steuert Prioritäten, nach denen entschieden wird, welcher Prozess vom Zustand wartend in den Zustand ausführbar überführt wird.



Task Control Block (TCB) Warteschlangen

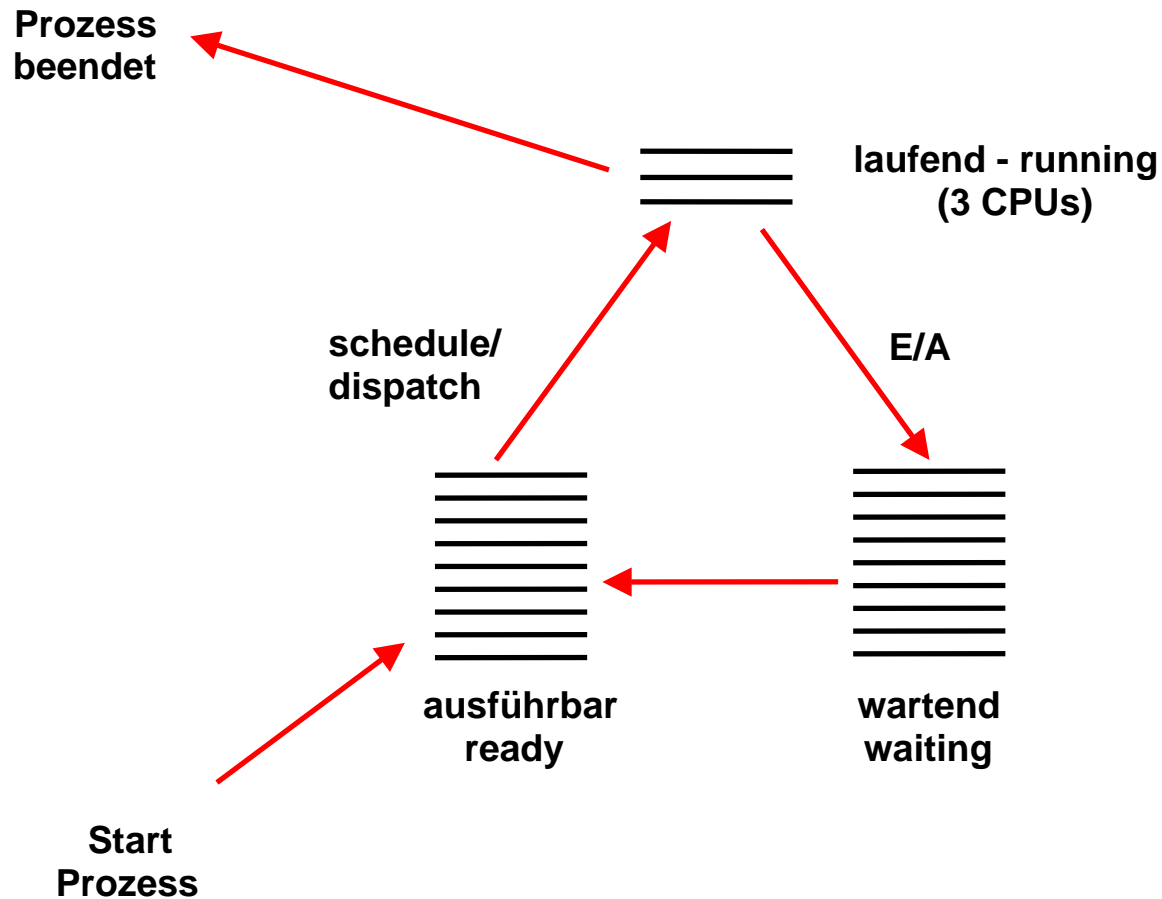
Ein Prozess wird durch einen Prozessleitblock beschrieben. Der Prozessleitblock enthält wichtige Funktionen für die Ausführung eines Prozesses.

z/OS verwendet die Bezeichnung Task Control Block (TCB). Andere Architekturen verwenden die Bezeichnung Process Control Block (PCB).

Zu jedem Zeitpunkt beanspruchen viele Prozesse Platz im Hauptspeicher und könnten ausgeführt werden. In einem Rechner mit einer CPU verarbeitet die CPU aber nur einen Prozess. In einem Mehrfachrechner mit n CPUs können gleichzeitig n Prozesse ausgeführt werden.

TCBs der wartenden und ausführbaren Prozesse werden vom Scheduler in Warteschlangen eingereiht und verwaltet.

Prozesse können inaktiv sein. Dies bedeutet, ihre TCBs und ihre Seitensind auf einen Plattenspeicher ausgelagert. Der Scheduler entscheidet, ob und wann ein inaktiver Prozess aktiviert wird.



Zustand von Prozessen

„Jede Linie entspricht einem Prozess, dargestellt durch seinen TCB (Task Control Block).“

Alle Prozesse sind entweder im Zustand laufend (werden von einer CPU verarbeitet), im Zustand wartend (haben eine I/O Operation eingeleitet und warten auf ihren Abschluss), oder im Zustand ausführbar (können vom Scheduler jederzeit in den Zustand laufend versetzt werden, sobald eine CPU frei wird).

Angenommen 3 CPUs – 3 Prozesse sind gleichzeitig im Zustand laufend

Inhalt des TCB

Ein Prozess wird durch seinen TCB repräsentiert und dargestellt. Der TCB ist ein Bereich im Hauptspeicher und enthält Informationen über den Prozess.

Mehrzweck Register , Gleitkommaregister usw sowie das PSW sind in einer CPU nur einmal vorhanden. Wird ein Prozess vom laufenden in den wartenden Zustand versetzt, wird der Inhalt der Register und des PSW in seinem TCB abgespeichert, ebenso alle weitere Information, die den Prozess charakterisiert. Wird der gleiche Prozess später einmal wieder vom ausführbaren in den laufenden Zustand versetzt, wird diese Information benutzt, um den bisherigen Zustand der CPU wieder herzustellen..

Scheduler - Dispatcher

Der Scheduler/Dispatcher ist die Komponente des Überwachers, welcher drei Warteschlangen für die laufenden, wartenden und ausführbaren TCBs verwaltet. Er überführt bei gegebenen Anlass einen TCB von einer Warteschlange in eine andere.

Wenn eine CPU frei wird, selektiert der Scheduler/Dispatcher aus der Warteschlange der ausführbaren TCBs einen Prozess und versetzt ihn in den Zustand laufend. Dies kann über Prioritäten gesteuert werden, und der z/OS Überwacher verfügt hierzu über eine Prioritätssteuerung.

In der Mehrzahl der Fälle bleibt ein Prozess nur kurze Zeit im Zustand laufend. Um zu verhindern, dass ein bestimmter Prozess eine CPU zu lange in Anspruch nimmt, verfügt der Scheduler/Dispatcher über eine Zeitscheibensteuerung. Eine Zeitscheibe ist ein Zeitintervall von typischerweise wenigen Millisekunden. Verbleibt ein Prozess im Zustand laufend über seine Zeitscheibenlänge hinaus, erfolgt eine (Seitenfehler, page fault) Unterbrechung durch einen Zeitgeber. Diese bewirkt, dass der laufende Prozess in den Zustand ausführbar versetzt wird, und ein anderer Prozess dafür in den Zustand laufend versetzt wird.

In der Regel laufen unterschiedliche Prozesse auf den CPUs eines Rechners. Es ist jedoch möglich, dass ein Prozess mehr als eine CPU in Anspruch nimmt. Ein Beispiel ist der CICS Transaktionsmonitor, der einen einzigen Prozess darstellt, der viele Transaktionen parallel verarbeitet.