

# Einführung in die Technische Informatik

FFF



Fugos furioser Fragenkatalog

# Mächtiges Vorwort

*Als ich 1984 das Licht der Welt erblickte... ;-)*

**Nur das allerwichtigste:** Dieser Fragenkatalog wurde im WS 2007 als Vorbereitung für die Prüfung erstellt, umfasst viele Fragen aus vergangenen Tests und stellt gleichzeitig die gesamte Theorie dar, die ich für die Prüfung gelernt habe. Es stecken einige Stunden Arbeit darin und vielleicht kann ja irgendjemand etwas mit diesem Dokument anfangen. Ich habe Gott sei Dank keine Verwendung mehr dafür.

Es steht jedem frei, dieses Dokument zu erweitern, zu verbessern, unter neuem Namen wieder einzustellen oder sonst was damit zu machen. Dass ich die Prüfung mit 50 von 100 möglichen Punkten gerade so geschafft habe, soll die Perfektion dieses Werkes nur noch unterstreichen.

Viel Erfolg auf all Euren Lebenswegen ;-)

**Inhalt**

1.....	W
ann tritt ein Page-Fault auf? .....	7
2.....	I
m Falle eines Page-Faults sind welche Heuristiken (Page Replacement Methoden) anwendbar? .....	7
3.....	W
as versteht man unter atomic actions? .....	7
4.....	W
as ist bei Prozessen der Unterschied zwischen den Zuständen blocked und suspended?.....	7
5.....	W
as versteht man unter einem Semaphor? .....	7
6.....	A
us welchen Datenbereichen besteht die Datenstruktur des Semaphors?.....	7
7.....	W
elche potentiellen Gefahren bestehen in einem Rechnersystem, wenn ein direkter Zustandsübergang von BLOCKED nach RUNNING zulässig ist? .....	7
8.....	W
as muss ein Betriebssystem machen, wenn zwischen zwei threads, die zum gleichen Prozess gehören, umgeschaltet wird (dh Scheduler führt einen Context Switch zwischen den threaden aus)?.....	7
9.....	N
ennen Sie die Adressierungsarten, die Sie kennen! (hier mit Erklärung) .....	7
10.....	W
as ist Interleaved Memory und wozu wird er eingesetzt?.....	8
11.....	N
ennen Sie die drei Replacement Strategien! .....	8
12.....	E
erklären Sie die Buffered-Write-Through-Methode im Zusammenhang mit Caches! Welche Probleme können dabei auftreten? .....	8
13.....	W
as versteht man unter DMA und warum wird es verwendet? .....	8
14.....	N
ennen Sie 2 Arten von DMA!.....	8
15.....	E
erklären Sie den Vorgang des DMA! .....	8
16.....	W
ozu dient ein virtueller Speicher? .....	9
17.....	W
elchen Vorteil besitzen die „von Neumann“ und die „Harvard-Architektur“? Geben Sie auch schematische Darstellungen dieser beiden Architekturen an! .....	9
18.....	W
elchen Vorteil besitzt die „Harvard Architektur“ gegenüber der „von Neumannschen“ und wo wird dieser genutzt? .....	9
19.....	W
as versteht man unter Flow Control?.....	9
20.....	W
as versteht man unter bedingten bzw. unbedingten Sprüngen? (Buch 148) .....	9
21.....	W
as versteht man unter dem Program Status Word? .....	9
22.....	W
as sind Subroutine Calls? .....	9
23.....	W
as sind Interrupts? .....	10
24.....	W
as versteht man unter Race Conditions?.....	10
25.....	W
as versteht man unter Trashing?.....	10
26.....	N
ennen Sie alle Layer des ISO OSI Reference Models! .....	10
27.....	W
elches Problem tritt beim Linken im Zusammenhang mit direkter Adressierung auf und wie kann es behoben werden? .....	10
28.....	W
as versteht man unter einem Deadlock? .....	10
29.....	W
ann entstehen Deadlocks? .....	10
30.....	K
ann es zwischen threads zu einem Deadlock kommen? .....	10

31.	.....	W
	elche Möglichkeiten gibt es für die Behandlung eines Deadlocks?.....	11
32.	.....	E
	erklären Sie die RISC Architektur! .....	11
33.	.....	W
	elche Weiterentwicklungen der RISC Architektur sind im Vorlesungsbuch beschrieben?.....	11
34.	.....	W
	elche löschbaren optischen Platten gibt es und auf welchen physikalischen Prinzipien beruhen sie?.....	11
35.	.....	W
	as versteht man unter einem endlichen deterministischen Automaten?.....	11
36.	.....	W
	as ist der Schmitt-Trigger? .....	11
37.	.....	D
	efinieren Sie die Hysterese beim Schmitt-Trigger .....	11
38.	.....	D
	ie ALU generiert zwei Signale N und Z. Geben Sie an, was diese bedeuten! .....	11
39.	.....	W
	as ist ein Zero-Crossing-Detector? .....	12
40.	.....	E
	erläutern Sie die Funktion der anti-parallelen Diodenschaltung im Gegenkopplungs-zweig des Operationsverstärkers eines Zero-Crossing Detectors .....	12
41.	.....	W
	as passiert mit dem Vorzeichenbit bei einem arithmetischen Rechtsshift? .....	12
42.	.....	W
	as passiert mit dem Vorzeichenbit bei einem arithmetischen Linksshift? .....	12
43.	.....	W
	as versteht man unter Context Switch Time? .....	12
44.	.....	E
	erklären Sie die Funktionsweise vom Round-Robin Scheduling.....	12
45.	.....	N
	ennen Sie alle Prozesszustände in einem Betriebssystem und zu welchem Zustand sie jeweils führen können. ....	12
46.	.....	W
	odurch unterscheiden sich bei USB die Connect Erkennung von Full-Speed und Low-Speed Geräten? .....	12
47.	.....	G
	eben Sie die Formel für die durchschnittliche Speicher-Zugriffszeit an. Die Trefferrate soll mit h angegeben werden. ....	12
48.	.....	W
	as versteht man unter dem Cycle Stealing Verfahren? .....	12
49.	.....	W
	elche anderen Refresh-Verfahren gibt es?.....	12
50.	.....	W
	as versteht man unter Interlocking beim Pipelining Konzept? .....	12
51.	.....	G
	eben Sie die Formel für die Kondensatorladung / Entladung an! .....	13
52.	.....	W
	elche Arten von Speicherbausteinen kennen Sie? .....	13
53.	.....	W
	as sind „0-aktive Eingänge“ bei ICs (integrierten Schaltungen)?.....	13
54.	.....	W
	elche Funktion hat der ENABLE Eingang bei einem Decoder? .....	13
55.	.....	G
	eben Sie die Übertragungskennlinie $U_a = f(U_e)$ eines Inverters an. Wo liegt die verbotene Zone? .....	13
56.	.....	W
	elche drei Sichtweisen gibt es beim Entwurf elektronischer Schaltungen laut dem Y-Modell? .....	13
57.	.....	W
	elche drei Arbeitsschritte sind zur Erstellung eines Moduls bzw. einer Komponente in VHDL erforderlich? .....	13
58.	.....	V
	on welchen zwei Faktoren hängen Zustandsänderungen bei Schaltwerken ab?.....	13
59.	.....	W
	ozu dient die Ziehkapazität bei einem Quarzoszillator? .....	13
60.	.....	W
	as versteht man unter „Fan In“? .....	14
61.	.....	W
	as versteht man unter „Fan Out“? .....	14
62.	.....	Z
	eichen Sie eine Gatterschaltung für die XOR-Funktion! .....	14
63.	.....	E
	erklären Sie die Abkürzung ASIC! .....	14
64.	.....	Z
	eichen Sie die Übertragungskennlinie eines invertierenden Schmitt-Triggers! .....	14

65.	.....	W
	as versteht man unter dem Tastgrad $g$ einer Rechteckimpulsfolge des Taktgenerators eines Rechners? .....	14
66.	.....	W
	as versteht man unter dem Scratchpad eines Prozessors? .....	14
67.	.....	W
	elchen Vorteil hat die Erhöhung der Baudichte bei Mikroprozessoren? .....	14
68.	.....	N
	ennen Sie einen Vorteil und einen Nachteil von Instruktionen mit variabler Länge im Vergleich zu Instruktionen mit fixer Länge. ....	14
69.	.....	W
	elche zwei Möglichkeiten gibt es, die Adressen der Ports für Input / Output-Operationen zu vergeben?.....	14
70.	.....	W
	as ist ein Trap? Geben Sie ein Beispiel!.....	15
71.	.....	W
	o werden die Return Adressen bei Unterprozeduraufrufen gespeichert?.....	15
72.	.....	W
	arum ist eine hohe Trefferquote der branch prediction Einheit bei modernen Prozessoren so wichtig? Denken Sie daran, dass die Prozessoren eine sehr lange Pipeline verwenden. ....	15
73.	.....	W
	as versteht man unter logischer Parallelität von Prozessoren? .....	15
74.	.....	N
	ennen Sie einen Nachteil von asynchronen Methoden zur Interprozesskommunikation (IPC) im Vergleich zu synchronen Methoden. ....	15
75.	.....	N
	ennen Sie zumindest drei Bestandteile des Prozessdeskriptors! .....	15
76.	.....	D
	urch welche drei im Buch erläuterten Arten der Parallelverarbeitung kann die Performance eines Prozessors gesteigert werden? .....	15
77.	.....	W
	as ist die Aufgabe eines Linkers?.....	15
78.	.....	W
	as versteht man unter dem Program Counter bei einem Mikroprozessor?.....	15
79.	.....	W
	elche zwei Möglichkeiten gibt es, die Adresse einer Interrupt Service Routine (ISR) zu bestimmen?.....	15
80.	.....	W
	elche zwei Komponenten benötigt jeder thread für sich alleine? .....	15
81.	.....	N
	ennen Sie mindestens zwei Probleme bei der Verwendung von Threads! .....	15
82.	.....	W
	as versteht man unter einem System Call? .....	16
83.	.....	W
	as versteht man unter Scheduling?.....	16
84.	.....	J
	ob-Scheduling .....	16
85.	.....	P
	rozess-Scheduling .....	16
86.	.....	F
	orderungen an Scheduling Algorithmen .....	16
87.	.....	S
	cheduling-Methoden .....	16
88.	.....	T
	hread-Scheduling .....	17
89.	.....	M
	essage Passing.....	17
90.	.....	W
	as versteht man unter einem Hazard? .....	17
91.	.....	W
	elche Arten von Pipeline-Hazards gibt es? .....	17
92.	.....	H
	albaddierer .....	17
93.	.....	V
	olladdierer .....	17
94.	.....	C
	odierer .....	17
95.	.....	M
	ultiplexer .....	17
96.	.....	D
	emultiplexer .....	17
97.	.....	O
	perationsverstärker.....	17

98.	.....	K
	omparatoren für analoge Signale .....	17
99.	.....	K
	omparatoren für digitale Signale .....	17
100.	.....	T
	orschaltungen .....	18
101.	.....	F
	ensterdiskriminator.....	18
102.	.....	U
	nivibrator .....	18
103.	.....	Q
	uarzoszillator.....	18
104.	.....	S
	ägezahngenerator.....	18
105.	.....	T
	ristate Outputs.....	18
106.	.....	O
	pen-Collector-Schaltungen.....	18
107.	.....	M
	oore Schaltwerk .....	18
108.	.....	M
	ealy Schaltwerk .....	19
109.	.....	P
	rozessor.....	19
110.	.....	R
	echenwerk .....	19
111.	.....	L
	eitwerk.....	19
112.	.....	W
	oraus ergibt sich die Cache-Performance .....	19
113.	.....	W
	as versteht man unter Swapping?.....	19
114.	.....	W
	as versteht man unter Paging? .....	19
115.	.....	W
	as versteht man unter Segmentierung?.....	19
116.	.....	W
	as versteht man unter einem Cache-Speicher? Wie ist er aufgebaut? .....	19
117.	.....	W
	elche Arten von Caches kennen Sie? .....	20
118.	.....	W
	as versteht man unter VHDL? .....	20
119.	.....	W
	elche Entwurfsebenen gibt es in VHDL? .....	20
120.	.....	W
	elche Schaltkreisfamilien kennen Sie? .....	20

## Wann tritt ein Page-Fault auf?

Ein Page-Fault tritt dann auf, wenn eine Adresse angefordert wird, die in einer Page ist, die nicht im Hauptspeicher, sondern auf der Platte liegt. Dabei wird das Programm unterbrochen und die benötigte Page wird in einen Page Frame geladen.

## Im Falle eines Page-Faults sind welche Heuristiken (Page Replacement Methoden) anwendbar?

- **First In First Out (FIFO):** Jede Page bekommt zum Zeitpunkt Ihres Ladens einen Zeitstempel. Wenn ein Page-Fault aufzulösen ist, wird die „älteste“ Page ersetzt.
- **Least Recently Used (LRU):** Jene Page wird ersetzt, deren letzte Referenz am weitesten zurück liegt
- **Least Frequently Used (LFU):** Jene Page ersetzen, die am wenigsten benutzt wurde
- **Not Used Recently (NUR):** Pages erhalten eine Kennung Referenced/Not Referenced, welche in gewissen Abständen zurückgesetzt wird.

## Was versteht man unter atomic actions?

Werden auch als unteilbare Operationen bezeichnet. Eine Folge von Instruktionen, die entweder ganz oder gar nicht durchgeführt werden müssen. Bei asynchronem Wechsel von Blocked auf Running dienen sie dazu, invaliden Daten vorzubeugen.

## Was ist bei Prozessen der Unterschied zwischen den Zuständen blocked und suspended?

- **Blocked:** Zustand wird eingenommen, wenn der Prozess an einem Punkt angekommen ist an dem er auf den Eintritt eines externen Ereignisses warten muss (zB Tastendruck auf Tastatur).
- **Suspended:** Entspricht einer Blockierung von außen, wird also in der Regel vom Betriebssystem oder einem anderen Prozess verursacht.

## Was versteht man unter einem Semaphor?

Semaphore sind ein geeignetes Konzept zur Vermeidung von Race Conditions. Ein Semaphor ist ein Objekt, besteht aus einem auf 0 initialisierten Counter und einer zunächst leeren Liste für Prozess-IDs. Eine Reihe von System Calls erlaubt nun die Synchronisation paralleler Prozesse.

## Aus welchen Datenbereichen besteht die Datenstruktur des Semaphors?

Ein Semaphor besteht aus einem Zähler und einer Warteschlange.

## Welche potentiellen Gefahren bestehen in einem Rechnersystem, wenn ein direkter Zustandsübergang von BLOCKED nach RUNNING zulässig ist?

Dadurch wird der aktuelle Prozess asynchron beendet, was zu invaliden Daten führen kann. Bei den meisten Betriebssystem ist dies nicht möglich. Bei Echtzeitsystem ist diese Funktion beispielsweise für Notabschaltungen möglich.

## Was muss ein Betriebssystem machen, wenn zwischen zwei threads, die zum gleichen Prozess gehören, umgeschaltet wird (dh Scheduler führt einen Context Switch zwischen den threads aus)?

- Threadspezifische Daten (Lokale Variablen, Stack) retten
- Register retten
- Threadspezifische Daten vom neuen thread zurückschreiben
- Register vom neuen thread zurückschreiben

## Nennen Sie die Adressierungsarten, die Sie kennen! (hier mit Erklärung)

- **Implied Mode:** Die implizite Adressierung ist keine Adressierungsart im eigentlichen Sinne. Die Gruppe von Befehlen, die dieses Verfahren verwendet, ist nur für einen bestimmten Operanden definiert, darum muss dieser nicht eigens in einem Adressfeld angegeben werden. Ein Beispiel für ein solchen Kommando ist die Instruktion „Enable Interrupts“.
- **Register Mode:** Die Register werden über die ihnen zugeordneten Adressen angesprochen, wobei diese auch in der Maschinen-Code-Schicht verwendet werden können. Am Mode-Field erkennt der Interpreter, dass es sich bei der Angabe nicht um eine Adresse im Hauptspeicher, sondern um die eines Registers handelt. (zB R3 ← R4 + R5).

- **Immediate Mode:** Keine Adressierung im eigentlichen Sinn, weil der Wert des Operanden direkt im Maschinenbefehl enthalten ist. Operandenfeld enthält Konstante statt einer Adresse. Verwendung zur Initialisierung von Registern oder Inkrementieren eines Schleifenzählers.
- **Direct Addressing:** Der Wert im Operandenfeld entspricht der Speicheradresse des gewünschten Operanden. Verwendung bei Transferoperationen und Sprungbefehlen.
- **Register-Indirect Mode:** Das Register, welches im Maschinen-Befehl als Operand angegeben ist, enthält die effektive Adresse des gewünschten Datenwortes.
- **Program-Counter-Relative-Addressing Mode:** Eine Variante der Base- bzw. Indexed-Adressierung. Dabei berechnet sich die effektive Adresse durch die Addition eines im Befehl angegebenen Offsets zum aktuellen Programmzählerstand. Das gestattet das Erstellen von Programmen, die an einer beliebigen Stelle des Arbeitsspeichers lauffähig sind.
- **Indirect-Addressing Mode:** Eine zweistufige Speicher-Adressierung. Mehrere sequenziell aufeinander folgende Adressberechnungen sind dafür notwendig. Das Ergebnis der ersten Berechnung liefert die Adresse eines Speicherwortes, das entweder die Adresse oder den Offset der folgenden Berechnung enthält.

### Was ist Interleaved Memory und wozu wird er eingesetzt?

Das Konzept des Interleaved Memory geht davon aus, dass meistens sequenziell auf den Speicher zugegriffen wird. Sequenziell direkt hintereinander liegende Befehle können schneller erreicht werden, da Adressen, die in verschiedenen Speicherblöcken liegen, schneller auf den Adressbus gelegt werden können, als es sonst – bedingt durch Zugriffszeichen der Bausteine – möglich wäre. Die Daten kommen dadurch ebenfalls schneller beim Prozessor an.

### Nennen Sie die drei Replacement Strategien!

- **LRU:** Least Recently Used. Im Fall mehrerer Kandidaten wird jener ausgewechselt, dessen letzter Aufruf am längsten zurück liegt.
- **LFU:** Least Frequently Used. Die Zeile wird ausgetauscht, die am seltensten in letzter Zeit verwendet wurde.
- **RANDOM:** Zufällige Selektion

### Erklären Sie die Buffered-Write-Through-Methode im Zusammenhang mit Caches! Welche Probleme können dabei auftreten?

Vereint die Vorteile von Copy-Back und Write Through. Bei diesem Verfahren wird der neue Wert gleichzeitig sowohl in den Cache als auch in einen zweiten, schnellen Zwischenspeicher (Buffer) eingetragen. Während der Prozessor bereits mit der weiteren Abarbeitung des Programms fortfahren kann, werden die gepufferten Daten in den Hauptspeicher übertragen. Auf diese Weise kann die Datenkohärenz erhalten bleiben. Problem: In dem seltenen Fall, dass mehrere Schreiboperationen direkt aufeinander folgen und der Puffer daher nicht schnell genug in den Hauptspeicher übertragen werden kann, muss der Prozessor warten.

### Was versteht man unter DMA und warum wird es verwendet?

*Direct Memory Access* ermöglicht es, die Kommunikation zwischen dem Prozessor und den meist sehr viel langsameren peripheren Geräten zu beschleunigen. Er dient zur direkten Übertragung großer Datenmengen vom bzw. zum Speicher, ohne die CPU dabei in Anspruch zu nehmen. Um Konflikte zu vermeiden, darf die CPU während des DMA nicht auf den Bus zugreifen. Somit ist nicht sichergestellt, dass der Prozessor in der Zeitspanne, die zum Transfer nötig ist, Aufgaben durchführt, die ohne externen Buszugriff möglich sind. Auch ist eine zusätzliche Kontrolleinheit nötig, der DMA-Controller.

### Nennen Sie 2 Arten von DMA!

- Transparent DMA
- Cycle Stealing DMA

### Erklären Sie den Vorgang des DMA!

1. Der Prozessor teilt dem DMAC die Adresse der Quelle (resource pointer), die des Ziels (destination pointer) die die Größe der zu übertragenden Daten (block length) mit. Hierauf kann die CPU mit der Abarbeitung des Programms fortfahren.
2. Der DMAC fordert nun vom entsprechenden Gerät die Daten an und wartet, bis es zum Transfer bereit ist.

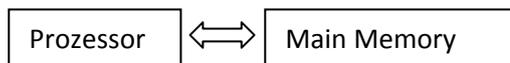
3. Nach dem Ende der Übertragung meldet der DMAC dem Prozessor den erfolgreichen Abschluss der Aktion meistens per Interruptsignal. Die Daten werden also direkt zwischen I/O-Device und Speicher ausgetauscht.

### Wozu dient ein virtueller Speicher?

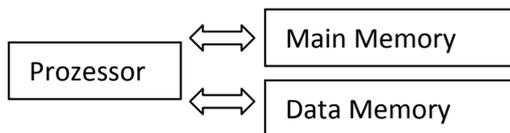
Bei einer großen Anzahl gleichzeitig speicherresidenter Programme waren vor der Erfindung dieses Konzeptes sehr große Speicher notwendig. In der Zeit der Kernspeicher waren jedoch der Kapazität Grenzen gesetzt. Diesem Problem wurde durch die Erfindung des virtuellen Speichers abgeholfen: Im Prinzip dient dabei ein Externspeicher als virtueller Hauptspeicher. Da ein Prozessor nur auf den physikalischen Hauptspeicher direkt zugreifen kann, werden die für die Exekution benötigten „Abschnitte“ einfach vom Externspeicher geladen. Zu jedem Zeitpunkt ist also nur der gerade benötigte Teil eines Programms speicherresident.

### Welchen Vorteil besitzen die „von Neumann“ und die „Harvard-Architektur“? Geben Sie auch schematische Darstellungen dieser beiden Architekturen an!

**Von Neumann Architektur:** Speicher enthält sowohl Daten als auch Programme. Übertragung über einen einzigen Datenbus (Flaschenhals → Caches)



Harvard Architektur: Getrennter Programm- und Datenspeicher. Übertragung in getrennten Bussen. Programme und Daten können gleichzeitig geladen bzw. geschrieben werden.



### Welchen Vorteil besitzt die „Harvard Architektur“ gegenüber der „von Neumannschen“ und wo wird dieser genutzt?

Bei der Harvard Architektur ist ein simultaner Zugriff über getrennten Speicher für Daten und Instruktionen möglich. Dieser Vorteil wird beim Instruction-Pipelining genutzt.

### Was versteht man unter Flow Control?

Flow-Control-Operationen sind Maschinenbefehle, die es ermöglichen, den sequenziellen Ablauf eines Programms zu unterbrechen.

### Was versteht man unter bedingten bzw. unbedingten Sprüngen? (Buch 148)

Sprünge können sowohl auf Register-Transfer-Ebene als auch auf Maschinen-Code-Ebene erfolgen. Bei dieser Operation wird der *Micro Instruction Counter* (MIC = Binärzähler) mit einem neuen Wert geladen. Bei bedingten Sprüngen (*Branch-operations*) kann zusätzlich eine Bedingung angegeben werden, die erfüllt sein muss, damit ein Sprung erfolgen kann.

### Was versteht man unter dem Program Status Word?

Bei der Ausführung von arithmetischen Operationen können Fehler wie zB ein Überlauf bei der Addition zweier großer Zahlen auftreten. Um dies zu dokumentieren und im weiteren Programmablauf nutzen zu können, besitzen die meisten Prozessoren ein PSW. Die Bits des PSW zeigen beispielsweise einen Überlauf oder Übertrag an und werden nur durch das Ergebnis einer Operation gesetzt oder gelöscht.

### Was sind Subroutine Calls?

Bei umfangreichen Programmen wird eine oft benötigte Folgen von Befehlen nur einmal programmiert und von verschiedenen Stellen aus aufgerufen. Danach wird die **Return-from-Subroutine** ausgeführt, die an der Stelle nach dem Aufruf der Subroutine im Programm fortsetzt. Es sind auch geschachtelte Aufruffolgen möglich. Die beiden Subroutine-Befehle führen stets folgende Kommandos aus:

#### Call-Subroutine

1. Retten des *Program Status Word* (PSW), der Registerinhalte und des *Program-Counters* (PC)
2. Laden des PC mit der Prozedur-Startadresse
3. Abarbeiten der Prozedur im Interpreter

#### Return-from-Subroutine:

1. Wiederherstellen des PSW und der Registerinhalte

2. Laden des PC mit dem Wert, der beim Call-Subroutine-Befehl gesichert worden ist
3. Fortsetzen des Programmablauf

### Was sind Interrupts?

Interrupts unterbrechen den herkömmlichen Programmablauf und springen in bestimmte – als ISR (Interrupt Service Routine) bezeichnete – Service-Routinen. Danach kann meist das unterbrochene Programm wieder fortgesetzt werden. Eine ISR ist ähnlich einer Subroutine. Auslöser ist ein prozessorinterner Ausnahmefall (zB Division durch Null) oder ein externes Ereignis.

### Was versteht man unter Race Conditions?

Als Race Condition (*Wettlaufsituation*) werden Konstellationen bezeichnet, in denen das Ergebnis einer Operation vom zeitlichen Verhalten bestimmter Einzeloperationen abhängt. Das Eingrenzen dieses Problems, das durch die parallele Abarbeitung entsteht, ist durch das relativ seltene und indeterministische Auftreten eine äußerst schwierige und zeitraubende Angelegenheit.

### Was versteht man unter Trashing?

Ein Phänomen, das beim Paging auftritt, wenn der Speicher zu klein ist um alle Working Sets aufzunehmen. Die Maschine ist praktisch ausschließlich damit beschäftigt, die referenzierten Pages vom Externspeicher zu laden.

### Nennen Sie alle Layer des ISO OSI Reference Models!

- Layer 1 – Physical Layer – Bitübertragungsschicht
- Layer 2 - Data Link Layer – Sicherungsschicht – Physikalische Adressierung mit Fehlererkennung, Fehlerreparatur und Datenflusskontrolle
- Layer 3 – Network Layer – Vermittlungsschicht – Logische Adressierung der Endgeräte, Routing, Traffic Shaping
- Layer 4 – Transport Layer – Transportschicht – Segmentierung von Datenpaketen und Stauvermeidung
- Layer 5 – Session Layer – Sitzungsschicht – Prozessorkommunikation zwischen Systemen, Aufbau von Sitzungen, Synchronisation
- Layer 6 – Presentation Layer – Konversion von Daten in passende Formate. Datenverschlüsselung
- Layer 7 – Application Layer – Anwendungsschicht – Stellt den Anwendungen Funktionalität zur Verfügung (zB Mail)

### Welches Problem tritt beim Linken im Zusammenhang mit direkter Adressierung auf und wie kann es behoben werden?

Beim Linken entstehen durch das Zusammenfügen von Modulen Veränderungen in den Adressen der Sprungziele. Bei direkter Adressierung muss der Linker eine Modifikation der Operanden durchführen, da sonst die Adressen der Sprungziele verfehlt werden. Für die Code-Segmente können Probleme dieser Art durch die Berücksichtigung der Konvention für Position Independent Code (PIC) gelöst werden. Dabei werden zB statt absoluter Sprungadressen nur die Distanzen (Displacements) von der aktuellen Adresse angegeben. Da sich durch die Verschiebung die Displacements nicht ändern, entfallen die Operandenmodifikationen völlig.

### Was versteht man unter einem Deadlock?

Unter einem Deadlock versteht man einen Zustand, in dem ein oder mehrere Prozesse auf Betriebsmittel warten, die dem Prozess selbst oder einem anderen beteiligten Prozess zugeteilt sind.

### Wann entstehen Deadlocks?

Für die Entstehung eines Deadlocks sind vier Bedingungen notwendig:

1. **Mutual Exclusion:** Ein bestimmtes Objekt kann zu jedem Zeitpunkt von höchstens einem Prozess okkupiert sein.
2. **Resource Wating:** Wenn ein beantragtes Objekt gerade besetzt ist, geht der anfordernde Prozess in den Zustand BLOCKED über, wartet also auf dessen Freiwerden.
3. **Partial Allocation:** Prozesse, die bereits im Besitz von Objekten sind, können die Zuteilung weiter beantragen.
4. **Nonpreemption:** Ein einmal zugeweiltes Objekt muss explizit durch den die Ressource haltenden Prozess wieder freigegeben werden, kann ihm also nicht zwangsweise entzogen werden.

### Kann es zwischen threads zu einem Deadlock kommen?

Es kann dazu kommen, weil die meisten Programme nicht für parallele, sondern für sequentielle Abarbeitung geschrieben sind. Je mehr Programme parallel ausgeführt werden müssen, desto größer ist die Wahrscheinlichkeit eines Fehlers, weil die Verwaltung komplizierter wird.

## Welche Möglichkeiten gibt es für die Behandlung eines Deadlocks?

- **Deadlock Detection and Recovery:** Im Zuge der ganz gewöhnlichen Zuteilungen und Freigaben von Objekten wird der Resource Allocation Graph aktualisiert und auf Zyklen (Deadlock Situationen) untersucht und gegeben falls durch Terminisierung von Prozessen aufgelöst. Probleme: Brutale Vorgehensweise kann zu Problemen führen. Analyse des Graphen verursacht System-Overhead.
- **Deadlock Prevention:** Beim Design des Betriebssystem wird darauf geachtet, eine der notwendigen Deadlock Bedingungen nicht zu erfüllen, damit dieser gar nicht erst auftreten kann, zB Ausschalten der Mutual Exclusion durch Spooling von Druckern
- **Deadlock Avoidance:** Vorausschauende Prüfung, ob Zuteilungen irgendwelche späteren Deadlocks zur Folge haben können. Die zwei dafür vorgesehen Algorithmen liefern nur beim Vorhandensein zusätzlicher Informationen „sichere“ Zuteilungsentscheidungen.

## Erklären Sie die RISC Architektur!

Reduced Instruction Set Computing – Das Arbeiten mit reduziertem Befehlssatz. Ein RISC-Befehlssatz verzichtet – zugunsten eines niedrigeren Dekodierungsaufwands auf Seiten der CPU – konsequent auf komplexe Befehle, wodurch die Ausführung einfacher als bei einem CISC (*Complex Instruction Set Computing*)-Befehlssatz ist. Zusätzlich wird auf optimale Implementierung der wenigen Befehle geachtet, was zu Performancesteigerung führt.

## Welche Weiterentwicklungen der RISC Architektur sind im Vorlesungsbuch beschrieben?

- **SPARC-Architektur: Scaleable Processor ARChitecture:** Konventioneller RISC Prozessor mit ausgefeilter Registertechnik. Offene Architektur, kann frei verwendet werden.
- **MIPS (Microprocessor without interlocking Pipelining Stages):** Performancesteigerung durch feinstufige Befehlspipeline und realisierte Speicherhierarchie

## Welche löschbaren optischen Platten gibt es und auf welchen physikalischen Prinzipien beruhen sie?

**Rein optische Platten:** Beim **Schreiben** wird ein dünner Film einer besonderen Metalllegierung mittels eines leistungsstarken Lasers zum Schmelzen gebracht, was einen Übergang vom kristallinen zum amorphen Zustand zur Folge hat, der weniger Licht reflektiert. Dies wird dann beim Abtasten mit einem schwächeren Laser ausgenutzt, indem man die Stärke des reflektierten Lichtes misst. Beim **Löschen** wird die gesamte Plattenoberfläche einheitlich erhitzt und alle Teilchen der Metalllegierung in den kristallinen zurückversetzt.

**Magneto-optische Platten:** Beim **Schreiben** wird ein dünner Film einer besonderen Metalllegierung mittels eines Lasers erhitzt, was die Umpolung von einzelnen magnetischen Bezirken mittels eines Magneten, der auf dem Schreib-Lesekopf angebracht ist, erleichtert. Das Metall kühlt schnell ab und die Information bleibt erhalten. Beim **Lesen** nützt man die Eigenschaft aus, dass die Magnetisierungsrichtung der magnetischen Bezirke die Schwingungsebene des polarisierten Lichtstrahles des Lasers an der Plattenoberfläche entweder nach links oder rechts dreht. Das **Löschen** erfolgt wiederum durch allgemeine Erhitzung der gesamten Plattenoberfläche, wobei alle magnetischen Bezirke gleichsinnig ausgerichtet werden.

## Was versteht man unter einem endlichen deterministischen Automaten?

**Endlich:** Die Anzahl der Zustände des Automaten ist begrenzt

**Deterministisch:** Aus der Eingangsinformation und dem Vorzustand lässt sich stets eindeutig vorherbestimmen, in welchen Zustand der Automat wechseln wird.

## Was ist der Schmitt-Trigger?

Ein Schmitt-Trigger ist ein Komparator, bei dem Ein- und Ausschaltpegel um die Schalhysterese  $U_{HST}$  differieren. Er wird mit einem analogen Signal beliebiger Kurvenform angesteuert. Überschreitet die Eingangsspannung eine vorgegebene obere Schaltschwelle, so nimmt der Ausgang einen binären Wert an. Unterschreitet die Eingangsspannung eine untere Schaltschwelle, nimmt der Ausgang den anderen binären Wert an.

## Definieren Sie die Hysterese beim Schmitt-Trigger

$U_{HST} = U_{SO} - U_{SU} \rightarrow$  Hysterespannung = Ober Schaltschwelle – Untere Schaltschwelle

Die Hysterese beim Schmitt-Trigger ist die Differenz zwischen Ein- und Ausschaltpegel. Durch die Hysterese kann ein Schmitt-Trigger auch bei langsamen Umschaltzeiten nicht schwingen.

## Die ALU generiert zwei Signale N und Z. Geben Sie an, was diese bedeuten!

**Z** = Nullbit (Zero). Zeigt an, ob das Ergebnis der vorhergehenden Rechenoperation Null ist.

N= Negativbit. Wird gesetzt, wenn das Ergebnis der Operation als negative Zahl zu interpretieren ist.

### Was ist ein Zero-Crossing-Detector?

Ein ZCD stellt einen Nullspannungsschalter dar. Er schaltet ausgangsseitig binär um, wenn die Eingangsspannung den Wert Null durchläuft. Von besonderem Interesse ist dabei, dass das binäre Ausgangssignal mit minimaler zeitlicher Verzögerung den Nulldurchgang der Eingangsspannung anzeigt, um damit Schaltvorgänge auszulösen, die an die Bedingung des Nulldurchgangs der Eingangsspannung geknüpft sind.

### Erläutern Sie die Funktion der anti-parallelen Diodenschaltung im Gegenkopplungszweig des Operationsverstärkers eines Zero-Crossing Detectors

Um einen gesättigten Betrieb der Ausgangsstufe des Operationsverstärkers und eine dadurch bedingte zusätzliche Schaltverzögerung zu vermeiden, schaltet man zwischen Ein- und Ausgang eine zweiseitige Begrenzerschaltung (**BG**) bestehend aus einer anti-parallelen Diodenschaltung zur Begrenzung der negativen und positiven Ausgangsspannung.

### Was passiert mit dem Vorzeichenbit bei einem arithmetischen Rechtsshift?

Das höchstwertige Bit (MSB) wird einfach wieder eingefügt und so das Vorzeichen erhalten.

### Was passiert mit dem Vorzeichenbit bei einem arithmetischen Linksshift?

Das Vorzeichenbit wird nicht verändert. Das Bit vor dem MSB geht verloren.

### Was versteht man unter Context Switch Time?

Prozessorwechselzeit – ist die für die Berechnung einer Scheduling Entscheidung notwendige Zeit und sollte minimal sein.

### Erklären Sie die Funktionsweise vom Round-Robin Scheduling

Das Rundlauf-Verfahren gewährt allen Prozessen nacheinander für jeweils einen kurzen Zeitraum Zugang zu den benötigten Ressourcen. Man nennt dies auch Arbitrierung.

### Nennen Sie alle Prozesszustände in einem Betriebssystem und zu welchem Zustand sie jeweils führen können.

CREATED → READY  
READY → SUSPENDED, RUNNING, DEAD  
RUNNING → READY, BLOCKED, DEAD  
BLOCKED → RUNNING, DEAD; SUSPENDED, READY  
SUSPENDED → READY, BLOCKED  
DEAD

### Wodurch unterscheiden sich bei USB die Connect Erkennung von Full-Speed und Low-Speed Geräten?

Beim USB-Gerät ist auf der Upstream-Seite eine Leitung über einen 15 kΩ Widerstand mit +3,3 V verbunden. Bei Fullspeed Geräten ist dies die D<sup>+</sup> Leitung, bei Low-speed Geräten die D<sup>-</sup> Leitung.

### Geben Sie die Formel für die durchschnittliche Speicher-Zugriffszeit an. Die Trefferrate soll mit h angegeben werden.

$$T_{\text{eff}} = h * T_{\text{(cache)}} + (1-h) * T_{\text{(main)}}$$

### Was versteht man unter dem Cycle Stealing Verfahren?

Das CSV ist ein Refresh Verfahren für DRAMs. Um eine lange Blockierung von Speicherzellen zu vermeiden, kann man die Refreshvorgänge für einzelne Teile des Speichers getrennt durchführen.

### Welche anderen Refresh-Verfahren gibt es?

**Burst-Refresh:** Der Normalbetrieb wird unterbrochen um bei allen Speicherzellen ein Refresh durchzuführen. Während dieser Zeit ist kein Zugriff auf den Speicher möglich.

**Transparent Refresh:** Bei diesem Verfahren synchronisiert man Refreshcontroller mit dem Prozessor um die laufenden Prozesse nicht anhalten zu müssen.

### Was versteht man unter Interlocking beim Pipelining Konzept?

Beim Pipelining wird der Mechanismus gestoppt sobald die Decoding Unit einen Sprungbefehl erkennt. Die Freigabe erfolgt erst sobald die Zieladresse des Sprunges ermittelt worden ist oder der Program Counter erneuert wurde.

### Geben Sie die Formel für die Kondensatorladung / Entladung an!

Aufladen:  $U_{(t)} = U_0 * (1 - e^{-(t/T)})$   
 Entladen:  $U_{(t)} = U_0 * e^{-(t/T)}$

### Welche Arten von Speicherbausteinen kennen Sie?

**Statisches RAM:** Information wird in Latches gespeichert. Sehr kurze Zugriffszeiten. Hoher Preis. Kaum hoch integrierbar

**Dynamisches RAM:** Information wird in Kondensatoren gespeichert. Refresh Cycle (Burst Refresh, Cycle Stealing, Transparent Refresh)

### Was sind „0-aktive Eingänge“ bei ICs (integrierten Schaltungen)?

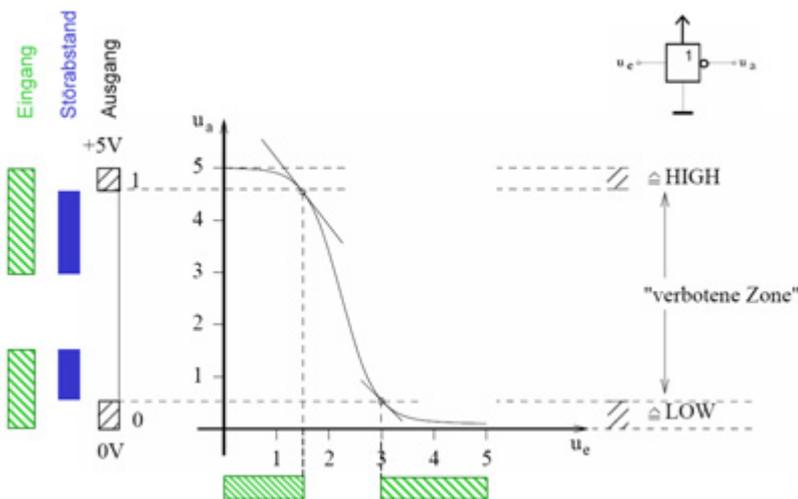
Ein 0-aktiver Eingang hat den Ruhezustand logisch 1 und wird bei logisch 0 aktiv.

### Welche Funktion hat der ENABLE Eingang bei einem Decoder?

Der Enable-Eingang legt fest, wann das Eingangssignal decodiert werden soll und wann nicht. Hat dieser Eingang den Wert logisch 1, dann wird das Eingangssignal decodiert, ansonsten (logisch 0) liegt allen Ausgängen logisch 0 an.

### Geben Sie die Übertragungskennlinie $U_a = f(U_e)$ eines Inverters an. Wo liegt die verbotene Zone?

Die verbotene Zone liegt in jenem Bereich, in dem die Kennliniensteigung dem Betrag nach größer gleich 1 ist.



### Welche drei Sichtweisen gibt es beim Entwurf elektronischer Schaltungen laut dem Y-Modell?

- Verhalten
- Struktur
- Geometrie

### Welche drei Arbeitsschritte sind zur Erstellung eines Moduls bzw. einer Komponente in VHDL erforderlich?

- Schnittstellenbeschreibung
- Architektur
- Konfiguration

### Von welchen zwei Faktoren hängen Zustandsänderungen bei Schaltwerken ab?

Von den Eingangswerten und dem aktuellen Zustand des Schaltwerks

### Wozu dient die Ziehkapazität bei einem Quarzoszillator?

Die Ziehkapazität eines Quarzoszillators dient dem Abgleich der Resonanzfrequenz bei Serienresonanz und wird mit dem Quarz in Serie geschaltet.

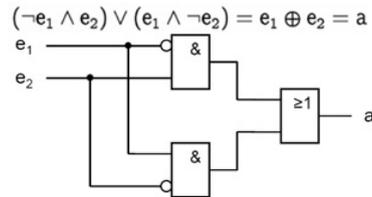
### Was versteht man unter „Fan In“?

Unter „Fan-In“ versteht man die maximale Anzahl an Gatterausgängen, die an den jeweiligen Gattereingang angeschlossen werden dürfen.

### Was versteht man unter „Fan Out“?

Unter „Fan-Out“ versteht man die maximale Anzahl an Gattereingängen, die an den jeweiligen Gatterausgang angeschlossen werden dürfen.

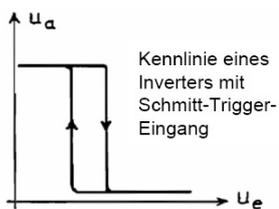
### Zeichnen Sie eine Gatterschaltung für die XOR-Funktion!



### Erklären Sie die Abkürzung ASIC!

ASIC bedeutet „Application Specific Integrated Circuit“. Es handelt sich dabei um einen Funktionspeicher, also einen integrierten Baustein, der zur Speicherung einer Funktion dient.

### Zeichnen Sie die Übertragskennlinie eines invertierenden Schmitt-Triggers!



### Was versteht man unter dem Tastgrad $g$ einer Rechteckimpulsfolge des Taktgenerators eines Rechners?

Der Tastgrad ist in diesem Zusammenhang das Verhältnis von Impulslänge zu Impulsperiodendauer.

$$g = \frac{T_i}{T}$$

### Was versteht man unter dem Scratchpad eines Prozessors?

Das Scratchpad (oder Register File) eines Prozessors ist eine Konfiguration, bei der die ALU (Arithmetic Logic Unit) die Register A und B nur mehr als Zwischenspeicher verwendet.

### Welchen Vorteil hat die Erhöhung der Baudichte bei Mikroprozessoren?

Reduzierung der Wegstrecken  $\rightarrow$  Erhöhung der Prozessorgeschwindigkeit, da sich elektrischer Strom mit etwa 0,7-facher Lichtgeschwindigkeit ausbreitet.

### Nennen Sie einen Vorteil und einen Nachteil von Instruktionen mit variabler Länge im Vergleich zu Instruktionen mit fixer Länge.

Häufig verwendete Maschineninstruktionen können mit kürzeren Bitmustern codiert werden als weniger oft verwendete. Dadurch kann die Länge von Programmen reduziert werden. Dem gegenüber steht jedoch, dass ein Interpreter, der Maschineninstruktionen variabler Länge verarbeiten muss, naturgemäß langsamer arbeitet als ein Interpreter, der lediglich mit Instruktionen fester Länge zu tun hat.

### Welche zwei Möglichkeiten gibt es, die Adressen der Ports für Input / Output-Operationen zu vergeben?

- **Independent I/O-System:** Hauptspeicher und Ports haben völlig voneinander unabhängige Adressen
- **Memory-mapped-I/O-System:** Ports werden so behandelt, als wären sie gewöhnliche Speicherstellen.

### **Was ist ein Trap? Geben Sie ein Beispiel!**

Ein Trap (auch Interrupt) genannt ist eine Unterbrechung der momentan vom Prozessor ausgeführten Programmsequenz, um auf einen im normalen Programmablauf nicht vorgesehen Zustand zu reagieren.

### **Wo werden die Return Adressen bei Unterprozeduraufrufen gespeichert?**

Die Speicherung der Return-Adressen erfolgt auf einem Stack.

### **Warum ist eine hohe Trefferquote der branch prediction Einheit bei modernen Prozessoren so wichtig? Denken Sie daran, dass die Prozessoren eine sehr lange Pipeline verwenden.**

Normalerweise werden der Pipeline die Daten sequentiell, wie sie kommen, zugeführt. Wird aber ein Sprung ausgeführt, sind alle Daten, die nach diesem Sprung in die Pipeline geraten sind, wertlos und müssen verworfen werden. Ziel der branch prediction ist ein möglichst frühes Erkennen eines Sprungbefehls und Erkennen seiner Sprungzieladresse, damit gleich die Daten der Zieladresse dem Sprungbefehl in die Pipeline folgen können.

### **Was versteht man unter logischer Parallelität von Prozessoren?**

Logische Parallelität von Prozessoren liegt dann vor, wenn mehrere Prozesse scheinbar gleichzeitig auf einem einzelnen Prozessor ausgeführt werden.

### **Nennen Sie einen Nachteil von asynchronen Methoden zur Interprozesskommunikation (IPC) im Vergleich zu synchronen Methoden.**

Der Prozess hat zu jeder Zeit damit zu rechnen, auf Grund eines eingehenden Signals in der Ausführung unterbrochen zu werden.

### **Nennen Sie zumindest drei Bestandteile des Prozessdeskriptors!**

- Prozessidentifikation (Prozess ID)
- Process State Information (Prozesszustand, Priorität, Registerinhalte)
- Process Control Information (Besitzer, Zugriffsrechte...)

### **Durch welche drei im Buch erläuterten Arten der Parallelverarbeitung kann die Performance eines Prozessors gesteigert werden?**

- Vektorverarbeitung
- Superskalare Verarbeitung
- Instruction-Pipelining

### **Was ist die Aufgabe eines Linkers?**

Der Linker fügt die als Relocatable Object Code vorliegenden Programmmodule, welche aus entsprechendem Sourcecode kompiliert worden sind, zu einem ausführbaren Programm zusammen.

### **Was versteht man unter dem Program Counter bei einem Mikroprozessor?**

Der PC ist ein Prozessorregister, welches die Speicheradresse speichert, an welcher der auszuführende Maschinencode gespeichert ist.

### **Welche zwei Möglichkeiten gibt es, die Adresse einer Interrupt Service Routine (ISR) zu bestimmen?**

- Fixe Zuordnung
- Interruptvektor

### **Welche zwei Komponenten benötigt jeder thread für sich alleine?**

- Registersatz des Prozessors
- Thread-spezifische Daten (lokale Variablen, Rücksprungadressen, etc.)

### **Nennen Sie mindestens zwei Probleme bei der Verwendung von Threads!**

- Mehrere Threads in einem Prozess verwenden denselben Adressraum. Ein Prozess kann daher durch Speicheroperationen andere Threads in demselben Prozess oder auch den Prozess selbst zum Absturz bringen.

- Um das Prozessverhalten mit den threads in Einklang zu bringen, ist meistens eine gesonderte bzw. zusätzliche Behandlung notwendig.
- Bei der Verwendung von Libraries, die nicht für den Einsatz in einem mehrfach genutzten Adressraum ausgelegt sind, können Probleme auftreten.

### Was versteht man unter einem System Call?

System Calls sind Betriebssystemaufrufe, mit Hilfe derer ein Prozess in der Lage ist, die verschiedensten Funktionen des Betriebssystems in Anspruch zu nehmen.

#####

Zusammenfassungsfragen von Klausur

#####

### Was versteht man unter Scheduling?

Als Scheduling bezeichnet man die Verteilung der Aufgaben eines Betriebssystems auf die tatsächlich vorhandenen Ressourcen. Scheduling wird in mehreren Ebenen durchgeführt:

- Job-Scheduling
- Prozess-Scheduling
- Thread-Scheduling

#### Job-Scheduling

Im Hochlastfällen ist es oft günstig, die angespannte Lage zu entschärfen und einige der konkurrierenden Prozesse zur Gänze auf Eis zu legen. Dadurch wird zB der System-Overhead beim Prozess-Scheduling verringert und damit die Nutzkapazität erhöht. Ziel des Job-Scheduling ist es also, dem Prozess-Scheduler gut bewältigbare Aufgaben zuzuteilen.

#### Prozess-Scheduling

Aufgabe des Prozess-Scheduling ist das Verteilen der Prozesse (*Context Switch*) auf die Prozessoren. Es muss dafür gesorgt werden, dass jeder Prozess, der sich im Zustand READY befindet, in bestimmten Abständen und für eine bestimmte Zeit einen Prozessor zugeteilt bekommt, also in den Zustand RUNNING wechselt. Prozesse in anderen Zuständen können ignoriert werden, da sie keinen Bedarf nach einem Prozessor haben. Wenn die Zuteilung sehr schnell geschieht, entsteht für jeden Prozess die Illusion einer kontinuierlichen Exekution.

#### Forderungen an Scheduling Algorithmen

- Fairness - Verteilung der Prozessorkapazität soll gerecht sein
- Effizienz – Prozessoren sollten möglichst optimal ausgelastet werden
- Durchsatz (Throughput) – Anzahl der verarbeiteten Jobs sollte maximiert sein
- Antwortzeiten (Response Time) – Prozesse sollten raschen Response liefern
- Prozessorwechselzeit (Context Switch Time) – Die Prozessorwechselzeit ist die für die Berechnung einer Scheduling-Entscheidung notwendige Zeit und soll minimal sein.

#### Scheduling-Methoden

Das Hauptproblem des Scheduler ist, dass er keine Vorhersagen über das Verhalten der Prozesse zur Verfügung hat. Daher muss er seine Entscheidungen heuristisch treffen. Wegen des geringeren Overheads werden primitive Methoden bevorzugt.

**First Come First Served (FCFS):** Eine non preemtive Technik. Werden heute nicht mehr angewendet.

**Round Robin Scheduling (RRS):** Hier bekommt nacheinander jeder READY Prozess für ein (kurzes) Zeitintervall (*Quantum* oder *Time Slice*) einen Prozessor zugeteilt. Läuft das Quantum des Prozesses ab, wird er ans Ende der READY Liste wieder eingereiht. Kritisch für das RRS ist die Länge des Time Slices. Ist sie zu klein, reduziert der Scheduling Overhead die nutzbare Prozessorleistung unzulässig. Ist sie zu groß, sind die Antwortzeiten unbefriedigend.

**Static Priority Scheduling (SPS):** Prozessen wird eine Priorität zugeordnet und die Prozessorkapazität wird entsprechend aufgeteilt. RUNNING ist immer jener Prozess mit höchster Priorität. Wechselt ein höherer Prozess nach READY, nimmt der Scheduler dem aktuellen RUNNING Prozess den Prozessor weg und bringt den neuen Prozess zur Ausführung. Diese preemtive Strategie ist einfach zu implementieren. Es kann aber passieren, dass ein niedrigprioriter Prozess nie einen Prozessor zugeteilt bekommt (=Starvation).

**Dynamic Priority Scheduling (DPS):** Eine adaptive Scheduling Strategie. Durch dynamische Änderung von Prioritäten wird sichergestellt, dass kein Prozess auf der Strecke bleibt. Dies kann zB durch Zuteilung eines Soll-Services passieren. Weitere Scheduling Verfahren: **Shortest Job First:** Basiert auf Schätzung der Ausführungszeit eines Prozesses.

## Thread-Scheduling

Das Threads Scheduling ergänzt das Process Scheduling um eine untergeordnete Ebene.

## Message Passing

Im Gegensatz zu Semaphoren handelt es sich beim Message Passing um einen Mechanismus, der eine Kommunikation im Sinne eines Datenaustausches zwischen Prozessen erlaubt. Üblicherweise desponiert dabei ein Senderprozess Nachrichten an einem Message Exchange, von wo sie ein Empfängerprozess abholen kann. Praktisch wird dies wieder durch eine Anzahl von System Calls ermöglicht.

## Was versteht man unter einem Hazard?

Hazards sind Fehler bei digitalen Schaltungen, die aufgrund eines undefinierten Zustands auftreten und zu falschen Werten führen.

## Welche Arten von Pipeline-Hazards gibt es?

- Strukturelle Hazards: Mehre Pipeline Stufen benötigen die selbe Ressource. Maßnahmen: Geeignete Architektur
- Control Hazards: Nachfolgebefehl hängt vom Ausgang des Sprungs ab. Maßnahmen: prediction, delayed branch
- Data Hazards: Berechnung erfordert Ergebnis des Vorgängerbefehls. Maßnahmen: forwarding, Code-Optimierung

## Halbaddierer

Zählt zwei einstellige Binärzahlen zusammen. Mögliche Überträge aus Vorstelle werden nicht berücksichtigt.

## Volladdierer

Addier Binärzahlen mit n Stellen unter Berücksichtigung des Übertrags.

## Codierer

Die Schaltung eines Codierers besitzt bei n Eingängen genau  $m = \lg(n)$  Ausgänge. Es kann immer nur ein Eingang aktiv sein. Der Codierer wandelt die Bitfolge an den Eingängen in eine Binärzahl an den Ausgängen um. Dh: e7 aktiv am Eingang ergibt 111 am Ausgang.

## Multiplexer

Der Multiplexer ist eine Erweiterung des Codierers. Er wählt aus m binären Eingängen jenen aus, dessen Nummer mit der Zahl übereinstimmt, die an den Steuervariablen anliegt und schaltet dessen Information unverändert an den Ausgang durch.

## Demultiplexer

Beim Demultiplexer wird durch die Steuereingänge bestimmt, auf welchen Ausgang die Information des Eingangs durchgeschaltet werden soll.

## Operationsverstärker

Häufig an der Schnittstelle zwischen analogen und digitalen Schaltkreisen. Charakteristisch ist eine sehr hohe Leerlaufspannungsverstärkung.

## Komparatoren für analoge Signale

Komparatoren für analoge Signale sind Schaltungen, mit deren Hilfe festgestellt werden kann, ob und zu welchem Zeitpunkt die momentane Amplitude eines Signals gleich einem konstanten oder zeitlich veränderbaren Referenzsignal ist. Das Ausgangssignal kennt nur zwei diskrete Signalzustände abhängig davon, ob die Eingangsspannung größer oder kleiner als die Referenzspannung ist.

## Komparatoren für digitale Signale

In der Digitaltechnik werden häufig Vergleiche für digitale Signale benötigt, um zu entscheiden, ob zwei binäre Variablen x und y gleich oder ungleich sind. Die Arbeitsweise des Komparators für digitale Signale kann durch die Äquivalenzfunktion beschrieben werden.  $X \equiv Y = Z$

## Torschaltungen

Unter Torschaltungen versteht man Netzwerke, die aus einem Signal für eine vorgegebene Zeit einen zeitlichen Anteil des Signals „herausschneiden“. Man bezeichnet solche Netzwerke auch als Zeitfilter. Die Arbeitsweise solcher Schaltungen hängt allein von der Zeitbedingung ab, die über einen Steuereingang (control input) binär vorgegeben wird.

## Fensterdiskriminator

Mit Hilfe eines Fensterdiskriminators kann festgestellt werden, ob der Wert der Eingangsspannung  $U_e$  unterhalb, innerhalb oder oberhalb des durch die Spannungsgrenzwerte  $U_{GO}$  und  $U_{GU}$  festgelegten Fenster liegt.

## Univibrator

Ein Univibrator stellt eine Schaltung dar, die ausgangsseitig zwei Zustände annehmen kann. Von diesen beiden Zuständen ist nur ein Zustand stabil, der andere Zustand ist quasi-stabil und kann nur für eine vordefinierte Zeit, die Verweilzeit  $T_0$ , nach einem eingangsseitigen Triggerimpuls angenommen werden. Es lassen sich Schaltzeiten von einigen Sekunden bis zu einigen Minuten realisieren.

## Quarzoszillator

Ein Quarzoszillator ist eine elektronische Schaltung zum Erzeugen von Schwingungen, die sich eines Schwingquarzes als frequenzbestimmendes Bauelement bedient. Im engeren Sinne ist ein Quarzoszillator eine fertig aufgebaute Oszillatorschaltung, die zusammen mit dem frequenzbestimmenden Quarz in einem Gehäuse eingebaut ist und als Standardbauteil erhältlich ist. Quarzoszillatoren sind in ihrer Frequenz sehr genau.

## Sägezahngenerator

Von den gegengekoppelten Schaltungen mit Operationsverstärkern eignet sich besonders der Miller-Integrator zur Erzeugung sägezahnförmiger Spannungsverläufe. Durch periodisches Schließen des Schalters  $S$  und die dadurch entstehende Entladung der Kapazität  $C$  entsteht ein sägezahnförmiger Verlauf.

## Tristate Outputs

Tristate Outputs stellen abschaltbare Ausgänge dar. Das Ein- und Abschalten des Ausganges erfolgt dabei über einen eigenen Steuereingang. Somit gibt es zu den normalen Ausgangszuständen logisch 0 und logisch 1 noch den dritten Zustand mit abgeschaltetem, hochohmigen Ausgang.

## Open-Collector-Schaltungen

Im Gegensatz zu Tristate Outputs können hier mehrere Ausgänge aktiv sein. Es werden Ausgänge mit offenem Kollektor-Ausgang eingesetzt. Diese besitzen einen npn-Transistor am Ausgang, wobei der Emitter an Masse liegt und der Kollektorausgang unbeschaltet an den Ausgang des Gatterbausteins herausgeführt wird. Solche Ausgänge kann man nun parallel schalten und mit einem gemeinsamen Kollektorwiderstand beschalten.

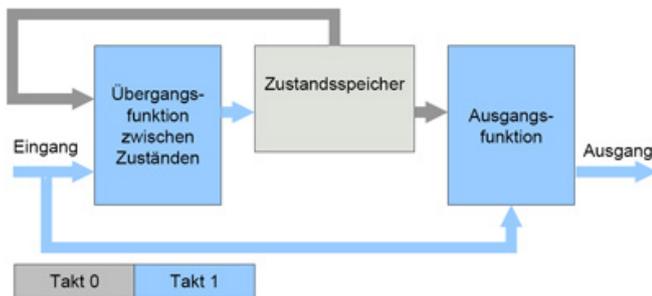
## Moore Schaltwerk

Beim Moore Schaltwerk können die Zustände in beliebiger Reihenfolge aufeinanderfolgen. Ein Zustand kann mehrere Folgezustände haben. Die Eingangssignale entscheiden, welcher dieser Folgezustände tatsächlich eingenommen wird. Beim Moore Schaltwerk sind die Ausgänge im einfachsten Fall mit den Latch Ausgängen identisch. Meist bildet aber eine Ausgangsfunktion die eigentlichen Ausgänge aus den Latch Ausgängen, um damit Latches einzusparen. Beim Moore Schaltwerk hängen die Ausgänge nur vom momentanen Zustand ab. Damit können die Eingänge die Ausgänge nicht direkt beeinflussen. Für Ausgangsänderungen ist ein Zustandswechsel notwendig, der immer erst bei der nächsten Taktrate erfolgen kann.



## Mealy Schaltwerk

Das Mealy Schaltwerk unterscheidet sich vom Moore Schaltwerk nur durch eine andere Ausgangsfunktion. Damit lassen sich Ausgangsreaktionen erreichen, die mit einem Moore Schaltwerk nicht möglich sind. Beim Mealy Schaltwerk hängen die Ausgänge nicht nur vom momentanen Zustand (Ausgänge der D-Latches) ab, sondern auch von den Eingängen. Für jeden Zustand sind – gesteuert von den Eingängen – verschiedene Ausgänge möglich.



## Prozessor

Ein Prozessor ist ein Schaltwerk zur Verarbeitung von Daten, das aus Rechenwerk und Leitwerk besteht.

## Rechenwerk

Funktionseinheit innerhalb eines digitalen Rechensystems, die Rechenoperationen ausführt.

## Leitwerk

Funktionseinheit eines digitalen Rechensystems, die

- die Reihenfolge steuert, in der die Befehle eines Programms ausgeführt werden
- diese Befehle entschlüsselt und modifiziert
- die für die Ausführung benötigten digitalen Signale abgibt

## Woraus ergibt sich die Cache-Performance

- Hit-Time
- Hit-Rate
- Miss-Penalty

## Was versteht man unter Swapping?

Beim Swapping (auch Roll-In/Roll-Out) werden Prozess-Images als Ganzes zwischen dem physikalischen und dem virtuellen Speicher hin- und her bewegt. Diese einfache Strategie benötigt sehr wenig Hardwareunterstützung, allerdings dauert das Kopieren des gesamten Images sehr lange. Dazu kommt noch die Speicherverschwendung durch die Fragmentation und die etwas unflexible Memory Protection.

## Was versteht man unter Paging?

Aufteilung des virtuellen Adressraums in gleich große Teile, die vom Programm unabhängig sind. Die Aufteilung erfolgt transparent

## Was versteht man unter Segmentierung?

Aufteilung des virtuellen Adressraums in unterschiedlich große Teile, die direkt auf Teile des Programms abgebildet werden können. Die Aufteilung erfolgt nicht transparent.

## Was versteht man unter einem Cache-Speicher? Wie ist er aufgebaut?

Ein Cache ist ein sehr schneller, nahe am Prozessor liegender Speicher, der sehr teuer und klein ist. Er dient als Zwischenspeicher für häufig benötigte Daten. Ein Cache ist transparent. Sind die angeforderten Daten im Cache (**Cache Hit**), so können sie sehr schnell zum Prozessor geliefert werden. Sind die Daten nicht im Cache (**Cache Miss**), werden sie aus dem Hauptspeicher nachgeladen. Die **Cache Hit Rate** gibt Auskunft über den Erfolg der Zugriffe.

Der Cache besteht aus mehreren Speicherzellen (Cache lines)

### Welche Arten von Caches kennen Sie?

- **Voll assoziativer Cache:** Daten müssen erst ersetzt werden, wenn der Cache voll ist. Suche nach vorhandenen Daten dauert länger.
- **Direct Mapping:** Jede Adresse wird auf eine bestimmte Cache Line abgebildet. Kommt ein neuer Datenwert, muss der alte Wert an dieser Adresse ersetzt werden. Datensuche sehr schnell

### Was versteht man unter VHDL?

Die Very (High Speed Integrated Circuit) Hardware Description Language ist eine Sprache für Hardwaredesign.

### Welche Entwurfsebenen gibt es in VHDL?

- Algorithmische Ebene
- Register-Transfer Ebene
- Logikebene

### Welche Schaltungsfamilien kennen Sie?

**TTL (Transistor-Transistor Logic):** Großer Fan-Out. Wegen hoher Gatterlaufzeit (propagation delay) heute kaum noch verwendet.

**ECL (Emitter Coupled Logic):** Geringe Gatterlaufzeit. Große Verlustleistung und hohe Kosten

**MOS (Metal Oxid Semiconductor):** Bei einfachen Gattern wird diese Technik selten eingesetzt, allerdings bei hoch integrierten Schaltungen

**CMOS (Complementary MOS):** Geringste Leistungsaufnahme. Großer Betriebsspannungsbereich. Standard im PC Bereich

### Erklären Sie die Abkürzung VLSI!

Bei der *Very Large Scale Integration* werden nach dem Chipdesign alle Bauteile und Verbindungsleitungen gleichzeitig auf einem Siliziumplättchen hergestellt.