

Klausur Rechnertechnologie Wintersemester 07/08.

Allgemeine Hinweise/Tipps:

Stichwortverzeichnis anlegen, Assembler wissen, meistens kommt aus jedem Kapitel etwas dran

1. Die Prozessoren können in 3 Gattungen eingeteilt werden. Nennen sie die typischen Einsatzgebiete und begründen sie in wieweit ein Wechsel zwischen den verschiedenen Gattungen sinnvoll ist.

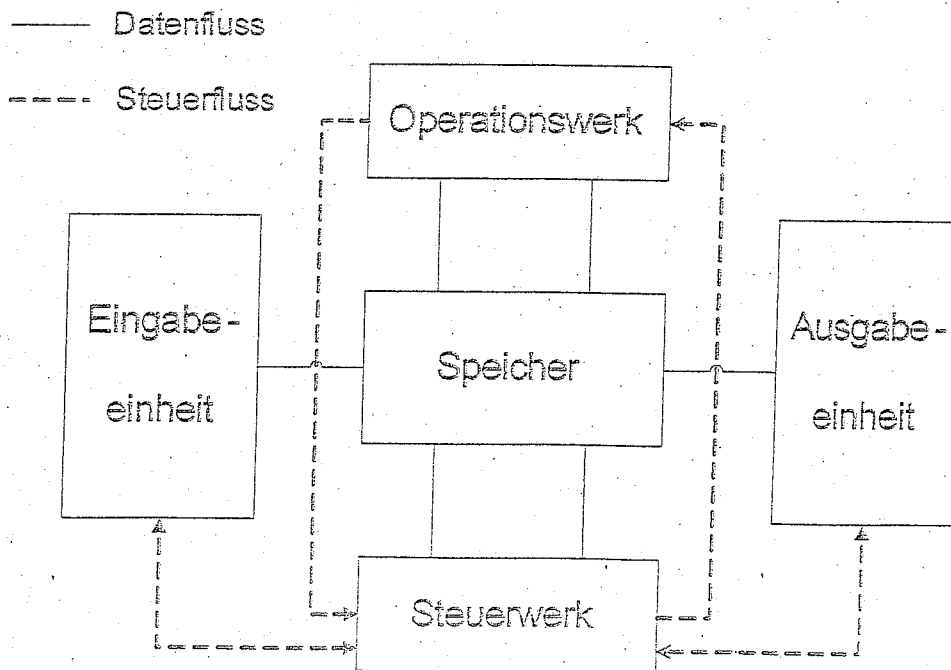
Die 3 Gattungen sind: Mikroprozessoren, Mikrocontroller und Signalprozessor.

- Mikroprozessoren: General Purpose Prozessor für Anwendungen die Betriebssystemunterstützung, grafische Interfaces und Massenspeicher erfordern. Bereitstellung großer Rechenleistung und großer Arbeitsspeicher. Verwendung eines Motherboards erforderlich
- Mikrocontroller: Anwendungen mit dezidierten Algorithmen, die keine Betriebssystemunterstützung erfordern, Bereitstellung von analogen und digitalen Schnittstellen, geringe Rechenleistung erforderlich. Implementation von funktionsfähigen Gesamtsystemen mit geringem Aufwand.
- Signalprozessor: Gleicher Anwendungsbereich wie Mikrocontroller aber mehr Rechenleistung.

Nur Wechsel zwischen Mikrocontrollern und Signalprozessoren sinnvoll! Gleicher Anwendungsbereich der beiden Prozessorgattungen.

2. Von-Neumann-Architektur.

- a) Skizzieren Sie den Aufbau in Form eines Blockdiagramms (wie im Skript) und benennen sie die einzelnen Komponenten.



b) Erläutern Sie das Funktionsprinzip anhand der Befehlsbearbeitung.

Die Befehlsbearbeitung erfolgt nach dem 2-Phasen-Prinzip (kann u.U. noch feiner auf gegliedert werden).

- Befehlshol- und Interpretationsphase (jeweils 1 Befehl wird geladen)
- Befehlsausführungsphase (jeweils ein Ergebnis wird bestimmt) → Prinzip: SISD (single instruction, single data)

c) Und nennen Sie die Vorteile.

- Einfacher Aufbau, - Ausnutzung des Speichers für Befehle, Adressen, Daten, - Generierung und Änderungen von Programmen während der Laufzeit, - volle Verfügbarkeit des gesamten Speicherbereichs.

3. In Prozessorsystemen werden unterschiedliche Arten elektronischer Speicher eingesetzt.

a) Erläutern Sie die Funktionsweise einer dynamischen und einer statischen RAM-Speicherzelle auf Transistorebene anhand jeweils eines elektrischen Schaltbildes und nennen Sie die wesentlichen Unterschiede der beiden Speichertypen.

(Schaltbilder!)

Unterschiede:

- Statischer Speicher hat kurze Zugriffszeiten und der dynamischer hat eine aufwendige Ansteuerung
- Statischer hat eine hohe Verlustleistung und dynamischer eine geringe
- Statischer hat eine relativ kleine Speichergröße und dynamischer hat große Speicherbereiche
- Dynamischer Speicher hat eine Notwendigkeit für Refresh-Ansteuerung

b) Was versteht man unter einem „Masken-ROM“ und was sind seine Einsatzbereiche.

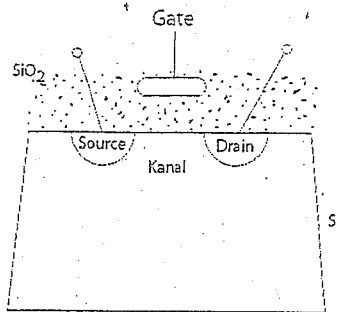
Als Masken-ROM bezeichnet man Festwertspeicher, bei denen die Information im Rahmen des Fertigungsprozesses fest angelegt werden, d.h. sie sind nur lesbar und können im normalen Betrieb nicht beschrieben werden.

Einsatzbereiche: Prozessorexterner Speicher z.B. BIOS des PC

c) Was versteht man unter einem „FL-ROM“ und was sind seine Einsatzbereiche.

(Fussible Link)

- d) Erläutern Sie die Funktionsweise einer UV-löschbaren und einer elektrisch löschbaren ROM-Speicherzelle anhand jeweils einer grafischen Darstellung der Speicherzellen (Schnitt-Darstellung des Silizium-Wafers).



UV-Löschabr: Das zum Löschen verwendete UV-Licht bewirkt eine großflächige Ionisation des Halbleitermaterials, so dass die vorher aufgebrachte elektrische Ladung des Floating Gate des Transistors wieder verlassen kann. Das Bitmuster ist dadurch gelöscht und das EPROM in seinem ursprünglichen Zustand zurückversetzt.

El-Löschbar: Beim Programmiervorgang wird auf dem Gate eine Ladung gespeichert (der Transistor sperrt). Beim Löschen wird diese Ladung wieder entfernt, indem durch einen hohen Spannungspuls die Ladung der Sperrschicht in den Ursprungszustand versetzt wird.

Spannungspuls die Ladung der Sperrschicht in den Ursprungszustand versetzt wird.

4. Schnittstellen werden in Rechnersystemen durch Schnittstellen-Controller unterstützt, die nach Programmierung weitgehend selbstständig die Daten Übertragung durchführen. Hierbei unterscheidet man die Betriebsarten „Polling (Abfragebetrieb)“ und „Interrupt-Betrieb“.

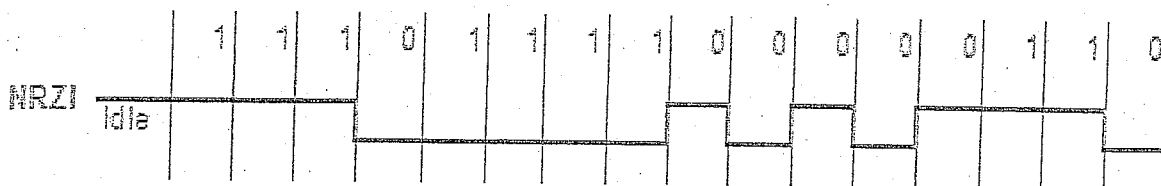
- a) Erläutern Sie diese Funktionsweise dieser beiden Betriebsarten und geben Sie die Unterschiede und Einsatzbereiche an.

Polling: Die serielle Schnittstelle wird ständig darauf untersucht, ob ein vollständiges Zeichen eingelaufen ist.

Interrupt-Betrieb: Die serielle Schnittstelle teilt selbst mit, dass sie ein vollständiges Zeichen empfangen hat.

Sie werden hauptsächlich da eingesetzt, wo von der Hardware der Status benötigt wird oder um Software abzufragen, z.B. gerade in PCs.

- b) Geben Sie die NRZI-Codierung der folgenden Bitfolge ausgehend vom Zustand IDLE = 1 an. (als Bitfolge oder auch grafisch wie im Skript)
111011110000110 + Warum verwendet man die NRZI-Codierung?



Damit man leichter die Taktfrequenz ermitteln kann (z.B. noch zusammen mit dem NRZ-Verfahren)

5. Im Speicher eines elektronischen Kilowattstunden-Zählers mit dem Mikroprozessor 8086 liegt eine Tabelle von 1000 Messwerten (16 Bit) des durch den Zähler fließenden Stroms. Je ein Wort stellt einen Messwert dar. Die Tabelle liegt im Datensegment. Die Messwerte seien bereits eingelesen. Das Register BX enthält die Anfangsadresse der Tabelle. Als eine Teilberechnung zur Bestimmung des Energieverbrauchs muss die Tabelle zunächst korrigiert werden. Da der Sensor in bestimmten Bereichen konstante Messabweichungen hat, muss jeder Messwert wie folgt korrigiert werden.

cmp AX, 1000H JB keine-Korrektur cmp AX, 5000H JA grossen

Ist der Messwert kleiner als 1000H, ist keine Korrektur erforderlich.

Liegt der Messwert zwischen 1000H und 5000H, muss der Wert 0CB3H addiert werden.

Ist der Messwert größer als 5000H, muss der Wert 02A5H subtrahiert werden.

Schreiben Sie ein Programm in Assembler für den Mikroprozessor 8086 oder Nachfolger, das die entsprechenden Messwerte korrigiert. Die korrigierte Tabelle soll sich anschließend an der gleichen Stelle im Speicher befinden. Es ist nur dieser Programmabschnitt zu schreiben.

6. In einem Computersystem soll eine Funktionszuordnung $y = f(x)$ durch einen Tabellenzugriff analog zum Befehl XLAT, aber ohne dessen Beschränkung auf nur 256 Werte, realisiert werden. Hierzu soll ein Assembler-Unterprogramm für den Prozessor Pentium geschrieben werden, dass den Tabellenzugriff über eine Integer-Variable A ermöglicht. In einem Hochsprachenprogramm werden daher ein Feld aus Integer-Werten (die Tabelle) und eine Variable A definiert. Die Tabelle wird im Hochsprachenprogramm bereits mit Funktionswerten belegt. Beim Aufruf des Unterprogramms werden die Adressen des Feldanfangs und der Variable A an das Assemblerprogramm übergeben. Beim Aufruf des Assemblerprogramms sieht der Stack wie folgt aus:

Zeiger auf Feld
Zeiger auf int A
Rücksprungadresse

Jeder Stack-Eintrag umfasst 4 Byte. Das AP soll also analog zum Befehl XLAT den Wert der Integer-Variablen A zur Startadresse des Feldes addieren, auf diese Adresse im Feld zureifen und den Wert der Variablen A durch den ausgelassenen Tabelleneintrag ersetzen.

Notieren Sie die Assemblerbefehle zur Realisierung dieses Tabellenzugriffs mit Umcodierung.

7. ?

8. ?

Aufgabe 1

Rechner lassen sich nach unterschiedlichen Architektur Prinzipien aufbauen, die für verschiedene Anwendungszwecke konzipiert wurden.

1.1 Welche Auswirkungen sind für den Anwender sichtbar, wenn die zugrunde liegende Architektur der von-Neumann-Architektur entspricht?

- Gemeinsamer Speicher für Daten und Programme
- Befehlsverarbeitung erfolgt nach 2-Phasen Prinzip:
 - Befehlshol- und Interpretationsphase (jeweils 1 Befehl wird geladen)
 - Befehlsausführphase (jeweils 1 Ergebnis wird bestimmt)=> SISD (single instruction, single data)
- Langsamer Speicherzugriff
- Daten können nur hintereinander gelesen werden => nicht gleichzeitig!!
- Während dem Programm sind Bibliotheken nachladbar
- Laden der Programme an x-beliebigen Stellen im Arbeitsspeicher

1.2 Welche Auswirkungen sind zu beobachten, wenn eine Harvard-Architektur zum Einsatz kam?

- Verwendung von getrennter Speicher für Daten und Programme
- Schneller Zugriff auf Prozessorexterne Daten
- In einem Zyklus können 2 Daten gleichzeitig gelesen werden

1.3 In welcher Prozessorgattung (Mikroprozessor, Mikrocontroller, Signalprozessor) ist die jeweilige Rechenarchitektur überwiegend anzutreffen? Begründung!

- von Neumann Architektur: alle Mikroprozessoren, ein Teil der Mikrocontroller
- Harvard Architektur: bei einigen Mikrocontrollern und allen Signalprozessoren.

Da Signalprozessoren eine höhere Rechenleistung bereitstellen sollen, muss die Harvard Arch. verwendet werden, da bei dieser mehrere Daten parallel verarbeitet werden können.

Bei Mikrocontroller können beide Architekturen vorkommen. Falls der Mikrocontroller 2 Daten in einem Zyklus verarbeiten muss, dann wird die Harvard Architektur verwendet, falls es jedoch ausreicht, das Daten hintereinander verarbeitet werden, ist die von Neumann Architektur angebracht.

Mikroprozessoren entsprechen der v. Neumann Architektur, weil ein Mikroprozessor z. B. keinen kontrollinternen Speicher hat, in dem er mehrere Daten parallel verarbeiten müsste.

Aufgabe 2)

In Rechnersystemen stellt der Speicheraufbau ein wesentliches, die Leistungsfähigkeit des Rechnersystems charakterisierendes Merkmal dar.

2.1 Erläutern Sie, welche Speichertypen (*DRAM, SRAM, EPROM* oder *Varianten*) an welcher Stelle im Rechner anzutreffen sind und warum gerade sie eingesetzt werden um die gewünschte Funktion zu realisieren!

Statischer Speicher (SRAM)

Stelle: Prozessorinterner Speicher => liegt im Adressraum des Prozessors
Prozessorexterner Speicher => liegt im Standardarbeitsspeicher eines Prozessors (insbesondere bei Signalprozessoren und Controllern)

Begründung für Einsetzung:

- Kurze Zugriffszeiten
- Als Prozessorinterne oder Prozessorexterne Speicher realisiert
- Lese- und Schreibgeschwindigkeit ist gleich.
- Wiederholtes Lesen möglich
- Der Speicherinhalt bleibt solange erhalten, solange Versorgungsspannung bereitsteht

Dynamischer Speicher (DRAM):

Stelle: Standardspeicher des Motherboards jedes Rechners => Kondensator

- Zur Aufrechterhaltung ist ein Refresh notwendig (auf Dauer automatisch Spannung verloren. Immer wieder erneuern damit die Spannung vorhanden bleibt)
- Geringe Speicherbereiche
- Geringe Verlustleistungen
- Längere Zugriffszeiten

EPROM

Stelle: befindet sich auf dem Motherboard, Ladung auf dem Gate bleibt auch bei Wegfall der Versorgungsspannung erhalten

- Prozessorinterner Speicher bei Signalprozessoren oder Controller
- Prozessorexterner Speicher z. B. BIOS des PC.
- EPROM (ist elektrisch schreibbar, löscherbar mit UV-Licht)
- EEPROM (ist elektrisch schreib- und löscherbar)
- FLASH-EPROM (ist elektrisch schreib- und löscherbar in Speicherzellen)

2.2 Welcher Vor- und Nachteile sind zu erwarten, wenn ein Prozessor mit einer Vielzahl von Registern aufwartet?

Vorteile	Nachteile
<ul style="list-style-type: none">- kürzeste Zugriffszeit- direkt Adressierbar- verfügbar als Adress-, Datenregister, Akkumulatoren- verwendet als Registerfile	<ul style="list-style-type: none">- kleine Speicherkapazität- muss oft adressiert werden

Aufgabe 3

Beim Entwurf eines Rechnersystems spielen neben der Wahl des Architekturprinzips auch Gestaltungsgrundsätze eine wesentliche Rolle.

3.1 Welche Auswirkungen sind für den Anwender zu erwarten, wenn der eingesetzte Prozessor eine virtuelle Speicherverwaltung aufweist?

- Physikalische Begrenztheit verbergen, z. B. Betrieb eines virtuellen Speichersystems in einem Rechner mit einer MMU (Memory Management Unit)
- Die aktiven Prozesse dürfen selbst niemals mehr Speicherplatz belegen als physikalisch zur Verfügung steht
- Virtuelle und nicht physikalische Adressen werden verwendet. Diese Umwandlung übernimmt die MMU.
- Diese virtuelle Speicherverwaltung ist heute beinahe in jedem modernen Betriebssystem.

3.2. Wie wirkt sich die dynamische Erweiterbarkeit, die einem Befehlssatz eines Prozessors zugrunde liegen möge, für den Anwender aus?

- Erweiterung des Umfeldes des Maschinenbefehlssatzes z. B. Mikroprogrammierung des Steuerwerks des Mikroprozessors

Aufgabe 4

In einem Pentium lassen sich unterschiedlich Datenformate verwenden, die teilweise unmittelbar an der Definition von „C“ anknüpfen.

4.1 Zählen Sie die Datenformate zur Darstellung von vorzeichenlosen Zahlen auf und erläutern Sie, wie diese Zahlen implementiert sind (Anzahl Bytes, Wertigkeit der einzelnen Bitstellen)

8 Bit	0 bis 255	Unsigned char	0 bis 0FFH	1 Byte
16 Bit	0 bis 65535	Unsigned short	0 bis 0FFFFH	2 Byte
32 Bit	0 bis $2^{32}-1$	Unsigned int Unsigned long	0 bis 0FFFFFFFFH	4 Byte

Verwendung => Implementierung:

- Zeiger in Hochsprache C
- Adressen und Adressrechenoperationen in Assembler
- Programmierung von Schnittstellen in Assembler

4.2. Zählen Sie die Datenformate zur Darstellung von vorzeichenbehafteten Zahlen auf und erläutern Sie wie diese Zahlen implementiert sind (Anzahl Bytes, Wertigkeit der einzelnen Bitstellen)

8 Bit	-128 bis 127	Char	80 FH .. 7FH	1 Byte
16 Bit	-32768 bis 32767	Short	8000FH .. 7FFFH	2 Byte
32 Bit	-2^{31} bis $2^{31}-1$	Int / long	80000000 .. 7FFFFFFFH	4 Byte

4.3 Zählen Sie die Datenformate zur Darstellung von gebrauchten rationalen Zahlen auf und erläutern Sie, wie diese Zahlen implementiert sind (Anzahl Bytes, Wertigkeit der Bitstellen)

Float	$-3,4 \times 10^{38}$ bis $+3,4 \times 10^{38}$
Double	$-1,7 \times 10^{308}$ bis $+1,7 \times 10^{308}$
Long double	$-1,1 \times 10^{4932}$ bis $+1,1 \times 10^{4932}$

Implementierung:

- Zusätzliche Darstellung des Wertebereichs $0 \leq x < 1$
- Gebrochen rationale Zahlen werden immer vorzeichenbehaftet dargestellt
- Darstellung als
 - Floating Point Zahl
 - Fix Point Zahl
- Floating Point Zahl / Definition in „C“ als float oder double
- Zahl ist grundsätzlich normiert entsprechend:

$$\text{Zahl} = \text{Vorzeichen (+/-)} * 1. \text{ Mantisse} * 2^{\text{Exponent}}$$

Problem: die Zahl 0 ist nicht darstellbar

Lösung: Verwendung einer Kombination aus dem Exponenten (negative Zahlen im Exponenten haben einen Wert mehr)

- Vorzeichen => Gesamtvorzeichen der Zahl (1 Bit)
- Mantisse => Grundsätzlich positiv (23 Bit float, 52 Bit double)
- Exponenten => vorzeichenbehaftet (8 Bit float, 11 Bit double)

Aufgabe 5

Wenn sie den Befehlssatz des Pentiums hinsichtlich des Arithmetischen Befehle betrachten so fällt auf, dass für die Addition und Subtraktion nicht zwischen vorzeichenbehafteten und vorzeichenlosen Zahlen unterschieden, bei der Multiplikation und der Division jedoch eine Unterscheidung getroffen wird.

5.1 Ist es dann noch sinnvoll bei der Definition von Zahlen in der Hochsprache C eine Unterscheidung zwischen vorzeichenlosen und vorzeichenbehafteten vorzunehmen, wenn diese Zahlen nur durch Additionen oder Subtraktionen verknüpft werden? (Begründen Sie Ihre Ansicht!)

Ja es ist sinnvoll, denn der Compiler muss wissen ob das MSB (Most Significant Bit) der Vorzeichenregelung entspricht.

5.2 Ist bei der Multiplikation und der Division auch eine Mischung zwischen vorzeichenlosen und vorzeichenbehafteten Zahlen in einem Befehl möglich? (Begründen Sie Ihre Ansicht!)

eine Mischung zwischen Vorzeichenbehafteten und vorzeichenlosen Zahlen ist nicht möglich, da sonst keine Unterscheidung mehr stattfinden kann.

Beispiel:

eine Zahl ≥ 128 multipliziert mit einer anderen

dann ist das MSB immer gesetzt

multipliziert man die Zahl mit einer negativen Zahl stimmt das Ergebnis nicht mehr

gilt natürlich nur bei einem Byte

Aufgabe 6:

Bei der Assemblerprogrammierung unterscheidet man zwischen verschiedenen Adressierungsarten.

6.1 Welche Unterschiede bestehen zwischen der direkten und der indirekten Adressierung?

Direkte Adressierung:

Bei dieser Adressierung wird die Adresse im Programmsegment platziert. Die direkte Adressierung kann auf Quell- und Zieloperanten angewendet werden. Bedingt durch die Platzierung der Adresse im Programmsegment ist diese während des Programmlaufs nicht mehr änderbar!!!

- Adresse des Operanten steht im Befehl
- In Hochsprache C für globale Variablen eingesetzt
- Z. B. MOV AX, [2034H]; Inhalt der Speicherstelle 2034H in AX
- Anzahl der Bytes hängt vom Registernamen ab

Indirekte Adressierung

Bei der indirekten Adressierung wird die Adresse in einem Adressregister platziert und kann dadurch während der Laufzeit des Programms abgeändert werden. Diese Adressierung ist auf Quell- und Zieloperanten anwendbar.

- Adresse des Operanten steht im Adressregister
- In der Hochsprache C für berechnete Variablen einsetzbar
- Z. B. MOV AX [BX], Speicherstelle auf die BX in AX
- Anzahl der Bytes hängt vom Registernamen ab

6.2 Lassen sich in einem Befehl 2 direkte Adressen verwenden? (Begründen Sie Ihre Ansicht!)

Keine Unterstützung des Befehlssatzes, mov verlangt mindestens einen Register oder Segment. Der Adressbus ist beschränkt und kann nicht zwei Speicherstellen gleichzeitig holen, deswegen muss immer ein Wert schon in einem Register liegen...

7.0 Aufgabe:

Zeiger int B	ESI [EBP+12]
Zeiger int A	EDI [EBP+8]
RET	
EBP	

```
...  
MOV EDI, [EBP+8]  
MOV ESI, [EBP+12]  
MOV EAX, [EDI]  
MOV EDX, [ESI]  
MOV [EDI], EDX  
MOV [ESI], EAX  
LEAVE 0,0  
RET
```

8.0 Aufgabe:

Zeile 30 mov [edi+8], edx (+4 muss es lauten)
Zeile 33 push ecx (macht schleife kaputt)
Zeile 36 pop ecx (sinnlos)