RHEINISCH-
WESTFÄLISCHE
TECHNISCHE
HOCHSCHULE
AACHEN

LEHRSTUHL
INFORMATIK XI
SOFTWARE FÜR
EINGEBETTETE
SYSTEME

Professor Dr.-Ing. Stefan Kowalewski
Dipl.-Inform. Andreas Polzer
Dipl.-Inform. Ralf Mitsching

Aachen, 11. Januar 2007
SWS: V2/Ü2, ECTS: 4

Einführung in die Technische Informatik

WS 2006/2007

Blatt 7: von-Neumann-Rechnermodell

Ihre Lösung zu den mit (⋆) gekennzeichneten Übungen sollen Sie am **19.01.2007** in der Übung abgeben. Die Bearbeitung der Aufgaben in Lerngruppen ist sinnvoll. Bitte geben Sie nur eine Lösung pro Lerngruppe ab.

Aufgabe 1: (⋆) von-Neumann-Rechnermodell

1946 wurde das von-Neumann-Rechnermodell vorgestellt, das die Rechnerarchitektur bis heute maßgeblich beeinflusst. Arbeiten Sie die grundlegenden Organisationsprinzipien und Besonderheiten dieses Modells anhand des achten Kapitels (im Lehrbuch von Oberschelp/ Vossen) heraus, indem Sie folgende Fragen möglichst prägnant und in eigenen Worten beantworten.

a) Mit dem von-Neumann-Rechnermodell wurde erstmalig das Konzept für einen echten „general-purpose Computer“ vorgeschlagen. Was ist darunter zu verstehen?

b) „Programme sind auch nur Daten“ ist eine grundlegende und eng mit dem von-Neumann-Rechnermodell verbundene Sichtweise. Was ist darunter zu verstehen?

c) Das von-Neumann-Rechnermodell setzt sich aus drei Hauptbestandteilen zusammen. Welche Bestandteile sind dies und welchem Zweck dienen sie?

d) Im von-Neumann-Rechnermodell ist der Datenprozessor ein Bestandteil der CPU. Welche Aufgaben werden von welchen Komponenten dieses Prozessors erfüllt?

e) Im von-Neumann-Rechnermodell ist der Befehlsprozessor ein Bestandteil der CPU. Welche Aufgaben werden von welchen Komponenten dieses Prozessors erfüllt?

f) Das von-Neumann-Rechnermodell unterscheidet zwischen Daten- und Adressbus. Warum macht das Sinn? Es ergeben sich auch Zusammenhänge zwischen der Größe (in Bits) des MAR, des MBR, des Speichers, einer Speicherzelle sowie der Speicherzellenanzahl. Welche?

g) Die Arbeitsweise eines von-Neumann-Rechners wird durch die Bezeichnung SISD allgemein charakterisiert. Welches Prinzip verbirgt sich hinter dieser Abkürzung?

h) Bahnbrechend neu am von-Neumann-Rechnermodell war das Konzept einer quasi universellen Programmierbarkeit. Erörtern Sie in diesem Zusammenhang die Begriffe Maschinencode, Assemblersprachen sowie Ein- und Mehr-Adress-Befehle.

i) Charakteristisch für das von-Neumann-Rechnermodell ist ein Zwei-Phasen-Konzept der Befehlsverarbeitung. Welches Problem wird damit auf welche Weise gelöst?

j) Die Architektur eines klassischen von-Neumann-Rechners führte schon bald zu einem gewichtigen Problem, dem von-Neumannschen „Flaschenhals“. Was ist darunter zu verstehen und wie versuchte man später dieses Problem zunächst zu umgehen?

Aufgabe 2: Alternative Konfigurationen eines von-Neumann-Rechners
Betrachten Sie die folgenden „Design-Alternativen“ für einen Rechner und diskutieren Sie deren Praktikabilität.

Alternative	MAR-Größe in Bits	Speichergröße	Länge eines Speicherwortes (in Bits)
a	32	2^{32}	28
b	32	2^{32}	12
c	30	2^{32}	32
d	34	2^{32}	32
e	32	32	2^{32}
f	2^{32}	32	32

Aufgabe 3: Adressierung bei Prozessoren

Ein einfaches Rechnersystem enthalte einen 2KB großen ROM-Chip, einen 2KB großen RAM-Chip und einen Parallelport mit vier Leitungen nach außen, die je 8 Bit breit sind. Der Adressbus sei 16 Bit breit.

Entwerfen Sie eine geeignete Adressierung dieser drei Chips. Weisen Sie den Chips dazu geeignete Adressbereiche zu und entwickeln Sie für jeden Chip eine Schaltung, über die sein CS(Chip Select)-Eingang an den Adressbus angeschlossen werden kann.

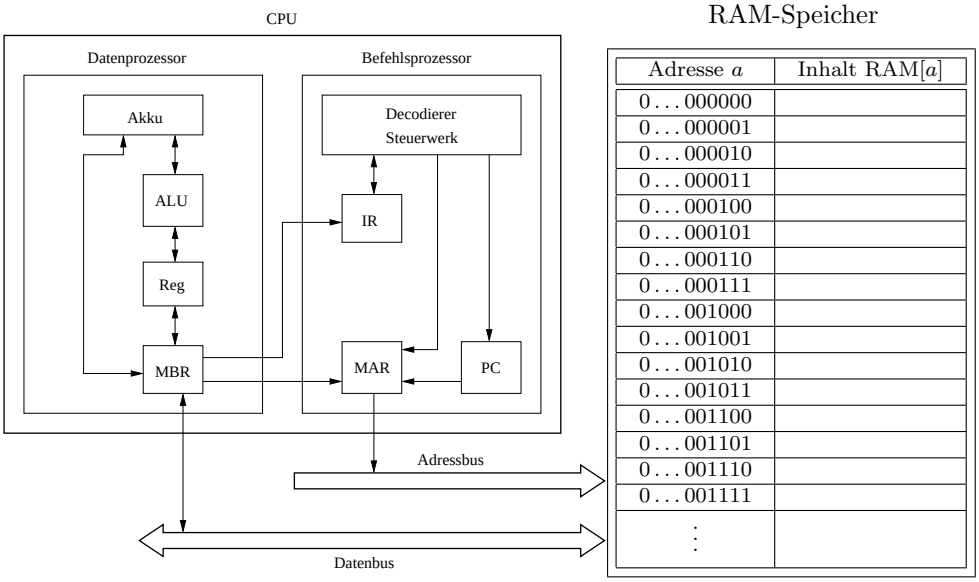
Beachten Sie, dass der Parallelport-Controller einen Adressbereich braucht, der es dem Chip erlaubt, aus der Adresse auf die angesprochene Leitung zu schließen.

Aufgabe 4: Hauptspeicherzugriffe
Nehmen Sie an, dass Cache-Zugriffe sechsmal schneller durchgeführt werden können als Hauptspeicherzugriffe. Welche Beschleunigung wird durch den Einsatz eines Caches erziehlt, wenn dieser in 90 % der Zeit genutzt werden kann?

$$\text{Beschleunigung} = \frac{\text{Ausführungszeit ohne Cache}}{\text{Ausführungszeit mit Cache}}$$

Aufgabe 5: ($\star$) Arbeitsweise einer CPU

Die folgende Abbildung zeigt eine CPU-Struktur und ihre Verbindung zum RAM-Speicher. Im Vergleich zu der in der Vorlesung gezeigten Struktur einer CPU (Folie 7.3) besitzt diese CPU ein zusätzliches Register Reg. Dafür wird hier zur Vereinfachung der genaue Aufbau des Registers Akku vernachlässigt: statt MR, L und A wird einfach nur Akku verwendet.



Für eine Speicheradresse a bezeichne RAM[a] die zugehörige Speicherzelle bzw. deren Inhalt.

Nehmen Sie an, dass diesem Rechner die in der folgenden Tabelle durch ihre Operationscodes identifizierten Befehle/Operationen zur Verfügung stehen:

Operationscode	Wirkung(en) der Operation	
0		$\text{PC} \leftarrow \text{PC} + 1$
1	$\text{RAM}[\text{RAM}[\text{PC}+1]] \leftarrow \text{Reg}$	$\text{PC} \leftarrow \text{PC} + 2$
2	$\text{Reg} \leftarrow \text{RAM}[\text{RAM}[\text{PC}+1]]$	$\text{PC} \leftarrow \text{PC} + 2$
3	$\text{Reg} \leftarrow \text{RAM}[\text{PC}+1]$	$\text{PC} \leftarrow \text{PC} + 2$
4	$\text{Reg} \leftarrow \text{Akku}$	$\text{PC} \leftarrow \text{PC} + 1$
5	$\text{Akku} \leftarrow \text{Reg}$	$\text{PC} \leftarrow \text{PC} + 1$
6	$\text{Akku} \leftarrow \text{Akku} + \text{Reg}$	$\text{PC} \leftarrow \text{PC} + 1$
7	$\text{Akku} \leftarrow \text{Akku} - \text{Reg}$	$\text{PC} \leftarrow \text{PC} + 1$
8	$\text{Akku} \leftarrow \text{Akku} \times \text{Reg}$	$\text{PC} \leftarrow \text{PC} + 1$
9	$\text{Akku} \leftarrow \text{Akku} \div \text{Reg}$	$\text{PC} \leftarrow \text{PC} + 1$
10		$\text{PC} \leftarrow \text{RAM}[\text{PC}+1]$
11	$\begin{cases} \text{Falls Akku} = 0\text{:} \\ \text{Falls Akku} \neq 0\text{:} \end{cases}$	$\begin{array}{l} \text{PC} \leftarrow \text{RAM}[\text{PC}+1] \\ \text{PC} \leftarrow \text{PC} + 2 \end{array}$

Der Rechner kenne nur ganze Zahlen (keine Gleikomma-Zahlen). Dementsprechend steht $\div$ für die ganzzahlige Division (z. B. $26 \div 7 = 3$).

a) Beschreiben Sie, wozu die Operationen mit den Operationscodes 1, 2, 3, 10, 11 dienen.

Im folgenden „Speicher-Auszug“ sind neben die Dualzahlen (mit denen „echte Rechner“ natürlich arbeiten) auch die entsprechenden Dezimalzahlen (klein und in Klammern) geschrieben.

In Ihren Lösungen brauchen Sie aber nur Dezimalzahlen anzugeben.

Es sei angenommen, dass die CPU hier bei jeder Befehlsverarbeitung die folgenden „Zuweisungen“ (in dieser Reihenfolge) durchführt:

MAR ← PC
MBR ← RAM[MAR]
IR ← MBR
Wirkung(en) der Operation IR
(PC-Änderung zuletzt!)

b) Protokollieren Sie den Inhalt der Register Akku, Reg, IR und PC nach jeder Befehlsverarbeitung bis PC den Wert 25 enthält, wobei zu Anfang alle Register 0 enthalten und der Speicher wie nebenstehend gefüllt ist.

Hinweis: PC sollte den Wert 25 nach 14 Befehlsverarbeitungen enthalten.

c) Betrachten Sie nun den nebenstehenden Speicherinhalt von Adresse 0 bis Adresse 21 (einschließlich) und interpretieren Sie ihn als ein Programm. Was wird durch dieses Programm berechnet?

Adresse a	Inhalt RAM[a]
(0) 0...000000	0...00000010 (2)
(1) 0...000001	0...00010110 (22)
(2) 0...000010	0...00000101 (5)
(3) 0...000011	0...00000010 (2)
(4) 0...000100	0...00010111 (23)
(5) 0...000101	0...00000111 (7)
(6) 0...000110	0...00000100 (4)
(7) 0...000111	0...00000001 (1)
(8) 0...001000	0...00011000 (24)
(9) 0...001001	0...00001011 (11)
(10) 0...001010	0...00011001 (25)
(11) 0...001011	0...00000011 (3)
(12) 0...001100	0...01100100 (100)
(13) 0...001101	0...00000101 (5)
(14) 0...001110	0...00000010 (2)
(15) 0...001111	0...00011000 (24)
(16) 0...010000	0...00001001 (9)
(17) 0...010001	0...00000100 (4)
(18) 0...010010	0...00000001 (1)
(19) 0...010011	0...00011000 (24)
(20) 0...010100	0...00001010 (10)
(21) 0...010101	0...00011001 (25)
(22) 0...010110	0...00001011 (11)
(23) 0...010111	0...00000100 (4)
(24) 0...011000	0...00000000 (0)
⋮	⋮

Sie kennen vermutlich schon die folgende Anekdote aus dem Leben von C. F. Gauß (bzw. eine Variante davon):

> Um seine Schüler (bzw. Gauß) eine Zeitlang zu beschäftigen (bzw. zu bestrafen), gab der Lehrer ihnen (bzw. ihm) die Aufgabe, die Zahlen von 1 bis 100 zu addieren. Doch der kleine Gauß hatte die Lösung 5050 schon nach kurzer Zeit gefunden.
>
> Gauß hatte nämlich erkannt, dass man statt $1 + 2 + 3 + \cdots + 98 + 99 + 100$ zu rechnen auch 50 „Paare“ $1 + 100$, $2 + 99$, $3 + 98$, ..., $50 + 51$ bilden kann, die jeweils 101 ergeben, und sich so die Lösung $50 \cdot 101 = 5050$ leicht berechnen lässt.

D. h.: Gauß benutzte die (nach ihm benannte) Summenformel $\sum_{k=1}^{n} k = \frac{1}{2}n(n+1)$.

Schreiben Sie je ein Programm, das die Summe s der ersten n natürlichen Zahlen berechnet ...

d) ... durch *fortgesetzte Addition* (also nach der Formel $s = \sum_{k=1}^{n} k$);

e) ... in sog. *geschlossener Form* (z. B. nach der Formel $s = \frac{1}{2}n(n+1)$).

Das Ergebnis s soll bei Adresse 3 gespeichert werden, wobei n bei Adresse 2 gespeichert sei und das Programm selbst bei Adresse 4 beginnt. Sie können davon ausgehen, dass $n \geq 0$ gilt. Kommentieren Sie Ihr Programm jeweils unbedingt ausführlich!

Adresse a	Inhalt RAM[a]
(0) 0...000000	0...00001010 (10)
(1) 0...000001	0...00000100 (4)
(2) 0...000010	*...******** (*n*)
(3) 0...000011	*...******** (*s*)
⋮	⋮

Hinweis zu Aufgabenteil d):
Beachten Sie, dass die Summationsreihenfolge durch $\sum_{k=1}^{n} k$ nicht festgelegt ist.[1]

Hinweis zu Aufgabenteil e):
Beachten Sie, dass sich $\frac{1}{2}n(n+1)$ auf verschiedene Arten berechnen lässt.[1]

[1] Nutzen Sie die Assoziativität, Kommutativität und ggf. Distributivität der Rechenoperationen aus, um möglichst einfache und schnelle (aber natürlich dennoch korrekte) Programme zu schreiben.