

Prüfungsfach „Architekturen von Rechnersystemen“

Studiengang:

Semester:

Prüfungstermin: 4.02.2002, 14:00 Uhr, E-014

Dauer der Prüfung: 60 Minuten

Anzahl der Aufgabenblätter: 9 (inkl. Deckblatt)

Hilfsmittel: persönliches Skript

Maximal erreichbare Punktzahl: 50

Note: _____

Punktzahl: _____

Erstkorrektor: _____

Zweitkorrektor: _____

Prüfer/Aufgabensteller der Prüfung: Theo Kupfer

Name: _____

Vorname: _____

Matrikel-Nr.: _____

Unterschrift: _____

Wichtige Hinweise:

1. Tragen Sie bitte, bevor Sie mit der Lösung der Aufgaben beginnen, die erforderlichen Angabe zu Ihrer Person im Deckblatt der Aufgabenblätter ein!
2. Die Lösungen sind auf den Aufgabenblättern einzutragen!(ggf. Rückseiten der Blätter verwenden)
3. Von mehreren Lösungen zur gleichen Aufgabe, ist die gültige Lösung eindeutig zu kennzeichnen!
4. Bei Formeln bitte Variablen und Parameter eindeutig erklären.

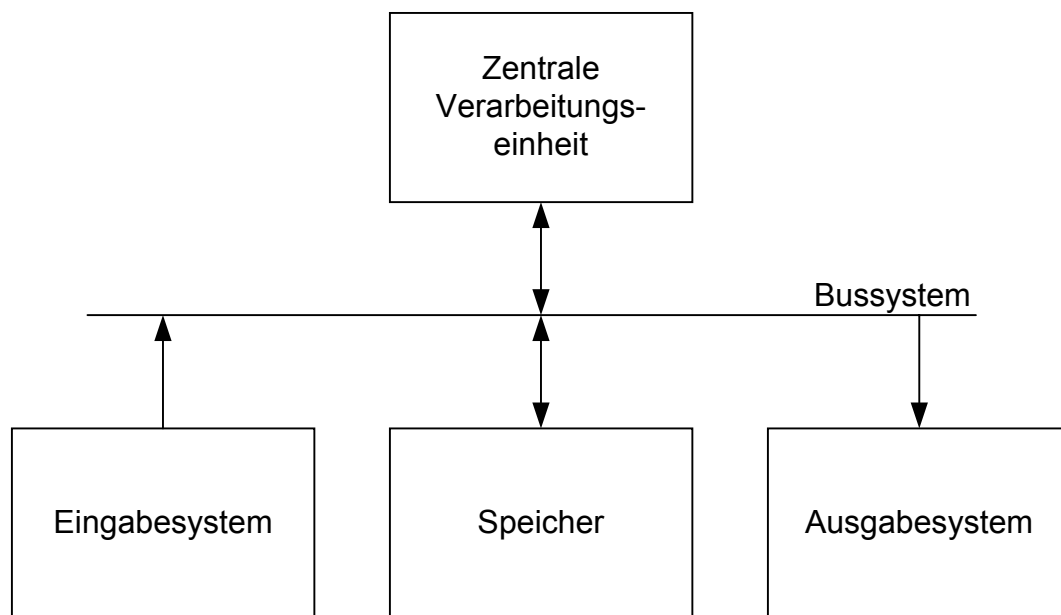
Aufgabe 1:

(Maximale Punktzahl: 6)

Erreichte Punktzahl: _____

Zeichnen Sie die Grundstruktur eines Rechnersystems nach “von Neumann” und erklären Sie kurz die Komponenten:

Ihre Lösung:



Zentrale Verarbeitungseinheit: Führt die Programmbefehle aus.

Speicher: Enthält Programm und Daten

Eingabesystem: Daten können damit von aussen eingelesen werden.

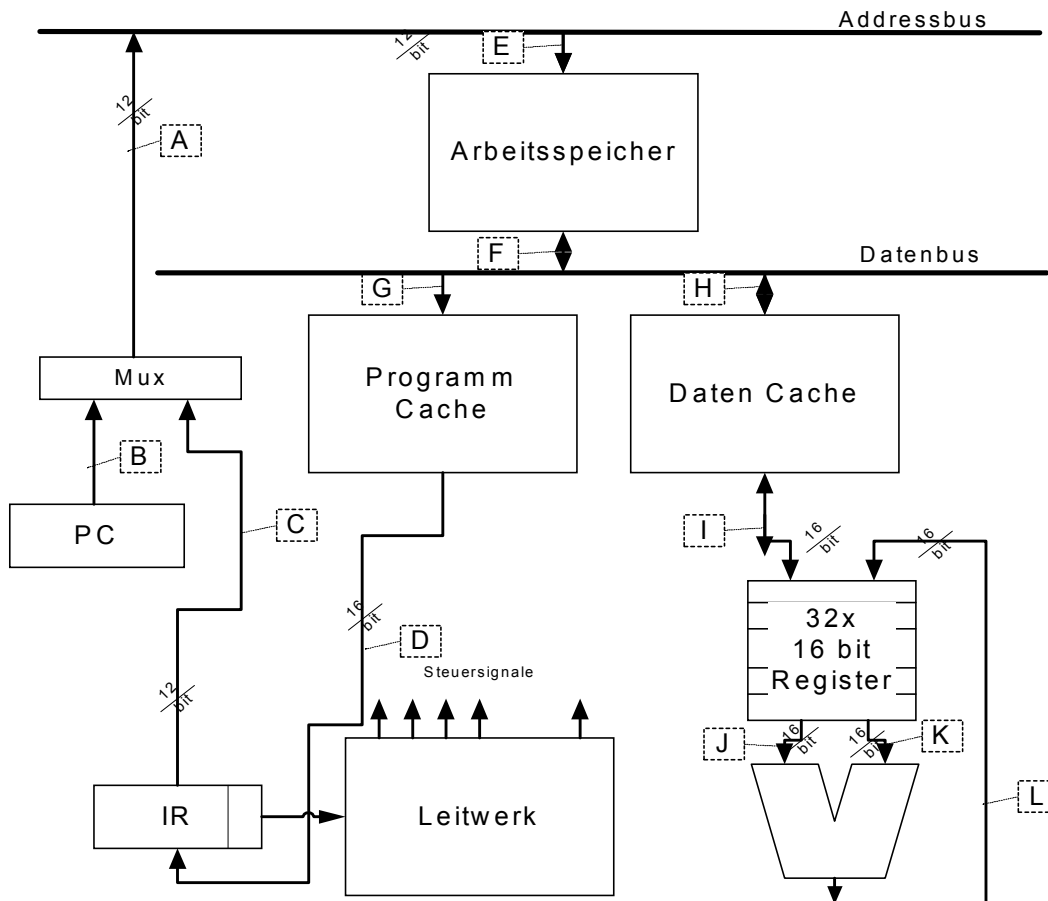
Ausgabesystem: Daten können damit nach aussen ausgegeben werden.

Aufgabe 2:

(Maximale Punktzahl: 12)

Erreichte Punktzahl: _____

Bild 1 zeigt das Blockschaltbild eines Rechnersystems. Der Rechner besteht aus Arbeitsspeicher, Datencache, Programmcache, Leitwerk, einem Befehls Register (IR), einem Programmzähler Register (PC) und mehreren Universalregistern. Beantworten Sie bitte untenstehende Fragen. Kreuzen die dabei das Ergebnis an und begründen sie die Wahl mit einem oder zwei kurzen Sätzen. Bei Frage d bitte die Kennbuchstaben der beteiligten Verbindungen auflisten.

**Bild 1**

- a) Beschriften sie die fünf nicht ausgefüllten Kästen in Bild 1.
- b) Welche der Klassifizierung aus der folgenden Auswahl trifft für die obige Architektur aus Sicht des Programmierers zu? (Nur ein Ergebnis richtig)

Lösung: Von Neumann Architektur
 Harvard Architektur

Begründung:

Programmspeicher und Datenspeicher sind aus Sicht des Programmierers nicht getrennt. Beide werden über den gleichen Adressbus adressiert. Der Cache Speicher ist für den Programmierer transparent.

- c) Welche Befehle/Adressierungsarten sind bei dieser Architektur möglich. Begründen sie Ihr Ergebnis:

Lösung:

Befehl/Adressierungsart	möglich	nicht möglich
c.1: Sprungbefehl	<input type="checkbox"/>	<input checked="" type="checkbox"/>
c.2: Implizite Adressierung	<input checked="" type="checkbox"/>	<input type="checkbox"/>
c.3: Registerindirekte Adressierung	<input type="checkbox"/>	<input checked="" type="checkbox"/>
c.4: Speicherdirekte Adressierung	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Begründung:

c.1: Keine Verbindung vom IR zum PC. Das heisst PC kann nicht mit neuen Werten geladen werden

c.2: Register werden über Leitwerk nach der Befehlsdekodierung adressiert.

c.3: Keine Verbindung von den Registern zu dem Adressbus.

c.4: Speicher wird über die oberen 12 bits des IR direkt adressiert.

- d) Welche Verbindungsleitungen (Im Bild 1 mit A bis L bezeichnet) werden beim Laden eines Befehls benutzt. Dabei soll die Vorbedingung "Cache ist leer" gelten.

Lösung:

A, B, E, D, F, G

Speicher wird über B, A, E adressiert. Der Befehl wird über F, G, D in das IR geladen.

Aufgabe 3:

(Maximale Punktzahl: 5)

Erreichte Punktzahl: _____

Erklären Sie das Prinzip des virtuellen Speichers. Aus welchem Grund wird virtueller Speicher eingesetzt?

Lösung:

Beim virtuellen Speicher werden logische Adresse (Adresse aus Sicht des Programms) und physikalische Adresse (Adresse zur Adressierung des Hauptspeichers) entkoppelt. Die Abbildung der logischen Adressen auf die physikalischen Adressen geschieht mit Hilfe der Memory Management Unit.

Virtueller Speicher wird immer dann angewandt wenn der adressierbare Speicher größer ist als der physikalische Speicher.

Durch Einsatz von Virtuellem Speicher kann z.B.:

- Programme ohne Wissen um die Größe und Position des physikalischen Speichers geschrieben werden.
- Speicherbereiche von unterschiedlichen Prozessen vor einander geschützt werden.

Aufgabe 4:

(Maximale Punktzahl: 5)

Erreichte Punktzahl: _____

Geben Sie fünf verschiedene Massenspeichertechnologien an. Beschreiben Sie jeweils den Einsatzbereich.

Lösung:

Massenspeichertechnologie	Einsatzbereich
Festplatte	Nichtflüchtiger Speicher im Rechner für Programme und Daten. Wird auch als externe Speichererweiterung für virtuellen Speicher verwendet.
CD-ROM	Archivierung, Datenaustausch, Auslieferung von Programmen
Bandlaufwerk	Backup, Archivierung
Magneto Optische Disk	Backup, Archivierung, Datenaustausch
Floppy Disk	Datenaustausch, Auslieferung von Programmen

Aufgabe 5: (Maximale Punktzahl: 4) Erreichte Punktzahl: _____

Welchen Vorteil bringt der Einsatz von Gleitkommazahlen bzw. Festkommazahlen?

Ihre Lösung:

Vorteil von Gleitkommazahlen:

Der Dynamikbereich ist wesentlich größer. Das heisst der Abstand zwischen der Zahl mit dem größten Betrag und der Zahl mit dem kleinsten Betrag.

Vorteil von Festkommazahlen:

Die Berechnungen mit Festkommazahlen sind weniger aufwendig und damit schneller.

Aufgabe 6: (Maximale Punktzahl: 4) Erreichte Punktzahl: _____

Kreuzen Sie den wichtigeren der folgenden zwei Parameter eines Rechnersystems einer Telefonvermittlung an. Wie lässt er sich verbessern?

Ihre Lösung:

Zuverlässigkeit

Verfügbarkeit

Begründung:

Es ist wichtiger, dass Das Telefonsystem dann zur Verfügung steht wenn es gebraucht wird. Es ist nicht so wichtig wenn es in den Zwischenzeiten kurz ausfällt.

Verbesserungsmöglichkeit:

Die Verfügbarkeit lässt sich verbessern durch:

Verbesserung der Zuverlässigkeit

Verkürzung der Reparaturzeiten

Automatische Fehlerbehebung durch Redundanzen

Aufgabe 7:

(Maximale Punktzahl: 4)

Erreichte Punktzahl: _____

Ein Applikationsprogramm schaut folgendermassen aus:

```
main()
{
    int a;
    while(1)
    {
        a=readinput();
        if ((a-10)>0)
        {
            writeoutput("ON");
        }
        else
        {
            writeoutput("OFF");
        }
        wait(10);
    }
}
```

Lässt sich dieses Programm durch Parallelisierung optimieren (z.B. Einsatz eines Multiprozessor-systems) oder nicht? Begründen Sie Ihre Antwort.

Lösung:

Die einzelnen Anweisungen des Programms hängen direkt voneinander ab. Sie sind nicht nebenläufig. Deshalb bringt eine Parallelisierung keine Vorteile.

Aufgabe 8:

(Maximale Punktzahl: 4)

Erreichte Punktzahl: _____

Klassifizieren Sie die unten aufgelisteten vier Schnittstellentypen:

Ihre Lösung:

Computerschnittstelle	Punkt zu Punkt	Sternnetz	Parallel Party line	Seriell Party line	Parallel Daisy Chain	Seriell Daisy Chain
RS-232	X					
I2C				X		
IDE			X			
IEC-Bus			X			

Aufgabe 9:

(Maximale Punktzahl: 6)

Erreichte Punktzahl: _____

Ein Prozessor mit fünf Phasen Pipeline Architektur hat folgende fünf Befehls-Phasen:

- Fetch: Befehl holen
- Decode: Befehl dekodieren
- Load: Operanden laden
- Execute: Befehl ausführen
- Write Back: Operanden abspeichern

Der Prozessor hat weiterhin eine Akkumulator Architektur d.h. er besitzt ausser dem Akkumulator kein weiteres Datenregister und gemeinsamen Programm- und Datenspeicher.

Zwischen welchen Phasen können strukturelle Hazards auftreten?

Wie lässt sich die Architektur optimieren?

Lösung:

Strukturelle Hazards treten zwischen den Fetch, Load und Write Back Phasen auf. Bei allen diesen Phasen wird der externe Arbeitsspeicher benutzt (Da keine Register befinden sich die Operanden im externen Arbeitsspeicher und Daten- und Programmspeicher sind nicht getrennt).

Durch Trennung von Daten und Programmspeicher können die Hazards mit der Fetch Phase vermieden werden.

Durch mehr interne Register können die Hazards zwischen der Load und Write Back Phase vermieden werden da die Daten nun intern bewegt werden.