Universität Bremen – Fachbereich 3
Mathematik & Informatik

Prof. Dr. Rolf Drechsler, drechsler@informatik.uni-bremen.de, MZH 3510
Dipl.-Inform. Daniel Große, grosse@informatik.uni-bremen.de, MZH 3490

2. Übungsblatt zur Vorlesung

Technische Informatik 1

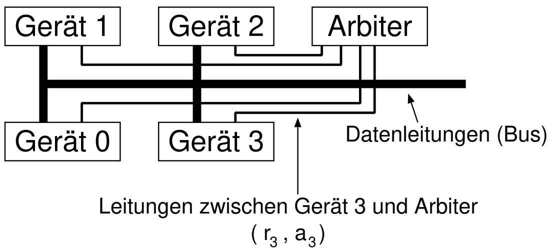
Aufgabe 1 (2 Punkte)
Sei S ein System, welches eine Aufgabe A bestehend aus n Teilaufgaben $A_1, \ldots, A_n$ ausführt, wobei S für alle Teilaufgaben die gleiche Zeit benötigt. T sei eine verbesserte Version von S, die 30% der Teilaufgaben von A 10-mal schneller ausführen kann als S. Die Ausführungszeit der restlichen Aufgaben sei bei S und T gleich.

Wie groß ist die Beschleunigung (speedup), d.h. der Quotient aus der alten und der neuen Ausführungszeit, die man bei der Ausführung von A durch den Einsatz von T gegenüber S erhält. Begründe Deine Überlegungen.

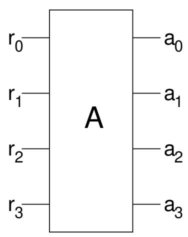
Aufgabe 2 (6 Punkte)
Ein Bus-Arbiter ist eine Komponente, die den Zugriff mehrer Geräte auf einen Bus regelt. Abbildung 1(a) zeigt die schematische Darstellungen eines Busses auf den vier Geräte zugreifen können. Da zu jeder Zeit nur ein Gerät die Datenleitungen benutzen darf, wird der Arbiter benötigt. Bevor ein Gerät den Bus benutzt signalisiert es die Anforderung (*Request*). Der Arbiter entscheidet dann, welchem Gerät der Bus zugeteilt wird und reagiert mit einer entsprechenden Bestätigung (*Acknowledge*).

Hier ist ein Bus-Arbiter für vier Geräte zu realisieren. Jedes Gerät hat eine Request-Leitung (r_0, r_1, r_2, r_3). Der Arbiter signalisiert einem Gerät über die Acknowledge-Leitung (a_0, a_1, a_2, a_3), ob es den Bus zugeteilt bekommt. Dabei dürfen keine zwei Geräte gleichzeitig ein Acknowledge erhalten.

Gib eine Boolesche Funktion $A : \mathbb{B}^4 \rightarrow \mathbb{B}^4$ durch eine Wahrheitstabelle und durch Boolesche Ausdrücke an, die diesen Arbiter realisiert. Dabei fordert Gerät i den Bus durch $r_i = 1$ an. Der Bus wird an Gerät i durch $a_i = 1$ zugeteilt. Priorität bei der Busvergabe hat das Gerät mit der kleinsten Zahl, d.h. wenn z.B. Gerät 1 und Gerät 2 den Bus zugleich anfordern, wird der Bus an Gerät 1 zugewiesen.



(a) Bus mit 4 Geräten und Arbiter



(b) Blockschaltbild des Arbiters

Abbildung 1: Bus-Arbiter

Aufgabe 3 (2 Punkte)

Zeige, dass es möglich ist, eine Zuweisung mittels

a) indirekter Adressierung (Rdest := Mem[Mem[Rsrc]])

b) indizierter Adressierung (Rdest := Mem[Rsrc+4])

nur unter Verwendung von absoluter Adressierung (Rdest := Mem[Rsrc]) zu realisieren.

Aufgabe 4 (6 Punkte)

Die *Fibonacci Folge* ist rekusiv definiert durch:

$$fib(n) = \begin{cases} 0 & , \quad n = 0 \\ 1 & , \quad n = 1 \\ fib(n-2) + fib(n-1) & , \quad n > 1 \end{cases}$$

Gib ein **iteratives** Assembler-Programm an, welches die n-te Fibonacci Zahl berechnet. Der Parameter n steht dabei zu Beginn in dem Register `$t0` zur Verfügung. Kommentiere die einzelnen Assemblerbefehle.

Aufgabe 5 (4 Punkte)

Für ein spezielles Einsatzgebiet soll ein neuer Rechner entworfen werden. Die Rechengenauigkeit aller Operationen soll 16 Bit betragen. Aufgrund der vorkommenden Operationen werden 95 Instruktionen benötigt. Der Rechner hat 28 Register bei einem Adressraum von maximal 68K. Es wird eine Load/Store-Architektur verwendet: Befehle, die auf den Speicher zugreifen, haben zwei Register als Operanden (Befehl 1: Rdest := Mem[Rsrc], Befehl 2: Mem[Rsrc] := Rdest). Kein Befehl hat mehr als drei Operanden, mindestens zwei Operanden sind Register, der dritte kann eine 8-Bit-Konstante oder ein Register sein.

a) Wie breit müssen die Register mindestens sein?

b) Wieviele Bit werden benötigt, um eine Instruktion zu codieren?

Begründe Deine Überlegungen.

Abgabetermin: zu Beginn der Vorlesung am 02.05.2005