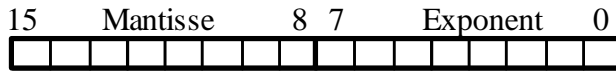


Klausur "Informatik I" vom 19.2.1999 Teil "Rechnerstrukturen"

Aufgabe 1: Binäre Informationsdarstellung (18 Punkte)

Folgende Gleitkommadarstellung werde im folgenden zugrundegelegt (Mantisse und Exponent seien beide 8-bit breite 2er-Komplementzahlen):



1.1 Geben Sie die folgenden Werte als Dezimalzahlen-Ausdrücke (z.B. $-8 \cdot 2^{-64}$) an! (6P)

minreal =

maxreal =

smallreal =

1.2 Wandeln Sie die Dezimalzahlen $a = 1,125$ und $b = -40,5$ in die oben beschriebene binäre Gleitkommazahlen-Darstellung und geben Sie die Darstellung der Summe an! (6P)

a =

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

b =

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

a + b =

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

1.3 Register D0 enthalte (als '.W'-Ganzzahl im 2er-Komplement) eine Zahl z mit $-100 < z < +100$. Geben Sie die AS68000-Assembler-Befehlsfolge an, welche die Gleitkommadarstellung von z in Register D2 ablegt! (6P)

- Hilfen: 1.) Die Darstellung in D2 muß nicht normalisiert sein!
 2.) Zur Erinnerung einige AS68000-Befehle:
- | | | | |
|------|--------|--------|---|
| MOVE | .B/W/L | Op, Dn | $Dn := Op$ |
| MOVE | .B/W/L | Dn, Op | $Op := Dn$ |
| LSL | .B/W/L | Dn, Dm | $Dm := \text{„Dm um Dn-Stellen nach links geschoben“}$ |
| LSR | .B/W/L | Dn, Dm | $Dm := \text{„Dm um Dn-Stellen nach rechts geschoben“}$ |
| AND | .B/W/L | Op, Dn | $Dn := Dn \text{ AND } Op$ |
| OR | .B/W/L | Op, Dn | $Dn := Dn \text{ OR } Op$ |

.....

.....

.....

.....

Name, Vorname, Matrikelnummer

BITTE UNBEDINGT LESERLICH AUSFÜLLEN !

Klausur "Informatik I" vom 19.2.1999, Teil "Rechnerstrukturen"

Seite 2 von 6 Seiten

.....

.....

Aufgabe 2: Schaltnetzentwurf

(16 Punkte)

Es soll ein Baustein V „Voter“ entwickelt werden.

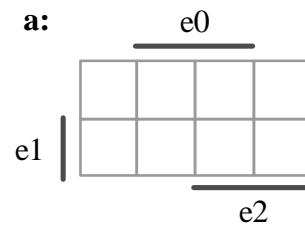
Er habe drei Eingänge e0, e1 und e2 sowie einen Ausgang a.

Ausgang a soll durch den Baustein jeweils mit demjenigen Wert belegt werden, den die Mehrheit der Eingänge gerade besitzt.

2.1 Tragen Sie unten links die Wertetabelle für das Schaltnetz ein!

(4P)

i	e2	e1	e0	a



2.2 Entwickeln Sie eine minimale disjunktive Form für die Funktion a per KV-Diagramm! Verwenden Sie dazu das Diagramm oben rechts! Geben Sie unten das Ergebnis an!

(6P)

a =

2.3 Wandeln Sie die Form aus 2.2 in eine möglichst einfache Form um, die ausschließlich NAND-Operationen verwendet!

(2P)

.....

2.4 Geben Sie zu 2.3 einen Schaltplan an!

(4P)

Aufgabe 3: Schaltwerkentwurf - Mealy-Automatenentwurf (16 Punkte)

Es soll ein synchroner Baustein HWZ „Haben wir zugenommen?“ entworfen werden.

Er habe (neben Takt- und Rücksetzeingang) 2 binäre Eingänge e_1 und e_0 , über die Betragszahlen aus $[0..2]$ eingegeben werden.

Es gebe einen Ausgang a . Er soll mit 1 belegt sein, wenn die aktuelle Eingabe eine größere Zahl als die unmittelbar vorangegangene Eingabe darstellt. Sonst soll er mit 0 belegt sein.

- 3.1** Entwerfen Sie einen entsprechenden Mealy-Automaten mit möglichst wenig Zuständen! Geben Sie unten die Alphabete sowie das Zustandsübergangsdiagramm an! (8P)

$E = \dots\dots\dots$ $A = \dots\dots\dots$

Übergangsdiagramm:

- 3.2** Geben Sie im Übergangsdiagramm oben eine duale Zustandskodierung mit möglichst wenig Bitstellen an! (8P)

Geben Sie in der Tabelle unten die Zustandsübergangsfunktion und die Ausgabefunktion des Automaten unter Verwendung dieser Zustandskodierung per Wertetabelle an!

Aufgabe 4: Schaltwerkentwurf - Schaltungsentwurf

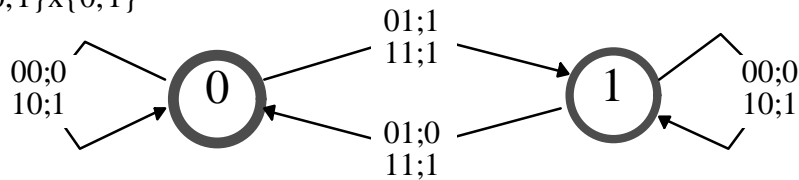
(16 Punkte)

Im folgenden soll die Schaltung eines synchronen Schaltwerk-Bausteins entworfen werden. Gegeben ist der Mealy-Automat zu diesem Baustein.

S: $s \in \{0, 1\}$

E: $(e1, e0) \in \{0,1\} \times \{0,1\}$

A: $a \in \{0,1\}$



4.1 Erstellen Sie unten die Wertetabelle der Zustandsübergangsfunktion und der Ausgabefunktion des Automaten! (8P)

<i>i</i>										

4.2 Geben Sie die Ausgabefunktion a in minimaler disjunktiver Form an! (2P)

a =

4.3 Zur Realisierung soll ein JK-Flipflop verwendet werden! Ergänzen Sie die Wertetabelle von Aufgabe 4.1 um die Spalten der Ansteuerfunktionen! (3P)

Hilfe: Ansteuertabelle des JK-Flipflops

Q	Q'	J	K
0	0	0	-
0	1	1	-
1	0	-	1
1	1	-	0

4.4 Geben Sie die Ansteuerfunktionen in minimaler disjunktiver Form an! (3P)

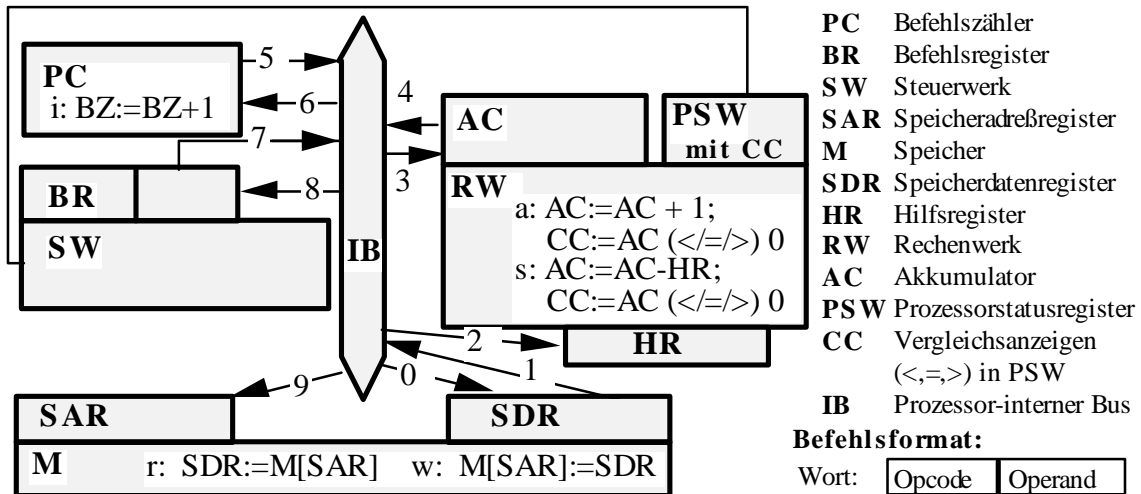
J =

K =

Aufgabe 5: Von-Neumann-Architektur / Registertransferebene

(16 Punkte)

Gegeben sei ein einfacher Prozessor (1-Adreßmaschine) mit folgendem Aufbau und Arbeitsspeicheranschluß. Die Wortbreite der Maschine sei 16 Bit.



Einzelne Buchstaben und Ziffern stehen für einzelne Registertransfer-Schritte:

- i: PC-Inhalt wird um 1 erhöht;
- r bzw. w: Speicher liest bzw. schreibt;
- a bzw. s: RW inkrementiert um 1 bzw. subtrahiert, vergleicht Ergebnis mit 0 und belegt CC in PSW entsprechend;
- 0..9: Transfer entsprechend Pfeil, dabei transferiert 8 in Wortbreite und 7 unter Ausblenden des Operationscodes.

Sie sollen unten Abläufe auf Registertransferebene durch entsprechende Folgen von Ziffern und Buchstaben angeben!

Aufgabe

PC enthalte 1000, M ab Adresse 1000 folgendes Programm, der Prozessor habe mit dem Befehlsholzyklus gerade noch nicht begonnen. Die Zahlen sind als Dezimalzahlen angegeben.

- 1000: Lade Speicher indirekt unter Postinkrement 2000
 - 1001: Subtrahiere Speicher indirekt 2000
 - 1002: Springe, falls kleiner, zu Adresse 1000
 - 1003: Lade Akku indirekt
 - 1004: Halte Rechner an
- 2000: 4000 4000: 4001 4001: 1 4002: 2

a) Der Akku enthalte initial den Wert 0. Geben Sie die Folge von Werten an, die der Belegung des Akkus während der Ausführung des Programmstücks entspricht? (4P)

AC: 0,
 vor 1000 nach 1000 nach 1001 nach 1002 nach 1003

b) Geben Sie nun *vollständig* einen Registertransferablauf an, der der Ausführung dieses Programmstücks (ohne Befehl in 1004) entspricht! (12P)

Befehlsadresse Registertransferschritte

1000 :

1001 :

Name, Vorname, Matrikelnummer

BITTE UNBEDINGT LESERLICH AUSFÜLLEN!

Klausur "Informatik I" vom 19.2.1999, Teil "Rechnerstrukturen"

Seite 8 von 6 Seiten

1002 :

1003 :

Name, Vorname, Matrikelnummer

BITTE UNBEDINGT LESERLICH AUSFÜLLEN !

Klausur "Informatik I" vom 19.2.1999, Teil "Rechnerstrukturen"

Seite 10 von 6 Seiten

.....