

Scheinklausur Rechnerstrukturen (SS 1999)

Name: _____

Vorname: _____

Matrikelnummer: _____

Benötigen Sie eine Schein ? (Wenn ja, bitte ankreuzen.) ☐

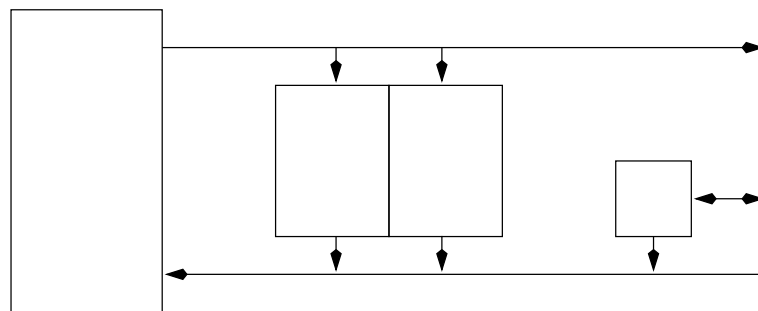
Bitte beachten:

- Tragen Sie die Lösungen auf den Aufgabenblättern ein. Sollte der Platz auf der Vorderseite nicht ausreichen, benutzen Sie die Rückseite.
- Die Klausurergebnisse hängen ab Donnerstag, 1. Juli aus und können auch über das WWW abgefragt werden.
- Die Klausuren können am LuFG abgeholt werden. Es gibt aber *keine* Fragestunde zur Klausur. Lediglich eine Musterlösung werden wir zur Verfügung stellen.
- Um einen Schein zu bekommen, benötigen Sie **55 von 90 Punkten**.

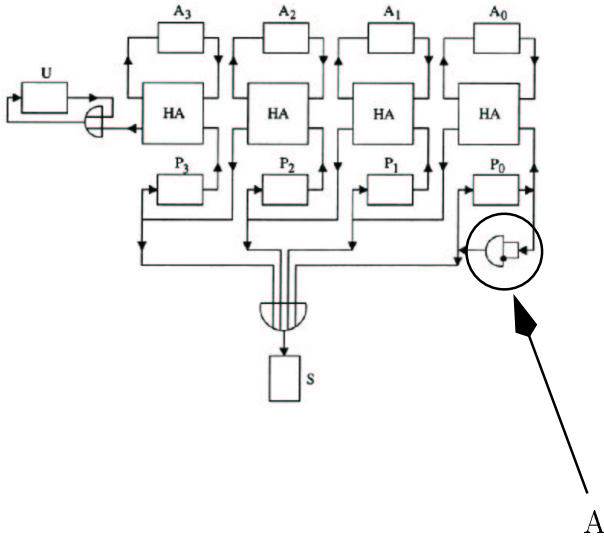
Nr.	Punkte	erreichte Punktzahl	Korrigiert von
1	22		
2	26		
3	16		
4	12		
5	14		
Gesamt	90		

Aufgabe 1: Zum Aufwärmen (22 Punkte)

- (a) Konstruieren Sie aus logischen Gattern Ihrer Wahl einen möglichst einfachen Halbaddierer. (2 P.)
- (b) Konstruieren Sie nun einen Volladdierer. Verwenden Sie dazu die Halbaddierer aus Teil (a). (2 P.)
- (c) Welches wichtige Konzept wird durch folgendes Blockbild (unvollständig) dargestellt? Beschriften Sie die Kästchen und die wichtigsten Pfeile. (4 P.)



(d) Beantworten Sie die folgenden Fragen zu dem abgebildeten von-Neuman-Addierwerk:



(i) Wozu dient das mit A gekennzeichnete Gatter? (4 P.)

(ii) Was zeigt das Delay S an? (4 P.)

(iii) Was ist der Vorteil eines von-Neumann-Addierers im Vergleich zu einem herkömmlichen Parallel-Addierwerk? **(6 P.)**

Aufgabe 2: Boolesche Algebra (26 Punkte)

Gegeben sei folgende vierstellige Boolesche Funktion F :

x_0	x_1	x_2	x_3	F
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0

x_0	x_1	x_2	x_3	F
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

- (a) Wie sind die Maxterme sowie die Maxtermdarstellung einer Booleschen Funktion f definiert?
(6 P.)

- (b) Geben Sie die einschlägigen Minterme für obige Funktion F an! Ist im Falle von F eher die DNF oder die KNF vorzuziehen (Begründung)?
(4 P.)

- (c) Minimieren Sie F mit Hilfe des folgenden Karnaugh-Diagramms. Geben Sie die minimierte Funktion dann explizit an. **(8 P.)**

		$\leftarrow x_0x_1 \rightarrow$			
		00	01	11	10
x_2x_3	\uparrow 00				
	01				
	\downarrow 11				
	10				

- (d) Beweisen Sie, daß die Boole'sche Funktion f funktional vollständig ist, wobei f mit Hilfe der oben definierten Funktion F wie folgt definiert ist:

$$f(y_0, y_1, y_2) := \begin{cases} F(y_0, 1, y_2, y_2) & \text{falls } y_0 = 0 \\ F(y_0, 1, y_1, y_2) & \text{falls } y_0 = 1 \end{cases}$$

(8 P.)

Aufgabe 3: Zahlensysteme und Zahlendarstellung (16 Punkte)

- (a) Können Sie die folgenden Zahlen ins Dezimalsystem konvertieren? Wenn ja, dann tun Sie es. Wenn nein, warum nicht?

(i) $(38)_9$ (2 P.)

(ii) $(10110)_{21}$ (2 P.)

(iii) $(25537)_7$ (2 P.)

(iv) $(4631)_7$ (2 P.)

- (b) Stellen Sie die Zahlen $(129)_{10}$ und $(-13)_{10}$ im Siebener-Komplement zur Basis 8 mit 3 Stellen dar. (4 P.)

- (c) Stellen Sie die Zahl -13.125 als normalisierte Fließkommazahl zur Basis 2 im Dualsystem dar. Die Mantisse habe dabei eine Länge von 23 Bit, der Exponent von 8 Bit. (4 P.)

Aufgabe 4: Ringzähler (12 Punkte)

Konstruieren Sie aus Ihnen genehmen Bauteilen einen Ringzähler, der erst die Primzahlen zwischen 0 und 15 aufzählt und dann mit den zusammengesetzten Zahlen incl. 0 und 1 fortfährt (also $\dots, 2, 3, 5, 7, 11, 13, 0, 1, 4, 6, 8, 9, 10, 12, 14, 15, 2, 3 \dots$).

Gehen sie folgendermassen vor:

- (a) Definieren Sie die benötigten Schaltfunktionen.
- (b) Minimieren Sie diese Funktionen m.H. von Karnaugh-Diagrammen
- (c) Entwerfen Sie m.H. der minimierten Funktionen ein Schaltwerk für den Ringzähler.

Aufgabe 5: Assembler (14 Punkte)

Sei folgendes WE32100 Assembler-Programm gegeben:

```
start:  pushw %r8                                d:      save %r7
        pushw %r7                                movw &0x0, %r0
        call -8(%sp), d                          movw (%ap), %r7
stop:   ...                                     movw 4(%ap), %r8
                                           loop:  subw %r8, %r7
                                           blb done          # springe zu done,
                                                                         wenn %r7 < %r8
                                           inc %r0
                                           jmp loop
                                           done:  restore %r7
                                           ret
```

Beim Programmstart sei das Register `%r7` mit `0x43`, das Register `%r8` mit `0xde` geladen. Der Programmlauf beginne bei `start`.

(a) Was berechnet die Prozedur `d`? (8 P.)

(b) Welchen Wert hat das Register `%r0`, wenn der Programmzähler `stop` erreicht? (2 P.)

(c) Der Assembler-Befehl `save` reserviert auf dem Stack Platz für alle Register, obwohl nicht immer alle dann auch tatsächlich dort gesichert werden. Geben sie dafür eine vernünftige Erklärung. (4 P.)