

INSTITUT FÜR MATHEMATISCHE MASCHINEN  
 UND DATENVERARBEITUNG  
 Lehrstuhl für Rechnerstrukturen (IMMD III)  
 Universität Erlangen-Nürnberg  
 Martensstr. 3, 91058 Erlangen

01.03. 1999

Klausur  
 zu

**"Organisation und Technologie von Rechensystemen 4"**

.....                      .....                      .....  
 Matrikelnummer            Geb.-Datum            Vorname                      Name

- Außer Taschenrechner sind keine Hilfsmittel erlaubt.
- Legen Sie den Ausweis (mit Lichtbild!) griffbereit auf den Platz.
- Dieses Aufgabenheft umfaßt 14 Seiten. Überprüfen Sie die Vollständigkeit.
- Gesondert beigelegte Blätter werden nicht bewertet!
- Schreiben Sie deutlich! Unleserliches wird nicht bewertet!
- Es darf nicht mit der Farbe rot geschrieben werden!
- Schreiben Sie die Antworten jeweils in den freien Raum hinter den Fragen; sollte der Platz nicht ausreichen, bitten Sie die Aufsicht um weitere Formblätter. Hier können auch Nebenrechnungen o. ä. stehen.
- Bekanntgabe der Ergebnisse: Aushang in den Semesterferien
- Einsichtnahme: Siehe Aushang
- Bei Bestehen wird der Schein direkt an das Prüfungsamt geschickt!

Durch meine Unterschrift bestätige ich

- den Empfang der vollständigen Klausurunterlagen
- die Kenntnisnahme der obigen Informationen.

Ich bin damit einverstanden, daß mein Prüfungsergebnis der Klausur zu OTR 4 unter Angabe der Matrikel-Nummer veröffentlicht wird.

Erlangen, den 01.03. 1999.....

(Unterschrift)

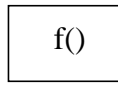
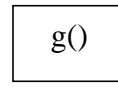
Aufgabe	1	2	3	4	5	6	Summe
max. Punktzahl	7	9	4	6	5	5	36
erreichte Punktzahl							

**Aufgabe 1: (Automaten)****(7 Punkte)**

Sie haben folgende Elemente zur Verfügung:



Zustandsspeicher

Übergangs-  
funktion

Ausgabefunktion

1a) Setzen sie diese Elemente zu einer schematischen Darstellung eines MOORE-Automaten zusammen.

1b) Setzen sie diese Elemente zu einer schematischen Darstellung eines MEALY-Automaten zusammen.

1c) Wieviele (vollständige) MOORE-Automaten mit  $x$  Eingabewerten,  $y$  Zuständen und  $z$  Ausgabewerten gibt es? Geben Sie die Herleitung des Ergebnisses mit an!

1d) Entwerfen Sie einen 3-Bit-Zähler mit folgenden Eigenschaften:

Der Zähler soll zwei Betriebsarten haben. Im ersten Fall wird in Einerschritten aufwärts gezählt, im zweiten in Zweierschritten abwärts. In beiden Fällen wird modulo der Anzahl der darstellbaren Zustände gezählt (also beim Aufwärtszählen folgt auf das Maximum die Null; beim Abwärtszählen entsprechend umgekehrt).

Stellen Sie den Zähler als MOORE-Automatengraphen (state diagram) dar.

**Aufgabe 2: (Schaltfunktionen)****(9 Punkte)**

2a) Erläutern Sie das Verfahren von Quine-McCluskey anhand der Minimierung folgender Funktion:

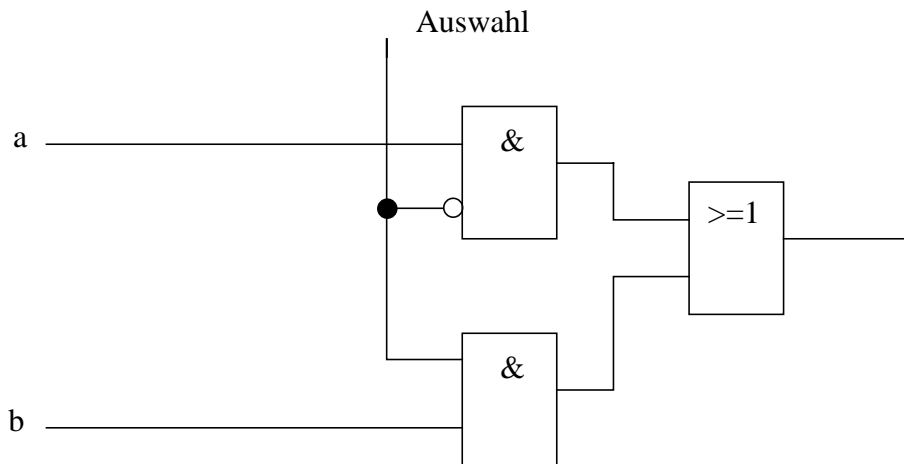
$$f(a, b, c) = abc + a\bar{b}c + \bar{a}bc + \bar{b}\bar{c}$$

2b) Bekanntlich lässt sich jede Schaltfunktion unter ausschließlicher Verwendung von NAND-Gattern mit zwei Eingängen realisieren. Formen Sie die Exklusiv-Oder-Funktion entsprechend um und geben Sie das minimale Schaltnetz an.

Hinweis: Überlegen Sie sich, in welchem Term sich die Oder-Funktion von der Exklusiv-Oder-Funktion unterscheidet!

2c) Zeichnen Sie einen geordneten binären Entscheidungsbaum für die Exklusiv-Oder-Funktion.

- 2d) Bekanntlich lässt sich aus einem Entscheidungsbaum ein Schaltnetz generieren, indem die Knoten durch Multiplexer ersetzt werden. Ein Multiplexer ist wie folgt aufgebaut:



Ersetzen Sie in dem Entscheidungsbaum aus Aufgabe 2c) die Knoten durch diesen Multiplexer und zeichnen Sie das entstehende Schaltnetz. Lassen Sie dabei Tautologien weg.

**Aufgabe 3: (MC68HC05-Assembler)****(4 Punkte)**

Beim Aufruf des folgenden Unterprogramms (mit: `jsr unterprog`) stehen in den Speicherzellen OP1High der Wert 4 und in OP1Low der Wert 250 (Hex FA). Am Eingang porta liege der Bitvektor 00011001 an.

```
unterprog:
    lda #8                ;load a with immediate value
    sta temp              ;store a in RAM
    lda OP1High           ;load a with value from RAM
    ldx OP1Low            ;load x with value from RAM
M1: lslx                  ;logical left shift x to carry
    rola                  ;rotate left a through carry
    bcs M2                ;branch if carry set
    cmp porta             ;compare: a-porta, set flags only
    bcs M3                ;
M2: sub porta             ;a ← a - porta
    incx                  ;increment x
M3: dec temp              ;decrement temp
    bne M1                 ;branch if not equal (not zero)
    rts                   ;return
```

Was ist die Funktion dieses Unterprogramms? Welche Werte stehen nach der Rückkehr im Accu (a) und im Indexregister (x)?

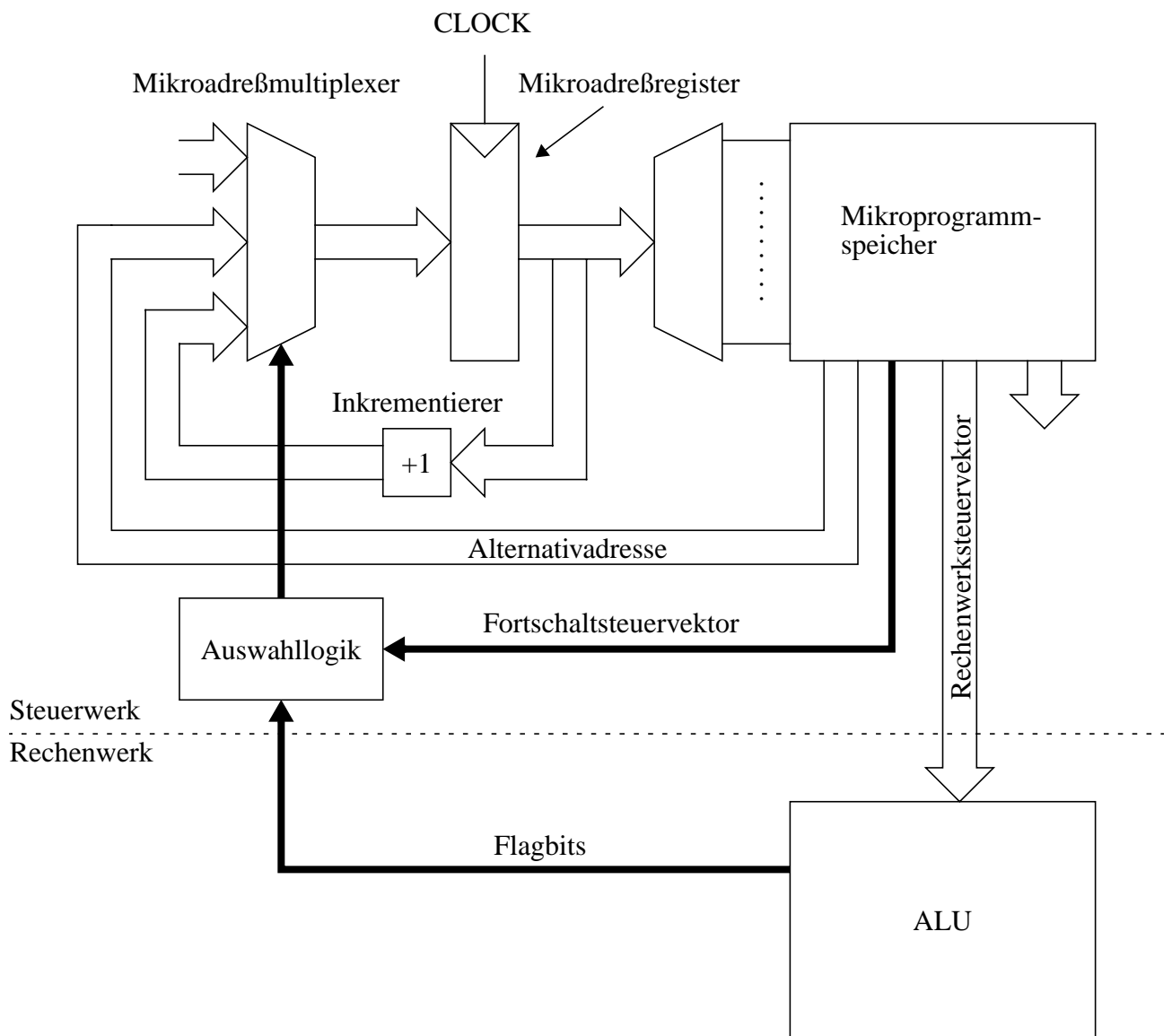


**Aufgabe 5: (Mikroprogrammsteuerung)**

**(5 Punkte)**

Ein mikroprogrammiertes Steuerwerk mit dem im Bild angegebenen Aufbau hat folgende maximale Verzögerungszeiten für die einzelnen Elemente:

Register	von der positiven Taktflanke (CLOCK) bis zum Gültigwerden der Ausgänge	1 ns
Mikroprogramm Speicher	Zugriffszeit	20 ns
ALU	vom Anlegen des Steuercodes bis zum Gültigwerden der Flags	15 ns
Auswahllogik		2 ns
Multiplexer		2 ns
Inkrementierer		12 ns



5a) Wie groß ist die maximal zulässige Frequenz des Taktes (CLOCK)? Geben Sie die Herleitung an!

5b) Zeichnen Sie in das Bild des Steuerwerkes ein mit CLOCK getaktetes Mikroinstruktionsregister ein und geben Sie die neue minimale Zykluszeit an! Geben Sie die Herleitung an!

**Aufgabe 6: (Mikroprogrammierung)****(5 Punkte)**

Ein mikroprogrammiertes Steuerwerk interpretiert jeden Maschinenbefehl durch eine Folge von Mikroschritten, die im zugehörigen Mikroprogramm festgelegt sind.

Für das hier betrachtete Steuerwerk gelte:

Zu Beginn jedes Mikroprogramms stehe der Opcode des Maschinenbefehls im Instruktionsregister IR und der Programmcounter PC zeige auf das Folgewort im Maschinenprogramm. Jedes Mikroprogramm terminiert mit einem Einsprung in das Mikroprogramm, das durch den Inhalt von IR spezifiziert ist. Das Speicherinterface bestehe aus dem Speicheradressregister SAR und dem Speicherdatenregister SDR. SP enthalte den Stackpointer, R sei ein Hilfsregister. Eine Konstante werde mit # bezeichnet.

Ein Mikrobefehl für einen 1-Bus-Datenpfad hat folgende Steuerfelder:

**BUS:** Der erste Operand bezeichnet die Quelle, die auf den Bus geschaltet wird. Der zweite Operand bezeichnet die Zielregister, die am Zyklusende das Datum vom Bus übernehmen.

**ALU** spezifiziert die Verknüpfung zwischen den ALU-Eingangsregistern A und B. Das Ergebnis wird am Zyklusende im Ergebnisregister C gespeichert.

**SPST** steuert die Speicherzugriffe: read lädt am Zyklusende das SDR mit dem durch SAR adressierten Wert; write schreibt den Inhalt von SDR in den Speicher.

**NEXT** bestimmt die Mikrobefehlsfolgeadresse.

Das folgende Mikroprogramm interpretiert den Maschinenbefehl **jump\_subroutine sprungziel** durch eine Steuersequenz.

BUS		ALU	SPST	NEXT
Quelle	Ziele			
PC	A, SAR	nop	nop	+1
#2	B	nop	read	+1
SDR	R	add	nop	+1
SP	A, SAR	nop	nop	+1
C	SDR	sub	nop	+1
C	SP	nop	write	+1
R	A, SAR	nop	nop	+1
-	-	add	read	+1
SDR	IR	nop	nop	+1
C	PC	nop	nop	jump(IR)



**Schmierblatt: diese Seite wird nicht bewertet**

**Schmierblatt: diese Seite wird nicht bewertet**