

Klausur zur Vorlesung

"Grundlagen der technischen Informatik" und "Grundlagen der Rechnerarchitektur" Wintersemester 2000/2001

1. Teil: GTI

Der erste Teil (GTI) der Klausur umfasst 6 Aufgaben und hat 11 Seiten, die in maximal 1 Stunde zu bearbeiten sind. **Es sind keine Hilfsmittel zugelassen.** Schreiben Sie Ihre Lösungen nur auf die dafür vorgesehenen Blätter! Verwenden Sie **kein eigenes Konzeptpapier**; notfalls erhalten Sie welches bei der Aufsicht.

Lassen Sie die Aufgaben und ihre Lösungen zusammengeheftet!

Schreiben Sie auf jedes Blatt (auch auf das Konzeptpapier) in Blockschrift Ihren **Namen** und Ihre **Matrikelnummer**.

Bei mehreren präsentierten Lösungen wird die Aufgabe nicht gewertet! Streichen Sie daher bei Angabe mehrerer Lösungsansätze die nicht zu bewertenden Lösungen durch!

Abschreiben und abschreiben lassen oder Hilfe Dritter führt zum Nichtbestehen der Klausur.

Es sind in diesem ersten Teil insgesamt 60 Punkte erreichbar.

Viel Erfolg!

Musterlösung

Vor- und Nachname: _____

Matrikelnummer: _____ Fachbereich: _____

Studienfach (ankreuzen):

Informatik	<input type="checkbox"/>
Ingenieur-Informatik (Elektrotechnik)	<input type="checkbox"/>
Ingenieur-Informatik (Maschinenbau)	<input type="checkbox"/>
Ingenieur-Informatik (Informatik)	<input type="checkbox"/>
Wirtschafts-Informatik	<input type="checkbox"/>
Informatik als Nebenfach	<input type="checkbox"/>
Lehramt Informatik	<input type="checkbox"/>

Aufgabe	1	2	3	4	5	6	Summe
Punkte	9	10	12	8	8	13	60
Erreicht							

Summe	GTI:		GRA:		Gesamt:		Note:	
-------	------	--	------	--	---------	--	-------	--

Name:

Matrikel-Nummer:

Aufgabe 1 (9 min, 9 Punkte): Zahldarstellung / Komplementrechnung / VHDL

- a) Führen Sie die folgenden Rechnungen im 2er Komplement mit 8-stelligen Dualzahlen (inklusive Vorzeichen-Bit) durch:

$$-79 + 55 \qquad -79 - 10$$

Geben Sie an, ob das Ergebnis richtig (gültig) oder falsch (Bereichsüberschreitung) ist.

Achtung: Kennzeichnen Sie Ihre Lösung eindeutig durch evtl. Durchstreichen der ersten Lösung.

Muster:

2er	V						
79	0	1	0	0	1	1	1
+55	0	0	1	1	0	1	1
Übertrag	1	1	1	1	1	1	1
	1	0	0	0	0	1	1
ankreuzen		richtig:			falsch: X		

Lösung:

2er	V						
-79	1	0	1	1	0	0	1
+55	0	0	1	1	0	1	1
Übertrag		1	1		1	1	1
-24	1	1	1	0	1	0	0
ankreuzen		richtig: X			falsch:		

2er	V						
-79	1	0	1	1	0	0	1
-10	1	1	1	1	0	1	1
Übertrag	1	1	1	1			
-89	1	0	1	0	0	1	1
ankreuzen		richtig: X			falsch:		

Ersatztabellen:

2er	V						
-79							
+55							
Übertrag							
ankreuzen		richtig:			falsch:		

2er	V						
-79							
-10							
Übertrag							
ankreuzen		richtig:			falsch:		

Name:

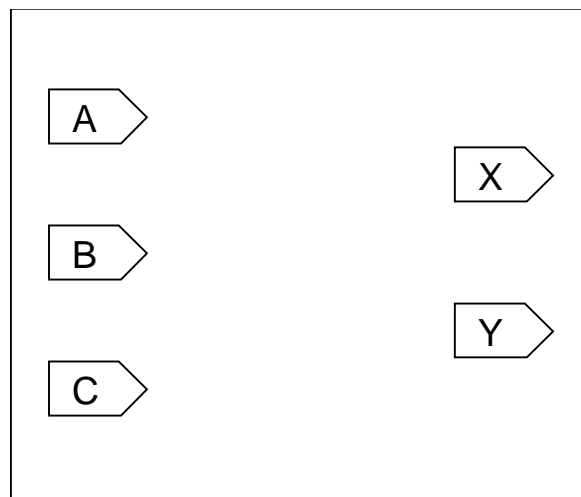
Matrikel-Nummer:

Fortsetzung zu Aufgabe 1):

b) Welche bool'sche Funktionen berechnet das folgende VHDL-Programm?

```
library IEEE;
use IEEE.std_logic_1164.all;
entity sig is
    port ( A, B, C: in std_logic;
          X, Y: out std_logic);
end sig;
architecture SIG_ARCH of sig is
    signal D: std_logic;
    begin
        SIGP: process (A,B,C,D)
        begin
            D <= A;
            X <= C and D;
            D <= B;
            Y <= C and D;
            D <= C;
        end process;
    end SIG_ARCH;
```

Geben Sie das Schaltnetzwerk mit entsprechend benötigten Gattern an:



Hinweis: Beachten Sie, dass:

- Signalzuweisungen erst bei der nächsten Aktivierung des Prozesses wirksam werden.
- Der Prozess SIGP bei jeder Wertänderung von A, B, C, D neu aktiviert wird.

Aufgabe 2 (10 min, 1 + 9 Punkte): Steuerwerk mit PLA

In der unten angegebenen Abbildung ist der Zustandsübergangsgraph für den Automaten zur Überwachung einer Cacheline zur Sicherstellung ihrer Konsistenz nach dem **MESI-Protokoll** eines PowerPC 60x angegeben.

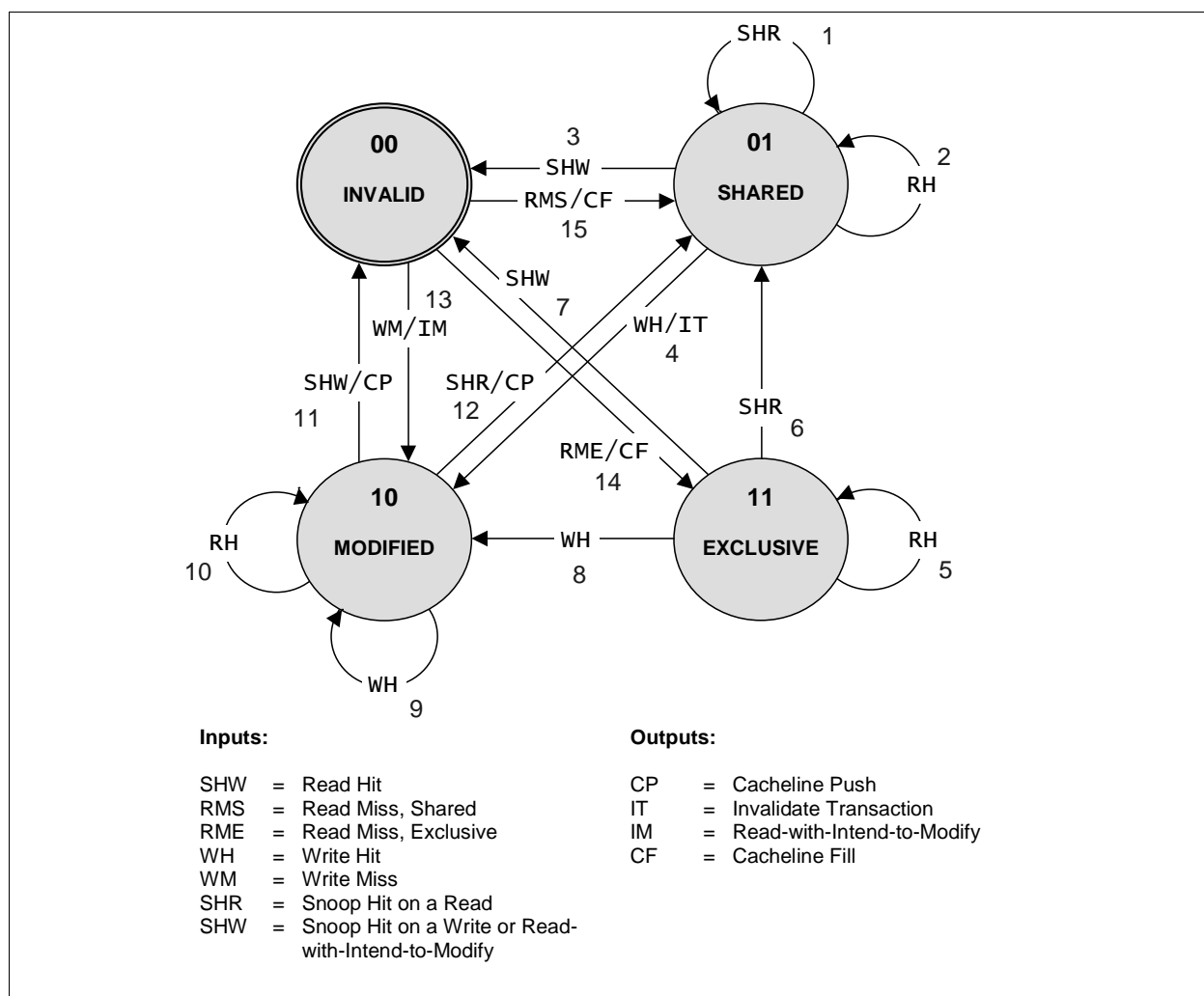
Der Automat besitzt **4 Zustände** (binär kodiert in S_1S_0):

- 00 : Cacheline invalid
- 01 : Cacheline shared
- 10 : Cacheline modified
- 11 : Cacheline exclusive

Der Automat besitzt **7 Eingangssignale** von der Bus-Logik, die den Zustand einer Cacheline beeinflussen. Der Automat kann **4 Ausgangssignale** generieren, die den Prozessor und seine Bus-Logik dann entsprechend steuern

a) Handelt es sich bei dem Automaten (bitte ankreuzen) um einen

- Mealy-Automaten oder
 Moore-Automaten ?



Hinweise: Lassen Sie sich nicht von der vermeintlichen Komplexität des Automaten beeindrucken: **Die Aufgabe ist recht einfach.**

Die Zahlen an den Zustandsübergängen bestimmen **eine Zeile des PLA!**

Zustand 01 bedeutet $S_1 = 0$ und $S_0 = 1$!

Name:

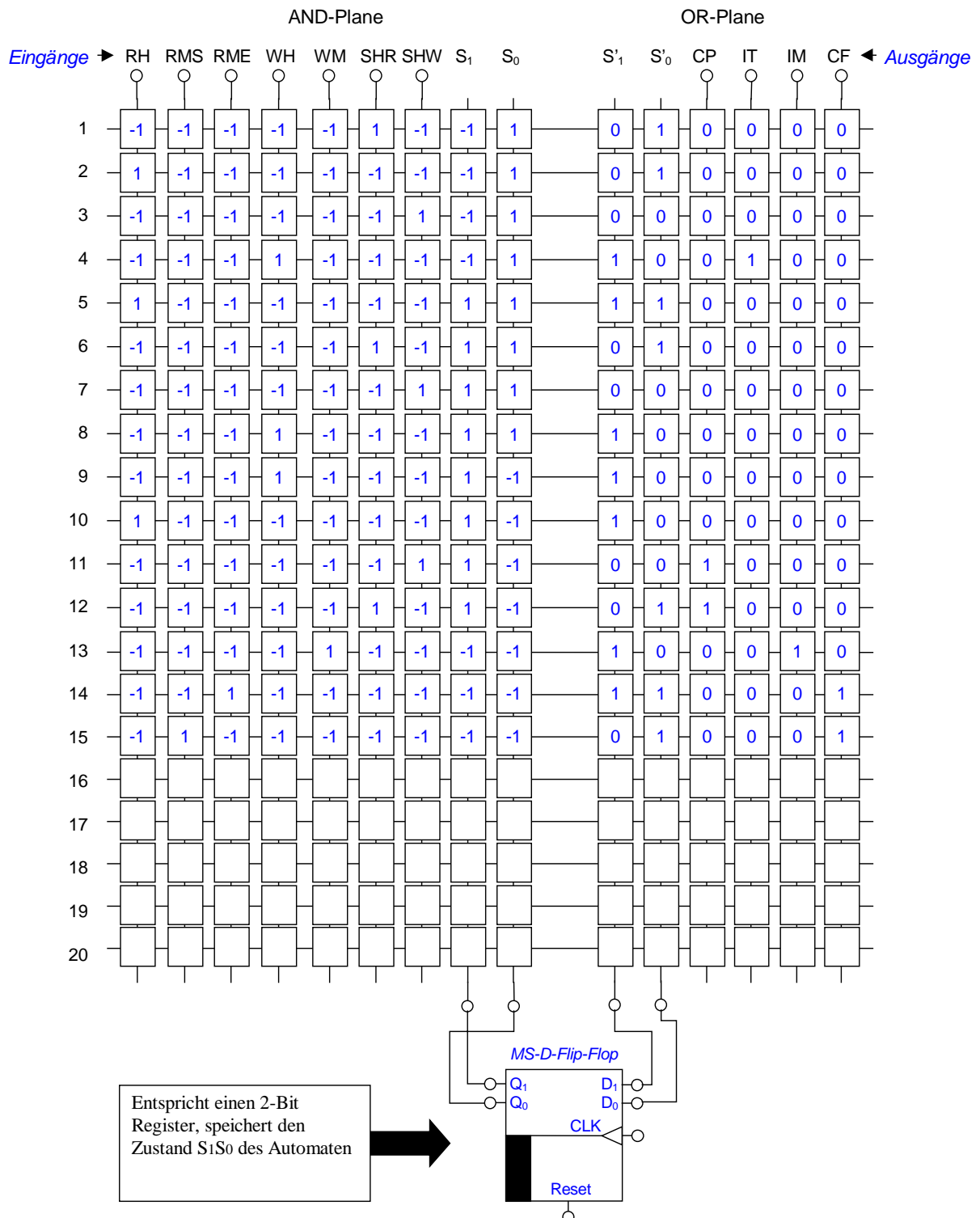
Matrikel-Nummer:

Bei Zustandsübergängen ohne Ausgabe soll der PLA keine Ausgabe erzeugen!
Eine Automatenumwandlung ist nicht erforderlich.

- b) Entwerfen Sie ein Steuerwerk für den angegebenen Automaten. Tragen Sie entsprechend -1, 1 und 0 in die Kästchen des PLA ein. Kästchen nicht benötigter Zeilen lassen Sie frei. Beachten Sie dabei die Zeilenangaben im Zustandsübergangsdiagramm!

Hinweis: Der Automat erhält nur dann einen vollen Taktzyklus an seinem CLK-Eingang, wenn er ein Eingangssignal verarbeiten soll. Es kann zu einem Zeitpunkt nur ein Eingangssignal logisch '1' sein.

Steuerwerk mit PLA:

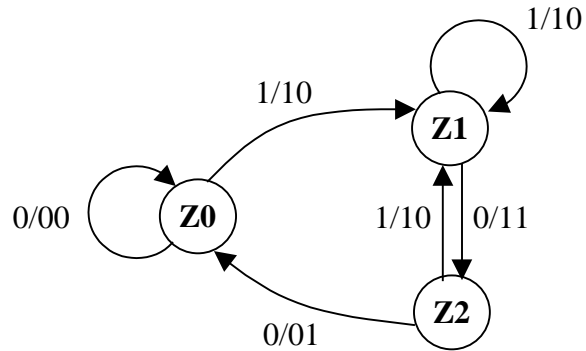


Name:

Matrikel-Nummer:

Aufgabe 3 (12 min, 12 Punkte): Automaten

Wandeln Sie den gegebenen Mealy-Automaten in einen äquivalenten Zustands-minimalen (*reduzierten*) Moore-Automaten um und geben Sie das Ergebnis auch als Zustandsübergangsgraphen an.



Z0, Z1, Z2 sind Zustände;
1/01 steht für Eingabe X = 1, Ausgabe: Y(1) = 0, Y(0) = 1.

Lösung:

Aufgabe 4 (8 min, 8 Punkte): Vereinfachung bool'scher Funktionen

Minimieren Sie die bool'sche Funktion

$$F = \bar{p} \times \bar{y} z + p \times \bar{y} \bar{z} + \bar{p} \times \bar{y} \bar{z} + p \bar{x} \bar{y} \bar{z} + \bar{p} \bar{x} \bar{y} \bar{z}$$

mit Hilfe des **aus der GTI-Vorlesung bekannten** Quine/McClusky Verfahrens. Das Verfahren benötigt hier 2 Iterationen. Geben Sie die Zwischenergebnisse nach den angegebenen Aktionen des Verfahrens in beiden Iterationen an. Bei der Summe der Konsensusse sind **alle** (d.h. **auch eventuell doppelt auftretende**) anzugeben!

Nr.	Aktion	Zwischenergebnisse
	Ausgangsfunktion F	$\bar{p} \times \bar{y} z + p \times \bar{y} \bar{z} + \bar{p} \times \bar{y} \bar{z} + p \bar{x} \bar{y} \bar{z} + \bar{p} \bar{x} \bar{y} \bar{z}$
1	Summe aller „Simplen Konsensusse“	$\bar{p} \times \bar{y} + x \bar{y} \bar{z} + p \bar{y} \bar{z} + \bar{p} \bar{y} \bar{z} + \bar{x} \bar{y} \bar{z}$
	Nach „Streichen von Verlängerungen“ und „Vereinfachen“	$\bar{p} \times \bar{y} + x \bar{y} \bar{z} + p \bar{y} \bar{z} + \bar{p} \bar{y} \bar{z} + \bar{x} \bar{y} \bar{z}$
2	Summe aller „Simplen Konsensusse“	$\bar{y} \bar{z} + \bar{y} \bar{z}$
	Nach „Streichen von Verlängerungen“ und „Vereinfachen“	$\bar{p} \times \bar{y} + \bar{y} \bar{z}$

Ersatztable:

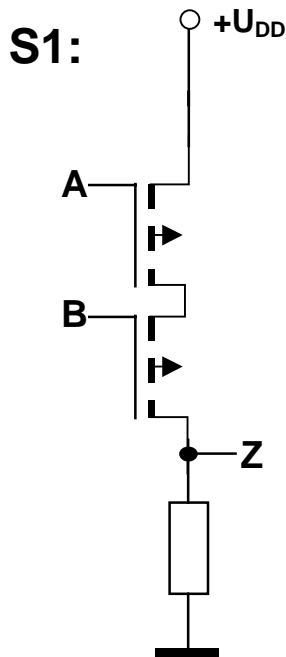
Wenn Sie diese Tabelle anstelle der oberen bewertet haben wollen, so streichen Sie die obere Tabelle durch!

Nr.	Aktion	Zwischenergebnisse
	Ausgangsfunktion F	$\bar{p} \times \bar{y} z + p \times \bar{y} \bar{z} + \bar{p} \times \bar{y} \bar{z} + p \bar{x} \bar{y} \bar{z} + \bar{p} \bar{x} \bar{y} \bar{z}$
1	Summe aller „Simplen Konsensusse“	
	Nach „Streichen von Verlängerungen“ und „Vereinfachen“	
2	Summe aller „Simplen Konsensusse“	
	Nach „Streichen von Verlängerungen“ und „Vereinfachen“	

ACHTUNG Korrekturseite: Bearbeiten Sie anstatt der Original-Seite 8 der Klausur zu GTI diese NEUE Seite!!!

Aufgabe 5 (8 min, 8 Punkte): MOS Schaltungen

a) Ergänzen Sie die untenstehende Arbeitstabelle zur PMOS-Schaltung **S1** und tragen Sie für **Z** die Werte H oder L ein:



Arbeitstabelle zur Schaltung **S1**

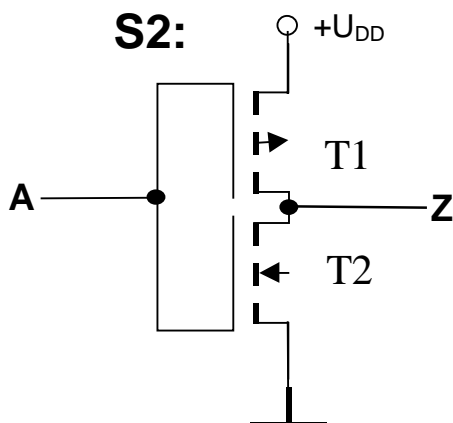
A	B	Z
L	L	L
H	L	L
L	H	L
H	H	H

L ≈ GND H ≈ +U_{DD}

b) Für die Schaltung **S2** soll die untenstehende Tabelle ergänzt werden:

A	T1	T2	Z
H	gesperrt	leitend	L
L	leitend	gesperrt	H

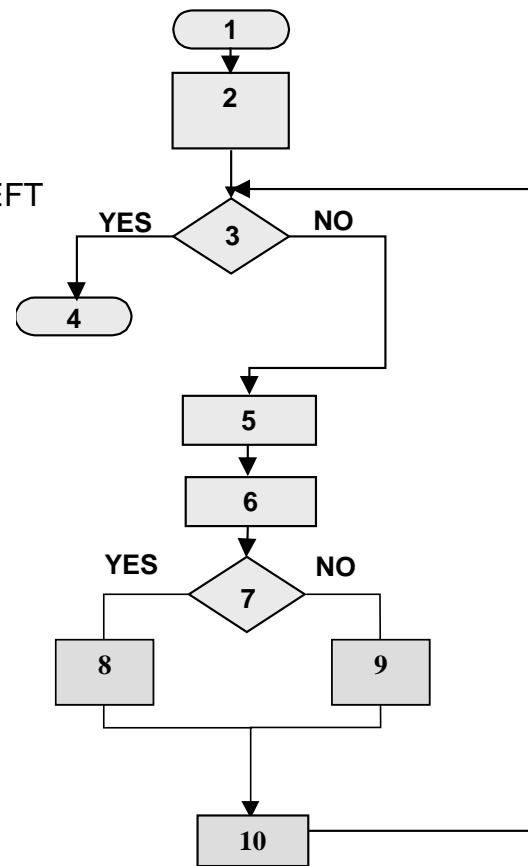
H ≈ +U_{DD} L ≈ GND



Aufgabe 6 (13 min, 13 Punkte): Division

a) Der folgende Ablaufplan implementiert den in der Vorlesung vorgestellten RESTORING Algorithmus zur Division von zwei binären Zahlen X durch Y. Das Ergebnis wird in zwei Registern AC und Q gespeichert. ($AC = AC_n AC_{n-1} \dots AC_0$, $Q = Q_{n-1} Q_{n-2} \dots Q_0$)
 Folgende Befehle sollen den Numerierungen des Ablaufplans zugeordnet werden:

- A) START
- B) END
- C) $i \leq 1$
- D) $AC \leftarrow „0\dots\dots 0“$
- E) $Q \leftarrow X$
- F) $AC \& Q \leftarrow$ SHIFT OF $AC \& Q$ 1 BIT LEFT
- G) $AC \leftarrow AC - Y$
- H) $Q[0] \leftarrow 1$
- I) $AC \leftarrow AC + Y$
- J) $i = n + 1$?
- K) $i \leq i + 1$
- L) $AC[n] = 1$?



Kreuzen Sie die Zuordnungen an:
 (Hinweis: Es ist mehr als 1 Kreuz pro Spalte erlaubt.)

	1	2	3	4	5	6	7	8	9	10
A	X									
B				X						
C		X								
D		X								
E		X								
F					X					
G						X				
H									X	
I								X		
J			X							
K										X
L							X			

Name:

Matrikel-Nummer:

Fortsetzung zu Aufgabe 6):

- b) Wenden Sie den aus der Vorlesung bekannten "Restoring Algorithmus" zur Division auf die Werte $X := 13$ und $Y = 4$ an.
 Das Register Q ist 4 Bit breit, AC besitzt ein 5tes Bit für das intern benötigte Vorzeichen.
Markieren Sie am Ende der Rechnung deutlich wo in den Registern das Ergebnis abzulesen ist!

AC (5 Bit Darstellung)	Q (4 Bit Darstellung)	Nächste Operation	Zyklus i?
00000	1101	Shift left	1
00001	1010		
+11100	1010	AC \leq AC - Y	
11101	1010	RESTORE	
00001	1010	Shift left	
00011	0100		2
+11100	0100	AC \leq AC - Y	
11111	0100	RESTORE	
00011	0100	Shift left	
00110	1000		3
+11100	1000	AC \leq AC - Y	
00010	1000	Q(0) \leq 1	
00010	1001	Shift left	
00101	0010		4
+11100	0010	AC \leq AC - Y	
00001	0011	Q(0) \leq 1	

Name:

Matrikel-Nummer:

Fortsetzung zu Aufgabe 6):

Ersatztable (streichen Sie Ihre falsche Lösung durch!):

AC (5 Bit Darstellung)	Q (4 Bit Darstellung)	Nächste Operation	Zyklus i?
00000	1101		