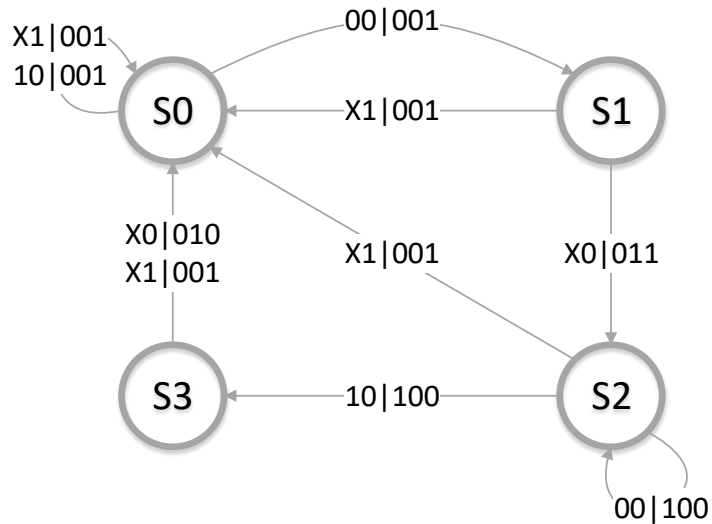


Aufgabe 1: Schaltwerksanalyse

- a) Gegeben Sei der durch den folgenden Zustandsübergangsgraphen definierte Automat mit dem Startzustand S0.



Um welchen Automatentyp handelt es sich? Begründen Sie Ihre Antwort.

Wieviele D-Flipflops werden für die Realisierung des Automaten benötigt? Begründen Sie Ihre Antwort.

Wieviele JK-MS-Flipflops würden für die Realisierung des Automaten benötigt werden? Begründen Sie Ihre Antwort.

Welche Ausgabesequenz resultiert aus der Eingabefolge 10 – 00 – 10 – 00 – 10 – 00

Ergänzen Sie die nachfolgende Zustandsübergangstabelle Dabei entspricht die Zustandskodierung der binären Repräsentation der Zustandsnummern (S0 – 00, S1 – 01 ...):

	$Z_1^{n+1} Z_0^{n+1}$ Eingabe $X_1 X_0$				$Y_2^n Y_1^n Y_0^n$ Eingabe $X_1 X_0$			
$Z_1^n Z_0^n$	00	01	10	11	00	01	10	11
00								
01								
10								
11								

- b) Es sei folgende Zustandsübergangsfunktion gegeben:

$$Z_1^{n+1} = \bar{Z}_1 Z_0 \bar{X}_0 + Z_1 \bar{Z}_0 \bar{X}_0.$$

Dieser Automat soll mit Hilfe von JK-MS-Flipflops realisiert werden. Ermitteln Sie die entsprechende Ansteuergleichung für das zum Zustandsbit Z_1 korrespondierende Flipflop.

Geben Sie nun die zum Zustandsbit Z_1 korrespondierende Ansteuergleichung für die Realisierung mit einem D-Flipflop.

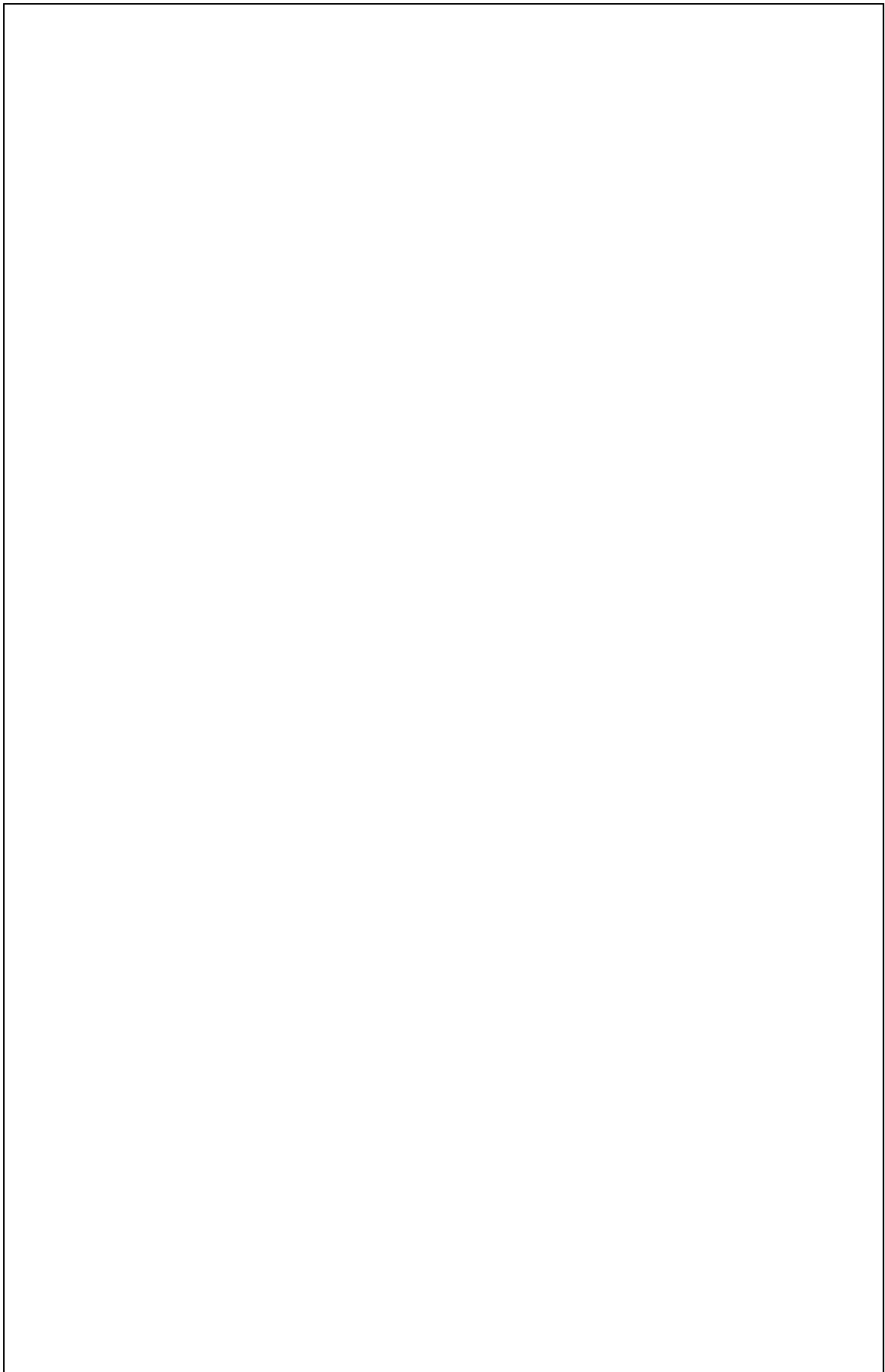
- c) Ein für die Eingabe X definierter Automat sei durch die Zustandsübergangsfunktionen $Z_0^{n+1} = X \bar{Z}_1^n$ und $Z_1^{n+1} = X Z_0^n$ sowie die Ausgabefunktion $y = Z_1 + Z_0$ gegeben.

Um welchen Automatentyp handelt es sich? Geben Sie eine Begründung an.

Matrikelnummer: _____Studiengang: _____

Matrikelnummer: _____Studiengang: _____

Geben Sie den Zustandsübergangsgraphen des Automaten an.



Realisieren Sie den Automaten als LogiFlash Schaltung unter der ausschließlichen Verwendung beliebig vieler der folgenden LogiFlash Komponenten: Buttons, Lampen, Und- & Oder-Gatter mit beliebig vielen gegebenenfalls negierten Eingängen und D-Flipflops (siehe Abbildung 2).

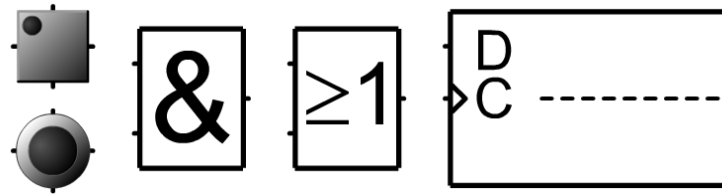


Abbildung 2: Erlaubte Komponenten

d) Gegeben sei folgende Zustandsübergangstabelle:

$Z_1^{n+1} Z_0^{n+1}$				
	ab			
$Z_1^n Z_0^n$	00	01	10	11
00	00	00	xx	01
01	11	01	xx	00
10	00	11	xx	11
11	11	00	xx	10

Geben Sie die Zustandsübergangsfunktion für Z_1^n in **disjunktiver** und für Z_0^n in **konjunktiver** Minimalform an! Nutzen Sie für die Bestimmung der Übergangsfunktionen folgende KV-Diagramme. Füllen Sie diese komplett aus und kennzeichnen Sie Ihre Minimierungen!

Z_1^{n+1}		$(a,b)^n$			
		00	01	11	10
$(Z_1 Z_0)^n$	00				
	01				
	11				
	10				

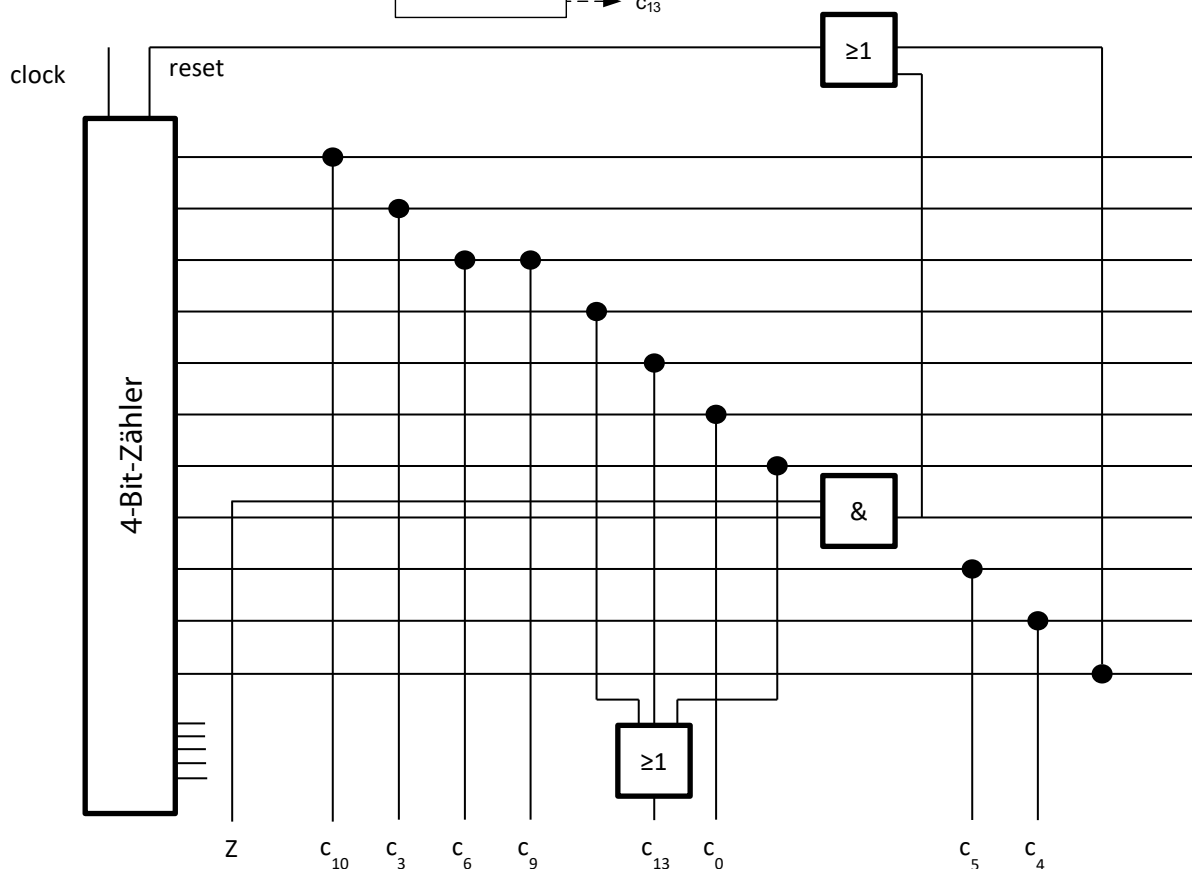
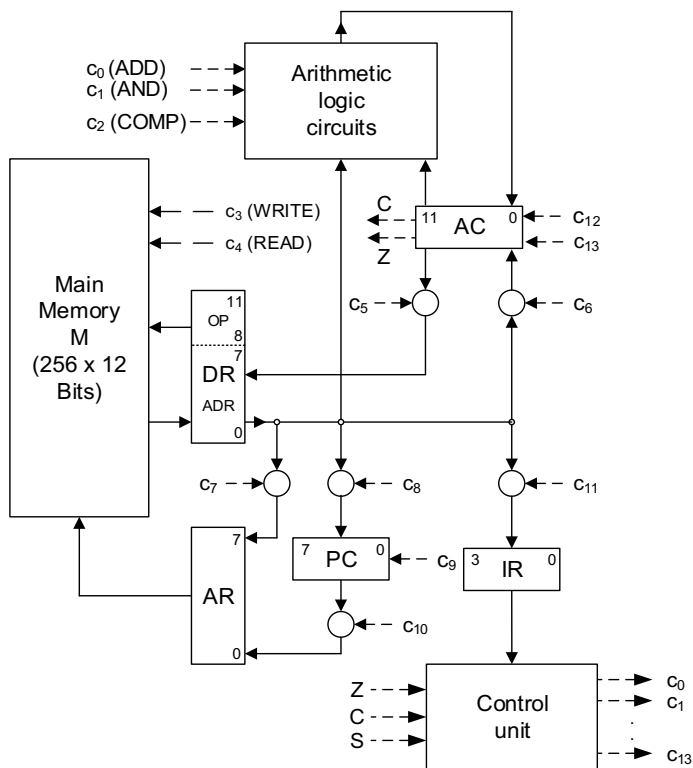
Z_0^{n+1}		$(a,b)^n$			
		00	01	11	10
$(Z_1 Z_0)^n$	00				
	01				
	11				
	10				

Zustandsübergangsfunktionen:

$\Sigma_{A1} =$ /17 Pkt

Aufgabe 2: Entwurf auf Registertransfer Ebene

- a) Für das angegebene Operationswerk ist ein festverdrahtetes Steuerwerk mit einem 4-Bit-Binärzähler gegeben. Das Z-Flag gibt an, ob $AC = 0$ erfüllt ist ($AC = 0 \Rightarrow Z = 1$).



Beschreiben Sie das durch den Automaten definierte Verhalten als RT-Programm bzw. in RT-Notation. Achten Sie auf das Timing!

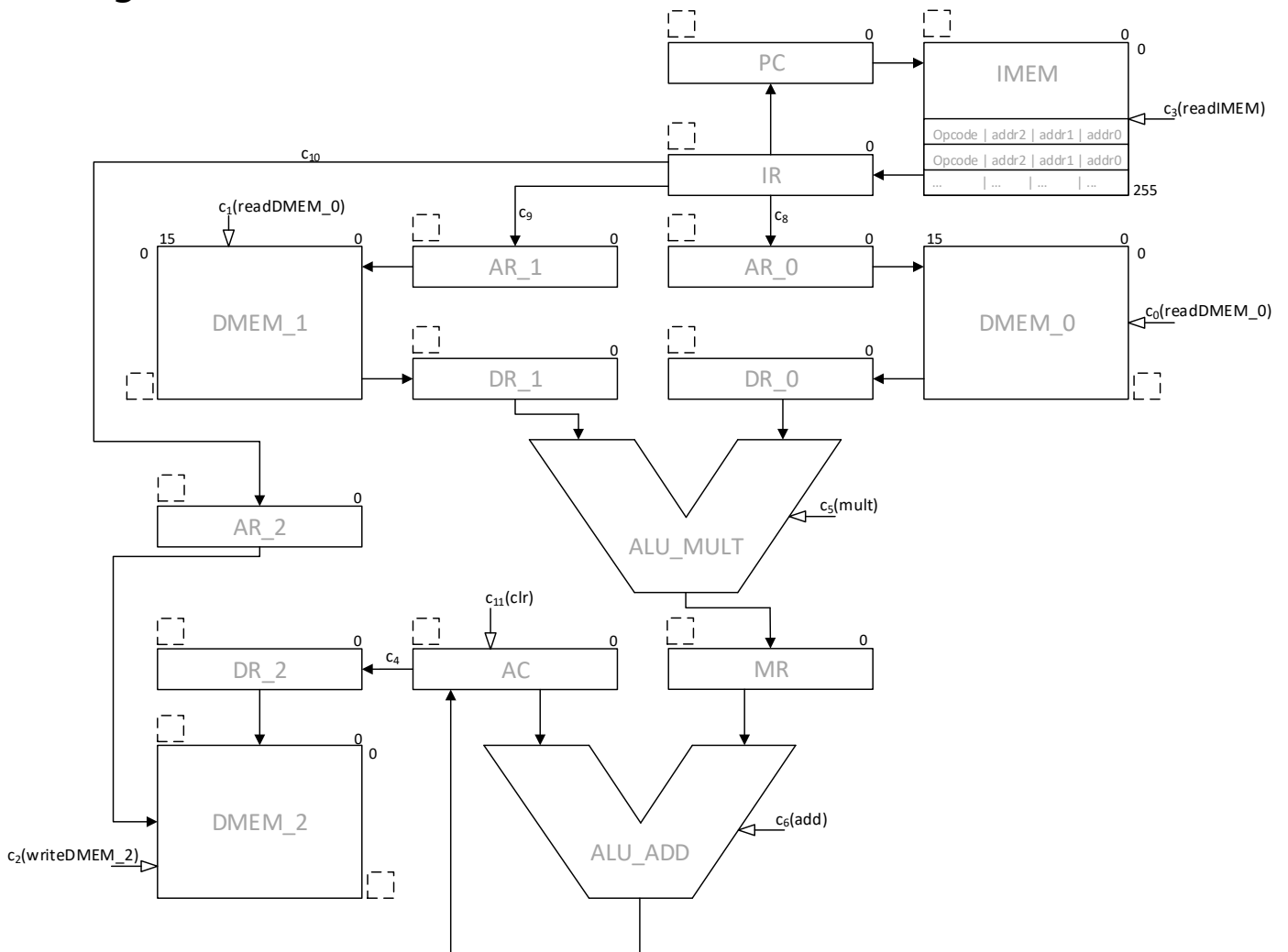
- b) Geben Sie die semantische Bedeutung (KEINE zeilenweise Beschreibung des RT-Codes) eines Durchlaufs an!

- c) Was ist der Nutzen der Abfrage des Z-Flags in diesem Programm? Wie könnte man diesen Nutzen noch verbessern?

- d) Wie sieht der Speicherinhalt aus, wenn das Programm unendlich lange läuft?

$\Sigma_{A2} =$ /10 Pkt

Aufgabe 3: CPU-Kontrolleinheit



Das oben vorgegebene Blockschaltbild zeigt das Operationswerk der in dieser Aufgabe gegebenen mikroprogrammierten CPU. Das Operationswerk ist darauf spezialisiert Matrixmultiplikationen ($C=A \cdot B$) mit einer maximalen Größe von 9×9 zu berechnen. Die Ausgangsmatrizen (A, B) sind in dem Memories **DMEM_0** und **DMEM_1** gespeichert. Das Ergebnis soll im Memory **DMEM_2** gespeichert werden.

Die **ALU_ADD** beherrscht ausschließlich die Addition (**add**), die **ALU_MULT** beherrscht ausschließlich die Multiplikation (**mult**).

Eine Registertransferbeschreibung einiger Befehle der CPU ist Ihnen vorgegeben:

Opcode	Befehl	Beschreibung
0	LOAD0 addr0	Lädt den Speicherinhalt aus DMEM_0 unter der Adresse addr0 in das Register DR_0 .
1	LOAD1 addr1	Lädt den Speicherinhalt aus DMEM_1 unter der Adresse addr1 in das Register DR_1 .
2	STORE addr2	Speichert den Registerinhalt von AC unter der Adresse addr2 im Speicher DMEM_2 ab und löscht Wert von AC .
3	ADD	Addiert den Wert von AC mit dem Ergebnis aus dem Register MR und schreibt das Ergebnis zurück in AC .
4	MULT	Multipliziert die Werte von DR_0 , DR_1 und schreibt das Ergebnis in MR .
5	LOAD addr0 addr1	Lädt den Speicherinhalt aus DMEM_0 unter der Adresse addr0 in das Register DR_0 und lädt gleichzeitig den Speicherinhalt aus DMEM_1 unter der Adresse addr1 in das Register DR_1 .
6	CALC addr0 addr1	Kombiniert die Befehle LOAD addr0 addr1 , ADD und MULT. Dabei soll soweit wie möglich parallelisiert werden. Die ALU_MULT und ALU_ADD arbeiten somit auf unterschiedlichen Ausgangsdaten.

Die folgenden Aufgaben sollen alle in dem aus der Vorlesung und Übung bekannten RT-Code umgesetzt werden.

- Ergänzen Sie im Blockschaltbild die fehlenden Indizes der Register- und Speicherbreiten in den vorgesehenen Kästchen (gestrichelt). Die Breiten und Tiefen sollen so klein wie möglich gewählt werden, sollen aber alle notwendigen Werte speichern können.
- Deklarieren Sie alle benötigten Speicher und Register.

Matrikelnummer: _____Studiengang: _____

- c) Nehmen Sie notwendige Initialisierungen unter dem Label `INIT` vor und implementieren Sie unter dem Label `FETCH` die Befehlsholphase. Verwenden Sie die Label `LOAD0`, `LOAD1`, `STORE`, `ADD` und `MULT`.

- d) Implementieren Sie unter dem Label `LOAD0` und `LOAD1` das zugehörige Verhalten aus der Tabelle.

- e) Implementieren Sie unter dem Label `STORE` das zugehörige Verhalten aus der Tabelle.

- f) Implementieren Sie unter dem Label `ADD` das zugehörige Verhalten aus der Tabelle.

- g) Implementieren Sie unter dem Label `MULT` das zugehörige Verhalten aus der Tabelle.

Hinweis: Eine Multiplikation können Sie mit „`*`“ darstellen.

Matrikelnummer: _____Studiengang: _____

- h) Implementieren Sie die angegebenen Teile des horizontalen Mikroprogramms. Verwenden Sie hierfür die Angabe der Befehlstabelle. FETCH habe die Sprungadresse 0001. Zur Bearbeitung stehen Ihnen ausschließlich die folgenden Condition Select Signale zur Verfügung:

Condition Select	Funktion
0	Nicht springen
1	Springe immer

i.

STORE addr2	Speichert den Registerinhalt von AC unter der Adresse addr2 im Speicher DMEM_2 ab.
--------------------	---

current_addr				CS	next_addr				c_horiz											
									c11	c10	c9	c8	c7	c6	c5	c4	c3	c2	c1	c0
0	1	0	0																	
0	1	0	1																	
0	1	1	0																	
0	1	1	1																	

ii.

LOAD addr0 addr1	Lädt den Speicherinhalt aus DMEM_0 unter der Adresse addr0 in das Register DR_0 und lädt gleichzeitig den Speicherinhalt aus DMEM_1 unter der Adresse addr1 in das Register DR_1 .
-------------------------	--

current_addr				CS	next_addr				c_horiz											
									c11	c10	c9	c8	c7	c6	c5	c4	c3	c2	c1	c0
1	0	0	0																	
1	0	0	1																	
1	0	1	0																	
1	0	1	1																	

iii.

CALC addr0 addr1	Kombiniert die Befehle LOAD addr0 addr1 , ADD und MULT. Dabei soll soweit wie möglich parallelisiert werden. Die ALU_MULT und ALU_ADD arbeiten somit auf unterschiedlichen Ausgangsdaten.
-------------------------	--

current_addr				CS	next_addr				c_horiz											
									c11	c10	c9	c8	c7	c6	c5	c4	c3	c2	c1	c0
1	1	0	0																	
1	1	0	1																	
1	1	1	0																	
1	1	1	1																	

Matrikelnummer: _____Studiengang: _____

- i) Geben Sie das Mapping ROM für die Implementierungen aus Aufgabenteil h an.

Opcode	Sprungadresse			

- j) Geben Sie eine Befehlsfolge an, welche eine Matrixmultiplikation von zwei 2x2 Matrizen mit einer minimalen Zahl an Befehlen durchführt.

Hinweis:

Die Matrizen sind zeilenweise in den Speicher (**DMEM_0**, **DMEM_1**) übertragen worden, sodass die Stelle oben links im Speicher unter der Adresse 0 steht. Ein Beispiel finden Sie im Folgenden für eine Matrix der Größe 2x2 (der Wert entspricht der Adresse):

0	1
2	3

$\Sigma_{A3} =$ /20 Pkt

Aufgabe 4: Allgemeine Fragen

Beantworten Sie die folgenden Fragen.

- a) Was ist der Unterschied zwischen einem Schaltnetz und einem Schaltwerk?

- b) Was ist ein Minterm, was ein Koppelterm?

- c) Wenn man nur einen binären Addierer zur Verfügung hat, wie kann man damit auch subtrahieren, multiplizieren und dividieren (mit Rest)?

$\Sigma_{A4} =$ /3 Pkt