

Klausur zur Vorlesung

Grundlagen der Technischen Informatik / Digitaltechnik (GTI/DT)

Prof. Marco Platzner
Fachgebiet Technische Informatik
Universität Paderborn

23.03.2015

- Die Bearbeitungsdauer beträgt für alle Studenten **90 Minuten**. Es sind **alle 5 Aufgaben** zu bearbeiten.
- Es sind keine Hilfsmittel zugelassen.
- Schreiben Sie nicht mit Bleistift oder Rotstift.
- Verwenden Sie kein eigenes Papier. Bei Bedarf bekommen Sie Papier bei der Klausuraufsicht.
- 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! Verwenden Sie kein Tipp-Ex.
- Abschreiben und abschreiben lassen oder Hilfe Dritter führt zum Nichtbestehen der Klausur.

Nachname: _____

Vorname: _____

Matrikelnummer: _____

Studiengang: _____

Aufkleber

Aufgabe	1	2	3	4	5	Σ
Punkte	15	15	15	20	25	90
Erreicht						

Aufgabe 1 (Multiple Choice)

[15 Punkte]

Bei den folgenden Fragen können keine, eine oder mehrere Antworten richtig sein. Kreuzen Sie die richtigen Antworten deutlich an.

(a) Welche der folgenden Grundgatter sind assoziativ?

☐ AND

☐ NAND

☐ OR

☐ XOR

(b) Welche der folgenden Gattermengen sind vollständig?

☐ {AND, OR, NOT}

☐ {AND, NOT}

☐ {XOR, NOT}

☐ {NAND}

(c) Eine binäre Zahl mit 6 Bit hat einen Wertebereich von $[-31, +31]$. Um welche Zahlendarstellung kann es sich handeln?

☐ 1er-Komplement

☐ 2er-Komplement

☐ Exzess-Code

☐ Vorzeichen/Betrag

NAME:

Matrikelnummer:

(d) Ein Code hat eine Hamming-Distanz von 3. Welche Aussagen sind korrekt?

- ☐ Zwei Fehler können erkannt werden.
- ☐ Es gibt mindestens ein Paar von Codeworten mit einer Hamming-Distanz von 3.
- ☐ Zwei Fehler können korrigiert werden.
- ☐ Jedes Paar von Codeworten hat eine Hamming-Distanz von 3.

(e) Beim Booth Multiplizierer wird ein Operand (Y) umcodiert, um Additionen/Subtraktionen einzusparen. Bei welchen der folgenden Operanden Y werden durch das Booth Recoding Additionen/Subtraktionen eingespart?

- ☐ $Y = 01110000$
- ☐ $Y = 01111110$
- ☐ $Y = 01101100$
- ☐ $Y = 01010101$

Aufgabe 2 (VHDL)

[15 Punkte]

- (a) In Listing 1 a) bis d) sind vier VHDL Entitäten samt ihrer Implementierung angegeben. Welche Funktionen implementieren diese Entitäten? Wählen Sie aus folgender Liste möglicher Funktionen:

Multiplexer	Demultiplexer
1-Bit Full Adder	Ripple-Carry-Adder mit Lookahead
Ripple-Carry-Adder	2^4 x 1-Bit ROM
2^3 x 1-Bit RAM	16-Bit DRAM
DDR-RAM	JK-Flip-Flop
D-Latch	Mealy-Zustandsautomat
Moore-Zustandsautomat	ALU
Schieberegister	D-Flip-Flop
Komparator	Turing Maschine

Notieren Sie hier Ihre Lösungen:

a) _____

b) _____

c) _____

d) _____

- (b) Entwickeln Sie nun Ihre eigenen VHDL Schaltungen und implementieren Sie folgende Funktionen. Nutzen Sie die Vorlagen in Listing 2 a) und b).

- a) AND-Gatter mit vier 1-Bit Eingängen und einem 1-Bit Ausgang
- b) D-Flip-Flop mit Reset (taktflankengesteuert, *nicht* aus Latches aufgebaut)

NAME:

Matrikelnummer:

Listing 1 a)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity mystery_a is
    port ( x,y,ci : in  STD_LOGIC;
           s,co : out  STD_LOGIC);
end entity;

architecture Behavioral of mystery_a
is
    signal sum1, sum2 : std_logic;
    signal carry1, carry2 : std_logic;
begin

    sum1    <= x xor y;
    carry1 <= x and y;

    sum2    <= sum1 xor ci;
    carry2 <= sum1 and ci;

    s <= sum2;
    co <= carry1 or carry2;

end Behavioral;
```

Listing 1 b)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity mystery_b is
    port( s,a,b: in  std_logic;
          z : out  std_logic);
end entity;

architecture Behavioral of mystery_b
is
begin

process (s, a, b)
begin
    case s is
        when '0' => z <= a;
        when '1' => z <= b;
        when others => z <= '0';
    end case;
end process;

end Behavioral;
```

Listing 1 c)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity mystery_c is
    Port (
        d : in  STD_LOGIC;
        e : in  STD_LOGIC;
        q : out  STD_LOGIC);
end entity;

architecture Behavioral of mystery_c
is
begin

-- VHDL Signale haben
-- 'Speichereigenschaften'.
-- Fuer e='0' behaelt q seinen
-- alten Wert.

q <= d when e = '1';

end Behavioral;
```

Listing 1 d)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity mystery_d is
    Port (
        a : in
            STD_LOGIC_VECTOR(3 downto 0);
        d : out  STD_LOGIC);
end entity;

architecture Behavioral of mystery_d
is
    constant r :
        std_logic_vector(15 downto 0)
        :=X"ABCD";
begin

d <= r(conv_integer(a));

end Behavioral;
```

Listing 2 a)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity and4 is
    port (
        _____
        _____
    );
end entity;

architecture Behavioral of and4
is
    _____
begin
    _____
    _____
    _____

end Behavioral;
```

Listing 2 a) (ERSATZ)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity and4 is
    port (
        _____
        _____
    );
end entity;

architecture Behavioral of and4
is
    _____
begin
    _____
    _____
    _____

end Behavioral;
```

Listing 2 b)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity dff is
    port (
        _____
        _____
    );
end entity;

architecture Behavioral of dff
is
    _____
begin
    _____
    _____
    _____
    _____
    _____
    _____
    _____
    _____

end Behavioral;
```

Listing 2 b) (ERSATZ)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity dff is
    port (
        _____
        _____
    );
end entity;

architecture Behavioral of dff
is
    _____
begin
    _____
    _____
    _____
    _____
    _____
    _____
    _____
    _____

end Behavioral;
```

NAME:

Matrikelnummer:

Aufgabe 3 (Quine McCluskey)

[15 Punkte]

Gegeben seien die Eingänge x_3, x_2, x_1 , und x_0 und je eine grüne (G) und eine rote (R) LED. Die jeweilige LED soll aufleuchten, wenn der korrespondierende Wert in der folgenden Wahrheitstabelle 1 annimmt. Es soll nun die entsprechende Schaltung optimiert werden.

x_3	x_2	x_1	x_0	G	R
0	0	0	0	1	0
0	0	0	1	1	0
0	0	1	0	1	0
0	0	1	1	0	1
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	0	1
0	1	1	1	1	0
1	0	0	0	1	0
1	0	0	1	0	1
1	0	1	0	1	0
1	0	1	1	0	1
1	1	0	0	0	1
1	1	0	1	1	0
1	1	1	0	1	0
1	1	1	1	1	0

- (a) Ermitteln Sie die minimale Summe von Primimplikanten für die Logikfunktion für die grüne LED (G) mit Hilfe des Quine-McCluskey Verfahrens.

Phase 1:

L_1			L_2			L_3		
Gruppe	Codewort	✓	Gruppe	Codewort	✓	Gruppe	Codewort	✓
Grp. 0			Grp. 0			Grp. 0		
Grp. 1			Grp. 1			Grp. 1		
Grp. 2			Grp. 2			Grp. 2		
Grp. 3			Grp. 3			Grp. 3		
Grp. 4			Grp. 4			Grp. 4		

Geben Sie die Menge der Primimplikanten in Form von Codeworten an:

{ _____ }

NAME:

Matrikelnummer:

Ersatz für Phase 1 (Ungültige Lösung streichen, sonst sind beide Lösungen ungültig!):

L_1			L_2			L_3		
Gruppe	Codewort	✓	Gruppe	Codewort	✓	Gruppe	Codewort	✓
Grp. 0			Grp. 0			Grp. 0		
Grp.1			Grp.1			Grp.1		
Grp. 2			Grp. 2			Grp. 2		
Grp. 3			Grp. 3			Grp. 3		
Grp. 4			Grp. 4			Grp. 4		

Geben Sie die Menge der Primimplikanten in Form von Codeworten an:

{ _____ }

Phase 2:

Überdeckungstabelle 1:

P	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$

Geben Sie die im ersten Schritt gefundenen essentiellen Primimplikanten an:

{ _____ }

Überdeckungstabelle 2:

P	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$

Geben Sie die minimale SOP-Form als Menge von Primimplikanten in Form von Codeworten an. Bei mehreren Lösungen geben Sie alle an:

{ _____ },
 { _____ },
 { _____ },
 { _____ }.

NAME:

Matrikelnummer:

Ersatz für Phase 2 (Ungültige Lösung streichen, sonst sind beide Lösungen ungültig!):

Überdeckungstabelle 1:

P	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$

Geben Sie die im ersten Schritt gefundenen essentiellen Primimplikanten an:

{ _____ }

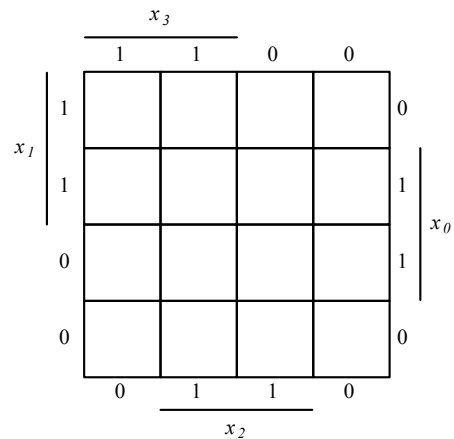
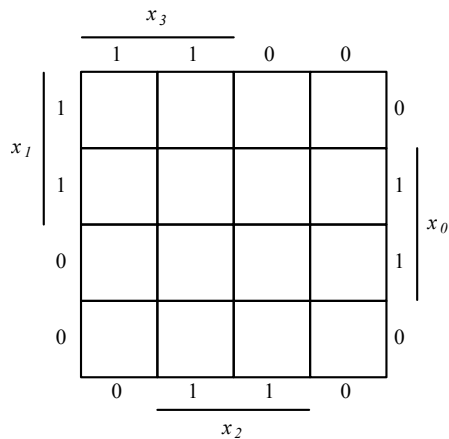
Überdeckungstabelle 2:

P	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$	$m_{\text{---}}$

Geben Sie die minimale SOP-Form als Menge von Primimplikanten in Form von Codeworten an. Bei mehreren Lösungen geben Sie alle an:

{ _____ },
 { _____ },
 { _____ },
 { _____ }.

- (b) Kontrollieren Sie Ihre Lösung(en), indem Sie für die in (a) erstellte Logikfunktion der grünen LED (G) die minimale SOP-Form mit Hilfe des untenstehenden Karnaugh-Diagramms bestimmen:



Ersatzdiagramm, ungültige Lösung streichen!

Minimale SOP-Form für G (Bei mehreren Möglichkeiten geben Sie alle an.):

_____;

_____;

_____;

_____.

- (c) Benutzen Sie die De Morgan'schen Regeln, um eine minimale POS-Form für die rote LED (R) anzugeben. Bei mehreren Lösungen wählen Sie eine Lösung Ihrer Wahl aus:

Aufgabe 4 (Gray Code)**[20 Punkte]**

Mithilfe der in Abbildung 1 dargestellten Drehscheibe ist ein Winkelsensor realisiert worden. Die Scheibe ist in 16 Segmenten mit jeweils 4 Feldern eingeteilt. Die Sensoren $S_0 \dots S_3$ liefern jeweils den Wert 0, wenn das darunter liegende Feld weiß ist und 1, wenn es schwarz ist. Die genauen Rückgabewerte der Sensoren bei der vorgegebenen Codierung der Scheibe sind in Tabelle 1 angegeben.

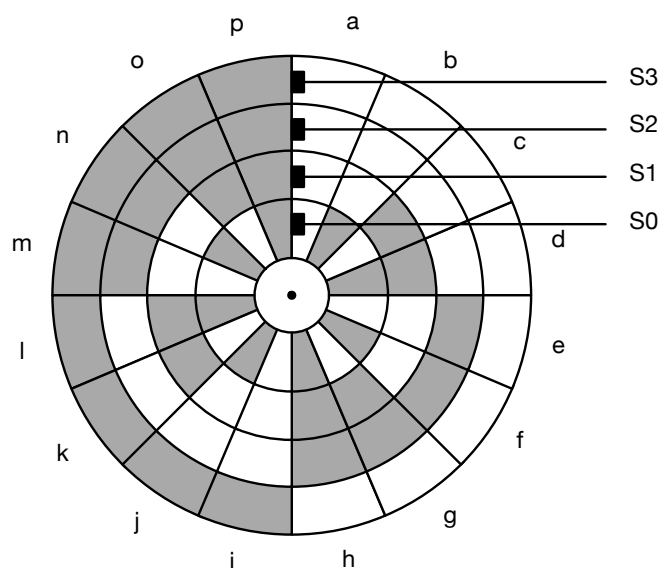


Abbildung 1: Drehscheibe mit Sensoren

Segment	Winkel	S3	S2	S1	S0	Segment	Winkel	S3	S2	S1	S0
a	0° - 22,5°	0	0	0	0	i	180° - 202,5°	1	0	0	0
b	22,5° - 45°	0	0	0	1	j	202,5° - 225°	1	0	0	1
c	45° - 67,5°	0	0	1	0	k	225° - 247,5°	1	0	1	0
d	67,5° - 90°	0	0	1	1	l	247,5° - 270°	1	0	1	1
e	90° - 112,5°	0	1	0	0	m	270° - 292,5°	1	1	0	0
f	112,5° - 135°	0	1	0	1	n	292,5° - 315°	1	1	0	1
g	135° - 157,5°	0	1	1	0	o	315° - 337,5°	1	1	1	0
h	157,5° - 180°	0	1	1	1	p	337,5° - 360°	1	1	1	1

Tabelle 1: Ausgaben der Sensoren bei vorgegebener Codierung der Scheibe

(a) Begründen Sie kurz, warum die in Tabelle 1 gewählte Codierung der Scheibe für die Winkelbestimmung ungünstig ist.

(b) Die Drehscheibe aus Abbildung 1 wird so gedreht, dass die Sektoren d, e, f, g und h hintereinander unter den Sensoren erscheinen. Wo können die in (a) beschriebenen Probleme auftreten? Welche Ausgaben der Sensoren können dabei auftreten?

(c) Entwickeln Sie eine verbesserte Codierung der Scheibe. Verwenden Sie dafür den **reflektierten Gray Code** und tragen Sie die Ausgabewerte der Sensoren S0...S3 bei Verwendung der neuen Codierung in Tabelle 2 ein.

Die rekursive Bildungsvorschrift für n -bit Codes lautet:

$$G_1 = \{0, 1\}; \quad n \geq 1 : G_{n+1} = \{0G_n, 1G_n^{ref}\};$$

Den reflektierten Code erhält man durch Spiegelung der Zeilen in der Mitte

$$G_1 = \begin{Bmatrix} 0 \\ 1 \end{Bmatrix} \rightarrow G_1^{ref} = \begin{Bmatrix} 1 \\ 0 \end{Bmatrix}$$

NAME:

Matrikelnummer:

Segment	Winkel	S3	S2	S1	S0	Segment	Winkel	S3	S2	S1	S0
a	0° - 22,5°					i	180° - 202,5°				
b	22,5° - 45°					j	202,5° - 225°				
c	45° - 67,5°					k	225° - 247,5°				
d	67,5° - 90°					l	247,5° - 270°				
e	90° - 112,5°					m	270° - 292,5°				
f	112,5° - 135°					n	292,5° - 315°				
g	135° - 157,5°					o	315° - 337,5°				
h	157,5° - 180°					p	337,5° - 360°				

Tabelle 2: Lösung zu Aufgabe 4 (c)

Segment	Winkel	S3	S2	S1	S0	Segment	Winkel	S3	S2	S1	S0
a	0° - 22,5°					i	180° - 202,5°				
b	22,5° - 45°					j	202,5° - 225°				
c	45° - 67,5°					k	225° - 247,5°				
d	67,5° - 90°					l	247,5° - 270°				
e	90° - 112,5°					m	270° - 292,5°				
f	112,5° - 135°					n	292,5° - 315°				
g	135° - 157,5°					o	315° - 337,5°				
h	157,5° - 180°					p	337,5° - 360°				

Tabelle 3: Lösung zu Aufgabe 4 (c) - Ungültige Lösung streichen!

- (d) Der Winkelsensor mit neuer Scheibenkodierung wird in einem Gerät eingesetzt. Wenn Winkel in den Intervallen zwischen 95° und 170,5° sowie zwischen 280° und 310° gemessen werden, können Schäden am Gerät entstehen. Deshalb wird ein Alarmsignal **AL** eingeführt, das 1 ist, wenn Winkel in diesen kritischen Intervallen detektiert werden, sonst 0. Bestimmen Sie die minimale SOP Form von **AL** mithilfe des Karnaugh Diagramms aus Abbildung 2.

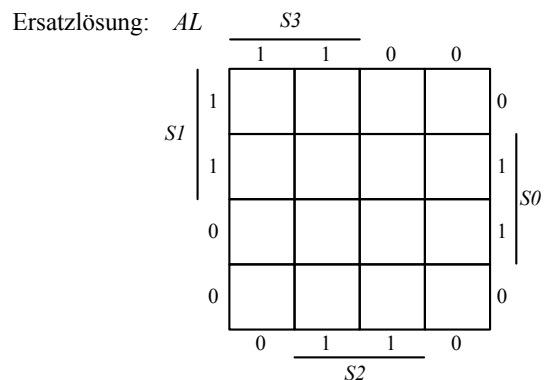
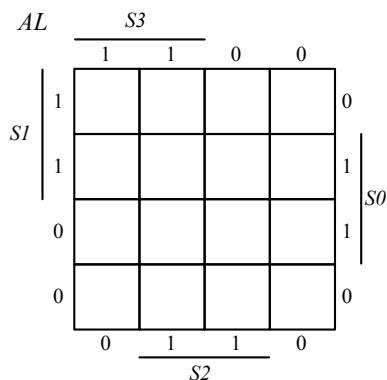


Abbildung 2: Karnaugh-Diagramm zu Aufgabe 4 (d). Ungültige Lösung streichen!

AL = _____

- (e) Entwerfen Sie eine Schaltung für **AL**. Benutzen Sie nur NAND-Gatter mit 2 Eingängen und Inverter. Verwenden Sie dabei die minimale Anzahl an Gattern.

Aufgabe 5 (Mikroprogrammierung)**[25 Punkte]**

Sie sollen für ein Gebäude den Controller für einen simplen Lastenaufzug entwerfen, der den ganzen Tag hoch und runter fährt. Der Controller mit dem Automatengraph aus Abbildung 3 realisiert das, indem er einer Motorsteuerung das Signal H für Hochfahren, oder R für Runterfahren gibt. Die Lastenkabine hat Sensoren, die erkennen, ob der Aufzug oben oder unten angekommen ist. Ein möglicher Automatengraph für den Controller ist in Abbildung 3 dargestellt, wobei in den Zuständen nur Ausgänge notiert sind, die auf 1 gesetzt werden müssen.

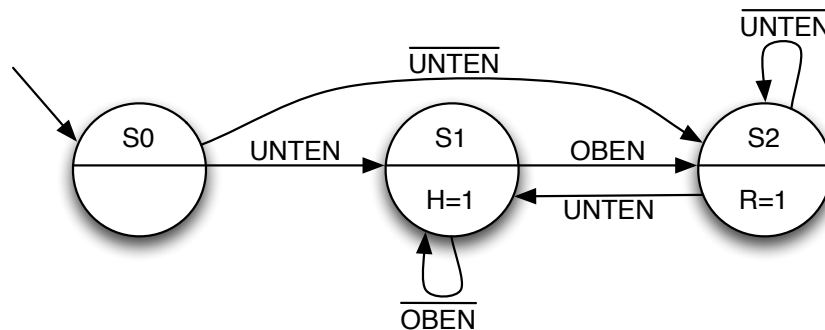


Abbildung 3: Automatengraph eines simplen Lastenaufzugs

- (a) Der Entwurf in Abbildung 3 wurde aus Sicherheitsgründen abgelehnt, da der Kunde den Aufzug für längeres Be- und Entladen anhalten können will. Der Aufzug wird daher um einen „Halten“-Knopf erweitert, der den Aufzug anhalten soll, solange das entsprechende HALTEN-Signal gesetzt ist. Sobald das HALTEN-Signal nicht mehr gesetzt ist, soll in die gleiche Richtung weitergefahren werden wie vor dem Signal. Der Rest des Verhaltens soll sich nicht ändern, der Aufzug soll also weiter in einer Endlosschleife ganz hoch und ganz runter fahren. Modifizieren Sie den Entwurf aus Abbildung 3 entsprechend, indem Sie Abbildung 4 ausfüllen!
- (b) Der Controller soll mit Hilfe eines Mikroprogrammsteuerwerks realisiert werden. Verändern Sie Ihren Automaten aus Teil a) so, dass er als Vorlage für ein Mikroprogramm verwendet werden kann (Splitting von Zuständen), und tragen Sie das Ergebnis in Abbildung 6 ein.
- (c) Entwerfen Sie mit Hilfe des in Abbildung 8 angegebenen Fragmentes das Mikroprogrammsteuerwerk für den Controller. Ergänzen Sie auch die Beschriftung des Steuerwerkes, d.h. geben Sie die Bitbreite der Leitungen an und benennen Sie die Leitungen wie in der Vorlesung definiert.

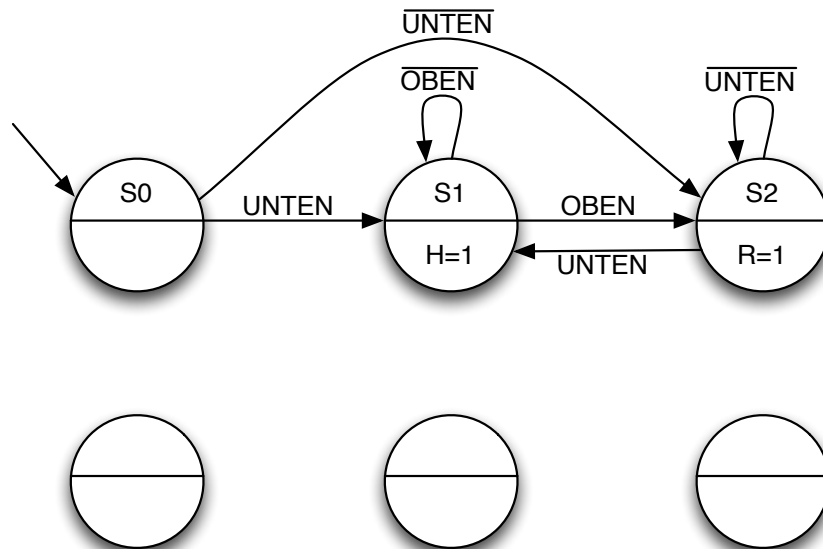


Abbildung 4: Vorlage für a), Automatengraph mit HALTEN-Signalverarbeitung

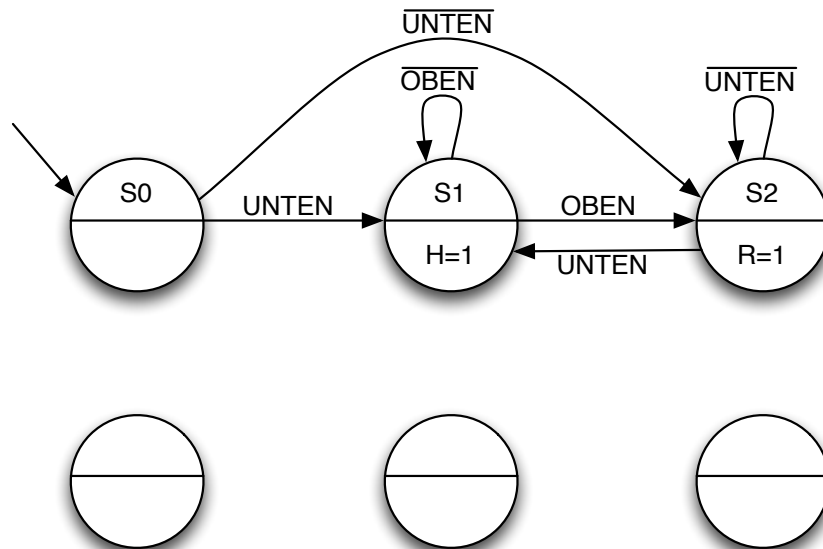


Abbildung 5: Ersatzvorlage für a), **ungültige Lösungen streichen!**

NAME:

Matrikelnummer:

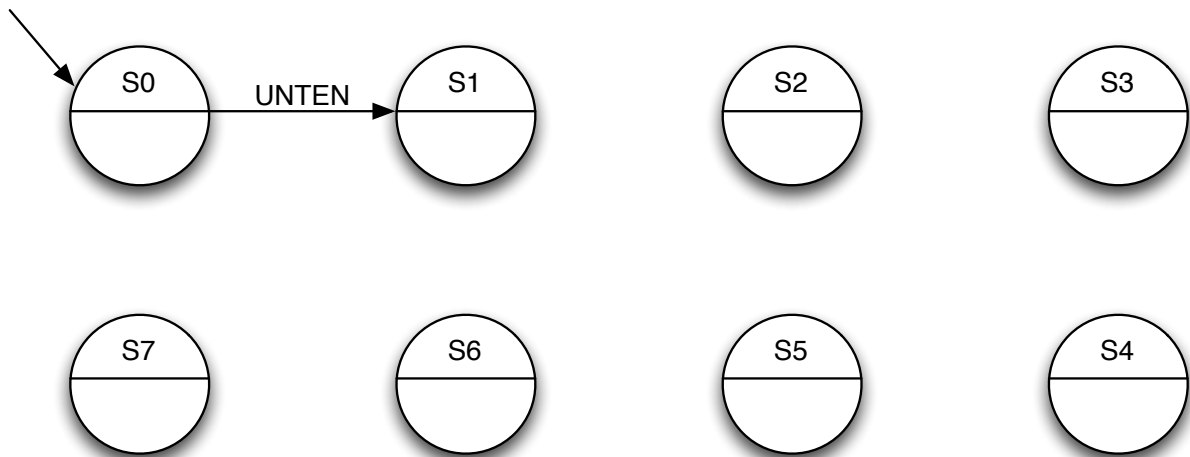


Abbildung 6: Vorlage für b), Automatengraph als Mikroprogrammvorlage

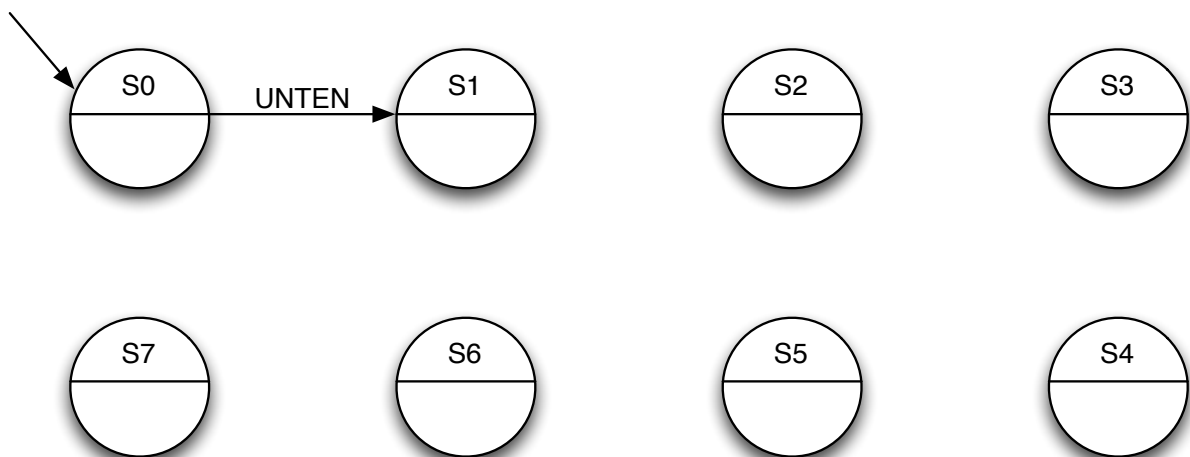


Abbildung 7: Ersatzvorlage für b), **ungültige Lösungen streichen!**

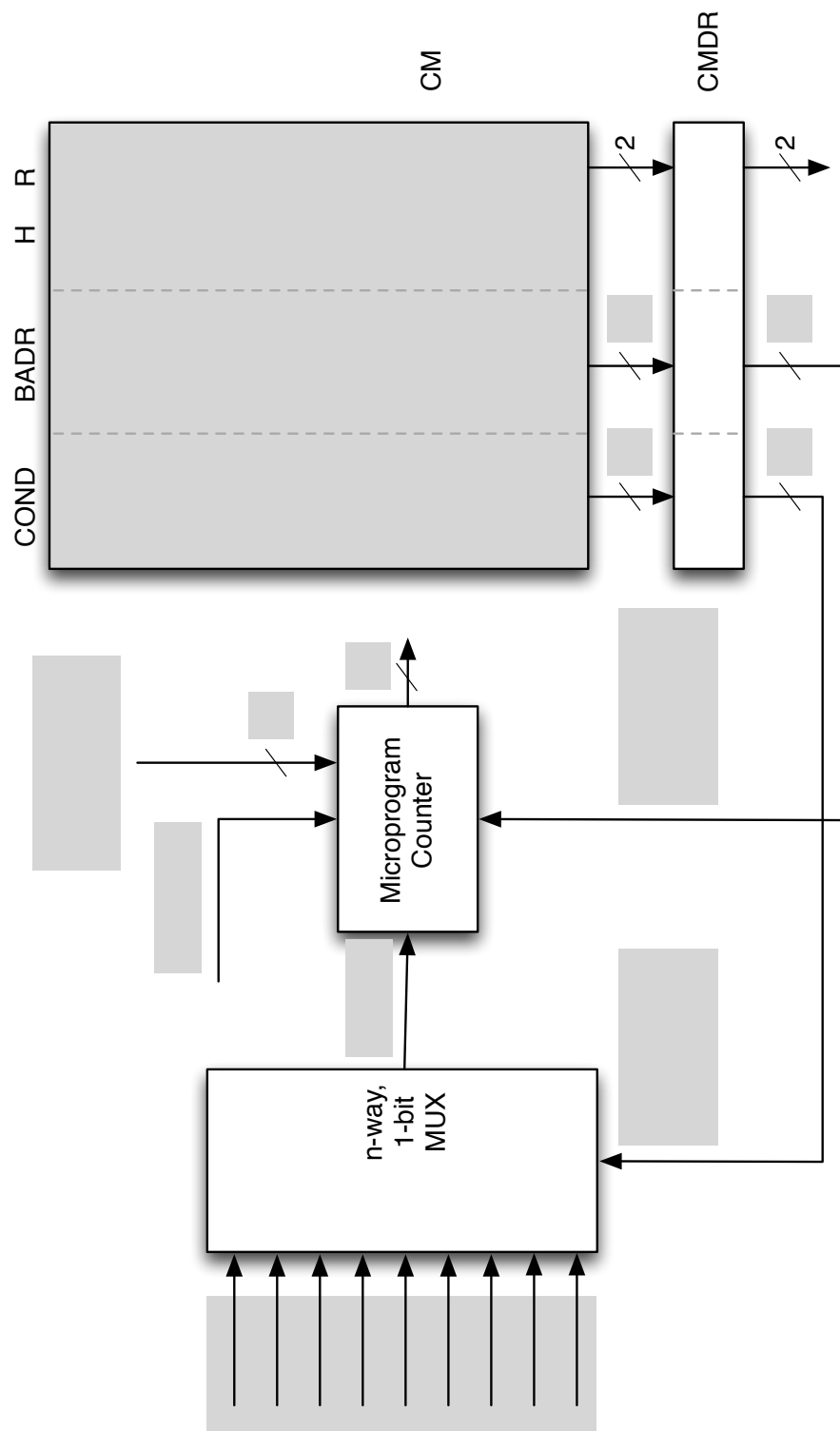


Abbildung 8: Vorlage für c), Mikroprogrammsteuerwerk

NAME:

Matrikelnummer:

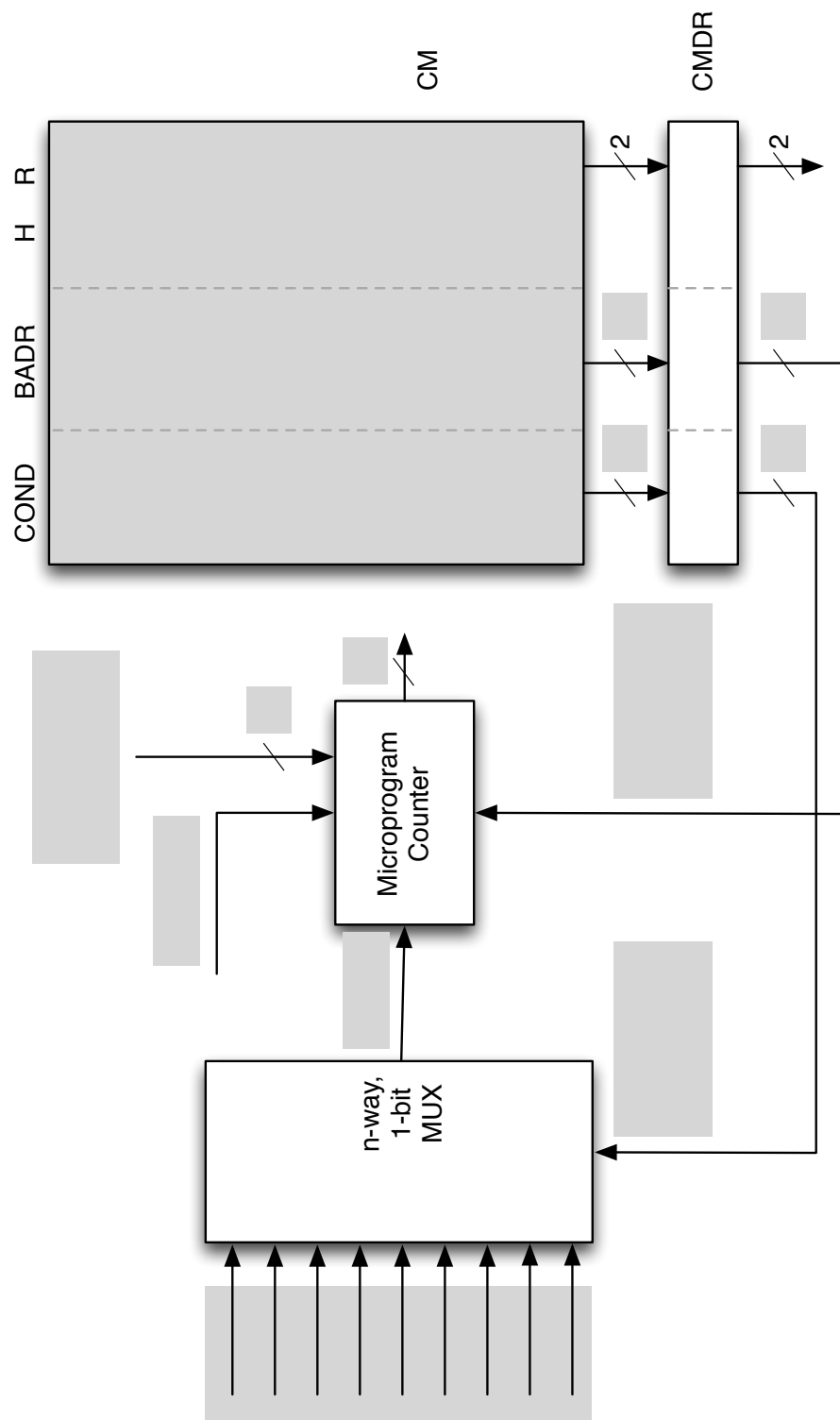


Abbildung 9: Ersatzvorlage für c), **ungültige Lösungen streichen!**

Konzeptpapier: Falls der Platz unter den einzelnen Aufgaben nicht ausreicht, können Sie diese Seiten für Zwischenrechnungen nutzen. Bitte Lösung und Lösungsweg eindeutig mit der Aufgabennummer markieren!