

UNIVERSITÄT KARLSRUHE
Institut für Industrielle Informationstechnik
- Prof. Dr.-Ing. habil. K. Dostert -

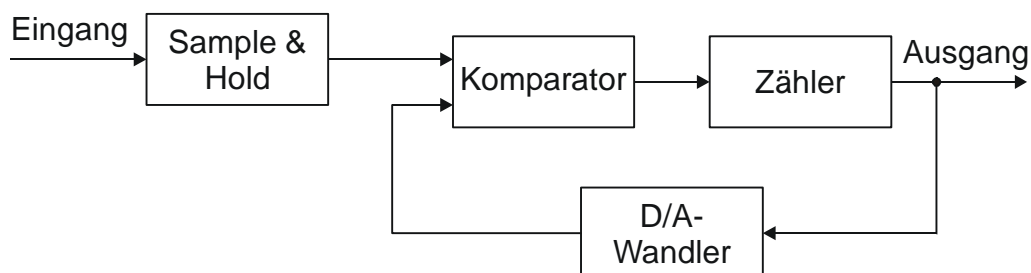
Vordiplomprüfung im Fach

Mikrorechnertechnik

Aufgabe 1: A/D- und D/A-Wandlung

(12 Punkte)

a)



(4)

b)

$$2^N$$

(1)

c)

1. FSM statt Zähler
2. zusätzlich sukzessives Approximationsregister

(2)

d)

$$N$$

(1)

e)

$$I_3 = \frac{U_{ref}}{2R}, \quad I_2 = \frac{U_{ref}}{4R}, \quad I_1 = \frac{U_{ref}}{8R}, \quad I_0 = \frac{U_{ref}}{16R}$$

(2)

f)

$$I = I_3 + I_1 + I_0 = \frac{11}{16} \frac{U_{ref}}{R}$$

(1)

g)

(1)

$$U_A = R \cdot I = \frac{11}{16} U_{ref}$$

Aufgabe 2: Zahlendarstellung in Mikrorechnerprogrammen**(8 Punkte)**

a), b)

(2), (1)

Register	Inhalt (dezimal)	Inhalt (binär)							
		MSB				LSB			
R0	-6	1	1	1	1	1	0	1	0
R1	19	0	0	0	1	0	0	1	1
A	-114	1	0	0	0	1	1	1	0

c)

(1)

$$-6/128 = -0,046875$$

d)

(4)

$$-6 = (-1)^1 \cdot 1,1_2 \cdot 2^{129-127}$$

$$\Rightarrow \text{Vorzeichenbit: } 1$$

$$\text{Exponent: } 129_{10} = 10000001_2$$

$$\text{Mantisse: } 100...0$$

$$19 = (-1)^0 \cdot 1,0011_2 \cdot 2^{131-127}$$

$$\Rightarrow \text{Vorzeichenbit: } 0$$

$$\text{Exponent: } 131_{10} = 10000011_2$$

$$\text{Mantisse: } 001100...0$$

Register	Inhalt (dezimal)	Inhalt (binär)																															
		MSB																LSB															
R0	−6	1	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
R1	19	0	1	0	0	0	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

Aufgabe 3: Verlustleistung von CMOS-Schaltungen**(10 Punkte)**

a)

(2)

Umschaltverluste: Verluste durch Umschaltströme, die während jeder Taktflanke beim Umschalten von Invertern und Gattern entstehen

Umladeverluste: Verluste beim Umladen von Kapazitäten (z. B. Busleitungen)

b)

(2)

$$\bar{I} = N_{inv} \cdot \frac{1}{T} \int_0^T i_d(t) dt = N_{inv} \cdot f \cdot 2 \cdot \int_0^{t_r} i_d(t) dt = N_{inv} \cdot f \cdot 2 \cdot \frac{1}{2} \cdot t_r \cdot I_{DP} = N_{inv} \cdot f \cdot t_r \cdot I_{DP}$$

$$\Rightarrow I_{DP} = \frac{\bar{I}}{N_{inv} \cdot f \cdot t_r} = \frac{40 \text{ mA}}{20000 \cdot 10 \text{ MHz} \cdot 1 \text{ ns}} = 200 \mu\text{A}$$

c) (2)

$$I_{\max} = N_{\text{inv}} \cdot I_{DP} = 20.000 \cdot 200 \mu\text{A} = 4 \text{ A}$$

$$P_{\max} = U \cdot I_{\max} = 3,3 \text{ V} \cdot 4 \text{ A} = 13,2 \text{ W}$$

d) (1)

Kondensator parallel zur Versorgungsspannung

e) (2)

Die Höhe einer Stromspitze ist unabhängig von der Taktfrequenz.

Für den mittleren Strom gilt: $\bar{I} = N_{\text{inv}} \cdot f \cdot I_{DP} \cdot t_r \Rightarrow \bar{I} \sim f$

D. h. der mittlere Strom verdoppelt sich auf 80 mA

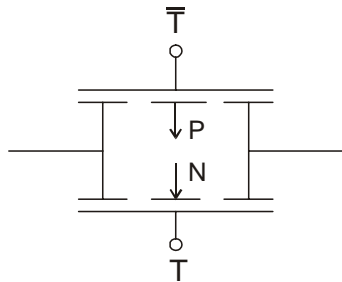
f) (1)

$$P_L = f^* \cdot C^* \cdot U^2 = 10 \text{ MHz} \cdot 6 \text{ nF} \cdot (3,3 \text{ V})^2 = 653,4 \text{ mW}$$

Aufgabe 4: CMOS-Transferrgates

(8 Punkte)

a) (2)

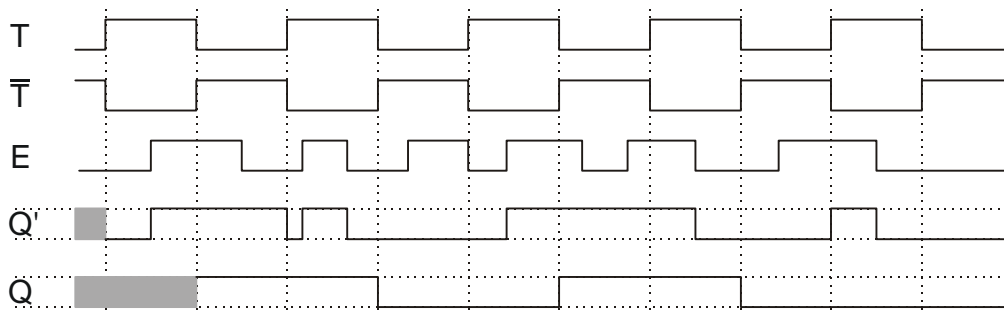


b) (2)

Transferrgate leitet für $T = 1, \bar{T} = 0$

Transferrgate sperrt für $T = 0, \bar{T} = 1$

c) (4)



Aufgabe 5: **Multiplizierer****(10 Punkte)**

a) 25 UND-Gatter (1)

b) 10 NAND-Gatter (1)

c) (4)

-2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
							a_0b_4	a_0b_3	a_0b_2	a_0b_1	a_0b_0
					1	a_1b_4	a_1b_3	a_1b_2	a_1b_1	a_1b_0	
					a_2b_4	a_2b_3	a_2b_2	a_2b_1	a_2b_0		
				a_3b_4	a_3b_3	a_3b_2	a_3b_1	a_3b_0			
			a_4b_4	a_4b_3	a_4b_2	a_4b_1	a_4b_0				
		$\overline{a_5b_4}$	$\overline{a_5b_3}$	$\overline{a_5b_2}$	$\overline{a_5b_1}$	$\overline{a_5b_0}$					
1	a_5b_5	$\overline{a_4b_5}$	$\overline{a_3b_5}$	$\overline{a_2b_5}$	$\overline{a_1b_5}$	$\overline{a_0b_5}$					

d) (4)

Schritt 1

							X	X	X	X	X
					X	X	X	X	X	X	
					X	X	X	X	X		
				X	X	X	X	X			
			X	X	X	X	X				
		X	X	X	X	X					
X	X	X	X	X	X	X					

Schritt 2

						X	X	X	X	X	X
				X	X	X	X	X	X		
				X	X	X	X	X			
	X	X	X	X	X	X					
X	X	X	X	X	X						

Schritt 3

			X	X	X	X	X	X	X	X	X
	X	X	X	X	X	X	X	X			
X	X	X	X	X	X	X					

Abschlussaddition

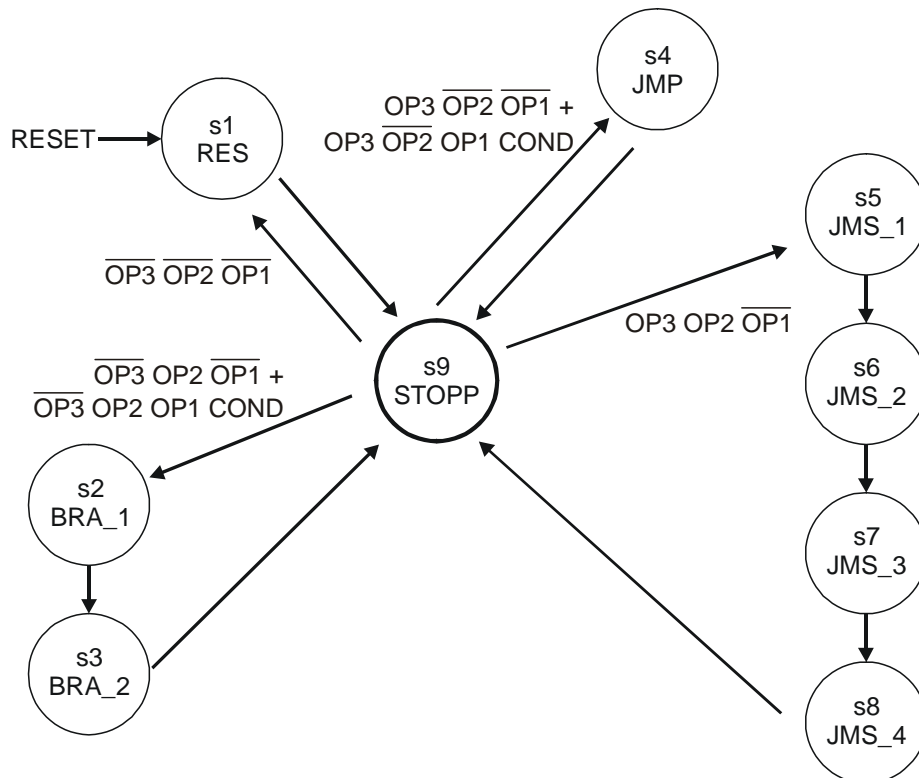
	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X		X	X			

Aufgabe 6: Beschreibung einer FSM

(12 Punkte)

a), b)

(2), (6)



c)

(4)

Befehl	OPC3	OPC2	OPC1
unbedingte Verzweigung	0	1	0
Sprung in Unterprogramm	1	1	0
Reset	0	0	0
bedingter Sprung	1	0	1

Aufgabe 7: A/D-Wandlung mit dem Mikrocontroller ADuC832

(10 Punkte)

a)

(3)

```

Add48:
    mov A,R0
    add A,#48
    mov R0,A
    ret
  
```

b) (4)

```

Divider:
    mov A,R1
    rrc A
    mov R1,A
    mov A,R0
    rr A
    mov ACC.7,C
    mov R0,A
ret

```

c) (3)

```

Senden:
    warten:
    jnb TI,warten
    clr TI
    mov SBUF,A
ret

```

Aufgabe 8: Programmierung der seriellen Schnittstelle im 8051**(10 Punkte)**

a) (3)

	Bit 7				Bit 0			
SCON	0	1	1/X	1	X	X	0	0
TCON	X	1	X	X	X	X	X	X
TMOD	0	0	1	0	X	X	X	X
IE	1	X	X	1	0	X	X	X

b) (3)

Überlaufrate von Timer 1 im Autoreload-Modus: $\frac{f_{osc}}{12} \cdot \frac{1}{256 - TH1}$

Wegen SMOD = 1 werden die Timer 1-Überläufe durch 16 dividiert

$$\Rightarrow \text{Baudrate} = \frac{f_{osc}}{12} \cdot \frac{1}{16} \cdot \frac{1}{256 - TH1}$$

$$\text{aufgelöst nach TH1: } TH1 = 256 - \frac{10 \text{ MHz}}{12 \cdot 16 \cdot 600 \text{ bit/s}} = 169,19 \approx 169$$

c) (2)

$$\text{tatsächliche Baudrate: } \frac{10 \text{ MHz}}{12} \cdot \frac{1}{16} \cdot \frac{1}{256 - 169} = 598,66 \text{ Baud}$$

$$\text{rel. Fehler: } \frac{600 - 598,66}{600} = 2,23 \cdot 10^{-3}$$

d) (2)

Minimale Baudrate: $TH1 = 0$

$$\Rightarrow \text{Baudrate} = \frac{10 \text{ MHz}}{12} \cdot \frac{1}{16} \cdot \frac{1}{256} = 203,45 \text{ Baud}$$

Aufgabe 9: Digitale Signalprozessoren

(10 Punkte)

a) (4)

Register	nach Befehl 1	nach Befehl 2
R1	\$ 0000	\$ 0003
X0	\$ 300000	\$ 300000
R6	\$ 0001	\$ 0001
Y0	\$ 000001	\$ C00000

b) (6)

Inhalte der Register vor MAC-Operation:

$$X0: \$200000 = 0010000...0000_b = \frac{1}{4}$$

$$Y0: \$EC0000 = 11101100...0000_b = -1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{16} + \frac{1}{32} = -\frac{5}{32}$$

$$A: \$0040000000000000 = 0000000001000000...0000_b = \frac{1}{2}$$

MAC-Operation:

$$A + X0 \cdot Y0 = \frac{1}{2} - \frac{1}{4} \cdot \frac{5}{32} = \frac{59}{128} = \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \frac{1}{64} + \frac{1}{128} = 001110110000...0000_b = \$3B0...0$$

Ergebnis:

A2	A1	A0
\$ 00	\$ 3B0000	\$ 000000

Aufgabe 10: Schaltungsbeschreibung mit VHDL

(10 Punkte)

a) (3)

```

ENTITY Automat is
  PORT (clk, res: IN  std_logic;
        ein:      IN  std_logic_vector (1 downto 0);
        aus:      OUT std_logic_vector (1 downto 0)
        );
END Automat;
```

b)

(5)

```
FSM: PROCESS (akt_zust, ein)
BEGIN
  case akt_zust IS
    WHEN S0 =>
      CASE ein IS
        WHEN "00"  => folg_zust <= S0;
        WHEN "01"  => folg_zust <= S3;
        WHEN "10"  => folg_zust <= S1;
        WHEN "11"  => folg_zust <= S2;
        WHEN others => folg_zust <= null;
      END CASE;
    WHEN S1 =>
      CASE ein IS
        WHEN "00"  => folg_zust <= S0;
        WHEN "10"  => folg_zust <= S2;
        WHEN others => folg_zust <= S1;
      END CASE;
    WHEN S2 =>
      CASE ein IS
        WHEN "00"  => folg_zust <= S1;
        WHEN "01"  => folg_zust <= S1;
        WHEN others => folg_zust <= S3;
      END CASE;
    WHEN S3 =>
      CASE ein IS
        WHEN "01"  => folg_zust <= S0;
        WHEN "11"  => folg_zust <= S0;
        WHEN others => folg_zust <= S3;
      END CASE;
    WHEN others => folg_zust <= S0;
  END CASE;
END PROCESS FSM;
```

c)

(2)

```
AUSGABE: PROCESS (akt_zust)
BEGIN
  CASE akt_zust is
    WHEN S0      => aus <= "00";
    WHEN S1      => aus <= "11";
    WHEN S2      => aus <= "01";
    WHEN S3      => aus <= "10";
    WHEN others  => aus <= "00";
  END CASE;
END PROCESS AUSGABE;
```