Bachelor in Electronic Engineering

Hardware Engineering:

Hardware Engineering Lab

Exercise 1

07.04.2022

Prof. Dr.-Ing. Ali Hayek

University of Applied Sciences Hamm-Lippstadt (HSHL)

Tel.: + 49 (2381) 8789 - 924

Room: L4.2-E02-370

E-mail: ali.hayek@hshl.de

1st List of Exercises

- Chapter 2: 2.2.16, 2.2.22, 2.4.11 and 2.4.21
- Chapter 3: 3.1.9, 3.2.16, 3.4.2 and 3.4.5
- Chapter 4: 4.1.20, 4.2.8, 4.3.26, 4.3.43, 4.4.33 and 4.5.8
- Chapter 5: 5.4.1, 5.5.10, 5.5.11, 5.5.12, 5.6.4 and 5.6.5

[1] B. J. Lameres, "Introduction to Logic Circuits & Logic Design with VHDL," Springer Publisher, 2017, 1st Edition

- Additional Exercises

EXERCISE 2.2.16

- Convert 252.987_{10} to hexadecimal. Treat all numbers as unsigned. Use an accuracy of four fractional digits and don’t round up.

SOLUTION 2.2.16

EXERCISE 2.2.22

- Convert $AB.D_{16}$ to octal. Treat all numbers as unsigned.

SOLUTION 2.2.22

EXERCISE 2.4.11

- What is the decimal value of the 8-bit, two's complement code $0111\ 1110_2$?

SOLUTION 2.4.11

EXERCISE 2.4.21

- Compute -99_{10} + -11_{10} using 8-bit two's complement addition. You will need to first convert each decimal number into its 8-bit two's complement code and then perform binary addition (i.e., (9910)+(1110)). Provide the 8-bit result and indicate whether two's complement overflow occurred. Check your work by converting the 8-bit result back to decimal.

SOLUTION 2.4.21

SOLUTION 2.4.21

Additional Exercise

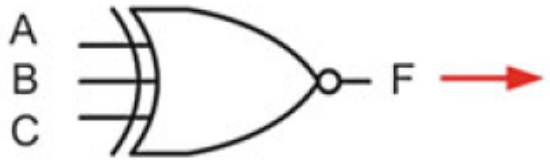
- Compute the following numbers using 8-bit two's complement addition and give the result as a decimal number:

  - 1001 0001 + 0011 0010
  - 0101 1101 + 0111 0010
  - 1010 1001 + 1001 1110
  - 1010 1010 + 0010 1111

EXERCISE 3.1.9

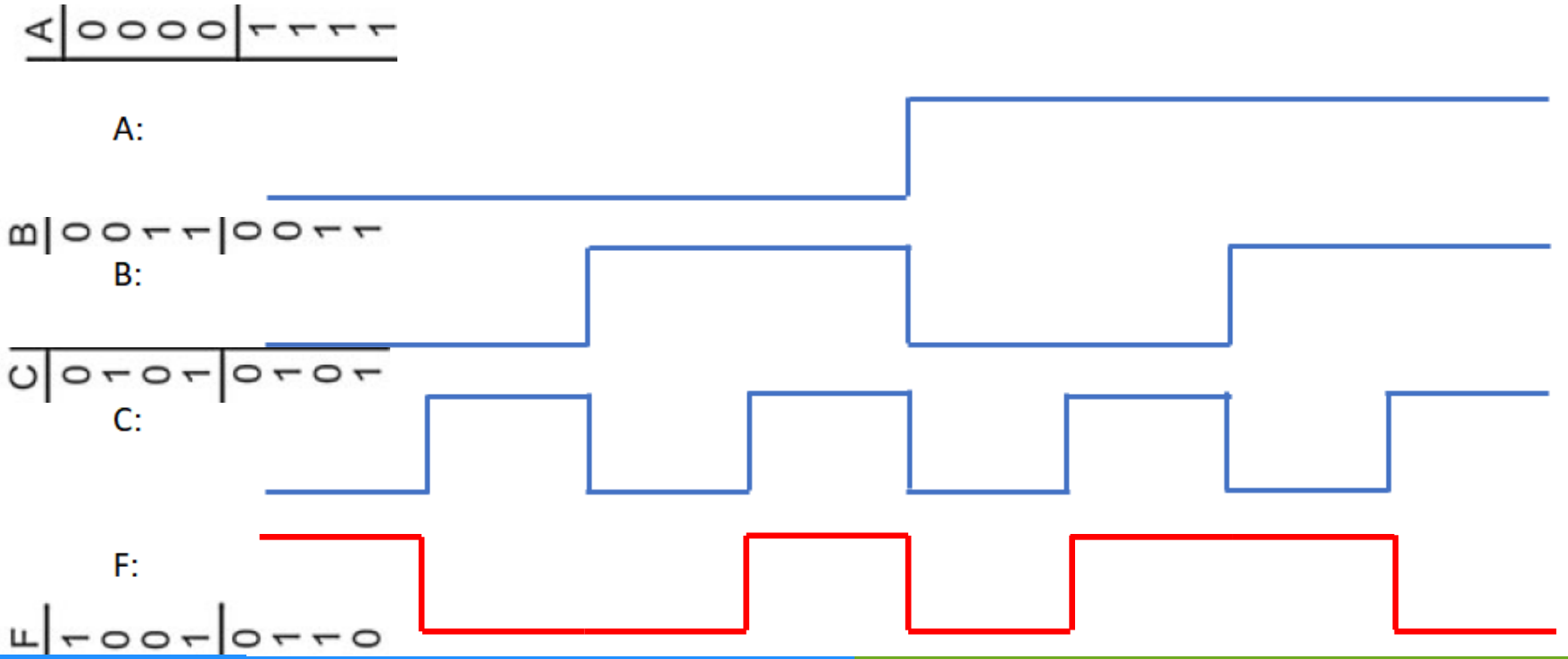
- Give the logic waveform for a 3-input XNOR gate with the input variables A, B, and C and output F.

SOLUTION 3.1.9



row	A	B	C	F
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

C \ AB	00	01	11	10
0	1	0	1	0
1	0	1	0	1



EXERCISE 3.2.16

- Using the data sheet excerpt from Fig. 3.20, give the maximum fall time (tf) for the 74HC04 inverter when powered with VCC = 2 V.

Switching Characteristics

over operating free-air temperature range, C_L = 50 pF (unless otherwise noted) (see Figure 1)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	V_{CC}	T_A = 25°C			SN54HC04		SN74HC04		UNIT
				MIN	TYP	MAX	MIN	MAX	MIN	MAX	
t_{pd}	A	Y	2 V		45	95		125		120	ns
			4.5 V		9	19		29		24	
			6 V		8	16		25		20	
t_t		Y	2 V		38	75		110		95	ns
			4.5 V		8	15		22		19	
			6 V		6	13		19		16	

SOLUTION 3.2.16

ADDITIONL EXERCISE

- Why are pull-down and pull-up resistors used in circuits? Explain your answer using the example of connecting a button to the input of a microcontroller.

EXERCISE 3.4.2

- For the pull-down driver configuration shown in Fig. 3.39, calculate the value of the pull-down resistor (R) in order to ensure that the output current does not exceed 20 mA.

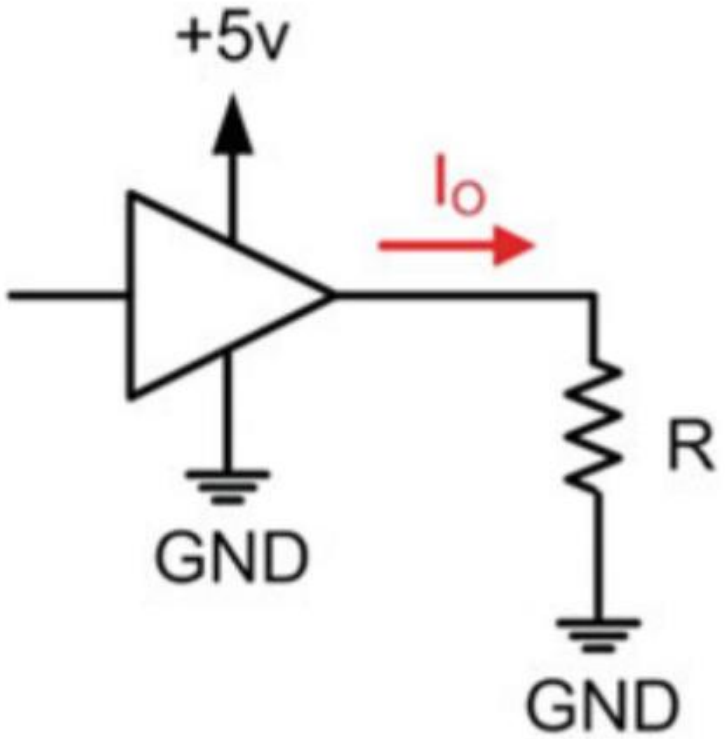


Fig. 3.39

SOLUTION 3.4.2

EXERCISE 3.4.5

- For the LED driver configuration shown in Fig. 3.42 where an output of LOW on the driver will turn on the LED, calculate the value of the resistor (R) in order to set the LED forward current to 5 mA. The LED has a forward voltage of 1.9 V.

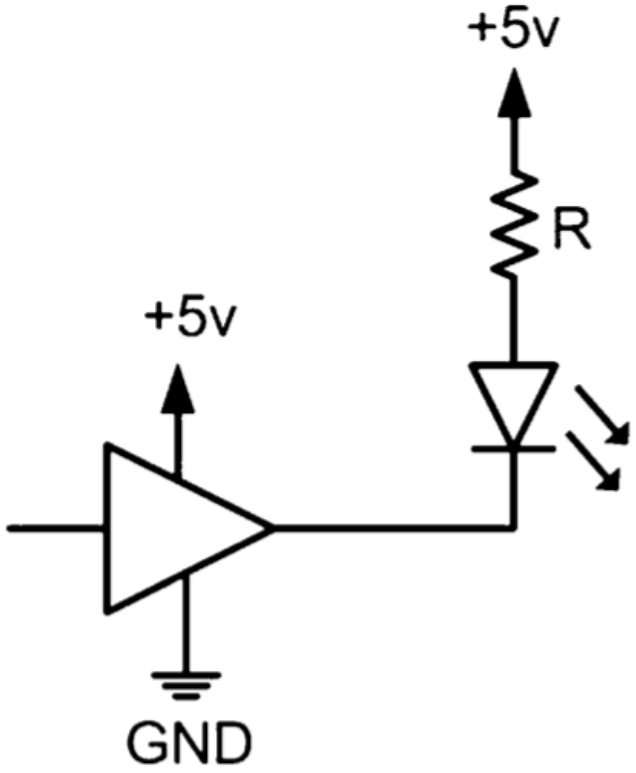


Fig. 3.42
Driver Configuration 7

SOLUTION 3.4.5

EXERCISE 4.1.20

- Use proof by exhaustion to prove that an AND gate with its inputs inverted is equivalent to an OR gate with its outputs inverted.

SOLUTION 4.1.20

EXERCISE 4.2.8

- For the logic diagram given in Fig. 4.29, give the truth table for the output F.

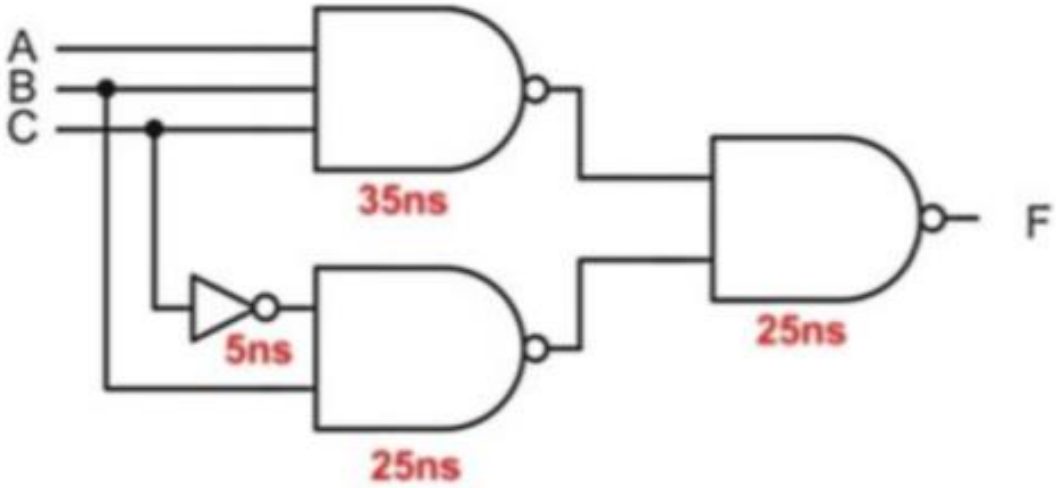


Fig. 4.29
Combinational Logic Analysis 3

$$F = ((A \cdot B \cdot C)' \cdot (B \cdot C')')'$$

SOLUTION 4.2.8

HOCHSCHULE HAMM-LIPPSTADT

F = ((A·B·C)'·(B·C')')'

A	B	C	C'	B·C'	(B·C')'	(A·B·C)'	(A·B·C)'·(B·C')'	F
0	0	0	1	0				
0	0	1	0	0				
0	1	0	1	1				
0	1	1	0	0				
1	0	0	1	0				
1	0	1	0	0				
1	1	0	1	1				
1	1	1	0	0				

EXERCISE 4.3.26

- For the 3-input minterm list in Fig. 4.34, give the canonical sum of products (SOP) logic diagram.

$$F = \sum_{A,B,C}(2,4,6)$$

Fig. 4.34
Combinational Logic Synthesis 5

SOLUTION 4.3.26

$$F = \sum_{A,B,C}(2,4,6)$$

L	A	B	C	F
0	0	0	0	
1	0	0	1	
2	**0**	**1**	**0**	
3	0	1	1	
4	**1**	**0**	**0**	
5	1	0	1	
6	**1**	**1**	**0**	
7	1	1	1	

EXERCISE 4.3.43

- For the 4-input minterm list in Fig. 4.37, give the canonical sum of products (SOP) logic expression.

$$F = \sum_{A,B,C,D}(4,5,7,12,13,15)$$

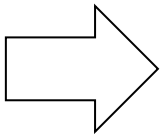
Fig. 4.37
Combinational Logic Synthesis 8

SOLUTION 4.3.43

$$F = \sum_{A,B,C,D}(4,5,7,12,13,15)$$

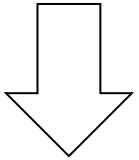
A	B	C	D	F(ABCD)	
0	0	0	0	0	
0	0	0	1	0	
0	0	1	0	0	
0	0	1	1	0	
0	1	0	0	1	A' · B · C' · D'
0	1	0	1	1	A' · B · C' · D
0	1	1	0	0	
0	1	1	1	1	A' · B · C · D
1	0	0	0	0	
1	0	0	1	0	
1	0	1	0	0	
1	0	1	1	0	
1	1	0	0	1	A · B · C' · D'
1	1	0	1	1	A · B · C' · D
1	1	1	0	0	
1	1	1	1	1	A · B · C · D

Optional



		AB			
		00	01	11	10
CD	00	0	1	1	0
	01	0	1	1	0
	11	0	1	1	0
	10	0	0	0	0

K-map



EXERCISE 4.4.33

- For the 4-input truth table and K-map in Fig. 4.54, give the minimal product of sums (POS) logic expression by exploiting “don’t cares.”

A	B	C	D	F
0	0	0	0	1
0	0	0	1	X
0	0	1	0	X
0	0	1	1	X
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	X
1	0	0	1	X
1	0	1	0	0
1	0	1	1	X
1	1	0	0	0
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

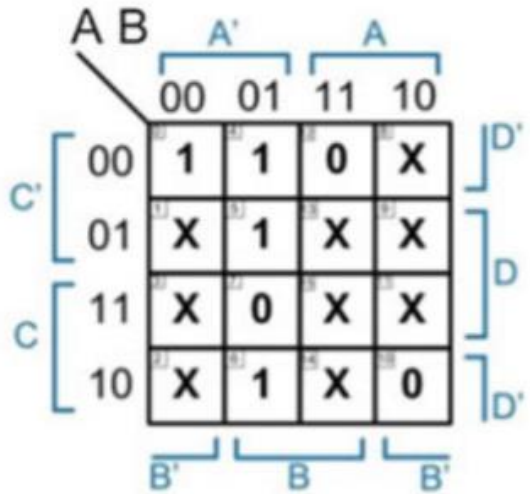


Fig. 4.54
Logic Minimization 16

SOLUTION 4.4.33

- K-map

		AB			
		00	01	11	10
CD	00	1	1	0	X
	01	X	1	X	X
	11	X	0	X	X
	10	X	1	X	0

EXERCISE 4.5.8

- For the 4-input truth table and K-map in Fig. 4.52, give the sum term that helps eliminate static-0 timing hazards in this circuit.

A	B	C	D	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

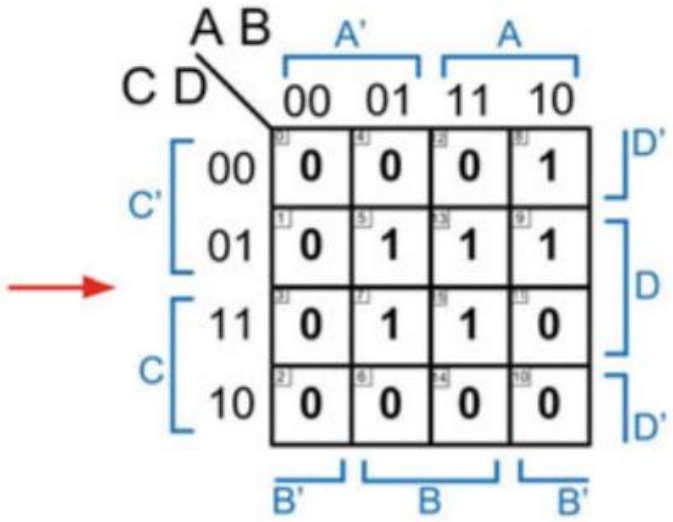
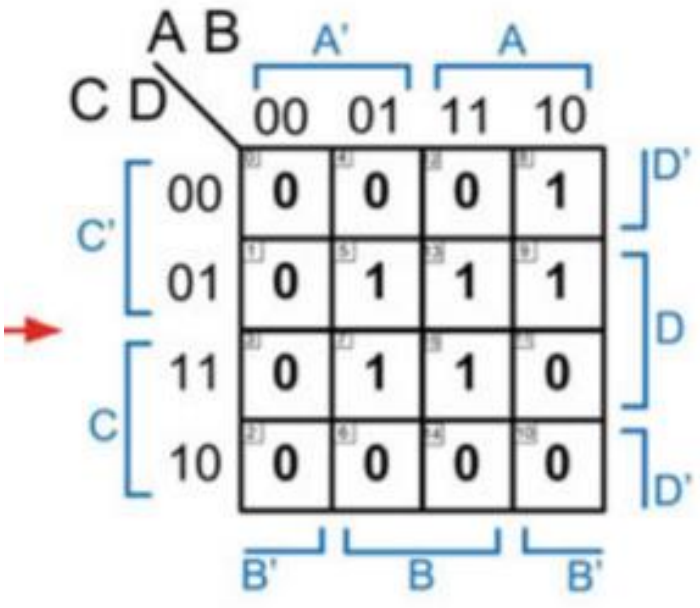


Fig. 4.52
Logic Minimization 14

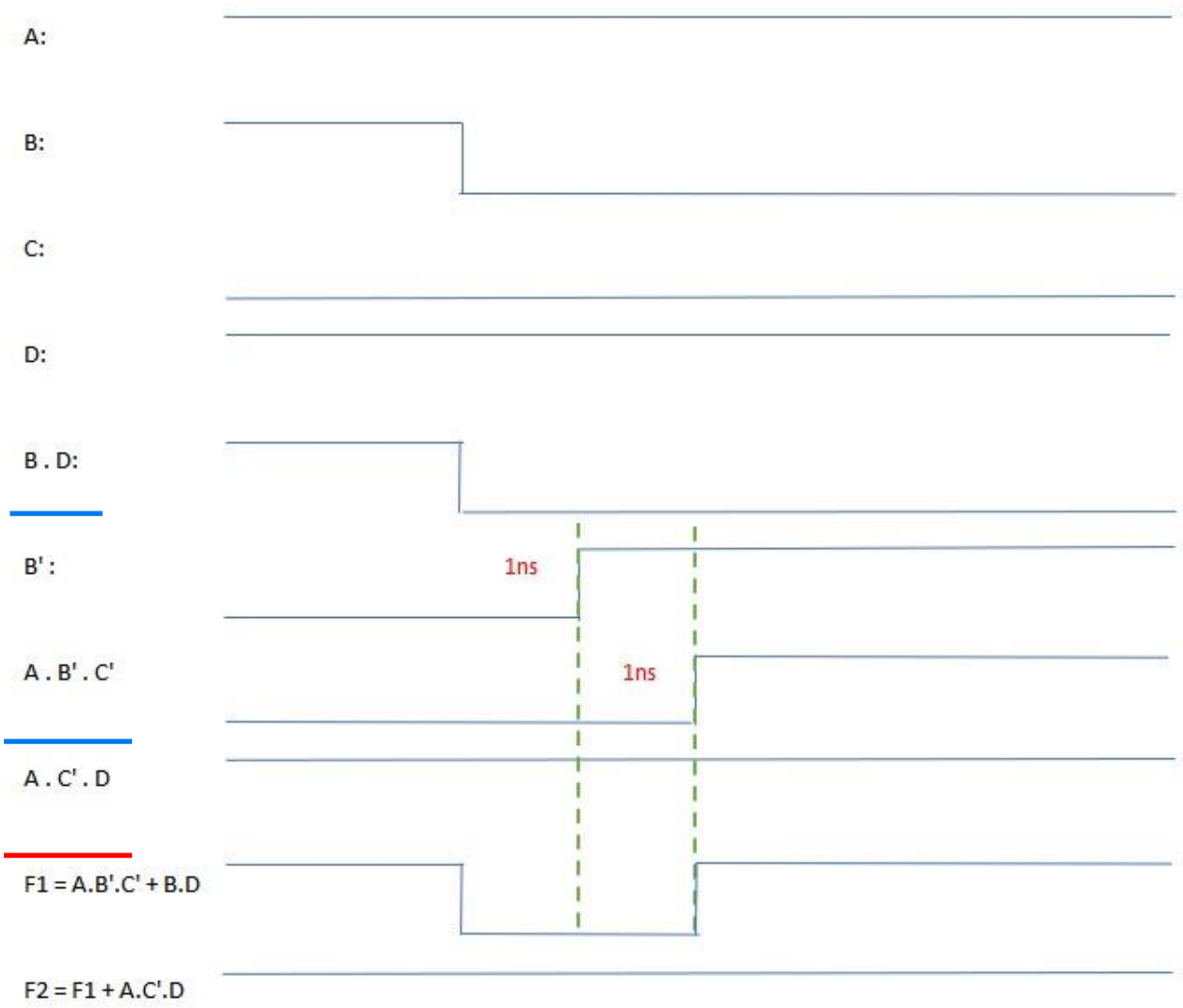
SOLUTION 4.5.8

HOCHSCHULE HAMM-LIPPSTADT

- K-map



SOLUTION 4.5.8



ADDITIONAL EXERCISE

- Simplify the following functions using Boolean algebra axioms und theorems:

- $F = A' \cdot C + B' \cdot C + D' \cdot B \cdot C + A \cdot D \cdot C$

- $Z = A' \cdot B \cdot C + B \cdot C + A \cdot B \cdot C + A \cdot B' \cdot C$

Exercise 5.4.1

- In which construct of VHDL are the inputs and outputs of the system defined?

Solution exercise 5.4.1

Exercise 5.5.10

- Design a VHDL model to implement the behavior described by the 4-input minterm list shown in Fig. 5.14. Use concurrent signal assignments and logical operators. Declare your entity to match the block diagram provided. Use the type bit for your ports.

$$F = \sum_{A,B,C,D}(1,3,9,11)$$

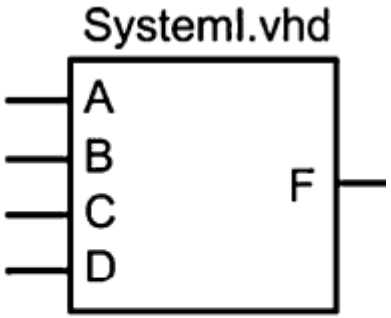
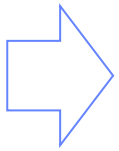


Fig. 5.14
System I Functionality

Solution exercise 5.5.10

- K-Map:

$F = \sum_{A,B,C,D}(1,3,9,11)$



AB CD

- 1: 00 01
- 3: 00 11
- 9: 10 01
- 11: 10 11

		AB			
		00	01	11	10
CD	00	0	0	0	0
	01	1	0	0	1
	11	1	0	0	1
	10	0	0	0	0

Exercise 5.5.11

- Design a VHDL model to implement the behavior described by the 4-input minterm list shown in Fig. 5.14. Use conditional signal assignments. Declare your entity to match the block diagram provided. Use the type bit for your ports.

$$F = \sum_{A,B,C,D}(1,3,9,11)$$

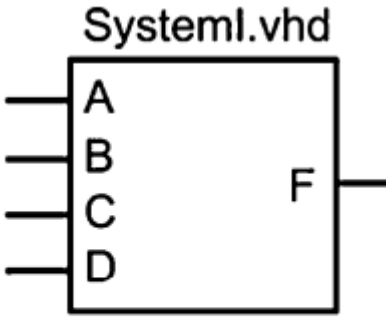


Fig. 5.14
System I Functionality

Solution exercise 5.5.11

Exercise 5.5.12

- Design a VHDL model to implement the behavior described by the 4-input minterm list shown in Fig. 5.14. Use selected signal assignments. Declare your entity to match the block diagram provided. Use the type bit for your ports.

$$F = \sum_{A,B,C,D}(1,3,9,11)$$

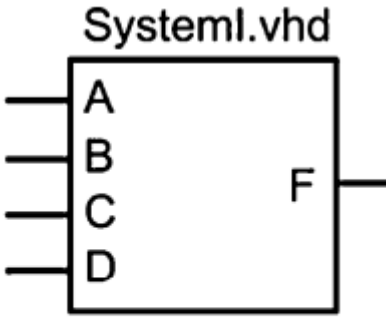


Fig. 5.14
System I Functionality

Solution exercise 5.5.12

Exercise 5.6.4

- Design a VHDL model to implement the behavior described by the 4-input minterm list shown in Fig. 5.14. Use a structural design approach and basic gates. You will need to create whatever basic gates are needed for your design (e.g., INV1, AND2, OR4) and then instantiate them in your upper level architecture to create the desired functionality. The lower level gates can be implemented with concurrent signal assignments and logical operators.

$F = \sum_{A,B,C,D}(1,3,9,11)$

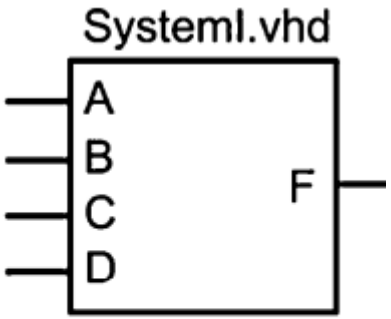


Fig. 5.14
System I Functionality

Solution exercise 5.6.4

Exercise 5.6.5

- Design a VHDL model to implement the behavior described by the 4-input maxterm list shown in Fig. 5.15. Use a structural design approach and basic gates. You will need to create whatever basic gates are needed for your design (e.g., INV1, AND2, OR4) and then instantiate them in your upper level architecture to create he desired functionality. The lower level gates can be implemented with concurrent signal assignments and logical operators (e.g., F <= not A). Declare your entity to match the block diagram provided. Use the type bit for your ports

$$F = \prod_{A,B,C,D}(0,1,2,3,6,8,9,10,11,14)$$

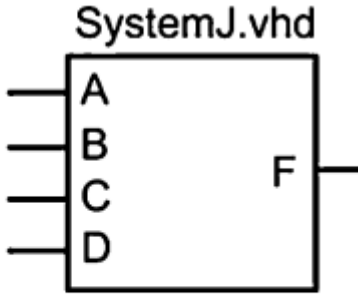


Fig. 5.15
System J Functionality

Solution 5.6.5

Solution 5.6.5

Thank you for your attention.

Prof. Dr.-Ing. Ali Hayek

University of Applied Sciences Hamm-Lippstadt (HSHL)

Tel.: **+ 49 (2381) 8789 - 924**

Room: **L4.2-E02-370**

E-mail: **ali.hayek@hshl.de**