Solution of Machine Structure 1 Examination

Exercise 1

In this exercise, numbers will be represented in 8 bits.

1. Code the decimal numbers in binary: **32** and **122**. (0.5*2pt)

$(32)_{10} = (00100000)_2$ and $(122)_{10} = (01111010)_2$

2. Code the integers **(+122)** and **(-32)** in 1's complement and in 2's complement. (2pt)

Decimal	1's Complement	2' Complement
+122	01111010	01111010
-32	11011111	11100000

3. Calculate, in binary, the addition of the two decimal numbers **122** and **(- 32)**. (1pt)

```
   01111010
 - 00100000
 -----------
   01011010
```

4. Can the result of adding the two numbers **(-122)** and **(-32)** be represented in sign and absolute value using 8 bits? If so, why? If not, what are the limitations ? (1pt)

No, we cannot represent -122 + -32 in binary using 8 bits because the result of the addition in decimal will be -154, and we cannot represent -154 on 8 bits according to the sign and absolute value representation. It requires 9 bits to represent it.

Exercise 2

The coding of a real number in floating point is done according to the IEEE 754 -32 standard:

$$(-1)^S \times (1,M_n) \times 2^E$$

- S: the sign bit.

- E: the exponent represented in 8-bit (coded with excess 127).

- M_n: the mantissa normalized to 23 bits.

1. What are the smallest and largest decimal values for the exponent ? (0.5*2pt)

The exponent is encoded with excess 127, meaning it is represented by the binary equivalent C such that: C = E + 127.

Let E_L and E_S be the largest and smallest value of the exponent.

As the representation of C is done in binary in 8 bits, therefore: $C_L = 2^8-1 = 255$; $C_S = 0$

Hence $C_L = E_L + 127 \leftrightarrow \mathbf{E_L = 255 - 127 = 128}$

$C_S = E_S + 127 \leftrightarrow \mathbf{E_S = 0 - 127 = -127}$

2. Code the following real numbers according to the IEEE standard 754-32: **(-17,75)** and **(+21,05)** (2pt)

- $(-17,75)_{10} = (10001,11)_2 = (-1)^{\mathbf{1}} \times (1,\mathbf{000111}) \times 2^{\mathbf{4}}$ <=> E=4 => C= 4+127 = 131= $(10000011)_2$

  $(-17,75)_{10} = (1\ \ 10000011\ \ 00011100000000000000000\ \ )_{\text{IEEE 754-32}}$

- $(+21.05)_{10} = (10101.00000101)_2 = (-1)^{\mathbf{0}} \times (1,\mathbf{010100000101}) \times 2^{\mathbf{4}}$ <=> E=4 =>C= 4+127 = 131= $(10000011)_2$

  $(+21.05)_{10} = (0\ \ 10000011\ \ 01010000010100000000000\ \ )_{\text{IEEE 754-32}}$

3. Convert to decimal the binary number (1 10000100 10010100000000000000000) representing a sequence of bits coded according to the IEEE 754-32 standard. (2pt)

$(1\ \ 10000100\ \ 10010100000000000000000)_{\text{IEEE 754-32}}$ ⇔ S= 1, $C=(10000100)_2= 132$=> E=132-127= 5 **0,5 pt**

$M_n = 100101 \Leftrightarrow (-1)^{\mathbf{1}} \times (1,100101\ ) \times 2^{\mathbf{5}}$ **0.5pt** => $(110010.1)_2 = \mathbf{(-50,5)_{10}}$ **1 pt**

Exercise 3 (6pt)

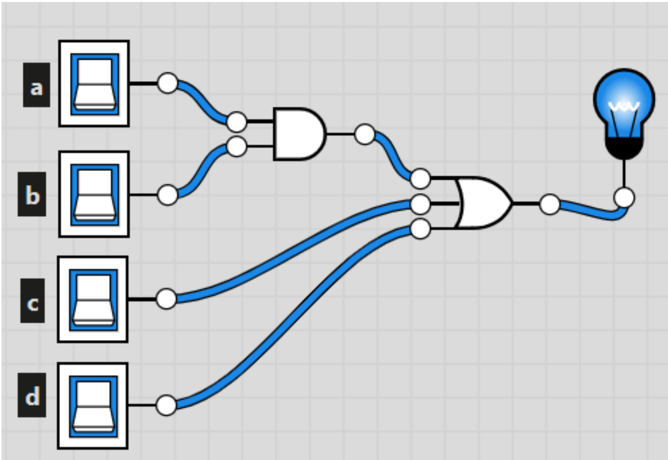
1. Simplify algebraically the following equation E = (a+c+d) (b+c+d) and give its circuit with minimum of gates. (2pt)

Method 1: **1pt**

E = (a+(c+d)) (b+(c+d)) = ab + a (c+d) + b (c+d) + (c+d) (c+d) = ab + a (c+d) + b (c+d) + (c+d) = ab + (c+d) (a + b + 1) = ab + c + d

Method 2:

E = (a + c + d) (b + c + d) = ab + ac + ad + bc +**cc**+ cd + bd + cd + **dd** = ab +ac + **ad** + bc + c + cd +**bd** + **d** = ab + (1+a+b+d) c + **(1+a+b) d** = ab + c + d



Logigram representing the function E. **1pt**

2. Express a ⊕ b in the **second canonical** form. (**0,5*2 pt**)

a	b	a ⊕ b	maxtermes
0	0	0	a + b
0	1	1	
1	0	1	
1	1	0	$\bar{a} + \bar{b}$

$$a \oplus b = (a + b)(\bar{a} + \bar{b})$$

3. When does (a ⊕ b ⊕ c) + (a ⊕ c) = 0? (**1pt**)

Method 1:

a	b	c	a ⊕ b	a ⊕ b ⊕ c	a ⊕ c	(a ⊕ b ⊕ c) + (a ⊕ c)
0	**0**	**0**	0	0	**0**	**0**
0	0	1	0	1	1	1
0	1	0	1	1	0	1
0	1	1	1	0	1	1
1	0	0	1	1	1	1
1	**0**	**1**	1	0	**0**	**0**
1	1	0	0	0	1	1
1	1	1	0	1	0	1

Method 2:

$$(a \oplus b \oplus c) + (a \oplus c) = 0 \Leftrightarrow a \oplus b \oplus c = 0 \; et \; a \oplus c = 0 \Leftrightarrow (a \oplus c) \oplus b = 0 \; et \; a = c \Leftrightarrow 0 \oplus b = 0 \; et \; a = c \Leftrightarrow b = 0 \; et \; a = c$$

4. Provide the simplified function while clearly showing the grouping in the Karnaugh table:(2pt)

ab \ cd	00	10	11	01
00	0	0	1	0
10	0	1	1	0
11	0	1	1	0
01	0	1	0	1

F_3 (a,b,c,d) = $ac+bc\bar{d}+\bar{b}cd+\bar{a}b\bar{c}d$

ab \ cd	00	10	11	01
00	1	1	1	0
10	1	0	1	0
11	1	0	1	0
01	1	0	1	1

F_4 (a,b,c,d) = $\bar{c}\bar{d}+cd+\bar{a}\bar{b}c+\bar{a}bd$

ab \ cd	00	10	11	01
00	0	0	0	0
10	1	1	1	1
11	0	1	1	0
01	0	1	1	0

F_1 (a,b,c,d) = $a\bar{b}+bc$

ab \ cd	00	10	11	01
00	1	0	1	1
10	0	0	0	0
11	0	0	0	0
01	1	1	0	1

F_2 (a,b,c,d) = $\bar{a}\bar{c}+\bar{a}\bar{b}d+\bar{a}b\bar{d}$

Exercice 4 (4pts)

Create a logic circuit to check whether a four-digit (a,b,c and d) binary number is prime .

1° Truth table: **(1pt)**

a	b	c	d	**F**	TypeTerm	Term
0	0	0	0	**0**	Max	$a+b+c+d$
0	0	0	1	**1**	Min	$\bar{a}\bar{b}\bar{c}d$
0	0	1	0	**1**	Min	$\bar{a}\bar{b}c\bar{d}$
0	0	1	1	**1**	Min	$\bar{a}\bar{b}cd$
0	1	0	0	**0**	Max	$a+\bar{b}+c+d$
0	1	0	1	**1**	Min	$\bar{a}b\bar{c}d$
0	1	1	0	**0**	Max	$a+\bar{b}+\bar{c}+d$
0	1	1	1	**1**	Min	$\bar{a}bcd$
1	0	0	0	**0**	Max	$\bar{a}+b+c+d$
1	0	0	1	**0**	Max	$\bar{a}+b+c+\bar{d}$

2°: Canonical Forms: (0.5*2pt)

1st Canonical From: Sum of MinTerm:

$F(a,b,c,d)= \bar{a}\bar{b}\bar{c}d+\bar{a}\bar{b}c\bar{d}+\bar{a}\bar{b}cd+\bar{a}b\bar{c}d+\bar{a}bcd+a\bar{b}cd+ab\bar{c}d$

2nd canonical Form: Product of MaxTerm:

$F(a,b,c,d)=(a+b+c+d)(a+\bar{b}+c+d)(a+\bar{b}+\bar{c}+d)(\bar{a}+b+c+d)(\bar{a}+b+c+\bar{d})(\bar{a}+b+\bar{c}+d)(\bar{a}+\bar{b}+c+d)(\bar{a}+\bar{b}+\bar{c}+d)(\bar{a}+\bar{b}+\bar{c}+\bar{d})$

1	0	1	0	**0**	Max	$\bar{a}+b+\bar{c}+d$
1	0	1	1	**1**	Min	$a\bar{b}cd$
1	1	0	0	**0**	Max	$\bar{a}+\bar{b}+c+d$
1	1	0	1	**1**	Min	$ab\bar{c}d$
1	1	1	0	**0**	Max	$\bar{a}+\bar{b}+\bar{c}+d$
1	1	1	1	**0**	Max	$\bar{a}+\bar{b}+\bar{c}+\bar{d}$

3° Simplification (0.5*2 pt)

ab \ cd	**00**	**10**	**11**	**01**
00	0	1	1	1
10	0	0	1	0
11	0	0	0	1
01	0	0	1	1

F(a,b,c,d)= $\bar{a}d+\bar{a}\bar{b}c+b\bar{c}d+\bar{b}cd$

4° Logigram (logical circuit) 1pt