

# LSP Exam 2022–01–19 / Zkouska LSP 2022–01–19 / LSP 考试 2022 年 1 月 19 日

**Course:** B0B35LSP – Logické systémy a procesory | BE5B35LSP – Logic Systems and Processors **University:** CVUT FEL (ČVUT) – České vysoké učení technické v Praze | Czech Technical University in Prague **Keywords:** LSP, Exam, Zkouska, 2022–01–19, truth table, De Morgan, RS latch, VHDL

[CN Version](#) | [EN Version](#) | [CZ Version](#)

---

## LSP 考试 2022 年 1 月 19 日

AI 推演版本 – 本试卷 PDF 无官方答案, 以下为推演解析

### 考试信息

- 日期: 2022 年 1 月 19 日
  - 语言: 捷克语
  - 总分: 50 分 (第一部分 25 分  $\geq 9$  + 第二部分 25 分  $\geq 9$ )
- 

### 第 1 题 – 真值表 (7 分)

题目: 根据电路图写出函数 Y 的真值表 [English] Write the truth table for function Y based on the circuit diagram

提示: 可以使用 Shannon 展开简化计算

卡诺图格式:

	B	
	A	00 01 11 10
DC	00	
	01	
	11	
	10	

### 第 2 题 – De Morgan 定律 (3 分)

题目: 重写逻辑表达式 F, 使“not”运算符仅出现在变量前面 [English] Rewrite the logic expression F so that “not” operators appear only before variables

原式：

$$F = \text{not} ((\text{not } A \text{ and not } C) \text{ xor } (A \text{ and } B \text{ and } C))$$

解答思路：1. 展开 xor:  $A \text{ xor } B = (A \text{ and not } B) \text{ or } (\text{not } A \text{ and } B)$  2. 应用 De Morgan:  $\text{not}(A \text{ or } B) = \text{not } A \text{ and not } B$

答案：

$$F = ((A \text{ or } C) \text{ and } (\text{not } A \text{ or not } B \text{ or not } C)) \text{ xor } ((\text{not } A \text{ and not } C) \text{ or } (A \text{ and } B \text{ and } C))$$

-- 简化后：

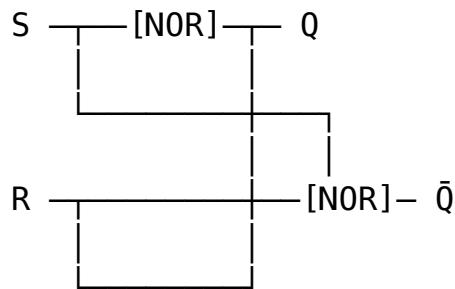
$$F = (A \text{ and not } B) \text{ or } (\text{not } A \text{ and not } C) \text{ or } (B \text{ and } C)$$

---

### 第 3 题 – RS 锁存器设计 (10 分)

题目：分别用 NOR 门和 NAND 门画出 RS 锁存器，并完成真值表 [English] Draw RS latch using NOR gates and NAND gates respectively, and complete the truth tables

#### NOR 门 RS 锁存器



真值表 (NOR): | S | R | Q | Q̄ | | | | | 0 | 0 | Q(保持) | Q̄(保持) | | 0 | 1 | 0 | 1 | | 1 | 0 | 1 | 0 |  
| 1 | 1 | 0 | 0 | (禁止) |

#### NAND 门 RS 锁存器

真值表 (NAND): 低电平有效 | S | R | Q | Q̄ | | | | | 0 | 0 | 1 | 1 | (禁止) | | 0 | 1 | 1 | 0 | | 1 | 0 |  
0 | 1 | | 1 | 1 | Q(保持) | Q̄(保持) |

---

### 第 4 题 – Moore/Mealy 自动机定义 (5 分)

题目：完成 Moore/Mealy 自动机定义 [English] Complete the Moore/Mealy automaton definition

$$M = < X, S, Z, \omega, \delta, s_0 \in S >$$

答案： –  $X$ : 有限输入符号集合 / Finite input alphabet –  $S$ : 有限状态集合 / Finite set of states  
–  $Z$ : 有限输出符号集合 / Finite output alphabet –  $\delta$ : 状态转移函数 / State transition function (Moore:  $S \times X \rightarrow S$ , Mealy:  $S \times X \rightarrow S$ ) –  $\omega$ : 输出函数 / Output function (Moore:  $S \rightarrow Z$ , Mealy:  $S \times X \rightarrow Z$ ) –  $s_0$ : 初始状态 / Initial state

---

## 第 5 题 – VHDL 代码分析 (10 分)

题目：分析压缩的 VHDL 代码，画出电路图 [English] Analyze the compressed VHDL code and draw the circuit diagram

```
library IEEE; use IEEE.STD_LOGIC_1164.all;
entity zzz is port (a : in std_logic; data : in std_logic; q, qn : out std_
architecture rtl of zzz is
    signal qv : std_logic;
begin
    process(a)
    begin
        if rising_edge(a) then
            qv <= not data;
            q <= qv;
        end if;
        qn <= not qv;
    end process;
end;
```

功能分析： –  $a$  的上升沿触发 –  $qv$  存储  $data$  的反值 –  $q$  输出前一个  $qv$  值（延迟一拍） –  $qn$  是  $qv$  的反值 – 这是一个带反相的移位寄存器

---

## 第 6 题 – 全加器设计 (3 分)

题目：画出带  $A$ 、 $B$ 、 $Cin$  输入和  $Y$ 、 $Cout$  输出的完整一位全加器电路 [English] Draw a complete one-bit full adder circuit with  $A$ ,  $B$ ,  $Cin$  inputs and  $Y$ ,  $Cout$  outputs

逻辑方程：

$$\begin{aligned} \text{Sum} &= A \oplus B \oplus \text{Cin} \\ \text{Cout} &= (A \wedge B) \vee (\text{Cin} \wedge (A \oplus B)) \end{aligned}$$

---

## 第 7 题 – VHDL 电路描述 (7 分)

题目：用最短的 VHDL 代码描述给定电路图，使用向量操作 [English] Describe the given circuit diagram with shortest VHDL code using vector operations

```

library ieee; use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
entity Zahadum is port(
    -- 根据电路图声明端口
);
end entity;
architecture rtl of Zahadum is
begin
    -- 使用向量操作简化代码
end architecture rtl;

```

---

### 第 8 题 – 并发 VHDL 描述 (5 分)

**题目:** 分析第 1 题电路的功能, 用简化的并发语句描述 (不是直接翻译逻辑方程) [English] Analyze the function of circuit from Q1, describe with simplified concurrent statements (not direct translation of logic equations)

```

library ieee; use ieee.std_logic_1164.all; use ieee.numeric_std.all;
entity yyy is port(a,b,c,d: in std_logic; y: out std_logic); end entity;
architecture dataflow of yyy is
begin
    -- 简化的并发语句
end architecture;

```

---

### 第 9 题 – 附加题: SOS 信标编码器 (10 分)

**题目:** 使用高级 VHDL 命令编写发送 SOS 信号的编码器 [English] Write an SOS signal encoder using advanced VHDL commands 信号序列: 01010100011011101110001010100

```

library ieee; use ieee.std_logic_1164.all; use ieee.numeric_std.all;
entity SOS is
    port (clk : in std_logic;
          X: in std_logic_vector(4 downto 0);
          Morse, STOP: out std_logic);
end entity;
architecture rtl of SOS is
    constant SOS_PATTERN : std_logic_vector(29 downto 0) :=
        "01010100011011101110001010100";
begin
    process(X)
        variable idx : integer;
    begin
        idx := to_integer(unsigned(X));
        if idx < 30 then

```

```
Morse <= SOS_PATTERN(29 - idx);  
STOP <= '0';  
else  
    Morse <= '0';  
    STOP <= '1';  
end if;  
end process;  
end architecture;
```

---

## 知识点总结

### 本次考试重点

1. 真值表构建
2. De Morgan 定律应用
3. RS 锁存器 (NOR 和 NAND 实现)
4. Moore/Mealy 自动机定义
5. VHDL 代码分析 (移位寄存器)
6. 全加器设计
7. 向量操作 VHDL
8. 电路简化分析
9. 莫尔斯码编码器