

# LSP Exam 2022-01-24 / Zkouska LSP 2022-01-24 / LSP 考试 2022 年 1 月 24 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, 2022-01-24, Shannon expansion, RS latch, VHDL

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

---

## LSP 考试 2022 年 1 月 24 日 (英文版)

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 24 日
  - 语言: 捷克语/英语双语
  - 总分: 50 分 (第一部分 25 分 >8 + 第二部分 25 分 >8)
- 

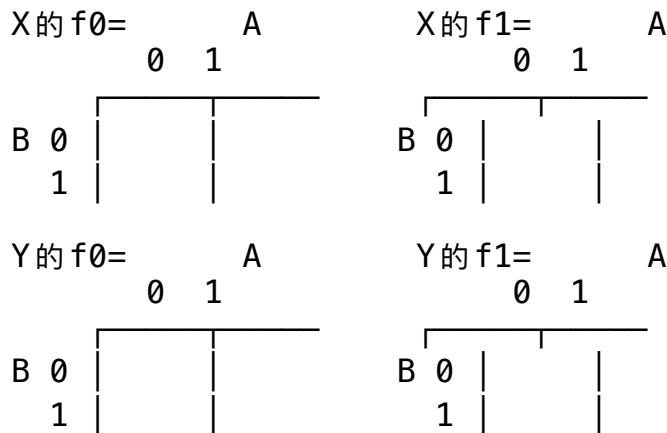
### 第 1 题 – Shannon 展开 (6 分)

题目: 将电路的  $X, Y = f(A, B, C)$  函数用 Shannon 展开分解为: [English] Decompose the circuit functions  $X, Y = f(A, B, C)$  using Shannon expansion into:

$$X, Y = (\bar{C} \wedge f_0(A, B)) \vee (C \wedge f_1(A, B))$$

将  $f_0$  和  $f_1$  写成卡诺图

分别为 X 和 Y 各画两个卡诺图:



## 第 2 题 – 有符号/无符号运算 (4 分)

题目: 在 8 位加法器上执行  $96+97+98+99$ , 结果作为 8 位数的十进制值是多少? [English] Perform  $96+97+98+99$  on an 8-bit adder, what is the decimal value of the result as an 8-bit number?

计算:  $- 96+97+98+99 = 390 - 390 \bmod 256 = 134$

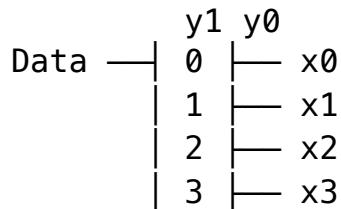
答案: a) **unsigned** (无符号) : 134 – b) **signed** (二进制补码有符号) :  $134 - 256 = -122$

---

## 第 3 题 – 1:4 解复用器设计 (10 分)

题目: 画出 1:4 解复用器的逻辑电路图 [English] Draw the logic circuit diagram of a 1:4 demultiplexer

符号:



逻辑方程:

$$\begin{aligned}x_0 &= \text{Data and } (\text{not } y_1) \text{ and } (\text{not } y_0) \\x_1 &= \text{Data and } (\text{not } y_1) \text{ and } y_0 \\x_2 &= \text{Data and } y_1 \text{ and } (\text{not } y_0) \\x_3 &= \text{Data and } y_1 \text{ and } y_0\end{aligned}$$

---

## 第 4 题 – RS 锁存器仿真 (5 分)

题目: 给定输入 A, B, C 在时间  $t_0-t_4$  的值, 写出 X 和 Y 输出的值 [English] Given inputs A, B, C values at times  $t_0-t_4$ , write the values of X and Y outputs

输入序列:

A = 1		0		1		0		1
B = 0		0		1		0		0
C = 0		1		0		0		0

$t_0 \quad t_1 \quad t_2 \quad t_3 \quad t_4$

---

## 第 5 题 – VHDL 代码 RTL 分析 (10 分)

题目: 画出以下 VHDL 代码的 RTL 视图, 并正确命名该电路 [English] Draw the RTL view of the following VHDL code and correctly name the circuit

```

library ieee; use ieee.std_logic_1164.all; use ieee.numeric_std.all;
entity XXX is port(a, b : in std_logic; c : out std_logic); end entity;
architecture beh of XXX is
begin
process (a,b)
variable d:std_logic:='0';
variable e:integer range 0 to 25:=0;
begin
if b='0' then
e:=0; d:='0';
elsif rising_edge(a) then
if e<25 then
e:=e+1;
else
e:=0; d:=not d;
end if;
end if;
c<=d;
end process;
end architecture;

```

功能分析: – b='0'是异步复位 – a 的上升沿触发 – 计数到 25 后输出翻转 – 名称: 26 分频器 (Frequency Divider by 26)

RTL 视图包含: – 5 位计数器 (0–25) – 比较器 (e < 25, e = 25) – D 触发器 (存储 d) – 异步复位逻辑

---

## 第 6 题 – 解复用器 VHDL 描述 (5 分)

题目:用最简单的方式描述第 3 题的解复用器 [English] Describe the demultiplexer from question 3 in the simplest way

```

library ieee; use ieee.std_logic_1164.all; use ieee.numeric_std.all;
entity yyy is
port(Data: in std_logic;
      y: in std_logic_vector(1 downto 0);
      x: out std_logic_vector(3 downto 0));
end entity;
architecture dataflow of yyy is
begin
x <= (others => '0');
x(to_integer(unsigned(y))) <= Data;
end architecture;

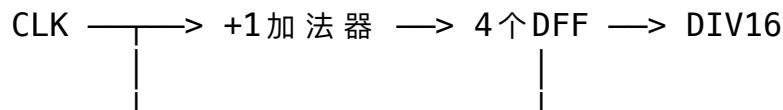
```

---

## 第 7 题 – 16 分频器设计 (5 分)

题目：在未完成的电路中添加最少元件，创建 CLK 的同步 16 分频器（使用 +1 加法器）[English]  
Add minimum components to the incomplete circuit to create a synchronous divide-by-16 of CLK (using +1 adder)

设计思路：



- 4 位计数器 (0-15 循环)
- DIV16 = Q3 (最高位)
- 每 16 个时钟周期完成一次

---

## 第 8 题 – 电机控制电路 (5 分)

题目：设计电路：按一次双极按钮开启电机，再按关闭。上电后电机必须关闭。[English] Design a circuit: press a double-pole button once to start motor, press again to stop. Motor must be off after power-up.

关键组件：– DFF 触发器 – 异步清零（上电复位）– 继电器驱动 – 防抖动电路

---

## 第 9 题 – 附加题：边沿检测器 (10 分)

题目：当 D 输入改变值时，Y 输出一个时钟周期的'1'。RESET 是同步复位。[English] When D input changes value, Y outputs '1' for one clock cycle. RESET is synchronous reset.

```
library ieee; use ieee.std_logic_1164.all; use ieee.numeric_std.all;
entity Edges is
  port (clk, Reset, D : in std_logic; Y: out std_logic);
end entity;
architecture rtl of Edges is
  signal D_prev : std_logic := '0';
begin
  process(clk)
  begin
    if rising_edge(clk) then
      if Reset = '1' then
        D_prev <= '0';
        Y <= '0';
      else
        if D /= D_prev then
          Y <= '1';
        else
          Y <= '0';
        end if;
      end if;
    end if;
  end process;
end architecture;
```

```
    Y <= '0';
end if;
D_prev <= D;
end if;
end if;
end process;
end architecture;
```

---

## 知识点总结

### 本次考试重点

1. Shannon 展开 (双输出)
2. 8 位有符号/无符号运算 (重要:  $390 \bmod 256$ )
3. 1:4 解复用器设计
4. RS 锁存器仿真
5. VHDL 代码 RTL 分析 (分频器识别)
6. 解复用器 VHDL 描述
7. 16 分频器设计
8. 电机控制电路
9. 边沿检测器设计

### 重要公式

- 8 位无符号范围: 0–255
- 8 位有符号范围: -128 to 127
- 溢出处理:  $\text{result} \bmod 2^n$