

Zkouska LSP 2022–01–24

Course: B0B35LSP – Logické systémy a procesory | BE5B35LSP – Logic Systems and Processors **University:** CVUT FEL (CVUT) – 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](#)

Zkouska LSP 2022–01–24 (anglická verze)

All derived versions – PDF files are not official answers; below are derived hints/solutions.

Informace o zkousce

- Datum: 2022–01–24
 - Jazyk: čeština / angličtina
 - Celkem: 50 bodů (část 1: 25 b > 8 + část 2: 25 b > 8)
-

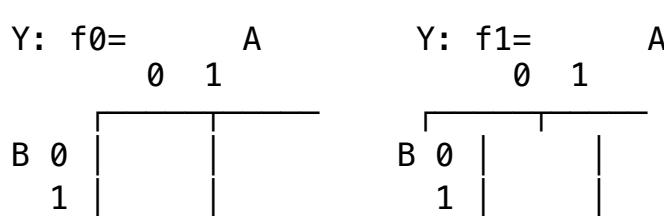
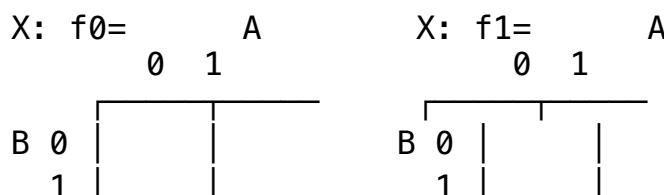
Úloha 1 –Shannonův rozklad (6 b)

Zadání: Rozložte funkce obvodu $X, Y = f(A, B, C)$ pomocí Shannonova rozkladu:

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

Zapište f_0 a f_1 pomocí Karnaughových map.

Pro X i Y nakreslete dvě K-map:



Úloha 2 –Podepsaná/nepodepsaná aritmetika (4 b)

Zadání: Provedte $96+97+98+99$ na 8bitové scítacče. Jaká je desetinná hodnota výsledku jako 8bitové číslo?

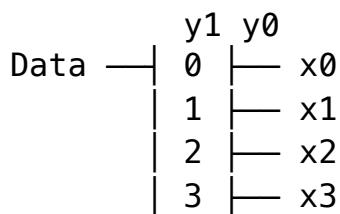
Výpočet: $- 96+97+98+99 = 390 - 390 \bmod 256 = 134$

Rешение: а) unsigned: 134 – б) signed (двоичный дополнение): $134 - 256 = -122$

Úloha 3 –Návrh demultiplexoru 1:4 (10 b)

Zadání: Nakreslete logické schéma demultiplexoru 1:4.

Symbol:



Rovnice:

$$x_0 = \text{Data} \text{ and } (\text{not } y_1) \text{ and } (\text{not } y_0)$$

$$x_1 = \text{Data} \text{ and } (\text{not } y_1) \text{ and } y_0$$

$$x_2 = \text{Data} \text{ and } y_1 \text{ and } (\text{not } y_0)$$

$$x_3 = \text{Data} \text{ and } y_1 \text{ and } y_0$$

Úloha 4 –Simulace RS záchytu (5 b)

Zadání: Pro dané hodnoty vstupů A, B, C v časech t0–t4 určete hodnoty výstupů X a Y.

Vstupní posloupnost:

$$\begin{array}{c|ccccc} A & 1 & 0 & 1 & 0 & 1 \\ B & 0 & 0 & 1 & 0 & 0 \\ C & 0 & 1 & 0 & 0 & 0 \\ \hline t_0 & t_1 & t_2 & t_3 & t_4 \end{array}$$

Úloha 5 –RTL analýza VHDL kódu (10 b)

Zadání: Nakreslete RTL pohled následujícího VHDL kódu a správně pojmenujte obvod.

```
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;

```

Funkční analýza: – $b='0'$ je asynchronní reset – spouštění na náběžnou hranu a – po dosažení 25 se výstup překládí – Název: dělicka frekvence /26

RTL obsahuje: – 5bitový čítač (0–25) – porovnání ($e < 25$, $e = 25$) – D klopný obvod pro d – asynchronní reset

Úloha 6 –VHDL popis demultiplexoru (5 b)

Zadání: Popisťe demultiplexor z Úlohy 3 co nejjednodušejí.

```

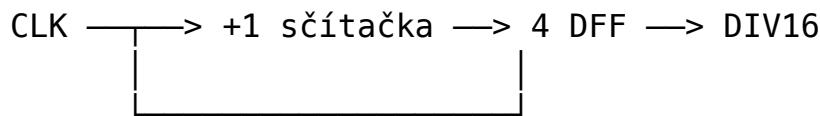
library ieee; use ieee.std_logic_1164.all; use ieee.numeric_std.all;
entity yyy is
  port(Data: in std_logic;
       y: in std_logic_vector(1 downto 0);
       x: out std_logic_vector(3 downto 0));
end entity;
architecture dataflow of yyy is
begin
  x <= (others => '0');
  x(to_integer(unsigned(y)))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

```

Úloha 7 –Návrh dělicky /16 (5 b)

Zadání: Doplňte minimální pocet prvků do neúplného schématu a vytvorťte synchronní dělicku CLK /16 (s využitím scítacky +1).

Návrh:



- 4bitový čítač (0–15)
- DIV16 = Q3 (MSB)
- perioda 16 taktů

Úloha 8 –Obvod řízení motoru (5 b)

Zadání: Navrhněte obvod: jedním stiskem dvoupólového tlacítka motor zapněte, dalsím stiskem vypněte. Po zapnutí napájení musí být motor vypnuty.

Klícové prvky: – D klopny obvod – asynchronní vynulování (power-on reset) – budic relé – odrusení tlacítka

Úloha 9 –Bonus: Detektor hrany (10 b)

Zadání: Když vstup D změní hodnotu, výstup Y bude '1' po jeden takt. RESET je synchronní.

```
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;
                D_prev <= D;
            end if;
        end if;
    end if;
end;
```

```
end process;  
end architecture;
```

Shrnutí témat

Hlavní okruhy

1. Shannonův rozklad (dva výstupy)
2. **8bitová signed/unsigned aritmetika** (důležité: $390 \bmod 256$)
3. **Demultiplexor 1:4**
4. Simulace RS záchytu
5. **RTL analýza VHDL** (rozpoznání dělicky)
6. VHDL popis demultiplexoru
7. Návrh dělicky /16
8. Rízení motoru
9. Detektor hrany

Důležité vzorce

- Rozsah 8bit unsigned: 0–255
- Rozsah 8bit signed: -128 až 127
- Prětecení: `result mod 2^n`