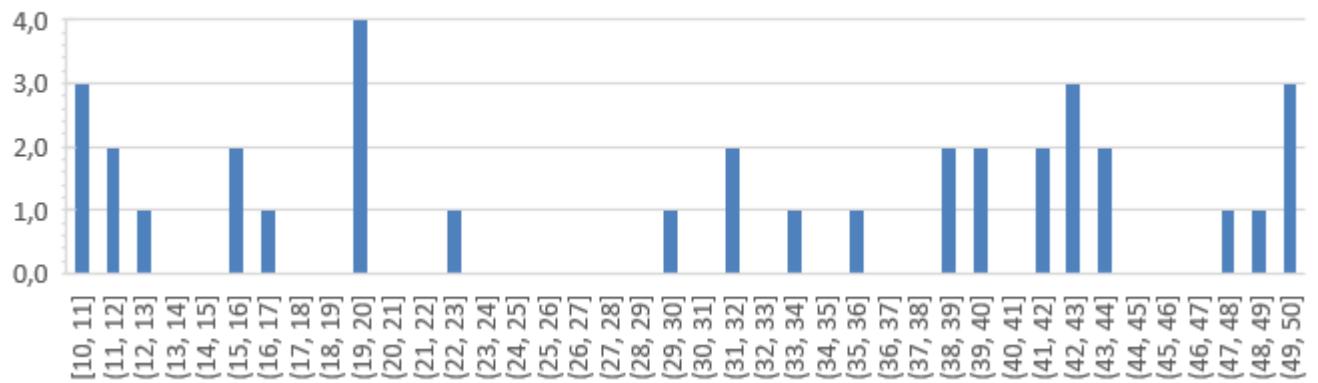
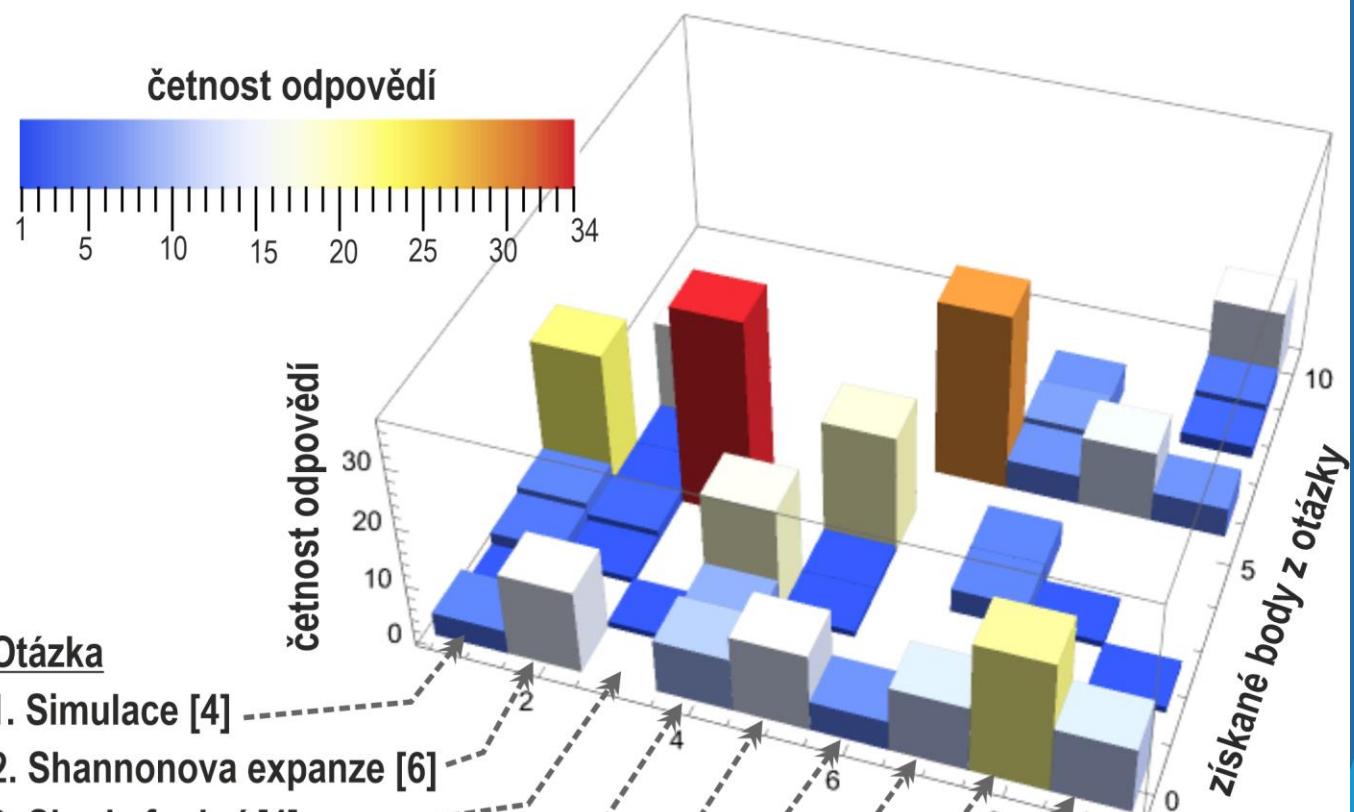


Statistické výsledky zkoušky LSP v úterý 28. května 2024

Četnost bodů z písemky



četnost odpovědí



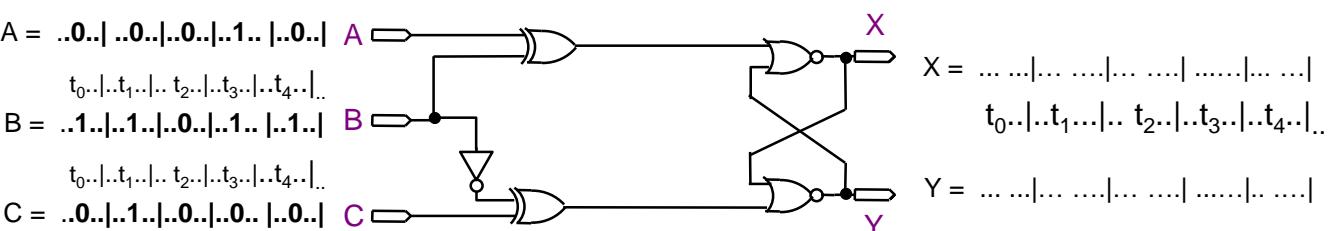
Odpovědi

1. Simulace [4]
2. Shannonova expanze [6]
3. Shoda funkcí [4]
4. Signed/Unsigned [2]
5. Definice automatu [4]
6. Kaskáda MUX [6]
7. Kaskáda ve VHDL [8]
8. Predikce skoků [6]
9. VHDL kód demultiplexoru [10]

Příjmení: Jméno:

Odpovědní arch písemky LSP ze dne 28.5.2024 - pište sem jen Vaše odpovědi

1. Vstupy A, B, C měly v časech t_0, t_1, t_2, t_3 hodnoty uvedené v obrázku. Napište hodnoty X a Y výstupů.
Předpokládejte, že intervaly mezi změnami vstupů jsou tak dlouhé, že lze zanedbat zpoždění hradel.



Zde
nepíšte

4

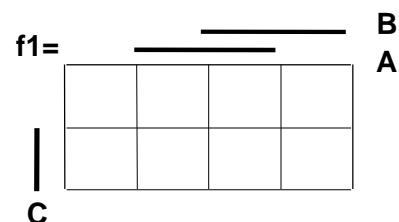
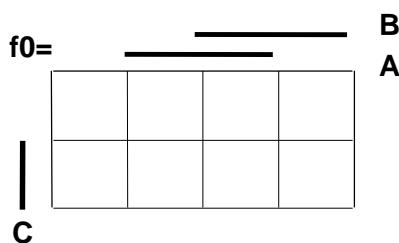
6

4

2

4

2. Funkci $X=f(A,B,C, X)$ obvodu z otázky 1, rozložte na tvar $X= (\text{not } X \text{ and } f_0(A, B, C)) \text{ or } (X \text{ and } f_1(A, B, C))$ pomocí Shannonovy expanze. **Výsledné funkce f0 a f1 zapište jako Karnaughovy mapy:**



Zde
nepíšte

4

6

4

2

4

3. Zaškrtnutím označte všechny logické funkce, které zde mají jinou funkci s nimi **shodnou**:

y1 <= (D or A) and (not D or C or A);

y1

y2 <= (not D and A) or (D and not A) or (C and A);

y2

y3 <= C or (D and C and B) or (not D and C and A);

y3

y4 <= (C and A) or (not D xor not A);

y4

Zde
nepíšte

4

6

4

2

4

4. Provedeme-li operaci s dekadickými čísly $254+255+256+257$ na 9 bitové binární sčítáčce, jakou hodnotu má výsledek, bereme-li ho jako 9bitové dekadické číslo a) bez znaménka (unsigned).....

b) se znaménkem v zobrazení ve druhém doplňku (signed).....

Zde
nepíšte

4

6

4

2

4

5. Doplňte: Automat Moore (Mealy) je uspořádaná šestice $M = < X, S, Z, \omega, \delta, s_0 \in S >$, kde

X je.....

S je.....

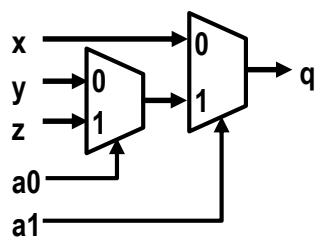
Z je.....

δ je zobrazení pro Moore..... pro Mealy.....

ω je zobrazení pro Moore..... pro Mealy.....

s_0 je

6. Použijte jen hradla AND, NAND, OR, NOR a invertory NOT a pomocí nich zapojte schéma vlevo.



Zde
nepíšte

6

qcon
4

qseq
4

7. Obvod z otázky 6 nahoře popište ve VHDL jak souběžným, tak sekvenčním příkazem

```
library ieee; use ieee.std_logic_1164.all;use ieee.numeric_std.all;  
entity Test20240528q6 is port( x, y ,z, a1,a0: in std_logic:= '0';  
qcon, qseq: out std_logic := '0')
```

```
end entity;
```

```
architecture rtl of Test20240528q6 is
```

```
begin
```

-- **qcon souběžným (concurrent) příkazem.....**

```
.....  
.....  
.....
```

-- **qseq sekvenčním příkazem.....**

```
iseq: process.....
```

```
.....begin.....
```

```
.....  
.....  
.....
```

```
.....  
.....  
.....
```

```
.....end process;.....
```

```
end architecture;
```

8. C program hledá **minimum** v poli **data**.

```
int data[] = { 0, 1, -2, -3, 4, -5, -6, 7, 8, 9 };
int min = INT_MAX; // =2147483647 (v <limits.h>)
for (int i = 0; i < sizeof(data)/sizeof(int); i++) // i<10
{ if (data[i] < min) min = data[i];
}
```

Kolik v něm bude chybných skokových predikcí za předpokladu, že for-loop s konstantnímimezemi se úsporně přeložila cyklem do-while a procesor používá pouze:

1bitové prediktory, které měly svůj výchozí stav Not-Taken, NT , misses=.....

2bitové prediktory, které měly svůj výchozí stav WT, Weakly Taken, misses=.....

6

9. a) Nakreslete symbolickou schematickou značku obvodu, který vznikne z následujícího kódu.

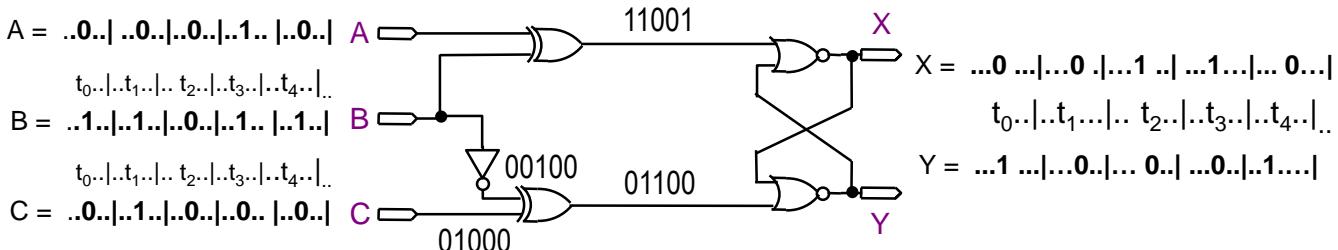
b) Nakreslete i jeho vnitřní zapojení, v němž použijete jen hradla OR, AND a NOT.

```
library ieee;use ieee.std_logic_1164.all;use ieee.numeric_std.all;
entity Test20240528q8 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 beh1 of Test20240528q8 is
begin
process (Data, y)
variable ix:integer range 0 to 3:=0;
begin ix:=to_integer(unsigned(y)); x<=(others=>'0'); x(ix)<=Data;
end process;
end architecture;
```

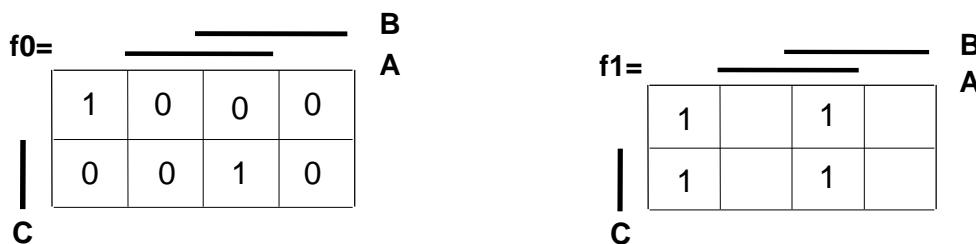
Symbol
4Schema
6

Výsledky

1. Vstupy A, B, C měly v časech t_0, t_1, t_2, t_3 hodnoty uvedené v obrázku. Napište hodnoty X a Y výstupů. Předpokládejte, že intervaly mezi změnami vstupů jsou tak dlouhé, že lze zanedbat zpoždění hradel.



2. Funkci $X=f(A,B,C)$ obvodu z otázky 1, rozložte na tvar $X= (\text{not } X \text{ and } f_0(A, B, C)) \text{ or } (X \text{ and } f_1(A, B, C))$ pomocí Shannonovy expanze. **Výsledné funkce f0 a f1 napište jako Karnaughovy mapy:**



3. $y_2 = y_4$

4. (*unsigned*).....510..... (*signed*)....-2.....

5. Automat Moore (Mealy): přednáška 8, snímek 19

6. Dvouvstupové multiplexory byly opakovaně v přednáškách,
jinak též Logické obvody na FPGA str. 78 až 79

7. správně takto, ale tolerovaly se i vnořené if a rozumně napsaný select a case

```
library ieee; use ieee.std_logic_1164.all; use ieee.numeric_std.all;
entity Test20240528q6 is port( x, y ,z, a1,a0: in std_logic:= '0';
                                qcon, qseq: out std_logic := '0');
end entity;
architecture rtl of Test20240528q6 is
begin
-- soubeznymi prikazy
  Qcon<= x when a1='0' else y when a0='0' else z;

-- sekvencni popis
  seq: process(x,y,z,a1,a0)
  begin
    if a1='0' then Qseq<=x;
    elsif a0='0' then Qseq<=y; else Qseq<=z;
    end if;
  end process;
end architecture;
```

8. součet chybných predikci for smyčky a vnitřního if příkazu je u obou prediktorů 7.

9. Kód popisuje demultiplexor, který se mnohokrát objevil v přednáškách.
Podrobně je popsán též v Logické obvody na FPGA str. 76 až 77.