

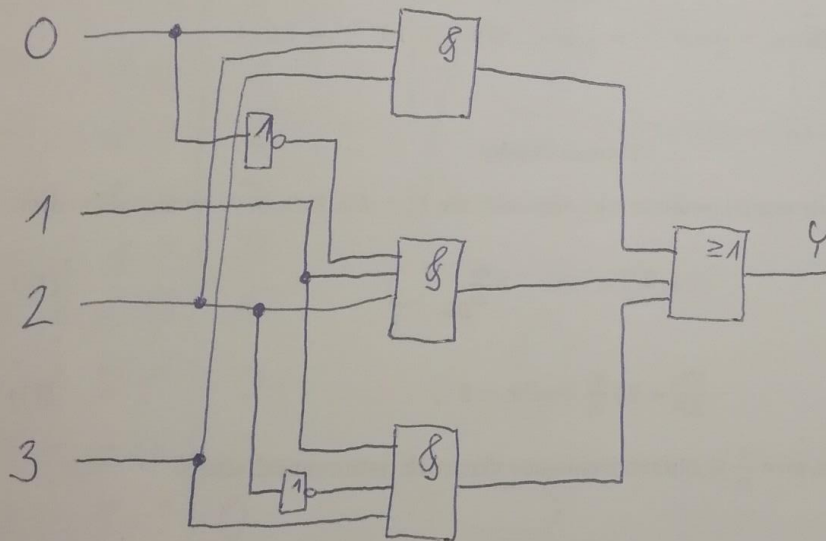
13

	a	b	c	d	Y
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	1
	0	1	1	0	1
	0	1	1	1	1
	1	0	0	0	1
	1	0	0	1	1
	1	0	1	0	1
	1	0	1	1	1
	1	1	0	0	1
	1	1	0	1	1
	1	1	1	0	1
	1	1	1	1	1

0000 0	1000 1	1100 1	0100 0
0010 0	1010 1	1110 1	0110 1
0011 0	1011 1	1111 1	0111 1
0001 0	1001 1	1101 1	0101 1

1 2 3

$$Y = a \cdot c \cdot d + \bar{a} \cdot b \cdot c + b \cdot \bar{c} \cdot d$$



MINIMÁLNÍ POČET HRADEL = 4.

$$g(x_3, x_2, x_1, x_0) = \bar{x}_3 \cdot (\bar{x}_2 \cdot x_1 + x_2 \cdot x_0) + (x_1 + x_0)$$

$$\left(\overline{x_3} \cdot \overline{x_2} \cdot x_1 + \overline{x_3} \cdot x_2 \cdot x_0 \right) + \left(\overline{x_1 + x_0} \right)$$

$$\overline{X_1 + X_0} = \overline{X_1} \cdot \overline{X_0}$$

X_3	X_2	X_1	X_0	
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	0	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Hand-drawn Karnaugh map for a 4-variable function with variables x_0, x_1, x_2, x_3 . The map shows a 4x4 grid of cells with binary values 0 or 1. The top row ($x_3=0$) contains all 1s. The second row ($x_3=1$) contains 1 in the first column ($x_1x_2=00$) and 0s elsewhere. The third row ($x_3=0$) contains 1 in the first column ($x_1x_2=00$) and 0s elsewhere. The bottom row ($x_3=1$) contains 0s in the first three columns and 1 in the last column ($x_1x_2=11$). Arrows indicate the grouping of the top row ($x_3=0$), the first column ($x_1x_2=00$), and the last column ($x_1x_2=11$).

$x_3 \backslash x_2 x_1 x_0$	000	010	100	110
0	1	1	1	1
1	1	0	0	0
0	1	0	0	0
1	0	0	0	1

$$4 = 0$$

SOUSEDICI SHČER V KM, TZN. JE ZDE
HAZARD.

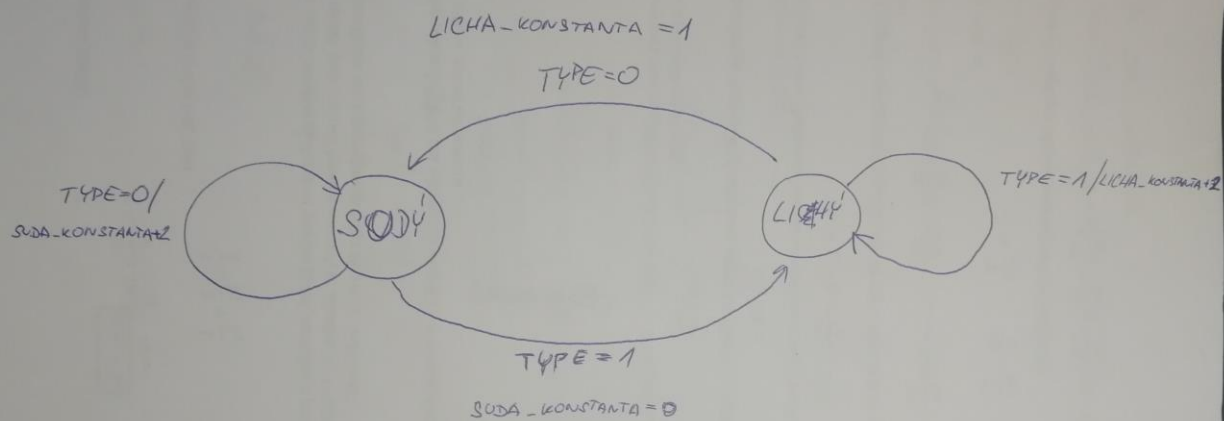
②

		x_3		x_2	
		x_3		x_2	
x_0	x_1	1	1	1	1
		1	0	0	0
		1	0	0	1
		0	0	0	1

5

$$Y = \overline{x_3} \cdot \overline{x_2} \cdot \overline{x_0} + \overline{x_3} \cdot \overline{x_2} \cdot x_0 + x_1 \cdot x_0 \cdot \overline{x_3} + \overline{x_3} \cdot x_2 \cdot x_0 + \overline{x_3} \cdot x_2 \cdot \overline{x_1} + x_1 \cdot \overline{x_2} \cdot \overline{x_3}$$

④



⑤

entity gen_pulse is

port (clk, reset_n : in std_logic;

data_o : out std_logic);

-- pyshupni pulsey

end entity;

architecture FULL of gen_pulse is

signal count : unsigned (4 downto 0);

begin

process (clk)

if (rising_edge(clk)) then

if (reset_n = '0') then

count := (others => '0');

data_o = '0';

elsif (count >= 0 and count <= 25) then

count := count + 1;

data_o <= '1';

else

count := count + 1;

data_o <= '0';

end if;

end if;

end process;

end Behavioral;