1. cvičenie

GitHub link

https://github.com/xnanko00/Digital-electronics-1

2. cvičenie

De Morganove zakony

vzorec

$$f(c,b,a) = \overline{b}\,a + \overline{c}\,\overline{b}$$

$$f(c,b,a)_{\mathrm{NAND}} = \overline{\overline{\overline{b}\,a}\,\overline{\overline{c}\,\overline{b}}}$$

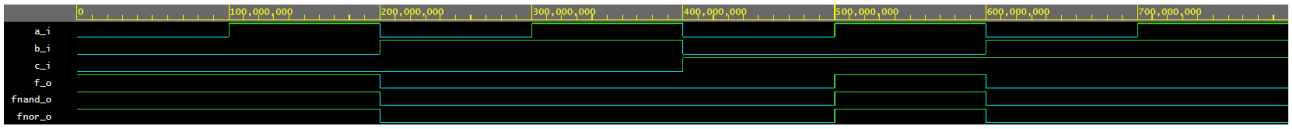
$$f(c,b,a)_{\mathrm{NOR}} = \overline{b+\overline{a}} + \overline{c+b}$$

VHDL kód

```
architecture dataflow of gates is
begin
        f_o <= ((not b_i) and a_i) or ((not c_i) and (not b_i));
    fnand_o <= not (not ((not b_i) and a_i) and (not((not c_i) and (not b_i))));
    fnor_o <= not (b_i or (not a_i)) or (not (c_i or b_i));

end architecture dataflow;
```

Screenshot



EDA Playground link

https://www.edaplayground.com/x/E2GK

Tabuľka

c	b	a	f(c,b,a)
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

3. cvičenie

Distributive laws

vzorec

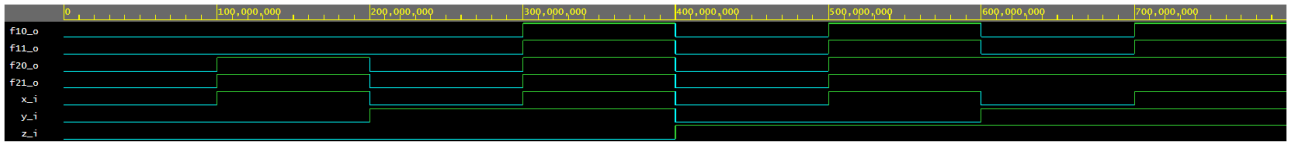
$$x \cdot y + x \cdot z = x \cdot (y + z)$$
$$(x + y) \cdot (x + z) = x + (y \cdot z)$$

VHDL kód

```
architecture dataflow of gates is
begin
    f10_o <= (x_i and y_i) or (x_i and z_i);
    f11_o <= (x_i and (y_i or z_i));
    f20_o <= (x_i or y_i) and (x_i or z_i);
    f21_o <= x_i or (y_i and z_i);

end architecture dataflow;
```

Screenshot



EDA Playground link

https://www.edaplayground.com/x/8Nqu

Tabuľka

z	y	x	x and y or x and z	x and (y or z)	(x or y) and (x or z)	x or (y and z)
0	0	0	0	0	0	0
0	0	1	0	0	1	1
0	1	0	0	0	0	0
0	1	1	1	1	1	1
1	0	0	0	0	0	0
1	0	1	1	1	1	1
1	1	0	0	0	1	1
1	1	1	1	1	1	1