

 xnanko00 / Digital-electronics-1

&lt;&gt; Code

! Issues

🔗 Pull requests

▶ Actions

📁 Projects

🛡 Security

📈 Insights

🔑 main ▾

...

Digital-electronics-1 / Labs / 01-gates / README.md



xnanko00 finálna úprava v2

🕒 History

👤 1 contributor

Raw

Blame



122 lines (85 sloc) | 2.3 KB

# 1. cvičenie

## GitHub link

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

## 2. cvičenie

## De Morganove zákony

vzorec

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

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

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

## VHDL kód

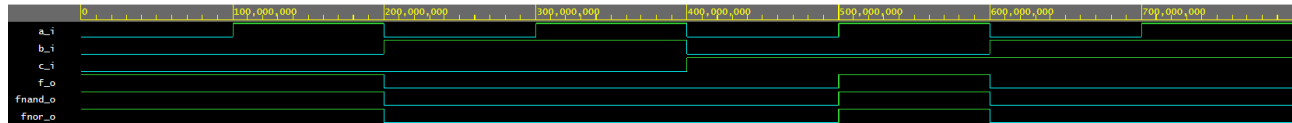
```

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

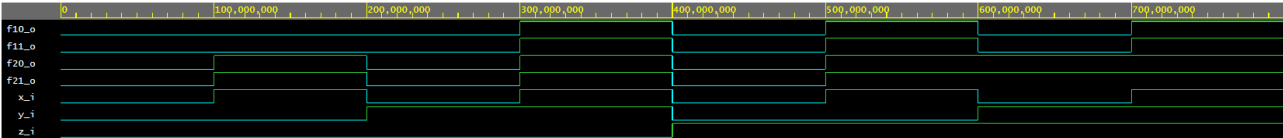
$$x \cdot y + x \cdot z = x \cdot (y + z)$$
$$(x + y) \cdot (x + z) = x + (y \cdot z)$$

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>

Tabul'ka

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

