



Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут»

Лабораторна робота №1

*з дисципліни «Автоматизоване проектування
комп'ютерних систем»*

«Побудова комбінаційної схеми (перетворювача коду)»

Виконав студент IV курсу

групи: КВ-11

ПІБ: Терентьєв Іван Дмитрович

Перевірів: _____

Київ 2024

Завдання для лабораторної роботи

- 1) Описати логічні елементи на мові Active-HDL згідно з варіанту в таблиці.
- 2) Побудувати логічну схему, використовуючи логічні елементи, описані мовою A-HDL, описати поведінку КС, побудувати часові діаграми роботи КС.
- 3) Описати поведінку перетворювача коду мовою A-HDL та побудувати часові діаграми роботи перетворювача коду.

Варіант 22(00010110)

Таблиця істинності

X ₁	X ₂	X ₃	Y ₁	Y ₂
0	0	0	0	1
0	0	1	1	1
0	1	0	0	0
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	0	1	1
1	1	1	0	1

Логічні елементи: ЗІ-НЕ, ЗАБО-НЕ

Час: 10 ps

Завдання №1: Описати логічні елементи на мові Active-HDL згідно з варіанту в таблиці

3I-HE(3NAND)

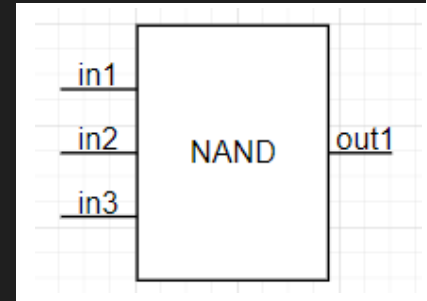
```
library IEEE;
use IEEE.std_logic_1164.all;

entity nand3 is
    port(
        in1 : in STD_LOGIC;
        in2 : in STD_LOGIC;
        in3 : in STD_LOGIC;
        out1 : out STD_LOGIC
    );
end nand3;

architecture nand3 of nand3 is
begin

    out1<= (not(in1 and in2 and in3)) after 10 ps;

end nand3;
```



3АБО-HE(3NOR)

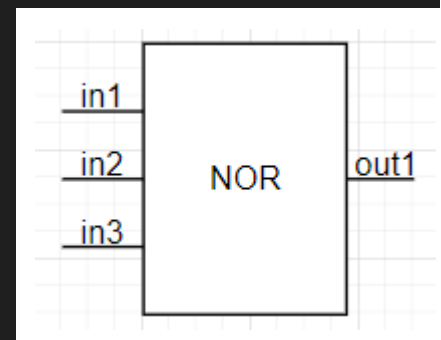
```
library IEEE;
use IEEE.std_logic_1164.all;

entity nor3 is
    port(
        in1 : in STD_LOGIC;
        in2 : in STD_LOGIC;
        in3 : in STD_LOGIC;
        out1 : out STD_LOGIC
    );
end nor3;

architecture nor3 of nor3 is
begin

    out1<= (not(in1 or in2 or in3)) after 10 ps;

end nor3;
```



Завдання №2: Побудувати логічну схему, використовуючи логічні елементи, описані мовою A-HDL, описати поведінку КС, побудувати часові діаграми роботи КС

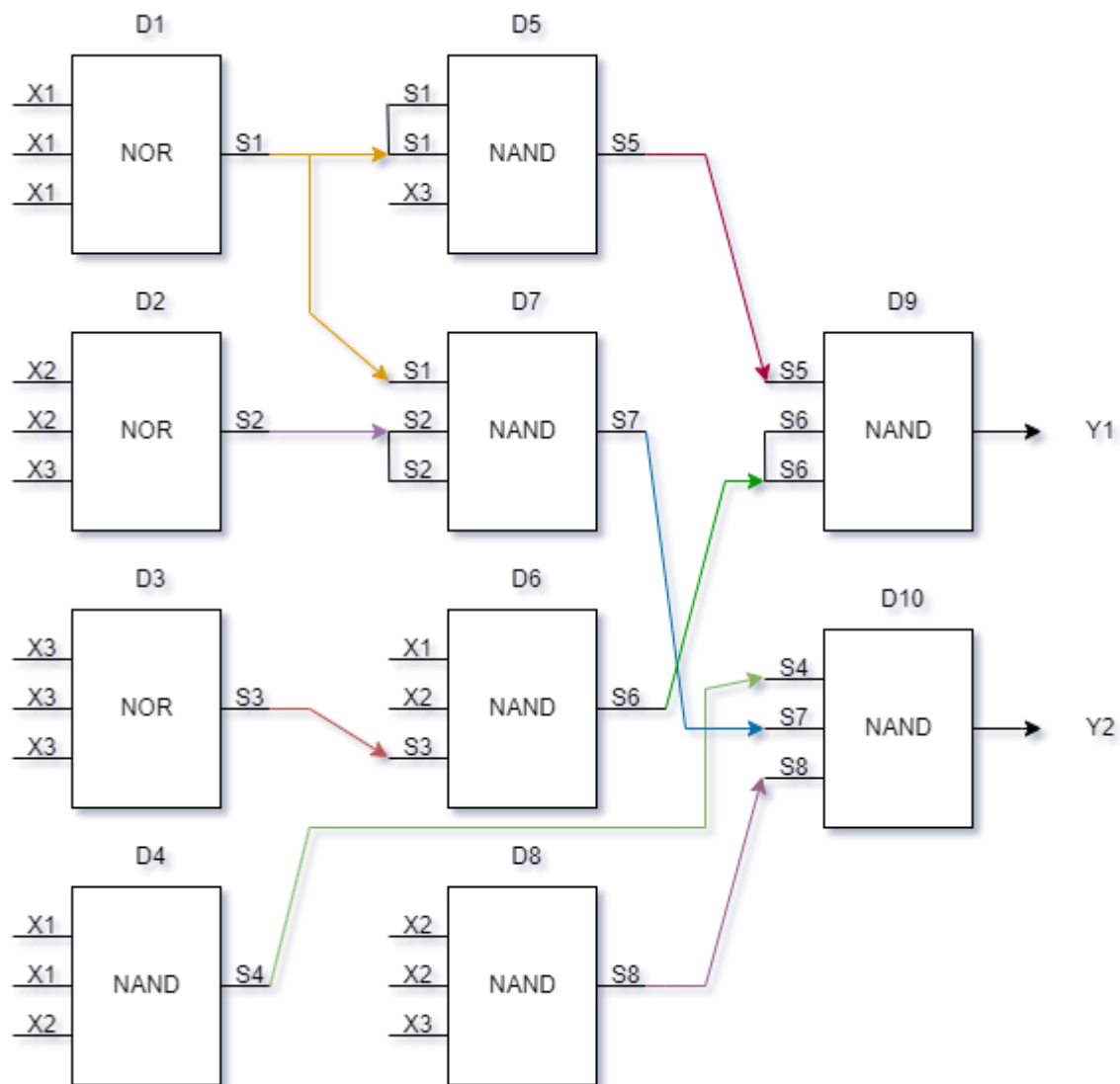


Рис. 1 – Логічна схема

```

library IEEE;
use IEEE.std_logic_1164.all;

entity task1 is
    port (
        X1: in STD_logic;
        X2: in STD_logic;
        X3: in STD_logic;
        Y1: out STD_logic;
        Y2: out STD_logic
    );
end task1;

architecture task1 of task1 is
    component nor3 is
        port (in1:in STD_logic; in2:in STD_logic;in3: in STD_logic; out1: out
STD_logic);
    end component;

    component nand3 is
        port (in1:in STD_logic; in2:in STD_logic;in3: in STD_logic; out1: out
STD_logic);
    end component;

    signal S1,S2,S3,S4,S5,S6,S7,S8:STD_Logic;

begin
    D1:nor3 port map(X1,X1,X1,S1);
    D2:nor3 port map(X2,X2,X2,S2);
    D3:nor3 port map(X3,X3,X3,S3);
    D4:nand3 port map(X1,X1,X2,S4);
    D5:nand3 port map(S1,S1,X3,S5);
    D6:nand3 port map(X1,X2,S3,S6);
    D7:nand3 port map(S1,S2,S2,S7);
    D8:nand3 port map(X2,X2,X3,S8);
    D9:nand3 port map(S5,S6,S6,Y1);
    D10:nand3 port map(S4,S7,S8,Y2);
end task1;

```

Логічна схема, описана мовою A-HDL

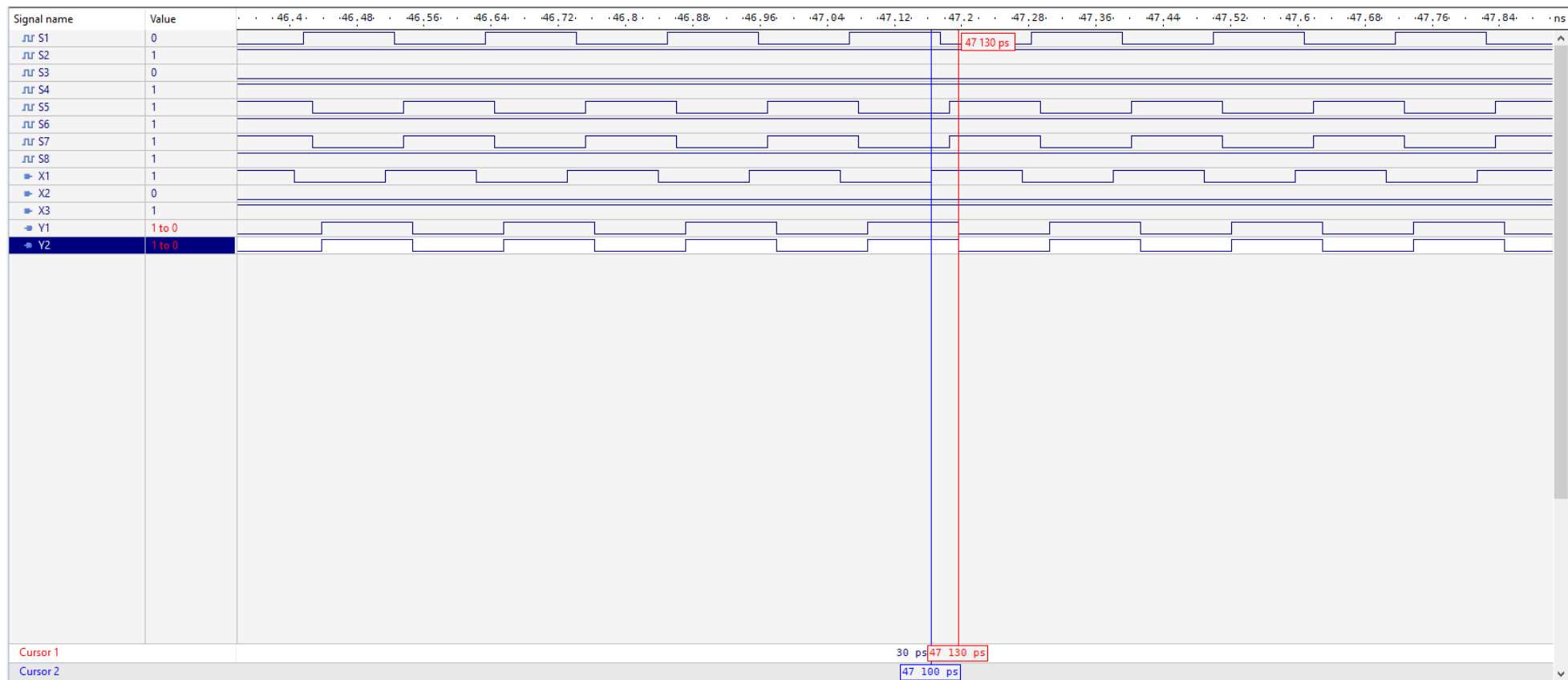


Рис. 2 – Часова діаграма роботи КС

Time	Delta	/task...	/task...	/task...	/task...	/task...	/task...	/task...	/task...	/task1/X1	/task1/X2	/task1/X3	/task1/Y1	/task1/Y2
25900 ps	0	1	1	0	1	0	1	0	1	1	0	1	1	1
25910 ps	0	0	1	0	1	0	1	0	1	1	0	1	1	1
25920 ps	0	0	1	0	1	1	1	1	1	1	0	1	1	1
25930 ps	0	0	1	0	1	1	1	1	1	1	0	1	0	0
26000 ps	0	0	1	0	1	1	1	1	1	1	0	1	0	0
26010 ps	0	1	1	0	1	1	1	1	1	0	0	1	0	0
26020 ps	0	1	1	0	1	0	1	0	1	0	0	1	0	0
26030 ps	0	1	1	0	1	0	1	0	1	0	0	1	1	1
26100 ps	0	1	1	0	1	0	1	0	1	1	0	1	1	1
26110 ps	0	0	1	0	1	0	1	0	1	1	0	1	1	1
26120 ps	0	0	1	0	1	1	1	1	1	1	0	1	1	1
26130 ps	0	0	1	0	1	1	1	1	1	1	0	1	0	0
26200 ps	0	0	1	0	1	1	1	1	1	0	0	1	0	0
26210 ps	0	1	1	0	1	1	1	1	1	0	0	1	0	0
26220 ps	0	1	1	0	1	0	1	0	1	0	0	1	0	0
26230 ps	0	1	1	0	1	0	1	0	1	0	0	1	1	1
26300 ps	0	1	1	0	1	0	1	0	1	1	0	1	1	1
26310 ps	0	0	1	0	1	0	1	0	1	1	0	1	1	1
26320 ps	0	0	1	0	1	1	1	1	1	1	0	1	1	1
26330 ps	0	0	1	0	1	1	1	1	1	1	0	1	0	0
26400 ps	0	0	1	0	1	1	1	1	1	0	0	1	0	0
26410 ps	0	1	1	0	1	1	1	1	1	0	0	1	0	0
26420 ps	0	1	1	0	1	0	1	0	1	0	0	1	0	0
26430 ps	0	1	1	0	1	0	1	0	1	0	0	1	1	1
26500 ps	0	1	1	0	1	0	1	0	1	1	0	1	1	1
26510 ps	0	0	1	0	1	0	1	0	1	1	0	1	1	1
26520 ps	0	0	1	0	1	1	1	1	1	1	0	1	1	1
26530 ps	0	0	1	0	1	1	1	1	1	1	0	1	0	0
26600 ps	0	0	1	0	1	1	1	1	1	0	0	1	0	0
26610 ps	0	1	1	0	1	1	1	1	1	0	0	1	0	0
26620 ps	0	1	1	0	1	0	1	0	1	0	0	1	0	0
26630 ps	0	1	1	0	1	0	1	0	1	0	0	1	1	1
26700 ps	0	1	1	0	1	0	1	0	1	1	0	1	1	1
26710 ps	0	0	1	0	1	0	1	0	1	1	0	1	1	1
26720 ps	0	0	1	0	1	1	1	1	1	1	0	1	1	1
26730 ps	0	0	1	0	1	1	1	1	1	1	0	1	0	0
26800 ps	0	0	1	0	1	1	1	1	1	0	0	1	0	0
26810 ps	0	1	1	0	1	1	1	1	1	0	0	1	0	0

Рис. 3 – Таблиця істинності з затримками

Як можна побачити з рис. 2, де відбувається перехід зі стану 0 0 1 до 1 0 1, час затримки коли змінилися Y_1 та Y_2 становив 30 ps. На рис. 3 можна побачити той самий перехід, але у вигляді таблиці істинності, затримка становила ті самі 30 ps.

Завдання №3: Описати поведінку перетворювача коду мовою A-HDL та побудувати часові діаграми роботи перетворювача коду

```
library IEEE;
use IEEE.std_logic_1164.all;
entity task2 is
generic(maxdelay:time:= 30ps);
port (
indata:in STD_logic_vector (2 downto 0);
outdata: out STD_logic_vector (1 downto 0)
);
end entity task2;
architecture task2 of task2 is
begin
with indata select
outdata<="10" after maxdelay when "000" ,
"10" after maxdelay when "001" ,
"00" after maxdelay when "010" ,
"01" after maxdelay when "011" ,
"01" after maxdelay when "100" ,
"10" after maxdelay when "101" ,
"01" after maxdelay when "110" ,
"11" after maxdelay when others;
end architecture task2;
```

Код перетворювача коду мовою A-HDL

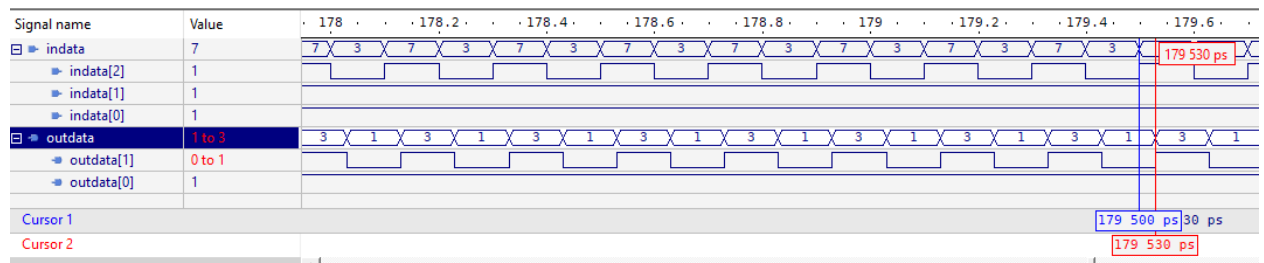


Рис. 4 – Часова діаграма роботи перетворювача коду

Висновки

В результаті виконання лабораторної роботи було розроблено та змодельовано комбінаційну схему перетворювача коду за допомогою мови А-HDL. Для побудови схеми використовувались логічні елементи 3І-НЕ та 3АБО-НЕ, що відповідали заданому варіанту. Було отримано таблицю істинності та часові діаграми роботи схеми, з яких видно, що затримка при переході між станами склала 30 ps, що відповідає очікуваним результатам. Також був успішно розроблений код перетворювача коду та перевірено його роботу шляхом моделювання часових діаграм.

Ця лабораторна робота дала змогу закріпити навички проектування комбінаційних схем, написання коду на мові опису апаратури та моделювання цифрових систем.