



Звіт

З лабораторної роботи № 1

З дисципліни «Моделювання комп'ютерних систем»

На тему: «Інсталяція та ознайомлення з середовищем розробки Xilinx ISE.

Ознайомлення зі стендом Elbert V2 - Spartan 3A FPGA»

Виконав: ст. гр. КІ-202

Побережник Ю.В..

Прийняв:

Козак Н. Б.

Мета: ознайомлення з середовищем розробки Xilinx ISE та побудова дешифратора 3-7.

Завдання

1. Створення облікового запису на www.xilinx.com.
2. Інсталяція Xilinx ISE та отримання ліцензії.
3. Побудова дешифратора 3->7 за допомогою ISE WebPACK™ Schematic Capture та моделювання його роботи за допомогою симулятора ISim.
4. Генерування Bit файла та тестування за допомогою стенда Elbert V2 – Spartan 3A FPGA.

Виконання завдання

Згідно завдання було розроблено схему дешифратора 3 в 7, використовуючи компоненти з бібліотеки, у новому, доданому до проекту Schematic файлу.

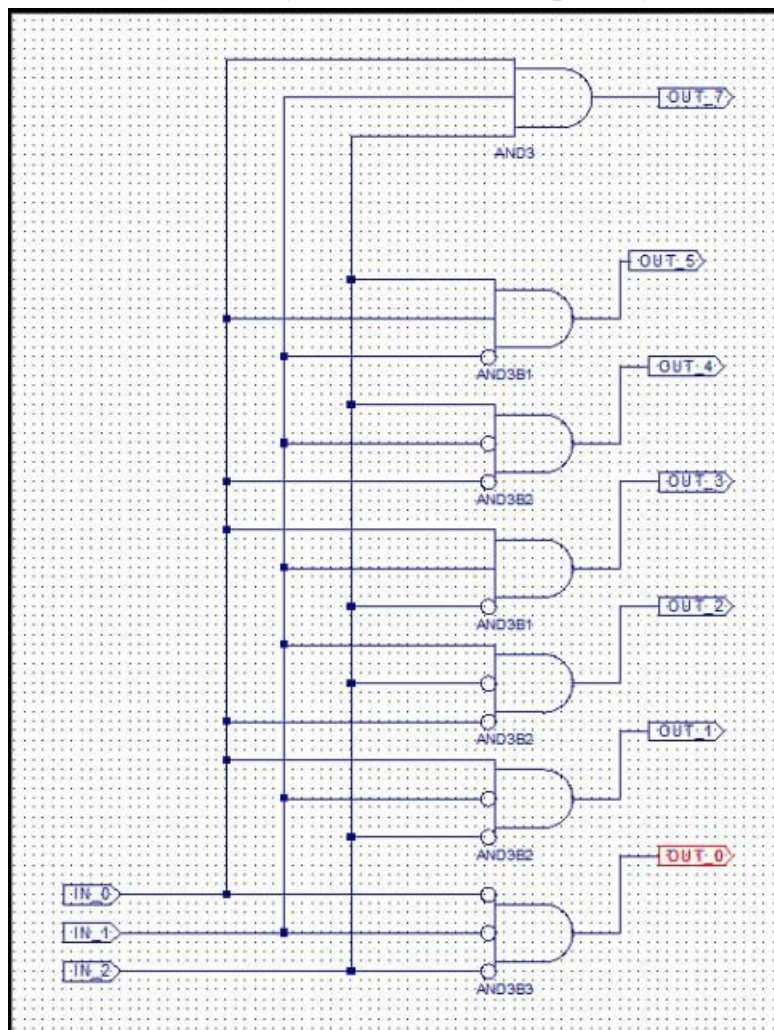


Рис.1. Схема дешифратора 3 в 7

Пізніше в проект було додано файл User Constraint, в якому змінений вміст файлу elbertv2.ucf був доданий.

```

5  # * Rename the used signals according to the your project                                     #
6  #+++++
7  #
8  # UCF for ElbertV2 Development Board
9  #+++++
10 CONFIG VCCAUX = "3.3" ;
11
12 # Clock 12 MHz
13 # NET "Clk" LOC = P129 | IOSTANDARD = LVCMOS33 | PERIOD = 12MHz;
14 #+++++
15 # LED
16 #+++++
17 NET "OUT_0" LOC = P46 | IOSTANDARD = "LVCMOS33" | SLEW = SLOW | DRIVE = 12;
18 NET "OUT_1" LOC = P47 | IOSTANDARD = "LVCMOS33" | SLEW = SLOW | DRIVE = 12;
19 NET "OUT_2" LOC = P48 | IOSTANDARD = "LVCMOS33" | SLEW = SLOW | DRIVE = 12;
20 NET "OUT_3" LOC = P49 | IOSTANDARD = "LVCMOS33" | SLEW = SLOW | DRIVE = 12;
21 NET "OUT_4" LOC = P50 | IOSTANDARD = "LVCMOS33" | SLEW = SLOW | DRIVE = 12;
22 NET "OUT_5" LOC = P51 | IOSTANDARD = "LVCMOS33" | SLEW = SLOW | DRIVE = 12;
23 # NET "LED[6]" LOC = P54 | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
24 NET "OUT_7" LOC = P55 | IOSTANDARD = "LVCMOS33" | SLEW = SLOW | DRIVE = 12;
25
26 #+++++
27 # DP Switches
28 #+++++
29 NET "IN_0" LOC = P70 | PULLUP | IOSTANDARD = "LVCMOS33" | SLEW = SLOW | DRIVE = 12;
30 NET "IN_1" LOC = P69 | PULLUP | IOSTANDARD = "LVCMOS33" | SLEW = SLOW | DRIVE = 12;
31 NET "IN_2" LOC = P68 | PULLUP | IOSTANDARD = "LVCMOS33" | SLEW = SLOW | DRIVE = 12;
32 # NET "DPSwitch[3]" LOC = P64 | PULLUP | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
33 # NET "DPSwitch[4]" LOC = P63 | PULLUP | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
34 # NET "DPSwitch[5]" LOC = P60 | PULLUP | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
35 # NET "DPSwitch[6]" LOC = P55 | PULLUP | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
36 # NET "DPSwitch[7]" LOC = P58 | PULLUP | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;

```

Рис.2. User Constraint файл

За допомогою симулятора ISim у режимі Simulation було перевірено роботу схеми.

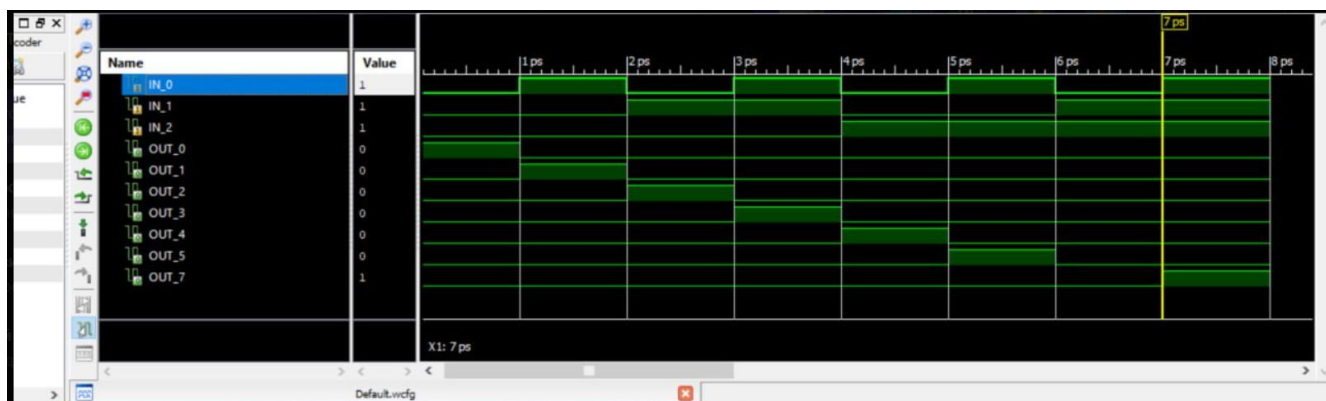


Рис.3. Запуск симуляції

Далі, використовуючи режим Simulation у симуляторі ISim я покроково перевіряв на роботу схему та код, використовуючи знання отриманні з методичних вказівок:



Рис. 4: Наступна симуляція

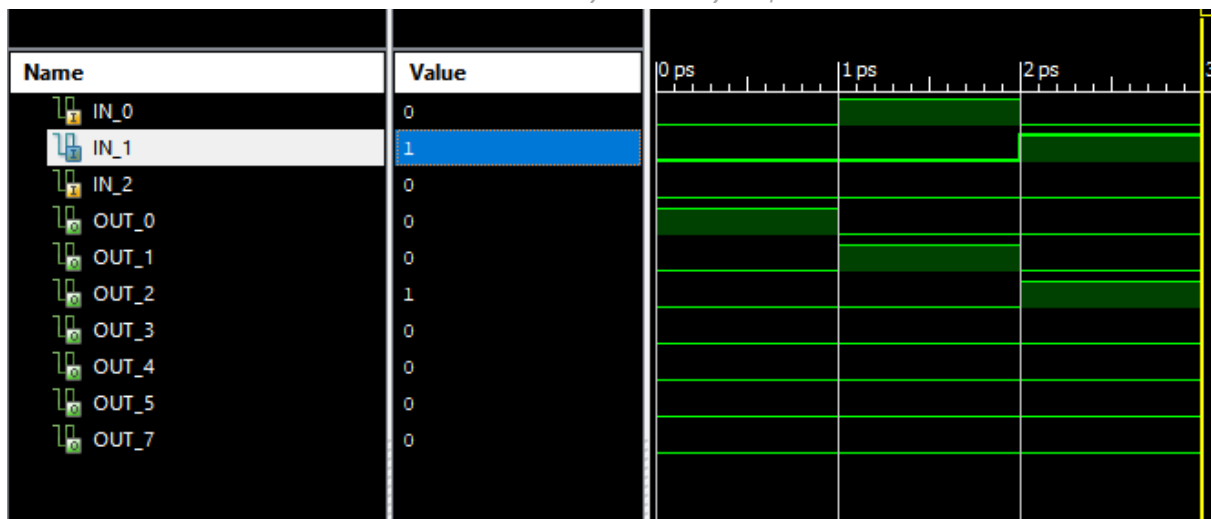


Рис. 5: Наступна симуляція

Кінцевий результат має наступний вигляд:

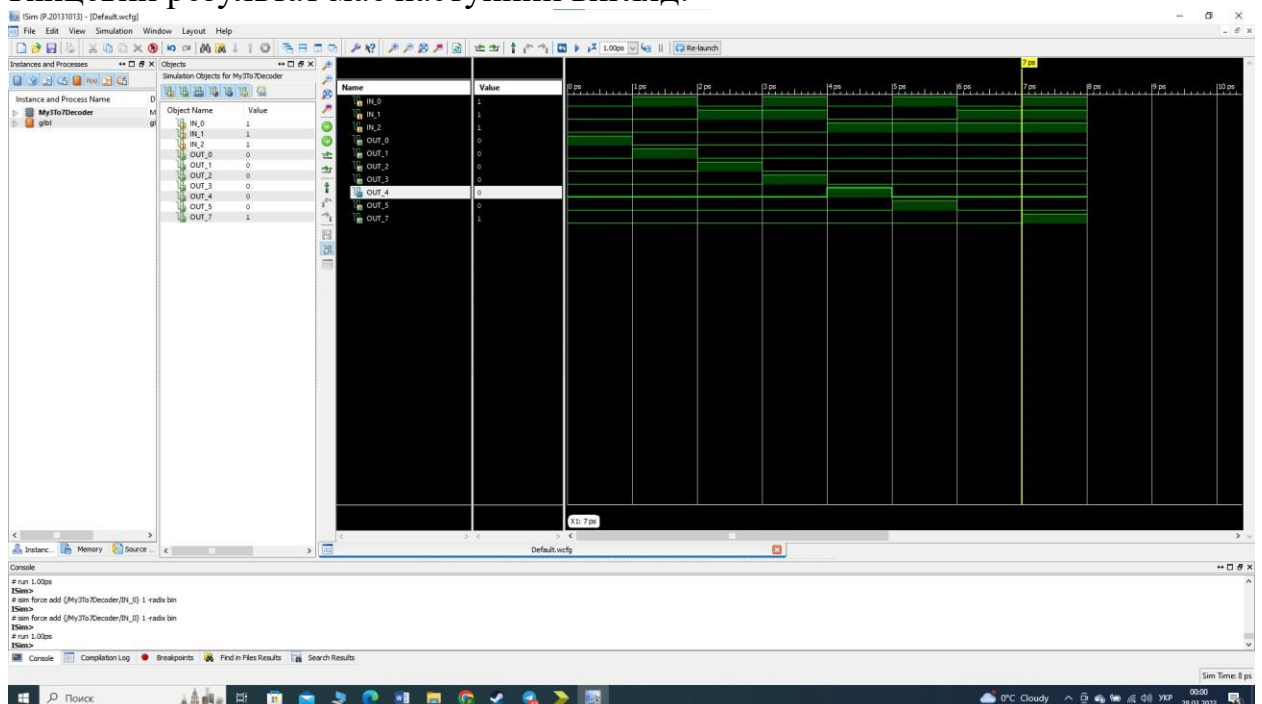


Рис. 6: Результат симуляція

Пізніше, після всіх перевірок на правильність коду та схеми, згідно методичних вказівок, було створено конфігураційний файл, і запущено в послідовності процеси, які успішно виконались:

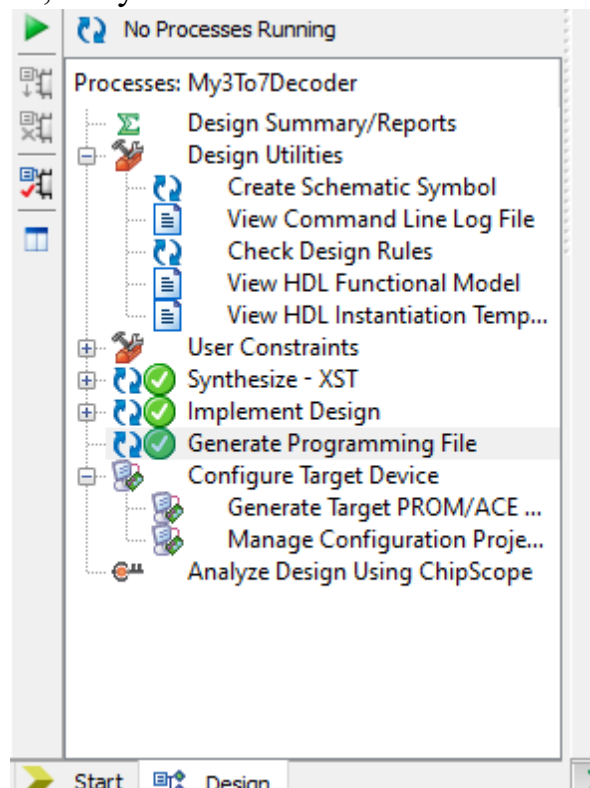
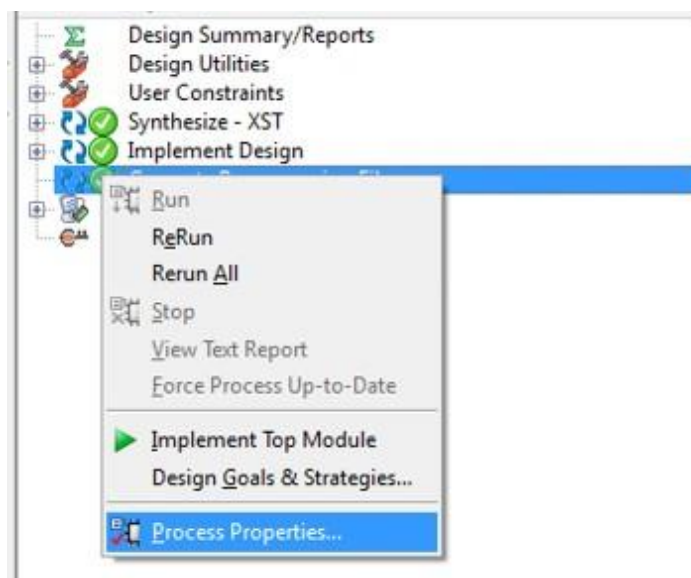


Рис. 7: Правильно виконані процеси

Після цього згенерував біт файл:



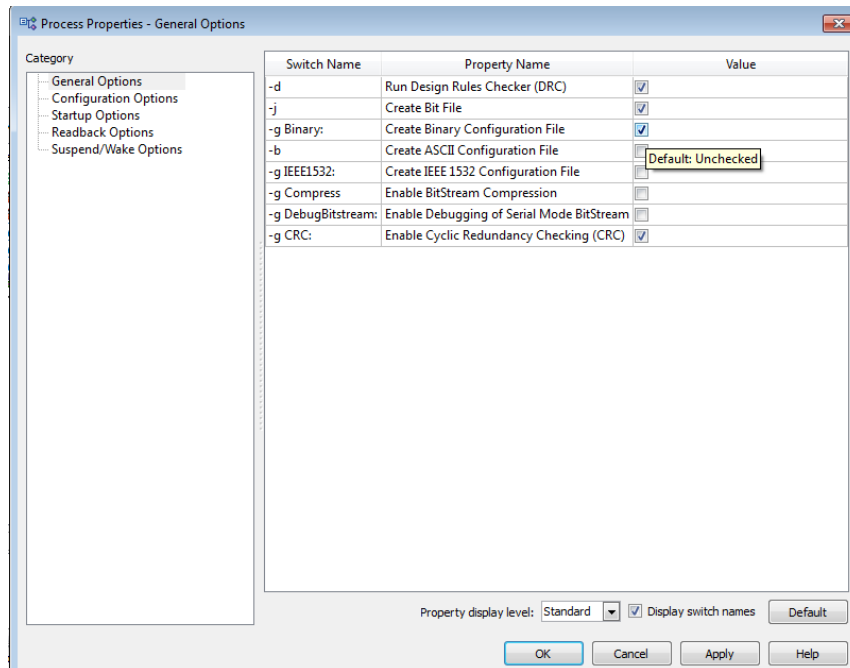


Рис. 8-9: Покрокове створення біт-файлу

Наступним кроком, з створеним біт-файлом та програмою для прошивки плати Elbert V2 FPGA Configuration tools запрограмовую плату, але спочатку потрібно зайти ком-порт через диспетчер пристроїв:

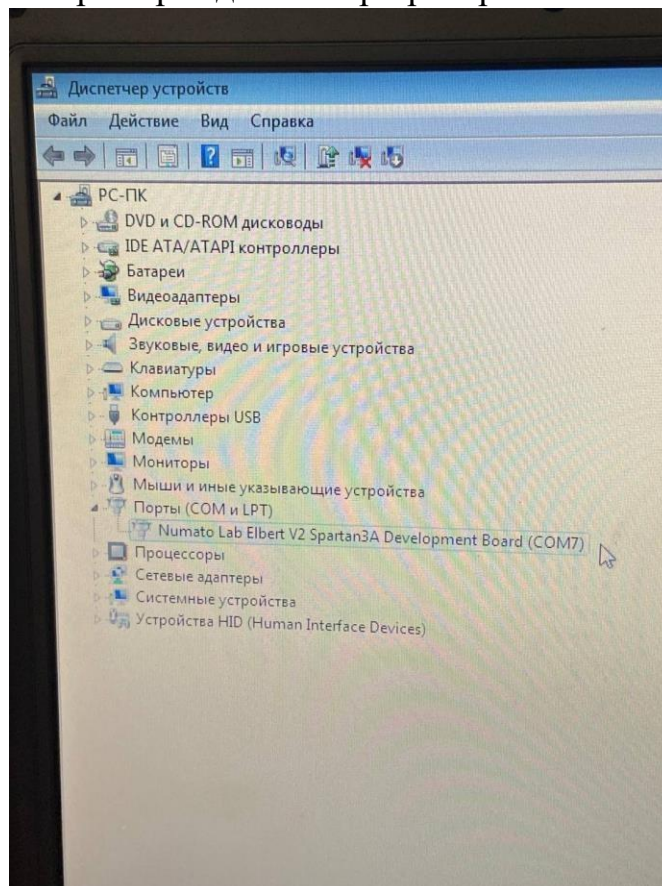


Рис. 10: Необхідний ком-порт, відображений у диспетчері пристроїв

Далі, вибираю у Elbert V2 FPGA Configuration tools вибираю необхідний ком-порт(COM7):

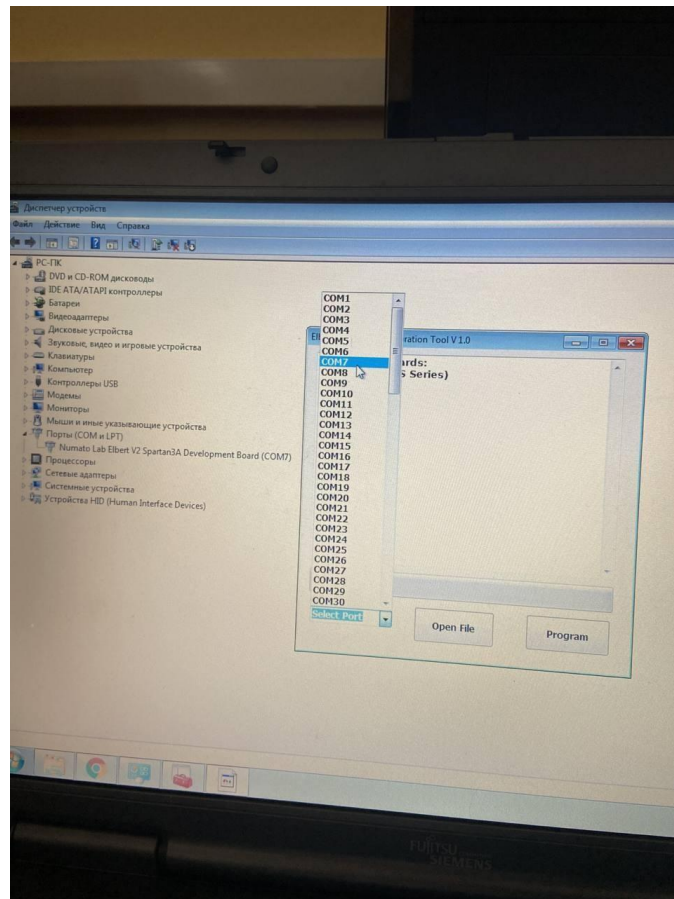


Рис. 11: Обраний ком-порт у програмі для прошивки плати (Elbert V2 FPGA Configuration tools)

Наступним і останнім кроком є завантаження у програму для прошивки плати Elbert V2 FPGA Configuration tools свого біт файлу та її програмування

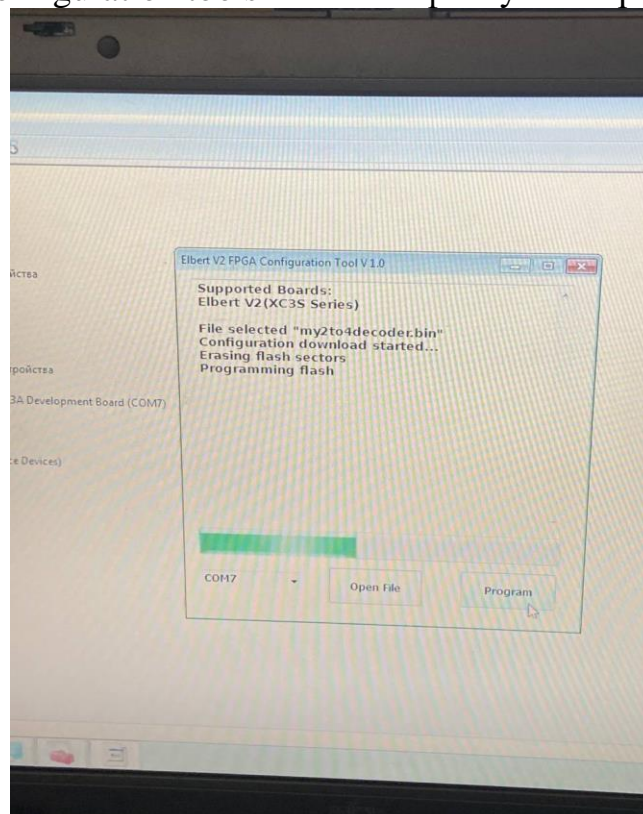


Рис. 12: Процес запрограмування плати своїм біт-файлу

Після завершення усіх процесів, плата буде мати наступний вигляд:

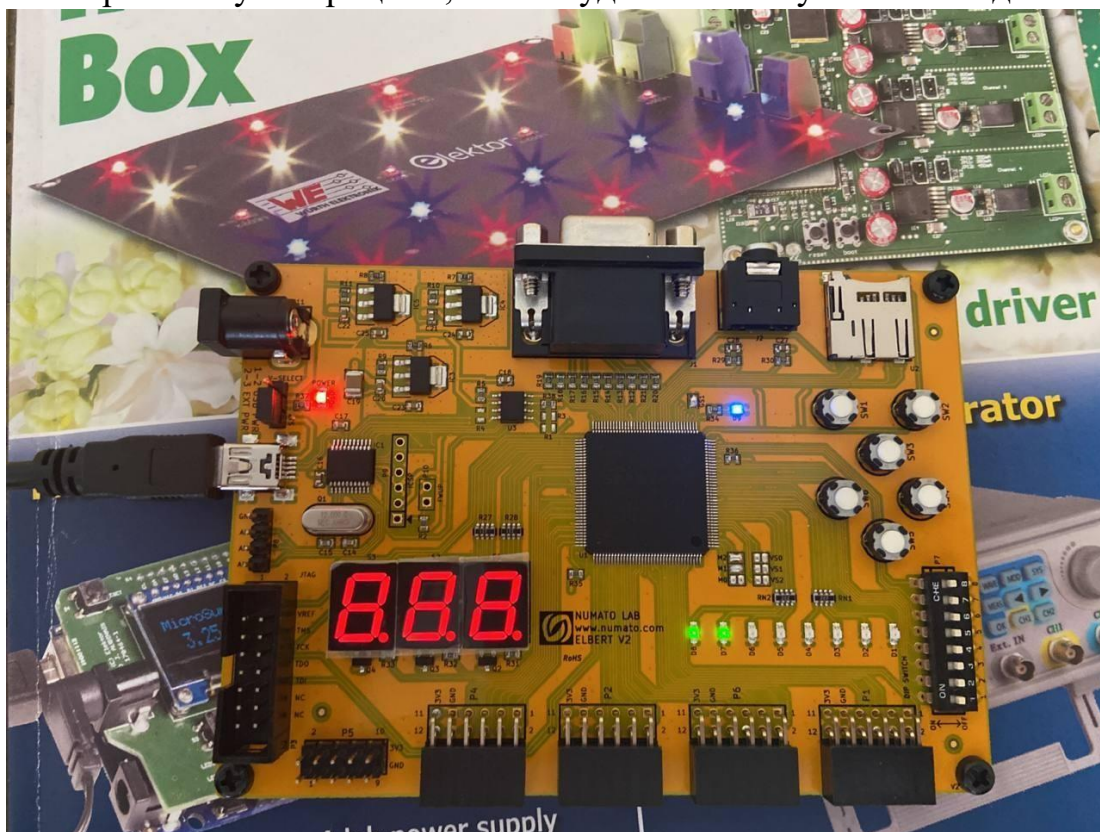


Рис. 13: Плата в робочому режимі

Висновок: У ході виконання цієї лабораторної роботи я створив дешифратор з трьома входами та сімома виходами, а також отримав досвід роботи з програмним середовищем Xilinx ISE.

Висновок: Під час виконання цієї лабораторної роботи я зміг побудувати 3-в-7 дешифратор і ознайомитися з середовищем Xilinx ISE.