МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Інститут комп'ютерних технологій, автоматики та метрології кафедра "Електронних обчислювальних машин"



Звіт

З лабораторної роботи №2

3 дисципліни: «Моделювання комп'ютерних систем»

На тему: «Структурний опис цифрового автомата Перевірка роботи автомата за допомогою стенда Elbert V2 – Spartan3A FPGA»

Варіант 5

Виконав:

ст. гр. КІ-202

Герцик М.А.

Прийняв:

Козак Н.Б.

ЛАБОРАТОРНА РОБОТА №2

Структурний опис цифрового автомата Перевірка роботи автомата за допомогою стенда Elbert V2 – Spartan3A FPGA

Мета роботи: На базі стенда реалізувати цифровий автомат світлових ефектів

Вхідні параметри

Стан#	LED_0	LED_1	LED_2	LED_3	LED_4	LED_5	LED_6	LED_7
0	1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1
2	0	1	0	0	0	0	0	0
3	0	0	0	0	0	0	1	0
4	0	0	1	0	0	0	0	0
5	0	0	0	0	0	1	0	0
6	0	0	0	1	0	0	0	0
7	0	0	0	0	1	0	0	0

- Пристрій повинен використовувати 12MHz тактовий сигнал від мікроконтролера IC1 і знижувати частоту за допомогою внутрішнього подільника. Мікроконтролер IC1 є частиною стенда Elbert V2 Spartan 3A FPGA. Тактовий сигнал заведено нв вхід LOC = P129 FPGA (див. Додаток 1).
- Інтерфейс пристрою повинен мати вхід синхронного скидання (RESET).
- Інтерфейс пристрою повинен мати вхід керування режимом роботи (MODE):
 - \circ Якщо MODE=0 то стан пристрою інкрементується по зростаючому фронту тактового сигналу пам'яті станів (0->1->2->3->4->5->6->7->0...).
 - \circ Якщо *MODE=1* то стан пристрою декрементується по зростаючому фронту тактового сигналу пам'яті станів (0->7->6->5->4->3->2->1->0...).
- Інтерфейс пристрою повинен мати однорозрядний вхід (TEST) для подачі логічної «1» на всі виходи одночасно:
 - \circ Якщо *TEST=о* то автомат перемикає сигнали на виходах згідно заданого алгоритму.
 - Якщо TEST=1 то на всіх виходах повинна бути логічна «1» (всі LED увімкнені).
- Для керування сигналом MODE використати будь який з 8 DIP перемикачів (див. Додаток – 1).
- Для керування сигналами *RESET/TEST* використати будь якI : PUSH BUTTON кнопок (див. **Додаток** 1).

Порядок виконання лабораторної роботи.

- 1. Інтерфейс пристрою та функціонал реалізувати згідно отриманого варіанту завдання.
- 2. Логіку переходів реалізувати з використанням мови опису апаратних засобів.
- 3. Логіку формування вихідних сигналів реалізувати з використанням мови опису апаратних засобів.
- 4. Згенерувати символи для описів логіки переходів та логіки формування вихідних сигналів.
- 5. Зінтегрувати всі компоненти логіку переходів логіку формування вихідних сигналів та пам'ять станів в єдину систему. Пам'ять станів реалізувати за допомогою графічних компонентів з бібліотеки.
- 6. Промоделювати роботу окремих частин автомата та автомата вцілому за допомогою симулятора ISim.
- 7. Інтегрувати створений автомат зі стендом додати подільник частоти для вхідного тактовового сигналу призначити фізичні виводи на FPGA.
- 8. Згенерувати файал та перевірити роботу за допомогою стенда Elbert V2 Spartan3A FPGA.
- 9. Підготувати і захистити звіт.

Виконання лабораторної роботи:

MODE	CURR_STATE(2)	CURR_STATE(1)	CURR_STATE(0)	NEXT_STATE(2)	NEXT_STATE(1)	NEXT_STATE(0)
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	1	1
0	1	0	0	1	0	0
0	1	0	1	1	0	1
0	1	1	0	1	1	0
0	1	1	1	1	1	1
1	0	0	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	0	0	1
1	0	1	1	0	1	0
1	1	0	0	0	1	1
1	1	0	1	1	0	0
1	1	1	0	1	0	1
1	1	1	1	1	1	0

Рис.1.1 (Логіка переходів для всіх станів автомата)

Логіка переходів на мові VHDL

 $NEXT_STATE(0) = not(CURR_STATE(0));$

NEXT_STATE(1) = ((not(MODE) and not(CURR_STATE(1)) and CURR_STATE(0)) or (not(MODE) and CURR_STATE(1) and not(CURR_STATE(0))) or (MODE and not(CURR_STATE(1)) and not(CURR_STATE(0))) or (MODE and CURR_STATE(1)) and CURR_STATE(1));

 $NEXT_STATE(2) <= ((not(MODE) \ and \ CURR_STATE(2) \ and \ not(CURR_STATE(1))) \ or \ (CURR_STATE(2) \ and \ CURR_STATE(1) \ and \ not(CURR_STATE(0))) \ or \ (MODE \ and \ not(CURR_STATE(2)) \ and \ CURR_STATE(2)) \ and \ CURR_STATE(1) \ and \ CURR_STATE(0)) \ or \ (MODE \ and \ not(CURR_STATE(2)) \ and \ not(CURR_STATE(1)) \ and \ not(CURR_STATE(0))));$

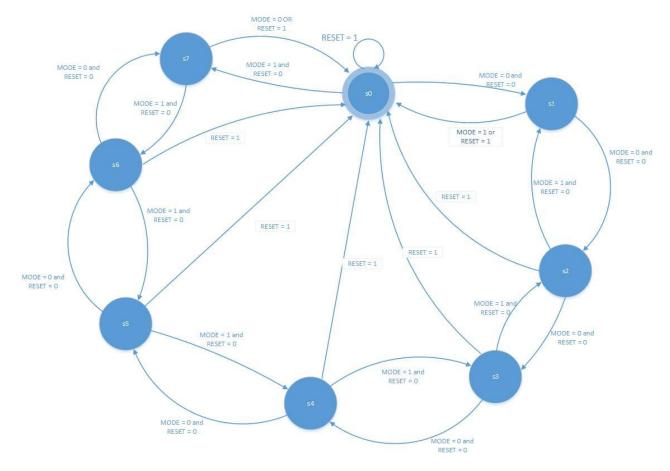


Рис.. (Граф переходів автомата між станами)

Логіка формування вихідних сигналів

```
OUT_BUS(0) <= ((not (IN_BUS(2)) and not (IN_BUS(1)) and not (IN_BUS(0))) or TEST) after 1 ns;

OUT_BUS(1) <= ((not (IN_BUS(2)) and IN_BUS(1) and not (IN_BUS(0))) or TEST) after 1 ns;

OUT_BUS(2) <= ((IN_BUS(2)) and not (IN_BUS(1)) and not (IN_BUS(0))) or TEST) after 1
```

ns;

```
OUT\_BUS(3) \mathrel{<=} ((IN\_BUS(2) \text{ and } IN\_BUS(1) \text{ and not } (IN\_BUS(0))) \text{ or } TEST) \text{ after } 1 \text{ ns};
```

OUT_BUS(4) <= ((IN_BUS(2) and IN_BUS(1) and IN_BUS(0)) or TEST) after 1 ns;

 $OUT_BUS(5) \le ((IN_BUS(2) \text{ and } not(IN_BUS(1)) \text{ and } IN_BUS(0)) \text{ or } TEST) \text{ after } 1 \text{ ns};$

OUT_BUS(6) <= ((not(IN_BUS(2)) and IN_BUS(1) and IN_BUS(0)) or TEST) after 1 ns;

OUT_BUS(7) <= ((not(IN_BUS(2)) and not(IN_BUS(1)) and IN_BUS(0)) or TEST) after 1 ns;

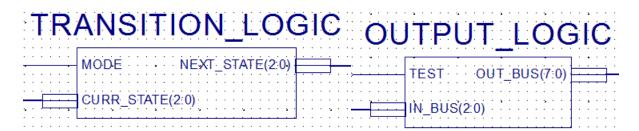


Рис.1.2 (Згенеровані схематичні схеми)

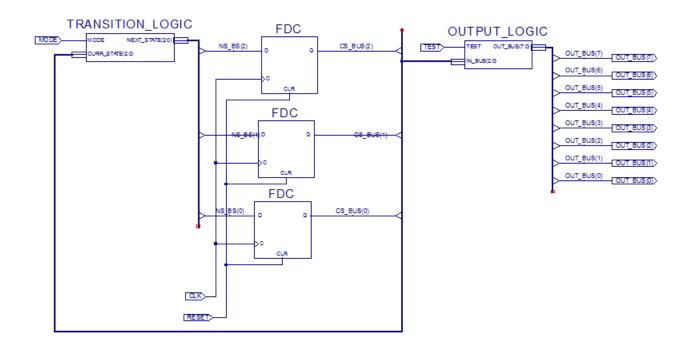


Рис.1.3 (Інтеграція всіх створених компонентів разом з пам'ятю станів автомата)

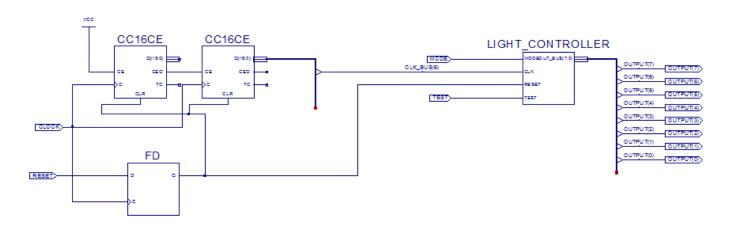
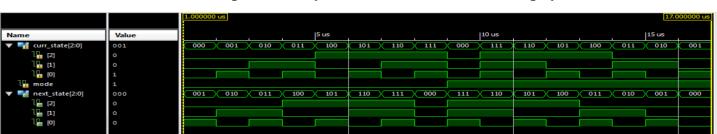


Рис.1.4 (Автомат світлових сигналів та подільник тактового сигналу)



Демонстрація симуляції схем наведених зверху

Рис.1.5 (Результати симуляції логіки переходів в ISim)

		0.000000 us								
Name	Value	0 us		2 us						8 us
▼ <table-of-contents> in_bus[2:0]</table-of-contents>	000	000	001	010	011	100	101	110	111	
1 [2]	0									
II [1]	0									
1 <mark>1.</mark> [0]	0									
ी₃ test	0									
🗸 🚟 out_bus[7:0]	00000001	00000001	10000000	00000010	01000000	00000100	00100000	00001000	00010000	
T _b [7]	0									
l [6]	0									
T [5]	0									
[4]	0									
[3]	0									
la [2]	0									
Ta [1]	0									
l [o]	1									
		X1: 0.000000 us								

Рис.1.6 (Результати симуляції логіки вихідних сигналів в ISim(TEST = 0))

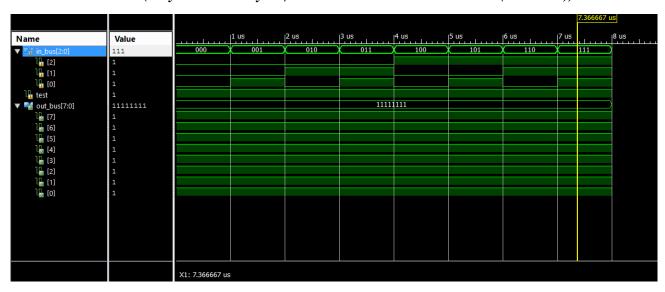
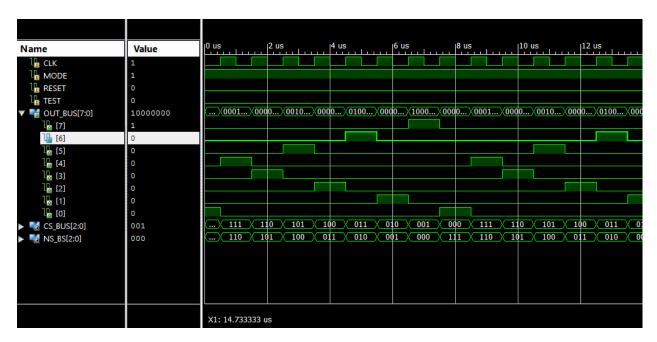


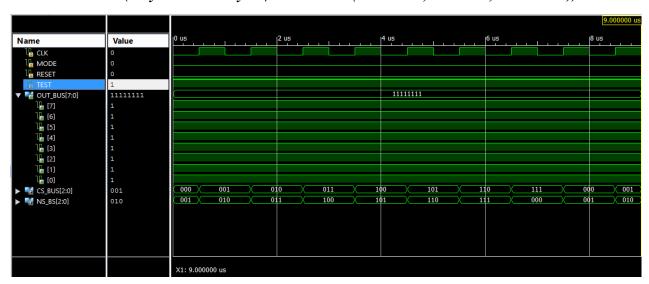
Рис.1.7 (Результати симуляції логіки вихідних сигналів в ISim(TEST = 1))



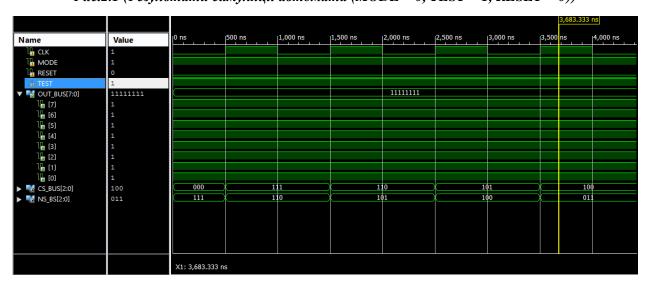
Puc.1.8 (Результати симуляції автомата (MODE = 0, TEST = 0, RESET = 0))



Puc.1.9 (Результати симуляції автомата (MODE = 1, TEST = 0, RESET = 0))



Puc.2.1 (Peзультати симуляції автомата (<math>MODE = 0, TEST = 1, RESET = 0))



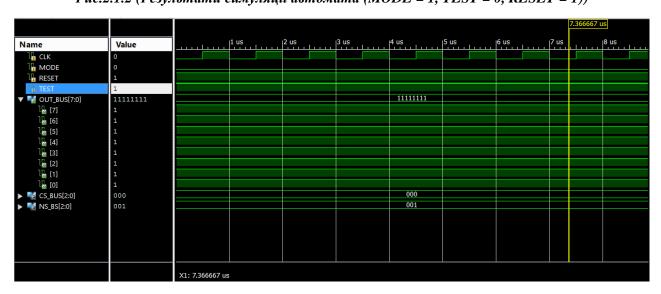
Puc.2.1 (Результати симуляції автомата (MODE = 1, TEST = 1, RESET = 0))

										3,683.333	ns
Name	Value	0 ns	500 ns	1,000 ns	1,500 ns	2,000 ns	2,500 ns	3,000 ns	3,500	ns	4,000 ns
₽ CLK	1										
™ MODE	0										
RESET	1										
l⊞ TEST	0										
▼ 🚟 OUT_BUS[7:0]	00000001	 				00000001					
l <u>a</u> [7]	0										
[6]	0										
[5]	0										
[4]	0										
[3]	0					 					
la [2]	0										
1⅓ [1] 1⅓ [0]	0										
► 5 CS_BUS[2:0]	000					000					
▶ ■6 NS_BS[2:0]	001					001					
₽ 145_55[2.0]	001					***					
		X1: 3,683.333 r	ıs								

Puc.2.1.1 (Результати симуляції автомата (MODE = 0, TEST = 0, RESET = 1))

										3,683.333	ns
Name	Value	0 ns	500 ns	1,000 ns	1,500 ns	2,000 ns	2,500 ns	3,000 ns	3,500	ns	4,000 ns
¼ CLK	1										
™ MODE	1										
¼ RESET	1										
TEST	0										
▼ 🛂 OUT_BUS[7:0]	00000001					00000001					
T [7]	0										
[6]	0	-									
1 [5]	0				 						
[4]	0										
l _m [3]	0										
다 [2] 다 [1]	0	1									
lm [0]	0										
► ■ CS_BUS[2:0]	000					000					
► NS_BS[2:0]	111					111					
► 6 N3_B3[2.0]	1	<u> </u>									
		X1: 3,683.333	ns								

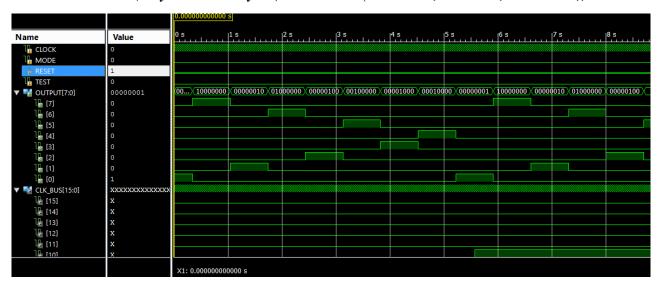
Puc.2.1.2 (Результати симуляції автомата (MODE = 1, TEST = 0, RESET = 1))



Puc.2.1.3 (Результати симуляції автомата (MODE = 0, TEST = 1, RESET = 1))

					_					3,683.333	ns
Name	Value	0 ns	500 ns	1,000 ns	1,500 ns	2,000 ns	2,500 ns	3,000 ns	3,500	ns	4,000 ns
Ū CLK	1										
™ MODE	1										
☐ RESET	1										
TEST	1										
▼ 📑 OUT_BUS[7:0]	11111111					11111111					
la [7]	1										
la [6]	1										
T [5]	1										
[4]	1										
[3]	1										
l [2]	1										
¼ [1]	1										
l a [0]	1					200					
► 1 CS_BUS[2:0]	000	<u> </u>				000					
▶ 📆 NS_BS[2:0]	111					111					
		X1: 3,683.333	ns								

Puc.2.1.4 (Результати симуляції автомата (MODE = 1, TEST = 1, RESET = 1))



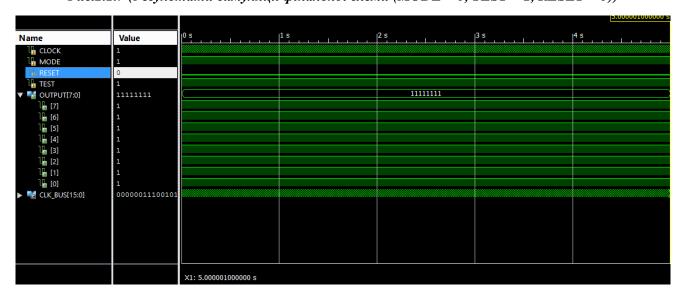
Puc.2.1.5 (Результати симуляції фінальної схеми (MODE = 0, TEST = 0, RESET = 0))



Puc.2.1.6 (Результати симуляції фінальної схеми (MODE = 1, TEST = 0, RESET = 0))

						5.0000010000000 s
Name	Value	0 s	1 s	2 s	3 s	4 s
¹₽ crock	1					
1 MODE	0					
RESET	0					
TEST	1					
▼ 🛂 OUTPUT[7:0]	11111111			11111111		
la [7]	1					
[6]	1					
[5]	1					
[4]	1					
I _a [3]	1					
[2]	1					
l [1]	1					
l [0]	1					
▼ N CLK_BUS[15:0]	00000011100101					
15]	0					
Vo [14]	0					
V _e [13]	0					
Va [12]	0					
Va [11]	0					
1101	0					
		X1: 5.000001000000 s				

Рис.2.1.7 (Результати симуляції фінальної схеми (MODE = 0, TEST = 1, RESET = 0))



Puc.2.1.7 (Результати симуляції фінальної схеми (MODE = 1, TEST = 1, RESET = 0))

Name	Value	0 s	1s	2 s	3 s	4 s
¹₽ clocк	1					
¹₽ MODE	0					
₽ RESET	1					
TEST	0					
▼ 📑 OUTPUT[7:0]	00000001	(0000001		
l [7]	0					
l [6]	0					
l [5]	0					
l _m [4]	0					
l _m [3]	0					
l [2]	0					
l [1]	0					
l∰ [o]	1					
▼ N CLK_BUS[15:0]	000000000000000			0000000000000000		
V ₀ [15]	0					
Va [14]	0					
V ₀ [13]	0					
Va [12]	0					
Va [11]	0					
10. [10]	0					
		X1: 5.000000000000 s				

Puc.2.1.7 (Результати симуляції фінальної схеми (MODE = 0, TEST = 0, RESET = 1))

Name	Value	0 s	1 s	2 s	3 s	4 s	5 s	6 s	7 s	8 s
¹₽ clock	0									
¼ MODE	1									
୍ଲା RESET	1									
児 TEST	0									
▼ № OUTPUT[7:0]	00000001	<u> </u>				00000001				
l [7]	0									
[6]	0									
[5]	0									
[4]	0									
[3]	0									
[2]	0									
la [1]	0									
[0]	1									
▼ No CLK_BUS[15:0]	xxxxxxxxxxxx				00	000000000000000000000000000000000000000				
	X									
1000	X X									
	X X									
	x									
	x									
		X1: 0.000000000	0000 s							

Puc.2.1.7 (Результати симуляції фінальної схеми (MODE = 1, TEST = 0, RESET = 1))

						10.0000000000000 s
Name	Value	0 s	2 s	4 s	6 s	8 s
¹₽ clock	1					
₩ MODE	0					
¹₽ RESET	1					
TEST	1					
▼ 1 OUTPUT[7:0]	11111111			11111111		
1 [7]	1					
Tag [6]	1					
Tag [5]	1					
[4]	1					
Ta [3]	1					
[2]	1					
¼ [1]	1					
¼ [o]	1					
▼ 1 CLK_BUS[15:0]	000000000000000			0000000000000000		
V _e [15]	0					
Va [14]	0					
Va [13]	0					
Va [12]	0					
Va [11]	0					
101	0					
		X1: 10.000000000000 s				

Puc.2.1.8 (Результати симуляції фінальної схеми (MODE = 0, TEST = 1, RESET = 1))

						10.0000000000000000 s
		·0 =		.d =		.0 -
Name	Value	0 s	2 s	4 s	6 s	85
$\mathbb{I}_{\overline{a}}$ сьоск	1					
I MODE	1					
¹₽ RESET	1					
¼ TEST	1					
▼ 🛂 OUTPUT[7:0]	11111111			11111111		
¼ [7]	1					
T _m [6]	1					
l _m [5]	1					
[4]	1					
l _m [3]	1					
l _m [2]	1					
l [1]	1					
¼ [o]	1					
▼ N CLK_BUS[15:0]	000000000000000	(000000000000000		
V _e [15]	0					
Va [14]	0					
16 [13]	0					
16 [12]	0					
Va [11]	0					
101	n					
		X1: 10.000000000000 s				

Puc.2.1.9 (Результати симуляції фінальної схеми (MODE = 1, TEST = 1, RESET = 1))

Виконання Test Bench

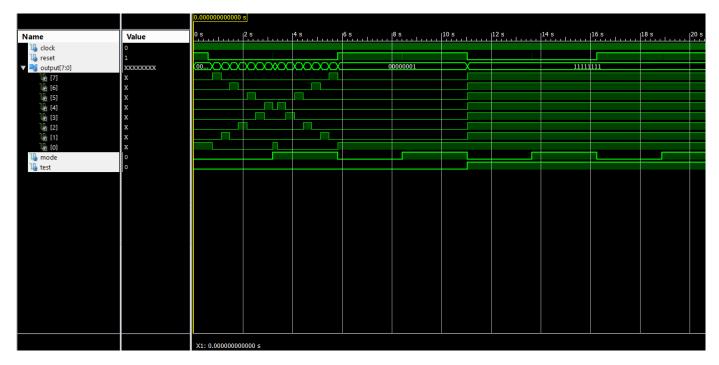


Рис.2.2 (Часова діаграма)

ISim> # run 30s ISim>

Рис. 2.3 (Консоль під час тестування)

```
UCF for ElbertV2 Development Board
   CONFIG VCCAUX = "3.3";
   # Clock 12 MHz
6
                         LOC = P129 | IOSTANDARD = LVCMOS33 | PERIOD = 12MHz;
  10
   12
                          LOC = P46 | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
13
      NET "OUTPUT (1) "
                          LOC = P47
                                  | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
     NET "OUTPUT(2)"
                          LOC = P48 | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
LOC = P49 | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
     NET "OUTPUT (3) "
16
     NET "OUTPUT (4)"
                                  | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
17
     NET "OUTPUT(5)"
NET "OUTPUT(6)"
                          LOC = P51 | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
LOC = P54 | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
19
     NET "OUTPUT (7) "
                          LOC = P55
                                  | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
20
  23
                             DP Switches
   .......
24
                  LOC = P70 | PULLUP | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
26
27
      29
                             Switches
   30
      NET "RESET"
                    LOC = P80 | PULLUP | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
LOC = P79 | PULLUP | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 12;
      NET "TEST"
33
```

Рис.2.2 (Призначення фізичних входів та виходів)

Висновок: В ході виконання цієї лабораторної роботи я реалізував на базі стенда Elbert V2 – Spartan3A FPGA цифровий автомат світлових ефектів згідно з заданими вимогами.