

Maciej Byczko Bartosz Matysiak	Prowadzący: dr inż. Jacek Mazurkiewicz	Numer ćwiczenia 4
PN 10:50 TP	Temat ćwiczenia: Układy wielobitowych wejść i wyjść	Ocena:
Grupa: B	Data wykonania: 8 Listopada 2021r.	

Spis treści

1	Zadanie 1	3
1.1	Polecenie	3
1.2	Rozwiązanie	3
1.2.1	Schemat stanów	3
1.2.2	Schemat układu	3
1.2.3	Kod VHDL	3
1.2.4	Symulacja	3
1.3	Fizyczna implementacja	3
1.3.1	Kod UCF	3
2	Zadanie 2	3
2.1	Polecenie	3
2.2	Rozwiązanie	4
2.2.1	Schemat stanów	4
2.2.2	Tabela prawdy	4
2.2.3	Siatki Karnaugh	4
2.2.4	Schemat układu	4
2.2.5	Kod VHDL	4
2.2.6	Symulacja	4
2.3	Fizyczna implementacja	4
2.3.1	Kod UCF	4
3	Zadanie 3	4
3.1	Polecenie	4
3.2	Rozwiązanie	4
3.2.1	Schemat stanów	4
3.2.2	Tabela prawdy	4
3.2.3	Siatki Karnaugh	4
3.2.4	Schemat układu	4
3.2.5	Kod VHDL	4
3.2.6	Symulacja	4
3.3	Fizyczna implementacja	4
3.3.1	Kod UCF	4
4	Zadanie 4	4
4.1	Polecenie	4
4.2	Rozwiązanie	5
4.2.1	Schemat stanów	5
4.2.2	Tabela prawdy	5
4.2.3	Siatki Karnaugh	5

4.2.4	Schemat układu	5
4.2.5	Kod VHDL	5
4.2.6	Symulacja	5
4.3	Fizyczna implementacja	5
4.3.1	Kod UCF	5
5	Wnioski	5

1 Zadanie 1

1.1 Polecenie

Detektor 2-znakowej sekwencji słów 8-bitowych: wejścia 2 znaków 8-bitowych, 1 wyjście 1-bitowe – sekwencja rozpoznana / sekwencja błędna. Źródło danych: początkowo "guziki" przystawki, potem klawiatura PC via terminal.

1.2 Rozwiązanie

Do rozwiązania problemu wymagane jest od nas podłączenie dwóch komparatorów 8-bitowych (COMP8), które po pobraniu wartości od użytkownika kolejno po sobie sprawdzają wprowadzone słowa. W zależności od wymagania można wprowadzić także obowiązkowe wprowadzanie wartości w odpowiedniej kolejności.

1.2.1 Schemat stanów

1.2.2 Schemat układu

1.2.3 Kod VHDL

1.2.4 Symulacja

1.3 Fizyczna implementacja

1.3.1 Kod UCF

2 Zadanie 2

2.1 Polecenie

Układ arytmetyczny pracujący na dwóch argumentach 4-bitowych wyrażonych w kodzie Aikena i generujący stosowny wynik w tymże kodzie.

2.2 Rozwiązanie

2.2.1 Schemat stanów

2.2.2 Tabela prawdy

2.2.3 Siatki Karnaugh

2.2.4 Schemat układu

2.2.5 Kod VHDL

2.2.6 Symulacja

2.3 Fizyczna implementacja

2.3.1 Kod UCF

3 Zadanie 3

3.1 Polecenie

Konwerter cyfry szesnastkowej zapisanej na czterech bitach od 0 do 9, A do F na kod ASCII tej cyfry – wyjście 8-bitowe. Prezentacja wyniku na diodach przystawki, potem na wyświetlaczu 7-segmentowym.

3.2 Rozwiązanie

3.2.1 Schemat stanów

3.2.2 Tabela prawdy

3.2.3 Siatki Karnaugh

3.2.4 Schemat układu

3.2.5 Kod VHDL

3.2.6 Symulacja

3.3 Fizyczna implementacja

3.3.1 Kod UCF

4 Zadanie 4

4.1 Polecenie

Komparator dwóch 4-bitowych cyfr: 2 wejścia po 4 bity, 3 wyjścia 1-bitowe: mniejszy, większy, równy pracujący w kodzie Aikena.

4.2 Rozwiązanie

4.2.1 Schemat stanów

4.2.2 Tabela prawdy

4.2.3 Siatki Karnaugh

4.2.4 Schemat układu

4.2.5 Kod VHDL

4.2.6 Symulacja

4.3 Fizyczna implementacja

4.3.1 Kod UCF

5 Wnioski