

LABORATORIUM - ELEKTRONIKA			
Projektowanie koderów, transkoderów i dekoderów w języku VHDL			
wykonali:	data:	podpis prowadzącego:	ocena:

Tabela 1.
Transkoder 4-bitowego kodu binarnego na kod wyświetlacza 7-segmentowego

dziesiętnie	wejścia				wyjścia						
	W3	W2	W1	W0	g	f	e	d	c	b	a
0	0	0	0	0	0	1	1	1	1	1	1
1	0	0	0	1							
2	0	0	1	0							
3	0	0	1	1							
4	0	1	0	0							
5	0	1	0	1							
6	0	1	1	0							
7	0	1	1	1							
8	1	0	0	0							
9	1	0	0	1							
10	1	0	1	0							
11	1	0	1	1							
12	1	1	0	0							
13	1	1	0	1							
14	1	1	1	0							
15	1	1	1	1							

WSKAZÓWKI DO PROJEKTU UKŁADU MNOŻĄCEGO 2 LICZBY 2-bitowe.

Projektując taki układ należy na początek zadać sobie 2 podstawowe pytania, czyli:

- ile taki układ powinien mieć wejść?
- ile taki układ powinien mieć wyjść?

Na pierwsze pytanie odpowiedź jest prosta.

DWIE liczby po DWA bity, więc w sumie potrzebujemy CZTERECH bitów wejścia.

Drugie pytanie też jest proste, choć wymaga chwili zastanowienia.

Liczba wyjść takiego układu zależy od tego, ile maksymalnie bitów zajmie nam wynik operacji. Operujemy na liczbach 2-bitowych. Maksymalną wartością do zapisania na 2 bitach jest 11, czyli dziesiętna 3. Skoro mnożymy 2 takie liczby przez siebie, to maksymalnym wynikiem jest $3 \cdot 3 = 9$. Skoro 9 zapisana binarnie to 1001, to potrzebujemy 4 bitów by ją zapisać, więc właśnie 4 bity ma wyjście projektowanego układu, a w konsekwencji tabela stanów projektowanego układu wygląda jak niżej. Wynik opisanej przed chwilą operacji zawiera ostatni (niebieski) wiersz tabeli, który zawiera w sobie informację że $11 \cdot 11 = 1001$. Inny przykład zawiera zielony wiersz: $10 \cdot 00 = 0000$, który zapisany dziesiętnie wyraża sobą równość $2 \cdot 0 = 0$.

Po zapoznaniu się z powyższą wiedzą, wypełnienie całości tabeli nie powinno sprawić większych trudności.

Tabela 2.
Układ mnożący dwie liczby 2-bitowe

wejścia				wyjścia			
A1	A0	B1	B0	w	x	y	z
0	0	0	0				
0	0	0	1				
0	0	1	0				
0	0	1	1				
0	1	0	0				
0	1	0	1				
0	1	1	0				
0	1	1	1				
1	0	0	0	0	0	0	0
1	0	0	1				
1	0	1	0				
1	0	1	1				
1	1	0	0				
1	1	0	1				
1	1	1	0				
1	1	1	1	1	0	0	1