



# Politechnika Wroclawska

## **Laboratorium Podstaw Techniki Cyfrowej i Mikroprocesorowej**

### **Ćwiczenie 2 Cyfrowe układy arytmetyczne**

#### **Zagadnienia do przygotowania**

- Wiadomości takie jak do ćwiczenia 1
- Kodowanie binarne wartości liczbowych:
  - kod naturalny binarny
  - kod stałoprzecinkowy
  - kod U2
- Zasady działania podstawowych układów arytmetycznych
- Hazard w układach kombinacyjnych i sposoby jego usuwania

#### **Literatura**

- [1] Wykład
- [2] Andrzej Skorupski, „Podstawy techniki cyfrowej” WKiŁ 2001
- [3] Instrukcja użytkowania makiet dydaktycznych
- [4] Instrukcja wprowadzająca do programu LTSpice

## 1. Zadania

Na początku zajęć student otrzymuje od prowadzącego zadanie z pośród przedstawionych poniżej. Należy dla nich zaproponować odpowiednie tablice prawdy, zminimalizować opisywaną przez nie funkcje logiczne metodą tablic Karnaugh.

Otrzymane wyrażenie algebraiczne należy zmodyfikować korzystając z reguł algebry Boole'a tak, aby zastosowane były w nim operandy odpowiadające bramkom, które są dostępne w makiecie, czyli NAND, NOR, NOT, jak opisano w [3].

Do tabeli prawdy należy dopisać kolumny z wartościami, jakie będzie przyjmowały każdy z implikantów, jak to opisano w [3]. Ułatwi to proces uruchamiania układu na makiecie.

Układ należy zasymulować w programie LTspice[4] sprawdzając prawidłowość wykonania projektu.

Oba układy należy stopniowo zmontować i uruchomić go na makiecie [3]. Należy wykazać prawidłowość działania układu oraz określić czas propagacji sygnału przez układ cyfrowy.

Jeśli w wyniku operacji arytmetycznej generowany jest wynik nie mieszczący się w określonej zadaniem liczbie bitów wyjściowych należy zastosować obcinanie wyniku sygnalizowane na dodatkowym wyjściu układu oznaczanym C lub O. Np. liczba  $10011_b$  obcięta do 4 bitów to są 4 najmłodsze bity, czyli  $0011_b$ . Takie zachowanie jest często spotykane w układach cyfrowych.

Po przetestowaniu układu należy doświadczalnie sprawdzić, czy występuje w nim hazard. Układ należy zmodyfikować tak, aby hazard usunąć sprawdzić jego działanie.

### 1.1. Półsumator

Półsumator jest układem o 2 wejściach i 2 wyjściach. Na wejścia podaje się sumowane bity. Jedno z wyjść zwane S jest wynikiem sumowania na bieżącej pozycji, a drugie z wyjść wyjściem przeniesienia C.

Zaproponować funkcję logiczne realizowane przez wyjścia S i C.

Zrealizować układ w oparciu o: a) różne rodzaje bramek, b) jeden rodzaj bramek np. NAND lub NOR.

### 1.2. Sumator

Zaprojektować układ dodający dwa bity wejściowe  $x_0$ ,  $x_1$  oraz bit przeniesienia C. Na wyjściu układu powinien znaleźć się wynik dodawania  $y = x_0 + x_1 + C$  oraz przeniesienie wyjściowe  $C_0$ .

### 1.3. Układ odejmujący

Zaprojektować układ realizujący odejmowanie bitów wejściowych  $x_0$  i  $x_1$  oraz bitu pożyczki B. Na wyjściu układu powinien znaleźć się wynik odejmowania  $y = x_0 - x_1 - B$  oraz pożyczka wynikowa  $B_0$ .

### 1.4. Komparator binarny

Komparator binarny jest układem porównującym 2 liczby binarne. W przypadku, gdy są równe na wyjściu pojawia się stan aktywny. Zaprojektować układ porównujący 2 bitowe liczby binarne. Sprawdzić poprawność realizacji. W jaki sposób można rozszerzyć układ do n bitów?

### 1.5. Komparator liczb dwubitowych.

Zaprojektujemy komparator realizujący funkcję  $A > B$  ( jeśli  $A > B$  – dostajemy na wyjściu 1, w przeciwnym przypadku – otrzymujemy 0 ). Dwubitowe liczby A i B będą przedstawione w postaci  $A = A_2A_1$  i  $B = B_2B_1$  (  $A_2, A_1, B_2, B_1$  są wejściami układu).

### 1.6. Układ kontroli parzystości

Zaprojektować układ generujący bit parzystości dla 4 bitowej liczby binarnej, oraz układ realizujący funkcję odwrotną czyli sprawdzający poprawność odebranej liczby 4 bitowej na podstawie bitu parzystości. Zaproponować układ realizujący tę funkcję, oraz sprawdzić jego poprawność.

### 1.7. Układ służący do podnoszenia do kwadratu liczb od 0 do 7

Rozważany układ podnosi do kwadratu liczby od 0 do 7 zapisane w naturalnym kodzie dwójkowym i zwraca wynik na 6-bitowe wyjście również w naturalnej postaci dwójkowej.

### 1.8. Układ obliczający pierwiastek kwadratowy z liczby od 0 do 15.

Rozważmy układ służący do obliczenia pierwiastka z liczby od 0 do 15 podanej w naturalnym czterobitowym kodzie dwójkowym.

Wyjście składa się z trzech bitów : na dwóch ( $O_1, O_2$ ) zapisujemy wynik pierwiastkowania jeśli jest liczbą naturalną; w przeciwnym wypadku zwracany jest błąd – stan '1' na trzecim bicie wyjścia ( $E$ ) i stan '0' na pozostałych bitach wyjścia.

### 1.9. Układ sprawdzający czy liczba od 0 do 15 jest podzielna przez 3.

Na wejście tego układu podajemy liczbę zapisaną w naturalnym kodzie dwójkowym – na wyjściu otrzymujemy 1 bądź 0 w zależności czy liczba jest (1) czy też nie jest (0) podzielna przez 3.

### 1.10. Układ zwiększający n-bitową liczbę binarną

Zaprojektować układ, który zwiększa o 1 liczbę binarną. Dla ustalenia uwagi układ otrzymuje na wejście 4 bitową liczbę. Na wyjściach powinna pojawić się liczba zwiększona o 1.

Układ posiada 4 wejścia i 4 wyjścia. Dla każdego wyjścia należy zapisać funkcję logiczną, zminimalizować ją i zrealizować na dowolnych bramkach.

### 1.11. Układ arytmetyczny

Zaprojektować układ arytmetyczny operujący na dwu dwubitowych słowach wejściowych A i B, wykonujący operację:

$$Y = A * B + \text{const}$$

$$Y = \text{const} * A + B$$

$$Y = \text{const1} * A + \text{const2} * B$$

lub podobne. Układ powinien posiadać 4-bitowe słowo wyjściowe. Stałe const z zakresu 1-6 podaje prowadzący podczas ćwiczenia.

## 2. Realizacja ćwiczenia

Realizując zadanie należy wykonać wszystkie czynności opisane w poniższej

tabeli:

<i>Nr</i>	<i>Czynność</i>	<i>Pkt.</i>	<i>Sprawozdanie</i>
1	Zapisanie prawidłowej tabeli prawdy	1	tak
2	Minimalizacja funkcji boolowskiej	2	tak
3	Przekształcenie funkcji boolowskiej aby była możliwa do zrealizowania przy użyciu dostępnych bramek	1	tak
4	Rozszerzenie tabeli prawdy o wartości implikantów	1	tak
5	Zestawienie układu w programie LTSpice	2	do wglądu przez prowadzącego
6	Analiza wyników symulacji	3	
7	Zestawienie układu na makiecie dydaktycznej	1	do wglądu przez prowadzącego
8	Uruchomienie układu na makiecie dydaktycznej	2	
9	Analiza wyników działania układu na makiecie dydaktycznej	3	
10	Sprawdzenie występowania hazardu	2	do wglądu
11	Wyeliminowanie hazardu	1	tak
12	Sprawdzenie działania układu z wyeliminowanym hazardem	1	do wglądu

<i>Skala ocen</i>					
<i>Suma punktów za zadania 1 i 2</i>	19-20	17-18	14-16	12-14	10-11
<i>Ocena</i>	5	4,5	4	3,5	3