

Architektury systemów komputerowych

Lista 6

$x_6 = 8$ (minimum na bdb)

1. (a) Jak zmieni się wartość liczby w reprezentacji uzupełnień do 2 jeśli z prawej strony dopiszemy do niej zero?
(b) Załóżmy, że 16-bitową liczbę w reprezentacji uzupełnień do 2 chcemy zapisać na 32 bitach. Jak to zrobić?
2. Zaprojektuj układ kombinacyjny, który konwertuje liczbę czterobitową w reprezentacji znak-moduł (pierwszy bit to bit znaku: 0-plus, 1-minus, następne trzy bity to wartość bezwzględna liczby zapisana w naturalnym kodzie binarnym) na reprezentację uzupełnień do dwóch.
3. W tym zadaniu rozważamy liczby dodatnie reprezentowane w naturalnym kodzie binarnym. Do dyspozycji masz układy dwóch rodzajów:
 - układ przyjmujący na wejściu dwie liczby dwubitowe i zwracający ich czterobitowy iloczyn
 - układ sumatora otrzymujący dwie dwubitowe liczby i dodatkowo wstępne przeniesienie, zwracający dwubitową sumę oraz bit przeniesienia.

Używając pewnej liczby układów powyższego typu skonstruuj układ przyjmujący na wejściu dwie liczby czterobitowe i zwracający ich ośmiobitowy iloczyn. Nie wolno używać żadnych dodatkowych bramek logicznych.

4. Zasymuluj działanie algorytmu Booth'a na następujących przykładach:
 - (a) $0101 \cdot 1001$
 - (b) $01110 \cdot 01000$
 - (c) $10100 \cdot 11111$
 - (d) $11100011 \cdot 00111100$
5. Znajdź opis jakiegoś algorytmu maszynowego dzielenia całkowitego liczb n -bitowych w naturalnym kodzie binarnym, dla którego liczba wykonywanych operacji jest rzędu n^2 . Omów jego implementację sprzętową i przedstaw działanie na przykładach.
6. Na wykładzie przedstawiłem dwa algorytmy mnożenia maszynowego: dla liczb bez znaku w naturalnym kodzie binarnym (notatki do wykładu 5b, rys. 1) i dla liczb w reprezentacji uzupełnień do 2 – alg. Booth'a (notatki do wykładu 6, rys. 1). Przypomnij ich działanie i wytłumacz dlaczego w drugim z nich nie potrzebujemy rejestru C , pamiętającego przeniesienie z ostatniego sumatora.
7. Ile operacji dodawań/odejmowań wykona pierwszy z algorytmów z poprzedniego zadania, a ile drugi, w przypadku, gdy mnożymy przez następującą liczbę o długości n bitów:
 - (a) 00000000...00000000
 - (b) 11111111...11111111
 - (c) 01010101...01010101
 - (d) 00110011...00110011
 - (e) 10000000...00000000
 - (f) 01111111...11111111

8. Przypomnij pomysł „podglądu przeniesienia” (carry-lookahead) budując według tej zasady sumator 4 bitowy. Używając go jako „bloku” zbuduj sumator 16 bitowy. Zakładając, że każda bramka logiczna powoduje opóźnienie 1 (z wyjątkiem bramki NOT, której opóźnienie można pominąć) wylicz całkowite opóźnienie przedstawionego układu.
9. Jak w zadaniu poprzednim, ale dla sumatora zbudowanego zgodnie z pomysłem „wyboru przeniesienia” (carry select). Zakładamy, że kolejne bloki (patrząc od prawej) mają długość 1,2,3,4,6 bitów.
10. Załóżmy, że stosując ideę carry select budujemy sumatory n -bitowy. Oszacuj czas działania takiego sumatora (funkcja zmiennej n), zakładając, że startujemy od bloku 1-elementowego, a każdy następny blok jest o jeden dłuższy.
11. Załóżmy teraz, że stosujemy ideę carry-select, ale chcemy używać bloków jednakowej długości. Jaka jest najlepsza długość bloku dla sumatora n -bitowego? Oszacuj czas działania całego układu w tym przypadku.

Emanuel Kieroński