

Architektury systemów komputerowych

Lista 10

$x_{10} = 7$ (minimum na bdb)

1. Rozważmy procesor jednocyklowy ze slajdu 16, wykład 10. Załóżmy, że czasy działania bloków I-Mem, Add, Mux, ALU, Regs, D-mem i Control (jednostki sterującej) wynoszą odpowiednio 400ps, 100ps, 30ps, 120ps, 200ps, 350ps i 100ps., a ich ceny 1000, 30, 10, 100, 200, 2000 i 500 (pomijamy czasy działania i koszty układów Sign Extend, Shift Left oraz ALU control). Rozważmy teraz dwa usprawnienia procesora:

- zastąpienie układów Add szybszymi (przyspieszenie o 20ps, wzrost ceny o 20)
- użycie większego pliku rejestrów Regs (zmodyfikowany układ Regs działa o 100ps wolniej, cena wzrasta o 200, ale zmniejsza się liczba odwołań do pamięci, co zmniejsza całkowitą liczbę wykonywanych rozkazów o 5 procent).

Na poniższe pytania odpowiedz rozważając warianty, w których obie modyfikacje wprowadzane są jednocześnie oraz każda z osobna.

- (a) Jaka jest długość cyklu zegarowego bez oraz z powyższymi modyfikacjami?
 - (b) Jakie przyspieszenie uzyskuje się dzięki wprowadzonym modyfikacjom?
 - (c) Porównaj stosunek ceny do efektywności procesora bez modyfikacji i z modyfikacjami.
2. Rozważmy następujące rozkazy: `lw $1, 40 ($6)` oraz `Label: bne $1, $2, Label`.
- (a) Jak zakodowane są powyższe rozkazy w kodzie maszynowym?
 - (b) Ponownie rozważamy procesor ze slajdu 16, wykład 10. Jakie wartości zostaną podane na wejścia *Read register 1*, *Read register 2* oraz *Write register* podczas wykonywania powyższych rozkazów?
3. Które z rozkazów rozważanych na wykładzie 10 będą działały poprawnie na procesorze jednocyklowym ze slajdu 16, wykład 10, jeśli pojawi się błąd ustawiający na stałe wartość jednego z sygnałów sterujących. Rozważamy następujące sytuacje:
- (a) `RegWrite=1` (`RegWrite` to sygnał zezwolenia na zapis rejestru)
 - (b) `Branch=1` (`Branch` to informacja o tym, czy rozkaz jest skokiem warunkowym)
 - (c) `MemRead=1` (`MemRead` to informacja o tym, że pamięć będzie odczytywana)
 - (d) `MemWrite=1` (`MemWrite` to sygnał zezwolenia na zapis do pamięci)
 - (e) `RegWrite=0`
 - (f) `Branch=0`
 - (g) `MemRead=0`
 - (h) `MemWrite=0`
4. Zaprojektuj na poziomie bramek logicznych układ Add Control ze slajdu 16, wykład 10. Przyjmij, że sygnały odpowiednich rozkazów dla ALU są następujące: AND – 0000, OR – 0001, add – 0010, sub – 0110, set on less then – 0111, NOR – 1100. Zakładamy, że mamy listę rozkazów ze slajdu 2, wykład 10. Wartości dwubitowego sygnału ALUop są następujące: 00 – wykryto rozkaz `lw/sw`, 01 – wykryto rozkaz `beq`, 10 – wykryto rozkaz arytmetyczno-logiczny/`slt`.
5. Rozważmy dodanie rozkazu `jr` (skocz pod adres zapisany w rejestrze o numerze podanym w argumencie rozkazu) do implementacji jednocyklowej. Opisz konieczne modyfikacje i rozszerzenia układu, dorysuj na rysunku potrzebne dodatkowe połączenia i układy.

6. Odpowiedz na pytanie z zadania 3, ale tym razem dla implementacji wielocyklowej ze slajdu 24, wykład 10 i następujących wartości sygnałów:
- (a) RegWrite=1
 - (b) MemRead=1
 - (c) MemWrite=1
 - (d) IRWrite=1 (zezwolenie na zapis do rejestru rozkazów)
 - (e) PCWrite=1 (zezwolenie na zapis do licznika rozkazów)
 - (f) PCWriteCond=1 (j.w., uaktywniane dla skoków warunkowych)
 - (g) RegWrite=0
 - (h) MemRead=0
 - (i) MemWrite=0
 - (j) IrWrite=0
 - (k) PCWrite=0
 - (l) PCWriteCond=0
7. Rozszerz procesor wielocyklowy o rozkaz lui (załaduj stałą 16-bitową do dwóch górnych bajtów wskazanego rejestru). Opisz konieczne modyfikacje i rozszerzenia oraz zaznacz je na rysunku.
8. Rozważmy modyfikację procesora wielocyklowego, w której układ rejestrów (na rysunku oznaczony jako **Registers**) ma tylko jedno wyjście (**Read Data 1**). Opisz konieczne zmiany, po których procesor nadal będzie poprawnie realizował naszą listę rozkazów. Jak zmieni się automat opisujący działanie jednostki sterującej (slajd 31, wykład 10)?
9. Załóżmy, że MIPS ma instrukcję **bcmp**, która porównuje dwa bloki pamięci. Zakładamy, że adres pierwszego bloku podany jest w rejestrze **\$t1**, drugiego w **\$t2**, a długość bloków (w słowach, liczba naturalna) w **\$t3**. Rozkaz zapisuje wynik w **\$t1** – wynikiem jest adres pierwszej znalezionej różnicy lub zero, jeśli bloki są identyczne. Zakładamy również, że podczas wykonania rozkazu mogą być modyfikowane rejestry **\$t1-\$t5**.
- Napisz program w assemblerze MIPS emulujący opisany powyżej rozkaz. Ile cykli potrwa jego wykonanie dla bloków długości 100 na procesorze wielocyklowym?