

Wykład 9

Przyspieszanie pracy mikroprocesorów

dr hab. inż. Bartłomiej Zieliński, prof. PŚ.

Przyspieszanie mikroprocesorów

Program:

- Przyspieszenie czy optymalizacja?
- Przetwarzanie potokowe
- Przetwarzanie superskalarne
- Przewidywanie skoków
- Pamięć podręczna

Przyspieszanie mikroprocesorów

- Przyspieszanie czy optymalizacja?
 - Zwiększanie częstotliwości taktowania
 - Fizyczne ograniczenia częstotliwości w strukturze półprzewodnika
 - Rodzaje przetwarzanie równoległego
 - Potokowość
 - Superskalarność
 - Pełna równoległość
 - RISC/CISC/FISC

Przyspieszanie mikroprocesorów

- Typowe cechy procesorów RISC
 - Stały format rozkazu
 - Łatwiejsze dekodowanie
 - Pamięć tylko w rozkazach odczytu i zapisu
 - Dużo rejestrów (≥ 32)
 - Brak wyróżnionego akumulatora
 - Przekazywanie argumentów przez rejestry (nie stos)
 - Mała liczba rozkazów
 - Szybkie wykonywanie rozkazów (1 Tclk/rozk.)
 - Proste układy dekodująco-sterujące
 - (architektura Harvard)

Przyspieszanie mikroprocesorów

- Przetwarzanie potokowe
 - Rozkaz podzielony na etapy (fazy)
 - Np. pobranie, dekodowanie, adresowanie, wykonanie, zapis
 - *Optimum* ≈ 8 etapów?
 - Kilka rozkazów przetwarzanych równolegle
 - **Każdy rozkaz na innym etapie**
 - Problemy
 - Zmienny czas przetwarzania rozkazu na różnych etapach $\rightarrow$ kolejki
 - Zakłócenie przetwarzania $\rightarrow$ opróżnienie potoku $\rightarrow$ duże opóźnienia
 - Wzajemne zależności rozkazów i danych

Przyspieszanie mikroprocesorów

- Przetwarzanie superskalarne
 - $\geq$ potok wykonawczy
 - $\gg 1$ mało wydajne z powodu zależności
 - Potoki mogą mieć różne możliwości
 - Ciąg rozkazów podzielony na potoki
 - Zależnie od właściwości potoków
 - Potoki (a)synchroniczne
 - Syn: Potok 1 czeka $\rightarrow$ potok 2 także czeka, nawet gdy nie ma powodu
 - Asyn: Potok 1 czeka $\rightarrow$ potok 2 kontynuuje pracę
 - » Rozkaz 2 zakończy się przed rozkazem 1
 - **Przetwarzanie poza kolejnością**
(Out-of-order execution)

Przyspieszanie mikroprocesorów

- Zależności między operandami
 - Odczyt po odczycie (Read after Read)
 - np. $B=B+C \parallel A=C$
→ dostęp z wielu potoków (**Dual Pipe Access**)
 - Odczyt po zapisie (Read after Write)
 - np. $A=A+B \parallel C=A; A+B \parallel [M]=A$
→ przekazywanie wyników lub danych (**Result forwarding, Operand forwarding**)
 - Zapis po odczycie (Write after Read)
 - np. $B=A \parallel A=A+C$
→ przemianowywanie rejestrów (**Register renaming**)
 - Zapis po zapisie (Write after Write)
 - np. $A=[M] \parallel A=A+B$

Przyspieszanie mikroprocesorów

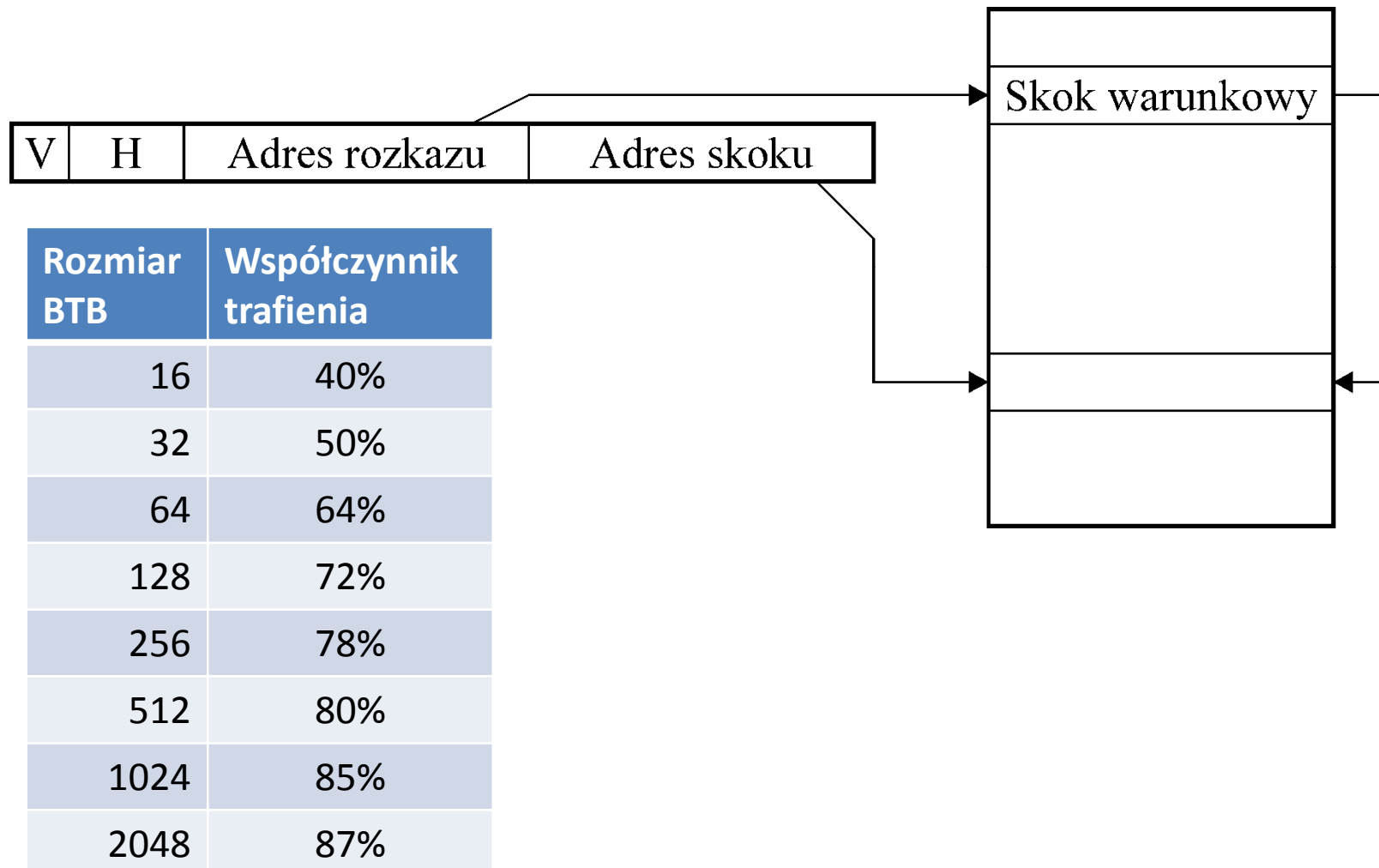
- Przewidywanie skoków
 - Rodzaje zakłóceń pracy potoku:
 - Przerwania
 - Skoki bezwarunkowe, wywołania
 - Skoki warunkowe
 - Skoki warunkowe
 - *Którą gałąź programu wprowadzić do potoku?*
 - Dłuższy potok → większy problem
 - Decyzja podejmowana pod koniec potoku
 - Opróżnienie potoku i napełnienie nową zawartością trwa do kilkudziesięciu Tclk

Przyspieszanie mikroprocesorów

- Przewidywanie skoków
 - Możliwe rozwiązania
 - Przewidywanie skoków
 - $P_{\text{success}} < 1$
 - Wykonanie wielościeżkowe
 - Zwiększenie zasobów procesora
 - Niejawne wykonanie (bez ujawniania wyników) do czasu, gdy wiadomo, która ścieżka jest prawidłowa
 - Dobry przykład: Intel Itanium

Przyspieszanie mikroprocesorów

- Przewidywanie skoków
 - Bufor przewidywania skokow (Branch target buffer)

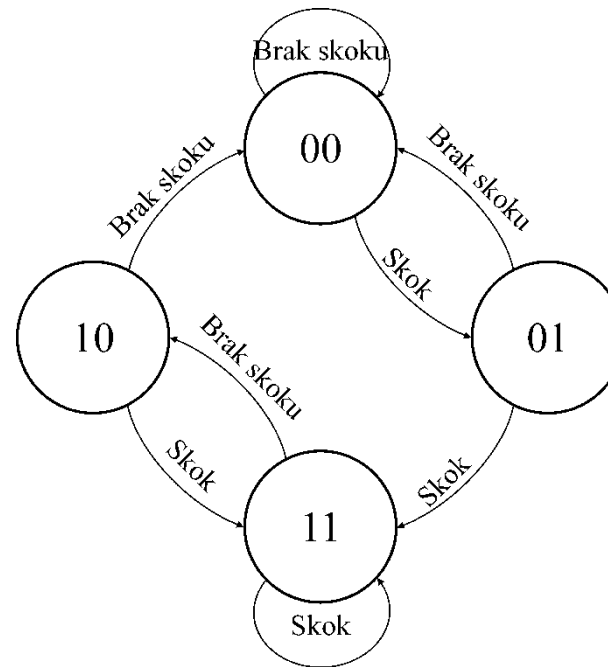


Przyspieszanie mikroprocesorów

- Przewidywanie skoków
 - Metody przewidywania skoków
 - Statyczne
 - Bit w rozkazie określony przez kompilator
 - » Wynik analizy możliwego wykonania programu
 - » *A jeśli kompilator popełni błąd?*
 - Według adresu skoku (Intel Pentium III)
 - » Ujemny (skok wstecz): koniec pętli → skok
 - » Dodatni (skok w przód): obsługa błędów → brak skoku

Przyspieszanie mikroprocesorów

- [illegible]



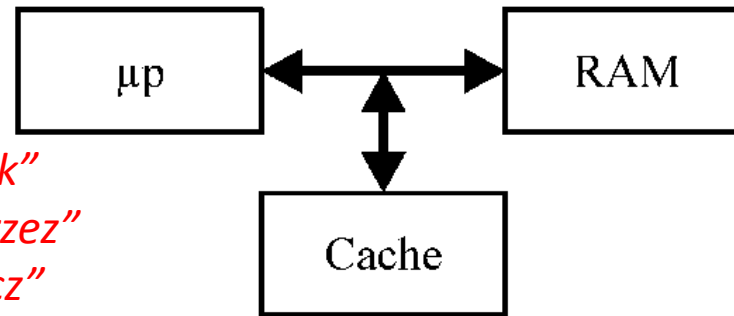
Przyspieszanie mikroprocesorów

- Optymalizacja
 - Dla danego typu procesora
 - Potoki synchroniczne:
 - Ręczne ustawianie instrukcji dla lepszego „parowania”

Przyspieszanie mikroprocesorów

- Pamięć podręczna (cache)
 - Lokalizacja w systemie

Look-aside

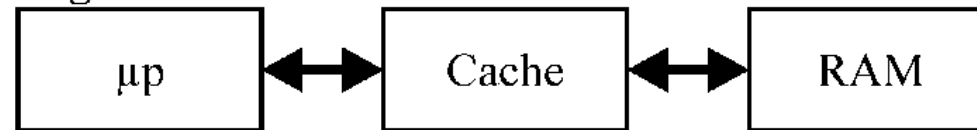


Look aside – „popatrz w bok”

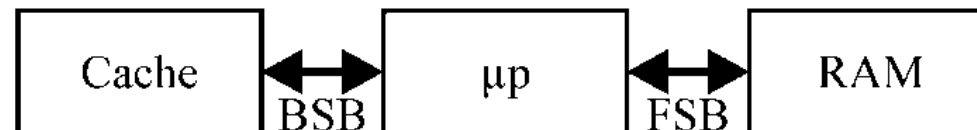
Look-through – „popatrz przez”

Look-back – „popatrz wstecz”

Look-through



Look-back



BSB – Back Side Bus (magistrala „tylna”)

FSB – Front Side Bus (magistrala „przednia”)

Przyspieszanie mikroprocesorów

- Pamięć podręczna
 - *Co lepsze?*
 - Mała pojemność, duża szybkość
 - Zwarty kod
 - Duża pojemność, mała szybkość
 - Kod rozproszony

Przyspieszanie mikroprocesorów

- Pamięć podręczna
 - Organizacja pamięci podręcznej

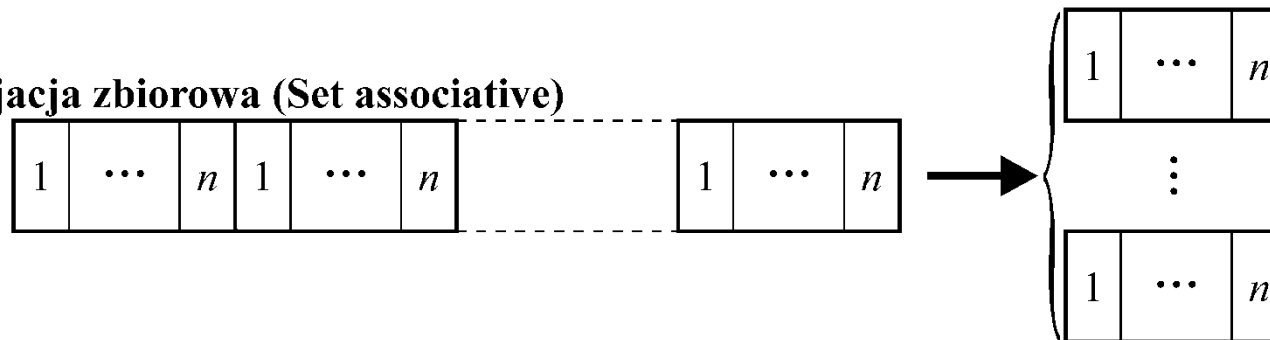
Mapowanie bezpośrednie (Direct-mapped)



Pełna asocjacja (Fully associative)



Asocjacja zbiorowa (Set associative)



Przyspieszanie mikroprocesorów

- Pamięć podręczna
 - Sposoby działania
 - Zapis jednoczesny (Write-through)
 - Zapis opóźniony (Write-back)
 - *Jak zapewnić spójność danych w pamięci podręcznej i operacyjnej w systemie wieloprocessorowym?*
 - Protokół MESI
 - Zmodyfikowany (Modified)
 - Wyłączny (Exclusive)
 - Współdzielony (Shared)
 - Nieważny (Invalid)