

RISC - taka sama długość, operacje na rejestrach

CISC - różne długości, bogactwo sposobów odnoszenie danych w pamięci.

$$1s = 10^9 \text{ ns} = 10^{12} \text{ ps} \quad 1KB = 2^{10} B$$

Częstotliwość zegara: $T_{\text{cykl}} = \frac{1}{f_{\text{częstotliwość zegara}}}$

CPI - liczba cykli potrzebna do wykonyania instrukcji!

$$\text{CPI} = \sum (\text{częstotliwość występowania} \cdot \text{CPI instrukcji})$$

$$\text{Speedup}_{\text{enh}} = \frac{\text{CPI}_{\text{org}}}{\text{CPI}_{\text{new}}} \quad \text{Frac}_{\text{enh}} = \frac{[\text{CPI instrukcji} \cdot \text{częstotliwość występowania}]}{\text{CPI org.}}$$

PRAWO AMDAHLA:

$$\frac{1}{(1 - \text{Frac}_{\text{enh}}) + \frac{\text{Frac}_{\text{enh}}}{\text{Speedup}_{\text{enh}}}}$$

BIG ENDIAN



LITTLE ENDIAN: (Pentium!)



Rejestr wskaźników i jego zastosowanie (rej. jedynkowe zmienne 6, 16, 32 bitowe).

przeniesienie, przeniesienie pomocnicze, nadmiar.

Stack i jego zastosowanie

zapamiętuje adresy powrotu do programu głównego (przy przeniesieniu lub wywołaniu podprogramu), przechowuje parametry do podprogramów, chwilowo przechowuje zawartości rejestrów w celu ich uzupełnienia.

Cache: miss rate + hit rate = 1

Poprawa wydajności AMAT: hit rate +, miss rate -, miss penalty -, hit time -

$$\text{AMAT} = \text{hit time} \cdot \text{hit rate} + \text{miss rate} \cdot \text{miss penalty}$$

$$\text{Throughput} = \frac{1s}{\text{czas najwolniejszego stopnia}}$$

Latency = liczba stopni potoku * czas najwolniejszego stopnia.

Harancy:

→ RAW, ↘ WAR, ↓ WAW

Tryby adresowania

Segmenty

real - address : $x \cdot 16 + y$

dl dh
0-7 8-15
bity bity

segmented $x + y$

EPIC = VLIW kompiluje kod, by jego czas wykonywania nieznacząco.

Tryby adresowania:

Rejestrowe:	Natychmiastowe:	Bezpośrednie:	Pośrednie rejestrowe:	Bezaw.-indeksowe:
mov ax, bx	mov bl, 44	mov al, DANA (adres komódki)	mov cx, [bx]	mov dx, [bx+di]

Rejestrowe względne:	Wygłszone bezaw.-indeksowe:	Adresowanie stałej:
mov ax, [bx+100h]	mov dh, [bx+di+20h]	pop i push

Adresowanie w instrukcjach skoku:

skok krótki - w tym samym segmencie, zakres od -128 do +127.

skok bliski - w tym samym segmencie, do dowolnej instrukcji.

skok daleki - dowolny inny segment, do dowolnej instrukcji.

Asembler Pentium - alokacja danych - define: db (def. bajt), dw (def. słowo), dd (def. podwójne słowo), dq (def. pochówe słowo), dt (def. 10 bajtów)

Adresowanie argumentów

Rejestrowe:	Natychmiastowe:	Bezpośrednie:	Pośrednie:
mov bx, cx	mov al, 77	mov al, response (zmienna)	mov ax, [bx]

Instrukcje: mov (kopij dane), xchg (zamiana argumentów), inc (incrementacja), dec (decrementacja), add (dodaj), sub (odejmij), cmp (porównaj).

Instrukcje skoku: jmp.

WZORY:

$$\text{Częstotliwość zegara} = \frac{T}{\text{cyk}} = \frac{f}{\text{częstotliwość}}$$

Prawo Andehle:

$$\frac{1}{(1 - \text{Frac}_{\text{enh}}) + \frac{\text{Frac}_{\text{enh}}}{\text{Speedup}_{\text{enh}}}} = \text{Speedup}_{\text{overall}}$$

$$\text{Speedup}_{\text{enh}} = \frac{\text{CPI}_{\text{org}}}{\text{CPI}_{\text{new}}} \quad \text{Frac}_{\text{enh}} = \frac{\text{CPI}_{\text{instr}} \cdot \text{częstotliwość występowanie}}{\text{CPI}_{\text{org}}}$$

$$AMAT = \text{hit time} \cdot \text{hit rate} + \text{miss penalty} \cdot \text{miss rate}$$

$$\text{Throughput} = \frac{1s}{\text{czas najwolniejszego stopnia}} \quad \text{d}$$

latency = ilość stopni w potoku + czas najwolniejszego stopnia

real address: $x \cdot 16 + y$ segmented: $x + y$