

BHMA 10

ΕΡΩΤΗΜΑΤΑ:

1) L1 Data cache:

size=65536 bytes

associativity = 2

L1 Instruction cache:

size=32768 bytes

associativity = 2

L2 cache:

size=2097152

associativity = 8

Το μέγεθος της cache line σύμφωνα με το αρχείο stats.txt είναι: cache_line_size=64

2) Για το πρώτο benchmark(specbzip) έχουμε τα ακόλουθα:

i) Χρόνος εκτέλεσης: sim_seconds = 0.084159sec

ii) CPI: system_cpu_cpi = 1.683172

iii) miss rate instruction cache: 0.000074

miss rate data cache: 0.014840

miss rate L2 cache: 0.281708

Για το δεύτερο benchmark(spechmmer):

i) Χρόνος εκτέλεσης: $\text{sim_seconds} = 0.059368\text{sec}$

ii) CPI: $\text{system_cpu_cpi} = 1.187362$

iii)miss rate instruction cache: 0.000205

miss rate data cache: 0.001645

miss rate L2 cache: 0.082246

Για το τρίτο benchmark(speclibm):

i) Χρόνος εκτέλεσης: $\text{sim_seconds} = 0.174681\text{sec}$

ii) CPI: $\text{system_cpu_cpi} = 3.493611$

iii)miss rate instruction cache: 0.000099

miss rate data cache: 0.060971

miss rate L2 cache: 0.999927

Για το τέταρτο benchmark(specmcf):

i) Χρόνος εκτέλεσης: $\text{sim_seconds} = 0.055477\text{sec}$

ii) CPI: $\text{system_cpu_cpi} = 1.109538$

iii)miss rate instruction cache: 0.000037

miss rate data cache: 0.002051

miss rate L2 cache: 0.724040

Για το πέμπτο benchmark(specsjeng):

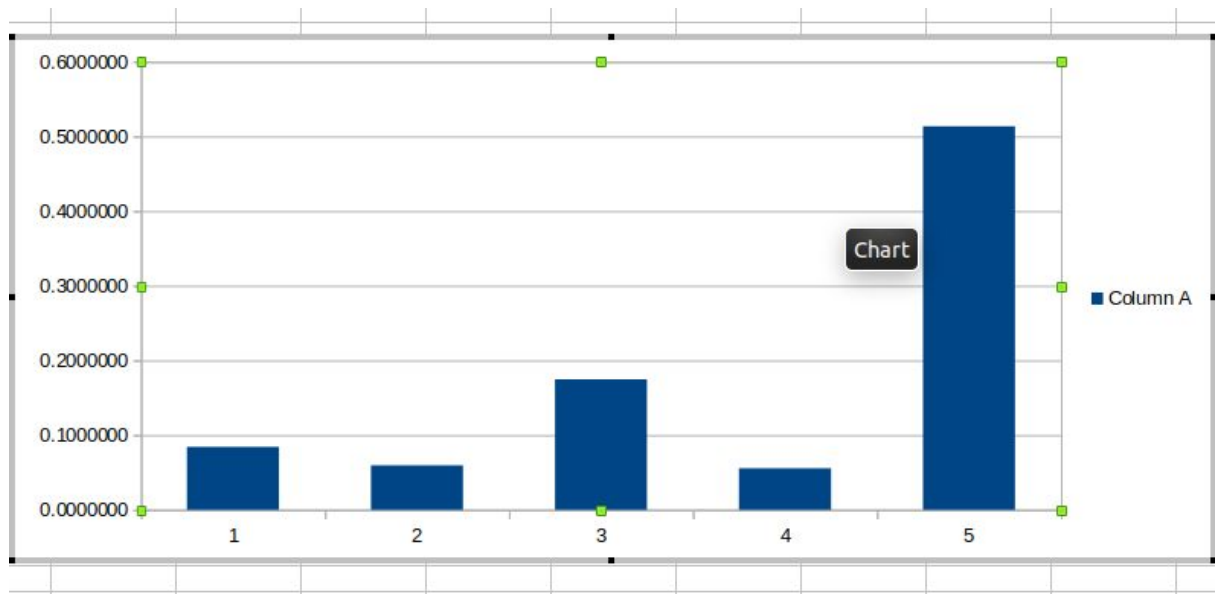
i) Χρόνος εκτέλεσης: $\text{sim_seconds} = 0.513541\text{sec}$

ii) CPI: $\text{system_cpu_cpi} = 10.270810$

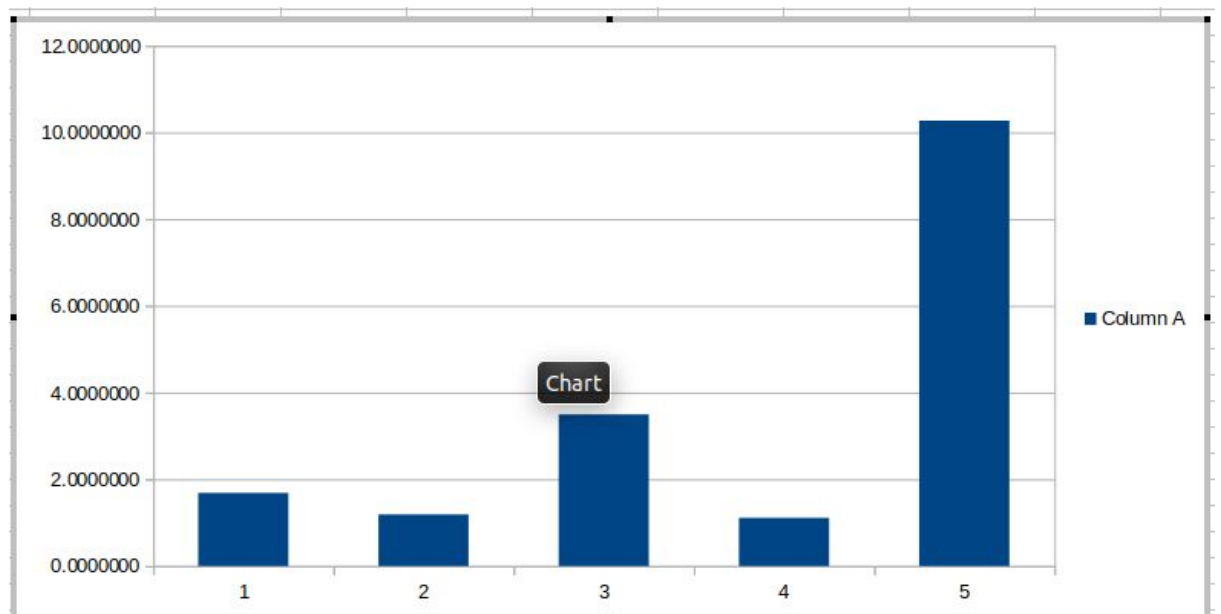
iii)miss rate instruction cache: 0.000020

miss rate data cache: 0.121829

miss rate L2 cache: 0.999979

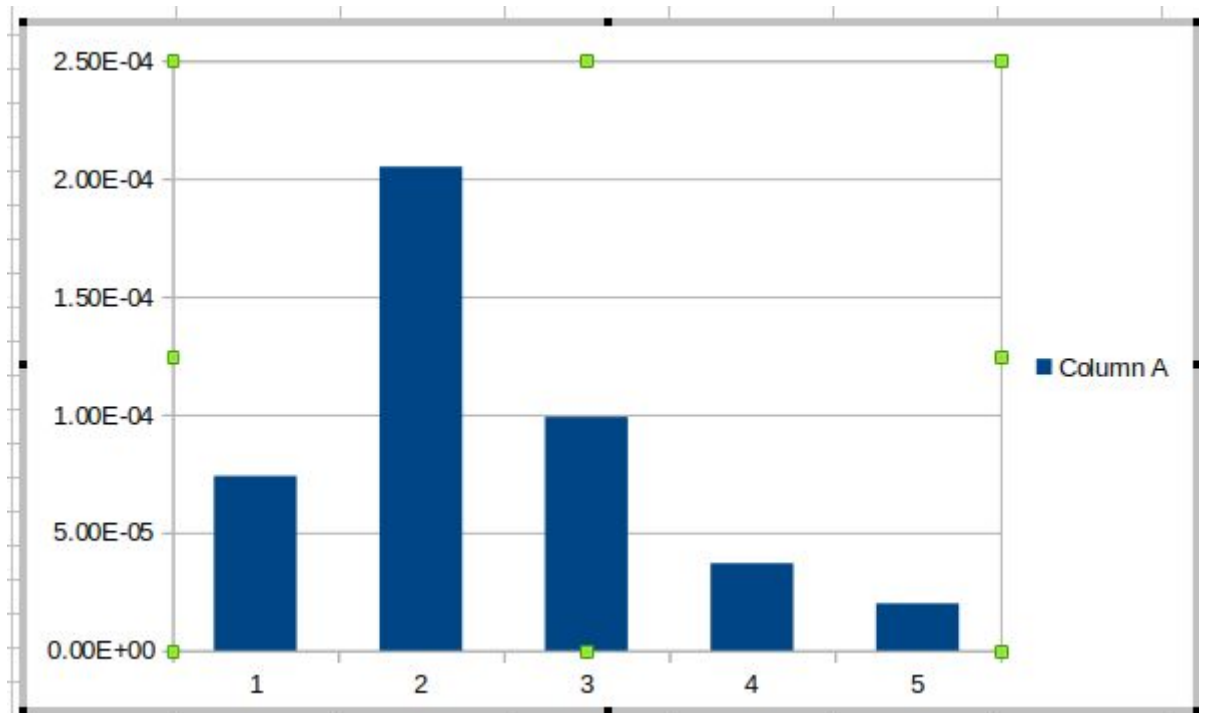


Η παραπάνω εικόνα είναι για τον χρόνο εκτέλεσης του καθενός από τα benchmarks. Παρατηρούμε ότι για το specsjeng χρειάστηκε πολύ παραπάνω χρόνος από τα υπόλοιπα.

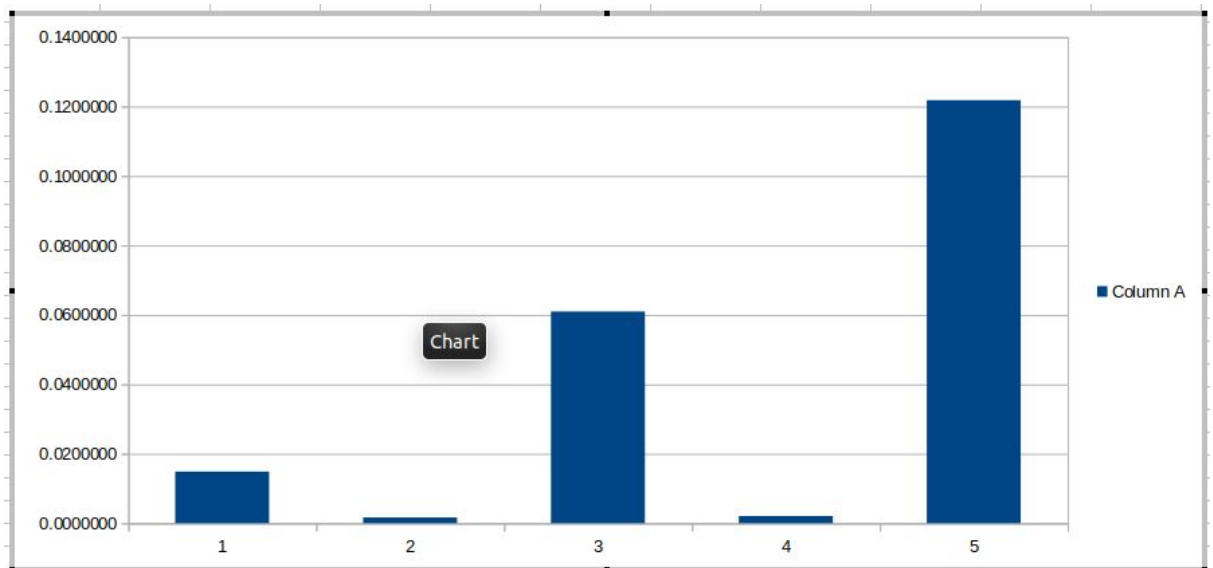


Το παραπάνω διάγραμμα επεικονίζει το cpi για το καθένα από τα benchmarks. Βλέπουμε για το specsfc έχουμε πολύ καλό cpi κοντά στην μονάδα,

ενώ για το `specsjpeg` είναι υπερβολικά μεγάλο και σίγουρα χρειάζεται διόρθωση.



Το παραπάνω διάγραμμα είναι για τα miss rates της instruction cache. Μία παρατήρηση είναι ότι για υψηλό cpi που έχουμε στο πέμπτο πρόγραμμα έχουμε πολύ χαμηλά miss rates της instruction cache.



Το παραπάνω διάγραμμα απεικονίζει τα miss rates για την data cache. Παρατηρούμε ότι στο πέμπτο benchmark τα miss rates data cache είναι υψηλά, ίσως γιt χρησιμοποιείται πολύ από το πρόγραμμα η data cache.

Παρατηρούμε ότι το 2ο benchmark δεν χρησιμοποιεί σχεδόν καθόλου την L2 cache ενώ το 3ο και το 5ο την χρησιμοποιούν αρκετά.

3) Και σε `cpu-clock=1GHz` και σε `cpu-clock=2GHz` έχουμε ακριβώς ίδιο `system.clk_domain.clock` το οποίο είναι ίσο με 1000. Όμως το `system.cpu_clk_domain.clock` στα 1GHz είναι 1000 ενώ στα 2GHz είναι 500. Στην πρώτη περίπτωση χρονίζονται όλα τα εξαρτήματα της motherboard να έχουν την ίδια συχνότητα. Αντίθετα, στην δεύτερη περίπτωση το `system.cpu_clk_domain.clock` χρονίζει μόνο την συχνότητα της cpu. Αυτό που χρονίζεται και στο 1GHz και στα 2GHz είναι μόνο η συχνότητα της cpu.

Αν προσθέσουμε και δεύτερο επεξεργαστή, η συχνότητα του θα είναι αυτή που θα του δηλώσουμε στην εντολή για τρέξιμο του προγράμματος.

Από το αρχείο `config.ini` ή `config.json` παρατηρούμε ότι αυτά που επηρεάζονται από την αλλαγή συχνότητας της cpu είναι τα ακόλουθα:

- a) `cpu`
- b) `data cache`
- c) `system.cpu.dtb.stage2_mmu.stage2_tlb.walker`
- d) `instruction cache`
- e) `cpu.itb.walker`

f) η L2 cache

g) toL2bus

Όσον αφορά το scaling, για το benchmark spechmmer έχουμε σχεδόν τέλειο scaling, γιατί στον διπλασιασμό της cpu clock έχουμε σχεδόν υποδιπλασιασμό του χρόνου που τρέχει το πρόγραμμα. Για το speclibm δεν έχουμε τόσο καλό scaling, από 0.17sec στα 2GHz έχουμε 0.26sec στα 1GHz, που όμως είναι καλό. Επίσης σχεδόν τέλειο scaling έχουμε και στο specmcf, και παρατηρούμε ότι όσο πιο μικρό πρόγραμμα έχουμε, τόσο καλύτερο scaling έχουμε.