*

EE6304: COMPUTER ARCHITECTURE

CACHE DESIGN

LIST OF CONTENTS

1. INTRODUCTION

Computer architectural designs performance mainly depends upon the processor speed up and memory latencies. Memory performance is very limited when compared to the processor performance especially for the data driven benchmarks. Memory design used today consists of hierarchical structure to give better performance. It gives the illusion of higher speed and greater size memory at minimal cost. Most integral part for this is to configure the cache sizes and levels to achieve optimal performance. In this project we will study 3 benchmarks (GCC, Anagram Alpha and Go) with fixed size L1 and L2 caches (256 KB and 1MB respectively).Simulation for optimal cache configuration with minimum CPI and cost. In the simulation approach first we calculate the CPI for multiple cache combinations (L1 separate - L2 unified, L1 seperate-L2 separate and L1 unified-L2 unified) with variable block sizes, set Associativity and page replacement policies (LRU, FIFO and Random). Then we map CPI with cache cost function

2. AIM OF PROJECT

To simulate a cache memory hierarchical configuration using simplescalar with minimum CPI at minimum cost, by using different block sizes, cache set-associativity and page replacement policy.

3. APPROACH

We have considered 3 different benchmarks (anagram, GCC and go)to run on Simple scalar 3.0 for the simulation. We considered two level cache (L1 and L2) with fixed size 256KB and 1MB.
Below are the steps followed to achieve optimal CPI

1) select one of the benchmarks from anagram,gcc and go
2) Select one of cache structure combination at a time (L1 separate-L2 seprate,L1-seprate,L2-unified,L1-unified-L2-unified
3) For each cache structure combination we varied the replacement policy iteratively (LRU,FIFO,Random)
4) For each cache structure combination and replacement policy we changed the block size 32,64bytes and set associativity(1,2,4 or 8)
5) Record the CPI for each iteration
6) Repeat steps 1 to 5 for all combinations
7) Calculate the cost Function C for the above cache configuration
8) Compare the results and shown through graphs

4. (a) TESTING BENCHMARKS

Simulation-Simplescalar- 3.0
Benchmarks

- Anagram
- GCC
- GO

(b) FINDING CPI

Given configuration is as follows:

Cache levels: Two levels.
Unified caches: Separate L1 data and instruction cache, unified L2 cache.
Size: 64K Separate L1 data and instruction caches, 1MB unified L2 cache.
Associativity: Two-way set-associative L1 caches, Direct-mapped L2 cache.

Block size: 64 bytes.
Block replacement policy: FIFO.

Also given are the values:

L1 miss penalty = 4 cycles
L2 miss penalty = 70 cycles
Cache hit time = 1 cycle

The formula to calculate CPI is as follows:

CPI = 1 + Miss Penalty of L1 ((Miss rate of L1*Miss access of L1) / Number of instructions)
+ Miss Penalty of L2 ((Miss rate of L2* Miss access of L2) / Number of instructions)
Ideal CPI = 1

CPI = 1 + 4(((il1 access*il1 miss rate) + (dl1 access*dl1 miss rate))/ Number of instructions) + 70 ((dl2 access*dl2 miss rate) / Number of instructions)

CPI for GCC:

Number of instructions = 337327104
il1 accesses = 337327104
il1 miss rate = 0.0047
dl1 accesses = 124102800
dl1 miss rate = 0.0106
dl2 accesses = 3330118
dl2 miss rate = 0.1311

CPI = 1.124994999

CPI for anagram:

Number of instructions = 25593315
il1 accesses =25593315
il1 miss rate = 0.0527
dl1 accesses = 2022
dl1 miss rate = 0.0786
dl2 accesses = 425
dl2 miss rate = 0.9953

CPI = 1.001000417

CPI for go:

Number of instructions = 709786
il1 accesses = 709786
il1 miss rate = 0.0010
dl1 accesses = 196786
dl1 miss rate = 0.0264
dl2 accesses = 9636
dl2 miss rate = 0.5583
CPI = 1.56383792

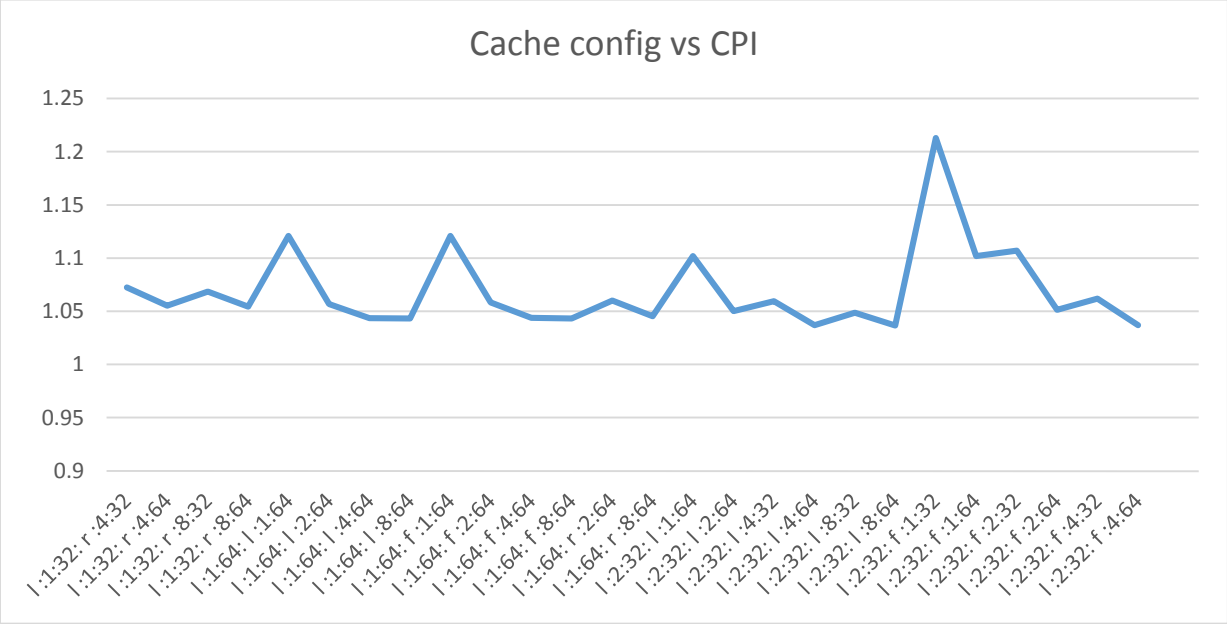
(c) OPTIMIZE CPI FOR EACH BENCHMARK

In this part all possible combinations of various parameters that determine the performance and hence affect the CPI of the benchmark were taken. To optimize the CPI for each benchmark, we have chosen 2 caches one is an L1 cache with the size of 256KB and another L2 cache with a size of 1 MB. The results are compared to the CPI for each benchmark, for the configurations:

- L1 Separate data and instruction cache, L2 Unified data and instruction cache
- L1 Separate data and instruction cache, L2 Separate data and instruction cache
- L1 Unified data and instruction cache, L2 Unified data and instruction cache
- Block size: 32 bytes, 64 bytes
- Associativity: 1-way, 2-way, 4-way, 8-way, and fully associative
- Replacement Policy: FIFO (f), Random(r), LRU (l)
- Number of sets : This parameter is calculated from the above parameters using the formula given below
  Number of sets = (cache size) / (associativity*block size)

  On running the simulations for benchmark the following results were obtained

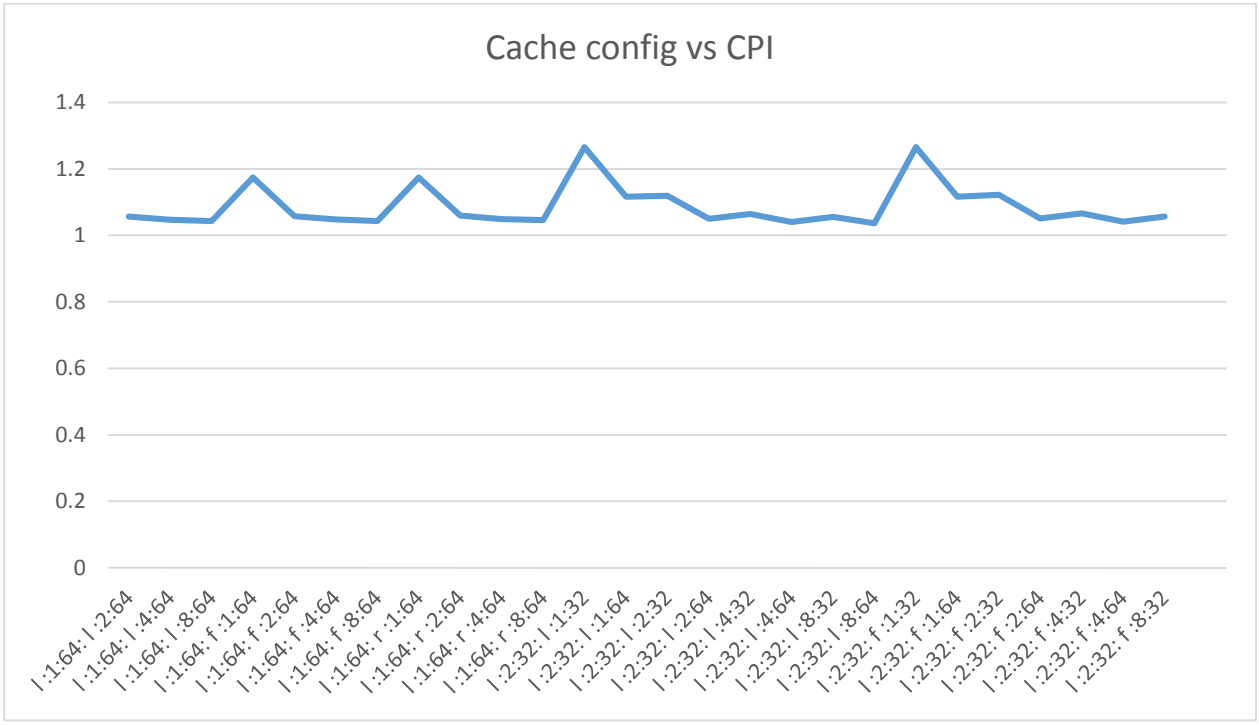
1. **GCC Benchmark**
   (a) L1 Separate data and instruction cache, L2 Unified data and instruction cache



The figure shows the CPI as a function of various configurations. The lowest value of CPI observed is **1.03653748** for the configuration **l: 2:32: l: 8:64** where L1 is 2-way and L2 is 8-way associative and replacement policy is LRU.

Cache config	dl1.accesses	dl1.miss_rate	dl2.accesses	dl2.miss_rate	il1.accesses	il1.miss_rate	CPI
l :1:32: l :1:32	124102798	0.012	3992178	0.1578	337327098	0.0058	1.17150768
l :1:32: l :1:64	124102798	0.012	3992178	0.0481	337327098	0.0058	1.080608371
l :1:32: l :2:32	124102798	0.012	3992178	0.0532	337327098	0.0058	1.084834181
l :1:32: l :2:64	124102798	0.012	3992178	0.0181	337327098	0.0058	1.055724572
l :1:32: l :4:32	124102798	0.012	3992178	0.0298	337327098	0.0058	1.065467341
l :1:32: l :4:64	124102798	0.012	3992178	0.0152	337327098	0.0058	1.053341276
l :1:32: l :8:32	124102798	0.012	3992178	0.0293	337327098	0.0058	1.065043805
l :1:32: l :8:64	124102798	0.012	3992178	0.0152	337327098	0.0058	1.053341276
l :1:32: f :1:32	124102798	0.012	3992178	0.1578	337327098	0.0058	1.17150768
l :1:32: f :1:64	124102798	0.012	3992178	0.0481	337327098	0.0058	1.080608371
l :1:32: f :2:32	124102798	0.012	3992178	0.0555	337327098	0.0058	1.086747042
l :1:32: f :2:64	124102798	0.012	3992178	0.0182	337327098	0.0058	1.055872736
l :1:32: f :4:32	124102798	0.012	3992178	0.03	337327098	0.0058	1.065630031
l :1:32: f :4:64	124102798	0.012	3992178	0.0152	337327098	0.0058	1.053341276
l :1:32: f :8:32	124102798	0.012	3992178	0.0293	337327098	0.0058	1.065043805
l :1:32: f :8:64	124102798	0.012	3992178	0.0152	337327098	0.0058	1.053341276
l :1:32: r :1:32	124102798	0.012	3992178	0.1578	337327098	0.0058	1.17150768
l :1:32: r :1:64	124102798	0.012	3992178	0.0481	337327098	0.0058	1.080608371
l :1:32: r :2:32	124102798	0.012	3992178	0.0598	337327098	0.0058	1.090293037
l :1:32: r :2:64	124102798	0.012	3992178	0.0208	337327098	0.0058	1.057965927
l :1:32: r :4:32	124102798	0.012	3992178	0.0382	337327098	0.0058	1.072447479
l :1:32: r :4:64	124102798	0.012	3992178	0.0176	337327098	0.0058	1.055347519
l :1:32: r :8:32	124102798	0.012	3992178	0.0335	337327098	0.0058	1.068506379
l :1:32: r :8:64	124102798	0.012	3992178	0.0163	337327098	0.0058	1.054257035
l :1:64: l :1:64	124102798	0.0099	2976687	0.1459	337327098	0.004	1.120814095
l :1:64: l :4:64	124102798	0.0099	2976687	0.0209	337327098	0.004	1.043597411
l :1:64: l :8:64	124102798	0.0099	2976687	0.0204	337327098	0.004	1.0432436
l :1:64: f :1:64	124102798	0.0099	2976687	0.1459	337327098	0.004	1.120814095
l :1:64: f :2:64	124102798	0.0099	2976687	0.0446	337327098	0.004	1.058189775
l :1:64: f :4:64	124102798	0.0099	2976687	0.0212	337327098	0.004	1.043735407
l :1:64: f :8:64	124102798	0.0099	2976687	0.0204	337327098	0.004	1.0432436
l :1:64: r :2:64	124102798	0.0099	2976687	0.0477	337327098	0.004	1.060123596
l :1:64: r :8:64	124102798	0.0099	2976687	0.0235	337327098	0.004	1.045191323
l :2:32: l :1:64	124102798	0.0075	2444105	0.1534	337327098	0.0032	1.101773567
l :2:32: l :2:64	124102798	0.0075	2444105	0.0518	337327098	0.0032	1.050233178
l :2:32: l :4:32	124102798	0.0075	2444105	0.0698	337327098	0.0032	1.05935465
l :2:32: l :4:64	124102798	0.0075	2444105	0.0255	337327098	0.0032	1.036892121
l :2:32: l :8:32	124102798	0.0075	2444105	0.0486	337327098	0.0032	1.048632002
l :2:32: l :8:64	**124102798**	**0.0075**	**2444105**	**0.0248**	**337327098**	**0.0032**	**1.03653748**
l :2:32: f :1:32	124102798	0.0075	2444105	0.3724	337327098	0.0032	1.212846737
l :2:32: f :1:64	124102798	0.0075	2444105	0.1534	337327098	0.0032	1.101773567

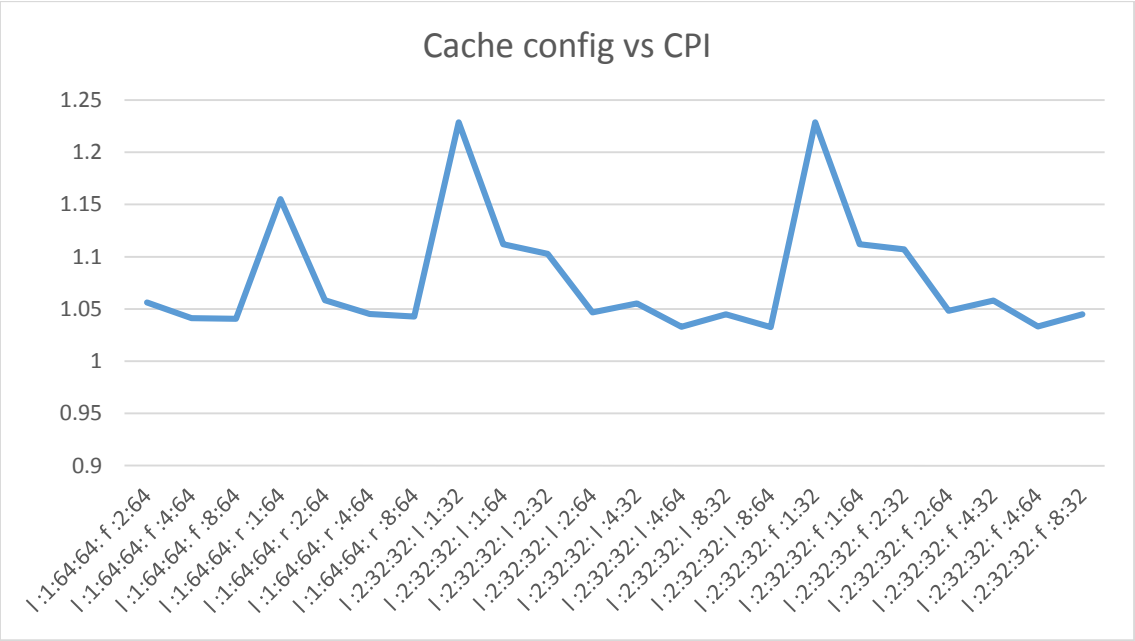
(b) L1 Separate data and instruction cache, L2 Separate data and instruction cache



The figure shows the CPI as a function of various configurations. The lowest value of CPI observed is **1.036546403** for the configuration **I: 2:32: I: 8:64** where L1 is 2-way and L2 is 8-way associative and replacement policy is LRU.

Cache config	sim_num_insn	dl1.accesses	dl1.miss_rate	dl2.accesses	dl2.miss_rate	il1.accesses	il1.miss_rate	il2.accesses	il2.miss_rate	CPI
I :1:32: r :8:64	337327098	124102798	0.012	2040901	0.0271	337327098	0.0058	1951277	0.0057	1.054521407
I :1:64: I :1:64	337327098	124102798	0.0099	1613361	0.1913	337327098	0.004	1363326	0.2815	1.174358859
I :1:64: I :2:64	337327098	124102798	0.0099	1613361	0.0581	337327098	0.004	1363326	0.023	1.056619104
I :1:64: I :4:64	337327098	124102798	0.0099	1613361	0.0435	337327098	0.004	1363326	0.008	1.047509044
I :1:64: I :8:64	337327098	124102798	0.0099	1613361	0.0308	337327098	0.004	1363326	0.008	1.043252523
I :1:64: f :1:64	337327098	124102798	0.0099	1613361	0.1913	337327098	0.004	1363326	0.2815	1.174358859
I :1:64: f :2:64	337327098	124102798	0.0099	1613361	0.0613	337327098	0.004	1363326	0.024	1.05799658
I :1:64: f :4:64	337327098	124102798	0.0099	1613361	0.0447	337327098	0.004	1363326	0.008	1.047889832
I :1:64: f :8:64	337327098	124102798	0.0099	1613361	0.0308	337327098	0.004	1363326	0.008	1.043253768
I :1:64: r :1:64	337327098	124102798	0.0099	1613361	0.1913	337327098	0.004	1363326	0.2815	1.174358859
I :1:64: r :2:64	337327098	124102798	0.0099	1613361	0.0657	337327098	0.004	1363326	0.0241	1.059474078
I :1:64: r :4:64	337327098	124102798	0.0099	1613361	0.0475	337327098	0.004	1363326	0.0089	1.049085111
I :1:64: r :8:64	337327098	124102798	0.0099	1613361	0.0392	337327098	0.004	1363326	0.0084	1.046181786
I :2:32: I :1:32	337327098	124102798	0.0075	1348871	0.4056	337327098	0.0032	1095234	0.5633	1.265528037
I :2:32: I :1:64	337327098	124102798	0.0075	1348871	0.1752	337327098	0.0032	1095234	0.1893	1.116037022
I :2:32: I :2:32	337327098	124102798	0.0075	1348871	0.1823	337327098	0.0032	1095234	0.1943	1.119154501
I :2:32: I :2:64	337327098	124102798	0.0075	1348871	0.07	337327098	0.0032	1095234	0.0293	1.050218444
I :2:32: I :4:32	337327098	124102798	0.0075	1348871	0.1227	337327098	0.0032	1095234	0.0264	1.064302193
I :2:32: I :4:64	337327098	124102798	0.0075	1348871	0.0521	337327098	0.0032	1095234	0.01	1.040804792
I :2:32: I :8:32	337327098	124102798	0.0075	1348871	0.0989	337327098	0.0032	1095234	0.018	1.055753042
I :2:32: I :8:64	**337327098**	**124102798**	**0.0075**	**1348871**	**0.0369**	**337327098**	**0.0032**	**1095234**	**0.01**	**1.036546403**
I :2:32: f :1:32	337327098	124102798	0.0075	1348871	0.4056	337327098	0.0032	1095234	0.5633	1.265528037
I :2:32: f :1:64	337327098	124102798	0.0075	1348871	0.1752	337327098	0.0032	1095234	0.1893	1.116037022
I :2:32: f :2:32	337327098	124102798	0.0075	1348871	0.191	337327098	0.0032	1095234	0.1956	1.121901567

(c) L1 Unified data and instruction cache, L2 Unified data and instruction cache

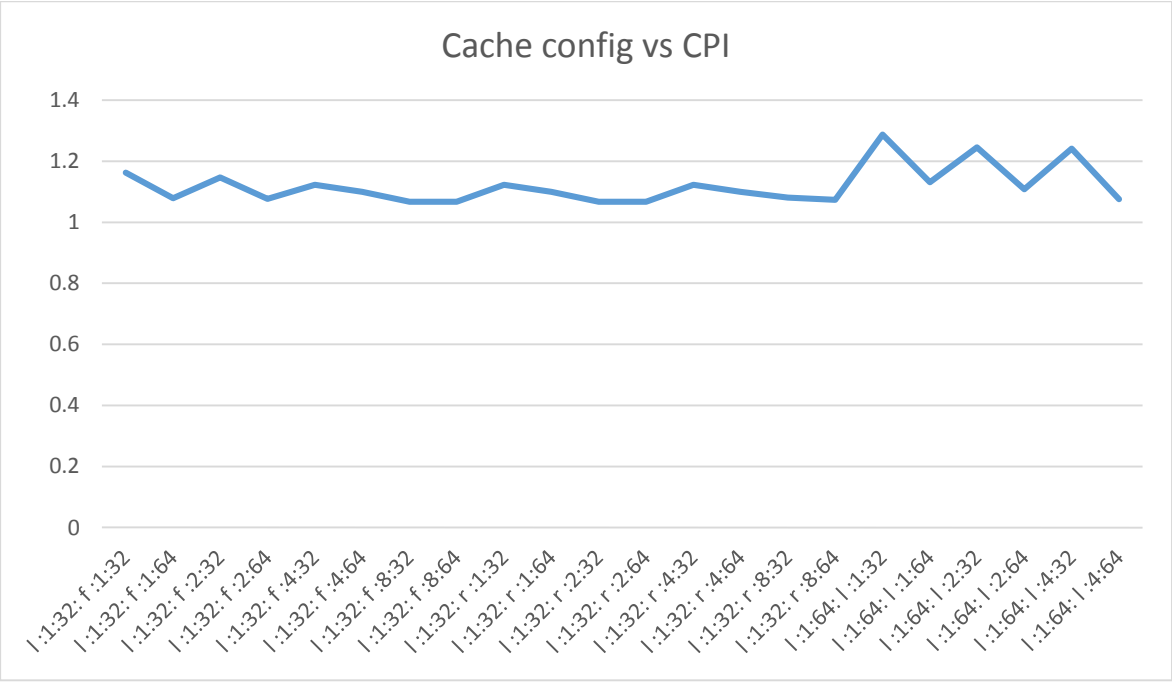


The figure shows the CPI as a function of various configurations. The lowest value of CPI observed is **1.032630637** for the configuration **l: 2:32: l: 8:64** where L1 is 2-way and L2 is 8-way associative and replacement policy is LRU.

Cache config	sim_num_insn	ul1.accesses	ul1.miss_rate	ul2.accesses	ul2.miss_rate	CPI
l :1:32:32: r :4:32	337327098	461429896	0.0066	3612705	0.0424	1.067705999
l :1:32:32: r :4:64	337327098	461429896	0.0066	3612705	0.0194	1.050488793
l :1:32:32: r :8:32	337327098	461429896	0.0066	3612705	0.0369	1.063628146
l :1:32:32: r :8:64	337327098	461429896	0.0066	3612705	0.018	1.049457242
l :1:64:64: l :1:64	337327098	461429896	0.0051	2769945	0.2211	1.155211788
l :1:64:64: l :2:64	337327098	461429896	0.0051	2769945	0.0456	1.054342275
l :1:64:64: l :4:64	337327098	461429896	0.0051	2769945	0.0225	1.041051644
l :1:64:64: l :8:64	337327098	461429896	0.0051	2769945	0.0219	1.040696173
l :1:64:64: f :1:64	337327098	461429896	0.0051	2769945	0.2211	1.155211788
l :1:64:64: f :2:64	337327098	461429896	0.0051	2769945	0.0487	1.056137476
l :1:64:64: f :4:64	337327098	461429896	0.0051	2769945	0.0227	1.041195659
l :1:64:64: f :8:64	337327098	461429896	0.0051	2769945	0.0219	1.040696173
l :1:64:64: r :1:64	337327098	461429896	0.0051	2769945	0.2211	1.155211788
l :1:64:64: r :2:64	337327098	461429896	0.0051	2769945	0.0527	1.058428635
l :1:64:64: r :4:64	337327098	461429896	0.0051	2769945	0.0295	1.045086748
l :1:64:64: r :8:64	337327098	461429896	0.0051	2769945	0.0254	1.042709264
l :2:32:32: l :1:32	337327098	461429896	0.0037	2128973	0.4723	1.22870267
l :2:32:32: l :1:64	337327098	461429896	0.0037	2128973	0.2082	1.112033685
l :2:32:32: l :2:32	337327098	461429896	0.0037	2128973	0.1875	1.102909307
l :2:32:32: l :2:64	337327098	461429896	0.0037	2128973	0.0604	1.046727251
l :2:32:32: l :4:32	337327098	461429896	0.0037	2128973	0.0798	1.055319979
l :2:32:32: l :4:64	337327098	461429896	0.0037	2128973	0.0293	1.032988598
l :2:32:32: l :8:32	337327098	461429896	0.0037	2128973	0.0558	1.044728479
l :2:32:32: l :8:64	**337327098**	**461429896**	**0.0037**	**2128973**	**0.0285**	**1.032630637**
l :2:32:32: f :1:32	337327098	461429896	0.0037	2128973	0.4723	1.22870267
l :2:32:32: f :1:64	337327098	461429896	0.0037	2128973	0.2082	1.112033685

2. Anagram Benchmark

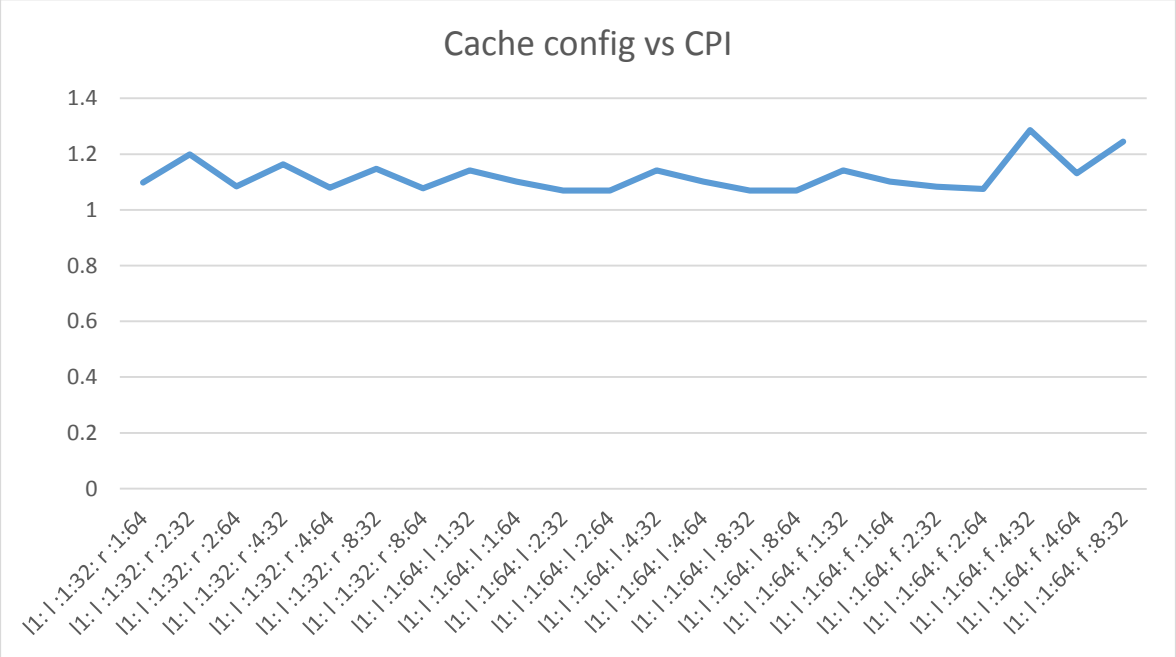
(a) L1 Separate data and instruction cache, L2 Unified data and instruction cache



The figure shows the CPI as a function of various configurations. The lowest value of CPI observed is **1.06725545** for the configuration **l: l: 32: r: 2:64** where L1 is 1-way and L2 is 2-way associative and replacement policy is Random.

Cache config	sim_num_insn	dl1.accesses	dl1.miss_rate	il1.misses	il1.miss_rate	dl2.accesses	dl2.miss_rate	CPI
l :1:32: l :1:32	25593315	11153944	0.0095	25593315	0	181827	0.2355	1.13380518
l :1:32: l :1:64	25593315	11153944	0.0095	25593315	0	181827	0.1184	1.07554777
l :1:32: l :2:32	25593315	11153944	0.0095	25593315	0	181827	0.2355	1.13380518
l :1:32: l :2:64	25593315	11153944	0.0095	25593315	0	181827	0.1184	1.07554777
l :1:32: l :4:32	25593315	11153944	0.0095	25593315	0	181827	0.4606	1.24576332
l :1:32: l :4:64	25593315	11153944	0.0095	25593315	0	181827	0.1624	1.0974449
l :1:32: l :8:32	25593315	11153944	0.0095	25593315	0	181827	0.3656	1.19852012
l :1:32: l :8:64	25593315	11153944	0.0095	25593315	0	181827	0.1348	1.08374757
l :1:32: f :1:32	25593315	11153944	0.0095	25593315	0	181827	0.293	1.16241421
l :1:32: f :1:64	25593315	11153944	0.0095	25593315	0	181827	0.1254	1.07903228
l :1:32: f :2:32	25593315	11153944	0.0095	25593315	0	181827	0.2615	1.14672027
l :1:32: f :2:64	25593315	11153944	0.0095	25593315	0	181827	0.1213	1.07700831
l :1:32: f :4:32	25593315	11153944	0.0048	25593315	0	91316	0.4604	1.12338222
l :1:32: f :4:64	25593315	11153944	0.0048	25593315	0	91316	0.3655	1.09967173
l :1:32: f :8:32	25593315	11153944	0.0048	25593315	0	91316	0.2357	1.06725545
l :1:32: f :8:64	25593315	11153944	0.0048	25593315	0	91316	0.2357	1.06725545
l :1:32: r :1:32	25593315	11153944	0.0048	25593315	0	91316	0.4604	1.12338222
l :1:32: r :1:64	25593315	11153944	0.0048	25593315	0	91316	0.3655	1.09967173
l :1:32: r :2:32	25593315	11153944	0.0048	25593315	0	91316	0.2357	1.06725545
l :1:32: r :2:64	25593315	11153944	0.0048	25593315	0	91316	0.2357	**1.06725545**
l :1:32: r :4:32	25593315	11153944	0.0048	25593315	0	91316	0.4604	1.12338222

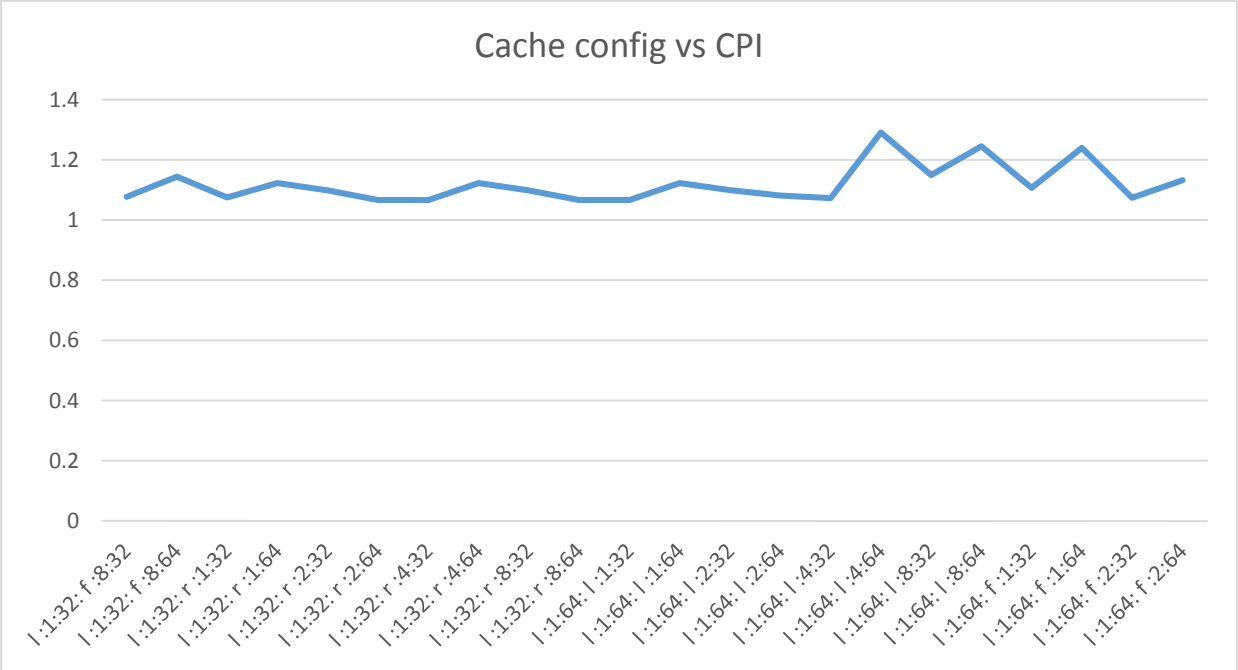
(b) L1 Separate data and instruction cache, L2 Separate data and instruction cache



The figure shows the CPI as a function of various configurations. The lowest value of CPI observed is **1.068766629** for the configuration **l1: l: 1:64: l: 2:32** where L1 is 1-way and L2 is 2-way associative and replacement policy is LRU.

Cache config	sim_num_insn	dl1.accesses	dl1.miss_rate	il1.access	il1.miss_rate	dl2.access	dl2.miss_rate	il2.access	il2.miss_rate	CPI
l1: l :1:32: l :4:64	25593315	11153944	0.0097	25593315	0	184256	0.1141	852	0.5751	1.07587958
l1: l :1:32: l :8:32	25593315	11153944	0.0097	25593315	0	184256	0.2278	852	1	1.134136981
l1: l :1:32: l :8:64	25593315	11153944	0.0097	25593315	0	184256	0.1141	852	0.5751	1.07587958
l1: l :1:32: f :1:32	25593315	11153944	0.0097	25593315	0	184256	0.4518	852	1	1.247052404
l1: l :1:32: f :1:64	25593315	11153944	0.0097	25593315	0	184256	0.1579	852	0.5751	1.09794081
l1: l :1:32: f :2:32	25593315	11153944	0.0097	25593315	0	184256	0.3558	852	1	1.198665941
l1: l :1:32: f :2:64	25593315	11153944	0.0097	25593315	0	184256	0.1142	852	0.5751	1.075895991
l1: l :1:32: f :4:32	25593315	11153944	0.0097	25593315	0	184256	0.2278	852	1	1.134136981
l1: l :1:32: f :4:64	25593315	11153944	0.0097	25593315	0	184256	0.1141	852	0.5751	1.07587958
l1: l :1:32: f :8:32	25593315	11153944	0.0097	25593315	0	184256	0.2278	852	1	1.134136981
l1: l :1:32: f :8:64	25593315	11153944	0.0097	25593315	0	184256	0.1141	852	0.5751	1.07587958
l1: l :1:32: r :1:32	25593315	11153944	0.0097	25593315	0	184256	0.4518	852	1	1.247052404
l1: l :1:32: r :1:64	25593315	11153944	0.0097	25593315	0	184256	0.1579	852	0.5751	1.09794081
l1: l :1:32: r :2:32	25593315	11153944	0.0097	25593315	0	184256	0.3562	852	1	1.198838251
l1: l :1:32: r :2:64	25593315	11153944	0.0097	25593315	0	184256	0.1304	852	0.5751	1.084065702
l1: l :1:32: r :4:32	25593315	11153944	0.0097	25593315	0	184256	0.2849	852	1	1.16292106
l1: l :1:32: r :4:64	25593315	11153944	0.0097	25593315	0	184256	0.1209	852	0.5751	1.079295707
l1: l :1:32: r :8:32	25593315	11153944	0.0097	25593315	0	184256	0.2532	852	1	1.146972754
l1: l :1:32: r :8:64	25593315	11153944	0.0097	25593315	0	184256	0.117	852	0.5751	1.077312767
l1: l :1:64: l :1:32	25593315	11153944	0.0056	25593315	0	102180	0.4671	490	1	1.141798981
l1: l :1:64: l :1:64	25593315	11153944	0.0056	25593315	0	102180	0.3217	490	1	1.101150085
l1: l :1:64: l :2:32	**25593315**	**11153944**	**0.0056**	**25593315**	**0**	**102180**	**0.2058**	**490**	**1**	**1.068766629**
l1: l :1:64: l :2:64	25593315	11153944	0.0056	25593315	0	102180	0.2058	490	1	1.068766629
l1: l :1:64: l :4:32	25593315	11153944	0.0056	25593315	0	102180	0.4671	490	1	1.141798981
l1: l :1:64: l :4:64	25593315	11153944	0.0056	25593315	0	102180	0.3217	490	1	1.101144615
l1: l :1:64: l :8:32	25593315	11153944	0.0056	25593315	0	102180	0.2058	490	1	1.068766629

(c) L1 Unified data and instruction cache, L2 Unified data and instruction cache

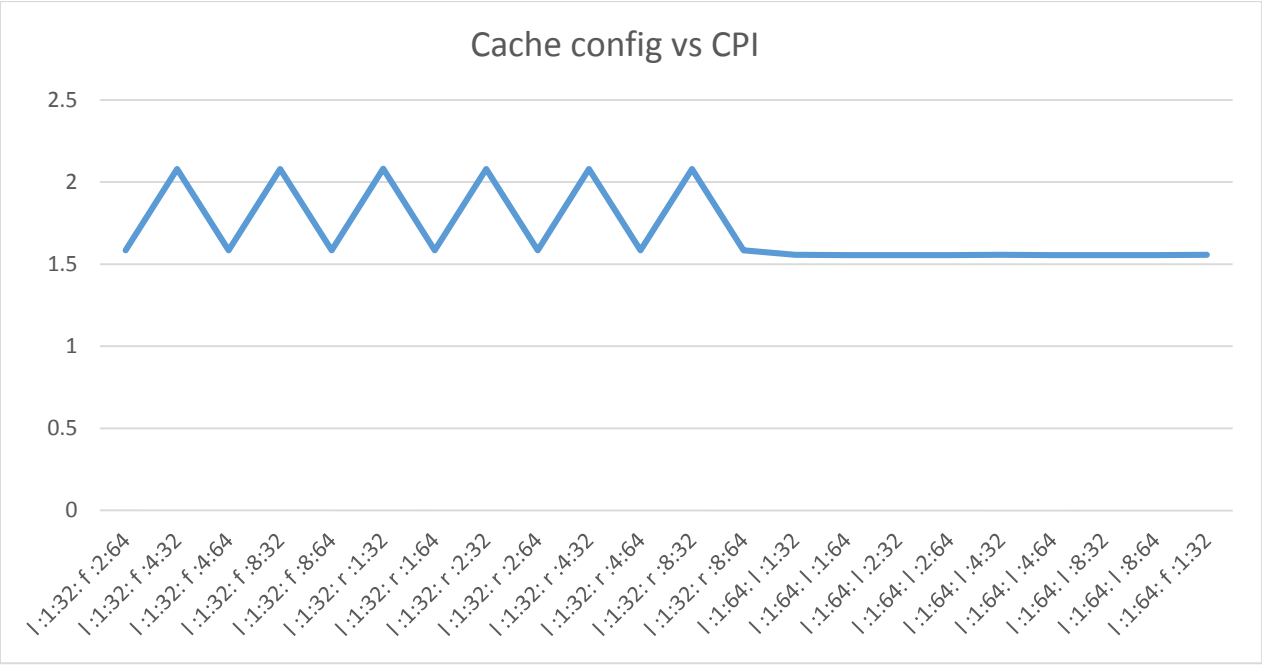


The figure shows the CPI as a function of various configurations. The lowest value of CPI observed is **1.06606116** for the configuration **l: 1:32: r: 2:64** where L1 is 1-way and L2 is 2-way associative and replacement policy is Random.

Cache config	sim_num_insn	ul1.accesses	ul1.miss_rate	ul2.accesses	ul2.miss_rate	CPI
l :1:32: l :1:32	25593315	36747259	0.0025	165565	0.13	1.07309385
l :1:32: l :1:64	25593315	36747259	0.0025	165565	0.5097	1.24502977
l :1:32: l :2:32	25593315	36747259	0.0025	165565	0.2423	1.1239501
l :1:32: l :2:64	25593315	36747259	0.0025	165565	0.4024	1.19643817
l :1:32: l :4:32	25593315	36747259	0.0025	165565	0.1311	1.07360531
l :1:32: l :4:64	25593315	36747259	0.0025	165565	0.2586	1.13135125
l :1:32: l :8:32	25593315	36747259	0.0025	165565	0.13	1.07309385
l :1:32: l :8:64	25593315	36747259	0.0025	165565	0.2586	1.13135125
l :1:32: f :1:32	25593315	36747259	0.0025	165565	0.13	1.07309385
l :1:32: f :1:64	25593315	36747259	0.0025	165565	0.5097	1.24502977
l :1:32: f :2:32	25593315	36747259	0.0025	165565	0.2423	1.1239501
l :1:32: f :2:64	25593315	36747259	0.0025	165565	0.4034	1.19692775
l :1:32: f :4:32	25593315	36747259	0.0025	165565	0.1537	1.08382361
l :1:32: f :4:64	25593315	36747259	0.0025	165565	0.3219	1.16001499
l :1:32: f :8:32	25593315	36747259	0.0025	165565	0.1384	1.07691751
l :1:32: f :8:64	25593315	36747259	0.0025	165565	0.2873	1.14432652
l :1:32: r :1:32	25593315	36747259	0.0025	165565	0.1335	1.07469662
l :1:32: r :1:64	25593315	36747259	0.0013	83346	0.5092	1.12327375
l :1:32: r :2:32	25593315	36747259	0.0013	83346	0.4019	1.09881932
l :1:32: r :2:64	25593315	36747259	0.0013	83346	0.2582	**1.06606116**
l :1:32: r :4:32	25593315	36747259	0.0013	83346	0.2582	1.06605842

3. Go Benchmark

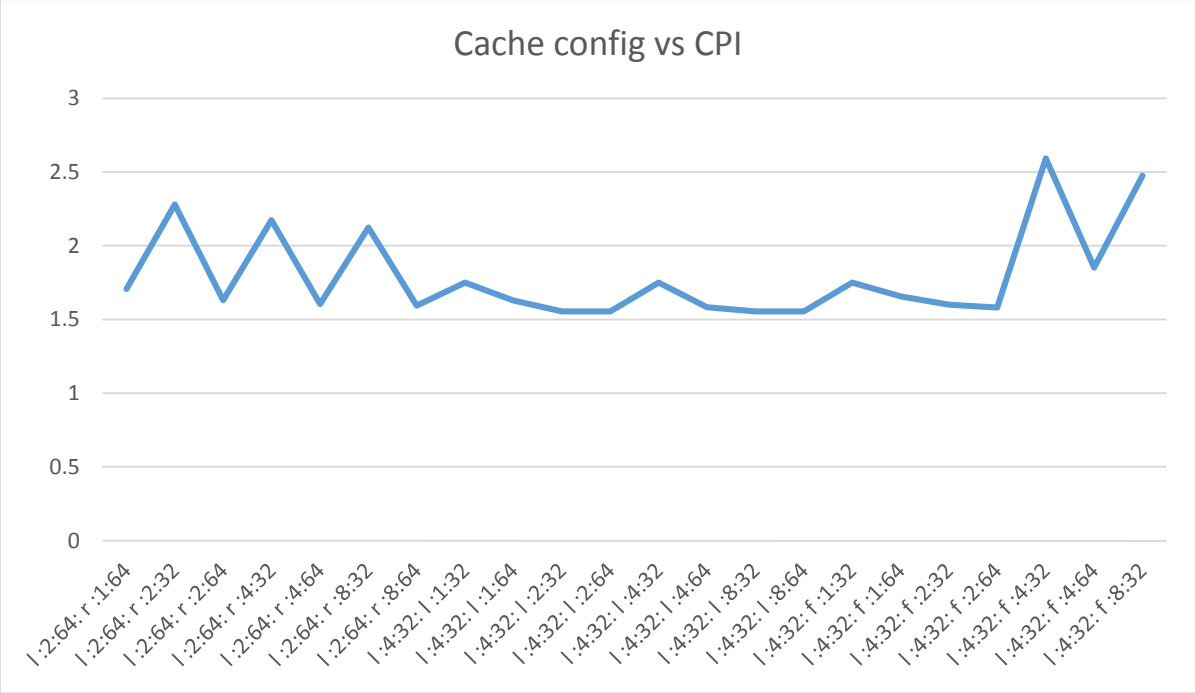
(a) L1 Separate data and instruction cache, L2 Unified data and instruction cache



The figure shows the CPI as a function of various configurations. The lowest value of CPI observed is **1.556971719** for the configuration **l: 1:64: l: 1:32** where L1 is 1-way and L2 is 1-way associative and replacement policy is LRU.

Cache config	sim_num_insn	dl1.access	dl1.miss_rate	il1.access	il1.miss_rate	dl2.access	dl2.miss_rate	CPI
l :1:32: l :4:64	718216	200572	0.0493	718216	0.0011	15896	0.3386	1.584178019
l :1:32: l :8:32	718216	200572	0.0493	718216	0.0011	15896	0.6585	2.079781013
l :1:32: l :8:64	718216	200572	0.0493	718216	0.0011	15896	0.3386	1.584178019
l :1:32: f :1:32	718216	200572	0.0493	718216	0.0011	15896	0.6593	2.080950578
l :1:32: f :1:64	718216	200572	0.0493	718216	0.0011	15896	0.3395	1.585542511
l :1:32: f :2:32	718216	200572	0.0493	718216	0.0011	15896	0.6585	2.079781013
l :1:32: f :2:64	718216	200572	0.0493	718216	0.0011	15896	0.3386	1.584178019
l :1:32: f :4:32	718216	200572	0.0493	718216	0.0011	15896	0.6585	2.079781013
l :1:32: f :4:64	718216	200572	0.0493	718216	0.0011	15896	0.3386	1.584178019
l :1:32: f :8:32	718216	200572	0.0493	718216	0.0011	15896	0.6585	2.079781013
l :1:32: f :8:64	718216	200572	0.0493	718216	0.0011	15896	0.3386	1.584178019
l :1:32: r :1:32	718216	200572	0.0493	718216	0.0011	15896	0.6593	2.080950578
l :1:32: r :1:64	718216	200572	0.0493	718216	0.0011	15896	0.3395	1.585542511
l :1:32: r :2:32	718216	200572	0.0493	718216	0.0011	15896	0.6588	2.080268332
l :1:32: r :2:64	718216	200572	0.0493	718216	0.0011	15896	0.3391	1.584957729
l :1:32: r :4:32	718216	200572	0.0493	718216	0.0011	15896	0.6585	2.079781013
l :1:32: r :4:64	718216	200572	0.0493	718216	0.0011	15896	0.3389	1.584567874
l :1:32: r :8:32	718216	200572	0.0493	718216	0.0011	15896	0.6585	2.079781013
l :1:32: r :8:64	718216	200572	0.0493	718216	0.0011	15896	0.3388	1.584372946
l :1:64: l :1:32	718216	200572	0.026	718216	0.0006	8407	0.6413	**1.556971719**
l :1:64: l :1:64	718216	200572	0.026	718216	0.0006	8407	0.6403	1.556192009

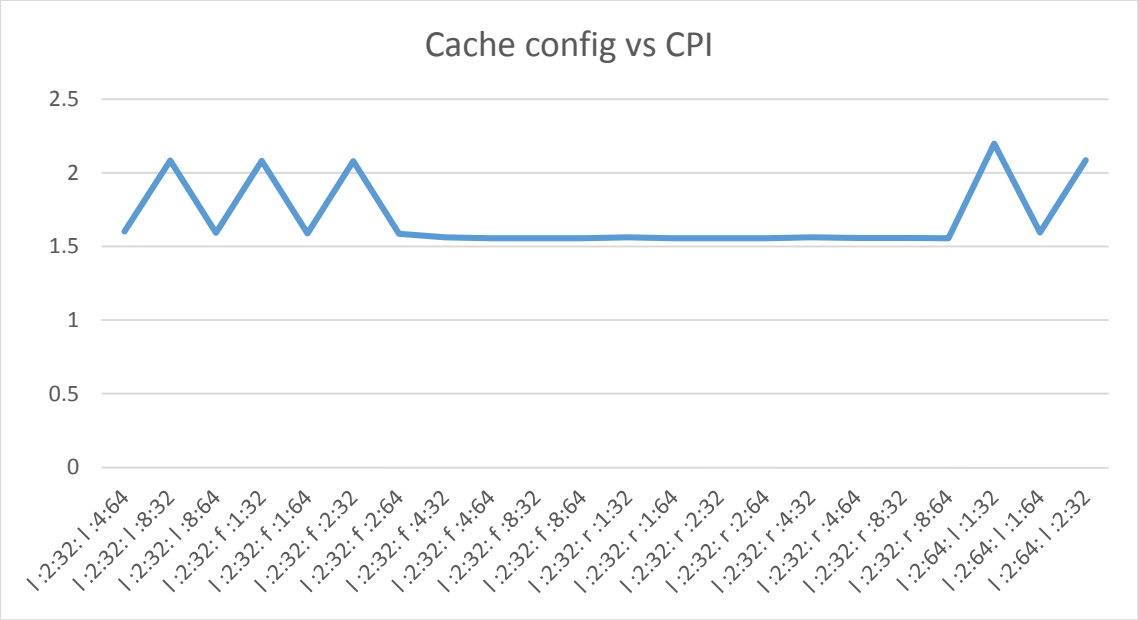
(b) L1 Separate data and instruction cache, L2 Separate data and instruction cache



The figure shows the CPI as a function of various configurations. The lowest value of CPI observed is **1.556971719** for the configuration **l: 1:64: l: 1:32** where L1 is 1-way and L2 is 1-way associative and replacement policy is LRU.

Cache config	sim_num_insn	dl1.access	dl1.miss_rate	il1.access	il1.miss_rate	dl2.access	dl2.miss_rate	CPI
l :1:32: l :4:64	718216	200572	0.0493	718216	0.0011	15896	0.3386	1.584178019
l :1:32: l :8:32	718216	200572	0.0493	718216	0.0011	15896	0.6585	2.079781013
l :1:32: l :8:64	718216	200572	0.0493	718216	0.0011	15896	0.3386	1.584178019
l :1:32: f :1:32	718216	200572	0.0493	718216	0.0011	15896	0.6593	2.080950578
l :1:32: f :1:64	718216	200572	0.0493	718216	0.0011	15896	0.3395	1.585542511
l :1:32: f :2:32	718216	200572	0.0493	718216	0.0011	15896	0.6585	2.079781013
l :1:32: f :2:64	718216	200572	0.0493	718216	0.0011	15896	0.3386	1.584178019
l :1:32: f :4:32	718216	200572	0.0493	718216	0.0011	15896	0.6585	2.079781013
l :1:32: f :4:64	718216	200572	0.0493	718216	0.0011	15896	0.3386	1.584178019
l :1:32: f :8:32	718216	200572	0.0493	718216	0.0011	15896	0.6585	2.079781013
l :1:32: f :8:64	718216	200572	0.0493	718216	0.0011	15896	0.3386	1.584178019
l :1:32: r :1:32	718216	200572	0.0493	718216	0.0011	15896	0.6593	2.080950578
l :1:32: r :1:64	718216	200572	0.0493	718216	0.0011	15896	0.3395	1.585542511
l :1:32: r :2:32	718216	200572	0.0493	718216	0.0011	15896	0.6588	2.080268332
l :1:32: r :2:64	718216	200572	0.0493	718216	0.0011	15896	0.3391	1.584957729
l :1:32: r :4:32	718216	200572	0.0493	718216	0.0011	15896	0.6585	2.079781013
l :1:32: r :4:64	718216	200572	0.0493	718216	0.0011	15896	0.3389	1.584567874
l :1:32: r :8:32	718216	200572	0.0493	718216	0.0011	15896	0.6585	2.079781013
l :1:32: r :8:64	718216	200572	0.0493	718216	0.0011	15896	0.3388	1.584372946
l :1:64: l :1:32	718216	200572	0.026	718216	0.0006	8407	0.6413	**1.556971719**
l :1:64: l :1:64	718216	200572	0.026	718216	0.0006	8407	0.6403	1.556192009

(c) L1 Unified data and instruction cache, L2 Unified data and instruction cache



The figure shows the CPI as a function of various configurations. The lowest value of CPI observed is **1.555002445** for the configuration **l: 2:32: f: 4:64** where L1 is 2-way and L2 is 4-way associative and replacement policy is FIFO.

Cache config	sim_num_insn	ul1.access	ul1.miss_rate	ul2.access	ul2.miss_rate	CPI
I :2:32: I :1:64	718216	918788	0.0115	12979	0.4147	1.583459572
I :2:32: I :2:32	718216	918788	0.0115	12979	0.8065	2.079062566
I :2:32: I :2:64	718216	918788	0.0115	12979	0.4147	1.583459572
I :2:32: I :4:32	718216	918788	0.0115	12979	0.8127	2.086859663
I :2:32: I :4:64	718216	918788	0.0115	12979	0.4284	1.60071065
I :2:32: I :8:32	718216	918788	0.0115	12979	0.8093	2.08257126
I :2:32: I :8:64	718216	918788	0.0115	12979	0.4208	1.591159206
I :2:32: f :1:32	718216	918788	0.0115	12979	0.8072	2.07993974
I :2:32: f :1:64	718216	918788	0.0115	12979	0.4177	1.587163193
I :2:32: f :2:32	718216	918788	0.0115	12979	0.8068	2.079452421
I :2:32: f :2:64	718216	918788	0.0115	12979	0.4159	1.584921528
I :2:32: f :4:32	718216	918788	0.0059	6744	0.8072	1.560967731
I :2:32: f :4:64	**718216**	**918788**	**0.0059**	**6744**	**0.7982**	**1.555022445**
I :2:32: f :8:32	718216	918788	0.0059	6744	0.7982	1.555022445
I :2:32: f :8:64	718216	918788	0.0059	6744	0.7982	1.555022445
I :2:32: r :1:32	718216	918788	0.0059	6744	0.8072	1.560967731
I :2:32: r :1:64	718216	918788	0.0059	6744	0.7982	1.555022445
I :2:32: r :2:32	718216	918788	0.0059	6744	0.7982	1.555022445
I :2:32: r :2:64	718216	918788	0.0059	6744	0.7982	1.555022445
I :2:32: r :4:32	718216	918788	0.0059	6744	0.8072	1.560967731
I :2:32: r :4:64	718216	918788	0.0059	6744	0.8023	1.557751429
I :2:32: r :8:32	718216	918788	0.0059	6744	0.8009	1.556776791
I :2:32: r :8:64	718216	918788	0.0059	6744	0.7995	1.555899618
I :2:64: I :1:32	718216	918788	0.0115	12326	0.9489	2.198948506
I :2:64: I :1:64	718216	918788	0.0115	12326	0.446	1.594770932
I :2:64: I :2:32	718216	918788	0.0115	12326	0.8549	2.085988059

(d) DEFINING A COST FUNCTION

Cost function plays a major role in determining Cache design. Cost of hardware increases rapidly if we choose parameters randomly for the least CPI. So In this part we have defined a cost function, in arbitrary cost units, using some of the parameters that determine cache design choice.

Cost function depends on the following parameters

- Performance
- Area overhead
  - Cache size

The performance depends on the following factors

1) Replacement policy - Increases with increase in the complexity and hardware

□□ LRU (l) = 20

□ FIFO (f) = 15

□□ Random(r) = 8

2) Set-Associativity –Cost increases with increase in associativity

□□ S’ = 2ln(S+1) where S is Set- associativity

8 way

□ 4 way

- 2 way
- 1 way
- Fully associative

3) Block size –Cost increases with increase in block size
- 32 bytes = 32
- 64 bytes = 64

4) CPI is obtained from tables from each cache design

5) Cache size of L1 –Cost increases with increase in cache size
- 128KB-25
- 256KB-50

6) Cache size of L2 –Cost increases with increase in cache size
- 512KB = 10
- 1MB = 20

The cost function is defined as:
Cost Function = cost of (performance factor + area overhead factor)

CF = [(costof(Replacement policy) + S'(Set-associativity) + cf(block size)) / CPI] + cost of L1 + cost of L2

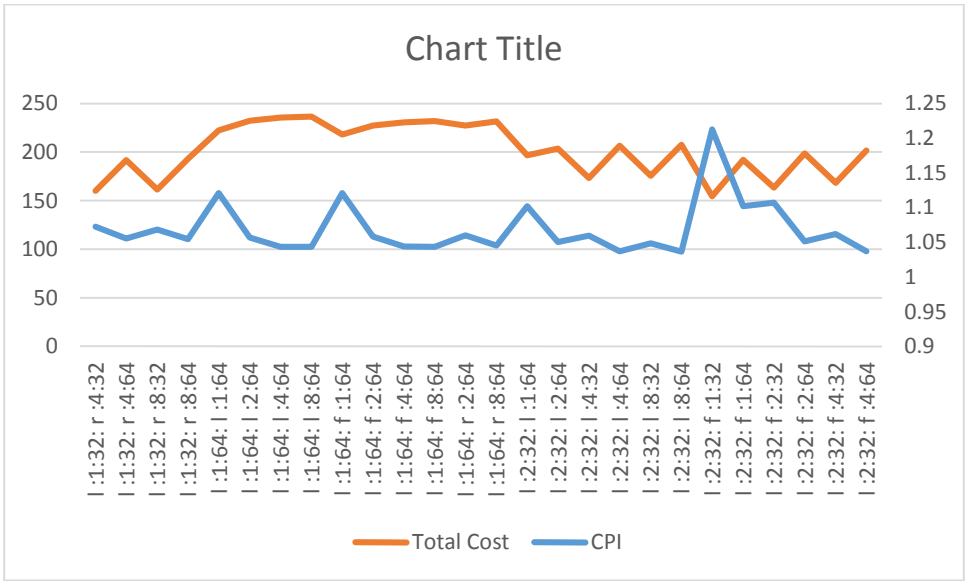
(e) OPTIMIZE CACHE
The cost function for all the cache design is obtained from plotting the Cache configuration vs CPI and Cache configuration VS Cost Function.
The Graph is plotted for the data given below. The optimal cost is obtained by comparing the minimal Cost Function with the minimum CPI.

1. **GCC Benchmark**
   (a)L1 Separate data and instruction cache, L2 Unified data and instruction cache
   L1cache-Replacement policy-LRU, Associativity-1-way, Size-32byte
   L2cache-Replacement policy-Random, Associativity-4-way, Size-32 byte

For the above configuration we got the optimum CPI (Cost Per Instruction)and Cost function

Cache config	CPI	Perf.Cost	Total Cost
I :1:32: r :4:32	1.072447479	90.07916195	160.0792
I :1:32: r :4:64	1.055347519	121.8604942	191.8605
I :1:32: r :8:32	1.068506379	91.5116142	161.5116
I :1:32: r :8:64	1.054257035	123.1016149	193.1016
I :1:64: I :1:64	1.120814095	152.3647762	222.3648
I :1:64: I :2:64	1.056730331	162.3720961	232.3721
I :1:64: I :4:64	1.043597411	165.3944025	235.3944
I :1:64: I :8:64	1.0432436	166.5773397	236.5773
I :1:64: f :1:64	1.120814095	147.9037331	217.9037
I :1:64: f :2:64	1.058189775	157.4231039	227.4231
I :1:64: f :4:64	1.043735407	160.5820489	230.582

l :1:64: f :8:64	1.0432436	161.7845952	231.7846
l :1:64: r :2:64	1.060123596	157.1359412	227.1359
l :1:64: r :8:64	1.045191323	161.4831081	231.4831
l :2:32: l :1:64	1.101773567	126.6898418	196.6898
l :2:32: l :2:64	1.050233178	133.6793125	203.6793
l :2:32: l :4:32	1.05935465	103.2856186	173.2856
l :2:32: l :4:64	1.036892121	136.3845839	206.3846
l :2:32: l :8:32	1.048632002	105.4628064	175.4628
l :2:32: l :8:64	**1.03653748**	**137.5653814**	207.5654
l :2:32: f :1:32	1.212846737	84.58077664	154.5808
l :2:32: f :1:64	1.101773567	122.1517043	192.1517
l :2:32: f :2:32	1.106946884	93.40506817	163.4051
l :2:32: f :2:64	1.051411233	128.7740181	198.774
l :2:32: f :4:32	1.061731305	98.34512734	168.3451
l :2:32: f :4:64	1.037031155	131.544843	201.5448



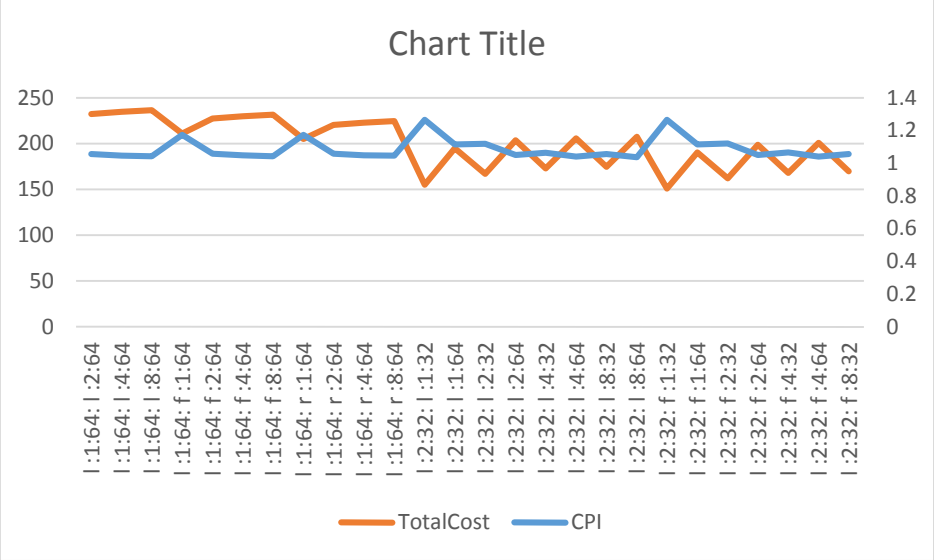
(b)L1 Separate data and instruction cache, L2 Separate data and instruction cache

L1cache-Replacement policy-LRU, Associativity-4-way, Size-32byte

L2cache-Replacement policy-FIFO, Associativity-2-way, Size-32 byte

For the above configuration we got the optimum CPI (Cost Per Instruction) and Cost function

Cache config	CPI	Perf.cost	TotalCost
I :1:64: I :2:64	1.056619104	162.3892	232.3892
I :1:64: I :4:64	1.047509044	164.7768	234.7768
I :1:64: I :8:64	1.043252523	166.5759	236.5759
I :1:64: f :1:64	1.174358859	141.1601	211.1601
I :1:64: f :2:64	1.05799658	157.4519	227.4519
I :1:64: f :4:64	1.047889832	159.9454	229.9454
I :1:64: f :8:64	1.043253768	161.783	231.783
I :1:64: r :1:64	1.174358859	135.1994	205.1994
I :1:64: r :2:64	1.059474078	150.6252	220.6252
I :1:64: r :4:64	1.049085111	153.0907	223.0907
I :1:64: r :8:64	1.046181786	154.6392	224.6392
I :2:32: I :1:32	1.265528037	85.01077	155.0108
I :2:32: I :1:64	1.116037022	125.0707	195.0707
I :2:32: I :2:32	1.119154501	96.85387	166.8539
I :2:32: I :2:64	1.050218444	133.6812	203.6812
I :2:32: I :4:32	1.064302193	102.8055	172.8055
I :2:32: I :4:64	1.040804792	135.8719	205.8719
I :2:32: I :8:32	1.055753042	104.7515	174.7515
I :2:32: I :8:64	**1.036546403**	137.5642	207.5642
I :2:32: f :1:32	1.265528037	81.05985	151.0599
I :2:32: f :1:64	1.116037022	120.5906	190.5906
I :2:32: f :2:32	1.121901567	92.16	162.16
I :2:32: f :2:64	1.051232771	128.7959	198.7959
I :2:32: f :4:32	1.066357201	97.9185	167.9185
I :2:32: f :4:64	1.041191182	131.0193	201.0193
I :2:32: f :8:32	1.056445931	99.94991	169.9499



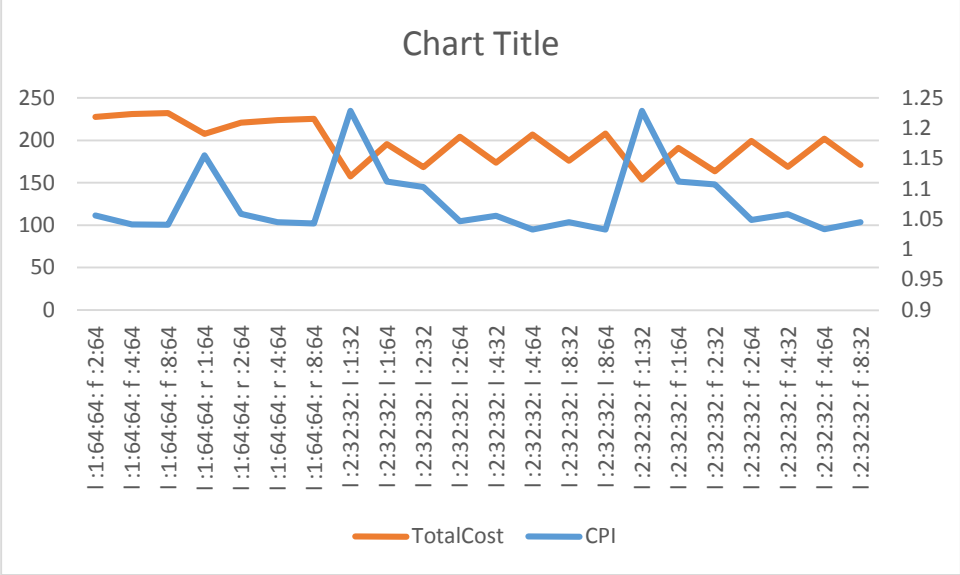
(c) L1 Unified data and instruction cache, L2 Unified data and instruction cache

L1cache-Replacement policy-LRU, Associativity-2-way, Size-32byte

L2cache-Replacement policy-FIFO, Associativity-4-way, Size-32 byte

For the above configuration we got the optimum CPI(Cost Per Instruction)and Cost function

Cache config	CPI	Perf.Cost	TotalCost
I :1:64:64: f :2:64	1.056137476	157.7290103	227.7290103
I :1:64:64: f :4:64	1.041195659	160.9737505	230.9737505
I :1:64:64: f :8:64	1.040696173	162.1806132	232.1806132
I :1:64:64: r :1:64	1.155211788	137.4402429	207.4402429
I :1:64:64: r :2:64	1.058428635	150.7739999	220.7739999
I :1:64:64: r :4:64	1.045086748	153.676401	223.676401
I :1:64:64: r :8:64	1.042709264	155.154221	225.154221
I :2:32:32: I :1:32	1.22870267	87.55862713	157.5586271
I :2:32:32: I :1:64	1.112033685	125.5209449	195.5209449
I :2:32:32: I :2:32	1.102909307	98.28047368	168.2804737
I :2:32:32: I :2:64	1.046727251	134.1270603	204.1270603
I :2:32:32: I :4:32	1.055319979	103.6804975	173.6804975
I :2:32:32: I :4:64	1.032988598	136.8999626	206.8999626
I :2:32:32: I :8:32	1.044728479	105.8568575	175.8568575
I :2:32:32: I :8:64	**1.032630637**	**138.0858447**	208.0858447
I :2:32:32: f :1:32	1.22870267	83.48929439	153.4892944
I :2:32:32: f :1:64	1.112033685	121.0246782	191.0246782
I :2:32:32: f :2:32	1.107059374	93.39557712	163.3955771
I :2:32:32: f :2:64	1.048329464	129.1525744	199.1525744
I :2:32:32: f :4:32	1.057909128	98.7004438	168.7004438
I :2:32:32: f :4:64	1.033135518	132.0408582	202.0408582
I :2:32:32: f :8:32	1.044842611	101.0598846	171.0598846



2. Anagram Benchmark

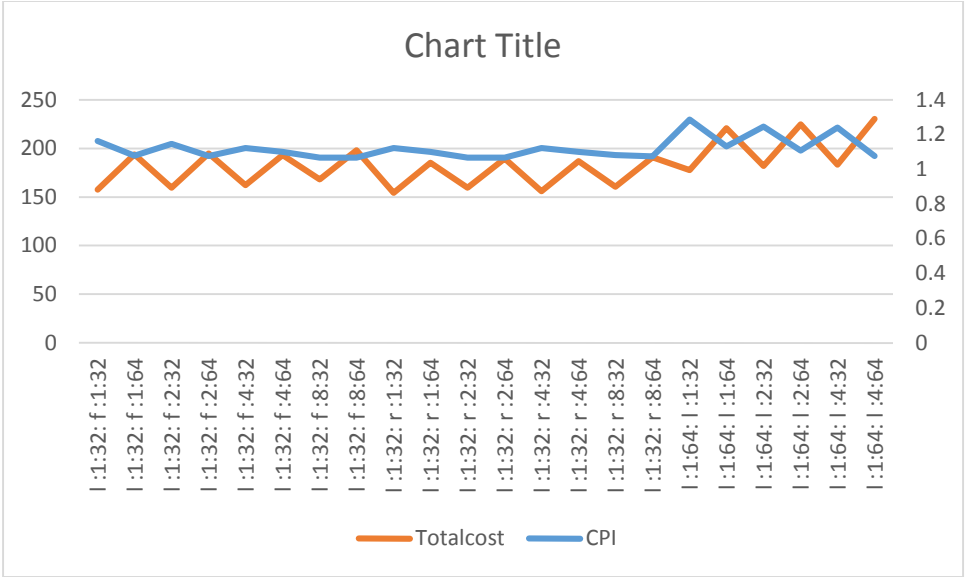
(a) L1 Separate data and instruction cache, L2 Unified data and instruction cache

L1cache-Replacement policy-LRU, Associativity-1-way, Size-32byte

L2cache-Replacement policy-Random, Associativity-2-way, Size-32 byte

For the above configuration we got the optimum CPI (Cost Per Instruction)and Cost function

Cache config	CPI	Perf.cost	Totalcost
l :1:32: f :1:32	1.162414209	87.55277	157.5528
l :1:32: f :1:64	1.079032279	123.9746	193.9746
l :1:32: f :2:32	1.146720267	89.45819	159.4582
l :1:32: f :2:64	1.077008313	124.9605	194.9605
l :1:32: f :4:32	1.123382219	92.22611	162.2261
l :1:32: f :4:64	1.099671731	123.3142	193.3142
l :1:32: f :8:32	1.067255453	98.17775	168.1778
l :1:32: f :8:64	1.067255453	128.1612	198.1612
l :1:32: r :1:32	1.123382219	84.36362	154.3636
l :1:32: r :1:64	1.099671731	115.2822	185.2822
l :1:32: r :2:32	1.067255453	89.56011	159.5601
l :1:32: r :2:64	**1.067255453**	119.5436	189.5436
l :1:32: r :4:32	1.123382219	85.99493	155.9949
l :1:32: r :4:64	1.099737373	116.9417	186.9417
l :1:32: r :8:32	1.081373984	90.42269	160.4227
l :1:32: r :8:64	1.073630946	120.8802	190.8802
l :1:64: l :1:32	1.28759049	107.777	177.777
l :1:64: l :1:64	1.131766987	150.8902	220.8902
l :1:64: l :2:32	1.245303275	112.088	182.088
l :1:64: l :2:64	1.107979916	154.8616	224.8616
l :1:64: l :4:32	1.240940808	113.3053	183.3053
l :1:64: l :4:64	1.075517142	160.4857	230.4857



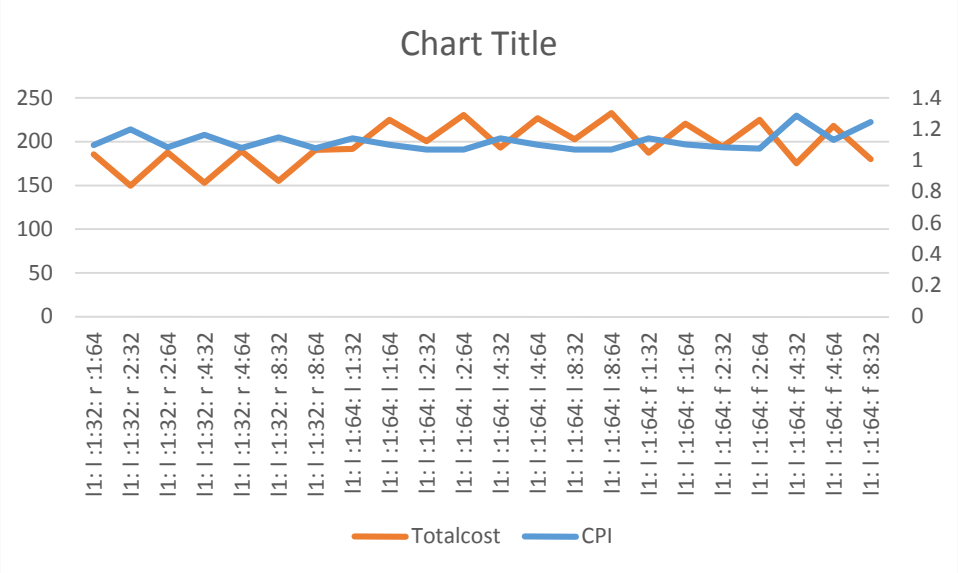
(b) L1 Separate data and instruction cache, L2 Separate data and instruction cache

L1cache-Replacement policy-LRU, Associativity-1-way, Size-32byte

L2cache-Replacement policy-Random, Associativity-4-way, Size-32 byte

For the above configuration we got the optimum CPI (Cost Per Instruction)and Cost function

Cache config	CPI	Perf.cost	Totalcost
I1: I :1:32: r :1:64	1.09794081	115.464	TotalCost
I1: I :1:32: r :2:32	1.198838251	79.73012	149.7301
I1: I :1:32: r :2:64	1.084065702	117.6898	187.6898
I1: I :1:32: r :4:32	1.16292106	83.07113	153.0711
I1: I :1:32: r :4:64	1.079295707	119.1566	189.1566
I1: I :1:32: r :8:32	1.146972754	85.25115	155.2511
I1: I :1:32: r :8:64	1.077312767	120.4671	190.4671
I1: I :1:64: I :1:32	1.141798981	121.5385	191.5385
I1: I :1:64: I :1:64	1.101150085	155.0857	225.0857
I1: I :1:64: I :2:32	**1.068766629**	130.6024	200.6024
I1: I :1:64: I :2:64	1.068766629	160.5435	230.5435
I1: I :1:64: I :4:32	1.141798981	123.1435	193.1435
I1: I :1:64: I :4:64	1.101144615	156.7507	226.7507
I1: I :1:64: I :8:32	1.068766629	132.6583	202.6583
I1: I :1:64: I :8:64	1.068766629	162.5993	232.5993
I1: I :1:64: f :1:32	1.141798981	117.1595	187.1595
I1: I :1:64: f :1:64	1.101502912	150.4967	220.4967
I1: I :1:64: f :2:32	1.082978153	124.2717	194.2717
I1: I :1:64: f :2:64	1.07523238	154.9279	224.9279
I1: I :1:64: f :4:32	1.286362982	105.4175	175.4175
I1: I :1:64: f :4:64	1.131682746	148.1026	218.1026
I1: I :1:64: f :8:32	1.245249121	109.8421	179.8421



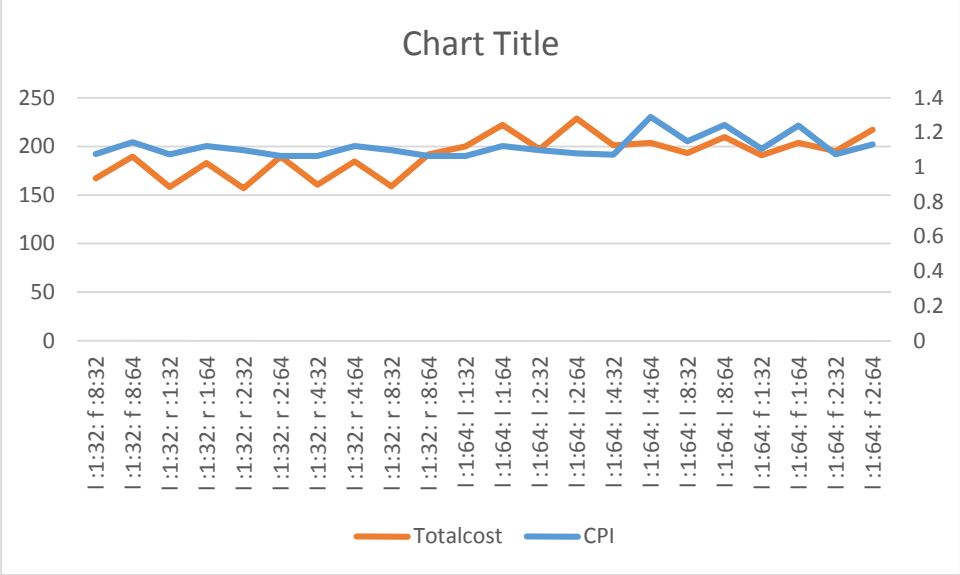
(c) L1 Unified data and instruction cache, L2 Unified data and instruction cache

L1cache-Replacement policy-LRU, Associativity-1-way, Size-32byte

L2cache-Replacement policy-Random, Associativity-1-way, Size-32 byte

For the above configuration we got the optimum CPI(Cost Per Instruction)and Cost function

Cache config	CPI	Perf.cost	Totalcost
I :1:32: f :8:32	1.076917508	97.29690787	167.2969079
I :1:32: f :8:64	1.144326517	119.5294713	189.5294713
I :1:32: r :1:32	1.074696615	88.18543521	158.1854352
I :1:32: r :1:64	1.123273753	112.8599225	182.8599225
I :1:32: r :2:32	1.098819321	86.98747569	156.9874757
I :1:32: r :2:64	**1.066061157**	119.6774858	189.6774858
I :1:32: r :4:32	1.066058422	90.61902069	160.6190207
I :1:32: r :4:64	1.123273753	114.4913872	184.4913872
I :1:32: r :8:32	1.098819321	88.98709886	158.9870989
I :1:32: r :8:64	1.066058422	121.7388661	191.7388661
I :1:64: I :1:32	1.066058422	130.1735307	200.1735307
I :1:64: I :1:64	1.123273753	152.0311395	222.0311395
I :1:64: I :2:32	1.098923254	127.0184414	197.0184414
I :1:64: I :2:64	1.080847049	158.7491209	228.7491209
I :1:64: I :4:32	1.072477676	131.1031206	201.1031206
I :1:64: I :4:64	1.290426152	133.7582704	203.7582704
I :1:64: I :8:32	1.149744103	123.315043	193.315043
I :1:64: I :8:64	1.245105724	139.5710743	209.5710743
I :1:64: f :1:32	1.106611746	120.8848444	190.8848444
I :1:64: f :1:64	1.239660161	133.7242205	203.7242205
I :1:64: f :2:32	1.073774265	125.3368826	195.3368826
I :1:64: f :2:64	1.13202893	147.1548248	217.1548248



3. Go Benchmark

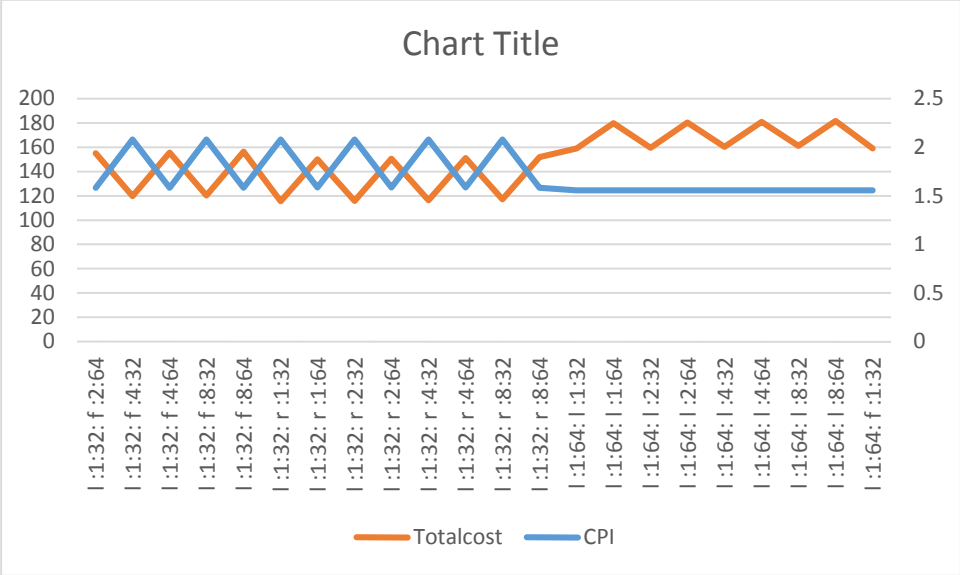
(a) L1 Separate data and instruction cache, L2 Unified data and instruction cache

L1cache-Replacement policy-LRU, Associativity-1-way, Size-32byte

L2cache-Replacement policy-Random, Associativity-1-way, Size-64 byte

For the above configuration we got the optimum CPI (Cost Per Instruction)and Cost function

Cache config	CPI	Perf.cost	Totalcost
I :1:32: f :2:64	1.584178019	84.9548	154.9548
I :1:32: f :4:32	2.079781013	49.81542	119.8154
I :1:32: f :4:64	1.584178019	85.5997	155.5997
I :1:32: f :8:32	2.079781013	50.38066	120.3807
I :1:32: f :8:64	1.584178019	86.34178	156.3418
I :1:32: r :1:32	2.080950578	45.54293	115.5429
I :1:32: r :1:64	1.585542511	79.95534	149.9553
I :1:32: r :2:32	2.080268332	45.94769	115.9477
I :1:32: r :2:64	1.584957729	80.49648	150.4965
I :1:32: r :4:32	2.079781013	46.44968	116.4497
I :1:32: r :4:64	1.584567874	81.16104	151.161
I :1:32: r :8:32	2.079781013	47.01492	117.0149
I :1:32: r :8:64	1.584372946	81.913	151.913
I :1:64: I :1:32	**1.556971719**	89.12981	159.1298
I :1:64: I :1:64	1.556192009	109.7375	179.7375
I :1:64: I :2:32	1.556192009	89.69556	159.6956
I :1:64: I :2:64	1.556192009	110.2586	180.2586
I :1:64: I :4:32	1.556971719	90.30682	160.3068
I :1:64: I :4:64	1.556192009	110.9151	180.9151
I :1:64: I :8:32	1.556192009	91.10749	161.1075
I :1:64: I :8:64	1.556192009	111.6705	181.6705
I :1:64: f :1:32	1.556971719	89.12981	159.1298



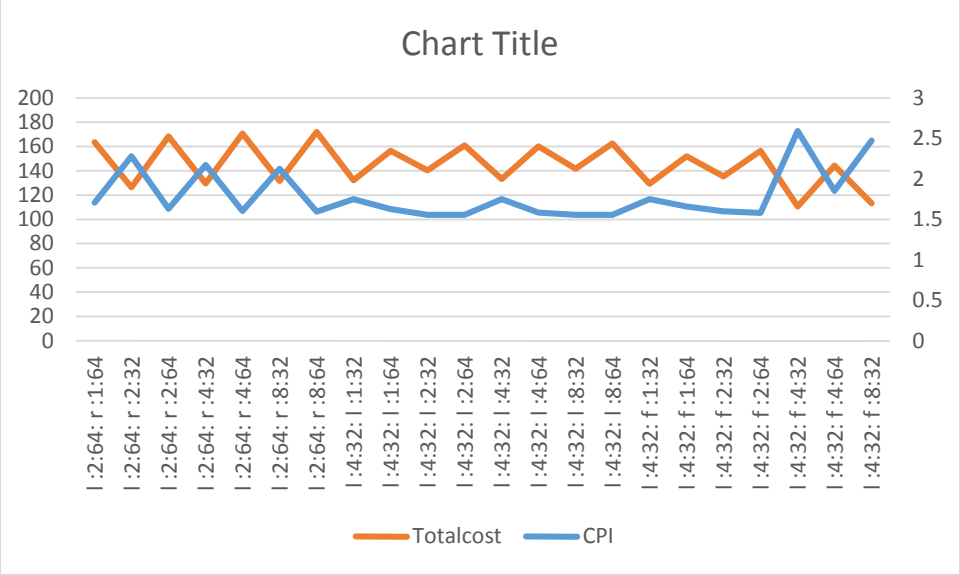
(b) L1 Separate data and instruction cache, L2 Separate data and instruction cache

L1cache-Replacement policy-LRU, Associativity-4-way, Size-32byte

L2cache-Replacement policy-LRU, Associativity-8-way, Size-32 byte

For the above configuration we got the optimum CPI (Cost Per Instruction) and Cost function

Cache config	CPI	Perf.cost	Totalcost
I :2:64: r :1:64	1.705718057	93.55797	163.558
I :2:64: r :2:32	2.278804705	56.34289	126.3429
I :2:64: r :2:64	1.631353242	98.31988	168.3199
I :2:64: r :4:32	2.172764182	59.56288	129.5629
I :2:64: r :4:64	1.604648184	100.5928	170.5928
I :2:64: r :8:32	2.122960224	61.51395	131.514
I :2:64: r :8:64	1.593927175	102.007	172.007
I :4:32: I :1:32	1.749724317	62.06988	132.0699
I :4:32: I :1:64	1.627992136	86.36723	156.3672
I :4:32: I :2:32	**1.555089277**	70.36001	140.36
I :4:32: I :2:64	1.555089277	90.93761	160.9376
I :4:32: I :4:32	1.749724317	63.11723	133.1172
I :4:32: I :4:64	1.582963899	89.98168	159.9817
I :4:32: I :8:32	1.555089277	71.77294	141.7729
I :4:32: I :8:64	1.555089277	92.35053	162.3505
I :4:32: f :1:32	1.749724317	59.21228	129.2123
I :4:32: f :1:64	1.657426178	81.81672	151.8167
I :4:32: f :2:32	1.599630195	65.27515	135.2751
I :4:32: f :2:64	1.579845061	86.34777	156.3478
I :4:32: f :4:32	2.592551544	40.66949	110.6695
I :4:32: f :4:64	1.851437451	74.233	144.233
I :4:32: f :8:32	2.474522985	43.08439	113.0844



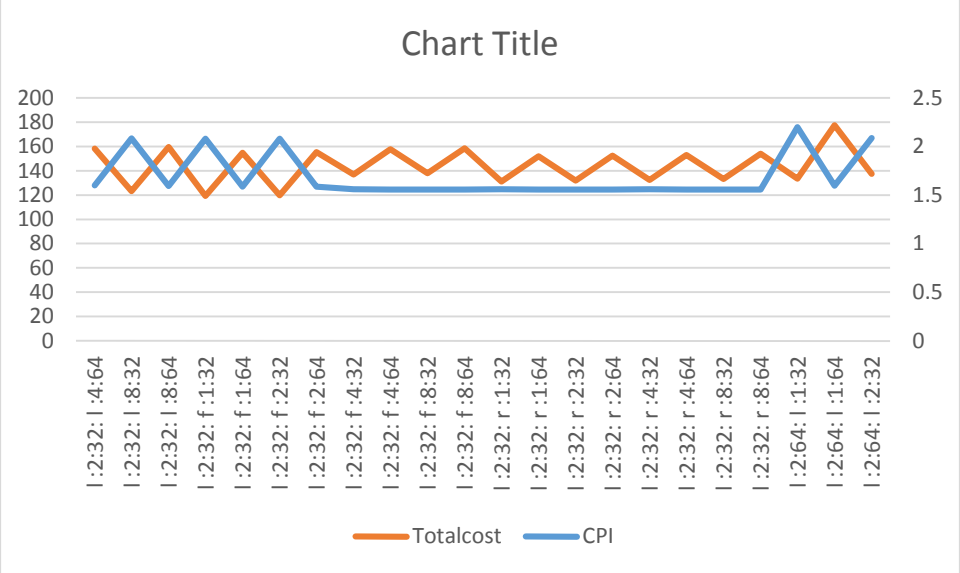
(c) L1 Unified data and instruction cache, L2 Unified data and instruction cache

L1cache-Replacement policy-LRU, Associativity-2-way, Size-32byte

L2cache-Replacement policy-Random, Associativity-4-way, Size-32 byte

For the above configuration we got the optimum CPI(Cost Per Instruction)and Cost function

Cache config	CPI	Perf.cost	Totalcost
I :2:32: I :4:64	1.60071065	88.34582	158.3458
I :2:32: I :8:32	2.08257126	53.10343	123.1034
I :2:32: I :8:64	1.591159206	89.61496	159.615
I :2:32: f :1:32	2.07993974	49.32043	119.3204
I :2:32: f :1:64	1.587163193	84.79501	154.795
I :2:32: f :2:32	2.079452421	49.72196	119.722
I :2:32: f :2:64	1.584921528	85.42659	155.4266
I :2:32: f :4:32	1.560967731	66.8919	136.8919
I :2:32: f :4:64	**1.555022445**	87.72613	157.7261
I :2:32: f :8:32	1.555022445	67.90363	137.9036
I :2:32: f :8:64	1.555022445	88.48211	158.4821
I :2:32: r :1:32	1.560967731	61.2335	131.2335
I :2:32: r :1:64	1.555022445	82.04609	152.0461
I :2:32: r :2:32	1.555022445	61.9891	131.9891
I :2:32: r :2:64	1.555022445	82.56759	152.5676
I :2:32: r :4:32	1.560967731	62.4075	132.4075
I :2:32: r :4:64	1.557751429	83.07879	153.0788
I :2:32: r :8:32	1.556776791	63.33064	133.3306
I :2:32: r :8:64	1.555899618	83.93323	153.9332
I :2:64: I :1:32	2.198948506	63.47739	133.4774
I :2:64: I :1:64	1.594770932	107.5913	177.5913
I :2:64: I :2:32	2.085988059	67.30357	137.3036



5. CONCLUSION

In this project, analysis is made to calculate the optimal configuration for the cache design based on the simulations of the different benchmarks and CPI for these benchmarks was calculated using the formula.

The cost function was defined over percentage change in the parameters and not involving the real time actual values for the cost. The optimal cache configuration for each benchmark, and the optimal configuration for all benchmarks (in terms of the average CPI) was determined. Graphs showing the trade-off between CPI and cost for different design choices was shown.