Advanced Computer Architecture
Assignment - 1 Report

Group ID - 9

Group Members :

1. **Rohit Arun Katariya - 20111050 -** rohitk20@iitk.ac.in
2. **Naga Durga Krishna Mohan Eaty - 20111037 -** kmohan20@iitk.ac.in

Introduction : Using the trace files of L1 cache misses from six application, L2 and L3 Caches are simulated for three different cases A) Inclusive B) Exclusive and C) NINE caches. Also, we simulated L2 and L3 fully associative caches and classified L3 Misses into Cold, Conflict and Capacity Misses. The replacement policy that is implemented in all these caches is LRU(Least Recently Used) policy.

Theory:

Inclusive Cache : The contents of L2 Cache should be a subset of L3 Cache. Eviction from L3 must invalidate the block in L2 cache. For a cold miss in L3 cache, we write into both L3 and L2 caches. Eviction from L2 cache writes the block into L3 cache.

Exclusive Cache : There should not be any intersection between the contents in L2 and L3 caches. When there is a miss in L3 cache, we first write into L3 using replacement policy after which we also write into L2 cache. When we have a miss in L2 and hit in L3 cache, we copy the block into L2 and invalidate the block in L3. Eviction of block from L2 writes back into L3 cache.

NINE Cache : Contents of L2 cache are not a subset of L3 cache and there is intersection between contents in L2 and L3 caches. When there is a miss in L3 cache, we first write into L3 using replacement policy after which we also write into L2 cache. Eviction from L3 does not require an eviction from L2. Similarly, eviction from L2 does not require any operation on L3.

Temporal Locality : Temporal Locality means that an instruction which is recently executed has high chances of execution again. So the instruction is kept in cache, such that it can be fetched easily and takes no time in searching for the same instruction.

Spatial Locality : Spatial Locality means that all those instructions which are stored nearby to the recently executed instruction have high chances of execution. It refers to the use of data elements(instructions) which are relatively close in storage locations.

<u>Problem 1 Results</u>:

Number of Hits and Misses in Inclusive L2 and L3 Caches

INCLUSIVE CACHE				
Application	**L2**		**L3**	
	Hits	**Misses**	**Hits**	**Misses**
bzip2	10628136	29491	17458	12033
gcc	14190701	420110	194447	225663
gromacs	3417117	14394	3246	11148
h264ref	2339836	8737	441	8296
hmmer	3501947	7818	81	7737
sphinx	10323485	429962	414304	15658

Number of Hits and Misses in NINE L2 and L3 Caches

NINE CACHE				
Application	**L2**		**L3**	
	Hits	**Misses**	**Hits**	**Misses**
bzip2	10628136	29491	17458	12033
gcc	14194082	416729	223522	193207
gromacs	3417117	14394	3246	11148
h264ref	2339836	8737	441	8296
hmmer	3501947	7818	81	7737
sphinx	10323485	429962	414304	15658

Number of Hits and Misses in Exclusive L2 and L3 Caches

EXCLUSIVE CACHE				
Application	**L2**		**L3**	
	Hits	**Misses**	**Hits**	**Misses**
bzip2	10628136	29491	17458	12033
gcc	14194082	416729	237544	179185
gromacs	3417117	14394	3246	11148
h264ref	2339836	8737	441	8296
hmmer	3501947	7818	81	7737
sphinx	10323485	429962	414304	15658

Analysing the Simulation:

- Here , we have implemented a Cache hierarchy with two cache levels (L2 is level 1, L3 is level 2). Following the given attributes of L2 and L3.

L2 : Cache size = 512 KB
8 - way set associative
64 Bytes block
Tag = 48 bits
Set = 10 bits
Blk Off. = 6 bits

L3 : Cache Size = 2 MB
16 - way set associative
64 Bytes Block
Tag = 47 bits
Set = 11 bits
Blk Off. = 6 bits

- **Cache misses (Inclusive) ~ Cache misses (NINE) ~ Cache misses (Exclusive)** (Except for gcc application)

As mentioned, for inclusive policy, we perform back invalidation for a particular block in L2 which was evicted from L3. We observe that there are no back invalidations performed for applications except **gcc .** This is the reason why all three policies are having almost similar number of cache misses. The only major difference between Inclusive and NINE policy is that there is no back validation required for NINE. So, in case there is no back invalidation done, inclusive and NINE policies gave a similar number of Cache misses.

- **Good Temporal and Spatial Locality**:

SInce we can see that , the number of cache miss values that we have obtained are quite less as compared to the number of total accesses, we can conclude that almost for all the application, the generated addresses resulted in good temporal and spatial locality. That means the same byte addresses are again being accessed in the near future. Also good spatial locality and high cache hit ratio imply that addresses belonging to the same block were accessed close to each other.

<u>Problem 2 Results</u> :

Classification Misses in Inclusive L3 Caches

INCLUSIVE				
Application	**L3 Misses**			
	Total Misses	**Cold Misses**	**Conflict Misses**	**Capacity Misses**
bzip2	12033	2048	9985	0
gcc	225663	2048	223615	192895
gromacs	11148	2048	9100	0
h264ref	8296	2047	6249	0
hmmer	7737	2048	5689	0
sphinx	15658	2048	13610	0

Fully Associative Cache : The simulator of fully associative L2 and L3 caches is done and the code is put in the zip. The run time for this is very high, so we could not run Compute misses for fully associative caches.