CS622:Assignment-1

Jayesh Patidar
20111025
jaysptdr20@iitk.ac.in

Parth Sharma
20111039
prthshrma20@iitk.ac.in

Problem 1

Solution

File : *bzip2.log_l1misstrace*

bzip	Inclusive	NINE	Exclusive
L2 access	10657627	10657627	10657627
L2 hit	5259461	5260051	2242279
L2 miss	5398166	5397576	8415346
L3 hit	3951778	3951730	338640
L3 miss	1446388	1445846	8076706

File : *gcc.log_l1misstrace*

gcc	Inclusive	NINE	Exclusive
L2 access	14610811	14610811	14610811
L2 hit	11574350	11581002	10545834
L2 miss	3036461	3029809	4664977
L3 hit	1663059	1663561	427410
L3 miss	1373402	1366248	3637567

File : *gromacs.log_l1misstrace*

gromacs	Inclusive	NINE	Exclusive
L2 access	3431511	3431511	3431511
L2 hit	3094660	3094787	1796730
L2 miss	336851	336724	1634781
L3 hit	166320	166265	458210
L3 miss	170531	170459	1176571

File : *h264ref.log_l1misstrace*

h264ref	Inclusive	NINE	Exclusive
L2 access	2348573	2348573	2348573
L2 hit	1378895	1382949	1157845
L2 miss	969678	965624	1188728
L3 hit	627532	632041	295511
L3 miss	342146	333583	893217

File : *hmmer.log_l1misstrace*

hmmer	Inclusive	NINE	Exclusive
L2 access	3509765	3509765	3509765
L2 hit	1766344	1774443	1662251
L2 miss	1743421	1735322	1847514
L3 hit	1352195	1358978	516273
L3 miss	391226	376344	1331241

File : *sphinx3.log_l1misstrace*

sphinx3	Inclusive	NINE	Exclusive
L2 access	10753447	10753447	10753447
L2 hit	1933098	1938317	821214
L2 miss	8820349	8815130	9932233
L3 hit	612987	609986	2577431
L3 miss	8207362	8205144	7354802

We implemented all this above caches to simulate the cache and count the cache Misses(i.e Cold misses ,Conflict Misses and Capacity Misses).

Firstly We Implemented Inclusion Policy ,In this particular implementation we have condition such that blocks present in L2 cache should always be Subset of blocks present in L3 cache(i.e. L2$\subseteq$ L3).To follow the above condition,if there is Miss in L2 and hit L3 then we need to put this L3 cache block in L2 cache.And if there is miss in L2 and L3 too then we need to take missed block in both L2 and L3 caches.Also we should remember that the victim block should be removed from L2 cache(Back Invalidation).

For implementation of NINE policy as name suggests we have condition Not Inclusion Not Exclusion.To follow above condition,if there is Miss in L2 cache and hit in L3 cache then we have to take the block present in L3 to L2 cache.And if the miss in both cache L2 and L3 then we need to take the Missed block in both caches.Unlike Inclusion Policy,We does not have need to remove the Victim block from L2 cache.

Now for the last Exclusive Implementation policy we have condition such that the Blocks present in L2 cache should not present in L3 cache.To follow up the above implementation,if there is miss in L2 cache and hit in L3 cache we need to take the block to L2 cache and invalidate the block present in L3.And if the block is evicted from L2 cache then this block must stored to L3 cache(if any).Now if,Block is missed from both cache then we need to take the block to L2 and write the evicted block to L3(if any).

Observation:

- Inclusion Policy is suffers bottleneck as Lower Level cache is subset of Higher Level Cache.
- Exclusion Policy may require more Bandwidth.As Every block evicted from L2 cache is Written to L3 cache.
- Overall we can say that Exclusion Policy increase the Capacity because there is no Common block present in both Cache at the same time(i.e. L2$\bigcap$L3=ϕ).

Problem 2

Solution

In this problem,We need to calculate the type of Misses

Cold Miss/Compulsory Miss: This is the Miss when we access the block for the first time.As name clearly indicates that the first time accessed block is always going to be miss here.

Conflict Miss: This miss Comes when there is Conflict in the Cache for Particular Set.One should need to be very careful in the calculating of conflict miss.To calculate the conflict miss one should simulate the Fully Associative cache to know the miss is of what type.If the Fully associative cache got hit then the miss in Set Associative cache is Conflict Miss or the capacity of fully associative cache is full then we can say that Miss would be Capacity Miss.

Capacity Miss: We can say the Miss to be Capacity miss when the Miss comes due to Capacity of Cache.

We Calculated Cold,Conflict and Capacity Misses with Implementation of Inclusion Policy.

Total Misses= Cold Miss + Conflict Miss + Capacity Miss

Conflict Miss=0 (in case of Fully associative)

Conflict Miss= Total Miss-Capacity Miss(in Fully Associative)

Fully Associative Cache implemented using LRU replacement policy

files	cold misses	conflict misses	capacity misses	total misses
bzip2	119753	84987	1241648	1446388
gcc	773053	3478	596871	1373402
gromacs	773053	1163	61406	170531
h264ref	63703	6266	272177	342146
hmmer	75884	14202	301140	391266
sphinx3	122069	-179886	8265179	8207362