Assignment 4
SMPCache - Locality of different programs
Student: Chao Chen

1. Introduction

The purpose of this project is to practice the knowledge of computer cache system with SMPCache program. We will conduct several experiments for different benchmark programs on an unique virtual machine(architecture), then verify and analysis the cache locality status.

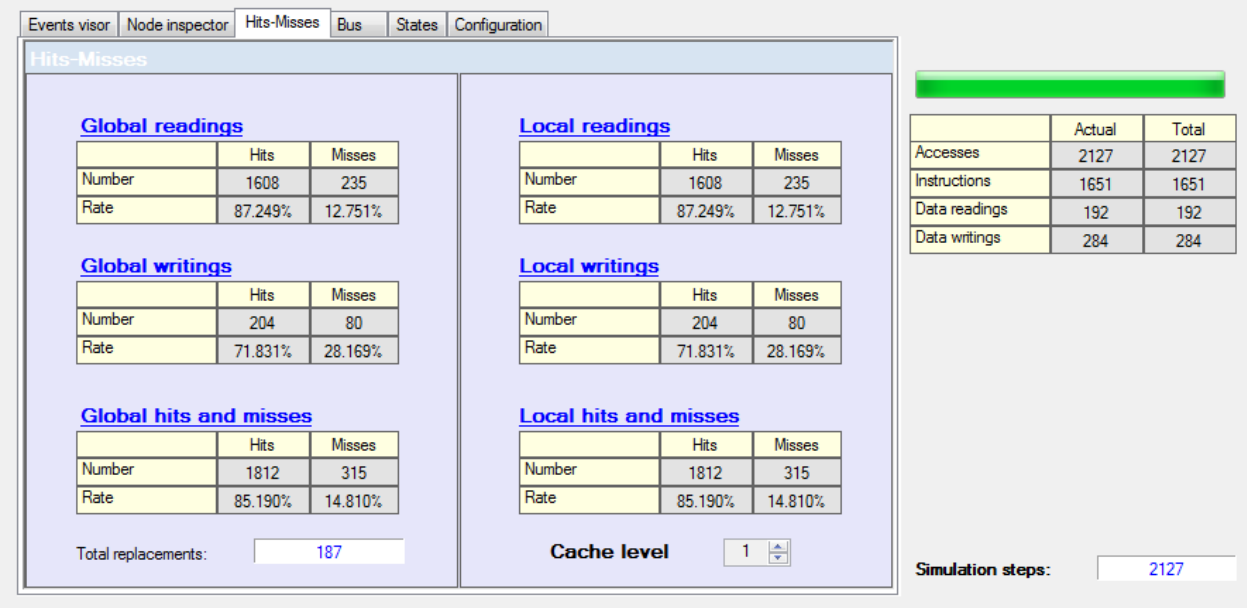
2. Experiments & Results

System configuration:

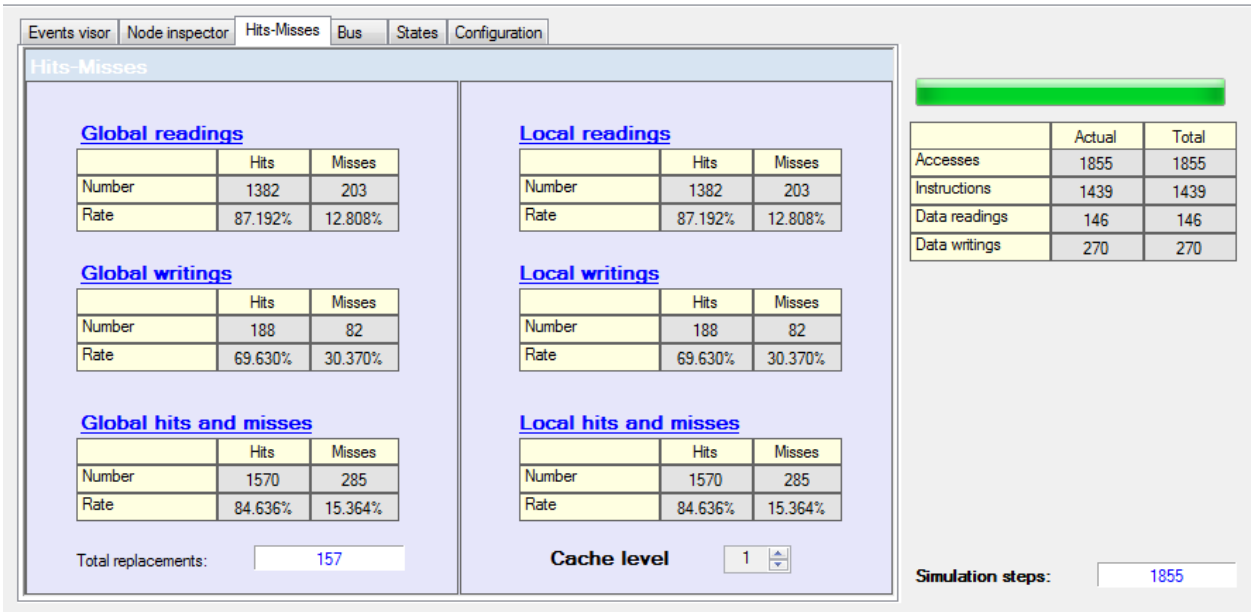
- Processors in SMP = 1
- Cache coherence protocol = MESI.
- Scheme for bus arbitration = Random.
- Word wide (bits) = 16.
- Words by block = 16 (block size = 32 bytes).
- Blocks in main memory = 8192 (main memory size = 256 KB).
- Blocks in cache = 128 (cache size = 4 KB).
- Mapping = Fully-Associative.
- Replacement policy = LRU.

The following figures show the result of cache missing/hit rate for each benchmark:

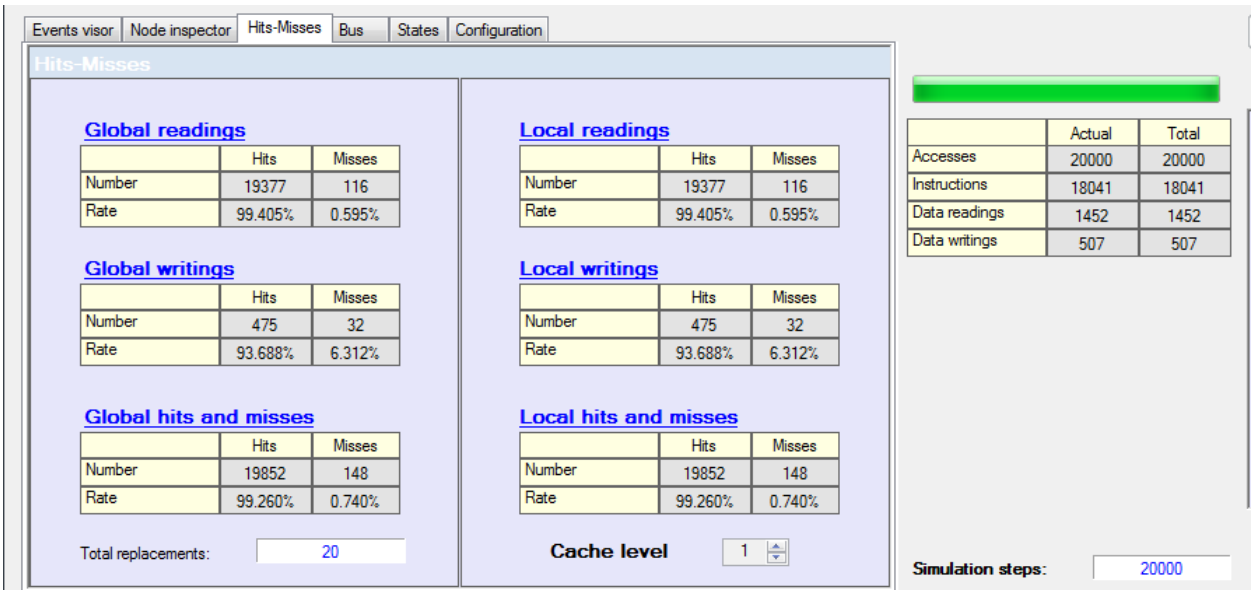
a). Hydro



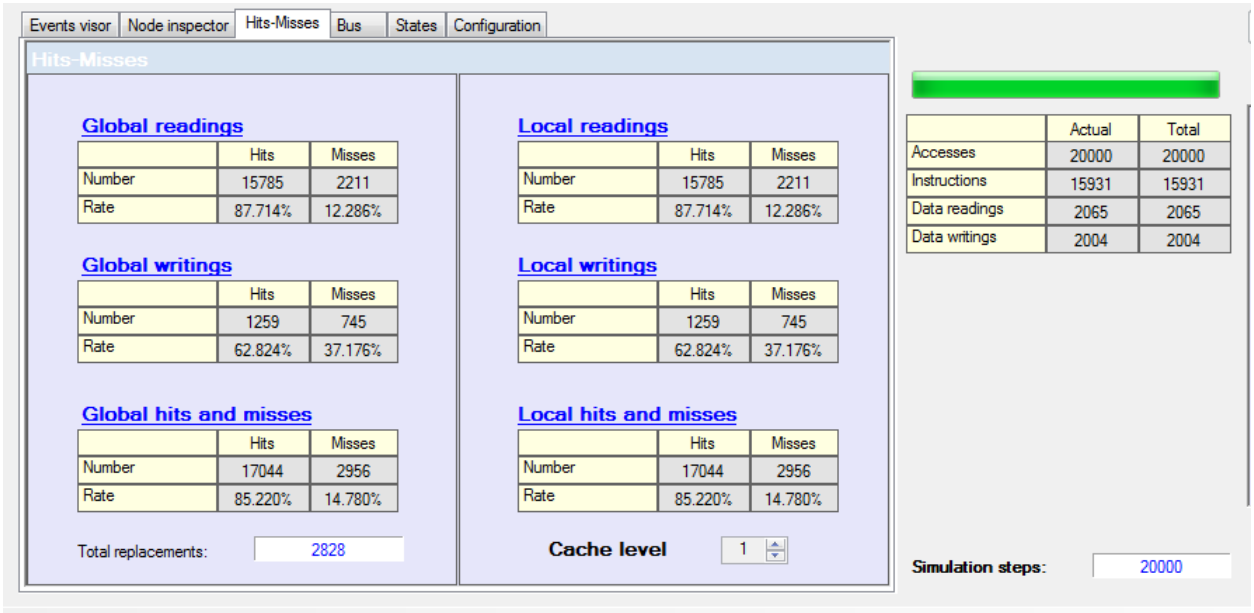
b). Nasa7



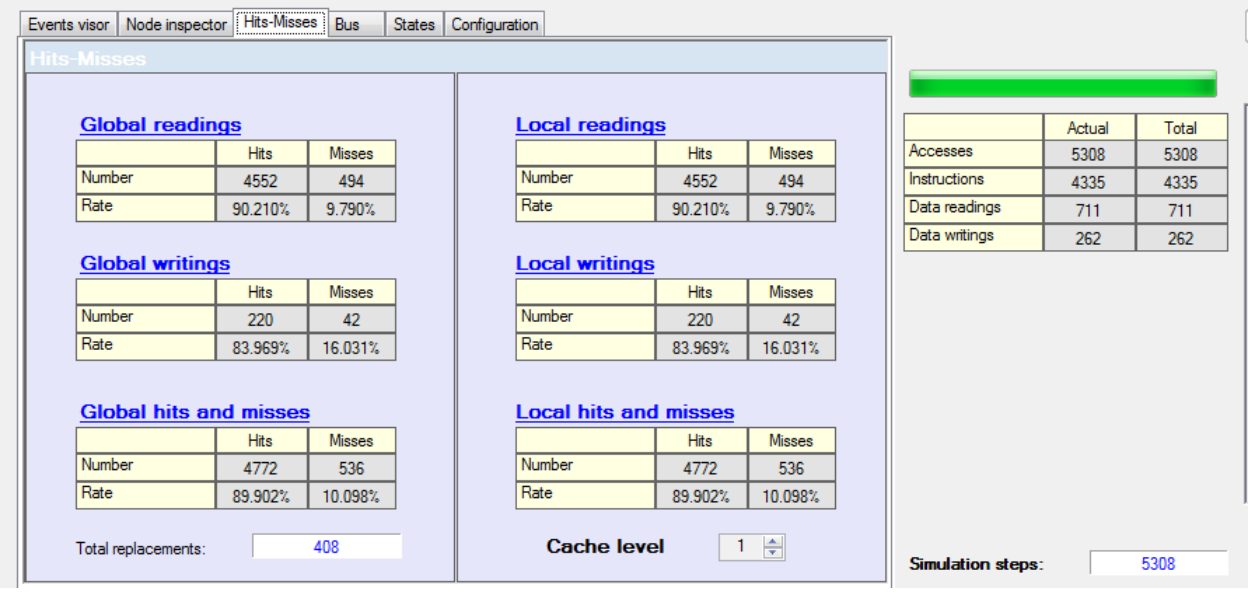
c). Cexp



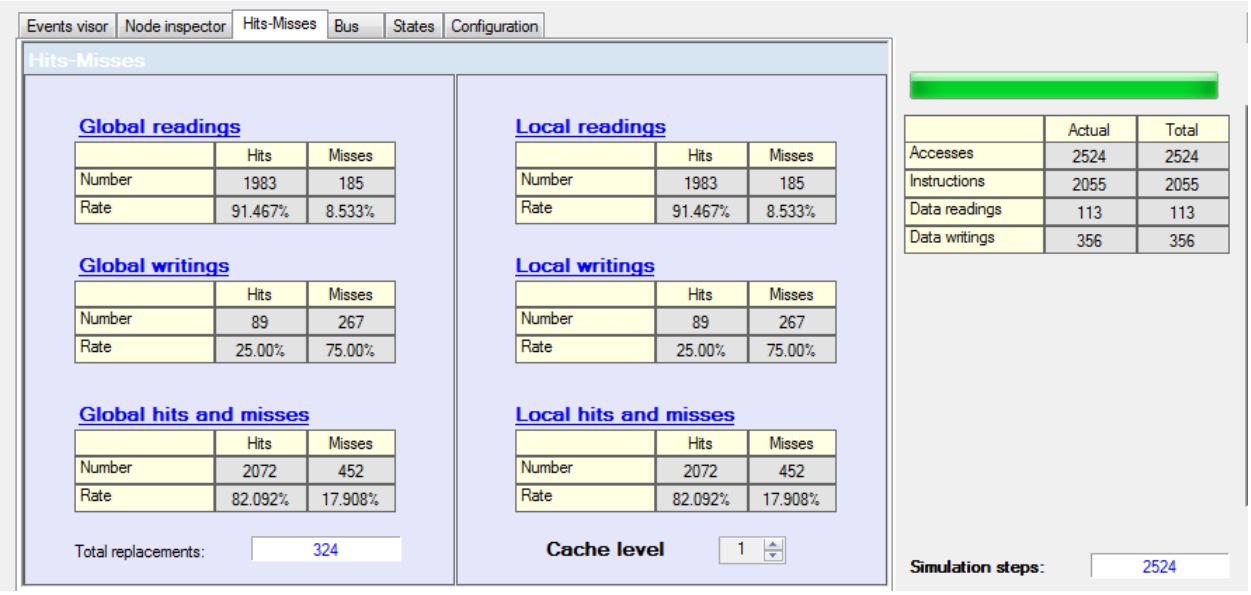
d). Mdljd



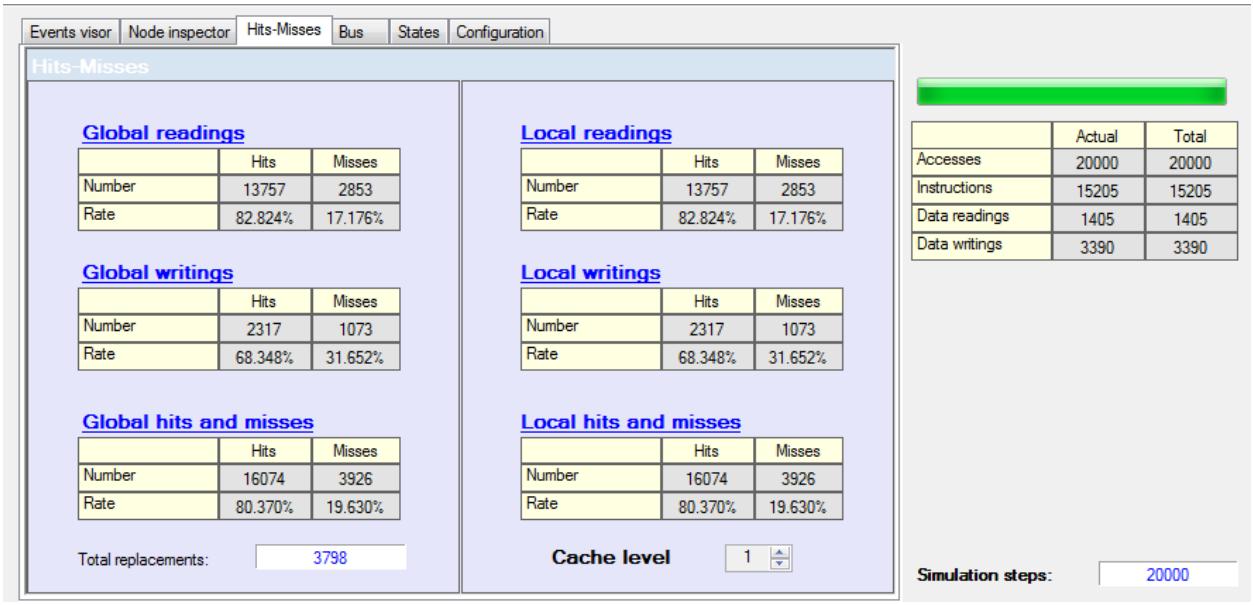
e). Ear


Events visor
Node inspector
Hits-Misses
Bus
States
Configuration
Hits-Misses
Global readings
Hits Misses
Number 4552 494
Rate 90.210% 9.790%
Global writings
Hits Misses
Number 220 42
Rate 83.969% 16.031%
Global hits and misses
Hits Misses
Number 4772 536
Rate 89.902% 10.098%
Total replacements: 408
Local readings
Hits Misses
Number 4552 494
Rate 90.210% 9.790%
Local writings
Hits Misses
Number 220 42
Rate 83.969% 16.031%
Local hits and misses
Hits Misses
Number 4772 536
Rate 89.902% 10.098%
Cache level 1
Actual Total
Accesses 5308 5308
Instructions 4335 4335
Data readings 711 711
Data writings 262 262
Simulation steps: 5308

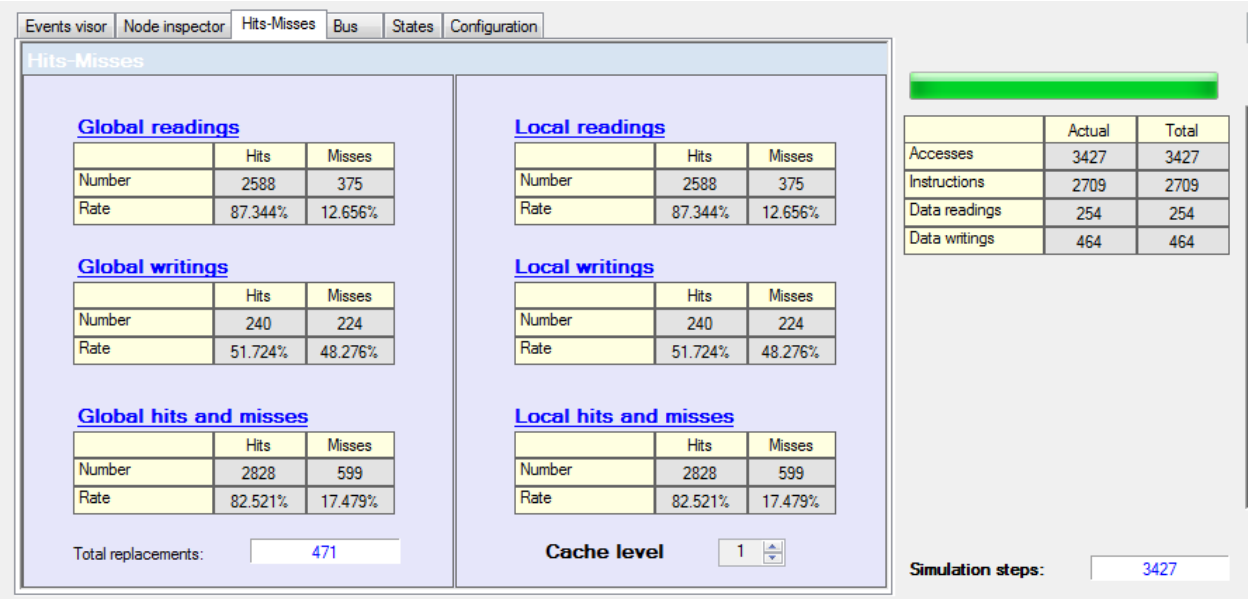
f). Comp


Events visor
Node inspector
Hits-Misses
Bus
States
Configuration
Hits-Misses
Global readings
Hits Misses
Number 1983 185
Rate 91.467% 8.533%
Global writings
Hits Misses
Number 89 267
Rate 25.00% 75.00%
Global hits and misses
Hits Misses
Number 2072 452
Rate 82.092% 17.908%
Total replacements: 324
Local readings
Hits Misses
Number 1983 185
Rate 91.467% 8.533%
Local writings
Hits Misses
Number 89 267
Rate 25.00% 75.00%
Local hits and misses
Hits Misses
Number 2072 452
Rate 82.092% 17.908%
Cache level 1
Actual Total
Accesses 2524 2524
Instructions 2055 2055
Data readings 113 113
Data writings 356 356
Simulation steps: 2524

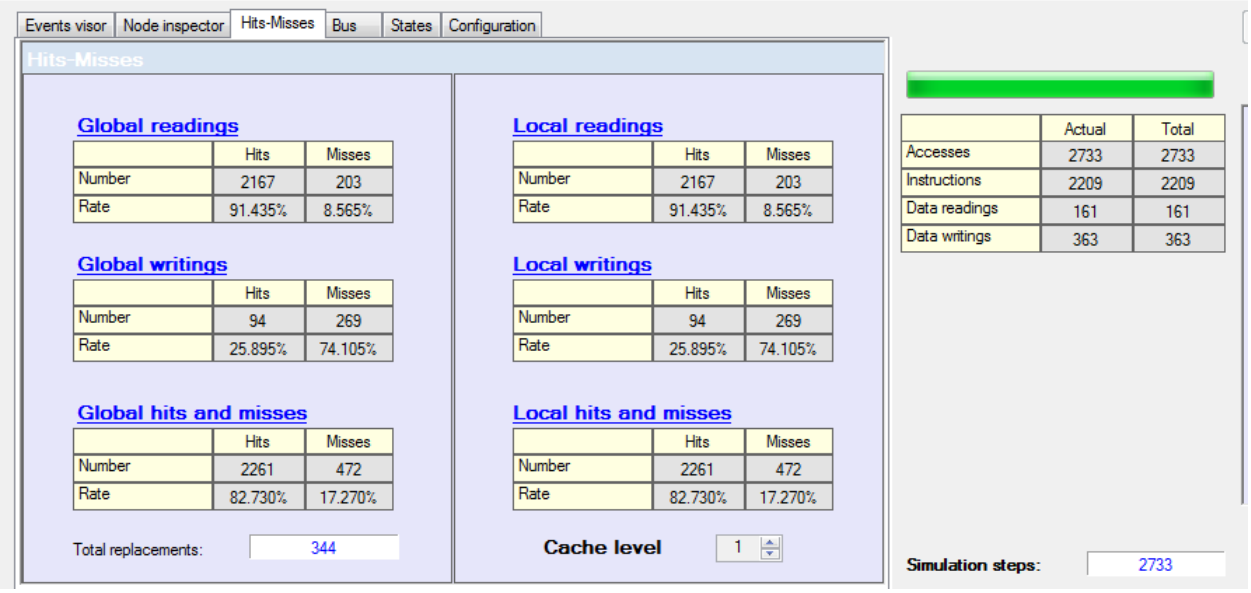
g). Swm


Events visor
Node inspector
Hits-Misses
Bus
States
Configuration
Hits-Misses
Global readings
Hits Misses
Number 13757 2853
Rate 82.824% 17.176%
Local readings
Hits Misses
Number 13757 2853
Rate 82.824% 17.176%
Global writings
Hits Misses
Number 2317 1073
Rate 68.348% 31.652%
Local writings
Hits Misses
Number 2317 1073
Rate 68.348% 31.652%
Global hits and misses
Hits Misses
Number 16074 3926
Rate 80.370% 19.630%
Local hits and misses
Hits Misses
Number 16074 3926
Rate 80.370% 19.630%
Total replacements: 3798
Cache level 1
Actual Total
Accesses 20000 20000
Instructions 15205 15205
Data readings 1405 1405
Data writings 3390 3390
Simulation steps: 20000

h). Wave


Events visor
Node inspector
Hits-Misses
Bus
States
Configuration
Hits-Misses
Global readings
Hits Misses
Number 2588 375
Rate 87.344% 12.656%
Local readings
Hits Misses
Number 2588 375
Rate 87.344% 12.656%
Global writings
Hits Misses
Number 240 224
Rate 51.724% 48.276%
Local writings
Hits Misses
Number 240 224
Rate 51.724% 48.276%
Global hits and misses
Hits Misses
Number 2828 599
Rate 82.521% 17.479%
Local hits and misses
Hits Misses
Number 2828 599
Rate 82.521% 17.479%
Total replacements: 471
Cache level 1
Actual Total
Accesses 3427 3427
Instructions 2709 2709
Data readings 254 254
Data writings 464 464
Simulation steps: 3427

i). Uncomp



3. Analysis

From the experiment statistics, we can answer the project question as below:

a). We can see that the locality performance various with different benchmarks. The best one is benchmark Cexp, while the worst one is benchmark Swm.

b). Optimization based on certain kinds of program will not improve the overall performance of this system. Because the behavior patterns are various for different programs, and the runtime cache hit/miss rate could also be various with the runtime environment(such as memory allocation and fragmentation situation) of the machine.

c). The missing rate decreases as the execution goes forward, because we deploy the LRU(Least Recently Used) strategy for remove least used memory blocks from cache, and most programs/applications' memory accessing pattern follow the simple LRU statistic rules. As the execution goes forward, the regular memory blocks in cache will be the most frequently referenced memory unit. Thus, the missing rate decreases as execution goes on.

4. Conclusion

Through this project, we practice our knowledge of cache system in virtual environment. We can see the cache locality various among different programs, and also see how the LRU cache strategy optimize the locality of a program.