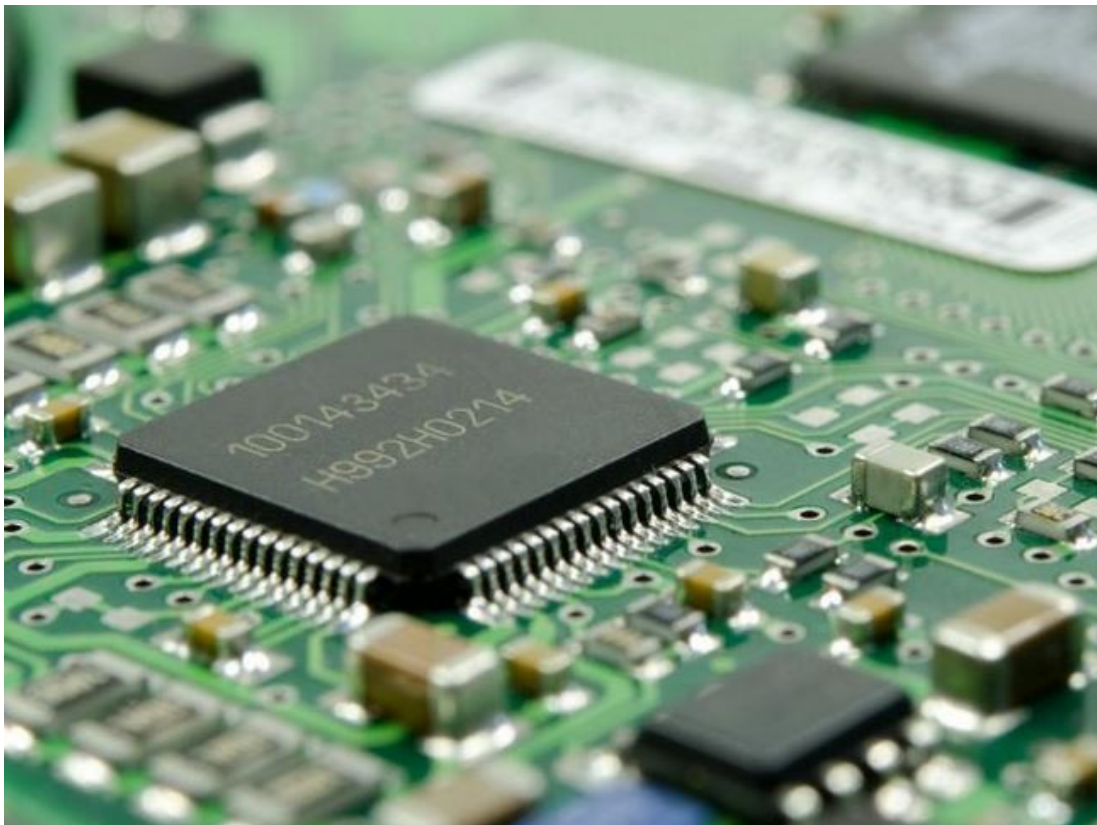


COL-216
Computer Architecture

Assignment 12
Cache Simulation

Submitted by:- Gaurav Chauhan
Entry number:- 2018CS50406



CACHE
SIMULATION

DESIGN OF THE SIMULATION

This cache was implemented by assuming that the input instructions are **block addressable**. I implemented this cache by making a vector of vector of **cell** which is a struct defined as below:-

```
typedef struct cell{
    bool valid;
    int tag;
    int data;
    int priority;
    int last_call;
    int dirty_bit;
} cell;
```

this cell maintains every vital information in an element of the set.

The set is defined as the vector of cells, this helps in maintaining the HIGH and LOW priorities dynamically.

This can be viewed as:-

Visual representation of cache with 2 sets and associativity 4			
Structure data stored Stores complete info. about data			

The main_memory was implemented as a vector<int> with values equal to the value of the index of it's position in the vector.

LRU AND REPLACEMENT POLICY

The LRU policy was maintained for the HIGH and LOW priorities individually.

In each instruction execution, first, the program updates the variable **last_call** for all elements of the cache. This helps in maintaining the LRU Individually for both and also helps in dynamically changing the number of elements in each group.

I maintain the number of elements dynamically in all the sets this is a better choice as the number of recently used in high priority would not be thrown if the associativity is almost equal to the value of the high priority group.

DIFFERENT CASES

1) READ HIT =>

This is done simply by first updating the cache by updating priorities of every set. Then the hit cell is taken to the first place and its priority is maintained as one or high.

2) WRITE HIT =>

This is similar to that in Read hit i.e. first update and then moving the desired cell to the upper place.

3) READ MISS =>

This has two cases i.e. if all the elements in the set are of HIGH priority and the second is in which some are of **HIGH** and some are of **LOW**. **We fetch the data from the main_memory which is used to update the cache and in this case the dirty status is 0.** This is done by handling the cases separately, in the first case it is done by updating and moving the element to the desired place. In second we shift the array and then place the data at the proper place. In both cases the entered element is in LOW priority.

4) WRITE MISS=>

This is similar to read miss and is only different in terms that we do not need main memory directly in it and we update the memory only when the data leaves the cache.

Whenever the data leaves the cache then the main memory is updated each time.

TEST CASE 1

```
input.txt
~/Documents/sem_4_final...
Open Save
8
2
2
2
7, W, 7
8, W, 8
1, W, 1
2, W, 2
2, W, 4
2, R
5, W, 5
6, W, 6
6, R
7, R
8, R
```

```
-----CACHE-----
#Data, Tag, Valid-status(valid=1), dirty-status(dirty=1)
Set No. 1 ->
8 | 4 | 1 | 0
6 | 3 | 1 | 1
Set No. 2 ->
7 | 3 | 1 | 0
5 | 2 | 1 | 1

Cache statistics:
Number of Accesses = 11
Number of Reads = 4
Number of Read Hits = 2
Number of Read Misses = 2
Number of Writes = 7
Number of Write Hits = 1
Number of Write Misses = 6
```

TEST CASE 2

```
Open  input.txt  Save  ~~/Documents/sem_4_final...  
16  
2  
2  
4  
1, W, 1  
5, W, 21  
1, W, 10  
9, W, 9  
1, R  
5, R  
5, R  
9, R  
6, W, 6  
0, W, 0  
0, R  
7, W, 7  
3, W, 3  
7, R  
5, W, 50  
3, R
```

```
The value in the requested position is 3  
-----CACHE-----  
#Data, Tag, Valid-status(valid=1), dirty-status(dirty=1)  
Set No. 1 ->  
0 | 0 | 1 | 1  
0 | 0 | 0 | 0  
Set No. 2 ->  
50 | 1 | 1 | 1  
9 | 2 | 1 | 0  
Set No. 3 ->  
6 | 1 | 1 | 1  
0 | 0 | 0 | 0  
Set No. 4 ->  
3 | 0 | 1 | 1  
7 | 1 | 1 | 1  
  
Cache statistics:  
Number of Accesses = 16  
Number of Reads = 7  
Number of Read Hits = 5  
Number of Read Misses = 2  
Number of Writes = 9  
Number of Write Hits = 2  
Number of Write Misses = 7  
Hit Ratio = 0.4375
```

TEST CASE 3

```
Open  input.txt  Save  ~/Documents/sem_4_final...
32
2
2
4
1, W, 1
9, W, 9
1, W, 2
9, W, 10
17, W, 17
4, W, 4
5, W, 5
25, W, 25
5, W, 50
9, R
25, R
41, W, 3
0, W, 0
```

```
-----CACHE-----
#Data, Tag, Valid-status(valid=1), dirty-status(dirty=1)
Set No. 1 ->
0 | 0 | 1 | 1
0 | 0 | 0 | 0
Set No. 2 ->
25 | 3 | 1 | 1
3 | 5 | 1 | 1
Set No. 3 ->
0 | 0 | 0 | 0
0 | 0 | 0 | 0
Set No. 4 ->
0 | 0 | 0 | 0
0 | 0 | 0 | 0
Set No. 5 ->
4 | 0 | 1 | 1
0 | 0 | 0 | 0
Set No. 6 ->
50 | 0 | 1 | 1
0 | 0 | 0 | 0
Set No. 7 ->
0 | 0 | 0 | 0
0 | 0 | 0 | 0
Set No. 8 ->
0 | 0 | 0 | 0
0 | 0 | 0 | 0

Cache statistics:
Number of Accesses = 13
Number of Reads = 2
Number of Read Hits = 2
Number of Read Misses = 0
Number of Writes = 11
Number of Write Hits = 3
Number of Write Misses = 8
Hit Ratio = 0.384615
```

TEST CASE 4

```
Open  inpu...  Save  -  x
~/Doc...

32
2
4
16
7, W, 7
9, R
2, W, 2
8, W, 8
21, W, 21
11, W, 11
15, W, 15
15, R
24, W, 24
18, W, 18
23, W, 23
25, R
23, W, 23
7, R
3, W, 3
8, W, 8
24, W, 24
21, R
12, W, 12
14, W, 14
3, R
32, W, 32
31, R
32, W, 32
5, W, 5
31, R
1, W, 1
27, R
27, W, 27

Plain Text  Tab Width: 8
```

```
gaurav@gaurav-Inspiron-5570: ~/Documents/sem_4_final/col21...

-----CACHE-----
#Data, Tag, Valid-status(valid=1), dirty-status(dirty=1)
Set No. 1 ->
32 | 8 | 1 | 1 | 1
24 | 6 | 1 | 1 | 1
8 | 2 | 1 | 1 | 1
12 | 3 | 1 | 1 | 0
Set No. 2 ->
21 | 5 | 1 | 1 | 1
1 | 0 | 1 | 1 | 0
5 | 1 | 1 | 1 | 0
25 | 6 | 1 | 0 | 0
Set No. 3 ->
14 | 3 | 1 | 1 | 0
18 | 4 | 1 | 1 | 0
2 | 0 | 1 | 1 | 0
0 | 0 | 0 | 0 | 0
Set No. 4 ->
27 | 6 | 1 | 1 | 1
31 | 7 | 1 | 0 | 1
3 | 0 | 1 | 1 | 1
7 | 1 | 1 | 1 | 1

Cache statistics:
Number of Accesses = 29
Number of Reads = 9
Number of Read Hits = 5
Number of Read Misses = 4
Number of Writes = 20
Number of Write Hits = 5
Number of Write Misses = 15
Hit Ratio = 0.344828
```

TEST CASE 5

gaurav@gaurav-Inspiron-5570: ~/Documents/s

-----CACHE-----

#Data, Tag, Valid-status(valid=1), dirty-s

Set No. 1 ->

32 | 8 | 1 | 1 | 1

24 | 6 | 1 | 1 | 0

8 | 2 | 1 | 1 | 0

20 | 5 | 1 | 1 | 0

Set No. 2 ->

41 | 10 | 1 | 0 | 0

21 | 5 | 1 | 1 | 0

1 | 0 | 1 | 1 | 0

5 | 1 | 1 | 1 | 0

Set No. 3 ->

18 | 4 | 1 | 1 | 1

14 | 3 | 1 | 1 | 0

2 | 0 | 1 | 1 | 0

0 | 0 | 0 | 0 | 0

Set No. 4 ->

7 | 1 | 1 | 1 | 1

27 | 6 | 1 | 1 | 1

31 | 7 | 1 | 0 | 1

23 | 5 | 1 | 1 | 0

Cache statistics:

Number of Accesses = 35

Number of Reads = 10

Number of Read Hits = 5

Number of Read Misses = 5

Number of Writes = 25

Number of Write Hits = 8

Number of Write Misses = 17

Hit Ratio = 0.371429

gaurav@gaurav-Inspiron-5570:~/Documents/sem_4_final/col216/assignment_12_2018CS5

0406\$

Open



inpu...
~/Doc...

Save



32
2
4
16
7, W, 7
9, R
2, W, 2
8, W, 8
21, W, 21
11, W, 11
15, W, 15
15, R
24, W, 24
18, W, 18
23, W, 23
25, R
23, W, 23
7, R
3, W, 3
8, W, 8
24, W, 24
21, R
12, W, 12
14, W, 14
3, R
32, W, 32
31, R
32, W, 32
5, W, 5
31, R
1, W, 1
27, R
27, W, 27
7, W, 7
20, W, 20
23, W, 23
32, W, 32
18, W, 18
41, R

Plain Text

Tab Width: 8

Ln 32, Col 6

INS

TEST CASE 6

```
Open ▾  inpu... Save  -  □  ×
~/Doc...

32
2
1
16
7, W, 7
9, R
2, W, 2
8, W, 8
21, W, 21
11, W, 11
15, W, 15
15, R
24, W, 24
18, W, 18
23, W, 23
25, R
23, W, 23
7, R

Plain Text ▾  Tab
```

```
gaurav@gaurav-Inspiron-5570: ~/Documents/sem_...  🔍  ☰  -  □  ×

Process for Instruction count 14
Value removed from 23
-----CACHE-----
#Data, Tag, Valid-status(valid=1), dirty-status(dirty=1)
Set No. 1 ->
0 | 0 | 0 | 0 | 0
Set No. 2 ->
0 | 0 | 0 | 0 | 0
Set No. 3 ->
18 | 1 | 1 | 1 | 0
Set No. 4 ->
0 | 0 | 0 | 0 | 0
Set No. 5 ->
0 | 0 | 0 | 0 | 0
Set No. 6 ->
21 | 1 | 1 | 1 | 0
Set No. 7 ->
0 | 0 | 0 | 0 | 0
Set No. 8 ->
7 | 0 | 1 | 0 | 0
Set No. 9 ->
24 | 1 | 1 | 1 | 0
Set No. 10 ->
25 | 1 | 1 | 0 | 0
Set No. 11 ->
0 | 0 | 0 | 0 | 0
Set No. 12 ->
11 | 0 | 1 | 1 | 0
Set No. 13 ->
0 | 0 | 0 | 0 | 0
Set No. 14 ->
0 | 0 | 0 | 0 | 0
Set No. 15 ->
0 | 0 | 0 | 0 | 0
Set No. 16 ->
15 | 0 | 1 | 1 | 1

Cache statistics:
Number of Accesses = 14
Number of Reads = 4
Number of Read Hits = 1
Number of Read Misses = 3
Number of Writes = 10
Number of Write Hits = 1
Number of Write Misses = 9
Hit Ratio = 0.142857
```

TEST CASE 7

```
Open ▾ inpu... ~/Doc... Save ≡ _ □ ×
```

```
32
2
4
16
7, W, 7
9, R
1, W, 2
6, W, 8
1, W, 21
11, W, 11
5, W, 15
15, R
4, W, 24
18, W, 18
3, W, 23
25, W, 8
2, W, 23
7, R
13, W, 8
8, R
```

```
Plain Text ▾ Tab Width: 8 ▾ Ln 10, Col 10 ▾ INS
```

```
4-----CACHE-----
#Data, Tag, Valid-status(valid=1), dirty-status(dirty=1)
Set No. 1 ->
8 | 2 | 1 | 0 | 0
24 | 1 | 1 | 1 | 0
0 | 0 | 0 | 0 | 0
0 | 0 | 0 | 0 | 0
Set No. 2 ->
21 | 0 | 1 | 1 | 1
8 | 3 | 1 | 1 | 0
8 | 6 | 1 | 1 | 0
15 | 1 | 1 | 1 | 0
Set No. 3 ->
23 | 0 | 1 | 1 | 0
18 | 4 | 1 | 1 | 0
8 | 1 | 1 | 1 | 0
0 | 0 | 0 | 0 | 0
Set No. 4 ->
7 | 1 | 1 | 1 | 1
23 | 0 | 1 | 1 | 0
15 | 3 | 1 | 0 | 0
11 | 2 | 1 | 1 | 0

Cache statistics:
Number of Accesses = 16
Number of Reads = 4
Number of Read Hits = 1
Number of Read Misses = 3
Number of Writes = 12
Number of Write Hits = 1
Number of Write Misses = 11
Hit Ratio = 0.125
```

TEST CASE 8

```
Open  inpu...  Save  ~/Doc...
32
2
4
16
7, W, 7
9, R
2, W, 2
8, W, 8
21, W, 21
1, W, 11
15, W, 5
5, R
24, W, 24
18, W, 8
23, W, 3
5, W, 6
3, W, 2
17, R
9, W, 3
18, R
24, W, 24
21, R
12, W, 12
14, W, 14
Plain Text  Tab
```

```
-----CACHE-----
#Data, Tag, Valid-status(valid=1), dirty-status(dirty=1)
Set No. 1 ->
24 | 6 | 1 | 1 | 1
12 | 3 | 1 | 1 | 0
8 | 2 | 1 | 1 | 0
0 | 0 | 0 | 0 | 0
Set No. 2 ->
6 | 1 | 1 | 1 | 1
21 | 5 | 1 | 0 | 0
3 | 2 | 1 | 1 | 0
17 | 4 | 1 | 0 | 0
Set No. 3 ->
8 | 4 | 1 | 1 | 1
14 | 3 | 1 | 1 | 0
2 | 0 | 1 | 1 | 0
0 | 0 | 0 | 0 | 0
Set No. 4 ->
2 | 0 | 1 | 1 | 0
3 | 5 | 1 | 1 | 0
5 | 3 | 1 | 1 | 0
7 | 1 | 1 | 1 | 0

Cache statistics:
Number of Accesses = 20
Number of Reads = 5
Number of Read Hits = 1
Number of Read Misses = 4
Number of Writes = 15
Number of Write Hits = 2
Number of Write Misses = 13
Hit Ratio = 0.15
```

TEST CASE 9

```
Open ▾ input... Save ≡ - □ ×
~/Doc...

32
2
4
16
7, W, 7
9, R
10, R
20, R
1, R
2, W, 13
5, R
2, W, 2
8, W, 8
21, W, 21
1, W, 11
18, R
24, W, 24
21, R
12, W, 12
14, W, 14
```

Plain Text ▾

```
-----CACHE-----
#Data, Tag, Valid-status(valid=1), dirty-status(dirty=1)
Set No. 1 ->
12 | 3 | 1 | 1 | 0
24 | 6 | 1 | 1 | 0
8 | 2 | 1 | 1 | 0
20 | 5 | 1 | 0 | 0
Set No. 2 ->
21 | 5 | 1 | 1 | 1
11 | 0 | 1 | 1 | 1
5 | 1 | 1 | 0 | 0
9 | 2 | 1 | 0 | 0
Set No. 3 ->
2 | 0 | 1 | 1 | 1
14 | 3 | 1 | 1 | 0
18 | 4 | 1 | 0 | 0
10 | 2 | 1 | 0 | 0
Set No. 4 ->
7 | 1 | 1 | 1 | 0
0 | 0 | 0 | 0 | 0
0 | 0 | 0 | 0 | 0
0 | 0 | 0 | 0 | 0

Cache statistics:
Number of Accesses = 16
Number of Reads = 7
Number of Read Hits = 1
Number of Read Misses = 6
Number of Writes = 9
Number of Write Hits = 2
Number of Write Misses = 7
Hit Ratio = 0.1875
```

TEST CASE 10

```
Open  inpu...  Save  ~/Doc...
32
4
4
16
7, W, 7
21, W, 21
1, W, 11
18, R
24, W, 24
21, R
12, W, 12
14, W, 14
9, R
10, R
20, R
1, R
2, W, 13
5, R
2, W, 2
8, W, 8

Plain Text  Tab Width: 8  Ln 20, Col 8  INS
```

```
-----CACHE-----
#Data, Tag, Valid-status(valid=1), dirty-status(dirty=1)
Set No. 1 ->
2 | 1 | 1 | 1 | 1
8 | 4 | 1 | 1 | 0
20 | 10 | 1 | 0 | 0
10 | 5 | 1 | 0 | 0
Set No. 2 ->
11 | 0 | 1 | 1 | 1
21 | 10 | 1 | 1 | 1
5 | 2 | 1 | 0 | 0
9 | 4 | 1 | 0 | 0

Cache statistics:
Number of Accesses = 16
Number of Reads = 7
Number of Read Hits = 2
Number of Read Misses = 5
Number of Writes = 9
Number of Write Hits = 1
Number of Write Misses = 8
Hit Ratio = 0.1875
```