CS2323 Computer Architecture

V Sri Charan Reddy
CS16BTECH11044

19 September 2016

This document is generated by LaTeX

1

Given address = 0x CDDBAA
Its representation in binary form = 0b 1100 1101 1101 1011 1010 1010
Cache size = No.of.blocks * Block size
= No of sets* No of ways * blocksize
Given, no of sets = 4096
no of ways =16
block size = 64Bytes
Cache size = 4MB
block offset = $log_2 64$ = 6 bits
set index= $log_2 4096$ = 12
The LSB starts with the blockoffset then the set index

Tag	Set Index	Offset
1100 11	01 1101 1011 10	10 1010

Its set index is 01 1101 1011 10 in binary
In base 10, set index = 1902
It will be mapped to any of the 16 ways depending if the place is occupied or not. The default assigning is to the first way that is available, if occupied the replacement tehniques are used and the used blocks are replaced.

2

Given ,
No of accesses per second = 50000000
leakage power = 0.08W
dynamic energy = 0.9nJ/access
In one second, leakage energy= leakage power*time
leakage energy = 0.08*1 =0.08J
dynamic energy = $.9 * 10^{-9} * 50000000$ =0.045J
leakage fraction = $\frac{leakageenergy}{leakageenergy+dynamicenergy}$
leakage fraction = $\frac{0.08}{0.08+0.045} = 0.64$
leakage percentage = leakage * 100
leakage = 64 percent

3

Given block size =4
offset = $log_2 4 = 2$ bits
8 sets means set index is $log_2 8 = 3$ bits

	Tag	Set Index	Offset
0	000	000	00
63	001	111	11
1	000	000	01
62	001	111	10
2	000	000	10
61	001	111	01
3	000	000	11
60	001	111	00
4	000	001	00
59	001	110	11
5	000	001	01
58	001	110	10
6	000	001	10
57	001	110	01
7	000	001	11
56	001	110	00
8	000	010	00
55	001	101	11
9	000	010	01
54	001	101	10
10	000	010	10
53	001	101	01
11	000	010	11
52	001	101	00

cache-1
Tag is the first 3 bits and set index is the next 3 bits and the offset is in next 2 bits.

3bits	3 bits	2 bits
Tag	Setindex	offset

cache-2
Tag is the last 3 bits, and set index is the before 3 bits and the rest is offset.

2 bits	3 bits	3 bits
Offset	Setindex	Tag

a)For accessing the sequence by cache 1 After comparing set index and tag the hit miss sequence of these address access is:

0	63	1	62	2	61	3	60	4	59	5	58	6	57	7	56	8	55	9
M	M	H	H	H	H	H	H	M	M	H	H	H	H	H	H	M	M	H

54	10	53	11	52
H	H	H	H	H

ratio $= \frac{hits}{access} = \frac{18}{24} = \frac{3}{4}$

For accessing the sequence by cache 2 After comparing set index and tag the hit miss sequence of these address access is:

0	63	1	62	2	61	3	60	4	59	5	58	6	57	7	56	8	55	9
M	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M

54	10	53	11	52
M	M	M	M	M

ratio $= \frac{hits}{access} = \frac{0}{24} = 0$

b)

	Tag	Setindex	offset
0	000	000	00
64	010	000	00
128	100	000	00
192	110	000	00
1	000	000	01
65	010	000	01
129	100	000	01
193	110	000	01
11	000	010	11
75	010	010	11
139	100	010	11
203	110	010	11
9	000	010	01
137	100	010	01
201	110	010	01
73	010	010	01

For accessing the sequence by cache 1 After comparing set index and tag the hit miss sequence of these address access is:

0	64	128	192	1	65	129	193	11	75	139	203	9	137	201	73
M	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M

ratio $= \frac{hits}{access} = \frac{0}{16} = 0$

For accessing the sequence by cache 2 After comparing set index and tag the hit miss sequence of these address access is:

0	64	128	192	1	65	129	193	11	75	139	203	9	137	201	73
M	H	H	H	M	H	H	H	M	H	H	H	M	H	H	H

ratio $= \frac{hits}{access} = \frac{12}{16} = \frac{3}{4}$

c)

In part a cache-1 is better than cache-2 as the ratio of hits:access is higher for cache-1

In part b cache-2 is better than cache-1 as the ratio of hits:access is higher for cache-2
As the hits:access changes , a particular kind of cache is not always better

4

Frequency of P1 = 2 GHz
Frequency of P2 = 1.5 GHz
No.of instruction in program = 10^6

No of instructions in A $=\frac{20}{100} * 10^6 = 2 * 10^5$

No of instructions in B $=\frac{25}{100} * 10^6 = 25 * 10^4$

No of instructions in C $=\frac{40}{100} * 10^6 = 4 * 10^5$

No of instructions in D $=\frac{15}{100} * 10^6 = 15 * 10^4$

CPI of P1 A=1,B=2,C=3,D=4
Total cycles for P1 to run a program = $\sum_{i=1}^{i=4} A_i$ * instructions in A_i
= 1*2 * 10^5 + 2 *25 * 10^4 + 3 *4 * 10^5 + 4 *15 * 10^4
=$25 * 10^5$
Time taken for P1 to run a program = $\frac{no.of.cycles}{frequency}$
$= \frac{25*10^5}{2GHz}$
CPI of P2 A=2,B=2,C=2,D=2
Total cycles for P1 to run a program = 2*2 * 10^5 + 2 *25 * 10^4 + 2 *4 * 10^5 + 2 *15 * 10^4
= 2 * 10^6
Time taken for P2 to run a program = $\frac{no.of.cycles}{frequency}$
$= \frac{2*10^6}{1.5GHz}$
As time taken for P1 is less than P2
So P1 is faster for this program

5

After every write miss P+1 bit will be 1 and after every read miss P+1 bit will be 0

P0	P1	P2	P3	P+1
1	0	0	0	0

After P0 has read miss

P0	P1	P2	P3	P+1
0	1	0	0	1

After P1 has write miss

P0	P1	P2	P3	P+1
0	0	1	0	1

After P2 has write miss

P0	P1	P2	P3	P+1
0	0	1	1	0

After P3 has read miss

P0	P1	P2	P3	P+1
0	0	0	1	1

After P3 has write miss

P0	P1	P2	P3	P+1
0	1	0	1	0

After P1 has read miss

6

Two applictions are running on L2 cache Given that number of misses are scaled linearly
So difference of misses between successive ways for application 1
$=\frac{1000-500}{6-2} = 125$
So difference of misses between successive ways for application 2
$=\frac{2000-1800}{6-2} = 50$

way	2	3	4	5	6
application1	1000	875	750	625	500
application2	2000	1950	1900	1850	1800

As the cache is given as 8 way,
To improve the performance we must have less misses
misses = ways in application1 * misses in application1 +ways in application2 *misses in application2

Application1(ways)	Application2(ways)	misses
2	6	2*1000+6*1800 =12800
3	5	3* 875+5*1850 =11875
4	4	4*750+4*1900 =10600
5	3	5*625+3*1950 =8975
6	2	6*500+2*2000 =7000

So misses are minimum if 6 ways for application1 and 2 ways for application2