Investigating Cache Technology

Objectives

At the end of this lab you should be able to:

- Investigate Directly Mapped cache organization
- Explain what cache hit and miss rates are
- Understand one drawback of Directly Mapped cache type
- Investigate 2-way Set-Associative cache mapping
- Investigate 4-way Set-Associative cache mapping
- Explain the effect of cache size and mapping scheme on cache performance

Lab Exercises - Investigate and Explore

The following exercises require the use of the **Data Cache** simulator which is part of the CPU-OS Simulator software. You can access the **Data Cache** simulator in CPU simulator window by selecting the **Cache-Pipeline** tab and then selecting the cache type Data and clicking on the **SHOW CACHE…** button.

Exercise 1 – Investigating Directly Mapped cache organization

Create a new program, call it **CacheTest1** and enter the following code:

```
MOV #0, R01
STB R01, @R01
CMP #63, R01
JEQ 31
ADD #1, R01
JMP 6
HLT
```

The above code writes numbers 0 to 63 in memory locations 0 to 63. Run it and observe the contents of the data in memory. To see the memory click on the **SHOW PROGRAM DATA MEMORY…** button.

Click on the **SHOW CACHE…** button to display the data cache window. Make sure the **Stay on top** check box is checked. Now, flush the cache by clicking on the **FLUSH** button and configure the cache with the following settings:

Block Size = 4
Cache Type = Direct Mapped
Cache Size = 16
Write Policy = Write-Back

Now insert the following code below the instruction **JMP 6** in the above code:

```
LDB 0, R00
LDB 1, R00
LDB 2, R00
LDB 3, R00
```

To execute the above LDB instructions, double-click on each of the above LDB instructions. Write down what you observe in the table below:

Addr	Data	Hits	Block
0000	00		0
0001	01	1	0
0002	02	1	0
0003	03	1	0

Also make a note of the following data displayed in the **Cache Stats** frame:

Hits	3	**%Hits**	75%
Misses	1	**%Misses**	25%

Note: %Hits = 100 - %Misses

Insert the following LDB instruction after the last LDB instruction above and execute it by double-clicking on it:

```
LDB 4, R00
```

Write down the additional contents (i.e. in addition to the above data) of the cache below:

Addr	Data	Block
0004	04	1
0005	05	1
0006	06	1
0007	07	1

Briefly explain your observations below:

A new block of 4 bytes is loaded into the cache.

Exercise 2 – Investigating a disadvantage of Directly Mapped cache

First flush the contents of the cache by clicking on the FLUSH button. Then enter the following instructions after the last LDB instruction in the above program:

LDB 16, R00
LDB 32, R00

Next execute only the following three instructions in the above program:

LDB 0, R00
LDB 16, R00
LDB 32, R00

Repeat the above two times and make note of what you observe below:

Addr	Data	Hits	Block
0032	20		0
0033	21		0
0034	22		0
0035	23		0

.

Hits	0	%Hits	0%
Misses	6	%Misses	100%

Briefly explain your findings below:

Every time the block is replaced by another block.

Exercise 3 - Investigating Set-Associatively Mapped cache organization

Now configure the cache with the following settings:

Block Size = 4
Cache Type = Set Associative
Cache Size = 16
Set Blocks = 2-way
Write Policy = Write-Back

Insert the following new LDB instructions after the **LDB 4, R00** instruction:

LDB 8, R00
LDB 12, R00

Execute the following set of LDB instructions one after the other in the order listed below:

LDB 0, R00
LDB 4, R00
LDB 8, R00
LDB 12, R00

Write down your observations below (the **Addr** field is filled in for you):

Addr	Set	Block
0000	0	0
0004	1	0
0008	0	1
0012	1	1

Next re-configure the cache so that the **Set Blocks** is set to **4-way**.

Execute the following set of LDB instructions one after the other in the order listed below:

LDB 0, R00
LDB 4, R00
LDB 8, R00
LDB 12, R00

Write down your observations below (the **Addr** field is filled in for you):

Addr	Set	Block
0000	0	0
0004	0	1
0008	0	2
0012	0	3

Clear the cache by clicking on the **FLUSH** button. Next execute only the following two LDB instructions in the above program:

LDB 0, R00
LDB 16, R00

Repeat the above three times and make note of what you observe below:

Addr	Data	Hits
0000	00	2
0016	10	2

.

Hits	4	**%Hits**	66.6%
Misses	2	**%Misses**	33.3%

Briefly explain your findings below:

Two blocks of the same set are loaded into the cache. The first two loads are misses, the rest is hits.

Exercise 4 - Investigating cache size and type on cache performance

Enter the following program, compile it and load it in CPU memory:

```
program CacheTest
    var a array(50) byte

    i = 49
    for n = 0 to 49
        p = a(n) + a(i)
        i = i - 1
    next
end
```

Now, make sure that the charts are enabled. Select the correct cache parameters and fill in the tables below against each of the cache sizes shown in the tables. You need to run the above program to completion for each of the cases below (this may take up to a minute). Make sure you click on the **RESET PROGRAM** button and slide the speed selector up to the fastest position prior to running the program in each case. Also make sure the **Write Policy** is set to **Write-Back**.

Direct mapping (for cache sizes 16 to 256)

16	32	64	128	256
43	24	06	06	06

2-way set-associative mapping (for cache sizes 16 to 256)

16	32	64	128	256
38	15	06	06	06

4-way set-associative mapping (for cache sizes 16 to 256)

16	32	64	128	256
16	10	06	06	06

8-way set-associative mapping (for cache sizes 32 to 128)

32	64	128
10	06	06

Briefly comment on the results (**Tip**: Switch from bar chart to line graph in the displayed charts and compare the steepness of the slopes of the graphs or compare the individual values of the bars in bar charts):

Increasing the associativity of the cache decreases the miss rate. Every increase the miss rate decreases less. From 4 to 8 set associative the effect is not noticable.