# **Lab 5: Cache Design**

In this lab, we would like to explore the cache performance as we change the cache design parameters. We will do so while executing the provided matrix multiplication code *matrix-vector-multiplication.asm.*

Fill in the table with the achieved ***hit ratio*** using the provided information from the Data Cache Simulator tool (Tools → Data Cache Simulator)

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Scenario 1 | Scenario 2 | Scenario 3 |
| Cache block size (words) | 2,4,8,16,32,64,128 | 2,4,8,16,32,64,128 | 2,4,8,16,32,64 |
| Cache size (bytes) | 256, 512, 1024 | 256, 512, 1024 | 512 |
| Placement policy | Direct Mapped | Full Associative | n-Way associative |
| Set size (blocks) | - | - | 2,4,8 |
| Block replacement policy | - | LRU, Random | LRU |

**Note that as you change the cache block size, you would need to adjust the number of blocks to maintain a fixed cache size.**

## Scenario 1: Direct mapped cache

We only test the impact of cache size and block size. Note that in direct-mapped cache you have a rigid placement and replacement strategy (revise lecture slides if not sure)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Block size (words) Cache Size (bytes) | **2** | **4** | **8** | **16** | **32** | **64** | **128** |
| 256 | 50% | 60% | 65% | 68% | 48% | 5% | x |
| 512 | 66% | 73% | 80% | 81% | 71% | 49% | 5% |
| 1024 | 71% | 82% | 88% | 91% | 86% | 75% | 52% |

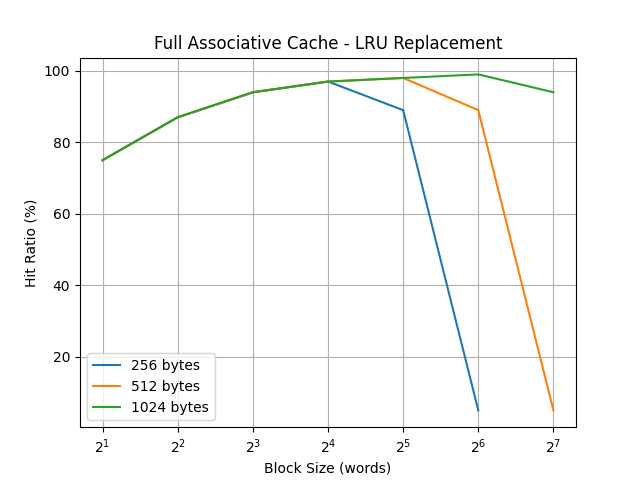
**Table 1.1:** Direct Mapped Cache

## Scenario 2: Full Associative Cache

### LRU replacement

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Block size (words) Cache Size (bytes) | **2** | **4** | **8** | **16** | **32** | **64** | **128** |
| 256 | 75% | 87% | 94% | 97% | 89% | 5% | x |
| 512 | 75% | 87% | 94% | 97% | 98% | 89% | 5% |
| 1024 | 75% | 87% | 94% | 97% | 98% | 99% | 94% |

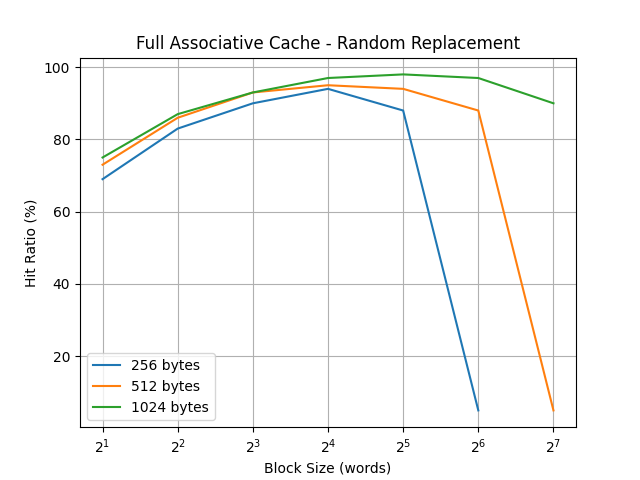
**Table 2.1:** Full Associative Cache with LRU replacement



### Random replacement

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Block size (words) Cache Size (bytes) | **2** | **4** | **8** | **16** | **32** | **64** | **128** |
| 256 | 69% | 83% | 90% | 94% | 88% | 5% | x |
| 512 | 73% | 86% | 93% | 95% | 94% | 88% | 5% |
| 1024 | 75% | 87% | 93% | 97% | 98% | 97% | 90% |

Table 1.2: Full Associative Cache with random replacement

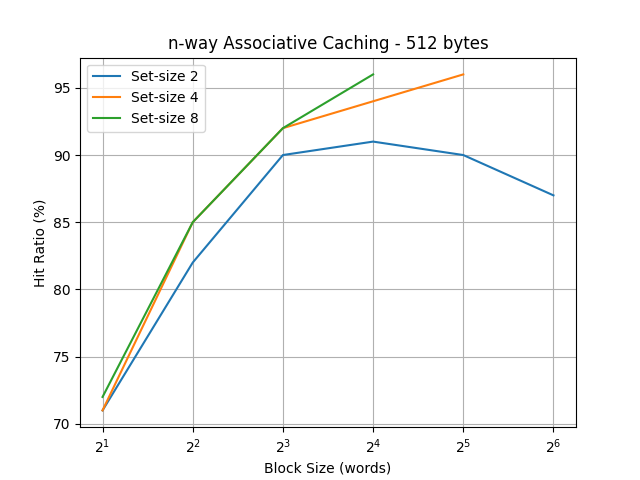


## Scenario 3: n-way Associative Caching

### Cache Size 512 Bytes

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Block size (words) set-size | **2** | **4** | **8** | **16** | **32** | **64** | **128** |
| 2 | 71% | 82% | 90% | 91% | 90% | 87% | x |
| 4 | 71% | 85% | 92% | 94% | 96% | x | x |
| 8 | 72% | 85% | 92% | 96% | x | x | x |

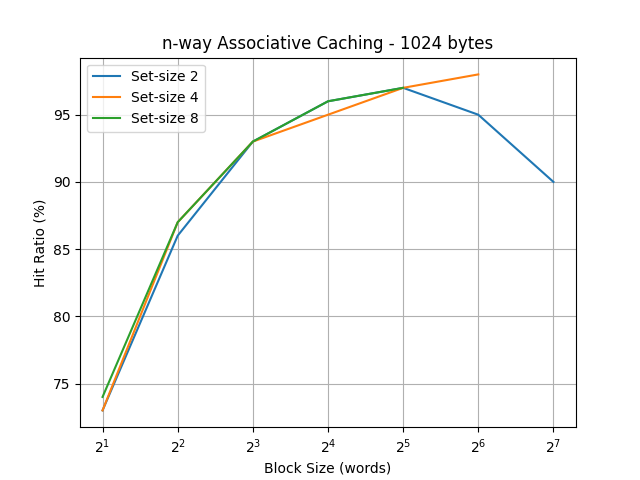
Table 3.1: n-Way associative (512 Kbyte)



### Cache Size 1024 Bytes

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Block size (words) set-size | **2** | **4** | **8** | **16** | **32** | **64** | **128** |
| 2 | 73% | 86% | 93% | 96% | 97% | 95% | 90% |
| 4 | 73% | 87% | 93% | 95% | 97% | 98% | x |
| 8 | 74% | 87% | 93% | 96% | 97% | x | x |

Table 3.1: n-Way associative (512 Kbyte)



# Discussion

1. What is the impact of cache size on the cache performance? What is the downside of using larger cache?

Having a larger cache size increases the cache hit ratio to an extent but it is also important to divide the cache into an appropriate amount of blocks for it to be effective.

The downside of using large amounts of cache is that it is expensive and so adding more and more cache leads to diminishing returns.

1. Discuss the impact of block size of cache performance illustrating
   1. Why we have a small hit ratio for small block sizes?
   2. Why we have a small hit ratio for large block sizes?
   3. What is the optimal block size for various cache configurations?

a)

1. How Associative caching perform in comparison to direct mapped caching? Why? What is the downside of using Associative cache?
2. Discuss the performance of LRU and Random replacement policies for various cache configurations.
3. Discuss the impact of set size in n-way associative cache. (Hint: consider a specific cache size)
4. Do you believe these conclusions would change if the array size if different? Why? (You may conduct more simulations)