

Assume that each cache block and memory block have the same size, please organize the cache as a Direct Mapped, 2-way set associative, and fully associative.

**Direct Mapped**

|  |  |
| --- | --- |
| Set # | Memory Blocks # |
| 00 | 00,04 |
| 01 | 01,05 |
| 02 | 02,06 |
| 03 | 03,07 |

There is just one place in the cache where a block can be placed. Although it makes it very easy to find a cache block, the blocks cannot be placed anywhere.

**2-way set associative**

|  |  |
| --- | --- |
| Set # | Memory Blocks # |
| 00 | 00,02,04,06 |
| 00 | 00,02,04,06 |
| 01 | 01,03,05,07 |
| 01 | 01,03,05,07 |

This cache consists of 2-block sets. As a result, the index is presently used to locate the set, while the tag is pre-owned to locate the block within it.

**Fully Associative**

|  |  |
| --- | --- |
| Set # | Memory Blocks # |
| 00 | 00,01,02,03,04,05,06,07 |
| 01 | 00,01,02,03,04,05,06,07 |
| 02 | 00,01,02,03,04,05,06,07 |
| 03 | 00,01,02,03,04,05,06,07 |

It's not necessary to create an index because a block of data can go where it wants in the cache. If you want to discover a block in the cache, you will need to compare every tag, however block placement is configurable