Direct-Mapped Cache Lookup

Ex: Cache Size = 16 KB
Block Size = 32 bytes
# blocks = 16 KB / 32 = 512

18

tag

9

block offset

5

data
(2-way) Set Associative Cache Lookup

Ex: Cache Size = 16 KB
Block Size = 32 bytes

\# Sets = \frac{16K}{32 \cdot 2} = 256
**Fully Associative Cache Lookup**

Ex: Cache Size = 16 KB
Block size = 32 bytes
# blocks = $\frac{16 \text{ K}}{32} = 5/2$
Victim Cache

Motivation:
Conflict misses usually isolated to a few sets

<table>
<thead>
<tr>
<th>Set</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>4</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

number of blocks that map to each set.

Primary Cache:
- Large (e.g., 16KB)
- Low associativity (e.g., Direct Mapped)

Victim Cache:
- Small (e.g., 16 blocks)
- Fully Associative

- On access, check both caches (in parallel)
- Eviction from primary cache → victim cache
- Victim cache hit → primary cache
- If both caches miss, fetch from main memory → primary cache.
Multi-Level Caches

- Goal: Fast + large cache
- As you go down hierarchy:
  - Larger cache
  - Larger block size
  - Higher associativity
- Inclusion

\[
\text{Avg Mem. Acc. Time} = \text{hit}_{L1} + \text{miss rate}_{L1} \left( \text{hit}_{L2} + \text{miss rate}_{L2} \left( \text{hit}_{L3} + \text{miss rate}_{L3} \cdot \text{miss penalty} \right) \right)
\]