Memory Hierarchy and Cache Design (3)

Main Memory Background

- Conventional DRAM system

- Performance Metrics of Main Memory:
  - Latency: Affects cache miss penalty
  - Access Time: time between the request and when the desired word arrives
  - Cycle Time: minimum time between requests
  - Bandwidth: Affects I/O performance & cache miss penalty (especially when a large block is used in the L2 cache)

- Main Memory uses DRAM (dynamic RAM)
  - Dynamic because of the need to be refreshed periodically (but requires only 1 transistor/bit)
  - Addresses are divided into 2 parts:
    - RAS or Row Access Strobe
    - CAS or Column Access Strobe

- Cache uses SRAM (static RAM)
  - No refresh (but requires 6 transistors/bit)
  - Address not divided for fast access

- Comparison
  - Capacity: DRAM is 4-8 times that of SRAM
  - Cycle time: SRAM is 8-16 times faster than DRAM
  - Cost: SRAM is 8-16 times more expensive than DRAM
Trends in DRAM

<table>
<thead>
<tr>
<th>Year of introduction</th>
<th>Chip-size</th>
<th>Slowest DRAM</th>
<th>Fastest DRAM</th>
<th>Column access delay (CAS)</th>
<th>Cycle time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1980</td>
<td>1 Mbit</td>
<td>180 ns</td>
<td>120 ns</td>
<td>75 ns</td>
<td>150 ns</td>
</tr>
<tr>
<td>1985</td>
<td>2 Mbit</td>
<td>180 ns</td>
<td>120 ns</td>
<td>90 ns</td>
<td>220 ns</td>
</tr>
<tr>
<td>1990</td>
<td>1 Mbit</td>
<td>120 ns</td>
<td>100 ns</td>
<td>75 ns</td>
<td>190 ns</td>
</tr>
<tr>
<td>1995</td>
<td>4 Mbit</td>
<td>100 ns</td>
<td>90 ns</td>
<td>70 ns</td>
<td>185 ns</td>
</tr>
<tr>
<td>1996</td>
<td>10 Mbit</td>
<td>80 ns</td>
<td>60 ns</td>
<td>70 ns</td>
<td>130 ns</td>
</tr>
<tr>
<td>1995</td>
<td>6 Mbit</td>
<td>75 ns</td>
<td>60 ns</td>
<td>73 ns</td>
<td>90 ns</td>
</tr>
</tbody>
</table>

Capacity improves by 60% per year
Row access time improves by 7% per year

Main Memory Organizations

- Basic Memory Organization
  - one-word wide bus
  - 4 clock cycles to send the address
  - 24 clock cycles for the access time per word
  - 4 clock cycles to send a word of data
- Example
  - cache block of 4 words
  - miss penalty = 4 x (6 + 24 + 4) = 128 cycles

Faster Memory System

1. Wider Main Memory
2. Simple Interleaved Memory
3. Independent Memory Banks
4. Avoiding Memory Bank Conflicts
5. DRAM-specific Interleaving

First Technique: Wider Main Memory

- Cache miss penalty
  - two-word wide bus
    - 2 x (6 + 24 + 4) = 64 cycles
  - four-word wide bus
    - 4 x (6 + 24 + 4) = 96 cycles
- Drawbacks
  - Higher bus costs
  - Multiplexer
  - Reduced expandability
  - More frequent "read-modify-write" in memories with error correction
Second Technique: Simple Interleaved Memory

- Memory consists of several DRAM Chips
  - Each chip is capable of autonomous operation
- Organize memory chips in banks and issue memory requests to all banks at the same time
  - Banks are one word wide

Four way interleaving

<table>
<thead>
<tr>
<th>Address</th>
<th>Bank 0</th>
<th>Address</th>
<th>Bank 1</th>
<th>Address</th>
<th>Bank 2</th>
<th>Address</th>
<th>Bank 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Memory Interleaving

- Mapping addresses to banks affects the behavior of the memory system
  - Optimized for sequential access
- May spread consecutive addresses to several banks
  - Interleaving Factor
    - Normally word interleaved
    - Can also be byte interleaved
    - Depends on the organization of the bank
- Goal: deliver information from new bank on each cycle
  - Need more banks than the number of cycles to access a bank
- As memory chip size increases - use fewer chips
  - Constructing multiple banks becomes difficult
- Restricted Expandability
  - Can increase memory only by doubling it
  
- Cache Miss Penalty
  - \(4 \times 24 + 4 \times 4 = 44\) cycles

Third Technique: Independent Memory Banks

- Multiple independent banks
  - Multiple memory controllers
  - Each bank uses separate address and data lines
Fourth Technique: Avoiding Memory Bank Conflicts

- Problem
  ```
  int x[256][512];
  for (j = 0; j < 512; j = j+1)
    for (i = 0; i < 256; i = i+1)
      x[i][j] = 2 * x[i][j];
  ```
  Even with 128 banks there are conflicts, since 512 is a multiple of 128

- Software solutions
  - loop interchange
  - Resizing the array

- Hardware solutions
  - Based on the Chinese Remainder Theorem
  - Prime number of banks
  - bank number = address mod number of banks
  - address within bank = address mod number of words in bank

Avoiding Memory Bank Conflicts

- Example

<table>
<thead>
<tr>
<th>Address within bank</th>
<th>Memory bank</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
</tr>
</tbody>
</table>

Fifth Technique: DRAM-Specific Interleaving

- Multiple column accesses: page mode
  - DRAM buffers a row of bits inside the DRAM for column access (e.g., 16 Kbits row for 64 Mbits DRAM)
  - Allow repeated column access without another row access
  - 64 Mbit DRAM: cycle time = 90 ns, optimized access = 25 ns

- New DRAMs
  - Example: RAMBUS
    - each chip acts like a memory system
    - uses a packet-switched bus
    - is synchronous to the CPU clock
    - returns a variable amount of data
    - even performs its own refresh

Summary of Faster Memory Systems

1. Wider Main Memory
2. Simple Interleaved Memory
3. Independent Memory Banks
4. Avoiding Memory Bank Conflicts
5. DRAM-specific Interleaving