SNOOPING PROTOCOL

- 3 states: Exclusive, Invalid, Shared.

- Main idea: only 1 processor should modify a specific piece of data. What would happen otherwise?

- Concept of "block ownership."

- Modus Operandi:

  1. Processor owns the sole copy of a cache block. If block is to be modified/or owned then:
     - change state from shared to exclusive
     - send invalidation signal to other processors.

  2. If another processor requests this block, state must be shared again.

  3. Snooping also sees misses.
     - knows when exclusive cache block has been requested by another processor and state should be made shared.
* Protocol implemented with a FSM @ each node

- controller responds to requests from:
  - processor
  - bus

- controller changes the state of the selected block
- controller uses bus to access data or invalidate it.

* Some operations involved in the protocol

<table>
<thead>
<tr>
<th>Request</th>
<th>Source</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read hit</td>
<td>processor</td>
<td>Read data in cache</td>
</tr>
<tr>
<td>Write hit</td>
<td>processor</td>
<td>Write data in cache</td>
</tr>
<tr>
<td>Read miss</td>
<td>Bus</td>
<td>Request data from cache or memory</td>
</tr>
<tr>
<td>Write miss</td>
<td>Bus</td>
<td>Request data from cache or memory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>perform any needed invalidates</td>
</tr>
</tbody>
</table>
State Transitions:

Based on CPU requests:

- Invalid
  - (CPU read)*
  - Place read miss on bus
  - (CPU write)*
  - Place write miss on bus

- Exclusive
  - (CPU read)*
  - Write back block
  - Place write miss on bus

- Shared
  - (CPU read)*
  - Place read miss on bus

Based on Bus requests:

- Invalid
  - Write Miss for this block

- Exclusive
  - Read Miss on this block

- Shared
  - CPU Miss*
Why does this work?

- Valid slots are in the following states:
  - *Shared*: in multiple caches
  - *Exclusive*: in exactly 1 cache.

- If a read miss occurs on the bus to a slot in *exclusive* state:
  - owning cache makes it *shared*
  - forces subsequent write to require exclusive ownership

- States map to: invalid, valid (and clean), dirty, slots in a uniprocessor cache.