Multiprocessors

|  |  |
| --- | --- |
| **Loosely Coupled (Multi-Computer)** | **Tightly-Coupled (Multi-Processor)** |
| Each CPU has its **own** memory, I/O and OS.  CPUs do **not** share physical memory. | CPUs **share** physical memory and I/O.  Inter-process communication via **shared memory**.  Symmetric multiprocessing (Shared copy of OS)  Critical sections protected by locks & semaphores.  Processes can migrate between CPUs. |

### Multiprocessor Cache Coherency

Since **each** CPU has its own cache, changes in memory need to be reflected in **all** caches that contain data for that memory address.

Solutions are based on **bus watching** or **snoopy** caches.

Snoopy caches watch all bus transactions and update their internal state according to their **cache coherency protocol**:

* Dragon
* **Firefly**
* **MESI**
* Berkeley
* Illinois
* **Write-Once**
* **Write-Through**

Cache-Coherency Protocols

|  |  |
| --- | --- |
| **Protocol** | **Description** |
| Write-Through | Uses **write-through** caches, and a **write-invalidate** protocol:  If a cache observes a bus **write** to a memory location, it invalidates the relevant cache line.  The next time the CPU accesses the memory location or cache line, the data is fetched from memory. |

|  |  |
| --- | --- |
| Write-Once | Uses **write-back** caches, and a **write-invalidate** protocol.  Each cache line is in one of the following states:   1. INVALID 2. VALID Cache line in 1+ caches, cache copy = memory copy. 3. RESERVED Cache line in 1 cache, cache copy = memory copy. 4. DIRTY Cache line in 1 cache, is the **only** correct copy.   Write-Through to VALID cache lines.  Write-Back to RESERVED / DIRTY cache lines.  When a memory location is **read** initially, it enters the cache as VALID.  A cache line in the VALID state changes to RESERVED when written to. Changes are **written through** to memory so that all other caches can **invalidate** their copies.  **Writes** to a RESERVED cache line will use **write-deferred** cycles. The cache line is marked as DIRTY as it is now the **only** correct copy.  Caches must monitor the bus for R/W to its RESERVED / DIRTY lines.   * If a RESERVED cache line is read, it changes the state to VALID. * If a DIRTY cache line is read, the cache must intervene and supply the data onto the bus to the requesting cache. Data is also written to memory. Both caches’ cache lines become VALID. * If a DIRTY cache line observes a write, it becomes INVALID. |

|  |  |
| --- | --- |
| Firefly | Each cache line has a **shared** and **dirty** flag.  When a cache line is **read** into the cache, the other caches will assert a common **shared bus line** if they contain the same cache line.  A cache knows if its cache lines are shared with other caches.  Writes to **exclusive** (not shared) cache lines are **write-deferred**.  Writes to shared cache lines are **write-through** and other caches which contain the same caches are updated together with memory. (write-update)  When a cache line is no longer shared, it needs an **extra** write-through cycle to find out that the cache line is no longer shared.  Sharing may be illusory:   1. The CPU may no longer be using a shared cache line. 2. Process migration between CPUs.   Since there is **no** INVALID state:   * At **reset**, the cache is placed in **miss** mode and a bootstrap program fills the cache with a sequence of addresses making it consistent with memory. * During **normal operation**, a location can be displaced if the cache line is needed to satisfy a miss. The protocol **never** needs to **invalidate** a cache line. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| MESI  (Intel) | Uses a **shared bus signal** so cache lines can enter the cache in the **shared** or **exclusive** state.  Uses a **write-invalidate** protocol.   |  |  | | --- | --- | | **Write-Once** | **MESI** | | 1. INVALID 2. VALID 3. RESERVED 4. DIRTY | 1. INVALID 2. SHARED 3. EXCLUSIVE 4. MODIFIED |   A cache line may enter cache and be placed **directly** in EXCLUSIVE state.  Write-once write-through cycles are **no longer necessary** if EXCLUSIVE. |