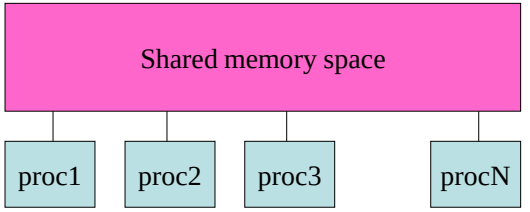
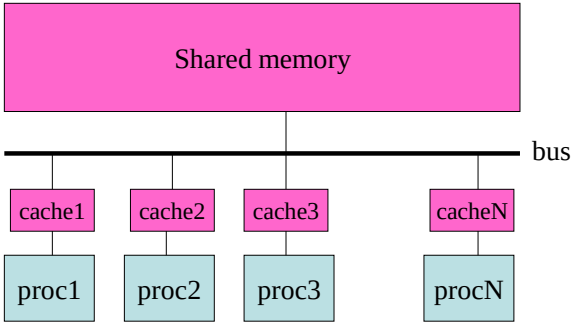
Shared Memory: A Look Underneath



1

Physical Implementation

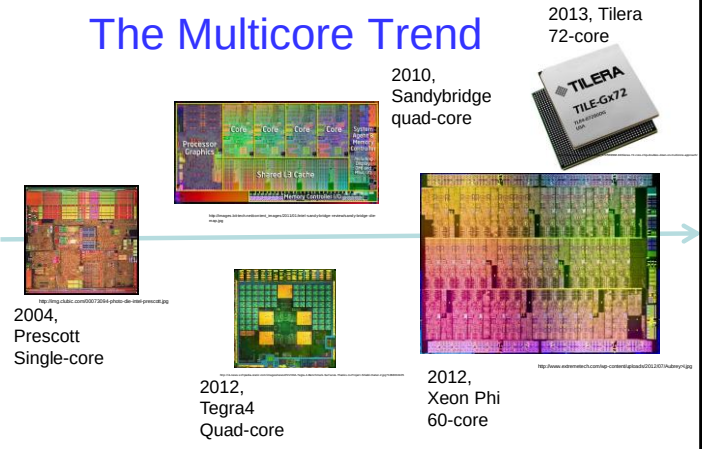


2

Multicore Processors Everywhere

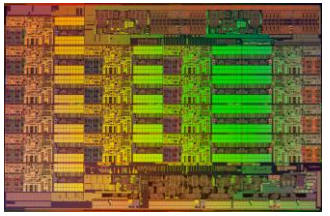
3

The Multicore Trend



4

Haswell Xeon E5 2699 V3



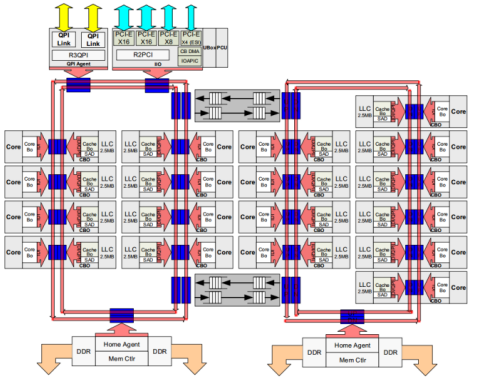
http://cdn4.wccftech.com/wp-content/uploads/2014/09/Xeon-E5-2600-V3-Die.jpg

2.3-3.66 GHz, 145W, 45M L3 cache,
2 sockets, 18 cores, 2 threads,
for a total of 72 hardware threads

5

Haswell: Logical Blocks

14-18 Core (HCC)



http://images.anandtech.com/doci/8730/1%20Die%20Config%2014-18C_678x452.png

6

Shared Memory Implementation

- Coherence - defines the behavior of reads and writes to the same memory location
  - ensuring that modifications made by a processor propagate to all copies of the data
  - Program order preserved
  - Writes to the same location by different processors serialized
- Synchronization - coordination mechanism
- Consistency - defines the behavior of reads and writes with respect to access to other memory locations
  - defines when and in what order modifications are propagated to other processors

7

Coherence

A multiprocessor memory system is coherent if the results of any execution of a program are such that, for each location, it is possible to construct a hypothetical serial order of all operations to the location that is consistent with the result of the execution and

- it ensures that modifications made by a processor propagate to all copies of the data
- program order is preserved for each process in this hypothetical order
- writes to the same location by different processors are serialized and the value returned by each read is the value written by the last write in the hypothetical order

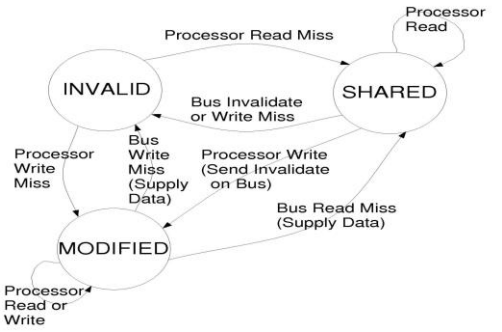
8

Snoop-Based Coherence

- Makes use of a shared broadcast medium to serialize events (all transactions visible to all controllers and in the same order)
  - Write update-based protocol
  - Write invalidate-based (e.g., basic MSI, MESI protocols)
- Cache controller uses a finite state machine (FSM) with a handful of stable states to track the status of each cache line
- Consists of a distributed algorithm represented by a collection of cooperating FSMs

9

A Simple Invalidate-Based Protocol - State Transition Diagram



10

Correctness Requirements

- Need to avoid
  - Deadlock – caused by a cycle of resource dependencies
  - Livelock – activity without forward progress
  - Starvation – extreme form of unfairness where one or more processes do not make forward progress while others do

11

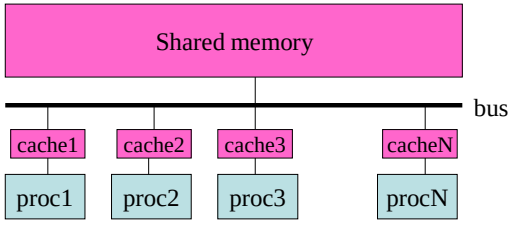
Design Challenges

- Cache controller and tag design
- Non-atomic state transitions
- Serialization
- Cache hierarchies
- Split-transaction buses

12

Snoop-Based or Broadcast Coherence

- Make use of a broadcast medium to manage replicas
- Benefit: Low metadata requirements
- Challenge: High bandwidth requirements

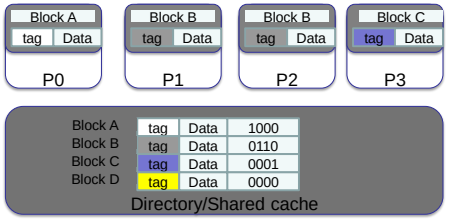


13

Solution: Directory-based Cache Coherence

Directory: maintain per-core sharer information to save bandwidth

Full map: associate sharing vector with tags of shared L2



14

Directory-Based Coherence

- Distribute memory, use point-to-point interconnect for scalability
- Need to manage coherence for each memory line – state stored in directory
  - Simple memory-based (e.g., DASH, FLASH, SGI Origin, MIT Alewife, HAL)
  - Cache-based (linked list (e.g., Sequent NUMA-Q, IEEE SCI)

15

Multiprocessor Interconnects

- Topology
- Routing algorithm
- Switching strategy (circuit vs. packet)
- Flow control mechanism

16

Interconnect Topologies

- Fully connected
  - Single large switch
  - Bus
- Linear arrays and rings
- Multi-dimensional meshes and tori
- Trees
- Butterflies
- Hypercube

17

Switching Strategy

- Circuit-switched: first packet sets up route, subsequent packets follow route without any header processing
- Packet-switched: each packet is independently routed
  - Store-and-forward: each hop receives all packets of a message before forwarding it on
  - Cut-through: each packet forwarded as soon as it is received
  - Virtual cut-through: cut-through routing, but buffer packets when there is contention
  - Wormhole routing: packet spread across multiple hops, in effect holding a circuit open.

18

Metrics

- Hardware cost – number of wires, pin count, length of wires, physical arrangement
- Topology diameter
  - Length of maximum shortest path between any two nodes in the network
- Latency
  - Overhead+routing_delay+channel_occupancy(bandwidth)+contention_delay
- Bandwidth – local, global, bisection
  - Bisection bandwidth
    - Sum of bandwidths of minimum set of channels/links that, if removed, partitions the network into 2 euqal unconnected sets of nodes

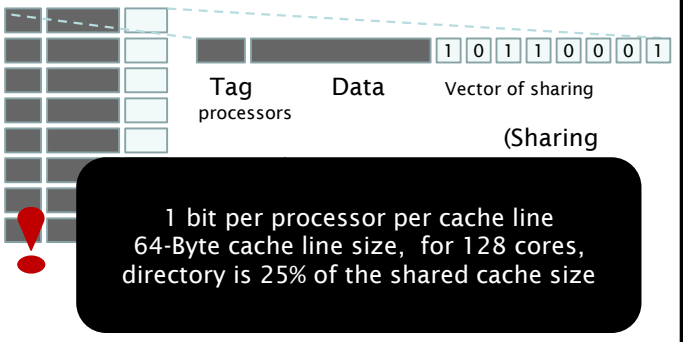
19

Simple Memory-based Directory Coherence

- Advantage
  - Precise sharing information
- Disadvantage
  - Space/storage proportional to PxM
- Work-around for either width or height
  - Increase cache block size
  - 2-level protocol
  - Limited pointer scheme
  - Directory cache

20

Conventional Full Map Directory



21

Cache-Based Directory Coherence

- Home main memory contains a pointer to the first sharer + state bits
- Pointers at each cache line to maintain a doubly-linked list
- Advantage – reduced space overhead
- Disadvantage – serialized invalidates (latency and occupancy)

22

A Framework for Sharing Patterns

- Predictable vs. unpredictable
- Regular vs. irregular
- Coarse vs. fine-grain (contiguous vs. non-contiguous in the address space)
- Near-neighbor vs. long range in an interconnection topology
- In terms of invalidation patterns
  - Read-only
  - Producer-consumer
  - Broadcast/multicast
  - Migratory
  - Irregular read-write

23