M²SI: An Improved Coherency Protocol in CMP

Pengyong Ma, Shuming Chen
School of Computer Science and Technology,
National University of Defense Technology, ChangSha. 410073 China
pyma@nudt.edu.cn

Abstract

Now, most applications are need a CMP. But in CMP, the shared on-chip cache will cause data inconsistent. This paper proposes an improved coherency protocol in CMP: M²SI. This protocol contains four states. It takes full advantage of that all LIDs in CMP can exchange data at high speed. It is at the cost of that the Tags are multi-ports. All the processors’ level one data caches (L1D) are linked on a ring bus. We then take a detailed comparison between this protocol and MESI. Simulation results show that the M²SI protocol has an improvement about 30% comparing with MESI. Comparing with MID[1] protocol, M²SI has only four states and need only two bits to denote these states.

1 Background

In chip multiprocessor (CMP), multi-copies of data can easily get inconsistent. Generally, there are two cache coherency protocol, snoop and directory. Snoop’s drawback is its scalable limitation and the shared cache’s (in this paper, L2 is shared) bandwidth. Directory needs large memory to store the directories. In this paper, we propose an improved coherency protocol M²SI which based on the MID protocol. In CMP, one L1D can rapidly get a data block from others without interfere the shared L2 as little as possible. One processor can read others Tag and then the write accesses can accurately know which L1D should be updated. These can reduce the load of L2 and enhance the chip’s performance.

2 Model Proposed

Figure 1. States transition diagram for the M²SI protocol.

In M²SI protocol, there are two bits to distinguish a block state, M: master bit, and I: valid bit. There are four states to distinguish a block state. “M”: master line and the data is clean; a data line has several copies in CMP, A block can have several slave copies, but it has only one master copy. State “MM”(M²) means this block is a master block and the data is Modified(dirty). State “S” means this block is a slave block. State “I” means this block is an invalid line. The L1D states transition diagram for the M²SI protocol is shown in Fig. 1. Comparing with MID[1] protocol, M²SI has only four states and need only two bits to denote these states.

Table 1. meanings of data cache symbol

<table>
<thead>
<tr>
<th>State</th>
<th>M, I bits</th>
<th>meanings</th>
</tr>
</thead>
<tbody>
<tr>
<td>MM</td>
<td>11</td>
<td>master block, modified data</td>
</tr>
<tr>
<td>M</td>
<td>10</td>
<td>master block, clean data</td>
</tr>
<tr>
<td>S</td>
<td>01</td>
<td>It is a slave block</td>
</tr>
<tr>
<td>I</td>
<td>00</td>
<td>This block is an invalid line</td>
</tr>
</tbody>
</table>

The L1D states transition diagram for the M²SI protocol is shown in Fig. 1. Comparing with MID[1]
protocol, M^2SI need only two bits to denote these states. A/B is marked on the line, it means the controller detects an affair A and then produces action B. “/” means that the controller doesn’t produce any actions. CrRd: a read access from own core. *: it denotes that this access hit other core’s L1D. CrWr: a write access from own core. ringRd: a read access coming from other core by ring bus. ringWr: a write access coming from other core by ring bus. ringM/ringMM: a Master symbol shift command from other core by ring bus. When a cache line with state M/MM is evicted by a new cache line, if other slave copy’s state turns to M/MM, It doesn’t need to write back this evicted line. ringMM: as same as ringM, but the evicted line state is MM. L2Rd: a read request to shared L2. Flush: a command from L2 to invalid L1D’s data line. WrBk: write back dirty data to L2.

2.1. Read Access
When a read access coming with data’s address, cache controller compares the address with the data in all Tags, if it hit its L1D, the succeeded steps are same as in single core chip; if the access doesn’t hit itself L1D but other core’s, the core will send request to the near core which includes the data. The request word has three segments: Source core ID, Dst core ID, and data address. The response word has three parts: Dst core ID, Ready and the data. If a read request doesn’t hit any L1Ds, the action is same as in single core.

2.2. Write Access
The write access is similar as reading. If the write access hits its L1D only, the responding process is same as in single core chip; if it hits other L1Ds, the core will send the write request word on the ring. The word contains four segments: destination core ID, width, data and address.

2.3. Master Symbol Changing
In M^2SI protocol, one data block may have several same copies in L1Ds. So when the cores receive a flush or a snooping request, it needs only the core with Master state to return the dirty data, others just clear their valid bits. When a “M”/”MM” line is evicted out of L1D, if there are other cores have the slave copies, it doesn’t need to write back this line to L2. The only work is to set one slaver copy’s to “M”/”MM” state.

3. Simulation
We establish a CMP model with 4 cores; the core is “YPHT-DSP” 700[3]. Every core has private level one data cache and instruction cache, Level two cache is shared by four cores.

![Simulation results](image)

Fig. 2 shows the results of six benchmarks running. They run in the single core, 4 cores with MESI protocol and M^2SI protocol respectively. The y-axis shows the run-time of the programs and the numbers came from detailed simulation.

References

[1] Pengyong Ma, Shuming Chen, MID a Novel Coherency Protocol in Chip Multiprocessor, CIT’06, P50