

INITIAL STATES OF CACHE LINES

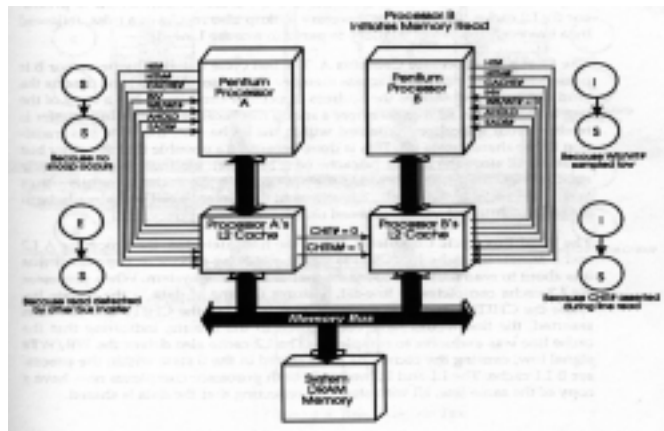
Scenario	Process Complex A Status	A's L2 Cache Line State	A's L1 Cache Line State
1	Cache state after reading target line from system memory.	E	S
2	Cache state after 1st internal write to target line.	M	E
3	Cache state after 2nd and subsequent internal writes to target.	M	M

3

READ SCENARIO ONE

1. INITIAL CACHE STATES ARE: COMPLEX A-L1 IN S STATE AND A-L2 IN E STATE.
2. A READ MISS OCCURS IN COMPLEX B (BOTH B-L1 & B-L2). COMPLEX B REQUESTS FOR A CACHE LINE-FILL.
3. AN ADDRESS SNOOP HIT IS DETECTED BY COMPLEX A.
4. A L2 CACHE ASSERTS *CHIT#* TO NOTIFY COMPLEX B THAT THE DATA IS SHARED. AFTER THE READ, THE CACHE LINE IS SET IN S STATE.
5. B-L2 CACHE DRIVES *WB/WT#* LOW TO CAUSE THE B L1 CACHE LINE TO BE STORED IN S STATE.

4



5

READ SCENARIO TWO

1. INITIAL CACHE STATES ARE: COMPLEX A-L1 IN E STATE AND A-L2 IN M STATE.
2. A READ MISS OCCURS IN COMPLEX B (BOTH B-L1 & B-L2). COMPLEX B REQUESTS FOR A CACHE LINE-FILL.
3. AN ADDRESS SNOOP HIT IS DETECTED BY COMPLEX A.
4. *CHIT#* AND *CHITM#* ARE LOWERED TO NOTIFY B-L2 TO BACK OFF. B-L2 CACHE ASSERTS *BOFF#* TO FORCE PROCESSOR B TO HALT THE CURRENT BUS CYCLE.
5. A-L2 USES *AHOLD* & *EADS#* TO INFORM PROCESSOR A TO SNOOP THE ADDRESS FOR THE READ FROM THE BUS MASTER.

6

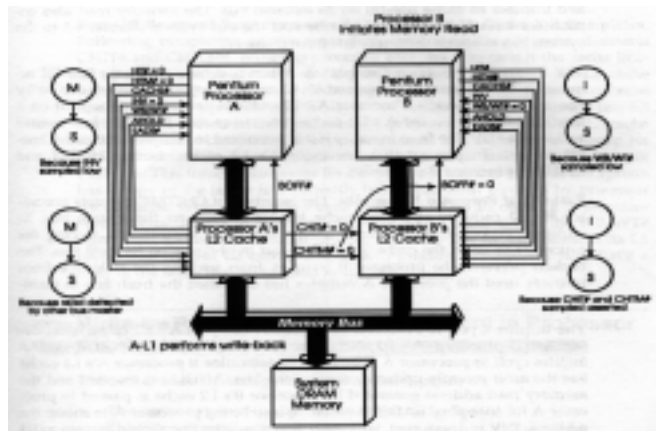
READ SCENARIO THREE

1. INITIAL CACHE STATES ARE: COMPLEX A-L1 IN M STATE AND A-L2 IN M STATE.
2. A READ MISS OCCURS IN COMPLEX B (BOTH L1 & L2). COMPLEX B REQUESTS FOR A CACHE LINE-FILL.
3. AN ADDRESS SNOOP HIT IS DETECTED BY COMPLEX A.
4. *CHIT#* AND *CHITM#* ARE LOWERED TO NOTIFY B-L2 TO BACK OFF. B-L2 CACHE ASSERTS *BOFF#* TO FORCE PROCESSOR B TO HALT THE CURRENT BUS CYCLE.
5. A-L2 USES *AHOLD* & *EADS#* TO INFORM PROCESSOR A TO SNOOP THE ADDRESS FROM THE BUS MASTER.

9

6. PROCESSOR A ASSERTS *HIT#* & *HITM#* TO NOTIFY A-L2 CACHE THAT THE INQUIRY RESULTS IS A HIT AND A-L2 CONTAINS STALE DATA.
7. PROCESSOR A PERFORMS A WRITE-BACK THROUGH A- L1 TO UPDATE MEMORY.
8. A-L1 CHANGES THE LINE STATE FROM M TO S. A-L2 TRANSITIONS THE CACHE LINE FROM M TO S.
9. A-L2 DEASSERTS *CHIT#* & *CHITM#* AND ALLOWS COMPLEX B TO PROCEED THE MEMORY FETCH.
10. COMPLEX A DETECTS A SNOOP HIT AGAIN AND ASSERTS *CHIT#*, TELLING B-L2 THE SHARED LINE IS READY IN MEMORY.

10



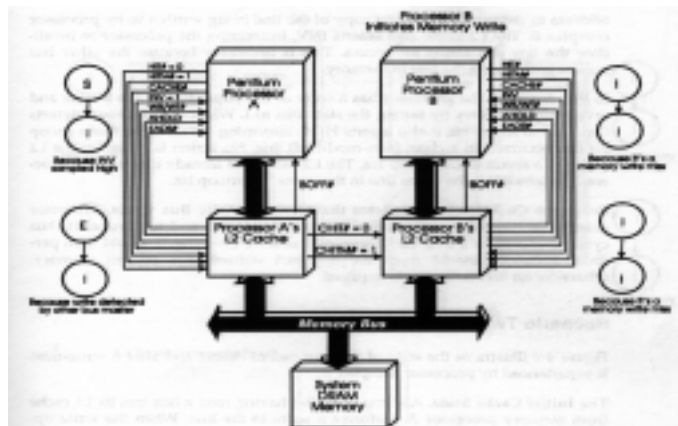
INITIAL STATES OF CACHE LINES

Scenario	Process Complex A Status	A's L2 Cache Line State	A's L1 Cache Line State
1	Initial cache line state after reading target line from system memory.	E	S
2	Cache line state upon completion of 1st internal write to target line after it's placed in the L1 cache.	M	E
3	Cache line state after 2nd and subsequent internal writes to target line by the processor.	M	M
4	Cache line state after another bus master reads from target line.	S	S

WRITE SCENARIO ONE

1. INITIAL CACHE STATES ARE : COMPLEX A-L1 IN S STATE AND A-L2 IN E STATE. COMPLEX B-L1 & B-L2 IN I STATE.
2. A WRITE MISS OCCURS IN COMPLEX B (BOTH L1 & L2). B-L2 REQUESTS FOR A MEMORY UPDATE.
3. A-L2 DETECTS AN ADDRESS SNOOP HIT. IT TRANSITIONS ITS CACHE LINE FROM E TO I.
4. A-L2 ASSERTS *AHOLD* & *EADS#*, PASSES ADDRESS TO A-L1, AND ASSERTS *INV* TO INVALIDATE THE LINE IF A SNOOP HIT OCCURS (A-L1 TRANSITIONS THE LINE FROM S TO I).
5. COMPLEX B COMPLETES THE WRITE OPERATION (W/T ALLOCATE-ON-WRITE).

13



14

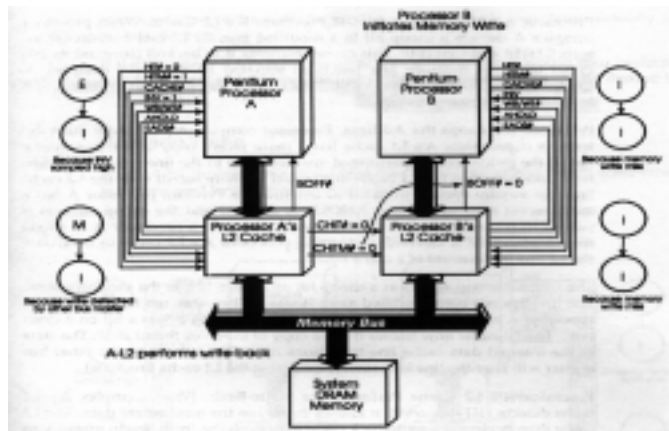
WRITE SCENARIO TWO

1. INITIAL CACHE STATES ARE : COMPLEX A-L1 IN E STATE AND A-L2 IN M STATE. COMPLEX B-L1 & B-L2 IN I STATE.
2. A WRITE MISS OCCURS IN COMPLEX B (BOTH L1 & L2). COMPLEX B REQUESTS FOR A MEMORY UPDATE.
3. AN ADDRESS SNOOP HIT IS DETECTED BY COMPLEX A.
4. *CHIT#* AND *CHITM#* ARE LOWERED TO NOTIFY B-L2 TO BACK OFF. B-L2 CACHE ASSERTS *BOFF#* TO FORCE PROCESSOR B TO SUSPEND THE CURRENT BUS CYCLE.
5. A-L2 USES *AHOLD#* & *EADS#* & *INV#* TO INFORM PROCESSOR A TO SNOOP THE ADDRESS FOR THE WRITE FROM THE BUS MASTER.

15

6. PROCESSOR A ASSERTS *HIT#* TO NOTIFY A-L2 CACHE THAT THE INQUIRY RESULTS IN A HIT AND A-L2 CONTAINS THE MOST RECENT DATA.
7. A-L1 CHANGES FROM E TO I.
8. A-L2 PERFORMS WRITE-BACK TO MEMORY, AND IT TRANSITIONS THE CACHE LINE FROM M TO I.
9. A-L2 DEASSERTS *CHIT#* & *CHITM#* AND ALLOWS COMPLEX B TO PROCEED THE MEMORY WRITE (W/T ALLOCATE-ON-WRITE).

16



17

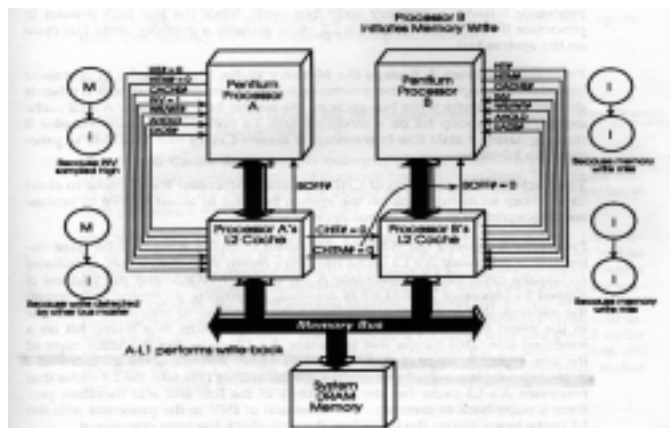
WRITE SCENARIO THREE

1. INITIAL CACHE STATES ARE: COMPLEX A-L1 IN M STATE AND A-L2 IN M STATE. COMPLEX B-L1 & B-L2 IN I STATE.
2. A WRITE MISS OCCURS IN COMPLEX B (BOTH L1 & L2). COMPLEX B REQUESTS FOR A MEMORY WRITE.
3. AN ADDRESS SNOOP HIT IS DETECTED BY COMPLEX A.
4. *CHIT#* AND *CHITM#* ARE LOWERED TO NOTIFY B-L2 TO BACKOFF. B-L2 CACHE ASSERTS *BOFF#* TO FORCE PROCESSOR B TO SUSPEND THE CURRENT BUS CYCLE.
5. A-L2 USES *AHOLD* & *EADS#* & *INV* TO INFORM PROCESSOR A TO SNOOP THE ADDRESS FOR THE READ FROM THE BUS MASTER.

18

6. PROCESSOR A ASSERTS *HIT#* & *HITM#* TO NOTIFY A-L2 CACHE THAT THE INQUIRY RESULTS IN A HIT AND A-L2 CONTAINS STALE DATA.
7. PROCESSOR A PERFORMS A WRITE-BACK THROUGH A-L1 TO UPDATE THE MEMORY.
8. A-L1 CHANGES FROM M TO I.
9. A-L2 TRANSITIONS THE CACHE LINE FROM M TO I.
10. A-L2 DEASSERTS *CHIT#* & *CHITM#* AND ALLOWS COMPLEX B TO PROCEED THE MEMORY WRITE (W/T ALLOCATE-ON-WRITE).

19

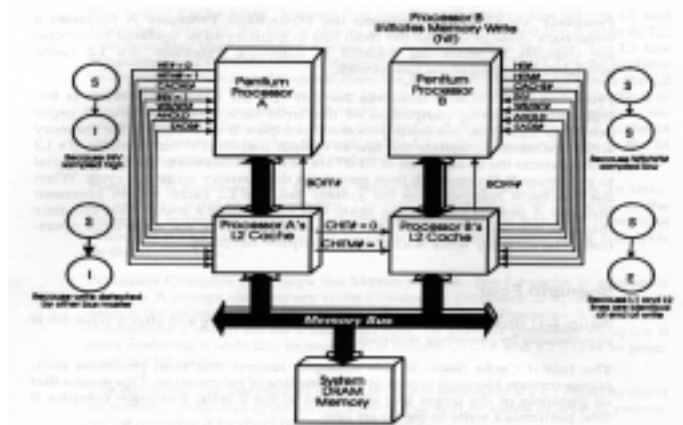


20

WRITE SCENARIO FOUR

1. INITIAL CACHE STATES ARE: COMPLEX A-L1 & A-L2 IN S STATE. COMPLEX B-L1 & B-L2 IN S STATE.
2. B-L1 & B-L2 EXPERIENCE A WRITE HIT. COMPLEX B INITIATES A MEMORY WRITE (WITH WRITE-THROUGH).
3. THE B-L1 & B-L2 CACHE LINES ARE UPDATED AND SET IN S AND E STATES, RESPECTIVELY.
4. A-L2 SNOOPS THE MEMORY WRITE. IT ASSERTS *AHOLD*, *EADS#* & *INV*, AND PASSES THE ADDRESS FOR A-L1 TO SNOOP THE ADDRESS.
5. A-L2 INVALIDATES THE LINE.
6. A-L1 INVALIDATES THE LINE IN CASE OF A SNOOP HIT.

21



22