Steps in Accessing Memory

• Compute effective address
• Split effective address into page number and offset
• Use page number as an index into the page table. (Check if too big.)
• If Resident bit is clear, generate a page fault.
• If Resident bit is set, get page address from the page table.

Steps in Accessing Memory

• Concatenate page address with offset to create the physical address.
• For direct L1 cache, get the middle bits of the physical address to determine which line to check.
• Check the tag value of the specified line to see if it matches the upper bits of the physical address.
• If there is a match, return the specified value from the cache.

Steps in Accessing Memory

• If the tag field does not match, check the second level cache.
• For n-way set associative L2 cache, get the middle bits of the physical address to determine which set to search.
• Search the tag values of all n lines in the specified set to see if any match the upper bits of the physical address.
• If there is a match, return the specified value from the cache.

Steps in Accessing Memory

• If none of the n tag fields for the set match the upper bits of the physical then determine which of the lines in the specified set of the L2 cache is the oldest. This line will hold the new data.
• If this line is dirty, write this block to RAM.
• Get the physical address data from RAM.
• Copy the block of data from RAM into the L1 and L2 caches. Send the CPU the data requested.