5.5   MESI adds the exclusive state to the Snoopy protocol, which thereby decreases traffic when we know that one specific cache exclusively has the data. We don't need to send invalidates, and according to the table, invalidating costs 15 cycles, meaning that is how much we save for both loops, as writing to both A and B would no longer mandate this.

5.10   a) By sending invalidates directly to the requester, the total amount of traffic in the protocol is lowered, as the directory no longer has to route messages to the requester, upping the system response time.

b) By having the owner directly forward the data to the requester, this will speed up the response time, since the requester no longer needs to wait for the memory to be written.

c) By having one of the sharers directly forward the data to the requester, this will have the same result as previously stated, speeding up the response time once again.

5.28   a) The cache line is split between the 4 processors, each computing a fourth of the total sum.

b) for (int p= 0; p<=3; p++) { // Each iteration of is executed on a separate processor.

sum[p] = 0;

for (int i = 0; i < n/4; i++) // n is size of word_count and is divisible by 4

ind = p + 4*i

sum[p] = sum[p] + word_count[ind];

}

total_sum = sum[0] + sum[1] + sum[2] + sum[3] //executed only on processor.

c) We can remove false sharing by giving each processor a different value of p.

5.29   In this case, that would mean that the cache block was very recently just kicked out of the cache, and being requested for once again. The directory should check to see if the cache block has changed its state at all (to shared) before sending the block back to the P1 cache.