COEN 313 HW 1

A one GHz CPU gets 4 read misses in back to back. They are to integers at addresses 24(25,26,27) , 56(57,58,59), 88(89,90,91), and 120(121,122,123).

- Assume the CPU has a 32-byte per line direct mapped cache (large enough to hold the result of these misses in separate cache lines).
    - The cache is a BLOCKING cache and does not use early restart or critical-word-first accesses.
- The width of the bus between the CPU and the memory controller is 8 bytes, and the width of the bus between the memory controller and the DDR memory is also 8 bytes.
- For simplification assume the CPU, the memory controller and the DDR memory are running at the same speed (actually not realistic).

1. Trace the trip of a single miss from the CPU miss until the data comes back to the cache.

I. Add up the delays encountered during this trip in terms of cycles. You will have to add some cycle delays on your own inside the cache controller and memory controller. Use your best engineering judgment.

CPU:

1 cycle	cache access
1 cycle	store request in outstanding buffer

Bus:

5 cycles	command on bus

Memory Controller:

1 cycle	register & store command
3 cycles	arbitration
2 cycles	arbitrate returns
4 cycles	CPU - memory data transfer

Memory:

10 cycles	precharge
10 cycles	activate
10 cycles	read command
4 cycles	read data

51 cycles total

(Because the CPU, memory controller and DRAM are assumed to be running at the same speed, we maintain the CPU cycles provided in class and divide the memory controller and DRAM cycles by 5 because it was assumed to be 5X slower in the example)

2. Add up the delays for the four misses when they happen to fall in four different rows of the same DDR bank.

Because we have a single bank with four different rows, we aren't able to parallel activate the rows, thus we have to add up each of the delays, or multiply by 4.

4 misses * 51 cycles = 204 total cycle delay

3. Solve part (1) when the cache uses early restart and critical-word-first techniques.

Early Restart: Request words in normal order, send missed work to the processor as soon as it arrives.

Critical Word First: Request missed word from memory first, send it to the processor as spoon as it arrives

Because we are dealing with integers, which are 4 bytes, each miss is a combined total of 16 bytes, which is half of the 32 bytes per cache line. This means that the latter half of the cache line isn't useful for attaining the data we want. Thus, by using the early restart and critical-word-first techniques, we can save 2 cycles for the CPU - memory data transfer per miss.

Thus, we would reduce the total delay for a single miss to 49 cycles.

4. Solve part (2) when the uses early restart and critical-word-first techniques and the cache is nonblocking (can allow 4 misses outstanding).

Using early restart and critical-word-first techniques, we are now saving 2 cycles for CPU - memory data transferring. Using a non-blocking cache allowing 4 misses outstanding, we can now overlap the misses to save penalty time. However, we still cannot overlap the 34 cycles of DDR processing.

Miss 1: 1-13 CPU + MemC, 14-47 Mem, 48-49 data delivery

Miss 2: 3-15 CPU + MemC, 48-81 Mem, 82-83 data delivery

Miss 3: 5-17 CPU + MemC, 82-115 Mem, 116-117 data delivery

Miss 4: 7-19 CPU + MemC, 116-149 Mem, 150-151 data delivery

Total cost: 151 penalty cycles

5. Solve part (4) for when the four misses happen to fall in the four different banks of the DDR.

Using 4 different banks, we can activate, precharge, and read independently, but all banks still share the same command and data buses, in which no conflict should take place.

Thus, the total memory cycles no longer need to be stalled for each miss.

Miss 1: 1-47 cycles miss penalty, 48-49 cycles data delivery

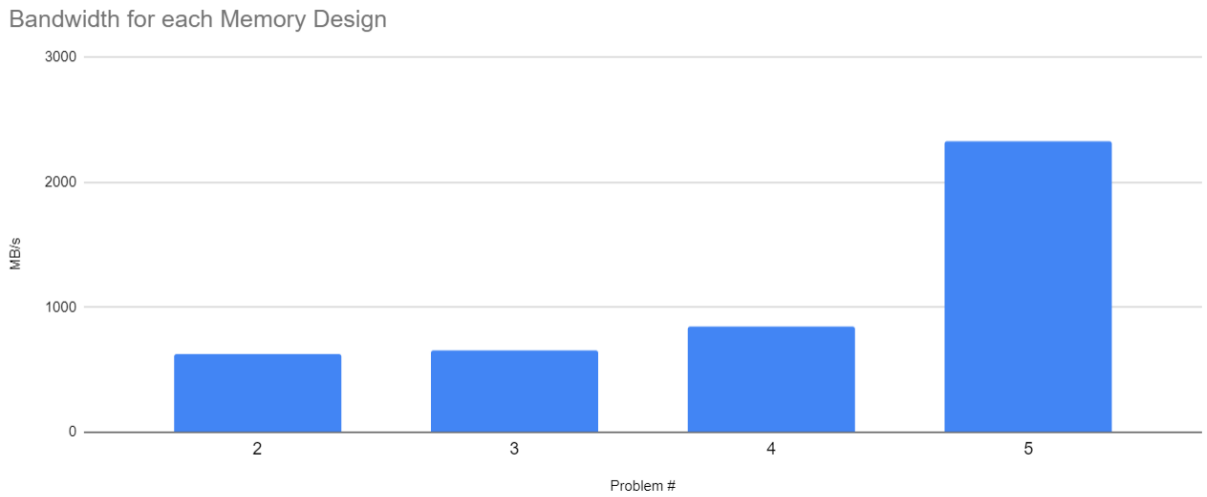
Miss 2: 3-49 cycles miss penalty, 50-51 cycles data delivery

Miss 3: 5-51 cycles miss penalty, 52-53 cycles data delivery

Miss 4: 7-53 cycles miss penalty, 54-55 cycles data delivery

Total cost: 55 penalty cycles

6. Draw a histogram (use Excel??) comparing the bandwidth achieved for parts 2 through 6.



For DDR delays, use the table shared in the course slides.