Homework Assignment 6

CS/ECE 6810: Computer Architecture
April 11, 2018

Main Memory

Due Date: April 18, 2018.
(90 points)

Important Notes:

- Solutions turned in must be your own. Please, mention references (if any) at the end of each question. Please refrain from cheating.
- All solutions must be accompanied by the equations used/logic/intermediate steps and assumptions. Writing only the final answer will receive **zero** credit.
- All units must be mentioned wherever required.
- Late submissions **(after 11:59 pm on 03/27/2018)** will not be accepted
- All submissions must be in PDF only. Scanned copies of handwritten solutions will not be accepted as a valid submission.

1. **Memory Scheduling Mechanisms (20 points):** For the following memory access pattern, estimate when each memory access completes for two different scheduling mechanisms: **open-page policy** and **close-page policy**. You are allowed to reorder requests already waiting in the memory controller. The access pattern only specifies the row being touched. All accesses are to the **same** bank. Assume that bus latencies are zero. Assume that the bank is already pre-charged at time 0. Assume that pre-charge takes 20 ns, loading a row buffer takes 20 ns, and cache line transfer to output pins also takes 20 ns.

Row being accessed	**Arrival time at memory controller**	**Open-page**	**Close-page**
X	10 ns	50 ns	50 ns
X	70 ns	90 ns	110 ns
Y	90 ns	150 ns	190 ns
X	100 ns	210 ns	130 ns
Y	180 ns	270 ns	210 ns

2. **Memory System Design (20 points):** Modern systems have processors with four memory channels. Assume that a memory channel has 64 bits for data and 24 bits for address and command. Assume that a memory channel is connected to eight x8 memory chips. Consider a new memory system that shrinks the data bus of the memory channel to 32 bits. Assume that such a channel would be connected to four x8 memory chips. Mention **two pros** and **two cons** of the new memory system (Be very concise).

   Solution:
   When you mention pros and cons, you must do an apples-to-apples comparison. For example, the capacity of the new system per chip is low because there are half as many chips, but you can get the same capacity as before by adding more DIMMs per channel, and hence more ranks. What truly decreases is capacity to pin ratio (see cons second point). Cost of the new chip decreases, but cost to capacity ratio, which is what matters increase. Energy per cache line is lower because you need to activate 4 chips instead of 8. What essentially happens is you have a smaller row buffer in the new system, thus reducing the advantage of row buffer hit rates in case the application has that. If not, its a better system, because to get the same capacity, you would need more ranks, and hence more banks, thus allowing for parallelism.

   Pros:

   i The energy per cache line fetch is reduced (only 4 chips activated and precharged).

   ii The parallelism in the system is higher (more ranks and hence more banks). This can lead to better bus utilization and lower queuing delays.

   Cons:

   i Fetching a cache line takes longer (16 transfers on the bus instead of 8).

   ii More processor pins are required for a given memory capacity and memory rd/wr bandwidth because the overhead of the address/command signals goes up (24 addr/cmd for every 32 bits of data).

3. **Virtually Indexed Physically Tagged Cache (20 points):** Assume that the OS uses a minimum page size of 4 KB. Assume that your L1 cache must be 8-way set-associative. If you're trying to correctly implement a virtually indexed physically tagged cache (with no additional support from the OS), what is the largest L1 cache that you can design?

   Solution:
   OS page size = 4 KB $\therefore$ OS page offset = 12
   Since for VIPT cache, the OS page offset must coincide with block_offset + index_bits, therefore 12 bits are available for block offset and index bits together. Since the L1 is 8-way set associative:

   $$Maximum\ capacity\ =\ 8\ *\ 2^{12}B = 32KB$$

4. **DRAM Control (10 points):** Express the latency of a memory load as a function of tRAS, tRP, tRCD, tCL in the following situations:

   i The correct row is already placed in the row buffer

   ii Another row is already placed in the row buffer.

Solution:

i tCL

ii tRP + tRCD + tCL

5. **DRAM Control Tasks Request Scheduling (20 points):** Find the total number of commands sent by an in-order command scheduler for the given sequence of memory addresses for reads, using the following address mapping scheme:

   Address = row(12):bank(3):rank(1):channel(0):column(16)
   Assume that all banks are initially precharged.

   Addresses:
   00040108
   01040101
   00161804
   01040104

   Solution:

   i ACT, RD: RAS, CAS

   ii PRE, ACT, RD: PRE, RAS, CAS

   iii ACT, RD: RAS, CAS

   iv RD: CAS