

Section 1 Computer Architecture.

A multicore processor system consists of three core processors P_1, P_2, P_3 , and three memory modules M_1, M_2, M_3 . The system is connected in a 3×3 cross bar structure, as shown in Figure. Let us assume that core P_i references memory module M_j with probability f_{ij} , where $i=1,2,3$ and $j=1,2,3$. By memory reference we mean the following cycle: a core processor sends a request through the cross bar to access a memory module, which then sends the requested data to the core. Assume that the time required to complete each memory access is t clock cycles, and the time for the data transfer through each cross bar switch is c clock cycles. Note that when two or more cores reference the same memory module, then there is a memory conflict which can be resolved by allowing one core through, however, the other cores must wait until the busy core completes its access. The choice is made randomly by the operating system, but can be also made on a priority basis.

- 1.1 Find the *expected* memory reference time in clock cycles for the above multicore processor model.
- 1.2 How is the above result going to be affected if there is a delay of no more than d cycles every time (the next in line) a core has to wait for the busy core to access a memory module. You may have to allow pre-emption of a busy core accessing, resuming work later.
- 1.3 To reduce the operating system overhead, design a hardware control mechanism based on priority queues to resolve memory access conflicts. For example, each memory module may have a queue or small cache that can be accessed by all corresponding cross bar switches which can deposit in the queue their requests.
- 1.4 Provide a simulation model for the behavior of this multicore processor system and its memory reference patterns and conflict resolution when there is a priority policy for core-memory accessing. Provide results for some test cases.

