IIT JODHPUR

Minor Examination: Network-on-Chip (Sept'24) [OPEN-BOOK]

Guidelines (Total time: 120 minutes, Maximum Marks: 25):

- Please read the question paper very carefully. **NO clarification is required in any question.** In case of any doubt, assume whatever you wish to and state that in your answer.

---

1. One important parameter in deciding NoC topology is the average hop (meaning number of times the message has to travel on physical links) length (H_{avg}) considering traffic patterns generalized across multiple application scenarios. Consider the wire/link delay as 6 ns, individual router delay as 16 ns and a channel bandwidth of 1 GB/s. If we know that the typical message over this network has header flit as 11 bits, payload (body) as 116 bits and 1 tail flit, find H_{avg} such that the average network delay is 120 ns? If we wish to reduce this delay to 80 ns, which component of delay is the easiest to optimize here? Why? [2 + 2 Marks]

2. Consider that we have a multi-core arrangement with a configuration of 64 cores (CPUs) and 64 caches. Each core has an individual cache associated with it under distributed memory management system. Which network topology you would decide to implement the interconnection of these cores and caches in order to achieve the shortest latency for the overall network (i.e., processor-cache system)? Draw your topology. Assume that the router available to you incurs a minimum delay of 10 ns and the physical link/wires between two routers incur a delay of 5 ns, calculate the value of the minimum latency that your network topology can provide for a data transfer from the first core to the last core? Design a proper routing algorithm to achieve the minimal network latency in the above scenario? [2 + 2 + 3 Marks]

3. For a particular usage scenario, it is required to have reliable data transmission at the cost of accommodating the delay in transmission to some extent. In the design of NoC for this application, we have a total of 64 routers arranged in 2D-mesh configuration. Which flow control is most suitable for this application? To ensure reliable data transmission, each message is required to receive an acknowledgement of 3 bits from the destination router after the header flit is received at the destination. Consider that the header of the message is 6 bits, payload(body) is 89 bits and tail flit is 1 bit. Find out the total number of clock cycles for transmitting these messages: (a) 13^{th} to 47^{th} router (b) 9^{th} to 59^{th} router? [1 + 2 + 2 Marks]

4. Design a router that is having the traffic pattern as shown below in order to minimize the network latency? You can consider that all other combinations of data movement are NOT allowed.

Source Dir.	Destination Dir.	Message Length (H + B + T)
EAST	NORTH	1 byte (1 + 6 + 1)
EAST	SOUTH	1.5 byte (1 + 10 + 1)
SOUTH	WEST	2 byte (1 + 14 + 1)
SOUTH	EAST	1.5 byte (1 + 10 + 1)
WEST	NORTH	2 byte (1 + 14 + 1)
NORTH	SOUTH	2 byte (1 + 14 + 1)
NORTH	WEST	1.5 byte (1 + 14 + 1)

Consider the buffers available to you have a capacity of 4 bytes each and the crossbar (point-to-point) style switching is employed. Take your assumptions in case of any missing information that you may need. (Note: Router design means drawing the schematic showing the channels, buffers, different input/output ports etc.) [4 Marks]

5. Consider that in a NoC design, we have 5 input directions (E, W, N, S and L) and the respective 5 output directions (E, W, N, S and L). For the input ports, the first 4 have virtual channels (VCs) allocated to them. We have 32 x 32 mesh configuration (meaning 32 IPs and 32 routers) and a crossbar switching style utilized in this design. Even though connections are available, we need to have switch allocation through some arbitration logic. Design 2 different types of arbiters for this purpose and compare their performance in terms of network throughput and latency? You need to provide the inputs, outputs and internals of your arbiter. You can design the arbiters as per your choice. However, for hints, it is given to you that fixed-priority and round-robin are two popular styles of arbiter implementation. [5 Marks]
