EE 789: Mini-Project 1

Madhav P. Desai

March 24, 2025

1 A switch

An NxM switch provides functionality to switch data from 1 of the N input ports to 1 of the M outputs. Let the input ports be numbered 0,1,2,..., N-1 and the output ports be numbered 0,1,2,...,M-1. Input packets arrive at the switch. The packet length is a multiple of 4-byte quantities (words). The first word in the packet is called the *header*, and its format is as follows:

Bits	Interpretation
[31:24]	Destination
[23:8]	Length
[7:0]	Sequence-id

When data enters the switch from a particular input port, the switch examines the header to figure out the destination of the packet. The switch is then responsible for sending the incoming packet to the destination output port in its entirety.

2 An output-queued switch

Two kinds of switches can be envisaged. A 2x2 output-queued switch has the structure shown in Figure 1.

Incoming data at an input demultiplexor is written into the appropriate destination queue. Note that if the destination queue has no room, then it will block and the input demultiplexor will have to wait until it is free.

At each output port, there is an arbiter which checks the two queues that it is managing. It serves the queues in a fair manner (that is, in the long term, neither of the two queues will get favoured). The packet in the selected queue is then sent out in its entirety, on to the output port.

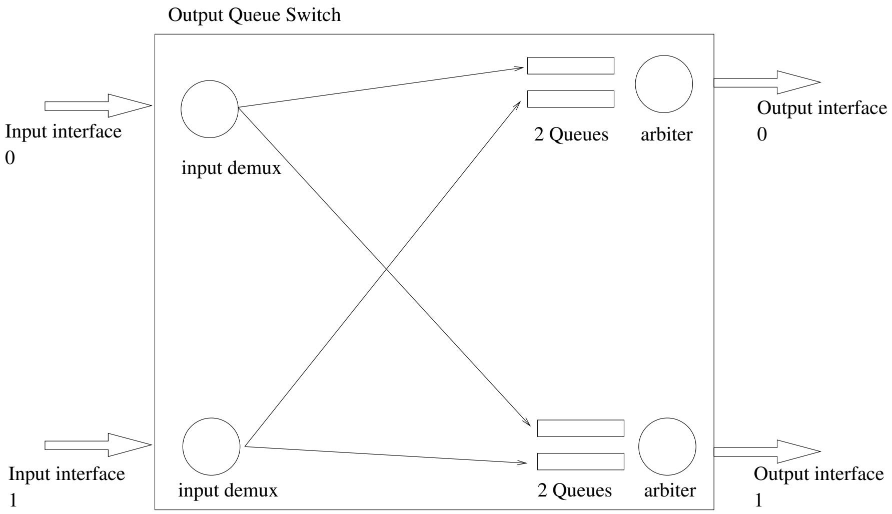

Figure 1: 2x2 Output Queued Switch

3 The mini-project

You will design a 4x4 switch, which can handle packets with a maximum size of 256 bytes. You will validate and characterize your switch for the following kinds of traffic:

- Data comes in from a single port and goes out to a single port.
- Data comes in from both ports and goes out to the same destination port.
- Data comes in from both ports and is distributed across the destination ports (incoming packet is equally likely to go to any of the destination ports).

4 How to go about it?

You will first figure out the algorithm, then implement the algorithm in C/Aa. You will write a test-bench to confirm that the algorithm will function correctly. Please ensure that the test-bench is sufficiently smart so that you can eventually use the same test-bench to characterize the behaviour of the switch for th three kinds of tests outlined above.

5 How will we help you?

We will present a mini 2x2 switch implementation with a test-bench that handles the third test scenario outlined above. We will show you how to write a build file (Makefile), the mechanics of verification, the generation of the VHDL and the use of the VHDL simulator to complete the validation and characterization of the switch.

You will then be responsible for the extension of this 2x2 switch description to a 4x4 switch.