EE 314 Term Project

Spring 2022
FPGA Implementation of Simple Quality of Service (QoS)
based Queuing

1 Introduction

In today's computer and wireless networks, the amount of data transferred from one point to another has increased tremendously. However, it is important to control the data traffic to ensure that overall system performance is optimized according to some criteria such as reliability or low latency requirements. For instance, some data should be delivered to the destination immediately after it is received in order to minimize the latency; however, some data may have higher priority so that it should be delivered without any information loss. Basically, Quality of Service (QoS) networks are developed to control the data traffic to optimize the network capacity. In this project, you are asked to implement a very simple queuing algorithm using Verilog HDL on FPGAs.

2 Project definition

2.1 Project Summary

You are going to implement your project on the FPGA boards that you will borrow from us. In the project, you will manage data traffic between two nodes, where four different message types are transmitted as shown in Figure 1. The input of the system will be a 4-bit sequence (call it as a packet) which is generated by the FPGA buttons. You may use one button for "0" and one for "1". By pressing the buttons, you will generate a bit sequence. In addition, before creating each bit sequence, you should press a start button to indicate that a packet is to be transmitted. The first two bits of the packet ("B0B1") are the header bits, and based on the header bits, you should fill the corresponding buffer with the remaining value in decimal form. As can be seen in Figure 1, there are four buffers, which are represented with different colors. The buffer depth is 6 so you can store at most 6 packets at each buffer. If a new packet arrives at a buffer, which is full, then the oldest data is deleted (packet dropping), and the newcomer is added to the queue after shifting the remaining data by one register. You may use the array structure of Verilog HDL to implement buffers. For example, if the bit-sequence "0110" is sent after pressing the start button, your code should make the next white box of "Buffer 2" blue and print "2" inside that box. The second part of the project is to read the data from suitable buffers. For this purpose, you should implement an algorithm that decides the buffer from which data is read, according to the given criteria. For a given buffer, the oldest data is read first, in a first-in-first-out manner.

The criteria can be described as follows,

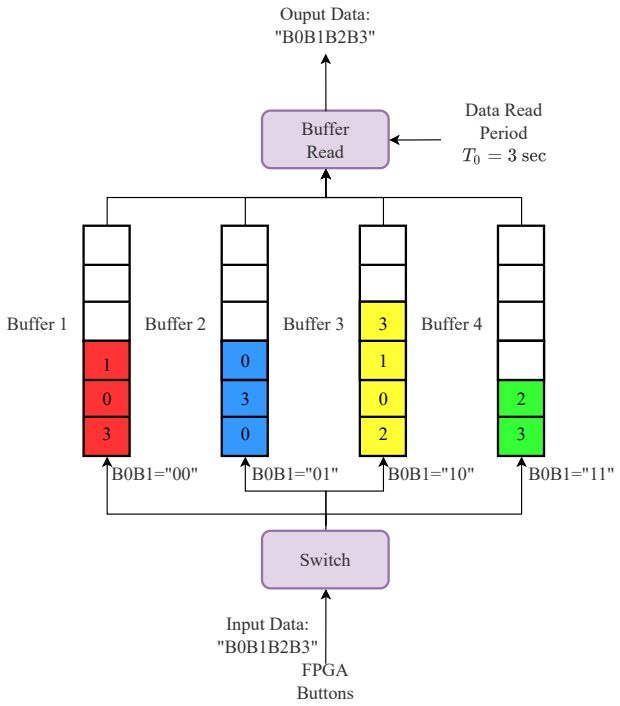
2.1.1 Latency Requirement

Latency precedence for the buffers is given as Buffer 1 > Buffer 2 > Buffer 3 > Buffer 4. This means that the lowest delay is desired for Buffer 1. For example, if some data are available at Buffer 1 and Buffer 2, Buffer 1 should be read first.

2.1.2 Reliability Requirement

Reliability precedence for the buffers is Buffer 4 > Buffer 3 > Buffer 2 > Buffer 1. This means that the lowest packet drop rate is desired for Buffer 4. But you should note that you should minimize the overall packet drops to achieve desired performance.

Also, once data is read from a buffer you should delete that data, and shift the remaining data by one register.



Number of transmitted packets transferred as Output Data

Buffer 1	Buffer 2	Buffer 3	Buffer 4	Total
40	20	15	50	125

Number of recieved packets from FPGA buttons

Buffer 1	Buffer 2	Buffer 3	Buffer 4	Total
50	25	15	50	140

Number of dropped packets

Buffer 1	Buffer 2	Buffer 3	Buffer 4	Total
10	5	0	0	15

Figure 1: Example Buffering System.

2.2 Project Requirements

To define the requirements one by one as a list:

- You are going to design the main screen using the VGA interface, at 640× 480 resolution. On your main screen, which may resemble Figure 1,
  1. You should present the buffers and the available data values inside the buffers.
  2. The number of transmitted packets, the number of received packets, and the number of dropped packets should be printed.
- When new data arrives via FPGA buttons, buffer contents and the corresponding numbers on the right-hand side should change.
- At every 3 seconds, a single register from one of the buffers is read and should be printed as output. Also, necessary deletion operations and number updates should be performed.

3 Final Words

This is not a last-night project. You must put an effort on it to be successful. On the other hand, do not worry. This is a straightforward project in general.

Please keep in mind that we have limited resources and we must benefit wisely so that new students can also benefit. FPGA boards are sensitive, fragile, and expensive boards. Now, they are in perfect condition. For the duration you borrowed them from us, you will be responsible for taking care of them. Once the time to give FPGA boards back is reached, you must give your borrowed FPGA back in perfect condition as well. Failing to give back to the FPGA board will result in a harsh penalty.

As you can notice also, some details regarding the project are left to you. For some parts, there are nearly infinite options. We would like to remind you that cheating is a serious issue, which will be strictly penalized. We hope that all of you will do creative and successful project demonstrations.

Enjoy.

4 Deliverables

To complete the project work, you need to submit a video presentation and a final report, and attend the demonstrations. Deadlines are as follows:

- **Video presentation** 12.06.2022
- **Project demonstrations** 04-06.07.2022
- **Final report** 07.07.2022 **at 10.00 AM**

4.1 Design proposal video presentation

Each group needs to prepare a design proposal presentation of their preliminary research about the term project. At this stage your solution does not need to be fully functional. However, you need to offer a solution to every requirement of the project.

Requirements for the video can be summarized as:

- A presentation needs to be prepared using PowerPoint or equivalent applications.
- Group members need to make the presentation over zoom and record it as a video.
- In the video, both presentation and presenter need to be visible.
- All group members need to contribute to the video equally.
- Video duration needs to be at most 5 minutes.
- Presentation must include the problem definition and solution proposal, including block and state diagrams.

4.2 Final report

You are going to submit a final report. In the report, you are going to explain your conceptual design or solutions to the requirements for the project. State diagrams defining your logic scheme, theoretical understanding of VGA and buttons and pictures of the working system should be included as well.

4.3 Project demonstrations

Demonstrations will be held in the laboratory after final exams. Each group member needs to be able to explain the design, present the system on the FPGA board and answer the assistants' questions.