Laboratory Exercise 12

Implementation of a FIFO memory

This is an exercise in implementation of a FIFO memory using LEDs, DIP Switches & Push Buttons.

Background

FIFO is an acronym for First In First Out, which describes how data is managed relative to time or priority. In this case, the first data that arrives will also be the first data to leave from a group of data. A FIFO Buffer is a read/write memory array that automatically keep track of the order in which data enters into the module and reads the data out in the same order. In hardware FIFO buffer is used for synchronization purposes. It is often implemented as a circular queue, and has two pointers:

- Read Pointer/Read Address Register
- Write Pointer/Write Address Register

Read and write addresses are initially both at the first memory location and the FIFO queue is Empty. When the difference between the read address and write address of the FIFO buffer is equal to the size of the memory array then the FIFO queue is Full.
FIFO can be classified as synchronous or asynchronous depending on whether same clock (synchronous) or different clocks (asynchronous) control the read and write operations.

Synchronous FIFO

A synchronous FIFO refers to a FIFO design where data values are written sequentially into a memory array using a clock signal, and the data values are read out sequentially from the memory array using the same clock signal. Figure 1 shows the flow of the operation of a typical FIFO.

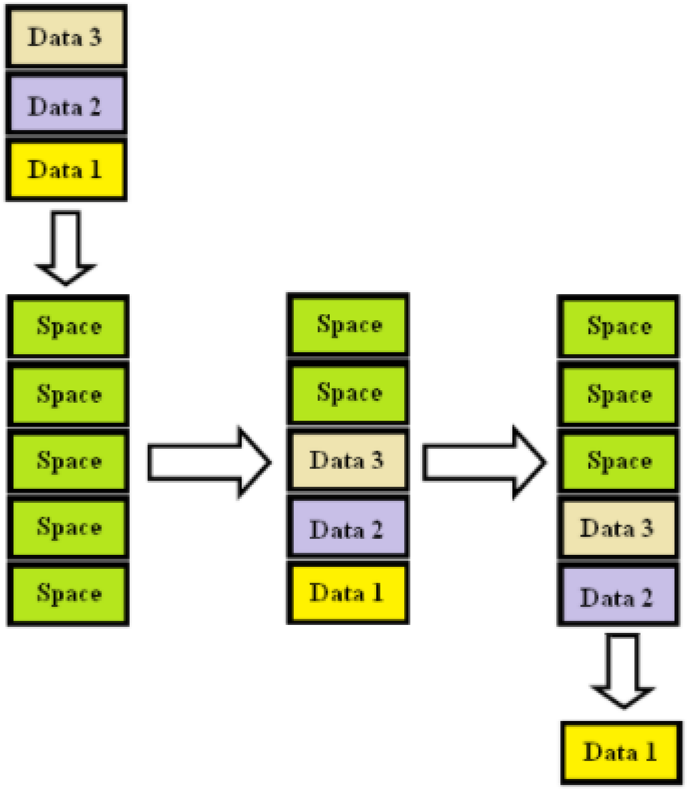

Figure 1: Operation of FIFO with the three data values.

Part I

Implement 8x8 FIFO memory as shown in Figure 2. This FIFO Buffer can store eight 8-bit values. The FIFO Buffer module consists of a 8-bit data input line, dataIn and a 8-bit data output line, dataOut. The module is clocked using the 1-bit input clock line Clk. The module also has a 1-bit enable line, EN and a 1-bit active high reset line, Rst.

The 1-bit RD line is used to signal a data read operation on the FIFO Buffer and the 1-bit WR line is used to signal a data write operation on the FIFO Buffer. Both the RD and WR lines are active high. The module also has two output lines FULL and EMPTY which are each 1-bit wide. The FULL line becomes high when the FIFO Buffer is or becomes full (internal counter becomes eight). The EMPTY line becomes high when the FIFO Buffer is or becomes empty (internal counter becomes zero).

You can also add further design details if you want (Since hardware implementation is not possible, so you will write Verilog code, and testbench (stimulus). Test you code using testbench waveform. Paste your Verilog code, testbench, and waveform screenshot in lab report).

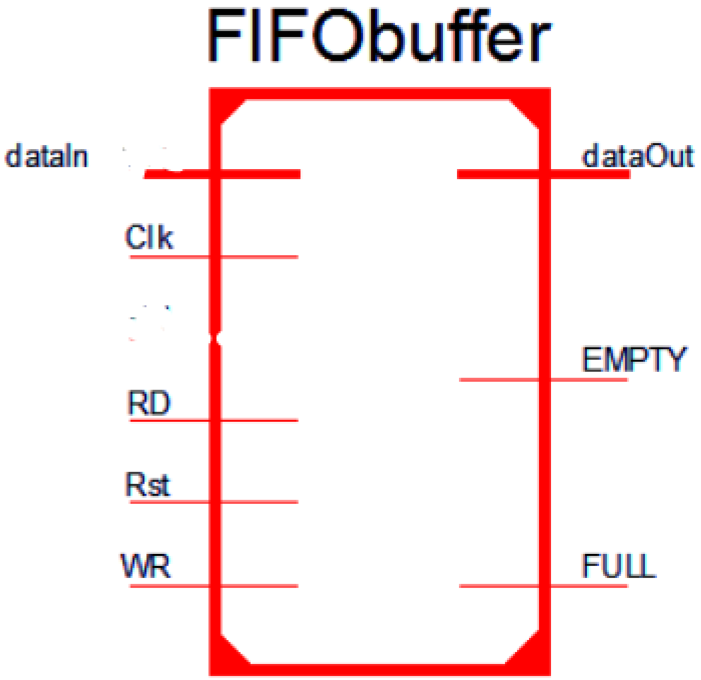

Figure 2: First in first out buffer.

Part II

Implement asynchronous FIFO with the same specifications as mentioned above for synchronous FIFO.
(Hint: Asynchronous FIFO is a FIFO design in which data is written from one clock domain to the FIFO buffer and read from the same FIFO buffer in another clock domain, which is asynchronous to each other.)