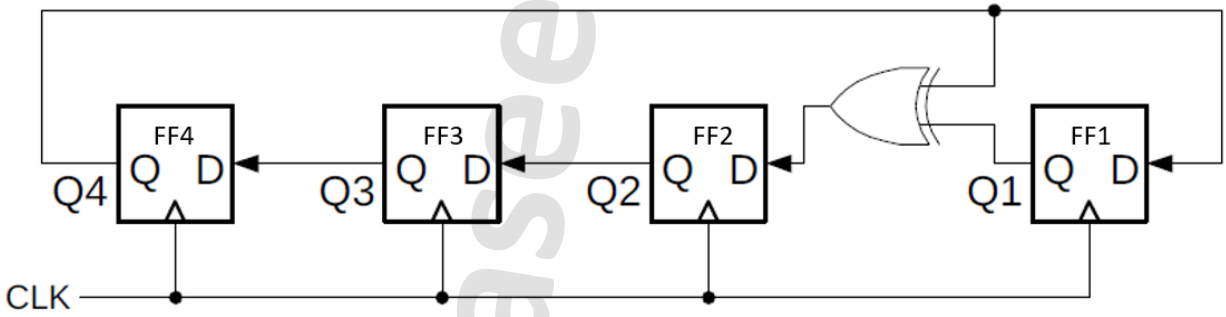
Assignment 4 Extra

Design the following circuits using Verilog **and create a testbench** for each design to check its functionality. **Create a do file for question 3.**

1) Implement the following Linear feedback shift register (LFSR)

- LFSR Inputs: clk, rst, set
- LFSR output: out (4 bits) where out[3] is connected to Q4, out[2] is connected to Q3, etc.



- LFSR can be used as a random number generator.
- The sequence is a random sequence where numbers appear in a random sequence and repeats as shown on the figure on the right

0001
0010
0100
1000
0011
0110
1100
1011
0101
1010
0111
1110
1111
1101
1001
0001

FF2, FF3, FF4 have the following specifications:

- D input
- Clk input
- Input async rst (active high) – resets output to 0
- Output Q

FF1 have the following specifications:

- D input
- Clk input
- Input async set (active high) – set output to 1
- Output Q

Note: the rst and set signals should be activated at the same time to guarantee correct operation

Testbench should self-check the control signals functionality. For normal operation, check the waveform for correct random sequence generated from the LFSR.

2) Implement N-bit parameterized Full/Half adder

- Parameters
  - WIDTH: Determine the width of input a,b, sum, default is 4
  - PIPELINE_ENABLE: if this parameter is high then the output of the sum and carry will be available in the positive clock edge (sequential) otherwise the circuit is pure combinational, default is 1. Valid values: 0 or 1.
  - USE_FULL_ADDER: if this parameter is high then cin signal will be used during the cout and sum calculation from the input signals, otherwise if this parameter is low ignore the cin input, default is 1. Valid values: 0 or 1.
- Ports

Name	Type	Description
a	Input	Data input a of width determined by WIDTH parameter
b		Data input b of width determined by WIDTH parameter
clk		Clk input
cin		Carry in bit
rst		Active high synchronous reset
sum	Output	sum of a and b input of width determined by WIDTH parameter
cout		Carry out bit

Create self-checking testbenches to fully verify the functionality of the design. WIDTH parameter is left with the default value while you have to verify the design in different parameter configurations of PIPELINE_ENABLE and USE_FULL_ADDER. **Hint:** 4 testbenches are required to fully verify the design

3) Implement shift register with the following specs:

Parameter:

1. SHIFT_DIRECTION: specify shifting direction either LEFT or RIGHT, default = “LEFT”
2. SHIFT_AMOUNT: specify the number of bits to be shifted, possible values are 1, 2, 3, 4, 5, 6, 7. Default = 1

Ports:

1. Inputs:
   - clk
   - rst (async active high)
   - load: control signal if high, register should be loaded with the input “load_value”
   - load_value: value to be loaded to the register
2. Outputs:
   - PO (8 bits): parallel out which represent the register to be shifted.

Create 2 testbench to test the operation of the register when shifting right and shift amount is 2 and the other testbench to test the shifting left and shift amount is 1. The following specs should be tested in each testbench:

1. Test reset that it forces the output to zero
2. Load signal to load a randomized value to the output
3. Test the shifting operation on the output
4. Load another randomized value to the output
5. Test the shifting again

Use Questasim waveform to check the above functionality

Deliverables:

1) The assignment should be submitted as a PDF file with this format <your_name>_Assignment4_Extra for example Kareem_Waseem_Assignment4_Extra
2) Snippets from the waveforms captured from QuestaSim for each design with inputs assigned values and output values visible.

Note that your document should be organized as 4 sections corresponding to each design above, and in each section, I am expecting the Verilog code, and the waveforms snippets