1-point Homework 1

Use the last 3 digits of your personal number (PN) to select which question you have to answer. Modulo divide your 3 digits with the number of questions, in this case, 3, plus 1. The resulting number is the question that you should answer.

!!! (PN mod 3) + 1 !!!

For example, if your PN ends in 730, then (730 mod 3) + 1 = 2, meaning you must answer question 2.

Make sure you comment on the top of each file you submit, your PN, name and the question you are answering.

!!! Important !!!

If you answer the wrong question, your submission will be invalid.

We can call you to explain your solution. If you cannot explain your answer, the homework will be invalid!

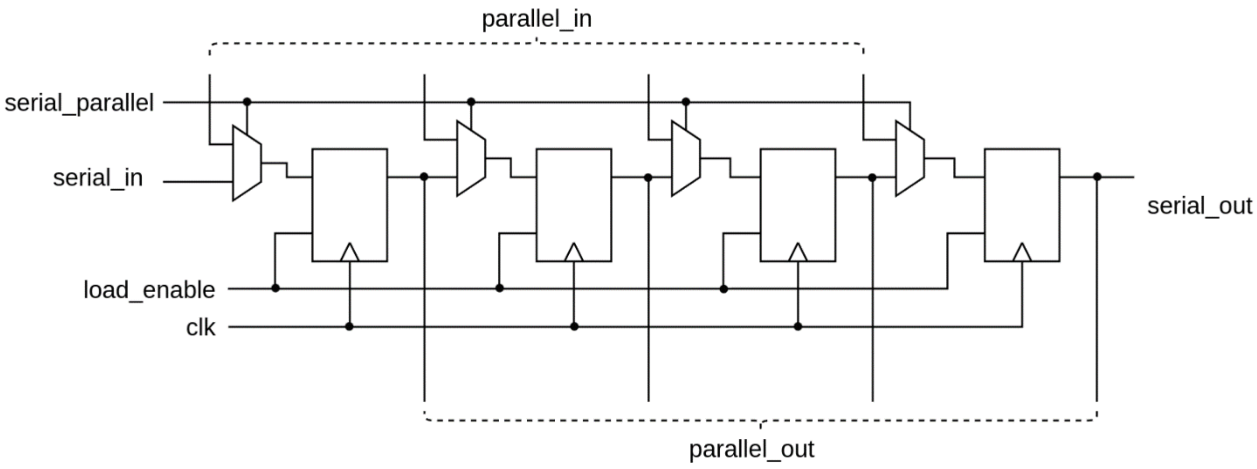
KTH has a zero-tolerance policy against cheating.

https://www.kth.se/en/student/stod/studier/fusk-1.997287

If you have questions or need clarifications, you can ask in the discussion forum in Canvas.

Questions

1.1 Question 1



Model using HDL, a parametric shift register that can accept data inputs both parallel and serially. Use an enable signal (load enable) to stop the flops from accepting new data. The enable signal should be 0 when you output in parallel in order to keep the output constant.

Module parameters

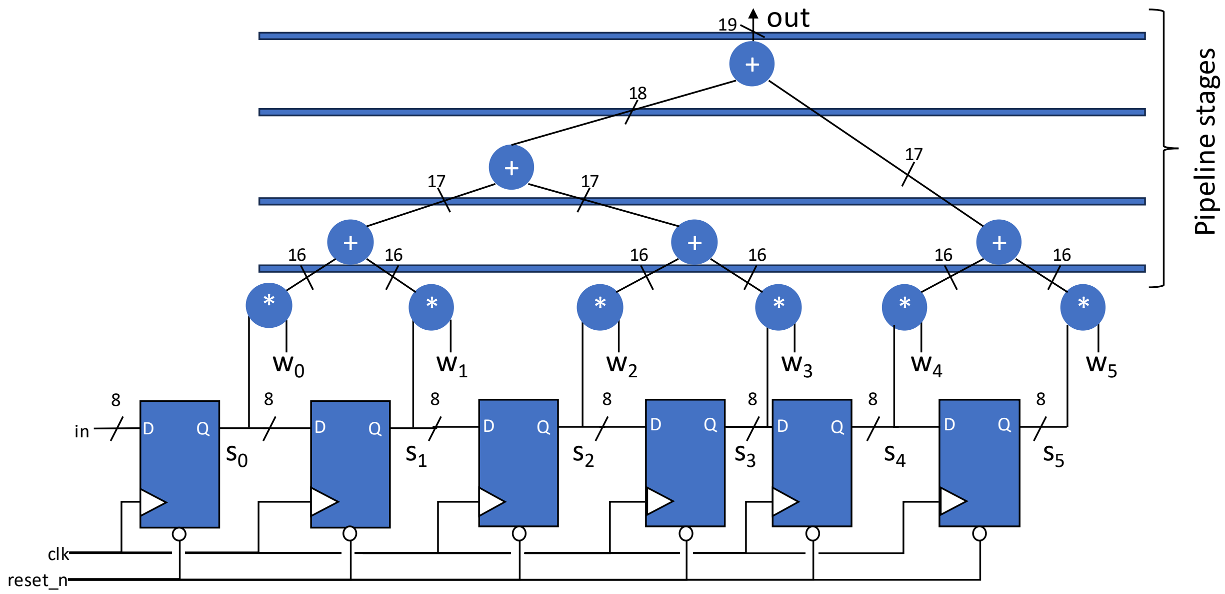
Name	Width	Description
N	1	Length of the LFSR

Module pinout

Name	Direction	Width	Description
clk	in	1	Clock signal
rstn	in	1	Active low reset
serial_parallel	in	1	If 0, load serially; if 1 load parallel
load_enable	in	1	If 1, flops accept new data
parallel_in	in	N	Parallel input to the shift register
serial_in	in	1	Serial input to the shift register
parallel_out	out	N	Parallel output of the shift register
serial_out	out	1	Serial output of the shift register

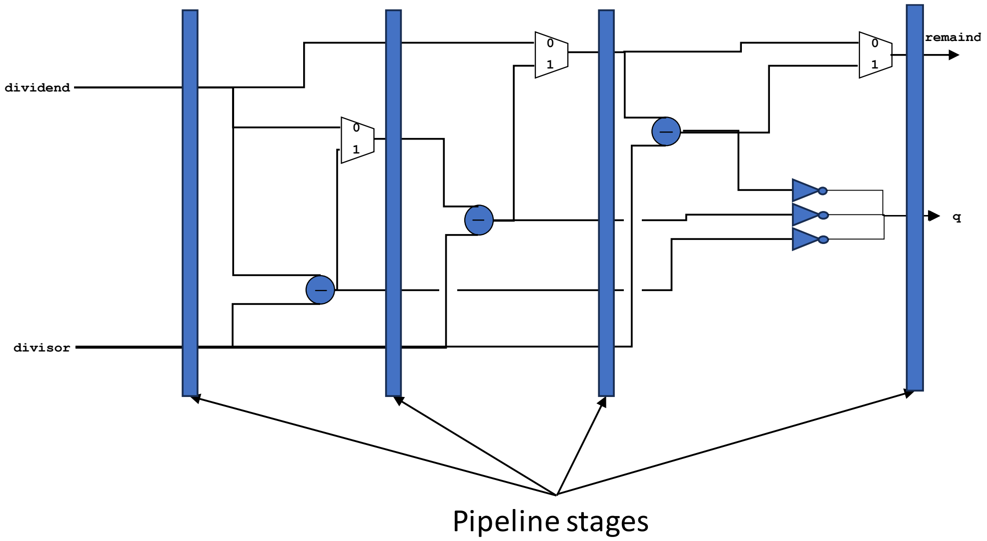
1.2 QUESTION 2

Model using HDL, a design that implements an FIR filter, shown below, with six taps. The input to the design should be 8 bits, and the output should be 19 bits. The design should be pipelined with five pipeline stages, as seen in the schematic below.



1.3 Question 3

Pipeline the divider logic given with five pipeline stages. The pipeline stages should be inserted as seen in the abstract schematic below.



```
module divider (
  input  logic [2:0] dividend,
  input  logic [1:0] divisor,
  output logic [2:0] q,
  output logic [1:0] remainder
);
  logic [2:0] sub[2:0];
  logic [2:0] sel[1:0];
  assign sub[2]    = {2'b00,dividend[2]}-{1'b0,divisor};
  assign sel[1]    = (sub[2][2])?{1'b0,dividend[2:1]}:{sub[2][1:0],dividend[1]};
  assign sub[1]    = sel[1]-{1'b0,divisor};
  assign sel[0]    = (sub[1][2])?{sel[1][1:0],dividend[0]}:{sub[1][1:0],dividend[0]};
  assign sub[0]    = sel[0]-{1'b0,divisor};
  assign remainder = (sub[0][2])?sel[0][1:0]:sub[0][1:0];
  assign q[2]      = ~sub[2][2];
  assign q[1]      = ~sub[1][2];
  assign q[0]      = ~sub[0][2];
endmodule
```