FH | JOANNEUM
System Test Engineering

STE – Digital Electronics
Lecture 8 – Lab ~~4~~ 5

DI Klaus Strohmayer – Digital Design and Verification Expert

Dice Part 1

Implement the RTL code for a dice.

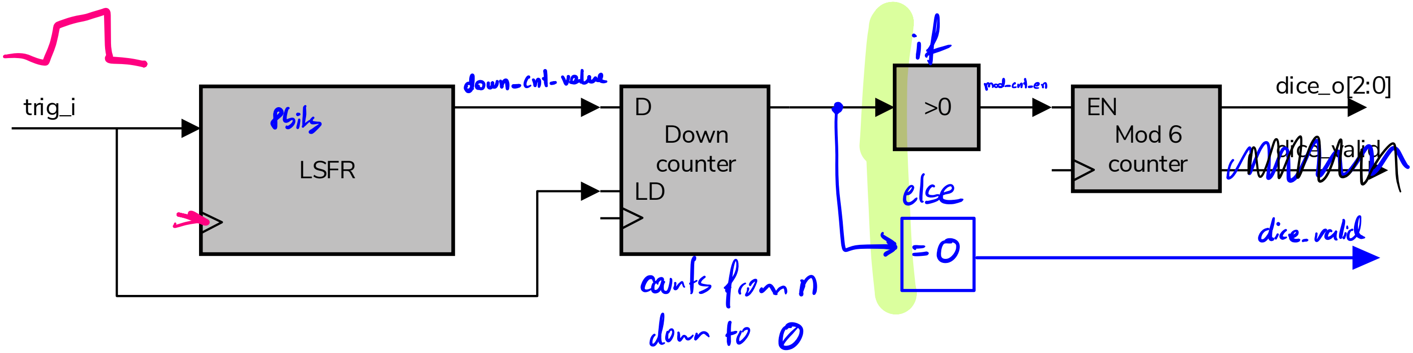
- Design
  - Trigger via input trig_i
  - LSFR based random number generation
    - Bit width 8bit – polyoma $x^8 + x^6 + x^5 + x^4 + 1$
  - Modulo 6 division (3bit) (dice_dout_o[2:0])
    - dice_done_o is one once calculation is done
    - dice_dout_o[2:0] is zero while calculating

Verification

- Trigger 400 times
- Collect number of occurrences for 1,2,..6
- Check if values 0 and 7 don't appear on the output (immediate assertion)

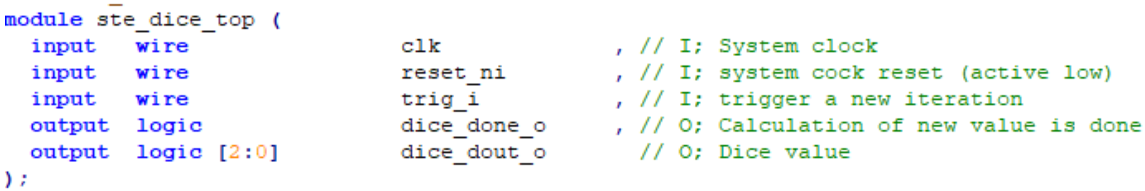
Consider split (RTL, TB)

Dice Part 1

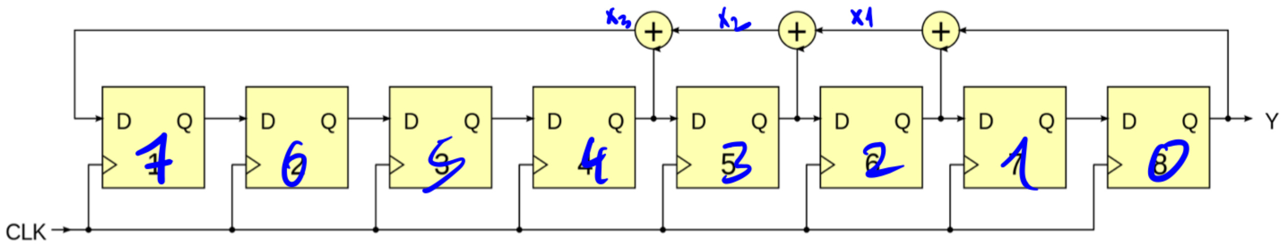


Block diagram

- All FFs must have an asynchronous reset

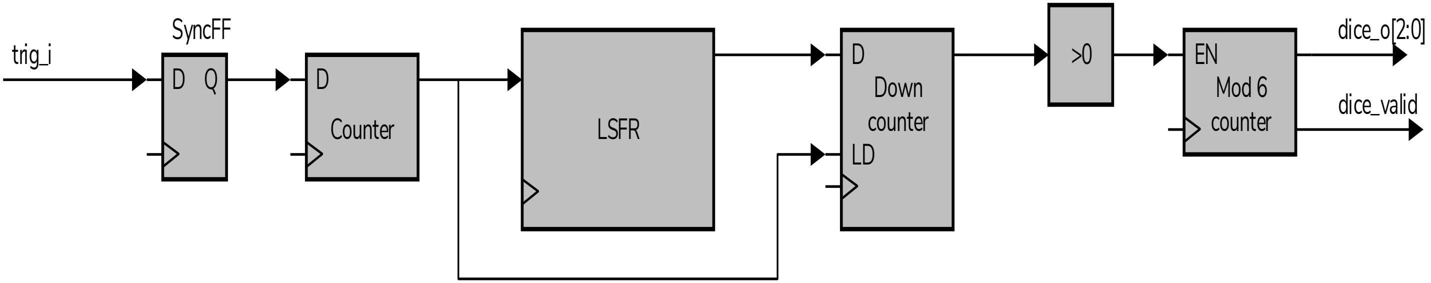


```
module ste_dice_top (
  input   wire            clk          , // I; System clock
  input   wire            reset_ni     , // I; system cock reset (active low)
  input   wire            trig_i       , // I; trigger a new iteration
  output  logic           dice_done_o  , // O; Calculation of new value is done
  output  logic [2:0]     dice_dout_o    // O; Dice value
);
```



$$x^8 + x^6 + x^5 + x^4 + 1$$

Dice Part 1a



Block diagram

- Integrate the given debouncer ste_debounce.sv

Dice Part 1 Deliverables

- Provide the RTL code
- Provide Testbench code
- Provide a Waveform screenshot
  - One single dice value generation
  - Select 4 to 6 relevant signals