Homework 2

RTL Design of a Randomizer

1 Purpose

- This HW is a pre-requisite for Lab2 which will be covered in another document
- The objectives of this homework are:
  - RTL implementation of a randomizer
  - Validation of the design using self-checking testbenches and simulation

2 Required tools

Notepad++ (source code editing)
ModelSim PE student editions (simulation)

3 Part A: PRBS Generator

- Using a linear feedback shift register (LFSR), design a pseudo random binary sequence (PRBS) generator that implements the polynomial $1 + X^{14} + X^{15}$ shown in Figure 1 which is used to randomize a sequence of binary inputs.
- The randomizer is initialized with the vector:
  - [LSB] 0 1 1 0 1 1 1 0 0 0 1 0 1 0 1 [MSB].
- In addition to clock (`clk`) and asynchronous reset (`reset`), include a synchronous seed load (`load`) and clock enable (`en`) inputs
- Validation: using the following input data sequence and the corresponding data output to validate your design
  - Input Data (Hex):
    - AC BC D2 11 4D AE 15 77 C6 DB F4 C9
  - Randomized Data (Hex):
    - 55 8A C4 A5 3A 17 24 E1 63 AC 2B F9

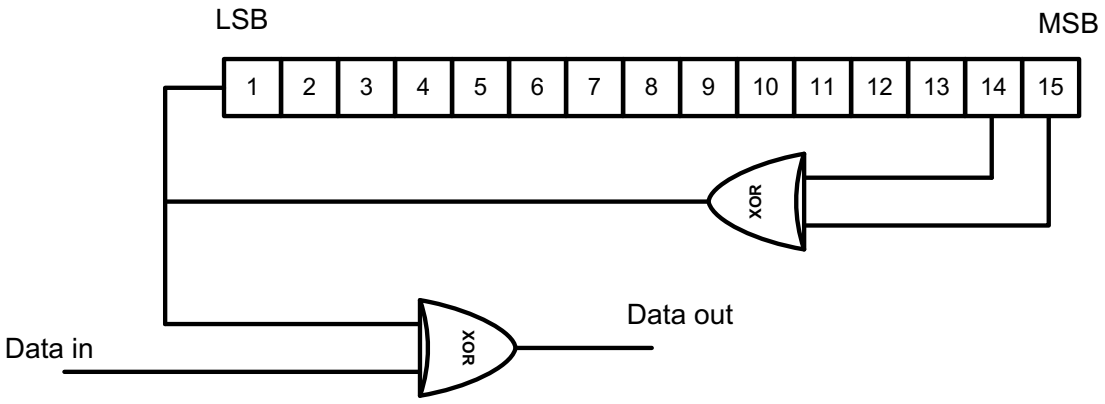

Figure 1: PRBS generator for Data randomization

Table 1: Intermediate data values for validation

Initialized vector 011 0111 0001 0101

No	Shift Register															XOR #1	Data in		Data out	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15		0x	0b	0b	0x
1	**0**	**1**	**1**	**0**	**1**	**1**	**1**	**0**	**0**	**0**	**1**	**0**	**1**	**0**	**1**	1		**1**	**0**	
2	1	0	1	1	0	1	1	1	0	0	0	1	0	1	0	1	**A**	**0**	**1**	**5**
3	1	1	0	1	1	0	1	1	1	0	0	0	1	0	1	1		**1**	**0**	
4	1	1	1	0	1	1	0	1	1	1	0	0	0	1	0	1		**0**	**1**	
5	1	1	1	1	0	1	1	0	1	1	1	0	0	0	1	1		**1**	**0**	
6	1	1	1	1	1	0	1	1	0	1	1	1	0	0	0	0	**C**	**1**	**1**	**5**
7	0	1	1	1	1	1	0	1	1	0	1	1	1	0	0	0		**0**	**0**	
8	0	0	1	1	1	1	1	0	1	1	0	1	1	1	0	1		**0**	**1**	
9	1	0	0	1	1	1	1	1	0	1	1	0	1	1	1	0		**1**	**1**	
10	0	1	0	0	1	1	1	1	1	0	1	1	0	1	1	0	**B**	**0**	**0**	**8**
11	0	0	1	0	0	1	1	1	1	1	0	1	1	0	1	1		**1**	**0**	
12	1	0	0	1	0	0	1	1	1	1	1	0	1	1	0	1		**1**	**0**	
13	1	1	0	0	1	0	0	1	1	1	1	1	0	1	1	0		**1**	**1**	
14	0	1	1	0	0	1	0	0	1	1	1	1	1	0	1	1	**C**	**1**	**0**	**A**
15	1	0	1	1	0	0	1	0	0	1	1	1	1	1	0	1		**0**	**1**	
16	1	1	0	1	1	0	0	1	0	0	1	1	1	1	1	0		**0**	**0**	
17	0	1	1	0	1	1	0	0	1	0	0	1	1	1	1	0		**1**	**1**	
18	0	0	1	1	0	1	1	0	0	1	0	0	1	1	1	0	**D**	**1**	**1**	**C**
19	0	0	0	1	1	0	1	1	0	0	1	0	0	1	1	0		**0**	**0**	
20	0	0	0	0	1	1	0	1	1	0	0	1	0	0	1	1		**1**	**0**	
21	1	0	0	0	0	1	1	0	1	1	0	0	1	0	0	0		**0**	**0**	
22	0	1	0	0	0	0	1	1	0	1	1	0	0	1	0	1	**2**	**0**	**1**	**4**
23	1	0	1	0	0	0	0	1	1	0	1	1	0	0	1	1		**1**	**0**	
24	1	1	0	1	0	0	0	0	1	1	0	1	1	0	0	0		**0**	**0**	
25	0	1	1	0	1	0	0	0	0	1	1	0	1	1	0	1		**0**	**1**	
26	1	0	1	1	0	1	0	0	0	0	1	1	0	1	1	0	**1**	**0**	**0**	**A**
27	0	1	0	1	1	0	1	0	0	0	0	1	1	0	1	1		**0**	**1**	
28	1	0	1	0	1	1	0	1	0	0	0	0	1	1	0	1		**1**	**0**	
29	1	1	0	1	0	1	1	0	1	0	0	0	0	1	1	0		**0**	**0**	
30	0	1	1	0	1	0	1	1	0	1	0	0	0	0	1	1	**1**	**0**	**1**	**5**
31	1	0	1	1	0	1	0	1	1	0	1	0	0	0	0	0		**0**	**0**	
32	0	1	0	1	1	0	1	0	1	1	0	1	0	0	0	0		**1**	**1**	
33	0	0	1	0	1	1	0	1	0	1	1	0	1	0	0	0		**0**	**0**	
34	0	0	0	1	0	1	1	0	1	0	1	1	0	1	0	1	**4**	**1**	**0**	**3**
35	1	0	0	0	1	0	1	1	0	1	0	1	1	0	1	1		**0**	**1**	
36	1	1	0	0	0	1	0	1	1	0	1	0	1	1	0	1		**0**	**1**	
37	1	1	1	0	0	0	1	0	1	1	0	1	0	1	1	0		**1**	**1**	
38	0	1	1	1	0	0	0	1	0	1	1	0	1	0	1	1	**D**	**1**	**0**	**A**
39	1	0	1	1	1	0	0	0	1	0	1	1	0	1	0	1		**0**	**1**	
40	1	1	0	1	1	1	0	0	0	1	0	1	1	0	1	1		**1**	**0**	
41	1	1	1	0	1	1	1	0	0	0	1	0	1	1	0	1	**A**	**1**	**0**	**1**

42	1	1	1	1	0	1	1	1	0	0	0	1	0	1	1	0		**0**	**0**	
43	0	1	1	1	1	0	1	1	1	0	0	0	1	0	1	1		**1**	**0**	
44	1	0	1	1	1	1	0	1	1	1	0	0	0	1	0	1		**0**	**1**	
45	1	1	0	1	1	1	1	0	1	1	1	0	0	0	1	1	**E**	**1**	**0**	**7**
46	1	1	1	0	1	1	1	1	0	1	1	1	0	0	0	0		**1**	**1**	
47	0	1	1	1	0	1	1	1	1	0	1	1	1	0	0	0		**1**	**1**	
48	0	0	1	1	1	0	1	1	1	1	0	1	1	1	0	1		**0**	**1**	
49	1	0	0	1	1	1	0	1	1	1	1	0	1	1	1	0	**1**	**0**	**0**	**2**
50	0	1	0	0	1	1	1	0	1	1	1	1	0	1	1	0		**0**	**0**	
51	0	0	1	0	0	1	1	1	0	1	1	1	1	0	1	1		**0**	**1**	
52	1	0	0	1	0	0	1	1	1	0	1	1	1	1	0	1		**1**	**0**	
53	1	1	0	0	1	0	0	1	1	1	0	1	1	1	1	0	**5**	**0**	**0**	**4**
54	0	1	1	0	0	1	0	0	1	1	1	0	1	1	1	0		**1**	**1**	
55	0	0	1	1	0	0	1	0	0	1	1	1	0	1	1	0		**0**	**0**	
56	0	0	0	1	1	0	0	1	0	0	1	1	1	0	1	1		**1**	**0**	
57	1	0	0	0	1	1	0	0	1	0	0	1	1	1	0	1	**7**	**0**	**1**	**E**
58	1	1	0	0	0	1	1	0	0	1	0	0	1	1	1	0		**1**	**1**	
59	0	1	1	0	0	0	1	1	0	0	1	0	0	1	1	0		**1**	**1**	
60	0	0	1	1	0	0	0	1	1	0	0	1	0	0	1	1		**1**	**0**	
61	1	0	0	1	1	0	0	0	1	1	0	0	1	0	0	0	**7**	**0**	**0**	**1**
62	0	1	0	0	1	1	0	0	0	1	1	0	0	1	0	1		**1**	**0**	
63	1	0	1	0	0	1	1	0	0	0	1	1	0	0	1	1		**1**	**0**	
64	1	1	0	1	0	0	1	1	0	0	0	1	1	0	0	0		**1**	**1**	
65	0	1	1	0	1	0	0	1	1	0	0	0	1	1	0	1	**C**	**1**	**0**	**6**
66	1	0	1	1	0	1	0	0	1	1	0	0	0	1	1	0		**1**	**1**	
67	0	1	0	1	1	0	1	0	0	1	1	0	0	0	1	1		**0**	**1**	
68	1	0	1	0	1	1	0	1	0	0	1	1	0	0	0	0		**0**	**0**	
69	0	1	0	1	0	1	1	0	1	0	0	1	1	0	0	0	**6**	**0**	**0**	**3**
70	0	0	1	0	1	0	1	1	0	1	0	0	1	1	0	1		**1**	**0**	
71	1	0	0	1	0	1	0	1	1	0	1	0	0	1	1	0		**1**	**1**	
72	0	1	0	0	1	0	1	0	1	1	0	1	0	0	1	1		**0**	**1**	
73	1	0	1	0	0	1	0	1	0	1	1	0	1	0	0	0	**D**	**1**	**1**	**A**
74	0	1	0	1	0	0	1	0	1	0	1	1	0	1	0	1		**1**	**0**	
75	1	0	1	0	1	0	0	1	0	1	0	1	1	0	1	1		**0**	**1**	
76	1	1	0	1	0	1	0	0	1	0	1	0	1	1	0	1		**1**	**0**	
77	1	1	1	0	1	0	1	0	0	1	0	1	0	1	1	0	**B**	**1**	**1**	**C**
78	0	1	1	1	0	1	0	1	0	0	1	0	1	0	1	1		**0**	**1**	
79	1	0	1	1	1	0	1	0	1	0	0	1	0	1	0	1		**1**	**0**	
80	1	1	0	1	1	1	0	1	0	1	0	0	1	0	1	1		**1**	**0**	
81	1	1	1	0	1	1	1	0	1	0	1	0	0	1	0	1	**F**	**1**	**0**	**2**
82	1	1	1	1	0	1	1	1	0	1	0	1	0	0	1	1		**1**	**0**	
83	1	1	1	1	1	0	1	1	1	0	1	0	1	0	0	0		**1**	**1**	
84	0	1	1	1	1	1	0	1	1	1	0	1	0	1	0	1		**1**	**0**	
85	1	0	1	1	1	1	1	0	1	1	1	0	1	0	1	1	**4**	**0**	**1**	**B**
86	1	1	0	1	1	1	1	1	0	1	1	1	0	1	0	1		**1**	**0**	
87	1	1	1	0	1	1	1	1	1	0	1	1	1	0	1	1		**0**	**1**	
88	1	1	1	1	0	1	1	1	1	1	0	1	1	1	0	1		**0**	**1**	
89	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	0	**C**	**1**	**1**	**F**

90	0	1	1	1	1	1	0	1	1	1	1	1	0	1	1	0		**1**	**1**	
91	0	0	1	1	1	1	1	0	1	1	1	1	1	0	1	1		**0**	**1**	
92	1	0	0	1	1	1	1	1	0	1	1	1	1	1	0	1		**0**	**1**	
93	1	1	0	0	1	1	1	1	1	0	1	1	1	1	1	0	**9**	**1**	**1**	**9**
94	0	1	1	0	0	1	1	1	1	1	0	1	1	1	1	0		**0**	**0**	
95	0	0	1	1	0	0	1	1	1	1	1	0	1	1	1	0		**0**	**0**	
96	0	0	0	1	1	0	0	1	1	1	1	1	0	1	1	0		**1**	**1**	

3.1 Submission

Name the RTL module as prbs.vhd. Create at testbench the verify the functionality of prbs.vhd using self-checking testbench. Name this testbench prbs_tb.vhd

4 Part B: PRBS verify wrapper

The PRBS verify wrapper (`prbs_verify`) is shown in Figure 2. It encloses the prbs module, a ROM (constant register) that holds the value of the seed (`seed_rom`), a second ROM (`in_data_rom`) that holds the test value of the input data sequence (`data_in`), a third ROM (`out_data_rom`) that holds the expected value of the output data sequence (`data_out`), and a block that contains a verifying logic to check the output data sequence against the values of the data stored in the output data ROM. If there is no mismatch, then the `pass` output is asserted. The `prbs_verify` module takes an input clock (`clk`), an active high `reset`, a seed load input (`load`), and an enable input (`en`). Those inputs are passed through to the `prbs` block. The input clock is connected to the 50 MHz clock on DE0-CV board. The reset input is connected to a push button switch. Both `load` and `en` are connecting to a sliding switch. The `pass` output is connected to a LED.

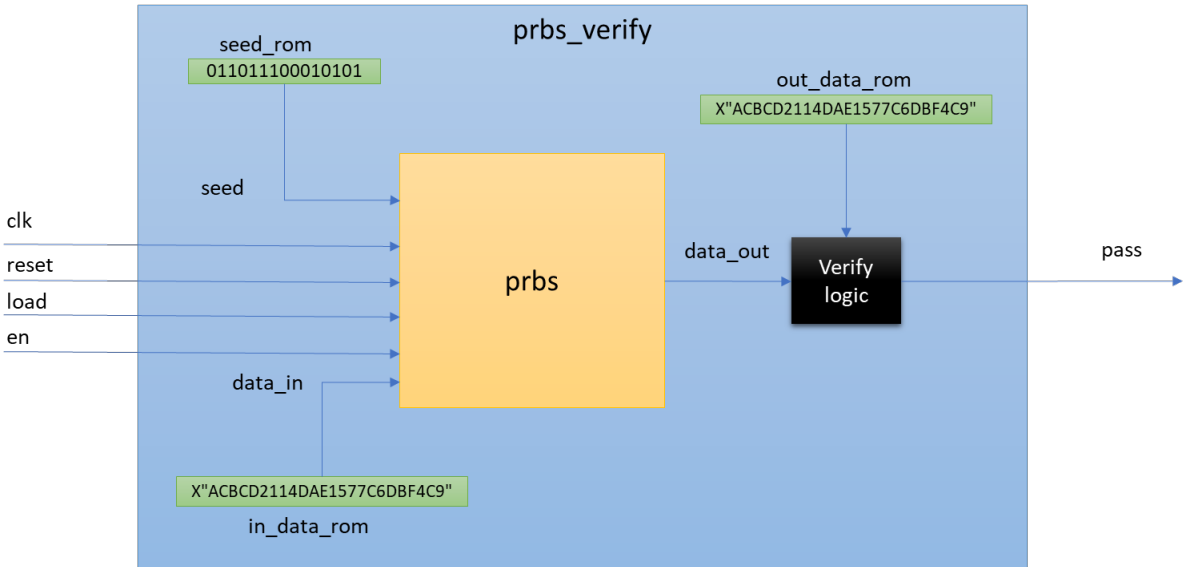

Figure 2: PRBS verify wrapper

4.1 Submission

a. Design an RTL implementation of the PRBS verifier shown Figure 2. Name the file prbs_verify.vhd
b. Design a testbench to verify the functionality of prbs_verify. Name this testbench prbs_verify_tb.vhd.
c. Create a simulation project and simulate the three files: prbs.vhd, prbs_verify.vhd, and prbs_verify_tb.vhd