PRNG USING LFSR WITH RESEEDING AND DYNAMIC TAPPING TECHNIQUES.

Sunday, July 27, 2025 1:48 PM

Important Terminology:

PRNG	Psuedo Random Number Generator
LFSR	Linear Feedback Shift Register
Tapping	The Mechanism of Feeding the output back into the shift register as input by performing some operations
Seeding	The process of giving the initial input to the register to perform shifting by feedback

Method of Implementation:

In this Method There will be 2 LFSR's,
-> One 64 bit LFSR(Main LFSR For Random Number Generation)
-> One 4 bit LFSR(Used for Seeding and Tapping the 64Bit LFSR Dynamically)

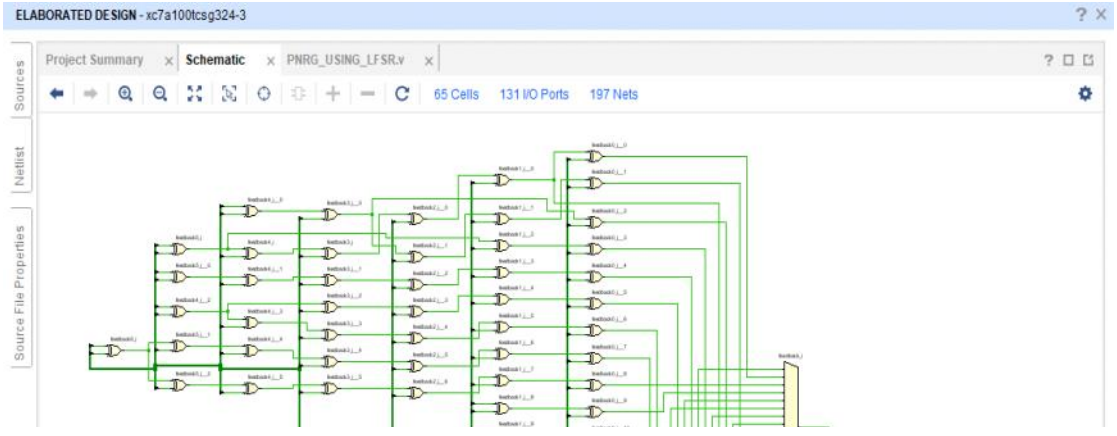
There will be a status switch called seed for controlling the operation of 64Bit LFSR. If (Seed==0) Shifitng of operation in register continues. If(seed==1) shift register get new seed value from 4 bit LFSR.

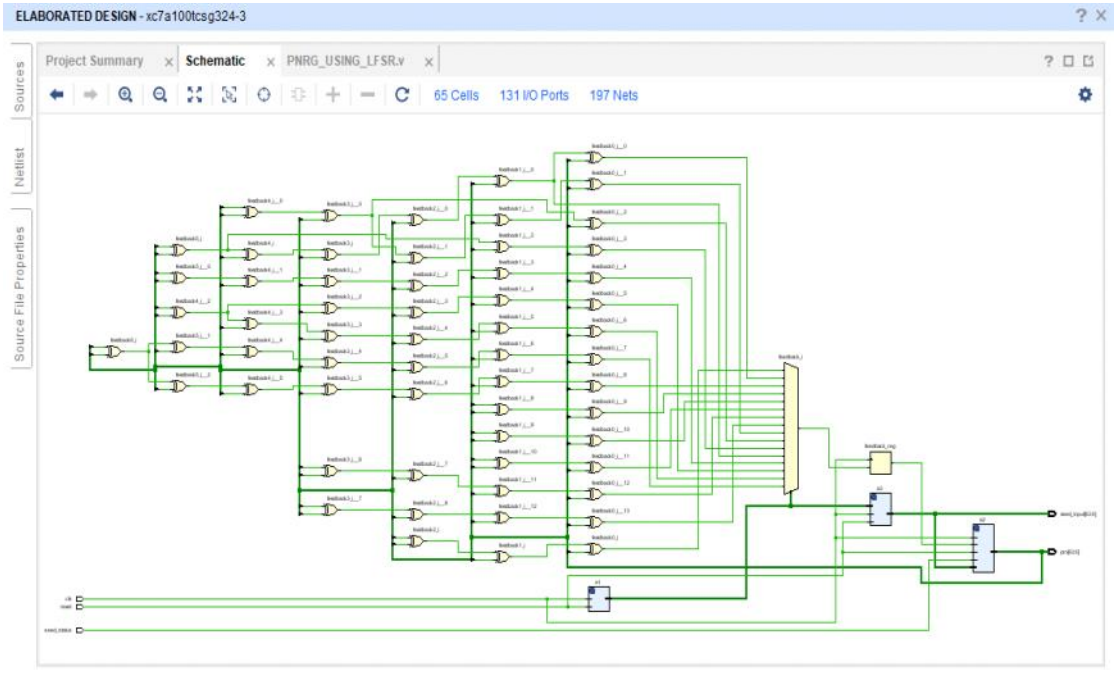
Based on the output of 4bit LFSR, The tapping mechanism decides. It assigns some primitive polynomials for every output of 4bit LFSR. So that for every clock the output of 4bit LFSR changes, simultaneously the tapping polynomial also changes for 64bit LFSR, which results in more randomness of numbers.

if 4bit output is ----
4'd0: feedback<= prn[63]^prn[3]^prn[2]^prn[0];
4'd1: feedback<= prn[63]^prn[7]^prn[6]^prn[5]^prn[2]^prn[1]^prn[0];
4'd2: feedback<= prn[63]^prn[5]^prn[4]^prn[0]^prn[1]^prn[3];
4'd3: feedback<= prn[63]^prn[5]^prn[4]^prn[1];
4'd4: feedback<= prn[63]^prn[7]^prn[4]^prn[2];
4'd5: feedback<=prn[63]^prn[7]^prn[6]^prn[5]^prn[2]^prn[1];
4'd6: feedback<= prn[63]^prn[9]^prn[8]^prn[6]^prn[1]^prn[5]^prn[2];
4'd7: feedback<= prn[63]^prn[10]^prn[9]^prn[8]^prn[1]^prn[6];
4'd8: feedback<= prn[63]^prn[10]^prn[8]^prn[7]^prn[1]^prn[6]^prn[5];
4'd9: feedback<= prn[63]^prn[12]^prn[11]^prn[9]^prn[8]^prn[7]^prn[5]^prn[4];
4'd10: feedback<= prn[63]^prn[12]^prn[13]^prn[11]^prn[10]^prn[9]^prn[8]^prn[6];
4'd11: feedback<= prn[62]^prn[60]^prn[59];
4'd12: feedback<= prn[61]^prn[58]^prn[57];
4'd13: feedback<= prn[60]^prn[58]^prn[55];
4'd14: feedback<= prn[61]^prn[62]^prn[57]^prn[59]^prn[58];
4'd15: feedback<= prn[61]^prn[60]^prn[57]^prn[56]^prn[55];
This is how the tapping changes dynamically.

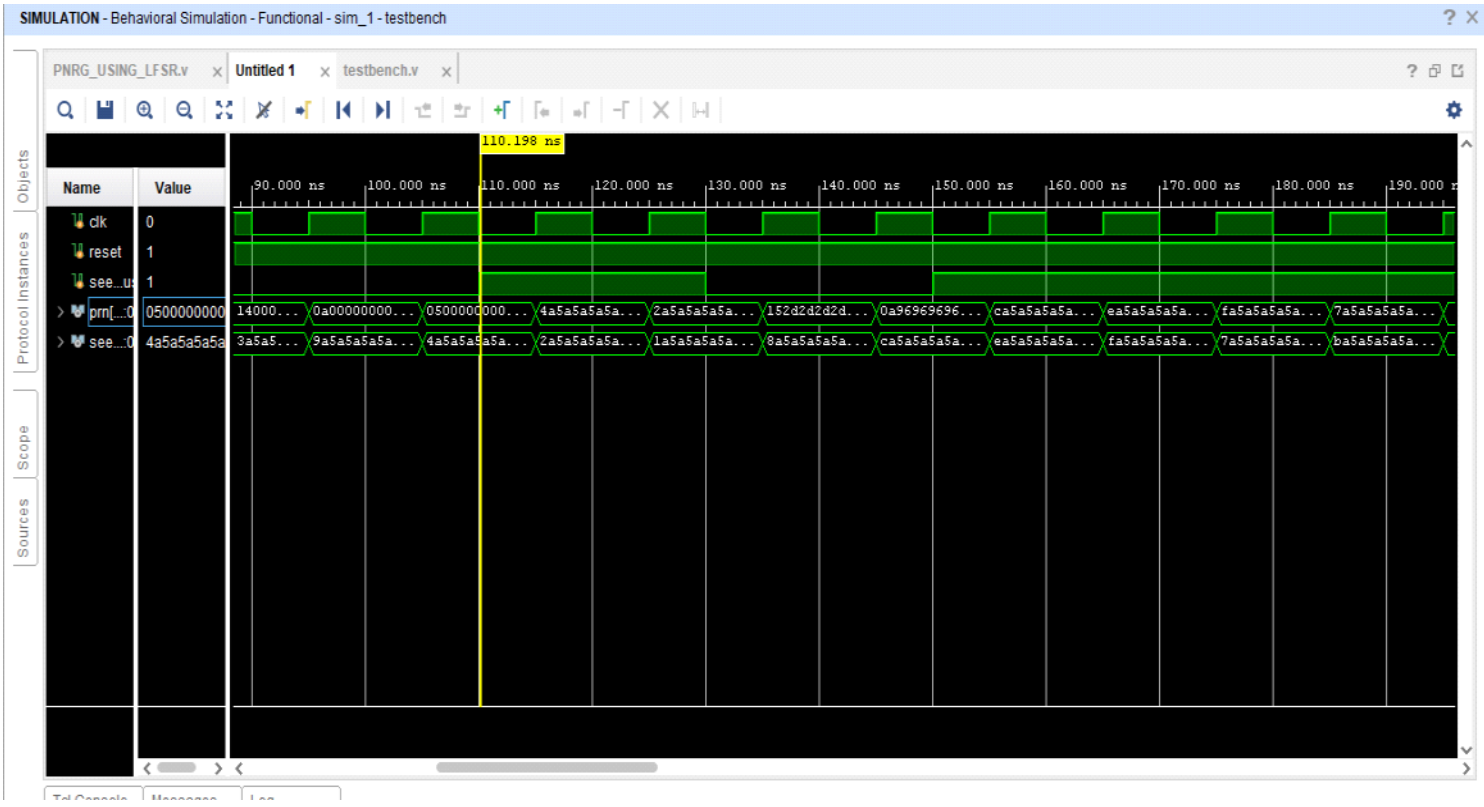
Converting 4bit LFSR output to 64Bit LFSR seed value:
To convert this I used a filler pattern, Which includes 60bits, this is common for all the outputs. But the first 4bit of the 64bit seed value contains the output of 4bit LFSR and other 60 bits are filler pattern(common for all outputs).
I Used filler pattern as: A5A5_A5A5_A5A5_A5A. -> Which Includes sequence of 1's and 0's.

Outputs:





Screen clipping taken: 7/28/2025 10:09 PM



Screen clipping taken: 7/28/2025 10:11 PM

Can observe when seed is 1, output PNRG is seeded by new seed input. Otherwise, the shifting continues.

My Notes:

1.Faced Difficulty in Converting 4bit LFSR to 64Bit LFSR Seed(Mechanism)
2. Unable to understand the behaviour of Seed_status at first.

References:

1. 2024 IEEE International Students' Conference on Electrical, Electronics and Computer Science
979-8-3503-4846-0/24/$31.00 ©2024 IEEE
Adaptive Reconfigurable LFSR: Dynamic Tapping and Reseeding for Enhanced Pseudo Random Number Generation
Adaptive Reconfigurable LFSR: Dynamic Tapping and Reseeding for Enhanced Pseudo Random Number Generation | IEEE Conference Publication | IEEE Xplore