XILINX®

Linear Feedback Shift Registers in Virtex Devices

Author: Maria George and Peter Alfke

Summary

This application note describes the implementation of Linear Feedback Shift Registers (LFSR) using the SRL macroavailable in the Virtex™ and Virtex-II series of FPGAs. The optimal implementation of a 15-bit LFSR, a 52-bit LFSR, and a 118-bit LFSR are discussed in this application note.

Introduction

The Virtex series of FPGAs have an SRL (Shift Register LUT) macro. This macro implements very efficient shift registers varying in length from one to sixteen bits as determined by the address lines. The length can be either set to a static value or it can be changed dynamically. This application note describes the use of the SRL to implement Linear Feedback Shift Registers in Virtex series devices. The configurable elements are called Configuration Logic Blocks (CLBs).

Each Virtex series CLB contains four logic cells, organized into two slices. A logic cell includes a 4-input look-up table, carry logic, and a storage element. Each CLB in a Virtex-II device has four identical slices. Each slice contains 2, 4-input LUTs, two registers, carry logic, and other dedicated logic.

The Virtex-II devices have a similar macro, a selectable SRL that utilizes one LUT to implement a 16-bit shift register. This macro has two outputs, one is the dedicated output from the 16th register, and the other is selected using the address lines. This enables the macros to be cascadable to implement 32-bit, 64-bit, and 128-bit shift registers in one CLB.
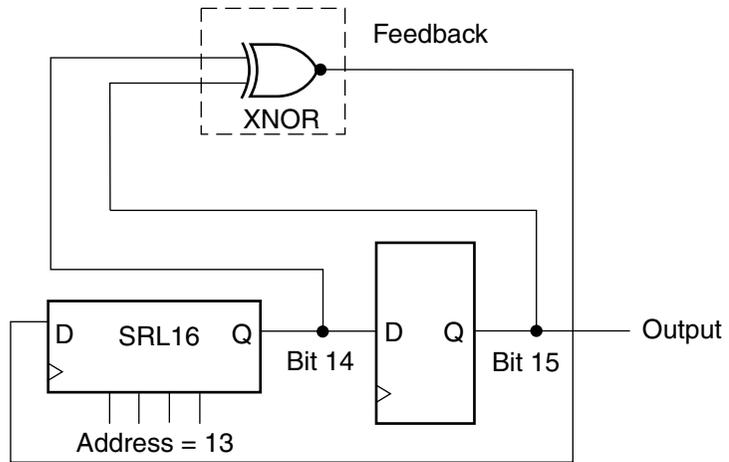
LFSR

Linear Feedback Shift Registers sequence through (2^n - 1) states, where *n* is the number of registers in the LFSR. At each clock edge, the contents of the registers are shifted right by one position. There is feedback from predefined registers or taps to the left most register through an exclusive-NOR (XNOR) or an exclusive-OR (XOR) gate. A value of all "1"s is illegal in the case of a XNOR feedback. A count of all "0"s is illegal for an XOR feedback. This state is illegal because the counter would remain locked-up in this state. The LFSR in this application note is implemented with XNOR feedback.

A 4-bit LFSR sequences through (2^4 - 1) = 15 states (the state `1111` is in the lock-up/illegal state). From Table 1 the feedback taps are 4, 3. On the other hand, a 4-bit binary up-counter would sequence through 2^4 = 16 states with no illegal states. LFSR counters are very fast since they use no carry signals. However, the dedicated carry in Virtex devices is rarely a speed-limiting factor because it is intrinsically fast. LFSRs can replace conventional binary counters in performance critical applications where the count sequence is not important (e.g., FIFOs). LFSRs are also used as pseudo-random bit stream generators. They are important building blocks in the implementation of encryption and decryption algorithms.
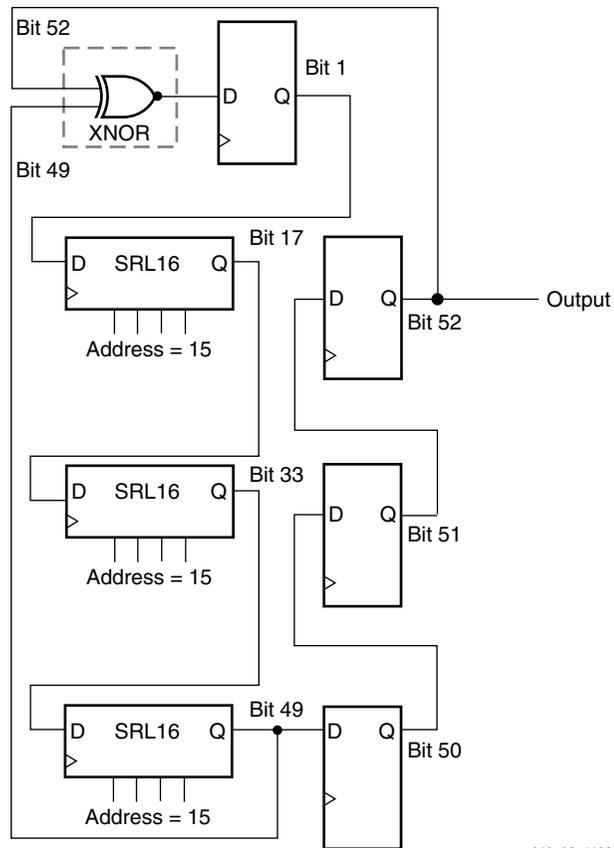
Implementation

The implementation of the 15-bit LFSR, using just one slice of a CLB is illustrated in Figure 1. One logic cell in this slice is used for a 2-input XNOR function. The other logic cell in this slice is used for the implementation of the 14-bit SRL and the fifteenth register of the 15-bit LFSR. By assigning a static value of decimal 13 to the address lines, the output length of the SRL is set to 14. The taps for the 15-bit LFSR are 14 and 15. Table 1 lists the appropriate taps for all LFSRs up to 168 bits.

x210_01_110200

Figure 1: **15-bit LFSR Implemented Using One Slice**

x210_02_110200

Figure 2: **52-bit LFSR Implemented Using Two Slices**
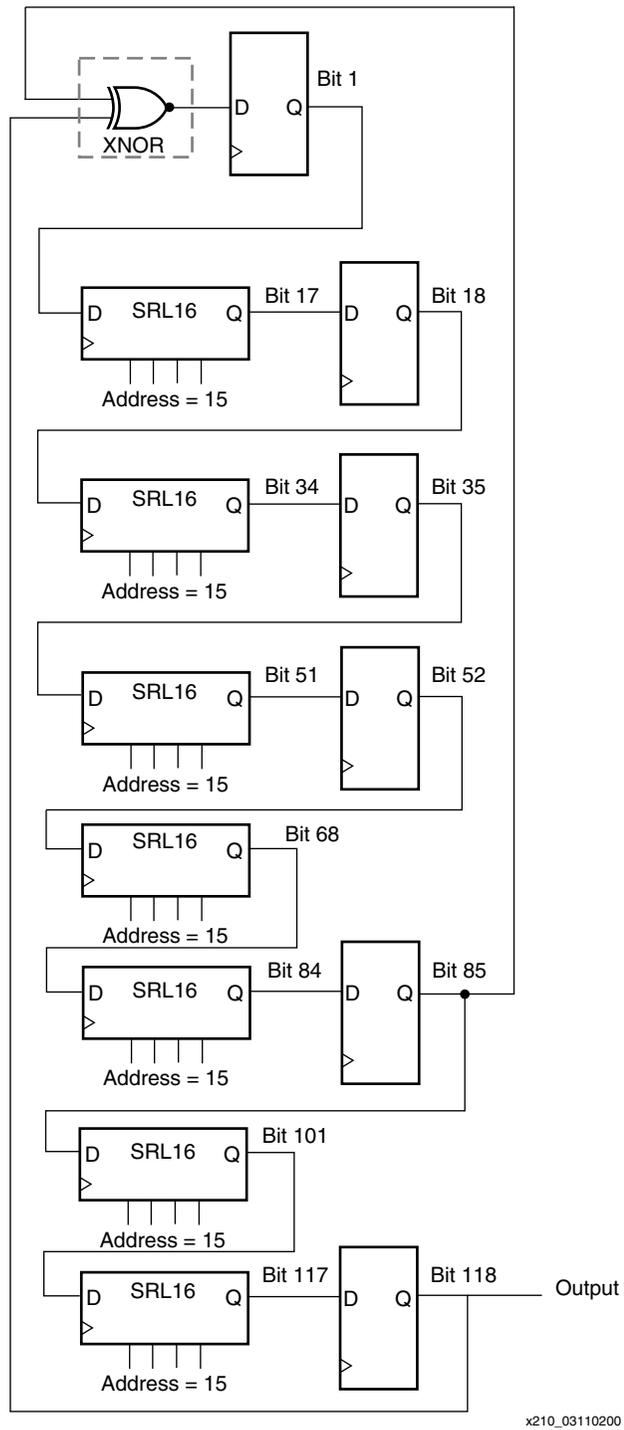**(One CLB for Virtex Series; Half a CLB for Virtex-II Series)**

x210_03110200

Figure 3: **118-bit LFSR Implemented Using Four Slices (2 CLBs for Virtex; 1 CLB for Virtex-II)**

Figure 2 is a 52-bit LFSR implementation using one CLB. The first logic cell in the CLB is used for the 2-input XNOR function and the first register of the 52-bit LFSR. The second logic cell is used to implement the 16-bit SRL and the 52nd register of the 52-bit LFSR. The connection structure shown is required to give access to the 49th register as one of the taps required for the 52-bit LFSR. The output length of the SRL is set to 16 by assigning a static value of decimal 15 to the address lines. From Table 1, the taps for the 52-bit LFSR are 49 and 52. A 118-bit LFSR implementation using two CLBs is shown in Figure 3. The taps for the 118-bit LFSR are 85 and 118 and are taken from Table 1.

Linear Feedback Shift Register Taps

Table 1 lists the appropriate taps for maximum-length LFSR counters of up to 168 bits. The outputs are designated as "1" through "n" with one as the first stage.

Table 1: **Taps for Maximum-Length LFSR Counters**

n	XNOR from	n	XNOR from	n	XNOR from	n	XNOR from
3	3,2	45	45,44,42,41	87	87,74	129	129,124
4	4,3	46	46,45,26,25	88	88,87,17,16	130	130,127
5	5,3	47	47,42	89	89,51	131	131,130,84,83
6	6,5	48	48,47,21,20	90	90,89,72,71	132	132,103
7	7,6	49	49,40	91	91,90,8,7	133	133,132,82,81
8	8,6,5,4	50	50,49,24,23	92	92,91,80,79	134	134,77
9	9,5	51	51,50,36,35	93	93,91	135	135,124
10	10,7	52	52,49	94	94,73	136	136,135,11,10
11	11,9	53	53,52,38,37	95	95,84	137	137,116
12	12,6,4,1	54	54,53,18,17	96	96,94,49,47	138	138,137,131,130
13	13,4,3,1	55	55,31	97	97,91	139	139,136,134,131
14	14,5,3,1	56	56,55,35,34	98	98,87	140	140,111
15	15,14	57	57,50	99	99,97,54,52	141	141,140,110,109
16	16,15,13,4	58	58,39	100	100,63	142	142,121
17	17,14	59	59,58,38,37	101	101,100,95,94	143	143,142,123,122
18	18,11	60	60,59	102	102,101,36,35	144	144,143,75,74
19	19,6,2,1	61	61,60,46,45	103	103,94	145	145,93
20	20,17	62	62,61,6,5	104	104,103,94,93	146	146,145,87,86
21	21,19	63	63,62	105	105,89	147	147,146,110,109
22	22,21	64	64,63,61,60	106	106,91	148	148,121
23	23,18	65	65,47	107	107,105,44,42	149	149,148,40,39
24	24,23,22,17	66	66,65,57,56	108	108,77	150	150,97
25	25,22	67	67,66,58,57	109	109,108,103,102	151	151,148
26	26,6,2,1	68	68,59	110	110,109,98,97	152	152,151,87,86
27	27,5,2,1	69	69,67,42,40	111	111,101	153	153,152

Table 1: **Taps for Maximum-Length LFSR Counters**

n	XNOR from	n	XNOR from	n	XNOR from	n	XNOR from
28	28,25	70	70,69,55,54	112	112,110,69,67	154	154,152,27,25
29	29,27	71	71,65	113	113,104	155	155,154,124,123
30	30,6,4,1	72	72,66,25,19	114	114,113,33,32	156	156,155,41,40
31	31,28	73	73,48	115	115,114,101,100	157	157,156,131,130
32	32,22,2,1	74	74,73,59,58	116	116,115,46,45	158	158,157,132,131
33	33,20	75	75,74,65,64	117	117,115,99,97	159	159,128
34	34,27,2,1	76	76,75,41,40	118	118,85	160	160,159,142,141
35	35,33	77	77,76,47,46	119	119,111	161	161,143
36	36,25	78	78,77,59,58	120	120,113,9,2	162	162,161,75,74
37	37,5,4,3,2,1	79	79,70	121	121,103	163	163,162,104,103
38	38,6,5,1	80	80,79,43,42	122	122,121,63,62	164	164,163,151,150
39	39,35	81	81,77	123	123,121	165	165,164,135,134
40	40,38,21,19	82	82,79,47,44	124	124,87	166	166,165,128,127
41	41,38	83	83,82,38,37	125	125,124,18,17	167	167,161
42	42,41,20,19	84	84,71	126	126,125,90,89	168	168,166,153,151
43	43,42,38,37	85	85,84,58,57	127	127,126		
44	44,43,18,17	86	86,85,74,73	128	128,126,101,99		

Conclusion

Virtex and Virtex-II devices can implement very efficient Linear Feedback Shift Registers. LFSRs are especially useful for generating pseudo-random serial bit streams used in wireless applications including PN Code Generators and Gold Code Generators.

References

Wayne Stahnke, *Primitive Polynomials Modulo Two*, private communication in 1970.

Woody Johnson, FreeCore, *Linear Feedback Shift Register*, 1997.

Peter Alfke, *Efficient Shift Registers, LFSR Counters, and Long Pseudo-Random Sequence Generators,* Xilinx application note **XAPP052.**

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
08/06/99	1.0	Initial Xilinx release.
03/14/00	1.1	Reformatted into new template
01/09/01	1.2	Updated for Virtex-II series