LFSR counters implement binary polynomial generators

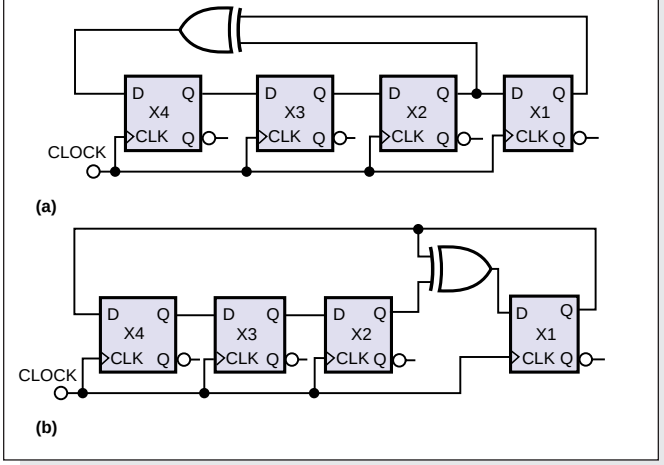
TOM BALPH, MOTOROLA SEMICONDUCTOR

LFSR counters make ideal pseudorandom-number generators, but make sure to provide a reset function to prevent lockup.

You can use linear-feedback shift registers (LFSRs) as alternatives to conventional bi-nary counters (**Reference 1**). An LFSR reduces the amount of required logic and minimizes routing complexity. A possible disadvantage is that the count sequence is not the normal binary increment or decrement sequence. An LFSR counter, in effect, implements a binary polynomial generator. These generators find common use for pseudorandom-number generation. This article provides some guidelines for implementing LFSR-based counters. Some general points include

- An LFSR with n flip-flops can implement only a $(2^n–1)$-state counter. The all-zeros state is normally not allowed because the counter locks up.
- Good design practice demands a reset condition that provides start-up in a known condition and also ensures that the counter does not power up in a zero condition and stay locked up.
- The choice of the polynomial used should ensure $2^n–1$ states—with no repeated states; such a polynomial is known as a "primitive," or maximal-length polynomial.

FIGURE 1



To implement an x^4+x+1-polynomial LFSR counter, you have the choice of a "many-to-one" configuration (a) or a "one-to-many" configuration (b).

TABLE 1—EXPONENTS FOR PRIMITIVE POLYNOMIALS

1:	0			
2:	1	0		
3:	1	0		
4:	1	0		
5:	2	0		
6:	1	0		
7:	1	0		
8:	6	5	1	0
9:	4	0		
10:	3	0		
11:	2	0		
12:	7	4	3	0
13:	4	3	1	0
14:	12	11	1	0
15:	1	0		
16:	5	3	2	0

To implement a counter with a divide ratio other than $2^n–1$, you must first select a primitive polynomial that has the proper degree. The degree, or power of two, must be large enough to allow the desired number range. As an example, a divide-by-35 counter must use a polynomial of the sixth degree (yielding $2^6–1=63$ possible states). You can typically find primitive polynomials in tables in textbooks that deal with testing and pseudorandom numbers. The values in **Table 1** derive from **Reference 2**.

The values in **Table 1** are the exponents of terms in primitive binary polynomials. The numbers listed represent the smallest number of terms for a primitive polynomial of each degree. For example, the entry 12: 7 4 3 0 represents the polynomial $x^{12}+x^7+x^4+x^3+1$. Once you select a polynomial, you

LFSR COUNTERS

first implement a counter for just that polynomial. As a design example, assume you need a divide-by-12 counter. From **Table 1**, you select the fourth-degree polynomial (x^4+x+1) because it allows as many as 15 possible states.

You can implement the polynomial in logic as either a "many-to-one" (**Figure 1a**) or a "one-to-many" (**Figure 1b**) design. Note that, although either approach implements the same polynomial, the count sequences differ. At this point, you should add a synchronous reset to the design to force an all-ones condition at reset.

Figure 2a shows the one-to-many design example. Through simulation, you can observe the count sequence and verify that the selected polynomial repeats after 2^n–1 states (in this case, 15 states) and that no state repeats within each sequence. **Table 2** gives the sequence for this example.

To complete the counter design, you can decode the desired final count and force the normal sequence to truncate back to the reset condition. For the divide-by-12 example, you decode state 12, in which the count value equals two hex. **Figure 2b** shows the reset-modification hardware.

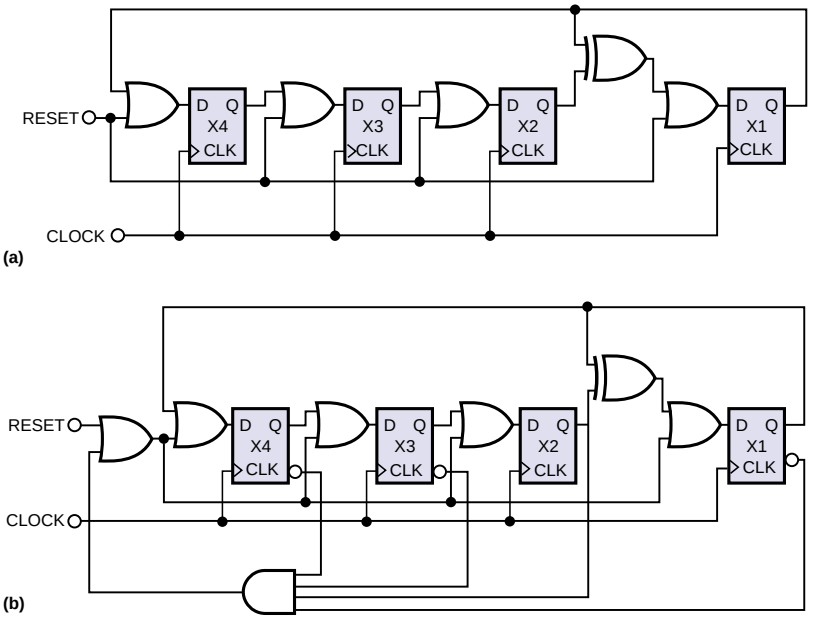
References

1. Dipert, Brian, "Shattering the programmable-logic speed barrier," *EDN*, May 22, 1997, pg 37.
2. Bardel, McAnney, and Savir, *Built-In Test for VLSI: Pseudo-random Techniques*, John Wiley and Sons, 1987.

Author's biography

Tom Balph is a 28-year veteran of Motorola (Tempe, AZ). He defines and designs system functions for Motorola's Semiconductor Products sector. Balph holds a BSEE from the University of Cincinnati and an MSEE from Arizona State University (Tempe, AZ). Balph's spare-time pursuits include woodworking and outdoor activities.

FIGURE 2



A synchronous reset forces an all-ones condition on reset in a divide-by-12 counter (a); decoding circuitry (b) forces the sequence to truncate back to the reset condition.

TABLE 2—SEQUENCE FOR DIVIDE-BY-12 COUNTER

State	Value (hex)
1	F
2	E
3	7
4	A
5	5
6	B
7	C
8	6
9	3
10	8
11	4
12	2 (decode to force a reset)
13	1
14	9
15	D