Pseudo-Random Number Generators

Types

- counter
- LCM
- LFSR

Sequence Generation

- time-based
- sequential

Linear Congruential Method (LCM)

$$X_{n+1} \leftarrow (aX_n + c) \bmod m$$

General Conditions

- $m > 0$
- $0 < a < m$
- $0 \le c < m$
- $0 \le X_0 < m$

Maximal Length Conditions

- m and c relatively prime
- a – 1 divisible by all prime factors of m
- a – 1 divisible by 4 if m is divisible by 4

Linear Feedback Shift Register (LFSR)

A maximal length LFSR (Linear Feedback Shift Register) can be made with any number of bits with the appropriate feedback (maximal length is $2^n - 1$, where n is the number of bits). The feedback is always from the last bit and one or more other bits and feeds back into the first bit, a ring configuration. The feedback bits are combined through either an XOR gate or an XNOR gate. If an XOR gate is used the only unused output pattern is all zeros and if an XNOR gate is used it is all ones. These are also the lock-up patterns, in that if the shift register gets into that state, it will never leave it.

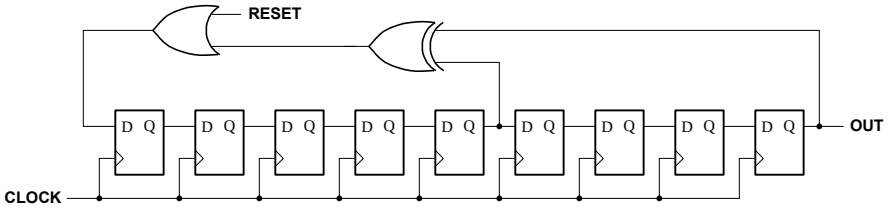
For use as random bit generators, LFSRs have the following properties. The number of 1's and 0's in the complete sequence ($2^n - 1$ clocks) is equal (actually within one due to the lock-up pattern). In a complete sequence half of the runs of 1 or 0 have length one, one quarter have length two, an eighth have length three, etc. And finally, the autocorrelation function of the output bit is a Kronecker delta at zero delay and $1/(2^n - 1)$ at all other delays (up to $2^n - 1$).

To form LFSRs of a given length the following feedback terms should be used (the bits are numbered from 0 to n–1 and the feedback is always to bit 0).

Number of Bits (n)	LFSR Sequence Length ($2^n - 1$)	Feedback Bits
3	7	2 1
4	15	3 2
5	31	4 2
6	63	5 4
7	127	6 5
8	255	7 5 4 3
9	511	8 4
10	1023	9 6
11	2047	10 8
12	4095	11 5 3 0
13	8191	12 3 2 0
14	16383	13 4 2 0
15	32767	14 13
16	65535	15 14 12 3
17	131071	16 13
18	262143	17 10
19	524287	18 5 1 0
20	1048575	19 16
21	2097151	20 18
22	4194303	21 20
23	8388607	22 17
24	16777215	23 22 21 16
25	33554431	24 21
26	67108863	25 5 1 0
27	134217727	26 4 1 0
28	268435455	27 24
29	536870911	28 26
30	1073741823	29 5 3 0
31	2147483647	30 27
32	4294967295	31 21 1 0
33	8589934591	32 19
34	17179869183	33 26 1 0
35	34359738367	34 32
36	68719476735	35 24
37	137438953471	36 4 3 2 1 0
38	274877906943	37 5 4 0
39	549755813887	38 34
40	1099511627775	39 4 3 2

Fibonacci LFSR

9-Bit Hardware Implementation

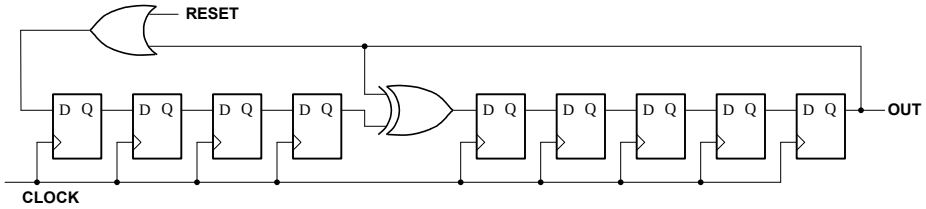


9-Bit Software Implementation

```
feedback ← ((lfsr >> 4) ^ (lfsr >> 8)) & 0x01
lfsr ← (lfsr << 1) | feedback
```

Galois LFSR

9-Bit Hardware Implementation



9-Bit Software Implementation

```
if ((lfsr & 0x100) != 0)
    lfsr ← (lfsr << 1) ^ 0x011
else
    lfsr ← (lfsr << 1)
```