Proyecto CLCript
Cuadernos de Laboratorio de Criptografía. Entrega nº 15. Ultima actualización 23/07/19
Autor: Dr. Jorge Ramió Aguirre (@criptored)
Práctica sobre registros de desplazamiento realimentados no linealmente NLFSR y realimentados linealmente LFSR, con FlujoLab

- Software FlujoLab: https://www.criptored.es/software/sw_m001m.htm
- Web de interés 1: https://www.jjj.de/mathdata/all-irredpoly.txt
- Web de interés 2: https://www.partow.net/programming/polynomials/index.html

Objetivos:

1. Entender cómo funcionan los registros de desplazamiento con realimentación no lineal NLFSR (Nonlinear Feedback Shift Register) y realimentación lineal LFSR (Linear Feedback Shift Register).
2. Comprobar el comportamiento de los registros LFSR cuando el polinomio asociado es de tipo factorizable, irreducible y primitivo.
3. Comprobar los postulados de Golomb en las secuencias de periodo máximo, m-secuencias.

I. Registro de desplazamiento realimentado no linealmente NLFSR de 4 celdas con esquema gráfico

Ejercicio 1

1.1. En FlujoLab, desde el menú Generadores – NLFSR – 4 etapas, introduce la semilla 0000 en las cajas inferiores que corresponden a $S_1S_2S_3S_4$ y genera la secuencia cifrante, guardando el informe de ejecución como CLCript15_1.1.

1.2. Comprueba que el periodo de la secuencia es máximo $2^4 = 16$ (secuencia de De Bruijn). Abre el archivo que has creado CLCript15_1.1.html y comprueba que el registro $S_1S_2S_3S_4$ pasa por todos sus estados posibles, desde 0000 hasta 1111, lógicamente no en orden.

1.3. ¿Si no existiese esa puerta NOT en el registro NLFR se podría poner usar esta semilla 0000?

1.4. Cambia las dos puertas lógicas OR como AND y con la misma semilla 0000 genera la secuencia sin guardar el informe. ¿Qué ha pasado?

1.5. Cambia ahora sólo una de las puertas lógicas OR por AND y con la misma semilla 0000 observa qué sucede. Si cambias ahora la sustitución de esa puerta OR por AND en sentido contrario, ¿qué sucede con el periodo? Justifica lo que ha pasado.

1.6. Pon las dos puertas lógicas OR en AND y con la semilla 1111 comprueba que tampoco generas ninguna secuencia. ¿Por qué?

1.7. Con la semilla $S_1S_2S_3S_4 = 0001$, genera las 4 secuencias posibles si las puertas lógicas son OR-OR, OR-AND, AND-OR y AND-AND. Observa que obtienes períodos iguales a 16, 16, 14 y 4. Es decir, el periodo máximo no está garantizado.

1.8. ¿Por qué buscamos que el registro nos entregue un periodo máximo de bits?

Comprueba tu trabajo:

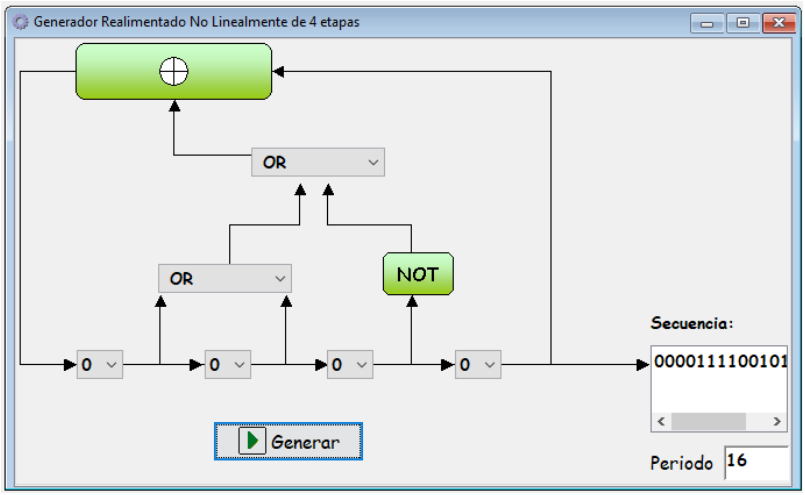

Figura 1. Secuencia de periodo máximo para el NLFSR de 4 celdas con semilla $S_1S_2S_3S_4$ = 0000.

Registros	Bit Si
0000	0
1000	0
1100	0
1110	0
1111	1
0111	1
0011	1
1001	1
0100	0
1010	0
1101	1
0110	0
1011	1
0101	1
0010	0
0001	1
0000	Se repite la semilla

Figura 2. El NLFSR de 4 celdas y semilla $S_1S_2S_3S_4$ = 0000 pasa por todos sus restos.

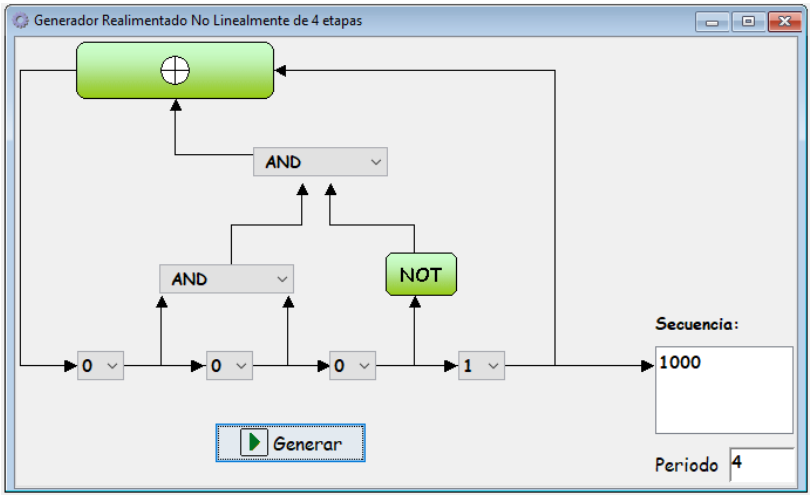

Figura 3. Secuencia cifrante para el NLFSR de 4 celdas con semilla $S_1S_2S_3S_4$ = 0001.

II. Registro de desplazamiento realimentado no linealmente NLFSR de 6 celdas con esquema gráfico

Ejercicio 2

2.1. En FlujoLab, desde el menú Generadores – NLFSR – 6 etapas, introduce la semilla 000000 en las cajas inferiores que corresponden a $S_1S_2S_3S_4S_5S_6$ y comprueba que, como ahora no tenemos ninguna puerta que niegue los bits (NOT, NAND o NOR), no podremos generar ninguna secuencia con esa semilla. El registro no evoluciona, se queda en 0.

2.2. En el registro NLFSR, establece la puerta lógica superior en AND y las dos puertas lógicas inferiores en OR. Introduce la semilla $S_1S_2S_3S_4S_5S_6$ = 110000, genera la secuencia y guarda el informe como CLCript15_2.2. Comprueba que se obtiene un periodo igual a 6.

2.3. Cambia la semilla a $S_1S_2S_3S_4S_5S_6$ = 111000, genera nuevamente la secuencia y guarda el informe CLCript15_2.3. Comprueba que se obtiene ahora un periodo igual a 30.

2.4. Cambia la semilla a $S_1S_2S_3S_4S_5S_6$ = 111100, genera nuevamente la secuencia y guarda el informe CLCript15_2.4. Comprueba que se obtiene ahora un periodo igual a 11.

2.5. Según lo que has visto en los apartados 2.2, 2.3 y 2.4, ¿es recomendable este tipo de generador de secuencia cifrante NLFSR en que según la semilla que se use, cambia el periodo de la secuencia de bits?

2.6. Si lo deseas comprobar, para ninguna semilla desde 000001 hasta 111111 se obtiene para este registro NLFSR el periodo máximo esperable de este registro $T = 2^6 - 1 = 63$. El menos 1 se debe a que la cadena de todos ceros, aquí 00000, está aquí prohibida como ya hemos visto al no haber ninguna puerta con negación de bits.

2.7. Comprueba que obtienes ese periodo $T = 2^6 - 1 = 63$ si el registro NLFSR tiene abajo una puerta AND y la otra OR, o una OR y la otra AND da igual el orden. Usa por ejemplo la semilla $S_1S_2S_3S_4S_5S_6$ = 111111.

Comprueba tu trabajo:

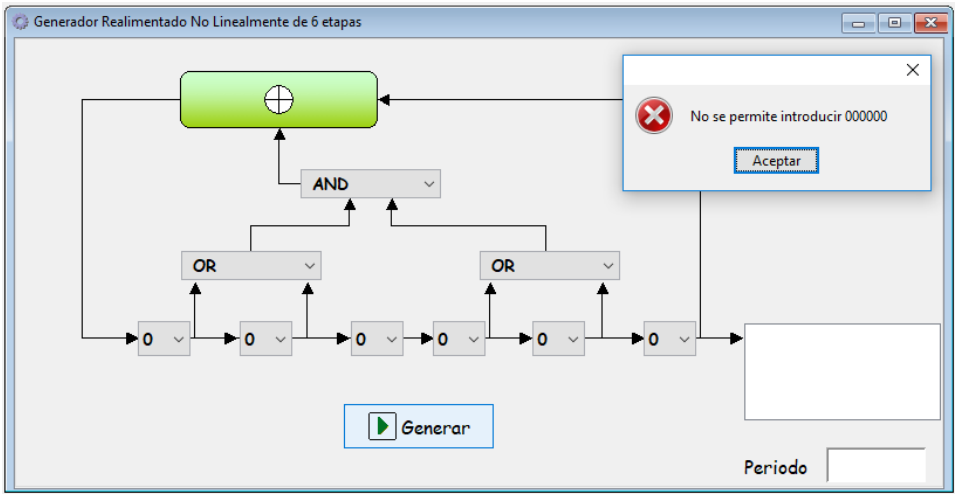

Figura 4. No se puede generar secuencia cifrante en el NLFSR de 6 celdas con puertas AND y OR, pero sin puertas NOT, NOR o NAND, con semilla $S_1S_2S_3S_4S_5S_6$ = 000000.

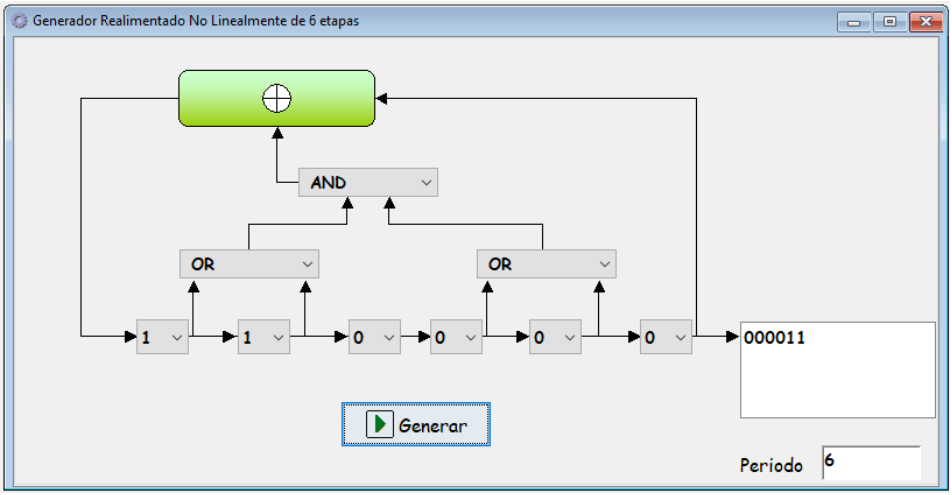

Figura 5. Registro NLFSR de 6 celdas con semilla $S_1S_2S_3S_4S_5S_6$ = 110000 y periodo T = 6.

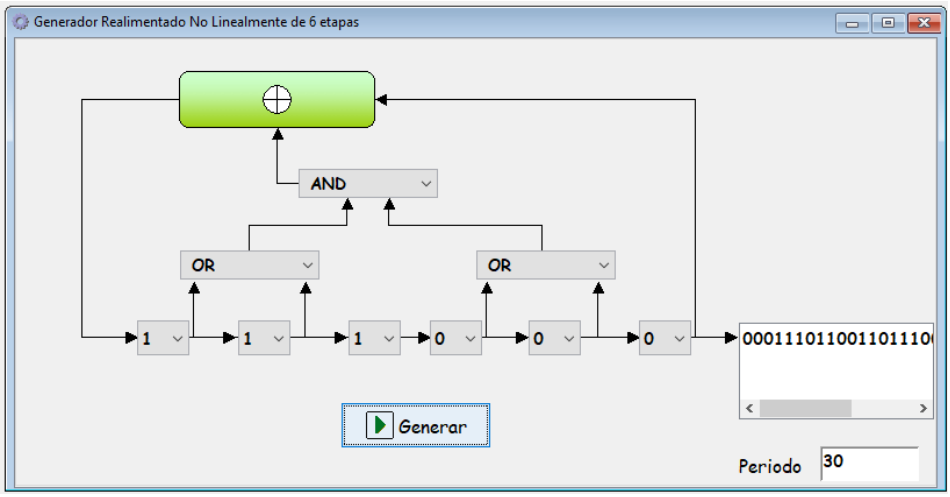

Figura 6. Registro NLFSR de 6 celdas con semilla $S_1S_2S_3S_4S_5S_6$ = 111000 y periodo T = 30.

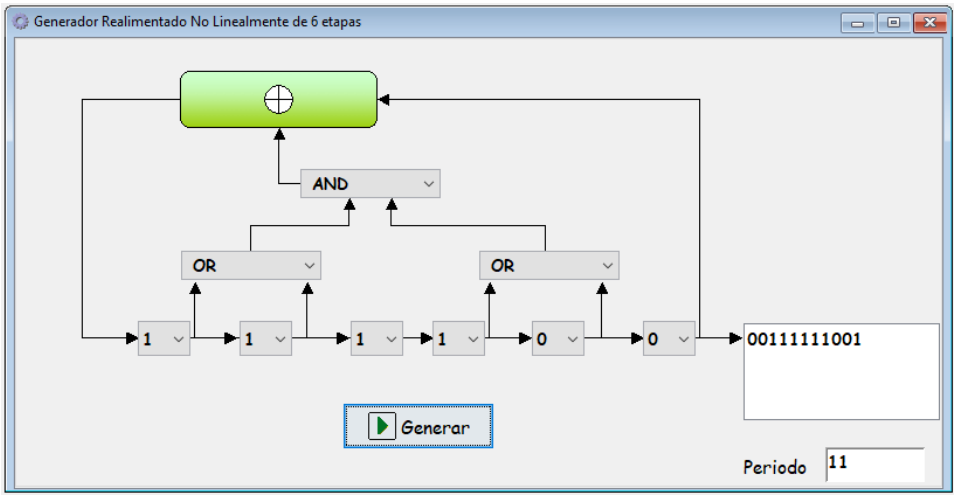

Figura 7. Registro NLFSR de 6 celdas con semilla $S_1S_2S_3S_4S_5S_6$ = 111100 y periodo T = 11.

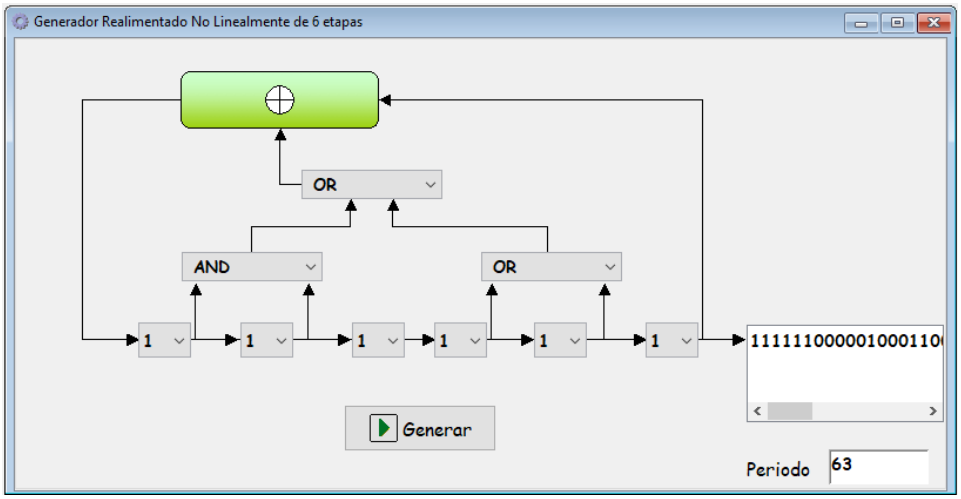

Figura 8. Registro NLFSR de 6 celdas con semilla $S_1S_2S_3S_4S_5S_6$ = 111111 y periodo T = 63.

III. Registros de desplazamiento realimentados linealmente LFSR de 4 y 6 celdas con esquema gráfico

Ejercicio 3

3.1. En FlujoLab, desde el menú Generadores – LFSR – 4 etapas, verás que solamente la última celda (S_4) está conectada al XOR. De las demás celdas, al menos una más deberá estar también conectada al XOR para que el registro evolucione y entregue los bits de la secuencia cifrante. Conecta la celda 1 (S_1) de la izquierda al XOR.

3.2. ¿Qué sucede si intentas generar una secuencia con la semilla $S_1S_2S_3S_4$ = 0000?

3.3. Introduce como semilla $S_1S_2S_3S_4$ = 0001 y genera la secuencia cifrante. Guarda el informe de la generación como CLCript15_3.3.html y comprueba que se genera un periodo máximo $T = 2^n - 1 = 2^4 - 1 = 15$ bits.

3.4. Conecta ahora todas las celdas ($S_1S_2S_3$) al XOR y con la semilla $S_1S_2S_3S_4$ = 0001 genera la cifrante. Guarda el informe de la generación como CLCript15_3.4.html y comprueba que ahora se genera un periodo de solamente 5 bits.

3.5. En FlujoLab, desde el menú Generadores – LFSR – 6 etapas, conecta la celda 1 (S_1) de la izquierda al XOR. Introduce como semilla $S_1S_2S_3S_4S_5S_6$ = 000001 y genera la secuencia cifrante. Guarda el informe de la generación como CLCript15_3.5.html y comprueba que se genera un periodo máximo $T = 2^n - 1 = 2^6 - 1 = 63$ bits.

3.6. Conecta ahora todas las celdas ($S_1S_2S_3S_4S_5$) al XOR y con la semilla $S_1S_2S_3S_4S_5S_6$ = 000001 genera la secuencia. Guarda el informe de la generación como CLCript15_3.6.html y comprueba que ahora se genera un periodo de solamente 7 bits.

3.7. ¿Por qué crees que manteniendo la misma semilla y sólo modificando las conexiones de las celdas en el registro, ha cambiado el periodo? Tanto en el registro de 4 celdas como en el registro de 6 celdas, se ha usado la misma semilla (la más pequeña posible 0001 y 000001), en un caso se conectaba solamente la primera celda y en el otro caso se conectaban todas las celdas, además de la celda S_n que por defecto siempre debe estar conectada.

3.8. Esto lo veremos y justificaremos en el siguiente Apartado IV Registros LFSR con polinomio asociado.

Comprueba tu trabajo:

Figura 9. Registro LFSR de 4 celdas con celda S_1 conectada y semilla $S_1S_2S_3S_4 = 0000$, no permitida.

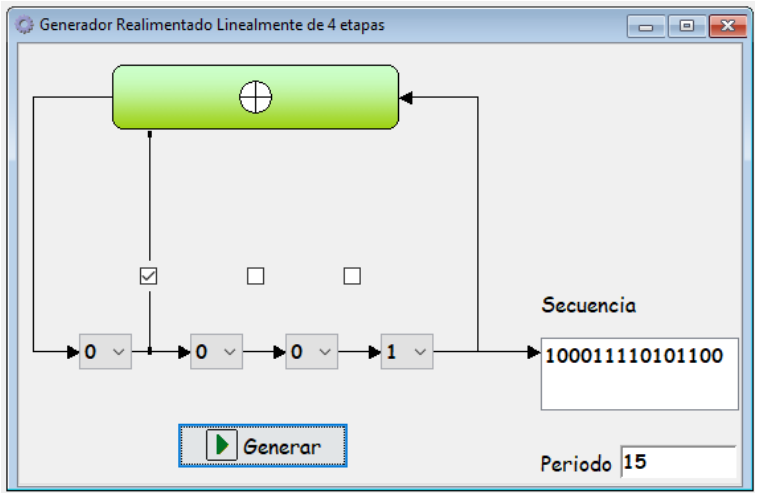

Figura 10. Registro LFSR de 4 celdas con celda S_1 conectada y semilla $S_1S_2S_3S_4 = 0001$, que entrega un periodo máximo $T = 2^n - 1 = 2^4 - 1 = 15$.

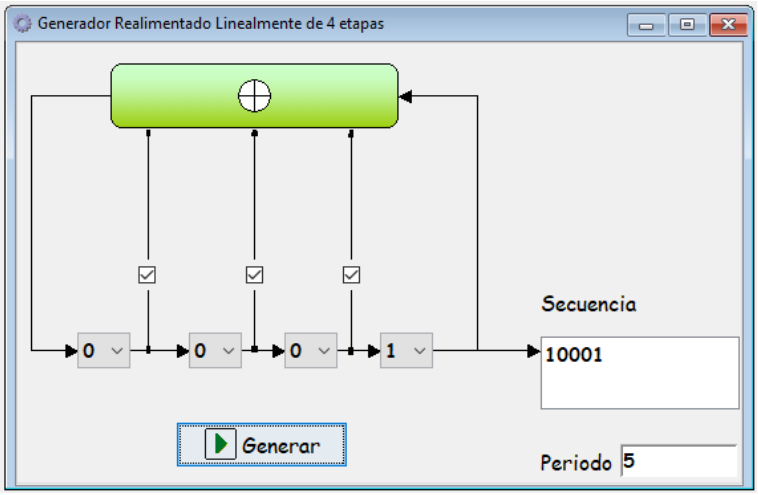

Figura 11. Registro LFSR de 4 celdas con celdas $S_1S_2S_3$ conectadas y semilla $S_1S_2S_3S_4 = 0001$, que entrega un periodo $T = 5$.

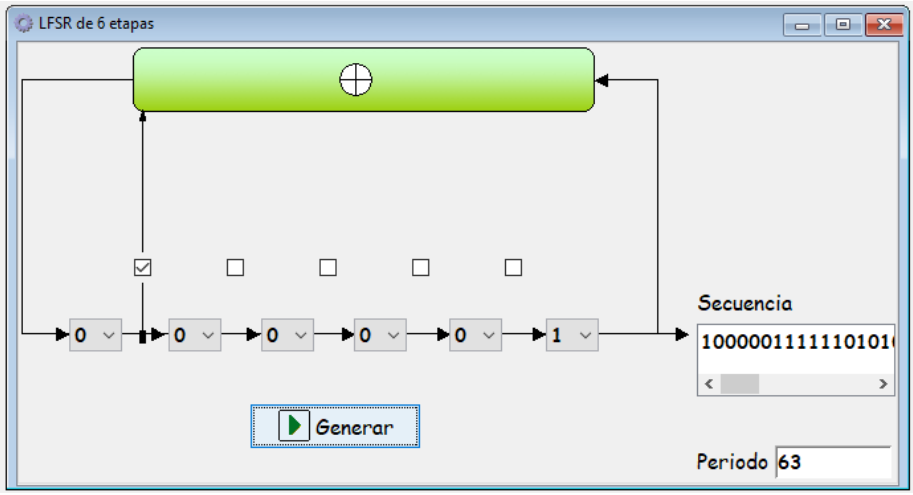

Figura 12. Registro LFSR de 6 celdas con celda S_1 conectada y semilla $S_1S_2S_3S_4S_5S_6$ = 000001, que entrega un periodo máximo T = $2^n - 1 = 2^6 - 1 = 63$.

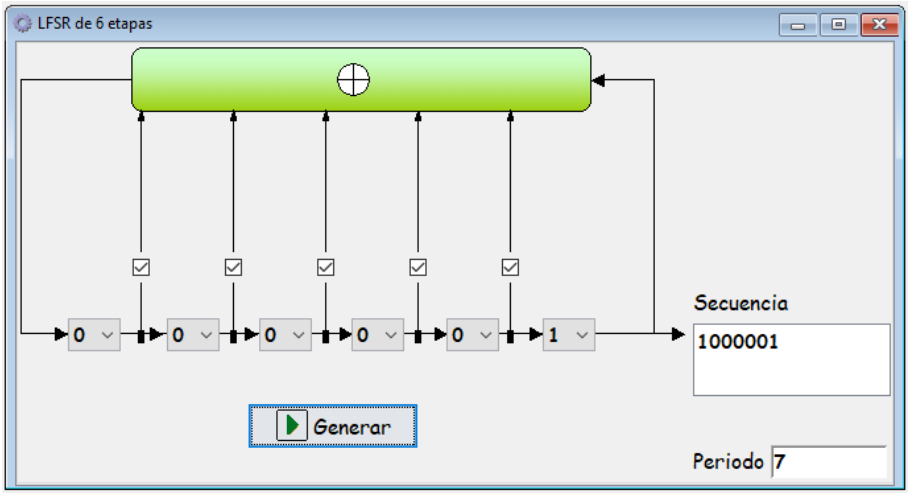

Figura 13. Registro LFSR de 6 celdas con celdas $S_1S_2S_3S_4S_5$ conectadas y semilla $S_1S_2S_3S_4S_5S_6$ = 000001, que entrega un periodo T = 7.

IV. Registros LFSR con polinomio asociado en registros de 4 celdas con esquema gráfico

Ejercicio 4

Generaremos secuencias cifrantes con registros LFSR con la figura de esquema de 4 celdas, a los que se les asociará un polinomio que indicará qué celdas están conectadas a la puerta XOR. De esta manera, tendremos tres tipos de polinomios: los factorizables, los irreducibles y los primitivos. El número total de polinomios de grado 4 con la celda 4 siempre conectada al XOR y, al menos, otra celda también conectada son estos siete:

$x^4 + x^3 + 1$	Polinomio primitivo	(conectadas la celda 4 y la celda 3)
$x^4 + x^2 + 1$	Polinomio factorizable	(conectadas la celda 4 y la celda 2)
$x^4 + x + 1$	Polinomio primitivo	(conectadas la celda 4 y la celda 1)
$x^4 + x^3 + x^2 + 1$	Polinomio factorizable	(conectadas la celda 4 la celda 3 y la celda 2)
$x^4 + x^3 + x + 1$	Polinomio factorizable	(conectadas la celda 4 la celda 3 y la celda 1)
$x^4 + x^2 + x + 1$	Polinomio factorizable	(conectadas la celda 4 la celda 2 y la celda 2)
$x^4 + x^3 + x^2 + x + 1$	Polinomio irreducible	(conectadas la celda 4 y las celdas 1, 2 y 3)

4.1. Polinomios irreducibles

4.1.1. Con el polinomio $x^4 + x^2 + 1$, genera la secuencia con semillas 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100,

1101, 1110 y 1111. Comprueba que el periodo es igual a 6, excepto para las semillas 0110, 1011 y 1101 en que el periodo es 3.

Factorización de $x^4 + x^2 + 1$ mod 2:

$(x^2 + x + 1)(x^2 + x + 1)$ mod 2 = $(x^4 + x^3 + x^2) + (x^3 + x^2 + x) + (x^2 + x + 1)$ mod 2

$(x^2 + x + 1)(x^2 + x + 1)$ mod 2 = $x^4 + 2x^3 + 3x^2 + 2x + 1$ mod 2 = $x^4 + x^2 + 1$

4.1.2. Con el polinomio $x^4 + x^3 + x + 1$, genera la secuencia y comprueba que los periodos obtenidos en función de la semilla que se use son los siguientes:

0001 genera secuencia con periodo T = 6.
0010 genera secuencia con periodo T = 3.
0011 genera secuencia con periodo T = 6.
0100 genera secuencia con periodo T = 3.
0101 genera secuencia con periodo T = 2.
0110 genera secuencia con periodo T = 3.
0111 genera secuencia con periodo T = 6.
1000 genera secuencia con periodo T = 6.
1001 genera secuencia con periodo T = 3.
1010 genera secuencia con periodo T = 2.
1011 genera secuencia con periodo T = 3.
1100 genera secuencia con periodo T = 6.
1101 genera secuencia con periodo T = 3
1110 genera secuencia con periodo T = 6.
1111 genera secuencia con periodo T = 1.

Factorización de $x^4 + x^3 + x + 1$ mod 2:

$(x^2 + x + 1)(x^2 + 1)$ mod 2 = $(x^4 + x^2) + (x^3 + x) + (x^2 + 1)$ mod 2

$(x^2 + x + 1)(x^2 + 1)$ mod 2 = $x^4 + x^3 + 2x^2 + x + 1$ mod 2 = $x^4 + x^3 + x + 1$.

4.1.3. Con el polinomio $x^4 + x^3 + x^2 + 1$, genera la secuencia con semillas 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110 y 1111. Comprueba que el periodo se mantiene constante e igual a 7, excepto para la semilla 1111 en que el periodo es T = 1.

Factorización de $x^4 + x^3 + x^2 + 1$ mod 2:

$(x^3 + x + 1)(x + 1)$ mod 2 = $(x^4 + x^3) + (x^2 + x) + (x + 1)$ mod 2

$(x^3 + x + 1)(x + 1)$ mod 2 = $x^4 + x^3 + x^2 + 2x + 1$ mod 2 = $x^4 + x^3 + x^2 + 1$.

4.1.4. Con el polinomio $x^4 + x^2 + x + 1$, repite el apartado 4.1.3. Observa que se obtiene lo mismo que en el apartado 4.1.3:

Factorización de $x^4 + x^2 + x + 1$ mod 2:

$(x^3 + x^2 + 1)(x + 1)$ mod 2 = $(x^4 + x^3) + (x^3 + x^2) + (x + 1)$ mod 2

$(x^3 + x^2 + 1)(x + 1)$ mod 2 = $x^4 + 2x^3 + x^2 + x + 1$ mod 2 = $x^4 + x^2 + x + 1$.

4.2. Polinomios irreducibles

4.2.1. Con el polinomio $x^4 + x^3 + x^2 + x + 1$ genera la secuencia con todas las semillas posibles. Observa que en este caso se obtiene un periodo constante T = 5 para todas las semillas. Observa que el periodo de la secuencia con polinomios irreducibles es un factor (5) del periodo máximo posible, en este caso $15 = 2^4 - 1$.

4.3. Polinomios primitivos

4.3.1. Con el polinomio $x^4 + x + 1$ genera la secuencia con todas las semillas posibles. Guarda el informe de la generación como CLCript15_4.3.html y comprueba que se genera un periodo máximo $T = 2^n - 1 = 2^4 - 1 = 15$ bits. Abre el archivo CLCript15_4.3.html y comprueba que el registro pasa por todos sus estados posibles desde 0001 hasta 1111, lógicamente no en orden.

4.3.2. Con el polinomio $x^4 + x^3 + 1$ repite el apartado 4.3.1 y comprueba que se obtiene el mismo resultado de periodo máximo constante.

Comprueba tu trabajo:

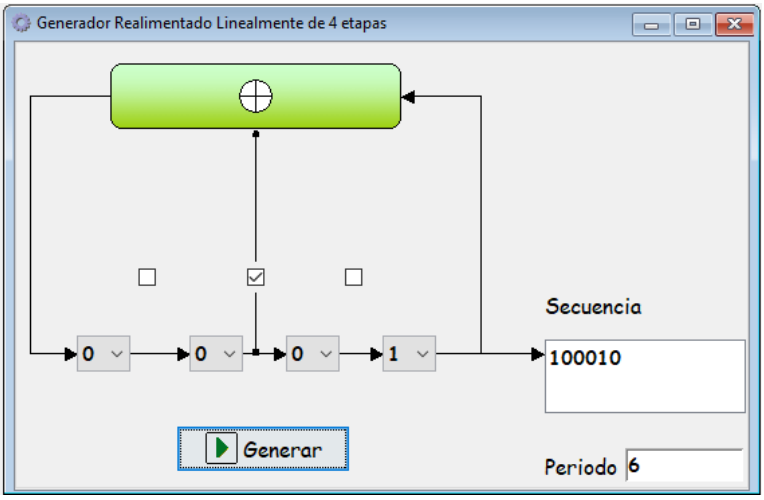

Figura 14. Registro LFSR de 4 celdas con polinomio factorizable $x^4 + x^2 + 1$, que entrega un periodo T = 6 para la semilla 0001.

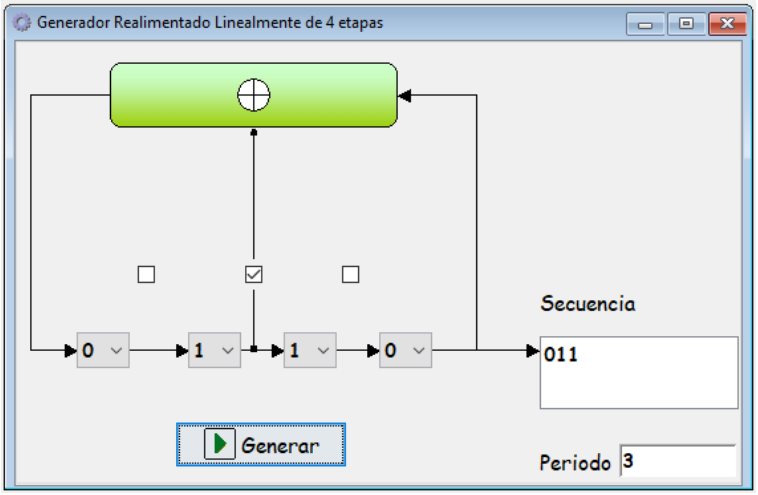

Figura 15. Registro LFSR de 4 celdas con polinomio factorizable $x^4 + x^2 + 1$, que entrega un periodo T = 3 para la semilla 0110.

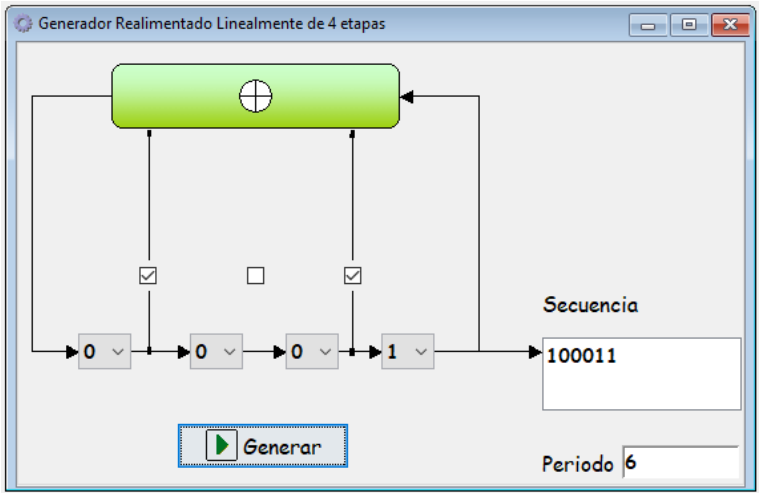

Figura 16. Registro LFSR de 4 celdas con polinomio factorizable $x^4 + x^3 + x + 1$, que entrega un periodo T = 6 para la semilla 0001.

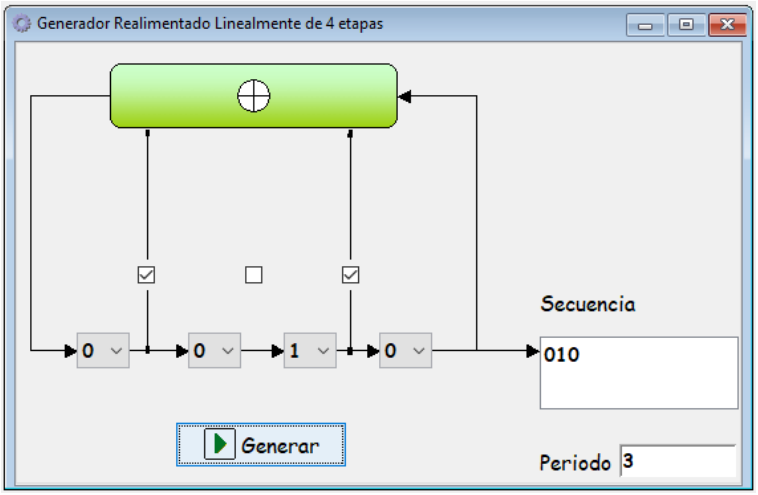

Figura 17. Registro LFSR de 4 celdas con polinomio factorizable $x^4 + x^3 + x + 1$, que entrega un periodo T = 3 para la semilla 0010.

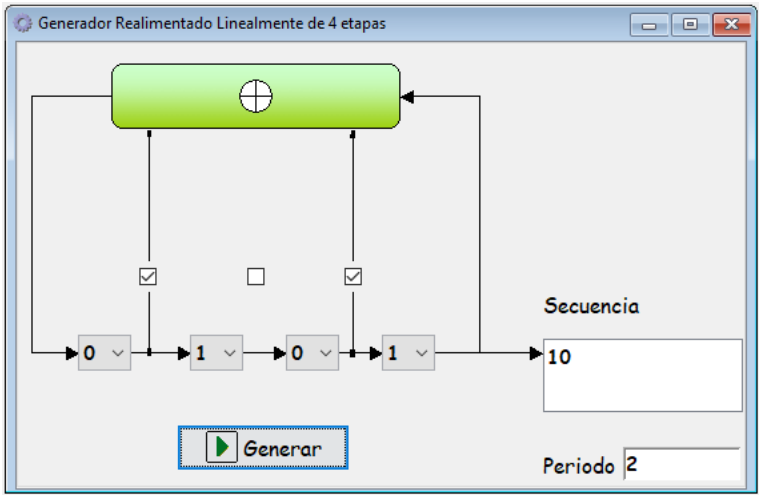

Figura 18. Registro LFSR de 4 celdas con polinomio factorizable $x^4 + x^3 + x + 1$, que entrega un periodo T = 2 para la semilla 0101.

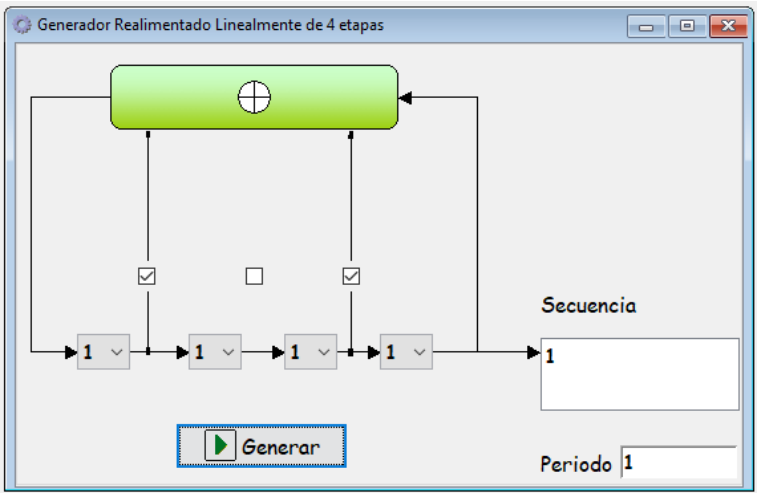

Figura 19. Registro LFSR de 4 celdas con polinomio factorizable $x^4 + x^3 + x + 1$, que entrega un periodo T = 1 para la semilla 1111.

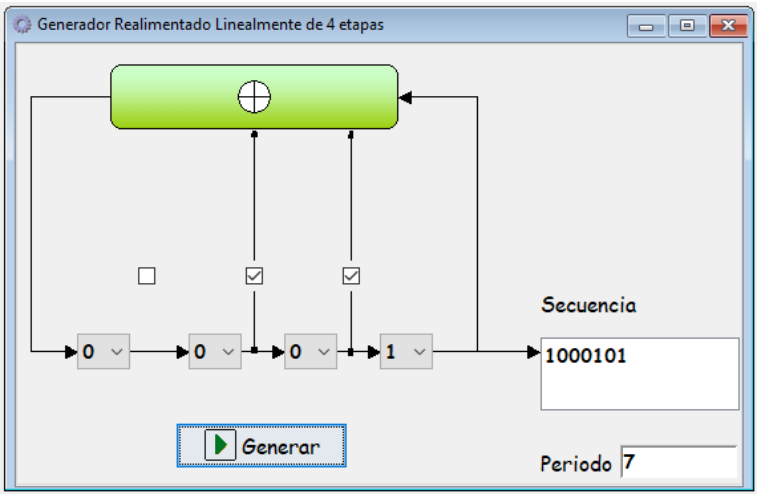

Figura 20. Registro LFSR de 4 celdas con polinomio factorizable $x^4 + x^3 + x^2 + 1$, que entrega un periodo T = 7 para la semilla 0001.

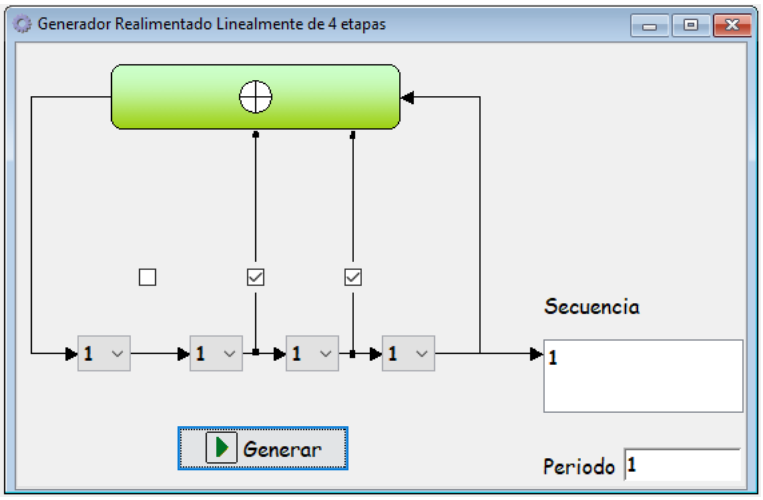

Figura 21. Registro LFSR de 4 celdas con polinomio factorizable $x^4 + x^3 + x^2 + 1$, que entrega un periodo T = 1 para la semilla 1111.

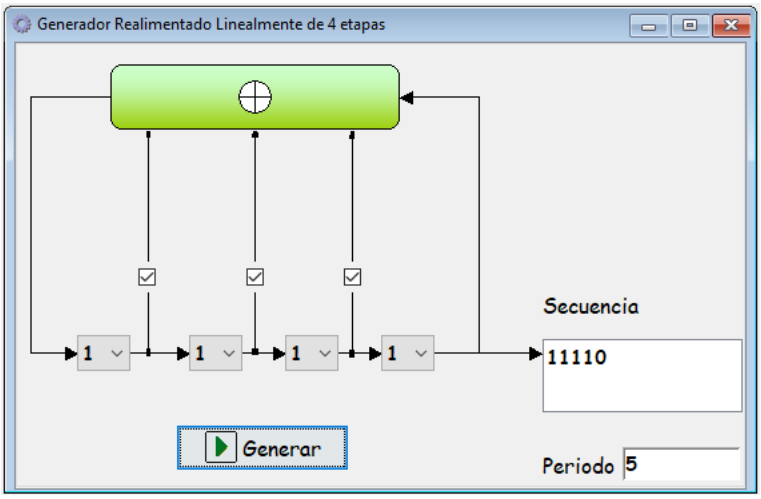

Figura 22. Registro LFSR de 4 celdas con polinomio irreducible $x^4 + x^3 + x^2 + x + 1$, que entrega un periodo T = 5 constante para cualquier semilla.

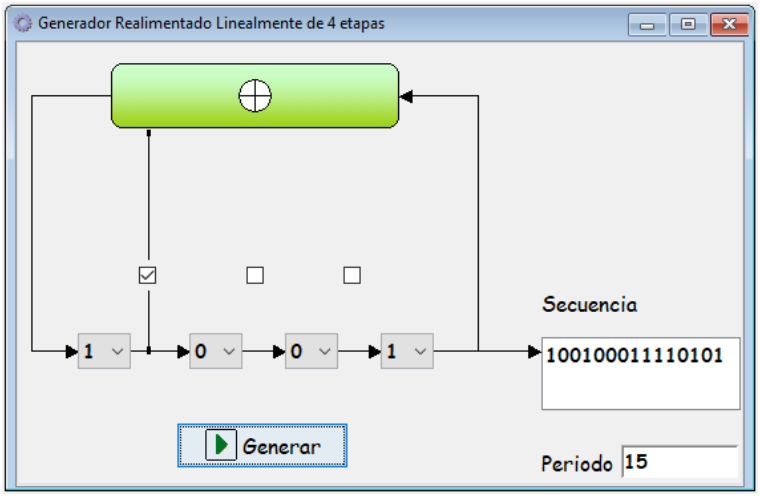

Figura 23. Registro LFSR de 4 celdas con polinomio primitivo $x^4 + x + 1$, que entrega un periodo máximo constante $T = 2^n - 1 = 2^4 - 1 = 15$ para cualquier semilla.

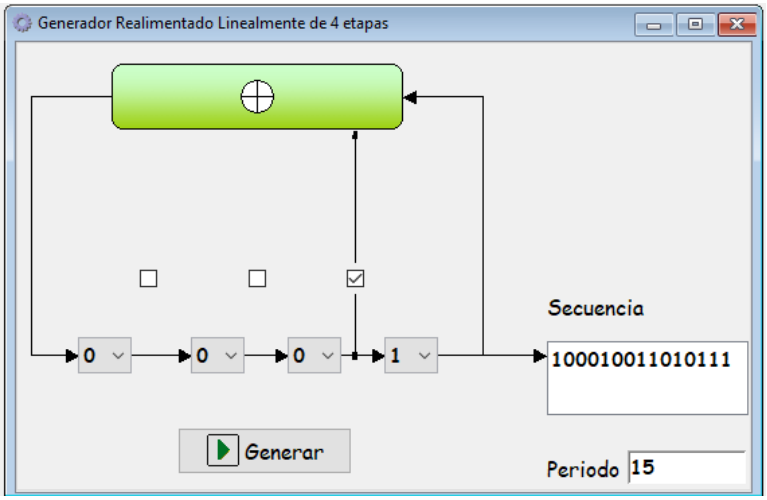

Figura 24. Registro LFSR de 4 celdas con polinomio primitivo $x^4 + x^3 + 1$, que entrega un periodo máximo constante $T = 2^n - 1 = 2^4 - 1 = 15$ para cualquier semilla.

V. Generación de secuencias con FlujoLab en modo general

Ejercicio 5

En FlujoLab, desde el menú Generadores – LFSR – General, podemos generar la secuencia de cualquier registro, simplemente indicando cuáles son las celdas conectadas que forman el polinomio y cuál es la semilla con la cual comienza el registro. El programa entrega una lista de algunos polinomios primitivos para prácticas de laboratorio; si se desea usar otros polinomios pueden elegirse desde los enlaces entregados al comienzo de este documento.

5.1. En esta sección, elige "Pols. Primitivos" y selecciona el polinomio primitivo de grado 8 (8,7,5,3,0 = $x^8 + x^7 + x^5 + x^3 + 1$). Con la semilla 11110001 genera la secuencia y guarda el informe como CLCript15_5.1.html. Comprueba que el periodo es $2^8 - 1 = 255$.

5.2. Selecciona el polinomio primitivo (15,1,0) y con semilla 111111111111111 genera la secuencia. Guarda el informe como CLCript15_5.2.html. Comprueba que el periodo es $2^{15} - 1 = 32.767$.

5.3. Desde https://www.partow.net/programming/polynomials/index.html observamos que (20,3,0 = $x^{20} + x^3 + 1$) es un polinomio primitivo de grado 20. Generamos la secuencia con una clave de todos 1. Aunque el tiempo de generación de la secuencia será excesivo (esto puede tardar incluso a varios minutos), guardaremos el informe de la clave como CLCript15_5.3.html porque la usaremos más adelante.

5.4. En la cifra de información en flujo bit a bit (que veremos en una próxima práctica) no se genera previamente toda la secuencia, sino que ésta se va generando a medida que se necesiten bits de secuencia de clave para el cifrado o el descifrado.

5.5. Lógicamente, se puede generar cualquier secuencia, introduciendo el polinomio a mano, por ejemplo 10,5,3,2,0.

Comprueba tu trabajo:

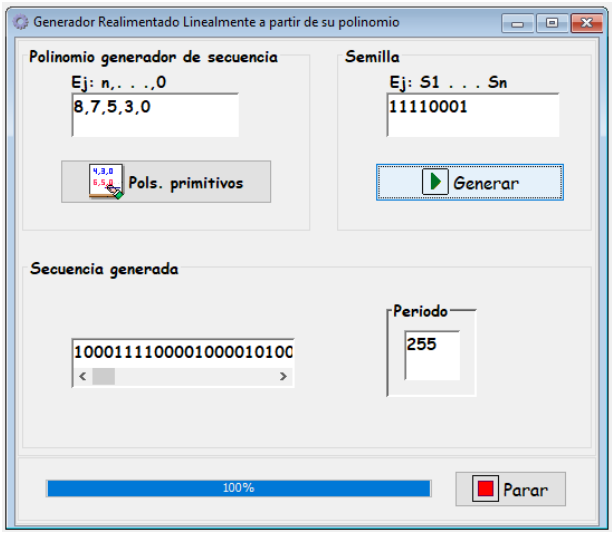

Figura 25. Registro LFSR con polinomio primitivo $x^8 + x^7 + x^5 + x^3 + 1$ y periodo $T = 2^8 - 1 = 255$.

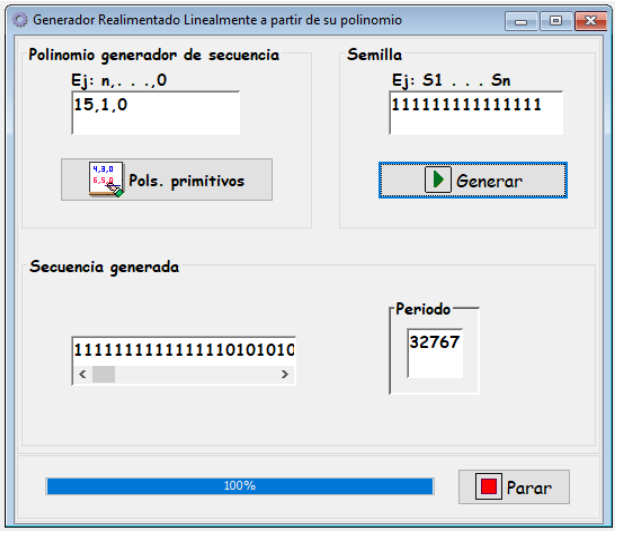

Figura 26. Registro LFSR con polinomio primitivo $x^{15} + x + 1$ y periodo $T = 2^{15} - 1 = 32.767$.

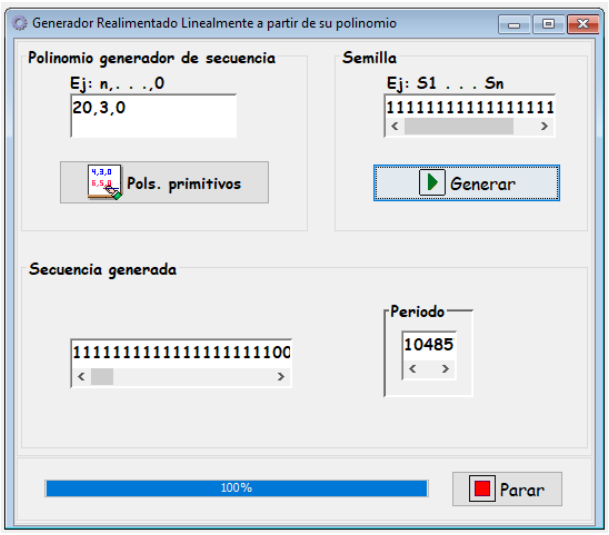

Figura 27. Registro LFSR con polinomio primitivo $x^{20} + x^3 + 1$ y periodo $T = 2^{20} - 1 = 1.048.575$

VI. Postulados de Golomb en m-secuencias

Ejercicio 6

6.1. Desde el menú Operaciones – Comprobar postulados, marca las pestañas del Primer Postulado, Segundo Postulado y Tercer postulado. Copia la cadena de 255 bits del archivo CLCript15_5.1.html en la ventana y pincha en Comprobar. Guarda el archivo como CLCript15_6.1.html. Observa las rachas en el postulado 2 de Golomb.

6.2. Repite el ejercicio 6.1 para la m-secuencia del archivo CLCript15_5.2.html.

6.3. Para el archivo CLCript15_5.3.html comprueba solamente el postulado 2 de Golomb (puede tardar varios minutos).

6.4. Observa que el total de rachas para un registro con polinomio primitivo de grado n es $2^n/2$. Así, para n = 8 tenemos $2^8/2 = 256/2 = 128$ rachas, para n = 15 tenemos $2^{15}/2 = 32.768/2 = 16.384$ rachas y para n = 20 tenemos $2^{20}/2 = 1.048.576/2 = 524.288$ rachas. Observa además cómo terminan las rachas para longitudes n y n-1.

6.5. ¿Por qué se da este comportamiento tan peculiar en rechas de longitud n y n-1?

Comprueba tu trabajo:

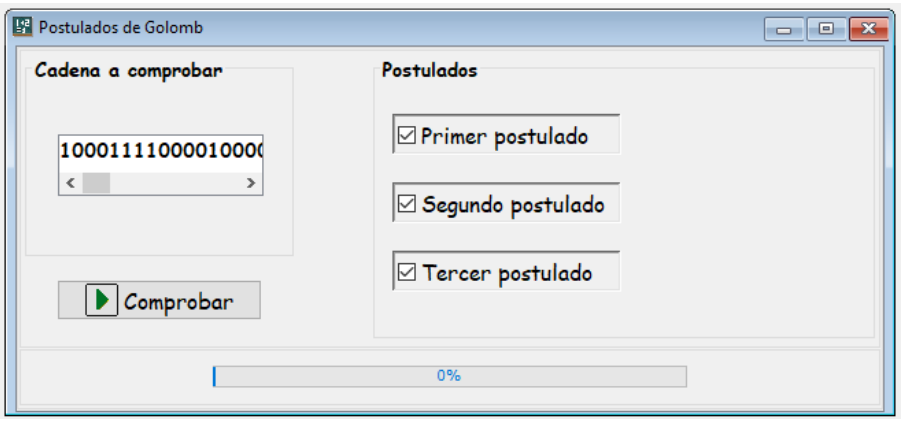

Figura 28. Introducción de datos para comprobar los postulados de Golomb en la m-secuencia de 255 bits.

Longitud	Rachas totales	Rachas de ceros	Rachas de unos
1	64	32	32
2	32	16	16
3	16	8	8
4	8	4	4
5	4	2	2
6	2	1	1
7	1	1	0
8	1	0	1
Totales	128	64	64

Figura 29. Distribución de rachas en una m-secuencia de grado 8, postulado 2 de Golomb.

Longitud	Rachas totales	Rachas de ceros	Rachas de unos
1	8192	4096	4096
2	4096	2048	2048
3	2048	1024	1024
4	1024	512	512
5	512	256	256
6	256	128	128
7	128	64	64
8	64	32	32
9	32	16	16
10	16	8	8
11	8	4	4
12	4	2	2
13	2	1	1
14	1	1	0
15	1	0	1
Totales	16384	8192	8192

Figura 30. Distribución de rachas en una m-secuencia de grado 15, postulado 2 de Golomb.

Longitud	Rachas totales	Rachas de ceros	Rachas de unos
1	262144	131072	131072
2	131072	65536	65536
3	65536	32768	32768
4	32768	16384	16384
5	16384	8192	8192
6	8192	4096	4096
7	4096	2048	2048
8	2048	1024	1024
9	1024	512	512
10	512	256	256
11	256	128	128
12	128	64	64
13	64	32	32
14	32	16	16
15	16	8	8
16	8	4	4
17	4	2	2
18	2	1	1
19	1	1	0
20	1	0	1
Totales	524288	262144	262144

Figura 31. Distribución de rachas en una m-secuencia de grado 20, postulado 2 de Golomb.

En un próximo cuaderno de laboratorio se estudiará la generación de secuencias con mayor complejidad lineal, operaciones de suma y multiplicación con LFSRs, cifrado y descifrado de textos y archivos, ataques de Berlekamp-Massey, y cifrado y descifrado con el algoritmo A5.

Madrid, 23 de julio de 2019
Dr. Jorge Ramió Aguirre