Cifrado en flujo

Registros de desplazamiento

eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea
KISA

Registros de desplazamiento

Características

- Son generadores de secuencias de bits.
- Permiten generar secuencias de períodos muy grandes.
- Tienen una estructura matemática bien conocida.
- Las secuencias que originan presentan buenas propiedades aleatorias.
- Son de fácil implementación en hardware.
- Ampliamente usados: generadores de secuencias aleatorias, test de circuitos, compresión de datos, ...
  GSM, GPS, Bluetooth, TV digital, PKZIP.

Registros de desplazamiento retroalimentados

Constan de:

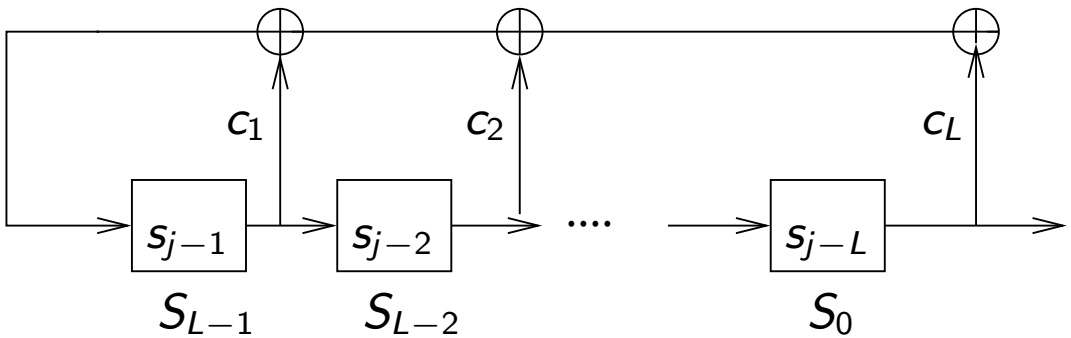
- Un **registro de desplazamiento**.
- Una **función feedback**.

Un *registro de desplazamiento retroalimentado* de *longitud* L es una estructura formada por L celdas de memoria $\{S_{L-1}, \ldots, S_0\}$, y una señal de reloj.

- Cada celda S_i puede almacenar un bit s_i, $i = 0, 1, \ldots, L-1$.
- Los bits $[s_{L-1}, \ldots, s_1, s_0]$ son el *estado inicial* del registro.
- A cada control de reloj:
  - El bit s_0 sale del registro.
  - Se produce un desplazamiento: el bit de S_i se desplaza a S_{i-1}.
  - Se calcula el nuevo bit de S_{L-1} mediante una función feedback.

Registros de desplazamiento retroalimentados **lineales** (linear feedback shift registers, LFSR)

- El registro de desplazamiento más simple es el lineal: LFSR
- La función de feedback es un XOR de ciertas posiciones del registro.



Ejemplo

Supongamos $L = 4$ y estado inicial $[s_3, s_2, s_1, s_0] = [0, 1, 1, 0]$.

t	S_3	S_2	S_1	S_0
0	0	1	1	0

En la primera unidad de tiempo:

- La salida del registro es $s_0 = 0$.
- El contenido de cada celda se desplaza a la derecha.
- El nuevo contenido s_4 de la celda S_3 se calcula mediante un XOR de algunos de los bits $\{s_3, s_2, s_1, s_0\}$. Por ejemplo,

$$s_4 \equiv s_3 + s_0 \quad \text{mód } 2.$$

t	S_3	S_2	S_1	S_0
0	0	1	1	0
1	0	0	1	1

Ejemplo

Repitiendo el proceso a cada control de reloj resulta

t	S_3	S_2	S_1	S_0
0	0	1	1	0
1	0	0	1	1
2	1	0	0	1
3	0	1	0	0
4	0	0	1	0
5	0	0	0	1
6	1	0	0	0
7	1	1	0	0

t	S_3	S_2	S_1	S_0
8	1	1	1	0
9	1	1	1	1
10	0	1	1	1
11	1	0	1	1
12	0	1	0	1
13	1	0	1	0
14	1	1	0	1
15	0	1	1	0

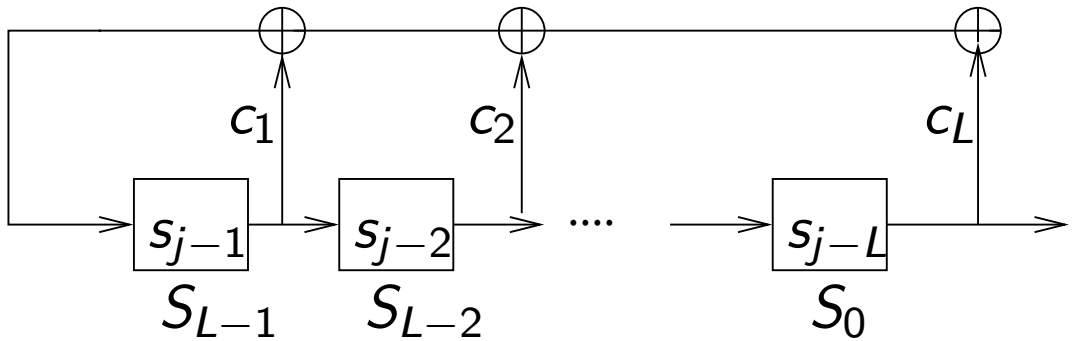
La secuencia de salida es

$$s_0, s_1, s_2, s_3, s_4, s_5, \cdots = 0, 1, 1, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 0, 1, \ldots$$

y es periódica con período 15.

Análisis de un LFSR:

Sea $s_{j-1}, \ldots, s_{j-L}$ el estado del registro en el instante t:



En el instante $t+1$ el nuevo contenido s_j de la celda S_{L-1} se calcula mediante un XOR de algunos de los bits del registro:

$$s_j \equiv c_1 s_{j-1} + c_2 s_{j-2} + \cdots + c_L s_{j-L} \quad \text{mód } 2,$$

donde c_j puede valer 0 o 1. Es decir,

$$1 \cdot s_j + c_1 s_{j-1} + c_2 s_{j-2} + \cdots + c_L s_{j-L} \equiv 0 \quad \text{mód } 2.$$

- Asociamos a esta expresión el polinomio

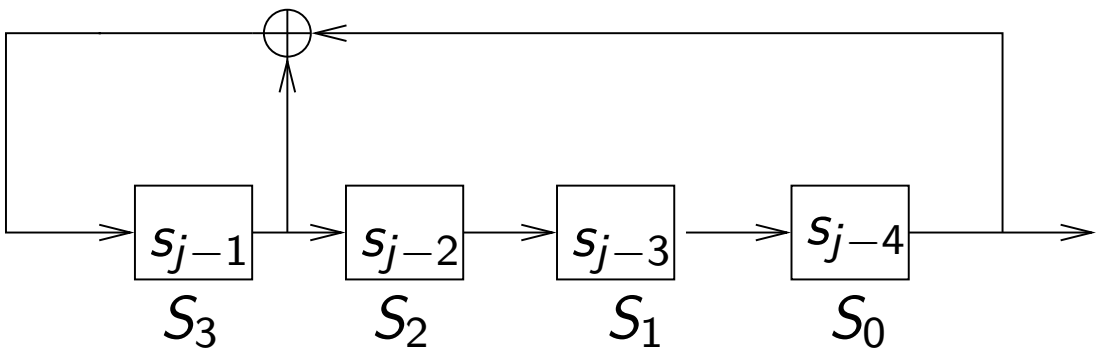
$$C(D) = 1 + c_1 D + c_2 D^2 + \cdots + c_L D^L.$$

Ejemplo (continuación)

En el ejemplo anterior,

$$s_j \equiv s_{j-1} + s_{j-4} \quad \text{mód } 2,$$

$$s_j + s_{j-1} + s_{j-4} \equiv 0 \quad \text{mód } 2.$$



$$L = 4, \qquad C(D) = 1 + D + D^4.$$

- El **LFSR queda definido** por $< L, C(D) >$:
  - L es la longitud del registro.
  - $c(D)$ se denomina **polinomio de conexión**.
- Si grado del polinomio = longitud del registro, el registro se denomina no singular.

- Un LFSR de longitud L puede presentar $2^L - 1$ estados internos.
  - Las secuencas producidas son periódicas, de período a lo sumo $2^L - 1$ (el estado $[0, 0, \ldots, 0]$ produce una secuencia de $0's$).
  - Bajo ciertas condiciones se alcanza el período máximo $2^L - 1$.

Períodos de las secuencias

Un LFSR $< L, C(D) >$ puede presentar $2^L - 1$ estados internos y por tanto el máximo período posible es $2^L - 1$.

Teorema

Si $C(D)$ es un polinomio primitivo de grado L, entonces la secuencia generada por el LFSR $< L, C(D) >$ es de período máximo $2^L - 1$ para cualquier estado inicial diferente de $[0, \ldots, 0]$.

¿Qué es un polinomio primitivo?

Definición (Polinomio irreducible)

Decimos que un polinomio $C(D)$ de grado mayor o igual que 1 es irreducible *si no es producto de otros dos polinomios de grado positivo.*

Ejemplo

$C(D) = D^5 + D^4 + D^3 + D$ **no** es irreducible en $\mathbb{Z}_2$:

$$D^5 + D^4 + D^3 + D = (D^3 + D + 1)(D^2 + D).$$

$C(D) = 1 + D^2 + D^4$ **no** es irreducible en $\mathbb{Z}_2$:

$$1 + D^2 + D^4 = (1 + D + D^2)^2.$$

Ejemplo

$C(D) = 1 + D + D^4$ es irreducible en $\mathbb{Z}_2$:

- Supongamos $C(D) = (a + D)(b + cD + dD^2 + D^3)$ con $a, b, c, d \in \mathbb{Z}_2$.
  Entonces $C(a) = (a + a)(b + ca + da^2 + a^3) = 0$. Pero

$$C(a) = 1 + a + a^4 = \left\{ \begin{array}{l} \text{si } a = 0 \text{ entonces } C(a) = 1, \\ \text{si } a = 1 \text{ entonces } C(a) = 1. \end{array} \right\} (\#)$$

- Supongamos $C(D) = (a + bD + D^2)(c + dD + D^2)$ con $a, b, c, d \in \mathbb{Z}_2$.

$$1 + D + D^4 = ac + (ad + bc)D + (a + bd + c)D^2 + (b + d)D^3 + D^4.$$

$$\Rightarrow \left\{ \begin{array}{ll} ac = 1 & \Rightarrow a = c = 1, \\ ad + bc = 1 & \Rightarrow d + b = 1, \\ a + bd + c = 0 & \\ b + d = 0 & \Rightarrow 1 = 0\ (\#) \end{array} \right.$$

Definición (Polinomio primitivo)

Un polinomio $C(D)$ de grado L con coeficientes en $\mathbb{Z}_2$ es un polinomio primitivo *si*

- *$C(D)$ es irreducible,*
- *para todo d divisor propio de $2^L - 1$, $C(D)$ no divide a $1 + D^d$.*

Observación. Todo polinomio irreducible $C(D)$ de grado L con coeficientes en $\mathbb{Z}_2$ divide a $1 + D^{2^L - 1}$.

Ejemplo

$C(D) = 1 + D + D^4$ es primitivo:

- $C(D)$ es irreducible en $\mathbb{Z}_2$.
- $2^4 - 1 = 15$. $C(D)$ no divide a $1 + D^5$ ni a $1 + D^3$.

El LFSR $< 4, C(D) >$ genera secuencias de período 15.

Ejemplo

$C(D) = 1 + D^2 + D^4$ no es primitivo (no es irreducible).

$s_j \equiv s_{j-2} + s_{j-4}$ mód 2. Estado inicial: $[s_3, s_2, s_1, s_0] = [0, 1, 1, 0]$

t	S_3	S_2	S_1	S_0
0	0	1	1	0
1	1	0	1	1
2	1	1	0	1
3	0	1	1	0

$= \left[\; s_3 \; \middle| \; s_2 \; \middle| \; s_1 \; \middle| \; s_0 \; \right]$

Período: 3.

Some Primitive Polynomials Mod 2

(1, 0)	(36, 11, 0)	(68, 9, 0)	(97, 6, 0)
(2, 1, 0)	(36, 6, 5, 4, 2, 1, 0)	(68, 7, 5, 1, 0)	(98, 11, 0)
(3, 1, 0)	(37, 6, 4, 1, 0)	(69, 6, 5, 2, 0)	(98, 7, 4, 3, 1, 0)
(4, 1, 0)	(37, 5, 4, 3, 2, 1, 0)	(70, 5, 3, 1, 0)	(99, 7, 5, 4, 0)
(5, 2, 0)	(38, 6, 5, 1, 0)	(71, 6, 0)	(100, 37, 0)
(6, 1, 0)	(39, 4, 0)	(71, 5, 3, 1, 0)	(100, 8, 7, 2, 0)
(7, 1, 0)	(40, 5, 4, 3, 0)	(72, 10, 9, 3, 0)	(101, 7, 6, 1, 0)
(7, 3, 0)	(41, 3, 0)	(72, 6, 4, 3, 2, 1, 0)	(102, 6 5 3 0)
(8, 4, 3, 2, 0)	(42, 7, 4, 3, 0)	(73, 25, 0)	(103, 9, 9)
(9, 4, 0)	(42, 5, 4, 3, 2, 1, 0)	(73, 4, 3, 2, 0)	(104, 11, 10, 1, 0)
(10, 3, 0)	(43, 6, 4, 3, 0)	(74, 7, 4, 3, 0)	(105, 16, 0)
(11, 2, 0)	(44, 6, 5, 2, 0)	(75, 6, 3, 1, 0)	(106, 15, 0)
(12, 6, 4, 1, 0)	(45, 4, 3, 1, 0)	(76, 5, 4, 2, 0)	(107, 9, 7, 4, 0)
(13, 4, 3, 1, 0)	(46, 8, 7, 6, 0)	(77, 6, 5, 2, 0)	(108, 31, 0)
(14, 5, 3, 1, 0)	(46, 8, 5, 3, 2, 1, 0)	(78, 7, 2, 1, 0)	(109, 5, 4, 2, 0)
(15, 1, 0)	(47, 5, 0)	(79, 9, 0)	(110, 6, 4, 1, 0)
(16, 5, 3, 2, 0)	(48, 9, 7, 4, 0)	(79, 4, 3, 2, 0)	(111, 10, 0)
(17, 3, 0)	(48, 7, 5, 4, 2, 1, 0)	(80, 9, 4, 2, 0)	(111, 49, 0)
(17, 5, 0)	(49, 9, 0)	(80, 7, 5, 3, 2, 1, 0)	(113, 9, 0)
(17, 6, 0)	(49, 6, 5, 4, 0)	(81, 4, 0)	(113, 15, 0)
(18, 7, 0)	(50, 4, 3, 2, 0)	(82, 9, 6, 4, 0)	(113, 30, 0)
(18, 5, 2, 1, 0)	(51, 6, 3, 1, 0)	(82, 8, 7, 6, 1, 0)	(114, 11, 2, 1, 0)
(19, 5, 2, 1, 0)	(52, 3, 0)	(83, 7, 4, 2, 0)	(115, 8, 7, 5, 0)
(20, 3, 0)	(53, 6, 2, 1, 0)	(84, 13, 0)	(116, 6, 5, 2, 0)
(21, 2, 0)	(54, 8, 6, 3, 0)	(84, 8, 7, 5, 3, 1, 0)	(117, 5, 2, 1, 0)
(22, 1, 0)	(54, 6, 5, 4, 3, 2, 0)	(85, 8, 2, 1, 0)	(118, 33, 0)
(23, 5, 0)	(55, 24, 0)	(86, 6, 5, 2, 0)	(119, 8, 0)
(24, 4, 3, 1, 0)	(55, 6, 2, 1, 0)	(87, 13, 0)	(119, 45, 0)
(25, 3, 0)	(56, 7, 4, 2, 0)	(87, 7, 5, 1, 0)	(120, 9, 6, 2, 0)
(26, 6, 2, 1, 0)	(57, 7, 0)	(88, 11, 9, 8, 0)	(121, 18, 0)
(27, 5, 2, 1, 0)	(57, 5, 3, 2, 0)	(88, 8, 5, 4, 3, 1, 0)	(122, 6, 2, 1, 0)
(28, 3, 0)	(58, 19, 0)	(89, 38, 0)	(123, 2, 0)
(29, 2, 0)	(58, 6, 5, 1, 0)	(89, 51, 0)	(124, 37, 0)
(30, 6, 4, 1, 0)	(59, 7, 4, 2, 0)	(89, 6, 5, 3, 0)	(125, 7, 6, 5, 0)
(31, 3, 0)	(59, 6, 5, 4, 3, 1, 0)	(90, 5, 3, 2, 0)	(126, 7, 4, 2, 0)
(31, 6, 0)	(60, 1, 0)	(91, 8, 5, 1, 0)	(127, 1, 0)
(31, 7, 0)	(61, 5, 2, 1, 0)	(91, 7, 6, 5, 3, 2, 0)	(127, 7, 0)
(31, 13, 0)	(62, 6, 5, 3, 0)	(92, 6, 5, 2, 0)	(127, 63, 0)
(32, 7, 6, 2, 0)	(63, 1, 0)	(93, 2, 0)	(128, 7, 2, 1, 0)
(32, 7, 5, 3, 2, 1, 0)	(64, 4, 3, 1, 0)	(94, 21, 0)	(129, 5, 0)
(33, 13, 0)	(65, 18, 0)	(94, 6, 5, 1, 0)	(130, 3, 0)
(33, 16, 4, 1, 0)	(65, 4, 3, 1, 0)	(95, 11, 0)	(131, 8, 3, 2, 0)
(34, 8, 4, 3, 0)	(66, 9, 8, 6, 0)	(95, 6, 5, 4, 2, 1, 0)	(132, 29, 0)
(34, 7, 6, 5, 2, 1, 0)	(66, 8, 6, 5, 3, 2, 0)	(96, 10, 9, 6, 0)	(133, 9, 8, 2, 0)
(35, 2, 0)	(67, 5, 2, 1, 0)	(96, 7, 6, 4, 3, 2, 0)	(134, 57, 0)

Fuente: Schneier, Applied Cryptography.

$(13, 4, 3, 1, 0)$ significa $D^{13} + D^4 + D^3 + D + 1$.

Ataques a los registros

A pesar de sus buenas condiciones de aletoriedad, las secuencias generadas por un LFSR $< L, C(D) >$ son predecibles:

Dada una parte de la secuencia producida por un registro de desplazamiento, es posible encontrar el registro que la origina, en ciertos casos.

Sea $s_0, s_1, s_2, \ldots, s_n$ una secuencia dada. Queremos encontrar el registro $< L, C(D) >$ que la genera. Denotamos

$$C(D) = 1 + c_1 D + c_2 D^2 + \cdots + c_L D^L,$$

sabemos que los términos de la sucesión deben satisfacer

$$s_j \equiv c_1 s_{j-1} + \cdots + c_L s_{j-L} \quad \text{mód } 2,$$

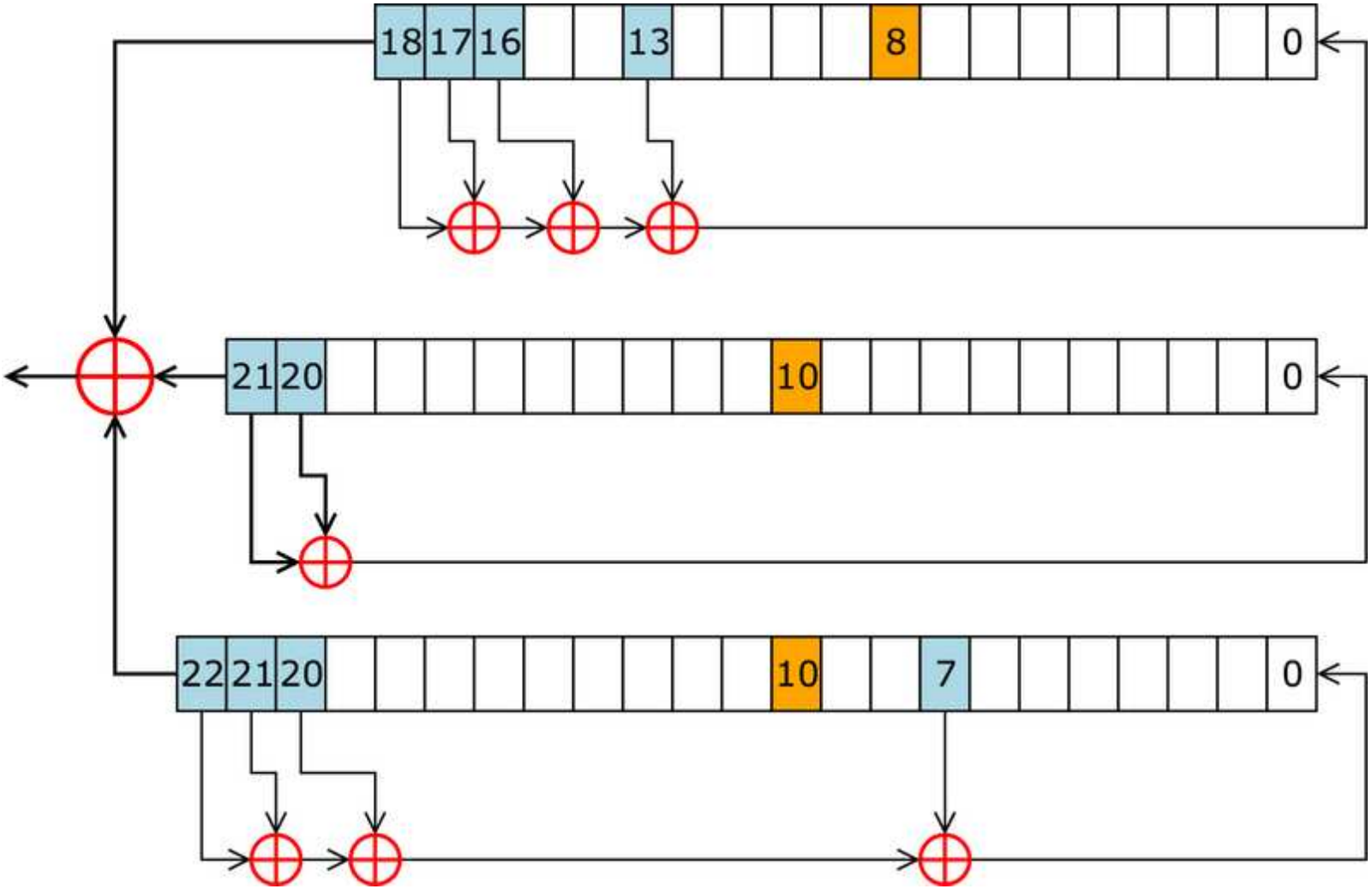
y queremos hallar los coeficientes $c_1, \ldots c_L$ del polinomio.

- Ataque 1: Conocida la longitud L del registro.
  Supongamos que conocemos $2L$ bits de una secuencia $s_j, s_{j-1}, \ldots, s_{j-2L+1}$. Podemos plantear el sistema de L ecuaciones con L incógnitas $(c_1, \ldots c_L)$:

$$\left.\begin{array}{lcll} s_j & \equiv & c_1 s_{j-1} + \cdots + c_L s_{j-L} & \text{mód } 2 \\ s_{j-1} & \equiv & c_1 s_{j-2} + \cdots + c_L s_{j-L-1} & \text{mód } 2 \\ & \vdots & & \\ s_{j-L+1} & \equiv & c_1 s_{j-L} + \cdots + c_L s_{j-2L+1} & \text{mód } 2 \end{array}\right\}$$

- Ataque 2: Algoritmo de Berlekamp-Massey.
  Dada una secuencia finita de bits $s^N = (s_0, s_1, \ldots, s_{N-1})$ permite encontrar, mediante un procedimiento iterativo, un LFSR $< L, C(D) >$ que la genera.

Ejemplo: Algoritmo A5/1, usado en GSM.



Fuente: https://es.wikipedia.org/wiki/A5/1

Fin de la sección