现代密码学

第三章 序列密码

信息与软件工程学院

现代密码学

流密码的基本概念

信息与软件工程学院

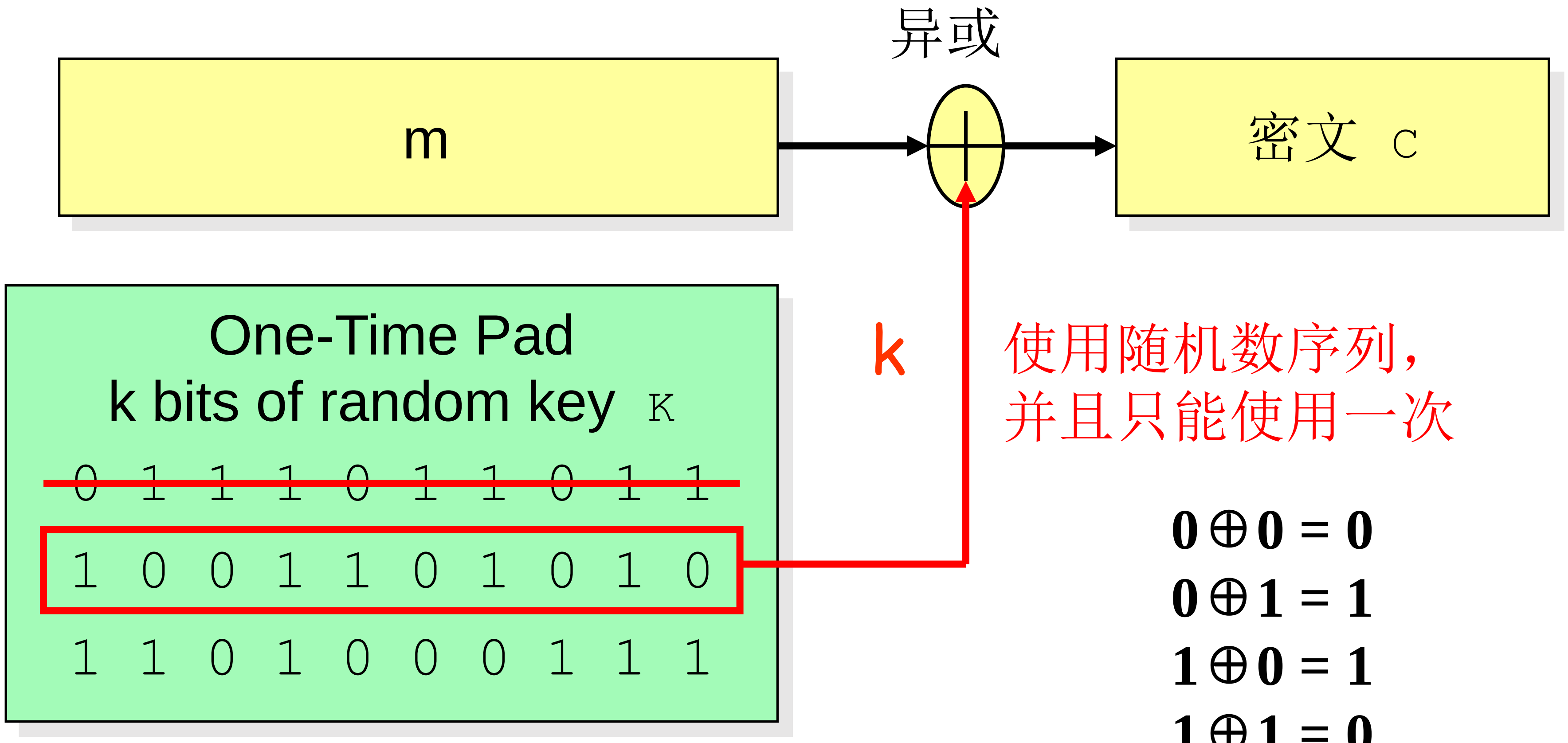
流密码的基本概念

- 一次一密
- 流密码的定义
- 同步流密码

一次一密密码

- 一种理想的加密方案，叫做一次一密密码（**one-time pad**），由**Major Joseph Mauborgne**和**AT&T**公司的**Gilbert Vernam1917**年发明的
- 明文：$x=x_0 x_1 x_2\ldots$
- 密钥：$k=k_0k_1k_2 \ldots$
- 密文：$y=y_0y_1y_2 \ldots$
- 加密函数：$y_i=x_i+k_i(\mathrm{mod}26)$
- 解密函数：$x_i=y_i-k_i(\mathrm{mod}26)$
- 注：密钥为随机产生的，而且只使用一次

一次一密密码



$$0 \oplus 0 = 0$$
$$0 \oplus 1 = 1$$
$$1 \oplus 0 = 1$$
$$1 \oplus 1 = 0$$

一次一密密码的特点

- 优点：
  - 密钥随机产生，仅使用一次
  - 无条件安全
  - 加密和解密为加法运算，效率较高
- 缺点：
  - 密钥长度至少与明文长度一样长，密钥共享困难，不太实用

流密码的基本概念

- 一次一密
- 流密码的定义
- 同步流密码

流密码概况

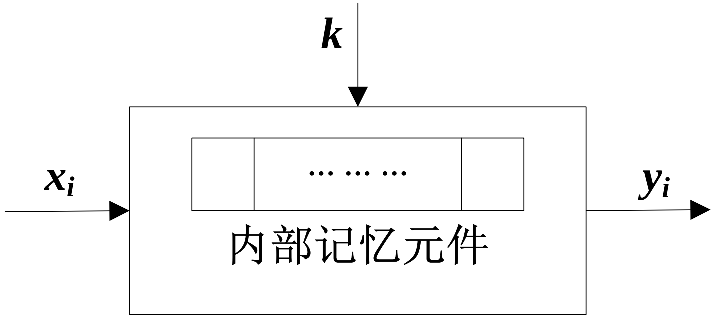
- 流密码(stream cipher)是一种重要的密码体制
  - 明文消息按字符或比特逐位加密
  - 流密码也称为序列密码(Sequence Cipher)
- 流密码在20世纪50年代得到飞跃式发展
  - 密钥流可以用移位寄存器电路来产生，也促进了线性和非线性移位寄存器发展
  - 流密码主要是基于硬件实现

流密码的基本思想

- 流密码的基本思想
  - 利用密钥k产生一个密钥流$z=z_0z_1z_2\ldots$，并使用如下规则对明文串$x=x_0x_1x_2\ldots$加密：

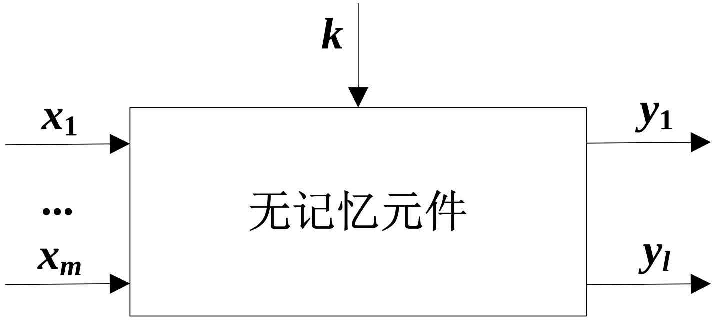
    $$y=y_0y_1y_2\ldots=E_{z_0}(x_0)E_{z_1}(x_1)\,E_{z_2}(x_2)\ldots,$$

- 密钥流
  - 由密钥流发生器f产生：$z_i=f(k,\sigma_i)$
  - σ_i是加密器中的记忆元件在时刻i的状态
  - f 是由k, σ_i产生的函数



$y_i=Ez_i(x_i)$

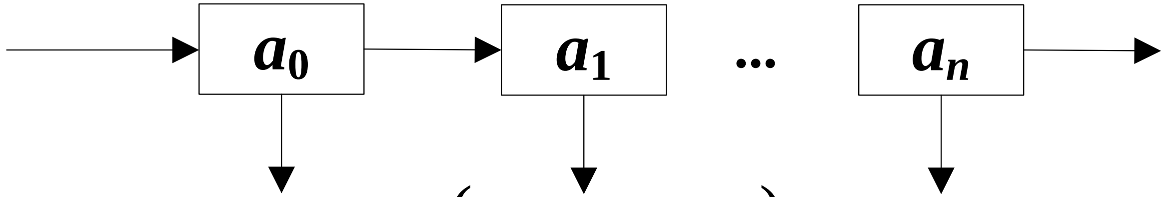
流密码



$y=E_k(x)$

分组密码

- 内部记忆元件由一组移位寄存器构成



$\sigma_i=(a_0,a_1,\cdots,a_n)$

流密码的基本概念

- 一次一密
- 流密码的定义
- 同步流密码

同步流密码

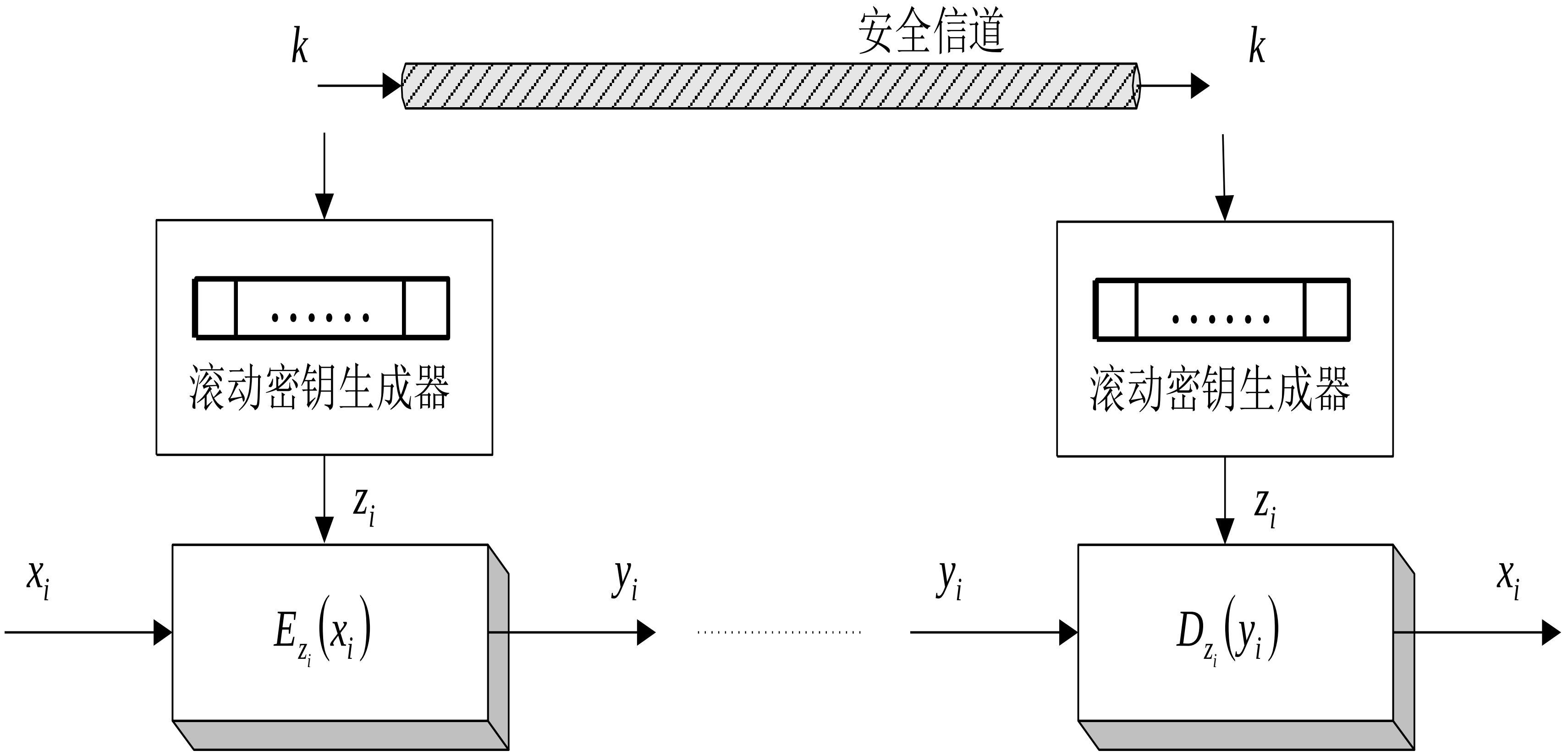
- 内部记忆元件的状态σ_i独立于明文字符的叫做同步流密码，否则叫做自同步流密码。

同步流密码

- 内部记忆元件的状态σ_i独立于明文字符的叫做同步流密码，否则叫做自同步流密码。

- 在同步流密码中，由于$z_i=f(k,\sigma_i)$与明文字符无关，因而此时密文字符$y_i=E_{z_i}(x_i)$也不依赖于此前的明文字符。因此，可将同步流密码的加密器分成密钥流产生器和加密变换器两个部分。

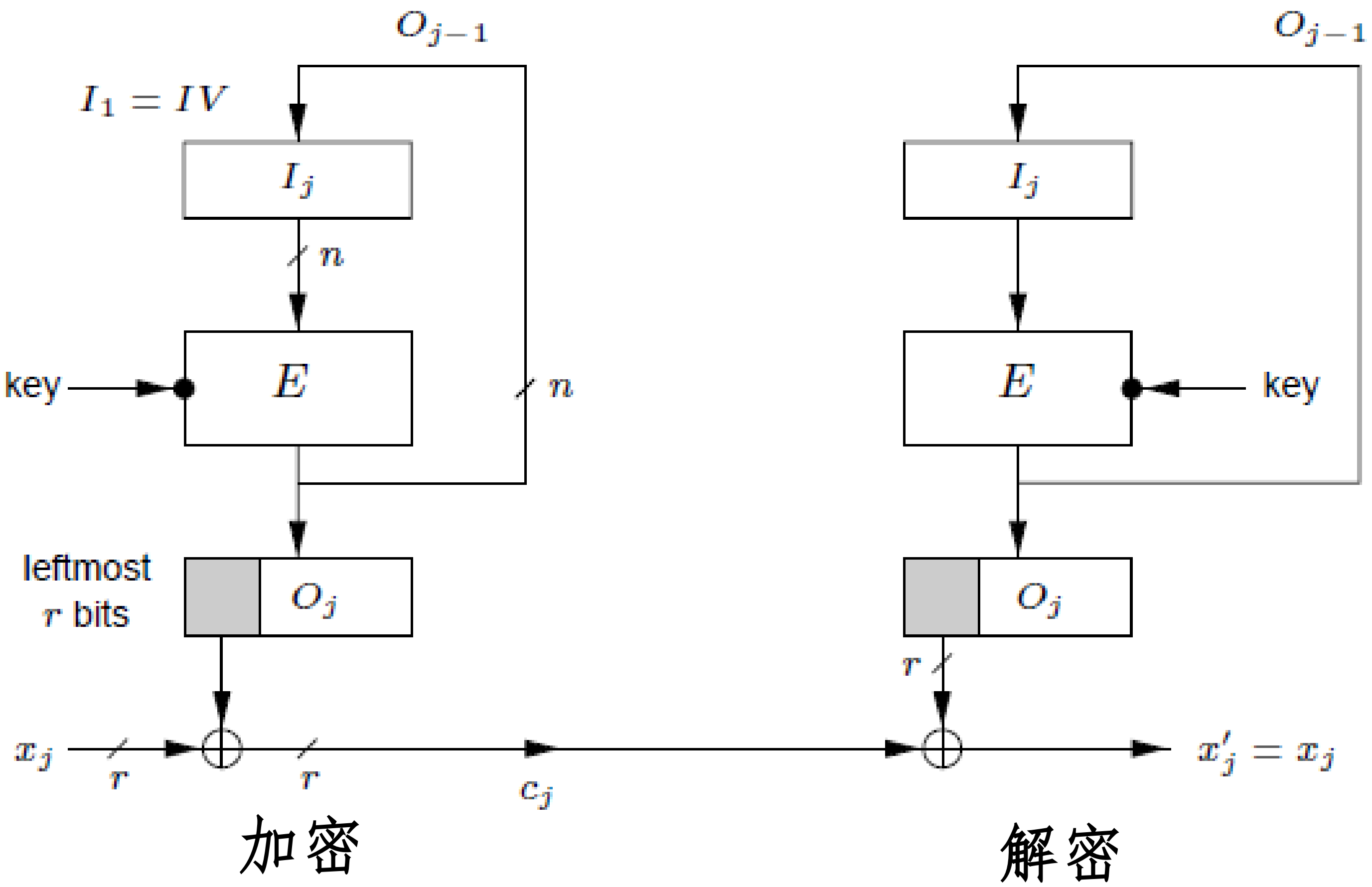
同步流密码体制模型



同步流密码体制模型

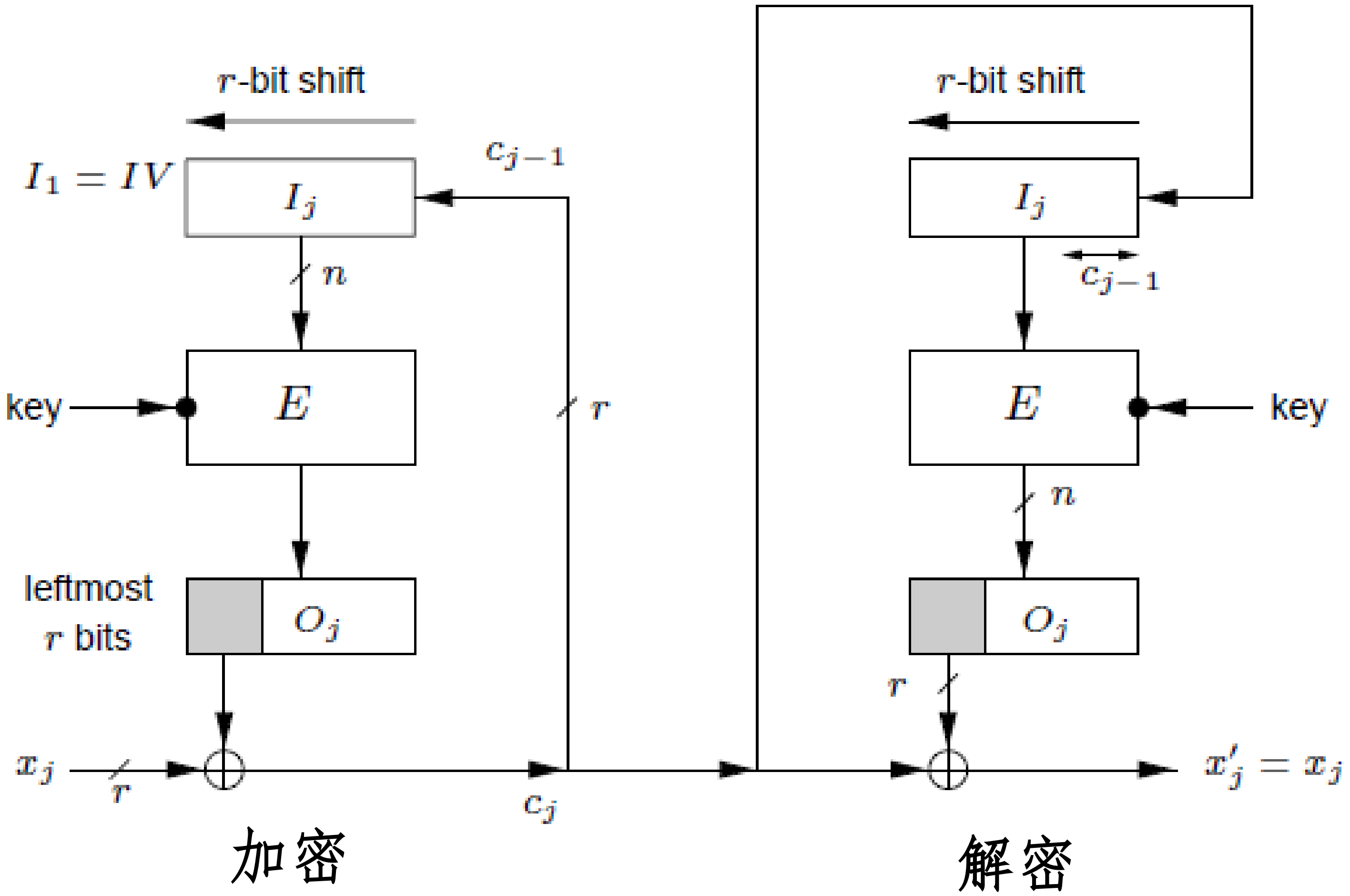
输出反馈OFB(Output Feedback)模式

- OFB模式在结构上类似于CFB模式，但反馈的内容是DES的输出而不是密文！



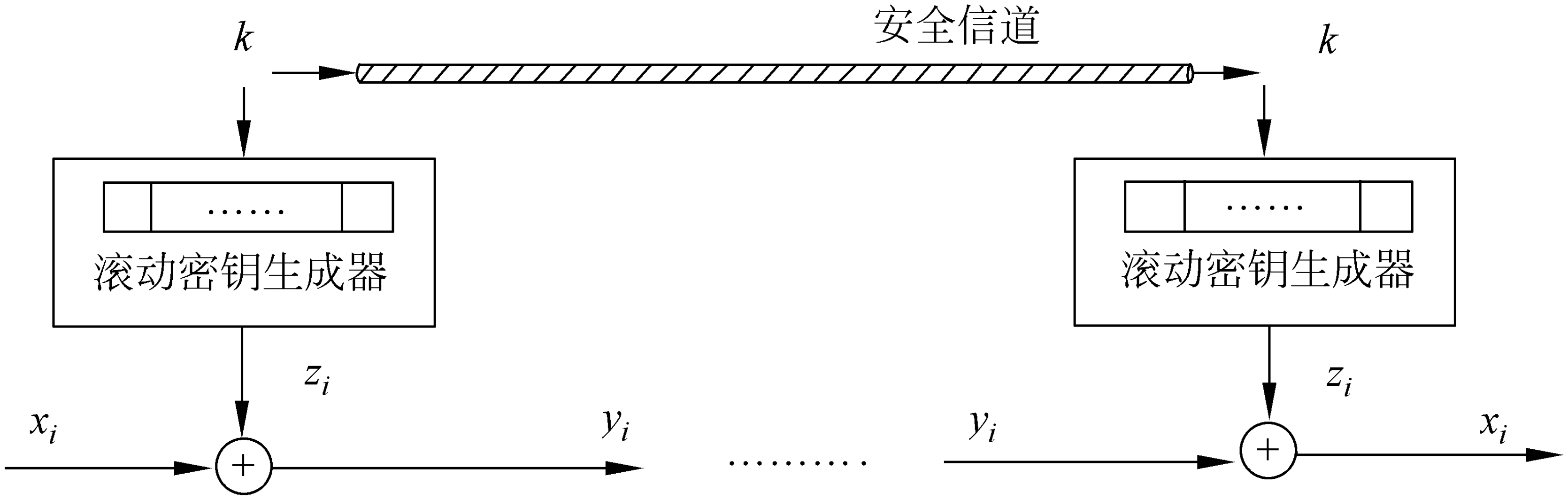
CFB的加密解密

- 若记$IV=c_{-l+1}\ldots c_{-1}\,c_0$， $|c_i|=r$，则加密过程可表示为： $c_i = x_i \oplus left_r(E_k(c_{i-l}\cdots c_{i-2}c_{i-1}))$



加法流密码体制模型

二元加法流密码是目前最为常用的流密码体制，其加密变换可表示为$y_i=z_i \oplus x_i$。



加法流密码体制模型

流密码的需求

- 一次一密密码是加法流密码的原型
  - 如果密钥用作滚动密钥流，则加法流密码就退化成一次一密密码。

流密码的需求

- 一次一密密码是加法流密码的原型
  - 如果密钥用作滚动密钥流，则加法流密码就退化成一次一密密码。
- 密码设计者的最大愿望是设计出一个滚动密钥生成器，使得密钥经其扩展成的密钥流序列具有如下性质：
  - 极大的周期
  - 良好的统计特性
  - 抗线性分析

现代密码学

二元序列的伪随机性

信息与软件工程学院

二元序列的伪随机性

- 二元序列的相关概念
- 伪随机序列

二元序列的伪随机性

- GF（2）上的一个无限序列

$$\vec{a} = (a_1, a_2, \dots, a_n, \dots)$$

称为二元序列，若 $a_i \in GF(2)$。

二元序列的伪随机性

- GF（2）上的一个无限序列

$$\vec{a} = (a_1, a_2, \ldots, a_n, \ldots)$$

  称为二元序列，若 $a_i \in GF(2)$。

- 周期：对于二元序列 $\underset{\rightarrow}{a}$，如果存在正整数 ***l***，使得对于一切正整数 ***k*** 都有

$$a_k = a_{k+l}$$

  则称 $\underset{\rightarrow}{a}$ 是周期的。

  满足上述条件的**最小正整数**称为 $\underset{\rightarrow}{a}$ 的周期记为 $p(\vec{a})$

二元序列的伪随机性

- 例：GF（2）上的一个无限序列

  100011101110001110111000111011···············

二元序列的伪随机性

- 例：GF（2）上的一个无限序列

  100011101110001110111000111011……………

- 周期：10. 当l= 20，30，40，……

  $a_k = a_{k+l}$ 任然成立

周期的性质

- 设GF（2）上的一个无限序列 $\vec{a} = (a_1, a_2, \dots, a_n, \dots)$ 是周期为 $p(\vec{a})$ 的二元序列，并设正整数l对任何非负整数k都有 $a_k = a_{k+l}$ ，则一定有

$$p(\vec{a})|l$$

- 证明：

周期的性质

- 设GF（2）上的一个无限序列 $\vec{a} = (a_1, a_2, \dots, a_n, \dots)$ 是周期为 $p(\vec{a})$ 的二元序列，并设正整数l对任何非负整数k都有 $a_k = a_{k+l}$ ，则一定有

$$p(\vec{a})|l$$

- 证明：

设 $l = qp(\underset{\to}{a}) + r$ ，其中 q, r 为正整数，且 $0 \le r < p(\underset{\to}{a})$ ，则有

$$a_k = a_{k+l}$$
$$\Rightarrow a_k = a_{qp(a)+r+k}$$
$$\Rightarrow a_k = a_{r+k}$$

又由于 $0 \le r < p(\underset{\to}{a})$ ，根据 $p(\underset{\to}{a})$ 的极小性可知 $r = 0$ ，因此 $p(\underset{\to}{a})|l$ 。

游程的定义

设$\underset{\rightarrow}{a}$是 GF（2）上周期为$p(\underset{\rightarrow}{a})$的周期序列。将$\underset{\rightarrow}{a}$的一个周期

$$(a_1,a_2,\ldots,a_{p(\underset{\rightarrow}{a})})$$

依次排列在一个圆周上使$a_{p(\underset{\rightarrow}{a})}$与$a_1$相连，把这个圆周上形如

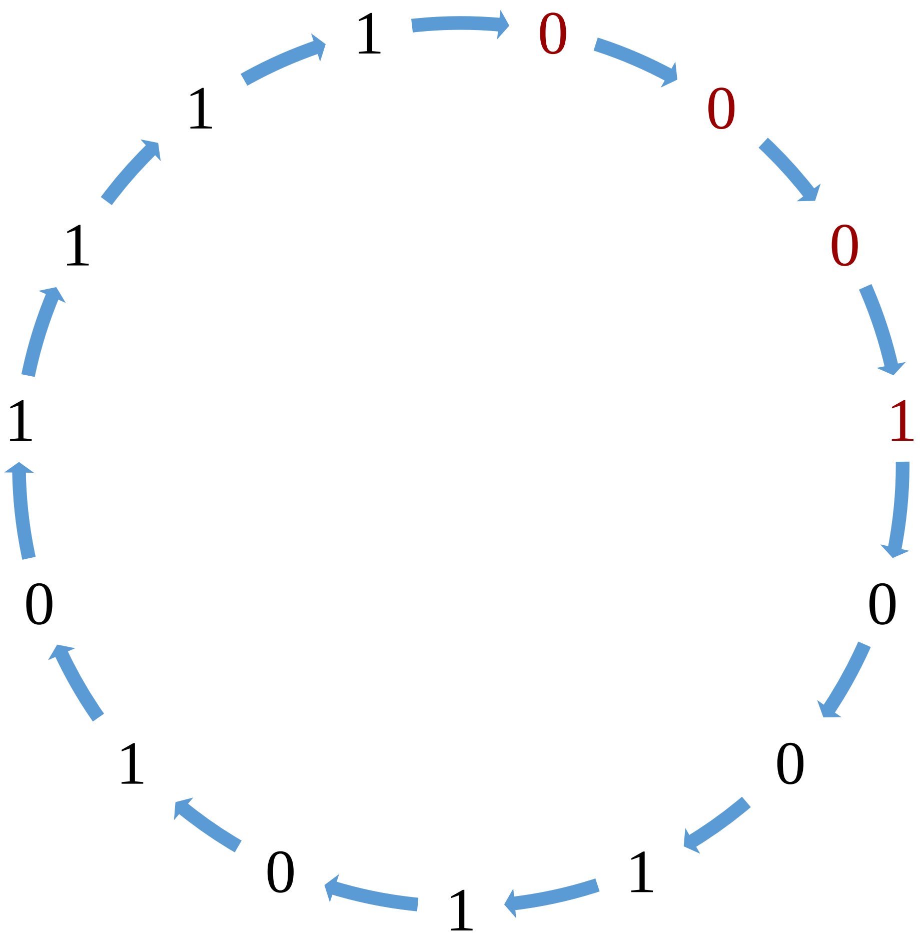
$$0\underbrace{11\cdots11}_{\text{都是1}}0 \text{ 或 } 1\underbrace{00\cdots00}_{\text{都是0}}1$$

的一连串两两相邻的项分别称为$\underset{\rightarrow}{a}$的一个周期中一个 1 游程或一个 0 游程。而 1 游程中 1 的个数或 0 游程中 0 的个数称为游程的长度。

游程的例子

周期为15的二元序列
100010011010111

011110为1的4游程
10001为0的3游程



自相关函数

GF(2)上周期为T的序列{a_i}的自相关函数定义为

$$R(\tau)=(1/T)\sum_{k=1}^{T}(-1)^{a_k}(-1)^{a_{k+\tau}} \quad 0\le\tau\le T-1$$

当τ=0时，R(τ)=1；当τ≠0时，称R(τ)为异相自相关函数。

二元序列的随机性

- 二元序列的相关概念
- 伪随机序列

Golomb伪随机公设

3个随机性公设：

① 在序列的一个周期内，0与1的个数相差至多为1。

- 说明$\{a_i\}$中0与1出现的概率基本上相同

Golomb伪随机公设

3个随机性公设：

① 在序列的一个周期内，0与1的个数相差至多为1。

- 说明$\{a_i\}$中0与1出现的概率基本上相同

② 在序列的一个周期内，长为i的游程占游程总数的$1/2^i$ $(i=1,2,\ldots)$，且在等长的游程中0的游程个数和1的游程个数相等。

- 说明0与1在序列中每一位置上出现的概率相同

Golomb伪随机公设

3个随机性公设：

① 在序列的一个周期内，0与1的个数相差至多为1。

- 说明$\{a_i\}$中0与1出现的概率基本上相同

② 在序列的一个周期内，长为i的游程占游程总数的$1/2^i$ (i=1,2,…)，且在等长的游程中0的游程个数和1的游程个数相等。

- 说明0与1在序列中每一位置上出现的概率相同

③ 异相自相关函数是一个常数。

- 意味着通过对序列与其平移后的序列做比较，不能给出其他任何信息

伪随机序列的定义

设 $\underset{\rightarrow}{a}=(a_1,a_2,\ldots,a_{p(\underset{\rightarrow}{a})},\ldots)$ 是 GF（2）上一个周期等于 $p(\underset{\rightarrow}{a})$ 的周期序列。

如果对于一切 $t\not\equiv 0(\mathrm{mod}\ p(\underset{\rightarrow}{a}))$，有

$$R(t)=-1$$

$$R(t)=\sum_{k=1}^{T}(-1)^{a_k}(-1)^{a_{k+t}},0\le t\le T-1$$

则称序列 $\underset{\rightarrow}{a}=(a_1,a_2,\ldots,a_{p(\underset{\rightarrow}{a})},\ldots)$ 为伪随机序列。

- 可以证明上述定义满足Golomb三个伪随机公设，详情参考
- 万哲先著。代数和编码（第三版）。高等教育出版社，2007.

伪随机序列还应满足的条件

C1．周期p要足够大，如大于10^{50}；

C2．序列$\{a_i\}_{i\geq 1}$产生易于高速生成；

C3．当序列$\{a_i\}_{i\geq 1}$的任何部分暴露时，要分析整个序列，提取产生它的电路结构信息， 在计算上是不可行的，称此为不可预测性。

C3决定了密码的强度，是流密码理论的核心。它包含了流密码要研究的许多主要问题，如线性复杂度、相关免疫性、不可预测性等等。

现代密码学

线性反馈移位寄存器

信息与软件工程学院

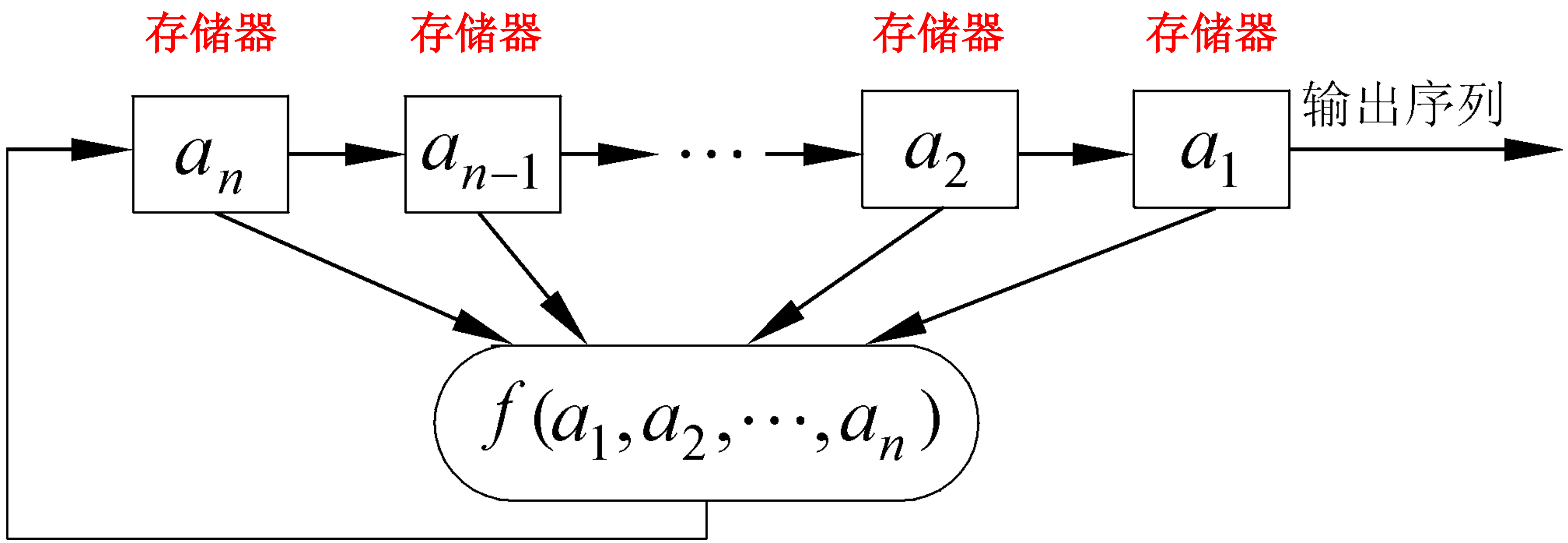
线性反馈移位寄存器

- 反馈移位寄存器
- 线性反馈移位寄存器

反馈移位寄存器

移位寄存器是流密码产生密钥流的一个主要组成部分。

GF(2)上一个n级反馈移位寄存器由n个二元存储器与一个反馈函数 $f(a_1,a_2,\ldots,a_n)$组成，如下图所示。



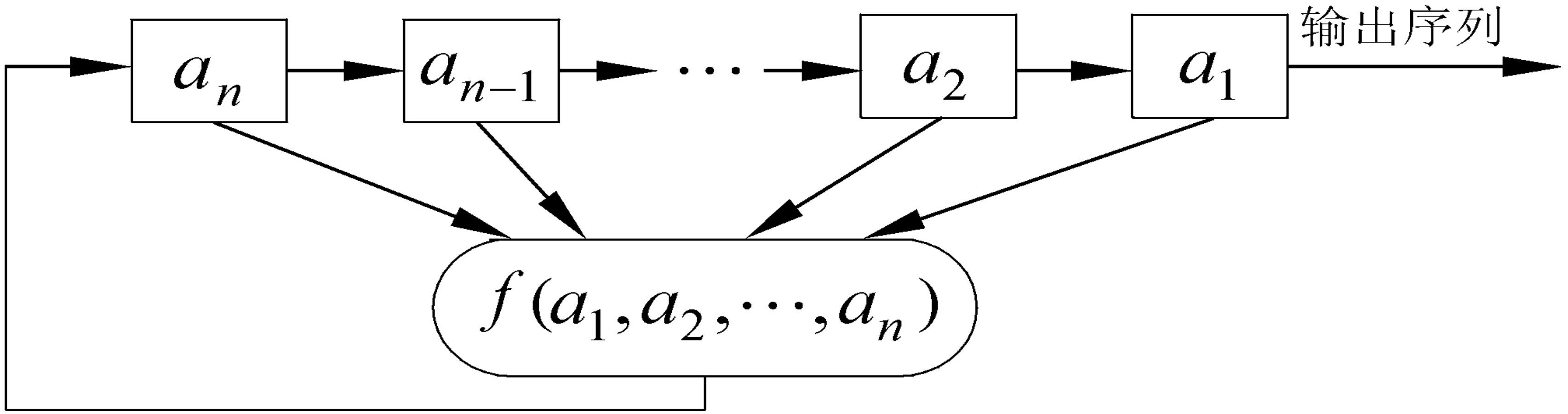
反馈移位寄存器的状态

在任一时刻，这些级的内容构成该反馈移位寄存器的状态，每一状态对应于**GF(2)**上的一个**n**维向量，共有2^n种可能的状态。

每一时刻的状态可用**n**维向量

$$(a_1,a_2,\ldots,a_n)$$

表示，其中a_i是第**i**级存储器的内容。

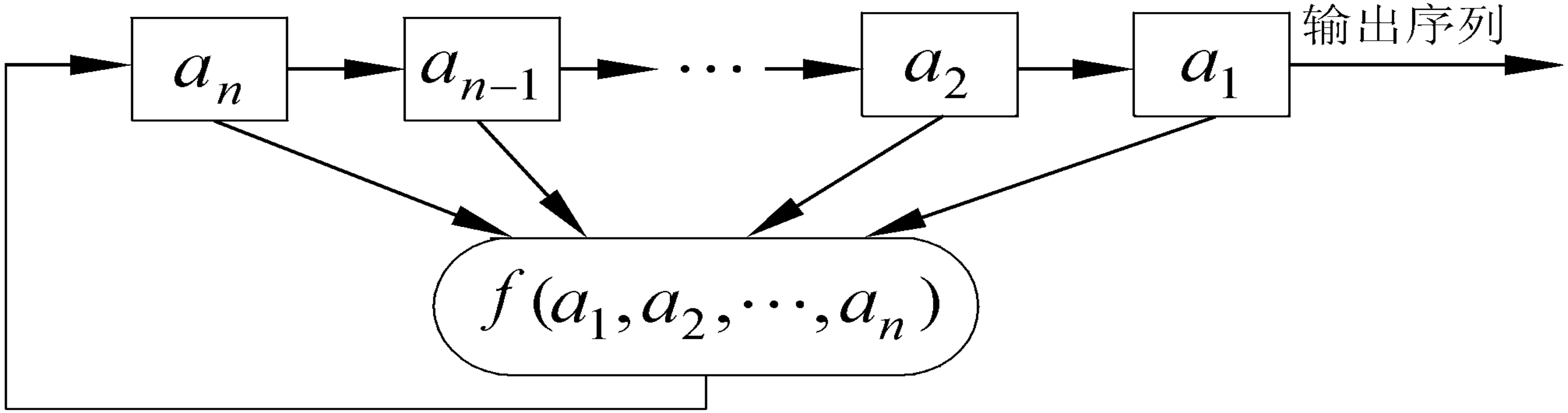


反馈函数

初始状态由用户确定。

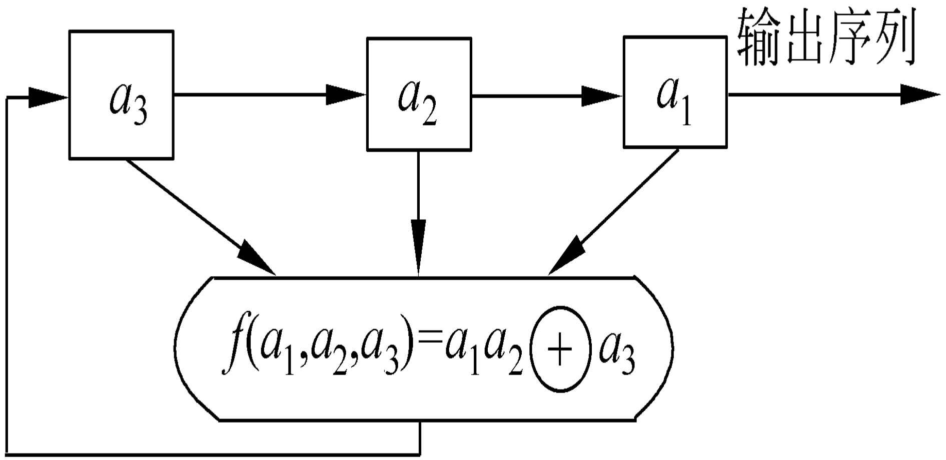
反馈函数$f(a_1,a_2,\ldots,a_n)$是n元布尔函数，即函数的自变量和因变量只取0和1这两个可能的值。

函数中的运算有逻辑与、逻辑或、逻辑补等运算。



反馈移位寄存器的例子

如图是一个3级反馈移位寄存器，其初始状态为$(a_1,a_2,a_3)=(1,0,1)$，输出可由右表给出。



一个3级反馈移位寄存器

即输出序列为101110111011…，周期为4。

一个3级反馈移位寄存器的状态和输出

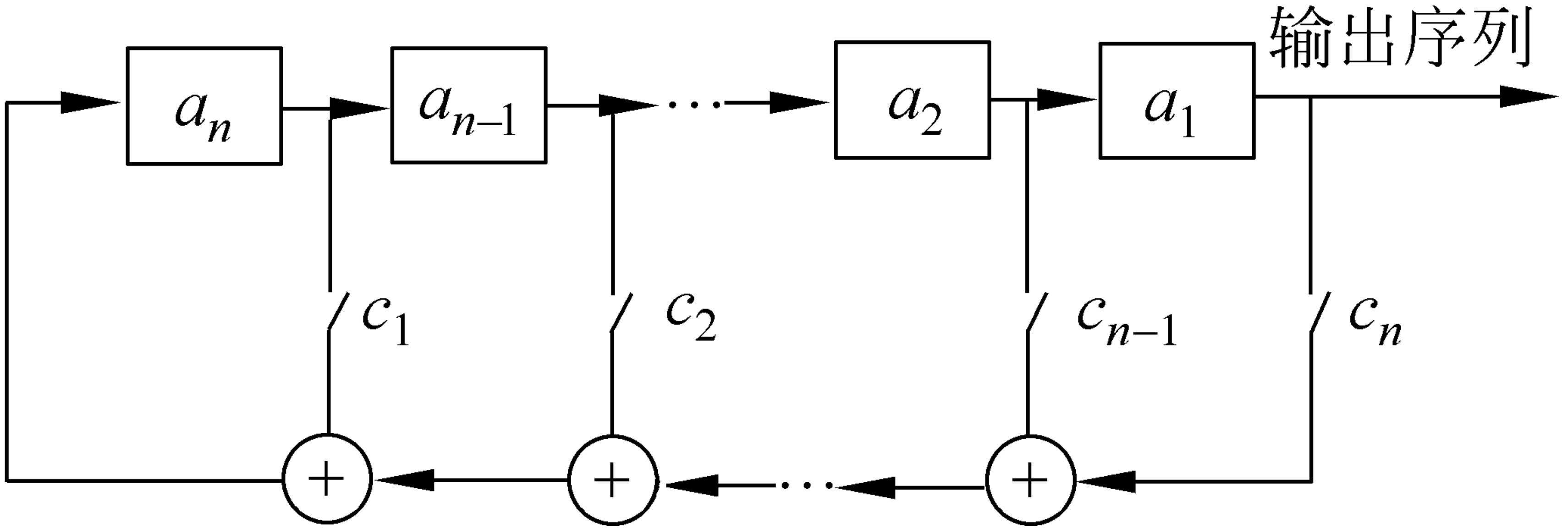
状态 (a_3,a_2,a_1)	输出
1 0 1	1
1 1 0	0
1 1 1	1
0 1 1	1
1 0 1	1
1 1 0	0

线性反馈移位寄存器

- 反馈移位寄存器
- 线性反馈移位寄存器

线性反馈移位寄存器LFSR（linear feedback shift register）

GF(2)上的n级线性反馈移位寄存器



$$f(a_1, a_2, \cdots, a_n) = c_1 a_n \oplus c_2 a_{n-1} \oplus \cdots \oplus c_n a_1$$

LFSR的反馈函数

输出序列$\{a_t\}$满足：

$$f(a_1,a_2,\cdots,a_n)=c_1a_n\oplus c_2a_{n-1}\oplus\cdots\oplus c_na_1$$

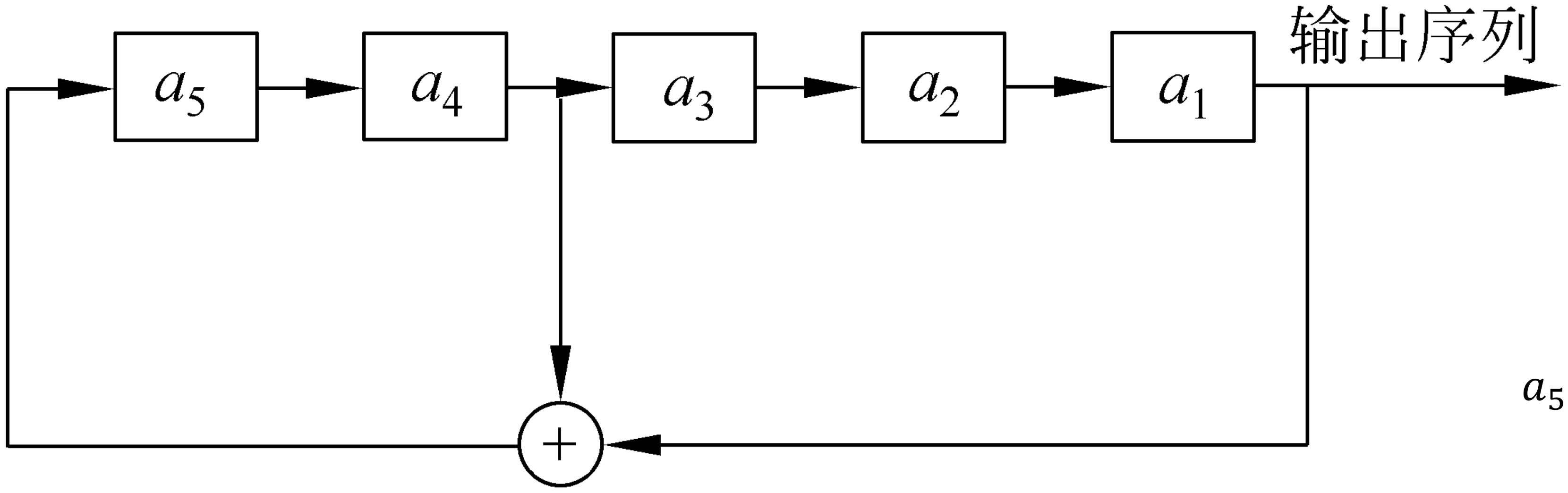
$$a_{n+1}=c_1a_n\oplus c_2a_{n-1}\oplus\cdots\oplus c_na_1$$
$$a_{n+2}=c_1a_{n+1}\oplus c_2a_n\oplus\cdots\oplus c_na_2$$
$$\cdots\cdots\cdots$$

$$a_{n+t}=c_1a_{n+t-1}\oplus c_2a_{n+t-2}\oplus\cdots\oplus c_na_t,t=1,2,\cdots$$

线性反馈移位寄存器：实现简单、速度快、有较为成熟的理论，成为构造**密钥流生成器的最重要的部件之一**。

LFER的实例

例 下图是一个5级线性反馈移位寄存器，其初始状态为（a_1, a_2, a_3, a_4, a_5）=(1,0,0,1,1)



反馈函数

$$a_{5+t} = a_{t+3} \oplus a_t, t = 1,2,\cdots$$

可求出输出序列为

<u>1001101001000010101110110001111</u>100110…

周期为31。

密钥流的周期

- 给定密钥流 $\{a_i\} = a_1, a_2, a_3, \cdots, a_n, \cdots$，如果存在整数r，使得对于任意 a_i，都有 $a_{i+r} = a_i$，则称 r为该密钥流的一个周期，称满足 $a_{i+r} = a_i$ 的最小正整数为该密钥流的最小周期或简称周期。

LFSR的性质

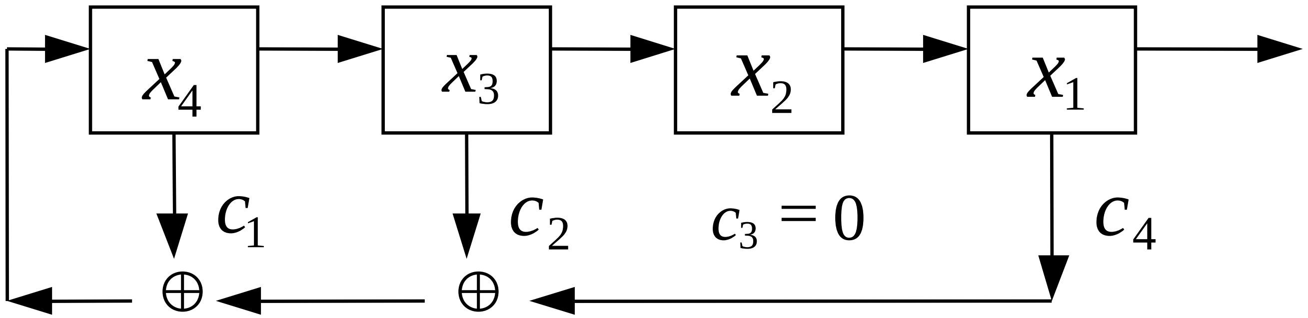
总是假定$c_1, c_2, \ldots, c_n$中至少有一个不为0，否则$f(a_1, a_2, \ldots, a_n) \equiv 0$。

总是假定$c_n=1$。

- LFSR输出序列的性质：完全由其反馈函数决定。
- n级LFSR状态数：最多有2^n个
- n级LFSR的状态周期：$\leq 2^n-1$
- 输出序列的周期=状态周期，$\leq 2^n-1$

- 选择合适的反馈函数可使序列的周期达到最大值2^n-1，周期达到最大值的序列称为m序列。

特征函数决定了周期只能是7的因子



反馈函数为： $f(x_1,x_2,x_3,x_4)=x_1+x_3+x_4$

称为结构常数： $[c_1,c_2,c_3,c_4]=[1,1,0,1]$

初始状态:(1000)，输出序列:$(1000110)^\infty$，周期为7

初始状态:(0010)，输出序列:$(0010111)^\infty$，周期为7

初始状态:(0110)，输出序列:$(0110100)^\infty$，周期为7

初始状态:(1111)，输出序列:$(1111111)^\infty$，周期为1

LFSR的性质

总是假定$c_1, c_2, \ldots, c_n$中至少有一个不为0，否则$f(a_1, a_2, \ldots, a_n) \equiv 0$。

总是假定$c_n=1$。

- LFSR输出序列的性质：完全由其反馈函数决定。
- n级LFSR状态数：最多有2^n个（向量$(a_1, a_2, \ldots, a_n)$个数）
- n级LFSR的状态周期：$\leq 2^n-1$
- 输出序列的周期=状态周期，$\leq 2^n-1$

- 选择合适的反馈函数可使序列的周期达到最大值2^n-1，周期达到最大值的序列称为m序列。

现代密码学

m-序列

信息与软件工程学院

线性移位寄存器的一元多项式表示

定义3.1 设n级线性移位寄存器的输出序列满足递推关系

$$a_{n+k}=c_1a_{n+k-1}\oplus c_2a_{n+k-2}\ \oplus\ \dots\ \oplus\ c_na_k \qquad (*)$$

用延迟算子 D($Da_k=a_{k-1}$) 作为未定元，给出的反馈多项式为：

$$p(D)=1+c_1D+\dots+c_{n-1}D^{n-1}+c_nD^n$$

$$\text{（即 } p(D)\ (a_{n+k}) = (1+c_1D+\dots+c_{n-1}D^{n-1}+c_nD^n)(a_{n+k})\text{）}$$

这种递推关系可用一个一元高次多项式

$$p(x)=1+c_1x+\dots+c_{n-1}x^{n-1}+c_nx^n$$

表示，称这个多项式为LFSR的特征多项式。

关于特征多项式的解释

$$a_{n+k} = c_1 a_{n+k-1} \oplus c_2 a_{n+k-2} \oplus \cdots \oplus c_n a_k$$
$$\Leftrightarrow a_{n+k} \oplus c_1 a_{n+k-1} \oplus c_2 a_{n+k-2} \oplus \cdots \oplus c_n a_k = 0$$

$D(a_k) = a_{k-1}$，用 $p(D) = 1 + c_1 D + \cdots + c_n D^n$ 作用于 a_{n+k} 后恰好就是上式的左边，即

$$\begin{aligned} p(D)(a_{n+k}) &= (1 + c_1 D + \cdots + c_n D^n)(a_{n+k}) \\ &= a_{n+k} + c_1 D(a_{n+k}) + \cdots + c_n D^n(a_{n+k}) \\ &= a_{n+k} + c_1 a_{n+k-1} + c_2 a_{n+k-2} + \cdots + c_n a_k \end{aligned}$$

生成函数

定义3.2 给定序列{a_i}，幂级数

$$A(x)=\sum_{i=1}^{\infty} a_i x^{i-1}$$

称为该序列的生成函数。

定义G(p(x))

根据初始状态的不同，由递推关系(*)生成的非恒零的序列有2^n-1个，记这2^n-1个非零序列的全体为G(p(x))。

- 定理3.1 设$p(x)=1+c_1x+\ldots+c_{n-1}x^{n-1}+c_nx^n$是GF(2)上的多项式，G(p(x))中任一序列$\{a_i\}$的生成函数A(x)满足：

$$A(x)=\frac{\varphi(x)}{p(x)}$$

- 其中

$$\varphi(x)=\sum_{i=1}^{n}(c_{n-i}x^{n-i}\sum_{j=1}^{i}a_jx^{j-1})$$

定理3.1的证明

证明： 在等式

$$a_{n+1}=c_1a_n\oplus c_2a_{n-1}\oplus\ldots\oplus c_na_1$$

$$a_{n+2}=c_1a_{n+1}\oplus c_2a_n\oplus\ldots\oplus c_na_2$$

$$\ldots$$

定理3.1的证明

证明： 在等式

$$a_{n+1}=c_1a_n \oplus c_2a_{n-1} \oplus \ldots \oplus c_na_1$$

$$a_{n+2}=c_1a_{n+1} \oplus c_2a_n \oplus \ldots \oplus c_na_2$$

$$\ldots$$

两边分别乘以$x^n, x^{n+1}, \ldots$，再求和，可得

$$A(x)-(a_1+a_2x+\ldots+a_nx^{n-1})$$

$$=c_1x[A(x)-(a_1+a_2x+\ldots+a_{n-1}x^{n-2})]$$

$$+c_2x^2[A(x)-(a_1+a_2x+\ldots+a_{n-2}x^{n-3})]+\ldots+c_nx^nA(x)$$

定理3.1的证明

证明： 在等式

$$a_{n+1}=c_1a_n\oplus c_2a_{n-1}\oplus\dots\oplus c_na_1$$

$$a_{n+2}=c_1a_{n+1}\oplus c_2a_n\ \oplus\dots\oplus c_na_2$$

$$\dots$$

两边分别乘以$x^n,x^{n+1},\dots$，再求和，可得

$$A(x)-(a_1+a_2x+\dots+a_nx^{n-1})$$

$$=c_1x[A(x)-(a_1+a_2x+\dots+a_{n-1}x^{n-2})]$$

$$+c_2x^2[A(x)-(a_1+a_2x+\dots+a_{n-2}x^{n-3})]+\dots+c_nx^nA(x)$$

移项整理

$$(1+c_1x+\dots+c_{n-1}x^{n-1}+c_nx^n)A(x)$$

$$=(a_1+a_2x+\dots+a_nx^{n-1})+c_1x(a_1+a_2x+\dots+a_{n-1}x^{n-2})$$

$$+c_2x^2(a_1+a_2x+\dots+a_{n-2}x^{n-3})+\dots+c_{n-1}x^{n-1}a_1$$

定理3.1的证明

证明： 在等式

$$a_{n+1}=c_1a_n\oplus c_2a_{n-1}\oplus\ldots\oplus c_na_1$$

$$a_{n+2}=c_1a_{n+1}\oplus c_2a_n\oplus\ldots\oplus c_na_2$$

…

两边分别乘以$x^n,x^{n+1},\ldots$，再求和，可得

$$A(x)-(a_1+a_2x+\ldots+a_nx^{n-1})$$
$$=c_1x[A(x)-(a_1+a_2x+\ldots+a_{n-1}x^{n-2})]$$
$$+c_2x^2[A(x)-(a_1+a_2x+\ldots+a_{n-2}x^{n-3})]+\ldots+c_nx^nA(x)$$

移项整理

$$(1+c_1x+\ldots+c_{n-1}x^{n-1}+c_nx^n)A(x)$$
$$=(a_1+a_2x+\ldots+a_nx^{n-1})+c_1x(a_1+a_2x+\ldots+a_{n-1}x^{n-2})$$
$$+c_2x^2(a_1+a_2x+\ldots+a_{n-2}x^{n-3})+\ldots+c_{n-1}x^{n-1}a_1$$

即

$$p(x)A(x)=\sum_{i=1}^{n}(c_{n-i}x^{n-i}\sum_{j=1}^{i}a_jx^{j-1})$$
$$=\varphi(x)$$

即 $$A(x)=\frac{\varphi(x)}{p(x)}$$

由此也可以清楚地看出

$$deg(\varphi)\leq n-1$$

- ※ φ(x)的次数最大可能是n-1。

- 因此φ(x)表达式共有n个系数，从而有2^n-1个不同的非0表达式，并且由初始状态a_1，$a_2,\cdots,a_n$完全确定。
- 这样每一个初始状态对应一条不同的序列，而每一条序列的生成函数A(x)唯一确定一个φ(x)，共有2^n-1个初始状态
- 所以初始状态、φ(x)、G(p(x))中的序列$\{a_i\}$三者之间一一对应

一些定理和定义

根据初始状态的不同，由递推关系(*)生成的非恒零的序列有2^n-1个，记这2^n-1个非零序列的全体为G(p(x))。

定理3.2 $p(x)|q(x)$的充要条件是$G(p(x))\subseteq G(q(x))$。

——该定理说明：可用n级LFSR产生的序列，也可用级数更多的LFSR来产生。

定义3.3 设p(x)是GF(2)上的多项式，使$p(x)|(x^p-1)$成立的最小正整数p称为p(x)的周期或阶。

定理3.3 若序列$\{a_i\}$的特征多项式$p(x)$定义在$GF(2)$上，p是$p(x)$的周期，则$\{a_i\}$的周期$r|p$。

—— 该定理说明：n级LFSR输出序列的周期r，不依赖于初始条件，而依赖于特征多项式p(x)。

一些定理和定义

根据初始状态的不同，由递推关系(*)生成的非恒零的序列有2^n-1个，记这2^n-1个非零序列的全体为G(p(x))。

定理3.2 $p(x)|q(x)$的充要条件是$G(p(x)) \subseteq G(q(x))$。

——该定理说明：可用n级LFSR产生的序列，也可用级数更多的LFSR来产生。

定义3.3 设p(x)是GF(2)上的多项式，使p(x)|(x^p-1)成立的最小正整数p称为p(x)的周期或阶。

定理3.3 若序列$\{a_i\}$的特征多项式$p(x)$定义在$GF(2)$上，p是$p(x)$的周期，则$\{a_i\}$的周期$r|p$。

—— 该定理说明：n级LFSR输出序列的周期r，不依赖于初始条件，而依赖于特征多项式p(x)。

一些定理和定义

根据初始状态的不同，由递推关系(*)生成的非恒零的序列有2^n-1个，记这2^n-1个非零序列的全体为G(p(x))。

定理3.2 $p(x)|q(x)$的充要条件是$G(p(x)) \subseteq G(q(x))$。

——该定理说明：可用n级LFSR产生的序列，也可用级数更多的LFSR来产生。

定义3.3 设p(x)是GF(2)上的多项式，使$p(x)|(x^p-1)$成立的最小正整数p称为p(x)的周期或阶。

定理3.3 若序列$\{a_i\}$的特征多项式$p(x)$定义在$GF(2)$上，p是$p(x)$的周期，则$\{a_i\}$的周期$r|p$。

—— 该定理说明：n级LFSR输出序列的周期r，不依赖于初始条件，而依赖于特征多项式p(x)。

定理3.3 若序列$\{a_i\}$的特征多项式$p(x)$定义在$GF(2)$上，p是$p(x)$的周期，则$\{a_i\}$的周期$r|p$。

证明：由p(x)周期的定义得$p(x)|(x^p-1)$

$\Rightarrow$ 存在q(x)，使得$x^p-1=p(x)q(x)$

定理3.3 若序列$\{a_i\}$的特征多项式$p(x)$定义在$GF(2)$上，p是$p(x)$的周期，则$\{a_i\}$的周期$r|p$。

证明：由p(x)周期的定义得$p(x)|(x^p-1)$

$\Rightarrow$ 存在q(x)，使得$x^p-1=p(x)q(x)$

又由$p(x)A(x)=\varphi(x)$

$\Rightarrow p(x)q(x)A(x)=\varphi(x)q(x)$

$\Rightarrow (x^p-1)A(x)=\varphi(x)q(x)$

定理3.3 若序列$\{a_i\}$的特征多项式$p(x)$定义在$GF(2)$上，p是$p(x)$的周期，则$\{a_i\}$的周期$r\mid p$。

证明：由p(x)周期的定义得$p(x)|(x^p-1)$

$\Rightarrow$ 存在q(x)，使得$x^p-1=p(x)q(x)$

又由$p(x)A(x)=\varphi(x)$

$\Rightarrow p(x)q(x)A(x)=\varphi(x)q(x)$

$\Rightarrow (x^p-1)A(x)=\varphi(x)q(x)$

由于q(x)的次数为p-n，$\varphi(x)$的次数不超过n-1

$\Rightarrow (x^p-1)A(x)$的次数不超过$(p-n)+(n-1)=p-1$

$\Rightarrow$ 对于任意正整数i都有$a_{i+p}=a_i$

定理3.3 若序列$\{a_i\}$的特征多项式$p(x)$定义在$GF(2)$上，p是$p(x)$的周期，则$\{a_i\}$的周期$r|p$。

证明：由p(x)周期的定义得$p(x)|(x^p-1)$

$\Rightarrow$ 存在q(x)，使得$x^p-1=p(x)q(x)$

又由$p(x)A(x)=\varphi(x)$

$\Rightarrow$ $p(x)q(x)A(x)=\varphi(x)q(x)$

$\Rightarrow$ $(x^p-1)A(x)=\varphi(x)q(x)$

由于q(x)的次数为p-n，$\varphi(x)$的次数不超过n-1

$\Rightarrow$ $(x^p-1)A(x)$的次数不超过$(p-n)+(n-1)=p-1$

$\Rightarrow$ 对于任意正整数i都有$a_{i+p}=a_i$

设$p=kr+t, 0\le t<r$，则$a_{i+p}=a_{i+kr+t}=a_{i+t}=a_i$，所以t=0，即r|p。（证毕）

不可约多项式

定理3.4 设p(x)是n次不可约多项式，周期为m，序列$\{a_i\}\in G(p(x))$，则$\{a_i\}$的周期为m。

证明：设$\{a_i\}$的周期为r，由定理3.3有$r|m$，所以$r\le m$。

不可约多项式

定理3.4 设p(x)是n次不可约多项式，周期为m，序列$\{a_i\}\in G(p(x))$，则$\{a_i\}$的周期为m。

证明：设$\{a_i\}$的周期为r，由定理3.3有r|m，所以$r\leq m$。

设A(x)为$\{a_i\}$的生成函数，$A(x)=\varphi(x)/p(x)$，即$p(x)A(x)=\varphi(x)\neq 0$，$\varphi(x)$的次数不超过n-1。而

不可约多项式

定理3.4 设p(x)是n次不可约多项式，周期为m，序列$\{a_i\}\in G(p(x))$，则$\{a_i\}$的周期为m。

证明：设$\{a_i\}$的周期为r，由定理3.3有r|m，所以r≤m。

设A(x)为$\{a_i\}$的生成函数，A(x)= φ(x)/p(x)，即p(x)A(x)= φ(x)≠0， φ (x)的次数不超过n-1。而

$$A(x)=\sum a_i x^{i-1}=a_1+a_2x+\ldots+a_rx^{r-1}+x^r(a_1+a_2x+\ldots+a_rx^{r-1})$$

$$+(x^r)^2(a_1+a_2x+\ldots+a_rx^{r-1})+\ldots$$

不可约多项式

定理3.4 设p(x)是n次不可约多项式，周期为m，序列$\{a_i\}\in G(p(x))$，则$\{a_i\}$的周期为m。

证明：设$\{a_i\}$的周期为r，由定理3.3有r|m，所以$r\le m$。

设A(x)为$\{a_i\}$的生成函数，$A(x)=\varphi(x)/p(x)$，即$p(x)A(x)=\varphi(x)\neq 0$， $\varphi(x)$的次数不超过n-1。而

$$A(x)=\sum a_i x^{i-1}=a_1+a_2x+\ldots+a_rx^{r-1}+x^r(a_1+a_2x+\ldots+a_rx^{r-1})$$

$$+(x^r)^2(a_1+a_2x+\ldots+a_rx^{r-1})+\ldots$$

$$=(a_1+a_2x+\ldots+a_rx^{r-1})/(1-x^r)$$

$$=(a_1+a_2x+\ldots+a_rx^{r-1})/(x^r-1)$$

$p(x)$是不可约多项式

$$A(x)=\frac{a_1+a_2x+\cdots+a_rx^{r-1}}{x^r-1}=\frac{\varphi(x)}{p(x)}$$

$$p(x)(a_1+a_2x+\cdots+a_rx^{r-1})=\varphi(x)(x^r-1)$$

$$p(x)|\varphi(x)(x^r-1)$$

$$\gcd(p(x),\varphi(x))=1$$

$p(x)|(x^r-1)$, 因此$m\le r$

故 $m=r$

总结-1

- 定理3.1 设$p(x)=1+c_1x+\ldots+c_{n-1}x^{n-1}+c_nx^n$是GF(2)上的多项式，G(p(x))中任一序列$\{a_i\}$的生成函数A(x)满足：

$$A(x)=\frac{\varphi(x)}{p(x)}$$

- 其中

$$\varphi(x)=\sum_{i=1}^{n}(c_{n-i}x^{n-i}\sum_{j=1}^{i}a_jx^{j-1})$$

- 生成函数与特征多项式的关系

总结-2

定理3.2 $p(x)|q(x)$的充要条件是$G(p(x)) \subseteq G(q(x))$。

不同特征多项式的关系，与它们所确定的序列集的关系

定理3.3 若序列$\{a_i\}$的特征多项式$p(x)$定义在$GF(2)$上，p是$p(x)$的周期，则$\{a_i\}$的周期$r|p$。

特征多项式决定了它的阶（周期），序列的周期与特征多项式的阶的关系

定理3.4 设p(x)是n次不可约多项式，周期为m，序列$\{a_i\} \in G(p(x))$，则$\{a_i\}$的周期为m。

特殊的特征多项式（不可约），序列的周期与特征多项式的阶的关系

m-序列产生的必要条件

定理3.5 n级LFSR产生的序列有最大周期2^n-1的必要条件是其特征多项式为不可约的。

m-序列产生的必要条件

定理3.5 n级LFSR产生的序列有最大周期2^n-1的必要条件是其特征多项式为不可约的。

结论：**n级LFSR产生的序列周期达到最大2^n-1，其特征多项式为p(x)，则G(p(x))中所有的序列的周期都是2^n-1。**

m-序列产生的必要条件

定理3.5 n级LFSR产生的序列有最大周期2^n-1的必要条件是其特征多项式为不可约的。

证明：设n级LFSR产生的序列周期达到最大2^n-1。

反证法：设特征多项式为p(x)，若p(x)可约，可设为p(x)=g(x)h(x)，其中g(x)不可约，且次数k<n。由于$G(g(x)) \subset G(p(x))$，而G(g(x))中序列的周期一方面不超过2^k-1，另一方面又等于2^n-1，这是矛盾的，所以p(x)不可约。

该定理的逆不成立，即LFSR的特征多项式为不可约多项式时，其输出序列不一定是m序列。

定理3.5 的反例

例3.4 $f(x)=x^4+x^3+x^2+x+1$为GF(2)上的不可约多项式，这可由$x, x+1, x^2+x+1$都不能整除$f(x)$得到。以$f(x)$为特征多项式的LFSR的输出序列可由

$$a_k=a_{k-1}\oplus a_{k-2}\oplus a_{k-3}\oplus a_{k-4}\quad(k\geq 4)$$

和给定的初始状态求出，设初始状态为0001，则输出序列为000110001100011…，周期为5，不是m序列。

m-序列产生的充要条件

定义3.4 若n次不可约多项式p(x)的阶为2^n-1，则称p(x)是n次本原多项式。

定理3.6 设$\{a_i\}\in G(p(x))$，$\{a_i\}$为m序列的充要条件是p(x)为本原多项式。

证明：若p(x)是本原多项式，则其阶为2^n-1，得$\{a_i\}$的周期等于2^n-1，即$\{a_i\}$为m序列。

（定理3.4 不可约多项式的阶 ➔序列的阶）

反之，若$\{a_i\}$为m序列，即其周期等于2^n-1，由定理3.5知p(x)是不可约的。由定理3.3知$\{a_i\}$的周期2^n-1整除p(x)的阶，而p(x)的阶不超过2^n-1，所以p(x)的阶为2^n-1，即p(x)是本原多项式。

对于任意的正整数n，至少存在一个n次本原多项式。所以对于任意的n级LFSR，至少存在一种连接方式使其输出序列为m序列

m-序列举例

例3.5 设$p(x)=x^4+x+1$，若LFSR以$p(x)$为特征多项式，则输出序列的递推关系为

$$a_k=a_{k-1} \oplus a_{k-4} (k\geq 4)$$

若初始状态为1001，则输出为

100100011110101100100011110101…

周期为$2^4-1=15$。

若初始状态为1000，则输出为

100011110101100100011110101…

100100011110101100100011110101…

m序列的随机性

m序列满足Golomb的3个随机性公设。

定理3.7 **GF(2)上的n长m序列$\{a_i\}$具有如下性质：**

① 在一个周期内，0、1出现的次数分别为$2^{n-1}-1$和2^{n-1}。

② 在一个周期内，总游程数为2^{n-1}；对$1\le i\le n-2$，长为i的游程有2^{n-i-1}个，且0、1游程各半；长为n-1的0游程一个，长为n的1游程一个。

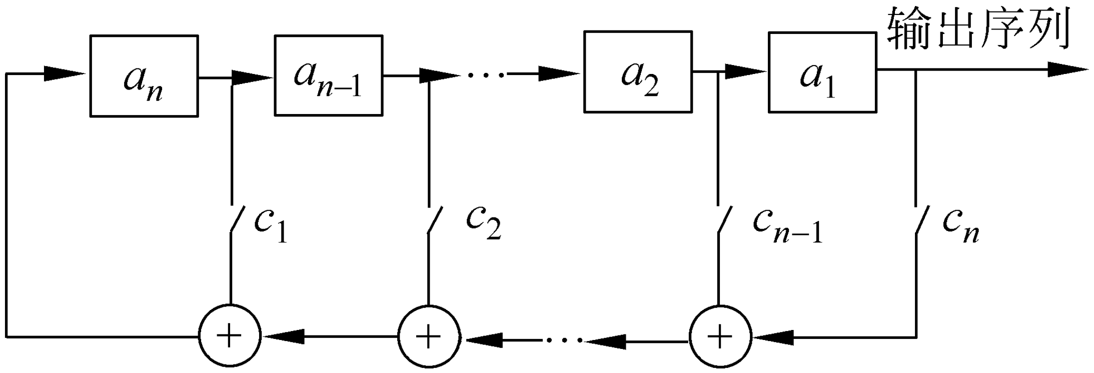
③ $\{a_i\}$的自相关函数为

$$R(\tau)=\begin{cases}1, & \tau=0\\ -\frac{1}{2^n-1}, & 0<\tau\le 2^n-2\end{cases}$$

定理3.7的证明

证明：① 在n长m序列的一个周期内，除了全0状态外，每个n长状态（共2^n-1个）都恰好出现一次。

LFSR的输出为每个状态的a_1，因此输出为1的状态必然是（**…*1）的形式，



而m序列这类状态共有2^{n-1}个，所以输出中1的个数为2^{n-1}个。

输出为0的状态必然是（**…*0）的形式，

而m序列这类状态共有$2^{n-1}-1$个（因为不能有全零状态），所以输出中0的个数为$2^{n-1}-1$个。

定理3.7的证明（续）

证明：

② 对n=1,2，易证结论成立。

对n>2，当$1\leqslant i\leqslant n-2$时，n长m序列的一个周期内，长为i的0游程数目等于序列中如下形式的状态数目： 100…01*… … *，其中n-i-2个*可任取0或1。这种状态共有2^{n-i-2}个。同理可得长为i的1游程数目也等于 2^{n-i-2}，所以长为i的游程总数为2^{n-i-1}。

定理3.7的证明（续）

由于寄存器中不会出现全0状态，所以不会出现n长的0游程，但必有一个n长的1游程，而且1的游程不会更大，因为若出现n+1长的1游程，就必然有两个相邻的全1状态，但这是不可能的。这就证明了n长的1游程必然出现在如下的串中：

$$0\underbrace{1\cdots1}_{n\text{个}1}0$$

当这n+2位通过移位寄存器时，便依次产生以下状态：

$$0\underbrace{1\cdots1}_{n-1\text{个}1}\qquad\underbrace{1\cdots1}_{n\text{个}1}\qquad\underbrace{1\cdots1}_{n-1\text{个}1}0$$

定理3.7的证明（续）

由于 $0\underbrace{1\cdots1}_{n-1个1}$ ， $\underbrace{1\cdots1}_{n-1个1}0$ 这两个状态只能各出现一次，所以不会再有1的n-1游程。

0的n-1游程只有一个： $1\underbrace{0\cdots0}_{n-1个1}1$

于是在一个周期内，总游程数为

$$1 + 1 + \sum_{i=1}^{n-2} 2^{n-i-1} = 2^{n-1}$$

定理3.7的证明（续）

③ **{a_i}是周期为2^n-1的m序列，对于任一正整数t(0<t<2^n-1)，{a_i}+{a_{i+t}}在一个周期内为0的位数正好是序列{a_i}和{a_{i+t}}对应位相同的位数。设序列{a_i}满足递推关系：**

$a_{h+n} = c_1 a_{h+n-1} \oplus c_2 a_{h+n-2} \oplus \cdots \oplus c_n a_h$

故$a_{h+n+t} = c_1 a_{h+n+t-1} \oplus c_2 a_{h+n+t-2} \oplus \cdots \oplus c_n a_{h+\tau}$

$a_{h+n} \oplus a_{h+n+t} = c_1(a_{h+n-1} \oplus a_{h+n+t-1}) \oplus c_2(a_{h+n-2} \oplus a_{h+n+t-2}) \oplus \cdots \oplus c_n(a_h \oplus a_{h+t})$

令$b_j = a_j \oplus a_{j+t}$,序列{b_j}满足：$b_{h+n} = c_1 b_{h+n-1} \oplus c_2 b_{h+n-2} \oplus \cdots \oplus c_n b_h$

因为对应同样的特征多项式，所以{b_i}也是m序列。

$$R(\tau) = (1/T)\sum_{k=1}^{T}(-1)^{a_k}(-1)^{a_{k+\tau}} = (1/T)\sum_{k=1}^{T}(-1)^{b_k} = \frac{2^{n-1}-1-2^{n-1}}{2^n-1} = -\frac{1}{2^n-1}$$

现代密码学

m-序列的安全性

信息与软件工程学院

m-序列的安全性

- 寻找m序列的递推关系式。
  - 已知一段序列，如果知道其反馈多项式，就可以将其后的序列依次求出
  - 已知序列能否获得相应的反馈多项式(或线性递推式)呢？
    - 解方程方法——已知序列$\{a_i\}$是由**n** 级线性移存器产生的，并且知道$\{a_i\}$的连续**2n**位，可用解线性方程组的方法得到反馈多项式
    - 线性反馈移位寄存器综合解——**Berlekamp-Massey**算法

m-序列的安全性

- 解方程方法
- 线性反馈移位寄存器综合

例1 设序列$\underline{a}=(01111000\ldots)$是由4级线性移存器所产生序列的连续8个信号，求该移存器的线性递推式。

解：设该4级移存器的线性递推式为：

$$\boxed{a_n = c_1 a_{n-1} \oplus c_2 a_{n-2} \oplus c_3 a_{n-3} \oplus c_4 a_{n-4} \quad (n \geq 4)}$$

由于知道周期序列的连续8个信号，不妨设为开头的8个信号，即

$$a_0 a_1 a_2 a_3 a_4 a_5 a_6 a_7 = 01111000$$

当n=4时，由递归式可得： $a_4 = c_1 a_3 \oplus c_2 a_2 \oplus c_3 a_1 \oplus c_4 a_0$

即： $c_1 \oplus c_2 \oplus c_3 = 1 \qquad (1)$

同理可得： $c_1 \oplus c_2 \oplus c_3 \oplus c_4 = 0 \qquad (2)$

$c_2 \oplus c_3 \oplus c_4 = 0 \qquad (3)$

$c_3 \oplus c_4 = 0 \qquad (4)$

解方程组得 $c_1 = 0,\ c_2 = 0,\ c_3 = 1,\ c_4 = 1$

故所求移存器递推式为： $\boxed{a_n = a_{n-3} \oplus a_{n-4}\ (n \geq 4)}$

例　　设敌手得到密文串：**101101011110010**

和相应的明文串：**011001111111001**

因此，可得相应的密钥流：**110100100001011**

进一步假定敌手还知道密钥流是使用5级线性反馈移位寄存器产生的，那么敌手可分别用密钥流中的前10个比特建立如下方程

$$(a_6 \rightleftharpoons a_7 \rightleftharpoons a_8 \rightleftharpoons a_9 \rightleftharpoons a_{10}) = (c_5 \rightleftharpoons c_4 \rightleftharpoons c_3 \rightleftharpoons c_2 \rightleftharpoons c_1)$$
$$\leftrightarrows \begin{pmatrix} a_1 \rightleftharpoons a_2 \rightleftharpoons a_3 \rightleftharpoons a_4 \rightleftharpoons a_5 \\ a_2 \rightleftharpoons a_3 \rightleftharpoons a_4 \rightleftharpoons a_5 \rightleftharpoons a_6 \\ a_3 \rightleftharpoons a_4 \rightleftharpoons a_5 \rightleftharpoons a_6 \rightleftharpoons a_7 \\ a_4 \rightleftharpoons a_5 \rightleftharpoons a_6 \rightleftharpoons a_7 \rightleftharpoons a_8 \\ a_5 \rightleftharpoons a_6 \rightleftharpoons a_7 \rightleftharpoons a_8 \rightleftharpoons a_9 \end{pmatrix}$$

$$(0 \rightleftharpoons 1 \rightleftharpoons 0 \rightleftharpoons 0 \rightleftharpoons 0) = (c_5 \rightleftharpoons c_4 \rightleftharpoons c_3 \rightleftharpoons c_2 \rightleftharpoons c_1)$$

即

$$\rightleftarrows \begin{pmatrix} 1 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \\ 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 0 \\ 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \\ 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \\ 0 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 0 \end{pmatrix}$$

而

$$\rightleftarrows \begin{pmatrix} 1 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \\ 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 0 \\ 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \\ 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \\ 0 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 0 \end{pmatrix}^{-1} = \begin{pmatrix} 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \\ 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \\ 0 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \\ 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 1 \\ 1 \nearrow\!\!\!\!\swarrow 0 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 1 \nearrow\!\!\!\!\swarrow 0 \end{pmatrix}$$

从而得到

$$\rightleftarrows (c_5 \rightleftharpoons c_4 \rightleftharpoons c_3 \rightleftharpoons c_2 \rightleftharpoons c_1)$$

$$= (0 \rightleftarrows 1 \rightleftarrows 0 \rightleftarrows 0 \rightleftarrows 0)\begin{pmatrix} 0 \nearrow 1 \nearrow 0 \nearrow 0 \nearrow 1 \\ 1 \nearrow 0 \nearrow 0 \nearrow 1 \nearrow 0 \\ 0 \nearrow 0 \nearrow 0 \nearrow 0 \nearrow 1 \\ 0 \nearrow 1 \nearrow 0 \nearrow 1 \nearrow 1 \\ 1 \nearrow 0 \nearrow 1 \nearrow 1 \nearrow 0 \end{pmatrix}$$

所以

$$(c_5 \rightleftharpoons c_4 \rightleftharpoons c_3 \rightleftharpoons c_2 \rightleftharpoons c_1) = (1 \rightleftarrows 0 \rightleftarrows 0 \rightleftarrows 1 \rightleftarrows 0)$$

密钥流的递推关系为

$$a_{i+5} = c_5 a_i \oplus c_2 a_{i+3} = a_i \oplus a_{i+3}$$

m-序列的安全性

- 解方程方法
- 线性反馈移位寄存器综合

线性反馈移位寄存器综合

根据密码学的需要，对线性反馈移位寄存器(LFSR)主要考虑下面两个问题：

（1）如何利用级数尽可能短的LFSR产生周期大、随机性能良好的序列。

这是从密钥生成角度考虑，用最小的代价产生尽可能好的、参与密码变换的序列。

（2）当已知一个长为N序列$\underline{a}$时，如何构造一个级数尽可能小的LFSR来产生它。

这是从密码分析角度来考虑，要想用线性方法重构密钥序列所必须付出的最小代价。

线性综合解

设 $\underline{a}=(a_0,a_1,\cdots,a_{N-1})$ 是 F_2上的长度为**N**的序列，而

$$\boxed{f(x)=c_0+c_1x+c_2x^2+\cdots+c_lx^l}$$

是 F_2 上的多项式，$\boldsymbol{c_0=1}$.

如果序列中的元素满足递推关系：

$$\boxed{a_k=c_1a_{k-1}\oplus c_2a_{k-2}\oplus\cdots\oplus c_la_{k-l},\ k=l,\ l+1,\ \cdots,\ N-1 \qquad (2)}$$

则称 $\langle f(x),l\rangle$ 产生二元序列$\underline{\boldsymbol{a}}$。其中 $\langle f(x),l\rangle$ 表示以***f*(*x*)**为特征多项式的***l***级线性移位寄存器。

如果***f*(*x*)**是一个能产生$\underline{\boldsymbol{a}}$并且级数最小的线性移位寄存器的特征多项式，***l***是该移存器的级数，则称 $\langle f(x),l\rangle$ 为序列$\underline{\boldsymbol{a}}$的线性综合解。

线性移位寄存器的综合问题

线性移位寄存器的综合问题可表述为：给定一个N长二元序列$\underline{a}$，如何求出产生这一序列的最小级数的线性移位寄存器，即最短的线性移存器。

1、特征多项式$f(x)$的次数$\le l$。因为产生$\underline{a}$且级数最小的线性移位寄存器可能是退化的，在这种情况下$f(x)$的次数$<l$；并且此时$f(x)$中的$c_l=0$，因此在特征多项式$f(x)$中仅要求$c_0=1$，但不要求$c_l=1$。

2、规定：0级线性移位寄存器是以$f(x)=1$为特征多项式的线性移位寄存器，且n长$(n=1, 2, \ldots, N)$全零序列，仅由0级线性移位寄存器产生。事实上，以$f(x)=1$为反馈特征多项式的递归关系式是：$a_k=0$，$k=0, 1, \ldots, n-1$.因此，这一规定是合理的。

3、给定一个N长二元序列$\underline{a}$，求能产生$\underline{a}$并且级数最小的线性移位寄存器，就是求$\underline{a}$的线性综合解。利用B-M算法可以有效的求出。

Berlekamp-Massey算法（B-M算法）

用归纳法求出一系列线性移位寄存器：

$$\langle f_n(x), l_n \rangle \qquad \partial^0 f_n(x) \le l_n, \quad n = 1,2,\cdots,N$$

每一个 $\langle f_n(x), l_n \rangle$ 都是产生序列$\underline{\boldsymbol{a}}$的前$\boldsymbol{n}$项的最短线性移位寄存器，在 $\langle f_n(x), l_n \rangle$ 的基础上构造相应的 $\langle f_{n+1}(x), l_{n+1} \rangle$ ，使得是 $\langle f_{n+1}(x), l_{n+1} \rangle$ 产生给定序列前**n+1**项的最短移存器，则最后得到的 $\langle f_N(x), l_N \rangle$ 就是产生给定$\boldsymbol{N}$长二元序列$\underline{\boldsymbol{a}}$的最短的线性移位寄存器。

B-M算法（续）

任意给定一个N长序列 $\underline{a} = (a_0, a_1, \cdots, a_{N-1})$ ，按n归纳定义

$\langle f_n(x), l_n \rangle$ $n = 0, 1, 2, \cdots, N-1$

1、取初始值： $f_0(x) = 1, \quad l_0 = 0$

2、设 $\langle f_0(x), l_0 \rangle, \langle f_1(x), l_1 \rangle, \cdots, \langle f_n(x), l_n \rangle$ $(0 \le n < N)$ 均已求得，

且 $l_0 \le l_1 \le \cdots \le l_n$

记： $f_n(x) = c_0{}^{(n)} + c_1{}^{(n)} x + \cdots c_{l_n}{}^{(n)} x^{l_n}, c_0{}^{(n)} = 1,$ 再计算：

$$d_n = c_0{}^{(n)} a_n + c_1{}^{(n)} a_{n-1} + \cdots + c_{l_n}{}^{(n)} a_{n-l_n}$$

称d_n为第n步差值。然后分两种情形讨论：

B-M算法（续）

（i） 若 d_n =**0**，则令：

$$f_{n+1}(x)=f_n(x),\quad l_{n+1}=l_n\text{。}$$

（ii） 若 d_n =**1**，则需区分以下两种情形：

① 当：$l_0=l_1=\cdots=l_n=0$ 时，

取： $f_{n+1}(x)=1+x^{n+1}, l_{n+1}=n+1$。

② 当有 **m**（$0\le m<n$），使：$l_m<l_{m+1}=l_{m+2}=\cdots=l_n$。

便置： $f_{n+1}(x)=f_n(x)+x^{n-m}f_m(x), l_{n+1}=\max\{l_n, n+1-l_n\}$

最后得到的 $\langle f_N(x), l_N\rangle$ 便是产生序列$\underline{\boldsymbol{a}}$的最短线性移位寄存器。

B-M算法举例

输入：S^8=10101111

n	d_n	f_n	L_n	m	f_m
0	1	1	0		
1	1	$1+x$	1	0	1
2	1	1	1	0	1
3	0	$1+x^2$	2		
4	0	$1+x^2$	2		
5	1	$1+x^2$	2	2	1
6	0	$1+x^2+x^3$	4		
7	1	$1+x^2+x^3$	4	5	$1+x^2$
8		$1+x^3+x^4$	4		

输出：$\langle 1+x^3+x^4, 4\rangle$

$$f_n(x) = c_0{}^{(n)} + c_1{}^{(n)}x + \cdots c_{l_n}{}^{(n)}x^{l_n}, c_0{}^{(n)} = 1,$$

$$d_n = c_0{}^{(n)}a_n + c_1{}^{(n)}a_{n-1} + \cdots + c_{l_n}{}^{(n)}a_{n-l_n}$$

（i） 若 d_n=0，则令：

$$f_{n+1}(x) = f_n(x), \quad l_{n+1} = l_n。$$

（ii） 若 d_n=1，则需区分以下两种情形：

① 当：$l_0 = l_1 = \cdots = l_n = 0$ 时，

取：$f_{n+1}(x) = 1 + x^{n+1}, l_{n+1} = n+1$。

② 当有 **m**（$0 \le m < n$），使：

③ $l_m < l_{m+1} = l_{m+2} = \cdots = l_n$。

便置：$f_{n+1}(x) = f_n(x) + x^{n-m} f_m(x),$
$l_{n+1} = \max\{l_n, n+1-l_n\}$

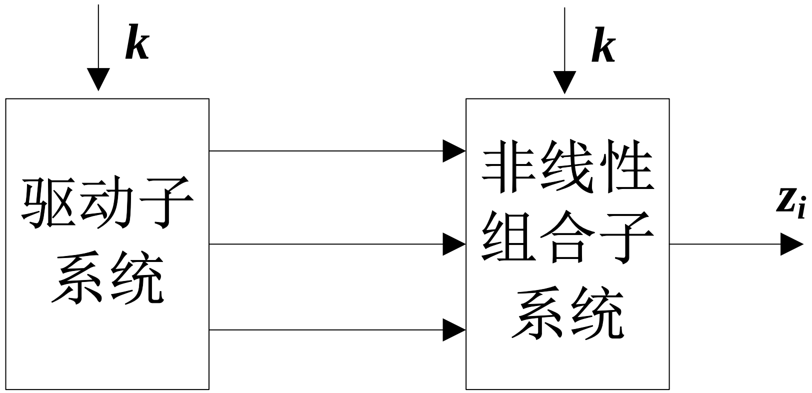
现代密码学

非线性序列

信息与软件工程学院

非线性序列

- 密钥流生成器可分解为驱动子系统和非线性组合子系统，如图所示



密钥流生成器的分解

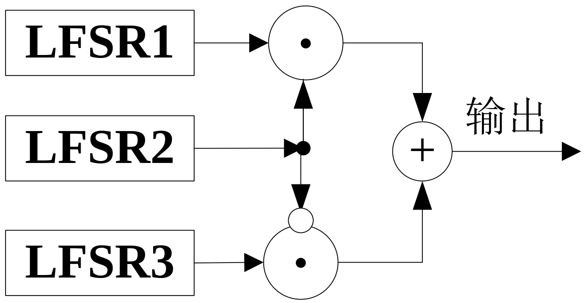
- 驱动子系统常用一个或多个线性反馈移位寄存器来实现
- 非线性组合子系统用非线性组合函数F来实现
- 为了使密钥流生成器输出的二元序列尽可能复杂，也应保证其周期尽可能大、线性复杂度和不可预测性尽可能高

非线性序列

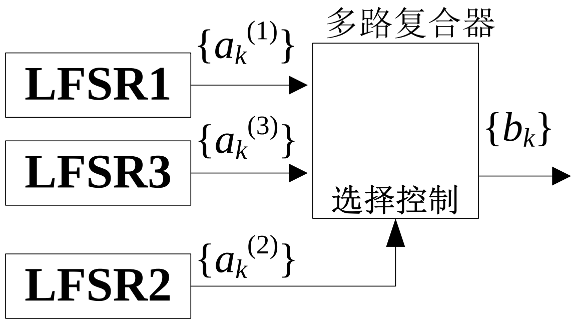
- Geffe序列生成器
- J-K触发器
- Pless生成器

• Geffe序列生成器

- Geffe序列生成器由3个LFSR组成，其中LFSR2作为控制生成器使用，如图所示



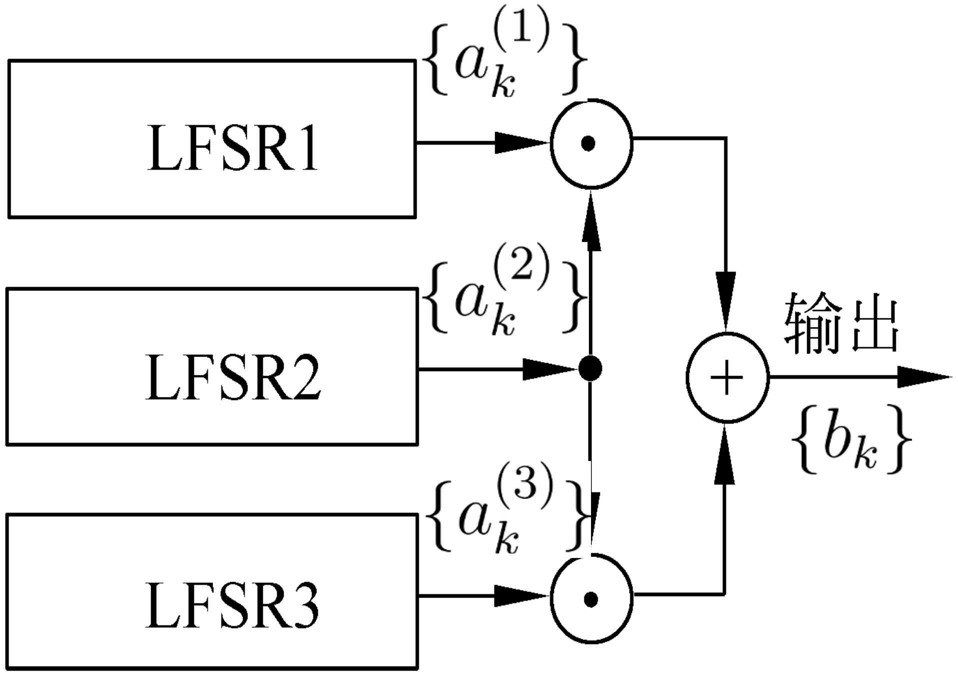
Geffe序列生成器



多路复合器表示**Geffe序列生成器**

- 当LFSR2输出1时，LFSR2与LFSR1相连接
- 当LFSR2输出0时，LFSR2与LFSR3相连接

Geffe序列生成器（续）



若设LFSRi的输出序列为$\{a_k^{(i)}\}$ (i=1,2,3)，则输出序列$\{b_k\}$可以表示为

$$b_k = a_k^{(1)}a_k^{(2)} + a_k^{(3)}\overline{a_k^{(2)}} = a_k^{(1)}a_k^{(2)} + a_k^{(3)}a_k^{(2)} + a_k^{(3)}$$

设LFSRi的特征多项式分别为n_i次本原多项式，且n_i 两两互素

则Geffe序列的周期= $\prod_{i=1}^{3}(2^{n_i}-1)$

Geffe序列的周期实现了极大化，

且**0**与**1**之间的分布大体上是平衡的。

线性复杂度= $(n_1+n_3)n_2+n_3$

非线性序列

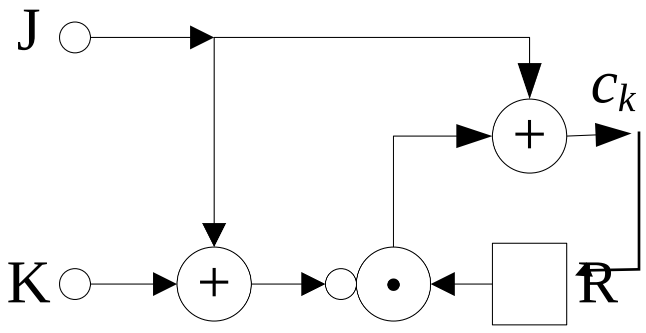
- Geffe序列生成器
- J-K触发器
- Pless生成器

J-K触发器

J-K触发器如图所示，它的两个输入端分别用J和K表示，其输出c_k不仅依赖于输入，还依赖于前一个输出位c_{k-1}，即

$$c_k = \overline{(x_1 + x_2)} \rightleftarrows c_{k-1} + x_1$$

其中x_1和x_2分别是J和K端的输入。由此可得J-K触发器的真值表，如下表所示

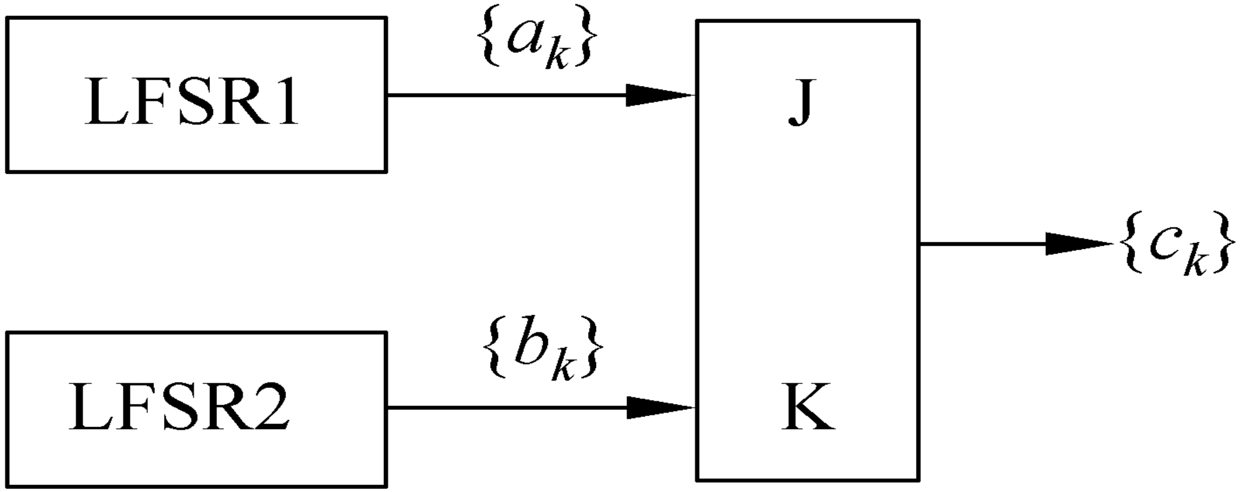


J-K触发器

J	K	c_k
0	0	c_{k-1}
0	1	0
1	0	1
1	1	$\overline{c_{k-1}}$

J-K触发器真值表

利用J-K触发器的非线性序列生成器



$\{a_k\}$:m级m序列　　　$\{b_k\}$:n级m序列

$$c_k = \overline{(a_k + b_k)} \cdot c_{k-1} + a_k = (a_k + b_k + 1) \cdot c_{k-1} + a_k$$

当**m**与**n**互素且**$a_0+b_0=1$**时，序列**$\{c_k\}$**的周期为**$(2^m-1)(2^n-1)$**。

利用J-K触发器的非线性序列生成器的实例

$$c_k = \overline{(a_k + b_k)} \rightleftarrows c_{k-1} + a_k = (a_k + b_k + 1) \rightleftarrows c_{k-1} + a_k$$

例2.7 令m=2,n=3，两个驱动m序列分别为

$$\{a_k\}=0,1,1,\ldots$$

和

$$\{b_k\}=1,0,0,1,0,1,1,\ldots$$

于是，输出序列$\{c_k\}$是0,1,1,0,1,0,0,1,1,1,0,1,0,1,0,0,1,0,0,1,0,…,

其周期为$(2^2-1)(2^3-1)=21$。

弱点

由$c_k=(a_k+b_k+1)c_{k-1}+a_k$可得

$$c_k=\begin{cases} a_k, & c_{k-1}=0 \\ \overline{b_k}, & c_{k-1}=1 \end{cases}$$

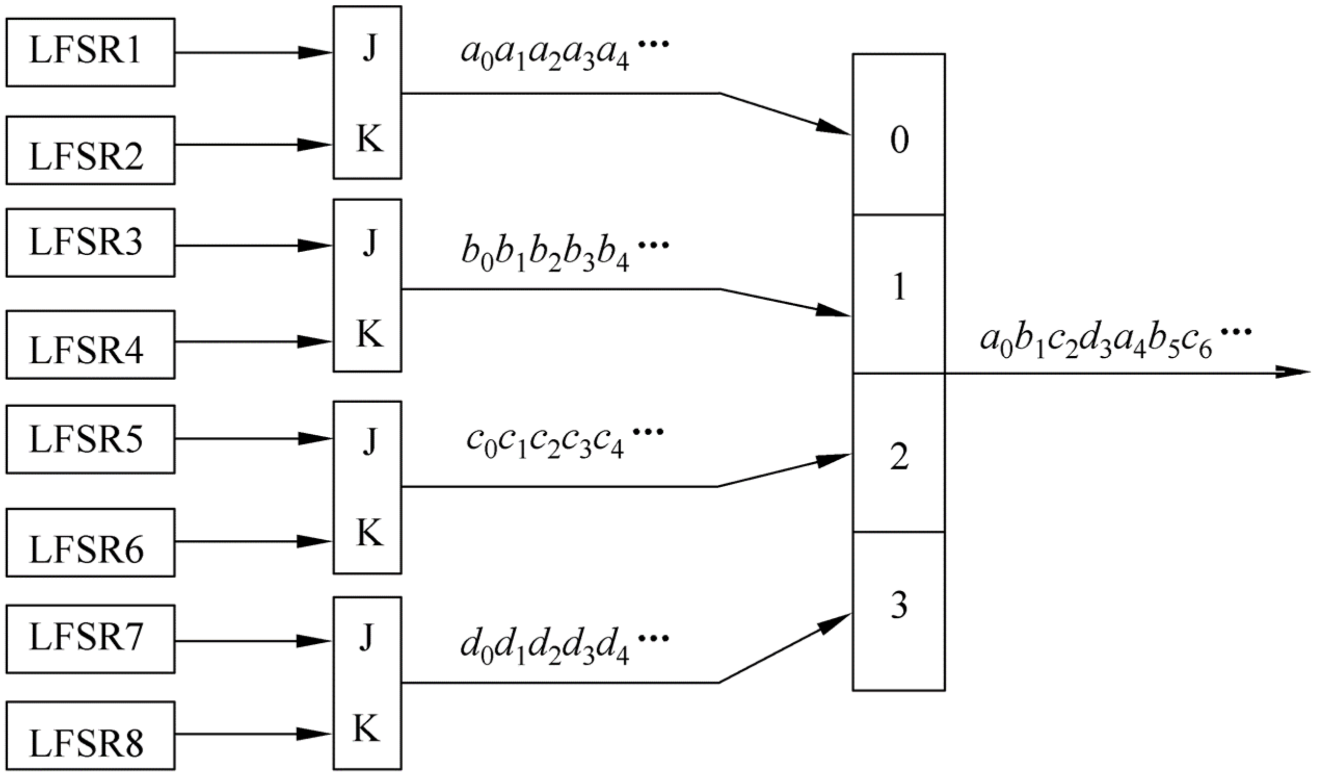
- 如果知道$\{c_k\}$中相邻位的值c_{k-1}和c_k，就可以推断出a_k和b_k中的一个。而一旦知道足够多的这类信息，就可通过密码分析的方法得到序列$\{a_k\}$和$\{b_k\}$。
- 为了克服上述缺点，**Pless**提出了由多个**J-K**触发器序列驱动的多路复合序列方案，称为**Pless**生成器。

非线性序列

- Geffe序列生成器
- J-K触发器
- Pless生成器

Pless生成器

Pless生成器由8个LFSR、4个J-K触发器和1个循环计数器构成，由循环计数器进行选通控制，如图所示。

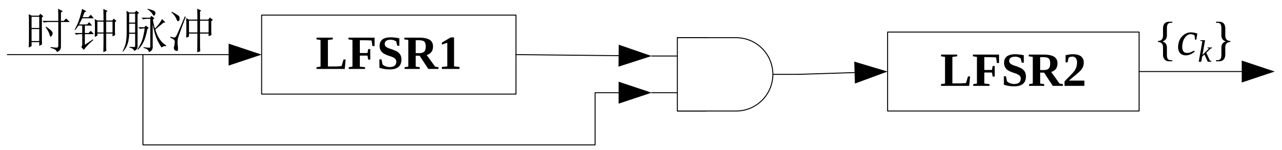


非线性序列

钟控序列生成器

钟控序列生成器模型

- 钟控序列最基本的模型是用一个LFSR控制另外一个LFSR的移位时钟脉冲，如图所示，一个最简单钟控序列生成器



  - 假设LFSR1和LFSR2分别输出序列$\{a_k\}$和$\{b_k\}$，其周期分别为p_1和p_2。
  - 当LFSR1输出1时，移位时钟脉冲通过与门使LFSR2进行一次移位，从而生成下一位。
  - 当LFSR1输出0时，移位时钟脉冲无法通过与门影响LFSR2。因此LFSR2重复输出前一位。

钟控序列的周期

- 假设LFSR1和LFSR2分别输出序列$\{a_k\}$和$\{b_k\}$，其周期分别为p_1和p_2。假设钟控序列$\{c_k\}$的周期为p，可得如下关系：
- $p=\dfrac{p_1p_2}{\gcd(w_1,p_2)}$，其中$w_1=\sum_{i=0}^{p_1-1} a_i$
- c_k的一个周期至少是LFSR1和LFSR2同时回到初始状态的时刻
  - 显然当运行$p_1\times p_2$个节拍后两个LFSR必然回到初态，因此周期至多是$p_1\times p_2$
  - LFSR1运行一个周期，LFSR2运行$w_1=dt$拍，$d=\gcd(w_1,p_2)$
  - 则LFSR1运行（p_2/d）个周期后，LFSR2刚好运行$dt\times p_2/d=tp_2$拍，即t个周期，于是两个LFSR都回到初态，这时运行了（p_2/d）$\times p_1$个节拍

钟控序列的周期（续）

- 若$\{a_k\}$和$\{b_k\}$的极小特征多项式分别为GF(2)上的m和n次本原多项式$f_1(x)$和$f_2(x)$，且$m|n$。
- 则$p_1=2^m-1$, $p_2=2^n-1$。
- 而w_1=为$\{a_k\}$一个周期内1的个数，因此$w_1=2^{m-1}$
- 故$\gcd(w_1, p_2)=1$，所以$p=p_1p_2=(2^m-1)(2^n-1)$。

钟控序列的线性复杂度

- 可推导出$\{c_k\}$的线性复杂度为$n(2^m\text{-}1)$，极小特征多项式为 $f_2(x^{2^m-1})$
  - 其对应的LFSR2的抽头每隔周期$p_1=2^m-1$一个，这样，参与运算的每个抽头对应的状态的节奏相同，从而相当于对LFSR2序列进行每2^{m-1}拍的抽样序列(不计由于LFSR1的0游程而产生的重复)，这个序列只是LFSR2的平移和按照LFSR1中的0游程进行迟延，而抽头应该与LFSR2的节奏一致，所以其极小多项式和线性复杂度如上

钟控序列的例子

- 例： 设LFSR1为3级m序列生成器，其特征多项式为$f_1(x)=1+x+x^3$。设初态为$a_0=a_1=a_2=1$，于是输出序列为$\{a_k\}$=1,1,1,0,1,0,0,…
- 又设LFSR2为3级m序列生成器，且记其状态向量为σ_k，则在上图的构造下σ_k的变化情况下：
  - σ_0，σ_1，σ_2，σ_3，σ_3，σ_4，σ_4，σ_4，
  - σ_5，σ_6，σ_0，σ_0，σ_1，σ_1，σ_1，
  - σ_2，σ_3，σ_4，σ_4，σ_5，σ_5，σ_5，
  - σ_6，σ_0，σ_1，σ_1，σ_2，σ_2，σ_2，
  - σ_0，σ_1，σ_2，σ_2，σ_3，σ_3，σ_3，
  - σ_4，σ_5，σ_6，σ_6，σ_0，σ_0，…
- $\{c_k\}$的周期为$(2^3-1)^2=49$，在它的一个周期内，每个σ_k恰好出现7次

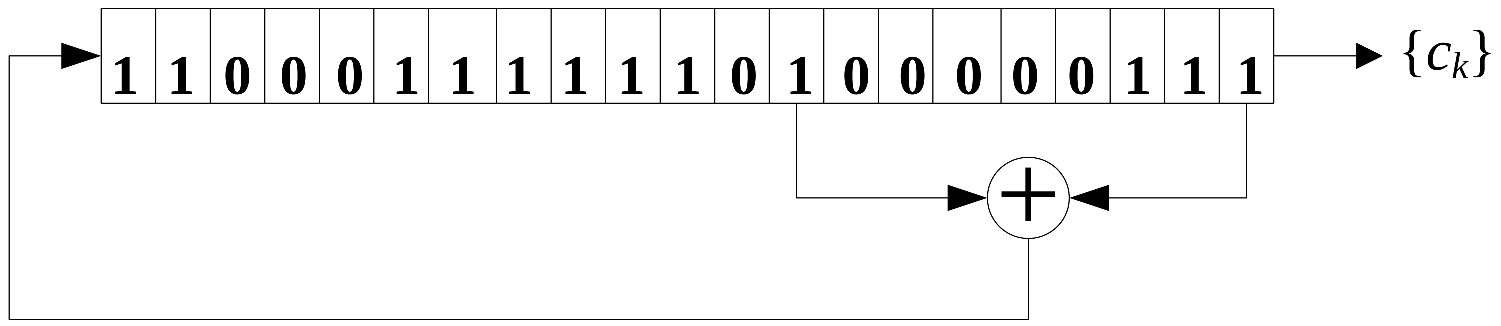
例（续）

- 设$f_2(x)=1+x^2+x^3$为LFSR2的特征多项式，且初态为$b_0=b_1=b_2=1$，则$\{b_k\}$=1,1,1,0,0,1,0,1,1,1…
  - 由σ_k的变化情况得$\{c_k\}$=1,1,1,0,0,0,0,0, 1,0,1,1,1,1,1, 1,0,0,0,1,1,1, 0,1,1,1,1,1,1, 0,0,1,1,0,0,0, 1,1,1,1,0,0,0, 0,1,0,0,1,1,…

σ_0，σ_1，σ_2，σ_3，σ_3，σ_4，σ_4，σ_4，
- σ_5，σ_6，σ_0，σ_0，σ_1，σ_1，σ_1，
- σ_2，σ_3，σ_4，σ_4，σ_5，σ_5，σ_5，
- σ_6，σ_0，σ_1，σ_1，σ_2，σ_2，σ_2，
- σ_0，σ_1，σ_2，σ_2，σ_3，σ_3，σ_3，
- σ_4，σ_5，σ_6，σ_6，σ_0，σ_0，…

状态（b_3,b_2,b_1）	输出
σ_0 1 1 1	1
σ_1 0 1 1	1
σ_2 0 0 1	1
σ_3 1 0 0	0
σ_4 0 1 0	0
σ_5 1 0 1	1
σ_6 1 1 0	0

- $\{c_k\}$的极小特征多项式为$1+x^{14}+x^{21}$，其线性复杂度为$3\cdot(2^3-1)=21$，下图是其线性等价生成器。



感 谢 聆 听！