密码学 之
序列密码

- 范明钰
- 信息安全研究中心

要点

例子和基本概念

应用背景、发展状况

序列密码的一般设计原理

- 同步序列密码、有限状态机
- 密钥序列生成器及其分解
- 密钥序列生成器的一般结构
- 典型的设计手段
- 线性反馈移位寄存器
- m序列：概念、分析和综合

序列密码的典型分析方法

常见的序列密码算法

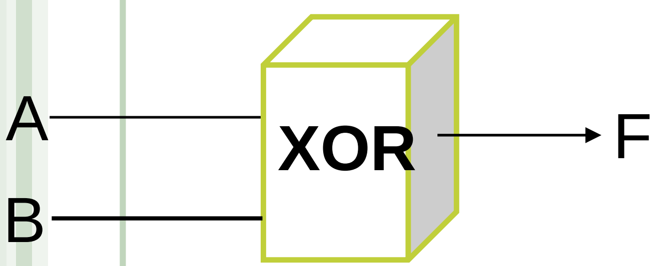
- Chameleon、FISH、Helix、ISAAC、MUGI、Panama、Phelix、Pike、SEAL、SOBER、SOBER-128、WAKE等
- RC4, used in Netscape‘s Secure Socket Layer (SSL) protocol
- A5, in the Global System for Mobile Communication (GSM)
- E0， Bluetooth stream cipher, standard for wireless short-range connectivity, specified by the Bluetooth Special Interest Group

序列密码加密方式举例

编码：明文转换为bit，例如采用ASCII编码，明文为TV（T=19=10011，V=21=10101），则编码为1001110101

密钥流：给定伪随机生成器，通信双方约定初态和起点，得到伪随机bit流，假定为1100000110，称为密钥流

运算：异或



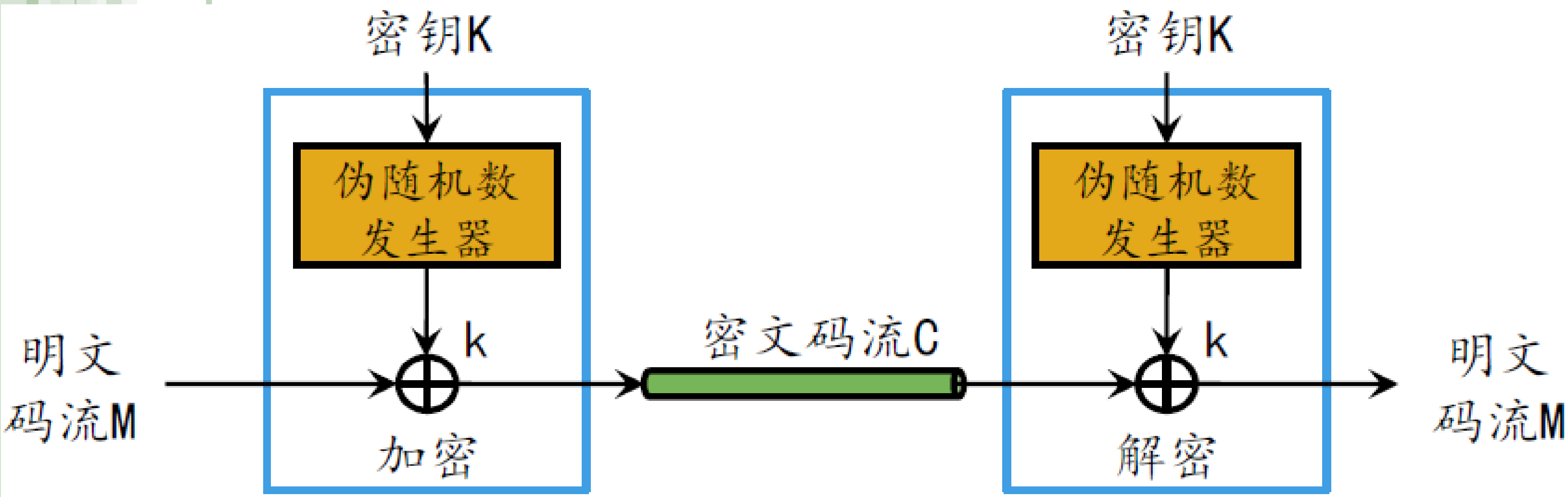
A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

A =明文，B=key

更一般的序列密码加密例子

plaintext:	0111001101011101000111010111011111011....
random seq.:	1011010110001001010000110110111010101....
ciphertext:	1100011011010100010111100001100101110....
random seq.:	1011010110001001010000110110111010101....
plaintext:	0111001101011101000111010111011111011....

模型、例子




```
Sender:     T       V                       Receiver:
        PT  1001110101                      CT  0101110011
   + k_i  1100000110                   + k_i  1100000110
          ----------                          ----------
        CT  0101110011                      PT  1001110101
                                                  T       V
```

基本情况

- 序列密码有广泛的理论基础，对于其各种设计原则已经进行了详尽的分析。然而在公开的文献中详尽的序列密码系统却相对较少
- 造成这种状况的部分原因是，实际中使用的大部分序列密码归私人所有或需要保密。相比之下，大量的分组密码建议已经公开，其中的一些已经标准化

应用背景

- 最初主要用于政府、军方等国家要害部门，因此，不像分组密码那样有公开的国际标准，大多数设计、分析成果都是保密的。但是随着序列密码的应用需求越来越广泛，从*NESSIE*工程开始，序列密码算法的设计与分析也列上了公开征集评测的日程
- 2000年1月欧洲启动的*NESSIE*工程中，有6个序列密码算法(*Leviathan*、*UIi-128*、*BMGL*、*SOBER-t32*、*SNOW*、*SOBER-tl*)进入了第二阶段评估，但是因为不符合*NESSIE*的征集准则而最终全部落选

发展状况

- 2003.3，日本密码研究与评估委员会(CRYPTREC)推荐了**3个流密码算法**：*MUGI*、*MULTI-S01*和*RC4-128*。
- 2004.2，***ECRYPT***欧洲第6框架研究计划(FP6)下*IST*基金支持的一个为期4年的项目，同年10.14—15在比利时举行了一个名为*SASC*的特别会议，引发了流密码算法的征集活动，并于2004.11发布征集公告，也是对***NESSIE***没有征集到流密码算法的补充。征集活动到2005.4.29结束，根据4个征集原则，一共征集到34个流密码算法
- 2007年4月进入第三轮评估，针对软件设计的候选算法有*CryptMT(Version3)*、*Dragon*、*Rabbit*、*HC(HC-128*和*HC-256)*、*LEX(LEX-128*、*LEX-192*和*LEX-256)*、*NLS(NLSv2*加密)、*Salsa20*和*SOSEMANUK*。针对硬件设计的候选算法有*DECIM(DECIMv2*和*DECIM-128)*、*F-FCSR(F-FCSR-Hv2*和*F-FCSR-16)*、*Edon80*、*Grain(Grainv1*和*Grain-128)*、*MICKEY(MICKEY2.0*和*MICKEY-1282.0)*、*Moustique*、*Trivium*和*Pomaranch (Version 3)*

基本概念

- 序列密码的基本思想是，利用密钥k产生一个密钥流$z=z_0z_1\cdots$，并使用如下规则对明文串$x=x_0x_1x_2\cdots$加密：
- $y=y_0y_1y_2\ldots=E_{z0}(x_0)E_{z1}(x_1)E_{z2}(x_2)\ldots$
- 换言之：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（存储器）在时刻i的状态，f是由密钥k和σ_i产生的函数。

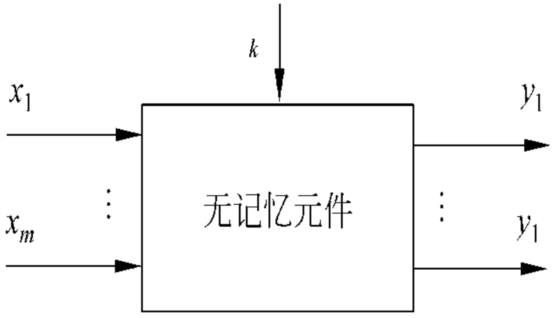
基本概念

序列密码将明文消息M连续地分成字符→bit，用密钥流来加密每个字符→bit

基本上，序列密码体制只使用**混乱**技术，而不使用散布技术。这使得这种体制没有错误扩散

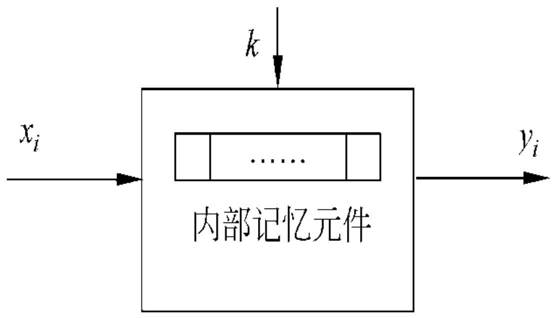
基本概念

- 分组密码与序列密码的主要区别，在于有无记忆
- 序列密码的滚动密钥$z_i=f(k,\sigma_0)$由函数f、密钥k和指定的初态σ_0完全确定。
- 由于输入的明文，可能影响内部记忆元件的状态σ_i，因而$\sigma_i(i>0)$可能依赖于（k，σ_0，x_0，x_1，…，x_{i-1}）等参数。



$y=E_k(x)$

(a) 分组密码



$y_i=E_{z_i}(x_i)$

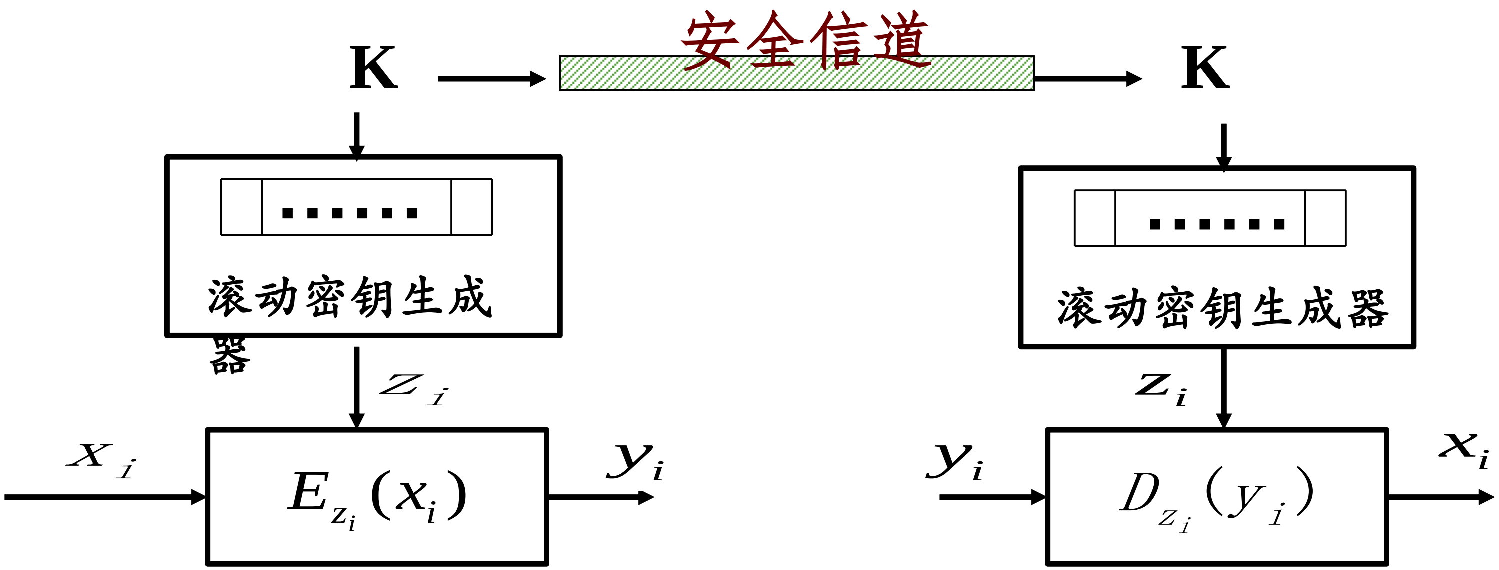
(b) 流密码

同步序列密码：概念

- 根据加密器中记忆元件的状态σ_i是否依赖于输入的明（或密）文字符，序列密码可进一步分成同步和自同步两种。
- σ_i独立于明（或密）文字符的叫做同步序列密码，否则叫做自同步序列密码。
- 由于自同步序列密码的密钥流的产生与明（或密）文有关，因而较难从理论上进行分析。目前大多数研究成果都是关于同步序列密码的。

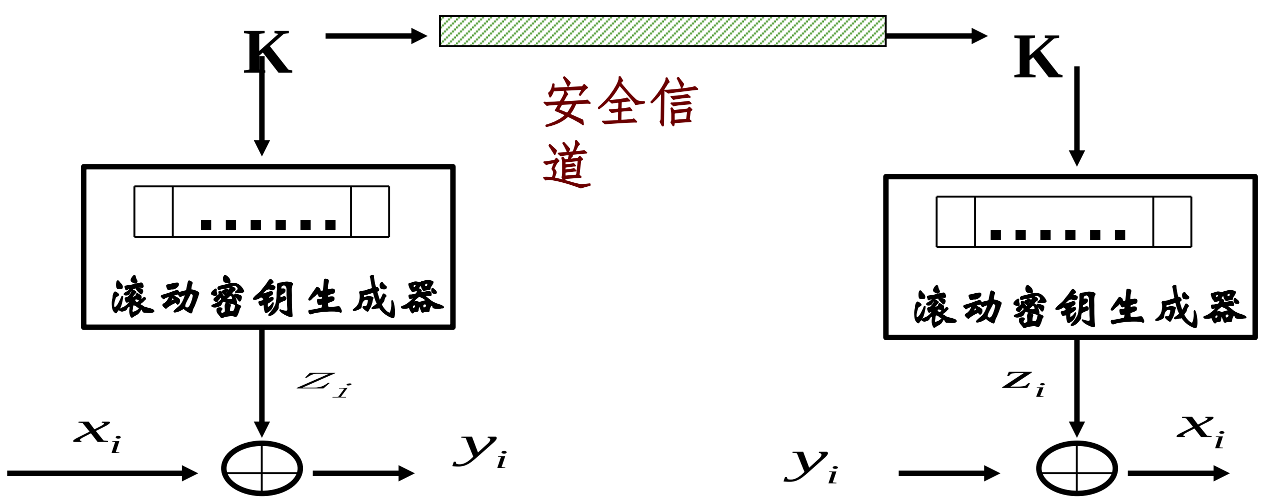
同步序列密码：模型

- 由于$z_i=f(k, \sigma_i)$与明文无关，因而密文字符$y_i=E_{z_i}(x_i)$也不依赖于此前的明文字符。因此，可将同步序列密码的加密器分成**密钥流产生器，和加密变换器**两个部分。
- 如果与上述加密变换对应的解密变换为$x_i=D_{z_i}(y_i)$，则同步序列密码体制的模型如下图。



同步序列密码：运算

- 同步序列密码的加密变换E_{zi}可以有多种选择，**只需要保证变换是可逆的。**
- 实际使用的，一般都是二元系统，因而在有限域$CF(2)$上讨论的二元加法序列密码。其加密变换可表示为
- $y_i=z_i \oplus x_i$。



同步序列密码：设计

- 要求：
- 设计出的滚动密钥生成器，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、抗统计分析……**

范明钰 2019年本科密码学

补充：同步序列密码的数学基础之一

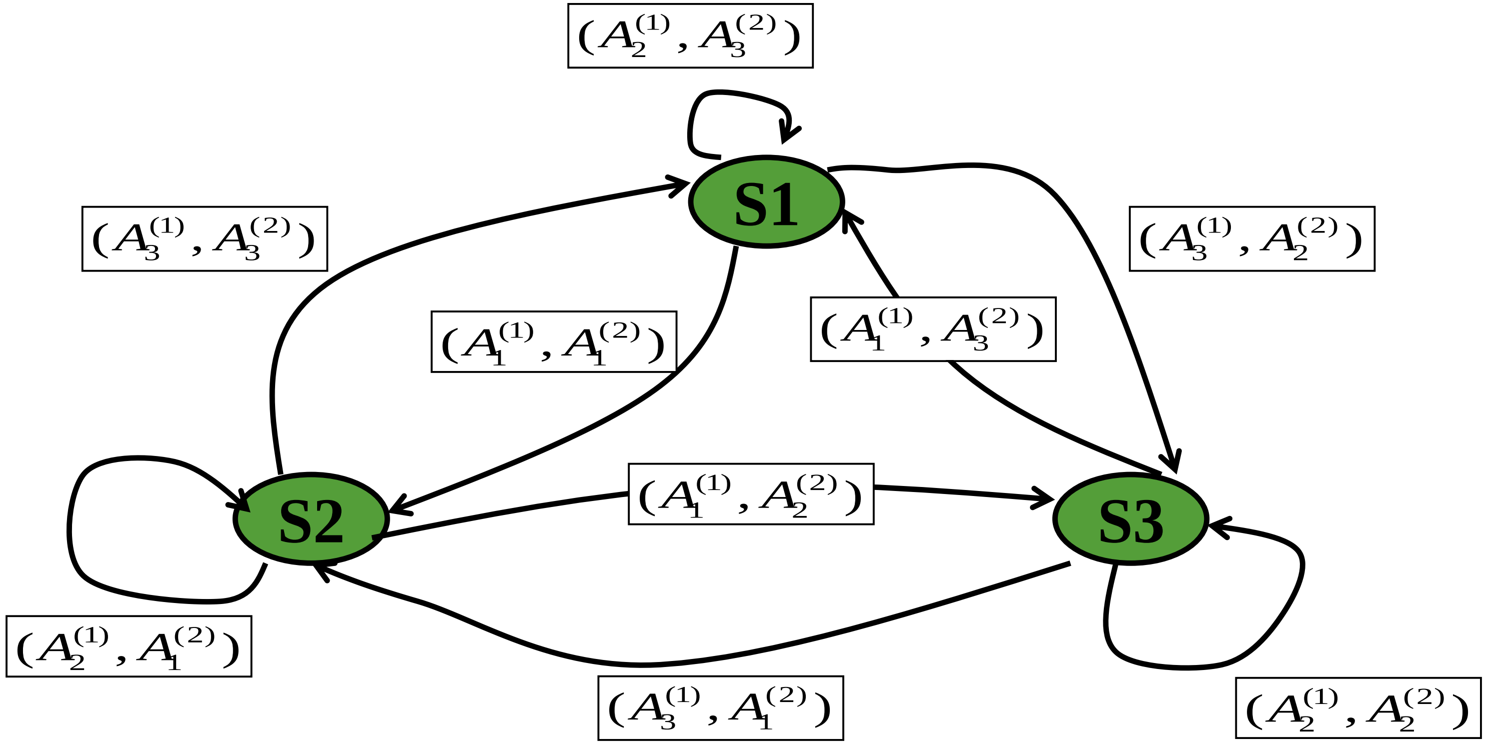
- 有限状态机
- 状态转移图
- 密钥序列生成器

有限状态自动机

- 有限状态自动机是具有离散输入和输出（输入集和输出集均有限）的一种数学模型，由以下**3部分组成**：
- ① 有限状态集 $S=\{s_i \mid i=1,2,\ldots.l\}$
- ② 有限输入字符集 $A_1=\{A_j^{(1)} \mid j=1,2,\ldots,m\}$ 和有限输出字符集 $A_2=\{A_k^{(2)} \mid k=1,2,\ldots,n\}$，
- ③ 转移函数 $A_k^{(2)}=f_1(s_i,A_j^{(1)}), s_k=f_2(s_i,A_j^{(1)})$ 即在状态为s_i，输入为$A_j^{(1)}$时，输出为$A_k^{(2)}$，而状态转移为s_k。

状态转移

◆ 有限状态自动机可用有向图表示，称为**转移图**。转移图的顶点对应于自动机的状态，若状态s_i在输入$A^{(1)}{}_i$时转为状态s_j，且输出一字符$A^{(2)}{}_j$，则在转移图中，从状态s_i到状态s_j有一条标有$(A^{(1)}{}_i, A^{(2)}{}_j)$的弧线。



例子

$$S=\{s_1,s_2,s_3\},A_1=\{A_1^{(1)},A_2^{(1)},A_3^{(1)}\},$$
$$A_2=\{A_1^{(2)},A_2^{(2)},A_3^{(2)}\}$$

- 转移函数由上一页图给出

写成表

f_1	$A_1^{(1)}$	$A_2^{(1)}$	$A_3^{(1)}$
s_1	$A_1^{(2)}$	$A_3^{(2)}$	$A_2^{(2)}$
s_2	$A_2^{(2)}$	$A_1^{(2)}$	$A_3^{(2)}$
s_3	$A_3^{(2)}$	$A_2^{(2)}$	$A_1^{(2)}$

f_2	$A_1^{(1)}$	$A_2^{(1)}$	$A_3^{(1)}$
s_1	s_2	s_1	s_3
s_2	s_3	s_2	s_1
s_3	s_1	s_3	s_2

  - 若输入序列为

$$A_1^{(1)}A_2^{(1)}A_1^{(1)}A_3^{(1)}A_3^{(1)}A_1^{(1)}$$

  - 初始状态为s_1，则得到状态序列：
  - $s_1s_2s_2s_3s_2s_1s_2$
  - 输出字符序列为

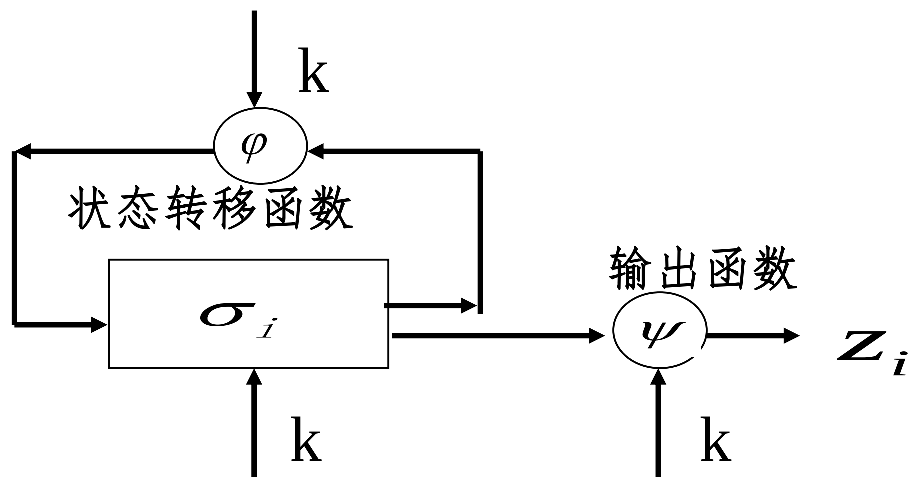
$$A_1^{(2)}A_1^{(2)}A_2^{(2)}A_1^{(2)}A_3^{(2)}A_1^{(2)}$$

提问：

- 如果上面的输入序列变为：$A_3^{(1)}A_1^{(1)}A_2^{(1)}A_1^{(1)}A_3^{(1)}A_2^{(1)}$
则上面的结果是什么，为什么？

同步序列密码密钥流产生器

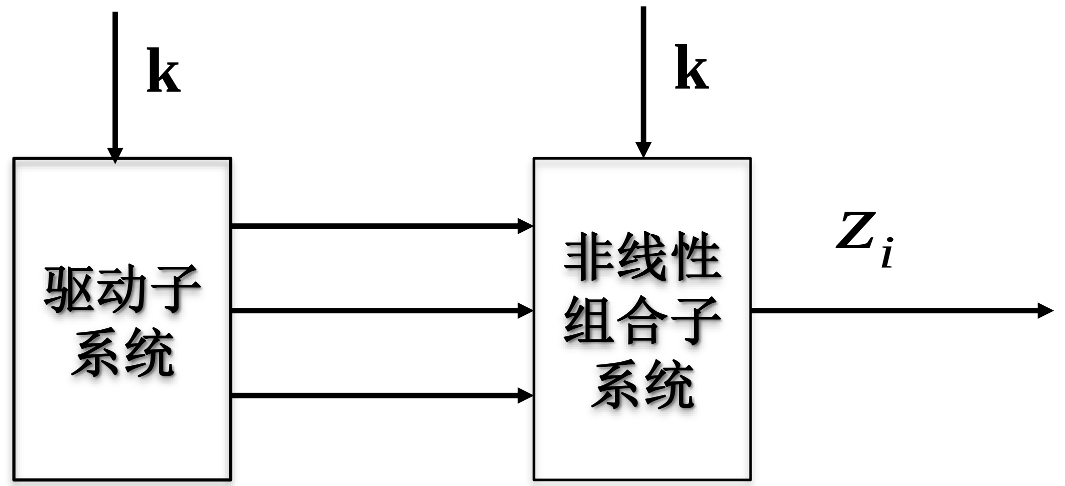
- 设计关键是**密钥流产生器**。一般可将其看成一个参数为k的有限状态自动机，由一个输出符号集Z、一个状态集∑、两个函数φ和ψ以及一个初始状态σ_0组成
- **状态转移函数**$\varphi:\sigma_i\rightarrow\sigma_{i+1}$，将当前状态$\sigma_i$变为一个新状态$\sigma_{i+1}$
- **输出函数**$\psi:\sigma_i\rightarrow z_i$，当前状态$\sigma_i$变为输出符号集中的一个元素$z_i$



- 关键问题：找出适当的**状态转移函数φ和输出函数ψ**，使得输出序列z满足要求的条件，并且容易实现
- 为了实现这一目标，必须采用**非线性函数**

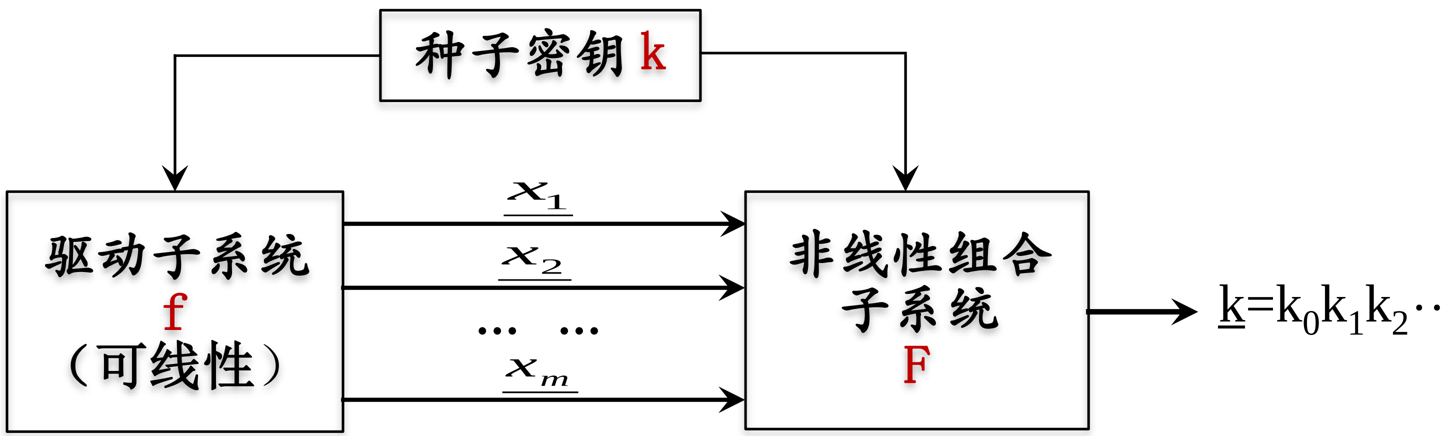
同步序列密码密钥流产生器

- 具有非线性φ的有限状态自动机理论很不完善，相应的密钥流产生器的分析工作受到极大限制。相反，当**采用线性的φ和非线性的ψ时**，将能够进行深入的分析并可以得到好的生成器。为方便讨论，将这类生成器分成**驱动部分**和**非线性组合部分**。
- 驱动部分控制生成器的状态转移，并为非线性组合部分提供统计性能好的序列；而非线性组合部分要利用这些序列组合出满足要求的密钥流序列。



KG的一般结构

- 通常，把KG设计成具有一定的结构特点，方便分析和论证其强度，增加使用者的置信度。有以下模式



  - f —— 一般由*m*-序列(或*M*-序列)生成器构成，提供若干周期大、统计特性好的序列 $\underline{x_1}, \underline{x_2}, \cdots, \underline{x_m}$ (称为**驱动序列**)
  - **F** —— 是把驱动序列综合在一起的非线性编码手段，目的是有效地破坏和掩盖多条驱动序列中存在的规律或关系，提高线性复杂度

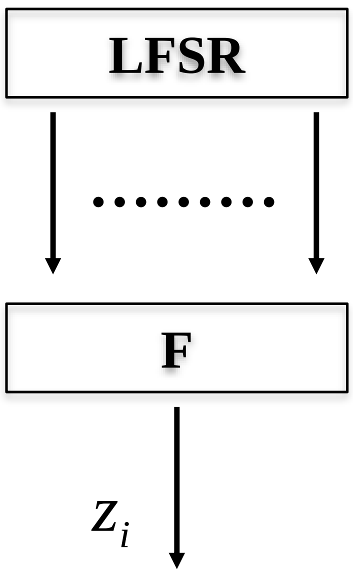
密钥序列生成器(KG)基本要求

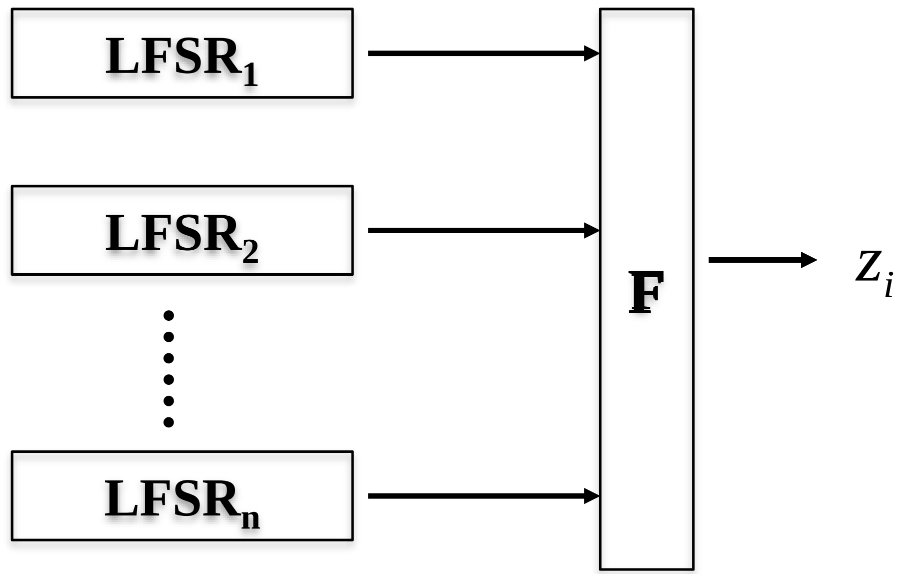
就目前的研究和预见，对KG提出了以下基本要求：

- 种子密钥k的变化量足够大，一般应在2^{128}以上；
- KG产生的密钥序列k具有极大周期，一般应不小于2^{55}；
- k具有均匀的n-元分布，即在一个周期环上，某特定形式的n-长bit串与其求反，两者出现的频数大抵相当(例如，均匀的游程分布)；k不可由一个低级(比如，小于10^6级)的LFSR产生，也不可与一个低级LFSR产生的序列只有比率很小的相异项；
- 利用统计方法由k提取关于KG结构或k的信息在计算上不可行；
- 混乱性：k的每一bit均与k的大多数bit有关；
- 扩散性：k任一bit的改变要引起k在全貌上的变化。

常见的两种密钥流产生器

◆ 目前最为流行和实用的密钥流产生器如图所示，其驱动部分是<u>一个或多个线性反馈移位寄存器</u>





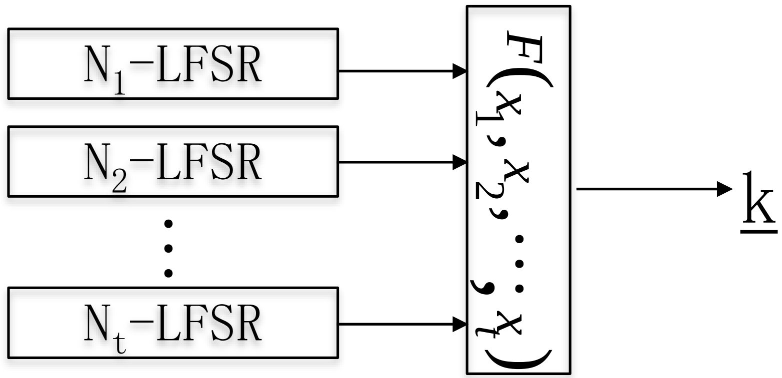
F的设计：两种典型手段

① 非线性组合生成器

② 钟控序列生成器

F的设计1. 非线性组合生成器

结构：



其中F是t元非线性布尔函数，一般要求：

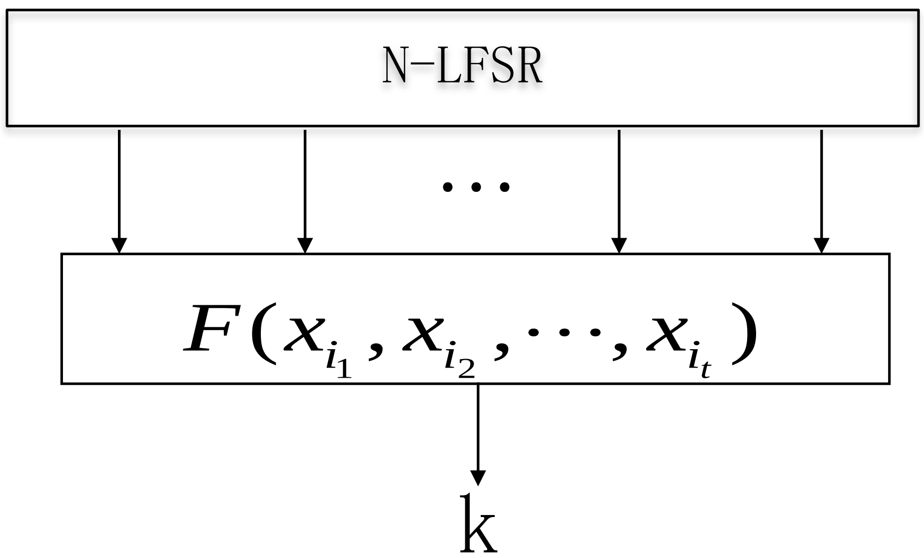
❶较高的非线性次数；❷是0,1平衡的

m个布尔变量的函数f的代数正规型是指乘积（与操作）的和（异或操作）。F的非线性次数等于代数正规型中最高次项的次数。

- 例：$F=1\oplus a\oplus b\oplus cd\oplus abcd$是代数正规型，非线性次数为4
- F的非线性次数越高，输出序列就具有**高线性复杂度**

非线性组合生成器：特殊情形

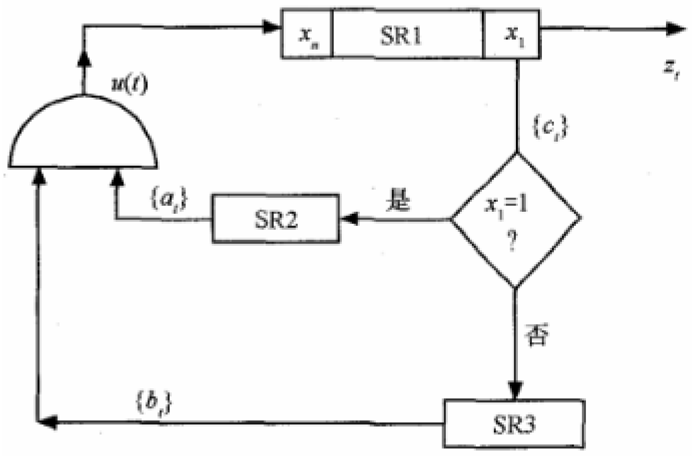
- 非线性滤波生成器，也称为前馈网络：



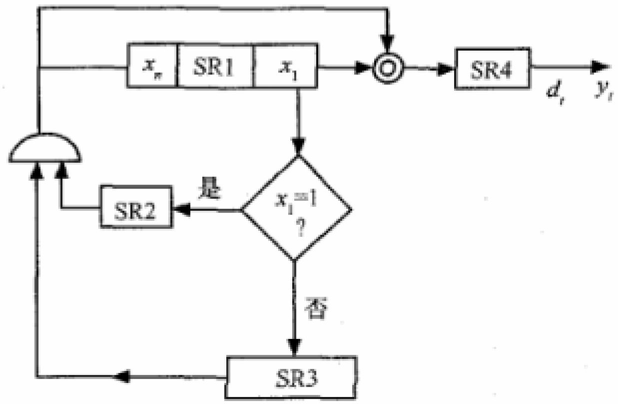
F的设计2. 钟控序列生成器

用一个寄存器序列作为时钟，控制另一寄存器序列（或自己控制自己）的运行，具有大的线性复杂度

结构



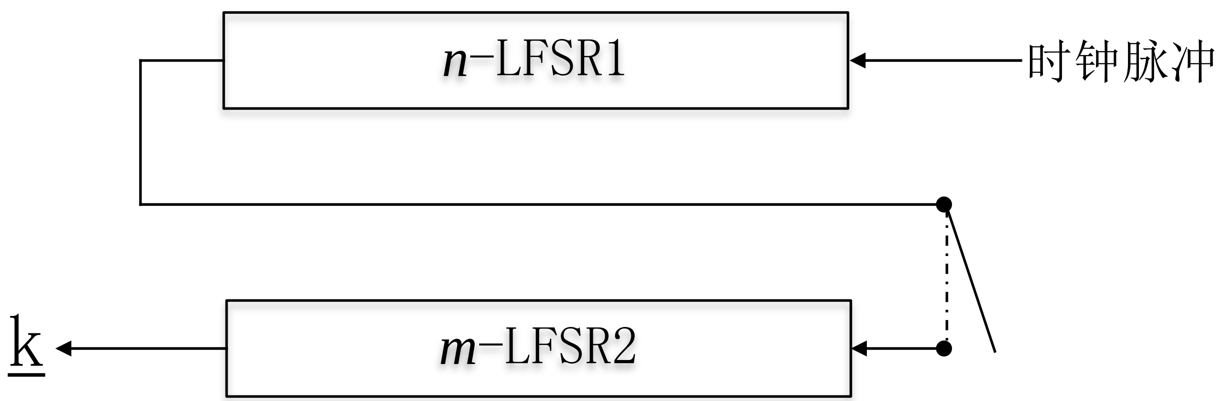
互控生成器模型 1



互控钟控生成器模型 2

钟控序列生成器：特殊情形

- “停走”(Stop-and-Go)生成器：



- 一个相关的结果：
- 若n-LFSR1与m-LFSR2都是m-序列生成器，且$n|m$或2^n-1为素数，则
- ①$p(\underline{k})=(2^n-1)(2^m-1)$，
- ②$L(\underline{k})$至少为$[m,n](2^{(m,n)}-1)$。

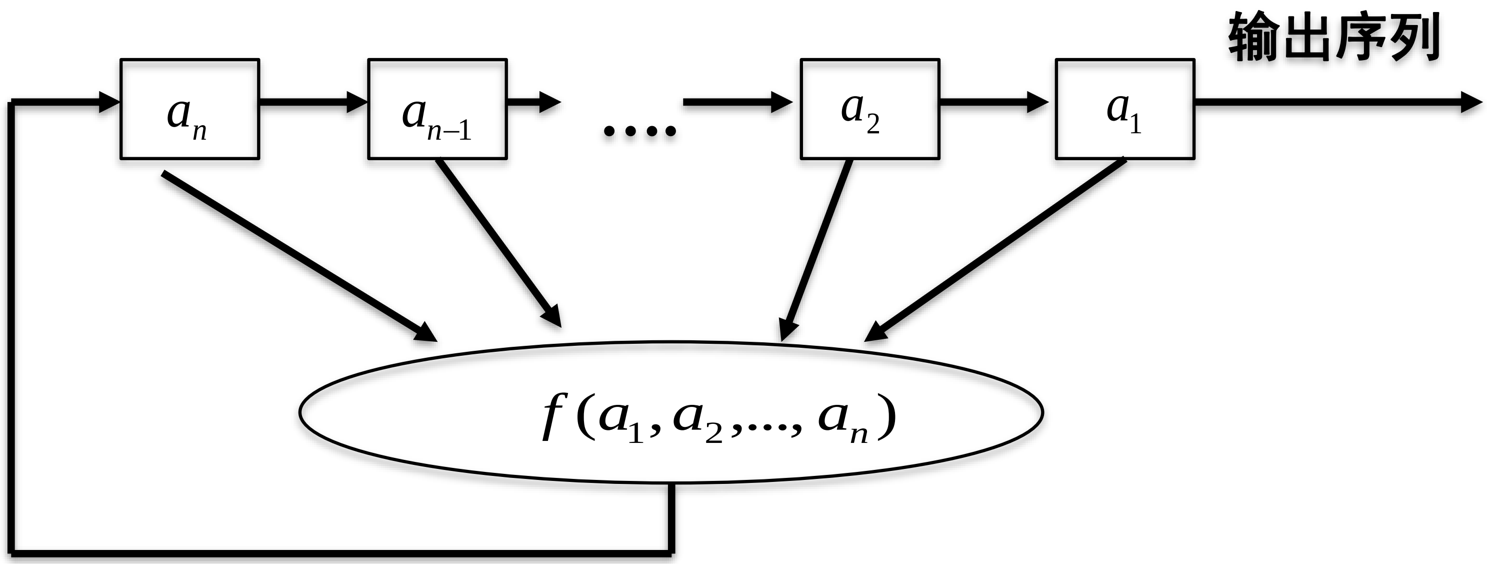
补充：LFSR的数学基础之二

- 定义
  - 多项式表示
  - 生成函数
  - 特征多项式
- m序列
  - 游程的概念
  - 随机性
- 线性反馈移位寄存器的综合
  - 线性复杂度
  - BM算法
- m序列密码的破译
- 几种序列生成器

线性反馈移位寄存器：LFSR

- 移位寄存器，是序列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
- 移位寄存器的反馈函数$f(a_1,a_2,\ldots,a_n)$是a_1，a_2，…，a_n的线性函数，则称之为**线性反馈移位寄存器LFSR**

（Linear Feedback Shift Register）



GF(2)上的N级反馈移位寄存器：构成

- **级**：每个存储器称为移位寄存器的一级，在任一时刻，这些级的内容构成该反馈移位寄存器的状态。
- **状态**：每个**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，用n长序列 $a_1,a_2,...,a_n$ 或n维向量$(a_1,a_2,...,a_n)$表示，其中a_i是第i级存储器的内容。

GF(2)上的N级反馈移位寄存器

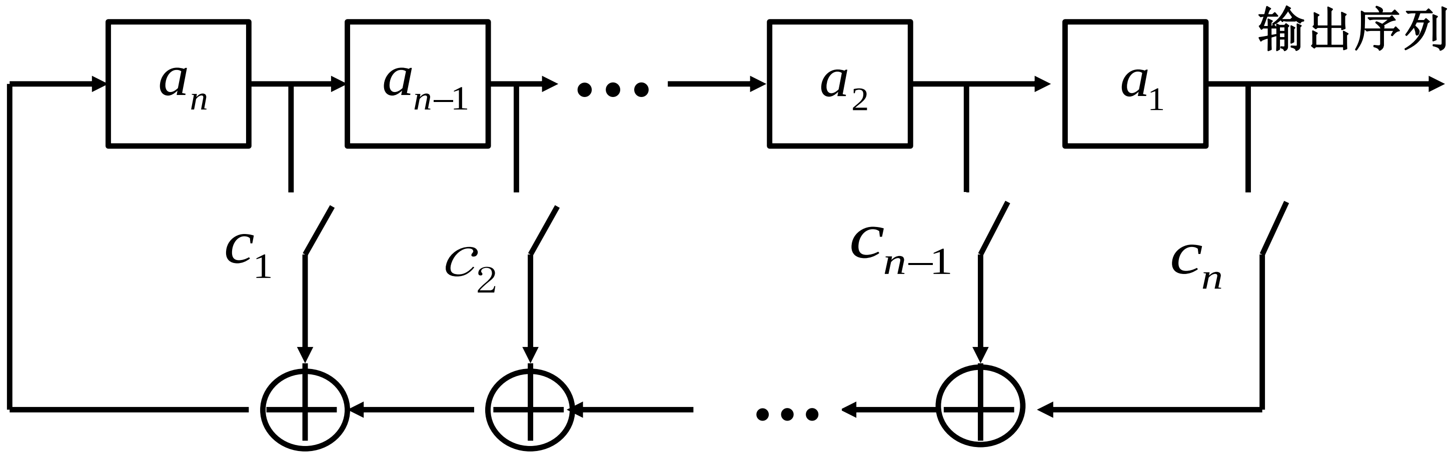
- 初始状态由用户确定，当第i个移位时钟脉冲到来时，每一级存储器a_i都将其内容向下一级a_{i-1}传递，并根据寄存器此时的状态$a_1,a_2,\ldots,a_n$计算$f(a_1,a_2,\ldots,a_n)$，作为下一时刻的a_n。
- 反馈函数$f(a_1,a_2,\ldots,a_n)$是n元布尔函数，即n个变元$a_1,a_2,\ldots,a_n$可以独立地取0和1值，最后的函数值也为0或1

线性反馈移位寄存器 —表示

- $f(a_1,a_2,\ldots,a_n)$是a_1，a_2，…，a_n的线性函数，f可写为

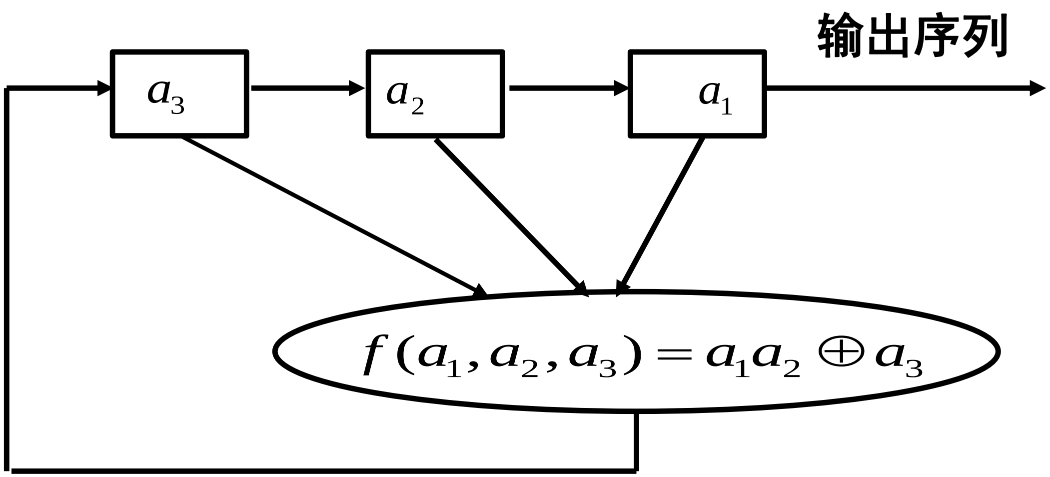
$$f(a_1,a_2,\ldots,a_n)=c_na_1\oplus c_{n-1}a_2\oplus\ldots\oplus c_1a_n,$$

- 其中常数c_i=0或1，$\oplus$是模2加法。
- c_i=0或1，可用开关的断开和闭合来实现



例子

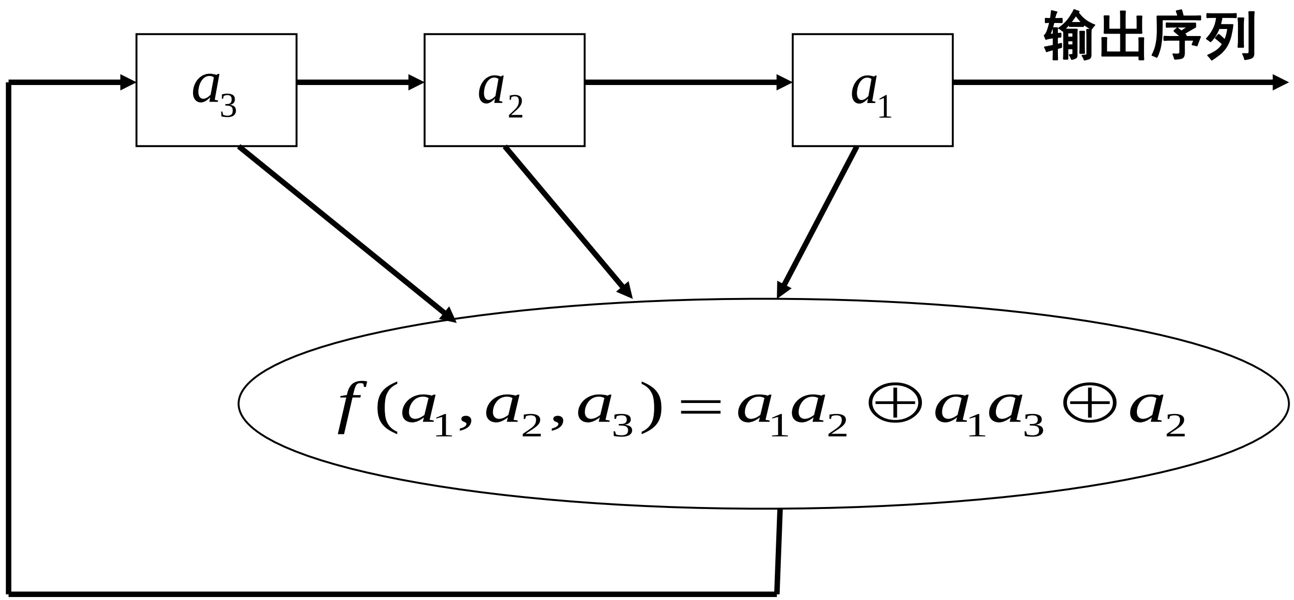
- 一个3级反馈移位寄存器，其初始状态为$(a_3,a_2,a_1)=(1,0,1)$，输出见下表，即输出序列为101110111011…，周期为4



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

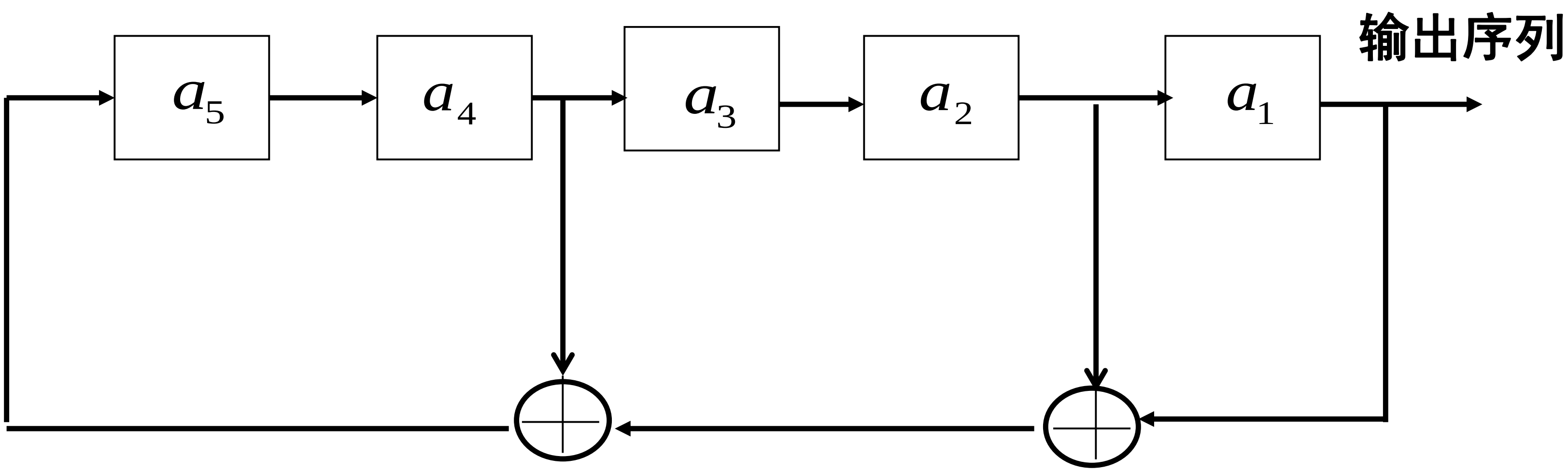
课堂练习-I

- 下图为3级LFSR，其初始状态为 $(a_3, a_2, a_1) = (1,1,0)$
- 试写出其输出序列，同时考虑如果初始状态不同，输出序列是否相同，序列的周期是否相同？



课堂练习-2

- 下图为一个5级LFSR，其初始状态为
  $$(a_5, a_4, a_3, a_2, a_1) = (1,1,0,1,0)$$
- 写出其一个周期的所有状态，及其输出序列



线性反馈移位寄存器

- 在LFSR中总是假定$c_1,c_2,...,c_n$中至少有一个不为0，否则$f(a_1,a_2,...,a_n)\equiv 0$，为什么？
- 若只有一个系数不为0，设仅有c_j不为0，实际上是一种延迟装置，为什么？
- **一般对于n级LFSR，总是假定$c_n=1$。**

线性反馈移位寄存器：研究结论

- LFSR输出序列的性质完全由其**反馈函数**决定。
- n级LFSR**最多有2^n个不同的状态**。若其初始状态为0，则其状态恒为0。若其初始状态非0，则其后继状态不会为0。
- 因此n级LFSR的状态周期小于等于2^n-1。其输出序列的周期与状态周期相等，也小于等于2^n-1。
- 只要选择**合适的反馈函数**便可使LFSR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**。

线性反馈移位寄存器

- 设计基础
  - 反馈函数的性质、要求
  - 多项式表示
  - 多项式环

线性移位寄存器的一元多项式表示

- LFSR的输出序列$\{a_i\}$满足递推关系

$$a_{n+k} = c_1 a_{n+k-1} \oplus c_2 a_{n+k-2} \oplus \cdots \oplus c_n a_k \qquad (1)$$

- 当k>=1时成立。写出来就是：

$$\begin{aligned}
a_{n+1} &= c_1 a_n \oplus c_2 a_{n-1} \oplus \cdots \oplus c_n a_1 \\
a_{n+2} &= c_1 a_{n+1} \oplus c_2 a_n \oplus \cdots \oplus c_n a_2 \\
a_{n+3} &= c_1 a_{n+2} \oplus c_2 a_{n+1} \oplus \cdots \oplus c_n a_3 \\
a_{n+4} &= c_1 a_{n+3} \oplus c_2 a_{n+2} \oplus \cdots \oplus c_n a_4 \\
&\vdots
\end{aligned}$$

特征多项式表示

在(1)式中两边同时加上 a_{n+k} 得到：

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

令$a_{n+k-1} = x$ 据其在输出序列 $\{a_1, \cdots, a_k, a_{k+1}, \cdots a_{n+k-1}, a_{n+k}\}$

中元素的左右位置，左移一位相当于乘以2，则有： $a_{n+k} = x^0 = 1$

$a_{n+k-1} = x$ $a_{n+k-2} = x^2$ $\cdots, a_k = x^n$

这种递推关系可以用一个一元高次多项式来表示：

$$p(x) = 1 + c_1 x + \cdots + c_{n-1} x^{n-1} + c_n x^n \qquad (*)$$

称这个多项式为**LFSR的特征多项式**。

LFSR与特征多项式

- 注意：LFSR与特征多项式是一一对应的，如果知道了LFSR的结构，可以写出它的特征多项式，同样可以根据特征多项式画出LFSR的结构。
- 设n级LFSR对应于递推关系(*)，由于$a_i \in GF(2)$ $(i=1,2,\ldots,n)$，所以共有2^n组初始状态，即有2^n个递推序列，其中非恒零的有2^n-1个，记2^n-1个非零序列的全体为$G(p(x))$

LFSR的数学基础之周期（略）

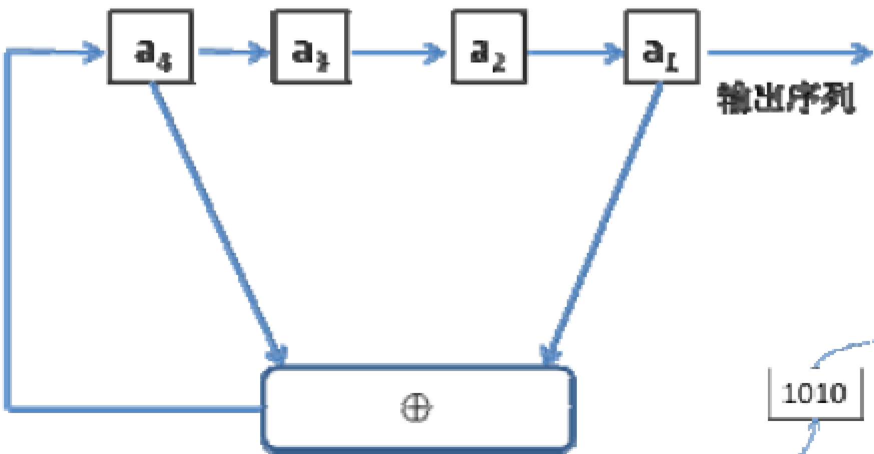
- 定理1：生成函数与特征函数生成的序列
- 定理2：多项式的整除的充要条件
- 定理3：多项式整除与序列周期的关系
- →*定理1-3：n级LFSR*输出序列的周期r不依赖于初始条件，而依赖于特征多项式$p(x)$
- 定理4：序列的周期与多项式的周期
- 定理5：序列最大周期的必要条件
- 定理6：序列最大周期的充分条件
- 定理7：m-序列的周期
- →P88

常用的本原多项式

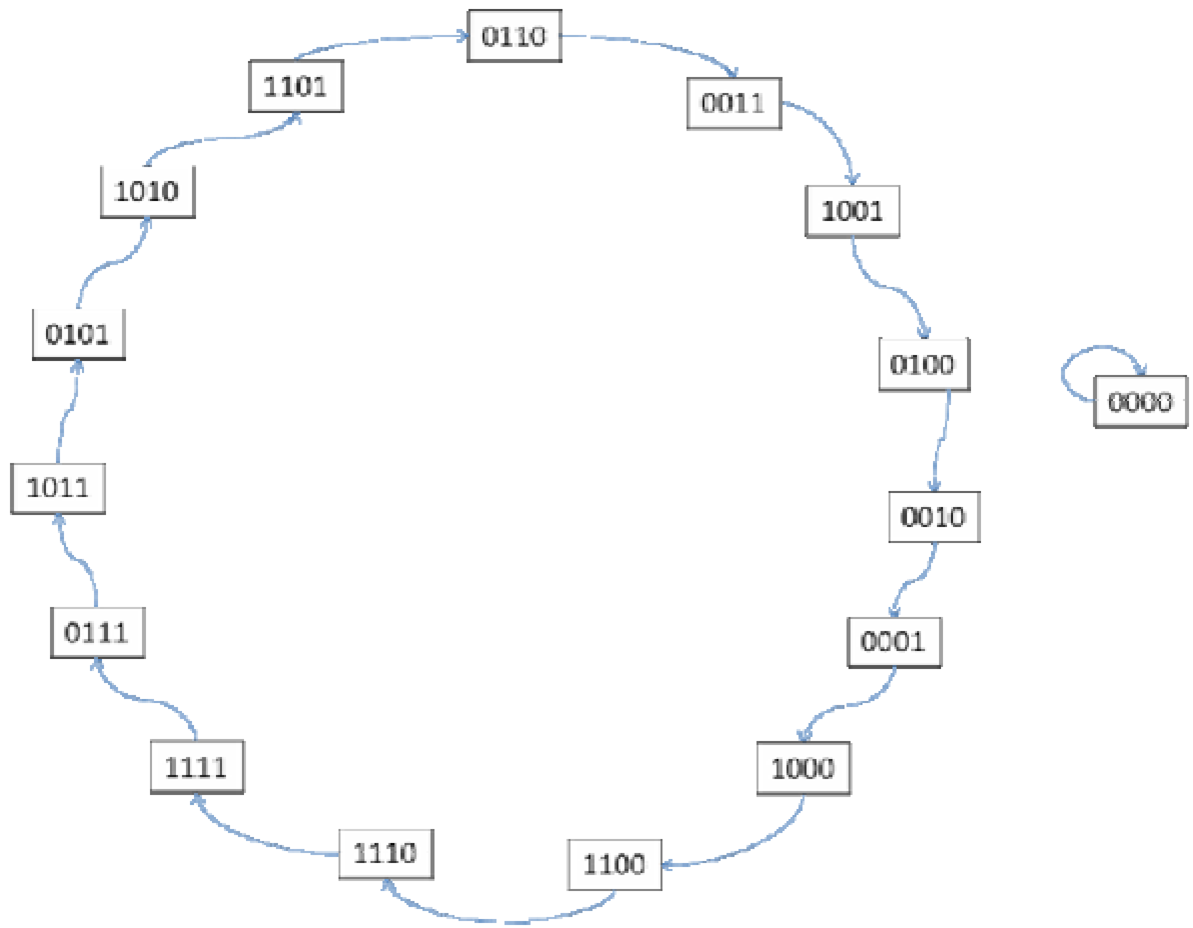
常用本原多项式

n	本原多项式		n	本原多项式	
	代　数　式	8 进数表示法		代　数　式	8 进数表示法
2	x^2+x+1	7	14	$x^{14}+x^{10}+x^6+x+1$	42103
3	x^3+x+1	13	15	$x^{15}+x+1$	100003
4	x^4+x+1	23	16	$x^{16}+x^{12}+x^3+x+1$	210013
5	x^5+x^2+1	45	17	$x^{17}+x^3+1$	400011
6	x^6+x+1	103	18	$x^{18}+x^7+1$	1000201
7	x^7+x^3+1	211	19	$x^{19}+x^5+x^2+x+1$	2000047
8	$x^8+x^4+x^3+x^2+1$	435	20	$x^{20}+x^3+1$	4000011
9	x^9+x^4+1	1021	21	$x^{21}+x^2+1$	10000005
10	$x^{10}+x^3+1$	2011	22	$x^{22}+x+1$	20000003
11	$x^{11}+x^2+1$	4005	23	$x^{23}+x^5+1$	40000041
12	$x^{12}+x^6+x^4+x+1$	10123	24	$x^{24}+x^7+x^2+x+1$	100000207
13	$x^{13}+x^4+x^3+x+1$	20033	25	$x^{25}+x^3+1$	200000011

例子：最长周期

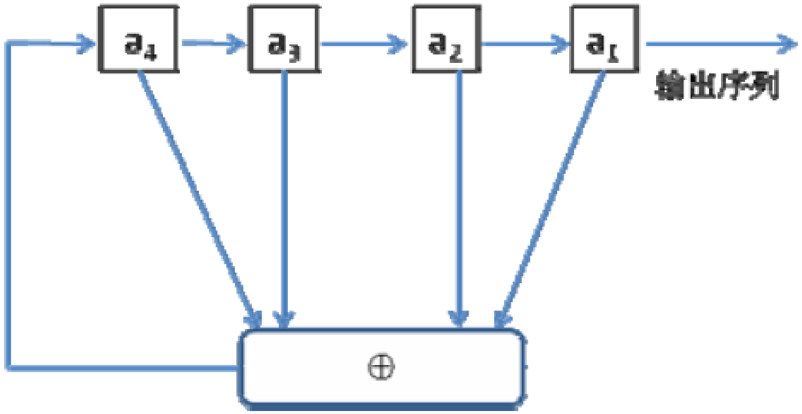


a_4
a_3
a_2
a_1
输出序列
$\oplus$



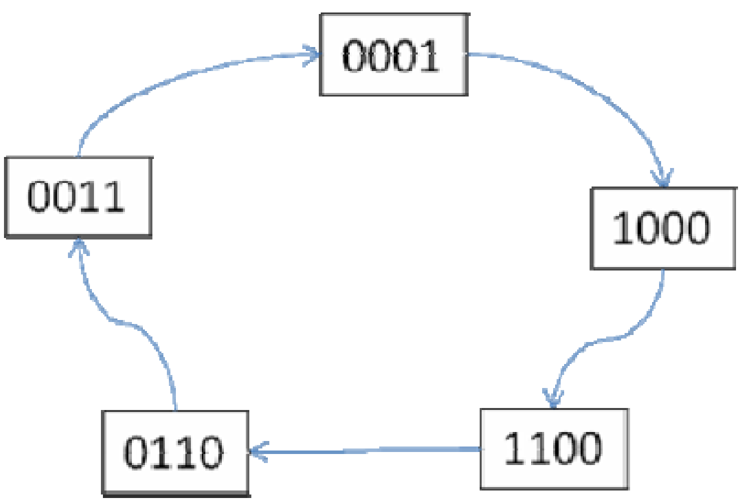
0110
0011
1001
0100
0010
0001
1000
1100
1110
1111
0111
1011
0101
1010
1101
0000

例子：多个循环

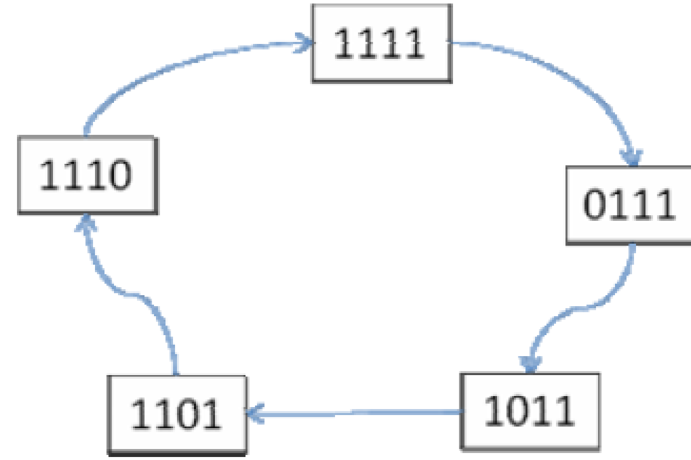


- 联接多项式 $x^4+x^3+x^2+x+1$

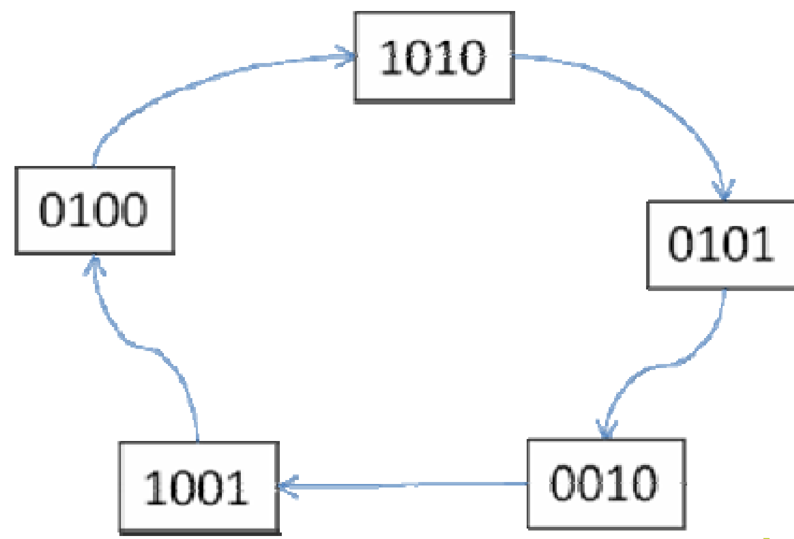
- 取初态 $(a_1, a_2, a_3, a_4)=(0,0,0,1)$



- 取初态 $(a_1, a_2, a_3, a_4)=(1,1,1,1)$



- 取初态 $(a_1, a_2, a_3, a_4)=(1,0,1,0)$



本原多项式的个数

- $\{a_i\}$为m序列的关键，在于$p(x)$为本原多项式，n次本原多项式的个数为 $$\frac{\phi(2^n-1)}{n}$$

其中ϕ为欧拉函数。

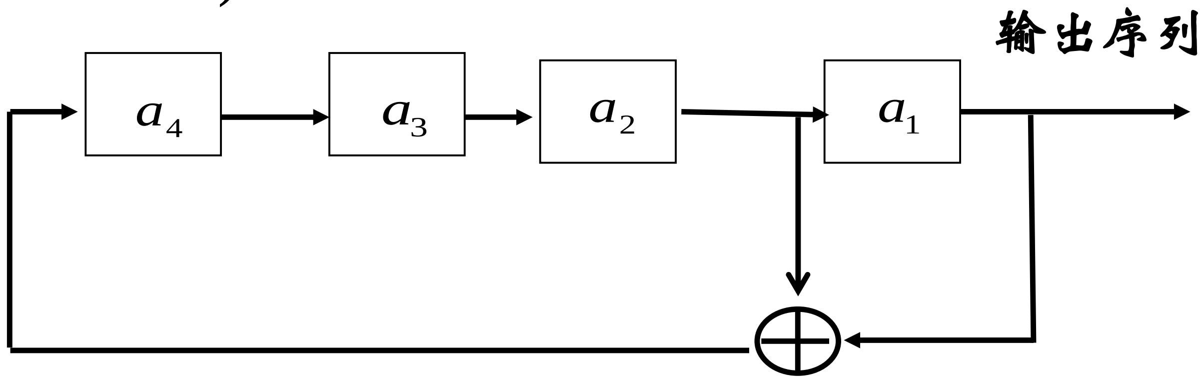
- 已经证明，对于任意的正整数n，至少存在一个n次本原多项式。所以对于任意的n级$LFSR$，至少存在一种连接方式使其输出序列为m序列。

例子

- 设$p(x)=x^4+x+1$，由于$p(x)|(x^{15}-1)$，但不存在小于15的常数l，使得$p(x)|(x^l-1)$，所以$p(x)$的阶为15。$p(x)$的不可约性可由x，$x+1$，x^2+x+1都不能整除$p(x)$得到，所以$p(x)$是本原多项式。
- 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…
- 状态序列为：1001，0100，0010，0001，1000，1100，1110，1111，0111，1011，0101，1010，1101，0110，0011，1001，0100，0010，0001……
- 可见，其周期为$2^4-1=15$，输出序列为m序列。

例子

$$p(x)=(1+x^3+x^4)$$



$$a_4a_3a_2a_1=1001$$

$$(\underline{10}01\underline{1010}111\underline{10}00)$$

- 游程：序列中取值相同的那些相继的元素合称为一个游程，或序列中连续不变的序列的数目
- 游程长度：一个游程中元素的个数
- 游程的总数为8，分别为 1，00，11，0，1，0，1111，0000。
- 其中有一半的游程长度为2，长度为2的游程为四分之一，有一个长度为4的游程和一个长度为3的游程 ⟹ 找出来

随机序列（略）

- 目的和重要性、要求
- 生成方式、检测方法
- 随机性公设

测试

- 随机分布
- 假设检验
- 序列测试（双比特测试）
- 游程测试
- 扑克测试
- 自相关测试
- SP 800-22 R1

自相关测试

用来检测序列s与其移位后所形成的序列之间的相关性

令d为固定整数， $1 \leqslant d \leqslant \lfloor n/2 \rfloor$。序列s与s发生d移位后所形成的序列中的不同比特数为

$$A(d)=\sum_{i=0}^{n-d-1} s_i \oplus s_{i+d}$$

所用统计量为

$$X_5=2\left[A(d)-\frac{n-d}{2}\right]\Big/\sqrt{n-d}$$

若n-d ⩾ 10，则该统计量近似服从N(0, 1)

SP 800-22 R1

美国国家标准与技术局商业部SP 800-22 Revision1

2008年8月

选取了15种统计测试

计算P-value：产生比测试序列更不随机的序列的概率

- 若P-value ≥ α, 则接受假设，序列是随机的
- 若P-value < α, 则拒绝假设，序列是非随机的

随机行走检测	模板检测		复杂性/可压缩性
频率	游程	Maurer通用统计	二进制矩阵秩
块内频率	块内”1”最长游程	串行测试	傅立叶谱
累积和	非重叠模板匹配	近似熵	~~LZ压缩~~
随机偏离	重叠模板匹配		线性复杂度
随机偏离变量			

随机序列的特性--自相关函数

- 定义：$GF(2)$上周期为T的序列$\{a_i\}$的**自相关函数**定义为
- $R(\tau)=(1/T)\sum(-1)^{ak}(-1)^{ak+\tau},0\leq\tau\leq T-1$
- 定义中的和式表示序列$\{a_i\}$与$\{a_{i+\tau}\}$（序列{ai}向后平移τ位得到）在一个周期内对应位相同的位数与对应位不同的位数之差。当$\tau=0$时，$R(\tau)=1$；当$\tau\neq0$时，称$R(\tau)$为**异相自相关函数**。

周期序列的自相关函数计算

- 周期p序列$\underline{a}=a_0a_1a_2\cdots$的自相关函数定义如下：

$$C_{\underline{a}}(t)=\frac{1}{p}\sum_{i=0}^{p-1}(-1)^{a_i}(-1)^{a_{t+i}}=\frac{1}{p}\sum_{i=0}^{p-1}(-1)^{a_i\oplus a_{t+i}}$$

- **自相关函数计算**：对给定的周期序列$\underline{a}$，
- ①找出$\underline{a}$的周期段：$a_0a_1a_2\cdots a_{p-1}$
- ②计算：

$$(-1)^{a_0}(-1)^{a_1}(-1)^{a_2}\cdots(-1)^{a_{p-1}}$$

（左环移t位）$(-1)^{a_t}(-1)^{a_{t+1}}(-1)^{a_{t+2}}\cdots(-1)^{a_{t-1}}$

对位相乘后再相加，即得$pC_{\underline{a}}(t)$

随机性公设

- Golomb对伪随机周期序列提出了应满足的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
  - 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。
  - ③ 异相自相关函$R(\tau)$数是一个常数

随机性公设的含义

- ①说明，$\{a_i\}$中0与1出现的概率基本上相同
- ②说明，0与1在序列中每一位置上出现的概率相同
- ③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
- 从密码系统的角度看，伪随机序列还应满足下面的条件：
  - ① $\{a_i\}$的周期相当大。
  - ② $\{a_i\}$的确定在计算上是容易的。
  - ③ 由密文及相应的明文的部分信息，不能确定整个$\{a_i\}$。

m序列的特性

- 一个n-LFSR（给定结构常数）具有“由一个短的种子产生一个长的序列”的功能：
- **以短的种子作为初态，产生的输出序列可以任意长！**
- 以上表明，任一n-LFSR都初步具有作为一个KG的资格；但从作为KG的效用来讲，自然更希望所使用的n-LFSR进一步是m-序列生成器。

线性复杂度、BM算法

- 概念：
  - 无穷序列$s=s_0,s_1,,s_2,\cdots$
  - 有限序列$s_n=s_0,s_1,,s_2,\cdots,s_{n-1}$
- 若某LFSR在某初始状态下输出序列的前n项为s_n的话，则称该LFSR生成有限序列s_n
- 可以设想用LFSR来描述序列的复杂程度
- LFSR是线性的，因此它描述的是线性的复杂程度

LFSR 的综合

- 前面讲过，m-序列是满足Golomb三条随机性公设的PN序列，但其不可以作为一个序列密码的密钥序列。
- 因为：对m-序列，知道其少量的比特以后是可以预测的！下面看怎么样仅凭已知的少量比特，找出整个序列所满足的线性递推关系。
- 一般地，从正反两个方面分为分析与综合：

$$\text{n-FSR的结构常数+初态} \underset{\text{综合}}{\overset{\text{分析}}{\rightleftarrows}} \text{n-LFSR序列}$$

LFSR 的综合方法

- LFSR的综合问题：**根据序列的少量bit，求出整个序列所满足的线性递推关系**
- 一个自然的想法是：先假定线性递推关系，然后由序列的各项应该满足的关系，得出线性方程组。这样的方法有其不易之处在于：
  - ①不容易确定所适用的LFSR的级数n，从而就不能导致恰当规模的线性方程组；
  - ②当上述的n很大时，求解相应规模的线性方程组也很困难。
- 对于LFSR的综合问题已经出现了著名的解法：**Berlekamp-Massey**迭代算法，简称B-M算法

B-M算法描述

Input：$S^N=a_0a_1a_2\cdots a_{N-1}$

Step1：置$f_0(x)=1$，$L_0=0$（初值）

Step2：设$<f_i(x),L_i>$，$i=0,1,2,\ldots,n$ （$0\le n<N$）均已求出，且 $L_0\le L_1\le L_2\le\cdots\le L_n$。设 $f_n(x)=1+c_1^{(n)}x+c_2^{(n)}x^2+\cdots+c_{L_n}^{(n)}x^{L_n}$，

由此计算 $d_n=a_n+c_1^{(n)}a_{n-1}+c_2^{(n)}a_{n-2}+\cdots+c_{L_n}^{(n)}a_{n-L_n}$ 。

Step3：当$d_n=0$时，取$f_{n+1}(x)=f_n(x),L_{n+1}=L_n$。

当$d_n=1$时，若$L_n=0$，则取$f_{n+1}(x)=x^{n+1}+1,L_{n+1}=n+1$；

否则，找出$m(0\le m<n)$使$L_m<L_{m+1}=L_{m+2}=\cdots=L_n$，取$f_{n+1}(x)=f_n(x)+x^{n-m}f_m(x)$，$L_{n+1}=\max\{L_n,n+1-L_n\}$。

对于$n=0,1,2,\cdots$，重复**Step2**与**Step3**，直至$n=N-1$

Output：$<f_N(x), L_N>$

举例

- 输入：$S^8=10101111$
- 输出：$<1+x^3+x^4,4>$
- 过程：

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		

有关B-M算法

- **定理1.** 应用B-M算法，若以N长序列S^N为输入，得到输出<$fN(x)$,LN>，则
- (1)以$f_N(x)$为联接多项式的L_N-LFSR是产生SN的最短LFSR，且当 $L_N \le \frac{N}{2}$ 时，迭代至第$2L_N$步就得到最终输出，即：$< f_{2L_N}(x), L_{2L_N} > = < f_N(x), L_N >$
- (2)当$L_N \le \frac{N}{2}$时，产生S^N的最短LFSR只是上述一个；当 $L_N > \frac{N}{2}$时，产生S^N的最短LFSR一共有 $2^{2L_N - N}$个。
- 由上述定理知，在前面的例子中，以$f_8(x)=1+x^3+x^4$为联接多项式的4-LFSR是唯一的产生$S^8=10101111$的最短LFSR。

考虑：

❶ $S^6=101011 \leftrightarrow f_6(x)=1+x^2+x^3$

❷ $S^{1+N} = 1\overbrace{0\cdots0}^{N} \leftrightarrow f_{1+N}(x) = 1$ 如何解释？

- 其实：
- 对❶，由于$L_6=4$，故4-LFSR [0,1,1,0] 生成S^6；
- 对❷，由于$L_{1+N}=1$，故1-LFSR[0]生成S^2（规定：f(x)=1产生全零序列）

有关B-M算法

- 对于周期序列，也可应用B-M算法求出产生它的最短LFSR的联接多项式，不过须注意：**一定是针对两个周期段去求才正确！**

- **定理2.** 对周期为p的序列$\underline{a}=a_0a_1a_2\cdots$,

- (1) 应用B-M算法于$S^{2p}=a_0a_1\cdots a_{p-1}a_0a_1\cdots a_{p-1}$求出$<f_{2p}(x),L_{2p}>$时，$f_{2p}(x)$的次数必为$L_{2p}$，且以$f_{2p}(x)$为联接多项式的$L_{2p}$-$LFSR$是唯一的产生$\underline{a}$的最短LFSR。

- (2) $< f_{2L_{2p}}(x), L_{2L_{2p}} >=< f_{2p}(x), L_{2p} >$

应用

求产生序列a的最低次多项式，这里

(1) $\underline{a}=111001$，(2) $\underline{a}=(111001)^{\infty}$

解：

n	d_n	f_n	L_n	m	f_m
0	1	1	0		
1	0	$1+x$	1		
2	0	$1+x$	1		
3	1	$1+x$	1	0	1
4	1	$1+x+x^3$	3	3	$1+x$
5	0	$1+x^2+x^3$	3		
6	1	$1+x^2+x^3$	3	3	$1+x$
7	0	$1+x^2+x^4$	4		
8～12	0	$1+x^2+x^4$	4		

可见答案为：(1) $1+x^2+x^3$；(2) $1+x^2+x^4$

线性复杂度概念

- 定义. 能够产生（有限长或周期）序列$\underline{a}$的最短LFSR的**级数**称为$\underline{a}$的**线性复杂度**，记为$L(\underline{a})$；约定：$L(0)=0$
- 若对序列a应用B-M算法产生的输出为$<f(x),L>$，则$L(a)=L$

线性复杂度的定义

- 无穷二元序列s的线性复杂度L(s)定义为：
  - 若s为零序列，即s=0,0,0,⋯，则L(s)=0
  - 若没有LFSR能够生成s，则L(s)=∞
  - 否则，L(s)就为生成s的最短LFSR的长度
- 有限二元序列s_n的线性复杂度$L(s_n)$定义为：
  - 生成以s_n为开始的二元序列的最短LFSR的长度
- 设L_N表示子序列$s_N=s_0,s_1,,s_2,\cdots,s_{N-1}$的**线性复杂度**，

则序列$L_1,L_2,\cdots$称为s的**线性复杂度轮廓**

线性复杂度的性质

- 设s和t为二元序列
  - 对任意$n\geq 1$，子序列s_n的线性复杂度满足$0\leq L(s_n)\leq n$
  - 若s的周期为N，则$L(s)\leq N$
  - $L(s_n)=0$，当且仅当s_n是长度为n的零序列
  - $L(s_n)=n$，当且仅当$s_n=0,0,0,\cdots,0,1$
  - $L(s\oplus t)\leq L(s)+L(t)$
- 线性复杂度轮廓的性质：
  - 若$j>i$，则$L_j\geq L_i$
  - 若$L_{N+1}>L_N$，则$L_N\leq N/2$
  - 若$L_{N+1}>L_N$，则$L_{N+1}+L_N=N+1$

范明钰 2019年本科密码学

m序列密码的破译(略)

- 上面说过，有限域上的二元加法序列密码是目前最为常用的序列密码体制，设滚动密钥生成器是线性反馈移位寄存器，产生的密钥是序列。又设和是序列中两个连续的长向量，其中

$$S_h=\begin{pmatrix}a_h\\a_{h+1}\\\vdots\\a_{h+n-1}\end{pmatrix},\quad S_{h+1}=\begin{pmatrix}a_{h+1}\\a_{h+2}\\\vdots\\a_{h+n}\end{pmatrix}$$

- 设序列{a_i}满足线性递推关系：

$$a_{h+n}=c_1a_{h+n-1}\oplus c_2a_{h+n-2}\oplus\cdots\oplus c_na_h$$

- 可表示为

$$\begin{pmatrix}a_{h+1}\\a_{h+2}\\\vdots\\a_{h+n}\end{pmatrix}=\begin{pmatrix}0&1&0&\cdots&0\\0&0&1&\cdots&0\\\vdots&&&&\\c_n&c_{n-1}&c_{n-2}&\cdots&c_1\end{pmatrix}\begin{pmatrix}a_h\\a_{h+1}\\\vdots\\a_{h+n-1}\end{pmatrix}$$

m序列密码的破译

- 或$S_{h+1}=M\cdot S_h$，其中

$$M=\begin{pmatrix} 0 & 1 & 0 & \cdots 0 \\ 0 & 0 & 1 & \cdots 0 \\ \vdots & & & \\ c_n & c_{n-1} & c_{n-2} & \cdots c_1 \end{pmatrix}$$

- 又设敌手知道一段长为2n的明密文对，即已知

$$x=x_1x_2\cdots x_{2n} \qquad y=y_1y_2\cdots y_{2n}$$

- 于是可求出一段长为2n的密钥序列 $z=z_1z_2\cdots z_{2n}$
- 其中 $z_i=x_i\oplus y_i=x_i\oplus(x_i\oplus z_i)$ ，由此可推出线性反馈移位寄存器连续的n+1个状态：

$$S_1=(z_1z_2\cdots z_n)\overset{\text{记为}}{=}(a_1a_2\cdots a_n)$$

$$S_2=(z_2z_3\cdots z_{n+1})\overset{\text{记为}}{=}(a_2a_3\cdots a_{n+1})$$

$$\cdots$$

$$S_{n+1}=(z_{n+1}z_{n+2}\cdots z_{2n})\overset{\text{记为}}{=}(a_{n+1}a_{n+2}\cdots a_{2n})$$

m序列密码的破译

◆做矩阵 $X=\begin{pmatrix} S_1 & S_2 & \cdots & S_n \end{pmatrix}$

◆而

$$\begin{pmatrix} a_{n+1} & a_{n+2} & \cdots & a_{2n} \end{pmatrix}=\begin{pmatrix} c_n & c_{n-1} & \cdots & c_1 \end{pmatrix}\begin{pmatrix} a_1 & a_2 & \cdots & a_n \\ a_2 & a_3 & \cdots & a_{n+1} \\ \cdots & & & \\ a_n & a_{n+1} & \cdots & a_{2n-1} \end{pmatrix}$$

$$=\begin{pmatrix} c_n & c_{n-1} & \cdots & c_1 \end{pmatrix}X$$

◆若X可逆，则 $\begin{pmatrix} c_n & c_{n-1} & \cdots & c_1 \end{pmatrix}=\begin{pmatrix} a_{n+1} & a_{n+2} & \cdots & a_{2n} \end{pmatrix}X^{-1}$

◆ 下面证明X的确是可逆的

m序列密码的破译

- 因为X是由$S_1,S_2,\dots,S_n$作为列向量，要证X可逆，只要证明这n个向量线性无关。
- 由序列递推关系：$a_{h+n}=c_1a_{h+n-1}\oplus c_2a_{h+n-2}\oplus\cdots\oplus c_na_h$
- 可推出向量的递推关系：

$$S_{h+n}=c_1S_{h+n-1}\oplus c_2S_{h+n-2}\oplus\cdots\oplus c_nS_h=\sum_{i=1}^{n}c_iS_{h+n-i}\pmod 2$$

- 设$m(m\le n+1)$是使$S_1,S_2,\dots,S_m$线性相关的最小整数，即存在不全为0的系数$l_1,l_2,\dots,l_m$，其中不妨设$l_1=1$，使得

$$S_m+l_2S_{m-1}+l_3S_{m-2}+\cdots+l_mS_1=0$$

- 即

$$S_m=l_mS_1+l_{m-1}S_2+\cdots+l_2S_{m-1}=\sum_{j=1}^{m-1}l_{j+1}S_{m-j}$$

- 对于任一整数i有

$$\begin{aligned}S_{m+i}&=M^iS_m=M^i\left(l_mS_1+l_{m-1}S_2+\cdots+l_2S_{m-1}\right)\\&=l_mM^iS_1+l_{m-1}M^iS_2+\cdots+l_2M^iS_{m-1}\\&=l_mS_{i+1}+l_{m-1}S_{i+2}+\cdots+l_2S_{m+i-1}\end{aligned}$$

m序列密码的破译

- 设$m(m\le n+1)$是使$S_1,S_2,\ldots,S_m$线性相关的最小整数，即存在不全为0的系数$l_1,l_2,\ldots,l_m$，其中不妨设$l_1=1$，使得

$$S_m + l_2 S_{m-1} + l_3 S_{m-2} + \cdots + l_m S_1 = 0$$

- 即 $$S_m = l_m S_1 + l_{m-1} S_2 + \cdots + l_2 S_{m-1} = \sum_{j=1}^{m-1} l_{j+1} S_{m-j}$$
- 对于任一整数i有

$$\begin{aligned} S_{m+i} &= M^i S_m = M^i \left(l_m S_1 + l_{m-1} S_2 + \cdots + l_2 S_{m-1} \right) \\ &= l_m M^i S_1 + l_{m-1} M^i S_2 + \cdots + l_2 M^i S_{m-1} \\ &= l_m S_{i+1} + l_{m-1} S_{i+2} + \cdots + l_2 S_{m+i-1} \end{aligned}$$

- 由此又推出密钥流的递推关系：

$$a_{m+i} = l_2 a_{m+i-1} \oplus l_3 a_{m+i-2} \oplus \cdots \oplus l_m a_{i+1}$$

- 即密钥流的级数小于m。若m≤n，则得出密钥流的级数小于n，矛盾。所以m=n+1，从而矩阵X必是可逆的。

m序列密码的破译

◆例：设敌手得到密文串101101011110010和相应的明文串011001111111001，因此可计算出相应的密钥流为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文串中的前10个比特和明文串中的前10个比特建立如下方程

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

◆即：

$$(0\ \ 1\ \ 0\ \ 0\ \ 0)=(c_5\ \ c_4\ \ c_3\ \ c_2\ \ c_1)\begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

而：

$$\begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}^{-1}=\begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

◆从而得到

$$(c_5\ \ c_4\ \ c_3\ \ c_2\ \ c_1)=(0\ \ 1\ \ 0\ \ 0\ \ 0)\begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

◆所以密钥流的递推关系为

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

$$a_{i+5}=c_5a_i\oplus c_2a_{i+3}=a_i\oplus a_{i+3}$$

几种经典序列生成器

① Geffe序列生成器

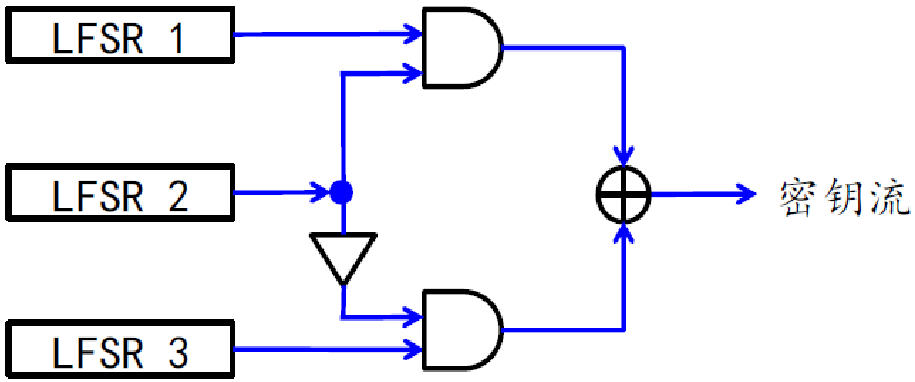
② J-K触发器

③ *Pless*生成器

④ 钟控序列生成器

GEFFE序列生成器

- 由3个互素的LFSR组成，其中LFSR2作为控制生成器使用：



Geffe序列生成器图

- 非线性组合函数：

$$f(x_1, x_2, x_3) = x_1x_2 \oplus (1+x_2) x_3 = x_1x_2 \oplus x_2x_3 \oplus x_3$$

- 密钥流周期 $(2^{L_1}-1) \cdot (2^{L_2}-1) \cdot (2^{L_3}-1)$

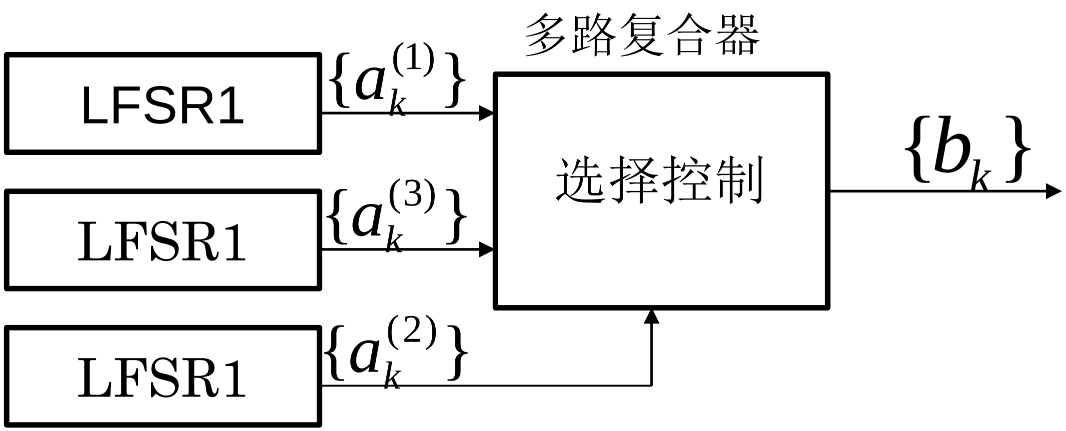
- 线性复杂度 $L=L_1L_2+L_2L_3+L_3$

GEFFE序列生成器

- Geffe序列生成器也可以表示为下图的形式，其中LFSR1和LFSR3作为多路复合器的输入，LFSR2控制多路复合器的输出。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序列的周期为

$$\prod_{i=1}^{3}\left(2^{n_i}-1\right)$$

- 线性复杂度为$\left(n_1+n_3\right)n_2+n_3$



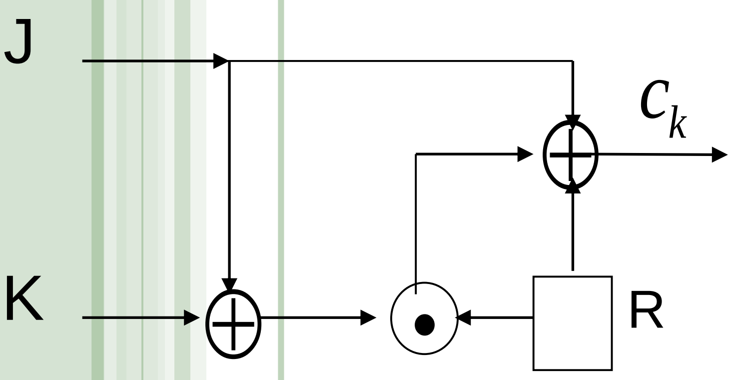
- Geffe序列的周期实现了极大化，且0与1之间的分布大体上是平衡的。

J-K触发器

- J-K触发器如下图所示，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

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

- 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。

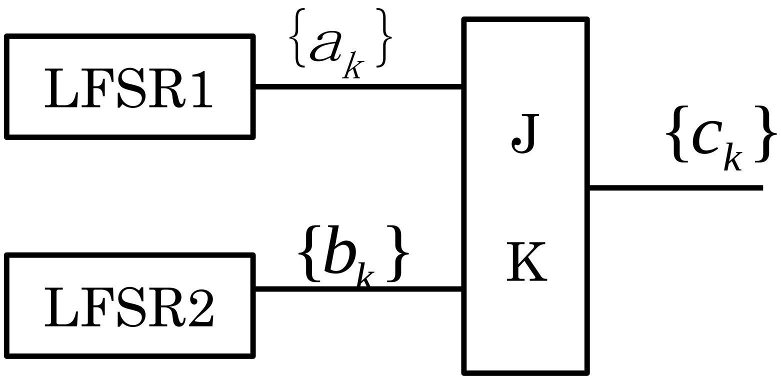


J-K触发器真值表

J	K	输出
0	0	
0	1	0
1	0	1
1	1	

利用J-K触发器组成非线性序列生产器

利用J-K触发器的非线性序列生成器



利用J-K触发器组成非线性序列生产器

- 令驱动序列$\{a_k\}$和$\{b_k\}$分别为m级和n级m序列，则有

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

- 如果令$c_{-1}=0$，则输出序列的最初3项为

$$c_0 = a_0$$

$$c_1 = (a_1 + b_1 + 1) a_0 + a_1$$

$$c_2 = (a_2 + b_2 + 1)\ ((a_1 + b_1 + 1) a_0 + a_1) + a_2$$

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

- 例：令$m=2,n=3$，两个驱动m序列分别为
- $\{a_k\}=0,1,1,\ldots$和 $\{b_k\}=1,0,0,1,0,1,1,\cdots$
- 于是，输出序列$\{c_k\}$是$0,1,1,0,1,0,0,1,1,1,0,1,0,1,0,0,1,0,0,1,0,\ldots$，其周期为$(2^2-1)(2^3-1)=21$。

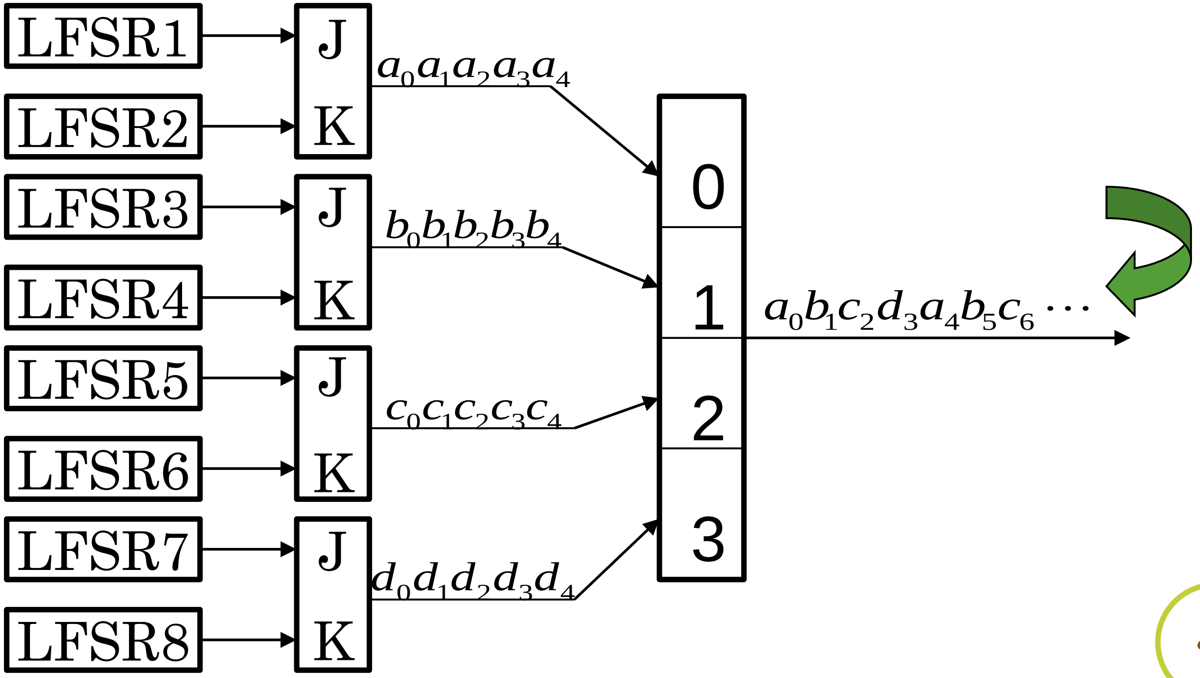
PLESS生成器

◆由表达式$c_k=(a_k+b_{k+1})c_{k-1}+a_k$可得 $c_k=\begin{cases} a_k, & c_{k-1}=0 \\ \overline{b_k}, & c_{k-1}=1 \end{cases}$

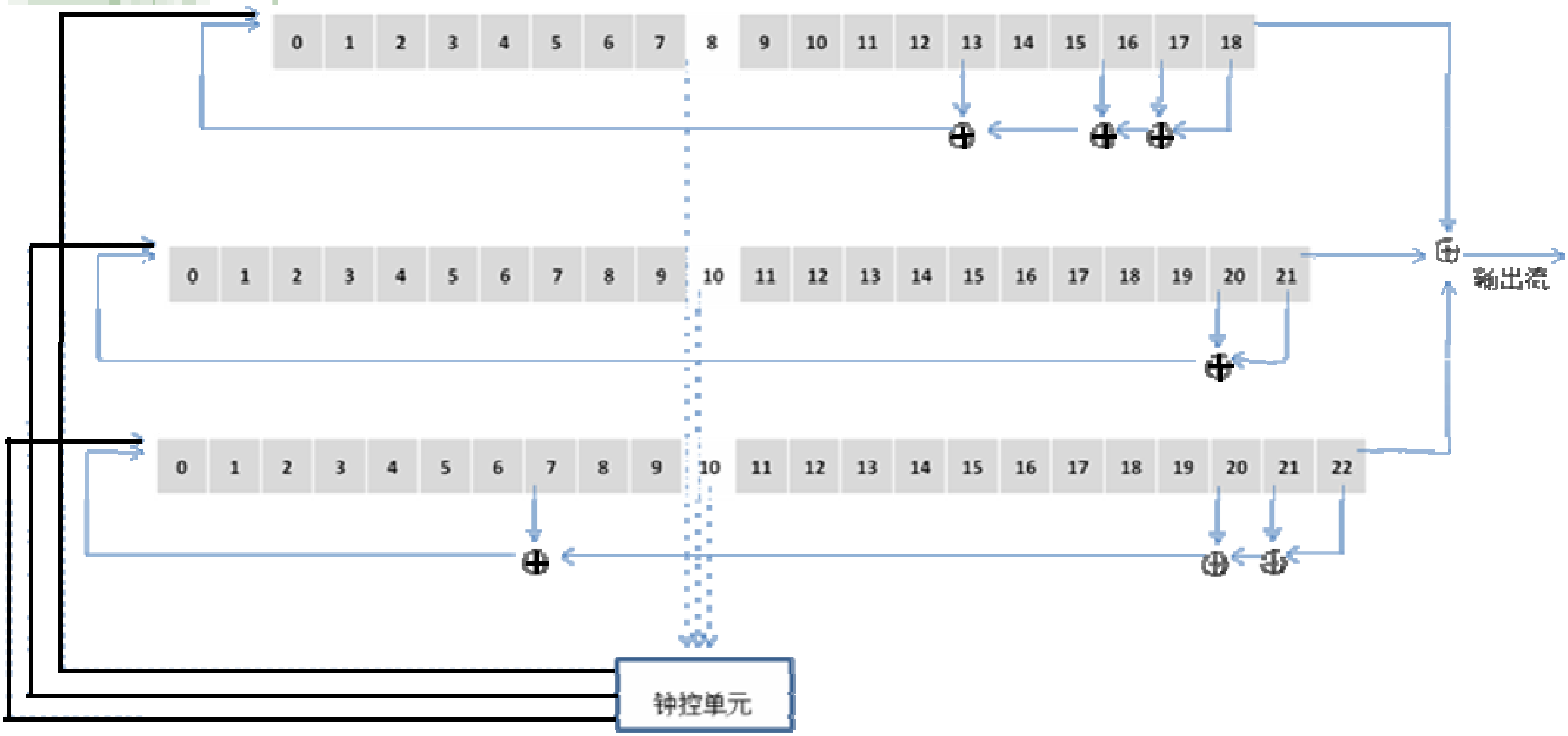
◆因此，如果知道$\{c_k\}$中相邻位的c_{k-1}和c_k，就可推断出a_k和b_k中的一个。而一旦知道足够多的这类信息，就可分析得到$\{a_k\}$和$\{b_k\}$。为了克服上述缺点，Pless提出了由多个J-K触发器序列驱动的多路复合序列方案，称为Pless生成器。

◆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。假定在时刻t输出第$t(mod\ 4)$个单元，则输出序列为：

$a_0b_1c_2d_3a_4b_5c_6\cdots$



例子：A5算法



- GSM 语音消息被转换成一系列的帧，每帧有228 bit，用A5算法加密。
- A5是一种典型的基于LFSR的流密码，由3个移位寄存器组成，是一种集互控和停走于一体的钟控模型。

A5算法

- 三个线性移位寄存器分别记为LFSR1、LFSR2 和LFSR3，记LFSRi中第 j1,…, jl比特为LFSRi[j1,…, jl]。LFSR1[8], LFSR2[10],LFSR3[10]为钟控单元。
- A5的钟控机制是：如果在某一时刻钟控单元中三个值的某两个或三个相同，则对应的移位寄存器在下一时刻被驱动，而剩下的一个(或0 个)值对应的移位寄存器则停走。

钟控单元			驱动情况		
$LFSR_1[8]$	$LFSR_2[10]$	$LFSR_3[10]$	$LFSR_1$	$LFSR_2$	$LFSR_3$
$1 \oplus c$	c	c	停走	驱动	驱动
c	$1 \oplus c$	c	驱动	停走	驱动
c	c	$1 \oplus c$	驱动	驱动	停走
c	c	c	驱动	驱动	驱动

- 初始化
- − 令所有 LFSR 的各级寄存器均为0；
- − 对 $i=0,\cdots,63$ 做：
- ♦ $LFSR1[0] = LFSR1[0] \oplus Key[i]$；

	$LFSR_1$	$LFSR_2$	$LFSR_3$
级数	19	22	23
抽头	13、16、17、18	20、21	7、20、21、22
联接多项式	$x^{19}+x^{18}+x^{17}+x^{14}+1$	$x^{22}+x^{21}+1$	$x^{23}+x^{22}+x^{21}+x^{8}+1$
钟控抽头	8	10	10

序列密码设计准则 —小结

序列密码的输出序列，必须满足

A1：输出序列确保一个最小的周期长度

A2：密文显得具有随机性

◆ 可分解为下述基本原则：

① 长周期。

② 高线性复杂度。

③ 统计性能好。

④ 足够的“混乱”。

⑤ 足够的“扩散”。

⑥ 抵抗不同形式的攻击。

作业

- 已知序列密码的密文串1010110110和相应的明文串0100010001，而且还已知密钥流是使用3级线性反馈移位寄存器产生的，试破译该密码系统。
- 下次上课交

- 下次课程：公开密码算法之RSA