上讲主要内容

- 分组密码的介绍
- **DES**密码算法
- **AES**密码算法
- 分组密码的操作方式

序列密码

主讲人：马春光

machunguang@hrbeu.edu.cn

本讲主要内容

- 序列密码的介绍
- 线性反馈移位寄存器
- 非线性序列
- 序列密码的算法举例(A5、RC4)

序列密码的简介

!伪随机序列是具有某种随机特性的确定的序列。它们是由移位寄存器产生确定序列，然而他们却具有某种随机特性的随机序列。

序列密码通常认为起源于20世纪20年代的维尔姆(Vernam)密码，Vernam密码中的密钥序列要求是随机的序列，由于随机的密钥序列的产生、存储以及分配等方面存在一定的困难，Vernam体制在当时并没有得到广泛的应用。随着微电子技术和数学理论的发展和完善，基于伪随机序列的序列密码得到了长足的发展和应用，其产生有比较成熟的数学理论工具。如果密钥序列是随机的序列，则序列密码就是“一次一密”密码体制。商农已经证明“一次一密”密码体制在理论上是不可破译的。目前，序列密码是世界各国的军事和外交等重要领域中使用的主要密码体制之一。

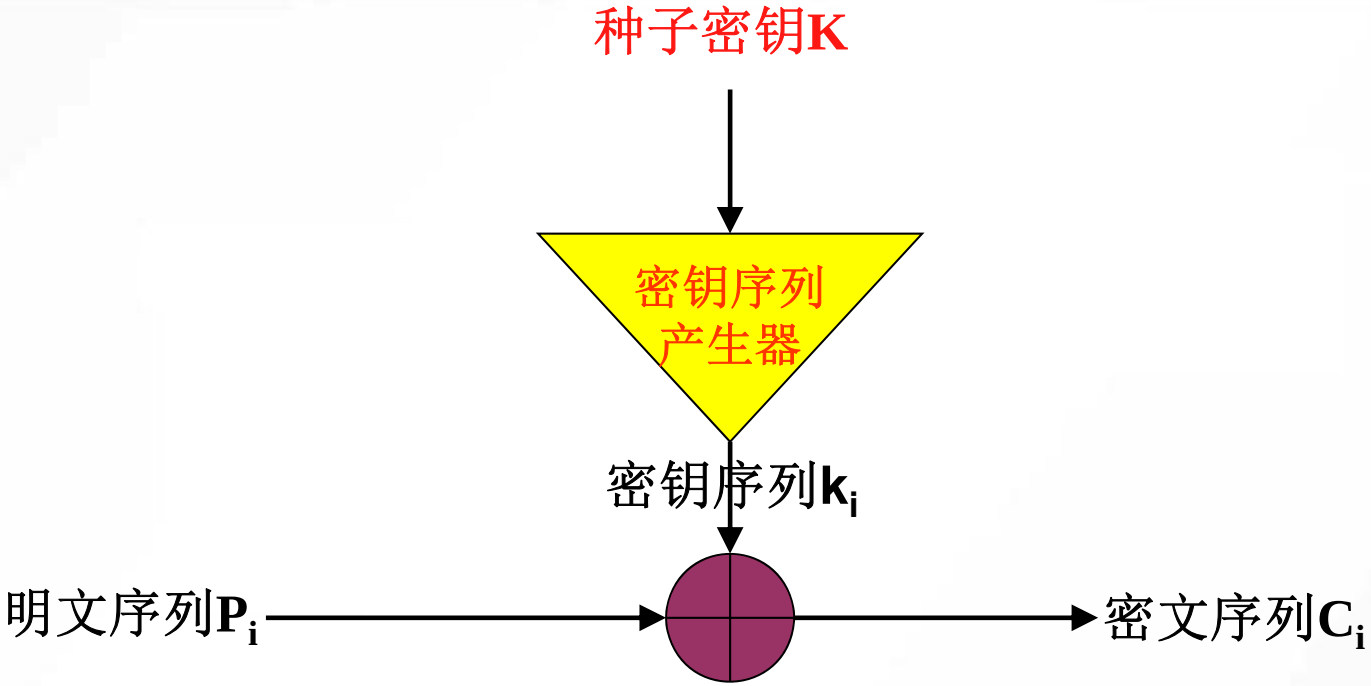
在序列密码中，加密和解密所用的密钥序列都是伪随机序列，序列密码是对称密码体制中的一类，又称为流密码。

序列密码的描述

在序列密码中，将明文消息按一定长度（长度较小）分组，然后对各组用相关但不同的密钥进行加密，产生相应的密文，相同的明文分组会因在明文序列中的位置不同而对应于不同的密文分组。解密用相同的密钥序列对密文序列进行分组解密以恢复出明文序列。

- 设明文为$p=p_0p_1p_2\cdots$　　$p_i\in GF(2),\ i\geq 0$
- 设密钥为$k=k_0k_1k_2\cdots$　　$k_i\in GF(2),\ i\geq 0$
- 设密文为$c=c_0c_1c_2\cdots$　　$y_i\in GF(2),\ i\geq 0$
- 则加密变换为$c_i=E_{ki}(p_i)$　　$i\geq 0$
- 则解密变换为$p_i=D_{ki}(c_i)$　　$i\geq 0$

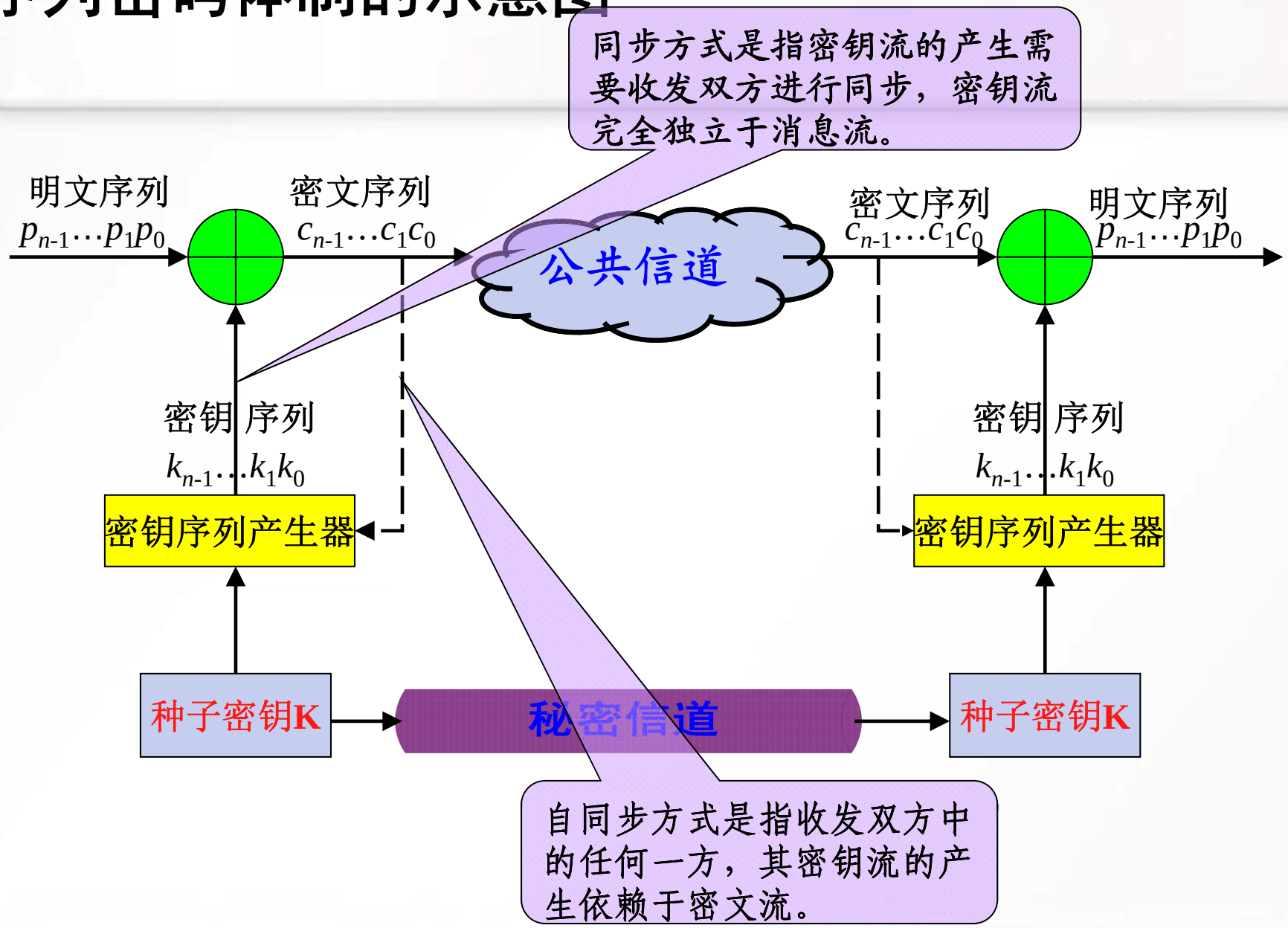
序列密码的原理



特点：

- 加解密运算只是简单的模二加运算。
- 密码安全强度主要依赖密钥流的安全性。

序列密码体制的示意图


同步方式是指密钥流的产生需要收发双方进行同步，密钥流完全独立于消息流。
明文序列
$p_{n-1}\cdots p_1p_0$
密文序列
$c_{n-1}\cdots c_1c_0$
公共信道
密文序列
$c_{n-1}\cdots c_1c_0$
明文序列
$p_{n-1}\cdots p_1p_0$
密钥 序列
$k_{n-1}\ldots k_1k_0$
密钥序列产生器
密钥 序列
$k_{n-1}\ldots k_1k_0$
密钥序列产生器
种子密钥K
秘密信道
种子密钥K
自同步方式是指收发双方中的任何一方，其密钥流的产生依赖于密文流。

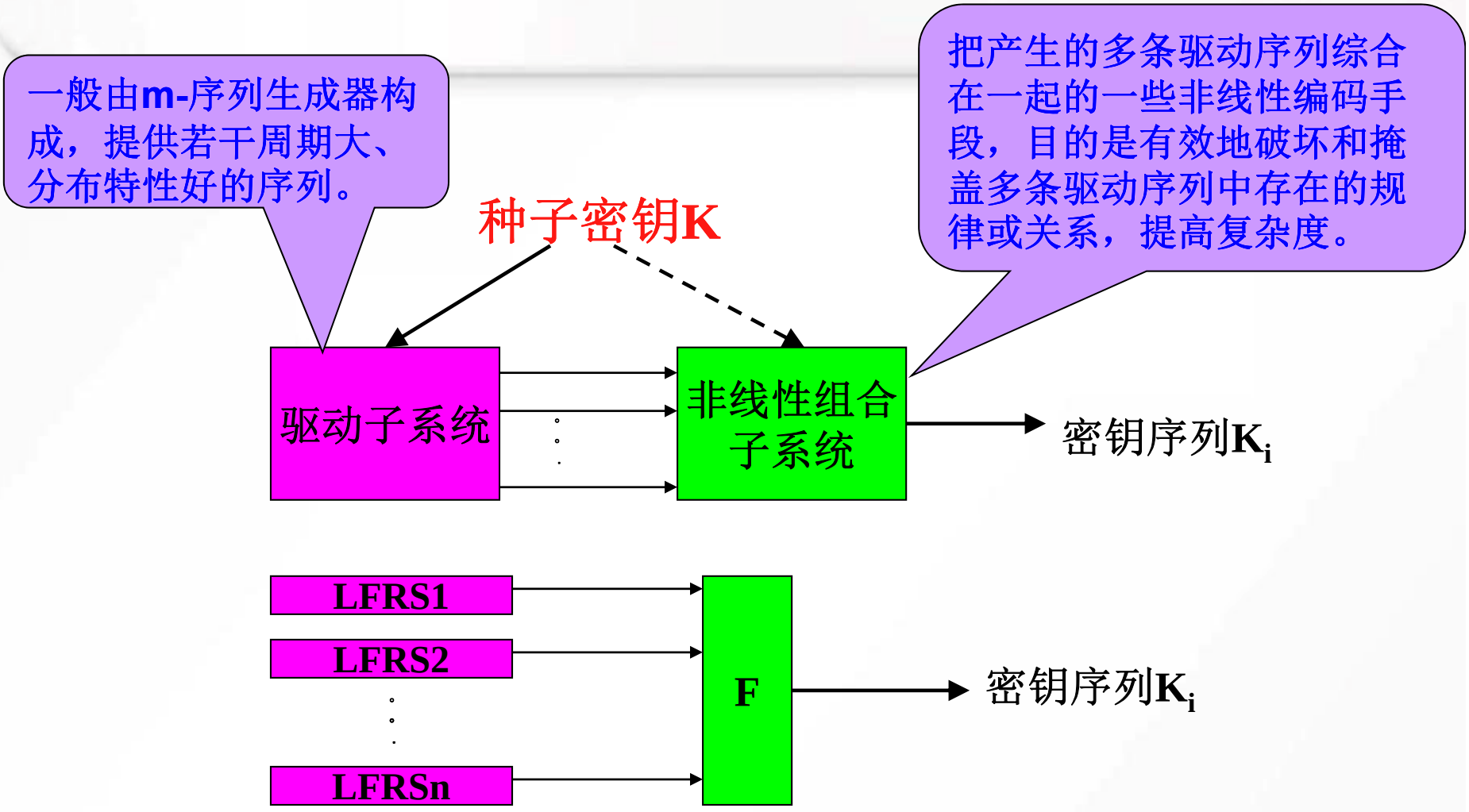
密钥序列产生器(KG)基本要求

- 种子密钥K的长度足够大，一般应在128位以上;
- KG生成的密钥序列 $\{k_i\}$ 具极大周期;
- $\{k_i\}$ 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);
- 利用统计方法由 $\{k_i\}$ 提取关于KG结构或K的信息在计算上不可行;

K:种子密钥

- 雪崩效应。即K任一位的改变要引起 $\{k_i\}$ 在全貌上的变化。
- 密钥流 $\{k_i\}$ 不可预测的。密文及相应的明文的部分信息，不能确定整个 $\{k_i\}$ 。

密钥序列产生器的分解



常用的密钥序列产生器

LFSR: Linear Feedback Shift Register

线性反馈移位奇存器理论的简介

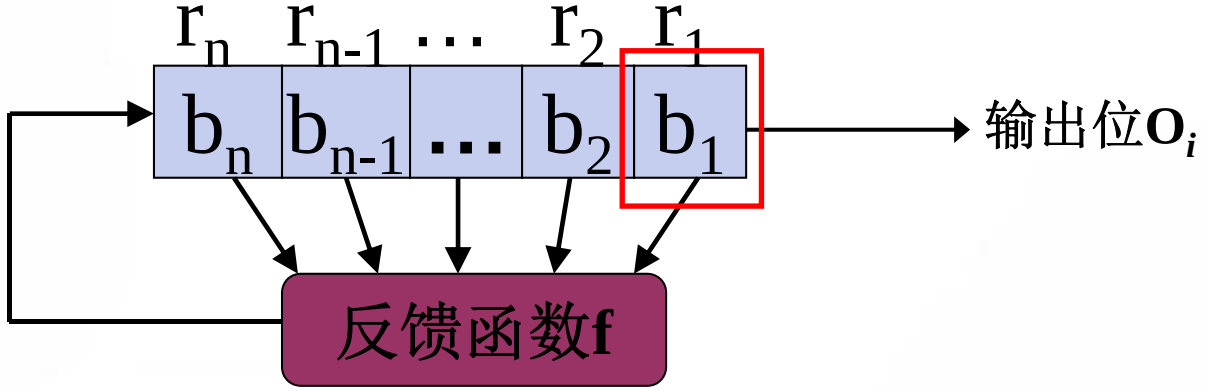
序列密码的关键是设计一个随机性好的密钥流发生器，为了研究密钥流发生器，挪威政府的首席密码学家**Ernst Selmer** 于**1965**年提出了移位寄存器理论，它是序列密码中研究随机密钥流的主要数学工具。

尤其，线性反馈移位寄存器因其实现简单、速度快、有较为成熟的理论等优点而成为构造密码流生成器的最重要的部件之一。

三个概念：
(1)移位寄存器（shift Register），
(2)反馈移位寄存器(Feedback shift Register)，
(3)线性反馈移位寄存器(Linear Feedback shift Register)

反馈移位寄存器

反馈移位寄存器(feedback shift register,FSR)是由n位的寄存器和反馈函数(feedback function)组成，如下图所示，n位的寄存器中的初始值称为移位寄存器的初态。

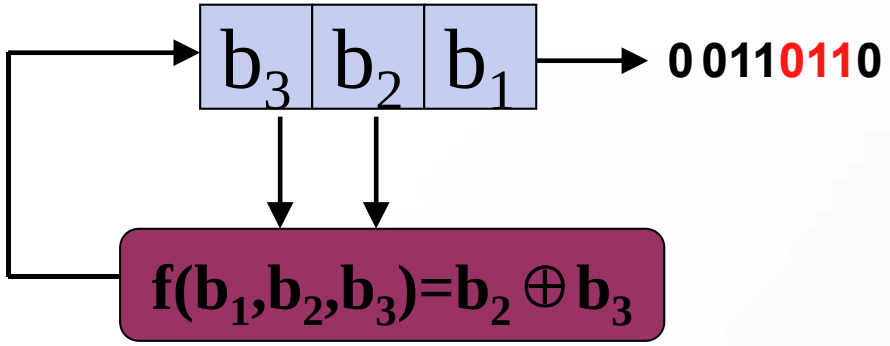


工作原理：移位寄存器中所有位的值右移1位，最右边的一个寄存器移出的值是输出位，最左边一个寄存器的值由反馈函数的输出值填充，此过程称为进动1拍。反馈函数f是n个变元$(b_1,b_2,\ldots,b_n)$的布尔函数。移位寄存器根据需要不断地进动m拍，便有m位的输出，形成输出序列$o_1,o_2,\ldots,o_m$。

反馈移位寄存器（举例）

一个3-级的反馈移位寄存器，反馈函数$f(x)=b_2 \oplus b_3$，初态为100，输出序列生成过程如下：

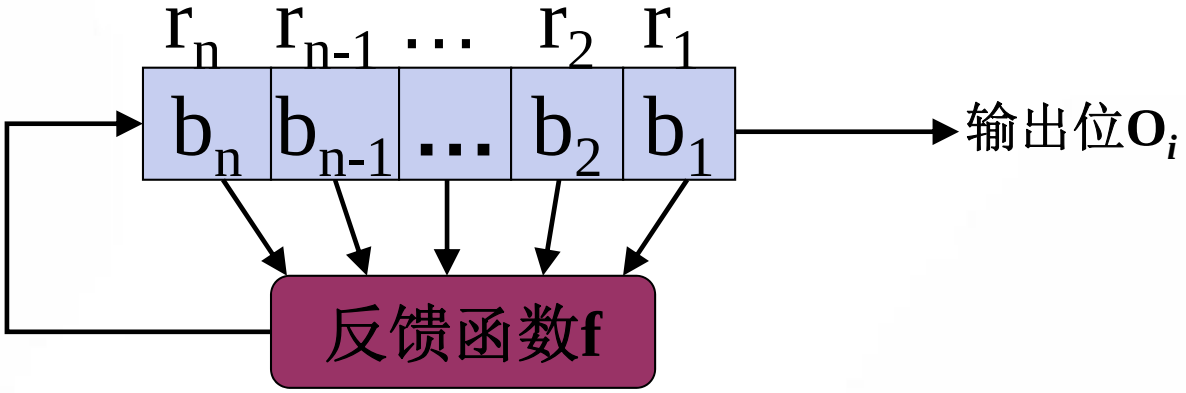
状态	输出位
100	0
110	0
011	1
101	1
110	0
011	1
101	1
110	0



移位寄存器的周期是指输出序列中连续且重复出现部分的长度。上面输出序列周期长度为3。

线性反馈移位寄存器

线性反馈移位寄存器(liner feedback shift register,LFSR)是一种特殊的FSR，其反馈函数是移位寄存器中某些位的异或，参与运算的这些位叫做抽头位。其一般组成结构如下图所示。



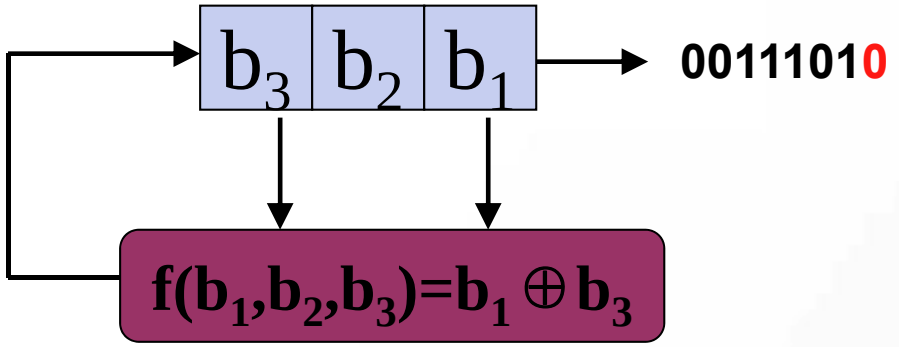
特征：n-级 LFSR的有效状态为2^n-1（全0状态除外，因全0状态的输出序列一直为全0），也即理论上能够产生周期为2^n-1的伪随机序列。但要产生最大周期的输出序列，抽头位要有要求。

线性反馈移位寄存器（举例）

一个3-级的反馈移位寄存器，反馈函数$f(x)=b_1 \oplus b_3$，初态为100，输出序列生成过程如下：

状态	输出位
100	0
110	0
111	1
011	1
101	1
010	0
001	1
100	0

回到了寄存器的初始状态



- ✓ 上面输出序列周期长度为7。已达到最大周期2^3-1=7
- ✓ 抽头位对输出周期长度的影响起着决定性的作用。
- ✓ 初态对输出周期长度没影响。

m序列的简介

线性反馈移位寄存器输出序列的性质完全由其反馈函数决定。n级线性反馈移位寄存器最多有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级线性反馈移位寄存器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。

m-序列特性

- **0,1平衡性：**在一个周期内，0、1出现的次数分别为$2^{n-1}-1$和2^{n-1}。
- **游程特性：**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。例如00110111,y依次称为0的2游程，1的2游程，0的1游程，和1的3游程, 总游程数为4
- **$\{x_i\}$的自相关函数为**

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

$R(j)=\dfrac{A-D}{P}$, $j=1,2,\cdots;p$为序列$\{x_i\}$的周期。

其中：$A=\left|\{0\le i<p, x_i=x_{i+j}\}\right|, D=\left|\{0\le i<p, x_i\ne x_{i+j}\}\right|,$

概念：生成多项式(generation polynomial)：由抽头序列加上常数1形成的多项式。

LFSR 的多项式表示

定义 特征多项式：设n级线性移位寄存器的输出序列$\{a_i\}=\{an,\cdots,a2,a1\}$满足递推关系：

$$a_{n+k}=c_1a_{n+k-1}\oplus c_2a_{n+k-2}\oplus\cdots\oplus c_{n-1}a_{k+1}\oplus c_na_k(k\geq 1)$$

例如：$a_{n+1}=c_1a_n\oplus c_2a_{n-1}\oplus\cdots\oplus c_{n-1}a_2\oplus c_na_1(k\geq 1)$

LFSR 的多项式表示

这种特征关系可以用一元高次多项式表示，该多项式称为LFSR特征多项式。反过来也可以用该特征多项式构成反馈函数*f*.

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

n表示LFSR的级数，$c_i \in \{0,1\}$

LFSR 的多项式表示

例如：3级LFSR的反馈函数如下：

$$f(a_3, a_2, a_1) = a_3 \oplus a_1$$

$$其中 c1 = 1, c2 = 0, c3 = 1$$

则其特征多项式为：

$$P(x) = 1 + x + x^3$$

LFSR 的多项式表示

反过来，一个特征多项式为：

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

则**n=4,c1=c2=c3=c4=1,**其对应的**LFSR**的反馈函数如下：

$$f(a_4,a_3,a_2,a_1)=c_1a_4\oplus c_2a_3\oplus c_3a_2\oplus c_4a_1$$

$$=a_4\oplus a_3\oplus a_2\oplus a_1,\text{也即递推关系：}$$

$$a_k=a_{k-1}\oplus a_{k-2}\oplus a_{k-3}\oplus a_{k-4}(k\geq 5)$$

LFSR所产生的序列是m序列，必要条件是其所对应的生成多项式是本原多项式！！！

相关定义与定理

定义 不可约多项式：仅能被常数或自身的常数倍除尽，但不能被其它多项式除尽的多项式称为不可约多项式。　　注意：x^2+1为非不可约多项式，因为它有因式$x+1$

定义 阶：设p(x)是GF(2)上的多项式，使$p(x) \mid (x^p-1)$ (P(x)整除x^p-1)的最小p称为p(x)的**周期或阶**。

定理：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

注：这里讲的多项式都是在GF(2)上的多项式！

Question：x^3+1是不可约多项式吗？

相关定义与定理

定义 本原多项式：若n次不可约多项式p(x)的阶是2^n-1,则称p(x)为**本原多项式**。

定理：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

定义 特征多项式：设n级线性移位寄存器的输出序列$\{a_i\}=\{an,\cdots,a2,a1\}$满足递推关系：

$$a_{n+k}=c_1a_{n+k-1}\oplus c_2a_{n+k-2}\oplus\cdots\oplus c_{n-1}a_{k+1}\oplus c_na_k(k\geq 1)$$

例如：$a_{n+1}=c_1a_n\oplus c_2a_{n-1}\oplus\cdots\oplus c_{n-1}a_2\oplus c_na_1(k\geq 1)$

m序列生成

为了使LFSR成为m-序列，由抽头序列加上常数1形成的特征多项式必须是本原多项式。

$\{a_i\}$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

例：n=4, ϕ (2⁴-1)= ϕ (15）=15*（1-1/3）（1-1/5）=8
与15互素的数为1,2,4,7,8,11,13,14
即4级LFSR有8/4=2个本原多项式。

现已证明，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。

注：$(x^{15}-1)\div P(x)=x^{11}+x^{8}+x^{7}+x^{5}+x^{3}+x^{2}+x+1$

m序列生成举例

例：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，另外p(x)都不能被$x,x+1,x^2+x+1$整除，所以p(x)是不可约多项式，其阶又为2^4-1，所以p(x)为本原多项式.

若LFSR以p(x)为特征多项式，则输出序列的递推关系为

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

$$f(a4,a3,a2,a1) = a_4 \oplus a_1$$

m序列生成举例

若初始状态{a4 a3 a2 a1}=(1001), 则输出为：

1001 0001 1110 101 1001 0001 1110 101 ……。周期为$2^4-1=15$，为m序列。

a4a3a2a1

100 1

010 0

001 0

000 1

100 0

110 0

111 0

111 1

011 1

101 1

010 1

101 0

110 1

011 0

001 1

1001→出现了重复状态

m序列密码的破译

设敌手知道一段长为2n的明密文对，即已知

$$x = x_1 x_2 \cdots x_{2n} \qquad y = y_1 y_2 \cdots y_{2n}$$

于是可求出一段长为2n的密钥序列

$$z = z_1 z_2 \cdots z_{2n}$$

其中

$$y_i = x_i \oplus z_i, \text{可得} \boxed{z_i = x_i \oplus y_i}$$

m序列密码的破译(续)

由此可推出线性反馈移位寄存器连续的**n+1**个状态：

$$S_1 = (z_1 z_2 \cdots z_n) \overset{\text{记为}}{=} (a_1 a_2 \cdots a_n)$$

$$S_2 = (z_2 z_3 \cdots z_{n+1}) \overset{\text{记为}}{=} (a_2 a_3 \cdots a_{n+1})$$

$\cdots$

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

序列$\{a_i\}$满足线性递推关系：

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

m序列密码的破译(续)

生成矩阵 $X=\begin{pmatrix} S_1 & S_2 & \cdots & S_n \end{pmatrix}$

而

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

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

前面的n个状态构成了一个矩阵第n+1个状态($a_{n+1},\ldots,a_{2n}$)用于求解系数。

若**X**可逆，则

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

m序列密码的破译举例

例： 设敌手得到密文串**1011010111110010**和相应的明文串**0110011111111001**，因此可计算出相应的密钥流为**1101001000001011**。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

第6个状态(a_6,a_7,a_8,a_9,a_{10})

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

m序列密码的破译举例(续一)

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

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

m序列密码的破译举例(续二)

从而得到

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

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

密钥流的递推关系为

a1a2a3a4a5
1 1 0 1 0　　010000101

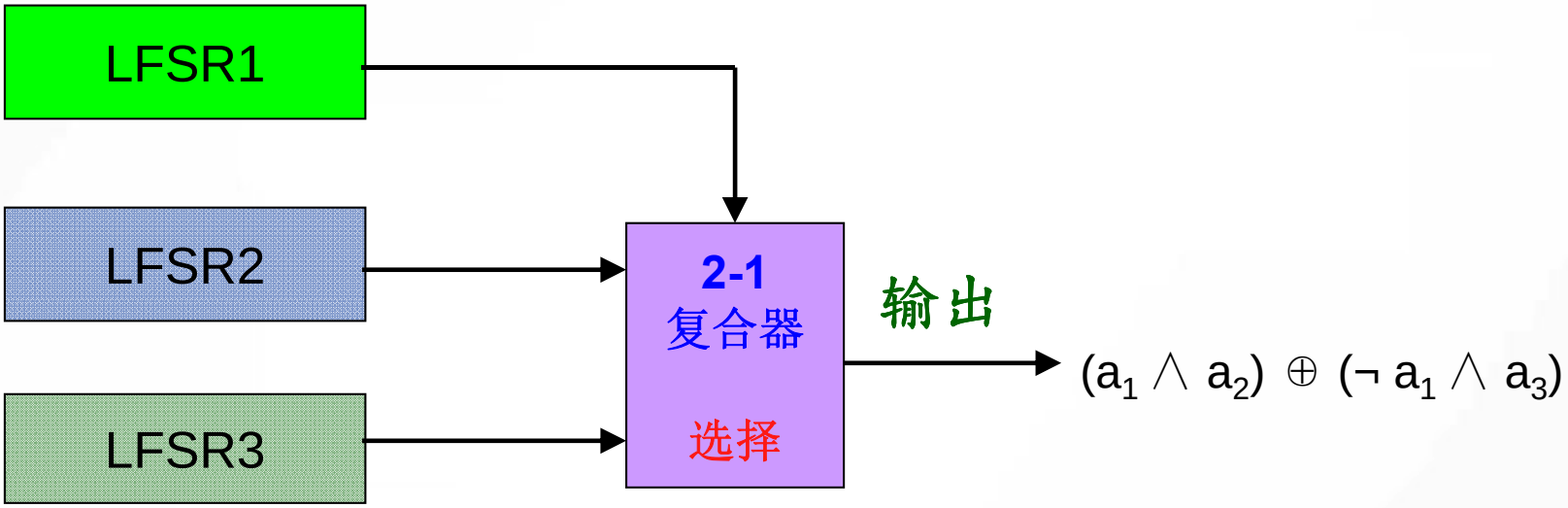
$$a_{i+5} = c_5 a_i \oplus c_2 a_{i+3} = a_i \oplus a_{i+3}$$

非线性序列

为了使密钥流生成器输出的二元序列尽可能复杂，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，因此常使用多个LFSR来构造二元序列，称每个LFSR的输出序列为驱动序列，显然密钥流生成器输出序列的周期不大于各驱动序列周期的乘积，因此，提高输出序列的线性复杂度应从极大化其周期开始。

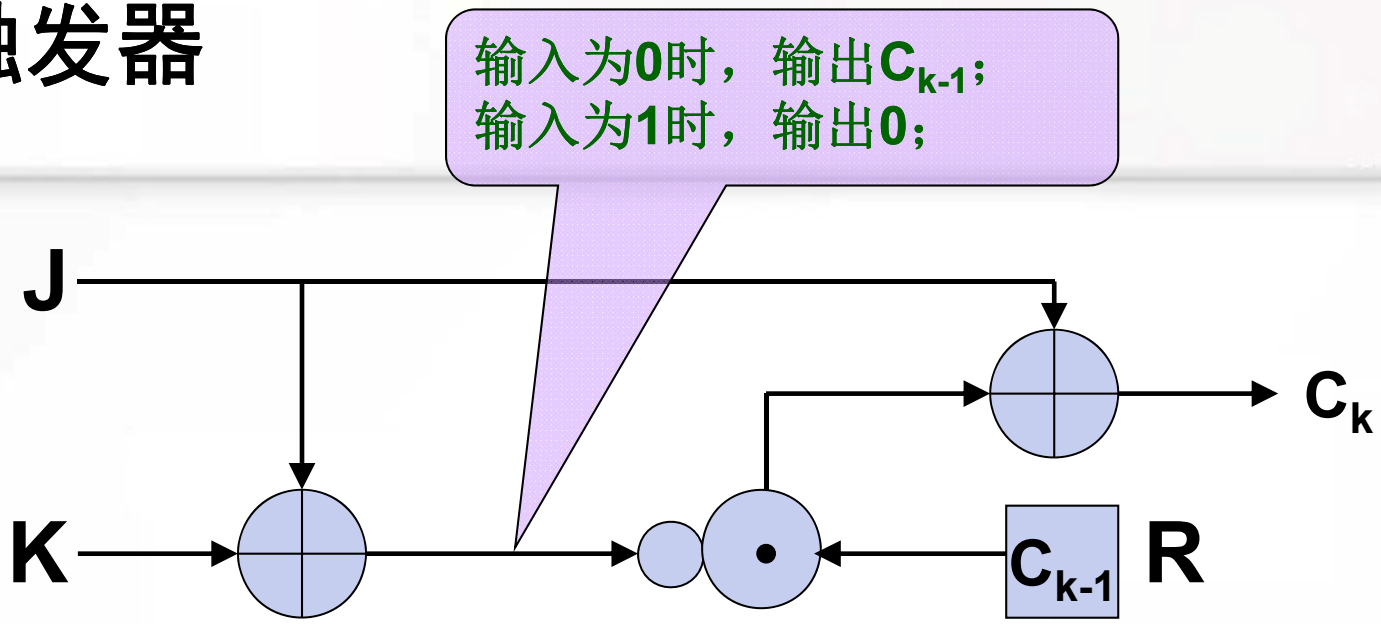
- **Geffe序列生成器**
- **J-K触发器**
- **Pless生成器**
- 门限发生器

Geffe序列生成器



这个发生器的周期是三个**LFSR**的周期的最小公倍数。假设三个本原反馈多项式的阶数互素，那么这个发生器的周期是三个**LFSR**的周期之积.

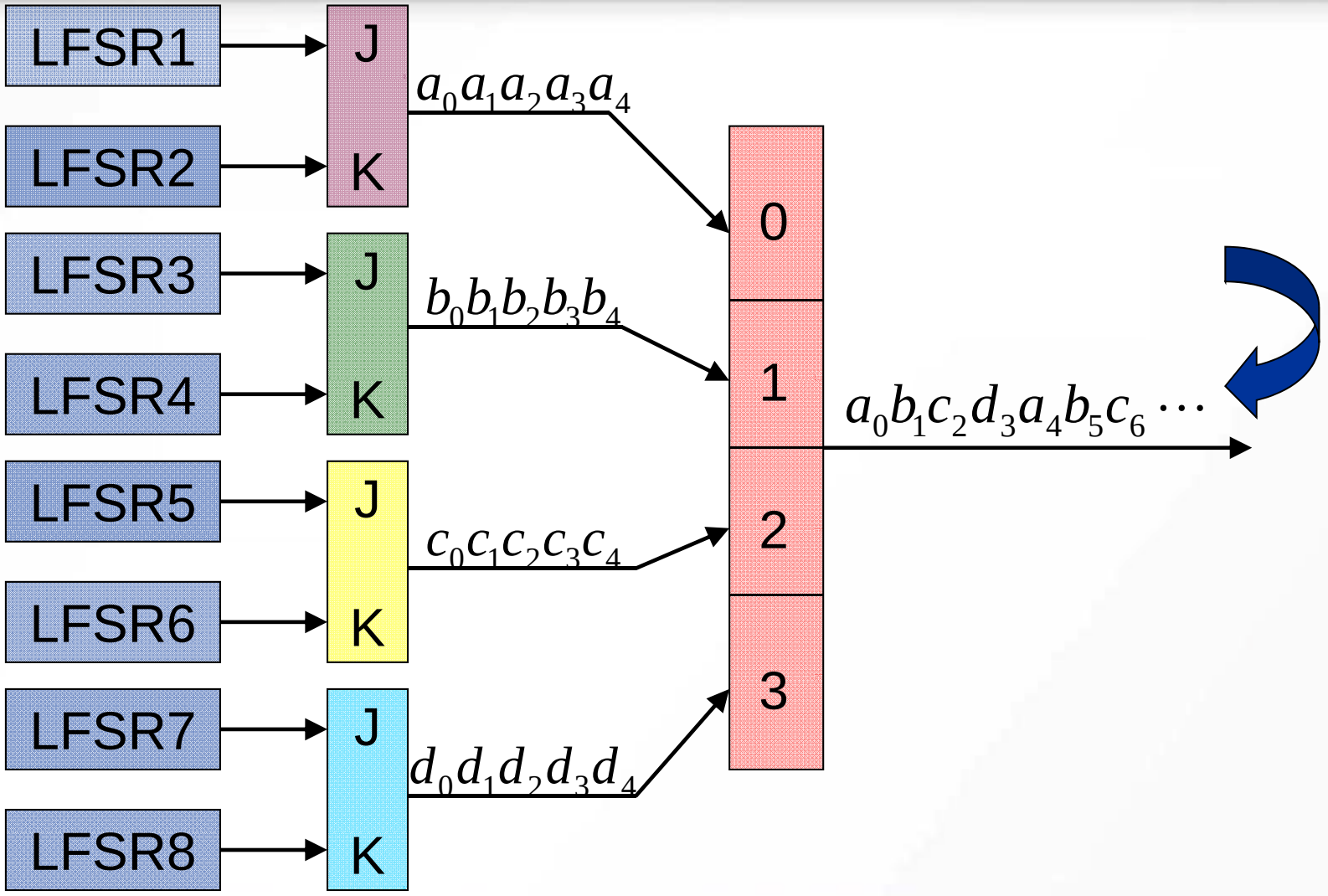
J-K触发器



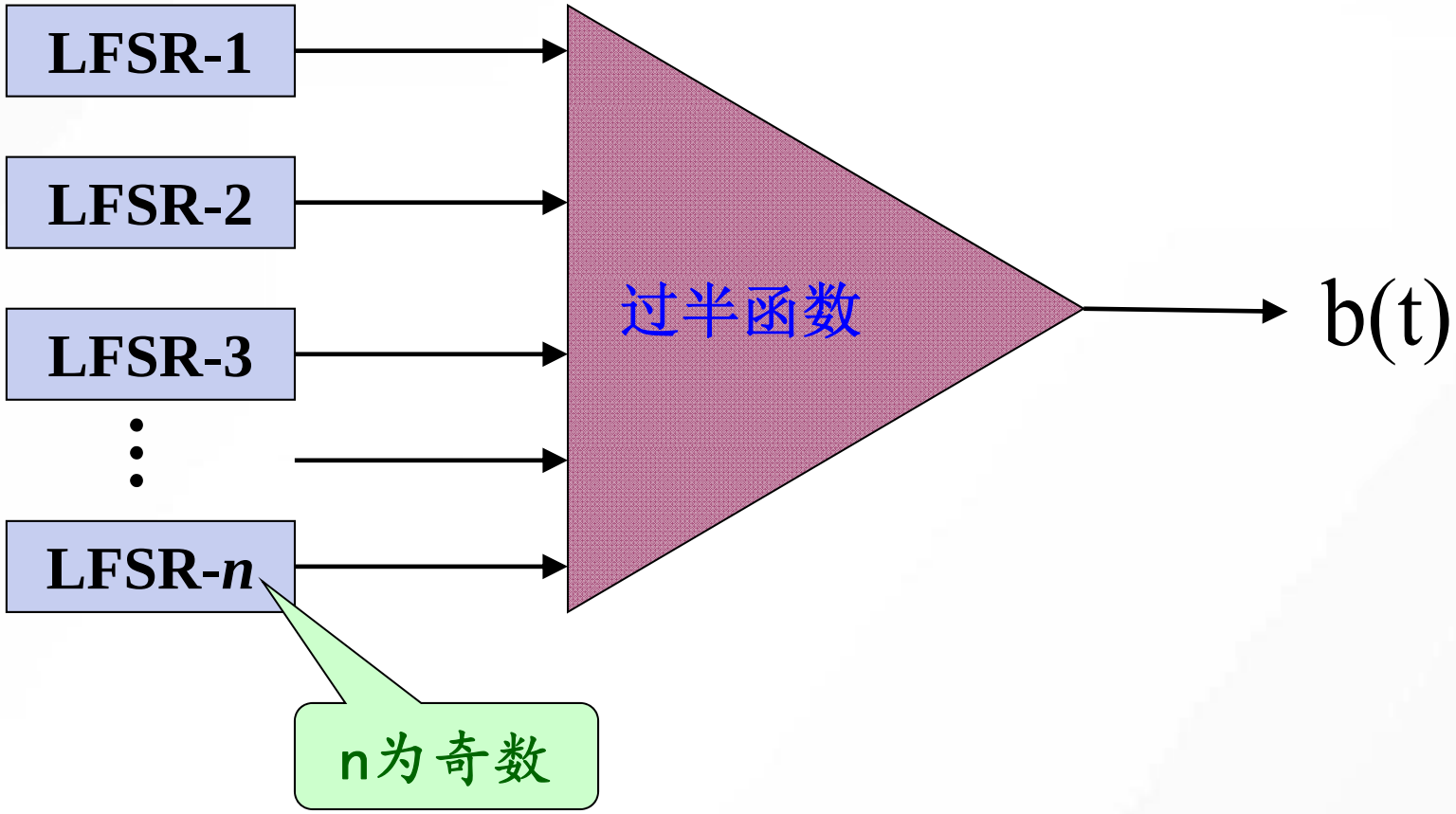
J-K触发器真值表

J	K	C_k
0	0	C_{k-1}
0	1	0
1	0	1
1	1	非C_{k-1}

Pless生成器


LFSR1
LFSR2
LFSR3
LFSR4
LFSR5
LFSR6
LFSR7
LFSR8
J
K
J
K
J
K
J
K
$a_0a_1a_2a_3a_4$
$b_0b_1b_2b_3b_4$
$c_0c_1c_2c_3c_4$
$d_0d_1d_2d_3d_4$
0
1
2
3
$a_0b_1c_2d_3a_4b_5c_6\cdots$

门限发生器



A5算法

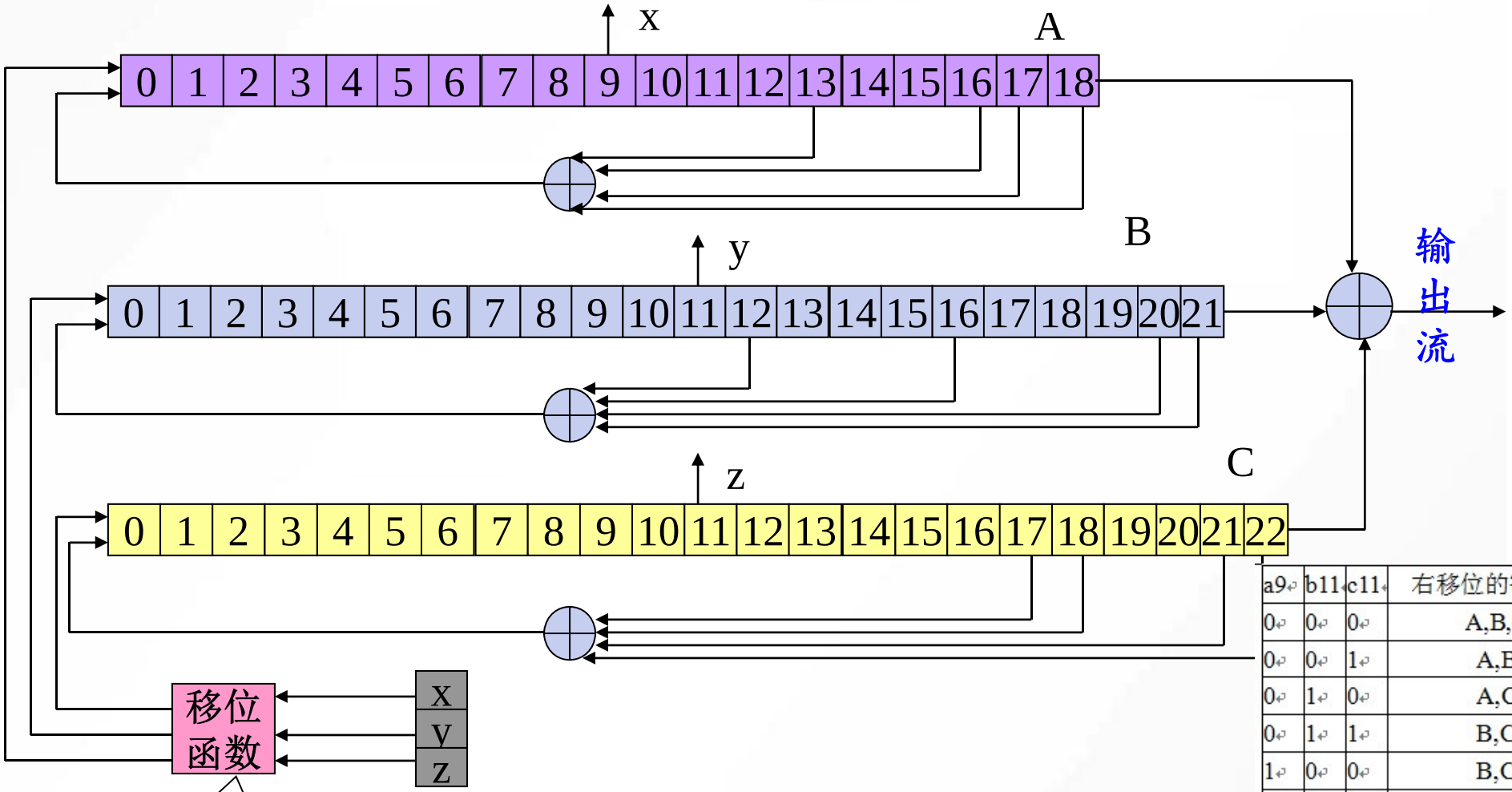
- A5算法已经被应用于GSM通信系统中，用于加密从手机到基站的连接，用来保护语音通信。一个GSM语言消息被转换成一系列的帧，每帧具有228位。每帧用A5进行加密。
- A5算法的主要组成部分是三个长度不同的线性移位寄存器，即A，B，C。其中A有19位， B有22位， C有23位。
- 移位是由时钟控制的，且遵循“择多”的原则。即从每个寄存器中取出一个中间位并进行判断，三个数中占多数的寄存器参加移位，其余的不移位。比如在取出的三个中间位中有两个为“1”，则为“1”的寄存器进行一次移位，而为“0”的不移。反过来，若三个中间位中有两个为“0”，则为“0”的寄存器进行一次移位，而为“1”的不移。

A5算法的示意图

$$f(A) = a18 \oplus a17 \oplus a16 \oplus a13$$

$$f(B) = b21 \oplus b20 \oplus b16 \oplus b12$$

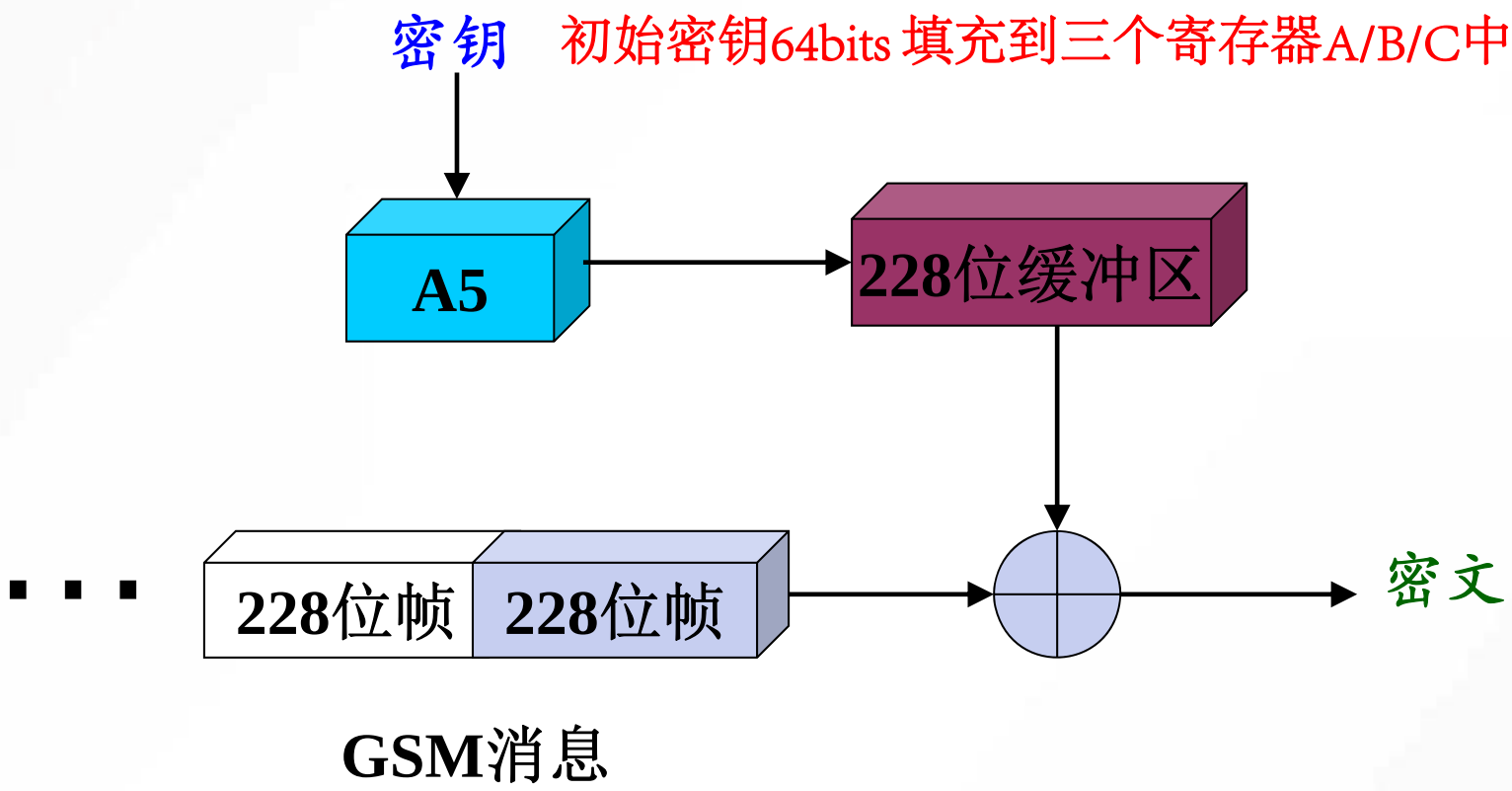
$$f(C) = c22 \oplus b21 \oplus c18 \oplus c17$$



a9	b11	c11	右移位的寄存器
0	0	0	A,B,C
0	0	1	A,B
0	1	0	A,C
0	1	1	B,C
1	0	0	B,C
1	0	1	A,C
1	1	0	A,B
1	1	1	A,B,C

遵循“多数原则”

GSM中使用A5流加密法



RC4的简介

- RC4（Ron Rivest Cipher）是RSA数据安全公司开发的可变密钥长度的序列密码，是世界上使用最广泛的序列密码之一，其优点是很容易用软件实现，加解密速度快（大约比DES快10倍）。
- 与基于移位寄存器的序列密码不同，RC4密码是一种基于非线性数据表变换的序列密码。它以一个足够大的数据表为基础，对表进行非线性变换，产生非线性的密钥流序列。
- 一个可变密钥长度、面向字节操作的序列密码，该算法以随机置换作为基础。分析显示该密码的周期大于10^{100}。

RC4算法简介

- RC4的大小根据参数n的值而变化，通常n=8，这样RC4可以生成总共有256（2^8）个元素的数据表S：S_0，S_1，…，S_{255}。
- 种子密钥长度为从1~256个字节（即8~2048位）的可变长度密钥，用于初始化一个256个字节的状态矢量S。
- RC4有两个主要的算法：

密钥调度算法KSA(Key-Scheduling Algorithm)

伪随机数生成算法PRGA (Pseudo Random-Generation Algorithm)

RC算法的基本思想

- 依据种子密钥，利用密钥调度算法对数据表S进行重新排列。

- 利用伪随机数生成算法，从已重新排列的数据表S中取出一个字节。每取出一个字节，数据表S将发生变化。

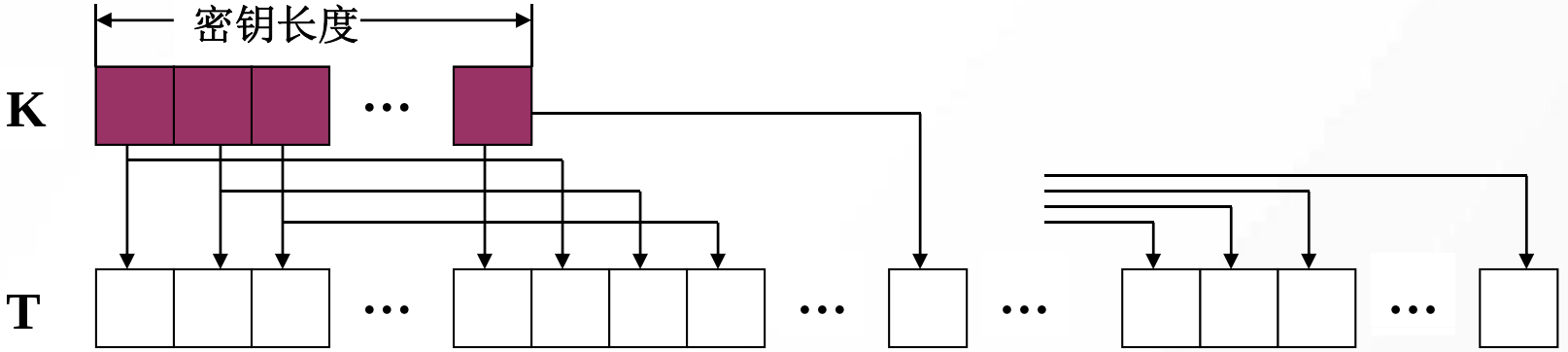
RC算法

- 数据表S的初始状态
- 数据表S的初始置换
- 密钥流的生成

数据表S的初始状态

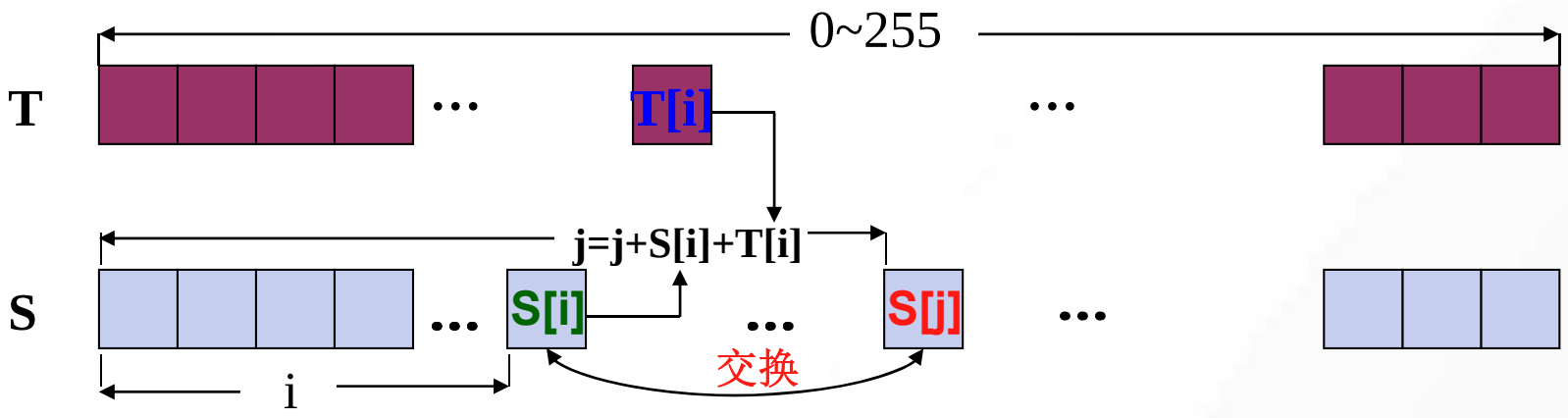
初始化S，即S(i):=i(一个字节),0≤i≤255。

用主(种子)密钥K按字节填充另一个T表，
即T(i):=K(i mod keylen)，0≤i≤255。



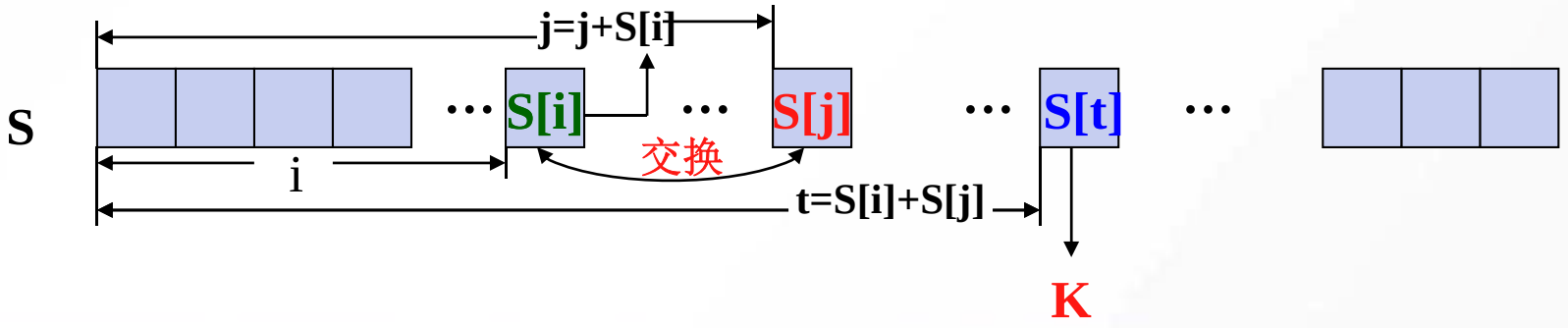
数据表S的初始置换

```
j:=0;
for i := 0 to 255 do
    begin
        j := (j + S[i] + T[i]) (mod 256);
        swap(S[i], S[j]);    // 交换S(i)和S(j)的内容;
    end
```



密钥流的生成

```
i,j:=0
while (true)
  begin
     i:= i + 1 (mod 256);
     j := j + S[i] (mod 256);
     swap(S[i], S[j]);
     t := S[i] + S[j] (mod 256);
     k := S[t];
  end
```



RC4算法说明

- 加密时，将k的值与明文字节异或；解密时，将k的值与密文字节异或。
- 为了保证安全强度，目前的RC4至少使用128位密钥。
- RC4算法可看成一个有限状态自动机，把S表和i,j索引的具体取值称为RC4的一个状态：$T=(S_0,S_1,\ldots,S_{255},i,j)$。对状态T进行非线性变化，产生出新的状态，并输出密钥序列中的一个字节k。大约有2^{1700}（$256!*256^2$）中可能状态，一个巨大的数字。
- 用大的数据表S和字长来实现这个思想是可能，即定义16-位RC4。

举例说明

假如使用3位（从0到7）的RC4，其操作是对8取模（而不是对256取模）。数据表S只有8个元素，初始化为：

S	0	1	2	3	4	5	6	7
	0	1	2	3	4	5	6	7

选取一个密钥，该密钥是由0到7的数以任意顺序组成的。例如选取5、6和7作为密钥。该密钥如下填入密钥数据表中：

K	5	6	7	5	6	7	5	6
	0	1	2	3	4	5	6	7

密钥调度算法KSA(举例)

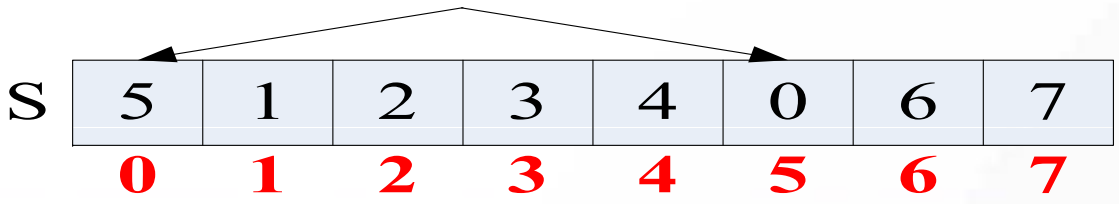
然后利用如下循环构建实际的S数据表：

```
j:=0;
for i=0 to 7 do
    j:=(j+s(i)+k(i)) mod 8;
    swap(S(i),S(j));
```

该循环以j=0和i=0开始。使用更新公式后j为：

$$j=(0+S(0)+K(0)) \bmod 8=5$$

因此，S数据表的第一个操作是将S(0)与S(5)互换。

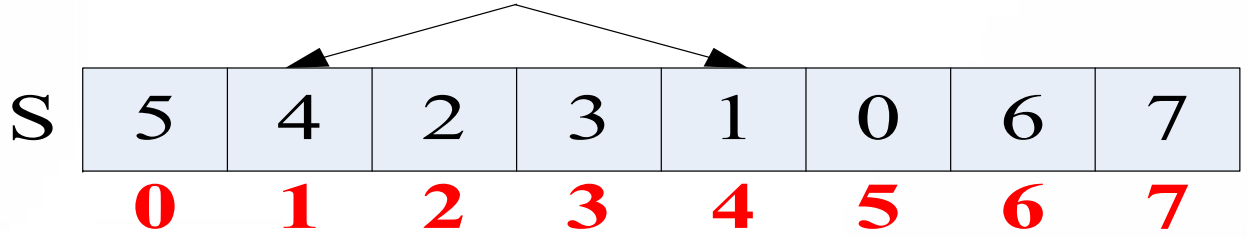


密钥调度算法KSA(举例)续

索引i加1后，j的下一个值为：

j=(5+S(1)+K(1)) mod 8=(5+1=6) mod 8=4

即将S数据表的S(1)和S(4)互换：



当该循环执行完后，数据表S就被随机化：

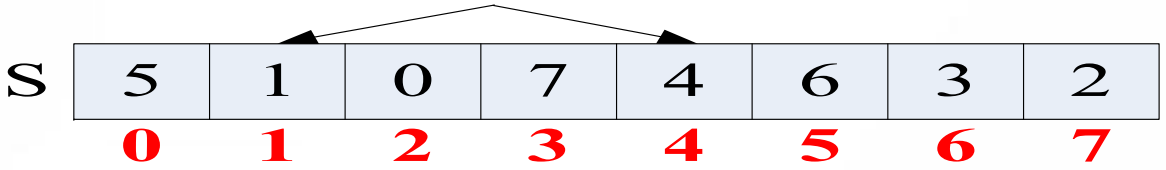
伪随机数生成算法PRGA（举例）

这样数据表S就可以用来生成随机的密钥流序列。从j=0和i=0开始，RC4如下计算第一个密钥字：

i=(i+1) mod 8=(0+1) mod 8 =1

j=(j+s(i)) mod 8=(0+s(1)) mod 8=(0+4) mod 8=4

swap S(1)和S(4)



然后如下计算t和k:

t=(S(j)+S(i)) mod 8=(S(4)+S(1)) mod 8=(1+4) mod 8=5

k=S(t)=S(5)=6

第一个密钥字为6，其二进制表示为110。反复进行该过程，直到生成的二进制的数量等于明文位的数量。

序列密码的优点（相对分组密码）

- 在硬件实施上，序列密码的速度一般要比分组密码快，而且不需要有很复杂的硬件电路。
- 在某些情况下（例如某些电信上的应用），当缓冲不足或必须对收到字符进行逐一处理时，序列密码就显得更加必要和恰当。
- 序列密码有较理想的数学分析。
- 序列密码能较好地隐藏明文的统计特性。

序列密码小结

- 安全强度取决于密钥序列的随机性和不可预测性
- 线性反馈移位寄存器理论上能够产生周期为$2^{|n|}-1$的伪随机序列
- 不存在数据扩展和错误转播
- 实时性好，运算速度快，加密、解密易实现
- 密钥分配困难

本讲主要内容

- 序列密码的介绍
- 线性反馈移位寄存器
- 非线性序列
- 序列密码的算法举例（**A5**、**RC4**）

谢 谢 ！