北京郵電大學
BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

《现代密码学》第五讲

PRG和流密码（一）

上讲内容回顾

北京郵電大學
BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

- 分组密码定义
- 分组密码的发展历史
- 分组密码算法设计思想
- 数据加密标准（DES）算法介绍
- 高级加密标准（AES）算法介绍
- 国密SM4算法简介
- 分组密码算法的应用

本讲主要内容

- OTP与伪随机数生成器
- 流密码技术的发展
- 基于LFSR的PRG
- RC4算法
- Estream 算法举例
- PRG安全与流密码安全应用

本章主要内容

- OTP与伪随机数生成器
- 流密码技术的发展
- 基于LFSR的PRG
- RC4算法
- Estream 算法举例
- PRG安全与流密码安全应用

OTP与伪随机数发生器

20世纪20年代的Vernam体制， 即“一次一密(one time pad)”密码体制，具有完善保密性。

- 明文：$m=m_1m_2\cdots m_i \in GF(2)$， $i>0$
- 密钥：$k=k_1k_2\cdots k_i \in GF(2)$， $i>0$
- 密文：$c=c_1c_2\cdots c_i \in GF(2)$， $i>0$
- 加密变换：$c_i=m_i+k_i \pmod 2$， $i>0$
- 解密变换：$m_i=c_i+k_i \pmod 2$， $i>0$

OTP与伪随机数生成器

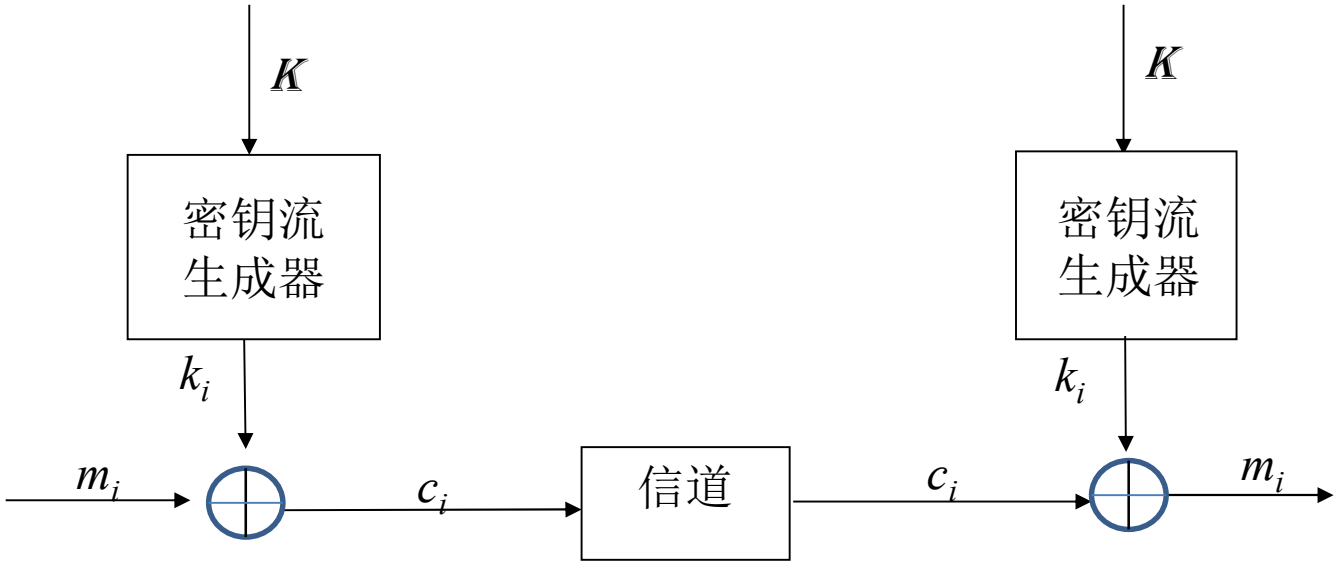
北京郵電大學 BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

完善保密 ➔ 随机密钥长度大于等于明文长度

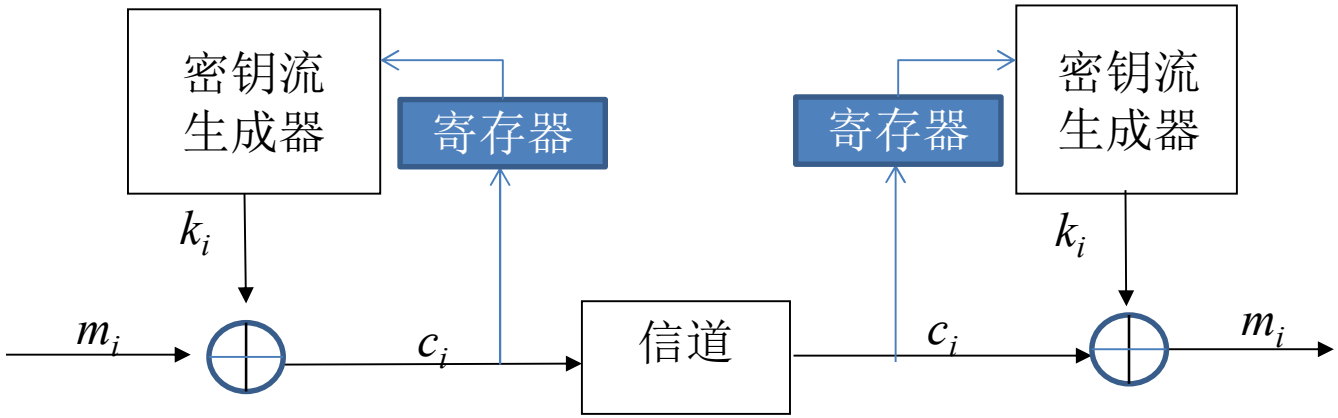
解决方法➔有限的密钥生成长的密钥序列

流密码的核心➔伪随机数发生器(PRG)

OTP与伪随机数发生器

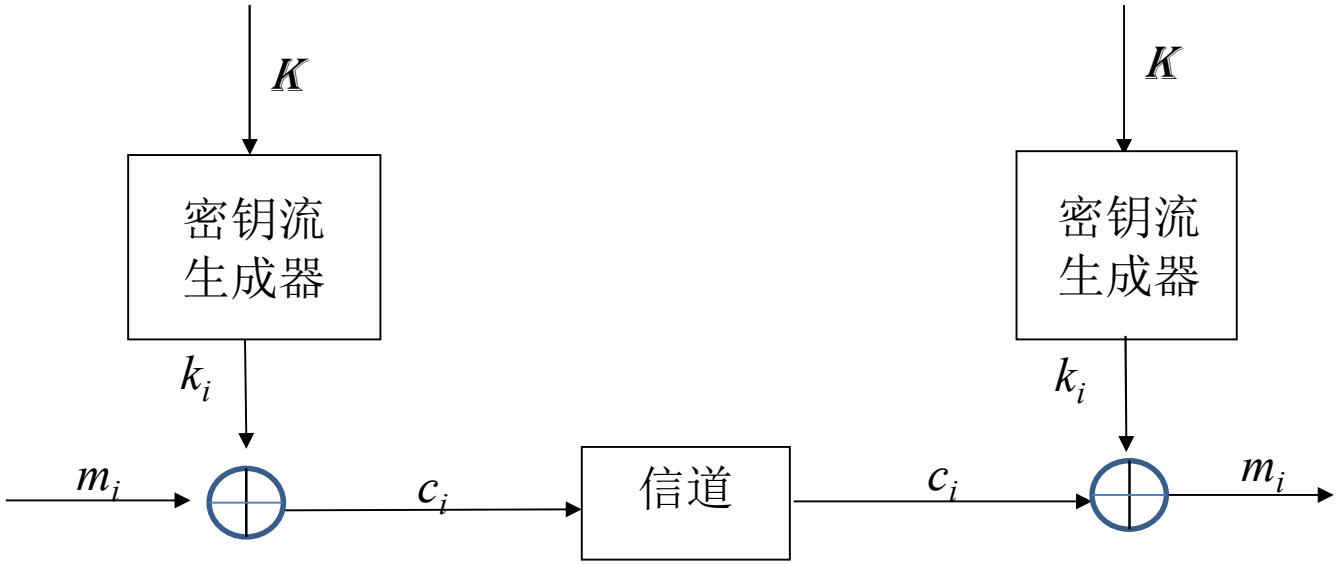


OTP与伪随机数发生器



- 自同步流密码的优点：有限错误传播，即使接收端和发送端不同步，只要接收端能连续地接受到n个正确的密文符号，就能重新建立同步。
- 缺点：评估自同步流密码的安全性困难得多

OTP与伪随机数发生器



- 在同步流密码中，密（明）文符号是独立的，一个错误传输只会影响一个符号，不影响后面的符号。
- 缺点：一旦接收端和发送端的种子密钥和内部状态不同步，解密就会失败，两者必须立即借助外界手段重新建立同步。

OTP与伪随机数发生器

北京郵電大學 BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

课堂练习：假设j=n/4,n为分组长度

对于DES，n=64，j=16；对AES，n=128,j=32

CFB模式为（　　）流密码？

CTR模式为（　　）流密码？

自同步、同步

本章主要内容

- OTP与伪随机数生成器
- 流密码技术的发展
- 基于LFSR的PRG
- RC4算法
- Estream 算法举例
- PRG安全与流密码安全应用

流密码技术的发展及分类

Profile 1 (SW)	Profile 2 (HW)
CryptMT (CryptMT Version 3)	**DECIM** (DECIM v2 and DECIM-128)
Dragon	**Edon80**
HC (HC-128 and HC-256)	**F-FCSR** (F-FCSR-H v2 and F-FCSR-16)
LEX (LEX-128, LEX-192 and LEX-256)	**Grain** (Grain v1 and Grain-128)
NLS (NLSv2, encryption-only)	**MICKEY** (MICKEY 2.0 and MICKEY-128 2.0)
Rabbit	**Moustique**
Salsa20	**Pomaranch** (Pomaranch Version 3)
SOSEMANUK	**Trivium**

流密码技术的发展及分类

Profile 1 (SW)	Profile 2 (HW)
HC-128	F-FCSR-H v2
Rabbit	Grain v1
Salsa20/12	MICKEY v2
SOSEMANUK	Trivium

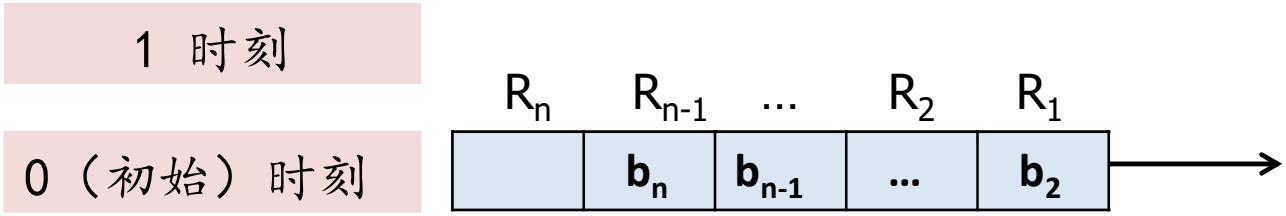
Profile 1 (SW)	Profile 2 (HW)
HC-128	~~F-FCSR-H v2~~
Rabbit	Grain v1
Salsa20/12	MICKEY v2
SOSEMANUK	Trivium

国标	
【标准编号】	GM/T 0001.1-2012
【标准名称】	祖冲之序列密码算法

本章主要内容

- OTP与伪随机数生成器
- 流密码技术的发展
- 基于LFSR的PRG
- RC4算法
- Estream 算法举例
- PRG安全与流密码安全应用

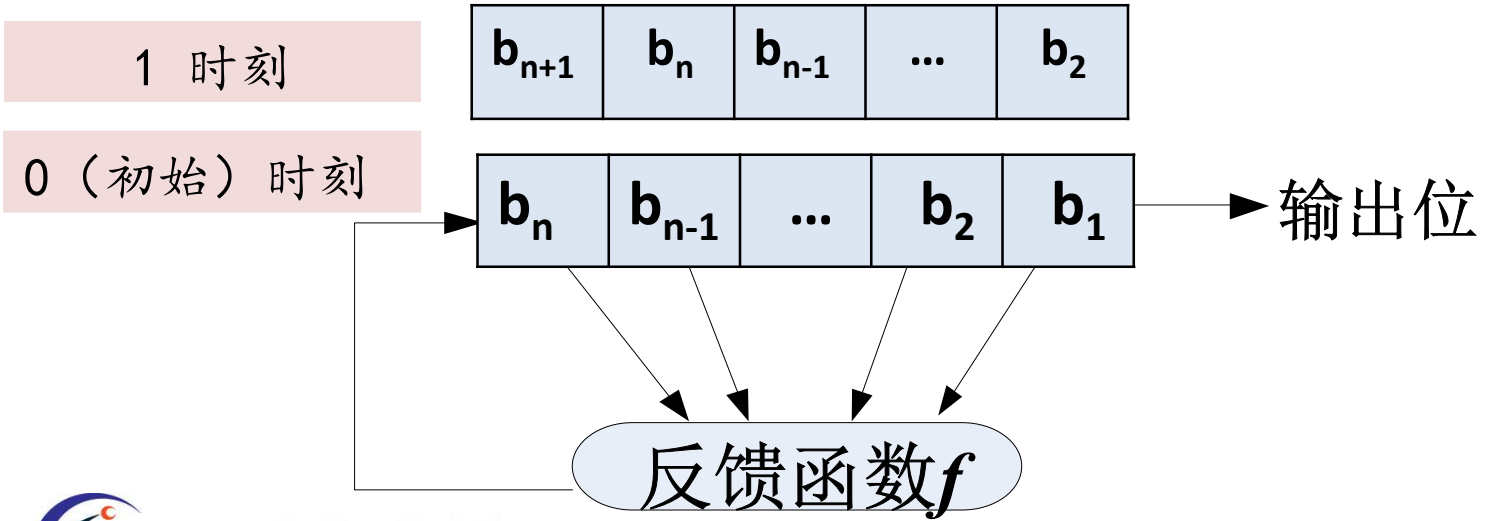
基于LFSR的PRG



- 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.
- 移位寄存器是指有n个寄存器（称为n-级移位寄存器）$R_1, R_2, \cdots, R_n$从右到左排列，每个寄存器中能存放1位二进制数，所有寄存器中的数可以统一向右（或向左）移动1位，称为进动1拍. 即R_1的值(b_1)右移1位后输出，然后R_2的值(b_2)送R_1 ，R_3的值(b_3)送R_2, ……最后，R_n的值(b_n)送R_{n-1}.

基于LFSR的PRG

- 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$(b_1, b_2, \cdots, b_n)$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$a_1, a_2, \cdots, a_m$.



基于LFSR的PRG

解：对应的n级LFSR的反馈函数为

$$b_{i+5} = b_{i+4} \oplus b_{i+3} \oplus b_{i+2} \oplus b_{i+1} \quad (i \geq 0).$$

状态		输出位
0001	→	1
1000	→	0
1100	→	0
0110	→	0
0011	→	1
0001	→	1
1000	→	0
1100	→	0

输出序列的周期为5

基于LFSR的PRG

- 线性反馈移位寄存器LFSR(linear feedback shift register）的反馈函数为线性函数
- 密钥流 $\{k_i\}$的周期一定要大

- 课堂练习：

初始状态分别为：$1001_D, 0010_D, 1111_D$

反馈函数： $b_{i+5} = b_{i+4} \oplus b_{i+3} \oplus b_{i+2} \oplus b_{i+1} \quad (i \geq 0).$

求输出序列？

10001,01001,11110

- n级LFSR输出的序列的周期r不依赖于寄存器的初始值，而依赖于反馈函数，或者说特征多项式$p(x)$

基于LFSR的PRG

- n级LFSR输出的序列的最大周期是$2^n - 1$
- LFSR的寄存器状态遍历$2^n - 1$个非零状态
- 初始状态为全零，则输出序列为0的循环

定义 当LFSR的寄存器状态遍历$2^n - 1$个非零状态时，序列的周期达到最大$2^n - 1$，这种序列被称为m序列。

基于LFSR的PRG

北京郵電大學 BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

定义：

设n级LFSR的反馈序列$\{b_i\}$满足递推关系

$$b_{i+n+1} = c_n b_{i+n} \oplus c_{n-1} b_{i+n-1} \oplus \cdots \oplus c_1 b_{i+1} \quad (i \geq 0).$$

这种递推关系可用一个一元高次多项式

$$p(x) = c_1 x^n + c_2 x^{n-1} + \cdots + c_n x + 1$$

表示，称这个多项式为LFSR的特征多项式。

课堂练习：反馈函数为 $b_{i+5} = b_{i+4} \oplus b_{i+3} \oplus b_{i+2} \oplus b_{i+1} \quad (i \geq 0).$
求对应的特征多项式？

基于LFSR的PRG

北京郵電大學 BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

定义 设 $p(x)$是 $GF(2)$上的多项式，使$p(x)|(x^L-1)$的最小的L 称为 $p(x)$ 的周期或者阶。

例：

$p(x)=x^4+x^3+x^2+x+1$ 为GF(2)上多项式，

以它为特征多项式的LFSR 的输出序列周期。

$$(x^5-1)=(x^4+x^3+x^2+x+1)(x-1)=p(x)(x-1)$$

$$p(x)\nmid x^L-1,\quad L<5$$

所以p(x)的周期为5

基于LFSR的PRG

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

定理 $\{k_i\}$是周期为2^n-1的m-序列的充要条件是其特征多项式$p(x)$为n阶本原多项式

基于LFSR的PRG

一个3-级的反馈移位寄存器，反馈函数$b_{i+4}=b_{i+3} \oplus b_{i+1}$，初态为100，输出序列？

生成多项式为：　$p(x)=x^3+x+1$

$$(x^7-1)=(x^4+x^2+x+1)(x^3+x+1)$$

$$p(x) \nmid x^L-1, \quad L<7$$

所以$p(x)$的周期为7

基于LFSR的PRG

初态为100放入寄存器，输出序列情况如下：

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

输出序列的周期为7，是m序列

基于LFSR的PRG

$$\begin{cases} k_{n+1} = b_1k_1 + b_1k_1 + \cdots + b_nk_n \\ k_{n+2} = b_1k_2 + b_1k_3 + \cdots + b_nk_{n+1} \\ \vdots \\ k_{2n} = b_1k_n + b_1k_{n+1} + \cdots + b_nk_{2n-1} \end{cases}$$

已知连续的$2n$比特输出序列$k_1, k_2, \ldots, k_{2n}$，可以构造n个线性方程包含n个未知数$b_1, b_2, \ldots, b_n$，所以可解出惟一b_i $(i = 0, 1, \cdots, n)$

n级LFSR，获得连续的$2n$比特输出序列$k_1, k_2, \ldots, k_{2n}$，

- ✓ 可恢复该线性反馈移位寄存器的反馈函数$\{b_i\}$；
- ✓ 进而推导之后的密钥序列$k_{2n+1}, \cdots$；
- ✓ 可以推导初始状态（初始密钥）。

基于LFSR的PRG

例：

5级线性反馈移位寄存器产生的密钥序列加密前的明文串为011001111111001，加密后的密文串为101101011110011。求该LFSR的反馈函数。

解：由明密文得相应的密钥序列为

1	1	0	1	0	0	1	0	0	0	0	1	0	1	0
k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8	k_9	k_{10}	k_{11}	k_{12}	k_{13}	k_{14}	k_{15}

利用前10个密钥序列比特建立如下方程：

基于LFSR的PRG

$$\begin{pmatrix} k_6 \\ k_7 \\ k_8 \\ k_9 \\ k_{10} \end{pmatrix} = \begin{pmatrix} k_1 & k_2 & k_3 & k_4 & k_5 \\ k_2 & k_3 & k_4 & k_5 & k_6 \\ k_3 & k_4 & k_5 & k_6 & k_7 \\ k_4 & k_5 & k_6 & k_7 & k_8 \\ k_5 & k_6 & k_7 & k_8 & k_9 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \end{pmatrix} \Leftrightarrow \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \Leftrightarrow \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

$$k_{i+6} = c_1 k_{i+1} + c_4 k_{i+4} = k_{i+1} + k_{i+4}$$

基于LFSR的PRG

1969年B-M算法的出现，从此m序列只能作为密钥生成器的一部分，而不能直接作为密钥流使用

- 给定长为N的序列$a = (a_0, a_1, \cdots, a_{N-1})$，用B-M算法可以得到$(p_N(x), l_N)$，则以$p_N(x)$为生成多项式，长为$l_N$的LFSR是生成序列$a$的最短LFSR。一旦给定了序列$a$，那么生成它的最短LFSR的长度$l_N$就确定了。
- 只需要捕获序列a连续的$2l_N$个比特，就能得到它的唯一解$(p_N(x), l_N)$，以$p_N(x)$为特征多项式的l_N级LFSR就可以生成整个序列a。

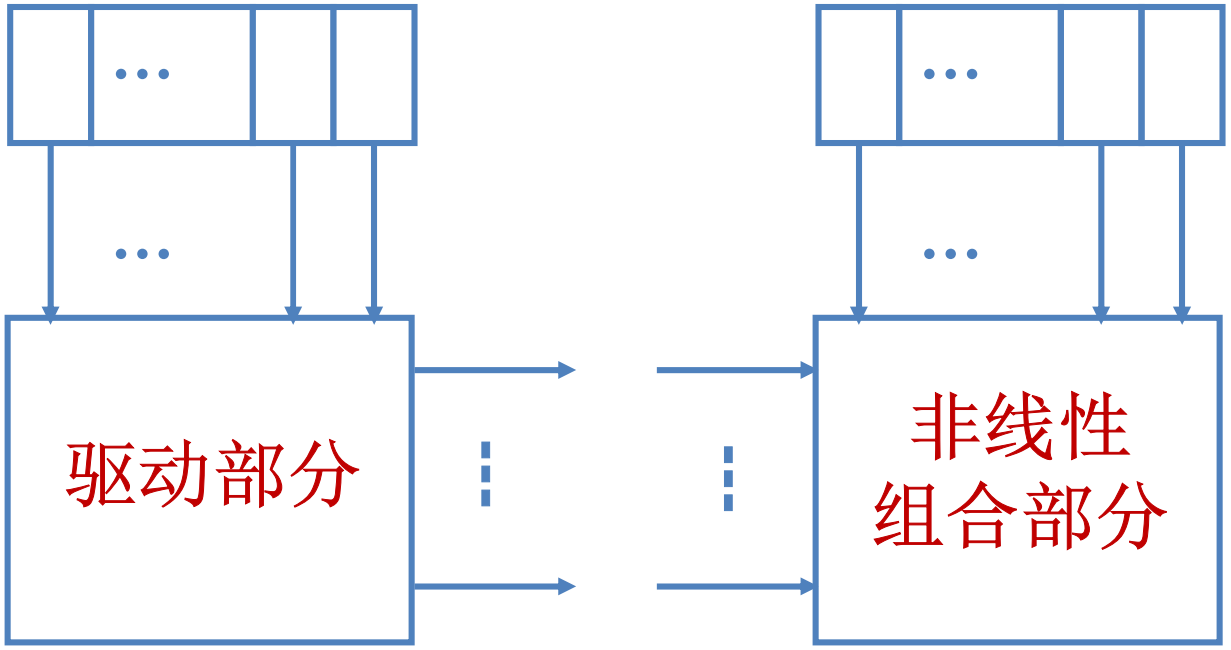
基于LFSR的PRG

思考：已知n，已知反馈系数 b2,..., bn（标准化需求），获得连续？？比特密钥序列，可以解密所有密文。

思考：已知连续？？比特明文序列及其对应密文序列，可以推导初始状态（初始密钥），继而解密所有密文比特。

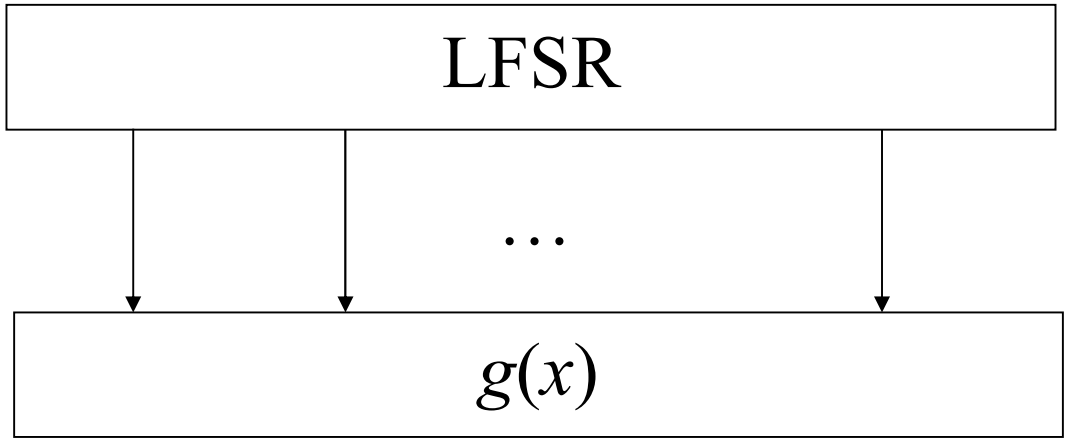
基于LFSR的PRG

为了提高密钥流序列的线性复杂度，密钥生成器重中必须使用非线性函数. 为了便于分析，Ruppe将密钥流生成器分成两部分：驱动部分和非线性组合部分.



基于LFSR的PRG

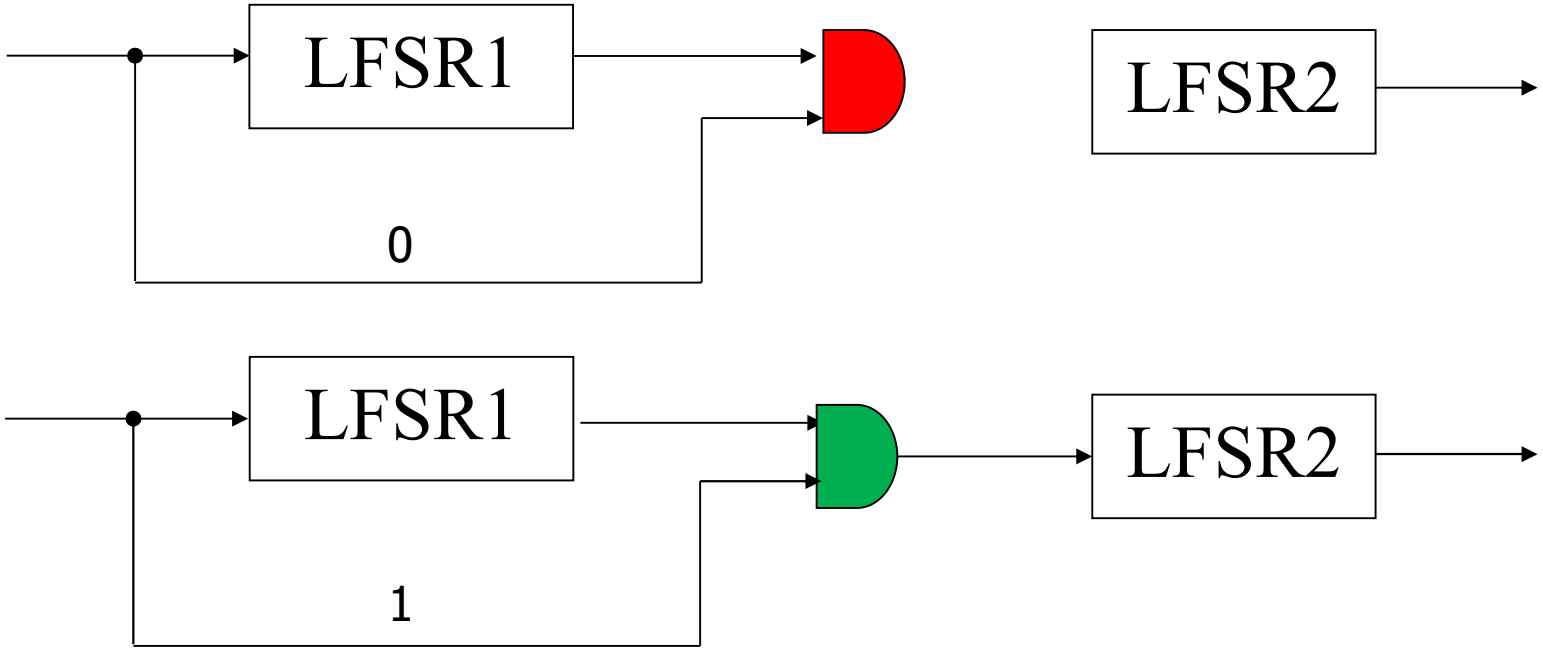
- 滤波生成器又叫前馈生成器，一般由LFSR和滤波前馈）函数两部分组成. LFSR可以是一个，也可以是几个，它们输出的序列共同作为滤波函数的输入.



滤波函数要求具有很好的非线性性质，以增强生成器的抗攻击能力.

基于LFSR的PRG

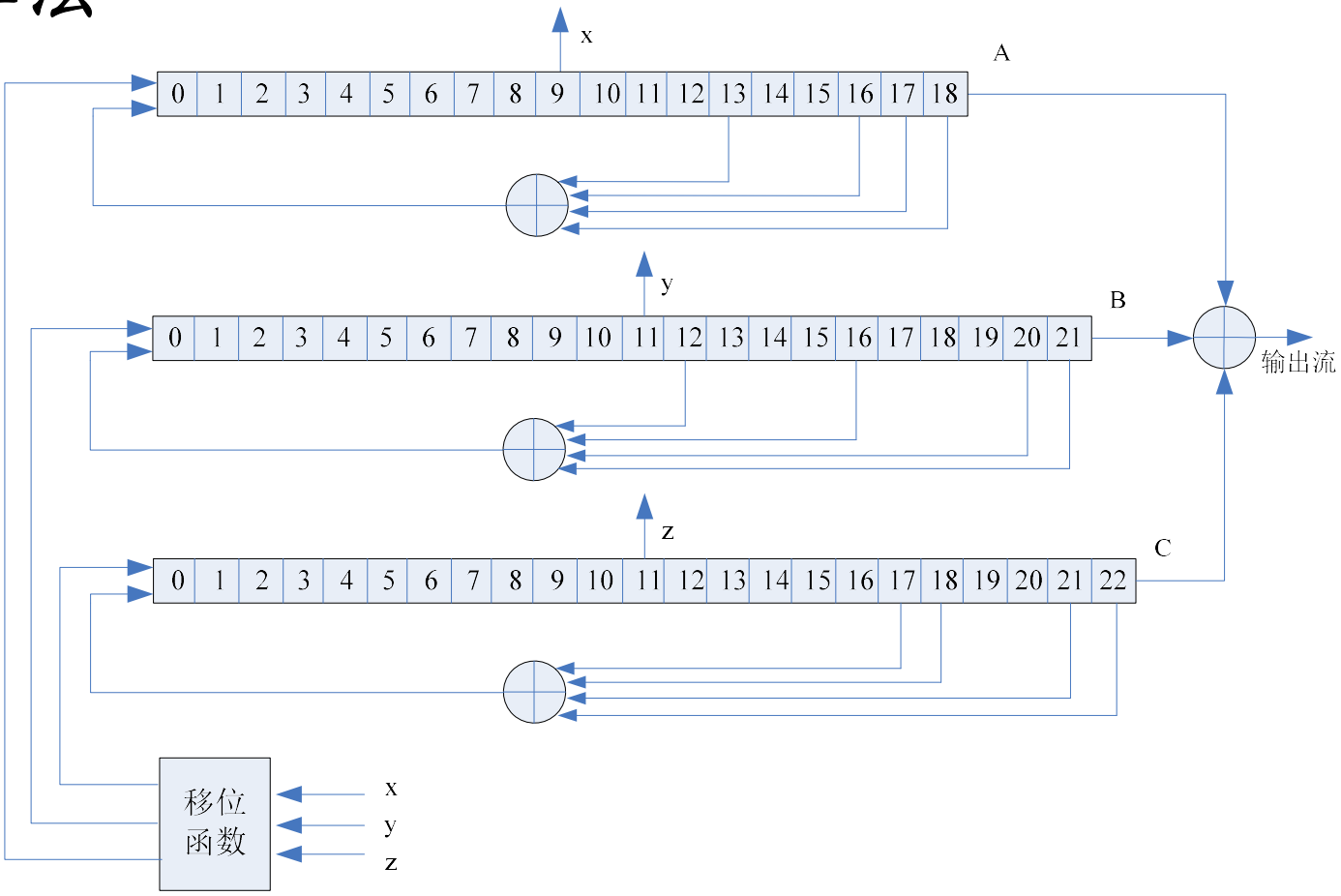
- **钟控生成器**是由一个或几个FSR输出序列，控制一个FSR的时钟。最简单的钟控生成器是用一个LFSR控制另一个LFSR的时钟脉冲，如图：



当LFSR1输出1时，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，时钟脉冲无法通过与门使LFSR2移位（走），从而LFSR2重复输出前一位（停）。因此，这种钟控生成器也被形象地称之为**走停生成器**（Sstop-and-Ggo generator）。

基于LFSR的PRG

- A5算法



本章主要内容

- OTP与伪随机数生成器
- 流密码技术的发展
- 基于LFSR的PRG
- RC4算法
- Estream 算法举例
- PRG安全与流密码安全应用

RC4算法

- RC4是由Rivest于1987年设计，被广泛应用于Windows、Lotus Notes和其它软件，还被用于安全套接字（SSL）和无线通信系统等。

- RC4适于软件实现，加密速度比DES大约快10倍. RC4可以支持不同密钥长度，美国政府特别限定，用于出口的RC4的密钥长度不得超过40位.

- RC4使用了一个256字节大小的非线性数据表（简称S表），依据表进行非线性变换，得到密钥流. S表的值 S_0，S_1，…，S_{255}是数字0到255的一个排列，RC4有两个计数器 *I*和*J*，初值都为0.

RC4算法

- RC4首先进行S表的初始化，过程如下：
- 对S表进行填充：$S_i=i$，$0 \leqslant i<255$；
- 用密钥填充另一个256字节的数组K，如果密钥长度小于256字节，则依次重复填充，直至填满这个数组：K_0，K_1，…，K_{255}；
- $j=0$；
- 对于i=0到255重复以下步骤：
  - $j=j+S_i+K_i$（mod 256）；
  - 交换S_i和S_j。
- RC4按下列步骤输出密钥流的一个字节z：
- $i=0$，$j=0$；
- $i=i+1$（mod 256）；
- $j=S_i+j$（mod 256）；
- 交换S_i和S_j；
- $t=S_i+S_j$（mod 256）；
- $z=S_t$。

RC4算法

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

RC4算法

北京郵電大學 BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

密钥调度算法KSA

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)) mod 8=5

S数据表的第一个操作是将S(0)与S(5)互换。

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

RC4算法

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：

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

当该循环执行完后，数据表S就被随机化为：

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

RC4算法

北京郵電大學 BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

伪随机数生成算法PRGA

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)

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

RC4算法

然后如下计算t和k：

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

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。

OTP与伪随机数生成器

Golomb伪随机性测试

周期为r（r足够大）的0—1序列{si}

- r是奇数，则一个周期内0的个数比1的个数多一个或少一个；若r是偶数，则0的个数与1的个数相等.
- 在长度为r的周期内，长为1的游程的个数为游程总数的1/2，长为2的游程的个数占游程总数的$1/2^2$,…，长为i的游程的个数占总游程的$1/2^i$. 而且对于任意长度，0的游程个数和1的游程个数相等.

例：

0110111101中，4个游程长度为1，1个游程长度为2，1个游程长度为4

OTP与伪随机数生成器

北京郵電大學 BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

- 异相自相关函数是一个常数.

  设一个周期为r的序列

  $$s_1, s_2, \cdots, s_r, s_{r+1}, s_{r+2}, \cdots ,$$

  序列平移T位得到另外一个序列

  $$s_T, s_{T+1}, \cdots s_{r+T}, s_{r+T+1}, \cdots,$$

  若$s_i = s_{i+T}$，则称对应第i位相等。

  设两个序列相同位的个数为n，不同位的个数为d，则R(T)=(n-d)/r为自相关函数

国标	
【标准编号】	GM/T 0005-2012
【标准名称】	随机性检测规范
【标准编号】	GB/T 32915
【标准名称】	信息安全技术--二元序列随机数检测方法

RC4算法及其应用

- 问题

$$C_1 \leftarrow m_1 \oplus PRG(k)$$

$$C_2 \leftarrow m_2 \oplus PRG(k)$$

窃听者获得C_1 和C_2

$$C_1 \oplus C_2 \rightarrow m_1 \oplus m_2$$

根据明文冗余可以猜测$m_1 \oplus m_2$

RC4算法及其应用

北京郵電大學 BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

- 例. 攻击者窃听公开信道，截获到2个密文

c_1：_sa痉す?↓杀?(犹烫烫烫蔌 5F7361BEB7A4B9F72F19C9B1CBFF28D3

c_2：#*3恣溘醑L潯僵m绿烫烫烫? 232A33EDA7E4E6E16D4C9DA183A26DC2

猜测m1： **Nice to see you!** 4E69636520746F2073656520796F7521

3230313630343036313031303130323030

则$m_2=c_1 \oplus c_2 \oplus m_1$

根据ASCI编码得m_2：2016040610001200

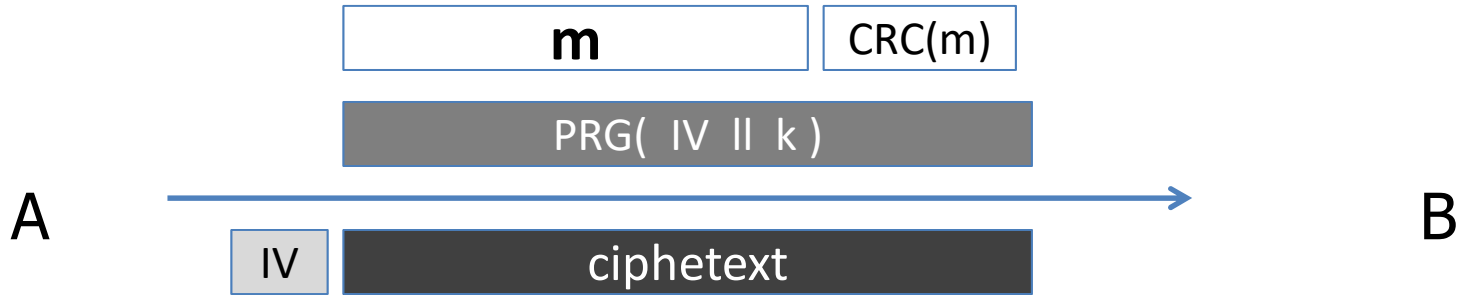
密钥不可重复使用

RC4算法及其应用

802.11b是无线局域网(WLAN)的安全性协议，WEP对在两台设备间无线传输的数据进行加密

IV长度：　　24 bits

- $2^{24} \approx 16M$ frames，IV重复
- 某些802.11 系统实现时，系统重启后IV自动重置为 0



本章内容小结

- OTP与伪随机数生成器
- 流密码技术的发展
- 基于LFSR的PRG
- RC4算法

THE END !