第三讲　常规加密

--- 分组密码

上海交通大学安全学院
郑燕飞

分组密码的原理

- 流密码和分组密码
- 当前对称加密算法几乎都基于Feistel分组密码
- 乘积密码
  - 连续执行两个或多个密码
  - Feistel提出用替代和置换交替的方式构造密码
  - Shannon提出用扰乱和扩散交替的方式构造密码
- 扩散和扰乱
  - 目的：挫败基于统计分析的密码分析方法
  - 扩散：明文的统计结构扩散到密文中
  - 扰乱：密文的统计特性与密钥的取值之间的关系尽可能复杂

Feistel密码

- Feistel加密

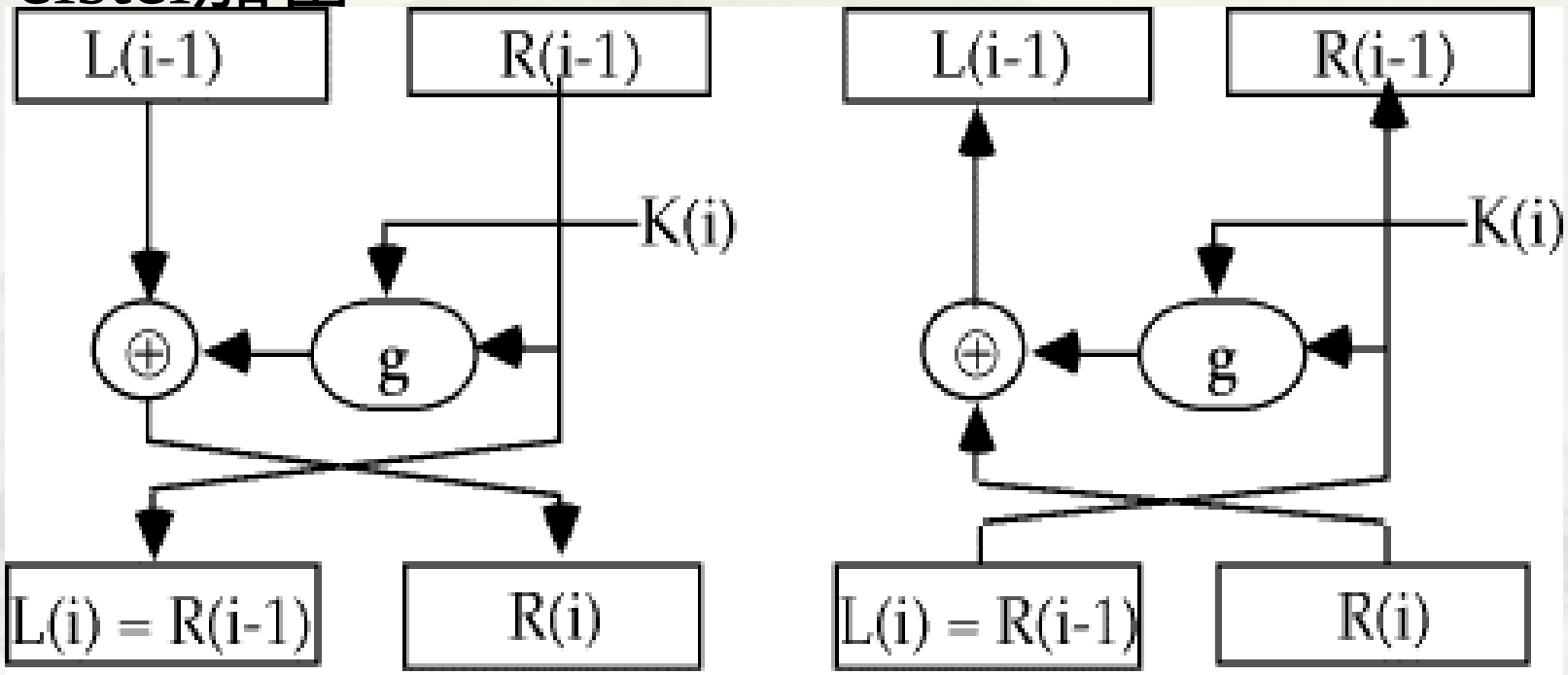


Fig 2.4 - A Round of a Feistel Cipher

Feistel密码

- 变换可以用下列函数表示:
- *L(i) = R(i-1)*
- *R(i) = L(i-1) XOR g(K(i), R(i-1))*
- 求逆很容易
- 实际中，一些这样的连续变换形成完整密码变换（典型：16轮）

Feiste

* Feistel解密

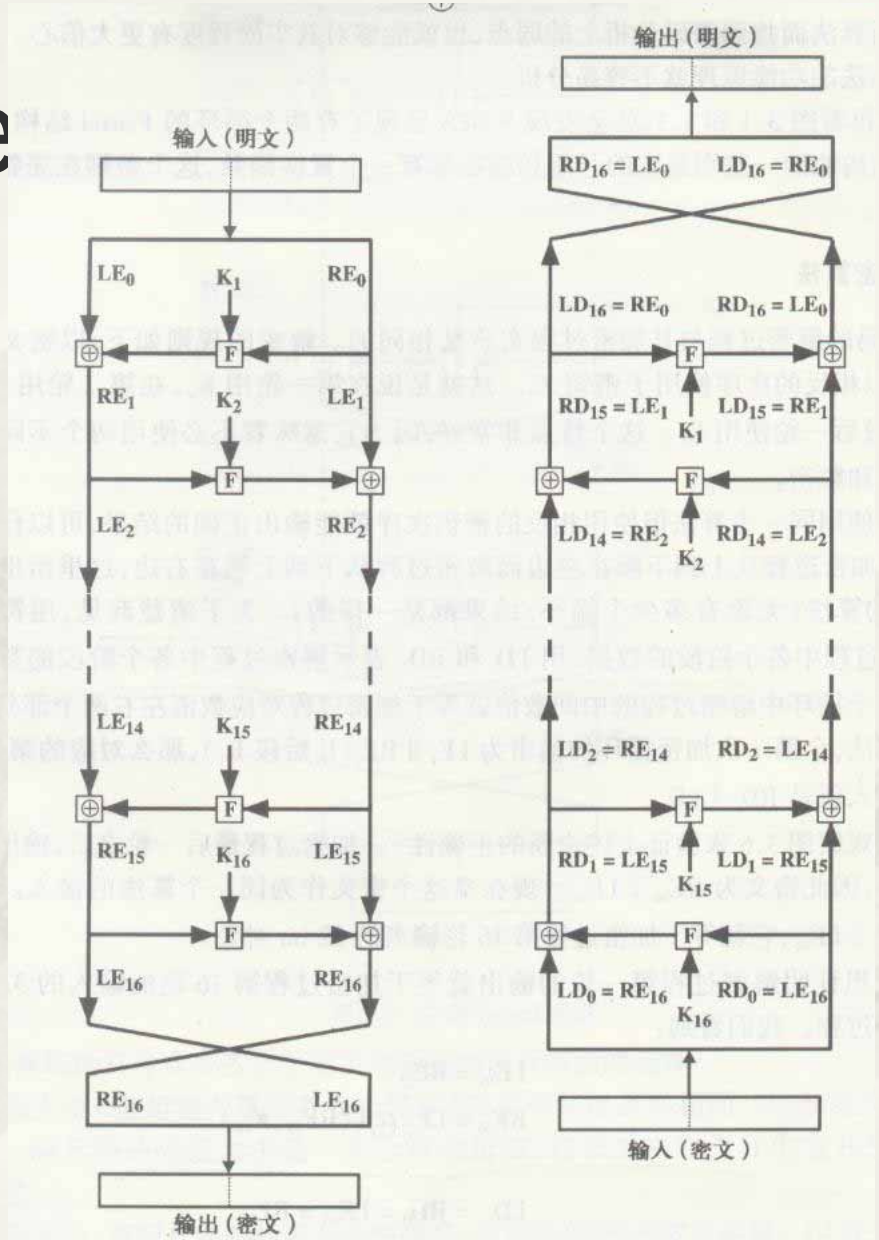


图 3.6 Feistel 加密和解密

Feistel密码的设计准则

- 分组大小
  - 增加分组长度会提高安全性, 但降低了密码运算速度
- 密钥大小
  - 增加密钥长度,可以提高安全性(使得穷搜索困难),同样,降低了密码速度
- 循环次数
  - 增加轮数可以提高安全性,但降低速度
- 子密钥产生算法
  - 子密钥生成越复杂,就越安全,但降低速度
- Round函数
  - 复杂的轮函数能够使的密码分析困难,但降低速度

Feistel密码的设计准则

- 所有问题就是平衡问题
  - 设计”安全”的密码算法并不难,只要使用足够多的轮数就可以,但降低速度
- 快速的软件加解密
- 便于分析

ＤＥＳ（数据加密标准）

- ＤＥＳ简介
  - 1977年被美国标准局作为数据加密标准
  - 64bit分组加密，密钥长度为56bit
  - 对称密钥体制
- ＤＥＳ的由来
- 对ＤＥＳ的评论
  - 密钥长度比较短
  - 内部结构的设计标准保密

DES

* DES
一般描述

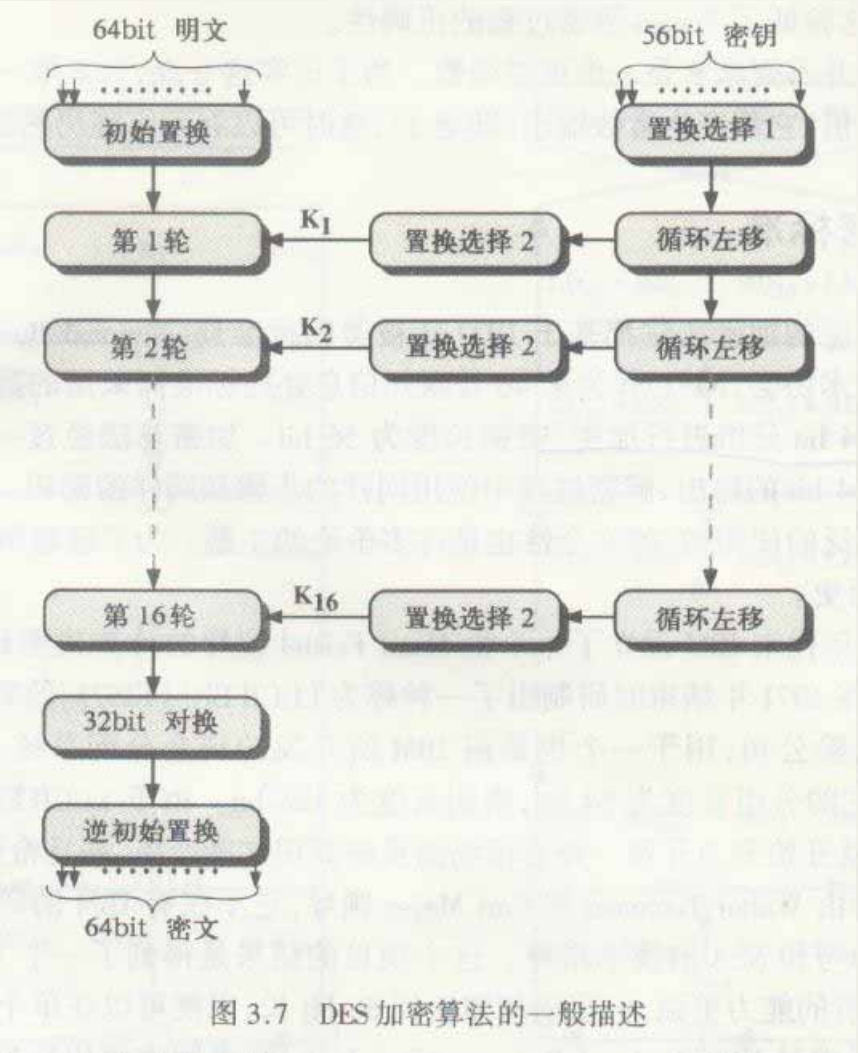
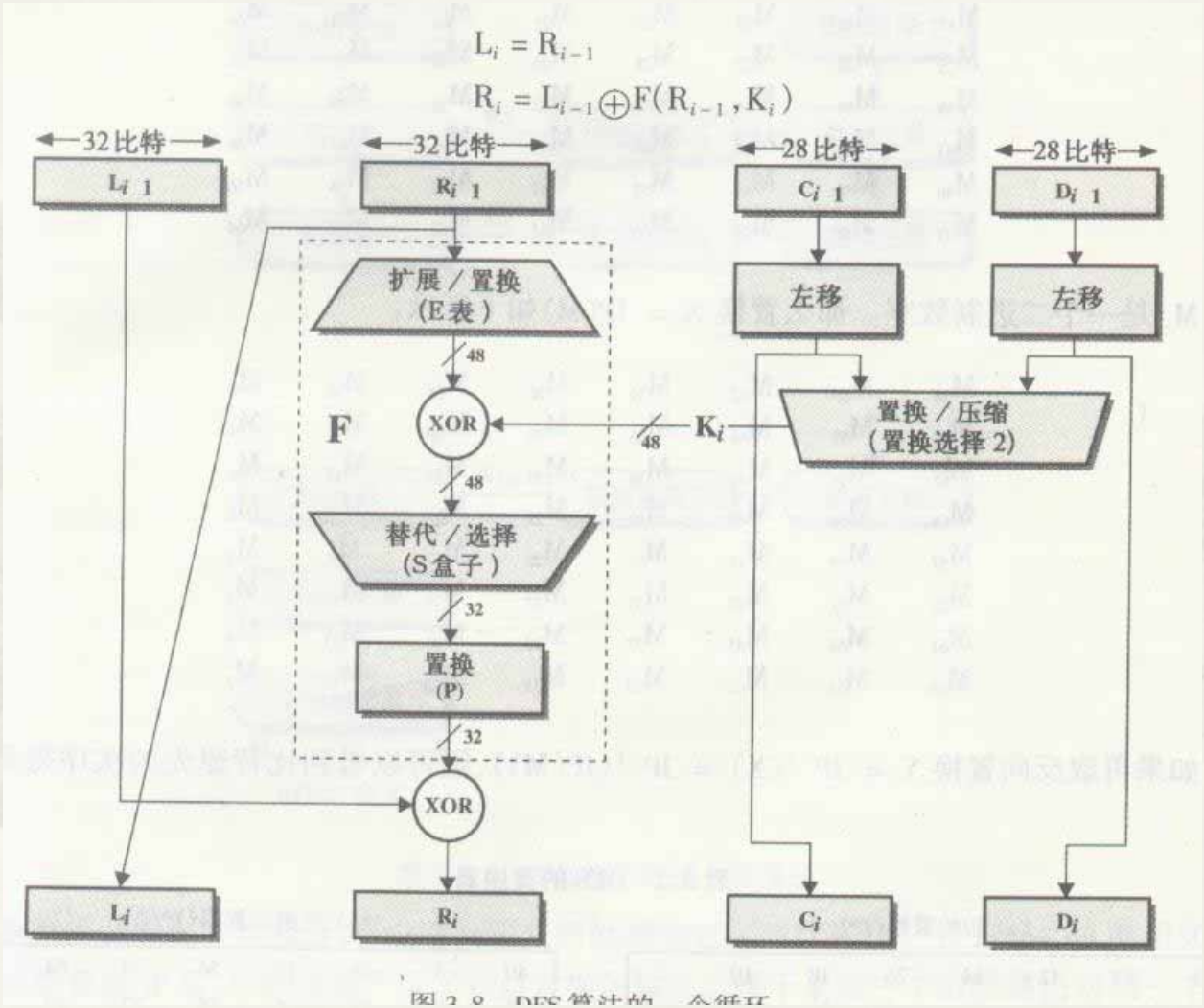


图 3.7 DES 加密算法的一般描述

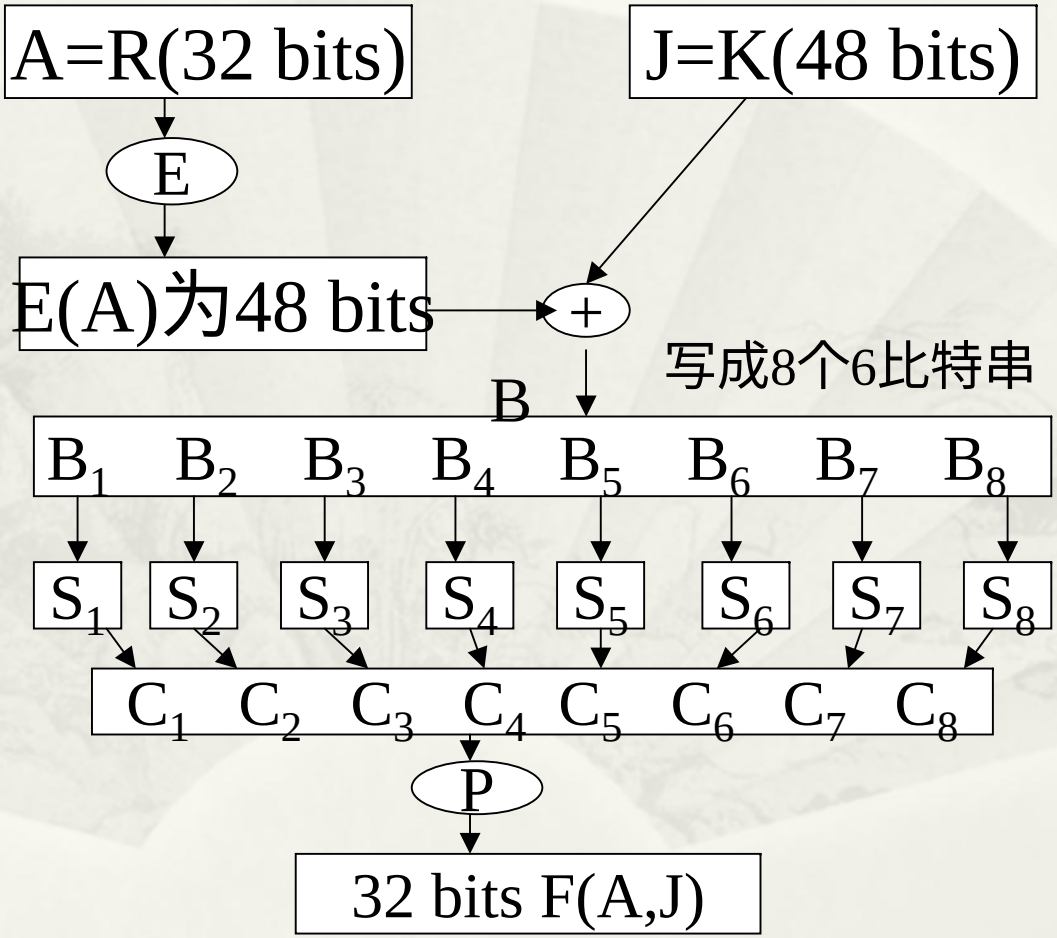
DES

* 每个循环的详细过程



DES

* F（R，K）

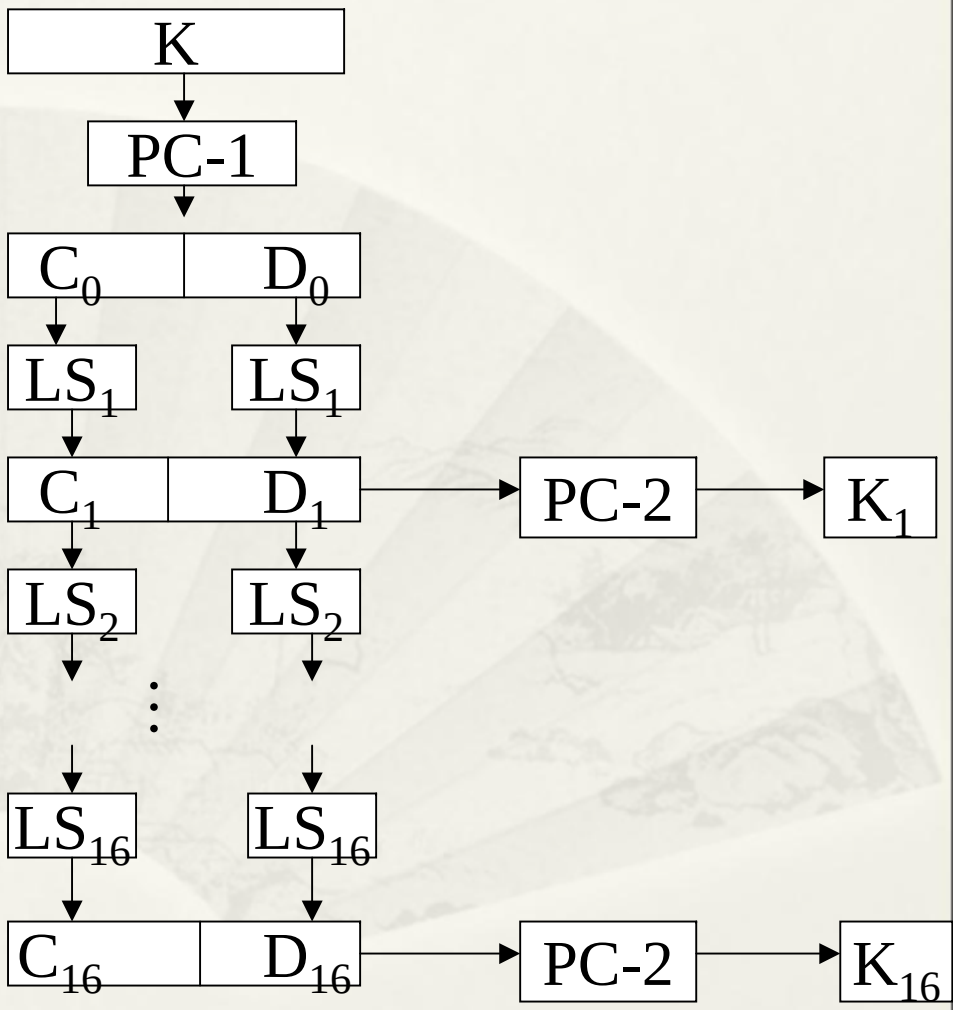

A=R(32 bits)
J=K(48 bits)
E
E(A)为48 bits
+
写成8个6比特串
B
B1 B2 B3 B4 B5 B6 B7 B8
S1 S2 S3 S4 S5 S6 S7 S8
C1 C2 C3 C4 C5 C6 C7 C8
P
32 bits F(A,J)

DES

- S 盒的作用
  - 6bit一》4bit
  - 输入的第一和最后两比特决定行
  - 输入的中间四比特决定列
  - 被选中的数字转换成4比特输出

DES

- 密钥的产生
  - 选择置换矩阵1
  - 循环左移一位或两位
  - 选择置换矩阵2
- 解密
  - 逆次使用子密钥



DES

- 雪崩效应

  明文或密钥的一点点变动应引起密文发生大的变化
- ＤＥＳ的雪崩效应

对ＤＥＳ的分析

- 密钥的大小
  - ＤＥＳ的密钥空间为2!56
- 算法的性质
  - 不公开S盒的设计准则

三重DES

* 双重DES

加密 $C = E_{K2}[E_{K1}[P]]$

解密 $P = D_{K1}[D_{K2}[C]]$

问题：下式成立吗？

$$E_{K2}[E_{K1}[P]] = E_{K3}[P]$$

* 中途攻击

三重DES

- 两个密钥的三重DES

$$C = E_{K1}[D_{K2}[E_{K1}[P]]]$$

- 目前，没有针对三重DES的攻击方法，它是一种较受欢迎的DES替代方案。

分组密码设计原理

- ＤＥＳ的设计准则（了解）
  - S盒

    提供输入bits混合作用 (confusion)
  - P盒

    提供扩散作用(diffusion)

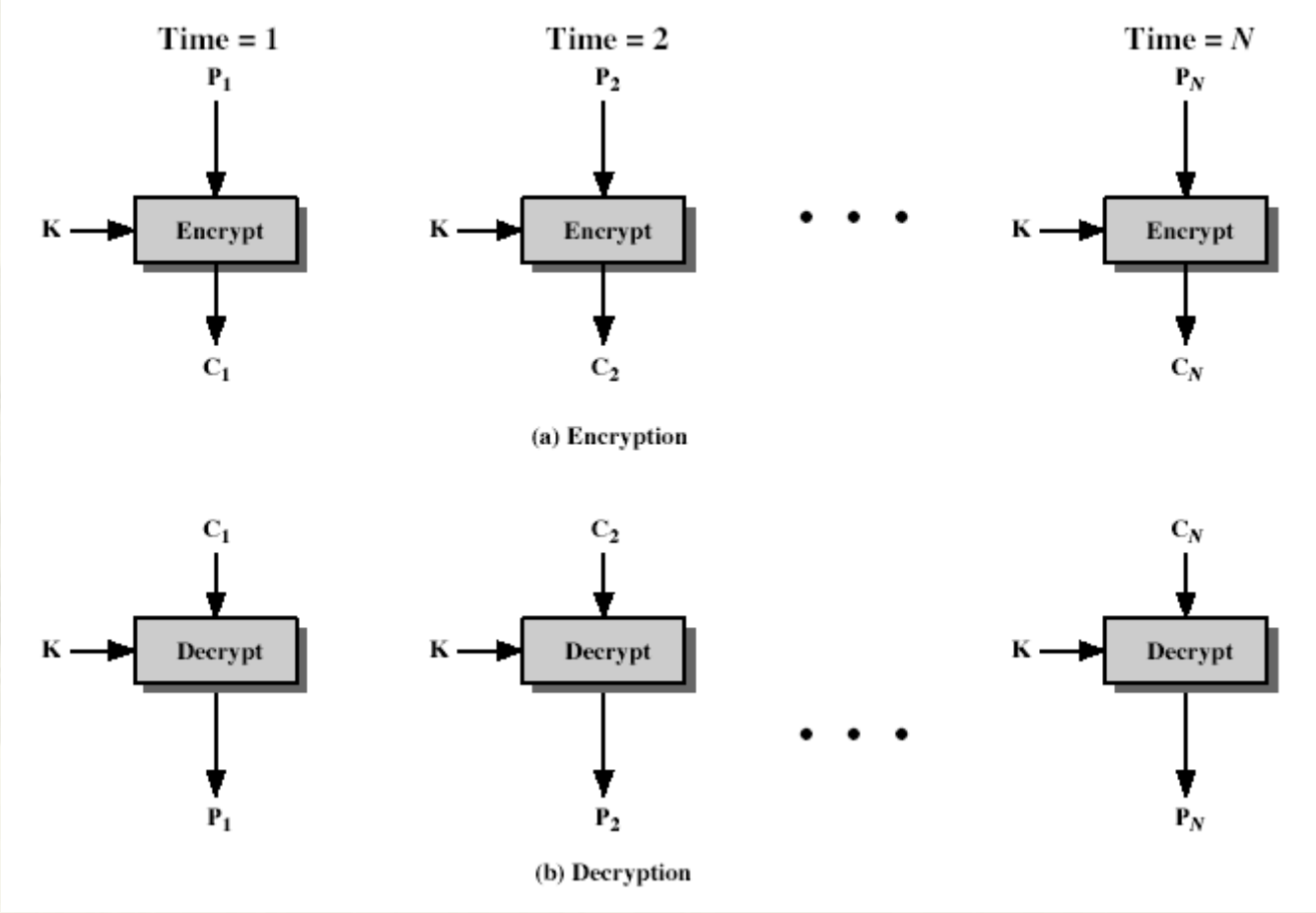
分组密码设计原理

- 循环次数
  - 以密码分析的工作量大小为依据
- 函数F的设计
  - 扰乱作用－F 应该是非线性的
  - SAC和BIC
    - SAC: 对于任何i,j,当然任何一个输入比特i变化时，一个S盒子的任何输出比特j变化的概率为1/2
    - BIC: 对于任意i,j,k,当任意输入比特变化时，输出j和k应当独立变化
  - 古典密码没有这些性质
- 密钥调度
  - 推测各子密钥和主密钥的难度尽可能大

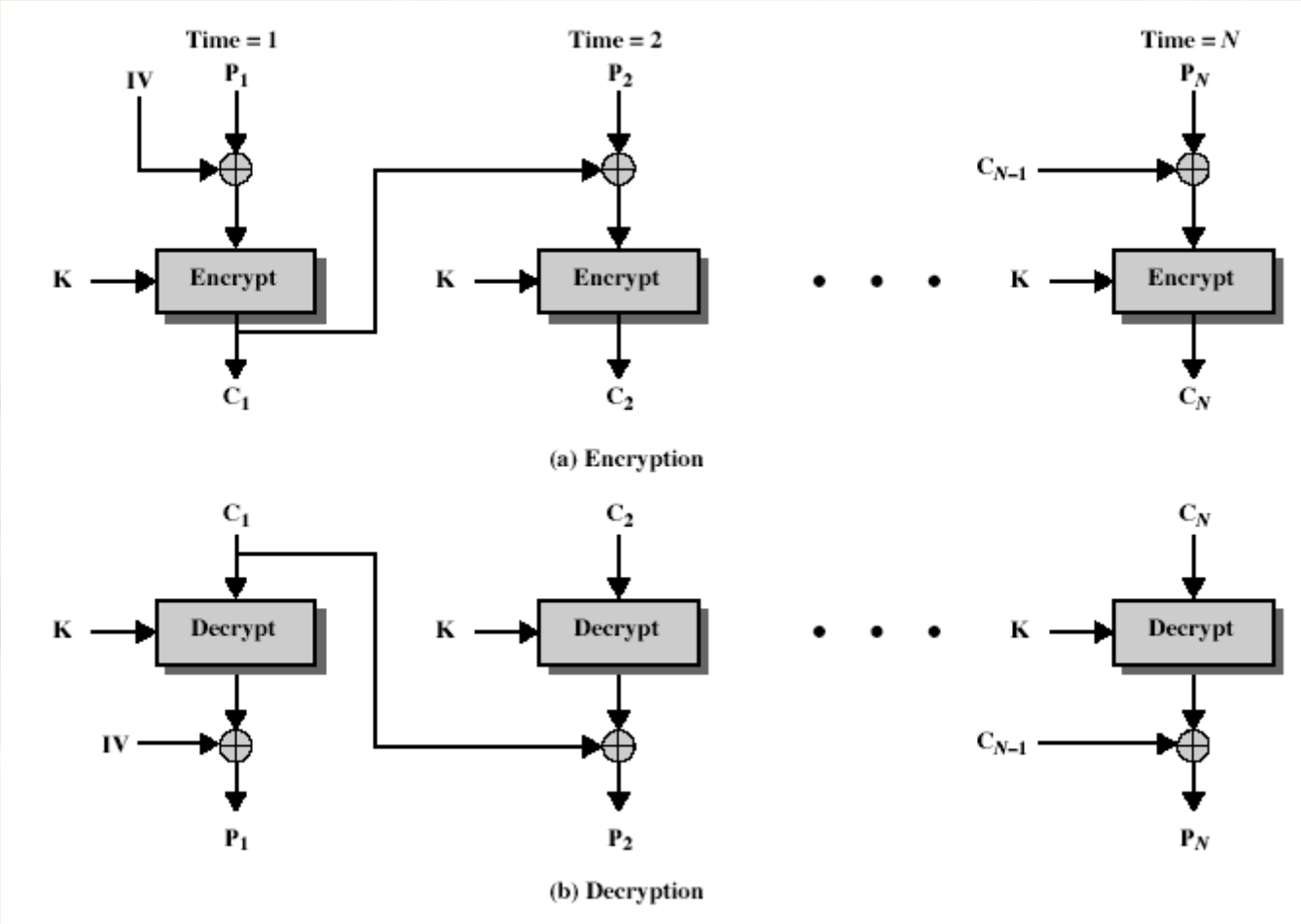
密码设计的评价

- “好的”密码设计具有: 雪崩特性,完备性,不可预料性(avalanche, completeness, unpredictability)
- 差的密码设计缺乏随机性,具有太大的可预料性
- 许多密码都被攻破 (incl. commercial products like Wordperfect, pkzip, all current mobile phone ciphers)
- 即使密码学专家也会犯这样的错误
- 最好的办法是测试, 通过实际检验证明它的安全性

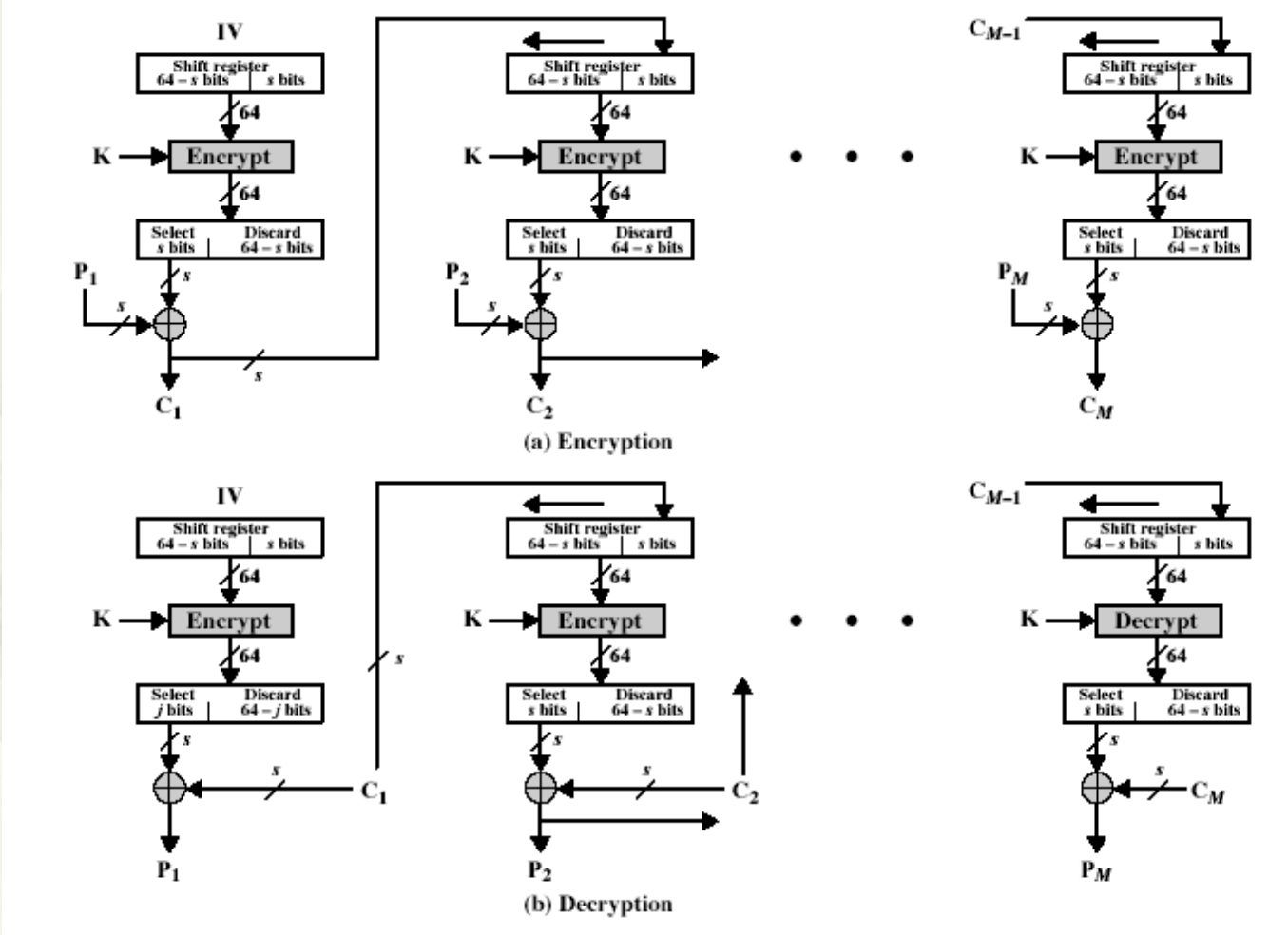
Electronic Codebook Book (ECB)



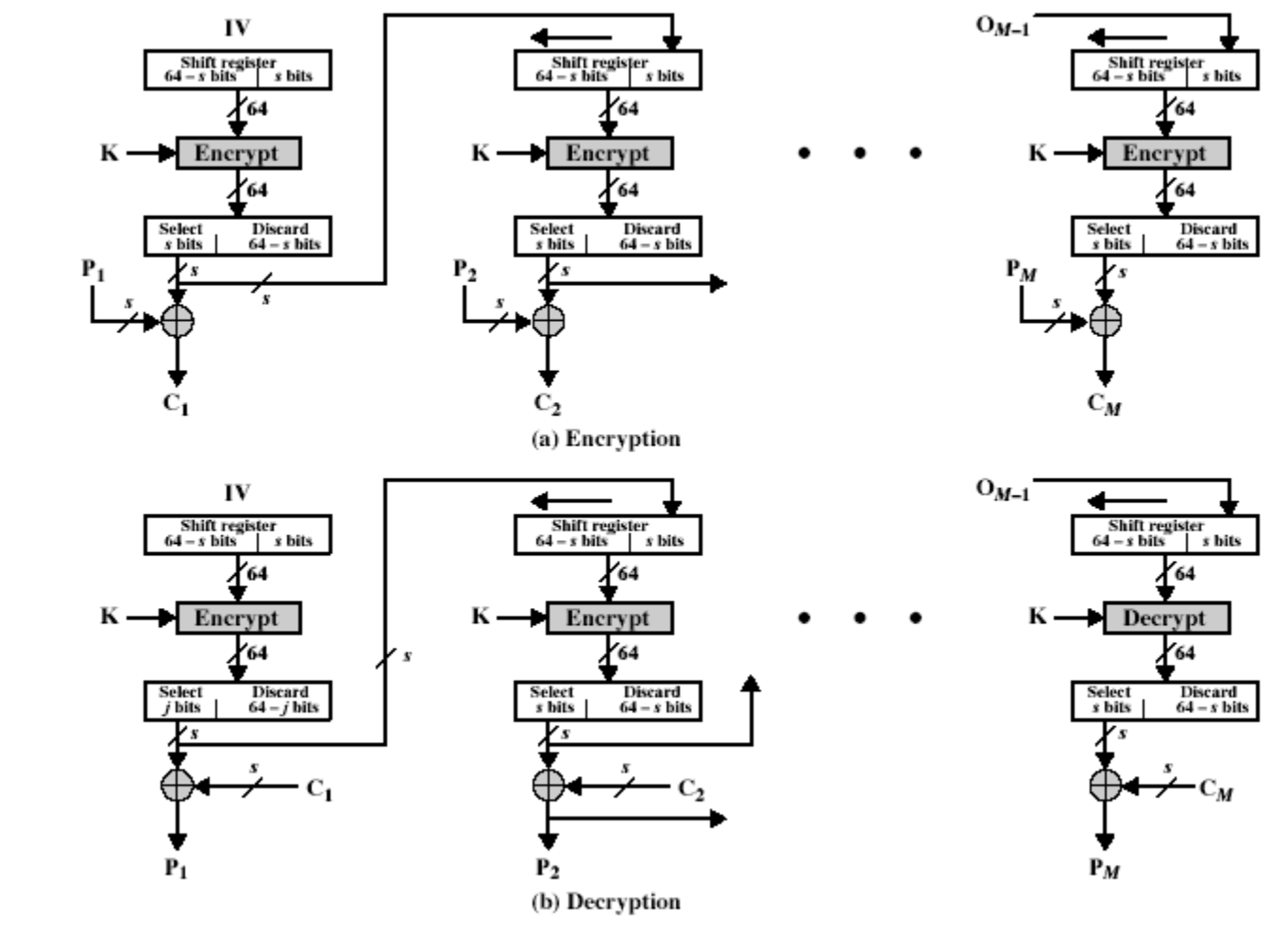
Cipher Block Chaining (CBC)



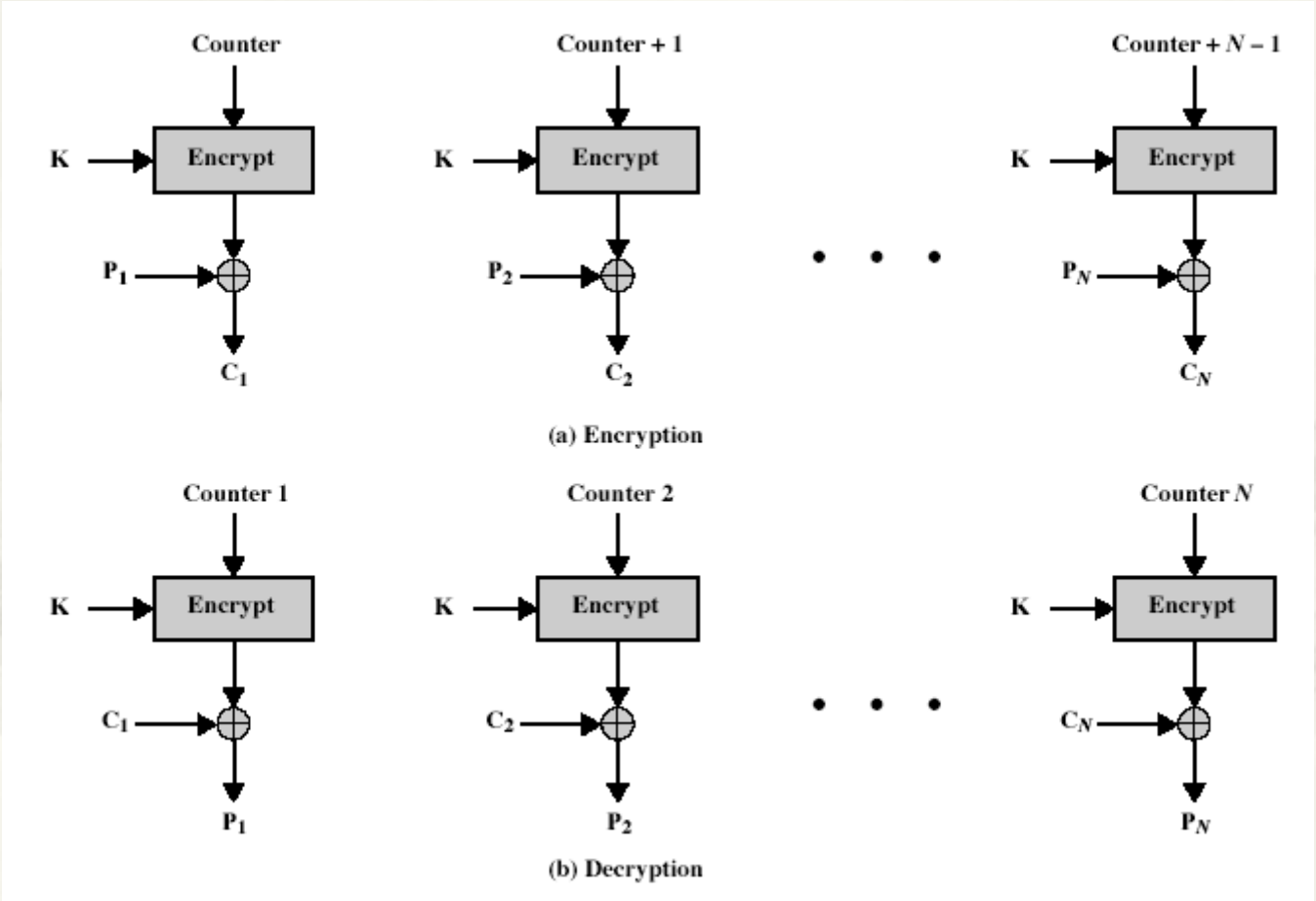
Cipher FeedBack (CFB)



Output FeedBack (OFB)



Counter (CTR)


Counter
Counter + 1
Counter + N – 1
K
Encrypt
P1
P2
PN
C1
C2
CN
(a) Encryption
Counter 1
Counter 2
Counter N
(b) Decryption

ＩＤＥＡ简介

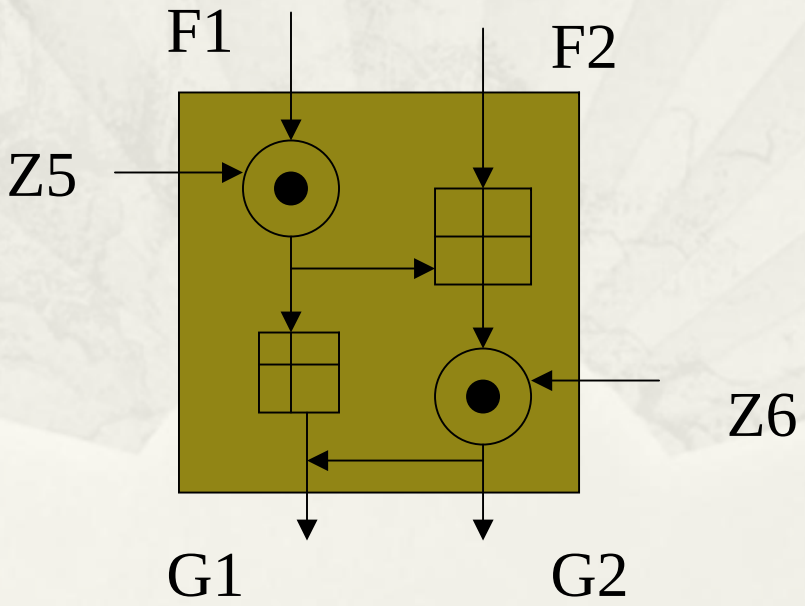
瑞士的Xuejia Lai和James Massey于1990年公布了IDEA密码算法第一版，称为PES (Proposed Encryption Standard)。为抗击差分密码攻击，他们增强了算法的强度，称IPES（Improved PES)，并于1992年改名为IDEA（International Data Encryption Algorithm，国际数据加密算法。）

ＩＤＥＡ

- IDEA是一个分组长度为64位的分组密码算法，密钥长度为128位（抗强力攻击能力比DES强），同一算法既可加密也可解密。
- IDEA的“混淆”和“扩散”设计原则来自三种运算，它们易于软、硬件实现（加密速度快）

ＩＤＥＡ简介

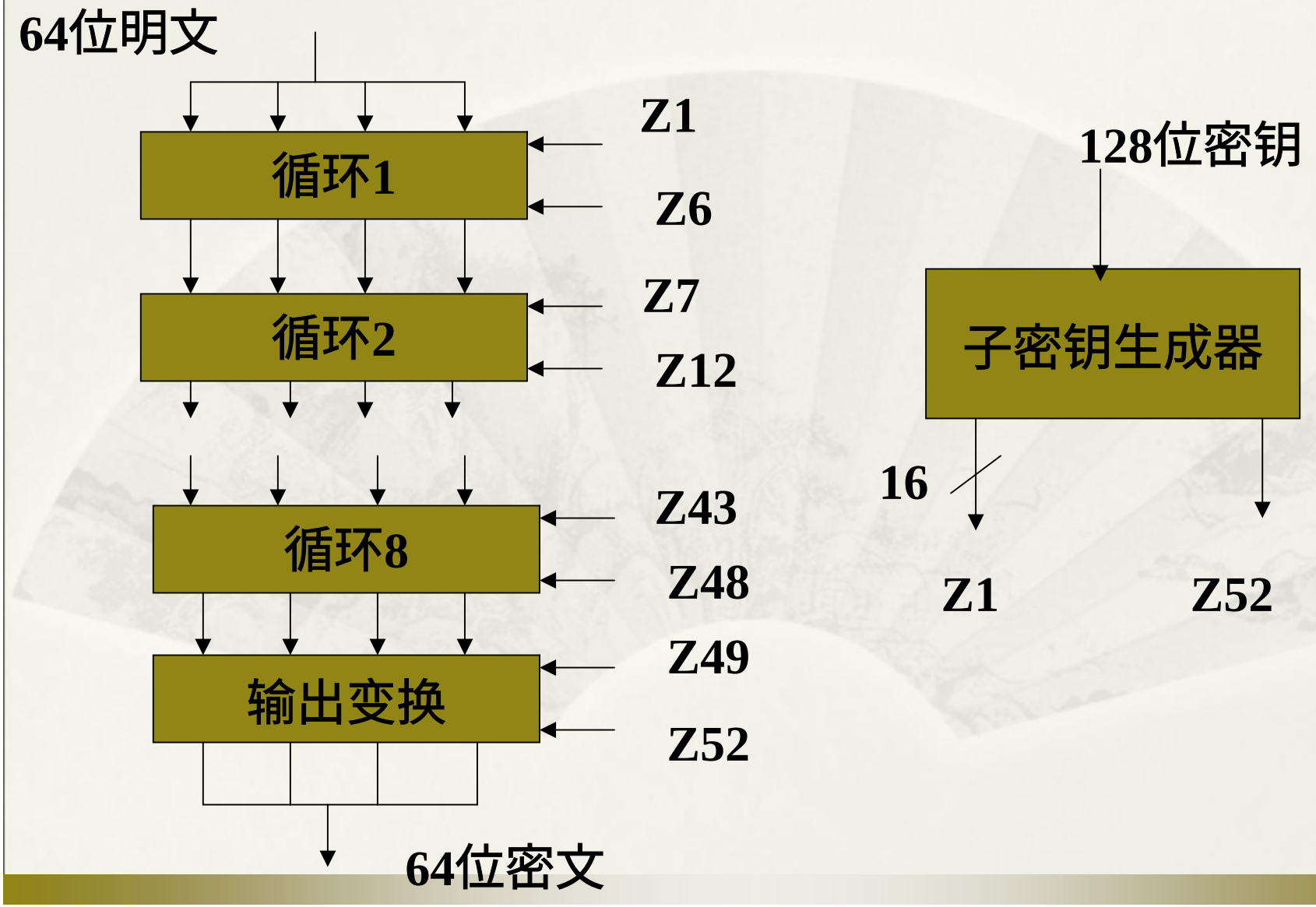
- 异或运算（ ⊕ ）
- 整数模2^{16}加（ ⊞ ）
- 整数模$2^{16}+1$乘（ ⊙ ）(IDEA的S盒)
- 扩散由称为MA结构的算法基本构件提供。



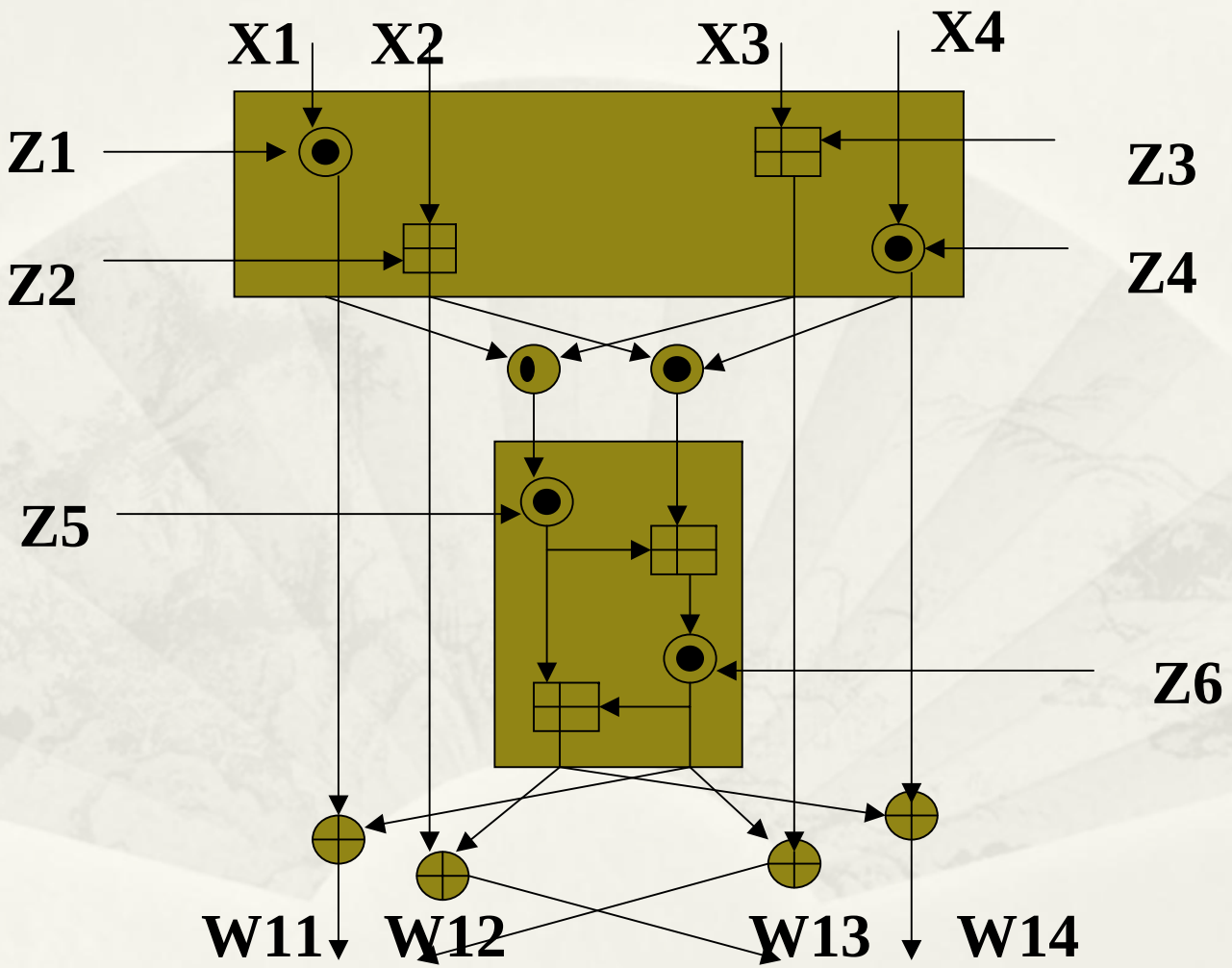
乘法逆元与加法逆元

- 一些概念
  - 因子
  - 素数
  - 互素
    - 最大公因子
- 乘法逆元
  - Gcd(d,f)=1, 则d有模f的乘法逆元
  - $d^{-1}d = 1 \bmod f$
- 加法逆元 $d' + d = 0 \bmod f$

ＩＤＥＡ加密的总体方案



ＩＤＥＡ加密的单个循环



ＩＤＥＡ密钥的产生

56个16bit的子密钥从128bit的密钥中生成

- 前8个子密钥直接从密钥中取出；
- 对密钥进行25bit的循环左移，接下来的密钥就从中取出；
- 重复进行直到52个子密钥都产生出来。

ＩＤＥＡ解密

- 加密解密实质相同，但使用不同的密钥；
- 解密密钥以如下方法从加密子密钥中导出：
  - 解密循环i的头4个子密钥从加密循环10－i的头4个子密钥中导出；解密密钥第1、4个子密钥对应于1、4加密子密钥的乘法逆元；2、3对应2、3的加法逆元；
  - 对前8个循环来说，循环i的最后两个子密钥等于加密循环9－i的最后两个子密钥；

ＩＤＥＡ

- 实现上的考虑
  - 使用子分组：16bit的子分组；
  - 使用简单操作（易于加法、移位等操作实现）
  - 加密解密过程类似；
  - 规则的结构（便于VLSI实现）。

ＩＤＥＡ

- IDEA是PGP的一部分；
- IDEA能抗差分分析和相关分析；
- IDEA似乎没有DES意义下的弱密钥；
- Bruce Schneier 认为IDEA是DES的最好替代

END