SM4 分组密码算法是一个迭代分钟密码算法，由加解密算法和密钥扩展算法组成，SM4 分组密码算法采用非平衡 Feistel 结构，明文分组长度为 128bit，密钥长度为 128bit。加密算法与密钥扩展算法都采用 32 轮非线性迭代结构。解密算法与加密算法的结构相同，只是轮密钥的使用顺序相反，解密轮密钥是加密轮密钥的逆序。

加密首先要生成一套加密密钥，从用户处获得一个 128bit 的初始密钥并将其分为 4 组，之后对应的与系统参数 FK 进行异或运算，之后将产生的结果进行 32 轮迭代（包含：与固定参数 CKi 的异或运算、S 盒替换、循环移位、移位之后的异或运算）最后生成 32 个轮子密钥，每个轮子密钥 32bit，分别供每一轮运算中使用。

SM4 算法的 S 盒替换与 AES 算法中的 S 盒替换类似：输入的前 4 位为行号，后 4 位为列号，行列交叉点处的数值即为替换结果。

SM4 算法的加密流程为首先从用户处获得 128bit 的明文，之后将明文分为 4 组，每组 32bit，之后将其进过轮函数 F 变换，一共进行 32 轮次，最后再经过反序变换之后的到加密后的结果。

加密算法

SM4 加密算法由 32 次迭代运算和 1 次反序变换 R 组成

设明文输入为(X0,X1,X2,X3)∈(Z232)4，密文输出为(Y0,Y1,Y2,Y3)∈(Z232)4,轮密钥为 rki∈Z232，i=0,1,・・・，31。加密算法的运算过程如下。

（1）首先执行 32 次迭代运算：

Xi+4=F(Xi,Xi+1,Xi+2,Xi+3,rki)=Xi XOR T(Xi XOR Xi+1 XOR Xi+2 XOR Xi+3 XOR rki),i=0,1,・・・31

（2）对最后一轮数据进行反序变换并得到密文输出：

(Y0,Y1,Y2,Y3)=R(X32,X33,X34,X35)=(X35,X34,X33,X32)。

其中，T：Z232→Z232 一个可逆变换，由非线性变换 τ 和线性变换 L 复合而成，即 T(・)=L(τ(・))。

非线性变换 τ 由 4 个并行的 S 盒构成。设输入为 A=(a0,a1,a2,a3)∈(Z28)4，非线性变换 τ 的输出为 B=(b0,b1,b2,b3)∈(Z28)4，即：

(b0,b1,b2,b3)= τ (A)=(Sbox(a0),Sbox(a1),Sbox(a2),Sbox(a3))。

设 S 盒的输入为 EF，则经 S 盒运算的输出结果结果为第 E 行、第 F 列的值，即 Sbox(EF)=0x84。

L 是线性变换，非线性变换 τ 的输出是线性变换 L 的输入。设输入为 B∈Z232，则：

C=L(B)=B XOR (B<<<2) XOR (B<<<10) XOR (B<<<18) XOR (B<<<24)。

解密算法：

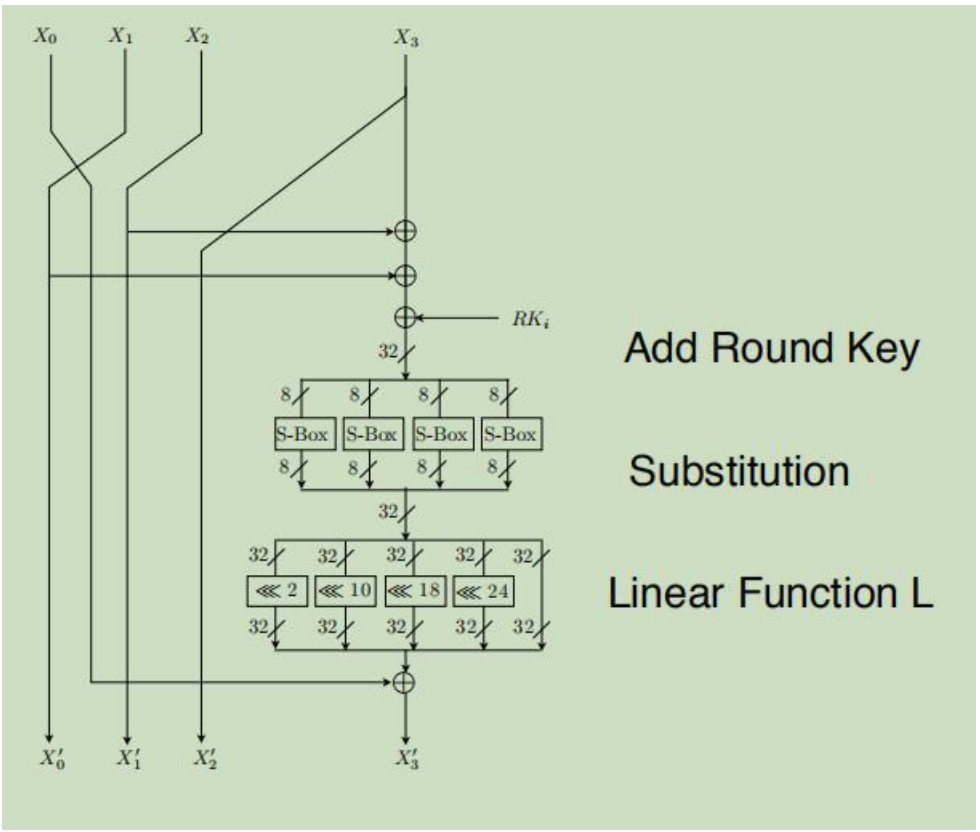
本算法的解密变换与加密变换结构相同，不同的仅是轮密钥的使用顺序，解密时使用轮密钥序(rk31,rk30,・・・,rk0)。

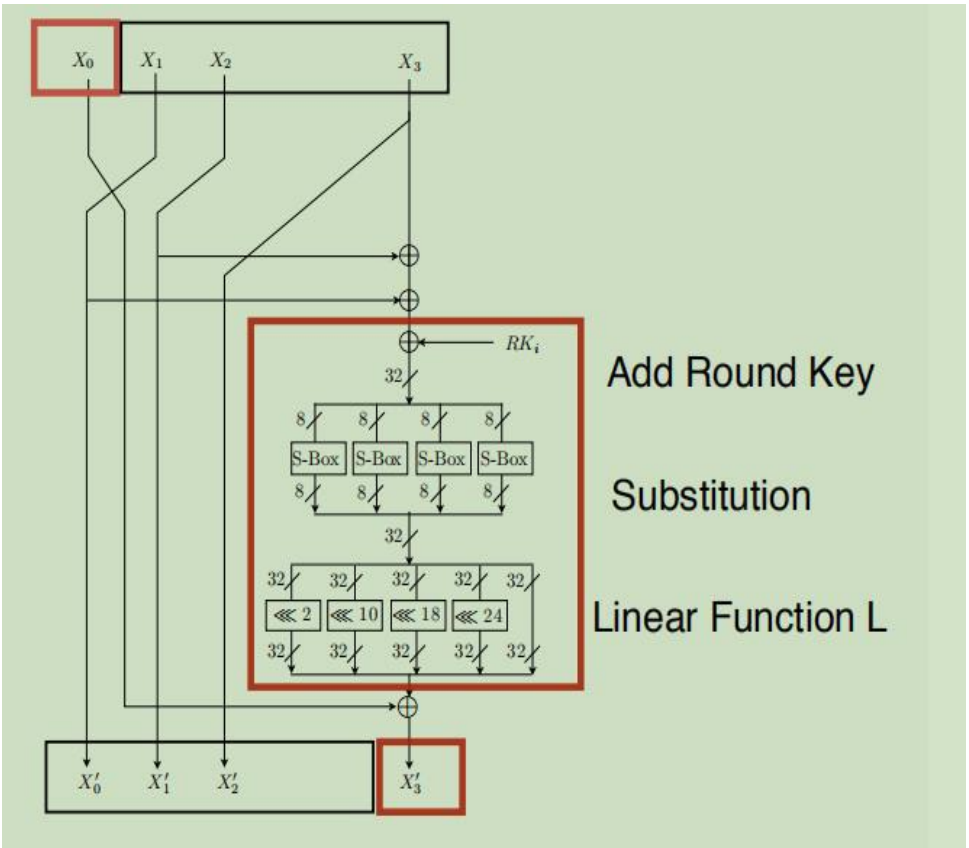
密钥扩展算法：

本算法轮密钥由加密密钥通过密钥扩展算法生成。设加密密钥为 MK，MK=(MK0,MK1,MK2,MK3)∈(Z232)4。

证明：
在已知(Y0,Y1,Y2,Y3)即(X35,X34,X33,X32)且 T 可逆的条件下，可以得到：
Y0⊕T(Y1⊕Y2⊕Y3⊕rk31) = X35⊕T(X34⊕X33⊕X32⊕rk31) = X31
由此式可递推得到 X30，X29.....,X1,X0，最后得到分组(X3,X2,X1,X0)
做反序变换，得到分组(X0,X1,X2,X3)。

流程图：


X_0 X_1 X_2 X_3
RK_i
32
8 8 8 8
S-Box S-Box S-Box S-Box
8 8 8 8
32
32 32 32 32 32
≪ 2 ≪ 10 ≪ 18 ≪ 24
32 32 32 32 32
X'_0 X'_1 X'_2 X'_3
Add Round Key
Substitution
Linear Function L


X_0 X_1 X_2 X_3
RK_i
32
8 8 8 8
S-Box S-Box S-Box S-Box
8 8 8 8
32
32 32 32 32 32
≪ 2 ≪ 10 ≪ 18 ≪ 24
32 32 32 32 32
X'_0 X'_1 X'_2 X'_3
Add Round Key
Substitution
Linear Function L

给北大 logo 加密

安装
Gmssl 和 imagemagick

把图片格式转换为 RGBA；
然后：
Cbc：
```
gmssl enc -sms4-ecb -e -in beida.rgba -out beida-cbc.rgba
gmssl sms4-ecb -d -in beida-cbc.rgba -out beida-cbc-dec.rgba
```
Ecb
```
gmssl enc -sms4-ecb -e -in beida.rgba -out beida-ecb.rgba
gmssl sms4-ecb -d -in beida-ecb.rgba -out beida-ecb-dec.rgba
```

然后转换为 JPG；