**信息科学与工程学院**

**2021－2022学年第一学期**

实 验 报 告

课程名称： 信息安全

实验名称： 实验四

专 业 班 级 通信工程 二班

学 生 学 号 201800121050

学 生 姓 名 孟麟芝

实 验 时 间 2021年10月22日

实验报告

### 【实验目的】

掌握DES加密的原理。

### 【实验要求】

编写64bit ECB DES加密程序。

### 【源代码】

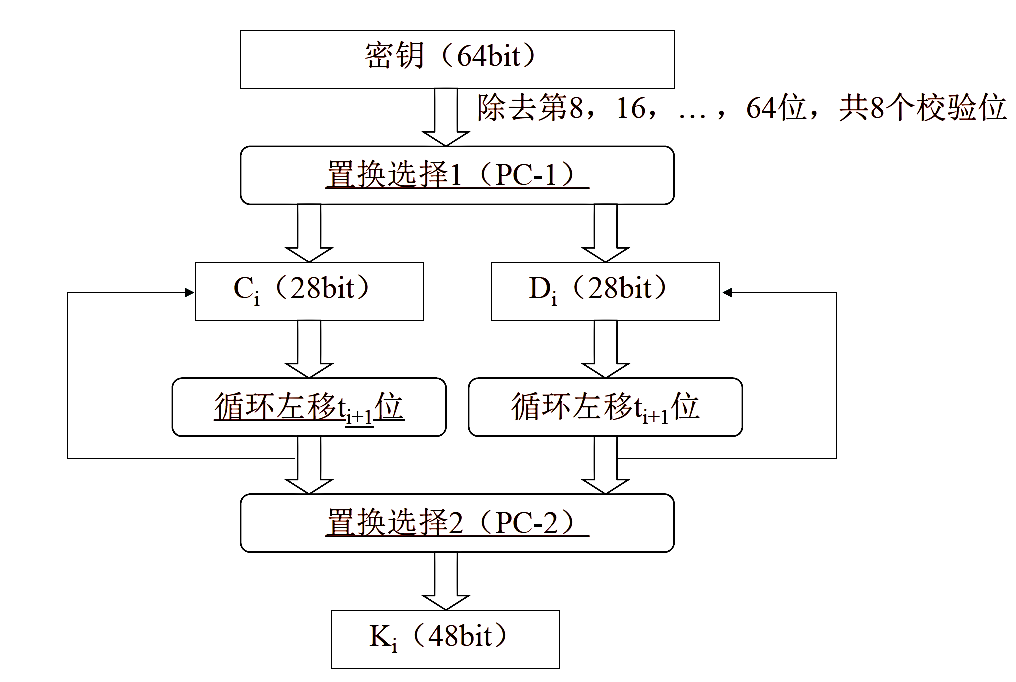
|  |  |
| --- | --- |
|  | function C=DES(M,K) |
|  | %  --------------------------------------------- |
|  | %   数据加密标准DES算法的Matlab实现 |
|  | %   M:输入明文，16位16进制数，字符串格式 |
|  | %   K:输入密钥，16位16进制数，字符串格式 |
|  | %  --------------------------------------------- |
|  |  |
|  | %  ------------第一步：数据初始化------------ |
|  | %将输入的明文和密钥转换成01比特 |
|  | MB=[]; |
|  | for i=1:16 |
|  | MBi=['0000',dec2bin(hex2dec(M(i)))]; |
|  | MBi=MBi(end-3:end); |
|  | MBi=[str2num(MBi(1)),str2num(MBi(2)),str2num(MBi(3)),str2num(MBi(4))]; |
|  | MB=[MB,MBi]; |
|  | end |
|  | M=MB; |
|  | KB=[]; |
|  | for i=1:16 |
|  | KBi=['0000',dec2bin(hex2dec(K(i)))]; |
|  | KBi=KBi(end-3:end); |
|  | KBi=[str2num(KBi(1)),str2num(KBi(2)),str2num(KBi(3)),str2num(KBi(4))]; |
|  | KB=[KB,KBi]; |
|  | end |
|  | K=KB; |
|  | %各个要用到的矩阵 |
|  | E=[32, 1, 2, 3, 4, 5; |
|  | 4, 5, 6, 7, 8, 9; |
|  | 8, 9,10,11,12,13; |
|  | 12,13,14,15,16,17; |
|  | 16,17,18,19,20,21; |
|  | 20,21,22,23,24,25; |
|  | 24,25,26,27,28,29; |
|  | 28,29,30,31,32,1]; |
|  | S1=[14, 4, 13, 1, 2, 15, 11, 8, 3, 10, 6, 12, 5, 9, 0, 7; |
|  | 0, 15, 7, 4, 14, 2, 13, 1, 10, 6, 12, 11, 9, 5, 3, 8; |
|  | 4, 1, 14, 8, 13, 6, 2, 11, 15, 12, 9, 7, 3, 10, 5, 0; |
|  | 15, 12, 8, 2, 4, 9, 1, 7, 5, 11, 3, 14, 10, 0, 6, 13]; |
|  | S2=[15, 1, 8, 14, 6, 11, 3, 4, 9, 7, 2, 13, 12, 0, 5, 10; |
|  | 3, 13, 4, 7, 15, 2, 8, 14, 12, 0, 1, 10, 6, 9, 11, 5; |
|  | 0, 14, 7, 11, 10, 4, 13, 1, 5, 8, 12, 6, 9, 3, 2, 15; |
|  | 13, 8, 10, 1, 3, 15, 4, 2, 11, 6, 7, 12, 0, 5, 14, 9]; |
|  | S3=[10, 0, 9, 14, 6, 3, 15, 5, 1, 13, 12, 7, 11, 4, 2, 8; |
|  | 13, 7, 0, 9, 3, 4, 6, 10, 2, 8, 5, 14, 12, 11, 15, 1; |
|  | 13, 6, 4, 9, 8, 15, 3, 0, 11, 1, 2, 12, 5, 10, 14, 7; |
|  | 1, 10, 13, 0, 6, 9, 8, 7, 4, 15, 14, 3, 11, 5, 2, 12]; |
|  | S4=[7, 13, 14, 3, 0, 6, 9, 10, 1, 2, 8, 5, 11, 12, 4, 15; |
|  | 13, 8, 11, 5, 6, 15, 0, 3, 4, 7, 2, 12, 1, 10, 14, 9; |
|  | 10, 6, 9, 0, 12, 11, 7, 13, 15, 1, 3, 14, 5, 2, 8, 4; |
|  | 3, 15, 0, 6, 10, 1, 13, 8, 9, 4, 5, 11, 12, 7, 2, 14]; |
|  | S5=[2, 12, 4, 1, 7, 10, 11, 6, 8, 5, 3, 15, 13, 0, 14, 9; |
|  | 14, 11, 2, 12, 4, 7, 13, 1, 5, 0, 15, 10, 3, 9, 8, 6; |
|  | 4, 2, 1, 11, 10, 13, 7, 8, 15, 9, 12, 5, 6, 3, 0, 14; |
|  | 11, 8, 12, 7, 1, 14, 2, 13, 6, 15, 0, 9, 10, 4, 5, 3]; |
|  | S6=[12, 1, 10, 15, 9, 2, 6, 8, 0, 13, 3, 4, 14, 7, 5, 11; |
|  | 10, 15, 4, 2, 7, 12, 9, 5, 6, 1, 13, 14, 0, 11, 3, 8; |
|  | 9, 14, 15, 5, 2, 8, 12, 3, 7, 0, 4, 10, 1, 13, 11, 6; |
|  | 4, 3, 2, 12, 9, 5, 15, 10, 11, 14, 1, 7, 6, 0, 8, 13]; |
|  | S7=[4, 11, 2, 14, 15, 0, 8, 13, 3, 12, 9, 7, 5, 10, 6, 1; |
|  | 13, 0, 11, 7, 4, 9, 1, 10, 14, 3, 5, 12, 2, 15, 8, 6; |
|  | 1, 4, 11, 13, 12, 3, 7, 14, 10, 15, 6, 8, 0, 5, 9, 2; |
|  | 6, 11, 13, 8, 1, 4, 10, 7, 9, 5, 0, 15, 14, 2, 3, 12]; |
|  | S8=[13, 2, 8, 4, 6, 15, 11, 1, 10, 9, 3, 14, 5, 0, 12, 7; |
|  | 1, 15, 13, 8, 10, 3, 7, 4, 12, 5, 6, 11, 0, 14, 9, 2; |
|  | 7, 11, 4, 1, 9, 12, 14, 2, 0, 6, 10, 13, 15, 3, 5, 8; |
|  | 2, 1, 14, 7, 4, 10, 8, 13, 15, 12, 9, 0, 3, 5, 6, 11]; |
|  | PC1=[57,49,41,33,25,17,9,1,58,50,42,34,26,18,10,2,59,51,43,35,27,19,11,3,6  0,52,44,36,63,55,47,39,31,23,15,7,62,54,46,38,30,22,14,6,61,53,45,37,29,21,13,5,28,20,12,4]; |
|  | PC2=[14,17,11,24,1,5,3,28,15,6,21,10,23,19,12,4,26,8,16,7,27,20,13,2,41,52  ,31,37,47,55,30,40,51,45,33,48,44,49,39,56,34,53,46,42,50,36,29,32]; |
|  | IP=[58,50,42,34,26,18,10,2,60,52,44,36,28,20,12,4,62,54,46,38,30,22,14,6,6  4,56,48,40,32,24,16,8,57,49,41,33,25,17,9,1,59,51,43,35,27,19,11,3,61,53,45,37,29,21,13,5,63,55,47,39,31,23,15,7]; |
|  | IP\_1=[40,8,48,16,56,24,64,32,39,7,47,15,55,23,63,31,38,6,46,14,54,22,62,30  ,37,5,45,13,53,21,61,29,36,4,44,12,52,20,60,28,35,3,43,11,51,19,59,27,34,2,42,10,50,18,58,26,33,1,41,9,49,17,57,25]; |
|  | P=[16,7,20,21,29,12,28,17,1,15,23,26,5,18,31,10,2,8,24,14,32,27,3,9,19,13,  30,6,22,11,4,25]; |
|  | %  ------------第二步：产生子密钥------------ |
|  | Ki=zeros(16,48); |
|  | K\_PC1=K(PC1);%置换选择1 |
|  | C0=K\_PC1(1:28); |
|  | D0=K\_PC1(29:56); |
|  | for i=1:16 |
|  | %第1、2、9、16轮循环左移1位 |
|  | if i==1||i==2||i==9||i==16 |
|  | C0=[C0(2:end),C0(1)]; |
|  | D0=[D0(2:end),D0(1)]; |
|  | else |
|  | %其它轮循环左移2位 |
|  | C0=[C0(3:end),C0(1:2)]; |
|  | D0=[D0(3:end),D0(1:2)]; |
|  | end |
|  | K\_LS=[C0,D0]; |
|  | Ki(i,:)=K\_LS(PC2);%置换选择2 |
|  | end |
|  | %  ------------第三步：DES加密------------ |
|  | M=M(IP);%明文初始置换 |
|  | L=M(1:32);%输入的左半部分明文 |
|  | R=M(33:64);%输入的右半部分明文 |
|  | %以下是16轮迭代 |
|  | for i=1:16 |
|  | E0=reshape(E',1,48);%把E盒转换成行向量 |
|  | R\_E=R(E0);%E盒扩展 |
|  | R\_Ki=xor(R\_E,Ki(i,:));%与48bit子密钥模二加 |
|  | %8个S盒构成选择压缩运算S |
|  | %S1 |
|  | B=R\_Ki(1:6); |
|  | x=B(1)\*2+B(6)+1; |
|  | y=B(2)\*8+B(3)\*4+B(4)\*2+B(5)+1; |
|  | C=['0000',dec2bin(S1(x,y))]; |
|  | C=C(end-3:end); |
|  | C1=[str2num(C(1)),str2num(C(2)),str2num(C(3)),str2num(C(4))]; |
|  | %S2 |
|  | B=R\_Ki(7:12); |
|  | x=B(1)\*2+B(6)+1; |
|  | y=B(2)\*8+B(3)\*4+B(4)\*2+B(5)+1; |
|  | C=['0000',dec2bin(S2(x,y))]; |
|  | C=C(end-3:end); |
|  | C2=[str2num(C(1)),str2num(C(2)),str2num(C(3)),str2num(C(4))]; |
|  | %S3 |
|  | B=R\_Ki(13:18); |
|  | x=B(1)\*2+B(6)+1; |
|  | y=B(2)\*8+B(3)\*4+B(4)\*2+B(5)+1; |
|  | C=['0000',dec2bin(S3(x,y))]; |
|  | C=C(end-3:end); |
|  | C3=[str2num(C(1)),str2num(C(2)),str2num(C(3)),str2num(C(4))]; |
|  | %S4 |
|  | B=R\_Ki(19:24); |
|  | x=B(1)\*2+B(6)+1; |
|  | y=B(2)\*8+B(3)\*4+B(4)\*2+B(5)+1; |
|  | C=['0000',dec2bin(S4(x,y))]; |
|  | C=C(end-3:end); |
|  | C4=[str2num(C(1)),str2num(C(2)),str2num(C(3)),str2num(C(4))]; |
|  | %S5 |
|  | B=R\_Ki(25:30); |
|  | x=B(1)\*2+B(6)+1; |
|  | y=B(2)\*8+B(3)\*4+B(4)\*2+B(5)+1; |
|  | C=['0000',dec2bin(S5(x,y))]; |
|  | C=C(end-3:end); |
|  | C5=[str2num(C(1)),str2num(C(2)),str2num(C(3)),str2num(C(4))]; |
|  | %S6 |
|  | B=R\_Ki(31:36); |
|  | x=B(1)\*2+B(6)+1; |
|  | y=B(2)\*8+B(3)\*4+B(4)\*2+B(5)+1; |
|  | C=['0000',dec2bin(S6(x,y))]; |
|  | C=C(end-3:end); |
|  | C6=[str2num(C(1)),str2num(C(2)),str2num(C(3)),str2num(C(4))]; |
|  | %S7 |
|  | B=R\_Ki(37:42); |
|  | x=B(1)\*2+B(6)+1; |
|  | y=B(2)\*8+B(3)\*4+B(4)\*2+B(5)+1; |
|  | C=['0000',dec2bin(S7(x,y))]; |
|  | C=C(end-3:end); |
|  | C7=[str2num(C(1)),str2num(C(2)),str2num(C(3)),str2num(C(4))]; |
|  | %S8 |
|  | B=R\_Ki(43:48); |
|  | x=B(1)\*2+B(6)+1; |
|  | y=B(2)\*8+B(3)\*4+B(4)\*2+B(5)+1; |
|  | C=['0000',dec2bin(S8(x,y))]; |
|  | C=C(end-3:end); |
|  | C8=[str2num(C(1)),str2num(C(2)),str2num(C(3)),str2num(C(4))]; |
|  | C=[C1,C2,C3,C4,C5,C6,C7,C8];%8个S盒的结果合并起来 |
|  | R\_P=C(P);%经过P盒置换 |
|  | %交换左右两半部分 |
|  | TEMP=L; |
|  | L=R; |
|  | R=xor(TEMP,R\_P); |
|  | end |
|  | %  ------------第四步：输出密文------------ |
|  | %逆初始置换 |
|  | C=[R,L]; |
|  | C=C(IP\_1); |
|  | CS=[]; |
|  | C=num2str(C); |
|  | pos=(C~=' '); |
|  | C=C(pos); |
|  | for i=1:4:61 |
|  | Ci=C(i:i+3); |
|  | CS=[CS,num2str(dec2hex(bin2dec(Ci)))]; |
|  | end |
|  | C=CS; |

### 【实验过程】

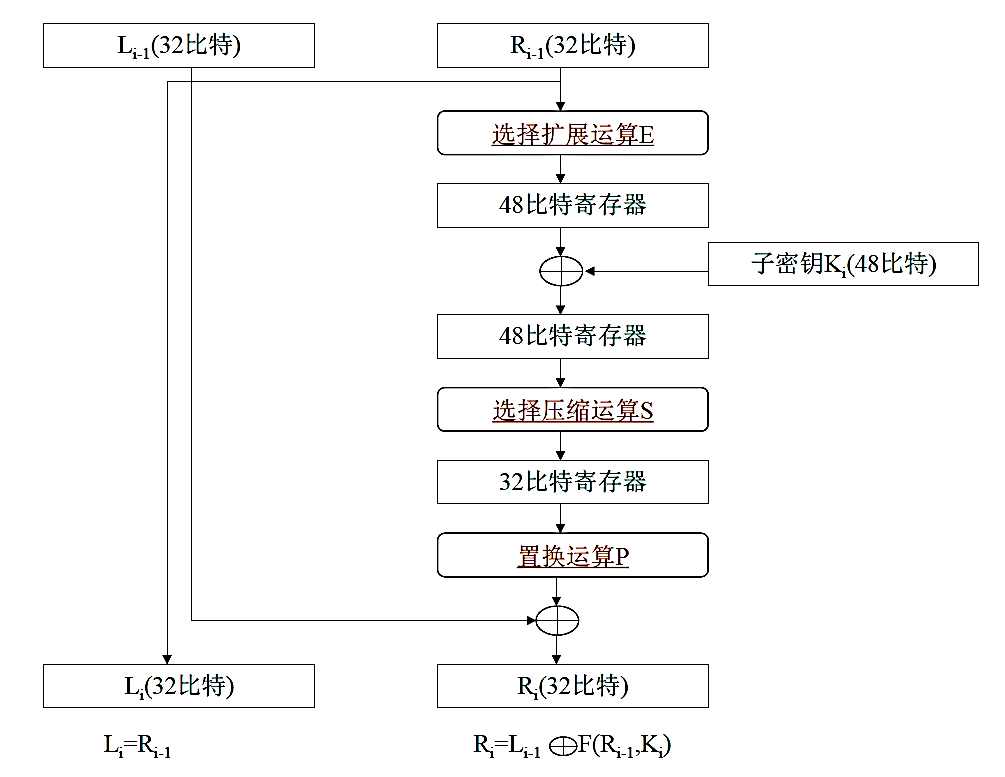
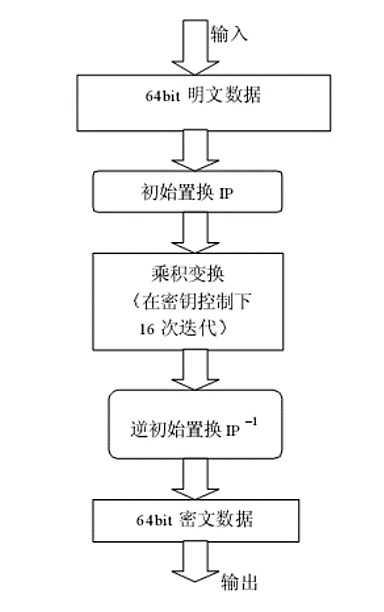
代码本身较为冗长，在这里稍作介绍。

程序中的“第一步”是数据的初始化，这一步只是为了初始化一些变量，方便后面的操作。

“第二步”为产生子密钥，其原理如下框图，其中PC-1、PC-2在代码中分别对应PC1和PC2。



“第三步”则是正式的DES加密过程，DES算法的整体流程如下左图，其中乘积变换的具体流程如下右图，要注意的是在实际代码中“寄存器”的概念并没有体现。



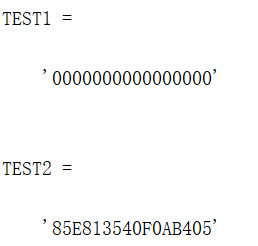
另外要注意的是逆初始置换是用如下公式计算的：

最后，我们对代码进行测试：

TEST1=DES('8787878787878787','0E329232EA6D0D73');

TEST2=DES('0123456789ABCDEF','133457799BBCDFF1');

上面的代码得到如下输出：



下用密码学工具cryptool进行验证，可见结果是正确的：

