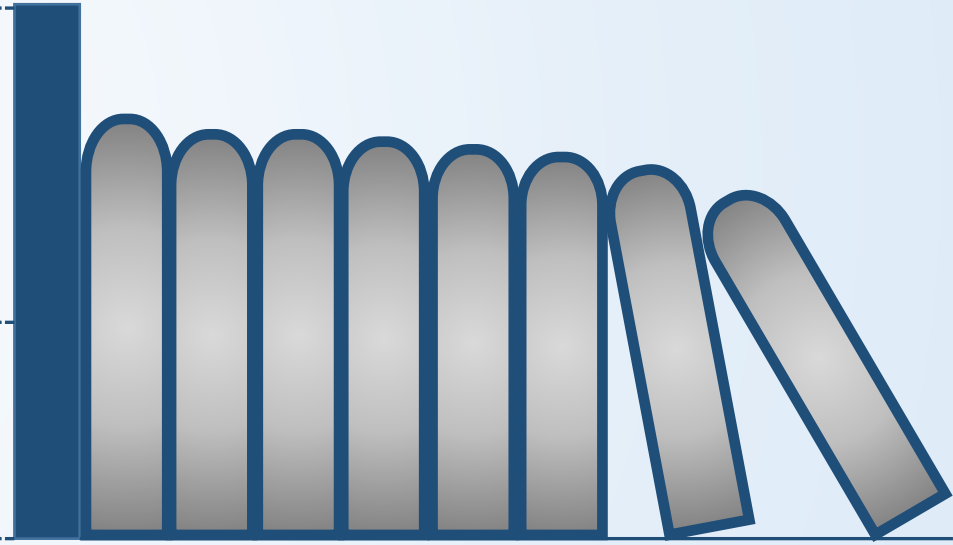
哈爾濱工業大學(深圳)

HARBIN INSTITUTE OF TECHNOLOGY, SHENZHEN

# DES密码算法实验


主讲教师：蒋琳

实验教师：苏婷


# 实验目的

- 理解对称密码体制的基本思想
- 掌握 DES 算法的基本工作原理，理解混淆和扩散的概念
- 编程实现DES密码加解密算法

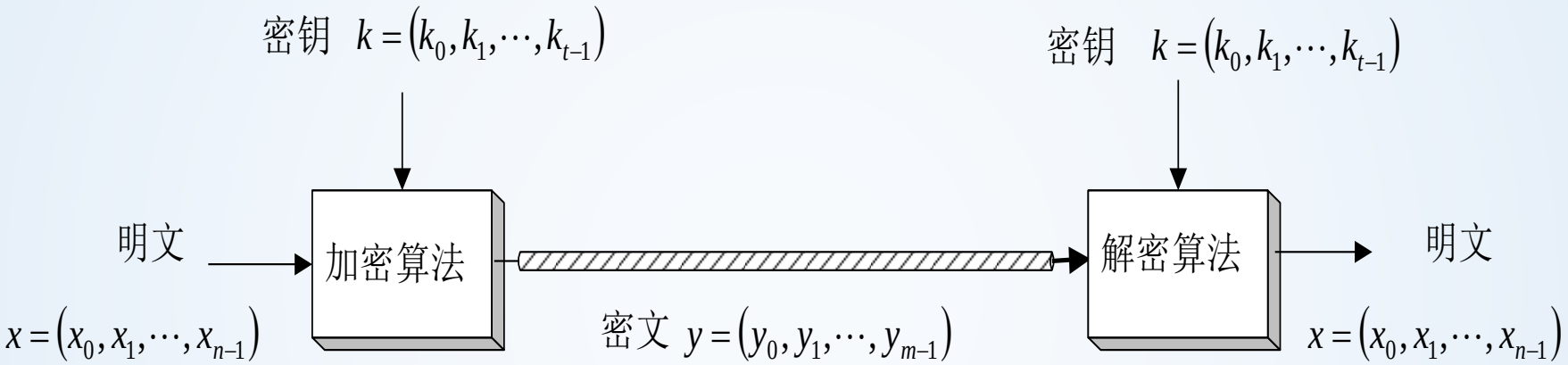
# 实验环境

- **开发环境：** CodeBlocks， Visual C++6.0 等
- **密码算法实验系统**

http://10.251.129.2/index/check.html


# 实验原理

## ➢ 分组密码算法




- 将明文划分为n比特长的块（DES为64位），每一块进行加密算法
- 分组密码的安全性主要依赖于密钥k
- 属于对称加密算法

# 实验原理

- **初始置换**

$$L_0R_0 = IP(X)$$

- **轮函数(Feistel结构)**

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$$

- **逆初始置换**

$$Y = IP^{-1}(R_{16}L_{16})$$

- **DES的加密算法具有可逆性，解密与加密算法相同，所不同的是子密钥顺序使用相反，依次为：**

$$k_{16}, k_{15}, \cdots, k_1$$

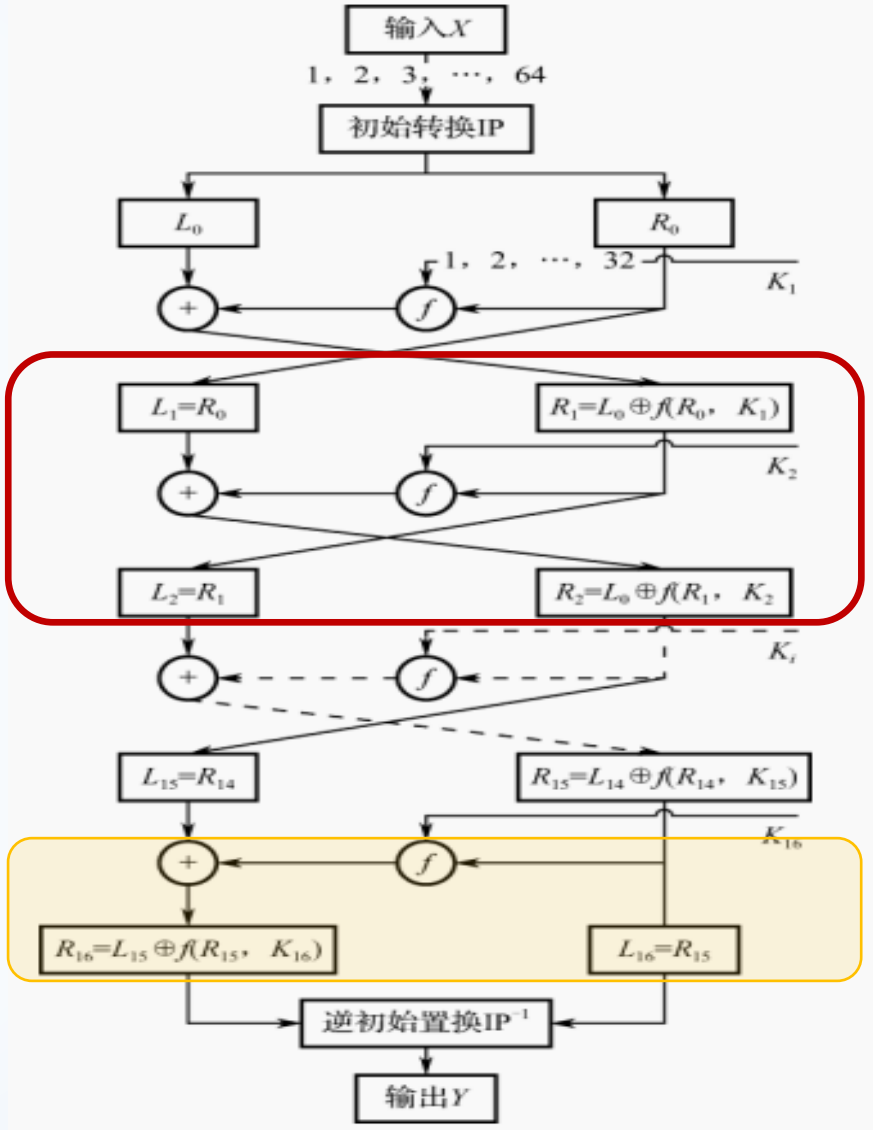



图 3.5　DES 加密算法

# 实验原理

## ➢ 初始置换

举例说明：明文X为16进制的字符串012345678ABCDEF，进行初始置换后分成左右两部分$L_0R_0$

| 明文 $(0123456789ABCDEF)_{16}$ | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |

表 3.1　初始置换 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 |
| 64 | 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 |

| | $X_0$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $L_0$ | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| $R_0$ | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |

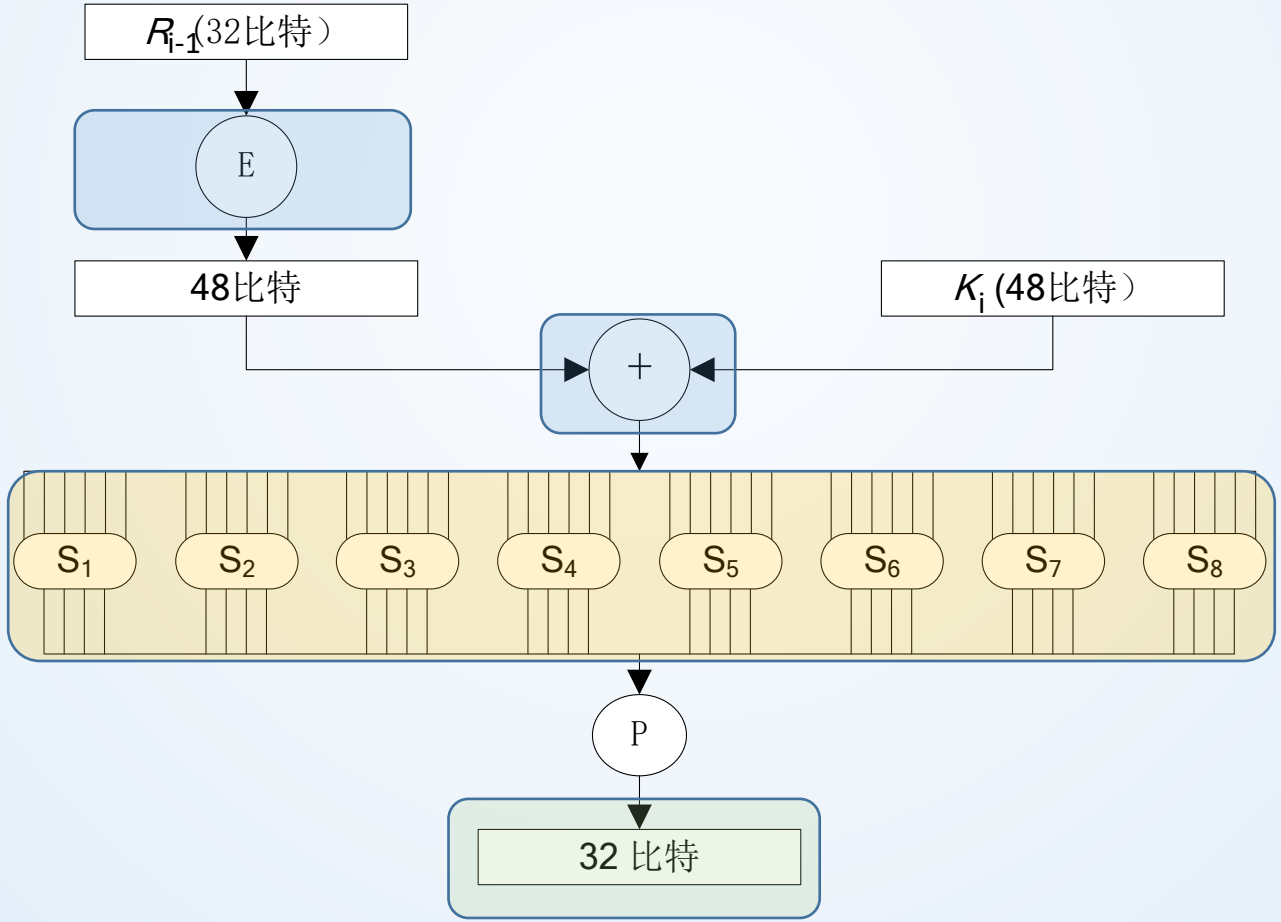
# 实验原理

## ➢ 轮函数

$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$$

### ◆ 扩展置换E， S盒代换， P盒置换



# 实验原理

◆ **扩展置换E， S盒代换， P盒置换**

| $R_0$ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |

32->48

扩展置换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 |

$E(R_0)$

011110 100001 010101 010101 011110 100001 010101 010101

⊕

$K_1$

000011 100011 001001 101100 011011 010010 011100 011101

$E(R_0)⊕K_1$

011101 000010 011100 111001 000101 110011 001001 001000

# 实验原理

◆ 扩展置换E， S盒代换， P盒置换

| $E(R_0)⊕K_1$ |
| --- |
| 011101 000010 011100 111001 000101 110011 001001 001000 |

011101其中中间四位1110为列，前后两位01为行

000010其中中间四位0001为列，前后两位00为行

| 行\列 | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S1 | 0 | 14 | 4 | 13 | 1 | 2 | 15 | 11 | 8 | 3 | 10 | 6 | 12 | 5 | 9 | 0 | 7 |
| | 1 | 0 | 15 | 7 | 4 | 14 | 2 | 13 | 1 | 10 | 6 | 12 | 11 | 9 | 5 | 3 | 8 |
| | 2 | 4 | 1 | 14 | 8 | 13 | 6 | 2 | 11 | 15 | 12 | 9 | 7 | 3 | 10 | 5 | 0 |
| | 3 | 15 | 12 | 8 | 2 | 4 | 9 | 1 | 7 | 5 | 11 | 3 | 14 | 10 | 0 | 6 | 13 |
| S2 | 0 | 15 | 1 | 8 | 14 | 6 | 11 | 3 | 4 | 9 | 7 | 2 | 13 | 12 | 0 | 5 | 10 |
| | 1 | 3 | 13 | 4 | 7 | 15 | 2 | 8 | 15 | 12 | 0 | 1 | 10 | 6 | 9 | 11 | 5 |
| | 2 | 0 | 14 | 7 | 11 | 10 | 4 | 13 | 1 | 5 | 8 | 12 | 6 | 9 | 3 | 2 | 15 |
| | 3 | 13 | 8 | 10 | 1 | 3 | 15 | 4 | 2 | 11 | 6 | 7 | 12 | 0 | 5 | 14 | 9 |

$S_1(011101) = 0011$

$S_2(000010) = 0001$

# 实验原理

◆ 扩展置换E， S盒代换， P盒置换

$S(E(R_0)\oplus K_1)$

0011 0001 0010 1100 0010 1110 0100 0110

| | | | |
|---|---|---|---|
| 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 |

P盒置换

$f(R_{i-1}, K_i)$= $P(S(E(R_0)\oplus K_1))$

$P(S(E(R_0)\oplus K_1))$

00010000 00110010 01010010 11101110

# 实验原理

## ➢ 轮函数

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$$

### ◆ 扩展置换E， S盒代换， P盒置换

$L_1 = R_0$ =11110000 10101010 11110000 10101010

$R_1 = L_0 \oplus f(R_0, K_1)$
= 11011100 00110010 10011110 00010001

$L_0$ = 11001100 00000000 11001100 11111111
$R_0$ = 1 1110000 10101010 11110000 10101010

f($R_0$,$K_1$)
00010000 00110010 01010010 11101110

依次类推，可以得出其它各轮运算结果。经过16轮迭代，最后一轮（第16轮）迭代输出结果的左右两部分为：

$R_{16}$=01110010 00010011 01001111 10010011

$L_{16}$=10001111 01011110 00000011 10111100

# 实验原理

## ➢ 逆初始置换

$$C = IP^{-1}(R_{16}L_{16})$$

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $R_{16}$ | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |
| | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| $L_{16}$ | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
| | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 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 |

| C | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

# 实验原理

## ➢ 生成子密钥Ki

### 置换PC_1，循环左移，置换PC_2

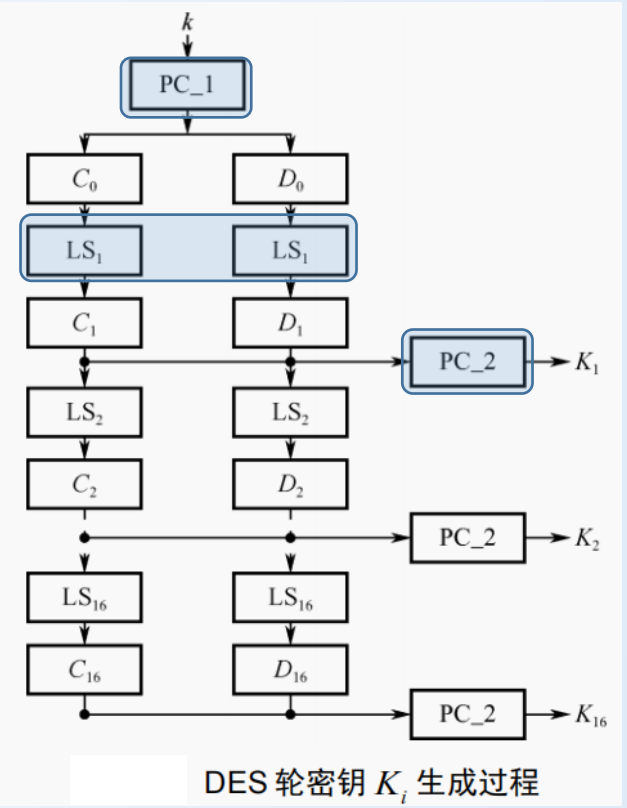
假设初始密钥为 $K = (123DAB779F658067)_{16}$

| K $(123DAB779F658067)_{16}$ | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 |

64->56

| PC-1 | | | | | | |
|---|---|---|---|---|---|---|
| 57 | 49 | 41 | 33 | 25 | 17 | 9 |
| 1 | 58 | 50 | 42 | 34 | 26 | 18 |
| 10 | 2 | 59 | 51 | 43 | 35 | 27 |
| 19 | 11 | 3 | 60 | 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 |

| $C_0$ | $D_0$ |
|---|---|
| 0101010 0101010 0010101 1100001 | 1001110 1101110 1000010 1101011 |



DES 轮密钥 $K_i$ 生成过程

说明：64位的初始密钥，其中有8位校验位，经过PC_1的置换变为56位。

其中8，16，24，32，56，64位没有替换，直接作为校验位省去。

# 实验原理

◆ **生成子密钥Ki：置换PC_1，循环左移，置换PC_2**

根据轮数的不同，循环左移的位数是不同的，其中1，2，9，16这几个轮次循环左移1位，其他轮次循环左移2位。

| $C_0$ | $D_0$ |
|---|---|
| 0101010 0101010 0010101 1100001 | 1001110 1101110 1000010 1101011 |

| 轮数 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 位数 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 1 |

| $C_1$ | $D_1$ |
|---|---|
| 1010100 1010100 0101011 1000010 | 0011101 1011101 0000101 1010111 |

# 实验原理

◆ **生成子密钥Ki：置换PC_1，循环左移，置换PC_2**

置换PC_2将56位的转换位48位，去掉第9、18、22、25、35、38、43、54位，从56位变成48位。

| $C_1$ | $D_1$ |
| --- | --- |
| 1010100 1010100 0101011 1000010 | 0011101 1011101 0000101 1010111 |

PC-2

| | | | | | |
| --- | --- | --- | --- | --- | --- |
| 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 |

56->48

| $K_1$ |
| --- |
| 00001110 00110010 01101100 01101101 00100111 00011101 |

# 实验原理

- **解密由同一算法实现**

**密文C作为输入，密钥逆序使用（$K_{16}...K_1$),输出明文X。**

**例如：加密最后一轮输出（ $R_{16}$, $L_{16}$)**

$$L_{16} = R_{15}$$

$$R_{16} = L_{15} \oplus f(R_{15}, K_{16})$$

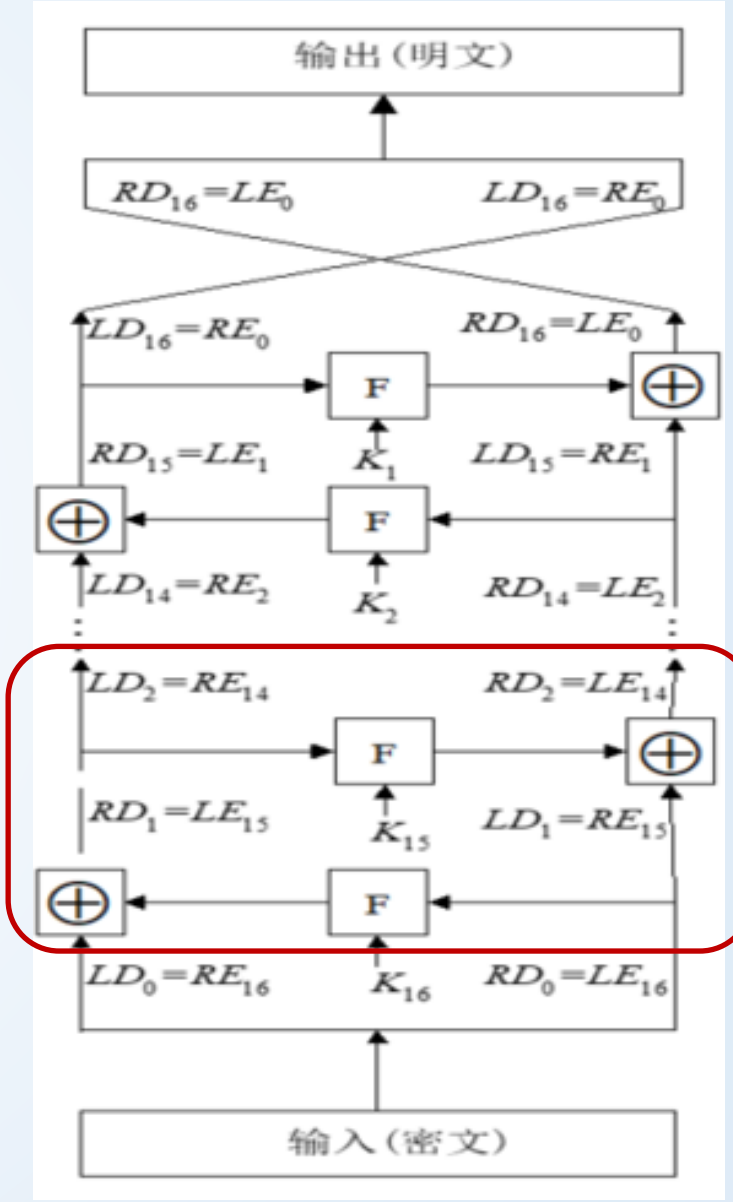
**解密：输入（ $R_{16}$, $L_{16}$)**

$$L'_1 = L_{16} = R_{15}$$

$$R'_1 = R_{16} \oplus f(R_{15}, K_{16}) = L_{15}$$

$$L'_2 = R'_1 = L_{15} = R_{14}$$

$$R'_2 = R_{15} \oplus f(R_{14}, K_{15}) = L_{14}$$




# 实验内容

1、编写程序实现DES算法的加密和解密过程。（请用 demo.c中的IP置换矩阵、PC1、PC2、E盒、S盒、P盒以及$IP^{-1}$）

2、由用户从文件或者终端中读入明文（或密文）和密钥,统一其中一个明文security；

3、过程中打印每一轮加密的结果（二进制或者16进制均可）；

4、选取明文相差1位的情况，观察每轮密文的差异；

# 思考题

1、编写程序实现DES算法的加密和解密过程。（请用 demo.c中的IP置换矩阵、PC1、PC2、E盒、S盒、P盒以及$IP^{-1}$）

2、由用户从文件或者终端中读入明文（或密文）和密钥,统一其中一个明文security；

3、过程中打印每一轮加密的结果（二进制或者16进制均可）；

4、选取明文相差1位的情况，观察每轮密文的差异；

```
本程序是DES密码算法的加解密程序，加密请输入0，解密请输入1：0
请输入明文（长度为8的ASCII字符串）：security
请输入密钥（长度为8的ASCII字符串）：test1234
L[ 1] = 00ffa015, R[ 1] = badcd18f
L[ 2] = badcd18f, R[ 2] = afflce5d
L[ 3] = afflce5d, R[ 3] = 5039766b
L[ 4] = 5039766b, R[ 4] = dc0fb1a4
L[ 5] = dc0fb1a4, R[ 5] = 246f3b15
L[ 6] = 246f3b15, R[ 6] = 17418e02
L[ 7] = 17418e02, R[ 7] = 8934c64d
L[ 8] = 8934c64d, R[ 8] = abf3eb80
L[ 9] = abf3eb80, R[ 9] = d90b4897
L[10] = d90b4897, R[10] = 6e178633
L[11] = 6e178633, R[11] = bf42f3df
L[12] = bf42f3df, R[12] = 90c608c8
L[13] = 90c608c8, R[13] = 35d463be
L[14] = 35d463be, R[14] = 419126b7
L[15] = 419126b7, R[15] = 75b6c041
L[16] = 75b6c041, R[16] = 160f43f5
密文为：9774f110e1a18f29
```


# 实验要求

- 根据密钥（*hitsz018*）生成对应的明文和密文，并输出每一轮加密后的结果。
- 请将结果截图内容及源代码打包成一个压缩包上传到系统中，命名格式如下：

  测试结果：“学号_姓名_实验2_DES”

  压缩包：“学号_姓名_实验2_DES”
- 提交要求：11月10日24点之前 http://10.251.129.2/index/check.html ；

# 谢谢