密码学原理课程作业报告

作业 2：分组密码体制的实践与分析

姓名	*******	院系	软件学院	学号	********
任课教师	刘绍辉		指导教师	刘绍辉	
实验地点			实验时间		

一、作业目的

要求：综述本次实验的基本目的。

1．编程实现 DES 标准算法

2．通过实例理解 DES 的雪崩效应

3. 掌握 S 盒的差分分析方法

二、作业内容

要求：对如下内容进行详细描述。

1． DES 密钥生成算法及其数据结构；

初始 Key 值为 64 位，DES 算法规定，其中第 8、16、.....64 位是奇偶校验位，不参与 DES 运算。故 Key 实际可用位数便只有 56 位。 即：经过缩小选择换位表 1 的变换后，Key 的位数由 64 位变成了 56 位，此 56 位分为 C0、D0 两部分，各 28 位，然后分别进行第 1 次循环左移，得到 C1、D1，将 C1（28 位）、D1（28 位）合并得到 56 位，再经过缩小选择换位 2，从而便得到了密钥 K0（48 位）。依此类推，便可得到 K1、 K2、.....、K15，不过需要注意的是，16 次循环左移对应的左移位数要依据下述规则进行：循环左移位数分别是 1,1,2,2,2,2,2,2,1,2,2,2,2,2,2,1

2． DES 加、解密算法的概要描述及其数据结构设计；

DES 加密和解密用到的数据结构主要是 S 盒以及 P 盒等 Table。

加密和解密的过程相同：首先经过初始置换，然后将 64 位序列分成两部分，各 32 位，再经过乘积变换，乘积变换总共进行 16 轮，每轮只对右边的 32 位进行加密变换，迭代结束之前将左边的 32 位与右边按位异或，作为下一轮右边的值，并将右边未经变换的值作为左边。乘积变换的过程是， 首先进入扩展置换（E 盒），32 位扩展为 48 位，然后进行密钥加运算，再然后进行 S 盒置换，将 48 位序列每 8 个分为一组，分别进入 8 个 S 盒，S 盒的输入为 6 位，输出为 4 位，经过 S 盒置换之后，48 位序列又重新变为 32 位。最后进行 P 置换。经过 16 轮之后，最后将得到的结果进行逆初始置换。

3. 用 DES 加、解密 BMP 灰度图像的算法；

BMP 灰度图像的每个像素实际上一个 8 位二进制数，存储了灰度值。因此可以将灰度图像处理的每 8 个像素作为一组，处理成 64 位，进行 DES 加密。需要注意的是，BMP 文件最开始的若干字节是文件头部，用于指示文件类型和有关信息。为了让加密之后的图像还能够打开，需要在加密时跳过头部。

4. 随机明文生成算法及其数据结构设计；

DES 密钥要求为 64 位，实际使用 56 位，其中第 8，16，24，32，40，48，56，64 位是奇偶校验位，可以简单地使用随机数生成算法生成 56 位二进制数，在进行奇偶校验得到最后的密钥。

5. DES 某个 S-盒的差分分布表计算算法；

S 盒的输入为 6 位，可能的输入差分有 64 个，输出为 4 位，可能的输出差为 16 个，分别计算这 64 个输入差情况下的输出差，对它们进行统计计数，便可得到差分分析表

6. 基于某个线性函数计算的 S-盒生成算法及其数据结构设计；

线性函数即输入和输出之间可以找到一个线性关系，利用线性关系可以很方便的生成一个 S 盒

7. 随机产生的 S-盒生成算法及其数据结构设计；

可以使用简单的随机函数，生成 0-15 之间的随机数，这样便生成了一个随机的 S 盒

8. 自行设计一种 S-盒生成算法及其数据结构设计(选做)。

三、作业结果及分析

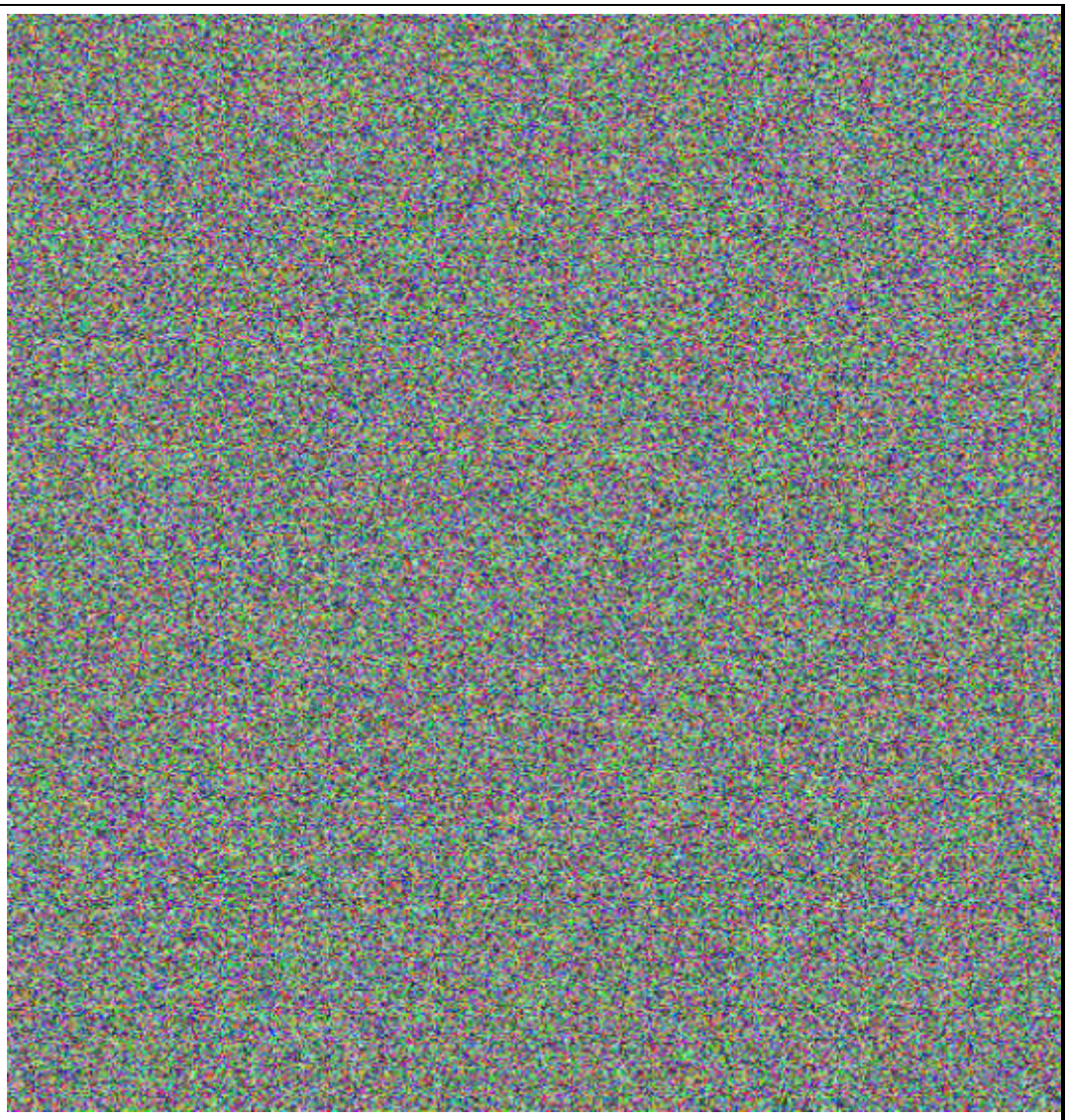
要求：将实验获得的结果进行描述，基本内容包括：

1. 用 DES 加、解密 BMP 灰度图像的明文图像和密文图像（ECB 和 CBC 模式分别给出）。

使用 Lena 图进行实验：

原图：

使用 CBC 加密的结果：



使用 ECB 加密的结果：

对于轮廓比较清晰的图像：

使用 CBC 加密的结果：

使用 ECB 加密的结果：

可以看到 ECB 算法加密的结果会暴露出

2. 给出 DES 的雪崩性质的分析结果：分别给出在加密第 1 轮后的不同位数，第 2 轮后的不同位数，…，以及第 16 轮后的不同位数。

使用 “lqjxliang”作为密钥，得到的结果如下：

After 0 round: 5
After 1 round: 18

After 2 round: 31
After 3 round: 34
After 4 round: 35
After 5 round: 34
After 6 round: 33
After 7 round: 38
After 8 round: 37
After 9 round: 35
After 10 round: 32
After 11 round: 29
After 12 round: 33
After 13 round: 30
After 14 round: 27
After 15 round: 30

可以看到随着轮数的增加，改变一位对于整个密文的影响大概在一半左右
‘

3. 给出 DES 的完整性性质的分析结果：统计出输出中每位为 0 或 1 频率。
得到的实验结果如下：
[156, 102, 103, 137, 140, 115, 119, 101, 107, 108, 147, 144, 128, 129, 152, 99, 136, 131, 131, 133, 158, 138, 96, 119, 131, 115, 139, 105, 141, 140, 127, 149, 113, 121, 145, 108, 140, 120, 164, 131, 98, 130, 157, 128, 134, 156, 128, 118, 138, 129, 103, 134, 135, 133, 104, 92, 135, 147, 146, 154, 118, 141, 116, 153]

[0.609375, 0.3984375, 0.40234375, 0.53515625, 0.546875, 0.44921875, 0.46484375, 0.39453125, 0.41796875, 0.421875, 0.57421875, 0.5625, 0.5, 0.50390625, 0.59375, 0.38671875, 0.53125, 0.51171875, 0.51171875, 0.51953125, 0.6171875, 0.5390625, 0.375, 0.46484375, 0.51171875, 0.44921875, 0.54296875, 0.41015625, 0.55078125, 0.546875, 0.49609375, 0.58203125, 0.44140625, 0.47265625, 0.56640625, 0.421875, 0.546875, 0.46875, 0.640625, 0.51171875, 0.3828125, 0.5078125, 0.61328125, 0.5, 0.5234375, 0.609375, 0.5, 0.4609375, 0.5390625, 0.50390625, 0.40234375, 0.5234375, 0.52734375, 0.51953125, 0.40625, 0.359375, 0.52734375, 0.57421875, 0.5703125, 0.6015625, 0.4609375, 0.55078125, 0.453125, 0.59765625]

可以看到概率基本在 0.5 左右

4. 给出 DES 的差分分析的分析结果：给出差分值的最大情况及其出现的位置。
S1 盒的差分分布表如下所示：

(0, [64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(1, [0, 0, 0, 6, 0, 2, 4, 4, 0, 10, 12, 4, 10, 6, 2, 4])
(2, [0, 0, 0, 8, 0, 4, 4, 4, 0, 6, 8, 6, 12, 6, 4, 2])
(3, [14, 4, 2, 2, 10, 6, 4, 2, 6, 4, 4, 0, 2, 2, 2, 0])
(4, [0, 0, 0, 6, 0, 10, 10, 6, 0, 4, 6, 4, 2, 8, 6, 2])
(5, [4, 8, 6, 2, 2, 4, 4, 2, 0, 4, 4, 0, 12, 2, 4, 6])

(6, [0, 4, 2, 4, 8, 2, 6, 2, 8, 4, 4, 2, 4, 2, 0, 12])
(7, [2, 4, 10, 4, 0, 4, 8, 4, 2, 4, 8, 2, 2, 2, 4, 4])
(8, [0, 0, 0, 12, 0, 8, 8, 4, 0, 6, 2, 8, 8, 2, 2, 4])
(9, [10, 2, 4, 0, 2, 4, 6, 0, 2, 2, 8, 0, 10, 0, 2, 12])
(10, [0, 8, 6, 2, 2, 8, 6, 0, 6, 4, 6, 0, 4, 0, 2, 10])
(11, [2, 4, 0, 10, 2, 2, 4, 0, 2, 6, 2, 6, 6, 4, 2, 12])
(12, [0, 0, 0, 8, 0, 6, 6, 0, 0, 6, 6, 4, 6, 6, 14, 2])
(13, [6, 6, 4, 8, 4, 8, 2, 6, 0, 6, 4, 6, 0, 2, 0, 2])
(14, [0, 4, 8, 8, 6, 6, 4, 0, 6, 6, 4, 0, 0, 4, 0, 8])
(15, [2, 0, 2, 4, 4, 6, 4, 2, 4, 8, 2, 2, 2, 6, 8, 8])
(16, [0, 0, 0, 0, 0, 0, 2, 14, 0, 6, 6, 12, 4, 6, 8, 6])
(17, [6, 8, 2, 4, 6, 4, 8, 6, 4, 0, 6, 6, 0, 4, 0, 0])
(18, [0, 8, 4, 2, 6, 6, 4, 6, 6, 4, 2, 6, 6, 0, 4, 0])
(19, [2, 4, 4, 6, 2, 0, 4, 6, 2, 0, 6, 8, 4, 6, 4, 6])
(20, [0, 8, 8, 0, 10, 0, 4, 2, 8, 2, 2, 4, 4, 8, 4, 0])
(21, [0, 4, 6, 4, 2, 2, 4, 10, 6, 2, 0, 10, 0, 4, 6, 4])
(22, [0, 8, 10, 8, 0, 2, 2, 6, 10, 2, 0, 2, 0, 6, 2, 6])
(23, [4, 4, 6, 0, 10, 6, 0, 2, 4, 4, 4, 6, 6, 6, 2, 0])
(24, [0, 6, 6, 0, 8, 4, 2, 2, 2, 4, 6, 8, 6, 6, 2, 2])
(25, [2, 6, 2, 4, 0, 8, 4, 6, 10, 4, 0, 4, 2, 8, 4, 0])
(26, [0, 6, 4, 0, 4, 6, 6, 6, 6, 2, 2, 0, 4, 4, 6, 8])
(27, [4, 4, 2, 4, 10, 6, 6, 4, 6, 2, 2, 4, 2, 2, 4, 2])
(28, [0, 10, 10, 6, 6, 0, 0, 12, 6, 4, 0, 0, 2, 4, 4, 0])
(29, [4, 2, 4, 0, 8, 0, 0, 2, 10, 0, 2, 6, 6, 6, 14, 0])
(30, [0, 2, 6, 0, 14, 2, 0, 0, 6, 4, 10, 8, 2, 2, 6, 2])
(31, [2, 4, 10, 6, 2, 2, 2, 8, 6, 8, 0, 0, 0, 4, 6, 4])
(32, [0, 0, 0, 10, 0, 12, 8, 2, 0, 6, 4, 4, 4, 2, 0, 12])
(33, [0, 4, 2, 4, 4, 8, 10, 0, 4, 4, 10, 0, 4, 0, 2, 8])
(34, [10, 4, 6, 2, 2, 8, 2, 2, 2, 2, 6, 0, 4, 0, 4, 10])
(35, [0, 4, 4, 8, 0, 2, 6, 0, 6, 6, 2, 10, 2, 4, 0, 10])
(36, [12, 0, 0, 2, 2, 2, 2, 0, 14, 14, 2, 0, 2, 6, 2, 4])
(37, [6, 4, 4, 12, 4, 4, 4, 10, 2, 2, 2, 0, 4, 2, 2, 2])
(38, [0, 0, 4, 10, 10, 10, 2, 4, 0, 4, 6, 4, 4, 4, 2, 0])
(39, [10, 4, 2, 0, 2, 4, 2, 0, 4, 8, 0, 4, 8, 8, 4, 4])
(40, [12, 2, 2, 8, 2, 6, 12, 0, 0, 2, 6, 0, 4, 0, 6, 2])
(41, [4, 2, 2, 10, 0, 2, 4, 0, 0, 14, 10, 2, 4, 6, 0, 4])
(42, [4, 2, 4, 6, 0, 2, 8, 2, 2, 14, 2, 6, 2, 6, 2, 2])
(43, [12, 2, 2, 2, 4, 6, 6, 2, 0, 2, 6, 2, 6, 0, 8, 4])
(44, [4, 2, 2, 4, 0, 2, 10, 4, 2, 2, 4, 8, 8, 4, 2, 6])
(45, [6, 2, 6, 2, 8, 4, 4, 4, 2, 4, 6, 0, 8, 2, 0, 6])
(46, [6, 6, 2, 2, 0, 2, 4, 6, 4, 0, 6, 2, 12, 2, 6, 4])
(47, [2, 2, 2, 2, 2, 6, 8, 8, 2, 4, 4, 6, 8, 2, 4, 2])
(48, [0, 4, 6, 0, 12, 6, 2, 2, 8, 2, 4, 4, 6, 2, 2, 4])
(49, [4, 8, 2, 10, 2, 2, 2, 2, 6, 0, 0, 2, 2, 4, 10, 8])

(50, [4, 2, 6, 4, 4, 2, 2, 4, 6, 6, 4, 8, 2, 2, 8, 0])
(51, [4, 4, 6, 2, 10, 8, 4, 2, 4, 0, 2, 2, 4, 6, 2, 4])
(52, [0, 8, 16, 6, 2, 0, 0, 12, 6, 0, 0, 0, 0, 8, 0, 6])
(53, [2, 2, 4, 0, 8, 0, 0, 0, 14, 4, 6, 8, 0, 2, 14, 0])
(54, [2, 6, 2, 2, 8, 0, 2, 2, 4, 2, 6, 8, 6, 4, 10, 0])
(55, [2, 2, 12, 4, 2, 4, 4, 10, 4, 4, 2, 6, 0, 2, 2, 4])
(56, [0, 6, 2, 2, 2, 0, 2, 2, 4, 6, 4, 4, 4, 6, 10, 10])
(57, [6, 2, 2, 4, 12, 6, 4, 8, 4, 0, 2, 4, 2, 4, 4, 0])
(58, [6, 4, 6, 4, 6, 8, 0, 6, 2, 2, 6, 2, 2, 6, 4, 0])
(59, [2, 6, 4, 0, 0, 2, 4, 6, 4, 6, 8, 6, 4, 4, 6, 2])
(60, [0, 10, 4, 0, 12, 0, 4, 2, 6, 0, 4, 12, 4, 4, 2, 0])
(61, [0, 8, 6, 2, 2, 6, 0, 8, 4, 4, 0, 4, 0, 12, 4, 4])
(62, [4, 8, 2, 2, 2, 4, 4, 14, 4, 2, 0, 2, 0, 8, 4, 4])
(63, [4, 8, 4, 2, 4, 0, 2, 4, 4, 2, 4, 8, 8, 6, 2, 2])

差分值最大为输入差分为 52 , 输出差分为 2 时。

5. 在某个线性函数计算的 S-盒，随机产生的 S-盒，自行设计一种 S-盒上依次给出上述分析结果。

线性计算的 S 盒：

雪崩效应：

After 0 round: 2
After 1 round: 3
After 2 round: 3
After 3 round: 4
After 4 round: 5
After 5 round: 5
After 6 round: 4
After 7 round: 5
After 8 round: 7
After 9 round: 8
After 10 round: 7
After 11 round: 5
After 12 round: 4
After 13 round: 3
After 14 round: 3
After 15 round: 4

完整性：

[246, 243, 17, 234, 245, 21, 21, 17, 241, 20, 26, 14, 236, 239, 246, 233, 16, 236, 233, 18, 240, 235, 20, 22, 24, 237, 246, 239, 11, 20, 238, 21, 243, 15, 16, 239, 18, 230, 17, 241, 15, 18, 21, 23,

234, 242, 15, 15, 17, 21, 16, 28, 243, 21, 18, 241, 241, 239, 244, 17, 236, 21, 241, 237]

[0.9609375, 0.94921875, 0.06640625, 0.9140625, 0.95703125, 0.08203125, 0.08203125, 0.06640625, 0.94140625, 0.078125, 0.1015625, 0.0546875, 0.921875, 0.93359375, 0.9609375, 0.91015625, 0.0625, 0.921875, 0.91015625, 0.0703125, 0.9375, 0.91796875, 0.078125, 0.0859375, 0.09375, 0.92578125, 0.9609375, 0.93359375, 0.04296875, 0.078125, 0.9296875, 0.08203125, 0.94921875, 0.05859375, 0.0625, 0.93359375, 0.0703125, 0.8984375, 0.06640625, 0.94140625, 0.05859375, 0.0703125, 0.08203125, 0.08984375, 0.9140625, 0.9453125, 0.05859375, 0.05859375, 0.06640625, 0.08203125, 0.0625, 0.109375, 0.94921875, 0.08203125, 0.0703125, 0.94140625, 0.94140625, 0.93359375, 0.953125, 0.06640625, 0.921875, 0.08203125, 0.94140625, 0.92578125]

差分分析：

(0, [64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(1, [64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(2, [0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(3, [0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(4, [0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(5, [0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(6, [0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(7, [0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(8, [0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(9, [0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(10, [0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(11, [0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(12, [0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(13, [0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(14, [0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0])
(15, [0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0])
(16, [0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0])
(17, [0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0])
(18, [0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0])
(19, [0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0])
(20, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0])
(21, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0])
(22, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0])
(23, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0])
(24, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0])
(25, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0])
(26, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0])
(27, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0])
(28, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0])

(29, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0])
(30, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64])
(31, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64])
(32, [64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(33, [64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(34, [0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(35, [0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(36, [0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(37, [0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(38, [0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(39, [0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(40, [0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(41, [0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(42, [0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(43, [0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(44, [0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(45, [0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(46, [0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0])
(47, [0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0, 0])
(48, [0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0])
(49, [0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0, 0])
(50, [0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0])
(51, [0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0, 0])
(52, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0])
(53, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0, 0])
(54, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0])
(55, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0, 0])
(56, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0])
(57, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0, 0])
(58, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0])
(59, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0, 0])
(60, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0])
(61, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64, 0])
(62, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64])
(63, [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 64])

差分值最大是 64，在分布表中有规律的出现

随机 S 盒：

雪崩效应：

After 0 round: 5

After 1 round: 15
After 2 round: 27
After 3 round: 34
After 4 round: 29
After 5 round: 25
After 6 round: 26
After 7 round: 25
After 8 round: 28
After 9 round: 31
After 10 round: 31
After 11 round: 31
After 12 round: 34
After 13 round: 36
After 14 round: 35
After 15 round: 31

完整性：

[135, 117, 126, 132, 134, 148, 161, 161, 159, 144, 95, 113, 126, 127, 132, 120, 97, 115, 117, 151, 125, 109, 136, 125, 132, 146, 132, 141, 104, 143, 139, 143, 131, 126, 109, 155, 127, 110, 141, 125, 137, 78, 112, 137, 108, 128, 113, 107, 168, 102, 141, 109, 116, 110, 106, 121, 119, 129, 156, 127, 125, 131, 137, 111]

[0.52734375, 0.45703125, 0.4921875, 0.515625, 0.5234375, 0.578125, 0.62890625, 0.62890625, 0.62109375, 0.5625, 0.37109375, 0.44140625, 0.4921875, 0.49609375, 0.515625, 0.46875, 0.37890625, 0.44921875, 0.45703125, 0.58984375, 0.48828125, 0.42578125, 0.53125, 0.48828125, 0.515625, 0.5703125, 0.515625, 0.55078125, 0.40625, 0.55859375, 0.54296875, 0.55859375, 0.51171875, 0.4921875, 0.42578125, 0.60546875, 0.49609375, 0.4296875, 0.55078125, 0.48828125, 0.53515625, 0.3046875, 0.4375, 0.53515625, 0.421875, 0.5, 0.44140625, 0.41796875, 0.65625, 0.3984375, 0.55078125, 0.42578125, 0.453125, 0.4296875, 0.4140625, 0.47265625, 0.46484375, 0.50390625, 0.609375, 0.49609375, 0.48828125, 0.51171875, 0.53515625, 0.43359375]

差分分析：

(0, [64, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
(1, [2, 0, 12, 0, 2, 0, 4, 4, 6, 4, 2, 8, 6, 6, 6, 2])
(2, [4, 2, 8, 0, 2, 6, 2, 4, 2, 4, 2, 4, 8, 6, 4, 6])
(3, [2, 0, 2, 2, 8, 2, 4, 4, 2, 14, 2, 6, 0, 6, 8, 2])
(4, [2, 6, 0, 0, 0, 6, 8, 2, 4, 6, 6, 10, 2, 0, 2, 10])
(5, [6, 4, 2, 10, 4, 0, 6, 8, 2, 2, 0, 4, 6, 6, 2, 2])
(6, [2, 0, 2, 4, 6, 6, 8, 4, 2, 6, 2, 4, 2, 10, 0, 6])
(7, [0, 14, 8, 0, 2, 0, 4, 4, 6, 2, 0, 4, 0, 2, 4, 14])
(8, [6, 6, 8, 6, 2, 0, 2, 6, 4, 8, 4, 0, 0, 4, 2, 6])

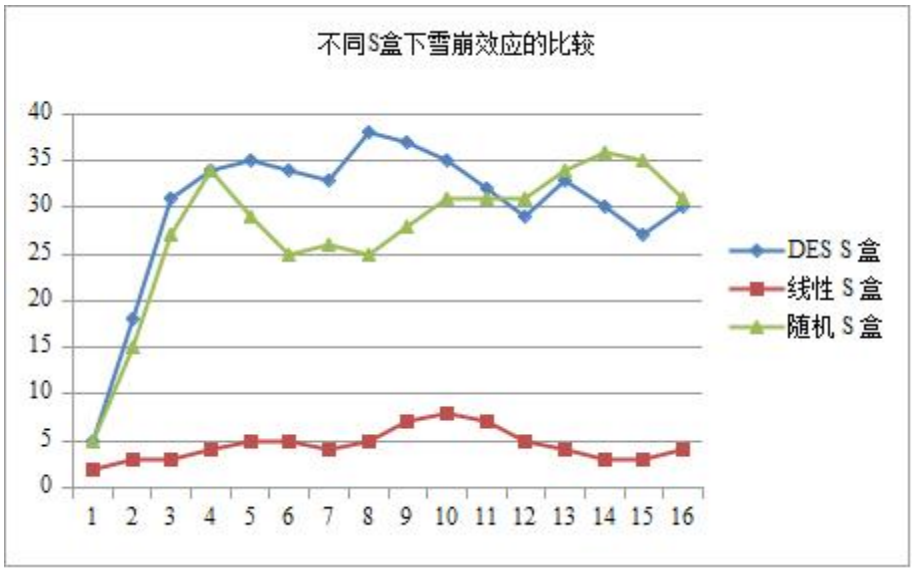
(9, [2, 10, 2, 4, 2, 2, 0, 2, 2, 2, 0, 8, 12, 6, 4, 6])
(10, [4, 2, 4, 2, 4, 14, 2, 4, 6, 2, 4, 6, 2, 4, 2, 2])
(11, [4, 4, 2, 6, 0, 2, 4, 6, 4, 10, 2, 2, 4, 2, 8, 4])
(12, [6, 4, 2, 2, 0, 0, 0, 6, 16, 4, 2, 2, 6, 2, 4, 8])
(13, [4, 2, 8, 2, 6, 4, 2, 8, 6, 4, 10, 2, 2, 2, 2, 0])
(14, [4, 0, 6, 4, 8, 2, 0, 4, 0, 2, 4, 2, 0, 6, 14, 8])
(15, [4, 4, 4, 8, 2, 6, 6, 6, 4, 6, 2, 2, 8, 0, 2, 0])
(16, [6, 6, 2, 0, 8, 4, 8, 6, 2, 2, 4, 4, 2, 4, 4, 2])
(17, [2, 0, 4, 10, 4, 2, 6, 0, 6, 0, 0, 4, 14, 10, 0, 2])
(18, [2, 6, 2, 2, 8, 8, 8, 0, 8, 4, 6, 4, 2, 4, 0, 0])
(19, [4, 2, 4, 4, 2, 2, 6, 0, 4, 8, 6, 6, 0, 4, 6, 6])
(20, [8, 4, 2, 0, 0, 2, 0, 4, 4, 4, 8, 8, 8, 4, 2, 6])
(21, [2, 0, 8, 6, 6, 2, 6, 10, 0, 6, 2, 6, 0, 2, 4, 4])
(22, [2, 4, 8, 6, 2, 4, 0, 2, 2, 6, 4, 6, 4, 10, 2, 2])
(23, [6, 4, 4, 6, 4, 8, 6, 2, 6, 4, 2, 2, 2, 0, 2, 6])
(24, [6, 6, 2, 6, 4, 4, 2, 6, 6, 4, 2, 2, 2, 6, 0, 6])
(25, [4, 4, 2, 2, 2, 2, 4, 4, 0, 2, 10, 10, 4, 4, 2, 8])
(26, [0, 4, 10, 2, 10, 10, 0, 4, 2, 0, 4, 4, 0, 4, 6, 4])
(27, [4, 4, 0, 4, 2, 2, 4, 4, 6, 16, 4, 4, 4, 4, 0, 2])
(28, [0, 4, 2, 2, 4, 4, 6, 6, 4, 6, 4, 8, 2, 2, 6, 4])
(29, [6, 4, 2, 2, 12, 6, 8, 0, 0, 2, 2, 4, 6, 6, 0, 4])
(30, [6, 4, 8, 0, 4, 6, 0, 0, 4, 6, 6, 8, 2, 2, 6, 2])
(31, [2, 4, 14, 4, 2, 0, 6, 4, 6, 6, 2, 0, 8, 2, 4, 0])
(32, [8, 6, 6, 4, 4, 2, 10, 0, 4, 2, 4, 4, 4, 4, 0, 2])
(33, [4, 4, 2, 0, 4, 2, 8, 4, 2, 8, 4, 6, 4, 6, 4, 2])
(34, [6, 6, 4, 2, 10, 0, 6, 2, 0, 0, 4, 0, 0, 0, 10, 14])
(35, [6, 2, 4, 2, 6, 0, 4, 4, 10, 4, 2, 0, 12, 6, 0, 2])
(36, [2, 0, 6, 2, 0, 4, 2, 8, 4, 6, 8, 6, 0, 10, 2, 4])
(37, [14, 8, 2, 2, 2, 6, 10, 4, 2, 0, 0, 2, 2, 0, 4, 6])
(38, [4, 4, 0, 0, 8, 6, 4, 2, 0, 6, 4, 8, 0, 6, 0, 12])
(39, [0, 4, 8, 6, 6, 2, 2, 4, 2, 10, 0, 4, 2, 4, 4, 6])
(40, [0, 2, 8, 0, 6, 6, 4, 6, 4, 2, 6, 8, 0, 6, 4, 2])
(41, [6, 2, 2, 0, 0, 2, 6, 6, 4, 8, 4, 0, 6, 10, 4, 4])
(42, [6, 4, 4, 6, 4, 6, 2, 8, 2, 6, 0, 2, 4, 6, 2, 2])
(43, [4, 6, 2, 8, 0, 6, 2, 0, 4, 2, 2, 6, 2, 6, 8, 6])
(44, [2, 12, 2, 2, 6, 4, 0, 0, 6, 4, 8, 6, 2, 6, 2, 2])
(45, [0, 4, 4, 8, 4, 2, 12, 6, 6, 2, 6, 0, 4, 0, 4, 2])
(46, [0, 2, 6, 2, 2, 2, 6, 4, 10, 2, 2, 2, 2, 6, 8, 8])
(47, [4, 0, 0, 8, 8, 6, 8, 2, 6, 4, 4, 4, 6, 0, 0, 4])
(48, [2, 4, 8, 2, 6, 6, 6, 10, 0, 0, 6, 4, 2, 2, 2, 4])
(49, [6, 6, 0, 0, 4, 2, 4, 2, 2, 10, 4, 2, 8, 8, 4, 2])
(50, [14, 2, 0, 4, 4, 4, 4, 4, 6, 0, 6, 2, 4, 0, 2, 8])
(51, [0, 2, 10, 6, 4, 0, 4, 2, 4, 2, 2, 4, 8, 10, 4, 2])
(52, [4, 2, 6, 6, 0, 4, 2, 0, 0, 4, 8, 4, 12, 8, 0, 4])

(53, [2, 10, 2, 4, 6, 0, 4, 8, 2, 4, 2, 8, 2, 4, 4, 2])
(54, [4, 4, 0, 4, 2, 4, 4, 6, 4, 4, 4, 2, 4, 0, 6, 12])
(55, [6, 2, 2, 4, 10, 2, 4, 6, 8, 4, 0, 4, 6, 4, 0, 2])
(56, [6, 6, 0, 6, 8, 8, 4, 2, 0, 2, 4, 6, 0, 8, 2, 2])
(57, [0, 0, 0, 6, 2, 2, 8, 6, 4, 10, 2, 8, 2, 6, 2, 6])
(58, [8, 0, 8, 2, 4, 0, 4, 6, 6, 2, 0, 0, 4, 6, 6, 8])
(59, [6, 2, 2, 4, 10, 4, 2, 2, 2, 8, 0, 6, 2, 4, 4, 6])
(60, [4, 4, 4, 0, 4, 6, 2, 0, 4, 0, 6, 8, 4, 4, 4, 10])
(61, [6, 4, 6, 4, 2, 2, 10, 6, 2, 4, 0, 0, 6, 8, 4, 0])
(62, [6, 2, 2, 2, 4, 6, 0, 6, 12, 6, 2, 2, 2, 0, 4, 8])
(63, [4, 4, 12, 2, 8, 4, 2, 0, 0, 2, 4, 6, 2, 2, 8, 4])

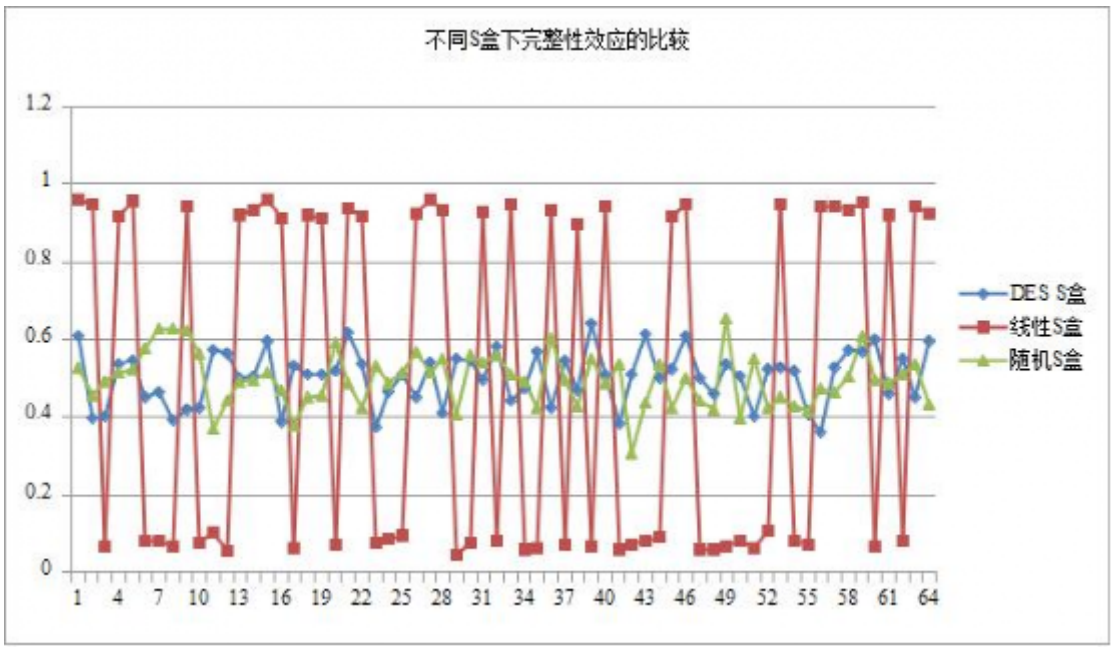
差分值最大在输入差为 27，输出差为 9 时。

6． 将四种不同 S-盒的上述分析结果放在一起进行比较。

通过将上面的结果比较，我们可以发现使用线性的 S 盒对于雪崩效应和完整性有着非常大的影响，随机 S 盒的效果接近与原来的 S 盒。三者比较效果如下：



同样，对于完整性，实验线性的 S 盒会导致输出结果出现两极分化，对比图如下：



对于差分分析，使用线性的 S 盒会使得差分分布表有些明显的变化规律。

程序设计成绩(4 分)		实验结果成绩(3 分)	
实验报告成绩(3 分)		**总成绩**	
指导教师签字		日期	