Justin Cabral

ECE/CS 578 Assignment 2

**1-A) 64 bits DES input and 56-bit key are given. Apply two round reduced DES and find the related ciphertext.**

**You can find the related permutation tables and S-boxes from the lecture notes or DES Standardization**

**Input:**   
01101010 00110101 01010011 00100001 01101000 10011111 11011100 00101010

* Step 1: Initial Input Permutation

01010101

01100110

01100010

00101110

01100000

10011011

11110001

10100101

* Step 2: Split Permutated Input into L0 and R0 which are 32bits each

**L0** **R0**

01010101 01100000

01100110 10011011

01100010 11110001

00101110 10100101

* Step 3: Transform Key to pass to F function in 1st round

**56-bit Key:**

10101100 01111101 11010011 11001110 00011001 10010110 00100011

**Split Key into two halves**

**C0** **D0**

10101100 11100001

01111101 10011001

11010011 01100010

1100 0011

**Key Rotation for Round 1, move bits left by 1**

**C1** **D1**

01011000 11000011

11111011 00110010

10100111 11000100

1001 0111

**Combine C1 + D1 (56bits)** **Permutate (48bits)** **Round 1 Key (48bits)**

01011000 01110101 01110101

11111011 10011111 10011111

10100111 00100011 00100011

10011100 00000111 00000111

00110011 01000011 01000011

00101100 00101110 00101110

01000111

* Step 4: Process the F Function

**Expand R0 from 32 to 48 bits**

**R0** **R0 Expansion**

01100000 101100

10011011 000001

11110001 010011

10100101 110111

111110

100011

110100

001010

**XOR the R0 expansion with the 1st Round Key**

**R0 Expansion**  **Round 1 Key** **XORed Output**

101100 01110101 110001

000001 10011111 011000

010011 00100011 101111

110111 00000111 010100

111110 01000011 111111

100011 00101110 010111

110100 111000

001010 100100

**Pass 6 bits of XOR output into each S-Box s1 through s8 get 4 bit output each**

110001 011000 101111 010100

**s1** **s2** **s3** **s4**

0101 1100 0111 1000

111111 010111 111000 100100

**s5** **s6** **s7** **s8**

0011 1110 0000 0100

**Permutate combined output from the S-Boxes for final function output**

**S-Box Output** **Permutated Function Output**

01011100 00110100

01111000 00101001

00111110 10000000

00000100 11111110

* Step 5: Finish Round 1

**R1 = L0 XOR Output of F Function**

**L0** **Function Output** **XORed = R1**

01010101 00110100 01100001

01100110 00101001 01001111

01100010 10000000 11100010

00101110 11111110 11010000

**L1 = R0** **R1**

01100000 01100001

10011011 01001111

11110001 11100010

10100101 11010000

**----= ROUND 1 COMPLETE =-----------------------------------------------------------------------------------------**

**----= ROUND 2 START =------------------------------------------------------------------------------------------------**

* Step 1: Transform Key to pass to function in 2nd round

**Key Halves**

**C1** **D1**

01011000 11000011

11111011 00110010

10100111 11000100

1001 0111

**Key rotation for round 2, move bits left by 1**

**C2** **D2**

10110001 10000110

11110111 01100101

01001111 10001000

0010 1111

**Combine C2 + D2(56bits)**  **Permutate(48bits)** **Round 2 Key(48bits)**

10110001 10111010 10111010

11110111 10111011 10111011

01001111 01101000 01101000

00101000 00000100 00000100

01100110 01001111 01001111

01011000 11010010 11010010

10001111

* Step 2: Process The F Function

**Expand R1 from 32 to 48 bits**

**R1** **R1 Expansion**

01100001 001100

01001111 000010

11100010 101001

11010000 011111

111100

000101

011010

100000

**XOR the R1 expansion with the 2nd Round Key**

**R1 Expansion**  **Round 2 Key** **XORed Output**

001100 10111010 100010

000010 10111011 101001

101001 01101000 000100

011111 00000100 110111

111100 01001111 111101

000101 11010010 000001

011010 100101

100000 110010

**Pass 6 bits of XORed output into each S-Box, s1 through s8 get 4-bit output each**

100010 101001 000100 110111

**s1** **s2** **s3** **s4**

0001 0011 1001 1011

111101 000001 100101 110010

**s5** **s6** **s7** **s8**

0101 1010 1101 0110

**Permutate combined output from the S-Boxes for final function output**

**S-Box Output** **Permutated Function Output**

00010011 11110110

10011011 01110110

01011010 01000001

11010110 01100011

* Step 3: Finish Round 2

**R2 = L1 XOR Output of Function**

**L1** **Function Output** **XORed = R2**

01100000 11110110 10010110

10011011 01110110 11101101

11110001 01000001 10110000

10100101 01100011 11000110

**L2 = R1** **R2**

01100001 10010110

01001111 11101101

11100010 10110000

11010000 11000110

* Step 4: Swap Sides, Combine, and Inverse Permutation

**Swap L2 & R2 Bit Position + Combine** **Inverse Permutation**

10010110 10110000

11101101 01101001

10110000 01110001

11000110 00110000

01100001 01000110

01001111 10011100

11100010 10111011

11010000 01011111

**--------------------------------------------= 2 ROUND DES COMPLETE =-------------------------------------------**

**Ciphertext Output**

10110000

01101001

01110001

00110000

01000110

10011100

10111011

01011111

**1-B) Choose one S-box (out of 8) and prepare Difference Distribution Table for the chosen S-box. List the number of output difference occurrences for every input difference. You can refer to Heys Differential Cryptanalysis Tutorial page 21. You can find the document on Canvas (idc tutorial.pdf).**

Difference Distribution Table for DES S-Box S1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| OUTPUT DIFFERENCE | | | | | | | | | | | | | | | | | |
| I  N  P  U  T  D  I  F  F  E  R  E  N  C  E |  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
| 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 |
| A | 0 | 8 | 6 | 2 | 2 | 8 | 6 | 0 | 6 | 4 | 6 | 0 | 4 | 0 | 2 | 10 |
| B | 2 | 4 | 0 | 10 | 2 | 2 | 4 | 0 | 2 | 6 | 2 | 6 | 6 | 4 | 2 | 12 |
| C | 0 | 0 | 0 | 8 | 0 | 6 | 6 | 0 | 0 | 6 | 6 | 4 | 6 | 6 | 14 | 2 |
| D | 6 | 6 | 4 | 8 | 4 | 8 | 2 | 6 | 0 | 6 | 4 | 6 | 0 | 2 | 0 | 2 |
| E | 0 | 4 | 8 | 8 | 6 | 6 | 4 | 0 | 6 | 6 | 4 | 0 | 0 | 4 | 0 | 8 |
| F | 2 | 0 | 2 | 4 | 4 | 6 | 4 | 2 | 4 | 8 | 2 | 2 | 2 | 6 | 8 | 8 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 14 | 0 | 6 | 6 | 12 | 4 | 6 | 8 | 6 |
| 11 | 6 | 8 | 2 | 4 | 6 | 4 | 8 | 6 | 4 | 0 | 6 | 6 | 0 | 4 | 0 | 0 |
| 12 | 0 | 8 | 4 | 2 | 6 | 6 | 4 | 6 | 6 | 4 | 2 | 6 | 6 | 0 | 4 | 0 |
| 13 | 2 | 4 | 4 | 6 | 2 | 0 | 4 | 6 | 2 | 0 | 6 | 8 | 4 | 6 | 4 | 6 |
| 14 | 0 | 8 | 8 | 0 | 10 | 0 | 4 | 2 | 8 | 2 | 2 | 4 | 4 | 8 | 4 | 0 |
| 15 | 0 | 4 | 6 | 4 | 2 | 2 | 4 | 10 | 6 | 2 | 0 | 10 | 0 | 4 | 6 | 4 |
| 16 | 0 | 8 | 10 | 8 | 0 | 2 | 2 | 6 | 10 | 2 | 0 | 2 | 0 | 6 | 2 | 6 |
| 17 | 4 | 4 | 6 | 0 | 10 | 6 | 0 | 2 | 4 | 4 | 4 | 6 | 6 | 6 | 2 | 0 |
| 18 | 0 | 6 | 6 | 0 | 8 | 4 | 2 | 2 | 2 | 4 | 6 | 8 | 6 | 6 | 2 | 2 |
| 19 | 2 | 6 | 2 | 4 | 0 | 8 | 4 | 6 | 10 | 4 | 0 | 4 | 2 | 8 | 4 | 0 |
| 1A | 0 | 6 | 4 | 0 | 4 | 6 | 6 | 6 | 6 | 2 | 2 | 0 | 4 | 4 | 6 | 8 |

**Table Continued........**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| I  N  P  U  T  D  I  F  F  E  R  E  N  C  E | 1B | 4 | 4 | 2 | 4 | 10 | 6 | 6 | 4 | 6 | 2 | 2 | 4 | 2 | 2 | 4 | 2 |
| 1C | 0 | 10 | 10 | 6 | 6 | 0 | 0 | 12 | 6 | 4 | 0 | 0 | 2 | 4 | 4 | 0 |
| 1D | 4 | 2 | 4 | 0 | 8 | 0 | 0 | 2 | 10 | 0 | 2 | 6 | 6 | 6 | 14 | 0 |
| 1E | 0 | 2 | 6 | 0 | 14 | 2 | 0 | 0 | 6 | 4 | 10 | 8 | 2 | 2 | 6 | 2 |
| 1F | 2 | 4 | 10 | 6 | 2 | 2 | 2 | 8 | 6 | 8 | 0 | 0 | 0 | 4 | 6 | 4 |
| 20 | 0 | 0 | 0 | 10 | 0 | 12 | 8 | 2 | 0 | 6 | 4 | 4 | 4 | 2 | 0 | 12 |
| 21 | 0 | 4 | 2 | 4 | 4 | 8 | 10 | 0 | 4 | 4 | 10 | 0 | 4 | 0 | 2 | 8 |
| 22 | 10 | 4 | 6 | 2 | 2 | 8 | 2 | 2 | 2 | 2 | 6 | 0 | 4 | 0 | 4 | 10 |
| 23 | 0 | 4 | 4 | 8 | 0 | 2 | 6 | 0 | 6 | 6 | 2 | 10 | 2 | 4 | 0 | 10 |
| 24 | 12 | 0 | 0 | 2 | 2 | 2 | 2 | 0 | 14 | 14 | 2 | 0 | 2 | 6 | 2 | 4 |
| 25 | 6 | 4 | 4 | 12 | 4 | 4 | 4 | 10 | 2 | 2 | 2 | 0 | 4 | 2 | 2 | 2 |
| 26 | 0 | 0 | 4 | 10 | 10 | 10 | 2 | 4 | 0 | 4 | 6 | 4 | 4 | 4 | 2 | 0 |
| 27 | 10 | 4 | 2 | 0 | 2 | 4 | 2 | 0 | 4 | 8 | 0 | 4 | 8 | 8 | 4 | 4 |
| 28 | 12 | 2 | 2 | 8 | 2 | 6 | 12 | 0 | 0 | 2 | 6 | 0 | 4 | 0 | 6 | 2 |
| 29 | 4 | 2 | 2 | 10 | 0 | 2 | 4 | 0 | 0 | 14 | 10 | 2 | 4 | 6 | 0 | 4 |
| 2A | 4 | 2 | 4 | 6 | 0 | 2 | 8 | 2 | 2 | 14 | 2 | 6 | 2 | 6 | 2 | 2 |
| 2B | 12 | 2 | 2 | 2 | 4 | 6 | 6 | 2 | 0 | 2 | 6 | 2 | 6 | 0 | 8 | 4 |
| 2C | 4 | 2 | 2 | 4 | 0 | 2 | 10 | 4 | 2 | 2 | 4 | 8 | 8 | 4 | 2 | 6 |
| 2D | 6 | 2 | 6 | 2 | 8 | 4 | 4 | 4 | 2 | 4 | 6 | 0 | 8 | 2 | 0 | 6 |
| 2E | 6 | 6 | 2 | 2 | 0 | 2 | 4 | 6 | 4 | 0 | 6 | 2 | 12 | 2 | 6 | 4 |
| 2F | 2 | 2 | 2 | 2 | 2 | 6 | 8 | 8 | 2 | 4 | 4 | 6 | 8 | 2 | 4 | 2 |
| 30 | 0 | 4 | 6 | 0 | 12 | 6 | 2 | 2 | 8 | 2 | 4 | 4 | 6 | 2 | 2 | 4 |
| 31 | 4 | 8 | 2 | 10 | 2 | 2 | 2 | 2 | 6 | 0 | 0 | 2 | 2 | 4 | 10 | 8 |
| 32 | 4 | 2 | 6 | 4 | 4 | 2 | 2 | 4 | 6 | 6 | 4 | 8 | 2 | 2 | 8 | 0 |
| 33 | 4 | 4 | 6 | 2 | 10 | 8 | 4 | 2 | 4 | 0 | 2 | 2 | 4 | 6 | 2 | 4 |
| 34 | 0 | 8 | 16 | 6 | 2 | 0 | 0 | 12 | 6 | 0 | 0 | 0 | 0 | 8 | 0 | 6 |
| 35 | 2 | 2 | 4 | 0 | 8 | 0 | 0 | 0 | 14 | 4 | 6 | 8 | 0 | 2 | 14 | 0 |
| 36 | 2 | 6 | 2 | 2 | 8 | 0 | 2 | 2 | 4 | 2 | 6 | 8 | 6 | 4 | 10 | 0 |
| 37 | 2 | 2 | 12 | 4 | 2 | 4 | 4 | 10 | 4 | 4 | 2 | 6 | 0 | 2 | 2 | 4 |
| 38 | 0 | 6 | 2 | 2 | 2 | 0 | 2 | 2 | 4 | 6 | 4 | 4 | 4 | 6 | 10 | 10 |
| 39 | 6 | 2 | 2 | 4 | 12 | 6 | 4 | 8 | 4 | 0 | 2 | 4 | 2 | 4 | 4 | 0 |
| 3A | 6 | 4 | 6 | 4 | 6 | 8 | 0 | 6 | 2 | 2 | 6 | 2 | 2 | 6 | 4 | 0 |
| 3B | 2 | 6 | 4 | 0 | 0 | 2 | 4 | 6 | 4 | 6 | 8 | 6 | 4 | 4 | 6 | 2 |
| 3C | 0 | 10 | 4 | 0 | 12 | 0 | 4 | 2 | 6 | 0 | 4 | 12 | 4 | 4 | 2 | 0 |
| 3D | 0 | 8 | 6 | 2 | 2 | 6 | 0 | 8 | 4 | 4 | 0 | 4 | 0 | 12 | 4 | 4 |
| 3E | 4 | 8 | 2 | 2 | 2 | 4 | 4 | 14 | 4 | 2 | 0 | 2 | 0 | 8 | 4 | 4 |
| 3F | 4 | 8 | 4 | 2 | 4 | 0 | 2 | 4 | 4 | 2 | 4 | 8 | 8 | 6 | 2 | 2 |