SECTION – I

Dr.Q,
Chief Technology Scientist,
Secret Inteligence Service.

M,
Chief of Secret Intelligence Service,
MI6.

Dear M,

1. What is your cipher called?

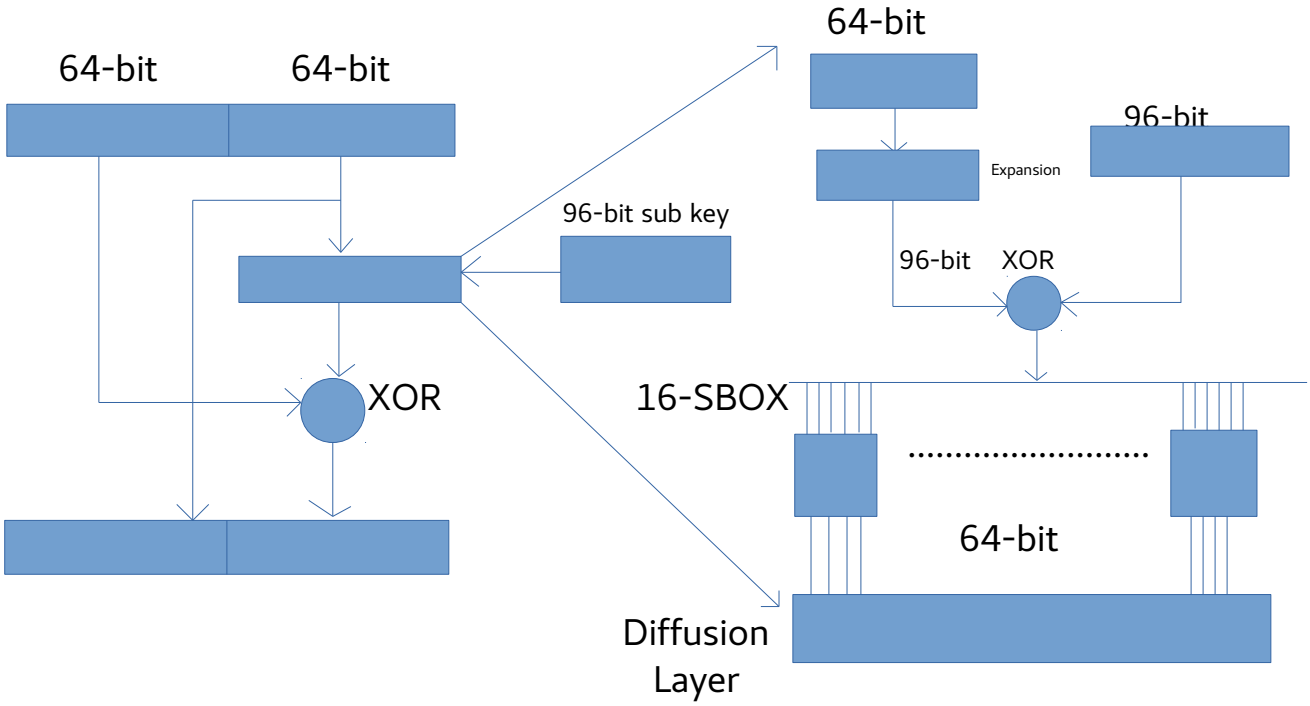
The cipher is called Descartes.

2. How many rounds does your cipher have?

The cipher has 20 rounds.

3. Explain one round of your cipher.

Assuming the encryption function takes N keys as input itself,



4. How did you arrive at the number of rounds for your cipher?

As we have a key input of 128 bits, the brute force method gives us a complexity of $\frac{2^{128}}{N}$. By using 20 rounds we have the time complexity for linear and differential cryptanalysis greater than that of brute force. Total bias in linear cryptanalysis is greater than $\frac{2^{128}}{N}$.

5. How did you choose the s-boxes size and type?

I have chosen a feistel cipher and the function has a compression sbox. Here I have taken a 96 bit sub key and also expanded the input to introduce non-linearity. Then used the compression sbox to give a 64-bit output. I have 16 6x4 compression sboxes for this purpose. Thus my implementation will be effecient and easy.

6. How did you go about choosing the s-box(es) mappings?

A sbox should satisfy many properties like non-linearity, balancedness, SAC etc. I have tried some of the mappings which obey most of the properties and also has a balanced propagation ratio.

7. For your s-box(es), explain / demonstrate how you satisfied the following properties -

The sbox I used for the design of the cipher is -

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

(i) The Balancedness Property

If we check the first row in binary form,

1	1	1	1
0	0	0	1
1	0	0	0
1	1	1	0
0	1	1	0
1	0	1	1
0	0	1	1
0	1	0	0
1	0	0	1
0	1	1	1
0	0	1	0
1	1	0	1
1	1	0	0
0	0	0	0
0	1	0	1
1	0	1	0

Each column has 8 1's and 8 0's. And the rest of the rows are permutation of the first row itself. So, the balacedness is satisfied

(ii) Strict Avalanche Criterion (SAC)

For this to be satisfied, f(x)$\oplus$f(x$\oplus\alpha$) is balanced. And α is such that its hamming weight is 1. So I took α to be 000001.

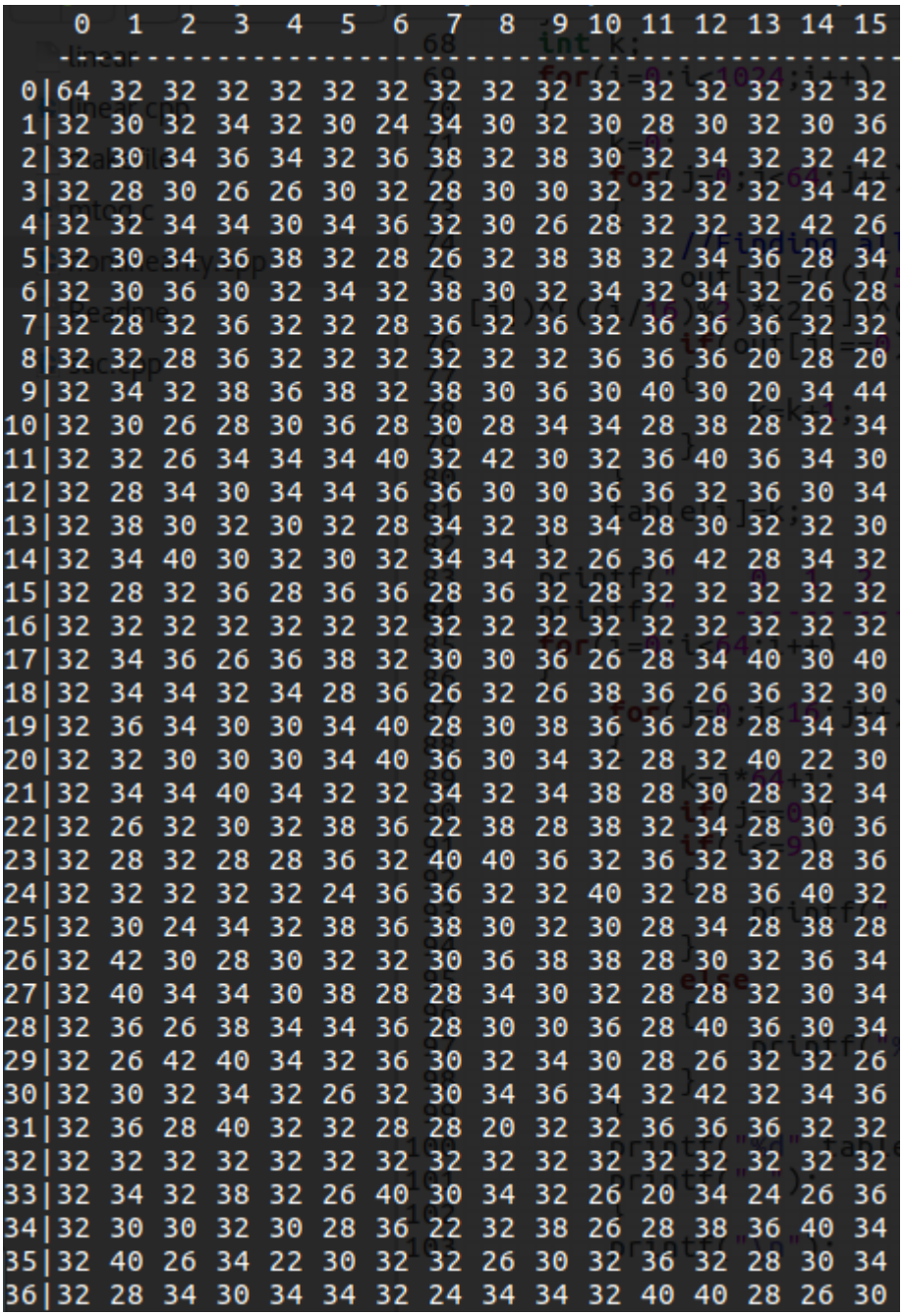
The code I have written to calculate the satisfaction of this phenomenon shows that the sbox doesn't satisfy SAC.

(iii) Non-linearity

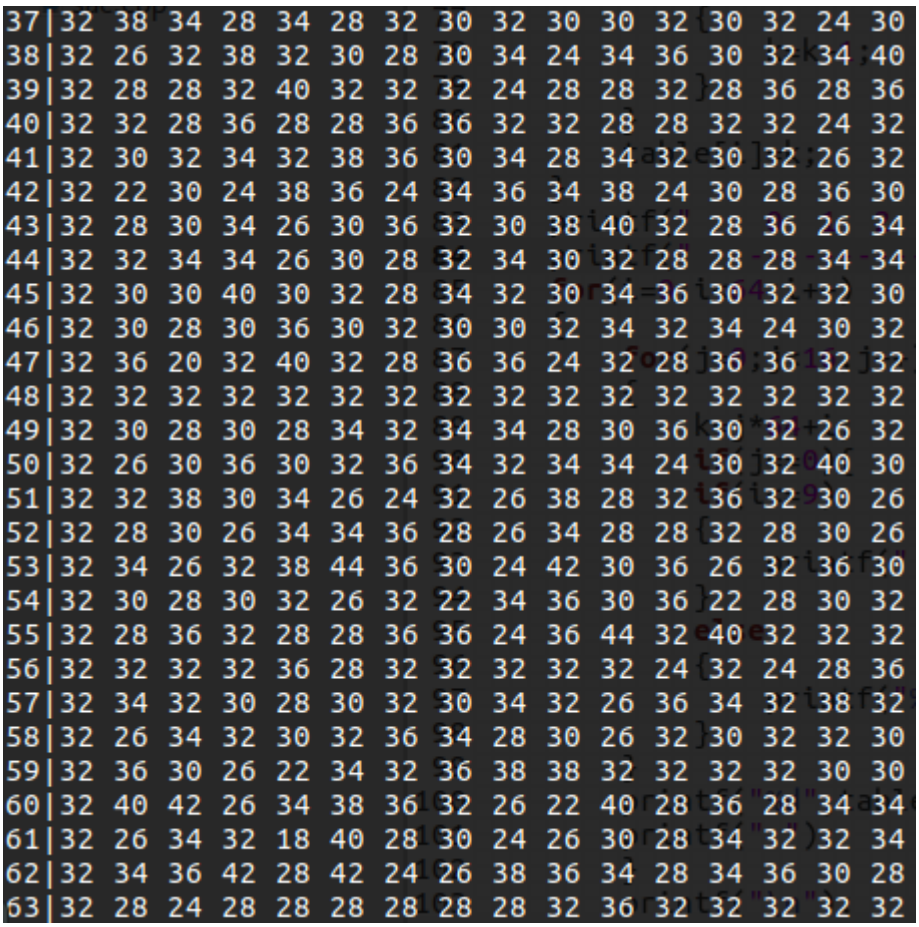
I have written a c++ code to check the non-linearity of all outputs and the corresponding values for y1,y2,y3,y4 are 22,24,22,22 respectively.

8. Draw the linear approximation table for your s-box(es).

I have written a code for the same and the output it gave is shown in the below figure -



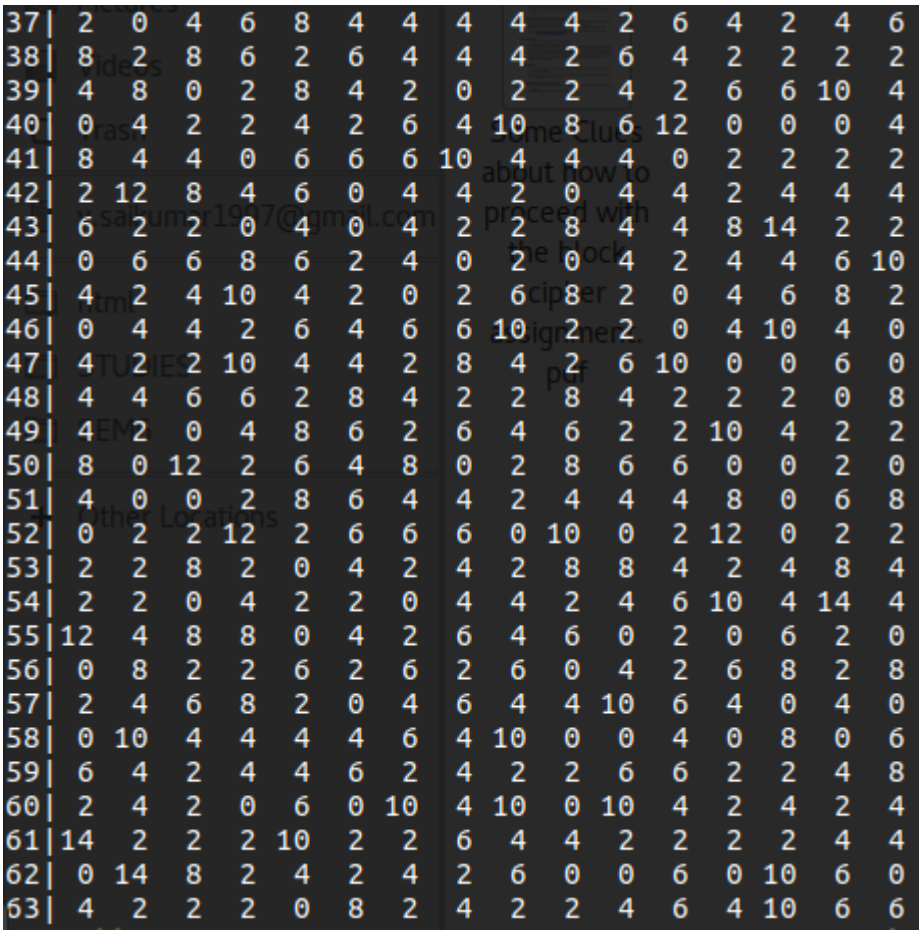
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	64	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32
1	32	30	32	34	32	30	24	34	30	32	30	28	30	32	30	36
2	32	30	34	36	34	32	36	38	32	38	30	32	34	32	32	42
3	32	28	30	26	26	30	32	28	30	30	32	32	32	32	34	42
4	32	32	34	34	30	34	36	32	30	26	28	32	32	32	42	26
5	32	30	34	36	38	32	28	26	32	38	38	32	34	36	28	34
6	32	30	36	30	32	34	32	38	30	32	34	32	34	32	26	28
7	32	28	32	36	32	32	28	36	32	36	32	36	36	36	32	32
8	32	32	28	36	32	32	32	32	32	32	36	36	36	20	28	20
9	32	34	32	38	36	38	32	38	30	36	30	40	30	20	34	44
10	32	30	26	28	30	36	28	30	28	34	34	28	38	28	32	34
11	32	32	26	34	34	34	40	32	42	30	32	36	40	36	34	30
12	32	28	34	30	34	34	36	36	30	30	36	36	32	36	30	34
13	32	38	30	32	30	32	28	34	32	38	34	28	30	32	32	30
14	32	34	40	30	32	30	32	34	34	32	26	36	42	28	34	32
15	32	28	32	36	28	36	36	28	36	32	28	32	32	32	32	32
16	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32
17	32	34	36	26	36	38	32	30	30	36	26	28	34	40	30	40
18	32	34	34	32	34	28	36	26	32	26	38	36	26	36	32	30
19	32	36	34	30	30	34	40	28	30	38	36	36	28	28	34	34
20	32	32	30	30	30	34	40	36	30	34	32	28	32	40	22	30
21	32	34	34	40	34	32	32	34	32	34	38	28	30	28	32	34
22	32	26	32	30	32	38	36	22	38	28	38	32	34	28	30	36
23	32	28	32	28	28	36	32	40	40	36	32	36	32	32	28	36
24	32	32	32	32	32	24	36	36	32	32	40	32	28	36	40	32
25	32	30	24	34	32	38	36	38	30	32	30	28	34	28	38	28
26	32	42	30	28	30	32	32	30	36	38	38	28	30	32	36	34
27	32	40	34	34	30	38	28	28	34	30	32	28	28	32	30	34
28	32	36	26	38	34	34	36	28	30	30	36	28	40	36	30	34
29	32	26	42	40	34	32	36	30	32	34	30	28	26	32	32	26
30	32	30	32	34	32	26	32	30	34	36	34	32	42	32	34	36
31	32	36	28	40	32	32	28	28	20	32	32	36	36	36	32	32
32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32
33	32	34	32	38	32	26	40	30	34	32	26	20	34	24	26	36
34	32	30	30	32	30	28	36	22	32	38	26	28	38	36	40	34
35	32	40	26	34	22	30	32	32	26	30	32	36	32	28	30	34
36	32	28	34	30	34	34	32	24	34	34	32	40	40	28	26	30



9. Draw the differential distribution table for your s-box(es).

I have written a code for the same and the output it gave is shown in the below figure -

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	64	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	2	0	4	6	4	0	0	4	6	10	10	12	6
2	0	0	0	0	0	4	0	12	0	10	10	8	8	6	2	4
3	0	12	6	4	6	6	6	4	2	6	2	4	4	0	2	0
4	0	0	0	2	0	4	6	8	0	10	12	4	6	0	8	4
5	0	6	6	4	10	4	4	2	6	2	6	6	2	2	2	2
6	0	2	2	6	0	14	0	8	0	4	2	4	6	6	6	4
7	0	4	10	6	0	4	2	2	8	4	4	4	4	4	0	8
8	0	2	0	6	0	4	12	4	0	2	6	8	0	8	6	6
9	0	4	2	8	2	4	10	2	6	4	2	6	4	8	2	0
10	0	8	6	2	8	0	2	10	6	2	2	2	4	4	4	4
11	0	6	6	6	4	2	4	4	6	4	6	2	6	0	4	4
12	0	6	4	6	8	2	0	2	10	10	0	0	4	4	2	6
13	0	6	8	4	4	6	2	2	2	0	2	6	2	8	4	8
14	0	6	8	6	12	0	4	0	8	2	4	2	0	0	8	4
15	0	2	6	2	10	6	6	0	10	4	2	2	4	4	2	4
16	0	2	0	6	0	2	6	4	0	12	6	6	10	2	6	2
17	4	6	4	6	4	6	4	2	4	4	4	4	4	4	0	4
18	2	0	2	6	2	4	8	8	2	0	4	4	2	4	10	6
19	4	6	8	6	2	4	2	0	6	4	6	4	4	6	0	2
20	16	2	4	8	2	8	2	6	4	2	0	0	2	0	2	6
21	4	2	2	2	2	4	6	2	8	4	0	6	6	6	4	6
22	6	2	6	0	6	2	2	4	4	12	4	2	6	2	2	4
23	0	8	2	2	10	6	6	2	0	2	4	6	10	0	4	2
24	4	4	4	6	6	0	6	2	8	2	8	4	4	4	0	2
25	6	2	4	0	6	6	6	2	0	2	2	4	2	8	2	12
26	8	12	2	0	6	2	0	2	4	2	2	2	0	10	10	2
27	4	0	8	6	6	4	0	4	4	2	4	4	6	6	4	2
28	0	4	4	4	4	6	4	2	6	2	6	6	0	6	8	2
29	2	6	6	4	0	8	6	4	2	4	2	6	4	2	6	2
30	0	6	2	2	6	0	4	16	4	4	2	0	0	4	6	8
31	4	2	6	6	2	2	2	4	8	6	10	6	4	0	0	2
32	0	0	0	6	0	10	8	4	0	4	0	2	12	2	8	8
33	4	12	8	0	2	4	2	0	2	4	8	6	2	6	0	4
34	8	4	0	0	2	6	2	2	4	8	4	8	8	0	4	4
35	4	2	4	4	0	0	8	14	4	0	6	4	2	4	4	4
36	10	0	8	4	2	10	2	0	4	8	2	0	4	2	0	8



37	2	0	4	6	8	4	4	4	4	4	2	6	4	2	4	6
38	8	2	8	6	2	6	4	4	4	2	6	4	2	2	2	2
39	4	8	0	2	8	4	2	0	2	2	4	2	6	6	10	4
40	0	4	2	2	4	2	6	4	10	8	6	12	0	0	0	4
41	8	4	4	0	6	6	6	10	4	4	4	0	2	2	2	2
42	2	12	8	4	6	0	4	4	2	0	4	4	2	4	4	4
43	6	2	2	0	4	0	4	2	2	8	4	4	8	14	2	2
44	0	6	6	8	6	2	4	0	2	0	4	2	4	4	6	10
45	4	2	4	10	4	2	0	2	6	8	2	0	4	6	8	2
46	0	4	4	2	6	4	6	6	10	2	2	0	4	10	4	0
47	4	2	2	10	4	4	2	8	4	2	6	10	0	0	6	0
48	4	4	6	6	2	8	4	2	2	8	4	2	2	2	0	8
49	4	2	0	4	8	6	2	6	4	6	2	2	10	4	2	2
50	8	0	12	2	6	4	8	0	2	8	6	6	0	0	2	0
51	4	0	0	2	8	6	4	4	2	4	4	4	8	0	6	8
52	0	2	2	12	2	6	6	6	0	10	0	2	12	0	2	2
53	2	2	8	2	0	4	2	4	2	8	8	4	2	4	8	4
54	2	2	0	4	2	2	0	4	4	2	4	6	10	4	14	4
55	12	4	8	8	0	4	2	6	4	6	0	2	0	6	2	0
56	0	8	2	2	6	2	6	2	6	0	4	2	6	8	2	8
57	2	4	6	8	2	0	4	6	4	4	10	6	4	0	4	0
58	0	10	4	4	4	4	6	4	10	0	0	4	0	8	0	6
59	6	4	2	4	4	6	2	4	2	2	6	6	2	2	4	8
60	2	4	2	0	6	0	10	4	10	0	10	4	2	4	2	4
61	14	2	2	2	10	2	2	6	4	4	2	2	2	2	4	4
62	0	14	8	2	4	2	4	2	6	0	0	6	0	10	6	0
63	4	2	2	2	0	8	2	4	2	2	4	6	4	10	6	6

10. How did you choose the diffusion layer for your cipher?

I have a permutation layer in which the outputs of my first sbox go to the first bits of 1st,2nd,3rd,4th sections of the output, outputs of my second sbox go to first bits of 5th,6th,7th,8th sections of the output and so on. The inputs and outputs are referred from left to right.

11. How many rounds would it take to obtain complete diffusion?

It would take minimum 3 rounds to achieve maximum diffusion as a toggle in any one of the inputs will have an effect on multiple sboxes and this will carry on.