DES

Klaus Pommerening
Fachbereich Physik, Mathematik, Informatik
der Johannes-Gutenberg-Universität
Saarstraße 21
D-55099 Mainz

April 7, 1997—English version February 6, 2015
last change January 20, 2021

The "Data Encryption Standard" (DES) is essentially a development by an IBM research group around Feistel and Coppersmith. The NSA was involved: It arranged for a modification of the S-boxes and a reduction of the key length to 56 bits. Contrary to all speculations both of these changes didn't weaken the security.

DES was published in 1975, and standardized by NBS (National Bureau of Standards—now NIST) in the USA in 1977. The objective was to provide a reliable cipher for sensitive (but not top secret) data of the administration for the next 10 or 15 years.

The standard requires a hardware implementation of the algorithm. The proper name of the algorithm is DEA, but usually also software implementations are denoted by DES. From 1989 to 1998 the US administration restricted the export of DES chips.

DES encrypts 64 bit blocks using a 56 bit key. The encryption of a block starts with a fixed (known) permutation, and ends with the inverse permutation. Although this permutation is known it yields a first bit of diffusion. In between there are 16 rounds that increase diffusion and confusion. The only difference between the single rounds consists in the selection of a different 48 bit subset from the key.

The decryption algorithm is almost identical with the encryption algorithm with the only difference that it runs through the key selection in the reverse direction.

In the following sections we describe the algorithmus in steps "outwards from the interior". In the figures $\oplus$ denotes the bitwise addition mod 2 (XOR).

1 The Kernel Map

The innermost layer of DES is the "kernel map"

$$f\colon \mathbb{F}_2^{32} \times \mathbb{F}_2^{48} \longrightarrow \mathbb{F}_2^{32},$$

that takes 32 text bits and 48 key bits as input. First some of the 32 text bits are repeated, blowing them up to 48 bits. This "expansion map"

$$E\colon \mathbb{F}_2^{32} \longrightarrow \mathbb{F}_2^{48}$$

is given by the following table:

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

The correct interpretation of this table is

$$E(b_1 b_2 \dots b_{32}) = b_{32} b_1 b_2 b_3 \dots b_{31} b_{32} b_1.$$

The expanded 48 bits and the 48 bit partial key are added (as binary vectors). The resulting 48 bits are divided into 8 groups each consisting of 6 bits. These groups are fed into the S-boxes 1 to 8:

$$S_j\colon \mathbb{F}_2^6 \longrightarrow \mathbb{F}_2^4 \quad (j = 1, \dots, 8).$$

The description of the S-boxes is in the next section.

The S-boxes together make up the substitution

$$S\colon \mathbb{F}_2^{48} \longrightarrow \mathbb{F}_2^{32}.$$

Finally we apply the "P-box" (permutation)

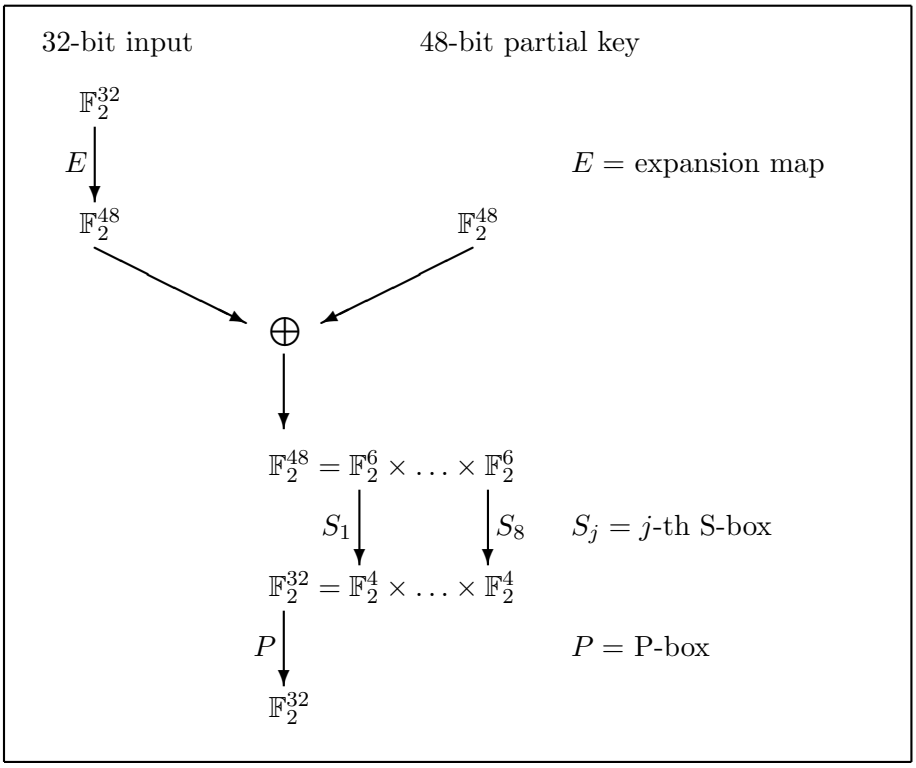
$$P\colon \mathbb{F}_2^{32} \longrightarrow \mathbb{F}_2^{32}$$

that is given by the following table meaning

$$P(b_1 b_2 \dots b_{32}) = b_{16} b_7 \dots b_4 b_{25}.$$

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

The complete kernel map is in the following figure:



2 The S-Boxes

Each of the eight S-boxes S_j is given by a 4×16-matrix defined by the table

S_1	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
S_2	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
S_3	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
S_4	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
S_5	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
S_6	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
S_7	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
S_8	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

Each row is a permutation of the numbers $0, \ldots, 15$. To get $S_j(b_1 \ldots b_6)$ we interpret $b_1 b_6$ as binary representation of a number in $\{0, 3\}$, and $b_2 b_3 b_4 b_5$ as binary representation of a number in $\{0, 15\}$. Then in the matrix for S_j we go to row $b_1 b_6$, column $b_2 b_3 b_4 b_5$, and find a number there. We take the binary representation of this number. Example:

$$S_3(101100) = 0011 \quad \rightarrow \quad \text{Row 2, Column 6, number is 3.}$$

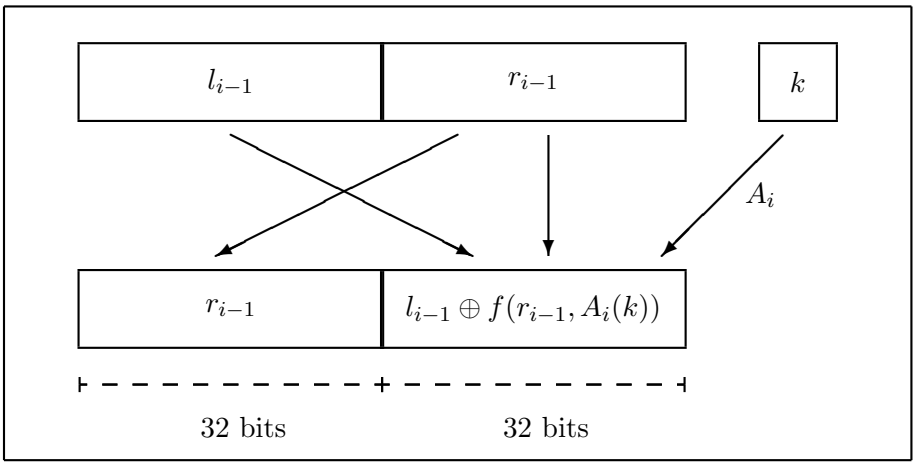
3 The Rounds

The 16 rounds of DES consist of maps

$$R_i \colon \mathbb{F}_2^{64} \times \mathbb{F}_2^{56} \longrightarrow \mathbb{F}_2^{64} \quad (i = 1, \ldots, 16),$$

that are defined in the following figure, using the i-th key selection

$$A_i \colon \mathbb{F}_2^{56} \longrightarrow \mathbb{F}_2^{48} \quad (i = 1, \ldots, 16).$$



The rounds only differ by their key selections $A_i(k)$. We recognize the FEISTEL scheme.

4 The Key Selection

To complete the description of the rounds we have yet to describe the key selection. First we expand the 56 bit key to 64 bits by appending a parity bit after each 7 bit subblock. However it doesn't matter which bit we append: the additional bits never enter the algorithm. In any case the first step is a map

$$\mathrm{Par}\colon \mathbb{F}_2^{56} \longrightarrow \mathbb{F}_2^{64}.$$

In the second step we extract the original 56 bits, but in a different order, given by the following table.

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

We have constructed a map

$$\mathrm{PC}_1\colon \mathbb{F}_2^{64} \longrightarrow \mathbb{F}_2^{56}$$

("Permuted Choice 1"). Now we divide the 56 bits into two 28 bit halves, and cyclically shift these to the left, all in all 16 times. This gives 16 maps

$$\mathrm{LS}_i\colon \mathbb{F}_2^{28} \longrightarrow \mathbb{F}_2^{28} \quad (i = 1, \dots 16).$$

the amount of the shifts is given by the table:

1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

The first two shifts are by one bit, the next 6 ones by two bits, and so on. To get the i-th key selection A_i we apply the "Permuted Choice 2" after the i-th shift:

$$\mathrm{PC}_2\colon \mathbb{F}_2^{56} \longrightarrow \mathbb{F}_2^{48}$$

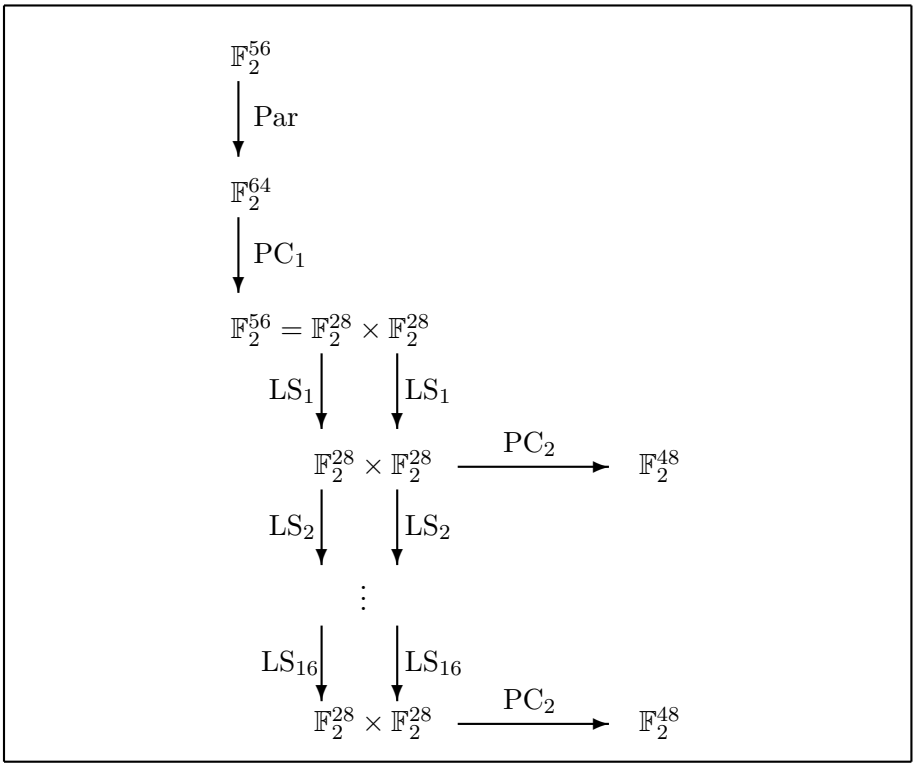
where the 48 bits are chosen in the order of the following table (omitting the bits 9, 18, 22, 25, 35, 38, 43, 54).

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

The complete key selection is

$$A_i = \mathrm{PC}_2 \circ \mathrm{LS}_i \circ \cdots \circ \mathrm{LS}_1 \circ \mathrm{PC}_1 \circ \mathrm{Par}\,.$$

We summarize this construction in the following diagram:



5 The Complete Algorithm

The last thing to do is to describe the initial permutation

$$\mathrm{IP} : \mathbb{F}_2^{64} \longrightarrow \mathbb{F}_2^{64}.$$

This is done by the following table:

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

The inverse of IP is the final permutation IP^{-1}. For convenience here is the corresponding table:

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

Now the complete DES algorithm DES_k with key $k \in \mathbb{F}_2^{56}$ is the composition

$$\mathbb{F}_2^{64} \xrightarrow{\mathrm{IP}} \mathbb{F}_2^{64} \xrightarrow{R_1(\bullet,k)} \dots \xrightarrow{R_{16}(\bullet,k)} \mathbb{F}_2^{64} \xrightarrow{T} \mathbb{F}_2^{64} \xrightarrow{\mathrm{IP}^{-1}} \mathbb{F}_2^{64}.$$

Here T is the interchange of the left and right 32 bit halves. The effect of this additional interchange is that DES_k^{-1} looks exactly like DES_k except that the order of the rounds is reversed.

Remark. The initial and final permutations maybe lead to a convenient wiring of input and output contacts on small processors. They have no cryptological effect because the cryptanalyst simply may strip them off. For a software implementation they function as brakes—but one must not omit them for a standard conforming implementation.