

第三章

對稱式金鑰密碼系統-

資料加密標準

3.1 S-DES概述

- * 1970年代Horst Feistel為美國IBM電腦公司研發出“Lucifer”系統。
- * 美國國家標準局(NBS, 現為 NIST)在1973 年徵求構想書，希望能訂定國際加密標準。
- * DES最後在1997年1月發表於《聯邦公報》稱為FIPS 46。
- * DES是一種對稱加密演算法其加密方式為區塊加密。

3.1 S-DES 架構圖

* 精簡版的DES(S-DES)不是一個安全的加密演算法。

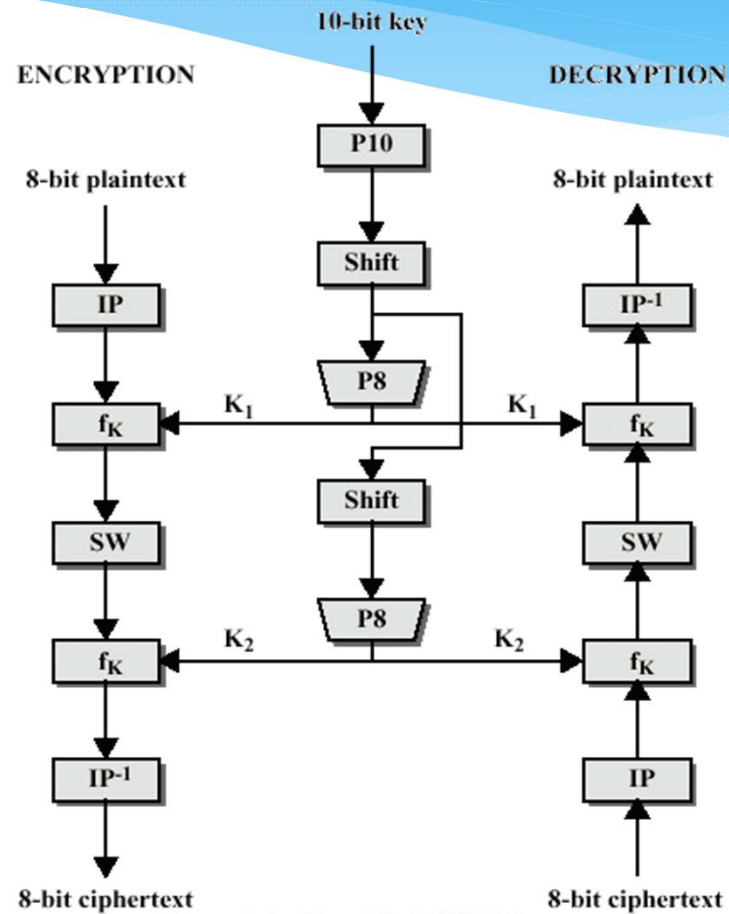


Figure 3.1 Simplified DES Scheme

3.1 S-DES 架構圖

- 總共有七種函數(Seven functions)
 - 密鑰產生函數(Key generation)
 - P10: 10位元的置換(permutation)
(3 5 2 7 4 10 1 9 8 6)
 - Shift: 分為左邊五個位元與右邊五個位元，個別向左移一位元。
 - P8: 10位元對應到8位元的函數
 - 加密函數(Encryption)
 - IP: 10位元初始置換(initial permutation)
 - f_K : 一個較複雜的函數(complex function)
 - SW: 左邊 4 位元和右邊 4 位元互換
 - IP^{-1} : IP的反函數。

3.1 S-DES 架構圖

- f_K : 8位元對應8位元的函數
 - 最複雜的部分
 - 結合排列和替換
 - $f_K(L, R) = (L \oplus F(R, SK), R)$
 - SK : 給定密鑰所產生的子密鑰 K_i ($i = 1, 2, \dots$)
 - L : input 的左邊 4 bits
 - R : input 的右邊 4 bits
 - F : 將 R 利用子密鑰 SK 加密的 4 位元對到 4 位元的函數

例題： 假設輸入為 (10111101) 且 $F(1101, SK) = (1110)$ 求 f_K

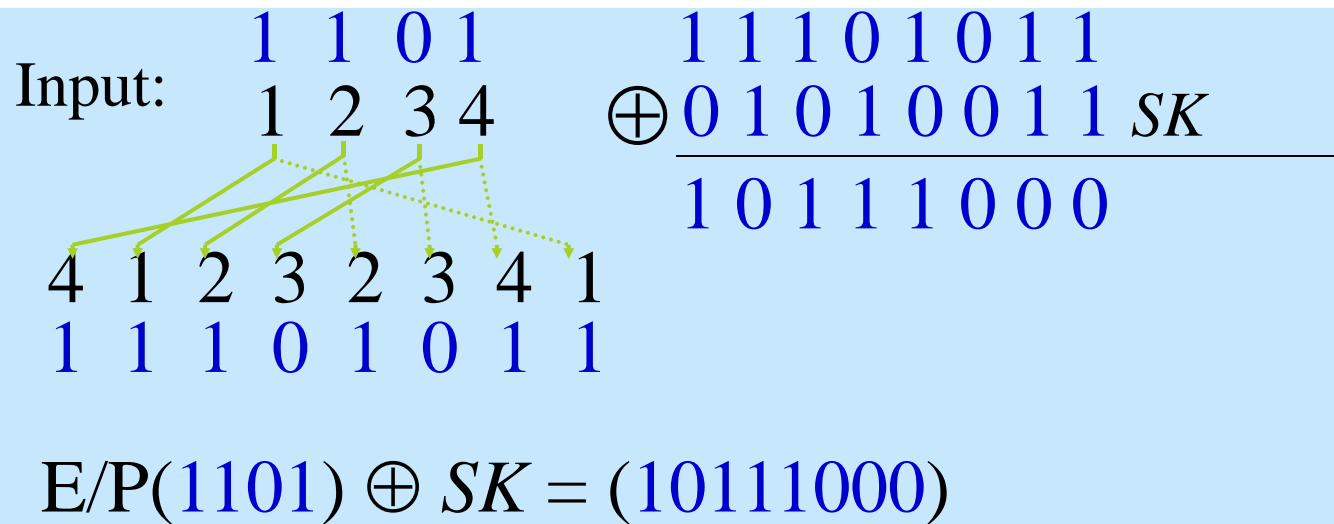
$$f_K(1011, 1101) = ((1011) \oplus (1110), (1101)) = (01011101)$$

3.1 S-DES 架構圖

F:將R利用子密鑰 SK 加密的4位元對應到4位元的函數。

- Input: 4-bit number $R=(n_1n_2n_3n_4)$.
- 先做一個擴展/置換(expansion/permutation)(E/P)的運算得8位元字串($n_4n_1n_2n_3n_2n_3n_4n_1$)
- 然後與子密鑰 SK 執行XOR運算

例題:



3.1 S-DES 架構圖

- F: 將R利用子密鑰SK加密的4位元對到4位元的函數。
 - Input: 4-bit 個數 $R = (n_1 n_2 n_3 n_4)$ 。
 - 計算 $(n_4 n_1 n_2 n_3 \ n_2 n_3 n_4 n_1) \oplus SK$ 得8位元字串 $(l_1 l_2 l_3 l_4 \ r_1 r_2 r_3 r_4)$ 。
 - 利用S-box S0與S1得到一個4位元 $(a_1 a_2 b_1 b_2)$ 。
 - 利用左邊 4 bits $l_1 l_2 l_3 l_4$ 與 S-box S0(4×4矩陣) 來產生 2-bit output $(a_1 a_2)$ 。
 - 利用右邊 4 bits $r_1 r_2 r_3 r_4$ 與 S-box S1(4×4矩陣) 來產生另外 2-bit output $(b_1 b_2)$ 。

3.1 S-DES 架構圖

- S-box 的4-bit input中, 第1、4 bits 用來決定要參考矩陣中的哪一行, 第2、3 bits 則用來決定要參考矩陣中的哪一列。
- 由此得出矩陣中的某一項, 然後將其化為二進位表示式。

例題：

$S0(0100)=3$ or $(11)_2$

$S1(0010)=1$ or $(01)_2$

$S0 =$

	0	1	2	3
0	1	0	3	2
1	3	2	1	0
2	0	2	1	3
3	3	1	3	2

$S1 =$

	0	1	2	3
0	0	1	2	3
1	2	0	1	3
2	3	0	1	0
3	2	1	0	3

3.1 S-DES 架構圖

- 再利用一個4位元的重置的 P4將重新排列得到一位元的輸出。

例題： 假設4位元的首置如下 $P4(b_1b_2b_3b_4)=b_2b_4b_3b_1$
重新排列為

$P4(S\text{-box}(01000100))$

$= P4(1110)$

$= 1011$

1 1 1 0

$b_1 b_2 b_3 b_4$

$b_2 b_4 b_3 b_1$

$S0 =$

	0	1	2	3
0	1	0	3	2
1	3	2	1	0
2	0	2	1	3
3	3	1	3	2

$S1 =$

	0	1	2	3
0	0	1	2	3
1	2	0	1	3
2	3	0	1	0
3	2	1	0	3

$S0(0100)=3$ or $(11)_2$

$S1(0100)=1$ or $(10)_2$

3.1 S-DES 架構圖

- * S-DES公式整理，加密金鑰的產生公式

$$K_1 = P8(\text{Shift}(P10(\text{key})))$$

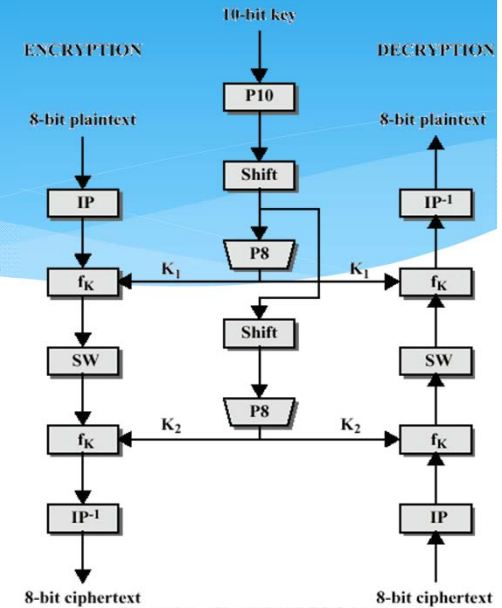
$$K_2 = P8(\text{Shift}(\text{Shift}(P10(\text{key}))))$$

- * 明文的加密公式:

$$\text{Ciphertext} = IP^{-1}(f_{K_2}(SW(f_{K_1}(IP(\text{plaintext}))))))$$

- * 密文的解密公式:

$$\text{Plaintext} = IP^{-1}(f_{K_1}(SW(f_{K_2}(IP(\text{ciphertext}))))))$$



例題(加密鑰是 K_1 的 K_2 產生)

- 密鑰產生函數
 - $P10=(3\ 5\ 2\ 7\ 4\ 10\ 1\ 9\ 8\ 6)$
 - $P8:(n_1n_2n_3n_4n_5n_6n_7n_8n_9n_{10}) \rightarrow (n_6n_3n_7n_4n_8n_5n_{10}n_9)$
- 給定的 $key=(1010000010)$
 - $K_1=P8(\text{Shift}(P10(key)))=P8(\text{Shift}(1000001100))$
 $=P8(0000111000)=10100100$
 - $K_2 = P8(\text{Shift}(\text{Shift}(P10(key))))$
 $= P8(\text{Shift}(\text{Shift}(1000001100)))$
 $= P8(0001010001)$
 $= 10010010$

例題

- 已知密鑰: $K_1=10100100$, $K_2=10010010$
- 加密函數: $IP=(2\ 6\ 3\ 1\ 4\ 8\ 5\ 7)$, $P4=(2\ 4\ 3\ 1)$
- 明文(Plaintext)=11110011
 - 密文(Ciphertext)= $IP^{-1}(f_{k_2}(SW(f_{k_1}(IP(\text{plaintext}))))))$
 $= IP^{-1}(f_{k_2}(SW(f_{k_1}(10111101))))$
 - $f_{k_1}(10111101) = (L \oplus F(R, K_1), R)$
 $= ((1011) \oplus F(1101, K_1), 1101)$

計算 $F(1101, K_1)$

- 先做一個 expansion/permutation(E/P) 的運算得8位元字串 $(n_4 n_1 n_2 n_3 n_2 n_3 n_4 n_1)$ ，將1101轉成得8位元字串(11101011)。
- 計算 $(11101011) \oplus K_1 = 01001111$
- 計算 $S_0(0100)=3=(11)_2$ 與 $S_1(1111)=3=(11)_2$
- 利用4位元的置換P4將重新排列得

4- bit output $F(1101, K_1) = P_4(1111) = 1111$

- $f_{k1}(10111101) = (L \oplus F(R, K_1), R)$
 $= ((1011) \oplus (1111), 1101)$
 $= 0100 \ 1101$

例題


- $$\begin{aligned}\text{Ciphertext} &= \text{IP}^{-1}(f_{k_2}(\text{SW}(f_{k_1}(\text{IP}(\text{plaintext})))) \\ &= \text{IP}^{-1}(f_{k_2}(\text{SW}(f_{k_1}(10111101)))) \\ &= \text{IP}^{-1}(f_{k_2}(\text{SW}(0100\ 1101))) \\ &= \text{IP}^{-1}(f_{k_2}(1101\ 0100))\end{aligned}$$
- $$\begin{aligned}f_{k_2}(1101\ 0100) &= (L \oplus F(R, K_2), R) \\ &= ((1101) \oplus F(0100, K_2), 0100)\end{aligned}$$

計算 $F(0100, K_2)$

- 先做一個 expansion/permutation(E/P) 的運算得8位元字串 $(n_4 n_1 n_2 n_3 n_2 n_3 n_4 n_1)$ ，將0100轉成得8位元字串 (0010 1000)。
- 計算 $(0010\ 1000) \oplus K_2 = 0110\ 1011$
- 計算 $S_0(1011)=1=(01)_2$ 與 $S_1(1010)=0=(00)_2$
- 利用4位元的permutation P4將重新排列得到一4-bit output $F(0100, K_2) = P_4(0100) = 1000$
- $f_{k_2}(1101\ 0100) = (L \oplus F(R, K_2), R)$
 $= ((1101) \oplus F(0100, K_2), 0100)$
 $= ((1101) \oplus (1000), 0100) = 01010100$

計算 $F(0100, K_2)$

- 密文 (Ciphertext) = $IP^{-1}(f_{k_2}(SW(f_{k_1}(IP(\text{plaintext}))))))$
 $= IP^{-1}(f_{k_2}(1101\ 0100))$
 $= IP^{-1}(01010100) = 10000101$

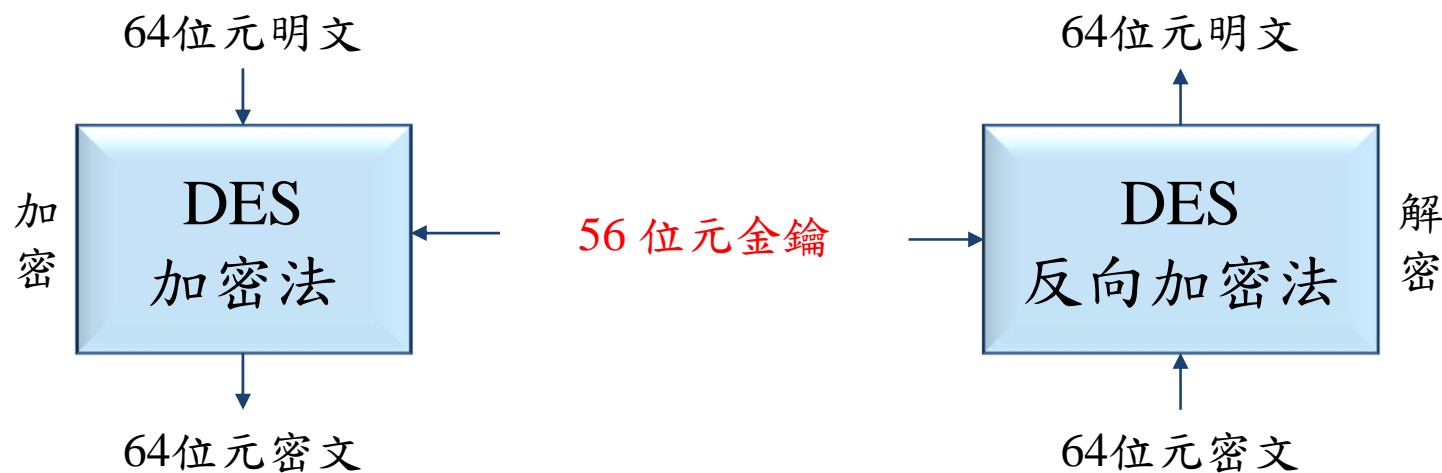
明文		密文
Plaintext		Ciphertext
11110011		10000101

3.2 資料加密標準(DES)

- 資料加密標準(Data Encryption Standard, DES)是由IBM在1970年代初期發展出來的演算法，DES使用很廣泛，特別在商業領域。
- DES使用56位元金鑰，且使用7個8位元的位元組(每個位元組的第8個位元是做為同位元檢查)做為金鑰的內容。
- DES屬於區塊式密碼(block cipher)，一次處理64位元明文。
- DES密碼共有16個循環，每個循環都使用不同的次金鑰(subkey)，且每一個金鑰都會透過自己的演算法取得16個金鑰。

3.2 資料加密標準(DES)

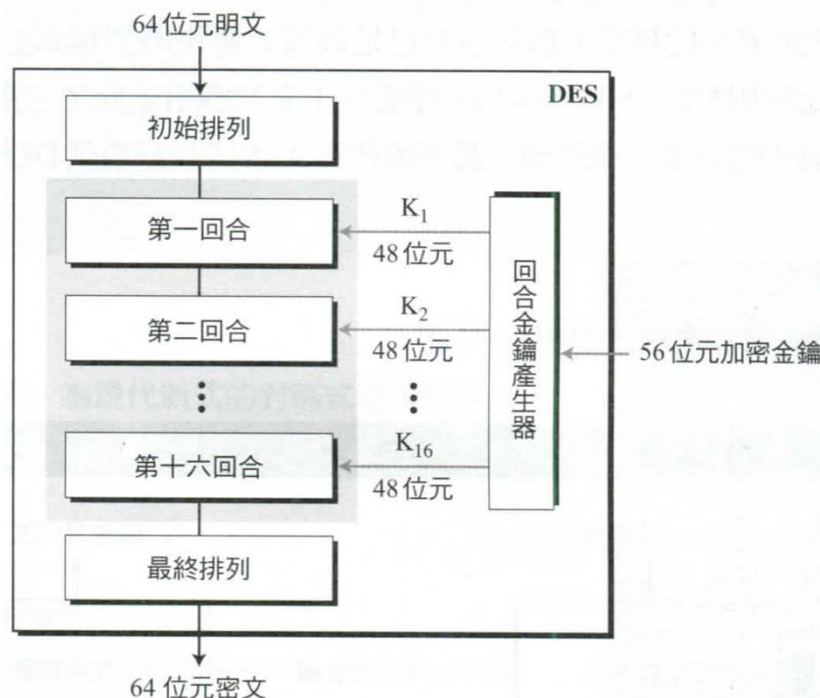
- * 使用56位元的鑰匙對64位元的區段加密。



- * 加密與解密使用相同的56位元的金鑰。

3.2 資料加密標準(DES) -初始排列與最終排列

- * DES加密程序由兩個排列(P-box，稱為初始排列與最終排列)以及十六個Feistel回合所組成。
- * 每個回合使用一把不同的48位元回合金鑰，該回合金鑰由加密金鑰透過一個預先定義的演算法所產生。
- * DES的一般結構

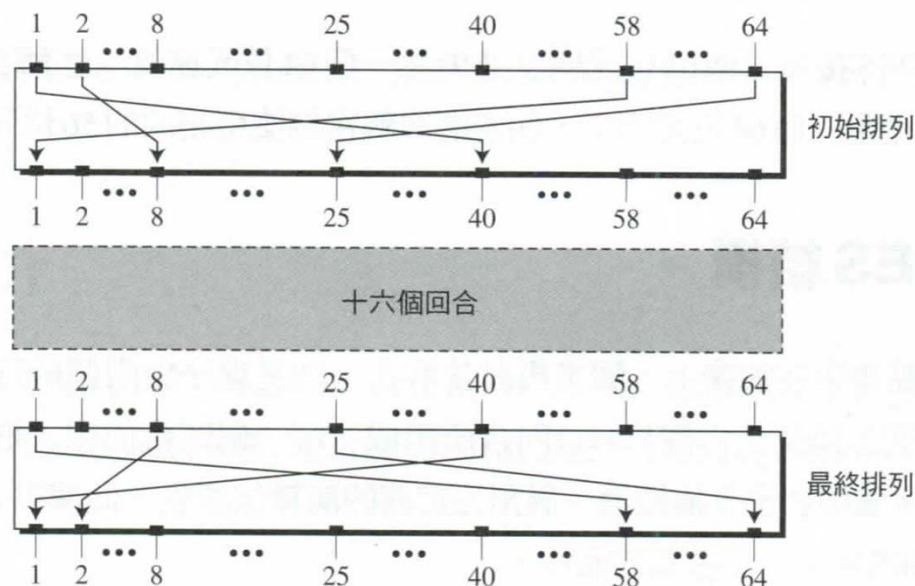


3.2 資料加密標準(DES)

-初始排列與最終排列

- * 每一個排列接受一個64位元輸入，並根據預先定義來規則排列。
- * P-box的排列為右表，表的每一個元素表示輸入為第幾位元。

DES初始排列與最終排列的步驟



初始排列與最終排列

初始排列	最終排列
58 50 42 34 26 18 10 02	40 08 48 16 56 24 64 32
60 52 44 36 28 20 12 04	39 07 47 15 55 23 63 31
62 54 46 38 30 22 14 06	38 06 46 14 54 22 62 30
64 56 48 40 32 24 16 08	37 05 45 13 53 21 61 29
57 49 41 33 25 17 09 01	36 04 44 12 52 20 60 28
59 51 43 35 27 19 11 03	35 03 43 11 51 19 59 27
61 53 45 37 29 21 13 05	34 02 42 10 50 18 58 26
63 55 47 39 31 23 15 07	33 01 41 09 49 17 57 25

3.2 資料加密標準(DES) -初始排列與最終排列

例題3.1 找出如下十六進制表示的初始排列的輸出結果：0002 0000 0000 0001

1. 首先將十六進制轉換為二進制：

000000000000000010 0000000000000000 0000000000000000 00000000000000001
123456789.....15.....64

輸入僅有兩的為1的位元(第15與第64個位元)

2. 依序填入初始排列表

初始排列							
58	50	42	34	26	18	10	02
60	52	44	36	28	20	12	04
62	54	46	38	30	22	14	06
64	56	48	40	32	24	16	08
57	49	41	33	25	17	09	01
59	51	43	35	27	19	11	03
61	53	45	37	29	21	13	05
63	55	47	39	31	23	15	07



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

3. 輸出初始排列的結果：

0000 0080 0000 0002

3.2 資料加密標準(DES) - 初始排列與最終排列

例題3.2 找出如下十六進制表示的**最終排列**的輸出結果：0000 0080 0000 0002

1. 首先將十六進制轉換為二進制：

0000000000000000 00000000**1**0000000 0000000000000000 0000000000000000**1**0
 123456789.....**25**.....**63**64

輸入僅有兩的為1的位元(第15與第64個位元)

2. 依序填入初始排列表

最終排列							
40	08	48	16	56	24	64	32
39	07	47	15	55	23	63	31
38	06	46	14	54	22	62	30
37	05	45	13	53	21	61	29
36	04	44	12	52	20	60	28
35	03	43	11	51	19	59	27
34	02	42	10	50	18	58	26
33	01	41	09	49	17	57	25



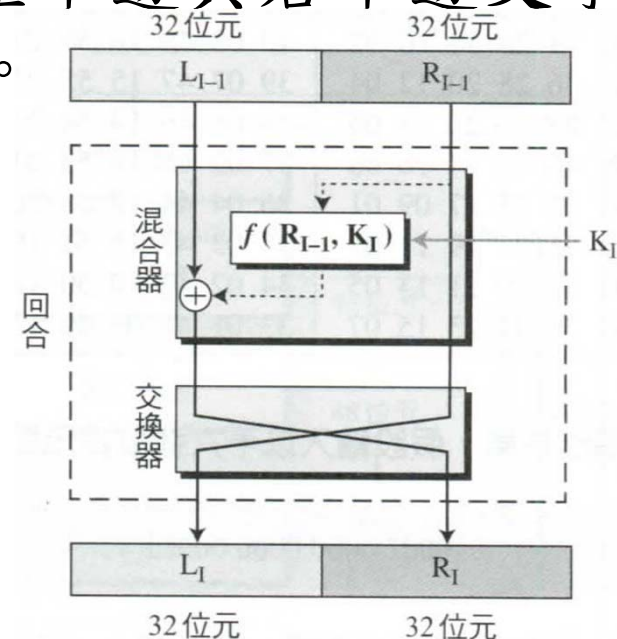
0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1

3. 輸出初始排列的結果：

000**2** 0000 0000 000**1**

3.2 資料加密標準(DES) - 回合

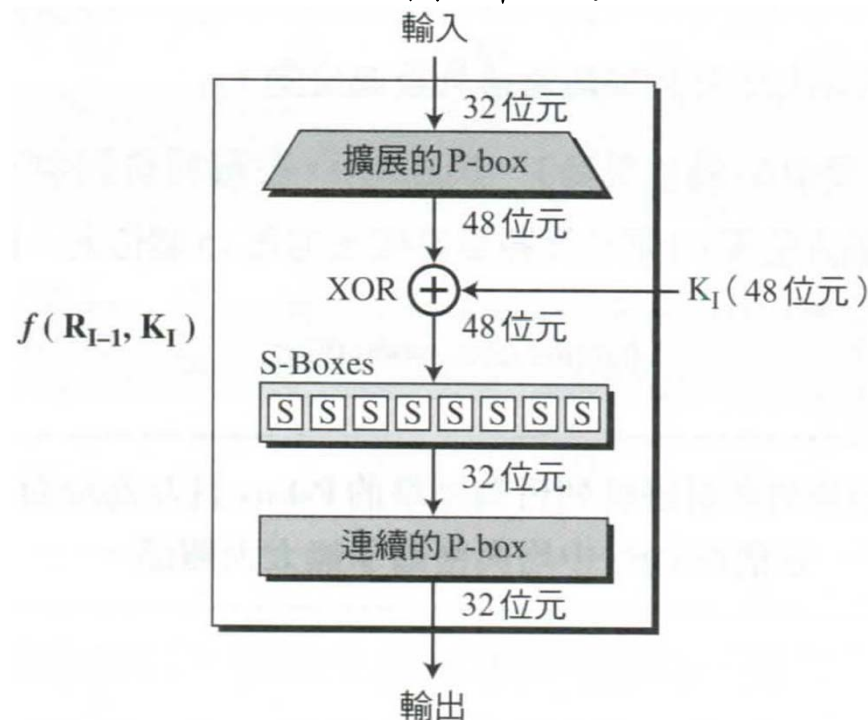
- * DES使用十六回合，每一回合是Feistel加密法，一回合中有輸入前一回合的 L_{I-1} 與 R_{I-1} (或初始P-box)並建立 L_1 及 R_1 。
- * 然後進入下一回合(或最終P-box)。
- * 交換器為可逆的，其主要為交換左半邊與右半邊文字。
- * 混合器為可逆，主要為XOR運算。



3.2 資料加密標準(DES)

-DES函數

- * DES的核心為DES函數。DES函數在最右邊的32位元(R_{I-1})再運用48位元的金鑰產生一個32位元的輸出。
- * 這個函數由四個部分組成：擴展的P-box、漂白器(負責加入金鑰)、S-box以及標準的P-box。



3.2 資料加密標準(DES)

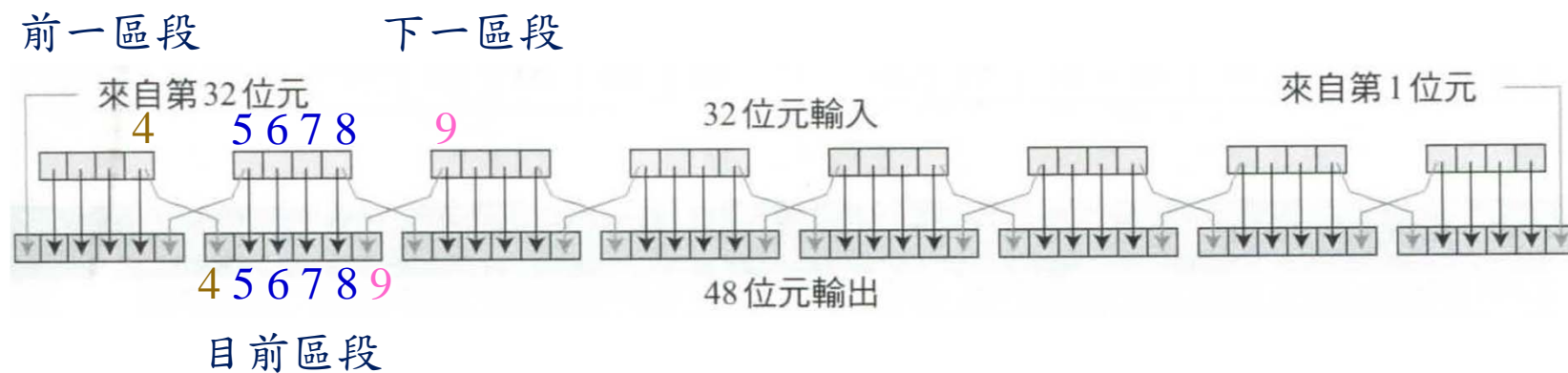
-DES函數：擴展的P-box

- * R_{I-1} 是32位元的輸入且 K_1 是一個48位元金鑰，一開始需要將 R_{I-1} 擴展為48位元。
- * 首先將 R_{I-1} 分為8個區段，每區段4位元，每區段擴展為6位元。
- * 規則為每一區段的第1、2、3及4位元複製輸出到2、3、4及5位元。
- * 輸出第1位元來自於前一區段的第4位元。
- * 輸出第6為原來自於下一區段的第1位元。

3.2 資料加密標準(DES)

-DES函數：擴展的P-box

* 擴展排列的P-box



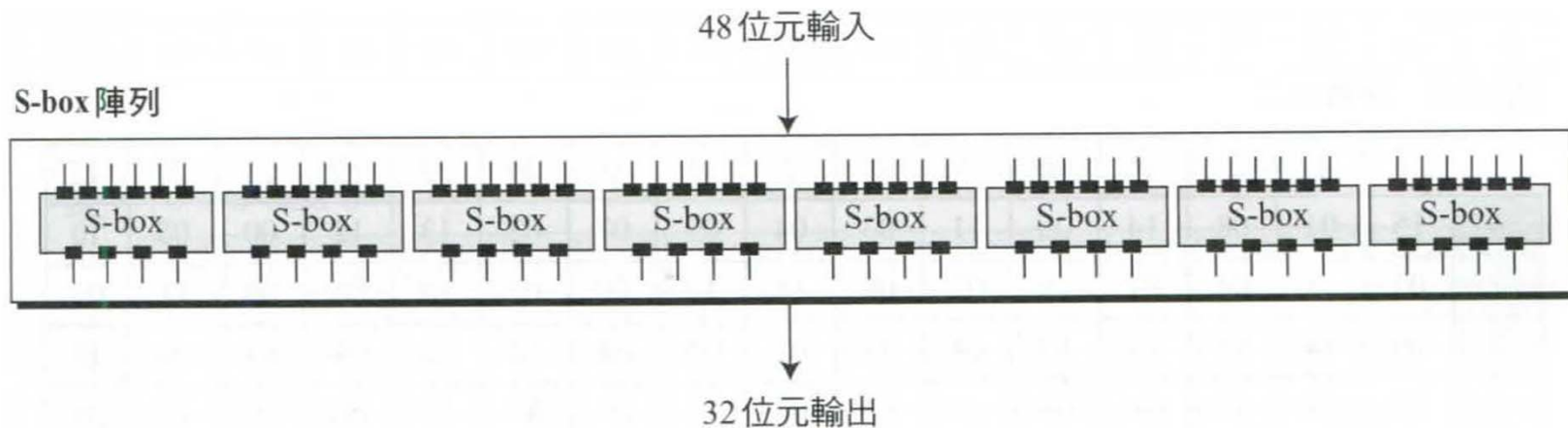
擴展的P-box表

32	01	02	03	04	05
04	05	06	07	08	09
08	09	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	31	31	32	01

3.2 資料加密標準(DES)

-DES函數：漂白器(XOR)、S-box

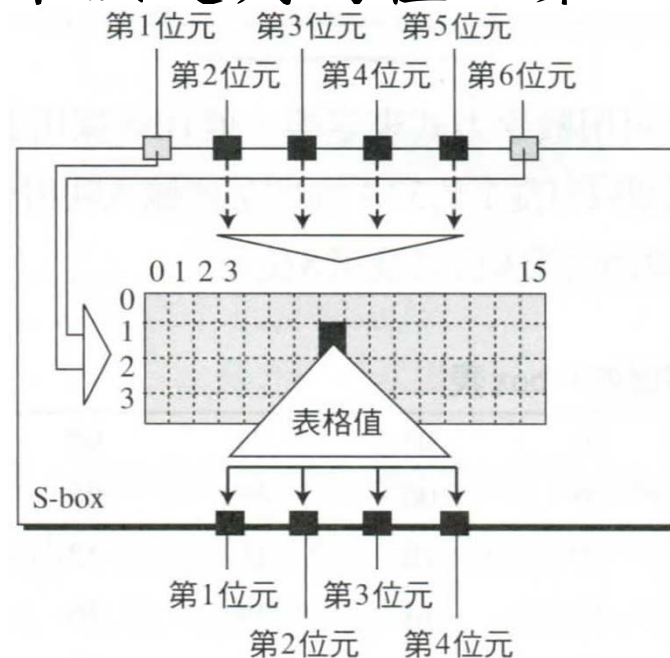
- * 漂白器(XOR)：擴展排列後，DES將擴展的右半部與回合金鑰做XOR運算，注意右半部與金鑰長度皆為48位元。
- * S-box：進行實際的混合(混淆)，DES使用8個S-box，每一個有6位元輸入及4位元輸出。



3.2 資料加密標準(DES)

-DES函數：S-box

- * 48位元的輸入分成八個6位元的區塊，每一區塊被餵入一個S-box，每一個S-box輸出結果為4位元的區塊，這些區塊結合起來為32位元。
- * 每一個S-box的取代方式為一個4列乘16行的表格；輸入的第1與6位元來決定列的值；第2~5位元決定行的值。



3.2 資料加密標準(DES)

-DES函數：S-box

- * 每一個S-box有自己的表格，有八個表格來定義這些S-box的輸出。輸入值(列數字與行數字)以及輸出值以十進位來表示，這些值需要轉換成二進制表示。

S-box 1

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	10	03	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13

S-box 2

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	15	01	08	14	06	11	03	04	09	07	02	13	12	00	05	10
1	03	13	04	07	15	02	08	14	12	00	01	10	06	09	11	05
2	00	14	07	11	10	04	13	01	05	08	12	06	09	03	02	15
3	13	08	10	01	03	15	04	02	11	06	07	12	00	05	14	09

3.2 資料加密標準(DES)

-DES函數：S-box

S-box 3

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	10	00	09	14	06	03	15	05	01	13	12	07	11	04	02	08
1	13	07	00	09	03	04	06	10	02	08	05	14	12	11	15	01
2	13	06	04	09	08	15	03	00	11	01	02	12	05	10	14	07
3	01	10	13	00	06	09	08	07	04	15	14	03	11	05	02	12

S-box 4

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	07	13	14	03	00	6	09	10	1	02	08	05	11	12	04	15
1	13	08	11	05	06	15	00	03	04	07	02	12	01	10	14	09
2	10	06	09	00	12	11	07	13	15	01	03	14	05	02	08	04
3	03	15	00	06	10	01	13	08	09	04	05	11	12	07	02	14

3.2 資料加密標準(DES)

-DES函數：S-box

S-box 5

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	02	12	04	01	07	10	11	06	08	05	03	15	13	00	14	09
1	14	11	02	12	04	07	13	01	05	00	15	10	03	09	08	06
2	04	02	01	11	10	13	07	08	15	09	12	05	06	03	00	14
3	11	08	12	07	01	14	02	13	06	15	00	09	10	04	05	03

S-box 6

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	12	01	10	15	09	02	06	08	00	13	03	04	14	07	05	11
1	10	15	04	02	07	12	09	05	06	01	13	14	00	11	03	08
2	09	14	15	05	02	08	12	03	07	00	04	10	01	13	11	06
3	04	03	02	12	09	05	15	10	11	14	01	07	10	00	08	13

3.2 資料加密標準(DES)

-DES函數：S-box

S-box 7

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	4	11	2	14	15	00	08	13	03	12	09	07	05	10	06	01
1	13	00	11	07	04	09	01	10	14	03	05	12	02	15	08	06
2	01	04	11	13	12	03	07	14	10	15	06	08	00	05	09	02
3	06	11	13	08	01	04	10	07	09	05	00	15	14	02	03	12

S-box 8

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	13	02	08	04	06	15	11	01	10	09	03	14	05	00	12	07
1	01	15	13	08	10	03	07	04	12	05	06	11	10	14	09	02
2	07	11	04	01	09	12	14	02	00	06	10	10	15	03	05	08
3	02	01	14	07	04	10	8	13	15	12	09	09	03	05	06	11

3.2 資料加密標準(DES)

-DES函數：S-box

例題3.3 若S-box 1的輸入為100010，其輸出為何？

1. 首先將第1位元與第6位元寫在一起，以二進位表示為10，以十進制表示為3，查表S-box 1表示列的2。
2. 剩下第2~5位元0001，以十進制表示為1，查表S-box 1表示行的1。
3. 第2列與第1行交叉處為十進制的01，以二進制(四位元)表示為0001。

S-box 1

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	10	03	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13

3.2 資料加密標準(DES)

-DES函數：標準的P-box

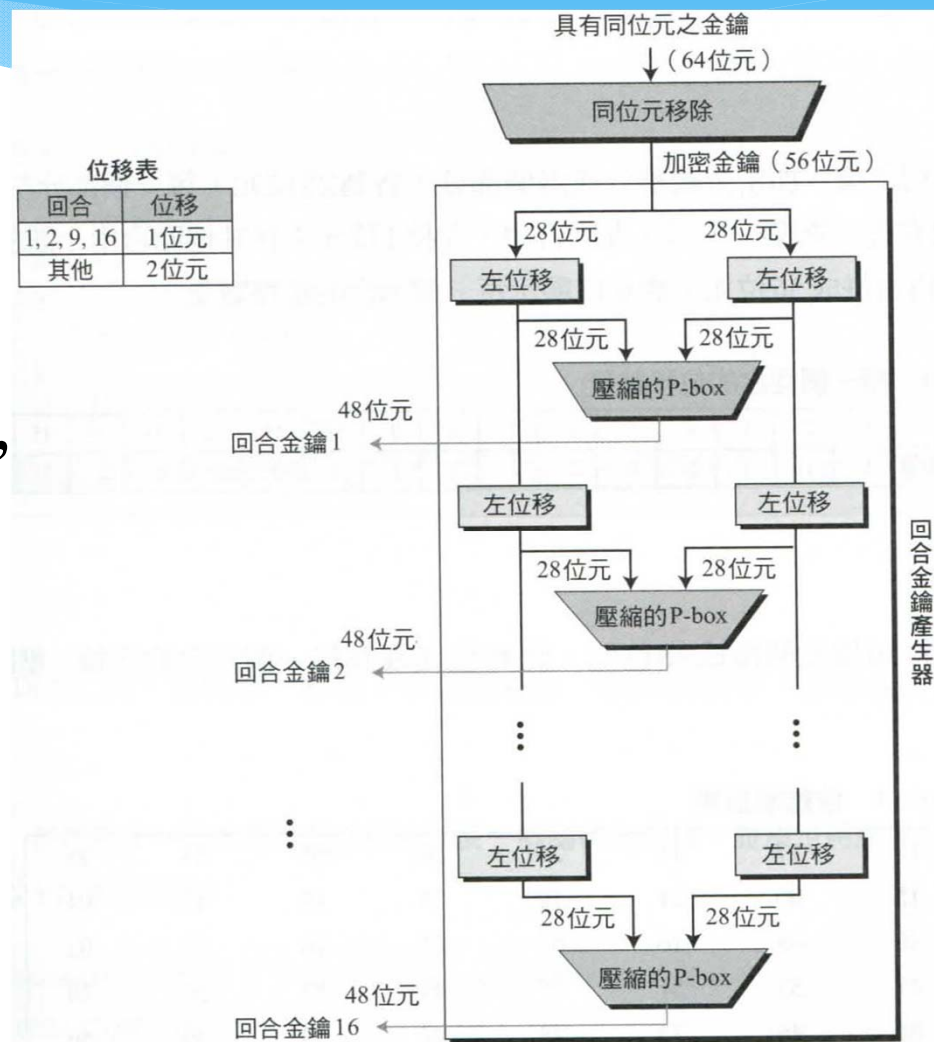
- * 標準排列(標準的P-box)：DES最後一個運算是32位元輸入及32位元輸出的標準排列，其輸入/輸出關係顯示表如下。

標準排列表

16	07	20	21	29	12	28	17
01	15	23	26	05	18	31	10
02	08	24	14	32	27	03	09
19	13	30	06	22	11	04	25

3.2 資料加密標準(DES) - 金鑰產生

- * 回合金鑰產生器(round-key generator)建立16個48位元金鑰，這些金鑰由56位元的加密金鑰而來。
- * 通常加密金鑰為64位元，其中包含額外8個位元為同位元檢查。
- * 這8位元在真正金鑰產生程序前將會被移除。



3.2 資料加密標準(DES)

- 金鑰產生：同位元移除

- * 金鑰產生的預先程序是壓縮排列，稱之為同位元移除 (parity bit drop)。
- * 從64位元金鑰中移除同位元(第8、16、24...64位元)，並根據同位元移除表排列剩下的位元。
- * 剩下的56位元金鑰為實際的加密金鑰，可用來產生回合金鑰。

同位元移除表

57	49	41	33	25	17	09	01
58	50	42	34	26	18	10	02
59	51	43	35	27	19	11	03
60	52	44	36	63	55	47	39
31	23	15	07	62	54	46	38
30	22	14	06	61	53	45	37
29	21	13	05	28	20	12	04

3.2 資料加密標準(DES)

- 金鑰產生：左移位

- * 在標準排列之後，加密金鑰被分為兩個部分各為28位元。
- * 每一部分左移(循環左移)，位移數量如下表所示。

每一回合的位移數量表

回合	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
位移數量	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

- * 此兩部分皆左移後再合併成56位元。

3.2 資料加密標準(DES)

- 金鑰產生：壓縮排列

- * 在壓縮排列將56位元轉換為48位元，此48位元成為每一回合的金鑰。

金鑰壓縮排列表

14	17	11	24	01	05	03	28
15	06	21	10	23	19	12	04
26	08	16	07	27	20	13	02
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32

3.2 資料加密標準(DES) - 回合金鑰例題

例題3.4 我們選擇一個隨機的金鑰，輸入金鑰以二進制表示成下列56bits的金鑰

0111000 0011100 1001100 1110111
0111011 1001001 1000010 1101100

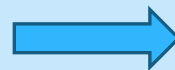
1. 加入同位元檢查後擴展成64bits

01110001 00111001 10011010 11101101
01110111 10010011 10000100 11011010

3.2 資料加密標準(DES) - 回合金鑰例題

2. 使用同位元移除表重新排列

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

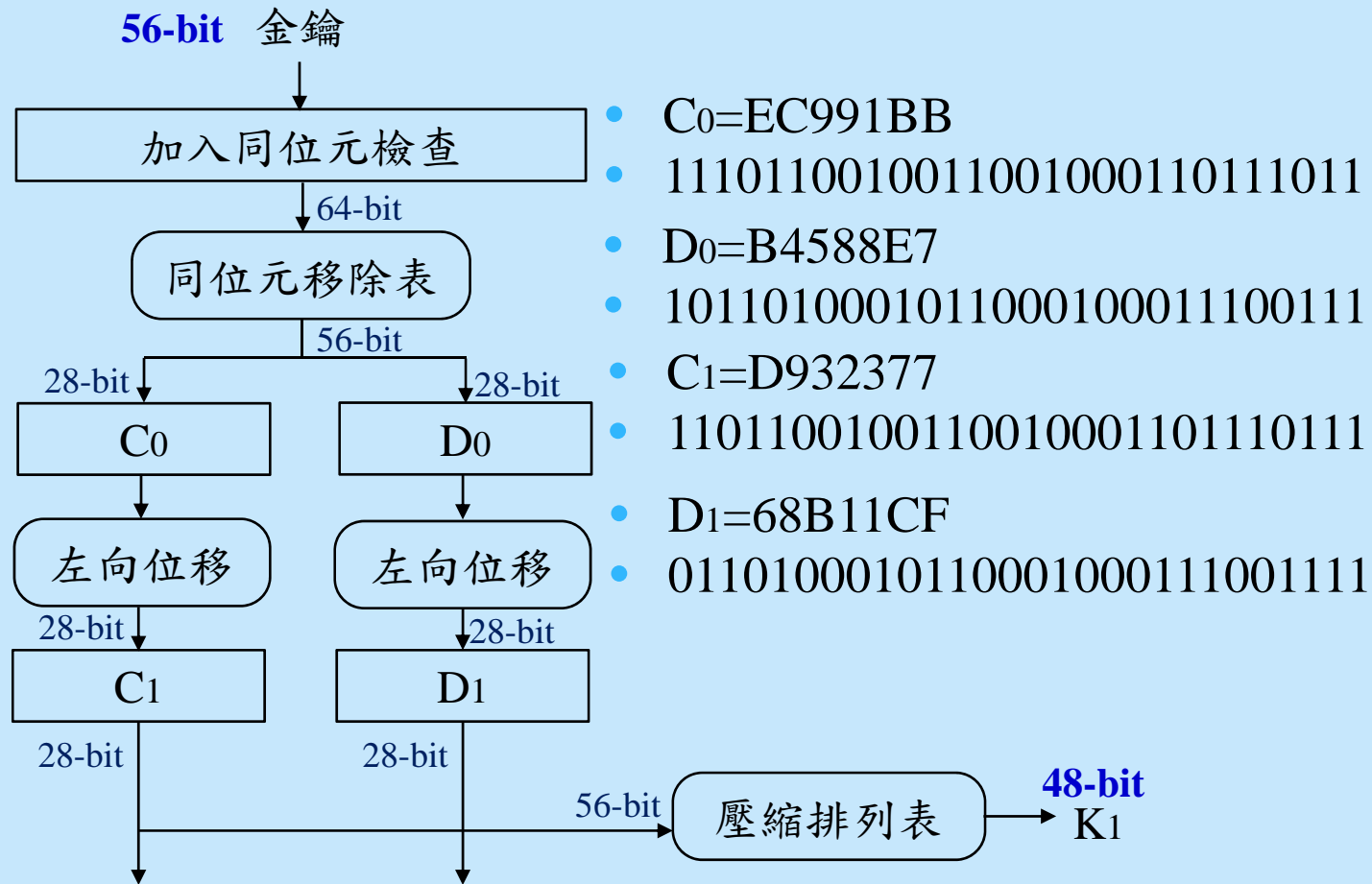


1	1	1	0	1	1	0
0	1	0	0	1	1	0
0	1	0	0	0	1	1
0	1	1	1	0	1	1
1	0	1	1	0	1	0
0	0	1	0	1	1	0
0	0	1	0	0	0	1
1	1	0	0	1	1	1

3.2 資料加密標準(DES)

-回合金鑰例題

3. 加密金鑰分為兩部分，各為28位元，且各別向左循環位移，第一回合只位移一次。



3.2 資料加密標準(DES)

- 回合金鑰例題

4. 使用壓縮排列表將56位元轉換為48位元。

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



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

5. 子金鑰 $K_1 = 3D8FCD373F48$

001111011000111111001101001101110011111101001000

3.2 資料加密標準(DES) -加密明文例題

例題3.5 我們選擇一個隨機的金鑰，輸入金鑰以二進制表示成下列56bits的金鑰

0111000 0011100 1001100 1110111
0111011 1001001 1000010 1101100

輸入明文(十六進制)如下，請使用DES加密明文
74657874626F6F6B

1. 首先將十六進制明文轉為二進制

01110100011001010111100001110100
01100010011011110110111101101011

3.2 資料加密標準(DES) -加密明文例題

2. 將明文利用初始排列表重新排列

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



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

$L_0 = \text{FF0D6BE2}$

11111111000011010110101111100010

$R_0 = \text{00FFE4F0}$

00000000111111111110010011110000

3.2 資料加密標準(DES)

-加密明文例題

3. 將 R_0 利用擴展P-box表排列，再與之前計算的金鑰 K_1 相加。

擴展P-box表					
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

→

R_0 經擴展P-box表排列					
0	0	0	0	0	0
0	0	0	0	0	1
0	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	0	0
0	0	1	0	0	1
0	1	1	1	1	0
1	0	0	0	0	0

⊕

金鑰 K_1					
0	0	1	1	1	1
0	1	1	0	0	0
1	1	1	1	1	1
0	0	1	1	0	1
0	0	1	1	0	1
1	1	0	0	1	1
1	1	1	1	0	1
0	0	1	0	0	0

3.2 資料加密標準(DES) -加密明文例題

4. 利用S-box 1~8表

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

S-box 1

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	03	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13

S-box 2

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	15	01	08	14	06	11	03	04	09	07	02	13	12	00	05	10
1	03	13	04	07	15	02	08	14	12	00	01	10	06	09	11	05
2	00	14	07	11	10	04	13	01	05	08	12	06	09	03	02	15
3	13	08	10	01	03	15	04	02	11	06	07	12	00	05	14	09

S-box 3

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	10	00	09	14	06	03	15	05	01	13	12	07	11	04	02	08
1	13	07	00	09	03	04	06	10	02	08	05	14	12	11	15	01
2	13	06	04	09	08	15	03	00	11	01	02	12	05	10	14	07
3	01	10	13	00	06	09	08	07	04	15	14	03	11	05	02	12

S-box 4

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	07	13	14	03	00	06	09	10	01	02	08	05	11	12	04	15
1	13	08	11	05	06	15	00	03	04	07	02	12	01	10	14	09
2	10	06	09	00	12	11	07	13	15	01	03	14	05	02	08	04
3	03	15	00	06	10	01	13	08	09	04	05	11	12	07	02	14

S-box 5

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	02	12	04	01	07	10	11	06	08	05	03	15	13	00	14	09
1	14	11	02	12	04	07	13	01	05	00	15	10	03	09	08	06
2	04	02	01	11	10	13	07	08	15	09	12	05	06	03	00	14
3	11	08	12	07	01	14	02	13	06	15	00	09	10	04	05	03

S-box 6

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	12	01	10	15	09	02	06	08	00	13	03	04	14	07	05	11
1	10	15	04	02	07	12	09	05	06	01	13	14	00	11	03	08
2	09	14	15	05	02	08	12	03	07	00	04	10	01	13	11	06
3	04	03	02	12	09	05	15	10	11	14	01	07	10	00	08	13

S-box 7

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	04	11	2	14	15	00	08	13	03	12	09	07	05	10	06	01
1	13	00	11	07	04	09	01	10	14	03	05	12	02	15	08	06
2	01	04	11	13	12	03	07	14	10	15	06	08	00	05	09	02
3	06	11	13	08	01	04	10	07	09	05	00	15	14	02	03	12

S-box 8

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	13	02	08	04	06	15	11	01	10	09	03	14	05	00	12	07
1	01	15	13	08	10	03	07	04	12	05	06	11	00	14	09	02
2	07	11	04	01	09	12	14	02	00	06	10	13	15	03	05	08
3	02	01	14	07	04	10	08	13	15	12	09	00	03	05	06	11

3.2 資料加密標準(DES) -加密明文例題

將S-box 1~S-box 8組合起來共32位元輸出
0001011011010001011011011011001

5. 接下來使用標準排列表(連續的P-box)

標準排列表(連續的P-box)

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



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

輸出：11011110000001010010110110011011

3.2 資料加密標準(DES) -加密明文例題

6.

輸出：11011110000001010010110110011011

\oplus

$L_0 = 11111111000011010110101111100010$

\parallel

R_1

第一回合的密文輸出：

$L_1 = R_0 = 00FFE4F0$

00000000111111111110010011110000 (32位元)

$R_1 = 21084679$

00100001000010000100011001111001 (32位元)