Lab Problems

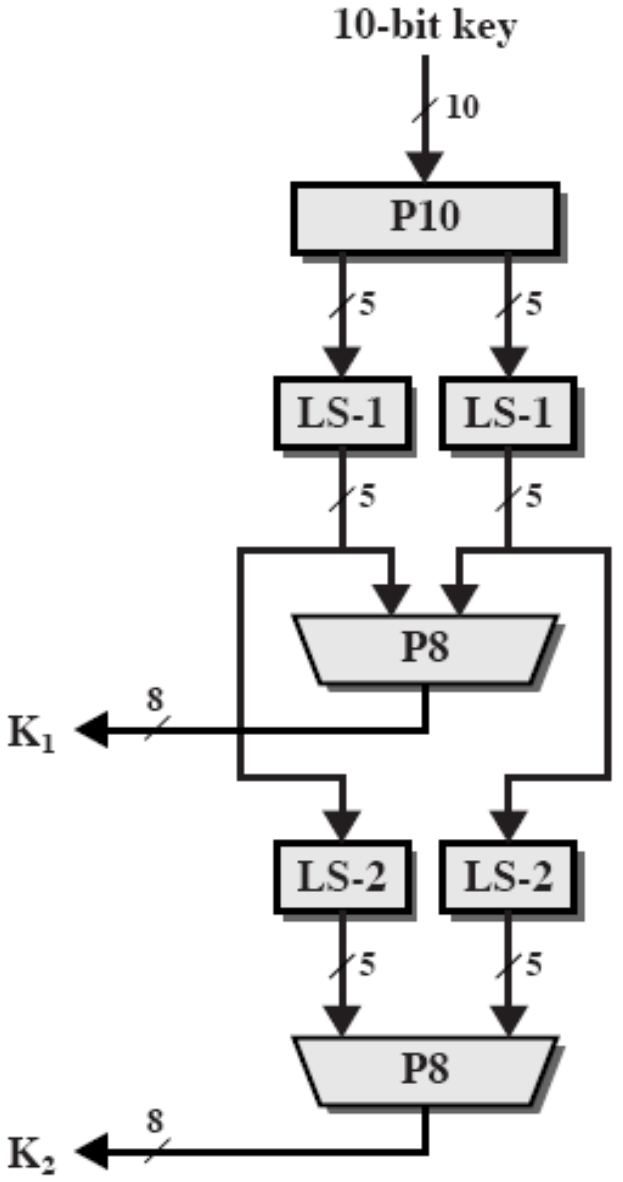
CSE-312

Lab Problem-12

Implement the Key Generation for S-DES
Input: 10 bit key
Output: K1 and K2

P_{10}									
3	5	2	7	4	10	1	9	8	6

P_8							
6	3	7	4	8	5	10	9



Lab Problem-13

Implement the S-DES Encryption

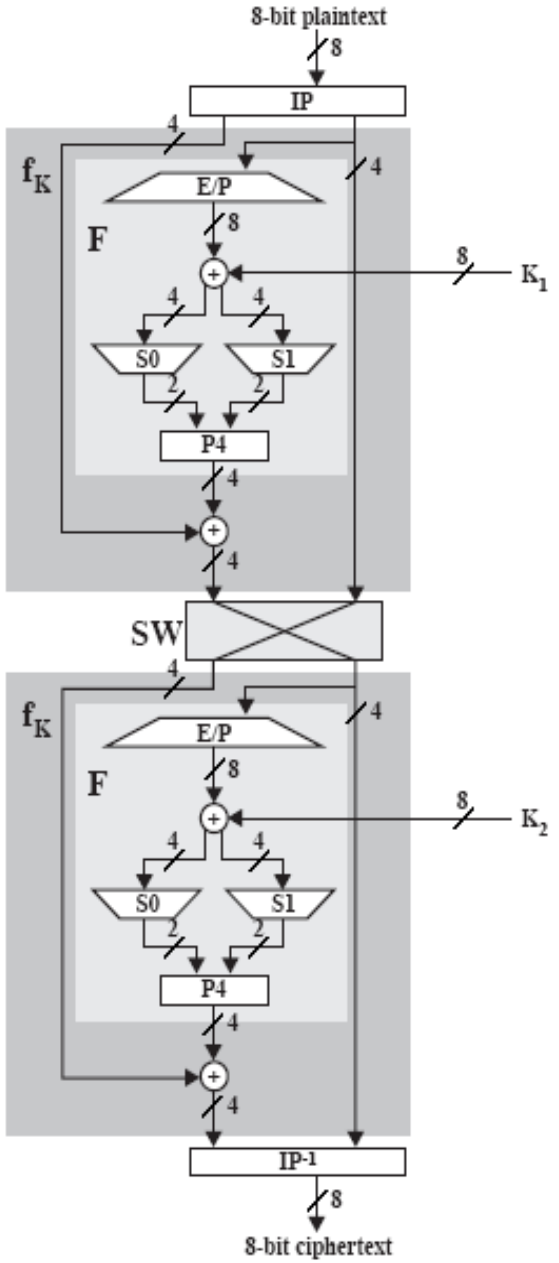
IP							
2	6	3	1	4	8	5	7

$$S0 = \begin{bmatrix} 1 & 0 & 3 & 2 \\ 3 & 2 & 1 & 0 \\ 0 & 2 & 1 & 3 \\ 3 & 1 & 3 & 2 \end{bmatrix},$$

P4 (2431)

$$S1 = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 2 & 0 & 1 & 3 \\ 3 & 0 & 1 & 0 \\ 2 & 1 & 0 & 3 \end{bmatrix}$$

IP^{-1}							
4	1	3	5	7	2	8	6



Lab Problem-14

Implement the S-DES Decryption

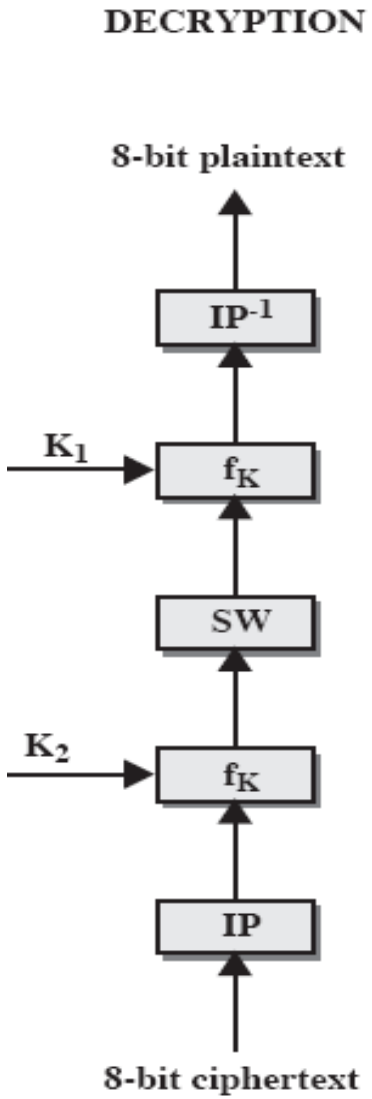
IP							
2	6	3	1	4	8	5	7

$$S0 = \begin{bmatrix} 1 & 0 & 3 & 2 \\ 3 & 2 & 1 & 0 \\ 0 & 2 & 1 & 3 \\ 3 & 1 & 3 & 2 \end{bmatrix},$$

P4 (2431)

$$S1 = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 2 & 0 & 1 & 3 \\ 3 & 0 & 1 & 0 \\ 2 & 1 & 0 & 3 \end{bmatrix}$$

IP^{-1}							
4	1	3	5	7	2	8	6



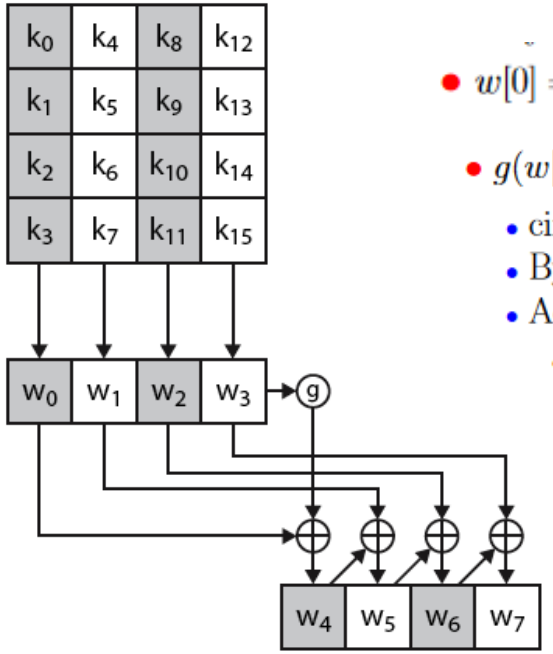
Lab Problem-15

1. Implement the Euclidean Algorithm for integers and polynomials

```
EXTENDED EUCLID(m, b)
1. (A1, A2, A3)=(1, 0, m);
   (B1, B2, B3)=(0, 1, b)
2. if B3 = 0
   return A3 = gcd(m, b); no inverse
3. if B3 = 1
   return B3 = gcd(m, b); B2 = b^-1 mod m
4. Q = A3 div B3
5. (T1, T2, T3)=(A1 - Q B1, A2 - Q B2, A3 - Q B3)
6. (A1, A2, A3)=(B1, B2, B3)
7. (B1, B2, B3)=(T1, T2, T3)
8. goto 2
```

Lab Problem-16

Implement AES Key Expansion



- $w[0] = (54, 68, 61, 74), w[1] = (73, 20, 6D, 79), w[2] = (20, 4B, 75, 6E), w[3] = (67, 20, 46, 75)$
  - $g(w[3])$:
    - circular byte left shift of $w[3]$: $(20, 46, 75, 67)$
    - Byte Substitution (S-Box): $(B7, 5A, 9D, 85)$
    - Adding round constant $(01, 00, 00, 00)$ gives: $g(w[3]) = (B6, 5A, 9D, 85)$
      - $w[4] = w[0] \oplus g(w[3]) = (E2, 32, FC, F1)$:

0101 0100	0110 1000	0110 0001	0111 0100
1011 0110	0101 1010	1001 1101	1000 0101
1110 0010	0011 0010	1111 1100	1111 0001
E2	32	FC	F1

- $w[5] = w[4] \oplus w[1] = (91, 12, 91, 88)$, $w[6] = w[5] \oplus w[2] = (B1, 59, E4, E6)$, $w[7] = w[6] \oplus w[3] = (D6, 79, A2, 93)$
- first roundkey: E2 32 FC F1 91 12 91 88 B1 59 E4 E6 D6 79 A2 93

Lab Problem-17

Implementation of RC4

```
/* Initialization */
for i = 0 to 255 do
S[i] = i;
T[i] = K[i mod keylen];
```

```
/* Initial Permutation of S */
    j = 0;
    for i = 0 to 255 do
       j = (j + S[i] + T[i]) mod 256;
       Swap (S[i], S[j]);
```

```
/* Stream Generation */
i, j = 0;
while (true)
        i = (i + 1) mod 256;
        j = (j + S[i]) mod 256;
        Swap (S[i], S[j]);
        t = (S[i] + S[j]) mod 256;
        k = S[t];
```

Lab Problem-18

Implement RSA Algorithm encryption

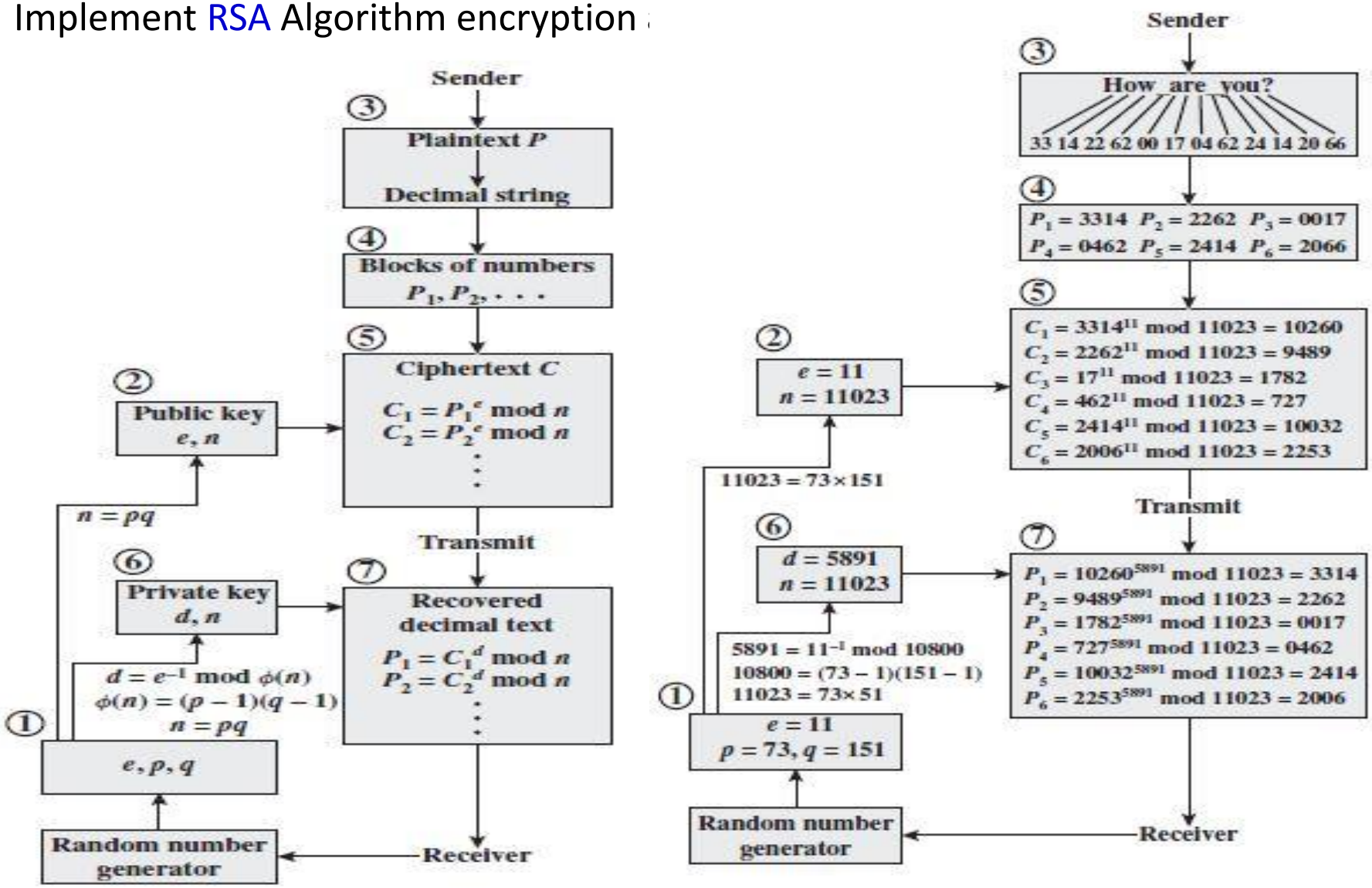


Figure 9.7 RSA Processeing of Multiple Blocks