TP 1 2 3 : Programmation en Matlab d'un crypto système à base du DES

1. Introduction

La sécurité informatique est un domaine très important dans notre vie qui protège les informations et les données dans les réseaux de communication, donc pour garantir la sécurité informatique on utilise la cryptographie. La cryptographie est l'art du secret désignant l'ensemble des techniques permettant de chiffrer des messages, c'est-à-dire permettant de les rendre inintelligibles.

La cryptographie utilise des concepts issus de nombreux domaines (informatique, mathématique, électrique). Toutes fois, les techniques évoluent et trouvent aujourd'hui régulièrement racine dans d'autres branches (biologie, physique, etc…)

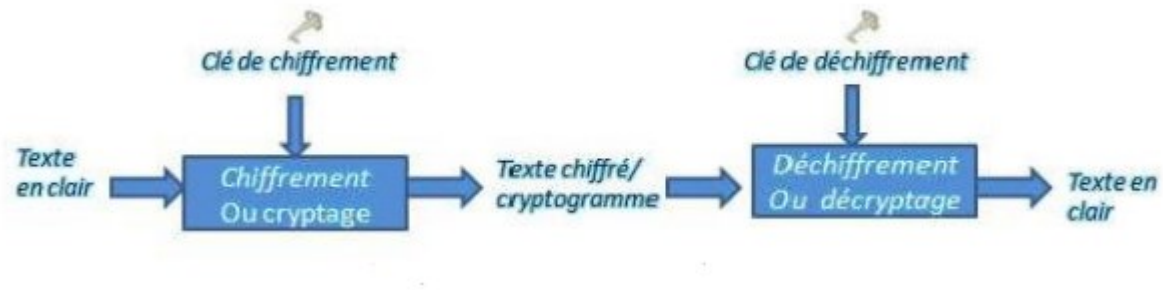

Figure1 : Principe de l'algorithme de chiffrement

2. But du TP :

Le but de ce TP est d'implémenter l'algorithme de Feistel sous Matlab. L'algorithme de Feistel représente une ronde de l'algorithme DES (Data Encryptian Standard), basée sur une succession de permutation et de substitution.

3. Préparation théorique :

- Définir les termes suivants : cryptographie, cryptographie symétrique et asymétrique.
- Quelle est la différence entre la cryptographie et le codage.
- Quelle sont les forces et faiblesses du DES.
- Les clés existantes sont-elles fortes ou non (donnez les types de clés).

4. Travail à faire en salle :

- Ecrire un script Matlab pour la génération des différentes clés du DES
- Ecrire un script Matlab pour la réalisation d'une ronde de Feistel
- Codez une suite binaire aléatoire de 64 bits. Et donnez le résultat de sortie de la première ronde de Feistel.

Principe du DES

Le D.E.S. (Data Encryptions Standard, c'est-à-dire Standard de Chiffrement de Données) est un standard mondial depuis la fin des années 1970. Au début de cette décennie, le développement des communications entre ordinateurs a nécessité la mise en place d'un standard de chiffrement de données pour limiter la prolifération d'algorithmes différents ne pouvant pas communiquer entre eux. Pour résoudre ce problème, L'Agence Nationale de Sécurité américaine (N.S.A.) a lancé des appels d'offres. La société I.B.M. a développé alors un algorithme nommé Lucifer, relativement complexe et sophistiqué. Après quelques années de discussions et de modifications, cet algorithme, devenu alors D.E.S., fut adopté au niveau fédéral le 23 novembre 1976.

A. Algorithme de chiffrement :

Le D.E.S. est un crypto système agissant par blocs. Cela signifie que D.E.S. ne chiffre pas les données à la volée quand les caractères arrivent, mais il découpe virtuellement le texte clair en blocs de 64 bits qu'il code séparément, puis qu'il concatène. Un bloc de 64 bits du texte clair entre par un coté de l'algorithme et un bloc de 64 bits de texte chiffré sort de l'autre côté. L'algorithme est assez simple puisqu'il ne combine que des permutations et des substitutions. C'est un algorithme de chiffrement à clé secrète. La clé sert donc à la fois à chiffrer et à déchiffrer le message. Cette clé a ici une longueur de 64 bits, c'est-à-dire 8 caractères, mais dont seulement 56 bits sont utilisés.

L'entière sécurité de l'algorithme repose sur les clés puisque l'algorithme est parfaitement connu de tous. La clé de 64 bits est utilisée pour générer 16 autres clés de 48 bits qu'on

utilisera lors de chacune des 16 itérations du D.E.S. Ces clés sont les mêmes quel que soit le bloc qu'on code dans un message. Cet algorithme est relativement facile à réaliser matériellement et certaines puces chiffrent jusqu'à 1 Go de données par seconde. Pour les industriels, c'est un point important notamment face à des algorithmes asymétriques, plus lents, tels que l'algorithme R.S.A.

1. La permutation initiale :

Les 64 bits du bloc d'entrée subissent la permutation de la figure (3). Ainsi avec la permutation initiale le bit se trouvant à la position 58 se retrouvera en première position et le 50ème se trouvera en deuxième position.

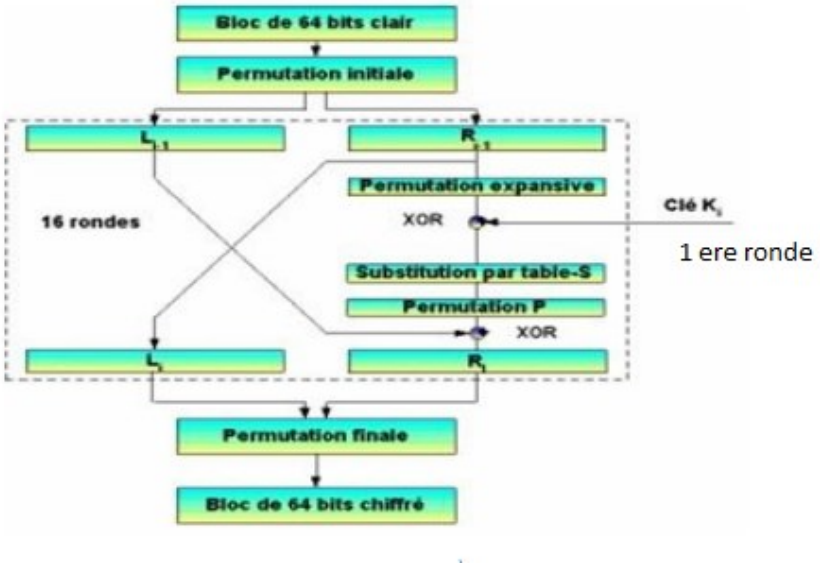

Figure 2 :Algorithme principal du DES

IP

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

Figure 3 : Permutation initiale

2. Division du message

La devisons du message en deux parties gauche et droite chacune étant sur 32bits.

3. Expansion :

Les 32 bits de droite sont étendus à 48 bits grâce à une table d'expansion (également appelée matrice d'extension). (Voir figure 4)

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

Figure 4 : Matrice d'expansion

4. Substitution par tables-S :

Cette équation va permettre à partir des 48 bits de retrouver 32 bits on a 8 table et chaque table a 6 entrées et 4 sorties. Les entrées de la table sont $b_1b_2b_3b_4b_5b_6$ les bits b_1b_6 constituent la ligne de la table, tandis que $b_2b_3b_4b_5$ constituent la colonne de la table1. (Table sur la dernière page)

La transformation S table est non linéaire donc elle confère au DES son niveau de sécurité.

5. Permutation P :

Les 32 bits sortant vont encore subir une permutation selon la matrice suivante.

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	

Figure 5 : Permutation P

6. Permutation finale :

Une fois ces calculs terminé (calcul des 16 rondes), on pratique la permutation inverse de la permutation initiale. Attention toutefois : il s'agit de l'inverse de la permutation initiale, en d'autres termes, cette table permet de retrouver la position de départ. Ce n'est pas l'inverse de la "matrice" de départ

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

Figure 5 : Permutation Finale

B. Génération de la clé :

La clé est constituée de 64 bits dont 56 sont utilisés dans l'algorithme. La clé initiale est de 64 bits. Le calcul a lieu en 4 étapes :

1. Réduction à 56 bits : les bits de parité sont enlevés. On procède ensuite à une permutation semblable à celle de la figure (6)

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**

Figure 6 : Matrice de réduction de la clé

2. Division en sous-clés de 28 bits : le résultat de l'étape précédente (56 bits) est scindé en deux sous-clés de 28bits.
3. Rotation de la clé : à chaque itération, chaque sous-clé de 28 bits subit une rotation d'1 ou 2 bits vers la gauche selon la table ci-dessous.

Ronde	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Nbre de décalage	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

4. Réduction : après concaténation des deux sous-clés précédentes, la clé résultante (56 bits) est réduite à une sous-clé de 48 bits sur base de la matrice de la figure (7).

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

Figure 7 : Matrice de réduction de la clé

C. Les S-tabes du DES

S_1

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

S_2

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

S_3

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

S_4

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

S_5

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

S_6

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

S_7

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

S_8

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