

الجمهورية الجزائرية الديمقراطية الشعبية  
République Algérienne démocratique et populaire

وزارة التعليم العالي و البحث العلمي  
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البليدة  
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا  
Faculté de Technologie

قسم الإلكترونيك  
Département d'Électronique



## Mémoire de Master

Mention Électronique

Spécialité : Télécommunications et réseaux

présenté par

HADJALA Sarra

&

HADJALA Wafaa

---

# Implémentation FPGA d'une transmission sécurisée par synchronisation chaotique adaptative

---

Proposé par : Mr. CHIKHI Mohamed Lazhar

Année Universitaire 2016-2017

## Remerciements

---

Nous remercions avant tout DIEU Allah tout puissant, qui nous a donné la volonté, la santé et la patience pour arriver au bout de nos études

Nous tenons à exprimer notre profonde gratitude et nos remerciements les plus sincères à notre encadreur Mr Lazhar Mohamed CHIKHI, par ses conseils, sa présence, sa disponibilité sa patience et ses encouragements.

Nous voudrions aussi remercier tous les membres du laboratoire LabSET qui nous ont apporté leur soutien tout au long de ce travail.

Nous exprimons également nos remerciements aux membres du jury, d'avoir accepté d'examiner et d'évaluer notre travail.

Nous exprimons également notre gratitude à tous les professeurs et enseignants qui ont collaboré à notre formation depuis notre premier cycle d'étude jusqu'à la fin de notre cycle universitaire.

Nous tenons également à remercier vivement Mr DERIOUCHE Ismail ingénieur du centre CRMET qui nous a apporté un soutien moral avec ses précieux conseils et ses encouragements.

Enfin, nous tenons à remercier profondément tous ceux qui ont contribué de près ou de loin à la réalisation du présent travail.

## *Dédicace*

Je dédie ce travail à mes chers et magnifiques parents "Kamel et Naima"  
pour leur patience et leurs encouragements qui m'ont bien soutenu pour  
arriver jusqu'ici.

A mes chers frères Abdelkader et Mohamed et sœurs Warda, Nasrine, et Sara  
qui était ma raison de honneur dans ma vie

Ainsi qu'à mon nouveau Nour

A mes meilleurs amis sans oublier mes camarades de promotion

*A tout ce qui comptent pour moi*

*HADJALA Wafaa*

# *Dédicace*

*Je dédie ce Mémoire*

*A mes très chers parents, dont l'incommensurable contribution à  
mon éducation,*

*à mon instruction et à tous les instants de ma vie, ravivera  
jusqu'à la fin de mes jours mon infinie tendresse.*

*A mes frères Mohamed et AEB, mes sœurs Warda, Nesrine et  
Wafaa pour leurs attention, leur soin et leurs encouragement  
sans oublier le nouveau Nooh.*

*A toute ma famille*

*A tous mes amis*

*HADJALA Sarra*

---

**ملخص:** تأمين تنقل المعلومات أصبح أكثر من ضروري في ظل تطور الاتصالات, و لهذا استعملت الأنظمة الفوضوية بمجال تشفير المعلومات, كونها غير قابلة للتنبؤ, جد حساسة للشروط الابتدائية و تبدو مثل ضجيج. في هذا العمل يتم تقديم نظام للإرسال الآمن للمعلومات قائم على أساس التزامن الفوضوي حلقة مغلقة بين نظامين فوضويين متمثلين ل Qi للسماح باسترجاع رسالة مشفرة, يتم عرض نتائج المحاكاة للنظام الجد فوضوي ونظام البعث الفوضوي, كما يتم عرض النتائج التجريبية لتطبيق النظام الجد فوضوي في دارة FPGA.

**كلمات المفاتيح :** إرسال فوضوي؛ التزامن الفوضوي؛ حلقة مغلقة أنظمة فوضوية؛ نظام جد فوضوي ل Qi؛ تشفير.

---

**Résumé :** La sécurisation de l'information à transmettre devient nécessaire avec l'évolution des communications. Pour cela, les systèmes chaotiques sont très utilisés dans le cryptage de données, car ils sont d'aspect imprévisible, très sensibles aux conditions initiales et présentent une forte ressemblance avec un bruit. Notre travail décrit ainsi un système de transmission sécurisée à base de synchronisation chaotique adaptative entre deux systèmes hyper-chaotique identiques de Qi, permettant la restitution d'un message crypté. Les résultats de simulation du système hyper-chaotique de Qi et du système de transmission chaotique sont présentés ainsi que les résultats expérimentaux de l'implémentation du système de Qi sur circuit FPGA.

**Mots clés :** systèmes chaotiques; système hyper-chaotique de Qi; transmission chaotique; synchronisation adaptative chaotique; cryptage; implémentation FPGA.

---

**Abstract :** The security of the information to be transmitted becomes necessary with the evolution of the communications. For this purpose, chaotic systems are widely used in data encryption, because they are unpredictable in appearance, very sensitive to initial conditions and have a strong resemblance to noise. Our work thus describes a secure transmission system based on adaptive chaotic synchronization between two identical hyper-chaotic systems of Qi, allowing the reproduction of an encrypted message. The results of the simulation of the hyper-chaotic system of Qi and the chaotic transmission system are presented as well as some experimental results of the implementation of the system of Qi on FPGA circuit.

**Keywords :** systems chaotic; systems hyper-chaotic of Qi; transmission chaotic; adaptive chaotic synchronization; encryption; implementation FPGA.

---

## Listes des acronymes et abréviations

$\frac{dx}{dt} = \dot{x}$  : dérivée de la variable x par rapport au temps.

$R^n$ : espace de phase de système dynamique.

$R^r$ : espace des paramètres de système dynamique.

$x_k$ : état de X au temps  $t=kT$ .

$x_{k+1}$ : état de X au temps  $t=(k+1)T$ .

p: vecteur d'entrée du système.

$x^*$ : point fixe.

P: point fixe des coordonnées (0 0 0 0).

$\lambda_i$ : exposant de Lyapunov.

FPGA: Field Programmable Gate Array.

CLB: Configurable Logic Block.

IOB: Input output bloc

SRAM: Static Random Access Memory

EPROM: Erasable Programmable Read-Only Memory

EEPROM: Electrically Erasable Programmable Read-Only Memory

HDL: Hardware Description Language.

VHDL: Very High speed integrated Hardware Description Language.

CAO: Conception Assistée par Ordinateur.

ISE: Integrated Software Environment.

AC '97: Audio Codec 97.

CAN: Convertisseur Analogique Numérique.

CNA: Convertisseur Numérique Analogique.

UCF: User Constraint File

# Table des matières

## Chapitre 1 : Généralités sur les systèmes dynamiques chaotiques

1.1	Introduction.....	3
1.2	Définition du système dynamique.....	3
1.2.1	Systèmes autonomes et non-autonomes.....	4
1.3	Système chaotique.....	4
1.3.1	Définition d'un système chaotique.....	4
1.3.2	Les différents types d'attracteur.....	5
1.4	Point fixe.....	7
1.5	Stabilité des points fixes.....	7
1.6	Exposants de Lyapunov.....	8
1.7	L'espace de phase.....	8
1.8	La section de Poincaré.....	9
1.9	La bifurcation.....	9
1.9.1	Types de bifurcation.....	10
1.9.2	Diagramme de bifurcation.....	11
1.10	Exemple d'attracteur.....	12
1.11	Conclusion.....	13

## Chapitre 2 : Analyse du système hyper chaotique de Qi

2.1	Introduction.....	14
2.2	Description du système.....	14
2.3	Analyse du système hyper-chaotique de Qi.....	15
2.3.1	Etude des points fixes.....	15
2.3.2	Evolution du système de Qi en fonction de temps.....	16
2.3.3	Plan de phase.....	19
2.3.4	Attracteur étrange.....	21
2.3.5	Exposants de Lyapunov.....	22
2.3.6	Section de Poincaré.....	24
2.3.7	Diagramme de bifurcation.....	24
2.4	Spectre en fréquence.....	27
2.5	Conclusion.....	28

## **Chapitre 3 : Communication sécurisée par synchronisation chaotique adaptative**

3.1	Introduction.....	29
3.2	Les classes de synchronisation.....	30
3.2.1	Synchronisation unidirectionnelle.....	30
3.2.2	Synchronisation bidirectionnelle.....	30
3.3	Les méthodes de synchronisation.....	31
3.3.1	Synchronisation par boucle fermée.....	31
3.3.2	Synchronisation impulsive.....	31
3.3.3	Synchronisation adaptative.....	32
3.4	Techniques de cryptage par chaos.....	32
3.4.1	Cryptage par addition.....	32
3.4.2	Cryptage par commutation (CSK).....	33
3.4.3	Cryptage mixte.....	33
3.4.4	Cryptage par modulation paramétrique.....	34
3.5	Etude de l'émetteur-récepteur chaotique.....	34
3.6	Résultat de la simulation.....	37
3.7	Perte de la synchronisation.....	45
3.8	Conclusion.....	47

## **Chapitre 4 : Implémentation FPGA du système chaotique de Qi**

4.1	Introduction.....	48
4.2	Présentation des circuits FPGA.....	48
4.2.1	Architecture des FPGA.....	49
4.2.2	Technologies de programmation.....	49
4.2.3	Critères de choix de la carte FPGA.....	50
4.3	Processus d'implémentation.....	50
4.4	Présentation des outils logiciels de travail.....	52
4.4.1	Présentation du logiciel ISE.....	52
4.4.2	Présentation du Co-simulateur System Generator.....	52
4.4.3	Présentation de ModelSim de Mentor Graphics.....	53
4.5	Réalisation expérimentale de l'implémentation.....	53



4.5.1	Plate forme de développement ML501-Virtex 5.....	53
4.5.2	Codec AC97.....	55
4.6	Implémentation du système hyper chaotique du Qi sur FPGA.....	56
4.7	Visualisation des signaux.....	59
4.8	Conclusion.....	62

## Liste des figures

<b>Figure 1.1.</b> Les différents types d'attracteurs.....	6
<b>Figure 1.2.</b> Les types de stabilité de point d'équilibre.....	7
<b>Figure1.3.</b> Section de Poincaré.....	9
<b>Figure 1.4.</b> Les différents types de bifurcations.....	10
<b>Figure1.5.</b> Diagramme de bifurcation de Rössler.....	11
<b>Figure1.6.</b> Attracteur de Lorenz ( $a=10, b=28, c=8/3$ ).....	12
<b>Figure 1.7.</b> Attracteur de Rössler ( $a=0.2, b=0.2, r=5$ ).....	13
<b>Figure 2.1.</b> Représentation du système de Qi sous MATLAB Simulink.....	16
<b>Figure 2.2.</b> L'état $x_1$ en fonction du temps t.....	17
<b>Figure 2.3.</b> L'état $x_2$ en fonction du temps t.....	17
<b>Figure 2.4.</b> L'état $x_3$ en fonction du temps t.....	18
<b>Figure 2.5.</b> L'état $x_4$ en fonction du temps t.....	18
<b>Figure 2.6.</b> Les états $x_1, x_2, x_3, x_4$ en fonction du temps t.....	19
<b>Figure 2.7.</b> Plan de phase $x_2$ en fonction de $x_1$ .....	19
<b>Figure 2.8.</b> Plan de phase $x_3$ en fonction de $x_1$ .....	20
<b>Figure 2.9.</b> Plan de phase $x_4$ en fonction de $x_3$ .....	20
<b>Figure 2.10.</b> Attracteur étrange $x_1, x_2$ et $x_3$ .....	21
<b>Figure 2.11.</b> Attracteur étrange $x_1, x_2$ et $x_4$ .....	21
<b>Figure 2.12.</b> Attracteur étrange $x_2, x_3$ et $x_4$ .....	22
<b>Figure 2.13.</b> L'interface de l'outil MATDS.....	22
<b>Figure 2.14.</b> Création du système de Qi en MATDS.....	23
<b>Figure 2.15.</b> Exposant de Lyapunov du système de Qi.....	23
<b>Figure 2.16.</b> La section de Poincaré.....	24
<b>Figure 2.17.</b> Diagramme de bifurcation de Qi.....	25
<b>Figure 2.18.</b> Attracteur périodique $x_2, x_3$ et $x_4$ lorsque $b= 0.1$ .....	25
<b>Figure 2.19.</b> Les états $x_1, x_2, x_3$ et $x_4$ en fonction du temps lorsque $b=0.1$ .....	26
<b>Figure 2.20.</b> Attracteur doublement de période $x_2, x_3$ et $x_4$ lorsque $b=1.5$ .....	26
<b>Figure 2.21.</b> Les états $x_1, x_2, x_3$ et $x_4$ en fonction du temps lorsque $b=1.5$ .....	27
<b>Figure 2.22.</b> les spectres de fréquence.....	27
<b>Figure 3.1.</b> Diagramme principal de la communication sécurisée.....	29

<b>Figure 3.2.</b> Synchronisation unidirectionnelle.....	30
<b>Figure 3.3.</b> Synchronisation bidirectionnelle.....	31
<b>Figure 3.4.</b> Synchronisation par boucle fermée.....	31
<b>Figure 3.5.</b> Synchronisation impulsive.....	32
<b>Figure 3.6.</b> Principe du cryptage par addition.....	33
<b>Figure 3.7.</b> Principe du cryptage par commutation.....	33
<b>Figure 3.8.</b> Principe du cryptage mixte.....	34
<b>Figure 3.9.</b> Principe du cryptage par modulation.....	34
<b>Figure 3.10.</b> Schéma synoptique d'une transmission sécurisée.....	35
<b>Figure 3.11.</b> Synchronisation des signaux $x_1$ et $y_1$ .....	38
<b>Figure 3.12.</b> Synchronisation des signaux $x_2$ et $y_2$ .....	38
<b>Figure 3.13.</b> Synchronisation des signaux $x_3$ et $y_3$ .....	38
<b>Figure 3.14.</b> Synchronisation des signaux $x_4$ et $y_4$ .....	39
<b>Figure 3.15.</b> Les erreurs de synchronisation $e_1, e_2, e_3$ et $e_4$ .....	39
<b>Figure 3.16.</b> Schéma de transmission d'une sinusoïde sous MATLAB.....	40
<b>Figure 3.17.</b> Message émis, crypté et reçu sans bruit.....	41
<b>Figure 3.18.</b> Message émis et reçu avec bruit ( $B = 1$ ).....	41
<b>Figure 3.19.</b> Message émis et reçu avec bruit ( $B = 10$ ).....	41
<b>Figure 3.20.</b> Message émis et reçu avec bruit ( $B=100$ ).....	42
<b>Figure 3.21.</b> Schéma de transmission d'une image sous MATLAB.....	43
<b>Figure 3.22.</b> Récupération de l'image sans bruit.....	44
<b>Figure 3.23.</b> Récupération de l'image avec bruit ( $B=0.000001$ ).....	44
<b>Figure 3.24.</b> Récupération de l'image avec bruit ( $B=0.0001$ ).....	44
<b>Figure 3.25.</b> Désynchronisation des signaux $x_1$ et $y_1$ .....	45
<b>Figure 3.26.</b> Désynchronisation des signaux $x_2$ et $y_2$ .....	45
<b>Figure 3.27.</b> Désynchronisation des signaux $x_3$ et $y_3$ .....	46
<b>Figure 3.28.</b> Désynchronisation des signaux $x_4$ et $y_4$ .....	46
<b>Figure 3.29.</b> Les erreurs de désynchronisation $e_1, e_2, e_3$ et $e_4$ .....	47
<b>Figure 4.1.</b> Architecture générique d'un circuit FPGA.....	49
<b>Figure 4.2.</b> Critères de choix de la carte FPGA.....	50
<b>Figure 4.3.</b> Etapes de conception sur FPGA.....	51
<b>Figure 4.4.</b> Interface Project Navigator ISE 14.2.....	52

<b>Figure 4.5.</b> Interface ModelSim de Mentor Graphics.....	53
<b>Figure 4.6.</b> La carte ML501 Virtex-5 (Vue de dessus).....	54
<b>Figure 4.7.</b> La carte ML501 Virtex-5 (Vue d'en dessous).....	55
<b>Figure 4.8.</b> Connexions du contrôleur au CODEC AC97.....	55
<b>Figure 4.9.</b> Contrôleur AC97 sous System Generator.....	56
<b>Figure 4.10.</b> Le système hyper chaotique du Qi sous System Generator.....	57
<b>Figure 4.11.</b> Les étapes de la conception.....	58
<b>Figure 4.12.</b> Dernière phase de programmation sur la carte FPGA.....	58
<b>Figure 4.13.</b> Dispositif expérimental de l'implémentation FPGA.....	59
<b>Figure 4.14.</b> Les signaux de simulation sous Modelsim.....	59
<b>Figure 4.15.</b> Les signaux $x_1, x_2$ en fonction du temps.....	60
<b>Figure 4.16.</b> Plan de phase $x_1$ en fonction de $x_2$ .....	60
<b>Figure 4.17.</b> Les signaux $x_1, x_4$ en fonction du temps.....	61
<b>Figure 4.18.</b> Plan de phase $x_1$ en fonction de $x_4$ .....	61
<b>Figure 4.19.</b> Ressources consommées lors de l'implémentation.....	62

## Liste des tableaux

<b>Tableau 1.1.</b> Caractérisation des attracteurs.....	8
--	---

# Introduction générale

---

Dans le domaine des télécommunications, où les échanges d'informations se développent rapidement, il est indispensable de pouvoir disposer de système sécurisé pour protéger les données et garantir la confidentialité. Il est donc nécessaire de développer un outil efficace de protection des données transférées et des communications. Le cryptage des données est un moyen efficace pour répondre à ces exigences [1].

Les systèmes dynamiques chaotiques sont des systèmes déterministes non linéaires, apériodique et borné. Les signaux qui évoluent dans ces systèmes sont en général à large bande et très sensibles aux conditions initiales, ce qui fait apparaître leur trajectoire comme un bruit pseudo aléatoire. En raison de ces propriétés, les signaux chaotiques sont de plus en plus utilisés dans les systèmes de communications [7], permettant ainsi de transférer les informations entre deux systèmes sans altérer le contenu. Afin d'augmenter le degré de sécurité, on utilise la cryptographie chaotique qui consiste à noyer l'information dans un signal chaotique.

Le chaos est né en 1890 par la découverte de Henri Poincaré démontrant que l'orbite de trois corps céleste agissante peut engendrer un comportement instable et imprévisible (le chaos est né mais pas encore mentionné)[4]. Plus tard, en 1963 Edward Lorenz découvrit le premier système chaotique par l'effet de papion [16] qui est souvent invoqué pour faire allusion à des petites causes pouvant avoir de grand effet.

L'usage du chaos pour la sécurisation de la télécommunication pose directement le problème de synchronisation du récepteur afin de suivre le signal chaotique employé à

l'émetteur [7], car l'aspect pseudo aléatoire du chaos nous amène à penser qu'il est impossible de le synchroniser.

Ce n'est qu'en 1990 que les deux chercheurs Pecora et Carroll ont montré que deux systèmes chaotiques identiques peuvent se synchroniser [4]. Les travaux de Pecora et Carroll ont permis de suggérer que les systèmes chaotiques pourraient être utilisés dans la communication, où leur nature semblable aux bruits améliorerait la sécurité [4]. En effet, une fois la synchronisation entre l'émetteur et le récepteur atteinte, il est possible de récupérer un message masqué par l'émetteur chaotique.

L'usage de chaos pour sécuriser la transmission des données fait l'objet de ce mémoire, qui repose sur la synchronisation adaptative et le cryptage paramétriques entre deux systèmes hyper-chaotiques de Qi.

Ce mémoire se compose de 4 chapitres à savoir :

Après une introduction générale, le chapitre 1 présente des définitions sur les systèmes dynamiques non linéaires notamment les systèmes chaotiques .

Le chapitre 2 est consacré à l'étude du système hyper-chaotique de Qi, qui sera utilisé pour la conception de l'émetteur-récepteur chaotique .

Le chapitre 3 décrit le système de communication sécurisée par la synchronisation adaptative et le cryptage paramétrique.

Le chapitre 4 présente l'implémentation du système hyperchaotique de Qi sur circuit FPGA et la présentation des résultats expérimentaux obtenus.

Enfin, nous terminons le mémoire par une conclusion générale et les perspectives futures de ce travail.

# Chapitre 1 Généralités sur les systèmes dynamiques chaotiques

---

## 1.1 Introduction

Le but principal de ce chapitre est de présenter quelques définitions liées aux systèmes dynamiques non linéaires et en particulier aux systèmes dynamiques chaotiques, ainsi que leurs principales caractéristiques et propriétés.

**Henri Poincaré** avait déjà mis en évidence le phénomène de sensibilité aux conditions initiales lors de l'étude astronomique du problème des trois corps [5]. Le mathématicien **Alexandre Lyapunov** effectua des recherches sur la stabilité du mouvement, et **Edward Lorenz** en 1963 découvrit le phénomène de sensibilité aux conditions initiales [4], principale caractéristique des systèmes chaotiques, c'est à dire de petites différences dans les conditions initiales engendraient à long terme une évolution des systèmes totalement différents.

## 1.2 Définition du système dynamique

Un système dynamique est une structure qui évolue au cours du temps de deux façons à la fois:

- Causale: ne dépend que des phénomènes du passé ou du présent
- Déterministe: une "condition initiale" donnée à l'instant " présent" va correspondre à un seul état « futur » possible.

L'évolution déterministe alors se modélise de deux façons distinctes [1] [2]

- Une évolution continue dans le temps, représentée par une équation différentielle ordinaire.

$$\frac{dx}{dt} = \dot{x} = f(x, t, p) \quad \text{où } x \in R^n \text{ et } p \in R^r \quad (1.1)$$



- Une évolution discrète dans le temps, l'étude théorique de ces modèles discrets est fondamentale (fonction itérative)

$$x_{k+1} = f(x_k, p), x_k \in R^n \text{ et } p \in R^r, k = 1, 2, 3, \dots \quad (1.2)$$

Où :  $R^n$  l'espace de phase de système dynamique.

$R^r$  l'espace des paramètres de système dynamique.

### 1.2.1 Systèmes autonomes et non-autonomes

Un système dynamique non-linéaire est dit autonome lorsqu'il ne dépend pas explicitement du temps; tout instant peut être alors considéré comme instant initial [3], sinon le système est non autonome.

$$\dot{x} = f(x, t) \quad (1.3)$$

- Pour un système autonome en temps continu, au moins 3 variables d'état sont nécessaires pour générer le chaos
- Pour un système non-autonome, il faut au moins deux variables d'état et une entrée indépendante.

**Remarque** : Par un changement de variable approprié, on peut toujours transformer un système dynamique non autonome de dimension  $n$  en un système dynamique autonome équivalent de dimension  $n + 1$ [8].

## 1.3 Système chaotique

### 1.3.1 Définition d'un système chaotique

Le terme chaos définit un état particulier d'un système dont le comportement ne se répète jamais qui est très sensible aux conditions initiales, et imprédictible à long terme.

Les systèmes chaotiques sont des systèmes dynamiques qui évoluent dans une région bornée, qui possèdent une infinité de trajectoire non périodique. Ils sont très sensibles aux conditions initiales. Leurs principales propriétés sont :

\* **La non-linéarité**: Un système chaotique est un système dynamique non linéaire. Un système linéaire ne peut pas être chaotique. La notion de système dynamique est relative à tous les systèmes dont l'évolution dépend du temps[1].

\* **Le déterminisme:** Un système chaotique a des règles fondamentales déterministes et non probabilistes. Il est généralement régi par des équations différentielles non linéaires qui sont connues [1].

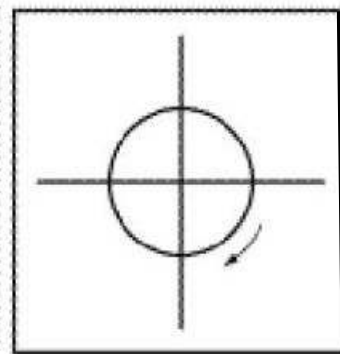
\* **imprévisible:** Il est impossible de prévoir une évolution à long terme du fait que l'état final dépend de manière si sensible de l'état initial.

\* **Sensibilité aux conditions initiales:** Cette propriété a été découverte par Edward Lorenz, c'est une explication scientifique de l'effet de papillon, démontrant que dans un système non linéaire, une modification infime de condition initiale peut entraîner des résultats imprévisibles sur le long terme [4].

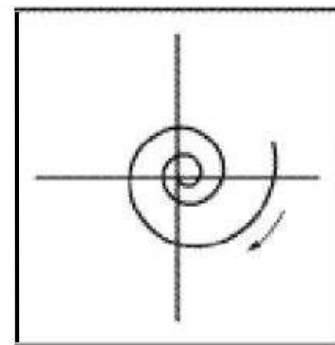
Un petit écart entre deux conditions initiales conduit à une divergence rapide des trajectoires au cours de temps.

### 1.3.2 Les différents types d'attracteur

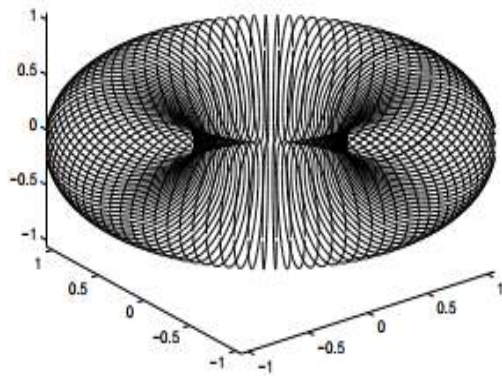
La région de l'espace de phases vers laquelle convergent les trajectoires d'un système dynamique dissipatif s'appelle "attracteur". Les attracteurs sont des formes géométriques qui caractérisent l'évolution à long terme des systèmes dynamiques. Il en existe quatre types distincts



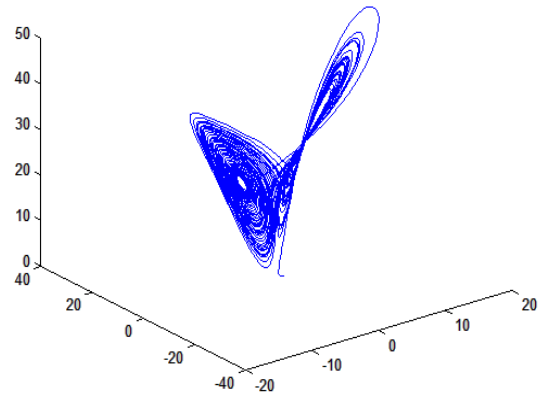
**a.** Attracteur Cycle limite.



**b.** Attracteur point fixe.



**c.** Attracteur Tore.



**d.** Attracteur étrange.

**Figure 1.1.** Les différents types d'attracteurs.

\* **L'attracteur "point fixe"**: C'est un point de l'espace de phase vers lequel tendent les trajectoires, c'est donc une solution stationnaire constante.

\* **L'attracteur "cycle limite"** : Il peut arriver que la trajectoire de phase se referme sur elle-même, c'est donc une solution périodique du système.

\* **L'attracteur "tore"**: Le système présente au moins deux périodes simultanées dont le rapport est irrationnel. La trajectoire de phase ne se referme pas sur elle-même, mais s'enroule sur une variété de dimension 2, c'est une solution pseudo-périodique du système.

\* **Les attracteurs étranges** : Ils sont bien plus complexes que les autres, on parle d'attracteur étrange lorsque la dimension fractale n'est pas entière.

L'attracteur étrange est une représentation d'un système chaotique dans un espace de phases bien précise. Dans le cas d'un système chaotique, la trajectoire converge vers une région particulière de l'espace qui est une signature de chaos : c'est ce qui différencie un signal chaotique d'un signal aléatoire.

### **Les caractéristiques de l'attracteur étrange**

- Dans l'espace des phases, l'attracteur est de volume nul.
- La dimension  $d$  de l'attracteur est fractale (non-entière) avec  $2 < d < n$ , où  $n$  est la dimension de l'espace des phases.
- Sensibilité aux conditions initiales : deux trajectoires de l'attracteur initialement voisines finissent par s'écarter l'une de l'autre [2].

## 1.4 Point fixe

On appelle un point fixe (ou point stationnaire ou point d'équilibre ou critique) du système (1.2) le point  $x^*$  de l'espace de phase tel que [2]:

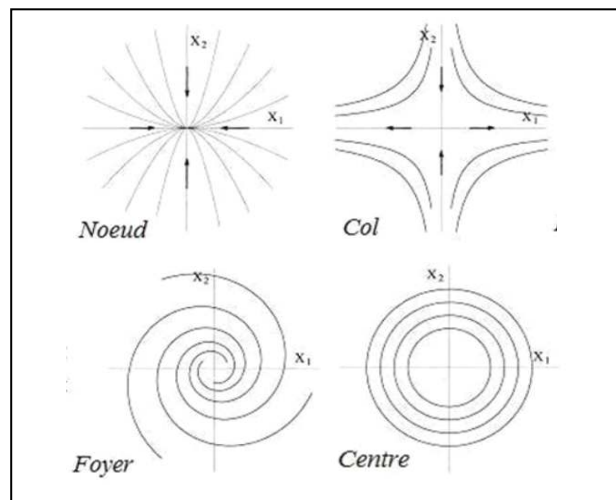
$$f(x^*) = 0 \quad (1.4)$$

Par le changement de variable  $X = x - x^*$ , on peut ramener le point  $x^*$  à l'origine.

## 1.5 Stabilité des points fixes

À partir des valeurs propres de la matrice jacobéenne  $\lambda_i$ , on peut distinguer les types de stabilités avec  $\lambda_i = A_i + jB_i$  [2][6].

- si toutes les valeurs propres  $\lambda_i$  ont leur partie réelle négative  $\rightarrow$  point d'équilibre asymptotiquement stable.
- si l'une ou plusieurs valeurs propres sont imaginaires pure  $B_i$ , les autres valeurs propres ayant leur partie réelle négative  $\rightarrow$  point fixe stable (centre).
- si l'une des valeurs propres à sa partie réelle positive  $A_i \rightarrow$  point fixe instable (point col).
- si toutes les valeurs propres  $\lambda_i$  ont leur partie réelle positive, et la partie imaginaire existe  $B_i \rightarrow$  point fixe nœud.
- si toutes les valeurs propres  $\lambda_i$  ont leur partie réelle négative, et la partie imaginaire non nulle  $B_i \rightarrow$  point fixe puits (foyer).



**Figure 1.2.** Les types de stabilité de point d'équilibre.

## 1.6 Exposants de Lyapunov

Les exposants de Lyapunov, présentés par Oseledec en 1968, jouent un rôle important dans l'étude des systèmes chaotiques [5]. ils qualifient le degré de divergence des trajectoires d'un système dynamique non linéaire soumis à des conditions initiales différentes. Cette divergence est exprimée par les exposants de Lyapunov [4]. Les exposants essayent si c'est possible de mesurer, sinon d'estimer la vitesse de divergence ou de convergence, qui caractérise le taux de séparation de deux trajectoires très proches [1], au sein de cet espace borné qu'est l'attracteur étrange. Le nombre d'exposants de Lyapunov est égal à la dimension de l'espace des phases et les caractéristiques du système dépendent de leurs signes d'après le tableau suivant :

Etat	Type d'attracteur	Signe des exposantes
Point fixe	Point	-,,-
Cycle limite périodique	Cercle	0,-,-
Cycle limite quasi-périodique	Tore	0,0,-
Chaotique	Attracteur étrange	+,0-
Hyper chaotique	Attracteur étrange	+,+,-

**Tableau 1.1.** Caractérisation des attracteurs

Pour un attracteur non chaotique, les exposants de Lyapunov sont tous inférieurs ou égaux à zéro et leur somme est négative. Un attracteur étrange possèdera toujours au moins trois exposants de Lyapunov dont un au moins doit être positif et leur somme est négative.

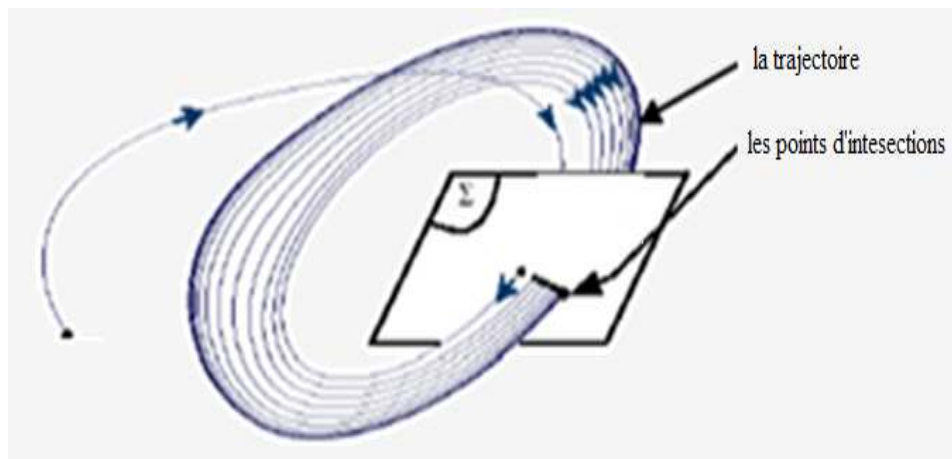
## 1.7 L'espace de phase

On définit à un instant donné, un point dans un repère ce point caractérise l'état de système dans l'espace à cet instant cet espace est appelée l'espace de phase. C'est technique qualitative simple et efficace permet la détermination de type de stabilité de point d'équilibre.

## 1.8 La section de Poincaré

La section de Poincaré est l'intersection d'une trajectoire (périodique, quasi-périodique ou chaotique) dans un espace d'au moins trois dimensions [3].

La section de Poincaré est un outil mathématique simple permettant de transformer un système dynamique continu en un système discret [8], souvent utilisé pour caractériser le chaos, et étudier la dynamique d'un système. C'est une visualisation par échantillonnage du système avec une paramétrisation qui doit être choisie convenablement pour accéder au maximum d'informations.



*Figure 1.3.* Section de Poincaré.

### Principe de la section de Poincaré

- **un unique point:** le système est périodique.
- **un petit nombre de points:** le système est périodique.
- **une courbe fermée:** le système est quasi-périodique.
- **un nuage de points:** le système est chaotique.

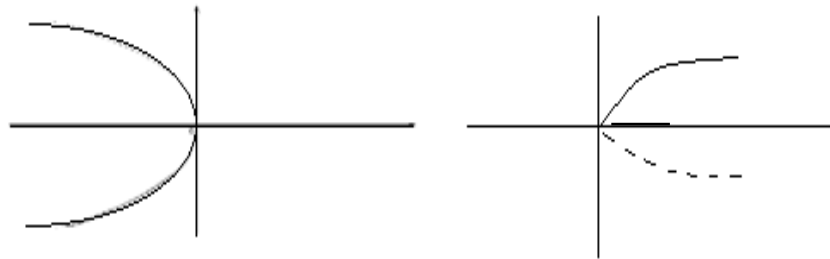
## 1.9 La bifurcation

La théorie de bifurcation est l'étude mathématique des changements qualitatifs ou topologiques de la structure d'un système dynamique.

Une bifurcation survient lorsqu'une variation quantitative d'un paramètre du système engendre un changement qualitatif des propriétés d'un système telles que la stabilité, le nombre de points d'équilibre ou la nature des régimes permanents [5]. Les valeurs des paramètres au moment du changement sont appelées valeurs de bifurcation.

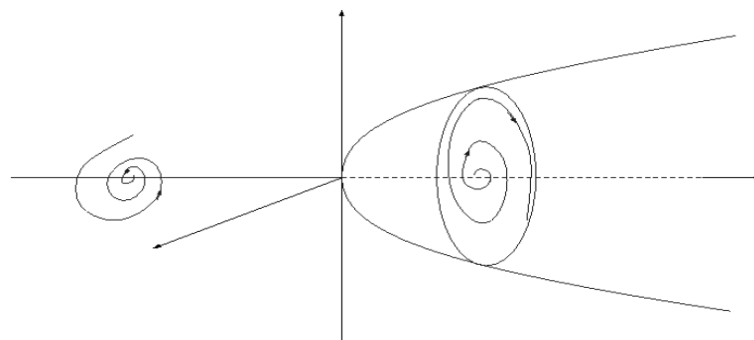
### 1.9.1 Types de bifurcation

La théorie des bifurcations consiste à classer les différents types de bifurcations en classes. Chaque classe correspond à une certaine symétrie dans le problème. Parmi les différents types de bifurcations, on trouve :



**a.** Les bifurcations de fourche.

**b.** Les bifurcations de col-nœud.



**c.** Les bifurcations de Hopf.

**Figure 1.4.** Les différents types de bifurcations.

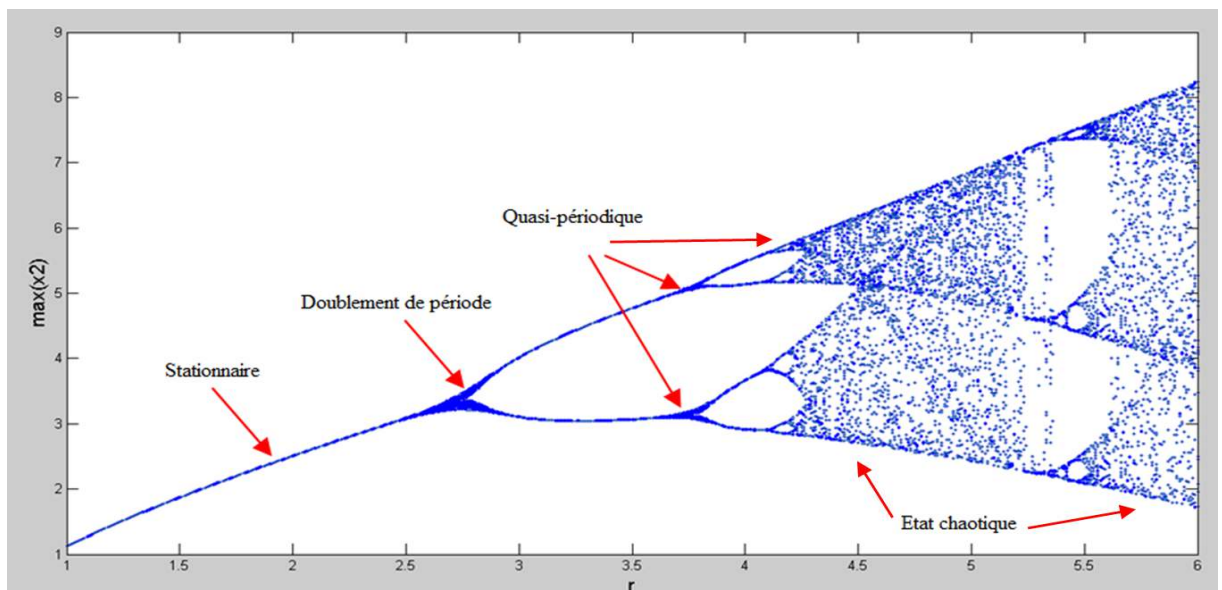
- Les bifurcations « de fourche »: Un équilibre stable se déstabilise en un équilibre instable, et deux équilibres stables sont créés.
- Les bifurcations col-nœud « saddle-node »: Deux points d'équilibres existent (un stable et un instable) avant la bifurcation. Après la bifurcation, plus aucun équilibre n'existe.

- Les bifurcations de Hopf: Ce sont des bifurcations oscillantes, comme l'attracteur de Lorenz.
- Les bifurcations de dédoublement de période: Ce sont des bifurcations qui mènent à des comportements chaotiques

### 1.9.2 Diagramme de bifurcation

Le diagramme de bifurcations unidimensionnel est un tracé repérant la nature des différentes solutions du système et leur stabilité lorsqu'un paramètre varie[8]. Ce diagramme résume toute l'information sur la bifurcation et permet de comprendre de ce fait comment évolue le système.

Dans les systèmes dynamiques, un diagramme de bifurcation montre les comportements possibles d'un système à long terme, en fonction des paramètres de bifurcation.



**Figure 1.5.** Diagramme de bifurcation de Rössler.

- pour  $1 < r < 2.7$ , le système possède un point fixe d'un cycle d'ordre 1 (stationnaire).
- pour  $2.7 < r < 3.7$ , le système possède un cycle limite d'un cycle d'ordre 2 (doublement de période).
- pour  $3.7 < r < 4.3$ , le système possède un tore d'un cycle d'ordre 4, 8, 16 .....ect (quasi-périodique).
- pour  $4.4 < r$ , le système possède un chaos (état chaotique).



## 1.10 Exemple d'attracteur

### ❖ Attracteur de Lorenz

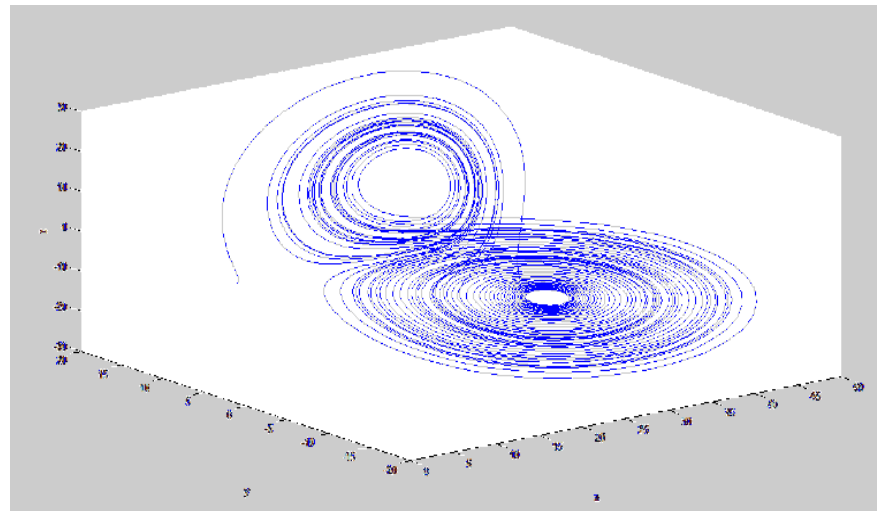
En 1963, le météorologue Edward Lorenz est le premier à mettre en évidence le caractère vraisemblablement chaotique de la météorologie, ce modèle est un système dynamique tridimensionnel qui engendre un comportement chaotique dans certaines conditions.

Lorenz a proposé un système différentiel possédant trois degrés de liberté, notés  $x$ ,  $y$ ,  $z$  qui s'écrit :

$$\begin{cases} \dot{x} = a(y - x) \\ \dot{y} = bx - y - xz \\ \dot{z} = xy - cz \end{cases} \quad (1.5)$$

Les variables dynamiques  $\dot{x}$ ,  $\dot{y}$  et  $\dot{z}$  représentent l'état du système à chaque instant.

Dans ces équations,  $a$ ,  $b$  et  $c$  sont trois paramètres réels strictement positifs ; nous les fixerons respectivement à 10, 28, et  $8/3$ .



**Figure 1.6.** Attracteur de Lorenz ( $a=10$ ,  $b=28$ ,  $c=8/3$ ).

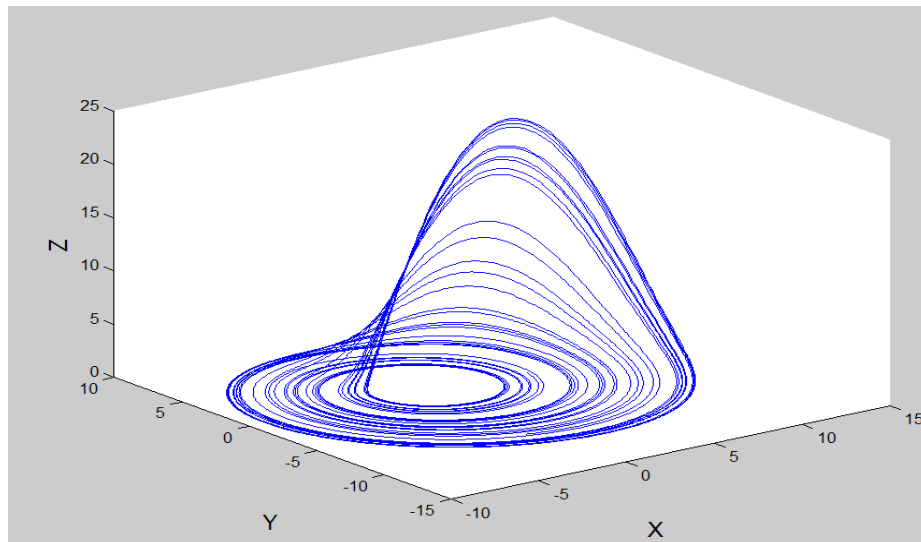
### ❖ Attracteur de Rössler

Le système de Rössler est proposé par l'Allemand Otto Rössler, lié à l'étude de l'écoulement des fluides. Les équations de ce système ont été découvertes à la suite de travaux en cinétique chimique.

Les équations de ce système sont les suivantes :

$$\begin{cases} \dot{x} = -y - z \\ \dot{y} = x + ay \\ \dot{z} = b + zx - zr \end{cases} \quad (1.6)$$

Les trois paramètres  $a$ ,  $b$ ,  $r$  sont réels strictement positifs, fixés respectivement à 0.2, 0.2, et 5.



**Figure 1.7.** Attracteur de Rössler ( $a=0.2$ ,  $b=0.2$ ,  $r=5$ ).

## 1.11 Conclusion

Dans ce chapitre nous avons présenté quelques notions et définitions de base des systèmes dynamiques non linéaire notamment chaotiques.

Nous avons présenté les caractéristiques fondamentales du chaos tels que : le plan de phase, la sensibilité aux conditions initiales où un petit écart entre deux conditions initiales conduit à une divergence rapide des trajectoires au cours de temps, la section de Poincaré, les attracteurs chaotiques, l'importance de l'exposant de Lyapunov pour montrer le comportement chaotique, et nous avons terminé par le diagramme de bifurcation qui met en évidence l'évolution d'un système vers le chaos.

# Chapitre 2 Analyse du système hyper chaotique de

Qi

---

## 2.1 Introduction

Nous allons consacrer ce chapitre à la présentation du système chaotique de Qi, qui est très utilisé dans le domaine des transmissions sécurisées de données, car il présente l'avantage d'être hyper chaotique en raison de ses deux exposants de Lyapunov positifs, et sa dynamique évolue dans une large bande de fréquence. Nous allons ainsi étudier ses différentes caractéristiques (section de Poincaré, exposant de Lyapunov, diagramme de bifurcation ...ect) à l'aide de MATLAB et ses outils (Simulink et MATDS).

## 2.2 Description du système

Le système hyper chaotique de Qi est donné par:

$$\begin{cases} \frac{dx_1}{dt} = \dot{x}_1 = a(x_2 - x_1) + x_2 x_3 \\ \frac{dx_2}{dt} = \dot{x}_2 = b(x_1 + x_2) - x_1 x_3 \\ \frac{dx_3}{dt} = \dot{x}_3 = -c x_3 - e x_4 + x_1 x_2 \\ \frac{dx_4}{dt} = \dot{x}_4 = -d x_4 + f x_3 + x_1 x_2 \end{cases} \quad (2.1)$$

Les variables  $x_1, x_2, x_3, x_4$  représentent l'état du système, et  $a, b, c, d, e$  et  $f$  sont des paramètres réels.

Pour la simulation numérique nous utilisons les paramètres suivants :

$$a=50, b=24, c=13, d=8, e=33 \text{ et } f=30.$$

## 2.3 Analyse du système hyper chaotique de Qi

### 2.3.1 Etude des points fixes

Pour déterminer les points fixes du système, nous devons résoudre le système d'équations algébriques suivant :

$$\frac{dx}{dt} = f(x) = 0 \quad (2.2)$$

C'est à dire:

$$\begin{cases} \dot{x}_1 = a(x_2 - x_1) + x_2 x_3 = 0 \\ \dot{x}_2 = b(x_1 + x_2) - x_1 x_3 = 0 \\ \dot{x}_3 = -c x_3 - e x_4 + x_1 x_2 = 0 \\ \dot{x}_4 = -d x_4 + f x_3 + x_1 x_2 = 0 \end{cases} \quad (2.3)$$

Nous obtenons une solution  $\dot{x}_1 = \dot{x}_2 = \dot{x}_3 = \dot{x}_4 = 0$ .

Pour étudier la stabilité de ce point fixe, nous déterminons les valeurs propres de la matrice jacobéenne.

Le système de Qi (2.3) sous forme matricielle s'écrit :

$$P = \begin{pmatrix} -a & a & 0 & 0 \\ b & b & 0 & 0 \\ 0 & 0 & -c & -e \\ 0 & 0 & f & -d \end{pmatrix} \quad (2.4)$$

Où P est le point fixe des coordonnées (0 0 0 0)

et la matrice jacobéenne:

$$N = \begin{pmatrix} \lambda & 0 & 0 & 0 \\ 0 & \lambda & 0 & 0 \\ 0 & 0 & \lambda & 0 \\ 0 & 0 & 0 & \lambda \end{pmatrix} \quad (2.5)$$

avec:  $\lambda$  la valeur propre de la matrice, et I la matrice identité (4\*4).

L'équation caractéristique est donné par:  $\det (P(0 0 0 0) - \lambda I) = 0$ , soit:

$$[(\lambda + a) (\lambda - b) - ab][(\lambda + c) (\lambda + d) + ef] = 0$$

Après la résolution de l'équation caractéristique nous trouvons les racines suivantes:

$$\lambda_1 = -63.68 \quad , \quad \lambda_2 = 38.68$$
$$\lambda_3 = -10.5 + i 31.36 \quad , \quad \lambda_4 = -10.5 - i 31.36$$

Nous avons quatre solutions dont deux réelles et deux complexes, alors nous avons un point fixe instable.

### 2.3.2 Evolution du système de Qi en fonction de temps

Nous avons utilisé MATLAB Simulink pour visualiser les différents signaux issus du système de Qi, tels que : les états  $x_i$  en fonction du temps, les plans de phase et les attracteurs chaotiques.

Pour cela, nous avons implémenté sous Matlab-Simulink, le circuit qui définit le système hyper-chaotique de Qi à partir des équations (2.1).

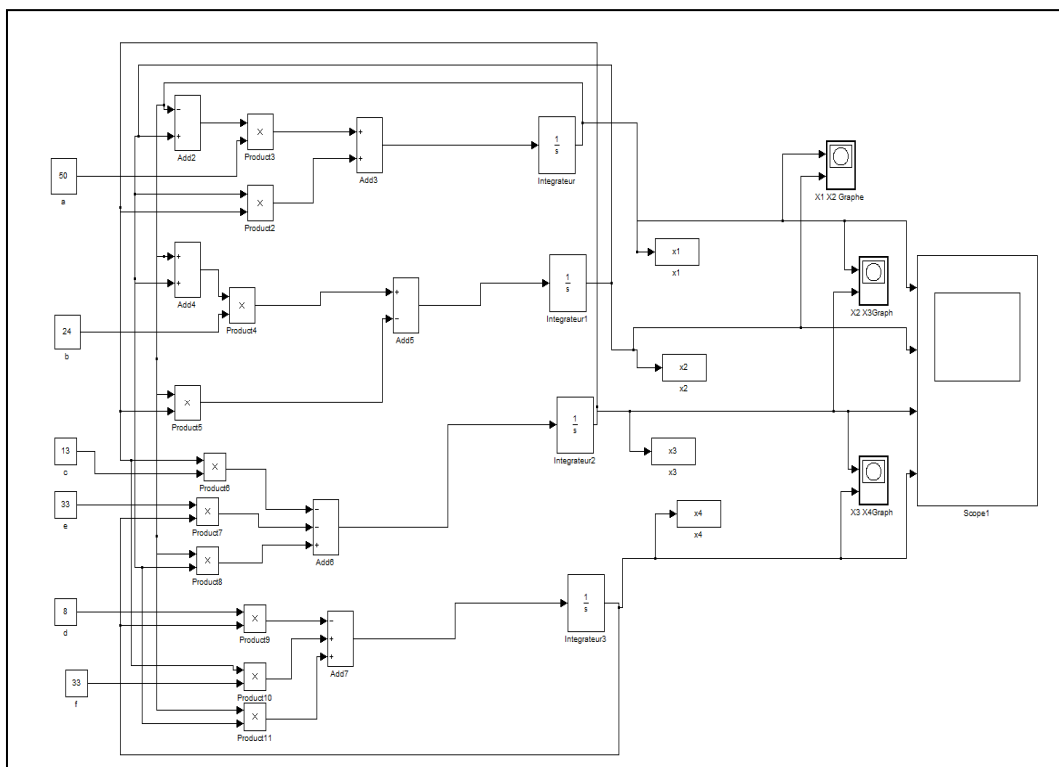
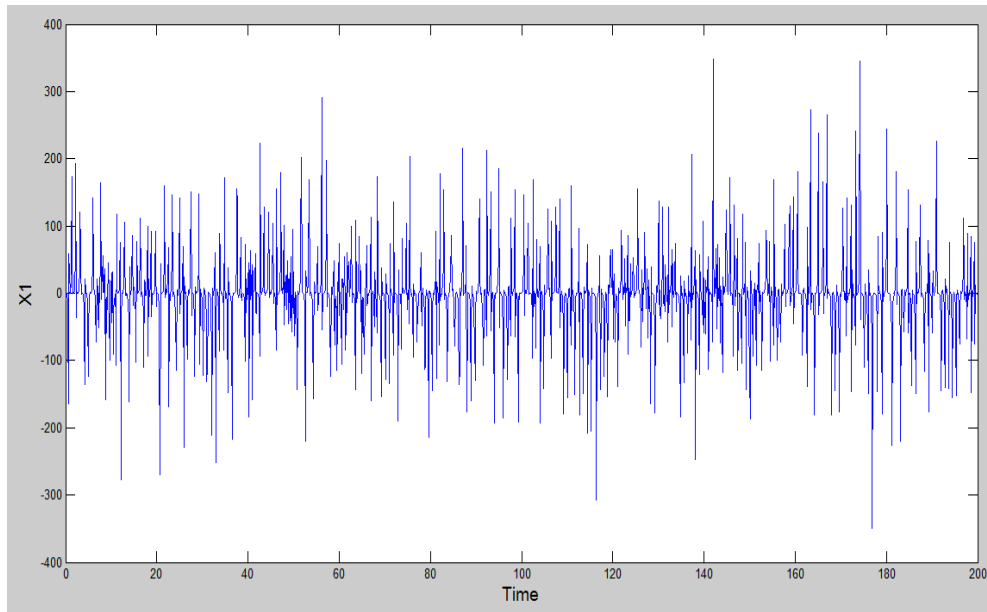
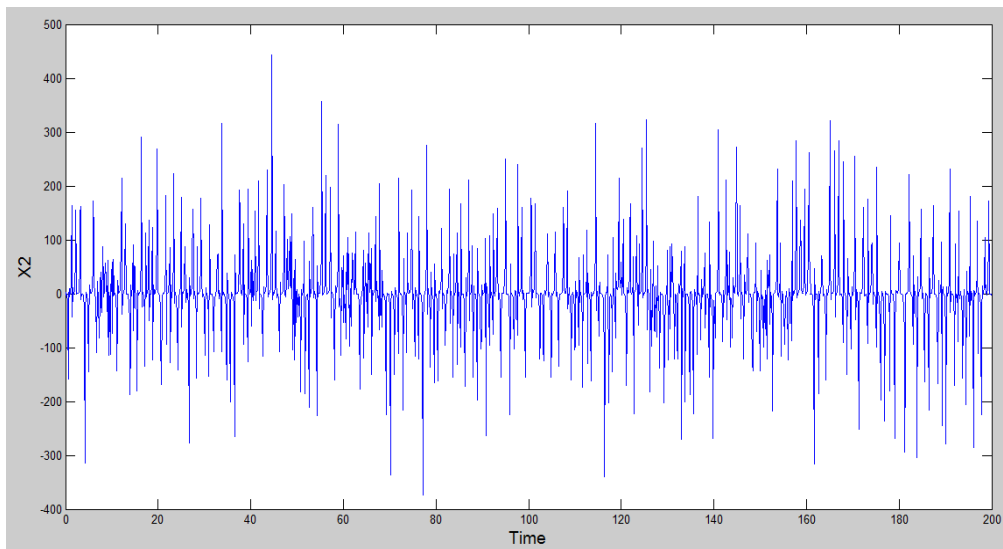


Figure 2.1. Représentation du système de Qi sous MATLAB Simulink.

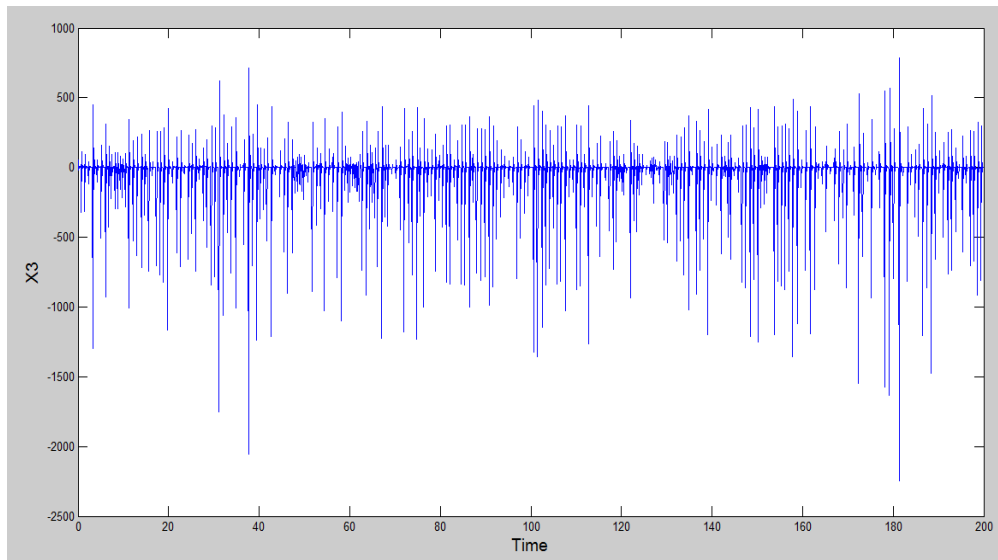
Les figures 2.2 à 2.6 représentent les différentes courbes des états  $x_1$ ,  $x_2$ ,  $x_3$ ,  $x_4$  en fonction du temps.



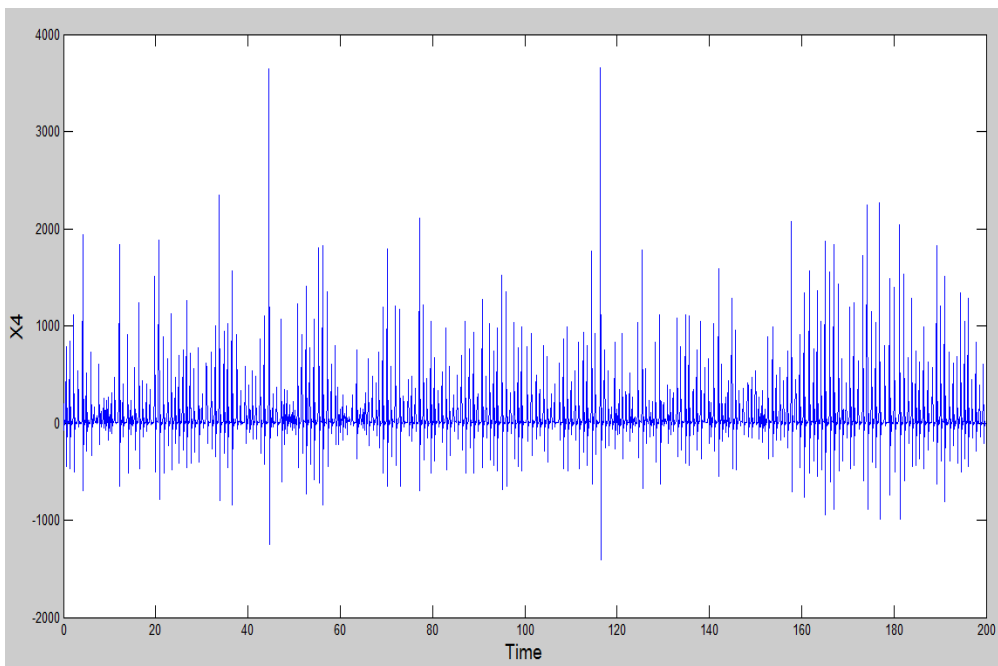
**Figure 2.2.** L'état  $x_1$  en fonction du temps t.



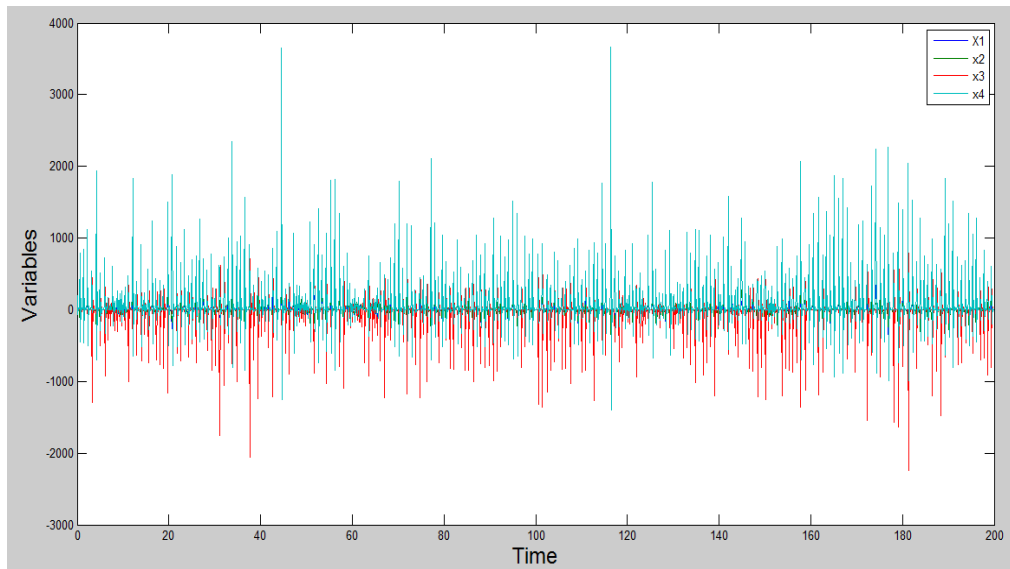
**Figure 2.3.** L'état  $x_2$  en fonction du temps t.



**Figure 2.4.** L'état  $x_3$  en fonction du temps  $t$ .



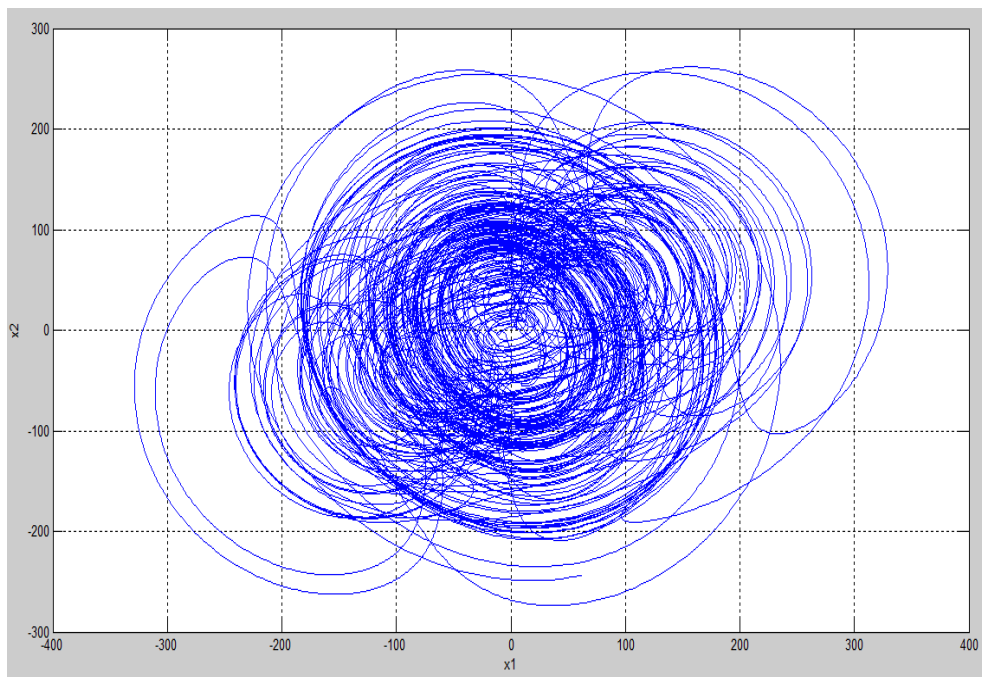
**Figure 2.5.** L'état  $x_4$  en fonction du temps  $t$ .



**Figure 2.6.** Les états  $x_1, x_2, x_3, x_4$  en fonction du temps  $t$ .

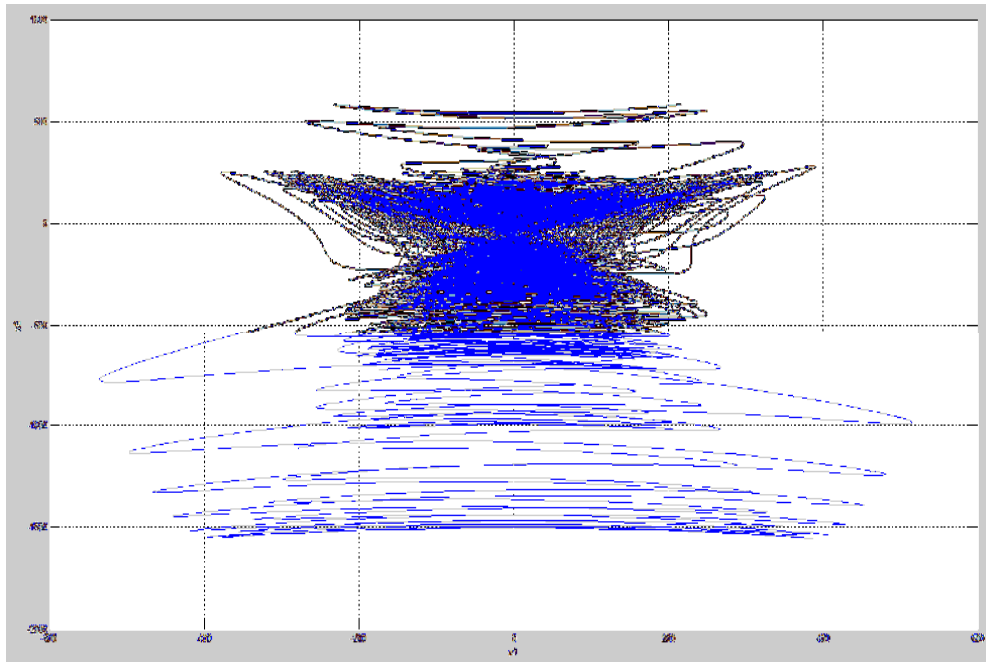
### 2.3.3 Plan de phase

Les figures 2.7 à 2.9 représentent les différents plans de phase du système hyperchaotique de Qi.

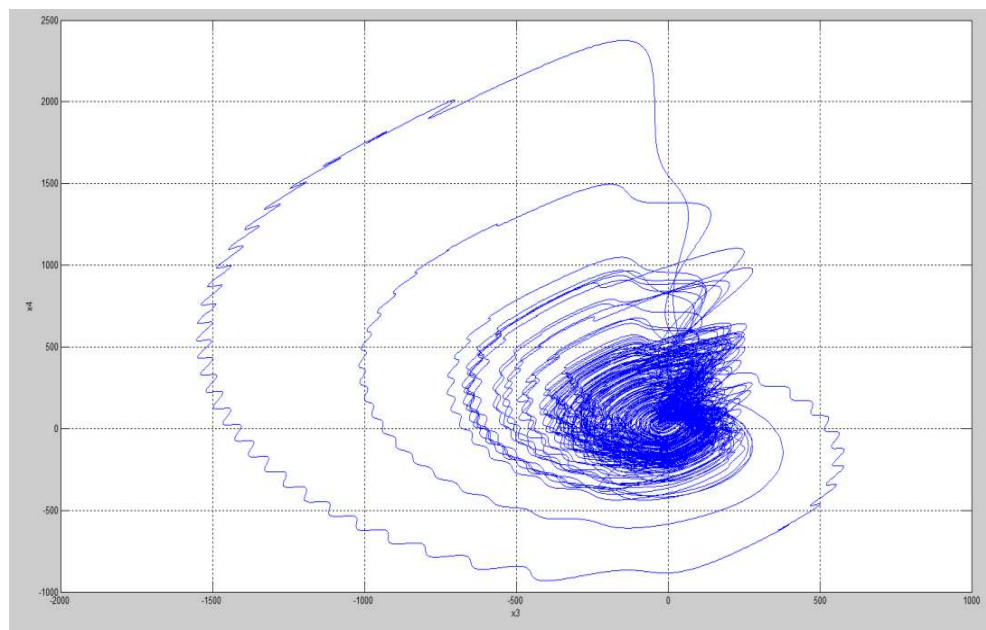


**Figure 2.7.** Plan de phase  $x_2$  en fonction de  $x_1$ .





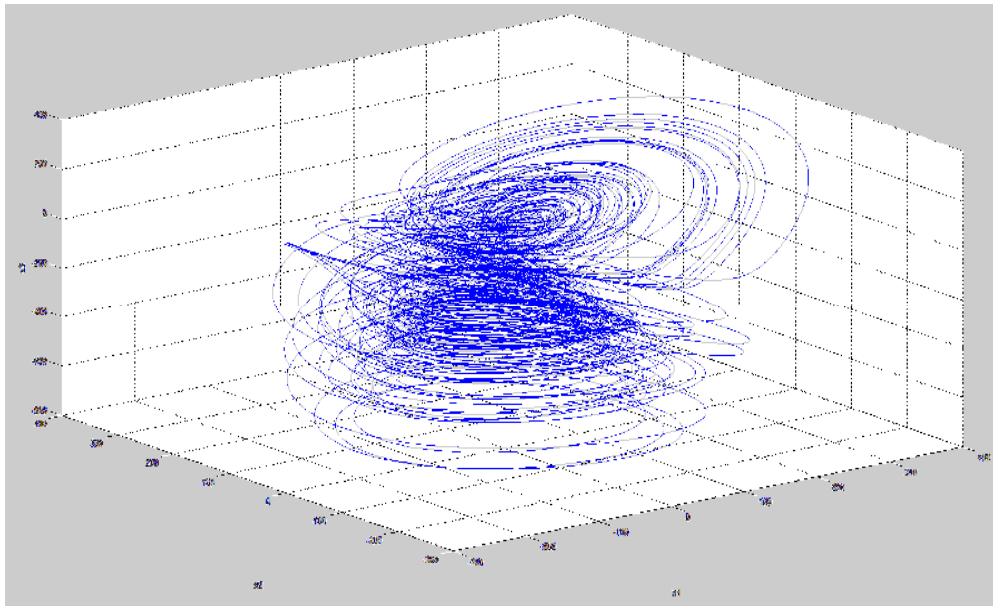
**Figure 2.8.** Plan de phase  $x_3$  en fonction de  $x_1$ .



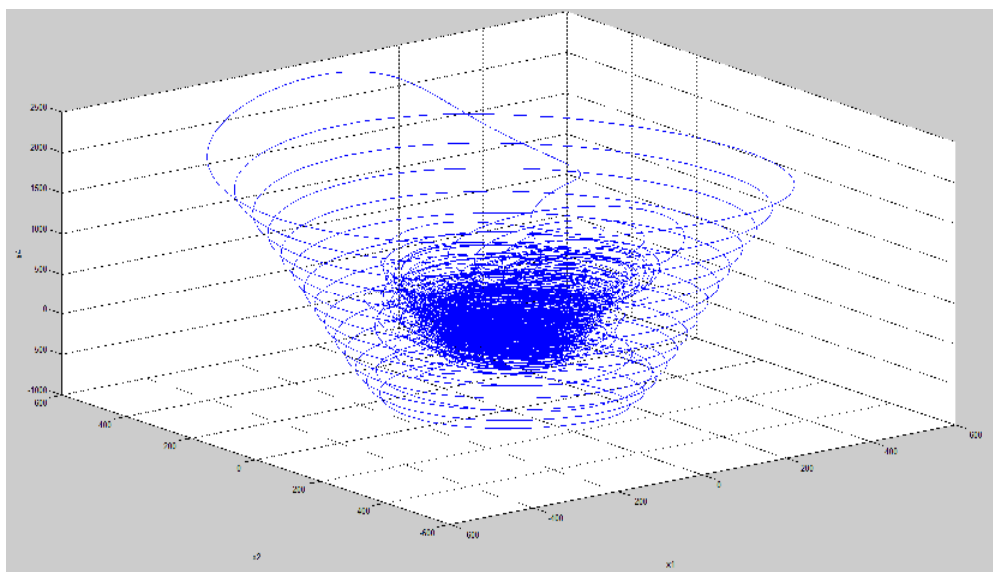
**Figure 2.9.** Plan de phase  $x_4$  en fonction de  $x_3$ .

### 2.3.4 Attracteur étrange

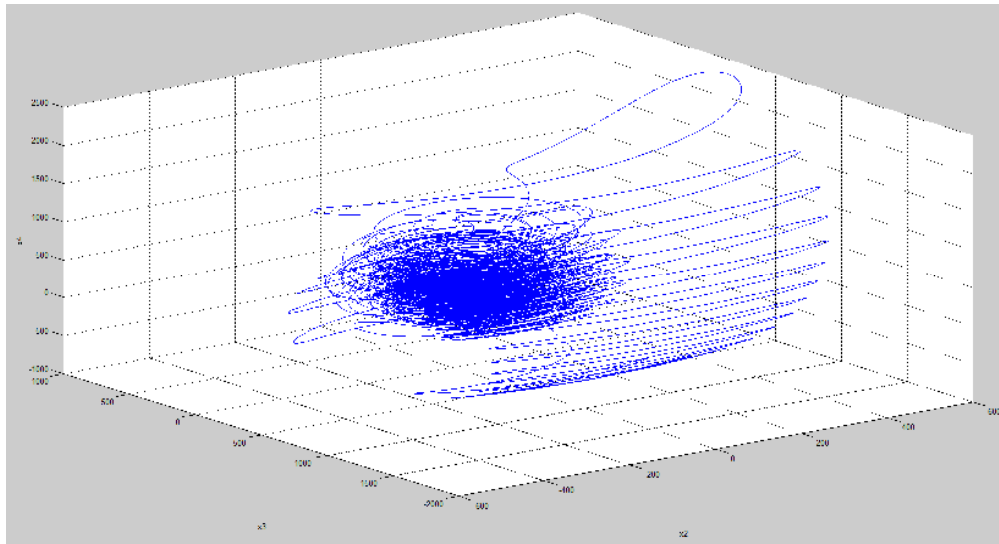
Les figures 2.10 à 2.12 représentent l'attracteur hyper-chaotique de Qi en 3D.



**Figure 2.10.** Attracteur étrange  $x_1$ ,  $x_2$  et  $x_3$ .



**Figure 2.11.** Attracteur étrange  $x_1$ ,  $x_2$  et  $x_4$ .



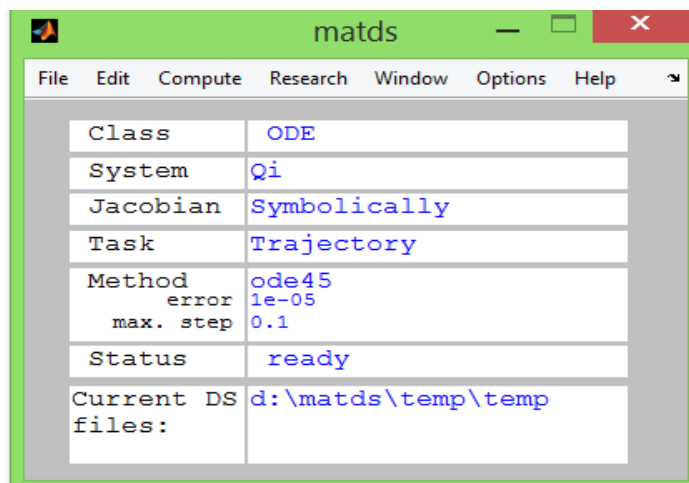
**Figure 2.12.** Attracteur étrange  $x_2$ ,  $x_3$  et  $x_4$ .

### 2.3.5 Exposants de Lyapunov

Nous avons utilisé l'outil MATDS sous environnement MATLAB qui permet l'étude des systèmes dynamiques, la détermination des points d'équilibres, le tracé de la section de Poincaré et le calcul des exposants de Lyapunov.

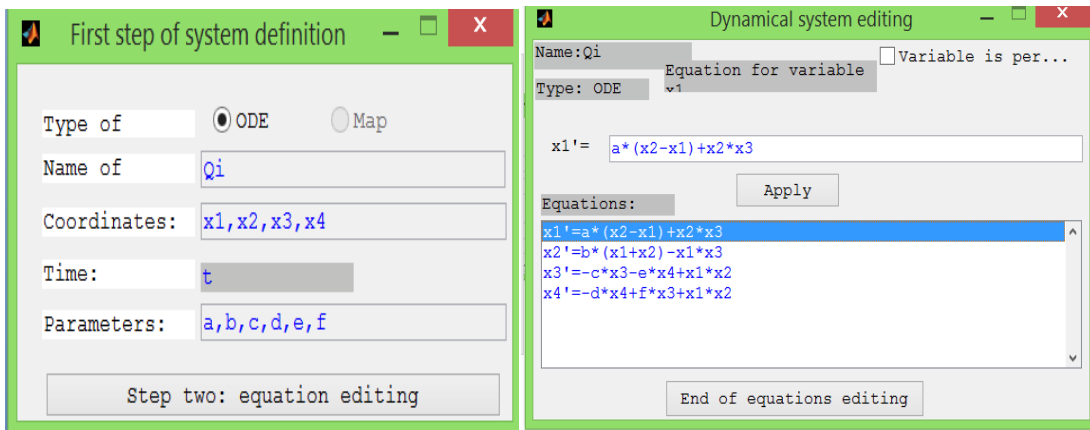
Nous allons commencer par la présentation de l'outil MATDS :

Au démarrage de MATDS la fenêtre suivante apparaît :



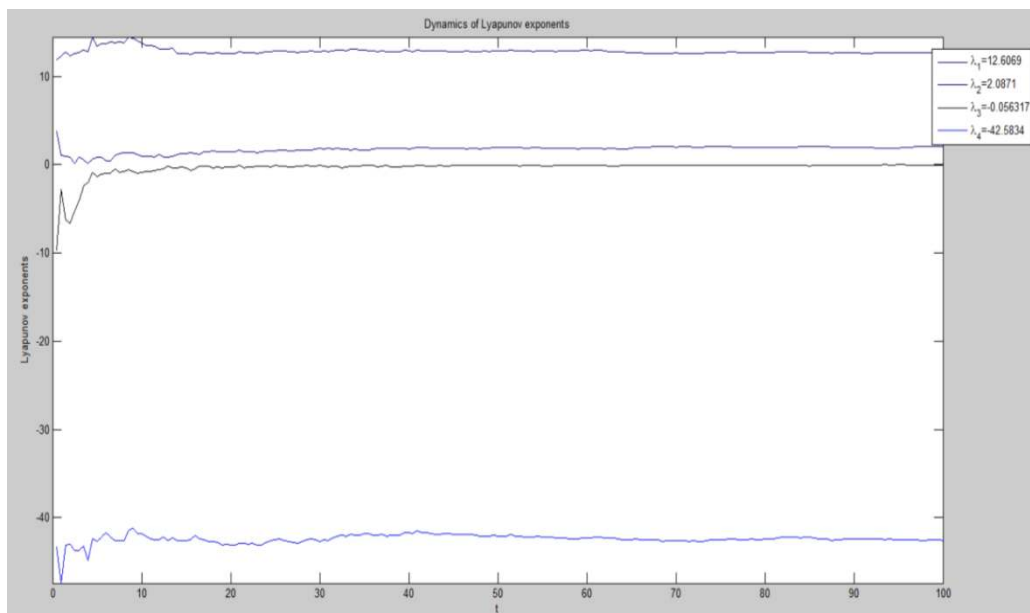
**Figure 2.13.** L'interface de l'outil MATDS.

Par la suite, nous allons créer et entrer les équations et les paramètres de notre système.



**Figure 2.14.** Création du système de Qi en MATDS.

Nous savons que pour un attracteur hyper-chaotique, il faut que la somme des exposants de Lyapunov soit négative et qu'au moins un de ses exposants soit positif. Pour notre système hyper-chaotique de Qi, les exposants de Lyapunov sont représentés sur la figure 2.15.



**Figure 2.15.** Exposant de Lyapunov du système de Qi.

Notre système hyper-chaotique de Qi a quatre équations différentielles (ordre 4), donc nous obtenons quatre exposants de Lyapunov:

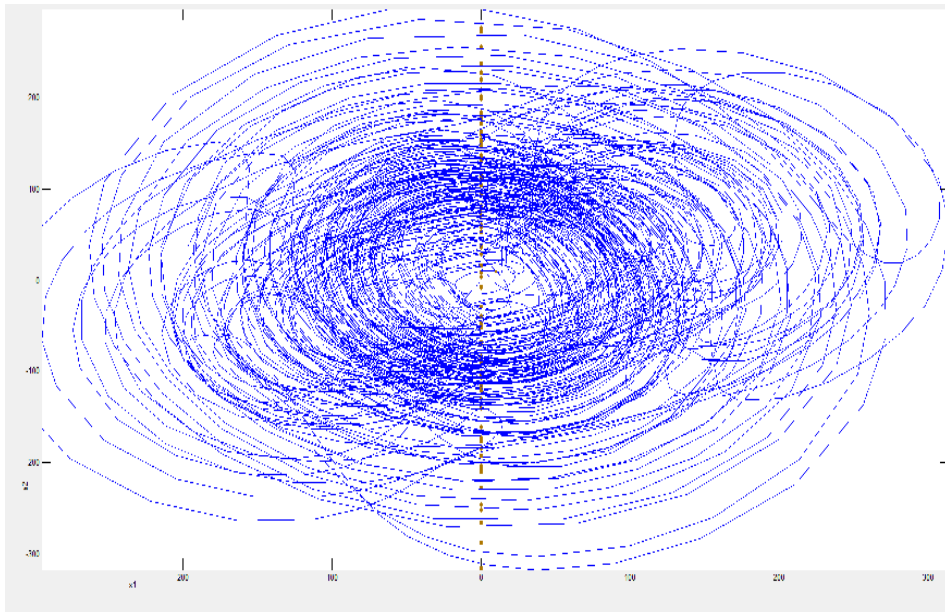
$$\lambda_1 = 12.6069 \quad , \quad \lambda_2 = 2.0871$$

$$\lambda_3 = -0.56317 \quad , \quad \lambda_4 = -42.5834$$

Nous avons trouvé quatre exposants dont deux positifs et deux négatifs, et dont la somme est négative.

### 2.3.6 Section de Poincaré

Nous savons que la section de Poincaré est l'intersection d'une trajectoire dans un espace qui nous permet de différencier un système chaotique d'un autre; donc nous traçons l'intersection du plan de phase  $x_2$  en fonction de  $x_1$  avec le plan  $x_1=0$ . La section de Poincaré est alors représentée par les points marrons sur la figure 2.16.

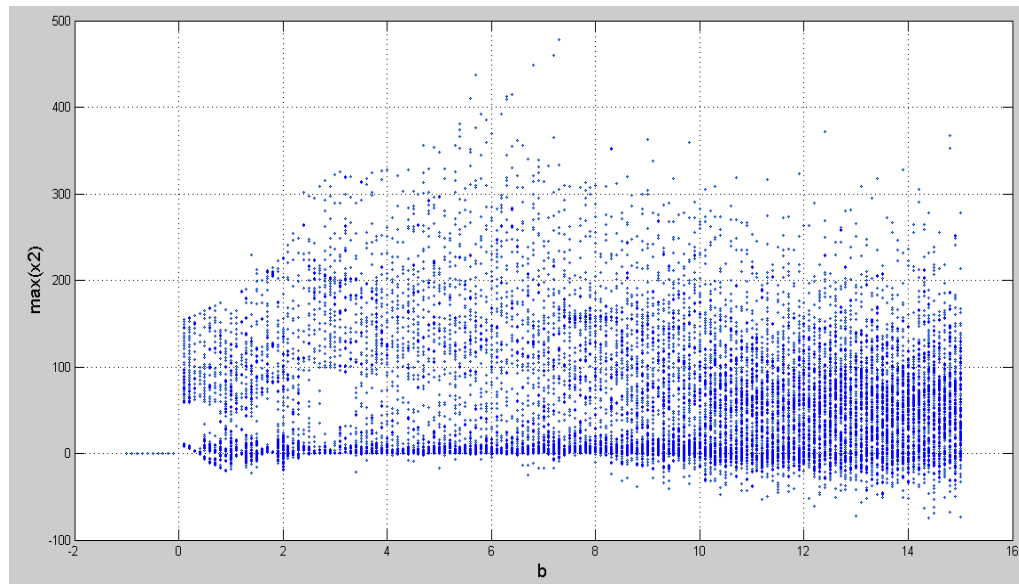


**Figure 2.16.** La section de Poincaré.

### 2.3.7 Diagramme de bifurcation

Pour tracer le diagramme de bifurcation, nous avons utilisé un programme sous MATLAB où le paramètre variable est  $b$ . Le diagramme obtenu est illustré sur la figure 2.17.

Nous remarquons que :

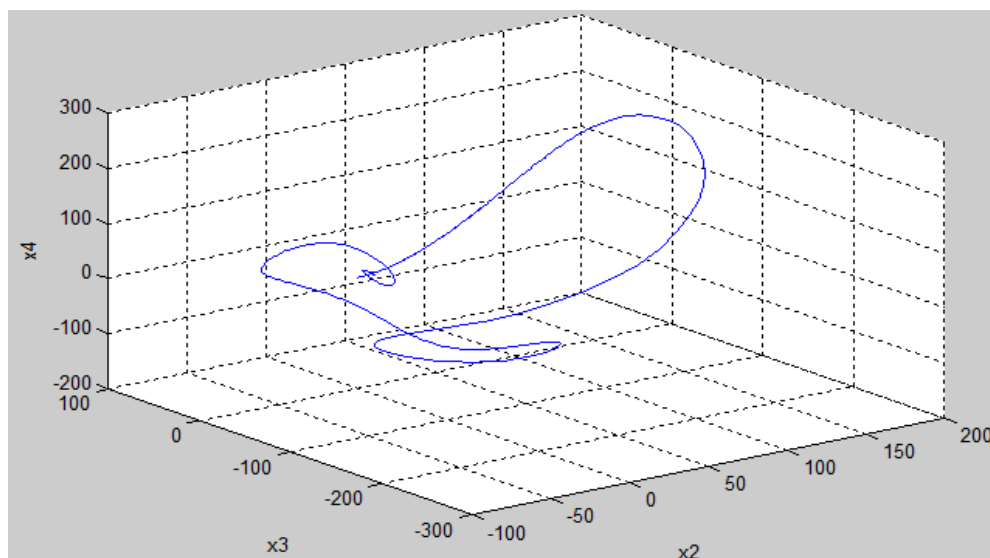


**Figure 2.17.** Diagramme de bifurcation de  $Q_i$ .

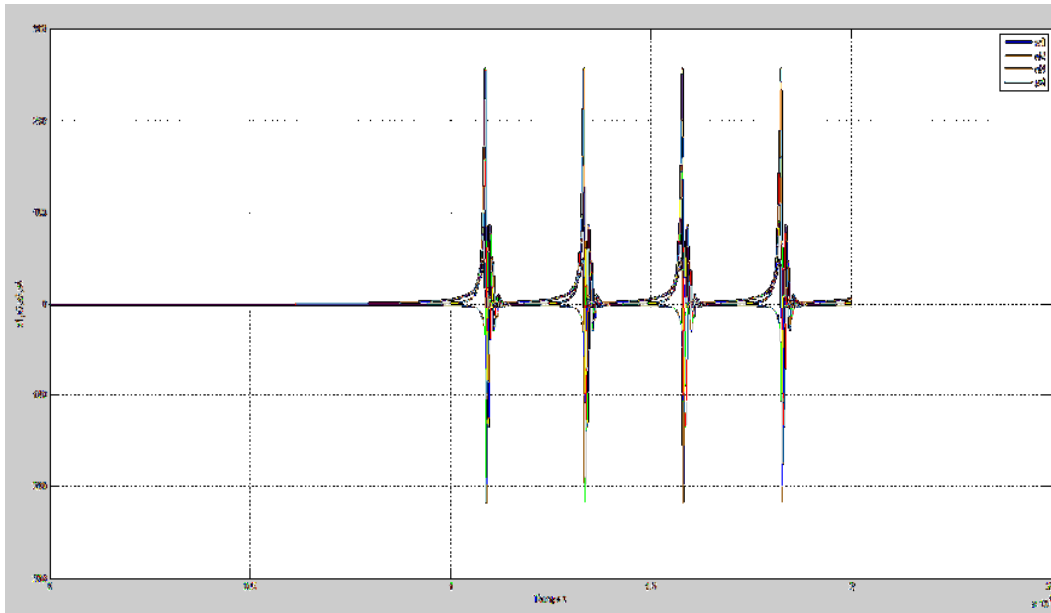
- pour  $b < 1$ , le système est périodique
- pour  $1 < b < 4$ , le système est doublement de période
- pour  $4 < b$ , le système est chaotique les figures précédentes le confirme

Les figures suivantes confirment les états non chaotiques du système

\* L'état périodique

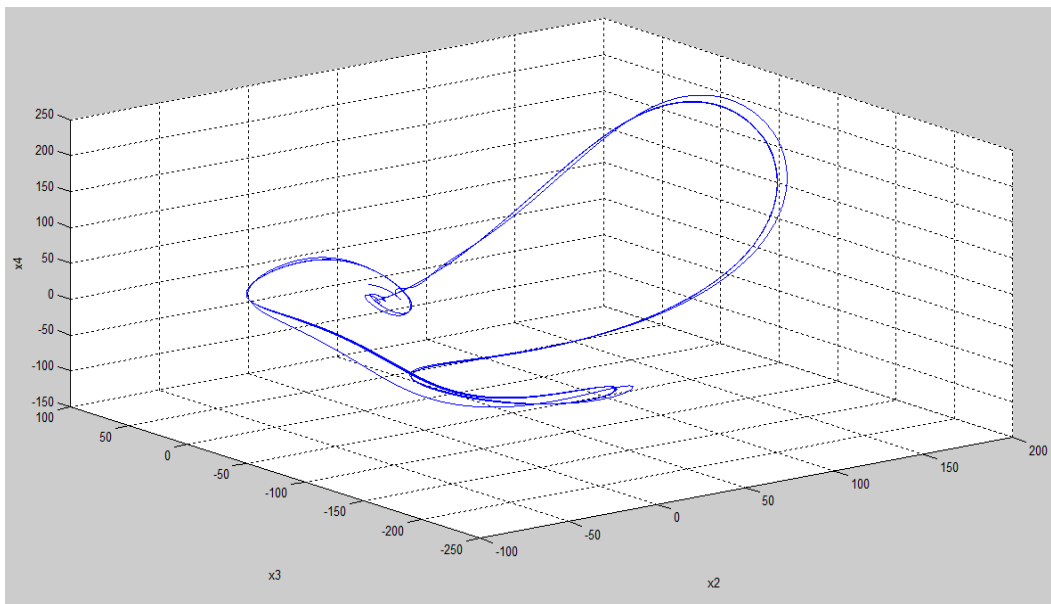


**Figure 2.18.** Attracteur périodique  $x_2$ ,  $x_3$  et  $x_4$  lorsque  $b=0.1$ .

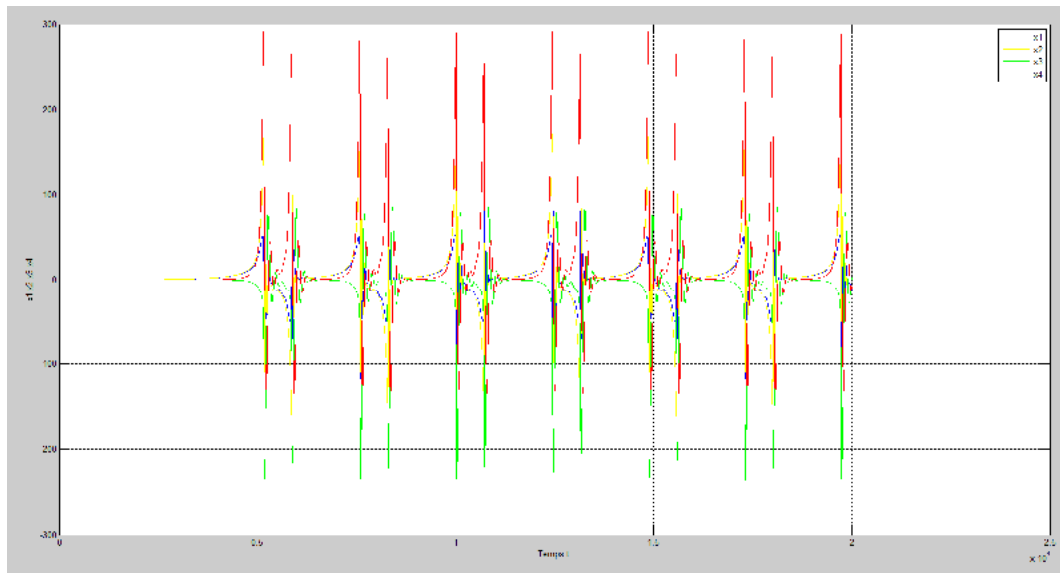


**Figure 2.19.** Les états  $x_1$ ,  $x_2$ ,  $x_3$  et  $x_4$  en fonction du temps lorsque  $b=0.1$ .

\* L'état doublement de période



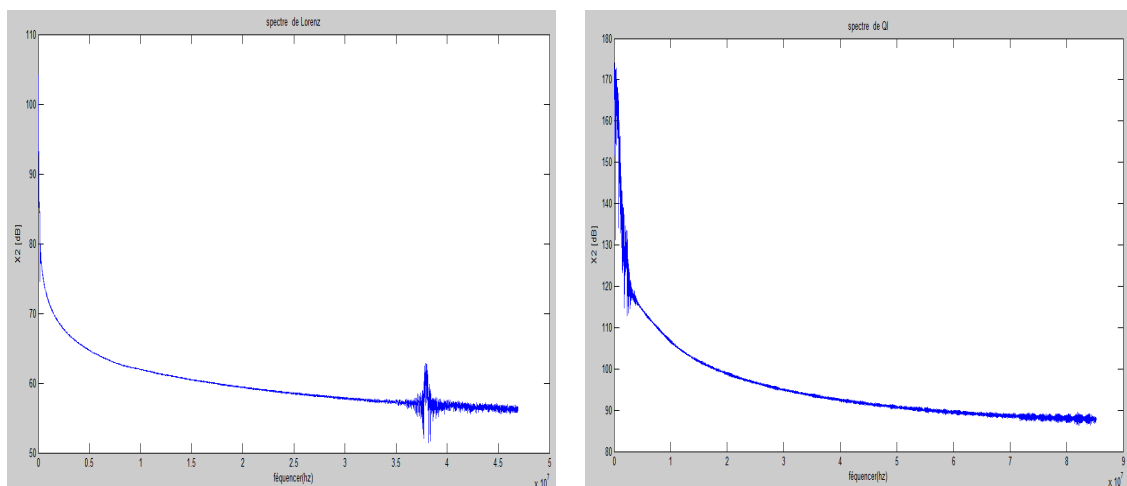
**Figure 2.20.** Attracteur doublement de période  $x_2$ ,  $x_3$  et  $x_4$  lorsque  $b=1.5$ .



**Figure 2.21.** Les états  $x_1$ ,  $x_2$ ,  $x_3$  et  $x_4$  en fonction du temps lorsque  $b=1.5$ .

## 2.4 Spectre en fréquence

Pour confirmer le spectre large bande du système hyper-chaotique de Qi par rapport aux autres systèmes chaotiques (Lorenz, Rossler, etc...), nous allons tracer le spectre de ce système et le comparer avec celle de Lorenz, à l'aide de programme sous MATLAB. Les résultats obtenus sont les suivants:



**a.** spectre de Lorenz.

**b.** spectre de Qi.

**Figure 2.22.** Les spectres de fréquence.

Nous constatons que la largeur spectrale de Qi est plus que de Lorenz, alors le système de Qi peut masquer une grande quantité d'information par rapport à Lorenz.



## 2.5 Conclusion

Dans ce chapitre nous avons étudié le système hyper-chaotique de Qi, en se basant sur les définitions présentées au chapitre précédent, ses caractéristiques et ses principales propriétés en mettant en évidence son comportement hyper-chaotique.

Nous avons confirmé que le système hyper-chaotique de Qi présente des propriétés très intéressantes (hyper-chaotique, large bande, etc.), permettant ainsi son utilisation dans le cryptage sécurisée de données, et qui sera développé dans le chapitre suivant.

# Chapitre 3 Communication sécurisée par synchronisation chaotique adaptative

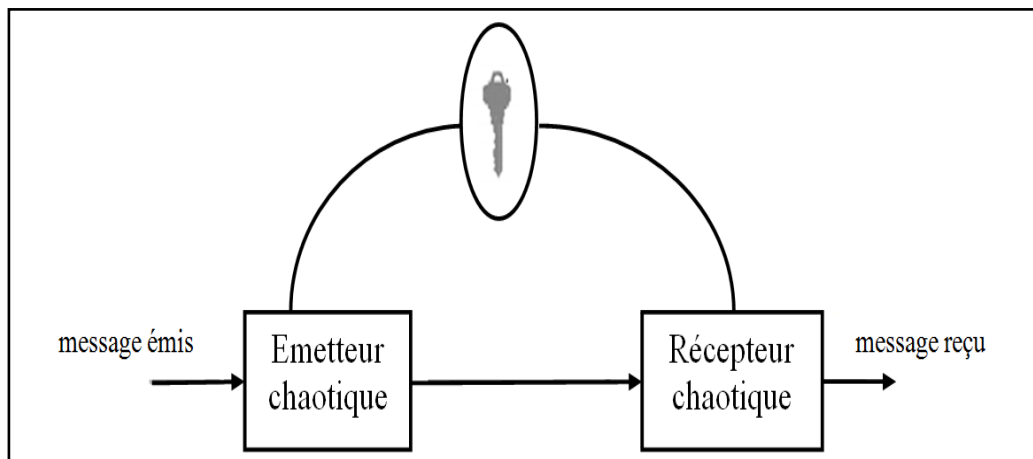
---

## 3.1 Introduction

Grâce aux propriétés naturelles des systèmes chaotiques, telles que leur sensibilité aux conditions initiales, l'aspect pseudo-aléatoire et l'évolution dans une large bande, les systèmes chaotiques sont devenus de bons candidats pour la transmission sécurisée.

L'idée d'utilisation du chaos dans les systèmes de communication a été inspirée de la découverte de Pecora-Carroll en 1990 : ils ont montré que deux systèmes chaotiques identiques avec des conditions initiales différentes peuvent se synchroniser [4].

Le diagramme principal de la communication sécurisée par le chaos est représenté sur la figure 3.1.



**Figure 3.1.** Diagramme principal de la communication sécurisée.

L'idée fondamentale exige que l'émetteur produise un signal chaotique pour masquer le message à transmettre. Le récepteur chaotique est induit pour synchroniser avec le signal entrant masqué, avec une simple opération de soustraction, nous pouvons récupérer le message crypté.

La clé du système de transmission est l'ensemble des paramètres des deux systèmes chaotiques qui doivent être synchronisés.

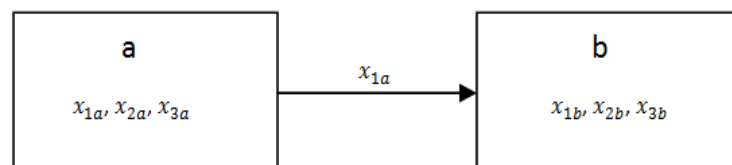
Dans ce chapitre nous allons exposer un système de communication en utilisant le chaos basé sur le cryptage paramétrique et la synchronisation adaptative qui comporte deux parties l'émetteur et le récepteur. Ainsi nous allons présenter les différentes techniques de cryptage et les principales méthodes de synchronisation et son importance pour reconstituer le message [8].

## 3.2 Les classes de synchronisation

Les méthodes traditionnelles de synchronisation sont en général basées sur l'utilisation de circuits identiques. Si par un moyen quelconque, on leur permet d'échanger de l'énergie, action que l'on nomme "couplage", il est alors possible de coupler les systèmes chaotiques dans un sens (couplage unidirectionnel) ou dans les deux sens (couplage bidirectionnel)[4].

### 3.2.1 Synchronisation unidirectionnelle

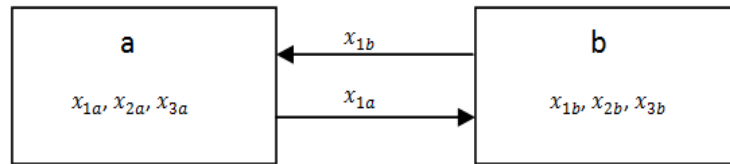
Dans le cas d'un couplage unidirectionnel, l'énergie est transférée d'un système à un autre, à l'aide d'un élément de couplage fonctionnant dans un seul sens comme un suiveur[4].



**Figure 3.2.** Synchronisation unidirectionnelle.

### 3.2.2 Synchronisation bidirectionnelle

Dans le cas d'un couplage bidirectionnel, l'énergie est transférée d'un système à un autre, à l'aide d'un élément de couplage fonctionnant dans les deux sens comme une résistance[4].



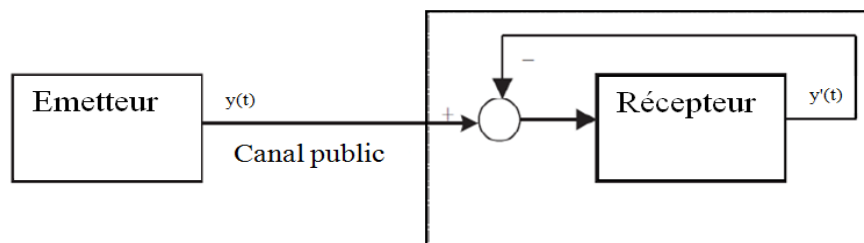
**Figure 3.3.** Synchronisation bidirectionnelle.

### 3.3 Les méthodes de synchronisation

Plusieurs méthodes ont été exposées pour la synchronisation des systèmes chaotiques. Nous citerons quelques une de ces méthodes.

#### 3.3.1 Synchronisation par boucle fermée

L'idée est d'appliquer une correction au système en fonction de l'erreur entre le signal transmis par le premier système et le signal régénéré par le deuxième. Cette erreur est ainsi injectée en contre-réaction [9].

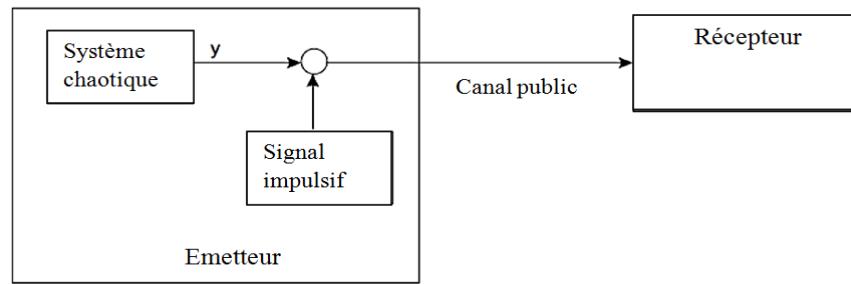


**Figure 3.4.** Synchronisation par boucle fermée.

#### 3.3.2 Synchronisation impulsive

Dans le but de réduire la redondance du signal transmis, la synchronisation impulsive a été proposée.

On définit un signal impulsif qui consiste en une suite d'instants discrets auxquelles un signal est envoyé par le système maître au système esclave, dont les variables d'état subissent un saut et un changement d'état.



**Figure 3.5.** Synchronisation impulsive.

### 3.3.3 Synchronisation adaptative

Dans cette méthode, tous les états du système récepteur doivent se synchroniser avec les états respectifs de l'émetteur. Des techniques spécifiques ont été appliquées pour l'analyse et la caractérisation du comportement des systèmes chaotiques.

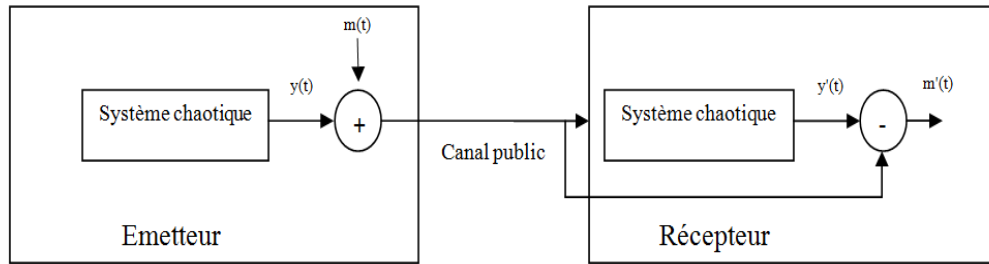
Pour réaliser la synchronisation, il est nécessaire de vérifier la convergence de l'écart  $e_i$  (erreur) entre l'émetteur et le récepteur, et pour y remédier, une solution consiste à mettre en œuvre une condition de stabilité  $k_i$  du système relatif à cet écart, constituant une condition d'unicité de la réponse [9].

## 3.4 Techniques de cryptage par chaos

Il existe plusieurs techniques qui peuvent servir comme moyen de masquage de l'information dans le chaos, nous décrivons ici quelques uns [1] :

### 3.4.1 Cryptage par addition

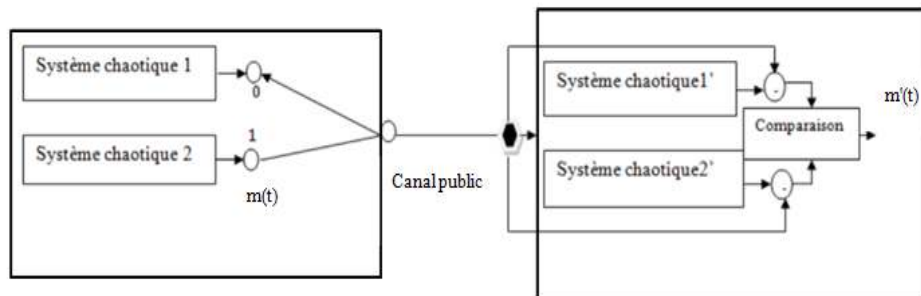
Dans cette méthode appelée masquage chaotique, l'émetteur est un système chaotique dont le signal de sortie  $y(t)$  est ajouté au signal du message  $m(t)$ . La somme de deux signaux est transmise au récepteur à travers le canal de transmission, qui est un canal public. Le récepteur est constitué d'un système chaotique identique à l'émetteur et un simple soustracteur. Ainsi, après la synchronisation des deux systèmes chaotiques (émetteur et récepteur), le message est extrait à l'aide d'une opération de soustraction.



**Figure 3.6.** Principe du cryptage par addition.

### 3.4.2 Cryptage par commutation (CSK)

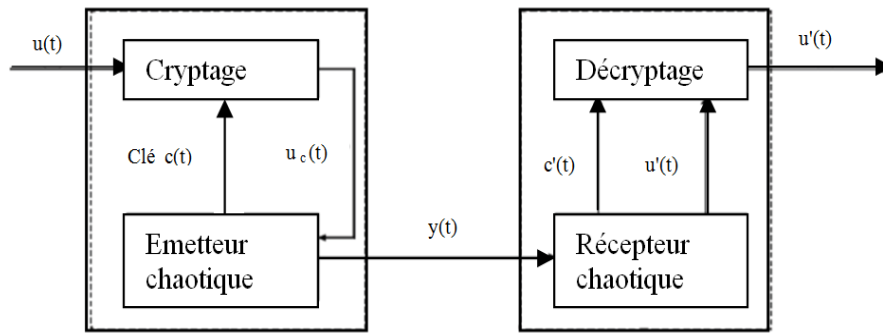
Cette méthode est utilisée pour transmettre un message binaire. L'émetteur est composé de deux systèmes chaotiques et pour chaque niveau de message  $m(t)$  (0 ou 1), l'un des systèmes envoie sa sortie sur la ligne de transmission. Ainsi, le signal transmis commute entre deux attracteurs étranges. Le récepteur est constitué de deux systèmes chaotiques identiques à ceux de l'émetteur et un bloc de comparaison permet de relever la valeur du message noté  $m'(t)$ .



**Figure 3.7.** Principe du cryptage par commutation.

### 3.4.3 Cryptage mixte

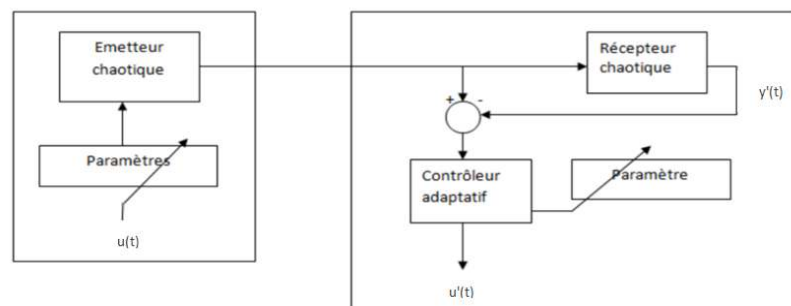
Le message contenant l'information est cryptée grâce à une clé, générée par l'émetteur chaotique. Le message crypté est alors injecté dans la dynamique du système chaotique, pour la rendre plus complexe, Ensuite, un signal fonction des variables d'états de l'émetteur est transmis au récepteur, qui établit une synchronisation avec l'émetteur. La clé est alors reconstituée par le récepteur, qui peut finalement décoder le message.



**Figure 3.8.** Principe du cryptage mixte.

### 3.4.4 Cryptage par modulation paramétrique

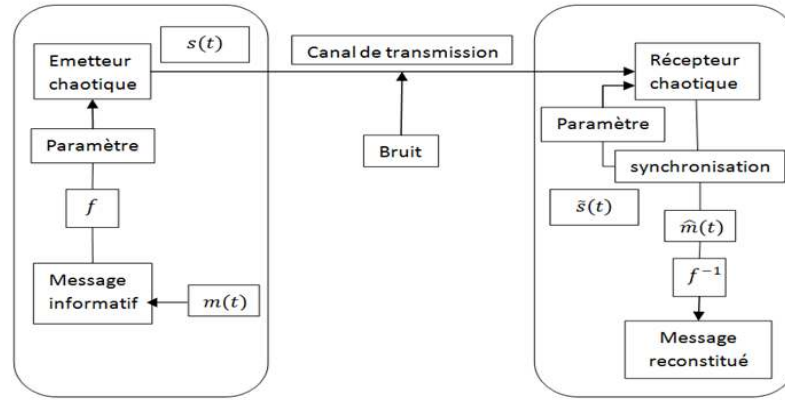
Cette technique utilise le message contenant l'information pour moduler un paramètre de l'émetteur chaotique. Un contrôleur adaptatif est chargé de maintenir la synchronisation au niveau du récepteur, tout en suivant les changements du paramètre modulé. Au niveau de l'émetteur, le fait de moduler un (ou plusieurs) paramètre(s) impose à la trajectoire de changer continûment d'attracteur, et de ce fait, le signal transmis est plus complexe qu'un signal chaotique normal. Cependant, la façon d'injecter le message et donc la fonction de modulation des paramètres ne doivent pas supprimer le caractère chaotique du signal envoyé au récepteur.



**Figure 3.9.** Principe du cryptage par modulation.

## 3.5 Etude de l'émetteur-récepteur chaotique

Le schéma synoptique de la transmission sécurisée basé sur le cryptage paramétrique et la synchronisation adaptative avec un seul paramètre variable  $c$  est représenté par la figure 3.10.



**Figure 3.10.** Schéma synoptique d'une transmission sécurisée.

Le système maître est constitué du système-hyper chaotique de Qi donné par le système d'équations suivants :

$$\begin{cases} \dot{x}_1 = a(x_2 - x_1) + x_2 x_3 \\ \dot{x}_2 = b(x_1 + x_2) - x_1 x_3 \\ \dot{x}_3 = -c x_3 - e x_4 + x_1 x_2 \\ \dot{x}_4 = -d x_4 + f x_3 + x_1 x_2 \end{cases} \quad (3.1)$$

Où  $x_i$  ( $i = 1, 2, 3, 4$ ) représentent les états du système, et  $a, b, c, d, e, f$  sont des paramètres réels.

Les équations suivantes décrivent le système esclave

$$\begin{cases} \dot{y}_1 = a(y_2 - y_1) + y_2 y_3 - k_1 e_1 \\ \dot{y}_2 = b(y_1 + y_2) - y_1 y_3 - k_2 e_2 \\ \dot{y}_3 = -\hat{c} y_3 - e y_4 + y_1 y_2 - k_3 e_3 \\ \dot{y}_4 = -d y_4 + f y_3 + y_1 y_2 - k_4 e_4 \end{cases} \quad (3.2)$$

Où  $y_i$  ( $i = 1, 2, 3, 4$ ) représentent les états,  $k_i$  ( $i = 1, 2, 3, 4$ ) représentent les gains et  $a, b, d, e, f$  sont des paramètres constants connus.

$e_i$  ( $i = 1, 2, 3, 4$ ) représentent les variables d'erreurs et  $\dot{e}_i$  ( $i = 1, 2, 3, 4$ ) sa dynamique.

avec  $e_1 = y_1 - x_1$ ,  $e_2 = y_2 - x_2$ ,  $e_3 = y_3 - x_3$ ,  $e_4 = y_4 - x_4$

et  $\dot{e}_1 = \dot{y}_1 - \dot{x}_1$ ,  $\dot{e}_2 = \dot{y}_2 - \dot{x}_2$ ,  $\dot{e}_3 = \dot{y}_3 - \dot{x}_3$ ,  $\dot{e}_4 = \dot{y}_4 - \dot{x}_4$

$e_c = \hat{c} - c$  où  $\hat{c}$  est la valeur estimée et  $c$  un paramètre constant connu.



Nous obtenons le système d'erreur par la soustraction de l'équation (3.1) avec l'équation (3.2)

$$\begin{cases} \dot{e}_1 = a(e_2 - e_1) + e_2 e_3 + e_2 x_3 + x_2 e_3 - k_1 e_1 \\ \dot{e}_2 = b(e_1 + e_2) - e_1 e_3 - e_1 x_3 - x_1 e_3 - k_2 e_2 \\ \dot{e}_3 = \hat{c} e_c - e e_4 + e_1 e_2 + e_1 x_2 + x_1 e_2 - k_3 e_3 \\ \dot{e}_4 = -d e_4 + f e_3 + e_1 e_2 + e_1 x_2 + x_1 e_2 - k_4 e_4 \end{cases} \quad (3.3)$$

Pour réaliser la synchronisation entre le système Maître (3.1) et le système Esclave (3.2), la dynamique de l'erreur représentée par le système (3.3) doit être asymptotiquement stable, pour que les erreurs tendent vers zéro.

La théorie de stabilité de Lyapunov est utilisée pour prouver le résultat asymptotiquement stable du système d'erreur (3.3).

Nous choisissons la fonction candidate suivante :

$$V(e_1, e_2, e_3, e_4, e_r) = \frac{1}{2} \left( \frac{1}{\alpha} e_1^2 + e_2^2 + e_3^2 + e_4^2 + \frac{1}{\beta} e_r^2 \right) \quad (3.4)$$

Avec  $\alpha, \beta > 0$

La dérivation par rapport à l'erreur dynamique nous donne:

$$\dot{V} = \frac{1}{\alpha} e_1 \dot{e}_1 + e_2 \dot{e}_2 + e_3 \dot{e}_3 + e_4 \dot{e}_4 + \frac{1}{\beta} e_r \dot{e}_r \quad (3.5)$$

$$\begin{aligned} \dot{V} = & e_c \left( \frac{1}{\beta} \dot{e}_c - x_3 e_3 \right) - e_1^2 \left( \frac{k_1}{\alpha} + \frac{a}{\alpha} - 5 \right) - e_2^2 \left( k_2 - b - 1 - \frac{y_4}{4} - \frac{x_4}{4} - \frac{\left( \frac{a}{\alpha} + \frac{y_3}{\alpha} + b - x_3 \right)^2}{4} \right) \\ & - e_3^2 \left( k_3 + \hat{c} - \frac{x_2^2}{4\alpha^2} - \frac{x_2^2}{4} \right) - e_4^2 \left( k_4 + d + \frac{e^2}{4} - \frac{f^2}{4} - \frac{x_2^2}{4} - \frac{x_1^2}{4} \right) \\ & - \left( e_1 - \frac{\left( \frac{a}{\alpha} + b + x_3 + y_4 - x_4 + \frac{y_3}{\alpha} \right)}{2} e_2 \right)^2 \\ & - \left( e_1 \left( \frac{x_2}{2\alpha} + \frac{x_2}{2} \right) e_3 \right)^2 - \left( e_1 - \frac{x_2}{2} e_4 \right)^2 - \left( e_3 - \left( \frac{e}{2} + \frac{f}{2} \right) e_4 \right)^2. \end{aligned} \quad (3.6)$$

$$\text{alors } e_c \left( \frac{1}{\beta} \dot{e}_c - x_3 e_3 \right) = 0 \quad \rightarrow \quad \dot{e}_c = \beta x_3 e_3 \quad (3.7)$$

$$\begin{aligned} \text{et } k_1 &\geq \left( 5 - \frac{a}{\alpha} \right) \alpha, & k_2 &\geq b + 1 + \frac{y_4}{4} + \frac{x_4}{4} + \frac{\left( \frac{a}{\alpha} + \frac{y_3}{\alpha} + b - x_3 \right)^2}{4} \\ k_3 &\geq -\hat{c} + \frac{x_2^2}{4\alpha^2} + \frac{x_2^2}{4}, & k_4 &\geq -d - \frac{e^2}{4} + \frac{f^2}{4} + \frac{x_2^2}{4} + \frac{x_1^2}{4} \end{aligned} \quad (3.8)$$

puis

$$\begin{aligned} \dot{V} \leq & - \left( e_1 - \frac{\left( \frac{a}{\alpha} + b + x_3 + y_4 - x_4 + \frac{y_3}{\alpha} \right)}{2} e_2 \right)^2 - \left( e_1 \left( \frac{x_2}{2\alpha} + \frac{x_2}{2} \right) e_3 \right)^2 \\ & - \left( e_1 - \frac{x_2}{2} e_4 \right)^2 \\ & - \left( e_3 - \left( \frac{e}{2} + \frac{f}{2} \right) e_4 \right)^2 < 0 \end{aligned} \quad (3.9)$$

Selon la théorie de stabilité du Lyapunov, l'équation  $\dot{V} \leq 0$  indique que  $v(t)$  converge vers zéro, soit:

$$e_1, e_2, e_3, e_4 \rightarrow 0 \quad \text{si} \quad t \rightarrow \infty$$

D'après les équations (3.3) et (3.6), on obtient :

$$\dot{e}_c = \beta x_3 e_3 \quad (3.10)$$

et

$$\begin{cases} \dot{e}_1 = a(e_2 - e_1) + e_2 e_3 + e_2 x_3 + x_2 e_3 - k_1 e_1 = 0 \\ \dot{e}_2 = b(e_1 + e_2) - e_1 e_3 - e_1 x_3 - x_1 e_3 - k_2 e_2 = 0 \\ \dot{e}_3 = \hat{c} e_c - e e_4 + e_1 e_2 + e_1 x_2 + x_1 e_2 - k_3 e_3 = 0 \\ \dot{e}_4 = -d e_4 + f e_3 + e_1 e_2 + e_1 x_2 + x_1 e_2 - k_4 e_4 = 0 \end{cases} \quad (3.11)$$

$$e_c \rightarrow 0 \quad \text{si} \quad t \rightarrow \infty \quad \text{et} \quad \dot{c} = 0$$

Par conséquent

$$\dot{\hat{c}} = \beta x_3 e_3 \quad (3.12)$$

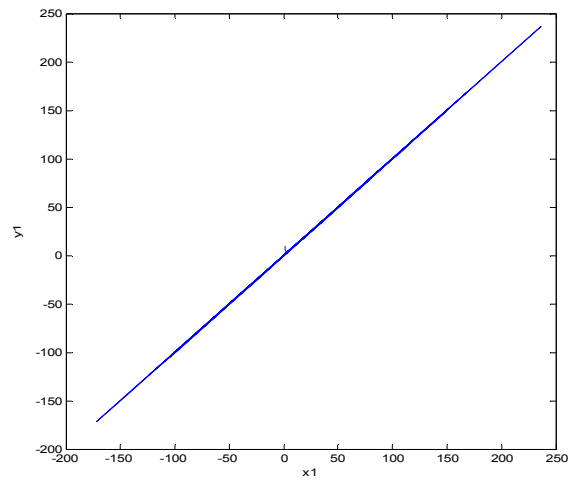
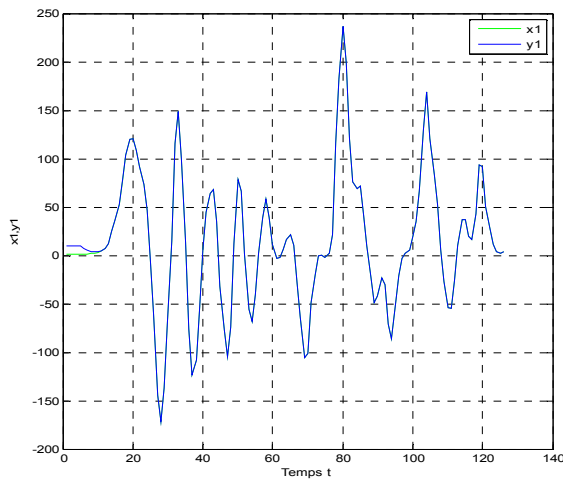
Le message récupéré au niveau du récepteur est représenté par  $\hat{c}$ .

### 3.6 Résultat de la simulation

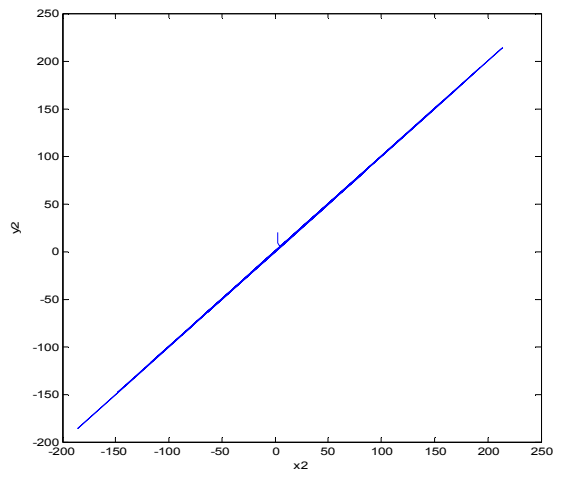
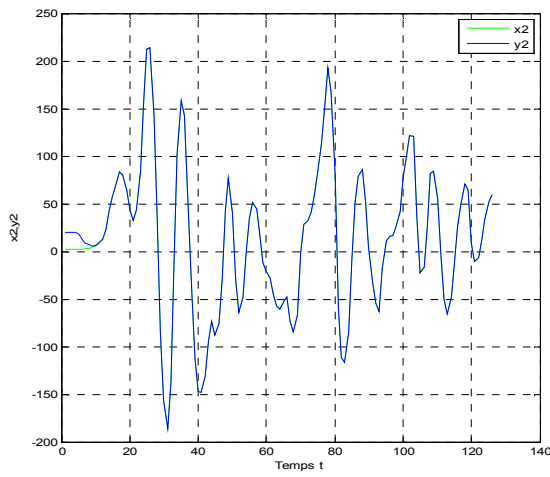
Nous avons utilisé MATLAB Simulink pour réaliser la synchronisation adaptative et le cryptage paramétrique d'un message entre l'émetteur et le récepteur.

Pour la simulation, les paramètres du système sont fixés :  $a=50, b=24, c=13, d=33, e=8, f=30$ , Les conditions initiales choisies du maître sont :  $x_1(0) = 1, x_2(0) = 2, x_3(0) = 3, x_4(0)=4$ , et celles de l'esclave sont :  $y_1(0)= 10, y_2(0) = 20, y_3(0)= 15, y_4(0)= 14$  et les gains  $k_i$  ( $k_1, k_2, k_3, k_4 = 150$ ) avec  $\beta = 10$ .

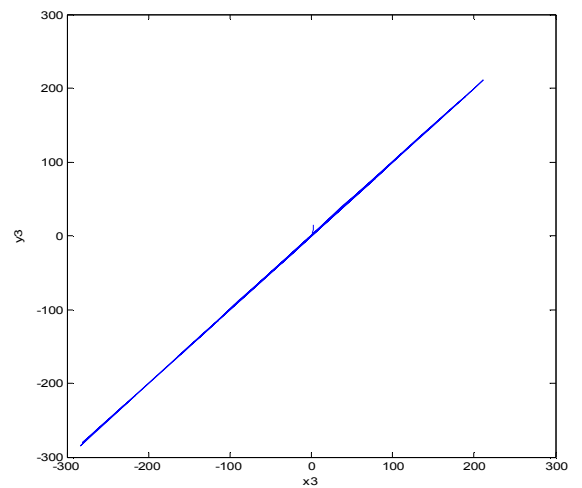
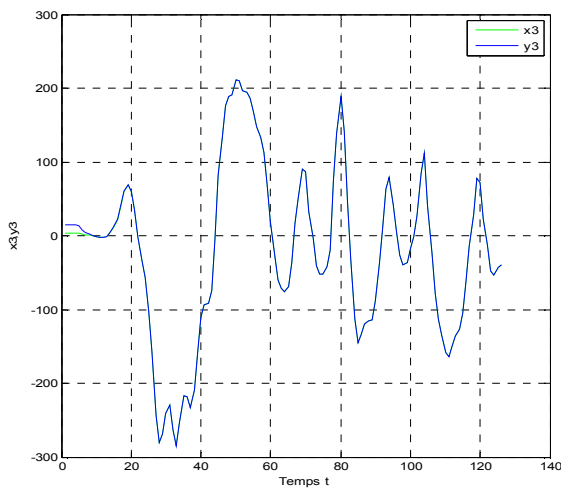
Les figures 3.11 à 3.15 montrent les réponses temporelles, la synchronisation entre l'émetteur (3.1) et le récepteur (3.2) ainsi que les erreurs de synchronisation.



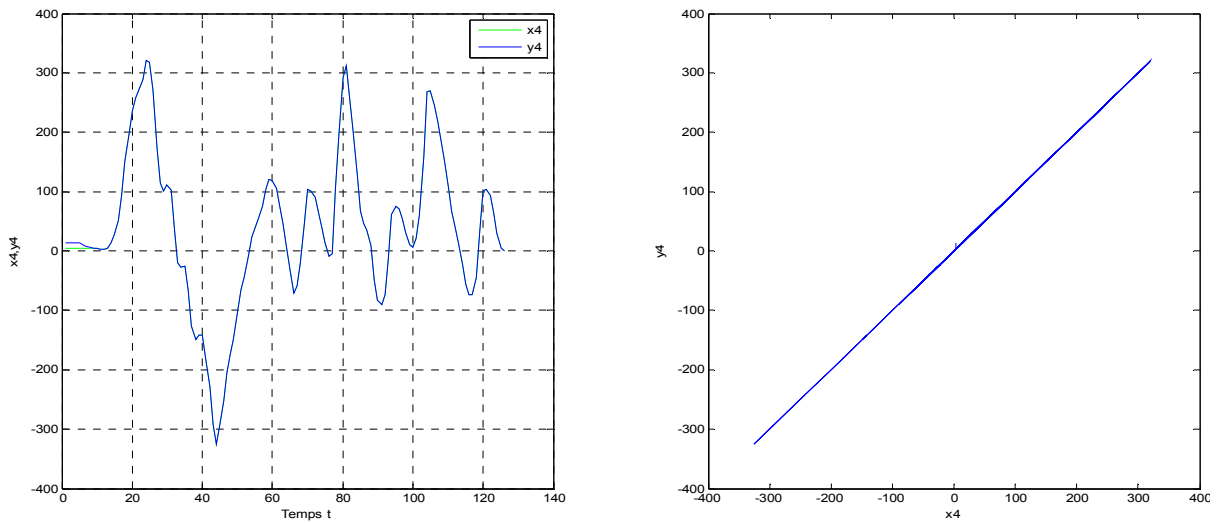
**Figure 3.11.** Synchronisation des signaux  $x_1$  et  $y_1$ .



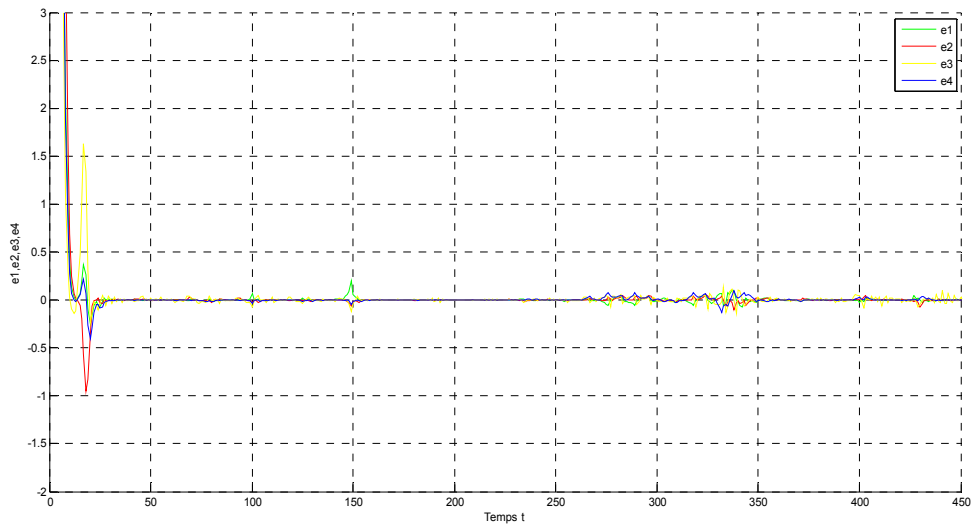
**Figure 3.12.** Synchronisation des signaux  $x_2$  et  $y_2$ .



**Figure 3.13.** Synchronisation des signaux  $x_3$  et  $y_3$ .



**Figure 3.14.** Synchronisation des signaux  $x_4$  et  $y_4$ .

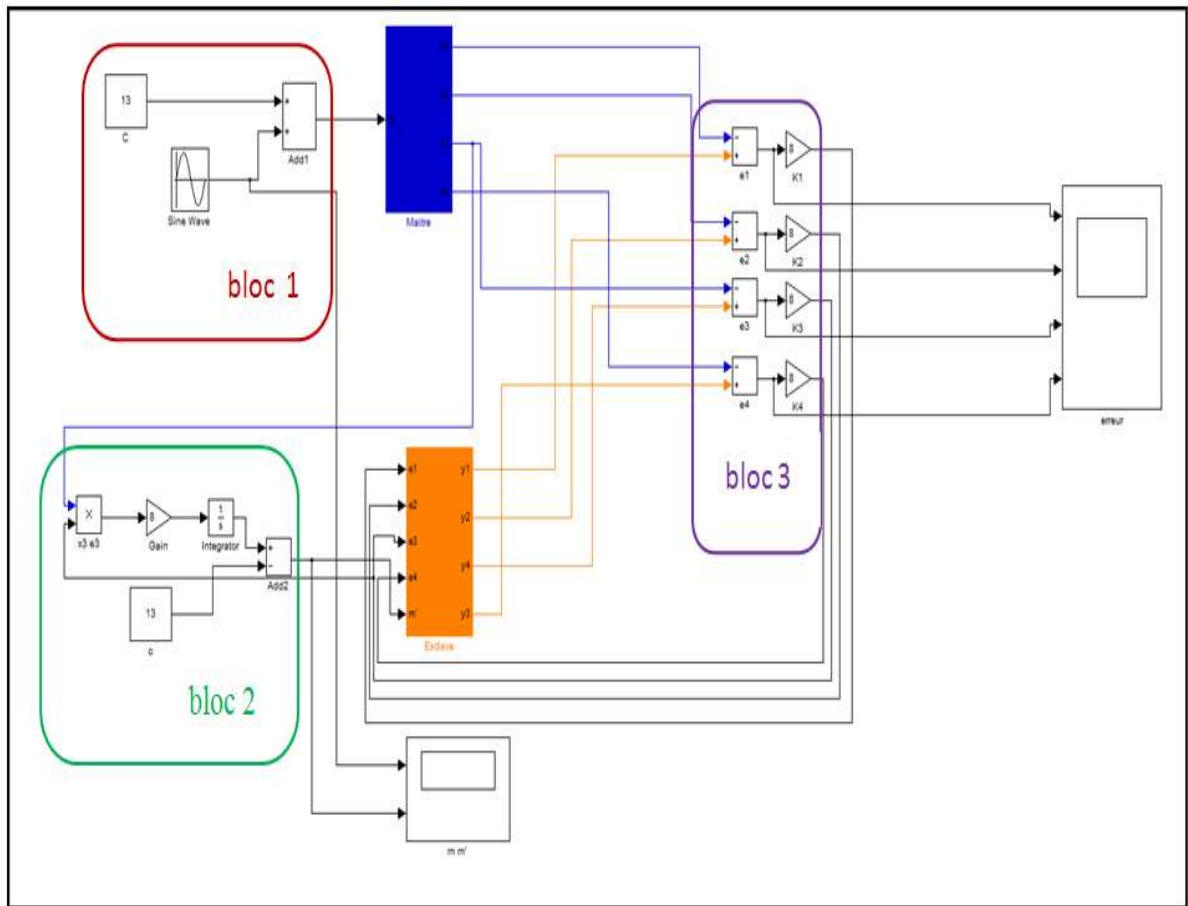


**Figure 3.15.** Les erreurs de synchronisation  $e_1, e_2, e_3$  et  $e_4$ .

À travers ces figures, nous constatons que les erreurs s'annulent après un certain temps, et les signaux coïncident, donc ils sont synchronisés.

### ❖ Transmission d'un signal sinusoïdale

Nous injectons une sinusoïde (message) d'amplitude  $A=10$  et de fréquence  $F=1/2\pi$  dans le paramètre  $c$  au niveau de l'émetteur, le schéma de synchronisation est représenté par la figure 3.16.



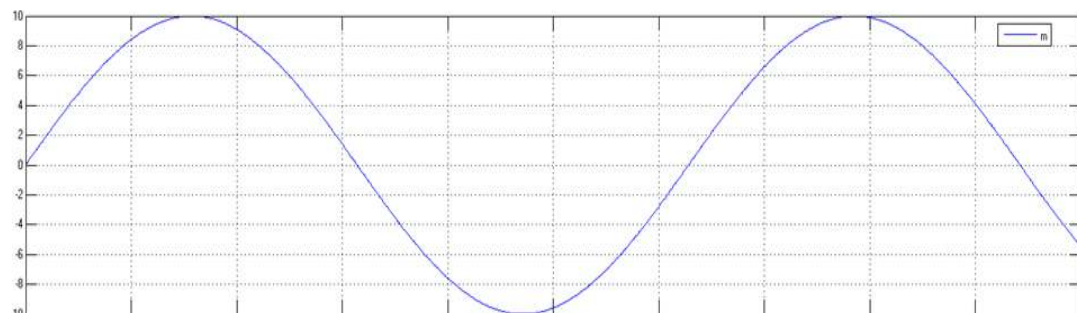
**Figure 3.16.** Schéma de transmission d'une sinusoïde sous MATLAB.

**bloc 1:** cryptage du message informatif (sinusoïde).

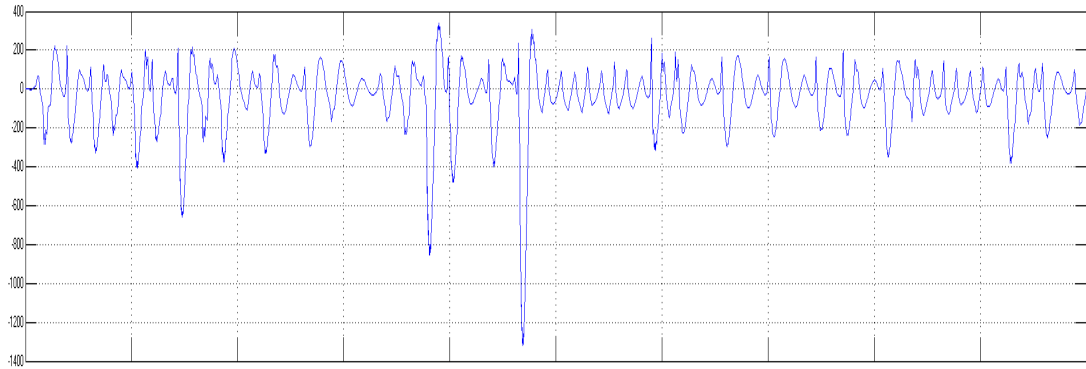
**bloc 2:** décryptage du message informatif (sinusoïde).

**bloc 3:** synchronisation adaptative.

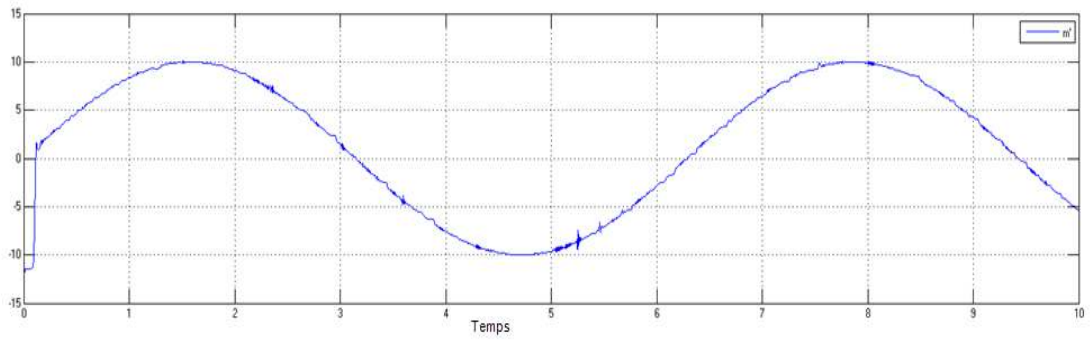
Pour étudier l'influence de bruit sur la qualité de message récupéré au niveau du récepteur nous ajoutons un bruit blanc (B) et à chaque fois nous augmentons sa puissance et nous visualisons les signaux résultants en comparant avec les signaux précédents. Les figures 3.17 à 3.20 représentent les résultats obtenus.



**a.** Message émis.

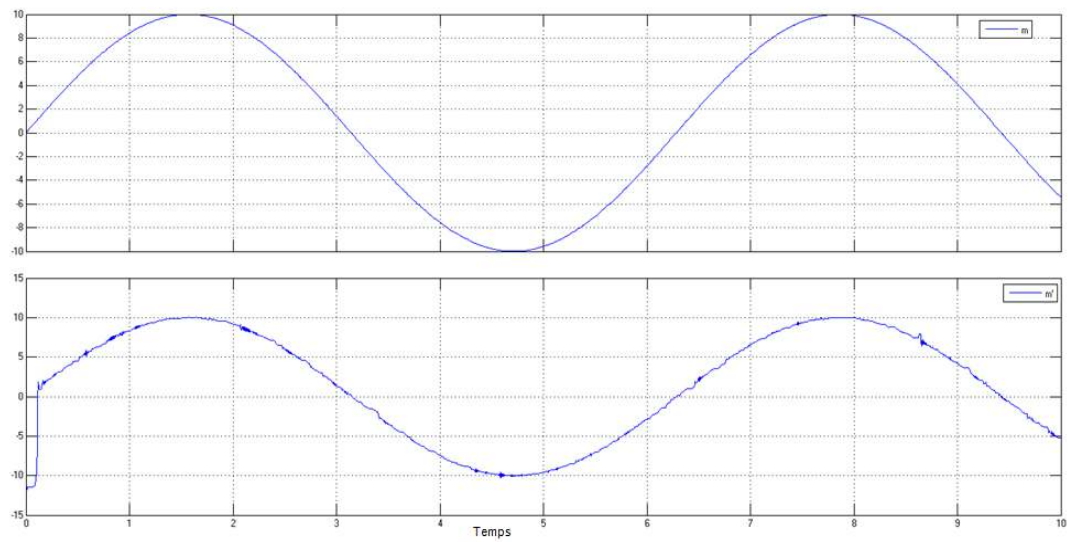


**b.** Message crypté.

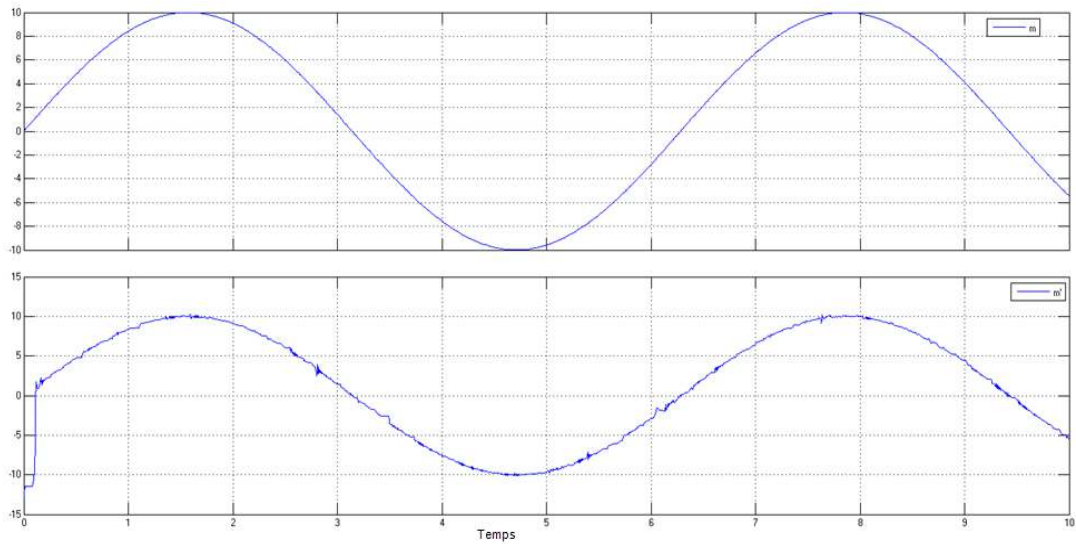


**c.** Message reçu.

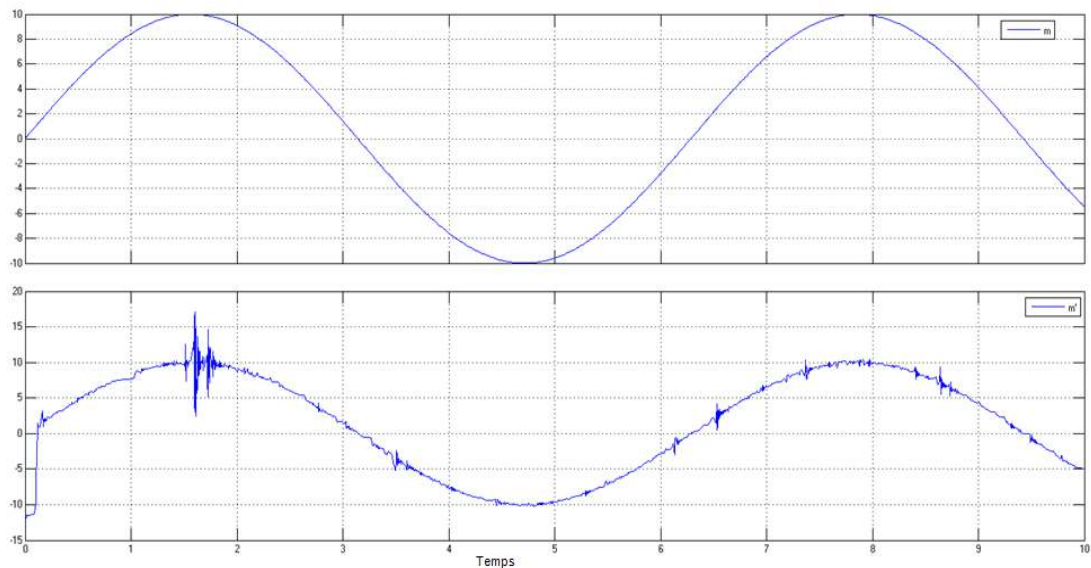
**Figure 3.17.** Message émis, crypté et reçu sans bruit.



**Figure 3.18.** Message émis et reçu avec bruit ( $B = 1$ ).



**Figure 3.19.** Message émis et reçu avec bruit ( $B = 10$ ).



**Figure 3.20.** Message émis et reçu avec bruit ( $B=100$ ).

Nous remarquons que plus le bruit augmente, plus le signal est distordu.

### ❖ Transmission d'une image

Dans cette partie, nous remplaçons la sinusoïde par une image en noir et blanc (message) de camera man (image Matlab) qui est définie comme une matrice de 256 lignes et 256 colonnes, qui sera converti en un vecteur de 65535 pixels, puis nous effectuons la même méthode d'analyse faite avec le message précédent (sinusoïde). Le schéma sous MATLAB est représenté sur la figure 3.21.

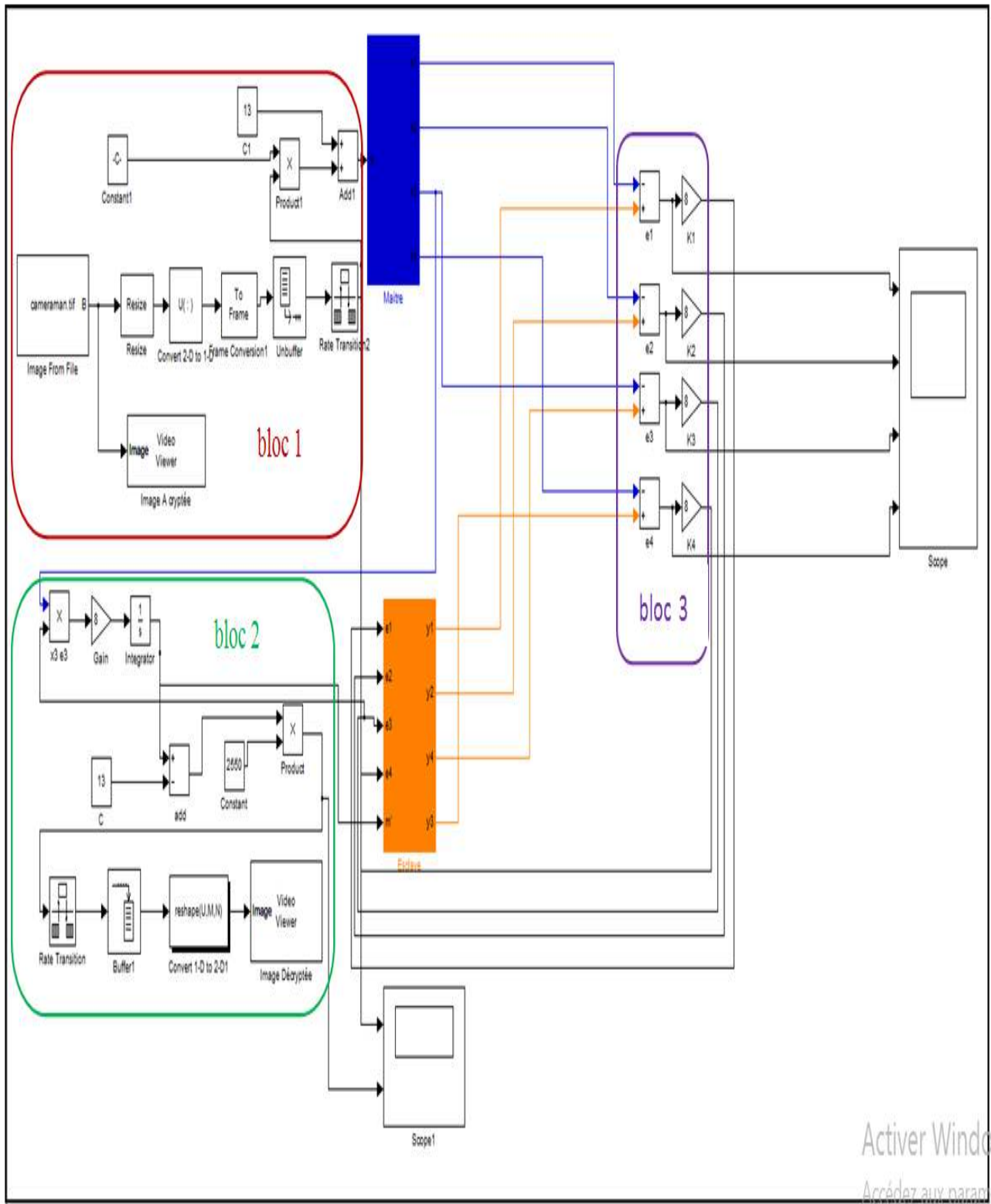


Figure 3.21. Schéma de transmission d'une image sous MATLAB.

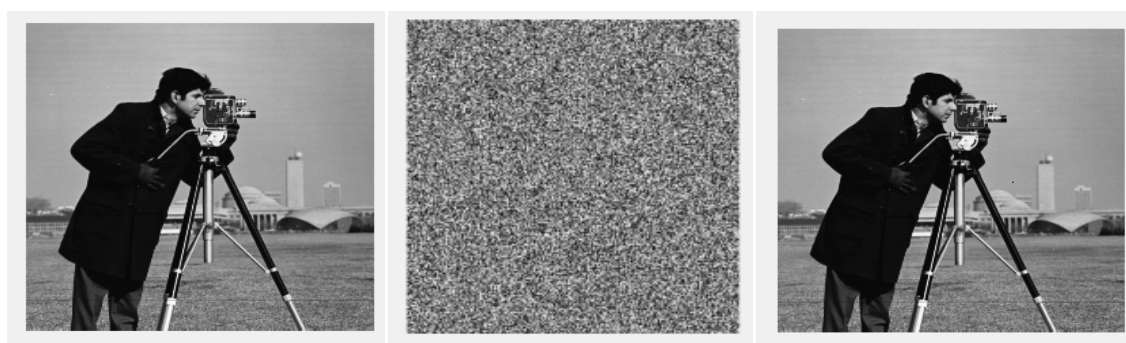
**bloc 1:** cryptage du message informatif (image).

**bloc 2:** décryptage du message informatif (image).

**bloc 3:** synchronisation adaptative.



Les résultats de simulation représentant l'image originale, cryptée et décryptée sans et avec bruit sont illustrés par les figures 3.22 à 3.24.

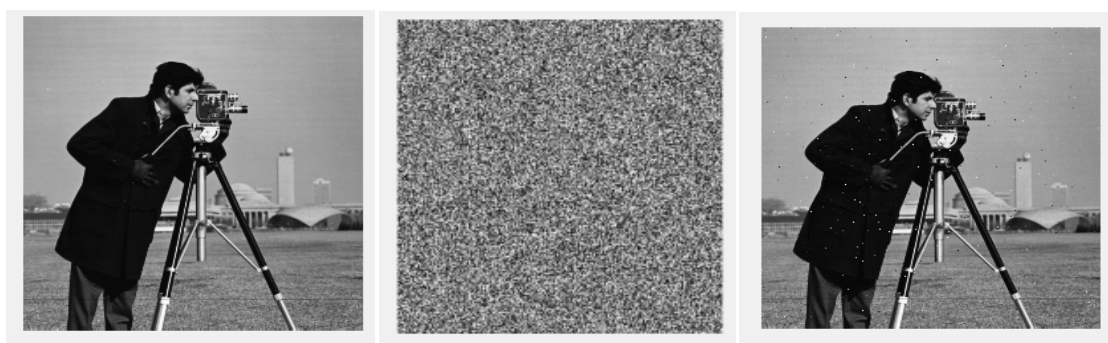


**a.** Image à crypter.

**b.** Image cryptée.

**c.** Image décryptée.

**Figure 3.22.** Récupération de l'image sans bruit.

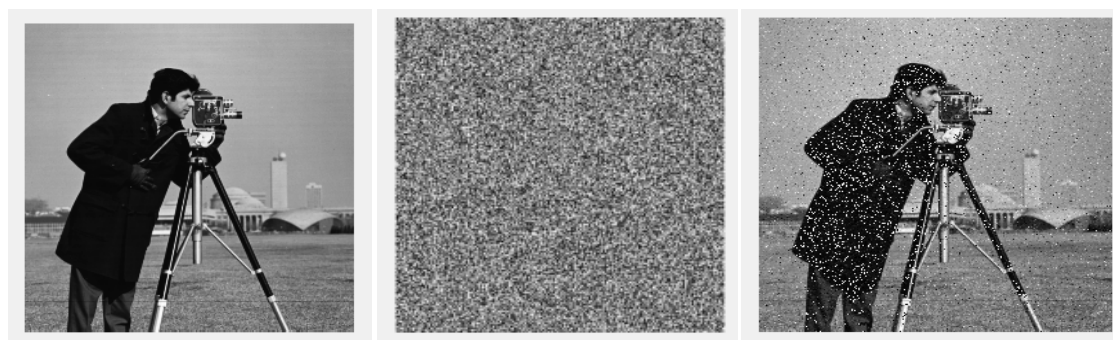


**a.** Image à crypter.

**b.** Image cryptée.

**c.** Image décryptée.

**Figure 3.23.** Récupération de l'image avec bruit ( $B=0.000001$ ).



**a.** Image à crypter.

**b.** Image cryptée.

**c.** Image décryptée.

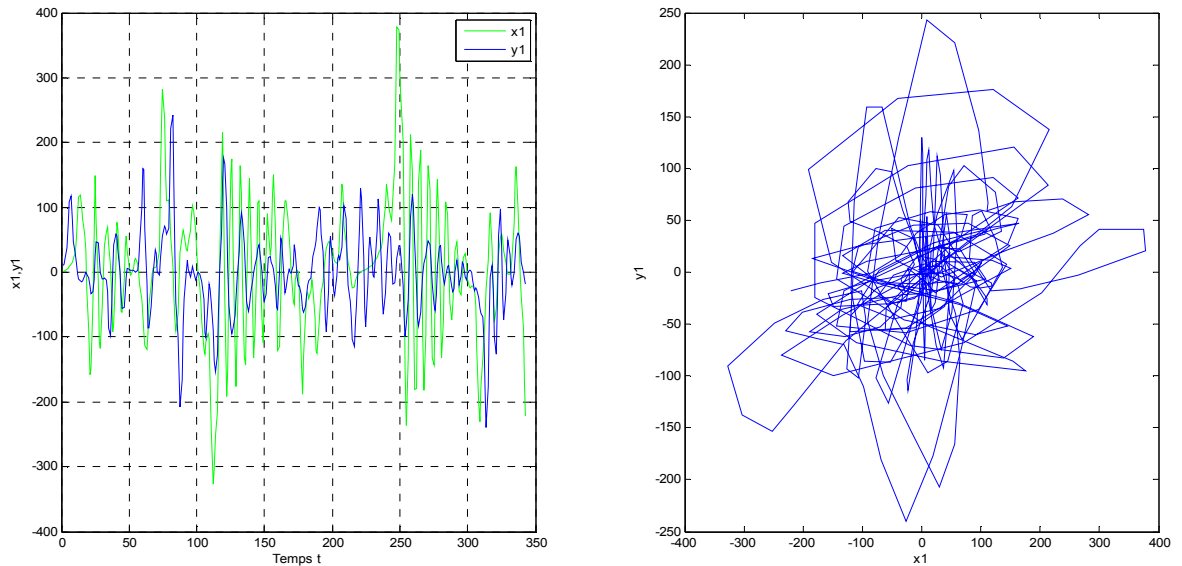
**Figure 3.24.** Récupération de l'image avec bruit ( $B=0.0001$ ).

Les figures précédentes montrent la synchronisation entre les deux systèmes et l'effet de bruit; ainsi plus nous augmentons le bruit, plus l'image est dégradée.

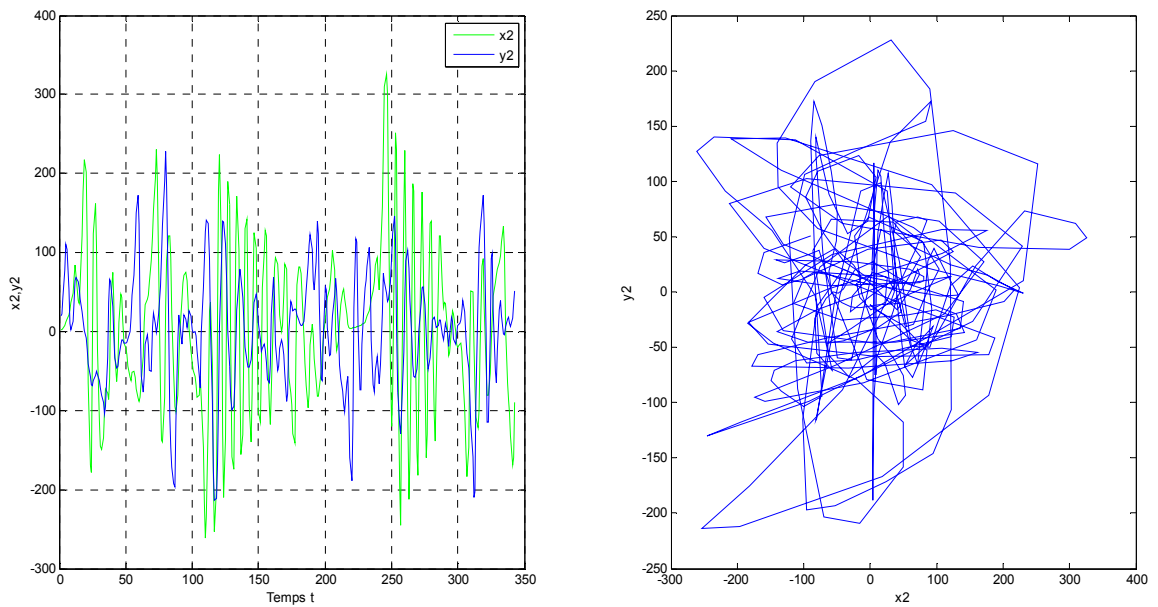
### 3.7 Pert de la synchronisation

Nous avons désynchronisé les systèmes par le non respect des conditions de synchronisation; nous avons choisis  $k_1=50$ ,  $k_2=-10$ ,  $k_3=3$ ,  $k_4=10$  et  $\beta=0$ .

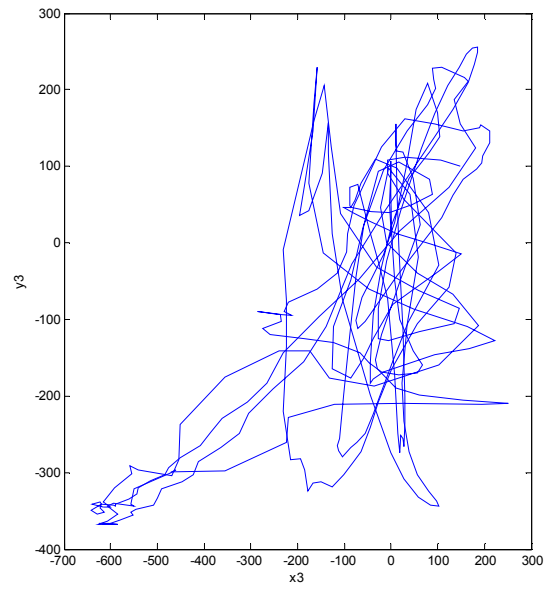
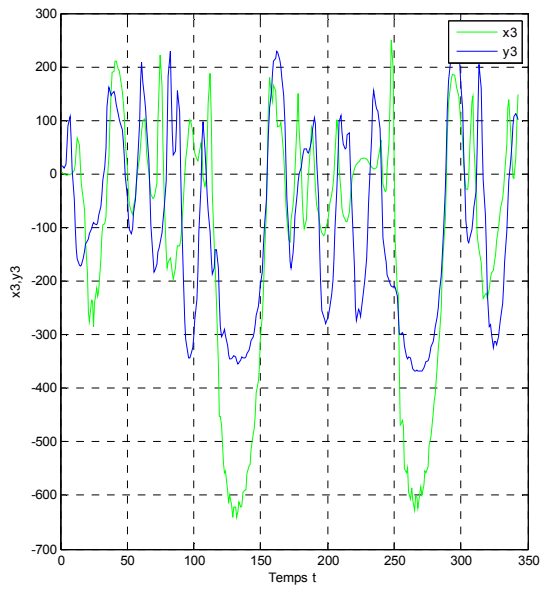
Les figures 3.25 à 3.29 montrent les réponses temporelles, la désynchronisation entre l'émetteur (3.1) et de son récepteur (3.2) ainsi que les erreurs de désynchronisation.



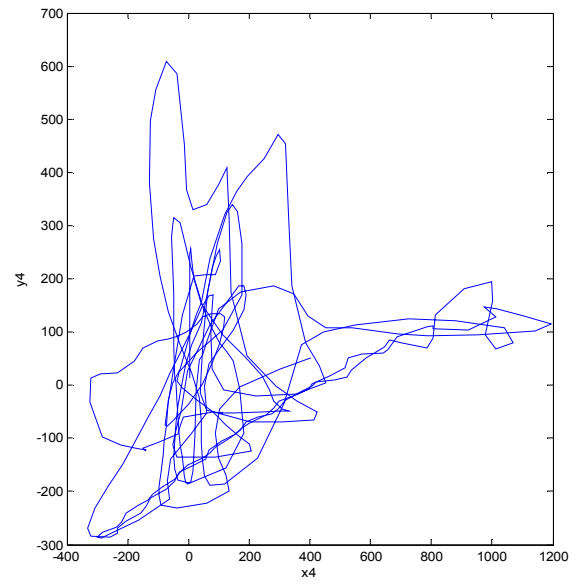
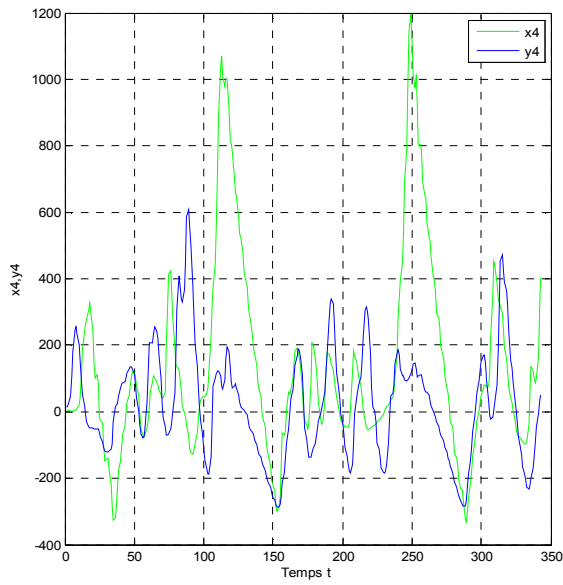
**Figure 3.25.** Désynchronisation des signaux  $x_1$  et  $y_1$ .



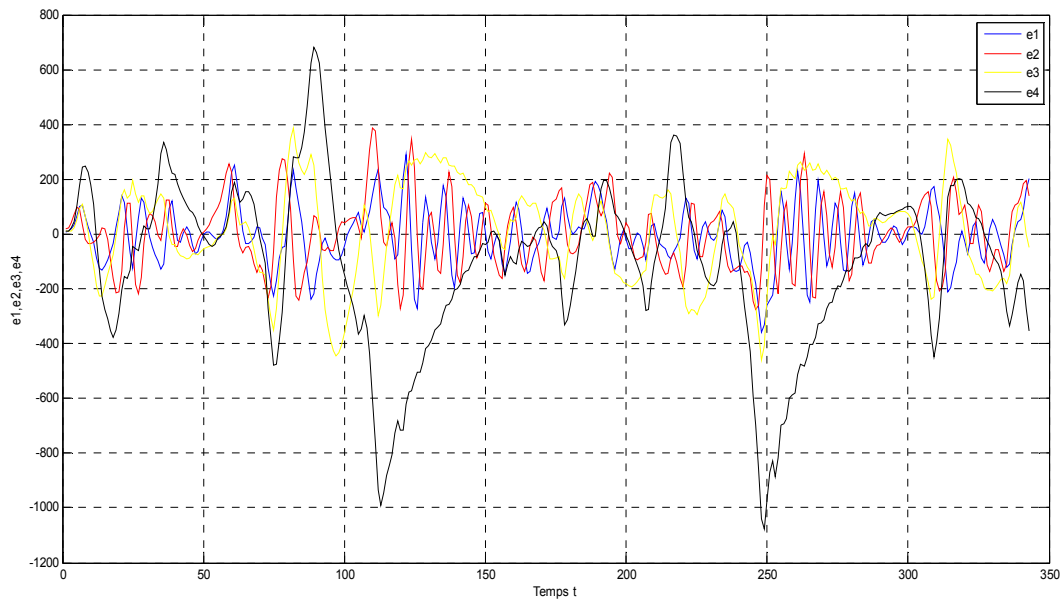
**Figure 3.26.** Désynchronisation des signaux  $x_2$  et  $y_2$ .



**Figure 3.27.** Désynchronisation des signaux  $x_3$  et  $y_3$ .



**Figure 3.28.** Désynchronisation des signaux  $x_4$  et  $y_4$ .



**Figure 3.29.** Les erreurs de désynchronisation  $e_1, e_2, e_3$  et  $e_4$ .

Les figures précédentes montrent la désynchronisation du système lorsque nous ne respectons pas les conditions de la synchronisation.

### 3.8 Conclusion

Dans ce chapitre nous avons présenté les différentes classes et les principales méthodes de synchronisation, ainsi que les différentes techniques de cryptage. L'étude de l'émetteur et du récepteur basés sur le système hyper-chaotique de Qi ont été décrits et utilisés dans le système de communication sécurisés par cryptage paramétrique et synchronisation adaptative.

Nous avons calculé les conditions de synchronisations, et ajouté un bruit pour étudier son influence proportionnelle sur la perturbation des différents signaux (sinusoïde, image) reconstitués, et dépend de leur puissance.

Le système chaotique utilise le concept de synchronisation afin d'augmenter la sécurité de la communication.

# Chapitre 4 Implémentation FPGA du système chaotique de Qi

---

## 4.1 Introduction

Il y a quelques années, la réalisation d'un montage en électronique numérique impliquait l'utilisation d'un nombre important de circuits intégrés logiques. Ceci a donc donné naissance aux circuits logiques programmables qui réalisent plusieurs fonctions logiques dans un seul circuit.

Les circuits FPGA (**F**ield **P**rogrammable **G**ate **A**rray) sont ainsi une famille de circuits reconfigurables constitués de composants ou entités à architecture modifiable afin de répondre à un objectif bien déterminé, pouvant être reprogrammés à volonté avec grande performance.

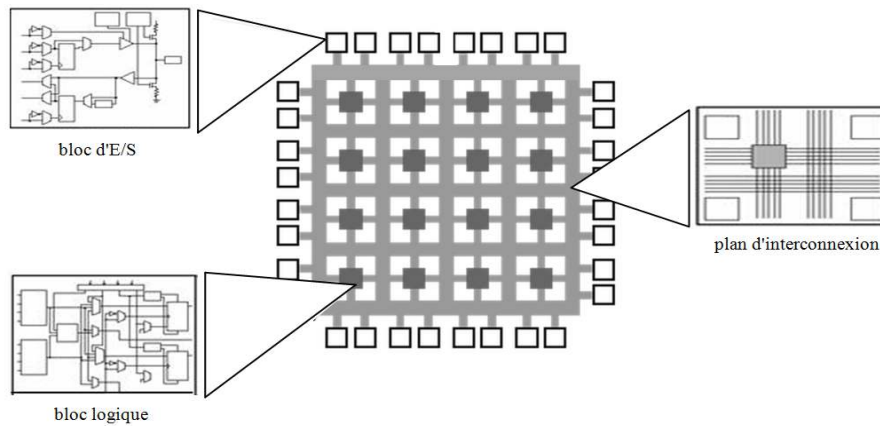
Dans ce chapitre, nous allons présenter les circuits FPGA, puis l'environnement logiciel ISE de la société Xilinx, qui est utilisé à la fois pour la conception et pour l'implantation sur puce. Les simulations comportementales VHDL comme langage de programmation matériel sont généralement réalisées avec l'outil ModelSim pour la simulation et la vérification des architectures.

## 4.2 Présentation des circuits FPGA

Le circuit FPGA est le circuit logique reprogrammable le plus couramment utilisé de nos jours, de part ses capacités, sa vitesse et sa grande flexibilité. Ces architectures mixent généralement des composants matériels et logiciels travaillant en concurrence afin de répondre le plus efficacement à la contrainte temporelle imposée [13].

### 4.2.1 Architecture des FPGA

Les circuits FPGA sont constitués d'une matrice de blocs logiques programmables CLB (**C**onfigurables **L**ogic **B**loc) entourés des blocs d'entrée et de sortie programmable IOB. L'ensemble est relié par un réseau d'interconnexions programmable (figure 4.1).



**Figure 4.1.** Architecture générique d'un circuit FPGA.

### 4.2.2 Technologies de programmation

Les différentes technologies de programmation utilisés sont :

- \* **La technologie SRAM** : Cette technologie permet d'avoir une reconfiguration rapide des FPGA. Son inconvénient est qu'elle nécessite beaucoup de place et il est nécessaire de sauvegarder le design dans une autre mémoire Flash

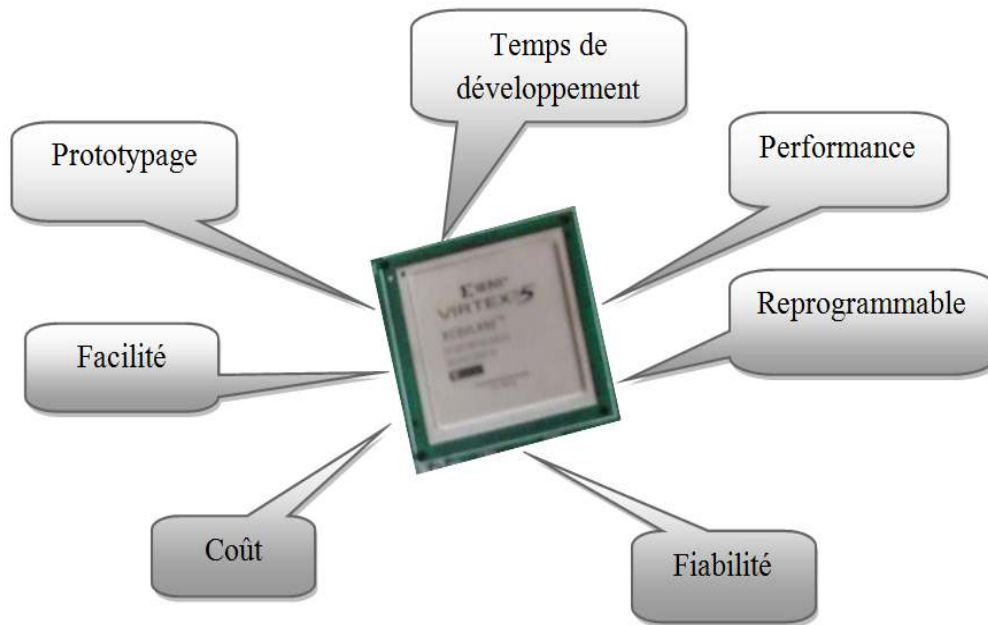
- \* **La technologie EPROM** : Cette technologie peut être effacé et reprogrammer par une source ultra violette, elle est en cours de disparition au profit de l'EEPROM.

- \* **La technologie EEPROM** : Cette technologie présente l'avantage de pouvoir être reprogrammer électriquement par rapport à la technologie EPROM.

- \* **La technologie FLASH** : Cette technologie garde sa configuration mais un nombre limité avec une configuration plus lente par rapport à SRAM.

### 4.2.3 Critères de choix de la carte FPGA

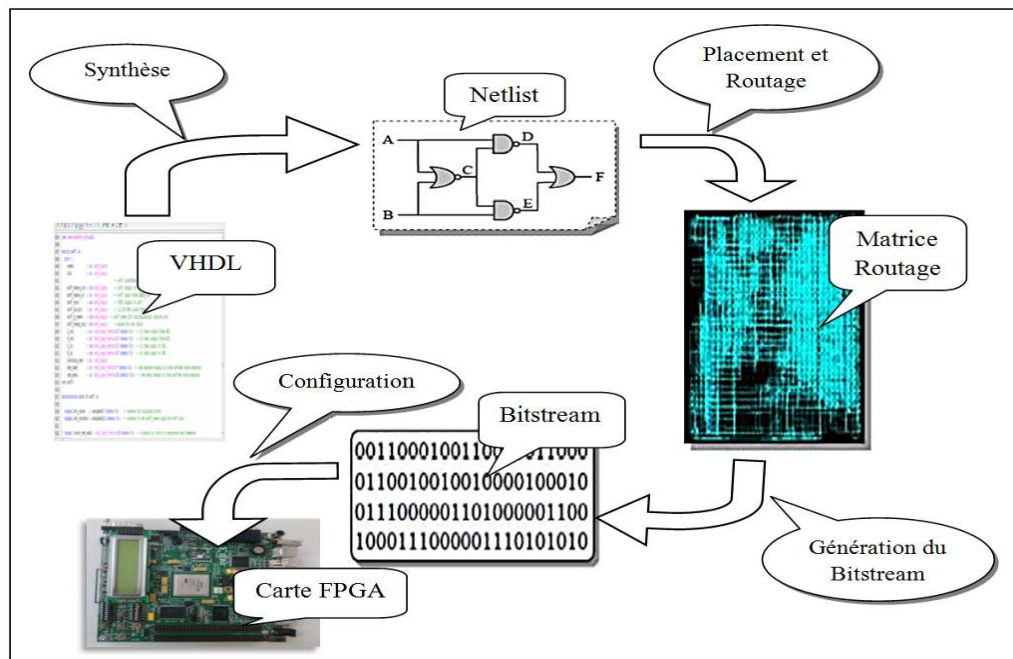
Avant de choisir l'utilisation de la carte FPGA il faut connaître ses critères de choix qui sont regroupés dans la figure 4.2 ci-dessous



*Figure 4.2.* Critères de choix de la carte FPGA.

### 4.3 Processus d'implémentation

Pour accélérer la conception de circuits très complexes et représenter les systèmes numériques, on a utilisé un langage HDL (Hardware Description Language) : c'est un langage permettant la description d'un circuit électronique purement numérique où les plus utilisés en industrie sont le VHDL et le Verilog. Le langage utilisé dans notre projet est le VHDL pour la synthèse et la simulation permettant ainsi de traduire la description matérielle dans un fichier de configuration pour FPGA.



**Figure 4.3.** Etapes de conception sur FPGA.

La figure 4.3 résume les différentes étapes de programmation d'un FPGA. Le synthétiseur des outils CAO (**C**onception **A**ssistée par **O**rdinateur) génère dans un premier temps un Netlist qui décrit la connectivité de l'architecture. Puis l'outil de placement-routage place de façon optimale tous les composants et effectue le routage entre les différentes cellules logiques. Ces deux étapes permettent de générer un fichier de configuration à télécharger dans la mémoire de configuration du FPGA. Ce fichier est appelé bitstream et peut être directement chargé sur FPGA à partir d'un ordinateur [12].

Pour générer le programme de description VHDL de notre système, nous avons utilisé le logiciel System Generator sous Simulink-Matlab qui nous permet d'obtenir, après une bonne configuration, des résultats similaires à ceux obtenus sous Simulink. Puis, nous importons ce fichier dans l'environnement de travail ISE de Xilinx pour que l'implémentation sur FPGA puisse être effectuée, en passant d'abord par des étapes de vérifications assurant son bon fonctionnement à l'aide de l'outil de simulation ModelSim.



## 4.4 Présentation des outils logiciels de travail

### 4.4.1 Présentation du logiciel ISE

ISE (Integrated Software Environment) est un environnement intégré de développement des systèmes numériques qui regroupent tous les outils nécessaires à la conception, la simulation et l'implémentation d'un projet ainsi qu'à la configuration de la carte. Le Navigateur de projet ISE sera utilisé comme outil de conception dans notre travail. Cet outil de XILINX permet de créer des projets comportant plusieurs types de fichier (HDL, UCF), de compiler, de déterminer l'emplacement des broches, et de créer des bancs d'essai de simulation (testbench) [12].

La figure 4.4 représente l'interface Project Navigator de ISE 14.2 permettant de réaliser les processus d'implémentations

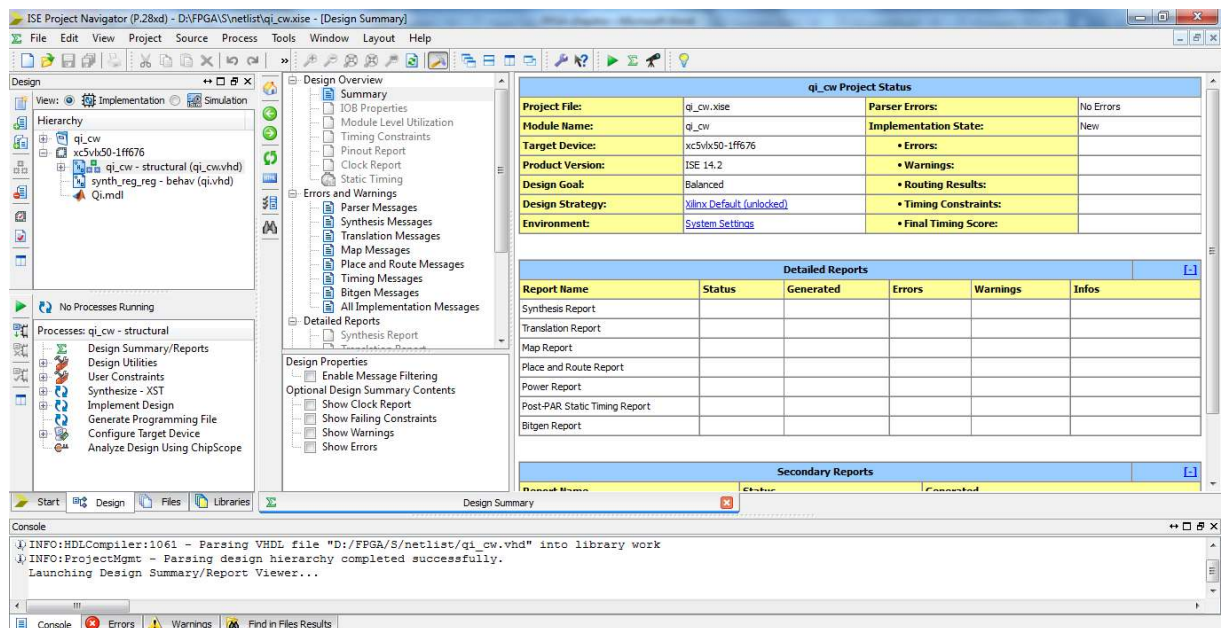


Figure 4.4. Interface Project Navigator ISE 14.2.

### 4.4.2 Présentation du Co-simulateur System Generator

System Generator est un outil qui permet d'utiliser l'environnement Matlab-Simulink pour la conception des applications sur circuits FPGA, c'est une interface entre MATLAB-Simulink et ISE –XILINX [10]. Ses principales tâches sont :

- Conception et simulation des systèmes dans un environnement graphique (Simulink).
- Génération automatique du code VHDL ou Verilog.

- Co-simulation logicielle (Simulink) matérielle (FPGA) par communication JTAG ou USB.

### 4.4.3 Présentation de ModelSim de Mentor Graphics

ModelSim est un outil de simulation HDL de Mentor Graphics. Il peut être intégré au flot de conception Xilinx. C'est un simulateur mixte combinant le VHDL et le Verilog. Les produits ModelSim ont une architecture unique pour des compilations et des simulations rapides.

La figure 4.5 représente l'interface de ModelSim de Mentor Graphics permettant la simulation des signaux.

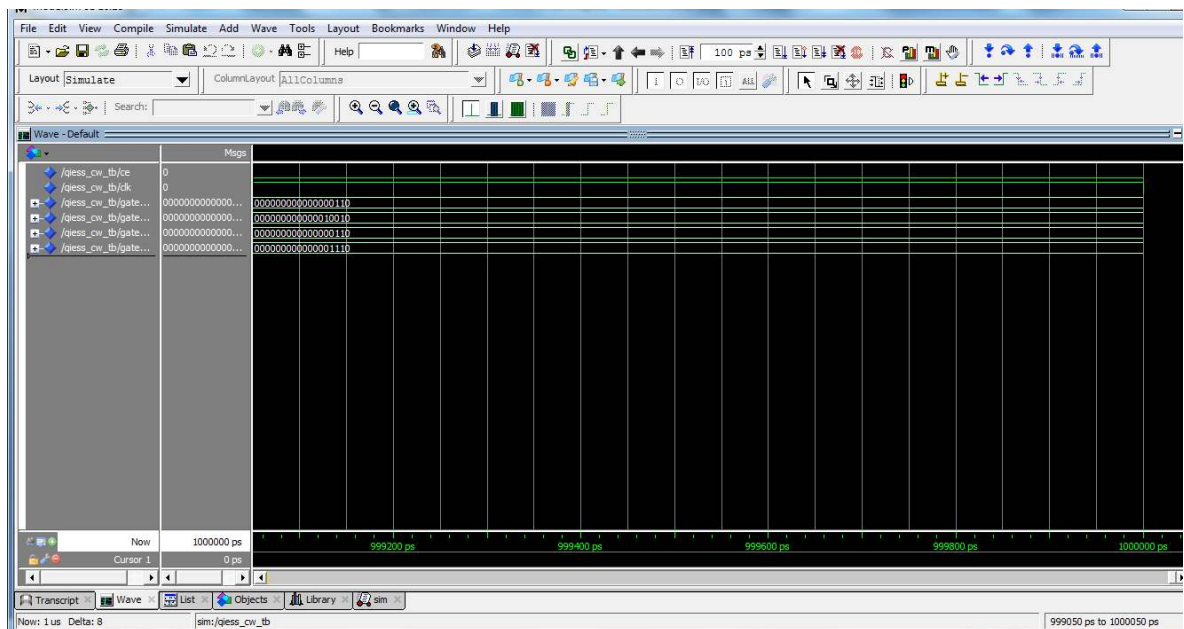


Figure 4.5. Interface ModelSim de Mentor Graphics.

## 4.5 Réalisation expérimentale de l'implémentation

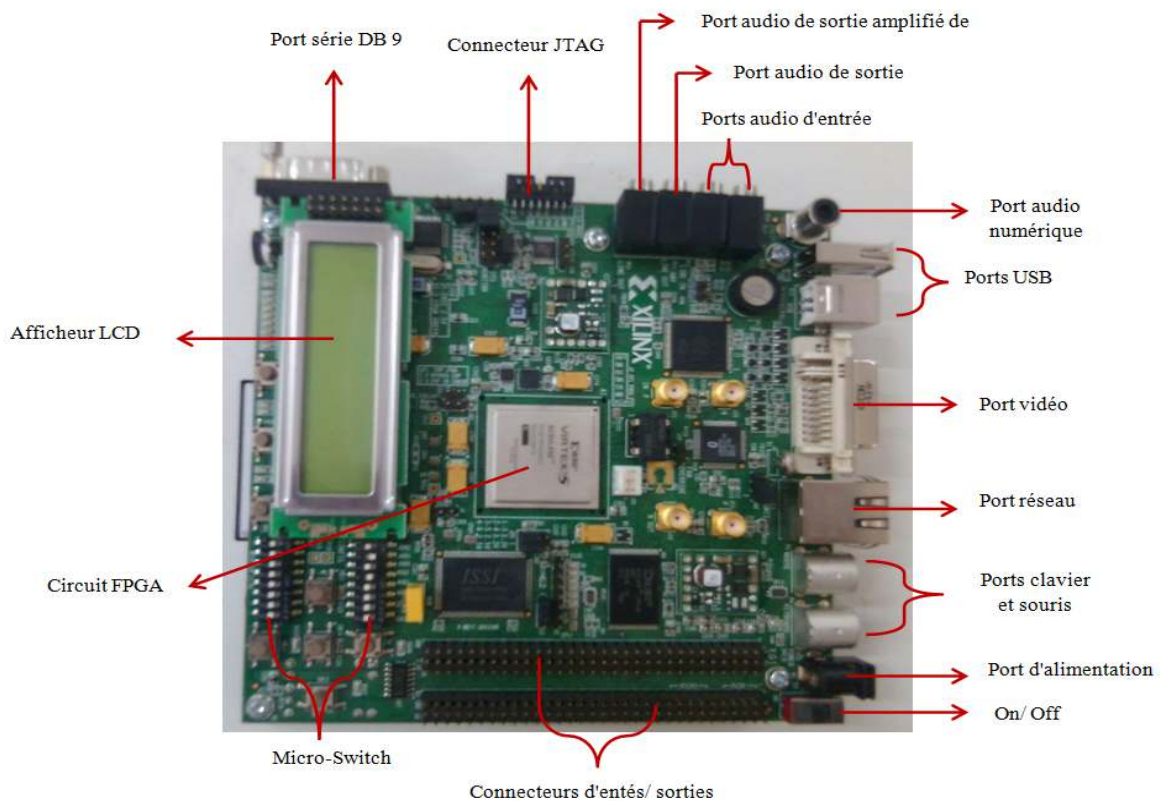
### 4.5.1 Plate forme de développement ML501-Virtex 5

La carte FPGA de développement ML501 a plusieurs caractéristiques et périphériques dont nous pouvons citer :

- FPGA : Virtex-5 XC5VLX50-1FFG676.
- Puce de générateur d'horloge programmable du système.
- Horloge de fréquence 100 MHz.

- Commutateurs DIP à usage général, voyants et boutons poussoirs.
- Codec audio stéréo AC97 avec line-in, line-out, 50 mW casque, entrée microphone, et SPDIF prises audio numérique.
- Port série RS-232.
- Ecran LCD 2 lignes x 16 caractères.
- Connecteur vidéo DVI (VGA pris en charge avec l'adaptateur fourni).
- Connecteurs de souris et clavier PS / 2.
- Contrôleur de configuration système ACE™ avec CompactFlash Type I CompactFlash connecteur.
- SRAM synchrone ZBT, 9 Mb sur le bus de données 32 bits avec quatre bits de parité
- Intel P30 StrataFlash® linéaires puces de mémoire flash (32 Mo).
- Serial Peripheral Interface™ (SPI) Flash (2 Mo).
- Connecteur RJ45 Ethernet.
- Puce d'interface USB avec ports hôtes et périphériques.

Les figures 4.6 et 4.7 représente la carte ML501 Virtex-5 utilisé dans le cadre de notre projet.



**Figure 4.6.** La carte ML501 Virtex-5 (Vue de dessus).

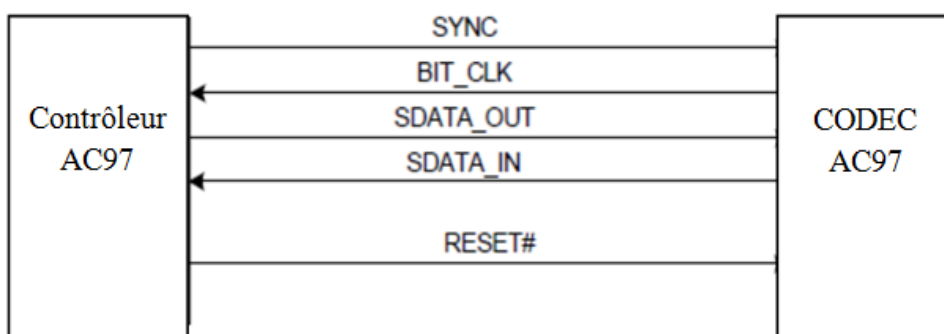


**Figure 4.7.** La carte ML501 Virtex-5 (Vue d'en dessous).

#### 4.5.2 Codec AC97

Le codec audio stéréo AC'97 est une norme développée par Intel Architecteur Labs en 1997. La norme était utilisée dans les cartes mères, les modems et les cartes son. Le codec utilisé dans cette implémentation est constitué d'une part d'un contrôleur AC97 intégré dans la carte FPGA qui joue le rôle d'interface entre deux entités (codec audio 97 et le circuit FPGA) et gère leur intercommunication en dirigeant le flux des données entre ces deux entités, et d'autre part d'un convertisseur CAN et d'un convertisseur CNA pour convertir les signaux d'entrées analogiques en signaux numériques pour leur traitement par le circuit FPGA puis leur reconversion analogique pour l'envoi des signaux vers la sortie.

La connexion entre le codec et le contrôleur est illustré sur la figure 4.8.



**Figure 4.8.** Connexions du contrôleur au CODEC AC97.

Le CODEC AC97 communique avec son contrôleur numérique via 5 fils qui consistent en : un signal d'horloge (BIT\_CLK), une rame de synchronisation (SYNC), des données d'entrée (SDATA\_IN), des données sorties (SDATA\_OUT) et une remise à zéro (RESET#) [13].

Le contrôleur AC97 sous Simulink System-Generator et ses principales fonctions de réglage sont représentés sur la figure 4.9.

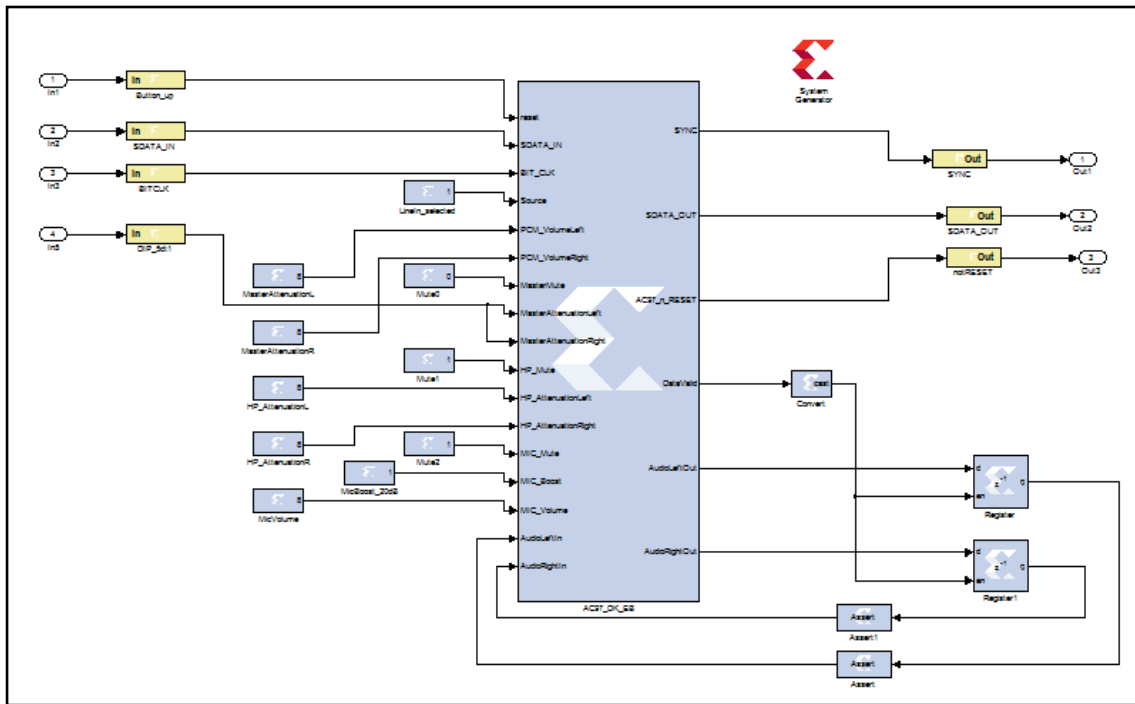


Figure 4.9. Contrôleur AC97 sous System Generator.

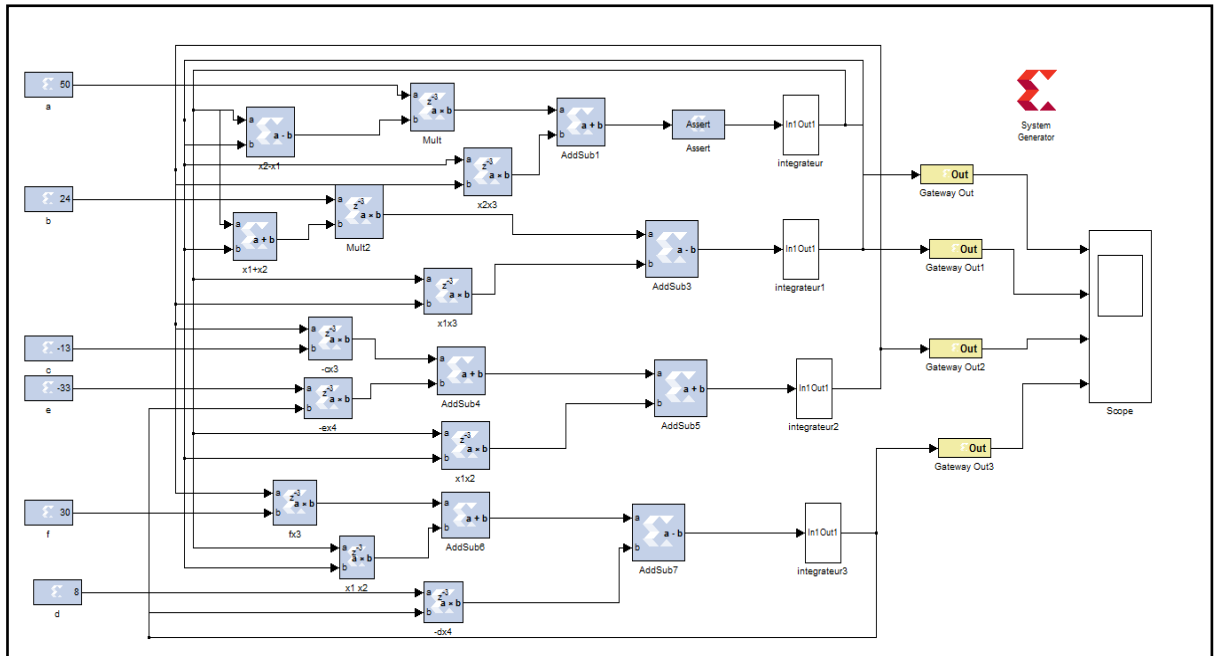
## 4.6 Implémentation du système hyper chaotique du Qi sur FPGA

Nous rappelons Le système hyper chaotique de Qi décrit dans le chapitre 2 :


$$\begin{cases} \frac{dx_1}{dt} = \dot{x}_1 = a(x_2 - x_1) + x_2x_3 \\ \frac{dx_2}{dt} = \dot{x}_2 = b(x_1 + x_2) - x_1x_3 \\ \frac{dx_3}{dt} = \dot{x}_3 = -cx_3 - ex_4 + x_1x_2 \\ \frac{dx_4}{dt} = \dot{x}_4 = -dx_4 + fx_3 + x_1x_2 \end{cases} \quad (4.1)$$

Les variables  $x_1, x_2, x_3, x_4$  représentent l'état du système, et a, b, c, d, e et f sont des paramètres réels.

La figure 4.10 représente le système hyper chaotique de Qi sous System Generator.

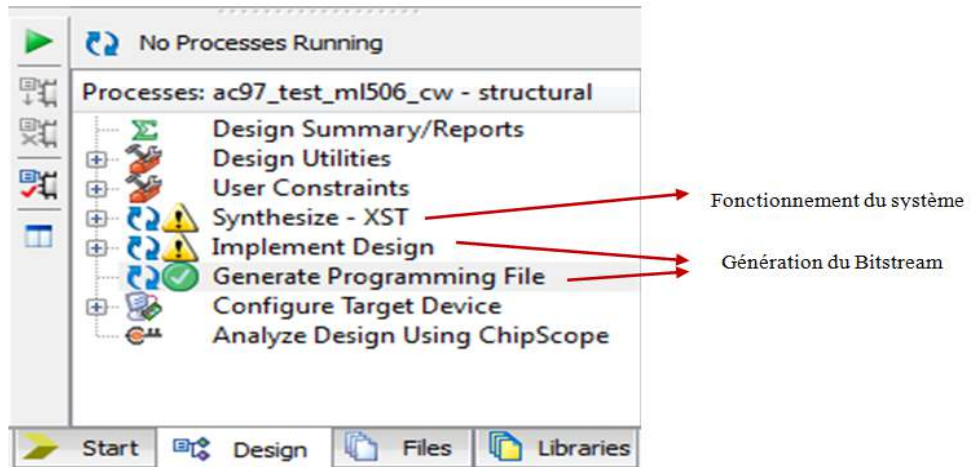


**Figure 4.10.** Le système hyper chaotique du Qi sous System Generator.

Nous avons utilisé le jeton  et les blocs du System Generator sous MATLAB-Simulink pour générer le programme VHDL dans l'environnement ISE-XILINX.

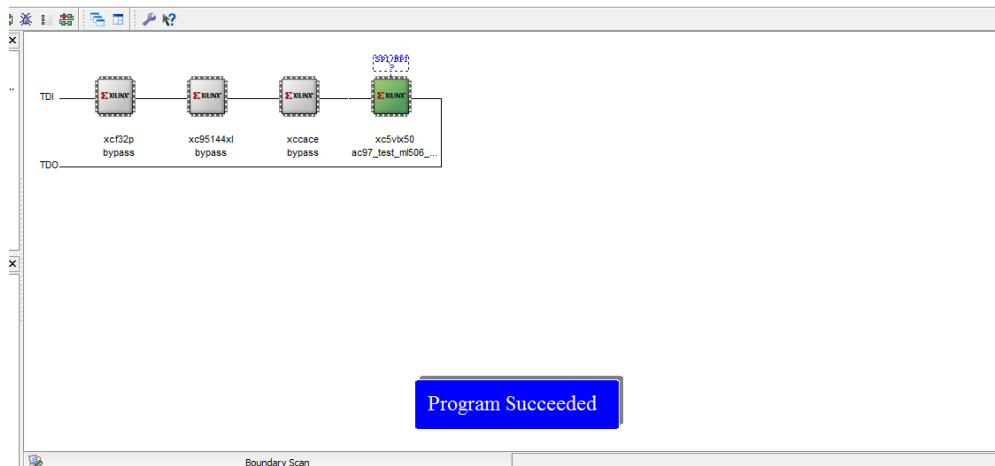
Pour adapter le codec AC'97 avec la carte FPGA, nous devons charger le fichier de l'extension « .ucf » qui contient l'emplacement des entrées et sorties (pin) conformément au datasheet de la carte ML\_501 dans l'environnement ISE-XILINX.

Après avoir chargé le fichier UCF et effectué les différentes étapes de la conception qui sont représentées sur la figure 4.11, nous passons à la dernière étape du processus d'implémentation en téléchargeant le fichier Bitstream sur la carte FPGA Virtex 5. Nous mettons sous tension la carte FPGA en la reliant au PC par câble JTAG; le fichier de programmation bitstream « .bit » sera prêt et placé dans le répertoire projet.



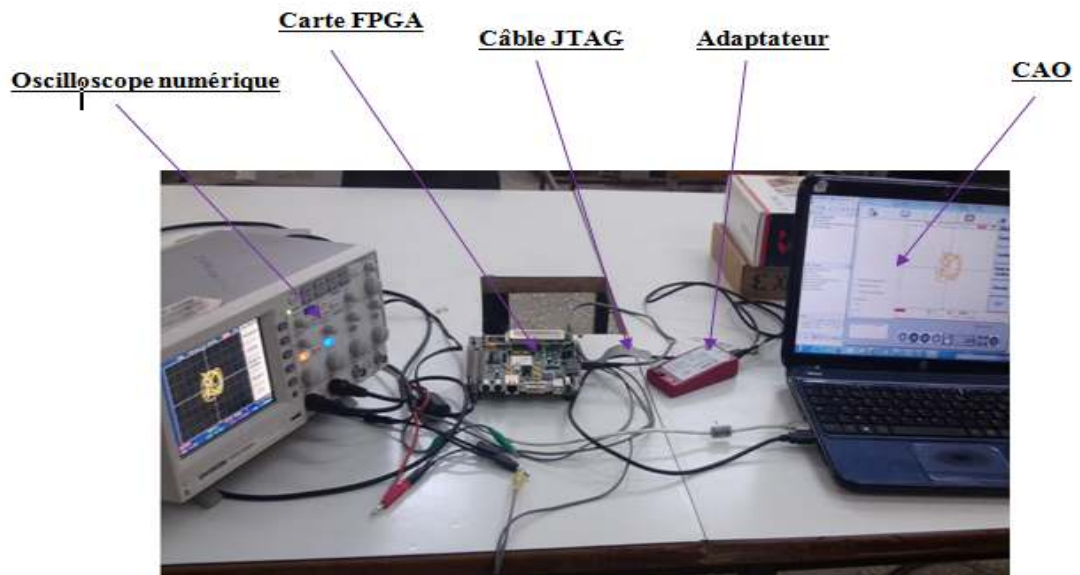
**Figure 4.11.** Les étapes de la conception.

La dernière étape indiquant que la programmation s'est effectuée avec succès est représentée sur la figure 4.12.



**Figure 4.12.** Dernière phase de programmation sur la carte FPGA.

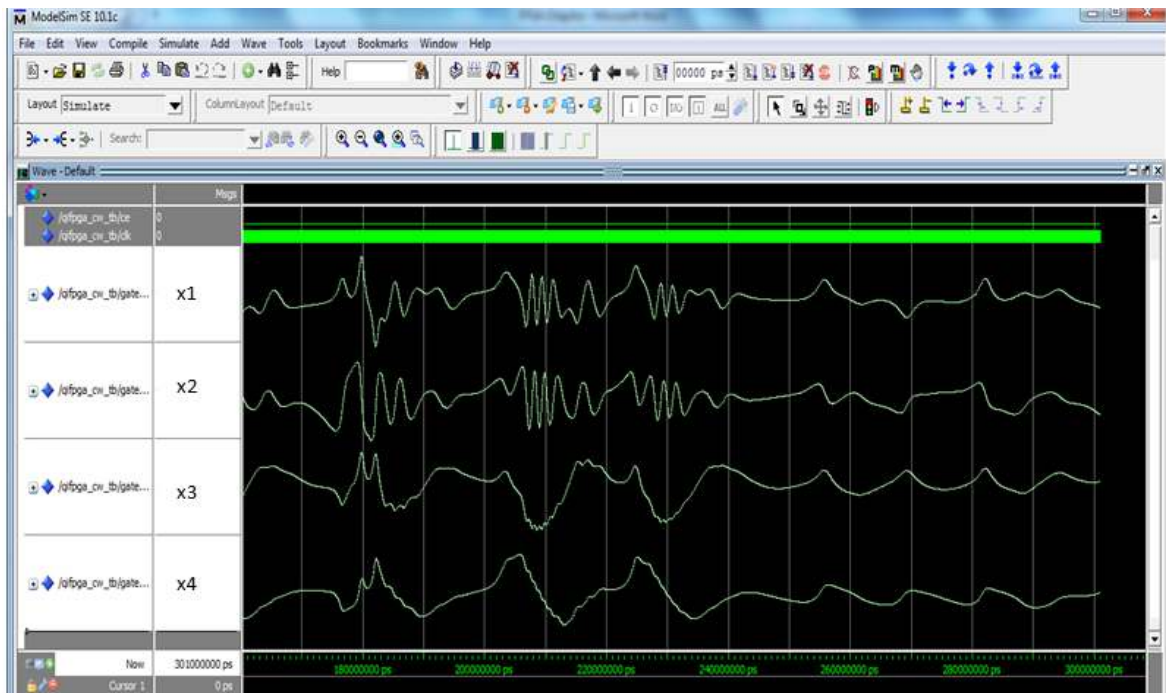
L'environnement de travail au sein du laboratoire LABSET pour l'implémentation du système hyper-chaotique de Qi sur la carte FPGA Virtex-5 est illustré sur la figure 4.13.



**Figure 4.13.** Dispositif expérimental de l'implémentation FPGA.

## 4.7 Visualisation des signaux

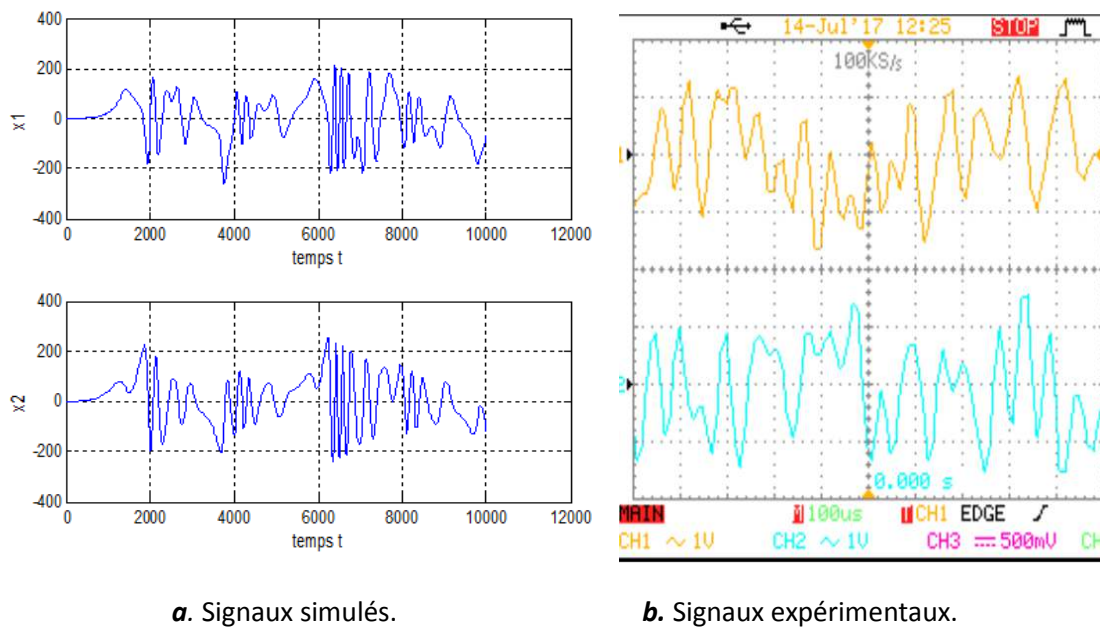
Avant l'implémentations du système de Qi sur la carte FPGA, nous utilisons le logiciel ModelSim pour visualiser les signaux qui sont représentés par la figure 4.14.



**Figure 4.14.** Les signaux de simulation sous Modelsim.



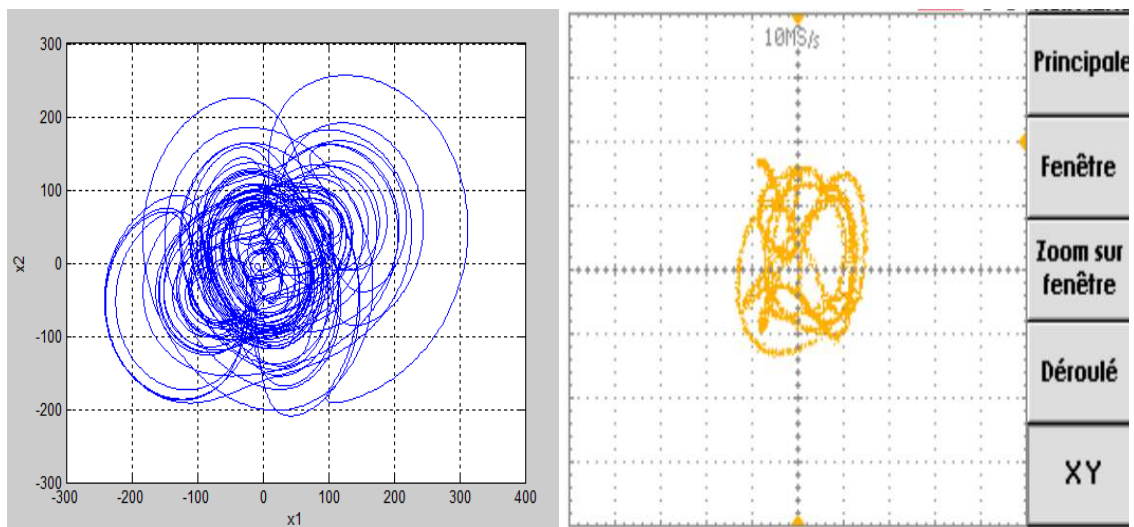
Les différents résultats expérimentaux visualisés sur un oscilloscope numérique et générés par System-Generator du système de Qi sont représentés par les figures 4.15 à 4.18.



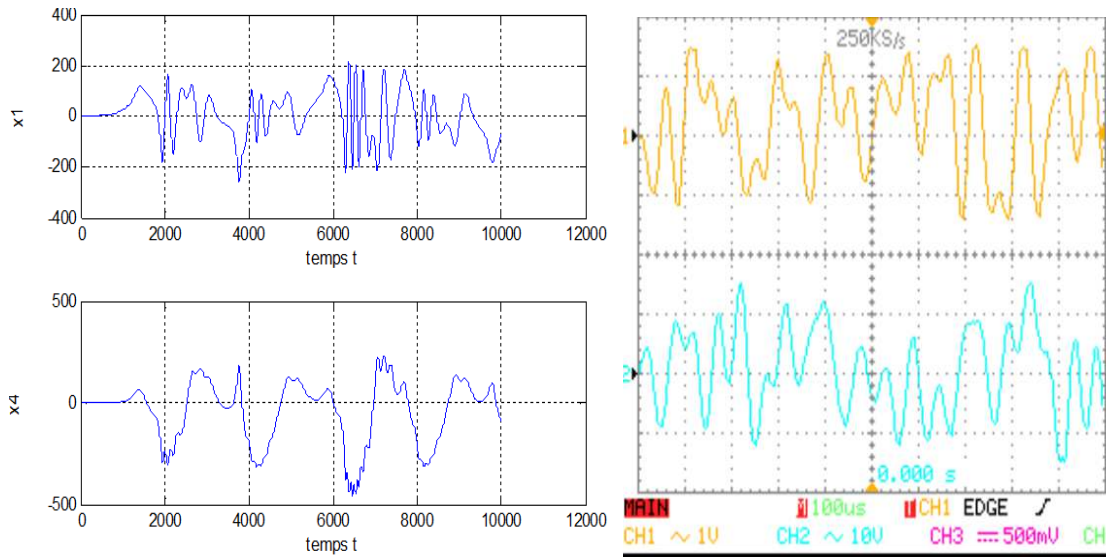
a. Signaux simulés.

b. Signaux expérimentaux.

**Figure 4.15.** Les signaux  $x_1$ ,  $x_2$  en fonction du temps.



**Figure 4.16.** Plan de phase  $x_1$  en fonction de  $x_2$ .



a. Signaux simulés.

b. Signaux expérimentaux.

Figure 4.17. Les signaux  $x_1, x_4$  en fonction du temps.

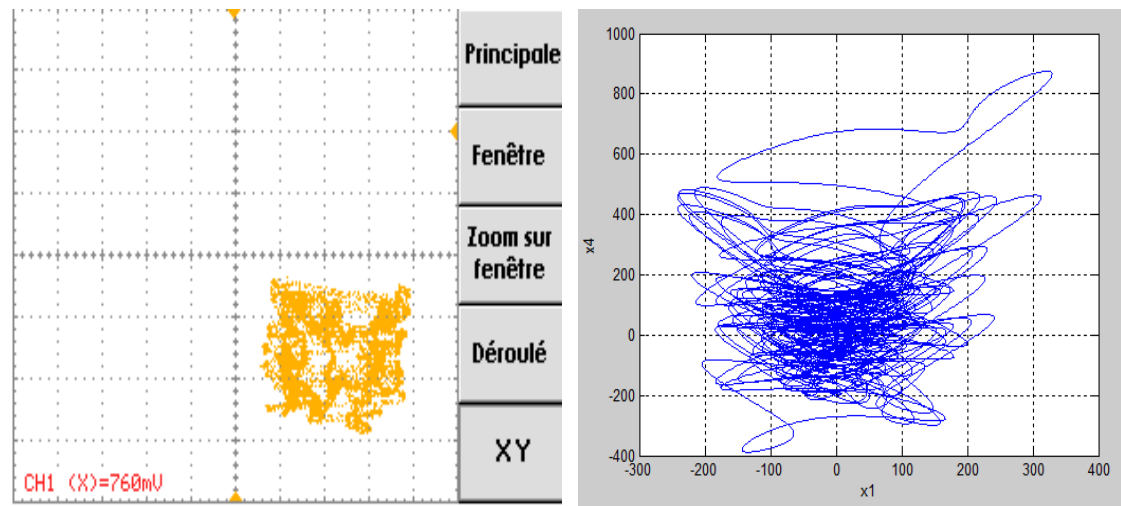


Figure 4.18. Plan de phase  $x_1$  en fonction de  $x_4$ .

D'après ces figures, nous remarquons que les résultats de la simulation et de la partie expérimentale présentent une similitude acceptable qui peut être améliorée en réglant de manière optimale le format des données des différents signaux et des différents paramètres mis en jeu.

L'environnement ISE nous permet de connaître les ressources consommées en pourcentage et en chiffre par la carte FPGA lors de l'implémentation sous forme de tableau contenant les informations utiles et les ressources utilisées liées au design. La figure 4.19 représente ces différents résultats.

ac97_test_ml506_cw Project Status (05/29/2017 - 12:01:11)			
<b>Project File:</b>	ac97_test_ml506_cw.xise	<b>Parser Errors:</b>	No Errors
<b>Module Name:</b>	ac97_test_ml506_cw	<b>Implementation State:</b>	Programming File Generated
<b>Target Device:</b>	xc5vlx50-1ff676	<b>• Errors:</b>	
<b>Product Version:</b>	ISE 14.2	<b>• Warnings:</b>	
<b>Design Goal:</b>	Balanced	<b>• Routing Results:</b>	<a href="#">All Signals Completely Routed</a>
<b>Design Strategy:</b>	<a href="#">Xilinx Default (unlocked)</a>	<b>• Timing Constraints:</b>	<b>X 1 Failing Constraint</b>
<b>Environment:</b>	<a href="#">System Settings</a>	<b>• Final Timing Score:</b>	5502656 ( <a href="#">Timing Report</a> )

Device Utilization Summary				
Slice Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Registers	421	28,800	1%	
Number used as Flip Flops	415			
Number used as Latch-thrus	6			
Number of Slice LUTs	1,075	28,800	3%	
Number used as logic	1,038	28,800	3%	
Number using O6 output only	880			
Number using O5 output only	106			
Number using O5 and O6	52			
Number used as exclusive route-thru	37			
Number of route-thrus	173			

**Figure 4.19.** Ressources consommées lors de l'implémentation.

## 4.8 Conclusion

Ce chapitre est dédié à l'implémentation sur la carte FPGA. Nous avons présenté les circuits FPGAs et leurs importances avec le besoin de composants plus performants, plus économiques et disponibles en grandes quantités. Ensuite, nous avons donné un aperçu sur le codec AC97 et le logiciel XILINX de ISE et le System Generator qui nous permet de créer et générer le langage VHDL à partir de MATLAB-Simulink. Enfin, nous avons implémenté dans la carte FPGA le système hyperchaotique de Qi et une comparaison entre les résultats expérimentaux et les résultats de simulation a été effectuée. Nous avons constaté que les résultats pratiques sont acceptable globalement, si nous prenons en considération la méthode et le système utilisé dans l'implémentation.

# Conclusion générale

---

Dans ce mémoire, nous avons étudié un système de transmission sécurisé de données par chaos, basé sur la synchronisation adaptative et le cryptage paramétrique entre deux systèmes Maître-Esclave.

Nous avons ainsi présenté dans le premier chapitre quelques notions et définitions de base sur les systèmes dynamiques non linéaires, en particulier les systèmes chaotiques, leurs principales caractéristiques, et l'importance des exposants de Lyapunov pour mettre en évidence le comportement chaotique et son évolution vers le chaos.

Le deuxième chapitre a été consacré à l'étude de système du Qi qui présente des caractéristiques très intéressantes (système hyper-chaotique, large bande spectrale ....) pour son utilisation dans le domaine de communications sécurisées.

Le système de transmission sécurisée est ainsi décrit dans le troisième chapitre. Le système est constitué de deux systèmes chaotiques identiques de Qi au niveau de l'émetteur et le récepteur avec des conditions initiales différentes. Dans notre travail, nous avons crypté un message informatif (sinusoïde, image) dans un signal chaotique au niveau de l'émetteur et décrypté au niveau du récepteur. Nous avons pu ainsi récupérer le message identique à l'original grâce à la synchronisation adaptative et le cryptage paramétrique. Nous soulignons l'intérêt de la synchronisation pour pouvoir restituer le message et sécuriser la transmission de l'information. Nous avons ajouté un bruit blanc pour étudier son influence sur la qualité du message récupéré; nous avons remarqué d'après les résultats de simulation que la récupération du message dépend de la synchronisation et de la puissance de bruit.

Nous avons terminé ce travail par l'implémentation du système hyper-chaotique de Qi sur carte FPGA et comparé les résultats expérimentaux obtenus avec les résultats de simulation. Nous avons remarqué que les résultats expérimentaux obtenus présente une similitude avec les résultats de simulation.

Finalement, au cour de ce travail, le système de transmission peut être amélioré par l'étude de la robustesse du système contre le bruit et l'utilisation de technique de cryptage plus complexe et par conséquent plus difficile à décrypter, afin de mieux protéger les données et éviter la dégradation des signaux à récupérer.

# Bibliographie

---

[1] BENHABIB Chouaib : 'Etude d'un système chaotique pour la sécurisation des communications ', Mémoire de master, Université de Tlemcen, 2014.

[2] DANG-VU Huyen et DELCARTE Claudine : 'Bifurcation et chaos', ellipses, 2000.

[3] L'HERNAULT-ZANGANEH Maryam : 'Faisabilité d'un système d'émission-réception analogique pour les communications sécurisées par le chaos', Thèse de doctorat, Université de Paris 6, 2007.

[4] HAMICHE Hamid : 'Insertion à gauche des systèmes dynamique hybrides chaotiques. Application à la transmission sécurisée de données', Thèse doctorat, Université de Mouloud Mammeri Tizi-Ouzou, 2011.

[5] MEGHERBI Ouerdia : 'Etude et réalisation d'un système sécurisé à base de systèmes chaotiques', Thèse de magister, Université Mouloud Mammeri de Tizi-Ouzou, 2013.

[6] TALBI Ibtissem : 'Système dynamique non linéaire et phénomène de chaos. Application à la Cryptographie', Thèse de magister, Université Mentouri de Constantine, 2010.

[7] KIHAL Ahmed Ridha : 'Systèmes chaotiques pour la transmission sécurisée de données', Mémoire de magister, Université Mohamed Khider de Biskra, 2013.

[8] IKHLEF Ameer : 'Contrôle chaotification et hyper chaotification des systèmes dynamiques', mémoire magister, Université de Mentouri Constantine 2007.

[9] REBHI Nada, BEN FARAH Mohamed Amine, KACHOURI Abdennaceur et SAMET Mounir, 'Analyse de sécurité d'une nouvelle méthode de cryptage chaotique', Sciences of Electronic, Technologies of Information and télécommunication, TUNISIA 2007.

[10] ABEB Abdelouahab et AROUS Sid Ahmed Amine : 'Modulation chaotique appliquée en communication', Mémoire de master, Université SAAD DAHLAB de Blida, 2015.

[11] CHIKHI Mohamed Lazhar : 'Application des systèmes dynamiques chaotiques en transmission de données', Thèse de magister, Université SAAD DAHLAB de Blida, 2012.

[12] SNAOUI Djouher : 'Commande Numérique en Force à Base de la carte FPGA d'une Architecteur du Télé-opération à un Seul Degré de liberté', Mémoire de magister, Université MAMMERI Mouloud de Tizi-Ouzou, 2013.

[13] GUETTAT Abdelghani : 'Conception et Implémentation d'un Corrélateur Numérique sur FPGA', Thèse de magister, Université d'Oran Mohamed BOUDIAF, 2012.

[14] HALILALI Abderrezak, TABATOUCHE Imene et HOUACINE Amrane : 'Contribution à l'implantation d'algorithmes de traitement de la parole sur circuit FPGA', Faculté d'Electronique et d'Informatique (FEI) Laboratoire de Communication Parlée et Traitement du Signal, 14-15 Janvier 2015.

[15] AZIB née BENZEMAM Djamilia : 'Systèmes chaotiques et hyper-chaotiques pour la transmission sécurisée de données', Thèse de magister, Université Aboubekr belkaid de Tlemcen, 2010.

[16] BOUKABOU Abdelkrim : 'Méthode de contrôle des systèmes chaotiques d'ordre élevé et leur application pour la synchronisation : Contribution à l'élaboration de nouvelles approches', Thèse de doctorat, Université de Constantine, Juin 2006.