République **A**lgérienne **D**émocratique et **P**opulaire
Ministère de L'**E**nseignement **S**upérieur et de la A **R**echerche **S**cientifique

UNIVERSITE **M**OULOUD **M**AMMERI DE **T**IZI-**O**UZOU

FACULTE DE **G**ENIE **E**LECTRIQUE ET D'INFORMATIQUE
DEPARTEMENT D'AUTOMATIQUE

Mémoire de Fin d'Etude de MASTER ACADEMIQUE

Spécialité : **automatique option commande des systèmes**

Présenté par

Samir ALLOUACHE
Nabil HAMMA

Mémoire dirigé par **Mr Redouane KARA**

Thème

Conception et réalisation d'un système de transmission sécurisé de données à base de systèmes chaotiques sur cartes Arduino

Mémoire soutenu publiquement le 09/07/2015 devant le jury composé de :

Mr Ahmed MAIDI
MCA, UMMTO, Président

Mr Redouane KARA
MCA, UMMTO, Encadreur

Mr Said DJENNOUNE
Professeur, UMMTO, Examinateur

Mr Hamid HAMICHE
MCA, UMMTO, Examinateur

Ce travail a été réalisé au laboratoire L2CSP

Remerciements

Remerciements

En premier lieu, nous tenons à remercier notre créateur tout puissant pour nous avoir donné la force et le courage pour accomplir ce modeste travail.

Nos vœux vont d'abord à nos promoteurs « «Mr Kara et Mr Hamiche » qui ont suivi l'évolution de notre travail, Ainsi que « Mlle Megherbi » et «M R.Sadaoui » pour leurs aide précieuse et pour leurs permanente disponibilité.

Nous tenons à remercier chaleureusement les membres du jury de nous faire l'honneur d'accepter d'évaluer ce projet.

Enfin, que tous ceux, qui de près ou de loin, ont participé à l'élaboration de ce travail trouvent ici l'expression de nos meilleurs remerciements.

Dédicaces

Dédicaces

Je dédie le fruit de mes années d'études à mes très chers parents, car sans leur soutient ce travail n'aurait jamais vu le jour

.

A mes chers frères Lyes et Ouahioune.
à ma chère petite sœur.

à mon binôme Samir.

à la mémoire de Djedi ,Yaya et à ma chère tante Samia , Allah yermhoume.

à toute ma famille, grand parents, tente, oncle, cousins et cousines.

à tous mes amis.

à toutes ces personnes et a celles que j'ai peut être oubliées, à qui j'adresse mes sentiments les plus chaleureux.

Nabil

Dédicaces

Je dédie le fruit de mes années d'études à mes très chers parents, car sans leur soutient ce travail n'aurait jamais vu le jour.

A mes chers frères et sœurs

à mes chers neveux, Amina, Islam, Hanane, Ahcene et la petite Chanez .

à mon binôme Nabil.

à toute ma famille, tente, oncle, cousins et cousines.

à tous mes amis.

à toutes ces personnes et a celles que j'ai peut être oubliées, à qui j'adresse mes sentiments les plus chaleureux.

Samir

Sommaire

Sommaire

Sommaire

Sommaire

Liste des figures

Liste des figures

Liste des figures

Liste des tableaux

Liste des tableaux

Liste des acronymes et abréviations

Liste des acronymes et abréviations

PWM Pulse-width modulation

USB Universal Serial Bus

LED Light Emitting Diode

LCD Liquid Crystal Display

MP3 Mpeg Audio Layer 3

GPS Global Positioning System

IDE Integrated Development Environment

CAN Convertisseur Analogique Numérique

UART Universal Asynchronous Receiver Transmitter

ASCII American Standard Code for Information Interchange

LSB Least Significant Bit

MSB Most Significant Bit

ICSP In Cicuit Serial Programing

Introduction générale

Introduction générale

La théorie du chaos s'intéresse à des phénomènes qui en apparence paraissent désordonnés et aléatoires mais qui sont gouvernés par des lois déterministes. Le premier à avoir remarqué ce phénomène est Henri Poincaré lors d'études consacrés à la stabilité du système solaire. Par la suite, plusieurs scientifiques se sont intéressés de près à la théorie du chaos, ainsi qu'aux méthodes de son contrôle. Un phénomène chaotique est défini comme étant un phénomène ayant généralement un comportement imprévisible particulier d'un système dynamique déterministe non-linéaire. Le chaos possède un large domaine d'application, on le retrouve dans la mécanique, l'électronique et même dans la biologie.

Le chaos est caractérisé par un certain nombre de caractéristiques dont la sensibilité aux conditions initiales et l'imprévisibilité, ce qui rend les systèmes chaotiques très intéressants dans le cryptage des données.

Introduite en 1990 par Pecora et Carroll, la synchronisation est une technique qui, étant donné deux systèmes chaotiques identiques (émetteur et récepteur), consiste à forcer la trajectoire du récepteur à suivre celle de l'émetteur. Plusieurs méthodes de synchronisation ont été proposées dans la littérature scientifique, elles se basent sur le principe du maitre-esclave et permettent de réduire l'erreur entre les trajectoires de l'émetteur et du récepteur.

La cryptographie est une science qui s'intéresse à la protection des messages à transmettre, et ce en le rendant incompréhensible. La cryptographie a évolué grâce au conflit qui a toujours opposé deux camps, l'un cherche à dissimuler une information et l'autre essaie par tous les moyens de trouver ce que on lui cache, à chaque fois que le premier trouve le moyen de chiffre ses messages, le second essaie par tous les moyens de trouver l'astuce qui va lui permettre de décrypter l'information. Autrefois pour dissimuler une information, on mélangeait, permutait ou décalait des lettres, d'autres remplaçait les mots par des nombres, et ce dans le but de rendre la lecture de message impossible. Mais la cryptographie n'a cessé d'évoluer, actuellement on chiffre le message clair d'une façon mathématique et algorithmique, plus l'inversion de la transformation est difficile plus la sécurité est élevé, et vice versa.

Dans ce travail, nous nous proposons de concevoir et de réaliser un système de transmission sécurisé de données basé sur le cryptage par le chaos. Le système est réalisé sur une carte électronique de type Arduino. L'intérêt de l'utilisation de chaos réside d'une part, dans le fait de la possibilité d'exploiter les propriétés des systèmes chaotiques dans le cryptage (sensibilité au conditions initiales, …)., et d'autre part sur la possibilité de la synchronisation par observateur.

Pour ce faire, nous utiliserons un système chaotique particulier appelé « système de lozi ». Deux systèmes chaotiques identiques seront utilisés, l'un nous servira d'émetteur qui est un générateur de signaux chaotique, et l'autre nous servira de récepteur. Pour récupérer le signal chaotique émis par l'émetteur, on synchronisera les deux systèmes a l'aide d'un observateur impulsif.

Le cryptage se fera en noyant le message utile dans le signal chaotique émis par l'émetteur, la récupération du message se fera par une simple soustraction du signal chaotique.

Ce travail comporte cinq chapitres :

-Le chapitre I est consacré aux généralités et aux notions de base sur les systèmes chaotiques.
-Le chapitre II traitera des généralités sur la synchronisation des systèmes chaotiques, ainsi que le principe de transmission avec les méthodes de cryptage et de décryptage.
-Le chapitre III est dédié à l'étude théorique et la synchronisation du « système de lozi » en utilisant la commande par observateur impulsive.
-Au chapitre IV la carte électronique « Arduino » et ses principaux composants sont présentés.
-Le chapitre V traitera de la réalisation pratique du schéma de transmission adopté et des différents tests effectués.

Enfin, on termine par une conclusion générale et des perspectives.

Chapitre I

Généralité sur les systèmes chaotiques

I.1 Introduction

A l'ère d'Isaac Newton (1642-1727) [1], le déterminisme dominait la science et les scientifiques croyaient pouvoir prédire et prévoir le futur d'une manière exacte à condition de connaitre les conditions initiales et les paramètres.

Cette théorie a été confirmé par Newton et Laplace qui affirmait la notion du déterminisme en disant qu'il pouvait prédire le futur de l'univers en connaissant juste son état présent, mais cette théorie a été contredite par Poincaré (1913) en avançant que il ne pouvait connaitre l'évolution et prédire le problème des trois corps de la mécanique céleste (exemple : lune, terre et soleil) et ceci malgré leur nature déterministe.

L'étude de la stabilité en comparant les trajectoires suivies par un des corps à partir de deux positions initiale très proches, ou on a conclu que les trajectoires étaient presque identiques à court terme [2], mais à long terme il y'avait une nette différence, donc on ne peut jamais prédire complètement l'évolution d'un système chaotique, cette signification a été avancée en 1908 par pierre Duhem.

La première visualisation de phénomène du chaos déterministe a été observée par coïncidence par Edward Lorenz en 1961, à la suite d'une série de calculs qui avaient pour but de prévoir des phénomènes météorologiques. Ce dernier se servait de son ordinateur (royal McBean lgp-300) pour calculer ses prévisions, en obtenant ses résultats finaux, il voulait les refaire une deuxième fois pour s'assurer, pour gagner du temps il a pris en compte que trois chiffres après la virgule au lieux de six en croyant que il aurait une petite variation dans les résultats, mais il a été stupéfait par ces résultats qui était totalement différents des premiers.

A partir de là, on a découvert le comportement chaotique d'un système non linéaire, une métaphore a contribué à l'essor de la théorie de Lorenz : « le simple battement d'aile du papillon au Brésil pourrait déclencher une tornade au Texas » [3].

I.2 Quelques définitions utiles

I.2.1 Système dynamique

C'est un modèle mathématique qui décrit l'évolution des phénomènes soit (mécanique, physique,…, etc) par rapport aux temps. Il est caractérisé par un plan de phase et un système d'état, il peut être décrit par un ensemble d'équations qui peuvent prendre des formes diverses (équations différentielles ordinaires, équations aux dérivées partielles, …,etc) [4].

I.2.2 Plan de phase

Le plan de phase d'un système dynamique est une représentation graphique de plusieurs trajectoires représentative dans l'espace de phase. Etant donné un système dynamique $\dot{x} = f(x,t)$, sans résoudre les équations, on peut toujours à un instant (t)

donnée, représenter graphiquement (à l'aide de flèches) le champ des x la lecture de cette représentation graphique sera très utile pour se faire une idée sur le comportement du système [5].

I.2.3 Système linéaire

On appelle système linéaire, un système dont le modèle mathématique est linéaire (équations différentielles, aux différences,..), qui obéit au principe de la superposition, et de proportionnalité.

Il peut être représenté par un système de la forme suivante :

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{I.1}$$

Avec :

$x(t)$: Vecteur colonne de variable d'état de dimension n (n : dimension de l'espace d'état)

$y(t)$: Vecteur colonne des sorties du système de dimension p.

A : matrice d'état de taille $(n * n)$.

B : matrice de commande de taille$(n * m)$.

C : matrice d'observation de taille$(p * n)$.

D : matrice de transmission de taille$(p * m)$.

I.2.4 Système non linéaire

Un système est dit non linéaire s'il ne respecte pas le principe de superposition et si la relation entre les grandeurs d'entrée et de sortie est une équation différentielle avec des coefficients non constants généralement. Il peut être représenté comme suite :

$$\begin{aligned} &\text{Équation d'état} \quad \dot{x}(t) = F(x(t), u(t)) \\ &\text{Équation de sortie} \quad y(t) = H(x(t), u(t)) \end{aligned} \tag{I.2}$$

Il est à noter que la plupart des systèmes physiques sont des systèmes non linéaires.

I.2.5 Système déterministe

Un système est dit déterministe, si pour exactement les mêmes paramètres, les mêmes conditions initiales et les mêmes conditions aux limites il donne les mêmes résultats uniques.

Soit le système suivant :

$$\dot{x} = f(x) \qquad \text{(I.2)}$$

Et D l'ensemble des conditions initiales du système (I.3), si pour tout $x_0 \in D$ il existe une solution unique $\varphi(t, x_0)$, alors on dira que le système est déterministe.

I.2.6 Exposant de lyapunov

Dans l'analyse d'un système dynamique l'exposant de Lyapunov est un coefficient qui mesure le taux de convergence ou de divergence de deux trajectoires voisines au départ dans l'espace des phases. Souvent représenté par le symbole lambda, permet d'approximer la durée du comportement prévisible d'un système dynamique et le moment où il basculera dans un comportement chaotique. Le nombre des exposants de lyapunov est égal à la dimension du système.

I.2.7 Attracteur

Tous les points de l'espace de phase sont caractérisés par des trajectoires. Ces dernières sont attirées vers un objet géométrique qui se nomme attracteur, qui est un ensemble où un espace vers lequel un système évolue de façon irréversible. Constituants de base de la théorie de chaos cinq types d'attracteurs sont définit (Ponctuel, Périodique, Ponctuel périodique, Spatial, Étrange.)

I.2.8 Cycle limite

Tout système non linéaire qui a un siège d'oscillations est dit cycle limite, caractérisé par leur amplitude et leur fréquence indépendante de la condition initiale x_0, et sans excitation extérieure.

I.2.9 Tore

Cas particulier de cycle limite, le système présente aux moins deux période simultanés dont le rapport est irrationnel (aléatoire), la trajectoire de phase ne s'annule pas sur elle-même.

I.2.10 Système autonome

On appelle système autonome tout système dynamique indépendant explicitement du temps, il est représenté comme suit

$$X = F(x, y)$$
$$Y = G(x, y) \qquad \text{(I.4)}$$

Avec $x\ et\ y$ vecteurs.

Pour un système autonome chaque instant peut être considéré comme instant initiale, et chaque état x(t) du système peut être considéré comme un état initial.

I.2.11 Causalité

Un système est causal, si une action n'a pas d'effet dans le passé sur le système.

On dit d'un signal causal si il n'est pas définit pour $t < 0$ on prend l'exemple de l'échelon unitaire :

$$\mathrm{X(t)} \begin{cases} \neq 0 \text{ si } t > 0 \\ 0\ si\ t < 0 \end{cases} \qquad \text{(I.5)}$$

$$\begin{cases} x(t) = 1\ si\ x\ [0\ + infini\ [\\ \\ 0\ ailleurs \end{cases}$$

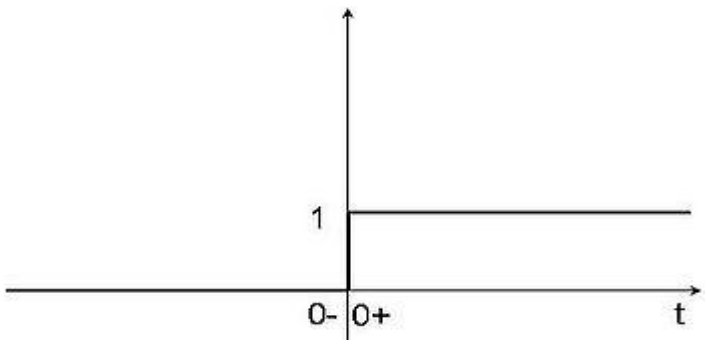


Figure (I.1) : Echelon unitaire causale.

I.3 Définition du chaos

Le chaos est un phénomène imprévisible désordonné impossible à prédire à long terme, qui dépend de plusieurs paramètres. Il est caractérisé par une extrême sensibilité aux conditions initiales, les systèmes chaotiques sont modélisés par des équations non linéaires, un système linéaire ne peut être chaotique.

On prend l'exemple du système de Rössler, modélisé par les équations suivantes [1]

$$\begin{cases} \dot{z}(t) = b + z(x - c) \\ \dot{y}(t) = x + ay \\ \dot{z}(t) = b + z(x - c) \end{cases} \quad \text{(I.6)}$$

Avec a, b et c paramètres du système.

I.4 Caractéristiques des systèmes chaotiques

On reconnaît un système chaotique pas l'analyse de ces caractéristiques.

I.5.1 Sensibilité aux conditions initiales

Pour des conditions initiales très proches les courbes se superposent, et petit à petit, elles se dissocient pour donner des valeurs complètement différentes, malgré son caractère déterministe il est impossible de prédire l'évolution de la trajectoire.

On va illustrer cette caractéristique par l'exemple de Lorenz (voir figure ci-dessous) :

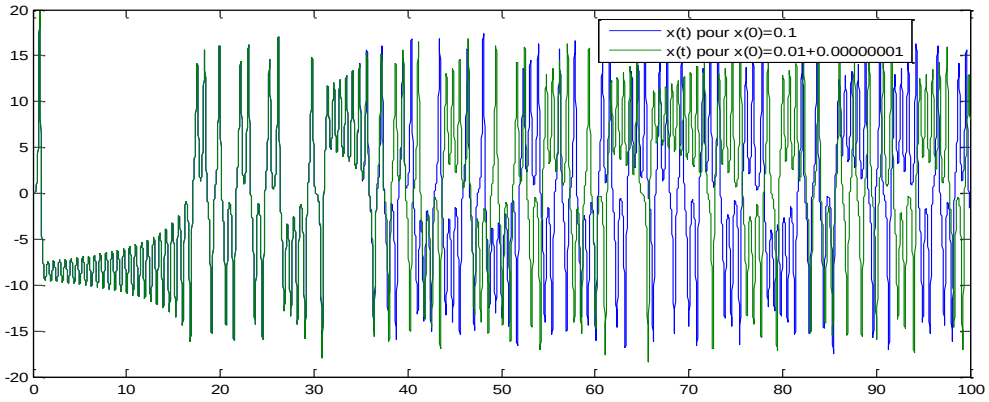


Figure (1.2) : Sensibilité aux conditions initiales du modèle de Lorenz

Les paramètres étant fixés aux valeurs suivantes :

$$x_0(0) = 0.1\ ;\ x_1(0) = 0.1 + 0.00000001 \quad \text{(I.7)}$$

On remarque que les deux trajectoires sont identiques au début, mais après un certain temps, elles divergent.

I.5.2 Aspect aléatoire

L'une des caractéristiques des systèmes chaotiques est l'aspect aléatoire de son évolution temporel, il est non périodique.

La figure (I.3) illustre le caractère de l'évolution de l'un des composants du système de Lorenz par rapport au temps (voir figure ci-dessous) :

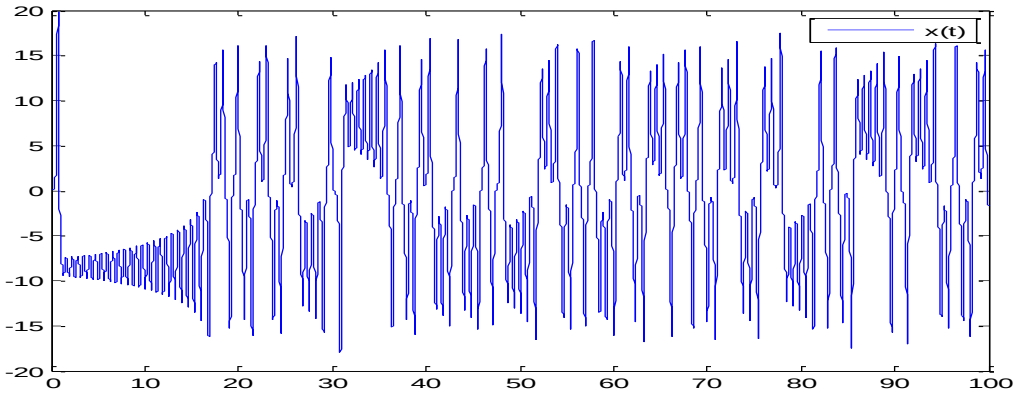


Figure (I.3) : Aspect aléatoire de du système de Lorenz.

I.5.3 L'attracteur étrange

La représentation de l'évolution d'un système chaotique se fait dans le plan de phase, à chaque instant son état est représenté par un point l'ensemble de ces points dessine une figure que on appelle attracteur. En général, ces attracteurs sont simples comme pour un pendule simple, dans le cas des systèmes chaotiques, l'allure est très complexe dite attracteur étrange.

Ce comportement étrange (voir figure ci-dessous) est due à la dimension fractale des systèmes chaotiques et à la présence d'aux moins d'un exposant de Lyapunov de signe différent ce qui fait apparaitre des étirements et des attractions des trajectoires mais sans jamais se coupé.

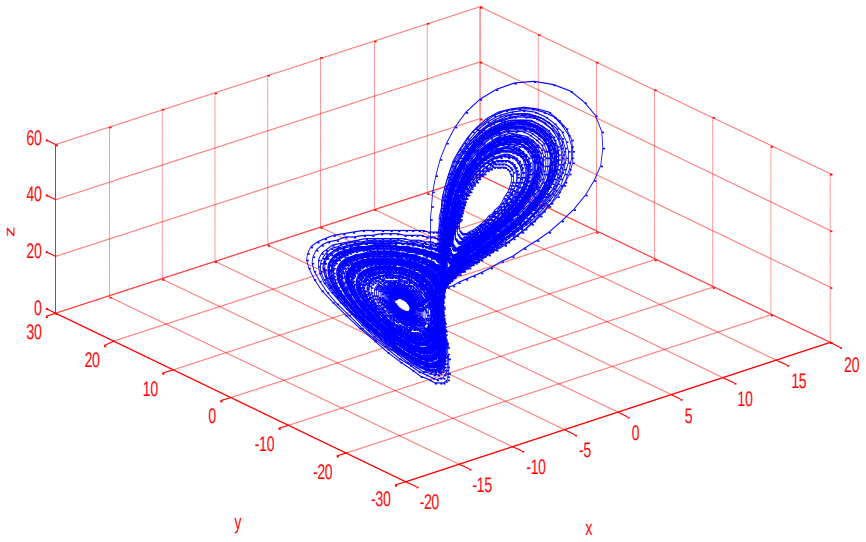


Figure (I.4) : Représentation des états x, y et z du modèle de Lorenz dans l'espace 3D.

L'attracteur de Lorenz a la forme d'un papillon, la trajectoire effectue plusieurs tours sur une aile puis elle bascule de façon imprévisible sur l'autre.

I.5.4 Le spectre de puissance

Pour un signal chaotique, le spectre de puissance a une large bande qui est riche en fréquences, ce qui donne un intérêt à utiliser les systèmes chaotique dans transmission de donné (voir figure (1.5)).

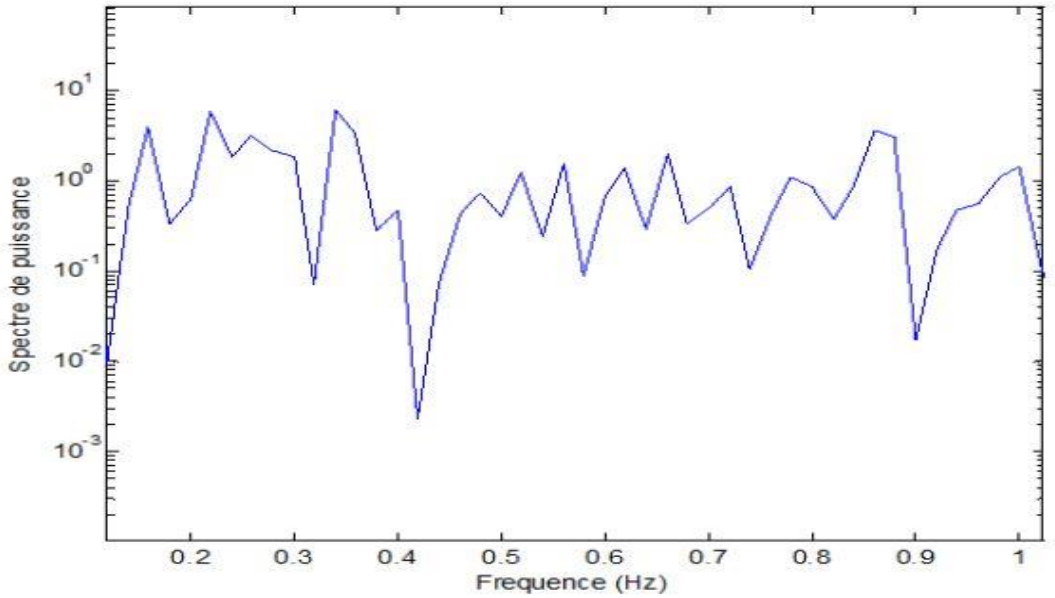


Figure (1.5) : Spectre de puissance du modèle de Lorenz

I.5.5 Les exposants de Lyapunov [6]

Un Exposant de Lyapunov est appelé aussi la vitesse de divergence ou de convergence. Pour qu'un système ait une dynamique chaotique trois conditions sont nécessaires :

- Au moins un exposant de Lyapunov positif qui fait diverger la trajectoire.
- Au moins un exposant de Lyapunov négatif qui fait replier la trajectoire.
- La somme des exposants doit être négative pour un système dissipatif (système qui évolue dans un environnement avec lequel il échange de l'énergie).

Un système discret chaotique possède au moins un exposant de lyapunov négatif.

Etat	Attracteur	Spectre Lyapunov	Exposants de Lyapunov
Point d'équilibre	Point	0	$\lambda_n \leq \ldots \leq \lambda_1 \leq 0$
Périodique	Cercle limite	1	$\lambda_1 = 0$ $\lambda_n \leq \ldots \leq \lambda_2 \leq 0$
Période d'ordre 2	Tore	2	$\lambda_1 = \lambda_2 = 0$ $\lambda_n \leq \ldots \leq \lambda_3 \leq 0$
Chaotique	-	Non entier	$\lambda_1 > 0$ $\sum_{i=2}^{n} \lambda_i < 0$
Hyper chaotique	-	Non entier	$\lambda_1 > 0$ $\lambda_2 > 0$ $\sum_{i=3}^{n} \lambda_i < 0$

Tableau (I.1) : Classification des régimes permanents en fonction du spectre Lyapunov.

I.5.6 Le diagramme de bifurcation

Tout système régit pas des équations différentielles a un comportement asymptotique en fonction de ses paramètres, un changement quantitatif de ces dernier produit un changement qualitatif (ex : un point d'équilibre dans le plan de phase devient un cycle limite), on appelle ce changement « bifurcation »

Les valeurs des paramètres associées est appelé valeur de bifurcation. L'analyse des bifurcations a pour but de localiser ces derniers.

I.5 Route vers le chaos

Il existe plusieurs routes d'évolution vers le chaos et celle-ci ne se produisent que pour un changement de valeurs d'un paramètre par ‘’bifurcation’’.

I.6.1 Doublement de période

L'augmentation d'un paramètre dans un système périodique fait doubler la fréquence du régime périodique double, puis elle est multipliée par 4, par 8, par 16 etc., jusqu'à l'apparition du chaos.

I.6.2 Quasi périodicité

Ce troisième scénario fait intervenir pour un système périodique l'apparition d'une autre fréquence dont le rapport avec le premier est irrationnel, un nouveau changement de paramètres fait apparaitre une troisième fréquence, et ainsi de suite jusqu'au chaos.

I.6.3 Intermittence

Est un phénomène qui se manifeste dans un système dynamique et périodique par des phases de périodicité stable et des phases chaotique après un certain temps, le comportement chaotique prend le dessus et il devient chaotique.

On prend l'exemple le la suite logistique [7]

$$x_{n+1} = a x_n (1 - x_n) \qquad \text{(I.8)}$$

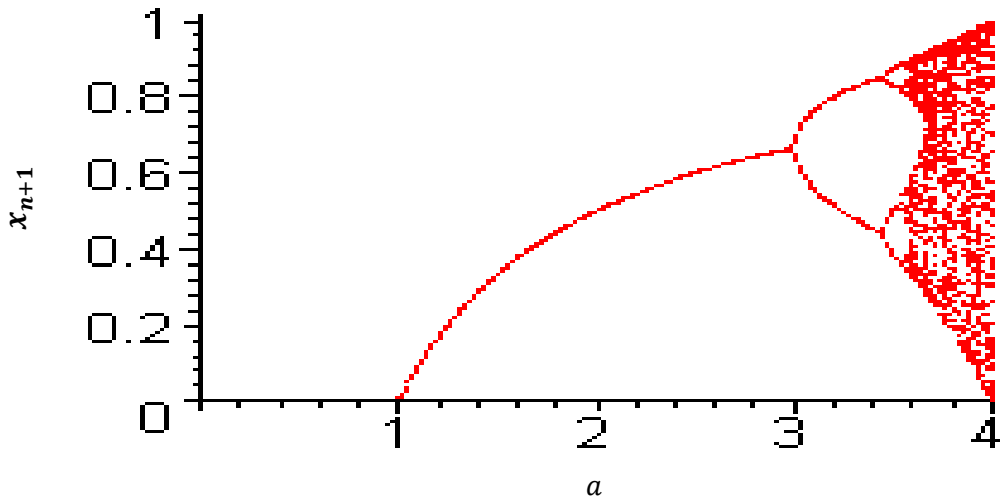


Figure (1.6) : Diagramme de bifurcation de la suite logistique.

La valeur du paramètre a est représentée sur l'axe des ordonnées, sur l'axe des abscisses sont représentées les valeurs prise par la suite logistique

- Pour $\boldsymbol{a} \in [\mathbf{0}\ \mathbf{3}[$ la suite à un comportement simple étant donné la présence d'un seul point fixe.
- Pour $\boldsymbol{a} \in [\mathbf{3}\ \mathbf{4}[$ augmentation des points périodiques, si on effectue une normalisation on va constater l'apparition de plusieurs périodes.
- Pour $\boldsymbol{a} = \mathbf{4}$ le comportement de la suite devient chaotique.

I.6 Génération du chaos

Il existe plusieurs systèmes qui ont une dynamique chaotique, on va prendre comme exemple le système de Lorenz pour le continue et celui de Hénon pour le discret.

I.7.1 Systèmes chaotiques continus

Le système le Lorenz prend la forme différentielle suivante [8]

$$\begin{aligned} \dot{x}(t) &= a(y-x) \\ \dot{y}(t) &= -xz + cx - y \\ \dot{z}(t) &= xy - bz \end{aligned} \tag{I.9}$$

Les valeurs de a et b sont respictivement egale a 10 et 8/3, c est un paramètre de contrôle, les conditions initiale sont fixées à (0.01 0.01 0.01).
Pour avoir une dynamique chaotique, on fait varier le paramètre, le comportement du système de Lorenz pour des valeurs différentes sont représentés dans les figure suivante :

- Pour $\boldsymbol{c = 10}$:

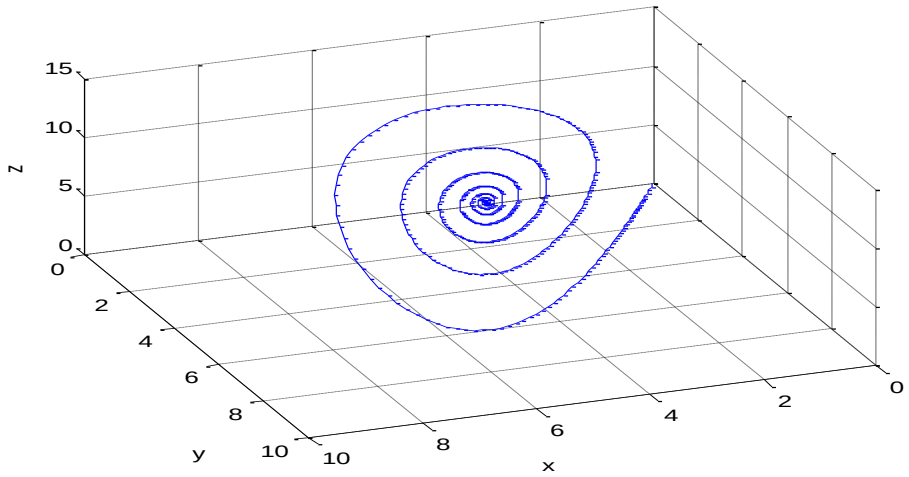
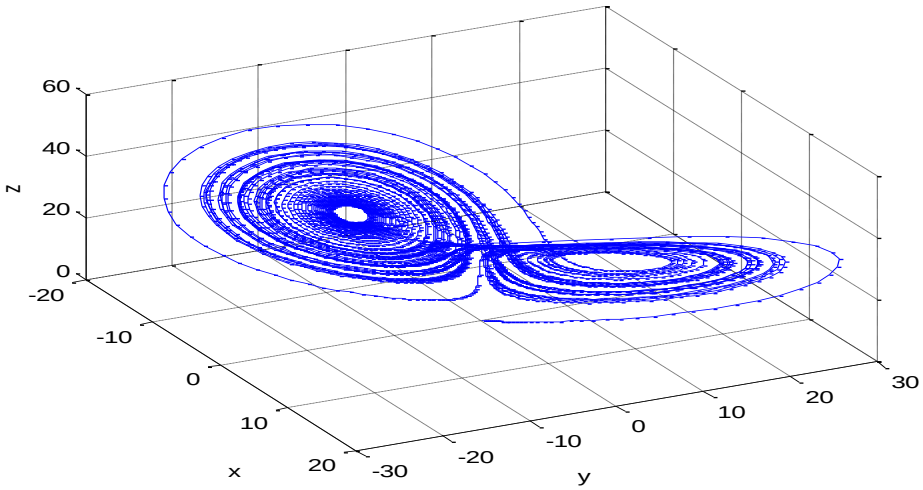


Figure (I.7) : Point fixe.

La trajectoire converge vers un point d'équilibre, pas de dynamique chaotique.

- Pour $\boldsymbol{c = 28}$:



-

Figure (I.8) : Attracteur étrange (effet papillon).

Un attracteur étrange apparait en forme d'un papillon caractéristique au système de Lorenz, la dynamique du système et chaotique.

- Pour $\boldsymbol{c = 160}$:

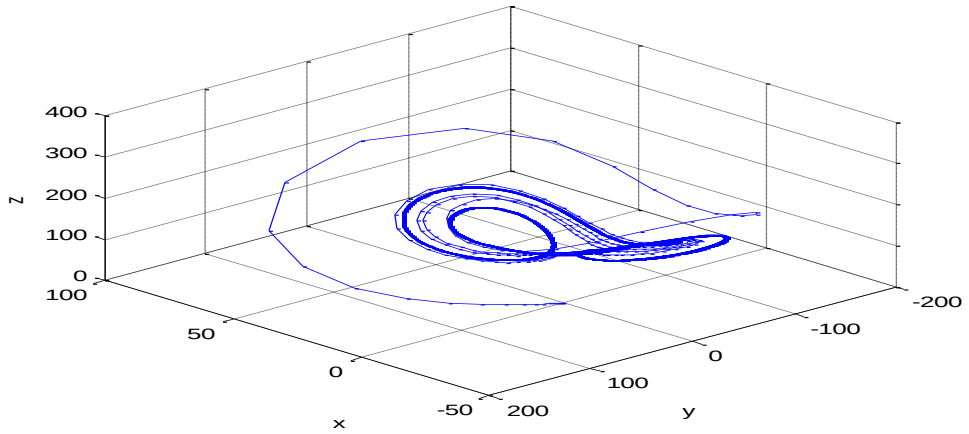


Figure (I.9) : Tore.

La trajectoire converge vers un Tore.

La Figure (I.10) illustre la propriété de sensibilité aux conditions initiales avec une différence de l'ordre de $\sigma = 10^{-6}$.

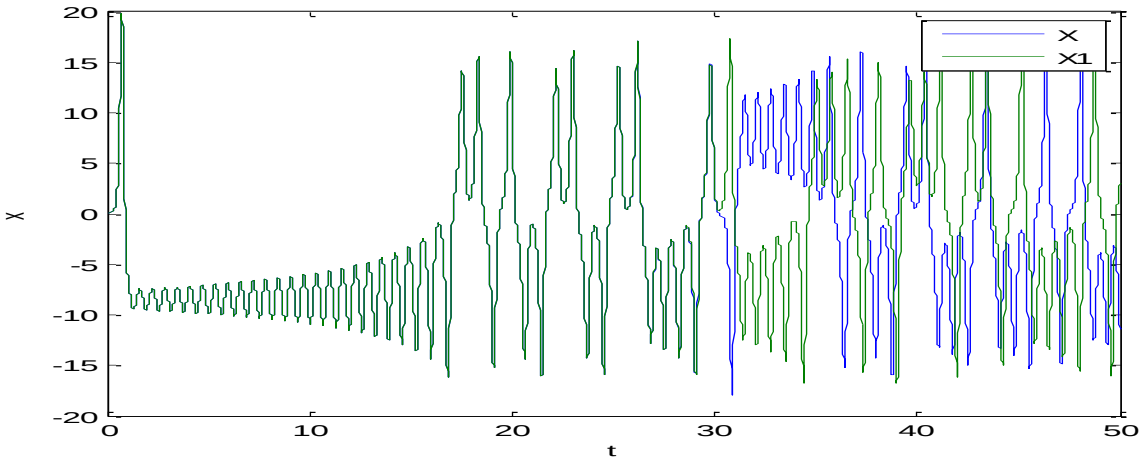


Figure (I.10) : Sensibilité aux conditions initiales.

I.7.2 Systèmes chaotiques discrets

On prend comme exemple le système de Hénon qui est à temps discret, présenté sous la forme d'équations aux différences suivante :

$$x_{k+1} = a - x_k^2 + by_k$$
$$y_{k+1} = x_k \qquad \text{(I.10)}$$

Avec (x, y) vecteur d'état et a, b paramètres du système.

Le système de Hénon modifié a une dynamique chaotique et sa trajectoire dans le plan de phase est un attracteur étrange pour les paramètres a=1.4, b=0.3 est de condition initiale (0.1, 0.1) [9].
Sur les figures suivantes sont représentés les trajectoires respectivement de x et y, ainsi la représentation dans le plan de phase.

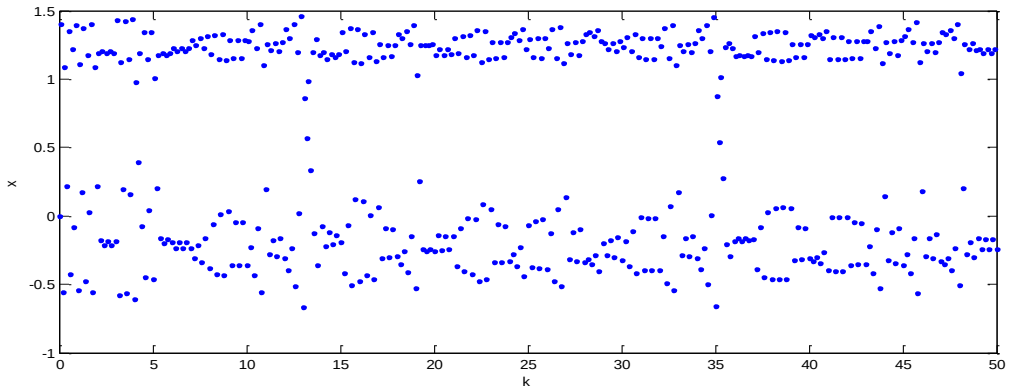


Figure (I.11) : Evolution aléatoire de x(k) système de Hénon

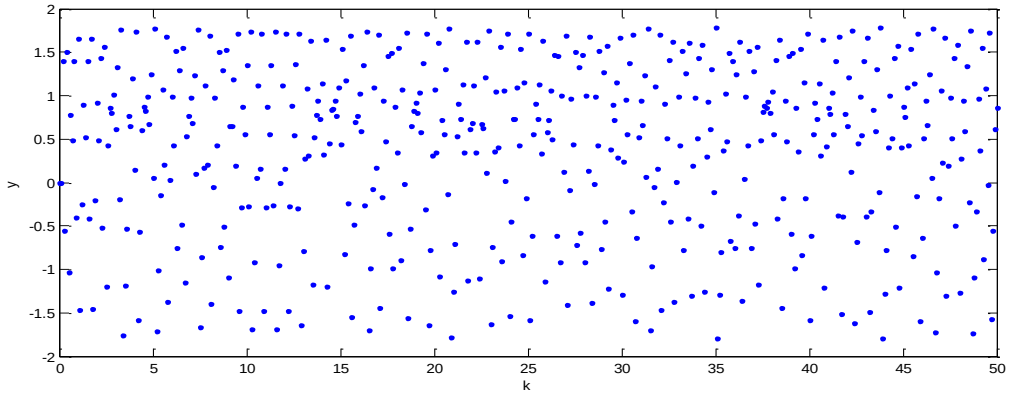


Figure (I.12) : évolution aléatoire de y(k) système de Hénon

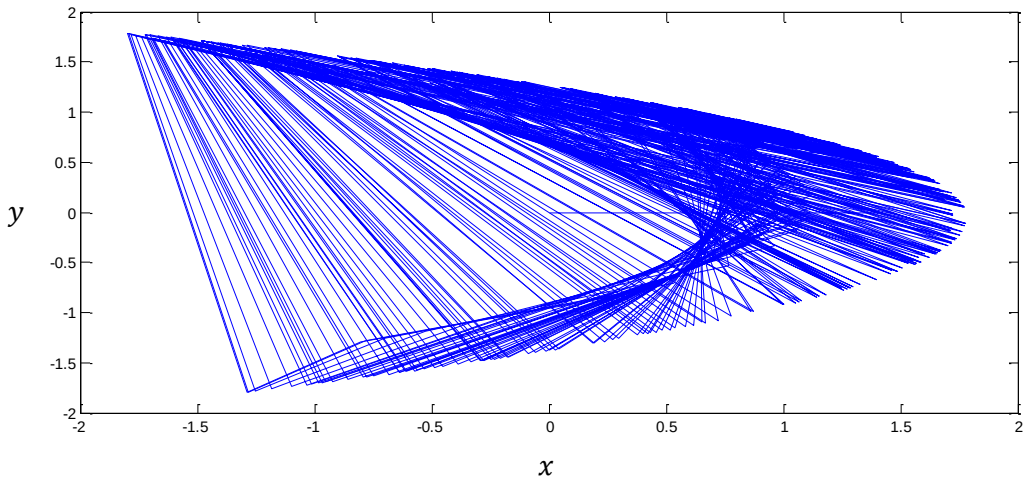


Figure (I.13) : Représentation des états $x\ et\ y$ du modèle de Hénon dans le plan de phase.

La **Figure (1.14)** illustre la propriété de sensibilité aux conditions initiales avec une différence de l'ordre de $\sigma = 10^{-6}$.

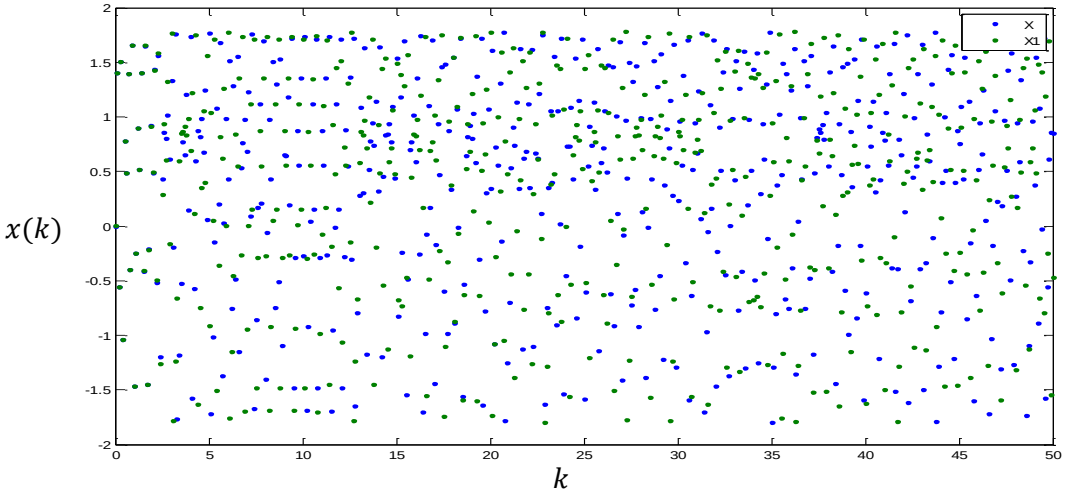


Figure (1.14) : Sensibilité aux conditions initiales du modèle de Hénon.

I.7 Domaines d'apparitions du chaos [10]

Le chaos est retrouvé dans divers domaines. Son analyse permet la compréhension du comportement des systèmes on le retrouve dans :

I.8.1 Météorologie

L'application des modèles chaotiques au mouvement atmosphérique et océanique semble une approche prometteuse. Elle est utilisée pour la reconnaissance des zones relativement stables et permet de fixer plus précisément la limite validité des prévisions. Pour un état très instable, ça varie entre quelque jours à une semaine, et voir plus pour un état stable.

I.8.2 Biologie

Son intérêt est d'expliquer des phénomènes plus variés :

- Description des particularités des rythmes biologiques avec la présence de deux oscillateurs, horloge interne et rythme externe.
- Détection d'un certain nombre de maladies provenant d'une très grande instabilité du corps :
- un rythme cardiaque trop irrégulier signale l'approche d'un accident cardio vasculaire.
- l'épilepsie et la maladie de Parkinson entrainent une augmentation de périodicité des systèmes nerveux.

I.8.3 Aéronautique

Utilisé pour l'adaptabilité et la rapidité de réaction, les régimes chaotiques sont intéressants dans le domaine de l'aviation :

- Pour un avion militaire, un avion proche de régime chaotique aura des vitesses de réaction des plus rapides.
- Pour l'aviation civile, le comportement quasi chaotique signifie des manœuvres plus faciles donc à moindre cout en carburant.

I.8.4 La cryptographie

Après avoir était qu'une théorie sans utilité, le chaos a trouvé une fonction dans la sécurisation des donné transmise par différents moyens ou voix. Mais la synchronisation de deux systèmes était une opération impossible vu leurs sensibilités aux conditions initiales.

I.8 Conclusion

L'objectif de ce chapitre est de donner quelques généralités sur les systèmes chaotiques. Nous avons parlé de l'historique et de l'évolution de la théorie du chaos. Ainsi, nous avons défini quelques notions utiles pour la résolution des systèmes chaotiques. Ensuite, nous avons donné leurs propriétés. Enfin, nous avons cité les principaux domaines d'utilisation de ces systèmes. Le chapitre suivant sera consacré à l'étude de la synchronisation des systèmes chaotiques.

Chapitre II

Synchronisation du chaos et transmission sécurisée de données

II.1 Introduction

L'utilisation du chaos pour sécuriser l'information est un domaine qui a débuté dans les années 1990. Dans les différentes applications. Les signaux chaotiques servent à véhiculer l'information et à réaliser le cryptage par différentes techniques que ce soit par addition, inclusion ou commutation. Le but reste le même à savoir transmettre l'information de façon sure et confidentielle.

Il restait l'obstacle de synchronisation des systèmes chaotiques qui s'avère une opération ardue vu leurs caractères imprévisibles, et leurs extrêmes sensibilités aux conditions initiales, qui à l'origine fessaient tout leur intérêt dans ce domaine. Plusieurs chercheurs se sont penchés sur la question, et ont proposé des méthodes de synchronisations plus ou moins intéressantes. Les premiers étaient Pecora et Carol [11] qui ont développé la méthode dite synchronisation identique.

D'autres méthodes ont vu le jour ensuite, on cite la synchronisation par observateur, synchronisation par boucle fermée et la synchronisation impulsive.

Dans ce chapitre nous allons présenter les différentes méthodes utilisées dans la synchronisation du chaos ainsi que celles utilisées pour le cryptage de l'information.

II.2 La synchronisation [12]

Synchrone est un terme grec, il est composé de « syn » qui signifie (avec) et de « chrone » qui signifie (temps), on dit que deux systèmes sont synchronises s'ils présentent un comportement identique aux mêmes instants.

Ce phénomène a été observé pour la première fois au début de XVIIème siècle quand le mathématicien hollandais Huggens a remarqué en étudiant deux horloges de fréquence légèrement différente, celles-ci affichent la même heure en les reliant l'une à l'autre avec un morceau de bois, des exemples de synchronisation existe dans la nature et dans le domaine technique.

Synchroniser deux systèmes dynamiques consiste à avoir une erreur nulle entre l'évolution des deux dynamiques ou cours du temps.

Soient les deux systèmes suivants :

$$\begin{cases} S_1: \quad \dot{x} = f_1(x) \\ S_2: \quad \dot{x}' = f_2(x') \end{cases} \tag{II.1}$$

Les deux systèmes sont dits synchronisés si :

$$\lim_{t\to\infty} e(t) = \lim_{t\to\infty} |x(t) - x'(t)| = 0 \tag{II.2}$$

Avec x et $x' \in \mathbb{R}^n$ et $f_1, f_2 \in \mathbb{R}^n * \mathbb{R}^n \to \mathbb{R}$ sont des champs de vecteur non linéaires, et $e(t)$ est l'erreur de synchronisation.

L'utilisation des circuits identiques dans la synchronisation des systèmes chaotiques est la base des méthodes traditionnelles, on supposera deux systèmes chaotiques identiques qui oscillent de manière indépendante, s'ils échangent de l'énergie entre eux, action de « couplage », ils finiront par avoir un comportement commun, ils se synchroniseront par « couplage unidirectionnel » ou « couplage bidirectionnel ».

D'autres méthodes ont été proposées ainsi pour la synchronisation unidirectionnelle on citera la synchronisation impulsive [6], par boucle fermée [3] et celle par décomposition du système [11], en plus de ces méthodes on a celle qui est caractérise par l'auto synchronisation qui est la méthode de Pecora-carol.

II.2.1 Méthodes de synchronisation

Il existe plusieurs méthodes de synchronisation :

II.2.2.1 Synchronisation bidirectionnelle et unidirectionnelle

Cette méthode consiste à coupler deux systèmes de manière à ce qu'ils puissent échanger de l'énergie entre eux. Dans le couplage unidirectionnel l'échange se fait dans un seul sens à l'aide d'un élément de couplage (ex : résistance), Dans le couplage bidirectionnel l'énergie circule dans les deux sens, les deux types de couplage peuvent être utilisés pour des systèmes non identiques [2].

II.2.2.2 Synchronisation par boucle fermée

Le principe de cette méthode consiste à injecter l'erreur d'estimation dans le récepteur, en contre réaction pour corriger l'évolution du récepteur et afin de réaliser la synchronisation, cette méthode peut être appliqué pour les systèmes non identiques [13].

Soient les deux systèmes suivants :

Emetteur :

$$\begin{cases} \dot{x} = f(x) \\ y = h(x) \end{cases} \quad \text{(II.3)}$$

Récepteur :

$$\begin{cases} \dot{\hat{x}} = f(\hat{x}) + g(y - \hat{y}) \\ \hat{y} = h(\hat{x}) \end{cases} \quad \text{(II.4)}$$

Avec g fonction de l'erreur entre y et $\hat{y}$, g est choisie de telle sorte à garantir la synchronisation.

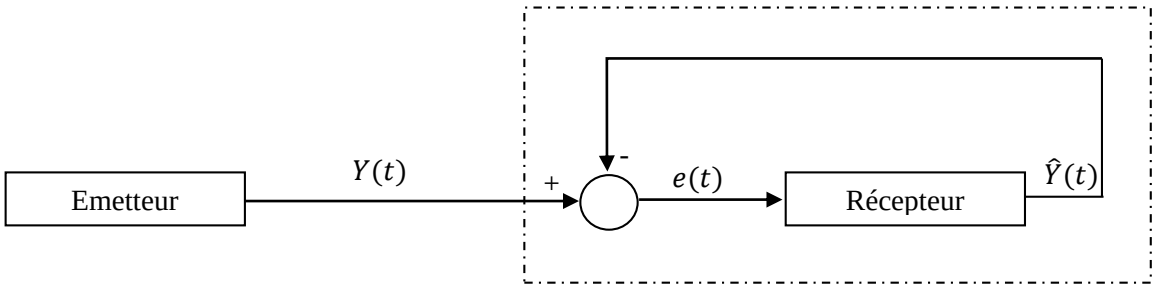


Figure (II.1) : Principe de la synchronisation par boucle fermée.

L'utilisation la méthode de synchronisation par boucle ouvert provoque une sensibilité aux variations des paramètres, afin d'éviter cet inconvénient, la synchronisation par boucle fermée a été proposées.

II.2.2.3 Synchronisation par décomposition de systèmes [6]

Certains systèmes chaotiques peuvent se décomposer en deux sous-systèmes, l'un maitre et l'autre esclave, ces derniers peuvent se synchroniser en les couplant avec un signal commun, l'avantage de cette méthode réside dans le fait que celle-ci présente une solution simple et performante, le but de cette méthode est qu'après un régime transitoire, le système esclave doit reproduire l'état du maître.

considerons le système chaotique suivant :

$$\begin{cases} \dot{x} = f(x) \\ y = h(x) \end{cases} \quad \text{(II.5)}$$

Ou $x = [x_1 \dots \dots \dots x_n]^T$ est le vecteur d'état.

On divisera le système initial en deux sous systèmes (S_1, S_2) :

$$\begin{cases} S_1 = \dot{X}_1 = F_1(x_1, x_2) \\ S_2 = \dot{X}_2 = F_2(x_1, x_2) \end{cases} \quad \text{(II.6)}$$

Ensuite nous allons concevoir un nouveau sous système S'_2 qui présente une même dynamique que S_2 et dont l'entrée est x_1.

$$S'_2 = \dot{\hat{x}}_2 = F_2(x_1, \hat{x}_2) \quad \text{(II.7)}$$

Le sous système S'_2 est un candidat qui peut se synchroniser avec la dynamique complète initiale, une synchronisation parfaite peut s'accomplir si ce dernier est stable,

ce qui veut dire que l'ensemble des coefficients de Lyapunov du sous-système S'_2 sont négatifs.

On appellera le système (S_1, S_2) maître et le sous-système (S'_2) esclave.

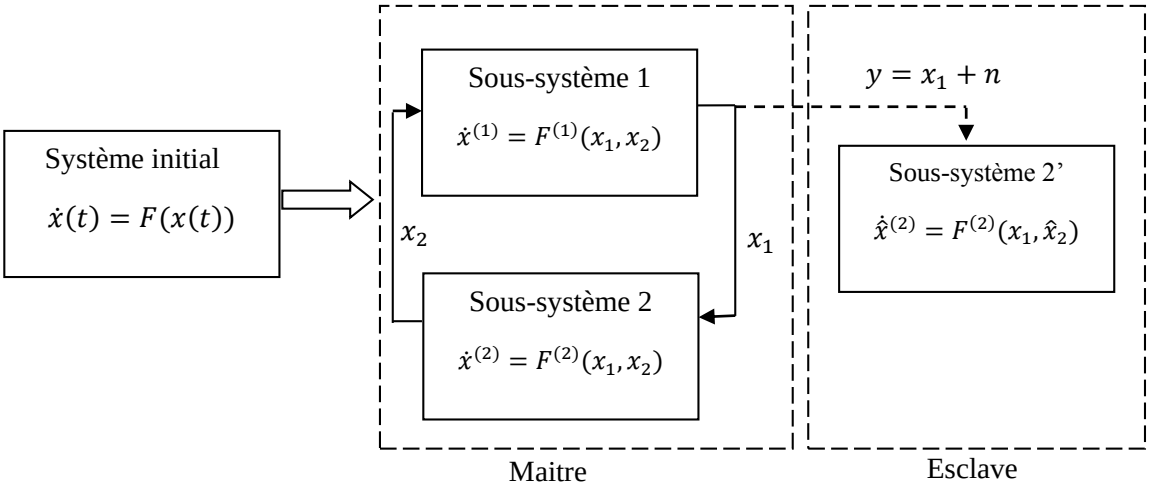


Figure (II.2) : Principe de la synchronisation identique

II.2.2.4 Synchronisation à l'aide d'observateur [13]

La connaissance des entrées, des sorties et du modèle d'un système dynamique permet la reconstruction d'un ou plusieurs états du système qui ne peuvent être mesurés directement, soit à cause de leur inaccessibilité ou par économie.

La synchronisation par observateur consiste à construire un système esclave qui soit un observateur du système maitre, et qui va permettre d'avoir une évolution identique.

Dans le cas non linéaire, le problème de la conception d'un observateur est défini comme suit :

Soient les deux systèmes suivant :

$$\begin{cases} S_1 = \dot{x} = f(x) \\ S_2 = \dot{\hat{x}} = \hat{f}(\hat{x}) \end{cases} \quad \text{(II.8)}$$

Si les systèmes S_1 et S_2 sont synchronisé on aura :

$$\lim_{t\to\infty} |x(t) - \hat{x}(t)| = 0 \quad \text{(II.9)}$$

$x(t)$: État du système.

$\hat{x}(t)$: État estimé.

Le principe de la synchronisation par observateur est illustré pas la figure suivante :

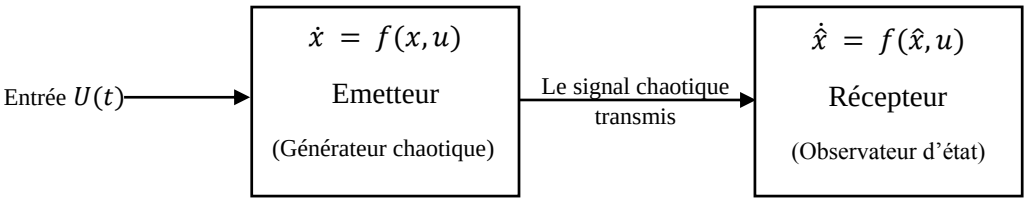


Figure (II.3) : Principe de la synchronisation à l'aide d'observateur

a. Observabilité :

Soit le système :

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \\ x(t_0) = x_0 \end{cases} \quad \text{(II.10)}$$

Où $x \in \mathbb{R}^n, y \in \mathbb{R}^p, u \in \mathbb{R}^m$, $A, C, B \; et \; D$ des matrices de dimension appropriées.

Le système est dit complètement observable, s'il existe un temps finie $t_1 > t_0$, tel que la connaissance de l'entrée $u(t)$ et de la sortie $y(t)$ pour $t \in [t_0, t_1]$ permet de reconstruire l'état $x(t)$.

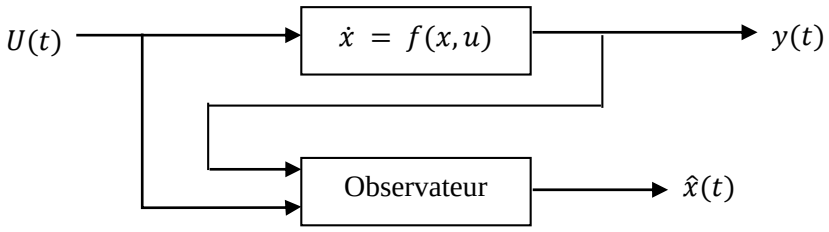


Figure (II.4) : synthèse d'observateur

i. Observabilité des systèmes linéaires :

Soit le système linéaire suivant :

$$\begin{cases} \dot{x}(t) = Ax(t) \\ y = Cx(t) \end{cases} \quad \text{(II.11)}$$

Pour que le système soit complètement observable, il faut que le rang d'observabilité de la matrice $O = [C\ CA \ldots CA^{n-1}]^T$ soit égal à la dimension du système.

$$\text{Rang(O)= Rang}\begin{bmatrix} C \\ CA \\ . \\ . \\ . \\ CA^{n-1} \end{bmatrix} = n \qquad \text{(II.12)}$$

ii. Observabilité des systèmes non linéaires [15] **:**

- **Cas continu :**

Soit le système non linéaire suivant :

$$\begin{cases} \dot{x}(t) = f(x) + g(x)u(t) \\ y(t) = h(x) \end{cases} \qquad \text{(II.13)}$$

La dérivée de lie est utilisée pour définir l'observabilité d'un système non linéaire, elle est définie comme suit :

$$Lfh(x) = \sum_{i=1}^{n} \frac{\partial h(x)}{\partial x_i} f_i(x) = \frac{\partial h(x)}{\partial x_1} f_1(x) + \frac{\partial h(x)}{\partial x_2} f_2(x) + \cdots \frac{\partial h(x)}{\partial x_n} f_n(x) \qquad \text{(II.14)}$$

Avec :

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ . \\ . \\ . \\ f_3(x) \end{bmatrix}$$

Le système (II.13) doit satisfaire la condition du rang d'observabilité $rang(M) = n$

Avec :

$$M = \left[dh; dL_f h; \ldots\ldots\ldots dL^{n-1}{}_f h\right]$$

$$\text{Rang(M)= Rang}\begin{bmatrix} dh \\ dL_f h \\ . \\ . \\ . \\ dL^{n-1}{}_f h \end{bmatrix} = n \qquad \text{(II.15)}$$

Où n est la dimension du système.

- **Cas discret :**

Soit le système non linéaire à temps discret suivant :

$$\begin{cases} x_{k+1} = f(x_k, u_k) \\ y_k = h(x_k) \end{cases} \qquad \text{(II.16)}$$

Ou $x_k \in \mathbb{R}^n$, $y_K \in \mathbb{R}^p, u_k = (u_{1k}, \dots, u_{mk})^T \in \mathbb{R}^m$.

Comme pour le cas continu, l'observabilité des systèmes en temps discret se vérifie par le rang d'observabilité.

$$\dim\big(doh(x_0)\big) = n \qquad \text{(II.17)}$$

Ceci peut être reformulé comme suit :

$$rang\ [span\{dh, d(f\ o\ h), \dots, d(f^{n-1}\ o\ h)\}] = n \qquad \text{(II.18)}$$

Avec n dimension du système.

b. Observateur de Luenberger :

Soit le système suivant :

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y = Cx(t) \end{cases} \qquad \text{(II.19)}$$

Et on a l'observateur dynamique de cette forme :

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L(y(t) - \hat{y}(t)) \\ \hat{y} = C\hat{x}(t) \end{cases} \qquad \text{(II.20)}$$

L'évolution de l'état est corrigée grâce au modèle en fonction de l'écart entre la sortie mesurée et la sortie reconstruite par l'observateur $(y(t) - \hat{y}(t))$.

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L(y(t) - C\hat{x}(t)) \\ \dot{\hat{x}}(t) = (A - LC)\hat{x}(t) + Bu(t) + Ly(t) \end{cases} \quad \text{(II.21)}$$

L'état x en fonction de la commande u et des mesures y est reconstruit par l'observateur, et la matrice de gain L est choisie de manière à ce que l'erreur converge exponentiellement vers zéro.

$$\lim_{t\to\infty} e(t) = \lim_{t\to\infty}(x(t) - \hat{x}(t)) = 0 \quad \text{(II.22)}$$

Pour que le système soit stable, et en utilisant la technique de placement de pôles, on choisit le gain L de l'observateur de telle sorte que les valeurs propres de la matrice $(A - LC)$ soient dans le demi plan complexe gauche (à partie réel négative), de ce fait on choisit une dynamique d'erreur plus rapide que celle de processus.

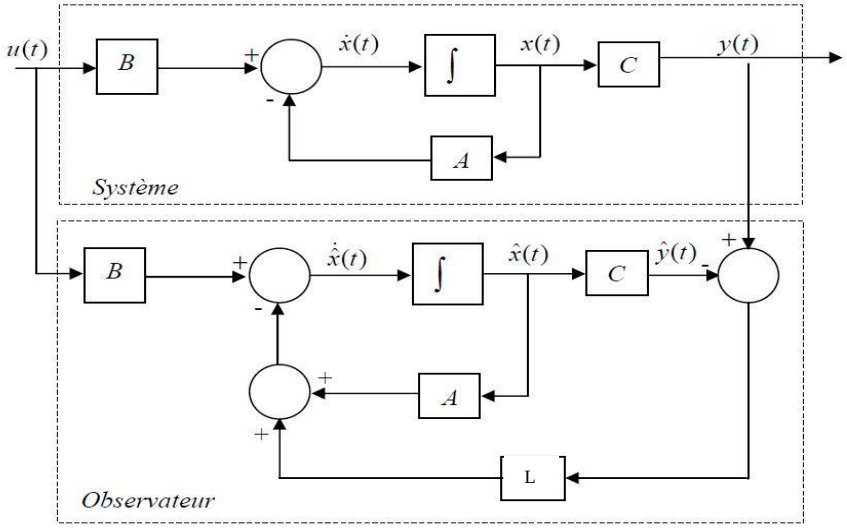


Figure (II.5) : Schéma de l'observateur de Luenberger.

c. Observateurs à modes glissants [14]

Un observateur a modes glissants est un observateur dont le terme correcteur est une fonction sgn, Il s'agit de contraindre, à l'aide des fonctions discontinues, les dynamiques du système à converger sur une "*surface de glissement*".
Soit le système :

$$\begin{cases} \dot{x} = f(x) + g(x)u \\ y = h(x) \end{cases} \quad \text{(II.23)}$$

L'observateur à modes glissants pour ce système s'écrit de la façon suivante :

$$\begin{cases} \dot{\hat{x}} = f(\hat{x}) + g(\hat{x})u + Ksgn(y - \hat{y}) \\ \hat{y} = h(\hat{x}) \end{cases} \quad \text{(II.24)}$$

Où K est une matrice de gain de dimension $n * p$ Dans ce cas, on impose l'évolution des dynamiques du système sur une variété s, sur laquelle l'erreur d'estimation de la sortie $e = (y - \hat{y})$ est nulle. Ainsi, cette erreur converge vers zéro au bout d'un temps fini, et la dynamique du système se réduit de n a $n - p$.

La figure (II.6) illustre le principe d'un observateur a modes glissants.

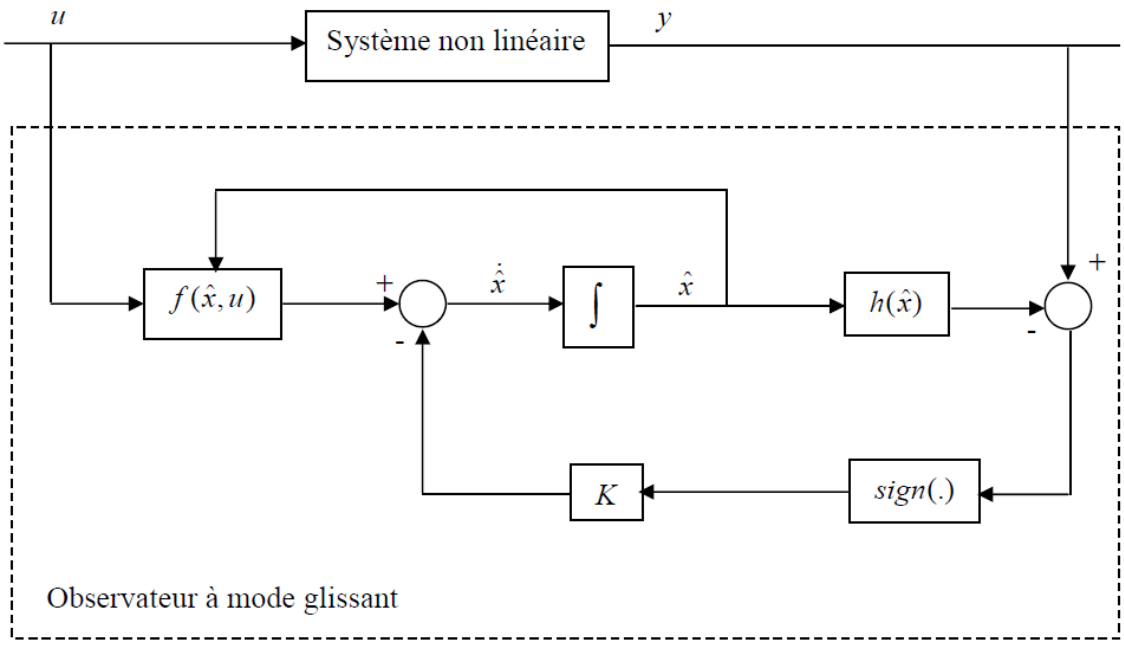


Figure (II.6) : Schéma de l'observateur à modes glissants.

d. Observateur impulsif :

Considérons le système suivant :

$$\begin{cases} \dot{x}_1(t) = f_1(x_1, x_2, t) \\ \dot{x}_2(t) = f_2(x_1, x_2, t) \\ y(t_k) = x_1(t_k) \end{cases} \quad \text{(II.25)}$$

Ou $x_1 \in \mathbb{R}^n, et\ x_2 \in \mathbb{R}^{n-p}$ sont les états du système, $y(t_k) \in \mathbb{R}^n$ est le vecteur de sortie.

Le principe consiste à contraindre l'observateur à suivre l'évolution du système original a des instants (t_k), un état du système est transmis sous forme d'impulsion pour réduire les redondances du signal, l'observateur prend la forme mathématique suivante [6]:

$$\begin{cases} \dot{\hat{x}}_1(t) = f_1(\hat{x}_1, \hat{x}_2, t) \\ \dot{\hat{x}}_2(t) = f_2(\hat{x}_1, \hat{x}_2, t) \\ \hat{x}_1(t_k) = x_1(t_k) \end{cases} \quad \text{(II.25)}$$

Le système d'erreur d'observation est donné comme suit :

$$\begin{cases} \dot{e}_1(t) = f_1(x_1, x_2, t) - f_1(\hat{x}_1, \hat{x}_2, t) \\ \dot{e}_2(t) = f_2(x_1, x_2, t) - f_2(\hat{x}_1, \hat{x}_2, t) \\ e_1(t_k) = 0 \end{cases} \quad \text{(II.26)}$$

Avec t_k instant ou l'impulsion est appliqué.

II.3 Le chaos dans la transmission sécurisée [16]

Utiliser le chaos pour transmettre de l'information en toute sécurité est possible, les signaux chaotiques sont semblables à un bruit blanc, ce qui permet de noyer des données dans une porteuse chaotique afin de les rendre indéchiffrables, ces dernières années beaucoup de travaux ont été menés pour l'exploration des possibilités qu'offre le chaos dans le domaine du cryptage de l'information.

II.3.1 Définitions [16]

- **Cryptanalyse :**

C'est une étude qui détermine les éventuelles faiblesses des systèmes cryptographiques et ce, en étudiant les probabilités de succès des attaques, le principal objectif de la cryptanalyse est de déchiffrer le message dans le but de le rendre claire, pour cela il est nécessaire de se mettre dans la peau du pirate.

- **Cryptographie**

Est une étude qui consiste à protéger le message à transmettre et ce, en lui appliquant une transformation qui le rend incompréhensible, ce qui est appelé chiffrement.

- **Crypter (chiffrer)**

Et l'opération de transformer un texte claire en un texte codé, le résultat est appelé chiffrement.

- **Décrypter (Déchiffrer)**

C'est l'opération qui consiste à traduire un message chiffré en un message claire, et c'en connaissant la clé, donc seul le destinataire légitime peut déchiffrer le message, car il est le seul détenteur de la clé.

II.3.2 Méthodes de cryptage

Il existe plusieurs méthodes de cryptage par chaos. Parmi ces méthodes on peut citer :

II.3.2.1 Cryptage par addition [17]

Du point de vue, chronologie, cette méthode est la première à être utilisée dans le cryptage par chaos, son principe est simple, il suffit d'ajouter au message utile $U(t)$, une porteuse chaotique, on utilise les mêmes systèmes du côté émetteur et du côté récepteur, le message original est obtenu par soustraction.

La figure (II.7) illustre la méthode de cryptage par addition.

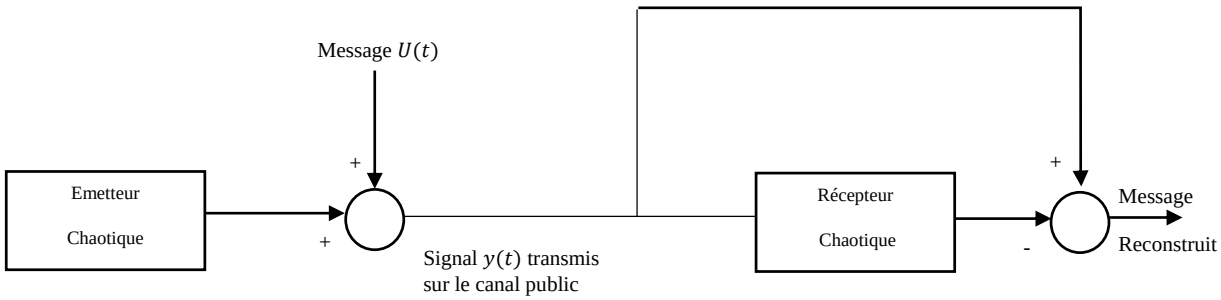


Figure (II.7) : Schéma de la méthode de cryptage par addition.

Cette technique peut être appliquée pour la transmission de messages continus ou discrets. Afin de garantir le secret et pour un cryptage efficace, il faut que l'amplitude du signal $U(t)$ soit inférieure à celle du signal porteuse chaotique.

II.3.2.2 Cryptage par inclusion

Le principe de cette méthode repose sur le fait d'inclure le message utile U(t) dans la structure du système chaotique l'organe émetteur, la récupération du message se fait par observateur à entrées inconnues, ou l'inversion du système émetteur, cette méthode présente beaucoup d'avantage et nécessite un seul canal de transmission [6].

La figure (II.7) illustre la méthode de cryptage par inclusion.

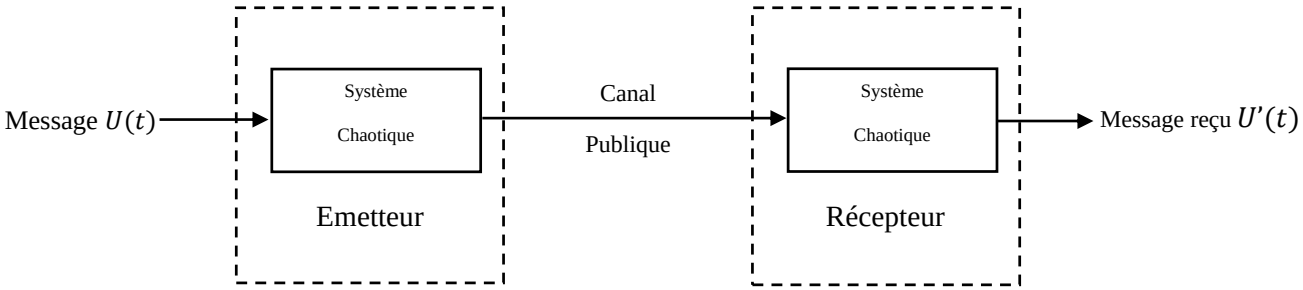


Figure (II.7) : Schéma de la méthode de cryptage par inclusion.

II.3.2.3 Par commutation chaotique [12]

La modulation chaotique, est aussi connue sous le nom de ‘’chaos shift-kaying ‘’ou ‘’chaotique switching ‘’. Du côté émetteur, à chaque symbole M_k correspond un émetteur chaotique S_i avec $i = 1 \dots n$. Le signal est transmis par le canal public. Le côté récepteur est constitué de n systèmes R_i, le signal reçu est soustrait à chaque système R_i. Après comparaison on détecte le symbole ou le caractère reçu.

Il existe deux méthodes de détection, cohérente et non cohérente.

La méthode non cohérente utilise des approches statistiques, elle analyse la corrélation entre $y_s(t)$ et $y_r(t)$. La méthode cohérente nécessite la synchronisation des systèmes S_i avec les systèmes R_i, les méthodes de synchronisations utilisées sont principalement celles utilisant les observateurs d'état.

La figure (II.9) illustre le principe de commutation pour un système binaire.

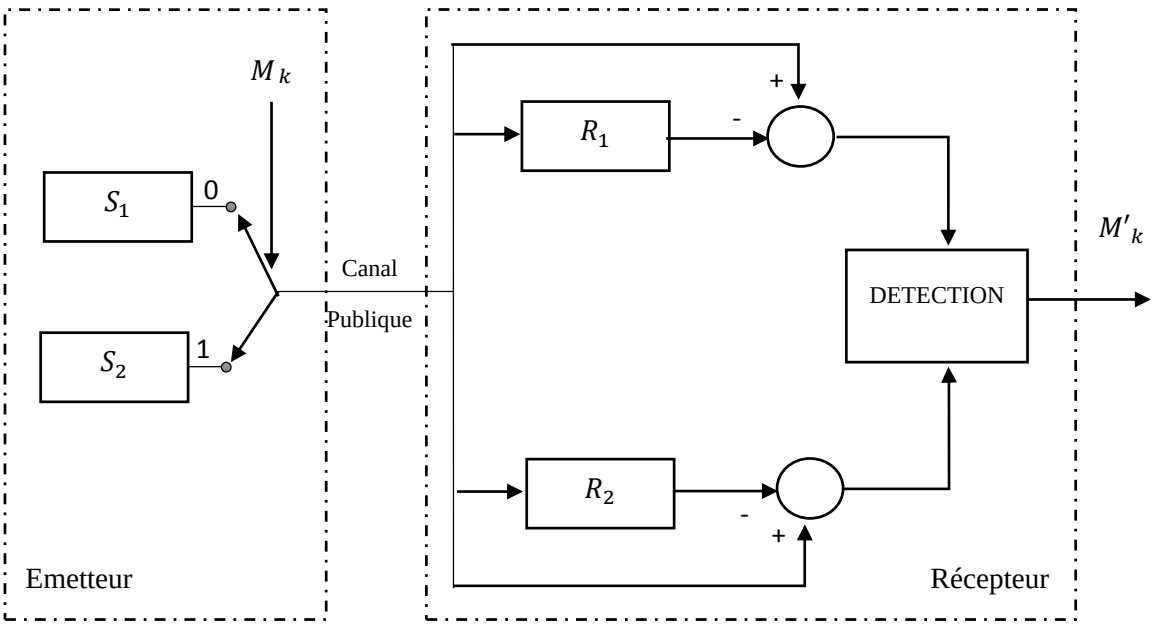


Figure (II.9) : Schéma de la méthode de cryptage par commutation.

S_1 Et S_2 sont des générateurs chaotiques, M_k message à transmettre en binaire, M_r message reçu.

R_1 , R_2 Observateurs respectifs des systèmes S_1 et S_2.

II.3.2.4 Transmission à deux voies [19]

Le principe de cette méthode consiste à transmettre le signal de synchronisation séparément du signal porteur du message, un seul générateur chaotique est utilisé, le principe est illustré par la figure (II.10).

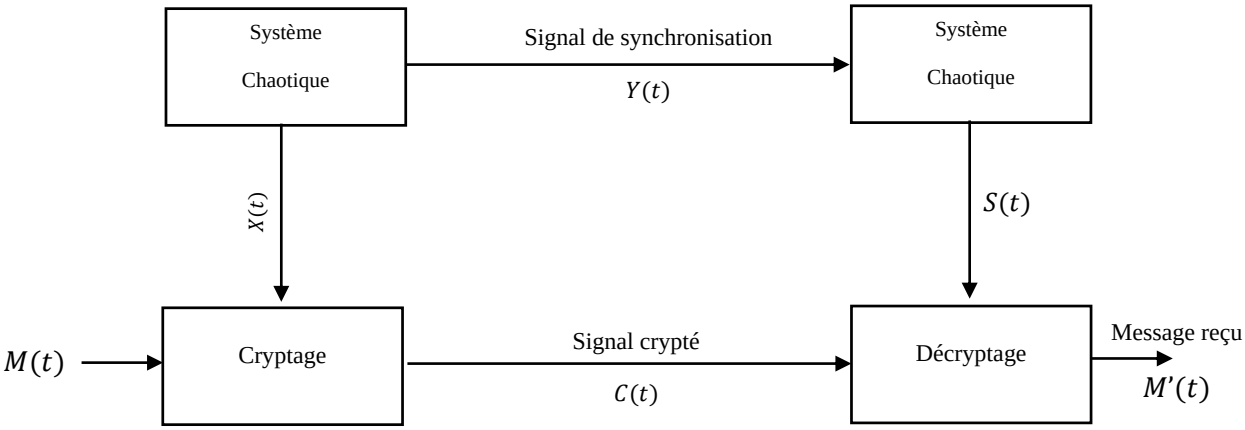


Figure (II.10) : Schéma de la méthode de la transmission à deux voies.

Tout l'intérêt de cette méthode réside dans l'indépendance entre la synchronisation et le cryptage, l'information transmise n'a aucune influence sur l'opération de synchronisation donc elle garantit une meilleure qualité du signal reçu. D'autre part elle ajoute un degré de complexité pour une éventuelle tentative de piratage.

II.4 Les Objectifs du cryptage

- Rendre une information confidentielle à toute personne autre que le destinataire.
- Connaitre l'origine des informations transmises.
- S'assurer qu'une information n'a pas été altérée ou détruite d'une manière accidentelle ou volontaire lors de la transmission.
- Interdire la possibilité d'affirmer avoir reçu ou émis une information.
- Garantir l'identité de l'expéditeur et ce, en s'assurant que les interlocuteurs sont bien ceux qu'ils prétendent être.

II.5 Conclusion

La synchronisation du chaos a connu un nouvel essor depuis quelque années, Pecora et Carroll ont démontré que deux systèmes dynamiques identique peuvent se synchroniser parfaitement en l'absence de bruit, et ce malgré leur extrêmes sensibilités aux conditions initiales, Un signal chaotique peut être généré en utilisant des circuits simples et peu couteux, et ça constitue l'un des aspects les plus intéressants de la transmission par porteuse chaotique.

La définition de la synchronisation et les différentes méthodes utilisées pour la synchronisation ont fait l'objet de la première partie de ce chapitre, dans la deuxième partie nous avons donné quelques notions sur le cryptage et la transmission sécurisée du chaos, enfin nous avons terminé par donner les objectifs du cryptage.

Chapitre III

Synchronisation du système de LOZI

III.1 Introduction

La synchronisation des systèmes chaotiques peut paraitre énigmatique et ambigüe. En effet la synchronisation de ces systèmes présente plus de difficulté contrairement au cas des oscillateurs périodiques.

La synchronisation est rendue possible par le fait qu'un système chaotique est déterministe. Il est donc possible de construire une réplique identique à ce système et d'essayer de les synchroniser de façon à ce que les deux signaux chaotiques issus des deux exemplaires soient identiques. L'opération consiste à rapprocher les trajectoires des deux systèmes jusqu'à ce qu'elles finissent par être confondues.

Dans ce chapitre nous allons synchroniser deux systèmes chaotiques identiques (émetteur et récepteur) à l'aide d'un observateur impulsive. Dans un premier temps, nous allons donner la définition d'un oscillateur. Par la suite, on étudiera un oscillateur chaotique spécifique appelé « système de lozi ».

III.2 L'Emetteur

Un oscillateur électronique va nous servir d'émetteur. Ce dernier génère et envoie un signal chaotique au récepteur via un canal de transmission.

III.2.1 Définition d'un oscillateur électronique

La fonction principale d'un oscillateur électronique est la production d'un signal périodique lors de sa mise sous tension. Il est définit comme étant un montage électronique autonome (sans entrée). La forme du signal qu'il produit peut être sinusoïdale, carré, voire en dent de scie, ou d'une quelconque forme.

III.2.2 Structure générale d'un oscillateur électronique

Un élément actif (circuit amplificateur) associé à un circuit passif (un filtre) sont les composants d'un oscillateur électronique, souvent l'élément actif est choisie comme étant un transistor bipolaire ou un amplificateur opérationnel, sa structure est celle d'un système bouclé dans lequel une fonction du signal de sortie est ramené à l'entrée pour l'auto entretien des oscillations.

Le schéma ci-dessous représente un oscillateur électronique :

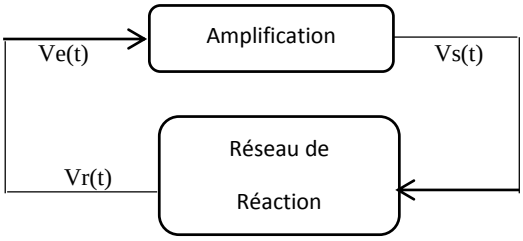


Figure (III.1) : schéma de représentation d'un oscillateur électronique

Pour boucler l'entrée de l'amplificateur il faut avoir le signal $Vr(t)$ identique à celui de $Ve(t)$, cette condition n'est satisfaite que pour une fréquence bien définie appelé fréquence d'oscillations. On reconnait la structure d'un tel oscillateur par le fait que la sortie de l'amplificateur est rebouclé sur l'entré via le réseau de réaction qui est un circuit passif.

III.2.3 Caractérisation du chaos dans le système de lozi

Notre étude se porte sur le système de lozi qui est un oscillateur chaotique, il est régit pas les équations suivantes :

$$\begin{cases} x_1(k+1) = 1 - a|x_1(k)| + bx_2(k) \\ x_2(k+1) = x_1(k) \\ y(k) = x_1(k) \end{cases} \quad \text{(III.1)}$$

Ou a et b sont des constantes positives.

On remarque que le terme x^2 qui apparait dans l'oscillateur de Hénon est remplacé par le terme $|x|$ dans le système de lozi. Notant que les deux systèmes sont linéaires pour $a = 0$.

a) Exposant de Lyapunov

Ce système comporte deux exposants de lyapunov sur tout un intervalle du paramètre a, b étant fixe et égale à 0,5. Le premier exposant varie aux voisinages de $L_1 \approx 0.2618$, et le deuxième exposant qui est négatif varie aux alentours de $L_2 \approx -0.9541$.

Ces résultats démontrent que le système de lozi est de nature chaotique.

La figure (III.3) représente les exposants de lyapunov pour le système de lozi.

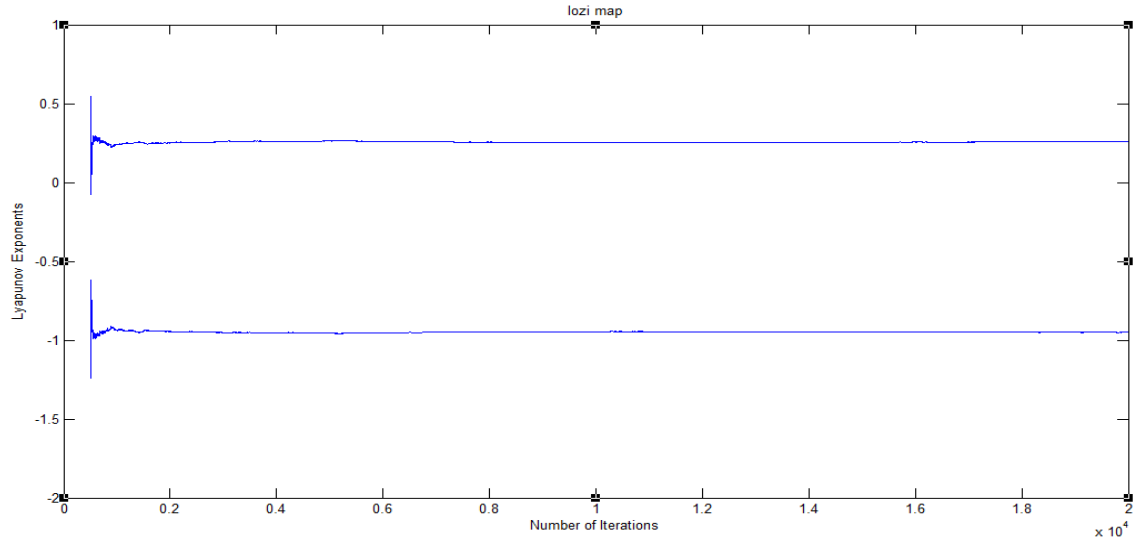


Figure (III.2) : les exposants de lyapunov de système de lozi

b) Spectre de puissance

La transformée de Fourrier de la fonction d'autocorrélation nous donne le spectre de puissance, ce dernier est continu il est riche en fréquences.

La figure $(III.3)$ représente le spectre de puissance du système de lozi.

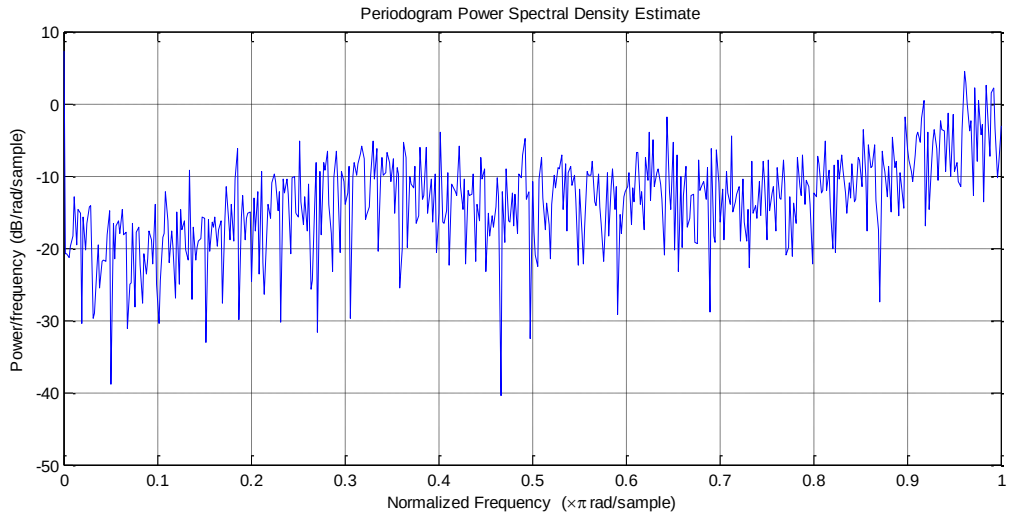


Figure (III.3) : le spectre de puissance pour le système de lozi.

Le spectre de puissance du système de lozi a une bande riche en fréquences.

III.3 Simulation sur Matlab

On a utilisé Matlab Simulink pour la simulation du système de Lozi pour des valeurs de $a = 1.7$ et $b = 0.5$, nous avons obtenu les résultats de simulation donnés aux figures suivantes :

III.3.1 Visualisation des états

- **Etat $x_1(k)$:**

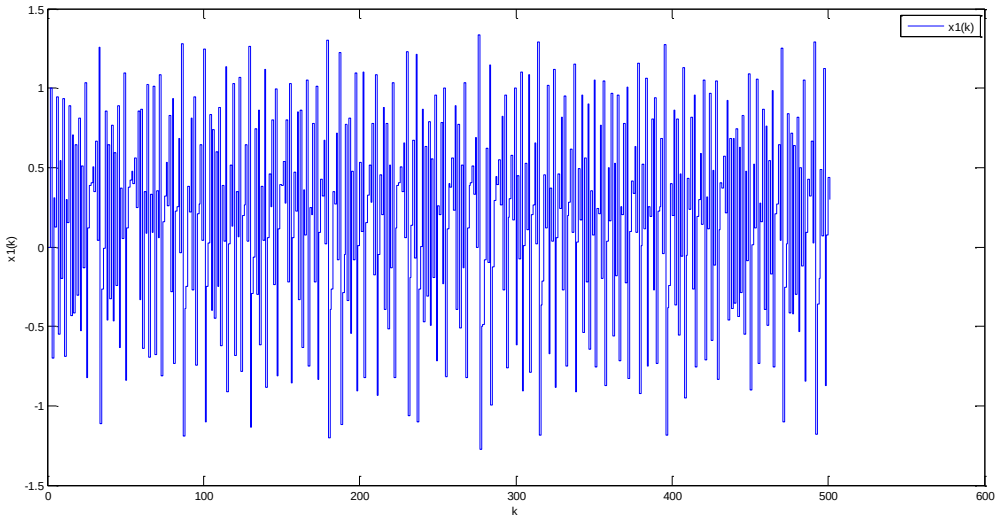


Figure (III.4) : graphe de l'état $x_1(k)$ du système de lozi.

- **Etat $x_2(k)$:**

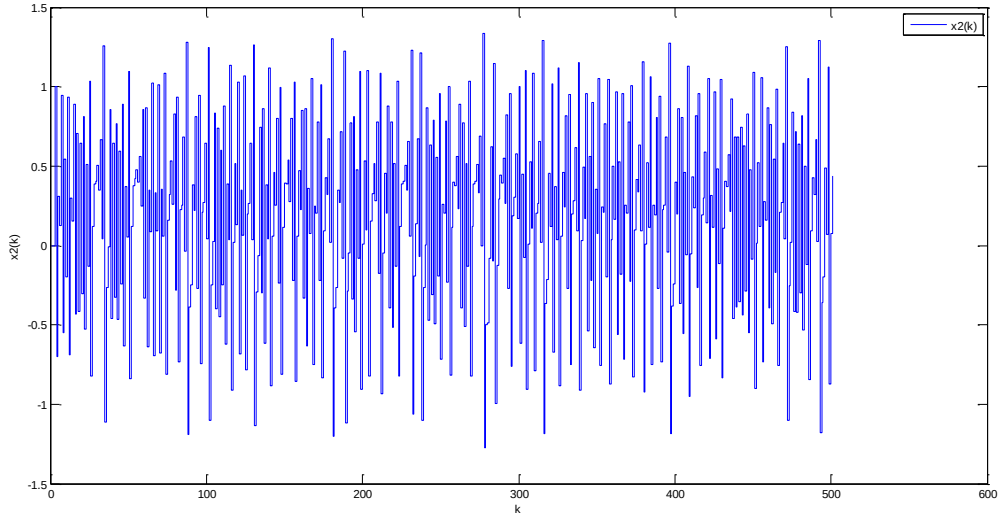


Figure (III.5) : graphe de l'état $x_2(k)$ du système de lozi.

Observation

Les signaux $x_1(k), x_2(k)$ ont des oscillations apériodiques et irrégulières ce qui indique leurs nature chaotique.

III.3.2 Visualisation de l'attracteur

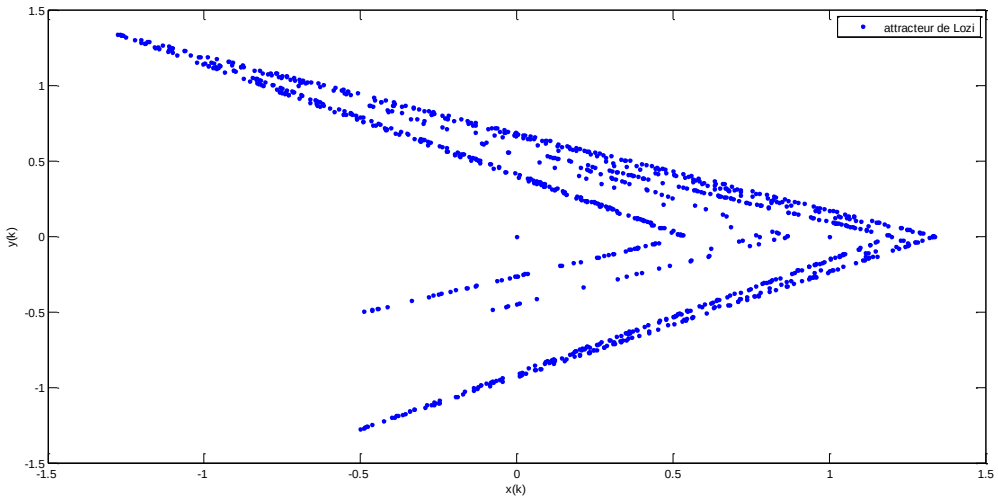


Figure (III.6) : l'attracteur de lozi

La figure est constitué de points avec entre eux des espaces inoccupés. L'objet est dit étrange en raison de sa structure. Une valeur différente de x_0 conduit à une toute autre suite qui après une courte phase, dessine la même image.

III.4 Le récepteur

Un système dynamique est dit observable, si on peut récupérer toute ses grandeurs à partir d'une combinaison de mesures ou de leurs dérivées. En 1997 Nijmeijer et Mareels [19] étaient les premiers à avoir considéré la synchronisation unidirectionnelle de deux systèmes chaotique comme étant un problème d'observateur non linéaire.

Un observateur est utilisé pour estimer les états non mesurables directement d'un système à partir d'un certain nombre de mesures effectuées sur les grandeurs mesurables du système dynamique considéré.

III.4.1 Condition du rang d'observabilité

On étudie l'observabilité d'un système non linéaire en étudiant le rang de la matrice d'observabilité (la Jacobienne). On dit qu'un système est localement observable si et seulement si sa matrice d'observabilité est de rang plein par les colonnes.

$$\begin{cases} x_1(k+1) = 1 - a|x_1(k)| + bx_2(k) \\ x_2(k+1) = x_1(k) \\ y(k) = x_1(k) \end{cases} \quad \text{(III.2)}$$

On a le système (III.2) sous la forme suivante :

$$\begin{cases} x(k+1) = f(x) \\ y(k) = h(x) \end{cases} \quad \text{(III.3)}$$

La matrice d'observation (O) du système (III.2) est définit comme suite :

$$O = \begin{bmatrix} dh \\ dfoh \end{bmatrix}$$

Nous avons

$$dh = [1 \;\; 0]$$

Et

$$foh = 1 - a|x_1(k)| + bx_2(k) \quad \text{(III.4)}$$

Pour $x_1(k) < 0$

$$foh = 1 + ax_1(k) + bx_2(k) \quad \text{(III.5)}$$

$$dhof = [a \;\; b]$$

Pour $x_1(k) > 0$

$$foh = 1 - ax_1(k) + bx_2(k)$$

$$dfoh = [-a \;\; b]$$

La Jacobienne est donnée par:

$$O = \begin{pmatrix} 1 & 0 \\ -a\, sign(x_1) & b \end{pmatrix} \quad \text{(III.6)}$$

$$det(O) = b.$$

Le système (III.2) est observable si $b \neq 0$.

III.4.2 Synchronisation du système de lozi

Le système de lozi représenté par les équations III.12 a pour observateur impulsif :

$$\begin{cases} \hat{x}_1(k+1) = 1 - a|\hat{x}_1(k)| + b\hat{x}_2(k), & k \neq t_i \\ \hat{x}_2(k+1) = \hat{x}_1(k), & k \neq t_i \\ \widehat{x_1}(t_i) = x_1(t_i), & k = t_i \end{cases} \quad \text{(III.7)}$$

Avec t_i ensemble discret des instant de temps tell que $0 < t_1 < t_2 < \cdots < t_i < t_{i+1} < \cdots$, $et\ i \in \mathbb{Z}$

A partir des systèmes (III.2) et (III.6) on obtient le système d'erreur d'observation :

$$\begin{cases} e_1(k+1) = x_1(k+1) - \hat{x}_1(k+1) \\ e_2(k+1) = x_2(k+1) - \hat{x}_2(k+1) \\ e(t_i) = 0 \end{cases} \quad \text{(III.8)}$$

Le schéma du principe est représenté dans la figure suivante :

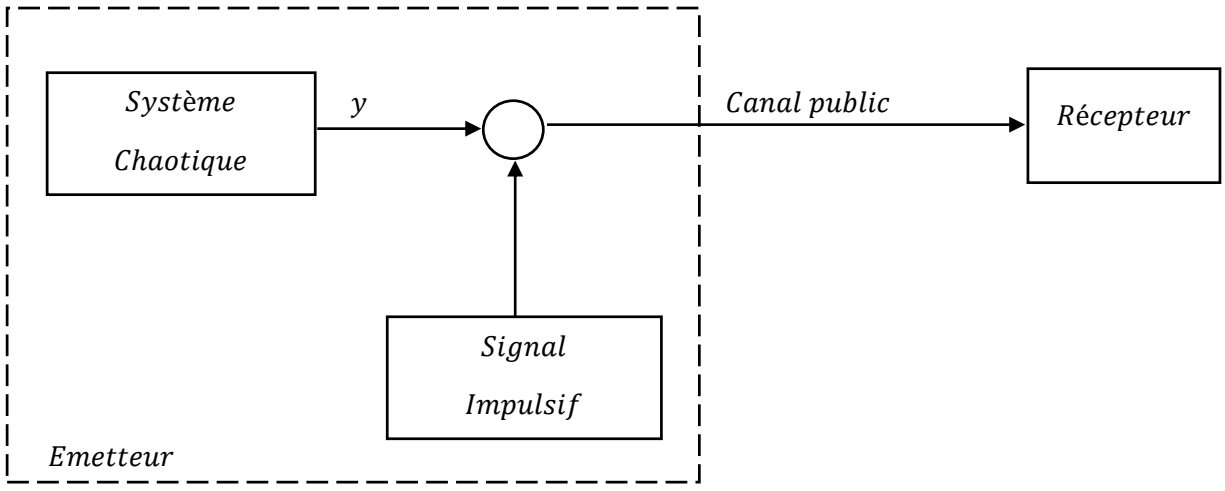


Figure (III.7) : Synchronisation impulsive

La simulation se fait sous Matlab Simulink, on transmet x(k) dont on a étudié l'observabilité, le signal est injecté via un switch à chaque période d de manière à forcer le système récepteur à suivre la trajectoire de l'émetteur.

Pour des valeurs différentes de d on a une synchronisation plus ou moins satisfaisante. Si on prend $d = 3$, on obtient les résultats suivants :

a) Visualisation des états estimés

- Etat $\hat{x}_1(k)$

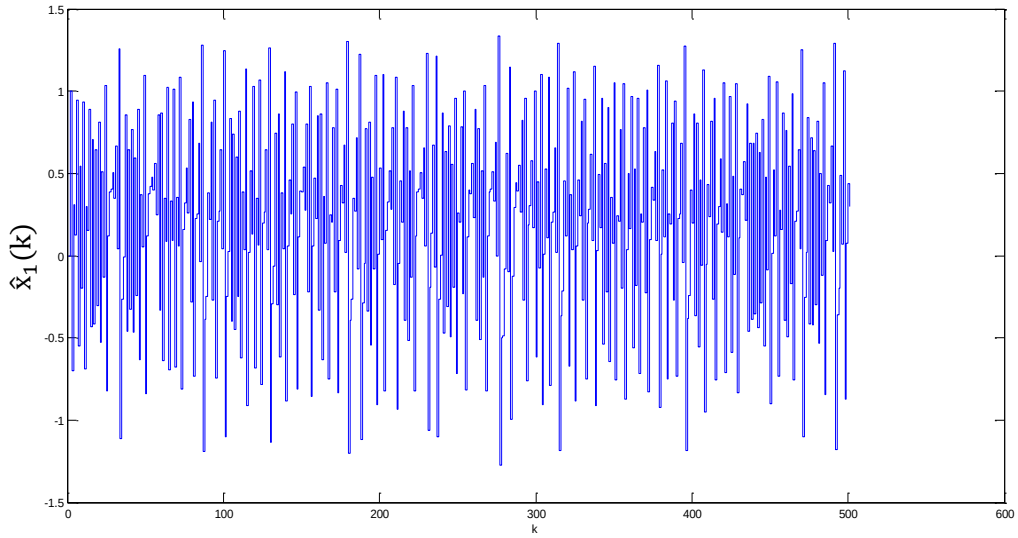


Figure (III.8) : graphe d'état $\hat{x}_1(k)$

- Etat $\hat{x}_2(k)$:

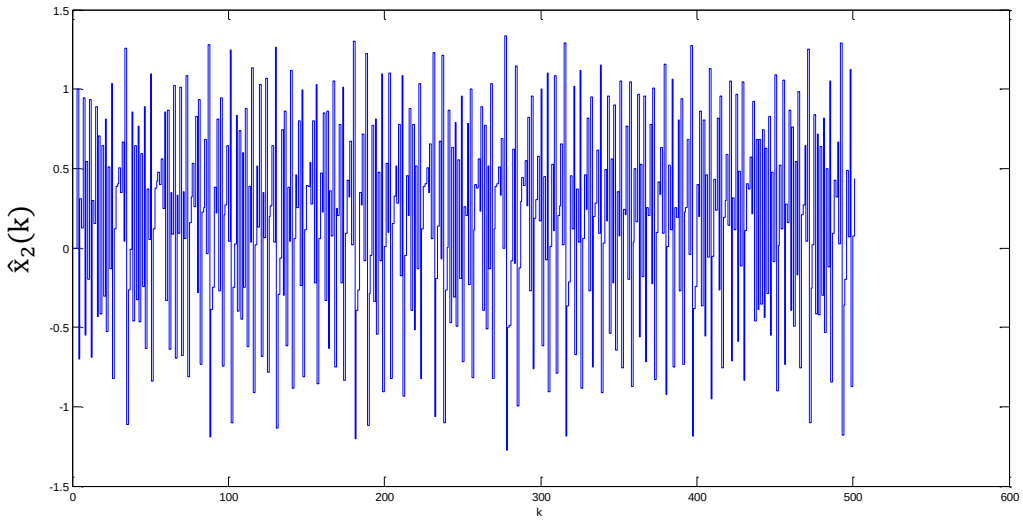


Figure (III.9) : graphe d'état $\hat{x}_2(k)$.

On remarque l'évolution apériodique, les signaux reconstruits sont chaotique.

b) Visualisation des écarts entre les états des deux systèmes

- Erreur entre $x_1(k)$ et $\hat{x}_1(k)$:

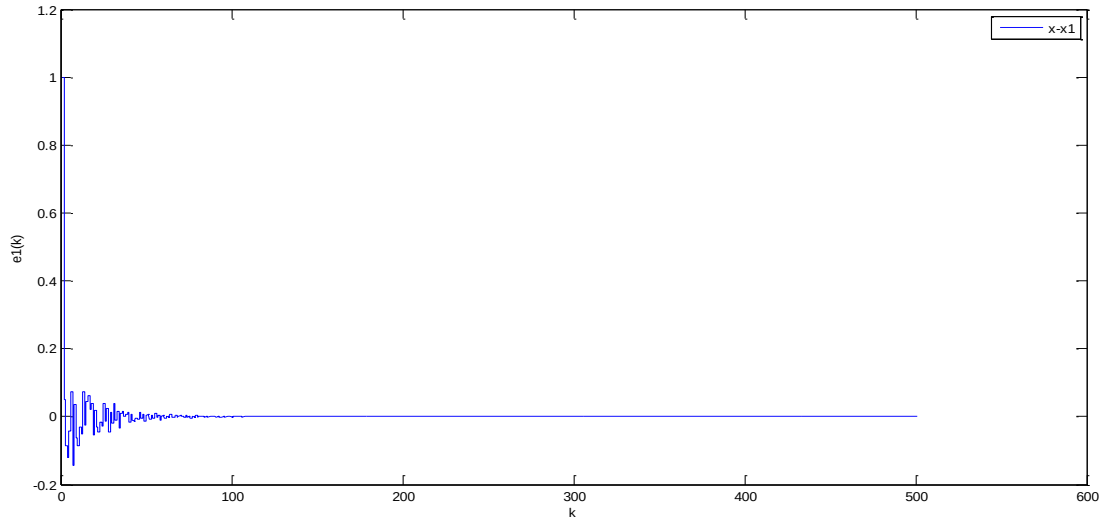


Figure (III.10) : graphe d'écart $(x_1 - \hat{x}_1)$

- Erreur entre $x_2(k)$ et $\hat{x}_2(k)$:

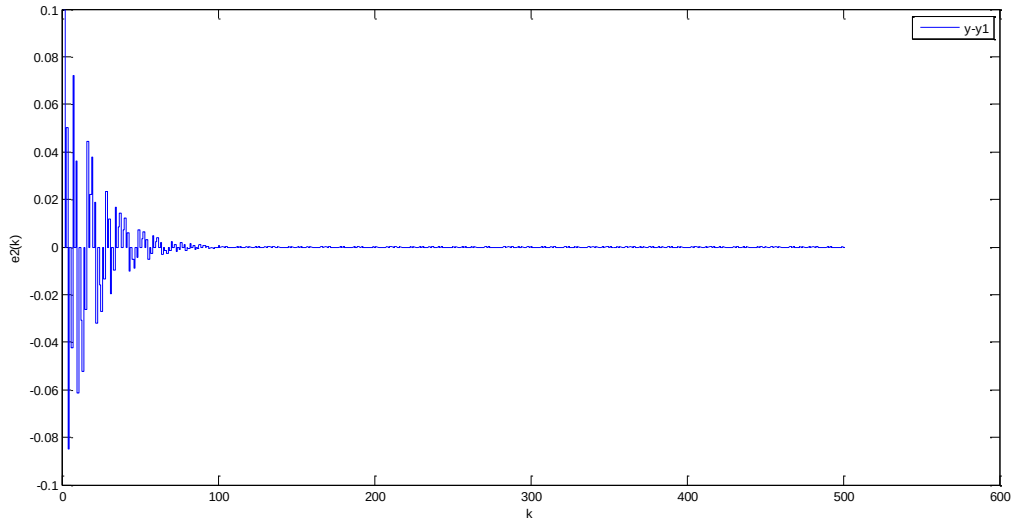


Figure (III.11) : graphe d'écart $(x_2 - \hat{x}_2)$

Observation :

La figure (III.10) et la figure(III.11), montre la synchronisation des deux systèmes, car on constate la convergence des erreurs d'état vers 0 quand k tend vers l'infini.

c) Récupération du message

On prend l'exemple d'un signal sinusoïdale numérique Figure (III.12)**,** que on va additionner a l'un des états ($x_2(k)$) de l'émetteur, du côté récepteur on fait l'opération inverse.

Le message crypté et représenté Figure (III.13), celui récupéré en Figure (III.14)

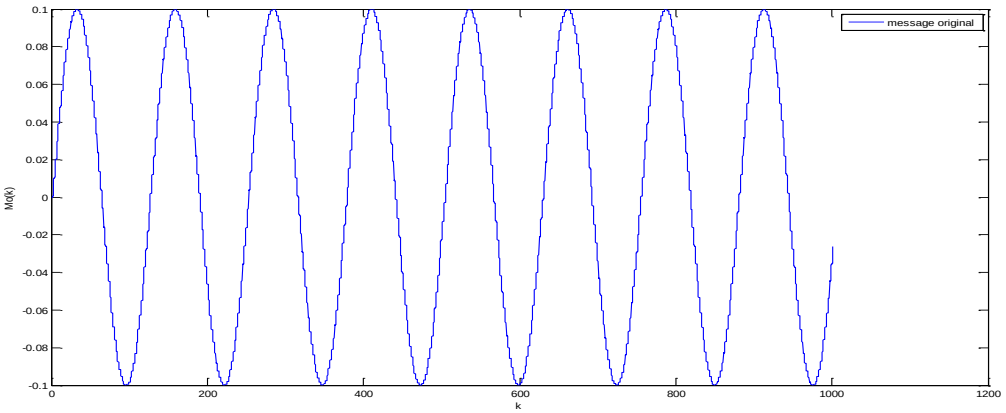


Figure (III.12) : message original

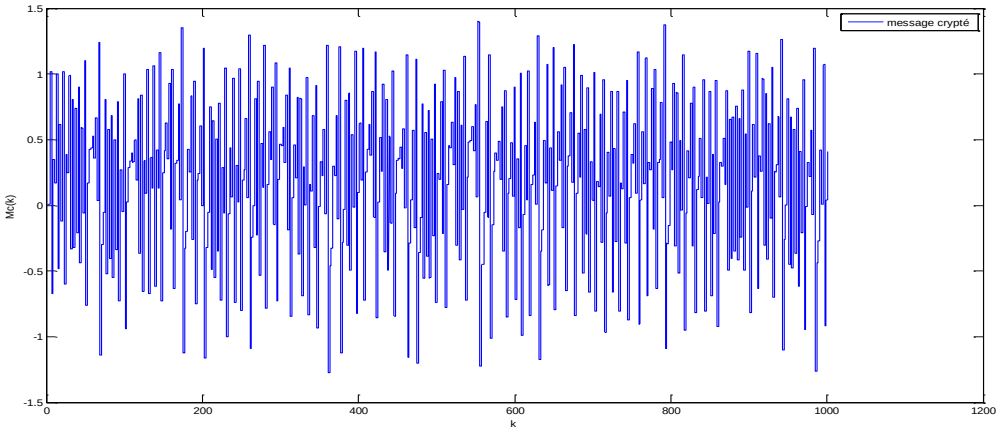


Figure (III.13) : message crypté

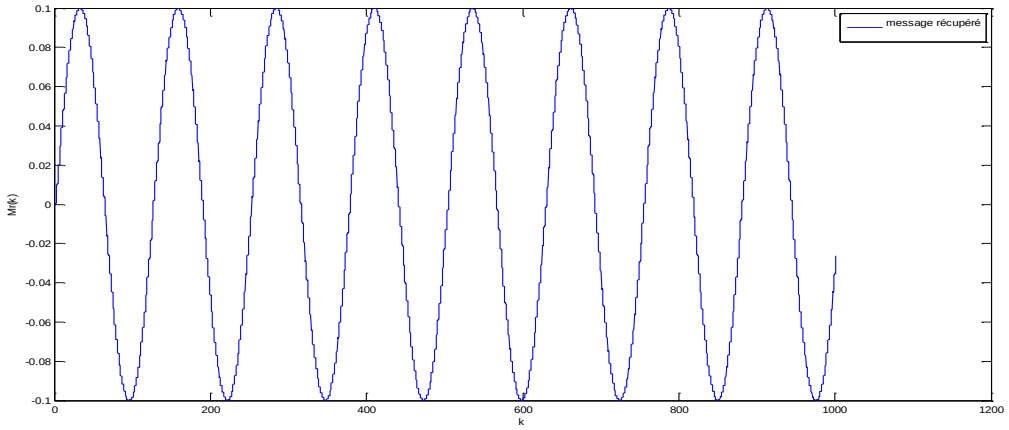


Figure (III.14) : message récupéré

Le message est bien noyer dans le signal chaotique.

III.5 Conclusion

Dans ce chapitre on a pu synchroniser deux systèmes de lozi à l'aide d'un observateur impulsive, afin de transmettre des données sécurisées entre un émetteur et un récepteur de signaux chaotiques. Le cryptage a été fait par addition en noyant le message dans le signal chaotique généré par l'émetteur, une simple soustraction entre le signal émis par l'émetteur et celui reconstruit pas l'observateur impulsif nous a permis de récupérer le message original.

Dans le chapitre suivant nous allons réaliser un dispositif de transmission sécurisée de données sur des cartes Arduino.

Chapitre IV

Réalisation

VI.1 Introduction

Une plateforme de prototypage Arduino correspond à une carte électronique à microcontrôleur et à un environnement de programmation. Cette dernière nous offre la possibilité de réaliser en pratique le système de transmission étudié dans le chapitre précèdent.

Dans notre étude, nous avons choisi d'utiliser l'oscillateur de lozi. L'intérêt de son utilisation se justifie par la simplicité de sa structure, et de son fonctionnement. Une fois les deux oscillateurs synchronisés, on pourra les exploiter pour une transmission de données.

Dans ce chapitre, nous allons dans un premier temps donner un synoptique du schéma de transmission sécurisée adopté. Par la suite, chaque étape correspondant à une fonction particulière est expliquée. Aussi, on abordera, la manière de programmation de la carte avec Matlab Simulink. Nous présentons par la suite, les différents tests et résultats obtenus.

VI.2 Présentation du système de transmission sécurisée de données à base d'oscillateur de lozi sur carte Arduino.

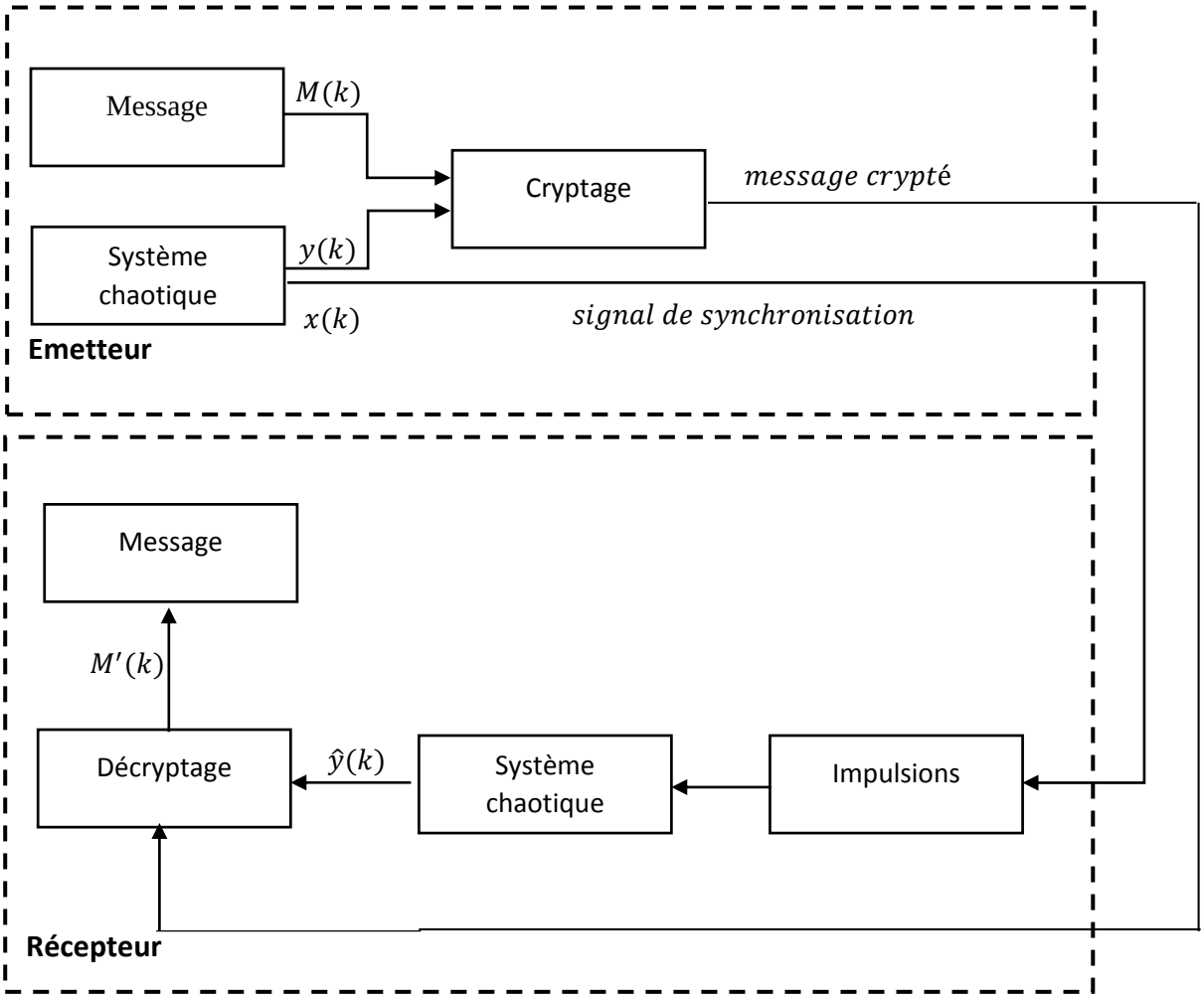
Dans ce qui suit nous allons expliquer le schéma de synchronisation impulsive adopté et ce dans l'objectif de concevoir un système de transmission de données sécurisée composé d'un bloc pour l'émission et d'un autre pour la réception,

L'émetteur est constitué d'une carte Arduino contentant les équations du système de Lozi, elle sert à la génération, et au cryptage du message.

Le récepteur est un système identique au système émetteur il a pour tâche le décryptage du message reçu.

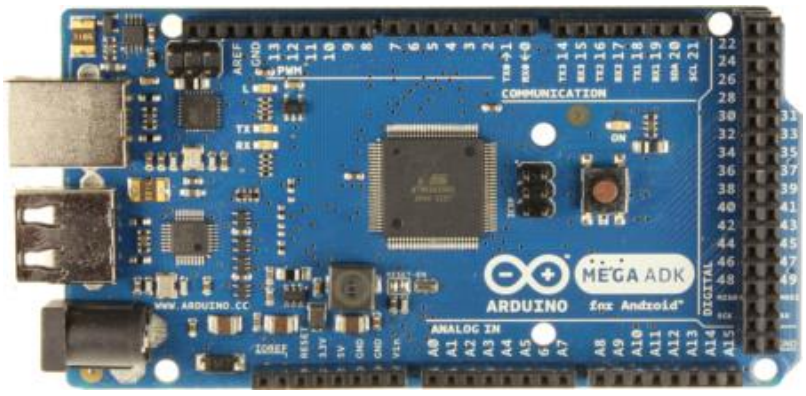
Les deux systèmes sont relier pas une liaison série, un canal sert à l'envoi du signal de synchronisation $y = x\,(k)$, et l'autre à l'envoi du message crypté par addition avec le signal $y(k)$, la période T des impulsions est choisi de manière a assuré une bonne synchronisation.

Le diagramme de la figure (VI.1) illustre le principe de la transmission utilisant la synchronisation impulsive.



Figure(VI.1) : schéma synaptique illustrant la synchronisation impulsive

VI.3 Arduino



Figure(VI.2) : Carte arduino Mega ADK

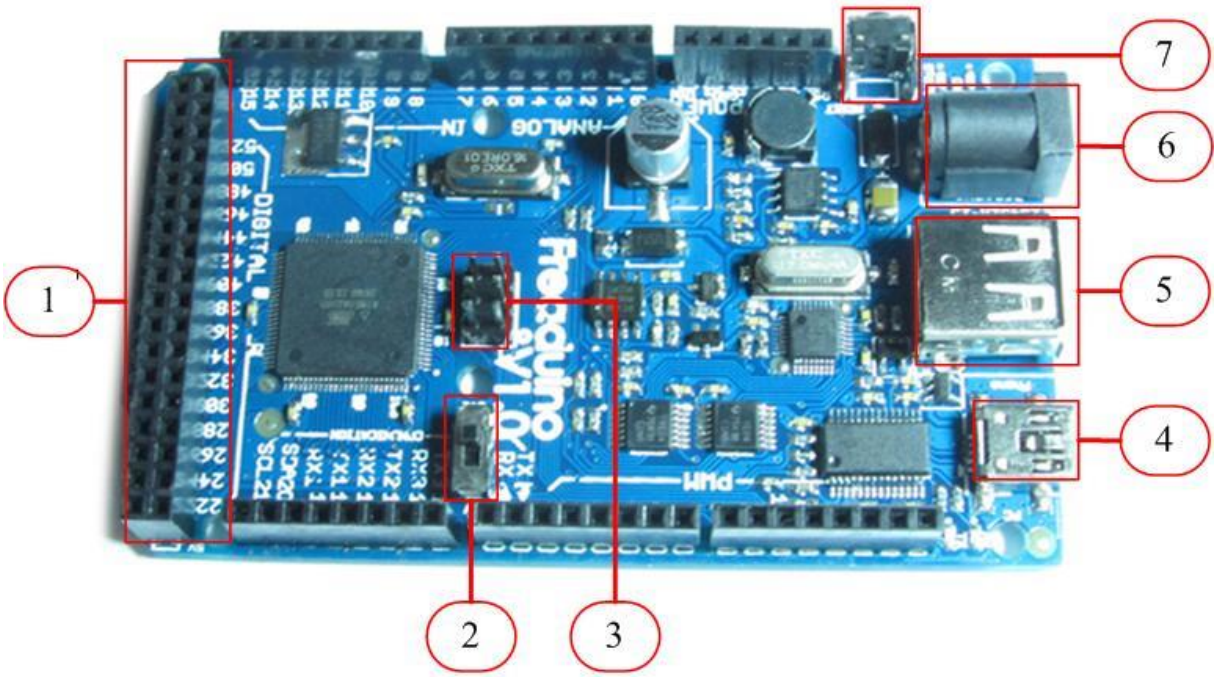
Le projet arduino a était créé par une équipe de développeurs, composée de Massimo Banzi, David Cuartielles, Tom, Igoe, Gianluca Martino, David Mellis et Nicholas Zambetti. Le "système Arduino" a était créé dans le but de permettre aux débutants,

amateurs ou professionnels de créer des systèmes électroniques plus ou moins complexes de manier aisé [20].

Le système Arduino, nous donne la possibilité d'allier les performances de la programmation à celles de l'électronique. Plus précisément, elle permet de programmer des systèmes électroniques. Le gros avantage de l'électronique programmée c'est qu'elle simplifie grandement les schémas électroniques et par conséquent, le coût de la réalisation, mais aussi la charge de travail à la conception d'une carte électronique.

Elle est basée sur un microcontrôleur Atmel ATmega 328 ou ATmega2560. Elle dispose dans sa version de base de 8 Ko de mémoire vive, et 256Ko de mémoire flash pour stocker ses programmes. Elle peut être connectée à 54 entrées ou sorties numériques, dont 15 PWM [21].

VI.3.1 Les composants



Figure(VI.3) : Les composants de la carte Arduino ADK

1) Broches d'E / S
2) Sélection du la source d'alimentation
3) Programmation de la Catre avec ICSP (In-System Programming)
4) USB pour connecter la carte à l'ordinateur.
5) USB A male pour connecter la carte a un appareille Android.
6) Pour une alimentation CC externe.
7) Bouton de réinitialisation.

Les cartes Arduino sont généralement équipées d'une puce ATmega328. Celle qui sont plus évolué sont doté d'une puce ATmega2560 qui a plus de mémoire et d'E/S. La méthode de programmation et pratiquement la même, les différences peuvent apparaitre pour des fonctions plus complexe.

L'interface USB permet une connexion direct avec un câble USB a un ordinateur pour la programmation ou l'échange de donnée avec un programme s'exécutant sur l'ordinateur (ex : MATLAB). Il assure aussi l'alimentation de la carte.

Suivant deux modes différents d'alimentation, les différents composants et sorties sont alimentés par ordinateur via l'USB ou de manier autonome branché a une source de tension (batterie ou pile). Un régulateur de tension est intégré à la carte à condition qu'elle soit alimentée avec une tension de 7 à 12 V.

L'Arduino dispose aussi de plusieurs entrées/sorties (54 pour la carte ADK), dont 15 peuvent assurer une sortie PWM, les entrées analogiques permettent la mesure de tension variable (de 0 à 5V).

On dispose de circuits additionnels :

- Ethernet : communication réseau.
- Bluetooth : communication sans fil.
- pilotage de moteurs (pas à pas ou à courant continu).
- pilotage de matrices de LED : pour piloter de nombreuses LED avec peu de sorties.
- écran LCD : pour afficher des informations.
- lecteur de carte mémoire : lire ou stocker des données.
- lecteur de MP3.
- GPS : pour avoir une information de position géographique.

VI.3.2 Le langage de programmation

La programmation ce fait dans un langage propre à Arduino dont la structure s'apparente aux langages C/C++. Mais lorsque on évoque une fonction Arduino, non standard C/C++, et pourtant reconnue et coloriée comme un mot-clé dans l'éditeur, on fait appel à une ou plusieurs bibliothèques rédigées en C ou C++ qui seront incluses à la compilation.

En enveloppant le langage C/C++ de cette manière, les concepteurs de l'Integrated Development Environment (IDE) ont pu simplifier sa syntaxe et l'adapter aux possibilités de la carte. De nombreuses fonctionnalités de haut niveau sont ainsi proposées à l'utilisateur novice qui n'a plus à se soucier de la logique interne du microcontrôleur.

Il existe plusieurs possibilités d'interfaçage avec d'autres logiciels notamment Matlab Simulink grâce à plusieurs bibliothèques, on cite **Arduino_io** qui nous donne la possibilité d'utiliser la carte en temps réel avec Simulink et donc d'avoir accès aux entrées/sorties, et aussi la bibliothèque **Support Package for Arduino Hardware** qui

inclue un compilateur qui permet la compilation directe du fichier MLD, et le téléversement sur la carte.

VI.3.3 Arduino_io.

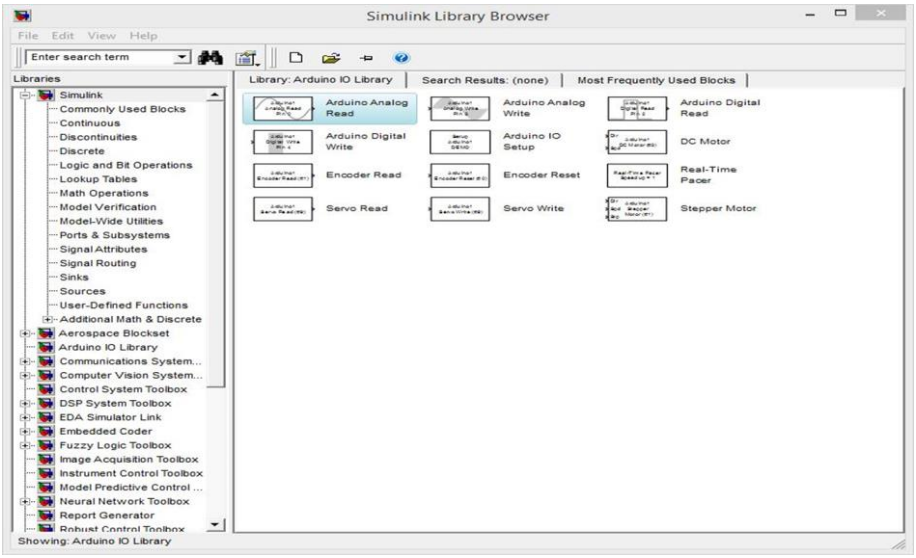
La bibliothèque **arduino_io**, permet d'utiliser la carte arduino comme une interface d'entrées/sorties, le principe est de charger un programme dans la carte pour qu'elle fonctionne comme serveur, écoutant les requêtes envoyées via la liaison série (USB) et de répondre à ces requêtes en renvoyant l'état d'une entrée ou en modifiant l'état d'une sortie. Ces mêmes entrées/sortie sont vues dans MATLAB comme des entrées logiques ou analogiques (utilisation du CAN) ou des sorties analogiques en mode PWM.

a) Installation du package arduino_io

La procédure et relativement simple il suffit de :

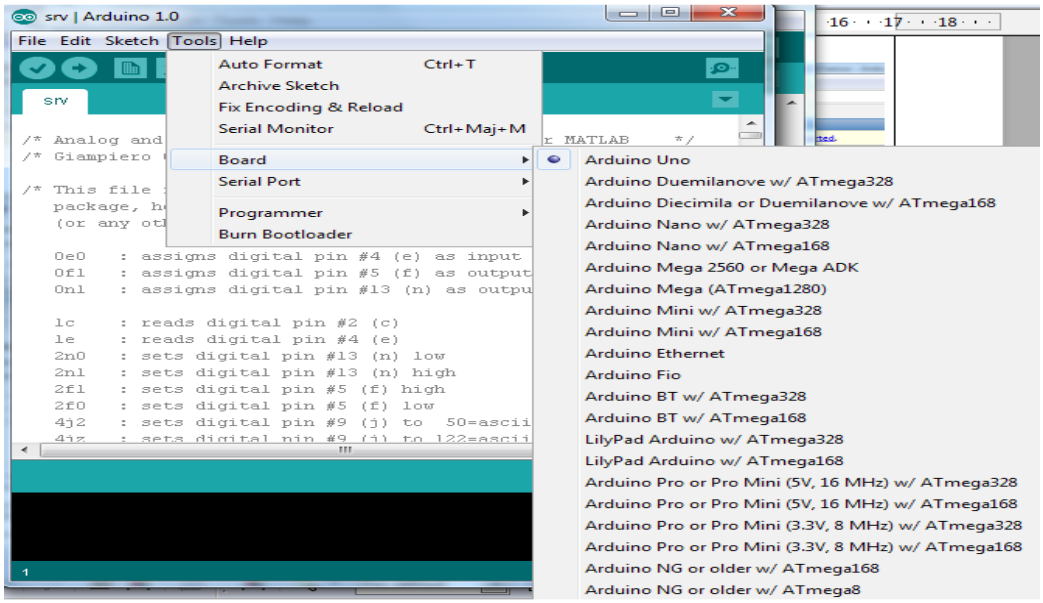
- Décompressé le fichier zip préalablement téléchargé sur le site *mathworks.com.*
- Lancer Matlab2012 en tant qu'administrateur et se placer dans ce même répertoire *arduinoio.*
- Sur le workspace on exécute la commande *install_arduino.*
- Relancer MATLAB et SIMULINK.
- Dans la bibliothèque on retrouve les blocs dans *arduino IO library*.

b) La bibliothèque arduino IO



Figure(VI.4) : bibliothèque Arduino/Simulink

c) Programmation du « Firmware » de l'arduino.



Figure(VI.5) : Choix de la programmation de la carte Arduino

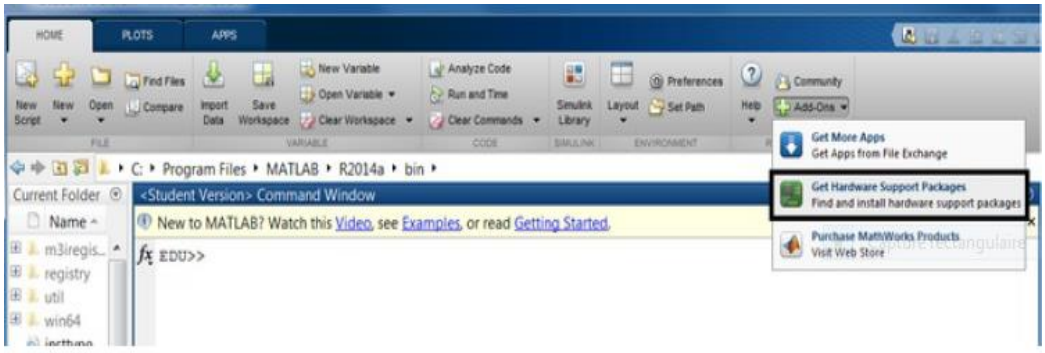
Avant l'utilisation il faut programmer la carte arduino *srv.pde* (qui se trouve dans le sous répertoire « *pde* » l'exécution se fait sur l'IDE arduino.

VI.3.4 Support Package for Arduino Hardware.

Elle offre une possibilité de programmer une carte Arduino, la bibliothèque génère automatiquement le code à partir du modèle Simulink qui va fonctionner alors sur la carte Arduino de façon autonome.

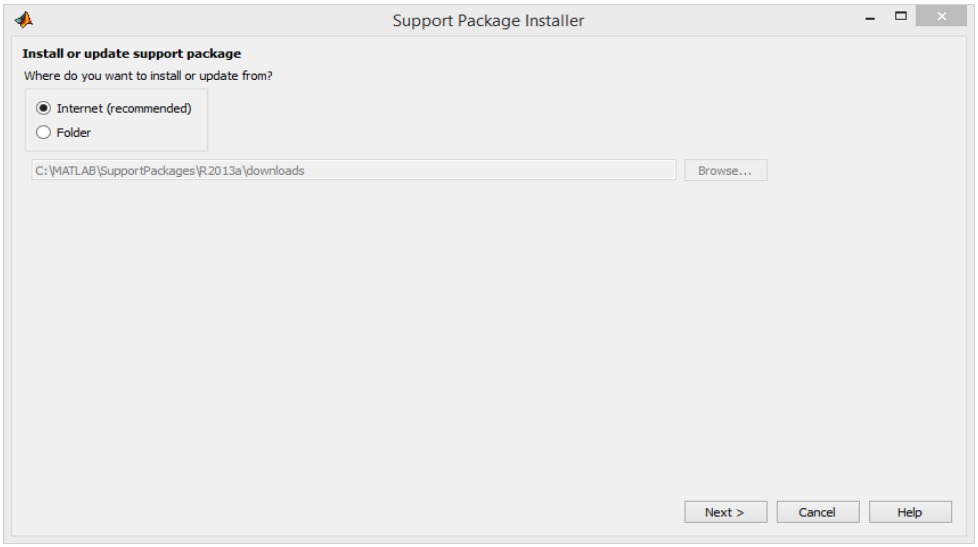
L'installation est relativement simple sur les dernières versions de MATLAB on prend l'exemple de la version Matlab R 2013a.

Après le démarrage de Matlab en tant qu'administrateur sur le droit dans l'angle **Add-Ons** on sélectionne **Get Hardware Support Packages**.



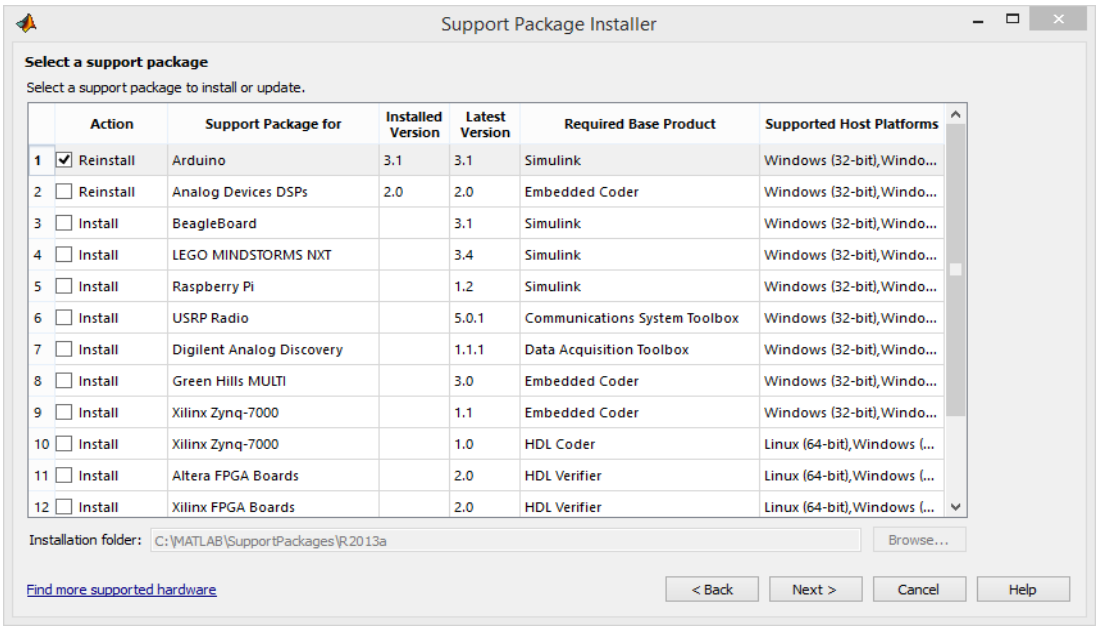
Figure(VI.6) : Support package Arduino

La fenêtre suivante s'ouvre



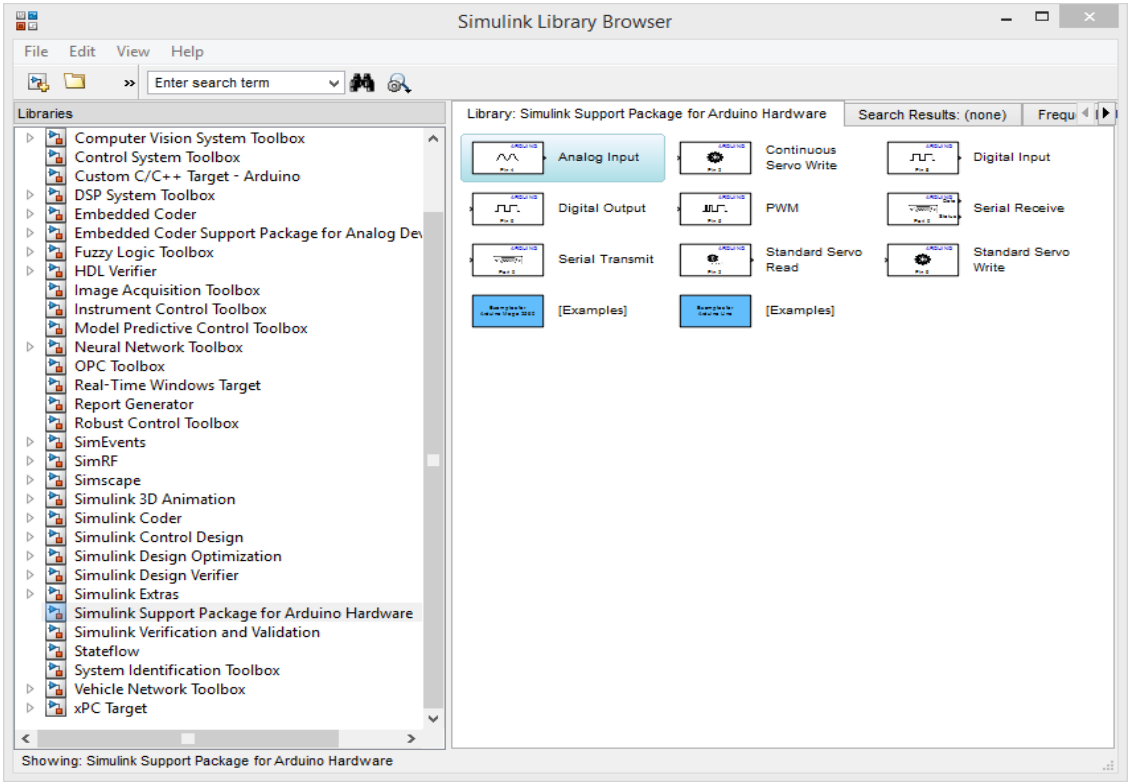
Figure(VI.7) : Fenêtre de téléchargement

Après avoir sélectionné la source on passe à l'étape suivante qui consiste à sélectionné arduino sur la liste, on clic sur « **next »** pour téléchargé et installé la bibliothèque.



Figure(VI.8) : Fenêtre de téléchargement

On retrouve la Tools box dans Simulink



Figure(VI.9) : Fenêtre de la Tools box dans Simulink

VI.3.5 La voie série UART

Le principale objectif de la communication et de transmettre de l'information, elle nécessite l'utilisation d'un langage commun ou d'un code commun : type de liaison, vitesse, format des données, détection d'erreurs.

a) Emetteur et récepteur

Lorsque l'on communique des informations, il faut nécessairement un **émetteur**, qui va transmettre les informations à communiquer, et un **récepteur**, qui va recevoir les informations pour les traiter.

b) Les différents types de communication

On peut citer trois types de conversations entre deux interlocuteurs :

- Simplex : l'un transmet des données l'autre les reçois sans répondre.
- Half-duplex : Chaque interlocuteur transmet des données à tour de rôle.
- full-duplex : chaque interlocuteur transmet en même temps que l'autre.

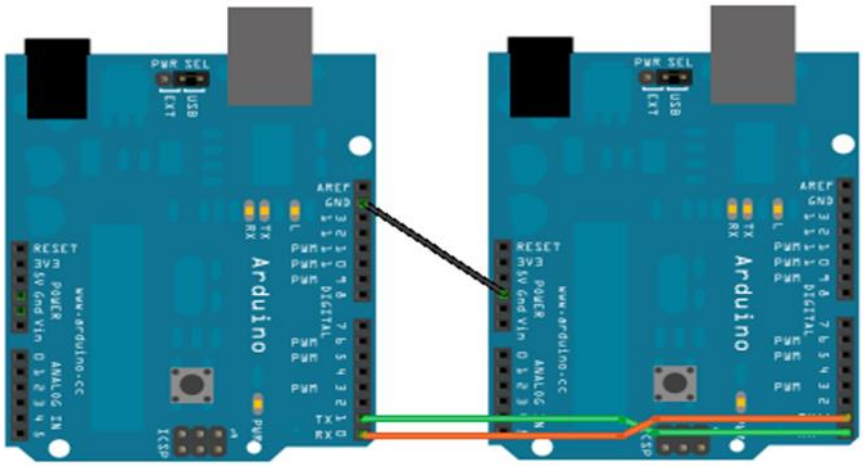
Arduino est capable de faire des communications de type full-duplex. Le protocole de communication est un ensemble de règles qui régissent la façon dont communiquent deux dispositifs entre eux, cela définit le rythme de la conversation, le débit de données, l'ordre des informations envoyées.

c) Application de la norme

Elle définit le signal électrique, et tout ce qui est lié à la connectique, le câblage, etc. ….

On prend l'exemple de la fig. (VI.10) :

- le premier câble est la référence électrique. Cela permet de prendre les mesures de tension en se fixant un même référentiel.
- Les deux autre câbles permettent la transmission des données l'un sert à l'envoi et l'autre à la réception.



Figure(VI.10) : Schéma illustrant la liaison entre deux cartes Arduino

d) Les tensions utilisées

Les bits sont des niveaux de tension imposée par la norme, ces derniers sont cités dans le tableau suivant :

	Niveau logique 0	**Niveau logique 1**
Tension électrique minimale	+3V	-3V
Tension électrique maximale	+15V	-15V

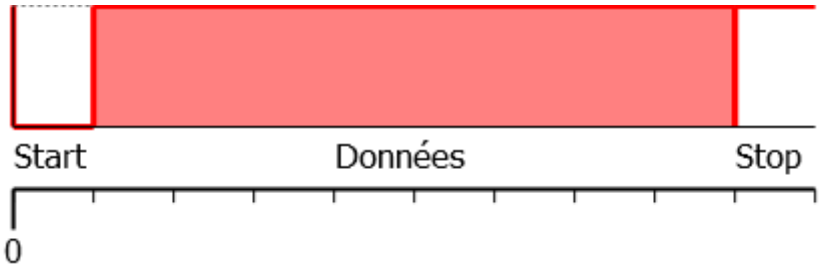
Tableau (VI.11) : Les tensions Utilisé pour la communication série

Toutes les tensions au-delà des valeurs imposé ou au-dessous sont hors normes, elles sont ignorées ça permet d'éviter un certain nombre d'erreurs de transmissions.

e) Les données

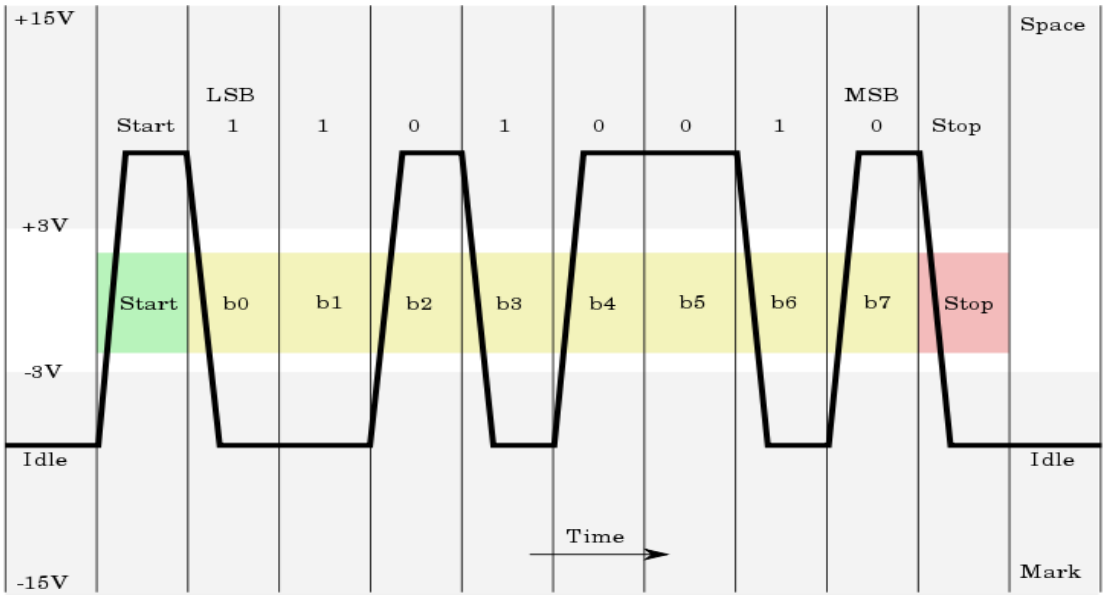
Les données qui transitent par la voie série sont transmises sous une forme binaire codée sur 8 bits selon la **table ASCII**. C'est à dire avec des niveaux logiques 0 et 1. La donnée

commence avec un bit Start et se termine avec un bit stop comme illustré dans la figure suivante :



Figure(VI.12) : Start et stop bit

Les données sont envoyées à l'envers, le bit de donnée qui vient après le bit Start s'appelle le **bit de poids faible** ou **LSB** en anglais pour Less Significant Bit, et le bit de **poids fort** ou **MSB** viens après le stop bit**.**



Figure(VI.13) : transmission d'un octet

f) La vitesse

La norme définit la vitesse à laquelle sont envoyées les données. Elles sont exprimées en bit par seconde (bit/s). Elle préconise des vitesses inférieures à 20 000 bits/s. Sauf qu'on pratique, il est très courant d'utiliser des débits supérieurs pouvant atteindre les 115 200 bits/s. Quand on va utiliser la voie série, on va définir la vitesse à laquelle sont transférées les données. Cette vitesse dépend de plusieurs contraintes qui sont : la longueur du câble utilisé reliant les deux interlocuteurs et la vitesse à laquelle les deux interlocuteurs peuvent se comprendre.

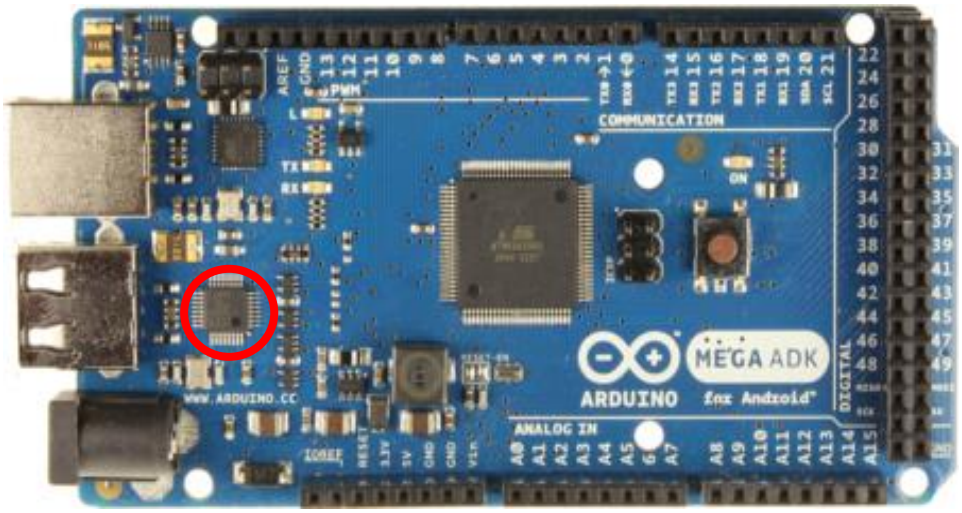
Débit en bit/s	Longueur du câble en mètres (m)
2 400	900
4 800	300
9 600	150
19 200	15

Tableau (VI.14) : Vitesse de transmission

Plus le câble est court, plus le débit est élevé car moins il y a d'affaiblissement des tensions et de risque de parasites. Tandis que si la distance séparant les deux interlocuteurs est grande, la vitesse de communication diminue de façon significative.

g) Émulation du port série

La voie série est émulée à travers l'USB c'est une liaison virtuelle de l'RS232, l'émulation est géré par un circuit intégré (entouré en rouge sur la figure (VI.15)), et le gestionnaire du port USB et périphérique de l'ordinateur.

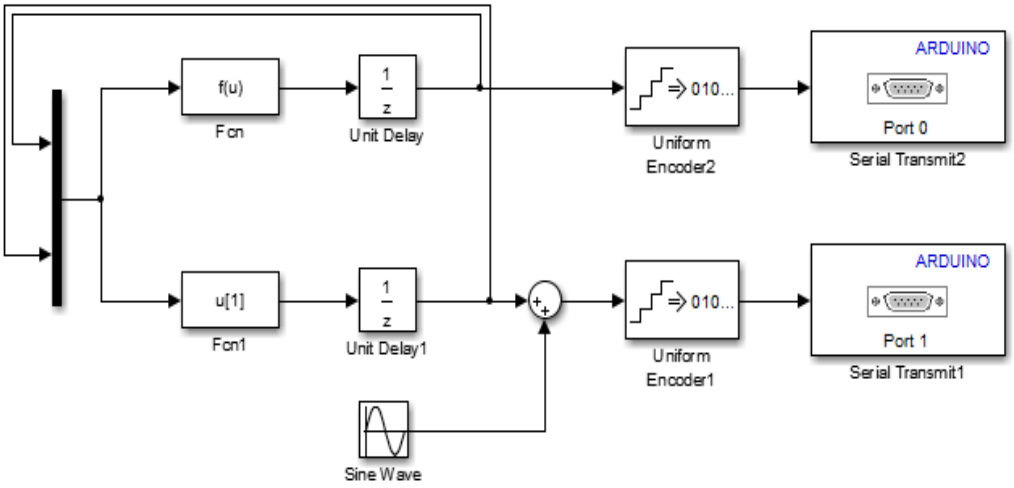


Figure(VI.15) : Circuit émulateur de port série

VI.4 Implémentation sur la carte

VI.4.1 Emetteur :

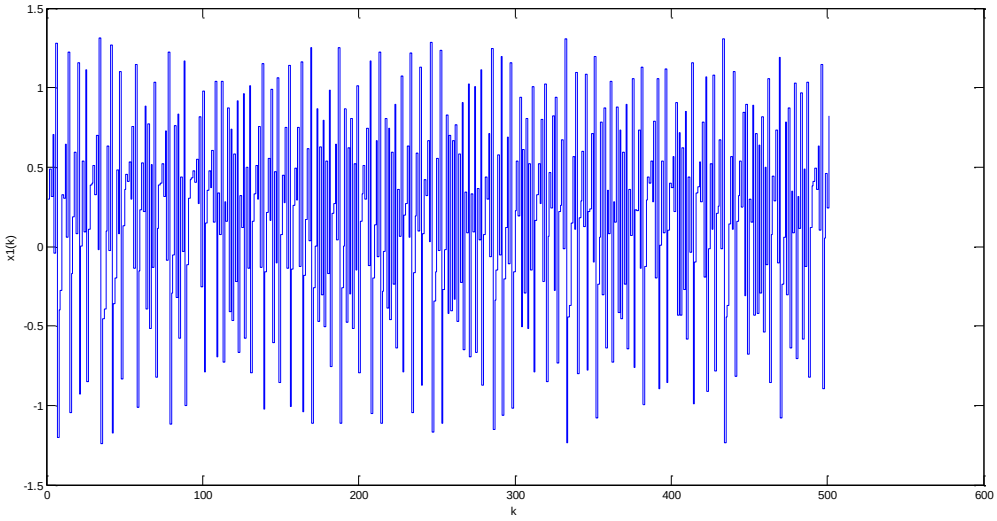
a) Schéma Simulink



Figure(VI.16) : Schéma de l'émetteur

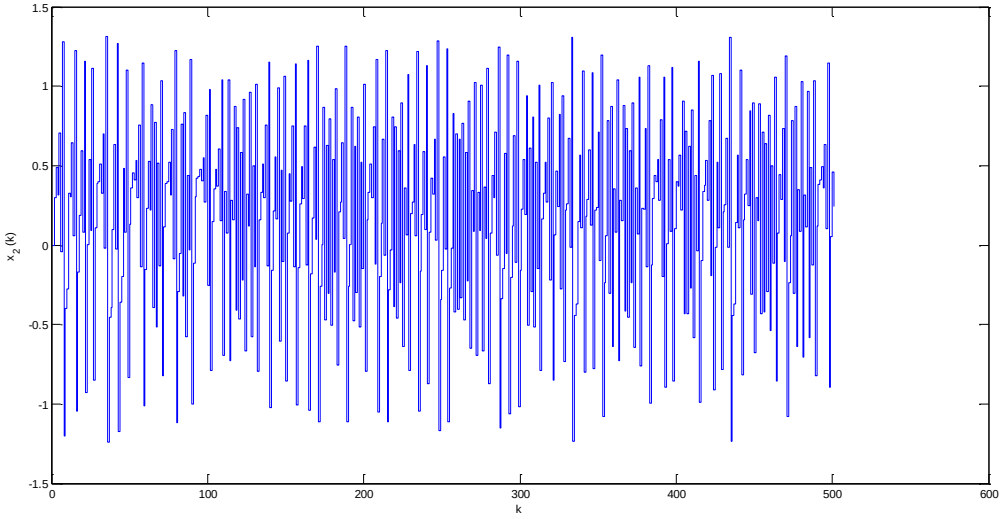
b) Visualisation des signaux

- $x_1(k)$



Figure(VI.17) : Etat $x_1(k)$

- $x_2(k)$

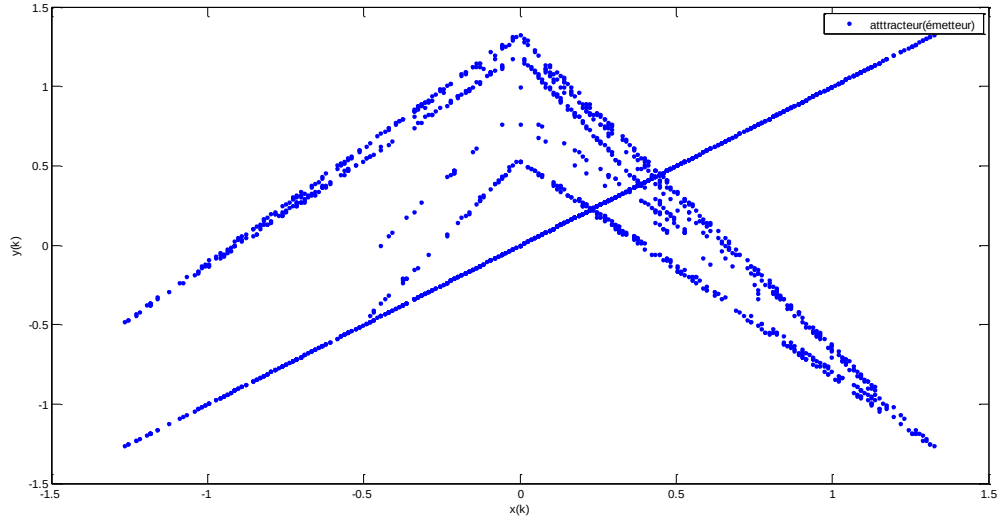


Figure(VI.18) : Etat $x_2(k)$

Observation

D'après les deux dernier figure nous remarquons la présence d'oscillations aléatoire de ce fait les signaux $x_1(k)$ et $x_2(k)$ sont chaotique.

- **Attracteur**



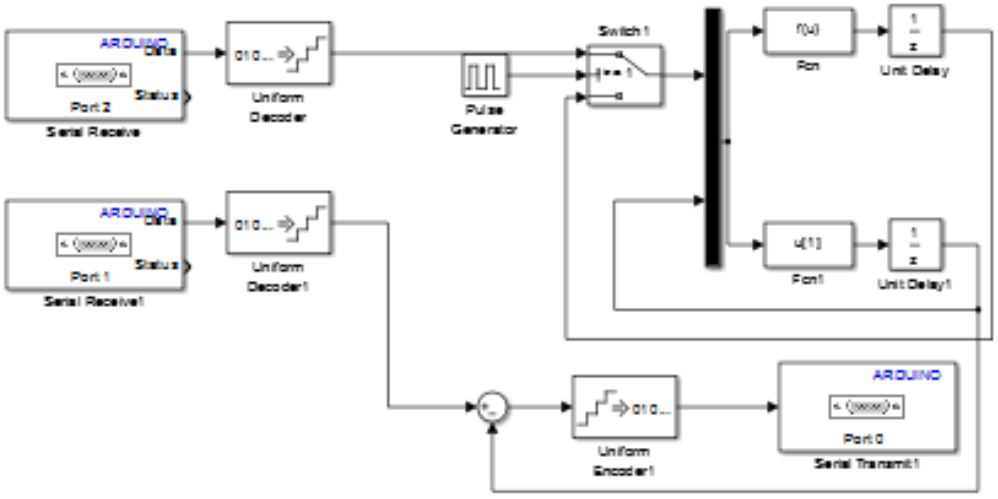
Figure(VI.19) : L'attracteur observée à l'émetteur

Observation

La figure (VI.19) démontre l'aspect chaotique du système, on distingue la forme particulière de l'attracteur de lozi.

VI.4.2 Récepteur

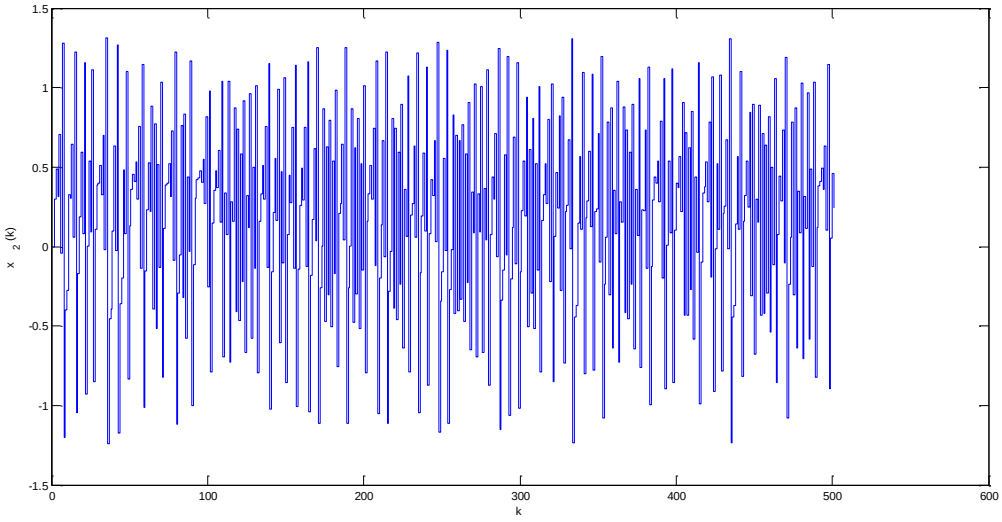
a) Schéma Simulink



Figure(VI.20) : Schéma du récepteur

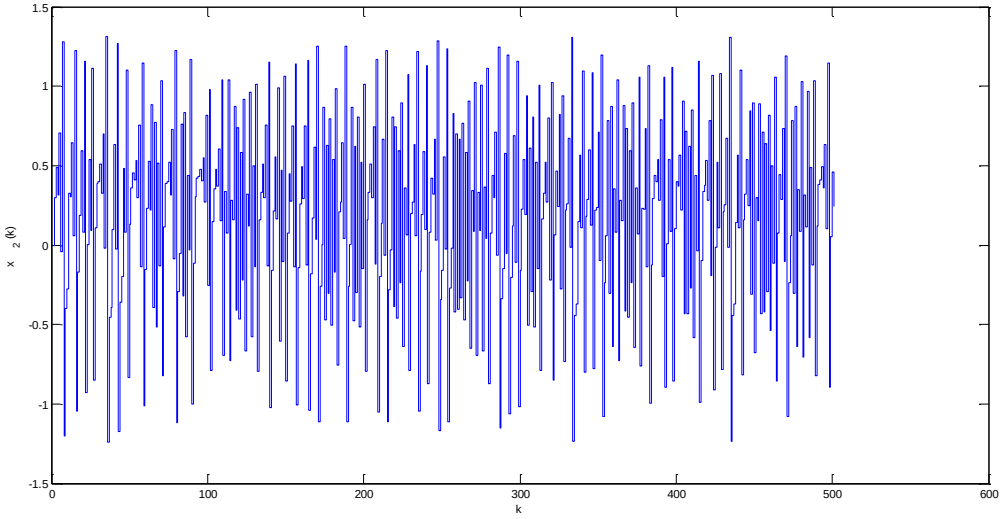
b) Visualisation des signaux

- $\hat{x}_1(k)$



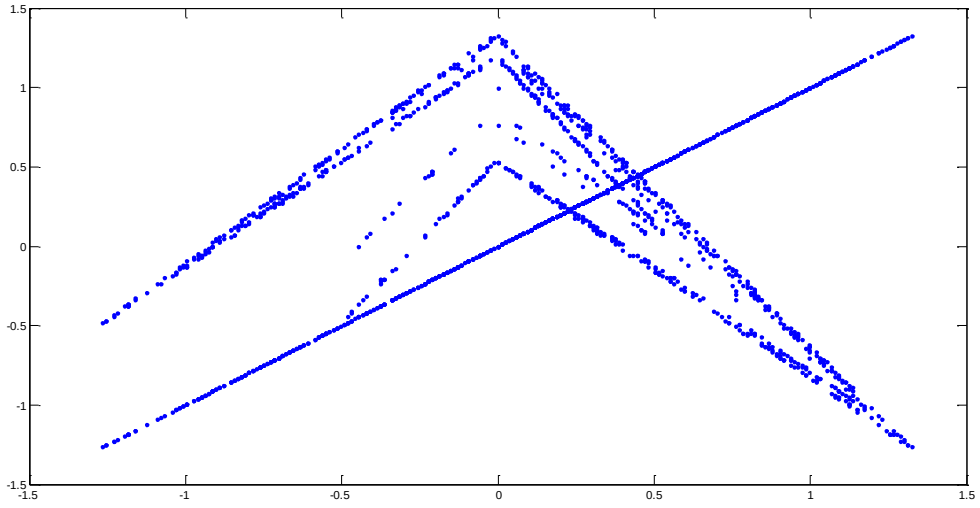
Figure(VI.21) : Etat $\hat{x}_1(k)$

- $\hat{x}_2\ (k)$



Figure(VI.22) : Etat $\hat{x}_2(k)$

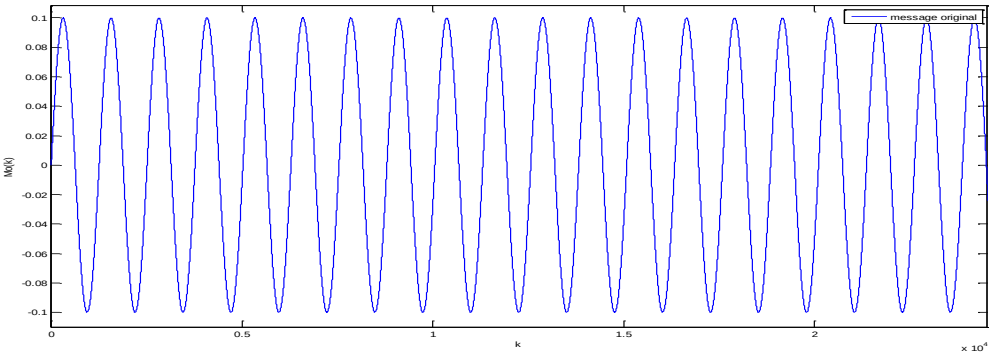
- **Attracteur**



Figure(VI.23) : L'attracteur observée à l'émetteur

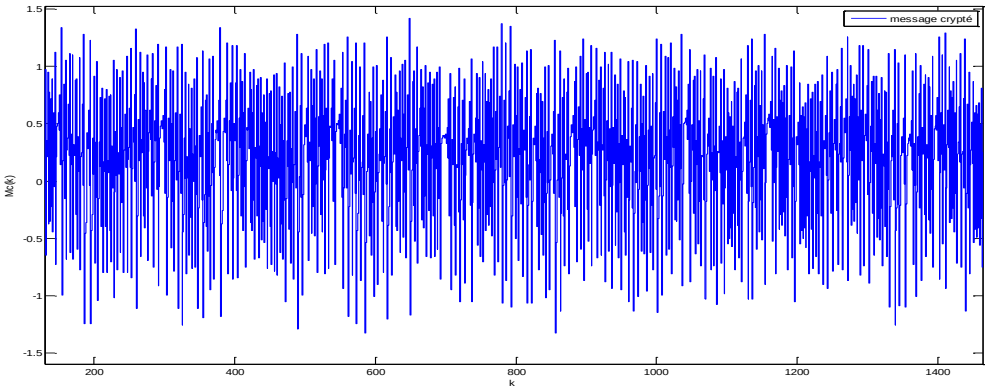
VI.5 visualisation des signaux

a) Message original



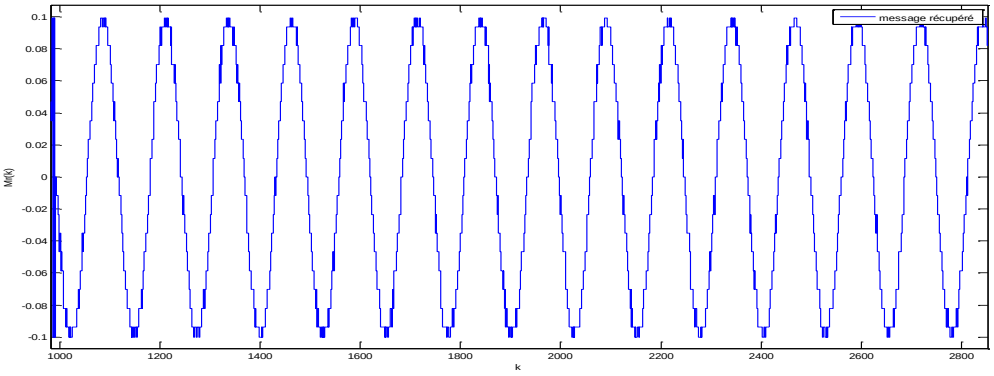
Figure(VI.24) : message envoyé

b) Message crypté



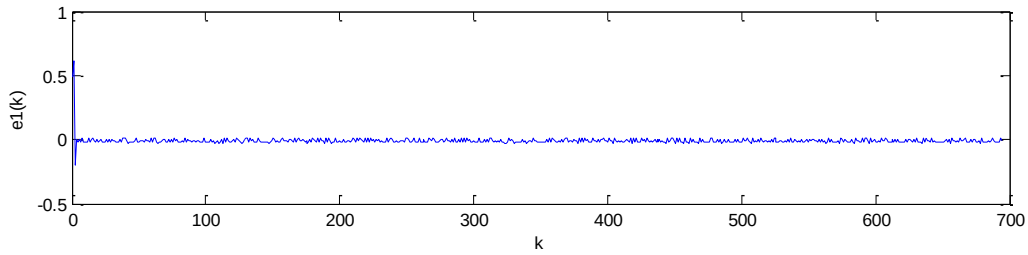
Figure(VI.25) : message crypté

c) Message récupéré

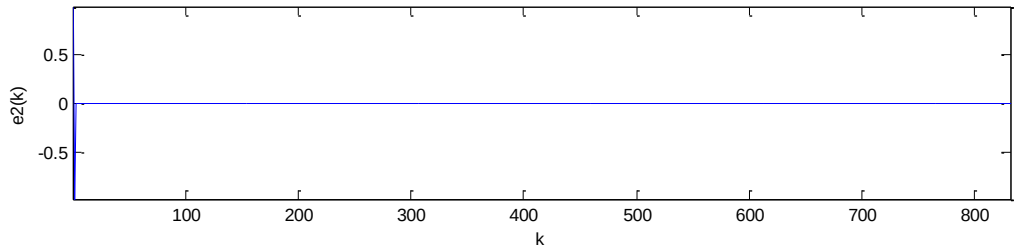


Figure(VI.26) : message récupéré

d) Erreur de synchronisation



Figure(VI.27) : erreur de synchronisation entre $x_1(k)$ et $\hat{x}_1(k)$



Figure(VI.28) : erreur de synchronisation entre $x_2(k)$ et $\hat{x}_2(k)$

Observation

D'après les trois dernières figures on a constaté que le message envoyé a bien était récupéré cela montre que la synchronisation impulsive qu'on étudiée et réalisée a bien fonctionné.

.

VI.6 Conclusion

Il existe plusieurs oscillateurs électroniques générateurs du chaos, Ces derniers peuvent être différenciés l'un des autres suivant leur structure, ou les éléments électriques que les composent.

Dans ce chapitre, on a pu synchroniser deux oscillateurs chaotiques de lozi réalisés pratiquement sur des cartes Arduino. Les résultats obtenus pratiquement ont montré que l'erreur de synchronisation est nulle. Aussi on a pu simuler l'oscillateur de lozi (émetteur-récepteur) sur Simulink, En dernier lieu, on a pu visualiser les différents signaux et attracteurs.

Conclusion Générale

Conclusion générale

Le travail que nous avons réalisé, nous a permis de toucher à un domaine très difficile qui est le chaos, nous avons étudié et réalisé une transmission de données sécurisé par le chaos. Nous avons présenté tous les points essentiels concernant ces systèmes, tel que leur définitions et leur caractéristiques, on a conclu que malgré la complexité de ces systèmes, que son étude et sa réalisation n'est pas impossible.

Dans le chapitre I de ce mémoire, nous avons définit la notion de systèmes chaotiques en abordant leurs propriétés les plus intéressantes comme le déterminisme, la sensibilité aux conditions initiales et l'aspect aléatoire de leurs trajectoire. On a terminé le premier chapitre en montrant la transition vers le régime chaotique, en effet le chaos peut se manifester par intermittence, quasi périodicité ou dédoublement de période dus aux changements de paramètre.

Dans le chapitre II, nous avons cité les méthodes utilisées pour la synchronisation de ces systèmes chaotiques, qui est une étape indispensable pour la transmission de données. Par la suite les notions de cryptage et de décryptage ont été introduites. Nous avons aussi expliqué comment chiffrer un message en le noyant dans un signal chaotique, puis l'envoyer dans un canal public. On a conclu en montrant comment récupérer le message transmis en faisant une simple soustraction.

Dans le chapitre III, après une brève définition d'un oscillateur électrique nous avons étudié l'observabilité du système de lozi avec l'observateur impulsif, ensuite nous avons simulé le système de transmission constitué de oscillateur identique régit pas les équations de lozi sous Matlab Simulink, et nous avons terminés par l'insertion d'un message par addition a un des états du système, ensuite le récupéré au niveau du récepteur.

La réalisation pratique exposée dans le chapitre IV, s'est faite par des cartes arduino, on a réussi à obtenir des résultats satisfaisants que ce soit sous Matlab, ou en pratique.

Nos résultats prouvent que la transmission sécurisé par le chaos étudié par simulation, fonctionnent en pratique, ces résultats ouvrent une possibilité de développement de ces méthodes dans le futur.

Pour finir, on souhaite vivement que notre modeste travail soit apprécié et fera objet d'une contribution aussi minime soit-elle dans le domaine de l'analyse et la synchronisation du chaos.

Annexe

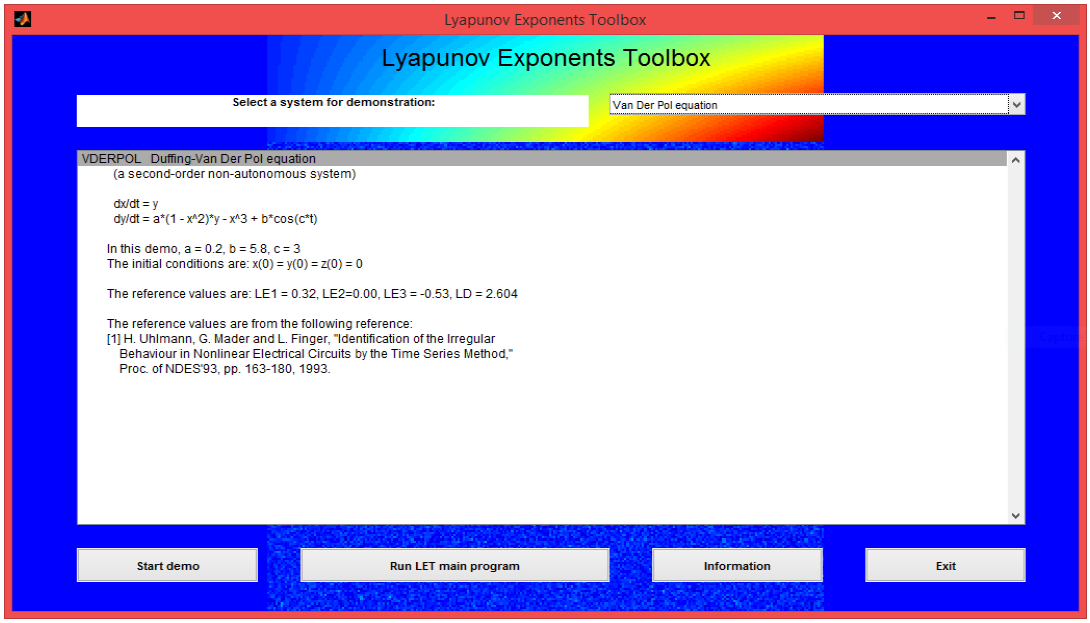
Annexe

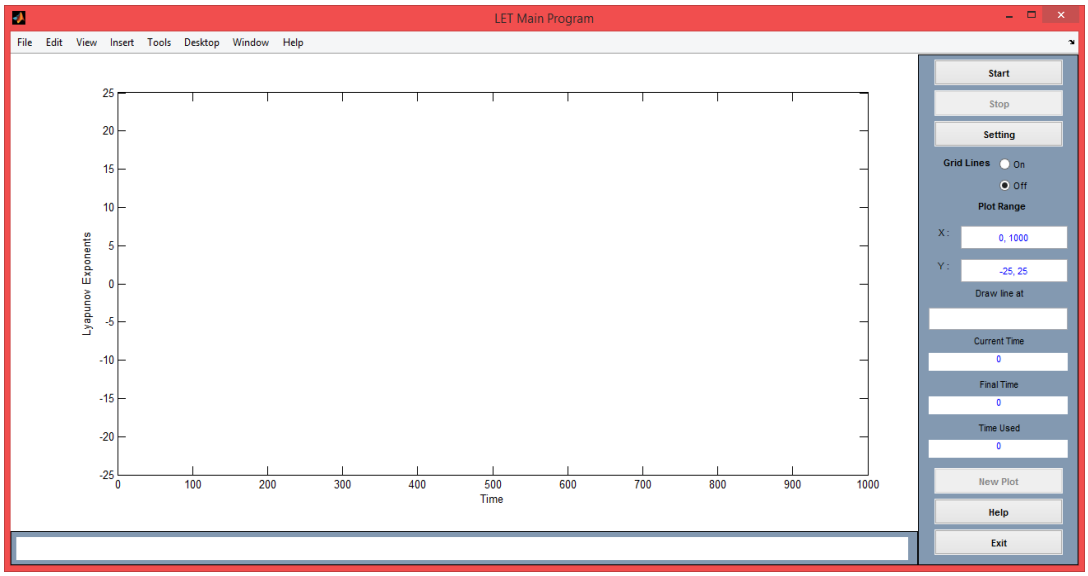
A. LET (Lyapunov Exponents Toolbox)

Lyapunov exposants Toolbox (LET) fournit une interface graphique utilisateur pour les utilisateurs afin de déterminer les ensembles complets d'exposants de Lyapunov et Lyapunov dimension des systèmes chaotiques continus et discrets

Cette boîte à outils ne peut fonctionner que sur MATLAB 5 ou versions supérieures de MATLAB. Il a été testé sous Windows et Unix et peut aussi fonctionner sur d'autres plates-formes.

La fenêtre sur la figure ci-dessous s'ouvre on a le choix entre plusieurs système chaotique, préprogrammer, le système de Lozi ne l'ai pas, la modification des algorithmes est possible.





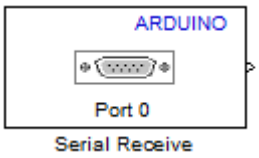
Comment utiliser le programme :

Pour exécuter le programme, entrez « let » dans MATLAB fenêtre workspace. Afin d'exécuter le programme correctement, tous les fichiers de cette boîte à outils doivent être dans le dossier de travail. Quand la fenêtre GUI apparaît, les utilisateurs peuvent exécuter un programme de démonstration en appuyant sur le bouton " Start demo " ou de démarrer le programme principal en appuyant sur le bouton " Run LET main program ". LET fournit certains systèmes chaotiques connus pour des démonstrations. Les utilisateurs peuvent choisir l'un d'eux dans le menu pop-up.

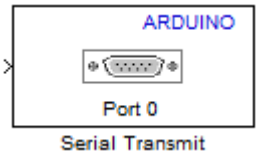
Pour calculer les exposants de Lyapunov et la dimension d'un système, suivez les étapes ci-dessous :

1. Écrire une fonction dans l'ODE qui décrit le système spécifié.
2. Entrez « *let* » dans MATLAB fenêtre de commande,
3. Appuyez sur le "Run " dans la fenêtre de démarrage,
4. Appuyer sur le bouton "Paramètres" dans la fenêtre principale,
5. D'entrer les paramètres souhaités dans la fenêtre de réglage,
6. Lorsque c'est fini, appuyez sur le bouton "OK",
7. Appuyez sur le bouton "Démarrer" dans la fenêtre principale pour démarrer le calcul.

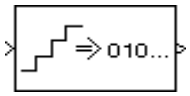
B. Liste descriptive des blocs Matlab Simulink utilisé



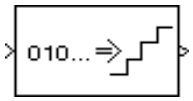
Arduino Serial Receive: reçois un octet de données par période de la mémoire tampon du port série spécifié, le bloc dispose de deux sorties **Data** et **Status ,** la sortie **Status** se met a 1 lorsque une donnée et disponible sur la sortie **Data.**



Arduino Serial Transmit : Envoyer des données mises en mémoire tampon sur le port série spécifié, le bloc accepte des données sous forme de vecteur ou scalaires ***uint8.***

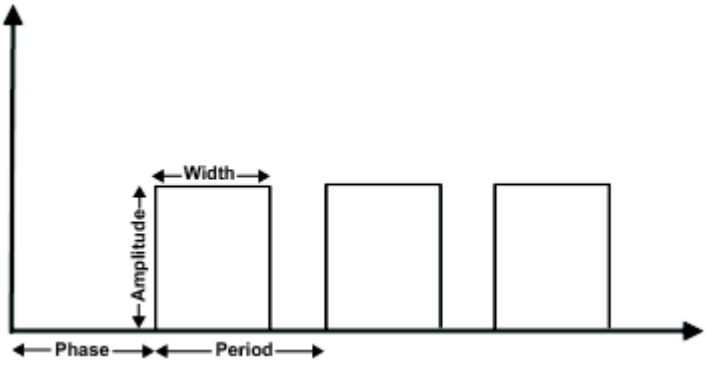


Uniform Encoder : Ce bloc effectue les deux opérations suivantes sur chaque échantillon d'entrée ou matrice, quantifie la valeur avec la même précision et encode la valeur à virgule flottante quantifié à une valeur entière

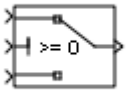


Uniform Decoder : fait l'opération inverse de **Uniform Encoder** et reconstruit, quantifiés les valeurs à virgule flottante à partir de l'entrée entière codé, Les entrées peuvent être des valeurs réelles ou complexes, les types de données traiter (***uint8, uint16, uint32, int8, int16, ou int32***).

Pulse Generator : Le bloc de générateur d'impulsions génère des impulsions carrées à intervalles réguliers. Les paramètres de forme d'onde du bloc, **Amplitude**, **largeur d'impulsion, Période**, et **retard de phase**, permettent de déterminer la forme d'onde de sortie. Le schéma suivant montre comment chaque paramètre affecte la forme d'onde.



Le générateur d'impulsions peut émettre scalaire, vecteur, ou matrice de tout type de données réel.

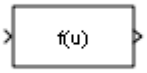


Switch : Le bloc fait passer la première entrée ou la troisième entrée sur la base de la valeur de la seconde entrée. Les première et troisième entrées sont appelées des entrées de données. La seconde entrée est appelée entrée de commande. Les tailles des deux entrées de données

peuvent être différentes ce bloc ne supporte pas les signaux d'entrée de taille variable, L'entrée de commande peut être de tout type.

Mux : Le bloc multiplexeur combine en une sortie unique plusieurs entrés. Une entrée peut être un signal scalaire ou vectoriel, toutes les entrées doivent être du même type de données numérique.

Sum, Add, Subtract, Sum of Elements : Le bloc **Sum** effectue une addition ou une soustraction sur ses entrées. Ce bloc peut sommer ou soustraire des scalaires, vecteurs, ou les éléments de matrice. Il peut aussi inverser les éléments d'un signal.



Fcn : Le bloc Fcn applique l'expression mathématique spécifiée à son entrée. L'expression peut inclure une ou plusieurs de ces composants :

- **u** - L'entrée du bloc. Si u est un vecteur, **u (i)** représente le i ème élément du vecteur ; **u (1)** ou u représente le seul premier élément.
- Les constantes numériques.
- Les opérateurs arithmétiques **(+ - * / ^).**
- Les opérateurs relationnels **(**== => <> = <= **!)** - L'expression renvoie **1** si la relation est vrai ; sinon, elle retourne **0**.
- Les opérateurs logiques **(! && ||)** L'expression renvoie 1 si la relation est vrai ; sinon, elle retourne **0**.
- Parenthèses.
- Les fonctions mathématiques et **tanh** .

Unit Delay : Le bloc **Unit Delay** retarde l'entrée durant une période spécifiez. Ce bloc est équivalent au z^{-1} opérateur à temps discret. Chaque signal peut être un scalaire ou un vecteur. Si l'entrée est un vecteur, le bloc retarde tous les éléments du vecteur durant la même période.

Bibliographie

Bibliographie

[1] H.POINCARE, *"Science et Methode",* G.ABRAHAM-FROIS(1994), 1909.

[2] J. integros, *Le problem des trois corps,* Paris , 1772.

[3] C. d. R. Philippe Etchecopar, *"Quelques éléments sur la théorie du chaos",* 2000.

[4] F. P. Yves Benoist, *"notes de cours systèmes dynamiques élémentaires",* 2003.

[5] J. S. H. GIÉ, *"Le portrait de phase des oscillateurs",* Paris, Mai 1992.

[6] O. Megherbi, these de magister "*Etude et réalisation d'un systéme sécurisé à base de systéme chaotiques",* Tizi ouzou , 2013.

[7] L. Victor, *La suite logistique : un système dynamique chaotique..*

[8] H. Dimassi, these de doctorat *"Synchronisation des systéemes chaotiques par observateurs et applications a la transmission d'informations",* Universit´e Paris Sud XI – Universit´e Tunis El-Manar, 2 Sep 2013.

[9] H. HAMICHE, these de doctorat "*Inversion à Gauche des Systèmes Dynamiques Hybrides Chaotiques. Application à la Transmission Sécurisée de Données",* Tizi Ouzou , 2011.

[10] Z. ELHADJ,these de magister "*ETUDE DE QUELQUES TYPES DE SYSTEMES CHAOTIQUES GENERALISATION D'UN MODELE ISSU DU MODELE DE CHEN",* C O N S T A N T I N E, 2006.

[11] T. L. Louis M.Pecora, *Synchronisation in chaotic Systems,* Washington D.C, 1989.

[12] H. DIMASSI, *Synchronisation des systémes chaotiques par observateurs et applications à la transmission d'informations,* Université Paris Sud XI – Université Tunis El-Manar, 2 Sep 2013.

[13] R. A. Essedik, *observateur à mode glissant d'ordre suppérieur et inversion à gauche,* Université de Tlemcen, 19 mai 2013.

[14] N. S. Boukhalfa, *Synthése d'observateurs non linéaires.*

[15] L. Larger, *Cryptographie par chaos à l'aide des dynamiques non linéaires à retard,* Franche-Comté .

[16] *Introduction à la cryptographie,* USA, 1999.

[17] A. LAYEC, *D´eveloppement de mod`eles de CAO pour la simulation systéme des systémes de communication. Application aux communications chaotiques.,* UNIVERSITE DE LIMOGES, 14 F´evrier 2006.

[18] L. M. S. Meghezzi Rabia, *Réalisation d'un transmission sécurisée de donnée par chaos,* Tizi Ouzou , 2012.

[19] I. M. Y. M. Henk Nijmeijer, An Observer Looks at Synchronization, OCTOBER 1997.

Bibliographie

[20] A. A. A. Elmasry, *Simplify Arduino,* Egypt, 2012.

[21] E. e. olyte, *Arduino pour bien commencer en électronique et en programmation,* openclassrooms, 2012.

Résumé :

Le chaos est caractérisé par un certain nombre de caractéristiques dont la sensibilité aux conditions initiales et l'imprévisibilité, ce qui rend les systèmes chaotiques très intéressants dans le cryptage des données, le travail que nous avons réalisé, nous a permis de toucher à un domaine très difficile qui est le chaos, nous avons étudié et réalisé une transmission de données sécurisé par le chaos. Nous avons présenté tous les points essentiels concernant ces systèmes, tel que leur définitions et leur caractéristiques, on a conclu que malgré la complexité de ces systèmes, que son étude et sa réalisation n'est pas impossible

Mots clés :chaos, chaotique, Lozi, Transmission, Synchronisation, Cryptage.