**R**épublique **A**lgérienne **D**émocratique et **P**opulaire
**M**inistère de L'**E**nseignement **S**upérieur et de la **R**echerche **S**cientifique

**U**NIVERSITE **M**OULOUD **M**AMMERI DE **T**IZI-**O**UZOU

**F**ACULTE DE **G**ENIE **E**LECTRIQUE ET D'INFORMATIQUE
**DEPARTEMENT D'ELECTRONIQUE**

# Mémoire de Fin d'Etudes de MASTER ACADEMIQUE

Spécialité : Electronique
Filière : Télécommunications et Réseaux

**Thème**

# Etude, Simulation et implémentation d'un émetteur hyper chaotique sur carte Arduino Uno.

Présenté par
**Mlle Katia HANNOUN**

Mémoire dirigé par :

- **Mr LAGHROUCHE Mourad, Professeur à l'UMMTO**
- **Mr HAMICHE Hamid, Maître de conférences classe A**

Promotion 2013//2014

# Remerciements

Je remercie vivement le corps enseignant, mes parents et amis tant pour leurs soutien et conseils que pour le temps qu'ils m'ont consacré chaque fois que sollicités.

Je tiens surtout à exprimer ma gratitude à mes enseignants pour tous leurs enseignements et l'aide qu'ils m'ont apportée depuis mon entrée à l'Université Mouloud Mammeri.

En particulier je tiens à remercier Messieurs

- Laghrouche Mourad, Professeur à l'UMMTO
- Hamiche Hamid, Maitre de conférences à l'UMMTO

pour leurs disponibilité et conseils avisés ainsi que pour avoir accepté d'encadrer mon travail et me faire bénéficier de leurs compétences.

Mes remerciements s'adressent également aux membres du Jury pour le temps et l'attention qu'ils accordent au présent mémoire.

# SOMMAIRE

Pages

## 1. Introduction générale :

Depuis les découvertes de Newton, les systèmes dynamiques étaient régis par des lois simples et déterministes. Mais les travaux d'un mathématicien français Henry Poincaré ébranlèrent cette théorie qui faisait de l'univers un système géant muni de lois simples qu'il suffisait de trouver. Beaucoup plus tard la réflexion scientifique engendrée par ces travaux donnera naissance à la théorie du chaos. La théorie du chaos s'applique aux systèmes dynamiques.

Parmi ces systèmes, on retrouve bien sûr une bonne partie des systèmes dynamiques courants (calcul de trajectoires : d'un élément mobile, d'une planète, etc.) ; mais on y retrouve aussi l'évolution des populations, des automates cellulaires, la météorologie et bien d'autres. En terme général, on dit d'un système qu'il est chaotique s'il est régi par des lois déterministes connues mais que son évolution échappe tout de même à toute prévision sur le long terme. L'origine de ce phénomène est la dépendance de ces systèmes aux conditions initiales. C'est d'ailleurs grâce aux recherches sur les prévisions météorologiques, que la théorie du chaos émergea aux yeux des scientifiques dans les années soixante.

L'utilisation du chaos dans la transmission sécurisée de l'information a été considérée comme étant une solution très prometteuse pour augmenter les performances des systèmes de transmission actuels. Ainsi, on trouve dans la littérature une multitude d'applications et d'études réalisées concernant plusieurs aspects de la transmission.

L'emploi du chaos dans les systèmes de communication peut permettre de renforcer la sécurité de transmission de l'information et réduire la probabilité d'interception. Dans les systèmes de communication, la synchronisation est fondamentale pour une transmission réussie entre émetteur et récepteur. La synchronisation chaotique au niveau du récepteur cherche à dupliquer le signal chaotique envoyé par l'émetteur. Cette synchronisation entre deux systèmes dynamiques chaotiques nécessaire à la récupération de l'information transmise n'est pas des plus simples à réaliser.

Ce travail porte sur l'étude des systèmes chaotiques et leur emploi dans les systèmes de communication. Nous verrons d'abord plus en détail les caractéristiques du phénomène chaotique, puis nous étudierons plusieurs exemples pour clarifier certains points de cette

théorie comme la dépendance aux conditions initiales ou l'apparition des attracteurs étranges à l'aide de programmes qui permettront de bien assimiler les différentes facettes de cette théorie. Ainsi, notre travail est organisé de la manière suivante :

Le premier chapitre introduit la théorie du chaos, en présentant les généralités sur les systèmes chaotiques par quelques définitions, puis en citant les deux classes de systèmes chaotiques, ainsi que leurs différentes caractéristiques, et en donnant aussi les simulations sous logiciel Matlab de chaque système étudié.

Le second chapitre fait le lien entre les systèmes chaotiques et le domaine des télécommunications. Celui-ci est centré sur le phénomène de synchronisation des systèmes chaotiques, ainsi que les techniques de transmission sécurisée d'informations qui reposent sur le principe de synchronisation chaotique, car l'emploi d'un signal chaotique dans le domaine des télécommunications pose directement le problème de synchronisation du récepteur dans le but de dupliquer le signal chaotique employé à l'émetteur. Ensuite, les différentes méthodes de synchronisation sont exposées. Un des objectifs du chaos étant de protéger l'information transmise, nous citons les différentes techniques de cryptages par le chaos.

Le troisième chapitre porte sur un schéma de transmission de données à base de la synchronisation de deux systèmes chaotiques. En décrivant la chaine de transmission, puis le principe de la méthode suivie (choix de l'émetteur, du récepteur et mise au point du processus de transmission de l'information.), ainsi que les résultats de simulation du système de transmission proposé.

Le dernier chapitre est destiné à la réalisation d'un émetteur chaotique sur une carte Arduino Uno, avec présentation de la carte Arduino Uno et de son Microcontrôleur ATMEL ATMega328. Les étapes d'installations et de mise en fonctionnement de la carte seront données, avec des exemples de programmations, pour enfin programmer la carte en tant qu'émetteur hyper chaotique.

## 1.1Introduction :

La théorie du chaos fait partie des sciences les plus récentes et est devenue l'un des domaines les plus avancés dans la recherche contemporaine. Les origines de cette nouvelle théorie s'étendent aux mathématiques et physiques des débuts du 20ème siècle, mais elle a émergé dans les années 1960-70.

Durant des années, le chaos était considéré comme incontrôlable et même inutilisable, malgré la mise en équation de certains phénomènes et la démonstration du déterminisme dans des aspects d'apparence aléatoire.

La théorie du chaos est définie comme une étude des systèmes dynamiques non-linéaires complexes ou les systèmes complexes qui sont exprimés par des récurrences et des algorithmes mathématiques et qui sont dynamiques non constants et non périodiques. Elle inclut l'étude qualitative et quantitative d'un comportement instable non périodique et aléatoire des systèmes dynamiques non linéaires déterministes. Le chaos peut être vu aussi comme un système avec des propriétés stochastiques. Dans toutes les définitions qui peuvent exister pour le chaos, un phénomène fondamental est indispensable : la sensibilité aux conditions initiales.

En effet, en programmant son ordinateur et en changeant par $10^{-4}$ les conditions initiales des prévisions météo, Edward Lorenz a découvert que pour certaine équation ou système d'équations non linéaires les résultats montrent une grande sensibilité aux conditions initiales. On peut dire que cette anecdote est la base du chaos déterministe.

La théorie du chaos influence l'explication de plusieurs phénomènes et trouve son application dans plusieurs domaines tels que :

- Economie : Prévision des cycles économiques, des mouvements commerciaux et des marchés financiers.
- Météo : Prévisions météorologiques.
- Santé : Prévision des crises d'épilepsie.
- Sciences sociales : Comportement des systèmes sociaux.
- Cryptage de l'information.

## 1.2 Définitions :

### 1.2.1 Système dynamique : [1]

Un système dynamique est un système physique qui évolue. Il peut évoluer dans le temps ou par rapport à une autre variable suivant l'espace de phase considéré.
La trajectoire d'un objet en mouvement dans le temps est donc un système dynamique, ainsi que le nombre d'individu d'une population quelconque dans le temps, ou encore les valeurs d'une fonction par rapport à la valeur de x.

Un système dynamique (discret ou continu) décrit par une fonction mathématique présente deux types de variables : dynamiques et statiques. Les variables dynamiques sont les quantités fondamentales qui changent avec le temps. Les variables statiques, encore appelées paramètres du système, sont fixes.

### 1.2.2 Système dynamique linéaire :

Un système physique est dit linéaire si la relation entre les grandeurs d'entrée et de sortie peut être définie par des équations différentielles linéaires (à coefficients constants). Ces dernières vérifient alors les principes de proportionnalité des effets aux causes, et de superposition.

### 1.2.3 Système dynamique non linéaire : [2]

Un système non linéaire est un système qui n'est pas linéaire, c'est-à-dire (au sens physique) qui ne peut pas être décrit par des équations différentielles linéaires à coefficients constants. Cette définition, ou plutôt cette non-définition explique la complexité et la diversité des systèmes non linéaires et des méthodes qui s'y appliquent. Il n'y a pas une théorie générale pour ces systèmes, mais plusieurs méthodes adaptées à certaines classes de systèmes non linéaires.

### 1.2.4 Système dynamique chaotique : [3]

La théorie du chaos traite des systèmes dynamiques déterministes qui présentent un phénomène fondamental d'instabilité appelé «sensibilité aux conditions initiales », ce qui les rend non prédictibles en pratique sur le «long » terme. Le chaos est défini généralement comme un comportement semblant aléatoire (ou imprévisible) d'un système dynamique défini par des équations déterministes.

Un système dynamique est défini à partir d'un ensemble de variables qui forment le vecteur d'état $X = \{x_i \in R\}$, i = 1…$n$ où $n$ représente la dimension du vecteur. Nous appelons état d'un système l'ensemble des variables qui, étant connues à l'instant initial, permettent de décrire l'évolution de ce système. L'ensemble de tous les états pouvant être pris par le système s'appelle l'espace des phases. Le processus évolue de manière déterministe si ses états futurs sont caractérisés par la connaissance de ses états présents et passés. La loi d'évolution dans le temps de ce système dynamique est généralement désignée par "dynamique ". En somme, la notion de déterminisme provient du fait que le système est caractérisé par son état initial et sa dynamique.

## 1.3 Classes de systèmes chaotiques :

Il existe plusieurs systèmes chaotiques qui sont utilisés pour générer les signaux chaotiques. Dans ce paragraphe, nous présenterons deux classes : les systèmes chaotiques à temps continus et les systèmes chaotiques à temps discret.

### 1.3.1 Systèmes chaotiques continus :

Un système chaotique à temps continu est décrit par un système d'équations différentielles de forme :

$$\dot{x} = f(t, x, u);\ y = h(t, x, u) \tag{1}$$

Où : $x \in U \subseteq \mathbb{R}^n$ est un vecteur de dimension n, $f : \mathbb{R}^n \to \mathbb{R}^n$ est une fonction non linéaire désignant le champ de vecteurs, h : $\mathbb{R}^n \to \mathbb{R}$ une fonction éventuellement non linéaire qui désigne le vecteur de sortie et $u \in V \subseteq \mathbb{R}^p$ représente l'entrée du système. Si ce système ne dépend pas de l'entrée, on aura alors :

$$\dot{x} = f(t, x) \tag{2}$$

Il existe plusieurs systèmes chaotiques continus, parmi eux l'on peut citer les systèmes de Lorenz, Rössler, Bogdanov, le circuit de Chua, etc.

- **Système de Lorenz :**

Le système de Lorenz est généré par le système d'équations suivant :

$$\begin{cases} \dot{x} = a(y - x) \\ \dot{y} = x(b - z) - y \\ \dot{z} = xy - cz \end{cases} \tag{3}$$

Cet exemple a été publié en 1963 dans un journal météorologique.
Les variables *x,y* et *z* représentent l'état du système à chaque instant. Et *a,b,c* les paramètres du système. Le système présente un comportement chaotique pour *a=10, b=28, c=8/3* et présente un attracteur étrange en forme d'ailes de papillon.

- **Système de Rössler :**

Le système de Rössler est donné par les équations suivantes :

$$\begin{cases} \dot{x} = -(y+z) \\ \dot{y} = x + ay \\ \dot{z} = b + z(x-c) \end{cases} \quad (4)$$

x,y et z sont les variables d'état du système. *a, b* et *c* sont les paramètres réels. Les paramètres et les conditions initiales de cette équation ont été choisis de la manière suivante :
*a = b = 0.1 et c = 14* $(x_0,y_0,z_0) = (0.01,0.01,0.01)$

Ces équations différentielles définissent un système dynamique continu et tridimensionnel qui présente des caractéristiques chaotiques. L'ensemble des trajectoires à long terme de ce système définissent un attracteur étrange aux propriétés fractales.

### 1.3.2 Systèmes chaotiques discrets :

Un système chaotique à temps discret est décrit par un système d'équations aux différences finies, dont le modèle général est le suivant :

$$x(k+1) = G\big(x(k),u(k)\big)y(k) = h\big(x(k),u(k)\big) \quad (5)$$

Où G : $\mathbb{R}^n \to \mathbb{R}^n \times \mathbb{Z}^+ \to \mathbb{R}^n$ désigne la dynamique du système en temps discret.

Parmi les systèmes chaotiques discrets, nous pouvons citer les systèmes de Hénon, Hénon modifié, Lozi, la fonction logistique, etc.

- **Système de Hénon : [9]**

Introduit par l'astronome Michel Hénon en 1976. Il est présenté par l'équation suivante :

$$\begin{cases} \boldsymbol{x_{n+1} = y_n + 1 - ax_n^2} \\ \boldsymbol{y_{n+1} = bx_n} \end{cases} \quad (6)$$

Tel que $(x_n,y_n) \in R^2$

Il dépend de deux paramètres, *a* et *b*, qui ont pour valeurs *canoniques : a = 1.4 et b = 0.3.*

Les conditions initiales prises sont $x_0=0.1$, $y_0=0$

Pour d'autres valeurs de *a* et *b*, il peut être chaotique, intermittent ou converger vers une orbite périodique.

- **Système Hénon-Heiles ou Hénon modifié :**

Il est donné par les équations suivantes :

$$\begin{cases} x\,(n+1) = a - y^2(n) - bz\,(n) \\ y(n+1) = x(n) \\ z(n+1) = y(n) \end{cases} \quad (7)$$

Pour avoir un comportement chaotique, les paramètres du système sont donnés comme suit : $a = 1.76$ *et* $b = 0.1$ et les conditions initiales du système : $x_0 = 0.1$ $y_0 = 0.1$ $z_0 = 0.1$.

## 1.4 Propriétés des systèmes chaotiques :

Bien qu'il n'y ait pas de définition mathématique du chaos universellement acceptée, une définition couramment utilisée stipule que, pour qu'un système dynamique soit classifié en tant que chaotique, il doit comporter les propriétés suivantes :

- Aspect aléatoire
- Sensibilité aux conditions initiales
- Notion d'attracteur
- Exposants de Lyapunov
- Fonction d'autocorrélation et spectre de puissance
- Bifurcation

### 1.4.1 Aspect aléatoire :

Les systèmes chaotiques se comportent, en effet, d'une manière qui peut sembler aléatoire. Cet aspect aléatoire du chaos vient du fait que l'on est incapable de donner une description mathématique du mouvement. Mais ce comportement est, en fait, décrit par des équations non-linéaires parfaitement déterministes, comme par exemple les équations de Newton régissant l'évolution d'au moins trois corps en interaction. Les figures ci-dessous illustrent les aspects aléatoires des différents systèmes chaotiques continus (3) et (4) et discrets (6) et (7).

- ***Aspect aléatoire du système de Lorenz :***

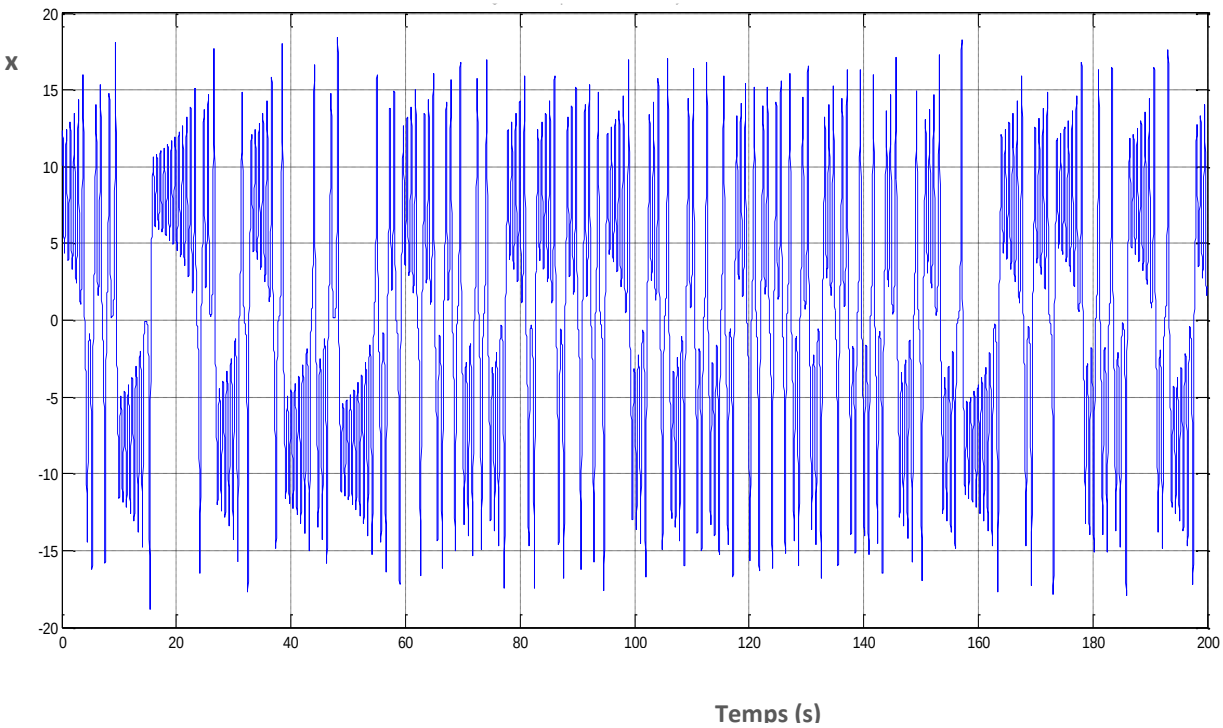



**Figure 1** : *Aspect aléatoire du système de Lorenz.*

- ***Aspect aléatoire du système de Rössler :***

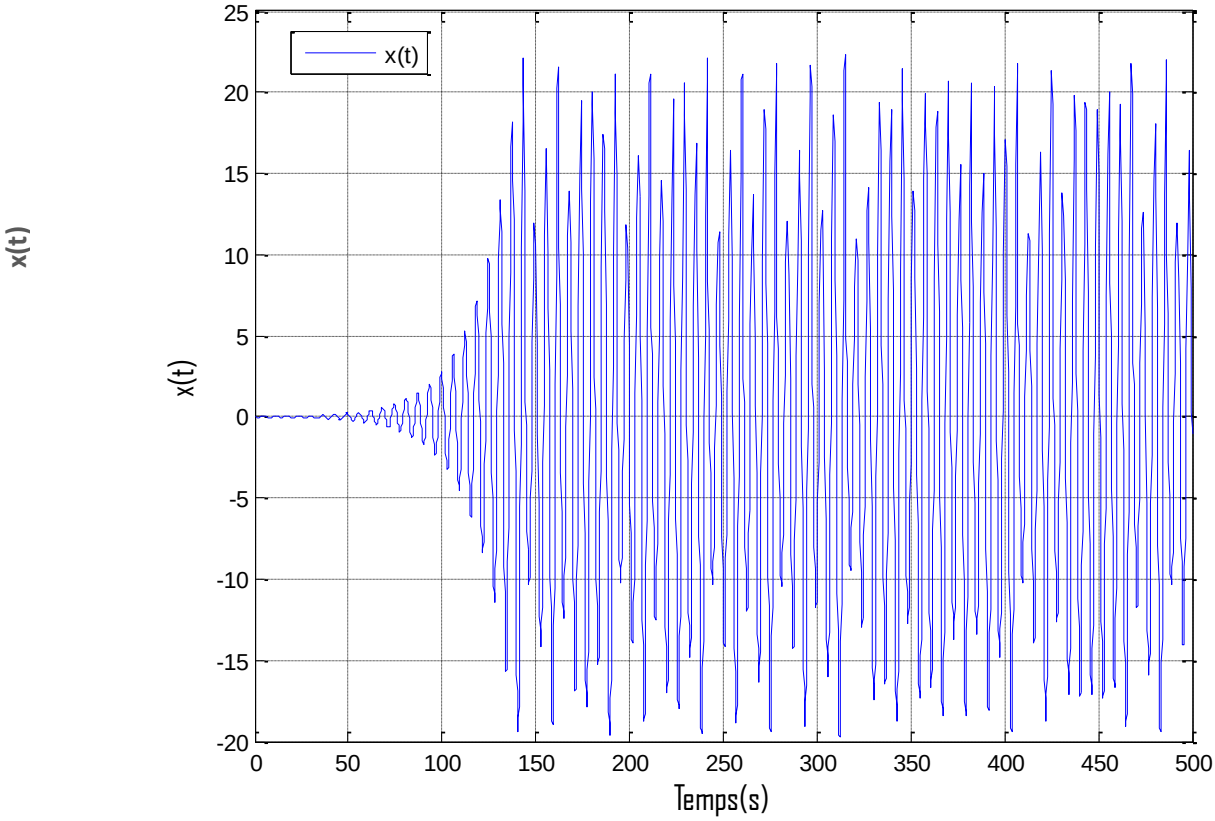



**Figure 2 :** *Aspect aléatoire du système de Rössler*

- ***Aspect aléatoire du système de Hénon :***

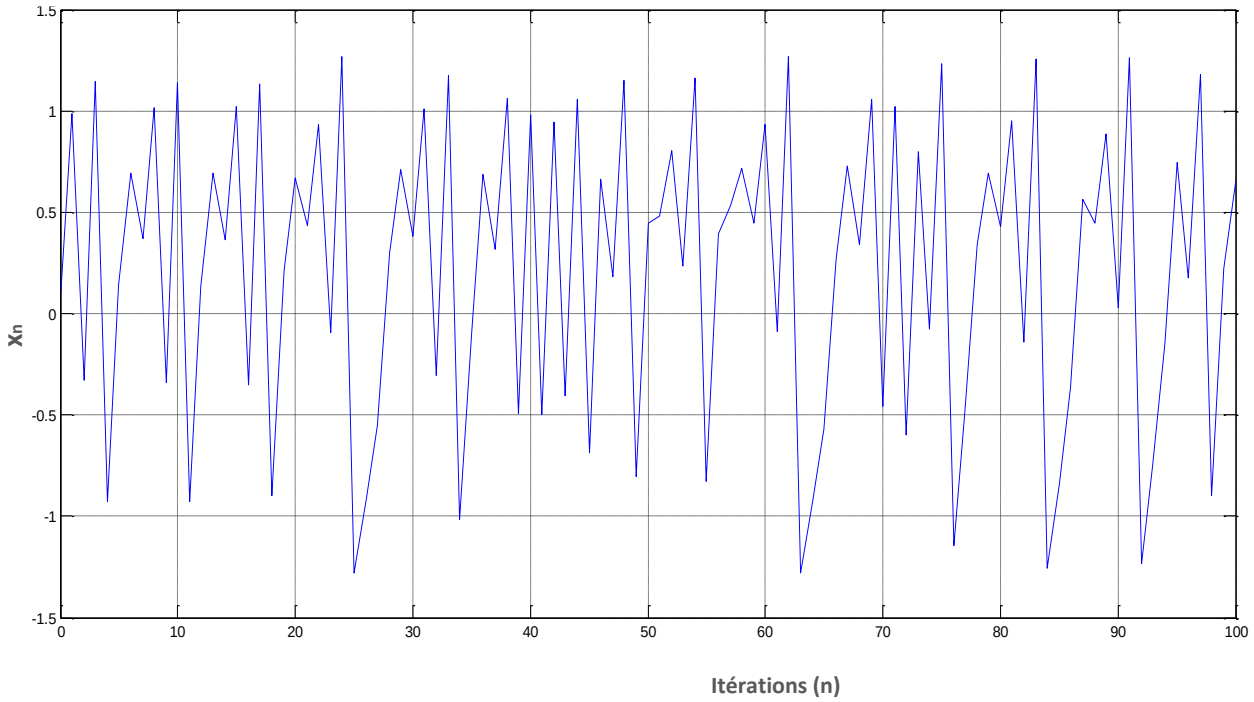



**Figure 3** : *Aspect aléatoire du système de Hénon.*

- ***Aspect aléatoire du système de Hénon-Heiles :***

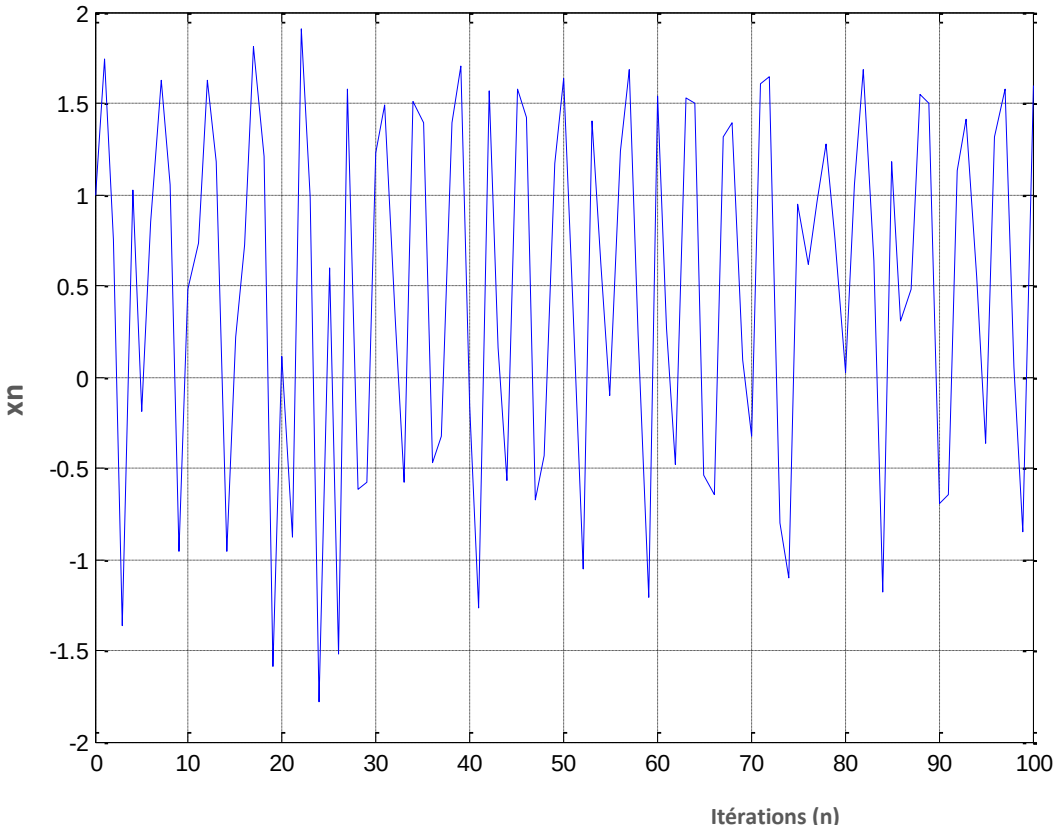



**Figure 4** : *Aspect aléatoire du système de Hénon-Heiles*

### 1.4.2 Sensibilité aux conditions initiales : [9]

En faisant la troncature de quelques chiffres de conditions initiales de son système de prévision météorologique, Lorenz a mis en relief le caractère le plus important des systèmes chaotiques qui est la sensibilité aux conditions initiales. Mais en fait c'est avant cette anecdote, que ce phénomène a été découvert. Vers la fin du 19ème siècle, Poincaré montrait que les trois orbites de 3 corps en mouvement sous une force centrale due à la gravité changent radicalement avec une petite modification des conditions initiales. Pour un système chaotique, une très petite erreur sur la connaissance de l'état initial $x_0$ dans l'espace des phases va se trouver (presque toujours) rapidement amplifiée.

Soit ($\chi$, d) un espace métrique compact et f : $\chi \rightarrow \chi$, une fonction.
Il existe un nombre réel $\beta > 0$, tel que pour tout $x_0 \epsilon \chi$ et pour tout $\varepsilon > 0$ , il existe un point $y_0 \epsilon \chi$ et un entier $k > 0$ vérifiant :

$$\boldsymbol{\delta(x_0, y_0) < \varepsilon \Rightarrow \ \delta(x_k, y_k) > \beta}$$

- ***Sensibilité aux conditions initiales du système de Lorenz :***

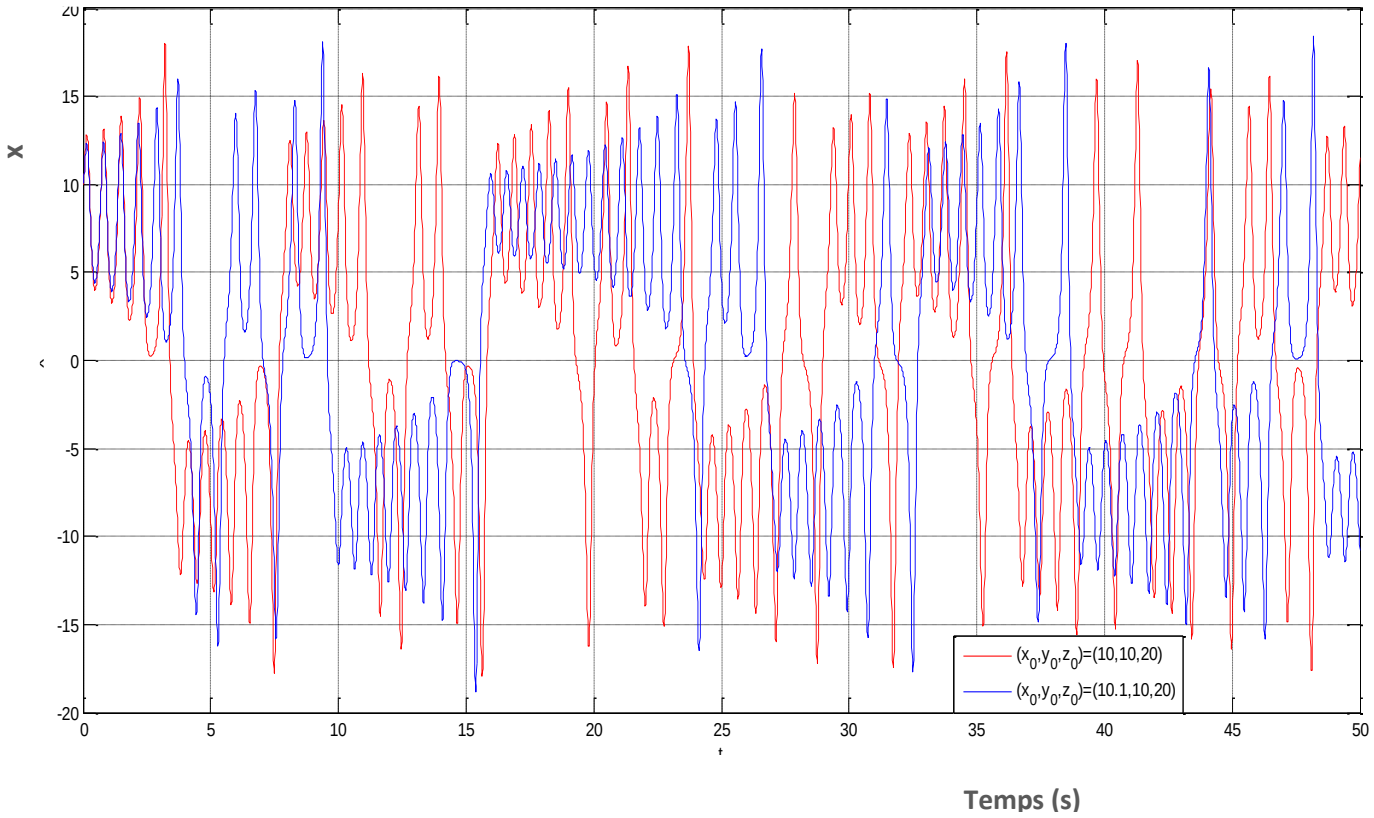



**Figure 5** : *Sensibilité aux conditions initiales (Système de Lorenz).*

- ***Sensibilité aux conditions initiales du système de Rössler :***

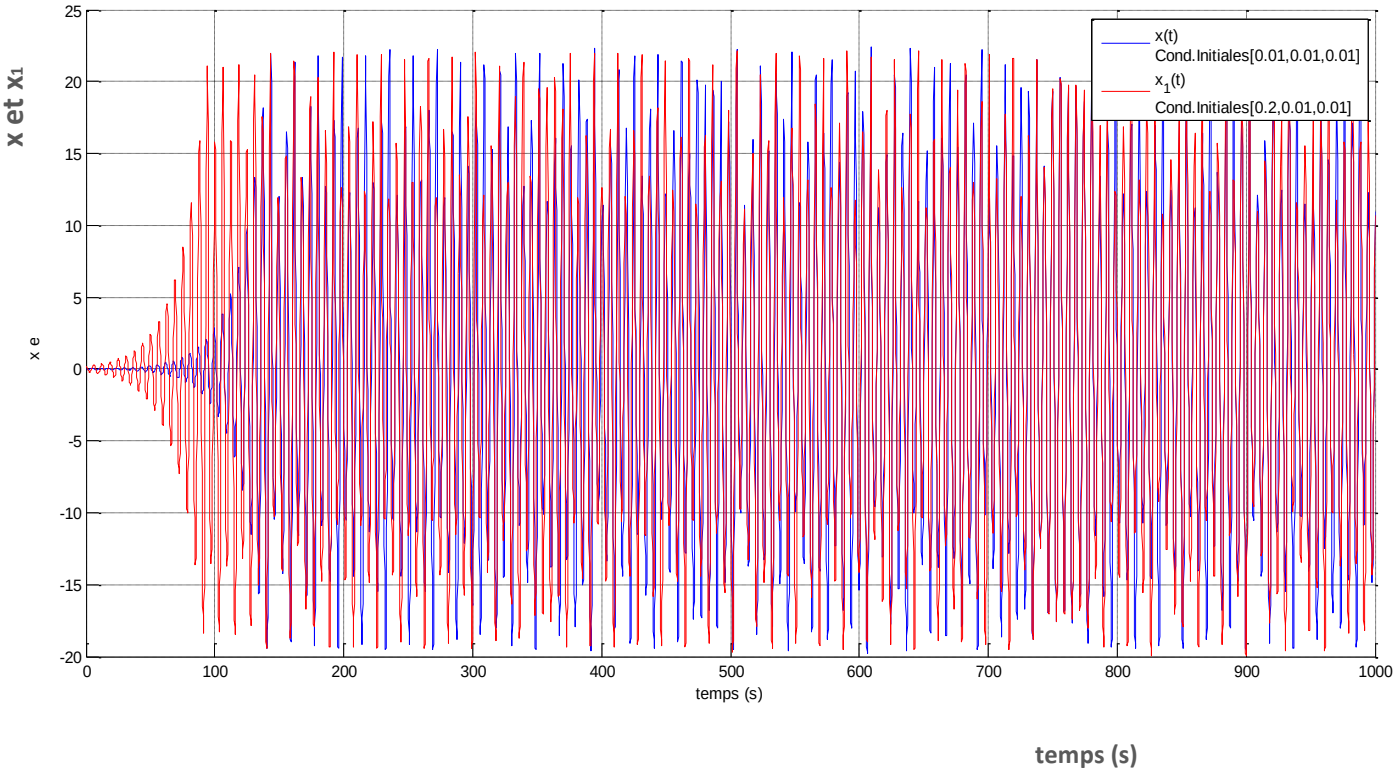



**Figure 6 :** *Sensibilité aux conditions initiales de l'état x (Système de Rössler).*

- ***Sensibilité aux conditions initiales du système de Hénon :***

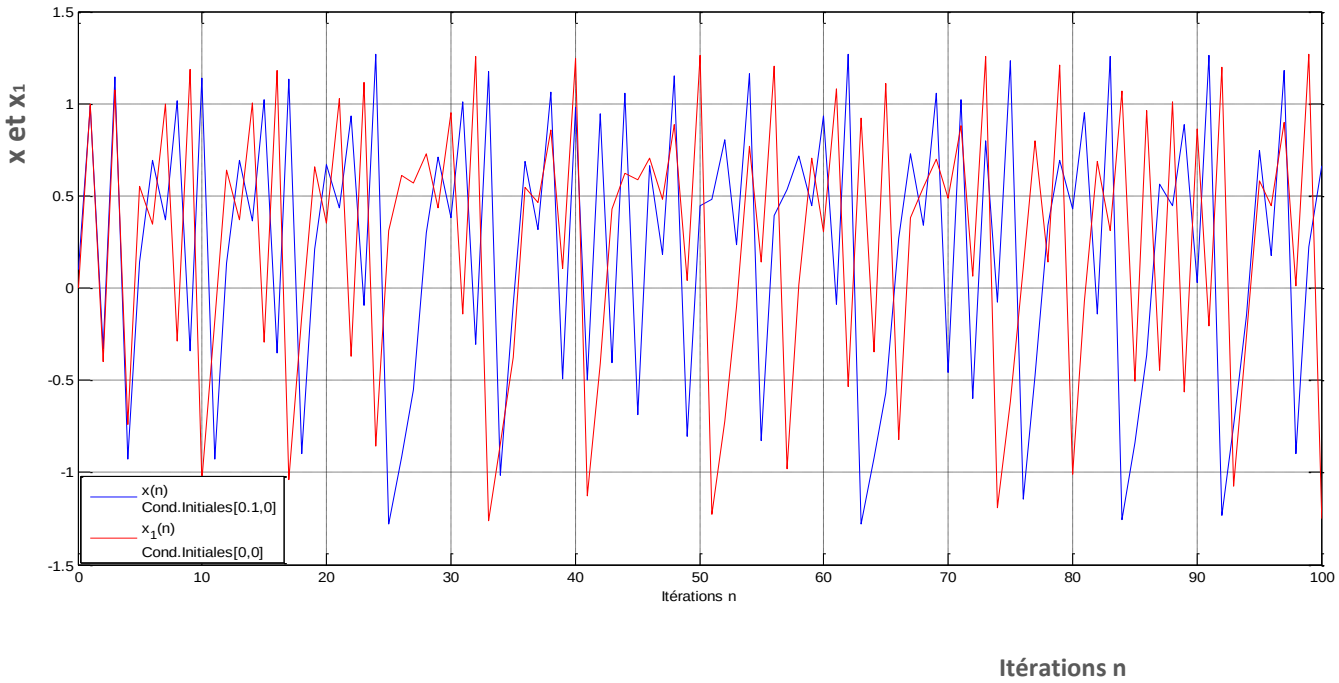



**Figure 7** : *Sensibilité aux conditions initiales de l'état x (Système de Hénon).*

- ***Sensibilité aux conditions initiales du système de Hénon-Heiles :***

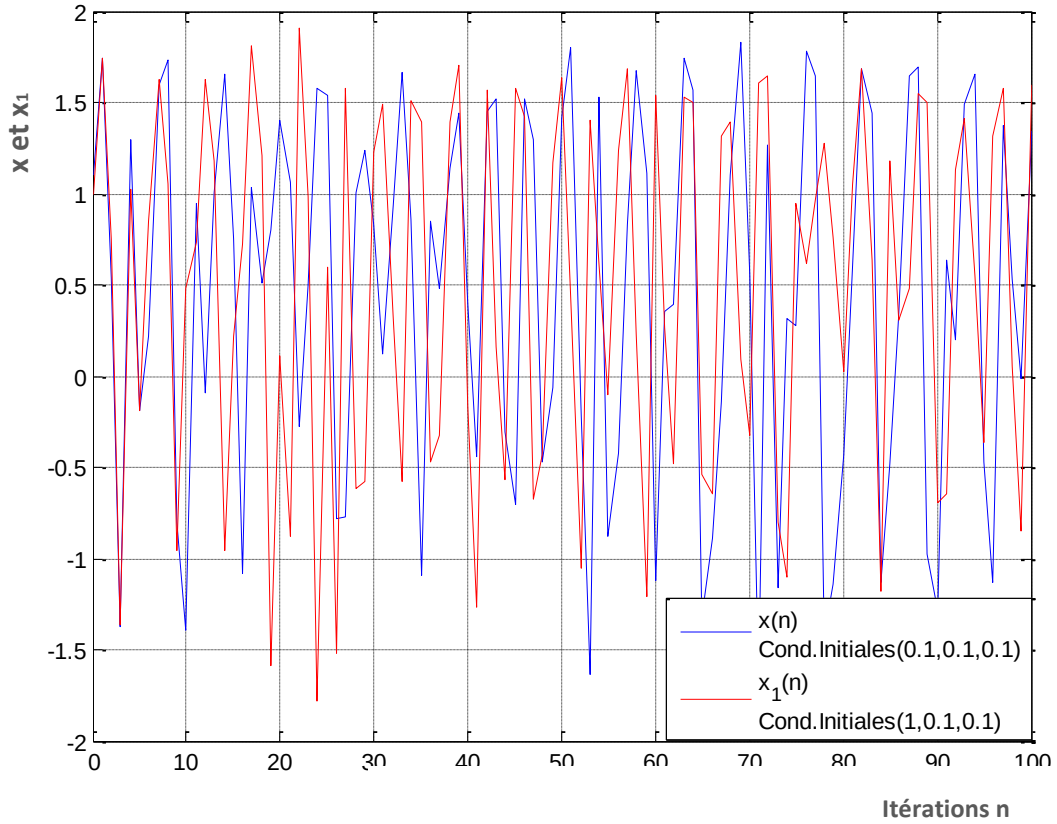



**Figure.8** : *Sensibilité aux conditions initiales de l'état x (Système de Hénon-Heiles)*

### 1.4.3 Notion d'attracteur :

Avant d'expliquer la notion d'attracteur, il faudrait d'abord définir ce qu'est l'espace des phases. Les trajectoires dynamiques des systèmes chaotiques sont fréquemment situées dans un espace appelé espace de phase. Les régions de l'espace sans l'existence permanente des dynamiques chaotiques seront inutiles puisque les points dans ces zones tendent vers l'infini et ne contribuent pas à la continuité du processus chaotique. Les variables qui construisent cet espace doivent contenir toute information sur la dynamique du système. On forme alors des équations chaotiques fonctionnant avec ces coordonnées dans l'espace et chaque itération de ces équations signifie l'incrémentation au temps suivant.

On peut maintenant définir un attracteur comme étant une limite asymptotique des solutions de toute condition initiale localisée dans un domaine de volume non nul ou bassin d'attraction.

Les trajectoires complexes dans l'espace de phase qui attirent les solutions du système chaotique sont alors des attracteurs. L'ensemble de points attirés vers l'attracteur constitue le bassin d'attraction. Autrement dit, l'attracteur est une figure géométrique de l'espace de phase (formant une structure feuilletée) indiquant le comportement d'un système chaotique. L'attracteur peut être étrange avec structure fractale (une courbe ou surface de forme irrégulière ou morcelée qui se crée en suivant des règles déterministes ou stochastiques impliquant une transformation ponctuelle de type homothétie interne) ou point fixe ou encore cercle limite. Parmi les premiers exemples des attracteurs étranges mentionnés dans l'histoire

du chaos, on cite l'attracteur de Lorenz. On donnera par la suite plusieurs exemples d'attracteurs étranges.

- **Attracteur de Lorenz :**

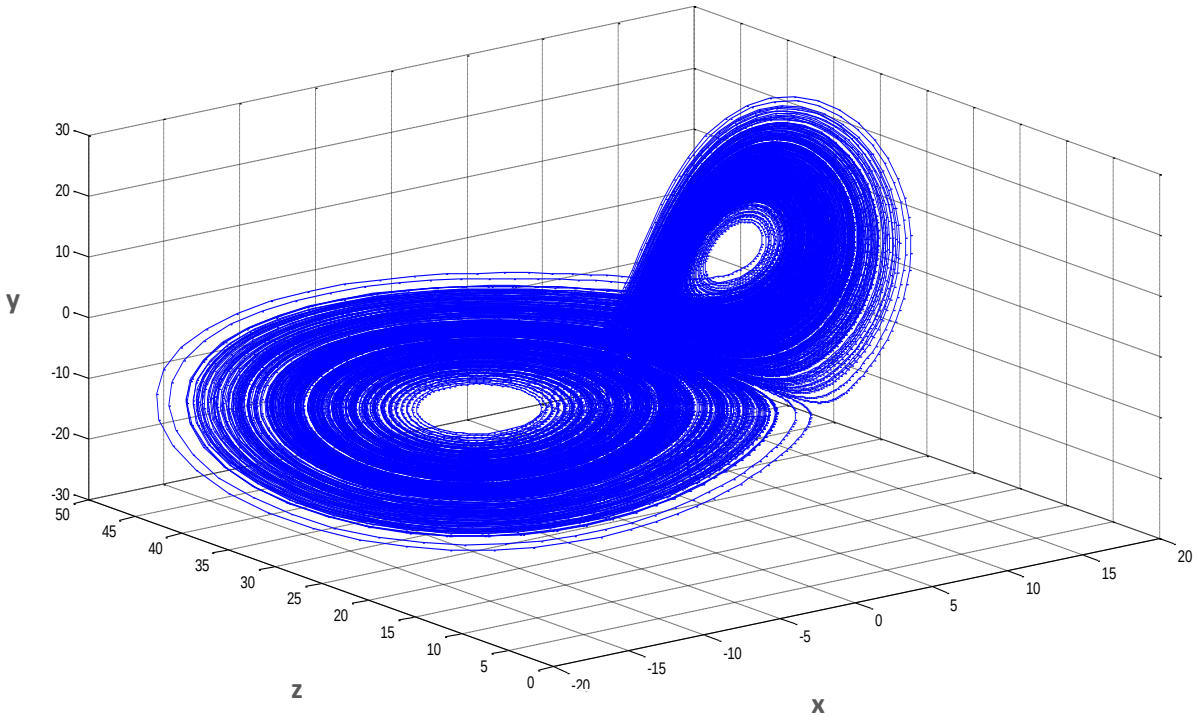



**Figure 9**: *Attracteur de Lorenz*

- **Attracteur de Rössler :**

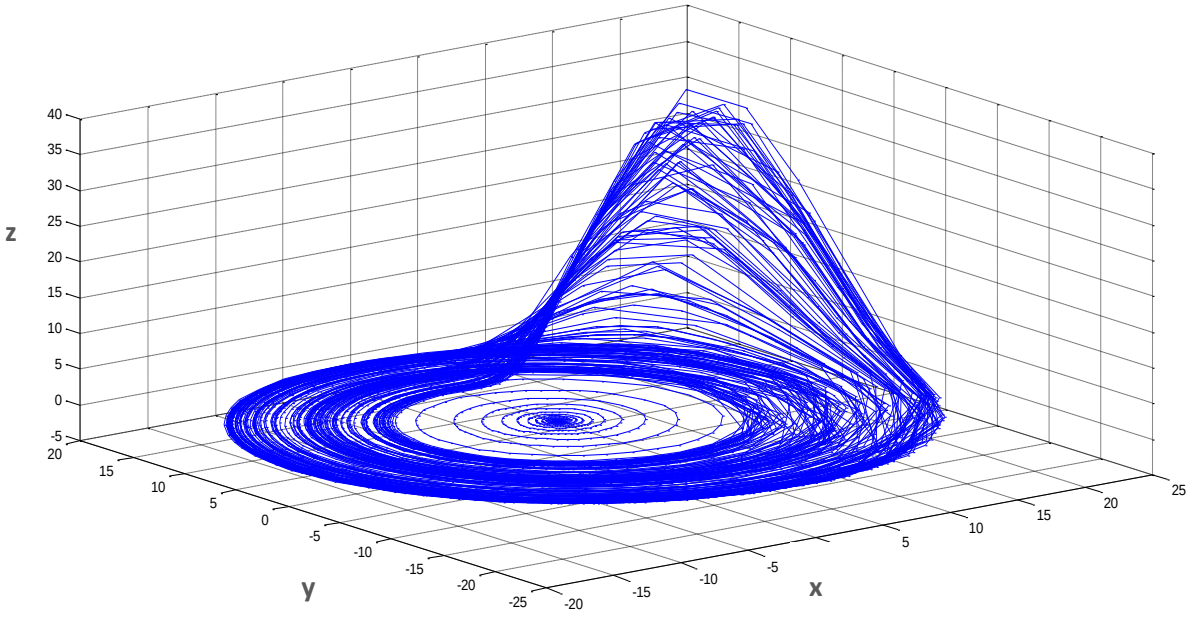



**Figure 10:** *Attracteur de Rössler*

- ***Attracteur de Hénon :***

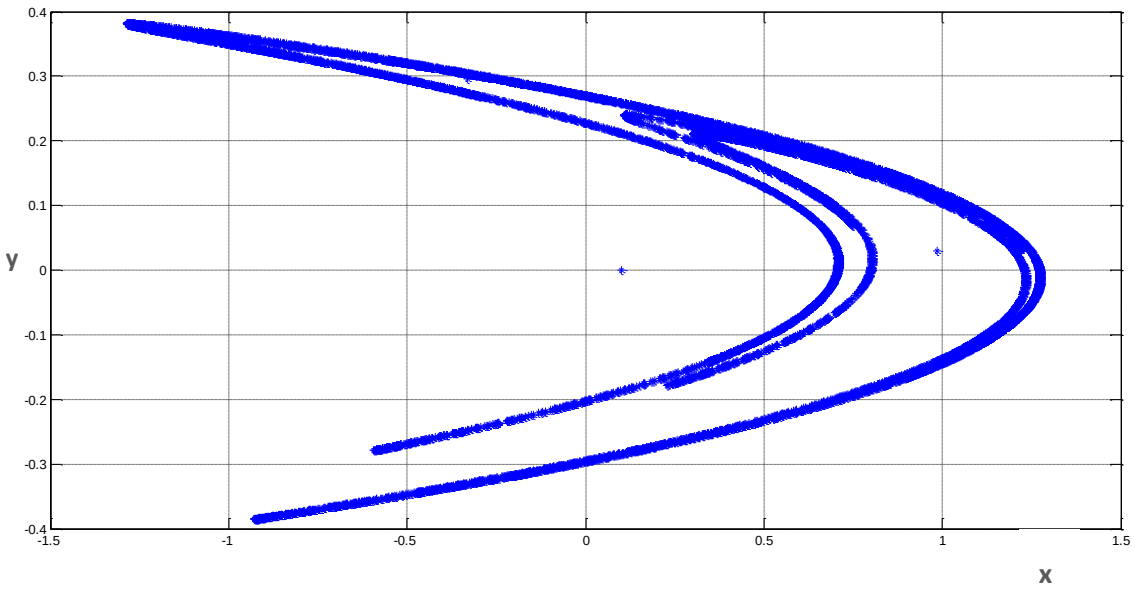



**Figure 11** : *Attracteur de Hénon.*

- ***Attracteur de Hénon-Heiles :***

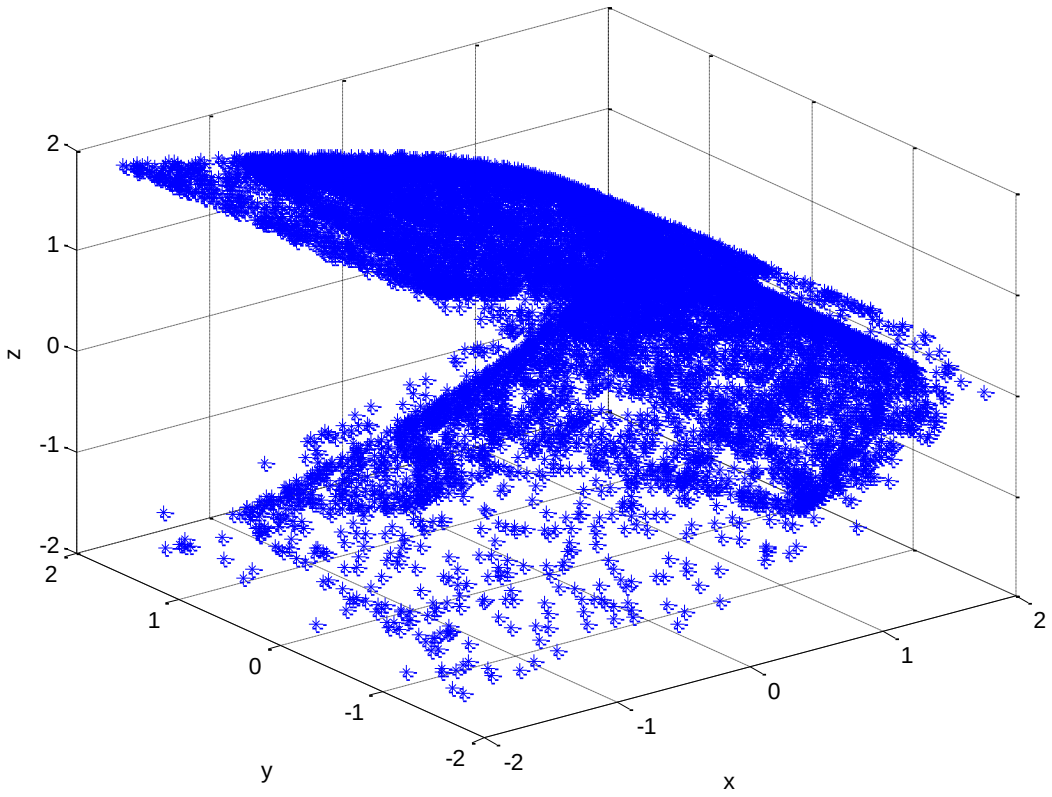



**Figure 12** : *Attracteur de Hénon-Heiles.*

### 1.4.4 Exposants de Lyapunov : [4]

Les exposants de Lyapunov permettent de caractériser le chaos temporel et plus particulièrement la sensibilité aux conditions initiales que peut présenter un attracteur étrange. Autrement dit, nous allons exposer comment calculer le taux de divergence entre l'évolution de trajectoires issues de conditions initiales proches au sein de cet espace borné qu'est l'attracteur étrange.

- Exposant pour une application unidimensionnelle : [11]

Considérons un système dynamique discret faisant intervenir une application $f$ et deux conditions initiales très proches, $x_0$ et $x_0 + \varepsilon_0$

La première itération conduit à : $x_1 + \varepsilon_1 = f(x_0) + \left(\frac{df(x_0)}{dx}\right)\varepsilon_0$ (8)

D'où l'on déduit : $\varepsilon_1 = \frac{df(x_0)}{dx}\varepsilon_0$. (9)

Après n itérations, il vient :

$$\boldsymbol{\varepsilon_n = \frac{df^n(x_0)}{dx}\varepsilon_0 = \left(\prod_{i=0}^{n-1}\frac{df(x_i)}{dx}\right)\varepsilon_0} \quad (10)$$

Les termes $\left(\frac{df^n(x_0)}{dx}\varepsilon_0\right)^{1/n}$ caractérisent la divergence. On définit alors **l'exposant de Lyapunov** par :

$$\boldsymbol{\lambda(x_0) = \lim_{n\to\infty}\frac{1}{n}\ln\left|\frac{df^n(x_0)}{dx}\right| = \lim_{n\to\infty}\frac{1}{n}\sum_{i=0}^{n-1}\ln|f'(x_i)|} \quad (11)$$

Un exposant positif indique que la divergence entre deux trajectoires voisines augmente exponentiellement avec le temps. Il s'agit bien là d'une caractérisation d'un attracteur étrange.

Il est possible d'étendre cette définition à une dimension plus élevée d'espace des phases. Pour un espace de dimension *p*, il y a *p* exposants de Lyapunov. Chacun d'entre eux mesure le taux de divergence suivant un des axes du système. Ils sont définis à partir de la matrice jacobienne de l'application *f* au point $x_0$ et de ses valeurs propres.

- Exposant pour une application multidimensionnelle :

Il s'agit de généraliser les concepts du paragraphe précédent à des trajectoires multidimensionnelles de type f : $R^m \rightarrow R^m$ : $x_{t+1} = f(x_t)$

Il faut savoir qu'un système m-dimensionnel possède m exposants de Lyapunov. Chacun d'entre eux mesure le taux de divergence suivant un des axes du système, de sorte qu'en moyenne un hyper-volume initial $V_0$ évolue selon une loi de type :

$$V = V_0 e^{(\lambda_1+\lambda_2+\cdots+\lambda_m)t} \quad (12)$$

Il est nécessaire qu'au moins un des $\lambda_i$ soit positif, pour avoir étirement et donc sensibilité aux conditions initiales selon au moins un axe. Mais il faut également que la somme des $\lambda_i$ soit négative. En effet, dans le cas contraire, le volume initial finirait par remplir tout l'espace dans lequel il est immergé. On n'aurait alors plus un attracteur de faible dimension, et donc plus affaire à du chaos déterministe.

Pour pouvoir définir et calculer $\lambda_i$ considérons une hyper-sphère dans l'espace m-dimensionnel de rayon $\varepsilon$ (petit) de conditions initiales et examinons son évolution. On s'intéresse à :

$$f^t(x_0 + \varepsilon) - f^t(x_0) \quad (13)$$

On pose $x'_0 = x_0 + \varepsilon$ et on opère un développement en série limité d'ordre 1 de $f'(x_0)$ au voisinage de $x'_0$ :

$$x_t - x'_t = \frac{df^t(x_0)}{dx}(x_0 - x'_0) = J^t(x_0)(x_0 - x'_0) \quad (14)$$

Où $J^t(x_0)$ dénote la matrice jacobienne de $f^t(.)$ au point $x_0$. Il s'agit d'une matrice carrée $m$x$m$. Si elle est diagonalisable, alors il existe une matrice inversible $P_t$ telle que :

$$D_m^t = P_t^{-1} J^t P_t \quad (15)$$

Où $D^t{}_m$ est une matrice diagonale contenant les valeurs propres de $J^t$. Dénotons celle-ci par $\Lambda_i^t$, $i$=1,…, $m$. On définit alors les $m$ exposants de Lyapunov de la manière suivante :

$$\lambda_i = \lim_{t\to\infty} \frac{1}{t} \ln[\Lambda_i^t] \qquad (16) \qquad (i = 1, \dots, m)$$

Le tableau suivant résume les différentes configurations d'exposants de Lyapunov évoquées précédemment.

| **Etat stable** | **Flot** | **Dimension de Lyapunov** | **Exposants de Lyapunov $\lambda_i$** |
|---|---|---|---|
| Point d'équilibre | Point | 0 | $\boldsymbol{\lambda_n \leq \cdots \leq \lambda_1 \leq 0}$ |
| Périodique | Cercle | 1 | $\boldsymbol{\lambda_1 = 0}$<br>$\boldsymbol{\lambda_n \leq \cdots \leq \lambda_2 \leq 0}$ |
| Période d'ordre 2 | Tore | 2 | $\boldsymbol{\lambda_1 = \lambda_2 = 0}$<br>$\boldsymbol{\lambda_n \leq \cdots \leq \lambda_3 \leq 0}$ |
| Période d'ordre K | K-tores | K | $\boldsymbol{\lambda_1 = \cdots = \lambda_K = 0}$<br>$\boldsymbol{\lambda_n \leq \cdots \leq \lambda_{K+1} \leq 0}$ |
| Chaotique | Attracteur chaotique | Non entier | $\lambda_1 > 0$<br>$\sum_{i=1}^{n} \lambda_i < 0$ |
| Hyperchaotique | Attracteur hypechaotique | Non entier | $\lambda_1 > 0$<br>$\lambda_2 > 0$<br>$\sum_{i=1}^{n} \lambda_i < 0$ |

**Tableau 1.1 :** *Attracteurs et exposants de Lyapunov*

Le logiciel LET (Lyapunov exponents toolbox) [12] nous a permis de calculer les exposants de Lyapunov sans passer par les méthodes de calculs mathématiques relativement longues que nous venons de citer.

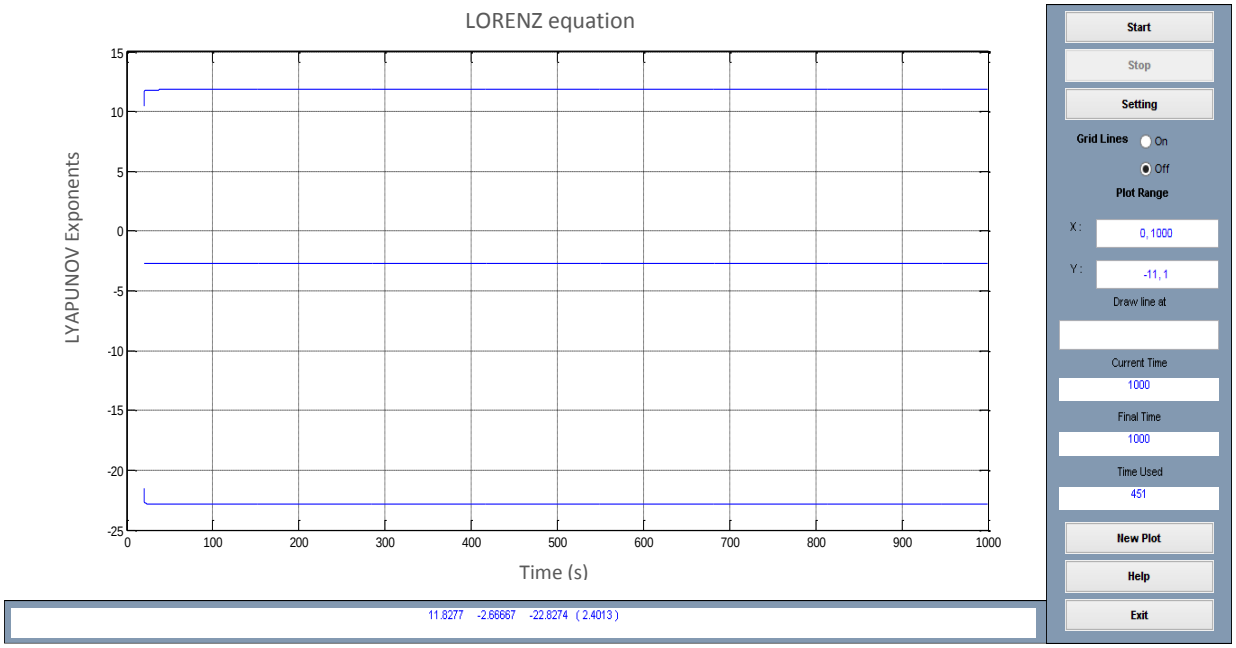



**Figure.13** : *Exposants de Lyapunov (Système de Lorenz)*

D'après la figure.13, nous avons 3 exposants de Lyapunov dont un positif :

$\lambda_1 = 11.8277$

$\lambda_2 = -2.66667$

$\lambda_3 = -22.8274$

Et $\lambda_1 + \lambda_2 + \lambda_3 = -13.66637$

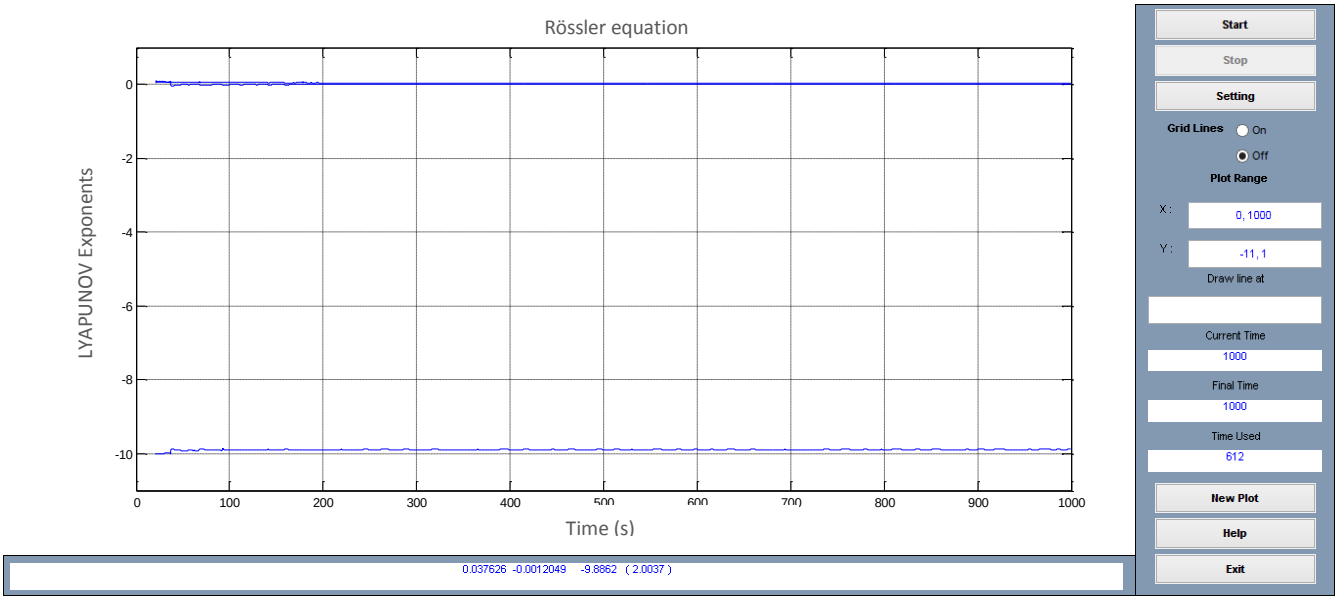



**Figure.14** : *Exposants de Lyapunov (Système de Rössler)*

D'après la figure.14, nous avons 3 exposants de Lyapunov dont un positif :

$\lambda_1 =$ 0.037626

$\lambda_2 =$ -0.0012049

$\lambda_3 =$ -9.8862

Et $\lambda_1 + \lambda_2 + \lambda_3 =$ -9.8497789

### 1.4.5 Fonction d'autocorrélation et spectre de puissance : [5]

Le spectre de puissance (ou densité spectrale d'énergie) d'un signal *f(t)* est la fonction :

$$\boldsymbol{\nu \rightarrow |TF\,[f](\nu)|^2}$$

TF : transformée de Fourier.

$|TF\,[f](\nu)|^2$ Mesure le « poids » de la fréquence ν dans la décomposition du signal *f(t)* en superposition de signaux élémentaires ($e^{i2\pi\nu t}$).

Dans le cas d'un système dynamique, le signal *f* pourra être une variable $X^i$ (t) de l'espace de phase, ou une observable O(X(t)). Un signal périodique ou quasi périodique (signe d'un attracteur en cycle ou tore limite) sera caractérisé par des pics isolés dans le spectre de puissance (pics correspondant à la fréquence fondamentale et aux différentes fréquences de battement présentes dans le signal).

Un système chaotique présentera des signaux avec des oscillations irrégulières. Ce qui caractérise les oscillations chaotiques sera la présence de bandes larges "continues" dans le spectre. Le système peut "passer continûment" d'une fréquence à l'autre dans la bande. Elles caractérisent donc une certaine perte d'information sur l'état initial due à la sensibilité aux conditions initiales. Il faudra néanmoins différencier cela d'un fond continu (bande continue qui s'étend sur tout ou presque tout le spectre) qui caractérise plutôt un bruit blanc, c'est à dire des fluctuations totalement aléatoires caractérisant une perte totale d'information à très court terme. De manière équivalente, on pourrait étudier la TF inverse du spectre de puissance :

$$\boldsymbol{TF^{-1}[|TF[f]|^2](t) = \int_{-\infty}^{+\infty} \overline{f(t-\tau)}\; f(\tau)d\tau = R_f(t)} \qquad \textbf{(17)}$$

*$R_f(t)$:* Fonction d'autocorrélation de *f*. On rappelle que la fonction d'autocorrélation *$R_f(t)$* en t, mesure la ressemblance du signal *f* avec lui-même décalé dans le temps d'une valeur de t. Pour un signal quasi périodique (signe d'un attracteur en tore limite), on s'attend donc à une fonction d'autocorrélation présentant des oscillations régulières avec de larges ventres centrés sur les différentes périodes présentes dans le signal et leurs harmoniques.

Pour un signal chaotique, on s'attend à une fonction d'autocorrélation présentant de fins petits pics (en sinus cardinal) formant une figure d'oscillations irrégulières. Cette autocorrélation est caractéristique de la propriété de mélange topologique (on a des petits pics d'autocorrélation car sur l'attracteur étrange qui est ergodique si le système passe par un point, il repassera dans le voisinage de celui-ci, mais de manière irrégulière et non-périodique). Pour un bruit blanc, on s'attend à un unique pic extrêmement fin en 0 (le signal décalé ne se ressemble jamais).

- **Exemple d'autocorrélation pour le système de Lorenz :**

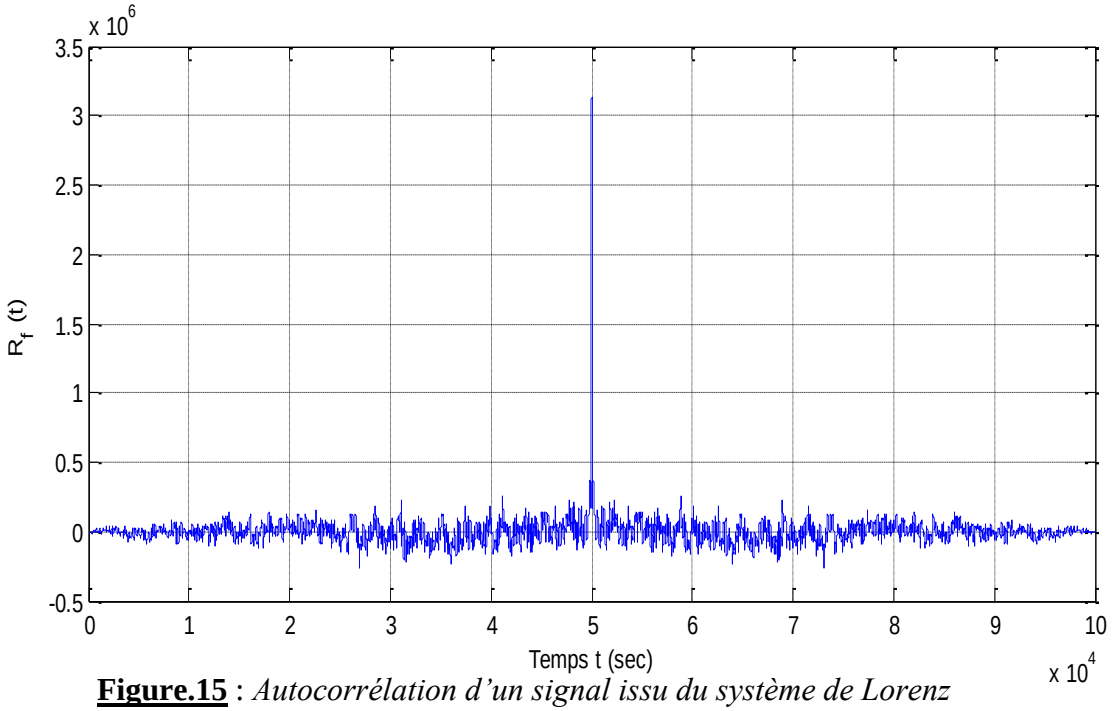



**Figure.15** : *Autocorrélation d'un signal issu du système de Lorenz*

- **Exemple d'autocorrélation pour le système de Rössler :**

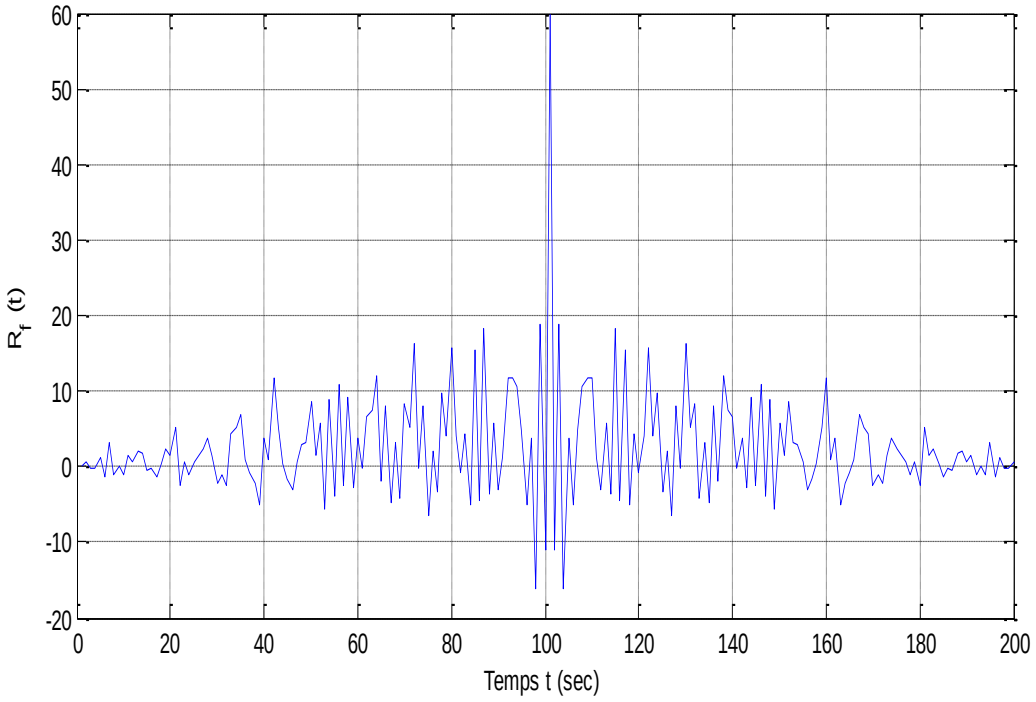



**Figure.16** : *Autocorrélation d'un signal issu du système de Rössler*

- **Exemple d'autocorrélation pour le système de Hénon:**

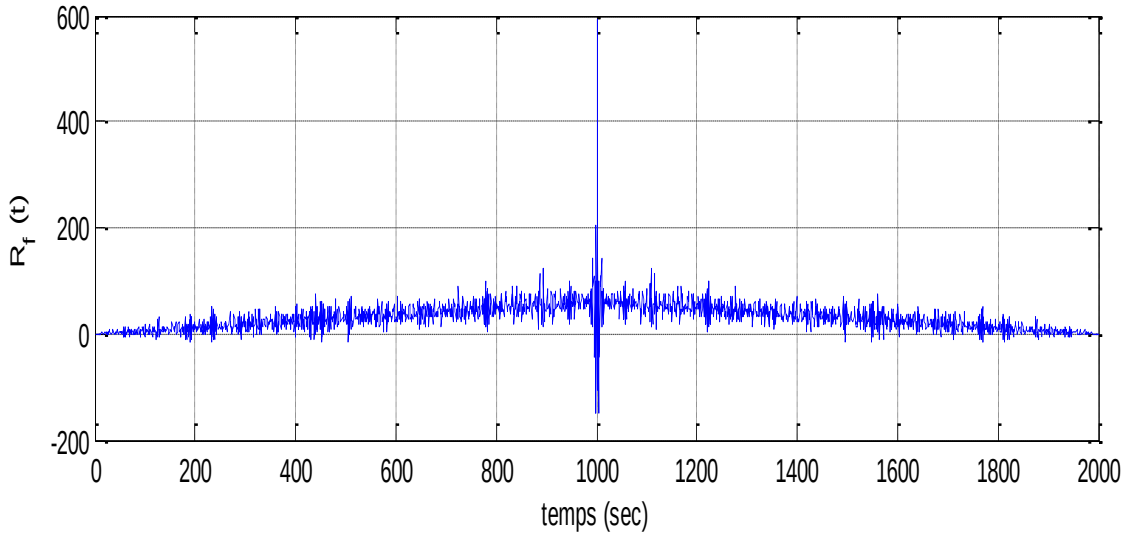



**Figure.17** : *Autocorrélation d'un signal issu du système de Hénon*

La fonction d'autocorrélation a une portée finie ; la similitude temporelle avec lui-même diminue et finie par disparaitre à des instants suffisamment éloignés l'un de l'autre. Au bout d'un certain temps, on ne peut plus prédire le comportement futur du signal, ce qui correspond à l'imprédictibilité du régime chaotique par perte progressive de sa similitude interne. Cette imprédictibilité est aussi à l'origine de l'écartement de deux trajectoires initialement voisines qui perdent toute similitude au bout d'un temps fini, ce qui correspond à la sensibilité aux conditions initiales des systèmes chaotiques.

Il est plus simple de représenter le spectre d'amplitude $|TF\ [f](\nu)|$ (figure 18) plutôt que le spectre de puissance $|TF\ [f](\nu)|^2$

- Spectre d'amplitude du système de Lorenz :

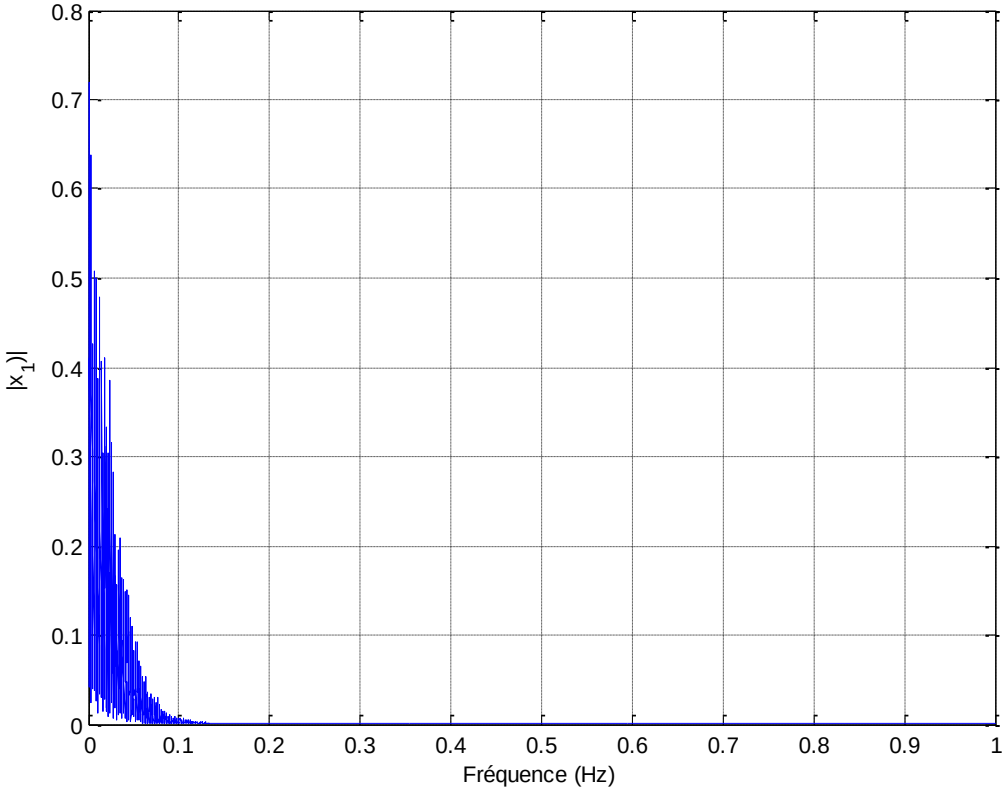



**Figure.18** : *Spectre d'amplitude du système de Lorenz*

### 1.4.6 Bifurcation : [6]

Le passage d'un point fixe à un cycle limite de période-2, puis à un cycle limite de période-4, est un évènement important dans la dynamique d'un système. On dit qu'il y a une bifurcation lorsqu'un tel changement qualitatif des solutions se produit à l'occasion de la variation d'un paramètre. Les graphiques qui explicitent ces bifurcations, sont logiquement appelés diagrammes de bifurcation. Cette notion est centrale dans l'étude du chaos. Lorsque l'on examine de tels graphiques, il faut faire attention aux axes. Sur un axe nous prenons le paramètre, et sur l'autre la variable d'état, formant l'espace paramétrique.

Un diagramme de bifurcation délimite des zones de l'espace paramétrique dans lesquelles le comportement qualitatif du système est similaire. On voit apparaitre aussi un enchainement très rapide de doublements de période qui mène à une situation chaotique. Ce mécanisme de doublements de période est une des routes vers le chaos. Un exemple de diagramme de bifurcation (système de Hénon modifié) est représenté sur la *figure 19*

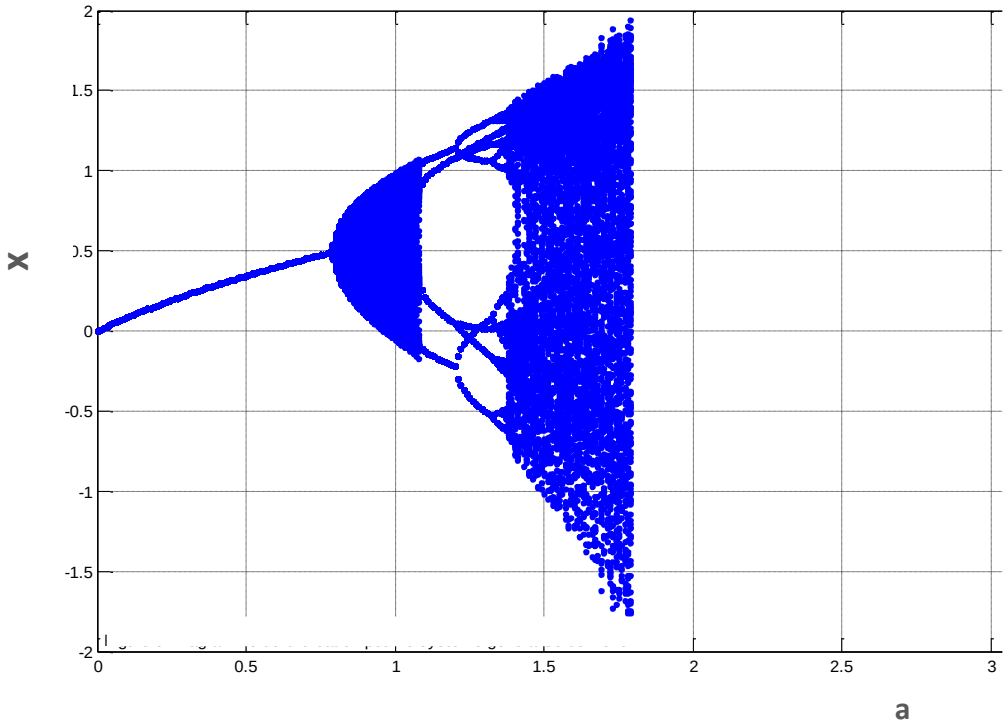



**Figure 19** : *Diagramme de bifurcation pour le système de Hénon*-Heiles

Dans notre exemple du Hénon modifié, lorsque le nombre d'itérations augmente :

- Pour a croissant de 0 à 0.8 : le système a tendance à se stabiliser autour d'une seule valeur, il est d'abord attiré par un cycle limite de période-1 ;
- Pour 0.8 < a < 1.06 : le système finit par osciller entre 2 valeurs, on dit qu'il possède un cycle de période-2 (dédoublement de période) ;
- Pour 1.06 < a < 1.37 : la longueur du cycle augmente de plus en plus rapidement;
- Pour 1.37< a < 1.8 : la longueur du cycle s'allonge et devient tellement complexe que l'on peut difficilement suivre son évolution, le système passe dans une phase chaotique.

Des bifurcations successives dans les phénomènes chaotiques peuvent engendrer une structure fractale. Les fractales et le chaos déterministe sont deux domaines mathématiques qui présentent beaucoup de points communs, bien que leurs caractéristiques soient différentes (imprévisibilité et sensibilité aux conditions initiales pour le chaos et autosimilarité et invariance d'échelle pour les fractales) [7] [8]. Ainsi, de nombreux phénomènes chaotiques présentent des structures fractales (par exemple, dans leurs attracteurs étranges), même si beaucoup d'objets fractals ne sont nullement chaotiques (triangle de Sierpinsky, courbe de Koch...).

## 1.5Scénarios vers le chaos : [10]

Le chaos naît toujours d'une instabilité liée à la présence d'un paramètre de contrôle dans les équations d'évolution. Lorsque ce dernier prend une valeur particulière, dite critique, le système subit une bifurcation : il change brusquement de comportement dynamique.

Une succession de bifurcations peut alors conduire à un comportement chaotique.

Les différents processus qui conduisent au chaos sont:

La cascade sous harmonique ou doublement de période, l'intermittence ou transition de Pomeau-Manneville et la quasi-périodicité.

- **Cascade sous harmonique ou dédoublement de période : cf paragraphe sur la bifurcation.**

Il existe un système simple qui illustre bien le mécanisme d'évolution vers le chaos par cascade sous harmonique. C'est l'oscillateur étudié dans les années 1980 par l'électronicien américain Léon Chua (annexe). Dans ce système, les grandeurs électriques évoluent en étant alternativement croissantes puis décroissantes.

Dans la *figure 20* a été représentée l'évolution d'une tension u prélevée dans le circuit pour différentes valeurs d'un paramètre de contrôle, en pratique une résistance *r*.
La réalisation expérimentale du circuit de Chua montre que la période des oscillations varie par doublements successifs, lorsque *r* dépasse des paliers $r_1$, $r_2$, $r_3$, Au fur et à mesure que l'on augmente *r*, la période, qui vaut initialement $T_0$, devient $2T_0$, puis $4T_0$, $8T_0$, etc: elle s'allonge indéfiniment. Le circuit devient apériodique, son évolution atteint un régime chaotique.

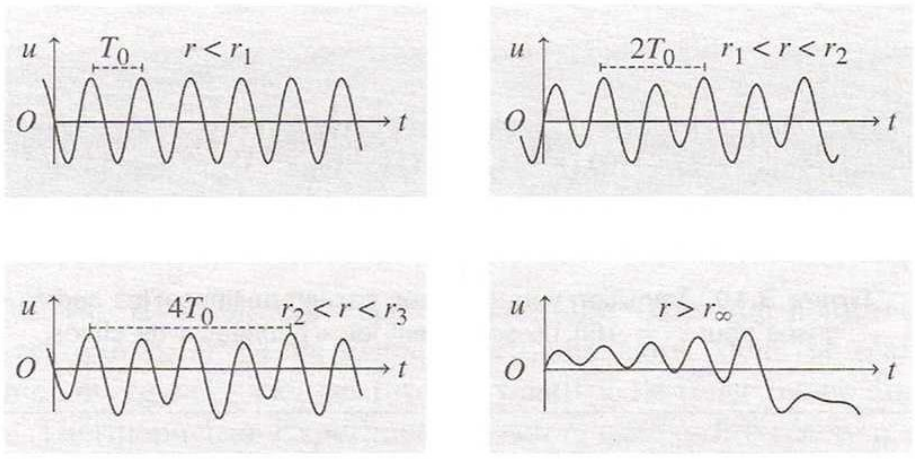



**Figure.20** : *Cascade sous harmonique dans le montage de Chua*

- **Intermittence ou transition de Pomeau-Manneville** :

Ce processus de transition d'un régime périodique à un régime chaotique par intermittence a été découvert par les physiciens français Yves Pomeau et Paul Manneville en 1980.

Dans leur publication, ces auteurs reprennent les équations du modèle de Lorenz de la convection, mais remplacent le nombre 28 par le paramètre *r* qu'ils font varier autour de la valeur critique $r_c$ = 166,06. Le calcul par simulation de l'évolution de la variable *z* montre l'apparition du chaos par intermittence : pour *r* = 160, *z* a un comportement périodique, alors que, si *r* est très légèrement supérieur à $r_c$, ce comportement est interrompu par des « bouffées » irrégulières de courte durée, dont l'apparition semble aléatoire; si *r* continue d'augmenter, ces perturbations sont de plus en plus fréquentes et plus longues, si bien qu'au final l'évolution de *z* apparaît complètement aléatoire et donc chaotique (*figure 21*).

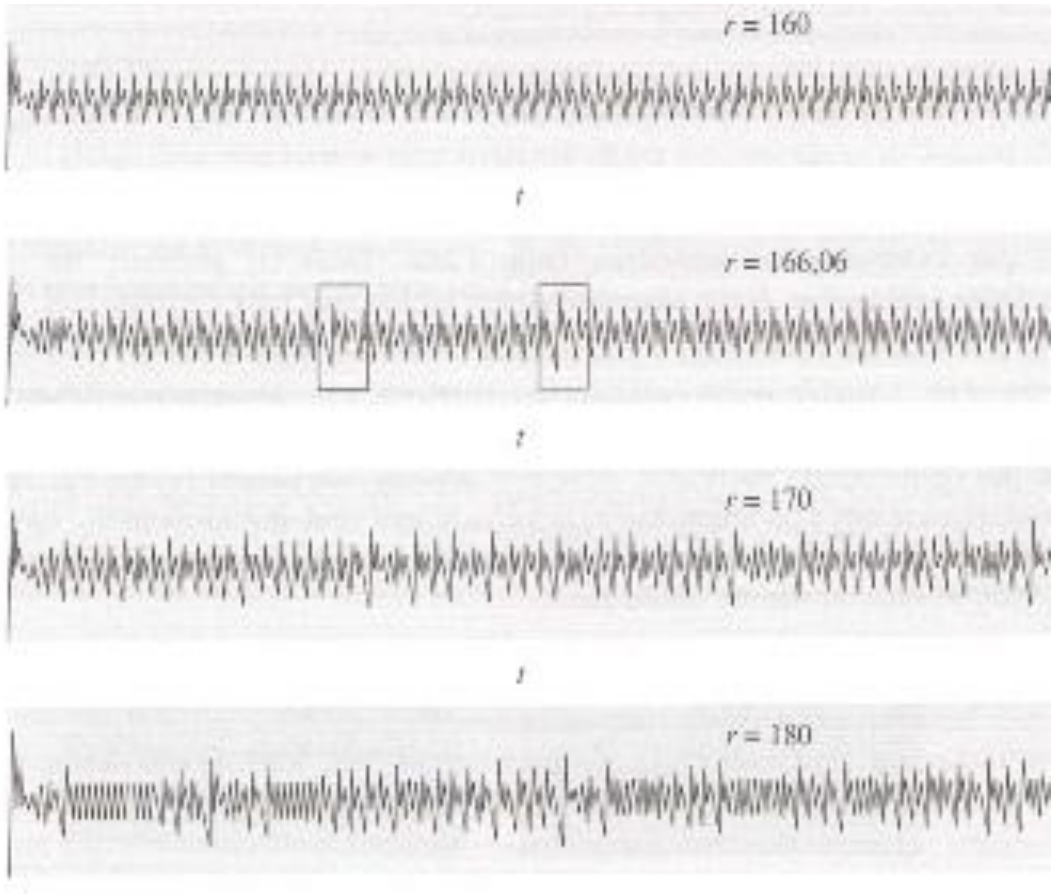



**Figure.21** : *Transition vers le chaos par intermittence ; les cadres grisés pour r=166.06 soulignent les « bouffées » de turbulence du chaos.*

- **Quasi-périodicité:**

Un système est *quasi-périodique* s'il est constitué de deux oscillateurs, de périodes respectives $T_1$ et $T_2$ telles que leur rapport ne soit pas un nombre rationnel. Le système semble présenter alors un mouvement périodique, mais il ne repasse jamais par le même état; au mieux se rapproche-t-il indéfiniment de ce dernier. La transition vers le chaos par quasi-périodicité intervient quand un oscillateur interagit avec un système initialement périodique : au fur et à mesure que le paramètre de contrôle (ou d'interaction) augmente, le système adopte un comportement quasi- périodique jusqu'à devenir chaotique.

## 1.6 Conclusion :

Dans ce chapitre, nous avons présenté quelques notions sur le chaos. La première partie de ce chapitre est consacrée à la définition des systèmes dynamiques linéaires, non linéaires et chaotiques. Toujours dans cette même partie, les deux classes de systèmes chaotiques sont présentées (continus et discrets) avec leurs exemples et leurs propriétés.

En fin de chapitre, les scénarios de transition vers le chaos des systèmes dynamiques ont été présentés. Ces notions seront exploitées dans les chapitres qui vont suivre où nous allons aborder le problème de la synchronisation des systèmes chaotiques.

## 2.1. Introduction :

La synchronisation est un phénomène qui caractérise deux systèmes se comportant de la même façon en même temps. Les manifestations de la synchronisation sont observées depuis le XVIIème siècle. Le mathématicien hollandais Huygens a notamment remarqué que deux horloges à balancier placées contre un mur finissaient par avoir des mouvements identiques. D'où la définition de la synchronisation : deux signaux périodiques sont synchronisés s'ils ont des périodes identiques. Par conséquent, la synchronisation semble à première vue réservée aux mouvements périodiques. Par ailleurs, l'une des propriétés qui caractérise le chaos est justement l'absence de périodicité. A priori, il peut sembler totalement inconcevable de relier les deux phénomènes : synchronisation et chaos. Et pourtant, cette idée reçue a volé en éclats tout d'abord en 1983, et surtout en 1990, lorsque les deux chercheurs Pecora et Carroll ont montré que deux systèmes chaotiques peuvent se synchroniser. Leurs travaux novateurs ont totalement changé la façon d'appréhender les manifestations du chaos. En effet, jusque-là, l'apparition de phénomènes chaotiques dans l'évolution de systèmes dynamiques était redoutée car incontrôlable, et donc évitée à tout prix. La théorie du contrôle du chaos, puis les articles de Pecora et de Carroll ont ainsi ouvert la voie à une recherche foisonnante sur les applications du chaos. C'est ainsi que l'utilisation du chaos dans la cryptographie a vu le jour. On peut noter qu'il existe différents types de synchronisation, selon la nature de la connexion entre le système émetteur et le système récepteur (unidirectionnelle, bidirectionnelle).

Dans ce chapitre, nous abordons ces deux types de synchronisation, ainsi que les différentes méthodes de synchronisation, mais aussi les différentes techniques de cryptage chaotique. Tous ces rappels, seront donnés dans le but de montrer comment insérer le processus de synchronisation au sein d'un système de communication.

## 2.2. Communication sécurisée à base du chaos : [13]

Dans ce chapitre, on s'intéresse aux techniques de transmission sécurisée d'informations qui reposent sur le principe de synchronisation chaotique.

L'emploi d'un signal chaotique dans le domaine des télécommunications pose directement le problème de synchronisation du récepteur dans le but de dupliquer le signal chaotique employé à l'émetteur. Dans le chapitre précédent nous avons montré la sensibilité très importante aux conditions initiales des systèmes chaotiques, et à première vue la synchronisation chaotique parait difficile à réaliser. A la différence de la synchronisation classique employée dans les systèmes de télécommunication où l'on cherche à reproduire juste une période d'oscillation, la synchronisation chaotique présente plus de contraintes.

De nombreux travaux ont été présentés ces dernières années exploitant les signaux chaotiques dans le contexte des télécommunications. Pecora et Carroll [14] ont défini la synchronisation chaotique connue sous le nom de synchronisation identique, développée sur la base de circuits chaotiques couplés, avec l'un maître et l'autre esclave ; Ces travaux ont ouvert la voie des applications du chaos aux télécommunications. Le point commun constaté dans la majorité des techniques développées dans la littérature est l'utilisation de la configuration maitre-esclave pour laquelle on dispose d'un émetteur chaotique (système maitre) qui génère le signal du texte

chiffré transmis dans le canal de communication vers un système récepteur (système esclave) qui a pour objectif de synchroniser avec le système maitre et de restaurer le signal d'information.

La ***figure.22*** présente l'architecture générale d'un système de communication basé sur le chaos. Un signal m(t) contenant l'information à transmettre est injecté à l'entrée d'un émetteur présentant un comportement chaotique. Celui-ci génère un signal y(t) chaotique qui est transmis au récepteur.

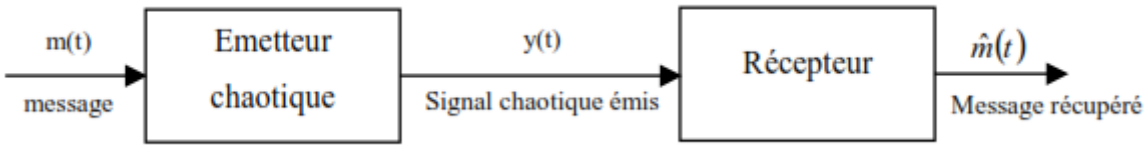



**Figure.22** : *Architecture d'un système de communication chaotique.*

Le récepteur est piloté par le signal y(t) et réalise l'opération inverse de celle effectuée à l'émission afin de récupérer le message transmis. L'émetteur, pour transmettre le message, mélange celui-ci avec un signal chaotique (différentes techniques pour coder le message avec un signal chaotique seront abordées dans un paragraphe ultérieur). De ce fait, il apparaît difficile pour le récepteur de récupérer l'information contenue dans le signal chaotique. Le récepteur est en général similaire à l'émetteur et peut présenter un comportement chaotique lorsqu'il n'est pas piloté par le signal y(t). Pour pouvoir extraire l'information du signal reçu, il est nécessaire de synchroniser les deux systèmes.

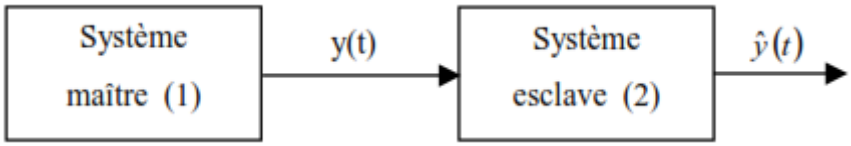



**Figure.23** : *Système maître-esclave pour réaliser la synchronisation.*

La notion de synchronisation est en général liée à un mouvement périodique. Deux signaux sont synchronisés si leur période ainsi que leur phase sont identiques. Cette définition n'est plus valable dans le cas de signaux chaotiques. Deux signaux chaotiques seront dits synchronisés s'ils sont asymptotiquement identiques lorsque t tend vers l'infini.

Considérons la Figure.23 dans laquelle, le système (2), dit système esclave, sera synchronisé avec le système (1) (système maître) si :

$$\lim_{t\to\infty}|\hat{y}(t) - y(t)| = 0 \qquad (2.1)$$

Et ce, quel que soit l'état initial des deux systèmes.

A priori, il paraît impossible d'arriver à synchroniser deux exemplaires d'un même système chaotique. D'une part parce que dans les systèmes réels, il est extrêmement difficile de construire deux circuits à l'identique à cause de la tolérance sur les composants ainsi que du bruit présent dans tout système électronique.

D'autre part, en supposant que l'on dispose de deux circuits identiques, il se pose le problème de la sensibilité aux conditions initiales qui se traduit par une instabilité au sens de Lyapunov. Une infime différence entre les conditions initiales des deux circuits conduira à des signaux totalement différents. Cela signifie que reproduire ces conditions initiales dans un système réel est impossible.

Cependant, dans le cas de la ***Figure.23***, c'est le signal pilote y(t) qui force le système esclave à suivre le comportement du système maître ce qui permet, à terme, une synchronisation entre les systèmes (1) et (2).

## 2.3. Méthodes de synchronisations :

Les méthodes traditionnelles de synchronisation chaotiques sont en général basées sur l'utilisation de circuits identiques. Supposons deux systèmes chaotiques identiques oscillant de façon totalement indépendante. Si par un moyen quelconque, on leur permet d'échanger de l'énergie, action que l'on nomme « couplage », les deux systèmes finiront par céder la place à un comportement commun : ils se synchronisent.

On distingue deux types de synchronisation, classés selon le sens dont l'énergie est échangée entre les deux systèmes chaotiques (notion de couplage) ; *la synchronisation par couplage unidirectionnel et la synchronisation par couplage bidirectionnel.*

Nous allons aussi présenter les différentes méthodes qui permettent de réaliser cette synchronisation (par couplages unidirectionnel et bidirectionnel).

### 2.3.1. Synchronisation par couplage unidirectionnel [13]:

Lors d'une synchronisation par couplage unidirectionnel, l'énergie est transférée d'un système à l'autre entre deux systèmes identiques a et b à l'aide d'un élément de couplage fonctionnant dans un seul sens, comme par exemple un suiveur.

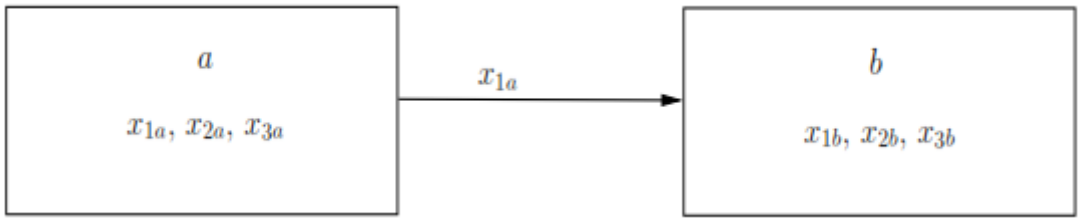



**Figure 24:** *Schéma de couplage unidirectionnel.*

### 2.3.2. Synchronisation par couplage bidirectionnel [13] :

Lors d'une synchronisation par couplage bidirectionnel, l'élément de couplage permet l'échange de l'énergie dans les deux sens, ceci peut être par exemple une simple résistance.

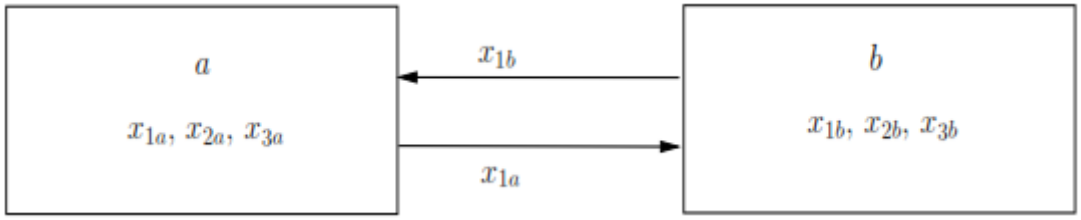



**Figure 25:** *Schéma de couplage bidirectionnel.*

### 2.3.3. Synchronisation par décomposition du système (synchronisation identique) : [20][21][22].

La synchronisation identique proposée par Pecora et Carroll [14] a l'avantage de représenter une solution simple et performante de synchronisation dont l'objectif est que l'esclave reproduise le plus fidèlement possible l'état du maître, après un régime transitoire.

Ce concept repose sur le constat qu'un système chaotique possède un ou plusieurs exposants de Lyapunov positifs et qu'il est instable. Il est donc impossible de construire une réplique identique à ce système et d'essayer de synchroniser tel quel les deux signaux chaotiques issus chacun des deux exemplaires. L'idée consiste à diviser le système d'origine en deux sous-systèmes de telle sorte que les variables dynamiques de départ soient réparties de part et d'autre de chacun des sous-systèmes. Il s'agit ensuite de reproduire les sous-systèmes à l'identique et de les mettre en cascade. Le signal issu du système de départ (système maître) sert à piloter

(synchroniser) le premier des deux sous-systèmes dupliqués mis en cascade qui lui-même permet de synchroniser le second sous-système dupliqué.

Considérons un système dynamique autonome (ANNEXE), en temps continu, de dimension n, représenté par la relation suivante :

$$\dot{\boldsymbol{u}} = \boldsymbol{F}(\boldsymbol{u}) \qquad (2.2)$$

$\boldsymbol{u} \in \mathbb{R}^{\boldsymbol{n}}$

Avec $u(t) = (u_1(t), \ldots u_n(t))$ et $F(u) = (F_1(u), \ldots F_n(u))$.

Ce système est divisé arbitrairement en deux sous-systèmes :

$$\dot{\boldsymbol{x}} = \boldsymbol{G}(\boldsymbol{x}, \boldsymbol{y_1}) \;\; \boldsymbol{et}\; \dot{\boldsymbol{y}} = \boldsymbol{H}(\boldsymbol{x_1}, \boldsymbol{y}) \qquad (2.3)$$

Avec $x(t) = (u_1(t), \ldots, u_m(t)) = (x_1(t), \ldots, x_m(t))$

*Et* $y(t) = (u_{m+1}(t), \ldots, u_n(t)) = (y_1(t), \ldots, y_p(t))$

Tel que $m + p = n$

Soient :

$$\begin{cases} \dot{\boldsymbol{x}}_{\boldsymbol{1}} = \boldsymbol{G}_{\boldsymbol{1}}(\boldsymbol{x}, \boldsymbol{y}_{\boldsymbol{1}}) \\ \quad . \\ \quad . \\ \quad . \\ \dot{\boldsymbol{x}}_{\boldsymbol{m}} = \boldsymbol{G}_{\boldsymbol{m}}(\boldsymbol{x}, \boldsymbol{y}_{\boldsymbol{1}}) \end{cases} \qquad (2.4) \qquad \boldsymbol{et} \qquad \begin{cases} \dot{\boldsymbol{y}}_{\boldsymbol{1}} = \boldsymbol{H}_{\boldsymbol{1}}(\boldsymbol{x}_{\boldsymbol{1}}, \boldsymbol{y}) \\ \quad . \\ \quad . \\ \quad . \\ \dot{\boldsymbol{y}}_{\boldsymbol{p}} = \boldsymbol{H}_{\boldsymbol{p}}(\boldsymbol{x}_{\boldsymbol{1}}, \boldsymbol{y}) \end{cases} \qquad (2.5)$$

- Figure représentant la séparation en deux sous-systèmes :

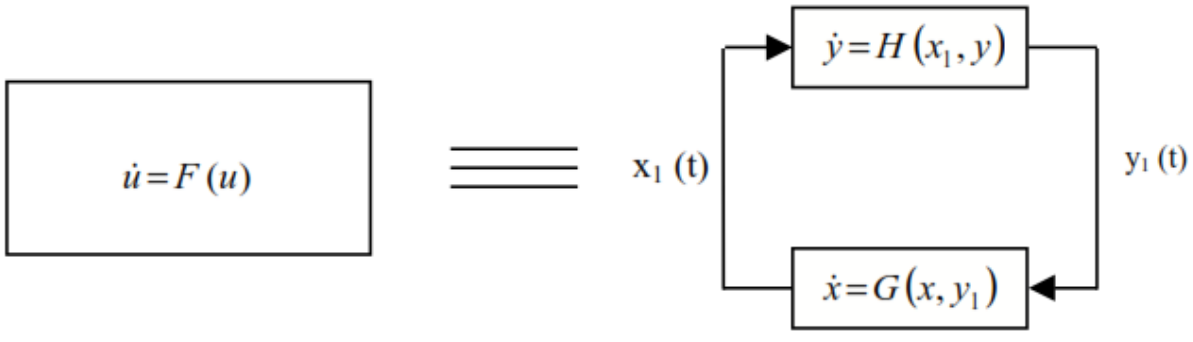



**Figure.26** : *Séparation du système F en deux sous-systèmes G et H*

Le sous-système G, dont les variables d'états sont décrites par le vecteur x, et le sous-système H, de variables d'états y, sont reliés par les variables $x_1(t)$ et $y_1(t)$.

Ces deux sous-systèmes sont ensuite dupliqués et mis en cascade comme le montre la figure.27.

Soient $\hat{G}$ $et$ $\hat{H}$ ces deux sous-systèmes.

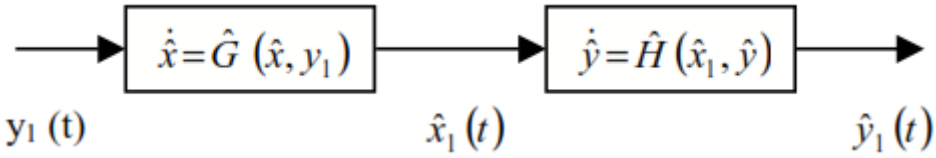



**Figure.27** : *Mise en cascade des deux sous-systèmes dupliqués.*

L'objectif, maintenant, est de synchroniser le signal $\widehat{y_1}(t)$ avec le signal $y_1(t)$ provenant du système d'origine (***Figure.28***)

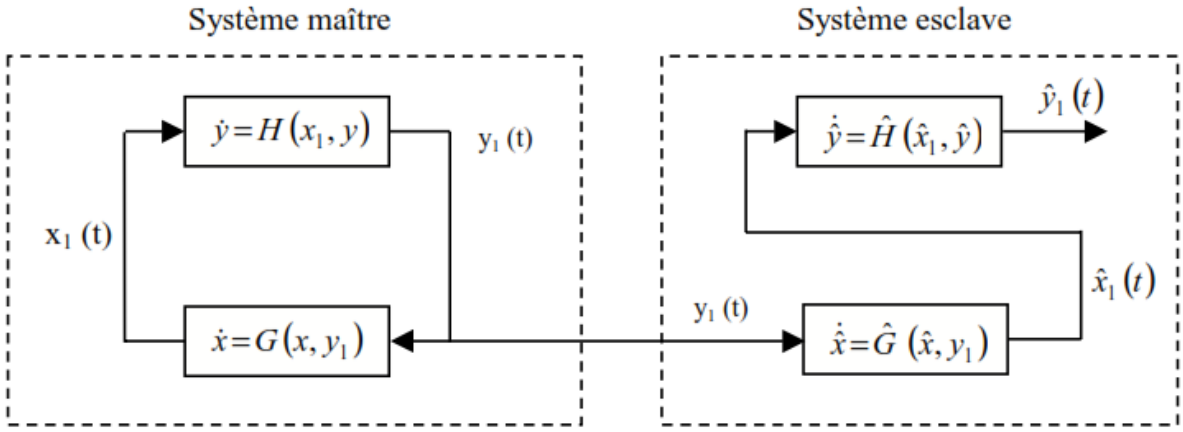



**Figure.28** : *Principe de synchronisation par décomposition en sous-systèmes*

Si l'ensemble des quatre sous-systèmes est considéré comme un système unique alors ce dernier peut être décrit par les équations d'états suivantes :

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{G}(\boldsymbol{x}, \boldsymbol{y_1}) \\ \dot{\boldsymbol{y}} = \boldsymbol{H}(\boldsymbol{x_1}, \boldsymbol{y}) \\ \dot{\hat{\boldsymbol{x}}} = \hat{\boldsymbol{G}}(\hat{\boldsymbol{x}}, \boldsymbol{y_1}) \\ \dot{\hat{\boldsymbol{y}}} = \hat{\boldsymbol{H}}(\hat{\boldsymbol{x}}_1, \hat{\boldsymbol{y}}_1) \end{cases} \qquad (2.6)$$

On parle alors de synchronisation par cascade.

La condition nécessaire pour obtenir la synchronisation est que tous les exposants conditionnels de Lyapunov (CLE) (annexe) du sous-système $\hat{G}$ soient négatifs. Les CLE représentent les exposants de Lyapunov dans le cas d'un système non autonome. Dans notre cas, le sous-système $\hat{G}$ est piloté par le signal $y_1(t)$. Les CLE caractérisent la stabilité du sous-système non autonome $\hat{G}$ et s'ils sont tous négatifs, les signaux $\widehat{x_1}\ (t) et\ x_1(t)$ se synchronisent et :

$$\lim_{t\to\infty} |\widehat{x_1}(t) - x_1(t)| = 0 \qquad (2.7)$$

La condition exposée précédemment est une condition nécessaire et suffisante pour obtenir la stabilité locale. En effet, la synchronisation du sous-système $\hat{H}$ dépend aussi du choix des paramètres statiques (par exemple les valeurs des composants d'un circuit) et des conditions initiales des variables dynamiques du système maître (x(t = 0), y(t = 0)) et du système esclave $(\hat{x}(t=0), \hat{y}(t=0))$. La difficulté réside dans les conditions initiales qui dans la pratique ne sont pas contrôlables.

### 2.3.4. Synchronisation complète: [15]

On considère un système maître représenté par les équations suivantes :

$$\dot{x} = f(t,x), \qquad y = h(x) \quad x \in \mathbb{R}^n, \qquad h: \mathbb{R}^n \to \mathbb{R}^n$$

Et un système esclave donné par :

$$\dot{\hat{x}} = \hat{f}(t,\hat{x},y), \qquad \hat{y} = \hat{h}(\hat{x}) \quad \hat{x} \in \mathbb{R}^p, \qquad h: \mathbb{R}^p \to \mathbb{R}^q$$

Où $(x,\hat{x})$ sont les états des systèmes et $(y,\hat{y})$ sont les sorties.

Soit $\varphi$ une fonction continue, qui décrit la relation entre le maître et l'esclave lors de la synchronisation :

$$\hat{y} = \varphi(y), \qquad \varphi: \mathbb{R}^m \to \mathbb{R}^q$$

La synchronisation est dite complète si $\hat{x}(t) = x(t)$.

Ce qui implique que ; $m = q$ et $\varphi$ est une identité

Si $\hat{f} = f$, la relation devient une synchronisation complète identique.

Si $\hat{f} \neq f$ c'est une synchronisation complète non identique.

La synchronisation complète est donc une coïncidence complète entre variables d'état des deux systèmes synchronisés. Les méthodes de synchronisation complète sont typiquement associées avec la synchronisation des systèmes identiques.

La majorité des concepts de synchronisation complète utilise un schéma de rétroaction et sont considérées comme étant bidirectionnels, car les deux systèmes sont à la fois source et destination.

### 2.3.5. Synchronisation généralisée :

La synchronisation généralisée est considérée comme une généralisation de la synchronisation complète pour synchroniser des systèmes chaotiques de modèle différent. Elle se manifeste par une relation fonctionnelle entre deux systèmes chaotiques couplés.

### 2.3.6. Synchronisation par contre-réaction (couplage diffusif): [16]

Les recherches qui ont suivies celles de Pecora et Carroll ont montré que la synchronisation par remplacement complet n'était qu'un cas très particulier de la méthode que nous allons maintenant présenter dans ce paragraphe.

Dans la globalité on garde les mêmes notations pour le système chaotique étudié mais sans le séparer en sous-systèmes. Pour que la synchronisation ait lieu, on prend au moins un des signaux $x_{ri}$ et on ajoute un facteur amortissant qui a pour valeur la différence $x_{ci}$ - $x_{ri}$ au système de réponse. On a alors :

$$\dot{\boldsymbol{x}}_{\boldsymbol{c}} = \boldsymbol{f}(\boldsymbol{x}_{\boldsymbol{C}}) \; \boldsymbol{et} \; \dot{\boldsymbol{x}}_{\boldsymbol{r}} = \boldsymbol{f}(\boldsymbol{x}_{\boldsymbol{r}}) + \; \mathcal{C} \; \boldsymbol{e}(\boldsymbol{x}_{\boldsymbol{C}} \; - \boldsymbol{x}_{\boldsymbol{r}}) \qquad (2.8)$$

Où $\mathcal{C}$ est le facteur de couplage et $e$ est une fonction linéaire qui définit la combinaison linéaire des signaux qui seront utilisés pour l'amortissement. Puis, similairement, on pose $\boldsymbol{\varepsilon} = \; \boldsymbol{x}_{\boldsymbol{C}} - \boldsymbol{x}_{\boldsymbol{r}}$ et on a :

$$\dot{\boldsymbol{\varepsilon}} = \boldsymbol{f}(\boldsymbol{x}_{\boldsymbol{C}}) - \; \boldsymbol{f}(\boldsymbol{x}_{\boldsymbol{r}}) - \mathcal{C} \; \boldsymbol{e}(\boldsymbol{x}_{\boldsymbol{C}} - \boldsymbol{x}_{\boldsymbol{r}}) \approx \left(\boldsymbol{J}_{\boldsymbol{f}} - \mathcal{C}.\boldsymbol{e}\right)\boldsymbol{\varepsilon} \qquad (2.9)$$

Pour avoir la stabilité asymptotique, on calcule les exposants de Lyapunov correspondants à l'équation variationnelle sur $(J_f - \mathcal{C}.e)$ en fonction de $\mathcal{C}$ et on choisit ce facteur de manière à avoir les exposants les plus négatifs possibles. Si le facteur de couplage tend vers $+\infty$ alors on se retrouve dans le cas de la synchronisation par remplacement complet car la matrice $e$ remplacera dans $f$ toutes les $x_{ri}$ par $x_{ci}$ Mais les exposants de Lyapunov des signaux utilisés ne seront pas forcement négatifs dans le cas limite.

- Montage de synchronisation :

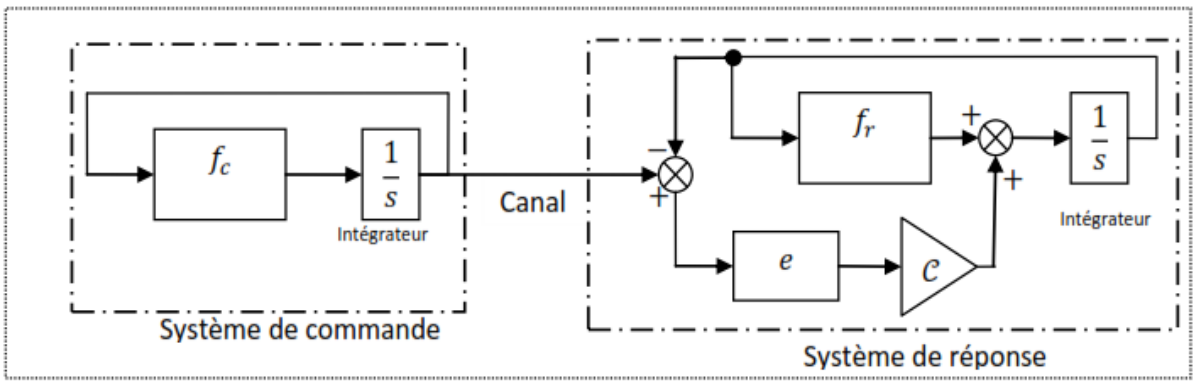



**Figure.29 :** *Synchronisation par contre-réaction.*

*Remarque* : le montage de synchronisation est unidirectionnel (composé d'un système maitre qui commande un système esclave).On peut aussi réaliser des montages de synchronisation bidirectionnels à couplage en ajoutant $\mathcal{C}'\ e'(x_r - x_C)$ à l'équation différentielle du système $f_c$ dans ce cas le résultat est similaire mais les exposants dépendront de deux variables.

Si on a $\mathcal{C}' = \mathcal{C}\ et\ e' = e$ alors les exposants de Lyapunov sont les mêmes que ceux du couplage unidirectionnel mais avec un facteur $2\mathcal{C}$.

### 2.3.7. Synchronisation impulsive [17] :

Dans un schéma de transmission usuel, un des états du système dynamique est transmis afin de réaliser la synchronisation par le récepteur. On propose la synchronisation impulsive (***figure.30***), dans le but de réduire la redondance du signal transmis.

Le contrôle impulsif d'un système signifie qu'à des moments choisis, les états du système changent soudainement.

Dans ce schéma de synchronisation, on considère un système maître de la forme générale suivante :

$$\dot{\boldsymbol{x}}(\boldsymbol{t}) = \boldsymbol{f}(\boldsymbol{x}(\boldsymbol{t}))$$

On définit un signal impulsif qui consiste en une suite d'instants discrets auxquels un signal $y(t)=Cx(t)$ est envoyé par le système maître au système esclave, dont les variables d'état subissent un saut et un changement d'état.

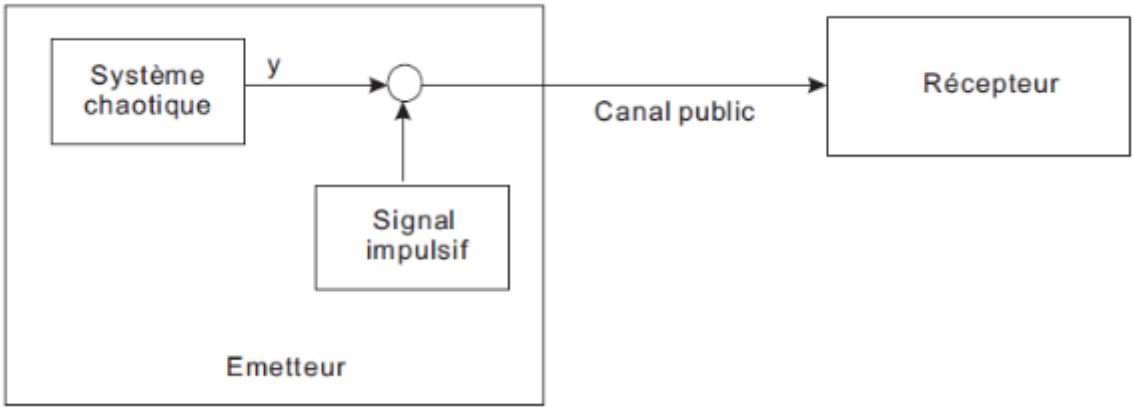



**Figure.30 :** *Synchronisation impulsive.*

### 2.3.8. La synchronisation Lag :

Elle est définie pour le cas où $\hat{\boldsymbol{x}}(\mathbf{t}) \approx \boldsymbol{x}(\boldsymbol{t}-\boldsymbol{\tau})$, $\tau$ est un nombre positif très petit.

### 2.3.9. La synchronisation anticipée :

Comme dans le cas de la synchronisation Lag, la relation entre les variable d'état des systèmes maître et esclave est donnée par :

$$\hat{\boldsymbol{x}}(\mathbf{t}) \approx \boldsymbol{x}(\boldsymbol{t}+\boldsymbol{\tau}), \quad \boldsymbol{\tau} > 0 \qquad (2.10)$$

### 2.3.10. La synchronisation de phase : [15]

Soient $\varphi_1 \; et \; \varphi_2$ les phases des systèmes, maître et esclave respectivement. La synchronisation de phase est réalisée si pour deux nombres entiers $m$ et $n$ il existe un nombre positif très petit ε tel que :

$$|\boldsymbol{m\varphi_1 - n\varphi_2}| < \boldsymbol{\varepsilon} \qquad (2.11)$$

Le phénomène de synchronisation de phase est totalement différent de ceux présentés précédemment. Généralement, lorsque la synchronisation chaotique est obtenue, les exposants de Lyapunov du système esclave sont tous négatifs. Donc le système esclave est un système non chaotique avec une sortie chaotique. Cependant, dans le cas de la synchronisation de phase, les exposants de Lyapunov peuvent prendre des valeurs positives.

## 2.4. Transmission basée sur la synchronisation des systèmes chaotiques:[18]

Dans cette partie du chapitre, on s'intéresse aux techniques de transmission sécurisée d'informations qui reposent sur le principe de synchronisation chaotique. Le point commun constaté dans la majorité des techniques développées dans la littérature est l'utilisation de la configuration maître-esclave pour laquelle on dispose d'un émetteur chaotique (système maître) qui génère le signal du texte chiffré transmis dans le canal de communication vers un système récepteur (système esclave) qui a pour objectif de se synchroniser avec le système maître et de restaurer le signal d'information.

Les signaux chaotiques peuvent être utilisés pour la transmission de l'information principalement dans deux objectifs. Le premier objectif est de protéger l'information transmise et dans ce cas les applications réalisées sont en compétition avec les méthodes de cryptographie classiques. Un deuxième objectif est d'étaler le signal informationnel avec tous les avantages des techniques à étalement de spectre. Dans ce deuxième cas, les méthodes développées doivent être comparées aux systèmes classiques à étalement de spectre.

Si on regarde du point de vue de la structure d'un tel système de transmission on peut définir deux approches. La première, représentée dans la ***figure.31*** remplace le signal porteur sinusoïdal par un modulateur chaotique contrôlé d'une manière quelconque par le signal informationnel. Cette solution a l'avantage d'être très simple à implémenter, mais par contre nécessite un système chaotique avec des contraintes fortes sur les paramètres intrinsèques et en plus celui-ci doit travailler à des hautes fréquences. En pratique, il est difficile de trouver des circuits permettant un tel fonctionnement, pour le moment cette solution est surtout considérée dans un cadre théorique.

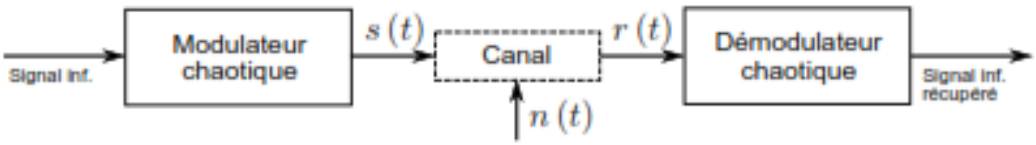



**Figure.31 :** *Modulation directe du signal informationnel par une porteuse haute fréquence chaotique*

Une deuxième solution est de moduler le signal informationnel par celui chaotique en bande de base, et après d'appliquer une transposition en haute-fréquence par l'intermédiaire d'une porteuse sinusoïdale. Ce schéma est présenté dans la ***figure.32***. Son avantage principal consiste dans une simplification importante du modulateur chaotique, mais avec une complexité générale du système plus importante [19].

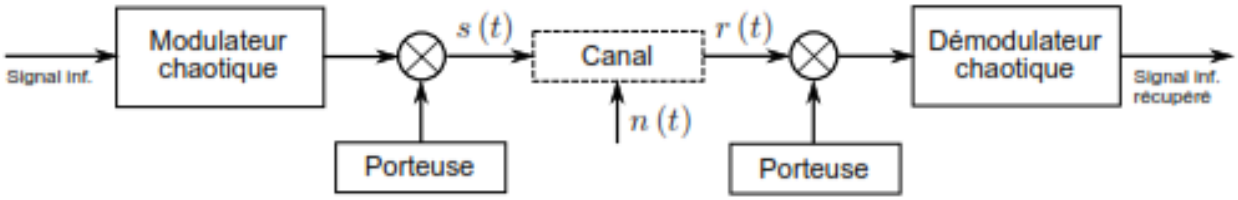



**Figure.32 :** *Modulation en bande de base du signal informationnel par le signal chaotique, combinée avec une mise sur porteuse classique.*

Parmi les techniques de communications traditionnelles à base du chaos, on cite : le masquage chaotique, la modulation paramétrique, la commutation chaotique, le cryptage par injection (inclusion), la transmission à deux voies et le cryptage combiné.

### 2.4.1. Masquage chaotique (cryptage par addition) : [13]

Le masquage chaotique est la technique de transmission d'information la plus simple et la plus élémentaire. L'idée de base consiste à additionner le message (analogique ou numérique) *m(t)* à transmettre à un signal chaotique *x(t)* en bande de base, la ***figure.33*** illustre le principe de base de cette technique. Au niveau du récepteur, un système chaotique identique à celui de l'émetteur essaie de se synchroniser avec le signal chaotique reçu *y(t).* Du point de vue de la synchronisation des deux systèmes chaotiques, le message *m(t)* est pris comme un élément perturbateur du signal *x(t).* En considérant un canal de propagation idéal non bruité et sans trajets multiples, le signal chaotique reçu s'écrit :

$$\boldsymbol{y(t) = m(t) + x(t)} \qquad (2.12)$$

En utilisant par exemple le principe de Pecora et Carroll pour réaliser la synchronisation et en supposant que les deux systèmes chaotiques soient dans des configurations favorables (problèmes des conditions initiales), l'erreur de synchronisation ne dépend que de *m(t).*

Donc, si l'amplitude du message *m(t)* est suffisamment faible par rapport au signal chaotique *x(t),* le signal contenant l'information ne modifie que modérément *y(t)* et la synchronisation s'effectue correctement. Ensuite, pour démoduler le message, il suffit de soustraire le signal à l'entrée du système chaotique *(y(t))* avec celui en sortie de ce même dispositif *(y'(t))* :

$$\boldsymbol{m'(t) = y(t) - y'(t)} \qquad (2.13)$$

Si la synchronisation est parfaite alors : *y'(t) = x(t) et m'(t) = m(t)*

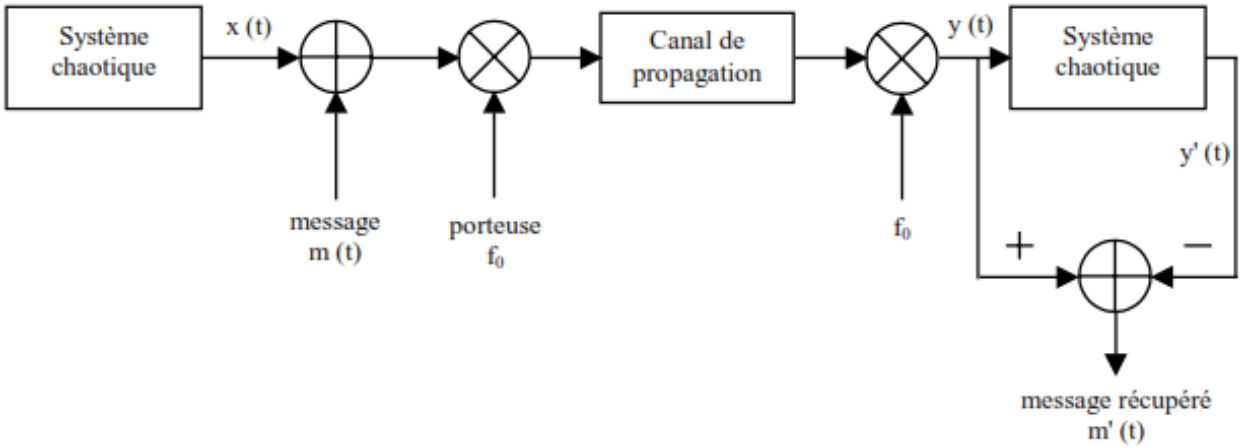



**Figure.33 :** *Architecture d'un système utilisant le masquage chaotique.*

En considérant un canal de propagation réel qui prend en compte la non-linéarité de l'amplificateur de puissance, les trajets multiples…, cette méthode ne se révèle pas très efficace car elle est sensible au bruit. Au cours de la démodulation, le bruit additif ne se distingue pas du message et il est nécessaire de pouvoir l'éliminer surtout si le niveau de bruit n'est pas suffisamment faible par rapport à celui du message.

Il existe bien évidemment d'autres méthodes plus efficaces pour crypter que de faire une simple somme de signaux, parmi celles-ci on note notamment celles de la modulation. Il existe différentes techniques de modulation pour le chaos comme par exemple la modulation de fréquence chaotique (chaotic frequency modulation : CFM), modulation par étalement de spectre (spread spectrum communication) ou encore la modulation par commutation de chaos (chaos shift keying : CSK).

### 2.4.2 Cryptage par commutation (Chaos Shift Keying, 'CSK'): [23]

Dans cette méthode, l'information est binaire et le principe consiste à transmettre un signal chaotique durant la transmission d'un bit "0" et un autre signal chaotique différent du premier pour un bit "1". Les deux signaux chaotiques peuvent soit provenir de deux systèmes différents, soit de deux systèmes possédant la même structure mais avec des paramètres modifiés. De cette façon, chaque bit est représenté par un attracteur étrange distinct.

Le schéma de principe de cette technique est représenté par la ***figure.34***. Au niveau de l'émetteur, on dispose de deux oscillateurs générant les signaux chaotiques A(t) et B(t). Le signal d'information de type binaire M(t) est utilisé pour commuter entre A(t) encodant le bit 1 et B(t) encodant le bit 0. Le signal résultant X(t) est transmis à travers le canal de transmission vers le système récepteur constitué de deux systèmes esclaves. Le premier système esclave

synchronise exclusivement avec le premier oscillateur (correspondant au signal chaotique A(t)) de telle façon que le bit 1 est détecté par la convergence de l'erreur de synchronisation vers zéro et par conséquent le signal d'information peut être enfin restauré à la fin du processus de détection.

- Schéma représentatif d'un système de transmission CSK :

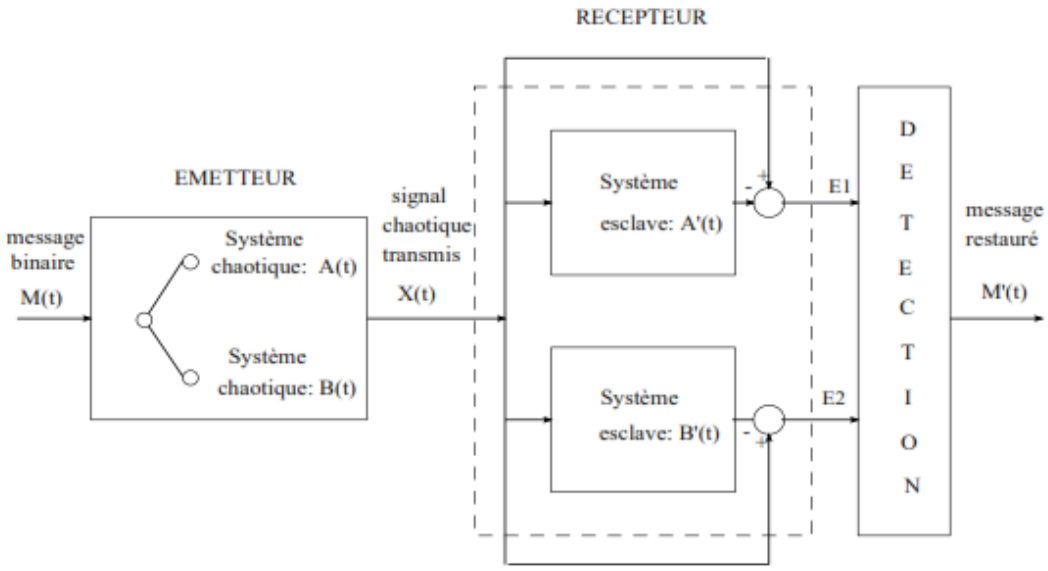



**Figure.34** : *Architecture d'un système de transmission CSK*.

Comparée à la technique de masquage chaotique, la commutation chaotique présente relativement plus de robustesse au bruit de canal ; néanmoins, les crypto systèmes utilisant cette technique possèdent une faible vitesse de transmission car à chaque changement de bit on doit tenir compte du temps de convergence nécessaire pour la mise en place de la synchronisation. Cette méthode est caractérisée par un faible niveau de sécurité puisqu'à chaque changement du niveau binaire, on peut observer la modification du signal du texte chiffré, surtout lorsque les deux oscillateurs utilisés au niveau de l'émetteur possèdent deux attracteurs très différents.

### 2.4.3. Cryptage par inclusion (injection) :

Cette technique de cryptage consiste à injecter le message dans la dynamique de l'émetteur chaotique, sans toutefois réaliser une modulation de paramètre. Le récepteur a pour but de se synchroniser avec l'émetteur et de reconstruire le signal d'information. Conformément au constat de Nijmeijer et Mareels [24]. La restauration de l'information se fait principalement par deux techniques, reposant soit sur les observateurs à entrées inconnues, soit sur l'inversion du système émetteur [25].

Le schéma de principe de cette technique est représenté dans la **figure.*35***.

- Schéma de principe de la technique de cryptage par injection :

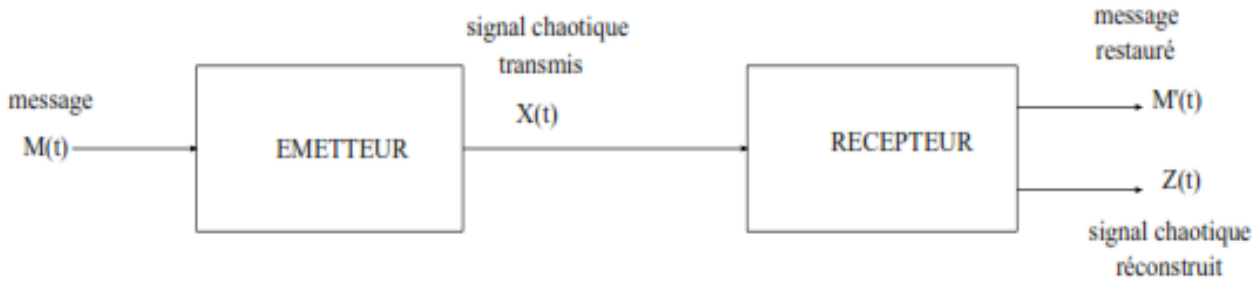



**Figure.35 :** *Schéma représentatif de la technique de cryptage par injection (inclusion).*

Cette technique est valable pour transmettre un message de nature binaire ou analogique, mais la puissance de ce dernier doit être suffisamment petite pour ne pas détériorer le comportement chaotique du système maître. Cette technique présente un niveau de sécurité nettement élevée par rapport aux techniques précédentes puisque le signal d'information est masqué dans la dynamique du système maître et que le signal chaotique disponible dans le canal public ne porte pas l'information d'une manière directe comme dans le cas de la technique de masquage chaotique.

### 2.4.4. Transmission à deux voies :

Le schéma de principe de la transmission à deux voies est illustré dans la ***figure.36***.

L'idée de base consiste à séparer les tâches de synchronisation et de cryptage en utilisant deux voies de communication. L'émetteur chaotique génère un signal chaotique Y(t) transmis dans un premier canal de communication (Canal 1) vers le récepteur qui doit se synchroniser avec le système maître. L'émetteur génère également un autre signal chaotique X(t) utilisé par une fonction de cryptage qui produit le signal du texte chiffré C(t) transmis dans un deuxième canal de transmission (Canal 2).

- Schéma de principe de la transmission à deux voies :

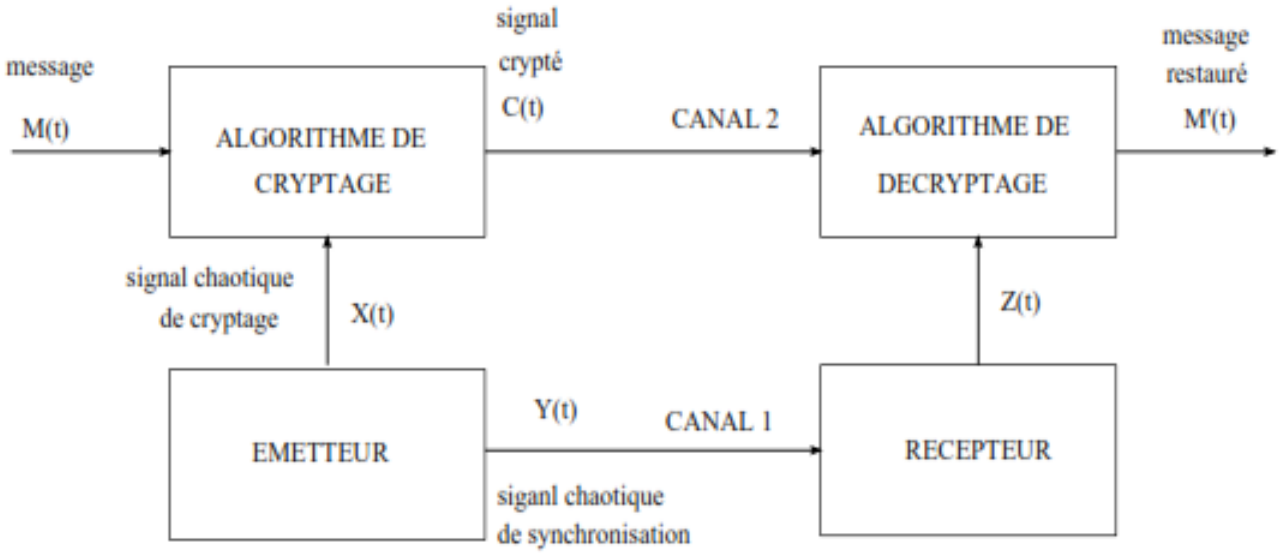



**Figure.36 :** *Schéma représentatif de la technique de transmission à deux voies.*

Grâce à cette indépendance entre les tâches de synchronisation et de cryptage, il n'y a pas de contrainte à imposer sur l'amplitude du signal d'information puisque dans ce cas, ce dernier n'agit ni sur la dynamique de l'émetteur chaotique ni sur le signal transmis Y(t) contrairement aux autres techniques. De plus, le signal d'information n'a aucune influence sur l'opération de synchronisation qui s'établit d'une façon idéale, ce qui permet de garantir une meilleure qualité de l'information récupérée au niveau du récepteur.

D'un autre côté, cette méthode garantit un meilleur niveau de sécurité par rapport aux autres techniques puisque la séparation entre les opérations de cryptage et de synchronisation permet de concevoir une fonction de cryptage de plus en plus complexe sans se soucier de détériorer l'aspect chaotique de l'émetteur ou de perdre la synchronisation entre les systèmes maître et esclave. Cependant, cette technique présente des mauvaises performances en présence du bruit de transmission puisque l'effet du bruit est doublé en agissant à la fois sur le signal transmis Y(t) dans la première voie et également sur le signal du texte chiffré C(t) présent dans la deuxième voie de transmission.

### 2.4.5. Cryptage combiné :

Cette technique est un mixage entre les systèmes cryptographiques classiques et les systèmes de communication reposant sur le principe de synchronisation des systèmes chaotiques. Le principe de cette méthode est illustré dans la **figure.37**. La fonction de cryptage utilise le signal d'information M(t) et un signal chaotique de cryptage X(t) généré par l'émetteur chaotique pour produire le signal crypté C(t) réinjecté dans la dynamique de l'émetteur chaotique. Au niveau de la réception, un système esclave se synchronise avec le système maître et génère les signaux C'(t) et Z(t) qui représentent les estimations respectives des signaux C(t) et X(t). Enfin, un algorithme de décryptage utilise les signaux obtenus pour restaurer le signal d'information en générant le signal M'(t).

- Schéma représentatif de la technique de cryptage combiné :

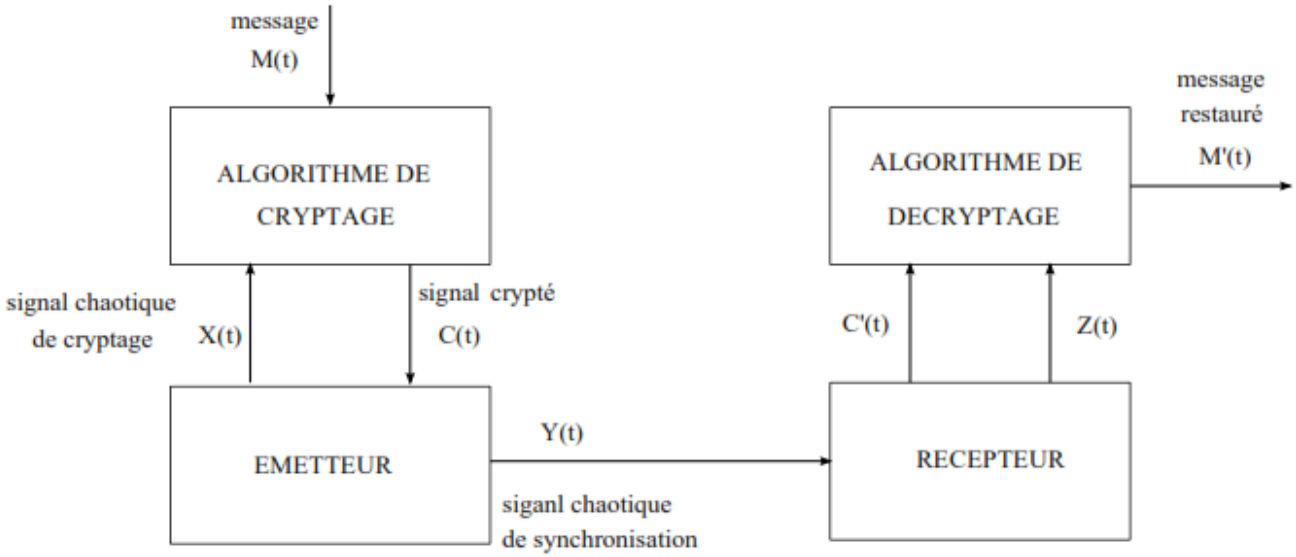



**Figure.37 :** *Schéma représentatif de la technique de cryptage combiné.*

Cette technique présente un bon niveau de sécurité grâce à la complexité de l'algorithme de cryptage, et plus de robustesse aux attaques cryptographiques.

### 2.4.6. Cryptage par modulation paramétriques :

Cette technique utilise le message contenant l'information pour moduler un paramètre de l'émetteur chaotique. Un contrôleur adaptatif est chargé de maintenir la synchronisation au niveau du récepteur, tout en suivant les changements du paramètre modulé. Le schéma correspondant est présenté à la ***figure.38***. Au niveau de l'émetteur, le fait de moduler un/plusieurs paramètre(s) impose à la trajectoire de changer continûment d'attracteur, et de ce fait, le signal transmis est plus complexe qu'un signal chaotique habituel. Cependant, la façon d'injecter le message et donc la fonction de modulation des paramètres ne doivent pas supprimer le caractère chaotique du signal envoyé au récepteur. Il est important de souligner

que cette technique exploite pleinement les qualités des systèmes chaotiques. Elle n'a pas d'équivalent parmi les systèmes de communications classiques. Cependant, le cryptage par modulation s'est avéré sensible à certaines attaques.

- Schéma de principe :

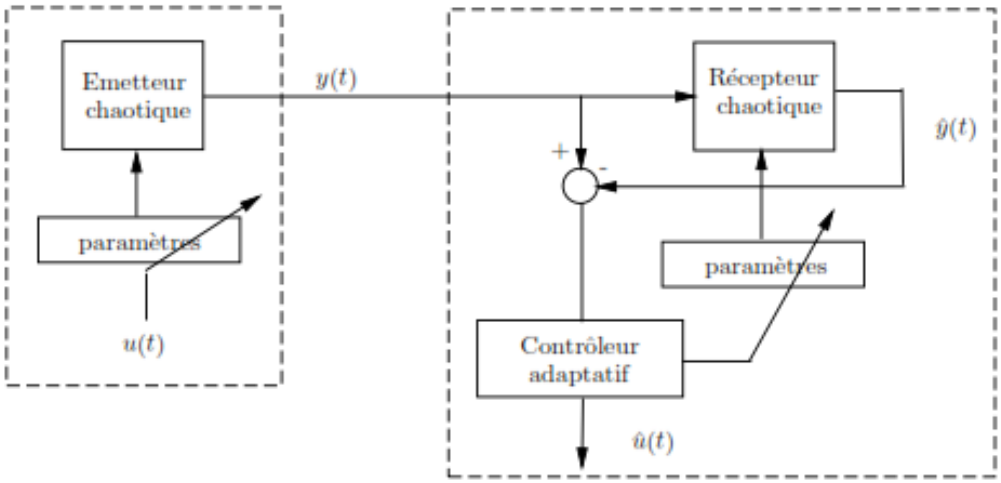



**Figure.38 :** *Principe de cryptage par modulation de paramètres.*

## 2.5. Conclusion :

Ce chapitre a comme objectif de faire le lien entre les systèmes dynamiques chaotiques et le domaine des télécommunications. Dans la première partie, nous avons introduit la notion de communication sécurisée à base du chaos, ainsi que les différentes méthodes de synchronisation et schémas de principe. Ensuite, nous nous sommes intéressés aux techniques de transmission sécurisée d'informations qui reposent sur le principe de synchronisation chaotique. Enfin, nous avons passé en revue les techniques de cryptage à base du chaos. Le chapitre suivant sera consacré à la proposition et à la simulation sous Matlab d'un nouveau schéma de transmission de données à base de la synchronisation de deux systèmes chaotiques.

## 3.1. Introduction :

Parallèlement aux grandes avancées réalisées dans la théorie du chaos, la synchronisation chaotique a reçu beaucoup d'attention ces dernières années. Les perspectives d'utilisation du chaos dans diverses applications, notamment en télécommunication, et plus particulièrement dans les communications sécurisées, ont motivé les chercheurs à étudier d'avantage la synchronisation chaotique. Grâce à son grand potentiel dans les applications technologiques, la génération de l'hyper chaos est récemment devenu un thème central de recherche. Il faut savoir que la plupart des travaux sur la synchronisation des systèmes chaotiques s'est consacrée aux systèmes dynamiques à temps continu. Mais récemment, plusieurs études ont adressé la synchronisation des systèmes à temps discret pour leurs divers avantages. Dans le domaine des communications privées, les clés d'un système de cryptage chaotique sont souvent les paramètres du système chaotique. La possibilité de reconstituer les clés d'un crypto système chaotique est équivalente à la possibilité d'identifier les paramètres du système chaotique. Par conséquent, un système de cryptage chaotique sûr doit être conçu de façon à ce que ses paramètres ne soient pas identifiables. Dans ce chapitre, quelques solutions sont apportées afin de sécuriser la transmission d'information basée sur la synchronisation chaotique. Notre schéma de transmission étudié est constitué de systèmes dynamiques chaotiques à temps discret. L'émetteur est composé d'un système chaotique à temps discret dit Hénon modifié, le message est introduit par inclusion dans l'une des dynamiques de ce système à temps discret. Le récepteur est composé d'un observateur à temps discret, qui permet de reconstruire les états de l'émetteur et récupérer l'information.

La première partie du chapitre présente le principe de la méthode proposée en étudiant l'émetteur et le récepteur du système de transmission. Puis, la seconde partie expose les résultats de simulation.

## 3.2. Description de la chaine de transmission privée :

Dans ce chapitre, on procède à la mise en œuvre d'un système de communication basé sur un système dynamique hyper chaotique discret. Le schéma général du système proposé pour une communication numérique sécurisée est montré dans la **figure 39.**

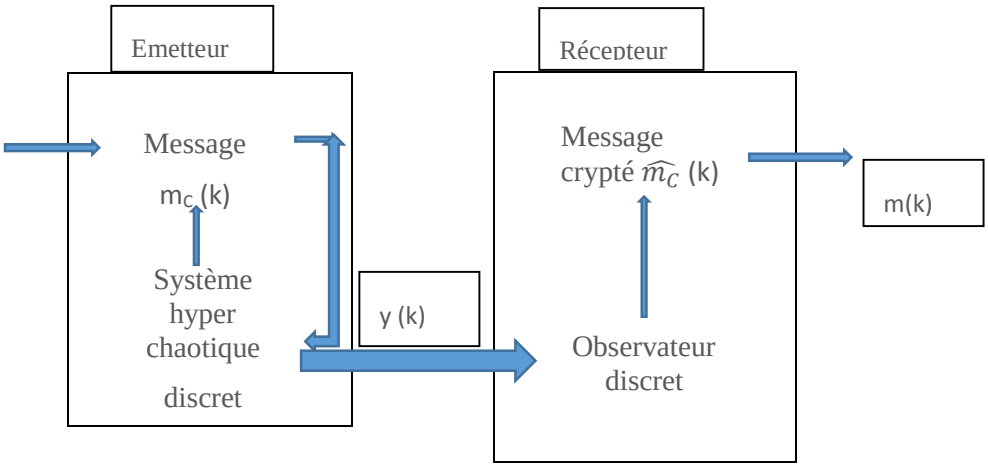



**Figure.39 :** Schéma général du système de transmission.

- Principe de la méthode :
    1. Choix de l'émetteur.
    2. Choix du récepteur.
    3. Mise au point du processus de transmission de l'information.

La méthode utilisée se présente comme suit :

### a) ***Présentation de l'émetteur :***

Le système hyper chaotique à temps discret est le Hénon modifié. Une version simplifiée du système à temps discret que l'on propose est :

$$\begin{cases} x\ (k+1) = a - y^2(k) - bz(k) \\ y(k+1) = x\,(k) \\ z(k+1) = y(k) \\ s(k) = y(k) \end{cases} \quad (3.1)$$

Où $x = [x, y, z]^T \in \mathbb{R}^3$ représente le vecteur d'état.

Le comportement chaotique ainsi que les réponses des états du système (3.1) comme le montrent les **figures 40** et **41**, respectivement, sont obtenus en posant les paramètres :

*a = 1.76 et b = 0.1* et les conditions initiales du système qui sont choisies à l'intérieur du bassin d'attracteur étrange sont : $x_0 = 0.1$ $y_0 = 0.1$ $z_0 = 0.1$.

$s(k) = y(k)$ *est la sortie du système.*

- Attracteur de Hénon-Heiles :

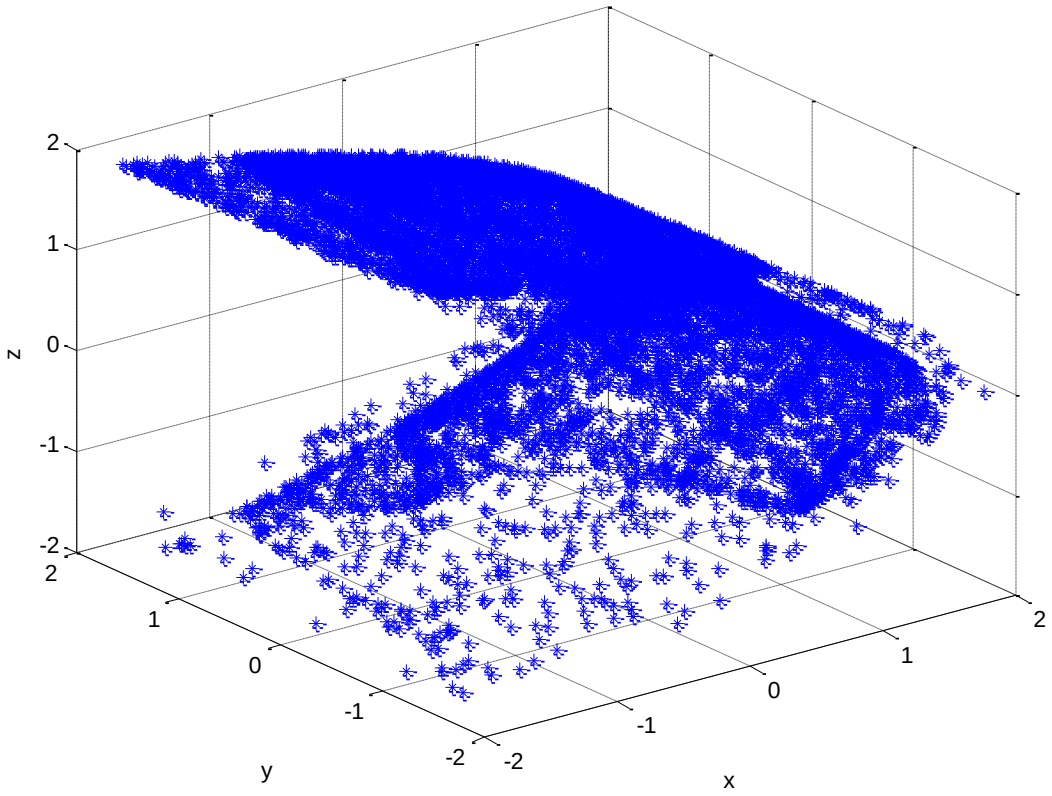



**Figure.40 :** *Attracteur de Hénon-Heiles.*

Dans les communications privées, l'objectif le plus important est d'augmenter la sécurité du système de transmission. Pour ce faire, il est intéressant de modifier le système (3.1).

Notre approche pour inclure le signal du message *m* dans le modèle de Hénon modifié est la méthode par inclusion (décrite au chapitre 2). Cette dernière consiste en l'ajout du message dans l'une des dynamiques du système (3.1).

Comme le montre la **figure.39**, le message *m(k)* à envoyer est introduit dans la troisième dynamique du système (3.1) dans l'intention de garder le comportement chaotique.

Après avoir ajusté les paramètres pour obtenir un comportement chaotique, le signal *m* est ajouté à l'itéré de l'état *z(k)* du système. Ainsi, l'état *z(k)* est modulé en fonction du message *m*. Par contre, le signal transmis au récepteur est l'état *y*, ce qui veut dire que l'on ne transmet pas directement l'état modulé au récepteur. Cela fait une différence considérable entre notre approche et la méthode par addition, dans laquelle le message *m* est ajouté à la sortie de l'émetteur et la somme est transmise directement au récepteur. Il faut noter que l'amplitude et la fréquence du message doivent être choisies de telle manière que l'on ne puisse pas détecter

de variations visibles relatives au message sur la sortie du système. De plus, nous supposons que le message est borné et assez petit afin de préserver le comportement chaotique.

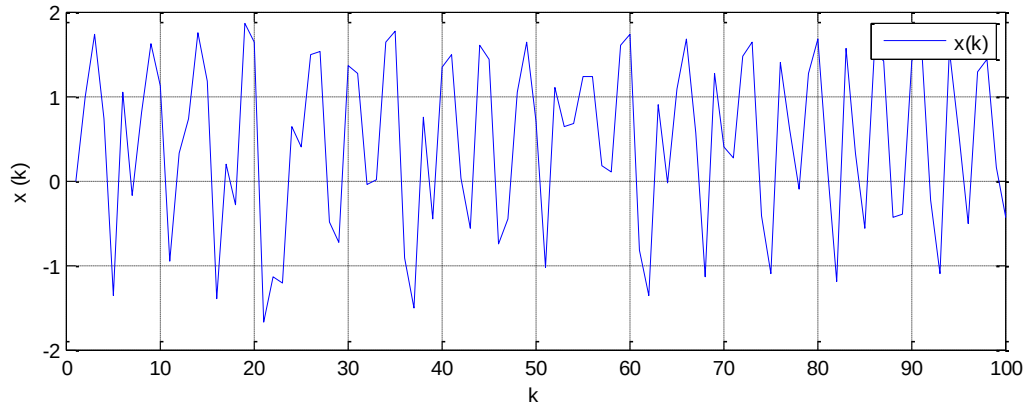
Nous introduisons le message *m* dans la troisième dynamique du système de transmission.

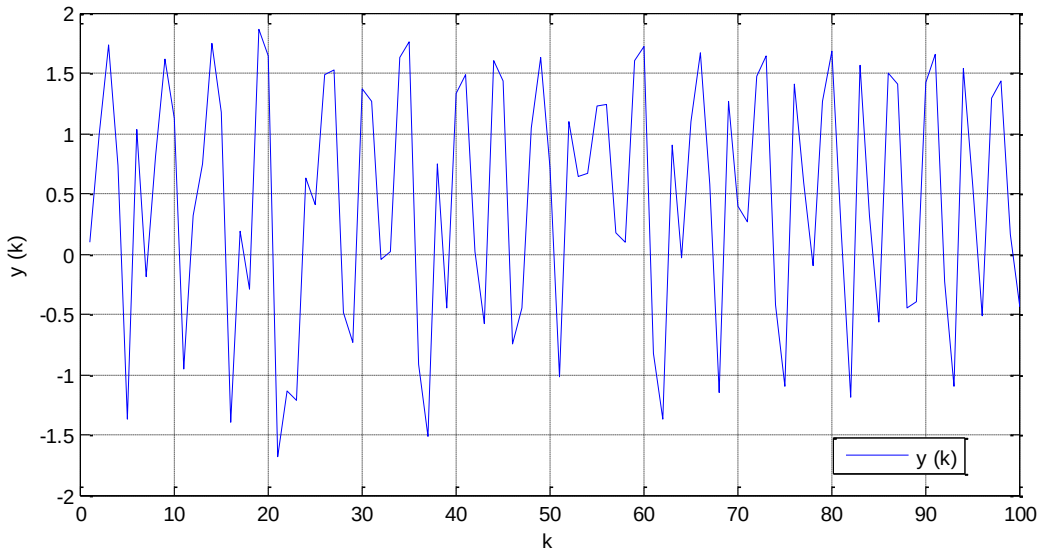
On obtient :

$$\begin{cases} x\ (k+1) = a - y^2(k) - bz(k) \\ y(k+1) = x\,(k) \\ z(k+1) = y(k) + m\,(k) \\ s(k) = y(k) \end{cases} \qquad (3.2)$$

Où $m\ (k)$ est un signal carré d'amplitude = 0.1

- <u>Représentation des états du système de Hénon modifié et du message (Emetteur) :</u>



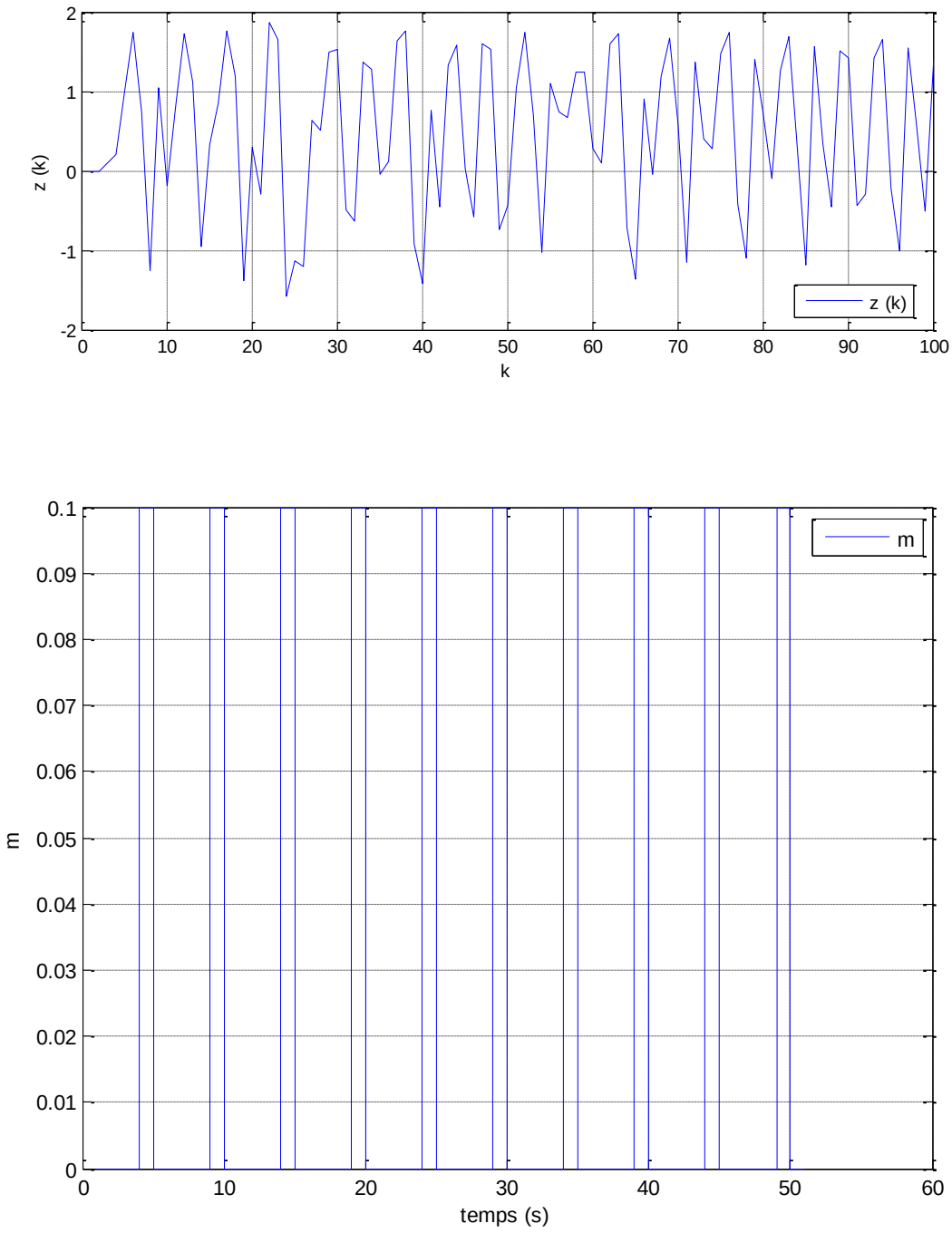



**Figure.41 :** Etats discrets x(k), y(k), z(k) et le message m(k).

### b) ***Présentation du récepteur :***

Dans cette partie, le système (3.2) avec la sortie $s(k) = y(k)$ est pris en considération. Pour la réception, nous concevons un observateur discret retardé en utilisant une période d'échantillonnage *T*. L'observateur est choisi afin de récupérer les états ainsi que le message du système de Hénon modifié. Dans ce qui suit, nous étudions le choix du signal de sortie dans le but de garantir l'observabilité du système.

***b.1) Condition d'observabilité et propriété d'inversion à gauche :***

On considère le système non linéaire suivant :

$$\begin{cases} x(k+1) = f\big(x(k)\big) + p\big(x(k)\big)w(k) \\ s(k) = h(x(k)) \end{cases} \quad (3.3)$$

Où *w(k)* représente une entrée inconnue, qui peut être une perturbation, une erreur ou dans notre cas, un message.

Les champs des vecteurs f, p : $f, p: \mathbb{R}^n \longrightarrow \mathbb{R}\ et\ h: U \subset \mathbb{R}^n \longrightarrow \mathbb{R}$ sont supposés réels. Le message doit être discret. La sortie du système (3.3) est transmise au récepteur, ce qui devrait générer un vecteur de sortie qui converge asymptotiquement vers le vecteur d'entrée de l'émetteur. Ceci constitue le problème de l'inversion à gauche. Il est possible de concevoir un observateur discret retardé pour le système (3.3). Pour cela, il est nécessaire de vérifier certaines conditions qui sont [26] :

1) Les états ainsi que la perturbation inconnue sont bornés.
2) L'espace vectoriel $span\,\{dh, d(f \circ h), \dots, d(f^{n-1} \circ h)\}$ est de rang n.
3) $O.p = ((dh)^T, (d(f \circ h)^T), \dots, (d(f^{n-1} \circ h))^T)^T.p = (0, \dots, \theta)^T$

Où $\theta$ est une fonction différente de zéro presque partout dans $U \subset \mathbb{R}^n \longrightarrow \mathbb{R}$

La condition 3) est appelée *condition d'observabilité,* elle garantit la propriété d'inversion à gauche, i.e., la possibilité de récupérer tous les états, et le message *w(k)* à partir de *s(k)* et ses itérations [27].

Dans ce qui suit, nous étudions le choix du signal de sortie dans le but de garantir l'observabilité du système. Ensuite, nous expliquons que l'inclusion du message vérifie l'inversion à gauche du système (3.2).

***b.2) L'observateur discret retardé proposé :***

On considère le système (3.2) qui peut être réécrit sous la forme du système (3.3). Dans ce qui suit, nous vérifions les hypothèses 1), 2) et 3).

- Tous les états, ainsi que le message *m* du système (3.2) sont bornés. Ceci nous assure que l'hypothèse 1) est vérifiée.

  - Observabilité du système (3.2) :

On étudie la faible observabilité locale de (3.2). On calcule la matrice d'observabilité dans le voisinage du point d'équilibre (0, 0, 0) du système (3.2) ci-dessous :

On a $h = [0\ y\ 0]\ et\ dh = [0\ 1\ 0], ensuite$:

$$O = \begin{pmatrix} dh \\ d(f \circ h) \\ d(f^2 \circ h) \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & -2x_2 & -b \end{pmatrix}$$

Avec ces valeurs, on trouve que le *rang (O) = 3 = n,* donc le système (3.2) est localement faiblement observable. L'hypothèse 2) est vérifiée.

Par conséquent, l'observateur donné ci-dessous permet de reconstituer tous les états du système (3.2). Ceci explique le choix de la sortie s.

- Condition d'observabilité du système (3.2) :

Dans notre cas, nous avons :

$$p = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

On calcule maintenant *O.p :*

$$O.p = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & -2x_2 & -b \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \theta = -b \end{pmatrix}$$

On note que la valeur de $\theta$ n'est pas égale à zéro. Ainsi, la condition d'observabilité donnée dans l'hypothèse 3) est vérifiée. Par conséquent, l'observateur donné ci-dessous permet de reconstituer le message *m* transmis du système (3.2). Ceci explique le choix de la méthode d'insertion du message *m* dans la troisième dynamique du système (3.2).

Le système (3.2) vérifie les hypothèses 1), 2), et 3). Nous allons donc concevoir un observateur discret retardé étape par étape qui fonctionne avec un pas d'échantillonnage *T,* (la reconstruction des états et du message est faite étape par étape, d'où son appellation.)

La première étape consiste à appliquer un pas de retard (n-1) sur la sortie et ainsi reconstruire le premier état du système de départ. Durant la seconde étape on applique deux pas de retard (n-2) sur la sortie et un pas de retard (n-1) sur l'état venant d'être reconstruit afin de reconstruire le second état. L'application de retards est faite sur tous les états jusqu'à la dernière information contenant l'entrée du système de départ. Chaque état reconstruit à l'itération n contribue à la reconstruction du prochain état à l'itération n-1. ([28], [29]).

La conception de l'observateur se fait comme suit :

- Reconstitutions de l'état $\hat{x}(k)$ :

A partir du système (3.2), on a :

$$\hat{y}(k+1) = \hat{x}(k)$$

En appliquant un pas de retard sur la sortie, on déduit l'état $\hat{x}(k)$ comme suit :

$$\hat{x}(k-1) = s(k) = x_0(k-1) \qquad (3.4)$$

- Reconstitution de l'état $\hat{z}(k)$ :

A partir du système (3.2), on a aussi :

$$\hat{z}(k) = \frac{a - \hat{x}(k+1) - \hat{y}^2(k)}{b} \; en\ sortie.$$

On applique maintenant, deux pas de retard sur la sortie, et en utilisant l'équation précédente on obtient l'état $\hat{z}$ comme suit :

$$\hat{z}(k-2) = \frac{a - s(k) - s^2(k-2)}{b} = z_0(k-2) \qquad (3.5)$$

- Reconstitution du message $\widehat{m}(k)$ :

A partir du système (3.2), on a :

$$\widehat{m}(k) = \hat{z}(k+1) - \hat{y}(k)$$

En utilisant les équations (3.4), (3.5), (3.6) et en appliquant trois pas de retard, on obtient :

$$\widehat{m}(k-3) = \frac{a - s(k) - s^2(k-2)}{b} - s(k-3) = m_0(k-3) \qquad (3.6)$$

Par la suite, les équations de l'observateur sont données par les équations (3.4), (3.5) et (3.6)

$$\begin{cases} \hat{x}(k-1) = s(k) \\ \hat{z}(k-2) = \dfrac{a - s(k) - s^2(k-2)}{b} \\ \widehat{m}(k-3) = \dfrac{a - s(k) - s^2(k-2)}{b} - s(k-3) \end{cases} \qquad (3.7)$$

- Représentation des états après synchronisation :

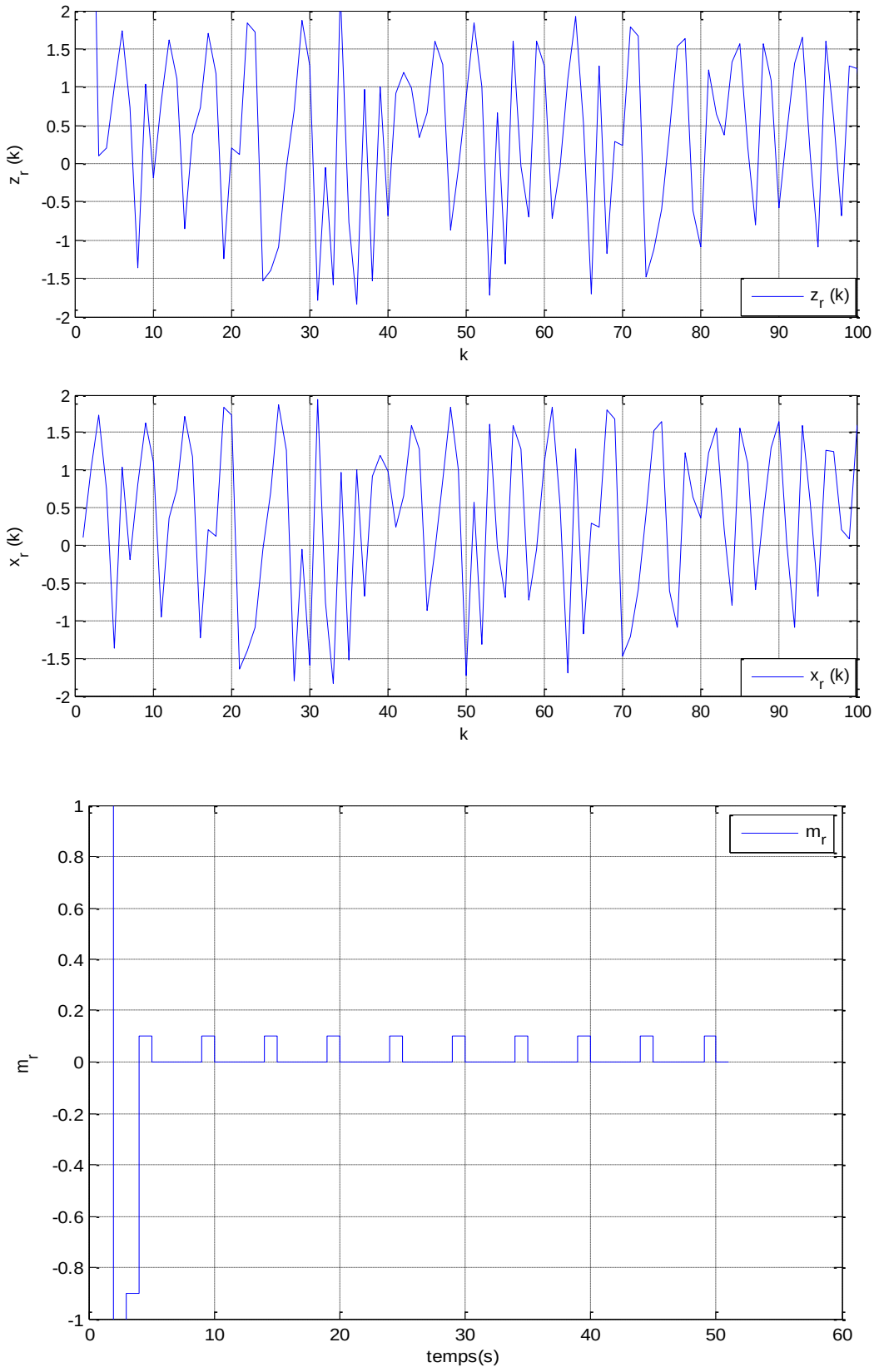
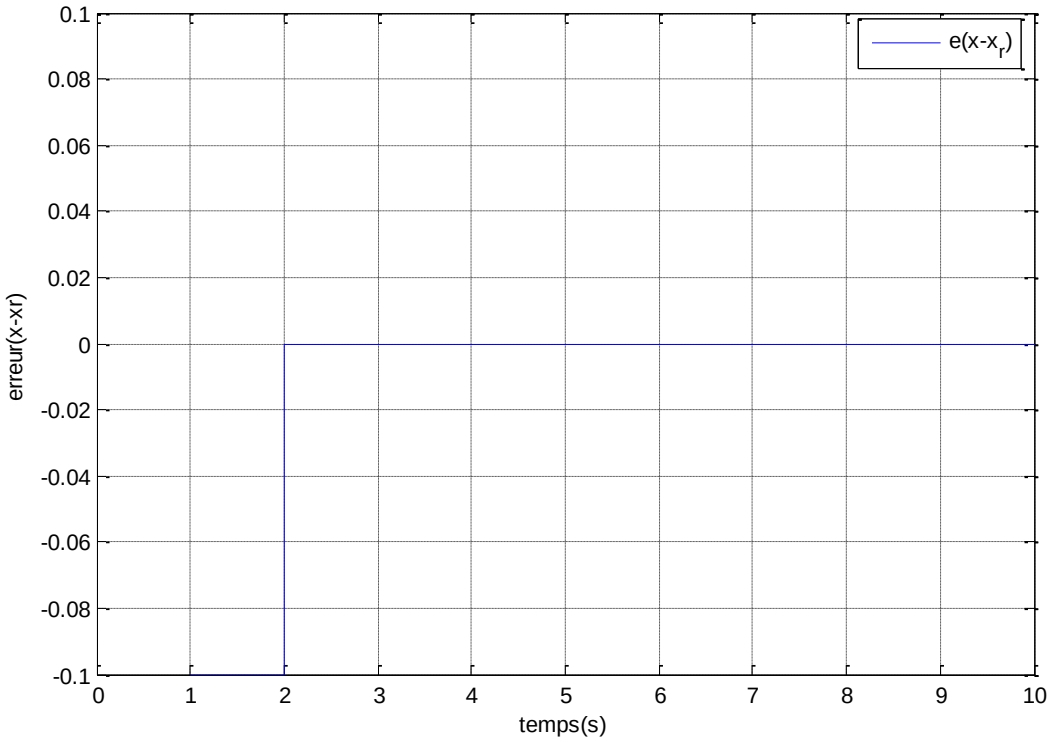
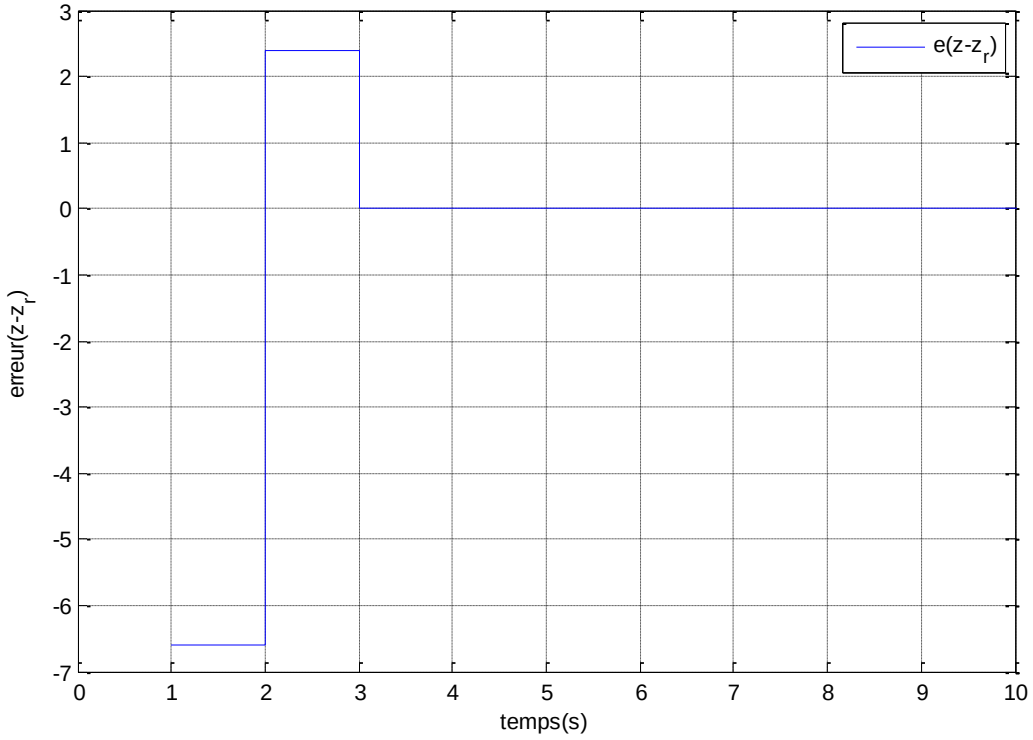



**Figure.42 :** Les états xr, zr et le message mr après synchronisation

- Représentation des erreurs de synchronisation :



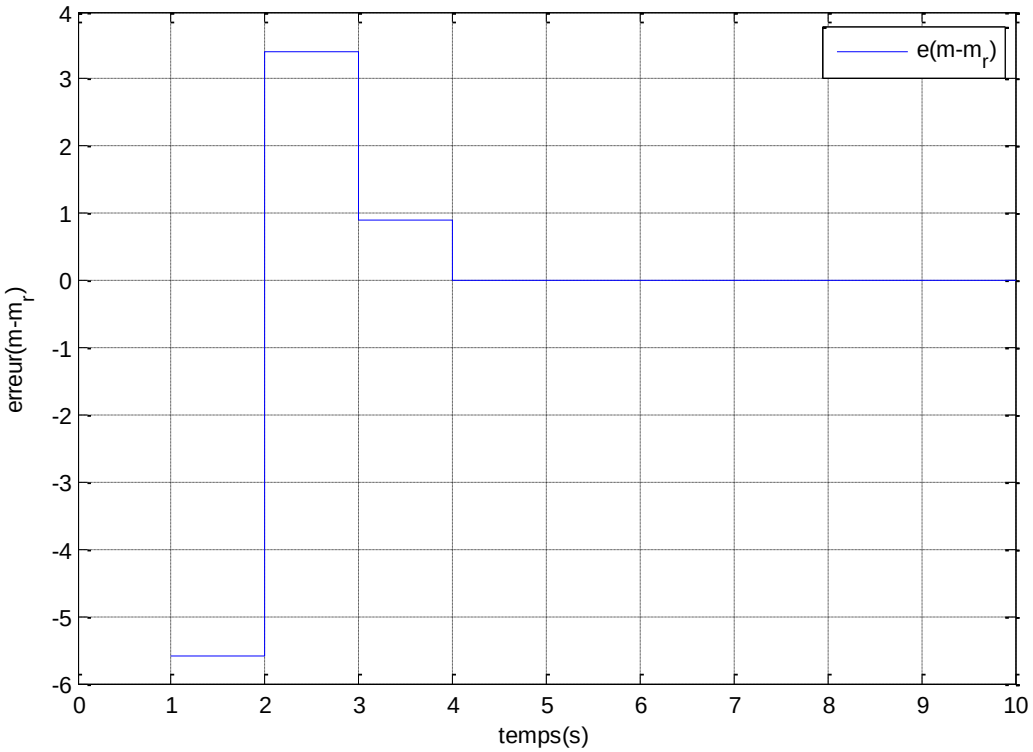



**Figure.43 :** Erreurs de synchronisation des états xr, zr et du message m.

## 3.3) Résultats de simulation :

Dans cette partie, on présente les résultats de simulation pour la synchronisation du système (3.2) (présenté dans l'émetteur) et son observateur (présenté dans le récepteur) donné dans le paragraphe 3.2).

Le message à envoyer est un signal carré avec une amplitude de 0.1. Dans notre simulation, la période *T* choisie est égale à 1s.

Les résultats de la simulation pour récupérer les deux états *x(k)*, *y(k)* et le message *m(k)* de l'émetteur sont montrés dans la figure.42.

La figure.43 donne les erreurs de synchronisation (entre émetteur et récepteur) sur les états *x(k)*, *z(k)* et le message *m*.

La reconstitution des deux états est montrée étape par étape, i.e., le première état reconstitué est *x(k)* et le second est *z(k)*, expliquée précédemment.

Enfin, le signal message *m(k)* est reconstitué après synchronisation de tous les états.

Ceci nous permet d'établir que l'erreur $e_x(k) = x(k) - \hat{x}(k)$ disparait au bout de $T = 1s$, ce qui correspond au retard d'un pas d'après l'équation (3.4) (voir figure.43).

L'erreur $e_z(k) = z(k) - \hat{z}(k)$ disparait au bout de $2T = 2s$, ce qui correspond à un retard de deux pas sur la sortie d'après l'équation (3.5) (voir figure.43).

Et enfin, l'erreur du message $e_m(k) = m(k) - \hat{m}(k)$ disparait au bout de $3T = 3s$, ce qui correspond à un retard de trois pas d'après l'équation (3.6) (voir figure.43).

## 3.4.Conclusion :

Dans ce chapitre, un schéma de transmission basé sur un système dynamique hyper chaotique discret a été conçu pour une communication privée. En premier lieu, le principe de la méthode est exposé avec présentation de l'émetteur (système chaotique à temps discret dit Hénon modifié) et du récepteur (observateur discret retardé). Trois conditions ont été vérifiées afin de concevoir l'observateur. Des figures illustrant la synchronisation ainsi que les erreurs de synchronisation de tous les états et du message ont été données. Et enfin, les résultats de simulation sont illustrés.

## 4.1. Introduction :

Les premières expérimentations de communications basées sur le chaos ont été réalisées sur des circuits électroniques analogiques [30], mais la sensibilité des systèmes à la déviation des composants a longtemps posé problème dans les systèmes de communication. En effet, les systèmes chaotiques sont très sensibles aux perturbations et le signal issu de deux systèmes très proches divergent très rapidement (coefficient de Lyapunov). En revanche, les systèmes numériques programmables (DSP, FPGA, microcontrôleurs) permettent de générer aisément et de manière reproductible des signaux issus de la discrétisation d'équations chaotiques [31]. Ainsi, notre choix s'est porté sur une carte Arduino Uno pour la mise en œuvre de l'émetteur hyper chaotique (Hénon modifié).

## 4.2. Présentation de la carte Arduino Uno:

Arduino est une plateforme de prototypage d'objets interactifs à usage créatif constituée d'une carte électronique à microcontrôleur et d'un environnement de programmation. Cet environnement matériel et logiciel permet à l'utilisateur de formuler ses projets par l'expérimentation directe avec l'aide de nombreuses ressources disponibles en ligne. Pont tendu entre le monde réel et le monde numérique, Arduino permet d'étendre les capacités de relations humain/machine ou environnement/machine.

Le modèle Uno de la société Arduino est une carte électronique dont le cœur est un microcontrôleur ATMEL de référence ATMega328. Le microcontrôleur ATMega328 est un microcontrôleur 8bits de la famille AVR dont la programmation peut être réalisée en langage C. L'intérêt principal des cartes Arduino (d'autres modèles existent) est leur facilité de mise en œuvre. Arduino fournit un environnement de développement s'appuyant sur des outils open source. Le chargement du programme dans la mémoire du microcontrôleur se fait de façon très simple par port USB. En outre, des bibliothèques de fonctions "clé en main" sont également fournies pour l'exploitation d'entrées-sorties courantes : gestion des E/S TOR, gestion des convertisseurs ADC, génération de signaux PWM, exploitation de bus TWI/I²C, exploitation de servomoteurs…

- Présentation de la carte Arduino Uno :

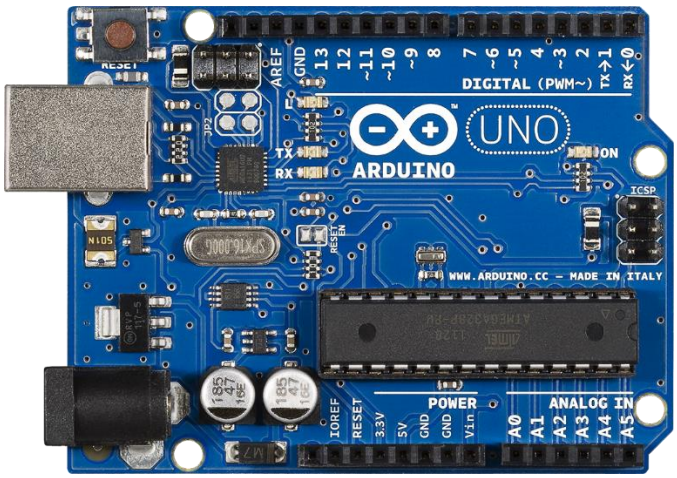



**Figure.44:** Carte Arduino-Uno.

### 4.2.1. Schéma simplifié de la carte Arduino-Uno :

Les signaux d'entrée-sortie du microcontrôleur sont reliés à des connecteurs selon le schéma ci-dessous :

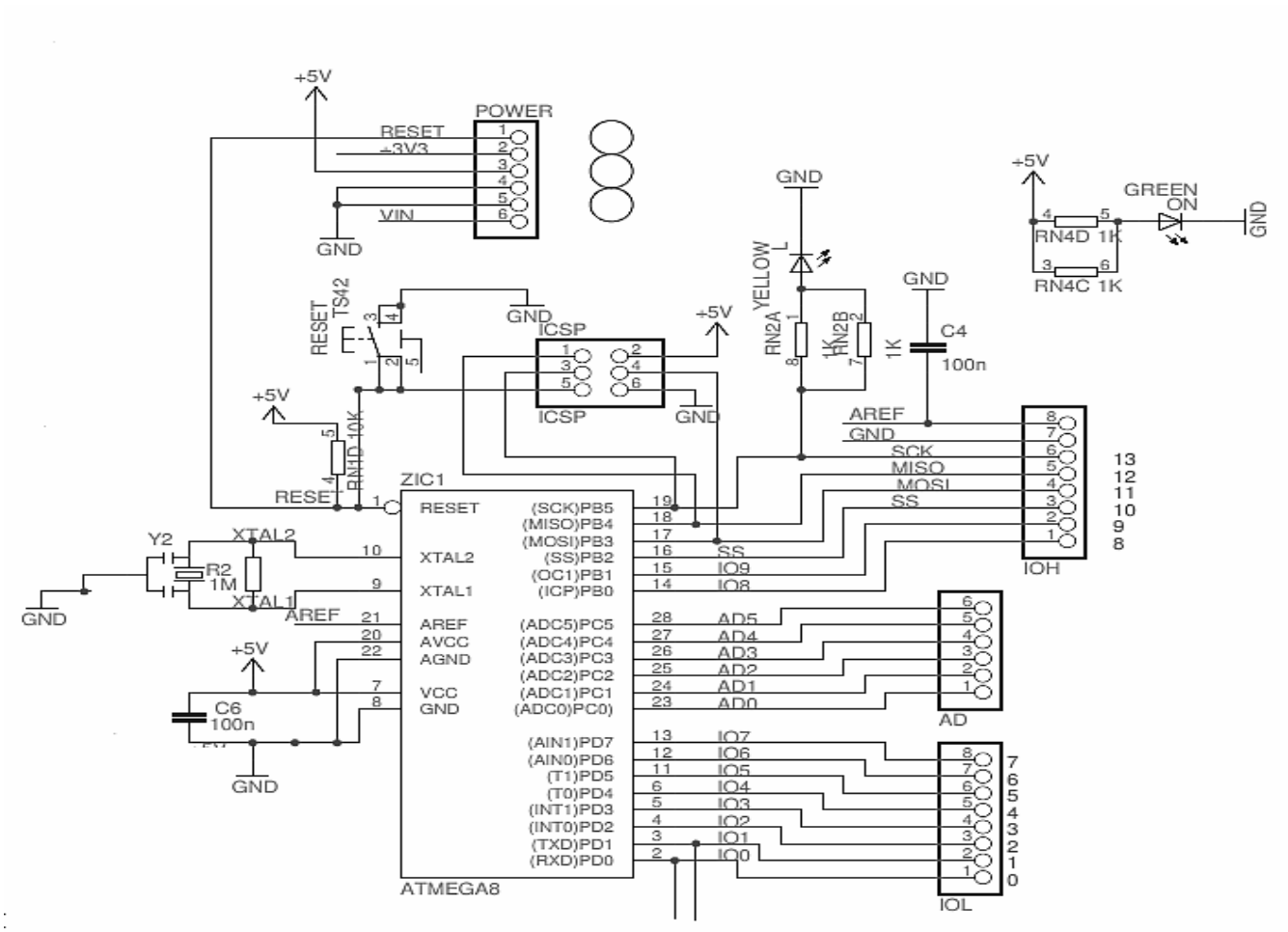



**Figure.45**: Schéma simplifié de la carte Arduino-Uno

### 4.2.2. Microcontrôleur ATMEL ATMega328:

Le microcontrôleur utilisé sur la carte Arduino UNO est un microcontrôleur **ATMega328**. C'est un microcontrôleur ATMEL de la famille AVR 8bits. [32]

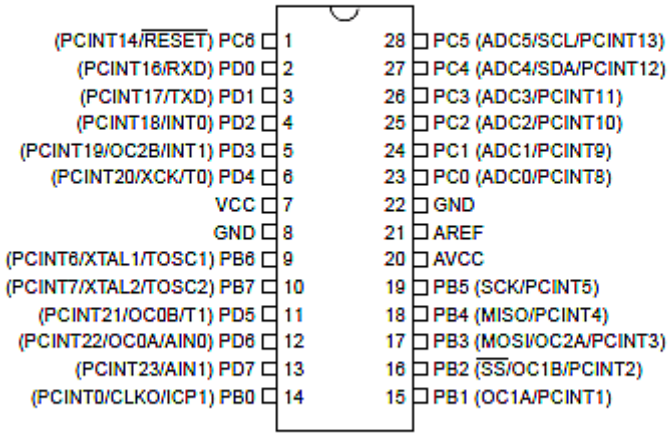



**Figure.46**: Microcontrôleur ATMEL ATMega328.

Les principales caractéristiques sont :

- *FLASH* = mémoire programme de 32Ko
- *SRAM* = données (volatiles) 2Ko
- *EEPROM* = données (non volatiles) 1Ko
- *Digital I/O (entrées-sorties Tout Ou Rien***)** = 3 ports PortB, PortC, PortD (soit 23 broches en tout I/O)
- *Timers/Counters* : Timer0 et Timer2 (comptage 8 bits), Timer1 (comptage 16bits)

Chaque Timer peut être utilisé pour générer deux signaux PWM. (6 broches OCxA/OCxB)

- *Plusieurs broches multifonctions :* certaines broches peuvent avoir plusieurs fonctions différentes choisies par programmation.
- *PWM* = 6 broches *OC0A(PD6), OC0B(PD5), 0C1A(PB1), OC1B(PB3), OC2A(PB3), OC2B(PD3)*
- *Analog to Digital Converter* (résolution 10bits) = 6 entrées multiplexées *ADC0(PC0) à ADC5(PC5)*
- *Gestion bus I2C (TWI Two Wire Interface)* = le bus est exploité via les broches *SDA(PC5)/SCL(PC4).*
- *Port série (USART)* = émission/réception série via les broches *TXD(PD1)/RXD(PD0)*
- *Comparateur Analogique* = broches AIN0(PD6) et AIN1 (PD7) peut déclencher interruption

- *Watchdog Timer programmable.*

- *Gestion d'interruptions (24 sources possibles*) **:** en résumé

✓ Interruptions liées aux entrées **INT0 (PD2) et INT1 (PD3)**
✓ Interruptions sur changement d'état des broches **PCINT0** à **PCINT23**
✓ Interruptions liées aux Timers 0, 1 et 2 (plusieurs causes configurables)
✓ Interruption liée au comparateur analogique
✓ Interruption de fin de conversion **ADC**
✓ Interruptions du port série **USART**
✓ Interruption du bus **TWI** (**I2C**)

On distingue plusieurs types de cartes Arduino, dont les caractéristiques sont données dans le tableau.4.1 :

| Name | Processor | Operating Voltage/Input Voltage | CPU Speed | Analog In/Out | Digital IO/PWM | EEPROM [KB] | SRAM [KB] | Flash [KB] | USB | UART |
|---|---|---|---|---|---|---|---|---|---|---|
| Uno | ATmega328 | 5 V/7-12 V | 16 Mhz | 6/0 | 14/6 | 1 | 2 | 32 | Regular | 1 |
| Due | AT91SAM3X8E | 3.3 V/7-12 V | 84 Mhz | 12/2 | 54/12 | - | 96 | 512 | 2 Micro | 4 |
| Leonardo | ATmega32u4 | 5 V/7-12 V | 16 Mhz | 12/0 | 20/7 | 1 | 2.5 | 32 | Micro | 1 |
| Mega 2560 | ATmega2560 | 5 V/7-12 V | 16 Mhz | 16/0 | 54/15 | 4 | 8 | 256 | Regular | 4 |
| Mega ADK | ATmega2560 | 5 V/7-12 V | 16 Mhz | 16/0 | 54/15 | 4 | 8 | 256 | Regular | 4 |
| Micro | ATmega32u4 | 5 V/7-12 V | 16 Mhz | 12/0 | 20/7 | 1 | 2.5 | 32 | Micro | 1 |
| Mini | ATmega328 | 5 V/7-9 V | 16 Mhz | 8/0 | 14/6 | 1 | 2 | 32 | - | - |
| Nano | ATmega168<br>ATmega328 | 5 V/7-9 V | 16 Mhz | 8/0 | 14/6 | 0.512<br>1 | 1<br>2 | 16<br>32 | Mini-B | 1 |
| Ethernet | ATmega328 | 5 V/7-12 V | 16 Mhz | 6/0 | 14/4 | 1 | 2 | 32 | Regular | - |
| Esplora | ATmega32u4 | 5 V/7-12 V | 16 Mhz | - | - | 1 | 2.5 | 32 | Micro | - |
| ArduinoBT | ATmega328 | 5 V/2.5-12 V | 16 Mhz | 6/0 | 14/6 | 1 | 2 | 32 | - | 1 |
| Fio | ATmega328P | 3.3 V/3.7-7 V | 8 Mhz | 8/0 | 14/6 | 1 | 2 | 32 | Mini | 1 |
| Pro (168) | ATmega168 | 3.3 V/3.35-12 V | 8 Mhz | 6/0 | 14/6 | 0.512 | 1 | 16 | - | 1 |
| Pro (328) | ATmega328 | 5 V/5-12 V | 16 Mhz | 6/0 | 14/6 | 1 | 2 | 32 | - | 1 |
| Pro Mini | ATmega168 | 3.3 V/3.35-12 V<br>5 V/5-12 V | 8 Mhz<br>16Mhz | 6/0 | 14/6 | 0.512 | 1 | 16 | - | 1 |
| LilyPad | ATmega168V<br>ATmega328V | 2.7-5.5 V/2.7-5.5 V | 8 Mhz | 6/0 | 14/6 | 0.512 | 1 | 16 | - | - |
| LilyPad USB | ATmega32u4 | 3.3 V/3.8-5V | 8 Mhz | 4/0 | 9/4 | 1 | 2.5 | 32 | Micro | - |
| LilyPad Simple | ATmega328 | 2.7-5.5 V/2.7-5.5 V | 8 Mhz | 4/0 | 9/4 | 1 | 2 | 32 | - | - |
| LilyPad SimpleSnap | ATmega328 | 2.7-5.5 V/2.7-5.5 V | 8 Mhz | 4/0 | 9/4 | 1 | 2 | 32 | - | - |

**Tableau.4.1 :** Cartes Arduino et caractéristiques.

## 4.3. Programmation de la carte Arduino :

La carte Arduino peut être programmée de différentes manières, nous en avons choisi deux :

### 4.3.1. Logiciel IDE (Integrated development environment) Arduino :

1. Pour télécharger le logiciel, il faut se rendre sur la page de téléchargement du site : http://arduino.cc/en/Main/Software. Décompresser le fichier avec un utilitaire de décompression (7-zip, WinRar,...).A l'intérieur du dossier se trouvent quelques fichiers et l'exécutable du logiciel.
2. Installer le logiciel
3. Dé-zipper le pilote FTDI USB Drivers.zip
4. Brancher l'Arduino et pointer l'installeur Windows vers le pilote
5. La carte est prête à accueillir un programme Utilisateur


**Figure.47 :** Branchement de la carte Arduino Uno.

Lorsque l'on connecte la carte à l'ordinateur sur le port USB, un petit message en bas de l'écran apparaît. Théoriquement, la carte que l'on veut utiliser doit s'installer toute seule. Cependant, si l'on est sous Windows 8.1, ce qui est notre cas, il se peut que ça ne marche pas automatiquement. Dans ce cas, il faut laisser la carte branchée puis ensuite allez dans le panneau de configuration. Une fois-là, cliquez sur "système" puis dans le panneau de gauche sélectionnez "gestionnaire de périphériques". Une fois ce menu ouvert, on voit un composant avec un panneau "attention" jaune. Faire un clic droit sur le composant et cliquer sur " Mettre à jour les pilotes ". Dans le nouveau menu, on sélectionne l'option " Rechercher le pilote moi-même". Enfin, il ne reste plus qu'à aller sélectionner le bon dossier contenant le driver. Il se trouve dans le dossier d'Arduino décompressé un peu plus tôt et se nomme « drivers ».

Après l'installation et une suite de clignotement sr les micro-LED de la carte, celle-ci devrait être fonctionnelle; une petite LED verte témoigne de la bonne alimentation de la carte :

On lance le logiciel en double-cliquant sur l'icône avec le symbole "infinie" en vert. C'est l'exécutable du logiciel. Après quelques secondes, le logiciel s'ouvre. Une fenêtre s'affiche :


**Figure.48** : Fenêtre du logiciel Arduino.

Ensuite, il faut dire au logiciel quel est le nom de la carte utilisée et désigner le bon port série (USB série) sur lequel elle est branchée, port que l'on ré-indique s'il y a changement de carte Arduino :

Sur Windows :

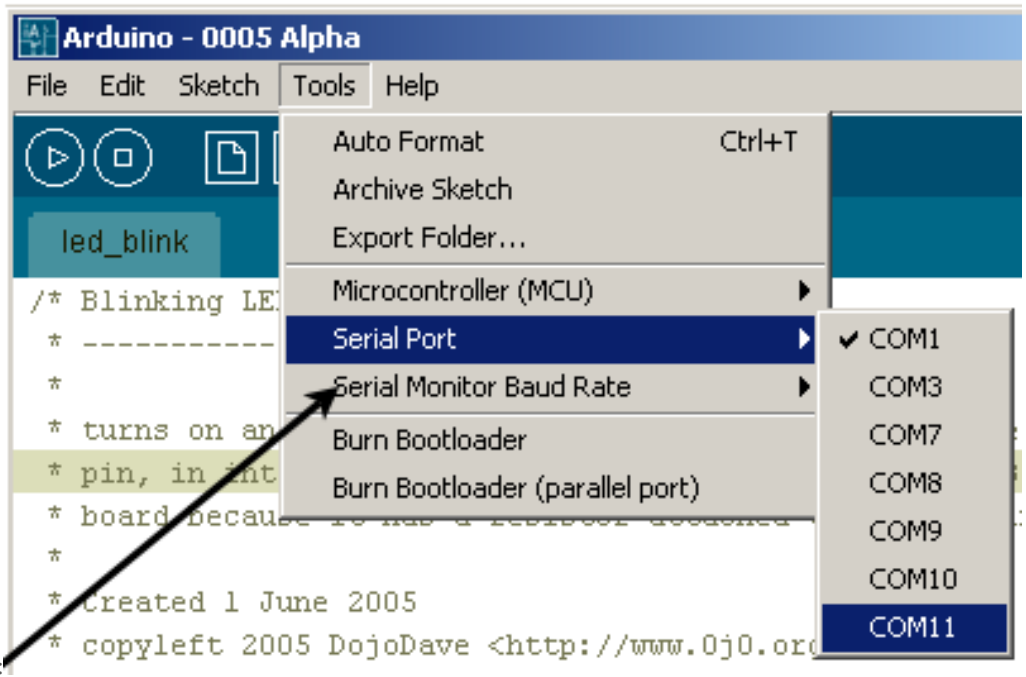



**Figure.49** : Choix du port de connexion de la carte.

Pour trouver le port de connexion de la carte, aller dans le *gestionnaire de périphérique* qui se trouve dans le *panneau de configuration.* A la ligne *Ports (COM et LPT)* devrait se trouver *Arduino Uno (COMX).*

Il faut ensuite choisir le type de carte Arduino utilisée :

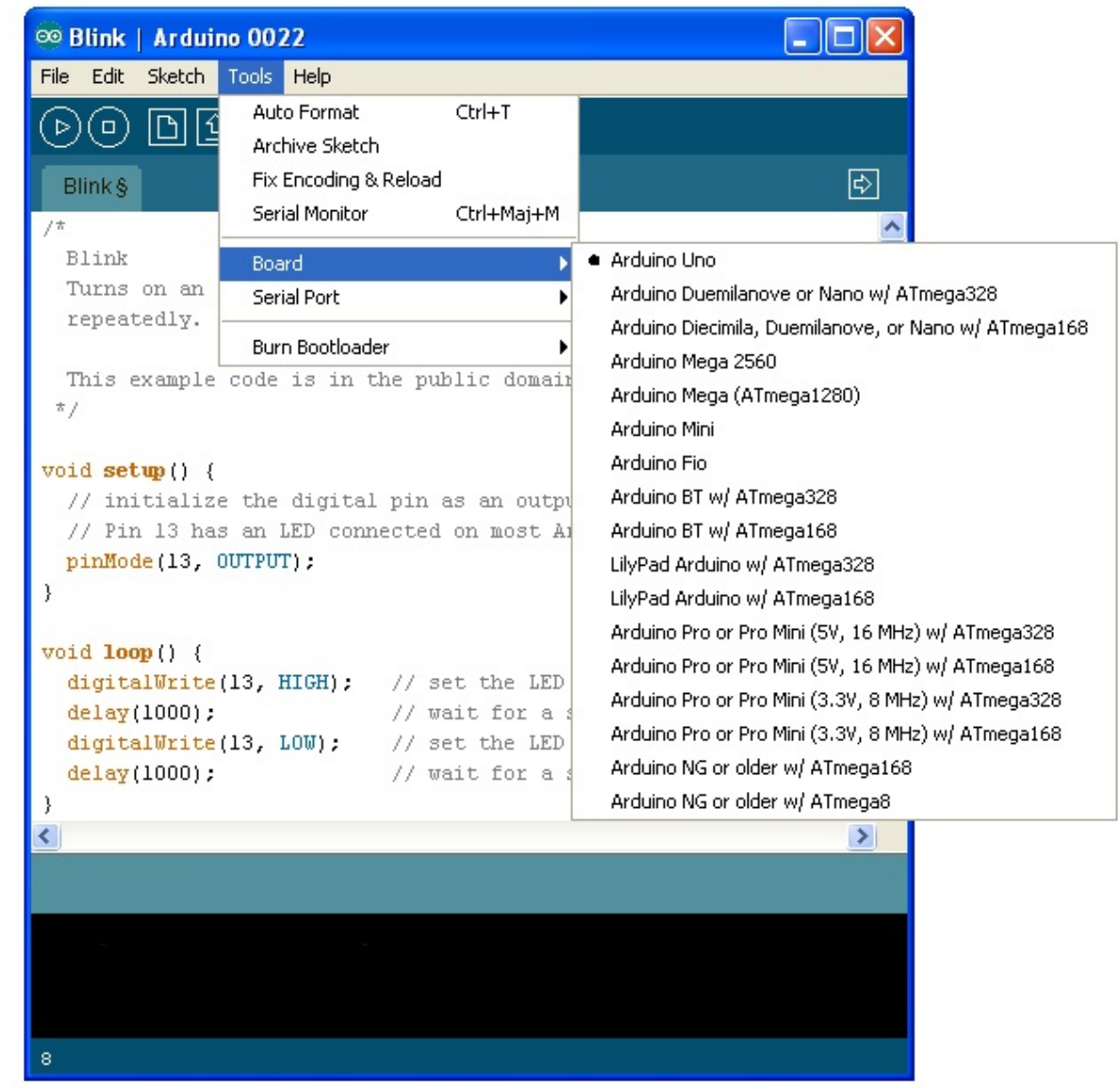



**Figure.50** : Choix de la carte Arduino-Uno.

La carte est maintenant prête à être utilisée, mais il faut quand-même s'assurer de son bon fonctionnement en faisant un test. Nous allons tester notre matériel en chargeant un programme qui fonctionne dans la carte. Le logiciel Arduino contient plusieurs exemple de programmes, que nous allons donc utiliser pour tester la carte.

L'exemple choisi est un simple programme qui consiste à faire clignoter une LED. Le nom de l'exemple est Blink, et se trouve dans la catégorie Basics des exemples de l'IDE Arduino.

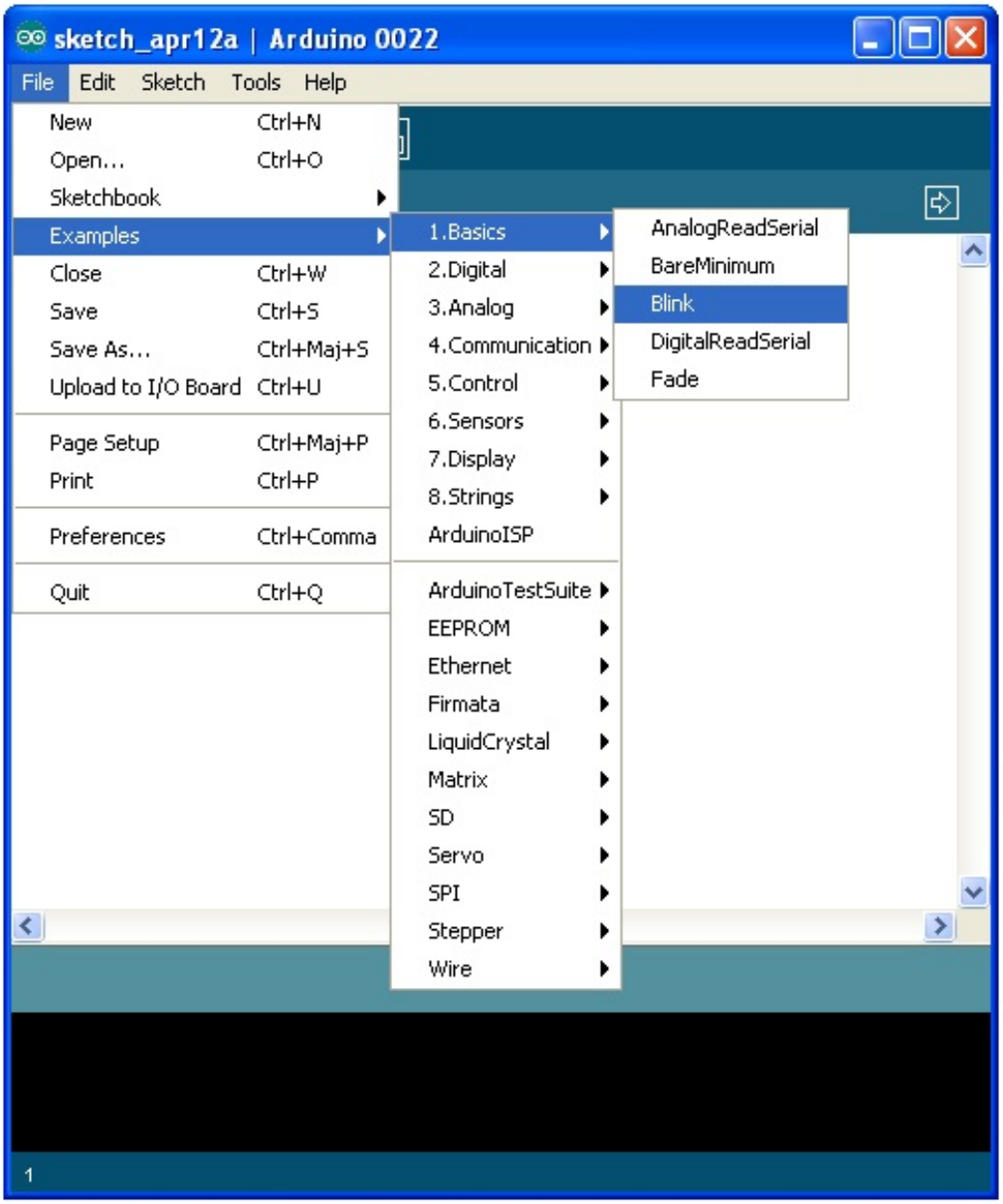



**Figure.51**: Ouvrir le programme Blink.

Une fois que l'on a cliqué sur *Blink*, une nouvelle fenêtre va apparaître. Elle va contenir le programme *Blink.*

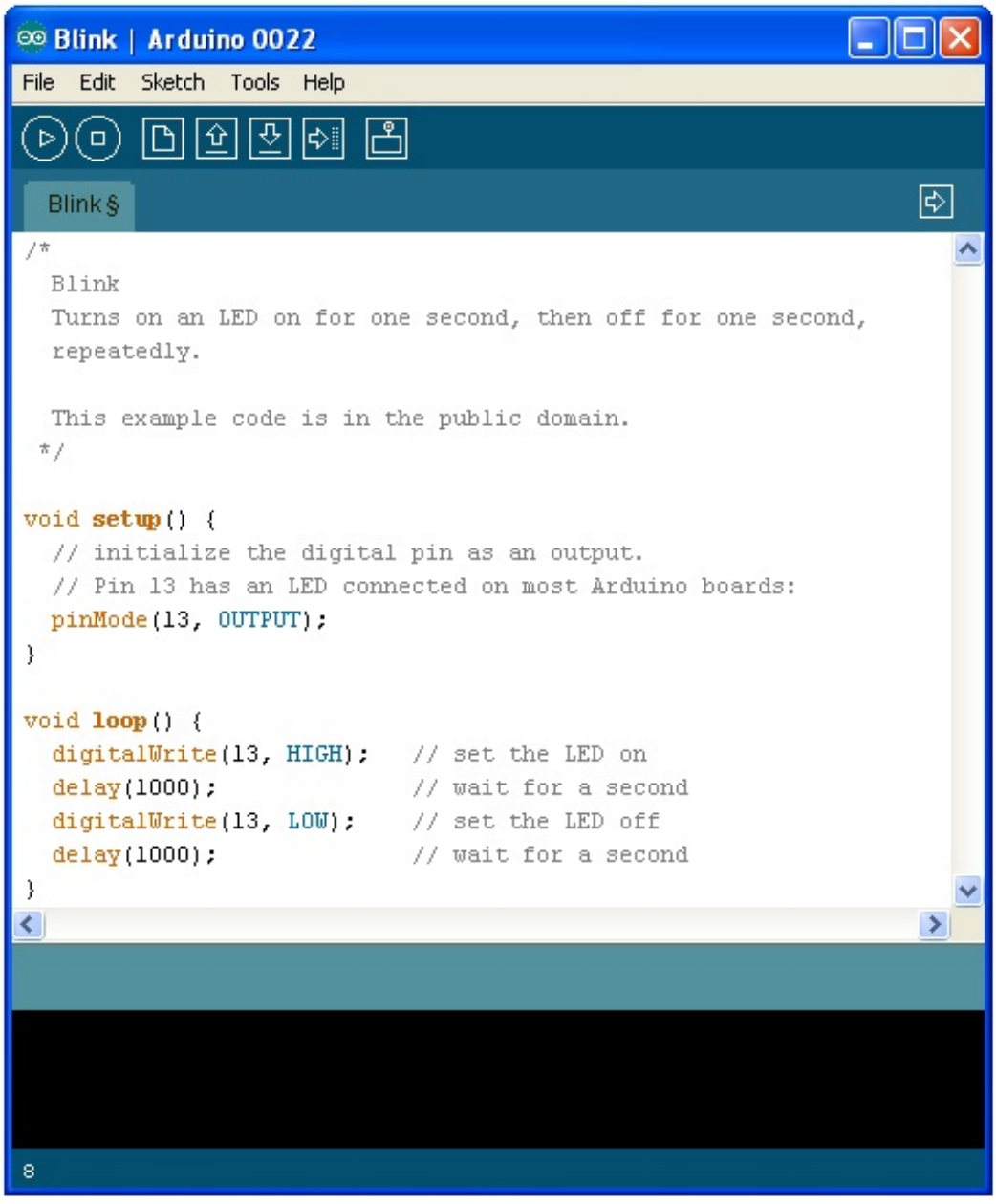



**Figure.52** : *Contenu du programme Blink.*

On peut maintenant envoyer le programme *Blink* vers la carte, puisque le nom de la carte, ainsi que le port sur lequel elle est branchée ont déjà été désignés.

La dernière étape consiste à envoyer le programme dans la carte. Pour ce faire, il suffit de cliquer sur le bouton U*pload* (ou « Déployer ») en jaune-orangé sur la photo :

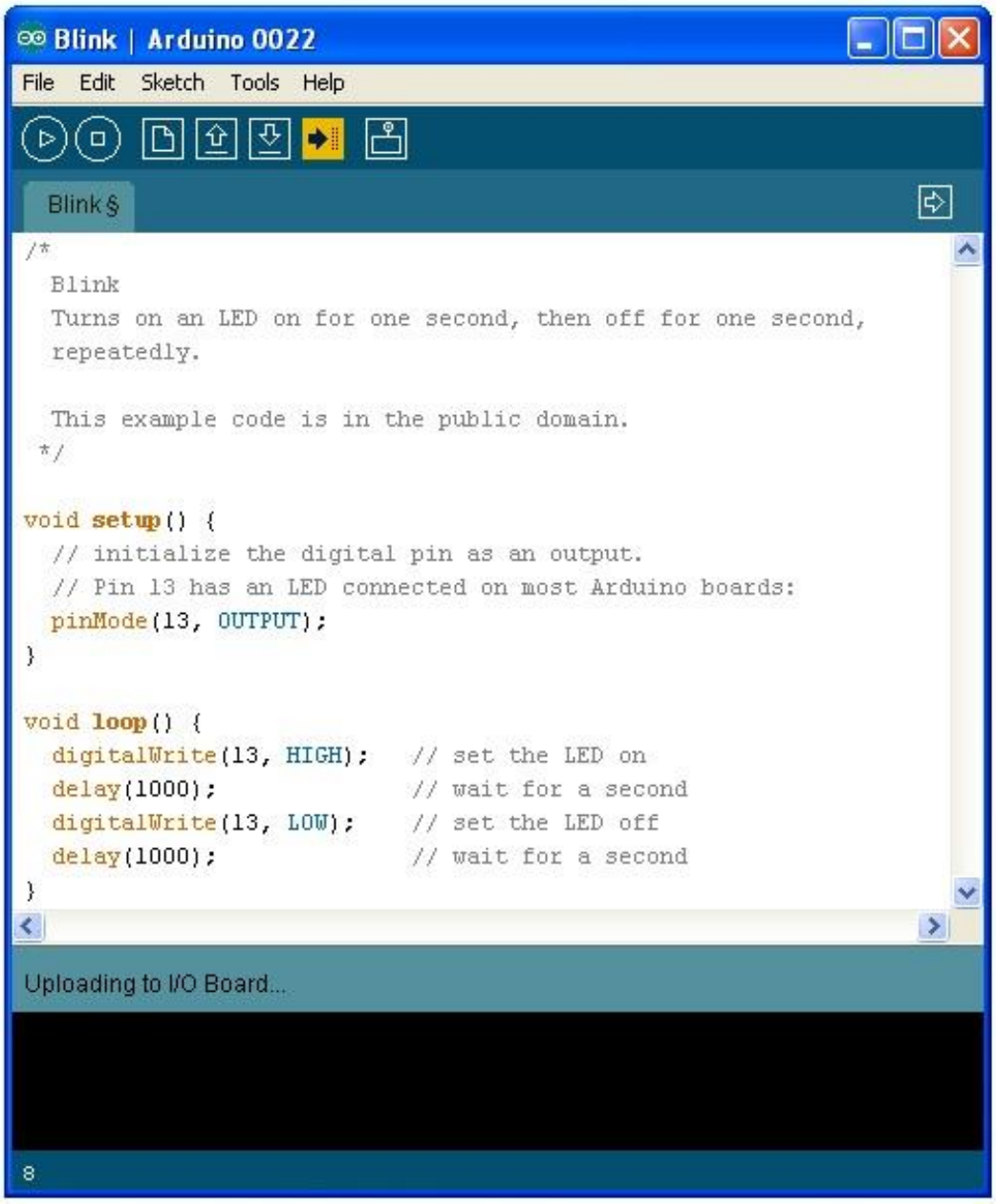



**Figure.53** : Envoi du programme Blink dans la carte.

En bas dans l'image, on peut lire : « *Uploading to I/O Board...* ». Cela signifie que le logiciel est en train d'envoyer le programme dans la carte.

Une fois que le logiciel a fini d'envoyer le programme dans la carte, il affiche un autre message:

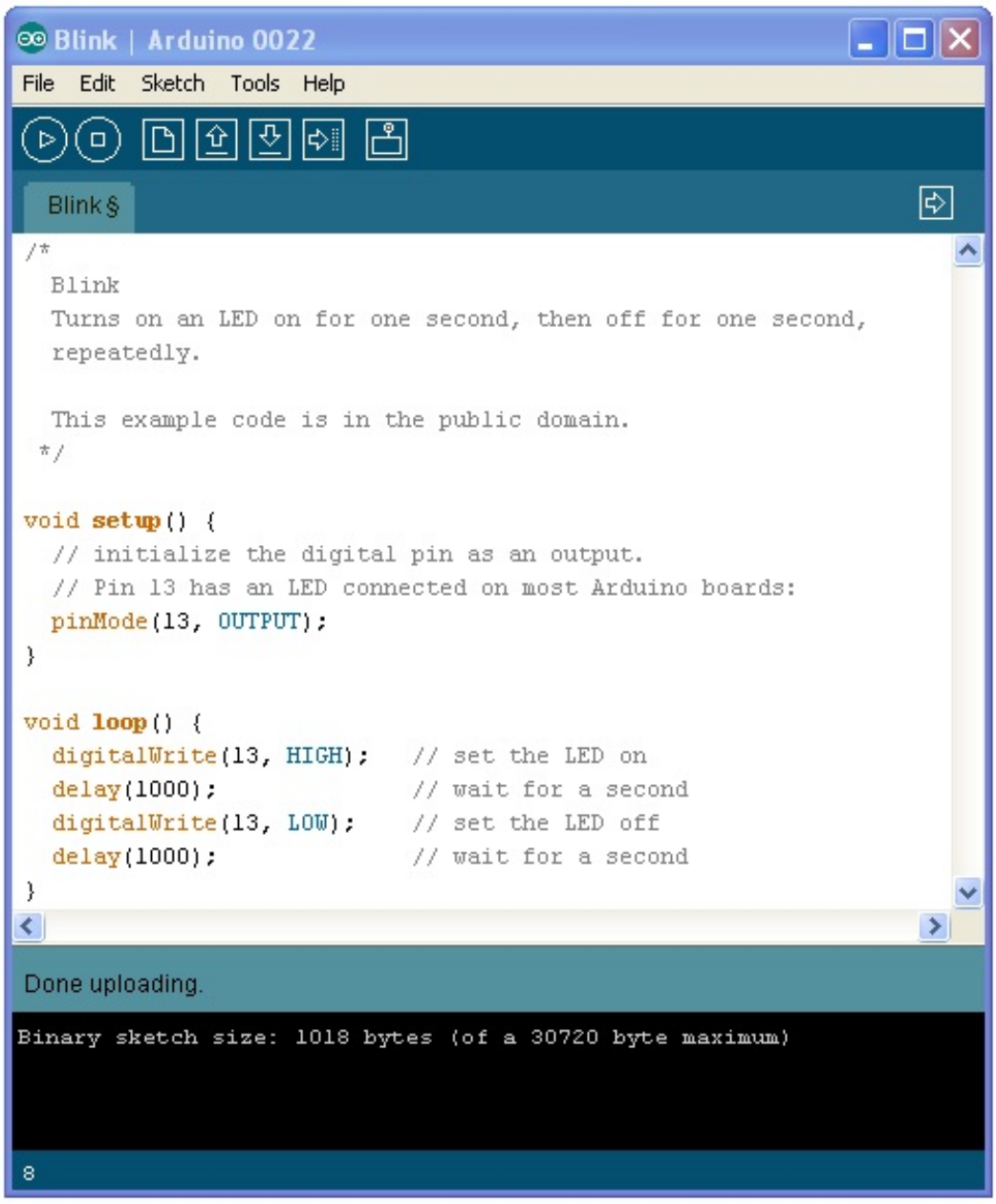



**Figure.54**: Fin de l'Upload.

Le message affiché : " *Done uploading* " signale que le programme à bien été chargé dans la carte. Si le matériel fonctionne, une LED sur la carte devrait clignoter :

Toutes ces étapes doivent être faites avant d'utiliser la carte pour vérifier son bon fonctionnement.

- **Structure d'un programme IDE Arduino :**

L'outil impose de structurer l'application de façon spécifique. Le compilateur utilisé est

AVR GCC (compilateur C/C++ pour processeur AVR). Le langage Arduino est très proche du C et du C++. La particularité est qu'une structure de programme est imposée au programmeur.

La structure d'un programme IDE Arduino a trois phases consécutives :

1. La définition des constantes et des variables.
2. void setup ( ) : La configuration des entrées et sorties (doit contenir les initialisations (times, interrupts...)
3. void loop ( ) : La programmation des interactions et comportements (fonction répétée indéfiniment)

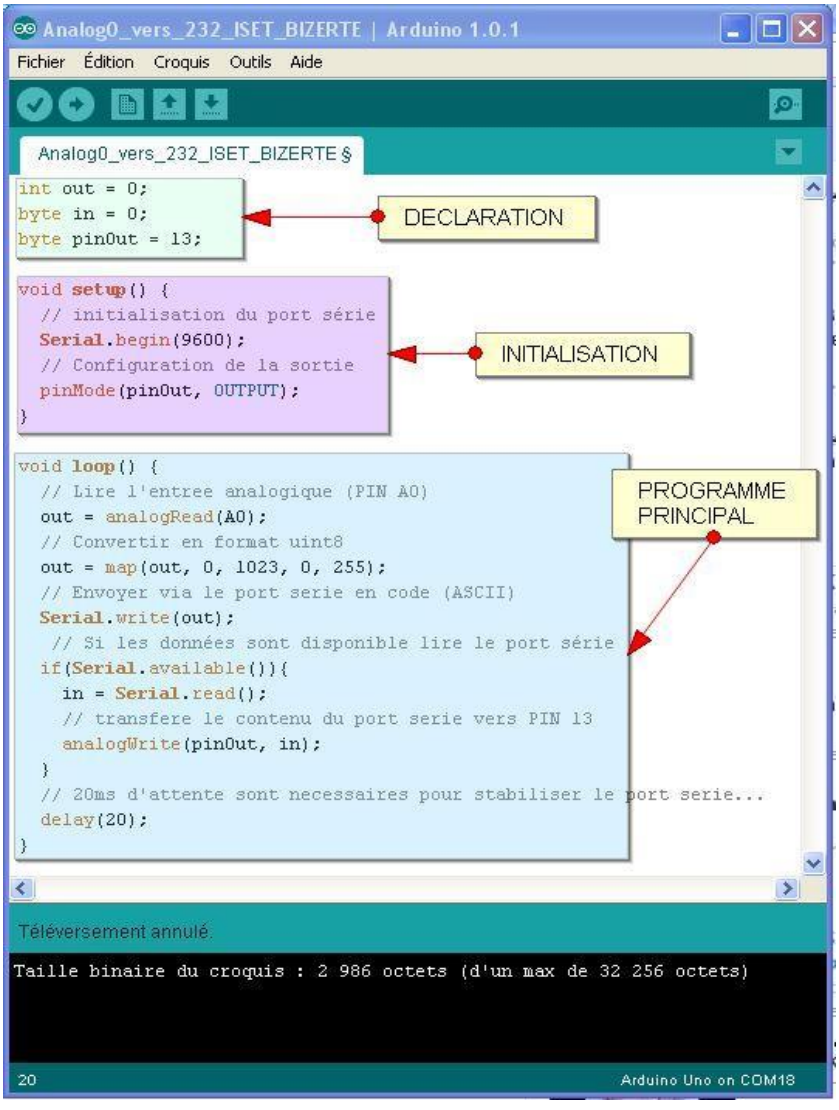



**Figure.55** : Structure d'un programme.

### 4.3.2. Logiciel Matlab :

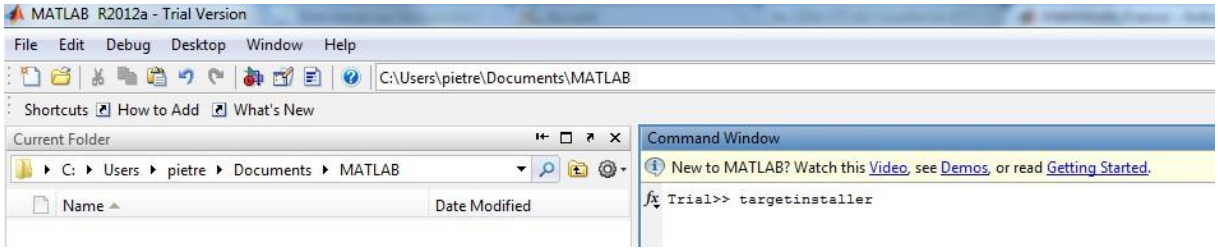
- *Pour utiliser l'Arduino en tant qu'interface d'entrée/sorties :*
  Télécharger Matlab support package for Arduino « Arduino-io package » à partir du site http://www.mathworks.com/hardware-support/arduino-simulink.html, le copier et le dé-zipper à la racine du disque par exemple dans *c:\arduinoio.*
- *Pour utiliser Arduino en tant que cible du programme Simulink compilé (ce qui est notre cas) :*

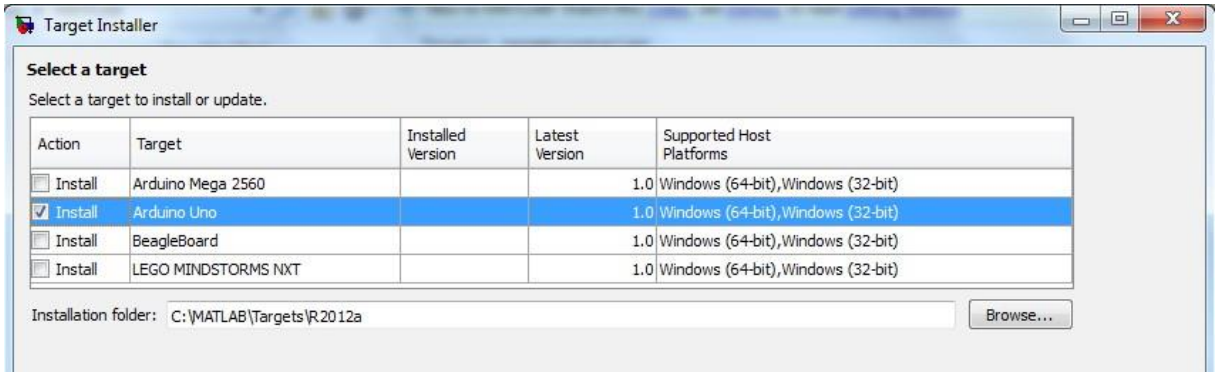
Ceci consiste à utiliser la carte Arduino comme une cible. Matlab compile le programme saisi depuis Matlab/Simulink et le transfert dans l'Arduino de façon transparente. Le programme est alors totalement autonome si on le souhaite. Il peut aussi converser avec la liaison série...

Installation de la cible depuis Matlab 2014a :

- Il est possible de l'installer sur Matlab 2012 et 2013, mais son utilisation y est beaucoup moins simple.

D'abord, à l'ouverture de Matlab, taper dans CommandWindow *targetinstaller,* comme le montre la figure suivante, suivie de la suite de l'installation :



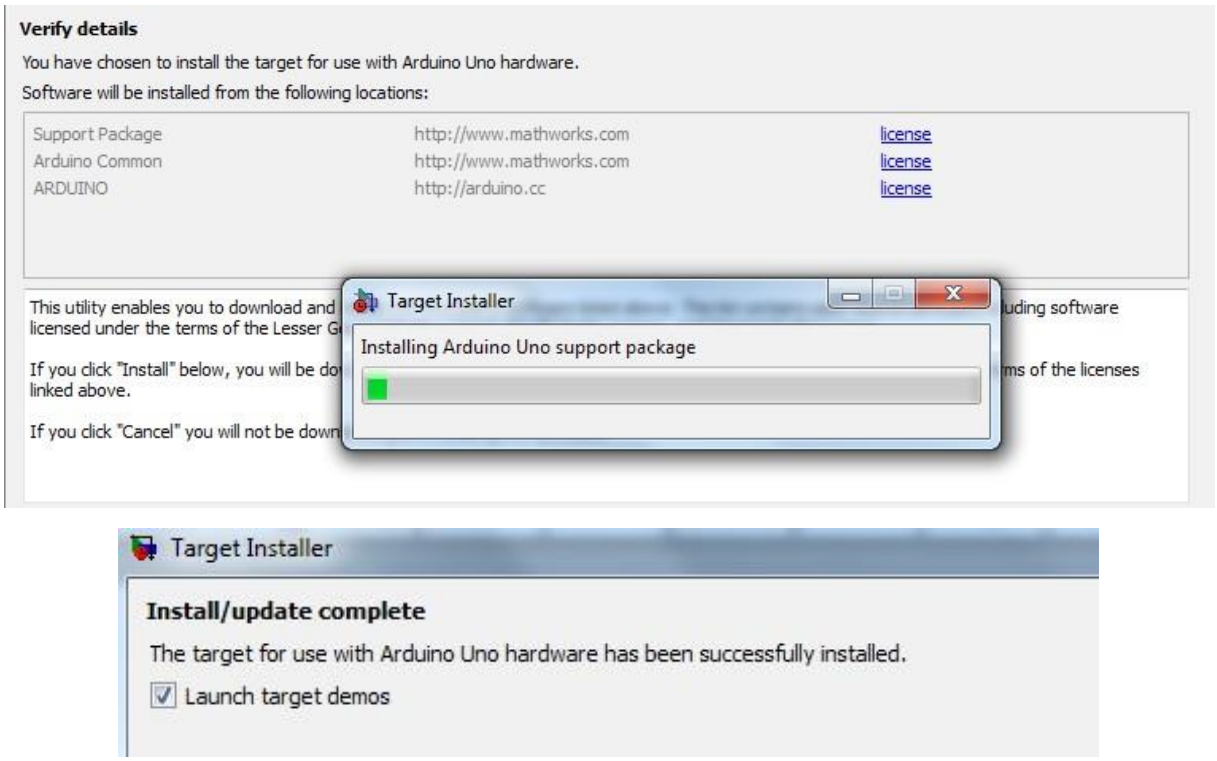



**Figure.56 :** Installation de l'Arduino Target depuis Matlab.

Une fois installée, nous avons accès à la bibliothèque Arduino-Target, qui va permettre de construire les différents blocs constituant notre programme Matlab utilisant une carte Arduino.

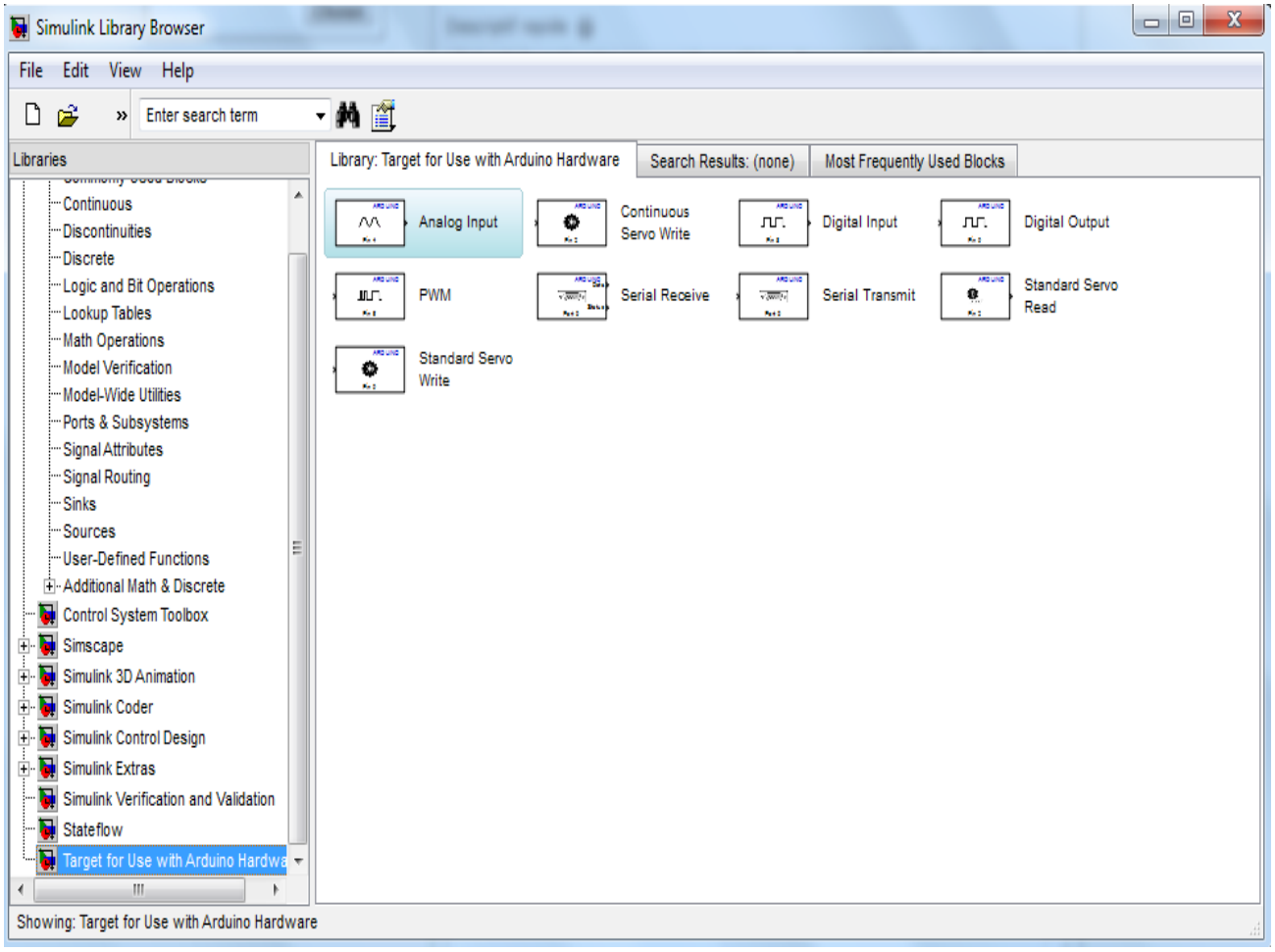



**Figure.57:** Bibliothèque Arduino Target.

Nous reprenons l'exemple précédent, fait sur l'IDE Arduino, en utilisant matlab simulink :

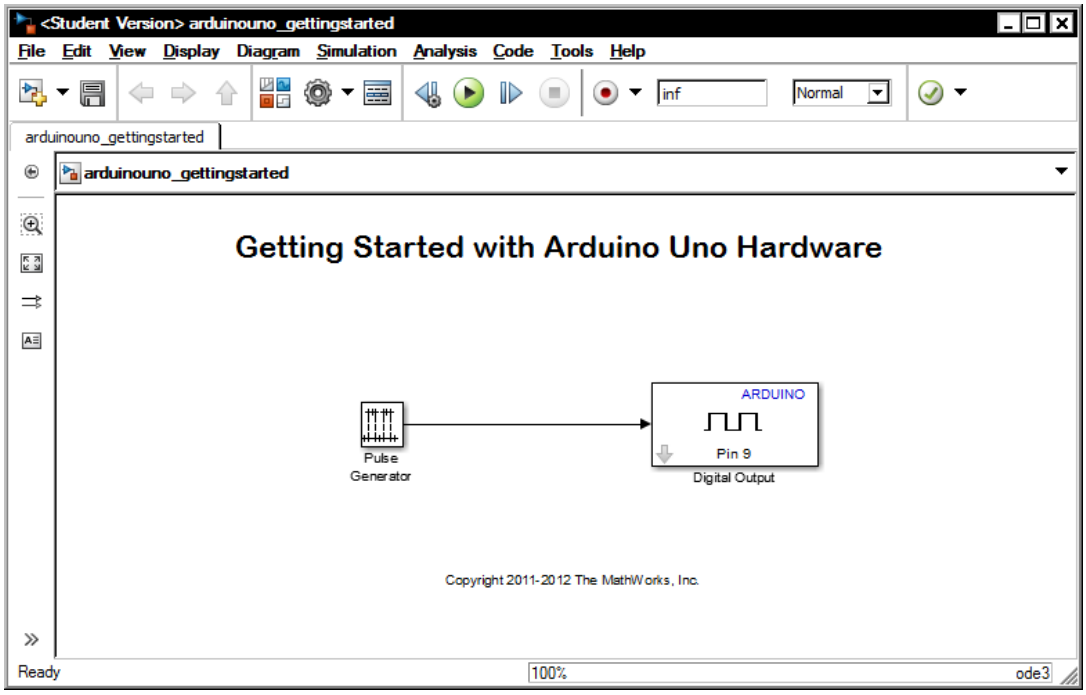



**Figure.58 :** Exemple de programme Simulink pour Arduino.

Pour envoyer ce programme dans la carte Arduino, il suffit d'enregistrer le model, en faisant : *File > Save As.* Ensuite, dans le model, sélectionner *Tools > Run on Target Hardware > Prepare to Run.* Cela va changer les paramètres de configuration du model Simulink. Lorsque la fenêtre *Run on Target Hardware* s'ouvre ; choisir le type de carte Arduino utilisée. Enfin, cliquer sur *Deploy to Hardware*. Cela va automatiquement déployer et faire fonctionner le programme dans la carte.

## 4.4. Réalisation de l'émetteur hyper chaotique sur carte Arduino-Uno :

Un système de communication basé sur un système dynamique hyper chaotique discret (Hénon modifié) peut être implémenté sur deux cartes Arduino-Uno. Dans ce chapitre, la première carte est programmée en tant qu'émetteur. Il est possible de programmer une seconde carte Arduino en tant que récepteur, puis faire une simple liaison filaire entre les deux cartes qui fera guise de canal de transmission. Il s'agit en fait d'une liaison série (ou bien faire une liaison SPI ou I²C) entre les deux Arduinos, c'est-à-dire relier les ports Tx et Rx entre eux, ainsi que les masses. D'autres liaisons sont possibles (hertziennes, infrarouges, Bluetooth…), pour lesquelles une étude supplémentaire du canal de transmission doit être faite (choix du protocole de transmission, etc.).

La fréquence d'oscillation du système de Hénon modifié est de 270Hz,

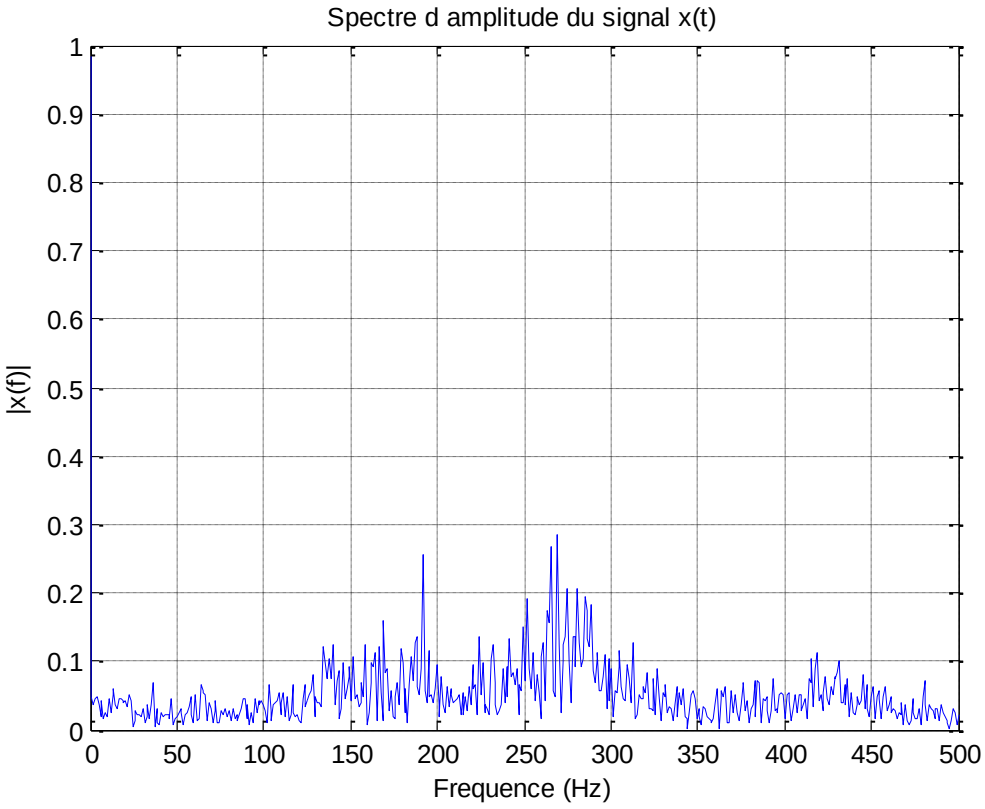



**Figure.59** : Spectre d'amplitude du signal x(t) du système de Hénon modifié.

### 4.4.1. Programmation de l'émetteur :

La carte Arduino est programmée en tant qu'émetteur à l'aide des deux programmes, Matlab (Simulink) et IDE Arduino.

Sous Matlab (Simulink), les états du système de Hénon modifié sont programmés en sorties PWM (*Pulse Width Modulation*)**.** La PWM est un signal de fréquence fixe (~490Hz environ pour la plupart des pins et ~980Hz pour les pins 5 et 6) qui a un rapport cyclique qui varie avec le temps suivant "les ordres qu'elle reçoit". Le *rapport cyclique*, désigne le fait que le niveau logique 1 peut ne pas durer le même temps que le niveau logique 0. Le rapport cyclique est mesuré en pour cent (%). Plus le pourcentage est élevé, plus le niveau logique 1 est présent dans la période et moins le niveau logique 0 l'est. Et inversement. Le rapport cyclique du signal est donc le pourcentage de temps de la période durant lequel le signal est au niveau logique 1.

Avec une résolution de 8-bits et une horloge de 16Mhz, la fréquence max de la sortie pwm (au niveau des pins 5 et 6 contrôlées par le Timer 0) en théorie serait d'environ 64Khz (16Mhz/256), suivant l'équation suivante :

$$f_{PWM} = \frac{f_{CLK}}{N \times 256} \qquad (4.1)$$

$$f_{PWM} = \frac{f_{CLK}}{N \times 510} \qquad (4.2)$$

$$f_{PWM} : fréquence\ pwm$$

$$f_{CLK} : fréquence\ de\ l'oscillateur\ \text{à}\ quartz$$

$$N : préscalaire\ N = \{1, 8, 64, 256, 1024\}$$

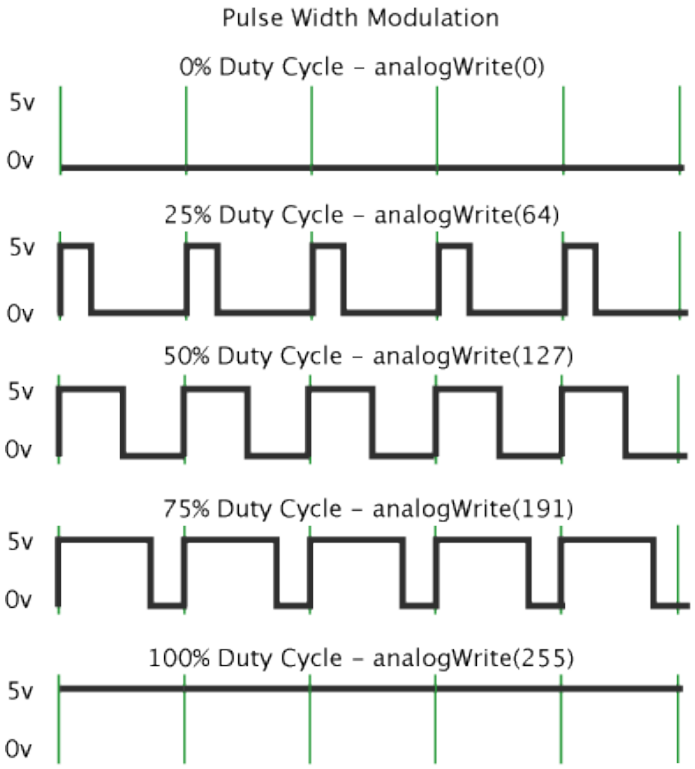



**Figure.60 :** Exemple de signal avec rapports cycliques différents.

Une fois le signal généré, il va nous falloir le transformer en signal analogique, pour qu'il puisse être envoyé via un canal de transmission. Sur une période d'un signal périodique, on peut calculer sa valeur moyenne. En fait, il faut faire une moyenne de toutes les valeurs que prend le signal pendant ce temps donné. De ce fait, si on modifie le rapport cyclique de la PWM de façon maitrisée, on va pouvoir créer un signal analogique de la forme que l'on souhaite, compris entre 0 et 5V, en extrayant la valeur moyenne du signal.

Pour extraire la valeur moyenne du signal de la PWM, il faut utiliser les propriétés du couple RC ou résistance-condensateur**.** On peut déterminer le temps de charge et de décharge du condensateur à partir d'un paramètre $\tau = R.C$

$$\tau : temps\ de\ charge/décharge\ en\ (s)$$

$$R : valeur\ de\ la\ résistance\ en\ (Ohm)$$

$$C : valeur\ de\ la\ capacité\ du\ condensateur\ en\ (Farad)$$

À chaque fois que le signal de la PWM sera au niveau logique 1, le condensateur va se charger. Dès que le signal repasse au niveau logique 0, le condensateur va se décharger. Et ainsi de suite. En somme, cela donne une variation de tension aux bornes du condensateur semblable à celle-ci:

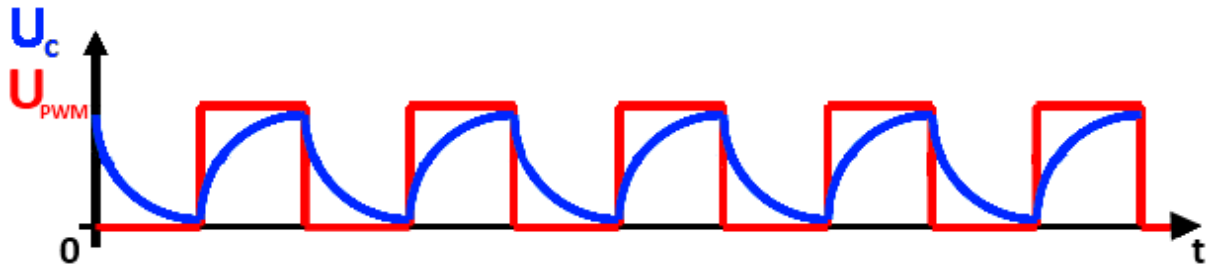



**Figure.61 :** Charge/décharge d'un condensateur sur une sortie PWM.

Une fois l'émetteur Arduino programmé et ses sorties PWM reliées aux couples RC (*R=10 kOhm, C=0.1 n.F*), on visualise les états x et z du système, à l'aide d'un oscilloscope TEKTRONIX TDS1012.

- *Etat x, sur CH1 :*

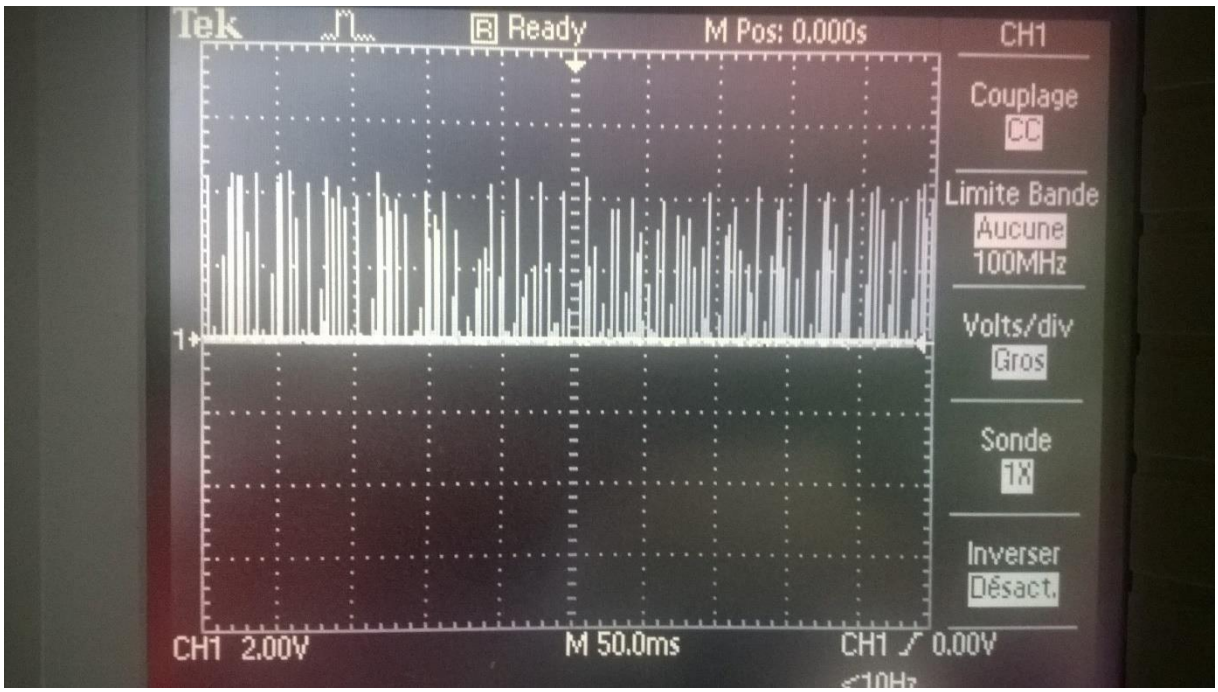



**Figure.62 :** Visualisation de l'état x du système de Hénon modifié.

- Etat z, sur CH2 :

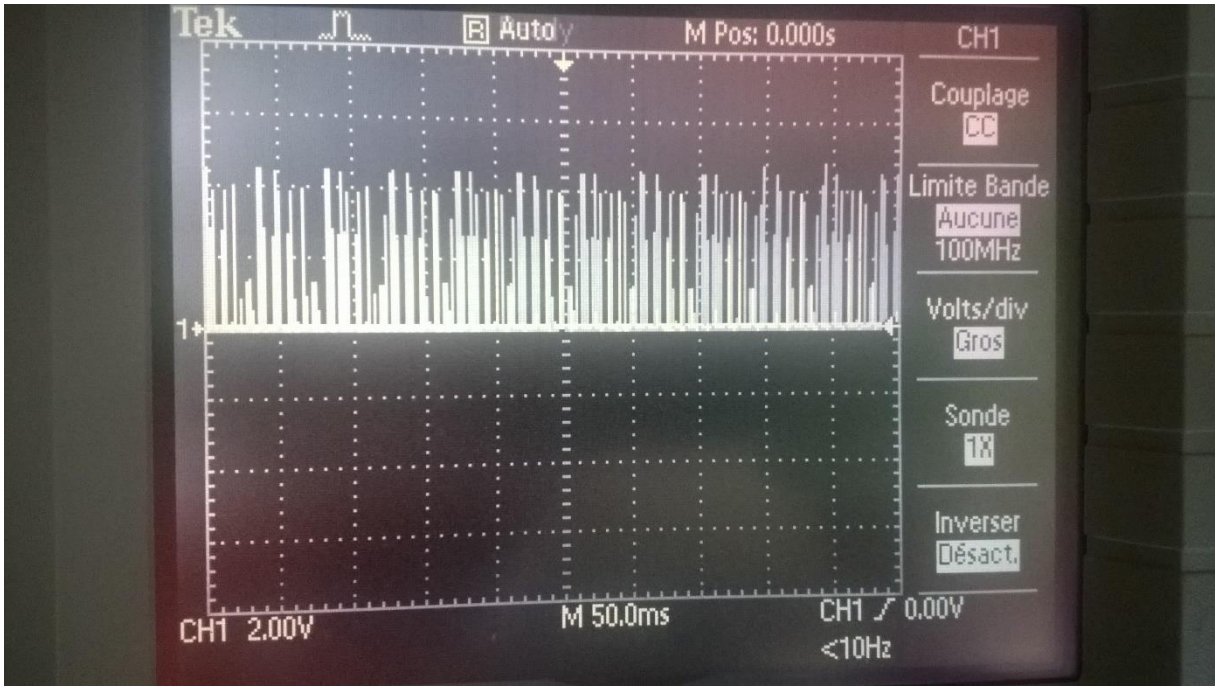



**Figure.63 :** Visualisation de l'état z du système de Hénon modifié.

En visualisant les états du système, on constate un comportement qui semble aléatoire. Cet aspect aléatoire est typique des signaux chaotiques.

- *Visualisation de la transformée de Fourier discrète (état z) :*

La FFT permet de visualiser les spectres continus à large bande qui caractérisent les signaux chaotiques.

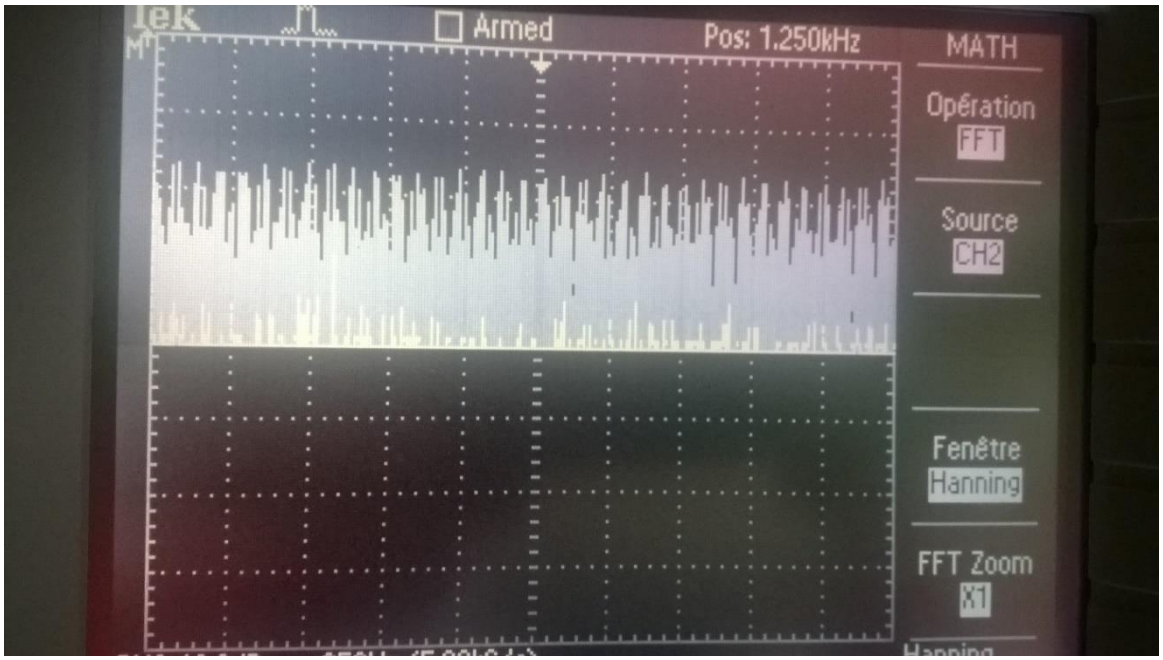



**Figure.64 :** Visualisation de la transformée de Fourier discrète de l'état z.

- Visualisation de la transformée de Fourier discrète pour l'état x :

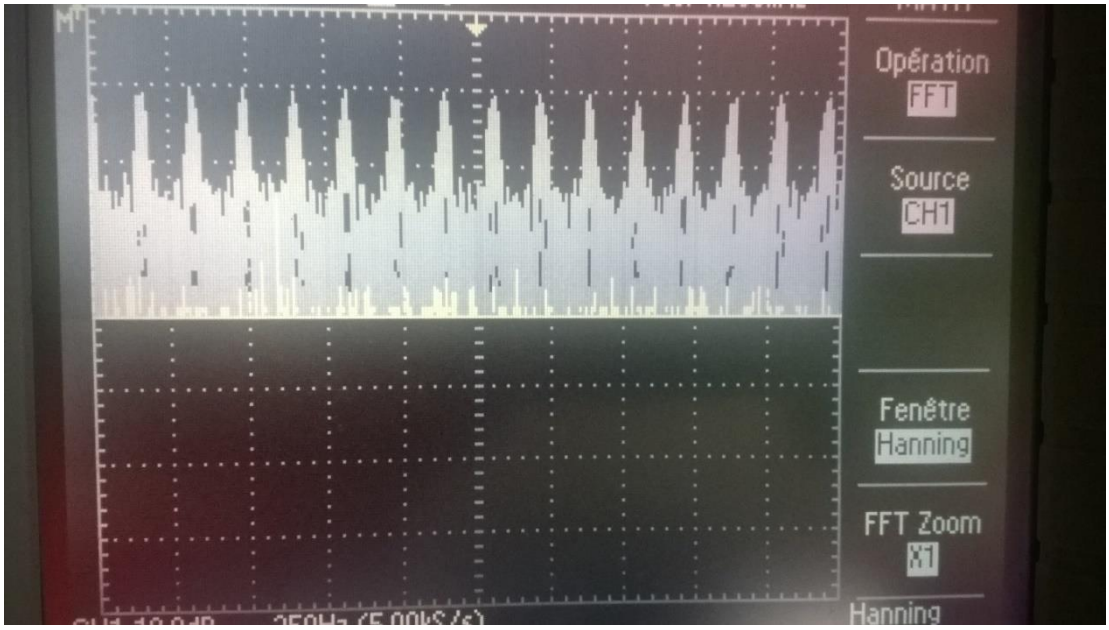



**Figure.65 :** Visualisation de la transformée de Fourier discrète de l'état x.

En visualisant la TFD des états du système, on trouve des trajectoires apériodiques et de toutes les périodes. Nous avons donc affaire à un large spectre continu de fréquences. Ceci engendre un spectre continu et irrégulier, qui peut parfois présenter des pics (relatifs au doublement de période). Ce sont les spectres continus à large bande qui caractérisent les signaux chaotiques.

- Attracteur chaotique :

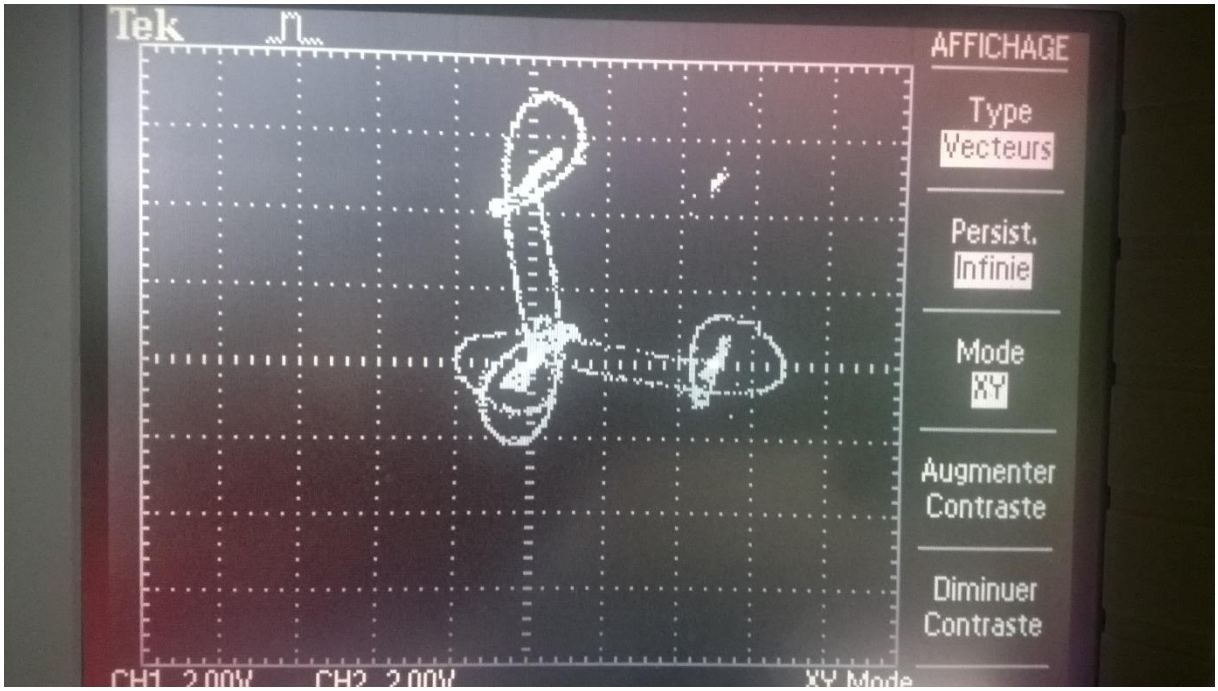



**Figure.66 :** Visualisation de l'attracteur chaotique (mode x-y).

La courbe visualisée ne repasse jamais par les mêmes points, et évolue toujours dans un espace délimité, puis finit par décrire une figure géométrique particulière qui représente son attracteur étrange.

- Programmes :
  1. IDE Arduino :

```
const float a=1.76;
const float b=0.1;
const int NbrPINs = 4;
const int mypin[] ={9,3,5,6};
double N=100;
float xN0=0.9; //0 < x0 < 1
float yN0=0.1;
float zN0=0.1;
float uN0=0;
float sN0=0.1;
float xN=xN0;
```

```
float yN=yN0;
float zN=zN0;
float uN=uN0;
float sN=sN0;
void setup(){
  Serial.begin(9600);
     for (int k=0;k<N;k++){
      pinMode(mypin[1,2,3],INPUT);
      delay (500);
           pinMode(mypin[0],OUTPUT);}
}
void loop()
{
 for(int k=0;k<N;k++){
  xN0=xN;
  yN0=yN;
  zN0=zN;
  sN0=sN;
  xN=a-(yN0*yN0)-b*zN0;
  yN=xN0;
  zN=yN0;
  sN0=yN0;
 tone(2,1);
 analogWrite((3),xN);
 analogWrite((5),yN);
 analogWrite((6),zN);
 analogWrite((9),sN0);
}}
```

2. Matlab Simulink :

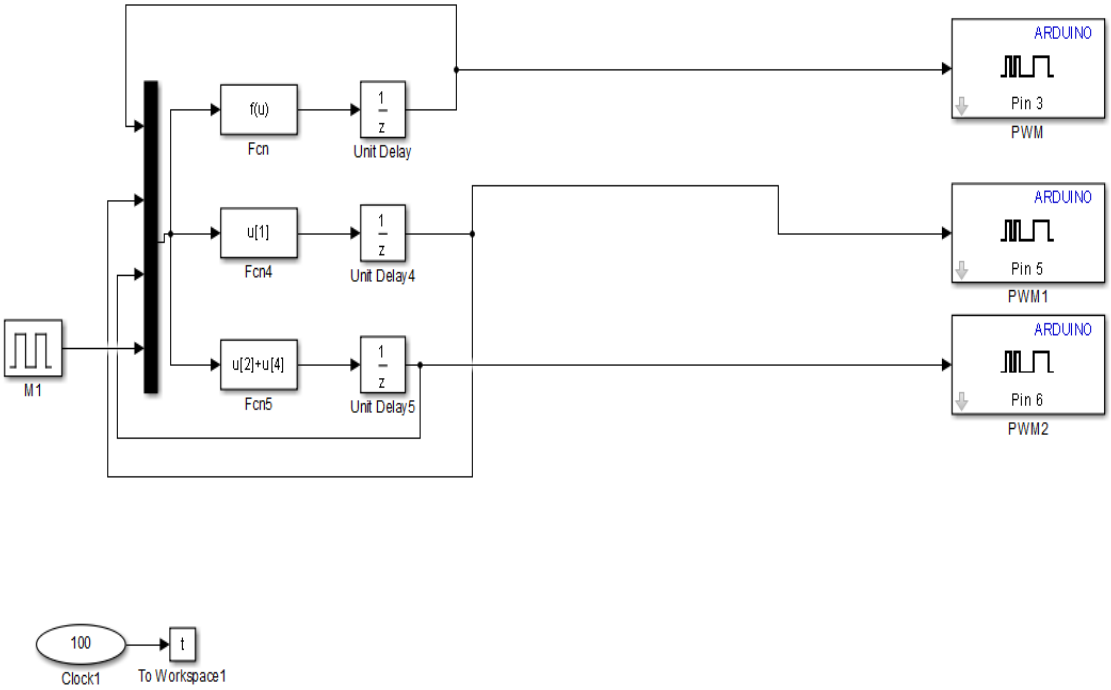



**Figure.67 :** Programme Simulink de l'émetteur.

## 4.5. Conclusion :

En début de chapitre, une présentation de la carte Arduino Uno et de son Microcontrôleur ATMEL ATMega328 a été faite. Les étapes d'installations et de mise en fonctionnement de la carte ont été données. La programmation de la carte à l'aide des logiciels Matlab et IDE Arduino a été expliquée à l'aide d'exemples. En fin de chapitre, la carte Arduino a été programmée en tant qu'émetteur hyper chaotique. Sur l'oscilloscope, nous avons pu visualiser les états du système et avons constaté un comportement semblant aléatoire qui est typique des signaux chaotiques, avec une fréquence pwm qui vaut 490Hz. De même, en visualisant la transformée de Fourier discrète de chaque état du système, celle-ci nous donne des spectres continus à large bande qui caractérisent les signaux chaotiques. Nous avons aussi pu visualiser en mode x-y une courbe qui ne repasse jamais par les mêmes points, qui évolue dans un espace délimité pour décrire enfin une figure géométrique qui représente l'attracteur étrange du système hyper chaotique.

# CONCLUSION GENERALE

Ce mémoire nous a conduits à étudier les systèmes dynamiques chaotiques, et à mettre en pratique ces notions en programmant un émetteur chaotique sur une carte Arduino-Uno.

Notre travail présente quelques notions sur le chaos, ainsi que quelques définitions importantes. Les deux classes de systèmes chaotiques et leurs propriétés sont présentées à l'aide d'exemples simulés sous logiciel Matlab. Une brève explication des différents scénarios vers le chaos est donnée à la fin du premier chapitre, en expliquant l'intermittence, la quasi-périodicité et le doublement de périodes. Toutes ces notions sont exploitées lorsque le phénomène de synchronisation des systèmes chaotiques est abordé. Nous avons introduit la notion de communication sécurisée à base du chaos, les différentes méthodes de synchronisation et schémas de principe, ainsi que les techniques de transmission sécurisée d'informations qui reposent sur le principe de synchronisation chaotique. Nous avons aussi passé en revue les techniques de cryptage à base du chaos.

Nous avons, ensuite, introduit un nouveau schéma de transmission de données basé sur la synchronisation de deux systèmes chaotiques, celui-ci est simulé sous logiciel Matlab. Le choix de l'émetteur s'est porté sur le système hyper chaotique à temps discret de Hénon-Heiles, tandis que le récepteur n'est autre qu'un observateur discret retardé qui est choisi afin de récupérer les états ainsi que le message du système de Hénon modifié. Le choix de l'observateur s'est fixé après vérification de trois conditions qui permettent la conception de celui-ci.

Le principe de la méthode de transmission est d'inclure le signal du message à envoyer par inclusion dans le modèle de Hénon modifié. Ceci consiste en l'ajout du message dans l'une des dynamiques de l'émetteur. Des figures illustrant la synchronisation ainsi que les erreurs de synchronisation de tous les états et du message ont été données avec illustration des résultats de simulation. Les résultats de simulation montrent les performances du système de transmission proposé. La synchronisation des deux systèmes s'est produite, et les différents signaux ont été récupérés au niveau du récepteur.

En plus, dans ce travail, nous avons mis en pratique l'émetteur hyper chaotique simulé précédemment sous logiciel Matlab. Cet émetteur est implémenté dans une carte Arduino-Uno qui fonctionne avec le microcontrôleur ATMEL ATMega328.

La programmation de l'émetteur se fait sur logiciels Matlab Simulink, ainsi que sur le logiciel IDE Arduino. Les étapes d'installation et de mise en fonctionnement de la carte ont été données, ainsi que les différentes figures montrant le comportement chaotique de l'émetteur.

La fréquence du signal PWM vaut environ 490Hz pour la plupart des sorties PWM et 980Hz pour les sorties 5 et 6. Ces fréquences peuvent être reprogrammées, mais comme la fréquence d'oscillation du système de Hénon modifié est de 270Hz, il n'est pas nécessaire de reprogrammer les fréquences PWM.

En perspective, nous comptons réaliser la partie récepteur afin de récupérer l'information cryptée par l'émetteur, et ainsi implémenter le système de transmission à base de deux cartes Arduino, dans des protocoles de transmissions Bluetooth, Xbee, etc.

1) Définitions :

- Système dynamique autonome :

Un système dynamique est dit autonome, si sa loi d'évolution ne dépend pas du temps (la loi est alors dite stationnaire). Ainsi, un circuit autonome est un circuit qui produit une sortie qui varie dans le temps, sans avoir une entrée qui varie dans le temps.

- Exposants conditionnels de Lyapunov (CLE) :

Ce terme est généralement utilisé en rapport avec la synchronisation du chaos, dans laquelle deux systèmes sont couplés, habituellement de façon unidirectionnelle, de manière à avoir un système de commande (maître) et un système de réponse (esclave). Les exposants conditionnels sont les exposants de Lyapunov du système esclave lorsque le système maître est considéré comme étant une simple source de signaux de commande chaotiques. Ces exposants sont donc dits « conditionnels » car ils dépendent du signal de commande. Ainsi, la synchronisation se produit lorsque tous les exposants conditionnels de Lyapunov dont tous négatifs.

2) Présentation du circuit de Chua :

Un circuit électronique doit respecter certaines conditions pour montrer un comportement chaotique, appelés critères chaotiques. Il doit contenir :

- Un élément non linéaire ou plus.
- Une résistance localement active ou plus.
- Trois éléments de stockage d'énergie ou plus (condensateur, inductance,...).

En 1983, l'ingénieur Leon Ong Chua a mis au point le plus simple circuit électronique respectant ces critères. Il comporte deux condensateurs, une bobine, une résistance active et une diode de Chua. Le circuit de Chua est devenu un modèle standard pour l'étude du chaos dans les systèmes décrits par des équations différentielles à dimension finie. Ses équations d'état sont :

$$\begin{cases} \dot{x} = \alpha(y - x - f(x)) \\ \dot{y} = x - y + z \\ \dot{z} = -\beta y - \gamma z \end{cases}$$

Où f(x) est une fonction non linéaire par morceaux qui traduit la caractéristique de la diode de Chua, qui est donnée par :

$$f(x) = bx + 1/2(a - b)(|x + 1| - |x - 1|)$$

Où $a < b < 0$ sont deux constantes. Les états du système sont tous mesurables.

Exemple de circuit de Chua :

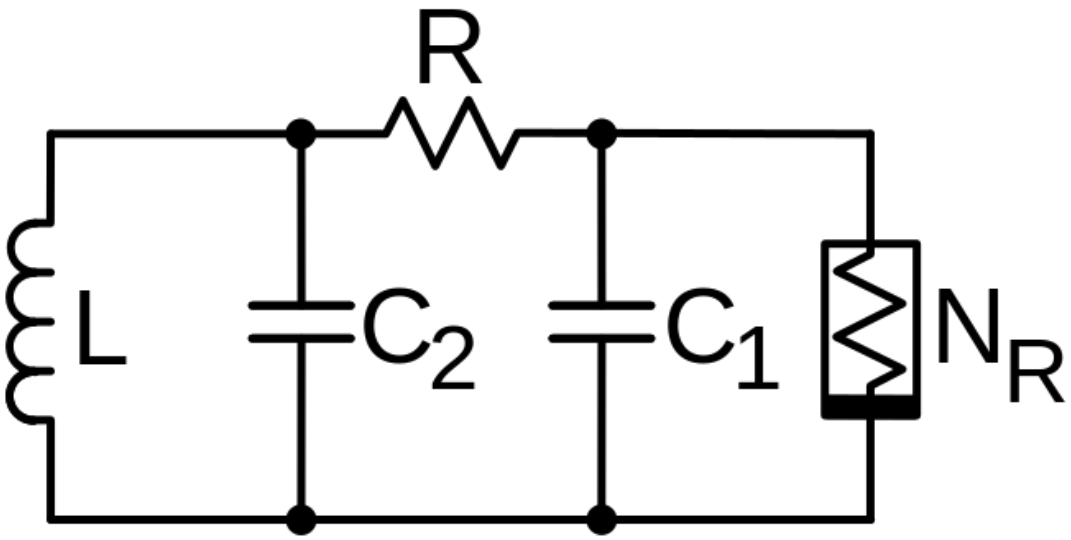



**Figure A.1 :** Exemple de circuit de Chua.

Le composant $N_R$ est une résistance négative non linéaire, appelée diode de Chua. Elle est, habituellement, construite à partir d'un circuit contenant un amplificateur à rétroaction positive.

Diode de Chua :

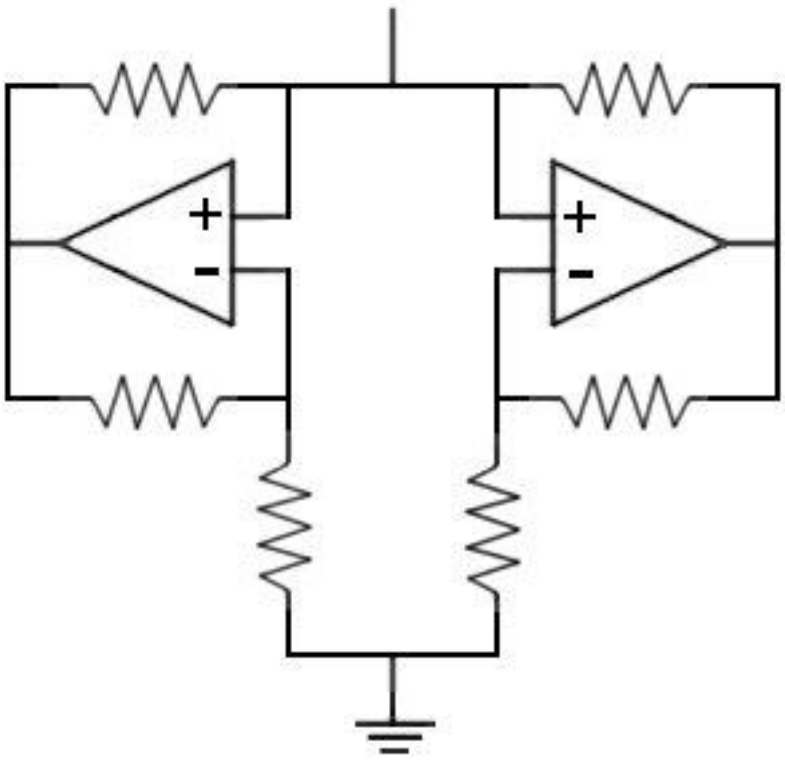
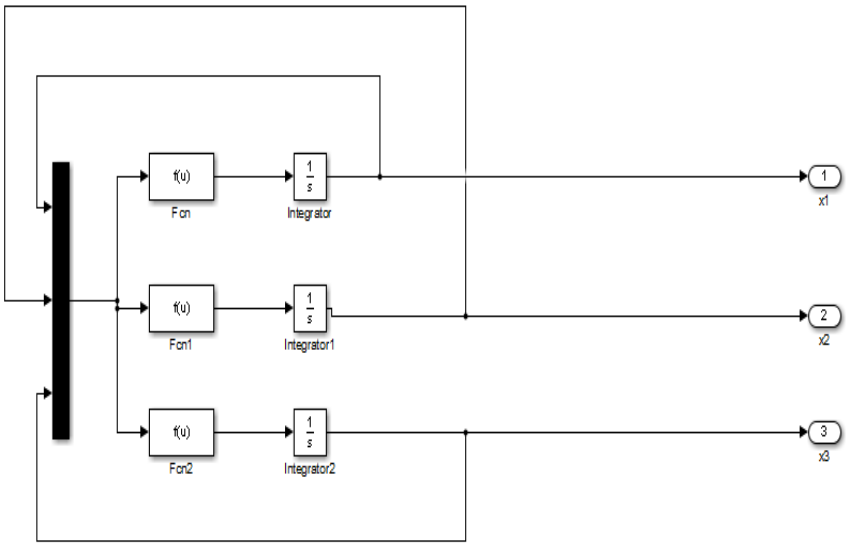


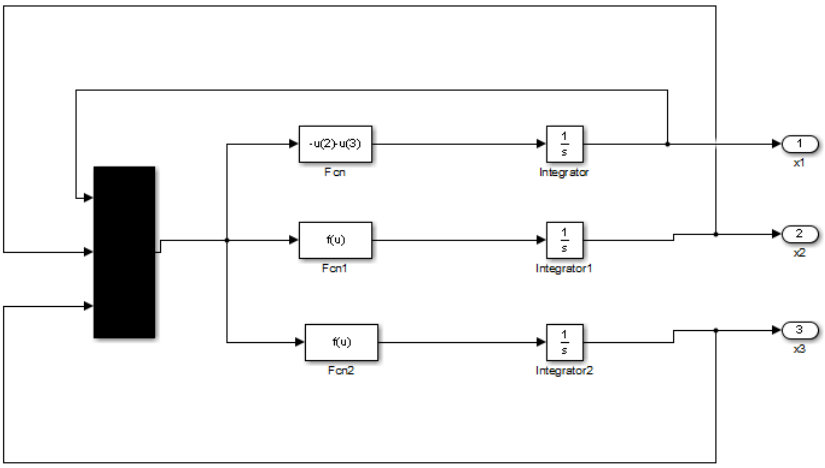

**Figure A.2** : Diode de Chua.

3) Programmes Matlab Simulink :

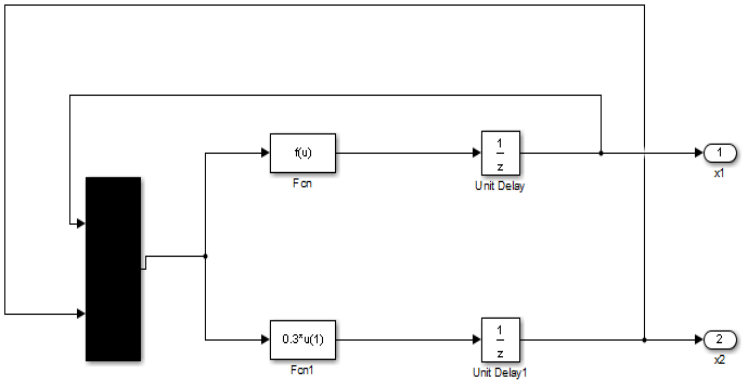
- Système de Lorenz :




- Système de Rössler :



- Système de Hénon :



# LISTE DES FIGURES

# LISTE DES FIGURES

**Liste des tableaux :**

[1] S.Sastry « Nonlinear Systems » Edition Springer, New York, 1999.

[2] Christian Jutten « Systèmes asservis non linéaires » Cours de troisième année du département 3i Options Automatique. Université Joseph Fourier - Polytech' Grenoble. 2006.

[3] G.Kaddoum « Contributions à l'amélioration des systèmes de communication multi-utilisateurs par chaos : synchronisation et analyse des performances ». Thèse de Doctorat de l'Université de Toulouse, 2008.

[4] E.Goncalvès « Introduction aux systèmes dynamiques et chaos.» Cours de l'Institut National Polytechnique de Grenoble, 2004.

[5] D.Viennot « Analyse Spectrale pour les Systèmes Dynamiques Classiques.». Cours Master Physique & Physique Numérique, Université de Franche-Comté.

[6] C.Morel « Analyse et contrôle de dynamiques chaotiques, application à des circuits électroniques non-linéaires.». Thèse de Doctorat de l'École Doctorale d'Angers. 2005.

[7] A. Oustaloup, J. Sabatier, and P. Lanusse. « From fractal robustness to the crone control » Fractional Calculus and Applied Analysis, 2, 1999.

[8] C. Ramus-Serment, X. Moreau, M. Nouillant, A. Oustaloup, and F. Levron. « Generalised approach on fractional response of fractal networks », Chaos Solitons & Fractals, 2002.

[9] G.Zaïbi **«** Sécurisation par dynamiques chaotiques des réseaux locaux sans fil au niveau de la couche MAC », Thèse de Doctorat de l'Université de Toulouse, 2012.

[10] M.Inoue and H.Kamifukumoto « Scenarios Leading to Chaos in a Forced Lotka-Volterra Model », Progress of Theoretical Physics, Vol. 71, No.5, May 1984.

[11] J.Malek « La Théorie du Chaos en Finance : Une application économétrique », Mémoire de Licence, Ecole de commerce Solvay, ULB, 1995.

[12] A.B.Zer & E.Akin « Tools For Detecting Chaos», Institut Des Sciences Et Technologies, Université Sakarya, Journal 9.Cilt, 1.Say1, Turquie, 2005.

M. Hénon & C. Heiles, « The Applicability of the Third Integral Of Motion: Some Numerical Experiments, » The Astrophysical Journal, 69 (1964), 73-79.

[13] : S.Penaud « Etude des Potentialités du Chaos Pour les Systèmes de Télécommunications, Evaluation des Performances de Systèmes à Accès Multiples à Répartition Par Les Codes (CDMA) Utilisant Des Séquences D'Etalement Chaotiques.», Thèse de Doctorat de l'Université de Limoges, 2001.

[14] L.M.Pecora, T.L.Carroll « Synchronization in Chaotic Systems », Physical Review Letters, Vol 64 n°8, 1990.

[15] I.Ameur « Synchronisation, Chaotification et Hyperchaotification des Systèmes Non-linéaires : Méthodes et Applications », Thèse de Doctorat de l'Université Mentouri de Constantine, 2011.

[16] T.Hoet, B.Lorenzi, S.Sahin « la cryptographie chaotique », Mémoire de Licence IMACS INSA Toulouse, 2012.

[17] O.Megherbi « Etude et réalisation d'un système sécurisé à base de systèmes chaotiques ». Mémoire de Magister en Automatique, Université Mouloud Mammeri de Tizi-Ouzou, 2013.

[18] Mihai Bogdan Luca « Apports du chaos et des estimateurs d'états pour la transmission sécurisée de l'information », Thèse de Doctorat de l'Université de Bretagne Occidentale, 2006.

[19] M. Hassler and T.Schimming. « Communications using chaos. » Int. Conf. On Signals and Electronic Systems, 2001.

[20] M.Abramowitz, I.A. Stegun « Handbook of mathematical functions with Formulas, Graphs and Mathematical tables », Dover Publications, Inc, New York, 1965.

[21] G.R.Cooper, R.W. Nettleton « spread spectrum technique for gigh capacity mobile communications », IEEE Trans. Veh. Tech, Vol. VT-27. 1978.

[22] G.R.Cooper, R.W. Nettleton « Spectral efficiency in cellular land-mobile communications: a spread spectrum approach », Final Report, TR-EE 78-44, Purdue University, West Lafayette, Ind. 1978.

[23] H.Dimassi « Synchronisation des systèmes chaotiques par observateurs et applications à la transmission d'informations », Thèse de Doctorat de l'Université de Paris Sud 11, 2012.

[24] H.Nijmeijet and I.Mareels. « An observer looks at synchronization. » IEEE Trans. On Circ. Syst. I : Fundamental Theory and Applications, 44(10) :882-890, 1997.

[25] H.Hamiche « Inversion à Gauche des Systèmes Dynamiques Hybrides Chaotiques, Applications à la Transmission Sécurisée de Données. » Thèse de Doctorat, Université Mouloud Mammeri de Tizi-Ouzou, 2011.

[26] K. Veselyand J. Podolsky, « Chaos in a modified Henon- Heiles system describing geodesics in gravitational waves, » Tech.Phys. Letters A, vol. 271, p 368–371, July 2000.

[27] M. Djemaï, J-P Barbot and I. Belmouhoub, « Discrete-Time Normal Form for Left Invertibility problem, » Eur. J. Control, vol. 15, p. 194–204, 2009.

[28] I. Belmouhoub, M. Djemaï and J.P. Barbot, « Observability quadtatic Normal Form for Discrete-Time systems », IEEE Transactions on Automatic Control, vol. 50, july 2005.

[29] M. Djemaï, J.P. Barbot, I. Belmouhoub, « Discrete-Time Normal Form for Left Invertibility Problem », European Journal of Control, vol. 15, pp. 194-204, 2009.

[30] L. M. Pecora and T. L. Caroll, « Synchronization in chaotic systems », Phys. Rev. Lett., vol. 64, pp. 973977, 1992.

[31] L.Cong and W. Xiaofu, « Design and realization of an fpga- based generator for chaotic frequency hopping sequences », IEEE Transactions on circuits and systems-I Fundamental theory and applications, vol. 48, pp. 521532, 2001.

[32] DataSheet ATMEGA328.

**Ressources Internet:**

http://just.loic.free.fr/index.php?page=elem

http://fr.questmachine.org/wiki/La_th%C3%A9orie_du_chaos

http://www.mathworks.com/matlabcentral/fileexchange/233-let

http://www.arduino.cc/