

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO



PROGRAMA DE POSGRADO

MAESTRÍA Y DOCTORADO EN CIENCIAS E INGENIERÍA

**IMPLEMENTACIÓN DE ENCRIPADO CAÓTICO
DE SEÑALES BIOMÉDICAS EN SISTEMA EMBEBIDO
PARA TELEMEDICINA SEGURA**

TESIS

que para cubrir parcialmente los requisitos necesarios para obtener el grado de

MAESTRO EN INGENIERÍA

presenta:

DANIEL MURILLO ESCOBAR

Ensenada, Baja California, México, Agosto de 2020.

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO

IMPLEMENTACIÓN DE ENCRİPTADO CAÓTICO
DE SEÑALES BIOMÉDICAS EN SISTEMA EMBEBIDO
PARA TELEMEDICINA SEGURA

TESIS

Que para obtener el grado de Maestro en Ingeniería presenta:

DANIEL MURILLO ESCOBAR

Aprobada por el siguiente comité:



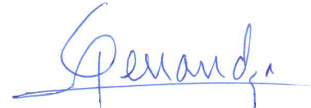
Dra. Rosa Martha López Gutiérrez
Directora



Dr. Miguel Ángel Murillo Escobar
Codirector



Dra. Liliana Cardoza Avendaño
Miembro del comité



Dr. César Cruz Hernández
Miembro del comité



Dr. Adrián Arellano Delgado
Miembro del comité

Ensenada, Baja California, México, Agosto de 2020.

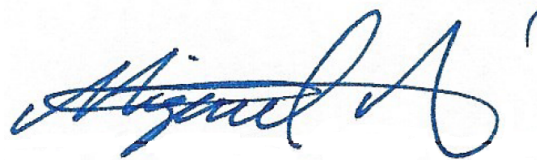
RESUMEN de la tesis de **Daniel Murillo Escobar**, presentada como requerimiento para obtener el grado de MAESTRO EN INGENIERÍA en ELÉCTRICA, del programa Maestría y Doctorado en Ciencia e Ingeniería de la Universidad Autónoma de Baja California. Ensenada, Baja California, México, agosto de 2020.

**IMPLEMENTACIÓN DE ENCRİPTADO CAÓTICO
DE SEÑALES BIOMÉDICAS EN SISTEMA EMBEBIDO
PARA TELEMEDICINA SEGURA**

Resumen aprobado por:



Dra. Rosa Martha López Gutiérrez
Directora



Dr. Miguel Ángel Murillo Escobar
Codirector

En este trabajo de tesis de maestría, se implementa en un sistema embebido, un algoritmo criptográfico basado en caos para brindar confidencialidad a señales biomédicas como el electrocardiograma para aplicaciones en telemedicina segura.

Se analizan nueve mapas caóticos y se determina utilizar el mapa Badola en el algoritmo criptográfico, ya que es un mapa hypercaótico con solo 2 dimensiones, requiere pocas operaciones aritméticas, amplio espacio de clave y tiene costo de implementación bajo. Además, se realiza una mejora al mapa para obtener mayores propiedades de aleatoriedad y aumentar la seguridad.

Finalmente, el encriptado caótico propuesto se implementa en un sistema embebido basado en un microcontrolador de 32 bits, pantalla de cristal líquido, botones de presionar y fuente de alimentación, donde la señal clara de electrocardiograma se obtiene mediante electrodos conectados a un módulo que trabaja con el microcontrolador. El sistema criptográfico es sometido a distintos análisis de seguridad como espacio de claves, sensibilidad a clave secreta, sensibilidad a señal clara, histogramas, correlación, entropía de la información y tiempo de encriptado. Los resultados muestran que el algoritmo de encriptado caótico propuesto es resistente a este tipo de ataques y puede ser aplicado en telemedicina segura a un bajo costo de implementación.

Palabras clave: caos, telemetría, encriptado caótico, señales biomédicas, microcontrolador, mapa badola, análisis de seguridad.

Abstract of the thesis presented by **Daniel Murillo Escobar**, as a requirement to obtain the MASTER IN ENGINEERING degree in ELECTRIC, of the program of Master and Doctorate in Science and Engineering of the Autonomous University of Baja California. Ensenada, Baja California, Mexico, august 2020.

IMPLEMENTATION OF CHAOTIC ENCRYPTION OF BIOMEDICAL SIGNALS IN EMBEDDED SYSTEM FOR SECURE TELEMEDICINE

Abstract approved by:



Dra. Rosa Martha López Gutiérrez

Directora



Dr. Miguel Ángel Murillo Escobar

Codirector

In this master's thesis work, a chaos-based cryptographic algorithm is implemented in an embedded system to provide confidentiality to biomedical signals such as the electrocardiogram for applications in secure telemedicine.

Nine chaotic maps are analyzed and it is determined to use the Badola map in the cryptographic algorithm, since it is a hyperchaotic map with only 2 dimensions, requires few arithmetic operations, a large key space and has a low implementation cost. In addition, an improvement is made to the map to enhance randomness and increase security.

Finally, the proposed chaotic encryption is implemented in an embedded system based on a 32-bit microcontroller, liquid crystal display, push buttons, and power supply, where the clear electrocardiogram signal is obtained through by using electrodes and a module that works with the microcontroller. The cryptographic system is subduced an analysis of security such as key space, secret key sensitivity, clear signal sensitivity, histograms, correlation, information entropy, and encryption time. The results show that the proposed chaotic encryption algorithm is resistant common attacks and can be applied in secure telemedicine at a low cost of implementation.

Keywords: chaos, telemetry, chaotic encryption, biomedical signals, microcontroller, badola map, security analysis.

A mi familia

Agradecimientos

A mis padres, Miguel Ángel y María Rosario. Por su apoyo durante estos 2 años y toda mi vida para la realización de este trabajo el cual sin ustedes no podría haberlo realizado.

A mis hermanos, Miguel, Yahaira, Angelina y Araceli. Por su cariño y buenos deseos que me alentaron a seguir adelante. En especial a mi hermano Miguel por sus consejos y apoyo brindado para la realización de este trabajo de maestría.

A mis sobrinos, Milton, David, Sofía y Ximena. Por darme alegría cada día.

A mi novia, Diana. Por estar siempre apoyándome en todo lo que hago en los últimos años. Por su amor y paciencia que me llena de felicidad.

A mis codirectores de tesis, Dra. Rosa Martha López Gutiérrez y Dr. Miguel Ángel Murillo Escobar, por dirigirme durante mis estudios de maestría y sus consejos para la culminación de mis estudios de maestría.

A mi comité de tesis, Dra. Liliana Cardoza Avendaño, Dr. César Cruz Hernández y Dr. Adrián Arellano Delgado, por sus consejos y observaciones durante el desarrollo de mis estudios de maestría para mejorar la calidad del trabajo.

Al grupo de investigación Sistemas Complejos de la FIAD, por su amistad, los consejos y los buenos momentos que pasamos juntos.

A mis amigos, Alexis, Uriel, Alma, Karla, Atondo, Loya, Rolando, Carlitos, Riaño y José. Por su amistad y el apoyo que me brindaron cuando lo necesité.

A la Universidad Autónoma de Baja California, por brindarme un espacio íntegro donde realizarme profesionalmente, por permitirme trabajar en el laboratorio de sistemas complejos el tiempo necesario para la realización de este trabajo. En especial a la Facultad de Ingeniería, Arquitectura y Diseño Ensenada.

Al Consejo Nacional de Ciencia y Tecnología (CONACyT), por el apoyo económico brindado para la realización de este trabajo.

Ensenada, Baja California, México, Agosto de 2020.

Daniel Murillo Escobar

Tabla de Contenido

Resumen	I
Abstract	II
Agradecimientos	IV
Lista de Figuras	VIII
Lista de Tablas	XI
1. Introducción	1
1.1. Motivación	4
1.2. Objetivos y alcances de la tesis	4
1.3. Organización del manuscrito	5
2. Telemedicina	6
2.1. Introducción	6
2.2. Características de la telemedicina	8
2.3. Electrocardiografía (ECG)	10
2.4. Seguridad en la telemedicina	11
2.5. Conclusiones	12
3. Caos	13
3.1. Introducción	13
3.2. Sistemas caóticos y sus propiedades	16
3.3. Exponente de Lyapunov	18
3.4. Mapas caóticos estudiados	18
3.4.1. Mapa Logístico	19
3.4.2. Mapa Chebyshev	20
3.4.3. Mapa Seno	20
3.4.4. Mapa Logístico 2D	21
3.4.5. Mapa Hénon	22
3.4.6. Mapa Seno-Logístico 2D	24
3.4.7. Mapa Badola	24
3.4.8. Mapa Ushio	25
3.4.9. Mapa Lotka-Volterra	27

3.5.	Selección de mapa caótico	28
3.5.1.	Espacio de claves en función de los parámetros de control y condiciones iniciales	28
3.5.2.	Tiempo de procesamiento	29
3.5.3.	Exponente de Lyapunov	30
3.5.4.	Máxima frecuencia de datos caóticos	30
3.5.5.	Mapa caótico seleccionado	31
3.6.	Conclusiones	33
4.	Criptografía	34
4.1.	Introducción	34
4.2.	Sistemas criptográficos	35
4.3.	Características de la criptografía	39
4.4.	Seguridad criptográfica	40
4.5.	Conclusiones	42
5.	Sistemas embebidos	43
5.1.	Introducción	43
5.2.	Microcontrolador	45
5.3.	Microcontrolador MCF51MM256	46
5.4.	Módulo de ECG	48
5.5.	Software CodeWarrior	50
5.6.	Conclusiones	50
6.	Algoritmo de encriptado caótico para señales biomédicas	51
6.1.	Introducción	51
6.2.	Algoritmo de encriptado caótico	54
6.2.1.	Definición de la clave secreta	54
6.2.2.	Cálculo de Z	54
6.2.3.	Proceso de encriptado	55
6.2.4.	Proceso de desencriptado	56
6.2.5.	Características de seguridad y eficiencia	56
6.3.	Conclusiones	57
7.	Implementación de encriptado caótico en microcontrolador	58
7.1.	Introducción	58
7.2.	Revisión de la literatura	59
7.3.	Resultados experimentales	59
7.3.1.	Adquisición de la señal clara ECG	61
7.4.	Sistema embebido	65
7.5.	Análisis de seguridad	73
7.5.1.	Espacio de claves	73
7.5.2.	Sensibilidad a la clave secreta	73
7.5.3.	Sensibilidad a la señal clara	75
7.5.4.	Histogramas	76

7.5.5. Correlación	77
7.5.6. Entropía de la información	77
7.5.7. Recursos de implementación y desempeño	78
7.6. Conclusiones	78
8. Conclusiones	79
8.1. Conclusiones generales	79
8.2. Trabajo a futuro	79
Bibliografía	80

Lista de Figuras

1.1. La telemedicina alude al uso de las tecnologías de comunicación.	1
1.2. La información en la telemedicina debe ser privada.	2
1.3. Estado x del sistema caótico de Lorenz.	3
2.1. La telemedicina permite la transmisión de información médica a distancia.	6
2.2. Comunicación en modo síncrono: Teleconferencia.	7
2.3. Comunicación en modo asíncrono: Correo electrónico.	8
2.4. Imagen de un monitor médico donde se visualizan los signos vitales a distancia.	9
2.5. Señal ECG.	10
2.6. Triangulo de Einthoven.	10
2.7. Ataques y vulnerabilidades en telemedicina.	11
3.1. Estado x del sistema de Lorenz con pequeñas variaciones en las condiciones iniciales.	14
3.2. Atractor extraño de Lorenz.	16
3.3. Estado x del mapa Logístico con dinámicas caóticas discretas.	19
3.4. Estado x del mapa Chebyshev con dinámicas caóticas discretas.	20
3.5. Estado x del mapa Seno con dinámicas caóticas discretas.	21
3.6. Estado x del mapa Logístico 2D con dinámicas caóticas discretas.	22
3.7. Estado x del mapa Hénon con dinámicas caóticas.	23
3.8. Atractor extraño del mapa Hénon.	23
3.9. Estado x del mapa Seno-Logístico 2D con dinámicas caóticas discretas.	24
3.10. Estado x del mapa Badola con dinámicas caóticas discretas.	25
3.11. Estado x del mapa Ushio con dinámicas caóticas discretas.	26
3.12. Atractor extraño del mapa Ushio.	26
3.13. Estado x del mapa Lotka-Volterra con dinámicas caóticas discretas.	27
3.14. Formato binary 32.	28
3.15. Distribución de 1000 datos del mapa Badola original.	32
3.16. Distribución de 1000 datos del mapa Badola mejorado.	32
4.1. El objetivo de la criptografía es transmitir información de forma segura.	35
4.2. Diagrama de un sistema criptográfico.	36
4.3. Diagrama de encriptado simétrico.	37
4.4. Diagrama de encriptado asimétrico.	38

4.5. Anonymous es el grupo de criptoanalistas con más ataques realizados en los últimos años.	41
5.1. Diagrama de un sistema embebido a nivel físico.	44
5.2. Torre TWR-MCF51MM utilizada para trabajar con el microcontrolador MCF51MM256.	47
5.3. Microcontrolador de 32 bits MCF51MM256 utilizado en el trabajo experimental de este trabajo de tesis.	47
5.4. Diagrama a bloques del microcontrolador MCF51MM256.	48
5.5. Módulo MED-ECG.	49
5.6. Diagrama a bloques del módulo MED-ECG.	49
5.7. Interfaz del software CodeWarrior.	50
6.1. Estructura de un algoritmo.	51
6.2. Seguridad médica en redes de telemedicina.	52
6.3. Diagrama a bloques del proceso de encriptado.	53
6.4. Diagrama a bloques del proceso de desencriptado.	53
7.1. Sistema para obtener la señal ECG: a) Módulo MED-ECG, b) Microcontrolador MCF51MM256 y c) Torre TWR-MCF51MM.	60
7.2. Material para obtención de la señal ECG: a) Electrodo médico y b) Cables aislados.	61
7.3. Ejemplo de conexión para obtener señal de ECG.	61
7.4. Conexión del sistema para obtención de señal ECG.	62
7.5. Interfaz de visualización para la señal ECG en computadora personal.	62
7.6. Señal clara de electrocardiograma utilizada en el proceso de encriptado.	63
7.7. Señal encriptada de electrocardiograma.	63
7.8. Diagrama a bloques del sistema embebido.	66
7.9. Pantalla LCD 20x4.	67
7.10. Función de los LED's del mando de control: a) LED Amarillo indica que el proceso se está ejecutando, b) LED Rojo indica inicio del proceso y c) LED Verde indica fin del proceso.	67
7.11. Sistema embebido utilizado.	68
7.12. Mensajes de inicio del sistema: a) Mensaje al encender la LCD, b) Mensaje al iniciar el programa y c) Mensaje de inicio de sistema.	69
7.13. Mensajes para la obtención de la señal de ECG: a) Mensaje: Obteniendo señal, b) Mensaje: Almacenando señal y c) Mensaje: Almacenado correcto.	70
7.14. Mensajes para el proceso de encriptado: a) Mensaje: Indicación para encriptar, b) Mensaje: Inicio de encriptado y c) Mensaje: Finalización de encriptado.	70
7.15. Mensajes para el proceso de desencriptado: a) Mensaje: Indicación para desencriptar, b) Mensaje: Extrayendo información y c) Mensaje: Finalización de desencriptado.	71
7.16. Mensaje: Finalización del sistema.	71

7.17. Archivos *.txt obtenidos de los procesos de encriptado y desencriptado correspondiente a un ECG de 1000 datos: a) Datos obtenidos encriptados, b) Datos obtenidos desencriptados.	72
7.18. Señal de ECG encriptada con la clave 1: a) Señal de ECG recuperada y b) Señal de ECG no recuperada.	74
7.19. Histograma de la señal clara.	76
7.20. Histograma de la señal encriptada.	76

Lista de Tablas

2.1. Ventajas y desventajas de la telemedicina.	8
2.2. Servicios de telemedicina y sus especialidades.	9
3.1. Tamaño máximo de clave de los mapas caóticos analizados.	29
3.2. Tiempo en generar 1000 datos caóticos discretos.	29
3.3. Máximo exponente de Lyapunov de cada mapa caótico.	30
3.4. Datos repetidos en 100 iteraciones de los mapas caóticos.	31
3.5. Comparación del máximo exponente de Lyapunov del mapa Badola original y mejorado.	32
5.1. Comparación de tres familias de microcontroladores.	46
6.1. Clave secreta.	54
7.1. Claves secretas utilizadas para el análisis de sensibilidad a la clave secreta.	73
7.2. Resultados de los análisis NPCR y UACI.	75
7.3. Recursos de implementación.	78

Capítulo 1

Introducción

La introducción de las nuevas tecnologías en la asistencia médica y el avance de las telecomunicaciones han impulsado la rápida repercusión de la telemedicina en los distintos sistemas de salud. Las nuevas tecnologías de la información y de la comunicación han abierto innumerables posibilidades en el intercambio de información en materia de salud y comienzan a hacer posibles nuevas formas de asistencia, incluso aquellas que se realizan a distancia entre el profesional médico y el paciente. Hoy en día, la transferencia de información generalmente se ve facilitada por el uso de algún tipo de red de telecomunicaciones [1].



Figura 1.1: La telemedicina alude al uso de las tecnologías de comunicación.

La telemedicina se ha considerado en los últimos años, como una disciplina científica a medio camino entre la medicina y la tecnología, la cual demanda desarrollo de nuevas herramientas más efectivas en las telecomunicaciones [2]. Uno de los grandes desafíos que enfrenta la humanidad en el siglo *XXI* es brindar atención médica de alta calidad a todos. Tradicionalmente, parte de la dificultad para lograr el acceso equitativo a la atención médica, ha sido que el médico y el paciente deben estar presentes en el mismo lugar y al mismo tiempo. Avances recientes en tecnologías de información y comunicación han creado oportunidades sin precedentes.

En la actualidad, la telemedicina es considerada como una nueva forma de atención médica que se caracteriza por entregar servicios de salud a distancia. La telemedicina promueve una mejor atención a los pacientes, brindando un diagnóstico rápido y eficiente además contribuye a reducir los costos y la concentración de pacientes en las entidades de salud [3]. El uso de las telecomunicaciones en el área de la salud, debe estar acompañado de medidas de seguridad adecuadas para garantizar la confidencialidad, disponibilidad e integridad de la información médica, ofreciendo protección a los pacientes, los profesionales de la salud y al recurso humano en general [4].

La seguridad de la información de salud es un proceso iterativo impulsado por mejoras en la tecnología, así como por cambios en el entorno de atención médica. A medida que mejora la calidad y la eficiencia de la atención en su práctica, también es igualmente importante reevaluar sus políticas de seguridad y la información de salud. Identificar los riesgos y proteger la información electrónica de salud puede ser un desafío para las pequeñas prácticas de atención médica y el desarrollo de políticas de seguridad apropiadas para proteger la información de salud electrónica.



Figura 1.2: La información en la telemedicina debe ser privada.

La criptografía se ocupa de dar solución a los problemas de identificación, autenticación y privacidad de la información en los sistemas informáticos [5]. Sin embargo, en los métodos tradicionales de encriptado de datos, los ataques para poder acceder a la información de los textos encriptados se han hecho más factibles debido al gran avance tecnológico y el desarrollo de las computadoras con las cuales descubrir las claves de los textos encriptados se ha vuelto una tarea cada vez más fácil.

En el diseño de un sistema criptográfico, los algoritmos de encriptado deben tener la propiedad de que los datos originales, denominados texto claro, deben ser recuperados a partir de los datos encriptados o criptogramas. La clave se debe mantener en secreto a pesar de dar a conocer el texto encriptado como información pública. La encriptación presenta algunas limitaciones prácticas visibles como son: los mensajes de entrada a los algoritmos no cuentan con protección, el intercambio de claves se realiza sobre los mismos canales en que se transmiten los datos, la encriptación no provee protección contra las modificaciones y esta puede ser detectada incluyendo como parte de los datos encriptados un número de bits de chequeo.

El presente trabajo trata de mejorar el nivel de seguridad en la transmisión de datos, para lo cual aplicamos la teoría del caos. El caos es una de las áreas de la matemática que investiga y analiza la dinámica no lineal. Asimismo, puede definirse como la tendencia de los sistemas iterativos, simples, determinísticos a ser muy sensibles a las condiciones iniciales y a ser altamente no predecibles.

El elemento central de todos los sistemas caóticos es el concepto de iteración. El estado actual del sistema es una función determinística del estado o valor anterior. Formalmente, una correspondencia caótica se especifica por medio de la siguiente expresión:

$$x_{n+1} = f(x_n) \quad (1.1)$$

En la ecuación (1.1), muestra la expresión de la correspondencia caótica, donde $f()$ es una función no lineal. En general, la teoría del caos surge de la necesidad de modelar mecanismos físicos tales como la predicción del tiempo atmosférico, la evolución de la población, la dinámica de fluidos, la teoría de gases, etc. Todos los ejemplos anteriores son sistemas iterativos por naturaleza. Un sistema caótico es un sistema dinámico, no lineal, determinístico que muestra una dependencia muy sensible a las condiciones iniciales y presenta una evolución a través de un espacio de fase que parece ser aleatorio además permite repetir la misma cadena de números siempre que se utilice la misma función de correspondencia caótica y valor inicial (ver figura 1.3). Estas propiedades proporcionan una ventaja para aplicaciones en criptografía ya que las predicciones a largo plazo de los sistemas caóticos son muy difíciles. Actualmente, se han desarrollado trabajos donde se protege la información médica (texto e imágenes) utilizando encriptado caótico [6–9].

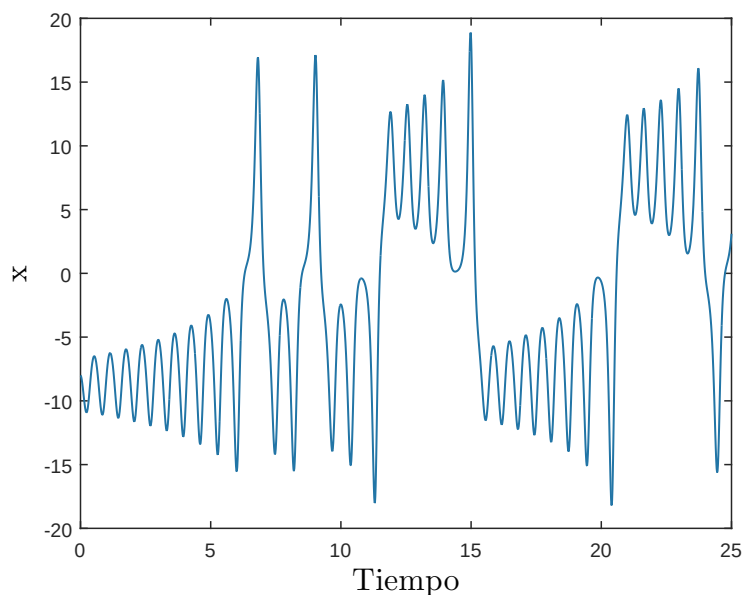


Figura 1.3: Estado x del sistema caótico de Lorenz.

Puesto que las funciones caóticas son muy sensibles a las condiciones iniciales, cualquier ligera diferencia en el valor inicial empleado, significará que el texto encriptado producido utilizando caos será muy diferente.

1.1. Motivación

En la actualidad, las señales biomédicas se transmiten a través de medios de comunicaciones inseguros como internet o redes de computadoras, por lo que son susceptibles a robo de información del paciente y puede verse afectado seriamente en lo económico, emocional y en la salud, por actividades ilícitas como estafas, fraudes, robo de identidad, o diagnóstico médico incorrecto [10].

Por lo tanto, surge la necesidad de desarrollar sistemas que brinden privacidad a la información médica para aplicaciones en telemedicina, particularmente en seguridad embebida. El uso de sistemas embebidos de bajo costo está presente cada vez más en aplicaciones de telemedicina: Telemetría, cuidado de la salud en casa, entrenamiento físico, red de sensores en hospitales, entre otros.

En este trabajo de tesis de maestría se desarrollará un esquema de encriptado caótico para proporcionar confidencialidad a las señales biomédicas en aplicaciones de seguridad embebida y así proteger las señales en su almacenamiento y transmisión en telemedicina.

1.2. Objetivos y alcances de la tesis

Debido al interés recientemente de resguardar la información de forma confidencial en telemedicina con el uso de encriptado caótico, surge la realización de este trabajo de tesis de maestría en el cual se planteó alcanzar el siguiente *objetivo general*:

Diseñar e implementar un algoritmo criptográfico basado en caos para proporcionar confidencialidad a señales biomédicas en sistema embebido.

Que para cumplir con el objetivo general, se plantea alcanzar los siguientes *objetivos específicos*:

1. Seleccionar e implementar en sistema embebido, el mapa caótico en base a sus características de aleatoriedad y procesamiento.
2. Implementar la adquisición de algunas señales biomédicas en sistema embebido.
3. Diseñar e implementar en sistema embebido, un algoritmo criptográfico basado en caos para encriptar señales biomédicas.
4. Determinar seguridad criptográfica y eficiencia del algoritmo propuesto para el encriptado de señales biomédicas.

El alcance de la presente tesis de maestría aplica de forma teórica y práctica. Se diseña un algoritmo criptográfico caótico, el cual se implementa en un microcontrolador de 32 bits para aplicaciones embebidas, se valida mediante distintos análisis de seguridad y es utilizable para la transmisión y almacenamiento de señales biomédicas.

1.3. Organización del manuscrito

Este trabajo de tesis de maestría está compuesto por 8 capítulos, los cuales se describen de manera breve a continuación:

- **Capítulo 1:** Se presenta la introducción a este trabajo de investigación, la motivación y los objetivos.
- **Capítulo 2:** Se introduce a la telemedicina y se muestran algunas características que la componen, así como se da a conocer la telemetría y la seguridad en ella.
- **Capítulo 3:** Se describe el caos y las propiedades que lo componen además se analizan distintos mapas caóticos y se determina el mapa caótico a utilizar para el algoritmo de encriptado de este trabajo de tesis.
- **Capítulo 4:** Se da a conocer lo que es un sistema criptográfico, así como sus características y propiedades donde se analizan junto con la seguridad criptográfica.
- **Capítulo 5:** Se describen de manera breve los sistemas embebidos y se da una introducción al microcontrolador que será utilizado para las aplicaciones experimentales de este trabajo.
- **Capítulo 6:** Se presenta el algoritmo de encriptado propuesto y las consideraciones para generar un encriptado con excelentes características para que pueda resistir los distintos ataques criptoanalíticos y además que sea eficiente.
- **Capítulo 7:** Se muestra la implementación del algoritmo de encriptado caótico propuesto en un sistema embebido basado en un microcontrolador de 32 bits y se realizan distintos análisis de seguridad para validar el algoritmo.
- **Capítulo 8:** Se mencionan las conclusiones generales de este trabajo de maestría y algunos puntos que se pueden hacer para trabajo a futuro.

Capítulo 2

Telemedicina

En este capítulo se da a conocer la telemedicina, así como sus características y algunas consideraciones que se deben tener en cuanto a la seguridad, se da una breve introducción a la telemetría, la cual, es la rama donde se enfocará este trabajo de tesis de maestría y se dan detalles sobre las señales biomédicas con las que se trabaja en telemetría.

2.1. Introducción

El prefijo “*tele*” deriva del griego “*distancia*”, por lo tanto, telemedicina es medicina a distancia, es decir, la prestación de cuidados médicos y el intercambio de información médica a través de la distancia. La telemedicina alude al uso de las tecnologías de comunicación e información sobre servicios médicos a distancia (ver figura 2.1). Estas tecnologías permiten intercambiar datos con el fin de examinar, diagnosticar, tratar un paciente y facilitar el acceso de la población, independiente del área geográfica donde habite [11].



Figura 2.1: La telemedicina permite la transmisión de información médica a distancia.

La mayoría de sus avances se han realizado en el siglo *XX*, en especial en los últimos 20 a 30 años, pero su historia es mucho más antigua. La telemedicina se empieza a desarrollar plenamente con los avances de la comunicación electrónica que tuvo dos fases:

- Primera fase. Se utilizaban las comunicaciones analógicas como: telegrafía, teléfono y radio.
- Segunda fase. Es la fase actual en donde se emplean las comunicaciones digitales.

Ejemplos de su uso en la primera fase se dieron en los barcos de guerra donde el radio era el único medio de comunicación con tierra y se enviaba información del estado de salud de la tripulación.

La explosión de las telecomunicaciones ha permitido el desarrollo de la telemedicina, que completa en el campo de la investigación, la transmisión de vídeos con imágenes, así como el electrocardiograma (ECG) desde el lugar de los hechos. Las técnicas de conexión inalámbricas también incluyen el uso de las comunicaciones por satélite de bajo costo para facilitar el acceso a internet (uso de correo electrónico).

La comunicación a distancia entre dos o más personas puede establecerse en tiempo real o en diferido. Lo mismo sucede con la comunicación entre médicos mediante sistemas de transmisión de información.

En [12], se mencionan dos modos de operación básicos, los cuales son:

1. **En tiempo real o modo síncrono:** usa el sistema de teleconferencia o videoconferencia (ver figura 2.2), en el que se une la imagen y el sonido para que dos o más personas tengan la posibilidad de comunicarse entre sí en tiempo real. En este caso, la imagen clínica suele ser una imagen de vídeo y tanto el paciente como el médico consultado están presentes físicamente.



Figura 2.2: Comunicación en modo síncrono: Teleconferencia.

2. **En tiempo diferido o modo asíncrono:** también, conocido como almacenar y enviar. Utiliza el sistema de correo electrónico (ver figura 2.3). En general, el médico consultado recibe un mensaje con una historia clínica, la estudia y emite por correo electrónico su opinión, diagnóstico y su consejo terapéutico. No es preciso que el médico consultor y el paciente coincidan en el tiempo con el médico consultado.



Figura 2.3: Comunicación en modo asíncrono: Correo electrónico.

La telemedicina cuenta con ventajas y desventajas en lo que se refiere a los pacientes, médicos profesionales y en los sistemas. En la siguiente tabla (ver tabla 2.1), se muestran algunas ventajas y desventajas sobre la telemedicina [13].

Tabla 2.1: Ventajas y desventajas de la telemedicina.

Ventajas	Desventajas
Diagnósticos y tratamientos más rápidos	Seguridad en el manejo de datos
Evita traslados	Falta de disponibilidad tecnológica
Mayor satisfacción de los usuarios	Aceptabilidad de los profesionales de la salud
Acceso a la información más fácil	Requiere recursos específicos
Mejora en la gestión de salir	Alto costo
Mejor utilización de los recursos	Falta de demanda
Aumento de uso de las nuevas tecnologías	Falta de infraestructura
Ahorro de tiempo	Mala adaptación de las tecnologías

2.2. Características de la telemedicina

Los servicios de telemedicina manejan diferentes tipos de información, datos de señales biomédicas, imágenes, videos, registros, etc., que permiten la comunicación a distancia entre diferentes especialistas. Existe un amplio conjunto de servicios de telemedicina entre los cuales se encuentran los servicios de teleconsulta, telediagnóstico y telemetría (ver tabla 2.2). Estos servicios tienen características principales en común, pero pueden variar en las especialidades consideradas. Por ejemplo: una clasificación por el tipo de información para transmitir (datos, audio o imágenes), o por la tecnología usada o el tipo de especialidades [14–16].

Una red de telemedicina por lo general, está compuesta por pacientes, centros de salud, médicos especialistas, equipos de comunicaciones y medios de transmisión. Esta red cuenta con servidores que permiten implementar aplicaciones, pero existen estándares regulatorios para redes de telemedicina [17].

Tabla 2.2: Servicios de telemedicina y sus especialidades.

Servicios	Especialidades
Teleconsulta	Registro clínico electrónico
Telediagnóstico	Teleendoscopia, Teledermatología, Teleoftamología
Teleterapia	Telepsiquiatría, Telefisioterapia, Teleprescripción
Telemetría	Teleradiología, Telepatología, Telecardiología

La existencia de estándares regulatorios sobre el manejo de información médica, ha permitido que las entidades de salud alcancen mayores índices de confiabilidad en la transmisión de información. Dentro de los estándares más representativos, se destaca el HIPAA y COBIT. Estos estándares establecen parámetros para preservar la regulación de transmisión de información. A continuación, se hará una breve descripción de cada estándar:

La Ley de Transferibilidad y Responsabilidad de Seguros de salud, llamada por sus siglas en inglés HIPAA, es un conjunto de estándares que aseguran la protección de información médica en aspectos como la transmisión, almacenamiento y acceso a la información de salud [18].

Objetivos de control para tecnología de la información y relacionadas, llamada por sus siglas en inglés COBIT, es un conjunto de mejores prácticas para seguridad, calidad, eficacia y eficiencia en las tecnologías de la información necesarias para identificar riesgos, gestionar recursos y medir el rendimiento que permitan alcanzar los objetivos de una organización para su mejora en cuestiones de seguridad [19].

El servicio de telemetría permite a los especialistas de la salud, por medio de imágenes, datos y videos, recolectar información de forma remota acerca del estado de salud de los pacientes para consulta y diagnóstico [20]. La telemetría es utilizada para el monitoreo de signos vitales como el electrocardiograma (ECG), electroencefalograma (EEG), presión arterial, temperatura, oximetría de pulso (SpO2) y exámenes de laboratorio mediante una función digital (ver figura 2.4), para enfermedades metabólicas que requieren llevar un control para tener un mejor cuidado de la salud.

**Figura 2.4:** Imagen de un monitor médico donde se visualizan los signos vitales a distancia.

2.3. Electrocardiografía (ECG)

La electrocardiografía registra las señales eléctricas del corazón en función del tiempo, de las variaciones de potencial eléctrico generadas por el conjunto de células cardíacas. Es la técnica más usada para el estudio electrofisiológico del corazón, debido a que es un método no invasivo y permite registrar la actividad eléctrica del corazón (ver figura 2.5). Mediante electrodos aplicados en varias regiones del cuerpo se puede obtener, tras amplificarlas, un registro de estas descargas eléctricas [21].

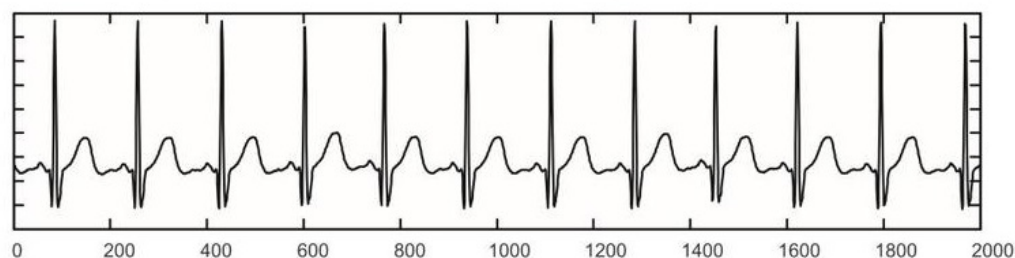


Figura 2.5: Señal ECG.

Las derivaciones electrocardiográficas son disposiciones específicas de los electrodos. Las derivaciones bipolares creadas por Willen Einthoven registran la diferencia de potencial eléctrico que se produce entre dos puntos conocido como triangulo de Einthoven (ver figura 2.6) [22].

- DI: Registra la diferencia de potencial entre el brazo izquierdo (polo positivo) y el derecho (polo negativo).
- DII: Registra la diferencia de potencial que existe entre la pierna izquierda (polo positivo) y el brazo derecho (polo negativo).
- DIII: Registra la diferencia del potencial que existe entre la pierna izquierda (polo positivo) y el brazo izquierdo (polo negativo).

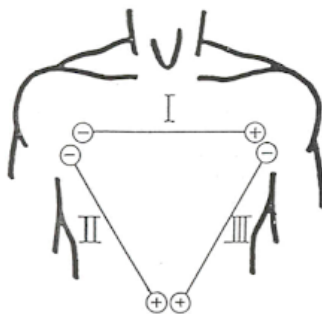


Figura 2.6: Triangulo de Einthoven.

2.4. Seguridad en la telemedicina

Es importante tener en cuenta que la información médica que se maneja en telemedicina, es en general de tipo confidencial y por lo tanto, requiere ser protegida de ataques y amenazas que puedan afectar el derecho a la intimidad, la privacidad y la protección de los datos de los pacientes. Estos ataques pueden afectar los datos que se encuentren encriptados o no, por medio de técnicas de hurto de información como lo son los programas espía, los virus y el acceso no autorizado a la información, la alteración, deterioro total o parcial de la información. Para ello, es importante definir metodologías y estándares que puedan ser aplicados, así garantizar la seguridad sobre el manejo de información médica en telemedicina.

La seguridad y confiabilidad sobre redes de telemedicina, son dos de los aspectos más relevantes para almacenar, acceder y transmitir información médica de los pacientes. Analizar estos dos aspectos, previene amenazas y ataques a los sistemas de telemedicina. Para poder establecer mecanismos de prevención y protección de la información en redes de telemedicina, se debe determinar las debilidades y defectos denominados vulnerabilidades (ver figura 2.7).

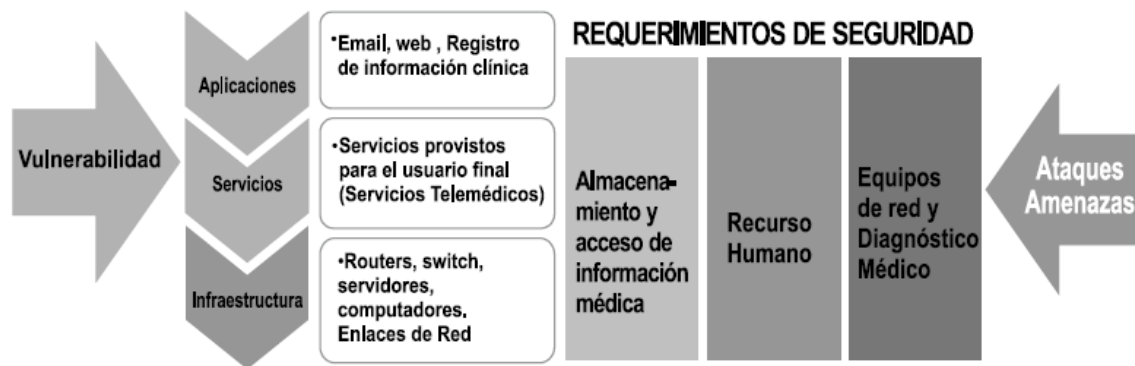


Figura 2.7: Ataques y vulnerabilidades en telemedicina.

El análisis de los riesgos a los cuales se expone la información en una red de telemedicina, permite determinar el alcance de los posibles daños a la integridad de la información médica, generados por las vulnerabilidades en una red de telemedicina.

La seguridad de la información es la protección de la información y los sistemas de información contra el acceso, uso, divulgación, interrupción, modificación o destrucción no autorizados. La seguridad de la información se logra garantizando la confidencialidad, integridad y disponibilidad de la información.

2.5. Conclusiones

En este capítulo se mostró que la telemedicina crece a la par con la tecnología brindando nuevas oportunidades para trabajar en conjunto, la telemedicina avanza rápidamente y representa una excelente alternativa para mejorar algunos programas de salud, ya que facilita el análisis y la toma de decisiones terapéuticas porque con el uso de sistemas embebidos en aplicaciones médicas se ha permitido desarrollar comunicaciones más fáciles entre médicos y pacientes a distancia a un bajo costo. En este trabajo se utilizará comunicación asíncrona o más comúnmente conocida como almacenar y enviar, ya que se obtendrán señales biomédicas las cuales serán encriptadas y almacenadas para después poder ser transmitidas por un canal inseguro.

Capítulo 3

Caos

En este capítulo, se da una introducción al caos y se explica su definición desde un punto de vista físico-matemático. Se muestran las propiedades del caos y se analiza el máximo exponente de Lyapunov, el cual se aplica a los distintos mapas caóticos que se estudiaron para obtener dinámicas caóticas y así poder compararlos para seleccionar uno en base a los resultados obtenidos y utilizarlo en el algoritmo de encriptado.

3.1. Introducción

La *teoría del caos* es una disciplina que ha aparecido hace unos años a partir del intento de establecer leyes físicas para la predicción del tiempo atmosférico y que está encontrando una amplia aplicación en muchos campos de la ciencia. El caos es la denominación en las ciencias exactas, que trata sobre comportamientos impredecibles en sistemas dinámicos no lineales (sistemas complejos que cambian o evolucionan con el estado del tiempo), es un comportamiento impredecible, surge en un sistema determinista debido a la gran sensibilidad a las condiciones iniciales que presenta. El caos surge en un sistema dinámico los cuales se definen como un conjunto de elementos interconectados entre sí y destinados a cumplir una tarea específica. Este conjunto de elementos, describe un comportamiento cuya evolución depende del tiempo, ya sea de forma explícita o implícita y de forma continua o discreta [23]. Si dos puntos arbitrarios iniciales y cercanos divergen exponencialmente tal que su comportamiento futuro es eventualmente impredecible, pero determinable [24]. El término determinista alude al hecho de que cualquier evolución futura del sistema es una consecuencia de las condiciones en las que se encuentra el sistema en el instante inmediatamente anterior [25].

La sensibilidad a las condiciones iniciales, también conocida como el efecto mariposa [26]. A saber: *Si hoy agita una mariposa sus alas en Barcelona, con su aleteo puede producir un huracán en Tokio el mes que viene.* Esto significa que, el mismo sistema caótico tendrá evoluciones temporales sumamente diferentes, a pesar que sus condiciones iniciales sean ligeramente distintas, siendo las trayectorias la distinta evolución del sistema según sea el punto de partida (ver figura 3.1). Obsérvese que después de 9 unidades temporales los dos sistemas (estructuralmente idénticos) comienzan a divergir.

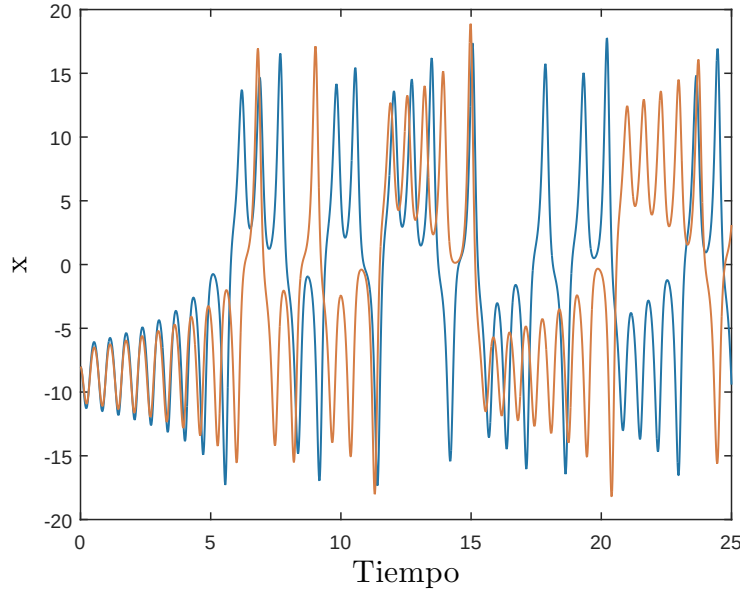


Figura 3.1: Estado x del sistema de Lorenz con pequeñas variaciones en las condiciones iniciales.

Los sistemas dinámicos pueden comportarse de diferentes formas y su función tiende a estabilizarse en uno o más valores. Este conjunto de valores para los cuales la función $f(x)$, se estabiliza cuando el número de iteraciones tiende a infinito, se denomina atractor. El concepto atractor va ligado al de los sistemas dinámicos. Para caracterizar un sistema dinámico, contamos con variables dinámicas y variables estáticas, estas últimas son los parámetros o constantes.

De forma general se puede decir que un atractor es una región o conjunto cerrado en el espacio de fase, una representación gráfica cartesiana donde cada eje es una variable dinámica y se representa la evolución del sistema dinámico, al cual convergen las trayectorias de un sistema, entendiéndose por trayectoria la evolución temporal del sistema a partir de una condición inicial específica [27]. Dicha trayectoria es arrastrada hacia una región del espacio de estados llamada atractor, que no es sino la manifestación de los parámetros fijos y de las ecuaciones que determinan los valores de las variables dinámicas. De esta forma el atractor aparece y se puede identificar a un sistema dinámico caótico por la presencia de un atractor extraño cuya peculiaridad es el poseer una dimensión fractal.

La palabra caos para designar el desorden y la teoría que lo expresa se debe principalmente a Henri Poincaré y a Edward Lorenz. La teoría del caos se define como el *estudio de la incertidumbre y la impredecibilidad en las matemáticas y en la naturaleza*. El objeto de estudio de la teoría del caos lo constituyen fenómenos dinámicos fuera de equilibrio de comportamiento no lineal [28].

El matemático Edward Lorenz usaba su ordenador para desentrañar la maraña matemática que él mismo había creado con sus doce ecuaciones para predecir el tiempo atmosférico en el Massachusetts Institute of Technology [29]. Era el año 1960. La predicción del tiempo se debía regir por ecuaciones. Para ello escogió 12 funciones, hojeando los rollos papel con datos numéricos que salían de su impresora, Lorenz ideó un método para que el ordenador señalara cada minuto el paso de un día imprimiendo una hilera de números.

En 1961, Lorenz cansado de observar ese vaivén numérico salido de la impresora de su ordenador, intentó atajar partiendo de una sucesión que tenía anteriormente, pero al traspasar los dígitos sólo tecleó 3 en vez de los 6 originales, esperando que el comportamiento no cambiara. Los resultados obtenidos fueron distintos y trajeron de cabeza a Lorenz pues no eran los esperados y revisó el software y hardware hasta darse cuenta finalmente, que el error lo cometió al truncar el valor inicial de la función cambiando el valor de 0.506127 a 0.506.

Había dado con el *efecto mariposa*. Este redondeo insignificante era el aleteo de la mariposa; y el comportamiento anómalo, o digamos inesperado, de la función del huracán que se produciría el próximo mes en Tokio. A su descubrimiento Lorenz lo llamo: *Dependencia sensitiva de las condiciones iniciales*. Con ello creó la base de una nueva ciencia: *el Caos*. Lorenz animado por su descubrimiento, decidió comenzar a experimentar con sus resultados en el campo de las corrientes de fluidos y sus 12 fórmulas se vieron reducidas a 3 simples ecuaciones no lineales.

$$\frac{dx}{dt} = \sigma(y - x), \quad (3.1a)$$

$$\frac{dy}{dt} = \rho x - y - xz, \quad (3.1b)$$

$$\frac{dz}{dt} = xy - \beta z \quad (3.1c)$$

donde x , y y z son los estados del sistema, x_0 , y_0 y z_0 son las condiciones iniciales, σ , ρ y β son los parámetros de control y t es el tiempo.

Lorenz representó gráficamente los resultados obtenidos con sus tres ecuaciones en una gráfica tridimensional, asignando el valor obtenido de cada ecuación a una de las tres dimensiones del plano. Al ver el gráfico resultante, llamado en adelante atractor de Lorenz (ver figura 3.2), Lorenz se encontró otra vez con el efecto mariposa, concretamente con sus alas. A pesar de lo anterior lo impredecible del sistema, lejos de ser un comportamiento al azar, tenía una curiosa tendencia a evolucionar dentro de una zona muy concreta del espacio de fases, situando una especie de pseudocentro de gravedad de los comportamientos posibles. Lorenz descubrió que su sistema contenía una dinámica extremadamente errática, las soluciones oscilaban irregularmente sin llegar a repetirse, pero en una región acotada del espacio de fases. Vio que las trayectorias rondaban siempre alrededor de lo que ahora definimos como atractor extraño.

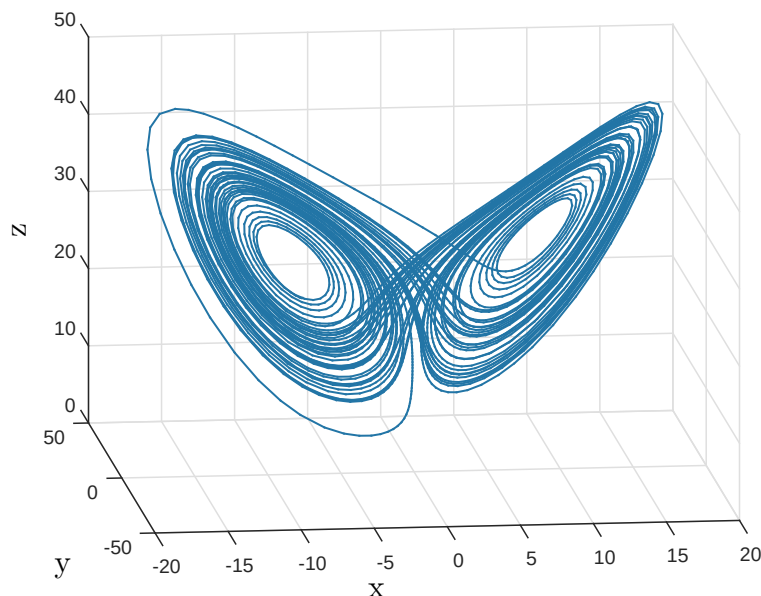


Figura 3.2: Atractor extraño de Lorenz.

Durante la última década, ha habido un tremendo interés en estudiar el comportamiento de los sistemas caóticos. El caos tiene aplicaciones potenciales en varios bloques funcionales en comunicaciones digitales: compresión, encriptación y modulación. La posibilidad para la autosincronización de oscilaciones caóticas [30–33], ha provocado una avalancha de trabajos sobre la aplicación del caos en criptografía en donde se han publicado algunos trabajos recientes [34]. La criptografía es generalmente reconocida como el mejor método de datos de protección contra fraude [35]. Una visión general de los desarrollos recientes en el diseño de algoritmos criptográficos se da en [36].

3.2. Sistemas caóticos y sus propiedades

Es en los sistemas dinámicos donde podemos usar el término caos y donde una variación mínima de las condiciones iniciales supone un comportamiento totalmente distinto del esperado por parte del sistema. Es decir, que un sistema podrá ser caótico cuando su comportamiento sea impredecible [37]. Los sistemas dinámicos, son sistemas que varían con el paso del tiempo.

En [38] habla sobre la clasificación de los sistemas dinámicos, los cuales son:

- Estables: tienden a un punto a lo largo del tiempo o siguen una misma órbita, sus ecuaciones características, condiciones iniciales, sus límites, elementos y relaciones nos permiten conocer su evolución a través del tiempo, es decir, sabemos hacia donde lo dirige su atractor.
- Inestables: cuando dos soluciones con condiciones iniciales diferentes acaban divergiendo por pequeñas que sean las condiciones iniciales.

- **Caóticos:** las soluciones se mueven en torno al atractor de manera irregular y pasado el tiempo ambas soluciones no son cercanas, si bien suelen ser cualitativamente similares. De esa manera, el sistema permanece confinado en una zona de su espacio de estados. Los sistemas caóticos, por su parte, manifiestan ambos comportamientos. Se pueden conocer sus ecuaciones y sus condiciones iniciales fijas, sin embargo, la más mínima variación provoca una evolución radical en su comportamiento.

Uno de los más citados ejemplos de sistema caótico es el clima atmosférico, del cual podemos predecir su comportamiento y elaborar pronósticos en base a ecuaciones, estudios de su comportamiento pasado y el conocimiento de sus condiciones iniciales. Sin embargo, no podemos conocer con exactitud los parámetros que fijan sus condiciones iniciales.

Un sistema caótico se puede describir por un conjunto de ecuaciones diferenciales o en diferencias no lineales, que generan secuencias caóticas que son deterministas, es decir, el valor futuro depende del valor actual y que, además, presentan las siguientes propiedades:

- *No linealidad.* Son sistemas de ecuaciones diferenciales (tiempo continuo) o en diferencias (tiempo discreto) no lineales, que no cumple con el principio de superposición.
- *Sensibilidad exponencial a condiciones iniciales y parámetros de control.* La dinámica o trayectoria del sistema caótico se vería altamente modificada si se varía ligeramente una condición inicial o parámetro de control.
- *Mezcla de datos.* Un pequeño rango de condiciones iniciales cubre la mayor parte del espectro caótico.
- *Ergodicidad.* La trayectoria caótica se mantiene confinada en un espacio conocido como atractor extraño con respecto al tiempo cubriendo en su totalidad su espacio para cualquier entrada de condición inicial o parámetro de control.
- *Exponente de Lyapunov positivo.* Un sistema de dimensión n posee n exponentes de Lyapunov; si uno de ellos es positivo, el sistema es caótico; si dos o más son positivos, el sistema es hypercaótico.
- *Atractor extraño con dimensión fractal.* La gráfica de fase del sistema genera lo que se conoce como atractor, que puede ser punto fijo (sistema estable), ciclo límite (sistema periódico) o atractor extraño (sistema caótico).

Muchas de las propiedades mencionadas están estrechamente relacionadas con propiedades criptográficas, por lo que los sistemas caóticos se utilizan para la protección de datos privados en sistemas de comunicaciones inseguros.

3.3. Exponente de Lyapunov

Esta medida de caos, fue introducida por el célebre matemático ruso Alexander Mi-jailovich Lyapunov a principios del siglo XX . Los exponentes de Lyapunov, como ahora se les conoce, son un conjunto de números que se emplean usualmente para detectar la presencia del caos en sistemas dinámicos [39]. Los exponentes de Lyapunov miden la tasa promedio de divergencia o convergencia exponencial de trayectorias cercanas en el espacio de fase. Dado que condiciones iniciales cercanas corresponden a estados iniciales prácticamente idénticos, la divergencia exponencial de las órbitas implica la pérdida de la predictibilidad del sistema. Cualquier sistema que contenga al menos un exponente de Lyapunov positivo, se define como caótico [40].

El exponente de Lyapunov está definido para una función dinámica con una configuración inicial de acuerdo a la siguiente expresión:

$$\lambda = \frac{1}{T} \ln \left| \frac{f^n(x_n - \delta_0) - f^n(x_n)}{\delta_0} \right| \quad (3.2)$$

donde λ es el exponente de Lyapunov, x_0 es una condición inicial, $x_0 = x_0 + \delta_0$ es otra condición inicial extremadamente cercana y T es el número de iteraciones.

Algunas de las propiedades del máximo exponente de Lyapunov son las siguientes:

- Si el sistema es conservativo (no existe disipación), la suma de todos los exponentes de Lyapunov debe ser cero.
- Si el sistema es disipativo, la suma será negativa.
- En un sistema dinámico hamiltoniano, la suma sólo puede ser positiva.

Este análisis es aplicado a cada uno de los mapas caóticos analizados para elegir uno que sea factible para ser utilizado en el encriptado caótico con aplicación en un microcontrolador. Si el mapa caótico es de n -dimensiones, se tendrá n exponentes de Lyapunov.

3.4. Mapas caóticos estudiados

Es este trabajo de tesis de maestría se analizan 9 mapas caóticos, tres de una dimensión y seis de dos dimensiones, los cuales son:

- **Una dimensión:** Logístico, Chebyshev y Seno.
- **Dos dimensiones:** Logístico 2D, Hénon, Seno-Logístico 2D, Badola, Ushio, Lotka-Volterra.

Se implementan en MatLab donde se obtienen las dinámicas caóticas de los estados, el máximo exponente de Lyapunov, histogramas, espectro de Lyapunov y el tiempo que le toma al software matlab en generar las secuencias dinámicas, se analizan los resultados y se selecciona un mapa caótico para la implementación del encriptado caótico. Todas las gráficas de los mapas caóticos son discretas.

3.4.1. Mapa Logístico

El mapa Logístico es un mapeo polinómico de grado dos. El mapa fue popularizado en [41] por el biólogo Robert May. En parte como un modelo demográfico de tiempo discreto análogo a la ecuación logística. Se usa una ecuación en diferencia no lineal para observar los pasos de tiempo discretos. Se llama mapa logístico porque asigna el valor de la población en cualquier paso de tiempo a su valor y la relativa simplicidad del mapa logístico lo convierte en un punto de entrada ampliamente utilizado para considerar el concepto de caos.

Matemáticamente, el mapa logístico está escrito:

$$x_{n+1} = \alpha x_n(x_n - 1), \quad (3.3)$$

donde x_n es la condición inicial con valores entre $0 < x_n < 1$ y α es el parámetro de control entre $3.57 < \alpha < 4$. Con valores en estos rangos se generan dinámicas caóticas discretas (ver figura 3.3).

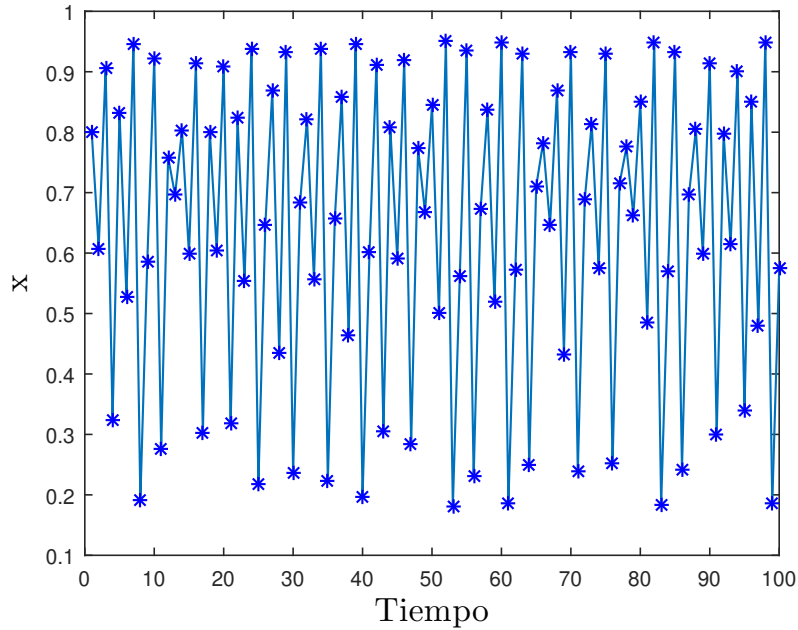


Figura 3.3: Estado x del mapa Logístico con dinámicas caóticas discretas.

Para el estado x y el máximo exponente de Lyapunov se utilizó como parámetro de control $\alpha = 3.800000$, $n = 100$ y condición inicial de $x_n = 0.800000$. El valor obtenido del máximo exponente Lyapunov es de $\lambda = 0.693100216810785$, por lo que se demuestra la existencia de caos.

3.4.2. Mapa Chebyshev

Los polinomios de Chebyshev son curvas de coseno con una perturbación en la escala horizontal, pero la vertical se mantiene constante [42].

El mapa chebyshev está definido por la siguiente ecuación:

$$x_{n+1} = \cos(\alpha \cos^{-1}(x_n)), \quad (3.4)$$

donde $x_n \in (-1, 1)$ y α debe ser $1 < \alpha$ para generar dinámicas caóticas discretas (ver figura 3.4).

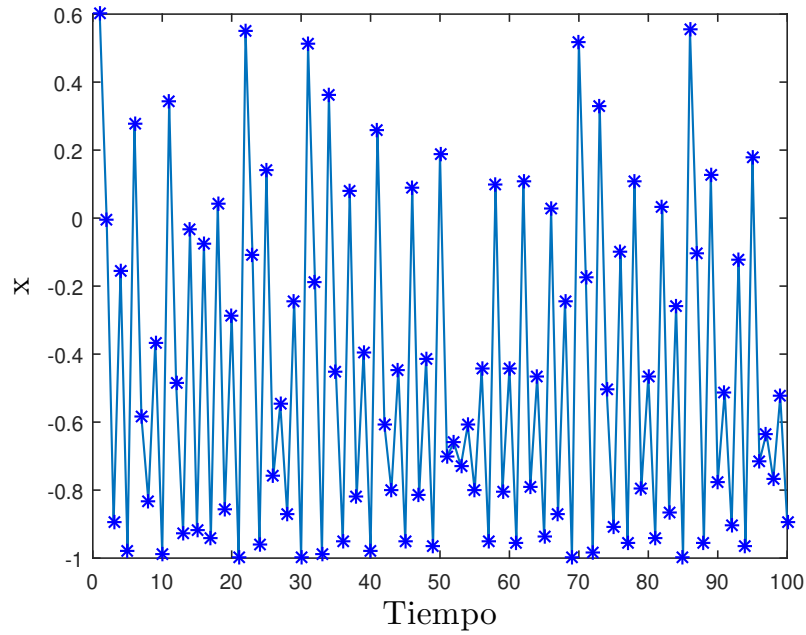


Figura 3.4: Estado x del mapa Chebyshev con dinámicas caóticas discretas.

Para el máximo exponente de Lyapunov se utilizan los valores de $\alpha = 1.700000$ con $n = 100$ y $x_n = 0.600000$. El resultado del máximo exponente de Lyapunov es de $\lambda = 0.544992557548549$, lo que demuestra que genera caos.

3.4.3. Mapa Seno

Cuando la función seno tiene entradas dentro del rango de $[0, \pi]$, sus salidas caen dentro del rango de $[0, 1]$. El mapa Seno se deriva de la función seno transformando sus entradas en $[0, 1]$. Se define matemáticamente en [43] como:

$$x_{n+1} = \beta \sin(\pi x_n), \quad (3.5)$$

donde $x_n \in (0, 1)$. El mapa sinusoidal es caótico cuando $\beta \in (0.87, 1)$ (ver figura 3.5).

Para el máximo exponente de Lyapunov se utilizan los valores de $\beta = 0.950000$ con $n = 100$ y $x_n = 0.500000$. El resultado del máximo exponente de Lyapunov es de $\lambda = 0.415232364706527$, lo que demuestra que genera caos.

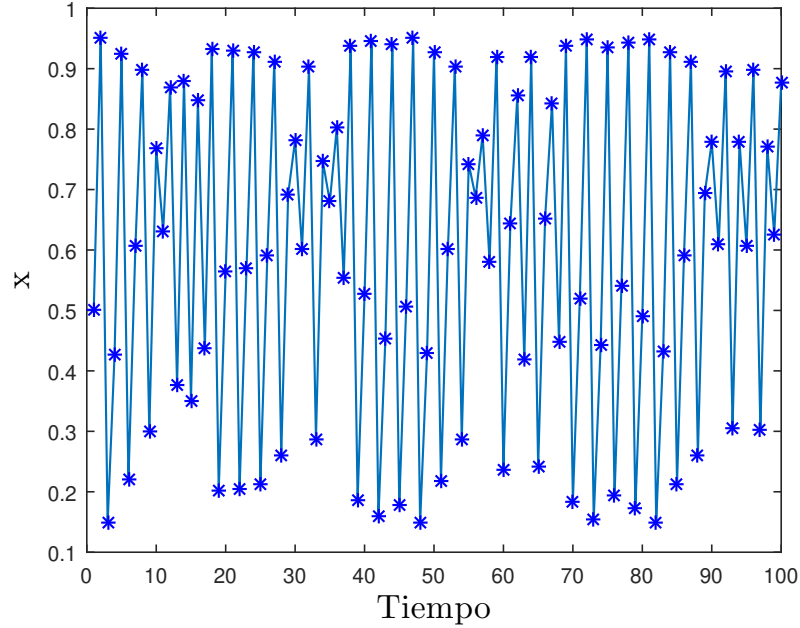


Figura 3.5: Estado x del mapa Seno con dinámicas caóticas discretas.

3.4.4. Mapa Logístico 2D

El mapa logístico 2D tiene un polinomio de segundo orden con el que ambas ecuaciones interaccionan entre sí [44], el cual se define como:

$$x_{n+1} = \alpha_1 x_n (1 - x_n) + \beta_1 (y_n)^2, \quad (3.6a)$$

$$y_{n+1} = \alpha_2 y_n (1 - y_n) + \beta_2 ((y_n)^2 + x_n y_n), \quad (3.6b)$$

donde $x_n \in (0.26, 0.90)$ y $y_n \in (0.19, 0.96)$, son las condiciones iniciales y α_1 , β_1 , α_2 y β_2 , son los parámetros de control, los cuales deben estar en $2.90 < \alpha_1 < 3.40$, $2.75 < \alpha_2 < 3.45$, $0.15 < \beta_1 < 0.21$ y $0.13 < \beta_2 < 0.15$, para generar dinámicas caóticas discretas (ver figura 3.6).

Para el cálculo del máximo exponente de Lyapunov del mapa logístico 2D, se utiliza las condiciones iniciales $x_n = 0.300000$ y $y_n = 0.500000$. Los parámetros de control son $\alpha_1 = 3.200000$, $\beta_1 = 0.160000$, $\alpha_2 = 2.800000$, $\beta_2 = 0.140000$ con $n = 100$. Los valores obtenidos son $\lambda_1 = 0.197298083719773$ y $\lambda_2 = -0.167443932851848$, lo cual nos indica que genera caos.

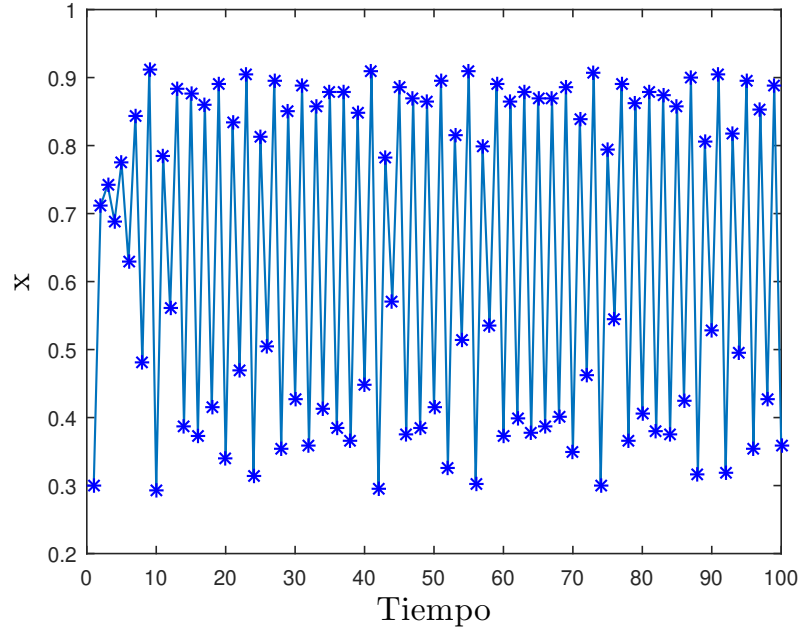


Figura 3.6: Estado x del mapa Logístico 2D con dinámicas caóticas discretas.

3.4.5. Mapa Hénon

Michel Hénon descubría en el instituto de Astrofísica de Paris un sistema dinámico de gran sencillez, mediante el cual se podían explicar las pequeñas oscilaciones que hacen que ciertos cuerpos celestes se desvíen levemente de su órbita elíptica [45]. El mapa de Hénon es un sistema dinámico de tiempo discreto. Es uno de los ejemplos más estudiados de sistemas dinámicos que exhiben un comportamiento caótico.

El sistema consta de dos ecuaciones, una de ellas cuadrática y dos constantes:

$$x_{n+1} = 1 - \alpha x_n^2 + y_n, \quad (3.7a)$$

$$y_{n+1} = \beta x_n, \quad (3.7b)$$

donde x y y representan los estados del sistema discreto y n las iteraciones. Para el mapa clásico de Hénon tiene valores de $1.34 < \alpha < 1.42$ y $\beta = 0.30$.

Para observar dinámicas caóticas discretas (ver figura 3.7) y el atractor extraño (ver figura 3.8) del mapa Hénon, se utiliza las condiciones iniciales $x_n = 0.330000$ y $y_n = 0.400000$, además de $\alpha = 1.40$ y $\beta = 0.30$. Estos mismos valores fueron utilizados para calcular el máximo exponente de Lyapunov, los cuales son $\lambda_1 = 0.171836769160141$ y $\lambda_2 = -1.203972804326008$, lo cual indica caos.

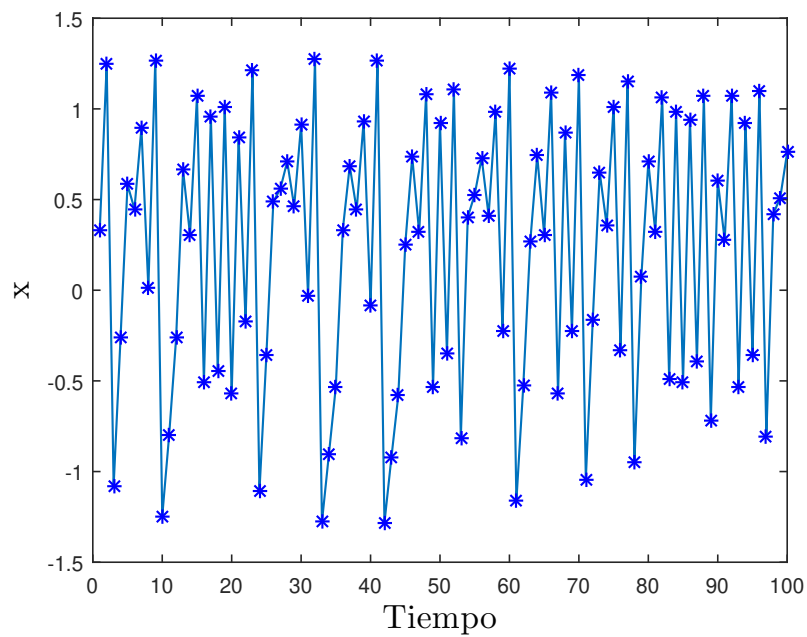


Figura 3.7: Estado x del mapa Hénon con dinámicas caóticas.

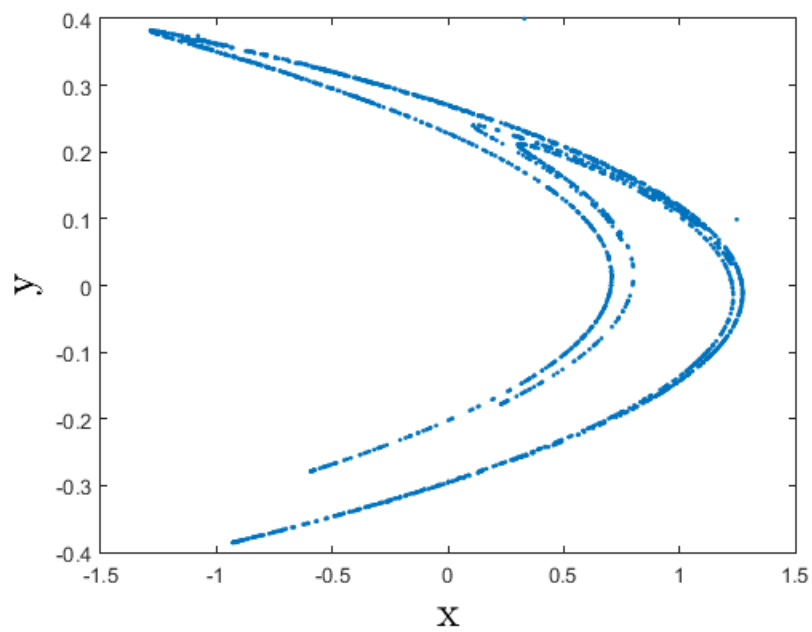


Figura 3.8: Atractor extraño del mapa Hénon.

3.4.6. Mapa Seno-Logístico 2D

Al realizar una combinación del mapa logístico y seno se obtiene las siguientes ecuaciones [46]:

$$x_{n+1} = \sin(\pi\mu(y_n + 3))x_n(1 - x_n), \quad (3.8a)$$

$$y_{n+1} = \sin(\pi\mu(x_n + 3))y_n(1 - y_n), \quad (3.8b)$$

donde $\mu \in (0.44, 0.93)$, es el parámetro de control, x_n y y_n , son las condiciones iniciales entre $(0 - 1)$. Con estos rangos de valores se obtienen dinámicas caóticas discretas (ver figura 3.9).

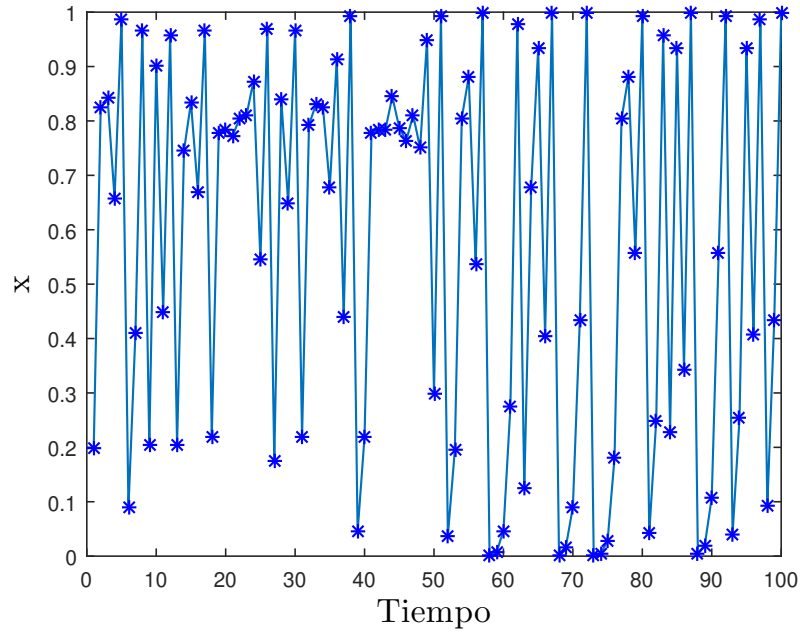


Figura 3.9: Estado x del mapa Seno-Logístico 2D con dinámicas caóticas discretas.

Para el máximo exponente de Lyapunov se utilizan los valores de $\mu = 0.550000$, con $n = 100$, $x = 0.200000$ y $y = 0.500000$. El resultado es de $\lambda_1 = 0.633951719998233$ y $\lambda_2 = 0.509821502321757$. Lo cual indica caos, pero al tener los dos valores positivos nos indica que el mapa es hipercaótico.

3.4.7. Mapa Badola

Considere dos mapas logísticos acoplados, el mapa puede exhibir un comportamiento único estable, periódico o caótico a medida que varía la fuerza de acoplamiento ϵ [47].

$$x_{n+1} = (1 - \epsilon)\mu x_n(1 - x_n) + \epsilon y_n, \quad (3.9a)$$

$$y_{n+1} = (1 - \epsilon)\mu y_n(1 - y_n) + \epsilon x_n, \quad (3.9b)$$

donde $\varepsilon = 0.09$ y $\mu \in (3.60, 3.72)$ son los parámetros de control, x_n y y_n son las condiciones iniciales entre $(0 - 1)$. Con estos rangos de valores se obtienen dinámicas caóticas discretas (ver figura 3.10).

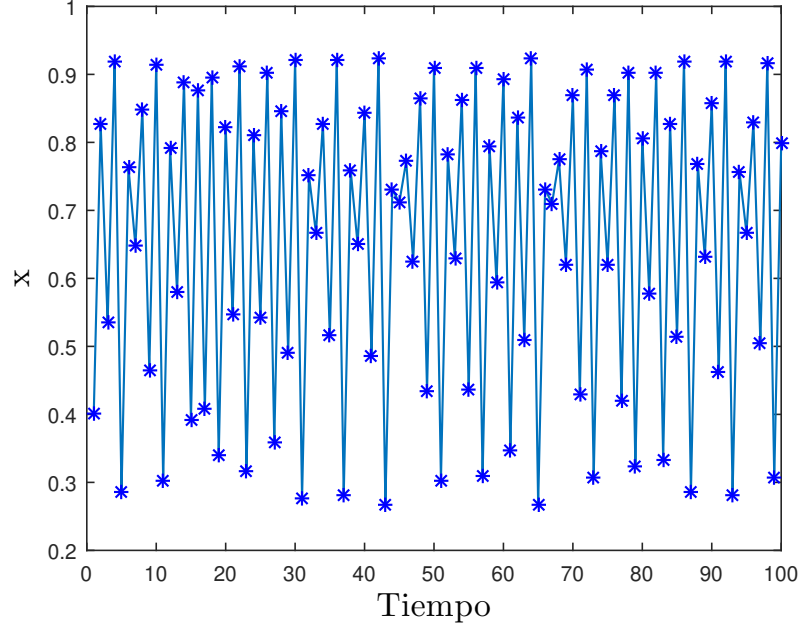


Figura 3.10: Estado x del mapa Badola con dinámicas caóticas discretas.

Para el máximo exponente de Lyapunov se utilizan los valores de $\mu = 3.700000$ y $\varepsilon = 0.09$, con $n = 100$, $x = 0.400000$ y $y = 0.200000$. El resultado es de $\lambda_1 = 0.205471507245726$ y $\lambda_2 = 0.153186969623357$, lo cual indica hypercaos.

3.4.8. Mapa Ushio

Este mapa fue reportado en [48], en dicho trabajo el sistema es utilizado para comprobar un método de sincronización denominado anti fase.

Las ecuaciones representativas del sistema son:

$$x_{n+1} = \rho x_n - x_n^3 + y_n, \quad (3.10a)$$

$$y_{n+1} = \frac{1}{2}x_n, \quad (3.10b)$$

donde $\rho > 0.5$ es el parámetros de control, x_n y y_n son las condiciones iniciales entre $(0 - 1)$. Con estos rangos de valores se obtienen dinámicas caóticas discretas.

Para el cálculo del máximo exponente de Lyapunov, se utilizan los valores de $\rho = 1.9$, con $n = 100$, $x = 0.800000$ y $y = 0.300000$. El resultado es de $\lambda_1 = 0.319915939369701$ y $\lambda_2 = -1.609437912434205$, lo cual indica caos.

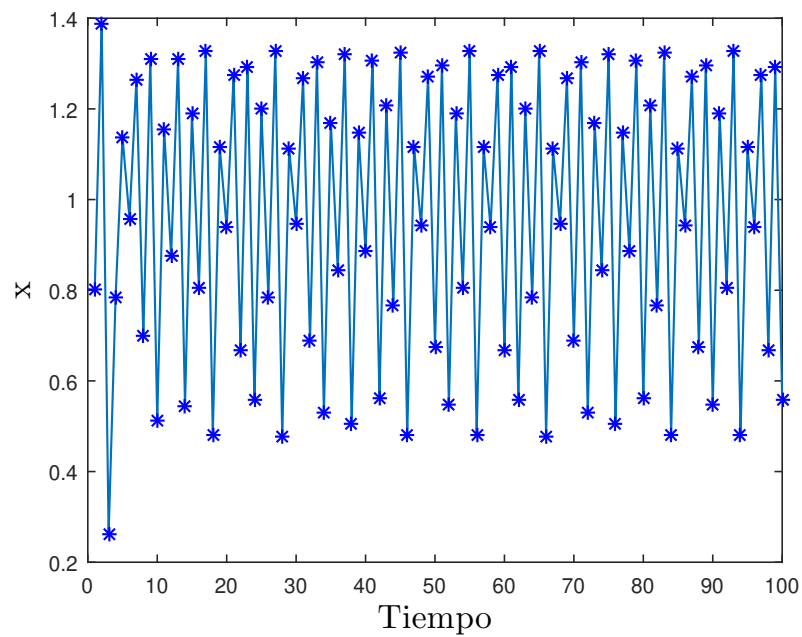


Figura 3.11: Estado x del mapa Ushio con dinámicas caóticas discretas.

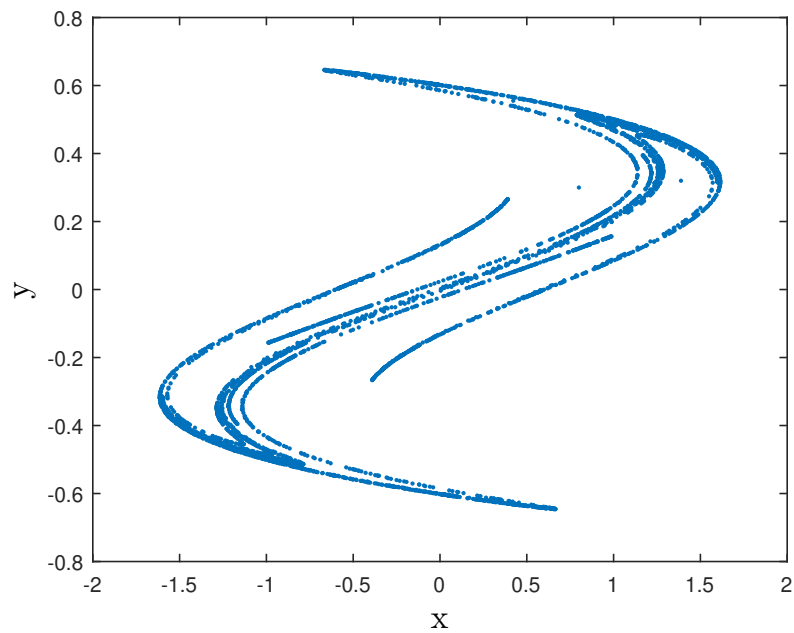


Figura 3.12: Atractor extraño del mapa Ushio.

3.4.9. Mapa Lotka-Volterra

El mapa Lotka-Volterra, son un par de ecuaciones diferenciales de primer orden no lineales que se usan para describir dinámicas de sistemas biológicos en el que dos especies interactúan, también conocido como depredador-presa o presa-predador. Las ecuaciones fueron propuestas de forma independiente por Alfred J. Lotka en 1925 y Vito Volterra en 1926.

El mapa ha sido reportado como hypercaótico en [49], con las siguientes ecuaciones:

$$x_{n+1} = \alpha x_n(1 - x_n) - \beta x_n y_n, \quad (3.11a)$$

$$y_{n+1} = \delta x_n y_n, \quad (3.11b)$$

donde $x_n \in (0, 0.87)$, $y_n \in (0, 1)$ son las condiciones iniciales, α , β y δ , son los parámetros de control, los cuales deben ser $\alpha = 3.5$, $\beta > 1.4$ y $3.7 < \delta < 4$. Con estos rangos de valores se obtienen dinámicas caóticas discretas (ver figura 3.13).

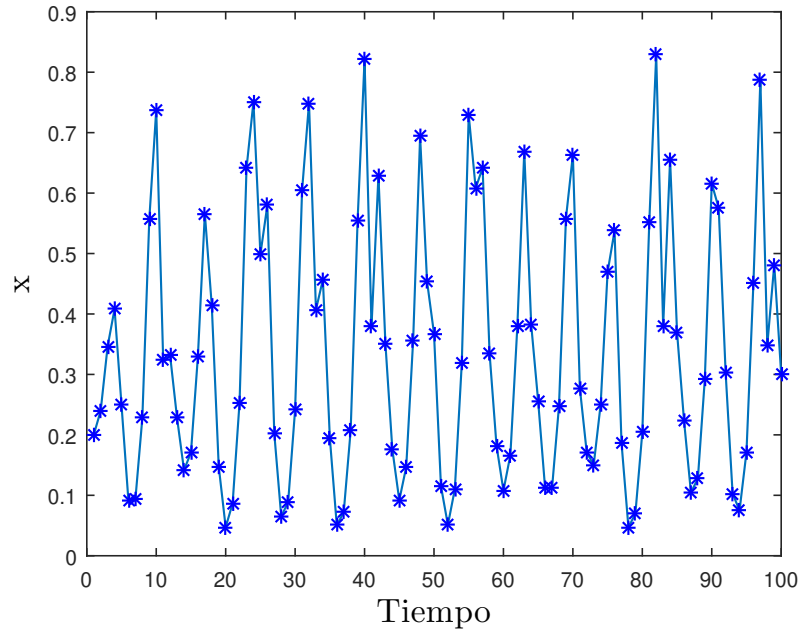


Figura 3.13: Estado x del mapa Lotka-Volterra con dinámicas caóticas discretas.

Para el cálculo del máximo exponente de Lyapunov, se utilizan los valores de $\alpha = 3.5$, $\beta = 3.2$, y $\delta = 3.8$, con $n = 100$, $x = 0.200000$ y $y = 0.500000$.

El resultado es de $\lambda_1 = 0.280374298206975$ y $\lambda_2 = 0.850258121549590$, lo cual nos indica hypercaos.

3.5. Selección de mapa caótico

Una vez confirmado el comportamiento caótico en los mapas presentados anteriormente, se comparan entre ellos para determinar cuál se adecua mejor a las necesidades de un sistema de encriptado. El núcleo de la criptografía digital basada en el caos es la selección de un buen mapa caótico para un esquema de encriptado dado [50]. En realidad, la presencia de caos no garantiza la seguridad de un algoritmo de encriptado. Un buen criptosistema digital basado en caos no debe ser solo la concomitancia de un mapa caótico y una arquitectura de encriptado, sino el resultado de su asociación sinérgica. De hecho, la calidad de un mapa caótico para la criptografía debe evaluarse no solo teniendo en cuenta sus propiedades dinámicas, sino también teniendo en cuenta las necesidades de la arquitectura de encriptado. Las consideraciones para la selección del mapa caótico son las siguientes:

3.5.1. Espacio de claves en función de los parámetros de control y condiciones iniciales

En algunos criptosistemas basados en el caos, los parámetros de control de los sistemas caóticos están determinados por la clave secreta. Si el vínculo entre la clave secreta y los parámetros de control no se establece cuidadosamente, entonces es posible que el sistema caótico evolucione de una manera no caótica, es decir, deben usarse sistemas caóticos robustos, en lugar de los no robustos [51].

Para el espacio de la clave, se considera utilizar una palabra digital de 32 bits en punto flotante para condiciones iniciales y parámetros de control en su implementación digital, para lo siguiente:

- Evitar degradación digital del caos.
- Reducir memoria.
- Reducir procesamiento de datos.

Se utiliza el estándar IEEE 754 (estándar para aritmética en coma flotante), específicamente el formato Binary 32 (formato en coma flotante de simple precisión), el cual el primer bit equivale al bit de signo, los siguientes 8 bits enteros y los 23 bits restantes para almacenamiento con los cuales se pueden trabajar (ver figura 3.14), Este formato proporciona una precisión de 6 a 9 dígitos decimales significativos.

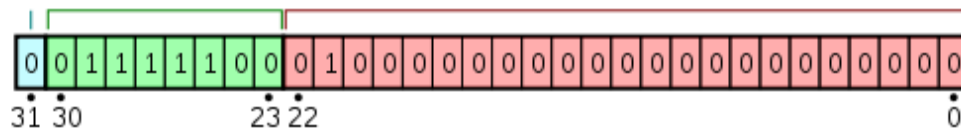


Figura 3.14: Formato binary 32.

Para este análisis, se considera como ejemplo el mapa Badola, en un algoritmo de encriptado caótico que utiliza dos mapas en el proceso de encriptado/desencriptado, con el mapa Badola, se tienen cuatro condiciones iniciales y ocho parámetros de control con variaciones entre $(0,1)$, al tener doce posibles palabras digitales en la cual en cada una podemos almacenar 23 bits, nos un total de 276 bits de almacenamiento para la clave secreta.

En la siguiente tabla se muestra el tamaño de la clave que se puede utilizar con cada mapa caótico en las consideraciones mencionadas anteriormente:

Tabla 3.1: Tamaño máximo de clave de los mapas caóticos analizados.

Mapa caótico	Tamaño de clave (bits)
Logístico	92
Chebyshev	92
Seno	92
Logístico 2D	276
Hénon	184
Seno-Logístico 2D	184
Badola	276
Ushio	138
Lotka-Volterra	230

3.5.2. Tiempo de procesamiento

Se determina el tiempo que le toma al software MatLab en obtener las secuencias caóticas de cada mapa, se obtiene el tiempo para 1000 datos caóticos discretos. Los resultados de esta prueba se observan en la tabla 3.2.

Tabla 3.2: Tiempo en generar 1000 datos caóticos discretos.

Mapa caótico	Tiempo (microsegundos)
Logístico	33
Chebyshev	40.3
Seno	45.4
Logístico 2D	65.7
Hénon	72.2
Seno-Logístico 2D	68.6
Badola	62.4
Ushio	78.8
Lotka-Volterra	63.6

3.5.3. Exponente de Lyapunov

Los exponentes de Lyapunov, son un conjunto de números que se emplean usualmente para detectar la presencia del caos, miden la tasa promedio de divergencia o convergencia exponencial de trayectorias cercanas en el espacio de fase. En este análisis además de comprobar que el sistema es caótico entre mayor sea el valor del máximo exponente de Lyapunov, más sensible será el sistema caótico. En la siguiente tabla se muestra el máximo exponente de Lyapunov de cada mapa caótico.

Tabla 3.3: Máximo exponente de Lyapunov de cada mapa caótico.

Mapa caótico	Lyapunov 1	Lyapunov 2
Logístico	0.693100216810785	x
Chebyshev	0.544992557548549	x
Seno	0.415232364706527	x
Logístico 2D	0.197298083719773	-0.167443932851848
Hénon	0.171836769160141	-1.203972804326008
Seno-Logístico 2D	0.633951719998233	0.509821502321757
Badola	0.205471507245726	0.153186969623357
Ushio	0.319915939369701	-1.609437912434205
Lotka-Volterra	0.280374298206975	0.850258121549590

Espectro de Lyapunov: los sistemas caóticos utilizados como base de los criptosistemas se definen de forma paramétrica, de modo que su dinámica depende de uno o varios parámetros de control. Además, esos sistemas caóticos son sistemas dinámicos que muestran un comportamiento caótico para ciertos valores de los parámetros de control asociados.

3.5.4. Máxima frecuencia de datos caóticos

En algunas arquitecturas de encriptado basadas en el caos, las propiedades de confusión y difusión dependen de la función de distribución de probabilidad de las secuencias derivadas de los sistemas caóticos seleccionados. Si esa distribución no es uniforme e independiente de los valores de los parámetros de control, entonces se reduce la calidad del proceso de difusión (la seguridad disminuye). Como remedio, los mapas caóticos con una función de distribución de probabilidad uniforme deben seleccionarse.

La máxima frecuencia se determina de los histogramas, estos ayudan a representar los datos obtenidos de forma que se pueda contabilizar el número de veces que se repite un valor en la secuencia obtenida. En la siguiente tabla se muestran los datos obtenidos de los histogramas de cada mapa caótico analizado.

Tabla 3.4: Datos repetidos en 100 iteraciones de los mapas caóticos.

Mapa caótico	Estado x	Estado y
Logístico	12	x
Chebyshev	8	x
Seno	9	x
Logístico 2D	14	15
Hénon	18	7
Seno-Logístico 2D	10	11
Badola	10	10
Ushio	50	28
Lotka-Volterra	6	6

3.5.5. Mapa caótico seleccionado

Se selecciona el mapa bidimensional Badola por las siguientes razones:

- Presenta dinámicas hypercaóticas (dos exponentes de Lyapunov positivos).
- Requiere pocas operaciones aritméticas (2 sumas, 4 restas y 6 multiplicaciones).
- No requiere funciones trigonométricas.
- Amplio espacio de claves: 276 bits.
- Tiempo de procesamiento rápido: 62.4 uS.
- Costo de implementación bajo: pocos recursos de memoria.
- No se ha utilizado en criptografía caótica en la literatura.

El mapa Badola presenta dos desventajas las cuales son:

- Distribución de datos caóticos semi uniforme (ver figura 3.15).
- Máximo exponente de Lyapunov bajo: $\lambda_1 = 0.205471507245726$ y $\lambda_2 = 0.153186969623357$.

Para la aplicación criptográfica de esta tesis de maestría, se utiliza el mapa Badola. Además, se le realiza una mejora para aumentar sus propiedades pseudoaleatorias [52], la cual consiste en eliminar los primeros tres dígitos después del punto de los valores caóticos generados por los mapas caóticos. Este proceso transforma todos aquellos valores muy cercanos a 0 (ejemplo 0.001321... a 0.321...) y muy cercanos a 1 (ejemplo 0.999231... a 0.231...), así podemos optimizar el mapa caótico y se obtienen valores con distribución uniforme.

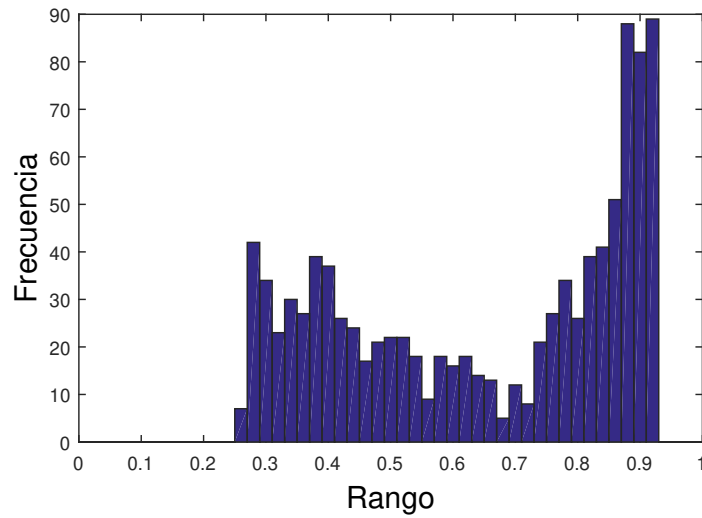


Figura 3.15: Distribución de 1000 datos del mapa Badola original.

Al realizar esta mejora se obtuvieron los dos exponentes de Lyapunov del mapa Badola más grandes (ver tabla 3.5), lo cual hace al mapa mucho más sensible y se mejoró considerablemente la distribución de los datos del mapa Badola (ver figura 3.16).

Tabla 3.5: Comparación del máximo exponente de Lyapunov del mapa Badola original y mejorado.

Mapa caótico	Lyapunov 1	Lyapunov 2
Badola	0.205471507245726	0.153186969623357
Badola mejorado	6.01812760247225	6.160579030992324

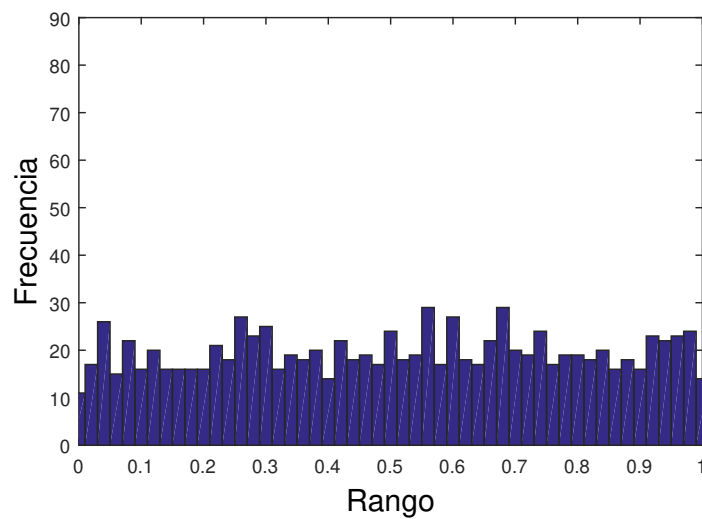


Figura 3.16: Distribución de 1000 datos del mapa Badola mejorado.

3.6. Conclusiones

En este capítulo se mostró la introducción al caos así como los tipos de sistemas caóticos y algunas de sus propiedades más importantes, se analizaron 9 mapas caóticos en donde se observaron sus dinámicas caóticas, además se presentó el cálculo del máximo exponente de Lyapunov para verificar la existencia de caos mediante simulaciones en MatLab y se realizaron cuatro análisis en donde se analizaba el espacio de claves de cada mapa, el tiempo de procesamiento, máximo exponente de Lyapunov y la máxima frecuencia de datos caóticos. En base a los resultados obtenidos se escogió el mapa Badola por las características antes mencionadas en donde además se le realizó una mejora a sus propiedades pseudoaleatorias, el mapa Badola será aplicado en el algoritmo de encriptado caótico para las partes experimentales de este trabajo de tesis de maestría.

Capítulo 4

Criptografía

En este capítulo presenta una introducción a la criptografía donde se describe brevemente su definición y se muestran los tipos de sistemas criptográficos junto con sus principales características. Además, se detallan los riesgos de la criptografía y las bases que debe tener para poder considerarse criptográficamente seguros.

4.1. Introducción

La palabra criptografía proviene del griego “*kryptos*”, que significa esconder y “*grafía*” que significa escribir, es decir, escritura escondida. La criptografía ha sido usada a través de los años para mandar mensajes confidenciales cuyo propósito es que sólo las personas autorizadas puedan entender el mensaje. Alguien que quiere mandar información confidencial aplica técnicas criptográficas para poder esconder el mensaje, manda el mensaje por una línea de comunicación insegura y después solo el receptor autorizado pueda leer el mensaje escondido. La criptografía, además de ser una disciplina que estudia los principios, métodos y medios de transformar los datos para ocultar su significado, garantiza su integridad, establece su autenticidad y previene su repudio [53].

La criptografía, mediante el ocultamiento de la información proporciona las siguientes características [54]:

- Garantiza que sólo las personas autorizadas tienen acceso a la información.
- Utiliza un mecanismo para verificar que la información proviene del lugar indicado y que esta no ha sido modificada.
- Forma de detectar que la información no ha sido alterada por alguien no autorizado.
- Restringir el acceso a la información.
- Servicio que garantiza la autoría del mensaje enviado.

Básicamente, el objetivo de la criptografía es permitir la transmisión de información secreta por un canal público. Este problema de la comunicación segura fue de gran interés en la antigüedad y sigue considerándose vital en la actualidad, debido a los millones de ordenadores intercambiando constantemente información por internet, una red pública al alcance de cualquiera y desde cualquier lugar del mundo.



Figura 4.1: El objetivo de la criptografía es transmitir información de forma segura.

La finalidad de la criptografía es, en primer lugar, garantizar el secreto en la comunicación entre dos entidades (personas, organizaciones, etc.) y en segundo lugar, asegurar que la información que se envía es auténtica en un doble sentido: que el remitente sea realmente quien dice ser y que el contenido del mensaje enviado, habitualmente denominado criptograma, no haya sido modificado en su tránsito.

4.2. Sistemas criptográficos

Un sistema criptográfico está usualmente compuesto por un algoritmo conocido, el cual es dependiente de un parámetro llamado clave secreta para encriptar y desencriptar información entre dos entidades (emisor y receptor).

Definiremos un sistema criptográfico como una quintupla (m, c, K, E, D) (ver figura 4.2), compuesta por los siguientes elementos [55]:

- **m** : representa el conjunto de todos los mensajes sin encriptar (lo que se denomina texto claro) que pueden ser enviados.
- **c** : representa el conjunto de todos los posibles mensajes encriptados o criptogramas.
- **K** : representa el conjunto de claves que se pueden emplear en el criptosistema.
- **E** : es el conjunto de transformaciones de encriptado o familia de funciones que se aplica a cada elemento de m para obtener un elemento de c .

- D : es el conjunto de transformaciones de descryptado, análogo a E .

En todo sistema criptográfico se debe cumplir la siguiente condición

$$D_K(E_K(m)) = m \quad (4.1)$$

Es decir, si un mensaje m se encripta con una función E y una clave K y después se descrypta con la misma clave K , se obtiene el mensaje original m .

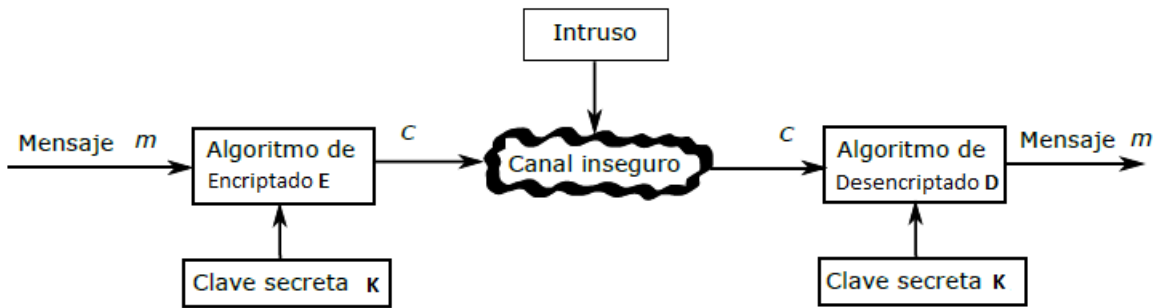


Figura 4.2: Diagrama de un sistema criptográfico.

En criptografía, los seis principios de Kerckhoffs relativos a las propiedades deseables de un sistema criptográfico son [56]:

1. Si el sistema no es teóricamente irrompible, al menos debe serlo en la práctica.
2. La efectividad del sistema no debe depender de que su diseño permanezca en secreto.
3. La clave debe ser fácilmente memorizable de manera que no haya que recurrir a notas escritas.
4. Los criptogramas deberán dar resultados alfanuméricos.
5. El sistema debe ser operable por una única persona.
6. El sistema debe ser fácil de utilizar.

Los sistemas criptográficos se pueden clasificar de la siguiente manera:

Distribución de clave secreta: los sistemas criptográficos se dividen en 2 dependiendo del tipo de clave que utilice, los cuales son [57]:

Criptosistemas simétricos o de clave privada: emplean la misma clave k tanto para encriptar como para descryptar.

La criptografía simétrica ha sido la más usada en toda la historia, ésta ha podido ser implementada en diferentes dispositivos, manuales, mecánicos, eléctricos, hasta los algoritmos actuales que son programables en cualquier computadora.

La idea general es aplicar diferentes funciones al mensaje que se quiere encriptar de tal modo que solo conociendo una clave pueda aplicarse de forma inversa para poder así descryptar.

El esquema general de encriptado y descryptado mediante algoritmos de clave privada (ver figura 4.3). A partir de un documento original se obtiene un documento encriptado al aplicar encriptado con una clave secreta; esa misma clave secreta se utiliza posteriormente para volver a obtener el documento original.

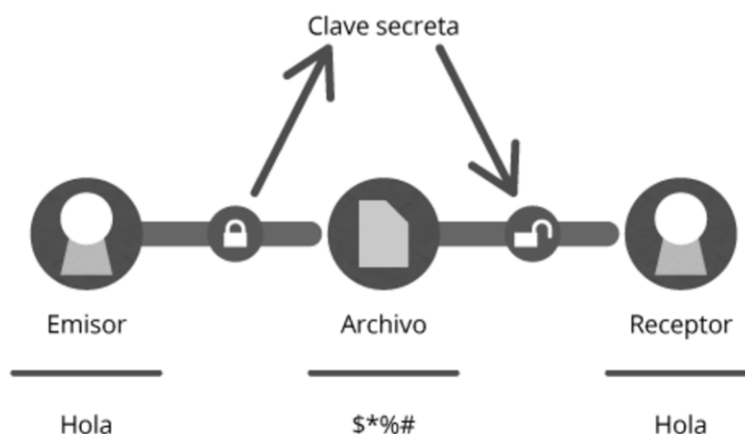


Figura 4.3: Diagrama de encriptado simétrico.

Criptosistemas asimétricos o de clave pública: emplean una doble clave (k_p , k_P), k_p se conoce como clave privada y k_P se conoce como clave pública. Una de ellas sirve para la transformación de encriptado y la otra para la transformación de descryptado. En muchos casos son intercambiables, esto es, si empleamos una para encriptar la otra sirve para descryptar y viceversa. Actualmente la criptografía asimétrica es muy usada; sus dos principales aplicaciones son el intercambio de claves privadas y la firma digital.

La seguridad de una criptografía de dos claves depende de cuestiones matemáticas de una manera que no se aplica a la criptografía de clave secreta y recíprocamente conectan el criptoanálisis con la investigación matemática de nuevas maneras de poder descryptar la información.

El esquema general de un criptosistema asimétrico (ver figura 4.4). A partir de un documento original se obtiene un documento encriptado al aplicar encriptado con una clave secreta; con una clave distinta secreta es utilizada para volver a obtener el documento original.

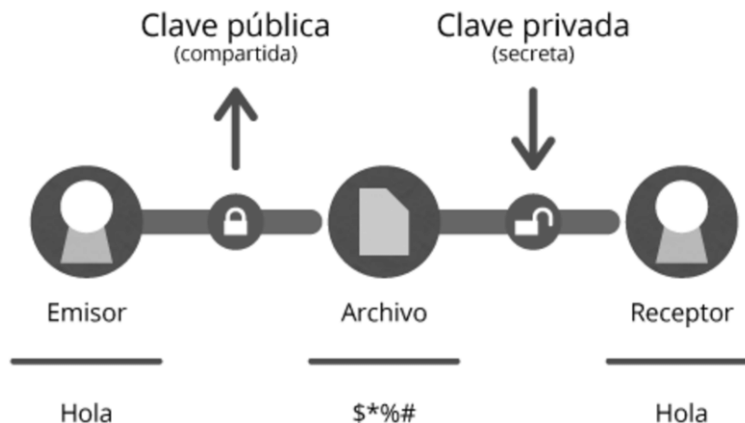


Figura 4.4: Diagrama de encriptado asimétrico.

Operaciones realizadas en el algoritmo: la seguridad depende del emisor y el receptor, dentro del encriptado simétrico existen dos conceptos relacionados con la teoría de la información y las comunicaciones seguras: La confusión y la difusión (el encriptado asimétrico se basa en operaciones matemáticas). Los algoritmos simétricos modernos como 3DES o AES implementan operaciones de confusión y difusión. A continuación, se describen [58]:

Confusión. Consiste en barajar los símbolos del mensaje original colocándolos en un orden distinto, de manera que el criptograma contenga los mismos elementos del texto claro, pero colocados de tal forma que resulten incomprensibles. En recepción, el legítimo receptor, con conocimiento de la transposición, recoloca los símbolos desordenados del criptograma en su posición original.

Difusión. Consiste en establecer una correspondencia entre las letras del alfabeto en el que está escrito el mensaje original y los elementos de otro conjunto, que puede ser el mismo o distinto alfabeto. De esta forma, cada letra del texto claro, se sustituye por su símbolo correspondiente en la elaboración del criptograma. Consiste en cambiar el valor a cada elemento del texto claro de manera desordenada con respecto a la clave secreta, para transformarlo en otro elemento del mismo alfabeto y generar elementos encriptados.

Los sistemas criptográficos también pueden ser clasificados dependiendo de cómo se realice la **estructura del algoritmo** de encriptado, esto es por:

Encriptado en flujo. Los sistemas del encriptado en flujo dividen el sistema a encriptar en caracteres (cuya longitud es menor que los bloques), para posteriormente encriptar cada carácter aplicando una función que varía con el tiempo cuya dependencia temporal está regida por las variables que definen el estado del sistema. Así, después del encriptado de cada carácter, el sistema evoluciona a un nuevo estado.

Encriptado en bloque. Los criptosistemas basados en encriptado en bloque dividen el mensaje a encriptar en diversos fragmentos, por lo general de longitud fija y aplican a cada uno de ellos una transformación criptográfica. La mayoría de estos algoritmos se basan en los conceptos de confusión y difusión. Algunos ejemplos de algoritmos convencionales son el DES, 3DES, RC5, AES.

4.3. Características de la criptografía

El origen de la criptografía se remonta sin duda a los orígenes del hombre, desde que aprendió a comunicarse. Entonces, tuvo que encontrar medios de asegurar la confidencialidad de una parte de sus comunicaciones. El encriptado por difusión y confusión son dos procedimientos de encriptado básico que se ha ido repitiendo en épocas posteriores hasta llegar a nuestros días.

Las herramientas que se utiliza en la criptografía convencional para diseñar los algoritmos de encriptado, son teoría de números, álgebra, curvas elípticas, entre otras, como los ya mencionados 3DES y AES. Por otro lado, existe la criptografía no convencional para algoritmos de encriptado en la que se utilizan herramientas matemáticas en estado de investigación como la criptografía cuántica [59], criptografía con ADN [60] y criptografía caótica [61].

Criptografía cuántica: se basa en el principio de incertidumbre de Heisenberg, esto es, que al observar un sistema cuántico este se perturba a sí mismo, e impide que el observador conozca su estado exacto antes de la observación. Por tanto, si un sistema cuántico se utiliza para transferir información, alguien que quiera espiar la comunicación, o incluso el receptor previsto, podrá verse impedido de obtener toda la información que fue transmitida.

Criptografía ADN: el ADN tiene propiedades que pueden ser útiles en un sistema de criptografía, tales como: capacidad de almacenar mucha información, paralelismo (fenómeno evolutivo, que produce un cambio equivalente en dos ramas de una agrupación contenida en un antepasado común) y poco consumo de potencia.

Criptografía caótica: se basa en ecuaciones no lineales diferenciales o en diferencias, las cuales, generan secuencias desordenadas o caóticas, pero que son deterministas y que presentan sensibilidad a condiciones iniciales. No existe una fórmula simple que, de un sistema caótico en cualquier punto dado, lo que se califica como un problema muy difícil de resolver, dotándolos de una ventaja para su aplicación en la criptografía, eliminando las desventajas fundamentales de la criptografía convencional.

Las características de los sistemas caóticos los hacen aptos para ser utilizados en un sistema criptográfico, pues, además de añadir ventajas al sistema, su aplicación no es tan costosa como lo es en los sistemas de criptografía no convencional de ADN.

Un criptosistema digital debe contener las siguientes propiedades [62]:

1. Se debe describir que sistema caótico utiliza el encriptado.
2. La degradación digital debe ser evaluada, en caso de que se discretize un sistema continuo.
3. El sistema criptográfico debe ser fácil de implementar con base a costos aceptables y buena velocidad de encriptado.
4. La clave secreta debe ser claramente definida.
5. El espacio de claves debe ser especificada solo para generar secuencias caóticas.
6. El efecto avalancha debe producirse para cualquier clave secreta: alta sensibilidad a la clave secreta.
7. Información parcial de la clave secreta, no debe revelar información parcial del texto claro, tampoco de la parte de la clave desconocida.
8. El proceso para generar secuencias caóticas a partir de la clave secreta debe estar claramente definido.
9. El encriptado debe tener alta sensibilidad al texto claro.
10. El encriptado debe generar un texto encriptado con distribución de probabilidad uniforme.

4.4. Seguridad criptográfica

El criptoanálisis es el estudio de los métodos para obtener el sentido de una información encriptada, sin acceso a la información secreta requerida para obtener este sentido normalmente. Típicamente, esto se traduce en conseguir la clave secreta. En el lenguaje no técnico, se conoce esta práctica como romper o forzar el código.

El criptoanálisis también se utiliza para referirse a cualquier intento de sortear la seguridad de otros tipos de algoritmos y protocolos criptográficos en general, y no solamente el encriptado. El criptoanálisis está íntimamente ligado a cada algoritmo de encriptado. Cuando alguien diseña un criptosistema, tiene que tener en cuenta todos los posibles ataques que éste puede sufrir y que cada mecanismo de ocultación que implementa está respondiendo a un hipotético procedimiento de criptoanálisis. El criptoanálisis consiste en comprometer la seguridad de un criptosistema. Esto se puede hacer descryptando un mensaje sin conocer la llave, o bien obteniendo a partir de uno o más criptogramas la clave que ha sido empleada en su codificación, se ocupa de romper esos procedimientos de encriptado para así recuperar la información original [63].

Un criptosistema se puede atacar de muchas formas; la más directa sería la que hace uso únicamente del análisis del mensaje encriptado. Se trata de un análisis pasivo. Pero en la realidad se pueden producir más ataques, apoyados en cierto conocimiento adicional o bien en cierto grado de intervención, en cuyo caso estaremos frente a un ataque activo.

Las posibilidades de éxito para atacar un criptosistema dependen en gran manera de las circunstancias que lo rodean y de la información de que se dispone. No es lo mismo atacar un criptosistema desconocido que uno conocido. Cuando un algoritmo es conocido, hay un ataque posible denominado de fuerza bruta y que consiste en probar todas las claves. Estas formas de ataque obligan a varias precauciones por parte del usuario para no dar facilidades al oponente.



Figura 4.5: Anonymous es el grupo de criptoanalistas con más ataques realizados en los últimos años.

El principio de Kerckhoffs indica que la seguridad de un sistema criptográfico debe recaer en la clave secreta y no sobre el algoritmo de encriptado. El algoritmo de encriptado se considera de dominio público. Un sistema criptográfico se considera vulnerado si un criptoanalista encuentra la forma de determinar la clave secreta y en consecuencia el texto claro. Por otra parte, el sistema criptográfico debe resistir los siguientes ataques, existen tres formas de vulnerar un sistema criptográfico [64]:

1. **Ataques teóricos (lógicos):** aplicar teoría de la información y criptoanálisis para quebrantar el algoritmo. La seguridad se evalúa mediante análisis basados en métodos matemáticos, donde se muestra que el mensaje encriptado y clave secreta no pueden ser revelados al implementar ataques conocidos.
2. **Ataques físicos:** vulnerabilidades del sistema criptográfico, que se pueden aprovechar para quebrantar el algoritmo mediante ataques físicos, por ejemplo, la información de tiempo, el consumo de energía, fugas electromagnéticas o incluso sonido, pueden proporcionar fuentes adicionales de información, que puede aprovecharse para romper el sistema criptográfico.
3. **Ataques humanos:** presionar mediante sobornos u otros medios a personas que poseen información privilegiada.

Algunos ataques de tipo teórico considerados poderosos son:

- Ataque exhaustivo: son ataques donde se tratan todas las posibles combinaciones de claves, por lo que, la clave debe contener más de 2^{100} opciones.
- Solo texto encriptado: en este ataque, el criptoanalista conoce el algoritmo y texto encriptado, pero no la clave secreta. Prácticamente es un ataque exhaustivo, por lo que se requiere que el espacio de claves sea suficientemente grande para resistir un ataque exhaustivo.
- Texto claro conocido: el criptoanalista conoce el texto claro de un texto encriptado y utiliza esta información para intentar determinar la clave secreta junto al algoritmo y así, descriptar otros criptogramas. Si el algoritmo cae ante este ataque, se considera no seguro.
- Texto claro elegido: es otro ataque diferencial poderoso en donde el criptoanalista elige su propio texto claro para encriptar, posteriormente hace una ligera modificación (un bit) al texto claro y se vuelve a encriptar para determinar una relación entre la entrada y salida para determinar la clave secreta y descriptar otros criptogramas.

Algunos ataques de tipo físico considerados poderosos son:

- Ataque de sincronización: ataques basados en medir la cantidad de tiempo que diversos cálculos llevan a cabo.
- Ataque de monitoreo de energía: ataques que hacen uso de consumo de energía en el hardware durante el cálculo.
- Ataques electromagnéticos: ataques basados en la radiación electromagnética filtrada que pueden proporcionar directamente textos claros y otra información. con un análisis de potencia se puede determinar la clave.

4.5. Conclusiones

La criptografía ha estado presente desde que el hombre quiso tener más seguridad en el intercambio de información, pero al querer tener seguridad en la información fue por que otras personas estaban interesadas en dicha información y así fue cómo surgió el criptoanálisis, la criptografía y el criptoanálisis han crecido a la par desde su existencia y han evolucionado con el tiempo.

Para este trabajo de tesis de maestría se desarrollará un algoritmo criptográfico basado en criptografía no convencional utilizando encriptado caótico, como se analizó los tipos de algoritmos en el apartado de sistema criptográficos. También, el algoritmo se basará en una única clave de encriptado y descriptado, utilizará operaciones de confusión y difusión para realizar un encriptado en flujo.

Capítulo 5

Sistemas embebidos

En este capítulo, se da una introducción a los sistemas embebidos y a su principal componente: el microcontrolador, se muestran las principales características de un sistema embebido y se presenta el microcontrolador que será utilizado en el trabajo experimental para este trabajo de tesis de maestría.

5.1. Introducción

Un *sistema embebido* es un sistema de computación diseñado para realizar una o algunas funciones dedicadas en un sistema en tiempo real. Al contrario de lo que ocurre con los ordenadores de propósito general (como por ejemplo una computadora personal o PC) que están diseñados para cubrir un amplio rango de necesidades, los sistemas embebidos se diseñan para cubrir necesidades específicas. En un sistema embebido la mayoría de los componentes se encuentran incluidos en la placa base (ver figura 5.1).

Un sistema embebido estaría formado por un microcontrolador y un software que se ejecute sobre éste, el término embebido hace referencia al hecho de que el microcontrolador está encerrado o instalado dentro de un sistema mayor y su existencia como microcontrolador puede no ser aparente. Todo sistema embebido necesitará una cierta cantidad de memoria, la cual puede incluso encontrarse dentro del mismo chip, por ejemplo, las memorias RAM, ROM o Flash, esta última en donde el programa se guarde para luego ser ejecutado en una aplicación específica. También, cuenta con una serie de salidas y entradas necesarias para comunicarse con el mundo exterior. Sus líneas de entrada/salida (I/O) soportan las conexiones de los sensores y actuadores del dispositivo a controlar y todos los recursos complementarios disponibles tienen como finalidad atender a sus requerimientos [65].

Estas son las únicas características que tienen en común los sistemas embebidos, todo lo demás será totalmente diferente para cada sistema embebido en particular debido a la variedad de aplicaciones disponibles en las cuales se pueden implementar los sistemas embebidos para cubrir ciertas necesidades.

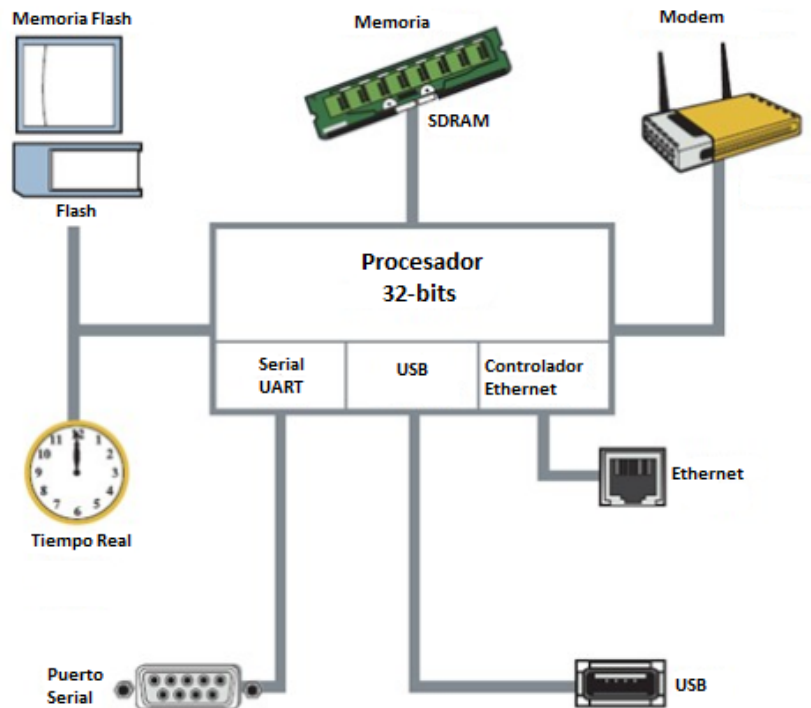


Figura 5.1: Diagrama de un sistema embebido a nivel físico.

Los sistemas embebidos se pueden programar directamente en el lenguaje del microcontrolador incorporado sobre el mismo, o también, utilizando los compiladores específicos, pueden utilizarse lenguajes como C o C++. Los sistemas embebidos cuentan con características que los definen algunas de las características básicas de los sistemas embebidos son las siguientes [66]:

- Bajo costo.
- Bajo consumo de potencia.
- Deben ser confiables.
- La confiabilidad: $R(t)$, es la probabilidad de que el sistema trabaje correctamente.
- La mantenibilidad: $M(d)$, es la probabilidad de que el sistema vuelva a trabajar correctamente después de un fallo.
- La disponibilidad: $A(t)$, es la probabilidad de que el sistema esté funcionando en el tiempo necesario.
- La seguridad informática, consiste en disponer de una comunicación confidencial y autenticada.

El diseño de un producto que incorpora sistemas embebidos está orientado a minimizar los costos y maximizar la confiabilidad, pero también es imprescindible incorporar en el diseño consideraciones de seguridad, incluyendo funciones y protocolos criptográficos que protejan la información durante todas las fases. Los sistemas embebidos a menudo operan en un ambiente dedicado con condiciones operacionales y escenarios muy específicos. Es importante que dichas condiciones y amenazas se tengan en cuenta cuando se diseñan las funciones de seguridad.

5.2. Microcontrolador

Un microcontrolador es un circuito integrado capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, los cuales cumplen una tarea específica. Incluye en su interior las tres unidades funcionales principales: Unidad central de procesamiento (CPU), memoria y periféricos de entrada y salida. A diferencia de los microprocesadores de propósito general, como los que se usan en las computadoras personales (PC), los microcontroladores son unidades auto suficientes, proporcionan un sistema programable de bajo costo, así como un control sofisticado en ciertas aplicaciones. El funcionamiento de los microcontroladores está determinado por el programa almacenado en su memoria y cuenta con un reset que permite reiniciar el microcontrolador en cualquier momento para que vuelva al inicio del programa. Este puede escribirse en distintos lenguajes de programación. Además, la mayoría de los microcontroladores actuales pueden programarse repetidas veces hasta cumplir un cierto límite de veces que se pueden programar [67].

Por las características antes mencionadas, los microcontroladores son ampliamente utilizados como el cerebro de una gran variedad de sistemas embebidos que controlan máquinas, componentes de sistemas complejos, aplicaciones industriales de automatización, robótica, domótica, equipos médicos, sistemas aeroespaciales, e incluso dispositivos de la vida diaria como automóviles, hornos de microondas, teléfonos y televisores [68].

Para desarrollar una aplicación basada en microcontroladores se compone de ciertas etapas a seguir para lograr el objetivo, las etapas principales son las siguientes:

- Desarrollo de software: esta etapa corresponde a la escritura y compilación del programa que rige las acciones del microcontrolador y los sistemas periféricos a controlar conectados a este.
- Programación del microcontrolador: en esta etapa el código de máquina correspondiente al programa desarrollado en la etapa anterior se descarga en la memoria del microcontrolador.
- Prueba y verificación: por último, el microcontrolador debe someterse a pruebas para asegurar el funcionamiento correcto del programa y los periféricos conectados al microcontrolador.

Las principales características de los microcontroladores son las siguientes:

- Unidad de Procesamiento Central (CPU): típicamente de 8 bits, pero también las hay de 4, 32 y hasta 64 bits, con bus de datos separada de las instrucciones y con el bus de datos e instrucciones compartidas.
- Memoria de Programa: puede ser una memoria ROM, EPROM, EEPROM o Flash que almacena el código del programa. En ella se encuentra almacenado el código de los programas que el sistema puede ejecutar, así como los datos.
- Generador del Reloj: usualmente un cristal de cuarzo de frecuencias que genera una señal oscilatoria.
- Interfaz de Entrada/Salida: son puntos en los que los periféricos se pueden conectar y pueden intercambiar información con la memoria y el procesador central.

Además de las características básicas descritas pueden llevar añadidas otras más sofisticadas que incrementan la potencia del microcontrolador. Distintas compañías han desarrollado familias microcontroladores, de los cuales cada uno cuenta con características generalmente similares con algunos cambios entre ellos, en la siguiente tabla se comparan las características generales de tres familias de microcontroladores de diferentes compañías [69–71].

Tabla 5.1: Comparación de tres familias de microcontroladores.

Características	NXP (MK53N512)	ATMEL (ATmega 8515)	Microchip (PIC16F877A)
Bits	32	8	16
SRAM	32kb	512b	368b
Flash	128kb	8kb	8kb
Reloj	50Mhz	16MHz	20MHz
Entradas (I/O)	64	35	33

5.3. Microcontrolador MCF51MM256

Para los trabajos experimentales de este trabajo de tesis de maestría se utiliza la torre TWR-MCF51MM [72] (ver figura 5.2), diseñada para conectar microcontroladores y módulos periféricos, cuenta con dos módulos de elevador que proporcionan regulación de potencia a los circuitos y la integridad estructural necesarios para todas las configuraciones de un sistema embebido de torre ensamblado.

La placa elevadora, proporciona circuitos de regulación de potencia y entradas de potencia desde un conector USB mini-B estándar, cuenta con cuatro conectores de borde de tarjeta de 164 pines para conectar el microcontrolador y dos conectores de 2 x 80 en la parte posterior para facilitar el acceso de la placa de montaje lateral donde se conectan periféricos externos.



Figura 5.2: Torre TWR-MCF51MM utilizada para trabajar con el microcontrolador MCF51MM256.

El microcontrolador MCF51MM256 de Freescale (ver figura 5.3), proporciona potencia de baja operación, conectividad USB y medición de datos sin igual, todo en un solo microcontrolador de 32 bits, permitiendo utilizarlos en dispositivos a un menor costo, cuenta con un reloj de 50 MHz, 256 kb de memoria flash, 32 kb de SRAM, proporciona módulos ADC y DAC de alta resolución, comunicación I2C, puertos de entrada y salida (GPIO) y amplificadores operacionales de propósito general.

El conjunto de este microcontrolador altamente integrado incluye un controlador USB 2.0 con host, serial múltiple y una interfaz de bus externo (ver figura 5.4) [73].



Figura 5.3: Microcontrolador de 32 bits MCF51MM256 utilizado en el trabajo experimental de este trabajo de tesis.

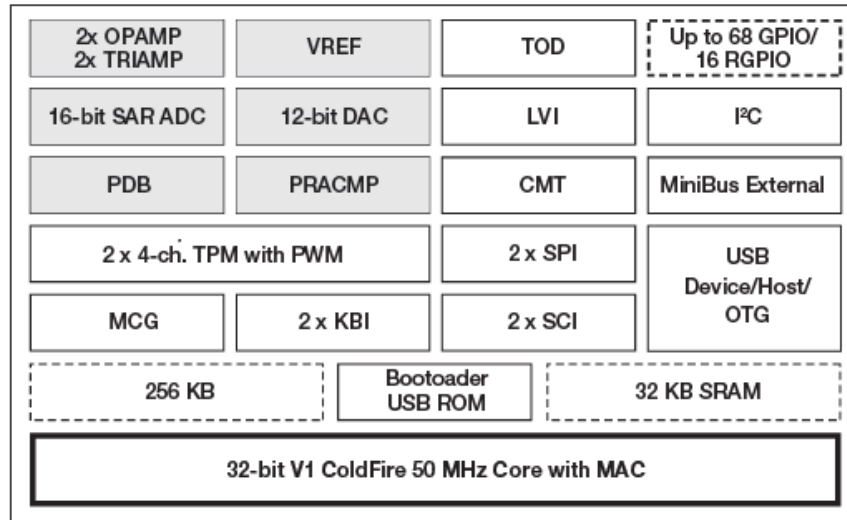


Figura 5.4: Diagrama a bloques del microcontrolador MCF51MM256.

El microcontrolador MCF51MM256 es ideal para aplicaciones médicas como:

- Presión de sangre.
- Ritmo cardíaco.
- Glucosa.
- Pulsioxímetro.

5.4. Módulo de ECG

Para aplicaciones del ritmo cardíaco y electrocardiografía, el microcontrolador MCF51MM256 cuenta con un conector médico de 20 pines para módulos externos. El módulo MED-ECG [74] (ver figura 5.5), es una placa de desarrollo de bajo costo que permite realizar rápidamente mediciones de electrocardiograma.

Utiliza los componentes analógicos externos o los módulos analógicos en el microcontrolador (ver figura 5.6), cuando se trata de medir la frecuencia cardíaca o mostrar un electrocardiograma, la primera consideración es la adquisición de una señal precisa y silenciosa con suficiente amplitud para ser cuantificada.

Mediante el uso de electrodos aplicados en varias regiones del cuerpo se puede obtener, se obtiene un registro de las descargas eléctricas del corazón, pasan mediante una serie de filtros y amplificadores operacionales, hasta ser enviada la señal mediante comunicacional I2C al microcontrolador.

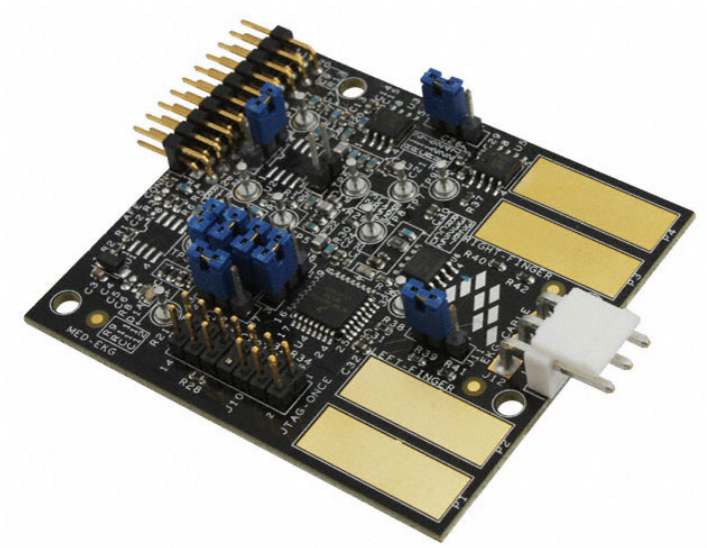


Figura 5.5: Módulo MED-ECG.

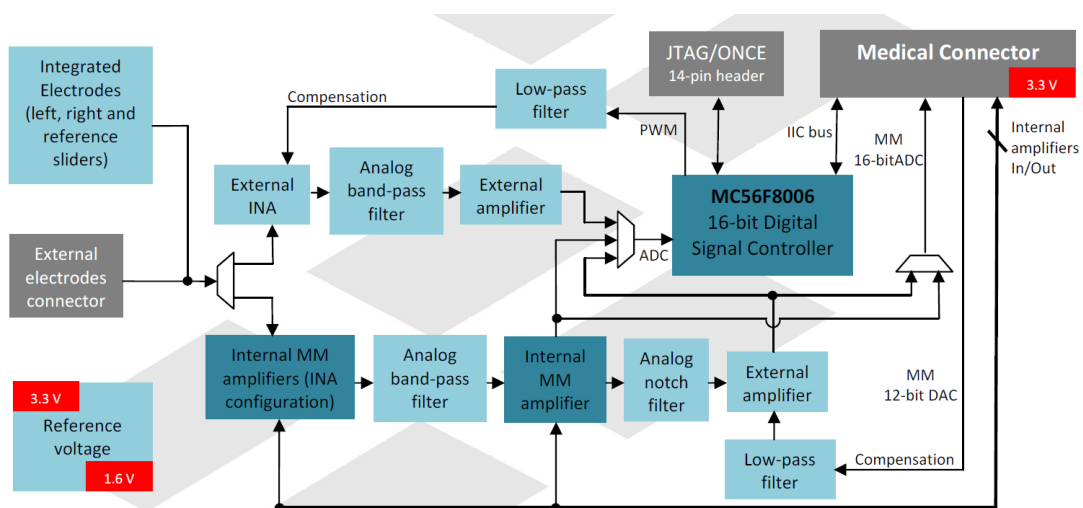


Figura 5.6: Diagrama a bloques del módulo MED-ECG.

5.5. Software CodeWarrior

La versión 6.3 del software CodeWarrior (ver figura 5.7), es un entorno de desarrollo integrado completo (IDE) que proporciona un marco altamente visual y automatizado para acelerar el desarrollo de las aplicaciones integradas más complejas, para editar, compilar y depurar el software para varios microcontroladores y microprocesadores (Freescale ColdFire, ColdFire +, Kinetis, Qorivva) en los lenguajes de programación C y C++.

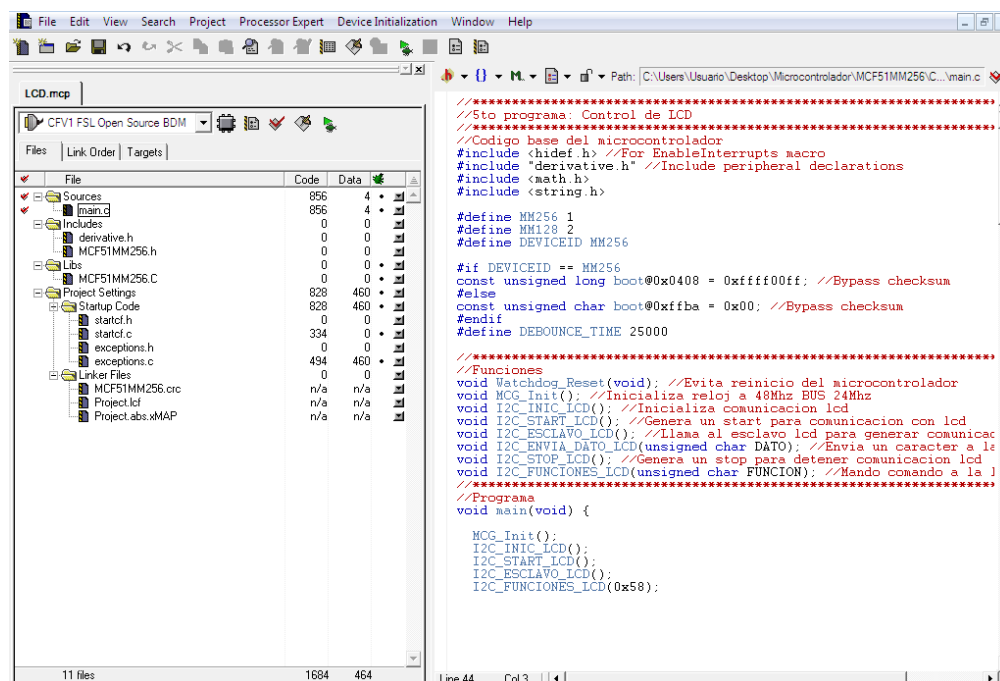


Figura 5.7: Interfaz del software CodeWarrior.

5.6. Conclusiones

En conclusión, el microcontrolador es un dispositivo que puede desempeñar distintas tareas específicas según su programación, siendo éste muy versátil dado que puede interactuar con el mundo exterior a través de sus periféricos adaptados según sea el requerimiento. Todo esto aunado a su pequeño tamaño de encapsulado y a su bajo costo comparado con otros equipos, además de poder ser utilizado en un sistema embebido para potencializar su funcionamiento.

El uso de sistemas embebidos en productos complejos implica un desafío de la seguridad para proteger la información contenida en el sistema embebido y también la que es transmitida desde y hacia el dispositivo por redes públicas o internet. Por tanto, cabe incluir funciones criptográficas, así como servicios de pruebas de seguridad y evaluaciones específicas para sistemas embebidos.

Algoritmo de encriptado caótico para señales biomédicas

6.1. Introducción

El resguardo de la seguridad tanto en la comunicación por cable como en la inalámbrica ha adquirido gran importancia en la transferencia de datos. El contenido multimedia y visual se ha utilizado con mucha frecuencia en muchos campos, como compartir información personal, militar y médica. Es por eso que se han iniciado estudios sobre el desarrollo de diferentes métodos para el encriptado de información [75].

52



Figura 6.2: Seguridad médica en redes de telemedicina.

La transmisión y el almacenamiento de información médica en canales de comunicación inseguros es cada día mayor y presentan una serie de riesgos, por lo cual se utiliza el encriptado de información, la cual además de que la información se considere segura, el algoritmo de encriptado debe ser eficaz para encriptar los datos de una manera rápida además de requerir poca memoria para que pueda ser aplicado en sistemas embebidos.

El algoritmo de encriptado propuesto utilizado en este trabajo, se basa en las siguientes características criptográficas [78]:

- *Encriptado simétrico*. El algoritmo utiliza la misma clave secreta para encriptar y desencriptar.
- *Arquitectura de confusión y difusión*. El algoritmo utiliza procesos para cambiar de posición y cambiar de valor a cada elemento claro en una sola operación.
- *Encriptado a flujo*. El algoritmo encripta cada elemento de la señal clara, uno a la vez hasta terminarlo.
- *Encriptado no convencional*. El algoritmo utiliza secuencias caóticas del mapa Badola, que son determinadas por la clave secreta de 276 bits para generar secuencias pseudoaleatorias para realizar el proceso de confusión y difusión.

En la figura 6.3, se muestra el diagrama a bloques del proceso de encriptado propuesto. El mapa Badola 2 se itera en base a la clave secreta, después se obtiene el valor de Z relacionado con la señal clara y secuencias caóticas del mapa Badola 2, posteriormente se itera el mapa Badola 1 en base a la clave secreta y el valor de Z para realizar los procesos de confusión y difusión sobre la señal clara y finalmente se agrega el valor de Z al criptograma para que el usuario autorizado pueda descryptar correctamente la señal.

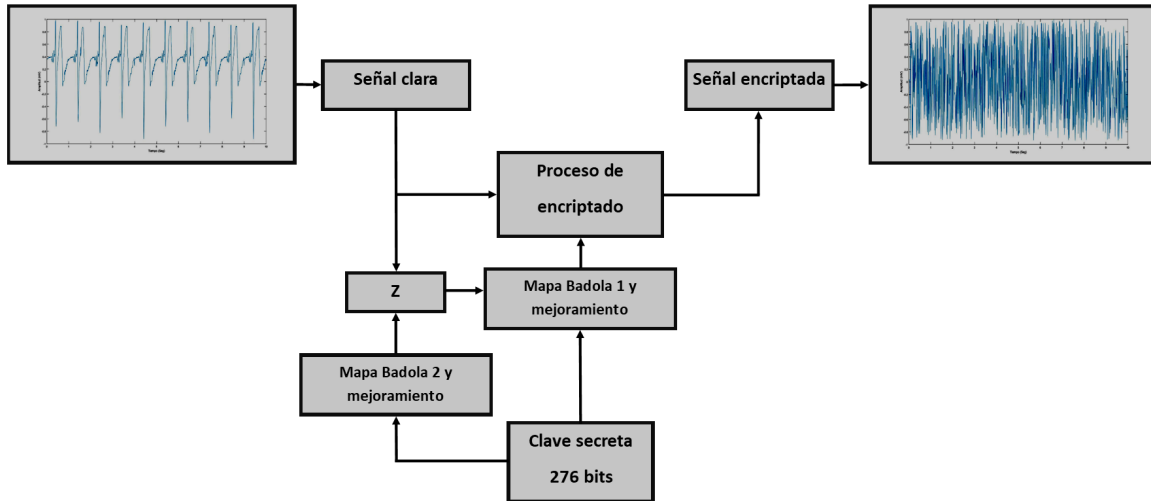


Figura 6.3: Diagrama a bloques del proceso de encriptado.

El proceso de desencriptado consta en invertir el proceso de encriptado. En la figura 6.4, se muestra el diagrama a bloques del proceso de desencriptado. Primero el valor de Z se extrae del criptograma (Z no se calcula ni se iteran datos caóticos del mapa Badola 2), después los datos caóticos son calculados del mapa Badola 1 con el uso de la clave secreta y el valor de Z , finalmente, se realizan los procesos de confusión y difusión inversos para recuperar la señal.

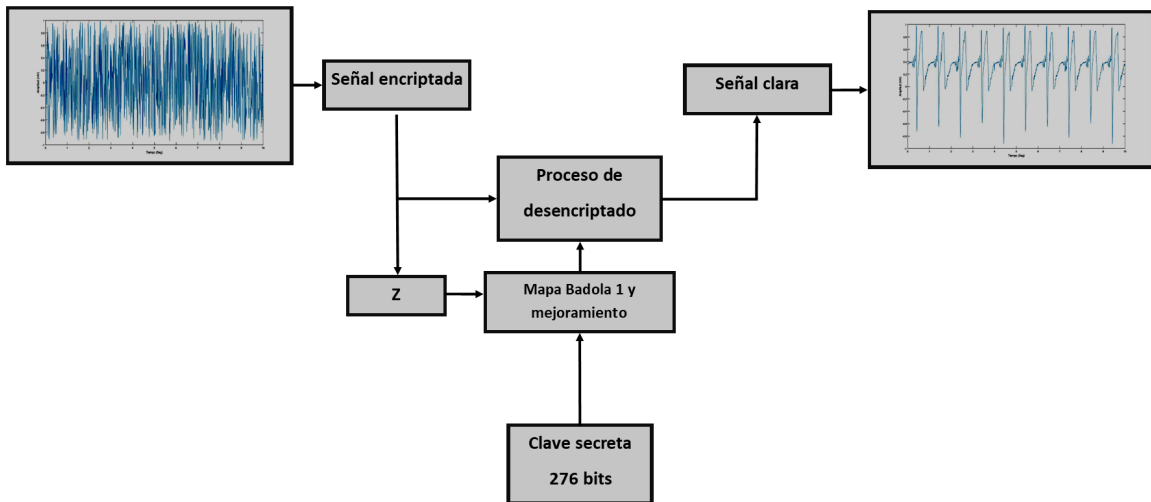


Figura 6.4: Diagrama a bloques del proceso de desencriptado.

6.2. Algoritmo de encriptado caótico

A continuación, se muestran los pasos que componen el algoritmo de encriptado caótico propuesto, el cual está basado en el algoritmo presentado en [79].

6.2.1. Definición de la clave secreta

La clave secreta está definida como una secuencia de 276 bits, caracterizada por 69 caracteres hexadecimales $K \in (0 - 9, A - F)$; la cual es dividida en 12 secciones $(A, B, C, D, E, F, G, H, I, J, K, L)$. Las condiciones iniciales y el valor de los parámetros de control son calculados de manera indirecta mediante la clave secreta. En la tabla 6.1, se muestran los cálculos correspondientes.

Tabla 6.1: Clave secreta.

Clave secreta	Parámetro de control			Condiciones iniciales
69 dígitos Hex	$H_1, H_2, \dots, H_{69}$ donde $H \in [0 - 9, A - F]$			
Cálculos	$A = \frac{(H_1, H_2, \dots, H_6)_{10}}{2^{32}+1}$	$B = \frac{(H_7, H_8, \dots, H_{12})_{10}}{2^{32}+1}$	$C = \frac{(H_{13}, H_{14}, \dots, H_{18})_{10}}{2^{32}+1}$	
	$D = \frac{(H_{19}, H_{20}, \dots, H_{24})_{10}}{2^{32}+1}$	$E = \frac{(H_{25}, H_{26}, \dots, H_{30})_{10}}{2^{32}+1}$	$F = \frac{(H_{31}, H_{32}, \dots, H_{36})_{10}}{2^{32}+1}$	
	$G = \frac{(H_{37}, H_{38}, \dots, H_{42})_{10}}{2^{32}+1}$	$H = \frac{(H_{43}, H_{44}, \dots, H_{48})_{10}}{2^{32}+1}$	$I = \frac{(H_{49}, H_{50}, \dots, H_{54})_{10}}{2^{32}+1}$	
	$J = \frac{(H_{55}, H_{56}, \dots, H_{60})_{10}}{2^{32}+1}$	$K = \frac{(H_{61}, H_{62}, \dots, H_{66})_{10}}{2^{32}+1}$	$L = \frac{(H_{67}, H_{68}, \dots, H_{69})_{10}}{2^{32}+1}$	
Badola 1	$\mu_{11} = \text{mod}[(A + B + Z), 1] * 0.01]$			$x_{10} = \text{mod}[(I + J + Z), 1] \quad y_{10} = \text{mod}[(K + L + Z), 1]$
Badola 2	$\mu_{21} = \text{mod}[(A + B), 1] * 0.01]$			$x_{20} = \text{mod}[(I + J), 1] \quad y_{20} = \text{mod}[(K + L), 1]$

6.2.2. Cálculo de Z

El valor Z incrementa la sensibilidad a pequeños cambios en la señal clara y a la clave secreta a nivel de bit. Al utilizar el valor de Z hace que el proceso de encriptado sea robusto ante ataques diferenciales como texto claro conocido y ataque de texto claro elegido. Para determinar el valor de Z , todos los elementos de la señal clara se suman con la secuencia de datos caóticos del mapa Badola 2. Primero, el mapa Badola 2 es iterado I_2 veces con el parámetro de control μ_{21} y condiciones iniciales x_{20}, y_{20} , para generar una secuencia caótica de datos $x^{B2} = x_1^{B2}, x_2^{B2}, x_3^{B2}, \dots, x_{I_2}^{B2}$ y $y^{B2} = y_1^{B2}, y_2^{B2}, y_3^{B2}, \dots, y_{I_2}^{B2}$ con $x^{B2} \in (0, 1)$ y una precisión decimal de 10^{-15} .

Posteriormente, la secuencia caótica x^{B2} es mejorada en uniformidad con la siguiente expresión:

$$x_i^{B2} = (x_i^{B2} * 1000) \quad (\text{mód } 1), \quad \text{para } i = 1, 2, 3, \dots, I_2, \quad (6.1)$$

donde I_2 es el número de iteraciones del mapa Badola 2 y mod es la operación de módulo.

Después, todos los elementos de la señal clara se suman con x^{B2} como sigue

$$S1 = \{S1 + [P_i * x_i^{B2}] + x_i^{B2}\} \quad (\text{mód } 1), \quad \text{para } i = 1, 2, 3, \dots, I_2, \quad (6.2)$$

donde P_i representa el elemento i de la señal clara, $S1$ es una variable inicializada en cero y x^{B2} corresponde a la secuencia caótica.

6.2.3. Proceso de encriptado

El mapa Badola 1 se itera I_1 veces con valores μ_{11} , x_{1_0} y y_{1_0} tomados de la tabla 6.1, para generar la segunda secuencia caótica de datos $x^{B1} = x_1^{B1}, x_2^{B1}, x_3^{B1}, \dots, x_{I_1}^{B1}$ y $y^{B1} = y_1^{B1}, y_2^{B1}, y_3^{B1}, \dots, y_{I_1}^{B1}$ con $x^{B1} \in (0, 1)$ y una precisión decimal de 10^{-15} .

Posteriormente, la secuencia x^{B1} es mejorado con la siguiente expresión:

$$x_i^{B1} = (x_i^{B1} * 1000) \pmod{1}, \text{ para } i = 1, 2, 3, \dots, I_1, \quad (6.3)$$

donde I_1 es el número de iteraciones para el mapa Badola 1.

De la secuencia caótica x^{B1} se determinan dos subsecuencias para los procesos de confusión y difusión. Para el proceso de confusión la subsecuencia se calcula con la siguiente expresión:

$$CF_i = \text{round} [x_{I_1-\ell+i}^{B1} * (\ell - 1)] + 1, \text{ para } i = 1, 2, 3, \dots, \ell, \quad (6.4)$$

donde ℓ es la longitud requerida y $CF \in (1, \ell)$ es el vector pseudoaleatorio para realizar el proceso de confusión. En un proceso de confusión eficiente, todos los elementos del texto claro se deben permutar entre sí mismos; sin embargo, la ec. (6.4), genera valores para re-posicionamiento repetido. Por tanto, los valores repetidos de CF son cambiados mediante programación como sigue

$$G_h = [K_h], \text{ con } h \ll \ell, \quad (6.5)$$

donde K_h es el valor que no está en CF ordenados de menor a mayor. El vector de valores repetidos G se divide en dos secciones y cada valor se asigna a CF de manera alternada donde un valor repetido aparece. Cuando este proceso termina, se tiene un vector para confusión con todas las posibles posiciones.

Una subsecuencia para difusión se determina de x^{B1} de la misma longitud ℓ . Aunque el mapa Badola presenta una mejor distribución de los datos, aún hay tendencias marcadas hacia ciertos valores, lo cual puede llevar en un proceso de difusión ineficiente. Una solución a este problema es eliminar los primeros tres decimales de los datos caóticos, proceso que se realiza sólo para una longitud determinada por ℓ y generar un proceso de difusión optimizado. La subsecuencia para difusión se calcula como:

$$DF_i = (x_{I_1-\ell+i}^{B1} + Z) \pmod{1}, \text{ para } i = 1, 2, 3, \dots, \ell, \quad (6.6)$$

donde $DF \in (0, 1)$ es el vector pseudoaleatorio para el proceso de difusión con ℓ .

El proceso de encriptado se calcula con la siguiente expresión:

$$E_i = P(CF_i) + DF_i, \text{ para } i = 1, 2, 3, \dots, \ell, \quad (6.7)$$

donde P es la señal clara y E es el criptograma.

6.2.4. Proceso de descriptado

El proceso de descriptado consiste en invertir todos y cada uno de los pasos desarrollados en el encriptado. Se debe utilizar exactamente la misma clave de 276 bits, ya que, si un bit cambia, no se podrá recuperar el mensaje claro correctamente.

Primero, el valor de Z debe ser recuperado, después, el mapa Badola 1 es iterado con la clave secreta y el valor de Z . Posteriormente, se calculan las subsecuencias CF y DF para confusión y difusión, respectivamente. Finalmente, el descriptado se realiza con la siguiente expresión:

$$D(CF_i) = E_i - DF_i, \text{ para } i = 1, 2, 3, \dots, \ell, \quad (6.8)$$

donde E es el criptograma y D es el mensaje recuperado.

6.2.5. Características de seguridad y eficiencia

El algoritmo de encriptado caótico propuesto en este trabajo de tesis, posee ciertas características de seguridad que aportan robustez ante los ataques criptoanalíticos. Estas características se muestran a continuación:

1. **Inicialización de secuencias caóticas.** La condición inicial y parámetro de control del mapa Badola, se determinan de manera indirecta a partir de una clave de 276 bits.
2. **Mejoramiento de secuencias caóticas.** Los datos caóticos del mapa Badola son modificados mediante una simple operación para generar una mejor distribución, lo que genera mejores procesos de confusión y difusión. Por tanto, un criptograma con mejores propiedades estadísticas.
3. **Cálculo del Z .** Al considerar las características de la señal clara para realizar el proceso de encriptado, se incrementa por mucho la sensibilidad a pequeños cambios en la señal clara. Además, se utilizan datos caóticos de un segundo mapa Badola para evitar ataques de sólo texto claro elegido. Por tanto, este proceso ayuda a dar robustez al algoritmo criptográfico ante los distintos ataques.
4. **Confusión y difusión optimizados.** Los vectores para el encriptado se calculan de tal forma que la confusión es 100 % aplicada en toda la señal clara. Mientras, el proceso de confusión se aplica sobre la señal clara con datos caóticos que presentan una distribución uniforme, lo que genera un encriptado con excelentes características estadísticas.
5. **Eficiencia de encriptado.** El sistema caótico posee ventajas de implementación tanto en velocidad de generación de datos como poco espacio de memoria requerido. Además, los procesos de confusión y difusión son aplicados a la señal clara en un mismo proceso.

6.3. Conclusiones

Un algoritmo criptográfico basado en caos debe cumplir con varios aspectos de seguridad para poder ser implementado en aplicaciones prácticas como en sistemas embebidos en donde se requiere resguardar o enviar información de forma segura. El algoritmo propuesto se basa en encriptado simétrico ya que utiliza la misma clave para encriptar y desencriptar, realiza procesos de confusión y difusión en una sola operación, encripta la secuencia de datos en flujo y utiliza secuencias caóticas del mapa badola. Además, se presentaron las características de seguridad y eficiencia que el algoritmo posee así aportar más robustez ante ataques criptoanalíticos.

Capítulo 7

Implementación de encriptado caótico en microcontrolador

Actualmente, muchos sistemas embebidos están conectados a internet para transmitir información privada o simplemente para almacenamiento de datos. Toda esta información sensible requiere ser encriptada antes de su transmisión y almacenamiento para aportar seguridad a la comunicación y en la base de datos.

En este capítulo, se describe la implementación en un microcontrolador de 32 bits, del algoritmo de encriptado caótico propuesto utilizado para este trabajo de tesis de maestría, para brindar confidencialidad a señales biomédicas en sistemas embebidos. Se verifica la seguridad y eficiencia mediante distintos análisis de seguridad como espacio de claves, sensibilidad a la clave y señal clara, histogramas, correlación, entropía de la información y recursos de implementación. Todos los análisis se realizan con el algoritmo de encriptado caótico propuesto utilizando el mapa Badola mediante el software de MatLab.

7.1. Introducción

Existen muchas áreas donde la seguridad en sistemas de comunicaciones es un problema, la milicia, industria, comercio electrónico, sistemas biométricos, telemedicina, datos personales, etc., son ejemplos en las cuales, se requiere mantener la integridad de la información mediante encriptado para evitar problemas de seguridad cuando la información se transmite a través de canales inseguros o en su almacenamiento en sistemas embebidos.

En este capítulo, se presenta una implementación en un microcontrolador de 32 bits del algoritmo de encriptado propuesto de este trabajo de tesis de maestría, con el propósito de proteger las señales biomédicas como ECG. En las siguientes secciones, se muestran detalles del encriptado, después la implementación y el análisis de seguridad para resistir distintos ataques.

7.2. Revisión de la literatura

La implementación en microcontroladores de algoritmos criptográficos basados en caos es reciente. En los últimos años, este tipo de trabajos se ha incrementado con el tiempo. Recientemente en [80], se implementaron osciladores caóticos en sistemas embebidos. Se presentan estudios utilizando como base sistemas caóticos de naturaleza discreta y sistemas caóticos en tres dimensiones en sus versiones continuas y discretas. Además, se realiza un estudio para reproducir los algoritmos numéricos en versión discreta utilizando microcontroladores PIC de 8 bits, dsPIC de 16 bits, PIC32 de 32 bits y un FPGA Altera. Se incluyen resultados analíticos, numéricos y experimentales en cada uno de estos estudios reportados.

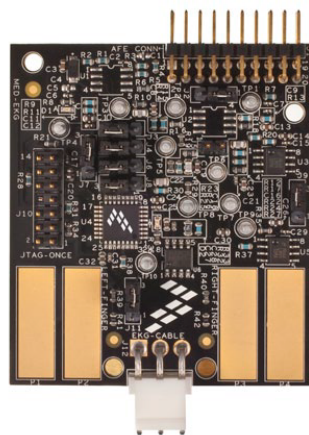
En [81, 82], se ha utilizado encriptado caótico para huellas dactilares, donde se muestra el proceso de encriptado y desencriptado, además se realiza un análisis de seguridad para validar el algoritmo propuesto. En [83] se ha utilizado encriptado caótico para texto, donde se encripta un párrafo de texto mediante un sistema embebido basado en un microcontrolador de 32 bits y se muestran distintos análisis de seguridad. En [84, 85], se ha utilizado encriptado caótico para imágenes, en donde en ambos se encripta la misma imagen a color pero solo uno muestra el procedimiento de encriptado y un extenso análisis de seguridad. En [86, 87], se ha utilizado encriptado caótico para señales biomédicas, en donde obtiene las señales de una base de datos, encriptan las señales a nivel software y no presentan un análisis de seguridad.

7.3. Resultados experimentales

El algoritmo de encriptado se implementa en el microcontrolador MCF51MM256 de 32 bits con procesador Coldfire V1 de Freescale, el cual, cuenta con una frecuencia de 48 Mhz. Para la programación, se utiliza el software CodeWarrior con la versión 6.3, el programa es almacenado en la memoria flash del microcontrolador. Para realizar el análisis de seguridad, se utiliza el software MatLab con la versión 8.5 en una computadora matebook con un procesador AMD Ryzen 2.10 Ghz, 8 Gb de memoria RAM y sistema operativo windows 10 de 64 bits; se utiliza representación punto flotante con precisión doble y una precisión de 10^{-15} . La clave secreta de 69 dígitos hexadecimales se introduce al microcontrolador mediante programación junto con el algoritmo de encriptado y la señal clara se obtiene mediante el módulo MED-ECG junto con los electrodos, la cual se someterá al proceso de encriptado para obtener el criptograma y posteriormente realizar el análisis de seguridad.

Se utiliza:

11223344556677889900AABBCCDDEEFF112233445566778811223344556677889900
A, como clave secreta y la señal de ECG se obtiene utilizando el módulo MED-ECG, el cual, se conecta al microcontrolador MFC51MM256, que a su vez el microcontrolador está conectado a la torre TWR-MCF51MM (Ver figura 7.1).



(a)



(b)



(c)

Figura 7.1: Sistema para obtener la señal ECG: a) Módulo MED-ECG, b) Microcontrolador MCF51MM256 y c) Torre TWR-MCF51MM.

7.3.1. Adquisición de la señal clara ECG

Para obtener la señal clara de ECG, se utilizan electrodos médicos de 2 in x 1 in, los cuales tienen un pegamento que se adhiere a la piel y se utilizan cables aislados para evitar interferencia de ruido (ver figura 7.2).

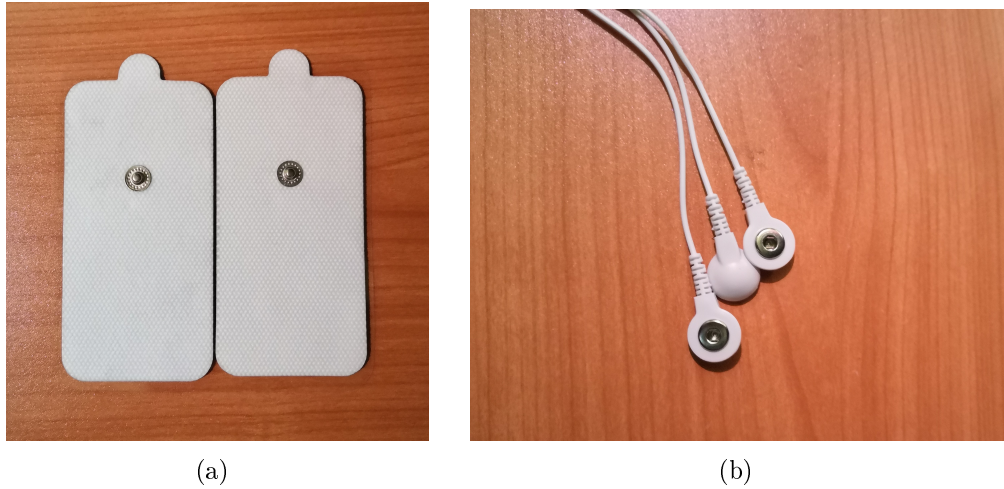


Figura 7.2: Material para obtención de la señal ECG: a) Electrodos médicos y b) Cables aislados.

La derivación (configuración de conexión) de los electrodos que se utiliza para visualizar la señal del ECG, se utiliza la configuración que se muestra en la figura 7.3 [88], la cual, corresponde a la conexión bipolar de Willen Einthoven (triangulo de Einthoven), registra la diferencia de potencial entre dos puntos en este caso de los antebrazos y el abdomen.

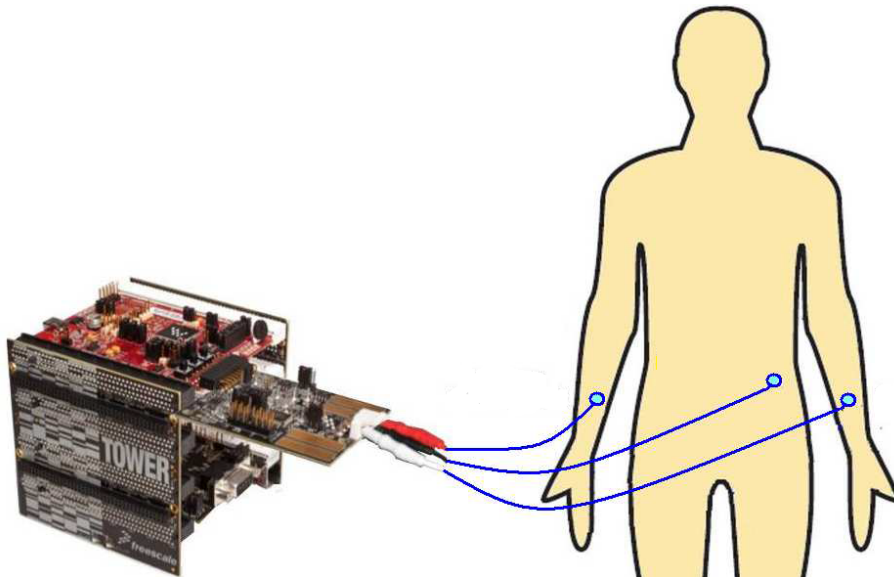


Figura 7.3: Ejemplo de conexión para obtener señal de ECG.

Para la visualización y adquisición del ECG, se utiliza el programa desarrollado en [88], que consiste de módulos analógico del microcontrolador MCF51MM256 junto con filtros de ruido para acondicionar la señal, después del ajuste de filtrado digital, la señal se envía vía I2C donde se puede visualizar la señal clara de ECG en la interfaz de laboratorio (ver figura 7.5).

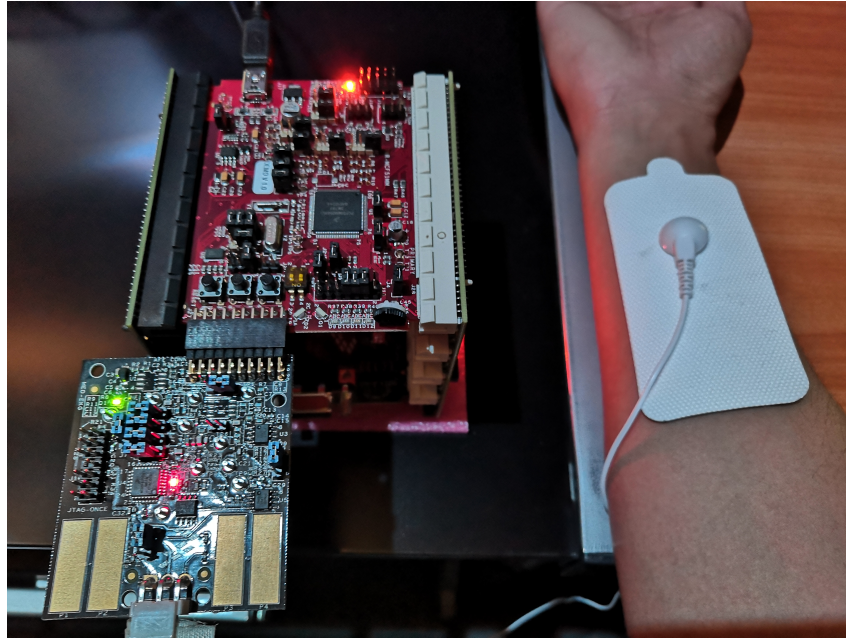


Figura 7.4: Conexión del sistema para obtención de señal ECG.

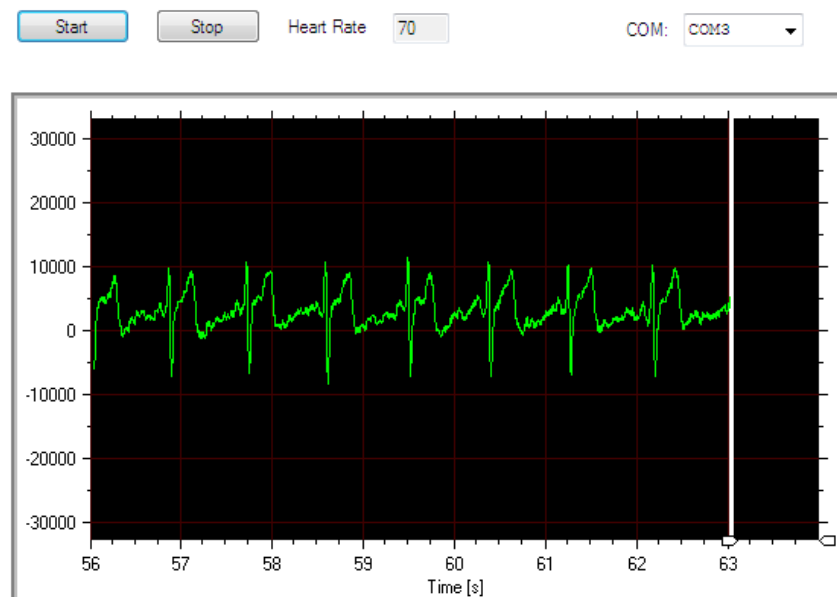


Figura 7.5: Interfaz de visualización para la señal ECG en computadora personal.

En la figura 7.6, se muestra una señal clara de ECG, normalizada de 0 a 1 y adquirida en un tiempo de 10 segundos a una tasa de muestreo de 100 Hz, la cual, es reproducida en MatLab.

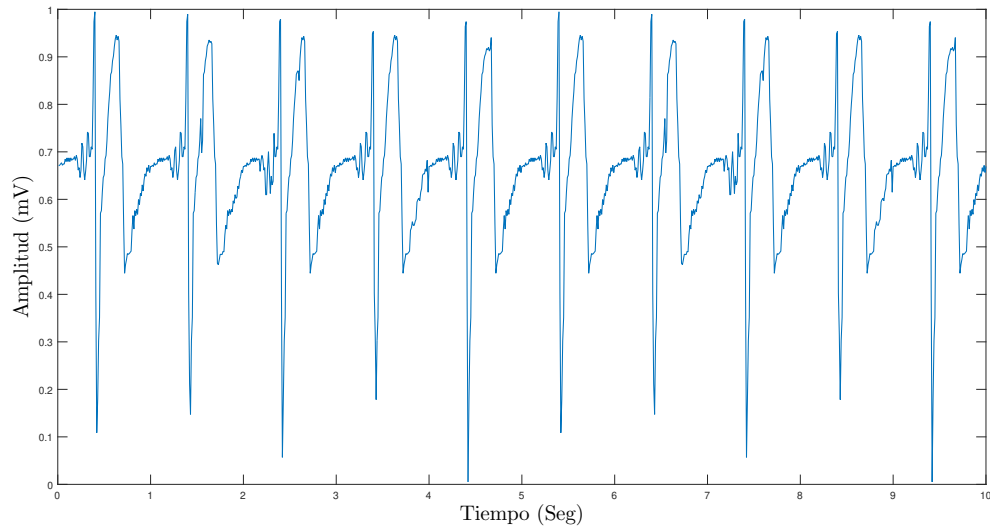


Figura 7.6: Señal clara de electrocardiograma utilizada en el proceso de encriptado.

En la figura 7.7, se muestra el criptograma correspondiente con el uso de la clave secreta antes mencionada.

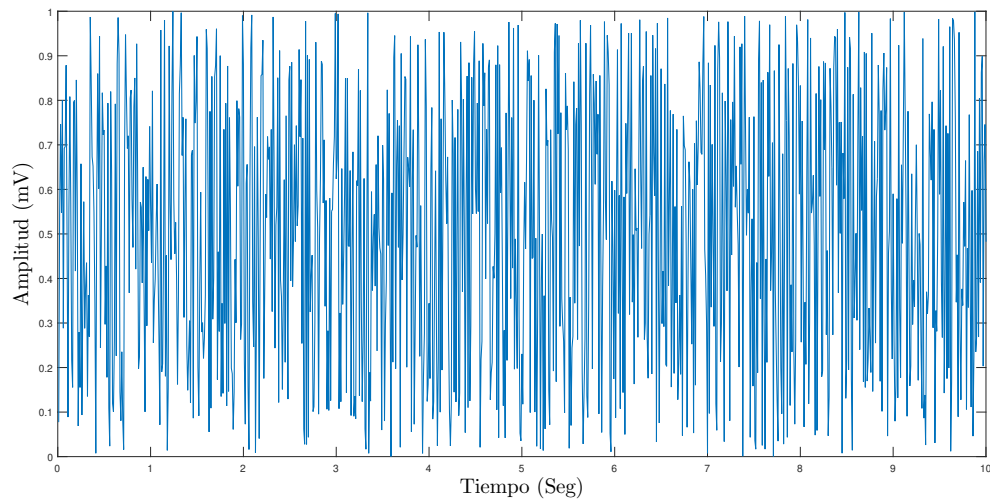


Figura 7.7: Señal encriptada de electrocardiograma.

El proceso del encriptado caótico propuesto en el capítulo 6, se realiza de la siguiente manera:

1. Definir la clave secreta.

Se utiliza una clave secreta de 276 bits definida por 69 caracteres hexadecimales, la clave se divide en doce secciones para la inicialización de secuencias caóticas de dos mapas Badola.

2. Transformación.

El sistema criptográfico (digital) obtiene la señal clara y determina máximos-mínimos para transformar las amplitudes de la misma en amplitudes con valores entre $(0, 1)$. Para obtener una nueva señal clara transformada, se determina el valor de A con $P_{min} = \min(A)$, donde $P_{min} \in R$ es un escalar que representa el valor mínimo de A y $\min$ representa la función para determinar el valor mínimo de A . También se encuentra el valor máximo en A con $P_{max} = \max(A)$, donde $P_{max} \in R$ es un escalar que representa el valor máximo en A y $\max$ representa la función para determinar el valor máximo de A . Después se determina la siguiente transformación:

$$AT = \frac{A - (P_{min} - 0.01)}{P_{max} + 0.01} \quad (7.1)$$

donde AT representa la señal clara transformada en valores entre $(0, 1)$.

3. Valor Z .

Todos los valores de la señal clara AT se suman con datos caóticos del mapa Badola 1 iterado $I = 1000$, con μ_2 , x_2 y y_2 . Los elementos de AT y datos caóticos se suman como sigue:

$$Z = Z + (AT_i + x_{I+1-i}^{B2}) \pmod{1}, \quad (7.2)$$

donde $i = 1, 2, 3, \dots, \ell$, $Z \in (0, 1)$ es una variable inicializada en cero para incrementar la seguridad. AT_i son los elementos de la señal clara transformada y x^{B2} es la secuencia caótica del mapa Badola.

4. Encriptado.

El mapa Badola 1 es iterado $T = 1000$ con μ_1 , x_1 y y_1 . Se determina una secuencia para permutación.

$$P_i = \text{round}(x_{T+i}^{B1} * (\ell - 1)) + 1, \quad (7.3)$$

donde $i = 1, 2, 3, \dots, \ell$, $P \in [1, \ell]$ es el vector pseudoaleatorio de permutación y round es la operación de redondeo al valor más cercano. Sin embargo, P tiene valores repetidos que son determinados por software y remplazados por los faltantes de forma automática para incrementar la seguridad. El vector para el proceso de difusión se determina de la siguiente forma:

$$D_i = (x_{T+i}^{B1} * 1000) + Z \pmod{1}, \quad (7.4)$$

donde $i = 1, 2, 3, \dots, \ell$, $D_i \in (0, 1)$. La multiplicación por 1000 se utiliza para obtener una distribución de datos del Badola más uniforme.

Finalmente, el proceso de encriptado se realiza con la siguiente operación:

$$C_i = (AT(P_i) + D_i) \pmod{1}, \quad (7.5)$$

donde $i = 1, 2, 3, \dots, \ell$, $C \in (0, 1)$ es el criptograma.

5. Agregar datos al criptograma.

El valor de P_{min} , P_{max} y Z son agregados al final del criptograma de la siguiente forma:

$$\begin{aligned} C_{\ell+1} &= Z, \\ C_{\ell+2} &= 1 \quad \text{si } P_{min} < 0 \quad \text{o} \quad C_{\ell+2} = 0 \quad \text{si } P_{min} \geq 0, \\ C_{\ell+3} &= \text{abs}(P_{min})/1000, \\ C_{\ell+4} &= 1 \quad \text{si } P_{max} < 0 \quad \text{o} \quad C_{\ell+4} = 0 \quad \text{si } P_{min} \geq 0, \\ C_{\ell+5} &= \text{abs}(P_{max})/1000, \end{aligned}$$

donde abs representa el valor absoluto.

6. Desencriptado.

El proceso de desencriptado consiste en invertir los pasos de encriptado. Para ello, se debe obtener del criptograma los valores de Z , P_{min} , P_{max} . Con el uso de la misma clave secreta, se calculan los vectores pseudoaleatorios de P y D de la misma forma que en el paso 4. El proceso de desencriptado se determina como sigue

$$D(P_i) = (C_i - D_i) \pmod{1}, \quad (7.7)$$

donde $i = 1, 2, 3, \dots, \ell$ y $D \in (0, 1)$ es la señal clara transformada desencriptada. Para obtener la señal clara con amplitudes originales, se escala D con el uso de P_{min} y P_{max} .

7.4. Sistema embebido

Para realizar los procesos de encriptado y desencriptado, se utiliza el microcontrolador MCF51MM256 donde por medio de programación en el software CodeWarrior se realizan los procesos, para visualizar información e indicaciones del programa se utiliza una pantalla LCD 20 x 4, matrix orbital modelo LK204-25, donde se muestran distintos mensajes, para activar los LED's y la alarma se utilizan los puertos GPIO del microcontrolador para activar el mando de control, el cual cuenta con 3 LED's, un buzzer y dos switches que interactúan con el programa y los datos obtenidos son almacenados en la memoria flash del microcontrolador.

En la figura 7.8, se muestra el diagrama a bloques del funcionamiento del sistema embebido propuesto para la parte experimental de este trabajo, donde se indica la forma en que se conectan y comunican los módulos.

El sistema embebido está conformado por 6 módulos independientes, cada uno con funciones específicas y controlados directo o indirectamente por el microcontrolador MCF51MM256.

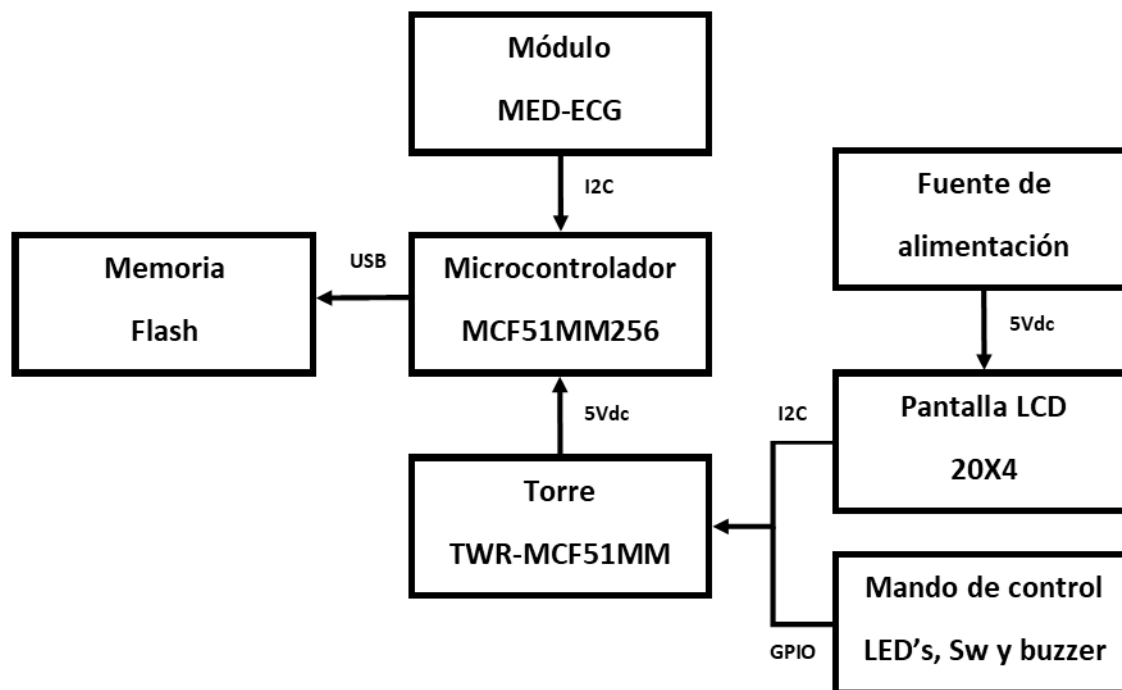


Figura 7.8: Diagrama a bloques del sistema embebido.

- **Microcontrolador.** El programa del sistema se ejecuta en este módulo, al iniciar, se establece la comunicación I2C con la LCD y el módulo MED-ECG, configura los puertos de entrada y salida. Despliega información en la LCD y reconoce los comandos que solicita el usuario por el mando de control, ya se para encriptar información o desencriptar.
- **Módulo MED-ECG.** Se encarga de obtener la señal de electrocardiograma del usuario.
- **Torre TWR-MCF51MM.** Se encarga de proporcionar voltaje al microcontrolador.
- **Fuente de alimentación.** Se encarga de proporcionar el voltaje para la LCD.
- **Pantalla LCD 20x4.** Este módulo recibe órdenes del microcontrolador por comunicación I2C, en la pantalla se despliegan los mensajes para dar información, indicaciones y de la actividad que está realizando el programa principal (ver figura 7.9).

- **Mando de control.** En este módulo se tienen dos switches, con los cuales se selecciona si se quiere encriptar (switch 2) o desencriptar (switch 1). También tiene 3 LED's como indicadores (ver figura 7.10). Además, cuenta con un buzzer el cual suena cuando se presiona un switch, cuando inicia y finaliza la operación que se está realizando.

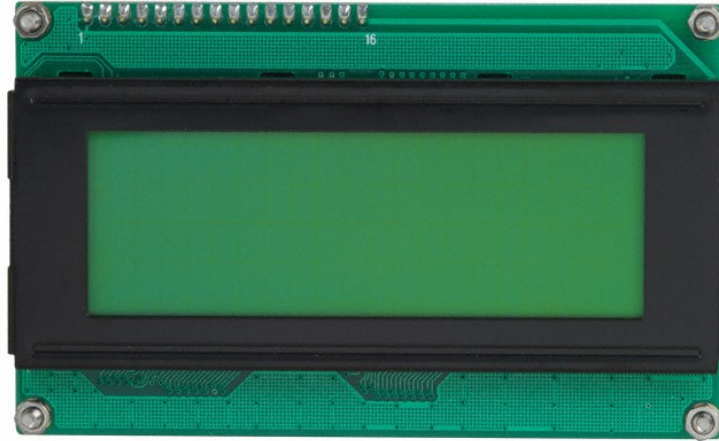


Figura 7.9: Pantalla LCD 20x4.

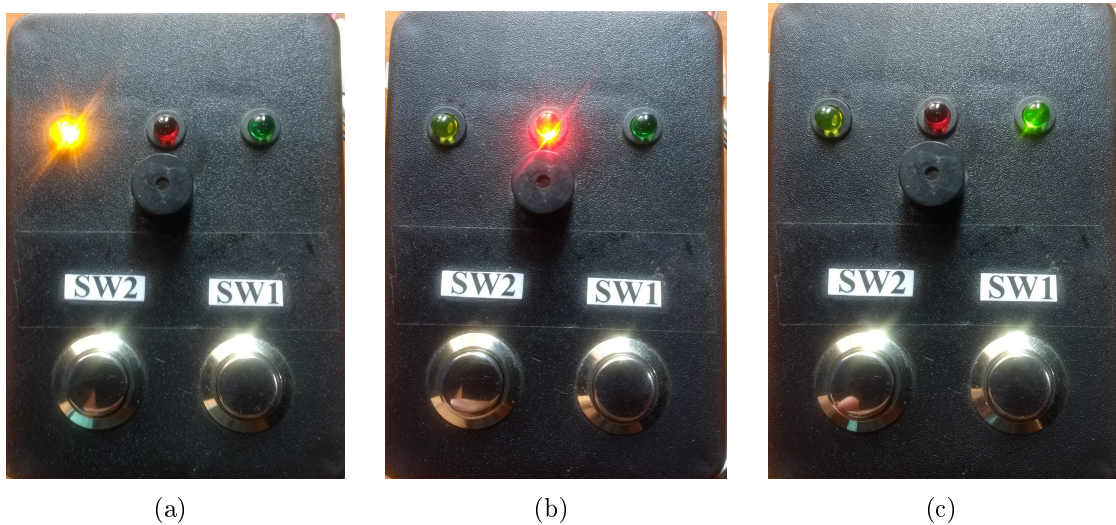


Figura 7.10: Función de los LED's del mando de control: a) LED Amarillo indica que el proceso se está ejecutando, b) LED Rojo indica inicio del proceso y c) LED Verde indica fin del proceso.

En la figura 7.11, se tiene una fotografía del sistema embebido utilizado y se ubican los módulos del sistema.

A continuación, se muestra los mensajes del sistema embebido para obtener la señal ECG, encriptar, desencriptar y finalizar el sistema.

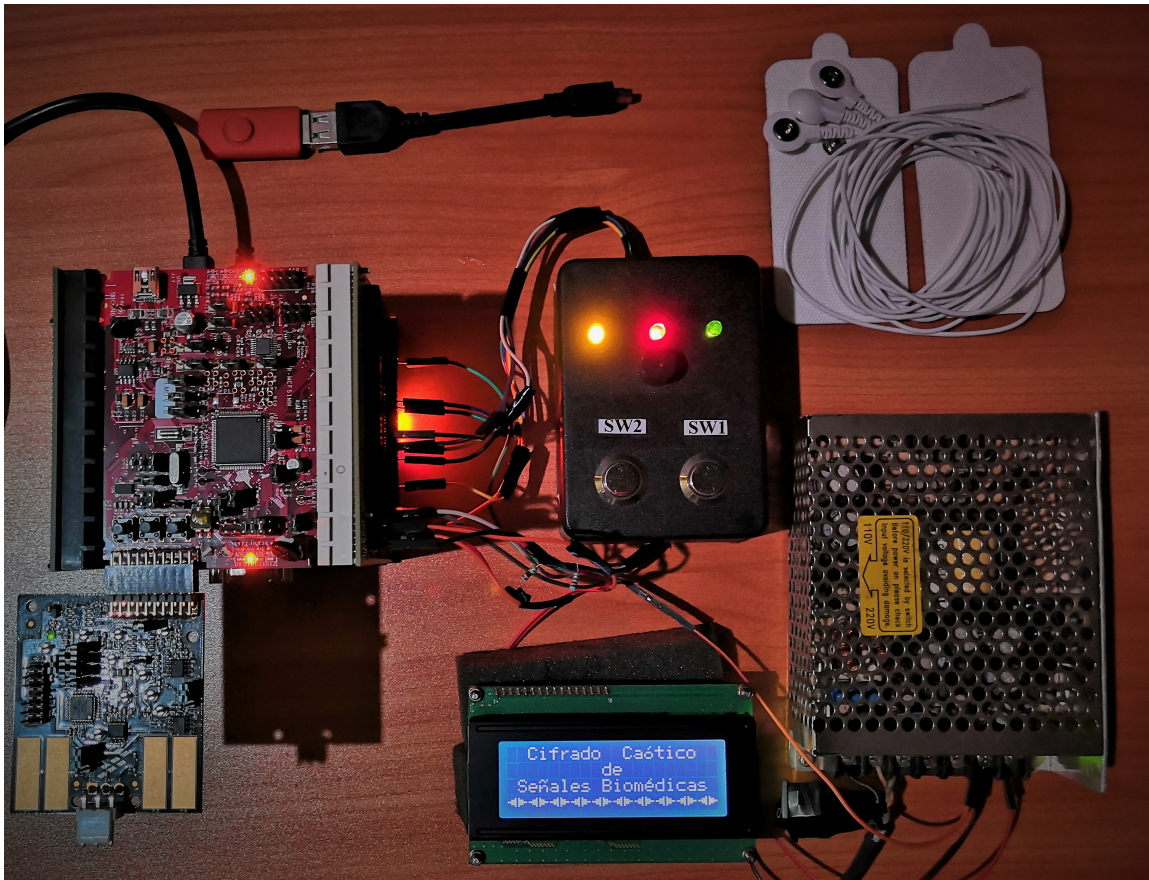


Figura 7.11: Sistema embebido utilizado.

Inicio de sistema:

Al encender la LCD se muestra un mensaje sobre el sistema, al iniciar el código se muestra un mensaje de inicio, después, aparece otro mensaje que indica que el sistema está iniciando (ver figura 7.12), cuando aparece este mensaje en el mando de control los tres LED's parpadean tres veces y el buzzer se activa indicando que el sistema está listo para funcionar.

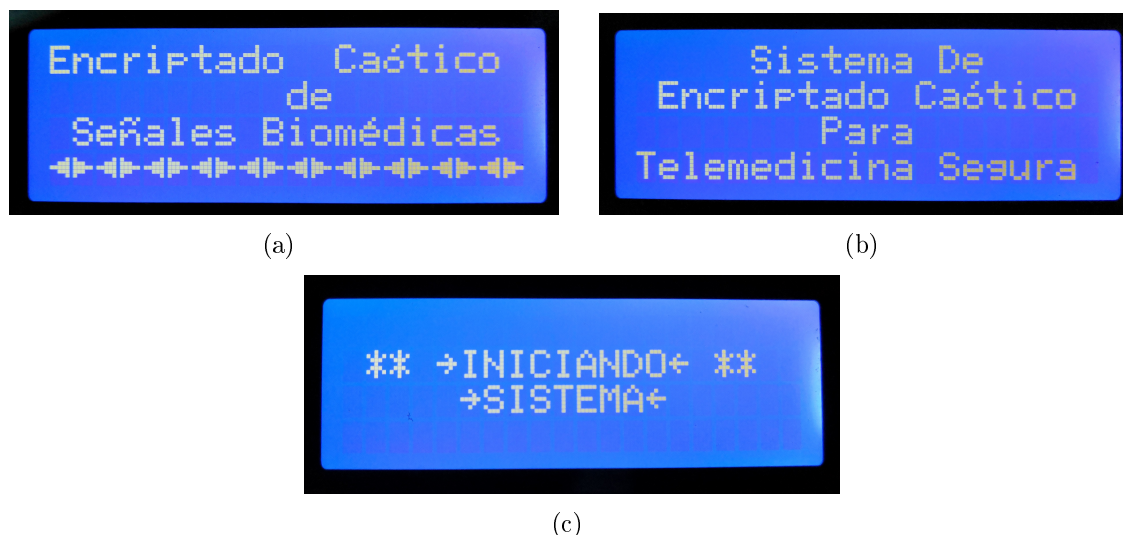


Figura 7.12: Mensajes de inicio del sistema: a) Mensaje al encender la LCD, b) Mensaje al iniciar el programa y c) Mensaje de inicio de sistema.

Adquisición de señal de electrocardiograma:

Después de iniciar el programa, aparece un mensaje el cual indica que se está obteniendo la señal de ECG (para esto ya debe estar el sistema conectado de electrodos), posteriormente un mensaje que indica que se está almacenando la señal obtenida de 10 segundos y para finalizar un mensaje que dice que la señal se ha almacenado correctamente (ver figura 7.13).

Encriptado:

Para el proceso de encriptado aparece un mensaje, el cual indica que se debe presionar el switch 2 para realizar el proceso de encriptado. Cuando se presiona el switch 2, inicia el proceso de encriptado y la pantalla de la LCD cambia, diciendo que se está encriptando y almacenando la información, mientras el LED rojo enciende una vez junto con el buzzer y el LED amarillo parpadea cinco veces indicando que se está realizando el proceso. Cuando este proceso finaliza, se muestra en la LCD que la información se encriptó y se almacenó correctamente, el buzzer suena y el LED verde parpadea una vez indicando que el proceso de encriptado ha finalizado con éxito (ver figura 7.14). Los datos (criptograma y clave secreta) se guardan en un archivo .txt (ver figura 7.17), el cual se extrae de la memoria flash donde se almacenó.

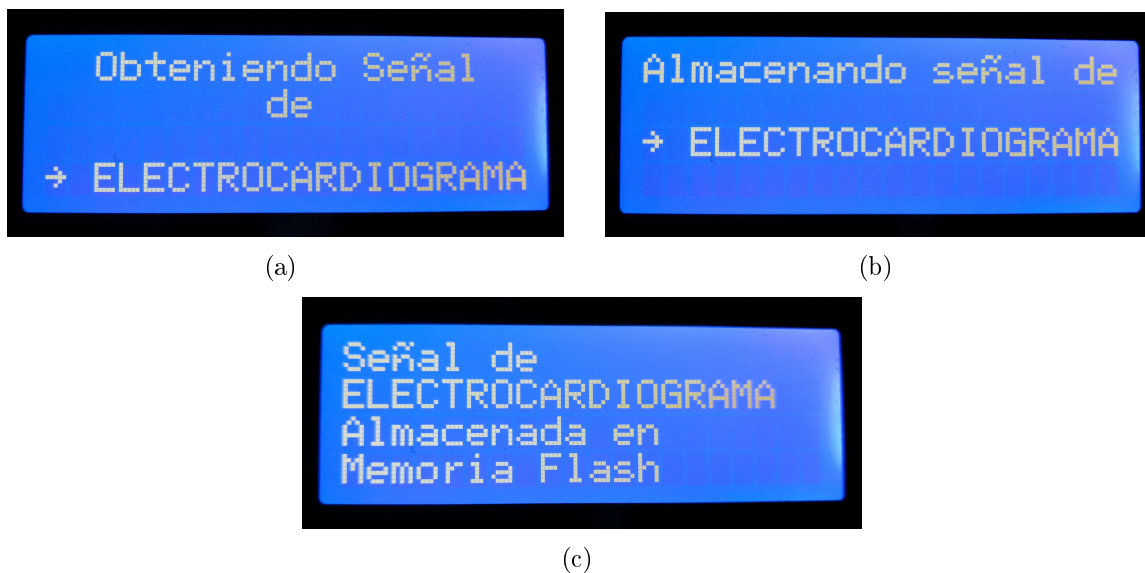


Figura 7.13: Mensajes para la obtención de la señal de ECG: a) Mensaje: Obteniendo señal, b) Mensaje: Almacenando señal y c) Mensaje: Almacenado correcto.

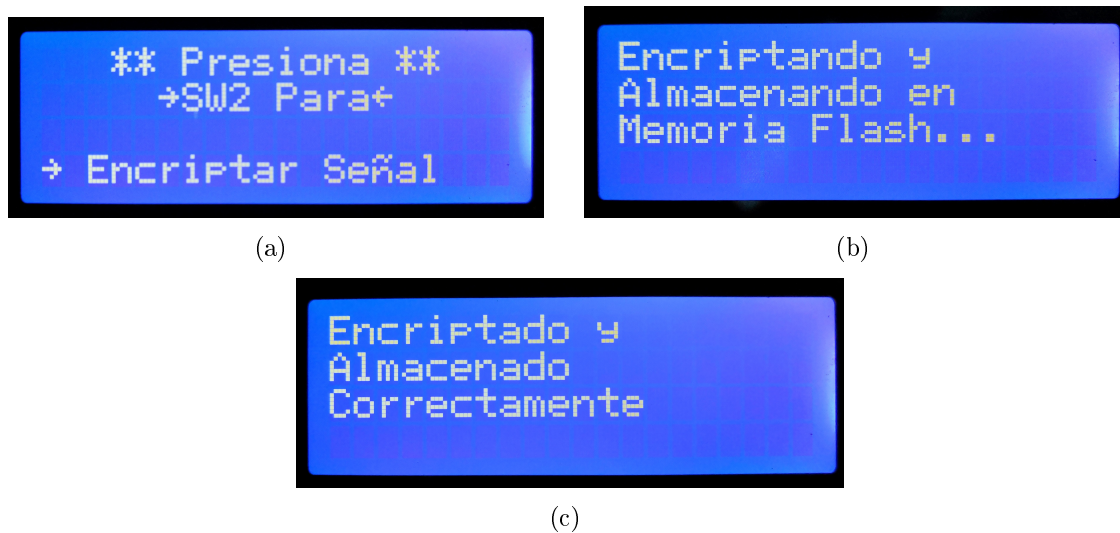


Figura 7.14: Mensajes para el proceso de encriptado: a) Mensaje: Indicación para encriptar, b) Mensaje: Inicio de encriptado y c) Mensaje: Finalización de encriptado.

Desencriptado:

Para el proceso de desencriptado aparece un mensaje, el cual indica que se debe presionar el switch 1 para realizar el proceso de desencriptado. Cuando se presiona el switch 1, la pantalla cambia y dice que se está extrayendo la información (criptograma) de la memoria flash para realizar el desencriptando, el LED rojo enciende una vez junto con el buzzer y el LED amarillo parpadea cinco veces indicando que se está realizando el proceso. Cuando finaliza, se muestra en el LCD que la información se desencriptó y se almaceno correctamente, el buzzer suena y el LED verde parpadea indicando que el proceso de desencriptado ha finalizado (ver figura 7.15). Los datos (señal clara y clave secreta) se guardan en un archivo .txt (ver figura 7.17).

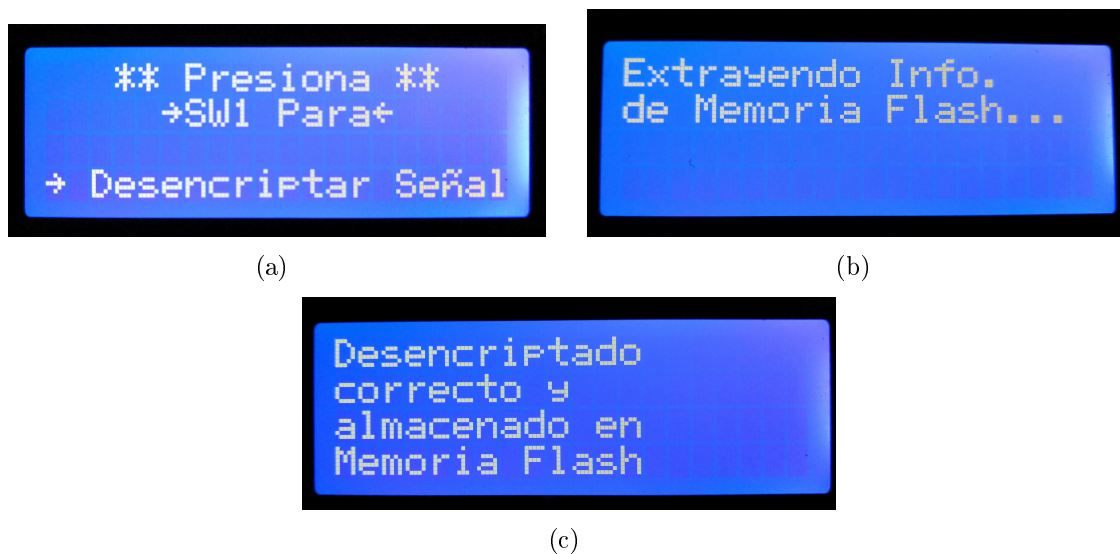


Figura 7.15: Mensajes para el proceso de desencriptado: a) Mensaje: Indicación para desencriptar, b) Mensaje: Extrayendo información y c) Mensaje: Finalización de desencriptado.

Fin de sistema:

Al terminar el proceso aparece un mensaje indicando que el sistema ha finalizado de forma correcta.

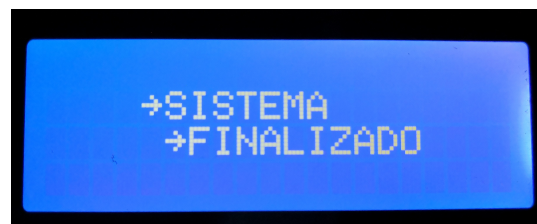


Figura 7.16: Mensaje: Finalización del sistema.

Clave Secreta: 11223344556677889900AABBCCDDEEFF 112233445566778811223344556677889900A Texto Encriptado: .0774365 .5440912 .74660596606895 .7459623 .4950055 Texto Claro: .3626745 .3650129 .36988783643679 .3353398 .3605678	Clave Secreta: 11223344556677889900AABBCCDDEEFF1 12233445566778811223344556677889900A Texto Claro: .3626745 .3650129 .36988783643679 .3353398 .3605678 Texto Encriptado: .0774365 .5440912 .74660596606895 .7459623 .4950055
(a)	(b)

Figura 7.17: Archivos *.txt obtenidos de los procesos de encriptado y desencriptado correspondiente a un ECG de 1000 datos: a) Datos obtenidos encriptados, b) Datos obtenidos desencriptados.

Al utilizar sistemas embebidos se debe pensar en la seguridad de la información contenida en el dispositivo y la transmitida por redes como internet. El diseño de un producto que incorpora sistemas embebidos generalmente está orientado a minimizar los costos y maximizar la confiabilidad, por lo que se deben incluir funciones criptográficas, diseño de protocolos y consultoría en análisis y verificación, así como servicios de pruebas de seguridad y evaluaciones específicas.

Como sucede con la mayoría del equipamiento industrial, los sistemas embebidos están pensados para ser seguros a nivel físico, incorporando medidas de reinicio en caso de fallo (watchdog), partes de hardware duplicadas, programación inmune a fallos, etc. Pero es necesario reforzar la seguridad de la información ya que el número de dispositivos embebidos está creciendo exponencialmente, acompañado de un incremento desproporcionado de los riesgos a los que se exponen, especialmente con la integración de capacidades de red.

Las razones para aplicar medidas de seguridad a los sistemas embebidos son muchas, por parte del usuario se pueden hacer cosas para mejorar la seguridad. Por ello, en este trabajo, se realizan distintos análisis de seguridad para validar el algoritmo criptográfico utilizado y además que sea resistente ante distintos ataques criptoanalíticos para proporcionar mayor seguridad en la información.

7.5. Análisis de seguridad

Para los análisis de seguridad [89], se extrae del microcontrolador la señal clara y su respectiva señal encriptada, para realizar los distintos análisis de seguridad en MatLab. En las siguientes secciones, se presentan los diferentes análisis de seguridad a los que se somete el criptograma con el algoritmo propuesto implementado en microcontrolador para un sistema embebido.

7.5.1. Espacio de claves

Todo sistema criptográfico es susceptible a un ataque exhaustivo (consiste en probar cada una de las posibles claves secretas hasta encontrar la señal clara), donde cada posible clave secreta se utiliza para descryptar un criptograma. Si el espacio de claves es pequeño, es decir, menor a 2^{56} posibilidades, el sistema criptográfico no es seguro ante un ataque exhaustivo o de fuerza bruta. Para proporcionar seguridad suficiente contra un ataque exhaustivo, el espacio de claves debe ser mayor a 2^{100} . Además, cada clave secreta se debe considerar fuerte, es decir, que genere secuencias caóticas y no periódicas. La clave secreta propuesta consiste de 69 dígitos hexadecimales (276 bits) y todas se consideran fuertes, por tanto, el algoritmo propuesto en esta tesis utiliza un espacio de claves de 2^{276} y puede resistir un ataque exhaustivo.

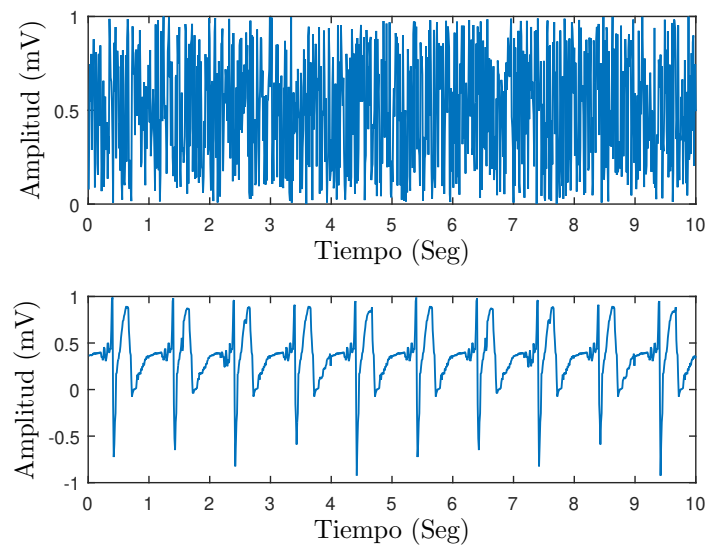
7.5.2. Sensibilidad a la clave secreta

Un algoritmo de encriptado debe ser altamente sensible a la clave secreta, incluso a nivel de bit. Esta propiedad aplica para el proceso de encriptado y descryptado; en el proceso de encriptado, si la misma señal clara es encriptada dos veces con dos claves secretas similares, la señal encriptado debe ser muy diferente entre ellos, mientras que, en el proceso de descryptado, únicamente la clave secreta correcta puede recuperar la señal original. En esta sección, la sensibilidad de la clave secreta en el proceso de encriptado y descryptado, se prueba y verifica con 2 claves secretas similares (ver tabla 7.1).

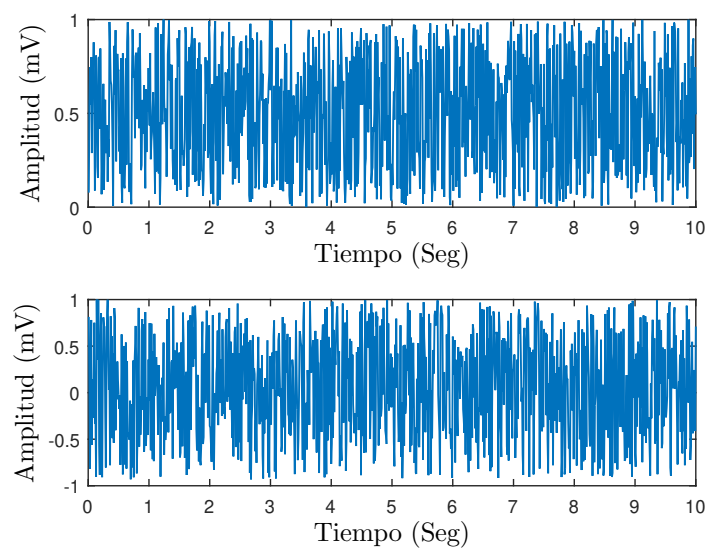
Para este análisis se encripta la señal de electrocardiograma utilizando las claves de la tabla 7.1. En la figura 7.18 (a), se observa que al encriptar con la Clave 1 y descryptar con la misma clave se recupera la señal clara mientras que en la figura 7.18 (b), al encriptar con la clave 1 y descryptar con la clave 2 no se recupera la señal clara.

Tabla 7.1: Claves secretas utilizadas para el análisis de sensibilidad a la clave secreta.

No. clave	Clave secreta
Clave 1	11223344556677889900AABBCCDDEEFF112233445566778811223344556677889900A
Clave 2	11223344556677189900AABBCCDDEEFF112233445566778811223344556677889901A



(a)



(b)

Figura 7.18: Señal de ECG encriptada con la clave 1: a) Señal de ECG recuperada y b) Señal de ECG no recuperada.

7.5.3. Sensibilidad a la señal clara

Un buen sistema criptográfico debe ser sensible con respecto a la señal clara, es decir, pequeños cambios (un bit) en la señal clara genera un gran cambio en la señal encriptada; si el algoritmo tiene esta propiedad, el encriptado puede resistir un ataque diferencial, el cual, básicamente es un ataque de texto claro conocido.

Dos medidas son utilizadas para determinar la sensibilidad a la señal clara: *NPCR* (del inglés, *Net Pixel Change Rate*), tasa de cambio de pixel neto y *UACI* (del inglés, *Unified Average Changing Intensity*), promedio unificado de cambio de intensidad.

Para resistir un ataque diferencial, se requiere que el algoritmo criptográfico presente sensibilidad a pequeños cambios en la señal clara y se utilizan dos conceptos para determinarlo: NPCR y UACI.

NPCR se calcula con la siguiente expresión:

$$NPCR = \frac{\sum_{i=1}^{\ell} W(i)}{\ell} \times 100 \quad (7.8)$$

con

$$W(i) = \begin{cases} 0 & \text{if } E_1(i) = E_2(i) \\ 1 & \text{if } E_1(i) \neq E_2(i) \end{cases} \quad (7.9)$$

y el valor de UACI se determina con

$$UACI = \frac{100}{\ell \times 95} \sum_{i=1}^{\ell} |E_1(i) - E_2(i)| \quad (7.10)$$

donde ℓ es la longitud de los datos, E_1 and E_2 son los dos criptogramas. El proceso para determinar los valores es como sigue: primero, la señal clara se encripta con la CLAVE 1 para generar E_1 ; después, el bit número 100 de la señal clara se cambia de **100** a **101**, y el proceso de encriptado se repite con la misma CLAVE 1 para generar E_2 .

La tabla 7.2, muestra los resultados de NPCR y UACI con E_1 and E_2 . Por tanto, el esquema propuesto es robusto ante ataques diferenciales, ya que el 99.8 % de los símbolos son diferentes con una diferencia de magnitud en promedio del 33.2 %.

Tabla 7.2: Resultados de los análisis NPCR y UACI.

Prueba	Resultado
NPCR(%)	99.805638267497536
UACI(%)	33.213315685367321

7.5.4. Histogramas

El histograma muestra la distribución de información mediante una gráfica y representa información estadística. Un sistema criptográfico es susceptible a un ataque de histogramas y puede resistir si la señal encriptada tiene un histograma uniforme (información impredecible).

El histograma de la señal encriptada debe ser uniforme para resistir un ataque estadístico en el lenguaje que fue encriptado originalmente. El histograma de la señal clara del electrocardiograma se muestra en la figura 7.19, se puede apreciar la característica estadística de la fuente; sin embargo, el histograma de la señal encriptada es más uniforme, como se aprecia en la figura 7.20. Por tanto, el algoritmo criptográfico propuesto es robusto ante un ataque de histograma, ya que no se puede identificar a qué tipo de información corresponde.

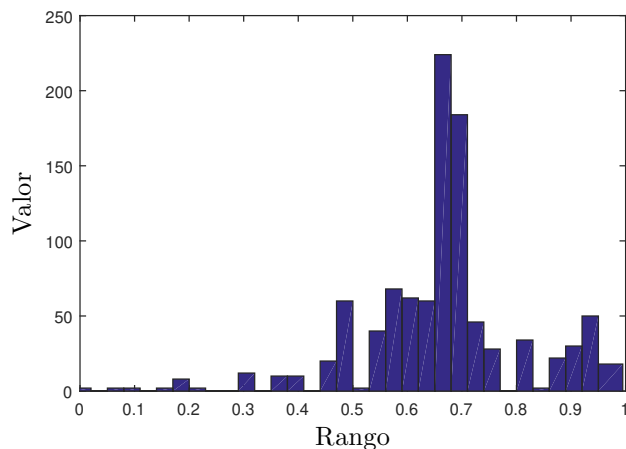


Figura 7.19: Histograma de la señal clara.

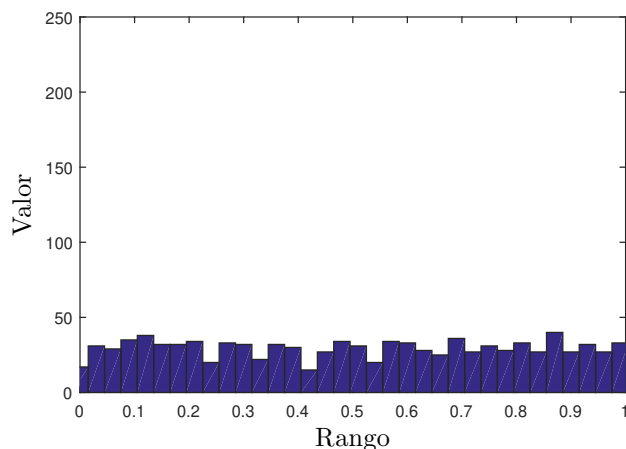


Figura 7.20: Histograma de la señal encriptada.

7.5.5. Correlación

La correlación se puede medir entre -1 y 1, donde 0 significa correlación nula. Un criptoanalista puede utilizar esta información en un ataque estadístico para encontrar la clave secreta y recuperar la señal clara. Por tanto, la señal encriptada debe tener correlación nula.

La fórmula para calcular la correlación es la siguiente

$$Cr = \frac{N \times \sum_{i=0}^N (x_i \times y_i) - \sum_{i=0}^N x_i \times \sum_{i=0}^N y_i}{\sqrt{\left(N \times \sum_{i=0}^N (x_i)^2 - \left(\sum_{i=0}^N x_i\right)^2\right) \times \left(N \times \sum_{i=0}^N (y_i)^2 - \left(\sum_{i=0}^N y_i\right)^2\right)}} \quad (7.11)$$

El valor de correlación es $Cr \in (-1, 1)$ donde 0 significa nula correlación y 1 significa alta correlación. El resultado es de 0.003601945578124, por lo que se puede decir que tiene correlación nula.

7.5.6. Entropía de la información

La entropía determina que tan impredecible es un mensaje, es decir, mide cuanto desorden genera el algoritmo de encriptado. Si el proceso de encriptado es bueno, este genera alto desorden en la señal encriptada; por tanto, mayor será la entropía. Caso contrario, si el proceso de encriptado no es suficientemente aleatorio, el algoritmo criptográfico puede estar sujeto a un exitoso ataque de entropía, porque el criptograma es predecible.

En esta sección, el desempeño del encriptado propuesto en la etapa de difusión es probado y verificado. La entropía $H(m)$ de un mensaje m puede calcularse como sigue

$$H(m) = \sum_{i=0}^{2^N-1} p(m_i) \log_2(1/p(m_i)), \quad (7.12)$$

donde N es el número de bits que representan la unidad básica del mensaje m , 2^N son todas las combinaciones de la unidad básica, $p(m_i)$ representa una probabilidad de m_i , $\log_2$ es el logaritmo base 2 y la entropía esta expresada en bits, donde la máxima entropía es N . Si un mensaje m es encriptado con 2^N posibles valores, la entropía debería ser idealmente $H(m) = N$, si m es puramente aleatorio.

En este análisis, la señal encriptada entre 0 y 1 se transforma a datos de 8 bits, es decir de 0-255. De manera que, la máxima entropía es de 8. El resultado es de 7.9595. La entropía de la señal encriptada es cercana a 8, por tanto, el proceso de difusión genera alto desorden para resistir un ataque de entropía.

7.5.7. Recursos de implementación y desempeño

En cualquier implementación es importante determinar los recursos de implementación empleados, como memoria requerida del programa, arquitectura del microcontrolador, módulos de comunicación, etc., que están relacionados con el costo de la aplicación. La tabla 7.3, muestra los recursos utilizados en la implementación en microcontrolador del algoritmo de encriptado propuesto en la tesis.

Un buen algoritmo de encriptado debe ser robusto ante ataques, pero además requiere ser rápido para aplicaciones de tiempo real en telemedicina. El tiempo de desencriptado es similar al encriptado. La velocidad de encriptado para una señal clara de 10 segundos y frecuencia de muestreo de 100 Hz es de 0.026241 segundos y para el desencriptado es de 0.029697 segundos.

Tabla 7.3: Recursos de implementación.

Característica	Usada/Total
Memoria Flash (KB)	(80 %)
Puertos comunicación	USB 3.0
Frecuencia (MHz)	(96 %)
Voltaje (Vdc)	5V
Costo sistema embebido (USD)	270
Dimensiones TWR-MCF51MM (in)	3.54×3.54

7.6. Conclusiones

En este capítulo, se implementó el algoritmo criptográfico basado en caos en un sistema embebido basado en un microcontrolador de 32 bits con el propósito de encriptar señales biomédicas para proporcionar confidencialidad en telemedicina. Un análisis de seguridad muestra la efectividad de la implementación y la seguridad que brinda el algoritmo criptográfico por lo que puede utilizarse para proteger la integridad de proteger señales biomédicas en telemedicina. Los resultados muestran que el algoritmo implementado en el sistema embebido presenta alta seguridad criptográfica y es eficiente para aplicaciones en telemedicina.

Capítulo 8

Conclusiones

8.1. Conclusiones generales

En este trabajo de tesis de maestría, se implementó en un sistema embebido de bajo costo, un algoritmo criptográfico basado en caos para la confidencialidad de señales biomédicas en aplicaciones de telemedicina. Se realizó un estudio de nueve mapas caóticos: Logístico, Chebyshev, Seno Logístico 2D, Hénon, Seno-Logístico 2D, Badola, Ushio y Lotka-Volterra, de los cuales, se obtuvieron el exponente de Lyapunov, tiempo de procesamiento y el espacio de clave, con base a sus características de desempeño se eligió el mapa Badola para el encriptado caótico, ya que presenta dinámicas hypercaóticas con solo 2 dimensiones y requiere pocas operaciones aritméticas.

El algoritmo criptográfico se basó en clave simétrica y arquitectura de confusión y difusión, donde se emplea la misma clave secreta para el proceso de encriptado y desencriptado de información. El sistema embebido utilizó una clave secreta de 69 números hexadecimales (276 bits), la cual es introducida mediante programación junto con el algoritmo criptográfico en el software CodeWarrior, donde se obtiene la señal clara de electrocardiograma utilizando un módulo médico el cual puede trabajar con el microcontrolador, los datos de la señal clara son encriptados con la clave.

Los resultados demostraron que el algoritmo criptográfico es seguro para la implementación en sistemas embebidos y para el encriptado de información en telemedicina. El sistema embebido fue de bajo costo y el análisis de seguridad demostró ser confiable para el manejo de información confidencial en telemedicina particularmente en telemetría.

8.2. Trabajo a futuro

Como trabajo a futuro se plantean las siguientes actividades:

- **Transmitir criptogramas a través de internet:** Realizar programación en el microcontrolador para emplear el conector ethernet y transmitir criptogramas de forma remota.

- **Analizar distintas señales biomédicas:** En este caso se trabajó con la señal de electrocardiograma. Sin embargo, el sistema puede aplicar distintas señales biomédicas como el pulsioxímetro (SpO2) y analizar los resultados de seguridad y eficiencia con el algoritmo de encriptado caótico con el mapa seleccionado.
- **Construir prototipo autónomo:** Agregar pantallas para desplegar señales médicas, banco de baterías recargables, sistemas autónomo, comunicación WiFi, etc.
- **Realizar análisis de seguridad a nivel físico del sistema embebido presentado:** Como análisis de la información de tiempo de cálculos, el monitoreo de consumo de energía o remanencia de datos, que puede proporcionar una fuente adicional de información que puede ser explotada para corromper el sistema criptográfico.

Bibliografía

- [1] Craig J. y Patterson V. (2005). Introduction to the practice of telemedicine. *Journal of Telemedicine and Telecare*, **11**(1): 3-9.
- [2] Meingast M., Roosta T. y Sastry S. (2006). Security and Privacy Issues with Health Care Information Technology. *Proceedings of the 28th IEEE Engineering in Medicine and Biology Society, New York, USA*, 5453-5458.
- [3] Komiya R. (2005). A Proposal for Tele-medicine Reference Model for Future Standardization. *Proceedings of the 7th International Workshop on Enterprise networking and Computing in Healthcare Industry, Busan, South Korea*, 224-228.
- [4] Wallauer J., Wangenheim A., Andrade R. y Macedo D. (2008). A Telemedicine Network Using Secure Techniques and Intelligent User Access Control. *Proceedings of the 21st IEEE International Symposium on Computer-Based Medical Systems, Jyväskylä, Finland*, 105-107.
- [5] Martínez S., Mateu V., Tomas R. y Valls M. (2012). Criptografía ordenable para bases de datos. *XII Reunión Española sobre Criptología y Seguridad de la Información, San Sebastián, España*, 35-40.
- [6] Murillo-Escobar M. A., Abundiz-Pérez F., Cruz-Hernández C. y López-Gutiérrez R. M. (2014). A novel symmetric text encryption algorithm based on logistic map. *Proceedings of the International Conference on Communications, Signal Processing and Computers, Interlaken, Suiza*, 49-53.
- [7] Andreatos A. S. y Leros A. P. (2013). Secure Image Encryption Based On a Chua Chaotic Noise Generator. *Engineering Science and Technology Review*, **6**(4): 90-103.
- [8] Atondo-Valdez O. F. (2019). Seguridad de datos personales en e-salud basado en caos. *Tesis de licenciatura, Universidad Autónoma de Baja California*, 1-67.
- [9] Crespo-Michel A. (2018). Información médica en esquema multiusuario. *Tesis de licenciatura, Universidad Autónoma de Baja California*, 1-70.
- [10] Cáceres-Méndez E., Castro-Díaz S. y Gómez-Restrepo C. (2011). Telemedicina: historia, aplicaciones y nuevas herramientas en el aprendizaje. *Universitas Médica*, **52**(1): 11-35.

- [11] Pinto G., Ramírez E. y Estupiñán L. (2011). Análisis de seguridad para el manejo de la información médica en telemedicina. *Ciencia e Ingeniería Neogranadina*, **21**(2): 57-89.
- [12] Monteagudo L., Serrano L. y Hernández C. (2005). La telemedicina: ¿ciencia o ficción?. *Anales del Sistema Sanitario de Navarra*, **28**(3): 309-323.
- [13] Prados-Castillejo J. A. (2013). Telemedicina, una herramienta también para el médico de familia. *Atención Primaria*, **45**(3): 129-132.
- [14] Xue Y. y Liang H. (2007). Analysis of Tele-medicine Diffusion: The Case of China. *IEEE transactions on information technology in biomedicine*, **11**(2): 231-233.
- [15] Sicurello F. (2001). Some Aspects on Tele-medicine and Health Network. *2nd International Symposium on Image and Signal Processing and Analysis, Pula, Croatia*, 651-654.
- [16] Toninelli A., Montanari R. y Corradi A. (2009). Enabling secure service discovery in mobile healthcare enterprise networks. *IEEE Wireless Communications*, **16**(3): 24-32.
- [17] Hu B., Bai J. y Ye D. (1997). An internet based communication server for Telemedicine. *Proceedings of the 19th International Conference Engineering in Medicine and Biology Society, Chicago, USA*, 981-982.
- [18] Groppe K. (2010). A Guide for Small Health Care Practices. *Transforming health through information and technology*, **1**(2): 1-11.
- [19] Brand K. y Boonen H. (2004). IT Governance based on COBIT 4.0: A management guide. *Van Haren*, 3ra Ed., pp. 209.
- [20] Knisley J. R. y Werchan H. A. (1994). Tele-metry: Enhancing tactical network management. *Proceedings of TCC'94 Tactical Communications Conference, Fort Wayne, USA*, 235-244.
- [21] Zavala-Villeda J. A. (2017). Descripción del electrocardiograma normal y lectura del electrocardiograma. *Electrocardiografía básica para anestesiólogos*, **40**(1): 210-213.
- [22] Gonzáles J. (2013). Freescale Solutions for Electrocardiograph and Heart Rate Monitor Applications. *Freescale Semiconductor*, AN4323.
- [23] Kocarev L. (2001). Chaos-based cryptography: a brief overview. *IEEE Circuits and Systems Magazine*, **1**(3): 6-21.
- [24] Gallego-Valencia J. P. (2010). Aplicación de la teoría de caos para el análisis y pronóstico de series de tiempo financieras en Colombia. *Tesis de Maestría, Universidad Nacional de Colombia*, 1-67.

- [25] Torres N. (2006). Caos en sistemas biológicos. *Revista digital de divulgación matemática de la Real Sociedad Matemática Española*, **2**(1): 1-13.
- [26] Madrid-Casado C. M. (2010). Historia de la teoría del caos contada para escépticos: Cuestiones de génesis y estructura. *Encuentros multidisciplinares, España*, **34**(3): 1-15.
- [27] Belmonte R., Noriega D. y Quintero A. (2006). Sincronización de sistemas complejos. *Trabajo de investigación, Centro de Investigación Científica y de Educación Superior de Ensenada*, 1-99.
- [28] Pisemskaya N. (2007). El lenguaje y la teoría del caos. *Opción*, **23**(53): 38-51.
- [29] Lorenz E. N. (1963). Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, **20**(2): 130-141.
- [30] Pecora L. M. y Carroll T. L. (1990). Synchronization in Chaotic Systems. *Physical Review Letters*, **64**(8): 821-825.
- [31] López-Gutiérrez R. M., Posadas-Castillo C., López-Mancilla D. y Cruz-Hernández C. (2009). Communicating via robust synchronization of chaotic lasers. *Chaos, Solitons and Fractals*, **42**(1): 227-285.
- [32] Arellano-Delgado A., López-Gutiérrez R. M., Méndez-Ramírez R. y Cruz-Hernández C. (2017). Emergence of hyperchaos and synchronization in a simple autonomous discrete system. *International Symposium on Nonlinear Theory and Its Applications, Cancun, México*, 4-7.
- [33] Sira-Ramírez H. y Cruz-Hernández C. (2001). Synchronization of chaotic systems: a generalized hamiltonian systems approach. *International Journal of Bifurcation and Chaos*, **11**(5): 1381-1395.
- [34] Jakimoski G. y Kocarev L. (2001). Chaos and Cryptography: Block Encryption Ciphers Based on Chaotic Maps. *IEEE Transactions on Circuits and Systems*, **48**(2): 163-169.
- [35] Schneier B. (1996). Applied Cryptography: Protocols, Algorithms, and Source Code in C. *John Wiley y Sons*, 2nd Ed., pp. 1027.
- [36] Preneel B., Rijmen V. y Bosselaers A. (1998). Recent Developments in the Design of Conventional Cryptographic Algorithms. *Lecture Notes in Computer Science*, **1528**(1): 105-130.
- [37] Rísquez F. (2008). La Teoría del Caos: Modelo de interpretación epistémica e instrumento de solución. *Trabajo de investigación, Universidad Central de Venezuela*, 1-25.
- [38] Alligood K., Sauer T. y Yorke J. (1997). An introduction to dynamical systems. *Springer Science y Business Media*, 1ra Ed., pp. 603.

- [39] Lara L., Stoico C., Machado R. y Catagnino M. (2003). Estimación de los exponentes de Lyapunov. *Revista de Mecánica Computacional*, **22**(1): 1441-1451.
- [40] Dubeibe F. L. (2013). Cálculo del máximo exponente de Lyapunov con Mathematica. *Revista Colombiana de Física*, **45**(1): 151-155.
- [41] May R. (1976). Simple mathematical models with very complicated dynamics. *Nature*, **261**: 459-467.
- [42] Prasad K. y Gnanajeyaraman R. (2009). Analysis of chaotic-chebyshev polynomials using on public key cryptosystems. *Computer Science y Telecommunications*, **6**(23): 116-123.
- [43] Saha B., Malasani S. y Seventline J. (2015). Application of Modified Chaotic Sine Map in Secure Communication. *International Journal of Computer Applications*, **113**(13): 9-13.
- [44] Ahmada M., Farooqb O. y Blackledge J. (2010). Chaotic image encryption algorithm based on frequency domain scrambling. *Information Processing Letters*, **13**(9): 1535-1539.
- [45] Hénon M. (1976). A Two-dimensional Mapping with a Strange Attractor. *Communications in Mathematical Physics*, **50**(1): 69-77.
- [46] Hua Z. y Zhou Y. (2016). Image encryption using 2D Logistic-adjusted-Sinemap. *Information Sciences*, **339**: 237-253.
- [47] Badola P., Tambe S. y Kulkarni B. (1992). Driving systems with chaotic signals. *Physical Review*, **46**(10): 635-637.
- [48] Ushio T. (1995). Chaotic synchronization and controlling chaos based on contraction mappings. *Physics Letters A*, **198**(1): 14-22.
- [49] Abrego F. J. y Moiola J. L. (2013). Lyapunov Exponent Analysis Applied to a Hyperchaotic Prey-predator Model. *IEEE Latin America Transactions*, **11**(1): 230-235.
- [50] Arroyo D., Álvarez G. y Li S. (2008). Some Hints for the Design of Digital Chaos-Based Cryptosystems: Lessons Learned from Cryptanalysis. *IFAC Proceedings Volumes*, **42**(7): 171-175.
- [51] Banerjee S., Yorke J. y Grebogi C. (1998). Robust Chaos. *Physical Review Letters*, **80**(1): 3049-3052.
- [52] Murillo-Escobar M. A. (2015). Diseño de un algoritmo de cifrado caótico y su implementación en microcontrolador para aplicaciones embebidas. *Tesis Doctoral, Universidad Autónoma de Baja California*, 1-142.
- [53] Contreras J. M. (2004). Introducción a la criptografía. *DYNA*, **79**(2): 6-10.

- [54] Palacios R. y Delgado V. (2006). Introducción a la criptografía: tipos de algoritmos. *Canales de mecánica y electricidad*, **1**(5): 42-46.
- [55] López M. (2017). Criptografía y seguridad en computadores. *Creative commons*, 4ta Ed., pp. 260.
- [56] Kerckhoffs A. (1883). La cryptographie militaire. *Journal des sciences militaires*, **9**: 5-38.
- [57] García E. y López M. (2005). Una introducción a la criptografía. *Universidad de castilla*, 1ra Ed., pp. 219.
- [58] Solana P. (2009). Antecedentes y perspectivas de estudio en historia de la Criptografía. *Proyecto final de carrera, Universidad Carlos de Madrid III*, 1-273.
- [59] Stix G. (2005). Criptografía cuántica comercial. *Investigación y ciencia*, **342**: 55-59.
- [60] Marwan S., Shawish A. y Nagaty K. (2016). DNA-based cryptographic methods for data hiding in DNA media. *Biosystems*, **150**: 110-118.
- [61] Gu G., Ling J., Xie G. y Li Z. (2016). A chaotic cipher based packet body encryption algorithm for JPEG 2000 images. *Signal Processing: Image Communication*, **40**: 52-60.
- [62] Alvarez G. y Li S. (2006). Some basic cryptographic requirements for chaos-based cryptosystems. *International Journal of Bifurcation and Chaos*, **16**(8): 2129-2151.
- [63] Días J. (2006). Principios básicos de la criptografía. *Documenta e Instrumenta*, **4**: 47-59.
- [64] Alvarez G., Montoya F., Romera M. y Pastor G. (2003). Cryptanalysis of a chaotic secure communication system. *Physics Letters*, **306**: 200-205.
- [65] Valvano J. (2003). Introducción a los sistemas de microcomputadora embebidos: Simulación de Motorola G811 Y G812. *International Thomson Editores*, 1ra Ed., pp. 480.
- [66] Sager E. (2013). Introducción general a los sistemas embebidos. *Simposio argentino de sistemas embebidos, Buenos Aires, Argentina*, 5-8.
- [67] Muñoz-Frías J. D. (2011). Sistemas empotrados en tiempo real. *José Daniel Muñoz Frías*, 1ra Ed., pp. 168.
- [68] Manuel-Cruz J. (2013). Sistema Embebidos: Sistemas de tiempo real. *Simposio argentino de sistemas embebidos, Buenos Aires, Argentina*, 3-12.
- [69] ATmega8515 Atmel. (2010). 8-bit Microcontroller with 8K Bytes In-System Programmable Flash. *Atmel Corporation*, 2512K-AVR-01/10.

- [70] PIC16F87XA Microchip. (2003). PIC16F87XA 28/40/44-Pin Enhanced Flash Microcontrollers. *Microchip Technology*, DS39582B-232.
- [71] MK53DN512ZCLQ10. (2011). K53 Sub-Family Reference Manual. *Freescale semiconductor*, K53P144M100SF2RM/REV 6.
- [72] NXP Freescale Semiconductor. (2009). TWR-MCF51MM. *Freescale Semiconductor*, User Manual Rev. 1.0.
- [73] MCF51MM2567128 Freescale Semiconductor. (2010). MCF51MM256/128: Integrated analog and ultra-low power with USB connectivity. *Freescale semiconductor*, MCF51MM256FS/REV 1.
- [74] NXP Freescale Semiconductor. (2010). MED-EKG. *Freescale Semiconductor*, User Manual Rev. 1.0.
- [75] Cavusoglu U., Kacar S., Pehlivan I. y Zengin A. (2017). Secure image encryption algorithm design using a novel chaos based S-Box. *Chaos, Solitons and Fractals*, **95**: 92-101.
- [76] Amigo J. M., Kocarev L. y Szczepanski J. (2007). Theory and practice of chaotic cryptography. *Physics Letters A*, **366**(3): 211-216.
- [77] Jakimoski G. y Kocarev L. (2001). Chaos and cryptography: block encryption ciphers based on chaotic maps. *IEEE Transactions on Circuits and Systems*, **48**(2): 163-169.
- [78] Kotulsk Z. y Szczepanski J. (1997). Discrete chaotic cryptography. *Proc. NEEDS 1997, Kolymbari, Crete*, 1-11.
- [79] Murillo-Escobar M. A., Cardoza-Avendaño L., López-Gutiérrez R. M. y Cruz-Hernández C. (2017). A Double Chaotic Layer Encryption Algorithm for Clinical Signals in Telemedicine. *Journal of medical systems*, **41**(4): 1-17.
- [80] Méndez-Ramírez R. D. (2018). Implementación de osciladores caóticos en sistemas embebidos y aplicaciones. *Tesis Doctoral, Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California*, 1-133.
- [81] Abundiz-Pérez F., Cruz-Hernández C., Murillo-Escobar M. A., López-Gutiérrez R. M. y Arellano-Delgado A. (2016). A Fingerprint Image Encryption Scheme Based on Hyperchaotic Rössler Map. *Mathematical Problems in Engineering*, **5**: 1-15.
- [82] Murillo-Escobar M. A., Cruz-Hernández C., Abundiz-Pérez F. y López-Gutiérrez R. M. (2015). A robust embedded biometric authentication system based on fingerprint and chaotic encryption. *Expert Systems with Applications*, **42**: 8198-8211.
- [83] Murillo-Escobar M. A., Cruz-Hernández C., Abundiz-Pérez F. y López-Gutiérrez R. M. (2016). Implementation of an improved chaotic encryption algorithm for realtime embedded systems by using a 32-bit microcontroller. *Microprocessors and Microsystems*, **45**: 297-309.

- [84] Murillo-Escobar M. A., Cruz-Hernández C., Abundiz-Pérez F., López-Gutiérrez R. M. y Acosta Del Campo O. R. (2015). A RGB image encryption algorithm based on total plain image characteristics and chaos. *Signal Processing*, **109**: 119-131.
- [85] Xi Y., Zhang X. y Ye R. (2016). Color image encryption based on multiple chaotic systems. *International Journal of Network Security and Its Applications*, **8**(5): 39-50.
- [86] Lin C., Shih S. y Zhu J. (2014). Chaos Based Encryption System for Encrypting Electroencephalogram Signals. *Journal of Medical Systems*, **38**(5): 38-49.
- [87] Raeiatibanadkooki M., Quchani S., Khalilzade M. y Bahaadinbeigy K. (2016). Compression and Encryption of ECG Signal Using Wavelet and Chaotically Huffman Code in Telemedicine Application. *Journal of Medical Systems*, **40**(3); 73-80.
- [88] Lab Tutorial for MCF51MM256. (2010). MCF51MM256 Electrocardiogram (EKG) with Freescale USB stack. *Freescale semiconductor*, MCF51MMELAB1 / REV 0.
- [89] Murillo-Escobar M. A., Meranza-Castillón M. O., López-Gutiérrez R. M. y Cruz-Hernández C. (2019). Suggested Integral Analysis for Chaos-Based Image Cryptosystems. *Entropy*, **21**(8); 815.