UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO ENSENADA

MEDICIÓN DE ECG EN TIEMPO REAL Y ENCRIPTADO CAÓTICO EN MICROCONTROLADOR PARA MONITOREO REMOTO EN E-SALUD

TESIS

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

INGENIERO EN ELECTRÓNICA

presenta:

ALVARO RODRIGUEZ HERNANDEZ

Ensenada, Baja California, México, Noviembre de 2023.

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO ENSENADA

MEDICIÓN DE ECG EN TIEMPO REAL Y ENCRIPTADO CAÓTICO EN MICROCONTROLADOR PARA MONITOREO REMOTO EN E-SALUD

TESIS

Que para obtener el grado de Ingeniero en Electrónica presenta:

Alvaro Rodriguez Hernandez

Aprobada por el siguiente comité:

Dra. Rosa Martha López Gutiérrez
Director del comité

Dr. Miguel Ángel Murillo Escobar
Codirector

Dr. César Cruz Hernández
Miembro del comité

Dra. Liliana Cardoza Avendaño
Miembro del comité

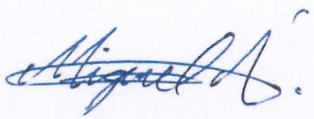
M.I. Daniel Murillo Escobar
Miembro del comité

Ensenada, Baja California, México. Noviembre de 2023

RESUMEN de la tesis de **Alvaro Rodriguez Hernandez**, presentada como requerimiento parcial para obtener el grado de INGENIERO en ELECTRÓNICA, del programa de Licenciatura de la Universidad Autónoma de Baja California. Ensenada, Baja California, México. Noviembre de 2023.

MEDICIÓN DE ECG EN TIEMPO REAL Y ENCRIPTADO CAÓTICO EN MICROCONTROLADOR PARA MONITOREO REMOTO EN E-SALUD

Resumen aprobado por:



Dra. Rosa Martha López Gutiérrez
Director de tesis

Dr. Miguel Ángel Murillo Escobar
Codirector de tesis

En este trabajo de tesis de licenciatura, se realiza el diseño e implementación en un sistema embebido de un algoritmo criptográfico basado en caos para el resguardo de señales de electrocardiograma (ECG) capturadas en tiempo real para implementación en sistemas de telemedicina.

Se realiza un análisis al mapa de Hénon y se exponen las razones por las que se ha seleccionado como base del sistema de cifrado. Se parte desde el diseño del algoritmo propuesto, haciendo una explicación sobre el funcionamiento del mismo, posteriormente se presenta el esquema de transmisión de la señal encriptada, la implementación del algoritmo en un sistema embebido de 32 bits y se exponen los resultados obtenidos.

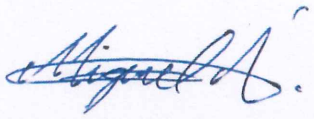
Finalmente, se realiza un breve análisis de los resultados obtenidos, y se realizan algunas evaluaciones de seguridad al sistema y los resultados obtenidos, esto con el fin de establecer si el sistema es seguro para transmitir datos en telemedicina, se exponen las conclusiones del trabajo y algunos puntos de mejora para trabajos a futuro.

Palabras clave: caos, sistema embebido, telemedicina, mapa Hénon, análisis de seguridad.

Abstract of the thesis presented by **Alvaro Rodriguez Hernandez**, as a partial requirement to obtain the degree in ELECTRONICS ENGINEER, of the program of the Autonomous University of Baja California. Ensenada, Baja California, Mexico. November, 2023.

REAL-TIME ECG MEASUREMENT AND CHAOTIC ENCRYPTION IN MICROCONTROLLER FOR REMOTE MONITORING IN E-HEALTH

Abstract approved by:



Dra. Rosa Martha López Gutiérrez
Thesis director

Dr. Miguel Ángel Murillo Escobar
Thesis codirector

In this thesis work, the design and implementation of a chaos-based cryptographic algorithm for safeguarding real-time electrocardiogram (ECG) signals on an embedded system are conducted.

An analysis of the Hénon map is performed, elucidating the reasons behind its selection as the basis for the encryption system. The process commences from the design of the proposed algorithm, providing an explanation of its functionality. Subsequently, the encrypted signal transmission scheme is presented, detailing the algorithm's implementation on a 32-bit embedded system, and showcasing the obtained results.

Finally, a brief analysis of the obtained results is carried out, along with security evaluations of the system and its outcomes, aimed at determining its suitability for use in a secure telemedicine system. The conclusions of the work are presented, along with some areas for improvement in future research endeavors.

Keywords: chaos, embedded system, telemedicine, Henon map, security evaluations.

A mi familia

Agradecimientos

A mi familia, mis padres, Paulino y María, quienes han sido mis soporte y me han motivado a seguir adelante en todos los proyectos que he tenido, a mis hermanos, Daniel y Xitlali, por estar conmigo y apoyarme.

A la Dra. Rosa Martha López Gutíerrez, por darme la oportunidad de realizar este trabajo de tesis y por el apoyo que me ha brindado a lo largo de mi formación en la carrera de Ingeniería en Electrónica.

Al Dr. Miguel Angel Murillo Escobar, por el apoyo brindado para la realización de este trabajo de tesis, por compartir su conocimiento y consejos, así como por el tiempo dedicado a ayudar en la mejora de este trabajo.

A mi comité de tesis, Dr. César Cruz Hernández, Dra. Liliana Cardoza Avendaño y M.I. Daniel Murillo Escobar, por su gran contribución a mi desarrollo académico.

A la Universidad Autónoma de Baja California (UABC), por haber sido el lugar donde me desarrolle como estudiante de Ingeniería en Electrónica y a los profesores que son parte de esta los cuales me facilitaron los conocimientos necesarios para mi formación académica.

A Cristina, por estar siempre al pendiente de mi y no dejar que me rindiera.

A la Sociedad Científica Juvenil (SCJ), por darme un espacio en el cual puedo compartir mis conocimientos y proyectos así como mis inquietudes y problemas.

Al Consejo Nacional de Humanidades, Ciencia y Tecnología (CONAHCYT), por el apoyo económico recibido a través del proyecto de Investigación en ciencia básica entre instituciones, “Sincronización de Sistemas Complejos y Algunas Aplicaciones”. Ref. 166654 y continuación (A1-S-31628).

Ensenada, B.C., México.
Noviembre de 2023

Alvaro Rodriguez Hernandez

Tabla de Contenido

Lista de Figuras

Lista de Tablas

Capítulo 1

Introducción

No podemos negar el notable avance en las tecnologías referentes a las telecomunicaciones en los últimos años. Hemos experimentado de primera mano el aumento en las velocidades de transmisión de datos y con ello un aumento considerable en la calidad de los diferentes servicios que típicamente utilizamos (música, video, correo electrónico, sistemas de mensajería instantánea), pero no solo este tipo de servicios se han beneficiado con estos avances, el sector salud también se ha visto enormemente beneficiado.
Hace no mucho tiempo, surgió una emergencia sanitaria a nivel global que dificultó la mayoría de los procesos en el mundo. Las personas se vieron en la necesidad de adaptarse a un estilo de vida que no habían experimentado antes. Diferentes servicios tuvieron que migrar de manera súbita a sistemas de atención a distancia, los trabajadores ahora tenían que aprender rápidamente a lidiar con problemas de conexión, y la inexperiencia de sus usuarios en cuestiones tecnológicas. El servicio de salud no fue una excepción a todo esto, los centros médicos se saturaron y crecía la necesidad de encontrar una manera en que los médicos pudieran atender a los pacientes sin necesidad de que estos salieran de sus hogares, fue ahí cuando la telemedicina entró en acción, pues, aunque ya se había explorado durante un tiempo, las aplicaciones de la telemedicina eran escasos, Pero durante la crisis provocada por el COVID-19, la telemedicina ganó relevancia no solo en el mundo, también en México [1,2].
La telemedicina hizo posible que el tratamiento de los casos fuera seguro y eficaz tanto para el paciente como para el médico, evitando el contacto directo se lograron prevenir contagios [3].
Los sistemas de telemedicina se basan principalmente en un módulo de consulta(paciente) y un módulo de diagnóstico(personal médico), estos están conectados de manera inalámbrica por medio de internet o señales de radio, durante la teleconsulta, el médico y el paciente se encuentran en un intercambio constante de información, la cual, como se mencionó anteriormente, pasa a través de medios inseguros.

Las redes médicas a distancia están usualmente conformadas por redes multipunto, es decir que múltiples usuarios se están comunicando al mismo tiempo, ya sean múltiples pacientes al centro médico o múltiples médicos monitoreando y diagnosticando al paciente [4].

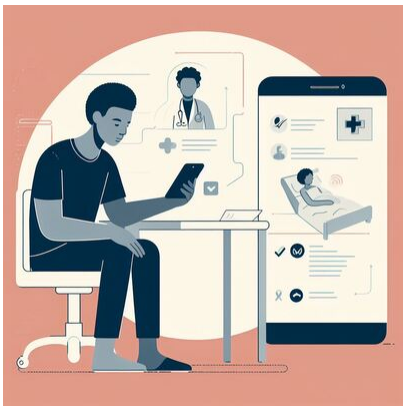


Figura 1.1: Medicina a distancia.

Debido a la crisis sanitaria mencionada anteriormente, el número de servicios y tratamientos referentes a la salud que se proporcionaron a distancia aumentó con rapidez, y con ello, el flujo de información médica que se compartía a través de medios electrónicos creció a la par. Desafortunadamente, canales como el internet no son medios de transmisión seguros, estos presentan grandes vulnerabilidades al ser medios de transmisión públicos, personas con el conocimiento suficiente, pueden interceptar las líneas de transmisión, secuestrar dicha información y usarla de manera negativa o alterarla.

Es aquí donde la criptografía entra en acción, ya que en ocasiones no basta con proteger el canal por el cual se envía la información, ya que, como se mencionó anteriormente, estos pueden ser vulnerados facilitando el robo de información. La criptografía permite que, a pesar de que los datos sean robados, aquel que lo haga no tiene la posibilidad de entender el mensaje que se envía, ya que este se encuentra oculto entre un cúmulo de datos aparentemente corruptos o aleatorios. El objetivo principal de la criptografía es, que dicha información se mantenga en un estado de confidencialidad, aun cuando esta haya sido secuestrada por terceros.

Dentro de la criptografía podemos encontrar dos clases importantes de métodos criptográficos, los convencionales como el 3DES y el AES, y los no convencionales, como lo son los métodos que utilizan dinámicas caóticas para ocultar los mensajes enviados.

En el caso de este trabajo de tesis, se trabajará en el diseño e implementación de un sistema de encriptado del segundo grupo mencionado anteriormente. Utilizaremos un sistema que genera una señal caótica como medio de encriptado y las condiciones iniciales de dicho sistema tomarán el papel de las claves a utilizar para el encriptado y desencriptado del mensaje, todo dentro de un ambiente digital.

1.1. Motivación

Con todo el avance que se ha visto en las tecnologías de telecomunicación y la manera en que los servicios se han adaptado a brindarse por estos medios, es obvio el gran aumento de información que se comparte a través de los canales de dichos medios. Es por este aumento, que se debe dar mucha más atención a la seguridad de los datos médicos que se comparten o se resguardan. La gran vulnerabilidad de la red informática con la que interactuamos a diario no es un secreto, todos sabemos que nuestros datos están en constante riesgo debido a que siempre estamos conectados a internet, ya sea de forma directa o indirecta, y esto es un panorama que causa miedo aún más cuando nos damos cuenta de que nuestro historial médico se encuentra albergado en bases de datos hospitalarias que están conectadas a nubes en internet.

El hecho de que la información médica de un paciente se encuentre en un servidor de una institución sin seguridad más allá de la contraseña del computador de dicha institución, abre la puerta a que esta información pueda caer en manos equivocadas y que se use para fines negativos. Un ejemplo de esto es la fuga de información del Hospital Presbiteriano de Nueva York y La Universidad de Columbia, durante este evento, la información de 6,800 pacientes terminó en buscadores en internet. Se sabe que la información de los pacientes es de carácter confidencial y que dicha información sólo se encuentra bajo resguardo en las instituciones de salud, por lo que dicho evento resultó en multas que llegaron a los 4,800,000.00 dólares por el mal manejo de la información [5,6]. Es aquí donde nos damos cuenta de la gran importancia que tiene el buscar nuevas formas de proteger la información del paciente. Es por esto que nace la idea de este trabajo de tesis, en el cual se realizará el resguardo de bioseñales con ayuda de criptografía basada en cáos.

1.2. Objetivos y alcances de la tesis

La idea de este trabajo de tesis surge de la necesidad de incrementar la seguridad de los datos médicos y garantizar el resguardo de las mismas, es por ello que se busca alcanzar el siguiente *objetivo general*:

Implementar un sistema de encriptado con caos para señales de ECG adquiridas en tiempo real en cuadros de 10 segundos y muestras de 100 Hz con implementación en microcontrolador con comunicación inalámbrica.

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

1. Realizar la lectura de ECG en tiempo real en sistema embebido 32 bits.

2. Implementar encriptado caótico en sistema embebido.

3. Transmitir de forma remota el criptograma al receptor autorizado.

4. Realizar análisis de seguridad y eficiencia computacional.

1.3. Organización del manuscrito

El contenido de este trabajo de tesis se distribuye de la siguiente manera:

- **Capítulo 1:** se realiza la introducción al trabajo de tesis y se plantean la motivación y los objetivos.
- **Capítulo 2:** se expone el concepto de telemedicina, se plantea un poco de contexto histórico del mismo y se presentan aspectos sobre la seguridad en los sistemas.
- **Capítulo 3:** se presentan los conceptos de caos y criptografía, se da un repaso histórico de ambos conceptos y se introduce la relación que estos tienen en aplicaciones.
- **Capítulo 4:** se explica el algoritmo propuesto basado en caos para encriptado de señales ECG en tiempo real.
- **Capítulo 5:** se presenta la implementación en Hardware del algoritmo de encriptado en tiempo real en un sistema embebido de 32 bits para su aplicación en un sistema de telemedicina, se realiza una serie de análisis de seguridad al criptograma.
- **Capítulo 6:** se plantean las conclusiones de este trabajo y algunos puntos a considerar para mejoras en trabajos futuros.

Capítulo 2

Telemedicina

En este capítulo, se describe el marco histórico de la telemedicina, se presenta al concepto de la misma y se exploran algunos puntos claves para entender los sistemas y la importancia de garantizar la seguridad en los mismos.

2.1. Introducción

Dando un breve vistazo al pasado, podemos darnos cuenta de que el punto más importante de contar con un sistema de salud, era justamente tener la accesibilidad a que un médico, presente de manera física, pudiera consultar y dar un diagnóstico sobre el malestar de una persona. Desafortunadamente, existen múltiples factores que generan dificultad al momento de querer brindar o acceder a servicios de salud, entre dichos factores podemos resaltar: la discapacidad por enfermedad, la mala distribución del servicio médico en las comunidades y la distancia que hay que recorrer para que, ya sea, los servicios o el paciente pasen de un punto a otro para la consulta [7].
La medicina o los servicios médicos, han sido siempre de gran importancia para el ser humano, es por ello que se debe dar mucha importancia a la búsqueda de avances tecnológicos en base a ello. Desafortunadamente, los servicios de salud son vulnerables al ambiente en el que se encuentran, es decir, que existen múltiples factores que afectan cuando se plantea el brindar los servicios médicos de manera adecuada [8].

El concepto de telemedicina se puede rastrear hasta principios de la década de los 70, dicho concepto surge con el desarrollo tecnológico de la época (equipos de cómputo, el internet, dispositivos móviles, etc.).
La telemedicina surge como una respuesta a las dificultades que se presentaron al momento de llevar los servicios de salud a zonas geográficas de difícil acceso, más enfocado a áreas rurales alejadas y países en vías de desarrollo [9].
Aunque parece algo sencillo, definir la telemedicina se vuelve algo complejo debido a la cantidad de definiciones y matices que encontramos a lo largo de su estudio, según una revisión literaria realizada en 2007 por un grupo de investigadores, existen alrededor de 100 formas de definir a la telemedicina [10], por lo que para abordarla en esta ocasión, tomaremos el concepto que no ofrece la Organización Mundial de la Salud (OMS).

Según la Organización Mundial de la Salud (OMS), podemos definir a la telemedicina como "la prestación de servicios de atención médica por todos los profesionales de la salud, mediante el uso de tecnologías de comunicación e intercambio de información válidas, tanto para el diagnóstico, como para el tratamiento o la prevención de enfermedades y lesiones" [11], dependiendo de la forma en que se de la comunicación entre los individuos, la aplicación de la telemedicina se puede dividir en dos métodos: el síncrono, es decir, que la comunicación médico-paciente se esté dando por medio de una videoconferencia o por audio en tiempo real (figura 2.1), y asíncrono, es decir, que la comunicación entre ambas partes se realiza por medio de correos electrónicos o mensajes de texto en los cuales se tiene un tiempo de espera entre respuestas que puede ser de pocos minutos hasta días (figura 2.2).

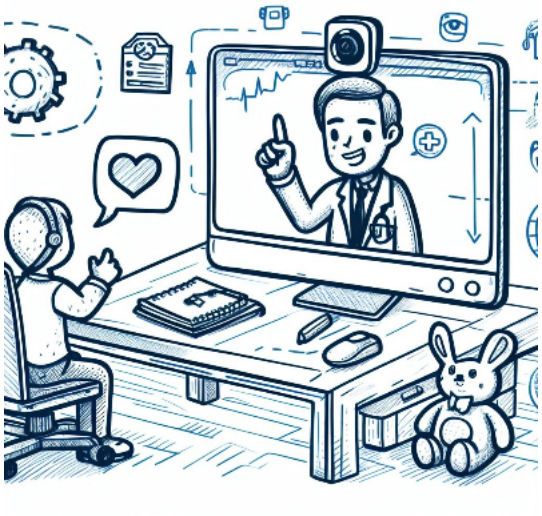


Figura 2.1: Comunicación síncrona (teleconsulta).

Para ambos escenarios, los requerimientos en infraestructura son muy similares, el cambio mayor se presenta en la espera del paciente y el médico para obtener datos y resultados, pero en ambos casos, el personal médico debe estar al tanto del paciente y el paciente debe estar siendo monitoreado con equipo especial (si lo requiere).

2.2. Recuento histórico

En los últimos años, a la telemedicina se le ha considerado como una disciplina que se encuentra a medio camino entre la medicina y la tecnología, la cual genera una fuerte demanda en el desarrollo de tecnologías de la comunicación. El concepto de telemedicina es relativamente nuevo, pero los sistemas que utilizan medios para transmitir información a distancia aparecieron mucho antes que el concepto que hoy engloba a la telemedicina.

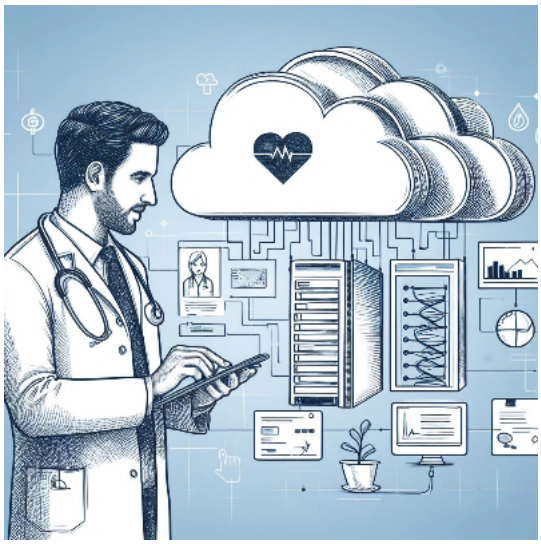


Figura 2.2: Comunicación asíncrona (base de datos).

Todo avance en cuestiones de ciencias de la salud ha estado ligado al constante avance en las tecnologías de comunicación, esto nos ha permitido sobrepasar situaciones adversas y hostiles, tales como desastres naturales, epidemias e incluso, conflictos armados.

La telemedicina, surge y se vuelve relevante siempre en escenarios en los que se encuentran planes de salud bastante inconsistentes, poco eficaces y normalmente asociados a una cobertura poblacional bastante baja, añadiendo a esto el poco apoyo de los gobiernos a estos proyectos, lo cual genera una amplia limitación al desarrollo de los mismos, obtenemos como resultado un panorama realmente desesperanzador para la aplicación óptima de estos sistemas [12].

Si bien, es difícil rastrear con precisión los orígenes de la telemedicina, para efectos de este trabajo, tomaremos como referencia el evento más antiguo mencionado en un artículo de la revista AITT, en el cual mencionan el uso de un Heliografo en 1347 para comunicarse a distancia entre puntos y de esta forma delimitar el territorio afectado por la peste bubónica, previniendo de esta manera la expansión de dicha enfermedad al dirigir los grupos migratorios lejos de estas zonas [12,13].
Entonces, partiendo de aquí, nuestro recuento histórico de los hitos más importantes de la telemedicina sería el siguiente, en orden cronológico:

- **1347:** Uso de un heliógrafo para direccionar a los grupos migratorios lejos de las áreas afectadas por la peste bubónica en Europa.
- **1861:** Uso del telégrafo para la transmisión de datos epidemiológicos, para la gestión y transporte tanto de pacientes como de insumos médicos durante la guerra civil Estadounidense.
- **1900:** Invención del teléfono, con esto se comenzó a agilizar el proceso de infor-

mar sobre situaciones en las que el personal médico era requerido en distintos lugares. Durante el mismo año, en Australia ,se comenzaron a realizar intentos por desarrollar equipos para transmitir radiografías por telégrafo.

- **1905:** Willem Einthoven, utiliza el prefijo "tele"por primera vez cuando diseñó un novedoso aparato que permitía compartir electrocardiogramas con un colega situado a 500 metros de su lugar de trabajo ubicado en Leiven.
- **1925:** Hugo Hernsback presenta una visión de lo que sería más adelante la telemedicina en la portada de la revista Radio News, en la revista presenta también un esquema de lo necesario para llevar a la realidad esta visión.
- **1930:** A finales de 1930, en Alaska y Australia, se comenzaba a utilizar el teléfono y el telégrafo como medios de transmisión para datos médicos como parte de programas nacionales. Esto se utilizará más adelante durante la guerra de Vietnam y Corea para controlar el tránsito y entrega de insumos médicos y naves de rescate.
- **1950:** Científicos de la NASA diseñaron y presentaron por primera vez un sistema para monitorear las funciones fisiológicas de los astronautas en el espacio.
- **1955:** El doctor Albert Jutras en Montreal, realiza la primera teleradiología, con el objetivo de evitar la exposición a las altas dosis de radiación que incidían en las fluoroscopias.
- **1959:** Por primera vez, se consigue la transmisión correcta de imágenes radiológicas a través de una línea telefónica. Casi al mismo tiempo, la doctora cecil Wittson, comienza a impartir sus primeros cursos de teleeducación y telepsiquiatría.
- **1972:** Se da inicio al programa STARPAHC el cual tenía como objetivo brindar asistencia médica a los nativos ubicados en Papago, Arizona.
- **1986:** Se realiza la primera videoconferencia entre médicos en Noruega.
- **1988:** La nasa lanza el programa **Space Bridge** con el propósito de colaborar con Armenia y Ufa socorriendo a estos en distintos momentos de vulnerabilidad después de catástrofes, las conexiones se realizaron con video en una dirección; voz y fax bidireccionales entre centros medicos armenios y 4 hospitales en estados unidos.
- **1991:** Se realiza la primera cuantificación de ADN a distancia del mundo, utilizando análisis de imágenes en la cátedra UNESCO de telemedicina.
- **1996:** El Dr. Carbajal Ramos y el Dr. Harry Miller realizaron un histórico hito médico al llevar a cabo los primeros casos de cirugía asistida por robot en México [15].
- **2000:** La empresa norteamericana Intuitive Surgical diseña y lanza el sistema Da Vinci, siendo un robot para cirugías [16].

- **2001:** Se realiza la primera intervención quirúrgica transatlántica, doctores norteamericanos extraen la vesícula a un paciente a distancia desde Nueva York hasta Francia.
- **2003:** Inicia el Proyecto Chileno Argonauta, para una red de telemedicina en la Antártica.
- **2006:** Mehran Anvari realiza una sutura a distancia a un paciente que vivía en la estación submarina Aquaris para simular una telecirugía en el espacio [17].
- **2017 - Actualidad:** Los investigadores plantean escenarios esperanzadores para la robótica en la telemedicina en cuestiones de asistencia y seguimiento a distancia de pacientes [18].

2.3. Clasificación y factibilidad de la telemedicina

Como bien sabemos, el cambio en los entornos es inevitable, el sistema de salud por su parte, no es ajeno estos cambios con el pasar del tiempo. Los sistemas sanitarios están en constante búsqueda de nuevas estrategias de organización y alternativas a los modos de prestar los servicios sanitarios. Con la llegada de avances tecnológicos, las clasificaciones y especialidades en el campo médico han cambiado. En el caso de la telemedicina, con cada nuevo avance se abre la posibilidad de surgimiento a una nueva aplicación o clasificación de esta, la telemedicina se puede clasificar por su modo de aplicación o por su área de especialidad, es así pues, que podemos obtener los siguientes grupos o clases [20].

Por su modo de operación (ya se ha mencionado anteriormente, pero hacemos énfasis):

- **Síncrona:** hace referencia a un seguimiento en tiempo real utilizando herramientas de comunicación como audio, vídeo o transmisión de imágenes de alta resolución en vivo.
- **Asíncrona:** hacer referencia al uso de dispositivos de captura y resguardo de la información en bases de datos para su posterior uso en análisis y diagnóstico, haciendo uso de medios como el correo electrónico o nubes de datos para la transmisión y recepción de los archivos.

Por su área de aplicación:

- **Teleconsulta:** La teleconsulta es un método de asistencia en el cual múltiples médicos o instituciones se reúnen para generar un diagnóstico sobre un caso o casos en específico.

- **Telepresencia:** Asistencia a distancia a un paciente por parte de un profesional de la salud, ya sea para consulta, diagnóstico o seguimiento.
- **Telemonitorización:** Vigilancia remota a los parámetros fisiológicos y biométricos del paciente [21].

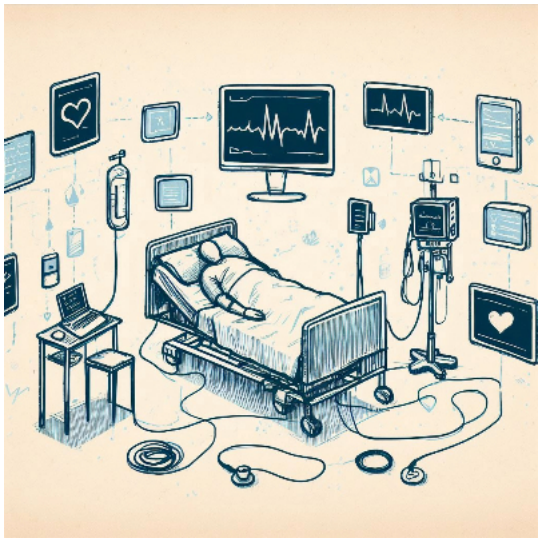


Figura 2.3: Telemonitorización.

- **Teleasistencia:** Provisión de cuidados básicos de salud a pacientes en su vida diaria, generalmente aplicado a ancianos o personas incapacitadas en condiciones delicadas [22].
- **Telecirugía:** La telecirugía es pues, a grandes rasgos la realización de intervenciones quirúrgicas a pacientes a distancia utilizando la asistencia de mecanismos y sistemas controlados por un médico.

Por su área de especialidad:

- **Telepediatría:** Asistencia a distancia a infantes por parte de un médico especialista.
- **Telepsiquiatría:** como se indica, es la prestación de servicios clínicos de psiquiatría a distancia [23].
- **Teledermatología:** Consiste en la evaluación de lesiones cutáneas y la evaluación de datos de laboratorios por parte de dermatólogos utilizando técnicas de telemedicina [24].
- **Teleaudiología:** Es la prestación de servicios audiológicos a pacientes a distancia, este rama se encuentra en estudio debido a las limitaciones de los sistemas de transmisión de audio que no garantizan al cien por ciento la fiabilidad de los resultados, aunque han demostrado estar muy cerca de lograrlo [25].

Además de los ejemplos mencionados, existen muchas más áreas de especialidad que se han adaptado o que se están adaptando a prestarse por medio de sistemas a distancia.

La telemedicina comenzó como un modo de aproximar los servicios de salud a más personas, posteriormente comenzó a utilizarse como un medio de capacitación para el personal médico, y hoy en día se utiliza como un sistema de mejora a los servicios que se han prestado durante mucho tiempo.

La telemedicina no está exenta a tener desventajas, por ello, enumeramos algunas ventajas y desventajas de la telemedicina en su estado actual.

Ventajas:

1. Facilita la equidad en el acceso a los servicios sanitarios.
2. Brinda a los pacientes la asistencia médica especializada en zonas donde no se dispone de esta.
3. Reduce los tiempos de espera para la obtención de un diagnóstico y seguimiento a un padecimiento.
4. Facilita el manejo temprano del paciente previo a la llegada de equipos de emergencia y traslado.
5. Brinda la posibilidad de realizar consultas remotas para la atención primaria de los pacientes.

Desventajas:

1. Menor exactitud en imágenes transmitidas con respecto a las originales (cuando se trata de servicios de monitoreo o diagnóstico visual).
2. Aspectos relacionados a la confidencialidad de los datos que se compranet entre médico y paciente (seguridad).
3. Aumento en la demanda de especialistas capacitados en el campo y un posible desabasto de estos debido al rápido aumento.
4. Riesgo de pérdida de datos e imágenes debido a la compresión de los mismos para su transmisión.
5. Estado actual de las comunicaciones, las cuales podrían no solventar los requerimientos para la correcta implementación de sistemas a gran escala.

2.4. Seguridad en telemedicina

Como hemos estado viendo, la transmisión de información es el punto clave en la telemedicina, pues es aquí en donde tiene el punto fuerte al dar las grandes ventajas de una asistencia a distancia. De manera desafortunada, esto representa una gran debilidad en los sistemas, pues se requiere de mucha atención en lo que respecta a materia de seguridad.
Sabemos que la información médica de los pacientes es de carácter totalmente confidencial y que los hospitales sólo la tienen en resguardo, por lo que estos deben garantizar la seguridad de dicha información. En [5] y [26] se dan a conocer varios casos de ataques a los sistemas de resguardo de información en distintas instituciones médicas, lo que provocó que mucha información se perdiera o se filtrara en internet, esto significó grandes demandas a las instituciones y riesgos para los pacientes cuya información fue robada.

Las tecnologías de la comunicación y la información, son en su totalidad la base sobre la que se sostienen los proyectos de telesalud, sin embargo, estos no solo obtienen las ventajas, también se exponen a los riesgos que tienen relación a los aspectos de seguridad informática, como la autenticación, la confidencialidad, la privacidad y obviamente el rechazo de los pacientes a que su información esté moviéndose a través de sistemas informáticos.
El almacenamiento y transmisión de datos sensibles de los pacientes son parte de los retos que se enfrentan para poder garantizar un correcto funcionamiento de los proyectos de telesalud. Se deben establecer protocolos de seguimiento para detectar actividades sospechosas ya sean de personal interno o de terceros que deseen destruir, manipular o secuestrar la información confidencial, así como la prevención de ataques a la infraestructura de telecomunicaciones.

2.5. Conclusiones

En este capítulo, se dio una introducción y explicación del concepto de telemedicina, realizamos un recuento histórico de aquellos eventos que marcaron puntos importantes en la evolución de la telemedicina. Repasamos algunas de sus características y su clasificación. En el caso de este trabajo de tesis haremos uso del modo de aplicación Asíncrono, mencionado en los apartados 2.1 y 2.3 ya que buscamos que a pesar de capturar y encriptar las señales en tiempo real, los datos serán resguardados al momento, o en su defecto, transmitidos a otro lugar para su posterior resguardo en una base de datos.

Capítulo 3

Caos y criptografía

En este capítulo, se introduce el concepto de caos y sus aspectos esenciales, partiendo de un repaso de los antecedentes que dieron inicio a la actual teoría del caos y se hace un breve análisis al mapa de Hénon, mismo que se ha utilizado para la realización del trabajo. Además, se expone la criptografía, sus fundamentos y un breve repaso histórico.

3.1. Caos

Un sistema dinámico, es aquel cuya respuesta varía con el tiempo. Estos se pueden clasificar en dos grandes grupos: lineales y no lineales. En esta ocasión nos centraremos en un tipo de sistemas perteneciente al segundo grupo, nos referimos pues, a los sistemas no lineales con una dinámica caótica. Pero antes de entrar en lo que es un sistema caótico, debemos dar un repaso y comprender lo que es el caos.
El concepto de caos es definido usualmente como una condición de gran desorden y confusión, pero en el campo científico, más específico en la física y las matemáticas la palabra Caos, está asociada a un tipo de respuesta dado por sistemas no lineales, la cual se caracteriza por ser aperiódica y de comportamiento aparentemente aleatorio. A pesar de esto, el estudio de estos comportamientos no solo se encuentra en las ciencias exactas, el estudio de los sistemas con comportamiento de este tipo también puede aplicarse a las ciencias sociales.

3.1.1. Antedecentes

En épocas pasadas, la totalidad de conocedores de la ciencia consideraban que todo movimiento gobernado por un sistema dinámico era estrictamente lineal, es decir, que su evolución en el tiempo era regular, ya que sus estados sucesivos se generaban continuamente, unos a partir de otros. Contrario a lo anterior, a finales del siglo XIX, el matemático Henri Poincaré, descubrió que algunos sistemas cuyo comportamiento se veía gobernado por un conjunto de ecuaciones no lineales, comenzaba a generar una dinámica aparentemente inestable bajo ciertas condiciones [28], esto quedó como un mero dato curioso en su tiempo, pero nos sirve como un punto de partida a lo que más

adelante sería conocido como Teoría del Caos.

El estudio e interés sobre este tipo de comportamientos fue retomado en los años posteriores por algunos investigadores, por ejemplo, el topólogo Stephen Smale el cual se encontró en una de sus investigaciones con la existencia del caos determinista, misma que era demostrada por su atractor extraño conocido como la herradura de Smale [29]. Pero no sería hasta 1963 cuando el concepto y el estudio del caos tomaron gran relevancia gracias a los estudios y publicaciones del meteorólogo Edward Lorenz, en dicho año, el meteorólogo descubriría el mismo comportamiento que Poincaré, pero esta vez dentro de un análisis numérico y con una perspectiva visual.
Lorenz, trabajaba en un sistema para modelar el movimiento de un fluido bajo un gradiente térmico, dicho sistema era constituido por un conjunto de 3 ecuaciones diferenciales ordinarias. Edward Lorenz, buscaba soluciones a su sistema haciendo uso de una computadora, imprimió los valores y posteriormente quiso confirmar los resultados, pero había un inconveniente, los datos impresos tenían un decimal menos, Lorenz no tomó importancia a esto y colocó las condiciones iniciales tal cual estaban impresas, posteriormente fue a prepararse un café y al volver, se toparía con una sorpresa, los resultados obtenidos en la simulación con las condiciones iniciales "nuevas" habían arrojado un resultado completamente diferente, ese día, Lorenz había descubierto por accidente una de las más importantes características de los sistemas caóticos "la sensibilidad a condiciones iniciales" [30].

Es aquí donde Lorenz, plantea por fin las bases de la teoría del caos moderno y pasa a ser conocido como el padre de la teoría del caos. Su sistema de 3 ecuaciones generaba un atractor caótico muy curioso con forma de mariposa con las alas extendidas, de ahí que a la característica antes mencionada se le conozca también como "el efecto mariposa" (figura 3.1)

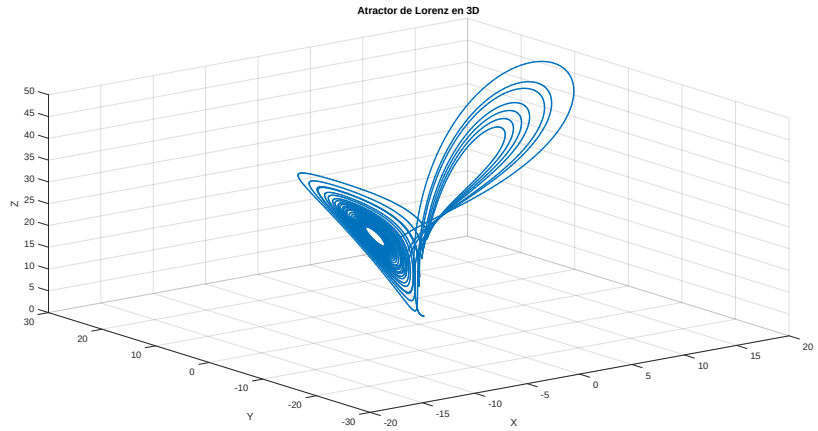


Figura 3.1: Sistema de Lorenz, mapa de fases en 3D.

3.1.2. Características de los sistemas caóticos

Un "sistema dinámico", es un concepto utilizado en física, matemática, y otras disciplinas para referirse a un sistema cuya evolución se ve afectada por el tiempo. En un sistema dinámico, las variables o componentes del sistema cambian a lo largo del tiempo de acuerdo con reglas o ecuaciones específicas. Estos sistemas pueden ser representados matemáticamente a través de ecuaciones diferenciales o ecuaciones en diferencias que describen cómo las variables del sistema se relacionan y cómo evolucionan con el tiempo o con la variable independiente [30].

Los sistemas dinámicos pueden separarse en 3 categorías:

- **Estables:** Los sistemas estables son aquellos que bajo un conjunto de condiciones iniciales, tiende a regresar a un estado de equilibrio o una órbita definida con el tiempo, en dichos sistemas, las pequeñas perturbaciones en las condiciones iniciales no llegan a representar un cambio significativo a largo plazo.

- **Inestables:** Los sistemas inestables son aquellos que bajo determinados cambios en las condiciones iniciales, comienza alejarse de las trayectorias fijadas después de un determinado tiempo. presentado comportamientos oscilatorios divergentes.

- **Caóticos:** Los sistemas caóticos son un tipo especial de sistemas, ya que estos presentan sensibilidades altas a cambios en sus condiciones iniciales y esto los vuelve altamente impredecibles bajo ciertas condiciones.

Los sistemas caóticos, se pueden describir como sistemas definidos por un conjunto de ecuaciones diferenciales o en diferencias el cual es altamente sensible a las condiciones iniciales, son deterministas y presentan pseudoaleatoriedad [31].

Podemos describir las siguientes propiedades de los sistemas caóticos:

1. **Sensibilidad a las condiciones iniciales:** como se ha mencionado anteriormente, la respuesta de este tipo de sistemas puede variar de manera exponencial aun cuando la diferencia entre las condiciones es de muy poco valor.
2. **Aperiodicida:** en este tipo de sistemas, las respuestas obtenidas no siguen una trayectoria o ni tienen comportamientos que se repiten con el tiempo, es decir que no presentan periodos.
3. **Presenta al menos un exponente Lyapunov positivo:** un sistema dinámico de N dimensiones, puede tener N cantidad de exponentes de Lyapunov, si al menos uno de estos exponentes resulta ser positivo, entonces el sistema puede considerarse un sistema caótico.

4. **Atractor extraño con dimensión fractal:** un sistema caótico graficado en un mapa de fases, genera un cuerpo llamado atractor extraño, este tiene una dimensión fraccionaria.

3.2. El mapa de Hénon

El mapa de Hénon fue seleccionado para ser utilizado como generador caótico en este trabajo de tesis, dicho mapa fue introducido por Michael Hénon en 1976 y es un mapa bidimensional el cual captura las dinámicas de estiramiento y plegado característicos de los sistemas hiperbólicos.
Este mapa es altamente estudiado debido a sus características, como la invertibilidad, su capacidad de generar atractores extraños y porque matemáticamente es mucho más sencillo que otros sistemas con el mismo comportamiento.

El mapa de Hénon es un sistema de dos dimensiones que presenta atractores extraños, en dichos atractores, se tiene que las trayectorias quedan confinadas en una región del espacio y cada una de estas se separa de sus vecinas de una manera exponencialmente rápida [32].

El mapa de Hénon se utilizó en su manera simplificada, en la cual una de sus variables se iguala a cero, dejando el sistema como uno dependiendo de dos variables X y Y, y dos parámetros constantes a y b , dicho de esta manera el sistema pasa a ser representado por las siguientes ecuaciones:

$$x_{n+1} = a - x_n^2 + b \cdot y_n$$
$$y_{n+1} = x_n$$

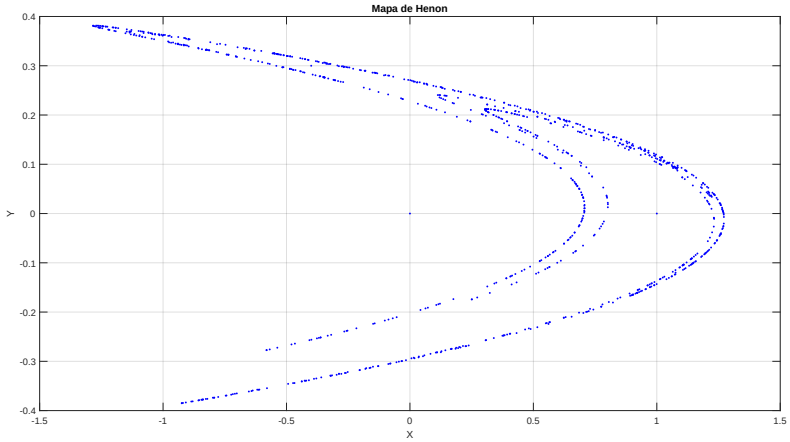


Figura 3.2: Atractor caótico del mapa de Hénon.

3.3. Criptografía

Diariamente compartimos cientos de mensajes y datos a través de internet, tal vez alguna vez nos topamos con una leyenda al abrir una aplicación de mensajería que dice "texto encriptado" o "seguridad con encriptado de extremo a extremo", sea cual sea el caso, es ahí cuando convivimos con la criptografía en la vida cotidiana.

La criptografía, es una disciplina muy antigua, la cual tiene como propósito el proteger información de carácter confidencial, desde sus orígenes ha sido utilizada para protección en entornos políticos, religiosos, militares, entre otros.
En la actualidad, el intercambio de información juega un papel fundamental, y con los grandes progresos de la tecnología, podemos encontrar resultados prometedores en la evolución y la mejora de la seguridad en su transmisión.
Según el número de claves utilizadas, los sistemas criptográficos se clasifican en simétricos y asimétricos, en los sistemas simétricos el emisor y el receptor utilizan la misma clave y en los sistemas asimétricos el emisor y el receptor utilizan claves distintas.[33]

En sus orígenes, la criptografía (Hoy conocida como Criptografía clásica) tenía la tarea de mantener la confidencialidad de los mensajes, pero hoy en día, la criptografía se enfoca principalmente en el concepto de comunicaciones seguras y al mismo tiempo, se busca que cumpla con tres propósitos principales:

1. **Confidencialidad:** es decir, que los mensajes solo sean vistos por aquellos que están autorizados a ver la información.
2. **Autenticación:** se refiere a que la identidad tanto del destinatario y el remitente sean las correctas y que ninguno sea un externo con una credencial falsa.
3. **Integridad:** se busca que el destinatario reciba el mensaje en óptimas condiciones, sin ningún tipo de distorsión o pérdida de datos.

A lo largo de la historia han surgido múltiples métodos de encriptado, a continuación haremos un pequeño recuento de estos métodos y sus características [34].

- **Jeroglíficos:** A pesar de que los jeroglíficos no eran un método de encriptado, se le consideró dentro de esta categoría ya que se tuvo que hacer un largo trabajo criptográfico para poder interpretarlos.
- **Atbash:** Este método de encriptado se menciona en la biblia y se hace énfasis a que era un sistema de sustitución de letras para ocultar los mensajes.
- **La carta de Belerofonte:** En la ilíada, homero cuenta sobre Belerofonte, un héroe mitológico el cual entregó una carta cifrada al rey de Lóbates de Licia

Figura 3.3: Jeroglíficos.

- **Escitala Espartana:** Los espartanos utilizaban un método llamado Escitala, el cual consistía en una tira de piel enrollada en una barra de madera, el mensaje se encripta utilizando el diámetro de esta barra.

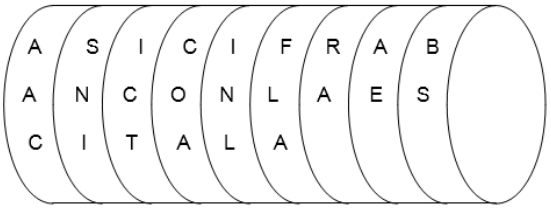


Figura 3.4: Escitala espartana funcionamiento.

- **encriptado de Julio césar:** El encriptado de Júlio César era un método de encriptado en el cual las letras se movían N posiciones hacia la derecha o izquierda, de esta manera los mensajes se ocultaban de los enemigos, según los autores, Julio Cesar utilizaba un desplazamiento de 3 letras.
- **Análisis de Frecuencia:** Al-Kindi mediante el estudio del Corán, logró diseñar un método de encriptado utilizando un análisis de la frecuencia en que se repetían los caracteres en un texto.
- **encriptado de Alberti:** Leon Battista Alberti, diseña un sistema de encriptado polialfabético basado en discos mecánicos, dando origen también a un texto que al día de hoy no se ha podido descifrar.
- **Máquina analitica:** Charles Babbage desarrolla una maquina con la cual comienza a trabajar en el desencriptado de textos con un sistema robusto de encriptado.
- **Enigma:** Alan Turing logró diseñar una máquina capaz de realizar trabajos de análisis de mensajes encriptados y de esta forma logró descifrar mensajes generados por el enemigo con la máquina enigma, un dispositivo capaz de encriptar

textos utilizando un método de confusión, dicho dispositivo utilizaba discos para generar los mensajes encriptados.

Figura 3.5: Máquina enigma.

- **DES y AES:** IBM desarrolló algoritmos de encriptado que se convirtieron en un estándar en los procesos de encriptado.

Los algoritmos de encriptado son una serie de programas que realizan el proceso de criptografía basándose en diferentes tipos de encriptado. A continuación, describiremos los algoritmos de encriptado más utilizados para los procesos de encriptado.

Algoritmos Simétricos:

- **DES (Data Encryption Standard):** Es un algoritmo de encriptado por bloques de 64 bits. Fue ideado por IBM y aceptado por el NIST (National Institute of Standards and Technology) en el año de 1976. Se trata de un algoritmo de 64 bits de clave de los cuales 56 bits componen la clave de encriptado propiamente dicha, mientras los 8 restantes son de sincronización y se usan para detección y corrección de errores.
- **Triple-DES:** El algoritmo Triple-DES es básicamente un método en el cual se aplica tres veces el encriptado DES, de esta manera se corrige la vulnerabilidad del sistema ante las capacidades de cómputo actual. de esta manera obtenemos un sistema robusto con un encriptado de 192 bits.

- **AES (Advanced Encryption Algorithm):** También conocido como Rijndael, es un esquema de encriptado por bloques adoptado como un estándar de encriptado para el gobierno de los Estados Unidos.

Algoritmos Asimétricos:

- **RSA:** Este algoritmo basa su seguridad en la dificultad de los sistemas para factorizar números enteros muy grandes. Los mensajes son representados como un conjunto de números y sus claves son dos números primos elegidos al azar que se mantienen en secreto.
- **Diffle-Hellman:** Este algoritmo es utilizado principalmente para determinar las claves simétricas que serán empleadas para el encriptado dentro de una sesión, se basa en el cálculo de logaritmos discretos.

Además de los algoritmos de encriptado, también existen algoritmos de criptoanálisis, es decir técnicas cuyo objetivo es romper la seguridad de los sistemas y obtener el mensaje sin conocer la clave [35].
A continuación se describen algunas de estas técnicas de criptoanálisis.

- **Ataque de fuerza bruta:** Este método es básicamente utilizar todas las combinaciones posibles en el espacio de claves que se ha utilizado para cifrar el mensaje, no existe una forma de impedir este tipo de ataques, afortunadamente, la capacidad de cómputo actual permite que al utilizar un espacio de claves amplio, el atacante no pueda realizar esta tarea en un tiempo corto.
- **Ataque a texto encriptado:** El atacante solo tiene el texto encriptado para poder realizar el análisis, de esta manera trabaja sobre este para poder descifrar el mensaje utilizando alguna relación entre los caracteres.
- **Ataque a Texto Sin Cifrar Elegido:** El atacante puede tener algo de mensaje que quiere, encriptado con la clave desconocida. La tarea consiste en determinar la clave utilizada para el encriptado. Algunos métodos de encriptado (por ejemplo, el algoritmo de clave pública o asimétrico RSA) son extremadamente vulnerables a los ataques de texto sin cifrar elegido.
- **Ataque MITM:** El atacante intercepta la comunicación entre el emisor y el receptor con la intención de obtener una clave secreta de alguna de las dos partes. de esta forma puede acceder a la conversación y obtener la información que se está compartiendo.
- **Ataque de diccionario:** Este tipo de ataques se caracteriza por no buscar obtener la clave de encriptado, en su lugar, busca capturar el mensaje original sin cifrar. Este tipo de algoritmo parte del robo de claves con las cuales el atacante se puede hacer pasar por el usuario para obtener la información íntegra [36].

podemos darnos cuenta de que el punto clave en la seguridad de los sistemas de encriptado se encuentra en la complejidad que este tiene en su proceso interno, es por ello que se debe tomar en cuenta al momento de su diseño [37].

3.4. Criptografía Caótica

Este es el punto de convergencia entre la criptografía y el caos. Vagamente, se le denomina criptografía caótica a todo sistema que utiliza de manera directa o indirecta los conceptos o métodos de la teoría de los sistemas caóticos.
Uno de los mecanismo de encriptado que aprovecha las dinámicas caóticas es el enmascaramiento del mensaje con una señal generada por una función de respuesta católica, de esta manera se aprovecha un mecanismo de sincronización para que el receptor pueda descifrar el mensaje a su llegada [38].

Los tres principales métodos que utilizan este mecanismo son:

- **Enmascaramiento Aditivo:** En este método, el mensaje se añade a un estado del sistema caótico utilizado, de esta manera el mensaje m se suma a la señal y se crea el criptograma c, para posteriormente ser recibido y desencriptado obteniendo el mensaje mi. La desventaja de este método, es que el mecanismo de sincronización genera distorsiones en el mensaje ya que este se añade a la única señal transmitida, por lo que nuestro mensaje recuperado solo será parecido al mensaje original.

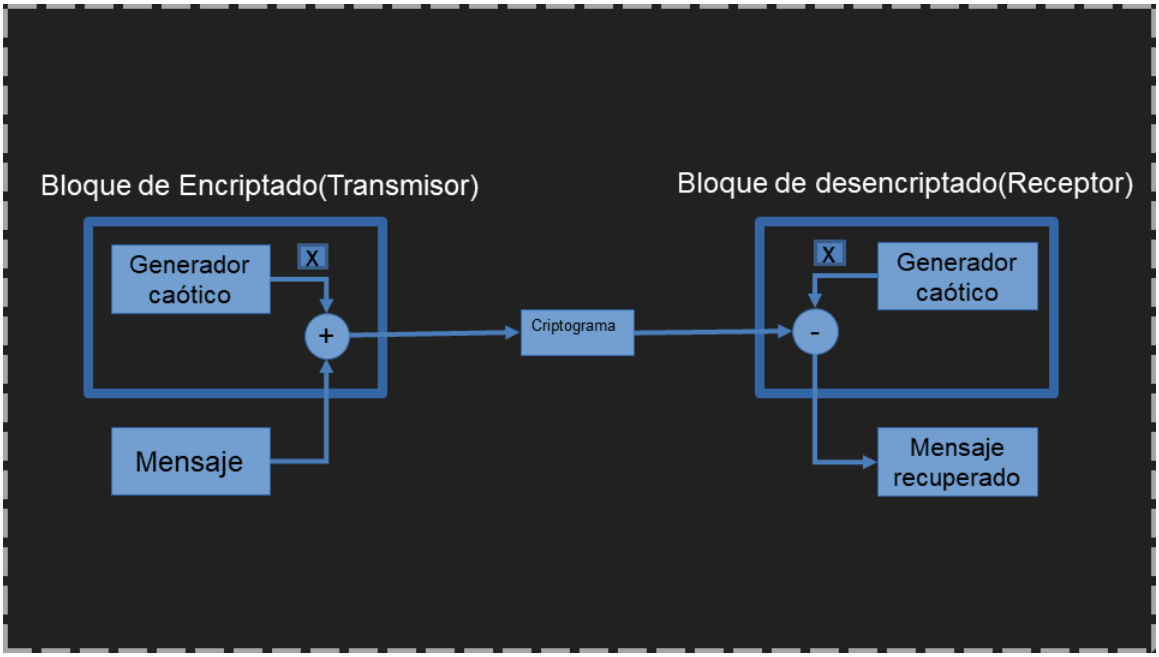


Figura 3.6: Diagrama de bloques de enmascaramiento aditivo.

- **Transmisión bicanal:** Este método es una mejora del anterior, ya que este utiliza dos canales para el trabajo, a través de un canal se envía uno de los estados generados por el sistema caótico el cual se utiliza como señal de sincronía para que los dos sistemas respondan de la misma forma, una vez que estos están sincronizados, a través de un segundo canal se envía otro estado del sistema al cual se le ha añadido el mensaje, de esta manera al ser una señal distinta la que se encarga de la sincronización, se garantiza que el mensaje recuperado será exactamente igual al que se ha enviado.

- **Reinyección de la información:** Este método utiliza los mismos elementos que la transmisión bicanal en el receptor, solo que en esta ocasión, cada elemento del mensaje se utiliza como un factor para el cálculo de la dinamica caotica del sistema transmisor, y la señal enviada será uno que dependa de las condiciones iniciales y del mensaje, aún se encuentra en investigación ya que se requiere de un buen sistema de sincronización para evitar distorsiones y pérdidas de datos [40].

3.5. Conclusiones

Hemos visto los fundamentos básicos sobre el caos y la criptografía, mismos que son necesarios para comprender el por qué los sistemas de criptográficos actualmente están siendo diseñados basados en la aplicación de sistemas caóticos. La criptografía ha estado presente en la vida del ser humano desde tiempos antiguos, siempre buscando cumplir su objetivo de mantener el resguardo de información aunque aplicada a las tecnologías presentes en cada época. La teoría del caos por su parte, ha avanzado mucho en cuestiones de diseño e implementación de sistemas que generan dicha dinámica de comportamiento, a pesar de que algunos avances se hacen sobre sistemas caóticos que ya han sido presentados, estos avances conforman una gran ventaja al momento de diseñar algoritmos que aprovechen sus características, ya sea por la simplificación de un sistema o por la robustez del mismo.
También se vio en este capítulo una breve introducción al mapa de Hénon, el sistema que se ha utilizado para el diseño del trabajo que se presenta, siendo elegido por su facilidad de cálculo en el sistema embebido que se ha elegido para la implementación práctica de algoritmo que ha diseñado.

Capítulo 4

Algoritmo propuesto para encriptado caótico de señales ECG en tiempo real

En este capítulo, se presenta el algoritmo de encriptado propuesto, el cual consiste en utilizar tres generadores caóticos, cada uno con una condición inicial de 16 dígitos decimales, Los generadores están basados en el sistema simplificado del mapa de Hénon. El encriptado será realizado en una estructura de flujo, es decir que se encriptará un dato después del otro en secuencia, el tipo de encriptado será del tipo aditivo o de difusión, es decir, que nuestro mensaje será agregado a la señal de nuestros generadores caóticos.

4.1. Introducción

La señal de electrocardiograma (figura 4.1), es uno de los parámetros fisiológicos más importantes para registrar entre los mamíferos, esto se debe a que se relaciona fácilmente con otras variables determinantes. El estudio de dicha señal en condiciones de estrés, como pueden ser temperaturas elevadas o por debajo de lo normal, el acoso de algún depredador, o alguna situación de riesgo. De esta manera, en la medicina, el estudio y análisis de estas señales puede dar pie a un primer diagnóstico de un paciente por parte del personal médico de emergencias [40].
Dentro de los procesos de la telemedicina, el uso de sistemas embebidos para la captura y procesamiento de datos llega a ser muy común.

La Telecardiología, es un tipo de Telemedicina que, como lo indica su nombre, está orientada al tratamiento de enfermedades del corazón, vasos sanguíneos y el sistema circulatorio en general de manera remota (figura 4.2). Las aplicaciones de este tipo de telemedicina se pueden encontrar en el diagnóstico temprano de los pacientes con problemas cardiacos, en [43] se menciona que los pacientes que presentan un infarto agudo de miocardio, deben recibir atención lo más pronto posible, si esto se realiza, las posi-

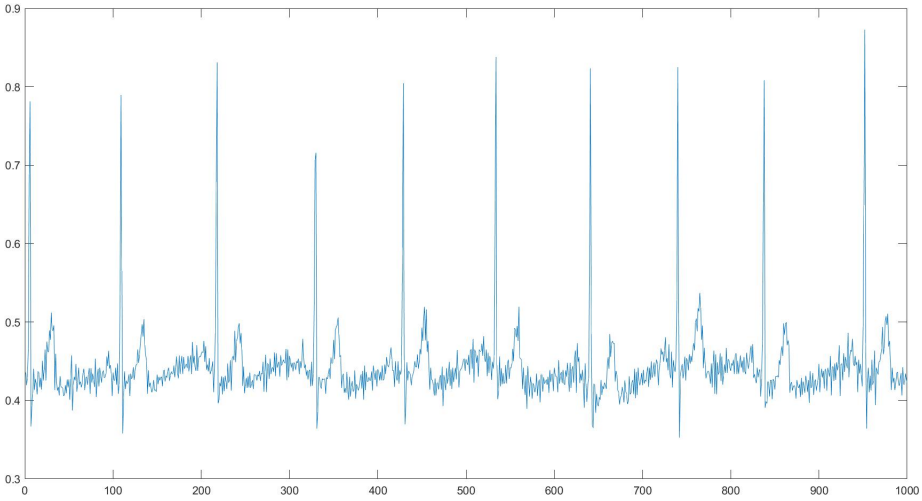


Figura 4.1: Señal de ECG.

bilidades de que la persona sobreviva aumenta, para ello los médicos deben determinar el tipo de afección y cómo tratarla, pero se dificulta sí el paciente se encuentra en un estado delicado y a gran distancia del mismo. Esta es una de las razones por las que la telecardiología existe, para permitir el análisis a distancia de las señales cardiacas y ayudar al diagnóstico temprano.
Otra aplicación de la telecardiología es el deporte, los médicos pueden monitorear a distancia el comportamiento y evolución de los atletas sin necesidad de estar en el lugar de entrenamiento de estos [44].

Figura 4.2: Telecardiología.

La implementación de sistemas de Telecardiología en unidades móviles como ambulancias, ha generado un aumento en el flujo de esta información por medio de canales

inseguros. En la actualidad, se han propuesto múltiples algoritmos de encriptado para bioseñales como el ECG, ejemplo de esto son las presentadas en [44] y [45]. Pero estos algoritmos se basan en el encriptado de señales que han sido alojadas y extraídas de una base de datos.

El algoritmo diseñado para este trabajo cuenta con las siguientes características:

- **Arquitectura de Difusión:** es decir, que el algoritmo cambia el valor de los datos utilizando la señal caótica como un enmascaramiento.
- **Encriptado no convencional:** El algoritmo aprovecha las características de los sistemas caóticos para mayor robustez ante los ataques y mejor seguridad.
- **Estructura de flujo lineal:** El sistema realiza el encriptado procesando un elemento a la vez hasta terminar el conjunto de datos.
- **encriptado simétrico:** ya que para el proceso de encriptado y desencriptado, tanto transmisor como receptor utilizan la misma clave, el sistema se considera un sistema de llave simétrica.

El algoritmo propuesto está basado en el presentado en [46], retirando la sección de confusión y otras características, esto debido a las limitaciones de procesamiento del sistema embebido seleccionado, se añadió también un bloque de captura de la señal, para que esta sea obtenida directamente del paciente.

4.2. Sistema Embebido utilizado

Para el trabajo de captura y encriptado se utiliza un sistema embebido ESP32 (figura 4.3), el cual es un chip Soc (System on Chip) con arquitectura ARM de 32 bits, el cual fue desarrollado por Espressif System Company. Actualmente, la compañía ofrece distintas opciones con respecto a los kits de desarrollo, pero todos comparten algunas características incluso en su modelo más básico, estas son [47]:

1. Dos núcleos Xtensa LX6 con tecnología de 40 nm, los cuales pueden ser controlados de manera independiente.
2. Módulo integrado Wifi 802.11
3. Bluetooth versión 4.2 en doble modo de operación.
4. Pines para conectar pulsadores táctiles
5. Comunicación I2C, I2s, CAN, UART y Ethernet MAC
6. Proporción acelerada de encriptado AES, SHA-2 y RSA

Este sistema fue elegido entre tres opciones debido a la capacidad de comunicación inalámbrica incluida en el mismo, así como su bajo costo. Si bien, existen otros sistemas con capacidades de procesamiento superiores, como El teensy 4.1, este no cuenta con comunicación inalámbrica interna y su precio, aunque bajo en comparación con otros sistemas, es alrededor de 4 veces el valor del chip ESP32 [48].

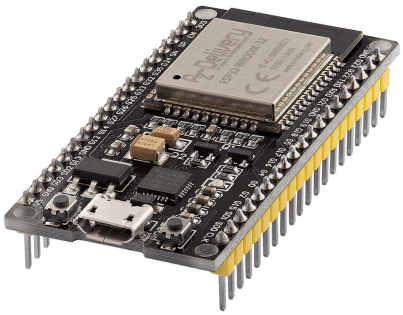


Figura 4.3: Sistema embebido ESP32.

4.3. Lógica del Algoritmo

El algoritmo parte de la captura de una señal de ECG en tiempo real, la cual es captada por un módulo de la marca SparkFun con un circuito integrado AD8232, el cual es un CI para mediciones de Biopotencial, específicamente centrado en la lectura de ECG [49]. La señal se captura con una frecuencia de 100 Hz, por lo que tendremos un vector que se ha grabado a 1 muestra cada 10 milisegundos hasta generar un vector de 1000 datos, el cual será la señal clara para el encriptado. Una vez que se tiene la señal guardada en el vector, se comienza el proceso de iteración de los 3 mapas de Hénon, cada uno con una condición inicial diferente.

Es así entonces que obtenemos los siguientes parámetros:

m=Señal ECG H_1=señal caótica 1 H_2=señal caótica 2 H_3=Señal caótica 3

Una vez que tenemos los parámetros iniciales necesarios para el proceso, podemos continuar al bloque de encriptado, en el cual se realizan algunas operaciones con las 3 señales pseudoaleatorias generadas por los mapas caóticos y posteriormente la señal resultante de estas operaciones se utiliza como máscara para la difusión de la señal de ECG.

En la figura 4.4 podemos ver el diagrama de bloques del algoritmo de encriptado.

Para el proceso de desencriptado, se realiza la operación inversa con respecto a los procesos que se han hecho sobre las señales pseudoaleatorias. El primer paso en este caso es generar las 3 secuencias caóticas, y realizar los procesos necesarios para recrear

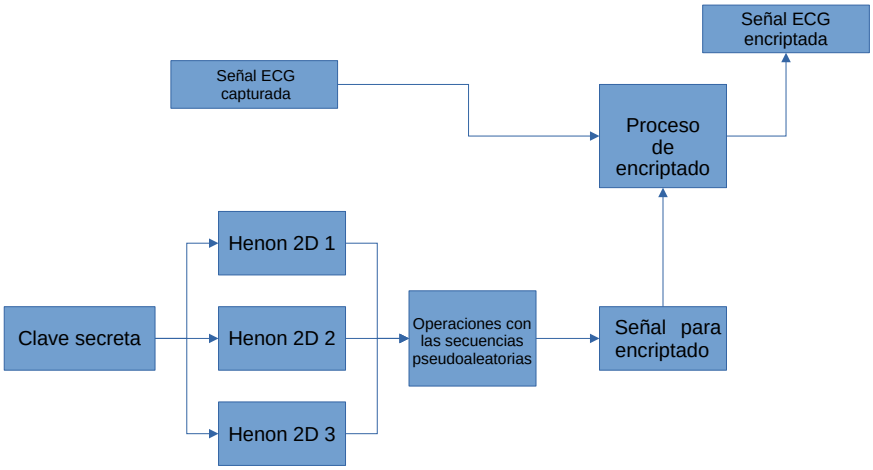


Figura 4.4: Diagrama de bloques de algoritmo de encriptado.

la señal de encriptado, posteriormente, se incluye el criptograma, el cual será procesado para retirar el enmascaramiento colocado en el proceso de encriptado. En la figura 4.5 podemos ver el diagrama de bloques del proceso de desencriptado.

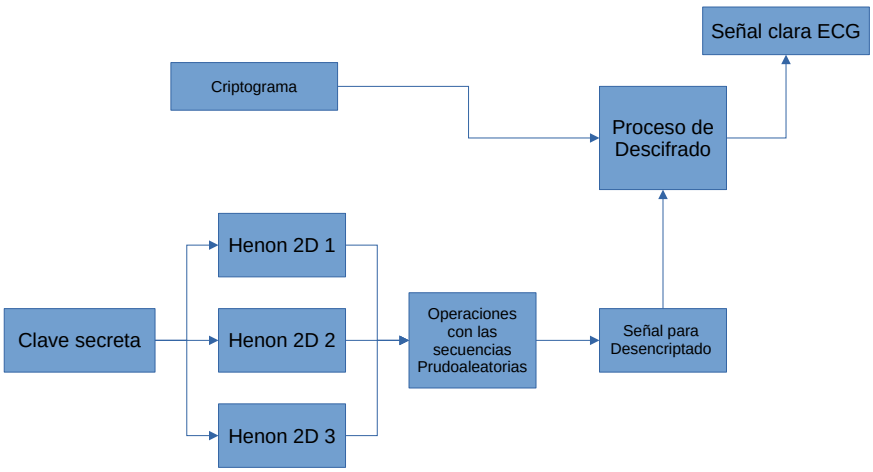


Figura 4.5: Diagrama de bloques de algoritmo de desencriptado.

4.4. Algoritmo propuesto

4.4.1. Captura de señal ECG

Utilizando el módulo con el CI AD8232 capturamos la señal de ECG. Esta se guarda en un vector de 1000 datos, pero antes de que sea guardada, la señal pasa por un proceso de normalización ya que los valores capturados por el módulo el sistema embebido los interpreta como valores entre 0 y 4095, en este caso utilizamos una operación de equivalencias para mapear la señal y reducirla a valores entre 0 y 1.

Para la normalización de la señal utilizamos la siguiente operación

$$m = ECG * 0.0002442002442 \tag{4.1}$$

donde:
m = el vector mensaje normalizado
ECG = la señal capturada con el microcontrolador

4.4.2. Clave secreta

Como se mencionó anteriormente, la clave secreta que se utiliza para el sistema es una clave de tipo simétrica, es decir que tanto el emisor como el receptor utilizan la misma clave para los procesos de encriptado y desencriptado. En este caso, nuestra clave secreta está conformada por un número menor a 0 con una precisión de 16 decimales. por lo tanto $K < 0$, este análisis se aplica a las 3 condiciones iniciales por lo que:

$$(K_1, K_2, K_3 \in R) < 0 \tag{4.2}$$

Símbolo	valor
K_1	0.946789876543156
K_2	0.513469854762535
K_3	0.245687215369874

Tabla 4.1: Valores de las condiciones inicales.

4.4.3. Proceso de encriptado

Partiendo de las condiciones iniciales, los mapas caóticos se iteran para obtener 1000 datos en la secuencia $x^H = x_1^H, x_2^H, x_3^H ... x_L^H$, donde L es la longitud del vector creado, en este caso los 1000 datos. Para mejorar el proceso cada valor de la señal generada se normaliza, es decir se procesa para transformarlo en un valor entre 0 y 1 para esto utilizamos la siguiente expresión:

$$x_{nor}^{H} = \frac{(x_n^H - x_{min}^H)}{(x_{max}^H - x_{min}^H)} \quad para\ n = 1, 2, 3, \ldots, \ell \tag{4.3}$$

donde:
x_{nor}^{H} = el valor del vector normalizado
x_n^H = el valor n del vector que se está normalizando
x_{max}^H= el valor máximo del vector
x_{min}^H= el valor mínimo del vector
L = la longitud del vector

Esto se realiza para los 3 mapas generados por lo que las cadenas pseudoaleatorias $x_1^H, x_2^H, x_3^H \in (0, 1)$.

La señal para encriptado se genera a partir de la siguiente operación:

$$SenC = ((((x_{nor}^{H2} * 2) * (x_{nor}^{H} + x_{nor}^{H3})) + (x_{nor}^{H2} + x_{nor}^{H3}))) \tag{4.4}$$

Para todos los valores de los vectores pseudoaleatorios.
donde:
$SenC$ = la señal para difusión
x_{nor}^{Hn} = los vectores pseudoaleatorios normalizados

En las siguientes figuras podemos observar como los valores se vuelven más uniformes después de realizar la normalización de los datos (figura 4.6 y figura 4.7) esto nos ayuda a aumentar la robustez de nuestro criptograma gracias a que al reducir la repetición de los valores evitamos un ataque de análisis estadístico.

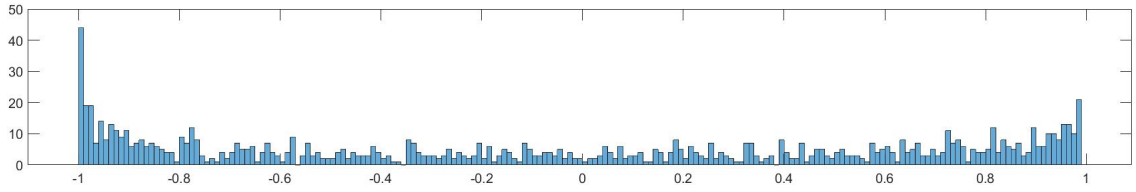


Figura 4.6: Histograma de señal caótica.

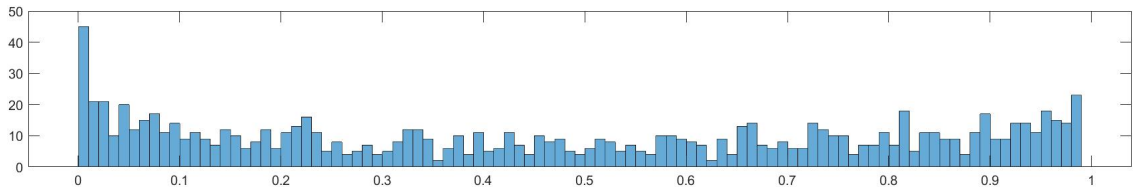


Figura 4.7: Histograma de señal caótica uniformizada.

Una vez que se tiene el vector de enmascaramiento, este se usa para el encriptado utilizando el método de adición y finalmente dividiendo el resultado entre 10, esto para mantener la relación de valores entre 0 y 1. por lo que la operación que modela el

proceso de encriptado es la siguiente:

$$Crip = ((((x_{nor}^{H2} * 2) * (x_{nor}^{H} + x_{nor}^{H3})) + (x_{nor}^{H2} + x_{nor}^{H3}) + m))/10 \tag{4.5}$$

4.4.4. Proceso de desencriptado

Para el proceso de desencriptado debemos invertir los pasos del proceso de encriptado, en este caso, el criptograma es aumentado 10 veces, esto para compensar la reducción que se realizó en el encriptado, ya con el criptograma aumentado, podemos proceder a realizar las operaciones del desencriptado. Se iteran los 3 generadores caóticos y se guardan para generar la señal de desencriptada. Con la señal generada, el proceso de desencriptado se expresa con la siguiente función:

$$ECG = (crip * 10) - (((x_{nor}^{H2} * 2) * (x_{nor}^{H} + x_{nor}^{H3})) + (x_{nor}^{H2} + x_{nor}^{H3})) \tag{4.6}$$

donde:
ECG = la señal de ECG recuperada
$Crip$ = el criptograma recibido
x_{nor}^{Hn}= los vectores pseudoaleatorios normalizados

4.5. Conclusiones

En este capítulo, hemos explicado el algoritmo propuesto, sus características y el proceso que se lleva a cabo de manera interna para poder obtener el criptograma el cual posteriormente es enviado por medio de una conexión inalámbrica entre el emisor y el receptor.

Capítulo 5

Implementación del encriptado caótico en tiempo real con microcontrolador

En este capítulo, se describe la implementación del algoritmo de encriptado en un sistema embebido de 32 bits ESP32. Se presentan los resultados del análisis de seguridad tales como espacio de claves, sensibilidad al cambio en las claves, análisis por histogramas, correlación y entropía de la información. Se demuestra cómo es que el algoritmo de encriptado es implementado de manera eficiente en el sistema embebido.

5.1. Resultados de la implementación

El algoritmo de encriptado en tiempo real ha sido implementado en un sistema embebido de 32 bits, en este caso se utilizó la placa de desarrollo ESP32, la cual a su vez nos permitió utilizar su conexión inalámbrica interna para la transmisión de los datos. Para la programación del algoritmo se usó el lenguaje de programación de Arduino y algunos comandos en lenguaje C, para la compilación y carga del mismo se utilizó la versión 2.1.3 de Arduino IDE. Para el análisis de seguridad se utilizó la plataforma MatLab V9.6 (R2019a) en una laptop con un procesador Ryzen 3 de 2.50 GHz, 12 GB de RAM y sistema operativo Windows 10 de 64 bits; Los valores se representan en variables de punto flotante de tipo single (32 bits) y una precisión de 15 decimales.

Todas las señales de ECG utilizadas durante la etapa experimental del trabajo fueron capturadas en tiempo real de un individuo. Estas se guardaron posteriormente para efectos de análisis y comparaciones. A continuación podemos observar dos ejemplos de las señales capturadas con el sistema (figura 5.1 y 5.2), se puede apreciar la señal normalizada y las ondas que la conforman.

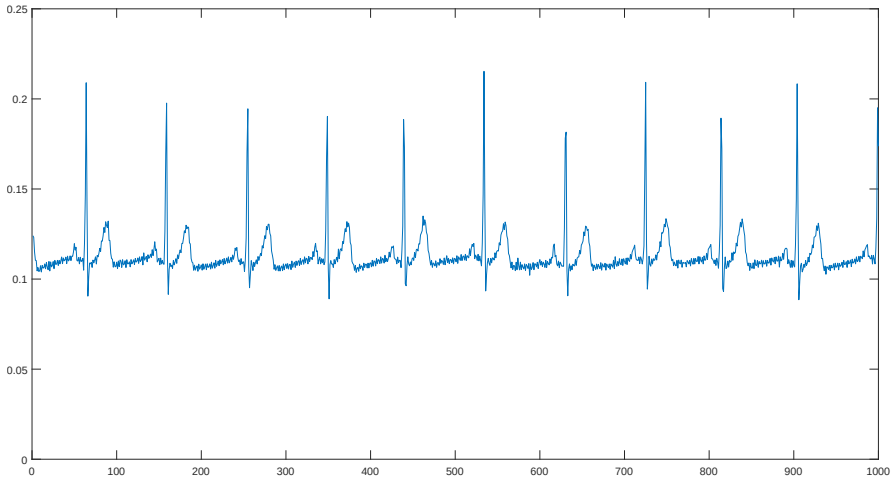


Figura 5.1: Señal ECG1 capturada con módulo AD8232 y ESP32.

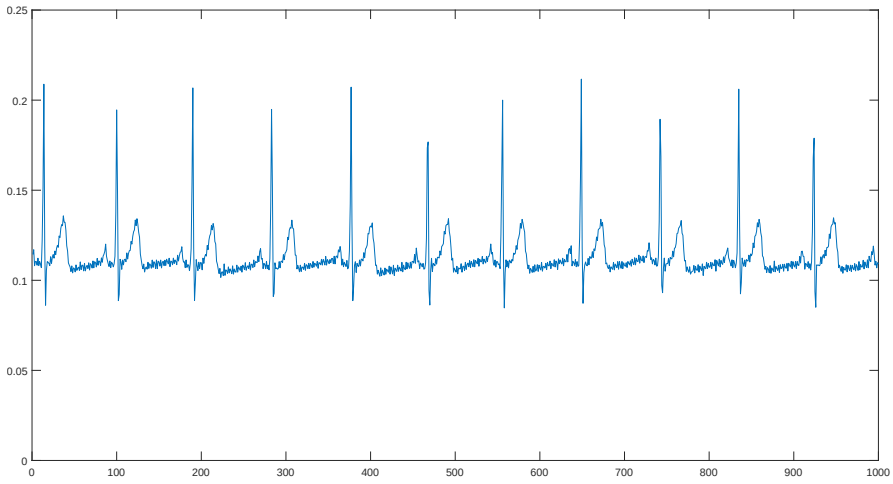


Figura 5.2: Señal ECG2 capturada con módulo AD232 y ESP32.

El proceso de encriptado caótico se realiza de la siguiente manera dentro del sistema embebido.

1. **Clave secreta**

   Dentro del bloque programado, se inicializan variables de tipo double (32 bits en sistema embebido) las cuales se usarán para asignarles el valor de las condiciones iniciales, las cuales serán tomadas como el conjunto de claves secretas a utilizar en la siguiente tabla podemos ver los valores seleccionados para las condiciones iniciales:

Símbolo	valor
K_1	0.946789876543156
K_2	0.513469854762535
K_3	0.245687215369874

Tabla 5.1: Conjunto de claves secretas.

Cada valor del conjunto de claves tienen una precisión de 15 decimales, traduciendo esto a base 2 nos da una un resultado de 2^{50} combinaciones, sumando el espacio de las 3 claves, nos da un total de 2^{150} combinaciones, por lo que el espacio de claves puede considerarse seguro. como se mencionó en el capítulo 4, los valores de los vectores pseudoaleatorios son normalizados utilizando la función:

$$x_{nor}^{H} = \frac{(x_n^H - x_{min}^H)}{(x_{max}^H - x_{min}^H)} \tag{5.1}$$

2. **Normalización de la señal ECG**

Para el proceso de normalizado, se utilizó una operación de equivalencias, sabemos que el valor máximo que nuestro sistema embebido puede leer es 4095, y el valor máximo que nosotros deseamos es de 1, por lo que, para obtener una equivalencia entre rangos, dividimos el valor unitario entre el número de niveles que nuestro microcontrolador nos ofrece, esto con el fin de obtener un valor G, el cual nos servirá para normalizar la señal. la operación sería:

$$G = \frac{1}{4095} = 0.0002442002442 \tag{5.2}$$

$$m = ECG(G) \tag{5.3}$$

donde:
G = el valor para normalizado
m = la señal de ECG normalizada

El proceso consta de multiplicar la señal capturada por el valor normalizador, de esta manera obtenemos un valor equivalente de la señal original en una escala de 0 a 1.

3. **encriptado**

Los mapas de Hénon se iteran hasta llegar a los 1000 datos. Una vez que estos mil datos se obtienen y se han normalizado, se procede a realizar la siguiente operación para obtener la señal de enmascaramiento:

$$SenC = ((((x_{nor}^{H2} * 2) * (x_{nor}^{H} + x_{nor}^{H3})) + (x_{nor}^{H2} + x_{nor}^{H3}))) \tag{5.4}$$

Esta operación se realiza para los 1000 datos de las secuencias.
finalmente, el proceso de encriptado se realiza de la siguiente manera:

$$Crip = ((((x_{nor}^{H2} * 2) * (x_{nor}^{H} + x_{nor}^{H3})) + (x_{nor}^{H2} + x_{nor}^{H3}) + m))/10 \tag{5.5}$$

Para cumplir con estas tareas, el sistema embebido se programa con una serie de ciclos los cuales trabajan bajo la dirección del usuario dependiendo del proceso que desea realizar.
A continuación se muestra un criptograma generado por el sistema y el conjunto de claves secretas utilizadas.

Símbolo	valor
K_1	0.946789876543156
K_2	0.513469854762535
K_3	0.145117696071469

Tabla 5.2: Conjunto de claves secretas utilizadas en ejemplo.

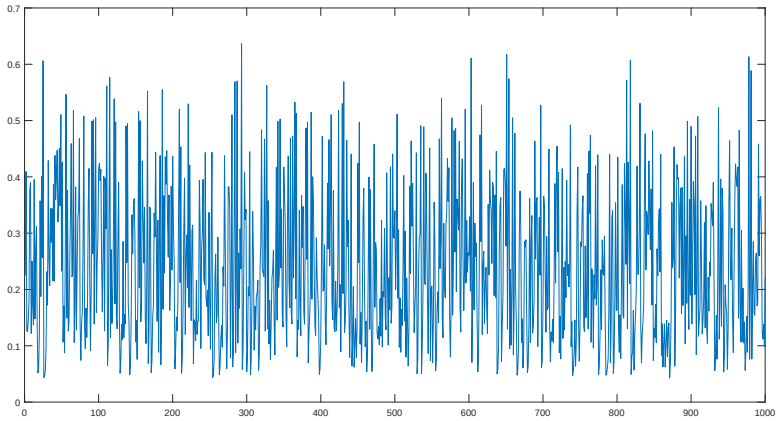


Figura 5.3: Criptograma generado con el sistema para señal ECG1.

4. Transmisión y resguardo

Una vez que el proceso de encriptado ha sido finalizado, el sistema embebido da al usuario 2 opciones: guardar el archivo del criptograma o transmitir el criptograma a un módulo receptor.
Para el resguardo de las señales, se utilizó una memoria micro SD de 8GB, esta se conecta al ESP32 por medio de un módulo de lectura y escritura para tarjetas de memoria. Para esto utilizamos la librería SPI, la cual nos permite establecer comunicación entre el sistema embebido y el módulo de lectura micro SD.
Para transmitir la señal de encriptado, utilizamos la comunicación inalámbrica interna del sistema embebido, más específico, la conexión wifi, pero no de modo que éstos se comuniquen a través de un servidor, ya que los módulos se encuentran a una distancia corta. Para fines experimentales, se utilizó la conexión peer

to peer con la librería ESP NOW, la cual es una librería que nos permite la conexión bidireccional entre sistemas ESP 32 por medio de sus direcciones MAC sin necesidad de tener un servidor o un cliente establecidos [51]. Los valores se transmiten dentro de un vector, el cual es recibido por el Módulo receptor como una lectura en datos individuales, es por esto que dichos valores son nuevamente guardados en un vector, pero en esta ocasión dentro del módulo receptor. El sistema de recepción también ofrece dos opciones al usuario, pero en esta ocasión las opciones son: Guardar y desencriptar. permitiendo al usuario receptor guardar el criptograma para un futuro análisis, o desencriptar y guardar directamente la señal clara.

5. **desencriptado**

   Para el proceso de desencriptado debemos invertir los procesos finales del encriptado. en este caso, el sistema embebido encargado de realizar esta tarea es el receptor, dentro del cual se han establecido las misma condiciones iniciales, y el mismo proceso de iteración y normalización de los vectores pseudoaleatorios para obtener la señal de enmascaramiento, misma que será retirada por medio de una diferencia entre el criptograma y la señal de enmascaramiento.

$$m = (Cripto * 10) - SenC \tag{5.6}$$

En la siguiente figura podemos observar la señal recuprada despues del proceso de desencriptado:

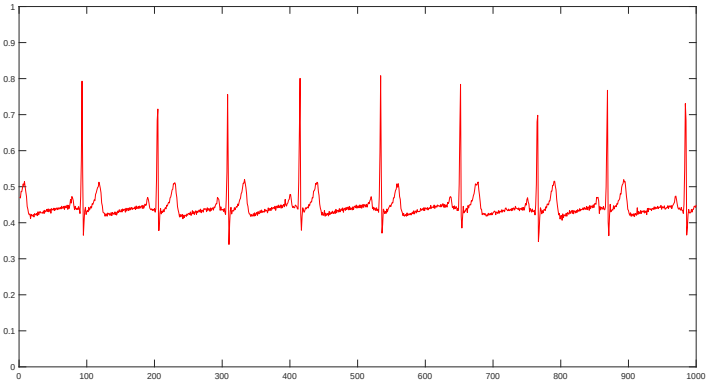


Figura 5.4: Señal de ECG recuperada en el sistema receptor.

5.2. Componentes físicos del sistema

Para los proceso de encriptado y desencriptado, se utilizaron dos módulos de desarrollo ESP32, la información se despliega en una pantalla LCD 16x2 la cual está conectada a la placa de desarrollo por medio de un módulo I2C, de esta manera se

reducen los pines necesarios para su conexión y la comunicación se hace más sencilla. Los datos encriptados se guardan en tarjetas de memoria micro SD de 8GB.
A continuación podemos observar el diagrama de bloques de las conexiones que se han realizado para el sistema las cuales son similares tanto emisor como receptor.

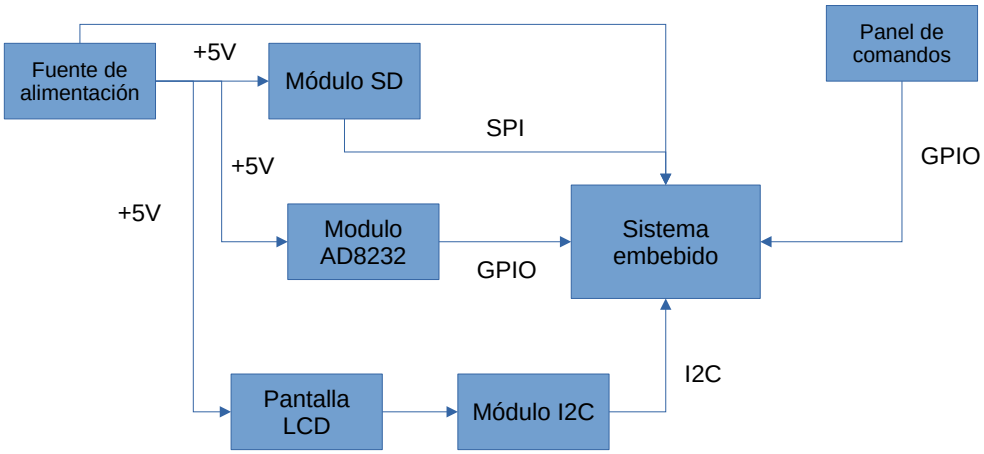


Figura 5.5: Diagrama de bloques de conexiones del sistema embebido.

En el diagrama de bloques de la figura 5.5 podemos observar las secciones que conforman nuestro sistema, las cuales son las siguiente:

- **Módulo SD:** Se trata de una pequeña placa la cual tiene soporte para una tarjeta micro SD, cuenta con un regulador interno por lo que brinda la alimentación de 3.3V necesarios para la tarjeta cuando se conectan 3.3V o 5V de alimentación.
- **Sistema embebido:** En este bloque se ejecutan las tareas del sistema de encriptado, al iniciar dicho sistema se inicializan a su vez los protocolos de comunicación necesarios(I2C, GPIO, SPI en este caso), también se configuran los puertos a utilizar como entradas y salidas del sistema, finalmente el sistema queda en espera de los comando del usuario.
- **LCD 16x2:** En este bloque se despliega un menú principal, en el cual se encuentran las opciones que el sistema le ofrece al usuario, los mensajes desplegados en este bloque dependen de la selección del usuario y de la etapa del proceso que se esté ejecutando.

- **Panel de Control:** El panel de control se compone de 3 botones, dos se encargan de permitir al usuario moverse dentro del menú y el tercero le permite seleccionar la opción que se está indicando en pantalla.
- **Sistema transmisor:** Cuando el sistema se inicia, la pantall LCD se enciende y despliega un mensaje de bienvenida "Sistema de Encriptado" por 5 segundos (figura 5.6), una vez que el tiempo pasa, se muestra el menú principal y se espera el comando del usuario (figura 5.7).

Figura 5.6: Diálogo de incio de sistema.

Figura 5.7: Menú principal desplegado en LCD.

- **Encriptado:** El usuario puede elegir entre las 3 opciones que se le presentan (ver figura 5.7), si el usuario selecciona la primera opción (figura 5.8), el sistema comenzara el proceso de captura de la señal de ECG y posteriormente el proceso de encriptado, una vez que este proceso finalice nos mostrará el mensaje de "proceso finalizado" y regresará al menú principal.

Figura 5.8: Sistema capturando ECG para encriptado.

- **Guardado y transmisión de la señal:** Una vez que se haya capturado y encriptado una muestra, el usuario puede elegir entre guardar el criptograma en la tarjeta o transmitirlo al sistema receptor. Cuando el usuario elige guardar, el sistema toma el vector creado y lo guarda en un archivo de texto dentro de la tarjeta de memoria. Si el usuario decide transmitir la señal, el sistema tomará el vector y lo traducirá en datos en bruto, de esta manera enviará los valores como lecturas uno a uno hasta completar los 1000 datos del criptograma (figura 5.9). al finalizar el proceso se despliega en la pantalla LCD el mensaje "Transmisión finalizada"

Figura 5.9: Sistema generando mapas para encriptado y resguardo de la señal.

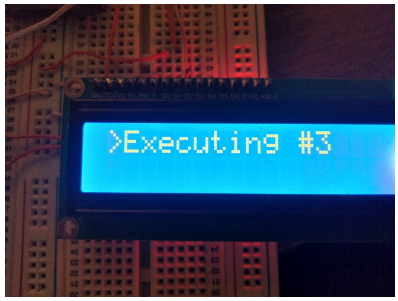


Figura 5.10: Sistema ejecutando la directiva de busqueda del receptor.

- **Sistema Receptor:** El módulo receptor cuenta con la misma estructura del transmisor, solo se descarta el módulo AD8232. Cuando se inicia, en la pantalla LCD se muestra el mensaje "Receptor de encriptado" por 5 segundos, después de que el tiempo pase, se despliega el menú principal del sistema en el cual aparecen 3 opciones, como se puede observar en la figura 5.11

Figura 5.11: Sistema receptor iniciando.

- **Recepción del criptograma:** Para la recepción correcta del criptograma, el módulo debe iniciarse y colocarse en modo de recepción (figura 5.12), (opción 1), posterior a esto, el transmisor debe iniciar el proceso y así el receptor comenzará a guardar los datos en un vector de 1000 datos.

Figura 5.12: Sistema receptor preparado.

- **Desencriptado y guardado:** Una vez que el criptograma fue recibido de manera correcta, el usuario receptor puede elegir entre dos opciones: guardar el criptograma para su futuro análisis, o desencriptar y guardar la señal clara. Cuando el usuario decide guardar el criptograma (figura 5.13), el sistema resguarda los datos en un archivo de texto dentro de su memoria SD. Si el usuario decide desencriptar la señal, entonces el sistema realizará el proceso de desencriptado y guardará únicamente la señal clara en el archivo de texto (figura 5.14).

Figura 5.13: Sistema guardando datos.

Figura 5.14: Sistema desencriptando datos

5.3. Análisis de seguridad

Los sistemas embebidos (como el utilizado en este trabajo de tesis) no están exentos de vulnerabilidades físicas, ya sea la extracción de datos o un ataque a la estructura del mismo, es por eso que debemos garantizar que al menos los datos que se están generando cumplan con algunos requerimientos de seguridad. A continuación se presentan los resultados de las pruebas de seguridad a las que se ha sometido al sistema.

5.3.1. Espacio de claves

Para que nuestro sistema sea robusto ante ataques de fuerza bruta, es decir, que se utilice un algoritmo para probar todas las combinaciones posibles de claves para llegar a la señal clara. Este debe tener un espacio de claves de al menos 2^100 posibles combinaciones, en el caso de nuestro sistema, este utiliza un conjunto de claves las cuales dan como resultado un total de 2^{150} posibles soluciones, por lo que solo con el total de combinaciones es posible concluir que este es lo suficientemente robusto como para soportar un ataque de este tipo.

5.3.2. Sensibilidad a las condiciones iniciales

Se debe comprobar el que nuestro sistema sea altamente sensible a las condiciones iniciales, esto para que los criptogramas generados no sean parecidos entre sí y eso pueda dar lugar a la posibilidad de un ataque por análisis estadístico. Para determinar esto, el algoritmo debe ser inicializado dos veces con dos claves secretas muy similares, el resultado de estos procesos deben ser dos criptogramas completamente diferentes y con nula correlación entre ellos.

El proceso se realiza y el resultado del análisis de correlación debe ser lo más cercano a 0 posible. Las claves secretas utilizadas en este caso, tienen un valor decimal de diferencia y con esto obtuvimos un valor de correlación de 0.050762392397523 , este resultado se obtuvo modificando solamente las condiciones iniciales, sin modificar los parámetros de control del sistema.

Con el resultado obtenido podemos concluir que la correlación es nula y que nuestro algoritmo diseñado presenta una alta sensibilidad a las condiciones iniciales.

Símbolo	conjunto 1	conjunto 2
K_1	0.946789876543156	0.9467898765431**6**6
K_2	0.513469854762535	0.5134698547625**3**4
K_3	0.245687215369874	0.24568721536**9**974

Tabla 5.3: Conjunto de condiciones inciales con diferencias mínimas.

5.3.3. Histogramas

Los histogramas son una representación gráfica de una variable en forma de barras, la longitud de cada barra nos muestra la frecuencia de los valores representados. Para que el sistema criptográfico no sea susceptible a ataques utilizando histogramas, este debe tener una imagen de histograma uniforme, lo que nos garantiza que la información generada es impredecible.
A continuación podemos observar los histogramas generados por la señal ECG (figura 5.15) y el criptograma (figura 5.16). Podemos observar que el histograma de la señal clara presenta una notable alza en ciertos valores, mientras que el histograma de nuestro criptograma presenta uniformidad en la mayoría de los valores, dejando así una clara diferencia entre las señales.

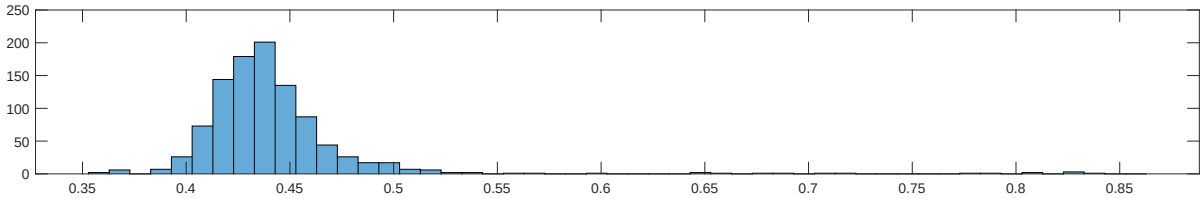


Figura 5.15: Histograma de señal clara.

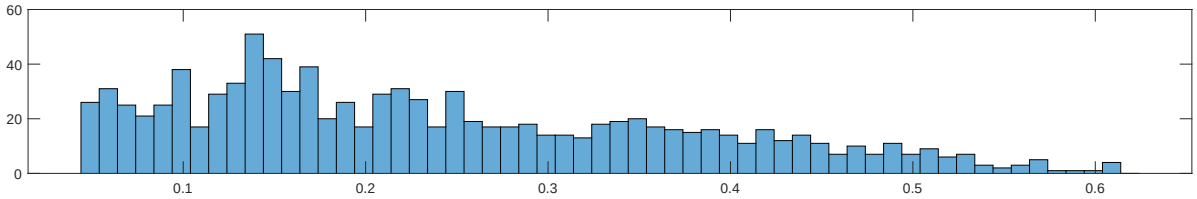


Figura 5.16: Histograma de señal encriptada.

5.3.4. Correlación

Otro análisis de correlación que debemos realizar es entre la señal clara y el criptograma. Es necesario confirmar que la señal clara es completamente diferente a la señal encriptada. el valor de correlación debe ser lo más cercano a 0 posible.

la correlación se calcula de la siguiente manera:

$$Cr = \frac{N \times \sum_{i=0}^{N}(x_i \times y_i) - \sum_{i=0}^{N} x_i \times \sum_{i=0}^{N} y_i}{\sqrt{\left(N \times \sum_{i=0}^{N}(x_i)^2 - \left(\sum_{i=0}^{n} x_i\right)^2\right) \times \left(N \times \sum_{i=0}^{N}(y_i)^2 - \left(\sum_{i=0}^{n} y_i\right)^2\right)}} \tag{5.7}$$

siendo $Cr \in (-1, 1)$ el coeficiente de correlación donde $Cr = 0$ significa correlación nula.

El resultado de la correlación entre nuestra señal clara y el criptograma es de 0.012118565930789 por lo que se concluye que la correlación es nula y el sistema puede ocultar la información de manera correcta.

5.3.5. Entropía de la información

La entropía hace referencia a que tan impredecible es la información que presentamos, de este modo, debemos asegurar que la información que generamos en el criptograma presente un valor alto de entropía, dicho valor se calcula de la siguiente manera:

$$H(m) = \sum_{i=0}^{2^N-1} p(m_i)\ log_2(1/p(m_i)), \tag{5.8}$$

En el análisis de entropía, la señal se transforma a valores de 8 bits, es decir que pasa de ser valores entre 0 y uno, a valores de $0 - 255$. de esta forma, el valor máximo de entropía que podemos obtener es de 8.
El valor de entropía obtenido es de 7.820566422151759, el resultado es muy cercano a 8, por lo que podemos decir que nuestro sistema puede generar un alto desorden, lo que nos garantiza resistencia a un ataque de entropía.

5.4. Conclusiones

A lo largo de este capítulo hemos presentado la implementación práctica en un sistema embebido de nuestro algoritmo que se explicó en el Capítulo 4 para el encriptado de señales ECG. Mostramos el funcionamiento del mismo y se expusieron los resultados del análisis de seguridad. dichos análisis se realizaron con ayuda de MatLab, se utilizaron los datos registrados con el sistema en las tarjetas SD para comparar resultados.
Los resultados de dichos análisis nos permiten concluir que el sistema se implementó de manera correcta y que este es seguro para su uso en un sistema de telemedicina.

Capítulo 6

Conclusiones

6.1. Conclusiones generales

En este trabajo de tesis, se realizó el diseño e implementación de un sistema de encriptado y transmisión de electrocardiogramas (ECG) en tiempo real, utilizando como base un sistema embebido de 32 bits. Dicho sistema se basa en la iteración del mapa de Hénon como señal de enmascaramiento. Se utiliza la conexión WIFI interna del sistema embebido para la transmisión de las señales con el protocolo peer to peer.
Durante la etapa de investigación, se analizaron el mapa logístico, el mapa de Hénon y el sistema simplificado de Lorenz, debido a las limitaciones de procesamiento del sistema embebido, se optó por utilizar una versión simplificada del mapa de Hénon ya que esta nos permite que el tiempo de cálculo fuera más rápido.

El mapa de Hénon demostró ser eficiente para el proceso, los cálculos se realizaron partiendo de un conjunto de condiciones iniciales con valor decimal entre 0 y 1, estos cuentan con una precisión de 15 decimales, esto para garantizar una mayos sensibilidad en el cálculo del mapa.
El sistema se probó en una transmisión a corta distancia de una señal de ECG, esta se realizó con dos sistemas ESP32 en conjunto para cumplir los roles de emisor y receptor.

Obtuvimos resultados favorables en las pruebas de seguridad, la prueba de histograma mostró que los datos generados eran uniformes por lo que al estar distribuidos es poco probable que se pueda identificar el tipo de señal enviada sólo analizando el histograma de la señal cifrada. Con el análisis de correlación (tanto de señal clara y criptograma como el de dos criptogramas) arrojaron datos satisfactorios, ya que comprobamos que las señales son completamente distintas entre sí, lo que nos garantiza que el sistema es robusto ante ataques de análisis estadístico.

6.2. Trabajo a futuro

Se proponen los siguientes puntos para realizar mejoras a futuro:

- Mejorar el sistema para que sea capaz de transmitir los datos conforme se realiza el encriptado para transformarlo en un sistema de monitoreo en tiempo real.
- Añadir soporte para encriptar y transmitir otros tipos de señales fisiológicas importantes para un diagnóstico.
- Transmitir los datos por medio de internet a larga distancia, ya que el sistema en su estado actual solo permite la transmisión WIFI entre dos módulos a una distancia muy pequeña.

Bibliografía

[1] Ena, J. (2020). Telemedicina aplicada a COVID-19. *2020 Rev. Clin. Española*, **2020**: (501–502)

[2] Fernandez J., Miriam & Hernandez M. (2010). Telemedicina: futuro o presente. *Rev haban cienc méd Edción Digítal* **2010**(9): (10-27)

[3] Vázquez, J. (2021). Telemedicina durante la pandemia por COVID-19. *NCT Neumol. Cirugia Torax*, vol. 80, n.º 2 **2021** (132–140)

[4] Bingyi, H.,Jing, B. & Datian, Y. (1997). Magnificent Milestones and Emerging Opportunities in Medical Engineering. Proceedings of the *19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* **1997**(3): (981-982)

[5] Nader, K. (2019). Riesgos en la seguridad informática en salud - 4 casos resonantes. Memorias congresos departamento de medicina de la universidad de pamplona. viii congreso de medicina retos en salud.

[6] Millérioux,G., Hernandez, A. & Amigo, J.M. (2005). Criptografía caótica con reinyección de la información. *III Congreso Iberoamericano de Seguridad Informática* **2005**(207-220)

[7] Soriano-Torres, O. & Lugo, L.M. (2019). Telemedicine: future or present?. *Revista Cubana Habanera de Ciencias Médicas* **2010**(9): (127-139)

[8] Parrasi-Castaño, Y., Celis-Carvajal, L., Bocanegra-García, J.J. & Pascuas-Rengifo, Y.S. (2016). Estado actual de la telemedicina: una revisión de literatura. *Ingeniare Digital Journal* **2016** (20)(105-120)

[9] Cáceres-Méndez, A., Castro-Díaz, M., Gómez-Restrepo, C. & Puyuna, J.C. (2011). Telemedicina: historia, aplicaciones y nuevas herramientas en el aprendizaje Universitas Médica. Pontificia Universidad Javeriana **2011**(52): (11-35)

[10] Sanjay, S., Mbarika, V., Shakhina, J., Dookhy, R., Doarn, C., Nupur, P. & Ronald, .C (2014). Brief Communication What Is Telemedicine? A Collection of 104 Peer-Reviewed Perspectives and Theoretical Underpinnings.

[11] Jena, E. (2020). Telemedicine For COVID-19. *Revista Clínica Española* **2020**(220): (501-502)

[12] Contreras, P. & Yanelis, M. (2021). Artículo de revisión: Telemedicina, una nueva herramienta para la atención en el servicio de Fonoaudiología. Universidad de Pamplona-Facultad de Salud **2021**

[13] Ramiro. F & Vaca-Narvaja, M.D. (2021). Historia de la Telemedicina. *Revista de la asociación iberoamericana de telesalud y telemedicina* **2021**(8): (7-11)

[14] Sampedro-Hernandez, J.L. (2013). Innovación y cambio microinstitucional en el sector salud: evidencia de la telemedicina en México. *Economía: Teoría y Práctica.* **2013**(39): (31-57)

[15] Nmas (2023). Fallece el Dr. Adrián Carbajal Ramos; Pionero en Cirugía Robótica en México. *N+ Edción digítal* **2023**

[16] FDA Clearance (2017). Da Vinci X robot 10 years of performance. *FDA Clearance On Site* **2017**

[17] Eveleth, R. (2014). Los cirujanos que operan a cientos de kilómetros de distancia. *BBC News Mundo Edición digítal* **2014**

[18] Drew Redacción interna (2020). Salud digital: Principales avances y aplicaciones de la telemedicina. *Drew — Business Insights* **2020**

[23] Rabanales-Sotos, J., Párraga-Martínez, I., López-Torres, J., Andres-Pretel, F. & Navarro-Bravo, B. (2011). Tecnologías de la Información y las Telecomunicaciones: Telemedicina. *Revista Clínica Médica Familiar* **2011**(40): (504-518)

[20] Wootton, R. (2001). Recent advances: Telemedicine. *BMJ* **2001**(7312): (557-560)

[21] Salazar, J. (2012). Telemedicina — gestion sanitaria. *gestion sanitaria Andaluz* **2012**

[22] Carabaño-Aguado, I. (2015). Servicios no presenciales: nuevas luces en el quehacer de siempre. *Revista Atención Primaria* **2015**(17): (299-300)

[23] Garay-Fernández, J.D. & Gómez-Restrepo, C. (2011). Telepsiquiatría: innovación de la atención en salud mental. Una perspectiva general. *Revista Colombiana de Psiquiatría* **2011**(40):(504-518)

[24] Santa-Vélez, C., Acosta-Madiedo, A., Pérez-Madrid, C., Galeano-Piedrahita, E., Morales-Restrepo, D. & Laasch-Restrepo, M. (2020). Estado del arte de la teledermatología. *CES de Medicina* **2020**(34): (198-206)

[25] Cardier, M., Manrique, R., Huarte, A., Valencia, M. L., Borro, D., Calavia, D., & Manrique, M. (2016). Telemedicina. estado actual y perspectivas futuras en audiología y otología. *Revista Médica Clínica Las Condes* **2016**(27): (840-847)

[26] Bergal, J. (2022). El robo de datos a hospitales pone en riesgo a los pacientes. *Chicago Tribune, Stateline Organization* (2022)

[27] González, M.J. (2009). La teoría del caos en las organizaciones. *Cuadernos unimetanos, Repositorio Institucional*. **2009**(18): (29-33)

[28] Poincaré, H. (1890). Sur le problème des trois corps et les équations de la dynamique. *Acta Mathematica* **1890**(13): (1-270)

[29] Sametband, M. (1999). Entre el orden y el caos. La complejidad (segunda edición). *Fondo de cultura económica* **1999**

[30] Madrid, C. (2010). Historia de la teoría del caos contada para escépticos. *Cuestiones de génesis y estructura. Encuentros Multidisiplinarios* **2010** Art. 3

[31] Escalante, R. (2016). Generación de sistemas dínamicos lienales por partes caóticos sin puntos de equilibrio (Publicación n.º 1173). Tesis de Maestira, *Instituto Potosino de Investigación Científica y Tecnología* **2016**

[32] Cortés, O. (2005). Predictibilidad en sistemas caóticos:el mapa de Henon (Publicación n.º 4117). Trabajo de Grado, UniAndes. *UniAndes Repositorio Institucional*. **2005**

[33] Moreno, J.P., Parra, F., Huérfano, R., Suarez, C., & Amaya, I. (2016). Modelo de Encriptacíon Simétrica Basada en Atractores Caíticos. *Revista de Ingeniería*, **2016** (21) (378-390)

[34] Velasco, J.J. (2014). Breve historia de la criptografía. *Redacción Interna elDiario Versión digítal* **2016**

[35] Sanjuan, L. (2016). Seguridad en desarrollo del Software. *DSpace Principal* **2016**

[36] Marrero-Travieso, Y. (2003). La Criptografía como elemento de la seguridad informática. *ACIMED, Formato digital*. **2003**

[37] Pushpalatha, G. S., & Ramesh, S. (2021). Chaotic based encryption algorithms for speech signal and cryptographic requirements: A brief survey. *Materials Today: Proceedings*. **2021**

[38] Cuerpo Académico: Sistemas Dinámicos y Criptografía. (2019). Avances en circuitos y sistemas. *Universidad Autónoma de San Luis Potosí*. **2019**

[39] Murillo-Escobar, M.A. (2016). Cifrado caótico simétrico de ECG y EEG para aplicaciones en telemedicina. *XVII CLCA Latin American Conference of Automatic Control* **2016** No.612617.

[40] Fernando, G. (2007). Encriptación y desencriptación de datos usando técnicas caóticas. *UACC Repositorío Institucional, Tesis doctoral*. **2007**

[41] Santos-Betacourt, A., Bistecel-Esquivel, R.A, & Mora-Macias, E.C. (2016). Diseño y análisis de tres canales de acondicionamiento de la señal de ECG para aplicaciones de Neuroetología. *Ingeniería Electrónica, Automática y Comunicaciones* **2016**(37): (47-65).

[42] Grima-Serrano, A., García-Porrero, E., Luengo-Fernández, E., & León-Latre, M. (2011). Cardiología preventiva y rehabilitación cardiaca. *Revista Española de Cardiología* **2011**(64): (66–72)

[43] Sorensen, J.T, Clemmensen, P., & Sejersten, M. (2013). Telecardiología: Pasado, presente y futuro. *Revista Española de Cardiología* **2013**(66): (212–218)

[44] Piedro-Roberto, M. (2003) Cardioloía del deporte. *Revista argentina de Cardiología*. **2003**(71): (126-137)

[45] Qintana-Ibarra, J.A. (2018). Cifrado caótico de señales electrofisiológicas basado en mapa de ushio para telemetría segura. *UABC, Repositorio Institucional*. **2018**

[46] Barbará-Morales, E., Rodríguez, O. & Alba, E. (2012). Algoritmo de codificación de señales electrocardiográficas mediante el modelo caótico de Lorenz. *16 Convención Científica de Ingeniería y Arquitectura*. **2012**

[47] Murillo-Escobar, M.A., Cardoza-Avedano, L., López-Gutiérrez, R.M. & Cruz-Hernández, C.(2017). A double chaotic layer encryption algorithm for clinical signals in telemedicine. *Journal of medical systems* **2017**(41): (1-17)

[48] Babiuch, M., Foltỳnek, P. & Smutnỳ, P. (2019). Using the ESP32 microcontroller for data processing. *20th International Carpathian Control Conference (ICCC)* **2019**(1-6)

[49] Martínez, A., Gonzales, C., Jaramillo, A., Cardenas, D. & Von Chong, A. (2022). Low-cost, microcontroller-based phase shift measurement system for a wireless power transfer prototype. *HardwareX Digital Journal*. **2022**(11)

[50] Mendez-Alves, J.J.,Campos-Prado, D., Coelho, L. & Krueger, E. (2023). AD8232 to Biopotentials Sensors: Open Source Project and Benchmark. *Electronics Digital Journal* **2023**(4): (833)

[51] Murillo-Escobar, M.A., Cruz-Hernández, C., Abundiz-Perez, F. & López-Gutiérrez, R.M. (2014). Cifrado caótico de plantilla de huella dactilar en sistemas biométricos. *Congreso Latinoamericano de Control Automático* **2014** (18-23)

[52] Castillon-Meranza, M.O. (2020). Cifrado caótico en sistemas embebidos y evaluación de seguridad criptográfica. *Centro de Investigación Científica y de Educación Superior de Ensenada* **2020**

[53] Pasic, R., Kuzmanov, I. & Atanasovski, K.(2021). ESP-NOW communication protocol with ESP32. *Journal of Universal Excellence*, **2021**(6): (53-60)

Apéndice A

Programa para módulo transmisor

En este apéndice se presenta el código en lenguaje C con el que se programó la placa ESP32 correspondiente al módulo transmisor para la captura y cifrado de la señal de electrocardiograma así como la transmisión del criptograma al módulo receptor.

```
/* Programa del módulo Transmisor de encriptado
caótico basado en caos con captura de señal ECG en tiempo real.
Alvaro Rodriguez Hernandez
Ingeniería en Electrónica
Universidad Autónoma de Baja California
Noviembre de 2023
*/

//librerias necesarias para el programa
#include <SD.h>
#include <SPI.h>
#include <Wire.h>
#include <esp_now.h>
#include <WiFi.h>
#include <LiquidCrystal_I2C.h>

#define SS 5

// configuración para envio de datos
//definición de la dirección MAC del receptor
//
    ------------------------------------------------------------------------------------

uint8_t RxMACaddress[] = {0xB0, 0xA7, 0x32, 0x2A, 0x78, 0xA0};
//definición de la estructura del dato a enviar
typedef struct TxStruct
{
  double CriptoTx;
}TxStruct;
TxStruct sentData;
//
    ------------------------------------------------------------------------------------

void OnDataSent(const uint8_t *mac_addr, esp_now_send_status_t status) //callback
    function
{
```

```
32   Serial.print("\r\nLast Packet Send Status:\t");
33   Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Delivery Success" : "Delivery
         Fail");
34 }
35 esp_now_peer_info_t peerInfo;
36 //
     ==========================================================================================

37
38 int sel = 0;//variable de seleeción para los botones
39
40 File archivo;//creación del objeto archivo para la tarjeta SD
41 //Numero de Muestras
42
43 LiquidCrystal_I2C lcd(0x27, 16, 2);
44
45 int upButton = 13 ;
46 int downButton = 12;
47 int selectButton = 14;
48 int menu = 1;
49
50 const int numMuestras = 1000;
51 // VARIABLES LOGISTICO:
52 double aa=3.999999999779998;
53
54 double CI_XX = 0.946789876543156;
55 double CI_XX1 = 0.513469854762535;
56 double CI_XX2 = 0.245687215369874;
57 double VAL_XX=0.0;
58 double VAL_ANT_XX=0.0;
59
60 double VAL_XX1=0.0;
61 double VAL_ANT_XX1=0.0;
62
63 double VAL_XX2=0.0;
64 double VAL_ANT_XX2=0.0;
65
66 double suma;
67 double y;
68 //inicialización de los vectores
69 double Log1D[numMuestras];
70 double Log1D2[numMuestras];
71 double Log1D3[numMuestras];
72 double Cripto[numMuestras];
73 double ECG[numMuestras];
74
75 void setup() {
76   Serial.begin(115200);
77   lcd.init();
78   lcd.backlight();
79
80   lcd.print("init SD card");
81     Serial.println("inicializando tarjeta...");
82   if(!SD.begin(SS)){
83     lcd.clear();
84     Serial.println("fallo en inicializacion");
85         lcd.clear();
86     lcd.print("fallo init SD");
87   }
```

```
88    else{
89      archivo = SD.open("/Lectura.txt",FILE_WRITE);
90      Serial.println("inicializacion correcta");
91      lcd.clear();
92      lcd.print("init SD correcta");
93    }
94
95  //inicialización de parametros para transferencia de datos
96    WiFi.mode(WIFI_STA);
97    //
          ------------------------------------------------------------------------------------------

98    if(esp_now_init() != ESP_OK)
99    {
100     Serial.println("Error initializing ESP-NOW");
101     return;
102   }
103   //
          ------------------------------------------------------------------------------------------

104   esp_now_register_send_cb(OnDataSent);
105   //
          ------------------------------------------------------------------------------------------

106   //esp_now_peer_info_t peerInfo;
107   memcpy(peerInfo.peer_addr, RxMACaddress, 6);
108   peerInfo.channel = 0;
109   peerInfo.encrypt = false;
110   //
          ------------------------------------------------------------------------------------------

111   if(esp_now_add_peer(&peerInfo) != ESP_OK)
112   {
113     Serial.println("Failed to add peer");
114     return;
115   }
116
117   lcd.setCursor(0, 0);
118   lcd.print(" Transmisor de ");
119   lcd.setCursor(0,1);
120   lcd.print(" Encriptado Caos");
121   delay(5000);
122   lcd.clear();
123   pinMode(upButton, INPUT_PULLUP);
124   pinMode(downButton, INPUT_PULLUP);
125   pinMode(selectButton, INPUT_PULLUP);
126   updateMenu();
127 }
128
129 void loop() {
130   if (!digitalRead(downButton)){
131     menu++;
132     updateMenu();
133     delay(100);
134     while (!digitalRead(downButton));
135   }
136   if (!digitalRead(upButton)){
137     menu--;
138     updateMenu();
```

```
    delay(100);
    while(!digitalRead(upButton));
  }
  if (!digitalRead(selectButton)){
    executeAction();
    updateMenu();
    delay(100);
    while (!digitalRead(selectButton));
  }
}

void updateMenu() {
  switch (menu) {
    case 0:
      menu = 1;
      break;
    case 1:
      lcd.clear();
      lcd.print(">Encriptar");
      lcd.setCursor(0, 1);
      lcd.print(" Guardar");
      break;
    case 2:
      lcd.clear();
      lcd.print(" Encriptar");
      lcd.setCursor(0, 1);
      lcd.print(">Guardar");
      break;
    case 3:
      lcd.clear();
      lcd.print(">Transmitir");
      lcd.setCursor(0, 1);
      lcd.print(" MenuItem4");
      break;
    case 4:
      lcd.clear();
      lcd.print(">Transmitir");
      lcd.setCursor(0, 1);
      lcd.print(">MenuItem4");
      break;
    case 5:
      menu = 4;
      break;
  }
}

void executeAction() {
  switch (menu) {
    case 1:
      lcd.clear();
lcd.print("Iniciando...");
      action1();
      break;
    case 2:
      lcd.clear();
lcd.print("Init. resguardo");
      action2();
      break;
    case 3:
```

```
198       lcd.clear();
199 lcd.print("Buscando Rx");
200       action3();
201       break;
202     case 4:
203       action4();
204       break;
205   }
206 }
207
208 void action1() {
209   lcd.clear();
210 lcd.print("Iniciando...");
211 Serial.println("iniciando proceso de captura y encriptado");
212 delay(5000);
213 lcd.clear();
214 lcd.print("Capturando ECG");
215 Serial.println("capturando ECG");
216 //Captura de la señal ECG
217  for (int i = 0; i < numMuestras; i++) {
218     float lec=((analogRead(A0))*0.0002442002442);
219     ECG[i] = lec,15;
220     delay(10); //
221   }
222 //
223 lcd.clear();
224 lcd.print("ECG Guardado");
225
226   Serial.println(" ");
227   Serial.print("ECG guardado");
228   Serial.println(" ");
229   Serial.println(" ");
230
231 /////
232 //Calculo del parametro y el cual se utiliza como condición inicial para el 3er
      mapa
233 /////
234   for (int i=0; i < numMuestras; i++)
235     {
236       suma = suma + ECG[i];
237     }
238       y= suma / numMuestras;
239 /////
240 // ININICALIZAR VALORES LOGISTICO mapa 1
241 /////
242 lcd.clear();
243 lcd.print("generando mapa 1");
244   Serial.print("generando primer mapa");
245   Serial.println(" ");
246   Serial.println(" ");
247
248   VAL_XX = CI_XX,15;
249   VAL_ANT_XX = VAL_XX,15;
250   Serial.print(CI_XX,15);
251   Serial.print(" ");
252   Log1D[0]=CI_XX,15;
253
254   for (int i = 1; i < numMuestras; i++)
255   {
```

```
256     // CALCUAR EL SIGUIENTE VALOR LOGISTICO
257     VAL_XX = aa * VAL_ANT_XX * (1.0000000 - VAL_ANT_XX );
258     Serial.print(VAL_XX,15);
259     Serial.print(" "); // imprimir el valor del estado en terminar virtual con
            salto de renglón
260     // ACTUALIZAR LOS VALOR ANTERIOR
261     VAL_ANT_XX = VAL_XX,15;
262
263     Log1D[i] = VAL_XX,15;
264     delay(1);
265   }
266
267   lcd.clear();
268 lcd.print("generando mapa 2");
269   Serial.println(" ");
270   Serial.println("generando segundo mapa");
271   Serial.println(" ");
272   Serial.println(" ");
273 /////
274 // ININICALIZAR VALORES LOGISTICO mapa 2
275 /////
276   VAL_XX1 = CI_XX1,15;//se utiliza el promedio de los valores de la señal ECG como
        valor inicial en el mapa 2
277   VAL_ANT_XX1 = VAL_XX1,15;
278   Serial.print(CI_XX1,15);
279   Serial.print(" ");
280   Log1D2[0]=CI_XX1,15;
281
282   for (int i = 1; i < numMuestras; i++)
283   {
284     // CALCUAR EL SIGUIENTE VALOR LOGISTICO
285     VAL_XX1 = aa * VAL_ANT_XX1 * (1.0000000 - VAL_ANT_XX1 );
286     Serial.print(VAL_XX1,15);
287     Serial.print(" "); // imprimir el valor del estado en terminar virtual con
            salto de renglón
288     // ACTUALIZAR LOS VALOR ANTERIOR
289     VAL_ANT_XX1 = VAL_XX1,15; // put your main code here, to run repeatedly:
290
291     Log1D2[i] = VAL_XX1,15;
292     delay(1);
293   }
294   lcd.clear();
295 lcd.print("generando mapa 3");
296   Serial.println("generando tercer mapa");
297   Serial.println(" ");
298   Serial.println(" ");
299 /////
300 // ININICALIZAR VALORES LOGISTICO mapa 3
301 /////
302   VAL_XX2 = CI_XX2,15;
303   VAL_ANT_XX2 = VAL_XX2,15;// put your setup code here, to run once:
304   //Serial.print(Log1D2[259],15);
305   //Serial.print(" ");
306
307   Log1D3[0]=Log1D2[259],15;
308
309   for (int i = 1; i < numMuestras; i++)
310   {
311     // CALCUALR EL SIGUIENTE VALOR LOGISTICO
```

```
312     VAL_XX2 = aa * VAL_ANT_XX2 * (1.0000000 - VAL_ANT_XX2 );
313     //Serial.print(VAL_XX2,15);
314     //Serial.print(" "); // imprimir el valor del estado en terminar virtual con
            salto de renglón
315     // ACTUALIZAR LOS VALOR ANTERIOR
316     VAL_ANT_XX2 = VAL_XX2,15; // put your main code here, to run repeatedly:
317
318     Log1D3[i] = VAL_XX2,15;
319     delay(1);
320   }
321 /////
322 //difusón de la señal
323 /////
324 lcd.clear();
325 lcd.print("generando Cripto");
326   Serial.print("generando Criptograma");
327   Serial.println(" ");
328   Serial.println(" ");
329
330   for(int i = 0; i < numMuestras; i++)
331   {
332     Cripto[i]=((((Log1D2[i]*2)*(Log1D[i]+Log1D3[i]))+(Log1D2[i]+Log1D3[i]))+ECG[i
          ])/10;
333   }
334
335 Serial.print("criptograma generado");
336
337 lcd.clear();
338 lcd.print("Fin Proceso");
339   //lcd.clear();
340   //lcd.print(">Executing #1");
341   delay(1500);
342 }
343 void action2() {
344   /////
345 //inicio de escritura en tarjeta
346 /////
347   archivo = SD.open("/Lectura.txt",FILE_WRITE);
348 /////
349 //Escritura del ECG en la SD
350 /////
351
352   archivo.println("ECG guardado");
353   archivo.println(" ");
354   archivo.println(" ");
355   for (int i = 0; i < numMuestras; i++)
356   {
357     archivo.print(ECG[i],15);
358     archivo.print(" ");
359     delay(1);
360   }
361
362   archivo.println(" ");
363   archivo.println(" ");
364   archivo.println(" ");
365   archivo.println(" ");
366   archivo.println(" ");
367
368 /////
```

```
369 //Escritura del criptograma en la SD
370 /////
371   archivo.println("Criptograma generado");
372   archivo.println(" ");
373   archivo.println(" ");
374   for (int i = 0; i < numMuestras; i++)
375   {
376     archivo.print(Cripto[i],15);
377     archivo.print(" ");
378     delay(1);
379   }
380
381   archivo.println(" ");
382   archivo.println(" ");
383   archivo.println(" ");
384   archivo.println(" ");
385   archivo.println(" ");
386
387
388
389   archivo.close();
390
391   Serial.println(" ");
392   Serial.println("proceso finalizado");
393   lcd.clear();
394 lcd.print("Fin Proceso");
395   //lcd.clear();
396   //lcd.print(">Executing #2");
397   delay(1500);
398 }
399 void action3() {
400   lcd.clear();
401   lcd.print(">Executing #3");
402   delay(1500);
403 }
404 void action4() {
405   lcd.clear();
406   lcd.print(">Executing #4");
407   delay(1500);
408 }
```

Apéndice B

Programa para módulo receptor

En este apéndice se presenta el código en lenguaje C con el que se programó la placa ESP32 correspondiente al módulo receptor para recibir el criptograma a tráves de la conexión wifi y finalmente guardar el criptograma o desenciptrar y guardar la señal de electrocardiograma.

```
/* Programa del módulo receptor de encriptado
caótico basado en caos con captura de señal ECG en tiempo real.
Alvaro Rodriguez Hernandez
Ingeniería en Electrónica
Universidad Autónoma de Baja California
Noviembre de 2023
*/

#include <SD.h>
#include <SPI.h>
#include <Wire.h>
#include <esp_now.h>
#include <WiFi.h>
#include <LiquidCrystal_I2C.h>

//selección del esclavo para SD
#define SS 5

LiquidCrystal_I2C lcd(0x27, 16, 2);

int upButton = 10;
int downButton = 11;
int selectButton = 12;
int menu = 1;

//
    ------------------------------------------------------------------------------------

typedef struct RxStruct
{
  double potVal;
}RxStruct;
RxStruct receivedData;
//
    ------------------------------------------------------------------------------------
```

```
33 void OnDataRecv(const uint8_t * mac, const uint8_t *incomingData, int len)
34 {
35   memcpy(&receivedData, incomingData, sizeof(receivedData));
36 }
37 //
     ==========================================================================================


38
39
40 int sel;//variable de seleeción para los botones
41
42 File archivo;//creación del objeto archivo para la tarjeta SD
43 //Numero de Muestras
44 const int numMuestras = 1000;
45 // VARIABLES LOGISTICO:
46 double aa=3.999999999779998;
47
48 double CI_XX = 0.946789876543156;
49 double CI_XX1 = 0.513469854762535;
50 double CI_XX2 = 0.245687215369874;
51 double VAL_XX=0.0;
52 double VAL_ANT_XX=0.0;
53
54 double VAL_XX1=0.0;
55 double VAL_ANT_XX1=0.0;
56
57 double VAL_XX2=0.0;
58 double VAL_ANT_XX2=0.0;
59
60 double suma;
61 double y;
62 //inicialización de los vectores
63 double Log1D[numMuestras];
64 double Log1D2[numMuestras];
65 double Log1D3[numMuestras];
66 double Cripto[numMuestras];
67 double ECG[numMuestras];
68
69 void setup() {
70   lcd.init();
71   lcd.backlight();
72   pinMode(upButton, INPUT_PULLUP);
73   pinMode(downButton, INPUT_PULLUP);
74   pinMode(selectButton, INPUT_PULLUP);
75   //inicialización de la tarjeta SD
76
77   Serial.println("inicializando tarjeta...");
78   if(!SD.begin(SS)){
79     Serial.println("fallo en inicializacion");
80   }
81   else{
82     archivo = SD.open("/Lectura.txt",FILE_WRITE);
83     Serial.println("inicializacion correcta");
84   }
85     WiFi.mode(WIFI_STA);
86   if (esp_now_init() != ESP_OK)
87   {
88     Serial.println("Error initializing ESP-NOW");
```

```
89      return;
90    }
91    esp_now_register_recv_cb(OnDataRecv);
92
93    lcd.setCursor(0, 0);
94    lcd.print("Receptor de ");
95    lcd.setCursor(0,1);
96    lcd.print("Encriptado Caos");
97    delay(6000);
98    lcd.clear();
99
100   updateMenu();
101 }
102
103 void loop() {
104   if (!digitalRead(downButton)){
105     menu++;
106     updateMenu();
107     delay(100);
108     while (!digitalRead(downButton));
109   }
110   if (!digitalRead(upButton)){
111     menu--;
112     updateMenu();
113     delay(100);
114     while(!digitalRead(upButton));
115   }
116   if (!digitalRead(selectButton)){
117     executeAction();
118     updateMenu();
119     delay(100);
120     while (!digitalRead(selectButton));
121   }
122 }
123
124 void updateMenu() {
125   switch (menu) {
126     case 0:
127       menu = 1;
128       break;
129     case 1:
130       lcd.clear();
131       lcd.print(">Recibir datos");
132       lcd.setCursor(0, 1);
133       lcd.print(" Guardar");
134       break;
135     case 2:
136       lcd.clear();
137       lcd.print(" Recibir datos");
138       lcd.setCursor(0, 1);
139       lcd.print(">Guardar");
140       break;
141     case 3:
142       lcd.clear();
143       lcd.print(">Desencriptar");
144       lcd.setCursor(0, 1);
145       lcd.print(" MenuItem4");
146       break;
147     case 4:
```

```
148       lcd.clear();
149       lcd.print(" MenuItem3");
150       lcd.setCursor(0, 1);
151       lcd.print(">MenuItem4");
152       break;
153     case 5:
154       menu = 4;
155       break;
156   }
157 }
158
159 void executeAction() {
160   switch (menu) {
161     case 1:
162       lcd.clear();
163       lcd.print("   Preparando   ");
164       lcd.setCursor(0, 1);
165       lcd.print("    receptor    ");
166       delay(5000);
167       action1();
168       break;
169     case 2:
170       lcd.clear();
171       lcd.print("   guardando    ");
172       lcd.setCursor(0, 1);
173       lcd.print("   criptogram   ");
174       delay(5000);
175       action2();
176       break;
177     case 3:
178       lcd.clear();
179       lcd.print(" desencriptando ");
180       lcd.setCursor(0, 1);
181       lcd.print("     datos      ");
182       delay(5000);
183       action3();
184       break;
185     case 4:
186       //action4();
187       break;
188   }
189 }
190
191 void action1() {
192
193   if(receivedData.potVal,15 > 0){ //el sistema solo guardará los datos si estos
        son mayores a 0, esto para prevenir datos basura
194 for(int i = 0; i < numMuestras; i++){
195     Cripto[i]=receivedData.potVal,15;
196     delay(50);
197    }
198   }
199
200   lcd.clear();
201   lcd.print("Proceso finalizado");
202   delay(1500);
203 }
204 void action2() {
205
```

```
206  /////
207 // ININICALIZAR VALORES LOGISTICO mapa 1
208 /////
209
210   Serial.print("generando primer mapa");
211   Serial.println(" ");
212   Serial.println(" ");
213
214   VAL_XX = CI_XX,15;
215   VAL_ANT_XX = VAL_XX,15;
216   Serial.print(CI_XX,15);
217   Serial.print(" ");
218   Log1D[0]=CI_XX,15;
219
220   for (int i = 1; i < numMuestras; i++)
221   {
222     // CALCUAR EL SIGUIENTE VALOR LOGISTICO
223     VAL_XX = aa * VAL_ANT_XX * (1.0000000 - VAL_ANT_XX );
224     Serial.print(VAL_XX,15);
225     Serial.print(" "); // imprimir el valor del estado en terminar virtual con
          salto de renglón
226     // ACTUALIZAR LOS VALOR ANTERIOR
227     VAL_ANT_XX = VAL_XX,15;
228
229     Log1D[i] = VAL_XX,15;
230     delay(1);
231   }
232   Serial.println(" ");
233   Serial.println("generando segundo mapa");
234   Serial.println(" ");
235   Serial.println(" ");
236 /////
237 // ININICALIZAR VALORES LOGISTICO mapa 2
238 /////
239   VAL_XX1 = CI_XX1,15;//se utiliza el promedio de los valores de la señal ECG como
        valor inicial en el mapa 2
240   VAL_ANT_XX1 = VAL_XX1,15;
241   Serial.print(CI_XX1,15);
242   Serial.print(" ");
243   Log1D2[0]=CI_XX1,15;
244
245   for (int i = 1; i < numMuestras; i++)
246   {
247     // CALCUAR EL SIGUIENTE VALOR LOGISTICO
248     VAL_XX1 = aa * VAL_ANT_XX1 * (1.0000000 - VAL_ANT_XX1 );
249     Serial.print(VAL_XX1,15);
250     Serial.print(" "); // imprimir el valor del estado en terminar virtual con
          salto de renglón
251     // ACTUALIZAR LOS VALOR ANTERIOR
252     VAL_ANT_XX1 = VAL_XX1,15; // put your main code here, to run repeatedly:
253
254     Log1D2[i] = VAL_XX1,15;
255     delay(1);
256   }
257   Serial.println("generando tercer mapa");
258   Serial.println(" ");
259   Serial.println(" ");
260 /////
261 // ININICALIZAR VALORES LOGISTICO mapa 3
```

```
262 /////
263   VAL_XX2 = CI_XX2,15;
264   VAL_ANT_XX2 = VAL_XX2,15;// put your setup code here, to run once:
265   //Serial.print(Log1D2[259],15);
266   //Serial.print(" ");
267
268   Log1D3[0]=Log1D2[259],15;
269
270   for (int i = 1; i < numMuestras; i++)
271   {
272     // CALCUALR EL SIGUIENTE VALOR LOGISTICO
273     VAL_XX2 = aa * VAL_ANT_XX2 * (1.0000000 - VAL_ANT_XX2 );
274     //Serial.print(VAL_XX2,15);
275     //Serial.print(" "); // imprimir el valor del estado en terminar virtual con
          salto de renglón
276     // ACTUALIZAR LOS VALOR ANTERIOR
277     VAL_ANT_XX2 = VAL_XX2,15; // put your main code here, to run repeatedly:
278
279     Log1D3[i] = VAL_XX2,15;
280     delay(1);
281   }
282 /////
283 //descifrado de la señal
284 /////
285
286   Serial.print("generando Criptograma");
287   Serial.println(" ");
288   Serial.println(" ");
289
290   for(int i = 0; i < numMuestras; i++)
291   {
292     ECG[i]=(Cripto[i]*10)-(((Log1D2[i]*2)*(Log1D[i]+Log1D3[i]))+(Log1D2[i]+Log1D3[
          i]));
293   }
294
295   lcd.clear();
296   lcd.print("Proceso Finalizado");
297   delay(1500);
298 }
299 void action3() {
300
301 /////
302 //inicio de escritura en tarjeta
303 /////
304   archivo = SD.open("/Lectura.txt",FILE_WRITE);
305 /////
306 //Escritura del criptograma en la SD
307 /////
308
309   archivo.println("Criptograma recibido");
310   archivo.println(" ");
311   archivo.println(" ");
312   for (int i = 0; i < numMuestras; i++)
313   {
314     archivo.print(Cripto[i],15);
315     archivo.print(" ");
316     delay(1);
317   }
318
```

```
319   archivo.println(" ");
320   archivo.println(" ");
321   archivo.println(" ");
322   archivo.println(" ");
323   archivo.println(" ");
324 
325 /////
326 //Escritura de señal clara en la SD
327 /////
328   archivo.println("ECG");
329   archivo.println(" ");
330   archivo.println(" ");
331   for (int i = 0; i < numMuestras; i++)
332   {
333     archivo.print(ECG[i],15);
334     archivo.print(" ");
335     delay(1);
336   }
337 
338   archivo.println(" ");
339   archivo.println(" ");
340   archivo.println(" ");
341   archivo.println(" ");
342   archivo.println(" ");
343 
344 
345 
346   archivo.close();
347 
348   Serial.println(" ");
349   Serial.println("proceso finalizado");
350 
351 
352 }
353 void action4() {
354   lcd.clear();
355   lcd.print(">Executing #4");
356   delay(1500);
357 }
```