UNIVERSIDAD AUTÓNOMA DE SAN LUIS POTOSÍ

INSTITUTO DE INVESTIGACIÓN EN COMUNICACIÓN ÓPTICA

ANÁLISIS ESTADÍSTICO DE CIFRADO PARCIAL, EN IMÁGENES DIGITALES

Tesis para obtener el grado de Maestro en Ciencias

Presenta: Ing. Físico Oliver Jesús Espinosa Olvera

Asesores:
Dra. Marcela Mejía Carlos
Dr. José Salomé Murguía Ibarra

Febrero - 2018

Universidad Autónoma de San Luis Potosí

Facultad de Ciencias

Los miembros del comité de tesis recomiendan la aceptación de la tesis titulada ANÁLISIS ESTADÍSTICO DE CIFRADO PARCIAL de Oliver Jesús Espinosa Olvera como cumplimiento parcial de los requisitos para obtener el grado de: Maestro en Ciencias Aplicadas con orientación en Fotónica.

Dr. José Salomé Murguía Ibarra
Asesor y Sinodal

Dra. Marcela Mejía Carlos
Asesor y Sinodal

Dr. Marco Tulio Ramírez Torres
Sinodal

Dr. Isaac Campos Cantón
Sinodal

Agradecimientos

A Dios, por todo.

A mis padres, Blanca Azucena y J. Jesús, por su apoyo incondicional a lo largo de mi vida, y por instruirme en la ciencia desde pequeño.

A mis asesores, la Dra. Marcela Mejía y el Dr. José Salomé Murguía, por su asesoramiento y guía a lo largo de mi posgrado.

Al Dr. Marco Tulio Ramírez, por su ayuda en la obtención de datos para realizar esta tesis.

Al Dr. Raúl Balderas, por su ayuda para realizar todos los trámites respectivos al posgrado.

Al Dr. Salvador Guel y al Dr. Gustavo Ramírez, por su amistad, y por adentrarme al mundo de la física. Y a los Doctores Ricardo Castro y David Ariza, por haber sido parte de mi formación académica al estudiar este posgrado.

A la Ing. Liliana Mendonza y a la Mtra. Griselda Rodríguez. Por el apoyo y amistad brindados a lo largo de mi carrera y mi posgrado. Y a mis compañeros, también por brindarme su amistad, y hacer ameno el tiempo que pasé con ellos. Gracias Milfits.

A mis tíos Martín y Juanela Ríos, Manuel y Perla García, y Uriel y María Ordorica. Por el cariño que me han dado a lo largo de mi vida, y por haber sido parte de mi formación como estudiante, como profesionista, y como persona. Este logro también es de ustedes.

A Christian Ordorica, Francisco Rivera e Ismael Vázquez. Por la gran amistad y el apoyo que me han brindado a lo largo de mi vida.

A mi Pastor Luis Ramos Cisneros. A mi hermano Víctor Hugo Guel González, y a la Iglesia Bíblica Bautista de San Luis Potosí.

"Porque Jehová da la sabiduría,
Y de su boca viene el conocimiento y la inteligencia.
El provee de sana sabiduría a los rectos;
Es escudo a los que caminan rectamente."
Proverbios 2:6-7

Resumen

La criptografía tiene muchas aplicaciones pero su función en general siempre ha sido la misma, dotar de seguridad a diferente tipo de información, tal como un mensaje, o a un valor numérico, para que este sea irreconocible ante un usuario no autorizado. El objetivo principal de este trabajo es realizar un análisis estadístico de un sistema de cifrado parcial aplicado a imágenes digitales con diferentes niveles de actividad óptica. El cifrado parcial de imágenes se refiere a que solo una cantidad determinada de bits de la imagen son cifrados, mientras que los bits restantes preservan su valor original.

El sistema de cifrado considerado, el cual está basado en la sincronización de autómatas celulares, realiza el cifrado de diferente tipo de señales como lo son las imágenes. Para tal efecto, la implementación numérica del sistema de cifrado, el cual proporciona seguridad a bloques de longitud $L = 2^n$ bits, se realiza con el software de LabVIEW, en particular con el paquete VISION; tal software nos permite manipular de manera flexible las imágenes, las cuales son sometidas a distintas pruebas después de haber sido cifradas. Los resultados obtenidos nos indican el nivel de seguridad que conlleva el respectivo cifrado realizado.

Índice general

Índice de figuras

Capítulo 1

Introducción

Desde siglos atrás, la comunicación tanto entre personas como entre naciones ha sido de vital importancia para el desarrollo humano y social, con la necesidad de mantener en secreto parte de esta comunicación. Con la finalidad de llevar a cabo lo anterior se considera la criptografía, la cual se define como el arte o la ciencia de cifrar un mensaje, para que este sea ilegible ante un usuario no autorizado. Con el uso de una *llave de cifrado* se modifican las letras y/o números de un mensaje, para alternarlos por otros que lo vuelvan ilegible. Esta *llave de cifrado* contiene los parámetros iniciales con los cuales la iteración para cifrar el mensaje inicia. Los mensajes se pueden cifrar con una sola llave para todo el mensaje, o una serie llaves que se generan a partir de la iteración de la primer llave, con el mensaje. Estas reglas para cifrar (o descifrar) se conocen como algoritmos de cifrado. Hay dos clases de algoritmos de cifrado: los algoritmos simétricos (o de llave secreta) y los asimétricos (o de llave pública) [1]. Los primeros usan la misma llave para realizar los procesos de cifrado y descifrado, mientras que los segundos usan una llave diferente para el cifrado y descifrado [2]. Hoy en día, la gran mayoría de información a cifrar se trata de datos binarios. Estos a su vez representan los números y/o letras que queremos esconder de los usuarios no autoriza-

dos. Dentro de las aplicaciones más comunes, se encuentran los datos bancarios, los cuales necesitan estar cifrados al momento de enviarse de una terminal a otra. Los mensajes militares, necesariamente deben estar cifrados al momento de enviarse de un punto a otro para evitar el robo de información que puede ser vital para una nación. Sin embargo, en muchos casos, debido al alto nivel de seguridad que representa el cifrado, no es necesario realizarlo de manera total, sino parcial. Ya sea para ganar tiempo o evitar el uso de muchos recursos por parte del ordenador, se puede optar por solamente cifrar una cierta cantidad de información, y que a su vez no represente una pérdida grande de seguridad en el mensaje, y este siga siendo ilegible ante un usuario no autorizado. Múltiples sistemas de cifrado se han realizado con el paso de los años. En particular hablaremos del sistema de cifrado basado en autómatas celulares (ESCA, por sus siglas en ingles) [3], el cual también ha sido desarrollado de manera matricial. Las condiciones con las cuales se generan las llaves de cifrado se obtienen de datos discretos establecidos desde un principio. Este sistema ha sido implementado para cifrar el valor binario de cada pixel en una imagen, transformándolo en otro valor y por ende en otro tono de color para el pixel. El propósito de esta tesis, es analizar el cifrado parcial aplicado en imágenes digitales, realizando diversas pruebas y ataques para evaluar el nivel de seguridad que dicho cifrado parcial proporciona. La estructura del presente trabajo es la siguiente. En el Capítulo 2 se presentan los fundamentos del sistema de cifrado utilizado. En el Capítulo 3 se describe la implementación numérica del sistema, así como su aplicación de manera parcial. También se describen brevemente las pruebas aplicadas a los resultados para su posterior análisis estadístico. En el Capítulo 4 se muestran los resultados obtenidos de haber aplicado las diferentes pruebas a un grupo de imágenes cifradas parcialmente. Finalmente en el Capítulo 5 se presentan las conclusiones a las cuales se llegaron del análisis de los resultados obtenidos.

Capítulo 2

Preliminares y fundamentos del sistema de cifrado

En este capítulo se presenta una descripción general de los procesos del sistema ESCA, los cuales realizan las etapas del cifrado y descifrado de datos, así como conceptos fundamentales utilizados para dichos procesos.

2.1. Fundamentos

2.1.1. Autómatas Celulares

Los autómatas celulares (AC) surgen en la década de 1940 con John Von Neumann, que intentaba modelar una máquina que fuera capaz de autoreplicarse, llegando así a un modelo matemático de dicha máquina con reglas complicadas sobre una red rectangular. Inicialmente fueron interpretados como conjunto de células que crecían, se reproducían y morían a medida que pasaba el tiempo. Su nombre se debe a esta similitud con el crecimiento de las células.

Un autómata celular es un modelo matemático para un sistema dinámico compuesto por un conjunto de celdas o células que adquieren distintos estados o valores [4]. Estos estados son alterados de un instante a otro en unidades de tiempo discreto, es decir, que se puede cuantificar con valores enteros a intervalos regulares. De esta manera este conjunto de células logran una evolución según una determinada expresión matemática, que es sensible a los estados de las células vecinas, y que se conoce como regla de *transición local*.

Una de las características de los Autómatas Celulares, es su **Función Local**, la cual, es la regla de evolución que determina el comportamiento del Autómata Celular. Dicha función se conforma de una célula central y sus vecindades y define como debe cambiar de estado cada célula dependiendo de los estados anteriores de sus vecindades. Tal función puede ser una expresión algebraica o un grupo de ecuaciones [4].

2.1.2. Regla 90

En la dinámica de un autómata celular para el caso unidimensional, se debe considerar una especie de rejilla de celdas, donde cada una de las cuales tiene un único valor binario, 0 o 1. La asignación de valores a todas las celdas, se conoce como configuración. El autómata recibe una configuración inicial y luego progresa a través de otras configuraciones en una secuencia de pasos de tiempo discretos. En cada paso todas las celdas se actualizan simultáneamente. Una regla pre-especificada determina el nuevo valor de cada celda como una función de su valor anterior y de los valores en sus celdas vecinas. Todas las celdas obedecen a la misma regla, la cual puede ser dada como una fórmula o como una tabla de reglas que especifica el nuevo valor para cada posible combinación de valores vecinos. La Regla 90, es una de las reglas elementales de autómatas celulares introducida por Stephen Wolfram en 1983. Esta regla especifica el valor en una celda dependiendo de los valores de sus vecinos inmediatos.

Los resultados de la regla están codificados en la representación binaria *90 = 01011010*. Esta regla se ilustra en la figura 2.1 junto el estado inicial de un autómata celular, y su respectiva evolución durante 15 pasos discretos de tiempo [5].

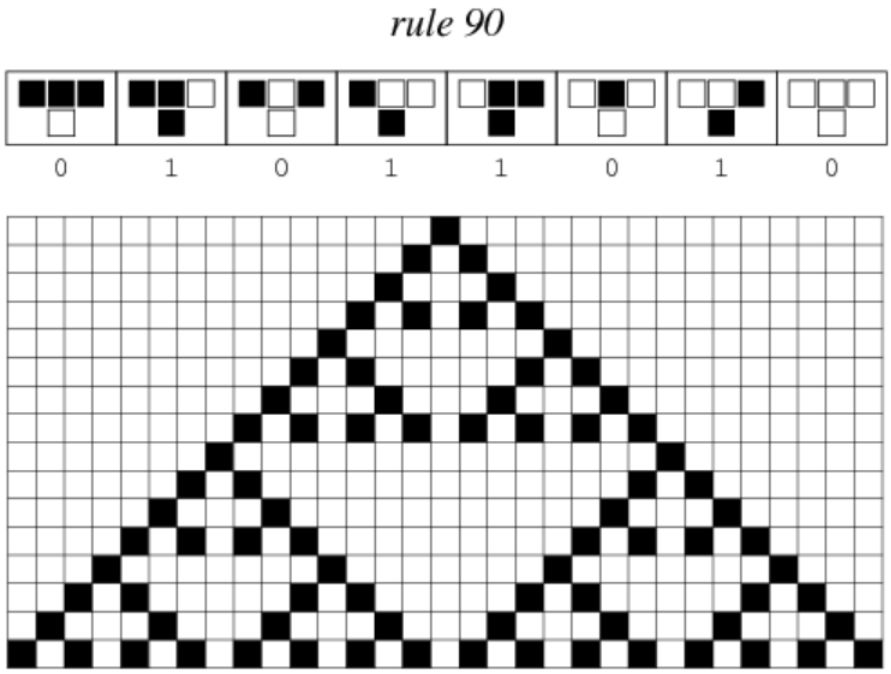

Figura 2.1: Representación de la regla 90, y evolución de un autómata celular bajo dicha regla.

2.2. Sistema ESCA

La figura 2.2 muestra un diagrama de bloques del sistema ESCA en el modo de cifrado. Se requiere de un *bloque de texto en claro m*, con un tamaño de L bits, donde $L = 2^j$, para $j = 1, 2, 3, ...$ (para fines prácticos y demostrativos, se tomará $L = 8$). *m* será el bloque a cifrar. Asimismo se requiere de un *bloque aleatorio z*, de tamaño $L + 1$ bits, con el cual se realizará un procesamiento a m, obteniendo a la salida un *bloque de datos procesado*, $\hat{m}$, de

tamaño L. Esta etapa corresponde al procesado.

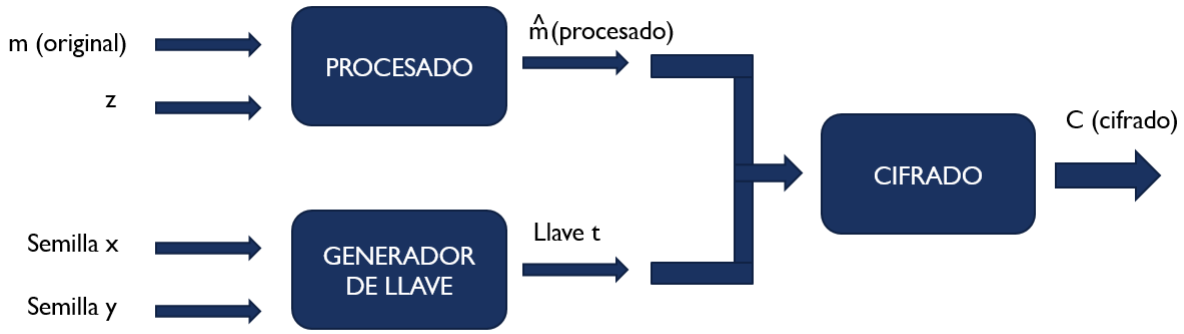

Figura 2.2: Proceso de cifrado del sistema ESCA.

Se requiere de igual manera de un par de semillas (x,y), de tamaño N y $N+1$ bits respectivamente, donde $N = 2^n - 1$, para $n = 1, 2, 3, ...$ y $N > L$. Estas 2 semillas interactuarán entre si para generar una *llave de cifrado t* de tamaño N. Con esta llave se procede a cifrar el *bloque de datos procesado*, obteniendo el *bloque cifrado c* de tamaño L bits. Existen varias formas de realizar las operaciones necesarias en la interacción de los datos de entrada, pero en este trabajo se considera el enfoque matricial utilizado en [3]. La expresión algebraica para obtener el *bloque cifrado c*, a partir del *bloque de datos procesado* $\hat{m}$ y la *llave de cifrado t* se muestra en la ecuación (2.1).

$$c = (P \oplus t \oplus Q \oplus \hat{m}) \tag{2.1}$$

donde **P** y **Q** son matrices de tamaño $L \times N$ y $L \times L$ respectivamente, y $\oplus$ corresponde a la operación *xor*. Dichas matrices se describirán más adelante.

En la figura 2.3 se muestra un diagrama de bloques del sistema ESCA ahora en el modo

de descifrado. En este caso se requiere del *bloque de texto cifrado c*, con un tamaño de L bits. Para poder descifrar este bloque se hace uso de la misma *llave* con la cual se cifró en un principio. Una vez descifrado, este *bloque de texto cifrado* pasa a ser el *bloque de datos procesado* $\hat{m}$. La fórmula para obtener dicho bloque a partir del del *bloque de texto cifrado c* y la *llave de cifrado t*, es:

$$\hat{m} = (\hat{R} \oplus t \oplus \hat{T} \oplus c) \tag{2.2}$$

donde $\hat{\mathbf{R}}$ y $\hat{\mathbf{T}}$ son matrices de tamaño $L \times N$ y $L \times L$ respectivamente. De igual manera que el caso anterior, dichas matrices se describirán más adelante.

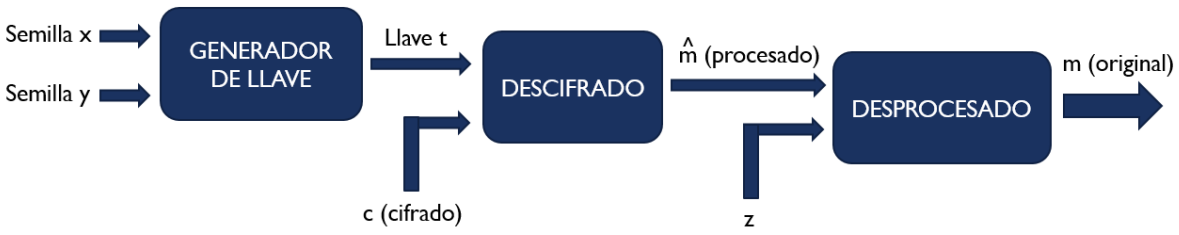

Figura 2.3: Proceso de descifrado del sistema ESCA.

Para desprocesar ese bloque obtenido, se requiere del mismo *bloque aleatorio* z que se usó para procesarlo. Ambos (z y $\hat{m}$) interactúan entre si para obtener a la salida el *bloque de texto en claro m*, de tamaño *L* bits. Las matrices usadas y las interacciones realizadas con los respectivos bloques para: procesado, cifrado, descifrado y desprocesado, se detallan a continuación.

2.2.1. Cifrado

Pre-Procesado

Para realizar el pre-procesado, se hace uso de la función de procesamiento (2.3), la cual es una función matricial donde los bloques de *texto en claro* y *aleatorio* se toman como vectores. La función es la siguiente:

$$\begin{pmatrix} \hat{m} \\ z_{k+1} \end{pmatrix} = \mathbf{V}_N \begin{pmatrix} m \\ z_k \end{pmatrix} \tag{2.3}$$

donde $\mathbf{V}_N$ es una matriz de tamaño $(2L+1) \times (2L+1)$, m es el *bloque de texto en claro*, z_k es el *bloque aleatorio*, $\hat{m}$ es el *bloque procesado* y z_{k+1} es un nuevo *bloque aleatorio* que se utilizará en la siguiente iteración de procesamiento. Formalmente, la matriz $\mathbf{V}_N$ se define como:

$$\mathbf{V}_N = \begin{pmatrix} \mathbf{H}_{NT} \\ \mathbf{H}_{NT} \\ b \end{pmatrix} \tag{2.4}$$

donde $\mathbf{H}_{NT}$ es una matriz de tamaño $L \times (2L+1)$ y b es un vector de tamaño $1 \times (2L+1)$. Para implementar la matriz $\mathbf{H}_{NT}$, en un principio se consideran sus 2 primeras filas, v y w, las cuales se conforman de la siguiente manera:

$$v = \begin{pmatrix} v_1, & 0, & \ldots & v_{L+2}, & 0, & \ldots & 0 \end{pmatrix} \tag{2.5}$$

$$w = \begin{pmatrix} 0, & w_2, & 0 & \ldots & w_{L+1}, & 0, & w_{L+3} & 0, & \ldots & 0 \end{pmatrix} \tag{2.6}$$

donde las posiciones denotadas por v_1, v_{L+2}, w_2, w_{L+1} y w_{L+3} toman el valor de 1. Por

tanto, para $L = 8$, las 2 primeras filas de la matriz $\mathbf{H}_{NT}$ son las siguientes:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

La regla que obedece este autómata celular para evolucionar se muestra en la figura 2.4. Consiste en hacer un corrimiento a la derecha en la fila de abajo, tomar como condición de frontera el añadir un cero en el espacio vacío de la primera célula de la izquierda, y aplicar la operación lógica xor entre los bits de la primera y segunda fila, para generar los bits de una tercera fila. Este proceso se lleva a cabo $L - 2$ veces, para obtener entonces las L filas.

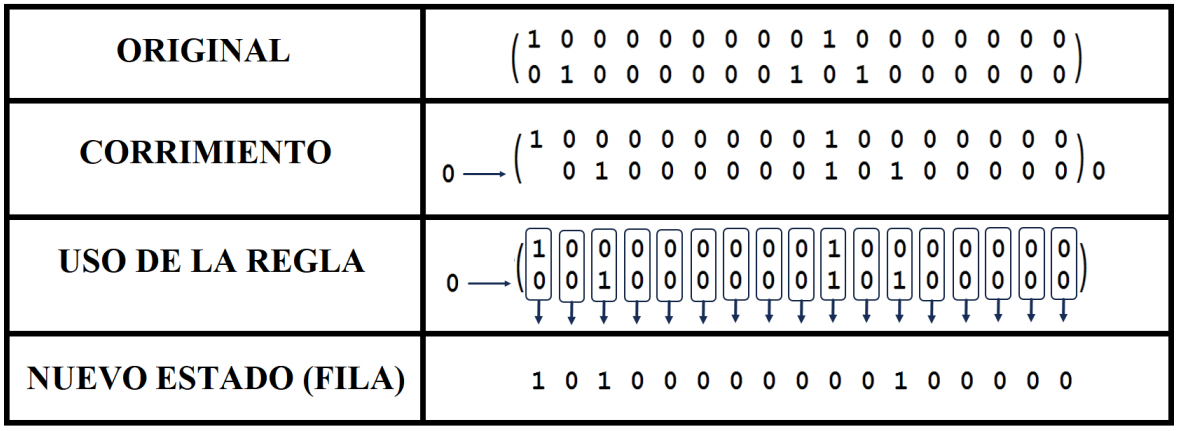

Figura 2.4: Proceso de evolución del estado inicial de $\mathbf{H}_{NT}$ para generar el resto de las filas.

Como ejemplo, la matriz $\mathbf{V}_N$ para $L = 8$ bits es:

$$\mathbf{V}_N = \begin{pmatrix} 1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&1&0&1&0&0&0&0&0&0&0 \\ 1&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0&1&0&1&0&1&0&0&0&0&0 \\ 1&0&1&0&1&0&0&1&0&0&0&1&0&1&0&0&0&0 \\ 0&1&0&0&0&1&0&0&1&0&0&0&1&0&1&0&0&0 \\ 1&0&0&0&1&0&1&0&0&0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&1&1&0&0&0&1&0&1&0&1&0 \\ 1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&1&0&1&0&0&0&0&0&0&0 \\ 1&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0&1&0&1&0&1&0&0&0&0&0 \\ 1&0&1&0&1&0&0&1&0&0&0&1&0&1&0&0&0&0 \\ 0&1&0&0&0&1&0&0&1&0&0&0&1&0&1&0&0&0 \\ 1&0&0&0&1&0&1&0&0&0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&1&1&0&0&0&1&0&1&0&1&0 \\ 0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0 \end{pmatrix} \tag{2.7}$$

Asimismo, el vector b de tamaño $1 \times L$, está formado por ceros, y un uno en la posición L. Al multiplicar $\mathbf{V}_N$ por el vector formado por el *bloque de texto en claro m* y el *bloque aleatorio z*, obtenemos el *bloque procesado* $\hat{m}$ y un nuevo *bloque aleatorio* como se muestra en la ecuación (2.8). Este nuevo *bloque aleatorio* es el utilizado en la siguiente iteración de procesamiento, por lo tanto para el usuario solo le es necesario incluir un primer *bloque aleatorio z*, y el sistema produce los siguientes a usar en las próximas iteraciones.

$$\begin{pmatrix} 1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&1&0&1&0&0&0&0&0&0&0 \\ 1&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0&1&0&1&0&1&0&0&0&0&0 \\ 1&0&1&0&1&0&0&1&0&0&0&1&0&1&0&0&0&0 \\ 0&1&0&0&0&1&0&0&1&0&0&0&1&0&1&0&0&0 \\ 1&0&0&0&1&0&1&0&0&0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&1&1&0&0&0&1&0&1&0&1&0 \\ 1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&1&0&1&0&0&0&0&0&0&0 \\ 1&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0&1&0&1&0&1&0&0&0&0&0 \\ 1&0&1&0&1&0&0&1&0&0&0&1&0&1&0&0&0&0 \\ 0&1&0&0&0&1&0&0&1&0&0&0&1&0&1&0&0&0 \\ 1&0&0&0&1&0&1&0&0&0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&1&1&0&0&0&1&0&1&0&1&0 \\ 0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0 \end{pmatrix} \begin{pmatrix} m1 \\ m2 \\ m3 \\ m4 \\ m5 \\ m6 \\ m7 \\ m8 \\ z_k1 \\ z_k2 \\ z_k3 \\ z_k4 \\ z_k5 \\ z_k6 \\ z_k7 \\ z_k8 \\ z_k9 \end{pmatrix} = \begin{pmatrix} \hat{m}1 \\ \hat{m}2 \\ \hat{m}3 \\ \hat{m}4 \\ \hat{m}5 \\ \hat{m}6 \\ \hat{m}7 \\ \hat{m}8 \\ z_{k+1}1 \\ z_{k+1}2 \\ z_{k+1}3 \\ z_{k+1}4 \\ z_{k+1}5 \\ z_{k+1}6 \\ z_{k+1}7 \\ z_{k+1}8 \\ z_{k+1}9 \end{pmatrix} \tag{2.8}$$

Para formar el nuevo *bloque aleatorio*, se utiliza el *bloque procesado* obtenido. Sin embargo, este *bloque procesado* es de tamaño $L = 8$ bits, y los *bloques aleatorios* son de tamaño $L + 1$ bits. Para obtener este bit, se toma el bit más significativo del primer *bloque aleatorio*, y pasa a ser el bit menos significativo en el nuevo *bloque aleatorio*. La figura 2.5 muestra a detalle este proceso para un tamaño $L = 8$ bits. Cabe destacar que debido a la naturaleza de este sistema, la retroalimentación permite procesar cada *bloque de texto en claro* con un *bloque aleatorio* diferente en cada caso.

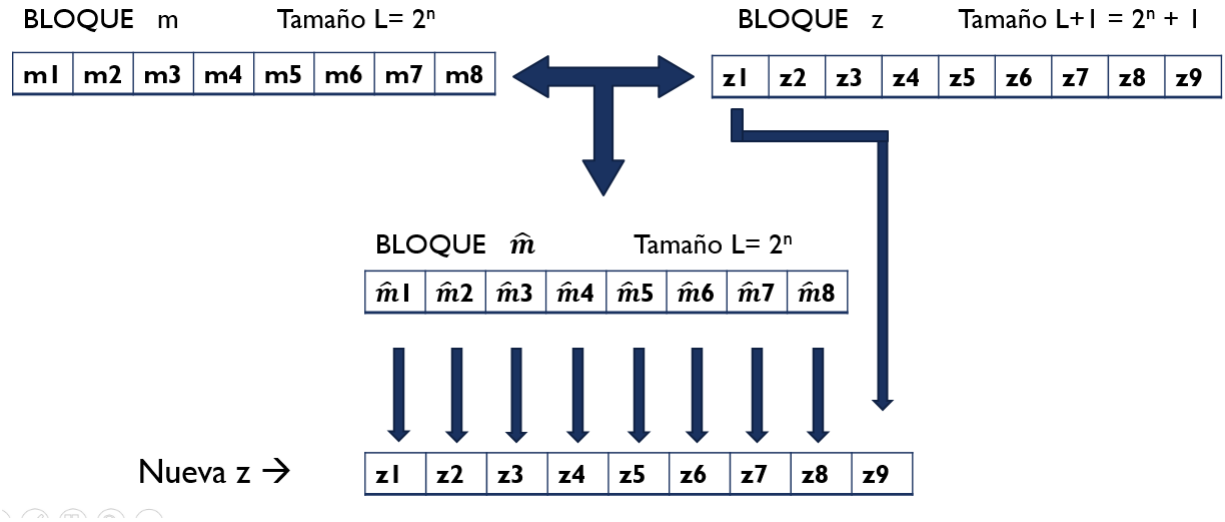

Figura 2.5: Diagrama de la retroalimentación de z para la siguiente iteración.

Las ecuaciones para obtener cada uno de los bits del *bloque procesado* $\hat{m}$ con $L = 8$ se incluyen en el apéndice A.

Generación de llave

Para poder cifrar el bloque ya procesado, se requiere de una *llave de cifrado t* de tamaño N. Para obtener esta llave, se hace uso la función matricial $\mathbf{H}_N$, y un par de semillas (x,y) de

tamaño N y $N+1$ bits respectivamente (2.10).$\mathbf{H}_N$ es una matriz conformada por:

$$\mathbf{H}_N = \begin{pmatrix} \mathbf{H}_{Nt} \\ \mathbf{H}_{Nb} \end{pmatrix} \tag{2.9}$$

donde la matriz $\mathbf{H}_{Nt}$ en este caso tiene dimensiones $(N * (2N+1))$ (no confundir con dimensiones usadas en (2.7)). La matriz $\mathbf{H}_{Nb}$ tiene dimensiones $(N+1) * (2N+1)$ y es la matriz identidad en las primeras $(N+1)$ columnas.

Al multiplicar $\mathbf{H}_N$ por el vector formado por ambas semillas, obtenemos la *llave de cifrado* t y una nueva semilla y. Los primeros N bits de esta nueva semilla y se toman de la anterior semilla x utilizada y el último bit (el menos significativo) se toma del más significativo de la semilla y anterior. Está nueva semilla y será utilizada en la siguiente iteración para generar otra *llave de cifrado*. La próxima semilla x a utilizar, será la misma llave t generada anteriormente, por lo tanto para el usuario solo le es necesario incluir un primer par de semillas, y el sistema produce las siguientes a usar en las próximas iteraciones. La figura 2.6 muestra un diagrama simplificado de dicho proceso.

$$\begin{pmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&1&0&0&0&0&1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0&1&0&0&0&0&0&0&0&0&0&0&0\\
1&0&1&0&1&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&1&0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&1&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&1&0&0&0&1&0&1&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&1&0&1&0&1&0&0&0&0&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&0&0&0&0\\
0&1&0&0&0&1&0&0&0&1&0&0&0&0&0&1&0&1&0&1&0&0&0&1&0&1&0&0&0&0&0&0\\
1&0&1&0&1&0&0&0&1&0&1&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&1&0&0&0&1&0&1&0&0&0&0&0&1&0&1&0&1&0&0&0&0\\
1&0&1&0&0&0&0&0&1&0&1&0&1&0&0&0&1&0&0&0&0&0&0&0&1&0&0&0&1&0&0&0\\
0&1&0&0&0&0&0&0&0&1&0&0&0&1&0&1&0&0&0&0&0&0&0&1&0&0&0&1&0&1&0&0\\
1&0&0&0&0&0&0&0&1&0&0&0&1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{pmatrix}
\begin{pmatrix} x1\\x2\\x3\\x4\\x5\\x6\\x7\\x8\\x9\\x10\\x11\\x12\\x13\\x14\\x15\\y1\\y2\\y3\\y4\\y5\\y6\\y7\\y8\\y9\\y10\\y11\\y12\\y13\\y14\\y15\\y16 \end{pmatrix}
=
\begin{pmatrix} t1\\t2\\t3\\t4\\t5\\t6\\t7\\t8\\t9\\t10\\t11\\t12\\t13\\t14\\t15\\y1\\y2\\y3\\y4\\y5\\y6\\y7\\y8\\y9\\y10\\y11\\y12\\y13\\y14\\y15\\y16 \end{pmatrix} \tag{2.10}$$

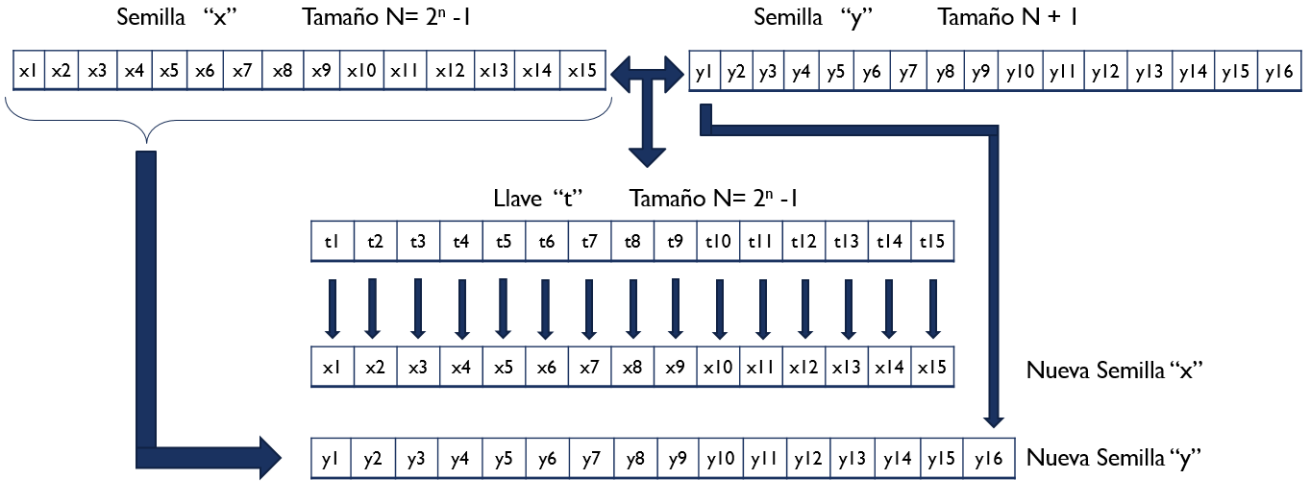

Figura 2.6: Diagrama de la retroalimentación de (x,y) para la siguiente generación de llave t.

Las ecuaciones para obtener todos los bits de la *llave de cifrado* para $N = 15$ se incluyen en el apéndice A.

Cifrado

Una vez realizado el pre-procesado y habiendo generado la *llave de cifrado* procedemos a cifrar el *bloque procesado*. Para esto, se hace uso de la función de cifrado (2.1), la cual es una función matricial donde el *bloque procesado* y la *llave de cifrado* de se toman como vectores. En dicha ecuación, la primer fila de la matriz **P** de tamaño $L \times N$, es generada por el vector

$$p = \left(p_1,\ p_2,\ \ldots\ p_N \right)$$

donde las componentes en la posición $j = (2^n + 1) - 2^{i+1}, i = 0, 1, 2, ..., (n-1)$ toman el valor de 1, mientras que en el resto de las posiciones se les asigna el valor de 0. En la figura 2.7 se muestra la forma de generar las siguientes $L-1$ filas. Consiste en hacer un corrimiento a la derecha en la fila, tomar como condición de frontera el añadir un cero en el espacio vacío

de la primera célula de la izquierda, y aplicar la operación lógica xor entre los bits de la fila y 0's para cada una de las células.

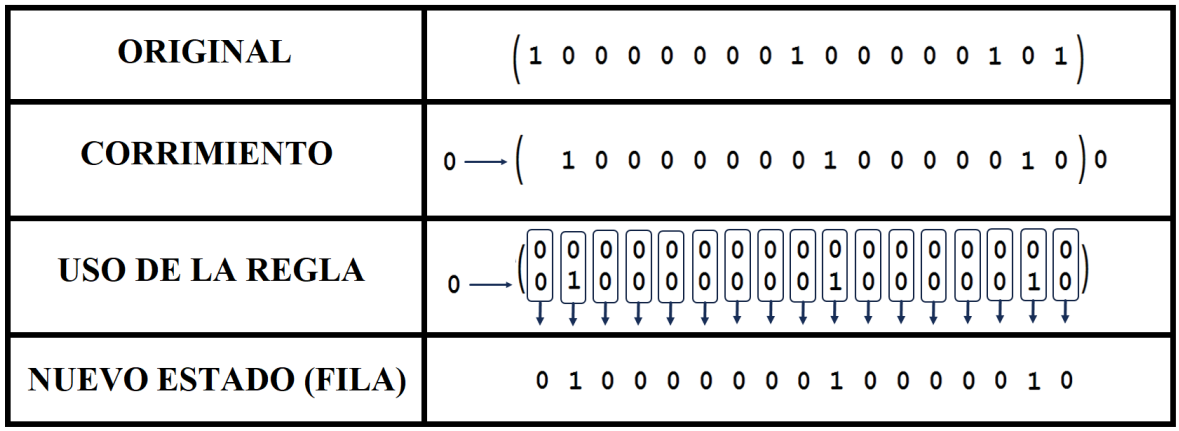

Figura 2.7: Proceso de evolución del estado inicial de **P** para generar el resto de las filas.

De este modo para $L = 8$ y $N = 15$ la matriz **P** es la siguiente:

$$\mathbf{P} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{2.11}$$

Por otra parte **Q** es una matriz de tamaño $(L \times L)$, inicialmente generada por el vector

$$a = \begin{pmatrix} a_1, & 0, & \ldots & 0 \end{pmatrix}$$

Donde la componente a_1 tiene valor 1. Para generar las siguientes $L - 1$ filas se hace uso de la regla 90, tomando como condición de frontera el valor de 0 en los límites derecho e

izquierdo. De este modo para $L = 8$ la matriz **Q** es la siguiente:

$$\mathbf{Q} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{2.12}$$

La matriz **Q** nos indicará cuales bits del *bloque procesado* se utilizarán para cifrar los bits deseados, mientras que la matriz **P** nos indicará cuales bits de la *llave de cifrado* se utilizarán en dicho proceso. De manera matricial, la ecuacion (2.1) queda de la siguiente manera:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \oplus \begin{pmatrix} t1 \\ t2 \\ t3 \\ t4 \\ t5 \\ t6 \\ t7 \\ t8 \\ t9 \\ t10 \\ t11 \\ t12 \\ t13 \\ t14 \\ t15 \end{pmatrix} \oplus \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \oplus \begin{pmatrix} \hat{m}1 \\ \hat{m}2 \\ \hat{m}3 \\ \hat{m}4 \\ \hat{m}5 \\ \hat{m}6 \\ \hat{m}7 \\ \hat{m}8 \end{pmatrix} = \begin{pmatrix} c1 \\ c2 \\ c3 \\ c4 \\ c5 \\ c6 \\ c7 \\ c8 \end{pmatrix} \tag{2.13}$$

Las ecuaciones para obtener los bits del *bloque de texto cifrado c*, para $L = 8$ se incluyen en el apéndice A.

2.2.2. Descifrado

Descifrado del bloque

Para poder decifrar el *bloque cifrado c* se hace uso de la función de descifrado (2.2), en la cual, este bloque y la *llave de cifrado t*, interactúan con 2 matrices $\hat{\mathbf{R}}$ y $\hat{\mathbf{T}}$ para obtener a la salida el *bloque procesado* $\hat{m}$. En dicha ecuación $\hat{\mathbf{R}}$ es una matriz de tamaño $N \times L$, obtenida por $\hat{\mathbf{R}} = [-\mathbf{Q}^{-1}\mathbf{P}]$. Esta matriz indica los bits de la *llave de cifrado t*, a utilizar. Por otra parte, $\hat{\mathbf{T}}$ es una matriz de tamaño $L \times L$ obtenida a partir de $\hat{\mathbf{T}} = \mathbf{Q}^{-1}$. Esta matriz $\hat{\mathbf{T}}$ indica los bits del *bloque cifrado c*, a utilizar en cada ecuación para llevar a cabo el decifrado completo del bloque. Dichas matrices se muestran a continuación para $L = 8$ y $N = 15$.

$$\hat{\mathbf{R}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{2.14}$$

$$\hat{\mathbf{T}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{2.15}$$

De manera matricial, la ecuación (2.2) queda de la siguiente manera:

$$\begin{pmatrix} 1&0&0&0&0&0&0&0&1&0&0&0&1&0&1 \\ 0&1&0&0&0&0&0&0&0&1&0&0&0&1&1 \\ 1&0&1&0&0&0&0&0&1&0&1&0&1&0&0 \\ 0&0&0&0&1&0&0&0&0&0&0&1&0&0&0 \\ 1&0&1&0&1&0&0&0&1&0&1&0&0&0&0 \\ 0&1&0&0&0&1&0&0&0&1&0&0&0&0&0 \\ 1&0&0&0&1&0&1&0&1&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&1&0&0&0&0&0&0&0 \end{pmatrix} \begin{pmatrix} t1 \\ t2 \\ t3 \\ t4 \\ t5 \\ t6 \\ t7 \\ t8 \\ t9 \\ t10 \\ t11 \\ t12 \\ t13 \\ t14 \\ t15 \end{pmatrix} \oplus \begin{pmatrix} 1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 1&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 1&0&1&0&1&0&0&0 \\ 0&1&0&0&0&1&0&0 \\ 1&0&0&0&1&0&1&0 \\ 0&0&0&0&0&0&0&1 \end{pmatrix} \begin{pmatrix} c1 \\ c2 \\ c3 \\ c4 \\ c5 \\ c6 \\ c7 \\ c8 \end{pmatrix} = \begin{pmatrix} \hat{m}1 \\ \hat{m}2 \\ \hat{m}3 \\ \hat{m}4 \\ \hat{m}5 \\ \hat{m}6 \\ \hat{m}7 \\ \hat{m}8 \end{pmatrix} \quad (2.16)$$

Desprocesamiento

Para poder recuperar el *bloque de texto en claro m*, después de realizar el descifrado, se hace uso de la función de desprocesamiento (2.17), en la cual $\hat{m}$ interactua con el *bloque aleatorio z* de la siguiente manera:

$$\begin{pmatrix} m \\ z_{k+1} \end{pmatrix} = \mathbf{W} \begin{pmatrix} \hat{m} \\ z_k \end{pmatrix} \quad (2.17)$$

En esta ecuación, la matriz **W** está formada de la siguiente manera:

$$\mathbf{W} = \begin{pmatrix} \mathbf{M}_N \\ \mathbf{H}_{NB} \\ b \end{pmatrix} \quad (2.18)$$

donde la matriz $\mathbf{M}_N$ se obtiene concadenando las matrices **Q** y **D**. La matriz **D** posee dimensiones $L \times (L+1)$. Contiene valores de uno en la diagonal, así como en las posiciones establecidas por $k_a = 2^a - 1$, para $a = 1, 2...n$. Todos los demás valores son iguales a cero. $\mathbf{H}_{NB}$ es la parte inferior de la matriz $\mathbf{H}_N$ y b es un vector de dimensiones $1 \times (2L+1)$ que contiene un uno en la posición $L+1$ y ceros en el resto.

A la salida de esta ecuación obtenemos el *bloque de texto en claro m*, así como un nuevo *bloque aleatorio z*, que será el siguiente bloque utilizado en la próxima iteración para despro-

cesar. El *bloque aleatorio z* usado en la primera iteración de este proceso, debe ser el mismo utilizado para llevar a cabo en el procesado del *bloque de texto en claro m*.

Las ecuaciones para realizar el desprocesado a cada uno de los bits de $\hat{m}$ y obtener el *bloque de texto en claro m*, se incluyen en el apéndice A.

Capítulo 3

Implementación numérica del sistema de cifrado, y su aplicación de manera parcial

A continuación se presenta una descripción de la modificación realizada al sistema ESCA para realizar el cifrado de manera parcial, así como su implementación en la plataforma LabVIEW[6] para la obtención de las imágenes cifradas, y las pruebas realizadas para evaluar dicho nivel de cifrado.

3.1. Cifrado Parcial

El sistema ESCA es capaz de cifrar bloques de bits de tamaño $L = 2^n$, sin embargo, no es necesario cifrar todos los bits del bloque. El cifrado parcial se refiere, a que solo una cantidad específica de bits del *bloque de texto en claro* son cifrados, mientras que el resto se mantienen inalterados. En este caso en particular, el *bloque de texto en claro m*, es de 8 bits. El cifrado parcial realizado a este bloque fue de 3 y 4 bits, dejando inalterados los 5 y 4 bits restantes (respectivamente). En ambos casos, se realizó el cifrado parcial comenzando por los

bits menos significativos hasta los más significativos.

Para el primer caso de 3 bits, se cifran los bits 1, 2 y 3, dejando inalterados los bits 4, 5, 6, 7 y 8. Para el segundo caso, se cifran los bits 2, 3 y 4, dejando inalterados los bits 5, 6, 7, 8 y 1. Este proceso se realizó de manera iterativa 8 veces recorriendo la posición de los bits cifrados. Para el caso del cifrado parcial de 4 bits el proceso es el mismo.

3.1.1. Codependencia entre el procesado con el descifrado parcial

Como se puede apreciar en la sección 2.2, las ecuaciones para descifrar cada uno de los bits del *bloque de texto cifrado c*, están en función de los bits de la *llave de cifrado t*, y del *bloque de texto cifrado c*, (indicados por la matrices R y T). Esta cuestión sigue presente aún habiendo realizado el cifrado de manera parcial. Sin embargo con cifrado parcial, no se obtienen todos los bits del *bloque de texto cifrado c*, sino una parte debido a los bits que se dejan inalterados (estos últimos, son bits *procesados*). Por lo tanto al realizar el descifrado, no se cuenta con todos los bits necesarios del *bloque de texto cifrado* en las ecuaciones correspondientes. Para resolver este problema, se hace una sustitución en las ecuaciones de descifrado. En lugar de utilizar el bit del *bloque de texto cifrado* (con el cual, no se cuenta) se sustituye por la ecuación para obtener dicho bit, la cual está en función de la *llave de cifrado t*, y el *bloque procesado* $\hat{m}$, con los cuales si se cuentan. Por lo tanto, el procesado debe realizarse de manera completa (o en su defecto, no realizarse), para poder llevar a cabo el descifrado parcial de manera correcta. Las ecuaciones de descifrado parcial obtenidas (para $L = 8$) para 3 y 4 bits, con procesado completo y sin procesado alguno, se incluyen en el apéndice A.

3.2. Implementación en LabVIEW

Utilizando el paquete VISION de National Instruments, dentro del entorno LabVIEW, hacemos uso de las distintas herramientas para manipular los pixeles de una imagen RGB. Cada uno de estos pixeles es convertido a su respectivo valor numérico y posteriormente a un valor binario de 24 bits, los cuales son separados en 3 bloques de 8 bits. Cada bloque representa el valor del tono rojo, verde, y azul, y el cifrado parcial se aplica a estos 3 bloques por separado. Una vez hecho esto, se reagrupan para formar nuevamente un valor binario de 24 bits, se convierte a valor real y posteriormente a pixel, para formar una nueva imagen. Este proceso se realiza con todos los pixeles de la imagen.

3.2.1. Introducción de número aleatorio y semillas

Lo primero a realizar es escribir 2 palabras de 6 letras cada una. LabVIEW escanea cada una de estas letras y las convierte a un número real dependiendo de su equivalencia en el código ASCII. Estos números son posteriormente convertidos a un número binario de 5 bits. En total, se obtienen 12 números de 5 bits, dando un total de 60 bits; 27 de estos, se usarán para formar 3 *bloques aleatorios z*, de 9 bits cada uno; 15 se usarán como la *semilla x*, y 16 se usaran como *semilla y*. La figura 3.1 muestra el diagrama de bloques donde se realiza dicho proceso.

3.2.2. Carga de imagen

Una vez introducidas las semillas y los *bloques aleatorios* procedemos a cargar la imagen a cifrar. Una vez seleccionada, en LabVIEW se convierte cada pixel a un valor numérico y se agrupa en un arreglo de 2 dimensiones. Este arreglo entra a su vez a 2 *ciclos for*, lo cual

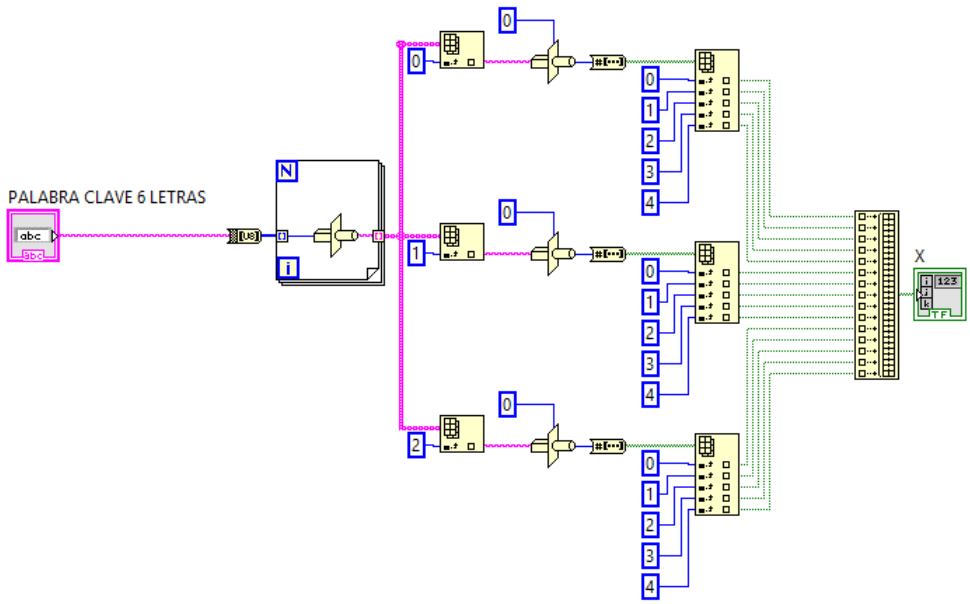

Figura 3.1: Diagrama de bloques para la introducción de la semilla x.

nos permite acceder a cada valor del arreglo de pixeles en orden. La figura 3.2 muestra el diagrama de bloques donde se realiza dicho proceso.

3.2.3. Procesado del bloque

Una vez que el valor numérico del pixel entra a ambos *ciclos for* se procede a manipularlo. Primero se realiza una conversión a un arreglo binario de 32 valores. Se toman los primeros 24 valores y se reagrupan en 3 arreglos de 8 bits cada uno. Estos serán los *bloques de texto en claro m*. Introducimos los *bloques aleatorios z*, y con ayuda de la operación lógica *xor*, uno a uno se realizan las conexiones necesarias entre los valores de m y z, de acuerdo a las ecuaciones ya antes mencionadas para este caso. Este proceso entrega los *bloques procesados* $\hat{m}$, y los *bloques aleatorios z*, para la siguiente iteración.

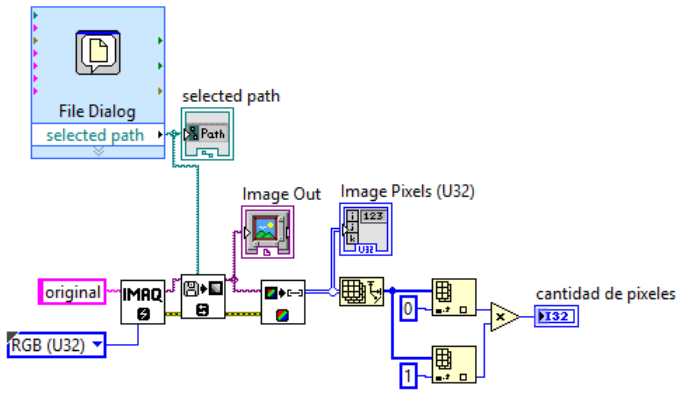

Figura 3.2: Diagrama de bloques para la carga de la imagen a cifrar.

3.2.4. Llave de cifrado

De la misma manera, se insertan los bloques binarios correspondientes a la *semilla x* y la *semilla y*. Con ayuda de la operación lógica *xor*, uno a uno se realizan las conexiones necesarias para obtener la *llave de cifrado t*. También se obtienen una nueva *semilla x* y una nueva *semilla y*, las cuales se usarán en la siguiente iteración.

3.2.5. Cifrado

Una vez obtenido el arreglo de bits ya procesados, y el arreglo de bits de la *llave de cifrado*, ambos ingresan a un sub-vi en el cual interactúan uno a uno en función de las ecuaciones de cifrado. Este sub-vi es un instrumento virtual que se crea dentro de LabVIEW y puede ser colocado dentro del entorno de otro instrumento virtual principal. Cuenta con conexiones de entrada y salida para trabajar en conjunto con dicho programa principal. La figura 3.3 muestra el diagrama de bloques de este proceso. Los bits resultantes se agrupan en orden en un nuevo arreglo. Este último es nuestro *bloque de texto cifrado*.

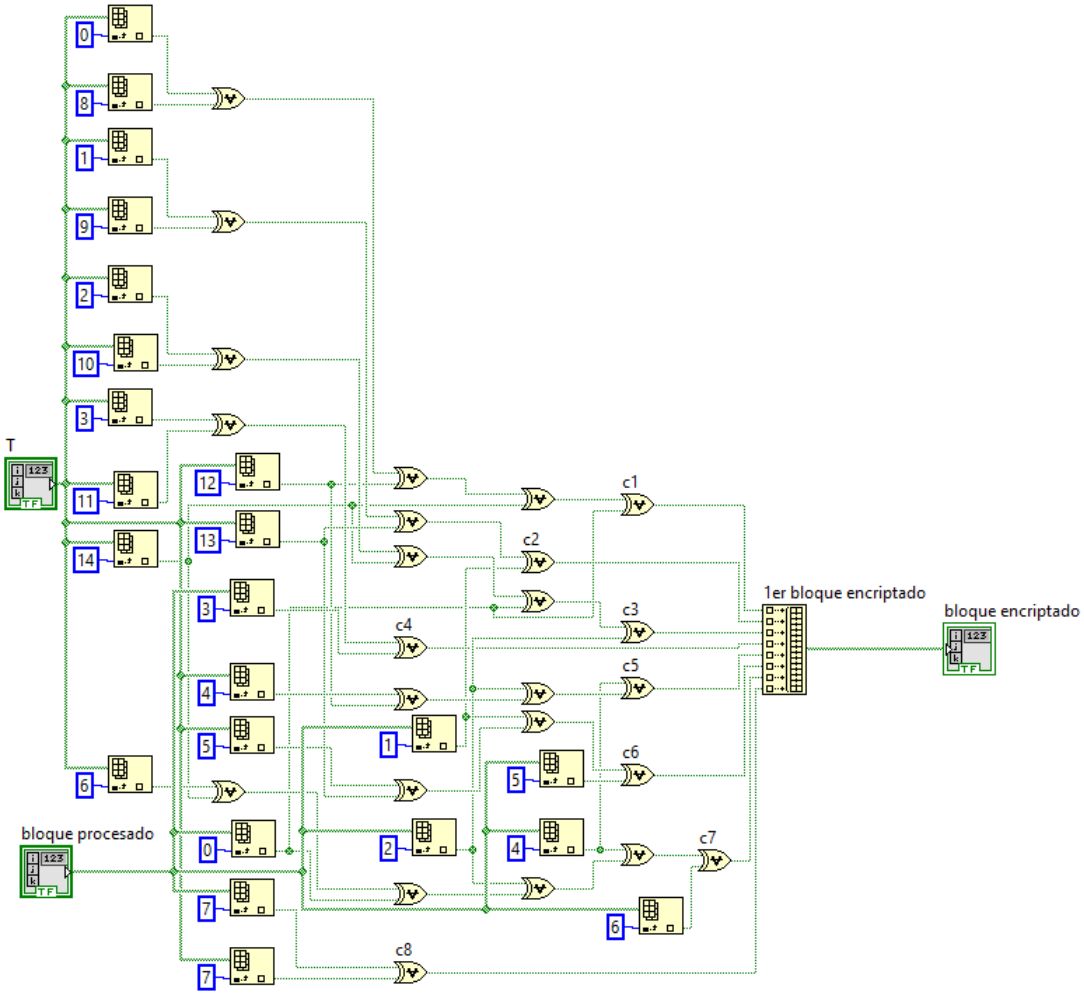

Figura 3.3: Diagrama de bloques del cifrado realizado al bloque procesado.

3.2.6. Reconstrucción de imagen

Una vez que se cuenta con los 3 *bloques de texto cifrado* se construye un arreglo con los 24 bits que lo conforman y los últimos 8 bits del pixel original. Ya que se tiene el arreglo binario de 32 bits, se transforma a número real y se reconstruye el pixel. La figura 3.4 muestra el proceso para reconstruir el pixel a partir del número indicado por el arreglo binario. Este proceso se repite para todos los pixeles de la imagen y se reconstruye una nueva imagen, ahora cifrada.

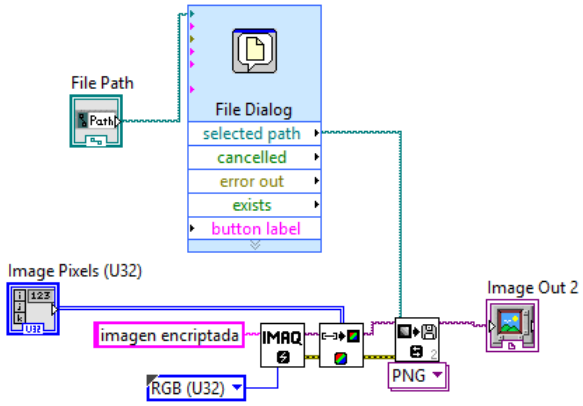

Figura 3.4: Diagrama de bloques de la reconstrucción del pixel.

3.3. Pruebas realizadas

Para medir la calidad y la robustez del sistema ESCA aplicado de manera parcial se ha evaluado con diferentes pruebas comunes. Se analiza la distribución de pixeles de las imágenes cifradas mediante histogramas, la correlación de pixeles entre las imágenes originales y las cifradas, la correlación entre los pixeles adyacentes en la imagen cifrada, y una prueba conocida como *choosen plain - image atack*, pruebas que describiremos de manera general.

3.3.1. Histograma

Un histograma muestra cómo se distribuyen los pixeles en una imagen trazando el número de estos mismos en cada nivel de intensidad de color. Si el histograma de una imagen cifrada tiene una distribución uniforme, entonces el cifrado aplicado fue capaz de ocultar la información de la imagen original. Se calcularon los histogramas a las diferentes imágenes a las cuales se les aplicó en cifrado parcial considerando los diferentes canales, rojo, verde, y azul. Las dimensiones de todas las imágenes fue de 512×512 pixeles. Como una breve ilustración de esta prueba, en la figura 3.5 se muestran los histogramas para la imagen original de Lena, mientras que en la figura 3.6 se muestran los histogramas de su versión cifrada de manera total. A partir de las figuras, se puede ver que los histogramas de las imágenes cifradas están uniformemente distribuidos, lo cual ilustra el efecto de cifrar, a diferencia de los respectivos histogramas de las imágenes planas.

Figura 3.5: Histogramas de los niveles Rojo, Verde, y Azul de la imagen original de Lena.

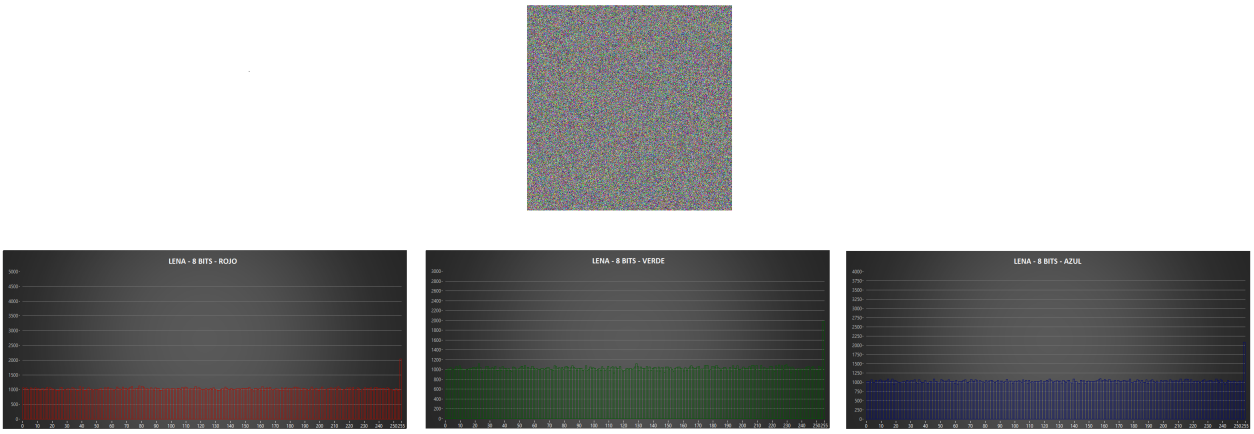

Figura 3.6: Histogramas de los niveles Rojo, Verde, y Azul de la imagen de Lena cifrada completamente.

3.3.2. Correlación

Para demostrar que la imagen cifrada es independiente de la imagen original se calcula el coeficiente de correlación entre ambas imágenes. Si el coeficiente es cercano a 0 sugiere que no hay correlación lineal o bien, que la correlación es muy débil. Por otra parte, si el coeficiente es cercano a 1 o -1, implicaría que las imágenes están fuertemente relacionadas, situación no deseable en las imágenes cifradas. Los coeficientes que se van a calcular corresponden a las combinaciones de las imágenes considerando los niveles rojo - rojo, rojo - verde, rojo - azul; verde - rojo, verde - verde, verde - azul; azul - rojo, azul - verde, y azul - azul.

3.3.3. Correlación Adyacente

Otra prueba estadística que se considera para la evaluación es la correlación adyacente de manera horizontal, vertical y diagonal, entre las imágenes originales y sus correspondientes versiones cifradas. Para esto, se seleccionan aleatoriamente 3000 pares de pixeles adyacentes

en cada dirección. En seguida, se calcula el coeficiente de correlación de cada par. Como ilustración, en la figura 3.7 se muestra la distribución del nivel rojo de los pixeles adyacentes en la dirección horizontal de la imagen Lena y su versión cifrada. Se puede observar que la correlación adyacente de la imagen plana en canal rojo se asemeja a una diagonal, lo cual implica una fuerte correlación, mientras que para la versión cifrada se observa una gran dispersión resultando en una muy débil correlación, situación bastante favorable en este caso.

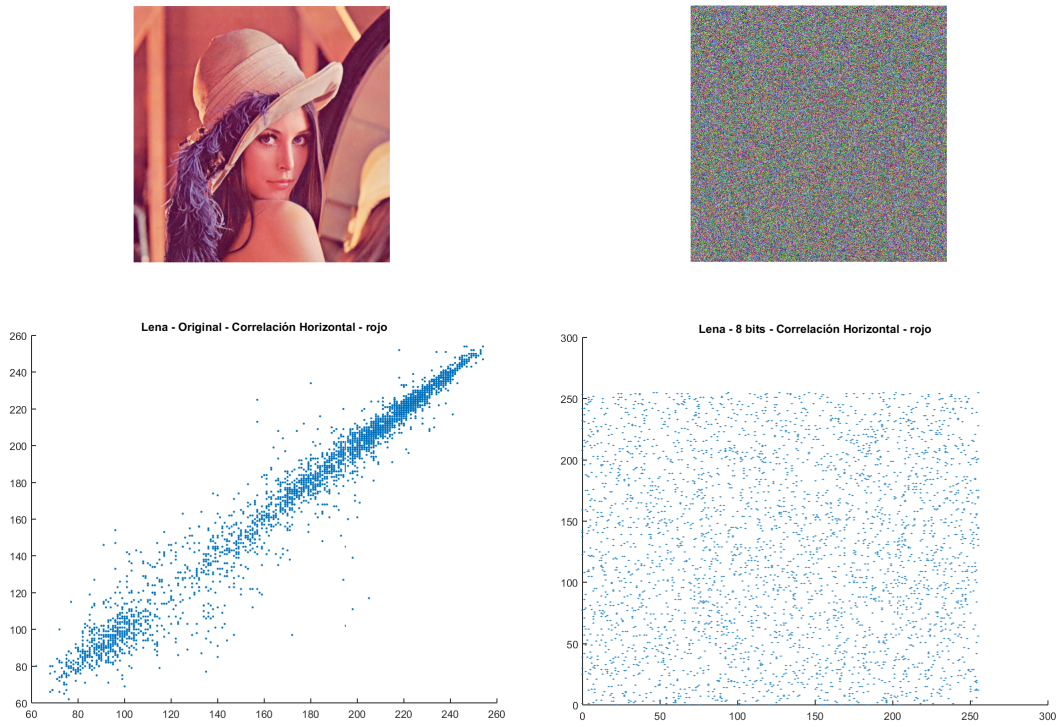

Figura 3.7: Correlación Horizontal del nivel Rojo, de la imagen original de Lena y la cifrada completamente.

3.3.4. Chosen-Plain Image Attack

La última prueba que nos auxilia a evaluar la seguridad de un criptosistema es la denominada prueba Chosen-Plain Image-Attack (CPIA).

Si un sistema de cifrado es capaz de soportar un ataque del tipo CPIA, entonces el sistema es también seguro ante los ataques criptoanalíticos denominados como ataque basado únicamente en la imagen cifrada (cipher-image-only-attack) y el ataque basado en una imagen dada y su versión cifrada correspondiente (known-plain image-attack). De hecho, el sistema de cifrado del ESCA presentaba debilidad ante tal ataque y fue hasta que se incorporó una etapa de pre-procesamiento en función de los elementos que conforman dicho sistema [3].

El procedimiento del CPIA consiste en que los intrusos pueden elegir algunas imágenes planas u homogéneas (los pixeles tienen el mismo valor), y obtener las las imágenes cifradas correspondientes, todas bajo las mismas condiciones de cifrado. Para realizar el ataque CPIA se siguen los siguientes pasos.

(a) En principio se considera una imagen plana ($\mathbf{I}_o$) y su respectiva versión cifrada ($\mathbf{I}_c$).

(b) El siguiente paso consiste en realizar la operación *xor* entre una imagen homogénea con su respectiva versión cifrada. Al resultado de esta operación se le conoce como enmascaramiento ($\mathbf{I}_m$).

(c) Finalmente, se realiza la operación *xor* entre las imágenes cifradas de la imagen original y la de la máscara, es decir, $\mathbf{I}_c \bigoplus \mathbf{I}_m$. Si la imagen resultante revela información de la imagen original, se tiene que el sistema de cifrado es inseguro.

Capítulo 4

Resultados

En este capítulo se presentan los resultados obtenidos de todas las pruebas realizadas, a las imágenes cifradas parcialmente considerando cifrar 3 y 4 bits, con procesado y sin procesado. La primera prueba presentada son los histogramas RGB para las 8 posiciones de cifrado parcial.

Posteriormente, se presenta la prueba de correlación lineal y la correlación adyacente para las 8 posiciones del cifrado parcial, con y sin la etapa de procesado. Por último, se presentan los resultados obtenidos de la prueba Image - Plain Attack se presenta para las 8 posiciones, y los 4 cifrados diferentes.

4.1. Histogramas

En esta sección, se presentan los histogramas obtenidos de las imágenes cifradas parcialmente considerando o no la etapa de pre-procesado en el sistema de cifrado.

En la figura 4.1 se ilustra el caso donde únicamente se cifraron 3 bits sin haber realizado el procesado previo. El cifrado se llevó a cabo partiendo desde los 3 menos significativos y haciendo un barrido en pasos de 1 bit hasta los más significativos. Las columnas 2, 3 y 4 muestran los histogramas de los canales rojo, verde y azul, respectivamente, de las imágenes cifradas. Se puede observar que algunas de las imágenes cifradas si revelan información de la imagen original. Además, los histogramas correspondientes a las imágenes cifradas considerando bits más significativos, como en el caso de la posición 6 donde el cifrado se aplico a los bits $b_8 b_7 b_6$, se presenta una distribución "más uniforme". Situación contraria con las posiciones donde se consideran los bits menos significativos.

De manera similar la figura 4.2 ilustra el caso donde se cifraron 4 bits sin haber realizado el procesado previo. En estos se observa que los histogramas más uniformes son los obtenidos de las imágenes donde el cifrado se llevó a cabo en los bits más significativos, siendo la posición 5 la que obtiene los mejores resultados.

Las figuras 4.3 y 4.4 ilustran los histogramas de las imágenes donde se cifraron 3 y 4 bits respectivamente, habiendo realizado el procesado previo. Estos histogramas muestran uniformidad para todos los casos posibles, tanto en el cifrado de los bits menos significativos como en los más significativos. Estos resultados indican el alto nivel de seguridad que proporciona el procesado previo al cifrado.

Histogramas - 3 bits - Sin procesado

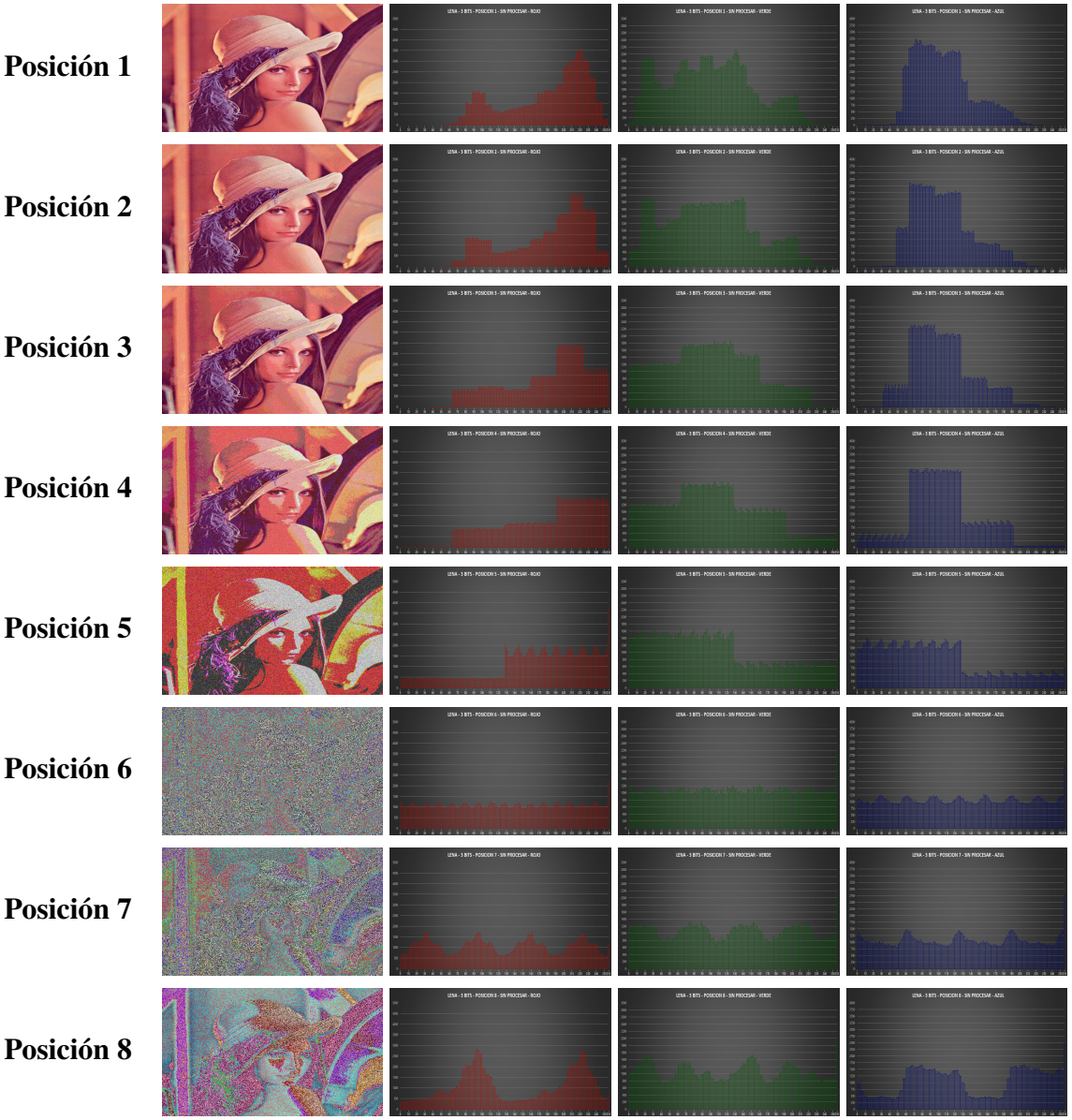

Figura 4.1: La primer columna muestra la imagen cifrada. Las columnas 2, 3 y 4 muestran los histogramas de los canales rojo, azul, y verde de las imágenes cifradas. El orden de los bits cifrados aumenta de manera descendente comenzando por los menos significativos.

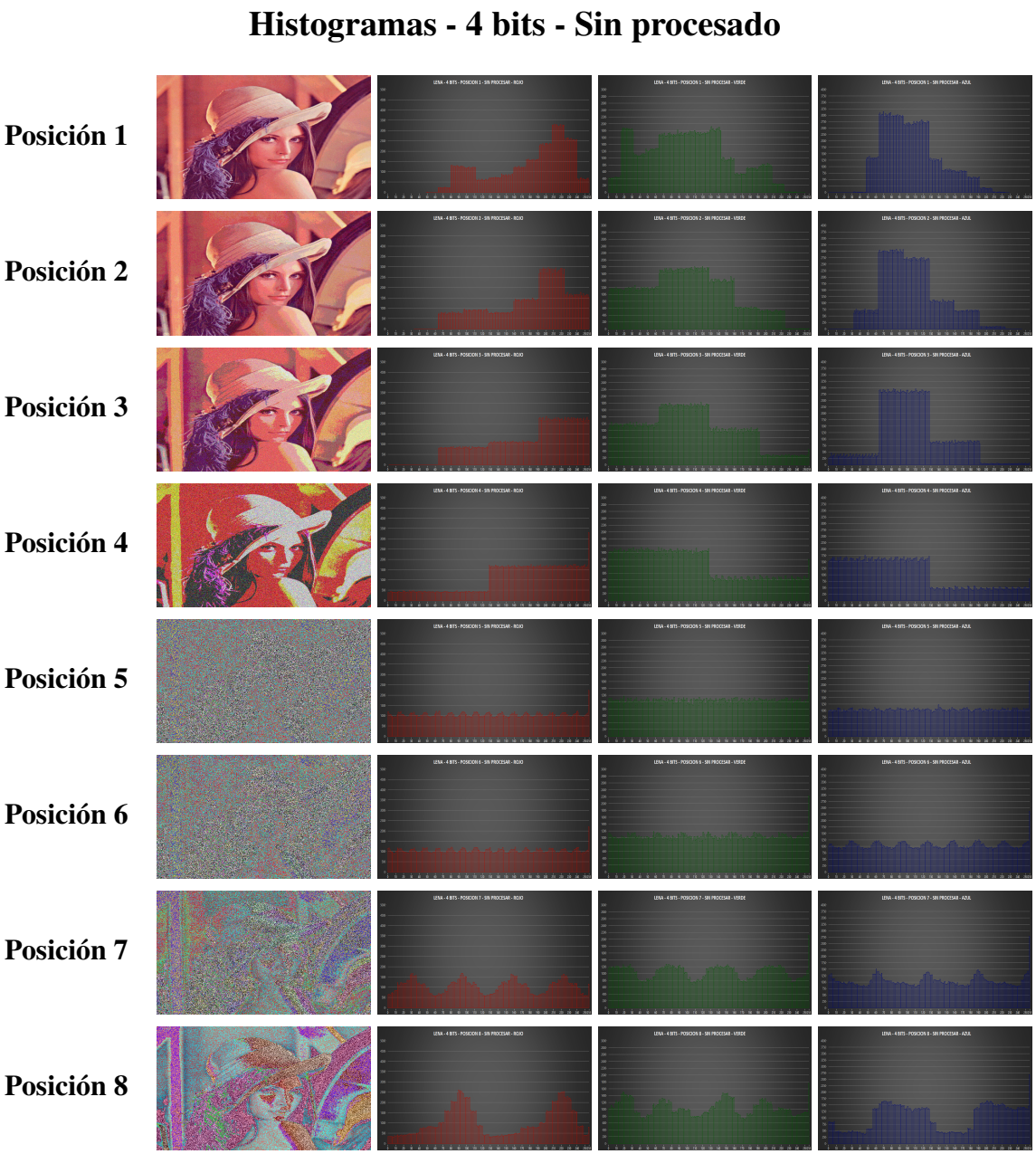

Figura 4.2: La primer columna muestra la imagen cifrada. Las columnas 2, 3 y 4 muestran los histogramas de los canales rojo, azul, y verde de las imágenes cifradas. El orden de los bits cifrados aumenta de manera descendente comenzando por los menos significativos.

Histogramas - 3 bits - Con procesado

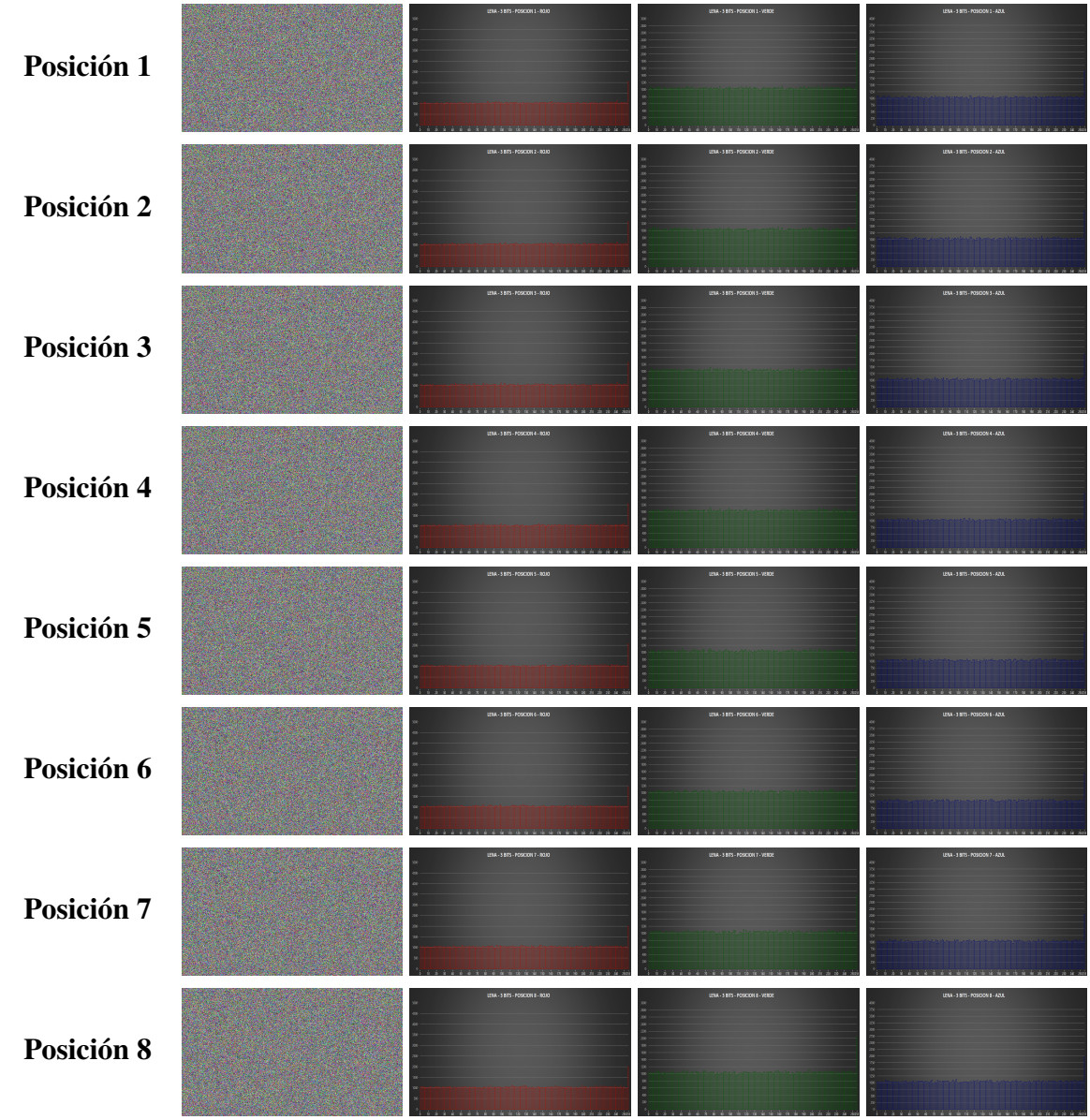

Figura 4.3: La primer columna muestra la imagen cifrada. Las columnas 2, 3 y 4 muestran los histogramas de los canales rojo, azul, y verde de las imágenes cifradas. El orden de los bits cifrados aumenta de manera descendente comenzando por los menos significativos.

Histogramas - 4 bits - Con procesado

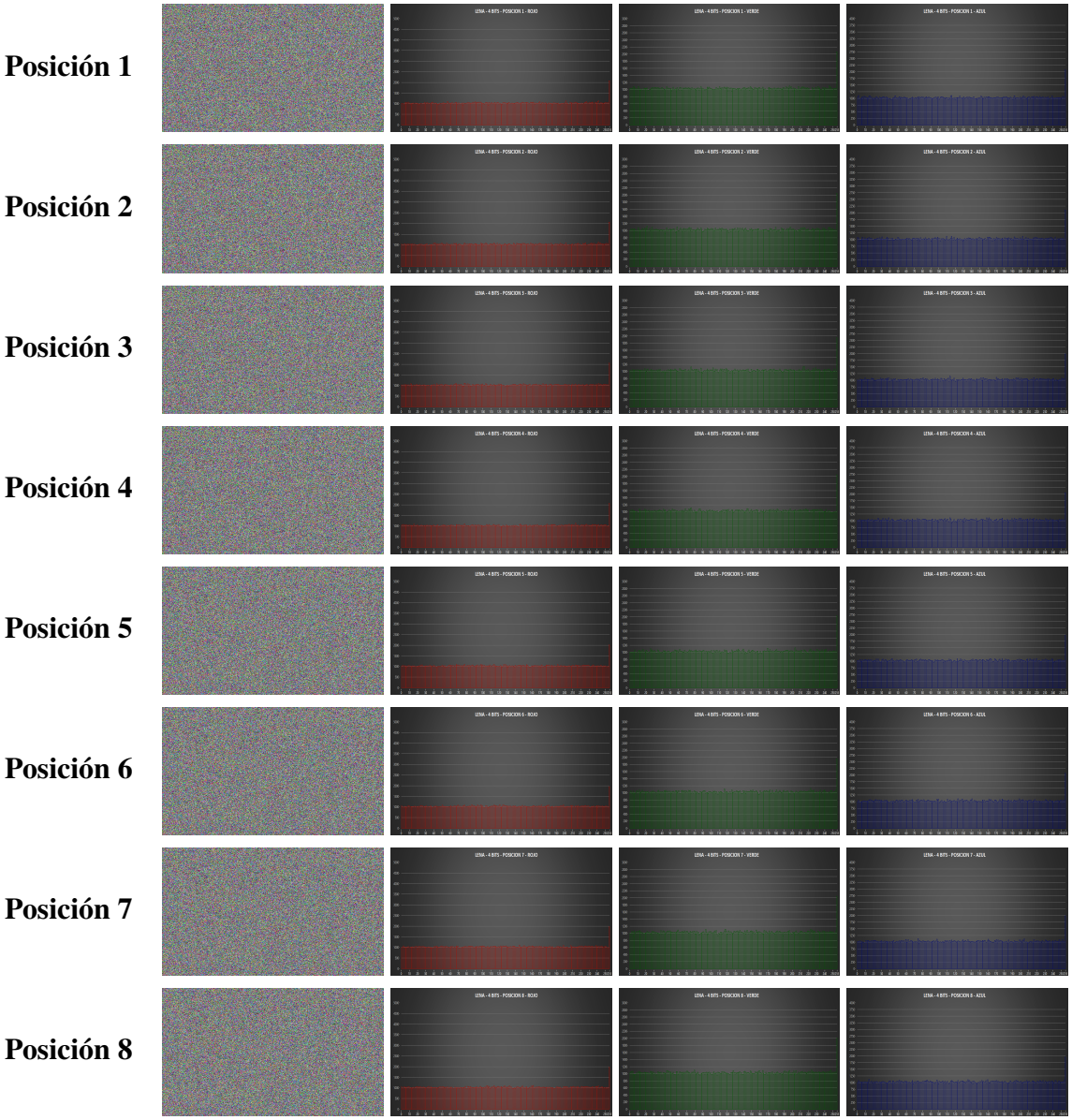

Figura 4.4: La primer columna muestra la imagen cifrada. Las columnas 2, 3 y 4 muestran los histogramas de los canales rojo, azul, y verde de las imágenes cifradas. El orden de los bits cifrados aumenta de manera descendente comenzando por los menos significativos.

4.2. Correlación

En esta sección, se presentan los niveles de correlación obtenidos de las imágenes cifradas parcialmente considerando o no la etapa de pre-procesado en el sistema de cifrado.

Las tablas 4.1 y 4.2 muestran los resultados para los casos en los cuales se cifraron 3 y 4 bits respectivamente, sin haber realizado el procesado previo. Se puede apreciar que para el caso de 3 bits, es la sexta posición, en la cual se cifraron los bits $b_8b_7b_6$, la que muestra los niveles más bajos de correlación, siendo hasta cien veces menores en comparación a los casos donde se cifran los bits menos significativos. En el caso de 4 bits, es la quinta posición, en la cual se cifraron los bits $b_8b_7b_6b_5$, la que muestra la correlación más baja, siendo hasta mil veces menor en comparación a los casos donde se cifran los bits menos significativos.

Las tablas 4.3 y 4.4 muestran los resultados para los casos en los cuales se cifraron 3 y 4 bits respectivamente, habiendo realizado el procesado previo. Se puede apreciar en ambas tablas que no hay una tendencia de aumento o decrecimiento en los resultados conforme cambia la posición de los bits cifrados, tanto en el caso de 3 bits como en el de 4 bits. Sin embargo, es claro que los valores obtenidos son cien o hasta mil veces menores a los casos sin procesado, incluso cifrando los bits menos significativos.

CANAL ROJO	CANAL VERDE	CANAL AZUL
Bits en posición 1		
$R-R: 0.9978109$	$G-R: 0.8766927$	$B-R: 0.9954808$
$R-G: 0.876974$	$G-G: 0.9981151$	$B-G: 0.9089930$
$R-B: 0.6734274$	$G-B: 0.9065232$	$B-B: 0.9954808$
Bits en posición 2		
$R-R: 0.9910648$	$G-R: 0.87180133$	$B-R: 0.6722399$
$R-G: 0.8723484$	$G-G: 0.9925243$	$B-G: 0.9039063$
$R-B: 0.664007$	$G-B: 0.8940387$	$B-B: 0.9823430$
Bits en posición 3		
$R-R: 0.9670444$	$G-R: 0.8491251$	$B-R: 0.9318062$
$R-G: 0.8569542$	$G-G: 0.9714854$	$B-G: 0.6526818$
$R-B: 0.6272589$	$G-B: 0.8447761$	$B-B: 0.8843049$
Bits en posición 4		
$R-R: 0.8927759$	$G-R: 0.7708962$	$B-R: 0.5952364$
$R-G: 0.7793781$	$G-G: 0.8975262$	$B-G: 0.8138184$
$R-B: 0.4920190$	$G-B: 0.6697437$	$B-B: 0.7527694$
Bits en posición 5		
$R-R: 0.6799582$	$G-R: 0.5436427$	$B-R: 0.4191530$
$R-G: 0.4949712$	$G-G: 0.6738060$	$B-G: 0.6244588$
$R-B: 0.3166857$	$G-B: 0.5523309$	$B-B: 0.6584760$
Bits en posición 6		
$R-R: -0.0032519$	$G-R: -0.0022971$	$B-R: -0.0005652$
$R-G: -0.0090390$	$G-G: -0.0090433$	$B-G: -0.0076014$
$R-B: 0.0044932$	$G-B: 0.0042350$	$B-B: 0.0019395$
Bits en posición 7		
$R-R: 0.0142268$	$G-R: 0.0226139$	$B-R: 0.0165801$
$R-G: 0.0096823$	$G-G: 0.0010747$	$B-G: 0.0041021$
$R-B: 0.0455671$	$G-B: 0.0557911$	$B-B: 0.0512491$
Bits en posición 8		
$R-R: 0.0877989$	$G-R: 0.0836285$	$B-R: 0.0954348$
$R-G: 0.1658272$	$G-G: -0.0640482$	$B-G: 0.0662984$
$R-B: 0.0372403$	$G-B: 0.1250094$	$B-B: -0.1176474$

Tabla 4.1: Correlación - 3 bits -sin procesado.

CANAL ROJO	CANAL VERDE	CANAL AZUL
Bits en posición 1		
$R-R: 0.9909502$	$G-R: 0.8717090$	$B-R: 0.6721859$
$R-G: 0.8722767$	$G-G: 0.9924452$	$B-G: 0.9038355$
$R-B: 0.6638348$	$G-B: 0.8938146$	$B-B: 0.9821311$
Bits en posición 2		
$R-R: 0.9666426$	$G-R: 0.8487654$	$B-R: 0.6524207$
$R-G: 0.8566652$	$G-G: 0.9711792$	$B-G: 0.8840497$
$R-B: 0.6267032$	$G-B: 0.8440309$	$B-B: 0.9309977$
Bits en posición 3		
$R-R: 0.8920117$	$G-R: 0.7702153$	$B-R: 0.5947865$
$R-G: 0.7783624$	$G-G: 0.8963255$	$B-G: 0.8127624$
$R-B: 0.4903973$	$G-B: 0.6673860$	$B-B: 0.7500895$
Bits en posición 4		
$R-R: 0.6785973$	$G-R: 0.5432738$	$B-R: 0.4196814$
$R-G: 0.4933432$	$G-G: 0.6714490$	$B-G: 0.6222909$
$R-B: 0.3152922$	$G-B: 0.5499455$	$B-B: 0.6559258$
Bits en posición 5		
$R-R: 0.0006313$	$G-R: 0.000048106418$	$B-R: -0.0001749$
$R-G: 0.0018308$	$G-G: 0.0007101$	$B-G: 0.0009700$
$R-B: 0.0012614$	$G-B: 0.000012388724$	$B-B: -0.0009523$
Bits en posición 6		
$R-R: -0.0032474$	$G-R: -0.0022875$	$B-R: -0.0005462$
$R-G: -0.0090454$	$G-G: 0.0042350$	$B-G: -0.0076055$
$R-B: 0.0044859$	$G-B: 0.0042260$	$B-B: 0.0019458$
Bits en posición 7		
$R-R: 0.0142530$	$G-R: 0.0226323$	$B-R: 0.0166050$
$R-G: 0.0097536$	$G-G: 0.0011720$	$B-G: 0.0042080$
$R-B: 0.0455461$	$G-B: 0.0557643$	$B-B: 0.0512263$
Bits en posición 8		
$R-R: 0.0875710$	$G-R: 0.1247674$	$B-R: 0.0953067$
$R-G: 0.1658067$	$G-G: 0.0835736$	$B-G: 0.0662741$
$R-B: 0.0372492$	$G-B: -0.0641362$	$B-B: -0.1177698$

Tabla 4.2: Correlación - 4 bits - sin procesado.

CANAL ROJO	CANAL VERDE	CANAL AZUL
Bits en posición 1		
$R-R:-0.0003365$ $R-G:-0.0019029$ $R-B:0.0044566$	$G-R:-0.0029895$ $G-G:-0.0009285$ $G-B:0.0065718$	$B-R:-0.0034849$ $B-G:-0.0011767$ $B-B:0.0063843$
Bits en posición 2		
$R-R:-0.0005647$ $R-G:-0.0020053$ $R-B:0.0042912$	$G-R:-0.0031594$ $G-G:-0.0010534$ $G-B:0.0064296$	$B-R:-0.0036236$ $B-G:-0.0012780$ $B-B:0.0063403$
Bits en posición 3		
$R-R:-0.0005612$ $R-G:-0.0020298$ $R-B:0.0043280$	$G-R:-0.0033862$ $G-G:-0.0011452$ $G-B:0.0063095$	$B-R:-0.0040338$ $B-G:-0.0012876$ $B-B:0.0061722$
Bits en posición 4		
$R-R:-0.0002015$ $R-G:-0.0020915$ $R-B:0.0046293$	$G-R:-0.0028088$ $G-G:-0.0011764$ $G-B:0.0064732$	$B-R:-0.0032300$ $B-G:-0.0009326$ $B-B:0.0064174$
Bits en posición 5		
$R-R:0.0004350$ $R-G:-0.0032426$ $R-B:0.0043879$	$G-R:-0.0014274$ $G-G:-0.0022537$ $G-B:0.0045326$	$B-R:-0.0012499$ $B-G:-0.0018267$ $B-B:0.0041201$
Bits en posición 6		
$R-R:-0.0022163$ $R-G:0.0006335$ $R-B:-0.0021579$	$G-R:-0.0008097$ $G-G:0.0011106$ $G-B:-0.0027662$	$B-R:0.0007034$ $B-G:0.0008054$ $B-B:-0.0022557$
Bits en posición 7		
$R-R:-0.0024703$ $R-G:0.0006687$ $R-B:-0.0024780$	$G-R:-0.0013400$ $G-G:0.0010683$ $G-B:-0.0030119$	$B-R:-0.000074355246$ $B-G:0.0003427$ $B-B:-0.0025886$
Bits en posición 8		
$R-R:-0.0029286$ $R-G:0.0019227$ $R-B:-0.0020670$	$G-R:-0.0026030$ $G-G:0.0022986$ $G-B:-0.0009272$	$B-R:-0.0019624$ $B-G:0.0013926$ $B-B:-0.0002492$

Tabla 4.3: Correlación - 3 bits - con procesado.

CANAL ROJO	CANAL VERDE	CANAL AZUL
Bits en posición 1		
$R-R:-0.0005695$	$G-R:-0.0031799$	$B-R:-0.0036396$
$R-G:-0.0020154$	$G-G:-0.001079$	$B-G:-0.0013077$
$R-B:0.0042895$	$G-B:0.0064308$	$B-B:0.0063518$
Bits en posición 2		
$R-R:-0.0006378$	$G-R:-0.0034616$	$B-R:-0.0040976$
$R-G:-0.0020347$	$G-G:-0.001138$	$B-G:-0.0012595$
$R-B:0.0043287$	$G-B:0.0063092$	$B-B:0.00631785$
Bits en posición 3		
$R-R:-0.0003127$	$G-R:-0.0028771$	$B-R:-0.0032729$
$R-G:-0.0020762$	$G-G:-0.0011290$	$B-G:-0.000855$
$R-B:0.0046471$	$G-B:0.00635575$	$B-B:0.0065177$
Bits en posición 4		
$R-R:0.0002026$	$G-R:-0.0016173$	$B-R:-0.0014041$
$R-G:-0.0033553$	$G-G:-0.0024045$	$B-G:-0.0019579$
$R-B:0.0042206$	$G-B:0.0043912$	$B-B:0.0040872$
Bits en posición 5		
$R-R:-0.002289$	$G-R:-0.0028889$	$B-R:0.0002269$
$R-G:0.0006056$	$G-G:0.0010273$	$B-G:0.0008251$
$R-B:-0.0021216$	$G-B:-0.0011136$	$B-B:-0.0024200$
Bits en posición 6		
$R-R:-0.002221$	$G-R:-0.0008302$	$B-R:0.0006874$
$R-G:0.0006234$	$G-G:0.0010856$	$B-G:0.0007756$
$R-B:-0.0021594$	$G-B:-0.0027648$	$B-B:-0.0022440$
Bits en posición 7		
$R-R:-0.0025471$	$G-R:-0.00141560$	$B-R:-0.0001382$
$R-G:0.0006639$	$G-G:0.0010755$	$B-G:0.0003709$
$R-B:-0.0024772$	$G-B:-0.0030120$	$B-B:-0.0025822$
Bits en posición 8		
$R-R:-0.0030598$	$G-R:-0.0026714$	$B-R:-0.0020054$
$R-G:0.0019378$	$G-G:0.0023457$	$B-G:0.0014699$
$R-B:-0.0020491$	$G-B:-0.0008428$	$B-B:-0.0001488$

Tabla 4.4: Correlación - 4 bits - con procesado

4.3. Correlación Adyacente

En esta sección, se presentan las correlaciones adyacentes obtenidas de las imágenes cifradas parcialmente considerando o no la etapa de pre-procesado en el sistema de cifrado.

En la figura 4.5 se ilustra los casos donde únicamente se cifraron 3 bits sin haber realizado el procesado previo. El cifrado se llevó a cabo partiendo desde los 3 menos significativos y haciendo un barrido en pasos de 1 bit hasta los más significativos. Las columnas 2, 3 y 4 muestran la distribución de los niveles rojo, verde y azul de los pixeles adyacentes en la dirección horizontal de dichas imágenes cifradas. Se puede observar que para los casos donde se cifraron los bits menos significativos, la correlación se asemeja a una diagonal, lo cual implica una fuerte correlación. Esta diagonal se uniformiza conforme los bits cifrados son más significativos, observando una gran dispersión para el caso de la posición 6. La figura 4.6 corresponde a los resultados de la correlación vertical, para el caso de 4 bits sin procesado previo. Fácilmente se observa que en la posición 5 se obtiene una correlación completamente uniforme a diferencia de los casos donde se cifraron los bits menos significativos Las figuras 4.7 y 4.8 muestran los resultados de la correlación diagonal correspondiente a los casos de 3 y 4 bits respectivamente habiendo realizado el procesado previo. Es claro que en todos los resultados la correlación es completamente uniforme sin importar la posición de los bits cifrados. Las tablas 4.5, 4.6, 4.7 y 4.8 muestran los datos numéricos correspondientes a las correlaciones obtenidas.

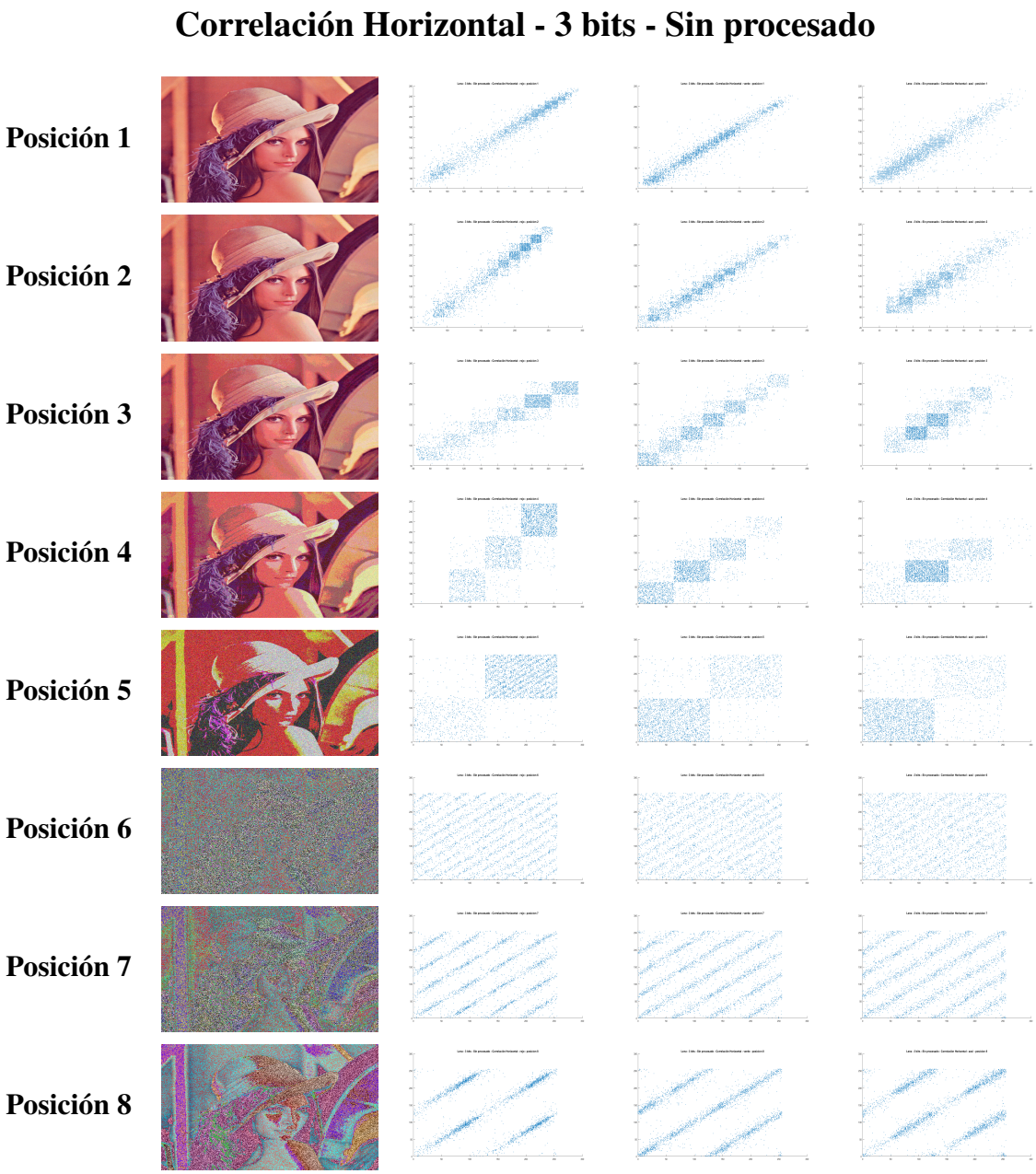

Figura 4.5: La primera columna muestra la imagen cifrada parcialmente, sin haber realizado el procesado previo. Las columnas 2, 3 y 4 muestran la correlación horizontal de los canales Rojo, Verde y Azul, respectivamente. Cada fila representa una posición de los 3 bits cifrados. El orden cambia de manera descendente.

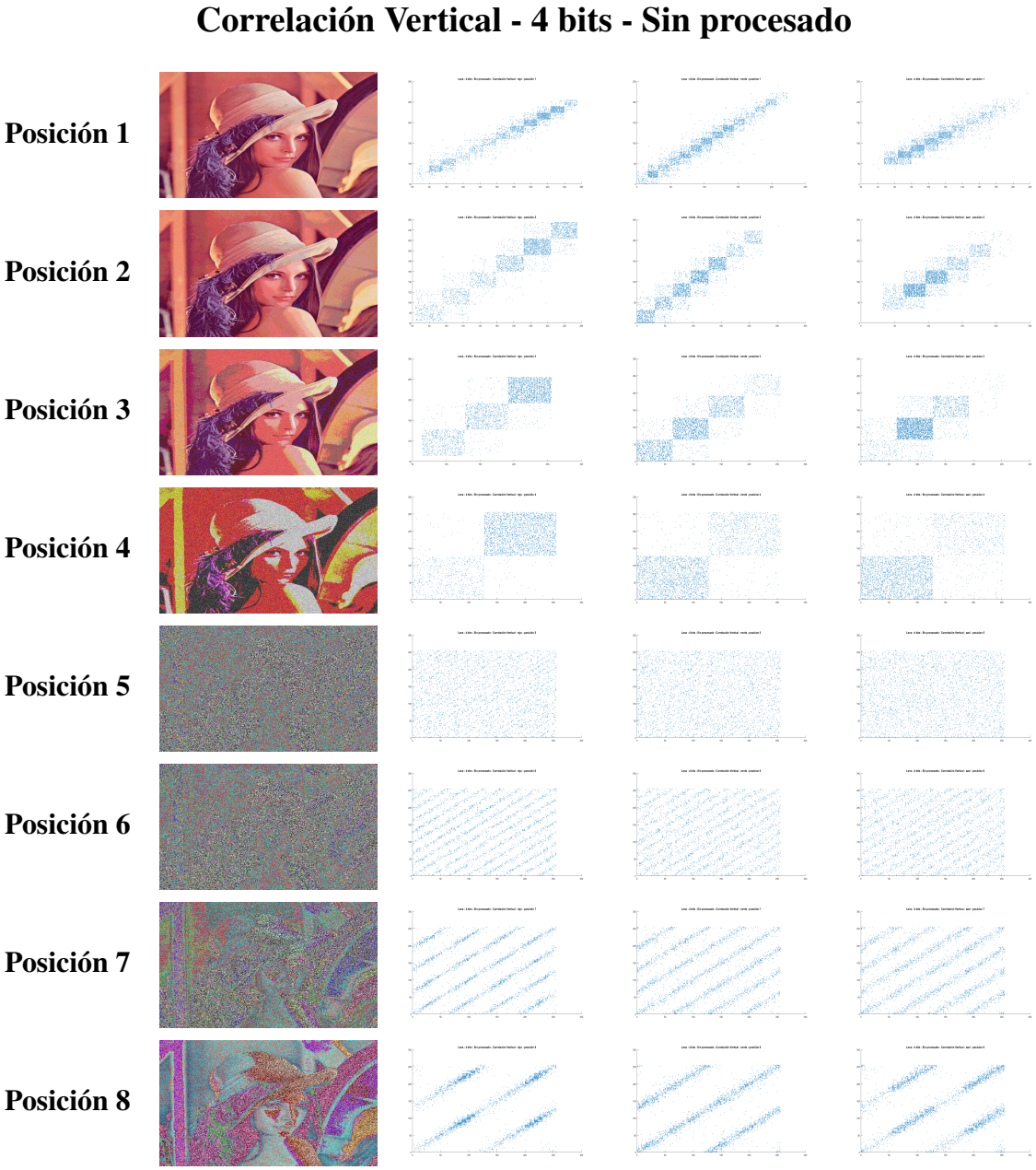

Figura 4.6: La primera columna muestra la imagen cifrada parcialmente, sin haber realizado el procesado previo. Las columnas 2, 3 y 4 muestran la correlación vertical de los canales Rojo, Verde y Azul, respectivamente. Cada fila representa una posición de los 4 bits cifrados. El orden cambia de manera descendente.

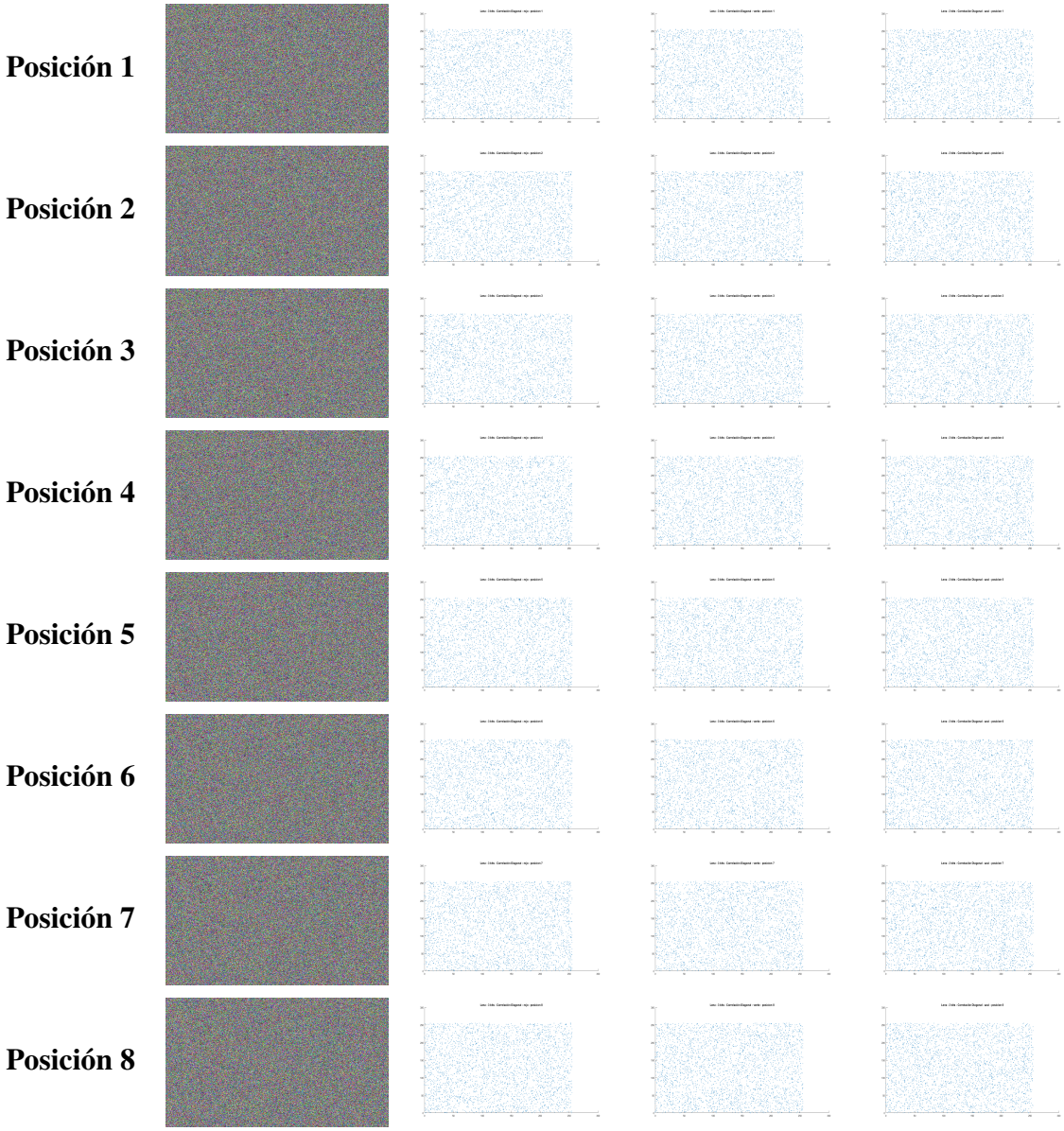

Figura 4.7: La primera columna muestra la imagen cifrada parcialmente, habiendo realizado el procesado previo. Las columnas 2, 3 y 4 muestran la correlación diagonal de los canales Rojo, Verde y Azul, respectivamente. Cada fila representa una posición de los 3 bits cifrados. El orden cambia de manera descendente.

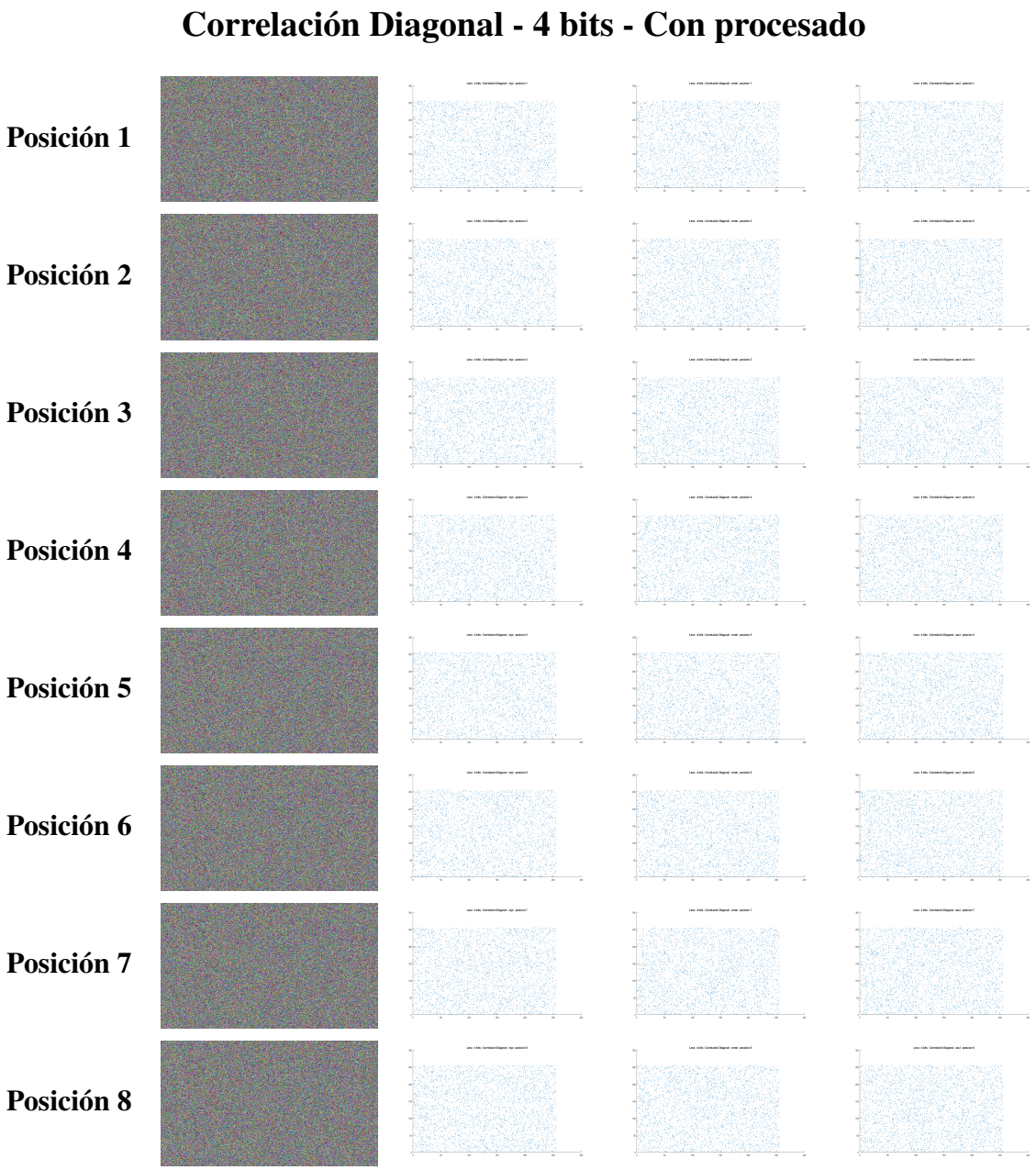

Figura 4.8: La primera columna muestra la imagen cifrada parcialmente, habiendo realizado el procesado previo. Las columnas 2, 3 y 4 muestran la correlación diagonal de los canales Rojo, Verde y Azul, respectivamente. Cada fila representa una posición de los 4 bits cifrados. El orden cambia de manera descendente.

POSICIÓN	CANAL ROJO	CANAL VERDE	CANAL AZUL
1	0.9744	0.9638	0.9265
2	0.9631	0.9570	0.8953
3	0.9294	0.9173	0.8166
4	0.8321	0.8333	0.6028
5	0.6034	0.6441	0.5150
6	0.0206	0.0024	-0.0161
7	0.0367	0.0239	0.0358
8	0.1473	0.1910	0.1223

Tabla 4.5: Correlación Horizontal de Lena, con cifrado parcial de 3 bits sin procesado.

POSICIÓN	CANAL ROJO	CANAL VERDE	CANAL AZUL
1	0.9739	0.90668	0.9298
2	0.9374	0.9388	0.8556
3	0.8491	0.8539	0.6359
4	0.6177	0.6568	0.5555
5	-0.0218	-0.0285	0.0035
6	0.0317	0.0315	0.0225
7	0.0150	0.0462	0.0404
8	0.1575	0.1864	0.1341

Tabla 4.6: Correlación Vertical de Lena, con cifrado parcial de 4 bits sin procesado.

POSICIÓN	CANAL ROJO	CANAL VERDE	CANAL AZUL
1	-0.0149	0.0030	-0.0018
2	-0.0046	0.0121	-0.0145
3	-0.0185	0.0083	-0.0209
4	-0.0345	0.0167	0.0024
5	0.0393	-0.0222	0.0029
6	0.0161	-0.0061	0.0082
7	-0.0032	-0.0015	-0.0188
8	0.0072	0.0054	-0.0233

Tabla 4.7: Correlación Diagonal de Lena, con cifrado parcial de 3 bits con procesado.

POSICIÓN	CANAL ROJO	CANAL VERDE	CANAL AZUL
1	0.0474	-0.0090	-0.0142
2	-0.00011237	0.0150	-0.0065
3	0.0165	-0.0049	0.0221
4	0.0020	0.0092	0.0158
5	-0.0226	-0.0127	-0.0339
6	0.00031504	-0.0340	-0.0298
7	0.0040	0.0365	0.0083
8	-0.0142	-0.0265	-0.0098

Tabla 4.8: Correlación Diagonal de Lena, con cifrado parcial de 4 bits con procesado.

4.4. Chosen-Image Plain Attack

En esta sección, se presentan los resultados de la prueba Chosen-Image Plain Attack (CPIA) de las imágenes cifradas parcialmente, considerando la etapa sin procesado y con procesado en el esquema de cifrado. En la figura 4.9 se ilustran los resultados del CPIA aplicado a la imagen de Lena cifrada parcialmente, considerando tres y cuatro bits (respectivamente), y sin haber realizado el procesado previo. Las imágenes de la primer columna de imágenes de la figura corresponden a las imágenes de Lena cifrada parcialmente ($\mathbf{I}_c$'s), considerando solo tres bits. La posición indica el inicio para ubicar los bits considerados, donde la posición 1 corresponde al bit menos significativo, mientras que la posición 8 al bit más significativo. En la segunda columna de imágenes se encuentran las versiones de las imágenes de enmascaramiento, las cuales fueron cifradas sólo considerando los mismos bits y en la misma posición que las $\mathbf{I}_m$'s. En la última columna se tienen los resultados obtenidos de aplicar la operación $\mathbf{I}_c \bigoplus \mathbf{I}_m$, donde se puede observar que se revela información de la imagen original, por tanto en este escenario el sistema de cifrado es inseguro. Una situación similar se aprecia en la figura 4.10 correspondiente al caso de cifrado parcial de 4 bits sin el procesado, en el cual la prueba reveló información de la imagen original. Esto indica que el cifrado parcial sin procesado, tanto para 3 bits como para 4 bits, no es lo suficientemente seguro ante este ataque en ninguna de las posiciones. Las figuras 4.11 y 4.12 muestran los resultados del CPIA aplicado a la imagen de Lena cifrada parcialmente, considerando tres y cuatro bits, respectivamente, donde la etapa de pre-procesado es considerado en el cifrado. Es fácil apreciar que la prueba no reveló información visual de la imagen en ninguno de los casos, sin importar la posición de los bits cifrados, indicando el alto grado de seguridad que proporciona el procesado.

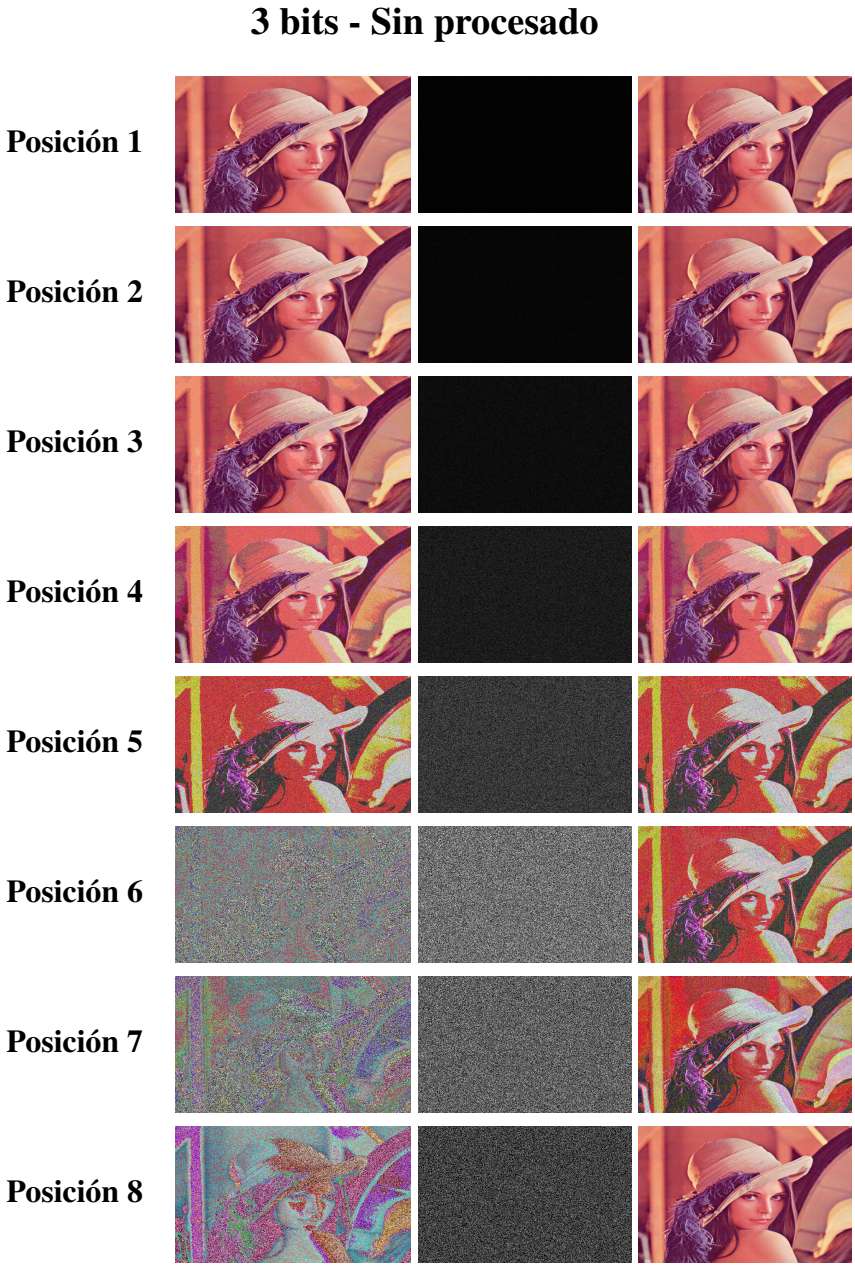

Figura 4.9: La primera columna muestra la imagen cifrada parcialmente, considerando 3 bits y sin el procesado previo. La columna 2 muestra la máscara cifrada parcialmente con las mismas llaves, y los mismos bits en la misma posición. La columna 3 muestran el resultado de la prueba, al aplicar xor entre ambas imágenes. La fila 1 corresponde al cifrado parcial realizado en los 3 bits menos significativos, mientras que en la fila 8 el cifrado parcial considera los bits b_8, b_1, b_2.

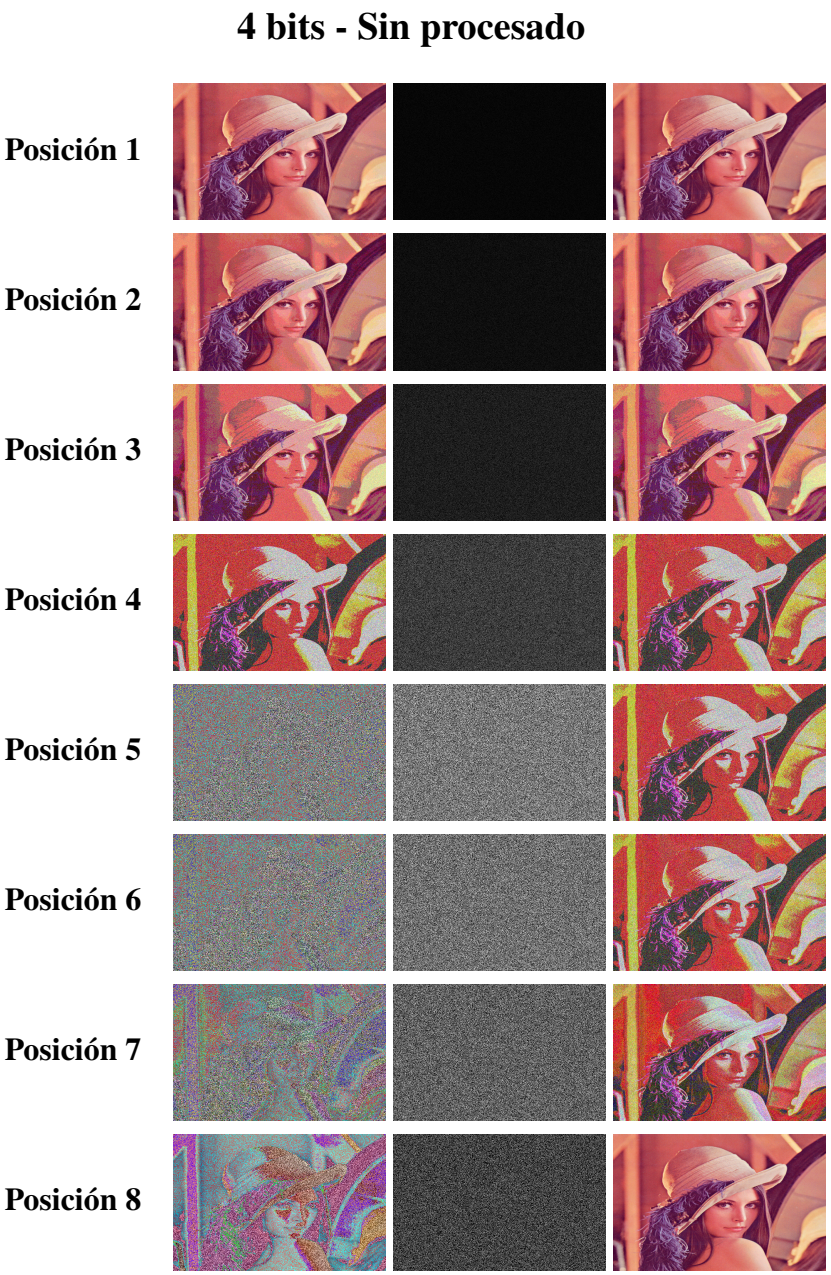

Figura 4.10: La primera columna muestra la imagen cifrada parcialmente, considerando 4 bits y sin el procesado previo. La columna 2 muestra la máscara cifrada parcialmente con las mismas llaves, y los mismos bits en la misma posición. La columna 3 muestran el resultado de la prueba.

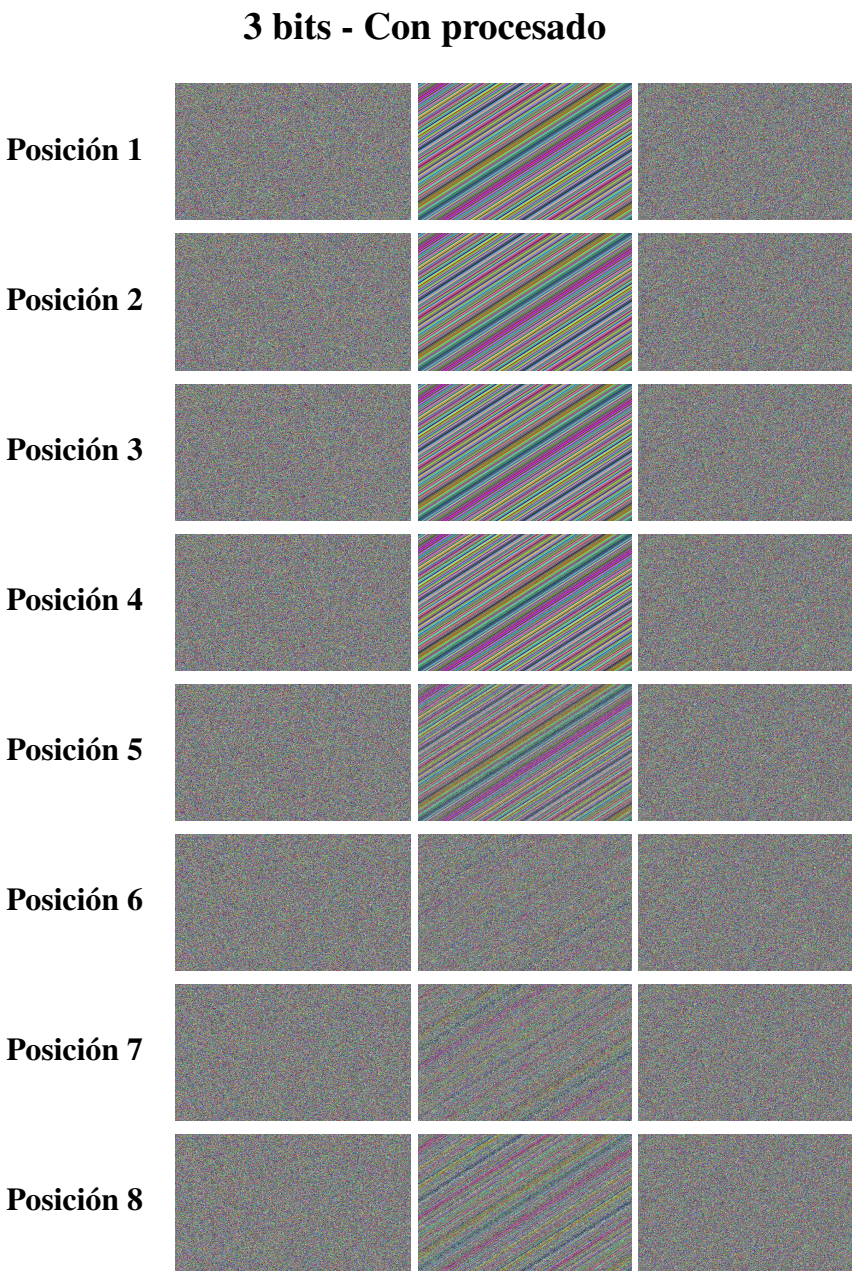

Figura 4.11: La primera columna muestra la imagen cifrada parcialmente, considerando 3 bits y habiendo realizado el procesado previo. La columna 2 muestra la máscara cifrada parcialmente con las mismas llaves, y los mismos bits en la misma posición. La columna 3 muestran el resultado de la prueba.

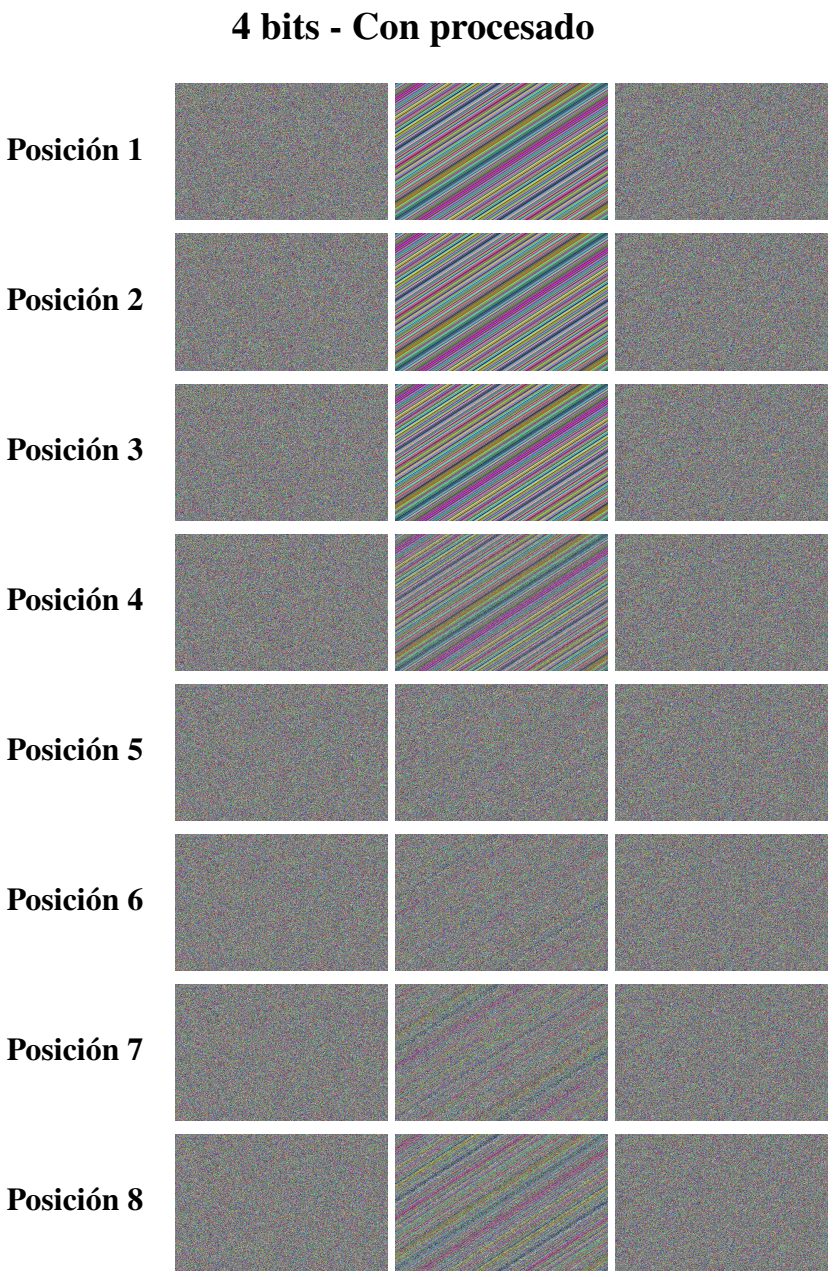

Figura 4.12: La primera columna muestra la imagen cifrada parcialmente, considerando 4 bits y habiendo realizado el procesado previo. La columna 2 muestra la máscara cifrada parcialmente con las mismas llaves, y los mismos bits en la misma posición. La columna 3 muestran el resultado de la prueba.

Capítulo 5

Conclusiones

El objetivo de este trabajo, fue realizar un análisis estadístico de los resultados de la aplicación parcial de un sistema de cifrado a una serie de imágenes. Se presentó el resultado de 128 pruebas aplicadas a 32 imágenes cifradas parcialmente. Al analizar los resultados, de manera general se observa que el cifrado de los bits más significativos proporciona un mayor nivel de seguridad, en comparación a cuando se aplica el cifrado a los bits menos significativos. En el caso del cifrado parcial de 3 bits sin procesado, es claro que únicamente cifrando los bits en la sexta posición se obtiene un buen resultado, ya que siendo visualmente uniforme, la imagen cifrada no revela información alguna ante el ojo humano. Los histogramas muestran un nivel de uniformidad que aumenta conforme la posición los bits cifrados se acerca a los más significativos. De la misma manera los niveles de correlación, incluyendo la correlación adyacente, descienden de manera drástica cuando se cifran dichos bits. Las gráficas obtenidas de estas pruebas muestran linealidad, que se uniformiza al llegar a la sexta posición. Sin embargo, la prueba choosen plain - image attack logró recuperar gran parte de la información original de la imagen. Esta última prueba nos indica que el cifrado parcial de 3 bits sin procesado otorga un nivel poco aceptable de seguridad, ya que si bien otorga una

uniformidad e incomprensibilidad ante el ojo humano, un ataque como CPIA sí revela gran parte de la información.

Para el caso de cifrado parcial de 4 bits sin procesado, se observa que en la quinta y sexta posición se obtiene el mejor resultado en las imágenes cifradas. En ambos casos, estas imágenes son visualmente uniformes y no revelan información alguna ante el ojo humano. Los histogramas para estos cifrados muestran un nivel de uniformidad que aumenta conforme la posición los bits cifrados se acerca a los más significativos, siendo la quinta posición en donde se obtiene la mayor uniformidad. Igualmente los niveles de correlación y correlación adyacente, no muestran ningún patrón de linealidad en las gráficas obtenidas al cifrar los bits más significativos. Sin embargo, la prueba choosen plain - image attack logró revelar gran parte de la información original de la imagen. Esta última prueba indica que el cifrado parcial de 4 bits sin procesado (al igual que en el caso de 3 bits) otorga un nivel poco aceptable de seguridad.

En el cifrado parcial con procesado, se concluye que el procesado otorgó un nivel muy alto de seguridad a las imágenes. Se aprecia fácilmente que tanto en el cifrado parcial de 3 bits como en el de 4 bits, la información obtenida visualmente de las imágenes es nula, sin importar la diferencia en cantidad de bits cifrados, o en la posición. Los histogramas muestran un nivel de uniformidad constante en todos los casos, tanto cifrando los bits menos significativos como los más significativos. De igual manera los niveles de correlación y correlación adyacente son hasta mil veces más bajos que en los casos sin procesar. Las gráficas obtenidas de dichas pruebas no presentan linealidad alguna, siendo completamente uniformes. Cabe destacar que este cifrado parcial superó exitosamente la prueba choosen plain - image

attack. La máscara utilizada para llevar a cabo esta prueba, muestra un patrón visible que se uniformiza conforme avanza la posición de los bits cifrados, siendo la posición sexta y quinta (para el cifrado de 3 y 4 bits respectivamente) las que no muestran ningún patrón. A pesar de esto, la prueba no reveló información visual de la imagen original en ninguno de los casos.

El caso en el que se cifran 3 o 4 bits sin procesado, ofrece un nivel aceptable de seguridad únicamente cifrando los bits más significativos. Sin embargo para fines de alta seguridad, es recomendable realizar el procesado previo a la imagen, para que una vez cifrada parcialmente, no se revele ningún tipo de información visualmente ante un usuario no autorizado, o un ataque. Este trabajo, representa un fundamento cuantitativo en referencia al cifrado parcial del sistema ESCA. Dentro de las posibles aplicaciones del cifrado parcial, se encuentra el cifrado en tiempo real. El hecho de cifrar menos bits representa una disminución de recursos por parte del ordenador, que puede traducirse en un proceso mucho más rápido y eficaz.

Apéndice A

Ecuaciones del sistema ESCA para cifrado total y parcial

A continuación se detallan las ecuaciones del sistema ESCA obtenidas para realizar las operaciones de procesado, cifrado, descifrado y desprocedado, tanto de manera total como parcial. Este último, cifrando 3 y 4 bits, con procesado completo y sin procesado alguno.

Referente al cifrado parcial para 3 y 4 bits, el primer grupo de ecuaciones comienza cifrando los menos significativos, posteriormente el siguiente grupo recorre su posición un bit y así sucesivamente hasta abarcar todas las posiciones realizando tal corrimiento ocho veces. Ya que el procesado se realiza en su totalidad, o no se realiza en lo absoluto, lo único que cambia en las ecuaciones con cifrado parcial son los procesos de cifrado y descifrado.

Ecuaciones para el cifrado completo para L = 8 bits

Procesado	**Desprocesado**
$\hat{m1} = m1 + z2$	$m1 = \hat{m1} + z2$
$\hat{m2} = m2 + z1 + z3$	$m2 = \hat{m2} + z1 + z3$
$\hat{m3} = m1 + m3 + z4$	$m3 = \hat{m1} + \hat{m3} + z2 + z4$
$\hat{m4} = m4 + z1 + z3 + z5$	$m4 = \hat{m4} + z1 + z3 + z5$
$\hat{m5} = m1 + m3 + m5 + z2 + z6$	$m5 = \hat{m3} + \hat{m5} + z2 + z4 + z6$
$\hat{m6} = m2 + m6 + z1 + z5 + z7$	$m6 = \hat{m2} + \hat{m6} + z3 + z5 + z7$
$\hat{m7} = m1 + m5 + m7 + z8$	$m7 = \hat{m1} + \hat{m3} + \hat{m5} + \hat{m7} + z4 + z6 + z8$
$\hat{m8} = m8 + z1 + z5 + z7 + z9$	$m8 = \hat{m8} + z1 + z5 + z7 + z9$
Cifrado	**Descifrado**
$c1 = t1 + t9 + t13 + t15 + \hat{m1}$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$
$c2 = t2 + t10 + t14 + \hat{m2}$	$\hat{m2} = t2 + t10 + t14 + c2$
$c3 = t3 + t11 + t15 + \hat{m1} + \hat{m3}$	$\hat{m3} = t1 + t3 + t9 + t11 + t13 + c1 + c3$
$c4 = t4 + t12 + \hat{m4}$	$\hat{m4} = t4 + t12 + c4$
$c5 = t5 + t13 + \hat{m3} + \hat{m5}$	$\hat{m5} = t1 + t3 + t5 + t9 + t11 + c1 + c3 + c5$
$c6 = t6 + t14 + \hat{m2} + \hat{m6}$	$\hat{m6} = t2 + t6 + t10 + c2 + c6$
$c7 = t7 + t15 + \hat{m1} + \hat{m3} + \hat{m5} + \hat{m7}$	$\hat{m7} = t1 + t5 + t7 + t9 + c1 + c5 + c7$
$c8 = t8 + \hat{m8}$	$\hat{m8} = t8 + c8$

Ecuaciones para la llave con $\mathbf{N = 15}$ bits

$$t1 = x1 + y2$$

$$t2 = x2 + y1 + y3$$

$$t3 = x1 + x3 + y4$$

$$t4 = x4 + y1 + y3 + y5$$

$$t5 = x1 + x3 + x5 + y2 + y6$$

$$t6 = x2 + x6 + y1 + y5 + y6$$

$$t7 = x1 + x5 + x7 + y8$$

$$t8 = x8 + y1 + y5 + y7 + y9$$

$$t9 = x1 + x5 + x7 + x9 + y2 + y6 + y10$$

$$t10 = x2 + x6 + x10 + y1 + y3 + y5 + y9 + y11$$

$$t11 = x1 + x3 + x5 + x9 + x11 + y4 + y12$$

$$t12 = x4 + x12 + y1 + y3 + y9 + y11 + y13$$

$$t13 = x1 + x3 + x9 + x11 + x13 + y2 + y10 + y14$$

$$t14 = x2 + x10 + x14 + y1 + y9 + y13 + y15$$

$$t15 = x1 + x9 + x13 + x15 + y16$$

Ecuaciones para el cifrado parcial para 3 bits - posición 1 a 4

Cifrado	Descifrado
Bit en posición 1	**Bit en posición 1**
$c1 = t1 + t9 + t13 + t15 + \hat{m1}$ $c2 = t2 + t10 + t14 + \hat{m2}$ $c3 = t3 + t11 + t15 + \hat{m1} + \hat{m3}$ $cj = \hat{mj}$ para $4 \leq j \leq 8$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$ $\hat{m2} = t2 + t10 + t14 + c2$ $\hat{m3} = t1 + t3 + t9 + t11 + t13 + c1 + c3$ $\hat{mj} = \hat{mj}$ para $4 \leq j \leq 8$
Bit en posición 2	**Bit en posición 2**
$c1 = m1$ $c2 = t2 + t10 + t14 + \hat{m2}$ $c3 = t3 + t11 + t15 + \hat{m1} + \hat{m3}$ $c4 = t4 + t12 + \hat{m4}$ $cj = \hat{mj}$ para $5 \leq j \leq 8$	$\hat{m1} = \hat{m1}$ $\hat{m2} = t2 + t10 + t14 + c2$ $\hat{m3} = t3 + t11 + t15 + \hat{m1} + c3$ $\hat{m4} = t4 + t12 + c4$ $\hat{mj} = \hat{mj}$ para $5 \leq j \leq 8$
Bit en posición 3	**Bit en posición 3**
$c1 = \hat{m1}$ $c2 = \hat{m2}$ $c3 = t3 + t11 + t15 + \hat{m1} + \hat{m3}$ $c4 = t4 + t12 + \hat{m4}$ $c5 = t5 + t13 + \hat{m3} + \hat{m5}$ $cj = \hat{mj}$ para $6 \leq j \leq 8$	$\hat{m1} = \hat{m1}$ $\hat{m2} = \hat{m2}$ $\hat{m3} = t3 + t11 + t15 + \hat{m1} + c3$ $\hat{m4} = t4 + t12 + c4$ $\hat{m5} = t3 + t5 + t11 + t13 + t15 + \hat{m1} + c3 + c5$ $\hat{mj} = \hat{mj}$ para $6 \leq j \leq 8$
Bit en posición 4	**Bit en posición 4**
$\hat{cj} = \hat{mj}$ para $1 \leq j \leq 3$ $c4 = t4 + t12 + \hat{m4}$ $c5 = t5 + t13 + \hat{m3} + \hat{m5}$ $c6 = t6 + t14 + \hat{m2} + \hat{m6}$ $cj = \hat{mj}$ para $7 \leq j \leq 8$	$\hat{mj} = \hat{mj}$ para $1 \leq j \leq 3$ $\hat{m4} = t4 + t12 + c4$ $\hat{m5} = t5 + t13 + \hat{m3} + c5$ $\hat{m6} = t6 + t14 + \hat{m2} + c6$ $\hat{mj} = \hat{mj}$ para $7 \leq j \leq 8$

Ecuaciones para el cifrado parcial para 3 bits - posición 5 a 8

Cifrado	Descifrado
Bit en posición 5	**Bit en posición 5**
$\hat{cj} = \hat{mj}$ para $1 \leq j \leq 4$ $c5 = t5 + t13 + \hat{m3} + \hat{m5}$ $c6 = t6 + t14 + \hat{m2} + \hat{m6}$ $c7 = t7 + t15 + \hat{m1} + \hat{m3} + \hat{m5} + \hat{m7}$ $c8 = \hat{m8}$	$\hat{mj} = \hat{mj}$ para $1 \leq j \leq 4$ $\hat{m5} = t5 + t13 + \hat{m3} + c5$ $\hat{m6} = t6 + t14 + \hat{m2} + c6$ $\hat{m7} = t1 + t5 + t7 + t9 + c1 + c5 + c7$ $\hat{m8} = \hat{m8}$
Bit en posición 6	**Bit en posición 6**
$\hat{cj} = \hat{mj}$ para $1 \leq j \leq 5$ $c6 = t6 + t14 + \hat{m2} + \hat{m6}$ $c7 = t7 + t15 + \hat{m1} + \hat{m3} + \hat{m5} + \hat{m7}$ $c8 = t8 + \hat{m8}$	$\hat{mj} = \hat{mj}$ para $1 \leq j \leq 5$ $\hat{m6} = t6 + t14 + \hat{m2} + c6$ $\hat{m7} = t1 + t5 + t7 + t9 + c1 + c5 + c7$ $\hat{m8} = t8 + \hat{m8}$
Bit en posición 7	**Bit en posición 7**
$c1 = t1 + t9 + t13 + t15 + \hat{m1}$ $\hat{cj} = \hat{mj}$ para $2 \leq j \leq 6$ $c7 = t7 + t15 + \hat{m1} + \hat{m3} + \hat{m5} + \hat{m7}$ $c8 = t8 + \hat{m8}$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$ $\hat{mj} = \hat{mj}$ para $2 \leq j \leq 6$ $\hat{m7} = t1 + t7 + t9 + t13 + c1 + c7 + \hat{m3} + \hat{m5}$ $\hat{m8} = t8 + \hat{m8}$
Bit en posición 8	**Bit en posición 8**
$c1 = t1 + t9 + t13 + t15 + \hat{m1}$ $c2 = t2 + t10 + t14 + \hat{m2}$ $\hat{cj} = \hat{mj}$ para $3 \leq j \leq 7$ $c8 = t8 + \hat{m8}$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$ $\hat{m2} = t12 + t10 + t14 + c2$ $\hat{mj} = \hat{mj}$ para $3 \leq j \leq 7$ $\hat{m8} = t8 + \hat{m8}$

Ecuaciones para el cifrado parcial para 4 bits - posición 1 a 4

Cifrado	Descifrado
Bit en posición 1	**Bit en posición 1**
$c1 = t1 + t9 + t13 + t15 + \hat{m1}$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$
$c2 = t2 + t10 + t14 + \hat{m2}$	$\hat{m2} = t2 + t10 + t14 + c2$
$c3 = t3 + t11 + t15 + \hat{m1} + \hat{m3}$	$\hat{m3} = t1 + t3 + t9 + t11 + t13 + c1 + c3$
$c4 = t4 + t12 + \hat{m4}$	$\hat{m4} = t4 + t12 + c4$
$cj = \hat{mj}$ para $5 \leq j \leq 8$	$\hat{mj} = \hat{mj}$ para $5 \leq j \leq 8$
Bit en posición 2	**Bit en posición 2**
$c1 = \hat{m1}$	$\hat{m1} = \hat{m1}$
$c2 = t2 + t10 + t14 + \hat{m2}$	$\hat{m2} = t2 + t10 + t14 + c2$
$c3 = t3 + t11 + t15 + \hat{m1} + \hat{m3}$	$\hat{m3} = t3 + t9 + t13 + t15 + \hat{m1}$
$c4 = t4 + t12 + \hat{m4}$	$\hat{m4} = t4 + t12 + c4$
$c5 = t5 + t13 + \hat{m3} + \hat{m5}$	$\hat{m5} = t3 + t5 + t11 + t13 + t15 + c3 + c5 + \hat{m1}$
$cj = \hat{mj}$ para $6 \leq j \leq 8$	$\hat{mj} = \hat{mj}$ para $6 \leq j \leq 8$
Bit en posición 3	**Bit en posición 3**
$c1 = \hat{m1}$	$\hat{m1} = \hat{m1}$
$c2 = \hat{m2}$	$\hat{m2} = \hat{m2}$
$c3 = t3 + t11 + t15 + \hat{m1} + \hat{m3}$	$\hat{m3} = t3 + t11 + t15 + \hat{m1} + c3$
$c4 = t4 + t12 + \hat{m4}$	$\hat{m4} = t4 + t12 + c4$
$c5 = t5 + t13 + \hat{m3} + \hat{m5}$	$\hat{m5} = t3 + t5 + t11 + t13 + t15 + c3 + c5 + \hat{m1}$
$c6 = t6 + t14 + \hat{m2} + \hat{m6}$	$\hat{m6} = t6 + t14 + \hat{m2} + c6$
$cj = \hat{mj}$ para $7 \leq j \leq 8$	$\hat{mj} = \hat{mj}$ para $7 \leq j \leq 8$
Bit en posición 4	**Bit en posición 4**
$\hat{cj} = \hat{mj}$ para $1 \leq j \leq 3$	$\hat{mj} = \hat{mj}$ para $1 \leq j \leq 3$
$c4 = t4 + t12 + \hat{m4}$	$\hat{m4} = t4 + t12 + c4$
$c5 = t5 + t13 + \hat{m3} + \hat{m5}$	$\hat{m5} = t5 + t13 + \hat{m3} + c5$
$c6 = t6 + t14 + \hat{m2} + \hat{m6}$	$\hat{m6} = t6 + t14 + \hat{m2} + c6$
$c7 = t7 + t15 + \hat{m1} + \hat{m3} + \hat{m5} + \hat{m7}$	$\hat{m7} = t5 + t7 + t13 + t15 + \hat{m1} + c5 + c7$
$c8 = \hat{m8}$	$\hat{m8} = \hat{m8}$

Ecuaciones para el cifrado parcial para 4 bits - posición 5 a 8

Cifrado	Descifrado
Bit en posición 5	**Bit en posición 5**
$\hat{cj} = \hat{mj}$ para $1 \leq j \leq 4$ $c5 = t5 + t13 + \hat{m3} + \hat{m5}$ $c6 = t6 + t14 + \hat{m2} + \hat{m6}$ $c7 = t7 + t15 + \hat{m1} + \hat{m3} + \hat{m5} + \hat{m7}$ $c8 = t8 + \hat{m8}$	$\hat{mj} = \hat{mj}$ para $1 \leq j \leq 4$ $\hat{m5} = t5 + t13 + \hat{m3} + c5$ $\hat{m6} = t6 + t14 + \hat{m2} + c6$ $\hat{m7} = t5 + t7 + t13 + t15 + \hat{m1} + c5 + c7$ $\hat{m8} = t8 + \hat{m8}$
Bit en posición 6	**Bit en posición 6**
$c1 = t1 + t9 + t13 + t15 + \hat{m1}$ $\hat{cj} = \hat{mj}$ para $2 \leq j \leq 5$ $c6 = t6 + t14 + \hat{m2} + \hat{m6}$ $c7 = t7 + t15 + \hat{m1} + \hat{m3} + \hat{m5} + \hat{m7}$ $c8 = t8 + \hat{m8}$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$ $\hat{mj} = \hat{mj}$ para $2 \leq j \leq 5$ $\hat{m6} = t6 + t14 + \hat{m2} + c6$ $\hat{m7} = t1 + t7 + t9 + t13 + c1 + c7 + \hat{m3} + \hat{m5}$ $\hat{m8} = t8 + \hat{m8}$
Bit en posición 7	**Bit en posición 7**
$c1 = t1 + t9 + t13 + t15 + \hat{m1}$ $c2 = t2 + t10 + t14 + \hat{m2}$ $\hat{cj} = \hat{mj}$ para $3 \leq j \leq 6$ $c7 = t7 + t15 + \hat{m1} + \hat{m3} + \hat{m5} + \hat{m7}$ $c8 = t8 + \hat{m8}$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$ $\hat{m2} = t2 + t10 + t14 + c2$ $\hat{mj} = \hat{mj}$ para $3 \leq j \leq 6$ $\hat{m7} = t1 + t7 + t9 + t13 + c1 + c7 + \hat{m3} + \hat{m5}$ $\hat{m8} = t8 + \hat{m8}$
Bit en posición 8	**Bit en posición 8**
$c1 = t1 + t9 + t13 + t15 + \hat{m1}$ $c2 = t2 + t10 + t14 + \hat{m2}$ $c3 = t3 + t11 + t15 + \hat{m1} + \hat{m3}$ $\hat{cj} = \hat{mj}$ para $4 \leq j \leq 7$ $c8 = t8 + \hat{m8}$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$ $\hat{m2} = t12 + t10 + t14 + c2$ $\hat{m3} = t1 + t3 + t9 + t11 + t13 + c1 + c3$ $\hat{mj} = \hat{mj}$ para $4 \leq j \leq 7$ $\hat{m8} = t8 + c8$

Ecuaciones para el cifrado parcial sin procesado para 3 bits - posición 1 a 4

Cifrado	Descifrado
Bit en posición 1	**Bit en posición 1**
$c1 = t1 + t9 + t13 + t15 + m1$ $c2 = t2 + t10 + t14 + m2$ $c3 = t3 + t11 + t15 + m1 + m3$ $cj = mj$ para $4 \leq j \leq 8$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$ $\hat{m2} = t2 + t10 + t14 + c2$ $\hat{m3} = t1 + t3 + t9 + t11 + t13 + c1 + c3$ $\hat{mj} = mj$ para $4 \leq j \leq 8$
Bit en posición 2	**Bit en posición 2**
$c1 = m1$ $c2 = t2 + t10 + t14 + m2$ $c3 = t3 + t11 + t15 + m1 + m3$ $c4 = t4 + t12 + m4$ $cj = mj$ para $5 \leq j \leq 8$	$\hat{m1} = m1$ $\hat{m2} = t2 + t10 + t14 + c2$ $\hat{m3} = t3 + t11 + t15 + m1 + c3$ $\hat{m4} = t4 + t12 + c4$ $\hat{mj} = mj$ para $5 \leq j \leq 8$
Bit en posición 3	**Bit en posición 3**
$c1 = m1$ $c2 = m2$ $c3 = t3 + t11 + t15 + m1 + m3$ $c4 = t4 + t12 + m4$ $c5 = t5 + t13 + m3 + m5$ $cj = mj$ para $6 \leq j \leq 8$	$\hat{m1} = m1$ $\hat{m2} = m2$ $\hat{m3} = t3 + t11 + t15 + m1 + c3$ $\hat{m4} = t4 + t12 + c4$ $\hat{m5} = t3 + t5 + t11 + t13 + t15 + m1 + c3 + c5$ $\hat{mj} = mj$ para $6 \leq j \leq 8$
Bit en posición 4	**Bit en posición 4**
$\hat{cj} = mj$ para $1 \leq j \leq 3$ $c4 = t4 + t12 + m4$ $c5 = t5 + t13 + m3 + m5$ $c6 = t6 + t14 + m2 + m6$ $cj = mj$ para $7 \leq j \leq 8$	$\hat{mj} = mj$ para $1 \leq j \leq 3$ $\hat{m4} = t4 + t12 + c4$ $\hat{m5} = t5 + t13 + m3 + c5$ $\hat{m6} = t6 + t14 + m2 + c6$ $\hat{mj} = mj$ para $7 \leq j \leq 8$

Ecuaciones para el cifrado parcial sin procesado para 3 bits - posición 5 a 8

Cifrado	Descifrado
Bit en posición 5	**Bit en posición 5**
$\hat{cj} = mj$ para $1 \leq j \leq 4$ $c5 = t5 + t13 + m3 + m5$ $c6 = t6 + t14 + m2 + m6$ $c7 = t7 + t15 + m1 + m3 + m5 + m7$ $c8 = m8$	$\hat{mj} = mj$ para $1 \leq j \leq 4$ $\hat{m5} = t5 + t13 + m3 + c5$ $\hat{m6} = t6 + t14 + m2 + c6$ $\hat{m7} = t5 + t7 + t13 + t15 + m1 + c5 + c7$ $\hat{m8} = m8$
Bit en posición 6	**Bit en posición 6**
$\hat{cj} = mj$ para $1 \leq j \leq 5$ $c6 = t6 + t14 + m2 + m6$ $c7 = t7 + t15 + m1 + m3 + m5 + m7$ $c8 = t8 + m8$	$\hat{mj} = mj$ para $1 \leq j \leq 5$ $\hat{m6} = t6 + t14 + m2 + c6$ $\hat{m7} = t7 + m1 + m3 + m5 + c7 + t15$ $\hat{m8} = t8 + m8$
Bit en posición 7	**Bit en posición 7**
$c1 = t1 + t9 + t13 + t15 + m1$ $\hat{cj} = mj$ para $2 \leq j \leq 6$ $c7 = t7 + t15 + m1 + m3 + m5 + m7$ $c8 = t8 + m8$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$ $\hat{mj} = mj$ para $2 \leq j \leq 6$ $\hat{m7} = t1 + t7 + t9 + t13 + c1 + c7 + m3 + m5$ $\hat{m8} = t8 + m8$
Bit en posición 8	**Bit en posición 8**
$c1 = t1 + t9 + t13 + t15 + m1$ $c2 = t2 + t10 + t14 + m2$ $\hat{cj} = mj$ para $3 \leq j \leq 7$ $c8 = t8 + m8$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$ $\hat{m2} = t12 + t10 + t14 + c2$ $\hat{mj} = mj$ para $3 \leq j \leq 7$ $\hat{m8} = t8 + m8$

Ecuaciones para el cifrado parcial sin procesado para 4 bits - posición 1 a 4

Cifrado	Descifrado
Bit en posición 1	**Bit en posición 1**
$c1 = t1 + t9 + t13 + t15 + m1$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$
$c2 = t2 + t10 + t14 + m2$	$\hat{m2} = t2 + t10 + t14 + c2$
$c3 = t3 + t11 + t15 + m1 + m3$	$\hat{m3} = t1 + t3 + t9 + t11 + t13 + c1 + c3$
$c4 = t4 + t12 + m4$	$\hat{m3} = t4 + t12 + c4$
$cj = mj$ para $5 \leq j \leq 8$	$\hat{mj} = mj$ para $5 \leq j \leq 8$
Bit en posición 2	**Bit en posición 2**
$c1 = m1$	$\hat{m1} = m1$
$c2 = t2 + t10 + t14 + m2$	$\hat{m2} = t2 + t10 + t14 + c2$
$c3 = t3 + t11 + t15 + m1 + m3$	$\hat{m3} = t3 + t9 + t13 + t15 + m1$
$c4 = t4 + t12 + m4$	$\hat{m4} = t4 + t12 + c4$
$c5 = t5 + t13 + m4 + m5$	$\hat{m5} = t3 + t5 + t11 + t13 + t15 + c3 + c5 + m1$
$cj = mj$ para $6 \leq j \leq 8$	$\hat{mj} = mj$ para $6 \leq j \leq 8$
Bit en posición 3	**Bit en posición 3**
$c1 = m1$	$\hat{m1} = m1$
$c2 = m2$	$\hat{m2} = m2$
$c3 = t3 + t11 + t15 + m1 + m3$	$\hat{m3} = t3 + t11 + t15 + m1 + c3$
$c4 = t4 + t12 + m4$	$\hat{m4} = t4 + t12 + c4$
$c5 = t5 + t13 + m3 + m5$	$\hat{m5} = t3 + t5 + t11 + t13 + t15 + c3 + c5 + m1$
$c6 = t6 + t14 + m2 + m6$	$\hat{m6} = t6 + t14 + m2 + c6$
$cj = mj$ para $7 \leq j \leq 8$	$\hat{mj} = mj$ para $7 \leq j \leq 8$
Bit en posición 4	**Bit en posición 4**
$\hat{cj} = mj$ para $1 \leq j \leq 3$	$\hat{mj} = mj$ para $1 \leq j \leq 3$
$c4 = t4 + t12 + m4$	$\hat{m4} = t4 + t12 + c4$
$c5 = t5 + t13 + m3 + m5$	$\hat{m5} = t5 + t13 + m3 + c5$
$c6 = t6 + t14 + m2 + m6$	$\hat{m6} = t6 + t14 + m2 + c6$
$c7 = t7 + t15 + m1 + m3 + m5 + m7$	$\hat{m7} = t5 + t7 + t13 + t15 + m1 + c5 + c7$
$c8 = m8$	$\hat{m8} = m8$

Ecuaciones para el cifrado parcial sin procesado para 4 bits - posición 5 a 8

Cifrado	Descifrado
Bit en posición 5	**Bit en posición 5**
$\hat{cj} = mj$ para $1 \leq j \leq 4$ $c5 = t5 + t13 + m3 + m5$ $c6 = t6 + t14 + m2 + m6$ $c7 = t7 + t15 + m1 + m3 + m5 + m7$ $c8 = t8 + m8$	$\hat{mj} = mj$ para $1 \leq j \leq 4$ $\hat{m5} = t5 + t13 + m3 + c5$ $\hat{m6} = t6 + t14 + m2 + c6$ $\hat{m7} = t5 + t7 + t13 + t15 + m1 + c5 + c7$ $\hat{m8} = t8 + m8$
Bit en posición 6	**Bit en posición 6**
$c1 = t1 + t9 + t13 + t15 + m1$ $\hat{cj} = mj$ para $2 \leq j \leq 5$ $c6 = t6 + t14 + m2 + m6$ $c7 = t7 + t15 + m1 + m3 + m5 + m7$ $c8 = t8 + m8$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$ $\hat{mj} = mj$ para $2 \leq j \leq 5$ $\hat{m6} = t6 + t14 + m2 + c6$ $\hat{m7} = t1 + t7 + t9 + t13 + c1 + c7 + m3 + m5$ $\hat{m8} = t8 + m8$
Bit en posición 7	**Bit en posición 7**
$c1 = t1 + t9 + t13 + t15 + m1$ $c2 = t2 + t10 + t14 + m2$ $\hat{cj} = mj$ para $3 \leq j \leq 6$ $c7 = t7 + t15 + m1 + m3 + m5 + m7$ $c8 = t8 + m8$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$ $\hat{m2} = t2 + t10 + t14 + c2$ $\hat{mj} = mj$ para $3 \leq j \leq 6$ $\hat{m7} = t1 + t7 + t9 + t13 + c1 + c7 + m3 + m5$ $\hat{m8} = t8 + m8$
Bit en posición 8	**Bit en posición 8**
$c1 = t1 + t9 + t13 + t15 + m1$ $c2 = t2 + t10 + t14 + m2$ $c3 = t3 + t11 + t15 + m1 + m3$ $\hat{cj} = mj$ para $4 \leq j \leq 7$ $c8 = t8 + m8$	$\hat{m1} = t1 + t9 + t13 + t15 + c1$ $\hat{m2} = t12 + t10 + t14 + c2$ $\hat{m3} = t1 + t3 + t9 + t11 + t13 + c1 + c3$ $\hat{mj} = mj$ para $4 \leq j \leq 7$ $\hat{m8} = t8 + c8$

Apéndice B

Resto de Imágenes cifradas

A continuación se presentan 2 casos más de imágenes que fueron cifradas parcialmente. Se optó por solo presentar el resultado de las pruebas aplicadas a una sola imagen, e incluir en este apéndice los otros 2 casos.

Estos 2 corresponden a las imágenes de prueba del mandril y la de pimientos. Ambas imágenes son utilizadas frecuentemente en el procesado de imágenes debido a que son ópticamente activas. Asimismo el contraste entre texturas, patrones y colores, las hizo adecuadas para su uso dentro del cifrado.

B.1. Segundo paquete de imágenes cifradas

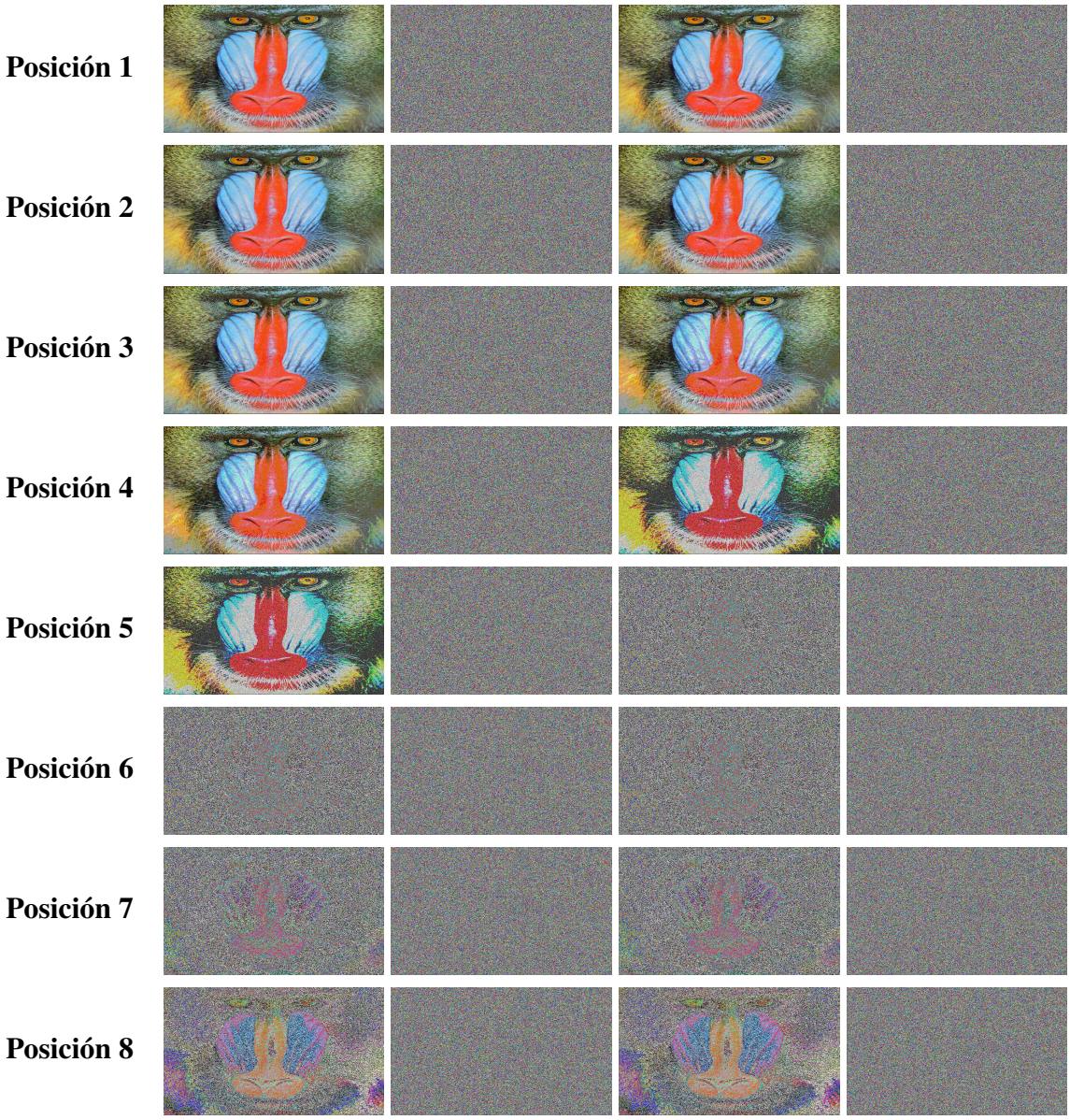

Figura B.1: La primera columna muestra la imagen cifrada. Las columnas 2, 3 y 4 muestran los histogramas Rojo, Verde y Azul, respectívamente. Las fila 1 corresponde al cifrado parcial realizado en los 3 bits menos significativos. El orden cambia de manera descendente.

B.2. Tercer paquete de imágenes cifradas

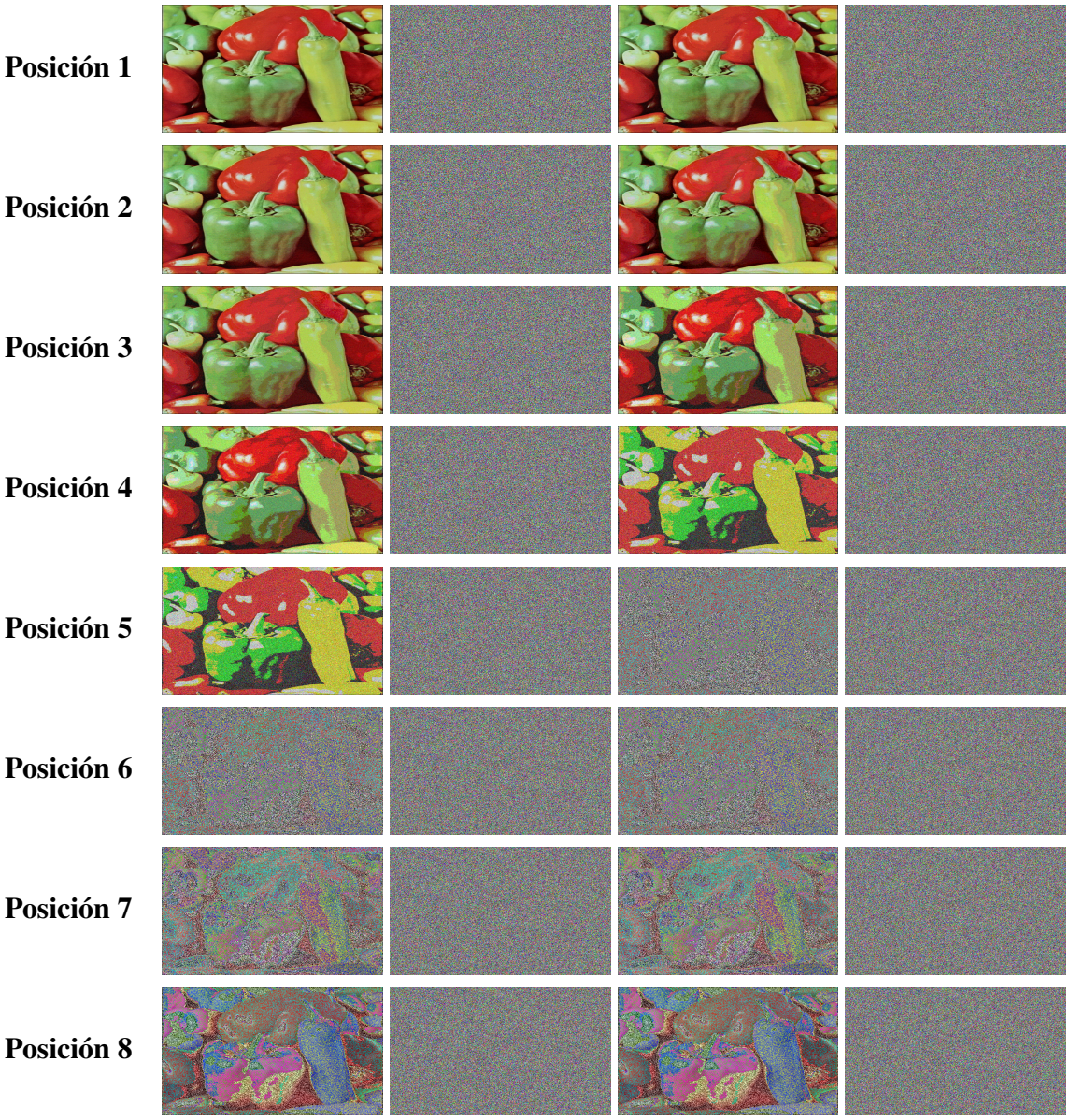

Figura B.2: La primera columna muestra la imagen cifrada. Las columnas 2, 3 y 4 muestran los histogramas Rojo, Verde y Azul, respectívamente. Las fila 1 corresponde al cifrado parcial realizado en los 3 bits menos significativos. El orden cambia de manera descendente.

Bibliografía

[1] Diffie Whitfield, *The first ten years of public-jey cryptography*, Proceedings of the IEEE,**75**, no. 5, May (1998)

[2] Mejía Carlos M., *Encriptación por Sincronización en Autómatas Celulares*,**1**, Dic (2001)

[3] M. T. Ramírez-Torres, J. S. Murguía and M. Mejía Carlos, *Image Encryption with an Improved Cryptosystem based on a Matrix Approach*, International Journal of Modern Physics C, Vol. 25, No. 10, article 1450054 (16 pages), 2014. ISSN: 0129-1831. DOI: 10.1142/S0129183114500545.

[4] Reyes Gómez D. A., *Descripción y Aplicaciones de los Autómatas Celulares*, Aug (2011)

[5] Stephen Wolfram, *Statistical mechanics of cellular automata.", Reviews of Modern Physiscs*, **55** 1983

[6] National Instruments, *http://www.ni.com/es-mx.html*

[7] A. Martín del Rey, G. Rodríguez Sanchez and A. de la Villa Cuenca, Hybrid Artif. Intell. Syst. Lecture Notes Comput. Sci. 7209, 78 (2012).